



Universidad  
Nacional  
de Córdoba

## Trabajo Especial

# Analisis multiobjetivo sobre DTNs

**Ulises Nicolás Torrella**

---

Licenciatura en Ciencias de la Computación  
Matr.-Nr.: 41152934

Directores: Pedro R. D'Argenio  
Juan A. Fraire



Esta obra está bajo una Licencia Creative Commons Atribución 4.0 Internacional.

---

# Analisis multiobjetivo sobre DTNs

Trabajo especial

de

Ulises Nicolás Torrella

Nombre

Numero de Matricula

Ulises Nicolás Torrella

41152934

Directores: Pedro R. D'Argenio  
Juan A. Fraire



Universidad  
Nacional  
de Córdoba

# Abstract

Las redes tolerantes a demoras (DTN) han sido propuestas y estudiadas como un protocolo acorde al problema de comunicación espacial e interplanetaria. Basándose en las dinámicas orbitales, un plan de contacto puede ser trazado a priori para proponer un plan de enrutamiento competente. En las constelaciones de satélites en la órbita terrestre baja (LEO) estos planes de contacto tienden a no ser fiables gracias a causas como el conocimiento impreciso de las posiciones y velocidades de los satélites, nodos defectuosos, y cambios impredecibles en las condiciones ambientales que pueden llevar a interferencia y pérdida de paquetes. Para resolver el enrutamiento en DTNs con planes de contacto inciertos los procesos de decisión de Markov (MDP) han sido utilizados para obtener planes de enrutamiento que garanticen una buena probabilidad de éxito en la transmisión de paquetes. Este trabajo profundiza este enfoque incorporando información sobre el consumo de energía y latencias de los contactos en el modelado de MDPs. Valiéndose de la herramienta STORM se realiza un análisis multi objetivo a una red espacial para exponer los compromisos entre los objetivos de minimización de tiempo y consumo de energía, con la maximización de la probabilidad de éxito.

Delay tolerant networks (DTN) have been proposed and studied as an architecture fit for deep space and interplanetary networking. Based on orbital dynamics a contact plan can be known in advance to propose a competent scheduler. Of interest for this work are low-Earth orbit (LEO) constellations in which uncertainty in contact plans can arise from various sources; such as imprecise knowledge of the satellite positions and velocities, fault nodes and unpredictable changes in the atmospheric conditions. To solve routing in DTNs under uncertain contact plans Markov decision processes (MDP) have been studied and implemented to find schedulers with a good success delivery probability (SDP). This thesis analyzes further the modeling of DTNs with MDPs in order to include energy consumption and delivery time. Applying multi-objective analysis on MDPs using the STORM model checker we can expose underlying trade offs of specific DTNs. As for proof of concept a Lunar Road Ring architecture is analyzed in these three objectives.

# Índice general

<b>Abstract</b>	<b>I</b>
<b>Nomenclatura</b>	<b>IV</b>
<b>1 Introducción</b>	<b>1</b>
1.1. MDPs para el enrutamiento . . . . .	1
1.2. Objetivos en conflicto . . . . .	2
1.3. STORM . . . . .	3
1.4. Estructura de la tesis . . . . .	3
<b>2 Teoría</b>	<b>5</b>
2.1. DTNs, planes de contacto y planes de contacto inciertos . . . . .	5
2.2. Costos de tiempo y energía . . . . .	6
2.3. PMC . . . . .	6
2.4. Proceso de Decisión de Markov . . . . .	7
2.5. Objetivos . . . . .	8
2.6. Puntos óptimos de Pareto . . . . .	9
2.7. Estrategias aleatorizadas . . . . .	10
2.8. Estrategias con memoria . . . . .	12
2.9. PCAA en STORM . . . . .	13
2.10. Conclusión del capítulo . . . . .	14
<b>3 Modelos</b>	<b>16</b>
3.1. Puntos de vista . . . . .	16
3.2. Idea . . . . .	16
3.3. Implementación . . . . .	18
3.4. Resultados . . . . .	19
3.5. Tablas de enrutamiento . . . . .	21
3.6. Topología LEO . . . . .	21
3.7. Conclusión de capítulo . . . . .	23
<b>4 Experimentación</b>	<b>24</b>
4.1. Metodología . . . . .	24
4.2. Lunar Ring Road . . . . .	24
4.3. Escenarios . . . . .	25
4.4. Representación . . . . .	26
4.5. Comparaciones . . . . .	26
4.5.1. Sin ISL . . . . .	26
4.5.2. Con ISL . . . . .	29
4.6. Elección de estrategias . . . . .	29
4.7. Diferencias entre redes con ISL y sin ISL . . . . .	32

---

<b>5</b>	<b>Discusión</b>	<b>33</b>
5.1.	A futuro . . . . .	33
5.2.	Caminos paralelos . . . . .	33
5.3.	Problemas no tomados en cuenta . . . . .	34
<b>6</b>	<b>Conclusión</b>	<b>35</b>
	Sin ISL . . . . .	36
	Con ISL . . . . .	39
	<b>Referencias</b>	<b>42</b>

# Nomenclatura

## Abreviaciones

Abreviación	Definición
DTN	Delay Tolerant Network
LEO	Low-Earth Orbit
MDP	Markov Decision Process
SDP	Success Delivery Probability
EDT	Expected Delivery Time
EEC	Expected Energy Cost
LRR	Lunar Road Ring

# 1

## Introducción

El desafío de desplegar redes satelitales en la órbita terrestre baja (LEO, low earth orbit) nos enfrenta a redes particularmente complicadas por su entorno. Interferencias en la red, escasez de opciones de enrutamiento y tiempos excesivos de round-trip hacen que estas redes no puedan valerse de los protocolos cotidianos vistos en redes terrestres. Para subsanar este tipo de problemas, mega-constelaciones como Starlink [11] pueden desplegar hasta 12.000 satélites para garantizar conectividad end-to-end de sus nodos, disminuir latencias y obtener así acceso en tiempo real a los datos de cada nodo participante de la red. Sin embargo, existen alternativas más sustentables que se pueden implementar cuando el acceso a los datos tiene tolerancia a una cierta demora. Las redes tolerantes a demoras (DTN, delay tolerant networks) [18] fueron propuestas en 2003 como una alternativa que se vale de la posibilidad de almacenamiento persistente de los nodos para conservar los paquetes en espera de una opción de enrutamiento futura.

El conocimiento a priori de las oportunidades de contacto es una virtud de las redes satelitales, que valiéndose de las órbitas de los nodos pueden trazar planes de contacto. Estos describen la conectividad de la red a lo largo del tiempo y sirve de entrada a algoritmos como CGR (enrutamiento en grafos de contacto) [7] que construye grafos acíclicos y dirigidos a partir del plan de contacto y lo sujeta a algoritmos de búsqueda tipo Dijkstra. El resultado es un plan de enrutamiento que describe las rutas a tomar para cada intención de envío.

Surgen complicaciones en este enfoque cuando los contactos difieren del plan original, ya sea por fallas o interrupciones en la transmisión, o por conocimiento incompleto/inexacto cuando se calcula el plan. Las DTNs espaciales son propensas a estos problemas, dando lugar a planes de contacto inciertos. Un plan de contacto incierto es, entonces, un plan de contacto acompañado de anotaciones para cada contacto con su probabilidad de éxito. La solución al enrutamiento en tales redes es una pregunta abierta de investigación [12].

### 1.1. MDPs para el enrutamiento

El problema de enrutamiento en planes de contacto inciertos coincide muy bien con las capacidades de modelado de los procesos de decisión de Markov (MDP, Markov decision process) combinando transiciones probabilísticas discretas, que representan fallas en el contacto; con transiciones no deterministas, que representan opciones de enrutamiento.

Dado un MDP podemos aplicar verificación de modelos probabilísticos (PMC, por sus siglas en ingles) para determinar la estrategia de enrutamiento con mayor probabilidad de entrega (SDP, success delivery probability). Esta idea se realizó de forma prototípica en [10], se elaboró profundamente en [16] y se estudió comparativamente en [12].

Las investigaciones mencionadas hasta ahora se enfocan en maximizar la probabilidad de éxito. Sin embargo, otros factores como el tiempo de entrega y el consumo de energía no han sido puestos en consideración por dichos trabajos. Es deseable tomar en consideración estos objetivos a la hora de elegir un plan de enrutamiento, considerando que quizás se encuentren en conflicto. La brecha entre el uso de MDPs para enrutamiento en DTNs espaciales y el análisis multi objetivo de MDPs es en la que se enfoca este trabajo.

Logrando tomar en valor objetivos que pueden variar su importancia circunstancialmente abrimos la puerta a el uso eficiente de redes de menor densidad. Por ejemplo, el tiempo de entrega resulta significativo para comunicación de control mientras que el consumo de energía es vital en estaciones espaciales donde los recursos energéticos no son abundantes. El análisis multi objetivo permite priorizar la elección de enrutamiento a partir de las circunstancias dadas.

## 1.2. Objetivos en conflicto

Es deseable para el enrutamiento en DTNs espaciales con planes de contacto incierto maximizar SDP mientras minimizamos el tiempo de entrega y el consumo de energía, teniendo en cuenta que estos objetivos pueden estar en conflicto. Veamos el ejemplo de la figura 1.1. Este es un plan de contacto incierto, donde se representa el nodo emisor ( $N_1$ ) en la parte más alta, y el nodo receptor ( $N_4$ ) en la más baja. Los espacios temporales avanzan horizontalmente y las flechas verticales indican una posibilidad de contacto anotada con su respectiva probabilidad de éxito. Consideramos el caso básico donde cada salto consume una unidad de energía.

Este plan de contacto incierto, diseñado para presentar estos objetivos en conflicto, tiene dos formas de llegar al nodo receptor. Una siguiendo un camino de tres saltos con bajas probabilidades de éxito, y otro más robusto de un solo salto pero con un tiempo de entrega necesariamente mayor.

Además, diferencias en el diseño del modelo pueden resultar en nuevas posibilidades de enrutamiento que encuentren un nuevo compromiso entre los objetivos. Por ejemplo, si el sistema permite duplicado de paquetes  $N_1$  puede enviarlos en sus dos contactos posibles aumentando su SDP y al mismo tiempo aumentando el consumo de energía esperado. Si se modela la red con asentimiento de arriba de paquetes (ACK) cuando  $N_1$  utiliza el contacto con  $N_2$  puede saber en  $T_2$  si  $N_2$  tiene el paquete o no. Gracias a esta característica la verificación de modelos puede encontrar un plan de enrutamiento donde dependiendo del éxito del primer contacto se envíe o no en  $T_5$ ; apuntando a minimizar el

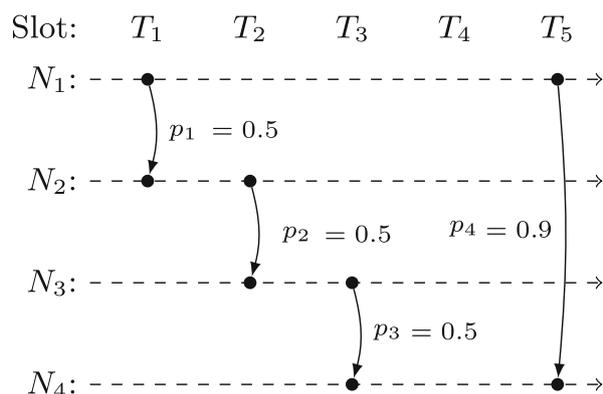


Figura 1.1: Uncertain contact plan

tiempo de entrega sin sacrificar demasiada probabilidad de éxito, pues si falla en el primer contacto aún puede utilizarse el último.

Existen entonces planes de enrutamiento que representan una cierta probabilidad de éxito, coste de tiempo y energía. Si representamos estos puntos en el espacio tri-dimensional veríamos todas las configuraciones de estos tres objetivos para el que existe un plan de enrutamiento que lo cumpla. La superficie que divide el espacio alcanzable del inalcanzable es llamado el «Frente de Pareto». Encontrar los vértices de esta superficie significa encontrar los puntos donde no existe un plan de enrutamiento que mejore algún objetivo, sin perjudicar a ninguno de los otros dos.

La verificación de objetivos múltiples sobre MDPs fue estudiada en [4], en [6] se plantearon algoritmos para encontrar la curva de Pareto con PMC. En [14] fue implementada con optimizaciones en STORM [17].

### 1.3. STORM

Este trabajo se valdrá de la herramienta STORM para llevar a cabo un análisis multi objetivo sobre MDPs que modelen planes de contacto correspondientes a DTNs espaciales. Se experimentó con topologías reales del caso de órbita lunar baja (LLO, por sus siglas en inglés), exponiendo conflictos entre SDP, la minimización del tiempo y la minimización del consumo de energía. STORM implementa análisis multi objetivo de MDPs valiéndose de algoritmos de aproximación de la curva de Pareto [6] para encontrar los puntos óptimos según los objetivos dados. No obstante STORM no cuenta con la obtención de las estrategias a aplicar en el MDP para alcanzar dichos puntos óptimos. Se proponen leves modificaciones a la herramienta STORM para obtener las estrategias de cada punto óptimo de Pareto encontrado, y así proponer planes de enrutamiento a cada nodo participante de la red.

### 1.4. Estructura de la tesis

En el siguiente capítulo se introduce la teoría necesaria para llevar a cabo los experimentos. Se definen conceptos ya mencionados y se presenta formalismos para MDP, objetivos, predicados y el análisis de un frente de Pareto. También se hace mención a la implementación propia de STORM que se tuvieron en cuenta para la implementar la extracción de estrategias.

En el tercer capítulo se habla concretamente de las decisiones de modelado tomadas. Se plantea un plan de contacto como ejemplo y se modela en PRISM. La verificación multi objetivo de este modelo simple sirve como herramienta para explicar mejor los resultados esperados en modelos más amplios. Se incluye en esta sección la explicación de como se construyen automáticamente modelos a partir de planes de contacto realistas.

El cuarto capítulo es de experimentación. Se plantea un intervalo de tiempo en un plan de contacto del Lunar Ring Road [9] y se analiza con distintas configuraciones de probabilidad, tiempo y energía. Los resultados demuestran como se pueden obtener planes de enrutamiento con altas probabilidades de éxito y disminuyendo costos de tiempo y energía. También se observa cómo incorporando datos sobre particularidades de la red se obtienen distintas intuiciones del sistema, con distintos puntos óptimos de Pareto válidos a la hora de priorizar los objetivos.

---

El quinto capítulo sirve para plantear trabajos a futuro, discutir decisiones tomadas durante el desarrollo y proponer caminos paralelos que pueden resultar de interés.

# 2

## Teoría

### 2.1. DTNs, planes de contacto y planes de contacto inciertos

Las DTNs fueron introducidas como una arquitectura estructurada alrededor de envío asíncrono de mensajes opcionalmente confiable, con pocas expectativas sobre la conectividad end-to-end y recursos de los nodos [18]. Su principal característica es hacer uso de una capa de paquete [3] con almacenamiento persistente de cada nodo. Esta capa decide a partir de un plan de enrutamiento si conservar un paquete para una oportunidad de transmisión futura, o utilizar ventanas abiertas en el momento. Las rutas en una DTN no pueden ser identificadas simplemente como una serie de pasos entre nodos, sino que serán una serie de contactos indexados por tiempo. Existen distintos tipos de DTN dependiendo de las características de la aplicación. Se puede clasificar como:

*Programadas*: cuando los contactos pueden ser previstos con exactitud. Este conocimiento se puede usar para estrategias de enrutamiento y aprovechamiento de los recursos [8].

*Probabilísticas*: los contactos son inferidos dinámicamente según la evolución de la red. El enrutamiento se basa en la probabilidad de que exista un contacto en el futuro y se utilizan múltiples copias por distintos caminos para incrementar la probabilidad de entrega. [2]

*Oportunista*: no existe ningún conocimiento de los contactos, se utilizan los contactos según exista la oportunidad.

*Programadas con incertidumbre*: fueron introducidas en [16] como DTNs donde la materialización de los contactos puede diferir del plan original según una probabilidad conocida.

Estas últimas son las que consideramos apropiadas para modelar redes espaciales en órbita baja, ya que estas sufren de fallas predecibles en sus contactos debido a causas como interferencias, conocimiento impreciso de las órbitas entre otras circunstancias propias del entorno. Así una DTN incierta, a diferencia de una DTN programada, puede funcionar en casos de comunicaciones propensas a falla; y, a diferencia de una DTN probabilista, puede aprovechar información de probabilidades dependientes al tiempo. Esto último permite el análisis de tiempo de arribo, uno de los objetivos que se plantean en este trabajo.

