UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO DE CIÊNCIAS EXATAS, NATURAIS E DA SAÚDE
DEPARTAMENTO DE COMPUTAÇÃO

JEFFERSON DE MORAES MISTURA

OBTENÇÃO DE LIMITANTES INFERIORES PARA O PROBLEMA DE TABELA-HORÁRIO DE UNIVERSIDADES UTILIZANDO PARTICIONAMENTO DE GRAFOS

ALEGRE - ES
2022

JEFFERSON DE MORAES MISTURA

OBTENÇÃO DE LIMITANTES INFERIORES PARA O PROBLEMA DE TABELA-HORÁRIO DE UNIVERSIDADES UTILIZANDO PARTICIONAMENTO DE GRAFOS

Trabalho de conclusão de curso apresentado ao Departamento de Computação do Centro de Ciências Exatas, Naturais e da Saúde da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.
Orientador: Prof. Dr. Edmar Hell Kampke

ALEGRE - ES
2022

JEFFERSON DE MORAES MISTURA

OBTENÇÃO DE LIMITANTES INFERIORES PARA O PROBLEMA DE TABELA-HORÁRIO DE UNIVERSIDADES UTILIZANDO PARTICIONAMENTO DE GRAFOS

Trabalho de conclusão de curso apresentado ao Departamento de Computação do Centro de Ciências Exatas, Naturais e da Saúde da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Aprovado em 13 de dezembro de 2023.

COMISSÃO EXAMINADORA

Prof. Dr. Edmar Hell Kampke
Universidade Federal do Espírito Santo
Orientador

Prof. Dr. Geraldo Regis Mauri
Universidade Federal do Espírito Santo

Prof. Dr. Dayan de Castro Bissoli
Universidade Federal do Espírito Santo

AGRADECIMENTOS

Ao concluir esta etapa significativa da minha vida e da minha jornada acadêmica, não posso deixar de expressar minha profunda gratidão a todas as pessoas que contribuíram para o sucesso deste trabalho. Este trabalho representa não apenas o resultado dos meus esforços, mas também a colaboração e apoio inestimáveis de todos que fizeram parte, mesmo indiretamente, da realização deste trabalho.

Primeiramente quero agradecer a minha família, por seu apoio incondicional, paciência e compreensão ao longo deste desafio. Seu incentivo constante foi minha fonte de força nos momentos mais difíceis, mesmo nas horas onde eu não acreditava em mim mesmo. Obrigado por serem a minha base e pilar.

Aos amigos e colegas de curso, agradeço pelas discussões construtivas, trocas de ideias e apoio mútuo. Mesmo que sejam poucos, com certeza essa jornada foi mais distraída e alegre graças a vocês. As experiências compartilhadas tornaram essa jornada mais enriquecedora.

Gostaria também de agradecer imensuravelmente ao meu orientador, Edmar, pela orientação sábia, valiosas sugestões ao longo deste processo e principalmente pela paciência incansável em me orientar. Sua orientação e expertise foram fundamentais para a execução e qualidade deste trabalho. Seu comprometimento é exemplar e seus conselhos são indispensáveis, não somente na execução deste trabalho, mas também durante a jornada acadêmica.

Expresso minha sincera gratidão também a todos os professores e colaboradores do curso, cuja intenção em compartilhar conhecimentos e experiências enriqueceu significativamente minha jornada acadêmica. Ao longo de toda a graduação, cada interação com vocês contribuiu para meu aprendizado de maneira inestimável.

Quero estender meu agradecimento a todos que, de alguma maneira, colaboraram para a realização deste trabalho, mesmo que não tenham sido mencionados individualmente. Cada palavra de incentivo, cada conversa, e cada conselho valioso desempenhou um papel crucial no meu desenvolvimento acadêmico e pessoal.

Agradeço a todos por serem parte desta jornada e por terem sido pilares que me ajudaram a atingir este marco em minha vida acadêmica. Sem a contribuição de vocês, este feito não teria sido possível.

RESUMO

Este trabalho trata do Problema de Tabela-Horário de Universidades (PTHU), que é um dos problemas mais conhecidos da área de Otimização Combinatória (OC). O problema consiste na alocação de aulas em salas disponíveis, sendo que restrições de tempo e disponibilidade dos alunos e professores são levadas em consideração, além de outras restrições previamente definidas. Existem na literatura várias formulações para o PTHU, visto que cada instituição possui diferentes restrições. Este problema é complexo devido ao grande número de variáveis e restrições. Neste trabalho foi considerado o PTHU baseado na formulação de currículos de cursos (CB-CTT) proposta na segunda edição do campeonato internacional de tabela- horário (ITC-2007). Portanto, o objetivo foi usar a abordagem *dividir para conquistar* para a obtenção de limitantes inferiores. Foi utilizada a biblioteca METIS para particionar o problema original, representado por grafos, em subproblemas que foram então resolvidos com o *solver* CPLEX. Experimentos computacionais foram executados com o objetivo de avaliar a capacidade do método em encontrar valores de limitantes inferiores nas instâncias do ITC-2007.

Palavras-chave: Problema de Tabela-Horário de Universidade; *Dividir para conquistar*; Grafos.

ABSTRACT

This study deals with the University Timetable Problem (UTP), which is one of the well-known problem in the Combinatorial Optimization field. The problem involves scheduling classes in available rooms, taking into consideration time constraints and the availability of students and professors, as well as other predefined constraints. Various formulations for UTP exist in the literature, as each institution has different constraints. The complexity of this problem arises due to the large number of variables and constraints. In this work, the UTP based on the Curriculum-Based Course Timetabling (CB-CTT) formulation proposed in the second edition of the International Timetabling Competition (ITC-2007) was considered. Therefore, the objective was to use the divide and conquer approach to obtain lower bounds. The METIS library was employed to partition the original problem, represented by graphs, into subproblems, which were then solved using the CPLEX *solver*. Computational experiments were conducted to assess the method's ability to find lower bound values in instances from the ITC-2007.

Keywords: University Timetable Problem; divide and conquer; Graphs.

LISTA DE FIGURAS

Figura 1 – Grafo G_U de uma instância fictícia particionado em 2 subproblemas. 20
Figura 2 - Fluxograma do método proposto. 21

LISTA DE TABELAS

Tabela 1: Número de partições k para os métodos *U-partition* e *U-partition-org* 23
Tabela 2: Limitantes inferiores obtidos pelos métodos *U-partition* e *U-partition-org* com os tempos de execução do *solver* limitados.....24
Tabela 3: Comparação de limitantes inferiores obtidos pelo método *U-partition-org* com outros métodos da literatura.....26

SUMÁRIO

1. INTRODUÇÃO 7
1.1. PROBLEMA E SUA IMPORTÂNCIA 8
1.2. OBJETIVOS 10
1.2.1. Objetivo geral 10
1.2.2. Objetivos específicos 11
2. REVISÃO DA LITERATURA 12
2.1. Modelo de duas fases de Lach e Lübbecke (2012) 13
2.2. Abordagem dividir para conquistar de Hao e Benlic (2011) 15
2.3. Novos métodos de Kampke (2020) 16
2.4. Métodos Heurísticos 16
3. METODOLOGIA 18
3.1. *U-partition* 18
3.2. *U-partition-org* 19
3.3. METIS 20
3.4. Fluxograma do método *U-partition-org* 21
4. RESULTADOS OBTIDOS 22
6. CONCLUSÃO 27
7. REFERÊNCIAS 28

1. INTRODUÇÃO

A Otimização Combinatória (OC) se mostra de grande importância em diversas áreas de conhecimento, sendo aplicada na resolução de muitos problemas do cotidiano como exemplo: roteamento de veículos, planejamento da produção e na elaboração de tabela-horário.

