Algoritmos e Formulações para Problemas de Empacotamento Unidimensional e Relacionados

Exame de Qualificação Específico de Doutorado

Aluno: Yulle Glebbyo Felipe Borges[1]

Orientador: Rafael Crivellari Saliba Schouery[2]

Co-orientador: Flávio Keidi Miyazawa[3]

Instituição sede: Instituto de Computação – Unicamp

Resumo

Nesse projeto de pesquisa pretendemos contribuir para o avanço científico nas áreas de Otimização Combinatória e Pesquisa Operacional, focando, em particular, em alguns problemas de corte e empacotamento unidimensional. Considerando a importância prática de tais problemas, faz-se necessário projetar algoritmos exatos e heurísticas para os mesmos, além de investigar a estrutura combinatória destes, explorando-as para alcançar novos patamares na resolução dos problemas.

Assim, nosso objetivo é o projeto e a implementação de algoritmos exatos e de heurísticas para problemas de corte e empacotamento que melhorem o estado-da-arte. Em particular, consideramos alguns problemas de corte e empacotamento com aplicações em diferentes industrias, dentre eles: o Problema da Mochila Compartimentada, o Problema do Empacotamento Colorido e o Problema do Empacotamento com Cenários. As abordagens utilizadas neste projeto variam desde Programação Linear Inteira como *Branch-and-Price*, meta-heurísticas como *Variable Neighborhood Search* até algoritmos de aproximação.

Para alguns destes problemas, alguns resultados já foram obtidos no decorrer dos dois primeiros anos de doutorado do aluno.

Palavras-Chave: Otimização Combinatória, Pesquisa Operacional, Problema do Corte de Estoque, Problema do Empacotamento, Problema da Mochila.

[1]Email: glebbyo@ic.unicamp.br

[2]Email: rafael@ic.unicamp.br

[3]Email: fkm@ic.unicamp.br

1 Introdução e justificativa

Nesse projeto de pesquisa para o doutoramento de Yulle Glebbyo Felipe Borges, no Instituto de Computação, UNICAMP, pretendemos contribuir para o avanço científico nas áreas de Otimização Combinatória e Pesquisa Operacional focando, em particular, em problemas de corte e empacotamento. A classe de problemas de corte e empacotamento engloba vários problemas encontrados em diversas indústrias, portanto o principal objetivo deste projeto é estudar alguns destes problemas sob abordagens práticas, sejam elas com algoritmos exatos ou heurísticos. Nesta seção, introduzimos alguns dos principais problemas da área e fazemos uma breve revisão bibliográfica sobre cada um deles. Em seguida, apresentamos pontos que acreditamos que justificam o estudo destes problemas no escopo deste projeto de doutorado.

O restante do texto é organizado da seguinte forma: Na Seção 2, descrevemos algumas abordagens que pretendemos utilizar para resolver os problemas propostos. Na Seção 3, apresentamos alguns problemas de nosso interesse, o estado da arte de cada um deles e os resultados que obtivemos em cada problema até então. Na Seção 4 apresentamos brevemente a metodologia de pesquisa utilizada no desenvolvimento deste projeto. Na Seção 5, detalhamos nosso plano de trabalho e um cronograma para o restante do projeto. Por fim, na Seção 6 apresentamos como é feita a análise dos resultados produzidos.

1.1 Preliminares

No **Problema do Corte de Estoque**, dadas barras de comprimento L e um conjunto de itens $\mathcal{I} = \{1, \ldots, n\}$ com comprimentos $l_1, \ldots, l_n$ e demandas $d_1, \ldots, d_n$, queremos cortar o menor número possível de barras com o objetivo de suprir a demanda de cada item. Isto é, queremos atribuir cada item $i \in \{1, \ldots, n\}$ a d_i barras (possivelmente com repetições), de forma que a soma dos comprimentos dos itens atribuídos a cada barra seja no máximo L e de forma que o número de barras com pelo menos um item atribuído seja mínimo. De maneira semelhante, podemos considerar que temos recipientes de comprimento L e desejamos empacotar os itens dentro destes recipientes.

Chamamos de **Problema do Empacotamento** (ou Problema do Corte de Estoque Binário) o caso particular do Problema do Corte de Estoque onde cada item i tem uma demanda $d_i = 1$. De fato, é possível reduzir o Problema do Corte de Estoque ao Problema do Empacotamento considerando d_i cópias do item i, porém, na prática, isto não é interessante de ser feito pois aumenta o número de itens na instância de maneira pseudo-polinomial além de criar simetrias no espaço de soluções. De forma geral, chamamos de **problemas de corte e empacotamento** os problemas com características parecidas com esses dois problemas, isto é, onde desejamos cortar (empacotar) itens menores a partir de (em) um objeto maior [25].

O Problema do Corte de Estoque é um problema clássico introduzido por Kantorovich [44] e Brooks et al. [11] no fim dos anos 30 e início dos anos 40 para modelar situações que ocorriam na indústria. De fato, problemas de corte e empacotamento são muito comuns na indústria, já que, em muitas aplicações, faz-se necessário cortar materiais (metal [24], vidro [56], papel [43], etc) em itens menores para se atender uma determinada demanda.

Apesar de muito comuns, mesmo as versões aparentemente mais simples de problemas de corte e empacotamento escondem uma grande complexidade. Por exemplo, o Problema do Empacotamento é um problema computacionalmente difícil, mais especificamente NP-difícil [34]. Com isso, não existem algoritmos que encontram soluções ótimas em tempo polinomial para este problema, a menos que a P=NP.

Em 1939, Kantorovich [44] apresentou a primeira formulação linear para o Problema do Corte de Estoque, no mesmo artigo em que o problema foi proposto. Posteriormente, em 1961, Gilmore e Gomory [36] propuseram uma nova formulação linear que se baseava nos possíveis padrões de corte para resolver o Problema do Corte de Estoque. Um padrão de corte é definido como um conjunto de itens que podem ser cortados em conjunto a partir de uma única peça de material de estoque. Como pode existir um número exponencial de padrões de corte, esta formulação pode apresentar um número de variáveis exponencial. Os autores consideraram a geração de colunas (variáveis) da formulação de maneira dinâmica e, posteriormente, arredondavam a solução fracionária encontrada. Em outro artigo, Gilmore e Gomory [37] sugeriram alterações no algoritmo para melhorar sua eficiência e lidar com restrições como, por exemplo, a limitação no número de facas no corte de papel. Em 1995, Scheithauer e Terno [61] conjecturaram que esta formulação tem a *propriedade do arredondamento inteiro modificada*, isto é, que o valor de qualquer solução ótima inteira é no máximo o valor da relaxação linear de tal formulação arredondado para cima mais um. Assim, se a conjectura for verdadeira, então a relaxação linear desta formulação é um excelente limitante inferior para o valor de uma solução ótima. De fato, a maior diferença conhecida entre o valor da relaxação e de uma solução ótima inteira é de 1,1666... [58].

Apesar da relaxação linear da formulação de Gilmore e Gomory oferecer um bom limitante inferior, o esforço computacional requerido para resolver este modelo pode ser muito alto devido ao número exponencial de variáveis. Tendo isto em mente, outras abordagens com foco em modelos mais compactos foram desenvolvidas na literatura, entre elas estão os modelos com número de variáveis e restrições pseudo-polinomial.

O primeiro destes modelos foi o *One-Cut*, proposto por Dyckhoff [26], que se baseia em dividir um recipiente em duas partes: a primeira consiste de um item já empacotado; e a segunda consiste ou de um outro item ou de um resíduo que pode ser dividido novamente ou utilizado para empacotar outro item. Esta divisão é feita recursivamente para todo resíduo ou novos recipientes até que todos os itens tenham sido empacotados. Posteriormente, talvez o mais importante dos modelos pseudo-polinomiais para o Problema do Corte de Estoque, o modelo de *Fluxo em Arcos* foi proposto por Valério de Carvalho [65]. Neste modelo a instância é descrita como um digrafo, no qual os vértices representam uma discretização dos possíveis tamanhos ocupados por itens em um recipiente e existe um arco entre dois vértices caso algum item possa ser empacotado utilizando exatamente o espaço representado por estes dois vértices. Desta forma, o problema pode ser modelado como um problema de fluxo neste grafo adicionado de restrições de demanda para cada item, ou seja, devem haver d_i unidades de fluxo passando por arcos que ligam vértices que representam o tamanho de item l_i. Em seguida tivemos o modelo proposto por Cambazard e O'Sullivan [12], *DP-flow*, no qual os estados de uma programação dinâmica de Bellman [7] para o Problema da Mochila (0-1) são representados como vértices de um grafo, e um caminho do vértice inicial (considerando o primeiro item e tamanho de recipiente 0) até o vértice final (considerando n items e tamanho de recipiente L) representam um empacotamento viável de um conjunto de itens. De uma maneira semelhante a da formulação de fluxo em arcos, esta formulação modela a escolha do menor número de caminhos do vértice inicial até o vértice final de forma que todas as demandas sejam satisfeitas. Por fim, temos o modelo de Brandão e Pedroso [10], *Fluxo em Arcos Generalizado com Compressão de Grafo*, que expande o modelo de fluxo em arcos para uso em um multigrafo, e utiliza diversas técnicas de compressão de grafos para diminuir o número de vértices e arcos inclusos na instância.

