57º SBPO
SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL
Gramado • RS
DE 5 A 9 DE OUTUBRO DE 2025

ALGORITMO HÍBRIDO *ITERATED LOCAL SEARCH* E *SIMULATED ANNEALING* PARA O PROBLEMA DE TABELA-HORÁRIO DE JOGOS DA SUPERLIGA BRASILEIRA DE VOLEIBOL

Filipe Pádua Ribeiro, Edmar Hell Kampke, Geraldo Regis Mauri
Universidade Federal do Espírito Santo
Alto Universitário, s/n, Campus Universitário, Alegre – ES, 29500-000
filipe.p.ribeiro@edu.ufes.br, edmar.kampke@ufes.br,
geraldo.mauri@ufes.br

RESUMO

Este trabalho apresenta uma solução baseada no algoritmo híbrido *Iterated Local Search* com *Simulated Annealing* para resolver o Problema de Tabela-Horário de Jogos da Superliga Brasileira de Voleibol. Este problema consiste em encontrar a melhor organização dos jogos do campeonato para que a distância percorrida pelas equipes seja mínima, reduzindo os gastos com o deslocamento e aumentando o tempo disponível para o descanso e a preparação dos atletas. O método proposto para a resolução do problema se baseia na meta-heurística *Iterated Local Search*, que a partir de uma solução ótima local, iterativamente sofre perturbações. As soluções obtidas após a perturbação são enviadas para o *Simulated Annealing*, responsável por realizar a busca local. Neste trabalho, foram obtidos resultados consideravelmente melhores do que a solução atualmente utilizada pela Superliga Brasileira de Voleibol, apresentando, em média, soluções 48,24% melhores.

PALAVRAS CHAVE. Superliga Brasileira de Voleibol, Tabela-Horário de Jogos, *Iterated Local Search*, *Simulated Annealing*, Meta-heurísticas.

MH – Meta-heurísticas; OD – Otimização Discreta

ABSTRACT

This work presents a solution based on the hybrid algorithm Iterated Local Search with Simulated Annealing to address the Schedule Timetabling Problem for Volleyball Games in the Brazilian League. This problem involves finding the best organization of championship games to minimize the distance traveled by teams, thereby reducing transportation costs and increasing the time available for rest and athlete preparation. The proposed method for solving the problem is based on the metaheuristic Iterated Local Search, which, starting from an optimal local solution, undergoes iterative perturbations. The solutions obtained after the disturbance are then sent to Simulated Annealing, which performs the local search. In this work, the proposed approach achieved significantly better results than the current solution used by the Brazilian League, yielding, on average, solutions that are 48.24% better.

KEYWORDS. Brazilian Volleyball Super League. Game Scheduling. Iterated Local Search. Simulated Annealing. Metaheuristics.

MH - Metaheuristics; OD - Discrete Optimization

DOI: 10.59254/sbpo-2025-212456
https://proceedings.science/p/212456?lang=pt-br

1. Introdução

Ligas profissionais de esportes são disputadas durante a maior parte do ano, e requerem que seus atletas, equipes e trabalhadores envolvidos em todo o processo de uma partida, mantenham um alto nível de desempenho. Diante disso, uma chave para esses níveis de rendimento é o calendário que as equipes jogam. Nenhum detentor de direitos de transmissão quer pagar grandes quantias apenas para que equipes pouco atraentes joguem em datas nobres. As equipes não querem ver seus grandes investimentos em jogadores e infraestrutura prejudicada por uma má programação dos jogos. Os torcedores, que fornecem boa parte da renda e popularidade para as ligas, também são muito afetados pelos horários dos jogos [Easton et al., 2001].

Portanto, a eficácia na programação de jogos não atende apenas às demandas dos envolvidos, como as equipes e detentores de direitos de transmissão, mas também o interesse e o apoio dos torcedores, elementos essenciais para a sustentabilidade e sucesso das ligas esportivas profissionais. Segundo Ribeiro e Urrutia [2007], calendários com tempos mínimos de viagem e que ofereçam condições semelhantes a todos os times são de grande interesse a todos os envolvidos em uma liga desportiva: times, patrocinadores, mídia e torcedores.

O Brasil é um país com dimensões continentais e, portanto, os campeonatos esportivos disputados nacionalmente sofrem com questões que envolvem o deslocamento das equipes. No caso do campeonato brasileiro de futebol, uma única viagem de Porto Alegre a Belém, por exemplo, possui um trajeto extenso e muitas paradas, devido à ausência de voos diretos para percorrer uma distância de aproximadamente 4.000 km. Com isso, a distância total percorrida torna-se uma variável importante a ser minimizada, de modo que se possa reduzir os custos de deslocamento e dar mais tempo aos jogadores para descanso e treinamento.

