



UNIVERSIDAD DE BUENOS AIRES  
FACULTAD DE CIENCIAS EXACTAS Y NATURALES  
DEPARTAMENTO DE COMPUTACIÓN

# Desarrollo y estudio de un algoritmo de control colaborativo de consumo eléctrico residencial para evitar interrupciones del suministro eléctrico

Tesis de Licenciatura en Ciencias de la Computación

Javier Garrone

Director: Dr. Pablo Turjanski

Buenos Aires, 2018

# DESARROLLO Y ESTUDIO DE UN ALGORITMO DE CONTROL COLABORATIVO DE CONSUMO ELÉCTRICO RESIDENCIAL PARA EVITAR INTERRUPCIONES DEL SUMINISTRO ELÉCTRICO

La utilización concurrente y excesiva de equipos de alto consumo eléctrico (por ejemplo, el aire acondicionado) en los hogares de la Ciudad de Buenos Aires y sus alrededores, sumado a otros factores externos, ocasiona múltiples interrupciones en el servicio eléctrico cada verano. Un equipo de estudiantes e investigadores de la Universidad de Buenos Aires propuso construir un sistema para controlar el uso desmedido que se le da a estos aparatos en el entorno residencial. En el esquema propuesto, los usuarios de la red eléctrica son seleccionados, de manera rotativa, para colaborar apagando sus equipos de alto consumo eléctrico durante un breve intervalo de tiempo, con el objetivo de no exceder el límite del suministro eléctrico y finalmente no sufrir fallas en el servicio. En el presente trabajo se analizan distintas estrategias para la selección y rotación de usuarios que se podrían implementar en dicho sistema, encontrando similitudes con el problema de la mochila. Primero, cada estrategia es dilucidada utilizando escenarios pequeños (pocos usuarios). Luego, se evalúa el desempeño de las estrategias sobre escenarios que intentan reproducir las mismas características que posee un entorno real. Las estrategias planteadas comienzan siendo elementales, demostrando rápidamente su baja efectividad incluso ante una estrategia que toma decisiones de manera aleatoria. Finalmente, se proponen estrategias más complejas que son puestas a prueba en el contexto de ciertos escenarios más realistas, obteniendo resultados más promisorios.

**Palabras claves:** suministro eléctrico, interrupción suministro, optimización, problema de la mochila.

# DEVELOPMENT AND STUDY OF A COLLABORATIVE CONTROL ALGORITHM FOR RESIDENTIAL ELECTRICAL CONSUMPTION TO AVOID POWER SUPPLY INTERRUPTIONS

The concurrent and excessive use of high-energy-consumption household appliances (e.g. air conditioning) in Buenos Aires and its suburbs, in addition to other external factors, causes multiple interruptions in the electric service every summer. A team of students and researchers of the University of Buenos Aires (UBA) suggested implementing a program to control the immoderate use of these appliances in residential areas. In the proposed scheme, the users of the electricity network are selected and systematically rotated to collaborate by shutting off their high-consumption appliances for a brief period of time, with the objective of not exceeding the electric supply limit and ultimately avoiding flaws in the electric service. In this dissertation, some strategies for the selection and rotation of users are analyzed, which may help execute this program, finding similarities to the knapsack problem. First, each strategy is elucidated, utilizing a small setting with few users. Afterwards, the performance of these strategies is assessed when applied to those settings which bear more resemblance to reality. At first, the posed strategies are elementary, quickly proving their low efficiency even compared to a strategy where decisions are randomly taken. Finally, more complex strategies are suggested and put to the test in more realistic contexts, thus, obtaining more promissory results.

**Keywords:** electricity supply, supply interruption, optimization, knapsack problem.

## Índice general

1..	Introducción . . . . .	1
1.1.	Escenario inicial . . . . .	5
1.2.	El problema de la mochila . . . . .	6
1.3.	Períodos y cronogramas . . . . .	7
2..	Desarrollo . . . . .	8
2.1.	Algoritmo de selección . . . . .	8
2.2.	Estrategias “High-To-Low” y “Low-To-High” . . . . .	9
2.3.	Estrategias “Alternate by Demand” y “Balance Demand” . . . . .	17
2.4.	Estrategia “Greedy-N-Times” . . . . .	22
2.5.	Estrategias de período múltiple . . . . .	29
2.5.1.	“Greedy-N-Times” con múltiples períodos . . . . .	31
2.5.2.	Búsqueda local sobre cronogramas . . . . .	33
3..	Casos de Estudio . . . . .	45
3.1.	Diseño de Escenarios . . . . .	45
3.2.	Resultados . . . . .	46
4..	Conclusiones y trabajo futuro . . . . .	53
4.1.	Conclusiones . . . . .	53
4.2.	Trabajo futuro . . . . .	54
5..	Anexo I: Datos de escenarios reales . . . . .	55
6..	Anexo II: Ejecución de los experimentos y datos resultantes . . . . .	56

## 1. INTRODUCCIÓN

Los usuarios residenciales de la red eléctrica de Argentina han experimentado, en los últimos años y usualmente en verano, numerosos cortes en el suministro eléctrico, los cuales han llegado a extenderse por varios días en algunas zonas. Según registros de la empresa Edenor, la cantidad de cortes ocurridos en el año 2014 superaron casi en un 100 % a la cantidad dada en el año 2011. En cuanto a la duración de los cortes, la diferencia entre tales períodos llegó a sobrepasar dicho porcentaje [1]. Estos cortes en el servicio eléctrico son ocasionados por distintos motivos. Algunos son originados por eventos extremos, generalmente relacionados con el clima [2], en cuyo caso el foco debe estar puesto en la preparación y recuperación ante estos sucesos [3]. Otras veces, los cortes se deben a fallas en el sistema eléctrico, de los cuales algunos podrían ser evitados.

El sistema eléctrico de Argentina se encuentra dividido en tres etapas: generación, transmisión y distribución. La energía eléctrica es generada en múltiples plantas generadoras de distintos tipos pertenecientes a decenas de empresas. La mayor parte de la energía eléctrica es generada en plantas de generación térmica (mediante la utilización de combustibles fósiles), dejando en segundo lugar a las de generación hidroeléctrica. La energía eléctrica es producida bajo demanda para los usuarios del Sistema Argentino de Interconexión eléctrica (SADI) y es transportada a grandes distancias a través de líneas de alta tensión. Previo a su transporte, la tensión de la energía eléctrica es aumentada en subestaciones transformadoras elevadoras, con el objetivo de minimizar el desperdicio eléctrico generado por el largo recorrido. Antes de ser entregada a la red de distribución, la tensión de la energía eléctrica es reducida en subestaciones transformadoras reductoras. Por último, las empresas distribuidoras se encargan de llevar la energía eléctrica hacia sus centros de transformación, donde su tensión es reducida nuevamente, y de allí es transportada hacia los consumidores mediante redes de baja tensión. Cada provincia de la República Argentina cuenta con entre una y dos empresas distribuidoras de energía eléctrica, a excepción de la Provincia de Buenos Aires, en donde existen un total de seis empresas distribuidoras: tres de ellas se encargan de la distribución en el interior de la provincia; otra empresa denominada EDELAP distribuye exclusivamente en la capital de la provincia; y dos empresas, denominadas EDENOR y EDESUR, se dividen el territorio del AMBA (Área Metropolitana de Buenos Aires), formado por la Ciudad de Buenos Aires y sus alrededores.

Una falla o deficiencia en cualquiera de las tres etapas necesarias para proveer energía eléctrica a los consumidores puede dejar sin suministro a un gran número de usuarios. Son recurrentes las fallas en los sistemas de distribución, muchas veces debido a la sobrecarga de alguno de los componentes de la red. Ésto se debe, entre otras cosas, a la gran demanda eléctrica necesaria para alimentar a los artefactos de alto consumo (por ejemplo, acondicionadores de aire) y a su utilización concurrente por gran parte de la población en horarios de mayor demanda.

Existe una clara relación entre los días con temperaturas extremas y los días con demanda eléctrica elevada, como podemos ver en la figura 1.1, lo cual sugiere que los artefactos utilizados para refrigerar y calefaccionar los hogares toman parte en la problemática de las interrupciones al servicio eléctrico. Analizando dicha figura, un interrogante que se puede plantear es: ¿por qué los cortes de electricidad predominan en verano, si su consumo

es equivalente al del invierno? Una posibilidad es que el consumo eléctrico se encuentre distribuido más uniformemente en los días de frío, es decir, que a pesar de tener un consumo alto durante el día, la potencia demandada en un momento dado no alcance un pico. Por otro lado, en los días de calor, la temperatura de los transformadores y otras partes de la infraestructura eléctrica es elevada, lo que produce una disminución en el rendimiento y la carga soportada por éstos.

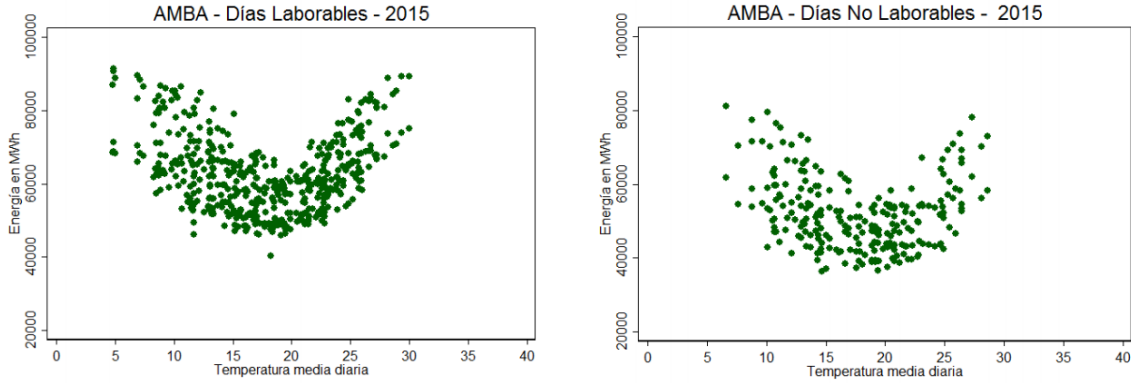


Fig. 1.1: Relación temperatura - consumo en días laborables y no laborables. Fuente: MINEM[4].

En la tabla 1.1 se muestra la potencia requerida por equipos de aire acondicionado (sin tecnología “Inverter”) de distintas frigorías, siendo de 1,3 kW en equipos de 2.200 frigorías y sobrepasando los 2 kW en equipos de 3.500 frigorías[5][6]. A modo de comparación, una heladera con freezer requiere 0,2 kW, un televisor a color requiere 0,1 kW y un ventilador de techo tan sólo 60 W. Esta gran diferencia de potencia entre el aire acondicionado y otros electrodomésticos indica que tener prendido el primero incide fuertemente en el consumo eléctrico hogareño. En [7] se deduce que el mayor potencial de ahorro eléctrico se encuentra en los artefactos de refrigeración y calefacción. Por otra parte, en la figura 1.2 podemos observar un crecimiento abrupto en la producción de acondicionadores domésticos en la última década. La cantidad de acondicionadores de aire para uso hogareño producidos a partir del año 2005 supera ampliamente a la producida en años anteriores, con un promedio de aproximadamente 80.000 unidades por año entre 1993 y 2004, contra un promedio también aproximado de 1.210.000 entre 2005 y 2016. Se prevee que en los próximos años la demanda eléctrica residencial dada por los equipos de aire acondicionado continuará en aumento [8]. El hecho de que hacia diciembre de 2013 un 38,7% de los hogares del AMBA[9] poseían al menos un equipo de aire acondicionado (casi la mitad adquiridos entre 2011 y 2013), sumado al alto consumo que requieren para funcionar, hace evidente que su utilización podría ser un factor determinante en el elevado consumo eléctrico residencial y, por consiguiente, en los cortes de luz.

Una solución posible a esta problemática sería controlar el uso simultáneo de los equipos de aire acondicionado, cuyo consumo supera por un orden de magnitud al resto de los electrodomésticos. Al detectar que la demanda eléctrica de un conjunto de hogares que comparten un mismo tendido eléctrico se acerca al límite soportado por tal infraestructura, se seleccionaría a un grupo de usuarios que residen en dichos hogares para que disminuyan su consumo apagando sus equipos de aire acondicionado. El grupo de usuarios cuyo consumo debe ser disminuido sería rotado cada cierto intervalo de tiempo, de manera que a ningún hogar se le niegue ser refrigerado por una cantidad excesiva de tiempo. Al

Frig. (kcal/h)	Pot. (kW)
2.200	1,3
3.000	2
3.500	2,15
5.000	2,64
9.000	3,5

Tab. 1.1: Potencia requerida por equipos de aire acondicionado de uso residencial según cantidad de frigorías.

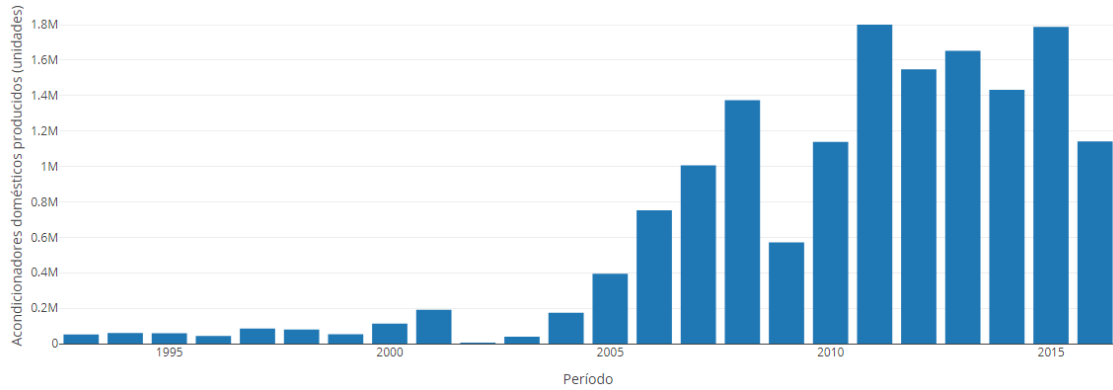


Fig. 1.2: Cantidad de acondicionadores de aire domésticos producidos entre los años 1993 y 2016. Fuente: INDEC[10].

optar por el apagado total de los equipos de aire acondicionado, nos aseguramos de que la demanda requerida para refrigerar los hogares de los usuarios seleccionados sea liberada completamente, simplificando el estudio de los escenarios. Una consecuencia de esto a tener en cuenta es que en el momento en el que se habilita a los usuarios a volver a encender sus equipos, éstos pueden llegar a ocasionar un pico de demanda al ser encendidos simultáneamente. Para mitigar este problema, se podría habilitar el encendido de los equipos con un pequeño intervalo de tiempo entre cada par de usuarios. Otro enfoque distinto al de apagar los equipos podría ser el de solicitar a los usuarios que ajusten el termostato de sus equipos a un valor más alto (por ejemplo, 24°C), lo cual disminuiría la cantidad de veces que los equipos deben encender su compresor (eventos causantes de los picos en la demanda de tales artefactos).

Esta solución puede ser categorizada como una técnica de *Demand Side Management* (DSM), y más específicamente, como *Demand Response* (DR). DSM es un conjunto de técnicas orientadas a manipular el consumo eléctrico mediante el accionar de los usuarios. Las estrategias de DR buscan reducir la utilización de los artefactos eléctricos durante los horarios de mayor consumo o mover las actividades que involucran a estos artefactos hacia horarios de menor consumo, con el objetivo de reducir los picos en la demanda. La aplicación de cualquier técnica de DR depende de que exista un sistema de información y comunicación implementado sobre la red eléctrica, lo cual se denomina *Smart Grid*. Una explicación más detallada de estos conceptos se puede ver en [11, 12].

Las estrategias de DR han recibido mucha atención en los últimos años al proporcionar beneficios tanto para los clientes como para las prestadoras del servicio eléctrico. Por un lado, el usuario percibe una reducción en su tarifa y sufre un menor número de interrup-

ciones al suministro eléctrico; por otro, el consumo eléctrico diario es aplanado, reduciendo la infraestructura necesaria para lidiar con altos picos en la demanda. En [13] se analiza el potencial de ahorro eléctrico y económico que poseen los usuarios residenciales mediante la implementación de técnicas de DR que actúen sobre los equipos de aire acondicionado. En [14] se propone un algoritmo de DR distribuido con incentivo en la disminución de la tarifa eléctrica, donde los usuarios se comunican entre sí (mediante dispositivos inteligentes interconectados sobre la red eléctrica) para optimizar el costo de la electricidad. En [15] se considera un modelo automático de DR aplicado al consumo de los aires acondicionados. En [16] se comparan distintas estrategias de control (curtailment, temperature reset, precooling) sobre sistemas HVAC (Heating, Ventilation, and Air Conditioning).

Un enfoque ideado por el equipo “Tecnótic@s”, formado por estudiantes, graduados y docentes/investigadores de la Facultad de Ciencias Exactas y Naturales de la Universidad de Buenos Aires, para la competencia “TECNOX” del año 2016[17], fue construir una aplicación para smartphones de uso masivo cuya finalidad sea detectar cortes de luz inminentes y solicitarles a los usuarios que no utilicen su aire acondicionado durante un intervalo de tiempo.

En contraste con la mayor parte de los estudios realizados en el campo de DR, este enfoque no requiere la implementación de nuevas tecnologías sobre la red eléctrica. Aunque la posibilidad de implementar *Smart Grids* en Argentina ya está siendo estudiada [18], pasará bastante tiempo antes de que esta tecnología sea realidad. La información necesaria para la implementación de este sistema podría ser obtenida a través de la aplicación móvil de la siguiente manera: al ocurrir un corte de luz, los usuarios afectados que se encuentren en su hogar utilizarían la aplicación para enviar la denuncia al organismo correspondiente, la cual se guardaría en un historial de denuncias. Las denuncias realizadas dentro de un mismo intervalo de tiempo pequeño y en ubicaciones cercanas, podrían ser utilizadas para agrupar a los usuarios que se encuentran bajo un mismo tendido eléctrico. Otro dato de importancia es la demanda de cada usuario en particular. Para inferirla, la aplicación realizaría algunas preguntas a cada uno de ellos: la cantidad de equipos de aire acondicionado que poseen en sus viviendas y las frigorías de cada equipo, de manera de aproximar la potencia requerida por cada usuario. Ésto, sumado a otros factores como la temperatura actual y la ubicación de los usuarios -brindada por la aplicación- para saber si se encuentran en sus hogares, daría como resultado el consumo minuto a minuto de ellos. Conociendo el consumo minuto a minuto de cada usuario y los grupos de usuarios que se alimentan de un mismo suministro eléctrico, sería posible estimar dos datos de valor: por un lado, se podría calcular la potencia que le estaba siendo demandada a cada tendido eléctrico al momento del corte, lo que permitiría aproximar su límite; por otro lado, monitoreando el consumo total de cada grupo, se podría conocer cuán cargada se encuentra la red eléctrica en todo momento.

Otro factor diferencial de esta idea de construir una aplicación para smartphones es el incentivo para el usuario, que por lo general es de carácter económico (mediante un bono por participación o una reducción en la tarifa eléctrica). El enfoque sugerido se apoya en la colaboración voluntaria de los usuarios, apelando al sentido de comunidad de las personas, y ofreciendo una aplicación con múltiples ventajas. El atractivo de la aplicación hacia el usuario, además de saber que está ayudando a prevenir posibles cortes eléctricos en el área y en su propia vivienda, vendría dado por algunas funcionalidades adicionales, las cuales le permitirían realizar y administrar denuncias y reclamos, y ver un mapa en tiempo real de los cortes de luz.



Cualquiera sea el enfoque, sería necesario diseñar un algoritmo que se encargue de realizar la selección de los usuarios que deberán disminuir su consumo, tanto al inicio como al momento de realizar cada una de las rotaciones que sean necesarias para no abusar de ningún usuario. El objetivo del presente trabajo es verificar la factibilidad de un algoritmo cuya complejidad temporal y espacial permita su utilización en la práctica. Dicho algoritmo deberá asegurar que la demanda se mantenga siempre por debajo del umbral impuesto por la carga máxima soportada por la red eléctrica. Es deseable minimizar la potencia del suministro no utilizada, lo cual puede suceder al realizar una selección de usuarios más amplia de lo necesario. De no existir este último requerimiento, una solución posible sería que ningún usuario pueda utilizar el aire acondicionado, lo cual resolvería el problema, pero no de manera óptima. También será requisito conservar algún sentido de “justicia” entre los usuarios, balanceando la cantidad de tiempo que cada uno de ellos se encuentra imposibilitado de utilizar su equipo de aire acondicionado.

### 1.1. Escenario inicial

Inicialmente trabajaremos con un escenario simplificado, lo cual nos facilitará enfocarnos en la esencia del problema. Luego iremos aumentando el nivel de complejidad de algunas de las características que han sido simplificadas y analizaremos sus consecuencias, mientras que otras quedarán pendientes para trabajos futuros. El escenario propuesto contará con las siguientes simplificaciones:

- Consumo negociable/no negociable: La demanda representativa de cada usuario estará dada únicamente por la potencia que requiere el/los equipo/s de aire acondicionado que posee cada uno de ellos. Denominaremos a este consumo como “negociable”. La demanda requerida por otras fuentes de consumo como la televisión, la iluminación y la heladera, entre otros, constituirá el consumo “no negociable”. Éste no será tenido en cuenta por el algoritmo de selección debido al bajo impacto que provocan tales fuentes sobre el consumo total, comparado con las fuentes responsables de la refrigeración del ambiente. Además, el apagado de tan esenciales artefactos sería más severo y provocaría rechazo, ya que se han vuelto indispensables en la vida cotidiana de las personas. Si bien existen otros electrodomésticos que podrían entrar en la categoría “negociable” y cuya demanda requerida para funcionar es relativamente alta (por ejemplo: la plancha y el lavarropas), creemos que estos artefactos no suelen ser utilizados en el horario pico de demanda.
- Cantidad de equipos de aire acondicionado por hogar: La disminución de consumo por parte de un usuario será por la totalidad de su consumo negociable. Aunque es posible que los usuarios posean más de un equipo de aire acondicionado, no tendremos en cuenta el caso en el que un usuario seleccionado disminuye su consumo en un porcentaje menor al 100 %, lo cual podría ocurrir al apagar algunos de sus equipos, pero no todos.
- Usuarios participantes: Por el momento asumiremos que todos los usuarios forman parte de la solución, es decir, no existen usuarios que se alimentan de la red eléctrica pero cuyo consumo es imposible disminuir. De no ser así, habría que tener en cuenta el consumo de estos usuarios externos al sistema para restarlo del límite del suministro.

- Usuarios comprometidos: Además de la posibilidad de ser seleccionados, supondremos que los usuarios elegidos para disminuir su consumo durante un período, se comprometerán a hacerlo. De esta manera podemos asegurarnos de que, si la cantidad de usuarios seleccionados para disminuir su consumo es suficiente para no sobrepasar la capacidad máxima, entonces no ocurrirá un corte eléctrico.
- Parámetros fijos: Tanto la cantidad de usuarios como el consumo de cada uno de éstos serán valores estáticos. No se agregarán, eliminarán o modificarán los usuarios a través de los períodos. Lo mismo aplica al límite del suministro eléctrico y a la duración de los períodos.

## 1.2. El problema de la mochila

El “Knapsack Problem” [19] (problema de la mochila) es un problema de optimización combinatoria en el que se desea llenar una mochila, la cual soporta un peso máximo, con cajas que poseen un peso y un valor dados. Una solución al problema es factible si la sumatoria de los pesos de las cajas agregadas a la mochila es menor o igual a su capacidad máxima. Una solución es óptima (pueden existir varias) si el valor añadido de las cajas agregadas es el mayor posible.

El problema de la mochila se asemeja al de seleccionar los usuarios que deberán disminuir su consumo de la siguiente manera: la potencia disponible de la red eléctrica para alimentar a los equipos de aire acondicionado puede verse como el peso máximo soportado por la mochila. Cada usuario representa una caja, siendo su peso la demanda requerida por el usuario para utilizar su acondicionador de aire. Los usuarios que se encuentren dentro de la mochila serán los únicos que podrán tener el aire acondicionado encendido, mientras que los equipos del resto deberán ser apagados con el objetivo de disminuir su consumo. La alternativa sería ver a los usuarios dentro de la mochila como los que no pueden utilizar el aire. Esta opción queda descartada, ya que de esta manera el peso total de las cajas dentro de la mochila debería ser al menos igual a su capacidad o superarla (equivalente a utilizar el suministro eléctrico por debajo de su límite), lo cual no encaja con la esencia del problema al que estamos referenciando, en el cual la capacidad de la mochila no puede ser excedida.