**Un contacto** es la oportunidad de establecer un vínculo temporal entre 2 nodos. En el caso de uso que nos concierne (constelaciones LEO) tenemos un conocimiento previo de los contactos, basado en la dinámica orbital de los satélites. Un contacto tiene un nodo emisor, un nodo receptor, un momento de apertura y uno de clausura que delimitan una ventana de comunicación entre ambos nodos. Listando todos los contactos conocidos a partir de un momento dado se traza un **plan de contacto**.

La incorporación del conocimiento inicial sobre las posibles fallas se plasma asignando una probabilidad a cada contacto y se traza un **plan de contacto incierto**. Un **plan de contacto incierto** es un plan de contacto acompañado por anotaciones de probabilidad de éxito para cada contacto.

En la figura 1.1 podemos ver un plan de contacto incierto. Nos basaremos en él para explicar los siguientes conceptos.

## 2.2. Costos de tiempo y energía

El **coste de energía esperado (EEC)** será el coste asignado a cada transmisión multiplicado por la probabilidad de que esa transmisión efectivamente ocurra según el plan de enrutamiento. Siguiendo con el ejemplo de la figura 1.1, si asignamos a cada transmisión una unidad de energía, el consumo de energía esperado haciendo los saltos los saltos  $N_1N_2N_3N_4$  será:

- 0.125 ->éxito, consume 3 unidades de energía.
- 0.125 ->fallo en el último envío, consume igualmente 3 unidades de energía.
- 0.25 ->fallo en el ante último envío, consume solo 2 unidades de energía porque no intenta el tercero.
- 0.5 ->falla en el primero, consume solo 1 unidad energía.

por lo tanto  $EEC = 0.5 + 0.25 + 0.125 + 0.125 = 1.75$ . El consumo esperado de energía es necesariamente menor o igual al consumo total de la ruta elegida.

El **tiempo de entrega esperado (EDT)** tiene una naturaleza similar, se pondera el tiempo de envío de cada camino según la probabilidad de ser tomado. Sin embargo, surge una dificultad que no existe a la hora de calcular el EEC. Debemos asignar un tiempo de entrega a caminos donde el paquete no llega a destino. Sería absurdo entender que una estrategia donde un paquete no llega a destino es más veloz a una en la que sí, o que un plan de enrutamiento que consista en no enviar el paquete tome 0 segundos. En este trabajo se considera que el tiempo de entrega para un camino no exitoso es el tiempo en el que el nodo receptor tiene su último contacto, significando que el nodo receptor puede asumir que ningún paquete llegó ni llegara. En la figura 1.1 este tiempo sería  $T_5$ , por lo que el EDT para los saltos  $N_1N_2N_3N_4$  será  $EDT = 0.125 * 3 + 0.875 * 5 = 4.75$ . Donde el SDP es 0.125 y 0.875 su complemento. El tiempo de entrega esperado es necesariamente mayor o igual al tiempo que toma la ruta elegida.

Con el concepto de tiempo de entrega esperado y consumo de energía esperado introducido, podemos entender que por más que un plan de enrutamiento tenga la intención de ser más veloz, su valoración en ese aspecto dependerá siempre de su probabilidad de éxito. Para modelar estos costes propios de una DTN en un MDP formalizaremos el concepto de MDP, estructura de costes y de objetivos.

## 2.3. PMC

La verificación de modelos probabilística (PMC, probabilistic model checking) extiende la verificación de modelos tradicional con herramientas para el análisis de fenómenos probabilísticos aproximados por aleatorización [19]. Se utiliza para verificar si un sistema probabilístico satisface ciertas propiedades y restricciones.

Un sistema probabilístico es uno cuyo comportamiento esta sujeto a aleatoriedad que puede surgir de acciones externas o procesos probabilísticos internos. Para aplicar PMC a un sistema se debe crear un modelo del sistema. Estos modelos capturan los comportamientos del modelo formalmente. Los procesos de decisión de Markov son los modelos de interés para este trabajo.

Una vez modelado el sistema se definen propiedades a partir de ciertos formalismos que permitan utilizar herramientas de PMC.

Estos son los pasos que daremos a continuación, definiendo un modelo y propiedades para finalmente analizar con STORM.

## 2.4. Proceso de Decisión de Markov

**Definición 1** Un Proceso de Decisión de Markov [10], **MDP** (Markov Decision Process) es una estructura matemática ideado para modelar sistemas no deterministas con comportamiento probabilístico. Formalmente un MDP  $\mathcal{M}$  es una tupla  $(S, Act, \mathbf{P}, l_{init})$  donde

- $S$  es un conjunto numerable de estados conteniendo  $l_{init}$ .
- $Act$  es un conjunto finito de acciones.
- $\mathbf{P} : S \times Act \times S \rightarrow [0, 1]$  es una función de probabilidad de transición tal que para toda  $s \in S$  y  $\alpha \in Act : \sum_{s' \in S} \mathbf{P}(s, \alpha, s') \in \{0, 1\}$
- $l_{init} \in S$ : es el estado inicial.

Se plantea formalmente el MDP correspondiente al plan de contacto de la figura 1.1 de la siguiente manera.

$$Nodos = \{N_1, N_2, N_3, N_4\}$$

$$\mathcal{M} = (S, Act, \mathbf{P}, (N_1, 0))$$

$$S = \{(n, t) | n \in Nodos \ \& \ 0 \leq t \leq 5\}$$

$$Act = \{N_1N_{2\_1}, N_2N_{3\_2}, N_3N_{4\_3}, N_1N_{4\_5}, next\}$$

$$l_{init} = (N_1, 0)$$

La función  $\mathbf{P}$  es la que describe las probabilidades de éxito acorde al plan de contacto. Dando una probabilidad de 1 a la acción *next* a todo estado  $(n, t)$  con  $n \in Nodos$  y  $t \in \mathbb{N}$  a su correspondiente  $(n, t + 1)$ . Una probabilidad de 0 a toda combinación invalida entre estados posibles y acciones. En cuando a las acciones que representen contactos válidos se define con la probabilidad de éxito y su complemento tal que la suma sea 1. Por ejemplo, el contacto entre  $N_1$  y  $N_2$  en  $T_1$  se define así:

$$\mathbf{P}((N_1, 0), N_1N_{2\_1}, (N_2, 1)) = 0,5$$

$$\mathbf{P}((N_1, 0), N_1N_{2\_1}, (N_1, 1)) = 1 - 0,5$$

**Definición 2** Un camino es una secuencia finita o infinita de la forma  $\pi = s_1\alpha_1s_2\alpha_2\dots s_n\alpha_n\dots$ . Un camino  $\pi$  sera válido para un MDP  $\mathcal{M}$  si  $s_1 = s_{init}$  y  $\forall \alpha_i$  tal que  $s_i\alpha_i s_{i+1}$  ocurre en  $\pi$ ,  $\mathbf{P}(s_i, \alpha_i, s_{i+1}) > 0$  con  $1 \leq i \leq n$ .

El conjunto de caminos válidos para un MDP  $\mathcal{M}$  se denota como  $\text{caminos}(\mathcal{M})$  y el último estado de un camino finito  $\pi$  se denota como  $\text{último}(\pi)$ .

Una distribución probabilística sobre un conjunto  $X$  es una función  $\mu = X \rightarrow [0, 1]$ , con soporte  $\text{sop}(\mu) = \{x \in X \mid \mu(x) > 0\}$  que cumpla  $\sum_{x \in \text{sop}(\mu)} \mu(x) = 1$ . Diremos que una distribución probabilística es *Dirac* si asigna 1 a algún elemento de  $X$ . Usaremos  $\text{Dist}(X)$  como el conjunto de distribuciones probabilísticas sobre  $X$ .

**Definición 3** Dado un MDP  $\mathcal{M}$ , una *estrategia* para  $\mathcal{M}$  sera una función  $\sigma : \text{caminos}(\mathcal{M}) \rightarrow \text{Dist}(\text{Act})$  tal que para una secuencia finita de estados de  $\mathcal{M}$  devuelve una distribución sobre las acciones posibles a continuación de  $\text{último}(\pi)$ . Una estrategia que sólo depende de  $\text{último}(\pi)$  sera llamada una estrategia sin memoria, nos referiremos a tales como *memoryless*. Una estrategia tal que para todo  $\pi \in \text{caminos}(\mathcal{M})$  devuelve una distribución *Dirac* se llamara una estrategia determinística, de no ser así se hablara de una estrategia aleatorizada. En caso de omitir esta última categorización estamos hablando del caso determinístico.

Nos referiremos a un MDP bajo una estrategia para hacer referencia a que el no determinismo es resuelto con dicha estrategia.

Cada resolución al no determinismo es representada con una estrategia responsable de elegir una acción basada en el estado del modelo y el historial de ejecución. El concepto de estrategia puede ser encontrado en la literatura como política, adversario o *scheduler*. En este trabajo mantendremos la palabra estrategia para referirnos a la función que resuelve el no determinismo de un MDP.

## 2.5. Objetivos

En este trabajo nos interesan dos tipos de propiedades sobre MDPs. Propiedades de alcance y propiedades de coste. La primera se refiere a la probabilidad de alcanzar un estado final, y la segunda el coste total esperada. Dado un MDP  $\mathcal{M} (S, \text{Act}, P, l_{\text{init}})$  definimos:

Las siguientes formalizaciones provienen de [6]

**Definición 4 Estructura de coste.** Una estructura de coste para un MDP  $\mathcal{M}$  es una función  $\rho : S \times \alpha \rightarrow \mathbb{R}$  que mapea acciones a números reales. Para un camino infinito  $\pi = s_0 a_0 s_1 a_1 \dots$  y un numero  $k \in \mathbb{N} \cup \{\infty\}$  el coste total en  $k$  pasos para  $\pi$  según  $\rho$  es  $\rho[k](\pi) \stackrel{\text{def}}{=} \sum_{i=0}^{k-1} \rho(s_i, a_i)$

**Definición 5 Propiedad de alcance.** Una propiedad de alcance  $[T]_{\sim p}^{\leq k}$ , con  $T \subseteq S$ ,  $\sim \in \{\leq, \geq\}$ ,  $p$  un umbral racional de probabilidad, y un umbral de tiempo  $k \in \mathbb{N} \cup \{\infty\}$ . Significa la probabilidad de llegar a un estado de  $T$  en  $k$  pasos que satisfaga  $\sim p$ . Formalmente, que un MDP satisfaga  $[T]_{\sim p}^{\leq k}$  bajo una estrategia  $\sigma$  se define así:

$$M, \sigma \models [T]_{\sim p}^{\leq k} \iff \text{Pr}_M^\sigma(\{s_0 a_0 s_1 a_1 \dots \in \text{caminos}(\mathcal{M}) \mid \exists i \leq k : s_i \in T\}) \sim p$$

Donde  $\text{Pr}_M^\sigma$  suma las probabilidades para un conjunto de caminos según un MDP  $\mathcal{M}$  con su no determinismo resuelto por  $\sigma$ .

**Definición 6 Propiedad de coste.** Una propiedad de coste  $[\rho]_{\sim r}^{\leq k}$  con  $\rho$  estructura de coste,  $\sim \in \{\leq, \geq\}$ ,  $r$  un umbral racional de coste, y un umbral de tiempo  $k \in \mathbb{N} \cup \{\infty\}$ . Significa el coste esperado acumulado en  $k$  pasos satisface  $\sim r$ . Formalmente, si  $M$  satisface  $[\rho]_{\sim r}^{\leq k}$  bajo la estrategia  $\sigma$  se define así:

$$M, \sigma \models [\rho]_{\sim r}^{\leq k} \iff \text{ExpTot}_M^{\sigma, k}(\rho) \sim r \text{ donde } \text{ExpTot}_M^{\sigma, k}(\rho) \stackrel{\text{def}}{=} \int_{\pi} \rho[k](\pi) d\text{Pr}_M^\sigma$$

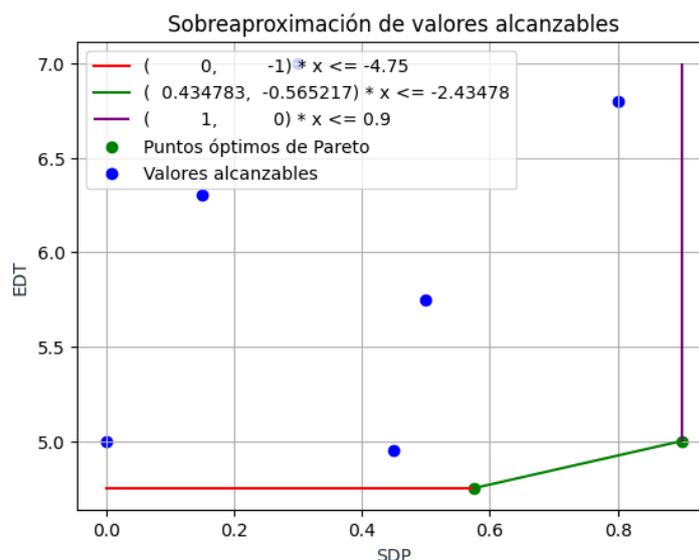


Figura 2.1: Frente de Pareto optimizando tiempo y probabilidad de éxito

El caso de ambas propiedades que utilizaremos para los MDPs de interés es con  $k = \infty$ , por lo que en la notación se puede omitir.

$$\text{donde } \text{ExpTot}_M^\sigma(\rho) \stackrel{\text{def}}{=} \lim_{k \rightarrow \infty} \text{ExpTot}_M^{\sigma,k}(\rho)$$

## 2.6. Puntos óptimos de Pareto

**Definición 7 Predicado multi objetivo.** Es un vector  $\psi = (\psi_1, \dots, \psi_n)$  con  $\psi_i$  predicados de alcance y coste. Decimos que  $\psi$  es satisfecho por MDP  $M$  bajo la estrategia  $\sigma$  cuando  $M, \sigma \models \psi \iff M, \sigma \models \psi_i$  para todo  $1 \leq i \leq n$ .

**Definición 8 Propiedad de Pareto.** Para un MDP  $\mathcal{M}$  una propiedad de Pareto toma la forma  $\text{pareto}([o_1]_{\star_1}^{\leq k_1}, \dots, [o_n]_{\star_n}^{\leq k_n})$  con  $\star_i \in \{\min, \max\}$  donde cada  $[o_i]_{\star_i}^{\leq k_i}$  con  $o_i$  siendo una estructura de coste  $\rho_i$  o un conjunto objetivo  $T_i$ . El conjunto de soluciones alcanzables es  $A = \{x \in \mathbb{R}^n \mid ([o_1]_{\sim_1 x_1}^{\leq k_1}, \dots, [o_n]_{\sim_n x_n}^{\leq k_n}) \text{ es satisfactible}\}$ . Con  $\sim_i = \geq$  si  $\star_i = \max$  y  $\sim_i = \leq$  si  $\star_i = \min$ . Decimos que para puntos  $x, y \in A$ ,  $x$  domina a  $y$  si  $x_i \sim y_i$  para todo  $i$  y  $x_j < y_j$  para algún  $j$ . Entonces:

$$\text{pareto}([o_1]_{\star_1}^{\leq k_1}, \dots, [o_n]_{\star_n}^{\leq k_n}) \stackrel{\text{def}}{=} \{x \in A \mid x \text{ no es dominado por ningún } y \in A\}$$

El conjunto  $\text{pareto}([o_1]_{\star_1}^{\leq k_1}, \dots, [o_n]_{\star_n}^{\leq k_n})$  se denomina frente de Pareto o Curva de Pareto y es una superficie convexa que delimita el espacio de soluciones alcanzables del de las soluciones inalcanzables.

**Example 2.6.1** Tomemos como ejemplo la figura 1.1 definiendo un costo sólo para el tiempo. Entonces podemos definir la estructura de costo de la siguiente manera para indicar que el coste de tiempo avanza

con toda acción tomada, excepto si el paquete ya llegó al nodo receptor  $N_4$ :

$$\rho(s, a) = \begin{cases} 0, & \text{si } s = N_4 \\ 1, & \text{en caso contrario} \end{cases}$$

La propiedad de Pareto sería:

$$\text{pareto}([\{(N_4, x) | x \in \mathbb{N}\}]_{\max}^{\leq 5}, [\rho]_{\min}^{\leq 5})$$

Vemos los resultados en la figura 2.1. Nos referiremos a los puntos óptimos de Pareto (0.575, 4.75) y (0.9, 5) como A y B respectivamente. El frente de Pareto esta representado por la intersección de tres sobre-aproximaciones distinguidas por colores. El espacio convexo delimitado por el frente representa los puntos alcanzables, ilustrados por una pequeña muestra en azul. Los puntos verdes, ubicados en los vértices del frente son los puntos óptimos de Pareto. Ambos representan estrategias simples del MDP, más precisamente: A es la estrategia que  $N_1$  prueba enviar en  $T_1$  y en caso de fallar vuelve a intentar en  $T_5$ . B es la estrategia donde  $N_1$  espera a  $T_5$  para enviar.

