ReGeo ISSN: 2177-3246

SISTEMA DE GERAÇÃO AUTOMATIZADA DE HORÁRIOS DE AULA BASEADO EM HEURÍSTICA CONSTRUTIVA MULTI-START

AUTOMATED CLASS SCHEDULING SYSTEM BASED ON A MULTI-START CONSTRUCTIVE HEURISTIC

SISTEMA DE GENERACIÓN AUTOMATIZADA DE HORARIOS DE CLASE BASADO EN HEURÍSTICA CONSTRUCTIVA MULTI-START

https://doi.org/10.56238/revgeov17n10-004

Data de submissão: 17/08/2026 **Data de publicação:** 01/10/2026

Kassandra de Oliveira Rodrigues
Graduanda em Análise e Desenvolvimento de Sistemas
Instituição: Instituto Federal de Rondônia (IFRO)
E-mail: kassandra.olv.rodrigues@gmail.com

Robert Barreto Vieira
Graduando em Análise e Desenvolvimento de Sistemas
Instituição: Instituto Federal de Rondônia (IFRO)
E-mail: robertterto@gmail.com

Leandro Ferrarezi Valiante
Mestre em Ciência da Computação
Instituição: Instituto Federal de Rondônia (IFRO)
E-mail: leandro.valiante@ifro.edu.br

RESUMO

A elaboração manual de horários acadêmicos é atividade de elevada complexidade combinatória, sujeita a conflitos de alocação e uso ineficiente da infraestrutura institucional. Este trabalho apresenta o módulo de geração automatizada de horários do PlanIFica, sistema web em operação no Instituto Federal de Educação, Ciência e Tecnologia de Rondônia (IFRO), Campus Porto Velho Calama. A geração emprega heurística construtiva gulosa multi-start com função de avaliação ponderada: as alocações são ordenadas pelo princípio fail-first, múltiplas grades candidatas são construídas descartando-se as configurações que violem restrições duras e, entre as viáveis, seleciona-se a de maior pontuação. O sistema foi avaliado sobre instância institucional real, composta por 14 cursos, 63 turmas, 141 docentes, 37 ambientes e 563 aulas, sob três regimes de rigidez das restrições brandas. Na configuração de referência, alcançou 99,11% de alocação sem qualquer violação de restrição obrigatória, com convergência já nas primeiras tentativas de construção, sugerindo que a ordenação fail-first responde pela maior parte do ganho, com contribuição marginal da diversificação. Diferentemente de propostas validadas apenas sobre instâncias processadas offline, o PlanIFica opera em produção desde setembro de 2024, com o módulo de geração incorporado em 2026, e este foi avaliado sobre os dados reais de planejamento do campus, o que confere evidência empírica de aplicabilidade prática.

Palavras-chave: Geração Automatizada de Horários. Heurística Construtiva Multi-start. Função de Avaliação Ponderada. Escalonamento Educacional. Timetabling.

ABSTRACT
The manual preparation of academic timetables is an activity of high combinatorial complexity, subject to allocation conflicts and inefficient use of institutional infrastructure. This paper presents the automated timetable generation module of PlanIFica, a web system in operation at the Federal Institute of Education, Science and Technology of Rondônia (IFRO), Porto Velho Calama Campus. The generation employs a greedy multi-start constructive heuristic with weighted evaluation function: allocations are ordered by the fail-first principle, multiple candidate grids are constructed by discarding configurations that violate hard constraints, and among the feasible ones, the one with the highest score is selected. The system was evaluated on a real institutional instance, composed of 14 courses, 63 classes, 141 instructors, 37 rooms and 563 lectures, under three regimes of soft constraint rigidity. In the reference configuration, it achieved 99.11% allocation without any violation of mandatory constraints, with convergence already in the first construction attempts, suggesting that fail-first ordering accounts for most of the gain, with marginal contribution from diversification. Unlike proposals validated only on instances processed offline, PlanIFica has been operating in production since September 2024, with the generation module incorporated in 2026, and this was evaluated on the campus's actual planning data, which provides empirical evidence of practical applicability.

Keywords: Automated Timetable Generation. Multi-start Constructive Heuristic. Weighted Evaluation Function. Educational Scheduling. Timetabling.

RESUMEN
La elaboración manual de horarios académicos es una actividad de elevada complejidad combinatoria, sujeta a conflictos de asignación y uso ineficiente de la infraestructura institucional. Este trabajo presenta el módulo de generación automatizada de horarios del PlanIFica, sistema web en operación en el Instituto Federal de Educación, Ciencia y Tecnología de Rondônia (IFRO), Campus Porto Velho Calama. La generación emplea heurística constructiva voraz multi-start con función de evaluación ponderada: las asignaciones se ordenan por el principio fail-first, múltiples cuadrículas candidatas se construyen descartando las configuraciones que violen restricciones duras y, entre las viables, se selecciona la de mayor puntuación. El sistema fue evaluado sobre instancia institucional real, compuesta por 14 cursos, 63 grupos, 141 docentes, 37 ambientes y 563 clases, bajo tres regímenes de rigidez de las restricciones blandas. En la configuración de referencia, alcanzó 99,11% de asignación sin ninguna violación de restricción obligatoria, con convergencia ya en los primeros intentos de construcción, sugiriendo que la ordenación fail-first responde por la mayor parte de la ganancia, con contribución marginal de la diversificación. A diferencia de propuestas validadas solo sobre instancias procesadas offline, PlanIFica opera en producción desde septiembre de 2024, con el módulo de generación incorporado en 2026, y este fue evaluado sobre los datos reales de planificación del campus, lo que confiere evidencia empírica de aplicabilidad práctica.

Palabras clave: Generación Automatizada de Horarios. Heurística Constructiva Multi-start. Función de Evaluación Ponderada. Escalonamiento Educacional. Timetabling.