El valor de una caja también será igual a la demanda del usuario que la representa, lo cual implica que el valor de una caja será igual a su peso. Esta característica convierte al problema de la mochila en un caso particular de tal problema, llamado “Value Independent Knapsack Problem” o “Subset-Sum Problem” [19]. En este problema particular, para cumplir con el objetivo de maximizar el valor de la mochila se debe maximizar su peso, lo cual equivale a minimizar el espacio vacío en la mochila. Siendo la mochila una representación del suministro eléctrico, minimizar el espacio sobrante es equivalente minimizar la potencia eléctrica desperdiciada, es decir, la diferencia entre el límite del suministro y la demanda utilizada para alimentar a los acondicionadores de aire activos. En resumen, utilizando la demanda de un usuario como el peso de la caja, intentaremos llenar la mochila lo más posible, lo que se corresponde con aprovechar al máximo el suministro eléctrico.

Desafortunadamente, el problema de la mochila es NP-hard [19], lo cual significa que no es posible obtener un algoritmo que resuelva el problema de forma exacta en tiempo polinomial. Las distintas estrategias para el algoritmo de selección que estudiaremos estarán basadas en heurísticas. Una heurística es un algoritmo que genera soluciones lo suficientemente buenas (aunque no óptimas) en un tiempo razonable.

### 1.3. Períodos y cronogramas

Seleccionado el primer subconjunto de usuarios que deberá disminuir su consumo, el siguiente paso es, luego de un intervalo de tiempo, realizar una rotación de los usuarios. Ésto implica seleccionar un nuevo subconjunto de usuarios que deberán disminuir su consumo, cuya intersección con el conjunto anterior puede o no ser vacía. Llamaremos “período” a cada intervalo de tiempo durante el cual un subconjunto de usuarios debe disminuir su consumo. Una secuencia de períodos sobre el mismo conjunto de usuarios será un “cronograma”.

Diremos que un usuario que ha sido seleccionado para disminuir su consumo durante un período ha sido “apagado” durante ese período, o que le ha sido “asignado un turno” durante el cual no puede utilizar el acondicionador de aire. Por el contrario, un usuario que ha “entrado a la mochila” en un período, es decir, que puede utilizar su equipo de aire acondicionado durante tal período, se encuentra “encendido”.

No es necesario definir la duración de un período de antemano, pero sí vamos a considerar que todos los períodos de un cronograma tendrán la misma duración. Sería deseable que un período sea lo suficientemente corto como para que un hogar se mantenga fresco habiendo apagado el acondicionador de aire durante tal intervalo de tiempo. Por otro lado, la duración de un período no puede ser demasiado reducida, de manera de no requerir el accionar de los usuarios constantemente. Se debe tener en cuenta que no existe un valor que se adecúe a todos los escenarios, ya que dependerá de variables tales como la cantidad de períodos consecutivos que un usuario puede tener sus equipos de aire acondicionado encendidos. Si queremos tener en mente un valor para la duración de un período, podríamos imaginar un tiempo de 15 minutos, lo cual implicaría que la duración de un cronograma de 30 períodos sería de 7 horas y 30 minutos.

También es deseable que un usuario no se encuentre imposibilitado de utilizar su equipo de aire acondicionado durante más de un período consecutivo, aunque ésto puede no ser posible en algunos casos. Buscaremos que la cantidad de períodos que cada usuario fue seleccionado para disminuir su consumo en un cronograma sea lo más equitativa posible. Más formalmente, un cronograma  $c$  posee un nivel de justicia  $k$  si se cumple que:

$$\forall i \in [1, \dots, P] \forall u_1, u_2 \in U : |t_{u_1}^i - t_{u_2}^i| \leq k \quad (1.1)$$

donde  $P$  es la cantidad de períodos en el cronograma;  $U$  es el conjunto de usuarios sobre el cual se aplica el cronograma; y  $t_u^i$  es la cantidad de turnos asignados al usuario  $u$  desde el inicio del cronograma y hasta el período número  $i$  (inclusive). La restricción limita, en cada período, la diferencia entre la mayor y la menor cantidad de turnos asignados a un usuario. Por ejemplo, para que un cronograma posea un nivel de justicia 1, no puede existir un período dentro del cronograma en donde un usuario haya sido asignado  $i$  veces, mientras que otro usuario ha sido asignado  $j$  veces, con  $j > i + 1$ .

Podemos ver a cada período dentro de un cronograma como una instancia nueva del problema de la mochila. Resolver una única instancia de dicho problema no alcanzaría para el armado del cronograma, ya que los usuarios que han sido seleccionados para disminuir su consumo en un período, deberían tener prioridad para entrar a la mochila (utilizar el aire) en los períodos contiguos, por lo que la solución construida para un período probablemente no sea válida para los siguientes. Para armar un cronograma será necesario encontrar una solución al problema de la mochila para cada uno de sus períodos, teniendo en cuenta la prioridad que cada usuario posee al momento de iniciar cada uno.

## 2. DESARROLLO

### 2.1. Algoritmo de selección

El algoritmo que debemos construir para el armado de cronogramas recibirá como entrada los siguientes datos:

1. Usuarios: el conjunto de usuarios con la demanda requerida por cada uno en watts. Esta demanda estará compuesta únicamente por la demanda negociable (aires acondicionados).
2. Límite del suministro: la potencia disponible del suministro, en watts, para el consumo negociable de los usuarios.
3. Períodos: una secuencia de períodos, identificados por sus nombres. El dato importante de este parámetro es su longitud, la cual denota la cantidad de subconjuntos de usuarios que se deben generar para la asignación de turnos.

Tanto la demanda requerida por los usuarios, como el límite del suministro eléctrico, omiten el consumo no negociable. Este consumo no será tenido en cuenta debido a su baja implicancia sobre el consumo total, como ya hemos visto, y porque los artefactos que lo constituyen son imprescindibles para los usuarios. Ésto nos permite abstraernos de tal variable: imaginamos que el consumo no negociable de todos los usuarios ya ha sido substraído de la capacidad del suministro, dejándonos con una potencia menor para abastecer a los equipos de aire acondicionado y usuarios cuya demanda también es la necesaria para alimentar únicamente a sus equipos de aire acondicionado.

La salida esperada -en caso de que la corrida sea exitosa- será un cronograma especificando el conjunto de usuarios que deben disminuir su consumo en cada período. Análogamente, la salida del algoritmo podría especificar cuáles son los usuarios que sí tienen la posibilidad de utilizar el aire acondicionado en cada período, lo cual es indistinto, ya que los conjuntos son complementarios.

La restricción dada por la ecuación 1.1 relaciona a los usuarios de un cronograma según la cantidad de veces que cada uno ha sido seleccionado para disminuir su consumo en distintos períodos. Es por esto que la selección de un conjunto de usuarios para un período no solo afectará al período en cuestión, sino que condicionará a los períodos siguientes. Para cumplir esta restricción de justicia, antes de realizar la selección para un período, el algoritmo deberá calcular la cantidad de veces que un usuario ha sido apagado hasta el momento. Llamemos a esta cantidad la “prioridad” del usuario en un período dado. La prioridad de todos los usuarios será nula inicialmente, pero irá aumentando a medida que los usuarios sean apagados a través de los períodos. Los usuarios que posean un nivel de prioridad mayor en un período, tendrán mayor probabilidad de entrar en la mochila -permanecer encendidos- que otros usuarios con menor prioridad durante el período en cuestión.

El algoritmo 1 muestra el pseudocódigo del procedimiento `BUILDSCHEDULE`, que será el encargado de construir un cronograma utilizando alguna de las estrategias que veremos en las secciones subsiguientes. Al inicio del procedimiento se ordena a los usuarios por

única vez de forma descendente según sus demandas. Luego se realiza una cantidad de iteraciones igual a la longitud de la lista de períodos, obteniendo en cada una el conjunto de usuarios asignados para disminuir el consumo en el período asociado al número de iteración. Cada iteración comienza invocando al procedimiento `DETERMINETARGETUSERS` (líneas 6-8 del algoritmo 1), que divide a los usuarios en dos subconjuntos disjuntos según sus prioridades. El proceso de selección para un período se realizará sobre uno solo de los dos subconjuntos. En caso de que la demanda total del conjunto de usuarios con mayor prioridad no supere el límite del suministro, éstos serán ignorados, y se utilizará alguna estrategia para seleccionar usuarios de entre los que tienen menor prioridad. Los usuarios de baja prioridad que no hayan sido seleccionados integrarán el conjunto de usuarios que no podrán utilizar el aire acondicionado durante el período en cuestión. En el caso contrario, la selección se realizará sobre los usuarios de mayor prioridad, mientras que los de menor prioridad serán asignados automáticamente. Nuevamente, los usuarios seleccionados por la estrategia quedarán libres, mientras que el resto de los usuarios serán asignados.

En el llamado al procedimiento `SELECTUSERSFROM` (líneas 9-10) del algoritmo 1), es donde entrarán en juego las distintas estrategias de selección, asociadas al problema de la mochila. Habiendo resuelto el requerimiento de prioridad, la estrategia puede seleccionar cualquier subconjunto de usuarios posible, agnóstico de sus prioridades. La entrada del procedimiento se compone de dos datos: el conjunto de usuarios sobre el cual se debe realizar la selección y el límite de potencia disponible. Notar que éstos pueden ser distintos a los datos originales que recibe el procedimiento `BUILDSCHEDULE`: el conjunto de usuarios será un subconjunto del total de los usuarios, mientras que la potencia disponible dependerá del primero. Al inicio, todos los usuarios poseen la misma prioridad, por lo que la selección será realizada sobre el total de los usuarios. El procedimiento deberá seleccionar a los usuarios que entran en la mochila, y por lo tanto, pueden utilizar el aire acondicionado. Al resto de los usuarios, que fueron involucrados en el proceso de selección, pero no fueron seleccionados, les será asignado un turno para disminuir su consumo.

La complejidad computacional del algoritmo 1 dependerá de la cantidad de usuarios y de períodos, y de la estrategia utilizada. El ordenamiento inicial de los  $n$  usuarios se encuentra en el orden de  $\mathcal{O}(n \cdot \log(n))$ . Para cada uno de los  $p$  períodos se debe separar a los usuarios por prioridad y calcular la demanda de al menos uno de los conjuntos resultantes, seguido de un llamado al procedimiento `SELECTUSERSFROM`. Si la complejidad de la estrategia utilizada por este procedimiento es  $e$ , la complejidad del armado de un cronograma se encuentra en el orden de  $\mathcal{O}(n \cdot \log(n) + p \cdot (n + e))$ .

## 2.2. Estrategias “High-To-Low” y “Low-To-High”

Una estrategia muy simple consiste en ordenar a los usuarios por demanda y comenzar agregando a la mochila a los usuarios cuya demanda es la mayor, completando con usuarios de menor demanda en caso de que uno más grande no entre cuando el espacio comienza a ser acotado. Llamaremos a esta estrategia “High-To-Low”, representada en forma de pseudocódigo en el algoritmo 2. La complejidad computacional del procedimiento `SELECTUSERSFROM` implementado por esta estrategia es del orden de  $\mathcal{O}(n)$ , por lo que el armado de un cronograma se encontrará en el orden de  $\mathcal{O}(n \cdot \log(n) + p \cdot n)$ .

A modo de ejemplo, observemos el comportamiento de esta estrategia en un escenario al que denominaremos “Escenario A”, en donde el suministro posee un límite de 18 kW y en donde existen 8 usuarios que se alimentan de él. De los 8 usuarios, 3 tienen una

**Algorithm 1** Armado de cronograma

---

```

1: procedure BUILDSCHEDULE(users, powerSupplyLimit, periods, strategy)
2:   orderedUsers  $\leftarrow$  users.sortedDescendentlyByDemand()
3:   priorityByUser  $\leftarrow$  {user: 0 for user in users}
4:   schedule  $\leftarrow$  []
5:   for i := 0 to periods.length() - 1 do
6:     targetUsersForSelection, remainingPower, assignedUsers
7:        $\leftarrow$  DETERMINETARGETUSERS(orderedUsers, priorityByUser,
8:                                   powerSupplyLimit)
9:     selectedUsers  $\leftarrow$  strategy.SELECTUSERSFROM(
10:      targetUsersForSelection, remainingPower)
11:     unselectedUsers  $\leftarrow$  targetUsersForSelection.difference(selectedUsers)
12:     assignedUsers.extend(unselectedUsers)
13:     schedule[i]  $\leftarrow$  assignedUsers
14:     for user in assignedUsers do
15:       priorityByUser[user] + +
16:     end for
17:   end for
18:   return schedule
19: end procedure

20: procedure DETERMINETARGETUSERS(orderedUsers, priorityByUser,  $\triangleright \mathcal{O}(n)$ 
                                   powerSupplyLimit)
21:   lowPriorityUsers  $\leftarrow$  FILTERUSERSOFLOWPRIORITY(orderedUsers,
22:                                                     priorityByUser)
23:   highPriorityUsers  $\leftarrow$  orderedUsers.difference(lowPriorityUsers)
24:   if powerSupplyLimit - highPriorityUsers.demand()  $\geq$  0 then
25:     assignedUsers  $\leftarrow$  []
26:     remainingPower  $\leftarrow$  powerSupplyLimit - highPriorityUsers.demand()
27:     targetUsersForSelection  $\leftarrow$  lowPriorityUsers
28:   else
29:     assignedUsers  $\leftarrow$  lowPriorityUsers
30:     remainingPower  $\leftarrow$  powerSupplyLimit
31:     targetUsersForSelection  $\leftarrow$  highPriorityUsers
32:   end if
33:   return targetUsersForSelection, remainingPower, assignedUsers
34: end procedure

35: procedure FILTERUSERSOFLOWPRIORITY(orderedUsers, priorityByUser)  $\triangleright \mathcal{O}(n)$ 
36:   minimumPriority  $\leftarrow$  min(priorityByUser.values())
37:   lowPriorityUsers  $\leftarrow$  []
38:   for user in orderedUsers do
39:     if priorityByUser[user] == minimumPriority then
40:       lowPriorityUsers.append(user)
41:     end if
42:   end for
43:   return lowPriorityUsers
44: end procedure

```

---

**Algorithm 2** Estrategia “High-To-Low”

---

```

1: procedure SELECTUSERSFROM(orderedUsers, powerSupplyLimit) ▷  $\mathcal{O}(n)$ 
2:   selectedUsers  $\leftarrow []$ 
3:   powerLeft  $\leftarrow$  powerSupplyLimit
4:   for user in orderedUsers do ▷ list is sorted descendently
5:     if user.demand()  $\leq$  powerLeft then
6:       selectedUsers.append(user)
7:       powerLeft  $\leftarrow$  powerLeft  $-$  user.demand()
8:     end if
9:   end for
10:  return selectedUsers
11: end procedure

```

---

demanda de 5 kW, 2 demandan 3 kW y los últimos 3 tan solo 1 kW. La potencia necesaria para abastecer a todos los usuarios con el aire prendido al mismo tiempo es de 24 kW, pero la demanda total no puede superar el límite de 18 kW impuesto por la capacidad del suministro, lo que significa que al menos 6 kW deben ser apagados en cada período. Un esquema del escenario se muestra en la figura 2.1.

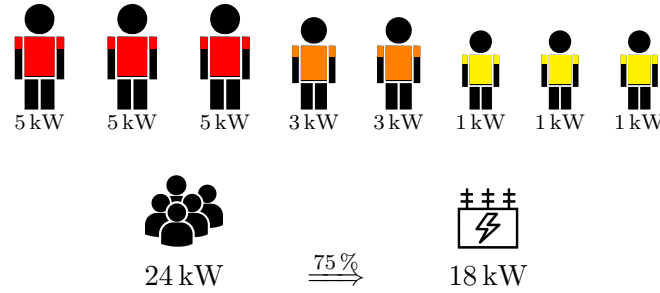


Fig. 2.1: Composición del Escenario A.

La tabla 2.1 muestra una representación de un cronograma adaptado al Escenario A. Cada fila se corresponde con un usuario y cada columna con un período distinto. Marcaremos con un tick (✓) a un usuario en un período si el usuario puede utilizar el aire (entra a la mochila) durante ese período. Los usuarios que sean seleccionados para disminuir su consumo tendrán más prioridad para el siguiente, lo cual denotaremos con una flecha apuntando hacia arriba (↑) antes del comienzo de cada período. Finalmente, debajo de cada período podremos visualizar tanto la utilización como el desperdicio del suministro.

La estrategia “High-To-Low” intentará introducir en la mochila usuarios con la mayor demanda posible, hasta que no entre ninguno más. En el primer período ningún usuario posee mayor prioridad que otro, por lo que se deberá seleccionar de entre la totalidad del conjunto. Todos los usuarios con una demanda de 5 kW juntos suman 15 kW, por lo tanto entran a la mochila y permanecerán prendidos durante este período. Los usuarios que siguen en la escala de mayor demanda son los de 3 kW, que es exactamente el espacio restante en la mochila, de modo que se agregará un único usuario de tal demanda. En la tabla 2.2, a modo de esquema, se muestran los usuarios seleccionados para utilizar el aire (A, B, C, D) durante el primer período. La demanda total para este conjunto de usuarios

Usuario	Demanda	Período											
		1		2		3		4		5		6	
A	5 kW												
B	5 kW												
C	5 kW												
D	3 kW												
E	3 kW												
F	1 kW												
G	1 kW												
H	1 kW												
Utilizado													
Desperdiciado													

Tab. 2.1: Cronograma vacío correspondiente al Escenario A.

es de 18 kW, por lo que en este período el suministro se utiliza al 100 % y el desperdicio es nulo. El resto de los usuarios (E, F, G, H) han sido seleccionados para disminuir su consumo, por lo que su prioridad se ve aumentada para los períodos que siguen.

Usuario	Demanda	Período											
		1		2		3		4		5		6	
A	5 kW	✓											
B	5 kW	✓											
C	5 kW	✓											
D	3 kW	✓											
E	3 kW		↑										
F	1 kW		↑										
G	1 kW		↑										
H	1 kW		↑										
Utilizado		18 kW											
Desperdiciado		0											

Tab. 2.2: Estrategia “High-To-Low”. Escenario A. Primer período.

En el segundo período la prioridad divide a los usuarios en 2 grupos: los que fueron apagados el período anterior (más prioridad) y los que no (menos prioridad). En caso de que la demanda total del grupo de usuarios con mayor prioridad no supere la capacidad de la mochila, estos usuarios serán agregados directamente a ella y luego se deberá seleccionar a los usuarios del grupo con menos prioridad que entrarán. Caso contrario, los usuarios de menor prioridad quedarán fuera de la mochila automáticamente y la selección se realizará sobre el grupo de mayor prioridad. Existe una excepción a la regla en el caso de que el espacio restante en la mochila admita sólo a algunos usuarios de menor prioridad, habiendo otro de mayor prioridad fuera: aunque no sería justo para el usuario de mayor prioridad, se agregará el usuario de menor prioridad para reducir el espacio desperdiciado.

En este período la demanda del grupo de mayor prioridad es de 6 kW, por lo que podemos meterlo en la mochila sin problemas, dejando un espacio restante de 12 kW para ser ocupado por usuarios de menor prioridad. Los primeros en entrar serán, nuevamente, los de mayor demanda. De los 3 usuarios que demandan 5 kW sólo entran 2, dejando



un hueco de 2 kW. Los usuarios que demandan 3 kW no entran y los usuarios con una demanda de 1 kW ya se encuentran en la mochila. Finalmente queda un espacio libre de 2 kW, lo que se traduce en 2 kW de suministro no utilizado durante el período, como se puede observar en la tabla 2.3.

Usuario	Demanda	Período										
		1		2		3		4		5		6
A	5kW	✓		✓								
B	5kW	✓		✓								
C	5kW	✓			↑							
D	3kW	✓			↑							
E	3kW		↑	✓	↑							
F	1kW		↑	✓	↑							
G	1kW		↑	✓	↑							
H	1kW		↑	✓	↑							
Utilizado		18kW		16kW								
Desperdiciado		0		2kW								

Tab. 2.3: Estrategia “High-To-Low”. Escenario A. Segundo período.

En el tercer período sucede algo similar al anterior: los usuarios que ya han sido apagados en alguno de los dos períodos anteriores suman una demanda total de 14 kW, por lo que van directo a la mochila, dejando un espacio de 4 kW. El grupo de usuarios de menor prioridad (los que todavía no han sido apagados) está conformado por 2 usuarios que demandan 5 kW, lo que significa que deberán quedar afuera durante este período, ya que si alguno de ellos es agregado superaría el límite de 18 kW que soporta la red. Por lo tanto en este período finalmente se genera un hueco de 4 kW en la mochila, o 4 kW desperdiciados. Ver tabla 2.4.

Usuario	Demanda	Período										
		1		2		3		4		5		6
A	5kW	✓		✓			↑					
B	5kW	✓		✓			↑					
C	5kW	✓			↑	✓	↑					
D	3kW	✓			↑	✓	↑					
E	3kW		↑	✓	↑	✓	↑					
F	1kW		↑	✓	↑	✓	↑					
G	1kW		↑	✓	↑	✓	↑					
H	1kW		↑	✓	↑	✓	↑					
Utilizado		18kW		16kW		14kW						
Desperdiciado		0		2kW		4kW						

Tab. 2.4: Estrategia “High-To-Low”. Escenario A. Tercer período.

Luego del tercer período todos los usuarios han sido asignados exactamente una vez, por lo que todos poseen el mismo nivel de prioridad. En la tabla 2.5 se ve que la secuencia de asignaciones realizadas por la estrategia durante los primeros 3 períodos se repite periódicamente. El tercer período de cada ciclo es el de mayor desperdicio, utilizando 14 kW

de los 18 kW disponibles (22 % desperdiciado). Cada ciclo posee un desperdicio total de 6 kW, lo cual se traduce en un promedio del 11 % del suministro no utilizado.

El problema de esta estrategia es que, luego de apagar a los usuarios de menor demanda durante los primeros periodos, el concepto de justicia empieza a tener más importancia que la demanda de los usuarios, por lo que llega un período en el que los usuarios de menor demanda que han sido apagados previamente son los primeros en entrar a la mochila. Luego, los únicos usuarios disponibles para llenar el espacio restante serán los de demanda alta, lo cual aumentará la probabilidad de dejar un hueco de tamaño considerable. Esta situación no sucede únicamente en un período, sino que puede darse en los últimos periodos de cada ciclo (seguidilla de periodos donde todos los usuarios han sido apagados al menos una vez).

Usuario	Demanda	Período											
		1		2		3		4		5		6	
A	5 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
B	5 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
C	5 kW	✓			↑	✓	↑	✓	↑		↑↑	✓	↑↑
D	3 kW	✓			↑	✓	↑	✓	↑		↑↑	✓	↑↑
E	3 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
F	1 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
G	1 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
H	1 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
Utilizado		18 kW		16 kW		14 kW		18 kW		16 kW		14 kW	
Desperdiciado		0		2 kW		4 kW		0		2 kW		4 kW	

Tab. 2.5: Estrategia “High-To-Low”. Escenario A. Cronograma completo.

Algo similar ocurre con la estrategia inversa, “Low-To-High”, que comienza agregando a la mochila a los usuarios de menor demanda (algoritmo 3). Existen 2 diferencias entre el algoritmo utilizado por esta estrategia y el que utiliza la estrategia “High-To-Low”: la más evidente es que recorre a los usuarios de forma inversa (línea 4), de manera de iterar primero sobre los de menor demanda; además, en la línea número 9 del algoritmo existe un *break*, ya que en el momento en que la demanda de un usuario supera la potencia restante, es sabido que ningún usuario más entrará. Sin embargo, en el peor caso (al menos todos los usuarios menos el último entren en la mochila), es necesario recorrer la lista entera de usuarios. La complejidad computacional de generar un cronograma utilizando esta estrategia también se encuentra en el orden de  $\mathcal{O}(n \cdot \log(n) + p \cdot n)$ .