A Superliga Brasileira de Voleibol (SBV) é um exemplo de campeonato disputado nacionalmente, contando atualmente com a participação de 12 equipes tanto para a modalidade Masculina quanto para a Feminina. A SBV é composta por quatro fases: classificatória, quartas de final, semifinal e final. A fase classificatória é organizada em 22 rodadas (turno e returno) de seis partidas cada, com todos os 12 times jogando entre si, sendo que no returno as equipes jogam invertendo o mando de campo dos jogos do turno, seguindo o modelo de tabela espelhada (se um time x joga em casa contra o time y na rodada de número r do turno, a partida de número r do returno deverá ser entre as equipes y e x, realizada na casa da equipe y). As demais fases são organizadas de acordo com o resultado das fases anteriores.

Dado este volume de partidas e as restrições de mando de campo impostas pela organização da SBV, as equipes devem viajar constantemente, limitando o tempo disponível para a recuperação, o treino dos atletas e, consequentemente, reduzindo o seu desempenho.

O Problema de Tabela de Horário de Jogos (PTHJ), que é observado na criação das tabelas da SBV, é constituído por um conjunto de possibilidades de alocação das rodadas. As rodadas devem ser preenchidas de modo que todos os 12 times do campeonato joguem, e devem respeitar todas as restrições impostas.

Com isso, dada a importância da resolução do PTHJ e as dificuldades enfrentadas na disputa da SBV, observa-se a possibilidade e a oportunidade de se desenvolver uma maneira de otimizar a construção da tabela de jogos do campeonato, beneficiando em primeira escala os atletas, disponibilizando mais tempo para sua preparação e recuperação, melhorando o nível de competitividade do campeonato e, consequentemente, atraindo mais público e patrocinadores para as partidas. Além disso, não foram encontrados na literatura trabalhos que abordam o PTHJ da SBV.

2. Revisão de Literatura

Uma obra fundamental para compreender as abordagens do PTHJ foi desenvolvida por Ribeiro [2012], na qual é realizada uma revisão introdutória dos problemas fundamentais e da importância do PTHJ, tornando o trabalho um importante guia para pesquisas relacionadas a esta

DOI: 10.59254/sbpo-2025-212456
https://proceedings.science/p/212456?lang=pt-br

área. Além disso, foram discutidas diferentes meta-heurísticas, como a Busca Tabu e Algoritmos Genéticos, aplicadas por diversos autores para a resolução do PTHJ nos mais variados esportes, evidenciando a versatilidade de abordagens para o tratamento do problema.

Um dos trabalhos mais importantes acerca do PTHJ foi realizado por Russell e Leung [1994], aplicado na Liga de Beisebol do Texas (LBT) e resolvido pelos autores utilizando uma abordagem de construção da tabela de jogos por dois fatores, seguindo a geração de padrões casa-fora e atribuições de equipes a esses padrões. No artigo, o PTHJ é descrito como um problema notoriamente difícil de ser tratado. Além das restrições determinadas pela organização para a elaboração do calendário de jogos, existem diversos fatores adicionais que devem ser considerados, como a preferência de jogos em casa dos times em datas específicas e o equilíbrio da tabela para que os jogos entre dois times sejam igualmente espaçados durante toda a temporada.

Bao [2009] realizou um estudo inspirado nas regras da Liga Americana de Basquete (*National Basketball Assossiacion* - NBA), na qual utiliza o formato de torneio itinerante, com as equipes viajando constantemente para realizar seus jogos. Para isso, o autor desenvolveu três algoritmos baseados no *Independent Lower Bound* (ILB) para resolver esse problema e apresentou todas as principais restrições presentes na temporada regular da NBA, separadas em três categorias: estruturais, externas e de justiça.

Assim, o PTHJ possui grande aplicação prática e muitos desafios a serem considerados. Esses desafios tornam o uso de métodos heurísticos e meta-heurísticos uma alternativa atraente para encontrar soluções. Dentre as meta-heurísticas mais aplicadas para a solução de problemas de otimização estão as meta-heurísticas *Iterated Local Search* (ILS) e *Simulated Annealing* (SA).

Dentre os trabalho da literatura, que abordam a meta-heurística ILS, destaca-se o trabalho de Costa et al. [2012] que aplicou a ILS para a resolução do Problema do Torneio Itinerante com Locais Pré-Definidos (PTILP). Em seu artigo, os autores afirmam que o uso dessa meta-heurística apresenta um desempenho muito superior em comparação com heurísticas baseadas em programação inteira, e ainda aprimora as melhores soluções conhecidas para as instâncias utilizadas.

Em relação a meta-heurística SA, Anagnostopoulos et al. [2006] aplicaram a SA ao Problema do Torneio Itinerante (PTI) levando em consideração as características mais importantes da *Major League Baseball* (MLB) nos Estados Unidos. Nesse contexto, foi proposto um algoritmo do SA adaptado para o PTI, capaz de explorar tanto soluções viáveis quanto inviáveis, bem como suas vizinhanças.

Blum e Roli [2003] conduziram uma pesquisa acerca das diferentes meta-heurísticas existentes para a aplicação em problemas de otimização combinatória, como a Busca Tabu, SA, ILS, entre outras. Nesse estudo, foram levantadas as mais importantes meta-heurísticas utilizadas atualmente, bem como suas semelhanças, conceitos e diferenças. Com as meta-heurísticas estudadas, os autores concluíram que a hibridização das meta-heurísticas apresentam, para muitos problemas, as melhores soluções conhecidas e, portanto, é uma área de pesquisa extremamente promissora para a resolução de problemas de otimização combinatória.