1 INTRODUÇÃO

A elaboração de horários acadêmicos constitui um dos problemas clássicos de escalonamento no contexto educacional, envolvendo a alocação de disciplinas, docentes, turmas e ambientes dentro de um conjunto de restrições institucionais e operacionais. Conhecido na literatura como *University Course Timetabling Problem*, o problema consiste em atribuir eventos a intervalos de tempo e locais sujeitos a diversas restrições (Davison; Kheiri; Zografos, 2025). Trata-se de um problema computacional "NP-difícil", de elevada complexidade combinatória, que exige a satisfação integral das restrições duras e a minimização das violações de restrições brandas; ainda assim, a maior parte das instituições permanece dependente do escalonamento manual, atividade demorada e frequentemente associada à alocação ineficiente de recursos (Zhu; Zhou; Luo, 2026). No Instituto Federal de Educação, Ciência e Tecnologia de Rondônia (IFRO) - *Campus* Porto Velho Calama, local onde se desenvolveu o presente trabalho, a grade é montada manualmente a cada semestre sobre uma instância de 14 cursos, 63 turmas, 141 docentes, 37 ambientes e 563 aulas. O responsável concilia, sem verificação automática de sobreposição, a disponibilidade declarada de cada docente, a compatibilidade entre disciplina e ambiente, a distribuição da carga horária semanal por turma e as preferências pedagógicas do campus. Disso decorrem três consequências recorrentes: tempo de elaboração elevado e concentrado em poucos servidores; conflitos identificados apenas após a publicação da grade, o que exige retrabalho; e ocupação ineficiente dos ambientes, cuja utilização não é avaliada de forma global. Qualquer alteração pontual, como o afastamento de um docente ou a criação de uma turma, obriga à revisão de toda a grade.

Experiências recentes indicam que soluções automatizadas reduzem substancialmente esse esforço e que a disponibilização de uma interface gráfica com indicadores de desempenho facilita a interpretação dos resultados e o apoio à decisão administrativa (El Machichi *et al.*, 2026; Ozkan *et al.*, 2025). Persiste, contudo, no contexto institucional analisado, a carência de uma solução que automatize a geração de horários respeitando as restrições duras e, simultaneamente, contemple as preferências pedagógicas e operacionais próprias do campus, condição que soluções genéricas atendem de forma limitada. Chen *et al.* (2021) observam que as metodologias consolidadas na literatura são pouco utilizadas em ambientes reais, possivelmente em razão de sua flexibilidade limitada diante das regras específicas de cada instituição.

Diante dessa lacuna, foi desenvolvido o "PlanIFica", um sistema web voltado ao gerenciamento e à geração automatizada de horários de aula, fundamentado em uma heurística construtiva gulosa *multi-start* com função de avaliação ponderada. Heurísticas gulosas constituem alternativa adequada a métodos de força bruta e evolutivos, que demandam tempo de execução elevado, e podem ser

diversificadas mediante a aplicação de múltiplos critérios de construção sobre uma mesma instância, selecionando-se ao final a solução de melhor qualidade (Coşar; Say; Dökeroğlu, 2023). O diferencial desta proposta reside na origem dos dados de avaliação. Boa parte da literatura valida suas contribuições sobre instâncias sintéticas. Mesmo os conjuntos padronizados de origem real, como as trinta instâncias da *International Timetabling Competition* 2019, extraídas do sistema UniTime (Müller; Rudová; Müllerová, 2025), são processados de forma offline, desvinculados da operação institucional. O PlanIFica, por sua vez, é um sistema em operação contínua desde setembro de 2024, e o módulo de geração aqui descrito foi avaliado sobre os dados reais de planejamento do *campus*. Reconhece-se, em contrapartida, que a avaliação se restringe a uma única instância institucional; para não interferir na operação, as execuções experimentais foram realizadas em ambiente de homologação, sobre cópia integral da base de produção.

Este trabalho tem como objetivo apresentar o módulo de geração automatizada de horários do PlanIFica, descrevendo a heurística implementada, os critérios de avaliação adotados e os resultados observados na avaliação conduzida. Como objetivos específicos, busca-se verificar a capacidade do método de produzir grades viáveis em escala institucional real, verificar a ausência de conflitos de alocação, avaliar o efeito da rigidez das restrições brandas sobre a completude da grade e delimitar as fronteiras operacionais do método, em particular, o efeito da ordenação *fail-first* sobre o ganho atribuível à diversificação *multi-start* e o custo computacional de cada construção em escala real.

O artigo está organizado da seguinte forma: a Seção 2 apresenta o referencial teórico sobre *timetabling*, restrições de escalonamento e heurísticas construtivas; a Seção 3 descreve a metodologia adotada; a Seção 4 detalha o desenvolvimento, incluindo a arquitetura do sistema, a modelagem do problema, o funcionamento da heurística e os critérios de avaliação; a Seção 5 apresenta e discute os resultados; e a Seção 6 expõe as conclusões e perspectivas futuras.

2 REFERENCIAL TEÓRICO

O problema de geração de horários acadêmicos consiste na distribuição de recursos educacionais, como disciplinas, docentes, turmas e ambientes, em intervalos de tempo previamente definidos, respeitando um conjunto de regras institucionais. Inserido no campo da otimização combinatória, exige a conciliação simultânea de disponibilidade docente, capacidade física dos ambientes, carga horária das disciplinas e conflitos entre turmas (Schaerf, 1999; Burke; Petrovic, 2002). Do ponto de vista computacional, Even, Itai e Shamir (1975) demonstraram que mesmo uma versão elementar do problema de horários é NP-completa, resultado que se estende às formulações usuais. Essa característica, somada ao crescimento do espaço de busca à medida que novas restrições

e recursos são adicionados, inviabiliza a aplicação de métodos exatos em cenários reais e consolida as abordagens heurísticas como alternativa para a construção eficiente de soluções (Schaerf, 1999). Revisão recente da literatura sobre o *University Course Timetabling Problem* confirma o predomínio de meta-heurísticas e abordagens híbridas nas propostas publicadas na última década (Abdipoor *et al.*, 2023).

A qualidade e a validade das soluções dependem do tratamento adequado das restrições, classificadas em duras (*hard constraints*) e brandas (*soft constraints*) (Burke; Petrovic, 2002). As restrições duras representam condições obrigatórias de validade da grade, como a impossibilidade de um docente ser alocado em mais de uma aula no mesmo período, a proibição de sobreposição de turmas e a limitação física dos ambientes, de modo que a violação de qualquer uma delas compromete a viabilidade da solução. Já as restrições brandas associam-se à qualidade organizacional: balanceamento da carga ao longo da semana, redução de intervalos ociosos, preferência por blocos contínuos e melhor aproveitamento dos espaços (Carter; Laporte, 1998), além da estabilidade de ambientes e do atendimento a horários preferenciais dos docentes (Ozkan *et al.*, 2025). Como a fronteira entre as duas categorias não é rígida na prática institucional, o tratamento equilibrado de ambas é determinante para a qualidade de qualquer sistema de geração automatizada (Burke; Petrovic, 2002).