En el caso de la estrategia “Low-To-High”, no son los últimos periodos de un ciclo el problema, sino los primeros, ya que al meter en la mochila los usuarios de menor demanda primero, el espacio restante debe ser llenado con usuarios de mayor demanda, al igual que ocurre con la estrategia anterior al término de un ciclo. Siguiendo con el ejemplo anterior, la estrategia “Low-To-High” genera una secuencia de asignaciones invertida con respecto a la generada por la estrategia “High-To-Low”. En la tabla 2.6 vemos que el conjunto de usuarios que ha seleccionado la estrategia “Low-To-High” para disminuir su consumo en el primer período (A, B) es el mismo que el seleccionado por la estrategia “High-To-Low” en su tercero (tabla 2.5). Análogamente, el conjunto de usuarios seleccionado para disminuir su consumo por la estrategia “Low-To-High” en el tercer período (E, F, G, H) es el mismo que el seleccionado por la estrategia “High-To-Low” en el primero. Las dos estrategias

realizan la misma selección de usuarios para el segundo período (C, D).

---

**Algorithm 3** Estrategia “Low-To-High”

---

```

1: procedure SELECTUSERSFROM(orderedUsers, powerSupplyLimit) ▷  $\mathcal{O}(n)$ 
2:   selectedUsers  $\leftarrow []$ 
3:   powerLeft  $\leftarrow$  powerSupplyLimit
4:   for user in orderedUsers.reverse() do ▷ list is sorted descendently
5:     if user.demand()  $\leq$  powerLeft then
6:       selectedUsers.append(user)
7:       powerLeft  $\leftarrow$  powerLeft  $-$  user.demand()
8:     else
9:       break ▷ following users will have higher demands
10:    end if
11:  end for
12:  return selectedUsers
13: end procedure

```

---

De distinta manera, las dos estrategias seleccionan exactamente una vez a cada usuario para disminuir su consumo a lo largo de cada ciclo (3 períodos), resultando en una idéntica baja eficiencia energética.

Usuario	Demanda	Período											
		1		2		3		4		5		6	
A	5 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
B	5 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
C	5 kW	✓			↑	✓	↑	✓	↑		↑↑	✓	↑↑
D	3 kW	✓			↑	✓	↑	✓	↑		↑↑	✓	↑↑
E	3 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
F	1 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
G	1 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
H	1 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
Utilizado		14 kW		16 kW		18 kW		14 kW		16 kW		18 kW	
Desperdiciado		4 kW		2 kW		0		4 kW		2 kW		0	

Tab. 2.6: Estrategia “Low-To-High”. Escenario A. Cronograma completo.

Las estrategias anteriores no solo poseen un comportamiento indeseable, sino que su desempeño puede estar por debajo de una estrategia de asignado aleatorio. Esta estrategia, a la que llamaremos “RandomPick”, no será completamente aleatoria: al ser utilizada dentro del procedimiento BUILD SCHEDULE del algoritmo 1, no realizará la selección sobre la totalidad de los usuarios, sino que al igual que las estrategias anteriores, el conjunto de usuarios involucrados en la selección dependerá de sus prioridades. De no ser así, la estrategia podría armar cronogramas donde se le asigna muchos turnos a algunos usuarios y pocos a otros, infringiendo la restricción de justicia (ecuación 1.1). Para evitar que ésto suceda, los usuarios con prioridad alta son los primeros en entrar a la mochila, sea cual sea la estrategia de asignado. Luego, “RandomPick” seleccionará de forma aleatoria los próximos usuarios que ingresan a la mochila. El algoritmo 4 muestra una posible implementación de esta estrategia, que selecciona usuarios aleatoriamente hasta que no

queda espacio para ninguno más.

---

**Algorithm 4** Estrategia “RandomPick”
 

---

```

1: procedure SELECTUSERSFROM(users, powerSupplyLimit)  $\triangleright \mathcal{O}(n)$ 
2:   selectedUsers  $\leftarrow []$ 
3:   powerLeft  $\leftarrow$  powerSupplyLimit
4:   fittingUsers  $\leftarrow$  users
5:   while powerLeft > 0 and fittingUsers.length() > 0 do
6:     randomUser  $\leftarrow$  random.from(fittingUsers)
7:     if randomUser.demand()  $\leq$  powerLeft then
8:       selectedUsers.append(randomUser)
9:       powerLeft  $\leftarrow$  powerLeft - randomUser.demand()
10:    end if
11:    fittingUsers.remove(randomUser)
12:  end while
13:  return selectedUsers
14: end procedure

```

---

Utilizando el mismo escenario (Escenario A) fueron generados computacionalmente cronogramas de 30 períodos aplicando las distintas estrategias. En el caso de “Random-Pick” se realizaron 10.000 corridas, las cuales dieron como resultado la misma cantidad de cronogramas, de los cuales se realizó un promedio para cada valor de interés y se marcó el desvío estándar de cada uno.

En la figura 2.2 se muestra una comparación de la potencia eléctrica desperdiciada entre las estrategias “High-To-Low”, “Low-To-High” y “RandomPick”. El comportamiento de las dos primeras se corresponde al observado en las tablas generadas manualmente (tabla 2.5 y tabla 2.6). Los ciclos generados por las estrategias “High-To-Low” y “Low-To-High”, los cuales están compuestos por 3 períodos consecutivos con desperdicios de 4 kW, 2 kW y 0, se repiten indefinidamente, dando un promedio de 2 kW desperdiciados por ciclo. Por otro lado, la estrategia aleatoria mantiene el desperdicio energético promedio por debajo de 1 kW en cada período, con un desvío estándar cercano a 1 kW. Aún teniendo en cuenta el desvío estándar, la estrategia “RandomPick” sobrepasa los 2 kW desperdiciados únicamente en el primer período. Calculando el área que se encuentra debajo de cada curva obtenemos el desperdicio total (30 períodos) de cada estrategia en cuanto a la utilización del suministro, el cual se muestra en la figura 2.4(a).

El límite del suministro eléctrico en el Escenario A está relacionado con la demanda total de los usuarios, siendo un 75 % de ésta. Para aumentar el alcance del experimento se volvieron a generar cronogramas sobre el Escenario A utilizando las mismas estrategias, pero asignando distintos valores al límite del suministro. Situamos este parámetro en el rango 40 % - 90 % de la demanda total de los usuarios, con un incremento del 1 % en cada corrida. Esto generó 51 escenarios nuevos, similares al Escenario A, pero con un valor distinto en el límite del suministro. En el caso de la estrategia “RandomPick”, nuevamente se realizaron múltiples corridas (10.000) para cada escenario, de manera de obtener un valor promedio más representativo para cada uno.

En la figura 2.3 se muestra la integral de la potencia desperdiciada total (30 períodos) por cada estrategia en cada variación del Escenario A. La estrategia “RandomPick” generó cronogramas con menor desperdicio a los generados por las otras dos estrategias en la

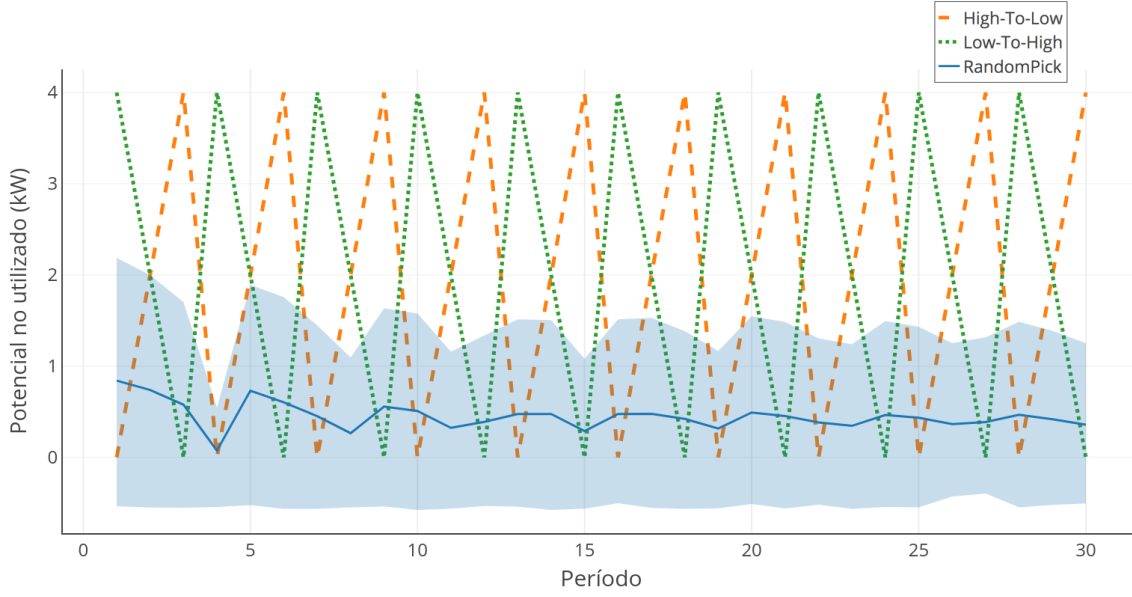


Fig. 2.2: Potencia no utilizada por período para distintas estrategias. Escenario A.

mayoría de los casos, llegando a obtener valores 2 o 3 veces más bajos. En los escenarios en los que el límite del suministro se encuentra en el 58 % o en el 79 % de la demanda total de los usuarios se produce el peor desempeño para las estrategias “High-To-Low” y “Low-To-High”. Estos porcentajes se traducen en valores para el límite del suministro iguales a 13,92 kW (58 %) y 18,96 kW (79 %). Siendo la demanda total de los usuarios de 24 kW, cuando el límite del suministro es de 13,92 kW, es necesario disminuir el consumo en  $24 \text{ kW} - 13,92 \text{ kW} = 10,08 \text{ kW}$ , mientras que un límite del suministro de 18,96 kW implica la necesidad de reducir el consumo en  $24 \text{ kW} - 18,96 \text{ kW} = 5,04 \text{ kW}$ . Notar que el primer valor es apenas más grande que dos usuarios de 5 kW juntos, que suman 10 kW, y el segundo valor es apenas más grande que un solo usuario de 5 kW. Las estrategias “High-To-Low” y “Low-To-High”, que seleccionan usuarios en orden según su demanda, generan situaciones en las que, habiendo disminuido el consumo en 10 kW o 5 kW para algún período, se ven obligados a apagar un usuario extra con demanda de 5 kW o 3 kW para alcanzar la demanda objetivo a disminuir en un período. Éstos períodos generan un gran desperdicio energético, reflejado en los altos valores de potencia no utilizada vistos en la figura. En la figura 2.4(b) se puede observar el promedio y el desvío estándar de las integrales resultantes para cada estrategia.

### 2.3. Estrategias “Alternate by Demand” y “Balance Demand”

Una forma de evitar períodos en los que únicamente sea posible seleccionar usuarios de alta demanda para agregar a la mochila, dando pie a que quede un espacio relativamente grande sin poder llenar, es agregar primero usuarios con demanda intermedia. Para cumplir con esto, una posibilidad es aplicar un algoritmo que apague usuarios alternando entre los de mayor y los de menor demanda. El conjunto de usuarios que será agregado a la mochila puede ser visto como la secuencia de usuarios ordenados por demanda sin sus extremos (con la posible excepción de un conjunto de usuarios ligeros que pueden ser agregados a lo último en caso de existir un hueco suficientemente grande para algunos de ellos). A esta

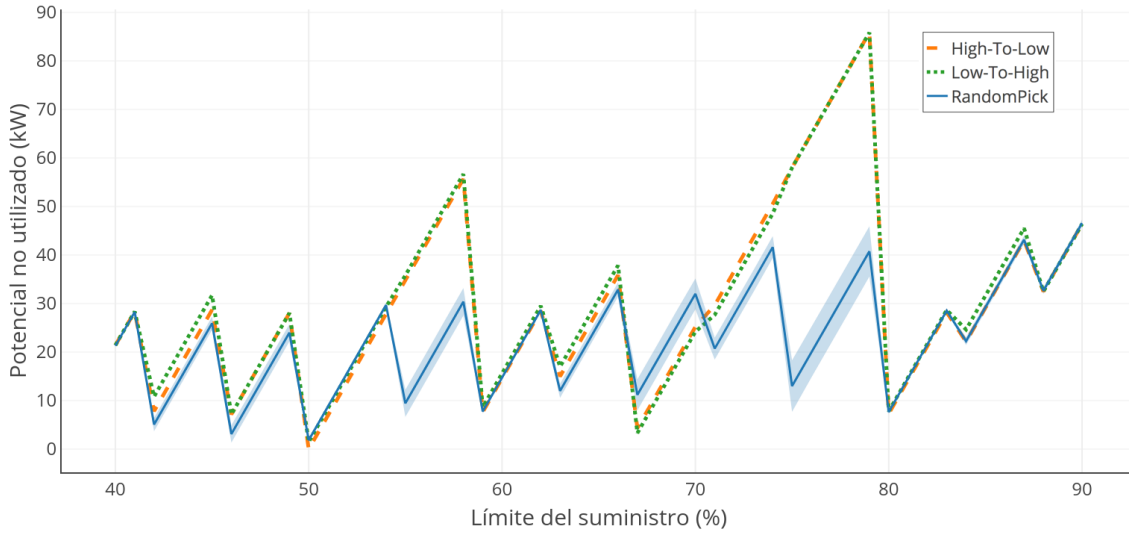


Fig. 2.3: Integral de la potencia no utilizada durante 30 períodos para distintas estrategias variando el límite del suministro. Escenario A.

estrategia la denominaremos “Alternate by Demand”. El funcionamiento de la estrategia se puede observar en el algoritmo 5. A diferencia de las estrategias anteriores, que iban seleccionando a los usuarios que entrarán en la mochila, es decir, que podrán utilizar el aire, esta estrategia comienza con la interpretación de que todos los usuarios se encuentran dentro de la mochila, y debe ir sacando uno por uno hasta que el peso de la mochila no supere su capacidad. En cada iteración quita de la mochila al usuario que se encuentra en alguno de los extremos, alternando entre los dos. La lista de usuarios se encuentra ordenada descendientemente según la demanda de los usuarios, por lo que en la primera iteración el usuario a ser removido es el de mayor demanda. La razón del procedimiento auxiliar llamado CHECKFORUNNECESSARYASSIGNMENTS, utilizado en las líneas 15-16 del algoritmo 5, es la siguiente: al quitar al último usuario de la mochila, si éste posee demanda alta, puede haber quedado un espacio en la mochila en el que quepan uno o más usuarios de menor demanda que han sido quitados previamente y pueden ser devueltos a la mochila, aprovechando mejor su capacidad.

En la tabla 2.7 se muestra el cronograma construido por tal estrategia para el Escenario A. En el primer período la estrategia comienza apagando un usuario de alto consumo, en este caso el usuario A. El consumo que es necesario disminuir en cada período impuesto por este escenario es de 6 kW, por lo que apagar únicamente al usuario A no alcanza. La estrategia se mueve hacia el otro extremo y apaga un usuario de bajo consumo, en este caso el usuario H. El apagado de estos dos usuarios (A, H) es suficiente para no sobrepasar el límite del suministro. En el segundo período las acciones son similares, pero los usuarios A y H tienen mayor prioridad al haber sido apagados el período anterior, por lo que el algoritmo selecciona otro par de usuarios de alta y baja demanda, en este caso los usuarios B y G son los seleccionados para disminuir su consumo. De la misma forma los usuarios C y F son seleccionados en el tercer período, mientras que en el cuarto período la única posibilidad es apagar a los usuarios con demanda intermedia (E, D). Estos 4 períodos forman un ciclo que se repetirá indefinidamente, generando un cronograma cuyo desperdicio eléctrico es nulo.

---

**Algorithm 5** Estrategia “Alternate by Demand”

---

```

1: procedure SELECTUSERSFROM(orderedUsers, powerSupplyLimit) ▷  $\mathcal{O}(n)$ 
2:   selectedUsers  $\leftarrow$  orderedUsers
3:   shouldAssignHighDemandUser  $\leftarrow$  true
4:   powerOverflow  $\leftarrow$  selectedUsers.demand() - powerSupplyLimit
5:   while powerOverflow > 0 do
6:     if shouldAssignHighDemandUser then
7:       userToAssign  $\leftarrow$  selectedUsers.first() ▷ list is sorted descendently
8:     else
9:       userToAssign  $\leftarrow$  selectedUsers.last()
10:    end if
11:    selectedUsers.remove(userToAssign)
12:    powerOverflow  $\leftarrow$  powerOverflow - userToAssign.demand()
13:    shouldAssignHighDemandUser  $\leftarrow$  not shouldAssignHighDemandUser
14:  end while
15:  selectedUsers  $\leftarrow$  CHECKFORUNNECESSARYASSIGNMENTS(
16:    orderedUsers, selectedUsers, powerSupplyLimit)
17:  return selectedUsers
18: end procedure

1: procedure CHECKFORUNNECESSARYASSIGNMENTS(users, selectedUsers, ▷  $\mathcal{O}(n)$ 
   powerSupplyLimit)
2:   powerLeft  $\leftarrow$  powerSupplyLimit - selectedUsers.demand()
3:   for assignedUser in users.difference(selectedUsers) do
4:     if assignedUser.demand()  $\leq$  powerLeft then
5:       selectedUsers.add(assignedUser)
6:       powerLeft  $\leftarrow$  powerLeft - assignedUser.demand()
7:     end if
8:   end for
9:   return selectedUsers
10: end procedure

```

---

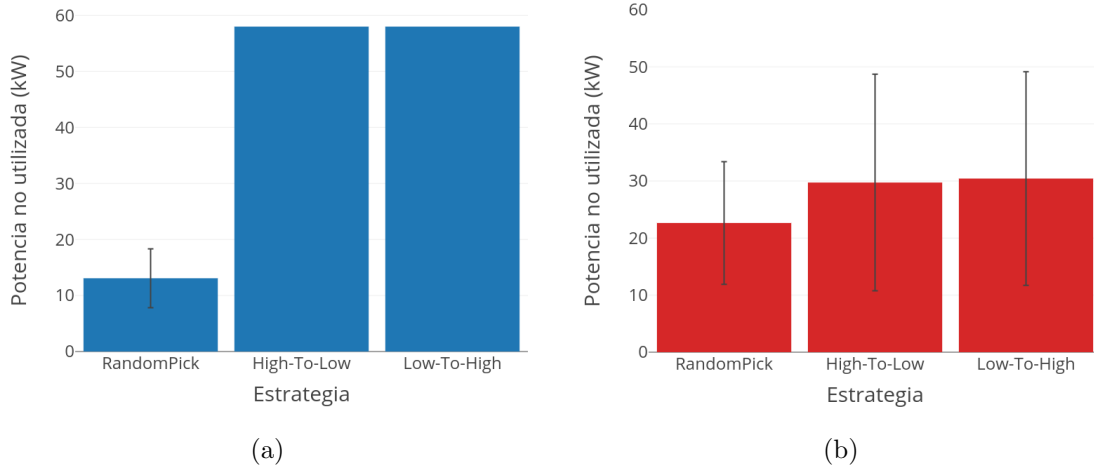


Fig. 2.4: Integral de la potencia no utilizada durante 30 períodos para cada estrategia. Escenario A.  
(a) Límite del suministro eléctrico al 75 % (b) Promedio variando el límite del suministro eléctrico (40 % - 90 %).

Usuario	Demanda	Período											
		1		2		3		4		5		6	
A	5 kW		↑	✓	↑	✓	↑	✓	↑		↑↑	✓	↑↑
B	5 kW	✓			↑	✓	↑	✓	↑	✓	↑		↑↑
C	5 kW	✓		✓			↑	✓	↑	✓	↑	✓	↑
E	3 kW	✓		✓		✓			↑	✓	↑	✓	↑
D	3 kW	✓		✓		✓			↑	✓	↑	✓	↑
F	1 kW	✓		✓			↑	✓	↑	✓	↑	✓	↑
G	1 kW	✓			↑	✓	↑	✓	↑	✓	↑		↑↑
H	1 kW		↑	✓	↑	✓	↑	✓	↑		↑↑	✓	↑↑
Utilizado		18 kW		18 kW		18 kW		18 kW		18 kW		18 kW	
Desperdiciado		0		0		0		0		0		0	

Tab. 2.7: Estrategia “Alternate by Demand”. Escenario A. Cronograma completo.

Aunque en el ejemplo anterior (Escenario A) la cantidad de usuarios con la misma demanda se encuentra balanceada, no será siempre el caso. De hecho, en un escenario real, se espera que el conjunto de usuarios con demanda alta, compuesto por hogares que poseen varias unidades de aire acondicionado y/o unidades de alto consumo necesario para refrigerar ambientes grandes, sea la minoría. Veamos lo que sucede en un escenario (Escenario B) compuesto por 4 usuarios con una demanda de 5 kW y 12 usuarios con demanda de 1,3 kW. El límite del suministro será de 26,7 kW, manteniéndose en un 75 % de la demanda total de los usuarios. La demanda total es de 35,6 kW, por lo que la demanda mínima de los usuarios que deben quedar fuera en cada período resulta en 8,9 kW. Nuevamente, el escenario se encuentra esquematizado en la figura 2.5.

Lo que sucederá con la estrategia que alterna usuarios de alta y baja demanda en el primer período es que intentará apagar un par de cada extremo (2 de 5 kW y 2 de 1,3 kW), ya que un único usuario con demanda de 5 kW no sería suficiente, aún siendo agrupado con 1 o 2 usuarios de 1,3 kW. Pero apagando únicamente al par de 5 kW es suficiente, por lo que los usuarios de 1,3 kW terminarán entrando a la mochila. En este primer período



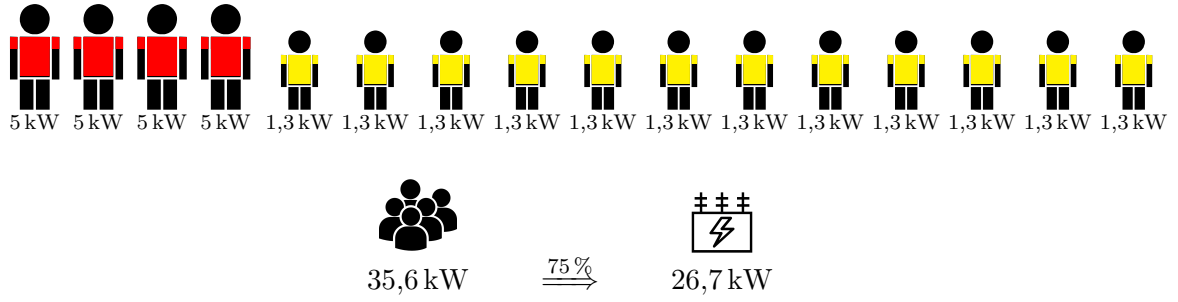


Fig. 2.5: Composición del Escenario B.

se estará utilizando 25,6 kW de potencia de los 26,7 kW disponibles, resultando en un desperdicio de 1,1 kW. Lo mismo sucede en el segundo período al apagar los 2 usuarios con demanda de 5 kW restantes. En el tercer período, los usuarios de 5 kW poseen mayor prioridad que el resto de los usuarios, lo cual impide que vuelvan a ser apagados en este turno. En consecuencia, 7 de los usuarios de 1,3 kW son asignados en el período número 3, sumando una demanda de 9,1 kW y desperdiciando 0,2 kW. En este punto quedan 5 usuarios de 1,3 kW que no han sido apagados en ningún período, por lo que son apagados automáticamente en el período 4. El próximo usuario que selecciona la estrategia para ser apagado en este período será de alta demanda, y como todos los usuarios de 5 kW poseen la misma prioridad, seleccionará cualquiera de ellos. Al apagar un usuario de alta demanda, queda lugar para 2 de los 5 usuarios de 1,3 kW que habían sido dejados afuera al principio debido a su baja prioridad, por lo que 2 de ellos son agregados. Al término del cuarto período, se observa un desnivel en las prioridades de los usuarios: los usuarios A le fueron asignados 2 turnos, mientras que ningún turno le fue asignado a los usuarios E y F, lo cual implica que el cronograma no valida la restricción de justicia de nivel 1 (ecuación 1.1). Ésto sucedió debido al procedimiento CHECKFORUNNECESSARYASSIGNMENTS, que intenta “rescatar” a usuarios que habían sido apagados en un período, pero que al apagar a un usuario de mayor demanda, pueden volver a ser encendidos. Si bien este algoritmo no cumple con la restricción de justicia (ecuación 1.1), no tendría sentido dejar apagados a usuarios que podrían no estarlo.