Desse modo, a hibridização das meta-heurísticas ILS e SA é uma boa alternativa, que pode resultar em uma solução mais refinada. Portanto, assim é justificada a hibridização das meta-heurísticas ILS e SA para a resolução do PTHJ da SBV, que é o objetivo deste trabalho.

3. Modelagem do Problema

A modelagem matemática do PTHJ da SBV segue a que foi proposta por Souza et al. [2005], tendo a solução s avaliada pela função objetivo f, com as seguintes definições:

- T: conjunto de times participantes da competição;
- D_i: deslocamento total do time $i \in T$;

DOI: 10.59254/sbpo-2025-212456
https://proceedings.science/p/212456?lang=pt-br

- C: conjunto de restrições que descrevem o problema;
- R: conjunto de rodadas, sendo $|R| = (|T| - 1) * 2$;
- d_{ij}: distância percorrida pelo time $i \in T$ na rodada $j \in R$;
- τ_c: penalidade por desrespeitar a restrição $c \in C$;
- inv_c: número de vezes que a restrição $c \in C$ é desrespeitada;

Dessa forma, o modelo matemático é descrito da seguinte forma:

$$\text{Minimizar} \quad f(s) = \sum_{i \in T} D_i + \sum_{c \in C} \tau_c * inv_c \tag{1}$$

Sujeito a:

$$D_i = \sum_{j \in R} d_{ij}, \forall i \in T \tag{2}$$

A função objetivo (1), que deve ser minimizada, é calculada pela soma da distância total percorrida pelas equipes na solução s e baseada em penalidades descritas pelas seguintes restrições:

- Um time não pode jogar mais de uma vez na mesma rodada;
- Dois times jogarão entre si duas vezes, uma no turno e outra no returno, alternando o mando de campo entre os mesmos;
- Nenhum time deve jogar mais de seis vezes consecutivas dentro de casa;
- Nenhum time deve jogar mais de seis vezes consecutivas fora de casa;
- A diferença entre os jogos feitos em cada turno em casa e fora de casa de um time não pode ser maior que seis unidades.

As duas primeiras restrições foram definidas de acordo com as regras da SBV disponibilizadas no *website* da Confederação Brasileira de Vôlei (CBV): `https://cbv.com.br`. Por fim, as três últimas restrições foram definidas com base no trabalho de Souza et al. [2005], sendo o valor seis definido com base na tabela atual dos jogos da SBV. O modelo matemático termina com a restrição (2), que determina o valor da distância percorrida por cada equipe durante o campeonato.

O valor de τ_c foi definido como $1.000.000$ para inviabilizar a solução, caso essa viole alguma restrição $c \in C$.

4. Método Proposto

A meta-heurística ILS foi definida formalmente por Lourenço et al. [2003], apesar de já ter sido usada anteriormente por Stützle [1998]. No Algoritmo 1 é mostrado o pseudocódigo genérico da meta-heurística ILS.

Basicamente o algoritmo ILS consiste em quatro fases: construção da solução inicial (linha 2), refinamento da solução (linha 3), perturbação da solução refinada (linha 6) e refinamento da solução após perturbação (linha 7), sendo que todos os refinamentos são realizados por uma busca local.

A iteração do algoritmo é feita enquanto o laço de repetição não atingir o Max_{Iter}, perturbando repetidamente a solução S^{Melhor} (linha 6). A solução perturbada é melhorada por um

DOI: 10.59254/sbpo-2025-212456
https://proceedings.science/p/212456?lang=pt-br

Algoritmo 1: Algoritmo *ILS* proposto por Lourenço et al. [2003]

Entrada: Max_{Iter}
Saída: Solução S^{Melhor}

```
1  início
2      S^Inicial ← GeraSolucaoInicial();
3      S^Melhor ← BuscaLocal(S^Inicial);
4      FO^Melhor ← f(S^Melhor);
5      para i ← 1 até Max_Iter faça
6          S^Atual ← GeraPerturbacao(S^Melhor);
7          S^Atual ← BuscaLocal(S^Atual);
8          se f(S^Atual) < FO^Melhor então
9              S^Melhor ← S^Atual;
10             FO^Melhor ← f(S^Atual);
11         fim se
12     fim para
13 fim
```

procedimento de busca local (linha 7). A solução que atende o critério de aceitação é armazenada em S^{Melhor} (linhas 9 e 10), que é retornada ao final da execução.

Lourenço et al. [2003] evidenciam em seu artigo que a versatilidade e a eficiência da metaheurística ILS se dá pela sua característica de focar a procura de soluções em subespaços localmente ótimos, e não focar em todo o espaço de soluções. Os autores ainda enfatizam a vantagem do método em sua capacidade de escapar de mínimos locais, aumentando assim a probabilidade de encontrar mínimos globais.