Os problemas de OC consistem em achar a melhor combinação de valores de um conjunto de variáveis, que obedecem determinadas restrições, para que uma função que define o problema, seja maximizada ou minimizada.

Dentre os problemas de OC, o conjunto de problemas envolvendo tabela-horário é muito abordado na literatura. Dentre eles, o Problema de Tabela-Horário de Universidades (PTHU) tem especial atenção, pois sua construção manual é muito difícil e demanda muito tempo, sendo que muitas vezes não atende o interesse e as disponibilidades de todas as pessoas envolvidas, podendo assim ter impacto direto no funcionamento da instituição e nas vidas dessas pessoas. Além disso, as universidades terão que se adequar para atendê-lo, sendo necessário resolver este problema semestralmente.

Os primeiros estudos relacionados ao problema de tabela-horário tiveram início nos anos 60, quando Appleby, Black e Newman (1961) descreveram métodos para a construção de soluções para o problema, mediante o uso de computadores. Com Gotlieb (1962) este problema ganhou grande destaque, ao descrever a primeira formulação completa para o problema de tabela horário para escolas. Desde então, diversos estudos e pesquisas vêm acontecendo e apesar de ser bastante explorado, sua resolução continua sendo um desafio (MOURA et al., 2004).

Com o intuito de incentivar a comunidade acadêmica e gerar pesquisas de diferentes abordagens, além de disponibilizar instâncias reais do problema, *a European Metaheuristics Network* em 2002, 2007, 2011 e 2019 organizou campeonatos internacionais de tabela-horário (*International Timetabling Competition-ITC*). A segunda edição do campeonato trata do problema de tabela-horário em universidades, dividindo-o em três tipos de formulações: tabela-horário de exames, tabela-horário de universidades baseando-se em matrícula (*Post-Enrolment-based Course Timetabling, PE-CTT*) e tabela-horário de universidades baseando-se em currículo de cursos (*Curriculum-Based Course Timetabling, CB-CTT*). Como a formulação CB-CTT do ITC-2007 e suas instâncias são as que mais se aproximam do

que é praticado nas universidades brasileiras, essa formulação e suas instâncias foram usadas neste trabalho.

1.1. PROBLEMA E SUA IMPORTÂNCIA

O PTHU da formulação CB-CTT é definido pela alocação semanal de aulas de um dado número de disciplinas, levando em conta um número limitado de salas e períodos em que as aulas podem ser alocadas durante o dia, considerando que os conflitos entre disciplinas são definidos pelo currículo do curso.

Vale ressaltar que as instâncias usadas no ITC-2007 foram baseadas em instâncias reais da Universidade de Udine, na Itália, e se aplicam a outras universidades internacionais, embora simplificadas em relação ao problema real, para que se mantivesse generalizada. Segundo Gaspero, McCollum e Schaerf (2007), na formulação CB-CTT são consideradas as seguintes informações ou entidades:

- **Dias, períodos e horários**: É dado um número de dias da semana onde haverá aulas. É definido um número fixo de períodos de aula, que é igual para todos os dias. Um horário é definido por um par composto de um dia e um período.
- **Disciplinas e professores**: Cada disciplina possui um número de aulas semanais que devem ser alocadas em períodos distintos. A disciplina é lecionada por um professor e assistida por um número de alunos. Cada disciplina possui um número de dias mínimos para que suas aulas sejam alocadas.
- **Salas**: Cada sala possui uma capacidade definida pelo número de assentos.
- **Currículos**: Grupo de disciplinas que possuem alunos em comum.
- **Indisponibilidades**: Alguns períodos são indisponíveis para determinadas disciplinas.

A partir daí, a formulação CB-CTT do ITC-2007 considera restrições fortes e fracas. As restrições fortes sempre devem ser respeitadas para a obtenção de uma solução aceitável. Caso uma restrição forte seja violada, será gerada uma tabela-horário inviável. Já as restrições fracas devem ser respeitadas o máximo possível, sendo que, quanto menor o número de restrições fracas violadas, melhor será a tabela-horário gerada. As restrições da formulação CB-CTT do ITC-2007 são definidas a seguir:

Restrições Fortes:

- Aulas: Todas as aulas das disciplinas devem ser alocadas e em horários diferentes. Uma violação ocorre se uma aula não é alocada ou se duas aulas da mesma disciplina são alocadas no mesmo horário.
- Ocupação de sala: Duas ou mais aulas de quaisquer disciplinas não podem ocupar a mesma sala no mesmo horário. É contada uma violação se duas ou mais aulas são alocadas na mesma sala e no mesmo horário.
- Conflitos: Aulas de disciplinas pertencentes ao mesmo currículo ou lecionadas pelo mesmo professor devem ser alocadas em horários diferentes. Duas aulas conflitantes alocadas no mesmo horário representam uma violação.
- Indisponibilidade: Se o professor da disciplina não estiver disponível para lecionar em um determinado período, então nenhuma aula da disciplina pode ser agendada neste período. Cada aula agendada em um período indisponível é uma violação.

Restrições Fracas:

- Capacidade da sala: Para cada disciplina, o número de alunos matriculados deve ser igual ou menor ao número de assentos disponíveis na sala em que a disciplina for lecionada. Cada aluno acima da capacidade representa uma violação.
- Número mínimo de dias de aula: Aulas de cada disciplina devem ser espalhadas em um determinado número mínimo de dias. Cada dia abaixo do mínimo representa uma violação.
- Aulas isoladas: Aulas de um mesmo currículo devem ser alocadas em horários adjacentes. Cada aula isolada representa uma violação.
- Estabilidade de sala: Todas as aulas de uma disciplina devem ser alocadas em uma mesma sala. Cada sala diferente representa uma violação.

Cada tabela-horário recebe um valor que representa a qualidade da solução. Cada restrição que é violada aumenta este valor de acordo com o peso e o número de violações. Sendo assim, uma solução viável e de qualidade é aquela que respeita todas as restrições fortes e minimiza a quantidade de restrições fracas violadas.

De acordo com a organização do ITC-2007, são considerados pesos diferentes para cada tipo de restrição fraca. A restrição de dias mínimos possui peso 5, aulas isoladas possui peso 2, capacidade da sala e estabilidade de sala possuem peso 1.