En los períodos iniciales de este escenario, la estrategia de alternar se comporta como “Low-To-High”, ya que intenta apagar la misma cantidad de usuarios de cada extremo, pero los de menor demanda son despreciables en comparación a los de mayor, por lo que termina apagando únicamente un subconjunto de estos últimos. El problema aquí es el desbalance que existe entre las demandas de los usuarios. Lo que se puede hacer para mitigar este problema es, en lugar de apagar un único usuario de cada extremo, intentar balancear la cantidad de usuarios que se apagan de cada uno. Una posibilidad es elegir un número  $k$  fijo y apagar  $k$  usuarios de baja demanda por cada usuario de alta demanda que apaguemos. Otra forma de balancear los extremos sería apagando usuarios ligeros hasta que la suma de sus demandas alcance la demanda del último usuario de alta demanda que ha sido asignado. A esta estrategia la denominaremos “Balance Demand”. La implementación de esta estrategia, que se puede observar en el algoritmo 6, es muy parecida a la implementación de la estrategia que alterna a los usuarios por demanda (algoritmo 5), con la diferencia de que la decisión de quitar a un usuario de mayor demanda de la mochila depende de los usuarios de menor demanda que hayan sido removidos recientemente.

En las figuras 2.6 y 2.7 se muestran los resultados de utilizar estas nuevas estrategias

Usuario	Demanda	Período							
		1		2		3		4	
A	5 kW		↑	✓	↑	✓	↑		↑↑
B	5 kW		↑	✓	↑	✓	↑	✓	↑
C	5 kW	✓			↑	✓	↑	✓	↑
D	5 kW	✓			↑	✓	↑	✓	↑
E	1,3 kW	✓		✓		✓		✓	
F	1,3 kW	✓		✓		✓		✓	
G	1,3 kW	✓		✓		✓			↑
H	1,3 kW	✓		✓		✓			↑
I	1,3 kW	✓		✓		✓			↑
J	1,3 kW	✓		✓			↑	✓	↑
K	1,3 kW	✓		✓			↑	✓	↑
L	1,3 kW	✓		✓			↑	✓	↑
M	1,3 kW	✓		✓			↑	✓	↑
N	1,3 kW	✓		✓			↑	✓	↑
O	1,3 kW	✓		✓			↑	✓	↑
P	1,3 kW	✓		✓			↑	✓	↑
Utilizado		25,6 kW		25,6 kW		26,5 kW		26,7 kW	
Desperdiciado		1,1 kW		1,1 kW		0,2 kW		0	

Tab. 2.8: Estrategia “Alternate by Demand”. Escenario B. Primer ciclo.

sobre el Escenario B (figura 2.5). Las estrategias de la sección anterior (“High-To-Low” y “Low-To-High”) volvieron a generar cronogramas poco eficientes sobre este escenario. La primera obtuvo un desperdicio promedio de potencia eléctrica un 34 % mayor que “RandomPick”; mientras que para el caso de la segunda estrategia, el desperdicio promedio fue de un 38 % mayor que la estrategia aleatoria. La estrategia que alterna usuarios de alta y baja demanda (“Alternate by Demand”) fue más eficiente que las primeras dos, pero tampoco logró superar a “RandomPick”, al obtener un desperdicio ligeramente mayor que ésta. Por otro lado, la estrategia que balancea la cantidad de usuarios de alta y baja demanda (“Balance Demand”) obtuvo mejores resultados, superando a la estrategia aleatoria, aunque no de manera significativa: en promedio obtuvo un desperdicio un 7,8 % menor (18,04 kW para “Balance Demand” contra 19,45 kW de la aleatoria, en promedio).

## 2.4. Estrategia “Greedy-N-Times”

La idea de realizar un balance entre la cantidad de usuarios de alta y baja demanda que se apagarán en cada período es prometedora en escenarios donde existen pocos usuarios de alta demanda y muchos usuarios de baja demanda, pero su buen desempeño depende de la elección del criterio sobre la cantidad de usuarios de baja demanda que son apagados por cada usuario de alta demanda. Una opción es probar varias alternativas dinámicamente: ordenamos el conjunto de usuarios por demanda y probamos dejar fuera de la mochila distintas cantidades de usuarios de cada extremo de la secuencia, dando lugar a varios subconjuntos posibles de usuarios que irán a la mochila. Un subconjunto será válido si la demanda total de los usuarios incluidos no supera el límite impuesto. Nos quedaremos con el subconjunto que se ajuste mejor a la capacidad de la mochila, es decir, cuya demanda

**Algorithm 6** Estrategia “Balance Demand”

---

```

1: procedure SELECTUSERSFROM(orderedUsers, powerSupplyLimit)  $\triangleright \mathcal{O}(n)$ 
2:   selectedUsers  $\leftarrow$  orderedUsers
3:   powerOverflow  $\leftarrow$  selectedUsers.demand() - powerSupplyLimit
4:   demandToBalance  $\leftarrow$  0
5:   while powerOverflow > 0 do
6:     if demandToBalance  $\leq$  0 then
7:       userToAssign  $\leftarrow$  selectedUsers.first()  $\triangleright$  list is sorted descendently
8:       demandToBalance  $\leftarrow$  userToAssign.demand()
9:     else
10:      userToAssign  $\leftarrow$  selectedUsers.last()
11:      demandToBalance  $\leftarrow$  demandToBalance - userToAssign.demand()
12:    end if
13:    selectedUsers.remove(userToAssign)
14:    powerOverflow  $\leftarrow$  powerOverflow - userToAssign.demand()
15:  end while
16:  selectedUsers  $\leftarrow$  CHECKFORUNNECESSARYASSIGNMENTS(
17:    orderedUsers, selectedUsers, powerSupplyLimit)
18:  return selectedUsers
19: end procedure

```

---

total se encuentre más cerca del límite. En caso de que existan varios subconjuntos válidos con la mayor demanda posible, tomaremos el de mayor longitud.

Para el armado de los subconjuntos válidos nos basaremos en la estrategia “High-To-Low”, la cual recorre la secuencia comenzando por el usuario de mayor demanda y agregando a la mochila a todos los usuarios posibles. Utilizaremos este mecanismo para armar cada subconjunto posible, cambiando la posición en la secuencia por donde comienza la búsqueda. Al igual que la estrategia “High-To-Low”, el primer subconjunto comenzará por el primer elemento de la secuencia, es decir, el usuario de mayor demanda. El segundo subconjunto comenzará por el segundo elemento, etcétera. Un pseudocódigo de la estrategia “Greedy-N-Times” se muestra en el algoritmo 7. El procedimiento SELECTUSERSFROM de la estrategia “High-To-Low” (algoritmo 2), cuya complejidad vimos que era del orden  $\mathcal{O}(n)$ , se invoca una vez por cada usuario para el armado de cada período. Por lo tanto, la complejidad del mismo procedimiento dentro de la estrategia “Greedy-N-Times” se encuentra en el orden de  $\mathcal{O}(n^2)$ . La complejidad computacional de esta estrategia para el armado de un cronograma completo entonces será del orden de  $\mathcal{O}(p \cdot n^2)$ .

En la tabla 2.9 se muestran los subconjuntos posibles para el primer período del Escenario B (figura 2.5). Recordemos que la capacidad del suministro para el Escenario B es de 26,7 kW. El armado del primer subconjunto (a) comienza en el usuario A y agrega a todos los usuarios hasta el I, dejando un resto de 0,2 kW, lo cual no es suficiente para agregar otro usuario. El segundo subconjunto (b) comienza por el usuario B, automáticamente dejando al usuario A afuera. El último usuario que agrega es M, que al ser incluido, se alcanza el límite de 26,7 kW, dejando un desperdicio nulo. El tercer subconjunto (c), que comienza por el usuario C, logra agregar a todos los usuarios que le siguen, dejando un desperdicio de 1,1 kW. El resto de los subconjuntos posibles estarán contenidos en el último subconjunto generado, por lo tanto, son descartados. Una alternativa sería que al

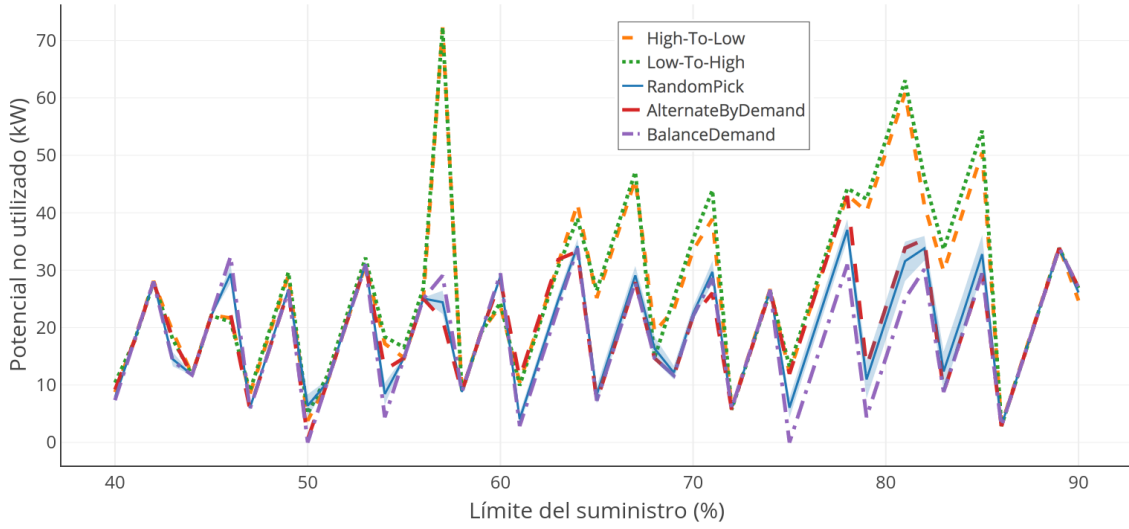


Fig. 2.6: Integral de la potencia no utilizada durante 30 períodos para distintas estrategias variando el límite del suministro. Escenario B.

llegar al final de la lista, se vuelva al principio de ésta, y se intente agregar al subconjunto a los usuarios que se encuentran al comienzo de la lista, es decir, que la lista sea circular. Esta alternativa no ha sido implementada, pero lograría generar una cantidad mayor de subconjuntos posibles. El subconjunto de usuarios elegido por la estrategia para entrar en la mochila en el primer período será el de menor desperdicio, en este caso, el subconjunto (b). Por lo tanto, los usuarios que deberán disminuir su consumo durante el primer período serán A, N, O y P.

En el segundo período, los usuarios  $\{A, N, O, P\}$ , que han sido asignados en el período anterior, poseen más prioridad que el resto de los usuarios. La justicia exige que sean agregados a la mochila obligatoriamente en caso de que sea posible (algoritmo 1). La demanda total de estos 4 usuarios es de 8,9 kW, lo que nos permite agregarlos sin problema. Los 8,9 kW son restados de la capacidad total, dejando  $26,7 \text{ kW} - 8,9 \text{ kW} = 17,8 \text{ kW}$  disponibles para el resto de los usuarios. Los subconjuntos posibles para el segundo período (teniendo en cuenta el nuevo límite de 17,8 kW) se muestran en la tabla 2.10. Los usuarios  $\{A, N, O, P\}$  han sido removidos de la secuencia, ya que la decisión de no apagarlos durante este período ha sido tomada. Nuevamente nos encontramos con 3 posibilidades, siendo la del medio (b) la de menor desperdicio, por lo que es la opción seleccionada. Los usuarios que son apagados en este período son B, K, L y M.

Al conjunto de usuarios que ganaron prioridad al ser apagados en el primer período se une el conjunto de usuarios que fueron apagados en el segundo: el nuevo subconjunto de usuarios con prioridad aumentada es  $\{A, B, K, L, M, N, O, P\}$ . Los usuarios son agregados a la mochila y su demanda es substraída de la capacidad total, obteniendo un nuevo límite de  $26,7 \text{ kW} - 17,8 \text{ kW} = 8,9 \text{ kW}$  para el tercer período.

En la tabla 2.11 vemos los posibles subconjuntos de usuarios a ser encendidos en el período número 3, en adición a los ya agregados. Lo primero que notamos es que el subconjunto de usuarios (a), constituido por los usuarios  $\{C, E, F, G\}$ , no es secuencial. Ésto se da porque luego de incluir al usuario A, cuya demanda es de 5 kW, queda un espacio de 3,9 kW (recordemos que la capacidad límite para este período bajó a 8,9 kW).

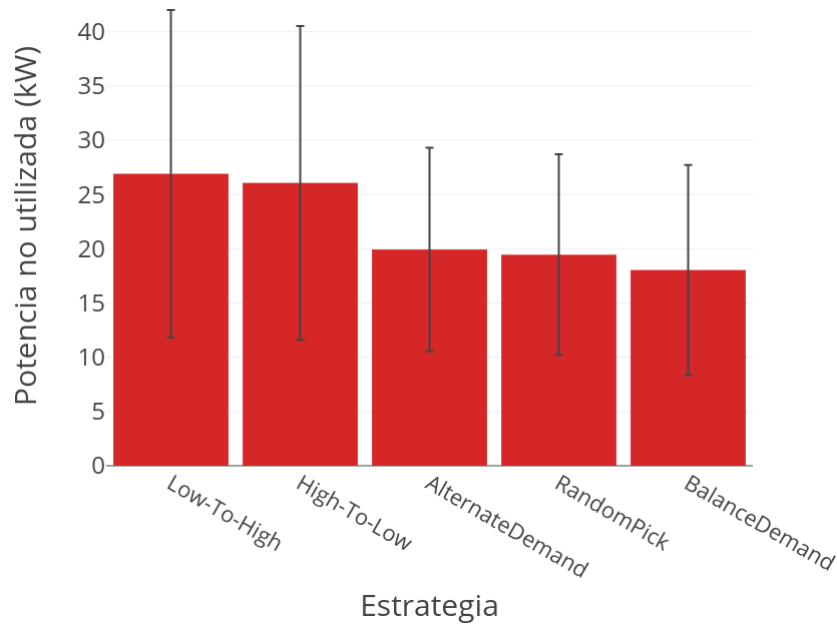


Fig. 2.7: Promedio de las integrales de potencia no utilizada durante 30 períodos para cada estrategia variando el límite del suministro eléctrico (40 % - 90 %). Escenario B.

A (5 kW)	A (5 kW)	A (5 kW)
B (5 kW)	B (5 kW)	B (5 kW)
C (5 kW)	C (5 kW)	C (5 kW)
D (5 kW)	D (5 kW)	D (5 kW)
E (1,3 kW)	E (1,3 kW)	E (1,3 kW)
F (1,3 kW)	F (1,3 kW)	F (1,3 kW)
G (1,3 kW)	G (1,3 kW)	G (1,3 kW)
H (1,3 kW)	H (1,3 kW)	H (1,3 kW)
I (1,3 kW)	I (1,3 kW)	I (1,3 kW)
J (1,3 kW)	J (1,3 kW)	J (1,3 kW)
K (1,3 kW)	K (1,3 kW)	K (1,3 kW)
L (1,3 kW)	L (1,3 kW)	L (1,3 kW)
M (1,3 kW)	M (1,3 kW)	M (1,3 kW)
N (1,3 kW)	N (1,3 kW)	N (1,3 kW)
O (1,3 kW)	O (1,3 kW)	O (1,3 kW)
P (1,3 kW)	P (1,3 kW)	P (1,3 kW)
$\Sigma = 26,5 \text{ kW}$	$\Sigma = 26,7 \text{ kW}$	$\Sigma = 25,6 \text{ kW}$
(a)	(b)	(c)

Tab. 2.9: Estrategia “Greedy-N-Times”. Subconjuntos válidos para el primer período del Escenario B. Límite permitido de demanda: 26,7 kW

El espacio restante de 3,9 kW no es suficiente para incorporar al usuario D, que posee una demanda de 5 kW, pero sí es posible incluir usuarios de 1,3 kW. Los 3,9 kW restantes dan

**Algorithm 7** Estrategia “Greedy-N-Times”

---

```

1: procedure SELECTUSERSFROM(orderedUsers, powerSupplyLimit)  $\triangleright \mathcal{O}(n^2)$ 
2:   bestUserSubset  $\leftarrow$  [ ]
3:   for userSubset in GENERATSUBSETS(orderedUsers, powerSupplyLimit) do
4:     if demandOf(userSubset) > demandOf(bestUserSubset) or
5:       (demandOf(userSubset) == demandOf(bestUserSubset) and
6:         userSubset.length() > bestUserSubset.length()) then
7:       bestUserSubset  $\leftarrow$  userSubset
8:     end if
9:   end for
10:  return bestUserGroup
11: end procedure

12: procedure GENERATESUBSETS(orderedUsers, powerSupplyLimit)  $\triangleright \mathcal{O}(n^2)$ 
13:  targetUsersForSelection  $\leftarrow$  orderedUsers.copy()
14:  while targetUsersForSelection.length() > 0 do
15:    userSubset  $\leftarrow$  HIGHTOLOWSTRATEGY.SELECTUSERSFROM(
16:      targetUsersForSelection, powerSupplyLimit)
17:    yield userSubset
18:    targetUsersForSelection.pop(0)
19:  end while
20: end procedure

```

---

B (5 kW)	B (5 kW)	B (5 kW)
C (5 kW)	C (5 kW)	C (5 kW)
D (5 kW)	D (5 kW)	D (5 kW)
E (1,3 kW)	E (1,3 kW)	E (1,3 kW)
F (1,3 kW)	F (1,3 kW)	F (1,3 kW)
G (1,3 kW)	G (1,3 kW)	G (1,3 kW)
H (1,3 kW)	H (1,3 kW)	H (1,3 kW)
I (1,3 kW)	I (1,3 kW)	I (1,3 kW)
J (1,3 kW)	J (1,3 kW)	J (1,3 kW)
K (1,3 kW)	K (1,3 kW)	K (1,3 kW)
L (1,3 kW)	L (1,3 kW)	L (1,3 kW)
M (1,3 kW)	M (1,3 kW)	M (1,3 kW)
$\Sigma = 17,6 \text{ kW}$	$\Sigma = 17,8 \text{ kW}$	$\Sigma = 16,7 \text{ kW}$
(a)	(b)	(c)

Tab. 2.10: Estrategia “Greedy-N-Times”. Subconjuntos válidos para el segundo período del Escenario B. Límite permitido de demanda: 17,8 kW

lugar para agregar a los 3 usuarios de 1,3 kW que siguen en la secuencia, que son  $\{E, F, G\}$ . Los subconjuntos (a) y (b) poseen desperdicio nulo (a diferencia del subconjunto (c), que queda descartado) e igual cantidad de usuarios. La elección en este caso dependerá de la implementación del algoritmo. Arbitrariamente, nos quedaremos con el subconjunto (b),

al igual que en los períodos anteriores.

C (5 kW)	C (5 kW)	C (5 kW)
D (5 kW)	D (5 kW)	D (5 kW)
E (1,3 kW)	E (1,3 kW)	E (1,3 kW)
F (1,3 kW)	F (1,3 kW)	F (1,3 kW)
G (1,3 kW)	G (1,3 kW)	G (1,3 kW)
H (1,3 kW)	H (1,3 kW)	H (1,3 kW)
I (1,3 kW)	I (1,3 kW)	I (1,3 kW)
J (1,3 kW)	J (1,3 kW)	J (1,3 kW)
$\sum = 8,9 \text{ kW}$	$\sum = 8,9 \text{ kW}$	$\sum = 7,8 \text{ kW}$
(a)	(b)	(c)

Tab. 2.11: Estrategia “Greedy-N-Times”. Subconjuntos válidos para el tercer período del Escenario B. Límite permitido de demanda: 8,9 kW

Hacia el cuarto período, todos los usuarios menos 4 han sido asignados exactamente una vez. Los usuarios que se mantuvieron encendidos durante los 3 primeros períodos son  $\{D, E, F, G\}$ , los cuales suman una demanda de 8,9 kW, el valor exacto para el consumo que necesitamos disminuir en cada período. La única posibilidad para este período es apagar a tales usuarios, de lo contrario, estaríamos generando un cronograma injusto.

En la tabla 2.12 podemos observar el primer ciclo de un cronograma armado por la estrategia “Greedy-N-Times” en el Escenario B. Este ciclo se extiende durante 4 períodos y se repite, ya que el período luego de un ciclo encontrará a la prioridad de los usuarios nivelada, al igual que en el inicio del cronograma.

Comparamos la nueva estrategia con “Balance Demand” y “RandomPick” utilizando el Escenario B. En la figura 2.8 se muestra, para cada estrategia, la potencia desperdiciada para cada límite del suministro, variándolo entre 40 y 90 %. En la figura 2.9 se puede observar el promedio y el desvío estándar de las integrales de cada estrategia.

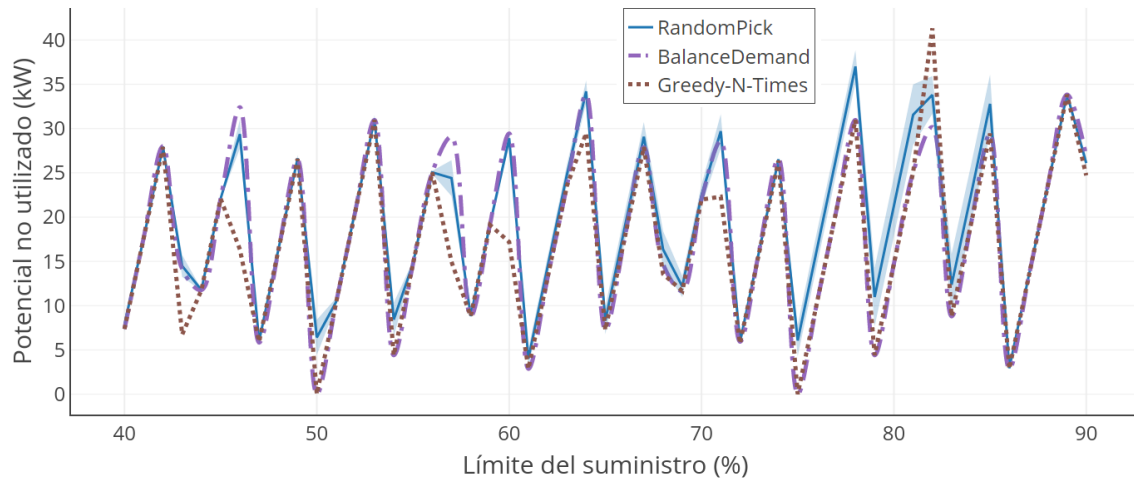


Fig. 2.8: Integral de la potencia no utilizada durante 30 períodos para distintas estrategias variando el límite del suministro. Escenario B.

Usuario	Demanda	Período							
		1		2		3		4	
A	5 kW		↑	✓	↑	✓	↑	✓	↑
B	5 kW	✓			↑	✓	↑	✓	↑
C	5 kW	✓		✓			↑	✓	↑
D	5 kW	✓		✓		✓			↑
E	1,3 kW	✓		✓		✓			↑
F	1,3 kW	✓		✓		✓			↑
G	1,3 kW	✓		✓		✓			↑
H	1,3 kW	✓		✓			↑	✓	↑
I	1,3 kW	✓		✓			↑	✓	↑
J	1,3 kW	✓		✓			↑	✓	↑
K	1,3 kW	✓			↑	✓	↑	✓	↑
L	1,3 kW	✓			↑	✓	↑	✓	↑
M	1,3 kW	✓			↑	✓	↑	✓	↑
N	1,3 kW		↑	✓	↑	✓	↑	✓	↑
O	1,3 kW		↑	✓	↑	✓	↑	✓	↑
P	1,3 kW		↑	✓	↑	✓	↑	✓	↑
Utilizado		26,7 kW		26,7 kW		26,7 kW		26,7 kW	
Desperdiciado		0		0		0		0	

Tab. 2.12: Estrategia “Greedy-N-Times”. Escenario B. Primer ciclo.