4.1. Construção da solução inicial

Neste trabalho foi utilizada uma abordagem parcialmente aleatória, baseada na obra de Biajoli [2007], utilizando o método do polígono [Colbourne e Dinitz, 2007] para gerar a tabela dos jogos do primeiro turno e espelhá-la para gerar a tabela do segundo turno.

A inicialização do método do polígono é feita a partir de um vetor V de tamanho n (número de times) em que $V[i]$ representa um time. A execução começa com a escolha aleatória de um time base, inserindo-o na primeira posição do vetor V. Os demais times também são dispostos aleatoriamente nas posições restantes ($i = 2, ..., n$).

Em cada rodada r_k, $k = 1, ..., n-1$, o time base $i = 1$ confronta o time da posição $i = 2$, definindo o jogo $T_{V[1]}, T_{V[2]}$. Os times da posições $i = 3, ..., (n/2)+1$ confrontam os times das posições $j = n - i + 3$, definindo os demais jogos $T_{V[i]}, T_{V[j]}$. Após a definição de todos os jogos da rodada r_k, os times das posições $i = 2, ..., n$ são rotacionados no sentido horário dentro do vetor, onde os times das posições $i = 3, ..., n$ passam para a posição $i - 1$, e o time da posição 2 passa para a posição n.

A seguir, a Figura 1 ilustra a aplicação desse método com um vetor de times, em que $n = 6$, sendo o time 3 escolhido para ser o time base.

Para garantir a viabilidade da solução inicial, a distribuição dos mandos de campo das partidas é feita de maneira alternada. A definição dos mandos de campo é baseada na ordenação dos times no vetor V e suas posições representadas pelo método dos polígonos, alternando o mando de campo apenas para as posições base $i = 1$ e seu adversário $i = 2$. Em rodadas de numeração ímpar, o time base $i = 1$ realiza seu confronto em casa contra o time da posição $i = 2$. Já para as rodadas de

DOI: 10.59254/sbpo-2025-212456
https://proceedings.science/p/212456?lang=pt-br

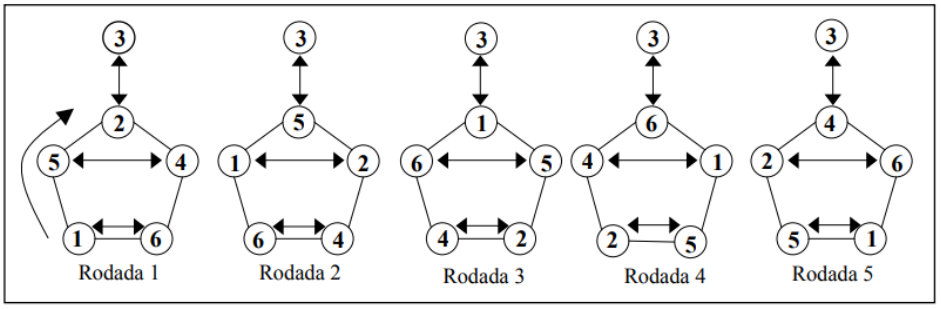


Figura 1: Ilustração para uma geração de rodadas com $n = 6$.
Fonte: Biajoli [2007].

numeração par, o mando de campo é invertido, com o time da posição $i = 2$ realizando o confronto em casa contra o time base de posição $i = 1$. Para os demais confrontos, fica definido que o time da posição ímpar sempre irá realizar seus confrontos em casa. Com isso, ao realizar o movimento de rotação no sentido horário, descrito pelo método do polígono, garante-se que a construção das partidas irá ser realizada com os mandos de campo alternados a cada rodada. Tomando como exemplo os confrontos definidos na Figura 1, temos as seguintes definições de mando de campo para as rodadas:

– Rodada 1: os times $3, 5, 6$ realizam seus confrontos em casa contra os times $2, 4, 1$, respectivamente;

– Rodada 2: os times $5, 1, 4$ realizam seus confrontos em casa contra os times $3, 2, 6$, respectivamente;

– Rodada 3: os times $3, 6, 2$ realizam seus confrontos em casa contra os times $1, 5, 4$, respectivamente;

– Rodada 4: os times $6, 4, 5$ realizam seus confrontos em casa contra os times $3, 1, 2$, respectivamente;

– Rodada 5: os times $3, 2, 1$ realizam seus confrontos em casa contra os times $4, 6, 5$, respectivamente;

Com isto, a solução inicial gerada pelo método ILS é sempre uma solução viável passível de melhorias por meio da busca local.

4.2. Perturbação

A perturbação no ILS é a etapa na qual a solução atual é modificada antes de ser enviada à busca local novamente. Essas modificações da solução visam escapar de ótimos locais e explorar novas regiões do espaço de busca.

As implementações de métodos de perturbação podem variar de acordo com o problema abordado. Em alguns casos, a perturbação é feita através da destruição parcial e reconstrução de uma solução. Entretanto, neste trabalho, a perturbação é realizada por meio da modificação parcial de uma solução, sempre executando, a cada iteração e de forma sequencial, uma quantidade definida de cada um dos três tipos de movimentos apresentados a seguir:

4.2.1. Troca de Ordem entre Rodadas

Para realizar este tipo de movimento, são escolhidas aleatoriamente duas rodadas $i, j \in \{1, ..., |R|/2\}$ $(i \neq j)$. Em seguida, suas ordens de execução são invertidas no turno e no returno, conforme ilustrado na Figura 2, em um exemplo com $n = 4$ e seis rodadas, onde foram selecionadas as rodadas 1 e 3 para serem permutadas.

