Universidad Nacional de Córdoba

Facultad de Ciencias Exactas Físicas y Naturales

Tesis Doctoral

Diseño de Plan de Contactos para Redes Satelitales Tolerantes a Disrupciones

Autor: Ing. Juan A. Fraire
Director: Dr. Pablo A. Ferreyra

Noviembre 2015

Diseño de Plan de Contactos para Redes Satelitales Tolerantes a Disrupciones

por

Ing. Juan A. Fraire

Dr. Pablo A. Ferreyra

Director

COMISIÓN ASESORA

Dr. Pablo A. Ferreyra
UNC

Dr. Jorge M. Finochietto
UNC-CONICET

Dr. Mario Hueda
UNC-CONICET

Esta Tesis fue enviada a la Facultad de Ciencias Exactas Físicas y Naturales de la Universidad Nacional de Córdoba para cumplimentar los requerimientos de obtención del grado académico de Doctor en Ciencias de la Ingeniería.

Córdoba, Argentina

Noviembre 2015

UNIVERSIDAD NACIONAL DE CORDOBA
Facultad de Cs. Exactas, Fisicas y Naturales

ACTA DE EXAMENES

Libro: 00001 Acta: 03250 Hoja 01/01
LLAMADO: 1 28/10/2015
CATEDRA - MESA:

DI002 TESIS DOCTORADO EN CIENCIAS DE LA INGENIERIA

NUMERO	APELLIDO Y NOMBRE	DOCUMENTO	INGRESO	COND.	NOTA	FIRMA
30968992	FRAIRE, Juan Andrés	DNI: 30968992	2012	T	APROBADO	

ALVAREZ - HAMELIN, José - LEGIZAMÓN, Mario G. - D'ARGENIO, Pedro - AGUIRRE, Nazareno - CORRAL BRI

Observaciones:

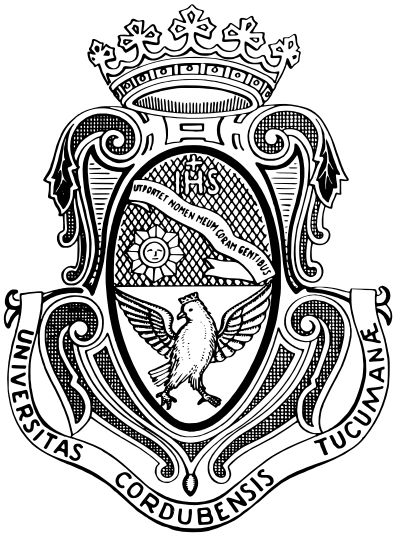
ALVAREZ HAMELIN J. Ignacio

Córdoba, ____/_____/_____-.
Certifico que la/s firma/s que ha/n sido puesta/s en la presente Acta pertenece/n a:_____

Inscriptos	Ausentes	Examinados	Reprobados (0-3)	Aprobados (4-10)
1	—	1	—	1

27/10/2015 09:08:06 Libro/Acta: 0000103250 Hoja: 01/ 01

Universidad Nacional de Córdoba



Tesis Doctoral

Diseño de Plan de Contactos para Redes Satelitales Tolerantes a Disrupciones

Autor:
Juan Andrés Fraire

Director:
Dr. Pablo Alejandro Ferreyra

Asesores:
Dr. Jorge M. Finochietto, Dr. Mario Hueda

Tesis para la Carrera de Doctorado en Ciencias de la Ingeniería

Laboratorio de Comunicaciones Digitales
Facultad de Ciencias Exactas Físicas y Naturales

10 de noviembre de 2015

"No repases tanto, Lobsang, que así se te atasca la memoria. Tienes que estar absolutamente tranquilo, como lo estás ahora, y verás cómo te brota el conocimiento."

T. Lobsang Rampa, El Tercer Ojo

UNIVERSIDAD NACIONAL DE CÓRDOBA

Resumen en Español

Universidad Nacional de Córdoba
Facultad de Ciencias Exactas Físicas y Naturales

Doctorado en Ciencias de la Ingeniería

Diseño de Plan de Contactos para Redes Satelitales Tolerantes a Disrupciones

por Juan Andrés Fraire

En los últimos 20 años, la industria de las comunicaciones satelitales ha mostrado un avance limitado en comparación con la evolución del fenómeno de Internet en la Tierra. Sin embargo, recientemente, y producto de un esfuerzo conjunto de diferentes agencias espaciales (NASA, ESA, CONAE, etc.), se ha empezado a estudiar y experimentar con estrategias de comunicaciones en red que toleren disrupciones o cortes, los cuales resultan inherentes a cualquier sistema distribuido espacial. En general, los protocolos existentes de Internet no toleran conexiones esporádicas y disruptivas por lo que no han podido ser considerados seriamente para su aplicación en redes de constelaciones de satélites. En efecto, estas redes se han clasificado y estudiado bajo el nombre de Redes Tolerantes a Disrupciones (DTN), y han sido objeto de recientes investigaciones de la comunidad científica. En particular, aquellas DTNs aplicadas al ámbito espacial, pueden aprovechar un *plan de contacto* compuesto por una lista de las próximas oportunidades de comunicación (*contactos*) para tomar decisiones eficientes de cómo y cuándo transmitir el tráfico generado o recibido desde otro satélite.

En general, y a pesar de recientes avances y exitosos experimentos en órbita, la comunidad de DTN ha asumido que todos los contactos en el plan de contactos pueden ser utilizados, lo cual en esta tesis es cuestionado con claros fundamentos basados en las limitaciones de recursos bajo los cuales suelen operar estas plataformas satelitales. En consecuencia, definimos y tratamos el problema de *diseño de plan de contactos* como el proceso de configurar y elegir apropiadamente estas oportunidades de transferencia de información de antemano con el fin de gestionar la utilización del sistema de acuerdo a las características de sus recursos, y al mismo tiempo, optimizarlo bajo un criterio dado para mejorar el flujo de datos (generalmente hacia la tierra) tanto de instrumentos científicos como de telemetría de la plataforma.

Sin embargo, y en general, debido a la gran cantidad de alternativas, variables, consideraciones, y restricciones involucradas, ejecutar este proceso de diseño de forma manual resulta imposible inclusive para los operadores de red mas expertos en los escenarios mas simples. De esta manera, en esta investigación aportamos un enfoque original en el que aprovechamos la propiedad de predictibilidad de todos estos fenómenos para proveer mecanismos de gestión, control, y toma de decisiones automatizados y optimizados para un problema cuya complejidad incrementa drásticamente con la cantidad de satélites, tiempo de evaluación, y otros componentes que descubriremos a lo largo del trabajo.

Con el fin de enfrentar un problema de optimización de esta magnitud, adoptamos una estrategia de inclusión progresiva de información derivando en propuestas de diseño incrementales en términos de complejidad y eficiencia, aunque aplicables en diferentes contextos de misión. En este sentido, se proponen mecanismos originales como FCP, RACP, y TACP basados en la combinación de técnicas de modelado lineal de enteros mixtos, algorítmicas, y de aproximación metaheurística como recocido simulado y esquemas genéticos. Finalmente, con el conocimiento obtenido desde la planificación óptima, hemos sido capaces de realizar otras importantes mejoras en el área de la implementación o aplicación de los planes de contacto para el cálculo de rutas eficientes por medio de procedimientos innovadores como C-CGR, PA-CGR y MG-CGR.

En general, estos métodos han recibido la aprobación e interés de la comunidad DTN por medio de publicaciones en revistas de alto impacto y congresos internacionales así como su implementación en simuladores y librerías de software diseñadas para su aplicación en un posible centro de control de misión satelital de CONAE. En efecto, el avance aquí logrado constituye un importante paso a la consolidación y consideración de las redes DTN para su efectiva aplicación en misiones operativas reales. Esto resulta de particular interés en el marco del Plan Espacial Argentino dirigido por CONAE donde el proyecto de Arquitectura Segmentada emerge como un claro, aunque no necesariamente el único, beneficiario del paradigma de gestión de redes espaciales generado en este trabajo doctoral.

UNIVERSIDAD NACIONAL DE CÓRDOBA

Resumen en Inglés
-Abstract-

Universidad Nacional de Córdoba
Facultad de Ciencias Exactas Físicas y Naturales

PhD. in Engineering Sciences

Contact Plan Design for Disruption Tolerant Satellite Networks

by Juan Andrés Fraire

The space industry have shown limited progress in comparison to Internet-based networks on Earth. Nonetheless, recent efforts from different space agencies (NASA, ESA, CONAE, etc.) have pushed the study and experimentation with networked communications strategies that could cope with the typical disruptions of spaceborne distributed systems. In general, existing Internet protocols fail to perform in an environment with sporadic link. In particular, these networks have been classified under the name of Disruption Tolerant Networks (DTN) by the scientific community. When DTNs are applied on space assets, they can take advantage of a *contact plan* comprised by a list of the forthcoming network communications opportunities (i.e. *contacts*) in order to take efficient routing decisions. Indeed, in DTN, routing implies to determine when and how to forward the information.

In general, and despite recent advances and successful in-orbit demonstrations, DTN community has assumed that all contacts in the contact plan can be used, which we call into question with sufficient argumentation based on the resource limitations on which space platforms usually operate. As a result, we define and tackle the *contact plan design problem* as the process of appropriately configuring and choosing among these communication opportunities in advance. Indeed, this allows to optimize the final plan with a certain criteria in order to improve data flow (generally towards Earth) both from on-board scientific instruments and the platform telemetry.

However, and in general, due to the increasing quantity of alternatives, variables, considerations and restrictions involved in the design process, the manual execution of the latter quickly becomes intractable even for the most experienced network operators in the simplest scenarios. Therefore, in this thesis we investigate and contribute with an original focus in which the predictability property of the spaceborne DTN system is exploited so as to derive optimized management, control, and decision-taking procedures. In fact, we prove the latter allows to solve a problem whose complexity drastically increases with the quantity of satellites, evaluation time, and other factors we discover along the present work.

With the end of tackling such a complex restricted optimization problem, we adopt an incremental strategy where the input information is accumulated through the evolution of different contact plan design procedures which indeed grow in terms of both complexity and efficiency. As a result, we contribute with several original mechanisms such as FCP, RACP, and TACP based on a combination of mixed integer linear programming modeling, algorithmic approaches, and metaheuristic techniques such as simulated annealing and evolutionary algorithms. Finally, with the knowledge obtained from this scheduling procedures, we were able to improve the state of the art of implementation and application of the designed contact plans for their usage in efficient route calculation by contributing with C-CGR, PA-CGR, and MG-CGR.

In general, these methods have received a wide acceptance and interest from the DTN community by means of journals and magazines publications and papers in international conferences as well as the implementation in software simulators and libraries designed to be included in a mission operations and control center (MOC) in the near future. Indeed, the contributions described throughout this thesis are an important step towards the consolidation and consideration of the application of DTN networks in real operative missions. In particular, this outcome is of specific interest in the context of the National Space Plan executed by the Argentina Space Agency (CONAE) where the Segmented Architecture emerge as a clear, though not necessarily the only beneficiary of the management paradigm derived from this PhD. thesis.

UNIVERSIDAD NACIONAL DE CÓRDOBA

Resumen en Portugués

-Resumo-

Universidad Nacional de Córdoba
Facultad de Ciencias Exactas Físicas y Naturales

PhD. em Ciências da Engenharia

Projecto de Plano de Contato para Redes de Satelites Tolerates as Interrupções

de Juan Andrés Fraire

A indústria espacial têm mostrado progressos limitados em comparação com redes baseadas em Internet na Terra. Porém, os recentes esforços de diferentes agências especiais (NASA, ESA, CONAE, etc.) promoveram o estudo e a experimentação das estratégias de comunicação em rede que poderia lidar com as interrupções típicas dos sistemas espaciais distribuídas. Em geral, os protocolos de Internet existentes não conseguem executar em um ambiente com ligação esporádica. Em particular, essas redes têm sido clasificadas con o nome de redes tolerantes interrupções ou em Inglês Delay Tolerant Network (DTN) por parte da comunidade científica. Quando DTNs são aplicados em itens do espaço, eles podem tirar proveito de um *plano de contato* composta por uma lista de oportunidades de comunicações de redes próximas (ou seja, *contatos*), ao fim de tomar decisões de roteamento eficientes. De fato, em DTN, roteamento implica determinar quando e como encaminhar as informações.

Em geral, e apesar dos avanços recentes e bem sucedidas demonstrações em órbita, a comunidade DTN assumiu que todos os contatos no plano de contato pode ser usado, coisa que nois questionamos com a argumentação suficiente com base nas limitações dos recursos em que plataformas espaciais operam normalmente. Como resultado, nós definir e resolver o *problema de Projecto de plano de contato* como o processo de configurar adequadamente e escolher entre isas oportunidades de comunicação com antecedência. Na verdade, o que permite optimizar o plano final com um determinado critério, a fim de melhorar o fluxo de dados (geralmente em direção à Terra), tanto de instrumentos científicos a bordo ea telemetria de plataforma.

No entanto, e em geral, devido à quantidade cada vez maior de alternativas, variáveis, considerações e restrições envolvidas no processo de concepção, a execução manual do último rapidamente se torna intratável mesmo para os operadores de rede mais experientes nos cenários mais simples. Portanto, nesta tese, investigar e contribuir com um foco original no qual a propriedade e previsibilidade do sistema DTN é explorada de modo a obter uma gestão optimizada, controle e processos de tomada de decisão. Na verdade, nós testámos que este último permite resolver um problema cuja complexidade aumenta drasticamente com a quantidade de satélites, o tempo de avaliação, e outros fatores que descobrimos ao longo do presente trabalho.

Com o fim de combater tal problema de otimização complexo com restrições, adotamos uma estratégia gradual onde a informação de entrada é acumulada através da evolução dos diferentes procedimentos de projeto de plano de contato que de fato crescer tanto em termos de complexidade e eficiência. Como resultado, nós contribuímos com vários mecanismos originais, tais como FCP, RACP e TACP com base em uma combinação de modelagem e programação linear inteira mista, abordagens algorítmicas, e técnicas metaheurísticas, tais como recozimento simulado e os algoritmos evolucionários. Finalmente, com o conhecimento obtido a partir deste procedimento de agendamento, fomos capazes de melhorar o estado da arte da execução e aplicação dos planos de contacto projetados para seu uso no cálculo da rota eficiente, contribuindo com C-CGR, PA-CGR, e MG-CGR.

Em geral, estes métodos têm recebido uma grande aceitação e interesse da comunidade DTN através de jornais e revistas de publicações e artigos em conferências internacionais, bem como a implementação em simuladores de software e bibliotecas projetadas para ser incluído em uma operação e controle de missão central (MOC) num futuro próximo. Na verdade, as contribuições descritas ao longo desta tese são um passo importante para a consolidação e análise do pedido de redes DTN em missões operacionais reais. Em particular, este resultado é de interesse específico no contexto do Plano Espacial Nacional executado pela Agência Espacial Argentina (CONAE), onde a Arquitetura Segmentada emerge como uma clara, embora não necessariamente o único beneficiário do paradigma de gestão derivada desta PhD. tese.

Agradecimientos

Sin dudas el principal agradecimiento va dirigido hacia Pablo Ferreyra y Jorge Finochietto quienes supieron generar el contexto adecuado tanto personal y humano como técnico, profesional y académico para poder desarrollar y guiar la actividad de investigación aquí resumida.

En general, resulta sumamente gratificante poder generar un aporte de este tipo e importancia con impacto en proyectos de relevancia nacional e internacional como los que Oscar Ignazi me ha permitido participar y liderar desde Servicios Tecnológicos Integrados (STI), la compañía de la cual formo parte. Gracias a él, Gaspar Pollano y Maxi Gusella hemos generado un vínculo empresa-academia ejemplar mediante la integración de la investigación como actividad conductora de proyectos tecnológicos de alta complejidad. Entre estos, el programa SARE originado en la Agencia Espacial Argentina (CONAE) por Alberto Ridner, adoptado y apoyado por Conrado Varotto, e inicialmente desarrollado por STI junto al Laboratorio de Comunicaciones Digitales (LCD) de la Universidad Nacional de Córdoba (UNC), resulta la fuente de inspiración y objeto final, aunque no exclusivo, de este trabajo.

En este contexto, agradezco a mis colegas y amigos de STI en este emprendimiento como Esteban Kocian, Nicolás Alvarez, Renzo Abdala, Lucas Gabutti, Ricardo Micarelli, Nicolás Casco, Mariano Volarik, Nicolás Tomatis y muchos otros de los cuales no dejo de aprender diariamente. Además, esta tesis forma una pequeña parte y se suscita gracias a la gran contribución y aporte del grupo del LCD formado por Graciela Briones, Mario Hueda, Pablo Madoery, Guillermo Riva, Juan Leal Licudis, Sergio Baudino, Fede Baldoni, Renato Cherini, Luis Rodriguez, Diego Murature, Nehuén Gonzalez, Horacio Medoza, Martín Ayarde, Toni Abdala entre otros excelentes ingenieros, investigadores y docentes.

En el plano personal no puedo dejar de agradecer a mi compañera Paula Sciolla quien además de ilustrarme permanentemente en diferentes aspectos de la vida, probablemente fue la persona que mas tiempo pasó a mi lado (mate de por medio) en el proceso de investigación. En este sentido también agradezco a mi querida familia, a mi Madre, Ricardo, Alejandro y Gabriela Aguirre y sus parejas Gisella y Alexis, a la siempre presente y cálida familia Sciolla, así como a mis amigos de la *Vecindad*, *Colinas*, el *IUA*, *Teatro*, y de la *Comparsa Afro* con su alma en el mundo del arte y la música quienes desde su correspondiente lugar han formado parte del proceso doctoral aquí manifestado.

Índice General

Índice de Figuras

Índice de Tablas

Índice de Algoritmos

Glosario

ACO	**A**nt **C**olony **O**ptimization
AMS	**A**synchronous **M**essage **S**ervice
AOCS	**A**ttitude and **O**rbit **C**ontrol **S**ystem
AS	**A**rquitectura **S**egmentada
BER	**B**it **E**rror **R**ate
BJI	**B**est **J**ain **I**ndex
BMD	**B**est **M**ean **D**elay
BP	**B**undle **P**rotocol
BSP	**B**erkeley **S**oftware **D**istribution
BUT	**B**est **U**nrouted **T**ime
CBHE	**C**ompressed **B**undle **H**eader **E**ncoding
CCGR	**C**ache-based **C**ontact **G**raph **R**outing
CCSDS	**C**onsultative **C**ommittee for **S**pace and **D**ata **S**ystems
CETD	**C**ost **E**ffective **T**opology **D**esign
CFDP	**C**CSDS **F**ile **D**elivery **P**rotocol
CGR	**C**ontact **G**raph **R**outing
CJI	**C**urrent **J**ain **I**ndex
CL	**C**ontact **L**ist
CLA	**C**onvergence **L**ayers **A**dapters
CMD	**C**urrent **M**ean **D**elay
CRC	**C**iclyc **R**edundancy **C**ode
CONAE	**CO**misión **N**acional de **A**ctividades **E**spaciales
COTS	**C**ommertial **O**f de **T**he **S**helf
CPD	**C**ontact **P**lan **D**esign
CPUP	**C**ontact **P**lan **U**pdate **P**rotocol

CRC	**C**oncurrent **R**esources **C**onstraints
CRP	**C**ontact **R**eview **C**rocedure
CSMA	**C**arrier **S**ense **A**ccess
CUT	**C**urrent **U**nrouted **T**ime
DARPA	**D**efense **A**dvanced **R**esearch **P**rojects **A**gency
DINET	**D**eep **I**mpact **N**etwork **E**xperiment
DMA	**D**irect **M**emory **A**ccess
DMC	**D**isaster **M**onitoring **C**onstellation
DSN	**D**eep **S**pace **N**etwork
DTN	**D**elay/Disruption **T**olerant **N**etworks
DTNRG	**D**elay/Disruption **T**olerant **N**etworks **R**earch **G**roup
DTNWG	**D**elay/Disruption **T**olerant **N**etworks **W**orking **G**roup
EA	**E**volutionary **A**lgorithms
EBCGR	**E**xtension **B**lock **CGR**
ECI	**E**arth **C**enter **I**nertial
ESA	**E**uropean **S**pace **A**gency
ESL	**E**arth **S**atellite **L**ink
F6	**F**uture **F**ast **F**lexible **F**ractionated **F**ree-**F**lying
FBP	**F**orward **B**undle **P**rocedure
FI	**F**irst **I**mprovement
FIFO	**F**irst **I**n **F**irst **O**ut
FSM	**F**inite **S**tate **M**achine
FPGA	**F**ield **P**rogramable **G**ate **A**rray
GEO	**G**eoestationary **E**arth **O**rbit
HPOP	**H**igh **P**erformance **O**rbital **P**ropagator
HTTP	**H**yper **T**ext **T**ransfer **P**rotocol
IBR	**I**nstitut **B**etriebssysteme **R**echnerverbund
ION	**I**nterplanetary **O**verlay **N**etwork
IP	**I**nternet **P**rotocol
IPNSIG	**I**nterplanetary **N**etworking **S**pecial **I**nterest **G**roup
ISL	**I**nter **S**atellite **L**ink
ISS	**I**nternational **S**pace **S**tation
JHUAPL	**J**ohns **H**opkins **U**niversity **A**pplied **P**hysics **L**aboratory

JPL	**J**et **P**ropoulsion **L**aboratory
LD	**L**ink **D**ensity
LEO	**L**ow **E**arth **O**rbit
LOS	**L**ine **O**f **S**ight
LTP	**L**icklider **T**ransmission **P**rotocol
MAC	**M**edium **A**ccess **C**ontrol
MCP	**M**ax Capacity **C**ontact **P**lan
MEO	**M**edium **E**arth **O**rbit
MFW	**M**erugu's **F**loyd **W**arshall
MG-CGR	**M**ulti **G**raph **C**GR
MILP	**M**ixed **I**nteger **L**inear **P**rogramming
MOC	**M**ission **O**perations and **C**ontrol Center
NASA	**N**ational **A**eronautic and **S**pace **A**dministration
PCC	**P**redictive **C**apacity **C**onsumption
PEN	**P**lan **E**spacial **N**acional
PEP	**P**erformance **E**nhancing **P**roxies
POSIX	**P**ortable **O**perating **S**ystem **I**nterfaces para **U**nix
RAAN	**R**ight **A**scension of the **A**scending **N**ode
RACP	**R**oute **A**ware **C**ontact **P**lan
RFC	**R**equest **F**or **C**omments
RTT	**R**ound **T**trip **T**ime
SA	**S**imulated **A**nnealing
SCF	**S**tore **C**arry and **F**orward
SD	**S**teepest **D**escent
SDNV	**S**elf **D**elimiting **N**umeric **V**alues
SMS	**S**hort **M**essage **S**ervice
SOC	**S**ystem **O**n **C**hip
SSN	**S**pace **S**ensor **N**etwork
SSTL	**S**urrey **S**atellite **T**echnology **L**td
STI	**S**ervicios **T**ecnológicos **I**ntegrados
STR	**S**pace **T**ime **R**outing
TACP	**T**raffic **A**ware **C**ontact **P**lan
TACP-LP	**T**raffic **A**ware **C**ontact **P**lan **L**inear **P**rogramming

TACP-GA	**T**raffic **A**ware **C**ontact **P**lan **G**enetic **A**lgorithm
TCP	**T**ransfer **C**ontrol **P**rotocol
TZC	**T**ime **Z**one **C**onstraints
UDP	**U**ser **D**atagram **P**rotocol
UNC	**U**niversidad **N**acional de **C**órdoba
ZCO	**Z**ero **C**opy de **O**bject

En la memoria de Daniel

Capítulo 1

Introducción y Marco Conceptual

1.1. Introducción

1.1.1. Problema de Investigación

En esta tesis trataremos la problemática de diseñar e implementar las oportunidades de comunicación (*contactos*) de una red de datos cuyos enlaces varían de manera predecible en el tiempo como es el caso de las comunicaciones de constelaciones satelitales. Dado que los protocolos existentes de Internet asumen conectividad permanente (las desconexiones se consideran un caso de error), este tipo de sistemas se han clasificado aparte bajo el término redes tolerantes a demoras o disrupciones (DTN). En general, las redes DTN han tomado una relevancia significativa en los últimos años por los beneficios operativos que pueden brindar en entornos extremos como el espacial. Sin embargo, y de acuerdo al estado del arte descrito en este capítulo, la comunidad ha asumido que los nodos disponen de recursos ilimitados al poder implementar todos los contactos venideros, suposición que en esta tesis ponemos en tela de juicio. Invalidar esta hipótesis deriva en la necesidad de configurar y elegir apropiadamente estos contactos de antemano, lo que permite adaptar la utilización del sistema de acuerdo a sus limitaciones de recursos y al mismo tiempo optimizarlo bajo un criterio dado para mejorar el flujo de datos (generalmente hacia la tierra) tanto de instrumentos científicos como de telemetría de plataforma. En efecto, el desarrollo de estas estrategias (inexistentes al comienzo de este doctorado), constituye un avance respecto al estado del arte de la materia.

En particular, la transitoriedad de las comunicaciones entre estos objetos orbitantes suele estar dada por su dinámica orbital o por la necesidad o decisión de reservar energía desactivando temporalmente los equipos de comunicaciones. Como también introduciremos en esta tesis, el desarrollo de arquitecturas satelitales mas sencillas suele implicar

limitaciones de los recursos de comunicaciones (transponders, antenas, etc.) lo que hace menester una elección entre aquellos disponibles. En efecto, el conjunto de contactos resultante de este proceso de diseño se definen como un *plan de contacto*, el que podrá ser administrado a los nodos (satélites) para que lo implementen en su cálculo de rutas de forma tal que el tráfico pueda fluir eficientemente por la constelación satelital.

En general, y como demostraremos mas adelante, realizar un diseño de plan de contacto de forma manual resulta rápidamente prohibitivo por la dificultad involucrada inclusive en los escenarios orbitales mas simples. En el enfoque original que brindamos en esta tesis aprovecharemos la propiedad de predictibilidad de todos estos fenómenos para proveer una toma de decisiones automatizada y optimizada de un problema cuya complejidad incrementa drásticamente con la cantidad de satélites, tiempo de evaluación, y otros componentes que descubriremos a lo largo del trabajo.

1.1.1.1. Relevancia del Problema

Como luego profundizaremos en el marco teórico, resolver este problema de optimización bajo diferentes criterios permitiría implementar de manera eficiente una red tolerante a disrupciones (DTN) de satélites de baja órbita con limitaciones en recursos. En efecto, los beneficios de una arquitectura de este tipo redundarían en constelaciones orbitales de gran cobertura capaces de realizar grandes pasadas de observación terrestre en períodos de tiempo considerablemente menor que sus análogos monolíticos (de un sólo satélite), entre otras ventajas que detallamos en la sección 1.3.1. Específicamente, contar con mecanismos automáticos de diseño verificados y validados constituye un importante paso hacia la seria consideración de las redes DTN para su uso operativo en el espacio.

En el caso particular de la Argentina, estas redes resultan de gran interés para conformación de constelaciones satelitales económicas, autónomas, eficientes, y flexibles como las actualmente ideada por la agencia espacial argentina (CONAE) para la observación terrestre bajo el nombre proyecto SARE. En particular, contar con estrategias de diseño y planificación eficientes como las desarrolladas en esta tesis permitirá a la agencia hacer un mejor uso de los valiosos recursos nacionales puesto en órbita. Sin lugar a dudas, aportar al éxito de esta cometida satelital tan innovadora es contribuir al crecimiento del desarrollo tecnológico nacional a tal punto que Argentina podría convertirse en el primer país en implementar DTN de manera operativa en baja órbita. Sin embargo, vale aclarar que los mecanismos aquí ideados no se limitan a este programa particular, si no que son aplicables a la generalidad de los casos de constelaciones satelitales.

1.1.2. Hipótesis y Objetivos

La hipótesis en la que nos basaremos en este trabajo de investigación es que la efectividad del diseño de planes de contacto para redes disruptivas con recursos limitados será proporcional a la predicción de la información con la que se cuente para el mismo. Por ejemplo, si se conoce información precisa respecto al tráfico que se generará en el sistema se podrá tomar decisiones de planificaciones de recursos de comunicaciones (plasmadas en un plan de contacto) mucho mas eficientes que las que se podrían tomar basados solamente en el conocimiento de la posición relativa de los satélites.

En efecto, el principal objetivo de esta tesis es demostrar la hipótesis planteada, por medio del logro de objetivos derivados tratados en cada uno de los capítulos de esta tesis. Entre estos, vislumbramos el desarrollo de mecanismos eficientes y automáticos de diseño de planes de contacto como resumimos en la lista de objetivos a continuación.

1. Demostrar la hipótesis de que *el rendimiento del sistema mejora con la mayor información incorporada para el diseño del plan de contactos*. Para demostrar esta hipótesis deberemos:
   - *a*) Comprender a la perfección el estado del arte de las redes DTN (capítulo 1).
   - *b*) Generar técnicas de modelado apropiadas para plantear formal y teóricamente el problema de diseño de plan de contacto así como herramientas para evaluar diferentes soluciones (capítulo 2).
   - *c*) Originar mecanismos eficientes de diseño de planes de contacto basados en:
     1) Disponibilidad de información topológica del sistema (capítulo 3).
     2) Disponibilidad de información del algoritmo de enrutamiento implementado en los nodos (capítulo 4).
     3) Disponibilidad de información del tráfico a ser generado en la red (capítulo 5).
   - *d*) Investigar posibles problemas de implementación o aplicación de los planes de contactos diseñados en un sistema DTN funcional (capítulo 6).

1.1.2.1. Metodología de Trabajo

La metodología que utilizaremos para buscar la prueba de esta hipótesis se puede observar claramente en la estructura interna de cada uno de los capítulos del presente trabajo. En particular, para cada mecanismo de diseño aportado, se presenta inicialmente un modelo teórico óptimo (generalmente basado en programación lineal entera o

MILP) con el cual buscaremos entender el problema y explorar espacios de soluciones de escenarios con constelaciones simples. Una vez comprendidos los compromisos del planteo, procederemos a estudiar la factibilidad de la implementación y ejecución de mayores instancias en tiempos y sistemas de cómputos realistas.

Como se observará a lo largo de la tesis, a medida que incrementamos el grado de complejidad de los esquemas, los tiempos de resolución del diseño de plan de contacto hacen que el planteo teórico pierda aplicabilidad en sistemas operativos. En consecuencia, y para los capítulos 3, 4 y 5, formalizamos una propuesta alternativa algorítmica o metaheurística viable para su uso en un centro de control de misión (MOC) de una constelación satelital. En el caso particular del capítulo 6, donde revisaremos detalles de implementación, los algoritmos propuestos combinan su ejecución centralizada en un MOC y distribuida en los nodos de vuelo.

Por último, a lo largo de la tesis presentaremos 3 casos de estudio y referencia de aplicación de constelaciones denominados A: *Topología Escalera* (sección 2.3.1), B: *Topología Lineal Ecuatorial* (sección 3.4.3) y C: *Topología en Tren* (sección 4.4.3). Estos ejemplos ciertamente pueden servir de inspiración para su aplicación real en misiones distribuidas de observación terrestre.

1.1.3. Estructura y Contribuciones de la Tesis

De acuerdo al estado del arte de las redes DTN para uso espacial, revisado y resumido en la sección 1.3 de este capítulo, se ha logrado una serie de avances en el modelado, algorítmica, implementaciones, y hasta experimentaciones en órbita de este esquema de comunicaciones disruptivas. Si bien los mismos resultan de sumo valor en la evolución de las aplicaciones en redes espaciales, los mismos asumen que todos los contactos del plan de contactos se pueden utilizar en el sistema lo cual demostramos no es necesariamente cierto en el capítulo 2, constituyendo la frontera del estado del arte a partir del cual se desarrolla esta tesis.

En consecuencia, en este trabajo descubrimos la problemática y presentamos los primeros mecanismos de diseño para el problema encontrado. Como se ilustra en la Figura 1.1, la estructura de la tesis se basa en una evolución de la complejidad y eficiencia de los esquemas de diseño de plan de contacto correspondiente con el proceso de profundización del conocimiento que se fue llevando a cabo en el trayecto doctoral iniciado a principios del año 2012. En efecto, el contenido de cada capítulo se respalda con una o mas publicaciones en revistas o conferencias internacionales revisada por pares de relevancia para el área de las comunicaciones espaciales.

Tabla 1.1: Estructura y publicaciones de la Tesis

	Título	**Contenido**	**Publicaciones Asociadas**
Capítulo 1	Introducción	Introducción y marco teórico	-
Capítulo 2	Plan de Contactos y Restricciones de Recursos	Definiciones, modelos y herramientas utilizados en la tesis	[1] J. Fraire, et al. Opencl overview, implementation, and performance comparison. Latin America Transactions, IEEE Latin America Transactions, 11(1):1548-0992, Feb. 2013.
Capítulo 3	Diseño de Plan de Contactos basado en Topología	Descripción y análisis del modelo teórico de justicia y algoritmo FCP	[2] J. Fraire et al. On the design and analysis of fair contact plans in predictable delay-tolerant networks. IEEE Sensors Journal, 14(11):3874–3882, Aug 2014. [3] J. Fraire et al. On the design of fair contact plans for delay tolerant networks. In 2013 IEEE International conference on wireless for space and extreme environments (WiSEE), Baltimore, USA, November 2013
Capítulo 4	Diseño de Plan de Contactos basado en Rutas	Descripción y análisis de las rutas y presentación de RACP	[4] J. Fraire et al. Routing-aware fair contact plan design for predictable delay tolerant networks. Elsevier Ad-Hoc Networks Journal, 25:303–313, Feb 2015.
Capítulo 5	Diseño de Plan de Contactos basado en Tráfico	Descripción y análisis del modelo teórico de tráfico y el algoritmo genético derivado	[5] J. Fraire et al. Design challenges in contact plans for disruption tolerant satellite networks. IEEE Communications Magazine, 53(5):163–169, May 2015. ISSN 0163-6804. [6] J. Fraire et al. Contact plan design for predictable disruption tolerant space sensor networks. Chapter 15 of Wireless sensors systems for extreme environments: space, underwater, underground, and industrial. ISBN 978-1-119-12646-1. [7] J. Fraire. Traffic aware contact plan design for scheduled disruption tolerant networks. Technical report LCD-1504-01. April 2015. [8] J. Fraire et al. Preliminary results of an evolutionary approach towards contact plan design for satellite DTNs. In 2015 IEEE International Conference on wireless for space and extreme environments (WiSEE). Orlando, USA, December 2015.
Capítulo 6	Implementación de Planes de Contacto	Problemas generales y particulares de TACP en la implementación de planes de contacto en la red satelital final	[9] J. Fraire et al. Leveraging routing performance and congestion avoidance in predictable delay tolerant networks. In 2014 IEEE International conference on wireless for space and extreme environments (WiSEE), Noordwick, Netherlands, October 2014. [10] J. Fraire et al. Congestion modeling and management techniques for predictable disruption tolerant networks. In proceedings of IEEE Conference on Local Computer Networks (LCN), October, 2015.
Capítulo 7	Conclusiones	Corolario y trabajo futuro	-

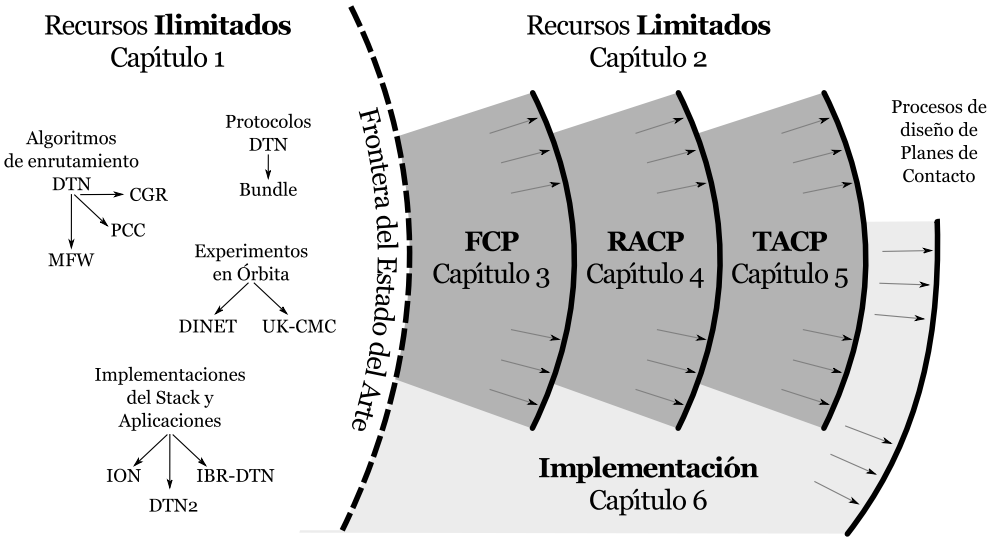


FIGURA 1.1: Contribuciones de la tesis al estado del arte

En particular, en el capítulo 1 presentamos esta introducción y brindaremos el marco conceptual necesario para entender el estado del arte de la tecnología de comunicaciones en redes tolerante a disrupciones. Luego, en el capítulo 2 introducimos una serie de definiciones, técnicas de modelado, y conocimientos de base necesarios para poder enfrentar el problema de diseño de planes de contactos. En efecto, el primer mecanismo (FCP) se describe en el capítulo 3 donde aportamos un algoritmo de diseño de plan de contacto que solamente asume un conocimiento de la topología de la red satelital. Al considerar una mayor cantidad de información como por ejemplo las rutas de comunicación entre los nodos se pudo contribuir con un segundo mecanismo de relevancia (RACP) en capítulo 4 de la presente tesis. Sin embargo, el estado del arte en la problemática de diseño se detalla en el capítulo 5 donde describimos el mecanismo mas complejo y eficiente (TACP) obtenido recientemente hacia la finalización del doctorado. A continuación, en el capítulo 6 revisamos otros aportes significativos realizados en un área directamente relacionada con el diseño de plan de contacto que es la implementabilidad de los mismos en la red real. Finalmente, resumimos las conclusiones en el capítulo 7.

En la Tabla 1.1 resumimos los capítulos, contenidos, y publicaciones asociadas a los mismos. Otras publicaciones derivadas como [11] y [12] se nombran a lo largo de la tesis pero no se listan en la tabla al ser considerados aportes secundarios del presente trabajo.

1.2. Marco Ideológico

Como disparador de la presente tesis doctoral, quisiera retomar un interesante y atinado texto del pedagogo brasilero Paulo Freire en el que discute el compromiso del profesional en la sociedad en la que se encuentra inmerso [13]. Para esto entendimos al investigador

(doctorando) como profesional para buscar dirigir el esfuerzo del mismo en el trabajo doctoral cuyo aporte se describe esta tesis final.

En particular, la primera condición que Freire establece para que un ser pueda ejercer un acto de compromiso, es que éste esté en condiciones de actuar y reflexionar, pues estas dos acciones son las que nos permiten modificar la realidad y conocer el cambio respectivamente. Un ser sin capacidad de reflexión sobre sí no puede trascender los límites que le son impuestos por el mismo mundo, quedando inmerso y neutro en él, siendo incapaz de emerger o alejarse para admirarlo. En efecto, solamente puede comprometerse un ser que es capaz de emerger de su contexto y admirarlo para, objetivándolo, transformarlo, y transformándolo, saberse transformado por su propia creación. De esta manera el compromiso implica una responsabilidad histórica, en contacto con la realidad concreta, donde se encuentran los hombres concretos. Al existenciarlo, los hombres ya no se dicen neutros, dado que la neutralidad sólo refleja un el miedo que se tiene de revelar el compromiso.

En cuanto al profesional o investigador, dice Freire, además del compromiso genérico ya descrito, le es propio el compromiso de profesional, producto de una deuda que este asume ante la sociedad al hacerse profesional. Esto se debe a que mientras más se capacita como tal, mayor es la capacidad de sistematizar sus experiencias; y cuanto más se sirve del patrimonio cultural (que es patrimonio de todos y al que todos deben servir), más aumenta su responsabilidad en los hombres. Es clave entonces que el investigador no se deje seducir por tentaciones míticas como la esclavitud a las técnicas, que siendo elaboradas por los hombres son sus siervas y no sus señoras; que no se admita como habitante de un extraño mundo de especialistas dueños de la verdad, propietarios del saber, que debe ser donado a los ignorantes o incapaces. Un investigador que proceda de esta manera se aleja del compromiso, se aliena como profesional y como ser humano.

El desafío del profesional es aún mayor en sociedades alienadas como las latinoamericanas, de economías periféricas, exportadoras de materias primas e importadoras no sólo de productos manufacturados si no que de ideas, de técnicas, de modelos. En gran medida se importan técnicas y tecnologías sin la debida *reducción sociológica* necesaria dado que las sociedades metropolitanas en las que se desarrollaron no son necesariamente idénticas. Considerar que la técnicas pueden ser trasplantadas de un contexto a otro, es considerarlas neutras, es des-humanizarlas. De aquí que el hombre latinoamericano quede alienado, inseguro y frustrado, incapaz de asumir un auténtico compromiso con su propia sociedad dice Freire. En este contexto, el compromiso del investigador encuentra un desafío y responsabilidad aún mayor debido a la falta de autenticidad en la sociedad en la que este se enmarca.

En particular, el autor de esta tesis sostiene que la soberanía de un pueblo se vincula con su capacidad de poder conocer y ser consciente del lugar que habita, actividad que en la modernidad se asocia de cerca a la observación terrestre llevada a cabo con varios hitos de relevancia internacional desde hace mas de 20 años por la agencia espacial Argentina (Comisión Nacional de Actividades Espaciales o CONAE). Sin embargo, y en general, el acceso al espacio de países con recursos limitados ha requerido de estrategias audaces que permitan adaptar sus técnicas a los estándares internacionales de excelencia, en su mayoría derivados en contexto de la guerra fría. Ninguna realidad latinoamericana contemporánea puede estar mas alejada de esta trama, por lo que se requiere de ideas adecuadas, apropiadas, y originales que permitan un acceso a la observación espacial acondicionada a las necesidades locales.

En este contexto pedagógico e ideológico se embarca el presente proyecto doctoral, en búsqueda de alternativas viables y adecuadas al contexto Argentino y latinoamericano en general, para acceder a realizar experimentaciones u observaciones en el espacio exterior. Actualmente el acceso a colocar un objeto en órbita es limitado para países en vías de desarrollo debido a los altos costos involucrados y desarrollos requeridos. Sin embargo, tras reconocer esta realidad de la sociedad, en este trabajo consideramos un conjunto de técnicas que con cierta adaptación pueden resultar revolucionarias en la tan conservadora industria espacial. En este sentido, en la próxima sección 1.3.1 retomaremos una visión del acceso al espacio originalmente ideada con fines militares por los Estados Unidos, pero hoy repensada con fines civiles para la Argentina y otros países latinoamericanos o en vías de desarrollo.

1.3. Marco Teórico

En esta sección describiremos la Arquitectura Segmentada como fuente de inspiración para la aplicación de redes de conexión esporádica en la cual se inspira el problema de diseño de plan de contacto tratado en esta tesis. Sin embargo, vale aclarar que la aplicabilidad de los aportes de esta tesis se extiende a cualquier constelación satelital.

1.3.1. La Arquitectura Segmentada para la Observación Terrestre

1.3.1.1. Sobre la Observación Terrestre

Actualmente se toman continuamente imágenes ópticas y de radas desde el espacio dado que estas se han convertido en herramientas científicas de suma utilidad para el mejor entendimiento y gestión de la Tierra y su entorno. Tradicionalmente, un satélite con un

sensor, radar, o carga útil fotográfica es capaz de adquirir datos de diferentes partes del mundo incluyendo sitios remotos probablemente inaccesibles por otros medios terrestres. Esto permite hacer de la observación terrestre desde el espacio un medio eficiente de proveer cobertura tanto a lo largo del tiempo como del espacio. Estas ventajas han sido entendidas por la Comisión Nacional de Actividades Espaciales (CONAE) y aplicadas en numerosas misiones exitosas de observación terrestre, colocando a la Argentina dentro de las principales agencias espaciales del mundo con importante misiones exitosas en su haber como las basadas en los satélites SAC-A, SAC-B, SAC-C y recientemente el SAC-D desarrollado en cooperación con la National Aeronautic and Space Administration (NASA) de los Estados Unidos.

1.3.1.2. Entorno Espacial

En un contexto mas general, el despliegue de aeronaves espaciales como satélites o misiones tripuladas en el espacio ha implicado sobreponerse a numerosos desafíos debido a los efectos provocados por la agresividad del ambiente espacial en el que numerosas causas potenciales de fallas transitorias o permanentes deben ser consideradas. Entre ellas se destacan colisiones con objetos orbitantes (*debris*), rotura de componentes debido a las altas vibraciones provocadas por el lanzador, altos rangos de variabilidad de temperatura, descomposición de materiales en vacío y efectos de radiación, entre otros. Este último es de particular interés científico dado que las aeronaves espaciales están fuertemente basadas en circuitos electrónicos, causa por la cual los efectos de radiación deben ser especialmente considerados para garantizar los grados de confiabilidad y seguridad requeridos para la misión [14].

En consecuencia, la probabilidad de falla de componentes relevantes a la capacidad de operación del sistema, es reducida con técnicas de confiabilidad como por ejemplo la inclusión de elementos redundantes, pero a costa de un incremento de la complejidad [15]. Este es el enfoque actualmente adoptado por la industria espacial en general, para entregar cierta funcionalidad con cierta confiabilidad: agregar márgenes y redundancia a nivel componentes, especialmente, en áreas críticas para la misión. Sin embargo, la incorporación de un mayor número de componentes en una misma plataforma física deriva en incrementos de la complejidad de la misma (dado que se requieren más partes con redes de interconexiones más intrincadas). Esto a su vez deriva en nuevos modos de fallas a ser considerados y mitigados, incrementando la fragilidad del sistema (probabilidad de que existan modos de fallos no modelados [16]). Además, a mayor complejidad, mayor tamaño físico de la aeronave, mayor costo, y mayores tiempos de ejecución de proyecto. En la mayoría de los casos, esto también dispara la compra de costosos seguros contra fallas del lanzador, incrementando aún más el presupuesto global para la misión. Como

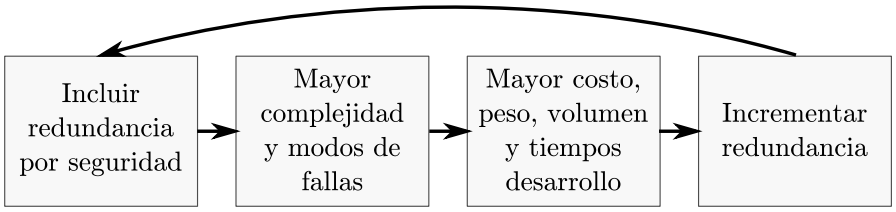


FIGURA 1.2: El efecto espiral de la redundancia

consecuencia del incremento de costos y tiempos descriptos, un mayor grado de confiabilidad tiende a ser impuesto para minimizar la posibilidad de pérdidas catastróficas. Esto a su vez resulta en mayores márgenes y redundancia, la misma razón por la cual las mismas fueron consideradas en primer lugar. Este fenómeno fue estudiado bajo el nombre de "espiral de la muerte"de la industria espacial en recientes investigaciones [16] y se ilustra en la Figura 1.2.

La complejidad asociada a aeronaves tradicionales tiende a concentrarse en una misma plataforma monolítica, la que en consecuencia se caracterizan por una alta densidad de la complejidad. Por otro lado, y para ilustrar este concepto, la literatura ejemplifica este fenómeno por medio de los dos productos de la ingeniería más complejos desarrollados por el hombre: *Internet* y el *transbordador espacial*. A pesar de que el primero es más complejo en cuanto al número de componentes, el segundo cuenta con un mayor registro de fallas catastróficas. La diferencia entre ambos sistemas es sencilla: la distribución espacial de la complejidad. En general, más allá del valor absoluto de la complejidad, la probabilidad de fallas de un sistema es principalmente proporcional a la densidad de la misma.

1.3.1.3. Surgimiento de la Arquitectura Segmentada

La combinación del "espiral de la muerte"de Brown [16], y los beneficios de la distribución espacial, llevó a la Agencia Espacial Argentina (CONAE), Servicios Tecnológicos Integrados (STI), y la Universidad Nacional de Córdoba (UNC) a desarrollar alternativas y soluciones originales bajo el nombre de Arquitectura Segmentada (AS). La AS se enmarca en el Plan Espacial Nacional (PEN) y se inspira en el proyecto F6 [17] del departamento de defensa de Estados Unidos (DARPA) para ser considerada y adaptada para brindar confiabilidad a una aeronave con fines científicos al introducir un enfoque innovador para diseñar, desarrollar, y operar misiones espaciales. En particular, la AS logra incrementar la confiabilidad al distribuir espacialmente la complejidad del sistema en varios módulos (denominados *segmentos*) interconectados de forma inalámbrica. Como explicaremos más en detalle, esto último permite actualizar, reemplazar, y redundar el sistema no sólo en la etapa de diseño y fabricación, sino que también durante la etapa de

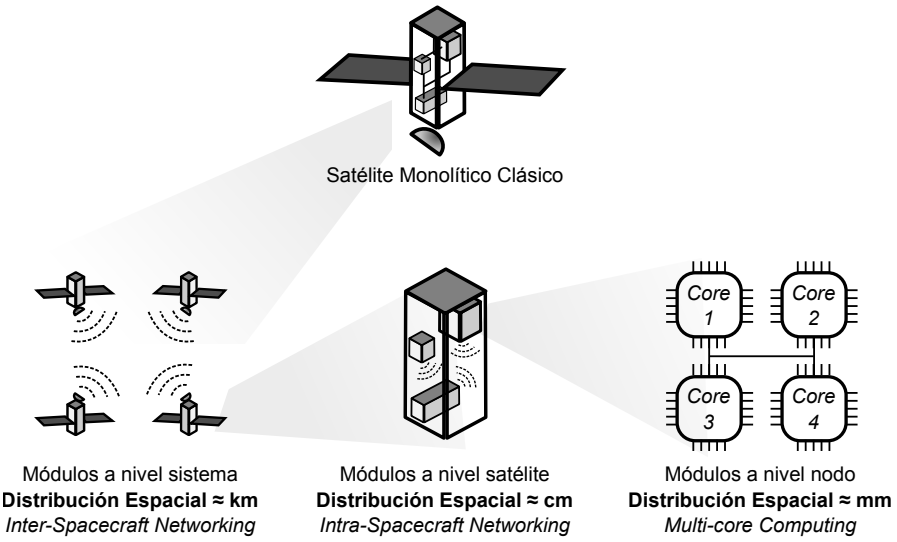


FIGURA 1.3: Distribución Espacial en la Arquitectura Segmentada

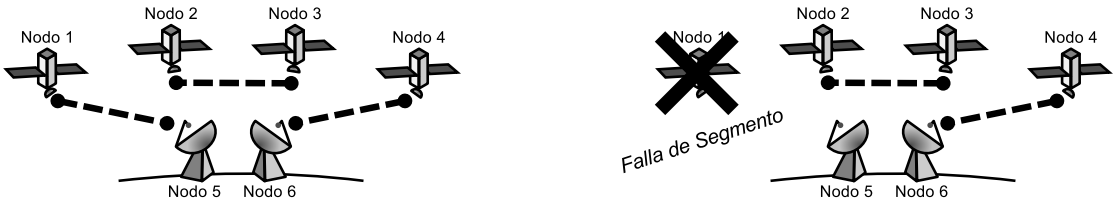
operación en órbita. Por último, la AS está pensada para nutrirse de otras tecnologías de distribución espacial a otras escalas como lo son las comunicaciones inalámbricas internas y los procesadores de múltiples núcleos para aplicaciones espaciales. Esta integración se ilustra en la Figura 1.3 y supone una serie de beneficios de importancia para la industria espacial local e internacional detallados a continuación.

1.3.1.4. Beneficios Operativos

Dentro de los beneficios que otorga la AS para la generalidad de las misiones espaciales se destaca la degradación paulatina de las prestaciones en caso de fallas. En particular, dada las características de severidad del entorno espacial y la imposibilidad de acceder a hacer reparaciones, la tolerancia a fallas resulta un tema crítico. En este contexto, la AS permite tolerar fallas sin necesidad de comprometer la totalidad del sistema como ocurriría en un sistema monolítico clásico. Este fenómeno deriva de la capacidad de explotar la diversidad espacial para aislar las fallas que de otra manera tendrían impacto global. La Figura 1.4 ilustra este efecto para el caso de las comunicaciones de bajada de datos donde la falla de un segmento no implica la pérdida de conectividad con el sistema como lo haría en un sistema monolítico tradicional. En consecuencia, la confiabilidad del sistema segmentado ya no es solo producto de la calificación de los componentes si no que de la cantidad de los mismos [11]. Esto último no resulta irrelevante ya que permite considerar componentes comerciales (*commercial of the shelf* en Inglés o COTS) de mayor prestaciones y disponibilidad en el mercado que aquellos calificados para el espacio.

Por otro lado, la Arquitectura Segmentada otorga otra serie de beneficios ilustrados en la Figura 1.5. Por un lado, al dividir la funcionalidad en segmentos de menor envergadura, permite diversificar los lanzamientos necesarios para colocar la misión en órbita. En consecuencia, la pérdida de un vector con la carga útil no implica la pérdida total de la misión como suele suceder en misiones monolíticas. Resulta mucho mas sencillo reparar el daño al necesitar reemplazar un sólo módulo con un nuevo lanzamiento que fabricar nuevamente la plataforma completa. Esta ventaja por otro lado es de particular interés para la industria espacial local o de países en vías de desarrollos los que deben acceder al espacio por medio de lanzadores extranjeros al no poseer el dominio de la tecnología de balística para colocar grandes volúmenes y masas en órbita. Sin embargo, y gracias al reciente desarrollo de un vector local (Tronador) de cargas menores ($500Kg$), la AS se convierte en un medio clave para que Argentina pueda diseñar, lanzar, y operar de manera totalmente autónoma e independiente misiones espaciales de observación terrestre. En otras palabras, el conjunto Tronador y Arquitectura Segmentada emergen como los dos pilares del PEN que buscan a largo plazo una verdadera soberanía espacial.

Además de las dos principales ventajas de degradación paulatina y diversidad de lanzadores, la AS ofrece otras novedades de interés general para la industria espacial. Como se muestra en la Figura 1.5, la AS permite efectuar actualizaciones y reparaciones en órbita. Ya sea para incorporar nuevas funcionalidad o reparar prestaciones perdidas por el cumplimiento de la vida útil de un segmento, el sistema puede ser modificado en órbita durante su operación nominal. Tradicionalmente, esta característica de misión reparable, era reservada para grandes agencias capaces de realizar caminatas espaciales tripuladas como lo supo hacer la NASA para reparar el telescopio orbital *Hubble* [18]. Además, los segmentos físicamente distribuidos permiten una reconfiguración en órbita para adaptarse a diferentes necesidades de misión. Finalmente, pero no menos importante, la distribución espacial de los segmentos permite aumentar la tasa de revisita (comunicaciones con tierra) y considerar estrategias de fabricación seriales y modulares que mejoren los tiempos y costos de manufactura.



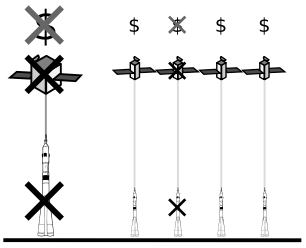
Taza de bajada de datos: ***100%***

Dos estaciones terrenas permiten obtener la máxima performance del sistema segmentado.

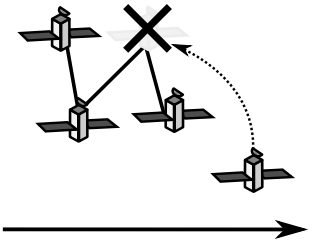
Taza de bajada de datos: ***50%***

En caso de falla, la capacidad de bajada se degrada pero no se anula por completo.

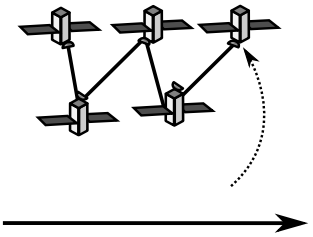
FIGURA 1.4: Degradación paulatina en caso de fallas



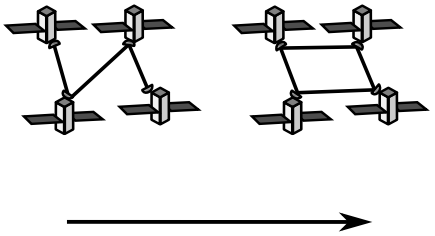
Diversidad de Lanzamientos
La varianza del riesgo se reduce dado que la falla de un lanzamiento no implica la perdida de la misión completa



Reparaciones en Órbita
La varianza del riesgo se reduce dado que la falla de un componente puede ser reparada con la incorporación de un nuevo módulo



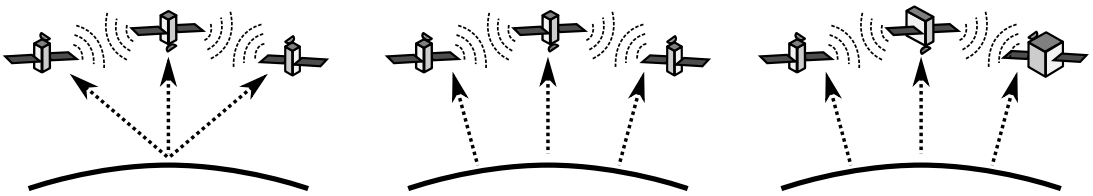
Mejoras en orbita
La incorporación de un nuevo módulo al sistema permite mejorar su capacidad y funcionalidad durante la etapa de operación



Adaptación
El sistema segmentado puede reconfigurarse en órbita para satisfacer diferentes requerimientos de misión.

FIGURA 1.5: Otros beneficios de la Arquitectura Segmentada

Sin embargo, además de los beneficios brindados por la arquitectura a la operación de la misión segmentada, existen una serie de posibilidades sin precedentes respecto a la elección de posibles cargas útiles de los satélites que impulsó a CONAE a desarrollar este concepto. Dado que un satélite permite obtener observaciones en lugares remotos con una periodicidad atractiva para la financiación de actividades espaciales, la posibilidad de integrar un grupo de estos nodos autónomos y espacialmente distribuidos (concepto conocido como Red de Sensores Espaciales o *Space Sensor Network* (SSN) en Inglés [19]) abre las puertas a considerar aplicaciones sin precedentes al extender significativamente la cobertura tanto en el espacio como en el tiempo. La Figura 1.6 lista estas posibilidades entre las que se encuentran la capacidad de incrementar la apertura del instrumento al poder contar con varias instancias del mismo en diferentes plataformas apuntado a un mismo objetivo. En efecto, esto da lugar a obtener diferentes adquisiciones y luego combinarlas en tierra ya sean del tipo ópticas (imágenes de super-resolución) o de radar (tecnología de antena partida [20]). Por otro lado, en caso de que los sensores apunten a zonas geográficas adyacentes, se puede explotar la ventaja de mayor cobertura al extender la pisada de la adquisición. Por último, también se puede combinar sensores de diferentes naturaleza para tomar diferentes mediciones de un mismo fenómeno en la



Mayor Apertura
La distribución de una serie de un mismo sensor permite incrementar la apertura del mismo. (super-resolución o antena partida)

Mayor Pisada
La distribución de una serie de un mismo sensor permite incrementar la pisada o cobertura sobre la tierra.

Diversidad de Sensores
La distribución de una serie de diferentes sensores permite obtener diferentes mediciones de un mismo fenómeno.

FIGURA 1.6: Posibilidades de cargas útiles segmentadas

tierra de manera análoga a la misión A-Train de NASA [21] (aunque la misma no cuenta con sistema de comunicación entre sus segmentos).

En cuanto a las cargas útiles, cabe destacar que no todos los instrumentos de observación terrestre pueden ser segmentados. Inclusive, tecnologías como la de antena partida sencillamente aún no están en el estado de arte necesario para ser implementados de manera operativa en órbita [20]. En consecuencia, hay satélites que no podrán segmentarse y por ende tampoco podrán beneficiarse de pequeños lanzamientos, pero si podrán integrarse a otros sistemas segmentados para hacer usos de sus servicios e interactuar con otros sensores o instrumentos como se ilustra en la Figura 1.7. En otras palabras, la AS incluye aquellos instrumentos monolíticos lo que, por definición, extiende sus capacidades al incorporarlos al sistema existente.

1.3.1.5. Desafíos Tecnológicos

De esta manera, como se introdujo en la sección 1.2, la AS se adapta a las necesidades de la industria espacial nacional con una serie de importantes beneficios que podrían

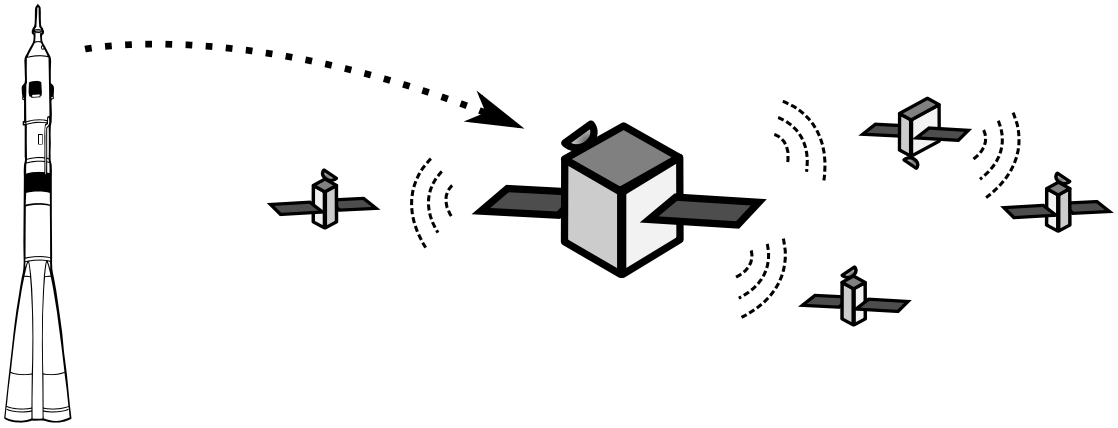


FIGURA 1.7: Instrumentos sin segmentar

dar lugar a un avance en la forma en que se accede al espacio a nivel mundial. Sin embargo, las consecuencias de dividir o segmentar aeronaves en un sistema distribuido que cumple con la misma funcionalidad que su análogo monolítico, no solo supone ventajas si no que importantes desafíos tecnológicos a resolver antes de que cualquier misión de AS pueda ser implementada. Entre estos se destacan la capacidad de comunicación inter-satelital (*Inter-Satellite Link* en Inglés o ISL) [22], la distribución de funciones entre segmentos (memoria, bajada de datos, procesamiento, etc.), la distribución de instrumentos (sistema óptico o de antena partida), y las comunicaciones en red entre segmentos orbitantes [23]. Dada la necesidad de la agencia espacial argentina y el grado de innovación del proyecto de AS, en la presente investigación doctoral se toma el tema de comunicación en red como eje, contexto, y posible beneficiario de la investigación.

A pesar de que numerosas tecnologías y protocolos de comunicaciones en red existen para aplicaciones terrestres como WiFi, o TCP/IP [24], la aplicación de las mismas en un entorno espacial resulta muchas veces imposible o poco eficiente [25] debido a que la naturaleza orbital de los satélites impide mantener conexiones permanentes. En paralelo, durante los últimos 20 años, las comunicaciones espaciales han mostrado un avance limitado en comparación con las redes basadas en Internet en la Tierra. Es por esto que recientemente la NASA, entre otras agencias espaciales, han decido moverse hacia redes basadas en paquetes mediante el uso de protocolos apropiados y específicos [26].

Además, a diferencia de Internet, las condiciones extremas del entorno espacial en el que los satélites deben operar obligan a enfrentar numerosas situaciones inexistentes en la Tierra como inaccesibilidad física, dinámica orbital cambiante, inestabilidad de hardware debido a temperaturas extremas, radiación, basura espacial, o restricciones energéticas [19, 27]. Dado que los protocolos de Internet existentes no fueron pensado para operar en estas condiciones, resulta menester replantear le filosofía basal de las comunicaciones en red para así poder adoptar un camino de desarrollo realista y apropiado que permita soportar el futuro desarrollo de la Arquitectura Segmentada para su uso a nivel nacional e internacional. En efecto, en la siguiente sección 1.3.2 repasamos el estado del arte de las comunicaciones móviles en tierra y su intento de aplicación en el entorno espacial.

1.3.2. El Problema de la Conectividad Permanente

1.3.2.1. Orígenes de Internet

A principios de los 60s, el departamento de defensa de los Estados Unidos decidió conectar sus redes de datos locales de universidades y centros militares con el fin de compartir

recursos y potenciar la comunicación nacional. En ese contexto, era clave la necesidad de mantenerse comunicados en caso de posibles ataques nucleares soviéticos, escenario en el cual, la conexión entre los diferentes componentes del sistema debía mantenerse. Este supuesto derivó en los primeros avances en flexibilidad y adaptabilidad para redes de datos, sentando las primeras piedras del paradigma comunicacional de Internet tal como hoy lo conocemos. Entre los desarrollos más destacados se destaca el set de protocolos TCP/IP [24], que al día de la fecha cuenta con más de 2,4 mil millones de usuarios (34 % de la población mundial [28]), demostrando una capacidad de escalar a dimensiones nunca pensadas.

1.3.2.2. Telefonía Móvil Celular

Luego, con el fin de traficar voz, surgieron las comunicaciones móviles cuyo principal desafío era la movilidad del terminal de usuario. El esquema ingenioso fue armar una malla de antenas fijas (de aspecto celular, de ahí su nombre) a las cuales el móvil se iría conectado en el transcurso de su camino, por medio de saltos (*handovers*) de tiempo tan ínfimo que el usuario no fuese capaz de notarlo. Actualmente, se estima que los usuarios de comunicaciones móviles supera los 6,8 mil millones (96 % de la población mundial). No fue hasta 1997 que se intentó acoplar este sistema con la red de datos aunque con la dificultad de tener que adaptar esta última a tolerar los cambios de radio-bases. En efecto, el desafío radicaba en que Internet estaba pensada para estar "siempre conectada" desde su origen. Dicha motivación derivó en la implementación de los actuales protocolos de movilidad, que pusieron al límite el concepto de redes, y que hoy nos permiten transmitir y recibir datos digitales desde nuestros celulares con tecnologías como 3G, 4G, entre otras [29]. El ingenio aquí pasa por hacer creer, al usuario, y al resto de Internet, de que están en permanente conexión por un camino fijo (ruta), cuando en realidad la comunicación es disruptiva por naturaleza. Este proceso se ilustra en la Figura 1.8 donde un terminal móvil tolera una desconexión de durante una pequeña fracción de tiempo (del orden de los mili-segundos) con el soporte de la red central quien almacena los datos temporalmente hasta que se resume la comunicación.

1.3.2.3. Países en Vías de Desarrollo

Sin embargo, dado que no resultan económicamente atractivos, algunos países del mundo en vías de desarrollo con escasa o nula infraestructura de comunicaciones no pudieron ser beneficiarios directo de un ingenio como el utilizado en las redes móviles. No son pocos los pueblos o comunidades en África que hoy no pueden participar directamente de una

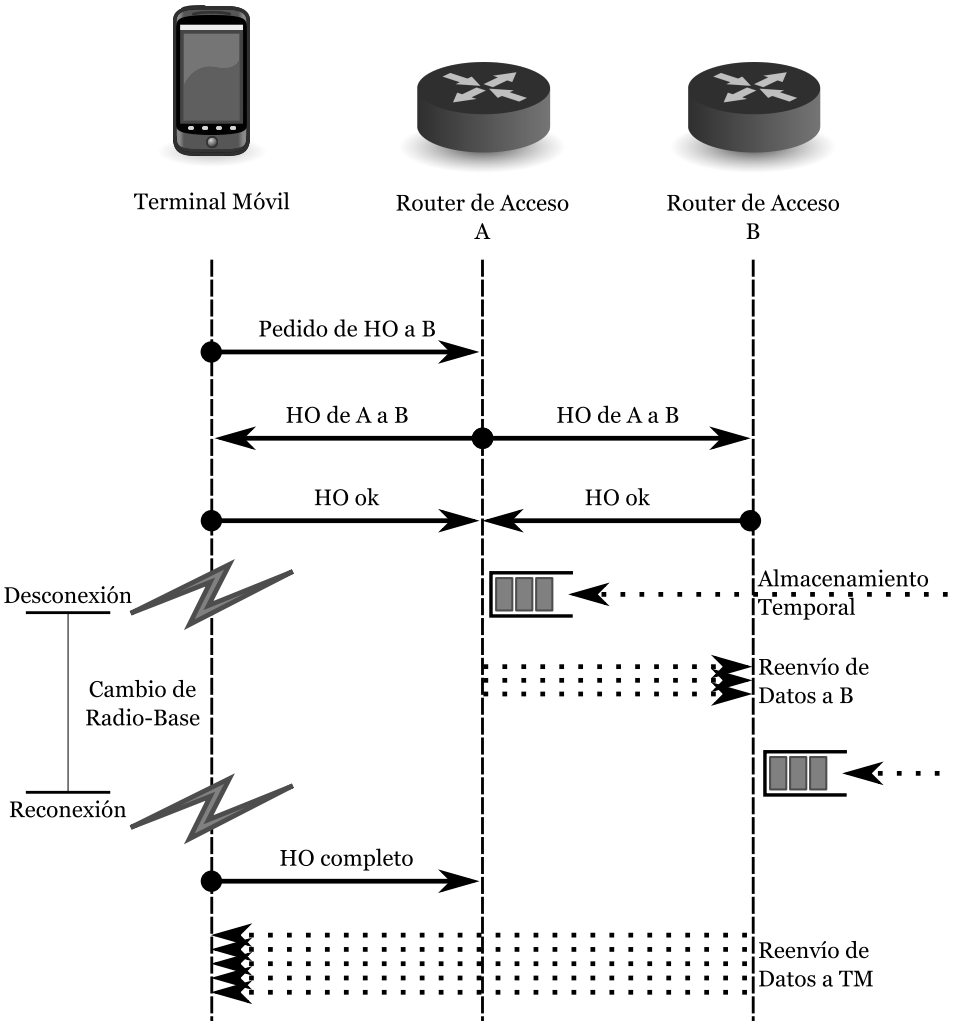


FIGURA 1.8: Proceso de Intercambio de Radio-Base (*Handover*)

experiencia de conexión a Internet; sencillamente, porque no pueden contar con un enlace físico telefónico o inalámbrico que los incluya en el sistema. Ingeniosas alternativas se propusieron como por ejemplo la de proveer recorridos periódicos de motocicletas [30] o inclusive pájaros [31] con enrutadores inalámbricos que, recorriendo los pueblos y tribus, recolectan vía tecnologías símil *WiFi* peticiones de *Wikipedia*, búsquedas en *Google*, e-mails, entre otros para luego enviarlas a Internet al arribar a una ciudad con infraestructura. Resulta sorprendente la similitud de este esquema con el paradigma postal. Sin embargo, dado que ni *Wikipedia*, *Google*, o ningún sistema de Internet soporta la espera del tiempo que toma el vehículo en hacer su recorrido, se deben implementar tecnologías de adaptación, nuevamente, para "engañar" a Internet sobre la existencia de las demoras descritas. Recientemente el *proyecto loon* de *Google* busca por medio de globos busca mantener el paradigma de conexión permanente por medio de un sistema masivo de globos aerostáticos [32]. Casos similares de necesidades de adaptación son las cada vez mas populares *redes de sensores*, mecanismos de identificación por radiofrecuencia (RFID), entre otros.

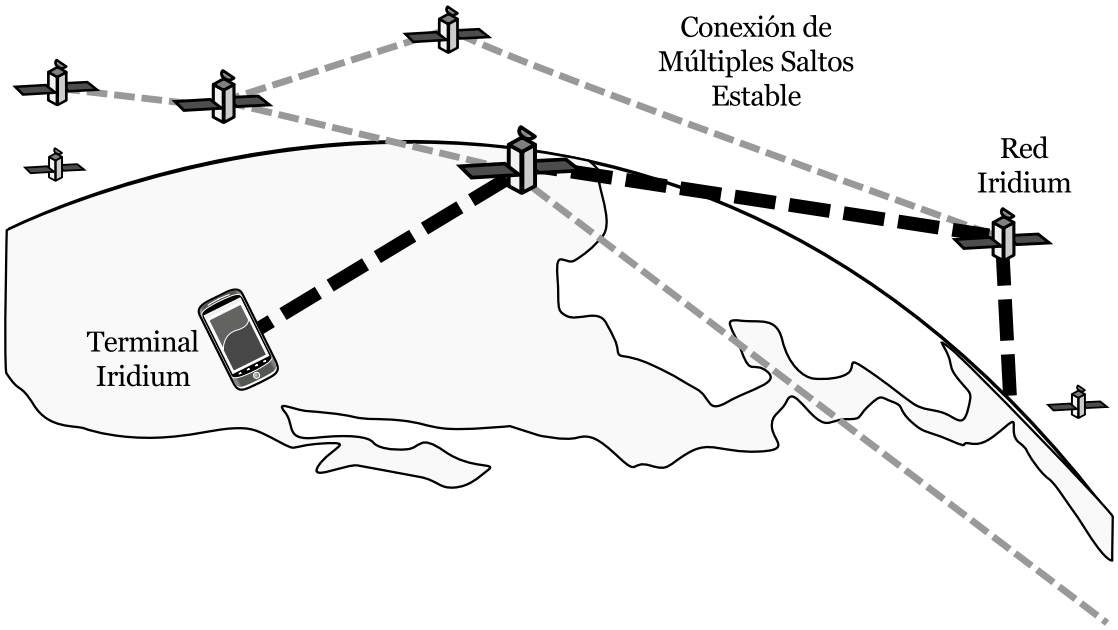


FIGURA 1.9: Operación de la red Iridium

1.3.2.4. Comunicaciones Espaciales

En el caso de las tecnologías espaciales la historia no difiere demasiado del caso de la telefonía móvil o las comunicaciones en África. En sistemas de satélites de baja órbita (LEO), las disrupciones suelen ser frecuentes debido a la mecánica orbital. Por ejemplo, con el fin de proveer servicios de tráfico de voz, la constelación de satélites Iridium [33] tubo que ser diseñada con mas de 66 satélites con suficiente margen de enlace para evitar disrupciones y soportar caminos de múltiples saltos de extremo a extremo en un contexto altamente cambiante a nivel global. Esta operación se ilustra en la Figura 1.9. De esta manera, la conectividad permanente fue alcanzada a costa de un sistema altamente complejo, caro, y controversial basado en *handoff* en órbita similares a los discutidos para redes de celulares móviles. Esta situación llevó a Iridium a caer en bancarrota antes de ser rescatado con fondos públicos del gobierno de los Estados Unidos.

Mas recientemente, un ambicioso proyecto del Defense Advanced Research Projects Agency (DARPA) del mismo país denominado F6 [17], propuso implementar un sistema distribuido de aeronaves interconectadas por redes de conexión permanente tipo malla (*mesh* en Inglés). Si bien la Arquitectura Segmentada hereda algunas de las ideas basales del F6, se necesita de una revisión y readaptación de las tecnologías a un contexto local dado que este último tuvo que ser cancelado luego de un drástico incremento de presupuesto debido a la complejidad que había tomado el proyecto.

Por otro lado, los satélites de repetición geoestacionarios (GEO) han sido tradicionalmente atractivos para la amplia cobertura de señal a lo largo de un territorio dada su

posición fija en relación a la rotación de la Tierra y a la distancia a la que se ubican (30000Km aprox.). De esta manera, los mismos utilizan repetidores del tipo *bent-pipe* que esencialmente reciben una transmisión de una estación terrestre para luego reflejarla de nuevo a una determinada superficie objetivo. Sin embargo, al ser considerados para comunicaciones interactivas de datos como la requerida por Internet, se deben resolver desafíos particulares como altos tiempos de respuesta (round trip times o RTT en Inglés) y frecuentes disrupciones [34]. En consecuencia, se suelen emplear *proxies* específicamente denominados *performance enhancing proxies* en Inglés (PEPs) [35] quienes permiten mitigar estos efectos. No obstante, los PEPs muestran impactos negativos en rendimiento al no poder contar con ciertas funcionalidades de seguridad e inter-operabilidad [34].

El caso de sistemas espaciales de espacio profundo es aún mas problemático dado que las distancias son aún mayores provocando mayores demoras y disrupciones mas severas debido a la rotación de los planetas [36]. Tradicionalmente las comunicaciones con el espacio exterior eran esencialmente punto a punto y esporádicas por naturaleza, ya que "Los planetas rotan y eso es algo que aún no hemos podido cambiar" dice Vincent Cerf en [37]. En consecuencia, los *rovers* (vehículos terrestres) en marte evolucionaron de mandar la información directamente a tierra a una baja tasa de datos (del orden de los $Kbps$), a utilizar satélites *relay* intermedios fuera de la atmósfera marciana con paneles solares más grandes, permitiendo mayores capacidades y velocidades ($Mbps$). Estos almacenes intermedios de información luego la envían hacia la tierra al generarse una linea de visibilidad. En este escenario, el famoso set de protocolos de Internet TCP/IP [24] fallan ante la esporadicidad y las demoras, sobre todo las del orden de 20 minutos de ida y vuelta (RTT) que puede haber hasta el planeta rojo. En efecto, al día de la fecha se analizan otros mecanismos alternativos que sostienen y forman parte de la idea de *red interplanetaria* (IPN) [38]. En general, IPN es un proyecto que aparenta ser de ciencia ficción, y que propone diseñar una red entre planetas en el sistema solar. Dentro de sus potenciales utilidades se destacan la de servir de plataforma de comunicación de futuras colonias en otros planetas y la de generar una *antena de apertura sintética* de inmensas dimensiones capaz de captar señales provenientes de nuestras propias sondas a enviarse más allá del sistema solar.