Entre as abordagens heurísticas, destacam-se as construtivas, que edificam a solução progressivamente a partir de decisões locais, adicionando novos elementos à solução parcial e respeitando restrições de viabilidade. Essa característica facilita o controle de conflitos e torna a abordagem adequada ao *timetabling* (Feo; Resende, 1995). A heurística gulosa, uma de suas formas mais comuns, realiza a cada passo a escolha localmente mais vantajosa e, embora não garanta o ótimo global, apresenta bons resultados práticos quando combinada com mecanismos de avaliação (Feo; Resende, 1995), a custo computacional inferior ao de métodos de força bruta e evolutivos (Coşar; Say; Dökeroğlu, 2023). Sua eficácia depende, contudo, da ordem de processamento: pelo princípio *fail-first*, atribuir primeiro os elementos mais propensos a falhar reduz conflitos nas etapas seguintes (Haralick; Elliott, 1980; Burke; Newall, 2004). Complementarmente, estratégias *multi-start* ampliam a exploração do espaço de busca ao executar múltiplas tentativas independentes de construção, cada uma com diferentes ordens de processamento ou critérios de escolha (Feo; Resende, 1995; Martí; Resende; Ribeiro, 2013).

Segundo Martí, Resende e Ribeiro (2013), os métodos *multi-start* mais consolidados alternam duas fases: a geração de uma solução e sua posterior melhoria por busca local, retendo-se em cada iteração global a melhor solução obtida. Nessa família situa-se o GRASP (*Greedy Randomized*

Adaptive Search Procedure), no qual a construção é gulosa e randomizada, com cada elemento sorteado de uma lista restrita formada pelos melhores candidatos segundo um critério guloso, e a solução construída é em seguida refinada por busca local (Feo; Resende, 1995). Trabalhos recentes demonstram, porém, a viabilidade de diversificação puramente construtiva, aplicando múltiplos critérios de ordenação a uma mesma instância e selecionando ao final o melhor resultado (Coşar; Say; Dökeroğlu, 2023). Na mesma direção, a combinação de estratégias gulosas com mecanismos de seleção adaptativa tem produzido resultados competitivos no escalonamento de cursos universitários, superando meta-heurísticas tradicionais na minimização de violações de restrições brandas (Zhu; Zhou; Luo, 2026). No plano dos sistemas, tais soluções costumam incorporar meta-heurísticas, abordagens híbridas ou hiper-heurísticas (Chen *et al.*, 2021), e sua comparabilidade apoia-se usualmente em conjuntos padronizados de instâncias e resultados de referência (Ceschia; Di Gaspero; Schaerf, 2023; Müller; Rudová; Müllerová, 2025). Esses conjuntos adotam modelo de restrições próprio, ao qual o PlanIFica não se ajusta sem reimplementação, uma vez que suas restrições derivam de regras de negócio específicas do *campus*, como a vedação de blocos que cruzam turnos. A avaliação recai, por essa razão, sobre a instância institucional real.

3 METODOLOGIA

A pesquisa é de natureza aplicada e caráter experimental, com abordagem quantitativa na avaliação do algoritmo. O estudo compreende o desenvolvimento do módulo de geração automatizada do PlanIFica e sua avaliação sobre uma instância institucional real, sob diferentes configurações de rigidez das restrições brandas.

A primeira etapa consistiu no levantamento das regras que regem a montagem da grade no IFRO, *Campus* Porto Velho Calama, junto ao setor responsável pelo planejamento acadêmico. Foram identificadas as condições obrigatórias de validade da grade e as preferências pedagógicas e operacionais do *campus*, distinção que orientou a separação entre restrições duras e restrições brandas adotadas no restante do trabalho.

A segunda etapa traduziu essas regras em um modelo computacional, definindo os elementos do problema (aulas, *slots*, turmas, docentes e ambientes), os vínculos entre eles e a restrição associada a cada um. O modelo resultante é apresentado no Quadro 1.

A terceira etapa compreendeu a implementação do módulo de geração como componente da camada de processamento do sistema já em operação. Optou-se por heurística construtiva gulosa *multi-start* com função de avaliação ponderada, escolha justificada pelo custo computacional inferior ao de

métodos exatos e evolutivos e pela previsibilidade exigida em ambiente de produção. O funcionamento do algoritmo e os critérios de avaliação implementados são detalhados na Seção 4.

A quarta etapa extraiu a instância real de planejamento do campus referente ao período em vigor, cuja dimensão foi apresentada na Seção 1, e a submeteu ao módulo de geração em ambiente de homologação, réplica do ambiente de produção com a mesma versão do código e cópia integral da base de dados, de modo a não interferir na operação regular do sistema. Definiu-se como configuração de referência o modo balanceado com tempo limite de 600 segundos, idêntico à configuração em uso pela instituição em produção. Para isolar o efeito da função de avaliação sobre o comportamento do algoritmo, a execução foi repetida sob três regimes de rigidez das restrições brandas: critérios desativados, em que a função de avaliação não orienta a escolha entre candidatos; critérios neutros, com os pesos-base ativos, correspondente à configuração de referência; e critérios obrigatórios, em que todas as restrições brandas são elevadas ao nível máximo e passam a atuar como restrições duras.

A quinta etapa registrou, para cada regime, o número de aulas alocadas, os conflitos gerados, o total de tentativas de construção, o tempo limite configurado e o custo médio por tentativa. As execuções ocorreram em um único ambiente computacional. Registram-se como limitações do protocolo a execução única de cada regime e a não uniformidade dos tempos-limite entre eles, razão pela qual a comparação é conduzida sobre o custo médio por tentativa, normalizado pelo tempo.

Cabe ressaltar que o custo médio por tentativa foi estimado como a razão entre o tempo limite e o número de tentativas executadas, uma vez que todas as execuções encerraram por esgotamento do tempo. O ambiente de homologação possui a mesma configuração de hardware e software do ambiente de produção.

4 DESENVOLVIMENTO

Esta seção detalha o módulo de geração automatizada de horários do PlanIFica. A Subseção 4.1 apresenta a arquitetura do sistema em que o módulo se insere; a 4.2 formaliza a modelagem do problema; a 4.3 descreve o funcionamento da heurística construtiva gulosa *multi-start*; e a 4.4 expõe os critérios que orientam a avaliação das grades geradas.

4.1 ARQUITETURA DO SISTEMA

O PlanIFica foi desenvolvido como aplicação web estruturada em três camadas: apresentação, processamento e persistência. As camadas separam responsabilidades, seguindo o princípio de reúso e baixo acoplamento característico dos padrões de projeto de software (Gamma *et al.*, 1994), organização amplamente adotada em aplicações corporativas (Fowler, 2002).

A camada de apresentação, construída com *HTML*, *CSS* e *JavaScript*, disponibiliza as funcionalidades de cadastro, visualização e gerenciamento de turmas, disciplinas, docentes, ambientes e grades horárias, além do acompanhamento da geração automatizada.