Apesar da qualidade do limitante inferior dado por estas formulações, o seu uso aplicado com um resolvedor de programação linear inteira baseado em algoritmos de *Branch-and-Bound* - que enumera os valores das variáveis inteiras e resolve uma relaxação linear do modelo em cada nó desta árvore de enumeração, fazendo podas sempre que possível - na busca por uma solução ótima inteira pode ser inviabilizado. Esta inviabilidade pode ser imposta pelo número de variáveis e restrições do modelo (como no caso dos modelos pseudo-polinomiais) ou pelo fato de que a simples solução do modelo por um algoritmo de *Branch-and-Bound* não é o suficiente para garantir otimalidade (como no caso dos modelos baseados em geração de colunas). Assim, vários algoritmos de *Branch-and-Price* - nos quais geração de colunas é aplicada não só na raiz, mas em todos os nós da árvore de *Branch-and-Bound* - foram propostos com base nestes modelos.

O primeiro trabalho nesta linha para o Problema do Empacotamento foi o de Vance et al. [67], que se baseia no modelo de Gilmore e Gomory e define uma regra de *branching* que permitia a geração de colunas de maneira independente em cada nó da árvore de *Branch-and-Bound*. Posteriormente, Vance [66] utilizou o modelo de Gilmore e Gomory e a formulação obtida do modelo de Kantorovich através da decomposição de Dantzig-Wolfe [20] por convexificação e técnicas de *Branch-and-Price* para encontrar soluções inteiras para o Problema do Corte de Estoque. A autora fez uma comparação experimental do desempenho das duas formulações. Vanderbeck [68] apresentou um algoritmo no qual a regra de *branching* se baseia em um conjunto de padrões viáveis. Em alguns nós da árvore de *Branch-and-Bound*, o *pricing* poderia ficar muito difícil, por isto o autor propôs o uso de planos de corte em certos nós para acelerar o tempo de convergência do algoritmo. Por isso, este algoritmo pode ser considerado um *Branch-and-Cut-and-Price*. Dadas as várias formas diferentes de se fazer o *branching* neste caso, Vanderbeck [68] comparou algumas destas, dando destaque a uma regra baseada na representação binária dos padrões de colunas do modelo de Gilmore e Gomory. Valério de Carvalho [65] propôs um algoritmo de *Branch-and-Price* que se baseia na formulação de fluxo em arcos onde o *pricing* é uma variação do Problema da Mochila e o *branching* é feito nas variáveis de fluxo de valor fracionário. Recentemente, Delorme et al. [22] apresentam uma *survey* detalhada sobre o Problema do Empacotamento e o Problema do Corte de Estoque sob uma perspectiva de algoritmos e modelos matemáticos exatos, apresentando experimentos extensíveis e implementações de todos os algoritmos mencionados.

O Problema do Empacotamento também foi muito importante para o estudo de algoritmos de aproximação, desde o notável trabalho de doutoramento de David S. Johnson em 1973, no qual ele demonstrou fatores de aproximação para heurísticas simples baseadas na ordenação dos itens [42]. Desde então, vários estudos sob o ponto de vista de algoritmos de aproximação surgiram para o Problema do Empacotamento. Coffman et al. [18] recentemente fizeram uma detalhada *survey* desta área.

Em relação a heurísticas, Wascher e Gau [72] fizeram um extenso estudo computacional comparando várias heurísticas baseadas em algum tipo de arredondamento da relaxação linear do modelo de Gilmore e Gomory para o Problema do Corte de Estoque. Todas as heurísticas apresentadas neste trabalho podem ser consideradas algoritmos de busca na vizinhança da solução ótima do programa linear (não inteiro). Em outro trabalho, Wascher [73] apresentou uma *survey* de problemas de corte e empacotamento, enquanto que Cheng et al. [15] apresentou uma *survey* sobre o Problema do Corte de Estoque em específico.

Fortemente relacionado ao Problema do Empacotamento, temos o **Problema da Mochila** (binário) onde, dado um recipiente de capacidade L e n itens com comprimentos $l_1, \ldots, l_n$ e valores $v_1, \ldots, v_n$, queremos encontrar um subconjunto de itens S e um empacotamento destes itens no recipiente (isto é, $\sum_{i \in S} l_i \leq L$) que maximize $\sum_{i \in S} v_i$. Uma interpretação comum para tal problema é que temos uma mochila de comprimento L

na qual desejamos guardar os itens, porém não sendo possível armazenar todos os itens, gostaríamos de maximizar a soma dos valores dos itens armazenados. Aplicações para o Problema da Mochila incluem o carregamento de carga, a escolha de investimentos e, até mesmo, o uso de teoria dos grupos em programação inteira [59].

Talvez o algoritmo mais famoso para o Problema da Mochila seja a programação dinâmica atribuída a Bellman [7]. Porém, existem diversos algoritmos para este problema, tais como algoritmos de *Branch-and-Bound*, enumerações e heurísticas, já que trata-se um problema clássico e importante para a Otimização Combinatória e para a Pesquisa Operacional [59]. Algumas *surveys* sobre o Problema da Mochila e variantes incluem a de Salkin e De Kluyver [59], a de Lin [48], a de Fréville [33] e a de Lust e Teghem [50].

Uma generalização interessante do problema do empacotamento consiste em empacotar vetores d-dimensionais em recipientes de mesma dimensão. No **Problema do Empacotamento de Vetores**, dado um recipiente de d recursos, cada um com capacidade 1 (capacidade do recipiente é representada pelo vetor 1^d) e n items cada um representado por um vetor d-dimensional de reais $[0,1]^d$ onde cada dimensão do vetor representa o custo do item em relação ao recurso correspondente, o objetivo é empacotar todos os items na menor quantidade de recipientes possível de forma que a soma dos custos dos items em cada recipiente seja menor ou igual a 1 para todas as dimensões. Se o número de dimensões d deste problema for 1 temos exatamente o Problema do Empacotamento, portanto o Problema do Empacotamento de Vetores é uma generalização do mesmo.

Este problema possui várias aplicações em alocação de recursos como atribuição de máquinas virtuais [55] e otimização de *queries* paralelas [13]. Woeginger [75] mostrou que, para $d \geq 2$, não existe um esquema de aproximação assintótico em tempo polinomial para o Problema do Empacotamento de Vetores a menos que P=NP. Chekuri e Khanna [13] apresentaram algoritmos de aproximação para este problema, enquanto Panigrahy et al. [55] apresentaram uma análise empírica de diversas heurísticas. Brandão e Pedroso apresentam um formulação pseudo polinomial baseado em uma generalização do modelo de fluxo em arcos para o Problema do Empacotamento de Vetores [10].

Problemas de corte e empacotamento também podem ser generalizados para suas respectivas versões geométricas com items bi-dimensionais (ou mesmo tri-dimensionais). Podemos pensar que os recipientes são quadrados e os itens são retângulos. Nesse caso, um empacotamento não inclui apenas quais itens estão alocados em quais caixas, mas também precisa especificar o posicionamento de tais itens de forma que todos os itens estejam dentro de um recipiente e garantir que não haja interseção entre o interior de um item com o interior de outro item ou com as bordas do recipiente. Para uma revisão da literatura para problemas de corte e empacotamento bidimensional, veja [49].

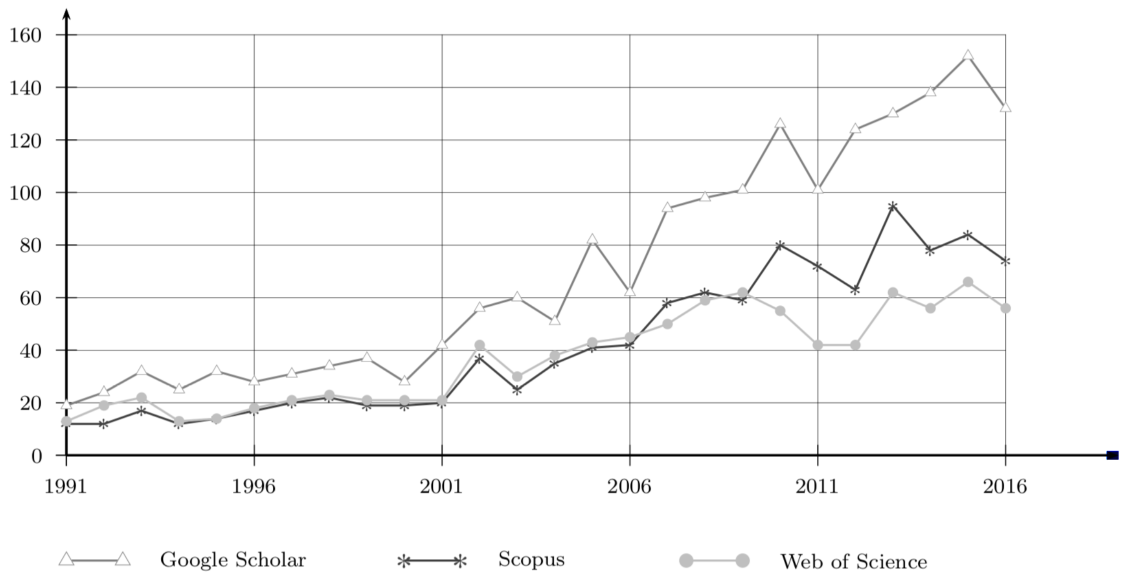

Figura 1: Número de artigos científicos relacionados aos problemas de corte de estoque ou empacotamento de acordo com Google Scholar, Scopus e Web of Science entre 1991 e 2016 [21].