A quantidade de execuções deste movimento é obtida pela multiplicação entre $|R|$ e o parâmetro de permutação de rodadas ρ.

DOI: 10.59254/sbpo-2025-212456
https://proceedings.science/p/212456?lang=pt-br

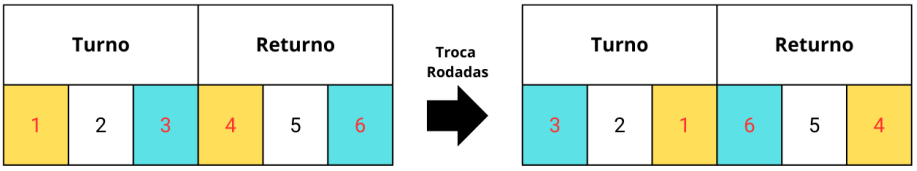


Figura 2: Exemplo do movimento Troca de Ordem entre Rodadas.

4.2.2. Inversão do Mando de Campo

O segundo movimento consiste em inverter o mando de campo de uma partida. Para isso, sorteia-se uma rodada do primeiro turno $j \in \{1, ..., |R|/2\}$ e um time $i \in T$. A partir desse sorteio, a partida do time i tem seu mando de campo invertido tanto na rodada j no turno quanto na rodada $j + |R|/2$ no returno. No exemplo ilustrado pela Figura 3, temos o time B sendo sorteado dentro de uma rodada e a configuração de suas partidas sendo invertidas entre turno e returno após o movimento.

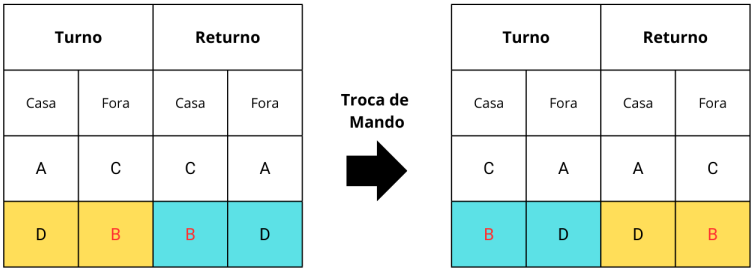


Figura 3: Exemplo do movimento Inversão de Mando de Campo.

A quantidade de execuções deste movimento de troca é obtida pela multiplicação do parâmetro de permuta de mando de campo γ com o total de jogos realizados no turno $(n/2 * |R|/2)$.

4.2.3. Permutação de Times nas Rodadas

O terceiro e último movimento visa modificar as partidas dentro das rodadas do campeonato. Para isso, dois times $p, q \in T$ $(p \neq q)$ são selecionados aleatoriamente e em seguida completamente permutados ao longo de todas rodadas do turno. Isso significa que todas as ocorrências de partidas envolvendo p são substituídas por q, e vice-versa, mantendo o mando de campo original do time substituído. Nas rodadas em que p e q se enfrentam, o mando de campo da partida é invertido.

Esse movimento pode ser ilustrado na Figura 4, onde os times A e D foram sorteados. No cenário original, o time A enfrenta o time C na primeira rodada e o time D na segunda, ambas como mandante (coloração branca) e encerra a terceira rodada jogando contra o time B fora de casa (coloração cinza). Por sua vez, o time D inicia a competição jogando em casa contra o time B, depois viaja para enfrentar o time A e, na terceira rodada, retorna ao seu mando de campo para enfrentar o time C. Após o movimento de permutação, o time A herda a configuração original da tabela do time D, realizando seus confrontos contra os times B, D e C, respectivamente. De maneira análoga, o time D assume a tabela original de A, enfrentando os times C, A e B, nesta ordem. Além disso, na rodada em que os times sorteados se enfrentam, apenas o mando de campo da partida é invertido.

De modo análogo aos movimentos anteriores, a quantidade de execuções dessa troca é obtida multiplicando n pelo parâmetro de perturbação de times λ.

DOI: 10.59254/sbpo-2025-212456
https://proceedings.science/p/212456?lang=pt-br

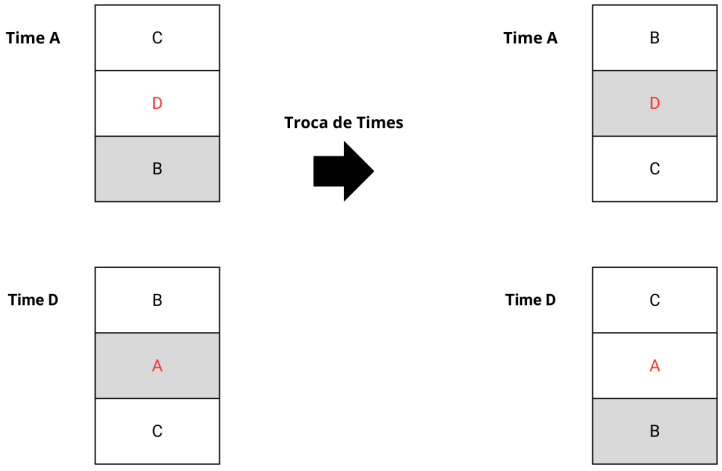


Figura 4: Exemplo do movimento Permutação de Times nas Rodadas.