Portanto, o objetivo do problema proposto na formulação CB-CTT é encontrar uma solução viável que minimize a soma ponderada de violações das restrições fracas. Considerando Ω o conjunto de todas as soluções viáveis, para cada solução $x \in \Omega$, o objetivo do problema é minimizar a função objetivo $f(x)$, definida a seguir:

$$f(x) = \sum_{i=1}^{4} \alpha_i \, \beta_i(x)$$

onde α_i representa os pesos atribuídos à cada restrição fraca i e $\beta_i(x)$ representa o número de vezes que cada restrição fraca i foi violada na solução viável $x \in \Omega$.

Mesmo após anos do acontecimento do ITC-2007, diversas pesquisas e trabalhos vêm sendo desenvolvidos para a solução do PTHU proposto na formulação CB-CTT, tendo em vista que as instâncias disponibilizadas são de conhecimento de toda comunidade acadêmica que estuda este problema. Além disso, algumas instâncias ainda não tiveram valores ótimos definidos.

Dessa forma, alguns estudos encontrados na literatura procuram apresentar modelos matemáticos e métodos para encontrar limitantes inferiores para essa formulação. Estes métodos são importantes pois demonstram uma avaliação das soluções obtidas por métodos de soluções já conhecidos e ainda provar a otimalidade das soluções. Ainda assim, são poucos os trabalhos encontrados na literatura que usam o método *dividir para conquistar* juntamente com a biblioteca *METIS* para encontrar limitantes inferiores para a formulação CB-CTT. Neste sentido, surge a pretensão do desenvolvimento deste trabalho.

1.2. OBJETIVOS

Essa seção é dividida em objetivo geral e objetivos específicos.

1.2.1. Objetivo geral

O objetivo deste trabalho foi estudar e compreender o problema de tabela-horário voltado para universidades, utilizando o modelo proposto para a formulação CB-CTT do ITC-2007. Além disso, foi proposto e aplicado o método *U-partition-org*, que usa a abordagem *dividir para conquistar* para encontrar valores de limitantes inferiores.

1.2.2. Objetivos específicos

- Realizar uma revisão da literatura sobre o problema de tabela-horário voltado para universidades;
- Compreender o modelo matemático que será usado neste trabalho para a obtenção de limitantes inferiores para o PTHU da formulação CB-CTT;
- Implementar o método *U-partition-org*, que utiliza a abordagem *dividir para conquistar*;
- Realizar testes computacionais do método proposto neste trabalho;
- Comparar os resultados obtidos neste trabalho com os resultados já encontrados na literatura.

2. REVISÃO DA LITERATURA

De acordo com Qu e Burke (2009), dentre os vários problemas de tabela-horário, o educacional é um dos mais estudados, pois é um problema que pode afetar um grande número de pessoas que estão envolvidas, além de ser recorrente em universidades e escolas.

Muitas das primeiras técnicas utilizadas para solucionar os problemas de tabela-horário foram baseadas na forma como as pessoas resolvem o problema manualmente, chamadas de heurísticas construtivas. Essa técnica consiste na inserção das aulas de forma gradual, até que todas as aulas sejam alocadas ou aconteça algum conflito. Em caso de conflito, é feito o remanejamento de algumas aulas já alocadas para que seja sanado o conflito.

Especialmente no caso do PTHU, as técnicas de solução para o problema têm recaído basicamente em três grupos: programação matemática, programação lógica e, principalmente, meta-heurísticas.

Os métodos de programação matemática são aqueles que procuram solucionar o problema procurando a solução ótima através de modelos matemáticos denominados exatos. Devido à complexidade do PTHU, a maioria dos métodos exatos levam um tempo considerável para ser executado, sendo assim, eles relaxam algumas restrições para encontrarem uma solução em menos tempo. Sendo o CB-CTT um problema de minimização, a solução encontrada representa um limitante inferior.

Em contrapartida, as meta-heurísticas procuram solucionar o problema da forma mais rápida, encontrando uma solução viável, nem sempre a melhor solução (solução ótima). Neste caso a solução do problema representa um limitante superior. Dessa forma, quanto menor for a distância dos resultados entre a solução heurística e a solução dos métodos de programação matemática (limitantes inferiores), melhor a qualidade da solução do método heurístico.

Neste trabalho foi dada maior importância aos trabalhos que abordam a formulação CB-CTT do ITC-2007. Seguido de alguns trabalhos que utilizam formulação matemática para a obtenção de limitantes inferiores.

Os primeiros trabalhos que apresentaram solução usando métodos matemáticos utilizavam modelos com muitas restrições e variáveis (BURKE et al; 2010). Com o passar dos anos foram apresentados métodos com menos variáveis e restrições

(LACH; LÜBBECKE, 2012). A partir disso, métodos que usam menos variáveis e usam a abordagem *dividir para conquistar* foram apresentados (HAO; BENLIC, 2011).

2.1. Modelo de duas fases de Lach e Lübbecke (2012)

Lach e Lübbecke (2012) apresentaram um modelo de Programação Linear Inteira (PLI) onde eram usadas menos variáveis e restrições do que o modelo proposto por Burke et al. (2010).

O modelo é dividido em dois modelos menores de PLI, onde são representados fases diferentes da resolução do problema. Na primeira fase, o modelo desconsidera a alocação das aulas em salas e leva em consideração apenas o período em que as aulas podem ser alocadas. Nesta fase, o número de aulas que é alocado em um período é limitado pela quantidade de salas disponíveis. Na primeira parte do modelo, o objetivo é minimizar as seguintes restrições fracas: capacidade de sala, número mínimo de dias de aula e aulas isoladas. Depois da resolução da primeira parte, é considerada a restrição de estabilidade da sala, e assim as aulas são alocadas nas salas.

Neste modelo são usados os seguintes conjuntos de variáveis:

- $\upsilon_{d,c}$ é igual a 1 se a disciplina c possuir pelo menos uma aula alocada no dia d e 0 caso contrário;
- q_c representa a diferença entre mld_c e o número de dias distintos que a disciplina c possui aulas alocadas, caso a diferença seja positiva e 0 caso contrário, sendo que mld_c é o número mínimo de dias sugeridos para a alocação das aulas da disciplina c;
- $z_{u,p}$ é igual a 1 se há uma aula isolada do currículo u alocada no período p e 0 caso contrário;
- $x_{p,c}$ é igual a 1 se a disciplina c possuir uma aula alocada no período p e 0 caso contrário;
- $\omega_{p,\psi,c}$ é igual a 1 se a disciplina c possuir uma aula alocada no período p em uma sala com capacidade menor que ψ e 0 caso contrário;
- $\varphi_{u,p}$ é igual a 1 se há uma aula isolada do currículo u alocada no período p e 0 caso contrário;