## 2.7. Estrategias aleatorizadas

Una estrategia simple es una estrategia sin aleatoriedad y con limitado espacio de memoria [13]. El frente de Pareto nos señala que existen estrategias para todos los puntos que conforman líneas roja, verde y purpura. Estas estrategias no son todas simples, pues existen infinitos valores en el frente y no existen infinitas estrategias simples, sino que casi todos estos valores corresponden a introducir un elemento de aleatoriedad entre dos estrategias. Es decir, si la estrategia  $\sigma_A$  corresponde al punto óptimo de Pareto A y se busca obtener una estrategia sobre el frente de Pareto hacia la izquierda (línea roja) en decremento del SDP, se puede introducir una pequeña probabilidad aleatoria de descartar el paquete en la capa de paquete del nodo receptor. Seleccionando la distribución adecuada para elegir entre estrategias podemos construir nuevas estrategias arbitrarias para cualquier punto del frente de Pareto. Luego, para construir estrategias que correspondan a puntos en la línea verde, alcanza con elegir una distribución justa para aleatorizar la elección entre la estrategia  $\sigma_A$  y  $\sigma_B$ . Finalmente las estrategias correspondientes a los puntos sobre la línea roja, corresponden simplemente a la estrategia  $\sigma_B$  modificada para esperar una cantidad justa de tiempo en el nodo receptor antes de considerar al paquete recibido. Esta última, aunque implementable en el sistema real, dependerá del modelado del sistema, el  $T_{max}$  escogido y la posibilidad de esperar una cantidad arbitraria de tiempo sin considerar al paquete como recibido. En los modelos planteados en este trabajo el tiempo máximo esta limitado al fin del plan de contacto y la acción *next* deshabilitada una vez entregado el paquete, ya que el objetivo siempre sera minimizar el tiempo. Pero a fines teóricos sirve visualizar el hecho de que si es posible entregarlo en tiempo  $t$  también es posible entregarlo en cualquier tiempo mayor a  $t$ .

Ubicados en el plano  $EDT \times SDP$  de la figura 2.1 se pueden generar estrategias  $\sigma_p$  para cualquier punto  $p$  combinando las estrategias correspondientes a los puntos óptimos y técnicas mencionadas de «esperar» y «descartar» para el movimiento vertical y horizontal. Claro esta que estos puntos son de interés puramente teórico y en la práctica las estrategias

que nos interesarán serán cercanas o sobre el frente de Pareto. Esto destaca la importancia de los puntos óptimos de Pareto, como las mejores estrategias para optimizar en una dirección, a partir de los cuales se puede formar una estrategia conveniente dentro del espacio convexo que demarcan.

Si tenemos  $q^1, q^2, \dots, q^n$  puntos sobre el frente de Pareto entonces para generar cualquier estrategia que corresponda a un punto  $r$  dentro del espacio de valores alcanzables basta con formar un vector peso  $p \in \mathbb{R}^n$  tal que  $r_i \leq \sum_{j=1}^n p_j * q_i^j$  con  $1 \leq i \leq n$ . Coloquialmente el vector peso asigna probabilidades a todos los puntos  $q^j$  de seleccionar la estrategia  $\sigma_{q^j}$ .

Las estrategias de interés para este trabajo son simples, acorde a los recursos limitados de los nodos [13] [1] y debido a que la técnica de aleatorización no es fiel al modelo descentralizado planteado. Una conjunción de estrategias aleatorizadas requiere que nodos incomunicados coordinen sus elecciones aleatorias, algo imposible si los planes de enrutamiento son distribuidos con antelación. Por eso los planes de enrutamiento producidos y de interés son solo aquellos correspondientes a los puntos óptimos de Pareto.

Las estrategias correspondientes a los puntos óptimos de Pareto que obtendremos a partir de estos modelos son solo estrategias simples o puras, es decir *memoryless* y determinísticas.

Todos los puntos óptimos corresponden a estrategias determinísticas. Esto se puede demostrar viendo en que consiste la aleatorización de estrategias. Una estrategia aleatorizada  $\sigma$  devuelve distribuciones no *Dirac*, es decir que partiendo en un camino  $\pi$  devuelve una distribución  $\mu$ , formalmente  $\sigma(\pi) = \mu$  entonces  $\exists \alpha, \beta : \mu(\alpha) > 0$  y  $\mu(\beta) > 0$  con  $\alpha \neq \beta$ . Estas distribuciones indican la probabilidad de tomar distintas acciones. Sea  $\sigma_r$  una estrategia aleatorizada donde sus decisiones son aleatorias en cada estado podemos seleccionar un estado  $s$  al que  $\sigma$  le asigna la distribución  $\delta$  con  $n$  acciones posibles y obtener  $n$  estrategias  $\sigma_r^1, \sigma_r^2, \dots, \sigma_r^n$  donde la elección en ese estado es determinística.  $\sigma_r$  es el resultado de aplicar el proceso de aleatorización entre  $\sigma_r^1, \sigma_r^2, \dots, \sigma_r^n$  según la distribución  $\delta$ . Si seleccionamos otro estado  $s'$  con  $m$  acciones aleatorizadas para cada estrategia  $\sigma_r^i$  para  $1 \leq i \leq n$  según una distribución  $\delta'_i$ . Obtendremos  $n * m$  estrategias  $\sigma_r^{1,1}, \dots, \sigma_r^{1,m}, \dots, \sigma_r^{n,m}$  aleatorizadas según la distribución que asigna a  $\sigma_r^{i,j}$  la probabilidad  $\delta(\sigma_r^i) * \delta'_i(\sigma_r^{i,j})$  para  $1 \leq i \leq n$  y  $1 \leq j \leq m$ . Donde al menos  $s$  y  $s'$  son determinísticas. Recordemos que para toda estrategia  $\sigma_r^i$   $m$  será el mismo ya que parten de una estrategia original solo modificada para  $s$ . Iterando por sobre el número de estados podemos obtener una lista de estrategias determinísticas y una distribución tal que obtenemos la estrategia aleatorizada  $\sigma_r$ .

Veamos en el caso de dos objetivos, como en la figura 2.1. La aleatorización de estrategias correspondiente a dos puntos resulta en una estrategia con un valor acorde a un punto de la recta que los une. Es decir, si aleatorizamos  $\sigma_1$  correspondiente a  $(sdp_1, edt_1)$  con  $\sigma_2$  correspondiente a  $(sdp_2, edt_2)$  según la distribución que asigna  $\delta(\sigma_1)$  la probabilidad de elegir la estrategia  $\sigma_1$ . Obtendremos que el punto correspondiente a  $\sigma_a$  sera  $(sdp_a, edt_a)$  con  $sdp_a = \delta(\sigma_1) * sdp_1 + (1 - \delta(\sigma_1)) * sdp_2$  y  $edt_a = \delta(\sigma_1) * edt_1 + (1 - \delta(\sigma_1)) * edt_2$ .

Decir que  $(sdp_a, edt_a)$  se encuentra en la recta única que une los puntos  $(sdp_1, edt_1)$  y  $(sdp_2, edt_2)$  es decir que la siguiente ecuación es verdadera:

$$edt_a = \Delta(sdp_a - sdp_1) + edt_1$$

Donde  $\Delta = \frac{edt_2 - edt_1}{sdp_2 - sdp_1}$  correspondiente a la pendiente de la recta.

$$\begin{aligned}
edt_a - \Delta sdp_a &= -\Delta sdp_1 + edt_1 \\
(\delta(\sigma_1) * edt_1 + (1 - \delta(\sigma_1)) * edt_2) - \Delta(\delta(\sigma_1) * sdp_1 + (1 - \delta(\sigma_1)) * sdp_2) &= -\Delta * sdp_1 + edt_1 \\
\delta(\sigma_1)(edt_1 - edt_2 - \Delta(sdp_2 - sdp_1)) &= edt_1 - edt_2 - \Delta(sdp_2 - sdp_1) \\
\delta(\sigma_1)(edt_1 - edt_2 + edt_2 - edt_1) &= edt_1 - edt_2 + edt_2 - edt_1 \\
0 &= 0
\end{aligned}$$

Por lo tanto la estrategia resultante a la aleatorización no puede salir de los límites trazados por las estrategias determinísticas cayendo siempre dentro o en el perímetro del poliedro convexo formado por el frente de Pareto.

En el caso tridimensional el resultado es similar, solo que el movimiento es sobre la recta que une ambos puntos en el espacio de tres dimensiones.

## 2.8. Estrategias con memoria

Las estrategias con memoria, aunque no imposibles, no son necesarias en los procesos de decisión de Markov que estamos estudiando. Por la naturaleza distribuida del sistema a modelar, plantear una estrategia dependiente de los pasos dados previamente significaría cierta coordinación entre las partes. En el siguiente capítulo cuando se discuten los puntos de vista veremos que al modelar un MDP desde el punto de vista del paquete no es descabellado asumir conocimiento de los pasos previos, solo que en un protocolo implementado de esta manera conllevaría acarrear metadata que puede no valer la pena. De todas formas este trabajo y sus predecesores utilizan estrategias *memoryless*.

La diferencia entre estrategias con y sin memoria puede impactar en el análisis multi objetivo de MDPs. En un momento justificaremos por qué en nuestros modelos no es el caso. Para obtener un escenario donde las estrategias con memorias son necesarias para el análisis multi objetivo plantearemos un MDP modelando una DTN con ciertas diferencias y objetivos múltiples disjuntos. El diagrama de la figura 2.2 es un MDP donde a diferencia del ejemplo dado previamente, el tiempo no es parte del espacio de estados y se considera que no hay ACK. Por lo tanto cada estado corresponde a un nodo  $S = Nodes$  excepto un estado que es el estado de «paquete perdido». Por el bien del ejemplo simplificamos los contactos que están siempre disponibles, situación posible si es un tiempo acotado y las ventanas de comunicación coinciden. Supongamos que queremos hacer un análisis multi objetivo de un envío de paquetes con dos tipos de destinatarios posibles. Los *labels delivery*, *delivery2* hacen referencia a los nodos finales para cada objetivo. Podemos ver que una estrategia simple que consista en enviar a  $N_1$  y luego al  $N_2$  garantiza un valor (0.7, 0.7) en el análisis de ambos objetivos. Por otro lado existe la posibilidad de enviar a  $N_3$  y a  $N_4$  obteniendo probabilidades de (0, 1).

Ahora si permitimos estrategias con memoria podemos tomar provecho de los ciclos y  $N_3$  puede actuar distinto la segunda vez que recibe el paquete. Entonces si  $N_0$  envía el paquete a  $N_3$ , este lo envía a  $N_5$  y  $N_5$  lo devuelve a  $N_3$  con una probabilidad de 0.8.  $N_3$  con memoria puede enviarlo en vez a  $N_4$ . Esto nos da un valor de (1, 0.8) en el análisis multi objetivo,

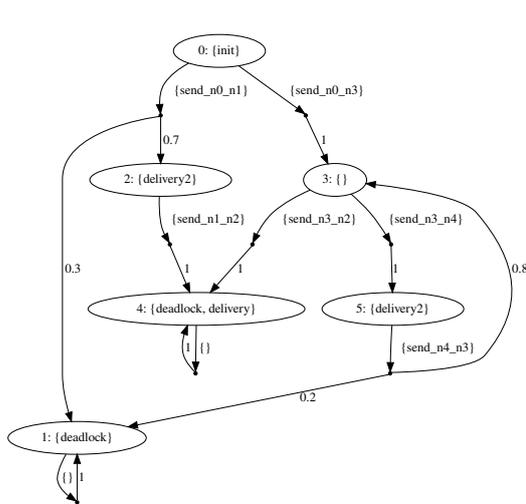


Figura 2.2: MDP representando estados como nodos y acciones como contactos

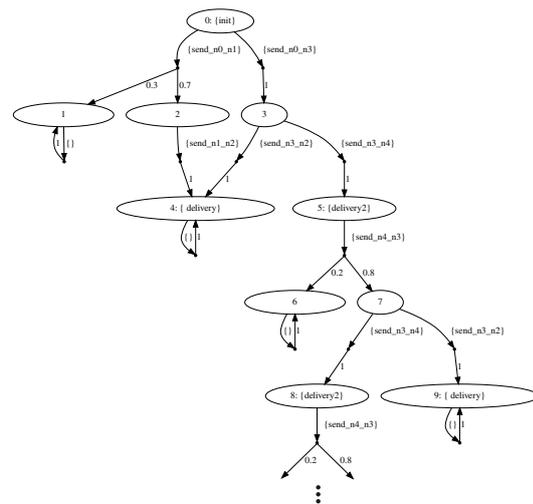


Figura 2.3: El tiempo incorporado en el espacio de estados

superador al  $(0.7, 0.7)$  de las estrategias simples. Este ejemplo es una adaptación del ejemplo dado en [13] al modelado de enrutamiento sobre DTNs inciertas.

En cambio cuando incluimos el tiempo en el espacio de estados, como en la figura 2.3 y en todos los modelos utilizados para este trabajo, obtenemos un diagrama en forma de árbol. Un diagrama sin ciclos, que significarían volver en el tiempo, sino en forma de cascada. Ahora los estados no corresponden a un nodo de la red, sino a un nodo en relación en un momento específico. Si hacemos el mismo análisis que con la figura 2.2, una estrategia simple basta para obtener el valor  $(1, 0.8)$ .

Debido a que toda la información necesaria esta contenida en los estados es que la verificación de modelos solo obtendrá valores correspondientes a estrategias *memoryless*. Por supuesto, incluir el tiempo en el espacio de estados trae consigo requerimientos de espacio de memoria para manipular el modelo.

## 2.9. PCAA en STORM

STORM implementa el algoritmo de verificación de propiedades multi objetivo como se plantean en [6] y un algoritmo distinto para el análisis de sólo estrategias puras [13]. Este último es demostrado un problema NP-Completo mientras que el primero es polinomial si el número de objetivos está fijado [13]. Debido al tamaño de los modelos utilizados por este trabajo para sus experimentos, el análisis de estrategias simples requeriría tiempos de análisis excesivamente prolongados. Por lo tanto, gracias a lo visto anteriormente, para el análisis se utilizo el algoritmo PCAA del primer enfoque, ya que los puntos óptimos corresponden a estrategias puras de todos modos.

Esta implementación aplica varias modificaciones sobre el modelo en una etapa de pre proceso, obteniendo estrategias incompatibles al modelo original aunque con los valores correctos a los puntos óptimos obtenidos. Para producir las tablas de enrutamiento en este trabajo se incluyo una contribución a la herramienta STORM. La contribución consiste en convertir las estrategias del modelo pre-procesado a estrategias válidas del modelo objetivo,

y que, por supuesto, satisfaga a los mismos valores para el predicado multi objetivo.<sup>1</sup>

Para explicar los conceptos necesarios para la extracción de estrategias este trabajo se limitará a las optimizaciones realizadas por [15]. Los detalles del algoritmo que efectúa la verificación multi objetivo escapan a lo que queremos abordar en esta tesis.

De interés a nuestra contribución son tres optimizaciones que realiza STORM en la etapa de preproceso del modelo: la eliminación de estados irrelevantes, la unión de estados equivalentes y la incorporación de memoria de objetivos en la representación de estados.

Se considera un estado irrelevante a aquellos estados que se pueda llegar solo desde estados donde el objetivo ya ha sido cumplido. En los modelos planteados serían esos estados donde el paquete ya arribo a destino. Esto significa estados  $s$  tales que  $\forall s' \in S : P(s', \alpha, s) > 0 \Rightarrow s'$  es un estado final.

Los estados equivalentes son absorbidos a un solo estado. En el caso de nuestro modelo estos serían estados donde el paquete ya está en el nodo receptor,  $(N_4, x)$  con  $x > 0$  para el MDP de la figura 1.1. La diferenciación de acciones a tomar una vez en estos estados son irrelevantes ya que el paquete esta en destino. Para que la estrategia sea equivalente al modelo original se necesitó hacer un mapeo del estado unificado a los estados que representa.