A camada de processamento concentra as regras de negócio, que incluem validação de dados, gerenciamento das restrições acadêmicas e execução da heurística. Foi adotado o padrão *Model-View-Controller* (MVC) provido pelo *framework* PHP CodeIgniter 4; o motor de geração, núcleo da contribuição deste trabalho, integra-se a essa camada como componente próprio, ao qual a camada de controle fornece os parâmetros de execução: tempo limite, pesos e níveis de aplicação dos critérios.

Por fim, a camada de persistência, apoiada no sistema gerenciador de banco de dados MySQL, armazena os dados de docentes, turmas, disciplinas, ambientes e horários gerados, garantindo integridade, rastreabilidade e reutilização das informações em novas execuções.

4.2 MODELAGEM DO PROBLEMA

O problema tratado consiste na atribuição de cada aula a um conjunto de *slots* temporais e a um ambiente compatível, respeitando a não sobreposição de docentes, turmas e ambientes. Um *slot* representa a unidade básica de tempo disponível para alocação, definido por dia da semana, horário inicial e horário final. Essa formulação corresponde à variante *course timetabling* descrita por Schaerf (1999), cujo levantamento estabelece a não sobreposição de docentes, turmas e salas como restrições básicas do problema. A modelagem adotada no PlanIFica preserva esse núcleo e acrescenta os elementos exigidos pelas regras do campus, cuja descrição consta do Quadro 1.

Quadro 1. Elementos do modelo de geração de horários do PlanIFica

Elemento	**Descrição**	**Restrição principal**
Docente	Professor vinculado à disciplina e à turma	Não pode ministrar mais de uma aula no mesmo *slot*
Turma	Grupo de estudantes associado a um conjunto de disciplinas	Não pode ter duas disciplinas alocadas no mesmo *slot*
Disciplina	Componente curricular com carga horária semanal definida	A carga horária semanal deve ser integralmente cumprida
Ambiente	Sala de aula ou laboratório com disponibilidade limitada	Não pode ser utilizado por duas turmas no mesmo *slot*
Slot	Intervalo de tempo disponível pertencente a um único turno	Deve respeitar os turnos institucionais e os horários permitidos da turma

Fonte: Elaborado pelos autores, 2026

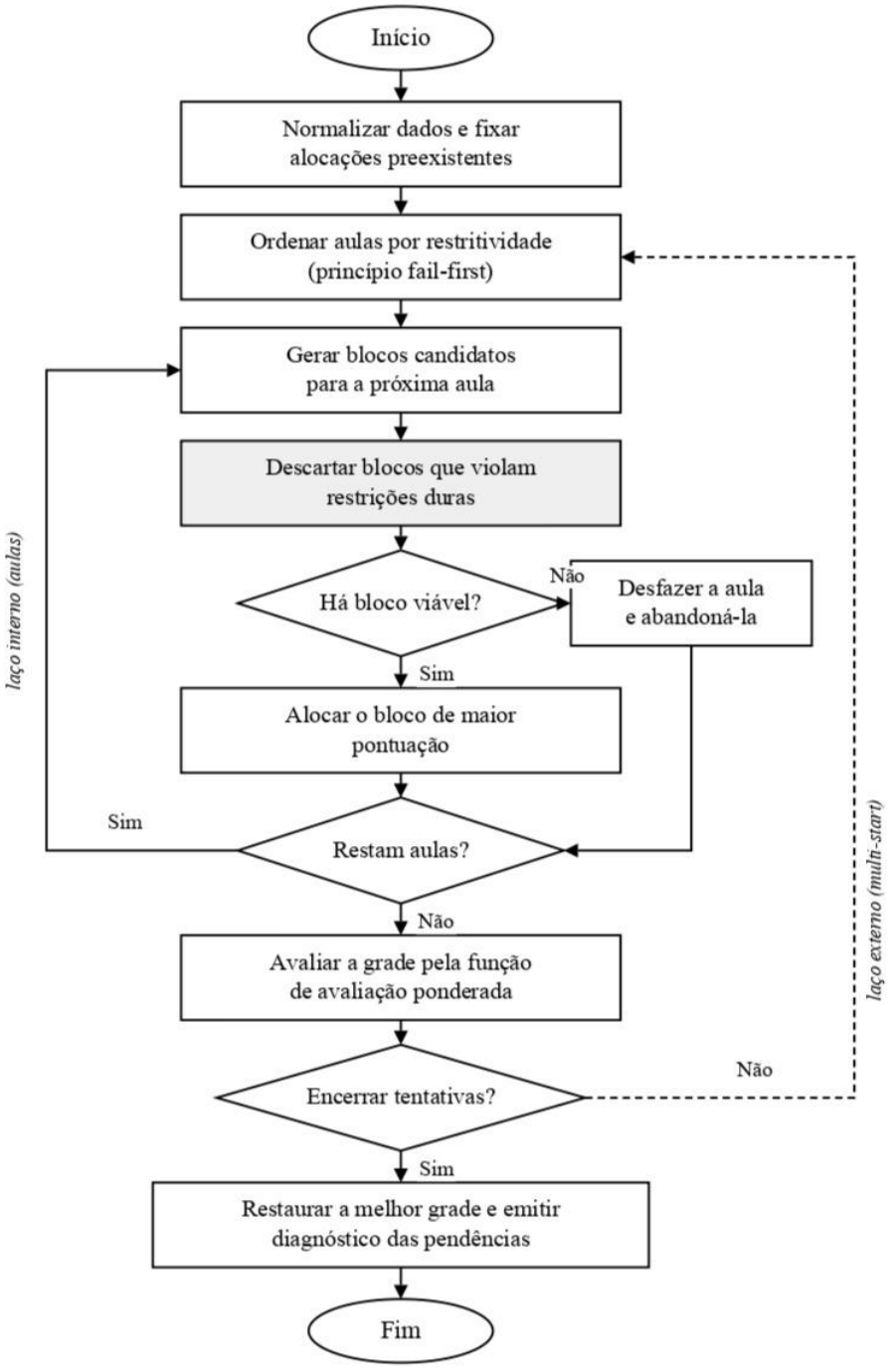
Cada turma possui um conjunto de *slots* permitidos, dentro dos quais a heurística realiza alocações, e um subconjunto de *slots* preferenciais, que orientam a distribuição das aulas segundo as

preferências institucionais. O conjunto de ambientes admissíveis de uma aula é derivado de sua disciplina, por vínculo direto ou por herança de grupos de ambientes; aulas cuja disciplina não possua vínculo cadastrado não dispõem de candidatos e são reportadas como não alocáveis. Além das condições de viabilidade, o sistema considera critérios de qualidade, como distribuição equilibrada da carga, redução de espaços ociosos e melhor aproveitamento da infraestrutura, incorporados por meio da função de avaliação ponderada descrita na Seção 4.4.