O modelo a seguir é o que representa a primeira fase do modelo proposto por Lach e Lübbecke (2012):

$$min \sum_{\psi \in \Psi} \sum_{p \in P} \sum_{\substack{c \in C, \\ c \in N_p}} \omega_{p,\psi,c} \,.\, \theta_{p,\psi,c} + 5 \sum_{c \in C} q_c + 2 \sum_{u \in U} \sum_{p \in P} z_{u,p}$$

Sujeito a

$$\sum_{p \in N_c} x_{p,c} = lct_c \qquad \forall c \in C \ (2.1)$$

$$\sum_{c \in N_p} x_{p,c} \leq |R| \qquad \forall p \in P \ (2.2)$$

$$x_{p,c} \geq \omega_{p,\psi,c} \qquad \forall \psi \in \Psi, \quad \mathrm{c} \in C_{\geq \psi}, \quad (\mathrm{c}, \mathrm{p}) \in N \ (2.3)$$

$$\sum_{\substack{c \in C, \\ c \in N_p}} (x_{p,c} - \omega_{p,\psi,c}) \leq \left| R_{\geq \psi} \right| \qquad \forall p \in P, \quad \psi \in \Psi \ (2.4)$$

$$\sum_{\substack{p \in P_d, \\ p \in N_c}} x_{p,c} \geq \upsilon_{d,c} \qquad \forall d \in D, \quad c \in C \ (2.5)$$

$$\sum_{d \in D} \upsilon_{d,c} \geq mld_c - q_c \qquad \forall c \in C \ (2.6)$$

$$\sum_{\substack{c \in C_u, \\ c \in N_p}} x_{p,c} = \varphi_{u,p} \qquad \forall u \in U, \quad p \in P \ (2.7)$$

$$-\varphi_{u,p-1} + \varphi_{u,p} - \varphi_{u,p+1} \leq Z_{u,p} \qquad \forall u \in U, \quad p \in P \ (2.8)$$

$$\sum_{\substack{c \in C_t, \\ c \in N_p}} x_{p,c} \leq 1 \qquad \forall p \in P, \quad t \in T \ (2.9)$$

$$x_{p,c} \in \mathbb{B} \qquad \forall (c,p) \in P (2.10)$$

$$\omega_{p,\psi,c} \in \mathbb{B} \qquad \forall \psi \in \Psi, \ \mathrm{c} \in C_{\geq \psi}, (c,p) \in N \ (2.11)$$

$$\upsilon_{d,c} \in \mathbb{B} \qquad \forall d \in D, c \in C \ (2.12)$$

$$q_c \in \mathbb{Z} \qquad \forall c \in C \ (2.13)$$

$$z_{u,p} \in \mathbb{B} \qquad \forall u \in U, p \in P \ (2.14)$$

$$\varphi_{u,p} \in \mathbb{B} \qquad \forall u \in U, p \in P \ (2.15)$$

Os três termos da função objetivo representam três restrições fracas: capacidade da sala, número mínimo de dias de aula e aulas isoladas, respectivamente. Também é importante ressaltar que o valor de $\theta_{p,\psi_i,c}$ é definido por:

$$\theta_{p,\psi_i,c} = min\{st_c - \psi_i, \psi_{i+1} - \psi_i\}$$

O valor de $\theta_{p,\psi_i,c}$ é assim definido para que a aula da disciplina c lecionada no período p, caso não possa ser alocada em uma sala com capacidade maior que o número de alunos matriculados, seja alocada preferencialmente na i-ésima sala cuja capacidade ψ_i é imediatamente inferior ao número de alunos matriculados.

O conjunto de restrições (2.1) - (2.4) modelam as restrições fortes. Já o conjunto de restrições (2.5) - (2.9) modelam as restrições fracas, e por fim, o conjunto de restrições (2.10) - (2.15) definem o domínio das variáveis de decisão.

A segunda fase do modelo é representada como a formulação padrão do problema de *matching* completo de custo mínimo em um grafo bipartido (AHUJA; MAGNANTI; ORLIN, 1993). Neste trabalho será usado somente o modelo da primeira parte do trabalho de Lach e Lübbecke (2012).

2.2. Abordagem dividir para conquistar de Hao e Benlic (2011)

O trabalho de Hao e Benlic (2011) teve como motivação o trabalho publicado por Lach e Lübbecke (2012). O método proposto segue o princípio *dividir para conquistar*, uma vez que o modelo é baseado na ideia de particionar o problema e resolver as partes separadamente.

No método proposto por Hao e Benlic (2011) um conjunto de restrições, do modelo que representa a primeira fase de Lach e Lübbecke (2012) é eliminado ou relaxado, possibilitando o problema ser dividido em subproblemas.

O método proposto pode ser definido nos seguintes passos:

1. Particionar o conjunto C de disciplinas.
2. Resolver cada subproblema, obtido em cada partição, com um *solver*.
3. Somar os valores encontrados na etapa anterior para calcular o limitante inferior do problema original.

No passo 1, Hao e Benlic (2011) executaram um algoritmo *Iterated Tabu Search*. Esse algoritmo define a partição do problema original em k partições através do grafo $G = (V, A)$, definido da seguinte maneira: o conjunto de vértices V possui um vértice para cada disciplina $c \in C$, e uma aresta $\{c1, c2\}$ pertence a A se existir pelo menos um

currículo u que tenha ambas as disciplinas. A partição é obtida removendo iterativamente arestas de G, até que o grafo tenha exatamente k componentes conexas. Ao final, caso alguma aresta que representa o currículo u tenha sido removida, todas as demais arestas que representam o mesmo currículo u e que ainda estiverem em G, também são removidas.

O trabalho de Hao e Benlic (2011) foi o primeiro a encontrar limitantes inferiores para todas as 21 instâncias da formulação CB-CTT do ITC-2007.

2.3. Novos métodos de Kampke (2020)

Kampke (2020) apresenta novas estratégias para encontrar limitantes inferiores para a formulação CB-CTT do ITC-2007. Como no trabalho de Hao e Benlic (2011), também foi adotado o princípio *dividir para conquistar*, sendo que o problema é particionado, algumas restrições são relaxadas, cada partição representará um subproblema e isso viabiliza alcançar limitantes inferiores válidos.

No trabalho de Kampke (2020) são apresentadas duas formas de representar o PTHU da formulação CB-CTT como um grafo $G = (V, A)$. A primeira, denominada G_U, cada currículo u é representado por um vértice de G e na outra, denominada G_C, cada disciplina c representa um vértice de G. Em G_U, a aresta $\{u_1, u_2\} \in A$, se, e somente se, os currículos u_1 e u_2 tiverem pelo menos uma disciplina em comum. Já no grafo G_C, a aresta $\{c_1, c_2\} \in A$, se, e somente se, as disciplinas c_1 e c_2 pertencerem simultaneamente a um mesmo currículo.

Também são apresentados dois métodos para encontrar os limitantes inferiores, *U-partition* e *C-partition*, que fazem uso de métodos da biblioteca *METIS,* para particionar respectivamente os grafos G_U e G_C. Após realizar o particionamento do problema, os métodos relaxam algumas restrições, sendo feita em seguida a resolução dos subproblemas gerados através do *solver* comercial CPLEX.