1.2 Justificativa

A importância do projeto de algoritmos para problemas de empacotamento e corte de estoque se dá pelo grande número de aplicações diretas nos setores de indústria e de serviços, já que a otimização de processos de logística nestes setores podem levar a menores custos. A logística é responsável por uma grande parcela dos custos embutidos no preço dos produtos e serviços, assim soluções melhores para problemas de corte e empacotamento podem tornar preços mais baixos e produtos mais competitivos. De acordo com relatório do Banco Mundial, estima-se que os custos na área de Logística no Brasil representaram 26% do PIB em 2008. Por outro lado, nos EUA estes custos foram estimados em 9,5% do PIB [57]. Outras áreas com impacto econômico podem apresentar uma diferença semelhante, levando a produtos e serviços mais caros e com baixa competitividade, frente ao produzido e oferecido pelas nações mais desenvolvidas. Percebemos também um grande interesse da literatura em tais problemas, principalmente nos últimos anos. Em sua tese de doutoramento, Maxim Delorme [21] levantou dados sobre artigos científicos que incluíssem em seu título os termos “bin packing”, “cutting stock” ou ambos de acordo com diferentes fontes bibliográficas entre os anos de 1991 e 2016. Na Figura 1 (extraída de [21]), podemos verificar o aumento do interesse nestes temas. Assim, é importante buscar novos resultados para tais problemas e relacionados, validando-os pela publicação em bons veículos de divulgação internacional.

É necessário resolver problemas de corte e empacotamento, obtendo soluções ótimas ou próximas de soluções ótimas, dentre de um limite de tempo aceitável, já que, na prática, é comum termos urgência no cálculo da solução. Devida a complexidade dos problemas considerados neste projeto, faz-se necessário propor novos algoritmos e adaptar técnicas já utilizadas com sucesso em outros problemas para obter novos patamares na resolução de tais problemas.

Além disso, é importante a formação de novos pesquisadores em território nacional que sejam capazes de abordar tais problemas com sucesso, seja na industria ou na academia. Em particular, acreditamos que seja interessante continuar a formação do candidato nesta área. O candidato concluiu o mestrado em Ciência da Computação na Universidade Estadual de Campinas em 2016 com o projeto "Problema do Corte de Estoque com Restrições de Classe", financiado pela FAPESP, onde investigou o Problema do Empacotamento com Classes, apresentando algoritmos de *Branch-and-Price* para o mesmo, bem como diversos algoritmos para o Problema da Mochila com Classes e variantes. Os resultados obtidos durante o mestrado foram submetidos a um periódico internacional, e esperamos uma resposta nos próximos meses. Dentre as disciplinas cursadas no mestrado estão "Algoritmos Probabilísticos" e "Complexidade de Algoritmos I", sendo que o candidato obteve conceito "A" em ambas. Durante o doutorado, o aluno cumpriu todos os pré-requisitos de disciplinas e línguas exigidos pelo programa. Dentre as disciplinas cursadas no doutorado estão "Tópicos em Otimização Combinatória: Pesquisa Operacional", "Teoria dos Jogos Algorítmica" e "Projeto de Redes Multimídia", obtendo conceito "A" em todas elas.

2 Abordagens de Pesquisa

A seguir, apresentamos as abordagens de pesquisa consideradas neste projeto.

2.1 Algoritmos Exatos

Na abordagem de *Algoritmos Exatos* procuramos desenvolver algoritmos que obtenham soluções exatas/ótimas explorando a estrutura do problema para minimizar o processo de enumeração envolvido na busca de uma solução ótima.

A principal estratégia que iremos considerar nesta linha é a de *Programação Linear Inteira*, onde o problema é formulado através de um programa linear onde algumas, ou mesmo todas, variáveis devem ser inteiras (Programa Linear Inteiro - PLI). Nesta estratégia, um algoritmo que usa o método *Branch-and-Bound* faz uso de uma árvore de enumeração implícita na qual a raiz da árvore é a relaxação da formulação do problema em

PLI (obtida removendo as restrições de integralidade das variáveis). Cada nó desta árvore representa um programa linear relaxado. A ramificação de cada nó é feita dividindo-se o espaço de busca através da inserção de novas restrições (tipicamente sobre variáveis fracionárias do programa linear) e com isso gerando outros nós filhos a partir de um nó da árvore. Sempre que um nó apresentar uma atribuição inviável ou apenas soluções piores ou iguais que alguma já existente, este ramo da árvore é podado. Esta abordagem usa o fato de que programação linear pode ser resolvida rapidamente e a resolução do programa linear associado a um nó pode aproveitar a estrutura da solução de nós anteriores, agilizando assim a busca da solução do nó atual. Além disso, a enumeração pode ser guiada na direção da função objetivo do problema. Métodos mais sofisticados consideram no processo de enumeração a inserção de novas desigualdades válidas para o problema (método *Branch-and-Cut*), a inserção de novas variáveis (método *Branch-and-Price*) ou mesmo a combinação destas duas (método *Branch-and-Cut-and-Price*).

2.1.1 Branch-and-Price

A seguir, ilustramos o uso do método *Branch-and-Price* na resolução do Problema do Empacotamento.

Seja S um subconjunto de itens tal que a soma dos tamanhos dos itens de S é no máximo L. Dizemos que S é um *padrão viável*. Seja $\mathcal{S}$ a família de padrões viáveis e, para cada item i, seja $\mathcal{S}_i$ a família de padrões viáveis que contém o item i. Note que toda solução viável do Problema do Empacotamento para esta instância é, de fato, uma coleção de padrões viáveis onde cada item aparece em algum padrão. Assim, podemos considerar a formulação de Gilmore e Gomory [36] para o problema, onde para cada $S \in \mathcal{S}$, temos uma variável x_S que indica se o padrão viável S está na solução ou não:

$$
\begin{array}{llll}
\text{(P)} & \text{minimize} & \displaystyle\sum_{S \in \mathcal{S}} x_S & \\
& \text{sujeito a} & \displaystyle\sum_{S \in \mathcal{S}_i} x_S \geq 1 & \forall\, 1 \leq i \leq n \qquad (1) \\
& & x_S \in \{0,1\} & \forall\, S \in \mathcal{S},
\end{array}
$$

onde a restrição (1) garante que cada item precisa estar em pelo menos um padrão viável escolhido. Se um item estiver em mais de um padrão, podemos escolher um padrão arbitrário e removê-lo dos outros padrões. Porém, é útil permitir que cada item esteja potencialmente em mais de um padrão.

O dual (D) da relaxação da formulação (P) é o seguinte:

$$
\begin{aligned}
\text{(D)} \quad & \text{maximize} \sum_{i \in \mathcal{I}} \nu_i & & \\
& \text{sujeito a} \quad \sum_{i \in S} \nu_i \leq 1 & & \forall\, S \in \mathcal{S} \\
& \qquad\qquad \nu_i \geq 0 & & \forall\, 1 \leq i \leq n.
\end{aligned}
\tag{2}
$$

Note que temos um número potencialmente exponencial de padrões viáveis e, portanto, de variáveis (P) e de restrições em (D). Porém, podemos resolver (P) pelo método de *Branch-and-Bound* com apenas alguns padrões iniciais, gerando novos padrões pelo processo de *pricing* em cada nó para obter uma solução ótima da relaxação linear de (P). Para encontrar uma nova variável a ser inserida em (P), basta encontrar uma restrição violada de (D), isto é, dado um vetor $\nu_1, \ldots, \nu_n$, decidir se existe um padrão viável S tal que $\sum_{i \in S} \nu_i > 1$, isto é, encontrar uma solução do Problema da Mochila com valor maior do que 1, considerando os mesmos itens porém com os valores dados pelas variáveis ν. Porém, dependendo da forma que for realizado o *branching*, conforme novos nós em diferentes níveis da árvore de *Branch-and-Bound* são criados, os problemas de *pricing* poderão apresentar restrições adicionais como, por exemplo, padrões viáveis que não podem ser considerados como solução da mochila.

2.1.2 Formulações Pseudo Polinomiais

Nesta seção, ilustramos uma formulação pseudo polinomial para resolver o *Problema do Corte de Estoque* através da descrição exemplificada da *Formulação de Fluxo em Arcos* proposta por Valério de Carvalho [65].

Dados recipientes de capacidade L e um conjunto de itens de tamanhos $l_1, \ldots, l_n$, quando os tamanhos dos itens e capacidade dos recipientes tem valores inteiros, podemos descrever o problema de determinar um subconjunto destes itens que podem ser empacotados em um único recipiente como o problema de encontrar um caminho em um digrafo acíclico com $L + 1$ vértices. Seja o digrafo $G = (V, A)$ com $V = \{0, 1, \ldots, L\}$ e $A = \{(i, j) : 0 \leq i < j \leq L, \exists u \in \mathcal{I} : j - i = l_u\}$, ou em outras palavras, existe um arco entre dois vértices (i, j) se existe algum item de tamanho $j - i$. Por completude, devemos considerar também arcos $(i, i + 1)$ para todo $i = l_{min}, \ldots, L - 1$ que representam porções não ocupadas do recipiente (perdas). Desta forma, podemos descrever um empacotamento viável para um único recipiente como um caminho partindo do vértice 0 até o vértice L, onde as arestas escolhidas neste caminho representam os tamanhos dos itens a serem empacotados. De acordo com Valério de Carvalho é possível reduzir possíveis

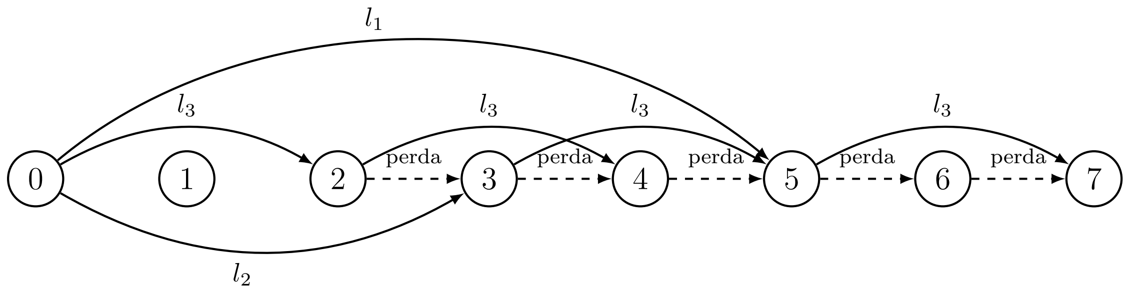

Figura 2: Digrafo de fluxo correspondente a instância de exemplo. Adaptado de [10].

simetrias no modelo se seguirmos a seguinte regra: Se ao procurarmos por uma solução, definirmos uma ordem fixa dos itens de maneira não-crescente de tamanho podemos considerar apenas caminhos nos quais os itens aparecem nesta ordem, consequentemente reduzindo o número de arcos no digrafo.