Por último, otra estrategia de optimización consiste en incluir dentro de la matriz representante de estados una estructura de memoria que indica si un objetivo ha sido obtenido. Esto sirve para escenarios donde hay varios objetivos de alcance. Sería necesario si se desea hacer un análisis multi objetivo de enrutamiento con múltiples nodos receptores como el ejemplo dado en la figura 2.2. Esta optimización multiplica la matriz por cuantos espacios de memoria sean necesarios. En este trabajo nos limitamos a un solo nodo receptor. por lo tanto un solo predicado de alcanzabilidad basta para el conjunto de estados donde el nodo receptor recibió el paquete.

El análisis multi objetivo de múltiples receptores es una posibilidad discutida en el capítulo 5, aunque no abordada en el trabajo. Por lo tanto, al haber un solo predicado de alcanzabilidad, la cantidad de estados se vuelve el doble. Para cada estado original corresponde un estado donde el objetivo ha sido cumplido y uno donde no. Esto facilita el análisis multi objetivo siendo que solo es necesario tomar en cuenta objetivos de coste. Para invertir este producto de memoria y estados hizo falta buscar en las estrategias finales según la memoria y el estado asignado al estado del modelo original correspondiente.

Al almacenar las representaciones de las estrategias en memoria durante el proceso de análisis multi objetivo crece significativamente el consumo de memoria. El tamaño requerido por las estrategias es lineal con respecto al tamaño de los modelos y el consumo total crece linealmente en relación a la cantidad de puntos óptimos de Pareto hallados.

## 2.10. Conclusión del capítulo

Con este trasfondo teórico sobre el análisis multi objetivo de MDPs, el uso de las mismas para enrutamiento en DTNs y los conceptos introducidos para su formalismo continuamos con las decisiones tomadas a la hora del modelado en PRISM. El trabajo busca plantear un análisis multi objetivo como herramienta para aprovechar los recursos de tiempo y energía

<sup>1</sup>Esta implementación no es oficial de STORM, sino que solo se utilizó para los modelos planteados durante este trabajo. Su funcionamiento en otros modelos esta aún siendo evaluada y pendiente a mejoras como trabajo a futuro.

en el enrutamiento de un único paquete de un nodo emisor a un nodo receptor. Conceptos que fueron mencionados en esta sección como receptores múltiples, congestión y estrategias aleatorizadas serán retomados solo en la sección de debate.

A partir de estos conceptos se plantea el siguiente predicado multi objetivo para obtener el frente de Pareto respectivo al MDP correspondiente al plan de contacto presentado previamente. En primer lugar un objetivo de alcance que indica el nodo receptor, seguido por dos objetivos de coste con sus estructuras correspondientes para el tiempo y la energía.

$$\rho_1(s, \alpha) = 1$$

$$\rho_2(s, \alpha) = \begin{cases} 0, & \text{si } \alpha = next \\ 1, & \text{en caso contrario} \end{cases}$$

$$pareto([\{(N_4, x) | x > 0\}]_{max}, [\rho_1]_{min}, [\rho_2]_{min})$$

Las estructuras de coste asignan una unidad a todo contacto, y se diferencian en su trato de la acción *next* siendo que cuesta tiempo pero no cuesta energía. Estas estructuras pueden ser moldeadas para representar información real de una red satelital. En el siguiente capítulo se presenta el modelado en PRISM de estos mismos conceptos para poder ser verificado por STORM.

# 3

## Modelos

### 3.1. Puntos de vista

El desafío de llevar a procesos de decisión de Markov el enrutamiento en una DTN presenta varias alternativas. Dos fueron analizadas en el proceso de este trabajo. La primera consiste en modelar desde el punto de vista del paquete, es decir que es el paquete quien toma la decisión del próximo salto y acumula la información de los saltos anteriores. Mientras que la segunda consiste en un enfoque más concreto donde se busca modelar a los nodos tomando las decisiones de a donde enviar el paquete de acuerdo a un reloj interno sincronizado con sus pares, formándose la red como la composición paralela de los nodos con el reloj sincronizado.

El primer enfoque parte de basarse en el modelo planteado en [10]. Su fortaleza viene de su simpleza, tanto de escritura como del diagrama final obtenido. Es también una ventaja el hecho que provenga de la literatura sobre la que basamos este trabajo.

El segundo enfoque termina resultando complejo a la escritura y a la hora del análisis del diagrama. Sus ventajas pueden ser en vista a trabajos futuros donde se quiera estudiar la potencial congestión de red u otros intereses propios del estado particular de los nodos. Contemplar la existencia de ACKs implica otro nivel de sincronismo que complejiza aún más el modelo. Por lo tanto, para los fines de este trabajo, no se consideró adecuado.

Por lo que se tomo la decisión de solo utilizar el primer enfoque desde el punto de vista del paquete para el resto del trabajo.

Como lenguaje de modelado se utilizó PRISM, al ser un lenguaje soportado por STORM que a su vez soporta el modelado de MDPs. A continuación plantearemos conceptos importantes que impactaran en los experimentos del siguiente capítulo.

### 3.2. Idea

Con el objetivo de plantear las pautas de diseño para este tipo de sistema nos enfocaremos en el plan de contacto incierto de la figura 1.1. Viendo los resultados que propone para este MDP podremos comprender más fácilmente los resultados de modelos de mayor tamaño que representen planes de contacto realistas.

En este plan de contacto existe una ruta de tres saltos en tres unidades de tiempo, pero con probabilidad de éxito baja. Existe otra alternativa de tomar un único salto con altas probabilidades de éxito, pero que ocurre luego de cinco unidades de tiempo.

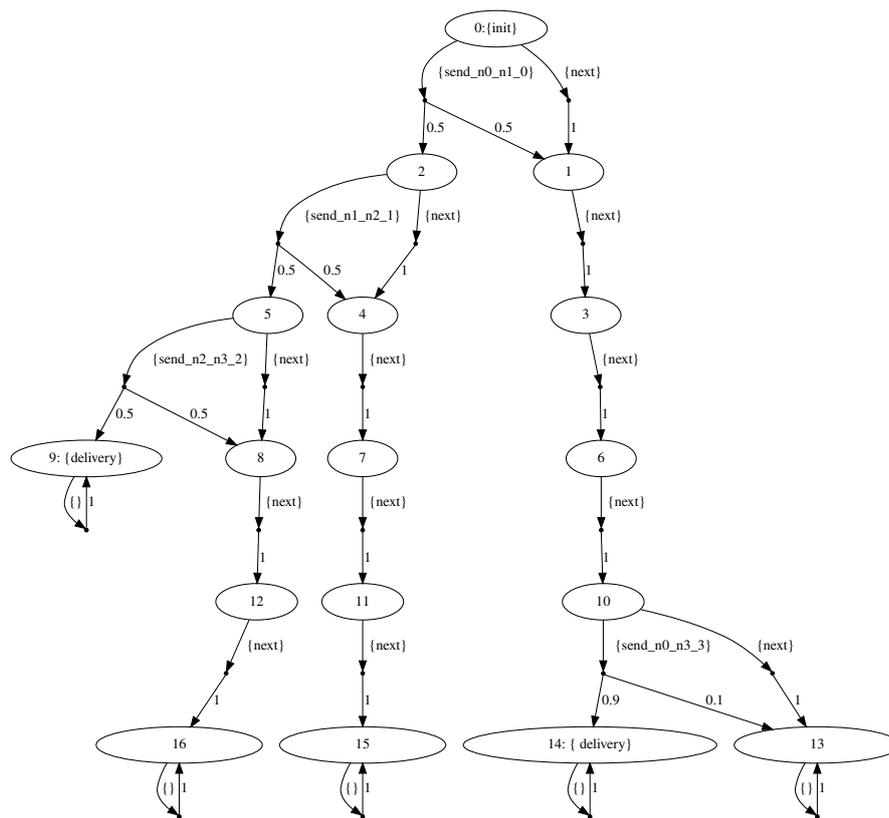


Figura 3.1: MDP para el ejemplo 1

El diagrama de la figura 3.1 nos facilita la visualización de la toma de decisiones que tendrá un paquete para llegar a destino. Los estados donde el paquete está en  $N_4$  están acompañados de la etiqueta (*delivery*). Solo existen dos tales estados (9 y 14), son aquellos en donde se utilizaron los únicos dos contactos disponibles que tiene  $N_4$  y son exitosos, uno con probabilidad de 0.5 y otro de 0.9.

Podemos leer el diagrama como una cascada escalonada, donde cada escalón representa una unidad de tiempo. Existen seis escalones para el plan de contacto de 5 unidades de tiempo. El último cuenta con los 4 estados finales posibles, tres que representan que el paquete queda en un nodo sin contactos futuros, y uno que representa el arribo en el último espacio de tiempo.

Cabe destacar una decisión de diseño desde el diagrama. Si vemos el estado 9, que representa el arribo del paquete en  $T_3$  desde  $N_3$  representado como  $(N_4, 3)$  en el MDP  $\mathcal{M}$ , podría continuar con los subsiguientes estados  $(N_4, 3 + n)$  para  $n \in \{1, 2, 3\}$ . La decisión de omitir estos estados irrelevantes conlleva deshabilitar la acción *next* una vez arribado el paquete.

El objetivo es modelar con PRISM este MDP y plantear una metodología de modelado para utilizar sobre distintos planes de contacto. A continuación se establecen las pautas de cómo representaremos estados, acciones y recompensas para lograrlo.

### 3.3. Implementación

Se modeló el paquete con el siguiente estado: nodos visitados, nodo actual y el tiempo transcurrido.

Pequeños detalles en el modelo cambian totalmente los resultados. Previamente se menciono que en este trabajo se asume la existencia de ACKs, por lo tanto en la actualización del estado de cada acción la variable *node* siempre tendrá un valor de un nodo válido. Por lo tanto, no pueden existir estados donde el paquete se considere perdido. Si el contacto falla, el paquete continúa en el nodo del lado emisor del contacto. Si el contacto es exitoso, el paquete pasa a estar en el nodo del lado receptor del contacto. Un paquete solo se considerará perdido cuando esté varado en un nodo sin contactos a futuro, como por ejemplo el estado 8 en la figura 3.1.

El motivo por el cual no podemos permitir duplicados del paquete se debe a que la verificación de modelos trabaja con información global del sistema, algo inaplicable en un sistema distribuido. De no existir esta restricción el análisis multi objetivo obtendría resultados insignificantes, pues estarían basados en estrategias imposibles de implementar. Si existieran estados donde el paquete puede tomar dos caminos paralelos en simultaneo y se encuentre en más de un nodo a la vez, la verificación de modelos obtendrá resultados donde un nodo toma decisiones a partir del conocimiento de todo el sistema. Por ejemplo, si el paquete fuera a tomar caminos en paralelo y fuese entregado exitosamente en uno entonces PMC utilizaría información de todo el sistema y decidiría abortar el camino paralelo consumiendo así menos energía. En el sistema real los nodos desconocen esta información, y si la estrategia utiliza copias los caminos se deberían ejecutar en completitud. Es por eso que PMC no corresponde al sistema de nodos distribuido que queremos modelar.

Se presupone en principio que cualquier estrategia que envíe el paquete de un nodo  $N_x$  a un nodo  $N_y$  en algún momento, nunca necesitará enviarlo del nodo  $N_y$  al nodo  $N_x$ , ya que esta acción consumiría tiempo y energía acarreando un riesgo de fallo.<sup>1</sup> Si aceptamos esta suposición podemos remover las acciones que signifiquen un ciclo del MDP y simplificar el modelo final. Para lograr este comportamiento se incluyen variables en el módulo paquete que representan si un nodo ya fue visitado. Se incluye en las guardas de las acciones que representan contactos la pre-condición de que el nodo no haya sido visitado previamente.

Las estructuras de costos se escriben utilizando el concepto de *rewards* de PRISM donde la guarda refiere al estado inicial de la transición. PRISM permite establecer *rewards* a estados y acciones. Se considero más apropiado modelar estos costes vinculándolos a las acciones en lugar de los estados con el objetivo de permitir mayor flexibilidad al diferenciar contactos. Esto nos permite establecer costes específicos a cada contacto de tiempo y energía.

El modelo planteado para el plan de contacto de la figura 1.1 es el siguiente:

```
1 mdp
2
3 const int NUM_OF_NODES = 4;
4 const int NUM_OF_TS = 5;
5
6 const int DELIVERY_NODE=3;
7
8 module bundle
```

<sup>1</sup>Esta suposición no es necesariamente verdadera para todo sistema y su uso esta abierto a discusión en el capítulo 5.

```

9     node: [0..NUM_OF_NODES - 1] init 0;
10    ts: [0..NUM_OF_TS] init 0;
11
12    n0: bool init false;
13    n1: bool init false;
14    n2: bool init false;
15    n3: bool init false;
16
17    // [action] guard -> error_prob : update + send_prob : update;
18    [send_n0_n1_0] ts=0 & node=0 & !n1 & node!=DELIVERY_NODE -> 0.500000: (
19      ts'=ts+1) + 1 - 0.500000: (ts'=ts+1) & (node'=1) & (n0'=true);
20    [send_n1_n2_1] ts=1 & node=1 & !n2 & node!=DELIVERY_NODE -> 0.500000: (
21      ts'=ts+1) + 1 - 0.500000: (ts'=ts+1) & (node'=2) & (n1'=true);
22    [send_n2_n3_2] ts=2 & node=2 & !n3 & node!=DELIVERY_NODE -> 0.500000: (
23      ts'=ts+1) + 1 - 0.500000: (ts'=ts+1) & (node'=3) & (n2'=true);
24    [send_n0_n3_3] ts=4 & node=0 & !n3 & node!=DELIVERY_NODE -> 0.100000: (
25      ts'=ts+1) + 1 - 0.100000: (ts'=ts+1) & (node'=3) & (n0'=true);
26    [next] ts < NUM_OF_TS & node!=DELIVERY_NODE -> (ts'= ts + 1) ;
27 endmodule
28
29 rewards "energy"
30   [send_n0_n1_0] node!=DELIVERY_NODE: 1;
31   [send_n1_n2_1] node!=DELIVERY_NODE: 1;
32   [send_n2_n3_2] node!=DELIVERY_NODE: 1;
33   [send_n0_n3_3] node!=DELIVERY_NODE: 1;
34 endrewards
35
36 rewards "time"
37   [send_n0_n1_0] node!=DELIVERY_NODE: 1;
38   [send_n1_n2_1] node!=DELIVERY_NODE: 1;
39   [send_n2_n3_2] node!=DELIVERY_NODE: 1;
40   [send_n0_n3_3] node!=DELIVERY_NODE: 1;
41   [next] node!=DELIVERY_NODE: 1;
42 endrewards
43
44 label "delivery" = node=DELIVERY_NODE;

```

El módulo modelado es un paquete, *bundle*, y sus acciones son todos los contactos según el plan de contacto descrito en 1.1 acompañados de la acción *next* que solo incrementa el reloj. Las acciones tienen las guardas circunstanciales más la guarda de la «no previa» visita al nodo. Se incluye una guarda *node!=DELIVERY\_NODE* para evitar acciones y consumos posteriores a la entrega del paquete. El objetivo final del modelo siempre es entregar el paquete, por ende cualquier efecto posterior es secundario.

Este modelo servirá de estándar para armar automáticamente MDPs a partir de planes de contacto. Las pautas y decisiones tomadas se mantienen en todos los modelos presentados en este trabajo.

### 3.4. Resultados

Una vez modelado en PRISM podemos hacer consultas multi objetivo a STORM, como la siguiente

```
multi(Pmax =? [ F "delivery" ], R{"time"}min =? [ C ], R{"energy"}min =? [C])
```

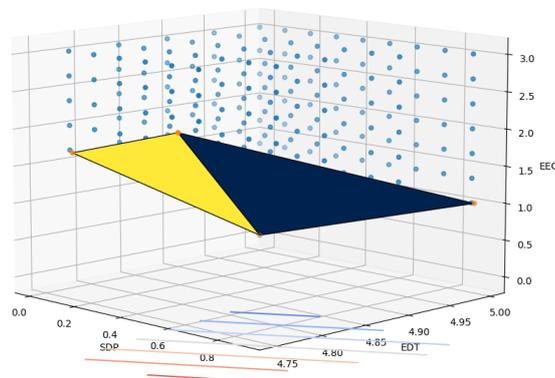


Figura 3.2

Se plantea un predicado multi objetivo donde el primer objetivo es obtener la probabilidad máxima de que se cumpla la formula «*F "delivery"*» que se puede leer como «eventualmente se entrega» sin importar que ocurre antes. Los otros dos objetivos se definen como la minimización de ambos *rewards* con la formula «*C*» haciendo referencia al coste acumulado total esperado.