2.4. Métodos Heurísticos

No contexto do ITC-2007, Müller (2009) e Lü e Hao (2010) apresentaram métodos heurísticos para abordar o problema CB-CTT. Müller desenvolveu um algoritmo de três fases, enquanto Lü e Hao propuseram o Adaptive Tabu Search. Müller (2009) foi o vencedor da competição na categoria CB-CTT, demonstrando resultados superiores em média em 10 execuções do seu método heurístico. Lü e Hao (2010) alcançaram o

segundo lugar, com seu método baseado em duas fases: inicialização e integração. Apesar de, em média, não superar os resultados de Müller, o Adaptive Tabu Search destacou-se, obtendo valores médios superiores em cinco instâncias e registrando um empate em duas instâncias, evidenciando sua competitividade como método heurístico.

Até o desenvolvimento deste trabalho, os resultados apresentados na literatura destacam-se pelos métodos *U-partition* e *C-partition* de Kampke (2020), propostos para encontrar limitantes inferiores para o PTHU na formulação CB-CTT do ITC-2007. Inspirados na estratégia dividir para conquistar proposta por Hao e Benlic (2011), esses métodos têm como motivação a importância de limitantes inferiores robustos para avaliar a qualidade de soluções obtidas por outros métodos, e em alguns casos, para provar a otimalidade dessas soluções. A abordagem adotada consiste na divisão do problema original em k subproblemas, utilizando a biblioteca METIS para o particionamento de grafos. Os resultados obtidos até o momento indicam que o método *C-partition* se destaca, apresentando limitantes inferiores que superam em média resultados da literatura, provando a otimalidade em 12 instâncias. Embora o método *C-partition* tenha obtido limitantes inferiores menores em média em comparação com trabalhos anteriores, como os de Bagger, Desaulniers e Desrosiers (2019) e Bagger, Sørensen e Stidsen (2019), é relevante notar que, com exceção de algumas instâncias específicas, os resultados se mantiveram aproximadamente equivalentes.

3. METODOLOGIA

Kampke (2020) apresenta duas novas estratégias para a obtenção de limitantes inferiores para a formulação CB-CTT do ITC-2007. Denominadas *C-partition* e *U-partition*, essas estratégias usam o princípio *dividir para conquistar* também usada em Hao e Benlic (2011), onde é feita a divisão do problema em subproblemas para a resolução de cada um. Também são apresentados os métodos *C-partition-null* e *C-partition-org*, que se diferenciam do *C-partition* pelo motivo de usarem variáveis auxiliares no momento de relaxar as restrições, para obter uma relaxação mais forte.
Ao realizar experimentos computacionais preliminares com os métodos *C-partition* e *U-partition*, sendo executados em todas as instâncias disponíveis no ITC-2007 para a formulação CB-CTT, disponíveis em http://www.cs.qub.ac.uk/itc2007, foi constatado que a média dos valores de limitantes inferiores encontrados pelo método *C-partition* foi melhor que o método *U-partition*, além de encontrar valores melhores que os conhecidos até então em algumas instâncias.
Considerando que o método *U-partition* foi ligeiramente inferior ao método *C-partition* nos experimentos preliminares executados por Kampke (2020), e que o método *U-partition* também pode ser adaptado para utilizar variáveis auxiliares e obter uma relaxação mais forte. Considerando por fim que esse método ainda não foi executado, é apresentado o método *U-partition-org*.

3.1. *U-partition*

No método *U-partition* apresentado em Kampke (2020), o grafo que representa o problema original G_U, é particionado em k partes, sendo que o particionamento minimiza a soma dos pesos das arestas que serão cortadas. Assim, a ideia principal do método é dividir o problema original em subproblemas, que consequentemente terão menos currículos que o problema original.
Ao particionar G_U em k partes, cada currículo (vértice) pertence a um e apenas um subproblema SP_i, onde i pode variar de 1 a k. No caso das disciplinas que pertencem a mais de um currículo e estes currículos pertencem aos subproblemas SP_i e SP_j, $i \neq j$, elas são representadas pelo conjunto de arestas cortadas A^-. Quando uma disciplina c não é representada por uma aresta pertencente a A^-, significa que todos os currículos u que possuem a disciplina c pertencem ao mesmo subproblema SP_i. Quando uma disciplina c é representada por uma ou mais arestas em A^-, o método

usa uma estratégia gulosa, onde é verificado em qual currículo possui mais disciplinas, ou seja, a disciplina c é considerada na função objetivo do subproblema que possui o currículo que possui mais disciplinas. Se houver empate entre os currículos a escolha é feita aleatoriamente.

Depois de particionar o grafo G_U e definir em qual função objetivo cada disciplina c é considerada, o método gera cada subproblema utilizando o modelo matemático da primeira fase de Lach e Lübbecke (2012).

3.2. *U-partition-org*

Nesta seção é proposto o método *U-partition-org*, que é baseado no método *U-partition*, apresentado por Kampke et al. (2020) e visto anteriormente. O método *U-partition-org* também particiona o grafo G_U, que representa o problema original, em k partes, assim minimizando o peso das somas das arestas cortadas. O método continua gerando subproblemas SP_i, onde $i = 1, \dots, k$.

De forma semelhante ao *U-partition*, no caso em que disciplinas pertencem a mais de um currículo e estes currículos pertencem aos subproblemas SP_i e SP_j, $i \neq j$, elas são representadas pelo conjunto de arestas cortadas A^-. Se uma disciplina c não é representada por uma aresta cortada pertencente a A^-, também significa que os currículos u que possuem a disciplina c pertencem ao mesmo subproblema SP_i.

No entanto, a principal diferença entre os métodos *U-partition* e *U-partition-org* está na relação das disciplinas c que são representadas como arestas cortadas que pertencem ao conjunto A^-. No método *U-partition*, se uma disciplina c é representada por uma ou mais arestas pertencentes a A^-, significando que a disciplina c pertence a dois ou mais currículos, sendo que estes currículos são localizados em subproblemas diferentes, é utilizada uma estratégia gulosa, para que, dentre os dois currículos, a disciplina seja considerada somente na função objetivo do subproblema que possui o currículo com mais disciplinas. Já no método *U-partition-org*, se uma ou mais disciplinas c são representadas por arestas pertencentes a A^-, a disciplina é considerada nas funções objetivo de todos subproblemas que possuem os currículos que contém a disciplina c. Nos subproblemas que possuem menor número de disciplinas, a disciplina c pertencente a A^- será representada por variáveis auxiliares.

Portanto, no método *U-partition*, a disciplina c representada por uma aresta em A^- será considerada somente em um subproblema, enquanto no método *U-partition-org*,

serão utilizadas variáveis auxiliares para que a disciplina c seja considerada em todos os subproblemas que possuem currículos onde a disciplina c está contida.