4.3 FUNCIONAMENTO DA HEURÍSTICA CONSTRUTIVA MULTI-START

A geração é realizada por uma heurística construtiva gulosa *multi-start* com função de avaliação ponderada, que constrói soluções factíveis de forma incremental e seleciona, entre múltiplas tentativas, a de melhor desempenho organizacional. O PlanIFica adota, em formulação simplificada, a lógica *multi-start* descrita na Seção 2: repetidas construções gulosas com variação controlada na ordenação e na seleção de candidatos, sem fase subsequente de busca local, distinguindo-se tanto do esquema clássico de duas fases (Martí; Resende; Ribeiro, 2013) quanto do GRASP (Feo; Resende, 1995). Cada grade gerada é avaliada por uma função ponderada e a de maior pontuação é selecionada, privilegiando baixo custo computacional e previsibilidade. O fluxo completo é sintetizado na Figura 1.

Figura 1. Fluxograma da geração automatizada de horários do PlanIFica



Fonte: Elaborado pelos autores, 2026

O processo inicia com a normalização dos dados cadastrados (disciplinas, docentes, turmas, ambientes e períodos), fixando as alocações preexistentes como estado-base ao qual cada tentativa retorna, conforme os passos iniciais da Figura 1. As aulas pendentes são ordenadas por restritividade segundo o princípio *fail-first* (Haralick; Elliott, 1980; Burke; Newall, 2004), em duas camadas hierárquicas: primeiro as turmas, pela soma da restritividade de suas aulas, e depois as aulas dentro de cada turma. A restritividade de uma aula combina, em ordem decrescente de magnitude, o inverso do

número de *slots* permitidos da turma e do número de ambientes compatíveis, a carga horária semanal, o número de docentes vinculados e a existência de limite diário de alocação; o termo de domínio temporal é o de maior peso quando a turma dispõe de poucos *slots*. Trata-se, portanto, de um *fail-first* aproximado, com componente relevante de carga horária.

A construção ocorre de forma gulosa e incremental, correspondendo ao laço interno da Figura 1. A cada passo, o tamanho do bloco é imposto pelo mínimo entre o limite diário e a carga restante; geram-se todos os pares (bloco contíguo × ambiente admissível), descartam-se os que violam restrições duras e pontuam-se os sobreviventes. Se nenhum bloco viável restar para a carga pendente, a aula é revertida por inteiro e abandonada, sem alocação parcial persistida. Não há retrocesso entre aulas: abandonar uma nunca remove outra já alocada. Cabe registrar que, quando a carga semanal de uma aula não cabe em um único dia e a tentativa é uma das três primeiras, o módulo de geração antecipa a construção bloco a bloco pela enumeração de planos completos de encontros disjuntos, ordenados por qualidade e aplicados integralmente, com reversão entre planos e truncamento em 250 blocos candidatos, 180 combinações e 120 planos (limites definidos empiricamente para manter o tempo de construção abaixo de 60s). Esse caminho alternativo, por não alterar a lógica geral do método, foi omitido da Figura 1 em favor da legibilidade.

A estratégia *multi-start* é implementada por múltiplas execuções independentes do processo construtivo, conforme discutido na Seção 2, e corresponde ao laço externo da Figura 1. A tentativa inicial é estritamente determinística: a lista de candidatos é truncada em um elemento e o bloco de maior pontuação é sempre escolhido, de modo que a primeira solução corresponde à construção gulosa pura e serve de linha de base para as demais. Nas tentativas seguintes, o bloco é sorteado uniformemente entre os k melhores candidatos, com semente derivada do índice da tentativa, da aula e da carga restante, logo, reprodutível. A ordenação também é perturbada, mas apenas em classes de congruência do índice da tentativa (reforço da carga horária quando $t \equiv 1 \bmod 3$ e deslocamento pseudoaleatório entre turmas quando $t \equiv 4 \bmod 5$) (classes escolhidas empiricamente); nas demais iterações a ordem repete a da tentativa inicial. A diversificação decorre, portanto, majoritariamente do sorteio entre candidatos, e não da reordenação, condição cujo efeito prático é examinado na Seção 5.1.

O sistema prevê três modos de processamento: rápido, balanceado e criterioso. Esses modos definem a amplitude do sorteio (três, seis ou dez melhores candidatos) e o teto de tentativas. Registra-se, contudo, que na versão avaliada a configuração é fixada em balanceado, de modo que todos os resultados aqui reportados correspondem a $k = 6$; os demais modos permanecem inativos. O laço encerra-se pelo primeiro dos três critérios: teto de tentativas derivado do tempo limite (⌊tempo limite / 10⌋ tentativas), esgotamento do tempo limite ou cancelamento pelo usuário. Esgotado o tempo durante

uma construção, as aulas restantes permanecem intocadas e a grade truncada ainda é avaliada, podendo tornar-se incumbente. Ao final, restaura-se a grade de maior pontuação e só então se produz, sobre ela, a lista das aulas incompletas com o respectivo diagnóstico.

Quadro 2. Pseudocódigo da heurística construtiva gulosa multi-start

```
Entrada: aulas, pesos, tempo_limite
Saída:   melhor grade, aulas não alocadas
1  normalizar aulas; base ← alocações preexistentes
2  melhor ← base; p* ← −∞; t ← 0
3  enquanto houver tempo e tentativas e não houver cancelamento faça
4     grade ← base
5     ordenar aulas por restritividade (fail-first, perturbada conforme t)
6     para cada aula faça
7        marco ← grade
8        enquanto houver carga pendente e houver tempo faça
9           candidatos ← blocos contíguos × ambientes admissíveis
10          descartar candidatos que violam restrições duras
11          se não houver candidato então
12             grade ← marco; abandonar aula; sair do enquanto
13          senão se t = 0 então
14             bloco ← candidato de maior pontuação
15          senão
16             bloco ← sorteio entre os k melhores candidatos
17          fim se
18          grade ← grade + bloco
19       fim enquanto
20    fim para
21    p ← avaliar(grade, pesos)
22    se p > p* então melhor ← grade; p* ← p fim se
23    t ← t + 1
24 fim enquanto
25 listar aulas incompletas de melhor com diagnóstico
26 retornar melhor, aulas não alocadas
```