Considere o seguinte exemplo [10]: Dada uma instância com recipientes de tamanho $L = 7$ com itens de tamanhos $5, 3, 2$ e demandas $3, 1, 2$ respectivamente, construímos o digrafo da Figura 2. Neste grafo, cada caminho de 0 até L corresponde a um possível empacotamento viável na instância correspondente. Podemos perceber que o número de arcos neste grafo é limitado por $O(nL)$.

Com isso, podemos formular o Problema do Corte de Estoque como o de determinar o fluxo mínimo do vértice 0 até o vértice L adicionado de restrições que garantam a soma dos fluxos nos arcos correspondentes a cada tipo de item seja maior ou igual à demanda deste item. Como a solução considerando um único recipiente corresponde a uma unidade de fluxo entre os vértices 0 até L, um caminho entre 0 e L com fluxo maior que 1 implica na utilização do padrão de empacotamento correspondente mais de uma vez na solução. Utilizamos então, um arco de *feedback* entre o o vértice L e o vértice 0 para contar a quantidade de fluxos que deverá ser usada. Considere então as variáveis de decisão x_{ij} associadas aos arcos do digrafo criado e a variável z como o contador de fluxo passado no arco de *feedback*. Podemos descrever o problema partir do seguinte modelo:

$$
\begin{aligned}
&\text{minimize } z \\
&\text{sujeito a} \quad \sum_{(i,j)\in A} x_{ij} - \sum_{(j,k)\in A} x_{jk} = \begin{cases} -z & \text{para } j = 0, \\ 0 & \text{para } j = 1, \ldots, L-1, \\ z & \text{para } j = L, \end{cases} \\
&\qquad \sum_{(i,j)\in A: j = i + l_u} x_{ij} \geq d_u \qquad \text{para } u = 1, \ldots, n, \\
&\qquad x_{ij} \geq 0 \text{ e inteiro} \qquad \forall\, (i,j) \in A, \\
&\qquad z \geq 0 \text{ e inteiro.}
\end{aligned}
$$

Note que esta formulação modela o Problema do Corte de Estoque, portanto itens de mesmo tamanho podem ser agrupados em um só item com demanda igual a soma das demanda dos itens de mesmo tamanho sem perda de generalidade. Embora este modelo apresente um número de variáveis pseudo polinomial (limitado por $O(nL)$), Valério de Carvalho [65] demonstra que o limitante inferior resultante da relaxação linear é o mesmo do modelo de Gilmore e Gomory que discutimos na Seção 1.

2.2 Algoritmos de Aproximação

Um algoritmo para um problema de otimização pode ser chamado de *Algoritmo de Aproximação* se seu tempo de execução é limitado por um polinômio no tamanho da entrada e a solução produzida tem valor objetivo dentro de um fator garantido do valor da solução ótima para todas as instâncias do problema. Estes algoritmos são úteis para tratar problemas NP-difíceis em casos nos quais seja inviável o uso de algoritmos exatos, porém ainda deseja-se ter alguma garantia sobre a qualidade da solução em relação a valor objetivo.

Um algoritmo de aproximação para um problema de minimização é chamado de uma α-aproximação se, dada uma instância I, o valor objetivo da solução encontrada $A(I)$ é menor ou igual a α vezes o valor objetivo da solução ótima $OPT(I)$, ou seja

$$OPT(I) \leq A(I) \leq \alpha OPT(I).$$

Outra definição importante é a de um *Esquema de Aproximação em Tempo Polinomial* (*Polynomial Time Approximation Scheme*, PTAS), que define uma família de algoritmos de aproximação acompanhados de uma constante $\varepsilon > 0$ tais que

$$OPT(I) \leq A(I) \leq (1+\varepsilon)OPT(I).$$

Neste caso, o algoritmo deve ter tempo de execução polinomial no tamanho da entrada, porém pode ser exponencial em ε. Entretanto, vários problemas de corte e empacotamento pertencem a classe *APX-difícil*, isto é não admitem um PTAS a menos que P=NP. Por isso é comum projetar algoritmos de aproximação com fator de aproximação assintótico, ou seja, $OPT(I) \leq A(I) \leq \alpha OPT(I) + \beta$, onde β é uma constante maior que 0. De maneira semelhante, podemos definir um *Esquema de Aproximação Assintótico em Tempo Polinomial* (*Asymptotic Polynomial Time Approximation Scheme*, APTAS) como uma família de algoritmos de aproximação acompanhados de uma constante $\varepsilon > 0$ e uma constante $\beta > 0$ tais que

$$OPT(I) \leq A(I) \leq (1 + \varepsilon)OPT(I) + \beta.$$

Algoritmos de aproximação são muito utilizados na literatura para tratar problemas de interesse deste projeto [18]. Para mais informações sobre esta abordagem, referimos aos livros de Vazirani [69] e Williamson e Shmoys [74].

2.3 Heurísticas e Meta-heurísticas

Heurísticas são procedimentos que buscam por soluções de boa qualidade, dentro das limitações dos recursos existentes para sua obtenção. No geral, as limitações se devem a restrições da capacidade computacional, de processamento e memória, e do tempo disponível para sua execução. Apesar de não necessariamente terem garantias de se encontrar soluções ótimas ou próximas de soluções ótimas, para muitos problemas de natureza prática, as heurísticas obtém bons resultados com a vantagem de computar uma solução rapidamente.

Já as *meta-heurísticas* são métodos de alto nível, não necessariamente definidas para um problema específico ou com passos bem definidos, mas com métodos que permitem aplicá-los a uma vasta gama de problemas, avançando a busca por novas soluções, seja fazendo modificações nas soluções existentes ou combinando-as na tentativa de se construir soluções que herdem as melhores características das soluções existentes. Algumas das principais meta-heurísticas são as de busca tabu, GRASP, *simulated annealing*, algoritmos genéticos, busca em vizinhança variável (VNS), algoritmos meméticos, BRKGA's, colônia de formigas, entre outros. Para informações detalhadas sobre várias destas meta-heurísticas, referimos ao livro de Gendreau e Potvin [35].

Apesar da ideia de se combinar heurísticas com programação matemática ser já antiga e bastante comum, até pouco tempo atrás, programação matemática tinha pouco a ver com meta-heurísticas e as principais meta-heurísticas (como as citadas anteriormente) tinham pouco ou quase nada de programação matemática [52]. Assim, surgiram as *Matheuristicas* que combinam meta-heurísticas e programação matemática, um conceito que pode trazer novos resultados interessantes para a solução de problemas de corte e empacotamento, além de muitos outros problemas de pesquisa operacional e otimização combinatória.

3 Problemas Investigados

A seguir, apresentamos problemas de corte e empacotamento nos quais estamos interessados. Porém, acreditamos que, ao decorrer do doutorado do candidato, poderemos nos deparar com outros problemas de corte e empacotamento interessantes de serem abordados. Assim, a lista a seguir demonstra a essência dos problemas abordados neste projeto, mas não é exaustiva e nem definitiva.

3.1 Gerenciamento no Lado-da-demanda em *Smart Grids*

Smart grids podem ser definidas como redes de eletricidade que são capazes de integrar as ações de todos os agentes conectados a ela de maneira inteligente, sejam os agentes geradores, consumidores ou mesmo ambos. Com isso, consideramos o *Gerenciamento no Lado-da-demanda* (*Demand-side Management*, DSM) como um conjunto de técnicas implementadas pelas empresas prestadoras de serviços com o intuito de influenciar o consumo de energia de seus usuários com o objetivo de atingir um uso mais eficiente da rede em relação a sua capacidade. Dentre estas técnicas, temos *smart pricing* na qual a empresa prestadora de serviços ajusta o preço da energia de acordo com o consumo agregado de todos os usuários da rede, encorajando alguns usuários a deslocar parte de seu consumo para horários fora do pico de consumo.