Como é apresentado na Figura 1, supondo que em uma instância existam as disciplinas $MatDisc$, $Calc1$, $Prog1$ e Vet e que elas compõem os currículos $Cur1$ e $Cur2$, respectivamente, da seguinte forma: $Cur1 = \{MatDisc,\ Calc1,\ Prog1\}$ e $Cur2 = \{Vet,\ Prog1\}$. No método *U-partition*, $Prog1$ seria considerada somente no subproblema SP_1, enquanto no *U-partition-org*, a disciplina $Prog1$ será representada em SP_1 e SP_2, mas em SP_2 será considerada com variáveis auxiliares.

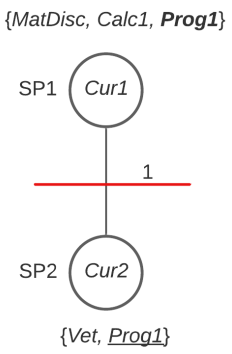

Figura 1 – Grafo G_U de uma instância fictícia particionado em 2 subproblemas.

Logo após a resolução dos k subproblemas através do *solver* CPLEX, se houver alguma resolução onde as variáveis auxiliares tenham contribuído para o aumento do valor da resolução da função objetivo em algum subproblema, então esse valor aumentado será decrementado após a soma de todos os valores das soluções dos k subproblemas, que representam a solução final.

3.3. METIS

Para realizar o particionamento do grafo G_U, que representa o problema original, foram utilizados métodos da biblioteca *METIS*. Essa biblioteca ao realizar o particionamento do grafo G_U, leva em consideração que as partições devem ter no máximo um vértice de diferença, para que todas as partições possuam quantidades semelhantes de vértices, evitando subproblemas que sejam resolvidos facilmente e outros sejam de difícil resolução.

A biblioteca foi desenvolvida no Departamento de Ciência da Computação e Engenharia da Universidade de Minnesota, sendo distribuída gratuitamente e disponível para *download* em *http://glaros.dtc.umn.edu/gkhome/views/metis*.

3.4. Fluxograma do método *U-partition-org*

Na Figura 2 é exibido o fluxograma para o método *U-partition-org*. A primeira etapa consiste na entrada dos dados. Logo após o problema é representado com o gráfico G_U. Em seguida o grafo é particionado em k subproblemas utilizando a biblioteca *METIS*. Logo após são definidas quais disciplinas serão representadas por variáveis originais ou auxiliares em cada subproblema. Já no final, os subproblemas são criados e resolvidos pelo *solver* CPLEX. O valor do limitante inferior é calculado, decrementando o valor das variáveis auxiliares do valor retornado pelo *solver* CPLEX em cada subproblema, e em seguida, soma-se esses valores terminando a execução do método.

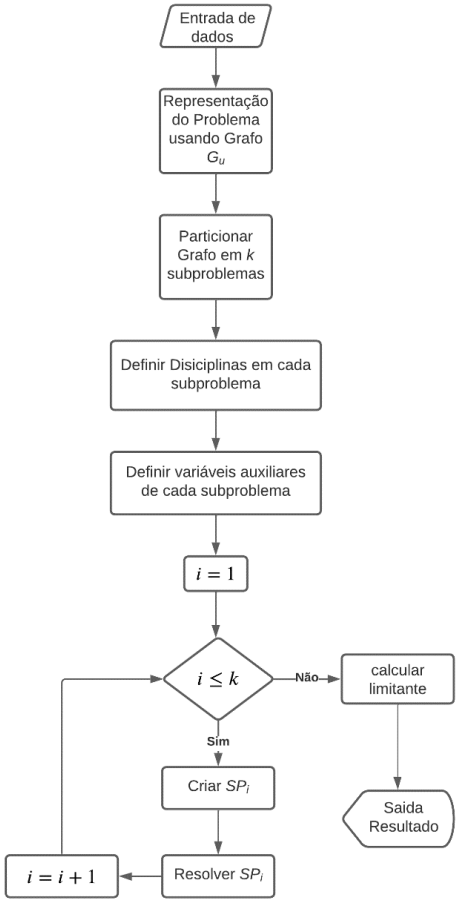

Figura 2 - Fluxograma do método proposto.

4. RESULTADOS OBTIDOS

Para avaliar o desempenho do método *U-partition-org*, o mesmo foi submetido a testes computacionais executados nas 21 instâncias do ITC-2007 para a formulação CB-CTT (comp01 - comp21).

O método *U-partition-org* foi implementado com a linguagem de programação C++ e compilado pelo compilador g++, versão 11.4.0 e flag de otimização -O3.

O grafo GU é particionado, ou seja, dividido em subproblemas, com a API da biblioteca METIS (versão 5.1.0 64 bits). Os modelos matemáticos dos subproblemas foram resolvidos com o *solver* CPLEX 12 64 bits para Linux.

Os experimentos computacionais foram realizados em um computador com processador AMD Ryzen 7 5700G 3.80GHZ, 16GB de memória RAM e sistema operacional Ubuntu de 64 bits.

Os testes computacionais executados para o método *U-partition-org* seguiram os seguintes passos: primeiramente foram executados os testes para todas as instâncias com o número de partições k variando de 2 até 10, anotando os resultados dos limitantes inferiores e do tempo que o *CPLEX* levava para resolver cada instância, para cada valor de k. Para realizar uma comparação equivalente aos trabalhos propostos na literatura, foi definido um tempo limite de execução do *solver* para cada subproblema, seguindo a regra do ITC-2007, onde é definido um tempo de 40τ. O valor de τ é definido pelo programa *benchmark*, disponibilizado pela organização do ITC-2007, onde o mesmo analisa as especificações do computador e define um tempo que seria equivalente ao tempo dos computadores usados no campeonato. Executando o programa *benchmark* no computador onde os testes foram realizados, obteve-se o valor de 108 segundos, assim restringindo o tempo de execução do *solver* em 4320 segundos para cada subproblema. Para definir os valores de k, são selecionados aqueles onde o valor do limitante inferior obtido (*lb*) foi melhor. Observando a Tabela 1, em quase todas as instâncias, exceto na comp12, o valor do melhor limitante inferior foi obtido sem que nenhum subproblema extrapolasse o tempo limite de 4320 segundos. Apenas para a instância comp12, o melhor valor de limitante inferior (*lb*) foi obtido com $k = 3$ e tendo algum dos subproblemas extrapolando o tempo limite de tempo definido. Com $k = 4$ (entre parênteses na Tabela 1), obteve-se um valor de limitante inferior pior, porém sem nenhum subproblema extrapolar o limite de tempo definido.

Dessa forma, é definido os valores do número de partições k para cada instância. Da mesma forma, foram executados os testes para o *U-partition*, com o código disponibilizado por Kampke (2020), para fins de comparação.

Na Tabela 1 são apresentados os valores dos números de partições para cada instância que obtiveram os melhores valores de limitantes inferiores.