Fonte: Elaborado pelos autores, 2026

4.4 CRITÉRIOS DE AVALIAÇÃO DA SOLUÇÃO

Após a construção de cada grade, o PlanIFica avalia a solução por meio de uma função de pontuação ponderada, que permite comparar as soluções das diferentes tentativas e selecionar a mais adequada ao contexto acadêmico. Essa abordagem é reconhecida como eficaz para lidar com as múltiplas restrições concorrentes do *timetabling* (Carter; Laporte, 1998). Durante a construção, são descartados antes da pontuação os blocos que violam restrições de recurso, como conflitos de docente, turma ou ambiente e impedimentos cadastrados, e os que contrariam regras institucionais, como a vedação de um docente atuar nos três turnos do mesmo dia e a proibição de blocos que cruzem turnos. A função de avaliação atua, portanto, apenas sobre soluções estruturalmente válidas, mensurando sua qualidade organizacional com base nos critérios sintetizados no Quadro 3. Duas dessas categorias, estabilidade de ambientes e atendimento a preferências docentes, foram incorporadas por decisão de

projeto, a partir das regras levantadas junto ao setor de planejamento do *campus*. Tratamento convergente é adotado por Ozkan *et al.* (2025) em aplicação a uma escola de negócios, na qual a estabilidade de ambientes e as preferências de horário dos docentes são incorporadas simultaneamente ao modelo de otimização, a primeira como restrição integrada, e a segunda como termo de maximização, em contraste com abordagens anteriores que as tratavam em dois estágios.

Quadro 3. Categorias de critérios da função de avaliação ponderada do PlanIFica

Categoria	Critérios considerados	Peso relativo
Completude da grade	Aulas com carga semanal integral versus incompletas	Maior peso: prioridade do sistema
Aderência ao horário preferencial	Uso de *slots* preferenciais da turma; distância dos alternativos	Intermediário-alto: penaliza a distância do preferencial
Preferências docentes	Atendimento aos horários preferidos dos professores	Intermediário: reforçado no perfil de conforto docente
Compactação de blocos	Tamanho dos blocos; aulas isoladas; blocos entre turnos	Intermediário: favorece a concentração de blocos
Estabilidade e uso de ambientes	Reaproveitamento e distribuição equilibrada de ambientes	Intermediário-baixo
Janelas e descanso docente	Intervalos ociosos; descanso mínimo entre turnos e dias	Intermediário-baixo: reforçado no perfil de conforto docente

Fonte: Elaborado pelos autores, 2026

Cada critério corresponde a um conjunto de pontuações específicas: positivas para situações desejáveis e negativas para as que se deseja evitar. Essas pontuações são somadas ao final de cada tentativa. A completude da grade recebe o maior peso relativo, refletindo a prioridade do sistema em alocar a totalidade da carga horária, ainda que isso implique desvios em critérios de menor prioridade.

Além dos pesos-base, o sistema permite configurar a intensidade de aplicação de cada critério por meio de cinco níveis (0 a 4), que variam entre a desconsideração total e a aplicação como restrição obrigatória. Em níveis intermediários, a pontuação é escalada proporcionalmente à intensidade configurada, permitindo ao gestor calibrar a importância relativa de cada fator. No nível máximo, o critério passa a atuar como restrição rígida: soluções que o violem são descartadas durante a construção, ainda que válidas sob os demais aspectos. Esse mecanismo permite que restrições originalmente brandas assumam caráter obrigatório conforme definido pela instituição, sem alteração na lógica do algoritmo. Burke e Petrovic (2002) apontam a customização de funções de custo por múltiplos critérios como direção de pesquisa, à qual esse mecanismo se alinha.

5 RESULTADOS E DISCUSSÕES

Esta seção apresenta os resultados obtidos na execução do PlanIFica sobre a instância descrita na Seção 3 e discute o desempenho da heurística construtiva gulosa *multi-start* implementada.

5.1 VIABILIDADE POR CONSTRUÇÃO E CONVERGÊNCIA DA BUSCA

O algoritmo não gerou nenhum conflito de docente, turma, ambiente, turno ou intervalo em nenhuma das três configurações testadas. O resultado evidencia uma característica central do módulo de geração: a viabilidade é assegurada pela forma como a solução é construída, e não por otimização posterior. Como a construção rejeita, a cada passo, qualquer alocação que viole uma restrição dura, a grade nasce válida. A consequência observável é que a ausência de conflitos independe da configuração, do número de tentativas e do resultado final, uma garantia determinística, alinhada ao tratamento das restrições duras diretamente no processo construtivo (Schaerf, 1999). Para um sistema em produção, essa previsibilidade é decisiva: o gestor nunca recebe uma grade com sobreposições.

Na configuração de referência, o algoritmo alocou 558 das 563 aulas (99,11%) e encontrou essa melhor solução já nas primeiras tentativas de construção, sem que as seguintes a superassem. A Tabela 1 apresenta o diagnóstico da execução.

Tabela 1. Diagnóstico da execução de referência

Métrica	Valor
Aulas alocadas	558 de 563 (99,11%)
Conflitos gerados	0
Tentativas de construção executadas	14
Tentativas com alocação integral	0
Tempo limite configurado	600 s
Custo médio por tentativa	≈ 43 s

Fonte: Elaborado pelos autores, 2026

Esse comportamento evidencia a eficácia da ordenação pelo princípio *fail-first*. Como a tentativa inicial é estritamente gulosa e determinística (Seção 4.3), ela funciona como linha de base do *multi-start*: ao alocar primeiro as aulas mais restritas, essa primeira construção já alcança uma solução de alta qualidade, e as tentativas seguintes, diversificadas pelo sorteio entre candidatos, apenas exploram alternativas de qualidade semelhante. Em outras palavras, a diversificação contribui pouco nesta instância, porque a heurística de ordenação já conduz a uma boa solução logo de início, resultado coerente com a literatura, que aponta a ordem de processamento das variáveis mais restritas como fator determinante da qualidade em métodos construtivos (Haralick; Elliott, 1980), e em particular com os

achados sobre adaptação de ordenações heurísticas em problemas de escalonamento educacional (Burke; Newall, 2004).

5.2 O PAPEL DA FUNÇÃO DE AVALIAÇÃO

Executado sob os três regimes de rigidez descritos na Seção 3, o algoritmo produziu os resultados da Tabela 2.

Tabela 2. Comportamento do algoritmo sob três regimes de rigidez

Indicador	Critérios desativados	Critérios neutros	Critérios obrigatórios
Status da solução	Parcial	Parcial	Inviável
Aulas alocadas	555	558	0
Aulas não alocadas	8	5	563
Taxa de alocação	98,58%	99,11%	0,00%
Conflitos gerados	0	0	0
Tentativas executadas	23	14	65
Tempo limite configurado	300 s	600 s	300 s
Custo médio por tentativa	≈ 13,0 s	≈ 42,9 s	≈ 4,6 s

Fonte: Elaborado pelos autores, 2026

Os dados sugerem que a função de avaliação influencia a capacidade do algoritmo de alocar aulas, ressalvando-se que os regimes diferem também em tempo limite (300s e 600s) e em número de tentativas, e que cada regime foi executado uma única vez. Com os critérios desativados, o módulo de geração alocou menos aulas (555) do que com eles ativos em peso neutro (558): sem a orientação da função de avaliação, as aulas são distribuídas de forma mais fragmentada e os *slots* são ocupados de modo menos eficiente, o que acaba inviabilizando algumas alocações.