La estrategia “Greedy-N-Times” obtuvo un desperdicio promedio un 6 % menor que “Balance Demand” y un 14 % menor que “RandomPick”. A pesar de que la nueva estrategia fue la ganadora, su puntaje no difiere significativamente de la estrategia aleatoria. Esto se debe a que el Escenario B, en el cual realizamos el experimento, tiene tan solo 2 tipos de usuario distintos (5 kW y 1,3 kW) y 16 usuarios en total, lo que no permite a cada estrategia destacarse por sus elecciones. Por ejemplo, cuando el límite del suministro en el Escenario B es de 26,7 kW, las opciones para un período son 3: dejar fuera a 2 usuarios de 5 kW, dejar fuera a 1 usuario de 5 kW y a 3 usuarios de 1,3 kW, o dejar fuera a 7 usuarios de 1,3 kW.

Veamos entonces el resultado de correr las distintas estrategias en un escenario con más usuarios: el Escenario C, que está compuesto por 30 usuarios y 3 valores para sus demandas, como se puede observar en la figura 2.10. En esta oportunidad, no realizaremos un primer experimento con un límite del suministro específico, sino que pasaremos directo a realizar el experimento en el cual variamos este valor entre el 40 % y el 90 % de la demanda total. En la figura 2.11 se puede observar la potencia desperdiciada para cada límite del suministro. La estrategia “Greedy-N-Times” obtuvo un mejor puntaje en casi todos los escenarios. La estrategia “Balance Demand” fue apenas superior únicamente cuando el límite del suministro se situó en el 63 % de la demanda total, mientras que hubo un empate en el 51 % y en el 88 %. La potencia eléctrica desperdiciada promedio de cada estrategia, junto con el desvío estándar, se muestra en la figura 2.12. La estrategia “Greedy-N-Times” generó cronogramas con una potencia desperdiciada promedio menor a la mitad que el desperdiciado por los cronogramas de las otras estrategias. También es interesante observar que la estrategia “Balance Demand” tuvo un peor desempeño promedio que la estrategia aleatoria en este escenario.



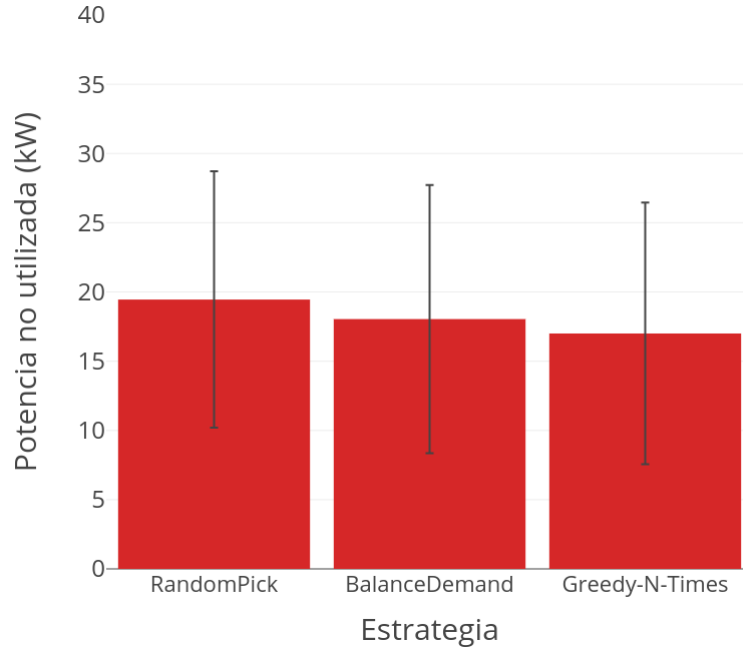


Fig. 2.9: Promedio de las integrales de la potencia no utilizada durante 30 períodos para cada estrategia variando el límite del suministro eléctrico (40 % - 90 %). Escenario B.

A pesar de obtener buenos resultados en la mayoría de los casos, la estrategia “Greedy-N-Times” no es infalible: imaginemos un escenario en el que la elección de un subconjunto de usuarios para el primer período, a pesar de ser la de menor desperdicio, desencadene en un desperdicio alto para los períodos subsiguientes, debido a que la prioridad de los usuarios fuerza a la estrategia a realizar tales elecciones. En estos casos, la estrategia puede fallar, es decir, generar un cronograma con un desperdicio mayor al que pudo haber generado una estrategia aleatoria. Para ejemplificar la situación, miremos nuevamente el Escenario B, cuya composición se muestra en la figura 2.5. El escenario cuenta con 4 usuarios que demandan 5 kW y 12 usuarios que demandan 1,3 kW. Supongamos que el límite del suministro para este escenario fuese de 20 kW. La estrategia “Greedy-N-Times” seleccionaría a los 4 usuarios de 5 kW como los que pueden utilizar el aire acondicionado durante el primer período, ya que el desperdicio de esta elección es nulo. El problema viene en el período siguiente: los usuarios de 1,3 kW, que suman 15,6 kW totales, pasarían a poseer mayor prioridad que los de 5 kW en el segundo período, dejando al algoritmo sin otra alternativa que agregarlos a todos a la mochila para que puedan utilizar el aire acondicionado. La capacidad restante de  $20 \text{ kW} - 15,6 \text{ kW} = 4,4 \text{ kW}$ , no alcanzaría para agregar a un usuario de 5 kW, desenvocando en un desperdicio de 4,4 kW. Peor aún, este ciclo que comprende dos períodos se repetiría sucesivamente a lo largo del cronograma.

## 2.5. Estrategias de período múltiple

Las estrategias anteriores buscaban optimizar un único período a la vez. Pero como vimos, los períodos se encuentran relacionados mediante la prioridad de los usuarios, ya que los usuarios apagados en un período condicionan al conjunto de usuarios que pueden ser seleccionados en los períodos siguientes. Así, para optimizar el cronograma, utilizaremos

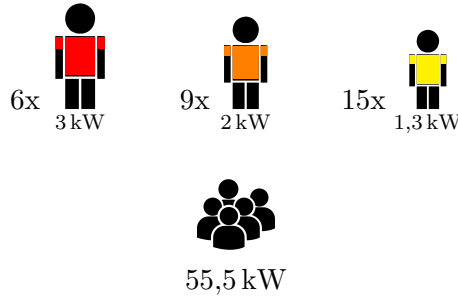


Fig. 2.10: Composición del Escenario C.

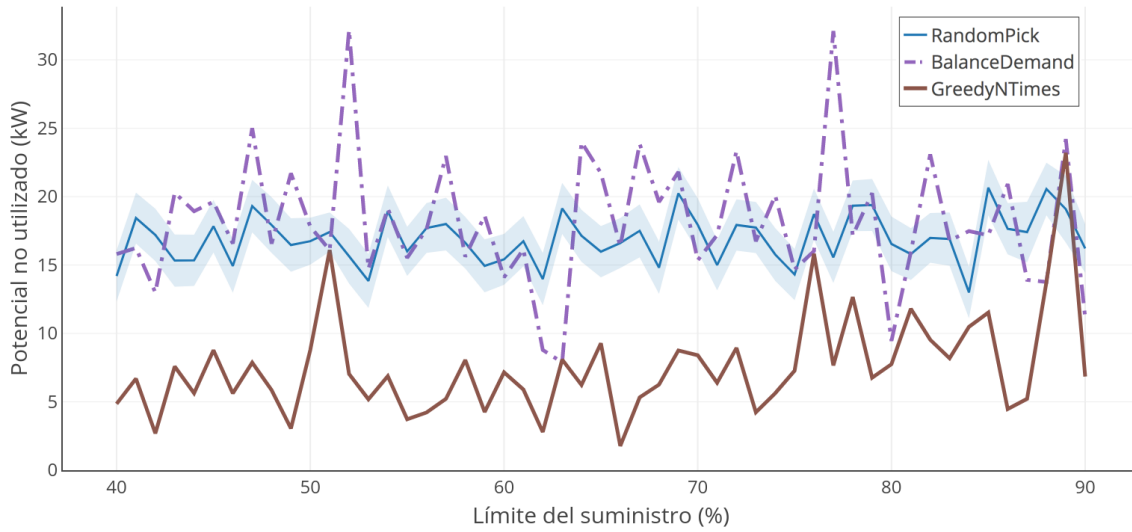


Fig. 2.11: Integral de la potencia no utilizada durante 30 períodos para distintas estrategias variando el límite del suministro. Escenario C.

una estrategia que tome decisiones teniendo en cuenta múltiples períodos a la vez. Una estrategia que trabaja con un período a la vez, sin mirar a futuro, es conveniente en el caso en el que el escenario sea propenso a cambios, por ejemplo usuarios que se suman o restan, o alteraciones en sus demandas, o que por alguna razón se vea modificado el límite del suministro. En dichas circunstancias, lo más razonable es optimizar los períodos más cercanos en el tiempo, ya que en caso de que ocurran tales cambios en el escenario, se deberá adaptar el cronograma creado previamente y generar otro que modele el nuevo escenario. En cambio, en un escenario estático, es posible calcular de antemano múltiples períodos (incluso un cronograma que abarque un día entero), intentando optimizar el cronograma de forma global, y no período a período. En este caso puede suceder que un período anterior a otro sea menos eficiente, lo cual sería aceptable si genera la posibilidad de mejorar la eficiencia global del cronograma.

En un período cada usuario puede entrar o no en la mochila, es decir, puede estar prendido o apagado dependiendo si se decide asignarle un turno o no. Si  $n$  es la cantidad total de usuarios, la cantidad de combinaciones posibles en un único período es de  $2^n$ . En el caso de querer calcular las combinaciones posibles para más de un período, se debe considerar a cada usuario adentro y afuera de la mochila que representa a cada uno de los períodos. La cantidad de combinaciones asciende a  $2^{n \cdot p}$ , donde  $p$  es la cantidad de

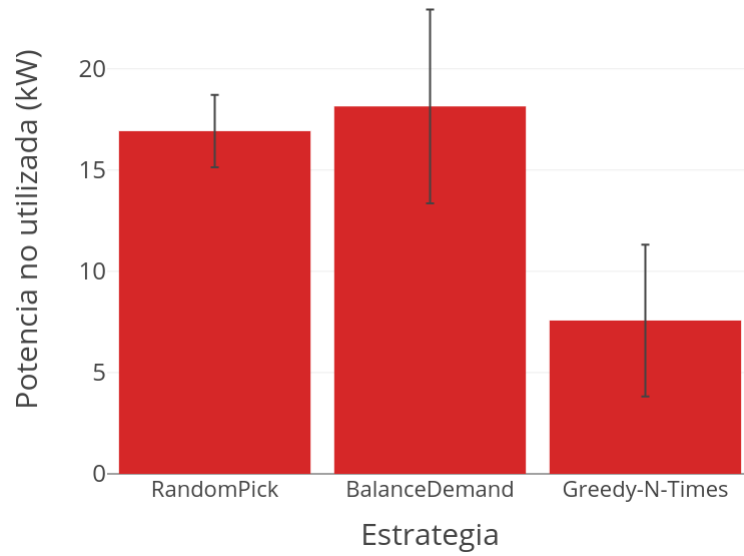


Fig. 2.12: Promedio de las integrales de la potencia no utilizada durante 30 períodos para cada estrategia variando el límite del suministro eléctrico (40 % - 90 %). Escenario C.

períodos. En realidad, esta cota es grosera, ya que al realizar la selección para un período, la prioridad de los usuarios recorta el espacio de búsqueda hasta el final del ciclo. Aún así, utilizar un algoritmo de fuerza bruta ya era inviable para una estrategia que resolviera de a un período a la vez, más aún lo será si necesitamos abarcar múltiples períodos.

### 2.5.1. “Greedy-N-Times” con múltiples períodos

La estrategia de período único “Greedy-N-Times” analizada en la sección anterior generaba, para cada período, distintos subconjuntos de usuarios posibles para ser seleccionados. El subconjunto de usuarios elegido por la estrategia sería el de mayor demanda total, implicando un mejor aprovechamiento del suministro eléctrico, y en caso de empate, el de mayor longitud, implicando una menor cantidad de usuarios apagados (algoritmo 7). Utilizando esta idea, podemos calcular los distintos subconjuntos de usuarios para cada período y combinarlos entre ellos, de manera de encontrar la combinación de subconjuntos que maximice el uso del suministro de forma global. El problema es que no cualquier combinación será válida, ya que necesitamos que la prioridad de los usuarios se encuentre balanceada. Ésto descarta la posibilidad de calcular todos los posibles subconjuntos al inicio y luego combinarlos entre ellos. Lo que sí es posible hacer, es ir fijando un subconjunto específico en cada período y, sujeto a estos subconjuntos anteriores, calcular los subconjuntos posibles para el siguiente, hasta llegar al último, para luego quedarse con la mejor combinación.

El algoritmo 8 muestra un pseudocódigo que representa el funcionamiento de la estrategia, a la que denominaremos “Multiple Greedy-N-Times”. La idea principal del algoritmo es construir un árbol en el cual cada nodo contenga un subconjunto de usuarios con potencial para ser seleccionado en el período correspondiente a la altura del nodo en el árbol. Cada nodo del árbol debe ser generado teniendo en cuenta la prioridad de cada usuario luego de aplicar al cronograma los subconjuntos de usuarios dados por los ancestros de dicho nodo. Los caminos entre la raíz del árbol y sus hojas serán posibles combinaciones

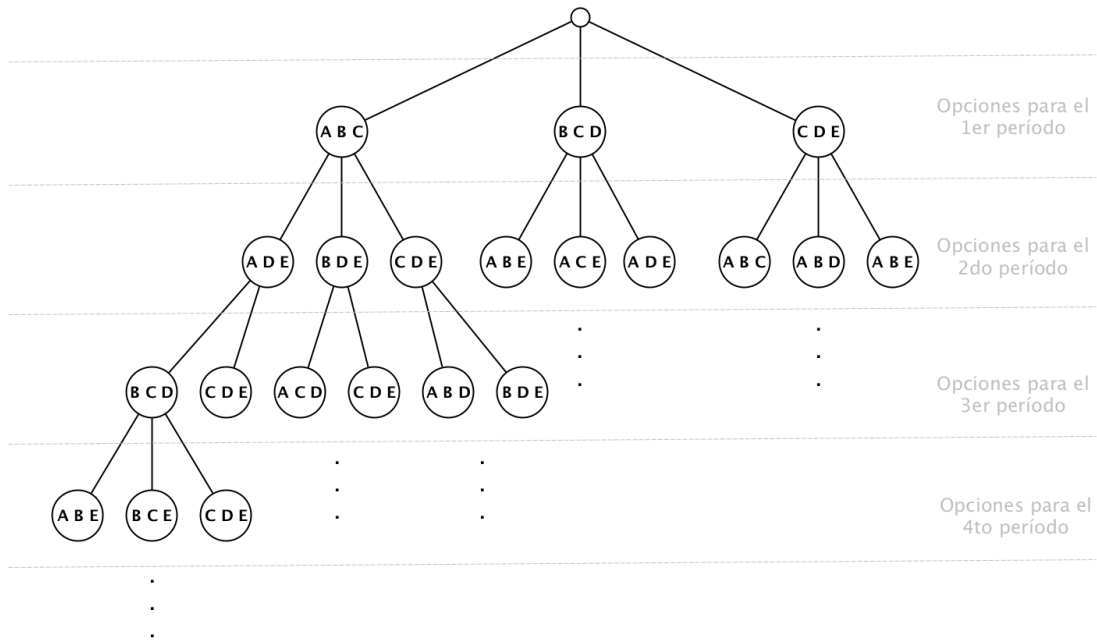


Fig. 2.13: Árbol de combinaciones generado por la estrategia “Multiple Greedy-N-Times” para un escenario que consta de 5 usuarios (A, B, C, D y E) y un límite del suministro que alcanza para 3 de ellos. En cada nodo figuran los usuarios que se encontrarían encendidos en el período correspondiente al nivel del nodo.

de subconjuntos de usuarios, donde cada subconjunto se corresponderá con un período del cronograma. El resultado del algoritmo será una secuencia de subconjuntos de usuarios, siendo el subconjunto en la posición  $i$  el que contiene a los usuarios que podrán utilizar sus equipos de aire acondicionado durante el período número  $i$ . La combinación elegida por el algoritmo será la de mayor energía demandada por los usuarios de todos los subconjuntos que componen la combinación, lo cual equivale al cronograma con menor desperdicio energético global.

En la figura 2.13 se puede observar el árbol que construiría la estrategia en un escenario con 5 usuarios (A, B, C, D y E) y un límite del suministro que alcanza para abastecer a cualquier combinación de 3 usuarios, lo cual implica que 2 usuarios deben quedar fuera de la mochila (no utilizar el aire acondicionado) en cada período. Cada nivel del árbol (excluyendo al nodo raíz) muestra los posibles subconjuntos de usuarios a ser seleccionados para entrar en la mochila (los que sí podrán utilizar el aire acondicionado) en el período correspondiente al nivel ( $1^{er}$  nivel  $\Rightarrow 1^{er}$  período). Por ejemplo, para el primer período los subconjuntos que pueden ser seleccionados son 3: {A, B, C}, {B, C, D} y {C, D, E}. Esta lista de subconjuntos no es extensiva, sino que son los subconjuntos generados por la estrategia “Greedy-N-Times” (algoritmo 7). La elección de un subconjunto de usuarios para el primer período modificará la prioridad de los usuarios (aumentando la prioridad de los usuarios a los que les sea asignado un turno), por lo que dependiendo de esta elección, habrá distintas posibilidades para el segundo período. Ésto genera un abanico de combinaciones, cuya longitud se multiplica en cada período. La cantidad de combinaciones posibles para el primer período es de 3, para los dos primeros períodos es de 9, para los tres primeros es de 18, luego de 54, etcétera.

Para armar el árbol de combinaciones debemos realizar una cantidad de iteraciones igual a la cantidad de períodos requeridos para armar el cronograma. En cada iteración se deben recorrer todas las hojas del árbol, ya que los caminos entre la raíz y una hoja representan todas las combinaciones halladas para la secuencia de períodos previos al actual. Para cada rama se pueden llegar a generar hasta  $n$  nodos más, correspondientes a los  $n$  posibles subconjuntos generados por la estrategia “Greedy-N-Times”, siendo  $n$  la cantidad de usuarios en el escenario. Ésto deriva en un crecimiento exponencial en la cantidad de nodos del árbol en función de su altura. La iteración número  $i$  se encontrará, en el peor caso, con una cantidad de combinaciones igual a  $n^{i-1}$ . La iteración de mayor complejidad será la última, ya que deberá recorrer la mayor cantidad de combinaciones, debiendo ejecutar el algoritmo “Greedy-N-Times” (cuya complejidad computacional está actodada por  $n^2$ ) sobre cada una. Si  $p$  es la cantidad de períodos que debe contener el cronograma generado, la complejidad de esta última iteración será  $n^{p-1} \cdot n^2 = n^{p+1}$ . Dado que se necesitan  $p$  iteraciones para construir el árbol, la complejidad del algoritmo se encuentra en el orden de  $\mathcal{O}(p \cdot n^{p+1})$ .

Para probar la estrategia vamos a utilizar el escenario C, esquematizado en la figura 2.10. Este escenario está formado por 30 usuarios, es decir,  $n = 30$ . Intentar armar un cronograma de 30 períodos ( $p = 30$ ) en una única corrida de la estrategia sería computacionalmente imposible considerando que, en este caso, la complejidad estaría dada por  $30 \cdot 30^{31}$ . Lo que haremos será partir el cronograma en porciones (conjuntos de períodos contiguos) y completarlas una por una, lo cual sería similar a lo que veníamos haciendo en las estrategias de período único, pero resolviendo de a más de un solo período. Llamemos a la longitud de una porción de períodos  $p'$ . Utilizar  $p' = 1$  en la estrategia de período múltiple sería lo mismo que utilizar la estrategia “Greedy-N-Times” de período único, mientras que un  $p' = 30$  equivaldría a calcular el cronograma entero. En este escenario, encontramos que un valor de  $p' > 5$  se vuelve impráctico, por lo que utilizaremos  $p' = 5$ . Este valor significa que la estrategia intentará optimizar los primeros 5 períodos juntos, luego lo hará con el período número 6 hasta el 10, y seguirá así hasta completar el cronograma de 30 períodos.

En la figura 2.14 vemos el resultado de comparar la nueva estrategia de período múltiple con la estrategia “Greedy-N-Times” en el escenario C, utilizando distintos límites para el suministro. Aunque en algunos casos la mejora respecto a la estrategia de período único llegó a sobrepasar el 40 % (por ejemplo para el límite del suministro 78 y 84), no se observa una mejora importante y constante a lo largo de los distintos valores para la capacidad del suministro.

### 2.5.2. Búsqueda local sobre cronogramas

Otra alternativa es mejorar un cronograma ya existente, generado por alguna de las estrategias de selección vistas anteriormente. Podemos ver a tal cronograma como un conjunto de mochilas con usuarios dentro, donde cada mochila representa a un período del cronograma y los usuarios dentro de ella son los que pueden utilizar el aire acondicionado en dicho período. Ninguna mochila se encuentra excedida en peso, pero tampoco permite más usuarios (caso contrario, el usuario habría sido ingresado a la mochila). Por lo tanto, el espacio libre en cada mochila será un valor mayor o igual a cero y menor a la demanda del usuario con demanda mínima. La idea es intercambiar usuarios entre las distintas mochilas, con el objetivo de posibilitar el agregado de 1 o más usuarios a alguna de las mochilas afectadas. El término “búsqueda local” refiere a la técnica que, mediante múlti-

**Algorithm 8** Estrategia “Multiple Greedy-N-Times”

---

```

1: procedure BUILDSCHEDULE(users, powerSupplyLimit, periods)      ▷  $\mathcal{O}(p' \cdot n^{p'+1})$ 
2:   orderedUsers  $\leftarrow$  users.sortedDescendentlyByDemand()
3:   combinationTree  $\leftarrow$  Tree()
4:   combinationTree.setRoot([])
5:   for period := 1 to periods.length() do
6:     for combination in combinationTree.paths() do
7:       priorityByUser  $\leftarrow$  CALCULATEUSERPRIORITIES(users, combination)
8:       targetUsersForSelection, remainingPower, assignedUsers
9:          $\leftarrow$  DETERMINETARGETUSERS(orderedUsers, priorityByUser,
10:                                     powerSupplyLimit)
11:       userSubsets  $\leftarrow$  list(GREEDYNTIMESSTRATEGY.GENERATESUBSETS(
12:         targetUsersForSelection, remainingPower)
13:       )
14:       for userSubset in userSubsets do
15:         unselectedUsers  $\leftarrow$  targetUsersForSelection.difference(userSubset)
16:         combination.leaf().addChild(assignedUsers.union(unselectedUsers))
17:       end for
18:     end for
19:   end for
20:   return combinationTree.pathWithMinDemand()
21: end procedure

22: procedure CALCULATEUSERPRIORITIES(users, combination)
23:   priorityByUser  $\leftarrow$  {user: 0 for user in users}
24:   for userSubset in combination do
25:     for user in userSubset do
26:       priorityByUser[user] ++
27:     end for
28:   end for
29:   return priorityByUser
30: end procedure

```

---

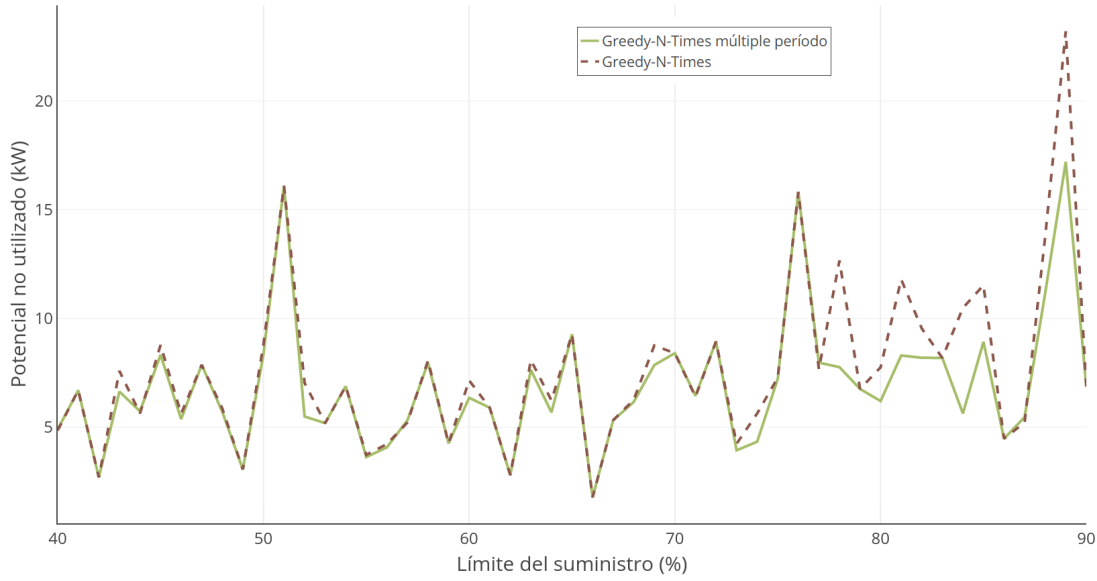


Fig. 2.14: Integral de la potencia no utilizada durante 30 períodos para “Greedy-N-Times” de período único y múltiple ( $p' = 5$ ) variando el límite del suministro. Escenario C.

ples iteraciones, busca una solución que se acerque a la óptima, siempre moviéndose entre soluciones vecinas. Dependiendo del estado inicial del cronograma, existe la posibilidad de encontrar una solución óptima, como también es posible estancarse en un máximo local.