A lo que STORM retorna cuatro puntos óptimos de Pareto:

4 Pareto optimal points found:

(	0.575,	4.75,	2.25 )
(	0.125,	4.75,	1.75 )
(	0.9,	5,	1 )
(	0,	5,	0 )

Indicando las 4 políticas posibles que podemos tomar:

1. Utilizar  $N_1 \rightarrow N_2$  y en caso de error utilizar el ultimo contacto en  $T_5$ .
2. Utilizar  $N_1 \rightarrow N_2$  y no utilizar el contacto en  $T_5$ , con menor gasto de energía.
3. Ignorar el primer contacto y fiarse del ultimo contacto,  $N_1 \rightarrow N_4$ .
4. No utilizar ningún contacto y gastar 0 energía.

Esto evidencia la simplicidad del ejemplo, pero da buenos indicios a tener en cuenta cuando analizamos los resultados de la aplicación real.

El frente de Pareto dibujado en dos dimensiones no presenta demasiadas dificultades a la interpretación. Lo que el analista busca es diferenciar el espacio de puntos alcanzables de los inalcanzables. Esta tarea se complejiza significativamente en el análisis en tres dimensiones. Para mejor comprensión de los diagramas presentados en este trabajo se proyecta una dispersión uniforme de puntos dentro del espacio alcanzable. En colores, las superficies triangulares formadas entre los puntos óptimos están representando la sección del plano

resultante de aleatorizar las estrategias correspondientes a los puntos óptimos. La superficie total representa un segmento del frente de Pareto donde solo existen aleatorizaciones de puntos óptimos, es decir, donde yacen las estrategias de mayor interés. Resaltados en los vértices del frente, se indican los puntos óptimos hallados. Adicionalmente, para facilitar la perspectiva, en el plano  $EDT = 0$  se dibujan las curvas de contorno de la superficie. Como es el caso de la figura 3.2, el frente de Pareto corresponde a un plano que divide el espacio diagonalmente. La ambigüedad de la representación tridimensional es aclarada con las curvas de contorno rectas que nos indican que estamos hablando de un plano.

### 3.5. Tablas de enrutamiento

El producto final de interés para el analista son las tablas de enrutamiento. Estas indicarán a cada nodo participe si debe conservar el paquete o utilizar un contacto, de acuerdo al TS. Gracias a la contribución de este trabajo a la herramienta STORM, podemos obtener las estrategias que solucionan el no determinismo para cada punto óptimo de Pareto propuesto. A partir de una simple transformación de las estrategias producidas por STORM se proponen tablas de enrutamiento como la siguiente:

node 0	
time stamp	send to
$T_1$	$N_1$
$T_5$	$N_4$

node 1	
time stamp	send to
$T_2$	$N_2$

node 2	
time stamp	send to
$T_3$	$N_3$

Esta tabla corresponde al ejemplo de 1.1, al punto óptimo de Pareto (1.) de la sección anterior. Solo es necesario indicar los momentos en donde un contacto es utilizado y a que nodo se debe enviar. Todos los TS que no figuran en la tabla de enrutamiento indican que en caso de tener un paquete para enviar se debe conservar para utilizar un contacto futuro.

### 3.6. Topología LEO

Siguiendo las pautas marcadas previamente se pueden modelar planes de contactos como MDPs. En este trabajo se utilizó información tomada por [9] describiendo la red «Lunar Ring Road», y en esta sección abordaremos la metodología utilizada. La información de planes de contacto de esta topología fue plasmada al lenguaje de modelado PRISM de manera automatizada.

El modelo final cuenta con un intervalo de tiempo que inicia en el momento que se quiere enviar el paquete, y termina cuando se considera que ya ha transcurrido un tiempo excesivo (*timeout*). Se diferencia el nodo fuente y el nodo objetivo como se vio en la sección anterior.

En el caso de estudio existen diversos tipos de nodos: sensores, satélites, bases lunares y bases terrestres. Este trabajo hace énfasis en estas distintas categorías para estudiar los distintos objetivos. Por ende, cada acción irá acompañada de una etiqueta que indique que categoría de nodos se están comunicando, sus identificadores y el intervalo del contacto. La acción de guardar el paquete y seguir esperando fue etiquetada como *next*.

En cuanto a los costes, la energía es modelada identificando los contactos por etiquetas de comunicación. La etiqueta *next* consume 0 de energía. Se consideran las categorías de nodos para establecer un índice de consumo de energía por contacto. El valor no es indicativo de ninguna unidad de medida energética, sino que relativo al concepto de «unidad de energía». El mismo criterio aplica para el tiempo en el modelo de ejemplo. En cuanto a los experimentos se utilizan milisegundos como unidad de tiempo.

El tiempo fue modelado de dos formas, considerando o no la latencia del contacto. En caso de asumir conocimiento inmediato del éxito o fracaso del contacto, el modelo pasa al siguiente estado consumiendo siempre la misma cantidad de tiempo. Por consiguiente, podríamos tomar la cantidad de estados visitados como medición del tiempo. No obstante, la latencia de transmisión entre un satélite a una base lunar no es equivalente a la latencia de transmisión entre una base lunar y una base terrestre. Valores como este serán interesantes a la hora de generar nuevos escenarios. En este trabajo se realizaron estudios de ambas formas, vinculando el tiempo a estados o a acciones. La primera manteniendo la simplicidad y la segunda modelando el tiempo como un costo acumulativo que varía según los contactos similarmente a la energía. Este último fue el que se utilizó en los experimentos para poder producir variaciones en las latencias.

A diferencia el plan de contacto simplificado que vimos en 1.1, en una red espacial los contactos pertenecen a un intervalo, no a un instante. Por lo tanto la guarda de las acciones pasa de corresponder a  $TS = i$  con  $i$  instante, a ser  $TS < end \ \& \ TS > start$  con  $start$  y  $end$  instantes de inicio y fin del intervalo, respectivamente. Este intervalo puede no coincidir con el fin de de la sección del plan de contacto seleccionada. Si un contacto expira en un momento posterior al *timeout* del modelo se trunca el contacto hasta ese momento. De no hacerlo, el modelo se extendería hasta que expire el ultimo contacto, y al no tener la acción *next* habilitada se vería obligado a tomar las acciones correspondientes al contacto vigente. Esta problemática, de no ser aplacada, provoca que las estrategias tengan un consumo de energía y tiempo de entrega mayor al real.

```
1 mdp
2
3 const int NUM_OF_NODES = 25;
4 const int NUM_OF_TS = 100000;
5
6 module bundle
7     node: [0..NUM_OF_NODES] init 23;
8     ts: [90000..NUM_OF_TS] init 90000;
9
10     n1: bool init false;
11     ...
12     n25: bool init false;
13
```

```
14 [send_6_1_54505] ts<100000 & ts>54505.682 & node=6 & !n1 -> 0.1: (ts'=ts
    +1000) + 1 - 0.1: (ts'=ts+1000) & (node'=1) & (n6'=true);
15 ...
16 [next] ts < NUM_OF_TS & node!=1 -> (ts'= ts + 1);
17 endmodule
18
19 rewards "energy"
20 [send_6_1_54505] & node!=1 : 1;
21 ...
22 [next] true : 0;
23 endrewards
24
25 rewards "time"
26 [send_6_1_54505] node!=1 : 1000;
27 ...
28 [next] node!=1 : 1;
29 endrewards
30
31 label "delivery" = node=1;
```

### 3.7. Conclusión de capítulo

La compresión y validación de estos modelos es necesaria para la integridad de los experimentos detallados a continuación. En este capítulo se plantea una estructura generalizada de MDP para modelar el envío de paquetes a través de una DTN espacial. La intención de flexibilidad en el modelado de *rewards* sera clave para estudiar escenarios y plantear la utilidad del análisis multi objetivo.

# 4

## Experimentación

### 4.1. Metodología

PMC no es ideal para enrutamiento en DTNs espaciales, por motivos ya mencionados y citados de [12]. El coste en tiempo y recursos es demasiado para ser utilizado como algoritmo de enrutamiento en tiempo real, y se basa en estrategias de información global. En este trabajo, a modo de referencia al desarrollo adecuado para este caso de uso, se tomarán planes de contacto reducidos a un intervalo de tiempo arbitrario. El planteo es enviar un paquete desde un nodo a otro topológicamente lejano. El intervalo de tiempo es el suficiente para que el paquete arribe, dando lugar a varias rutas posibles. Las especificaciones serán discutidas con detalle en la próxima sub-sección. Basado en el escenario planteado, sin información de los detalles de la red, se analiza la red en los 3 objetivos discutidos. Todos los nodos son tomados como pares, por lo que todos contactos tienen la misma latencia, coste energético y probabilidad de éxito. A continuación se incorpora información razonable según la topología de la red. En una primera instancia se varían las probabilidades de éxito de los contactos según los tipos de nodos involucrados. En segunda instancia los costes energéticos visibilizan la diferencia entre contactos caros energéticamente y aquellos más simples. En tercera instancia las latencias de los contactos es variada de acuerdo a las distancias aproximadas de los nodos. Por último, se estudia un caso particular donde un tipo de nodo es considerado particularmente ineficaz y contactos que lo involucren sufren grandes probabilidades de falla.

### 4.2. Lunar Ring Road

La topología en cuestión es la de «Lunar Ring Road» (LRR) [9]. Donde una infraestructura basada en «Ring Road» [5] es planteada para el ambiente lunar. Los satélites LEO se utilizan para acarrear paquetes planteando una DTN. Por ende se considera la LRR como una topología de interés para el análisis de este trabajo.

La idea planteada consiste de constelaciones de satélites en la órbita lunar baja (LLO) que colectan información de ubicaciones desprendidas, en particular aquellas sin energía suficiente para llegar a la tierra, o en la cara oculta de la luna. En la figura 4.1 se plantea un escenario donde un sensor en la cara oculta de la luna envía información a una estación terrestre. La información espera una oportunidad de comunicación con un satélite, que

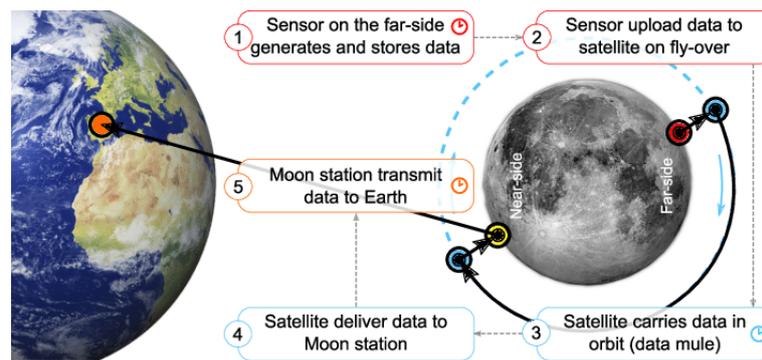


Figura 4.1: Visualización de LRR tomada de [9]

acarrea la información hasta una estación lunar que será la encargada de transmitirlo a la estación terrestre.

La ubicación de los nodos resulta crucial para su energía. Mientras algunas regiones solares con buena elevación pueden tener 90 % de disponibilidad solar durante el año, otras regiones no polares deben recurrir a fuentes de energía alternativas y constar de baterías de alta capacidad para sostenerse [9]. Es un escenario donde la energía es un recurso muy valioso y de gran peso a la hora de elegir un plan de enrutamiento. Con estas categorías de nodos se pueden plantear diferencias razonables de coste energético y latencia. Por un lado, los satélites y sensores suelen ser menos robustos que las estaciones, por ende contactos que involucren estos nodos serán menos eficaces. Por el otro, tanto el coste energético como la latencia es mayor en contactos entre estaciones terrestres y estaciones lunares. En esta topología no existen contactos directos de satélites o sensores lunares a estaciones terrestres, por lo que los contactos entre estaciones lunares y estaciones terrestres pueden significar un cuello de botella para todo plan de enrutamiento.

### 4.3. Escenarios

En [9] se publicaron tres escenarios distintos, donde se utilizan distintas bases lunares. En este trabajo se tomó como base para los experimentos el plan de contacto generado con bases lunares polares y de alta altitud. Los planes de contacto consisten de un periodo de 14 días de los cuales este trabajo tomó un intervalo arbitrario. La situación de estudio es el envío desde un sensor (id: 23) a una estación terrestre (id:1) en el intervalo de tiempo (90000 ms - 104000 ms) teniendo un tiempo límite de 14000ms para enviar el paquete antes de darse un *timeout*. Para la instancia inicial una probabilidad de error de 0.1 es tomada para todo contacto. El consumo de energía es de una unidad para todo contacto, y la latencia de 1000ms.

En la segunda instancia la probabilidad de error en contactos involucrando satélites y sensores es incrementada a 0.15, y los contactos entre estaciones terrestres y lunares decrementada a 0.05.

En la tercer instancia, los costes energéticos son incrementados a 1.5 unidades entre satélites/sensores y estación lunar. El coste energético del contacto entre estación terrestre y lunar es incrementado a 2 unidades.

En la cuarta instancia las latencias son incrementadas para comunicaciones entre satélite/sensor a estaciones lunares a 1500ms. En cuanto a contactos entre estaciones terrestres y

estaciones lunares la latencia es de 2000ms.

Por ultimo para el caso hipotético se asume una calidad empeorada de los sensores y todo contacto que involucre sensores tiene una probabilidad de error de 0.25.

El mismo experimento se repitió para el caso con y sin *Inter Satellite Link (ISL)*. Este ultimo disponiendo de contactos entre los satélites para habilitar más rutas posibles. Aunque los resultados con ISL corresponden a mayor SDP, y por ende más deseable, suponen un modelo de mayor tamaño. Mientras los modelos sin ISL rondan en el orden del millón de estados, aquellos con ISL se incrementan a la decena de millón de estados. Este crecimiento en un orden de magnitud es despreciable en el caso acotado de estudio, pero un factor a considerar en las implementaciones ad hoc, más aún con el incremento en el uso de memoria introducido por la extracción de estrategias.

## 4.4. Representación

Para visualizar las opciones propuestas se presenta en las tres dimensiones (SDP, EDT, EEC) el frente de Pareto como la superficie triangular entre los puntos óptimos. En el punto  $(0, T_{max}, 0)$  el punto que corresponde a la estrategia donde el paquete nunca se envía existe en todos los gráficos y sirve de referencia entre los escenarios. Esta superficie divide el espacio donde hacia arriba existen los valores alcanzables y hacia abajo los inalcanzables. En el plano  $EEC = 0$  las curvas de superficie nos brindan más detalle sobre la superficie.

Acompaña la visualización tri-dimensional la visualización de los ahorros de tiempo y energía acorde al punto de mayor SDP. En el origen el punto correspondiente a la estrategia que se conseguiría con PMC de objetivo simple. Los puntos dispersos representan estrategias están ubicados según la diferencia de EDT y EEC son coloreados y acompañados con anotaciones al SDP correspondiente.

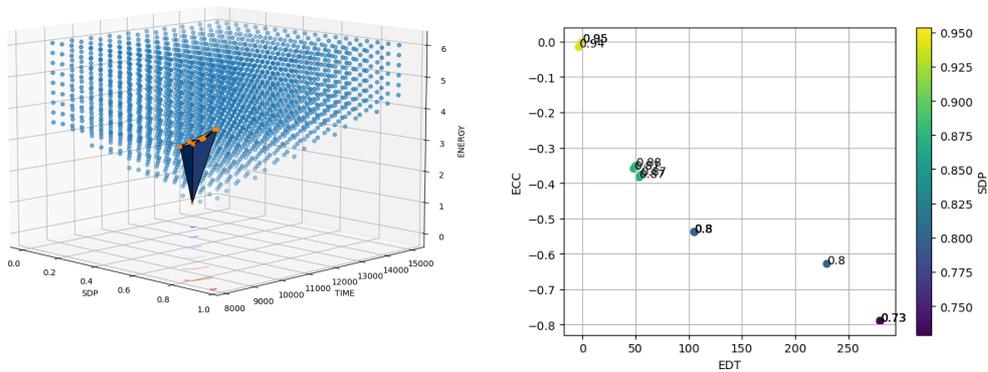
Esta segunda representación es más efectiva a la hora de visualizar la verdadera dispersión que propone el análisis multi objetivo y cuanto se puede ganar en cuanto a energía y tiempo. Se descarta de la muestra la estrategia correspondiente a no enviar el paquete.