4.3. Busca local

A estratégia de busca local que foi utilizada neste trabalho é a meta-heurística SA, que foi proposta por Kirkpatrick et al. [1983]. No Algoritmo 2 é mostrado o pseudocódigo genérico da meta-heurística SA, com os seguintes parâmetros: Solução inicial ($S_{inicial}$), Temperatura inicial ($T_{inicial}$), Temperatura final (T_{final}), Taxa de resfriamento (β) e o Número de soluções vizinhas (N_v) que são geradas a cada iteração.

O algoritmo inicia com a $T_{inicial}$ alta e é gradualmente resfriado até alcançar a T_{final}. A cada iteração, para cada temperatura T, são geradas N_v soluções vizinhas. Se a solução vizinha gerada apresentar valor de f melhor do que a solução atual, ela é atualizada. No entanto, se a solução vizinha apresentar valor de f pior, ela ainda pode ser aceita com uma probabilidade calculada como $e^{-\Delta f/T}$, sendo que Δf representa a diferença entre os valores de f da solução vizinha e da solução atual, e T é a temperatura atual. Quanto maior for a diferença entre Δf e menor a temperatura T, menor será a possibilidade de aceitar uma solução vizinha. No início, quando a temperatura está alta, o algoritmo aceita uma ampla gama de soluções. À medida que a temperatura diminui, apenas poucas soluções piores são aceitas, intensificando a busca em uma vizinhança específica. A cada iteração, a temperatura T é multiplicada pelo valor β, em que β é uma taxa com valor entre 0 e 1. O algoritmo termina quando a temperatura T for menor que T_{final}.

4.3.1. Geração de Soluções Vizinhas

A etapa de geração de vizinhos é essencial para o funcionamento do SA. Nela é definida como o espaço de soluções atual é explorado. Para isso, foram utilizados os movimentos descritos nas Subseções 4.2.1, 4.2.2 e 4.2.3. A cada iteração do SA, um desses três movimentos é sorteado aleatoriamente para ser executado uma única vez, gerando assim uma nova solução vizinha (linha 7 do Algoritmo 2).

5. Resultados Obtidos

Nesta seção são apresentados os resultados obtidos, através de experimentos computacionais, pelo método proposto (ILS+SA). As subseções seguintes detalham, respectivamente, os experimentos computacionais realizados, os resultados obtidos, e por fim a análise e a discussão desses resultados.

DOI: 10.59254/sbpo-2025-212456
https://proceedings.science/p/212456?lang=pt-br

Algoritmo 2: Algoritmo *SA* adaptado de Rocha [2013]

Entrada: Solução $S_{inicial}$, $T_{inicial}$, T_{final} = 0.01, β, N_v
Saída: Solução S^{Melhor}

```
início
    T ← T_inicial;
    S^Melhor ← S_inicial;
    S^Atual ← S_inicial;
    enquanto T > T_final faça
        para i ← 1 até N_v faça
            S^Vizinho ← GeraVizinho(S^Atual);
            Δf ← f(S^Vizinho) − f(S^Atual);
            se Δf < 0 então
                S^Atual ← S^Vizinho;
                se f(S^Vizinho) < f(S^Melhor) então
                    S^Melhor ← S^Vizinho;
                fim se
            senão
                Gera um número aleatório ρ ∈ [0, 1];
                se ρ < e^(−Δf/T) então
                    S^Atual ← S^Vizinho;
                fim se
            fim se
        fim para
        T ← T * β;
    fim enquanto
fim
```

5.1. Experimentos Computacionais

Foram realizados experimentos computacionais para avaliar a qualidade das soluções geradas pelo método proposto. As instâncias utilizadas para realizar os experimentos computacionais são as instâncias reais da SBV Masculina e Feminina das temporadas 21-22, 22-23, 23-24 e 24-25, resultando em oito instâncias.

Os parâmetros das meta-heurísticas foram calibrados utilizando o *software* de configuração automática *Iterated racing package* ($iRace$) [López-Ibáñez et al., 2016]. Para isso, são definidos os intervalos possíveis para os parâmetros e fornecidas as instâncias utilizadas no problema. Ao final da execução o $iRace$ exibe os valores obtidos pela calibração, como pode ser observado na Tabela 1. O parâmetro T_{final} foi definido com o valor 0,01, conforme apresentado no Algoritmo 2.

O algoritmo foi desenvolvido na linguagem de programação C++, compilado com *GCC Compiler* 6.3.0 e todos os testes foram executados em um computador com processador Intel Core i5 1.80 GHz, 16GB de memória RAM e sistema operacional Windows 11 Pro. Além disso, o método ILS+SA foi executado 10 vezes para cada instância com os parâmetros calibrados descritos acima, sendo que o número de iterações do ILS (Max_{Iter}) foi substituído pelo tempo computacional de cinco minutos (300 segundos).