Tabela 1: Número de partições k para os métodos *U-partition* e *U-partition-org*

Instâncias	k	
	U-partition	*U-partition-org*
comp01	2	2
comp02	2	2
comp03	2	2
comp04	2	2
comp05	4	4
comp06	2	2
comp07	2	2
comp08	2	2
comp09	2	2
comp10	2	2
comp11	2	2
comp12	3(4)	3(4)
comp13	2	2
comp14	2	2
comp15	2	2
comp16	2	2
comp17	2	2
comp18	2	2
comp19	2	2
comp20	2	2
comp21	2	2

A Tabela 2 apresenta os valores de limitantes inferiores dos métodos *U-partition* e *U-partition-org* obtidos nos testes realizados. Vale ressaltar que o número de partições para cada instância e para cada método foram apresentados na Tabela 1. Também é apresentado na Tabela 2 a soma do tempo em que o *solver* CPLEX levou para resolver cada subproblema para cada instância.

A partir dos dados mostrados na Tabela 2, percebe-se que para as 21 instâncias executadas, a maioria dos limitantes inferiores foram iguais, com o método *U-partition-org* apresentando melhor resultado apenas na instância comp12. Em geral, o método levou vantagem de tempo para a resolução em 17 das 21 instâncias. Em média, o

método *U-partition-org* ficou bem próximo do *U-partition* em relação aos limitantes inferiores, e levou vantagem no tempo de resolução das 21 instâncias.

Tabela 2: Limitantes inferiores obtidos pelos métodos *U-partition* e *U-partition-org* com os tempos de execução do *solver* limitados

Instâncias	*U-partition*		*U-partition-org*	
	lb	T(*seg.*)	*lb*	T(*seg.*)
comp01	4	0	4	0
comp02	16	607	16	602
comp03	60	695	60	704
comp04	35	2	35	2
comp05	183	2483	183	2518
comp06	22	336	22	314
comp07	6	5	6	5
comp08	37	3	37	2
comp09	96	1433	96	1100
comp10	4	3	4	2
comp11	0	0	0	0
comp12	143	11938	**144**	11635
comp13	59	15	59	14
comp14	51	17	51	17
comp15	60	716	60	698
comp16	18	22	18	21
comp17	56	473	56	472
comp18	25	132	25	130
comp19	42	4	42	4
comp20	4	86	4	84
comp21	74	746	74	725
Media	47,38	938,86	47,43	907,10

A Tabela 3 apresenta os resultados de estudos encontrados na literatura para as 21 instâncias usadas. Os valores de *ub* são os melhores resultados dos limitantes superiores encontrados da literatura para as 21 instâncias testadas. Também é apresentando para cada método, o valor do limitante inferior encontrado para a respectiva instância. Ao lado, o valor da diferença percentual entre o valor de *ub* e o respectivo valor do limitante inferior *lb* é apresentado.

Em **negrito** é destacado o melhor valor de *lb* encontrado para cada instância. Também se encontra abaixo da tabela uma legenda com os trabalhos usados na comparação dos resultados.

Como podemos ver na Tabela 3, o método *U-partition-org* apresentou, em média, resultados melhores que 4 dos métodos comparados. Podemos também destacar que o método *U-partition-org* identificou dois novos limitantes inferiores mais eficazes para as instâncias comp03 e comp15, previamente desconhecidos pelos trabalhos comparados. Além disso, provou a otimalidade de 14 das 21 instâncias testadas, assim ultrapassando o número encontrado pelo modelo proposto por Bagger, Desaulniers e Desrosiers (2019) que prova a otimalidade de 13 das 21 instâncias.

O *gap* médio em relação aos melhores valores de limitantes superiores para cada instância encontrado pelo método *U-partition-org* foi melhor do que 4 dos 6 métodos comparados, chegando bem perto dos valores de Bagger, Desaulniers e Desrosiers (2019) e de Bagger, Sorensen e Stidsen (2019).

Analisando os resultados obtidos nos testes, é possível concluir que o método *U-partition-org* obteve bons resultados em relação aos trabalhos comparados da literatura, provando a otimalidade de 14 das 21 instâncias, com um *gap* médio de 12,06%.

Tabela 3: Comparação de limitantes inferiores obtidos pelo método *U-partition-org* com outros métodos da literatura

Instancias	*ub*	HB11[1]		CCRT13[2]		AN14[3]		BDD16[4]		BKSS17[5]		BSS18[6]		*U-partition-org*	
		lb	*gap*(%)	*lb*	*gap*(%)	*lb*	*gap*(%)	*lb*	*gap*(%)	*lb*	*gap*(%)	*lb*	*gap*(%)	*lb*	*gap*(%)
comp01	5	4	20,00	**5**	0,00	0	100,00	0	100,00	**5**	0,00	0	100,00	4	20,00
comp02	24	12	50,00	16	33,33	16	33,33	**24**	0,00	8	66,67	20	16,67	16	33,33
comp03	64	38	40,63	52	18,75	28	56,25	54	15,63	38	40,63	58	9,38	**60**	6,25
comp04	35	**35**	0,00	**35**	0,00	**35**	0,00	**35**	0,00	**35**	0,00	**35**	0,00	**35**	0,00
comp05	284	183	35,56	166	41,55	48	83,10	210	26,06	186	34,51	**247**	13,03	183	35,56
comp06	27	22	18,52	11	59,26	**27**	0,00	26	3,70	16	40,74	23	14,81	22	18,52
comp07	6	**6**	0,00	**6**	0,00	**6**	0,00	**6**	0,00	**6**	0,00	**6**	0,00	**6**	0,00
comp08	37	**37**	0,00	**37**	0,00	**37**	0,00	**37**	0,00	**37**	0,00	**37**	0,00	**37**	0,00
comp09	96	72	25,00	92	4,17	35	63,54	**96**	0,00	74	22,92	92	4,17	**96**	0,00
comp10	4	**4**	0,00	2	50,00	**4**	0,00	**4**	0,00	**4**	0,00	**4**	0,00	**4**	0,00
comp11	0	**0**	0,00	**0**	0,00	**0**	0,00	**0**	0,00	**0**	0,00	**0**	0,00	**0**	0,00
comp12	294	109	62,93	100	65,99	99	66,33	175	40,48	142	51,70	**248**	15,65	144	51,02
comp13	59	**59**	0,00	57	3,39	**59**	0,00	**59**	0,00	**59**	0,00	**59**	0,00	**59**	0,00
comp14	51	**51**	0,00	48	5,88	**51**	0,00	**51**	0,00	44	13,73	49	3,92	**51**	0,00
comp15	62	38	38,71	52	16,13	28	54,84	54	12,90	38	38,71	58	6,45	**60**	3,23
comp16	18	16	11,11	13	27,78	**18**	0,00	**18**	0,00	13	27,78	17	5,56	**18**	0,00
comp17	56	48	14,29	48	14,29	**56**	0,00	53	5,36	44	21,43	**56**	0,00	**56**	0,00
comp18	61	24	60,66	**52**	14,75	27	55,74	**52**	14,75	36	40,98	**52**	14,75	25	59,02
comp19	57	56	1,75	48	15,79	46	19,30	**57**	0,00	56	1,75	51	10,53	42	26,32
comp20	4	2	50,00	**4**	0,00	**4**	0,00	**4**	0,00	0	100,00	3	25,00	**4**	0,00
comp21	74	61	17,57	68	8,11	42	43,24	**74**	0,00	57	22,97	71	4,05	**74**	0,00
Média		41,76	21,27	43,43	18,06	31,71	27,41	51,86	10,42	42,76	24,98	**56,48**	11,62	47,43	12,06

[1]HB11 - Hao e Benlic (2011) - Abordagem *dividir para conquistar* - Tempo Limite de 25200 segundos por subproblema.