Com os critérios obrigatórios, o algoritmo entrou em colapso: nenhuma aula alocada e status inviável. Ao tratar toda preferência como restrição dura, a construção não encontra candidatos válidos, e nenhuma das 65 tentativas produziu grade parcial, resultado que demonstra o limite do mecanismo de descarte, pois condições excessivas esvaziam o espaço de soluções viáveis. Em conjunto, os três regimes evidenciam que o módulo de geração opera bem dentro de uma faixa intermediária de rigidez, na qual a função de avaliação orienta as escolhas sem bloquear a construção; fora dessa faixa, ou a completude cai ou a solução se inviabiliza.

5.3 CUSTO COMPUTACIONAL E LIMITES DA INSTÂNCIA

O número de tentativas executadas foi baixo em todos os regimes, muito abaixo do limite configurado. O custo de cada construção é o principal limitante do *multi-start* em escala real. Montar uma grade com 563 aulas exige avaliar, para cada aula, todos os blocos possíveis e os ambientes compatíveis antes de decidir, e esse custo cresce com o tamanho da instância, de modo que poucas tentativas cabem no tempo disponível. Como a melhor solução já surge nas primeiras tentativas (Seção 5.1), ampliar esse número traria pouco retorno. Mais promissor seria substituir parte das reconstruções por uma fase de busca local sobre a melhor solução obtida, ausente na formulação atual e presente no *multi-start* clássico (Martí; Resende; Ribeiro, 2013).

Das 563 aulas, 5 (0,89%) não foram alocadas na configuração de referência. O módulo de geração emitiu o diagnóstico de cada uma: em alguns casos, os *slots* da turma já estavam totalmente ocupados; em outros, havia *slots* livres, mas dispersos demais para formar blocos contínuos. Quatro dessas cinco aulas falharam também no regime de critérios desativados. Sob ordenações distintas, portanto, elas permaneceram inviáveis, o que aponta para uma limitação dos dados de entrada e não do algoritmo. O regime obrigatório nada informa nesse aspecto, já que nele nenhuma aula foi alocada.

5.4 SÍNTESE DO DESEMPENHO

A Tabela 3 reúne as propriedades do algoritmo observadas nos experimentos e a evidência que as sustenta.

Tabela 3. Síntese das propriedades algorítmicas observadas

Propriedade do algoritmo	Evidência nos dados
Viabilidade por construção	Nenhum conflito nos regimes com 555 e 558 aulas alocadas
Eficácia da ordenação *fail-first*	Convergência para a melhor solução nas primeiras tentativas
Função de avaliação favorece a completude	558 aulas com critérios ativos contra 555 sem critérios (execução única; tempos-limite distintos)
Limite do mecanismo de descarte	Colapso (0 aulas) no regime de critérios obrigatórios
Custo por construção como gargalo	Poucas tentativas (14 a 65) e ≈ 43 s cada na referência
Diagnóstico de inviabilidade	Quatro das cinco aulas falham também no regime desativado

Fonte: Elaborado pelos autores, 2026

As propriedades observadas convergem para um algoritmo cujo principal gargalo é o custo de cada construção. A taxa máxima de 99,11% de alocação confirma a adequação do método ao problema real, enquanto o colapso no regime obrigatório e o alto custo por tentativa delimitam suas fronteiras de operação. Cabe registrar que os resultados referem-se a uma única instância institucional e a

execuções não repetidas, o que restringe sua generalização e recomenda cautela na comparação direta com trabalhos validados sobre conjuntos padronizados (Ceschia; Di Gaspero; Schaerf, 2023).

6 CONCLUSÃO

Este trabalho apresentou o módulo de geração automatizada de horários do PlanIFica, sistema em operação no IFRO *Campus* Porto Velho Calama, descrevendo a heurística construtiva gulosa *multi-start* implementada e avaliando seu comportamento em um cenário institucional real, sob diferentes configurações de rigidez das restrições.

Os objetivos propostos foram atendidos. A heurística produziu grades viáveis em escala real, com taxa de alocação de 99,11% e sem violação de restrições duras em nenhuma das configurações testadas, evidência direta da capacidade de gerar grades válidas e de eliminar os conflitos de alocação que caracterizam o processo manual. Essa viabilidade, assegurada pela própria construção da solução e não por correção posterior, confere ao sistema a previsibilidade exigida em ambiente de produção. O mecanismo de níveis ajustáveis, por sua vez, confirmou a flexibilidade de adaptação às regras institucionais, permitindo tratar cada critério desde a desconsideração total até a obrigatoriedade sem qualquer alteração na lógica do algoritmo, fidelidade que se manteve mesmo no cenário extremo, no qual o sistema preferiu não gerar solução a produzir uma grade em desacordo com as condições impostas.

Quanto às fronteiras do método, os experimentos delimitaram três:

- A primeira fronteira é a convergência precoce: por efeito da ordenação fail-first, a construção gulosa determinística inicial já alcança solução de alta qualidade, de modo que a diversificação multi-start, na formulação puramente construtiva aqui adotada, acrescenta pouco nesta classe de instância.
- A segunda fronteira é o custo computacional de cada construção completa, que limita o número de tentativas cabíveis no orçamento de tempo e restringe a estratégia em instâncias maiores.
- A terceira fronteira é o limite do mecanismo de descarte, evidenciado pelo colapso da solução quando todos os critérios brandos são elevados à obrigatoriedade.

Reconhecem-se ainda, como limitações do estudo, a avaliação sobre uma única instância institucional, a execução única por regime, a ausência de comparação com conjuntos padronizados de instâncias e o fato de a amplitude do sorteio ter permanecido fixa em $k = 6$, uma vez que os demais modos de processamento não estão ativos na versão auditada.

Dessas observações emergem as perspectivas de evolução, em ordem de prioridade: (i) incorporar uma fase de busca local, a exemplo do *Tabu Search*, operando sobre a melhor solução obtida, conforme o esquema clássico do *multi-start* (Martí; Resende; Ribeiro, 2013); (ii) consolidar a evidência experimental por meio de execuções repetidas, padronização dos tempos-limite e testes de significância; e (iii) estender a avaliação a múltiplas instituições e a conjuntos públicos de instâncias.

