

Análise do Problema do Vencedor do Torneio Usando Fluxo Máximo

Maximus Borges da Rosa
Universidade Federal do Rio Grande do Sul

18 de maio de 2025

Resumo

Este trabalho apresenta uma análise do problema do vencedor do torneio, que consiste em determinar se uma equipe específica ainda tem chances matemáticas de vencer uma competição em andamento. A solução proposta utiliza o algoritmo Ford-Fulkerson de fluxo máximo em uma rede especialmente construída para representar os jogos restantes e as restrições de vitórias. São realizados experimentos variando parâmetros como número de equipes, viés de vitórias e fração de jogos restantes, determinando como esses fatores afetam a probabilidade da equipe de interesse vencer o torneio. Os resultados demonstram a eficácia da abordagem e fornecem insights sobre as condições que favorecem a possibilidade de vitória em diferentes cenários de competição.

1 Introdução

O problema do vencedor do torneio é uma questão de teoria dos grafos e possui aplicações práticas em competições esportivas: dada uma competição em andamento, é possível que determinada equipe ainda se torne campeã?

Formalmente, o problema pode ser definido da seguinte maneira:

- Temos n equipes em um torneio
- Cada equipe i já possui w_i vitórias
- Entre cada par de equipes i e j , há g_{ij} jogos ainda a serem disputados
- Queremos determinar se a equipe 1 ainda pode terminar como vencedora única, ou seja, com mais vitórias que qualquer outra equipe

À primeira vista, este problema parece exigir a enumeração de todas as possíveis combinações de resultados dos jogos restantes, o que seria computacionalmente inviável para torneios de tamanho razoável. No entanto, uma abordagem mais eficiente é possível através da teoria de fluxo em redes, permitindo resolver o problema em tempo polinomial.

Este trabalho apresenta uma solução baseada no algoritmo Ford-Fulkerson para fluxo máximo, seguindo a abordagem proposta inicialmente por Schwartz [1]. Além disso, realizamos uma análise experimental sistemática para entender como diferentes parâmetros do torneio afetam a probabilidade de uma equipe específica vencer.

2 O Problema do Vencedor do Torneio

2.1 Descrição Formal

Suponha que temos um torneio de n equipes e que elas já ganharam w_1, \dots, w_n vezes até agora. Para cada par de equipes, ainda temos g_{ij} jogos pela frente (g é simétrico). A equipe 1 ainda pode terminar em primeiro lugar, ou seja, ter o maior número de vitórias?

Para a equipe i , seja $r_i = \sum_j g_{ij}$ o número de jogos restantes. Precisamos que:

- i) a equipe 1 vença todos os seus r_1 jogos restantes, portanto, tem $w_1 + r_1$ vitórias, e
- ii) todas as outras equipes $i \in T = [2, n]$ vençam no máximo m_i jogos, dado por $w_i + m_i < w_1 + r_1$, i.e., $m_i = w_1 + r_1 - w_i - 1$ (excluindo empates).

Caso algum $m_i < 0$, a equipe 1 já não pode ganhar mais. Caso contrário, uma redução para um problema de fluxo é como segue. Cria um grafo com vértices $s, G = \binom{T}{2}, T, t$ e com os seguintes arcos:

- (s, g) para todo $g = (i, j) \in G$ de capacidade g_{ij} ,
- (g, i) e (g, j) para todo $g = (i, j) \in G$ de capacidade ∞ ,
- (i, t) para todo $i \in T$ de capacidade m_i .

Temos:

Lemma 1. *A equipe 1 ainda pode vencer se o grafo acima possui um fluxo $s-t$ que satura (i.e., de valor $\sum_{(i,j) \in G} g_{ij}$).*

2.2 Solução Através de Fluxo Máximo

A elegante solução para este problema utiliza um grafo de fluxo construído da seguinte maneira:

1. Criamos um vértice fonte s
2. Para cada par de equipes (i, j) com $i, j \geq 2$ (excluindo a equipe 1), criamos um vértice que representa os jogos restantes entre essas equipes
3. Criamos um vértice para cada equipe i com $i \geq 2$ (excluindo a equipe 1)
4. Criamos um vértice sumidouro t
5. Adicionamos arestas da fonte para cada vértice de jogo (i, j) , com capacidade igual a g_{ij}
6. Adicionamos arestas de cada vértice de jogo (i, j) para os vértices das equipes correspondentes i e j , com capacidade infinita
7. Adicionamos arestas de cada vértice de equipe i para o sumidouro, com capacidade igual ao máximo de vitórias adicionais permitidas para essa equipe: $w_1 + r_1 - w_i - 1$

O fluxo neste grafo representa uma atribuição de vitórias para os jogos restantes. A equipe 1 pode vencer o torneio se e somente se o fluxo máximo neste grafo for igual ao número total de jogos restantes entre as equipes 2 até n .