5.2. Resultados Computacionais

Os resultados obtidos para cada instância estão descritos na Tabela 2, que é dividida entre as categorias masculina e feminina. Para cada instância, tem-se o menor valor de f obtido (f_{best})

DOI: 10.59254/sbpo-2025-212456
https://proceedings.science/p/212456?lang=pt-br

Tabela 1: Calibração dos Parâmetros.

Parâmetro	Tipo	Intervalo	Melhor
β	Real	[0.9, 0.999]	0.9362
N_v	Categórico	{10, 100, 300, 500}	500
$T_{inicial}$	Categórico	{10, 50, 100, 200}	100
ρ	Real	[0.3, 0.9]	0.3802
γ	Real	[0.3, 0.9]	0.8537
λ	Real	[0.3, 0.9]	0.3939

dentre as 10 execuções, a média (f_{avg}) dos resultados, o coeficiente de variação $\Delta = (((f_{agv} - f_{best}) \ / \ f_{avg}) \ * \ 100)$, calculado com base na diferença percentual entre f_{best} e f_{avg}, e por fim, o tempo (ω), em segundos, necessário para que o algoritmo encontrasse a solução f_{best}.

Tabela 2: Resultados Obtidos pelo ILS+SA.

	Instâncias	f_{best}	f_{avg}	$\Delta(\%)$	$\omega(seg.)$
M	21-22	72.140	75.287,4	4,18	167
	22-23	125.071	127.670,1	2,04	248
	23-24	77.837	80.994,1	3,90	16
	24-25	75.932	80.882,8	6,12	227
F	21-22	70.537	72.461,9	2,66	32
	22-23	77.695	79.278,7	2,00	158
	23-24	70.436	75.764,5	7,03	96
	24-25	76.198	80.755,0	5,64	46

É possível observar na Tabela 2 que, para as oito instâncias analisadas, em três delas o algoritmo foi capaz de encontrar a melhor solução em menos de um minuto. Além disso, em apenas duas instâncias a melhor solução foi encontrada nos últimos 100 segundos de execução, sugerindo que, na maioria dos casos, a convergência para uma boa solução ocorreu em um estágio inicial ou intermediário da execução.

Além disso, a coluna Δ demonstra que o algoritmo foi robusto e conseguiu convergir rapidamente para um espaço de busca promissor, uma vez que a maior variação entre a média e a melhor solução encontrada foi de apenas 7,03%. Isso indica uma estabilidade na exploração de soluções, reduzindo grandes oscilações entre os valores encontrados de f_{best} e f_{avg}.

5.3. Análise e Discussão

Afim de validar o método proposto, é importante comparar a eficácia de seus resultados com os resultados apresentados atualmente pela SBV, como evidenciado pela Tabela 3. Na coluna f_{sbv} temos o valor de f calculado para a tabela (solução) proposta pela SBV em cada instância do campeonato, enquanto a coluna f_{best} apresenta os valores obtidos pelo método ILS+SA proposto neste trabalho. Por fim, a coluna $\Delta = (((f_{sbv} - f_{best}) \ / \ f_{sbv}) \ * \ 100)$ apresenta a diferença percentual entre os resultados de f_{best} e f_{sbv}.

Dentre as instâncias, destaca-se a 22-23 da edição masculina. Os valores elevados de f_{sbv} e f_{best} estão relacionados ao fato de que o campeonato contava com a participação de um time (Rede Cuca) da região Nordeste, com sede em Fortaleza/CE. Com isso, os demais times constantemente realizavam viagens mais longas do que as habituais, saindo do eixo majoritariamente composto por times das regiões Sul e Sudeste, elevando a distância total percorrida pelas equipes. Já para a

DOI: 10.59254/sbpo-2025-212456
https://proceedings.science/p/212456?lang=pt-br

Tabela 3: Comparação de Resultados.

	Instâncias	f_{sbv}	f_{best}	$\Delta(\%)$
M	21-22	137.788	72.140	47,64
	22-23	207.342	125.071	39,68
	23-24	156.566	77.837	50,28
	24-25	164.309	75.932	53,79
F	21-22	137.503	70.537	48,70
	22-23	147.048	77.695	47,16
	23-24	139.873	70.436	49,64
	24-25	149.542	76.198	49,05
Média		154.996,3	71.715,4	48,24

Rede Cuca, um cronograma mal otimizado ocasionou em um gasto de tempo e recurso superior aos seus adversários, que dispuseram de uma janela maior para treino, preparação e recuperação dos atletas. Com o método proposto apresentando um valor de f quase 40% melhor, a SBV poderia implementar uma tabela mais justa e ponderada para todas as equipes, minimizando os possíveis impactos causados pela distância geográfica. Vale ressaltar que a Rede Cuca ficou na última colocação da SBV naquela edição, e consequentemente foi rebaixada, não retornando ao campeonato desde então.