## 4.5. Comparaciones

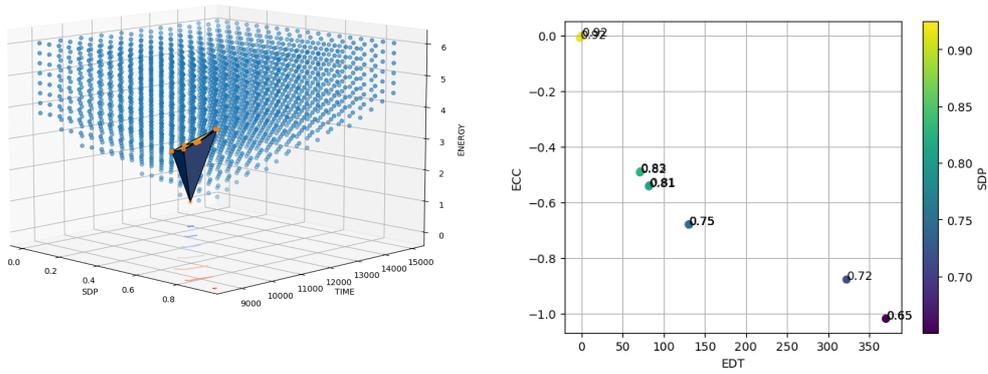
Observamos en paralelo las representaciones tridimensionales del frente de Pareto y el espacio de valores alcanzable para los 3 objetivos, con la dispersión de estrategias según su diferencia a la estrategia de mayor SDP. Esta última sería la estrategia obtenida por PMC de objetivo simple si maximizamos probabilidad de arribo, de ahora en adelante se la llamara estrategia «original» y el resto serán llamadas estrategias «alternativas». La estrategia original se ubica naturalmente en el origen del gráfico de diferencias. Se desestima de este último gráfico la estrategia correspondiente a no enviar el paquete.

### 4.5.1. Sin ISL

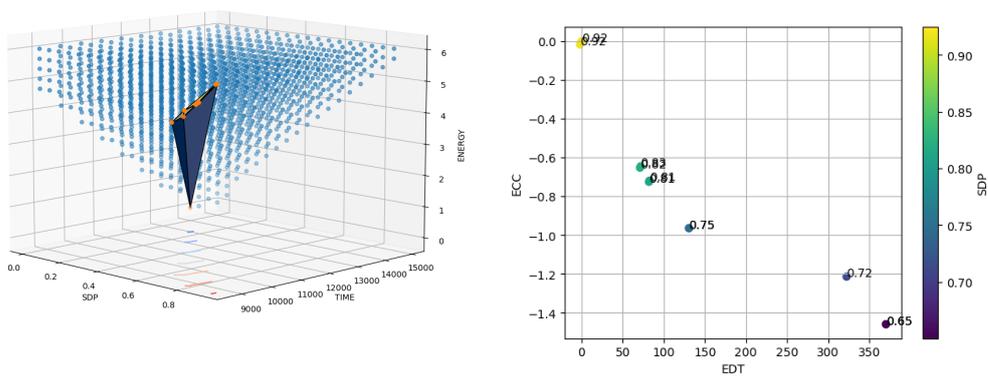
Viendo la figura 4.2 notamos que en el escenario inicial, aquel sin demasiada información del sistema, obtiene un frente de Pareto donde sus puntos óptimos no proponen significativas diferencias, y donde un ligero ahorro de energía trae a cuestras una significativa baja de SDP. Por otro lado ninguna estrategia es superadora en velocidad a la estrategia original, ya que todo salto supone la misma probabilidad de éxito y una misma latencia.



(a) Inicial

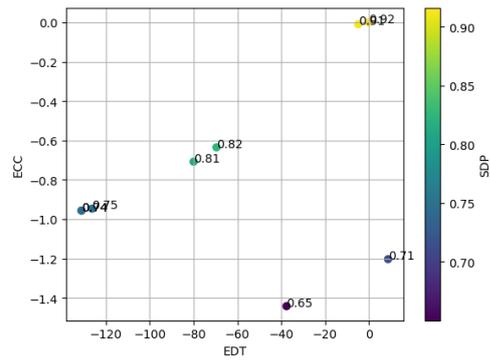
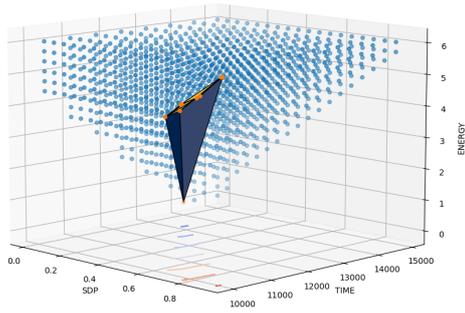


(b) Diferentes probabilidades

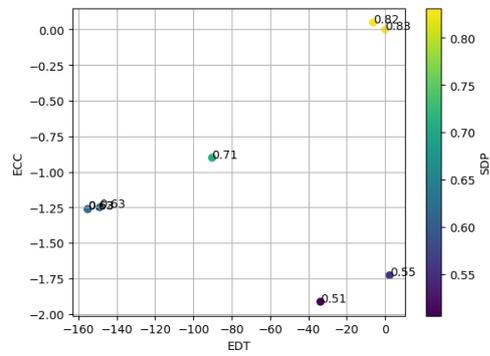
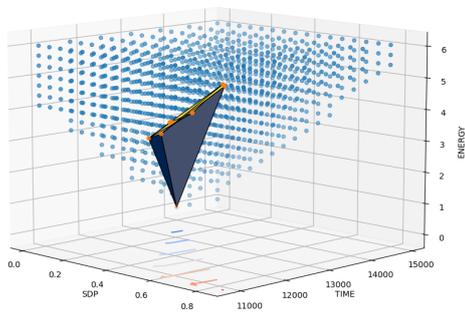


(c) Diferentes consumos energéticos

Figura 4.2: Experimentos sobre planes de contactos



(d) Diferentes latencias



(e) Sensores con mayor probabilidad a fallo

Figura 4.3: Experimentos sobre planes de contactos

En el segundo y tercer escenario las relaciones son muy similares, se distinguen tres «islas» en cuanto a las estrategias alternativas donde se consume más tiempo, menos energía a probabilidad de arribo similar. Esto se debe a leves variaciones de distintos caminos posibles.

En la figura 4.3 se observa un caso distinto en los siguientes escenarios (d) y (e) donde la incorporación de latencias resulta en una mayor variedad de estrategias que obtienen mejor tiempo y menor consumo de energía que la original.

#### 4.5.2. Con ISL

En la figura 4.4 y 4.5 apreciamos los resultados obtenidos en los mismos escenarios habilitando contactos entre satélites. Naturalmente esta incorporación provoca que los resultados obtenidos del escenario (b) cambien drásticamente de aquellos obtenidos del escenario (a). Exponiendo estrategias que a cambio de una pequeña depreciación del SDP obtienen tiempos de entrega de hasta 1.2 segundos más veloces consumiendo 0.8 unidades menos de energía.

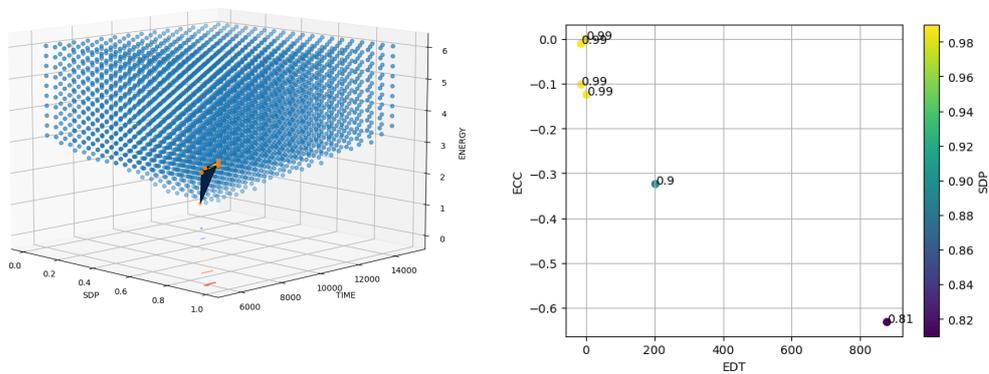
La incorporación de distintos consumos energéticos no propone grandes cambios en el panorama general, y la incorporación de diferentes latencias revierte los cambios ganados en el escenario (b). Remarcando la importancia de la precisión de estas informaciones incorporadas al modelo para no caer por falsas promesas de optimización.

### 4.6. Elección de estrategias

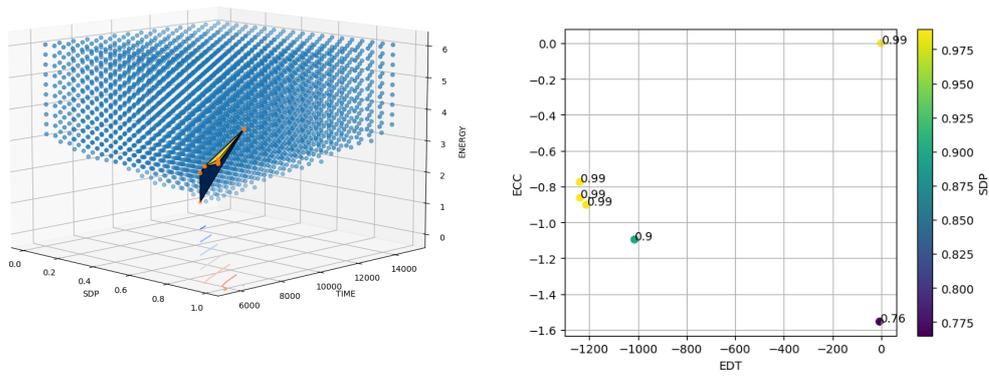
El frente de Pareto es una buena visualización de los conflictos entre objetivos que se utiliza interdisciplinariamente en la toma de decisiones donde los objetivos interactúan entre si. Aunque útil para un analista que estudie una red antes de ser desplegada, la ponderación de los objetivos puede estar sujeta a circunstancias variadas en el tiempo. Por eso, la elección de estrategias de enrutamiento según algún criterio va a variar y depender de situaciones propias al envío de datos. Estas circunstancias probablemente no serán conocidas a la hora de desplegar los planes de enrutamiento distribuidos entre nodos. Es por eso que una técnica automática de elección de estrategias de enrutamiento debe ser coordinada entre nodos para poder funcionar en paralelo.

La utilización de vectores peso que indiquen la dirección deseada de los objetivos es una forma de coordinar esta elección con antelación. Por ejemplo un vector  $(0.\hat{3}, 0.\hat{3}, 0.\hat{3})$  en este mismo caso de uso sería un vector peso que asigna igualdad de prioridades entre los objetivos de SDP, EDT, EEC. Si trazamos la recta descrita por ese vector encontraremos el punto donde se interseca con la superficie del frente de Pareto y podemos construir una estrategia correspondiente. Como ya se menciona en el capítulo 2 los puntos del frente de Pareto que coincidan con la recta trazada por el vector peso escogido pueden no coincidir con una estrategia determinística, sino a una aleatorización entre varias estrategias simples. Por lo que sería imposible garantizar que se pueda obtener un plan de enrutamiento a partir de la estrategia seleccionada. El algoritmo que automatiza la elección de estrategias debe contemplar la estrategia óptima más cercana a optimizar el peso escogido. Esta tarea escapa de lo que buscamos abordar en este trabajo.

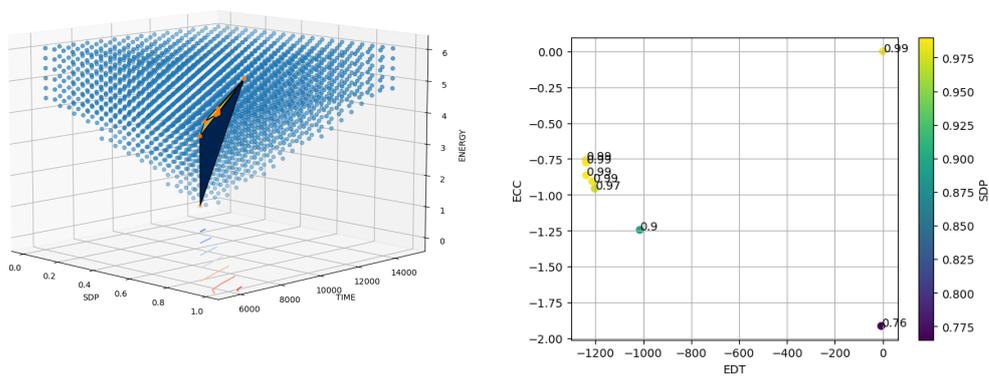
Una dificultad de distribuir el plan de enrutamiento será evitar el colapso con otros planes de enrutamiento que pueden ser obtenidos con el mismo peso de los objetivos. Por ejemplo, el nodo  $x$  y el nodo  $y$  pueden plantear tablas de enrutamiento para enviar un paquete al



(a) Inicial

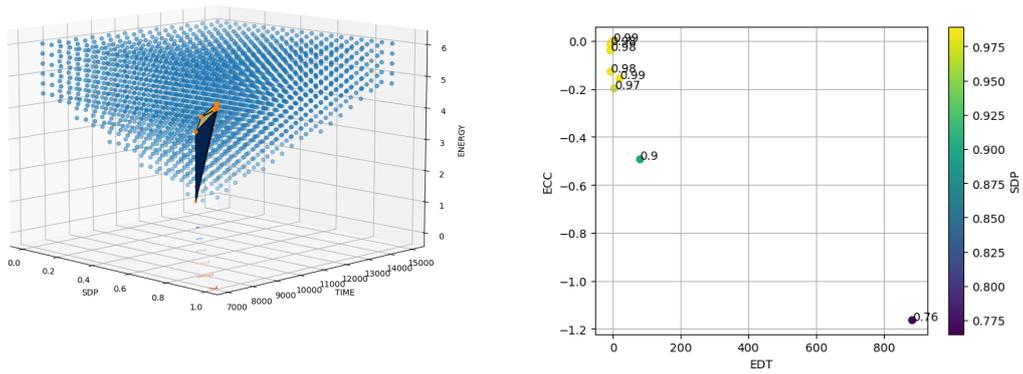


(b) Diferentes probabilidades

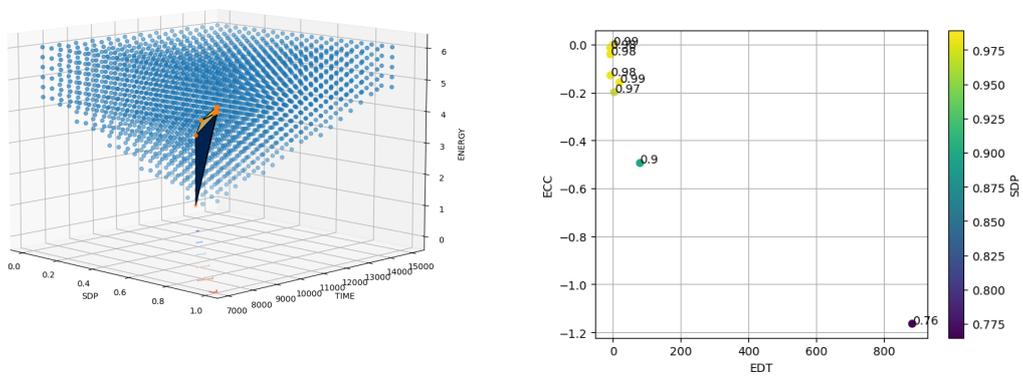


(c) Diferentes consumos energéticos

Figura 4.4: Experimentos sobre planes de contactos con ISL



(d) Diferentes latencias



(e) Sensores con mayor probabilidad a fallo

Figura 4.5: Experimentos sobre planes de contactos con ISL

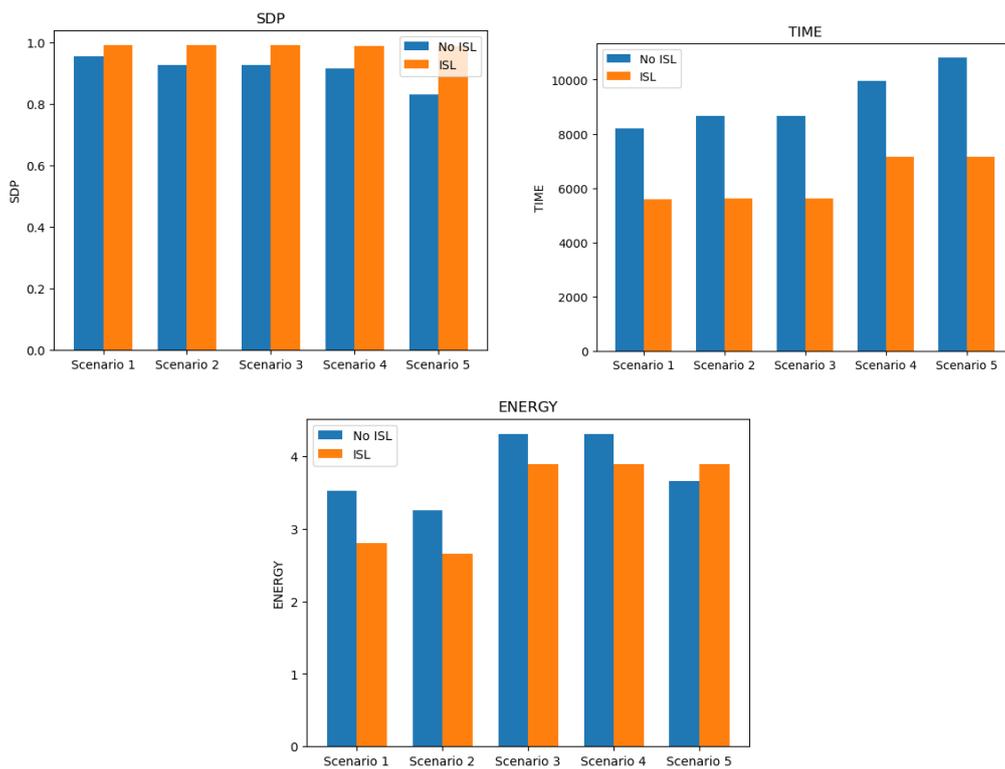


Figura 4.6: Mejores resultados entre ISL y no ISL para cada objetivo

nodo  $z$  que bajo la misma priorización a los objetivos requiera un distinto uso del nodo intermedio  $t$ . Una alternativa muy costosa en memoria sería que cada nodo mantenga tablas de enrutamiento para envíos de todos los nodos hacia todos los nodos, estas deberían ser actualizadas periódicamente y significarían un coste de recursos.