Algorithm 1 Verificação de Possibilidade de Vitória do Time 1

Require: Número de times n , vetor de vitórias w , matriz de jogos restantes g

Ensure: True se o time 1 pode vencer, False caso contrário

```
1:  $r_1 \leftarrow \sum_{j=2}^n g_{1j}$  ▷ Jogos restantes do time 1
2:  $maxWinsTeam1 \leftarrow w_1 + r_1$  ▷ Máximo possível de vitórias do time 1
3: for  $i \leftarrow 2$  to  $n$  do
4:   if  $w_i > maxWinsTeam1$  then return False ▷ Time  $i$  já tem mais vitórias que o
   máximo possível do time 1
5:   end if
6:    $maxAdditionalWins_i \leftarrow maxWinsTeam1 - w_i - 1$  ▷ Máximo de vitórias
   adicionais permitidas para o time  $i$ 
7: end for
8: Construir grafo de fluxo conforme descrito na Seção 2.2
9:  $maxFlow \leftarrow \text{FordFulkerson}(grafo)$ 
10:  $totalGames \leftarrow \sum_{i=2}^n \sum_{j=i+1}^n g_{ij}$  ▷ Total de jogos restantes entre times 2 a  $n$ 
    return  $maxFlow = totalGames$ 
```

3 Metodologia

3.1 Implementação

A solução foi implementada utilizando uma abordagem orientada a objetos. O código está organizado nas seguintes classes principais:

- **Tournament:** Encapsula os dados do torneio (número de equipes, vitórias atuais, jogos restantes) e implementa a lógica de verificação de possibilidade de vitória.
- **Graph:** Implementa a estrutura de grafo direcionado com capacidades nas arestas.
- **Edge:** Representa uma aresta no grafo, com origem, destino, capacidade e fluxo.

O algoritmo de Ford-Fulkerson foi implementado para calcular o fluxo máximo no grafo utilizando a estratégia de busca "Caminho mais gordo" (melhor desempenho obtido no trabalho 2), com uma fila de prioridade feita com um Max-Heap 8-ário (melhor valor obtido para k no trabalho 1). A implementação inclui uma verificação inicial das condições matemáticas necessárias antes de construir e processar o grafo de fluxo, o que permite descartar rapidamente casos impossíveis.

3.2 Geração de Testes

Para analisar o comportamento do algoritmo e entender como diferentes fatores afetam a probabilidade de vitória da equipe 1, foram gerados diversos conjuntos de testes usando scripts automatizados. Os testes variam os seguintes parâmetros:

- **Número de equipes (n):** Variando de 4 a 100 equipes, com incremento de 4. Para este conjunto de testes, foi gerado um torneio para cada valor de n , mantendo a fração de jogos realizados em 0.5 e o viés em 0.1.
- **Viés de vitórias (β):** Um valor entre -0.9 e 0.9 com incremento de 0.1, que determina a vantagem ou desvantagem inicial da equipe 1 em relação às demais. Um viés positivo significa que a equipe 1 começa com mais vitórias, enquanto um viés negativo indica o contrário. Para cada valor de viés, foram gerados 30 torneios aleatórios, fixando o número de equipes em 16 e a fração de jogos realizados em 0.5.
- **Fração de jogos realizados (α):** Um valor entre 0.1 e 0.9 com incremento de 0.1, que indica qual percentual dos jogos totais já foi realizado. Quanto maior este valor, menos jogos restam para serem disputados. Para cada valor de fração, foram gerados 30 torneios aleatórios, fixando o número de equipes em 16 e o viés em 0.1.
- **Combinação viés-fração:** Para entender a interação entre o viés de vitórias e a fração de jogos realizados, foi gerado um conjunto de testes combinando todos os valores de β (-0.9 a 0.9) com todos os valores de α (0.1 a 0.9). Para cada combinação, foram gerados 10 torneios aleatórios, mantendo o número de equipes fixo em 16.

No total, foram gerados aproximadamente 1.400 torneios de teste para cobrir adequadamente o espaço de parâmetros e obter resultados estatisticamente significativos. A execução desses testes permitiu coletar dados para analisar tanto o tempo de execução quanto a probabilidade de vitória da equipe 1 sob diferentes condições.

3.3 Cálculo das Probabilidades

As probabilidades de vitória apresentadas neste trabalho foram calculadas através de uma abordagem estatística empírica (simulação Monte Carlo), seguindo os seguintes parâmetros:

Para cada configuração específica, a probabilidade de vitória P foi calculada como:

$$P = \frac{W}{N} \quad (1)$$

onde W é o número de torneios nos quais a equipe 1 poderia vencer (resultado "sim") e N é o número total de torneios para aquela configuração específica ($N = 30$ para os testes de viés e fração individuais, e $N = 10$ para os testes combinados).