Ya sea una red interplanetaria, telefonía celular, o comunicaciones de comunidades africanas, hay una realidad: Internet como hoy la entendemos no fue pensada para ninguna de ellas; se necesitan adaptaciones para participar de un sistema que se jacta de ser universal. El nombrado supuesto inicial de conectividad permanente resulta entonces restrictivo a nivel tecnológico e inclusive a nivel cultural. Hoy nos cuesta concebir un esquema de Internet tolerante a demoras como el que existe hace años en el sistema postal. No es trivial que en caso de no tener una conexión a Internet estable, luego de introducir la dirección de una página, los exploradores nos devuelvan la frase de *Error*.

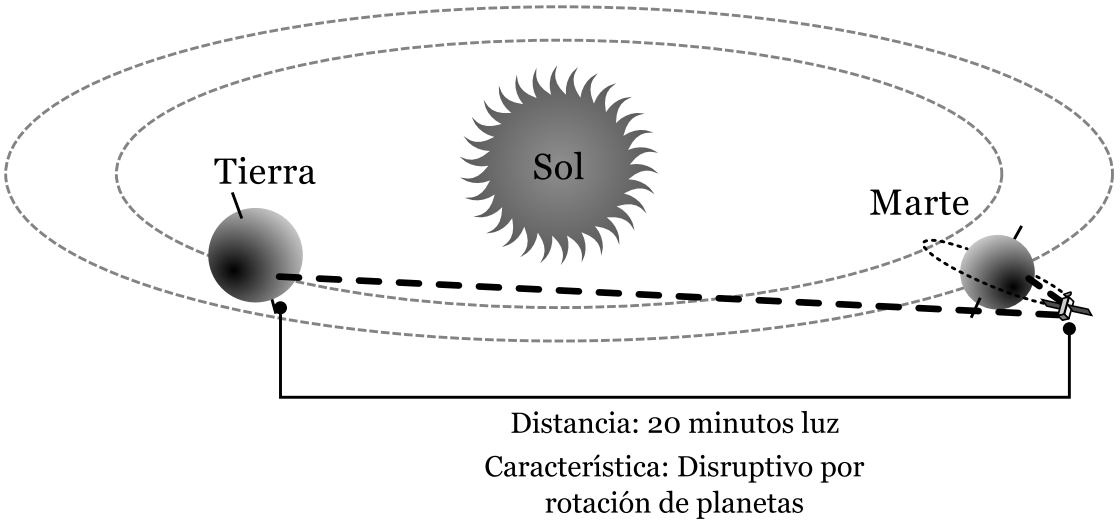


FIGURA 1.10: Esquema del proyecto de Red Inter-planetaria

Esto no es si no evidencia de que el sistema no está preparado para tal cosa; por lo que hoy es válido preguntarse si el mismo es realmente un error o una limitación. En efecto, a lo largo de esta sección hemos intentado demostrar que desde hace tiempo se viene "parchando" métodos de comunicaciones para incluirse en un sistema por el mero hecho de tomarlo como cierto, válido y absoluto.

Sin embargo y evidentemente, sería poco razonable pensar en la sociedad tal como hoy la conocemos sin redes sociales, correo electrónico, sin Internet. Pero lo que sabemos es que si queremos que sea el sistema de comunicaciones del mañana, debemos revisar las bases filosóficas y tecnológicas fundamentales de la arquitectura de Internet. No se puede concebir una red del futuro sin que adopte casos más generales de características disruptivas y altas demoras (hoy considerados casos de falla). De haberlo hecho de un comienzo hoy las redes celulares, las motocicletas africanas, las redes de sensores, y las satelitales estarían mucho mas integrados en un sistema que permita la interoperación de estos sub-sistemas de comunicaciones de manera mas transparente.

1.3.2.5. El Surgimiento de las Redes DTN

En general, ya sea por la complejidad asociada a los sistemas de redes satelitales de baja órbita, o la imposibilidad física en las comunicaciones geoestacionarios y de espacio profundo, la conectividad extremo a extremo ha resultado difícil de adaptar al dinamismo del entorno espacial [5]. Como se introdujo previamente, esto se debe a que la intermitencia en los enlaces genera pérdida de paquetes la que debe ser evitada a cualquier costo si se pretenden usar protocolos de Internet. En particular, el protocolo TCP [24] garantiza confiabilidad al retransmitir los datos perdidos con una menor tasa de datos, pero si la pérdida es severa, la sesión es perdida y un mensaje de error es enviado al usuario. La

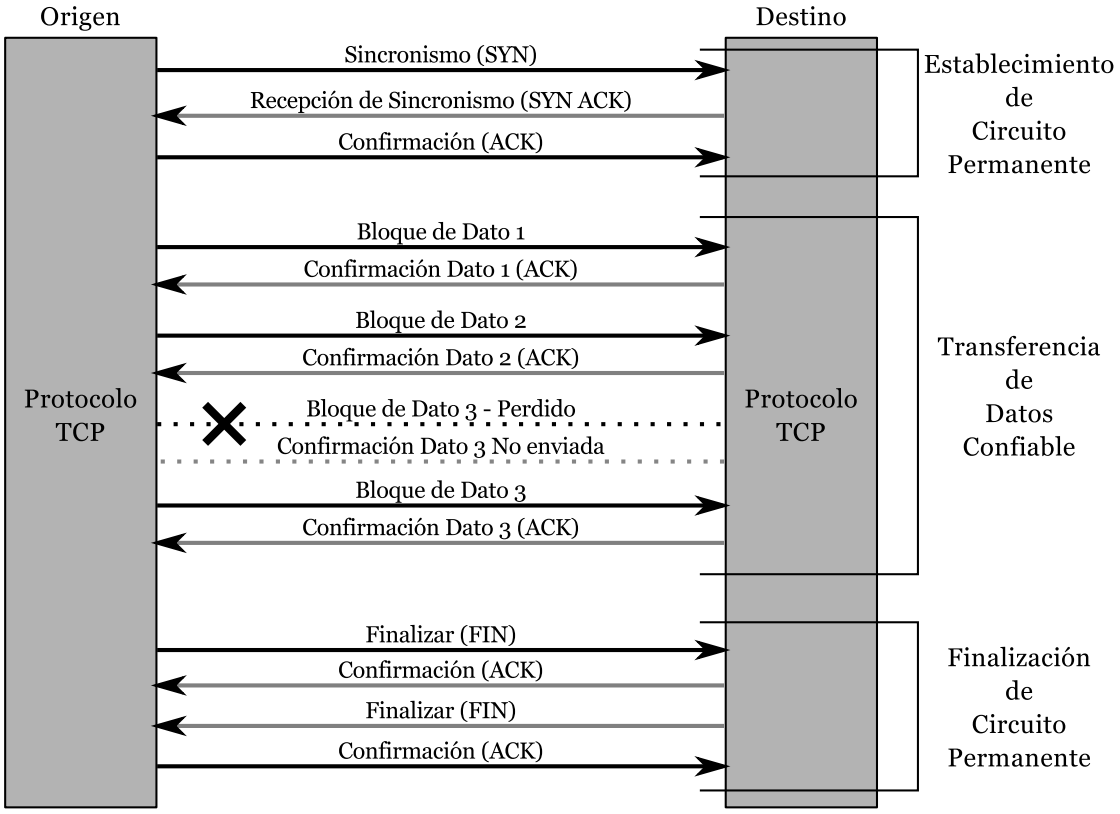


FIGURA 1.11: Funcionamiento Conversacional del Protocolo TCP

Figura 1.11 ilustra los intercambios de mensajes típicos de una sesión TCP en el que la pérdida de datos es recuperada. Sin embargo, los protocolos basados o similares a TCP resultan altamente conversacionales y siempre impondrán una barrera tecnológica hacia aquellas comunicaciones disruptivas como las redes de sensores inalámbricas operando en el espacio exterior [25, 39].

En particular, las comunicaciones que no cumplen con el principio de conectividad permanente permanecen fuera de Internet formando redes independientes con protocolos de comunicaciones especializados y por ende incompatibles. En consecuencia los nodos de estas redes son eficientes en comunicarse entre ellos pero no necesariamente con otras redes. De esta manera, cada sistema tiene sus propiedades únicas de tiempo de retorno (*round-trip-time* en Inglés o RTT), asimetría en los enlaces, tasa de errores, confiabilidad, mecanismos de calidad de servicio, entre otros. Las comunicaciones espaciales son un caso particular de estas redes, pero otras como redes militares en el campo de batalla, redes submarinas donde la propagación de la señal es significativamente mas lenta que en el aire, redes remotas de gestión ambiental y de fauna, entre otras requieren de un agente traductor de protocolo en caso de que requieran interoperar entre sí como se ilustra en la Figura 1.12.

Este tipo de redes disruptivas han sido recientemente estudiadas dentro del área de redes

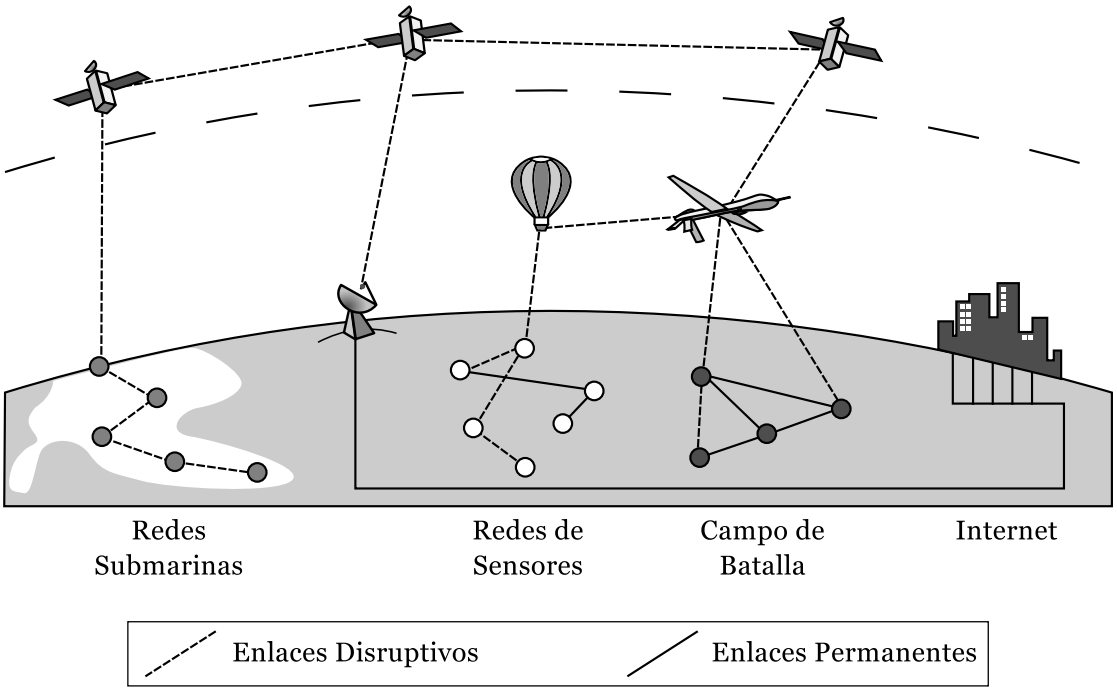


FIGURA 1.12: Redes de comunicaciones esporádicas integradas a redes de conexión permanente

de computadoras y comunicaciones espaciales bajo el nombre de redes tolerantes a demoras y disrupciones o *Delay and Disruption Tolerant Networks* (DTN) [40] en Inglés en búsqueda de buscar estrategias comunes que permitan resolver los problemas de esporacidad de los enlaces de manera eficiente y al mismo tiempo garantizar interoperabilidad entre las implementaciones DTN.

1.3.3. Redes Tolerante a Demoras y Disrupciones

1.3.3.1. Orígenes, Aplicaciones, y Avances

De acuerdo a lo discutido en la Sección 1.3.2, hacia principios del año 2000 existían una serie de sistemas de comunicaciones incapaces de utilizar las soluciones existentes de Internet para resolver la transferencia de información en un contexto disruptivo. En consecuencia, en el 2003, K. Fall [40] acuña la primera definición de Delay and Disruption Tolerant Network (DTN) o redes tolerantes a demoras y disrupciones con el fin de lograr la estandarización e interoperabilidad entre estos sistemas de conexiones esporádicas. Uno de los líderes de esta iniciativa fue (y es) V. Serf, el creador de TCP/IP, quien justamente luego de fundar Internet al unificar redes en los 60s, hoy busca repetir el logro expandiendo Internet para incluir demoras y disrupciones como parte de la esencia del sistema. Desde entonces, DTN ha recibido gran atención de diferentes ámbitos de la comunidad científica y tecnológica internacional.

En general, una DTN es una red de redes mas pequeñas (incluyendo Internet) que permite que las mismas estén interconectadas por medio de una estrategia de superposición (*overlay*) adaptando diferentes protocolos y características subyacentes. En consecuencia, las DTN ha recibido una importante atención durante los últimos años al ser propuestas para entornos donde las comunicaciones se encuentren en situaciones de demora extrema, ancho de banda limitado, errores, o problemas de inestabilidad [40]. Originalmente las DTN fueron estudiadas para implementar soluciones de comunicaciones interplanetarias (IPN), aunque recientemente han sido reconocidas como una solución válida para aplicaciones satelitales disruptivas [34] al poder lidiar con canales intermitentes típicos de redes de baja órbita (LEO) [41].

Recientemente, la formación de un grupo de trabajo en la Internet Engineering Task Force (IETF) con foco en este tipo de redes tolerante a demoras y disrupciones (*delay and disruption-tolerant working group* o DTNWG) [42] promete estandarizar y acordar pautas y estrategias entre diferentes entes para extender las fronteras de Internet con el fin de incluir este tipo de comunicaciones intermitentes. Este nuevo grupo se adosa al ya existente *delay and disruption-tolerant research group* o DTNRG [43] quien desde el 2002 se encarga de generar los espacios de investigación necesarios para discutir el futuro de estas redes. Junto al DTNWG y DTNRG, también existe el *InterPlanetary Networking Special Interest Group* (IPNSIG) [44] focalizado especialmente en fomentar el desarrollo de un sistema de interplanetario de comunicaciones a nivel sistema solar, dentro del cual DTN se plantea como la arquitectura central.

Dentro de los avances significativos de DTN se destaca la especificación de su arquitectura en la *Request for Comments* (RFC) 4838 [45] la que define los principios, conceptos y objetivos generales sobre los cuales basarse para el desarrollo de tecnologías comunes a los diferentes escenarios disruptivos y con demora. Esta RFC describe los problemas específicos de aquellas redes cuyas características operativas y funcionales hacen que los protocolos tradicionales de Internet sean inviables o ineficientes, y propone conceptos y terminologías comunes. En consecuencia, y de acuerdo a esta arquitectura, se especifica el *Bundle Protocol* (BP) o protocolo Bundle descrito a continuación.

1.3.3.2. El Protocolo Bundle

Una vez acordada la arquitectura DTN en [45], el grupo de investigación DTN especificó el protocolo Bundle o Bundle Protocol (BP) en Inglés en la RFC-5050 [46] como un esquema de transporte de datos diseñado para sobreponerse a las limitaciones de la conectividad permanente. Este protocolo logra esto al ubicarse en la capa de aplicación del modelo OSI [47] en cada una de las redes constitutivas del sistema DTN formando

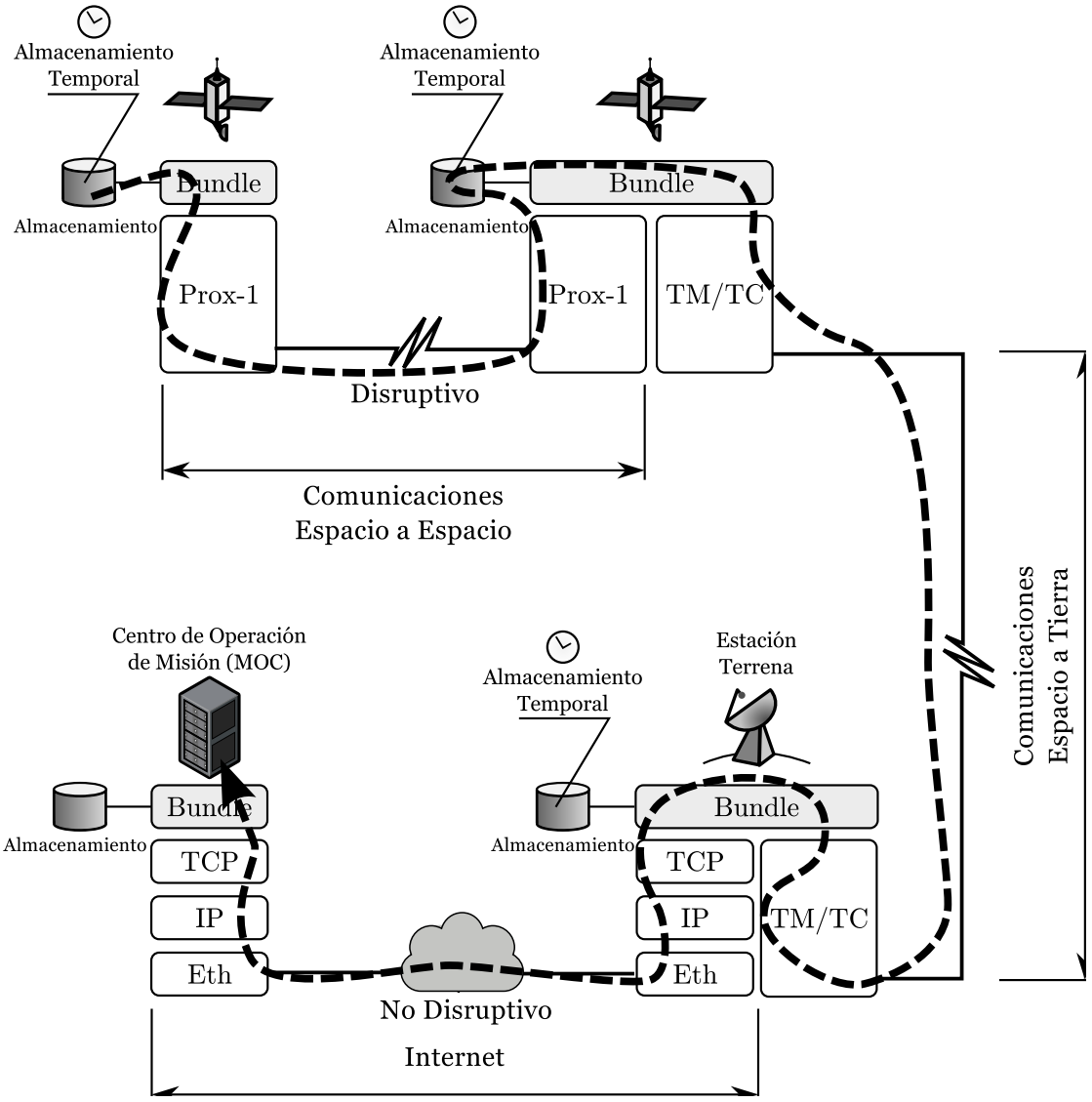


FIGURA 1.13: El Protocolo Bundle como una capa de superposición con capacidad de almacenar temporalmente

una red de superposición u *overlay* en Inglés. Esto se ilustra en ls Figura 1.13 donde el protocolo Bundle se posiciona por sobre un conjunto heterogéneo de protocolos de enlace, red, o transporte, interconectando redes de naturaleza diferentes. Esto permite utilizar el mismo agente de BP a lo largo de todo el sistema sin mayores necesidades de adaptaciones. Para lograr esto, las implementaciones del protocolo deben incorporar capas de convergencia o *Convergence Layers Adapters* (CLA) [48, 49] en Inglés quienes se encargan de adaptar los paquetes bundle a las interfaces de los protocolos subyacentes.

El protocolo Bundle para DTN se basa en el envío de mensajes asíncrono y oportunístico con bajas expectativas de conectividad extremo a extremo utilizando el principio de almacenar, transportar, y enviar (store-carry-and-forward o SCF en Inglés) información a medida que los enlaces se hacen disponibles. Este mecanismo no es innovador dado que es

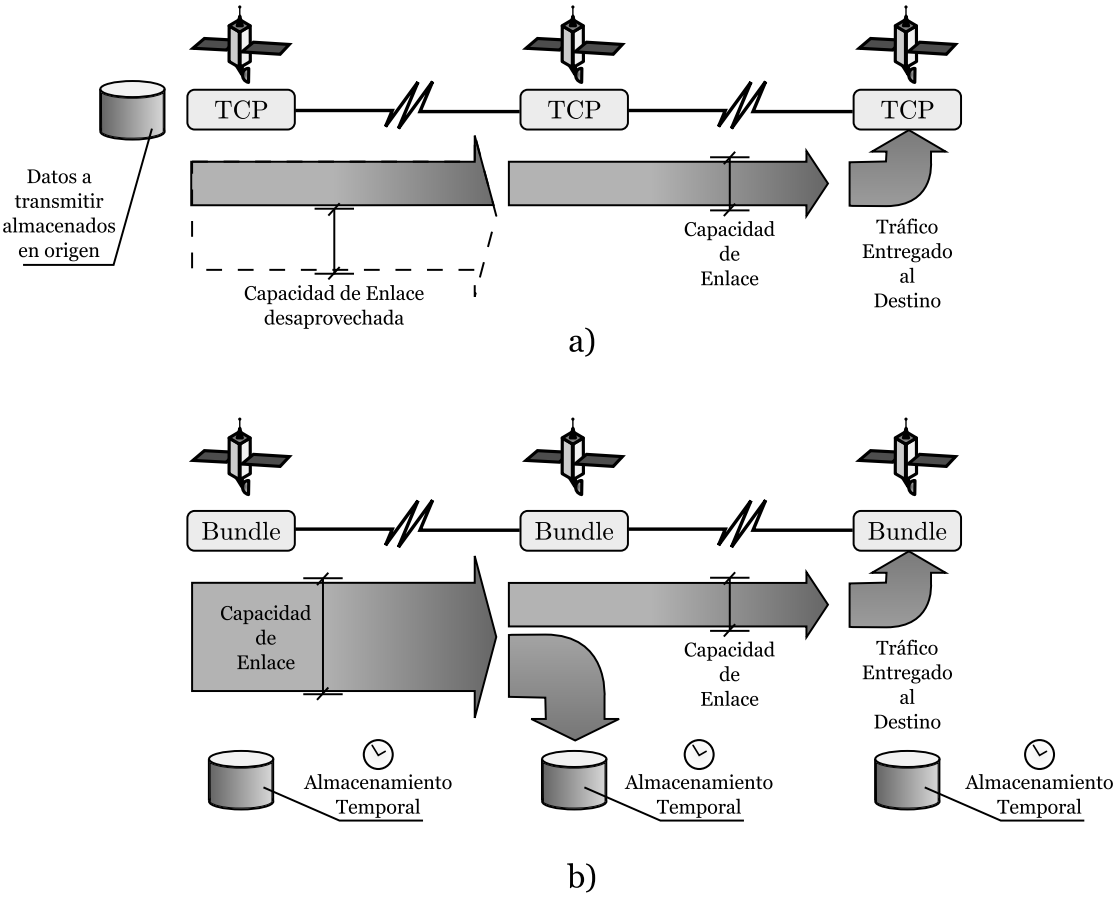


FIGURA 1.14: El Protocolo Bundle en (b)) hace un mejor uso de los recursos que TCP (a)) al utilizar almacenamientos persistentes

el esquema que utilizaron los sistemas postales por años donde los mensajes (o porciones de los mismos) son movidos y transportados de un lugar de almacenamiento a otro a lo largo de un camino determinado hasta que estos arriban a su destino. De alguna manera, este procedimiento es que hoy implementan los sistemas de mensajes de voz, *Whatsapp*, sistemas mensajes cortos (SMS) y correos electrónicos aunque con una topología estrella, es decir, que los nodos origen y destino contactan un único almacenamiento central que se encarga de distribuir los datos. Sin embargo, las DTN también contemplan casos donde los saltos de un almacenamiento a otro sea por medio de múltiples nodos como se ilustra en un caso de redes espaciales en la Figura 1.13.

Los espacios de almacenamiento en DTN pueden mantener datos de manera indefinida, por lo que se los denomina *almacenamientos persistentes* a diferencia de aquellos de corto plazo como utilizados por enrutadores de Internet. En general, los equipos de Internet almacenan paquetes por un período de tiempo acotado (del orden de los milisegundos) mientras esperan ser despachados por el enlace al próximo salto en su ruta. Dado que estos enlaces están permanentemente activos los paquetes nunca permanecen un tiempo significativo en estas memorias. Sin embargo, los enrutadores DTN como los ilustrados en

la Figura 1.13 requieren de almacenamientos persistentes dado que el enlace al próximo salto puede no estar disponible por un tiempo considerable (horas o días). Por otro lado, también puede suceder que un nodo vecino envíe datos a una tasa mayor de la que tiene el enlace al próximo salto. En este caso el nodo DTN puede almacenar en la memoria los paquetes excedentes haciendo un mejor uso de los recursos (TCP regularía la velocidad de todos los enlaces involucrados al mínimo). Este fenómeno se ilustra en la Figura 1.14 donde TCP hace uso de una comunicacion conversacional punto a punto, mientras que Bundle hace uso de los almacenamientos temporales intermedio mejorando el uso de los recursos. Finalmente, en caso de pérdida de bundles (paquetes de protocolo Bundle) por errores en el canal, el nodo que tiene almacenada la información puede volverse *custodio* de la misma garantizando un reenvío para asegurar la confiabilidad de la transmisión.

Esencialmente, y de acuerdo a [46] los paquetes de BP consisten en una cabecera, datos de usuario origen, parámetros de control que establecen como procesar dicha información, y una cola de trama opcional. En general, la longitud de los bundles no está limitada por la especificación, permitiendo generar paquetes de tamaño arbitrarios. Una vez encolado para ser transmitido por medio de un enlace, el protocolo subyacente (por ejemplo TCP, o Proximity) se encarga de fragmentarlo de manera acorde al enlace en cuestión. Sin embargo, existe la posibilidad de que el agente Bundle fragmente los bundles con el fin de fraccionar el envío de información para que la longitud de la misma se corresponda con el tiempo esperado del contacto esporádico en uso. La Figura 1.15 ilustra los campos del paquete.

Dado que los bundles pueden transitar enlaces con demoras significativas, la especificación del protocolo no obliga a utilizar confirmaciones de recepción, si no que las deja opcional para cada aplicación en particular (ver banderas opcionales en el formato en Figura 1.15). Por otro lado, la especificación permite la retransmisión de información perdida salto a salto por medio del uso de transferencia con custodia. Estas transferencias se acuerdan de antemano entre los nodos DTN. Si el nodo receptor es capaz de hacerse responsable del dato recibido (es decir, transformarse en custodio), este envía una confirmación de aceptación de custodia liberando al nodo origen de mantener una copia local del dato. En caso de no aceptar la custodia, el nodo origen puede retransmitir el bundle al mismo u otro nodo. De esta manera, la transferencia con custodia permite mejorar la confiabilidad de entrega de datos, aunque no necesariamente la garantiza [50].

Si bien la especificación del protocolo Bundle ha permitido dimensionar y probar conceptos de redes tolerante a demoras y disrupciones, el mismo aún cuenta con una serie de inconvenientes a resolver antes de poder ser implementado en redes operativas [50]. Entre ellos se encuentra la complejidad asociada a la cabecera y esquemas de direccionamiento, el uso de codificación SDNV (*Self-Delimiting Numeric Values*), la incorporación

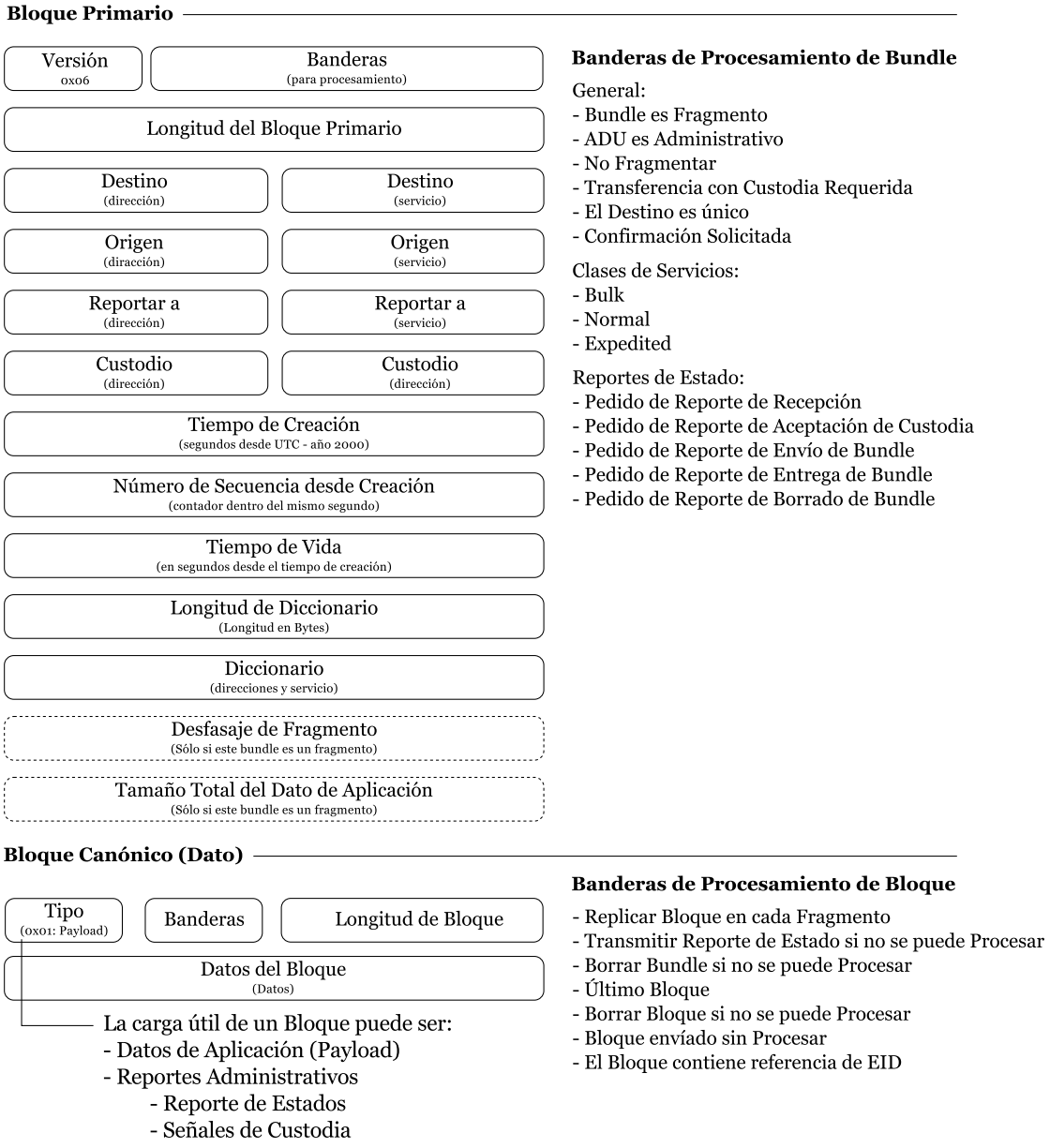


FIGURA 1.15: Campos y Formato de un Bundle de Acuerdo a la RFC5050

de un código de redundancia cíclica (CRC), vulnerabilidades de seguridad, entre otros. En efecto, el grupo de estandarización de DTN (DTNWG) [42] actualmente está discutiendo estas temáticas en foros abiertos para resolverlas en la próxima revisión de la especificación RFC5050 [46].

De esta manera, DTN está diseñada para soportar comunicaciones intermitentes entre redes con protocolos subyacentes heterogéneos al explotar el uso de almacenamientos temporales intermedios. En consecuencia, esta arquitectura permite incorporar nodos móviles cuya dinámica de trayectoria derive en particiones de la red que impliquen la pérdida de conectividad extremo a extremo con el destino final [38]. DTN tolera estos

eventos por medio del uso de un esquema SCF que permite aislar momentáneamente particiones del sistema que no pueden comunicarse entre ellas. Evidentemente, en este contexto, el sistema de comunicaciones de Internet resulta un caso particular de redes DTN donde todos sus componentes se encuentran permanentemente conectados. En otras palabras, la arquitectura DTN es una generalización de las comunicaciones tal como hoy las conocemos que incluye aplicaciones que hoy están fuera del concepto de Internet.

1.3.3.3. Enrutamiento en redes DTN

En general, las intermitencias se pueden clasificar en *oportunísticas*, *probabilísticas* o *planificadas* [45] y existen diferentes estrategias de enrutamiento de la información para cada una de ellas. Las comunicaciones oportunísticas son aquellas en que los nodos establecen enlaces esporádicos en tiempos que resultan impredecibles. Cuando dos personas, vehículos o aeronaves hacen contacto de línea de vista (*Line of Sight* o LOS en Inglés), los nodos pueden establecer el enlace y aprovechar e intercambiar información. De alguna manera, este modelo de comunicación es similar al que se usa verbalmente cuando nos encontramos con algún conocido. Existen al día de la fecha diferentes protocolos de ruteo que permiten el envío de datos bajo este tipo de condiciones con diferentes resultados de acuerdo al contexto en el que se aplican [51–53]. Dado que estos últimos se basan en la diseminación indiscriminada de copias de paquetes, son conocidos con el nombre de *enrutamientos epidémicos* en la literatura [54].

Sin embargo, de acuerdo a la naturaleza de la aplicación, la ocurrencia de oportunidades puede mostrar tendencias probabilísticas de acuerdo a algún modelo de movilidad de nodos [55, 56]. En este caso, existen algoritmos de cálculo de rutas DTN que explotan estas características de intermitencias probabilísticas [57] especificada en la RFC-6693 [58]. En general, estas estrategias utilizan reconocimiento de patrones de conexiones para establecer una historial de comunicaciones que permite predecir qué oportunidad tiene mas posibilidades de ocurrir en el futuro así como las características de la misma. En [57] se muestra que estas estrategias mejoran significativamente el desempeño de los ruteos epidémicos utilizados para redes oportunísticas.

Por otro lado, en el caso particular del espacio, prácticamente todo los objetos se encuentran en una situación de movimiento relativo, aunque con trayectorias generalmente predecibles. Por ende, los nodos DTN que se muevan en una determinada trayectoria orbital pueden predecir o ser informados de su futura posición de la cual a su vez se pueden derivar las oportunidades de comunicación de antemano. Esto permite a este tipo de nodos poder configurar sus equipos de comunicaciones apropiadamente con una

serie de beneficios (ahorro de energía, optimización de recursos, correcta configuración de parámetros de comunicación, etc.). Por otro lado, en el caso particular de comunicaciones interplanetarias, estas oportunidades pueden implicar el envío de mensajes entre nodos que no estén inicialmente en situación de contacto visual (LOS), pero que luego de que el mensaje viaje la distancia necesaria al destino, este último logrará estar en posición y condiciones de recibirlo. Dado el carácter de predecible de estas comunicaciones, el esquema de ruteo *Contact Graph Routing* o CGR fue especificado en [59], analizado en [60] y validado en vuelo en la misión DINET (descrita en la sección 1.3.3.5) probando que aprovechar la predictibilidad de estas redes puede aportar desempeños muy superiores a los mostrados por las estrategias epidémicas y probabilísticas. Explicaremos CGR en profundidad en el capítulo 6 donde realizamos algunos aporte de relevancia a este esquema. Por otro lado, también se destaca otro mecanismo de ruteo predecible denominado *Merugu's Floyd Warshall* (MFW) [61] basado en un sistema de cálculo y distribución de rutas centralizado que retomaremos en el capítulo 5 para los diseño de plan de contacto basado en rutas.

En general, las oportunidades de comunicación se conocen bajo el término de *contacto*, el cual en general se define por un nodo origen, un nodo destino, un tiempo de inicio, un tiempo de finalización y posiblemente otros datos específicos de las comunicaciones como ancho de banda, tasa de error de bits (*Bit Error Rate* o BER en Inglés), entre otros. Esta visión de comunicaciones predecibles es la que se adopta en el presente trabajo doctoral como medio para optimizar el uso de recursos de red entre satélites comunicados con el fin de soportar a la Arquitectura Segmentada discutida en la Sección 1.3.1.

1.3.3.4. Implementaciones

Al día de la fecha existen numerosas implementaciones del *stack* de protocolo Bundle con diferentes características y objetivos operativos. Entre los mas destacados, DTN2 [62] es la implementación de referencia de la arquitectura DTN definida en [45] por lo que tiene el objetivo de incluir todos los componentes detallados en la misma. En consecuencia, DTN2 provee una arquitectura de software flexible para poder realizar experimentos, modificaciones, y eventualmente (aunque no es su objetivo final) dar soporte a aplicaciones de Bundle Protocol en entornos operativos. Actualmente esta versión de software se provee para sistemas operativos basados en Linux y MAC OS X.

Por otro lado, el instituto *Institut für Betriebssysteme und Rechnerverbund* (IBR) desarrolló IBR-DTN [63] con el fin de ser una aplicación liviana, portable, y altamente modular para uso en plataformas limitadas en procesamiento como terminales móviles (celulares). En consecuencia, IBR-DTN incluye capas de convergencia (CLAs) para una

importante variedad de protocolos de enlace y otros como TCP, UDP, IEEE 802.15.4, etc. El código está diseñado para trabajar sobre Linux incluyendo distribuciones para dispositivos portátiles basados en Android y puede ser obtenido bajo licencia GNU de [64] directamente desde los repositorios de *Google*.

Además de DTN2 e IBR-DTN, existen otras implementaciones de menor relevancia en la comunidad como Postellation [65] o Java-DTN [66], entre otros que si bien están disponibles, en su mayoría, no han pasado de un estadío de prototipo.

Por último, pero no menos importante, existe una implementación de suma relevancia para el uso de DTN en el entorno espacial desarrollada por el *Jet Propulsion Laboratory* (JPL) de NASA con colaboración de la Universidad de Ohio, el *Johns Hopkins University Applied Physics Laboratory* (JHUAPL), y la Universidad de Colorado bajo la licencia Berkeley Software Distribution o BSD. El software *Interplanetary Overlay Network* (ION) [67] es probablemente el desarrollo del protocolo Bundle mas maduro de la comunidad para aplicaciones espaciales. Incluye CLAs para protocolos de *Consultative Committee for Space Data Systems* (CCSDS) como *Licklider Transmission Protocol* (LTP), *CCSDS File Delivery Protocol* (CFDP), y *Asynchronous Message Service* (AMS). ION está disponible bajo licencia GPL [68] y está diseñado para operar en sistemas embebidos de uso espacial soportando despliegues de alta velocidad y pocos recursos como aquellos usados en sistemas robóticos en el espacio exterior. Sin embargo, también ha probado desempeñarse adecuadamente en computadoras de escritorio tradicionales [69].

El sistema de ION está completamente escrito en el lenguaje C, con comunicaciones entre tareas por medio de memoria compartida, incluyendo un sistema de base de datos embebida, y mecanismos de *zero-copy* de objetos (ZCO) para mejorar el desempeño [67]. ION utiliza interfaces estándares Portable Operating System Interface para Unix (POSIX) para facilitar la portabilidad a sistemas operativos como Linux, OS/X, FreeBSD, Solaris, uClibc, VxWorks, RTEMS, y Windows. Actualmente, la versión 3.3.0 fue puesta a disposición en [68] con funcionalidades como Contact Graph Routing (CGR), compresión de cabecera de Bundle (Compressed Bundle Header Encoding o CBHE) [70], protocolo de seguridad de bundle (Streamlined Bundle Security Protocol o SBSP) [71], funciones de gestión de red (Delay Tolerant Network Management Protocol o DTNMP) [72], y numerosas capas de convergencias como TCPCL [73], UDP y LTP [74], entre otras.

Dado que ION puede considerarse como la implementación de referencia de Bundle para aplicaciones de DTN predecibles de uso espacial, se han realizado importantes estudios, análisis y demostraciones en órbita con este software. En consecuencia, dentro de los aportes secundarios de este trabajo doctoral, se realizó la implementación de una capa de convergencia de ION para el protocolo SpaceWire [75] el cual se está convirtiendo en el estándar *de-facto* para comunicaciones internas de los satélites [76]. En el trabajo

"Implementation and Evaluation of a Space-Wire Convergence Layer Adaptor" [12] se presento y discutió el valor de una implementación como esta para poder extender el uso del protocolo Bundle por sobre sistemas de cableados de redes de plataformas espaciales. Esto permite comunicarse por medio del direccionamiento DTN con diferentes sub-sistemas internos al mismo satélite. La implementación se validó en un System-on-Chip (SoC) basado en el procesador LEON-3-FT [77] sobre una FPGA con el sistema operativo RTEMS [78] mostrando resultados de impacto relacionados a las máximas prestaciones que se puede esperar de ION sobre esta plataforma tan popular para aplicaciones espaciales.

En efecto, a lo largo de esta tesis adoptamos una visión de red DTN inspirada en las funcionalidades incluidas en ION, particularmente en la capacidad del esquema de ruteo CGR de aprovechar la planificación de las futuras oportunidades de comunicaciones derivada de la trayectoria y característica de los sistemas inalámbricos de los objetos en el espacio.

1.3.3.5. Experimentos en Órbita

A pesar del reciente nacimiento del concepto de redes DTN y la especificación e implementaciones del Protocolo Bundle, así como el elevado costo que caracteriza las misiones espaciales, al día de la fecha existen dos experimentos del uso de esta tecnología en satélites reales cuyos resultados se describen a continuación.

Misión DINET En Octubre y Noviembre del 2008 el JPL de NASA realizó experimentos llamados *Deep Impact Network Experiment* (DINET) [79] en los que se validó el uso de DTN con ION a bordo de la sonda de espacio profundo EPOXY, la cual tenía el objetivo inicial de estudiar el cometa *Tempel 1* para luego sobrevolar el cometa Hartley 2 [80]. Una vez terminada la tarea de observación de EPOXY, la misión DINET toma lugar sin interferir con el objetivo principal a una distancia aproximada de 21 millones de Km (70 segundos luz aproximadamente) de la Tierra.

Debido al tamaño de las antenas y la energía disponible, las tasas de datos esperada a esta distancia rondaba los $256Kbps$ a $6Mbps$ de bajada (ciencia y telemetría) y $2Kbps$ a $1Kbps$ de subida de datos (comandos). Con el fin de no interferir con la misión principal, ION fue cargado en la memoria de una computadora de a bordo secundaria la cual completaba una topología de un satélite orbital del tipo repetidor (relay) y dos objetos en la superficie de Marte (*Mars* y *Phobos*) simulados en tierra [79]. Este experimento se ilustra en la Figura 1.16 donde estos dos últimos nodos se ubican físicamente en la

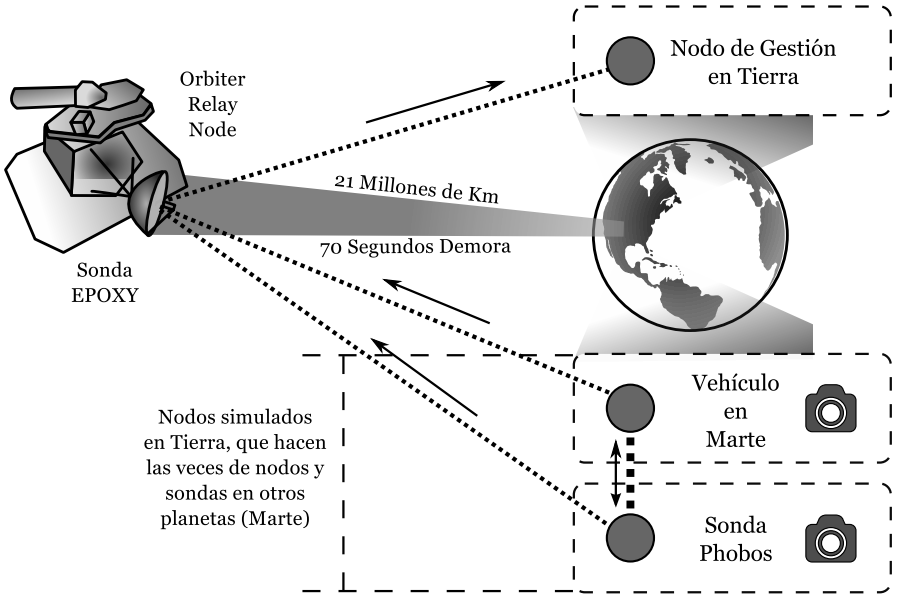


FIGURA 1.16: Experimento DINET realizado en la sonda de espacio profundo EPOXY

Tierra, aunque su comportamiento es idéntico al que tendrían en caso de estar sobre otro planeta.

Los resultados fueron positivos al lograr mantener el sistema operativo por un total de 4 semanas totalizando 8 pasadas sobre la red de espacio profundo (DSN) de NASA traficando un total de 292 imágenes (por un total de 14,5$MBytes$) desde los nodos con cargas útiles de adquisición de imagen. Se validaron capacidades de prioridades de tráfico y se verificó la entrega del mismo sin corrupciones. El uso de Bundle permitió operar sobre protocolos LTP en el caso de espacio profundo y TCP en las redes internas de NASA sin problemas de interoperabilidad, verificando y validando la arquitectura DTN [45].

Misión UK-DMC Por otro lado, en el contexto de órbitas bajas (LEO), se destaca la misión *Disaster Monitoring Constellation* (DMC), llevada a cabo por *Surrey Satellite Technology Ltd.* (SSTL) de Reino Unido (UK) [81]. En particular, la misión tiene el objetivo de tomar imágenes de la superficie del blogo para luego almacenarlas y bajarlas a tierra por medio de protocolos comerciales como IP [82] combinados con adaptaciones de TCP realizadas por el SSTL como el protocolo *Saratoga* [83], para el cual se desarrolló una capa de convergencia con resultados similares a los que proporciona LTP para el espacio profundo [84]. Sin embargo, Saratoga no ha tenido mayor transcendencia en la comunidad quien se ha inclinado por el uso del protocolo Bundle con una mayor promesa de interoperabilidad.

Cronológicamente, la misión UK-DMC fue la primera prueba del protocolo Bundle desde el espacio ya que la misma fue lanzada en 2003 con un router CISCO a bordo (*Cisco router in Low Earth Orbit* o CLEO) [85]. La plataforma provee una tasa de transferencia de 8,1$Mbps$ para bajada de ciencia y telemetría y 9,6$Kbps$ de subida de comandos. Sin embargo, dado que el espacio de memoria para el software del equipo de CISCO era limitado, se usó el stack DTN2 [62] en lugar del software ION [67].

Las pruebas de DTN realizadas en el UK-DMC permitieron bajar imágenes de 160$MBytes$ en dos bundles de 80$MBytes$ con *Saratoga* como capa de convergencia. Vale la pena aclarar que el tamaño de bundles utilizados en esta misión difieren de otras visiones que proponen un tamaño mas pequeño similar al esperado para un *datagrama* UDP del orden de unos pocos $KBytes$. En general, al día de la fecha se mantiene la discusión sobre cual es la mejor estrategia de fragmentación (proactiva o reactiva) [9]. Los científicos que realizaron este experimento documentaron comentarios menores sobre problemas en el planteo de Bundle [81] y proponen una alternativa basada en el protocolo de transferencia de contenido web HTTP [86]. Sin embargo esta propuesta data del 2008, momento desde el cual no ha logrado captar un interés significativo en la comunidad DTN.

1.3.4. Frontera del Estado del Arte

A pesar de que al día de la fecha el paradigma DTN cuenta con demostraciones en órbita funcionales con resultados exitosos como DINET y UK-DMC (tratados en la sección 1.3.3.5), estos se basan en arquitecturas de un nodo simple en el espacio. Es decir, a nivel experimental, nunca se ha implementado un sistema DTN multi-nodo o múltiple salto en un escenario orbital. Sin duda, un logro de este tipo dentro del programa de la Arquitectura Segmentada (tratada en la sección 1.3.1) pondría a la Agencia Espacial Argentina en una posición única en el marco global.

Sin embargo, y en general, las redes DTN están conceptualmente pensadas y diseñadas para poder automatizar el envío y recepción de datos por medio de múltiples nodos intermedios sin conexiones permanentes entre ellos [40] (problemática tratada en la sección 1.3.2). En particular, para el caso de aplicaciones espaciales, este envío se puede optimizar al conocer de antemano las trayectorias de los nodos orbitales [87]. De esta manera, en el caso de la aplicación de la Arquitectura Segmentada para observación terrestre, la tecnología DTN resulta de particular interés como medio genérico para enfrentar los desafíos de comunicaciones intermitentes de una constelación satelital.

En efecto, la implementación ION de NASA (revisado en la sección 1.3.3.4), hace uso de un plan de contacto en el cual se detallan las oportunidades futuras de comunicación. De esta manera el algoritmo CGR (introducido en la sección 1.3.3.3 y luego retomado

en el capítulo 6) permite utilizarlo para optimizar la entrega del tráfico en un sistema de múltiples nodos. Profundizaremos estos conceptos en el capítulo 2 donde sentamos las bases conceptuales para el desarrollo de esta tesis.

En resumen, si bien existe un antecedente conceptual e implementaciones de DTN para múltiples nodos, las mismas no han sido validadas en misiones reales en órbita. Por otro lado, dentro de las pruebas de emulación realizadas en la Tierra [34, 67, 87], todas determinan y utilizan directamente el plan de contacto de acuerdo a las factibilidades físicas por medio de propagadores orbitales [88] asumiendo que los nodos siempre contarán con los recursos necesarios para implementar dichas comunicaciones. Sólo existe a nivel académico algunos aportes en mecanismos para optimizar estos planes en término de costo [89] o confiabilidad [90], pero ninguno de ellos contempla las limitaciones de recursos en los nodos.

En consecuencia, y como se resume en la Figura 1.1 al comienzo de este capítulo, en esta tesis contribuiremos con una serie de mecanismos de diseño de plan de contactos que contemplen la limitación de recursos hasta hoy no considerada por otros trabajos que conforman el estado del arte. Como mostraremos en el capítulo 2, esta problemática se vuelve poco trivial para la gestión y optimización de un sistema DTN naturalmente limitado en recursos como la constelación satelital propuesta para la Arquitectura Segmentada.

Capítulo 2

Plan de Contactos y Restricciones de Recursos

2.1. Introducción

En Internet, la operación de protocolos se basa principalmente en el uso de información (rutas, direcciones, etc.) descubierta oportuna e inmediatamente en el momento en que la misma es requerida. Esto es posible gracias a la conexión permanente y libre de errores, demoras, y disrupciones. Sin embargo, en una red DTN, la obtención de información por medio del auto-descubrimiento podría tomar el tiempo suficiente para generar la pérdida de vigencia de los datos a transmitir. En lugar de esto, la operación de protocolos DTN puede aprovechar información configurada de antemano en los nodos etiquetada con los tiempos en la cual la misma se vuelve relevante. Un ejemplo de esta información son las oportunidades de comunicación producto de la trayectoria y orientación de los satélites.

En este capítulo definiremos formalmente estas oportunidades de comunicaciones bajo el concepto de contacto, topología de contacto, y plan de contactos. Luego introduciremos los criterios de modelado en los que nos basaremos para tratar la problemática de limitación de recursos contempladas en los mecanismos de diseño de planes de contacto. El paradigma, ideas, y conceptos que definimos en este capítulo fueron publicados en la revista IEEE Communications Magazine en Mayo del 2015 [5] bajo el título "Design Challenges in Contact Plans for Disruption-Tolerant Satellite Networks" y sientan las bases para luego comprender los esquemas de diseño detallados en los siguientes capítulos 3, 4, y 5 así como los aportes en el área de implementación del 6.

2.2. Plan de Contactos

2.2.1. Definición de Contacto

Dentro de la generalidad de las redes tolerante a demoras y disrupciones (DTNs), aquellas formadas por satélites orbitantes tienen la particularidad de que las trayectorias y orientaciones de los nodos constitutivos se vuelven predecible por medio del uso de modelos matemáticos precisos [88] combinados con información del sistema de control órbita (Attitude and Orbit Control System o AOCS en Inglés). Esto permite que al combinar esta información con modelos de los sistemas de comunicaciones implementados (potencia de transmisión, sensibilidad, ganancias y posición de antenas, etc.) se puedan determinar de antemano las oportunidades de comunicación o *contactos* [60].

En general, se entiende por *contacto* a la posibilidad de comunicación entre dos nodos DTN definida en un intervalo de tiempo en el cual un nodo origen puede enviar datos a un destino en determinada condición y forma. La Figura 2.1 ilustra dos contactos, el primero entre dos satélites y el segundo entre un satélite y una estación terrena. Implícitamente, la comunicación de datos se hará por medio de alguna capa de convergencia (CLA) tanto en el transmisor y en el receptor. Cada contacto se caracteriza por su tiempo de inicio, su tiempo de finalización, las identificaciones de los nodos transmisor y receptor involucrados, y otros datos de relevancia para la implementación de DTN. Por ejemplo, en el caso del *stack* de protocolos en ION [67], el contacto también incluye la tasa de datos que se puede lograr en ese intervalo. Sin embargo también se podría considerar otras informaciones como identificadores de antenas, modulaciones a utilizar, potencias de transmisión necesarias, entre otros.

2.2.1.1. Sobre los Esquemas de Múltiple Acceso

A lo largo de este trabajo, las comunicaciones entre satélites se considerarán punto a punto, dejando de lado la consideración de mecanismos de control de acceso al medio (Medium Access Control o MAC en Inglés) [91]. En general, estos últimos requieren de mecanismos de negociación autónomos que se basan en el intercambio de información entre los nodos participantes para negociar el acceso al medio compartido. Este fenómeno se suele realizar paquete en paquete en esquemas de Carrier Sense Multiple Access (CSMA) como WiFi y si bien puede resultar transparente en aplicaciones terrestres con distancias del orden de los $100m$ a $300m$, las demoras a la que las mismas deben estar sometidas en el espacio (limitadas por la velocidad de la luz) con distancias del orden de los $1000Km$ generan ineficiencias considerables y requieren de adaptaciones y evaluaciones actualmente en discusión en la comunidad [92, 93].

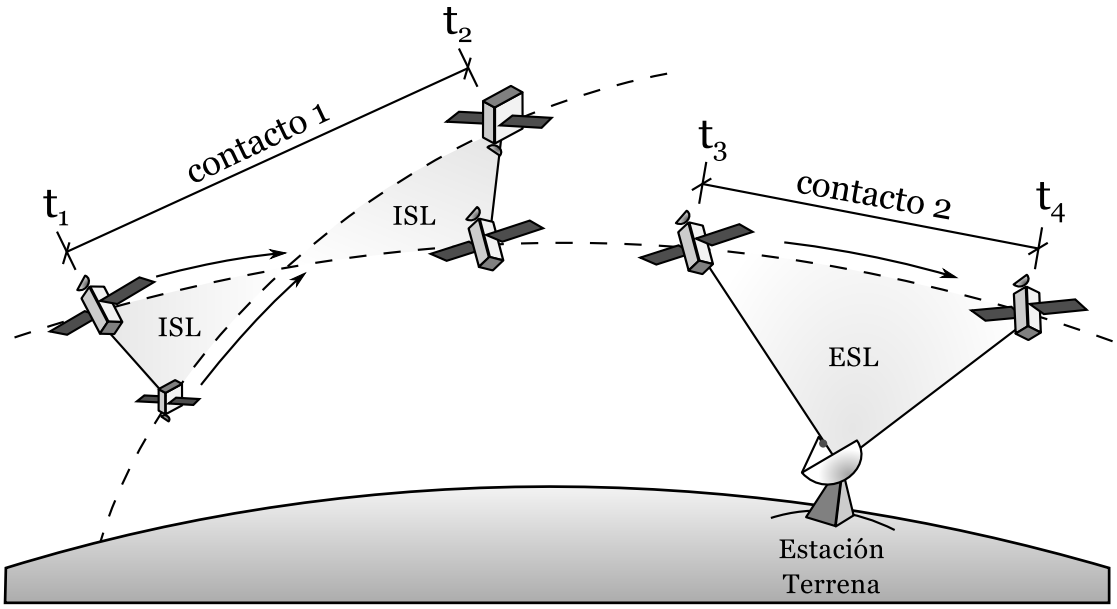


FIGURA 2.1: Ilustración de un contacto ISL y uno ESL

TABLA 2.1: Tiempos de propagación en función de las distancias

Distancia [Km]	Tiempo [us]
0.1	0.34
0.299	1
2.997	10
14.989	50
29.979	100
59.958	200
149.896	500
299.792	1000
1498.962	5000
2997.924	10000

Por ejemplo, la Figura 2.2 a) ilustra el impacto que tiene el tiempo de propagación de la señal en una transferencia normal de una trama entre un nodo A y B por medio de CSMA. Como se muestra en la Tabla 2.1, para distancias del orden de los $3000Km$, una demora de propagación de $10ms$ implicaría un desperdicio de $40ms$ (RTS, CTS, Datos y ACK) mas el tiempo de envío de datos por cada trama transmitida (alrededor de $1ms$). Esto limitaría la tasa de tramas a 25 por segundo que a un promedio de $2KB$ por trama el máximo teórico esperable de CSMA es de $390Kbps$ [94]. Por último, en caso de que un tercer nodo intente enviar datos, la probabilidad de colisiones y consecuente pérdida de datos se incrementa drásticamente llevando el sistema de comunicaciones a tener prestaciones sumamente limitadas a distancias típicas y nominales de los enlaces satélite-tierra (ESL).

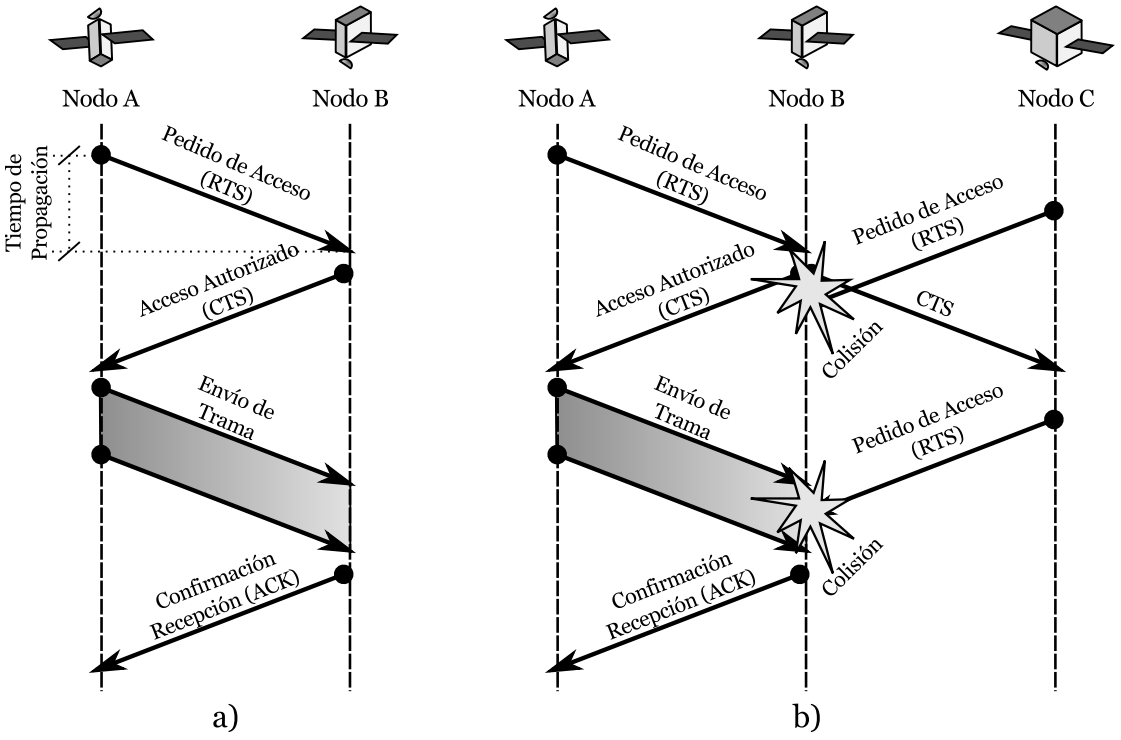


FIGURA 2.2: Impacto de los tiempos de propagación en a) la demora de negociación y b) la probabilidad de colisiones

Por otro lado, siempre existe la posibilidad de diseñar *clusters* de satélites cuyas distancias relativas sean los suficientemente cortas como para permitir el uso de estas tecnologías, aunque en general demandan sistemas de control de AOCS sumamente estrictos y complejos que requieren una significativa cantidad de combustible limitando la vida útil del sistema [95]. Además, una topología tan compacta impide aprovechar diversidades de adquisiciones como las descritas en la Sección 1.3.1.4.

De todas maneras, la visión de contacto como una abstracción de las comunicaciones puede utilizarse directamente en esquemas de múltiple acceso por medio de solapamientos. Por ejemplo, un nodo A puede implementar dos contacto simultáneos con dos vecinos B y C en tiempos solapados con tasa de datos que representen las capacidades que el enlace podría ofrecer con tres nodos compartiendo el acceso. Sin embargo la aleatoriedad asociada de estos esquemas depende específicamente del equipamiento de comunicaciones utilizado. En general, dada la naturaleza predecible del sistema que se está diseñando, implementar mecanismos aleatorios de este tipo podría resultar ineficiente y de utilidad cuestionable. En consecuencia, en este trabajo doctoral asumiremos comunicaciones punto a punto a menos que se indique lo contrario.

2.2.2. Definición de Topología de Contacto

Una vez definido el concepto de *contacto*, es necesario agrupar aquellos que se puedan llegar a predecir en un cierto intervalo de tiempo para poder distribuir esta información de topología a los nodos DTN. Esta agrupación recibe el nombre de *topología de contacto* la que suele expresarse como una lista de contactos para configurar el software ION [67] quien a su vez puede aprovechar esta información para tomar decisiones de enrutamiento efectivas. Sin embargo, como explicamos en la Sección 2.3, existen otros métodos para poder expresar y caracterizar las topologías y planes de contactos.

En general, el intervalo de tiempo que se debe considerar para agrupar estos contactos varía en cada escenario y suele depender de la precisión de los modelos de propagación y de la anticipación de las operaciones de la misión. Sin embargo para una configuración típica de constelaciones de baja órbita es factible considerar topologías de contacto de una semana de duración, que en caso de necesidad puede ser actualizado a medida que pasa el tiempo y si se cuenta con predicciones mas precisas. De esta manera, considerar un margen de tiempo a futuro permite contar con una planificación de emergencia en caso de que no se pueda distribuir una nueva versión de la misma.

2.2.3. Definición de Plan de Contacto

A pesar de que la topología de contactos definen todas las posibilidades físicas de comunicaciones entre nodos dadas sus características de antenas, potencia, y canal de comunicaciones, fenómenos ajenos a estos pueden causar que este contacto no se puede o no se quiera concretar en el sistema real. Entre estos se destacan la disponibilidad de recursos y energía, arquitectura de nodos, o simplemente una decisión operativa de no transmitir sobre cierta región geográfica como describimos luego en la sección 2.4.

En consecuencia, una vez determinada la topología de contactos, la misma debe ser diseñada y adaptada para que la misma pase a ser un *plan de contacto* capaz de ser implementado en la red satelital con los recursos que esta realmente dispone. En general, el plan de contacto es un subconjunto de los contactos expresados en la topología de contactos, donde algunos de los intervalos asociados a cada oportunidad de comunicación pueden inclusive ser reducidos.

Por ejemplo, considerando el ejemplo mostrado en la Figura 2.1, puede pasar que condiciones externas a las posibilidades físicas restrinjan dichas oportunidades derivando un en plan de contacto diferente de la topología de contacto como se ilustra en la Figura 2.3. Asumiendo que en este escenario ilustrativo $t_1 = 100$, $t_2 = 400$, $t_3 = 500$, y $t_4 = 800$ segundos, y que el contacto C_1 sólo se puede utilizar parcialmente dado limitaciones de

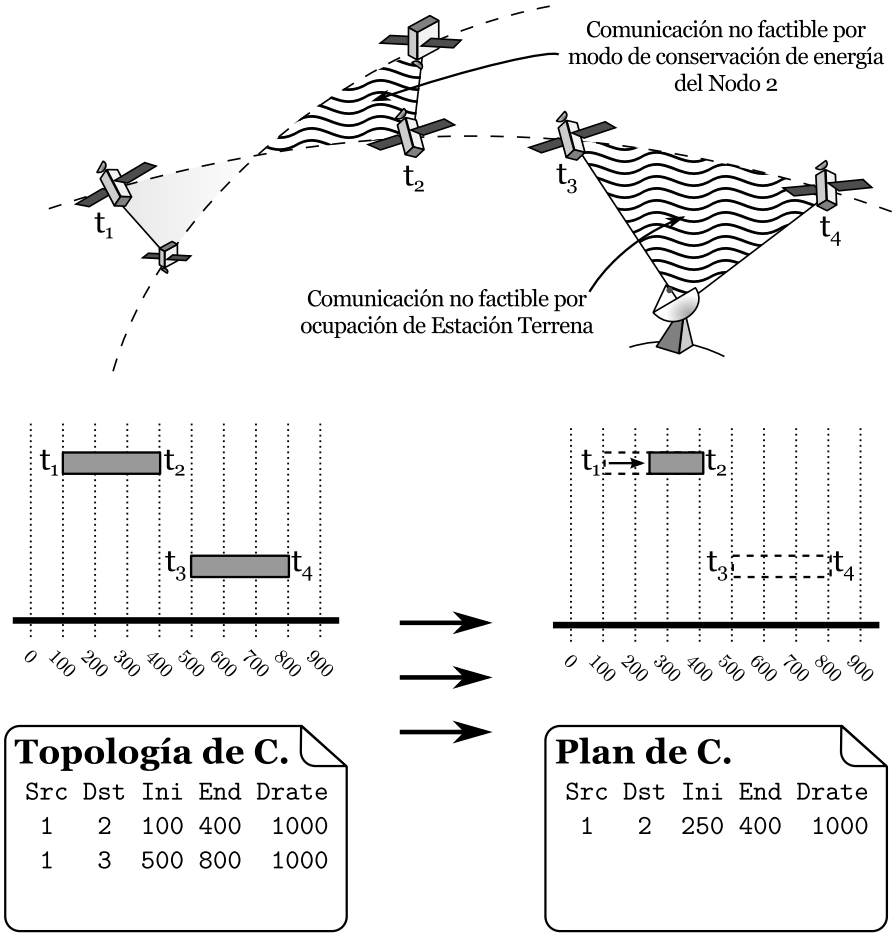


FIGURA 2.3: Diseño de Plan de Contacto

energía del nodo N_2, y que el contacto C_2 no se puede utilizar por que la estación terrena no está disponible, se diseña un plan de contacto acorde a las capacidades del sistema.

Sin embargo, y como veremos en la sección 2.4, las restricciones de diseño del plan de contacto no son necesariamente tan triviales y tienen un impacto significativo en cómo el tráfico del sistema circulará a través de los contactos planificados.

2.2.3.1. Implementación de los Planes de Contactos

Tradicionalmente, el centro de control de misión o MOC es el encargado de planificar las comunicaciones, la gestión de la plataforma y el uso de las cargas útiles del satélite. En el caso de múltiples nodos, el MOC también podría ser el encargado de predecir y gestionar los contactos entre satélites (ISLs) y entre satélite y Tierra (ESLs) que se generarán en el sistema orbital creando la *topología de contactos* como se ilustra en la primera etapa de la Figura 2.4.

Como se explicó en la sección 2.2.3, a diferencia del plan de contactos, la *topología de contactos* enumera todas las posibilidades de comunicaciones físicamente posibles dada las configuraciones de antenas y característica de radio frecuencia ignorando consideraciones de recursos del sistema. Casualmente el MOC es el elemento de red que tiene el conocimiento mas actualizado del estado de los sub-sistemas de los satélites ya que es el último receptor de la telemetría de los mismos. Además desde este centro se gestionan las cargas útiles del sistema, dándole una capacidad única de predecir el tráfico que se requiere cursar por la red. En consecuencia el mismo MOC tendría la facultad e información necesaria para transformar la topología de contactos en el plan de contacto final como se ilustra en la segunda etapa de la Figura 2.4. Los capítulos 3, 4 y 5 de este trabajo describen los aportes realizados (metodologías) en esta etapa de diseño.

Una vez diseñado el plan de contacto, es necesario distribuirlo entre los nodos DTN del sistema para que los mismos puedan basar sus futuros cálculos de ruta en ellos. Este fenómeno se ilustra en la tercer etapa de la Figura 2.4 donde el segmento terreno envía este plan a cada satélite ya sea utilizando la misma red DTN existente y planificada anteriormente (en-banda), o por un canal externo a la misma (fuera de banda) por medio de comunicaciones punto a punto similares a las tradicionalmente utilizadas en satélites monolíticos. En caso de ser en-banda, protocolos como Contact Plan Update Protocol (CPUP) [96] permiten hacerlo con paquetes Bundles. Por otro lado, el mecanismo fuera de banda es el que probablemente se deba utilizar en la etapa de acondicionamiento de la red una vez puesta en órbita por primera vez.

Finalmente, una vez que cada satélite tiene la información que describe el comportamiento de la red a lo largo del tiempo, los mismos pueden aprovecharla para tomar decisiones eficientes respecto al enrutamiento de los datos originados por ellos mismos o recibidos de otro vecino. Como se muestra en la cuarta fase de la Figura 2.4, conocer de antemanos los contactos locales y remotos (es decir, aquellos que experimentarán los nodos vecinos) permite calcular caminos óptimos para los datos en el nodo local. El criterio de optimalidad suele ser el menor el tiempo de entrega al destino final como en el caso del esquema de CGR [60], aunque el mismo puede variar en otras aplicaciones. Así, el nodo que calcula este camino óptimo puede o bien simplemente enviar el paquete al próximo vecino [59] o incorporarle al mismo el vector de ruta calculado para que evitar que el próximo salto tenga que efectuar nuevos cálculos [60, 97]. El capítulos 6 (implementación) de este trabajo discutimos importantes consideraciones respecto a este etapa.

De esta manera, se completa el ciclo de implementación de plan de contactos desde su gestación hasta su utilización final para dar soporte al cálculo de rutas en cada uno de los nodos DTN del sistema. Luego, en la sección 2.3 haremos énfasis en técnicas necesarias

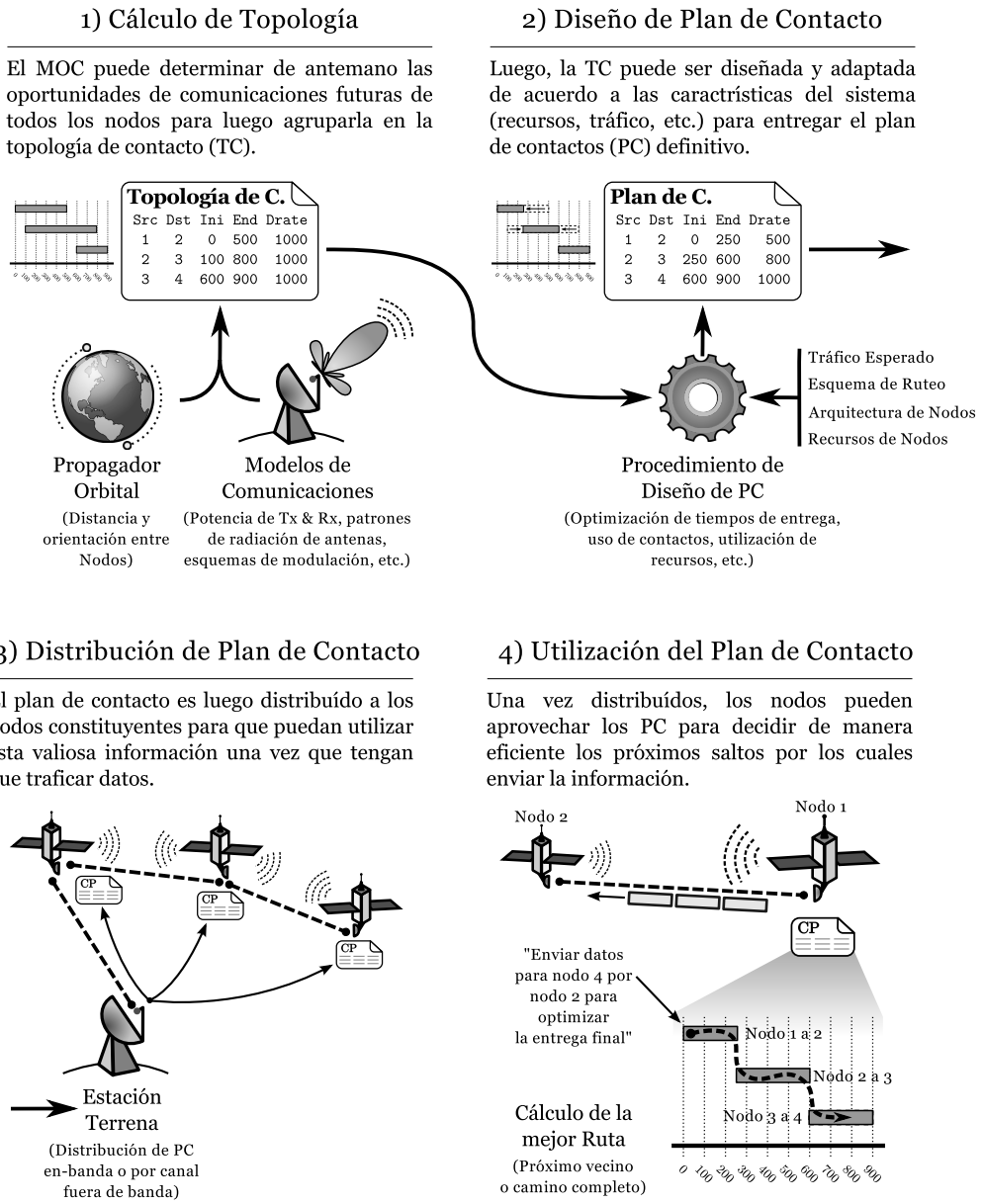


FIGURA 2.4: El procedimiento de creación, diseño, distribución e implementación de planes de contactos

para poder diseñar efectivamente planes de contactos (etapa 2 de la Figura 2.4), tema central de esta tesis doctoral. Finalmente, se realiza algunos aportes a la problemática de la congestión y calculo de ruta en la implementación (etapa 4) en el capítulo 6.

2.3. Modelado de Plan de Contactos

Antes de poder atacar el problema del diseño de plan de contactos es necesario contar con una estrategia de modelado y abstracción para representar y trabajar sobre los mismos. Por ende, en esta sección planteamos dos estrategias de modelado: una basada en un

planteo de máquina de estado finito, y otra similar a la lista de contactos descrita en la sección 2.2. Con el fin de ilustrar estas estrategias nos basaremos un caso de estudio y referencia simple pero de relevancia para la Arquitectura Segmentada que usaremos a lo largo de los subsiguientes capítulos para demostrar los aportes realizados en esta tesis doctoral.

2.3.1. Caso de Referencia y Estudio A: Topología Escalera

En esta sección se presenta un caso de referencia y estudio de interés para la observación terrestre y relevante para la Arquitectura Segmentada planteada en el capítulo 1. Se utilizará este ejemplo para describir el modelado y luego las formulaciones y análisis del problema de diseño de plan de contactos. A lo largo de la tesis iremos aportando otros casos de estudios de relevancia como el caso de referencia y estudio B (topología lineal ecuatorial) en la sección 3.4.3 del capítulo 3 y el C (topología en tren) en la sección 4.4.3 del capítulo 4.

Se plantea entonces un red de 4 satélites de órbita baja (LEO a $600Km$), heliosíncrona (plano orbital constante respecto al sol) y polar [88] los cuales cuentan con sensores de observación terrestre (pueden ser tanto activos o pasivos) como cargas útiles. El sistema de comunicaciones para esta constelación debe ser del tipo tolerante a demoras para tolerar y anticipar disrupciones en contactos con la tierra así como entre los mismos satélites. Los parámetros orbitales precisos que describen la trayectoria de estos objetos orbitales se listan en la Tabla 2.2 y están deliberadamente diseñados para que un satélite orbite al frente de otro en sentido del vector velocidad separados por un ángulo de 5° de argumento de perigeo entre cada uno. Además, también están distanciados por 5° de *Right Ascension of the Ascending Node* (RAAN) lo que provoca que también se separen perpendicularmente al vector velocidad sobre zonas cercanas al Ecuador, pero que recuperen la formación lineal hacia la zona de los polos.

La Figura 2.5 muestra dos imágenes 3D obtenidas con *GLOrbit*, una herramienta especialmente diseñada para estudiar y analizar topologías de constelaciones para este trabajo de investigación (descrita en la sección 2.6.6). GLOrbit incorpora una implementación del propagador SGP-4 [88] ampliamente utilizado en la comunidad que determina la posición (x, y, z) y velocidad (v_x, v_y, v_z) de un objeto orbitante en el en marco de referencia inercial centrado en la tierra (Earth Center Inertial o ECI en Inglés). Si bien este propagador es lo suficientemente preciso para los fines de este trabajo, un MOC de una misión real podría utilizar propagadores de mayor precisión como el HPOP (High Performance Orbital Propagator) disponible en la popular herramienta comercial Satellite ToolKit (STK) [98] de la empresa AGI.