Otra alternativa es el acarreo de la tabla de enrutamiento específico para el paquete en la *metadata* del paquete, calculada por el nodo emisor. Se debe tener en cuenta que dependiendo de la extensión de la ruta y el tiempo de *timeout* esta tabla puede llegar a ser de tamaño significativo. Por otro lado, esta opción también permitiría utilizar estrategias aleatorizadas.

## 4.7. Diferencias entre redes con ISL y sin ISL

Una ventaja del análisis multi objetivo de estas redes es poder observar las diferencias entre la misma red con y sin ISL. La figura 4.6 busca evidenciar la diferencia al optimizar cada objetivo entre ambos casos. Los datos están tomados a partir de la máxima optimización de cada objetivo para los diferentes escenarios, donde se descartan estrategias de costo energético 0 por ser irrelevantes. El único caso donde las redes con ISL no son superadoras en cuanto a un objetivo a las no-ISL es el caso hipotético del escenario 5, dando a entender que las redes con ISL bajo las circunstancias dadas conllevan un coste energético mayor incluso cuando se optimiza este objetivo.

# 5

## Discusión

### 5.1. A futuro

Esta investigación sirve como base y prueba de concepto para mayores investigaciones. El análisis multi objetivo por PMC sí puede ser muy útil para un estudio previo de los compromisos subyacentes en una constelación tipo LEO, pero no como algoritmo de enrutamiento. Para apuntar a esas mejoras es necesario llevar a cabo implementaciones ad hoc, como se hizo en [16] RUCoP para el caso del objetivo simple. Además, el mismo modelo no soporta técnicas como duplicación de paquetes, por lo tanto deberían proponerse cambios para hacer provecho de esta técnica. En lo inmediato se puede experimentar con no limitar el análisis multi objetivo a SDP, tiempo y energía.

Otros factores pueden ser tomados en cuenta. Por ejemplo, incluir múltiples destinos. Un satélite sensor puede tener como objetivo enviar un paquete a una estación terrestre, indiferentemente de que estación es la que finalmente recibe el paquete, haciendo uso de múltiples predicados de alcanzabilidad o incluyendo más nodos en el conjunto de estados finales. Este último se puede implementar con lo presentado en este trabajo.

Explorando nuevos escenarios de interés podemos utilizar diversos criterios a la hora de asignar costes. Podemos recurrir, por ejemplo, a situaciones donde una estación terrestre busca enviar información a un satélite en órbita lunar. Las estaciones terrestres tienen a su disposición fuentes de energía económicas que pueden ser ignoradas, disminuyendo el coste energético del contacto a 0.

### 5.2. Caminos paralelos

También se implementaron en este trabajo, para planes de contacto pequeños, modelos cíclicos que emulen el comportamiento de las órbitas en los planes de contacto. Con *timeout* mayor a varios ciclos se podía observar planes de enrutamiento que optaban por probar múltiples veces el mismo contacto robusto, antes que contactos débiles. Esto abre las puertas a enrutamientos más económicos en coste energético sacrificando aún más tiempo. Esta idea fue eventualmente descartada, principalmente por ser considerada irrelevante para eventuales casos realistas, ya que las órbitas son de significativa mayor duración que un tiempo de envío razonable, y por el hecho de que los planes de contacto reales sufren correcciones periódicas

debido a cambios en las orbitas de los satélites. Sin embargo, puede ser de interés para otros tipos de DTNs, ya que presenta significativas mejoras en la minimización del coste energético.

La verificación estadística de modelos puede ser también implementada para el análisis multi objetivo de DTNs, como se hizo para SDP en [12]. Este trabajo puede servir como punto de comparación para estas implementaciones, que pueden soportar el envío de múltiples copias.

Otra alternativa en el trabajo es modificar la forma que se pondera el costo de la energía en la estructura de costos del modelo. El planteo intuitivo utilizado entiende que un contacto vinculando nodos entre la superficie lunar y la superficie terrestre consumiría más energía que un contacto entre una base lunar y un satélite en orbita baja. Se puede cambiar el modelo para que el coste energético haga referencia al valor que tiene la energía para cada nodo. Un nodo en un polo lunar que tiene acceso a energía solar por prolongadas horas del día puede tener mayor libertad de consumir energía, dando lugar a priorizar otros objetivos como pueden ser el tiempo de entrega y la probabilidad de éxito. Una diferenciación explicita a partir no solo de los tipos de nodo sino también su situación particular sería interesante a la hora de llevar a la practica estas ideas.

### 5.3. Problemas no tomados en cuenta

La suposición durante el diseño del modelo de evitar ciclos en el camino no se cumple necesariamente en todo plan de contacto. Por ejemplo, un plan de contacto puede constar de un nodo con un contacto muy robusto al nodo receptor final, pero en un intervalo de tiempo pequeño; por lo tanto una estrategia válida sería enviar el paquete a ese nodo y en caso de fallar lo retorne y continúe el camino por otro nodo. Esta complejidad no fue tomada en cuenta ya que los escenarios planteados no presentaban grandes diferencias entre los contactos individualmente.

Otra problemática abordada en la literatura que no ha sido contemplada en este trabajo es la de la congestión de la red. El análisis multi objetivo puede llevar a tendencias que sobrecarguen ciertos nodos con propiedades deseables para alguno de los objetivos. Un nodo puede ser eficiente en cuanto a energía, veloz, o poseer contactos robustos. El análisis planteado en este trabajo ignora otros paquetes que puedan estar utilizando la red y no da lugar a ninguna solución. Para afrontar este problema, una comunicación entre los nodos debería alertar sobre el estado de la red para modelar la no-congestión como un objetivo. Por motivos ya discutidos PMC no es apto para plantear estrategias de múltiples copias en redes distribuidas, por lo tanto otras técnicas deberán ser barajadas.

# 6

## Conclusión

Este trabajo busca proponer la investigación de métodos de análisis multi objetivo en el enrutamiento en DTNs espaciales. Se utiliza PMC para tomar valores ideales que servirán de puntos de referencia en futuras implementaciones. Se estudian las dificultades y opciones disponibles en el uso de MDPs para este caso de uso. El modelado es prototípico en base a trabajos realizados en [10]. Se llevaron a cabo experimentos sobre el modelo a partir de planes contacto realistas tomados de [9] y se enseña la variedad de opciones posibles que pueden surgir en cuanto se enriquece el modelo. Existen alternativas a ser exploradas en el campo para obtener resultados aún más acertados como las mencionadas en el capítulo anterior. Las técnicas de análisis multi objetivo, como el planteo de frentes de Pareto, que se llevaron a cabo en este trabajo pueden ser útiles a la hora de un análisis previo al despliegue de la red, pero una alternativa automatizada donde se elijan las estrategias debe ser planteada para el uso práctico de esta tecnología. Adicionalmente se propuso una contribución a la herramienta STORM para extraer estrategias correspondientes a los puntos óptimos de Pareto en su implementación de PCAA, permitiendo estudiar las estrategias de los puntos óptimos obtenidos sobre estos modelos y otros MDPs soportados. En conclusión el análisis multi objetivo es una herramienta que puede aportar gran valor en DTNs espaciales donde el tiempo y la energía son bienes críticos.

A continuación los resultados obtenidos con STORM para los 5 escenarios sin ISL y los 5 Escenarios con ISL:

## Sin ISL

a)

```

1 Overapproximation of achievable values: Polytope with 19 Halfspaces:
2 ( 1, 0, 0) * x <= 0.953452
3 ( 0, -1, 0) * x <= -8199.65
4 ( 0, 0, -1) * x <= 0
5 ( 0, -0.000744592, -0.999255) * x <= -9.83376
6 ( 0.778762, -3.82285e-05, -0.2212) * x <= -0.504769
7 ( 0.817901, 0, -0.182099) * x <= 0
8 ( 0, -0.00676945, -0.993231) * x <= -59.7722
9 ( 0, -0.000637622, -0.999362) * x <= -8.92671
10 ( 0, -0.00190222, -0.998098) * x <= -19.5684
11 ( 0, -0.00336673, -0.996633) * x <= -31.7098
12 ( 0, -0.00143669, -0.998563) * x <= -15.7024
13 ( 0.705123, -9.49062e-05, -0.294782) * x <= -1.32056
14 ( 0.638372, -0.000261217, -0.361367) * x <= -3.02048
15 ( 0, -0.00644237, -0.993558) * x <= -57.0826
16 ( 0.838733, -9.74385e-05, -0.16117) * x <= -0.694152
17 ( 0, -0.00296654, -0.997033) * x <= -28.407
18 ( 0.995818, -0.00418191, 0) * x <= -33.3544
19 ( 0.673444, -0.00131049, -0.325245) * x <= -11.5094
20 ( 0.663381, -0.000127034, -0.336492) * x <= -1.77848
21
22 15 Pareto optimal points found:
23 (0.9534518093, 8202.949069, 4.309490899 )
24 (0.9533861927, 8202.913346, 4.309417992 )
25 (0.7289999271, 8483.009041, 3.519999919 )
26 (0.8812808093, 8253.031369, 3.958300899 )
27 (0.8018270927, 8308.086176, 3.771918992 )
28 (0.8011709927, 8308.420787, 3.771189992 )
29 (0.8011709993, 8432.926699, 3.681189999 )
30 ( 0, 14000, 0 )
31 (0.728999271, 8483.010143, 3.519999919 )
32 (0.8674370927, 8256.451106, 3.925818992 )
33 (0.8739980927, 8251.287599, 3.950208992 )
34 (0.8011709271, 8308.420831, 3.771189919 )
35 (0.7289999993, 8483.008999, 3.519999999 )
36 (0.9396080927, 8199.652529, 4.294108992 )
37 (0.8740637093, 8258.039599, 3.933181899 )

```

b)

```

1 Overapproximation of achievable values: Polytope with 17 Halfspaces:
2 ( 1, 0, 0) * x <= 0.924537
3 ( 0, -1, 0) * x <= -8668.78
4 ( 0, 0, -1) * x <= 0
5 ( 0, -0.000809953, -0.99919) * x <= -10.5771
6 ( 0.790698, -3.37625e-05, -0.209268) * x <= -0.433526
7 ( 0.820711, 0, -0.179289) * x <= 0
8 ( 0, -0.0071959, -0.992804) * x <= -66.6152
9 ( 0, -0.000656343, -0.999344) * x <= -9.1888

```

```

10 ( 0, -0.00177242, -0.998228) * x <= -19.1886
11 ( 0, -0.00324565, -0.996754) * x <= -32.1287
12 ( 0, -0.00141005, -0.99859) * x <= -16.0006
13 ( 0, -0.00645614, -0.993544) * x <= -60.196
14 ( 0.702109, -0.000103516, -0.297788) * x <= -1.44057
15 ( 0.618525, -0.000288677, -0.381186) * x <= -3.44837
16 ( 0, -0.00281388, -0.997186) * x <= -28.3511
17 ( 0.843625, 0, -0.156375) * x <= 0.111446
18 ( 0.650455, -0.000144187, -0.349401) * x <= -2.01862
19
20 13 Pareto optimal points found:
21 (0.8234481515, 8741.707218, 3.782793778 )
22 ( 0, 14000, 0 )
23 (0.9245370326, 8670.676387, 4.275078126 )
24 (0.8126493279, 8753.344295, 3.736045345 )
25 ( 0.65025, 9040.953934, 3.256973684 )
26 (0.7151124375, 8993.030509, 3.397499934 )
27 (0.7475436557, 8801.379889, 3.595263059 )
28 (0.8095490578, 8752.645805, 3.732781903 )
29 (0.6502499995, 9040.953935, 3.256973684 )
30 (0.8272433763, 8742.561525, 3.786788751 )
31 (0.9161087765, 8668.783306, 4.266206278 )
32 (0.7477753078, 8801.261746, 3.595506903 )
33 (0.7475436562, 8801.379888, 3.595263059 )

```

c)

```

1 Overapproximation of achievable values: Polytope with 16 Halfspaces:
2 ( 1, 0, 0) * x <= 0.924537
3 ( 0, -1, 0) * x <= -8668.78
4 ( 0, 0, -1) * x <= 0
5 ( 0, -0.00108772, -0.998912) * x <= -14.132
6 ( 0, -0.00402079, -0.995979) * x <= -40.1668
7 ( 0.813277, -5.53215e-05, -0.186668) * x <= -0.764711
8 ( 0.86122, 0, -0.13878) * x <= 0.00226656
9 ( 0, -0.000866894, -0.999133) * x <= -12.1365
10 ( 0, -0.00923773, -0.990762) * x <= -85.7712
11 ( 0, -0.00206145, -0.997939) * x <= -22.9313
12 ( 0, -0.0068374, -0.993163) * x <= -64.8449
13 ( 0, -0.00508356, -0.994916) * x <= -49.5055
14 ( 0, -0.00863439, -0.991366) * x <= -80.5445
15 ( 0.797357, -7.12083e-05, -0.202572) * x <= -0.991401
16 ( 0.875889, 0, -0.124111) * x <= 0.0946423
17 ( 0.762137, -0.000181469, -0.237681) * x <= -2.16767
18
19 12 Pareto optimal points found:
20 (0.8272433763, 8742.561525, 5.117261878 )
21 (0.8234481515, 8741.707218, 5.109271931 )
22 (0.6502499898, 9040.953951, 4.302697347 )
23 (0.9245370326, 8670.676387, 5.762153128 )
24 (0.9161087765, 8668.783306, 5.744409431 )
25 (0.8095490578, 8752.645805, 5.036501306 )
26 ( 0, 14000, 0 )
27 (0.8126493279, 8753.344295, 5.04302819 )
28 (0.7151124375, 8993.030509, 4.547624868 )
29 (0.7477868904, 8801.26772, 4.79810069 )
30 (0.7475436562, 8801.379888, 4.797588618 )