O fato de o PlanIFica encontrar-se em operação real desde 2024, com o módulo de geração incorporado em 2026, confere ao estudo evidência concreta de aplicabilidade prática, indicando que a heurística construtiva gulosa *multi-start* constitui alternativa viável para a geração automatizada de horários acadêmicos em contextos institucionais com regras de negócio específicas.

FINANCIAMENTO

Agradecemos ao Instituto Federal de Educação, Ciência e Tecnologia de Rondônia (IFRO) pelo apoio ao projeto de pesquisa "Desenvolvimento de um Sistema de Gerenciamento e Geração Automatizada de Horários de Aula", financiado por meio do EDITAL Nº157/2025/CALAMA, DE 15 DE AGOSTO DE 2025. Agradecemos também ao Departamento de Pesquisa, Inovação e Pós-Graduação do IFRO – Campus Porto Velho Calama, pelo apoio financeiro à publicação deste trabalho, concedido por meio do Edital de Apoio à Comunicação Científica e Literária n° 43/2026/PVCAL.

REFERÊNCIAS

ABDIPOOR, S.; YAAKOB, R.; GOH, S. L.; ABDULLAH, S. Meta-heuristic approaches for the University Course Timetabling Problem. Intelligent Systems with Applications, v. 19, art. 200253, set. 2023. DOI: https://doi.org/10.1016/j.iswa.2023.200253.

BURKE, E. K.; NEWALL, J. P. Solving examination timetabling problems through adaption of heuristic orderings. Annals of Operations Research, v. 129, n. 1/4, p. 107-134, 2004. DOI: https://doi.org/10.1023/B:ANOR.0000030684.30824.08.

BURKE, E. K.; PETROVIC, S. Recent research directions in automated timetabling. European Journal of Operational Research, v. 140, n. 2, p. 266-280, 2002. DOI: https://doi.org/10.1016/S0377-2217(02)00069-3.

CARTER, M. W.; LAPORTE, G. Recent developments in practical course timetabling. In: BURKE, E.; CARTER, M. (ed.). Practice and Theory of Automated Timetabling II: PATAT 1997. Berlin; Heidelberg: Springer, 1998. p. 3-19. (Lecture Notes in Computer Science, v. 1408). DOI: https://doi.org/10.1007/BFb0055878.

CESCHIA, S.; DI GASPERO, L.; SCHAERF, A. Educational timetabling: problems, benchmarks, and state-of-the-art results. European Journal of Operational Research, v. 308, n. 1, p. 1-18, 2023. DOI: https://doi.org/10.1016/j.ejor.2022.07.011.

CHEN, M. C.; SZE, S. N.; GOH, S. L.; SABAR, N. R.; KENDALL, G. A survey of university course timetabling problem: perspectives, trends and opportunities. IEEE Access, v. 9, p. 106515-106529, 2021. DOI: https://doi.org/10.1109/ACCESS.2021.3100613.

COŞAR, B. M.; SAY, B.; DÖKEROĞLU, T. A new greedy algorithm for the curriculum-based course timetabling problem. Düzce Üniversitesi Bilim ve Teknoloji Dergisi, v. 11, n. 2, p. 1121-1136, 2023. DOI: https://doi.org/10.29130/dubited.1113519.

DAVISON, M.; KHEIRI, A.; ZOGRAFOS, K. G. Modelling and solving the university course timetabling problem with hybrid teaching considerations. Journal of Scheduling, v. 28, n. 2, p. 195-215, 2025. DOI: https://doi.org/10.1007/s10951-024-00817-w.

EL MACHICHI, N.; KHARROUBI, F.; MASSOUS, H.; OUALI, S.; EL YOUSSFI, A. Genetic algorithm-based university course timetabling: a practical optimization framework with graphical interface. International Journal of Computer Engineering and Data Science (IJCEDS), v. 5, n. 2, p. 9-25, jul. 2026. Disponível em: https://www.ijceds.com/ijceds/article/view/141. Acesso em: 03 set. 2026.

EVEN, S.; ITAI, A.; SHAMIR, A. On the complexity of time table and multi-commodity flow problems. In: ANNUAL SYMPOSIUM ON FOUNDATIONS OF COMPUTER SCIENCE (FOCS), 16., 1975, Berkeley. Proceedings [...]. Berkeley: IEEE, 1975. p. 184-193. DOI: https://doi.org/10.1109/SFCS.1975.21.

FEO, T. A.; RESENDE, M. G. C. Greedy randomized adaptive search procedures. Journal of Global Optimization, v. 6, n. 2, p. 109-133, 1995. DOI: https://doi.org/10.1007/BF01096763.

FOWLER, M. Patterns of enterprise application architecture. Boston: Addison-Wesley Professional, 2002.

GAMMA, E.; HELM, R.; JOHNSON, R.; VLISSIDES, J. Design patterns: elements of reusable object-oriented software. Boston: Addison-Wesley Professional, 1994.

HARALICK, R. M.; ELLIOTT, G. L. Increasing tree search efficiency for constraint satisfaction problems. Artificial Intelligence, v. 14, n. 3, p. 263-313, 1980. DOI: https://doi.org/10.1016/0004-3702(80)90051-X.

MARTÍ, R.; RESENDE, M. G. C.; RIBEIRO, C. C. Multi-start methods for combinatorial optimization. European Journal of Operational Research, v. 226, n. 1, p. 1-8, 2013. DOI: https://doi.org/10.1016/j.ejor.2012.10.012.

MÜLLER, T.; RUDOVÁ, H.; MÜLLEROVÁ, Z. Real-world university course timetabling at the International Timetabling Competition 2019. Journal of Scheduling, v. 28, n. 2, p. 247-267, 2025. DOI: https://doi.org/10.1007/s10951-023-00801-w.

OZKAN, A.; ULUCAN, A.; DIRIK, C.; ATICI, K. B. University course timetabling with multi-section courses, room stability and lecturer preferences: an application in a business school. Computational Management Science, v. 22, n. 1, art. 3, 2025. DOI: https://doi.org/10.1007/s10287-025-00529-2.

SCHAERF, A. A survey of automated timetabling. Artificial Intelligence Review, v. 13, n. 2, p. 87-127, 1999. DOI: https://doi.org/10.1023/A:1006576209967.

ZHU, Z.; ZHOU, Y.; LUO, Q. A hyper-heuristic algorithm based on genetic and greedy strategy for university course scheduling problem. Applied Soft Computing, v. 186, pt. C, art. 114150, 2026. DOI: https://doi.org/10.1016/j.asoc.2025.114150.