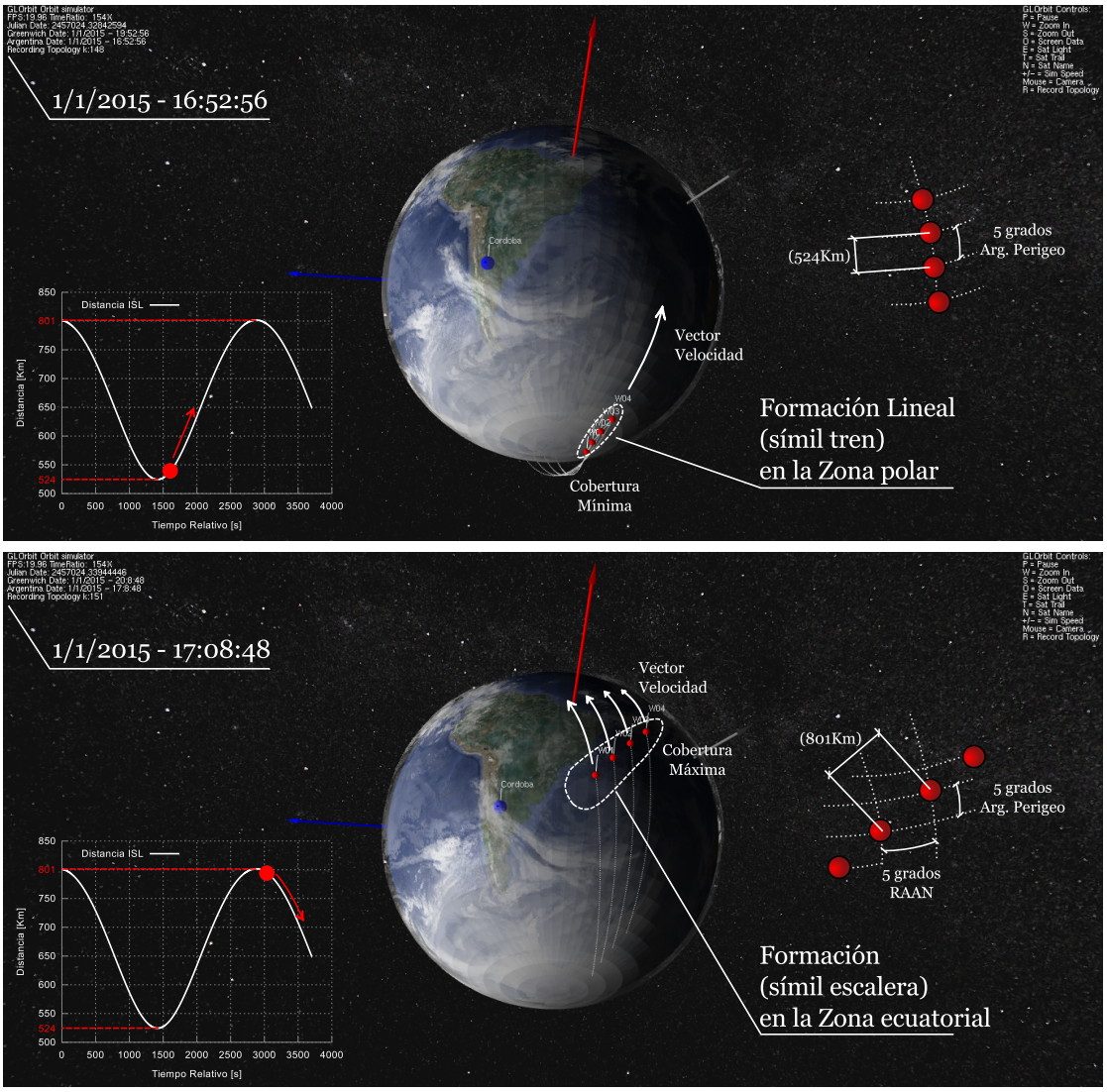


FIGURA 2.5: Caso de estudio de constelación de 4 satélites (Imagen de la herramienta GLOrbit).

Ambas figuras muestran dos momentos sucesivos de la disposición espacial de los satélites en órbita al pasar por el polo sur y luego por la zona ecuatorial. En la primera los segmentos de vuelo se alinean prácticamente en línea recta a una mínima distancia relativa de $524Km$, para luego distanciarse paulatinamente hasta los $801Km$ en la zona del ecuador. A medida que aumenta la distancia, mayor es la cobertura que los sensores de a bordo tienen sobre la superficie terrestre. En particular, este fenómeno se da sobre zonas pobladas aumentando la utilidad de observación de la constelación propuesta. Finalmente, el mismo ciclo se vuelve a repetir en el polo norte y en la otra cara de la tierra. El trayecto de polo a polo toma 2835 segundos (poco mas de 47 minutos), por lo que el ciclo de órbita completa lleva 5670 segundos o 14,92 revoluciones por día estelar (Movimiento Medio).

Dada las condiciones físicas de esta constelación, se propone un sistema de comunicación

Tabla 2.2: Tiempos y Parámetros Orbitales del Caso de Referencia y Estudio

Inicio del Intervalo de Topología	Ene-1st, 2015, 0hs 0min 0sec
Fin del Intervalo de Topología	Ene-1st, 2015, 3hs 22min 36sec
Coeficiente Bstar (/ER)	0
Inclinación (grados)	98°
RAAN (grados)	0°, 5°, 10°, y 15°
Eccentricidad	0
Argumento del Perigeo (deg)	180°, 185°, 190°, 195°
Anomalía Media (deg)	0°
Movimiento Medio (rev/day)	14,92 rev/day
Altura sobre el nivel del Mar (Km)	600 Km

ISL cuyo rango de alcance omnidireccional máximo permite comunicarse con un nodo vecino a una distancia de $700Km$. En general, el alcance es producto de un presupuesto de enlace (*link budget* en Inglés) el que considera potencia de transmisión, ganancia de antena transmisora y receptora, sensibilidad del receptor, y detalles de ancho de banda, modulación y esquema de corrección de errores. En este ejemplo asumimos que la combinación de estos parámetros permiten mantener una comunicación a una tasa de $1Mbps$ bi-direccional (*full-duplex* en Inglés). Además, las comunicaciones ESL son por el momento dejada de lado pero la estación terrena simplemente debe considerarse un nodo mas en la red capaz de generar contactos son los satélites.

Cabe destacar que dado que la distancia entre satélites oscila entre $524Km$ y $801Km$ como se ilustra en la Figura 2.5, los enlaces ISLs tendrán estados de conexión y desconexión periódicos al entrar y salir del rango de $700Km$, por lo que este sistema no podría ser considerado para una red permanente conectada. Sin embargo, considerar la arquitectura DTN aporta una flexibilidad sin precedentes en permitir considerar constelaciones mas distribuidas y de sistemas de comunicaciones mas limitados para la transferencia de datos. Esto último es de sumo valor para agencias en países en vía de desarrollos al permitirle minimizar complejidades y por ende costos y esfuerzos.

Finalmente, propagamos esta topología por un tiempo total de 3 horas, 22 minutos, y 33 segundos lo que permite que la constelación sobrevuele 4 veces por los polos (inicialmente por el polo Norte) completando dos órbitas sobre la Tierra. Es importante notar que a pesar del tiempo acotado considerado, y la simplicidad de red (sólo 4 nodos), enfrentaremos importantes desafíos pocos triviales a la hora de diseñar y modelar el plan de contacto.

2.3.2. Modelado como Máquina de Estados

Inicialmente planteamos una técnica de modelado basada en el paradigma de máquina de estados o autómata finito o Finite State Machine (FSM) en Inglés. Tanto esta estrategia de modelado como la de lista de contacto descrito en la sección 2.3.3, son aplicables tanto a la topología de contacto como al plan de contacto.

Esencialmente este modelo se desplaza de un estado a otro a medida que avanza el tiempo de órbita. Por ejemplo, consideremos el primer vuelo por sobre los polos (media órbita) de la trayectoria de la red de 4 satélites planteada en la sección 2.3.1 ilustrada en la Figura 2.6 a). La naturaleza evolutiva de los contactos entre los segmentos puede ser capturada por medio de grafos cuyos vértices y arcos simbolizan los nodos DTN y sus oportunidades de establecer enlaces inalámbricos en un estado dado respectivamente.

En consecuencia, la topología puede ser representada y discretizada por un conjunto de intervalos de tiempos o puntos de división k de la forma $[t_k, t_{k+1}]$ como se muestra en la Figura 2.6. Se puede observar que cada estado tiene un grafo simbolizando las oportunidades de comunicación en ese intervalo de tiempo de duración $i_k = t_{k+1} - t_k$. De esta manera, la formulación del tiempo en el modelo FSM del plan de contactos (o topología) puede ser codificada en dos matrices $T = \{t_k\}$ y $I = \{i_k\}$ de tamaño K que representan el tiempo de inicio y duración de cada estado k respectivamente. Por último, una matriz $C = \{c_{k,i,j}\}$ define las características de todos los arcos entre los nodos i y j para cada estado k. En particular, $c_{k,i,j}$ tendría un valor equivalente a la capacidad de traficar datos en el arco i-j en el intervalo $i_k = t_{k+1} - t_k$, siendo igual a $c_{k,i,j} = 0$ en el caso de que la oportunidad de comunicación sea inexistente.

En general, para cada inicio o fin de contacto, existe una evolución de estado de k_a a k_{a+1} en la máquina de estados. Por ende, y debido a que un contacto dado puede extenderse a lo largo de varios estados, denominaremos *arcos* a la comunicación entre nodos cuando nos refiramos a un estado k en particular. En el intervalo de topología propuesto, un total de 7 estados k son suficientes para describir la topología de contacto durante la primer media órbita del sistema de referencia. El modelo FSM resultante se ilustra en la Figura 2.6 b) donde se puede notar que el estado k_4, de 1458 segundos de duración, representa la formación o topología en forma de tren que se da cuando la constelación se alinea sobre el polo norte con contactos posibles entre N_1 a N_2, N_2 a N_3 y N_3 a N_4.

2.3.2.1. Sobre el Fraccionamiento de Estados

En el caso de querer diseñar un plan de contacto partiendo de la topología de contacto expresada en FSM como se realiza en la Figura 2.3, se debe tener especial consideración

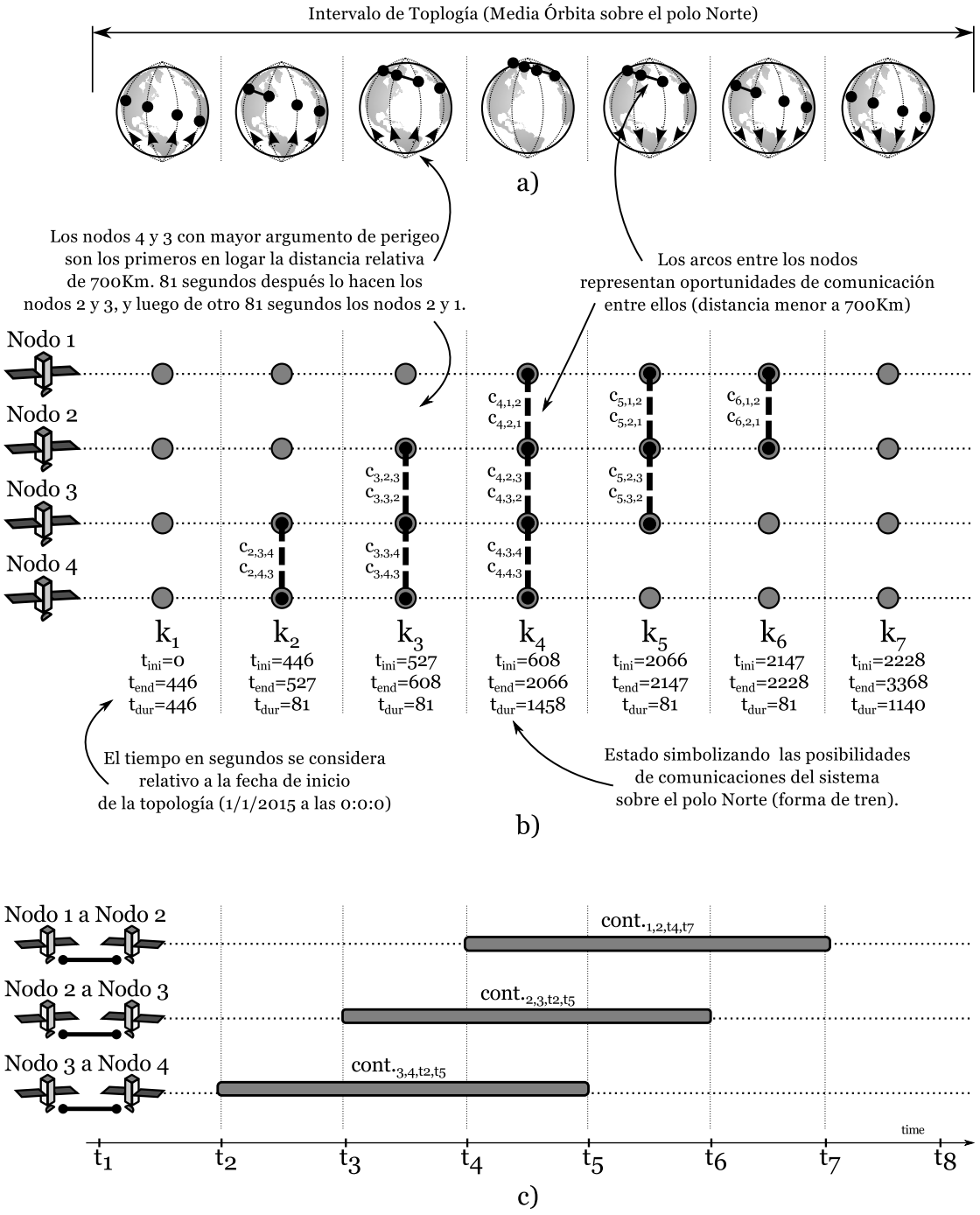


FIGURA 2.6: Representaciones del caso de referencia con trayectorias a), modelos de máquina de estado (FSM) b), y modelo de lista de contacto (CL) c)

si se desea acortar un contacto dado que nuevo valor de inicio o fin del mismo puede no coincidir con la discretización de los tiempos expresadas en los valores t_k o t_{k+1}. En consecuencia, suele ser necesario fraccionar un estado existente k_a en dos k_{a1} y k_{a2} de manera tal de que el tiempo de finalización de k_{a1} o t_{ka} coincida con el nuevo tiempo de inicio o fin del contacto modificado. De esta manera se pueden activar o desactivar los arcos $c_{k,i,j}$ necesarios logrando que el modelo FSM coincida con la realidad de plan de

contacto que se pretende diseñar. La Figura 2.7 ilustra este proceso de manera gráfica.

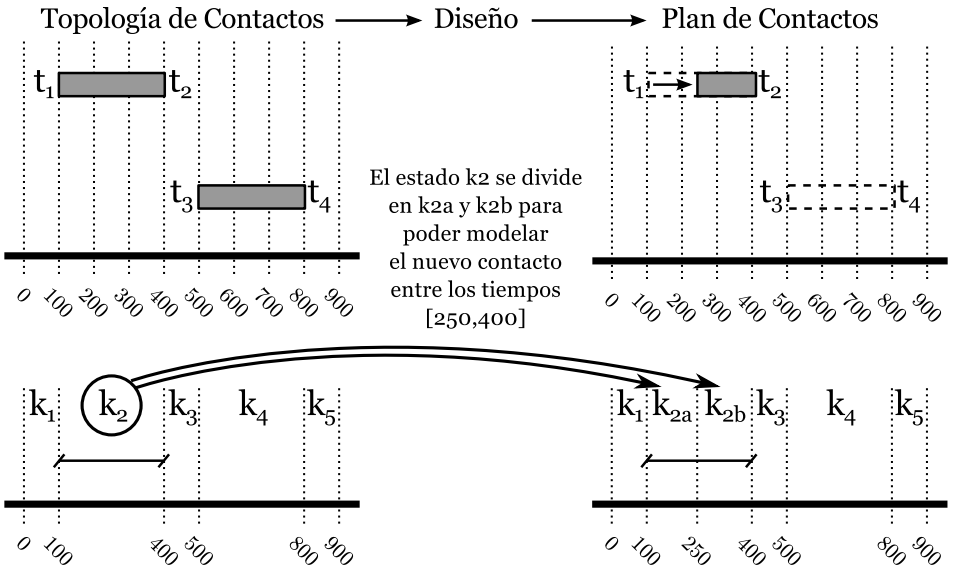


FIGURA 2.7: Fraccionamiento del estado k_2 en k_{2a} y k_{2b} para permitir el diseño de un nuevo contacto entre los 250 y 400 segundos

Por último, como trataremos a lo largo de esta tesis, el fraccionamiento de estados también resulta de utilidad para permitir un diseño de plan de contacto de mayor granularidad y precisión. En general, y debido a esta flexibilidad, nos basaremos en esta técnica de modelado para describir y formalizar las metodologías de diseño de plan de contacto en los capítulos 3, 4, y 5, así como para tratar cuestiones de implementación en el capítulo 6.

2.3.3. Modelado como Lista de Contactos

Sin embargo, el modelado FSM requiere de volúmenes de datos significativos para expresar las matrices T, I, y C, especialmente cuando se realizan fraccionamientos de estados. Alternativamente, existe una manera mas simple de expresar los planes de contactos o topologías de contacto por medio de una lista de contactos o contact list (CL) en Inglés.

La representación por medio de CL implica enumerar los contactos del sistema en la forma: nodo fuente, nodo destino, tiempo de inicio de contacto, tiempo de fin de contacto, y opcionalmente la capacidad de datos ($cont._{fuente,destino,inicio,final}$). De esta manera, la primera media órbita del caso de referencia consta de 3 contactos: N_1 a N_2 desde t_4 a t_7, N_2 a N_3 desde t_3 a t_6, y N_3 a N_4 desde t_2 to t_5. Por ende, el modelo CL para este escenario se ilustra en la Figura 2.6 c) y resulta mucho mas compacto y elemental que un sistema de matrices de adyacencias con el utilizado para FSM.

En general, dadas las características de síntesis de este esquema de modelado, el mismo es utilizado para completar las bases de datos de ION [67] y para distribución de información de contactos en el protocolo CPUP [96]. Sin embargo, dado el grado de compresión de CL resulta poco conveniente para el diseño de planes de contactos en comparación con FSM, la cual además resulta sumamente apropiada para el planteo de modelos de programación lineal como los realizados en el capítulo 3. De todas maneras, la traducción entre FSM y CL y viceversa es directa fácil, directa y transparente de implementar.

2.4. Restricciones de Recursos

Como se definió en la sección 2.2 de este capítulo, el conjunto de todas aquellas oportunidades de comunicación en un cierto intervalo de topología conforman la *topología de contacto*, sobre la cual se pueden implementar mecanismos de diseño que modifiquen la misma para satisfacer necesidades de recursos del sistema. En esta sección describimos con mayor detalle estos casos y los clasificamos para finalmente modelarlos y considerarlos para las metodologías de planificación y diseño de *plan de contactos* contenidas en el capítulo 3.

En general, las plataformas satelitales se diseñan con el objetivo de optimizar el tamaño, volumen, y consumo para minimizar los altos costos asociados al lanzamiento de los mismos. En consecuencia, la arquitectura de estos nodos suele estar restringida en cantidad de transponders (equipos de comunicaciones), disponibilidad de potencia (disponible por medio de paneles solares), cantidad y tamaño de antenas, combustible disponible para propulsión, entre otros que limitan la posibilidad de concretar contactos físicamente factibles. Estos fenómenos deben ser tenidos en cuenta para diseñar planes de contactos para una red satelital DTN de múltiples saltos como la planteada para la Arquitectura Segmentada. En efecto, y de acuerdo el mejor conocimiento del autor de este trabajo, estos efectos no han sido investigados anteriormente en la comunidad.

Con el fin de lograr diseñar un plan de contacto para redes satelitales, en esta sección enumeramos, clasificamos y describimos las características de estos recursos limitados y de las arquitecturas de la plataforma que puedan llegar a tener un impacto en la implementabilidad del plan de contacto final. En efecto, clasificamos estas restricciones en dos grupos: uno representando aquellas que hacen que un contacto particular en un tiempo dado se vuelva directamente inviable, y otro que limita la cantidad de contactos que un nodo puede mantener simultáneamente. Denominamos a las primeras restricciones de tiempo y zona o *time-zone constraints* (TZC) en Inglés y restricciones de recursos concurrentes o *concurrent resources constraints* (CRC) en Inglés.

2.4.1. Restricciones de Tiempo y Zona

En general las restricciones de tiempo y zona (TZC) prohíben la emisión de radiofrecuencia en un área geográfica determinada en un tiempo determinado ya sea para evitar interferencias no deseadas u otras razones políticas o diplomáticas específicas de las agencias espaciales. Dado que las constelaciones LEO como la planteada en el caso de referencia de la sección 2.3.1 orbitan prácticamente sobre toda la superficie terrestre, cumplir con todas las regulaciones internacionales puede resultar desafiante.

En particular, como se muestra en la Figura 2.8, los enlaces entre satélites de naturaleza tangencial a la superficie terrestre (en el caso de que ambos tengan la misma altitud), pueden *iluminar* otros satélites en la órbita geoestacionaria ubicada sobre el plano ecuatorial [99]. Algunos de estos satélites GEO deben ser especialmente considerados dado que dan soporte de comunicaciones a misiones espaciales tripuladas como la Estación Espacial Internacional o International Space Station (ISS) en Inglés por medio de la red DSN de NASA. En efecto, una política apropiada de irradiación deber ser aplicada con el fin de evitar generar (o recibir) interferencia mas allá de lo permitido por la normativa internacional especificada por la Organización Internacional de Telecomunicaciones o International Telecommunications Union (ITU) en Inglés [100].

Por otro lado, pueden existir otras razones específicas dentro de las agencias espaciales que impidan irradiar señales sobre alguna zona geográfica específica. Estas pueden ser clasificadas como restricciones de tiempo y zona que directamente obligan a cualquier contacto que ocurra en ese intervalo entre nodos que sobrevuelen la zona en cuestión a ser desactivado. Además, de acuerdo a las normativas de la ITU [100], las interferencias no sólo se limitan por la cantidad de potencia interferente que llega al satélite interferido, si no que el porcentaje de tiempo mensual que esta supera un determinado valor [99]. En consecuencia, en lugar de anular contactos directamente también es válida la opción de elegir posibles combinaciones de utilización de los mismos que no infrinja las leyes internacionales.

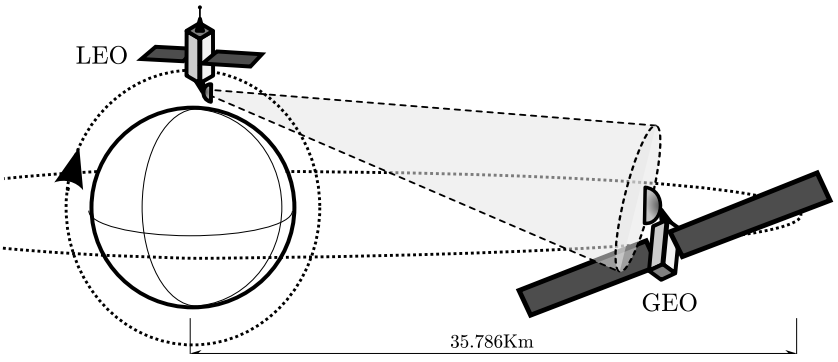


FIGURA 2.8: Interferencia generada por enlaces ISL sobre los polos a satélites GEO

En conclusión, las TZC pueden ser o bien directamente aplicadas a la topología de contactos, o bien ser analizadas en un espacio combinatorio de manera que los contactos interferentes no superen el porcentaje de tiempo permitido. En general, el estudio de estas últimas quedan fuera del alcance de esta tesis pero las estrategias aquí detalladas pueden sin duda servir de inspiración para el diseño de plan de contactos compatibles con las normativas internacionales de telecomunicaciones.

2.4.2. Restricciones de Recursos Concurrentes

En otro grupo, las restricciones de recursos concurrentes (CRC) resultan levemente mas complejas que las TZC dado que involucran la resolución de un problema combinatorio derivado de limitaciones arquitecturales o de potencia en el bus de la plataforma satelital. En general, la solución yace en explorar diferentes combinaciones de contactos dentro de la topología de contacto hasta encontrar alguna que satisfaga las restricciones. En caso de que haya mas de una se debe especificar un criterio a optimizar para elegir la mas eficiente de este espacio de soluciones.

Probablemente la CRC mas común es aquella que surge cuando el patrón de radiación de una antena de un satélite permite alcanzar dos o mas nodos destinos como se ilustra en la Figura 2.9 a). En este caso, los esquemas de múltiple acceso podrían automáticamente negociar y compartir el acceso al medio de comunicaciones, sin embargo, dado los problemas listados en la sección 2.2.1.1, las ineficiencias de estos esquemas no justifican su uso en estos contextos. Por el contrario, el correcto diseño del plan de contacto permite el uso de enlaces punto a punto mas eficientes que realmente utilicen todo el espectro disponible sin perder capacidad en negociaciones innecesarias y sobre todo evitables en un esquema de comunicaciones tan predecible como el caso de redes espaciales.

Por otro lado, cuando un satélite se diseña para poder utilizar enlaces ISLs desde diferentes direcciones como en el caso de referencia, suele ser necesario colocar mas de una antena sobre la estructura del satélite. Por ejemplo, considerando la Figura 2.9 b), un conmutador de potencia podría permitir utilizar diferentes elementos radiantes sin necesidad de multiplicar los equipos de comunicaciones. En consecuencia, un solo contacto puede ser utilizado en un momento dado debido a que el conmutador de potencia deja la segunda antena inutilizada. En otras palabras, si un contacto existe en la topología de contactos para cada una de las antenas, una decisión debe tomarse en la etapa de diseño del plan de contacto definitivo.

Una arquitectura mas compleja se ilustra en la Figura 2.9 c), donde dos contactos simultáneos pueden ser implementados por medio de dos sistemas cooperativos de comunicaciones. En este caso la limitación radica en dos contactos al mismo tiempo (uno

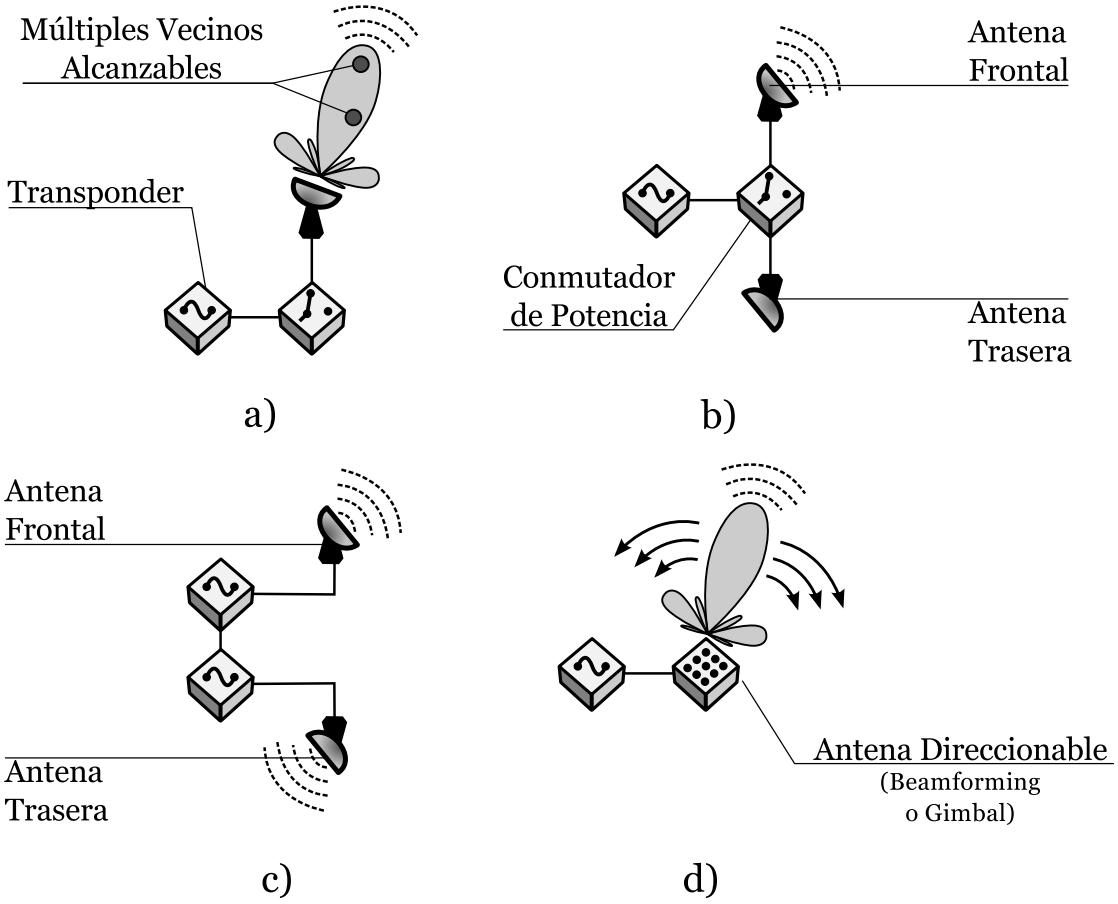


FIGURA 2.9: Arquitectura de plataformas satelitales con a) múltiples posibles vecinos, b) un conmutador de potencia, c) dos equipos de comunicaciones, y d) una antena direccionable electrónica o mecánicamente

por cada antena) siempre y cuando la potencia de la plataforma lo permita. En caso contrario, se mantendría la misma restricción de recursos de la figura 2.9 b).

Por último, al considerar antenas direccionables como se muestra en la Figura 2.9 d) se abre un interesante panorama que da lugar al diseño de plan de contactos para antenas haz dinámico. En general, existen antenas que se pueden apuntar mecánicamente (gimbal en Inglés) con el fin de apuntar la máxima ganancia del lóbulo de radiación hacia una zona de interés. Por otro lado, también existen antenas dirigibles electrónicamente (beam-forming en Inglés) que evitan realizar un movimiento mecánico y por ende el desgaste y limitación a la vida útil derivado del mismo. En general, este tipo de antenas implican necesariamente una selección de contacto en caso de que mas de uno sea factible. En efecto, una estrategia valida para la conformación de la topología de contacto de estos sistemas es considerar que la antena tiene la máxima ganancia en todas las direcciones dentro del rango de operación de la misma. Luego, el diseño del plan de contacto define una dirección específica de apuntamiento en función de la conveniencia de elección del contacto a nivel sistema.

En conclusión, y en general, las restricciones CRC implican un proceso directo de selección de contactos derivando en problemas combinatorios poco triviales para grandes constelaciones o largos intervalos de topología.

2.5. Modelado de Restricciones

Por un lado, las restricciones TZC resultan aplicables directamente sobre la topología de contacto, pero las CRC tienen un origen variado y pueden resultar complejas de tratar en conjunto a la hora de diseñar planes de contactos que respeten esta variedad de arquitecturas. En consecuencia, es necesario un modelado de las mismas que permita general planes de contactos implementables con los recursos existentes en el sistema.

En efecto, la estructura lógica ilustrada en la Figura 2.10 permite modelar la totalidad de las restricciones concurrentes al limitar en diferentes niveles la cantidad y tipo de recursos que se puedan considerar simultáneamente. Se puede expresar estos ya sea de manera *jerárquica* (en relación al recurso inmediatamente superior) o *absoluta* (en relación al segmento). La arquitectura de modelo que se propone parte del segmento (satélite) como origen, del cual se desprenden directamente los sub-sistemas de comunicaciones o transponders. Tradicionalmente existe uno de estos por satélite, pero en un segmento de la Arquitectura Segmentada podrían existir mas. A su vez, de los transponders se desprenden antenas asociadas a los mismos, las cuales pueden mantener contactos u oportunidades de comunicaciones.

Por ejemplo, la problemática planteada en la Figura 2.9 a) se puede limitar en el modelo estableciendo que la máxima cantidad de contactos por antena (o segmento en caso de que haya una sola antena) es 1. El caso b) se modelaría limitando la máxima cantidad de antena por transponder (o segmento en caso de que haya un solo transponder) a 1. También, el caso en la Figura 2.9 c) se podría limitar a un transponder máximo por segmento en caso de que exista restricciones de energía para su utilización. Finalmente, el caso d) requiere de una estrategia específica en la que se limite a un contacto por antena, pero que además el modelo de antena cuente con un patrón de radiación modificado que contemple la máxima ganancia que la misma puede tener en todo el rango de direcciones posible.

Para el caso de referencia planteado, en los polos se da la situación en la que un satélite tiene dos oportunidades de contacto con el vecino frontal y trasero. Con el fin de optimizar el diseño de la arquitectura para obtener satélites de tamaño suficientemente pequeños para las primeras pruebas de la Arquitectura Segmentada de CONAE, planteamos contar con un sólo transponder con un conmutador de potencia que alimente una

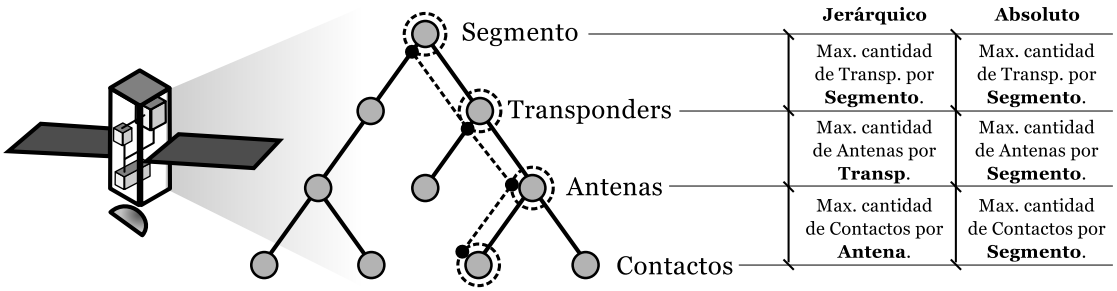


FIGURA 2.10: Modelado de Restricciones de Recursos y Arquitectura

antena frontal y otra trasera (similar a la arquitectura ilustrada en la Figura 2.9 b)). En consecuencia el modelo de restricciones a utilizar es aquel que limite el uso máximo de una antena por transponder o segmento y además limite a un contacto máximo por antena. En efecto, esta situación es idéntica a modelar el caso de un contacto máximo por segmento que también modelaría el escenario donde cada segmento cuente con una única antena omnidireccional. En otras palabras, limitar el modelo de restricciones a un contacto por segmento permitiría diseñar planes de contacto para segmentos de dos antenas y un transponder, o una antena omnidireccional indistintamente.

Cabe destacar que en este esquema estructurado jerárquicamente no resulta posible modelar un sistema de matrices de antenas donde un conjunto de transponders puedan acceder de manera compartida a un grupo de antenas. En general esta arquitectura necesitaría de un modelo diferente, pero en este trabajo es dejada de lado dada la complejidad que podría llegar a tener la implementación real de una matriz cruzada de potencia de este tipo para aplicaciones espaciales.

2.6. Diseño de Plan de Contactos

En general, la limitación de recursos planteada en la sección 2.4 podría limitar la cantidad o simultaneidad de contactos para uno o mas nodos del sistema, requiriendo de una criteriosa selección entre las diferentes posibilidades existentes. En otras palabras, un nodo puede tener potenciales oportunidades con mas de un vecino en un tiempo dado pero solo hacer uso de un de estas oportunidades debido a la restricción de recursos. Por otro lado, otros inconvenientes como interferencia a otros satélites o zonas geográficas pueden surgir y ser solucionados con un correcto planteo de plan de contactos.

2.6.1. Definición de Diseño de Plan de Contactos

Habiendo definido plan de contactos en la sección 2.2.3 y topología de contactos en la sección 2.2.2, llamaremos *diseño de plan de contactos* o *contact plan design* (CPD) en

Inglés al proceso cuya salida o entregable es un sub-conjunto de la topología de contactos que satisface el conjunto de restricciones descritas en la sección 2.4. Las elecciones tomadas en dicho proceso de diseño resultan determinante en el rendimiento futuro del sistema una vez implementadas en los nodos DTN [2]. En consecuencia, el criterio con el cual se concreta esta elección puede considerar diferentes fuentes de información para optimizar el uso de la red de satélites como topología existente [2], esquemas de ruteo [4], o tráfico planificado pudiendo entregar planes de contactos lo mas útiles posibles para la constelación.

Tradicionalmente, el problema de CPD ha recibido escasa atención de la comunidad dado que en general, como mostramos en el estado del arte en el capítulo 1, se ha asumido que todos los contactos en la topología de contactos pertenecen al plan de contacto final. En otras palabras, se daba por hecho que los nodos siempre contaban con los recursos necesarios para implementar todas las posibilidades físicas de contactos. Sin embargo, resolver el problema de CPD resulta poco trivial para redes como las planteadas en el caso de estudio de referencia como ilustramos a continuación (inclusive para intervalos de tiempos de unas pocas horas) requiriendo de mecanismos automáticos de diseño como los planteados en el capítulo 3.

2.6.2. Posibles Planes de Contactos del Caso de Referencia

Para ilustrar el concepto de diseño de plan de contacto, retomamos el caso de referencia para mostrar dos posibles planes de contactos que satisfacen las restricciones de arquitecturas como la mostrada en la Figura 2.9 que limita al sistema a contar con un contacto máximo por segmento. Dado que cada nodo cuenta con dos antenas pero un único recurso de comunicación (transponder), se debe tomar una decisión para los nodos N_2 y N_3 en los estados k_3, k_4, y k_5 en el modelo FSM del sistema.

En efecto, si se mantiene la discretización de estados como está originalmente planteada (es decir, no se realiza ningún fraccionamiento de estado), se pueden considerar dos posibles planes de contactos ilustrados en la Figura 2.11 a) y b). Si se elije el primer plan de contacto, la red proveerá el máximo *throughput* (capacidad de transmisión total acumulada) dado que se activan la máxima cantidad de contactos posibles en el intervalo. Sin embargo, dicha elección implica obtener una red discontinua al evitar que los grupos de nodos N_1 y N_2 se puedan comunicar con N_3 y N_4. En este sentido, el plan de contacto mostrado en la Figura 2.11 b) favorece la distribución mas equitativa y justa de los enlaces dando lugar a una red mas comunicada. En otras palabras, ambos planes de contactos son factibles y satisfacen las restricciones de recursos del sistema, sin embargo

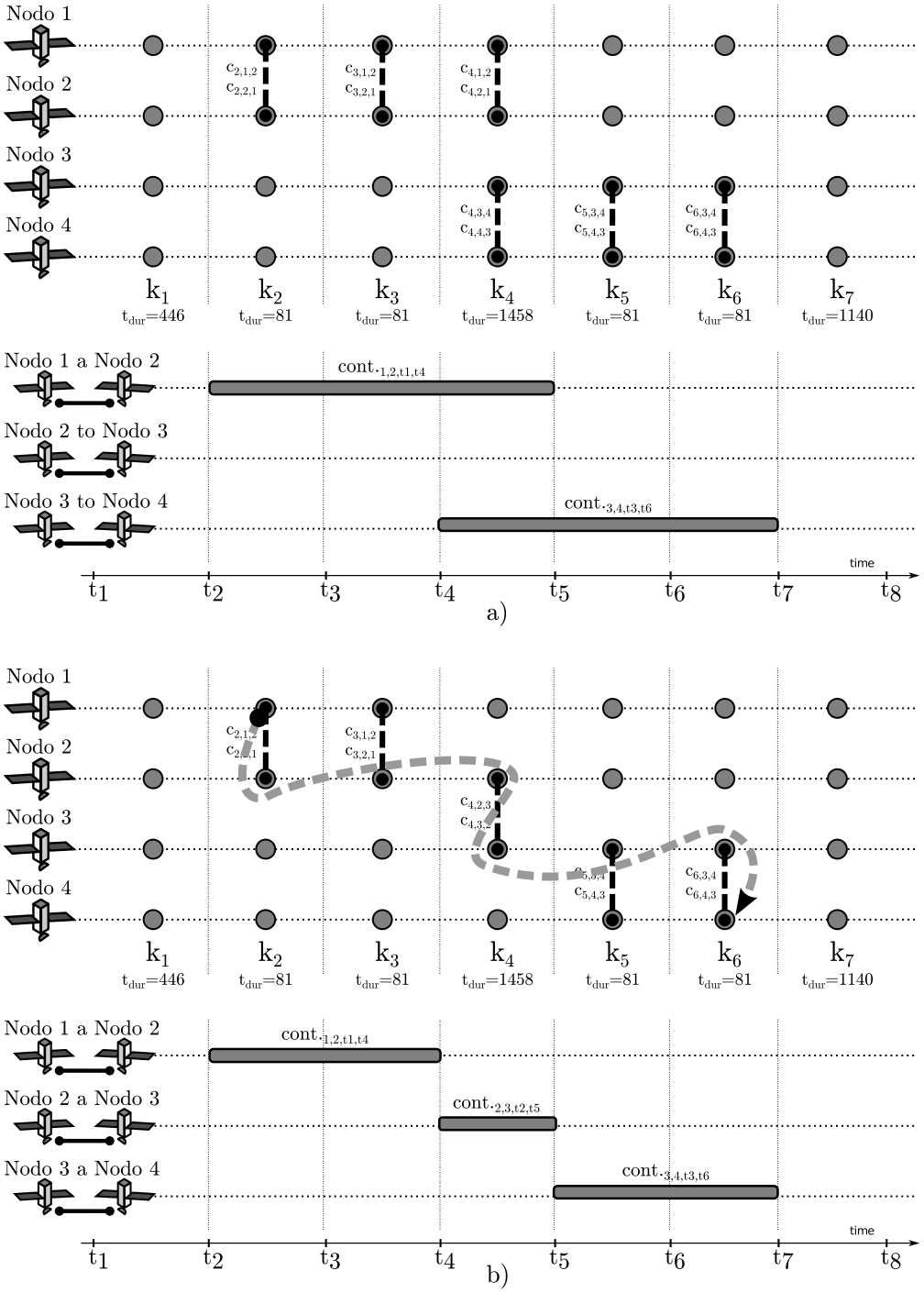


FIGURA 2.11: Dos posibles planes de contactos para el caso de referencia que priorizan a) máximo throughput y b) justicia de asignación de enlaces

estos favorecen criterios de selección diferentes: máxima capacidad de sistema o justicia en asignación de los enlaces.

2.6.3. Desafíos, Problemas y Compromisos de Diseño de Plan de Contactos

Una vez definido el proceso de diseño de planes de contactos, pasamos a discutir brevemente los desafíos que el mismo implica, que problemas requiere resolver, y que compromisos se ponen en juego.

2.6.3.1. Complejidad del Proceso

En general, y a pesar de que el ejemplo resulte fácil de incorporar al mostrar dos posibles soluciones para unos pocos estados (una sola pasada por el polo norte), mientras mas largo se hace el intervalo de topología, mas estados se incorporen al modelo, se consideren mas nodos o antenas, o se habiliten mas contactos en la topología de contactos, el proceso de selección se hace mas y mas complejo. En efecto, el problema de CPD termina derivando en una combinatoria poco trivial cuya complejidad incrementa exponencialmente y que el operador del sistema orbital debe resolver antes de obtener el plan de contacto definitivo que mantenga la red en funcionamiento.

2.6.3.2. Criterios e Información de entrada

Por otro lado, el diseño de plan de contacto puede estar regido por otros criterios mas complejos como los que revisaremos en los capítulos 3, 4 y 5, los que pueden considerar no solo criterios de salto único como los ilustrados en la Figura 2.11 a) y b), si no que caminos de múltiple saltos que evalúen métricas de rutas entre los nodos para finalmente decidir que plan de contacto se adapta mejor y permite obtener el mejor desempeño del sistema. Un ejemplo de estas rutas se ilustra del N_1 al N_4 por medio de una flecha discontinua en la Figura 2.11 b), para la cual el segundo plan de contacto puede resultar apropiada. Sin embargo, considerar las rutas implica que se debe conocer y predecir las decisiones del esquema de ruteo que utilizarán los nodos (por ejemplo CGR [60] o MFW [61]), lo cual no siempre es posible o las mismas no resultan del todo eficientes.

Además, y en las aplicaciones espaciales general, el tráfico que genera cada satélite es o bien del tipo telemetría (información de estado de la plataforma del objeto orbitante) o dato de ciencia o carga útil (imágenes de la tierra, imágenes de radar, de otros sensores, etc.). Casualmente ambos resultan también de naturaleza predecible ya que la telemetría se suele generar en manera periódica según configuración del satélite, y la adquisición de datos de ciencia es generalmente comandado por un operador en tierra quien establece el tiempo específico en el cual se deben usar los instrumento de a bordo. En consecuencia, el MOC suele contar esta valiosa información que puede ser de utilidad para el diseño de

un eficiente plan de contacto. Sin embargo, tener en cuenta el tráfico (como haremos en el capítulo 5) implica que las deciciones de rutas tomadas por los nodos coincidan con la asumida en el diseño de plan de contacto. Sin embargo, esto no es siempre posible dado que hay efectos de congestión indeseados que no permiten se respete un enrutamiento ideal.

2.6.3.3. Sobre la Implementación del Plan de Contacto

De esta manera, a medida que incorporamos mayor información sobre el sistema de red satelital, el diseño de plan de contacto puede explotarlas para mejorar la eficiencia de diseño, aunque incrementando la complejidad computacional. Sin embargo, cuando ya se utiliza el tráfico como información de entrada, el CPD puede asumir y asignar tráfico a contactos particulares, que puede no ser respetada por los nodos al momento de enrutar los paquetes. Esta discrepancia puede darse por que los nodos comparten el mismo plan de contacto derivando en la inhabilidad de los segmentos de predecir un fenómeno llamado congestión que abordaremos en el capítulo 6. En consecuencia, una posible estrategia es diseñar CPs específicos para cada nodo de manera que los recursos de contactos de la topología esté dividida de antemano para cada nodo como propondremos en el mecanismo MG-CGR.

En general, dado que pueden existir diferentes criterios, estrategias, y consideraciones de implementación para un diseño de CP cuya complejidad incremente drásticamente, se vuelve menester encontrar esquemas y procedimientos de diseños automatizados que permitan ayudar a los operadores de la red a administrar estos sistemas de manera automática. Este es el objetivo último de esta tesis para el cual exploraremos diferentes criterios de diseño que hacen uso de diferentes fuentes de información para proveer diseños de planes de contactos acordes y eficientes para redes de la Arquitectura Segmentada.

2.6.4. Metodologías Existentes

Respecto a las metodologías existentes, en el 2010, Huang et. al. en [89] analiza el diseño de topología para redes DTN predecibles bajo el procedimiento cost efficient topology design (CETD) cuyo objetivo es mantener la conectividad entre pares de nodos en el sistema, que el costo de esta conexión se mantenga dentro de un cierto umbral, y minimizar el costo total del diseño. Mas adelante, en el 2012 los mismos autores en [90] extienden el trabajo al considerar probabilidades de falla de los enlaces, donde el nuevo proceso de diseño buscaba que para cualquier par de nodos el camino que los una tenga una confiabilidad superior a un umbral establecido.

Sin embargo, ninguno de estos estudios incorporan consideración de uso de recursos como los planteados en este capítulo y asumen que todos los contactos pueden ser utilizados sin limitación. En general, y como se mostró en la sección 2.4, esto no es posible en el caso de redes satelitales y requiriendo de consideraciones especiales tratados únicamente y por primera vez en este trabajo.

2.6.5. Metodologías Propuestas

En consecuencia, en esta tesis proponemos un conjunto de mecanismos de diseño de plan de contacto que incorporan diferentes informaciones de entrada en base a los cuales se elabora la hipótesis que sus rendimientos sean proporcionales a los mismos. De manera análoga a lo propuesto en la sección 1.1.2, en la Figura 2.12 se ilustra esta expectativa (hipótesis) para los mecanismos basados solamente en la topología de contactos (Fair Contact Plan o FCP), basados en las rutas derivadas entre nodos (Route-Aware Contact Plan o RACP) y basados en el tráfico del sistema (Traffic-Aware Contact Plan o TACP) descritos en los capítulos 3, 4,y 5 de esta tesis doctoral.

2.6.6. Herramientas Desarrolladas

Con el fin de enfrentar la problemática del diseño de plan de contacto planteada, se han desarrollado un conjunto de herramientas (software) que utilizaremos a lo largo del desarrollo de la tesis como listamos a continuación.

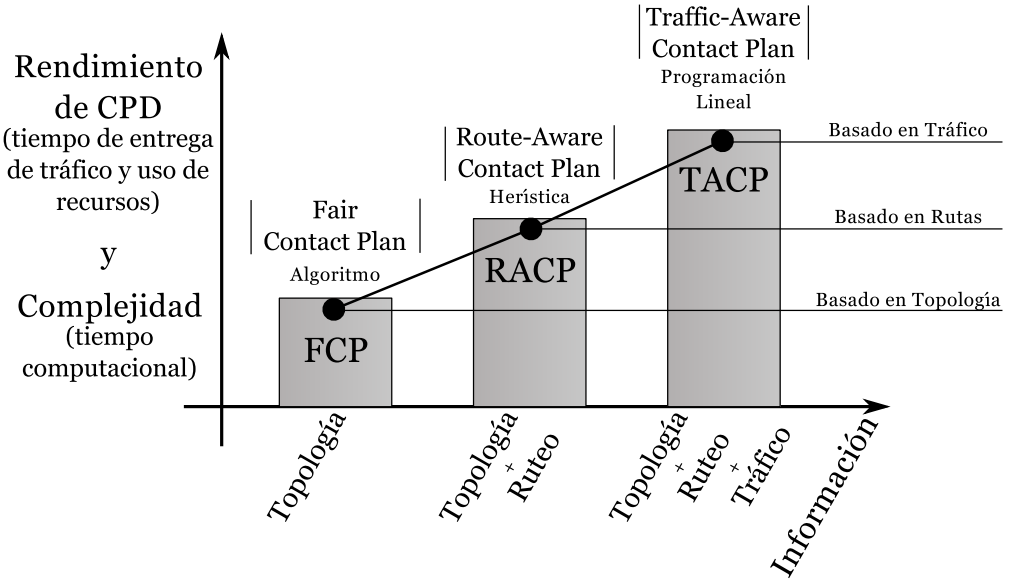


FIGURA 2.12: Rendimiento esperado de los mecanismos FCP, RACP y TACP

1. **GLOrbit:** Esta herramienta es el punto de entrada para el diseño de plan de contacto dado que es la encargada de tomar parámetros orbitales y rangos de comunicaciones para a la salida entregar la topología de contacto DTN (expresada bajo el paradigma de modelado FSM). GLOrbit cuenta con una interface gráfica basada en OpenGL para verificar la topología evaluada (Figura 2.5) e implementa el algoritmo SGP4 [88] implementado de manera paralela con la técnica OpenCL lo que a su vez derivó en una de las publicaciones de esta tesis [1].

2. **TopologySolver:** Una vez obtenida la topología de contacto con GLOrbit, la misma debe ser procesada por las metodologías FCP [2], RACP [4], y TACP [5] descritas a lo largo de los capítulos 3, 4, y 5 respectivamente. La aplicación TopologySolver centraliza estas funcionalidades y procedimientos en una librería en C++, para finalmente generar archivos de salida en forma gráfica, con el mismo modelado FSM, o directamente formato de plan de contacto como el utilizado por la aplicación ION [67].

3. **TotSim:** Con el fin de evaluar estos planes de contactos bajo la presencia de un tráfico realista, se generó un simulador basado en Omnet++ que incluye un modelado específico del protocolo bundle y los procedimientos de enrutamientos MFW [61], CGR [59], PCC [97], C-CGR [9], PA-CGR, y MG-CGR [10] tratados en profundidad en el capítulo 6.

Capítulo 3

Diseño de Plan de Contactos basado en Topología

3.1. Introducción

En este capítulo presentamos un mecanismo de diseño basado en información existente de topología (topología de contactos) para el diseño de plan de contacto llamado Fair Contact Plan o FCP dado que su objetivo final es una asignación justa de las posibilidades de comunicación entre los nodos de la red. Este mecanismo consta de un modelo formal del problema y una alternativa algorítmica siendo el primer aporte del trabajo doctoral expuesto en el congreso internacional IEEE Wireless for Space and Extreme Environments (WiSEE) [3] (Baltimore, USA) en el 2013 y publicado en la revista IEEE Sensors Journal [2] en el 2014. Cabe destacar que del planteo formal del problema también derivamos Max Capacity Contact Plan (MCP) un segundo criterio de diseño basado en topología que busca optimizar la máxima cantidad de contactos en todo el sistema.

3.1.1. Suposiciones del Esquema

En un contexto de redes DTN predecibles, el esquema basado en topología asume que la única fuente de información disponible para diseñar el plan de contacto es la topología física, es decir, el conjunto de contactos físicamente posibles expresados en la topología de contactos definida en la sección 2.2.2. En otras palabras, se ignora que esquema de enrutamiento se usa en los nodos y que tráfico se cursará en el sistema. En consecuencia el criterio de diseño se basa en una asignación justa de las comunicaciones de manera que la red puede permitir una conectividad del tipo todos contra todos. En general, este

esquema es propicio para el paso de telemetría el cual suele ser de bajo volumen de datos pero periódico en el tiempo.

3.2. Planteo Formal del Problema

Con el fin de poder tratar el problema de diseño aquí planteado, se propone un planteo formal del problema expresado en un modelo de programación lineal mixta de enteros (MILP) cuyo objetivo es maximizar la justicia en el plan de contactos diseñado.

3.2.1. Definición de Justicia

En esta sección definiremos un criterio de justicia al cual nos ajustaremos para el diseño. Entre las diferentes definiciones de justicia, probablemente la mas popular es la basada en el criterio de justicia *min-max*, el cual considera que la equivalencia se logra cuando ya no se puede incrementar una asignación determinada sin decrementar otra capacidad por un valor igual o menor [101].

Con el fin de lograr satisfacer el criterio de justicia *min-max*, el mecanismo sugerido en [101] propone en una primera etapa maximizar la mínima asignación entre todos los agentes de distribución, para finalmente minimizar el máximo sin perder la capacidad total obtenida en la primera etapa. En otras palabras, y enfocados en nuestro caso de estudio, la metodología supone una asignación de contactos igualitaria entre todos los pares de nodos, para luego distribuir lo mas justo posible el remanente. Este esquema se ilustra en la Figura 3.1 y resulta análogo a elevar un *piso* de mínimo (t_{min}) para luego bajar un *techo* máximo (t_{max}) sin perder la capacidad máxima lograda.

3.2.2. Modelo MILP Etapa 1

Una vez definido el criterio de justicia, debemos describir las variables a utilizar. En este modelo, llamaremos i_i a la máxima cantidad de contactos que un nodo i puede implementar en un momento dado. De esta manera, la matriz $[I] = i_i$ codifica las restricciones de interfaces del sistema (de acuerdo a lo modelado en la sección 2.5). La cantidad de nodos se establece en N por lo que en general $0 \leq i \leq N$. Por otro lado, la topología de contacto se representa por medio de una matriz $[P] = p_{k,i,j}$ donde $p_{k,i,j}$ representa la existencia de un arco o posibilidad de contacto entre los nodos i y j en el estado k donde $0 \leq i \leq K$ siendo K el número de estados presentes en la topología de contactos. En este modelo $p_{k,i,j}$ adopta un valor de 1 cuando el contacto se puede establecer y 0 cuando no hay contacto posible. Finalmente, el modelo entrega a

la salida una matriz $[L] = l_{k,i,j}$ donde los arcos $l_{k,i,j}$ adoptan el mismo valor que el $p_{k,i,j}$ correspondiente cuando el arco se decide activado o 0 cuando se desactiva.

En consecuencia, basaremos el modelo formal de descripción del diseño de plan de contacto basado en topologías en el criterio de justicia *min-max*. En concreto, para la etapa 1 proponemos un primer modelo MILP que optimice la asignación de capacidades de contactos con el fin de distribuir un mínimo tiempo de contacto entre todos los pares de nodos existentes en el sistema. Una vez logrado esto el modelo deberá distribuir la capacidad residual entre los pares restantes intentando maximizar la capacidad total del sistema. La Tabla 3.1 resume las variables a utilizar en el modelo planteado en las ecuaciones (3.5) a (3.4) con función objetivo (3.1).

En cuanto a las restricciones del modelo MILP, la ecuación (3.2) cumple la función de mantener la bi-direccionalidad de la elección de contactos. Esto lo logra con una expresión que o bien hace que ambos arcos (ida y vuelta) sean activados o desactivados en conjunto con el fin de mantener un esquema de enlace bi-direccional o full-duplex como el planteado. Por otro lado, la ecuación (3.3) es la que evita que el número de contactos simultáneos elegidos supere la cantidad de interfaces (o antenas) disponibles en el nodo i en un estado k determinado. Luego, la restricción (3.4) caracteriza las variables $l_{k,i,j}$ a un rango binario para que con un valor de 1 representen un contacto que deba estar presente

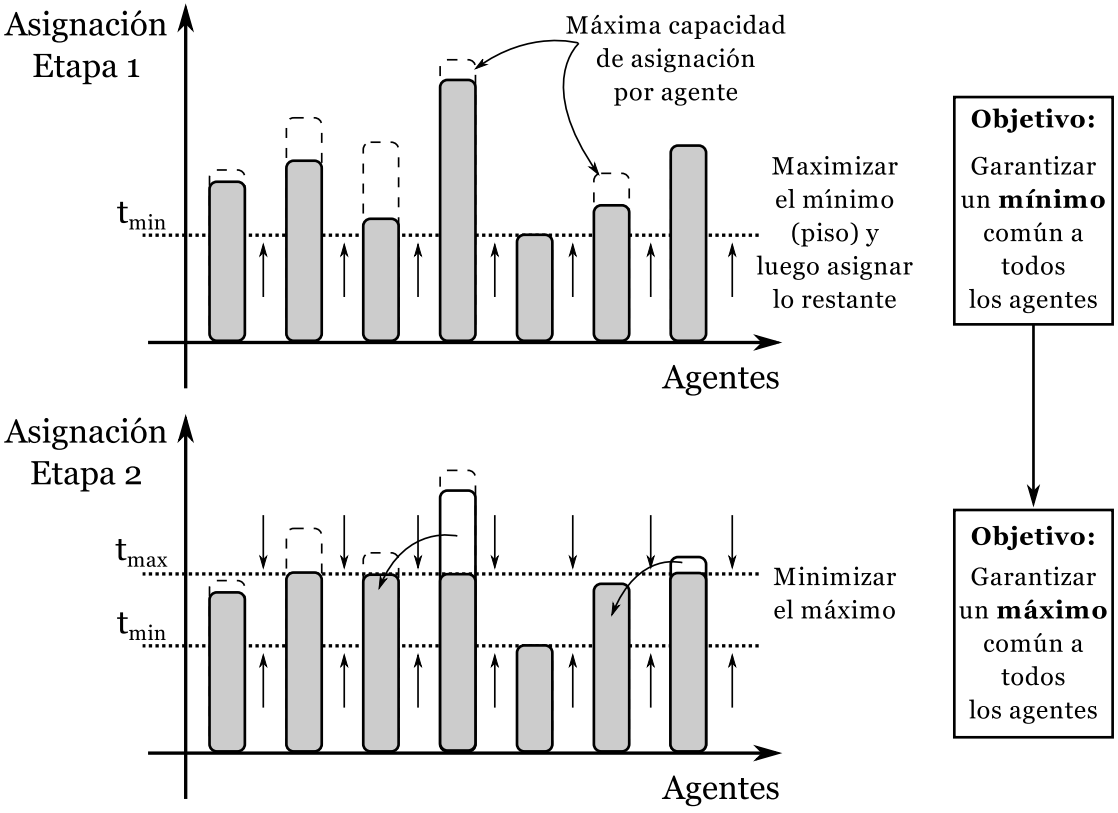


FIGURA 3.1: Distribución justa de acuerdo al criterio *min-max*

Tabla 3.1: Coeficientes y variables del modelo MILP de diseño de plan de contactos basado en topología

Coeficiente o Variable	Descripción
N	Número de nodos
K	Número de estados
$[P]_{k,i,j}$	Conjunto de $p_{k,i,j}$: topología de contactos
$[I]_i$	Conjunto de i_i: Máxima cantidad de contactos simultáneos
$[T]_k$	Conjunto de t_k: Duración de cada estado k
$[L]_{k,i,j}$	Conjunto de $l_{k,i,j}$: plan de contacto
t_{min}	Mínimo tiempo asignado a un par de nodos $\forall i, j$
t_{max}	Máximo tiempo asignado a un par de nodos $\forall i, j$
t_{obj}	Capacidad de sistema obtenida en la 1era etapa
ϵ	Ponderación de la capacidad de sistema en la 1era etapa $(0 \leq \epsilon)$
β	Porción de t_{obj} que se debe obtener en la 2da etapa $(0 \leq \beta \leq 1)$

en el plan de contacto o con 0 para indicar su ausencia. Finalmente, la ecuación (3.5) es entonces la que limita la variable auxiliar t_{min} que representa la mínima asignación (capacidad) entre todos los contactos para la totalidad del intervalo de topología, es decir, todos los estados k.

En consecuencia, como se puede observar en la función objetivo (3.1), maximizar t_{min} implica habilitar todos los contactos necesarios para que todos los pares de nodos i-j (agentes) tengan al menos una capacidad t_{min} asignada. Esto permite lograr el objetivo de la primera etapa de igualdad de asignación de capacidad. Por otro lado, a medida que la justicia mejora con esta aproximación inicial, un segundo componente a maximizar en la función objetivo (3.1) fuerza al modelo a también optimizar la capacidad total del

$$\text{maximizar:} \quad t_{min} + \epsilon \left(\sum_{k=1}^{K} \sum_{i=1}^{N} \sum_{j=1}^{N} l_{k,i,j} * t_k \right) = t_{min} + \epsilon * t_{obj} \tag{3.1}$$

sujeto a:

$$l_{k,i,j} = l_{k,j,i} \quad \forall k, i, j \tag{3.2}$$

$$\sum_{j=1}^{N} l_{k,i,j} \leq i_i \quad \forall k, i \tag{3.3}$$

$$l_{k,i,j} \in \{0, 1\} \quad \forall k, i, j \tag{3.4}$$

$$\sum_{k=1}^{K} l_{k,i,j} * t_k \geq t_{min} \quad \forall i, j \tag{3.5}$$

sistema por medio de la suma de todos las variables $l_{k,i,j}$ multiplicadas por la duración en tiempo de ese contacto t_k. Dado que en la función objetivo existen dos términos, un multiplicador ϵ permite ponderar el peso de estos criterios para ajustar el balance capacidad-justicia de la solución final.

3.2.2.1. Plan de Contacto de Máxima Capacidad

En efecto, cuando $\epsilon \gg t_{min}$, esta primera etapa del modelo MILP entrega un plan de contacto de máxima capacidad despreciando cualquier consideración de conservación de justicia. Este modelo derivado es válido como referencia de un esquema de diseño de plan de contacto de máxima capacidad o throughput (como el mostrado en la Figura 2.11 a)) que llamaremos Plan de Contacto de Máxima Capacidad o Max Capacity Contact Plan (MCP) en Inglés. En general, utilizaremos el modelo MCP a lo largo del trabajo, aunque cabe destacar que en general, este tipo de criterios puede derivar en redes discontinuas por lo que deben ser utilizados cuidadosamente o en combinación con otros esquemas en aplicaciones reales.

3.2.3. Modelo MILP Etapa 2

Si se considera un parámetro ϵ lo suficientemente grande en la primera etapa de la formulación MILP, se puede cumplir el objetivo de maximizar t_{min} para luego, con menor prioridad, obtener la máxima capacidad obtenible en el sistema. A pesar de que

$$\text{minimizar:} \quad t_{max} \tag{3.6}$$

sujeto a:

$$\sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N} l_{k,i,j} * t_k \geq t_{obj} * \beta \tag{3.7}$$

$$\sum_{k=1}^{K} l_{k,i,j} * t_k \leq t_{max} \quad \forall i,j \tag{3.8}$$

$$\sum_{k=1}^{K} l_{k,i,j} * t_k \geq t_{min} \quad \forall i,j \tag{3.9}$$

$$l_{k,i,j} = l_{k,j,i} \quad \forall k,i,j \tag{3.10}$$

$$\sum_{j=1}^{N} l_{k,i,j} \leq i_i \quad \forall k,i \tag{3.11}$$

$$l_{k,i,j} \in \{0,1\} \quad \forall k,i,j \tag{3.12}$$

este paso permite obtener cierta justicia del tipo *min-max*, la misma puede ser mejorada al re-distribuir el excedente de la capacidad total (t_{obj}) con el fin de minimizar el valor mas grande de capacidad entre los pares de contactos en juego (ahora medido con la variable t_{max}). En consecuencia se plantea una segunda iteración del modelo MILP detallado en las ecuaciones de restricción (3.7) a (3.12) con función objetivo (3.6).

Las restricciones de esta segunda etapa ((3.10), (3.11), y (3.12)) se mantienen con el mismo propósito que su instancia idéntica en la primera parte ((3.2), (3.3), y (3.4)). Por otro lado, a pesar de la semejanza entre las ecuaciones (3.9) y (3.5), en (3.5), t_{min} era una variable a resolver y determinar, mientras que en (3.9) es un coeficiente constante con el valor del t_{min} calculado en la primera etapa del modelo MILP. Además, la nueva ecuación (3.8) se incorpora con el objetivo de restringir la máxima capacidad (t_{max}) asignada entre todos los pares de nodos i y j en el resultante plan de contacto codificado en $[L] = l_{k,i,j}$. Es decir, ningún par de nodos cuenta con un tiempo de contacto superior a t_{max} al considerar la suma de todos los estados k.

Mas importantemente, la restricción (3.7) fuerza que esta nueva selección de contactos entregue una capacidad igual o mejor que $t_{obj} * \beta$ para $0 \leq \beta \leq 1$. Esto implica que la máxima capacidad final del sistema en el plan de contacto diseñado debe mantenerse en este valor producto de la máxima capacidad obtenida en la primer etapa (t_{obj}) y un coeficiente β configurado por el usuario. En otras palabras, β es un parámetro de configuración que permite determinar que fracción de la capacidad del sistema se puede relegar con el fin de mejorar la asignación equitativa de tiempo de contacto (minimizando t_{max} y mejorando el criterio *min-max*). En consecuencia, un valor de β cercano a 1 ayuda a mejorar la métrica de justicia min-max, sin penalizar la capacidad del sistema; por otro lado, un valor cercano a 0 permitirá que t_{max} sea minimizado lo máximo posible obteniendo mejores parámetros de justicia pero a costa de una perdida importante de capacidad total. En efecto, dado que nuestro objetivo es lograr distribuir equitativamente la máxima capacidad posible, asumiremos un $\beta = 1$, y llamaremos a este modelo de dos etapas FCP.

3.2.4. Sobre la Complejidad del Modelo Formal

El modelo de diseño de CPD aquí planteado cuenta con múltiples restricciones lineales con una cantidad significativas de variables binarias. En general, estos problemas combinatorios son conocidos por resultar del tipo NP-Complejos (tiempo polinomial no determinista o *nondeterministic polynomial time* en Inglés) dado que su tiempo de resolución aumenta exponencialmente (y no polinomialmente) respecto a la cantidad de

variables de este tipo en juego. Esto implica que a medida que la red satelital considerada aumenta en su intervalo de topología (mayor cantidad de estados k), incrementa la cantidad de nodos (vértices), o las posiblidades de comunicaciones (arcos), el problema se vuelve mas y mas intratable en tiempos razonables. En consecuencia, resulta apropiado explorar posibilidades algorítmicas que provean soluciones de calidad similar a costos computacionales razonables. En efecto, en la siguiente sección 3.3 proponemos un algoritmo para FCP (FCP-A) que cumple este fin.

3.3. Planteo Algorítmico

Si las restricciones de recursos discutidas en la sección 2.4 limitan los satélites a poder implementar un sólo contacto a la vez ($i_i = 1 \quad \forall i$, como en la arquitectura adoptada en el caso de referencia), el problema de diseño de plan de contacto puede verse como un problema de asignación o matching en Inglés que debe ser resuelto estado a estado. En este contexto, un contacto debe conectar dos nodos DTN de manera tal que ningún nodo sea el extremo de mas de un contacto a la vez. En consecuencia, el planteo resulta similar al ya conocido problema de asignación, cuya definición es: dado un grafo $G(V, E)$, una asignación M en G es un conjunto de arcos no adyacentes dispuestos de tal manera que ningún arco comparte un vértice con otro. En general, las soluciones existentes a este tipo de problema podrían utilizarse con el fin de generar (asignar) pares de nodos en un estado determinado de manera eficiente.

3.3.1. Algoritmos de Asignación no Bipartito

En general, el problema de asignación bipartito es ampliamente conocido y eficientemente resuelto con el algoritmo Húngaro [102]. Sin embargo, dado que para cada estado existe

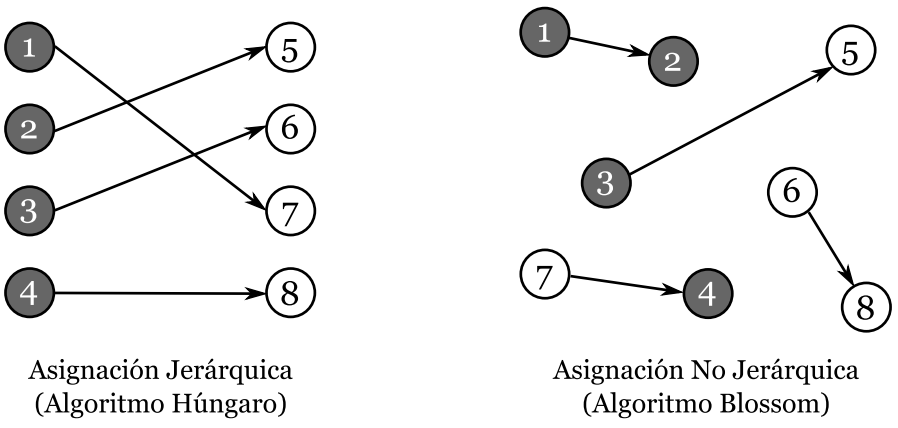


FIGURA 3.2: Asignaciones de cardinalidad 1 y algoritmos existentes que las resuelven

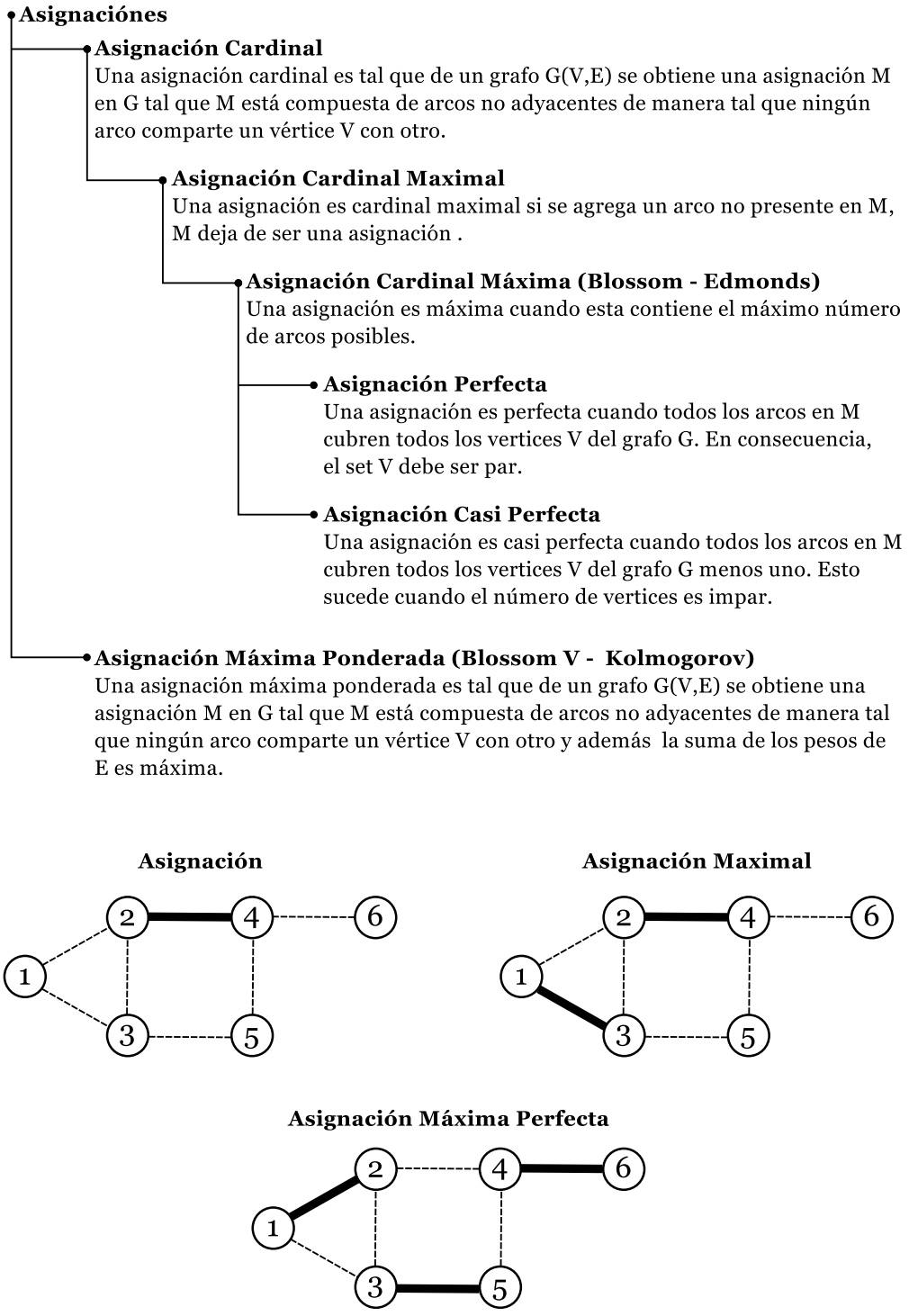


FIGURA 3.3: Clasificación y tipos de asignaciones no bipartitas

un grafo general que describe las posibilidades de comunicación de todos contra todos, no existe una jerarquía bipartita previa entre los nodos (es decir, cualquiera se podría juntar con cualquiera como se ilustra en la Figura 3.2) problema para el cual se debe considerar el caso general de asignación comúnmente conocido como problemas de *1-Matching* y *Edge Covering* en redes no dirigidas. La Figura 3.3 lista la clasificación de

las asignaciones en la literatura e ilustra algunos ejemplos, de los cuales la asignación máxima ponderada resulta de particular interés para diseñar el algoritmo FCP como describimos a continuación.

3.3.1.1. Algoritmo Blossom

Afortunadamente para nuestra búsqueda, existe un algoritmo denominado *Blossom* [103, 104] que resuelve este tipo de planteos en tiempo polinómico. El algoritmo Blossom fue descubierto por Edmonds en 1961 [103] y cumple con el objetivo de encontrar la *asignación cardinal máxima*. El algoritmo propuesto por Edmonds permite obtener una asignación por medio de mejoras iterativas de una asignación principal por medio de caminos aumentados [104]. En cada iteración el esquema o bien encuentra un camino aumentado, encuentra o un *blossom* (flor en Inglés o ciclos de arcos) para luego aplicar una recursión sobre este grafo contraído, o concluye que no existen caminos aumentados. En general, en el caso de que no existan estos ciclos, el algoritmo se reduce a la asignación bipartita del algoritmo Húngaro.

Por otro lado, en 2009, un aporte de Kolmogorov en [105] permitió contar con una implementación combinatoria de Blossom llamada Blossom V la cual resuelve el problema de *asignación ponderada perfecta de mínimo costo*. En efecto, la Figura 3.4 ilustra el resultado ponderado de Blossom V el cual no necesariamente adopta la asignación de

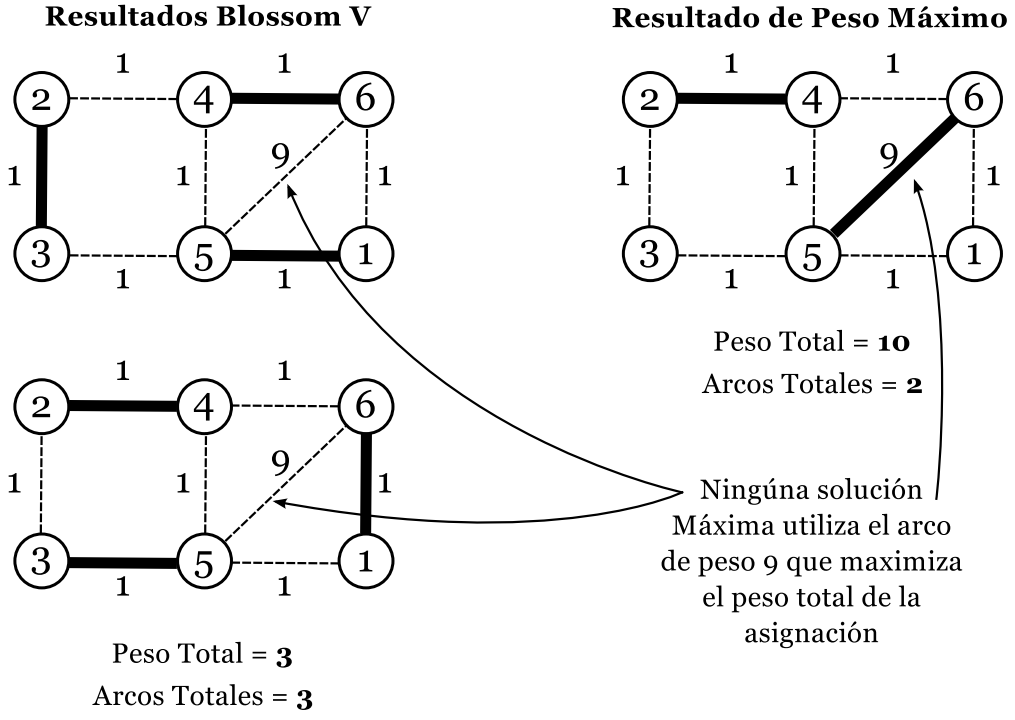


FIGURA 3.4: Asignación de Blossom V no respeta necesariamente el máximo peso

máximo peso total, si no que prioriza la perfección de la misma. En otras palabras, busca maximizar el peso dentro del grupo de soluciones formado por aquellas que tengan el máximo número de arcos posibles (3 arcos en la Figura 3.4). Sin embargo, esta solución puede tener un peso total menor que otras con una menor cantidad de enlaces como la ilustrada con un peso total de 10 con 2 arcos.