[2]CCRT13 - Cacchiani et al. (2013) - *Geração de Colunas* - Tempo Limite de 22800 segundos.

[3]AN14 - Achá e Nieuwenhuis (2014) - Codificações SAT - Tempo Limite de 105 segundos.

[4]BDD16 - Bagger, Desaulniers e Desrosiers (2019) - Relaxações no modelo matemático de padrões - Tempo Limite de 20800 segundos.

[5]BKSS17 - Bagger et al. (2019) Conexão de dois modelos independentes através de técnicas de formulação de fluxo - Tempo Limite de 8320 segundos.

[6]BSS18 - Bagger, Sorensen e Stidsen (2019) - Geração de Colunas - Tempo Ilimitado.

6. CONCLUSÃO

Neste estudo foi apresentado o método *U-partition-org*, para a resolução do PTHU da formulação CB-CTT proposta no ITC-2007, encontrando valores de limitantes inferiores. O objetivo deste trabalho foi utilizar a abordagem *dividir para conquistar*, já abordadas por outros trabalhos da literatura, e, além disso, analisar a viabilidade do método proposto ao encontrar valores de limitantes inferiores, podendo assim, avaliar a qualidade das soluções obtidas por outros métodos e provar a otimalidade das soluções, uma vez que ainda existem instâncias em que as soluções encontradas ainda não foram provadas a sua otimalidade.

A aplicação do método para a resolução do PTHU se mostrou promissora. Os resultados obtidos demonstraram que o método foi, em média, melhor na resolução do problema, do que 4 dos 6 métodos comparados neste trabalho. O método também conseguiu encontrar a solução ótima para 14 das 21 instâncias testadas, destacando que método identificou dois novos limitantes inferiores mais eficazes para as instâncias comp03 e comp15 em relação aos trabalhos comparados. Além disso, o *gap* médio em relação aos melhores valores de limitantes superiores para cada instância, encontrado pelo método *U-partition-org,* foi melhor do que 4 dos 6 métodos comparados, chegando bem perto do de Bagger, Desaulniers e Desrosiers (2019) e de Bagger, Sorensen e Stidsen (2019).

Em comparação ao método *U-partition*, de onde o método proposto foi inspirado, os resultados se mostraram equivalentes, levando vantagem apenas na instância comp12, onde encontrou valor do limitante inferior maior e o *gap* médio em relação aos melhores valores de limitantes superiores foi ligeiramente melhor (0,02%)

Como resultado, o método *U-partition-org* demonstrou ser uma boa abordagem para a resolução do PTHU da formulação CB-CTT do ITC-2007, apresentando resultados consistentes em relação a estudos encontrados na literatura. Contudo, é importante levar em consideração que o tempo de execução para solucionar os subproblemas é de vital importância para encontrar resultados consistentes.

7. REFERÊNCIAS

ACHÁ, R. A.; NIEUWENHUIS, R. Curriculum-based course timetabling with sat and maxsat. Annals of Operations Research, v. 218, n. 1, p. 71–91, 2014.

AHUJA, R. K.; MAGNANTI, T. L.; ORLIN, J. B. Network flows: Theory, algorithms, and applications. Upper Saddle River, NJ, USA: Prentice-Hall Incorporation, 1993.

APPLEBY, J. S.; BLACK, D. V.; NEWMAN, E. A. Techniques for producing school timetabling on a computer and their application to other scheduling problems. The Computer Journal, 3, 237-245, 1961.

BAGGER, N.-C. F.; DESAULNIERS, G.; DESROSIERS, J. Daily course pattern formulation and valid inequalities for the curriculum-based course timetabling problem. Journal of Scheduling, v. 22, n. 2, p. 155–172, 2019.

BAGGER, N.-C. F.; SøRENSEN, M.; STIDSEN, T. R. Dantzig–wolfe decomposition of the daily course pattern formulation for curriculum-based course timetabling. European Journal of Operational Research, v. 272, n. 2, p. 430–446, 2019.

BURKE, E. K.; MERECĚK, J.; PARKES, A. J.; RUDOVÁ, H. Decomposition, reformulation, and diving in university course timetabling. Computers Operations Research, v. 37, n. 3, p. 582 – 597, 2010.

CACCHIANI, V. et al. A new lower bound for curriculum-based course timetabling. Computers Operations Research, v. 40, n. 10, p. 2466–2477, 2013.

GASPERO, L. D.; MCCOLLUM, B.; SCHAERF, A. The second international timetabling competition (itc-2007): Curriculum-based course timetabling (track 3). Proceedings of The International Conference on Automated Planning and Scheduling. Providence, Rhode Island, USA: ICAPS Incorporation. p. 1–5, 2007.

GOTLIEB, C. The Construction of class-teacher timetabling. Proceeding of the IFIP Congress. p. 73-77, 1962.

HAO, J.-K.; BENLIC, U. Lower bounds for the itc-2007 curriculum-based course timetabling problem. European Journal of Operational Research, v. 212, n. 3, p. 464–472, 2011.

KAMPKE, E. H. Novas Estratégias para Obtenção de Limitantes Inferiores para o Problema de Tabela-Horário de Universidades. Tese (Doutorado em Ciência da Computação). Universidade Federal do Espírito Santo. Vitória. 2020.

LACH, G.; LÜBBECKE, M. E. Curriculum based course timetabling: new solutions to udine benchmark instances. Annals of Operations Research, v. 194, n. 1, p. 255–272, 2012.

Lü, Z.; HAO, J.-K. Adaptive tabu search for course timetabling. European Journal of Operational Research, v. 200, n. 1, p. 235–244, 2010.

MOURA, A.; SCARAFICCI, R.; SILVEIRA, R.; SANTOS, V. Técnicas meta-heurísticas aplicadas à construção de grades horárias escolares. XXXVI SBPO - Simpósio Brasileiro de Pesquisa Operacional, v.1, p.1-20, São João Del Rey, MG, Brasil, 2004.

MÜLLER, T. Itc2007 solver description: a hybrid approach. Annals of Operations Research, v. 172, n. 1, p. 429–446, 2009.

QU, R; BURKE, E. K.; Hybridizations within a graph-based hyper-heuristic framework for university timetabling problems, Journal of the Operational Research Society 60(9) 1273–1285, 2009.