Dada a complexidade e o caráter aberto deste problema, vários modelos foram propostos na literatura. Ao melhor do nosso conhecimento, o primeiro artigo a considerar *smart pricing* assim como técnicas de teoria dos jogos para DSM foi o de Fahrioglu e Alvarado [29], no qual consideraram o uso das informações de demando dos clientes para construir contratos de uso que fossem benéficos para a rede. Posteriormente, Mohsenian-Rad et al. [54] propuseram um modelo para DSM na forma de um jogo iterativo entre os consumidores e a empresa prestadora de serviços. Samadi et al. [60] propõem um modelo de *smart pricing* utilizando um mecanismo de Vickrey-Clarke-Groves (VCG) [70, 17, 39]. Neste contexto, o mecanismo VCG pode ser entendido como um tipo de leilão em que os agentes são incentivados financeiramente a reportar os valores corretos para seus lances, assim podemos dizer que um mecanismo deste tipo é a prova de estratégia uma vez que reportar valores estratégicos não traz nenhum benefício aos agentes. No trabalho de Samadi et al. [60] os usuários devem reportar suas demandas para a empresa prestadora de serviço, que calcula os preços individuais de cada usuário a partir do consumo agregado da rede. Ma et al. [51] propuseram um mecanismo melhorado de Arrow-d'.Aspremont-Gerard-Varet [2, 19] para o modelo de Samadi et al. [60] que mantém a propriedade de ser a prova de estratégia.

Vários outros artigos consideraram DSM em *smart grids* porém sob pontos de vista diferentes. Como exemplo destes pontos de vista temos: a inclusão de dispositivos de armazenamento de energia na rede [3, 64, 14, 79, 71]; a modelagem da natureza estocástica na geração de energia renovável [4, 80]; e a modelagem do desconforto dos consumidores na função de utilidade [47, 71]

Nossa contribuição nesta área foi a partir de uma colaboração com o Prof. Nelson L.S. da Fonseca, o aluno de doutorado Lucas P. Melo do Instituto de Computação da Unicamp e o Prof. Fabrizio Granelli do Departamento de Engenharia da Informação e Ciência da Computação da Universidade de Trento na Itália. Consideramos um cenário no qual usinas de energia convencionais são combinadas com usinas de energia renovável para suprir uma pequena comunidade utilizando majoritariamente energia renovável.

Propomos um jogo baseado no mecanismo VCG, no qual a empresa prestadora de serviços propaga para todos usuários da rede um conjunto de planos de consumo. Cada usuário deve oferecer um lance para cada plano de consumo oferecido de acordo com a utilidade monetária (individual de cada usuário) que o plano de consumo lhe apresenta. Do lado dos consumidores temos que os lances podem ser computados de maneira independente por cada usuário e o modelo de DSM não depende da forma como os lances são calculados. Entretanto apresentamos um modelo que se baseia na formulação deste problema como uma variante do Problema das Múltiplas Mochilas (versão da mochila do Problema do Empacotamento de Vetores) que pode ser utilizado para a geração de lances ótimos para cada usuário através de um PLI compacto. Do lado da empresa prestadora de serviços temos a atribuição dos planos de consumo aos usuários restrita à capacidade de geração de energia renovável da rede e ao custo de energia não-renovável. Para a alocação dos planos de consumo um modelo centralizado que utiliza os valores dos lances de todos os usuários da rede para fazer a alocação que maximize o "bem-estar social" da rede, que por consequência minimiza os picos de consumo considerando as restrições de capacidade de produção de energia. Este modelo pode ser descrito como outra variante do Problema das Múltiplas Mochilas, modelado como um PLI. Demonstramos também que qualquer solução ótima do modelo de alocação de energia apresenta uma solução justa para os usuários da rede.

Também apresentamos uma extensão de nosso modelo para considerar equipamentos de armazenamento de energia renovável tanto no lado dos consumidores quanto no lado do produtor.

Realizamos simulações numéricas a partir de dados de geração de energia renovável da região CSUD da Itália, e uma extensão do modelo de representação de uso de diversos eletrodomésticos utilizado por Samadi et al. [60]. Com esta extensão, conseguimos representar um número maior de situações comuns no mundo real, como por exemplo a recarga de carros elétricos. Os resultados das simulações mostram que nosso modelo, além de minimizar os picos de consumo, motiva os usuários a consumir mais energia renovável e a utilizar mais eletrodomésticos quando comparado com um modelo de precificação que varia com o tempo.

Os resultados obtidos deste trabalho foram compilados em um artigo que foi submetido a um periódico internacional da área de pesquisa operacional, e atualmente, se encontra em fase de avaliação.

3.2 Problema do Empacotamento com Cenários

O **Problema do Empacotamento com Cenários** é uma generalização do Problema do Empacotamento no qual além de um conjunto de itens $\mathcal{I}$ e recipientes de tamanho L, temos também um conjunto $\mathcal{K} = \{1, \ldots, d\}$ de cenários. Para cada cenário $k \in \mathcal{K}$, há um subconjunto de itens que pertencem ao mesmo, que denotamos como S_k, de forma que $\bigcup_{k \in \mathcal{K}} S_k = \mathcal{I}$. Nesta versão do problema, devemos encontrar um empacotamento de $\mathcal{I}$ na forma $\mathcal{B} = \{B_1, B_2, \ldots, B_r\}$ tal que para cada $k \in \mathcal{K}$ e $1 \leq b \leq r$, $\sum_{i \in B_b \cap S_k} s_i \leq L$, ou seja, para cada cenário consideramos apenas o tamanho dos itens pertencentes a este cenário na restrição da mochila, assim todo empacotamento deve ser viável apenas individualmente para cada cenário, já que na prática, apenas um dos possíveis cenários se concretiza. O objetivo é minimizar o número de recipientes para o pior cenário, isto é, minimizar

$$\max_{k \in \mathcal{K}} |\{b : B_b \cap S_k \neq \emptyset\}|.$$

Bodis e Balogh [8] propuseram este problema em meados de 2018. Embora vários problemas de otimização combinatória tenham sido estudados de um ponto de vista de cenários para modelar a incerteza na demanda [30, 31], ao melhor de nosso conhecimento, este foi o primeiro trabalho a considerar o Problema do Empacotamento neste contexto. Embora os autores tenham proposto o problema de modo geral, os resultados apresentados são apenas para a sua versão *online*, na qual não há conhecimento prévio sobre a lista de itens, uma vez que eles chegam em sequencia para a entrada. Foram propostas algumas funções objetivo alternativas para o problema, assim como a análise de competitividade de adaptações de alguns algoritmos *online* bem conhecidos na literatura do Problema do Empacotamento.

Este problema tem sua aplicação por exemplo, no processo de garantia de qualidade no processo de produção. Quando alguma falha é descoberta em um lote de produção após sua entrega, a empresa deve fazer o *recall* destes produtos. Isso pode ser alcançado através da minimização do número de regiões para onde produtos de um mesmo lote são entregues. Do ponto de vista do Problema do Empacotamento com Cenários, os itens pode ser vistos como os produtos, que são organizados de acordo com os lotes de produção a partir do qual eles são manufaturados, que podem ser modelados aqui como cenários. Os recipientes são os armazéns regionais onde os produtos são armazenados antes da entrega. Assim, o objetivo será minimizar o número de armazéns regionais nos quais um possível *recall* será necessário [8].

Acreditamos que exista uma demanda na literatura para o estudo mais aprofundado deste problema do ponto de vista *offline*. É viável o estudo deste problema a partir de abordagens heurísticas e exatas, dado sua motivação prática. Além disso, este problema se mostra interessante do ponto de vista teórico, o que o torna susceptível para o estudo sob uma abordagem de algoritmos de aproximação.

Nossa contribuição para este problema foi a partir de uma colaboração com o prof. Thiago A. de Queiroz do Instituto de Matemática e Tecnologia da Universidade Federal de Goiás e o prof. Lehilton L. C. Pedrosa e o aluno de doutorado Vinícius L. de Lima do Instituto de Computação da Unicamp. Consideramos a versão *offline* do problema sob abordagens de algoritmos de aproximação, algoritmos exatos e meta-heurísticas.

Projetamos um algoritmo de aproximação para o Problema do Empacotamento com Cenários através de uma redução do Problema do Empacotamento de Vetores. Nesta redução, cada cenário é representado através de uma dimensão do vetor a ser empacotado, assim teremos itens como vetores de d dimensões e o tamanho do item i na k-ésima dimensão é l_i se o item pertence ao k-ésimo cenário e 0 caso contrário. Os recipientes permanecem com tamanho L em todas as dimensões. Com isso, conseguimos mostrar que se existe uma α-aproximação para o Problema do Empacotamento de Vetores então existe uma uma $\sqrt{d}\alpha$-aproximação para o Problema de Empacotamento com Cenários. Mostramos também que este fator é de fato justo.

Projetamos também um Esquema de Aproximação Assintótico em Tempo Polinomial para o caso onde o número de cenários d é uma constante inteira maior que zero. Este resultado se baseia em técnicas bem conhecidas para obter esquemas de aproximação assintóticos para problemas de empacotamento, como a divisão dos itens entre grandes e pequenos, que limita o número de recipientes necessário para empacotar os itens grandes. Para o empacotamento dos itens pequenos, entretanto, foi necessário desenvolver um método específico para o Problema do Empacotamento com Cenários.

Para a parte experimental de nosso trabalho, fizemos um modelo PLI baseado na formulação de Gilmore e Gomory para o Problema de Empacotamento e uma meta-heurística baseada em VNS. No modelo PLI, propomos um algoritmo de *Branch-and-Price* utilizando *Dual Feasible Functions* para melhorar os limitantes inferiores durante o processo. A meta-heurística se baseia em operações de troca de itens de recipientes e reempacotamento de recipientes através de uma heurística de *first-fit*. Experimentos preliminares mostram que o modelo exato consegue encontrar soluções ótimas para instâncias de até

50 itens, e apresenta um *gap* pequeno (2 a 3%) para instâncias de 100 itens. Ainda estamos em fase de experimentos para a meta-heurística, portanto não temos resultados concretos para apresentar. Estamos escrevendo um artigo relatando estes resultados e pretendemos, assim que terminamos os experimentos e análises, submetê-lo para um periódico internacional.