En general, resulta de particular utilidad para nuestro diseño de algoritmo FCP, que el esquema de asignación permita obtener el máximo peso independientemente de si la cantidad de arcos elegidas es máxima o no. En otras palabras, en FCP estaríamos dispuestos a relegar la elección de un conjunto de arcos con el fin de elegir una menor cantidad pero con mayor peso total. Además, también se requiere que se levante la condición de vértices pares en el grafo $G(V, E)$. En consecuencia, retomamos el uso de una reducción al grafo de entrada a Blossom V que nos permite obtener estas características finales propuesta por Guido Schafer en [106].

El proceso de reducción de Schafer implica duplicar el tamaño del grafo para clonando los vértices existentes y conectándolos a sus respectivas copias con arcos auxiliares. Además, también se clonan los arcos del grafo original. Este proceso permite resolver el problema de la asignación máxima ponderada (no perfecta) y se define formalmente a continuación y se ilustra en la Figura 3.5. Cabe destacar que el grafo original de la figura no podría ser sometido a una asignación máxima dado que el número de nodos es impar (3).

1. Dado un grafo G de vértices V, arcos E, y pesos w $(G(V, E, w))$,
2. Se crea una copia $G^*(V^*, E^*, w^*)$ tal que:
   a) Cada vértice $v^* = v$,
   b) Cada arco $e^* = e$,
   c) Cada peso $w^* = w$,
3. Se considera la suma $G'(V', E', w') = G(V, E, w)+G^*(V^*, E^*, w^*)$ mas un conjunto de vértices de todo $v*$ a v con peso $w' = 0$ tal que:
   a) $V' = VV^*$,
   b) $E' = EE^*\{vv^* : v$está en $V y v^*$ en $V^*\}$,
   c) $w' = \{w$ cuando en E, w^* cuando en E^*, y 0 cuando en $vv^*\}$,
4. Ahora, si se resuelve la asignación perfecta máxima en G' (Blossom), se obtiene la asignación máxima ponderada (no perfecta) en G.

De esta manera, el algoritmo Blossom es capaz de encontrar una asignación máxima de manera tal que cada vértice es asignado con un arco como máximo en el resultado final.

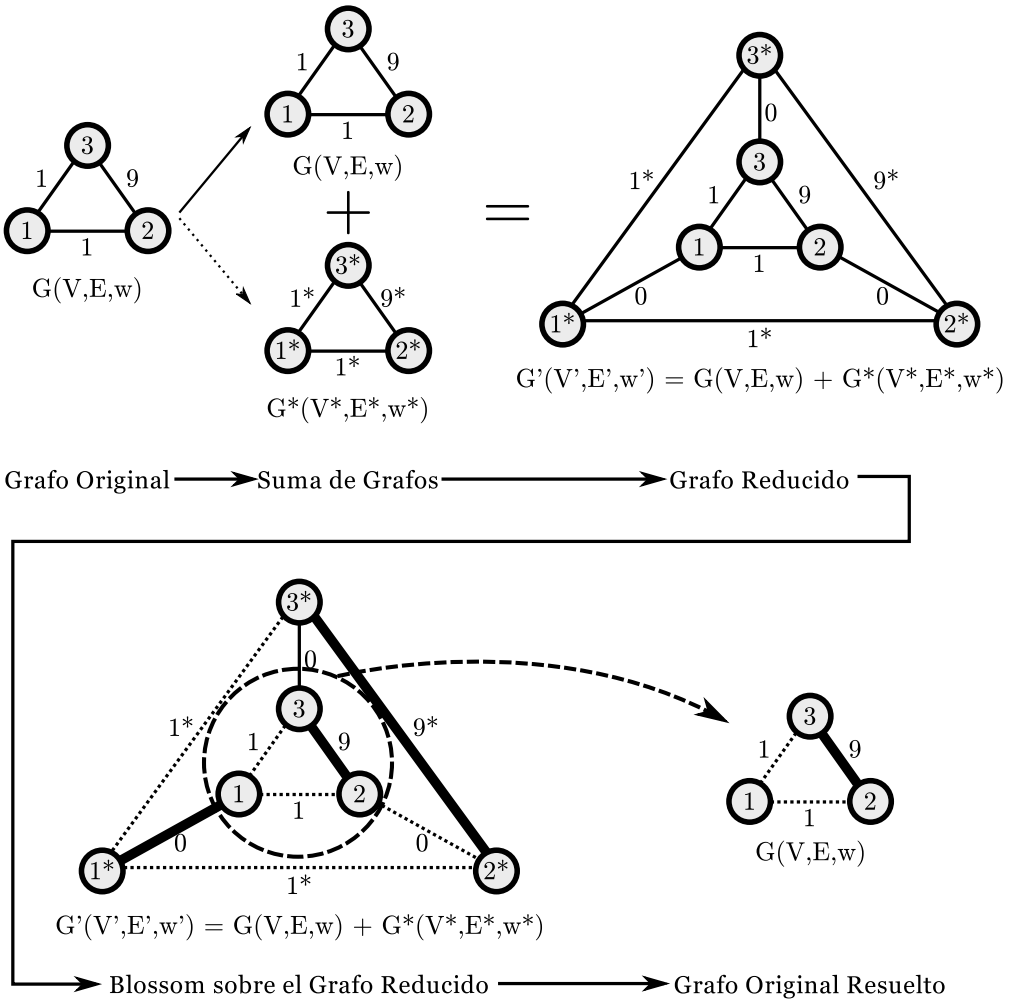


FIGURA 3.5: Reducción de Schafer para el cálculo de la asignación máxima ponderada no perfecta

Además, con la reducción planteada el mismo puede considerar pesos de manera tal que el grafo resultante otorgue el máximo peso posible en la asignación final. Este efecto es de particular interés para resolver el plan de contacto como describimos a continuación.

3.3.2. Algoritmo FCP

En síntesis, dada una matriz de topología de contacto $[P]$, Blossom encuentra una matriz $[L]$ cuyos arcos son un subconjunto de $[P]$ y respetan la condición $[I]_i = 1 \forall i$ para todo k. Por otro lado, este algoritmo permite decidir el conjunto de asignaciones óptimas en función de un peso o afinidad entre los pares de nodos. Dado que en nuestro caso buscamos mejorar el parámetro de justicia a lo largo del tiempo, el peso de los arcos podría estar determinado de manera proporcional al tiempo en el que ese arco se ha mantenido sin elección anteriormente. En efecto, con una aproximación de programación dinámica, el cálculo de FCP tiene como objetivo el maximizar el mínimo y minimizar

Algoritmo 1: Algoritmo de justicia FCP

input : Topología de Contacto $[P]$ de tamaño $K \times N \times N$
Tiempo de Estados $[T]_k$
output: Plan de Contacto $[L]$ de tamaño $K \times N \times N$

1 $DCT_{i,j} \longleftarrow 0 \quad \forall i, j;$
2 **for** $k \leftarrow 0$ **to** K **do**
3 $\quad [W]_{k,i,j} \longleftarrow DCT_{i,j} \quad \forall i, j$
4 $\quad$ `Blossom(`$[P]_k$`,`$[L]_k$`,`$[W]_k$`);`
5 $\quad$ **if** $[L]_{k,i,j} = 0$ **then**
6 $\quad\quad DCT_{i,j} \longleftarrow DCT_{i,j} + t_k \quad \forall i, j$

el máximo tiempo de asignación de los arcos. En otras palabras, a medida que un arco acumula tiempo sin elección, el mismo toma mayor prioridad sobre otro que ha sido electo mas frecuentemente.

El Algoritmo 1 muestra la definición formal de FCP como aquí lo planteamos. A través de K iteraciones (una por cada estado del modelo FSM del sistema), el algoritmo FCP mantiene en memoria la cantidad de tiempo que cada contacto i a j se ha mantenido desactivado ya sea por imposibilidad física ($p_{k,i,j} = 0$) o por que no ha sido elegido previamente ($l_{k,i,j} = 0$). Esta memoria se conserva en la matriz Tiempo de Contacto Deshabilitado o *Disabled Contact Time* (DCT) en Inglés la cual es inicializada a $DCT_{i,j} = 0 \quad \forall i, j$ en la línea 1 del algoritmo. En cada iteración, los pesos $[W]_{i,j}$ de los arcos a operar se calculan en base a estos valores de DCT los cuales son acumulados ($DCT_{i,j} = t_{k1} + t_{k2} + ... + t_{kn}$) para cada par i, j en la línea 3. En consecuencia, los enlaces que tienen pocas (o ninguna) ocurrencia en el plan de contacto final en $[L]$, obtienen un peso significativamente mayor que aquellos cuyo tiempo de elección es alto, y por ende tienen prioridad de elección al ser sometidos al procedimiento Blossom ponderado de la línea 4 ya descrito en la sección 3.3.1.1. Finalmente, $DCT_{i,j}$ se actualiza en la línea 6 de acuerdo a las decisiones tomada por Blossom en esta iteración. De esta manera, el algoritmo FCP concluye una vez que se repitió este procedimiento para cada uno de los estados del modelo FSM. La Figura 3.6 ilustra gráficamente las iteraciones sobre una topología ejemplo.

Respecto a la complejidad, al día de la fecha existen Implementaciones eficientes de Blossom como Blossom V [105] que resuelven este algoritmo en tiempos polinómicos del orden de $O(n^2 l)$ donde n es el número de nodos y l la cantidad de arcos. Dado que en nuestro caso resolvemos el problema de asignación ponderada máxima no perfecta por medio de una reducción de grafos, la complejidad del proceso resulta $O(2n^2(2l + n))$. Finalmente, el algoritmo FCP itera a través de la totalidad de estados K del modelo FSM, se demuestra que la complejidad final de FCP resulta $O(2kn^2(2l + n))$.

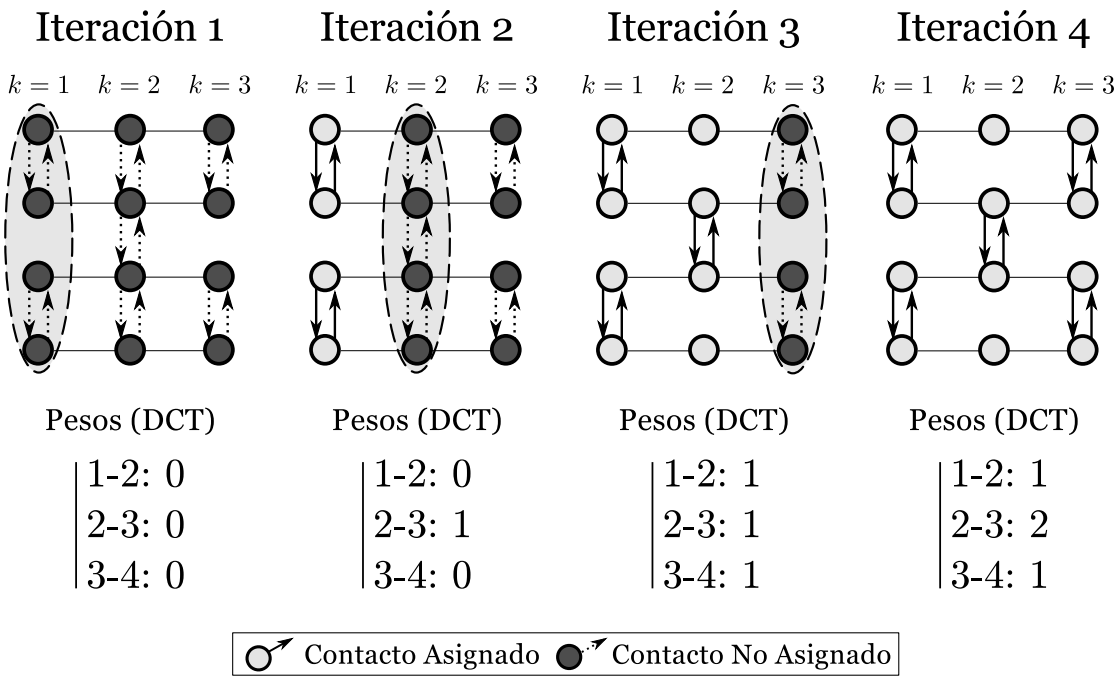


FIGURA 3.6: Comportamiento del algoritmo FCP sobre una topología simple

Por último, pero no menos importante, la naturaleza de programacion dinámica (evolutiva en un modelo de tiempo) en la que se basa el esquema FCP, permite distribuir la asignación de los contacto no sólo equitativamente entre los pares de nodos i, j si no que también a lo largo del tiempo de intervalo de topología. Esta cualidad resulta de particular interés para implementaciones de tráficos enrutados reales en redes DTN como mostraremos en el análisis de FCP en la siguiente sección 3.4.

3.4. Análisis de Plan de Contacto Basados en Topología

En esta sección realizaremos un análisis de los esquemas planteados para el diseño de plan de contacto basados en topologías tanto a nivel modelado formal (modelos MILP) como los planteos computacionalmente eficientes basados en algoritmos.

3.4.1. Métricas de Evaluación

Con el fin de cuantificar las características de los planes de contactos diseñados, proponemos el siguiente conjunto de métricas de evaluación para usar en futuros análisis y comparaciones.

- Tiempo de contacto de arco (Arc Contact Time):
  $arcConT_{i,j} = \sum_k l_{k,i,j} * t_k$

- Tiempo de contacto de sistema (System Contact Time):
  $sysConT = \sum_{k} \sum_{i} \sum_{j} l_{k,i,j} * t_k$

- Índice de justicia Min-Max (Min-Max Fairness Index):
  $minMaxTRatio = \frac{min_{i,j}(\sum_{k} l_{k,i,j} * t_k)}{max_{i,j}(\sum_{k} l_{k,i,j} * t_k)}$

- Índice de justicia Raj-Jain (Raj-Jain Fairness Index) [107]:
  $JIndexRatio = \frac{(\sum_{i} \sum_{j} \sum_{k} l_{k,i,j} * t_k)^2}{(i*j) * \sum_{i} \sum_{j} (\sum_{k} l_{k,i,j} * t_k)^2}$

La métrica *Arc Contact time* permite tener una medición directa del tiempo total que un par de nodos i, j permanecen conectados (en contacto) a lo largo del intervalo de topología en el plan de contacto final expresado en $[L]$. Por otro lado, la suma de estos tiempos para todos los pares de nodos del sistema se refleja en el métrica *Total System Contact* y da una noción de la cantidad de recursos de comunicación efectivamente disponibles para su utlización en el sistema orbital como un todo. En otras palabras, estas métricas cuantifican la capacidad de comunicación que comparten dos nodos i, j y la capacidad total de la red respectivamente.

Además de las métricas de capacidad nombradas, definimos otras que nos permiten analizar con que justicia esa capacidad fue distribuida en el plan de contacto. En efecto, la métrica *Min-Max Fairness Index* compara el arco mas penalizado contra el mas beneficiado en la topología resultante. Si bien esta métrica es coherente con el criterio de optimización de el modelo MILP de dos etapas detallado en la sección 3.2, esta peca de ignorar la justicia de los contactos o pares de nodos no extremos (intermedios) entre el mínimo y el máximo. Por ende, incorporamos una segunda medición de justicia *Raj-Jain Fairness Index* basada en el criterio de justicia de Raj-Jain [107] que nos permita completar el análisis del diseño.

3.4.2. Análisis Sobre Topología de Contactos Aleatorias

Con el fin de evaluar el algoritmo FCP y los modelos formales, proponemos un estudio sobre topologías de contactos aleatorias para entender el comportamiento general de los mismos. En efecto, realizamos un primer análisis sobre una topología aleatoria y puntual, para luego considerar una serie de las mismas.

3.4.2.1. Topología de Contacto Puntual

En particular, sugerimos un primer caso aleatorio de $K = 30$ estados con $t_k = 10$ unidades de tiempo de duración por estado, $N = 10$ nodos, y una densidad de existencia de oportunidad de comunicación (enlace) entre ellos del 30 %. Específicamente, la densidad de enlace hace referencia a las posibilidades de que un conjunto de contactos uniforme y aleatoriamente distribuidos tiene de existir en el contacto de topología en $[P]$.

Como se explicó en la sección 3.2, el modelo de justicia basado en MILP (aquí denominado Fair_LP) cuenta con dos parámetros de configuración ϵ y β que en este caso puntual de evaluación serán establecidos en 0,1 y 1 respectivamente. Esto implica que el valor mínimo de tiempo de contacto entre todos los pares de nodos (t_{min}) debe ser elevado en la etapa 1 sin importar el costo de capacidad del sistema en su totalidad, y que no se perderá nada de capacidad en la etapa 2.

Además del modelo de justicia formal Fair_LP, también incluimos el esquema de máxima capacidad (MCP) aquí denominado MaxC_LP para contar con una referencia válida que ilustre la cota superior de la métrica $sysConT$. Cabe recordar que MaxC_LP se obtiene de la primer etapa de Fair_LP para $\epsilon \gg t_{min}$. De esta manera, MaxC_LP nos permitirá determinar la penalidad en capacidad en la que incurrirán los esquemas Fair_LP y el algoritmo FCP al optimizar la justicia en los planes de contactos resultantes. El algoritmo FCP utiliza los mismos parámetros detallados en la sección 3.3. En general, la hipótesis es que Fair_LP entregará las mejores métricas de justicia, que MaxC_LP entregará mas mejores métricas de capacidad global, y que FCP generará soluciones sub-óptimas pero eficientes en términos computacionales.

La Figura 3.7 ilustra un histograma en el cual el par de nodos i, j se ubica en el eje de las abscisas y su correspondiente tiempo de contacto acumulado ($arcConT_{i,j}$) en el eje de coordenadas. Para estudiar el comportamiento de los esquemas de diseño los pares de nodos se ordenan de acuerdo a su capacidad de contacto en la topología de contacto $[P]$. En la figura se comparan los valores de estas métricas para los esquemas Fair_LP, MaxC_LP, FCP y la capacidad en el topología de contactos denominada capacidad física o PhyC. Por otro lado, las otras métricas del sistema se resumen en la Tabla 3.2.

Tabla 3.2: Métricas de Topología de Contacto Puntual

	Modelo Fair_LP	FCP-DTN	Modelo MaxC_LP
sysConT	2640	2640	2640
minMaxTRatio	0.5	0.25	0
Jain Index	0.841	0.825	0.731

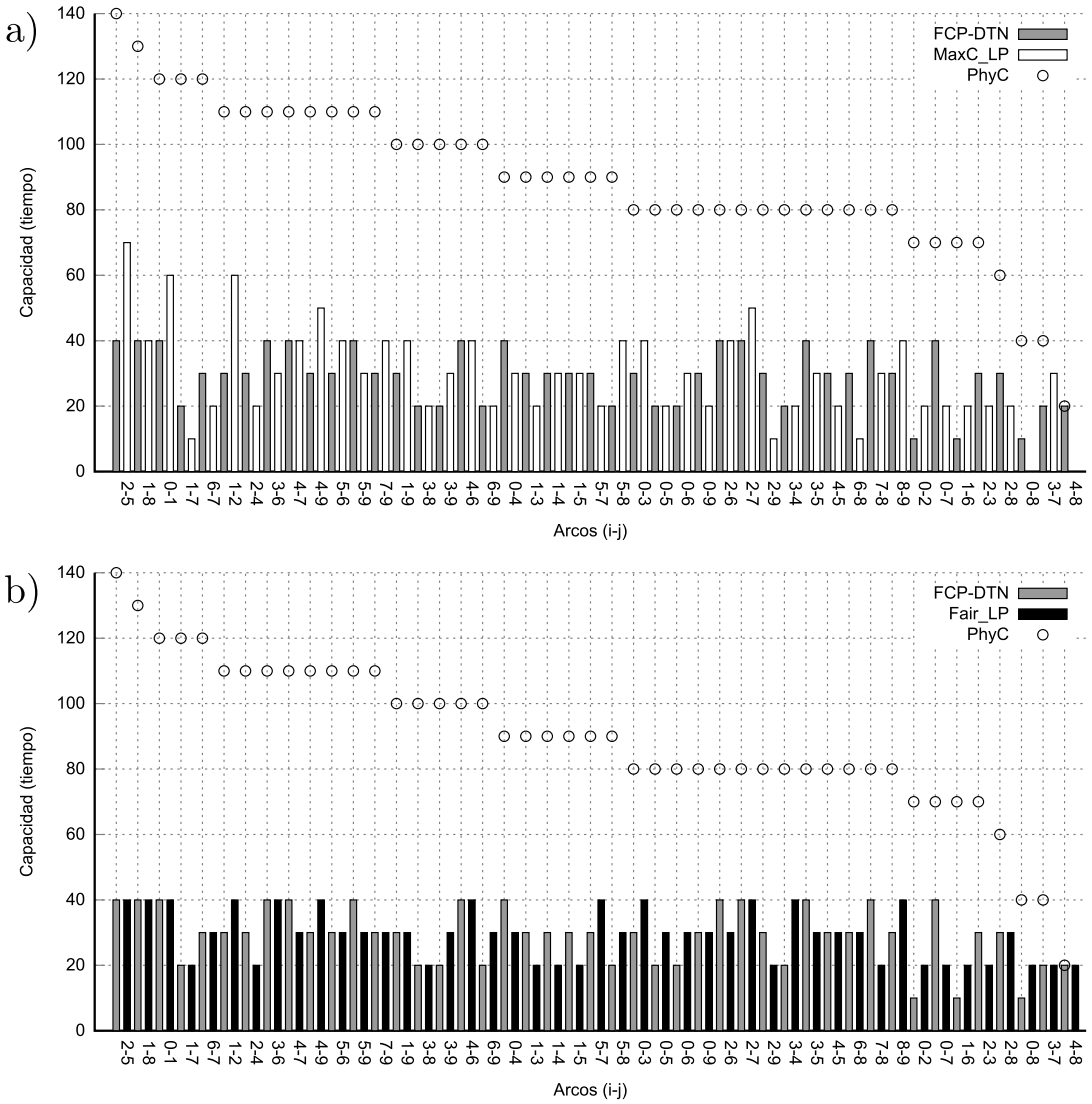


FIGURA 3.7: Distribución de capacidades de arcos para FCP con comparación con a) MaxC_LP y b) Fair_LP

A pesar de la desigual distribución de la capacidad de física sugerida por la curva formada por los valores PhyC de la Figura 3.7, los esquemas Fair_LP y FCP muestran una administración equitativa de la capacidad de cada par de contactos. En particular, aquellos con menor capacidad física (ubicados hacia la derecha del histograma) obtienen tratamiento prioritario al contar con una asignación de enlace tan alta como sea posible (t_{min}). Como se había hipotizado previamente, el esquema de máxima capacidad MaxC_LP no evidencia este fenómeno y tiende a ajustarse a la realidad física original del sistema (pendiente de la curva PhyC). En otras palabras, MaxC_LP proporciona planes de contactos proporcionales a la distribución de la topología de contactos, derivando en posibles pares de nodos i, j privados de cualquier conexión durante el intervalo de topología.

Por ejemplo los arcos $(4, 8)$ y $(0, 8)$, de capacidad física 20 y 40 respectivamente, resultan de capacidad nula (0 y 0) al ser diseñados con MaxC_LP, mientras que FCP los configura con 20 y 10 unidades de tiempo y Fair_LP con 20 y 20 respectivamente. En consecuencia, la capacidad nula de estos arcos en MaxC_LP deriva en que su métrica $minMaxTRatio$ resulte $min_{i,j}(\sum_k l_{k,i,j} * t_k) = 0$ como se ilustra en la Tabla 3.2.

Por otro lado, es interesante observar que a pesar de que las métricas de justicia tanto de Fair_LP y FCP mejoran las generadas por MaxC_LP, todos cuentan con la misma capacidad total de sistema ($sysConT$). Esto sugiere que para esta topología en particular, un conjunto de varias soluciones pueden aportar una capacidad global de 2640 unidades de tiempo entre las cuales la óptima en términos del criterio min-max es elegido por el esquema Fair_LP y un sub-óptimo por FCP. Sin embargo, en la siguiente sección 3.4.2.2 mostramos que estadísticamente Fair_LP genera pérdidas de capacidad con el fin de mejorar la justicia (optimizar el valor de t_{min}).

3.4.2.2. Topología de Contacto General

Con un medio mas general para analizar los métodos de programación lineal y algorítmicos propuestos, en esta sección los sometemos a un estudio estadístico sobre una serie de topologías aleatorias con diferentes densidades de contactos. En particular, se generaron 10000 topologías de contacto $[P]$ con una probabilidad uniforme de existencia de oportunidades de comunicación entre nodos variable entre 6 % a 24 % para un total de $K = 30$ estados y $N = 10$ satélites. En general, estos valores de configuración permiten a los modelos MILP ser resueltos en tiempos razonables por solvers disponibles como GLPK [108]. Por otro lado, el planteo algorítmico no evidencia esta restricción temporal al contar con una complejidad de incremento polinómico como el discutido en la sección 3.3.2.

Los resultados estadísticos obtenidos se ilustran en la Figura 3.8 y se analizan a continuación. En la Figura 3.8 a), la capacidad total del sistema representada en la métrica $sysConT$ ilustra el caso general del comportamiento descrito en la tabla 3.2. En particular, tanto MCP como FCP entregan la capacidad óptima del sistema para la totalidad de los casos evaluados, mientras que Fair_LP evidencia una pérdida en promedio proporcional a la densidad de enlace de las topologías de contactos generadas. Esta pérdida de capacidad corresponde con el comportamiento de optimizar (maximizar) la variable t_{min} inclusive si esto deriva en la anulación de un número significativo de arcos. Por otro lado, FCP selecciona estado a estado aquella asignación producto de la combinación mas justa de arcos que satisfaga el criterio de máximo peso del algoritmo Blossom explicado en la sección 3.3.1.1. En consecuencia, FCP mantiene una asignación de capacidad óptima

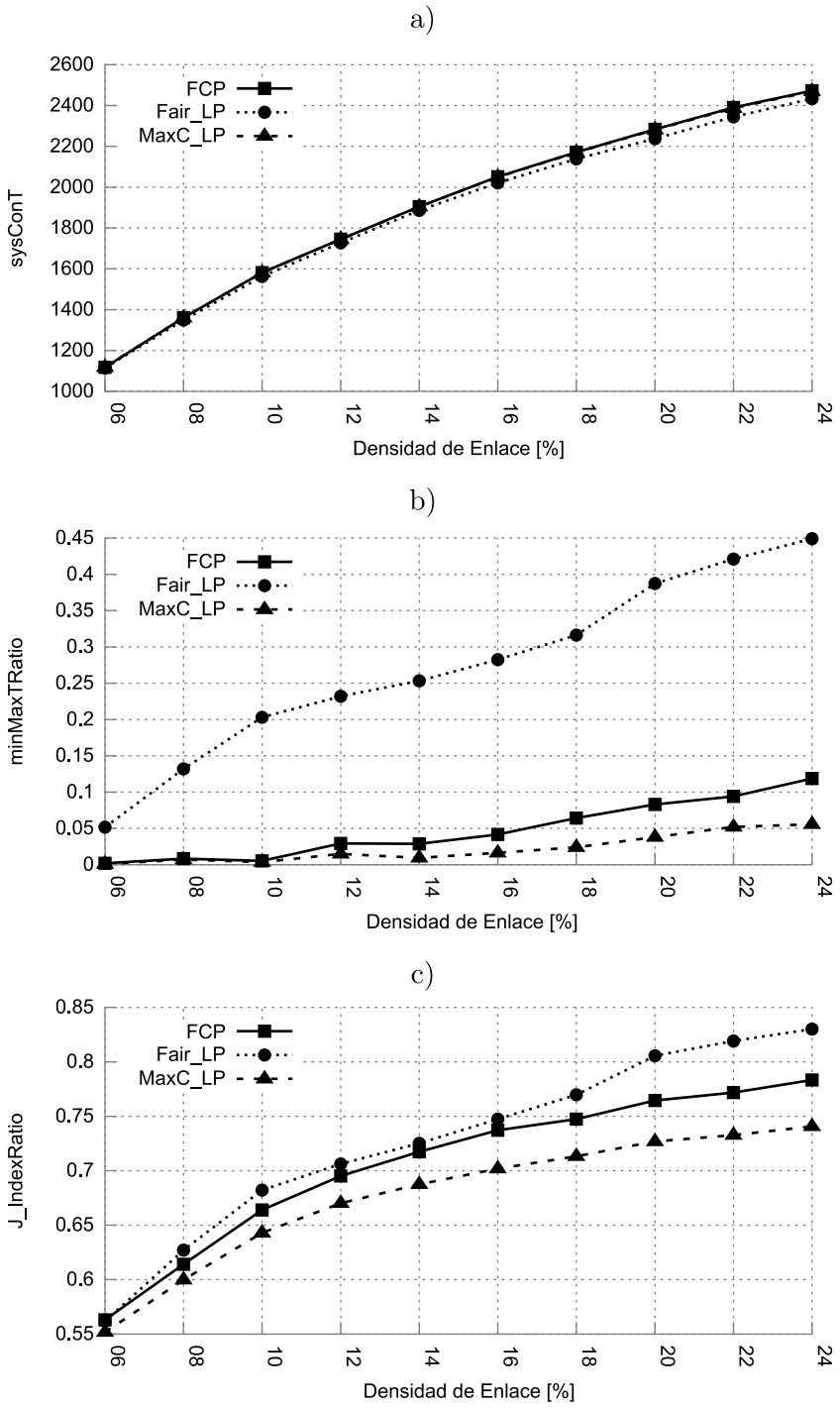


FIGURA 3.8: Métricas estadísticas para 10000 ejecuciones de a) capacidad de sistema, b) justicia min-max, y c) ratio de Jain Index

siempre y cuando ningún arco tenga mayor peso que la suma de sus competidores en el mismo estado k, lo que es poco probable en una topología de contacto basado en una distribución de contactos uniformes.

En general, FCP, como fue planteado en la sección 3.3.2, toma decisiones menos drásticas que el modelo formal Fair_LP pero otorgando un mejor rendimiento en términos de

capacidad total de sistema (*sysConT*). Por otro lado, en lo relativo a las métricas de justicia, las Figuras 3.8 b) y c) demuestran que FCP aporta planes de contactos con justicia intermedias entre MCP y Fair_LP para la generalidad de los escenarios aleatorios generados en este análisis.

En conclusión, FCP permite el diseño de planes de contactos con un llamativo balance entre justicia y capacidad total de sistema de particular interés para la planificación general de redes de satélites DTN como discutimos en la siguiente sección 3.4.3.

3.4.3. Caso de Referencia y Estudio B: Topología Lineal Ecuatorial

Muchas trabajos previos se han enfocado en desarrollar topologías satelitales que den soporte a la provisión de cobertura global para servicios de comunicación extremo a extremo permanente (por ejemplo de voz) [109]. Sin embargo, esta rama sigue como

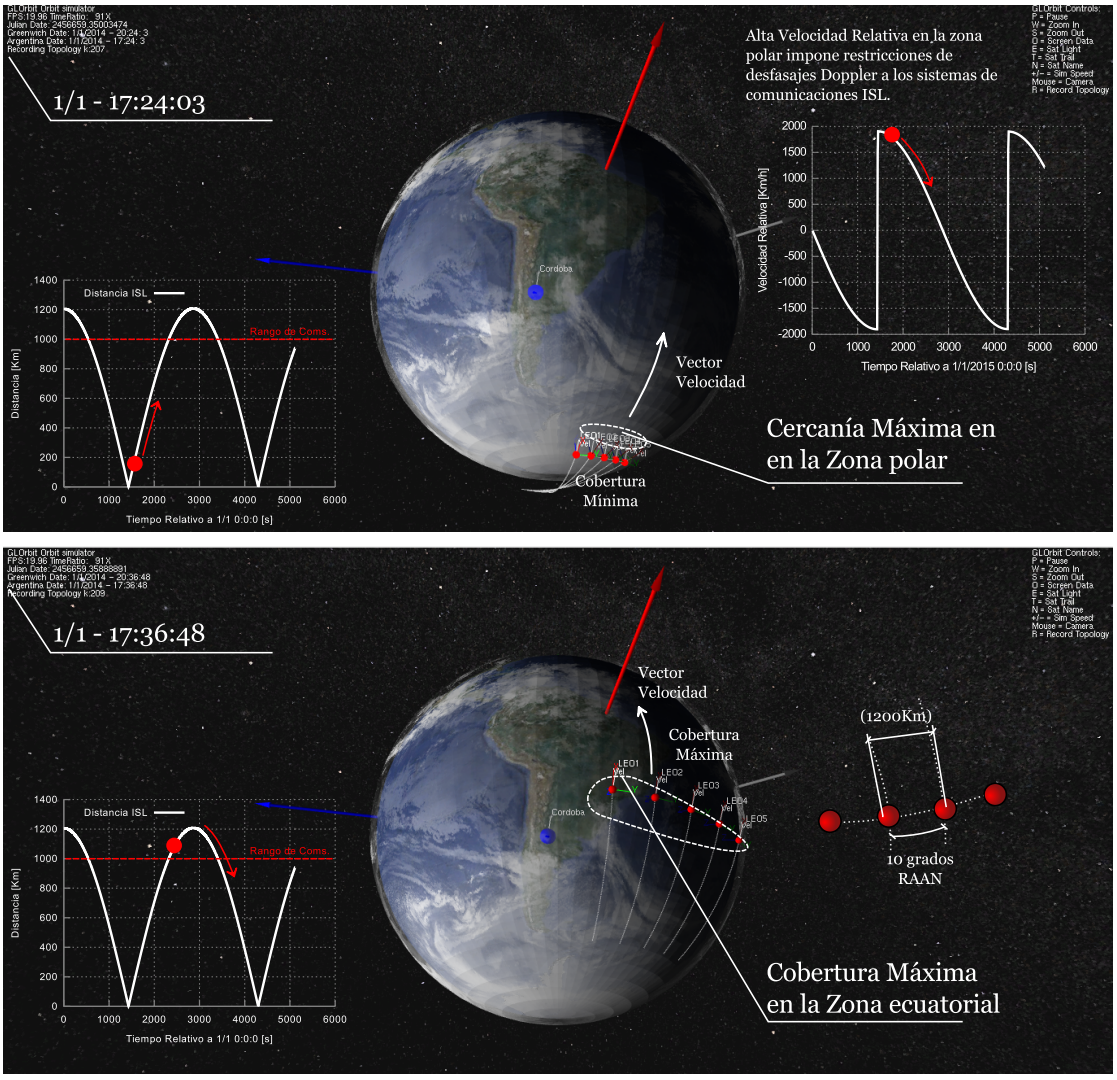


FIGURA 3.9: Representación del Caso de Estudio B en GLOrbit

tema pendiente de investigar para el caso mas general de redes DTN, donde el término *topología eficiente* probablemente necesite ser re-definido en un contexto donde la conectividad extremo a extremo ya no es una propiedad del sistema. En general, el término eficiencia se relaciona directamente con la aplicación final de la red [110]. En este contexto, y con el fin de evaluar el desempeño del algoritmo FCP en una constelación satelital real proponemos una formación en linea paralela al ecuador donde la constelación de sensores permite una cobertura óptima en el zona poblada y numerosas posibilidades de comunicación a medida que los sensores se aproximan a la zona de los polos como se muestra en la Figura 3.9.

En particular, estudiaremos un escenario con 5 satélites con una inclinación orbital de 90° con un argumento de perigeo levemente desfasado (0°, 0,1°, 0,2°,0,3°, and 0,4°) para evitar la colisión en la zona de los polos. Un movimiento medio de 15,0756 revoluciones diarias y una excentricidad de 0 (órbita circular) derivan en un eje orbital de 6921km 550km de altura sobre el nivel del mar). Una separación del plano orbital de 10° se logra al variar el ángulo RAAN proporcionando distancias intersatelitales máximas de 1220Km en la zona ecuatorial. Dado que asumiremos que la distancia de las comunicaciones no pueden superar los 1000km con antenas omnidireccionales la red oscilará entre estados de conexión y desconexión en cada período orbital. El intervalo de topología considerado es de 24 horas de propagación en base a la cual se genera la topología de contactos $[P]$ a analizar. La Tabla 6.6 resume la lista de parámetros utilizados para generar este caso de estudio de topología de contactos lineal ecuatorial.

Cabe destacar que el caso de estudio propuesto difiere del ilustrado en la sección 2.3.1 del capítulo 2 en una mayor cantidad de segmentos orbitales y una mayor distancia en el plano ecuatorial. Sin embargo, esta topología implica una velocidad relativa entre los sensores en la zona de los polos de importancia que debe tenerse en cuenta al considerar el efecto Doppler (corrimiento de frecuencias). Como se puede observar en la parte superior de la Figura 3.9, se pueden esperar velocidades cercanas a los 2000 Km/h condición para la cual se deben utilizar sistemas de comunicaciones con altas tolerancia al efecto Doppler. Por otro lado, la formación en tren ilustrada en la Figura 2.5 permite utilizar antenas direccionales mas eficientes, mientras en el caso de este sección se deben utilizar antenas de mayor amplitud (omnidireccionales) que permitan alcanzar satélites en diferentes direcciones laterales como se da en la zona polar.

Una vez procesado este caso de estudio en el intervalo de topología propuesto de 24 horas, al modelado como máquina de estado (FSM) hace uso de $k = 241$ estados para representación de la topología de contactos. En ese intervalo se generan exactamente 30 pasadas por los polos (15 por el polo sur y 15 por el polo norte) como se ilustra en el modelado FSM de la Figura 3.10. En la misma se puede observar la evolución de los

Tabla 3.3: Tiempos y Parámetros Orbitales del Caso de Estudio Lineal Ecuatorial

Inicio del Intervalo de Topología	Ene-1st, 2014, 0hs 0min 0sec
Fin del Intervalo de Topología	Ene-2nd, 2014, 0hs 0min 0sec
Coeficiente Bstar (/ER)	0
Inclinación (grados)	90°
RAAN (grados)	0°, 10°, 20°, 30°, y 40°
Eccentricidad	0
Argumento del Perigeo (deg)	0°, 0,2°, 0,4°, 0,6° y 0,8°
Anomalía Media (deg)	0°
Movimiento Medio (rev/day)	15,0756 rev/day
Altura sobre el nivel del Mar (Km)	600 Km

estados en una misma órbita donde a medida que los satélites se acercan a los polos las posibilidades de contactos se extiende de los vecinos inmediatamente contiguos al siguiente, y así sucesivamente hasta llegar a poder comunicarse con todos los segmentos de la constelación (estado $k = 5$). Luego, a medida que acercan hacia el ecuador el efecto inverso deriva en el estado $k = 9$ donde la distancia ISL supera los 1000 Km de rango de comunicaciones. Por último, dado que ningún estado con arcos de contactos en ellos supera la duración de 500 segundos, no se utiliza ninguna estrategia de fragmentación en esta topología de contactos $[P]_{k,i,j}$.

Las métricas colectadas de los planes de contactos diseñados por MaxC_LP (MCP), Fair_LP, y FCP basados en la topología de contacto lineal ecuatorial se ilustran en la Tabla 3.4. En la misma se puede observar que en general, la hipótesis elaborada en las secciones previas se mantiene válida dado que la métrica de $sysConT$ de FCP evidencia el valor óptimo de capacidad mientras que la medición de justicia $minMaxTRatio$ se ubica entre los valores arrojados por los métodos de MCP y Fair_LP. Por otro lado, el fenómeno de que el índice de Jain ($JIndexRatio$) evidencia un rendimiento ligeramente superior en FCP en comparación a Fair_LP requiere de los histogramas de la Figura 3.11 para su explicación.

En particular, FCP asigna parte de la capacidad del arco de menor capacidad (arco $(1, 5)$) a otros contactos de manera equitativa mejorando la distancia media cuadrática en la cual se basa el índice de $JIndexRatio$. Sin embargo, esta acción reduce la capacidad del arco mas penalizado de la topología lo que tiene impacto directo en la métrica de $minMaxTRatio$ la cual considera los arcos con capacidades extremas (mínimas y máximas). En efecto, una situación de compromiso se crea entre $minMaxTRatio$ y $JIndexRatio$ para Fair_LP y FCP. En este análisis vale la pena destacar que precisamente el arco $(1, 5)$ representa la única y escasa oportunidad de comunicación entre los nodos 1 y 5 (en el estado $k = 5$ por ejemplo) quienes justamente son los extremos de la línea de la formación orbital generada.

Tabla 3.4: Métricas de Fair_LP, FCP y MCP para el Caso de Estudio A (topología lineal ecuatorial)

	Fair_LP Model	FCP-DTN	MaxC_LP Model
sysConT [secs]	213940	213940	213940
minMaxTRatio	0.430	0.255	0.157
Jain Index	0.481	0.487	0.469

En general, de estos análisis se puede concluir que mas allá de las ventajas computacionales que caracteriza al algoritmo FCP, las métricas de justicia de los planes de contactos diseñados con esta técnica resultan de una capacidad homogéneamente distribuida, y de una capacidad de sistema satisfactoria para la planificación general de un sistema de constelaciones como el planteado.

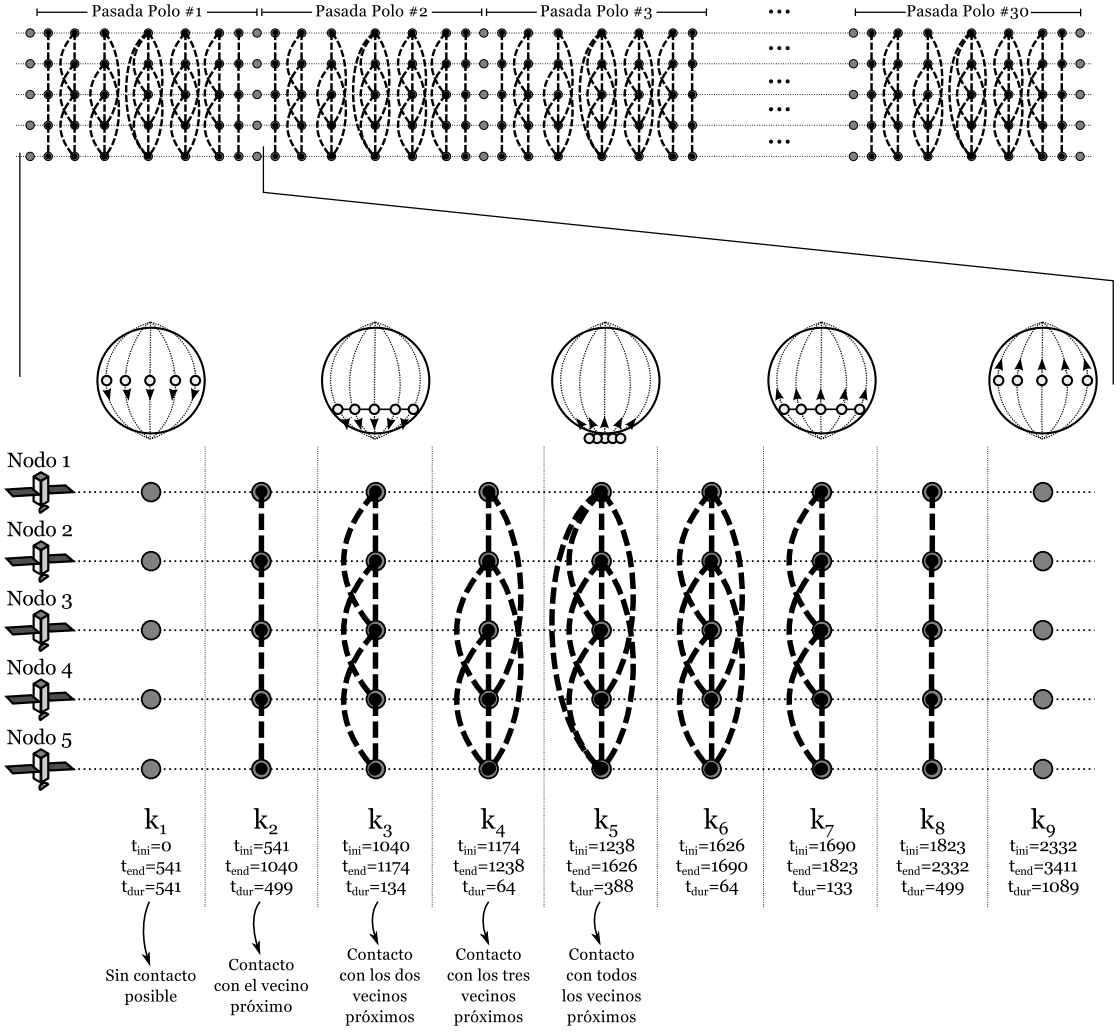


Figura 3.10: Modelado FSM del caso de estudio A (topología lineal ecuatorial)

3.4.4. Resultados de Simulación

3.4.4.1. Descripción del Simulador

En general, las métricas de capacidad y justicia utilizadas en la sección 3.4.3 y descritas en la sección 3.4.1, resultan válidas para el análisis de los planes de contactos diseñados con MCP, FCP y Fair_LP, pero en un sistema real podrían resultar demasiado abstractas y de escasa aplicación a la hora de la toma de decisiones en un centro de control de sistema real. En consecuencia, en esta sección ofrecemos una configuración de red satelital DTN típica para la cual generamos un entorno de simulación adecuado que pueda tomar los planes de contactos resultantes de la sección 3.4.3 e inyectarle tráfico para estudiar el comportamiento final del mismo.

En efecto, se generó un entorno de simulación satelital en el marco de trabajo (*framework* en Inglés) OMNeT++ [111] denominado TotSim (detallado en la sección 2.6.6). En este contexto, se desarrolló e implementó un modelo del protocolo Bundle [46] y una versión del esquema de ruteo Contact Graph Routing (CGR) [59, 60] los cuales se instancian para cada módulo de los segmentos de vuelo. Simulaciones e implementaciones similares a este se pueden encontrar en [87, 96] para referencia. El simulador toma como entrada los planes de contactos calculados en la sección 3.4.3 con el fin de analizar el tiempo promedio que un bundle (unidad de protocolo Bundle) tarda en llegar a su destino final para cada esquema de diseño (MCP, Fair_LP, y FCP) bajo un patrón de tráfico uniforme del tipo "todos contra todos". La métrica de este estudio de simulación es la edad media promedio de bundle o *average bundle age* en Inglés, la que esencialmente representa el intervalo de tiempo desde que el paquete es creado (al inicio de la simulación), hasta que el mismo es entregado finalmente al destino final.

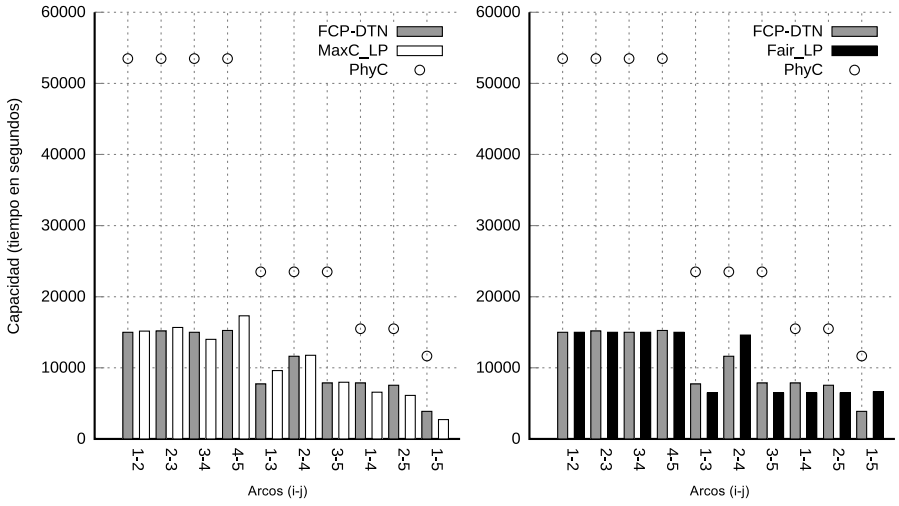


FIGURA 3.11: Distribución de arcos en caso la topología lineal ecuatorial

Los módulos (satelites) del simulador se configuraron con sistemas de comunicaciones full-duplex (comunicaciones bi-direccionales) con una tasa de datos de $10Kbps$ (acorde para operar una distancia ISL de 1000 Km como la planteada) y un tiempo de adquisición del canal (sincronismo) de 2 segundos que deben ser utilizados en cada inicio de un contacto. Por otro lado, se incorpora una cabecera de capa de enlace de 12 Bytes para una carga útil de enlace máxima de $2043Bytes$. En otras palabras, los bundles tendrán que ser fragmentados a este tamaño para poder transmitidos de un satélite a otro. Por último, dado que no se consideran errores en el canal (para concentrar el análisis en el flujo de tráfico sin errores a nivel Bundle), la capa de enlace no tiene configurada ningún tipo de confirmación de recepción resultado su configuración global muy similar al servicio expeditivo del protocolo CCSDS Proximity-1 Data Link protocol [112].

En particular, variamos la generación de datos en la red desde 4 a $7{,}6MBytes$ la cual se deberá distribuir equitativamente entre todos los sensores orbitales que lo componen. En consecuencia, cada uno de ellos generará de 1 a $1{,}9MBytes$ de información para cada uno de los 4 vecinos correspondientes en la red. Además, una segunda simulación con $10MBytes$ se considerará para estudiar el comportamiento de la red en su punto de saturación. El tráfico es generado al comienzo de la simulación y es evacuado bajo el paradigma *store-carry-and-forward* descrito en la sección 1.3.3.2 a medida que el tiempo avanza hasta completar las 24 horas de duración de la topología.

3.4.4.2. Análisis de Resultados

La curva de la Figura 3.12 ilustra el comportamiento de la edad promedio de bundle para diferentes volúmenes de tráfico (todos contra todos) para cada plan de contacto

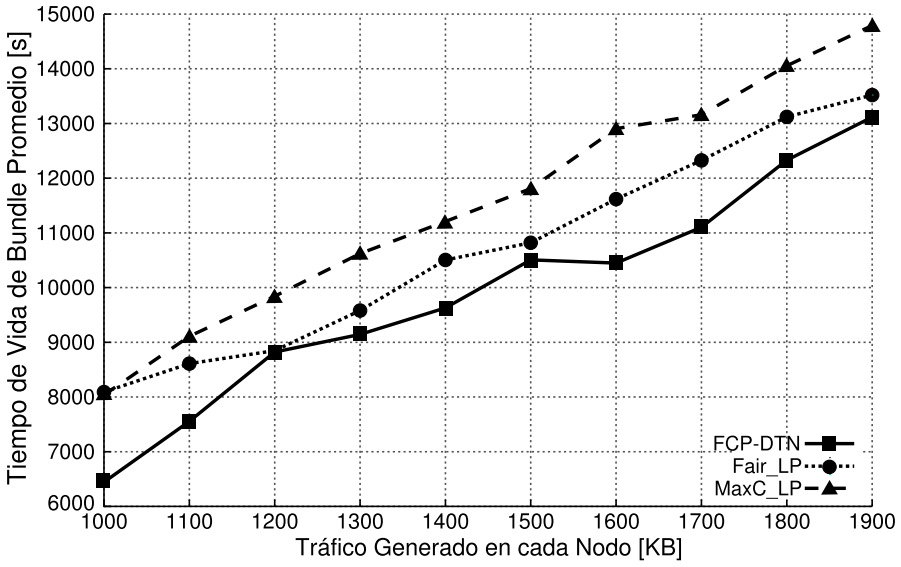


FIGURA 3.12: Edad promedio de Bundle para diferentes valores de generación de tráfico

generado con MCP, Fair_LP, y FCP. De la misma resulta interesante observar que el plan de contacto generado con FCP mejora el rendimiento de la métrica *AverageBundleAge* en relación a los otros esquemas propuestos (Fair_LP y MaxC_LP) para prácticamente todos los volúmenes de tráfico estudiados. La misma situación se repite para el punto de saturación a medida que la carga de tráfico llega a los $8MBytes$, generando los valores ilustrados en la Tabla 3.5. En general, la edad de bundle para puntos mayores a la saturación resultan constantes a medida que el tráfico aumenta dado que el sistema opera al máximo de su capacidad y la métrica se toma para aquellos bundles que efectivamente han llegado a su destino (aquellos que quedan almacenados en nodos intermedios no son tenido en cuenta).

En general, de acuerdo a la hipótesis generada en el planteo formal de los problemas de diseño de plan de contacto basado en criterios de justicia, se esperaba que para un tráfico del tipo *todos contra todos*, los planes de contactos diseñados con Fair_LP evidencien una mejora sustancial a aquellos obtenidos con MaxC_LP (MCP). Sin embargo, puede que el hecho de que las métricas de FCP muestren un mejor rendimiento del algoritmo que su correspondiente planteo formal en el modelo MILP en la sección 3.2 puede resultar llamativo por lo que requiere de la siguiente explicación. A diferencia del planteo formal, FCP se basa en un paradigma de programación dinámica (dynamic-programming en Inglés) en el cual la justicia entre los enlaces se va ajustando en un proceso de estado a estado. Por otro lado, el modelo MILP trabaja con una visión global de la topología y no considera que la justicia se aplique de manera sostenida o igualitaria a lo largo del tiempo. En otras palabras, el esquema MILP podría considerar activar varios arcos de manera conjunta al comienzo de la topología (lo que puede resultar justo al observar la topología en su totalidad), mientras que FCP busca aplicar el criterio estado a estado. Es decir, FCP no sólo es justo en la asignación de capacidad entre pares de nodos, si no que también a lo largo del tiempo (ver Figura 3.6) lo cual resulta de extremo beneficio en redes DTN como las analizadas al entregar planes de contactos con rutas entre todos los nodos si es que es posible.

En conclusión, en este caso de estudio, hemos implementado los planes de contacto en una red realista y enrutada logrando validar la suposición de que un plan de contacto diseñado bajo el criterio de justicia favorece el cursado de tráfico del tipo *todos contra todos*. En particular, FCP además de ser computacionalmente conveniente, fue capaz de entregar planes de contactos que mejoraron las métricas inclusive de sus planteos teóricos elaborados en este capítulo.

Tabla 3.5: Edad promedio de Bundle para el punto de saturación del sistema

	Fair_LP Model	FCP-DTN	MaxC_LP Model
A.Bundle Age [s]	43000	41000	44500

3.5. Comentarios Finales Sobre el CPD Basado en Topología

A lo largo del capítulo 3 hemos investigado el diseño de planes de contactos basados en la información de topología con un criterio de justicia. En general, este criterio nos permitió crear y diseñar planes de contactos eficientes a pesar de no utilizar información ni de esquema de ruteo a utilizar ni del tráfico que finalmente cursará el sistema. Sin embargo, al someter estos planes de contactos en un entorno simulado, los resultados observados resultaron satisfactorios especialmente para el planteo algorítmico (FCP). El mismo, a pesar de ser limitado a la especificidad de restricción de una interfaz ($i_i = 1$) para todos los nodos, ha mostrado métricas que inclusive superan los planteos teóricos y formales.

Siendo uno de los primeros avances en el área de diseño de plan de contactos, el algoritmo FCP fue publicado en la conferencia IEEE Wireless for Space and Extreme Environments (WiSEE) [3] (Baltimore, USA) en el 2013 y luego extendido con análisis de simulación para ser publicado en la revista IEEE Sensor Journal [2] en el 2014. Sin lugar a dudas estos trabajos abrieron las puertas a considerar las redes DTN como un esquema viable en el cual se pueden aprovechar estrategias específicas de planificación para obtener el mejor rendimiento del sistema.

De esta manera, una vez explorado el campo de CPD basado solamente en información topológica, se avanzó a considerar una mayor cantidad de información como por ejemplo el esquema de enrutamiento que los nodos utilizarán para determinar las rutas del tráfico generado y en curso. En general, y dado que ya los modelos planteados en este capítulo resultan de complejidad considerable inclusive para tiempos de análisis acotados, la incorporación de nuevas fuentes de información suponen un desafío de importancia a la hora de considerar procedimientos eficientes para el diseño de planes de contactos mas específicos. A continuación, en el capítulo 4 afrontamos este desafío con el uso de metodologías heurísticas con interesantes y prometedores resultados.

Capítulo 4

Diseño de Plan de Contactos basado en Rutas

4.1. Introducción

En el capítulo 3 se introdujo la problemática del diseño de plan de contactos basado en la información disponible de topología la cual es suficientemente predecible de acuerdo a modelos de propagadores orbitales mas o menos precisos [88]. En efecto, el esquema FCP mostró ser eficiente a la hora de dar servicio a un patrón de tráfico distribuido entre los nodos. Sin embargo, y en general, en el caso de la redes satelitales como las aquí tratadas, se suele tener un mayor conocimiento del sistema que da lugar a la generación de planes de contactos que se ajusten de mejor manera a las necesidades específicas de la red, redundando en operaciones mas eficientes. En particular, en este capítulo abordamos la incorporación de información de rutas al diseño, la cual nos permite evolucionar de una evaluación de un salto simple como el estudiado en el capítulo 3 a un análisis de múltiples saltos por nodo.

El desafío aquí enfrentado es significativamente mas complejo que el basado en topología requiriendo de estrategias alternativas a los modelos óptimos. En efecto, en este trabajo se derivan una serie de mecanismos de interés que fueron aceptados en la comunidad DTN por medio del artículo "Routing-Aware Fair Contact Plan Design for Predictable Delay Tolerant Networks"publicado en la revista Ad-Hoc Networks de la editorial ElSevier [4] a principios del 2015. En el mismo obtenemos un esquema de diseño de planes de contacto basado en rutas o Route-Aware Contact Plan (RACP) detallado a continuación.

4.1.1. Suposiciones del Esquema

El esquema de diseño de planes de contactos basado en rutas, al igual que el basado en topología, asume que existe una topología de contactos disponible que agrupa todas las posibilidades de comunicaciones entre nodos (contactos) en un período de tiempo determinado (intervalo de topología). Por otro lado, el esquema también asume que se conoce de antemano la *estrategia de enrutamiento* que utilizarán los nodos de vuelo para dirigir los paquetes tanto generados por si mismo como aquellos recibidos de otros vecinos con un destinatario diferente de él mismo. En general, este comportamiento es factible de ser determinado dado que el centro de operación de misión (MOC) suele tener un grado de conocimiento suficiente del código de software que controlan los satélites.

4.2. Planteo Formal del Problema

En esta sección plantearemos y detallaremos el concepto de ruta en DTN, para luego introducir uno de los esquemas mas clásicos para la determinación de las mismas, y finalmente plantear el problema de diseño de plan de contacto basado en rutas.

4.2.1. Definición de Ruta en DTN

Por definición, *encaminamiento* (*enrutamiento* o *ruteo*) es la función de buscar un camino entre todos los nodos posibles en una red. En particular, un camino se define como una secuencia de nodos por medio los cuales el tráfico pueda llegar a su destino, aunque en DTN deberemos incorporar el tiempo como mostraremos a continuación en esta sección. En general, lo que se busca es una ruta que optimice alguna métrica en particular como por ejemplo la mínima cantidad de saltos (nodos intermedios), el mínimo costo (suma de los costos de cada enlace utilizado), entre otros. Sin embargo, cuando las redes tienen enlaces disruptivos (DTN) estas métricas deben ser re-pensadas y definidas en un contexto diferente al de las redes permanentemente conectadas.

En particular, existen dos propiedades en las topologías de redes DTN que las diferencian de las de conexión permanentes (red IP o de Internet) para las cuales existen abundantes esquemas y métricas de ruteo;

1. La topología varía en el tiempo a medida los enlaces se crean y se deshacen por efecto del movimiento de los nodos de la red.

2. La topología se encuentra altamente particionada o dividida lo que evita que exista un camino o ruta extremo a extremo en un momento dado. En efecto, la observación

de la misma en cualquier momento del intervalo de topología, el grafo resulta discontinuo.

A pesar de que existen protocolos de movilidad que pueden mantener rutas en un entorno dinámico, todos asumen que existe un camino extremo a extremo entre fuente y destino para iniciar la conexión o transferencia de datos. Dado que esta premisa no es necesariamente válida en redes DTN, estos esquemas de ruteo resultan de escasa utilidad para el caso aquí tratado.

En consecuencia, el enrutamiento en DTN debe basarse, considerar y utilizar el principio de *store-carry-and-forward* tratado en el capítulo 1. En efecto, una ruta en DTN se forma con una secuencia de contactos $Route = \{C_1, C_2, C_3, ...C_n\}$ por medios de los cuales el tráfico de paquetes deberá fluir para llegar a su destino. Utilizar contactos en lugar de nodos permite incorporar el factor tiempo en la definición de ruta. En efecto, cuando los contactos se dan en tiempos diferentes, el paquete debe ser almacenado en un almacenamiento persistente de algún nodo intermedio hasta que el siguiente contacto se habilite.

4.2.1.1. Mecanismos de Enrutamiento

Existen diferentes aproximaciones la implementación y determinación de las rutas en DTN como listamos y describimos a continuación.

1. Una alternativa es centralizar el calculo de las rutas para luego ser distribuidas a los nodos de antemano de manera similar a lo que se hace con el plan de contacto como lo proponen Merugu et al. en su algoritmo basado en Floyd-Warshall (MFW) [61]. Describiremos esta técnica en detalle en la sección 4.2.2.

2. Otra opción es que cada satélite utilice el plan de contacto para derivar posibles caminos para el tráfico generado o recibido con destino no local como lo plantea el esquema de Contact Graph Routing (CGR) [60]. Describiremos profundamente este esquema tan popular en la sección 6.3.1 del capítulo 6. A su vez, esta opción puede darse de dos maneras.

   a) Que cada nodo intermedio calcule el camino para cada paquete generado o recibido. Este esquema es el utilizado por CGR y resulta el mas simple [59] dado que permite que cada nodo pueda determinar la ruta para el tráfico de acuerdo a la visión local de la topología (expresada en un plan de contacto). Sin embargo requiere de un considerable y valioso tiempo de procesamiento a bordo. Vale destacar que como un aporte secundario del trabajo doctoral aquí

presentado, se generó una alternativa interesante a esta visión denominada Cache-CGR (C-CGR) [9] en el que se mostró una significativa mejora al rendimiento de procesamiento de CGR. Describiremos en detalle este aporte en la sección 6.4.1 del capítulo 6.

b) Que el segmento origen calcule la ruta completa y la incluya como información de cabecera en el paquete para que los siguientes nodos en el camino utilicen esa información [113]. Este esquema, también conocido como Extension-Block CGR o (EB-CGR), permite un ahorro importante de procesamiento en los nodos intermedios, pero incurre en una sobrecarga de información en cada paquete (bundle).

La Tabla 4.1 resume los mecanismos de enrutamiento revisados y los clasifica según su flujo de trabajo.

Tabla 4.1: Mecanismos de Enrutamiento Existentes

	Cálculo en nodo centralizado y distribución de rutas	Cálculo en nodo origen y envío de rutas en cabecera de bundle	Cálculo en todos los nodos que reciben un bundle
Manual	X		
MFW	X		
CGR			X
EB-CGR		X	
C-CGR		X	X

4.2.1.2. Métricas de Enrutamiento

Independiente del mecanismo de implementación de las rutas, el cálculo de las mismas se debe dar con el objetivo de optimizar alguna métrica determinada. En general la métrica mas buscada en aplicaciones satelitales de baja órbita es la ruta que permita la entrega del tráfico en el menor tiempo posible (latencia o *best delivery time* en Inglés) [59]. A pesar de que esta es la métrica que utilizaremos en este trabajo, también se puede dar lugar a considerar optimizaciones de cantidad de contactos requeridos para implementar la misma (*contact utilization* en Inglés) la cual es tenida en cuenta para aplicaciones de DTN en el ámbito interplanetario o de espacio profundo (Deep Space en Inglés). Por último, también se podría considerar una métrica de *costo* si es que el uso de los mismos tiene algún impacto económico.

En general, y dado que nuestro campo de aplicación son las redes satelitales de baja órbita (LEO), adoptaremos un criterio de best delivery o mejor tiempo de entrega a

menos que se indique lo contrario. En efecto, y a pesar de que difieran en su implementación (Tabla 4.1), tanto los esquemas de enrutamiento de MFW o los basados en CGR permiten proporcionar rutas guiadas por esta métrica. Entre estos, adoptaremos el esquema MFW también denominado *enrutamiento de espacio y tiempo* (STR) para integrarlo en nuestro procedimiento de diseño de plan de contacto basado en rutas. La argumentación de esta elección es que STR, a diferencia de otros esquemas, calcula las rutas de todos los nodos contra todos los nodos de manera centraliza, lo cual resulta apropiado para su consideración en un centro de operación para el diseño de planes de contacto.

4.2.2. Enrutamiento de Espacio y Tiempo

El mecanismo Merugu's Floyd-Warshall (MFW), también conocido como enrutamiento de espacio y tiempo o *space-time* routing (STR) framework en Inglés, fue propuesto por Merugu en [61] para redes basadas en el paradigma *store, carry and forward*. Su nombre deriva de su cierta proximidad al conocido algoritmo de Floyd-Warshall para el cómputo del camino mas corto entre todos los pares de vértices de un grafo [114].

Dado que tanto Floyd-Warshall como CGR se basan internamente en el algoritmo del camino mas corto de Dijkstra [115], las rutas entregadas por MFW y CGR resultan análogas para los fines de aplicación en este capítulo. Como se verá en el capítulo 6, CGR [60] incluye mecanismos de mitigación de la congestión que puede hacer que sus rutas difieran de las MFW para casos de tráfico excesivo. Sin embargo, el uso que se le da a MFW en el diseño de planes de contactos basados en rutas (RACP) no contempla volúmenes de tráfico, por lo que no se genera congestión, temática que abordaremos de lleno en la discusión de implementación de planes de contactos en el capítulo 6 de esta tesis.

Como entrada, MFW toma un grafo del sistema basado en el modelado FSM descrito en la sección 2.3.2. A la salida, MFW entrega un conjunto de K matrices $[R]_{k,i,j}$ que especifica tanto el próximo salto que el nodo i debe considerar para llegar al destino final j en el estado k así como el tiempo esperado de entrega relativo de ese tráfico $d_{k,i,j}$. La Figura 4.1 ilustra estas matrices de rutas para dos planes de contactos ejemplo.

En esencia, MFW es una aplicación del algoritmo Floyd-Warhshall a un modelado del tipo FSM de las redes DTN. La originalidad del algoritmo de MFW radica en la creación de arcos *temporales* y *espaciales*. Los últimos son exactamente aquellos que representan una factibilidad de comunicación entre dos nodos del sistema (contacto) mientras que los arcos temporales son creaciones auxiliares que unen un mismo nodo en el estado pasado k_{a-1} y futuro k_{a+1}. En efecto, el grafo resultante es denominado *grafo en el espacio y*

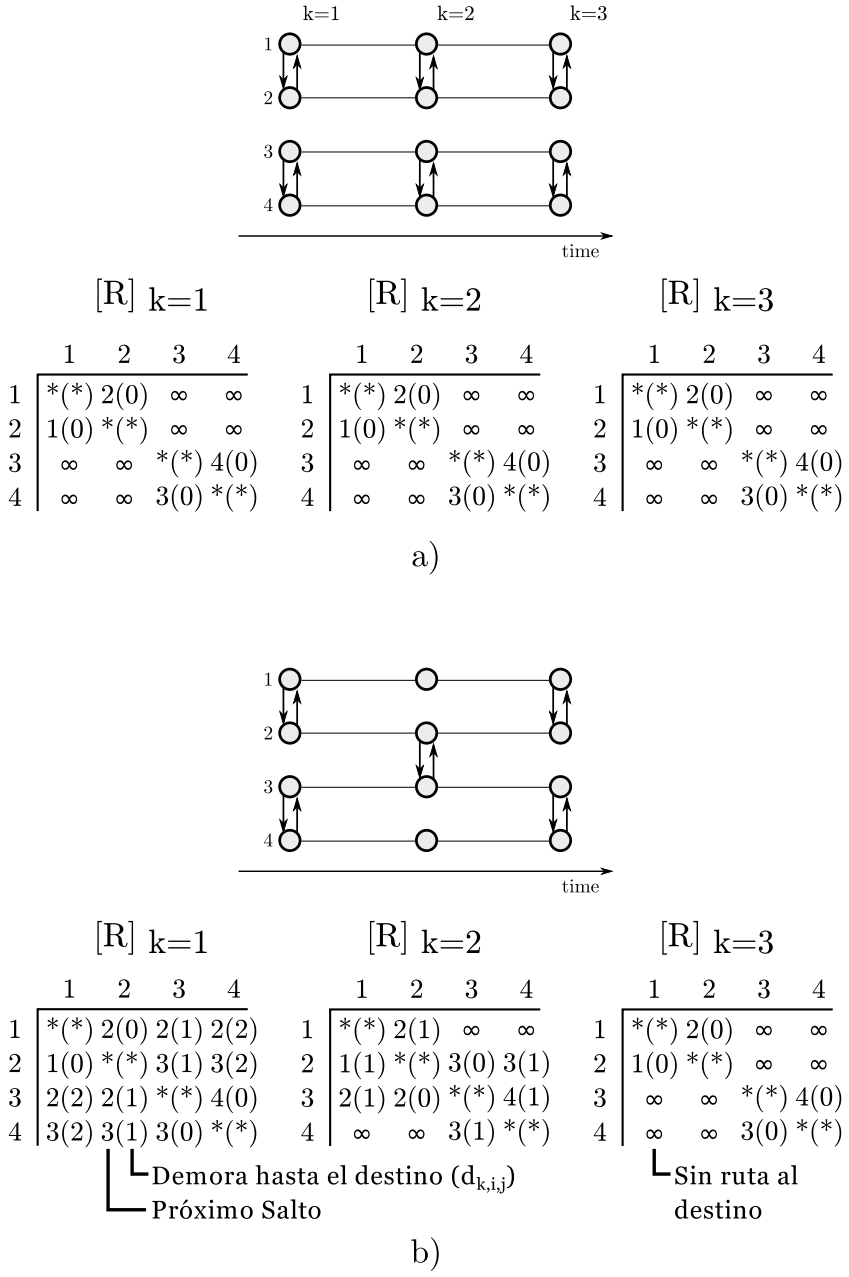


FIGURA 4.1: Matrices de rutas entregada por MFW para dos planes de contactos

tiempo o *space-time graph* en Inglés. Este grafo es luego sometido a una formulación especial de Floyd-Warshall que permite resolver los caminos mas cortos entre todos los nodos con una complejidad computacional de $O(N^3K)$ para un sistema con N nodos y K estados.

4.2.3. Formulación del Problema

En el diseño de planes de contacto basados en rutas o *Route-Aware Contact Plan* (RACP) en Inglés, planteamos como objetivo que el plan de contacto resultante cuente

con rutas factibles a todos los destinos (grafo conexo) por un lado, y que las mismas resulten de un tiempo medio de entrega (demora fuente-destino) mínimo. Esto permite a largo plazo diseñar un plan de contacto capaz de proporcionar un buen servicio a un patrón de tráfico general del tipo todos contra todo. En consecuencia denominaremos a este problema diseño de plan de contacto basado en rutas RACP. Además, al igual que en el capítulo 3, incluiremos como métrica a analizar y controlar el índice de justicia de Jain (Jain Index en Inglés) [107] de distribución de los arcos.

En general, la naturaleza de las métricas de factibilidad de rutas (*unrouted-time* o UT in Inglés), de mínimo tiempo medio de demora (*mean-delay* o MD en Inglés) e índice de justicia Jain (*Jain-index* o JI en Inglés), pueden resultar inconmensurables y por momentos contraproducentes al querer optimizarlas en conjunto. En efecto, el problema se transforma en un planteo de múltiples objetivos cuyo criterio de optimalidad fue tratado y definido para el caso general por Pareto [116] quien concluye que una solución óptima se forma por una frontera o superficie en la que el conjunto de variables (MD, UT, JI) resultan óptimas en un sentido mas amplio.

En consecuencia, planteamos el problema de RACP formalmente de la siguiente manera. Se busca la determinación de un plan de contactos representado en un vector solución $[L]$ que satisfaga las restricciones de recursos del sistema (discutidos en la sección 2.4) expresado de la forma:

$$L = [l_{1,1,1}, l_{1,1,2}, \quad ... \quad l_{k,i,j}]^T \quad \forall k, i, j \quad : \quad l_{k,i,j} \in \{0, 1\} \tag{4.1}$$

Donde i y j representan los nodos y k el estado del sistema. Además, el plan de contacto en $[L]$ deberá buscar optimizar la función objetivo F_{obj} que minimice la demora promedio de entrega de datos (MD), minimice la cantidad de tiempo que una ruta entre un nodo i y j es inexistente (UT), y que maximice el criterio de justicia de Jain (JI) como se plantea a continuación.

$$F_{obj} = [min : MD, min : UT, max : JI]^T \tag{4.2}$$

En general, ya se demostró en el capítulo 3 que un planteo óptimo del problema de justicia resulta intratable en términos computacionales. En consecuencia, considerar este planteo de RACP con mayores variables, al menos empeoraría esta condición ya que se incluye la necesidad de calcular las rutas entre nodos en un grafo de espacio-tiempo para redes DTN. Por ejemplo, inclusive para una topología simple de $k = 10$ estados y 10 nodos con una densidad de enlaces del 0,3 genera alrededor de 300 contactos por seleccionar

y procesar resultando en un total de 2^{300} planes de contactos sobre los cuales ejecutar, uno por uno, el algoritmo MFW.

Si bien la existencia de un algoritmo de asignación como Blossom [105] permitió la derivación de una alternativa eficiente como FCP [2] para el diseño de planes de contacto basado en justicia, la especificidad de la función objetivo planteada nos obliga a buscar alternativas de optimización genéricas y eficientes para atacar el problema de CPD con información de rutas o RACP. En efecto, en la siguiente sección 4.3 exploramos diferentes técnicas y estrategias metaheurísticas de búsqueda local y de recocido simulado para resolver el problema formulado en esta sección.

4.3. Planteo Algorítmico

En general, a pesar de que para escenarios lo suficientemente simples (tamaño acotado de nodos, arcos, y estados) podrían ser optimizados de manera óptima por medio de un planteo teórico similar al presentado en la sección 3.2, estos métodos fallan en entregar soluciones factibles para la generalidad de los escenarios en tiempos razonables. En consecuencia, existen métodos alternativos y aproximados (sub-óptimos) que vale la pena considerar. Entre estos, las metodologías metaheurísticas de trayectoria han probado ser de utilidad en la resolución de problemas altamente complejos como los aquí desarrollados [117].

Con el fin de tratar el problema del diseño de plan de contactos basado en el criterio de rutas RACP, proponemos una aproximación de dos etapas en la que inicialmente se determine un plan de contactos justo (utilizando el algoritmo FCP desarrollado en la sección 3.3.2) para luego iterar y mejorar las métricas de la función objetivo ($F_{obj} = [min : MD, min : UT, max : JI]^T$). En general, ya se mostró en la sección 3.4.4 que un plan de contacto justo diseñado con FCP resulta beneficioso para la rápida entrega de un patrón de tráfico todos contra todos. Sin embargo este fenómeno resulta de manera indirecta luego de buscar la justicia entre los contactos. En particular, si el esquema de enrutamiento es conocido, este plan de contactos puede mejorarse al mantener la métrica de justicia dentro de un rango razonable para luego estudiar las rutas resultantes en un espacio de soluciones dado.

En efecto, las iteraciones deberán ser guiadas por la evolución en conjunto de las métricas MD, UT, y JI con el fin de buscar una solución balanceada bajo diferentes criterios de búsqueda. La Figura 4.2 ilustra este estrategia donde luego de obtener una primera solución factible (e inicialmente optimizada) L', se utilizan y exploran diferentes metodologías de trayectoria conocidas como descenso empinado o *Steepest Descent* (SD),

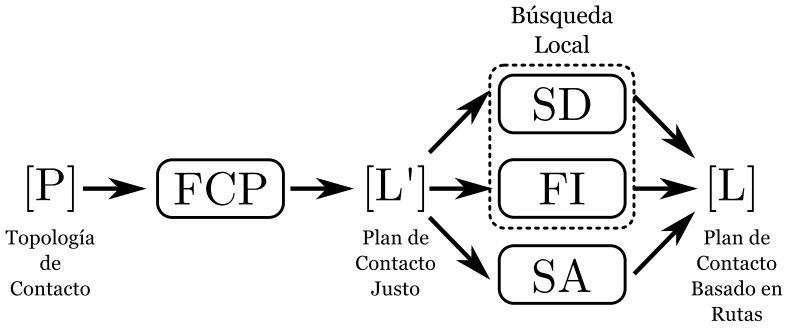


FIGURA 4.2: Flujo de diseño de contacto basado en rutas

primera mejora o *First Improvement* (FI) y recocido simulado o *Simulated Annealing* (SA) para entregar un plan de contacto final L mejorado en términos de las rutas múltiple salto entre los satélites que conforman la red. A continuación detallaremos estas metodologías para finalmente comparar su rendimiento y concluir con una propuesta concreta para el diseño de plan de contactos basado en rutas que denominaremos RACP.

4.3.1. Algoritmo de Primera Mejora

El algoritmo de primera mejora o *first improvement* (FI) en Inglés se basa en una estrategia agresiva de exploración en la que partiendo de una solución inicial, se explora el vecindario de solución de manera tal que el primer vecino que muestra una mejora en las métricas evaluadas es adoptado como la nueva solución del problema para la continuación de la búsqueda. Este comportamiento se ilustra en la Figura 4.3 a) y tiene la ventaja de ser simple y eficiente en términos de implementación pero peca de caer rápidamente en óptimos locales por su escasa capacidad de exploración. Entendemos a la exploración como la capacidad de la búsqueda de aceptar soluciones potencialmente malas con el fin de explorar vecindarios lejanos al local en búsqueda de un óptimo global [117] (en el caso de la Figura 4.3 el óptimo global es el valor 6 ubicado en la parte superior del espacio de solución).