Una iteración en la búsqueda local estará dividida en 3 etapas: el primer paso consiste en realizar el intercambio entre un usuario de una mochila con un usuario de otra. La forma de llevar esto a cabo es buscar un par de períodos y un par de usuarios que sean complementarios con respecto a tales períodos (un usuario se encuentra prendido en el período que el otro se encuentra apagado y viceversa). Una vez encontrada esta combinación de períodos y usuarios, se cambia el estado de cada usuario en cada uno de los dos períodos. Esta acción no debe causar un desborde de potencia utilizada en ninguno de los dos períodos. Aunque el intercambio de usuarios no impacta sobre el desperdicio del cronograma ni sobre la prioridad de los usuarios, abre la posibilidad de prender usuarios en alguno de los períodos afectados. Como segundo paso, se intenta agregar usuarios a la mochila que ahora posee más espacio libre gracias al paso anterior. Si el espacio disponible no diera lugar a ningún usuario, se revierte el intercambio y se continúa ciclando en busca de una nueva combinación de períodos y usuarios con potencial para ser intercambiados. En caso de que al menos un usuario haya sido encendido, se continúa con el tercer y último paso, el cual consiste en realizar un chequeo sobre cada usuario para detectar si su prioridad permite quitarlo de alguna mochila, dejando lugar a otro usuario con mayor demanda, que pueda ajustarse mejor a la capacidad máxima de tal mochila y así reducir el desperdicio.

En la figura 2.15 se encuentra esquematizado un ejemplo minimal de los dos primeros pasos de una iteración, en un escenario que consta de 5 usuarios con demandas de 6 kW, 5 kW, 4 kW, 3 kW y 2 kW, y donde el límite del suministro (capacidad de una mochila) es de 10 kW. En la primera imagen (figura 2.15(a)) vemos un par de mochilas, que representan a 2 períodos cualesquiera de un posible cronograma generado para este escenario. Dentro de cada mochila existen 2 usuarios que suman un total de 9 kW en cada una, lo cual implica

un desperdicio total de 2 kW entre las 2 mochilas. En cada mochila el espacio restante es de 1 kW, por lo que ni siquiera el usuario de menor demanda (2 kW) tiene lugar suficiente. En la segunda imagen (figura 2.15(b)) se ha realizado un intercambio entre los usuarios de 3 kW y 4 kW. El desperdicio continúa siendo de 2 kW, pero ahora este valor se concentra en la segunda mochila, dejando un espacio suficientemente grande para el usuario de 2 kW. En la tercera imagen (figura 2.15(c)) el usuario de 2 kW ha sido agregado a la mochila, finalizando la iteración con un desperdicio total igual a cero.

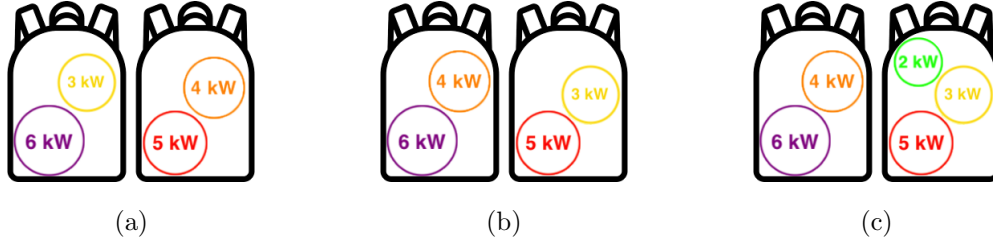


Fig. 2.15: Iteración de la búsqueda local sobre dos períodos de un escenario con 5 usuarios y un límite del suministro de 10 kW. (a) Estado inicial. (b) Intercambio entre los usuarios de 3 kW y 4 kW. (c) Ingreso del usuario de 2 kW.

Volviendo con el escenario A (figura 2.1), que utilizamos para analizar las estrategias de período único, vemos que es posible realizar un intercambio entre los dos primeros períodos del cronograma generado por la estrategia “High-To-Low” (tabla 2.5) para dicho escenario. Los usuarios dentro de la mochila para los períodos 1 y 2 son  $\{A, B, C, D\}$  y  $\{A, B, E, F, G, H\}$ , respectivamente. El usuario D, cuya demanda es de 3 kW, se encuentra dentro de la mochila en el período número 1 y fuera de la mochila en el número 2. Inversamente, el usuario F, cuya demanda es de 1 kW, se encuentra fuera de la mochila en el período número 1 y dentro de la mochila en el número 2. Intercambiamos los usuarios, es decir, el usuario D es sacado de la mochila que representa al período 1 para ser agregado a la mochila representante del período 2, mientras que lo contrario sucede con el usuario F. El cronograma modificado se puede ver en la tabla 2.13.

Usuario	Demanda	Período											
		1		2		3		4		5		6	
A	5 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
B	5 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
C	5 kW	✓			↑	✓	↑	✓	↑		↑↑	✓	↑↑
D	3 kW	✗	⊕	⊙	↑	✓	↑	✓	↑		↑↑	✓	↑↑
E	3 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
F	1 kW	⊙	✗	✗	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
G	1 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
H	1 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
Utilizado		16 kW		18 kW		14 kW		18 kW		16 kW		14 kW	
Desperdiciado		2 kW		0		4 kW		0		2 kW		4 kW	

Tab. 2.13: Búsqueda local sobre el cronograma de la tabla 2.5. Primera iteración - Primer paso: intercambio de usuarios D y F en períodos 1 y 2.

El desperdicio energético, que era nulo en el primer período y de 2 kW en el segundo,



pasa a ser de 2 kW en el primero y nulo en el segundo. La suma del desperdicio de los dos períodos continúa siendo la misma, lo que parece indicar que el intercambio no produjo cambios en los resultados, pero si observamos el primer período, vemos que en la mochila quedó un espacio de 2 kW. Por otro lado, existen dos usuarios (G y H) con 1 kW de demanda que no se encuentran en esa mochila, por lo que pueden ser agregados. En la tabla 2.14 vemos el resultado de prender a los usuarios G y H en el período 1. Al hacerlo, en los dos períodos se alcanza a utilizar el suministro en toda su capacidad, por lo que el desperdicio de ambos es nulo.

Usuario	Demanda	Período											
		1		2		3		4		5		6	
A	5 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
B	5 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
C	5 kW	✓			↑	✓	↑	✓	↑		↑↑	✓	↑↑
D	3 kW		↑	✓	↑	✓	↑	✓	↑		↑↑	✓	↑↑
E	3 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
F	1 kW	✓			↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
G	1 kW	⊙	↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
H	1 kW	⊙	↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
Utilizado		18 kW		18 kW		14 kW		18 kW		16 kW		14 kW	
Desperdiciado		0		0		4 kW		0		2 kW		4 kW	

Tab. 2.14: Búsqueda local sobre el cronograma de la tabla 2.5. Primera iteración - Segundo paso: agregado de usuarios G y H al período 1.

Los usuarios que fueron intercambiados no sufrieron cambios en sus prioridades -a nivel de cronograma- al haber sido encendidos en un período pero apagados en otro. En cambio, los usuarios G y H, que se encontraban apagados en el primer período y fueron encendidos, no necesitaron ser apagados en algún otro período, por lo que su prioridad disminuyó. Al finalizar los seis períodos del cronograma, su prioridad resulta en una unidad menos que la del resto de los usuarios. Ésto nos da la libertad de asignarle un turno a tales usuarios en algún otro período que sea conveniente, ya que al hacerlo, sus prioridades aumentarán, pero no serán mayores a las de otros usuarios.

Los períodos con desperdicio no nulo, en los cuales se puede aplicar una mejora, son los períodos 3, 5 y 6. Si quitáramos al usuario G o H del período 3 o 5, el usuario en cuestión quedaría fuera de la mochila durante 2 períodos consecutivos (3-4 o 4-5), lo cual no sucedía en el cronograma original. El período seleccionado para ser optimizado será entonces el número 6, el cual posee un desperdicio de 4 kW. Si el espacio restante en la mochila para tal período fuese de 5 kW, podríamos agregar a uno de los dos usuarios de 5 kW que quedaron fuera. Aprovechando que los usuarios G y H resultaron con menor prioridad, apagamos a uno de ellos en el período 6, por ejemplo a G. Ahora sí es posible agregar a algún usuario de 5 kW, por ejemplo al usuario A. Vemos estos cambios reflejados en la tabla 2.15. El período 6 bajó su desperdicio de 4 kW a 0, a costa de aumentar la cantidad de turnos asignados de un usuario, pero al mismo tiempo disminuyendo la de otro. Si hubiésemos intentado realizar este cambio previo a encender a los usuarios G y H en el primer período, sus prioridades habrían ascendido a 2 al término del cronograma, mientras que otro usuario habría finalizado con un valor de prioridad igual a 0. La restricción de justicia de nivel 1 habría sido infringida al dejar a un usuario con una cantidad de turnos

asignados que supera en 2 unidades a otro, al que además no le habría sido asignado ningún turno.

Usuario	Demanda	Período											
		1		2		3		4		5		6	
A	5 kW	✓		✓			↑	✓	↑	✓	↑	⊙	↑↓
B	5 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
C	5 kW	✓			↑	✓	↑	✓	↑		↑↑	✓	↑↑
D	3 kW		↑	✓	↑	✓	↑	✓	↑		↑↑	✓	↑↑
E	3 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
F	1 kW	✓			↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
G	1 kW	✓		✓		✓			↑	✓	↑	✗	↑⊕
H	1 kW	✓		✓		✓			↑	✓	↑	✓	↑
Utilizado		18 kW		18 kW		14 kW		18 kW		16 kW		18 kW	
Desperdiciado		0		0		4 kW		0		2 kW		0	

Tab. 2.15: Búsqueda local sobre el cronograma de la tabla 2.5. Primera iteración - Tercer paso: quitado el usuario G y agregado el usuario A al período 6.

En una segunda iteración, intercambiamos los usuarios D y F en los períodos 4 y 5: el usuario D, que se encontraba encendido en el período 4 pero no en el 5, pasa a estar encendido en el período 5 pero no en el 4. Lo contrario sucede con el usuario F, como vemos en la tabla 2.16. El segundo paso es encender usuarios en alguno de los dos períodos que fueron afectados por el intercambio. En este caso, encendemos a los usuarios G y H en el período 4 (tabla 2.17); al hacerlo, estos usuarios pasan a tener una prioridad igual a 0 en el quinto período -no han sido asignados hasta este punto- mientras que a otros usuarios les han sido asignados 2 turnos antes de tal período. Pero la restricción de justicia (ecuación 1.1) menciona que en un cronograma de nivel 1 no pueden existir dos usuarios con diferencia de prioridad mayor a 1.

Para que la restricción de justicia no impacte fuertemente sobre el mejoramiento de un cronograma, vamos a relajarla un poco: buscaremos que un cronograma continúe siendo justo, pero sólo al término de éste. Esto significa que no vamos a restringir que en un período intermedio las prioridades difieran en varias unidades, pero sí vamos a mantener dicha restricción para el final del cronograma, luego del último período. La restricción de justicia modificada se muestra en la ecuación 2.1, la cual implica que para que un cronograma posea nivel de justicia  $k$ , la diferencia entre la cantidad de turnos asignados a cualquier par de usuarios, al término del cronograma, no debe ser mayor que  $k$ .

$$\forall u_1, u_2 \in U : |t_{u_1}^P - t_{u_2}^P| \leq k \quad (2.1)$$

A causa de este cambio en la restricción de justicia, algo que puede llegar a suceder luego de una iteración de la búsqueda local, es que un usuario se encuentre apagado en varios períodos consecutivos, lo cual a pesar de no estar infringiendo la restricción de justicia, es indeseable. Para que ésto no suceda vamos a agregar otra restricción: el máximo número de turnos consecutivos asignados a un usuario en el nuevo cronograma no puede ser mayor al del cronograma original. Esta nueva restricción que relaciona al cronograma original con el cronograma al que se le aplicó la búsqueda local se muestra en la ecuación 2.2. Si el máximo número de turnos consecutivos asignados a cualquier

usuario en el cronograma original es  $M$ , entonces no puede existir un usuario en el nuevo cronograma con una cantidad de turnos consecutivos mayor a  $M$ .

$$\forall i \in [M, \dots, P] \forall u \in U : t_u^i - t_u^{i-M} \leq M \quad (2.2)$$

Ésta restricción genera la idea de que la máxima cantidad de turnos consecutivos asignados a un mismo usuario sea parametrizable. El algoritmo de selección puede ser utilizado en distintos paradigmas: una aplicación móvil controlada por los usuarios; un sistema automatizado utilizando el esquema de Internet of Things; o alguna otra forma en la que sea posible controlar el prendido/apagado de los equipos de aire acondicionado. Si se fuera a utilizar la primera idea (aplicación móvil), puede que apagar y prender el aire varias veces en una tarde sea tedioso para los usuarios, por lo que sería más conveniente que los turnos asignados a un usuario se encuentren pegados, o lo más separados en el tiempo posible. Para este caso estaría permitido una mayor cantidad de turnos consecutivos asignados a un mismo usuario. En cambio, en un sistema automático, el trabajo de prender o apagar el equipo de aire acondicionado cuando finaliza o comienza un turno, no es un problema, por lo que los turnos asignados a un usuario pueden ser distribuidos de forma intercalada, logrando que los usuarios no sufran una seguidilla de turnos consecutivos.

Volviendo al ejemplo, vemos que a pesar de haber relajado la restricción de justicia, el estado del cronograma a este momento (tabla 2.17) no sería válido, ya que al término del cronograma existen varios usuarios con 2 turnos asignados, mientras que el usuario H no posee ninguno. Afortunadamente, resta realizar un paso de la iteración: es el momento de analizar la prioridad de los usuarios y detectar si es necesario realizar modificaciones. El período 3 continúa teniendo un desperdicio de 4 kW, lo cual podemos subsanar sacando a uno de los usuarios de baja prioridad y agregando uno con demanda de 5 kW al período. El usuario de baja prioridad que elegimos para quitar es G, y el que añadimos es B. Por último, el usuario H, al que no le ha sido asignado ningún turno, lo quitamos del período 2 y agregamos al usuario F, el cual acumulaba 2 turnos asignados. Estos cambios se muestran en la tabla 2.18.

Finalizada la segunda iteración, el cronograma queda completamente optimizado (tabla 2.19) y continúa cumpliendo la restricción de justicia de nivel 1, aunque ahora relajada, ya que en algunos períodos (4 y 5) la prioridad de algunos usuarios es 2 unidades mayor que la de otros, pero al término del cronograma las prioridades vuelven a estar balanceadas.

Un pseudocódigo de la búsqueda local se muestra en los algoritmos 9 y 10. En el primero se muestra el procedimiento principal de la estrategia (*improveSchedule*), mientras que en el segundo se pueden ver las funciones auxiliares con más detalle. Cada ejecución del procedimiento *improveSchedule* corresponde a una sola iteración de la búsqueda local. Éste toma como parámetro de entrada un cronograma válido y busca una combinación de períodos y usuarios sobre los cuales pueda aplicar los cambios descritos anteriormente. El algoritmo recorre la lista de todos los pares distintos de períodos, la cual posee una longitud de  $\frac{p(p-1)}{2}$ , por lo que su complejidad computacional es de orden  $\mathcal{O}(p^2)$ , donde  $p$  es la cantidad de períodos en el cronograma. Sobre cada par de períodos debe encontrar un par de usuarios tales que uno de ellos se encuentre prendido en uno de los períodos pero no en el otro, mientras que el segundo usuario esté prendido en el período en el que el primer usuario se encuentra apagado y prendido en el otro. La cantidad de pares de usuarios a recorrer para cumplir este objetivo también se encuentra en el orden cuadrático, pero esta vez sobre la cantidad total de usuarios ( $n$ ), por lo que su complejidad es de orden  $\mathcal{O}(n^2)$ . Si se encuentra un par válido de usuarios para el intercambio (es decir, si la diferencia entre sus

Usuario	Demanda	Período											
		1		2		3		4		5		6	
A	5 kW	✓		✓			↑	✓	↑	✓	↑	✓	↑
B	5 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
C	5 kW	✓			↑	✓	↑	✓	↑		↑↑	✓	↑↑
D	3 kW		↑	✓	↑	✓	↑	✗	↑⊕	⊙	↑↑	✓	↑↑
E	3 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
F	1 kW	✓			↑	✓	↑	⊙	↑	✗	↑↑	✓	↑↑
G	1 kW	✓		✓		✓			↑	✓	↑		↑↑
H	1 kW	✓		✓		✓			↑	✓	↑	✓	↑
Utilizado		18 kW		18 kW		14 kW		16 kW		18 kW		18 kW	
Desperdiciado		0		0		4 kW		2 kW		0		0	

Tab. 2.16: Búsqueda local sobre el cronograma de la tabla 2.5. Segunda iteración - Primer paso: intercambio de usuarios D y F en periodos 4 y 5.

Usuario	Demanda	Período											
		1		2		3		4		5		6	
A	5 kW	✓		✓			↑	✓	↑	✓	↑	✓	↑
B	5 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
C	5 kW	✓			↑	✓	↑	✓	↑		↑↑	✓	↑↑
D	3 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
E	3 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
F	1 kW	✓			↑	✓	↑	✓	↑		↑↑	✓	↑↑
G	1 kW	✓		✓		✓		⊙	↑	✓	↑		↑
H	1 kW	✓		✓		✓		⊙	↑	✓	↑	✓	↑
Utilizado		18 kW		18 kW		14 kW		16 kW		18 kW		18 kW	
Desperdiciado		0		0		4 kW		0		0		0	

Tab. 2.17: Búsqueda local sobre el cronograma de la tabla 2.5. Segunda iteración - Segundo paso: agregado de usuarios G y H al período 4.

demandas no ocasiona un exceso en alguno de los períodos afectados), se genera una nueva copia del cronograma y se realiza el intercambio sobre ésta. Efectuado el intercambio, se intenta prender a usuarios que se encuentren apagados en el período que ha sido aliviado gracias al intercambio, lo cual tiene costo  $\mathcal{O}(n)$ . En caso de que al menos un usuario que se encontraba apagado haya podido ser encendido, se considera que la iteración es exitosa. Antes de retornar el nuevo cronograma, se intenta intercambiar usuarios a nivel de período, buscando prender un usuario y apagar otro de menor demanda sin desbalancear las prioridades. Este paso se ejecuta únicamente en caso de que el cronograma haya sido mejorado previamente, por lo que podríamos considerar que se encuentra fuera del ciclo principal del procedimiento y por lo tanto no influye en la complejidad computacional de éste. Finalmente, la complejidad de cada iteración de la búsqueda local, que recorre cada par de períodos y cada par de usuarios dentro de ellos, está en el orden de  $\mathcal{O}(p^2 \cdot n^3)$ .

Nuevamente realizamos una comparación utilizando el escenario C (figura 2.10) entre la estrategia “Greedy-N-Times” sin búsqueda local y la misma estrategia con búsqueda local activada. En la figura 2.16 se muestra la integral de la potencia no utilizada para

Usuario	Demanda	Período											
		1		2		3		4		5		6	
A	5 kW	✓		✓		⊙	↑	✓	↑	✓	↑	✓	↑
B	5 kW	✓		✓		⊙	↗	✓	↗	✓	↗		↗
C	5 kW	✓			↑	✓	↑	✓	↑		↑↑	✓	↑↑
D	3 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
E	3 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
F	1 kW	✓		⊙	↗	✓	↗	✓	↗		↗	✓	↗
G	1 kW	✓		✓		✗	↑	✓	↑	✓	↑		↑↑
H	1 kW	✓		✗	⊕	✓	⊕	✓	⊕	✓	⊕	✓	⊕
Utilizado		18 kW		18 kW		18 kW		18 kW		18 kW		18 kW	
Desperdiciado		0		0		0		0		0		0	

Tab. 2.18: Búsqueda local sobre el cronograma de la tabla 2.5. Segunda iteración - Tercer paso: quitado el usuario G y agregado el usuario B al período 3; quitado el usuario H y agregado el usuario F al período 2.

Usuario	Demanda	Período											
		1		2		3		4		5		6	
A	5 kW	✓		✓			↑	✓	↑	✓	↑	✓	↑
B	5 kW	✓		✓		✓		✓		✓			↑
C	5 kW	✓			↑	✓	↑	✓	↑		↑↑	✓	↑↑
D	3 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
E	3 kW		↑	✓	↑	✓	↑		↑↑	✓	↑↑	✓	↑↑
F	1 kW	✓		✓		✓		✓			↑	✓	↑
G	1 kW	✓		✓			↑	✓	↑	✓	↑		↑↑
H	1 kW	✓			↑	✓	↑	✓	↑	✓	↑	✓	↑
Utilizado		18 kW		18 kW		18 kW		18 kW		18 kW		18 kW	
Desperdiciado		0		0		0		0		0		0	

Tab. 2.19: Cronograma optimizado resultante de la búsqueda local sobre el cronograma de la tabla 2.5.

distintos límites del suministro. La diferencia entre las curvas es más pronunciada que en el experimento anterior (figura 2.14), se observa una mejora importante en la mayoría de los casos.

El promedio de los valores obtenidos tanto en el experimento de “Multiple-Greedy-N-Times” como en el de búsqueda local se pueden ver en la figura 2.17. Los cronogramas generados por la estrategia que resuelve varios períodos a la vez mejoró, en promedio, un 9 % a los cronogramas generados por “Greedy-N-Times”. La búsqueda local devolvió mejoras con un promedio del 15 % respecto de “Greedy-N-Times”. Habiendo obtenido mejores resultados utilizando estrategias que abarcan múltiples períodos en simultáneo, un factor a considerar es el alto costo computacional que poseen estas estrategias.