3.3 Problema do Empacotamento Colorido

O **Problema do Empacotamento Colorido** é uma generalização do Problema do Empacotamento onde cada item i tem uma cor c_i e não podemos empacotar dois itens de cores iguais lado a lado no mesmo recipiente.

Tal problema tem aplicação, por exemplo, na alocação de intervalos comerciais em programas de televisão. Neste caso, podemos considerar as janelas de programação (por exemplo, de duração de uma hora) como os recipientes, os blocos dos programas como itens de cor branca, com o seu tamanho dado pela duração do bloco, e os intervalos comerciais como itens de cor preta, novamente com o seu tamanho dado pela duração do bloco. Neste caso, queremos intercalar blocos de programa e intervalos, utilizando o menor número possível de janelas de programação para exibir o conteúdo. Podemos considerar também tal problema no contexto de vendas de anúncios *online*, onde temos um *banner* e desejamos alocar propagandas com o objetivo de maximizar o lucro obtido porém com a restrição de que duas propagandas de um mesmo anunciante não apareçam lado a lado para não prejudicar a divulgação do conteúdo.

A versão *online* do Problema do Empacotamento Colorido com duas cores foi introduzida por Balogh et al. [6], onde os autores apresentaram algoritmos *online* e limitantes inferiores para o problema e casos particulares. Balogh et al. [5] posteriormente apresentaram algoritmos de aproximação para a versão *offline* deste problema. Bohm et al [9] consideraram a versão *online* do Problema do Empacotamento Colorido (com múltiplas cores), apresentando algoritmos *online* e limitantes inferiores.

Apesar das motivações práticas para o problema, não existe na literatura nenhum artigo que contenha heurísticas ou algoritmos exatos para o Problema do Empacotamento Colorido. Assim, acreditamos que essa abordagem possa levar a novos resultados para o problema que sejam interessantes para a resolução na prática. Além do Problema do Empacotamento Colorido propriamente dito, podemos considerar o Problema do Corte de Estoque Colorido, o Problema da Mochila Colorida e versões bidimensionais de tais problemas, projetando algoritmos exatos e heurísticas para os mesmos.

Nossa contribuição até então para este problema foi sob uma abordagem prática com algoritmos exatos e meta-heurísticas. Propomos uma formulação de PLI baseada em geração de colunas inspirada no modelo de Gilmore e Gomory. A partir da relaxação linear deste modelo, propomos uma algoritmo simples de arredondamento das variáveis fracionárias através de uma adaptação da heurística FFD para as restrições de cores. Modelos baseados na formulação de Gilmore e Gomory geralmente apresentam limitantes inferiores fortes para problemas similares [27, 53], então utilizamos estes tipo de limitantes para avaliar outros algoritmos propostos. Entretanto, percebemos que a performance de nosso modelo para instâncias grandes era baixa devido à dificuldade dos subproblemas, que foram resolvidos como um modelo PLI compacto através de um *solver* comercial. Por isso projetamos também um algoritmo baseado em BRKGA com codificação por ordenação dos itens, que seriam empacotados utilizando uma adaptação da heurística *First Fit*. Como testes preliminares, comparamos a heurística de arredondamento e o BRKGA utilizando um conjunto de instâncias adaptado de um trabalho anterior [38] e observamos que o BRKGA tem um desempenho melhor para maioria das instâncias testadas, apresentando *gap* relativamente pequeno para todas elas.

Pretendemos reformular o problema utilizando algumas características combinatórias para obter um modelo com subproblemas mais compacto. Com isso, planejamos explorar algoritmos exatos para a versão colorida do problema da mochila para viabilizar o estudo de uma abordagem de *Branch-and-Price* para o problema de empacotamento colorido.

3.4 Problema da Mochila Compartimentada

No **Problema da Mochila Compartimentada**, dados n itens de comprimentos $l_1, \dots, l_n$, valor $v_1, \dots, v_n$, quantidade $\rho_1, \dots, \rho_n$ e classe $c_1, \dots, c_n$, onde $c_i \in \{1, \dots, Q\}$ para todo $1 \leq i \leq n$, desejamos empacotar um subconjunto dos itens em uma mochila de capacidade L de forma que cada item i empacotado fique em um compartimento do tipo c_i. Um compartimento de tipo c é, por si só, uma mochila que induz uma perda de espaço S_c na mochila original, tem um custo k_c para ser instalado e apresenta um limitante inferior $L^c_{\min}$ e um limitante superior $L^c_{\max}$ na soma dos tamanhos dos itens empacotados e S_c. O objetivo é maximizar a soma dos valores dos itens empacotados menos o custo dos compartimentos utilizados.

Tal problema tem aplicações, por exemplo, no corte de aço. Em um primeiro estágio, a bobina é cortada em sub-bobinas (os compartimentos) com a mesma grossura da bobina original. Posteriormente, cada sub-bobina tem a sua grossura ajustada para atender a demanda de itens de mesma grossura, mas diferentes da grossura original da bobina. Assim, os itens são classificados por sua grossura, cada corte da bobina em sub-bobinas leva a uma perda de material, cada sub-bobina tem um tamanho máximo e um tamanho mínimo e tem um custo para ter sua grossura diminuída [24].

Para tal problema, Marques e Arenales [24] apresentaram um modelo não-linear inteiro e heurísticas, Hoto et al. [40] apresentaram um método de geração de colunas, e Leão et al. [46] apresentaram formulações e heurísticas baseadas na geração de colunas e Leão [45], apresentou resultados para a versão bidimensional do problema. Recentemente, Inarejos et al. [41] apresentaram o primeiro modelo linear para este problema.

Uma variante deste problema é o **Problema da Mochila com Classes**, onde podemos ter no máximo um número C de compartimentos na mochila, $\rho_i = 1$ para todo item i, $L^c_{\min} = 0$, $L^c_{\max} = \infty$, $k_c = 0$ e $S_c = 0$ para toda classe $c \in \{1, \ldots, Q\}$. De forma análoga, podemos considerar o **Problema do Empacotamento com Classes**.

O Problema do Empacotamento com Classes tem aplicação, por exemplo, em serviços de vídeo sob demanda onde um servidor pode suportar no máximo uma quantidade C de filmes diferentes além de poder atender no máximo L requisições de usuários diferentes. Assim, em tal problema, queremos distribuir cópias dos filmes nos servidores respeitando L e C, atendendo a demanda dos usuários do sistema e utilizando o menor número possível de servidores [78].

Tal problema foi introduzido por Shachnai e Tamir [62], considerando que todos os itens têm o mesmo tamanho, que apresentaram um PTAS para o mesmo e um PTAS dual, isto é, um algoritmo que utiliza a mesma quantidade de recipientes que uma solução ótima, mas tais recipientes têm tamanho $(1+\varepsilon)L$ para o ε escolhido. Eles apresentaram também um FPTAS (*fully polynomial time approximation scheme*) para o Problema da Mochila com Classes (novamente, considerando que todos os itens têm o mesmo tamanho). Shachnai e Tamir [63] apresentaram limitantes inferiores para a versão *online* do Problema do Empacotamento com Classes e para o Problema da Mochila Múltipla com Classes (onde temos mais do que um recipiente) quando todos os itens têm o mesmo tamanho e analisaram a razão de competitividade de alguns algoritmos para os mesmos. Xavier e Miyazawa [78] consideraram o problema com itens de tamanhos potencialmente diferentes e apresentaram uma $(1 + 1/C)$-aproximação assintótica, juntamente com resultados experimentais que mostram que esse algoritmo obtém bons resultados na prática. Eles mostraram também que não existe algoritmo *online* com espaço limitado (*bounded space*) e razão de competitividade constante, além de apresentarem um algoritmo *online* (não é de espaço limitado) 2,75-competitivo. Epstein et al. [28] apresentaram um AFPTAS para o caso onde Q é constante, provaram que não existe APTAS quando Q faz parte da entrada e apresentaram limitante inferiores para a razão de competitividade de algoritmos *online*. Recentemente, Fischer e Röglin [32] apresentaram análises probabilísticas de diversos algoritmos simples para a versão *online* do Problema do Empacotamento com Restrições de Classe.

Durante o mestrado do candidato, em colaboração com o Prof. Eduardo C. Xavier do Instituto de Computação da Unicamp, o candidato, o orientador e o co-orientador projetaram algoritmos exatos baseados no método *Branch-and-Price* para o Problema do Empacotamento com Classes, além de diversos algoritmos exatos para o Problema da Mochila com Classes e variantes que surgem durante o projeto de algoritmos de *Branch-and-Price* para o Problema do Empacotamento com Classes. Tais resultados compõem a dissertação de mestrado do candidato e um artigo que já foi submetido para um periódico internacional.

Considerando o trabalho do candidato, do orientador e do co-orientador no Problema do Empacotamento com Classes, acreditamos ser possível projetar novos algoritmos exatos e heurísticas para o Problema da Mochila Compartimentada em suas versões uni e bi-dimensional, melhorando o estado-da-arte para tal problema, além de considerar outras variantes do problema que possam se mostrar interessantes, tais como versões do Problema do Corte de Estoque e do Empacotamento com compartimentos.