El algoritmo de FI se detalla en el algoritmo 2 donde el esquema parte de un plan de contacto inicial $[L]$ obtenido de aplicar FCP a la topología de contacto $[P]$ en la linea 2. Luego, sobre este plan de contacto, se calculan las rutas de menor tiempo de entrega (matriz $[R]_{k,i,j}$) por medio de la aplicación de MFW explicado en la sección 4.2.2 en la linea 3, para finalmente obtener las métricas de la matriz de rutas en la línea 5. Las métricas se almacenan en BMD (mejor tiempo de entrega o *best-mean-delivery*), BUT (mejor tiempo sin ruta o *best-unrouted time*) y BJI (mejor indice Jain o *best-jain-index*). Una vez obtenida y evaluada la primera posible solución, se da inicio a la búsqueda metaheurística a partir de la línea 7.

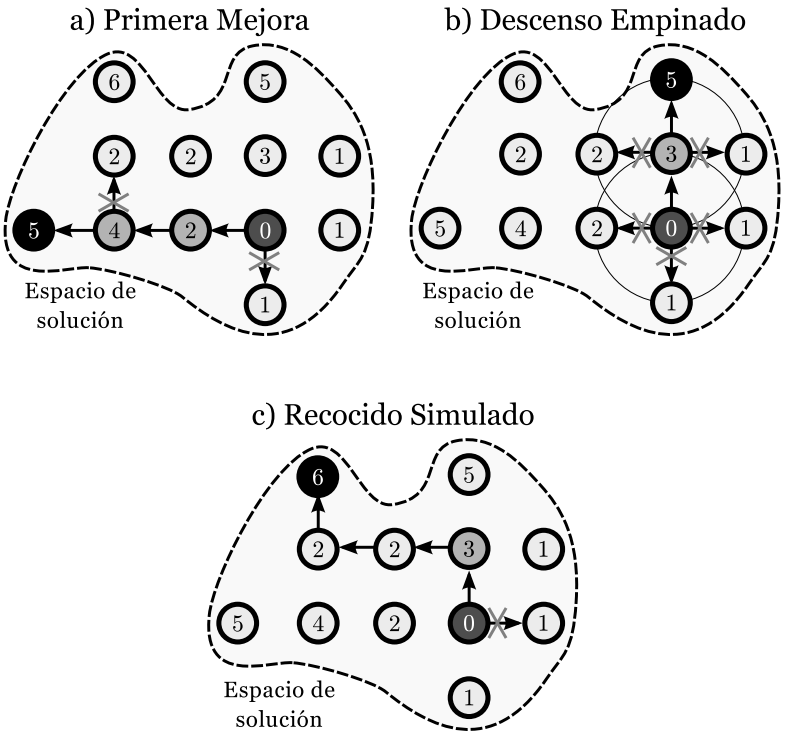


FIGURA 4.3: Estrategias de primera mejora, descenso empinado, y recocido simulado

En la línea 8 se aplica un operador al plan de contacto $[L]$ con el fin de obtener un primer vecino $[L]'$ del mismo. El operador utilizado para determinación del vecindario se detalla en la próxima sección 4.3.1.1. Luego, se aplica nuevamente el algoritmo MFW para calcular las rutas de este nuevo plan de contacto $[L]'$ en la linea 9 para luego determinar las métricas asociadas a $[L]'$ en la linea 11. De manera análoga a BMD, BUT, y BJI, se determinan CMD (tiempo de entrega actual o *current-mean-delivery*), CUT (tiempo sin ruta actual o *current-unrouted time*) y CJI (indice Jain actual o *current-jain-index*). En efecto, en la linea 13 se determina si estas métricas son mejoras que las obtenidas ya sea en la solución inicial (linea 5) o en alguna iteración anterior. En caso de que $[L]'$ resulte con mejores estadísticas que $[L]$, este se adopta como la mejor solución hasta el momento y se actualizan las mejores métricas (BMD, BUT, y BJI) en las lineas 15 y 16 respectivamente.

Finalmente, se obtiene un esquema que permite mejorar iterativamente un vector de soluciones de acuerdo al criterio de optimalidad de Pareto [118] a medida que avanza el tiempo de búsqueda. El algoritmo retorna entonces cuando el número de iteraciones llega a $MaxIterations$.

Algoritmo 2: Algoritmo de primera mejora (FI)

input : Topología de Contacto $[P]$ de tamaño $K \times N \times N$ y Tiempos de Estado $[T]_k$
output: Plan de Contact $[L]$ de tamaño $K \times N \times N$

```
1  Resolver plan de contacto inicial con FCP
2  [L] ⟵ FCP([P],[T]);
3  [R] ⟵ FloydWarshall([L]);
4  Calcular rutas y sus métricas
5  [BMD, BUT, BJI] ⟵ [MaxD([R]), UnRT([R]), JainI([L])];
6  Búsqueda de Primera Mejora
7  for i ← 0 to MaxIterations do
8      [L]' ⟵ GetNeighbor([L]);
9      [R]' ⟵ FloydWarshall([L]');
10     Calculo de nuevas rutas y sus métricas
11     [CMD, CUT, CJI] ⟵ [MaxD([R]'), UnRT([R]'), JainI([L]')];
12     Comparación de métricas
13     if (CMD <= BMD)&(CUT <= BUT)&(CJI >= BJI) then
14         Adopción de nueva topología
15         [L] ⟵ [L]';
16         [BMD, BUT, BJI] ⟵ [CMD, CUT, CJI];
```

4.3.1.1. Operador de Vecindario

En general, en los algoritmos de búsqueda metaheurística, la búsqueda de vecinos se basa en operadores sencillos que simplemente se desplazan una cierta distancia en el espacio de soluciones [117]. Sin embargo, nuestro espacio de soluciones está formado por planes de contactos cuya estructura de por si es compleja (ver modelado en sección 2.3). Lo que aún empeora mas esta dificultad de exploración del espacio de soluciones es que el vecino que un operador pueda obtener debe satisfacer las restricciones de recursos planteadas originalmente. Es decir, el espacio de soluciones que debemos explorar se compone solamente de aquellos planes de contacto que satisfacen las restricciones.

En efecto, el operador que genere potenciales vecinos a analizar debe garantizar dos cosas: a) poder explorar la totalidad del espacio de soluciones definido, y b) garantizar que las soluciones que devuelve satisfagan las restricciones de recursos originales ($[C]_i = 1 \quad \forall i$). En efecto, el algoritmo FCP utilizado inicialmente satisface estas dos características que deben sostenerse a lo largo de la búsqueda del algoritmo. En caso de tener una restricción de $[C]_i \neq 1 \quad \forall i$, se deberá prescindir de FCP para incorporar otro mecanismo mas genérico como Fair_LP que pueda solucionar la topología. Con este fin, definimos un operador de generador de vecinos de $[L]$ ($GetNeighbor([L])$) que cada vez que es llamado, el mismo se comporta de la siguiente manera:

1. Obtener el plan de contacto original $[L]$.
2. Generar un mapa $ResourceMap[k][i]$ de longitud $MapMax$ de nodos i en estados k que estructuren aquellos momentos y segmentos que impliquen una decisión de arcos o contactos.
3. Generar un número aleatorio entre 0 y $MapMax$ para apuntar a alguna de los puntos críticos de decisión.
4. Provocar un cambio de la decisión en $ResourceMap[k][i]$ (desactivar un arco y activar otro) respetando la bidireccionalidad de los arcos. Es decir, si desactivo el arco 0-1, deberé desactivar su correspondiente arco opuesto 1-0.
5. Revisar la consistencia de la solución generada. Puede suceder que un cambio de arco genere otra inconsistencia de recursos en otro nodo vecino directamente vinculado a i. Si es el caso se deberá reparar esta situación desactivando también el arco vecino en conflicto.
6. Devolver el plan de contacto vecino generado $[L]'$

En resumen, el operador como se plantea, busca cambiar una elección tomada ante la restricción de interfaces o puertos con el fin de generar un plan de contacto alternativo para su evaluación, que siga satisfaciendo las restricciones originales. Esta técnica de verificación de restricciones se conoce como *garantía de restricciones por reparación* en la bibliografía [117].

4.3.2. Algoritmo de Descenso Empinado

A pesar de que el algoritmo de primera mejora resulta simple y eficiente de implementar, el mismo puede resultar poco inteligente al aceptar y adoptar la primera mejor solución encontrada como guía absoluta de búsqueda sin considerar un vecindario de mayor tamaño. En efecto, extender la búsqueda de esta manera antes de decidir reemplazar el mejor plan de contacto encontrado hasta el momento, permitiría no apresurase en la toma de decisiones, pero requeriría de una mayor memoria al tener que almacenar el conjunto completo de soluciones obtenido del vecindario.

En consecuencia, como se muestra en la Figura 4.3 b), una estrategia de descenso empinado o *steepest descent* (SD) en Inglés permiten una evaluación mas eficiente del espacio de solución local. En este ejemplo ilustrativo, el esquema de primera mejora (FI) rápidamente decide avanzar la exploración por una solución con métrica de 2, cuando en realidad hay otro vecino con métrica de 3 que no llega a ser evaluado. Por otro lado, un

algoritmo de SD contempla un conjunto predefinido ($MaxNeighbors$) de vecinos a los que el operador (definido en la sección 4.3.1.1) permite llegar permitiendo avanzar en un camino de soluciones posiblemente mas óptimas.

El comportamiento detallado de la estrategia de descenso empinado se muestra en el Algoritmo 3 el cual se comporta de la misma manera que el de primera mejora en las primeras líneas 2 y 3 al generar una solución inicial basada en FCP [2]. Luego se almacenan las métricas de BMD (mejor tiempo de entrega o *best-mean-delivery*), BUT (mejor tiempo sin ruta o *best-unrouted*) y BJI (mejor indice Jain o *best-jain-index*) para esta solución. Finalmente, se da comienzo a la búsqueda de steepest descent [117] en la línea 7.

A diferencia que en FI, este esquema genera una vecindad $[L]'_n$ de planes de contactos de tamaño $MaxNeighbors$ en la línea 9 y 10. Una vez completada esta lista de posibles

Algoritmo 3: Algoritmo de descenso empinado (SD)

input : Topología de contacto $[P]$ de tamaño $K \times N \times N$ y tiempo de estados $[T]_k$
output: Plan de contacto $[L]$ de tamaño $K \times N \times N$

```
1  Resolver la topología inicial con FCP
2  [L] ⟵ FCP([P],[T]);
3  [R] ⟵ FloydWarshall([L]);
4  Calcular rutas y sus métricas
5  [BMD, BUT, BJI] ⟵ [MaxD([R]), UnRT([R]), JainI([L])];
6  Búsqueda de descenso empinado
7  for i ← 0 to MaxIterations do
8      Generar grupo de vecinos
9      for n ← 0 to MaxNeighbors do
10         [L]'_n ⟵ GetNeighbor([L]);
11     Adoptar mejor vecino
12     BN ⟵ −1;
13     for n ← 0 to MaxNeighbors do
14         Calcular nueva rutas y sus métricas
15         [R]'_n ⟵ FloydWarshall([L]'_n);
16         [CMD, CUT, CJI] ⟵ [MaxD([R]'_n), UnRT([R]'_n), JainI([L]'_n)];
17         Comparación de métricas
18         if (CAD <= BAD)&(CUT <= BUT)&(CJI >= BJI) then
19             Adopción de la mejor solución
20             BN ⟵ n;
21             [L] ⟵ [L]'_n;
22             [BMD, BUT, BJI] ⟵ [CMD, CUT, CJI];
23     Retornar si no se encuentran mejores vecinos
24     if NoBestNeighbor then
25         exit
```

soluciones, se inicia la evaluación de cada una de las mismas con el fin de determinar la mejor de ellas. En efecto una bandera BN se inicializa en -1 para indicar finalmente si se encontró un mejor vecino en cuyo caso afirmativo devuelva un índice al mismo. La evaluación comienza en la línea 13 y repite el mismo esquema que se mostró para el algoritmo de primera mejora en la sección 4.3.1 que calcular las rutas con MFW y obtener las métricas temporales en CMD, CUT, y CJI. Paso a paso, se comparan estas últimas en la línea 18 con las mejores conocidas hasta el momento y en caso de mejorarlas se actualiza el mejor vecino conocido ($[L]_{k,i,j}$) en las líneas 19 a 22. Si en la totalidad de la evaluación de $MaxNeighbors$ no se encontró ningún mejor vecino el algoritmo termina y retorna al control principal.

En general, tanto el algoritmo de primera mejora como el de descenso empinado apuntan a mejorar una topología inicial en función de parámetros de rutas para redes DTN, ambos se detienen en las primeras mejores soluciones encontradas en su camino de búsqueda. Este comportamiento es ampliamente conocido y estudiado en la literatura de metahuerísticas [117] quienes indican que se deben tomar consideraciones especiales para evitar caer en óptimos locales. De no lograr esto se reduce inevitablemente el espacio de búsqueda disminuyendo la calidad del procedimiento de búsqueda desarrollado. En consecuencia, en la siguiente sección 4.3.3 exploramos una tercer y última estrategia llamada recocido simulado cuyo objetivo es solucionar específicamente estos problemas.

4.3.3. Algoritmo de Recocido Simulado

En general, cuando la solución óptima se encuentra en la vecindad del la solución entregada por FCP, los esquemas de FI y SD evidencian comportamientos eficientes y de utilidad para el diseño de planes de contacto basados en rutas. Sin embargo, cuando un óptimo global se ubica a mas de dos aplicaciones del operador de vecino (sección 4.3.1.1) se requieren de estrategias con mejores capacidades de exploración como la de recocido simulado o *simulated annealing* (SA) [119, 120] en Inglés, una popular técnica metaheurística sobre todo para espacio de soluciones discretos como el aquí tratado.

En general, el algoritmo de SA tiene la particularidad de aceptar soluciones de menor calidad que las ya conocidas con el fin o esperanza de en el futuro poder explorar zonas del espacio de soluciones que de otra manera no se hubiese podido llegar. Es decir, en SA se acepta la aplicación del operador en soluciones sub-óptimas de manera controlada para poder detectar posibles óptimos en áreas alejadas de el inicio de la exploración. Como se puede observar en la figura 4.3, el algoritmo de recocido simulado luego de adoptar la mejor solución en 3, decide aceptar aquella de métrica 2 (inferior) con el fin de extender el rango de exploración para finalmente tener la suerte de encontrar un

óptimo global de métrica 6. Si bien claramente este no es el caso general, es válido como explicación e ilustración del mecanismo de SA.

Sin embargo, esta capacidad de exploración debe realizarse de manera controlada para que a medida que avanzan las iteraciones el algoritmo tienda a converger en soluciones aceptables. En efecto, el nombre de SA deriva de una analogía de la industria metalúrgica donde un metal es enfriado suavemente de manera que la entropía interna se acomode y se fortalezca [119]. Bajo esta filosofía, el recocido simulado lleva la cuenta de una temperatura $[T]_i$ que disminuye a lo largo de las iteraciones y que al mismo tiempo esta la incide en la probabilidad de aceptar una solución de menor calidad o no. En consecuencia este algoritmo tiene altas probabilidades de explorar al comienzo (escapar del óptimo local) para finalmente enfocar el esfuerzo de búsqueda en obtener el valor mas cercano al óptimo.

El Algoritmo 4 detalla el comportamiento del recocido simulado implementado para el diseño de plan de contactos basados en rutas. Nuevamente en las líneas 1 a 4 se determina la solución inicial y sus métricas para luego ingresar en el lazo de búsqueda en la línea 6. Aquí, se genera un vecino $[L']$ de acuerdo al operador $GetNeighbor([L])$ y se evalúan sus rutas en la línea 8 ($FloydWarshall([L'])$) y sus métricas derivadas. Luego, si esta nueva solución es efectivamente mejor que la conocida anteriormente se ejecuta el proceso de reemplazo del nuevo plan de contacto en las lineas 13 y 15. La originalidad de SA viene a continuación: en la línea 17 la solución no-óptima se somete a una prueba probabilística en la que en función de la temperatura T_i y las distancias al óptimo del valor obtenido (ΔMD, ΔUT, ΔJI), la misma puede ser elegida o no para búsquedas futuras. En este punto la elección está guiada por la función de aceptación $A(T_i, \Delta_i) \quad : i = JI, MD, UT$ detallada en la ecuación 4.3.

$$A() = exp(-\frac{\Delta MD}{T_{MD}}) * exp(-\frac{\Delta UT}{T_{UT}}) * exp(\frac{\Delta JI}{T_{JI}}) \tag{4.3}$$

En efecto, en caso de que la función apruebe la elección, en las lineas 19 y 20 se adopta la solución (al igual que en las lineas 14 y 15). En su defecto, se restaura la mejor solución previamente conocida como se detalla a continuación.

4.3.3.1. Estrategias de Recocido y Retorno a Base

A lo largo de las iteraciones la temperatura T_i decrementa a medida que se van eligiendo vecinos de menor calidad que el mejor conocido hasta el momento a medida que se avanza en la exploración. En consecuencia, al inicio del esquema cabe la posibilidad

de aceptar una seguidilla de numerosas soluciones sub-óptimas mientras que al final solamente se adoptan aquellas que realmente mejoren la mejor solución conocida. El efecto de la disminución de la temperatura es conocido como *annealing schedule* [121] en la literatura y en esta caso es implementado de manera lineal para cada temperatura individual $[T] = [T_{MD}, T_{UT}, T_{JI}]$ como lo sugieren Suppapitnarm y Parks en [122].

Por otro lado, se adopta la estrategia de retorno a base o *return-to-base strategy* [123] dado que en caso de no aceptar una nueva solución se rescata la mejor conocida hasta el momento (linea 22 y 23). Por último, y en general, en [123] también se asegura que el esquema SA es capaz de alcanzar la frontera de Pareto siempre que se consiga el número adecuado de iteraciones. De esta manera, y finalmente, al igual que FI y SD, el algoritmo SA retorna el mejor plan de contacto encontrado en la búsqueda en la linea 25.

Algoritmo 4: Algoritmo de recocido simulado (SA)

input : Topología de Contacto $[P]$ de tamaño $K \times N \times N$ y tiempo de estado $[T]_k$
output: Plan de Contacto $[L]$ de tamaño $K \times N \times N$

1 *Solución inicial con FCP*
2 $[L] \longleftarrow$ `FCP(`$[P]$`,`$[T]$`)`;
3 $[R] \longleftarrow$ `FloydWarshall(`$[L]$`)`;
4 $[BMD, BUT, BJI] \longleftarrow$ [`MaxD(`$[R]$`)`, `UnRT(`$[R]$`)`, `JainI(`$[L]$`)`];
5 *Búsqueda de recocido simulado*
6 **for** $i \leftarrow 0$ **to** $MaxIterations$ **do**
7 $[L]' \longleftarrow$ `GetNeighbor(`$[L]$`)`;
8 $[R]' \longleftarrow$ `FloydWarshall(`$[L]'$`)`;
9 *Cálculo de nueva ruta y sus respectivas métricas*
10 $[CMD, CUT, CJI] \longleftarrow$ [`MaxD(`$[R]'$`)`, `UnRT(`$[R]'$`)`, `JainI(`$[L]'$`)`];
11 *Comparación de métricas*
12 **if** $(CMD <= BMD)\&(CUT <= BUT)\&(CJI >= BJI)$ **then**
13 *Adopción de la nueva topología*
14 $[L]_{best} \longleftarrow [L] \longleftarrow [L]'$;
15 $[BMD, BUT, BJI] \longleftarrow [CMD, CUT, CJI]$;
16 **else**
17 **if** $exp(-\frac{\Delta MD}{T_{MD}}) * exp(-\frac{\Delta UT}{T_{UT}}) * exp(\frac{\Delta JI}{T_{JI}}) > rand(0,1)$ **then**
18 *Adopcion de topología sub-óptima*
19 $[L] \longleftarrow [L]'$;
20 $[T]_i \longleftarrow [T]_i - 1 \quad : i = JI, MD, UT$
21 **else**
22 *Restauración de mejor topología*
23 $[L] \longleftarrow [L]_{best}$;
24 *Retorna la mejor topología encontrada*
25 $[L] \longleftarrow [L]_{best}$;

Tabla 4.2: Métricas de análisis de plan de contacto basados en rutas

	Descripción	Objetivo
Mean Delivery (MD)	Tiempo medio de entrega de tráfico entre todos los estados k para todos los posibles nodos destinos.	Minimizarlo
UnroutedT (UT)	Cantidad de tiempo acumulado que se desconoce la ruta para un nodo destino determinado.	Minimizarlo
Jain Index (JI)	Índice de justicia que indica la equidad con la que son distribuidos los arcos en el plan de contacto.	Maximizarlo

4.4. Análisis de Plan de Contacto Basado en Rutas

En esta sección se realiza un análisis comparativo de los diferentes esquemas de diseño de plan de contactos basado en rutas tratados en esta sección: First Improvement (FI) tratado en la sección 4.3.1, Steeptest Descent (SD) en la sección 4.3.2, y Simulated Annealing (SA) en la sección 4.3.3 para finalmente concluir con un candidato final para el problema de RACP planteado en este capítulo.

4.4.1. Métricas de Evaluación

Las métricas de evaluación que usaremos son aquellas que esencialmente forman la función objetivo planteada en la ecuación 4.2. En particular estudiaremos el comportamiento independiente de sus tres dimensiones ($AvgDelay$, $UnroutedT$ y $JainIndex$) detalladas en la Tabla 4.2.

4.4.2. Análisis Sobre Topología de Contactos Aleatorias

Con el fin de evaluar los algoritmos propuestos, los analizaremos sobre un conjunto de topología de contactos $[P]_{k,i,j}$ generadas de manera aleatoria, para finalmente evaluar los planes de contactos $[L]_{k,i,j}$ diseñados de acuerdo a las métricas de la Tabla 4.2. Al igual que en los análisis del capítulo 3, las topologías aleatorias son generadas variando la densidad de existencia de enlaces entre nodos (*link density* en Inglés o LD). En efecto, a mayor LD, la mayor población de contactos en el sistema, y mayor el tamaño del espacio de búsqueda.

En general asumiremos valores de LD desde 0,08 a 0,20 en pasos de 0,01 para topología de contactos con $N = 6$ nodos y $K = 10$ estados de duración aleatoria entre $10 < t[k] < 20 \quad \forall k$. Para cada paso se generan un total de 500 topologías de contactos cuyos planes de contactos correspondientes son evaluados y sus resultados promediados. Por otro lado, el algoritmo de SA se inicializa con 3000 iteraciones de búsqueda con una temperatura inicial de $T_i = 2000 \quad \forall i$, mientras que los esquemas de FI y SD se configuran para terminar cuando ya no puedan encontrar una mejor solución.

La Figura 4.4 muestra las curvas de resultados obtenidas para las configuraciones descritas. Además de las métricas de los algoritmos FI, SD y SA, se incorpora el valor de las mismas para la topología de contacto sin diseñar bajo el nombre de PHY (de *physical* en Inglés). Efectivamente estas asumen que el sistema no cuenta con restricciones de interfaces ($[I]_i = 1 \quad \forall i$) por lo que sus rendimientos resultan notablemente superior. Sin embargo, es válida su consideración como cotas superiores de comportamiento.

De los resultados mostrados, se puede verificar que la aplicación de restricciones de

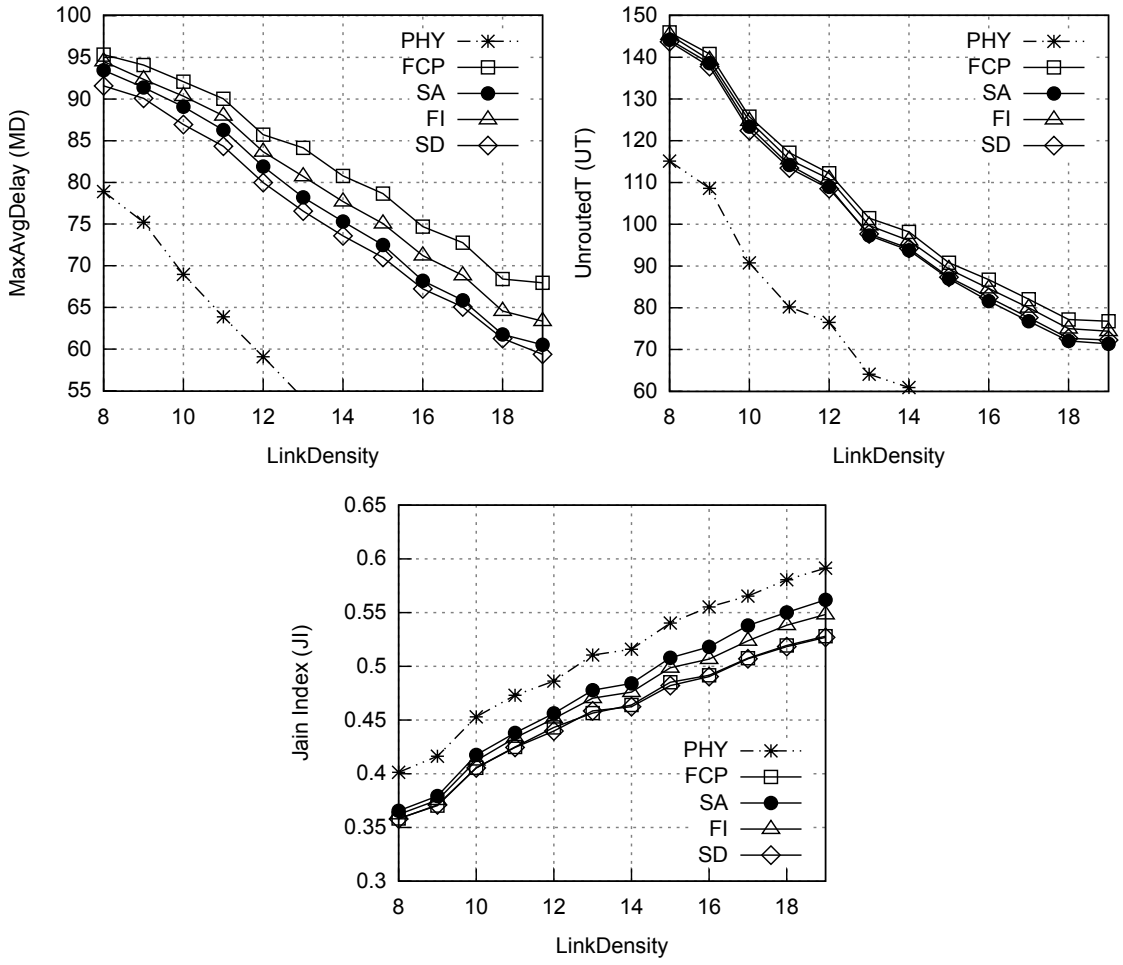


FIGURA 4.4: Evaluación de planes de contactos generados por los algoritmos de diseño con criterio de rutas

interfaces implica una pérdida de rendimiento considerable. En general, el fenómeno es mas notable para mayores densidades de enlaces (LD) dado que un mayor número de contactos (arcos) deben ser desactivados para cumplir con las restricciones de $[I]_i$. Por otro lado, es evidente que a pesar de la eficiencia de FCP demostrada en el capítulo 3, y discutida en [2], los algoritmos propuestos (FI, SD y SA) cumplen su función de mejorar sus métricas acercando sus curvas hacia los espacios de rendimientos mostrados por la topología de contacto.

Entre los esquemas propuestos, el de recocido simulado muestra un comportamiento superior a los esquemas de búsqueda local al ser el único capaz de minimizar la métrica de tiempo promedio de entrega (MD) sin comprometer el parámetro de justicia (JI). En efecto, esto confirma la hipótesis de que la capacidad de exploración con la que cuenta SA permite hacer una mejor evaluación general del espacio de soluciones con potenciales beneficios de los que carecen esquemas mas locales como FI y SD.

Sin embargo, si bien el algoritmo de recocido simulado entrega mejores planes de contactos, los mecanismos de descenso empinado y primera mejora también son capaces de ofrecer una mejora interesante y valiosa a los planes de contactos entregados por FCP. Mas importantemente, logran esto con un mínimo de esfuerzo computacional en comparación con las 3000 iteraciones que realiza el método de SA. En general, y de acuerdo a [117], los esquemas de recocido simulado se basan en decisiones probabilísticas mientras que FI y SD son totalmente determinísticos (es decir, avanzan hasta no obtener mejoras y terminan), por lo que los segundos tienen cotas de cálculo acotadas de mejor comportamiento en términos de procesamiento.

Por último, si bien analizamos las métricas en términos generales, resulta complejo determinar con certeza cual es la *mejor* tripla (conjunto de tres componentes) de MD, UT, JI entre los aquí obtenidos. En efecto, este es una característica estudiada por [116] quien propone un criterio formal de optimalidad basado en una frontera que comprenda el conjunto de todas aquellas soluciones consideradas mejores en algún aspecto. En consecuencia, analizaremos este comportamiento en un nuevo caso de estudio en la próxima sección 4.4.3.

4.4.3. Caso de Referencia y Estudio C: Topología en Tren

Con el fin de generar un análisis mas preciso del comportamiento de las métricas de los algoritmos FI, SD, y SA, en esta sección proponemos un caso de estudio puntual basado en un topología realista. En efecto, esta propuesta es el tercer caso de estudio (caso de estudio C) de redes satelitales que se incorpora a los ya propuestos en las secciones 2.3.1 y 3.4.3 (casos de estudio A y B respectivamente). Sobre este caso particular de topología

"en tren", buscaremos generar la frontera de soluciones factibles de Pareto para realizar una comparación formal.

En general, diferentes argumentos sustentan la propuesta de constelación lineal perpendicular al ecuador en forma de tren en órbita LEO como se ilustra en la Figura 4.5. Entre ellos, el hecho de que los objetos orbitantes dispuestos de esta manera (a una distancia lo suficientemente acotada) perciben prácticamente las mismas perturbaciones gravitatorias lo que permite un ahorro significativo de propelente al minimizar las maniobras de manutención de órbita (station keeping en Inglés). Por otro lado, esta disposición es conveniente desde una perspectiva del lanzador quien no requiere de complejas maniobra de cambio de plano orbital para desplegar la constelación (suponiendo que todos los segmentos se lanzan en el mismo vector). Esta condición no se cumple en los casos de topología en escalera y lineal ecuatorial discutidas en las secciones 2.3.1 y 3.4.3.

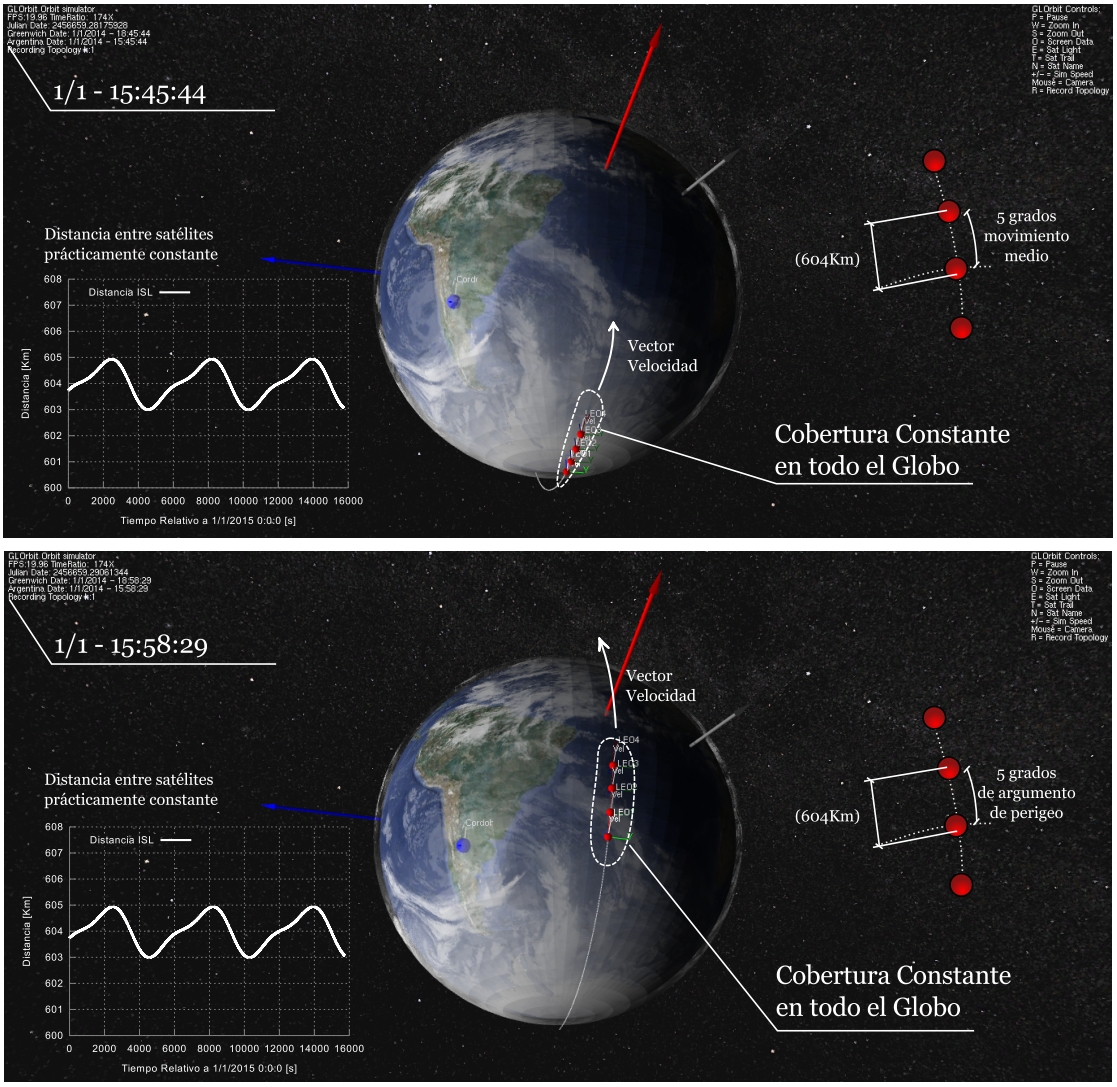


FIGURA 4.5: Representación del Caso de Estudio C en GLOrbit

Tabla 4.3: Tiempos y Parámetros Orbitales del Caso de Estudio de Topología en Tren

Inicio del Intervalo de Topología	Ene-1st, 2014, 0hs 0min 0sec
Fin del Intervalo de Topología	Ene-1st, 2014, 2hs 13min 20sec
Coeficiente Bstar (/ER)	0
Inclinación (grados)	90°
RAAN (grados)	359,9991°
Eccentricidad	0
Argumento del Perigeo (deg)	0°, 5°, 10° y 15°
Anomalía Media (deg)	0°
Movimiento Medio (rev/day)	15,0756 rev/day
Altura sobre el nivel del Mar (Km)	600 Km

Estas razones llevaron a la creación de constelaciones existentes en la actualidad como el A-Train de NASA [21], el cual, a pesar de no contar con enlaces inter-satelitales (ISL), resulta de inspiración para el diseño de una topología de redes DTN. Cabe destacar que si bien esta configuración permite una conexión del tipo permanente entre los nodos (no hay disrupciones en los ISL), la consideración de DTN permite hacer un mejor uso de los recursos al permitir apagar los equipos de comunicaciones en momentos claves, así como poder disponer de arquitecturas mas simples como las discutidas en la sección 2.4.2.

En particular, proponemos una constelación de 4 satélites en tren con los parámetros detallados en la Tabla 4.3. El tiempo de propagación que se toma es de un total de 8000 segundos (2 horas 13 minutos 20 segundos), sobre el cual aplicaremos un fraccionamiento de estados cada 1000 lo que deriva en una topología de contacto de $K = 8$ estados. Por otro lado, una variación de 5° en el argumento de perigeo para cada satélite nos permite distanciarlos aproximadamente unos 604 Km entre ellos. Como se ilustra en la Figura 4.5, esta distancia varía de manera despreciable (entre 603 y 605 Km) a lo largo de la órbita. Finalmente, la formación lineal en tren se obtiene al proporcionar a todos los satélites con el mismo ángulo RAAN en este caso de 359,9991°.

Una vez descrito el caso de estudio, procedemos a analizar los resultados de la aplicación de los esquemas de diseño de plan de contactos basados en rutas tratados en este capítulo. La Figura 4.6 a) muestra la topología de contacto inicial fraccionada antes de ser sometida a los esquemas de diseño. En efecto, las ilustraciones en la Figura 4.6 b) muestran el plan de contacto obtenido por la técnica de recocido simulado, mientras que c) ilustra aquellos resultados generados para este caso particular por FCP, el algoritmo de primera mejora, y de descenso empinado. De las mismas se puede concluir que ni FI ni SD pudieron mejorar la solución inicialmente propuesta por FCP, entregando exactamente las mismas métricas que la metodología de diseño basado en topología (MD : 2700, UT : 2000, JI : 0,358). Por otro lado, SA evidencia un conjunto de métricas superior (MD : 2500, UT : 1200, JI : 0,375) en todos los componentes, entregando

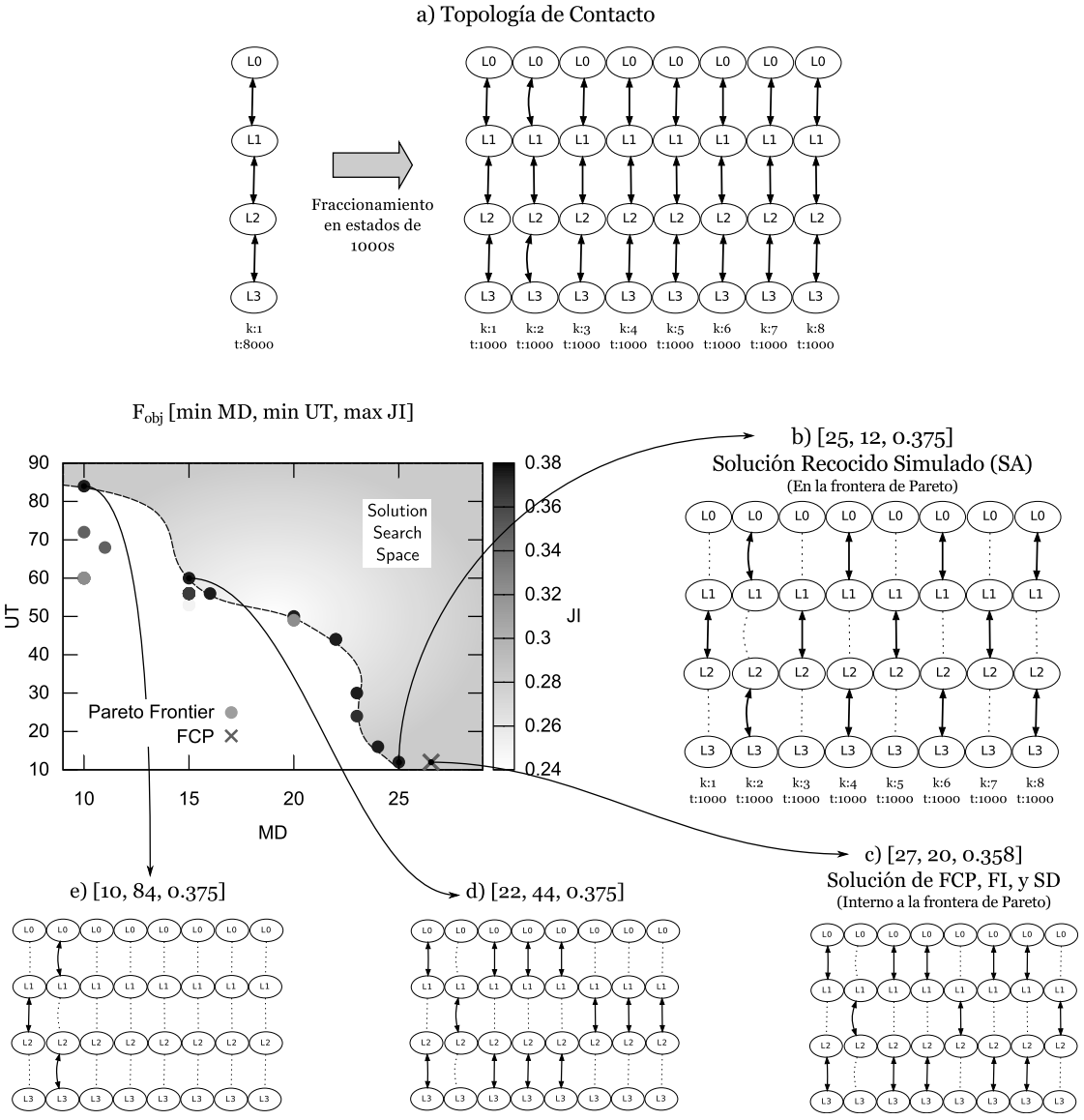


FIGURA 4.6: Evaluación del plan de contacto basado en rutas para el caso de estudio C

un plan de contacto cuyas características lo ubican en la frontera de Pareto [116]. De acuerdo a la terminología de la bibliografía, la solución c) se denomina *dominada*, y la solución b) (en la frontera) *dominante*.

En la misma Figura 4.6 d) y e) se muestran otros posibles planes de contactos con sus respectivas métricas que también estarían dentro del espacio de búsqueda de soluciones. En particular, d) ofrece el mínimo valor de MD (22) obtenible para un UT de 44 y JI de 0,375, mientras que e) el mínimo valor de MD (10) obtenible para un UT de 84 y JI de 0,375. Dado que estos conjuntos de soluciones son óptimos en el sentido de optimalidad de Pareto [116], se ubican en la frontera del espacio de solución mostrado en la gráfica. Sin embargo resulta interesante discutir el grado de utilidad de los mismos, particularmente del caso e), donde en el afán de minimizar al máximo la variable de demora promedio

de entrega, se penaliza drásticamente el tiempo sin ruta. En consecuencia, ese plan de contacto si bien es óptimo a nivel pareto (dominante), resulta de escasa utilidad para su aplicación en un sistema de redes satelitales DTN.

En general, de acuerdo a [116], es fundamental el desarrollo de tomadores de decisión (*decision makers* o DM en Inglés) que permitan no sólo obtener soluciones óptimas en término de dominancia, si no que también involucren inteligencias suficientes para no caer en casos como los ilustrados en las Figuras 4.6 d) y e). En efecto, los esquemas FI, SD, y SA mostrados en los algoritmos 2, 3, y 4 cuentan con esta capacidad de DM al iniciar su búsqueda con una topología de FSM y evitando que ninguna de las 3 métricas del plan de contacto diseñado sea menor a las de esta topología inicial.

4.5. Comentarios Finales Sobre el CPD Basado en Rutas

En este capítulo planteamos el problema de diseño de planes de contactos basados en rutas (RACP) y exploramos algunas alternativas metaheurísticas para su solución. Estos procedimientos se diseñaron con el objeto de guiar el diseño hacia planes de contactos cuyo promedio de demora en las rutas (caminos de múltiples saltos) contra los vecinos sea optimizado. De esta manera se obtienen planificaciones que permiten mejorar la utilización de los recursos siempre escasos en este tipos de redes satelitales. Por otro lado, si bien nos hemos enfocado en restricciones de interfaces simples ($[I]_i = 1 \quad \forall i$, el caso mas común), los procedimientos aquí descritos resultan trivialmente extensible a un caso mas general.

El trabajo descrito en este capítulo y las técnicas de búsqueda algorítmicas propuestas fueron resumidas y publicadas en la revista Elsevier Ad-Hoc Networks [4], particularmente en la edición especial “New Research Challenges in Mobile, Opportunistic and Delay-Tolerant Networks”publicada en Febrero del 2015. El título del artículo es “Routing-Aware Fair Contact Plan Design for Predictable Delay Tolerant Networks” y ha tenido un buen recibimiento de la comunidad en base a preguntas y propuestas recibidas por correo electrónico luego de su publicación.

En efecto, estas metodologías prueban redundar en un mejor comportamiento del flujo de datos en un sistema DTN predecible para el cual el volumen de tráfico a futuro permanece un incógnita. Sin embargo, este último es generalmente predecible particularmente en redes satelitales donde el tráfico de telemetría (información de la plataforma) se genera de manera periódica (conocida de antemano) y el de ciencia se produce bajo demanda de un operador en tierra (MOC). Así como en los mecanismos de diseño de plan de contacto basado en rutas (RACP) explotamos el conocimiento de las rutas, la inclusión

del conocimiento del tráfico del sistema nos permite considerar esquemas de diseño basados en tráfico como planteamos en el siguiente capítulo 5.

Capítulo 5

Diseño de Plan de Contactos basado en Tráfico

5.1. Introducción

En general, y retomando lo definido en la sección 2.6 del capítulo 2, el problema de diseñar un plan de contacto yace en seleccionar aquellos de tal manera que la selección cumpla las restricciones de TZC y CRC y al mismo tiempo optimicen algún criterio del sistema [5]. A lo largo de la tesis hemos recorrido esta problemática del diseño de planes de contactos utilizando solamente la información de la topología de contactos considerando un solo salto en el capítulo 3, para luego incorporar una mayor inteligencia de múltiples saltos (rutas) en el capítulo 4. Sin embargo, en la red final, una ruta no sólo se define por la cantidad de nodos intermedio que un dato dado debe visitar hasta llegar a su destino, si no que también depende del volumen del mismo.

En este capítulo abordamos estas consideraciones específicas de las características del tráfico con el fin de mejorar la calidad de los planes de contactos diseñados para redes DTN predecibles. En efecto, en este apartado se hace un aporte de un modelo teórico a ser publicado a principios del 2016 en el Capítulo 15 ("Contact Plan Design for Predictable Disruption Tolerant Space Sensor Networks") del libro "Wireless Sensor Systems for Extreme Environments: Space, Underwater, Underground and Industrial" de la editorial Wiley [6]. El lector interesado puede encontrar un resumen de este aporte en un reporte técnico [7] mantenido por el Laboratorio de Comunicaciones Digitales. Además, dado que el modelo teórico solo puede resolver casos simples, se detalla y analiza una propuesta metaheurística de algoritmo evolutivo como otro aporte realizado, a ser publicado a finales del 2015 en la conferencia IEEE International Conference on Wireless for Space and Extreme Environments (WiSEE) en Orlando, Florida [8].

5.1.1. Suposiciones del Esquema

En general, en aplicaciones espaciales, el tráfico de la red resulta en gran parte predecible y determinístico dado que en el caso de datos de ciencia los mismos son generados bajo demanda controlada de un operador del instrumento o carga útil en tierra (probablemente desde en centro de operación de misión o MOC), mientras que los datos de telemetría que acusan métricas de la plataforma (estado de baterías, posibles alertas, etc.) se generan de manera periódica y conocida para el administrador del sistema. En efecto, en este capítulo asumiremos esta condición que se adosa a las previamente asumidas en el capítulo 3 de conocimiento previo de la topología por medio de propagadores orbitales[88], y en el capítulo 4 de predicción del comportamiento del esquema de ruteo implementado en el software de vuelo de los satélites.

5.2. Planteo Formal del Problema

En esta sección nos encargamos de plantear formalmente el problema de diseño de plan de contactos basado en el criterio de tráfico esperado en el sistema. Este planteo fué detallado en un reporte técnico [7] y está pendiente de evaluación para una revista y también es parte del eje de un capítulo de libro [6]. En efecto, este enfoque constituye una mejora a los mecanismos revisados en los capítulos 3 y 4, aunque supone un mayor desafío en cantidad de datos y variables a analizar.

De hecho, en el caso de los esquemas basados en topología y rutas era factible realizar una inspección manual en casos simples para verificar su comportamiento (sección 4.4.3), sin embargo, resulta poco factible comprender la optimalidad de un plan de contacto basado en tráfico por mas sencillo que sea. En consecuencia nos vemos forzados a plantear formalmente el problema con un nuevo modelo de programación lineal de enteros mixtos (Mixed Integer Linnear Programming en Inglés o MILP) de complejidad notablemente superior al propuesto para el criterio de justicia en la sección 3.2.

Antes de ofrecer y detallar la formulación de este modelo teórico del problema, en la siguiente sección 5.2.1 definimos precisamente el concepto de tráfico en redes DTN y como impacta su comportamiento en el sistema final.

5.2.1. Definición de Tráfico en DTN

Previamente, en el capítulo 4 hemos utilizado la definición de rutas como una secuencia de contactos de la forma $Route = \{C_1, C_2, C_3, ...C_n\}$ por medio de los cuales los datos deberán fluir para llegar a su destino final. En efecto, cuando dos contactos no se dan

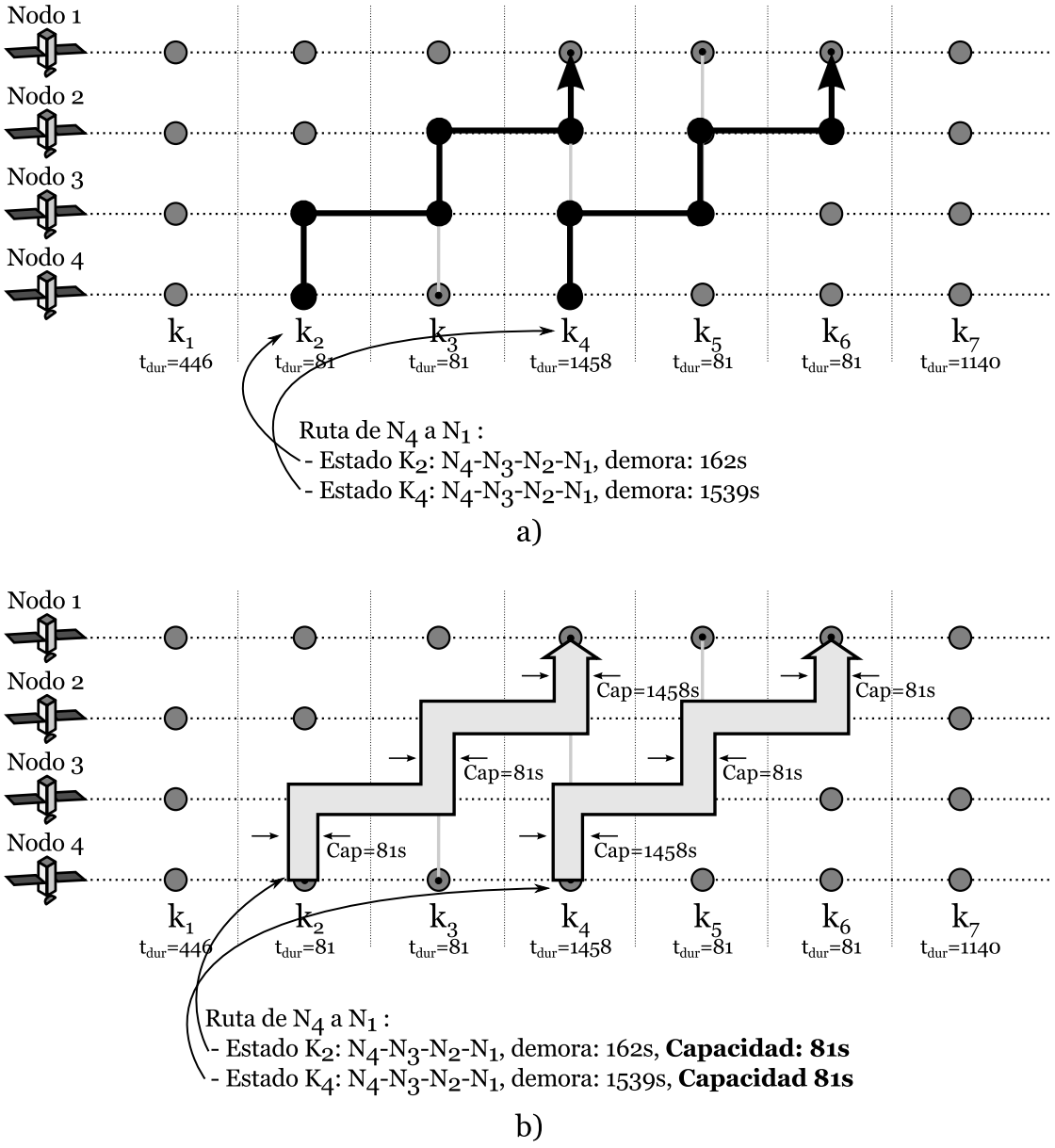


FIGURA 5.1: Rutas con capacidad de tráfico en DTN

de manera simultánea, el dato debe ser almacenado en la memoria del nodo intermedio. Esta definición se ilustra en la Figura 5.1 a) para la ruta de N_4 a N_1 en una topología basada en el caso de estudio A (detallado en la sección 2.3.1) para dos estados k_2 y k_4.

Sin embargo, en esta definición se deja afuera el concepto de tráfico o volumen de datos que fluirá por esa ruta. Es decir, se ignora cuantos datos es capaz de manejar la ruta especificada. En las redes de Internet tradicionales este parámetro no resulta de mayor relevancia dado que sobre las rutas extremo a extremo se ejecutan protocolos como TCP que regulan la tasa de transferencia de datos entre los nodos origen y destino. Pero cuando se consideran disrupciones no se puede contar con la retroalimentación (*feedback*

en Inlgés) de TCP, por lo que conocer la capacidad de antemano puede ayudar a resolver importantes conflictos embotellamiento como detallamos a continuación.

Como se puede observar en la Figura 5.1 b), la definición de flujo de tráfico no sólo engloba la secuencia de contactos (ruta) si no que también la capacidad o volumen de datos. En otras palabras, un flujo de datos en una red DTN se define como una secuencia de contactos $Route = \{C_1, C_2, C_3, ...C_n\}$ sobre los cuales puede viajar un valor máximo de datos (capacidad). Cabe destacar que medir la capacidad en unidades de tiempo resulta equivalente a la tasa de datos si se asume que la misma es constante en los transponders de todos los nodos del sistema ($dataRate * tiempo = volumenTrafico$).

Por ejemplo, en la ilustración de la figura, el flujo de N_4 a N_1 en el estado k_4 tiene entre N_4 y N_3 una capacidad de 1458 segundos, y una de 81 segundos entre N_3 y N_2, y N_2 y N_1. Esto implica que el tráfico transmitido en origen por N_4 no puede superar la capacidad de 81 segundos a pesar de que su primer enlace con N_3 puede acomodar un total de 1458 segundos. En otras palabras, el plan de contacto resultante de la Figura 5.1 b) podría satisfacer la evacuación de un tráfico N_4-N_1 de volumen 162 segundos generado antes del tiempo 446 segundos (k_1) para entregarlo al final del estado k_6.

Por otro lado, en este ejemplo se puede inferir una problemática mayor en relación a que si no se cumple esta condición, se genera un efecto denominado *congestión* el cual se debe evitar con una distribución apropiada de información. Abordaremos esta interesante problemática en el capítulo 6 de este trabajo doctoral.

En resumen, el tráfico o flujo de datos se puede ver como una definición mas completa que la definida para ruta en el capítulo 4, la cual mantiene la cualidad de predecible para el caso de redes espaciales. En particular, se puede considerar de antemano que *volumen*, en que *tiempo*, y con que *destino* se generará tráfico en la red, lo que permite diseñar planes de contactos capaces de evacuarlos de la manera mas eficiente en términos de métricas de tiempo de entrega.

5.2.2. Modelo MILP

Una vez definido el flujo de tráfico, en esta sección formalizaremos el planteo teórico del problema por medio de su expresión en un modelo MILP. En efecto, en el mismo incluiremos variables y restricciones que modelen el tráfico esperado, capacidades en los enlaces, y memorias en los nodos. De ahora en adelante denominaremos a esta aproximación como diseño de plan de contacto basado en tráfico o Traffic Aware Contact Plan (TACP) en Inglés. Dado que en esta formulación en particular nos basamos en un

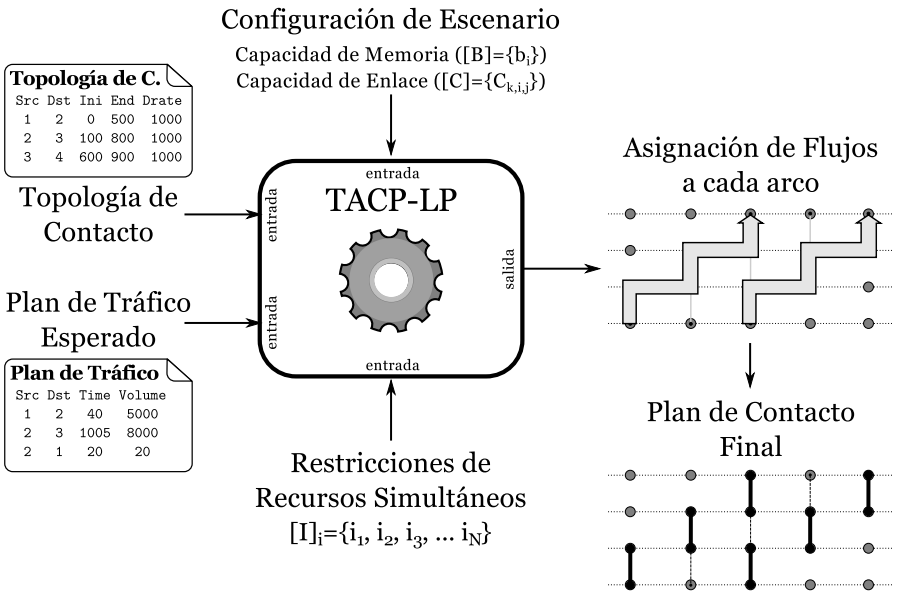


FIGURA 5.2: Flujo de entradas y salidas de TACP

modelo de programación de lineal la especificaremos como TACP-LP para distinguirla de una segunda propuesta metaheurística TACP-GA explicada en la sección 5.3.

A la salida, el modelo TACP-LP entrega una asignación de flujos a cada contacto (*forwarding assignment* en Inglés) libre de congestiones que minimiza el tiempo total de entrega para el tráfico especificado a la entrada. En este proceso, TACP-LP toma decisiones de activación de interfaces que cumplen la lista de restricciones de límites de recursos por arquitectura explicados en la sección 2.4. Finalmente, de esta asignación de flujos ajustadas a las limitaciones de CRC se puede derivar el plan de contacto final optimizado para el escenario planteado. El flujo completo de trabajo de TACP en general se puede observar en la Figura 5.2.

En la próxima sección 5.2.2.1 listamos y detallamos las variables y coeficientes internos a TACP-LP para luego en la sección 5.2.2.2 integrarlas en el planteo formal final.

5.2.2.1. Variables de Decisión y Coeficientes

En general, en su núcleo, TACP utiliza una representación de la topología en función del tiempo basada en el modelo de estado finito o FSM explicado en la sección 2.3.2 del capítulo 2. Sobre este, se formula un esquema similar al problema ya conocido de flujos con múltiples destinos o *multi-commodity flow problem* en Inglés [124]. En particular, en TACP-LP, se la da un enfoque temporal basado en el modelo de máquina de estado (FSM) como se detalla a continuación.

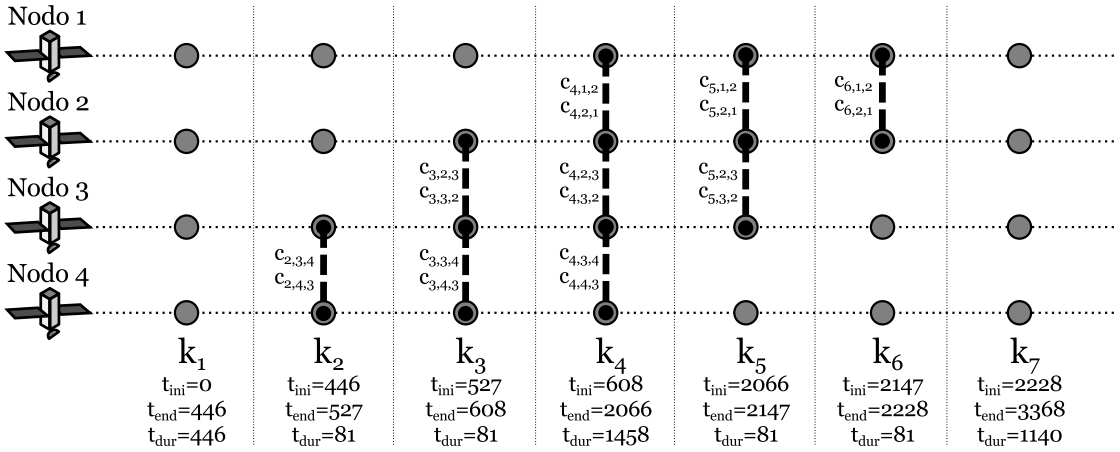


FIGURA 5.3: Topología de contacto para la primera media órbita del caso de estudio A

Dentro los coeficientes relacionados con el modelado del tiempo y estados de la topología de contacto de entrada, TACP asume que dos matrices $T = \{t_k\}$ y $I = \{i_k\}$ se completan al inicio. Fundamentalmente, $T = \{t_k\}$ se compone de los tiempos de inicio de cada estado mientras que $I = \{i_k\}$ contiene los intervalos de duración de cada estado k. La Figura 5.3 ilustra estos tiempos para cada estado de la primera media órbita del caso de estudio y referencia A (topología escalera) explicado en la sección 2.3.1. En efecto, y como se explicó en la sección 2.3.2.1, el fraccionamiento de estados puede resultar una herramienta útil a la hora de aplicar el modelo MILP para incrementar la granularidad del modelo dando lugar a planes de contactos mas precisos a la salida. En general, una topología de contacto muy granular impacta directamente en la complejidad de los procesos de diseño y puede generar inconvenientes en el momento de aplicar el plan de contacto final debido al mayor número de interrupciones de las comunicaciones. En efecto, una alta tasa de cambios en encendido y apagado de transponders puede volver al sistema extremadamente sensible a pérdidas de sincronismos entre nodos. En general es aconsejable adoptar un criterio balanceado a la hora de considerar el fraccionamiento puntualmente para el planteo TACP-LP.

Una vez modelado los tiempos y estados de la máquina FSM, se deben modelar las capacidades de los arcos de manera que se pueda a su vez evaluar el tráfico que puede cursar sobre ellos. En consecuencia la topología de contacto se expresa por medio de un conjunto de capacidades $c_{k,i,j}$ para cada estado k, entre un nodo i y uno j. En otras palabras hay un $c_{k,i,j}$ que representa el volumen de tráfico que se puede transmitir entre estos nodos en el intervalo $[t_{k-1}, t_k]$. De esta manera, como se muestra en la Figura 5.3, cada arco del modelo cuenta con una capacidad asociada la cual en general se puede determinar por la duración del intervalo como producto de la tasa de transmisión de los equipos de comunicaciones en ese momento ($c_{k,i,j} = rate_{i,k} * t_k$). En general, esta tasa de datos $rate_{i,k}$ es la que se configura en el plan de contactos del software ION [67]

detallado en la sección 1.3.3.4. En nuestro modelado particular, el valor de $c_{k,i,j} = 0$ nos sirve para mostrar que no es posible utilizar el contacto entre el nodo i y j en el estado k. Finalmente, la topología de contacto puede modelarse completamente por medio una matriz $C_{k,i,j}$ de tamaño $K \times N \times N$ correspondiente a las matrices de tiempo $T = \{t_k\}$ y $I = \{i_k\}$.

Por otro lado, además del modelado de la topología de contacto, al incorporar el tráfico en el planteo, se hace menester estudiar y reflejar el uso de la memoria de almacenamiento persistente típicos de una red DTN. Ciertamente, dado que en DTN se usa el paradigma *store-carry-and-forward* (SCF), la capacidad del sistema en su totalidad no solamente se relaciona con la tasa de datos de los arcos (expresada en la matriz $[C]$) si no que también con la capacidad de almacenamiento de datos en cada nodo intermedio. En efecto, en este modelo incluimos un coeficiente b_i en el que codificamos la máxima cantidad de datos que un satélite i puede almacenar en todo momento (es decir, para todo k). Como resultado, modelaremos la utilización efectiva de ese espacio con una variable $B_{k,i}^{y,z}$ que indica la cantidad de ocupación generada en el estado k, en el nodo i, debido a un flujo de tráfico que tiene como origen a y y destino a z. Cabe destacar que cuando el tráfico se almacena en un nodo intermedio de la ruta $i \neq y$. Efectivamente, la sumatoria de todas las variables $B_{k,i}^{y,z}$ para todo y y z en cierto estado k no deberá exceder la capacidad máxima expresada en b_i. De esta manera, la matriz $[B] = \{b_i\}$ es también requerida en la entrada del modelo para acotar la capacidad del sistema a analizar.

En consecuencia, el conjunto de coeficientes $[C]_{k,i,j} = \{c_{k,i,j}\}$ y $[B]_i = \{b_i\}$ definen la capacidad de la topología de contacto sobre la cual se deberá calcular el flujo de tráfico. Cabe destacar que la información referida a este último no había sido tenida en los trabajos previos [2, 4] ya que se asumía desconocida. En consecuencia, TACP es el primer planteo formal que explota la utilización de esta valiosa información generalmente disponible para este tipo de redes. En particular, para modelar el flujo de tráfico a lo largo del tiempo (o estados), proponemos la incorporación de un conjunto de variables $X_{k,i,j}^{y,z}$ que codifiquen la cantidad de tráfico de fuente original y y destino z, que fluye sobre el enlace i a j en el estado k. Evidentemente, la sumatoria de todos los flujos para un conjunto de y-z diferentes no deberá sobrepasar la capacidad máxima del contacto i-j, es decir, a $c_{k,i,j}$. Por otro lado, tampoco deberá provocar un *overflow* del buffer al que el flujo deberá arribar en el nodo receptor (b_j). En este punto vale la pena aclarar que dado que estamos modelado redes de satélites LEO, el modelo solamente necesita modelar las disrupciones del sistema ya que las demoras resultan despreciables. En otras palabras, el flujo $X_{k,i,j}^{y,z}$ se asume que llega instantáneamente al destino j. A pesar de esta decisión, el modelo puede ser fácilmente extendido en el futuro para su aplicación en redes interplanetarias por medio de un parámetro de demora de propagación como se propone en [125].

Una vez definido el conjunto de variables que modelarán el flujo en el sistema, es necesario definir los disparadores de los mismos, es decir, las fuentes de tráfico. De esta manera, cada flujo $X_{k,i,j}^{y,z}$ resultante en el sistema debe estar motivado a existir por un origen de tráfico que expresaremos en una matriz $[D]$, donde $[D] = \{d_k^{i,j}\}$ conocida de antemano. En particular, este plan de tráfico se forma de un conjunto de volúmenes de datos $d_k^{i,j}$ representando la cantidad de información a ser generada en el tiempo t_k (estado k), en el nodo i con destino final j. En esta formulación plantearemos el modelo para que el mismo sea capaz de modelar múltiples generadores de trafico $d_k^{i,j}$ para los mismos pares de nodos i-j a lo largo de la evolución de los estados k. Este modelado del tráfico, combinado eficientemente con técnicas de fraccionamiento, permite colocarlos cronológicamente en cualquier punto de tiempo del intervalo de topología. Es decir, si es necesario generar un tráfico en un punto intermedio de un estado k_n, se puede particionar el mismo para que queden dos estados k_{n1} y k_{n2} de manera que el inicio de k_{n2} (o lo que es lo mismo, el final de k_{n1}) coincida con el tiempo de generación del tráfico. Este proceso se ejemplifica en la Figura 5.4 donde una fuente de tráfico debe ubicarse en el tiempo relativo de

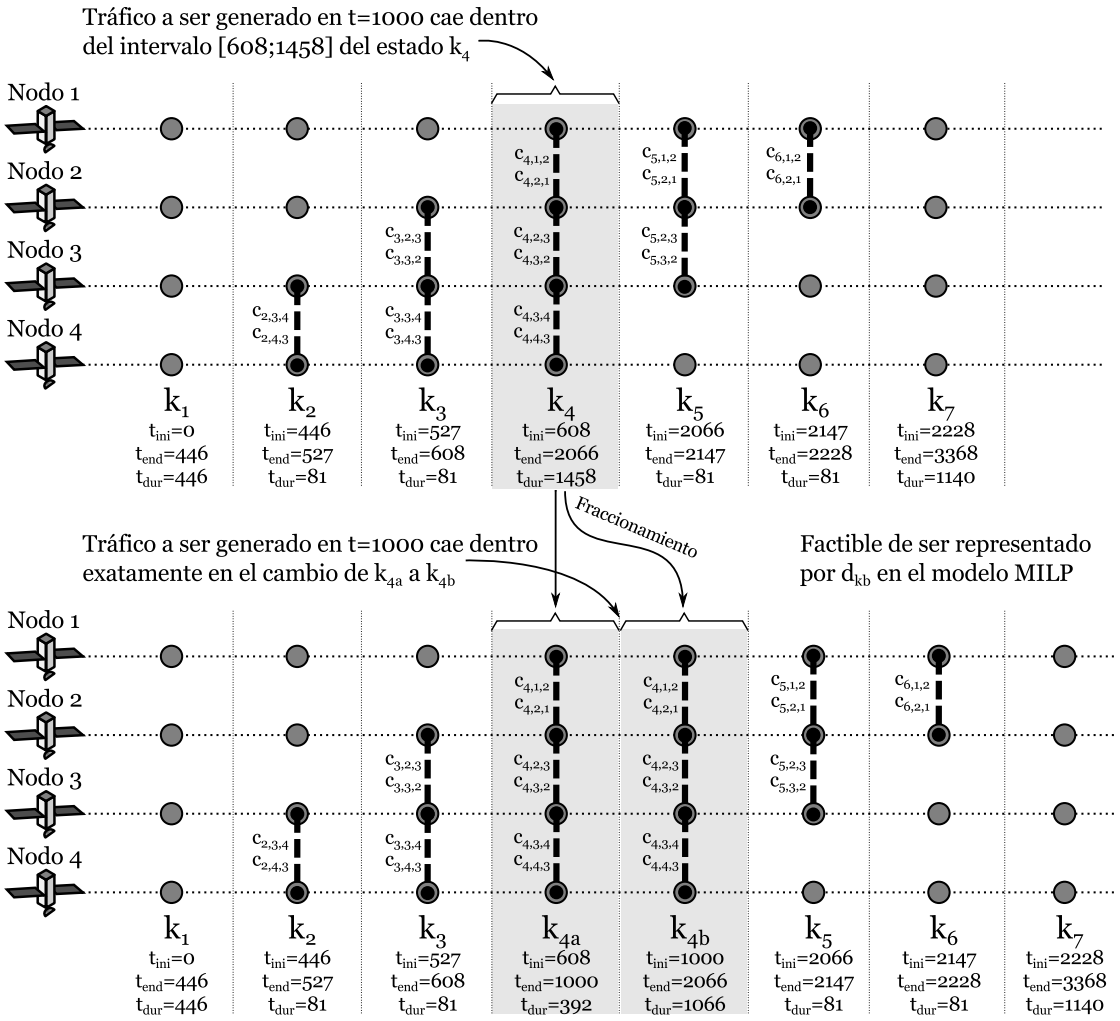


FIGURA 5.4: Utilización del fraccionamiento para el modelado de creación de tráfico

Tabla 5.1: Parámetros del modelo MILP de TACP

Coeficientes de Entrada	
N	Cantidad de nodos
K	Cantidad de estados en la topología
$T = \{t_k\}$	Tiempo de inicio del estado k
$I = \{i_k\}$	Intervalo de duración del estado k ($i_k = t_{k+1} - t_k$)
$C = \{c_{k,i,j}\}$	Capacidad de i a j en el estado k
$B = \{b_i\}$	Capacidad de almacenamiento del nodo i
$D = \{d_k^{i,j}\}$	Tráfico desde i a j originado en k
$P = \{p_i\}$	Número de puertos simultáneos en el nodo i
Variables de Salida	
$\{X_{k,i,j}^{y,z}\}$	Tráfico de y a z en k en el arco i a j
$\{B_{k,i}^{y,z}\}$	Ocupación de memoria del nodo i por tráfico de y a z en k
$\{Y_{k,i,j}\}$	Selección de puerto de i a j en el estado k

1000 segundos cayendo dentro del estado k_4 de la topología del caso de estudio A. La utilización del fraccionamiento permite crear un estado auxiliar k_{4b} cuyo inicio coincide con el momento en el que se debe generar el tráfico permitiendo el modelado del mismo bajo el coeficiente $d_{k4b}^{i,j}$.

Por último pero no menos importante, las unidades de los coeficientes C, B, D y las variables $X_{k,i,j}^{y,z}$ deben ser obligatoriamente los mismos, adoptado típicamente unidades de bits, Bytes, o inclusive paquetes o bundles si el tamaño de los mismos es constante. Cabe repetir entonces que en caso de que las tasas de transmisiones en todos los nodos de la red sean iguales, las unidades de estos parámetros pueden ser directamente vinculados o traducidos a unidades de tiempo de acceso al canal.

Al ya definir la capacidad de la topología de contacto en $[C]$ y $[B]$, y el tráfico a ser evacuado en $[D]$, el modelo puede ser capaz de calcular las asignaciones óptimas de flujo a cada una de las variables $X_{k,i,j}^{y,z}$. En esta etapa el modelo puede ser utilizado como esquema de ruteo en redes DTN predecibles al igual que MFW y CGR, pero con mejores prestaciones en término de información de tráfico global. Sin embargo difícilmente pueda ser considerado para un nodo de vuelo dada la complejidad computacional asociada a la resolución de este tipo de problemas [124]. Por otro lado, podemos seguir extendiendo el modelo para que el mismo ahora considere las limitaciones de recursos traducidas en la máxima cantidad de puertos que un satélite puede implementar en un momento dado. Con este fin, proponemos el uso de una matriz $P = \{p_i\}$ cuyos componentes p_i codifican la máxima cantidad de contactos simulatáneos que el nodo i puede utilizar. En particular, este planteo de $[P]$ otorga al modelo TACP-LP una flexibilidad única de

considerar múltiples interfaces a diferencias de FCP [2] que está limitado a $p_i = 1 \quad \forall i$. En consecuencia, y para modelar la naturaleza combinatoria de este problema de elección incluimos un conjunto de variables $Y_{k,i,j}$ que pueden adoptar un valor binario de 1 en caso de que el puerto de i a j sea elegido en el estado k, y 0 en su defecto.

Finalmente, el modelo TACP-LP, puede ser utilizado para obtener una selección de las interfaces o puertos que optimicen la entrega del tráfico de entrada especificado en la matriz $[D]$ a través de la topología de contacto expresada en $[C]$ con capacidades de almacenamiento $[B]$ y restricciones de recursos $[P]$. Como resultado, la formulación del problema entrega una asignación de flujos a los arcos en las variables $X^{y,z}_{k,i,j}$, una utilización de las capacidades de memoria en $B^{y,z}_{k,i}$ y una lista de elecciones de interfaces en $Y_{k,i,j}$. La Tabla 5.1 resume y lista todos coeficientes y variables discutidos para finalmente entrar en la formulación de la función objetivo y las restricciones del problema MILP en la siguiente sección 5.2.2.2.

5.2.2.2. Función Objetivo y Restricciones

Una vez definidas los coeficientes y variables, en esta sección procedemos a vincularlas para finalmente concretar el modelo TACP-LP como un planteo MILP completo capaz de diseñar planes de contactos eficientes para los tráficos esperados. En efecto, esta vinculación la realizamos por un lado con la expresión de una función objetivo que tienda a minimizar el tiempo de entrega del tráfico $[D]$ así como también el uso de los arcos que permitan dicha entrega; y por otro con la formulación de restricciones a las variables en juego.

La función objetivo se expresa en la ecuación (5.1) y fuerza el modelo a minimizar la suma del producto de las unidades de datos ($X^{y,z}_{k,i,j}$) con el tiempo asociado al estado k (t_k) en el cual se envían las mismas. Además, cada término de esta suma se multiplica por una función de ponderación $w(t_k)$ que a mayor peso, mayor la importancia del uso de arcos de manera temprana, mientras a que a menor valor, mayor importancia a minimizar la cantidad de arcos usados. Por ejemplo, consideremos el escenario específico ilustrado en la Figura 5.5 donde se espera que fluya un único flujo de tráfico desde el N_4 al N_1 ya sea por el camino en a) o el mostrado en b). Si el coeficiente $w(t_k)$ se configura con $w(t_k) = t_k$, el plan de contacto b) permite minimizar la función a un valor de 300, entregando una asignación de arcos que minimiza la cantidad de enlaces usados pero no entrega el tráfico en el menor tiempo posible. En caso de que el tiempo de entrega deba ser optimizado de manera prioritaria, una mayor cantidad de arcos pueden ser habilitados en los estados k_1 y k_2 en lugar del seleccionado en k_3. Para lograr este efecto, el coeficiente $w(t_k)$ puede ser configurado por un valor mayor en proporción al

$$\text{minimizar:} \quad \sum_{k=1}^{K}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{y=1}^{N}\sum_{z=1}^{N} w(t_k) * X_{k,i,j}^{y,z} \tag{5.1}$$

Sujeto a:

$$\sum_{j=1}^{N} X_{k,j,i}^{y,z} - \sum_{j=1}^{N} X_{k,i,j}^{y,z} = B_{k,i}^{y,z} - (B_{k-1,i}^{y,z} + d_k^{i,z}) \quad \forall k, i, y, z \tag{5.2}$$

$$B_{k,i}^{y,z} <= b_i \quad \forall k, i, y, z \tag{5.3}$$

$$B_{0,i}^{y,z} = 0 \quad \forall i, y, z \tag{5.4}$$

$$\sum_{y=1}^{N}\sum_{z=1}^{N} X_{k,i,j}^{y,z} <= c_{k,i,j} \quad \forall k, i, j \tag{5.5}$$

$$\sum_{k=1}^{K}\sum_{j=1}^{N} X_{k,i,j}^{y,z} = \sum_{k=1}^{K} d_{k,i,z} \quad \forall i = y, z \tag{5.6}$$

$$\sum_{k=1}^{K}\sum_{i=1}^{N} X_{k,i,j}^{y,z} = \sum_{k=1}^{K} d_{k,y,j} \quad \forall y, j = z \tag{5.7}$$

$$\sum_{j=1}^{N} Y_{k,i,j} <= p_i \quad \forall i, k \tag{5.8}$$

$$\sum_{j=1}^{N}\sum_{y=1}^{N}\sum_{z=1}^{N} X_{k,i,j}^{y,z} <= M * Y_{k,i,j} \quad \forall i, k \tag{5.9}$$

valor de t_k. En general, un $w(t_k) = K * t_k^2$ es suficiente para garantizar el menor tiempo de entrega posible, entregando valores de la función objetivo de 17000 (óptimo) y 27000 para los planes de contactos de la Figure 5.5 a) y b) respectivamente.