**Algorithm 9** Búsqueda local

---

```

1: procedure IMPROVESCHEDULE(schedule, users, powerSupplyLimit)    ▷  $\mathcal{O}(p^2 \cdot n^3)$ 
2:   for each pair of periods  $\langle p_i, p_j \rangle$  in schedule /  $p_i \neq p_j$  do
3:     for each pair of users  $\langle user_k, user_l \rangle$  in users /  $user_k \neq user_l$  do
4:       if ISSWAPVALID(schedule,  $p_i, p_j, user_k, user_l$ )) then
5:         newSchedule  $\leftarrow$  PERFORMSWAP(schedule,  $p_i, p_j, user_k, user_l$ )
6:         lightweightPeriod  $\leftarrow p_i$  if  $user_k.demand() > user_l.demand()$  else  $p_j$ 
7:         improved  $\leftarrow$  APPLYIMPROVEMENTS(newSchedule, lightweightPeriod)
8:         if improved then
9:           return newSchedule
10:        end if
11:      end if
12:    end for
13:  end for
14: end procedure

```

---

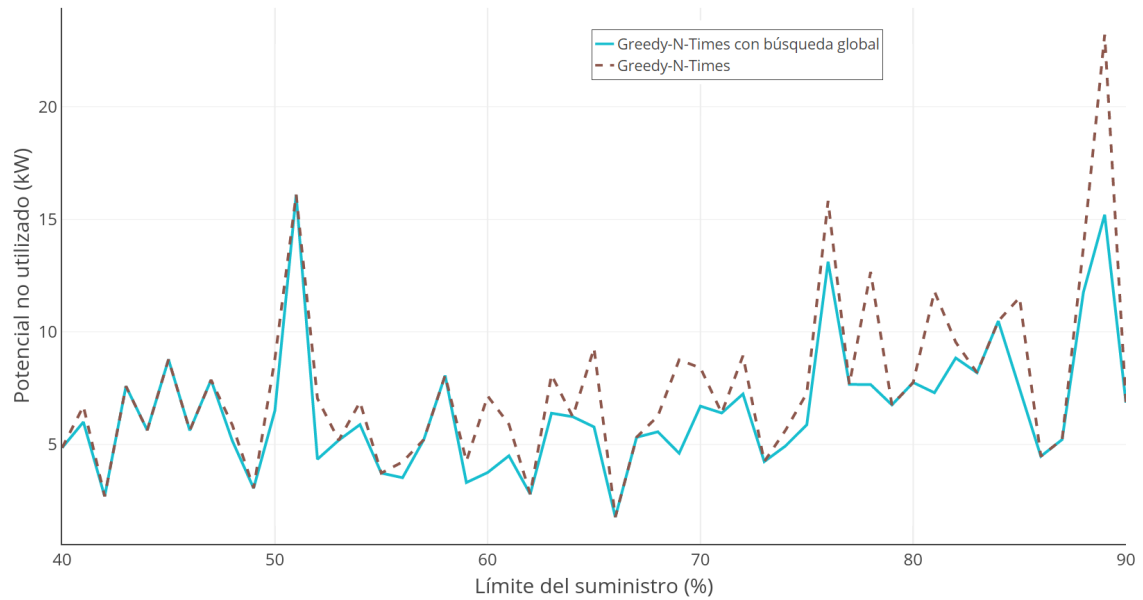


Fig. 2.16: Integral de la potencia no utilizada durante 30 períodos para “Greedy-N-Times” con y sin búsqueda local variando el límite del suministro. Escenario C.

**Algorithm 10** Búsqueda local (métodos auxiliares)

---

```

1: procedure ISWAPVALID(schedule, pi, pj, userk, userl) ▷  $\mathcal{O}(n)$ 
2:   usersAreComplementary ← userk in schedule[pi] and userk not in schedule[pj]
3:   and userl not in schedule[pi] and userl in schedule[pj]
4:   if userk.demand() < userl.demand() then
5:     heavyUser ← userl
6:     heavyPeriod ← pi
7:   else
8:     heavyUser ← userk
9:     heavyPeriod ← pj
10:  end if
11:  powerUsed ← demandOf(users) – demandOf(schedule[heavyPeriod])
12:  powerNotExceeded ← powerUsed + heavyUser.demand() ≤ powerSupplyLimit
13:  return usersAreComplementary and powerNotExceeded
14: end procedure

15: procedure PERFORMSWAP(schedule, pi, pj, userk, userl) ▷  $\mathcal{O}(1)$ 
16:  newSchedule ← schedule
17:  newSchedule[pi].remove(userk)
18:  newSchedule[pi].append(userl)
19:  newSchedule[pj].remove(userl)
20:  newSchedule[pj].append(userk)
21:  return newSchedule
22: end procedure

23: procedure APPLYIMPROVEMENTS(newSchedule, lightweightPeriod) ▷  $\mathcal{O}(n)$ 
24:  powerUsed ← demandOf(users) – demandOf(newSchedule[lightweightPeriod])
25:  powerLeft ← powerSupplyLimit – powerUsed
26:  freedUsers ← [ ]
27:  for user in newSchedule[lightweightPeriod] do
28:    if user.demand() ≤ powerLeft then
29:      newSchedule[period].remove(user)
30:      powerLeft ← powerLeft – user.demand()
31:      freedUsers.append(user)
32:    end if
33:  end for
34:  improvementsMade ← freedUsers.length() > 0
35:  if improvementsMade then
36:    MAKEMOREIMPROVEMENTS(newSchedule, freedUsers)
37:  end if
38:  return improvementsMade
39: end procedure

```

---

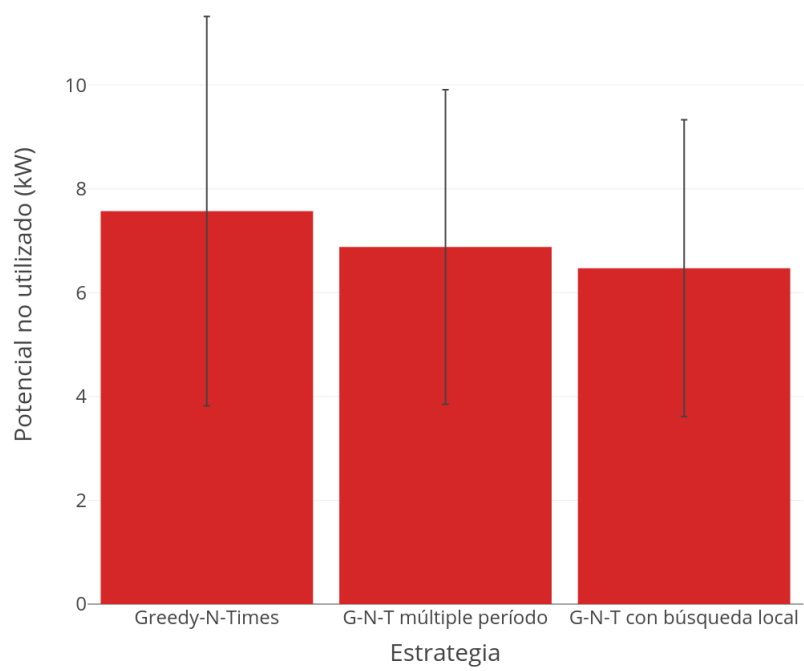


Fig. 2.17: Promedio de las integrales de potencia no utilizada variando el límite del suministro para estrategias de período múltiple. Escenario C.



### 3. CASOS DE ESTUDIO

#### 3.1. Diseño de Escenarios

Con la finalidad de evaluar las estrategias de asignación de turnos en escenarios que se asemejen a la realidad, le fue solicitado al ENRE (Ente Nacional Regulador de la Electricidad de Argentina) información sobre centros de transformación (CT) reales. La información provista por el ENRE incluye, para un conjunto de CT de diferentes características, los siguientes datos: número identificador, potencia aparente, localidad, cantidad de usuarios conectados y consumo total anual (ver Anexo I en la sección 5 para mayor información).

A partir de estos datos se construyeron tres escenarios, correspondientes a tres CT distintos. Estos fueron elegidos por poseer la mayor cantidad de usuarios conectados y por representar a distintas zonas de GBA (Zona Norte, Capital Federal y Zona Sur), factores que enriquecen los experimentos. Para cada escenario se planteó una combinación posible de factores que se ajustara al consumo total anual. Estos factores son:

- **Consumo base:** una lista de los electrodomésticos más utilizados en los hogares de GBA, junto con el porcentaje de hogares propietarios de éstos, sus horas de uso diario y el consumo eléctrico por hora de cada uno. La lista fue armada en base a información orientativa [5][20][21][22] provista por varios organismos nacionales.
- **Aire acondicionado:** partiendo de dos tipos de unidades de aire acondicionado con potencias de 1,3 kW y 2,15 kW, se armó una lista de posibles combinaciones, suponiendo que en un hogar pueden existir 0, 1 o más unidades de cada tipo. La demanda requerida por un usuario para utilizar el aire acondicionado será la sumatoria de las potencias de todos los equipos que posea. No todos los hogares cuentan con aire acondicionado: a Diciembre de 2013, tan sólo un 38,7 % de los hogares del AMBA poseen al menos una unidad [9]. A pesar de que este porcentaje es un promedio y por lo tanto el valor real para cada escenario podría ser más alto o más bajo, nos apegaremos a dicho porcentaje para todos los escenarios a falta de datos más específicos para cada zona. Ésto se verá reflejado en los escenarios, en los cuales la cantidad de usuarios con aire será poco más que la tercera parte de la cantidad total de usuarios en cada uno. A estos usuarios, que sí poseen al menos una unidad, se los distribuirá en las distintas combinaciones de aire acondicionado posibles con alguna probabilidad, asignándole una demanda requerida para utilizar el aire acondicionado a cada usuario.
- **Calefacción:** el consumo dado por los artefactos correspondientes a la calefacción no formará parte del consumo base, ya que éstos no suelen ser utilizados paralelamente con el aire acondicionado. Es decir, los escenarios planteados se sitúan en meses de alta temperatura, en los que los equipos de aire acondicionado son utilizados para refrigerar los hogares por la gran mayoría de los usuarios. Por lo tanto, el consumo necesario para calefaccionar los hogares influirá en el consumo anual, pero no restará potencia disponible al transformador para el uso de los equipos de aire acondicionado.

En cada escenario, la sumatoria del consumo base, el consumo dado por los equipos de aire acondicionado y el dado por los artefactos de calefacción, conforman el consumo

total anual estimado. Los datos restantes fueron ajustados para que este valor difiera lo menos posible del consumo total anual real (suministrado por el ENRE) de manera de tener escenarios que reflejen de cierta forma la realidad.

El paso siguiente fue calcular la potencia disponible para abastecer a los equipos de aire acondicionado, que dependerá de dos valores:

- **Potencia del transformador:** La potencia de los transformadores fue suministrada por el ENRE, aunque los valores dados corresponden a la potencia aparente de cada uno, que es la sumatoria entre la potencia activa y la potencia reactiva. La potencia activa (útil) de los transformadores se calcula como el producto entre la potencia aparente y el factor de potencia, cuyo valor puede encontrarse entre 0 y 1 (siendo 1 el mayor grado de eficiencia posible y 0 el menor). En los transformadores, el factor de potencia suele estar situado cerca del 0,8, lo cual significa que la potencia disponible será el 80 % de su potencia aparente especificada (figura 3.1(a)).
- **Consumo base por hora:** El consumo base varía según la hora del día en la que nos encontremos, por lo que dividir el consumo base diario sobre las 24 horas del día no sería muy realista. CAMMESA (Compañía Administradora del Mercado Mayorista Eléctrico) divide al día en 3 períodos, según el consumo promedio en cada uno: “Valle” (23 hs a 5 hs), “Resto” (5 hs a 18 hs) y “Pico” (18 hs a 23 hs), siendo el consumo del segundo un 20 % más alto que el del primero y el consumo del tercero un 15 % más alto que el del segundo. En los experimentos supondremos que nos encontramos en el horario de mayor consumo, es decir, en la zona horaria denominada “Pico”, donde el consumo base será mayor al promedio. Además, asumiremos que todos los equipos de aire acondicionado se encuentran encendidos durante esa franja horaria, lo cual constituye el peor caso posible.

A la potencia útil del transformador de cada escenario le substraemos la demanda necesaria para el consumo base en horario “Pico”, siendo el valor restante la potencia disponible para la utilización de los equipos de aire acondicionado (figura 3.1(b)). A este valor es al que llamamos “límite del suministro” durante el presente trabajo, el cual no hace referencia a la potencia total del transformador, sino que representa a la potencia destinada al abastecimiento de los equipos de aire acondicionado. La diferencia entre la demanda de todos los equipos de aire acondicionado incluidos en un escenario y el límite de suministro del mismo escenario será el “exceso” en demanda. En cada período de un cronograma se deberá asignar turnos a un conjunto de usuarios tal que la demanda de éstos (usuarios apagados) iguale o supere el exceso, de manera de mantener la utilización del suministro por debajo de su límite.

En la tabla 3.1 se muestra un resumen de los escenarios contruidos, identificados por la locación de los CT: “Palermo”, “Olivos” y “Avellaneda”. Se presentan únicamente los datos vinculados a la utilización del aire acondicionado, ya que son los parámetros requeridos por el algoritmo de selección. El resto de la configuración de los escenarios se puede encontrar en el archivo descrito en el Anexo I (sección 5).

### 3.2. Resultados

El primer experimento realizado sobre los escenarios reales, presentados en la tabla 3.1, se llevó a cabo manteniendo el límite del suministro especificado en dicha tabla. Luego se

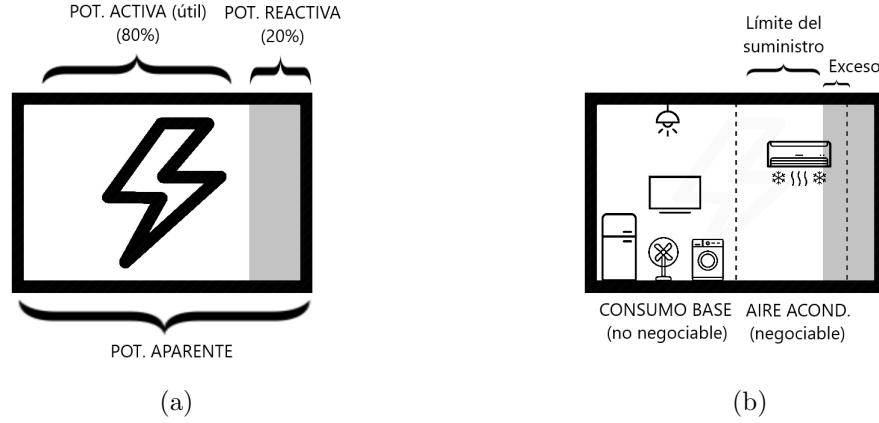


Fig. 3.1: (a) Diferencia entre los distintos tipos de potencia generados por un transformador. (b) División del transformador entre el consumo base y el aire acondicionado.

Demanda por usuario (kW)	Cantidad de usuarios según demanda		
	Palermo	Olivos	Avellaneda
1,3	136	58	166
2,15	98	58	125
2,6	59	14	83
3,45	39	58	42
3,9	20	14	0
4,3	20	14	0
4,75	20	29	0
6,05	0	43	0
Total usuarios con aire	392	288	416
Total demanda de aire (kW)	934,45	949,3	845,25
Límite del suministro (kW)	519,22	276,55	501,46

Tab. 3.1: Resumen de los valores principales para los escenarios Palermo, Olivos y Avellaneda. Los valores del límite del suministro son los disponibles para aire acondicionado.

realizaron experimentos modificando tal valor, estableciéndolo como un porcentaje de la demanda que debe abastecer, es decir, la demanda necesaria para la utilización del aire acondicionado.

En las figuras 3.2 y 3.3 se muestran los resultados del primer experimento, utilizando las estrategias analizadas en las secciones 2.2 (“High-To-Low” y “Low-To-High”), 2.3 (“Alternate by Demand” y “Balance Demand”) y 2.4 (“Greedy-N-Times”). En el caso de la estrategia aleatoria (“RandomPick”) se realizaron 1.000 corridas, por lo que se muestra el promedio y desvío correspondientes a dicha estrategia. En la primera figura se puede observar, para cada estrategia, la potencia no utilizada en cada período, es decir, el desperdicio. Recordemos que el desperdicio se refiere a la potencia disponible del suministro que no fue aprovechada, por lo que vamos a querer mantener este valor lo más bajo posible. En la segunda figura se muestra la integral del área debajo de cada curva de la figura anterior, donde se visualiza el puntaje total obtenido por cada estrategia en cuanto a potencia desperdiciado (más es peor). La estrategia vista en la sección 2.4, “Greedy-N-Times”, es la mejor opción en los tres escenarios, logrando aprovechar la potencia del transformador

casi al máximo en la mayoría de los períodos.

En la sección 2.5 se analizaron estrategias que buscaban mejorar “Greedy-N-Times” al trabajar con más de 1 período a la vez. El problema de éstas es el tiempo que requieren para encontrar una solución, el cual aumenta junto con la cantidad de usuarios y de períodos. En el caso de “Multiple Greedy-N-Times” el aumento es exponencial ( $\mathcal{O}(p \cdot n^{p+1})$ ,  $n$  cantidad de usuarios,  $p$  cantidad de períodos), mientras que la búsqueda local es cuadrática en la cantidad de períodos y cúbica en la cantidad de usuarios ( $\mathcal{O}(p^2 \cdot n^3)$ ). Los escenarios reales poseen cientos de usuarios, por lo que utilizar estas estrategias implica un alto costo temporal del cómputo. Por otro lado, el espacio de mejora sobre los resultados obtenidos por la estrategia “Greedy-N-Times” es muy bajo, ya que el desperdicio eléctrico de los cronogramas construidos por esta estrategia es casi nulo en los tres escenarios reales. A modo de ejemplo, el cronograma construido por “Greedy-N-Times” para el Escenario Olivos, el cual le tomó a la estrategia un tiempo menor a 1 minuto en generar, posee una disminución del desperdicio de potencia eléctrica de aproximadamente el 94 % comparado con el promedio de la estrategia aleatoria. Al aplicar la búsqueda local vista en la sección 2.5.2, que tardó varias horas en finalizar, se logró una mejora sobre la estrategia aleatoria del 97 %. La gran mejora obtenida por “Greedy-N-Times”, junto con la poca diferencia que agrega la búsqueda local a esta última, indica que ésta no justifica el tiempo de cómputo invertido.

En cuanto al segundo experimento, se varió el valor del límite del suministro entre el 40 % y el 90 % de la demanda total. En la figura 3.4 se puede observar el promedio y el desvío estándar de los puntajes obtenidos para las distintas estrategias. La estrategia “Greedy-N-Times” se mantiene como la mejor opción en los tres escenarios.

Aunque la optimización de la potencia eléctrica utilizada por cada estrategia es el principal foco del trabajo, se obtuvieron otros indicadores de los experimentos, resumidos en la tabla 3.2. Los resultados corresponden al primer experimento realizado sobre los escenarios reales, donde los parámetros de cada escenario se muestran en la tabla 3.1. Definir un valor para la duración de los períodos fue necesario en este caso, ya que algunos indicadores utilizan el tiempo como medida. El valor seleccionado para la duración de un período fue de 15 minutos. Los indicadores reunidos podrían servir para llegar a conclusiones acerca de cuál es la mejor estrategia bajo circunstancias en las cuales el desperdicio eléctrico no es la preocupación principal. Algunas observaciones interesantes que salen de los datos presentados en la tabla 3.2 son, por ejemplo:

- La estrategia “Low-To-High” posee el menor tiempo con consumismo reducido promedio por usuario en los tres escenarios.
- El intervalo máximo con consumo reducido de la estrategia “BalanceDemand” es igual al mínimo valor en cada escenario.
- En cuanto a la cantidad máxima de equipos de aire acondicionado encendidos simultáneamente, nuevamente la estrategia “Low-To-High” es la ganadora, con el máximo valor en todos los escenarios.

	High-To-Low	Low-To-High	RandomPick	AlternateDemand	BalanceDemand	GreedyNTimes
Escenario: Palermo						
Tiempo con consumo reducido promedio por usuario (minutos)	194,14	182	194,4	194,38	194,5	193,68
Intervalo máximo con consumo reducido (minutos)	45	45	45	30	30	45
Cantidad de veces que debe encender promedio por usuario	12,98	12,94	12,96	12,96	12,97	12,91
Cantidad total de veces que se debe encender	4997	4983	4989	4989	4992	4971
Tiempo medio y desvío estándar entre cambios de estado (minutos)	17,27	17,37	21,97	17,87	17,4	21,01
	5,43	5,23	9,7	6,18	5,52	9,4
Cantidad máxima de AA encendidos simultáneamente	278	283	233	240	238	280

Escenario: Olivos

Tiempo con consumo reducido promedio por usuario (minutos)	322,08	317,24	319,13	318,7	320,52	319,27
Intervalo máximo con consumo reducido (minutos)	75	90	90	75	75	90
Cantidad de veces que debe encender promedio por usuario	21,47	21,15	21,27	21,25	21,37	21,28
Cantidad total de veces que se debe encender	6184	6091	6127	6119	6154	6130
Tiempo medio y desvío estándar entre cambios de estado (minutos)	25,56	25,22	26,5	25,26	25,42	26,48
	12,05	11,98	17,32	12,19	12,12	15,24
Cantidad máxima de AA encendidos simultáneamente	137	141	96	93	109	136

Escenario: Avellaneda

Tiempo con consumo reducido promedio por usuario (minutos)	185,23	182,1	183,3	182,92	183,64	182,6
Intervalo máximo con consumo reducido (minutos)	45	30	45	45	30	45
Cantidad de veces que debe encender promedio por usuario	12,35	12,14	12,22	12,19	12,24	12,17
Cantidad total de veces que se debe encender	5137	5050	5083	5073	5093	5064
Tiempo medio y desvío estándar entre cambios de estado (minutos)	18,27	18,4	22,42	18,72	18,82	21,03
	6,23	6,32	10,49	6,85	6,88	9,22
Cantidad máxima de AA encendidos simultáneamente	297	297	259	265	266	297

Tab. 3.2: Otros indicadores para estrategias de período único aplicadas sobre escenarios reales.

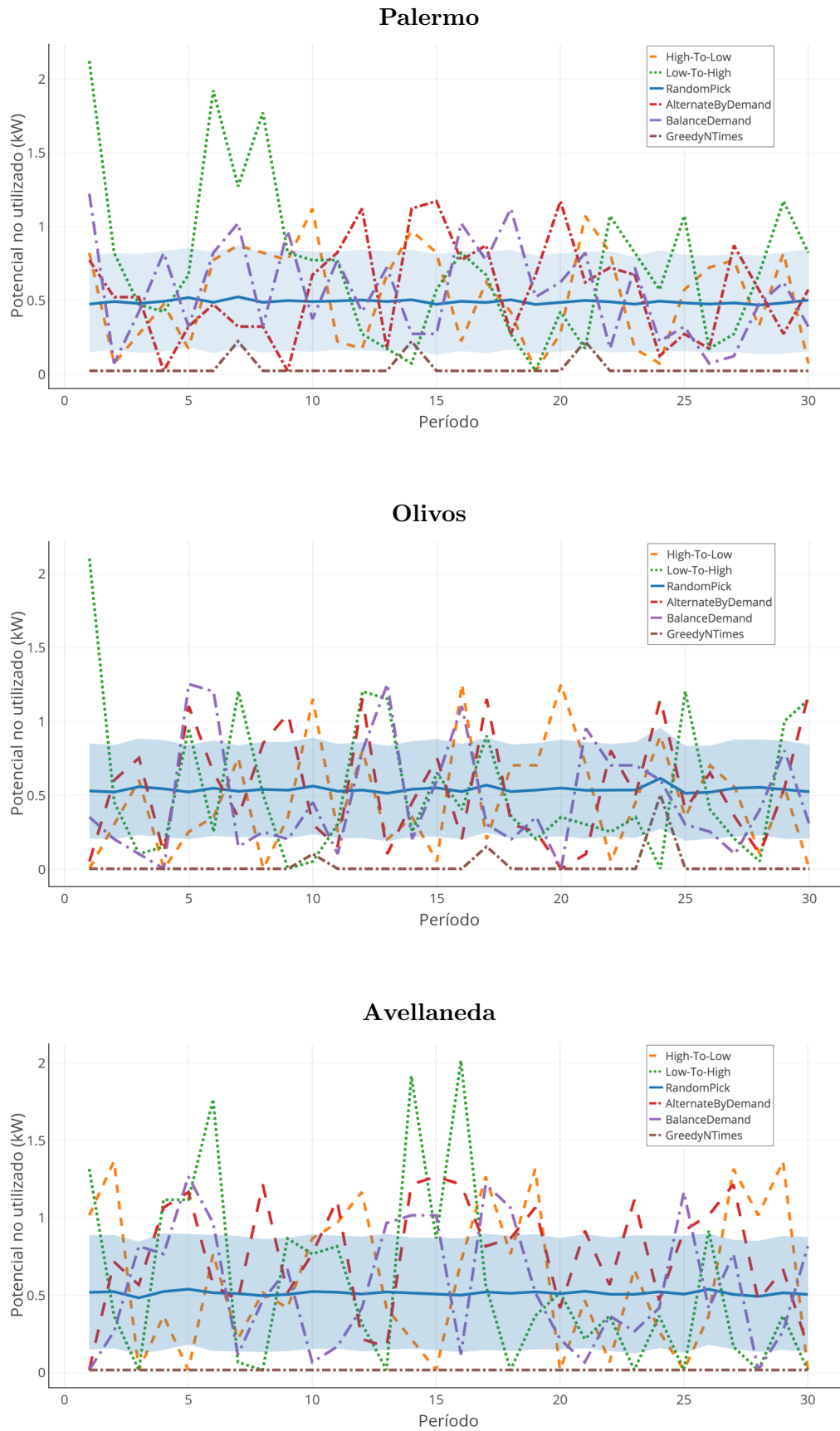


Fig. 3.2: Potencia no utilizada por período para distintas estrategias sobre escenarios reales.