4 Metodologia

O projeto envolve não somente a elaboração de novos algoritmos exatos e heurísticas para os problemas considerados, mas também a implementação de tais algoritmos para que eles possam ser testados empiricamente. Assim, será necessário o uso de servidores de processamento para realizar tais testes, já que uma simples bateria de testes pode durar dias. Felizmente, o *Laboratório de Otimização e Combinatória* do Instituto de Computação da UNICAMP dispõe de alguns destes servidores e que poderão ser utilizados pelo candidato. O Instituto provê também livros para os seus alunos em parceria com a biblioteca do Instituto de Matemática, Estatística e Computação Científica da UNICAMP, sendo que muitos dos livros mais relevantes da área na qual esse projeto se insere está disponível para consulta dentro do próprio Laboratório de Otimização e Combinatória. Além disso, o Laboratório em si oferece espaço, computadores e impressoras para os alunos membros, que será o caso do candidato durante o doutorado.

Durante a execução do projeto, o candidato cursou disciplinas do Instituto de Computação para consolidar sua formação como pesquisador na área, além de estudar livros e artigos importantes de áreas como, por exemplo, programação linear [16], programação linear inteira [76, 77], algoritmos de aproximação [69, 74], e metaheurísticas [35]. Observamos que ainda durante o seu mestrado, o candidato já cursou disciplinas com notas que permitem sua dispensa no Exame de Qualificação Geral de Doutorado na área de Teoria da Computação. O candidato continua estudando também artigos recentes relacionados aos problemas de pesquisa considerados e relacionado a problemas de empacotamento e corte de estoque, bem como artigos sobre outros problemas de Otimização Combinatória e Pesquisa Operacional, com o objetivo que possa se aprofundar em tais conteúdos e traçar paralelos entre a pesquisa que realizaremos e o que existe na literatura. Além disso, planejamos reuniões semanais entre o candidato, o orientador e o co-orientador para discutir soluções para os problemas abordados e garantir que o projeto ande com fluidez.

5 Plano de trabalho e cronograma

Planejamos trabalhar em cada um dos problemas discutidos na Seção 3 por pelo menos dois semestres. Para o problema de DSM em *Smart Grids*, concluímos nossas contribuições e estamos esperando o resultado sobre a submissão do artigo. Para o Problema do Empacotamento com Cenários, já estamos finalizando os experimentos e elaborando um artigo para reportar os resultados obtidos. Para o Problema do Empacotamento Colorido, temos apenas resultados iniciais e planejamos trabalhar intensamente neste problema durante este ano. Por fim, para o Problema da Mochila Compartimentada, pretendemos dedicar boa parte do tempo restante de projeto.

Para cada um dos problemas restantes, é necessário realizar um levantamento bibliográfico complementar ao apresentado neste projeto, em busca de novas técnicas que possam apresentar potencial para cada problema. Após isso devemos projetar os algoritmos, implementá-los e testá-los para medir a sua eficiência. Estas três tarefas estão interligadas e, portanto, podem ser realizadas concomitantemente. Por fim, faremos a escrita dos artigos, que serão submetidos às boas conferências e periódicos internacionais da área, de forma que planejamos trabalhar em cada problema por cerca de um ano.

Ressaltamos também que o aluno já cumpriu as seguintes exigências do Programa de Pós-Graduação: Estágio Docente; disciplinas para o cumprimento de créditos; dispensa do Exame de Qualificação Geral; e proficiência em língua estrangeira. Assim, as únicas exigências restantes são este Exame de Qualificação Específico e a defesa da tese.

Com isso, descrevemos aqui apenas o planejamento com a distribuição de esforços em relação ao restante do período deste projeto, abrangendo apenas os problemas restantes. Separamos o período restante em trimestres e os problemas em: uma fase de estudo de novas técnicas; uma fase de projeto e implementação dos algoritmos; uma fase de testes e experimentos; e uma fase de formalização dos resultados na forma de um artigo. No cronograma da Tabela 1, descrevemos como planejamos dividir nossos esforços em cada uma destas tarefas ao decorrer do restante do projeto.

Tabela 1: Cronograma

	2019				2020			
	1ºT	2ºT	3ºT	4ºT	1ºT	2ºT	3ºT	4ºT
Empacotamento Colorido								
Estudo de Técnicas	•	•						
Implementação	•	•	•					
Experimentos		•	•					
Escrita do Artigo				•				
Mochila Compartimentada								
Estudo de Técnicas				•	•	•		
Implementação					•	•	•	
Experimentos						•	•	
Escrita do Artigo								•
Outras Exigências								
Escrita da Tese							•	•
Defesa								•

6 Forma de análise dos resultados

Os algoritmos propostos serão analisados através da comparação empírica com algoritmos existentes na literatura, analisando a qualidade da solução encontrada e o tempo de processamento. Para tanto, utilizaremos instâncias geradas computacionalmente e instâncias já conhecidas que estão disponíveis para a utilização como, por exemplo, a OR-Library [1], compilada por J. E. Beasley e a BPP-Library [23], compilada e mantida por Delorme et al.. Além disso, os progressos do projeto poderão ser avaliados pela qualidade das publicações geradas em conferências e periódicos com os eventuais resultados obtidos. Tais conferências e periódicos trabalham com uma rigorosa política de revisão por pares, ou seja, a publicação em tais meios já é uma forma de avaliação da qualidade e importância do trabalho realizado.

Referências

[1] OR-library. http://people.brunel.ac.uk/∼mastjjb/jeb/orlib/binpacktwoinfo.html.

[2] Kenneth Joseph Arrow. The Property Rights Doctrine and Demand Revelation Under Incomplete Information, 1977.

[3] I. Atzeni, L. G. Ordóñez, G. Scutari, D. P. Palomar, and J. R. Fonollosa. Noncooperative and cooperative optimization of distributed energy generation and storage in the demand-side of the smart grid. 61(10):2454–2472, 05 2013.

[4] I. Atzeni, L. G. Ordóñez, G. Scutari, D. P. Palomar, and J. R. Fonollosa. Noncooperative day-ahead bidding strategies for demand-side expected cost minimization with real-time adjustments: A gnep approach. 62(9):2397–2412, May 2014.

[5] J. Balogh, J. Békési, G. Dósa, L. Epstein, H. Kellerer, and Z. Tuza. Online results for black and white bin packing. *Theory of Computing Systems*, 56(1):137–155, 2015.

[6] J. Balogh, J. Békési, G. Dosa, H. Kellerer, and Z. Tuza. Black and white bin packing. In *Proceedings of the 10th International Workshop on Approximation and Online Algorithms*, pages 131–144, 2013.

[7] R. Bellman. *Dynamic Programming*. Princeton University Press, 1957.

[8] Attila Bódis and János Balogh. Bin packing problem with scenarios. *Central European Journal of Operations Research*, pages 1–19, 2018.

[9] Martin Böhm, György Dósa, Leah Epstein, Jiří Sgall, and Pavel Veselỳ. Colored bin packing: Online algorithms and lower bounds. *Algorithmica*, 80(1):155–184, 2018.

[10] Filipe Brandao and João Pedro Pedroso. Bin packing and related problems: general arc-flow formulation with graph compression. *Computers & Operations Research*, 69:56–67, 2016.

[11] R. L. Brooks, C. A. B. Smith, A. H. Stone, and W. T. Tutte. The dissection of rectangles into squares. *Duke Mathematical Journal*, 7(1):312–340, 1940.

[12] Hadrien Cambazard and Barry O'Sullivan. Propagating the bin packing constraint using linear programming. In *International Conference on Principles and Practice of Constraint Programming*, pages 129–136. Springer, 2010.

[13] Chandra Chekuri and Sanjeev Khanna. On multi-dimensional packing problems. In *Proceedings of the tenth annual ACM-SIAM symposium on Discrete algorithms*, pages 185–194. Society for Industrial and Applied Mathematics, 1999.

[14] H. Chen, Y. Li, R. H. Y. Louie, and B. Vucetic. Autonomous demand side management based on energy consumption scheduling and instantaneous load billing: An aggregative game approach. 5(4):1744–1754, July 2014.

[15] C. H. Cheng, B. R. Feiring, and T. C. E. Cheng. The cutting stock problem—a survey. *International Journal of Production Economics*, 36(3):291–305, 1994.

[16] V. Chvátal. *Linear Programming.* W. H. Freeman and Company, 1980.

[17] Edward H. Clarke. Multipart Pricing of Public Goods. *Public Choice*, 11:17–33, September 1971.

[18] Edward G. Coffman, János Csirik, Gábor Galambos, Silvano Martello, and Daniele Vigo. Bin packing approximation algorithms: survey and classification. In *Handbook of combinatorial optimization*, pages 455–531. Springer, 2013.

[19] Claude d'. Aspremont and Louis-André Gérard-Varet. Incentives and incomplete information. *Journal of Public Economics*, 11(1):25–45, February 1979.

[20] G. B. Dantzig and P. Wolfe. Decomposition principle for linear programs. *Operations Research*, 8(1):101–111, 1960.

[21] Maxence Delorme. *Mathematical Models and Decomposition Algorithms for Cutting and Packing Problems.* PhD thesis, Università di Bologna, April 2017.

[22] Maxence Delorme, Manuel Iori, and Silvano Martello. Bin packing and cutting stock problems: Mathematical models and exact algorithms. *European Journal of Operational Research*, 255(1):1–20, 2016.

[23] Maxence Delorme, Manuel Iori, and Silvano Martello. Bpplib: a library for bin packing and cutting stock problems. *Optimization Letters*, 12(2):235–250, 2018.

[24] F. do P. Marques and M. N. Arenales. The constrained compartmentalised knapsack problem. *Computers & Operations Research*, 34(7):2109–2129, 2007.

[25] H. Dyckhoff. A typology of cutting and packing problems. *European Journal of Operational Research*, 44(2):145–159, 1990.