Por exemplo, se em 30 torneios aleatórios com viés de 0.5, o time 1 consegue vencer em 21 deles, então a probabilidade de vitória para $\beta = 0.5$ é $P(0.5) = \frac{21}{30} = 0.7$ ou 70%.

Esta abordagem de simulação Monte Carlo fornece uma estimativa empírica das chances de vitória sob diferentes condições, em vez de uma solução analítica exata. A geração de múltiplas instâncias para cada configuração de parâmetros permite identificar padrões estatísticos robustos e minimizar o impacto da aleatoriedade nas conclusões.

3.4 Métricas Analisadas

Para cada conjunto de testes, foram analisadas as seguintes métricas:

- **Tempo de execução:** O tempo necessário para determinar se a equipe 1 pode vencer o torneio.
- **Probabilidade de vitória:** A proporção de torneios gerados com uma determinada configuração onde a equipe 1 ainda pode vencer.

Os resultados foram coletados e organizados em arquivos CSV para posterior análise e visualização gráfica.

4 Resultados

4.1 Tempo de Execução vs. Número de Equipes

A primeira análise examina como o tempo de execução do algoritmo varia com o número de equipes no torneio.

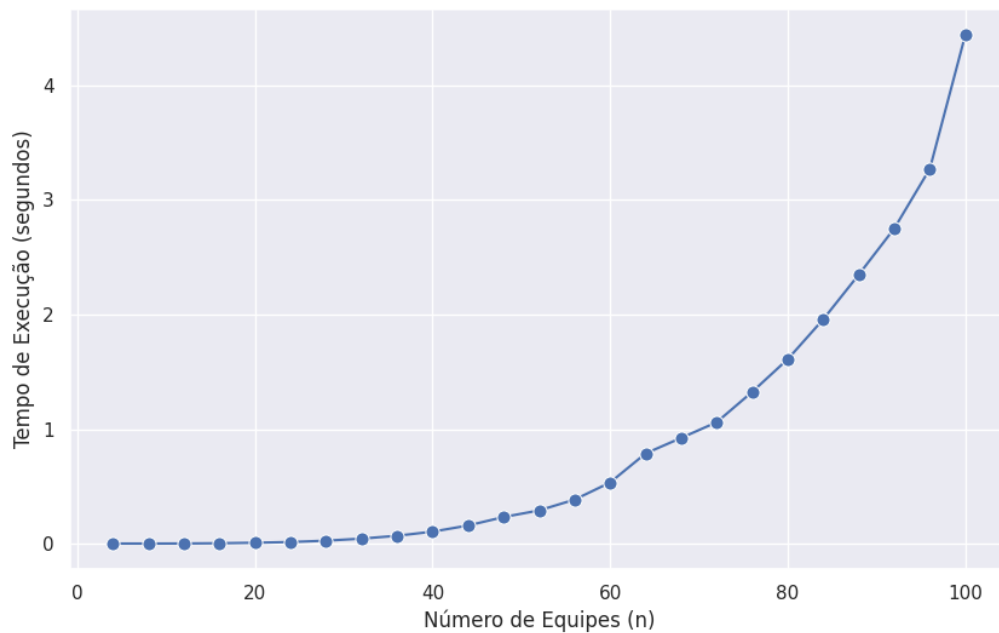


Figura 1: Tempo de execução em função do número de equipes

O gráfico mostra que o tempo de execução cresce de forma aproximadamente quadrática com o número de equipes, o que é esperado dado que o número de vértices e arestas no grafo de fluxo cresce quadraticamente com n .

4.2 Probabilidade de Vitória vs. Viés de Vitórias

Analisamos como o viés de vitórias da equipe 1 afeta sua probabilidade de ainda poder vencer o torneio.