En general, vale la pena enfatizar que la propuesta de $w(t_k) = K * t_k^2$ mantiene el principio de que $w(t_k)$ como un coeficiente independiente de las variables de decisión $X_{k,i,j}^{y,z}$, $B_{k,i}^{y,z}$ y $Y_{k,i,j}$. Es decir, la elevación a la segunda potencia no pone en jaque la linealidad de la formulación del problema. Por otro lado, se puede observar que $w(t_k)$ puede incrementar de manera drástica en unos pocos contactos, por lo que se debe mantener bajo inspección permanente con el fin de evitar sobrepasar los límites numéricos de los solvers MILP. En pocas palabras, la función objetivo como se plantea en (5.1) tiene un objetivo dual y ponderable de optimizar el tiempo de entrega de todo el tráfico configurado y de minimizar la cantidad de arcos utilizados.

Por otro lado, entre las restricciones, la ecuación (5.2) vincula el desbalance de flujos en cada nodo i con su variación de ocupación de memoria (*buffer*) para todos los estados k y todos los pares de generadores de tráfico (y, z). Además, $d_k^{i,z}$ se incluye en el desbalance para modelar la generación instantánea del tráfico en el nodo i con destino z.

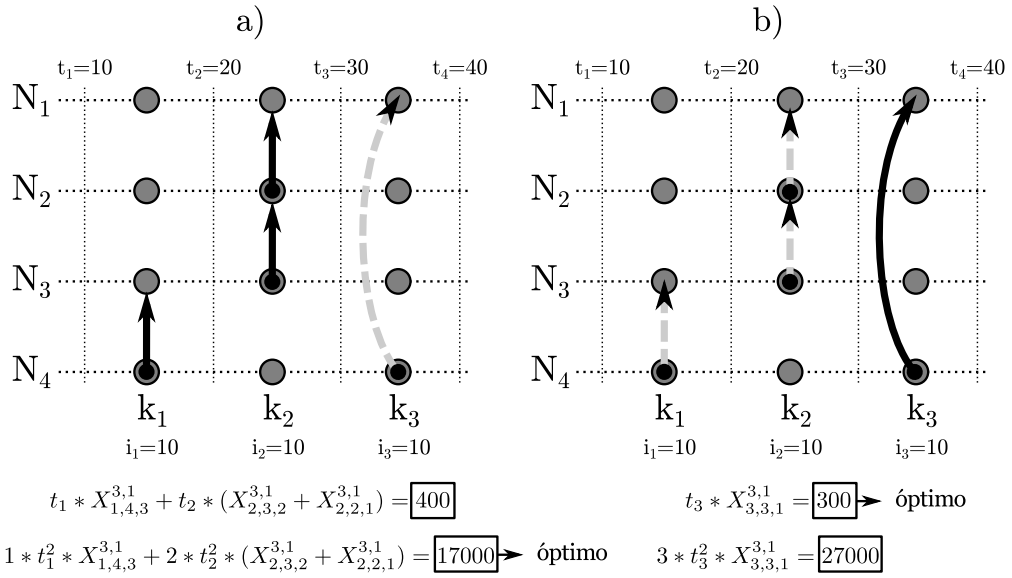


FIGURA 5.5: El impacto del coeficiente $w(t_k)$ en la función objetivo de TACP-LP

En efecto, $d_k^{i,z}$ es o bien transmitido instantáneamente (incrementado la variable $X_{k,i,j}^{y,z}$), bien almacenado en la memoria local (incrementando $B_{k,i}^{y,z}$). Luego, las ecuaciones (5.3) y (5.4) imponen una cota superior a las memorias b_i y configuran la condición inicial de las mismas $B_{0,i}^{y,z} = 0$ para cada nodo i y cada flujo y, z respectivamente. Para estados futuros $k > 0$, la ocupación de los buffers podrá incrementar tanto por la generación de tráfico local $d_k^{i,z} > 0$ o la recepción de flujos de otros nodos expresado en las variables $X_{k,j,i}^{y,z}$ y decrementar por la transmisión de datos. La máxima capacidad de cada arco se establece en la ecuación (5.5). En particular, si todos los nodos transmiten con la misma tasa de datos, todos los arcos existentes dentro de un mismo estado tendrán los mismos valores de $c_{k,i,j}$. Luego, las ecuaciones (5.6) y (5.7) generan el desbalance entre las fuentes y destinos de cada par de nodos (y, z). En otras palabras, estas ecuaciones obligan al nodo transmisor a efectivamente enviar el dato y al receptor a recibirlo. Además, la ecuación (5.2) cumple la función de evitar que un tráfico $d_k^{i,z}$ sea transmitido antes del estado k específico para el cual es planificado. De esta manera, y en resumen, las ecuaciones (5.2) a (5.6) modelan la topología de contacto y las capacidades de memoria así como también obligan a la transmisión y recepción de los datos planificados.

Además de las restricciones nombradas, se deben considerar aquellas relativas a las limitaciones de recursos del sistema. En este sentido, las ecuaciones (5.8) y (5.9) limitan la cantidad máxima de contactos simultáneos en cada nodo i. En particular, la ecuación (5.8) verifica que la sumatoria de las variables binarias $Y_{k,i,j}$ satisfaga el límite p_i para el nodo i en cuestión. Por otro lado, la ecuación (5.9) vincula la selección de puerto o interface con la habilitación del envío de datos $X_{k,j,i}^{y,z}$ por medio de ese recurso. En efecto, si un puerto es desactivado ($Y_{k,i,j} = 0$), ninguno de los flujos correspondientes al envío o recepción de tráfico deberá tener datos para el arco respectivo. Para lograr este

objetivo, utilizamos un método ampliamente conocido para el modelo LP conocido como el método de la "gran M" o "big M" en Inglés. Este esquema permite que en el caso de que un puerto sea activado ($Y_{k,i,j} = 1$), un coeficiente lo suficientemente grande M multiplica la variable binaria $Y_{k,i,j}$ de manera tal que parte izquierda de la ecuación con los flujos correspondientes a esa interfaz puedan crecer en valor sin ninguna limitación. Al contrario, cuando el puerto no sea elegido, el término $M * Y_{k,i,j} = 0$ fuerza a todas las variables $X^{y,z}_{k,j,i}$ del otro lado de la ecuación a que tomen un valor de 0 (las $X^{y,z}_{k,j,i}$ no pueden tomar valores negativos). Para lograr este efecto, el valor de M debe ser lo suficientemente grande para permitir que la sumatoria de todos los posibles flujos sobre ese contacto pueda darse sin cotas (es decir $M > c_{k,i,j}$). Si bien la ecuación (5.9) permite al modelo incluir un mecanismo válido para generar una selección de una cantidad determinada de puertos, es sabido que la aproximación de "big M" puede provocar inestabilidad numérica en la mayoría de los solvers especialmente cuando $M >> \sum X^{y,z}_{k,j,i}$. En consecuencia, el valor de M debe ser cuidadosamente elegido para satisfacer el requerimiento de la ecuación con el mínimo valor posible.

Para resumir, el modelo TACP-LP permite combinar eficiente la información de la predicción de la topología de contactos y el tráfico planificado de la red para proveer una asignación eficiente de los mismos a cada enlace de comunicación respetando las restricciones de recursos del sistema. Como se ilustra en la Figura 5.2, esta asignación puede ser utilizada para finalmente derivar el plan de contacto para suministrar a los nodos DTN orbitantes para que puedan tomar decisiones de enrutamiento eficientes. Sin embargo, la formulación MILP planteada es una extensión del problema conocido como *multi-commodity flow problem* el cual es catalogado como fuertemente no polinómico (strongly NP-complete en Inglés) [124] implicando que el esfuerzo computacional requerido para resolverlo incrementa exponencialmente con la cantidad de variables ($X^{y,z}_{k,j,i}$, $B^{y,z}_{k,i}$, y $Y_{k,i,j}$ en este caso). En consecuencia, la aplicabilidad de TACP-LP resulta limitada en aplicaciones de DTN reales, pero sirve de cota de rendimiento para dar lugar a mecanismos alternativos como los planteados en la siguiente sección 5.3.

5.3. Planteo Algorítmico

En general, a pesar de la complejidad del tipo NP-Complete de un problema como el formulado para TACP-LP en la sección 5.2.2, los mismos pueden ser utilizados para casos simples y acotados brindando soluciones óptimas por medio de mecanismos conocidos como *brunch and bound* para problemas de enteros mixtos (MILP) ampliamente disponibles en *solvers* comerciales (IBM ILOG CPLEX Optimizer [126]) o gratuitos

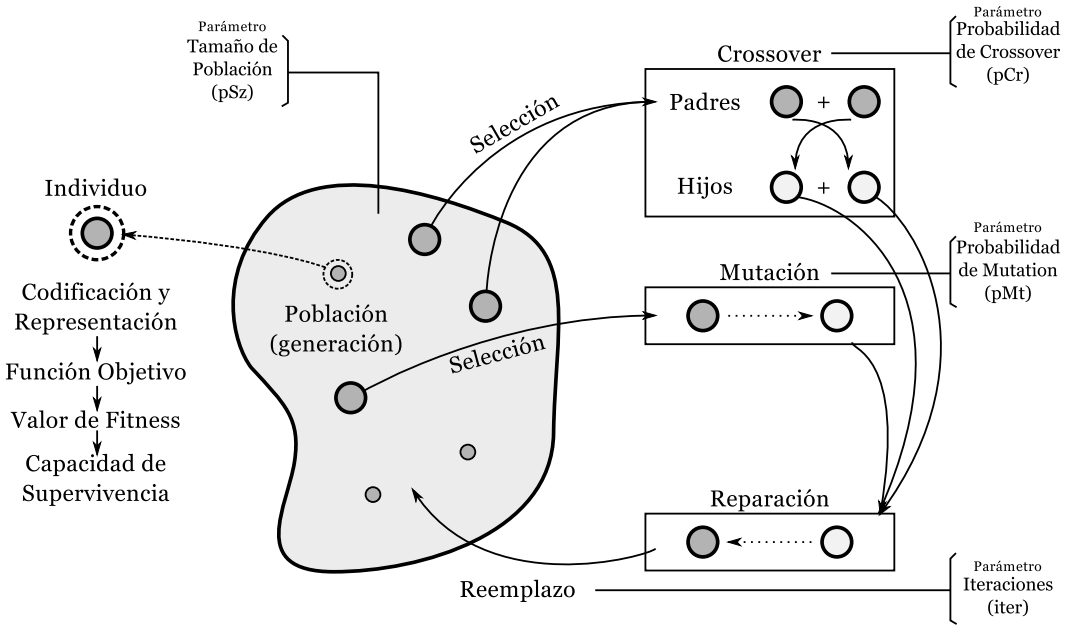


FIGURA 5.6: Procedimiento general de los algoritmos evolutivos

(GLPK [108]). Sin embargo, al considerar instancias lo suficientemente grande del problema (en término de variables), estos métodos óptimos fallan en entregar soluciones en tiempos razonables dado que el esfuerzo de procesamiento requerido aumenta exponencialmente con la dimensión de las variables de decisión. Para estos casos, se necesitan de metodologías aproximadas como las metaheurísticas que entreguen soluciones subóptimas pero de todas maneras aceptables para el uso general del sistema. Al día de la fecha, no existen soluciones similares para este problema.

5.3.1. Algoritmo Genético

Las metaheurísticas son parte de la familia de las técnicas de optimización aproximadas, y han probado de ser de extremo valor en la resolución y generación de buenas soluciones en un tiempo acorado de problemas altamente complejos en el campo de la ciencia y la ingeniería [117]. En general, las metaheurísticas poblacionales (o *P-Metaheuristics* en Inglés) mejoran iterativamente una población o conjunto de soluciones posibles con el objetivo de ir aproximándose al valor óptimo de una función objetivo dada. Dentro de esta clasificación, los algoritmos evolutivos resultan los mas estudiados particularmente para problemas combinatorios (discretos) [117] como el TACP planteado en la sección 5.2.2. En efecto, en esta sección diseñamos un método alternativo bajo este paradigma computacional con el fin de poder tratar el problema de diseño de plan de contacto basado en tráfico para redes DTN de tamaño medio o grande [8].

Los algoritmos evolutivos (*evolutionary algorithms* o EA en Inglés) se basan en la noción de *competencia* al imitar la evolución de las especies [127] (otros esquemas como la optimización de colonia de hormigas o ACO se basan en la *cooperación*). En nuestra formulación [8] particular de EA, la *población* inicial de soluciones es generada de manera aleatoria, donde cada *individuo* representa una solución factible al problema (es decir, conjunto de variables de decisión), para el cual una *función objetivo* permite definir su aptitud (o *fitness* en Inglés) el cual a su vez determina la capacidad de supervivencia en el entorno generado por el sistema. Entre estos individuos, en cada iteración del algoritmo, un conjunto de *padres* (o *parents* en Inglés) se seleccionan para ser sometidos al *cruzamiento* (o *crossover* en Inglés) o operadores de *mutación*. Luego, las soluciones *hijo* (u *offsprings* en Inglés) deben ser analizadas para reemplazar otros individuos de la población por medio de mecanismos y políticas de *reemplazo*. En efecto, a medida que se repite este procedimiento, la población evoluciona de manera estocástica guiado por un conjunto de estrategias hacia una solución cada vez mas óptima.

La Figura 5.6 ilustra gráficamente este procedimiento para el cual hemos diseñado estrategias específicas que se adaptan a la resolución del problema del diseño de plan de contactos basado en tráfico planteado en este capítulo. En las próximas secciones 5.3.2, 5.3.3, 5.3.4 y 5.3.5 describiremos en detalle los diferentes componentes del algoritmo evolutivo que aquí denominamos TACP-GA (TACP basado en *genetic algorithm*).

5.3.2. Representación y Codificación

Probablemente el problema y punto mas importante a la hora de diseñar un algoritmo evolutivo es la representación y codificación de las soluciones en el población. En general, la codificación se denomina *genotipo* mientras que solución representada *fenotipo* de acuerdo a la literatura [117]. Una correcta codificación puede tener un impacto significativo en el rendimiento del esquema metaheurístico por lo que se requiere de un extremo cuidado y prolijidad en su determinación. Por otro lado, una buena codificación debe permitir determinar la función objetivo asociada a ese genotipo y la manipulación del mismo (cruzamientos y mutaciones) sin mayores inconvenientes. En esta sección detallamos la estrategia de codificación y representación de los planes de contactos en cada uno de los individuos de la población.

En nuestro caso particular, las soluciones representan posibles planes de contactos que satisfagan las restricciones de recursos y al mismo tiempo optimicen algún criterio de optimalidad. Como se estableció anteriormente en la sección 5.2.2, nuestras principales variables de decisión son el conjunto de $\{Y_{k,i,j}\}$, las cuales representan la selección de interfaces entre los nodos i y j en cada estado k del intervalo de topología. Estas variables resultan

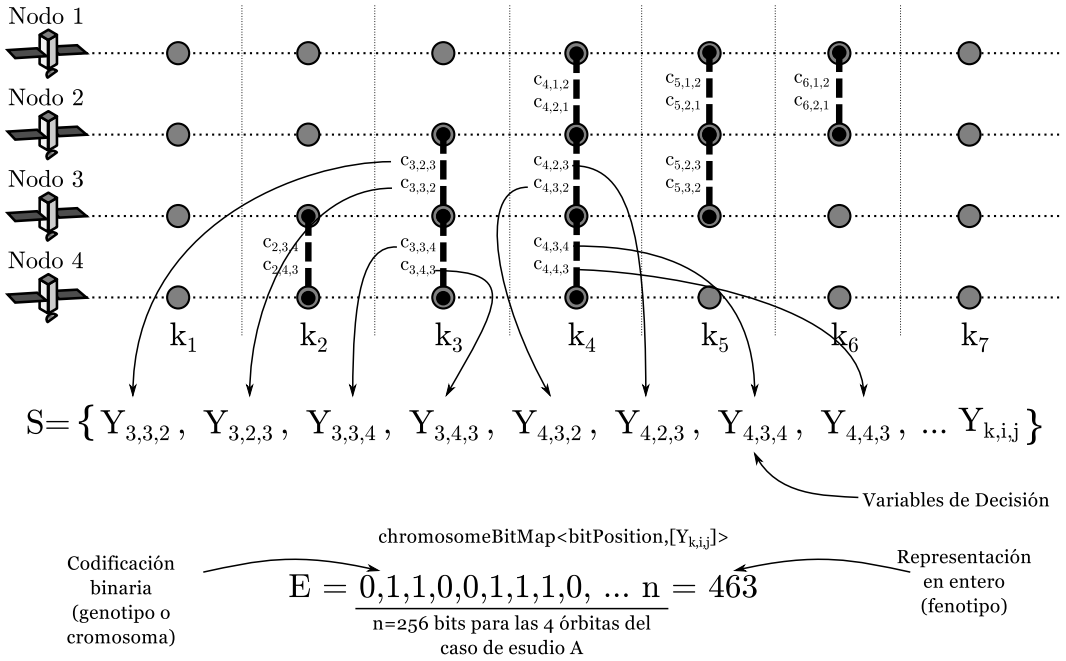


FIGURA 5.7: Representación y codificación de individuos en TACP-GA

de naturaleza binaria y esencialmente codifican si un contacto *i-j* deberá ser utilizado ($Y_{k,i,j} = 1$) o no ($Y_{k,i,j} = 0$). En consecuencia, $S = \{Y_{1,1,2}, Y_{1,1,3}, Y_{1,2,1}, ...Y_{k,i,j}\} \quad \forall i, j, k$ permite en efecto derivar las demás variables del modelo MILP ($X^{y,z}_{k,j,i}$ y $B^{y,z}_{k,i}$) determinando unívocamente una decisión de plan de contacto para el problema de diseño (CPD), aunque no necesariamente factible.

De esta manera, definimos una estructura basada en el conjunto $S = \{Y_{k,i,j}\}$ que pueda utilizarlas para conformar el *genotipo* del problema evolutivo de una manera directa. En particular, la *codificación binaria* es un mecanismo clásico y popular para codificar problemas del tipo toma de decisiones como el famoso problema de la mochila o del agente viajante [117] (del tipo si-no). En esta aproximación se genera un tren de 1s y 0s representando la decisión de si o no de una variable en particular del problema. La Figura 5.7 ilustra la propuesta de codificación binaria aplicada al problema de diseño de plan de contactos detallando el correspondiente genotipo y fenotipo de acuerdo a la clasificación de la literatura [127], donde cada bit del cromosoma representa una decisión de un arco de la topología de contacto. Con el fin de reducir el espacio de soluciones, sólo codificamos en el cromosoma E las variables $Y_{k,i,j}$ directamente involucradas en una toma de decisión. Por ejemplo, en la Figura 5.7 no hay necesidad de incluir la variable $Y_{2,3,4}$ dado que la misma no impacta en la elección o no de un arco dado que no incurre en una violación de las restricciones de recursos. Por último, con el fin de ayudar a la representación de la solución codificada en el cromosoma E, incorporamos un mapa denominado $chromosomeBitMap < bitPosition, [Yk, i, j] >$ que relaciona cada posición binaria en E con el correspondiente arco en la topología de contacto. Este último

resulta esencial en nuestra implementación dado que en la misma decodificamos todos los cromosomas de la población en cada iteración para someterlos a una evaluación de enrutamiento para determinar su función objetivo como describimos a continuación en la sección 5.3.3.

Finalmente, es necesario aclarar que esta estrategia de codificación es simple y sencilla de operar y modificar, pero peca de que su alcance es demasiado amplio permitiendo codificar y representar soluciones que resultan no factibles dadas las condiciones de limitación de recursos. Por ejemplo, se puede pensar en un individuo con un cromosoma E representado el conjunto de variables de decisión donde $Y_{3,3,4} = 1$ y $Y_{3,3,2} = 1$, la cual no respeta la cota de una interfaz por nodo. En consecuencia, se adopta una técnica específica para el problema de CPD de control de restricciones que describimos en la sección 5.3.4. Como resultado de esta codificación, un total de 256 bits son necesarios para representar la totalidad de las variables de decisión de las 4 órbitas del caso de estudio A (sección 2.3.1) derivando en un rango fenotípico de 0 a $1{,}1579209e + 77$.

5.3.3. Función Objetivo y Aptitud

Con el fin de determinar las posibilidades de supervivencia de los individuos a medida que evolucionan por medio de los operadores de cruzamiento y mutaciones, se utiliza el parámetro de aptitud o *fitness*. En efecto, este valor dependerá de la función objetivo del problema específico en cuestión. En general, los operadores de mutación y cruzamiento se aplican sobre los genotipos (representación binaria del cromosoma) mientras que el fitness se enfoca en el fenotipo del individuo en estudio [117]. Sin embargo, en nuestra aplicación particular, hacemos que el fitness dependa de la métrica de mejor tiempo de entrega (o *best delivery time* en Inglés) del plan de contacto correspondiente a ese individuo.

En particular, mientras menor es el tiempo de entrega, mayor la capacidad de supervivencia del individuo, de manera de poder ir mejorando la población del algoritmo. En consecuencia, definimos aptitud como la inversa de la función objetivo. Además, esta última debe contemplar un parámetro extra que represente o mida la cantidad de tráfico que no ha podido ser entregado para el plan de contacto en cuestión. Este parámetro, denominado $UndeliveredTraffic$ tiene un impacto significativo en la calidad del plan de contacto por lo que deberá penalizar fuertemente a la función objetivo de manera tal que una solución con este tráfico fuera de banda minimice sus posibilidades de supervivencia. Esta formulación implica una función objetivo diferente (de mas alto nivel) que la planteada en el modelo de la sección 5.2.2.2 (ecuación (5.1)). La formalización de la

función objetivo para el algoritmo evolutivo de CPD basado en tráfico se puede observar en la ecuación (5.10) y el fitness derivado de la misma en la ecuación (5.11).

$$ObjectiveFunc(S) = BestDeliveryTime + UndeliveredTraffic^2 \tag{5.10}$$

$$Fitness(S) = 1/ObjectiveFunc(S) \tag{5.11}$$

Para lograr obtener la métrica $BestDeliveryTime$, el plan de contacto debe ser evaluado por algún esquema de enrutamiento que permite determinar el flujo de tráfico en el plan resultante. Para ese fin, podremos reutilizar el planteo LP tomado de las ecuaciones (5.1) a (5.7) que conforman un problema lineal continuo (no entero al prescindir de las variables $Y_{k,i,j}$), el cual se puede resolver eficientemente por mecanismos existentes basados en la técnica *simplex* [117]. Esto es posible dado que la cualidad de entero era solamente impuesta por las variables de decisión $Y_{k,i,j}$ las que cuales no son necesarias para el calculo de enrutamiento específicamente, dado que la decisión de los arcos ya está representada en la codificación del cromosoma. En efecto, la principal ventaja del presente planteo de algoritmo evolutivo radica en la posibilidad de liberarse de un problema lineal entero (MILP) por varias resoluciones de un problema lineal continuo (LP) el cual resulta sumamente mas sencillo de resolver en tiempos acotados.

Finalmente, es importante observar que a pesar de que la función objetivo difiere de la formulación teórica de la ecuación (5.1) en la sección 5.2.2.2, ambas buscan el mismo objetivo de entregar la máxima cantidad de tráfico posible en el menor tiempo de topología. De hecho, el modelo teórico evaluado en TACP se evalúa en términos de tiempo de entrega.

5.3.4. Gestión de Restricciones

De acuerdo a lo previamente explicado en la sección 5.3.2,las soluciones codificadas no necesariamente representan planes de contactos que satisfagan las restricciones de recursos que limitan el uso de interfaces o contactos simultáneos. Por otro lado, también puede pasar que el plan de contacto tampoco respete la bi-direccionalidad de la elección de los arcos como se explica a continuación. En consecuencia, tanto la población inicial (generada aleatoriamente) así como la generación de nuevos individuos producto de los operados de mutación y cruzamiento, deben ser analizados y eventualmente modificados para que respeten las condiciones de contorno del problema.

En general, estas técnicas se conocen como gestión de restricciones o *constraints handling* en Inglés, entre las cuales las técnicas de reparación o *repairing strategies* en Inglés son una elección popular para los problemas de toma de decisiones codificados con el

esquema binario [117] tal como diseño de plan de contacto aquí planteado. En particular, estas estrategias de reparación cumplen la tarea de transformar (es decir, reparar) una solución no factible en una factible. En otras palabras, el procedimiento de reparación se aplicará a aquellos individuos cuyos cromosomas codifiquen planes de contactos que violen las condiciones de recursos de interfaces. En nuestro caso particular, la formulación de algoritmo genético y su correspondiente codificación y representación requieren de un esquema de reparación que generen cromosomas que respeten la bi-direccionalidad de la elección de arcos (si $Y_{k,i,j} = 1$, entonces $Y_{k,j,i} = 1$ y viceversa) y además detecten posibles violaciones a los límites puertos (es decir $\sum_{i}^{N} Y_{k,i,j} \leq maxInterfaces[i]$). La Figura 5.8 detalla precisamente las tareas necesarias para ejecutar este procedimiento.

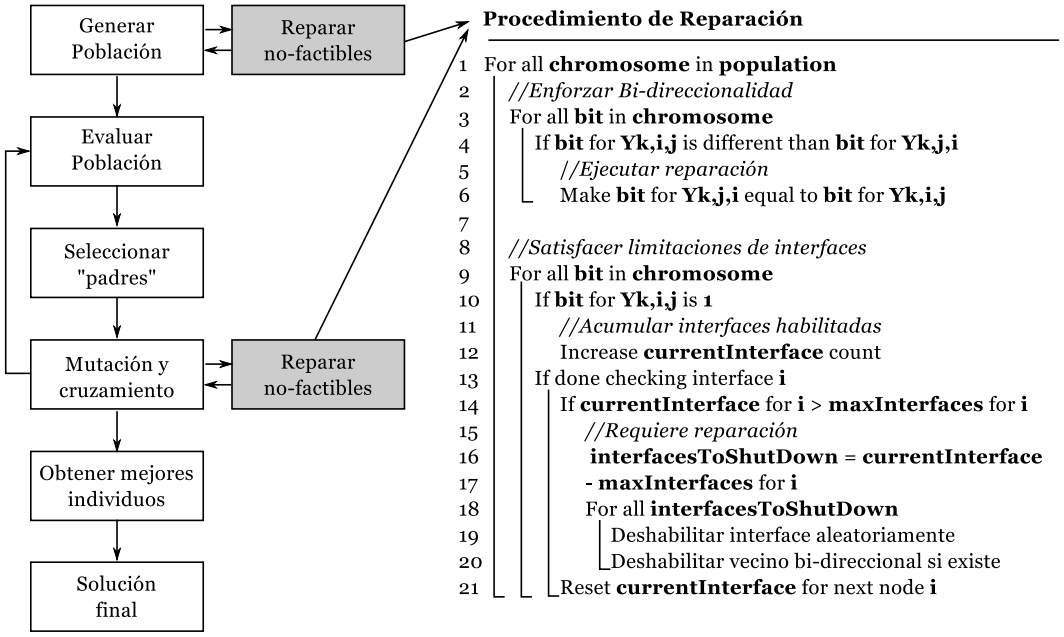


FIGURA 5.8: Estrategia de reparación para el diseño de plan de contacto basado en tráfico

En general, la estrategia de reparación propuesta es relativamente directa y repasa toda la población con el fin de reparar cromosomas no factibles. En las líneas 1 a 6 de la Figure 5.8, todos los bits de la codificación son evaluados en búsqueda de que las correspondencias de vecinos bi-direccionales se cumpla. Esta búsqueda debe ser condicionada dado que como se explicó en la sección 5.3.2, no todas las variables $Y_{k,i,j}$ forman parte de la codificación del cromosoma. En caso de encontrar una inconsistencia, se igualan los dos arcos de ida y vuelta correspondientes de acuerdo al valor del primero encontrado. Luego, desde las líneas 8 a 21 de la misma figura, el proceso de reparación deshabilita las interfaces en caso de que las mismas generen una violación de las restricciones. En particular, para cada cromosoma, y para todos los arcos de cada nodo i, los puertos implementados se acumulan en $currentInterface$ entre las líneas 10 a 12. Si el resultado final de la misma es mayor que el permitido (en la matriz $[P]$), se activa el procedimiento

de reparación especificado en las líneas 16 a 20, donde la cantidad de interfaces a desactivar en $interfacesToShutDown$ se apagan aleatoriamente. En efecto, cuando cada una se apaga, el arco vecino correspondiente (si es que existe en el cromosoma), también es desactivado para mantener coherencia con el planteo de bi-direccionalidad.

Finalmente, y como resultado, cada vez que se generan nuevas soluciones (tanto en la generación de la población inicial, como luego de la aplicación de cruzamientos y mutaciones), esta técnica de reparación nos garantiza que las mismas siempre serán factibles y validas representaciones de planes de contactos para el sistema DTN evaluado.

5.3.5. Inicialización, Selección, Reemplazo, y Criterio de Parada

Una vez que la representación, función objetivo y la gestión de restricciones fueron discutidas, quedan pendientes de concretar otras características menores para completar la descripción de funcionamiento del algoritmo evolutivo propuesto para el diseño de plan de contactos basado en tráfico. En efecto, el flujo general del mismo se ilustra en la izquierda de la Figura 5.8 cuya descripción se completará en esta sección.

En el primer paso del esquema, la población inicial debe ser generada. En particular, adoptamos una tamaño de 20 individuos el cual es producto de una recomendación general para este tipo de formulaciones evolutivas [117]. A pesar de que existen diferentes estrategias, en esta primera propuesta adoptamos el mecanismo de *inicialización aleatoria* o *random initialization* en la que todas las variables de decisión en el conjunto binario E se distribuyen uniformemente bit a bit. En consecuencia, es necesario aplicar una técnica de reparación (detallada en la sección 5.3.4) para garantizar de que la misma termine cumpliendo con la característica de las restricciones del sistema (bi-direccionalidad y límites de interfaces).

Luego de aplicar la reparación, la población debe ser evaluada por medio de la aplicación de mecanismos de enrutamiento al plan de contacto representada por cada uno de los individuos. Esto permitirá en efecto determinar el tiempo de entrega (*delivery time*) y el valor de aptitud descrito en la sección 5.3.3. Con el parámetro de fitness derivado, la probabilidad de *selección* de un individuo p es obtenida por medio de a división de este valor entre la totalidad de los que pertenecen a la población P como se especifica en la ecuación (5.12). Luego, esta probabilidad $pSelection_p$ es sometida a una elección del tipo ruleta o *roulette wheel selection* como se lo conoce en la literatura, la cual es la mas utilizada en este tipo de algoritmos evolutivos [117]. En esta, una ruleta es girada independientemente para cada uno de los individuos elegidos como padres (o *parents* en Inglés). Una vez que la asignación de padres es completada, el operador de cruzamiento es aplicado entre ellos de acuerdo a una probabilidad de cruzamiento o *probability of*

crossover en Inglés (pCr). En pocas palabras, el parámetro pCr determina si un par dado de los individuos p ya elegidos pueden ser considerados para el cruzamiento. Por otro lado, la mutación es aplicada directamente entre cada generación de acuerdo al parámetro de probabilidad de mutación (pMt).

$$pSelection_p = \frac{fitness_p}{\sum_{n=1}^{P} fitness_n} \tag{5.12}$$

Una vez que los operadores de mutación y cruzamiento se aplicaron, se debe seguir una política de reemplazo (*replacement strategy* en la literatura) que permita determinar cuales de los nuevos individuos reemplazarán a los ya existentes en la población de la anterior generación. En general estas estrategias implican mecanismos igualitarios y justos entre los cuales se considera la supervivencia de individuos no óptimos (es decir con bajo valor de fitness) cuyos resultado de la función objetivo no son necesariamente satisfactorios para el problema. El argumento ante este comportamiento yace en que un individuo poco apto puede, luego de un cruzamiento derivar en uno mas apto en el futuro. Sin embargo en nuestra formulación en particular de EA, la combinación del mecanismo de codificación y las técnicas de reparación proveen suficiente capacidad de exploración al generar nuevas soluciones con modificaciones drásticas y aleatorias. En la Figura 5.9 demostramos este planteo al observar que un reemplazo del tipo *elitista* provee en promedio mejores desempeños del algoritmo TACP-GA en comparación de otros mecanismos mas aleatorios. Esta prueba se basa en una configuración con los mejores parámetros obtenidos en el procedimiento de calibración que describiremos en la sección 5.4.3.

Finalmente, respecto al criterio de parada (o *stopping criteria* en Inglés), adoptamos una estrategia de finalización estática en la que el algoritmo termina de iterar de acuerdo a un número de $maxIter = 30$ fijas en nuestro caso. Particularmente, en la sección 5.4.3 mostramos que este número de 30 es suficiente para entregar buenas soluciones en un tiempo suficientemente acotado. Una vez que las 30 iteraciones se completan, el mejor individuo de los 20 totales de la población es rescatado y ofrecido como solución final de TACP-GA.

5.4. Análisis de Plan de Contacto Basado en Tráfico

En esta sección analizaremos los esquemas basados en tráfico inicialmente comparando el desempeño del modelo teórico de TACP-LP descrito en la sección 5.2.2 con los esquemas de FCP [2] y RACP [4] (basado en el algoritmo de recocido simulado). En esta

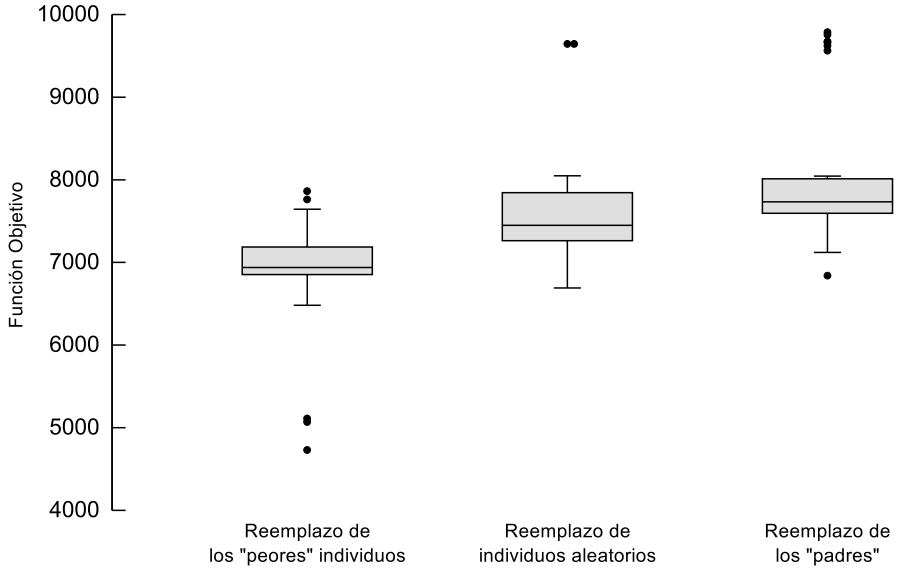


FIGURA 5.9: Función objetivo obtenida para diferentes estrategias de reemplazo para $iter = 30$, $pCr = 0{,}6$, and $pMt = 0{,}1$

etapa de análisis inicial de la sección 5.4.2 obtendremos los resultados óptimos que se pueden esperar de TACP, los que luego, en la sección 5.4.3, utilizaremos para calibrar los parámetros de su formulación metaheurística (TACP-GA) descrita en la sección 5.3.1. Para realizar este análisis, primero retomaremos el caso de estudio y referencia A (topología en escalera) descrito en la sección 2.3.1 del capítulo 2 al cual completaremos con la información del tráfico a considerar en la siguiente sección 5.4.1.1.

En general, asumiremos que el algoritmo de recocido simulado de RACP se configura con un total de 10000 iteraciones con una máxima temperatura de 10000 y un criterio de optimización de rutas para un patrón todos contra todos. Por otro lado, configuramos $w(t_k) = K * t_k^2$ para el modelo teórico de TACP-LP el cual fue discutido en la sección 5.2.2.2 con el fin de generar planes de contactos que entreguen el tráfico en el menor tiempo posible sin importar el número de arcos activados para lograrlo. Sin embargo, este parámetro no es de relevancia en el caso de estudio y referencia A, dado que para el tráfico planteado en la próxima sección 5.4.1.1 sólo se puede usar el camino N_4 a N_3 a N_2 a N_1.

5.4.1. Configuración del Escenario

5.4.1.1. Patrón y Características del tráfico

En esta sección retomamos el caso de referencia y estudio A detallado en la sección 2.3.1 del capítulo 2. La misma contaba de una topología de 4 satélites en formación *escalera*

evaluados en un tiempo de topología de 3 horas 22 minutos y 36 segundos permitiendo un total de 4 vuelos por sobre los polos (uno cada 48 minutos).

En la misma se aplica una fragmentación de estados para todos aquellos k_n con una duración mayor a 500 segundos ($i_k > 500$) derivando en que el estado k_4 se subdivida en k_{4a}, k_{4b}, y k_{4c} con $i_{4a} = 500s$, $i_{4b} = 500s$, y $i_{4c} = 458s$ de acuerdo a los mostrado en la Figura 5.10. En efecto, el mismo resultado se obtiene para las siguientes 4 iteraciones en los estados k_{10}, k_{16} y k_{22} en la parte superior de la misma figura.

Con el fin de evaluar y comparar los diferentes procedimientos de diseño de plan de contacto, debemos especificar un modelo de tráfico como mecanismo de enrutamiento de datos a través de los planes de contactos resultantes. Para esto rehusamos el planteo del problema de *multi-commodity flow* explicado en la sección 5.2.2 (sin las ecuaciones de decisiones (5.8) y (5.9)). En efecto, una asignación óptima de los flujos se puede obtener del uso de este planteo sobre el plan de contacto a evaluar. Como discutiremos en el capítulo 6, este esquema de enrutamiento es optimista dado que se basa en una visión

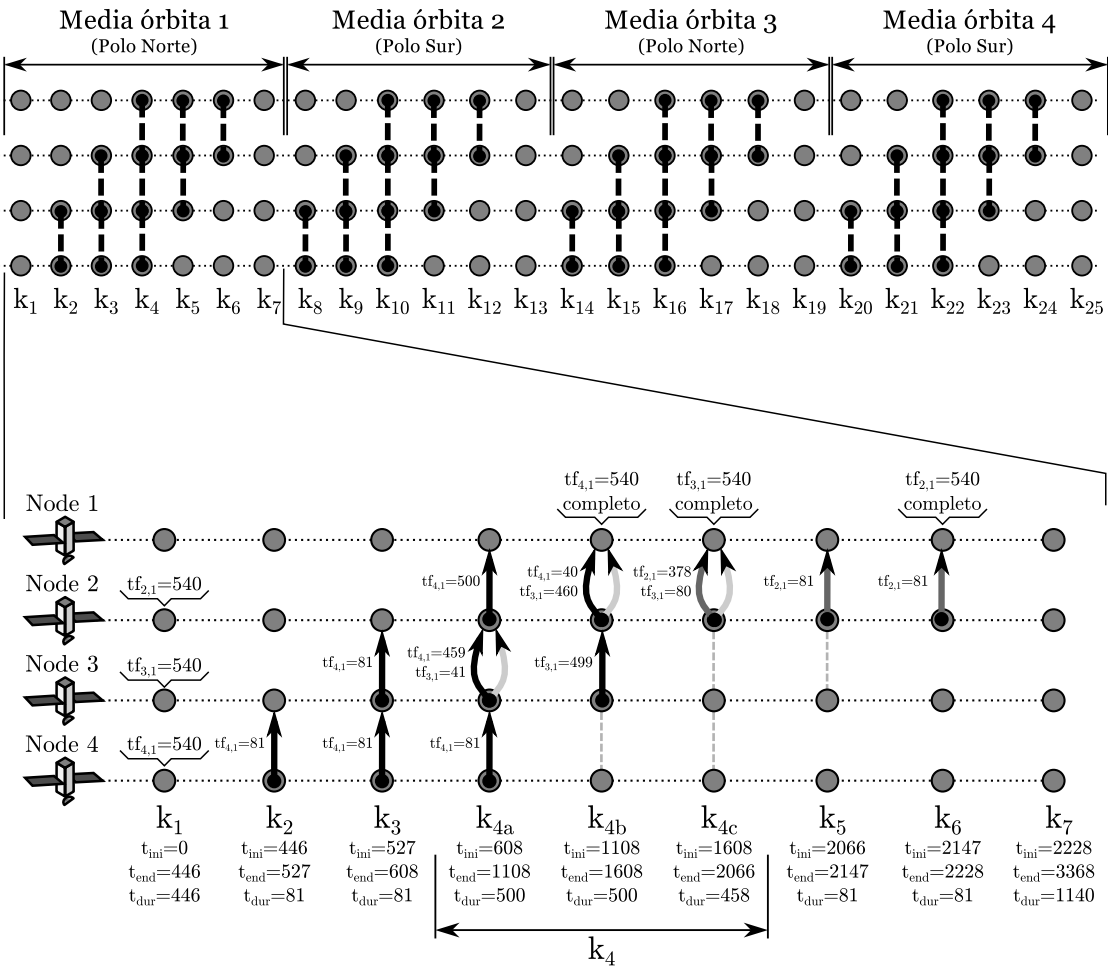


FIGURA 5.10: Topología de contacto y tráfico resultante con TACP-LP para $\rho = 1$

global del sistema la cual difícilmente se pueda tener en cada nodo de un escenario distribuido en el cual los contactos se dan de manera esporádica. En general, esto último es un problema específico del enrutamiento específico de DTN que deriva en retransmisiones de tráfico que impactan en las prestaciones finales del sistema [9] como discutiremos en el capítulo 6.

Tener un modelo de tráfico nos permite determinar el comportamiento del mismo en el sistema. A continuación pasamos a describir los fuentes de tráfico y sus correspondientes destinos en el caso de estudio y referencia A. Configuramos que los nodos N_2, N_3 y N_4 generen la misma cantidad de tráfico en primer estado de la topología (k_1) para luego fluir hacia el nodo destino N_1 el que se espera luego pueda descargar los datos de toda la red a tierra por medio de una comunicación espacio-tierra (ESL) de alta velocidad no considerada en esta topología por razones de simplicidad. Este patrón de tráfico simula un adquisición de imágenes en la zona ecuatorial de manera independiente pero sincronizada entre todos los nodos del sistema.

En particular, establecemos que las comunicaciones del caso de estudio A se limitan a una por segmento de vuelo con una tasa de transmisión máxima de $1Mbps$ del tipo full-duplex (bi-direccional) para un rango máximo de $700Km$. Con el fin simplificar el análisis asumiremos que los bundles (paquetes) transmitidos tienen un tamaño de $1Mbit$ o $125KByte$ los que efectivamente ocupan el canal por el tiempo de 1 segundo a la tasa de datos especificada. En general, y como se explica en la sección 2.2.1.1, este último parámetro jamás podrá ser determinístico en un esquema de acceso al medio compartido debido a su naturaleza estocástica. Por otro lado, en este caso y al igual que los capítulos anteriores asumimos que el tiempo de propagación del paquete desde el nodo transmisor al destino es despreciable (es decir no contemplamos demoras). Por último, variaremos la carga de tráfico desde $\rho = 1$ ($540Mbit$, $67{,}5MByte$ o 540 paquetes por nodo) a $\rho = 0{,}1$ ($54Mbit$, $6{,}75MByte$ o 54 paquetes por nodo), donde $\rho = 1$ es la carga del sistema que satura la topología de contactos (sin restricciones) como explicamos a continuación.

El flujo de saturación para la primer órbita de la topología de contacto (CT) se ilustra en la Figura 5.10 donde una asignación optima del mismo permite la entrega de un total de 540 paquetes de N_2, N_3 y N_4 a N_1. En la figura, cada flujo de N_{src} a N_{dst} se mide en cantidad de paquetes (que es equivalente a segundos a la tasa de datos de $1Mbps$) y unívocamente identificados como $tf_{N_{src},N_{dst}}$. Vale aclarar que no se puede enrutar mas datos a N_1 en esta topología de contacto (durante la 1er media órbita) dado que los contactos entre k_4 y k_6 tienen una duración de 1620 segundos el cual es el límite para transmitir un total de $540 \times 3 = 1620$ paquetes de 1 segundo de duración cada uno. En efecto, a pesar de que otros arcos como N_4 a N_3 (k_{4b} y k_{4c}) y N_3 a N_2 (k_{4c} y k_5)

permanezcan sin uso, esta asignación del flujo representa la saturación del sistema como se lo plantea.

Es importante destacar que este desempeño de media órbita sólo es factible cuando las restricciones de recursos no son consideradas dado que los nodos N_2 y N_3 transmiten y reciben datos simultáneamente en varios estados (k_3, k_{4a}, y k_{4b}). Si la restricción de una sola interfaz por satélite, el tiempo de entrega se pospondrá hasta que nuevos contactos puedan ser establecidos en las próximas órbitas para que el tráfico pueda llegar a su destino.

5.4.1.2. Métricas de Evaluación

Con el fin de comparar los diferentes esquemas de diseño de planes de contactos, la cantidad de tráfico finalmente entregado al destino final (*delivery ratio*) resulta probablemente la métrica mas relevante. Sin embargo, si el intervalo de topología es lo suficientemente largo (varias órbitas), podemos asegurar que el tráfico se entregará totalmente. En efecto, resulta necesario medir y comprender cómo fue que dicho tráfico llegó al destino, el cual a su vez, corresponde con la eficiencia con la cual se diseño el plan de contacto que se utilizó para enrutar el mismo.

Para medir esta eficiencia, observaremos el tiempo total de contacto utilizado hasta que el tráfico se entregue completamente al destino. En otras palabras, resulta importante la suma de todos los contactos del sistema acumulados hasta el momento de tiempo en el cual el tráfico fue completamente recibido en el destino. Llamaremos a esta métrica tiempo de contacto de sistema o *system contact time* en Inglés. En general, es deseable tener un tiempo de contacto de sistema tan bajo como sea posible.

Por otro lado, evidentemente estudiaremos el tiempo en el cual el tráfico se entregó completamente al nodo destino (N_1). En este caso donde el trafico se genera en el inicio de la topología, el tiempo de entrega corresponde con la demora de entrega. Llamaremos esta métrica tiempo de entrega o *delivery time* en Inglés. En efecto, mientras menor la métrica de mejor calidad el plan de contacto.

5.4.2. Análisis del Modelo Teórico

En esta sección, evaluaremos y compararemos los algoritmos FCP derivado en el capítulo 3 (basado en el algoritmo Blossom), RACP en el capítulo 4 (basado en recocido simulado) y el modelo teórico de TACP-LP descrito en la sección 5.2.2. También tendremos en cuenta la topología de contacto (o *contact topology* o CT en Inglés) como cota superior de desempeño. La comparación se realizará en el escenario generado por el caso

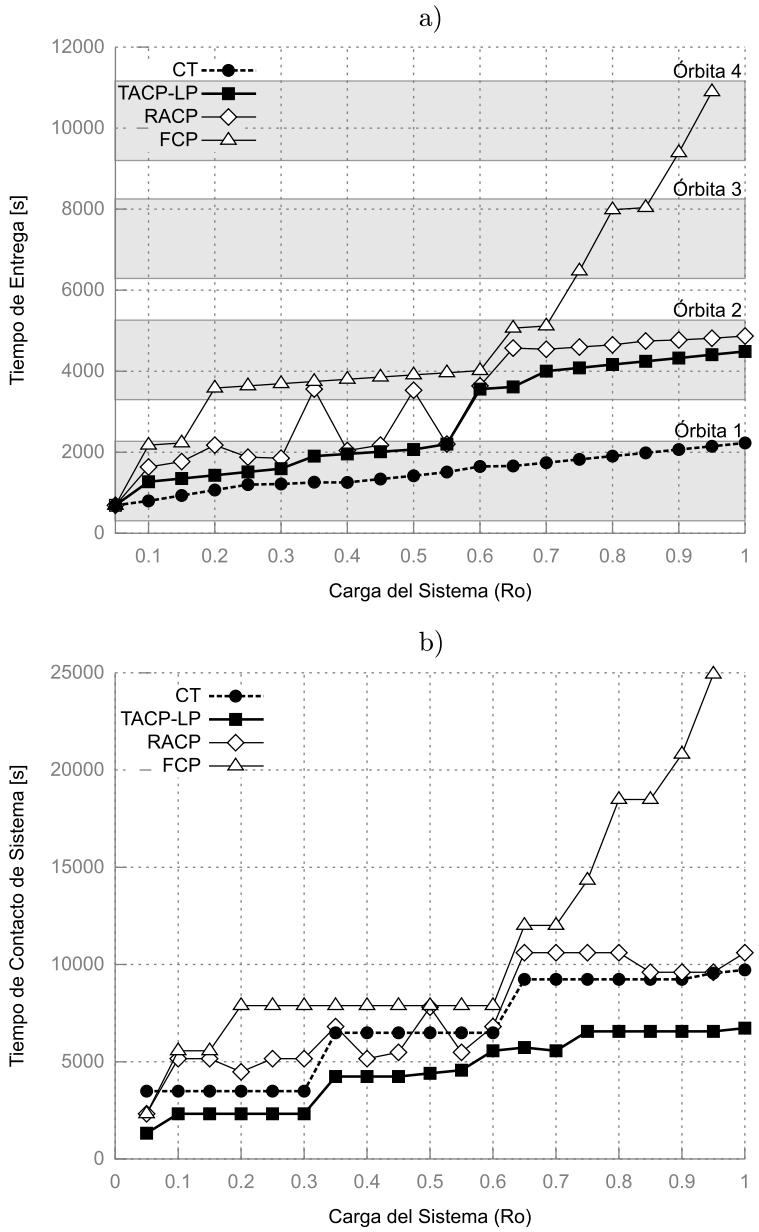


FIGURA 5.11: Tiempo de entrega en a) y tiempo de contacto de sistema en b) para diferentes ρ del caso de estudio

de estudio y referencia A cuyos detalles físicos se dan en la sección 2.3.1, patrones de tráfico a analizar en 5.4.1.1 y métricas en 5.4.1.2.

En particular estudiaremos las métricas de *delivery time*, y *system contact time* para una variación de carga del sistema de $0 \leq \rho \leq 1$ para todos los esquemas evaluados. Los resultados de estos análisis se ilustran y resumen en la Figura 5.11. Entre estas, el tiempo de entrega mostrado en la Figura 5.11 a) contiene demarcado las oportunidades de comunicaciones que se da en la zona de los polos. En efecto, el tráfico siempre será entregado al destino (N_1) en estos períodos dado que fuera de estas regiones no hay posibilidades de transferencia de datos. En otras palabras, si en una de estas franjas

no se puede completar la transmisión de los datos, la misma será pospuesta hasta la siguiente pasada por los polos y así sucesivamente hasta lograr la entrega final.

Como se planificó en la sección 5.4.1.1, el tiempo de entrega del tráfico para la topología de contacto (CT) mostrado en la Figura 5.11 a) aumenta hasta un total de $2228s$ para $\rho = 1$, el cual es precisamente el tiempo en el cual el estado k_6 (y el primer período de media órbita) terminan (ver Figura 5.10). En la gráfica de tiempo de contacto de sistema, el plan de contacto de CT muestra un incremento de esta métrica a medida que el tráfico necesita de un nuevo estado para enviar los datos. En efecto, vale aclarar que esta medición sólo considera todos los arcos activos hasta que la entrega total del tráfico está completo. Por ejemplo, para $\rho = 0{,}3$ la entrega se completa a los $1094s$ el cual se encuentra entre el estado k_4, mientras que para $\rho = 0{,}35$ el mismo se da a los $1175s$ que corresponde al estado k_5. En esta evolución de ρ un nuevo arco es requerido para evacuar el flujo de tráfico, y como el caso de CT no sólo tiene los contactos entre N_2 a N_1 activos sino que todos los demás (N_4 a N_3 y N_3 a N_2), el tiempo total de contacto del sistema aumenta drásticamente.

En general, enrutar datos usando la topología de contacto (sin restricciones) siempre aporta el mejor tiempo de entrega al contar con la posibilidad de utilizar todos los contactos físicamente posible. Sin embargo, esto implica que varios de ellos permanecerán activos sin ser usados por el tráfico dejando un rendimiento pobre en término de uso de recursos de sistema. Mas aún, implementar directamente la topología de contacto puede resultar imposible dado las condiciones de recursos o limitaciones arquitecturales del sistema como ya se trató en la sección 2.4.

Al considerar el modelo TACP-LP como fue planteado en la sección 5.2.2, se obtiene un tiempo de contacto de sistema totalmente optimizado dado que cada arco que se decide activo es utilizado al 100 % en plan de contacto final. Todos los arcos que no son necesarios, se desactivan por lo que ninguno de ellos permanece inutilizado luego de aplicar los flujos de tráfico a los mismos. Esto permite a TACP-LP ser el esquema as efectivo en términos de utilización de recursos de sistema, pero como comentaremos en la sección 5.4.2.1, este procedimiento puede resultar contraproducente dado que no deja margen de capacidad en el plan de contacto final. En efecto, cualquier error de predicción de contacto o de planificación de tráfico se vería significativamente afectado en este caso.

En cuanto al tiempo de entrega (Figura 5.11 a)), TACP-LP no es capaz de aportar el rendimiento que da la topología de contacto sin restricciones de interfaces. Esto es evidente dado que CT es capaz de utilizar varias comunicaciones por nodo simultáneas como se muestra en la Figura 5.10, mientras que TACP-LP, RACP, y FCP se diseñan y configuran para nunca utilizar mas de un arco por nodo a lo largo de todo el período

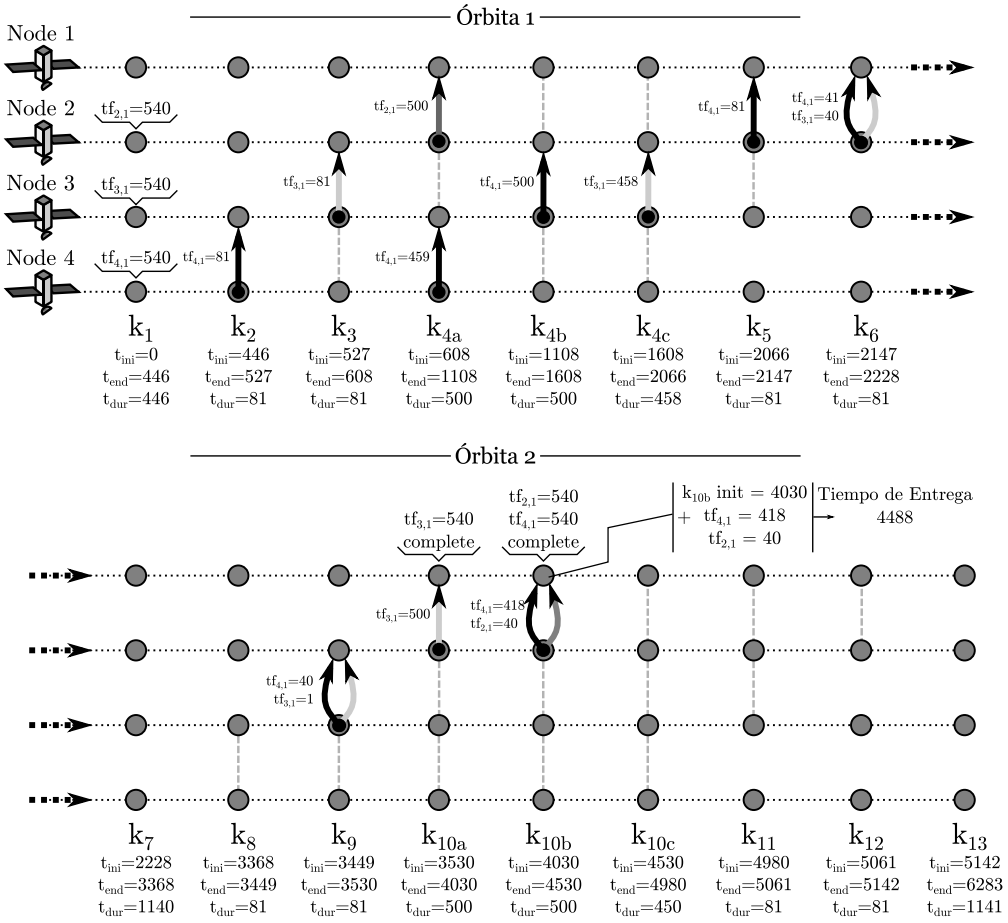


FIGURA 5.12: Plan de contacto diseñado con TACP-LP para $\rho = 1$

topológico. Sin embargo, entre todos estos esquemas, TACP aporta el menor tiempo de entrega para la variación total de la carga ρ. En particular, el plan de contacto generado para la carga $\rho = 1$ se ilustra en la Figura 5.12 para el cual el tráfico se puede entregar a los 4488 segundos en el estado k_{10b} del segundo período orbital. Es interesante notar que en esta figura ningún nodo utiliza mas de un arco en todo momento, y que dada esta condición, el planteo MILP de la sección 5.2.2 nos garantiza que esta es la solución optima en términos de tiempo de entrega o *delivery time*.

Es interesante observar que TACP-LP es seguido muy de cerca por RACP (diseño de plan de contacto basado en rutas) explicado en el capítulo 4. Dado que RACP implementa técnicas metaheurísticas (recocido simulado) para explorar el espacio de solución de la topología de contacto restringida, puede probabilísticamente encontrar un plan de contacto muy satisfactorio en términos de tiempo de estrega. En efecto, esta métrica era parte intrínseca del esquema RACP dado que el mismo buscaba minimizar el promedio de demora de las rutas para un patrón todos-contra-todos. De hecho, RACP mostró un comportamiento superior al esperado al generar planes de contactos con tiempos de entregas tan bajos como TACP (para algunos casos de ρ) sin la necesidad de explotar

el conocimiento del tráfico del sistema. Sin embargo, esta carencia de información hace que RACP deba activar mas arcos para favorecer un tráfico de todas las fuentes contra todos los posibles destinos. En efecto esto deriva en que el plan de contacto de RACP evidencie numerosos arcos sin utilizar lo que hace que su métrica de tiempo de contacto de sistema sea considerablemente mayor que TACP.

Finalmente, analizamos el esquema de menor información de entrada, pero probablemente el mas eficiente en términos de procesamiento computacional: FCP. En general, y gracias al uso de un algoritmo conocido como Blossom [105], FCP ciertamente tiene la delantera en términos de tiempo de procesamiento para resolver el plan de contacto. Sin embargo, dado que FCP sólo evalúa contactos basados en un criterio de salto único con el fin de optimizar una métrica de justicia de asignación [2], el plan de contacto que el mismo genera no aporta ningún beneficio particular a la especificidad de tráfico detallada en la sección 5.4.1.1. En consecuencia, el plan de contacto diseñado por FCP require de un período de tiempo significativamente mayor que los otros mecanismos para entregar el tráfico de todos los nodos al nodo N_1. En efecto, el mismo logra la entrega total de los datos recién en el cuarto período de media órbita del caso de estudio y referencia A, derivando en un significativo impacto del tiempo total de contacto de sistema requerido como se puede ver en la Figura 5.11 b).

Para resumir el análisis aquí mostrado, TACP-LP ha ofrecido un mejor rendimiento que el segundo esquema en términos de información utilizada (RACP) por un 58 % en utilización de los recursos de sistema, mientras que también mejora el tiempo de entrega por un %10 (para $\rho = 1$). Mas aún, esta mejora en rendimiento será mas y mas drástica para topologías de mayor tamaño y tiempo de evaluación dado que este fenómeno resulta acumulativo. En general, las métricas aquí mostradas resultan claras y contundentes y se corresponden con la hipótesis general de la tesis discutida en la sección 1.1 y el gráfico relativo mostrado en la Figura 2.12.

5.4.2.1. Tolerancia a Fallos e Imprecisiones

Como se mostró en la sección 5.4.2, el esquema TACP-LP fue capaz de proveer los mejores planes de contactos en término de tiempo de entrega y tiempo de contacto de sistema. Sin embargo, esto se logra al utilizar la información no solo de topología si no que de trafico esperado en el sistema, el cual se asume preciso y certero. En el caso de el mismo no sea de esta manera, las consecuencias a nivel rendimiento pueden resultar catastróficas al quedar información almacenada en la memoria de los nodos sin posibles contacto futuros por medio de cual enviarlos (a pesar de que físicamente exista posibilidad). Esto fenómeno no se da de esta manera en los esquemas FCP y RACP

dado que los mismos activan contactos hasta el final del intervalo topológico ofreciendo segundas oportunidades aunque empeorando las métricas de uso de recursos.

Dos posibles estrategias emergen como alternativa respecto al uso de TACP-LP (y TACP en general) en implementaciones reales. Por un lado se pueden considerar la inclusión de márgenes de error en la formulación de las planificaciones de tráfico. Es decir, que la misma se sobredimencione para que modele posibles datos extras no considerados, y que por otro lado se agreguen flujos adicionales en tiempos específicos para simular posibles retrasos en la generación del tráfico original.

Otra aproximación factible es resolver el sistema con TACP-LP hasta el tiempo de entrega de sistema, para luego, de ahí en adelante utilizar un esquema agnóstico al tráfico como FCP o RACP para completar la topología lógica con un uso equitativamente distribuido de los contactos y arcos remanentes. En general, dejamos el estudio y análisis de posibles combinaciones híbridas de FCP, RACP, y TACP como trabajos futuros de esta tesis.

5.4.3. Análisis del Algoritmo Genético

A pesar del rendimiento de TACP-LP mostrado en el análisis del modelo en la sección 5.4.2, el mismo resulta ser un modelo teórico basado en una programación lineal basada en variables enteras MILP cuya complejidad computacional puede resultar significativamente superior sobre todo para escenarios mas complejos que el caso de estudio y referencia A planteado en este análisis. En efecto, se propuso una alternativa algorítmica basada en formulaciones evolutivas en la sección 5.3.1 de este capítulo denominada TACP-GA. Sin embargo, antes de evaluar este último, los parámetros del mismo deben ser calibrados, y para lograr esto necesitamos de resultados de referencias como los mostrados en la sección 5.4.2.

Entre los parámetros de TACP-GA a calibrar destacamos a la probabilidad de cruzamiento (pCr), la probabilidad de mutación (pMt), y la cantidad de iteraciones ($iter$) del algoritmo. Las mismas pueden tener un impacto significativo en el comportamiento general del algoritmo por lo que resulta menester evaluar la mejor combinación entre las mismas de acuerdo a lo sugerido por la literatura [117]. En efecto, la misma recomienda un estudió paramétrico, exploratorio y empírico de los parámetros por medio del uso de métodos estadísticos generalmente representado por gráficos de cajas o *box-plots*. Esta aproximación de configuración se la conoce como estrategia *off-line* dado que los parámetros se determinan previamente a la ejecución del algoritmo. Otra alternativa es que los mismos se vayan calibrando a medida que avanza el procedimiento de búsqueda

pero requiere de un análisis mucho mas detallado del que se le da en esta implementación en particular.

Para realizar la calibración retomaremos el caso de estudio y referencia A de la sección previa cuyos detalles físicos se dan en la sección 2.3.1, patrones de tráfico a analizar en 5.4.1.1 y métricas en 5.4.1.2. Una vez calibrado los parámetros procederemos a comparar los resultados de TACP-GA con aquellos de TACP-LP (modelo teórico) en la sección 5.4.3.2.

5.4.3.1. Calibración de Parámetros

Una vez determinada la solución para el parámetro $\rho = 1$ (es decir 540 paquetes por nodo) ilustrada en la Figura 5.12 con valores de delivery time de 4488 segundos, procedemos a utilizar el algoritmo TACP-GA bajo el mismo entorno para estudiar que conjuntos de parámetros pCr y pMt aproximan mas la solución obtenida al valor óptimo entregado por el modelo teórico (TACP-LP). La cantidad de iteraciones óptimas serán discutidas en la próxima sección 5.4.3.2.

En efecto, evaluaremos de manera empírica como se sugiere en [117] un conjunto de parámetros para determinar cual de ellos le permite a TACP-GA entregar el mejor rendimiento para el diseño de planes de contactos basado en tráfico. Con este fin, realizamos un estudio estadístico de 1000 ejecuciones de TACP-GA para diferentes valores de probabilidad de cruzamiento o crossover (pCr) y mutación (pMt) de manera combinada ($pCr = 0{,}3$,$pMt = 0{,}01$), ($pCr = 0{,}3$,$pMt = 0{,}1$), ($pCr = 0{,}6$,$pMt = 0{,}01$) y

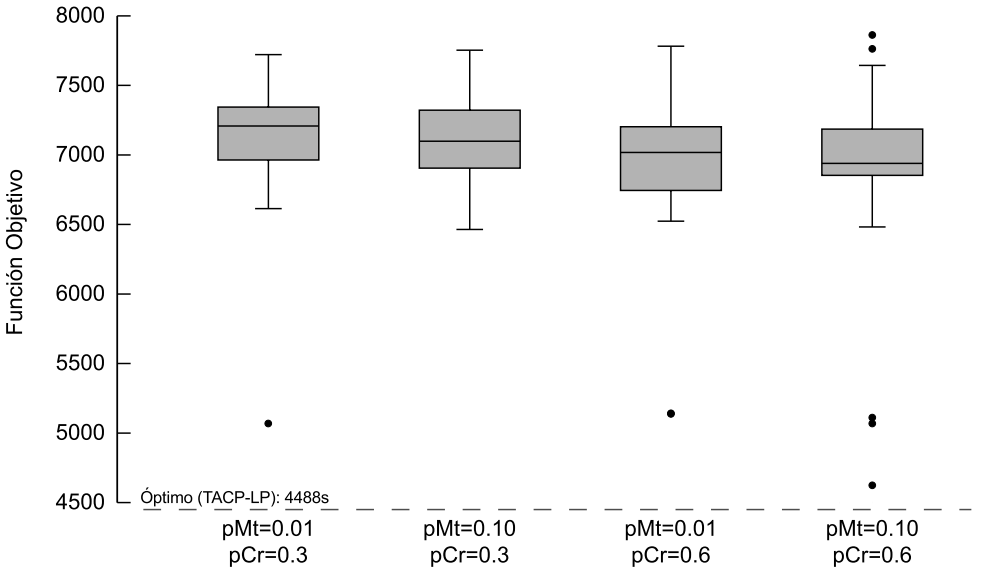


FIGURA 5.13: Box-plots para diferentes combinaciones de probabilidad de cruzamiento (pCr) y mutación (pMt)

Tabla 5.2: Tiempos de procesamiento de solvers de enteros mixtos (MIP)

MIP Solver	Tiempo de Ejecución
IBM-ILOG Solver	107s
GLPK Solver	18141s

($pCr = 0{,}6$,$pMt = 0{,}1$). Los resultados de estas numerosas simulaciones se entregan en el formato de *box-plot* en la Figura 5.13, las que dan una sensación de la mediana (línea interna a la caja), los cuartiles superiores e inferiores, y los valores fuera de límites ilustrados como puntos negros.

Del análisis previo se puede derivar que el conjunto de parámetros ($pCr = 0{,}6$, $pMt = 0{,}1$) entrega, en promedio los mejores valores de la función objetivo. La media exacta es 6926,64 segundos la cual puede resultar, *a priori*, aceptable al compararla con el óptimo de 4488 entregado por el modelo teórico TACP-LP. Por otro lado, de acuerdo las clasificaciones de la bibliografía [117], este conjunto de parámetros resulta mas robusto al tener los segundos y tercer cuartiles mucho mas próximos entre si que los otros esquemas. En efecto, esto indica que las soluciones entregadas tienden a estar mas cerca del promedio. Sin embargo el esquema arroja algunos valores excepcionales tanto para dar mejores resultados (mas cerca del óptimo) como de peores rendimientos.

En general, TACP-GA entrega este tipo de resultados relativamente satisfactorios pero en un mejor tiempo de ejecución. Como veremos en la próxima sección 5.4.3.2, TACP-LP requiere un tiempo computacional significativamente superior en solvers gratuitos en comparación con TACP-GA, lo que hace que las soluciones del mismo, si bien subóptimas, sean de extremo valor para la planificación de redes DTN mas complejas.

5.4.3.2. Discusión de Desempeño

Finalmente, en esta sección analizamos el desempeño en términos de complejidad y tiempo computacional de TACP-GA. En efecto, hemos sometido a los modelos teóricos y evolutivos a una comparación estadística sobre la misma plataforma de hardware para entender los esfuerzos de cálculo requerido por cada uno de ellos. Por disponibilidad se eligió una plataforma con un procesador Intel Core $i3$ ($2nd$ Gen) $2310M$ trabajando a $2{,}1GHz$ con $8GB$ de memoria DDR3 SDRAM a una frecuencia de acceso de $1333MHz$ sobre el cual ejecutamos un sistema operativo abierto y gratuito basado en Linux (Ubuntu).

La Tabla 5.2 enumera los tiempos de procesamiento requeridos para resolver el modelo MILP de TACP-LP para diferentes solvers: IBM-ILOG [126] (comercial) y el popular

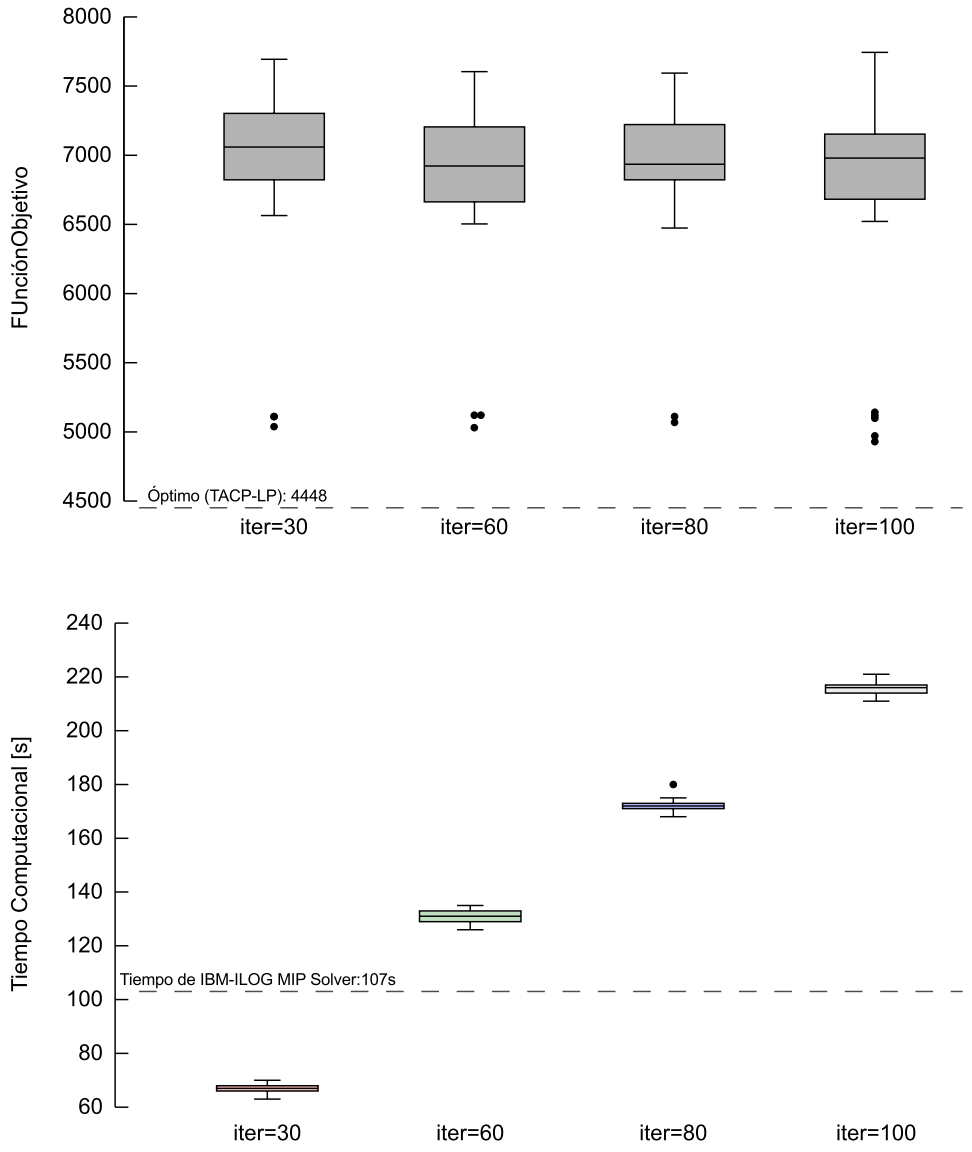


FIGURA 5.14: Función objetivo y tiempo de ejecución para diferentes iteraciones del algoritmo TACP-GA

GLPK [108] (gratuito y disponible). En general, se puede observar una amplia diferencia entre estos dos mecanismos de resolución en la que el software comercial provee desempeños notablemente superiores (incluye mecanismos de optimización específicos) aunque a un costo significativo dado que una licencia para el mismo suele superar los 1000 dólares estadounidenses. En particular, para realizar este trabajo se consiguió una licencia académica, la cual arrojó en solo 107 segundos el resultado óptimo, mientras que el GLPK se tomó mas de 5 horas para calcularlo.

Por otro lado, la Figura 5.14 ilustra el valor objetivo para diferentes valores de iteración o repetición (*iter*) del algoritmo genético y su correspondiente tiempo de ejecución. De este último se puede concluir que, a pesar de unos pocos resultados fuera de banda (*outliers*

en Inglés) ilustrados como puntos negros en el gráfico, el algoritmo resulta bastante insensible al incremento de la cantidad de iteraciones. En efecto, el exceso en la cantidad de las mismas resulta en un incremento importante del tiempo de cálculo computacional. Sin embargo, y en general, de todas las ejecuciones evaluadas, ninguna ni si quiera se compara con el tiempo de solución de 5 horas de GLPK, aunque resultan comparables con el solver comercial para este caso sencillo. Efectivamente, para casos mas complejos inclusive el software IBM-ILOG toma horas para converger mientras que el TACP-GA entrega soluciones aceptables en tiempos acotados. Sin embargo, resulta poco practico realizar estos análisis debido al tiempo de cómputo que requieren. En consecuencia, determinamos que con un valor de 30 iteraciones se pueden obtener buenas soluciones en un período de cómputo acotado para el caso de estudio y referencia A con el volumen de trafico planteado en la sección 5.4.1.1.

Finalmente, vale la pena enfatizar que el solver de IBM es capaz de entregar la solución óptima en 107 segundos para este caso particularmente simple. Sin embargo el tiempo de resolución de este tipo de problemas incrementa exponencialmente (respecto al número de satélites, estados, arcos, etc.) en contraste de un incremento controlado por e número de iteraciones como el caso de TACP-GA. En este contexto, se verifica que el planteo de TACP-GA es de suma utilidad para resolver el problema de diseño de plan de contactos basado en tráfico para redes satelitales DTN complejas en un tiempo acotado.

5.5. Comentarios Finales Sobre el CPD Basado en Tráfico

En este capítulo enfrentamos el desafío de generar un procedimiento de diseño de planes de contactos que utilicen la información de tráfico para determinar los planes de contactos óptimos para una red DTN de satélites. Este fue detallado en un reporte técnico [7] y está en evaluación para una revista especializada. Además este aporte teórico es el eje del capítulo 15 del libro “Wireless Sensor Systems for Extreme Environments: Space, Underwater, Underground and Industrial” de la editorial Wiley actualmente en prensa [6]. En efecto, a pesar de la complejidad asociada, demostramos que el uso de estos esquemas supera notablemente los anteriormente desarrollados (FCP en capítulo 3 y RACP en capítulo 4) demostrando ser de suma utilidad para la planificación de este tipo de redes con restricciones de recursos y arquitectura.

Debido a la complejidad que se debe incluir en el modelado teórico expresado por medio de un modelo MILP (llamado TACP-LP), propusimos una alternativa de algorítmica basada en otra rama de la metaheurística llamada algoritmos evolutivos. En efecto, propusimos TACP-GA, un mecanismo que mejora el uso de recursos computacionales

entregando soluciones sub-óptimas pero de valor para la necesidad general de planificación. Este esquema será publicado a fianles del 2015 en la conferencia IEEE International Conference on Wireless for Space and Extreme Environments (WiSEE) en Orlando, Florida [8].

Si bien el aporte realizado con TACP probablemente resulte el de mayor visibilidad a nivel publicaciones en libros, revistas y conferencias, el mismo supone que el sistema orbital será capaz de tomar decisiones de enrutamientos óptimas lo que no es necesariamente en el caso general. En efecto, en el próximo capítulo 6, analizamos los inconvenientes en el área de implementación de planes de contacto y gracias a la visión de TACP describimos un conjunto de aportes derivados pero no menos importante en esta área.

Capítulo 6

Implementación de Planes de Contacto

6.1. Introducción

A lo largo del capítulo 3, 4, y 5 hemos realizado una profunda exploración al problema del diseño de planes de contactos, para que luego, los nodos de la red puedan utilizarlos para tomar decisiones de enrutamiento apropiadas y eficientes. Sin embargo, en todos estos planteos, y particularmente en TACP, hemos asumido que los nodos tomarán decisiones de tráfico óptimas sobre dichos planes, lo cual no necesariamente es cierto como se insinuó en la sección 5.2.1 del capítulo 5. De hecho, de querer optimizar el sistema al máximo, se debería no sólo distribuir los planes de contactos, si no que también la asignación específica (*scheduling*) de cada tráfico (prácticamente paquete a paquete) como lo calculó TACP. Sin embargo, por razones obvias, esto resulta imposible en un caso práctico, debiendo confiar en esquemas de enrutamiento distribuidos en cada nodo que tomen decisiones correctas respecto del trafico generado y en tránsito.