```

```
31 (0.6502499995, 9040.953935, 4.302697367 )
```

d)

```
1
2 Overapproximation of achievable values: Polytope with 14 Halfspaces:
3 ( 1, 0, 0) * x <= 0.915877
4 ( 0, -1, 0) * x <= -9956.96
5 ( 0, 0, -1) * x <= 0
6 ( 0, -0.00133393, -0.998666) * x <= -17.7035
7 ( 0, -0.0115925, -0.988408) * x <= -120.158
8 ( 0.832178, -4.57801e-05, -0.167776) * x <= -0.629602
9 ( 0.861545, 0, -0.138455) * x <= 0.00189921
10 ( 0, -0.00108822, -0.998912) * x <= -15.235
11 ( 0, -0.00515707, -0.994843) * x <= -56.1117
12 ( 0.998699, -0.0013014, 0) * x <= -12.2123
13 ( 0, -0.977937, -0.0220629) * x <= -9737.38
14 ( 0.81868, -6.26884e-05, -0.181257) * x <= -0.873151
15 ( 0.81417, -0.000156015, -0.185674) * x <= -1.83635
16 ( 0.876614, 0, -0.123386) * x <= 0.0941488
17
18 10 Pareto optimal points found:
19 (0.9112400297, 10083.29626, 5.734159437 )
20 (0.7428293437, 9956.971381, 4.78766375 )
21 ( 0, 14000, 0 )
22 (0.8232162968, 10018.58753, 5.108783816 )
23 (0.7120235468, 10096.91502, 4.541121941 )
24 (0.9158769218, 10088.13176, 5.743921316 )
25 (0.6501687187, 10050.55861, 4.30252625 )
26 (0.7429065609, 9956.957559, 4.787826313 )
27 (0.809317203, 10008.15589, 5.036013191 )
28 (0.747543453, 9961.793066, 4.797588191 )
```

e)

```
1 Overapproximation of achievable values: Polytope with 13 Halfspaces:
2 ( 1, 0, 0) * x <= 0.830883
3 ( 0, -1, 0) * x <= -10803.7
4 ( 0, 0, -1) * x <= 0
5 ( 0, -0.00164706, -0.998353) * x <= -21.6478
6 ( 0, -0.0131133, -0.986887) * x <= -145.927
7 ( 0, -0.0011887, -0.998811) * x <= -16.6419
8 ( 0.87312, 0, -0.12688) * x <= 0.0276615
9 ( 0, -0.00533738, -0.994663) * x <= -61.9516
10 ( 0.998705, -0.00129468, 0) * x <= -13.3563
11 ( 0.828868, -6.71372e-05, -0.171065) * x <= -0.934204
12 ( 0.867981, 0, -0.132019) * x <= 0
13 ( 0.892814, 0, -0.107186) * x <= 0.144386
14 ( 0.818776, -0.000160587, -0.181064) * x <= -2.00095
15
16 9 Pareto optimal points found:
17 (0.8248679297, 10952.62236, 5.623673437 )
18 (0.8308826543, 10958.89297, 5.573836016 )
19 (0.6264210937, 10803.73867, 4.31096875 )
20 (0.5543435918, 10961.26602, 3.845164141 )
21 (0.5061867187, 10925.17539, 3.65940625 )
22 (0.7106482793, 10868.65391, 4.672273516 )
23 ( 0, 14000, 0 )
```

```

24 (0.6324959355, 10809.99852, 4.323757891 )
25 (0.6264812109, 10803.72791, 4.311095312 )

```

## Con ISL

a)

```

1 Overapproximation of achievable values: Polytope with 8 Halfspaces:
2 ( 1, 0, 0) * x <= 0.989813
3 ( 0, -1, 0) * x <= -5582
4 ( 0, 0, -1) * x <= 0
5 ( 0, -0.000605889, -0.999394) * x <= -6.61967
6 ( 0, -0.000378835, -0.999621) * x <= -5.25166
7 ( 0.775623, 0, -0.224377) * x <= 0.0248547
8 ( 0, -0.00799674, -0.992003) * x <= -47.9479
9 ( 0, -0.11836, -0.88164) * x <= -663.702
10
11 7 Pareto optimal points found:
12 (0.809999919, 6474.380186, 2.79999991 )
13 ( 0, 14000, 0 )
14 (0.9898126039, 5596.350728, 3.43180098 )
15 (0.9896506199, 5581.998488, 3.4212346 )
16 (0.9889127999, 5598.47372, 3.307692 )
17 (0.8990999028, 5799.012051, 3.107999892 )
18 (0.9896498982, 5582.819853, 3.330233798 )

```

b)

```

1 Overapproximation of achievable values: Polytope with 9 Halfspaces:
2 ( 1, 0, 0) * x <= 0.989758
3 ( 0, -1, 0) * x <= -5615.15
4 ( 0, 0, -1) * x <= 0
5 ( 0, -0.000411775, -0.999588) * x <= -5.47389
6 ( 0.776333, 0, -0.223667) * x <= 0.0262069
7 ( 0, -0.00545624, -0.994544) * x <= -33.9701
8 ( 0, -0.00037111, -0.999629) * x <= -5.19554
9 ( 0, -0.000540475, -0.99946) * x <= -6.26785
10 ( 0, -0.115832, -0.884168) * x <= -653.45
11
12 7 Pareto optimal points found:
13 (0.989757992, 6854.090645, 4.209285956 )
14 (0.9893311804, 5615.145691, 3.434620285 )
15 (0.9893287176, 5615.968924, 3.346242693 )
16 ( 0, 14000, 0 )
17 (0.7649999994, 6847.733159, 2.655263157 )
18 (0.9866252812, 5641.894983, 3.307340428 )
19 (0.8969624979, 5839.754969, 3.11329605 )

```

c)

```

1 Overapproximation of achievable values: Polytope with 10 Halfspaces:
2 ( 1, 0, 0) * x <= 0.989758
3 ( 0, -1, 0) * x <= -5615.15
4 ( 0, 0, -1) * x <= 0
5 ( 0, -0.00060135, -0.999399) * x <= -8.00108

```

```

6 ( 0.835502, 0, -0.164498) * x <= 0.019271
7 ( 0, -0.00562183, -0.994378) * x <= -36.4815
8 ( 0, -0.000542963, -0.999457) * x <= -7.60148
9 ( 0, -0.000835658, -0.999164) * x <= -9.43201
10 ( 0, -0.00170099, -0.998299) * x <= -14.4494
11 ( 0, -0.110236, -0.889764) * x <= -623.463
12
13 9 Pareto optimal points found:
14 (0.8969624973, 5839.75497, 4.5557796 )
15 (0.9732712499, 5651.853753, 4.843929605 )
16 (0.9893311804, 5615.145691, 5.045267242 )
17 (0.9866252812, 5641.894983, 4.894037105 )
18 (0.764999988, 6847.733198, 3.885526291 )
19 ( 0, 14000, 0 )
20 (0.9893311804, 5615.145691, 5.025711713 )
21 (0.9892652519, 5616.01957, 4.936999245 )
22 (0.989757992, 6854.090645, 5.801195183 )

```

d)

```

1 Overapproximation of achievable values: Polytope with 10 Halfspaces:
2 ( 1, 0, 0) * x <= 0.989116
3 ( 0, -1, 0) * x <= -7144.88
4 ( 0, 0, -1) * x <= 0
5 ( 0, -0.000733984, -0.999266) * x <= -9.78053
6 ( 0.835513, 0, -0.164487) * x <= 0.01893
7 ( 0, -0.00523262, -0.994767) * x <= -42.2711
8 ( 0, -0.000651003, -0.999349) * x <= -9.11404
9 ( 0, -0.00109677, -0.998903) * x <= -12.483
10 ( 0, -0.0160803, -0.98392) * x <= -119.745
11 ( 0, -0.139899, -0.860101) * x <= -1003.87
12
13 9 Pareto optimal points found:
14 (0.9829157144, 7144.883864, 5.008854579 )
15 (0.9829157144, 7144.883864, 5.035241716 )
16 (0.9829157144, 7145.586309, 4.920985829 )
17 (0.989116003, 7152.335898, 5.04909054 )
18 (0.764904375, 8035.668129, 3.885325 )
19 ( 0, 14000, 0 )
20 (0.8968503797, 7232.580495, 4.555543562 )
21 (0.9724369219, 7156.157055, 4.851002969 )
22 (0.9858074072, 7172.055445, 4.892315265 )

```

e)

```

1 Overapproximation of achievable values: Polytope with 10 Halfspaces:
2 ( 1, 0, 0) * x <= 0.989116
3 ( 0, -1, 0) * x <= -7144.88
4 ( 0, 0, -1) * x <= 0
5 ( 0, -0.000733984, -0.999266) * x <= -9.78053
6 ( 0.835513, 0, -0.164487) * x <= 0.01893
7 ( 0, -0.00523262, -0.994767) * x <= -42.2711
8 ( 0, -0.000651003, -0.999349) * x <= -9.11404
9 ( 0, -0.00109677, -0.998903) * x <= -12.483
10 ( 0, -0.0160803, -0.98392) * x <= -119.745
11 ( 0, -0.139899, -0.860101) * x <= -1003.87
12
13 9 Pareto optimal points found:

```

```
14 (0.9829157144, 7144.883864, 5.008854579 )
15 (0.9829157144, 7144.883864, 5.035241716 )
16 (0.989116003, 7152.854572, 5.049725391 )
17 (0.9829157144, 7145.586309, 4.920985829 )
18 (0.764904375, 8035.668129, 3.885325 )
19 ( 0, 14000, 0 )
20 (0.8968503797, 7232.580495, 4.555543562 )
21 (0.9724369219, 7156.157055, 4.851002969 )
22 (0.9858074072, 7172.055445, 4.892315265 )
```

# Referencias

- [1] L. Benini, A. Bogliolo, G. Paleologo y G. De Micheli, "Policy optimization for dynamic power management," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 18, n.º 6, págs. 813-833, jun. de 1999, Conference Name: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, ISSN: 1937-4151. DOI: 10.1109/43.766730.
- [2] S. Jain, K. Fall y R. Patra, "Routing in a delay tolerant network," en *Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, ép. SIGCOMM '04, New York, NY, USA: Association for Computing Machinery, 30 de ago. de 2004, págs. 145-158, ISBN: 978-1-58113-862-7. DOI: 10.1145/1015467.1015484. dirección: <https://dl.acm.org/doi/10.1145/1015467.1015484> (visitado 26-04-2023).
- [3] K. Scott y S. C. Burleigh, "Bundle Protocol Specification," Internet Engineering Task Force, Request for Comments RFC 5050, nov. de 2007, Num Pages: 50. DOI: 10.17487/RFC5050. dirección: <https://datatracker.ietf.org/doc/rfc5050> (visitado 15-04-2023).
- [4] K. Etessami, M. Kwiatkowska, M. Y. Vardi y M. Yannakakis, "Multi-Objective Model Checking of Markov Decision Processes," *Log.Meth.Comput.Sci.*, vol. 4, n.º 4, pág. 8, 12 de nov. de 2008, ISSN: 18605974. DOI: 10.2168/LMCS-4(4:8)2008. arXiv: 0810.5728[cs]. dirección: <http://arxiv.org/abs/0810.5728> (visitado 07-02-2023).
- [5] S. Burleigh y E. Birrane, "Toward a communications satellite network for humanitarian relief," 18 de dic. de 2011. DOI: 10.1145/2185216.2185280.
- [6] V. Forejt, M. Kwiatkowska y D. Parker, *Pareto curves for probabilistic model checking*, 24 de jun. de 2012. arXiv: 1206.6295[cs]. dirección: <http://arxiv.org/abs/1206.6295> (visitado 11-01-2023).
- [7] G. Araniti, N. Bezirgiannidis, E. Birrane et al., "Contact graph routing in DTN space networks: overview, enhancements and performance," *IEEE Communications Magazine*, vol. 53, n.º 3, págs. 38-46, mar. de 2015, Conference Name: IEEE Communications Magazine, ISSN: 1558-1896. DOI: 10.1109/MCOM.2015.7060480.
- [8] J. Fraire y J. M. Finochietto, "Routing-aware fair contact plan design for predictable delay tolerant networks," *Ad Hoc Networks*, New Research Challenges in Mobile, Opportunistic and Delay-Tolerant Networks, vol. 25, págs. 303-313, 1 de feb. de 2015, ISSN: 1570-8705. DOI: 10.1016/j.adhoc.2014.07.006. dirección: <https://www.sciencedirect.com/science/article/pii/S1570870514001371> (visitado 17-04-2023).
- [9] M. Feldmann, J. A. Fraire y F. Walter, "Tracking lunar ring road communication," en *2018 IEEE International Conference on Communications (ICC)*, Kansas City, MO: IEEE,

- mayo de 2018, págs. 1-7, ISBN: 978-1-5386-3180-5. DOI: 10.1109/ICC.2018.8423031. dirección: <https://ieeexplore.ieee.org/document/8423031/> (visitado 31-01-2023).
- [10] F. D. Raverta, R. Demasi, P. G. Madoery, J. A. Fraire, J. M. Finochietto y P. R. D'Argenio, "A markov decision process for routing in space DTNs with uncertain contact plans," en *2018 6th IEEE International Conference on Wireless for Space and Extreme Environments (WiSEE)*, Huntsville, AL, USA: IEEE, dic. de 2018, págs. 189-194, ISBN: 978-1-5386-4244-3. DOI: 10.1109/WiSEE.2018.8637330. dirección: <https://ieeexplore.ieee.org/document/8637330/> (visitado 03-01-2023).
- [11] J. Foust, "SpaceX's space-Internet woes: Despite technical glitches, the company plans to launch the first of nearly 12,000 satellites in 2019," *IEEE Spectrum*, vol. 56, n.º 1, págs. 50-51, ene. de 2019, Conference Name: IEEE Spectrum, ISSN: 1939-9340. DOI: 10.1109/MSPEC.2019.8594798.
- [12] P. R. D'Argenio, J. A. Fraire y A. Hartmanns, "Sampling distributed schedulers for resilient space communication," en *NASA Formal Methods*, R. Lee, S. Jha, A. Mavridou y D. Giannakopoulou, eds., vol. 12229, Series Title: Lecture Notes in Computer Science, Cham: Springer International Publishing, 2020, págs. 291-310, ISBN: 978-3-030-55753-9 978-3-030-55754-6. DOI: 10.1007/978-3-030-55754-6\_17. dirección: [https://link.springer.com/10.1007/978-3-030-55754-6\\_17](https://link.springer.com/10.1007/978-3-030-55754-6_17) (visitado 03-01-2023).
- [13] F. Delgrange, J.-P. Katoen, T. Quatmann y M. Randour, *Simple Strategies in Multi-Objective MDPs (Technical Report)*, 17 de feb. de 2020. DOI: 10.48550/arXiv.1910.11024. arXiv: 1910.11024[cs]. dirección: <http://arxiv.org/abs/1910.11024> (visitado 07-02-2023).
- [14] T. Quatmann, S. Junges y J.-P. Katoen, "Markov automata with multiple objectives," *Form Methods Syst Des*, 29 de mar. de 2021, ISSN: 1572-8102. DOI: 10.1007/s10703-021-00364-6. dirección: <https://doi.org/10.1007/s10703-021-00364-6> (visitado 07-02-2023).
- [15] T. Quatmann y J.-P. Katoen, "Multi-objective optimization of long-run average and total rewards," en *Tools and Algorithms for the Construction and Analysis of Systems*, J. F. Groote y K. G. Larsen, eds., vol. 12651, Series Title: Lecture Notes in Computer Science, Cham: Springer International Publishing, 2021, págs. 230-249, ISBN: 978-3-030-72015-5 978-3-030-72016-2. DOI: 10.1007/978-3-030-72016-2\_13. dirección: [http://link.springer.com/10.1007/978-3-030-72016-2\\_13](http://link.springer.com/10.1007/978-3-030-72016-2_13) (visitado 03-01-2023).
- [16] F. D. Raverta, J. A. Fraire, P. G. Madoery, R. A. Demasi, J. M. Finochietto y P. R. D'Argenio, "Routing in delay-tolerant networks under uncertain contact plans," *Ad Hoc Networks*, vol. 123, pág. 102663, dic. de 2021, ISSN: 15708705. DOI: 10.1016/j.adhoc.2021.102663. dirección: <https://linkinghub.elsevier.com/retrieve/pii/S1570870521001761> (visitado 03-01-2023).
- [17] C. Hensel, S. Junges, J.-P. Katoen, T. Quatmann y M. Volk, "The probabilistic model checker storm," *Int J Softw Tools Technol Transfer*, vol. 24, n.º 4, págs. 589-610, 1 de ago. de 2022, ISSN: 1433-2787. DOI: 10.1007/s10009-021-00633-z. dirección: <https://doi.org/10.1007/s10009-021-00633-z> (visitado 27-01-2023).

- 
- [18] K. Fall, "A delay-tolerant network architecture for challenged internets,"
- [19] "Storm – A Modern Probabilistic Model Checker – Properties." (), dirección: <https://www.stormchecker.org/documentation/background/properties.html> (visitado 21-04-2023).

---

Los abajo firmantes, miembros del Tribunal de evaluación de tesis, damos fe que el presente ejemplar impreso se corresponde con el aprobado por este Tribunal.