[26] Harald Dyckhoff. A new linear programming approach to the cutting stock problem. *Operations Research*, 29(6):1092–1104, 1981.

[27] Samir Elhedhli. Ranking lower bounds for the bin-packing problem. *European Journal of Operational Research*, 160(1):34 – 46, 2005. Applications of Mathematical Programming Models.

[28] L. Epstein, C. Imreh, and A. Levin. Class constrained bin packing revisited. *Theoretical Computer Science*, 411(34–36):3073–3089, 2010.

[29] M. Fahrioglu and F. L. Alvarado. Designing incentive compatible contracts for effective demand management. 15(4):1255–1260, 11 2000.

[30] Esteban Feuerstein, Alberto Marchetti-Spaccamela, Frans Schalekamp, René Sitters, Suzanne van der Ster, Leen Stougie, and Anke van Zuylen. Scheduling over scenarios on two machines. In *International Computing and Combinatorics Conference*, pages 559–571. Springer, 2014.

[31] Esteban Feuerstein, Alberto Marchetti-Spaccamela, Frans Schalekamp, René Sitters, Suzanne van der Ster, Leen Stougie, and Anke van Zuylen. Minimizing worst-case and average-case makespan over scenarios. *Journal of Scheduling*, 20(6):545–555, 2017.

[32] Carsten Fischer and Heiko Röglin. Probabilistic analysis of online (class-constrained) bin packing and bin covering. In *Latin American Symposium on Theoretical Informatics*, pages 461–474. Springer, 2018.

[33] A. Fréville. The multidimensional 0–1 knapsack problem: An overview. *European Journal of Operational Research*, 155(1):1–21, 2004.

[34] M. R. Garey and D. S. Johnson. *Computers and Intractibility: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, 1979.

[35] Michel Gendreau and Jean-Yves Potvin. *Handbook of metaheuristics*, volume 2. Springer, 2010.

[36] P. C. Gilmore and R. E. Gomory. A linear programming approach to the cutting stock problem. 9:849–859, 1961.

[37] P. C. Gilmore and R. E. Gomory. A linear programming approach to the cutting stock problem-part ii. *Operations research*, 11(6):863–888, 1963.

[38] Yulle Glebbyo, Flávio K. Miyazawa, Rafael C. S. Schouery, and Eduardo C. Xavier. Algoritmos branch-and-price para o problema de empacotamento em recipientes com restrições de classe. In *XXXVI Congresso da Sociedade Brasileira de Computação-1º Encontro da Teoria da Computação (ETC)*, pages 820–823, 2016.

[39] T. Groves. Incentives in Teams. *Econometrica*, 41(4):617–631, July 1973.

[40] R. Hoto, N. Maculan, and A. Borssoi. A study of the compartimentalized knapsack problem with additional restrictions. *IEEE Latin America Transactions*, 8(3):269–274, 2010.

[41] Osvaldo Inarejos, Robinson Hoto, and Nelson Maculan. An integer linear optimization model to the compartmentalized knapsack problem. *International Transactions in Operational Research*, 2017.

[42] David S. Johnson. *Near-optimal bin packing algorithms*. PhD thesis, Massachusetts Institute of Technology, 1973.

[43] J. Kallrath, S. Rebennack, J. Kallrath, and R. Kusche. Solving real-world cutting stock-problems in the paper industry: Mathematical approaches, experience and challenges. *European Journal of Operational Research*, 238(1):374–389, 2014.

[44] L. V. Kantorovich. The mathematical method of production planning and organization. *Management Science*, 6:363–422, 1939.

[45] A. A. de S. Leão. *Extensões em problemas de corte: padrões compartimentados e problemas acoplados*. PhD thesis, Universidade de São Paulo, 2013.

[46] A. A. S. Leão, M. O. Santos, R. Hoto, and M. N. Arenales. The constrained compartmentalized knapsack problem: mathematical models and solution methods. *European Journal of Operational Research*, 212(3):455–463, 2011.

[47] C. Li, X. Yu, W. Yu, G. Chen, and J. Wang. Efficient computation for sparse load shifting in demand side management. 8(1):250–261, Jan 2017.

[48] E. Y. Lin. A bibliographical survey on some well-known non-standard knapsack problems. *INFOR*, 36(4):274–317, November 1998.

[49] A. Lodi, S. Martello, and M. Monaci. Two-dimensional packing problems: A survey. *European Journal of Operational Research*, 141(2):241–252, 2002.

[50] T. Lust and J. Teghem. The multiobjective multidimensional knapsack problem: a survey and a new approach. *International Transactions in Operational Research*, 19(4):495–520, 2012.

[51] J. Ma, J. Deng, L. Song, and Z. Han. Incentive mechanism for demand side management in smart grid using auction. 5(3):1379–1388, 05 2014.

[52] V. Maniezzo, T. Stützle, and S. Voß, editors. *Matheuristics: Hybridizing Metaheuristics and Mathematical Programming*, volume 10 of *Annals of Information Systems*. Springer US, 2010.

[53] Odile Marcotte. The cutting stock problem and integer rounding. *Mathematical Programming*, 33(1):82–92, 1985.

[54] A. H. 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. 1(3):320–331, 12 2010.

[55] Rina Panigrahy, Kunal Talwar, Lincoln Uyeda, and Udi Wieder. Heuristics for vector bin packing. *research. microsoft. com*, 2011.

[56] K.T. Park, J. Ryu, H.-K. Lee, and I.-B. Lee. Development of a heuristic algorithm for cutting stock problems in flat glass production processes. *Journal of Chemical Engineering of Japan*, 45(3):219–227, 2012.

[57] K. Rantasila and L. Ojala. Measurement of national-level logistics costs and performance. Technical report, International Transport Forum, 2012. `http://www.internationaltransportforum.org/jtrc/discussionpapers/dp201204.pdf`.

[58] J. Rietz, G. Scheithauer, and J. Terno. Tighter bounds for the gap and non-IRUP constructions in the one-dimensional cutting stock problem. *Optimization*, 6:927–963, 2002.

[59] H. M. Salkin and C. A. De Kluyver. The knapsack problem: a survey. *Naval Research Logistics Quarterly*, 22(1):127–144, 1975.

[60] Pedram Samadi, Hamed Mohsenian-Rad, Robert Schober, and Vincent W S Wong. Advanced Demand Side Management for the Future Smart Grid Using Mechanism Design. 3(3):1170–1180, 09 2012.

[61] G. Scheithauer and J. Terno. The modified integer round-up property of the one-dimensional cutting stock problem. *European Journal of Operational Research*, 84(3):562–571, 1995.

[62] H. Shachnai and T. Tamir. Polynomial time approximation schemes for class-constrained packing problems. *Journal of Scheduling*, 4(6):313–338, 2001.

[63] H. Shachnai and T. Tamir. Tight bounds for online class-constrained packing. *Theoretical Computer Science*, 321(1):103–123, 2004. Latin American Theoretical Informatics.

[64] H. M. Soliman and A. Leon-Garcia. Game-theoretic demand-side management with storage devices for the future smart grid. 5(3):1475–1485, May 2014.

[65] J.M. Valério De Carvalho. Exact solution of bin-packing problems using column generation and branch-and-bound. *Annals of Operations Research*, 86:629–659, 1999.

[66] P. H. Vance. Branch-and-price algorithms for the one-dimensional cutting stock problem. *Computational Optimization and Applications*, 9(3):211–228, 1998.

[67] P. H. Vance, C. Barnhart, E. L. Johnson, and G. L. Nemhauser. Solving binary cutting stock problems by column generation and branch-and-bound. *Computational Optimization and Applications*, 3(2):111–130, 1994.

[68] F. Vanderbeck. Computational study of a column generation algorithm for bin packing and cutting stock problems. *Mathematical Programming*, 86(3):565–594, 1999.

[69] Vijay V. Vazirani. *Approximation algorithms*. Springer Science & Business Media, 2013.

[70] William Vickrey. Counterspeculation, Auctions, and Competitive Sealed Tenders. *The Journal of Finance*, 16(1):8–37, March 1961.

[71] K. Wang, H. Li, S. Maharjan, Y. Zhang, and S. Guo. Green energy scheduling for demand side management in the smart grid. PP(99):1–1, 2018.

[72] G. Wäscher and T. Gau. Heuristics for the integer one-dimensional cutting stock problem: A computational study. *Operations-Research-Spektrum*, 18(3):131–144, 1996.

[73] G. Wäscher, H. Haußner, and H. Schumann. An improved typology of cutting and packing problems. *European Journal of Operational Research*, 183(3):1109–1130, 2007.

[74] David P. Williamson and David B. Shmoys. *The design of approximation algorithms*. Cambridge university press, 2011.

[75] Gerhard J. Woeginger. There is no asymptotic ptas for two-dimensional vector packing. *Information Processing Letters*, 64(6):293–297, 1997.

[76] L. A. Wolsey. *Integer Programming*. Wiley-Interscience publication, 1998.

[77] L. A. Wolsey and G. L. Nemhauser. *Integer and Combinatorial Optimization.* Wiley-Interscience, 1999.

[78] E. C. Xavier and F. K. Miyazawa. The class constrained bin packing problem with applications to video-on-demand. *Theoretical Computer Science*, 393:240–259, 2008.

[79] F. Ye, Y. Qian, and R. Q. Hu. A real-time information based demand-side management system in smart grid. 27(2):329–339, Feb 2016.

[80] J. Zazo, S. Zazo, and S. Valcarcel Macua. Robust worst-case analysis of demand-side management in smart grids. 8(2):662–673, March 2017.