En particular, las decisiones de enrutamiento en las que hemos basado el diseño de TACP se basan el conocimiento no sólo de la totalidad de la topología (lo cual se podría lograr por medio de la correcta distribución del plan de contacto), si no que también del tráfico que generarán todos los nodos del sistema. En general lograr un conocimiento global de esto último en cada uno de los nodos de manera distribuida también resulta poco realista sobretodo en aplicaciones DTN de gran escala. En consecuencia, dado que una aproximación distribuida debe contar con información mas local, los esquemas de enrutamiento existentes como Contact Graph Routing (CGR) [59] muestran, en diferentes aspectos, un rendimiento menor que el modelo adoptado para TACP, dificultando la

correcta implementación o aplicación de los planes de contactos en la generalidad de los casos.

En este capítulo repasaremos las problemáticas específicas de implementabilidad o aplicabilidad de plan de contactos en la sección 6.2, para luego resumir el estado del arte esquemas existentes en la sección 6.3, y finalmente discutir tres aportes específicos realizados en esta área. Entre estos presentaremos Cache-CGR (C-CGR) [9] como una mejora al uso del procesador de CGR, el cual fue publicado en la conferencia IEEE International Conference on Wireless for Space and Extreme Environments (WiSEE) en el 2014 en Noordwick, Holanda. Por otro lado, describiremos Path-Aware CGR [10] el cual es un aporte novedoso para extender funcionalidad de congestión de CGR a ser publicado en conjunto con colegas del laboratorio APL de NASA en la próxima conferencia IEEE Conference on Local Computer Networks (LCN) a finales del 2015. Finalmente, en la misma conferencia se presentará CGR Multi-Grafo (MG-CGR) [10] como una propuesta para asegurar la implementabilidad de los planes de contactos diseñados con TACP.

6.2. Descripción del Problema

Ya en la sección 4.2.1 del capítulo 4 definimos formalmente una ruta en DTN como una secuencia de contactos $Route = \{C_1, C_2, C_3, ...C_n\}$ por medios de los cuales el tráfico de paquetes deberá fluir para llegar a su destino. Inicialmente, en el capítulo 2, mostramos que los elementos de planificación en tierra se basan en diseñar el plan de contacto para luego enviarlo a los nodos quienes luego ejecutarán algoritmos mas o menos eficientes para calcular sus rutas. Sin embargo, al tratar los mecanismos basados en rutas (RACP) y en tráfico (TACP) en los capítulos, 4 y 5 respectivamente, hemos asumido conocer o bien la forma en que los nodos calculan estas rutas, o bien el resultado del calculo directamente.

En otras palabras, cuando calculamos las rutas en tierra utilizamos mas información (tráfico) que la que normalmente utilizan los nodos en modo distribuido, situación en la cual, se puede generar una diferencia entre la planificación y la implementación como describimos en la próxima sección 6.2.1. En particular, y dentro de estas diferencias, el tráfico calculado con TACP no evidencia un fenómeno que es común en los esquemas distribuidos denominado *congestión* en DTN que también abordamos en la sección 6.2.2. Por último, si bien existen esquemas que minimizan la congestión, los mismos muestran un compromiso con el procesamiento como trataremos en la sección 6.2.3.

6.2.1. Discrepancias en la Planificación

Para ilustrar la problemática que buscamos enfrentar en este capítulo, proponemos retomar brevemente el análisis de TACP en el capítulo 5 tratado en la sección 5.4.2. La Figura 6.1 muestra el resultado final de una simulación de implementación del contacto diseñado con TACP para la carga máxima ($\rho = 1$ o 540 paquetes por nodo) para CGR: el esquema de enrutamiento actual de DTN. En particular, en la Figura 6.1 a) se ilustra la asignación de flujo asumida por el modelo teórico de TACP, mientras que en la b) se muestra las decisiones tomadas en la simulación por CGR sobre el mismo plan de contacto.

Debe notarse que el flujo final entregado por el esquema de enrutamiento distribuido CGR difiere del calculado por el modelo teórico de TACP particularmente en los arcos destacados por un círculo. Esta diferencia es producto de que, como explicaremos en la sección 6.3.1, CGR decide las próximas rutas en orden cronológico en como el tráfico

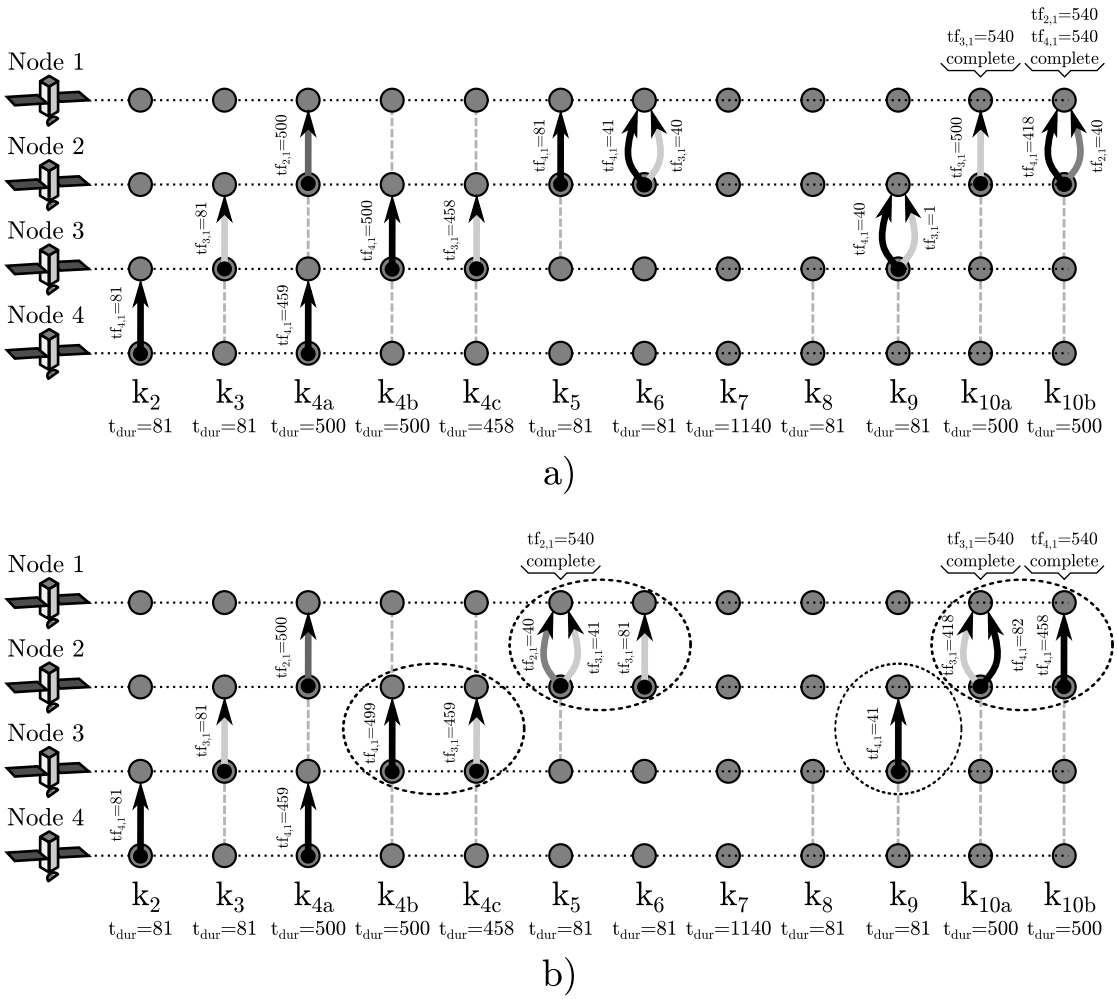


FIGURA 6.1: Flujo de tráfico asumido por TACP en a) y flujo real obtenido en simulación con CGR en b)

le va llegando de los vecinos (o es generado por el mismo nodo). En efecto, este comportamiento es análogo al de una cola del tipo primero que entra, primero que sale o *first in first out* (FIFO), el cual resulta lógico y simple en una primera estrategia de enrutamiento en DTN. Este fenómeno queda completamente claro al observar que en la Figura 6.1 b) el flujo $tf_{2,1}$ obtiene prioridad sobre $tf_{3,1}$ el cual a su vez recibe tratamiento prioritario sobre el tráfico $tf_{4,1}$.

En este caso en particular, es interesante destacar que la diferencia del flujo de tráfico entre lo asumido por TACP y CGR no tiene impacto en las métricas de tiempo de entrega final (o *delivery time*) ni en la de tiempo de contacto de sistema (relacionada con el uso de los recursos disponibles), los cuales se mantienen igual a los concluidos por el análisis de la sección 5.4.2 (4488s y 3240s respectivamente). Sin embargo, y como demostraremos en este capítulo, este ciertamente no resulta el caso general (sobre todo en topologías y patrones de tráfico mas complejos), dado que este tratamiento diferenciado de tráfico puede cambiar el uso de contactos esperados derivando en potenciales problemas significativos en el rendimiento del sistema [10].

En general el problema yace en que los algoritmos distribuidos no son capaces de explotar la información sobre los tráficos que se generan o generarán en los vecinos derivando en potenciales problemas de *congestión* que hacen que las decisiones tomadas en la planificación (diseño del plan de contacto) no se puedan implementar en el sistema.

6.2.2. Congestión en DTN

En redes tradicionales de Internet, existe una clara distinción entre las etapas de enrutamiento (típicamente vinculadas con la capa de red del modelo OSI [47]) y la gestión y control de flujo (asociada a la capa número 4: Transporte). En efecto, el protocolo de Internet o Internet Protocol (IP) en Inglés se encarga de usar una tabla de enrutamiento para dirigir el tráfico, mientras que el protocolo de transporte TCP [24] se basa en retroalimentaciones (*feedback*) permanentes para asegurar que el flujo de transmisión extremo a extremo no supere las capacidades del canal. De esta forma, diferentes tráficos pueden compartir una ruta (o parte de ella) sin generar una sobrecarga del canal o*congestión*.

Sin embargo, en DTN, resulta imposible mantener un flujo constante de información de retorno que permita al transmisor regular su tasa de transmisión de datos debido a la característica disruptiva de los enlaces. En consecuencia, y a diferencia de los mecanismos reactivos en los que se basa Internet, DTN necesita de estrategias proactivas en las cuales se llegue a evitar la congestión de antemano. Particularmente para DTN, el problema de la congestión se puede definir como el intento de enviar mas paquetes (o bundles) a un

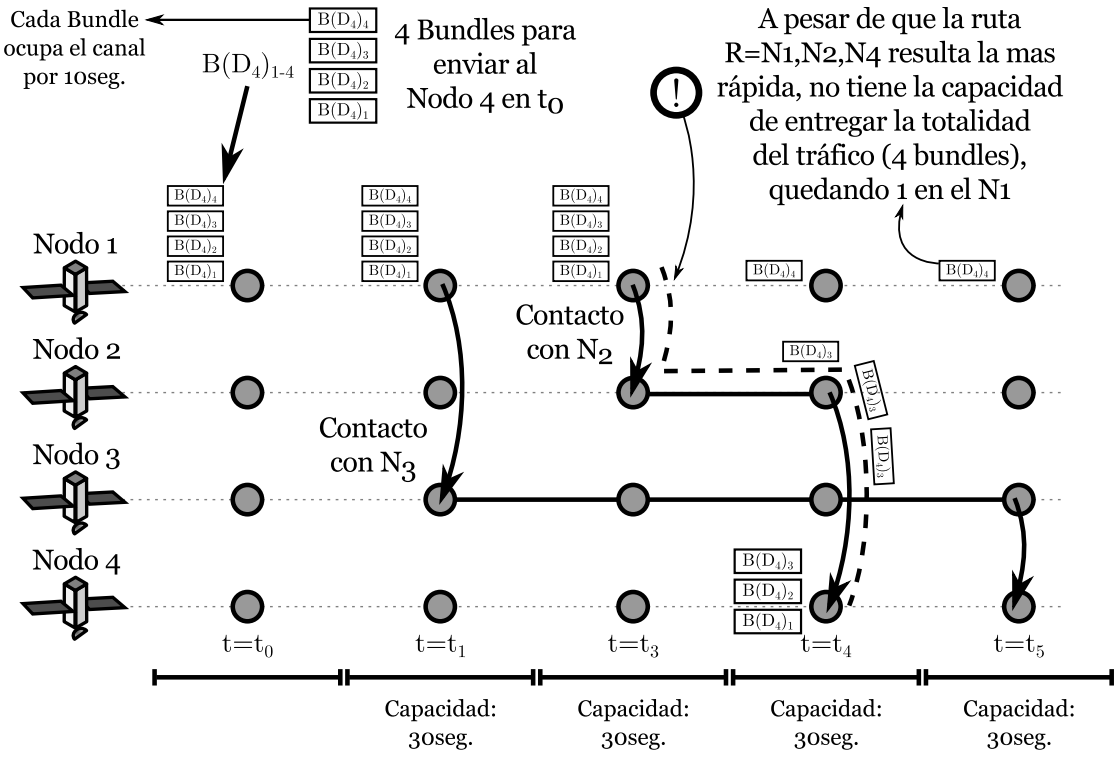


FIGURA 6.2: El problema de la congestión en DTN

nodo vecino que los que el canal, en su condición de tráfico, permite enviar. En general, estos casos de congestión puede darse por:

1. La capacidad física del canal ($dataRate \times time = capacidad$) no permite transmitir la totalidad de los datos.

2. La capacidad física del canal es suficiente pero está total o parcialmente ocupada por otro tráfico. En otras palabras, la capacidad de remanente del mismo no alcanza para evacuar los datos.

La Figura 6.2 ilustra un típico caso de congestión por insuficiencia de la capacidad física del canal. En particular, se plantea un ejemplo de 4 nodos en el cual el nodo N_1 genera 4 bundles que ocupan el canal por 10 segundos cada uno para el destino final N_4. La topología cuenta con contactos de duración de 30 segundos cada uno lo que implica que cada uno de ellos pueden colocar a lo máximo 3 bundles para el siguiente vecino. Efectivamente existen dos posibles rutas al nodo destino a saber $R_1 = \{N_1, N_2, N_4\}$ y $R_2 = \{N_1, N_3, N_4\}$, ambas con una capacidad máxima de 3 bundles máximo.

La figura entonces muestra el comportamiento que tendría la aplicación del algoritmo de ruteo mas simple basado en Dijkstra [115] el que entrega la ruta mas corta como lo hace el algoritmo Merugu's Floyd Warshall (MFW) [61]. En efecto, estos esquemas

entregan la ruta mas corta al destino que en el ejemplo resulta ser $R_1 = \{N_1, N_2, N_4\}$ la cual entregaría el tráfico en t_4. Sin embargo, dado que ninguno de estos esquemas mantiene un conocimiento de las capacidades a consumir por el tráfico enrutado, no pueden reaccionar ante el hecho de que no se podrán enviar los 4 bundles por R_1 (el cuarto quedará en la memoria del N_1 luego de t_4) derivando en un caso de congestión. Evidentemente el mismo podría haberse evitado si uno de los bundles se hubiese enviado por medio de la ruta alternativa $R_2 = \{N_1, N_3, N_4\}$.

En general, si bien el algoritmo de enrutamiento MFW no permite predecir y reaccionar al problema de la congestión física, el esquema de Contact Graph Routing (CGR) [60] incorpora un mecanismo para mitigar estos efectos mejorar el uso del sistema [9]. Sin embargo, el mecanismo propuesto en CGR sólo contrarresta la congestión de manera parcial (no soluciona la congestión por otros tráficos) por lo que se han propuesto otras alternativas como Predicción de Consumo de Capacidad o *Predictive Capacity Consumption* en Inglés (PCC) [97] que a su vez resultan sub-óptimas en algunos contextos específicos. En consecuencia, sucede que ninguna de las soluciones de enrutamiento existentes es capaz de garantizar de que un plan de contacto diseñado por TACP se pueda implementar con los flujos asignados en la etapa de planificación.

6.2.3. Procesamiento en el Enrutamiento

Por otro lado, si bien CGR y sus derivados permiten mitigar hasta cierto punto la congestión, logran esto por medio de un esquema que se basa en una ejecución a nivel paquete o bundle, es decir, una vez por cada uno transmitido. De esta manera se puede tener una base de datos con el estado de la topología como explicaremos en la sección 6.3.1. Por otro lado, MFW [61] no cuenta con esta desventaja en términos de procesamiento ya que las rutas se calculan una sola vez y se almacenan en una matriz para duración de la topología. Sin embargo, como se discutió en la sección 6.2.2, este esquema no tiene consideración alguna a los efectos de la congestión. En efecto, la mitigación de la congestión viene a un precio de procesamiento importante que hace que la implementabilidad de los planes de contacto diseñados resulte un problema sin resolver.

En resumen, Como complemento a los esquemas de diseño en los capítulos 3, 4 y 5, en este capítulo repasaremos en detalle el estado del arte en la sección 6.3 para luego efectuar diferentes aportes a las problemáticas aquí descritas y resumidas a continuación:

1. *Procesamiento:* Como veremos en la sección 6.3.1.1, CGR cuenta con un problema fundamental de alto requerimiento de procesamiento a nivel paquete. En este contexto, en la sección 6.4 propondremos un mecanismo novedoso llamado C-CGR [9]

basado en la filosofía de caché que permite mejorar drásticamente el rendimiento para grandes planes de contacto.

2. *Congestión:* Como también demostraremos en la sección 6.3.1.1, el algoritmo CGR cuenta con una capacidad limitada de gestión de congestión. En este sentido, hemos realizado otro aporte novedoso para extender su capacidad de congestión llamado PA-CGR [10] y descrito en la sección 6.5.1.

3. *Diferencia de Planificación:* Finalmente atacaremos el problema de diferencia de rutas planificadas al proponer un mecanismo sencillo y eficiente para garantizar que los tráficos óptimos supuestos y calculados por TACP puedan respetarse sin necesidad de distribuir un planificación paquete a paquete. Llamaremos a este mecanismo MG-CGR [10] y lo describiremos en detalle en la sección 6.5.2.

6.3. Estado del Arte

En general, y probablemente el algoritmo mas sencillo de enrutamiento es el propuesto por Merugu (MFW) [61] (utilizado y explicado en el planteo del esquema de diseño RACP [4] en el capítulo 4), el mismo carece de aplicabilidad en una red real distribuida dado que calcula las rutas de manera centralizada, lo que deriva en la necesidad de su distribución posterior. A pesar de que esto último podría llegar a ser factible (aunque seriamente mas complejo que la distribución de los planes de contactos), como se explicó en la sección 6.2.2 el esquema de MFW carece de capacidad de reacción ante problemáticas de congestión limitando seriamente su implementabilidad como algoritmo de enrutamiento en redes DTN de uso espacial.

Afortunadamente, futuros desarrollos de algoritmos de enrutamiento permitieron un mejor uso de los recursos para realizar este tipo de cálculo de manera distribuida, siendo el enrutamiento basado en grafos de contacto o Contact Graph Routing (CGR) [59, 60] probablemente el mejor logro obtenido en el área. Tal es así que CGR es parte del software Interplanetary Overlay Network (ION) [67] de NASA descrito en 1.3.3.4, el cual fue validado en vuelo en la sonda DINET como también se explicó en la sección 1.3.3.5 del capítulo 1. En esta sección describiremos el algoritmo básico de CGR así como una serie de extensiones al mismo entre las cuales destacamos la inclusión de una funcionalidad de caché en la sección 6.4 y la capacidad de evaluar capacidades de rutas en la sección 6.5.1 ambas propuestas y publicadas por el autor de esta tesis doctoral.

6.3.1. Contact Graph Routing

6.3.1.1. Descripción General

De acuerdo a la definición descrita en [59], CGR es un conjunto de procedimientos para el cómputo eficientes de rutas por medio del cual protocolos basados en DTN como el Bundle Protocol [46] pueden basar su decisión de envío de datos. En particular, el algoritmo CGR presenta una aproximación heurística al cálculo de rutas basados en una topología global y su evolución en el tiempo (planes de contacto) [60]. Es decir, que CGR está específicamente diseñado para su aplicación en redes DTN predecibles como las aquí tratadas.

Para lograr este cometido, CGR plantea estructurar el plan de contacto en un grafo de contactos (de ahí su nombre) con pesos del tipo $G = (V, E, T, c)$ tal que:

- V : Contiene el conjunto de vértices (nodos en la red)
- E : Contiene los contactos del sistema representando las oportunidades de comunicaciones entre los nodos.
- T : Contiene el conjunto de vértices que conforman un camino óptimo. En efecto, como mínimo, en este conjunto se encuentran los nodos fuente y destino del tráfico.
- c : Contiene una función de costos para cada contacto E del sistema.

De esta representación se deriva la definición de cálculo de ruta utilizada por CGR de la siguiente manera. Se obtendrá una ruta entre el nodo fuente y destino por medio

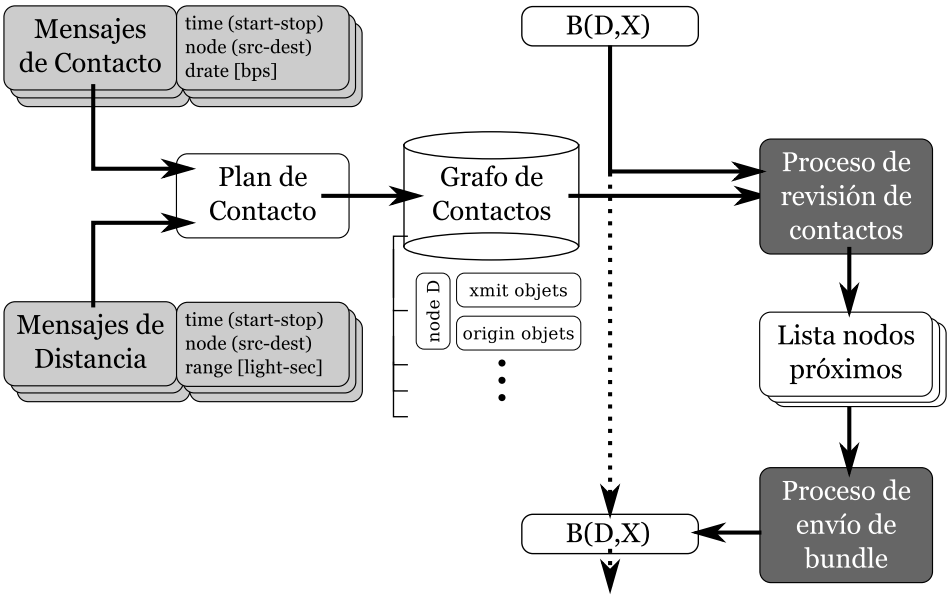


FIGURA 6.3: Flujo de operación del algoritmo CGR

Tabla 6.1: Definición de variables para los procedimientos CRP y FBP

Variables de los procedimientos de CRP y FBP del algoritmo CGR	
Variable	**Descripción**
B	Bundle a enrutar
D	Destino final
X	Tiempo de vida de bundle
m	Contacto
m_start	Tiempo de inicio de contacto
m_stop	Tiempo de finalización de contacto
m_res	Capacidad residual de contacto
T	Último momento de envío
L	Latencia de envío
ECC	Consumo de capacidad estimada
proxNodes{}	Lista de próximos vecinos
exclNodes{}	Lista de nodos excluidos del cálculo
forfeit	Mínimo tiempo de los m_stop en la ruta
bestDelivery	Mejor tiempo de entrega para el vecino

del grafo G que minimice una función objetivo de costo k por medio de una serie de contactos E que unen una serie de nodos V.

De esta manera, CGR construye un grafo de contactos ponderados en el cual diferentes posibles rutas o caminos al destino pueden ser evaluadas. En efecto, cada nodo puede en base a este derivar el próximo vecino para enviar el paquete o bundle, quien en su momento volverá a ejecutar el mismo algoritmo con su visión local de la topología (plan de contacto). Cuando un nuevo bundle $B(D, X)$ con destino final D y tiempo de vida X, llega a un nodo ya sea de un vecino o desde las aplicaciones trabajando en el mismo, se sigue el proceso ilustrado en la Figura 6.3.

En particular, se utiliza el grafo de contactos generado a partir de diferentes mensajes de contacto y de distancia entre nodos (este último para determinar las demoras entre nodos lo cual no evaluamos en esta tesis), para realizar el proceso de revisión de contacto o *Contact Review Procedure* (CRP) en Inglés. Del mismo se determina una lista de próximos y potenciales vecinos, de la cual a su vez un proceso de envío de bundle o Forward Bundle Procedure (FBP) en Inglés permite elegir el mas óptimo (bajo algún criterio) para finalmente poner el paquete en cola de transmisión. A continuación describiremos los procesos de CRP y FBP en detalle.

Algoritmo 5: Proceso de revisión de contactos de CGR

input : D, X
output: $proxNodes\{N_1, N_2...\}$

```
1  exclNodes ← D for each m : m_dest = D do
2      if m_start ≥ X − T then
3          Ignorar m
4      else
5          if m_source = Nodo Local then
6              Computar ECC por salida local if m_res ≤ ECC then
7                  Ignorar m
8              else
9                  if D is in proxNodes{} then
10                     Ignorar m
11                 else
12                     if m_stop ≤ forfeit then
13                         forfeit = m_stop
14                     if m_start ≥ bestDelivery then
15                         bestDelivery = m_start
16                     proxNodes ← D;
17                     Almacenar forfeit y bestDelivery
18         else
19             if S is in exclNodes{} then
20                 Ignorar m
21             else
22                 if m_stop ≤ forfeit then
23                     forfeit = m_stop
24                 if m_start ≥ bestDelivery then
25                     bestDelivery = m_start
26                 Calcular L;
27                 contactReviewP.(S,Min(m_stop − L, D));
28 exclNodes → D Recuperar el stack de forfeit y bestDelivery;
29 return N_i;
```

La Tabla 6.1 lista las variables utilizadas internamente en los procedimientos de CRP y FBP del CGR. En particular, el Algoritmo 5 detalla el procedimiento de revisión de contacto de CGR por medio del cual se obtienen la lista de potenciales vecinos para enviar el bundle. Inicialmente, CRP recorre todos los contactos m con destino D (parámetro de entrada del método CRP). Si alguno de estos tiene un tiempo de inicio m_start mayor que el tiempo de vida X del bundle, es descartado en la linea 3. Si no, se evalúa si el nodo origen del contacto m es el nodo local (es decir el nodo que está ejecutando el algoritmo), caso en el cual se encontró una ruta para la transmisión del mismo por lo

Algoritmo 6: Proceso de envío de bundles de CGR

input : D, X, B, $exclNodes$
output: $forwNodes\{\}$

```
1 if proxNodes ≠ {} then
2     if B_priority = Q_critical then
3         for each N_x in prxoNodes{} do
4             forwNodes{} ← N_i;
5     else
6         N_i = selectNeighbour.(proxNodes);
7         forwNodes{} ← N_i;
8 else
9     Route Error;
```

que se actualizan valores de uso de capacidad y métricas de $forfeit$ y $deliverytime$ en las lineas 6 a 17. En caso de que la capacidad residual del contacto evaluado sea suficiente, esta ruta es entonces guardada en incluida en la lista de $proxNodes$ para su futura evaluación. Por otro lado, si el nodo no es el local, aún estará recorriendo contactos que no tienen un enlace directo con el origen por lo que en la línea 22 y 25 se actualizan las métricas acumuladas y se vuelve a llamar al mismo procedimiento de $CRP(D, X)$ pero ahora con destino final el origen del contacto (S), y un tiempo de vida igual al mínimo tiempo de finalización de contacto m_stop en la línea 27. En efecto, este procedimiento es recursivo, y evidencia una complejidad computacional de $O(VE + V^2logV)$ donde V es la cantidad de nodos y E la cantidad de arcos [60].

A esta altura la lista $proxNodes$ incluye todos aquellos posibles vecinos por medio de los cuales el destino D puede ser alcanzado así como las métricas de esas rutas como su máximo tiempo de transmisión ($forfeit$) y su mejor tiempo de entrega posible ($bestdelivery$). Estos parámetros alimentan el procedimiento de envío de bundles o FBP cuyo comportamiento define el criterio de elección de vecino de acuerdo a la aplicación específica de la red DTN. El Algoritmo 6 detalla el procedimiento de FBP el cual es sencillo y describimos a continuación. Inicialmente, en caso de que no exista ningún vecino cargado en la lista de $proxNodes$, el procedimiento levanta una bandera de error ya que esto quiere decir que el proceso de CRP no pudo encontrar ningún vecino apropiado para entregar el tráfico al destino D. Luego, en la linea 2, si el bundle resulta del tipo crítico (uno de los posibles tipos de bundles), el mismo deberá ser enviado por todos los posibles vecinos listados en la lista de $proxNodes$. En su defecto (es un bundle normal), se deberá elegir uno de los vecinos bajo algún criterio de los listados en la Tabla 6.2. En particular resulta de nuestro interés el de menor $bestdelivery$ dado que el mismo es sumamente deseable en el caso de redes espaciales basadas en satélites.

Tabla 6.2: Criterios de selección de próximo salto

Criterio	Descripción	Aplicaciones
Menor *bestDelivery*	Entrega de los datos en el menor tiempo posible	Aplicaciones satelitales de baja o media órbita
Menor *forfeit*	Entrega de los datos con el mínimo tiempo de circulación en la red	Aplicaciones de espacio profundo (DS)
Menor *costo*	Entregad de los datos por una ruta cuyo costos de los contactos sea minimizado	Aplicaciones en la que la información deba circular por redes no locales posiblemente rentadas

De esta manera, el proceso global de CGR [59] permite encontrar vecinos apropiados para acercar el dato al destino final en una red DTN. En el contexto de la congestión, es de interés notar que en el procedimiento de CRP, el algoritmo internamente mantiene el estado de capacidades residuales de los contactos locales. En efecto, a medida que se van tomando decisiones de transmitir bundles por las interfaces del nodo, las capacidades consumidas por estas se van actualizando de manera que en caso de que las mismas se acaben, el contacto ya no será tenido mas en cuenta para futuros cálculos de enrutamiento. En efecto, y retomando el caso problemático tratado en la sección 6.2.2, en la Figura 6.4 ilustramos como CGR evita el problema de congestión en el cual caía MFW al no tener en cuenta capacidades de contactos.

En este escenario, al enrutar los 4 bundles generados en el t_0, CGR ejecuta los procedimientos de CRP para cada uno de ellos. En efecto, para los primeros 3 la mejor ruta es la $R = \{N_1, N_2, N_4\}$, pero a medida que se va tomando esta decisión, las capacidades del contacto local N_1 a N_2 se van consumiendo hasta llegar a 0 luego del bundle B_3. Luego, al repetir la ejecución del procedimiento CRP para el bundle B_4, cuando se evalúe el contacto N_1 a N_2, el vecino N_2 será ignorado por falta de capacidad remanente (línea 7 del Algoritmo 5). En consecuencia, la lista *proxNodes* sólo contendrá al N_3 como potencial vecino por lo cual ese será elegido específicamente para el bundle B_4, el cual de esta manera es capaz de llegar a su destino final en el tiempo t_4.

En general este mecanismo de mitigación de congestión resulta sumamente útil en relación al procedimiento de MFW, pero aún carece de importantes limitaciones a saber:

- El control de capacidad de contacto sólo se realiza sobre los contactos locales, es decir, sobre aquellos contactos en el cual el nodo ejecutor de CGR es el nodo receptor (D). Los autores de CGR detallan que esta estrategia se basa en que el nodo

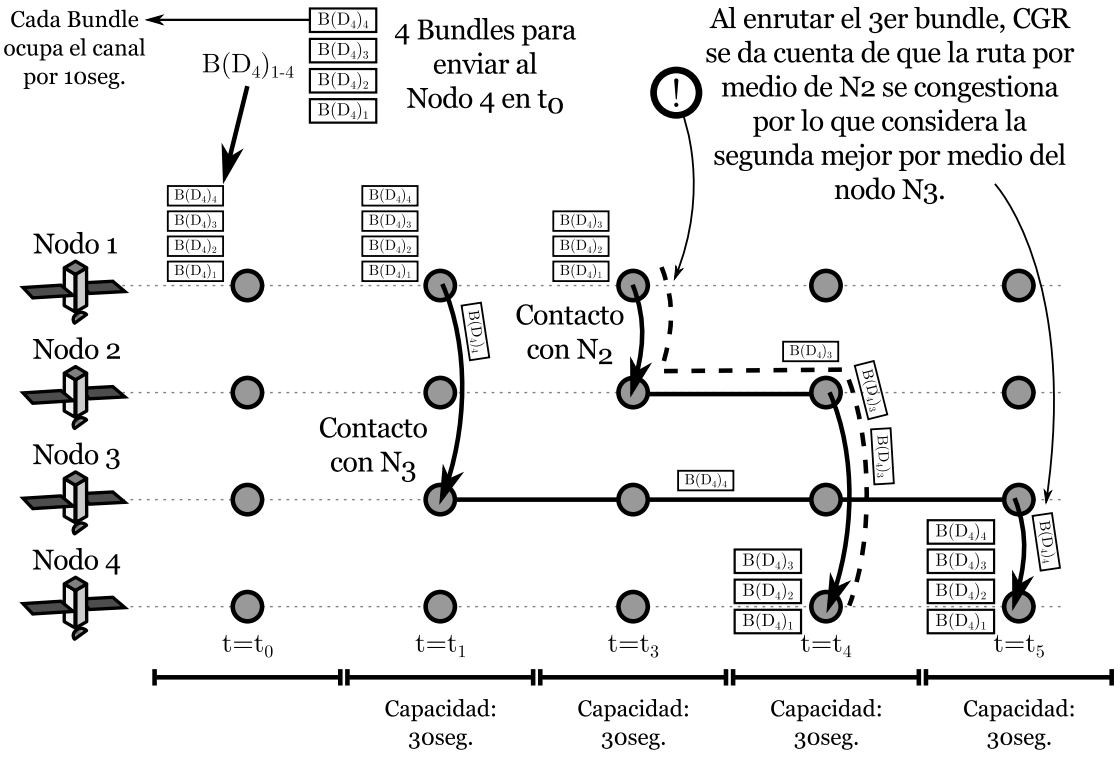


FIGURA 6.4: Solución parcial al problema de la congestión en DTN con CGR

local es el único con autoridad suficiente para determinar que capacidad remanente queda por medio de esa interfaz, pero como mostraremos en la sección 6.5.1 considerar no solo los arcos locales si no que también los remotos permite una significativa mejora del uso del sistema [9]. Este último es otro de los aportes de esta tesis.

- Por otro lado, si bien CGR permite mitigar la congestión sobre los contactos locales, carece de la inteligencia para predecir o conocer tráficos de otros nodos que puedan congestionar contactos futuros asumidos como libres mas adelante en la ruta calculada. Por ejemplo, en la Figura 6.4, podría darse que el contacto entre N_2 y $N-4$ en t_4 esté previamente cargado con tráfico de N_2 a $N-4$ provocando que los bundles B_1, B_2 y B_3 no puedan llegar a destino por la ruta calculada por CGR. Trataremos este fenómeno de conocimiento de otros tráficos en la sección 6.3.2 para finalmente proponer MG-CGR en la sección 6.5.2 como solución definitiva a este problema de congestión.

Por último, pero no menos importante, un problema importante de CGR como se plantea en [59] y [60] es la necesidad de ser ejecutado para cada bundle que se deba enviar. En efecto, esto implica una exigencia importante sobre las computadoras de a bordo de los satélites las cuales suelen ser limitadas en energía y capacidad de cálculo. Como demostramos en [12] y [9], el uso excesivo de CGR tiene impacto final en la máxima

tasa de datos que se puede esperar de estos sistemas. En general estas métricas resultan alarmantes por lo cual otro aporte secundario de esta tesis fue realizar una propuesta alternativa llamada cache-CGR o C-CGR la cual permite hacer un ahorro significativo del procesador. Describimos C-CGR en detalle en la sección 6.4.

6.3.1.2. Sobre la Política de Retorno al Nodo Previo

Una de las principales desventajas del mecanismo de gestión de la congestión con la que cuenta el algoritmo de enrutamiento CGR, es que la visión de la capacidad remanente del plan de contacto se limita a los contactos locales. En efecto, esta aproximación de este algoritmo clasificado como codicioso (o *greedy* en Inglés) puede derivar en visiones diferentes de la red la que a su ves favorezca la toma de decisiones de enrutamiento que generen conflicto con las tomadas en los nodos vecinos.

En particular, estos conflictos pueden derivar en la formación de lazos de enrutamiento (o *routing loops* en Inglés) en el cual dos nodos queden enviándose permanentemente un bundle pensado que el otro es el mejor próximo vecino. De acuerdo a lo expuesto en [59] y su implementación en el software ION [67], esta problemática se contrarresta con una simple política de prohibir la devolución del bundle al nodo previo. En general, la misma resulta útil y cumple el objetivo de mitigar el problema de lazos de rutas, pero como el autor de esta tesis detectó en [9], también representa una barrera importante a la hora de aplicar decisiones reactivas para evitar la congestión.

Por ejemplo, como se ilustra en la topología de la Figura 6.5, un próximo contacto $c_{k,2,3}$ con una capacidad de 100 es considerado por el nodo 1 y el nodo 2 como parte de la mejor ruta (más rápida) hacia el nodo destino 3. Además, un contacto futuro $c_{k,1,3}$ puede ser considerado como una ruta alternativa aunque mas tardía en comparación a la que utiliza $c_{k,2,3}$. Cuando el nodo 2 envía 100 unidades en el estado k_2, el mismo actualiza la base de datos local de la capacidad del contacto $c_{k,2,3} = 0$ que es la capacidad final que el contacto $c_{k,2,3}$ tendrá una vez que el tráfico sea enviado por medio de el. En este estado, el nodo 1 simplemente no tiene manera de conocer que la capacidad del contacto $c_{k,2,3}$ fue completamente tomada por el nodo 2, razón por la cual se genera un situación de diferencia de conocimiento de capacidad distribuida.

Como veremos mas adelante en la sección 6.3.2, el esquema PCC permitiría actualizar estos desfases de conocimiento sobre las capacidades de la topología por medio de envío asíncrono de mensajes de actualización. Sin embargo los mismos pueden resultar poco útiles en escenarios muy disruptivos necesitando de mejores estrategias como MG-CGR tratada en la sección 6.5.2. Para los otros casos, esta falta de sincronismo puede no tener solución y requerir de los nodos puedan reaccionar e encontrar rutas alternativas por

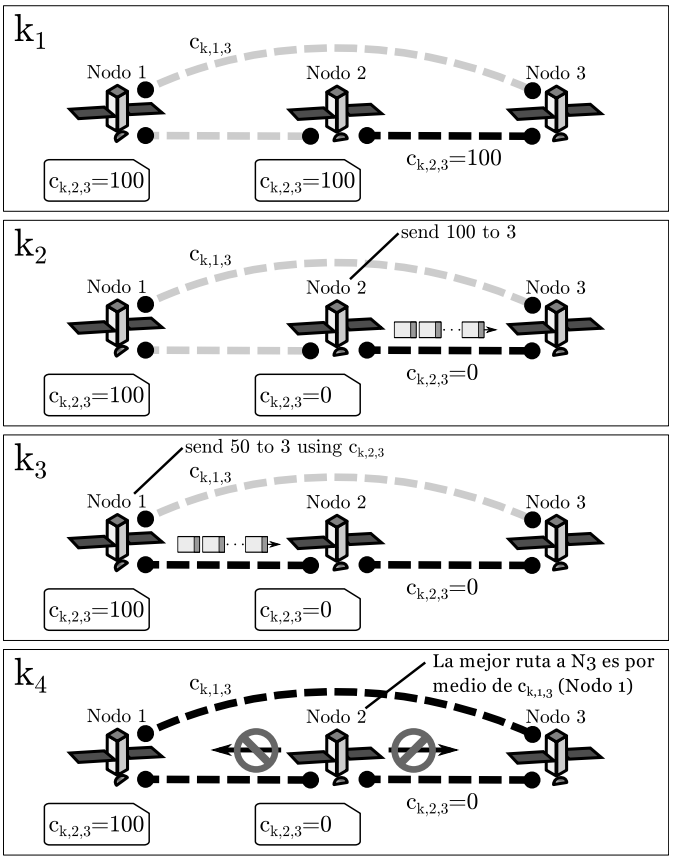


FIGURA 6.5: Impacto de la política de retorno a nodo previo en DTN

mas de que el tráfico ya haya avanzado por un primer camino que finalmente resultó congestionado. Sin embargo, como mostramos a continuación, la política de no retorno al nodo previo puede hacer esto imposible.

Continuando con el escenario de la Figura 6.5, cuando el nodo 1 intente enviar 50 unidades de datos en el estado k_3 (también hacia el nodo 3 como destino final), su cálculo de CGR local acusará una ruta óptima por medio del nodo 2 dado que el contacto hacia el mismo nodo tiene capacidad, y que a su vez el segundo contacto en el camino se predice (erróneamente) con $c_{k,2,3} = 100$. De esta manera, cuando el nodo 2 recibe estos datos (hacia el final del estado k_3), el mismo calculará sus rutas cayendo en cuenta de que el contacto $c_{k,2,3}$ ya no tiene capacidad remanente. En consecuencia, deberá elegir una ruta alternativa, la cual efectivamente existe por medio del contacto $c_{k,1,3}$. Sin embargo, para utilizar ese camino, los bundles deberán volver al nodo 3, evento que resulta prohibido de acuerdo a la política implementada para evitar lazos de enrutamiento en CGR. Finalmente, el tráfico queda estancado en el nodo 2 sin capacidad alguna de reacción derivando por un lado en una ocupación de memoria permanente y por otro en un tráfico que nunca llegó a su destino final.

A pesar de que la política aquí tratada resulte necesaria en algoritmos *greedy* de salto

simple como CGR, sus efectos resultan catastróficos en términos de capacidad de reacción a la congestión. En particular, este fenómeno aleja significativamente la capacidad de implementabilidad de un plan de contacto diseñado con un esquema de tráfico como TACP. Por esta razón, en esta tesis se realiza un último aporte hacia garantizar la mejor implementabilidad por medios de mecanismos como MG-CGR en la sección 6.5.2.

6.3.2. Predicción de Consumo de Capacidad

Inicialmente propuesto en [97], el esquema de predicción de consumo de capacidad o *Predictive Capacity Consumption* (PCC) en Inglés, extiende CGR con el fin de incluir un mecanismo que permita acceder al conocimiento de tráfico provocado por otros nodos vecinos. En efecto, esto permitiría mejorar las capacidades de gestión de la congestión de CGR como se lo plantea en su especificación [59] y se detalló en la sección 6.3.1.1.

PCC logra esto al inferir el tráfico por observación una versión extendida de la cabecera de la cabecera del paquete la que ahora incluye el camino completo que el mismo recorre así como las capacidades residuales vista por los nodos previos. Si la capacidad observada de los contactos futuros es menor que la almacenada en la base de datos local, la misma es actualizada (sincronizada) bajo un cierto valor de confianza o *confidence value* (CV) en Inglés donde $0 \leq \text{CV} \leq 1$. A mayor peso de CV mayor la confianza en los valores de capacidad acusada por los nodos vecinos. Por otro lado, en el nodo origen se utiliza el conocimiento de capacidad de toda la ruta para el cálculo del camino a codificar en la cabecera del paquete y hace uso de un mecanismo original de mensajes de retroalimentación asíncronos por medio del cual un nodo puede compartir la capacidad residual de sus contactos. En efecto, estos mensajes permitirían que en el caso problemático tratado en la sección 6.3.1.2 (Figura 6.5) el nodo N_2 pueda enviar una actualización respecto de la capacidad residual del contacto $c_{k,2,3} = 0$ para el nodo N_1 evite enviar mas datos por medio de ese contacto. Para mayores detalles del funcionamiento de PCC sugerimos al lector referirse a [97].

De esta manera, PCC mejora el conocimiento local de las capacidades remota permitiendo evitar congestiones en ciertas topologías. Sin embargo el mismo requiere de una sobrecarga significativa de la cabecera del paquete al necesitar codificar no sólo el camino de la ruta completo ($R = C_1, C_2, C_3, ...C_n$) si no que también sus capacidades residuales ($Cap = \{c_{1,1,1}, c_{1,2,1}, c_{3,1,2}, ...c_{k,i,j}\}$), listas que pueden resultar extremadamente largas para grandes redes o con alto grado de fragmentación en la fuente [128]. Además no existe un acuerdo en el formato (tamaño en bytes) necesario para codificar los identificadores de contacto (en [97] se sugiere incluir identificadores de m_start, m_end, S, y D) ni el tipo de variable para definir capacidades remanentes, aunque seguramente los

mismos superarán los varios bytes por cada salto. En este trabajo, y en los futuros análisis ignoraremos estas penalidades en cabeceras (dejándolas para investigaciones futuras) para concentrarnos en el comportamiento algorítmico de PCC.

A pesar de estos problemas, la inspección de cabecera de PCC resulta especialmente efectiva en redes con tráfico intenso donde el intercambio de datos permite mantener las capacidades de los nodos remotos actualizada y en sincronismo. Sin embargo es necesario destacar que un tráfico en dirección entrante no necesariamente actualizad el camino saliente (retorno) dado que los contactos (y por ende su capacidad residual) en DTN son direccionales. En estos casos o bien aceptaciones de custodias de bundle o mensajes de feedback opcionales pueden aliviar esta situación, sin embargo bajo que políticas y criterios generar los mismos permanece un tema de investigación abierto sin una solución *a priori*. En esta tesis retomaremos el mecanismo PCC como se describe en [97] en la sección 6.5.3 para compararlo con CGR [60] y las dos propuestas elaboradas en este capítulo: PA-CGR en la sección 6.5.1 y MG-CGR en la sección 6.5.2.

6.4. Aportes en la Eficiencia de Procesamiento

6.4.1. CGR con Extensión de Cache

Si bien hemos tratado y descrito el algoritmo CGR [59] como una mejora significativa al problema de la congestión que se da al aplicar otros esquemas como MFW [61] (ver ejemplo de la sección 6.2.2), existe otra diferencia sustancial entre ambos a favor del segundo. En particular, nos referimos a la eficiencia de procesamiento la cual no es para nada menor como el autor de la tesis supo publicar en [9]. En general, la mejora en la gestión de la congestión en CGR viene a costa de realizar ejecuciones del algoritmo tantas veces como bundles se deban transmitir. De esta manera se puede ir actualizando una base de datos (Figura 6.3) con el grafo de contacto en la cual se mantiene un registro detallado de las capacidades remanentes.

Como se muestra en la Figura 6.6 a) cuando 4 bundles (B_1 a B_4) con destino D_4 se reciben para transmitir por CGR, se realiza una ejecución del esquema detallado en la sección 6.3.1.1 para cada uno de los 4 paquetes $B(D_4)_{1-4}$ con el fin de derivar el próximo vecino (N). En este proceso CGR evalúa las capacidades remanentes en la base de datos de contacto, ya que si las mismas se encuentran agotadas no serán consideradas como rutas válidas. Por ejemplo, en el caso ilustrado en la Figura 6.2, el contacto con N_2 sólo tiene capacidad de transmitir 3 bundles razón por la cual el número 4 se encola en la cola para el nodo N_3 como finalmente se evidencia en la Figura 6.4. En efecto, cuando un nuevo contacto comienza con un vecino N_x, la cola correspondiente se vacía ordenando

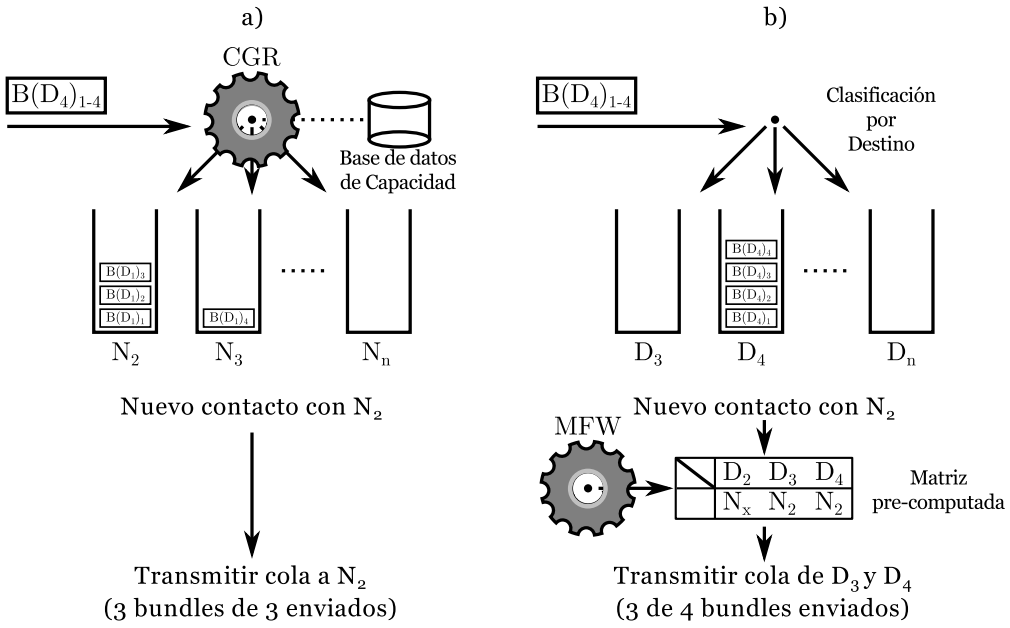


FIGURA 6.6: Proceso de encolado y transmisión de CGR y MFW

la transmisión. Sin embargo este comportamiento viene a costa de una ejecución por bundle la cual resulta prohibitiva en escenarios limitados en potencia de cómputo y sobre todo con altos niveles de fragmentación.

En la definición de Bundle Protocol [46], la fragmentación es el proceso por medio del cual un bundle de mayor tamaño se divide en varios mas pequeños con el objeto de poder ser enviados por contactos de menor capacidad, optimizar el uso de los mismos, y evitar retransmisiones de bundles parcialmente transmitidos [45]. Al día de la escritura de esta tesis el criterio de aplicación de esta estrategia permanece en discusión en el grupo investigación [43] y estandarización [42] de DTN. Sin embargo, y en general, la misma deriva en la existencia de un mayor número de bundles lo que tiene un impacto negativo en la aplicación de CGR cuya complejidad es proporcional a la cantidad de los mismos.

Sin embargo, y por otro lado, MFW es sumamente mas eficiente al evitar hacer cálculos por paquetes si no que con una mayor granularidad como mostramos en la Figura 6.6 b). En particular, MFW clasifica cada uno de los $B(D_4)_x$ de acuerdo a su destino de manera que cuando el contacto con el vecino N_2 comienza, la transmisión se concreta luego de buscar en una matriz de rutas a que destinos se puede llegar por medio de ese nodo N_2. Esta matriz traduce vecinos N_x a destinos D_y y es precisamente el resultado de una sola ejecución de MFW (probablemente centralizada y luego distribuida) como se detalla en [61]. En consecuencia, al calcular los destinos una sola vez, las necesidades de procesamiento en MFW resultan sumamente menor que en CGR.

Algoritmo 7: Algoritmo de CGR con inclusión de cache (C-CGR)

input : t, B_{size}, $B_{d.Line}$, $B_{dest.}$, C_{DB}
output: N_i

```
1  if cache.find(B_dest.)==false then
2      P_i=contactReviewP.(B_dest.,B_d.Line);
3      N_i=forwardBundleP.(P_i);
4      C_DB[N_i] → RC_j− = ECC;
5      cache[B_dest.]=N_i;
6  else
7      if ((C_DB[N_i] → RC_j − ECC)<= 0) ∨ (C_DB[N_i] → cEnd_j <= t)) then
8          cache.erase(B_dest.);
9          N_i =C-CGR(t, B_size, B_d.Line, B_dest., C_DB);
10     else
11         C_DB[N_i] → RC_j− = ECC;
12         N_i=cache[B_dest.];
13 return N_i;
```

Inspirados en esta comparación realizada se realizó otro aporte secundario en esta tesis que fue publicado en [9] donde el autor formuló una estrategia novedosa de aplicación de CGR basada en el uso de un caché o memoria temporaria en la que el vecino generado por una ejecución de CGR se almacena para evitar ejecuciones similares en el futuro. Por ejemplo, si consideramos una serie de bundles B_n a ser enviados al mismo destino D_d, CGR entregará el mismo vecino N_i hasta que la capacidad C_j sea completamente agotada. Suponiendo que en C_j entran B_m bundles, el algoritmo CGR devolverá la misma solución m veces desperdiciando valiosa energía de procesamiento.

De esta manera, en analogía con la jerarquía de memoria de la rama de la ciencia de computación, propusimos estudiar la utilidad de incorporar un cache de vecino denominado *next-neighbor cache* [9] para así diseñar Cache-CGR o C-CGR. Esta memoria mantendrá en la misma el próximo vecino N_i para cada destino así como la capacidad residual al contacto asociado al mismo RC_j. En efecto, RC_j deberá actuar como un puntero a la base de datos de contactos ya que se debe garantizar la consistencia del mismo a lo largo de llamados a C-CGR con diferentes destinos D_x.

El algoritmo 7 detalla el comportamiento propuesto para C-CGR el que toma como entrada el tiempo t, el tamaño de bundle B_{size}, su tiempo de vida $B_{d.Line}$, su destino $B_{dest.}$, y un puntero a la base de datos de contactos C_{DB} como parámetros de entrada. C-CGR es llamado por cada bundle a transmitir, pero el mismo actúa como una barrera a los llamados a CGR el cual resulta sumamente mas costoso en términos computacionales. En efecto, si la entrada $B_{dest.}$ no existe en el cache (es decir, *cache miss* en la línea 1) se deberá ejecutar un procedimiento completo de CGR en las líneas 2 a 4 cuyo resultado se

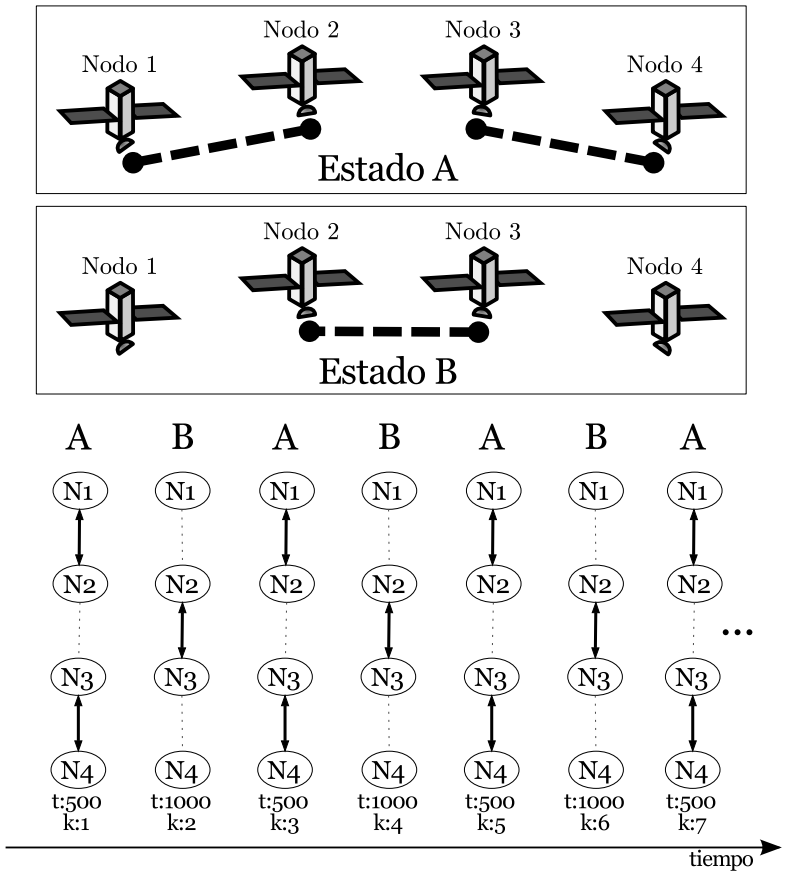


FIGURA 6.7: Topología en tren para análisis de C-CGR

agregará al cache en la línea 5. Por otro lado, si la entrada ya existe, se produce un acierto de cache (*cache hit* en Inglés), aunque se necesita de mayores validaciones en la línea 7 antes de declararlo exitoso. En consecuencia, se controla que la capacidad residual RC_j es suficiente para la capacidad consumida estimada o Estimated Capacity Consumption (ECC) en Inglés y que el tiempo de contacto almacenado $cEnd_j$ no ha expirado para ese bundle. Si dicha validación falla, se declara un error de caché (*cache miss*) y se debe ejecutar CGR nuevamente. Sin embargo, si la entrada permanece válida se evita una repetición de llamado a esta rutina con significativos ahorros de procesamiento como mostramos a continuación.

6.4.2. Análisis de Procesamiento por Simulación

En esta sección proveemos un análisis del aporte realizado en C-CGR. Dado que C-CGR es una mejora al rendimiento computacional de CGR mantenemos este análisis separado del realizado por el tema de congestión de la sección 6.5.3 donde compararemos PA-CGR explicado en la sección 6.5.1 y MG-CGR detallado en la sección 6.5.2.

Tabla 6.3: Parámetro para el análisis de C-CGR

Parámetro	Valor
Bundle + Cabecera	1024B
Cantidad de Bundles	5000 bundles
Intervalo de Generación de Bundles	0
tasa de Datos en Contacto	5Kbps to 100Kbps

En [9] se hace un análisis del ahorro de ejecuciones de CGR en el escenario de estudio y referencia C (topología en tren) descrito en la sección 4.4.3 del capítulo 4. La topología luego de ser diseñada con FCP [2] (propuesto en el capítulo 3) oscila entre dos estados diferentes A y B como se muestra en la Figura 6.7. La Tabla 6.3 detalla los parámetros de tráfico bajo los cuales realizamos la comparación donde un tamaño de bundle de $1024KB$ es considerado como producto de una posible fragmentación de un tráfico de mayor volumen ($5000x1024KB = 5{,}12MB$). Además variaremos la capacidad de contacto para estudiar el efecto sobre C-CGR. En este escenario medimos las posibles resultados de los llamados a la rutina de C-CGR las cuales pueden provocar un *cache hit* (el vecino está en la tabla y es válido) o un *cache miss* (vecino no está en tabla o es invalido por falta de capacidad o vencimiento de contacto). En efecto, cada acierto de caché es una ejecución menos del algoritmo de ruteo CGR.

La Figura 6.8 muestra los resultados de cada uno de los llamados a C-CGR para cada uno de los 5000 bundles en cada uno de los nodos del sistema. En efecto, el nodo N_4 no se ilustra dado que en el no existe ninguna ejecución del algoritmo de enrutamiento por que es el destino final del tráfico. En general, para cada una de las barras hay al menos un *cache miss* debido a una causa de caché vacío o *empty cache* en Inglés. Esta sucede la primera vez que se llama a C-CGR con el $D = N_4$ derivando en la primera ejecución del algoritmo CGR. Las otras razones de falta de capacidad residual o *No residual Capacity* Inglés y vencimiento de contacto o *contact overdue* en Inglés se analizan a continuación para cada escenario.

En total se ejecutan 4 simulaciones para tasas de datos de 5, 10, 50, y $100Kbps$. En el caso general, se observa que los aciertos de cache se incrementan a mayores tasa de datos lo que coincide con nuestra hipótesis original de que a mayor bundles por contacto (con un mismo destino D) mayor ahorro de procesamiento se hace posible. Para el caso particular de $100Kbps$ un total de 4999 aciertos permite ahorrar esa cantidad de ejecuciones de CGR (todos se envían en el mismo contacto en menos de 500 segundos). Por otro lado, la métrica de desaciertos por *no residual capacity* (falta de capacidad residual) tiende a pasar mas frecuentemente a medida que la tasa de datos (y por ende la capacidad de los contacto) decrementa. En efecto, cada vez que una ejecución de C-CGR concluye con

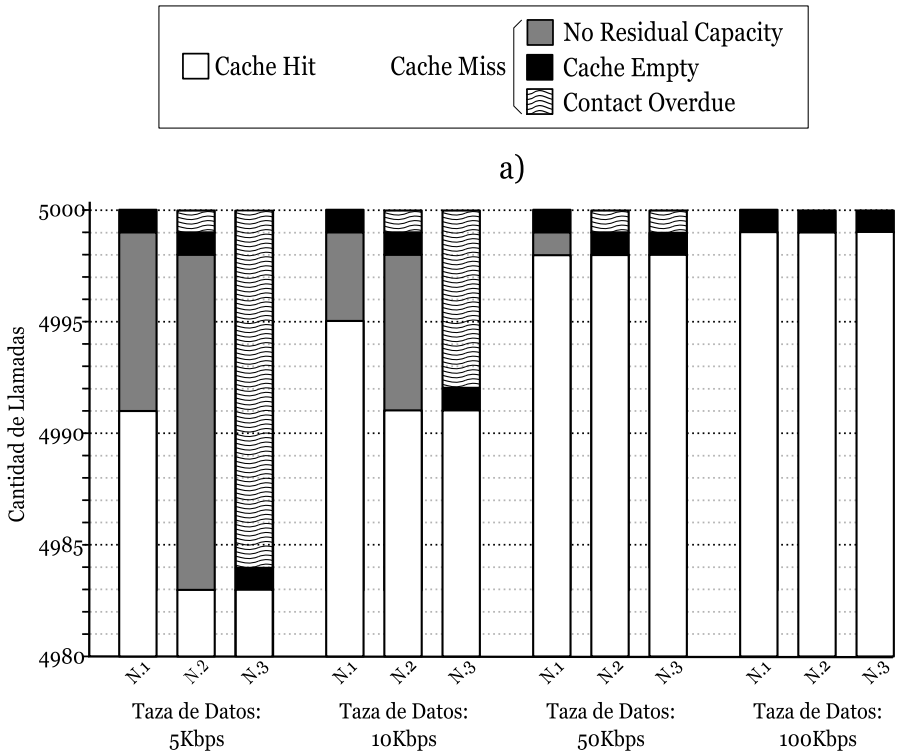


FIGURA 6.8: Cantidad de llamadas a C-CGR Para diferentes capacidades de contacto

este resultado es por que uno de los contactos de la topología fue agotado y se requiere de una nueva ejecución de CGR para determinar una ruta alternativa. Por ejemplo, en el caso de $10Kbps$, el N_1 percibe un total de 4 desaciertos pro esta razón mostrando que se han usado 4 contactos diferentes para llegar al N_2 (el próximo salto a N_4). Estos son precisamente $K : 1, 3, 5$ y 7 de la Figura 6.7.

La métrica de vencimiento de contacto no tiene impacto inicialmente en el nodo N_1 dado que todo el tráfico se genera en el tiempo $t = 0$ y encolado correspondientemente a todos los contactos futuros. Sin embargo, en el nodo N_2 y N_3 C-CGR se llama a medida que los bundles llegan al nodo en tiempos que evolucionan con la topología provocando posibles casos en los que el vecino almacenado en la tabla del cache está asociado a un contacto que ya ha concluido. Por ejemplo, en el caso de $50Kbps$, N_1 puede transmitir a N_2 un máximo de 3051 bundles en $k : 1$ y los restantes 1949 en $k : 3$; por ende, N_2 transmitirá 3051 bundles en $k : 2$ y 1949 en $k : 4$ a N_3. Aquí, el nodo N_2 experimentará un desacierto de caché al llamar a C-CGR para el bundle número 3052 (en $k : 4$) dado que el contacto a N_3 en $k : 2$ almacenado en la tabla de cache ya está vencido (ha concluido) y necesita ser reemplazado por un valor actualizado (el contacto a N_3 en $k : 4$). Este fenómeno se aplica para N_3 por igual y explica la razón por los desaciertos de cache debido al vencimiento de los contactos en la Figura 6.8.

6.4.3. Análisis de Procesamiento por Implementación

Con el fin de evaluar el impacto de C-CGR en la tasa de datos real de una aplicación DTN realizamos un análisis de implementación sobre el software de ION descrito en la sección 1.3.3.4 del capítulo 1. En la siguiente sección 6.4.3.1 describiremos el banco de prueba sobre el cual ejecutaremos las mediciones resumidas en la sección 6.4.3.2. Este análisis también fue publicado en [9] en el congreso WiSEE 2014 en la agencia espacial europea (ESA) Holanda, y dado que el mismo se basa en implementaciones reales tuvo un impacto significativo en la comunidad.

6.4.3.1. Banco de Prueba

Como se nombró previamente en la sección 6.3, la implementación de referencia de DTN desarrollada por NASA incluye una versión de CGR sobre la cual evaluamos la utilización del esquema C-CGR con caché. Para esto proponemos el uso de la versión

Tabla 6.4: Parámetros de configuración del banco de prueba

Parámetros de Gaisler Ethernet	
Nodos	2
tasa de Datos	100Mbps
Parámetros de RTEMS	
Versión	4.10
Max Semaphores	20
Max Message Queue	10
Max Tasks	40
POSIX Threads	40
MUTEX POSIX	10
POSIX Condition Variables	10
POSIX Semaphores	100
POSIX Message Queue	10
Parámetros de ION	
Nodos	2
Working Memory	30MB
Heap Memory	10MWords
Underlying Protocol	UDP and TCP
Tiempo de contacto	Permanente
Rango de contacto	1s

3,1,3 de ION sobre el sistema operativo de tiempo real RTEMS O.S. (versión 4,10) sobre dos computadoras de vuelo basadas en el procesador LEON3 System-on-Chip (SoC) de Gaisler embebido en una FPGA ProAsic3. El sistema se configuró con un procesador LEON3-FT con un cache de $8 + 4KB$ a una frecuencia de reloj de $25MHz$ de acuerdo a lo recomendado por la hoja de datos de la empresa Actel [77], lo que permite entregar un total de 20 DMIPS. Por otro lado se utilizaron memorias RAM de $256MB@190MHz$ DDR2 con DMA habilitado. El procesador usa un bus AMBA del tipo Advanced High-performance Bus (AHB) [129] por medio del cual se conecta con un IP Core Ethernet de Aeroflex Gaisler el que a su vez transmite a una tasa de datos máxima de $100Mbps$. Mayores datos de la plataforma sobre la que ejecutamos este análisis se pueden consultar en [12] y revisar en la Tabla 6.4.

Por último, utilizaremos herramientas de medición de uso de CPU de RTEMS para seguir de cerca el hilo *ipnfw* de ION ya que el mismo es el encargado de ejecutar las rutinas de enrutamiento del mismo. Los resultados se resumen en la próxima sección 6.4.3.2.

6.4.3.2. Mediciones

Al medir la tasa de datos (o *throughput* en Inglés) sobre el banco de prueba especificado, podemos inferir el impacto en el procesamiento que tiene CGR y la mejora propuesta C-CGR. En efecto, la Figura 6.9 muestra los valores de transmisión y recepción de datos en función de la cantidad de contacto en el plan de contacto utilizado para enrutar el dato. En general, mientra mayor la cantidad de contactos, mayor el tiempo necesario por CGR para calcular la ruta de cada bundle de $1024B$ antes de encolarlo al próximo vecino. En particular, la Figura 6.9 a) muestra el throughput medido para el algoritmo CGR como se lo plantea en la sección 6.3.1 y la nueva propuesta C-CGR.

Es interesante observar que sin importar la variación en la cantidad de contactos, la versión de C-CGR entrega una tasa de datos efectiva constante mientras que CGR claramente muestra un compromiso entre desempeño y throughput final. Esto es un claro indicio de que C-CGR minimiza el tiempo de cómputo para el cálculo de rutas confirmando la hipótesis planteada inicialmente en la sección 6.4.1. Por otro lado, la medición del uso de CPU también sustenta el resultado obtenido. Por último vale aclarar que la diferencia en la tasa de datos en transmisión y recepción en el protocolo UDP son consecuencia de la falta de control de flujo característica de ese protocolo lo que deriva en una pérdida de paquete o *packet drop* en Inglés. En efecto, al usar TCP los flujos se auto regulan y convergen en un valor único de transmisión y recepción.

En conclusión, en [9] hemos realizado un aporte de importancia hacia la implementabilidad de los planes de contactos al mejorar el desempeño que tiene el popular algoritmo

de enrutamiento CGR. Hemos demostrado por medio de simulaciones e implementaciones que C-CGR permite considerar CGR como un esquema de enrutamiento capaz de utilizar como entrada planes de contactos de tamaño considerable facilitando la implementación de los mismos en redes DTN satelitales de gran número de nodos o de largos períodos orbitales.

6.5. Aportes en la Congestión e Implementabilidad

En la sección 6.4 abordamos la problemática de la eficiencia de procesamiento en la cual aportamos con C-CGR para permitir una mejor implementación de los planes de contactos en nodos con escasa capacidad de procesamiento. En esta sección abordaremos inicialmente el tema de la congestión con la sección 6.5.1 al tratar PA-CGR, para luego, en la sección 6.5.2, derivar en MG-CGR: un método mas eficiente aún que además permite garantizar la implementabilidad de los diseños realizados con TACP.

6.5.1. CGR con Registro de Ruta

Como se describió en la sección 6.3.1, Contact Graph Routing (CGR) es un conjunto de procedimientos por medio de los cuales se pueden computar rutas eficientes basadas en un plan de contacto de entrada. En general, este plan de contacto se distribuye de manera idéntica a cada nodo para que los utilicen con este fin. Cabe destacar que en esta distribución uniforme será cuestionada en la sección 6.5.2 donde introduciremos una estrategia de múltiples grafos en el sistema.

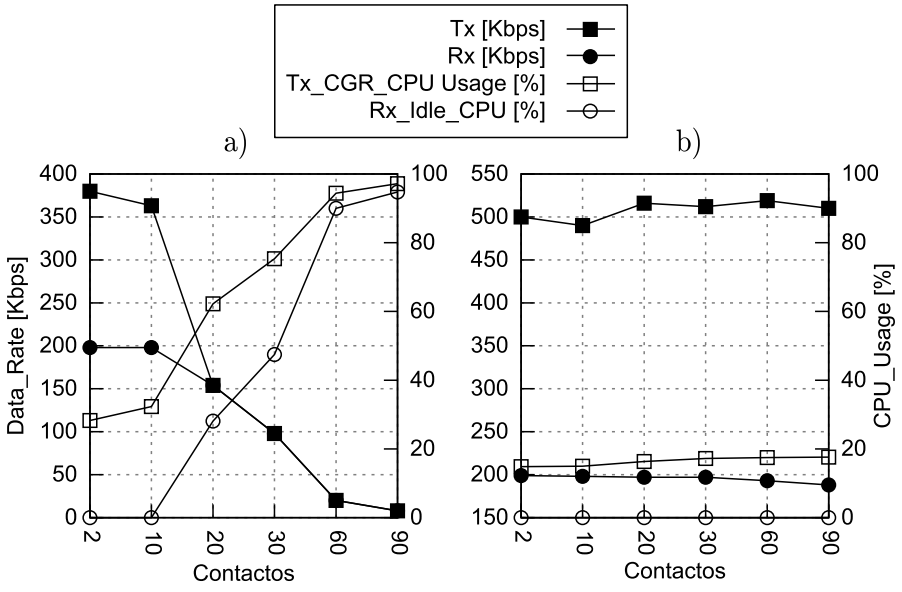


FIGURA 6.9: tasa de datos y utilización de CPU para a) CGR y b) C-CGR

Una vez que el plan de contacto es recibido por el nodo, CGR lo almacena en una base de datos en la cual además mantiene información sobre las capacidades residuales de los contactos *locales* (es decir, aquellos en lo que el nodo participa como transmisor o S) para mitigar los efectos de la congestión como se introdujo en la sección 6.3.1.1. En [59] se argumenta que sólo se consideran contactos locales dado que el nodo transmisor es la única autoridad capacitada para asegurar que tráfico está encolado en ese contacto. Es decir, la predicción de la congestión en CGR sólo contempla un sólo salto en adelante. Si bien el algoritmo PCC [97] permite extender este rango, el mismo peca de una sobrecarga significativa en la cabecera del paquete DTN como se argumentó en la sección 6.3.2.

La Figura 6.10 ilustra el problema con el mismo ejemplo utilizado al comienzo de este capítulo. Aquí, CGR pudo efectivamente predecir la congestión en el contacto local N_1 a N_2 con una capacidad máxima de 30 segundos evitando enviar el bundle 4 por este contacto aprovechando el que se da entre N_1 y N_3 en t_1. Sin embargo, en este ejemplo particular el contacto N_2 a N_4 (el segundo salto de la ruta $R = \{N_1, N_2, N_4\}$) sólo cuenta con una capacidad de 20 segundos por lo que puede enviar un máximo de 2 bundles. En efecto, el algoritmo de CGR no mantiene un registro de estas capacidades por lo que ignora este cuello de botella derivando en que el bundle 3 quede estancado en el nodo intermedio N_2 en espera de una nueva ruta.

Con el fin de mitigar este problema, en esta sección introduciremos una extensión a

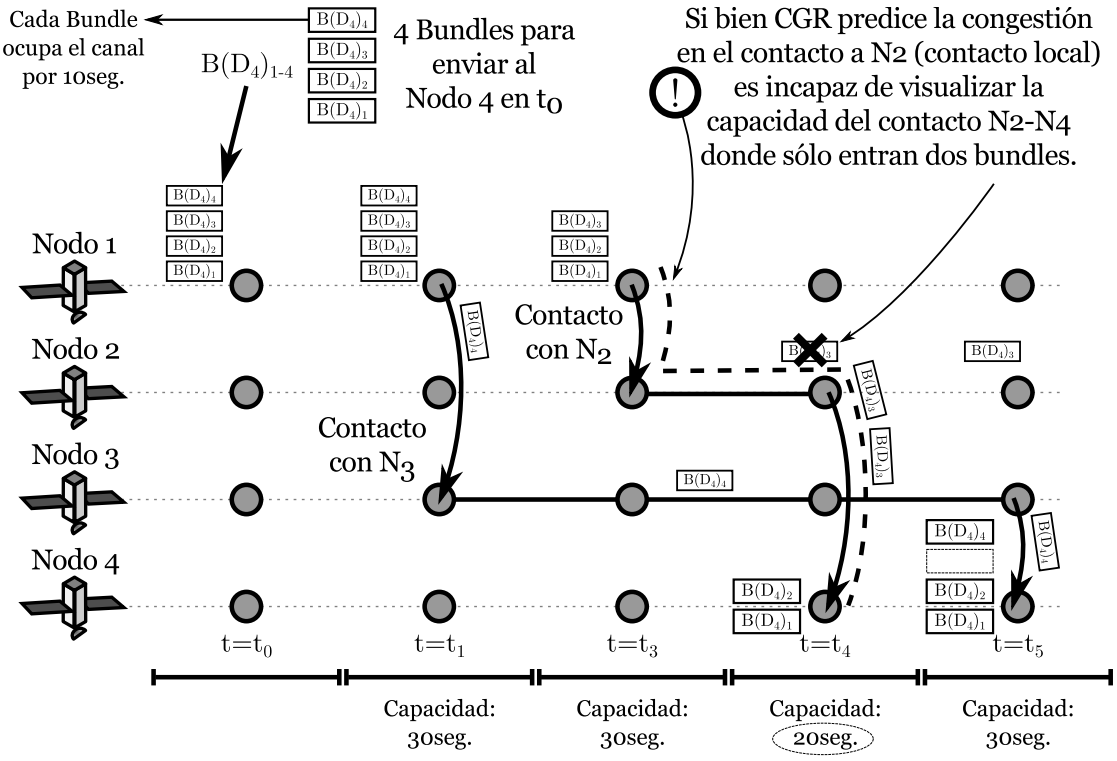


FIGURA 6.10: Problema de la congestión en contactos remotos con CGR

CGR denominada CGR con Registro de Ruta o *Path-Aware CGR* (PA-CGR) [10] en Inglés. PA-CGR hace uso de información disponible en el plan de contacto sin necesidad de tener que inspeccionar cabeceras de paquetes en tránsito. En efecto, la información de las capacidades de los contactos no locales también esta codificada dentro del plan de contacto y en general sin uso eficiente en CGR [59]. De esta manera, PA-CGR utiliza información de capacidad a lo largo de toda la ruta calculada por CGR en lugar de solo el contacto local. En general, la capacidad de ruta o *route capacity* es determina por el

Algoritmo 8: Proceso de revisión de contacto con registro de ruta PA-CGR

global : $Forfeit = \infty$, $RouteCap$, Ecc, $Route = \emptyset$, $BestDel$,$ExclNodes \leftarrow PrevHop$, $Cplan$
input : D, X
output: $ProxNodes$

```
1  ExclNodes ← D;
2  PrevForfeit = Forfeit;
3  PrevBestDel = BestDel;
4  PrevRouteCap = RouteCap ;
5  for ∀xmit ∈ Cplan | xmit_D = D, xmit_t.end ≤ X do
6      if D == xmit_D then
7          Route ← xmit;
8          RouteCap = xmit_Rate * (xmit_t.end − max(xmit_t.str, T));
9          Forfeit = xmit_t.end;
10         BestDel = max(xmit_t.str, T);
11     else
12         Route ← xmit;
13         if xmit_ResCap < RouteCap then
14             RouteCap = xmit_ResCap;
15         if xmit_t.end < PrevForfeit then
16             Forfeit = xmit_t.end;
17         if xmit_t.str > PrevBestDel then
18             BestDel = xmit_t.str;
19     if RouteCap > Ecc then
20         if xmit_S == ThisNode then
21             ProxNodes ← D(xmit, Route...);
22         else
23             if xmit_S ∉ ExclNodes then
24                 CGR-PA-CRP(xmit_S, xmit_t.end)
25     Route.Pop;
26     RouteCap = PrevRouteCap;
27     Forfeit = PrevForfeit;
28     BestDel = PrevBestDel;
29 ExclNode.Pop;
30 return ProxNodes;
```

contacto de menor capacidad residual o *residual capacity* en un camino en particular. En consecuencia, cuando CGR declara que un camino es posible por medio del vecino N_x, el procedimiento de PA-CGR decrementa las capacidades de toda la ruta a diferencia de hacerlo solamente sobre el contacto local, expandiendo la capacidad de gestión de la congestión a contactos remotos.