Além disso, todas as instâncias apresentaram uma melhoria significativa de f quando comparadas à solução atualmente utilizada pela SBV. Mesmo na edição em que f_{sbv} mais se aproximou de f_{best}, a solução obtida ainda foi aproximadamente 40% melhor. No melhor dos casos, o ILS+SA gerou uma solução quase 54% superior à adotada pela SBV, sendo que, em média, os resultados do algoritmo se mostraram 48,24% mais eficientes que a solução usada pela SBV.

Portanto, é possível afirmar que o algoritmo proposto para a resolução do PTHJ da SBV apresentou-se extremamente satisfatório, obtendo em todos os casos soluções melhores do que as dispostas atualmente. Além da qualidade das soluções, o tempo de execução demonstrou-se prático e viável, tornando o método adequado para aplicações reais.

6. Conclusão

Neste trabalho foi abordado o Problema de Tabela-Horário de Jogos (PTHJ) da Superliga Brasileira de Voleibol (SBV). Foi proposto um método que utiliza a meta-heurística ILS juntamente com a meta-heurística SA na etapa de busca local. Aplicando três movimentos de perturbação baseados na troca de ordem entre as rodadas, a inversão do mando de campo das partidas e a permutação de times nas rodadas, buscou-se uma ampla exploração do espaço de soluções para que o algoritmo conseguisse encontrar soluções de qualidade.

O método proposto gerou resultados consistentemente superiores em comparação com os atualmente utilizados pela SBV, alcançando uma melhoria média de 48,24% e no melhor cenário, de 53,79%. Estes ganhos refletem uma melhoria significativa para todas as equipes do campeonato, reduzindo as distâncias e o tempo gasto com o deslocamento para as partidas, permitindo uma alocação mais eficiente dos recursos e promovendo uma melhoria no cenário competitivo da liga.

Deste modo, os resultados obtidos reforçam a relevância e a aplicabilidade de métodos para otimizar a tabela de jogos em campeonatos, destacando seu potencial para futuras implementações e aprimoramentos.

Agradecimentos: Os autores agradecem ao Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq (processo 303850/2022-1).

DOI: 10.59254/sbpo-2025-212456
https://proceedings.science/p/212456?lang=pt-br

Referências

Anagnostopoulos, A., Michel, L., Hentenryck, P. V., e Vergados, Y. (2006). A simulated annealing approach to the traveling tournament problem. *Journal of Scheduling*, 9(2):177–193.

Bao, R. (2009). *Time relaxed round robin tournament and the NBA scheduling problem*. Doutorado em engenharia industrial, Cleveland State University.

Biajoli, F. L. (2007). Novas heurísticas para o problema de geração de escalas de jogos para torneios esportivos. Mestrado em computação aplicada, Instituto Nacional de Pesquisas Espaciais.

Blum, C. e Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, 35(3):268—-308.

Colbourne, C. e Dinitz, J. (2007). *Handbook of combinatorial designs*. CRC press, Boca Raton, FL, USA.

Costa, F. N., Urrutia, S., e Ribeiro, C. C. (2012). An ILS heuristic for the traveling tournament problem with predefined venues. *Annals of Operations Research*, 194(1):137–150.

Easton, K., Nemhauser, G., e Trick, M. (2001). The traveling tournament problem description and benchmarks. In *Principles and Practice of Constraint Programming — CP 2001*, p. 580–584, Berlin, Heidelberg, Germany. Springer Berlin Heidelberg.

Kirkpatrick, S., Gelatt, C. D., e Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598):671–680.

López-Ibáñez, M., Dubois-Lacoste, J., Cáceres, L. P., Birattari, M., e Stützle, T. (2016). The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, 3:43–58.

Lourenço, H. R., Martin, O. C., e Stützle, T. (2003). Iterated local search. In Glover, F. e Kochenberger, G., editors, *Handbook of metaheuristics*, p. 320–353. Springer, Boston, MA, USA.

Ribeiro, C. C. (2012). Sports scheduling: Problems and applications. *International Transactions in Operational Research*, 19(1-2):201–226.

Ribeiro, C. C. e Urrutia, S. (2007). Heuristics for the mirrored traveling tournament problem. *European Journal of Operational Research*, 179(3):775–787.

Rocha, W. S. (2013). Algoritmo grasp para o problema de tabela-horário de universidades. Mestrado em informática, Universidade Federal do Espírito Santo.

Russell, R. A. e Leung, J. M. (1994). Devising a cost effective schedule for a baseball league. *Operations Research*, 42(4):614–625.

Souza, M. J. F., Mine, M. T., Silva, M. d. S. A., Ochi, L. S., e Pontes, R. C. V. (2005). Programação de jogos da primeira divisão do campeonato brasileiro de futebol por meio da metaheurística iterated local search. In *Anais do Simpósio de Pesquisa Operacional e Logística da Marinha*, volume 8, Rio de Janeiro, RJ, Brasil. CASNAV.

Stützle, T. (1998). Applying iterated local search to the permutation flow shop problem. Technical report, Citeseer.

DOI: 10.59254/sbpo-2025-212456
https://proceedings.science/p/212456?lang=pt-br