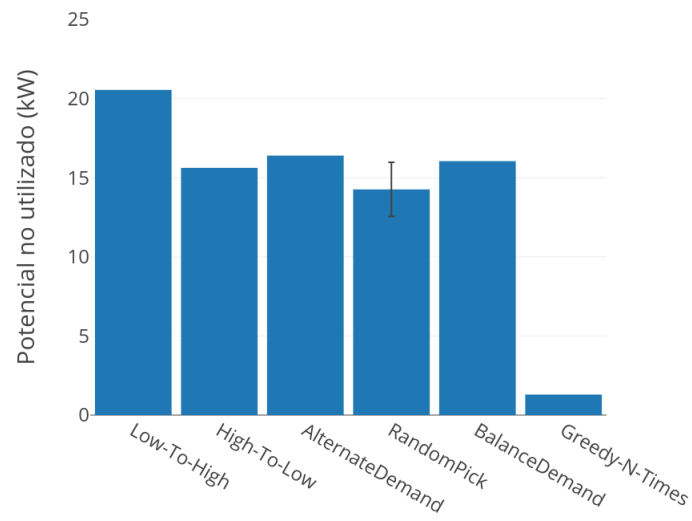
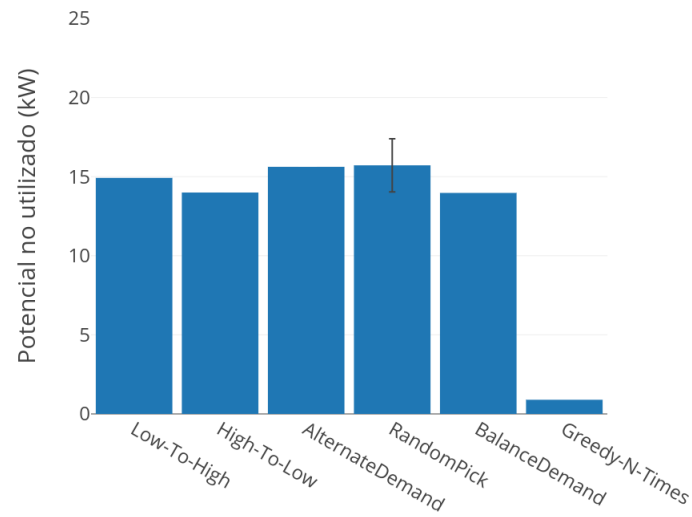
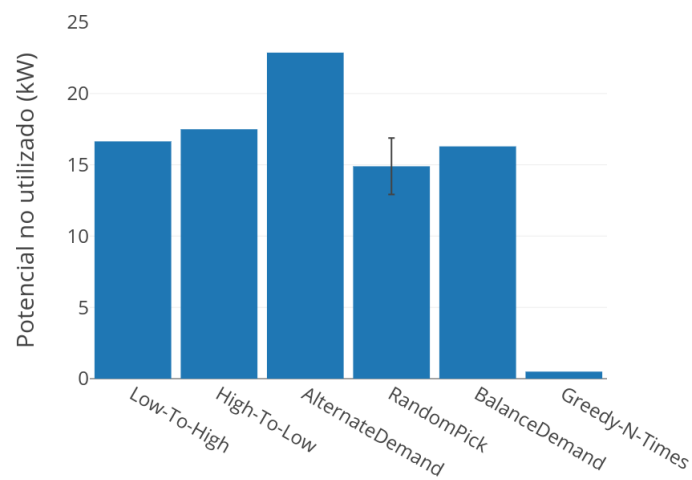
**Palermo****Olivos****Avellaneda**

Fig. 3.3: Integral de la potencia no utilizada para cada estrategia sobre escenarios reales.

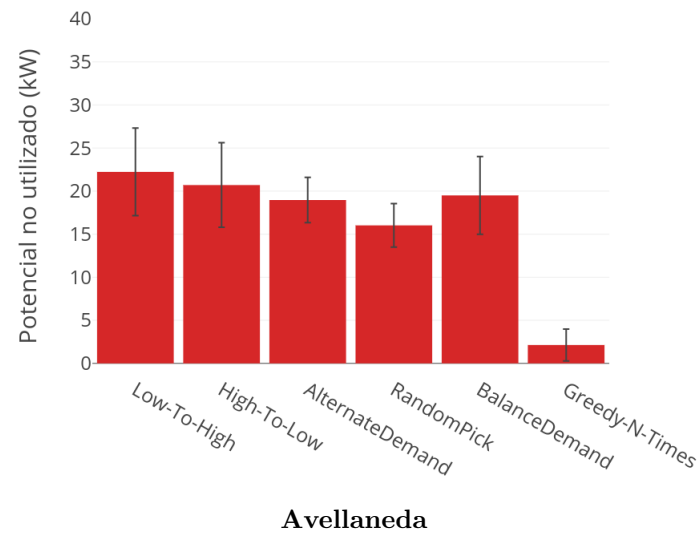
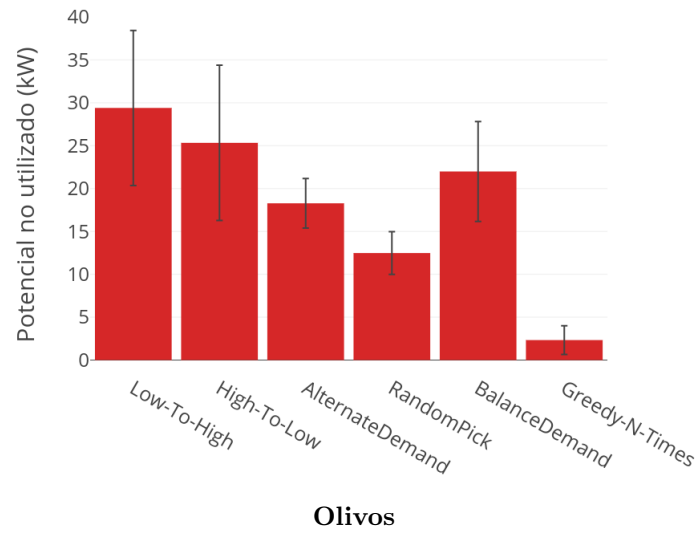
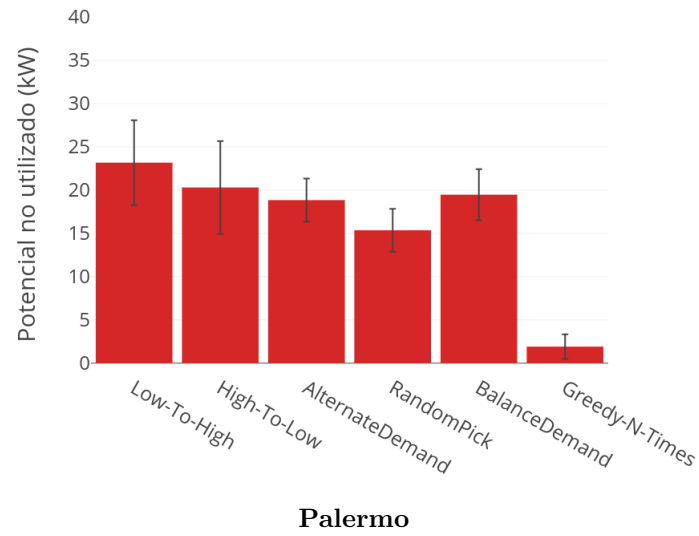


Fig. 3.4: Promedio de las integrales de la potencia no utilizada para cada estrategia variando el límite del suministro eléctrico (40 % - 90 %) sobre escenarios reales.



## 4. CONCLUSIONES Y TRABAJO FUTURO

### 4.1. Conclusiones

Los equipos de aire acondicionado provocan un gran aumento en la demanda eléctrica residencial, por lo que son vistos como un factor importante en las interrupciones al servicio eléctrico. En el presente trabajo se analizó la idea de construir un sistema para controlar su uso, que seleccione a los usuarios que apagarían sus equipos de aire acondicionado y, cada un intervalo de tiempo, los rote. El problema de seleccionar a los usuarios sin superar el límite del suministro se asemeja al problema de la mochila, en el cual se debe llenar una mochila de capacidad limitada con elementos que poseen un valor y un peso. El algoritmo requerido por el sistema debe resolver varias instancias del problema de la mochila: uno para realizar la selección inicial y uno por cada rotación que se desee realizar. Cada instancia es distinta ya que el algoritmo debe tener en cuenta la prioridad de los usuarios: de ser posible, un mismo usuario no debería tener que apagar el aire acondicionado varias veces consecutivas.

Las primeras estrategias analizadas para utilizar como algoritmo de selección fueron las denominadas “High-To-Low” y “Low-To-High”, que seleccionan a los usuarios que podrán utilizar el aire acondicionado según su demanda, de forma descendente o ascendente, respectivamente. Luego se estudió la posibilidad de seleccionar un subconjunto de usuarios con demanda intermedia, dejando fuera a los usuarios de mayor y menor demanda, estrategia a la que llamamos “Alternate by Demand”. Su nombre hace referencia a que una forma de implementar esta estrategia es seleccionar al usuario de demanda máxima (o mínima) como uno de los que no podrá utilizar el aire acondicionado, luego al usuario cuya demanda se encuentre en el otro extremo, y así sucesivamente. Una estrategia similar a la anterior, que denominamos “Balance Demand”, consiste en equilibrar la cantidad de usuarios que seleccionamos de los extremos: luego de incluir un usuario de demanda máxima en el subconjunto de usuarios que no podrán utilizar el aire, se incluye a un subconjunto de usuarios de mínima demanda, tal que la demanda total de estos usuarios logre igualar a la del usuario de mayor demanda. Luego vimos una estrategia que explora varios subconjuntos de usuarios con potencial para utilizar el aire acondicionado, eligiendo la opción que mejor se ajuste al límite del suministro. La forma de armar los subconjuntos es ordenando a los usuarios por demanda y quitando los extremos, realizando los cortes de tales extremos en distintos puntos. Esta última estrategia fue denominada “Greedy-N-Times”.

Ideamos dos maneras de mejorar la estrategia “Greedy-N-Times”. Por un lado, vimos una nueva estrategia inspirada en esta última, la cual genera un árbol de combinaciones para utilizar en más de un período al mismo tiempo, denominada “Multiple Greedy-N-Times”. Por otro lado, aplicamos un algoritmo de mejoramiento a las soluciones generadas por la estrategia “Greedy-N-Times” (aunque puede ser utilizado sobre los cronogramas generados por cualquier estrategia), que intercambia el estado (encendido o apagado) de algunos usuarios en un par de períodos, con el objetivo de generar un hueco lo suficientemente grande en un período tal que otro usuario pueda ser encendido. Vimos que a pesar de percibir una mejor relativa a “Greedy-N-Times”, el tiempo excesivo necesario para computar los cronogramas que generan estas estrategias supera al beneficio de utilizarlas. Este mismo resultado fue corroborado en escenarios reales. Sin embargo, no debemos dejar

de mencionar que la estrategia “Greedy-N-Times”, al igual que todas las analizadas en el presente trabajo, es heurística, por lo que pueden existir escenarios en los que no produzca buenos resultados.

Damos por cumplido el objetivo del presente trabajo, el cual consistía en analizar la factibilidad de un algoritmo de selección y rotación de los usuarios residenciales de la red eléctrica Argentina, cuya finalidad sería la de evitar interrupciones al suministro eléctrico en caso de una utilización elevada de la red.

## 4.2. Trabajo futuro

La justicia entre usuarios fue tomada como un factor de gran importancia, por lo que la prioridad de los usuarios tomó un rol en el algoritmo de armado de cronogramas al restringir la selección de los usuarios que habían sido asignados previamente. Relajar esta restricción de justicia podría optimizar la utilización del suministro, siendo necesario encontrar un balance entre justicia y eficiencia energética.

En la realidad existe la posibilidad de que los usuarios ignoren turnos que le han sido asignados, lo cual desembocaría en una utilización del suministro eléctrico por encima del límite, haciendo que los esfuerzos de los demás usuarios, que sí respetan sus turnos y apagan sus equipos de aire acondicionado, sean en vano. El algoritmo de armado de cronogramas debería incluir algún mecanismo para reducir la probabilidad de que ésto suceda, por ejemplo, restando al límite del suministro un valor que represente la sumatoria de la demanda de los usuarios que vayan a ignorar su turno. Este valor a substraer podría ser fijo, como por ejemplo un porcentaje de la demanda total de los usuarios, o podría ser calculado mediante alguna distribución probabilística que involucre a la cantidad de usuarios y sus demandas, y a la probabilidad de que un usuario ignore un turno.

Actualmente, tanto el consumo base (no negociable) como el consumo de los equipos de aire acondicionado de los usuarios se mantienen fijos durante el transcurso de un cronograma. Sería interesante parametrizar estos valores en cada período: por un lado, el cambio en el consumo base de los usuarios no modificaría la demanda de los usuarios (ésta no incluye el consumo base) pero implicaría un valor distinto para el límite del suministro en cada uno (a mayor consumo base, menor porcentaje del suministro dedicado a los equipos de aires acondicionados). Por otro lado, la demanda de los usuarios para utilizar el aire acondicionado también podría ser variable, reflejando el hecho de que un usuario enciende y apaga sus equipos de aire acondicionado en múltiples ocasiones durante un día.

Como ya se mencionó en la sección 2.4, existe la posibilidad de mejorar la estrategia “Greedy-N-Times” al convertir la lista de usuarios en una lista circular: al momento de generar los subconjuntos, los que empiezan por un usuario que se encuentra cerca del final de la lista, podrían añadir usuarios del principio de la lista y así obtener nuevos subconjuntos.

Si bien no fue probado, una posibilidad a futuro es explorar cómo funcionaría una adaptación de algoritmos genéticos para el armado de cronogramas, los cuales son muy utilizados en múltiples variantes del problema de la mochila.

## 5. ANEXO I: DATOS DE ESCENARIOS REALES

En la tabla 3.1 de la sección 3.1 se presentan algunos datos sobre los escenarios reales que se armaron para realizar los experimentos. La información completa sobre estos escenarios se puede visualizar en el archivo “Escenario.xlsx” que se encuentra en el directorio `material/`. Este archivo es una hoja de cálculo que consta de tres pestañas: “Palermo”, “Olivos” y “Avellaneda”, correspondientes a los escenarios vistos en la sección 3.1. Cada pestaña inicia con los datos provistos por el ENRE para cada centro de transformación: el número de centro, la localidad, la cantidad de usuarios conectados, el consumo total anual y promedio mensual por usuario, y la potencia aparente. Se puede corroborar esta información en el archivo “ENRE.pdf” que se encuentra en el mismo directorio (`material/`), el cual contiene un escaneo del documento provisto por el ENRE. Luego se muestra un armado especulativo de la lista de artefactos que componen el consumo de los usuarios, tanto para el consumo base, como para el aire acondicionado y la calefacción. En cada escenario se propone una distribución distinta en la cantidad de usuarios asignados a cada combinación de equipos de aire acondicionado, con el objetivo de probar las estrategias en escenarios que difieran entre sí. Para tener una especie de verificación de los consumos planteados, se compara el consumo especificado por el ENRE y el consumo estimado (el cual se desprende de la cantidad de usuarios y sus consumos), intentando que la diferencia entre estos dos valores se acerque lo mas posible a cero. Por último, se calcula la potencia restante del transformador al substraer el consumo base de todos los usuarios, que puede ser utilizada por los equipos de aire acondicionado.

## 6. ANEXO II: EJECUCIÓN DE LOS EXPERIMENTOS Y DATOS RESULTANTES

Para la ejecución de los experimentos se provee un script llamado “main.py”, que se utiliza de la siguiente manera:

1. Instalar Python y pip (<https://www.python.org/downloads/>)
2. Abrir una terminal en el directorio raíz del entregable y ejecutar

```
$ cd src
$ pip install -r requirements.txt
$ python main.py -s ESTRATEGIA -e ESCENARIO [-l]
```

Donde el parámetro **ESTRATEGIA** es el nombre de la estrategia de asignación que se utilizará, sin guiones y en minúscula, y debe ser uno de los siguientes: **hightolow**, **lowtohigh**, **balancedemand**, **alternatebydemand**, **randompick**, **greedyntimes** y **multipleperiodsgreedyntimes**. El parámetro **ESCENARIO** debe ser la ruta hacia un archivo con el siguiente formato: la primera línea debe estar dada por el límite del suministro en watts; cada línea subsiguiente debe estar compuesta por dos valores separados por un espacio, donde el primer valor representa la demanda de un usuario en watts y el segundo a la cantidad de usuarios que poseen tal demanda. Los escenarios ya existentes son: **A**, **B**, **C**, **Avellaneda**, **Palermo** y **Olivos** (pueden ser especificados utilizando estos nombres). Los archivos que describen estos escenarios se encuentran en el directorio **experiments/settings/**. El parámetro *l* es opcional y puede ser especificado para activar la búsqueda local.

La salida generada por el script consta de dos archivos, cuyos nombres se definen a partir del escenario y estrategia utilizados. Por ejemplo, al especificar la estrategia **hightolow** y el escenario **A**, los archivos generados serán **A\_HIGH-T0-LOW** y **A\_HIGH-T0-LOW\_range**. El primero de los archivos contendrá el resultado correspondiente a un único cronograma, generado utilizando el límite del suministro especificado en el escenario. La información dada será la integral de la potencia no utilizada total del cronograma y la potencia no utilizada en cada período de éste (promedio y desvío estándar en el caso de la estrategia **RandomPick**). El segundo archivo corresponde a los cronogramas generados variando el límite del suministro entre el 40 % y el 90 % de la demanda total de los usuarios. Cada porcentaje contenido en el intervalo denota un escenario particular. Para cada uno, se muestra la integral de la potencia no utilizada total en el cronograma generado para dicho escenario (nuevamente, promedio y desvío estándar para la estrategia aleatoria). Además, se informa el promedio y el desvío estándar de las integrales.

El set de datos resultante de los experimentos, utilizado para realizar los gráficos del presente trabajo, se encuentra en el directorio **experiments/results/**. Éstos pueden ser generados en su totalidad utilizando el script descrito en esta sección.

## Bibliografía

- [1] Edenor. Estándares de calidad. [http://www.edenor.com.ar/cms/SP/EMP/ACE/COM\\_EST\\_estandares.html](http://www.edenor.com.ar/cms/SP/EMP/ACE/COM_EST_estandares.html). Accedido: 2018-Febrero-24.
- [2] Mathaios Panteli and Pierluigi Mancarella. Influence of extreme weather and climate change on the resilience of power systems: Impacts and possible mitigation strategies. *Electric Power Systems Research*, 127:259 – 270, 2015.
- [3] Richard J. Campbell. Weather-related power outages and electric system resiliency. <http://docs.jalite-group.com/data/docs/pdf/R42696.pdf>, 2012. Accedido: 2019-Febrero-26.
- [4] Ministerio de Energía y Minería. La temperatura y su influencia en la demanda de energía eléctrica. <http://datos.minem.gob.ar/dataset/f3552056-dfb9-4b99-a1b8-cebf9e3d6a24/resource/2cf973bf-6de9-43a6-81d0-c9fcd88db94/download/as14946098911.pdf>. Accedido: 2018-Febrero-24.
- [5] INTI. Uso residencial - artefactos eléctricos. <http://www.inti.gob.ar/energia/index.php?seccion=uResidencial>. Accedido: 2018-Febrero-24.
- [6] FUNDELEC. Tarifas en el sistema eléctrico argentino (septiembre de 2006). <http://www.fundelec.com.ar/informes/info0013.pdf>. Accedido: 2018-Agosto-21.
- [7] Karen Herter, Patrick McAuliffe, and Arthur Rosenfeld. An exploratory analysis of california residential customer response to critical peak pricing of electricity. *Energy*, 32(1):25 – 34, 2007.
- [8] Morna Isaac and Detlef P. van Vuuren. Modeling global residential sector energy demand for heating and air conditioning in the context of climate change. *Energy Policy*, 37(2):507 – 521, 2009.
- [9] FUNDELEC. Informe verano 2014-2015. <http://fundelec.com.ar/informes/2005-01-info.pdf>. Accedido: 2018-Febrero-24.
- [10] INDEC. Estadísticas de productos industriales. [https://www.indec.gov.ar/nivel4\\_default.asp?id\\_tema\\_1=3&id\\_tema\\_2=6&id\\_tema\\_3=18](https://www.indec.gov.ar/nivel4_default.asp?id_tema_1=3&id_tema_2=6&id_tema_3=18). Accedido: 2018-Febrero-24.
- [11] Hamid Gharavi and Reza Ghafurian. Smart grid: The electric energy system of the future [scanning the issue]. *Proceedings of the IEEE*, 99(6):917 – 921, 2011.
- [12] M.H. Albadi and E.F. El-Saadany. A summary of demand response in electricity markets. *Electric Power Systems Research*, 78(11):1989 – 1996, 2008.
- [13] N. Zhu, X. Bai, and J. Meng. Benefits analysis of all parties participating in demand response. In *2011 Asia-Pacific Power and Energy Engineering Conference*, pages 1–4, March 2011.

- 
- [14] A. Mohsenian-Rad, V. W. S. Wong, J. Jatskevich, R. Schober, and A. Leon-Garcia. Autonomous demand-side management based on game-theoretic energy consumption scheduling for the future smart grid. *IEEE Transactions on Smart Grid*, 1(3):320–331, Dec 2010.
  - [15] Saurabh Chanana and Monika Arora. Demand response from residential air conditioning load using a programmable communication thermostat. *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, 7(12):1670 – 1676, 2013.
  - [16] Srinivas Katipamula and Ning Lu. Evaluation of residential hvac control strategies for demand response programs. *ASHRAE Transactions*, 112(1):535 (1-12), 2 2006.
  - [17] DC UBA. Competencia TECNOx año 2016. <https://noticias.dc.uba.ar/2016/04/27/tecnnox/>. Accedido: 2018-Agosto-19.
  - [18] Ing. Oscar Medina. Redes inteligentes ¿realidad, utopía o futuro? [https://www.energia.gov.ar/contenidos/archivos/Reorganizacion/contenidos\\_didacticos/publicaciones/redes\\_inteligentes.pdf](https://www.energia.gov.ar/contenidos/archivos/Reorganizacion/contenidos_didacticos/publicaciones/redes_inteligentes.pdf). Accedido: 2018-Septiembre-30.
  - [19] Silvano Martello and Paolo Toth. *Knapsack Problems: Algorithms and Computer Implementations*. John Wiley & Sons, Inc., New York, NY, USA, 1990.
  - [20] ENRE. Consumo indicativo de algunos artefactos eléctricos. <https://www.enre.gov.ar/web/web.nsf/Consumo?OpenPage>. Accedido: 2018-Junio-6.
  - [21] EDENOR. Consumo de artefactos eléctricos. [http://www.edenor.com.ar/cms/SP/CLI/HOG/USO\\_consumo.html](http://www.edenor.com.ar/cms/SP/CLI/HOG/USO_consumo.html). Accedido: 2018-Junio-6.
  - [22] INDEC. El gasto de consumo de los hogares urbanos en la argentina. [https://www.indec.gov.ar/ftp/cuadros/sociedad/engho\\_serie47.pdf](https://www.indec.gov.ar/ftp/cuadros/sociedad/engho_serie47.pdf). Accedido: 2018-Junio-6.