La implementación de PA-CGR puede ser fácilmente integrada con la existente de CGR dado que sólo implica cambios menores en el algoritmo como mostramos en el Algoritmo 8. En particular PA-CGR incorpora dos nuevas variables globales: una lista de rutas ($Route$) y la capacidad de la misma ($RouteCap$) las cuales se mantendrán a medida que las recursiones del algoritmo avancen. El *stack* de recursión se inicializa en las lineas 1 a 4 para luego iterar entre los diferentes contactos del plan de contacto ($xmit$). Al igual que en CGR los contactos se evalúan por nodo destino, por lo que las variables de tiempo máximo de envío ($Forfeit$) y mejor tiempo de entrega ($BestDel$) se inicializan en las lineas 7 a 10 en caso de encontrar el vecino D; en su defecto, las mismas se actualizan en las líneas 12 a 18. Luego, en la línea 19 se evalúa si la ruta calculada puede enrutar el tamaño de dato requerido. En caso de no encontrar el destino, se ejecuta la recursión en la línea 23. Si la capacidad no es suficiente el contacto es descartado en las líneas 26 y 26 para finalmente recuperar las variables globales en las líneas 27 y 28.

Cabe destacar que PA-CGR puede ser considerado para incluirse en el esquema C-CGR basado en caché tratado en la sección 6.4.1 con la leve modificación de que el puntero al contacto de la tabla de caché debe reemplazarse con uno por cada contacto en la ruta al destino de entrada D_x. De esta manera, PA-CGR permite extender la gestión de congestión de CGR mas allá del contacto local sin mayores modificaciones ni a los paquetes bundles ni al algoritmo en si mismo. En general PA-CGR es una mejora significativa a la congestión pero como se puede deducir, sólo trata la congestión física por capacidad de canal. Como se introdujo en la sección 6.2.2 también puede existir un fenómeno de congestión provocado por tráfico provenientes de otros nodos razón por la cual proponemos MG-CGR en la próxima sección 6.5.2.

6.5.2. CGR Multi Grafo

En esta sección describiremos un aporte original realizado al tratamiento de la congestión provocada por la existencia del tráfico de nodos vecinos. Denominaremos esta estrategia como CGR Multi Grafo o *Multi-Graph CGR* (MG-CGR) Inglés dado que nos basaremos en un esquema de planes de contacto independiente para cada nodo en lugar de uno único para toda la red.

Como se puede ver en la Figura 6.11, si bien PA-CGR ahora es capaz de predecir la congestión en el contacto remoto N_2 a N_4 (evitando enviar mas de dos bundles por esta ruta), es incapaz de predecir que el nodo N_2 tiene tráfico en cola de antemano para enviar al N_4. Este tráfico en la figura se ilustra como un paquete con una textura rayada, y ocupa el lugar en el canal de N_2 a N_4 en el estado t_4. Esto deriva en que bundle 2 quedará almacenado en N_2 sin ruta posible hacia el destino final N_4, cuando en realidad había una posibilidad de evacuarlo en t_2 por medio de N_3.

Vale aclarar que de disponer de mensajes asíncronos enviados oportunamente (precisamente a comienzos de t_4) como recomienda PCC se podría actualizar al nodo N_1 sobre el estado de la cola de N_2 a N_4, pero como se argumentó en la sección 6.3.2 aún no existe un criterio claro de como ni cuando generar estos mensajes. Además, lo mas temprano que el N_1 podría enterarse de esta realidad es en el estado t_4, tiempo para el cual la oportunidad de enviar el bundle remanente (por medio de N_1 a N_3 en el t_2) ya caducó.

Con el fin de solucionar esta problemática, e inspirados en los avances logrados con TACP en el capítulo 5, proponemos basarnos en la cualidad de predictibilidad del tráfico en redes DTN espaciales ya argumentada en la sección 5.1.1 del mismo capítulo. En efecto, como mostramos en la Figura 6.12, el *plan de tráfico* y la *topología de contacto* alimentan un esquema de asignación de tráfico como por ejemplo TACP-LP. Este último es quien utiliza estas dos valiosas informaciones para determinar un plan de contacto final para

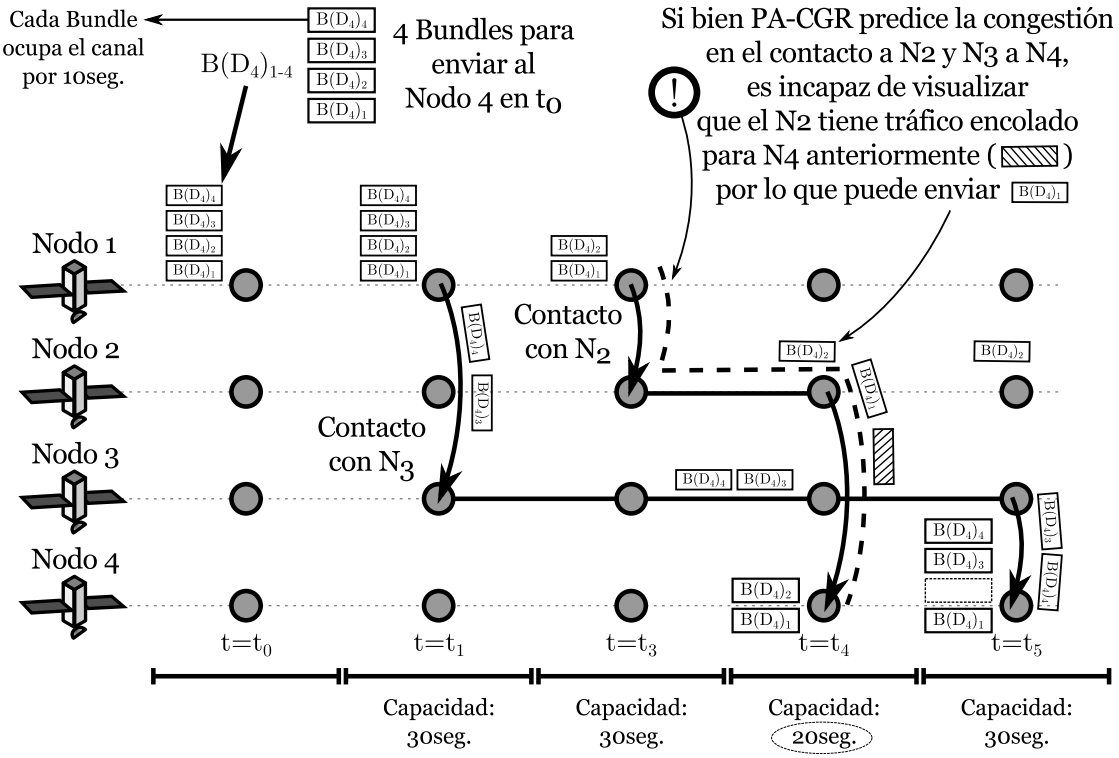


FIGURA 6.11: Problema de la congestión por tráfico remoto con PA-CGR

cada nodo donde se especifican solamente las capacidades de contacto asociadas al tráfico generada por ese nodo. Los contactos que ese nodo no utilice deberán ser incluidos en el plan de contacto pero con una capacidad de 0 por razones que explicaremos a continuación.

Una vez que los nodos reciban su plan de contacto específico, los mismos lo utilizarán para calcular las rutas de su propio tráfico (es decir, generado localmente) con PA-CGR [10] utilizando una base de datos de las capacidades a nivel sistema. Luego, codificarán la ruta en la cabecera utilizando un esquema como CGR con bloque de extension o Extension Block CGR (EB-CGR) [60] antes de enviarlo al próximo salto. En consecuencia, el próximo nodo utilizará esa información de cabecera para decidir el próximo contacto por medio del cual reenviar el paquete. De esta manera el nodo intermedio no ejecuta nuevamente CGR y no necesita saber las capacidades que ese tráfico afectará en el camino. Es decir, el tráfico se enruta en origen utilizando las capacidades específicamente detalladas en el plan de contacto de ese nodo. Los demás nodos honrarán esa decisión y a los sumo podrán ejecutar una validación del camino original (para lo cual necesitan tener registro de todos los contactos del sistema) pero sin modificarlo.

Como mostramos en la Figura 6.12, la parte intensa de procesamiento toma lugar en el diseño del plan de contacto donde un nodo centralizado (típicamente un centro de control de misión o MOC) puede utilizar el conocimiento del plan de tráfico, topología de contacto, y las características del sistema para decidir la asignación de tráfico por un esquema como TACP-LP o TACP-GA si la topología necesita de diseño o simplemente el modelo MILP de enrutamiento del mismo TACP si no hay conflictos de restricciones. Con esta información se puede derivar la manera óptima en la que debe fluir el tráfico con el fin de optimizar tanto su tiempo de entrega como evitar completamente la congestión. En efecto, de esta asignación se puede determinar que capacidad de la topología en su totalidad será asignada a cada nodo para que disponga de ella libremente por medio de esquemas como PA-CGR. En otras palabras, MG-CGR es una partición del plan de contacto a nivel nodo en el cual se le asigna una porción del mismo a cada uno para su futuro uso. En esa asignación se evitan problemas de congestión y se garantiza que la demanda de tráfico original planteada en el diseño se pueda satisfacer siempre y cuando se considere un flujo a transmitir máximo por cada nodo. Aclararemos mas sobre este último punto en la sección 6.5.4.

Vale la pena aclarar que a pesar de que MG-CGR sólo implica la codificación de la ruta en la cabecera de los paquetes dejando la capacidad de los mismos afuera. Por otro lado, el mismo requiere que se distribuya un plan por nodo de manera específica impidiendo el uso de mecanismos multi-cast factibles de ser utilizado en caso de usar un esquema de un mismo plan para todos los nodos. En general, este es el costo que se paga por

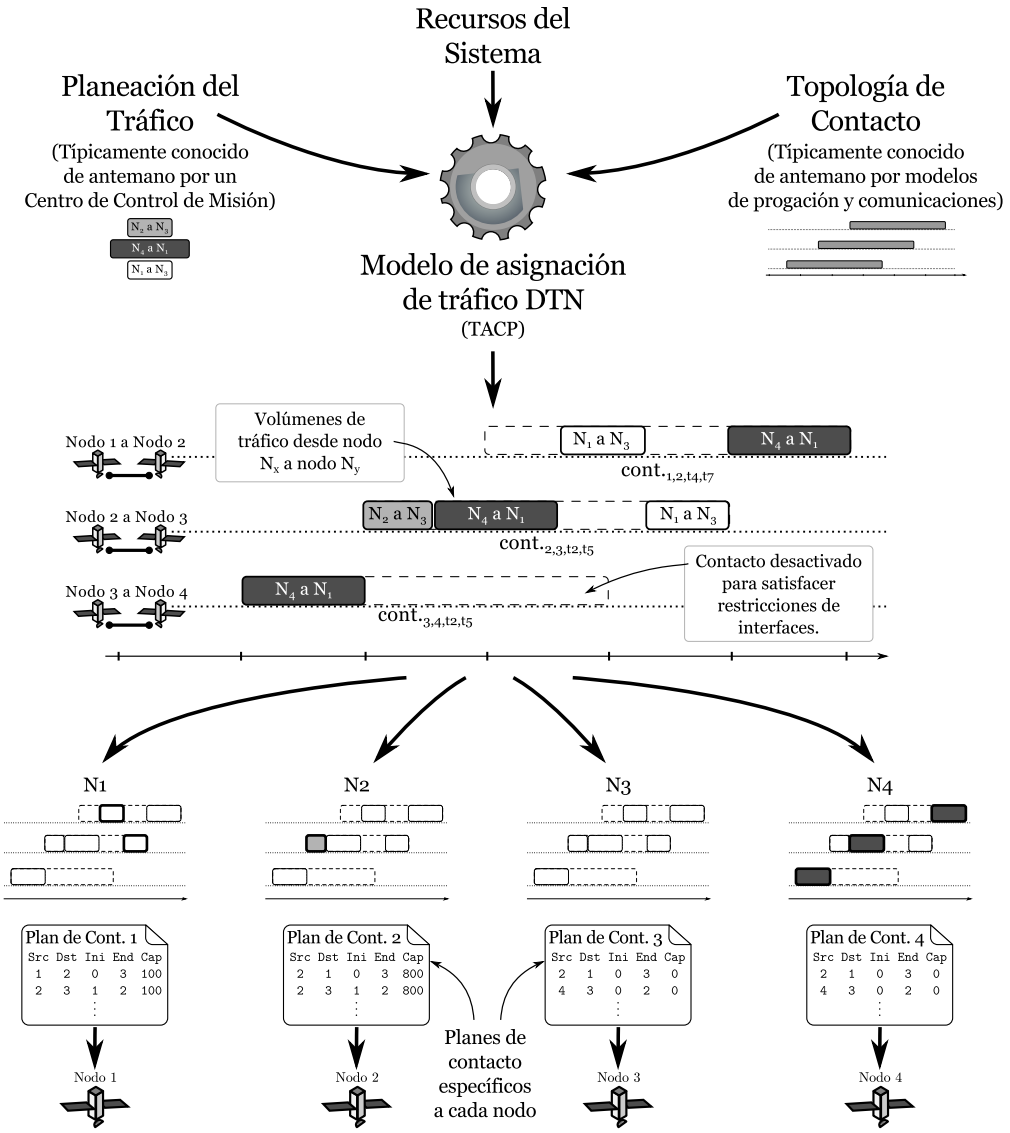


FIGURA 6.12: Procedimiento de CGR Multi Grafo o MG-CGR

tener un sistema de gestión de tráfico completamente libre de congestión y lazos de rutas sin llegar a distribuir una planificación paquete a paquete. En otras palabras MG-CGR aporta beneficios de importancia a la comunidad DTN con una sobrecarga (*overhead*) mas que aceptable en comparación con otros mecanismos menos eficientes como PCC.

Finalmente, y para resumir todas las técnicas de gestión de congestión aquí revisadas (CGR, PA-CGR, PCC, y MG-CGR tratadas en las secciones 6.3.1, 6.5.1, 6.3.2, y 6.5.2 respectivamente), proveemos la siguiente Tabla 6.5. En la misma distinguiremos las técnicas de mitigación o eliminación de la congestión de acuerdo a congestión provocada por límite de capacidad física o por tráfico de otros nodos.

Por último, cabe destacar que a medida que los esquemas son mas robustos ante los problemas de congestión, aumenta la capacidad de cada uno de ellos pueda implementar

Tabla 6.5: Comparación de capacidades de la gestión de congestión en DTN

	Sobrecarga (overhead)	**Información Utilizada**	**Gestión de Congestión por Capacidad**	**Gestión de Congestión por Tráfico**
CGR	Ninguna	Plan de Contacto único	Contactos locales	Ninguna
PA-CGR	Ninguna	Plan de Contacto único	Contactos locales y remotos	Ninguna
PCC	Ruta y Capacidad en cabecera	Plan de Contacto único	Contactos locales y remotos	Parcial por medio de aprendizaje
MG-CGR	Ruta en cabecera	Plan de Contacto específico por nodo y matriz de tráfico en planificador central	Contactos locales y remotos	Total por medio de planificador central

de manera correcta un plan de contacto diseñado por TACP. Esto se debe a que si no existen fenómenos de congestión ni por cuestiones físicas o de otros tráficos, se podrán obtener una asignación óptima de los flujos del sistema. En efecto, el hecho de que MG-CGR evite la congestión en lugar de reaccionar ante ella (como en PCC, PA-CGR o CGR) permite garantizar que siempre que se cumpla que cada nodo a lo sumo cuenta con una solo flujo de datos a un destino dado, el plan de contacto diseñado con TACP resulta implementable. Cuando un nodo cuenta con mas de un tráfico pueden surgir problemas por la cronología en la que debe operar CGR como describiremos en mayor detalle en la sección 6.5.4.

6.5.3. Análisis de Congestión

6.5.3.1. Descripción del Escenario

Con el fin de evaluar la capacidad de gestión de la congestión de cada una de las técnicas descritas a lo largo de la sección 6.5 de este capítulo, retomaremos el caso de estudio y referencia C detallado en la sección 4.4.3 del capítulo 4. Al considerar la inclusión de comunicaciones con una estación terrena podremos generar un cuello de botella realista en el cual podremos apreciar los problemas y capacidades de gestión de la congestión de cada estrategia.

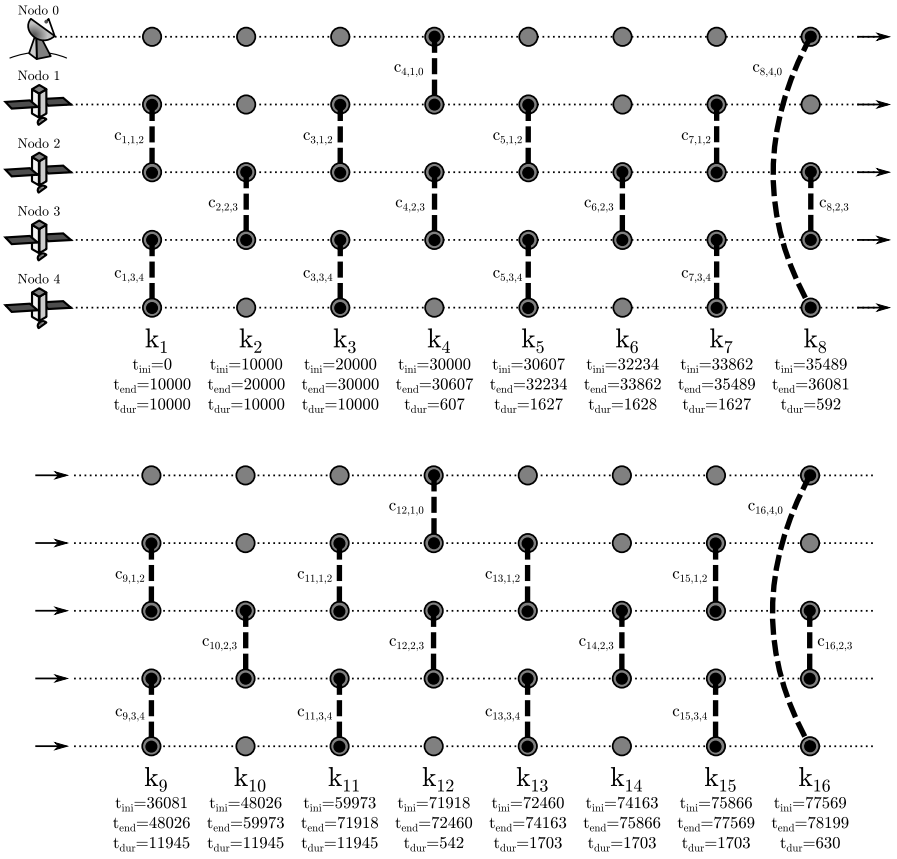


FIGURA 6.13: Topología del caso de estudio y referencia C diseñada con FCP

En particular, para este análisis incluiremos el nodo de la estación terrena o segmento de tierra (ESL) el cual aporta contactos de muy bajo expansión de tiempo (del orden de los 10 minutos promedios) en comparación con los enlaces ISL de duración prácticamente permanente. Además, mantendremos la restricción de un equipo de comunicaciones por satélite por lo que se requiere un diseño de la topología por lo que utilizaremos el criterio basado en rutas (RACP) [4] con un patrón de rutas en la que todos los nodos transmiten a la estación terrena. En total evaluaremos 4 pasadas por la estación terrena de ubicación en Córdoba, Argentina ($-65°$ Latitud y $-32°$ Longitud) donde el nodo N_1 y N_4 alternarán su uso del transponder de bajada a tierra en cada uno para balancear el desgaste energético. La Figura 6.13 ilustra la topología resultante para un tiempo de topología listado en la Tabla 6.6.

En este escenario traficaremos datos desde todos los satélites hacia el segmento terreno en Córdoba identificado como nodo N_0. Combinando un enlace de $100Kbps$ con un

TABLA 6.6: Parámetros de tiempo para obtener 4 pasadas por la estación terrena Córdoba

Inicio de intervalo de topología	Ene-1, 2015, 0hs 0min 0seg
Fin de intervalo de topología	Ene-1, 2015, 21hs 43min 18seg

tamaño de paquetes (bundles) de 12,5$KBytes$ (1 paquete por segundo) y un tiempo total de contacto con tierra de 1149 segundos por el nodo N_1 y 1222 del nodo N_4, el sistema total debería ser capaz de entregar una bajada total de 29,637$MBytes$ o 2371 paquetes en el tiempo de topología propuesto en la Tabla 6.6. En efecto, el máximo tráfico que cada nodo podrá enviar es de 7,4 MBytes o 592 paquetes. De ahora en adelante consideraremos esta carga de red como 1 ($\rho = 1$) para la cual se genera un total de 2371 en todo el sistema. Sin embargo, la exitosa entrega de esta carga supone una gestión apropiada del tráfico (y su congestión) en la constelación. En particular, esto último no resulta trivial en un caso como el planteado donde el contacto con N_0 es claramente un cuello de botella y se alterna entre los dos extremos de la formación lineal en la que se basa el caso de referencia C.

Finalmente, ejecutaremos diferentes simulaciones para PA-CGR, PCC, y MG-CGR y el esquema de enrutamiento óptimo MILP basado en las ecuaciones (5.1) a (5.7) de TACP (sin diseño dado que la topología ya está diseñada de antemano en este caso como se muestra en la Figura 6.13). Incluiremos este último para medir la diferencia de cada esquema con una cota superior de rendimiento. En efecto, en caso de que alguno de estos esquemas coincida con el resultado del modelo MILP estaremos en un caso de plan de contacto implementable con TACP.

Consideraremos como métricas para la comparación de estos esquemas la cantidad total de carga útil (*payload* o *delivered packets* en Inglés) efectivamente entregada a la estación terrena (N_0). Sin embargo también es necesario entender como esta entrega se realiza por lo que también mediremos el tiempo total de contacto (ISL y ESL) de sistema (*system contact time* en Inglés) así como el tiempo de entrega (*delivery time* en Inglés). Finalmente, para obtener estos resultados volveremos a utilizar el simulador basado en Omnet++ (TotSim) introducido en la sección 2.6.6 y ya utilizado en otros capítulos de esta tesis.

6.5.3.2. Resultados

Los resultados de simulación obtenidos se resumen en las curvas de la Figura 6.14 donde el eje de las abscisas representan las diferentes carga de tráfico a la que se somete la red desde $\rho = 0{,}08$ (50 paquetes o 625$KByte$ por nodo) hasta $\rho = 1$ (592 paquetes o 7,4$MByte$ por nodo). En el eje de las ordenadas, se muestra la medición de las métricas *Delivery ratio*, *system contact time*, y *delivery time* para cada uno de los esquemas CGR, PA-CGR, PCC, y el modelo MILP de TACP sin diseño. En particular, la Figura 6.14 c) muestra 4 áreas destacadas representando los estados k_4, k_8, k_{12} y k_{16} donde la constelación tiene contactos con la estación terrena en Córdoba por medio de los arcos

$c_{4,1,0}$, $c_{8,4,0}$, $c_{12,1,0}$ y $c_{16,4,0}$ (ver Figura 6.13). En general, y como se espera de acuerdo a lo discutido a lo largo de este capítulo, CGR y el modelo MILP de TACP se ubican siempre como cotas inferior y superior de rendimiento respectivamente.

En general, para bajos volúmenes de tráfico, todos los esquemas de congestión permiten

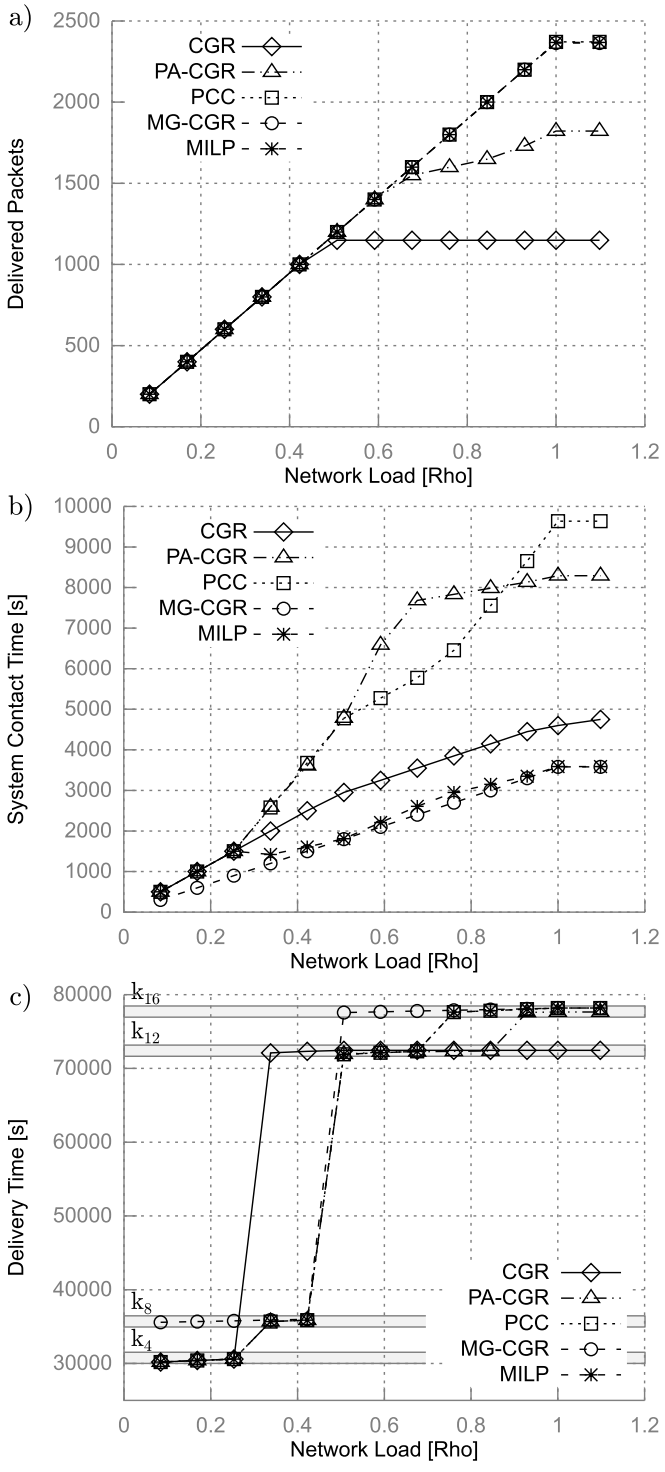


FIGURA 6.14: Resultados de la comparación entre PA-CGR, PCC, MG-CGR, y el modelo óptimo MILP

garantizar una entrega óptima de la carga útil de los datos generados (Figura 6.14 a)). Sin embargo un punto de inflexión claro se muestra para CGR para una carga de $\rho = 0{,}5$ (300 paquetes por nodo o 1200 en total en el sistema). Para poder explicar en detalle este fenómeno nos referiremos a la Figura 6.15 a) la cual ilustra los flujos de tráfico ($tf_{i,j}$ desde el nodo i a j) para CGR para el caso particular de $\rho = 1$. En esta gráfica, CGR solamente hace uso de los contactos $c_{4,1,0}$ y $c_{12,1,0}$ (con una capacidad agregada de 1149) para alcanzar al nodo de estación terrena N_0 con una entrega máxima de 1149 paquetes para $\rho \geq 0{,}5$. Esto es principalmente por que cuando cada nodo enruta el tráfico generado por ellos mismos consideran que el contacto $c_{4,1,0}$ completamente disponible ignorando su capacidad remanente luego de ser utilizada por los otros tráficos (recordemos que CGR solo es consciente de los propios tráficos en contactos locales). En efecto, sólo el nodo N_1 es capaz de acusar noticia de que la capacidad física del contacto $c_{4,1,0}$ se agota al recibir tráfico de $tf_{2,0} : 592$ en k_1, y $tf_{3,0} : 592$ mas $tf_{4,0} : 592$ en k_3. Sin embargo, a pesar de que este nodo puede calcular rutas alternativas por medio de $c_{8,4,0}$, el mismo no puede utilizarla debido a la política de no retorno a nodo previo explicada en la sección 6.3.1.2. En consecuencia, el tráfico queda estancado hasta que un segundo contacto a la estación terrena se hace factible en el contacto en el estado k_{12}. Este fenómeno hace que el sistema

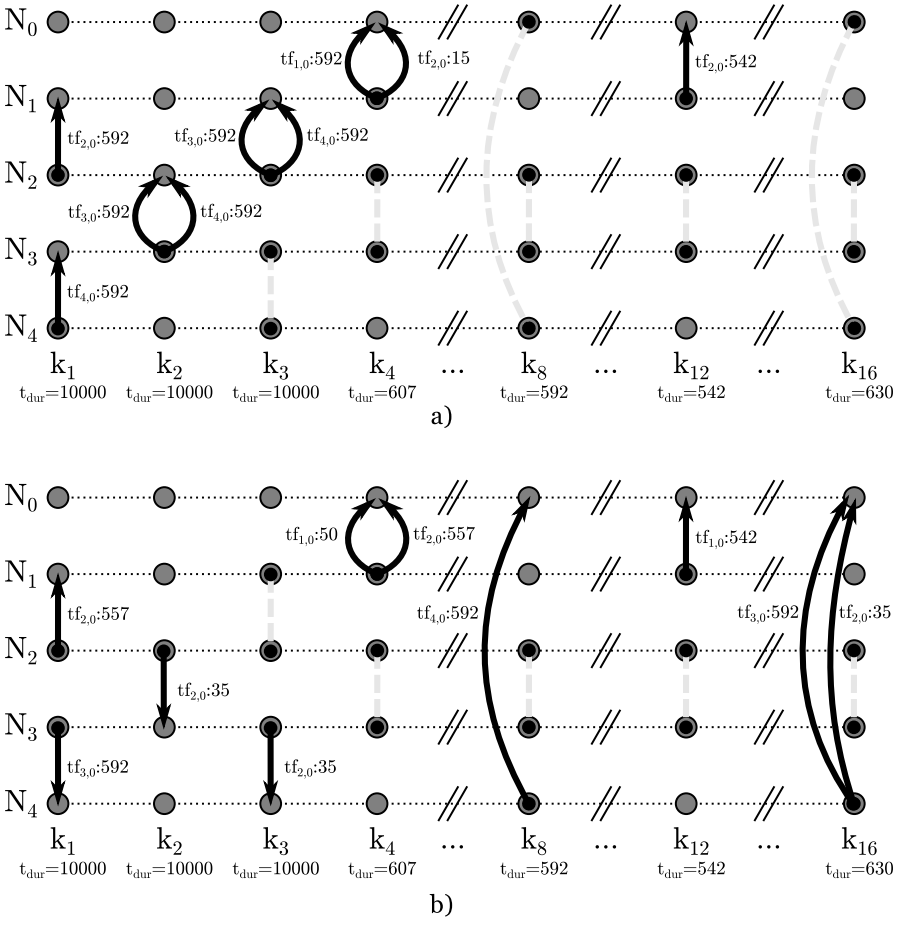


FIGURA 6.15: Flujo de tráfico para a) CGR, y b) modelo MILP de TACP para $\rho = 1$

de CGR evidencie un uso de los recursos relativamente bajo (con pequeños incrementos a medida que aumenta ρ), con un tiempo de entrega dentro del estado k_{12} para $\rho \geq 0{,}3$ (mas de 607 paquetes). El echo de que CGR muestre mejores tiempo de entrega que el modelo MILP en la Figura 6.14 c) se explica al observar de que tan temprana recepción se logra sólo para un total de 1149 paquetes en lugar de los 2368 totales. Finalmente, vale la pena notar que en caso de quitar la política de retorno al nodo previo en CGR deriva en un tiempo total de sistema superior a los 86000 segundos por todos los lazos de enrutamiento que se genera.

Por otro lado, Path-Aware CGR mejora la métrica de entrega de paquetes al permitir a los nodos N_4, N_3, y N_4 predecir y reaccionar a la congestión física por capacidad de contacto de antemano. En contraste con los flujos de CGR mostrados en la Figura 6.15 a), ni el nodo N_3 ni el N_2 eligen enviar 1184 (592+592) paquetes en los estados k_2 o k_3. Esto es por que son capaces de ver que la capacidad del futuro contacto $c_{4,1,0}$ estará completamente utilizada antes de que el tráfico efectivamente arribe al nodo local N_1. En consecuencia, los paquetes son enviados por rutas alternativas incluyendo los contactos $c_{8,4,0}$, $c_{12,1,0}$ y $c_{16,4,0}$ como se puede inducir por los tiempos de entrega de la Figura 6.14 c). Sin embargo, debido a que PA-CGR ignora los tráficos de los otros nodos (ver Tabla 6.5), una importante carga de tráfico es re-enrutada desde el nodo N_1 hacia nodos anteriores quienes forman lazos de rutas temporarias hasta agotar las capacidades de los contactos. Claramente este es un efecto no deseado que hace que PA-CGR solamente pueda alcanzar un 76 % de entrega de la máxima capacidad del sistema.

Por otro lado, la capacidad de PCC de aprender el tráfico de otros nodos por medio de la inspección de cabeceras de paquetes, ayuda a acelerar las actualizaciones de las base de datos de capacidades locales de cada nodo sobre todo de los contactos no locales. Esto permite que PCC pueda entregar la totalidad de la carga del sistema inclusive para el máximo volumen de tráfico ($\rho = 1$ o 2368 paquetes). Sin embargo, a nivel tiempo de contacto de sistema, PCC muestra métricas similares a las de PA-CGR (es decir, existe cierto efecto de *rebote* por reacción a la congestión). En esta métrica, un mayor uso de los recursos se puede notar a medida que se acerca la saturación del sistema ($\rho = 1$). Vale aclarar y recordar que en estas simulaciones ignoramos el efecto de sobrecarga en el volumen de tráfico causado por la codificación de rutas y capacidades residuales en la cabecera de los paquetes de PCC. Esta última puede tener efectos significativos en el rendimiento final que dejamos como tema de investigación futura (ver sección 6.3.2). Por último, es interesante destacar que en caso de aplicar la política de no retorno a nodo previo en PPC y PA-CGR hace que ambos entreguen las mismas métricas que CGR debido a que se le quita la capacidad de reaccionar ante la congestión.

Por último, la mejor tasa de entrega o *delivery ratio*, uso de recursos, y tiempo de entrega se logran solo con el modelo MILP derivado de TACP en el capítulo 5. Dado que un enrutamiento de esta naturaleza puede utilizar el conocimiento de toda la topología en su totalidad (además de la matriz de tráfico), el mismo permite evitar completamente la congestión y optimizar el uso de los enlaces (sin ningún efecto *rebote* como en PA-CGR y PCC). Esto permite a este esquema reducir la utilización de recursos a un 32 % en comparación con PA-CGR y PCC. Los flujos de tráfico ilustrados en la Figura 6.15 b) para $\rho = 1$ permiten sostener el argumento previo. En particular, el nodo N_3, en lugar de intentar enviar sus 592 paquetes al primer contacto disponible $c_{4,1,0}$ como en todas técnicas previas, decide confiar en el N_4 el que puede mas adelante hacer uso de otros contactos con el nodo N_0 ($c_{8,4,0}$ y $c_{16,4,0}$) generando un flujo prolijo y balanceado en la red. Lo mismo sucede con el nodo N_2 que envía 35 paquetes hacia un nodo anterior dado que los mismos no entrarán en la capacidad residual de los contactos $c_{4,1,0}$ y $c_{12,1,0}$ una vez que el nodo N_1 complete su transacción con el nodo N_0.

Como se discutió anteriormente, implementar el modelo MILP de TACP de manera distribuida en cada nodo resulta prácticamente imposible en escenarios reales por cuestiones de requerimientos de cómputo y de distribución de la información necesaria. En consecuencia propusimos MG-CGR como alternativa para obtener las mismas prestaciones sin necesitar de estas condiciones utópicas para un sistema DTN satelital. Esto lo logra con el uso de planes de contactos específicos por cada nodo que en este caso particular deben codificar una topología capaz de evacuar los 592 paquetes hacia el nodo N_0. Por ejemplo, en la Figura 6.15 b) el plan de contacto para el nodo N_1 solamente deberá contar con los contactos $c_{4,1,0}$ y $c_{12,1,0}$ con una capacidad de 50 y 542 paquetes (o segundos) respectivamente. De manera análoga, el nodo N_3 deberá informarse de los contactos $c_{1,3,4}$ y $c_{16,4,0}$ ambos con una capacidad de 592 paquetes. Esta especificidad de planes de contactos permiten a MG-CGR a desempeñarse óptimamente hasta la carga máxima del sistema ($\rho = 1$). Sin embargo, a medida que el tráfico del sistema difiere del planificado (asumimos que el planificado es $\rho = 1$ para todos los casos de ρ en MG-CGR), el esquema MG-CGR evidencia una leve variación en las métricas de tiempo de contacto de sistema y tiempo de entrega del modelo MILP. En particular, una penalidad de tiempo insignificante de entrega se observa para $0 \leq \rho \leq 0{,}3$ y $0{,}5 \leq \rho \leq 0{,}7$.

De esta manera, MG-CGR se posiciona como una excelente y prometedora propuesta para la gestión del tráfico de redes DTN predecibles. Además, en casos con flujos único por nodos (como el aquí evaluado), permite garantizar la implementabilidad de esquemas de diseño complejos de alto rendimiento como TACP explicado en el capítulo 5. En la próxima sección 6.5.4 discutimos algunos temas a tener en cuenta con MG-CGR respecto a su implementabilidad y tolerancia a cambios en la red.

6.5.4. Análisis de Implementabilidad

Es importante destacar que MG-CGR, al igual que TACP está diseñado para gestionar el tráfico planificado de manera exacta y precisa. En efecto, en caso de que la cantidad de datos en el sistema real sea diferente puede causar comportamientos negativos para el rendimiento final del sistema. En particular, si se genera mas tráfico que el planificado, sencillamente el plan de contacto no tendrá capacidad de evacuarlo quedando el mismo (o el original) estancado en algún nodo intermedio u origen. En general, para evitar esto se puede considerar el modelado de márgenes de guarda para el tráfico del lado de TACP, o bien en incluir un segundo plan de contacto por nodo en MG-CGR que cuente con *contactos de contingencia* que puedan ser usados en caso de que la capacidad asignada al plan de contacto específico se agote en ese nodo. En efecto, estas son estrategias necesarias a considerar en la etapa de implementación de MG-CGR.

Por otro lado, y en general, los planes de contactos diseñados con TACP tienen escasa implementabilidad con esquemas como CGR debido a los problemas de congestión que el mismo evidencia. Con esquemas mejorados como PA-CGR y PCC la capacidad de respetar lo calculado por TACP mejora, pero no se mantiene para la mayoría de los casos como se mostró en la sección 6.5.3.2. Sin embargo, se mostró que MG-CGR permite garantizar la implementabilidad del modelo de enrutamiento MILP de TACP para casos con un único flujo por nodo.

Sin embargo, cuando los nodos empiezan a contar con múltiples generadores de tráficos, MG-CGR puede diferir de la planificación de TACP como mostramos a continuación. Por ejemplo, si consideramos el simple escenario planteado en la Figura 6.16, un plan de contacto con dos fuentes de tráfico en el nodo N_1 se deben enrutar para los nodos destinos N_2 y N_3 respectivamente con una flujo de 10 unidades de capacidad a lo largo de 4 arcos de la misma capacidad de 10 unidades. En la Figura 6.16 a) se muestra el diseño de plan de contacto con TACP por medio del cual se puede aplicar MG-CGR. En efecto, el plan de contacto enviado al nodo N_1 contará con los tres primeros arcos (el arco $c_{6,1,3}$ no es necesario) que según el modelo MILP de asignación son suficientes para evacuar todo el tráfico al final del estado k_5. Sin embargo, cuando este plan se implemente en el N_1 sucederá lo que se muestra en la Figura 6.16 b), donde el primer tráfico que se genera ($tf_{1,2}$) tiene una ruta óptima por medio del contacto $c_{3,1,2}$. Esto no solo que difiere con lo calculado por TACP si no que interrumpe la ruta del tráfico generado en el estado siguiente (k_2) hacia el nodo N_3 que se queda sin capacidad de ser evacuado. En efecto, el plan de contacto resulta *no implementable* por mas que se utilice MG-CGR.

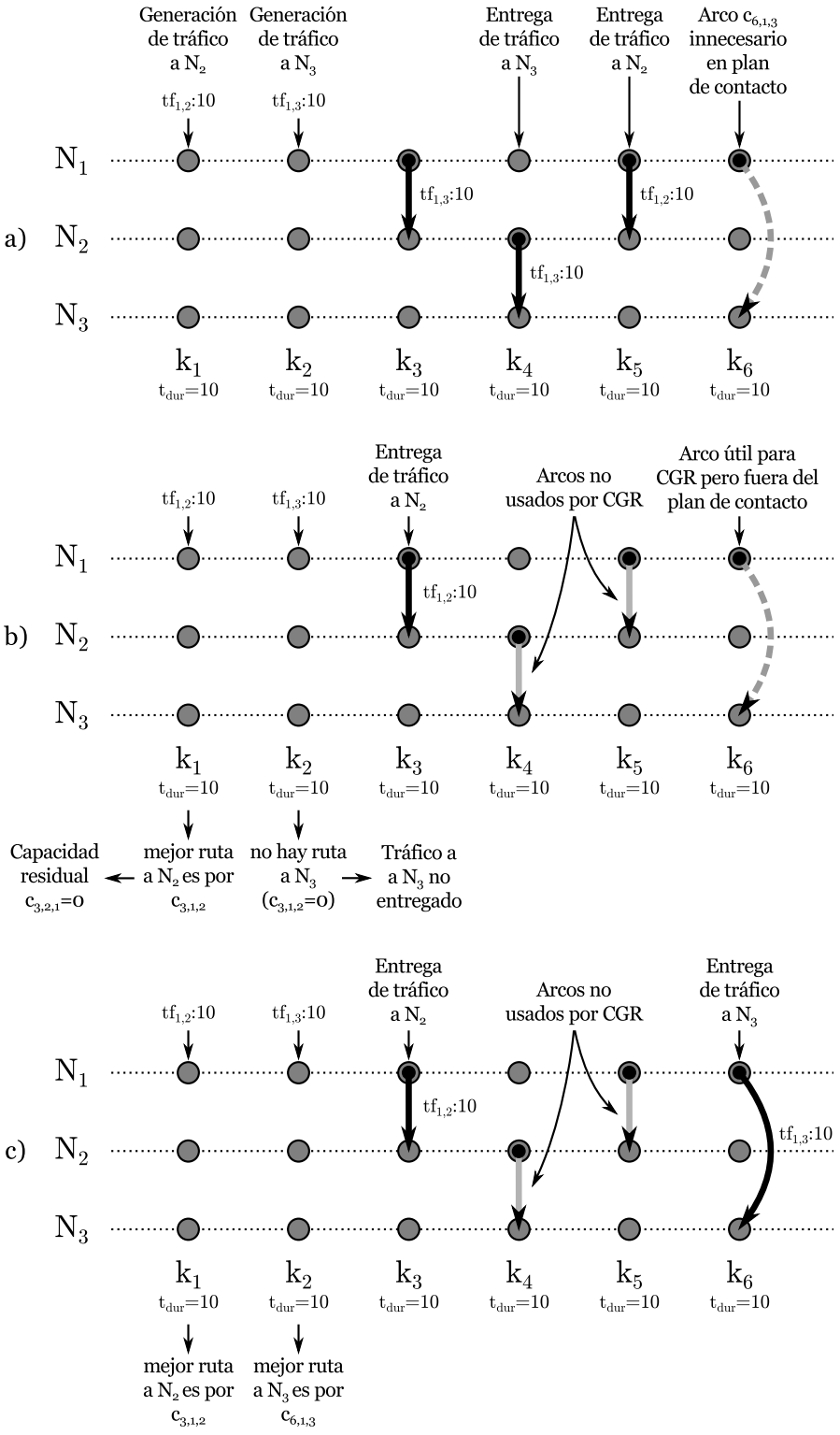


FIGURA 6.16: Plan diseñado por TACP en a), flujo final con MG-CGR sobre el mismo en b) y un posible plan implementable en c)

En general, la condición de no implementabilidad aquí mostrada se basa en que en MG-CGR el algoritmo CGR (al igual que si derivado PA-CGR) calcula las rutas a medida que el tráfico se genera en el nodo. En consecuencia, en la cronología de este cálculo

puede suceder que no se respete el flujo óptimo y se generen diferencias por mas que se cuente con un único plan de contacto por nodo. Claramente este fenómeno no sucede para los casos de la sección 6.5.3 donde se considera un solo flujo (generador de tráfico) por nodo.

Una posible solución al problema es extender la aplicabilidad de MG-CGR a un plan de contacto por flujo de tráfico en lugar que uno por nodo. Efectivamente esto llevaría a que se levante la propiedad cronológica descrita pero derivaría en la necesidad de un numero significativamente mayor de planes de contactos en el sistema. En efecto, un MG-CGR de estas características se asemejaría mucho a un esquema de enrutamiento paquete a paquete el cual intentamos evadir por la distribución de información necesaria para el mismo. Otra solución mas interesante es considerar este efecto en el diseño y buscar diseñar planes de contactos *implementables*, es decir, que no sufran de la situación mostrada en la Figura 6.16 b). Un ejemplo de esto es que el esquema de diseño genere un plan de contacto como el mostrado en Figura 6.16 c), donde si bien el tráfico fluye de forma sub-óptima ($deliveryTime = 60$) en comparación con el plan de contacto a) ($deliveryTime = 50$), el mismo resulta implementable en nodos con PA-CGR utilizando un único plan de contacto. Dejaremos el estudio de dicho procedimiento como trabajo a futuro y posible continuación de esta tesis doctoral.

6.6. Comentarios Finales de la Implementación de Planes de Contacto

En este capítulo planteamos diferentes problemáticas asociadas a la implementación de los planes de contactos diseñados por los diferentes esquemas desarrollados en los capítulos 3, 4 y 5 de esta tesis. En particular describimos temáticas relacionadas a las diferencias de decisiones tomadas por algoritmos de enrutamiento distribuidos en DTN respecto de los centralizados utilizados en el diseño de planes de contacto. En efecto encontramos que los problemas de congestión son la principal causa de deficiencias en el desempeño de los mismos, así como que los mecanismos existentes demandaban un esfuerzo computacional significativo para su funcionamiento.

En consecuencia describimos C-CGR [9], un aporte específico al área de la eficiencia de procesamiento de CGR en la sección 6.4 el cual fue publicado en la conferencia IEEE International Conference on Wireless for Space and Extreme Environments (WiSEE) en el 2014 en Noordwick, Holanda. Luego propusimos mejoras a los esquemas de reacción a la congestión existentes con la descripción de PA-CGR y MG-CGR [10] con importantes resultados en comparación con los mecanismos existentes a espera de ser publicados en

la conferencia IEEE Conference on Local Computer Networks (LCN) en Florida a finales del 2015. Al día de la fecha el grupo de trabajo junto con colegas del Laboratorio APL de NASA esta desarrollando un artículo de revista que extienda los resultados obtenidos con MG-CGR.

De esta manera concluimos la descripción del flujo de investigación desarrollado a lo largo de la tesis que inicialmente exploró diferentes mecanismos de diseño de plan de contactos, para derivar en el mas eficiente basado en el tráfico de la red (TACP). En particular, este capítulo permitió discutir cuestiones de implementación específicas de este último esquema para acercar su uso a una red DTN satelital real.

Capítulo 7

Conclusión

En el inicio de esta tesis planteamos la problemática del diseño de *plan de contactos* para redes de comunicaciones tolerante a disrupciones (DTN) para la observación terrestre basada en sistemas distribuidos e interconectados de satélites autónomos. Dicho planteo surge de que la comunidad DTN ha ignorado que las limitaciones de recursos típicas de estas plataformas espaciales pueden limitar la efectivización de las futuras oportunidades de comunicación. En consecuencia, definimos el *diseño de plan de contactos* como el proceso de configurar y elegir apropiadamente las oportunidades de comunicación de antemano con el fin de gestionar y controlar la utilización del sistema de acuerdo a las características de sus recursos y al mismo tiempo optimizarlo bajo un criterio dado para mejorar el flujo de datos [5]. De esta manera, se planteó la hipótesis inicial de que el rendimiento de un sistema limitado en recursos mejora con la mayor información incorporada para el diseño del plan de contactos, la cual buscamos verificar con la contribución y evaluación de algunos esquemas automáticos de diseño inexistentes al comienzo del proceso de investigación.

A lo largo de la tesis, perseguimos la validación de la hipótesis al proceder de manera incremental en lo relativo a la información utilizada para ejecutar el diseño de plan de contacto. A lo largo del trabajo, hemos sido capaces de generar tres estadíos claros de disponibilidad de información para el diseño de planes de contacto cuyos rendimiento y complejidad demostraron ser directamente proporcionales a la cantidad de datos considerados. En particular, luego de describir el marco conceptual en el capítulo 1 y de establecer las bases del problema y modelado del mismo en el capítulo 2, propusimos una primera instancia de diseño llamada FCP [2, 3] basada solamente en la predicción de la topología de contacto en el capítulo 3. En el capítulo 4 incluimos la información de la rutas entre los nodos para derivar RACP [4], para luego proponer TACP [5, 8]: un esquema que asume que se conoce con exactitud el tráfico que será generado en el sistema

en el capítulo 5. En efecto, aprovechamos que la generación de datos también suele ser predecible dado que los instrumentos científicos o de carga útil a bordo de los satélites son gestionados desde tierra, y la telemetría de plataforma se genera en volúmenes y períodos conocidos de antemano. De esta manera, la contribución realizada por FCP, RACP y TACP ha permitido avanzado la frontera del estado del arte de la planificación de redes espaciales tolerantes a demoras con limitaciones de recursos.

Por otro lado, al lograr el mejor rendimiento del diseño de plan de contacto en TACP, nos percatamos de que el mismo suponía la existencia de algoritmos de enrutamiento distribuidos óptimos a bordo de los satélites, lo cual no es necesariamente cierto de acuerdo a lo observado en el estado del arte de DTN. En consecuencia, y con la visión de asignación de tráfico óptima ya desarrollada y comprendida para TACP, fuimos capaces de delinear estrategias originales que faciliten y mejoren la eficiencia de implementación o aplicación de los planes de contactos diseñados con este esquema. En efecto, en el capítulo 6 describimos C-CGR [9], PA-CGR y MG-CGR [10] como aportes derivados pero no menos importantes para el mejor uso de los procesadores de vuelo, la gestión y la anulación de la congestión respectivamente. Estos aportes secundarios resultan de vital importancia para complementar las tareas de planificación con algoritmos eficientes en órbita que permitan que la red satelital pueda reaccionar independiente ante imprecisiones de planificación o fallas de los segmentos de vuelo. De hecho, y como explicaremos en la Sección 7.1, esta área es una importante derivación de esta tesis en la que actualmente el grupo de trabajo esta realizando aportes.

En general, un esquema de diseño de plan de contacto de la eficiencia de TACP, en combinación con RACP y FCP, así como sus respectivas formas adecuadas de implementarlos, promete tener un impacto significativo en mejorar y optimizar la entrega de información desde una red espacial satelital. En efecto, al explotar toda la información disponible de un sistema orbital, este paradigma de planificación y operación está permitiendo cambiar la forma en que las constelaciones satelitales son tradicionalmente concebidas. En particular, dado que estas misiones se desarrollan en el espacio exterior, las mismas deben operar en un entorno inaccesible y de características extremas, lo que las vuelve particularmente costosas y con período de vida útil acotado. En consecuencia, optimizar la entrega de datos así como el uso de sus recursos como se buscó a lo largo de esta tesis, resulta extremadamente valioso para comunidad espacial y la Argentina en particular, sentando las bases para evaluar y fomentar la implementación de soluciones DTN para las futuras redes satelitales funcionales y operativas.

7.1. Trabajo Futuro

El desarrollo doctoral aquí plasmado se ha enfocado en proponer estrategias eficientes de desarrollo de planes de contactos (conjunto de oportunidades futuras de comunicaciones) para que los nodos DTN puedan optimizar el uso de los recursos generalmente escasos en plataformas espaciales. En general, estas estrategias se han basado en la predictibilidad de estas redes, asumiendo que no existen imperfecciones en la predicción y que los nodos operan en condiciones carente de fallas. Si bien esto resulta de suma utilidad en la comunidad y es válido en términos generales, estas suposiciones pueden no resultar del todo precisas particularmente en el ámbito espacial donde las predicciones cuentan con un margen de incertidumbre y las fallas de los componentes ocurren con mayor frecuencia que en la tierra.

En consecuencia, como proyecto postdoctoral se ha iniciado una extensión de las estrategias desarrolladas en la tesis doctoral para que las mismas puedan contemplar casos de fallas en los nodos así como imprecisiones en las predicciones orbitales. Cumplimentar este objetivo resulta de suma importancia de cara a implementar sistemas DTN en redes operativas dado que si bien las mismas se revisten de este carácter inherentemente predictivo, también es necesario considerar márgenes de incertidumbre hasta el momento no contempladas en la comunidad. En este contexto, actualmente se está investigando la problemática particular de la supervivencia en estas redes de transporte de datos, entendida como la capacidad de una red de comunicaciones para recuperar el transporte de datos ante diferentes escenarios de falla. La supervivencia depende de la disponibilidad de recursos ociosos (en Inglés, *spare*), los cuales ante un escenario sin fallas no están en uso pero que permiten recuperar el transporte de datos ante un escenario de falla.

Esta investigación se está llevando a cabo en el marco de proyectos PICT-2015 y PIDDEF-2015 en vías de evaluación. En particular, este nuevo campo de trabajo podría resultar de particular interés para instituciones locales actualmente relacionadas con el grupo de investigación como la Fábrica Argentina de Aviones (FADeA) en su proyecto de redes UAVs (también predictivas, pero con márgenes de incertidumbre mayor) así como la Comisión Nacional de Actividades Espaciales (CONAE) en proyectos de arquitecturas de redes satelitales segmentadas (programa SARE). En particular, el desarrollo de esta nueva actividad fortalecerá la capacidad de transferencia y vinculación tanto del Laboratorio de Comunicaciones Digitales (LCD) como de la del Laboratorio de Circuitos y Sistemas Robustos (LCSR) actualmente en formación en la FCEFyN.

Bibliografía

[1] **Fraire, J.**, P. Ferreyra, and C. Marques. Opencl overview, implementation, and performance comparison. *Latin America Transactions, IEEE (Revista IEEE America Latina)*, 11(1):274–280, Feb 2013. ISSN 1548-0992.

[2] **Fraire, J.**, P. Madoery, and J. Finochietto. On the design and analysis of fair contact plans in predictable delay-tolerant networks. *IEEE Sensors Journal*, 14 (11):3874–3882, Aug 2014.

[3] **Fraire, J.**, P. Madoery, and J. Finochietto. On the design of fair contact plans for delay tolerant networks. In *2013 IEEE International Conference on Wireless for Space and Extreme Environments (WiSEE)*, 2013.

[4] **Fraire, J.**, P. Madoery, and J. Finochietto. Routing-aware fair contact plan design for predictable delay tolerant networks. *Elsevier Ad-Hoc Networks*, 25:303–313, Feb 2015.

[5] **Fraire, J.** and J. Finochietto. Design challenges in contact plans for disruption-tolerant satellite networks. *Communications Magazine, IEEE*, 53(5):163–169, May 2015. ISSN 0163-6804. doi: 10.1109/MCOM.2015.7105656.

[6] **Fraire, J.**, P. Madoery, and J. Finochietto. Contact plan design for predictable disruption tolerant space sensor networks, 2015. Chapter 15 of Wireless Sensor Systems for Extreme Environments: Space, Underwater, Underground and Industrial (ISBN: 978-1-119-12646-1). Editors: H. Rashvand and A. Abedi, Wiley, In Press.

[7] **Fraire, J.A.** Traffic aware contact plan design for scheduled disruption tolerant networks. Technical Report LCD-1504-01, Digital Communications Research Lab, National University of Córdoba, Apr 2015.

[8] **Fraire, J.**, P. Madoery, and J. Finochietto. Preliminary results of an evolutionary approach towards contact plan design for satellite dtns. In *Proceedigns of 2015 IEEE International Conference on wireless for space and extreme environments (WiSEE)*, Orlando, Florida, USA, Dec. 2015. In Press.

[9] **Fraire, J.**, P. Madoery, and J. Finochietto. Leveraging routing performance and congestion avoidance in predictable delay tolerant networks. In *2014 IEEE International Conference on Wireless for Space and Extreme Environments (WiSEE)*, pages 1–7, Noordwick, Netherlands, October 2014. IEEE.

[10] **Fraire, J.**, P. Madoery, J. Finochietto, and E. Birrane. Congestion modeling and management techniques for predictable disruption tolerant networks. In *Proceedings of IEEE Conference on Local Computer Networks (LCN)*. IEEE, Oct 2015.

[11] **Fraire, J.** and Ferreyra P. Assessing dtn architecture reliability for distributed satellite constellations: Preliminary results from a case study. In *2014 IEEE Biennial Congress of Argentina (ARGENCON)*, pages 564–569, June 2014. Bariloche, Argentina.

[12] M. Alfonzo, **Fraire, J.**, E. Kocian, and N. Alvarez. Implementation and evaluation of a space-wire convergence layer adaptor. In *IEEE Argencon*, June 2014. Bariloche, Argentina.

[13] P. Freire. *Educación y Cambio*. Galerna, Buenos Aires, 1 edition, 2002.

[14] R. Velazco, P. Fouillat, and R. Reis. *Radiation Effects on Embedded Systems*. Springer, 2007.

[15] Herbert Hecht. *Systems Reliability and Failure Prevention*. Artech House, November 2003. ISBN-10: 1580533728.

[16] O. Brown and P. Eremenko. The value proposition for fractionated space architectures. In *AIAA-2006-7506, AIAA Space 2006*, San Jose, CA, 2006.

[17] DARPA Tactical Technology Office. System f6 program. `http://www.darpa.mil/Our_Work/TTO/Programs/System_F6.aspx`.

[18] C. Joppin and D. Hastings. On-orbit upgrade and repair: The hubble space telescope example. *Journal of Spacecraft and Rockets*, 43(3):614–625, May 2006.

[19] H.F. Rashvand, A. Abedi, J.M. Alcaraz-Calero, P.D. Mitchell, and S.C. Mukhopadhyay. Wireless sensor systems for space and extreme environments: A review. *IEEE Sensors Journal*, 14(11):3955–3970, Sep 2014.

[20] N.A. Goodman, Sih Chung Lin, D. Rajakrishna, and J.M. Stiles. Processing of multiple-receiver spaceborne arrays for wide-area sar. *Geoscience and Remote Sensing, IEEE Transactions on*, 40(4):841–852, Apr 2002. ISSN 0196-2892.

[21] NASA: A-Train, 10.26.10 (april 2012). `http://atrain.nasa.gov/`.

[22] Paul Muri and Janise McNair. A survey of communication sub-systems for inter-satellite linked systems and cubesat missions. *Journal of Communications*, 7(4), 2012.

[23] C. Jing, G. Jian, and E. Gill. Fractionated space infrastructure for long-term earth observation missions. In IEEE, editor, *IEEE Aerospace Conference*, pages 1–9, Big Sky, MT, 2003.

[24] J. Postel. RFC-793: Transmission control protocol specification. Request for Comments RFC 793, Network Working Group, IETF, Sep 1981.

[25] R. Durst, G. Miller, and E. Travis. Tcp extensions for space communication. *ACM/Kluwer WINET Journal*, 3(5):389–403, Oct 1997.

[26] J. Jackson. The interplanetary internet. *IEEE Spectrum*, 42(8):31–35, Aug 2005.

[27] W.J. Larson and J.R. Wertz. *Space Mission Analysis and Design, 3rd edition*, volume 8. Microcosm, Inc., Torrance, CA (US), 3 edition, 1999.

[28] Miniwatts Marketing Group. World stats. in: Internet world stats. `http://www.internetworldstats.com/`, June 2012.

[29] F. Abduljalil and S. Bodhe. A survey of integrating ip mobility protocols and mobile ad hoc networks. *IEEE Communications Surveys and Tutorials*, 9(1), 2007 2007.

[30] M Al-Siyabi, H. Cruickshank, and Z. Sun. Delay and disruption tolerant network architecture for aircrafts datalink on scheduled routes. *Personal Satellite Services*, 43:235–248, 2010.

[31] J. Scholl and A. Lindgren. Considering pigeons for carrying delay tolerant networking based internet traffic in developing countries. *Electronic Journal of Information Systems in Developing Countries*, 54, 2012.

[32] K. Doowon. A survey of balloon networking applications and technologies. In *CSE570S*. Washington University, St. Luis, 2013. Available On-line: `http://www.cse.wustl.edu/~jain/cse570-13/ftp/balloonn.pdf`.

[33] T.P. Garrison, M. Ince, J. Pizzicaroli, and P.A. Swan. Systems engineering trades for the iridium constellation. *Journal of Spacecraft and Rockets*, 34(5):675–680, Oct 1997.

[34] C. Caini, H. Cruickshank, S. Farrell, and M. Marchese. Delay and disruption tolerant networking: An alternative solution for future satellite networking applications. In *Proceedings of the IEEE*, volume 99, pages 1980–1997, November 2011.

[35] M. Muhammad, M. Berioli, and T. de Cola. A simulation study of network-coding-enhanced pep for tcp flows in geo satellite networks. In *Communications (ICC), 2014 IEEE International Conference on*, pages 3588–3593, June 2014. doi: 10.1109/ICC.2014.6883878.

[36] C. Caini and V. Fiore. Moon to earth dtn communications through lunar relay satellites. In *6th Advanced Satellite Multimedia Systems Conference (ASMS) and 12th Signal Processing for Space Communications Workshop (SPSC)*, pages 89–95, Baiona, Sep 2012.

[37] TEDx-MidAtlantic. Vint cerf: Interplanetary internet. `http://youtu.be/XTmYm3gMYOQ`, 2011.

[38] S. Burleigh and A. Hooke. Delay-tolerant networking: an approach to interplanetary internet. *IEEE Comms. Magazine*, (41):128–136, 2003.

[39] I.F Akyildiz, O.B. Akan, C. Chen, J. Fang, and W. Su. Wireless sensor systems for space and extreme environments: A review. *Elsevier Computer Networks Journal*, 43(2):75–113, Oct 2003.

[40] K. Fall. A delay-tolerant network architecture for challenged internets. In *Proceedings of ACM SIGCOM*, August . Karlsruhe, Germany.

[41] C. Caini and R. Firrincieli. DTN for LEO satellite communications. pages 186–198. Springer, 2011.

[42] Internet Engineering Task Force (IETF). Delay Tolerant Networking Working Group (DTNWG). `https://datatracker.ietf.org/wg/dtnwg/charter/`.

[43] Delay Tolerant Network Research Group (DTNRG), . `http://www.dtnrg.org`.

[44] InterPlanetary Networking Special Interest Group (IPNSIG). `http://ipnsig.org`.

[45] V. Cerf et al. RFC-4838: Delay-tolerant networking architecture. Network Working Group, IETF, April 2007.

[46] K. Scott and S. Burleigh. RFC-5050: Bundle protocol specification. Network Working Group, IETF, November 2007.

[47] *Information technology - Open Systems Interconnection – Basic Reference Model: The basic model (ISO/IEC 7498-1)*. ITU-T Study Group 17, 1994.

[48] M. Demmer, J. Ott, and S. Perreault. RFC-7242: Delay-tolerant networking tcp convergence-layer protocol. Request for Comments RFC 7242, Internet Research Task Force (IRTF), Jun 2014.

[49] H. Kruse, S. Jero, and S. Ostermann. RFC-7122:datagram convergence layers for the delay and disruption tolerant networking (dtn) bundle protocol and licklider transmission protocol (ltp). Request for Comments RFC 7122, Internet Research Task Force (IRTF), Mar 2014.

[50] Lloyd Wood, Wesley M Eddy, and Peter Holliday. A bundle of problems. In *Aerospace conference, 2009 IEEE*, pages 1–17. IEEE, 2009.

[51] J. Burgess, B. Gallagher, D. Jensen, and B. Levine. Maxprop: Routing for vehicle-based disruption-tolerant networks. IEEE INFOCOM, April 2006.

[52] A. Balasubramanian, B. Levine, and A. Venkataramani. DTN routing as a resource allocation problem. ACM SIGCOMM, August 2007.

[53] T. Spyropoulos, K. Psounis, and S. Cauligi. Spray and wait: an efficient routing scheme for intermittently connected mobile networks. ACM SIGCOMM, 2005.

[54] Yahui Wu, Su Deng, Hongbin Huang, and Yiqi Deng. Performance analysis of epidemic routing in delay tolerant networks with overlapping communities and selfish nodes. *International Journal of Computers Communications and Control*, 8(5), 2013.

[55] Tracy Camp, Jeff Boleng, and Vanessa Davies. A survey of mobility models for ad hoc network research. *Wireless Communications and Mobile Computing*, 2(5): 483–502, 2002. ISSN 1530-8677.

[56] Yong Wang, Wei Peng, Xilong Mao, and Zhenghu Gong. Rwpad: A mobility model for dtn routing performance evaluation. In *Embedded and Ubiquitous Computing (EUC), 2010 IEEE/IFIP 8th International Conference on*, pages 460–465, Dec 2010.

[57] Anders Lindgren, Avri Doria, and Olov Schelén. Probabilistic routing in intermittently connected networks. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7(3): 19–20, July 2003. ISSN 1559-1662. doi: 10.1145/961268.961272.

[58] A. Lindgren, A. Doria, E. Davies, and Grasic. RFC-6693: Probabilistic routing protocol for intermittently connected networks.

[59] S. Burleigh. Contact graph routing, IETF-Draft. Jul 2010.

[60] E. Birrane, S. Burleigh, and N. Kasch. Analysis of the contact graph routing algorithm: Bounding interplanetary paths. *Acta Astronautica*, 75:108–119, July 2012.

[61] S. Merugu, M. Ammar, and Zegura E. Routing in space and time in networks with predictable mobility. Technical report, Georgia Institute of Technology, 2006. Tech Report GIT-CC 04-7.

[62] DTN2: a DTN reference implementation, . `http://www.dtnrg.org/wiki/Dtn2Documentation`.

[63] Sebastian Schildt, Johannes Morgenroth, Wolf-Bastian Pöttner, and Lars Wolf. Ibrdtn: A lightweight, modular and highly portable bundle protocol implementation. In *Electronic Communications of the EASST*, 2011.

[64] IBR-DTN: A modular and lightweight implementation of the bundle protocol. `http://trac.ibr.cs.tu-bs.de/project-cm-2012-ibrdtn`.

[65] Postellation: a Lean and Deployable DTN Implementation. `http://postellation.viagenie.ca/`.

[66] A. Sudarsono and T. Nakanishi. An implementation of secure data exchange in wireless delay tolerant network using attribute-based encryption. In *Computing and Networking (CANDAR), 2014 Second International Symposium on*, pages 536–542, Dec 2014.

[67] S. Burleigh. Interplanetary overlay network: An implementation of the dtn bundle protocol. In *4th IEEE Consumer Comms. and Networking Conf., CCNC 2007*, pages 222–226, Las Vegas, NV, 2007.

[68] Interplanetary Overlay Network (ION). `http://sourceforge.net/projects/ion-dtn/`.

[69] Wolf-Bastian Pottner, Johannes Morgenroth, Sebastian Schildt, and Lars C Wolf. An empirical performance comparison of DTN bundle protocol implementations. In *ACM MobiCom 2011 Workshop on Challenged Networks (CHANTS'11)*, Las Vegas, Nevada, USA, 9 2011.

[70] S. Burleigh. RFC-6260: Compressed bundle header encoding specification. Network Working Group, IETF, Nov 2011.

[71] E. Birrane. Internet-Draft: Streamlined bundle security protocol specification. Network Working Group, IETF (`https://datatracker.ietf.org/doc/draft-birrane-dtn-sbsp/`), May 2014.

[72] E. Birrane and V. Ramachandran. Internet-Draft: Delay tolerant network management protocol. Network Working Group, IETF (`https://tools.ietf.org/html/draft-irtf-dtnrg-dtnmp-01`), Dec 2014.

[73] M. Demmer and J. Ott. Delay-tolerant networking tcp convergence-layer protocol. Network Working Group, IETF (`https://tools.ietf.org/html/rfc7242`), Jun 2014.

[74] H. Kruse, S. Jero, and S. Ostermann. Datagram convergence layers for the delay- and disruption-tolerant networking (dtn) bundle protocol and licklider transmission protocol (ltp). Network Working Group, IETF (`https://tools.ietf.org/html/rfc7122`), Mar 2014.

[75] Spacewire standard document ECSS-E-ST-50-12C. ESA Publications Division, July 2008. The Netherlands.

[76] A. Senior, J.-P. Coetzee, and J. Ilstad. The draft ecss spacewire backplane standard. In *SpaceWire Conference (SpaceWire), 2014 International*, pages 1–4, Sept 2014.

[77] Aeroflex-Gaisler SPARC V8 32-bit Processor LEON3FT CompanionCore Data Sheet. `http://www.actel.com/ipdocs/LEON3_DS.pdf`.

[78] Real-Time Executive for Multiprocessor Systems (RTEMS). `http://www.rtems.org`.

[79] J. Wyatt, S. Burleigh, J. Ross, L. Torgerson, and Wissler S. Disruption Tolerant Networking Flight Validation Experiment on NASAs EPOXI Mission. In *IEEE SPACOMM*, 2009.

[80] author. Inquiry into the heart of a comet. *Science and Children*, 48(6):46–49, Feb 2011.

[81] W. Ivancic, W. Eddy, D. Stewart, L. Wood, J. Northam, and C. Jackson. Experience with delay-tolerant networking from orbit. *Int. Journal of Satellite Comms. and Networking*, 28(5-6):335–351, September 2010.

[82] L. Wood, W. Ivancic, D. Hodgson, E. Miller, B. Conner, S. Lynch, C. Jackson, A. Da Silva Curiel, D. Cooke, D. Shell, J. Walke, and D. Stewart. Using internet nodes and routers onboard satellites. *International Journal of Satellite Communications and Networking*, pages 195–216, 2007.

[83] L. Wood, W. Eddy, C. Smith, W. Ivancic, and C. Jackson. Internet-Draft: Saratoga: A scalable data transfer protocol. Network Working Group, IETF (`https://tools.ietf.org/html/draft-wood-tsvwg-saratoga-16`), Oct 2014.

[84] L. Wood, W.M. Eddy, W. Ivancic, J. McKim, and C. Jackson. Saratoga: a delay-tolerant networking convergence layer with efficient link utilization. In *Satellite*

and Space Communications, 2007. IWSSC '07. International Workshop on, pages 168–172, Sept 2007.

[85] L. Wood, D. Shell, W. Ivancic, B. Conner, E. Miller, D. Stewart, and D. Hodgson. Cleo and vmoc: enabling warfighters to task space payloads. In *Military Communications Conference, 2005. MILCOM 2005. IEEE*, pages 3052–3058 Vol. 5, Oct 2005.

[86] L. Wood and P. Holliday. Internet-Draft: Using http for delivery in delay/disruption-tolerant networks. Network Working Group, IETF (`https://tools.ietf.org/html/draft-wood-dtnrg-http-dtn-delivery-09`), Jun 2014.

[87] C. Caini and R. Firrincieli. Application of contact graph routing to LEO satellite DTN communications. In *IEEE International Conference on Communications (ICC)*, pages 3301–3305, June 2012.

[88] D. A. Vallado. *Fundamentals of Astrodynamics and Applications - 4th Edition.* Microcosm, Hawthorne, CA, 2007.

[89] M. Huang, S. Chen, Y. Zhu, and Y. Wang. Cost-efficient topology design problem in time-evolving delay-tolerant networks. In *IEEE GLOBECOM*, pages 1–5, Dec 2010.

[90] M. Huang, S. Chen, F. Li, and Y. Wang. Topology design in time-evolving delay-tolerant networks with unreliable links. In *IEEE GLOBECOM*, pages 5296–5301, 2012.

[91] Paul Muri and Janise McNair. A survey of communication sub-systems for inter-satellite linked systems and cubesat missions. *Journal of Communications*, 7(4), 2012.

[92] K. Sidibeh and Tanya Vladimirova. Ieee 802.11 optimisation techniques for inter-satellite links in leo networks. In *Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference*, volume 2, pages 1177–1182, Feb 2006.

[93] Kawsu Sidibeh. *Adaption of the IEEE 802.11 protocol for inter-satellite links in LEO satellite networks.* PhD thesis, University of Surrey, 2008. Thesis submitted for the Degree of Doctor of Philosophy, University of Surrey. Copyright remains with the author.

[94] Tanya Vladimirova and K. Sidibeh. Wlan for earth observation satellite formations in leo. In *Bio-inspired Learning and Intelligent Systems for Security, 2008. BLISS '08. ECSIS Symposium on*, pages 119–124, Aug 2008.

[95] A. Krishnamurthy and R. Preis. Satellite formation, a mobile sensor network in space. In *Proceedings of 19th IEEE International Parallel and Distributed Processing Symposium*, Apr 2005.

[96] N. Bezirgiannidis, F. Tsapeli, Diamantopoulos S., and Tsaoussidis V. Towards flexibility and accuracy in space dtn communications. In *Proceedings of the 8th ACM MobiCom workshop on Challenged networks (CHANTS)*, pages 43–48. ACM, 2013.

[97] E. Birrane. Congestion modeling in graph-routed delay tolerant networks with predictive capacity consumption. In *2013 IEEE Global Communications Conference (GLOBECOM)*, pages 3016 – 3022, Atlanta, GA, December 2013. IEEE.

[98] AGI Systems Tool Kit (STK). `http://www.agi.com/STK`.

[99] H. Mendoza and G. Corral-Briones. Interferencia en sistemas distribuidos de satélites de orbita media y baja. In *XV Reunión de Trabajo Procesamiento de la Información y Control (RPIC)*, pages 1110–1115, San Carlos de Bariloche, Argentina, Sep 2013.

[100] ITU-R. Recommendation itu-r s.1325-3. International Telecommunication Union, 2003.

[101] J. Jaffe. Bottleneck flow control. *IEEE Transactions On Communications*, 9(7): 954–961, 1981.

[102] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955. ISSN 1931-9193.

[103] J. Edmonds. Maximum matching and a polyhedron with 0-1 vertices. *Journal of Research at the National Bureau of Standards*, 69:125–130, 1965.

[104] J. Edmonds. Path, trees, and flowers. *Canadian Journal of Mathematics*, 17: 449–467, 1965.

[105] V. Kolmogorov. Blossom v: A new implementation of a minimum cost perfect matching algorithm. *Mathematical Programming Computation*, 1(1):43–67, Jul 2009.

[106] G. Schäfer. Weighted matchings in general graphs. Master's thesis, Universität des Saarlandes, Saarbrücken, Germany, 2000.

[107] R. Jain et al. A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. Technical report, DEC Research Report TR-301, 1984.

[108] Makhorin, Andrew. GLPK (GNU Linear Programming Kit). `https://www.gnu.org/software/glpk/`. Department for Applied Informatics, Moscow Aviation Institute, Moscow, Russia.

[109] J. Walker. Satellite constellations. *J . Brit. Interplanetary Soc.*, pages 559–571, 1984.

[110] P. Muri et al. Topology design and performance analysis for networked earth observing small satellites. In *MILCOM Proceedings*, pages 1940–1945, Baltimore, Maryland, Nov 2011.

[111] A Varga. OMNeT++ Discrete Event Simulation System. `http://www.omnetpp.org/doc/manual/usman.html`.

[112] Consultative Committee for Space Data Systems (CCSDS). *CCSDS Proximity-1 Space Link Protocol Data Link Layer, Blue Book*. Number 4. Jul 2006.

[113] E. Birrane. Internet-Draft: Contact graph routing extension block. Network Working Group, IETF (`https://tools.ietf.org/html/draft-irtf-dtnrg-cgreb-00`), Oct 2013.

[114] R.W. Floyd. Algorithm 97: Shortest path. *CACM*, 1962.

[115] E. Dijkstra. A note on two problems in connexion with graphs. *NUMERISCHE MATHEMATIK*, 1(1):269–271, 1959.

[116] V. Pareto. Cours d'économie politique. *Rouge: Lausanne*, 1 and 2, 1896.

[117] El-Ghazali Talbi. *Metaheuristics: From Design to Implementation*. Wiley, June 2009. ISBN: 978-0-470-27858-1.

[118] Hwang C.L. et al. *Multiple objective decision making, methods and applications: a state-of-the-art survey*. Springer-Verlag, 1979.

[119] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(13):671–680, 1983.

[120] V. Černý. Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Optimization Theory and Applications (JOTA)*, 45: 41–51, 1985.

[121] W.H. Press et al. *Numerical Recipes: The Art of Scientific Computing*, volume Section 10.12. Cambridge University Press, 3rd ed. edition, 2007.

[122] A. Suppapitnarm et al. Simulated annealing: an alternative approach to true multi-objective optimization. *Eng. Opt.*, 33:59–85, 2000.

[123] B. Suman et al. A survey of simulated annealing as a tool for single and multiobjective optimization. *JORS*, (57):1143–1160, 2006.

[124] S. Even, A. Itai, and A. Shamir. On the complexity of time table and multicommodity flow problems. In *Foundations of Computer Science, 1975., 16th Annual Symposium on*, pages 184–193, Oct 1975.

[125] J. Alonso and K. Fall. A linear programming formulation of flows over time with piecewise constant capacity and transit times. Techical Report IRB-TR-03-007, Intel, 2003.

[126] IBM. IBM ILOG CPLEX Optimizer. `http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/`.

[127] John H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. MIT Press, Cambridge, MA, USA, 1992. ISBN 0262082136.

[128] M. Pitkanen, A. Keranen, and Ott J. Message fragmentation in opportunistic dtns. In *2011 7th EURO-NGI Conference on Next Generation Internet (NGI)*, pages 1–6, Kaiserslautern, June 2011. IEEE.

[129] Advanced Microcontroller Bus Architecture Specification (V2.0) (AMBA). `http://www.arm.com`.