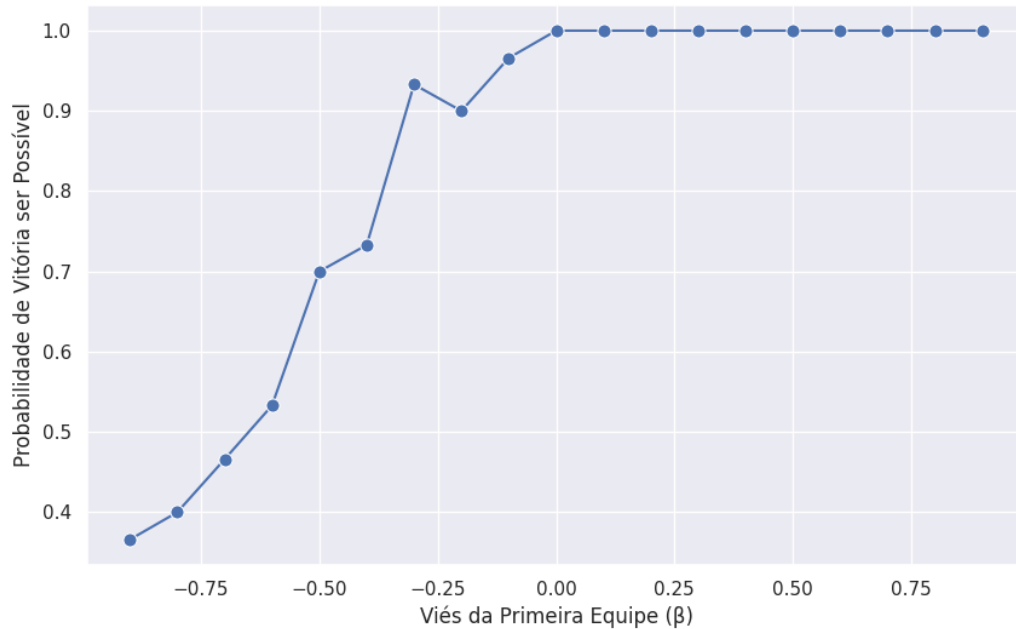


Figura 2: Probabilidade de vitória em função do viés de vitórias

Os resultados mostram uma clara correlação entre o viés de vitórias e a probabilidade da equipe 1 poder vencer. Quanto maior o viés positivo (maior vantagem inicial), maior a probabilidade de vitória. A curva apresenta, em geral, um comportamento que se assemelha a uma curva sigmóide, indicando uma transição gradual entre "praticamente impossível" até "praticamente garantido" em uma faixa específica de viés.

4.3 Probabilidade de Vitória vs. Fração de Jogos Realizados

Também analisamos como a fração de jogos já realizados afeta a probabilidade de vitória da equipe 1.

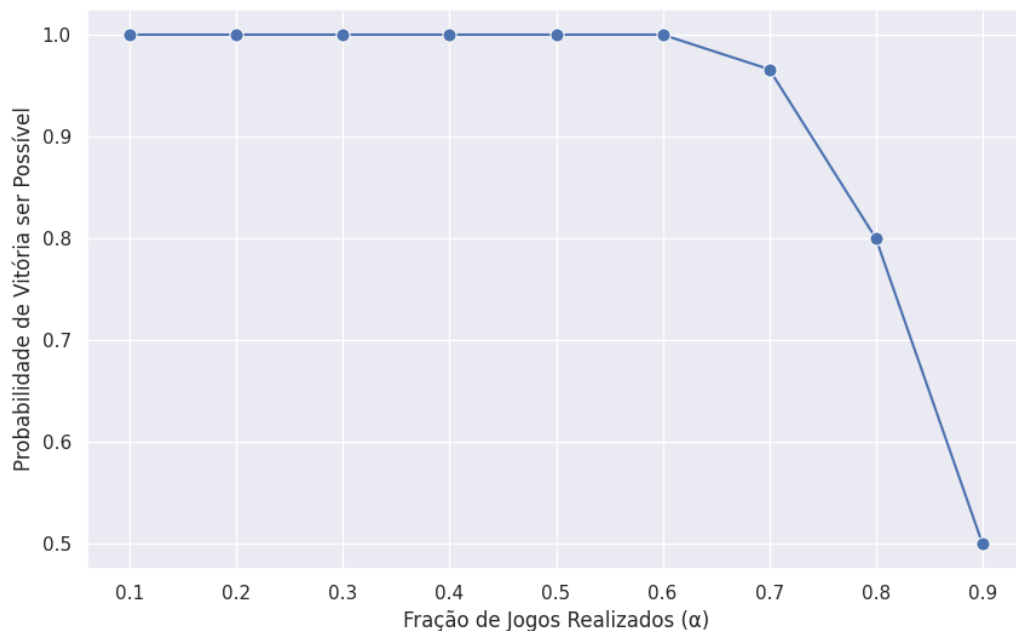


Figura 3: Probabilidade de vitória em função da fração de jogos realizados

Este gráfico mostra que quanto maior a fração de jogos já realizados (menos jogos restantes), menor a probabilidade da equipe 1 poder vencer, especialmente se ela não estiver em vantagem. Isto é intuitivo: quanto menos jogos restam, menos oportunidades existem para recuperar uma desvantagem ou criar uma vantagem.

4.4 Análise Combinada: Viés vs. Fração de Jogos

Finalmente, analisamos a interação entre o viés de vitórias e a fração de jogos realizados.

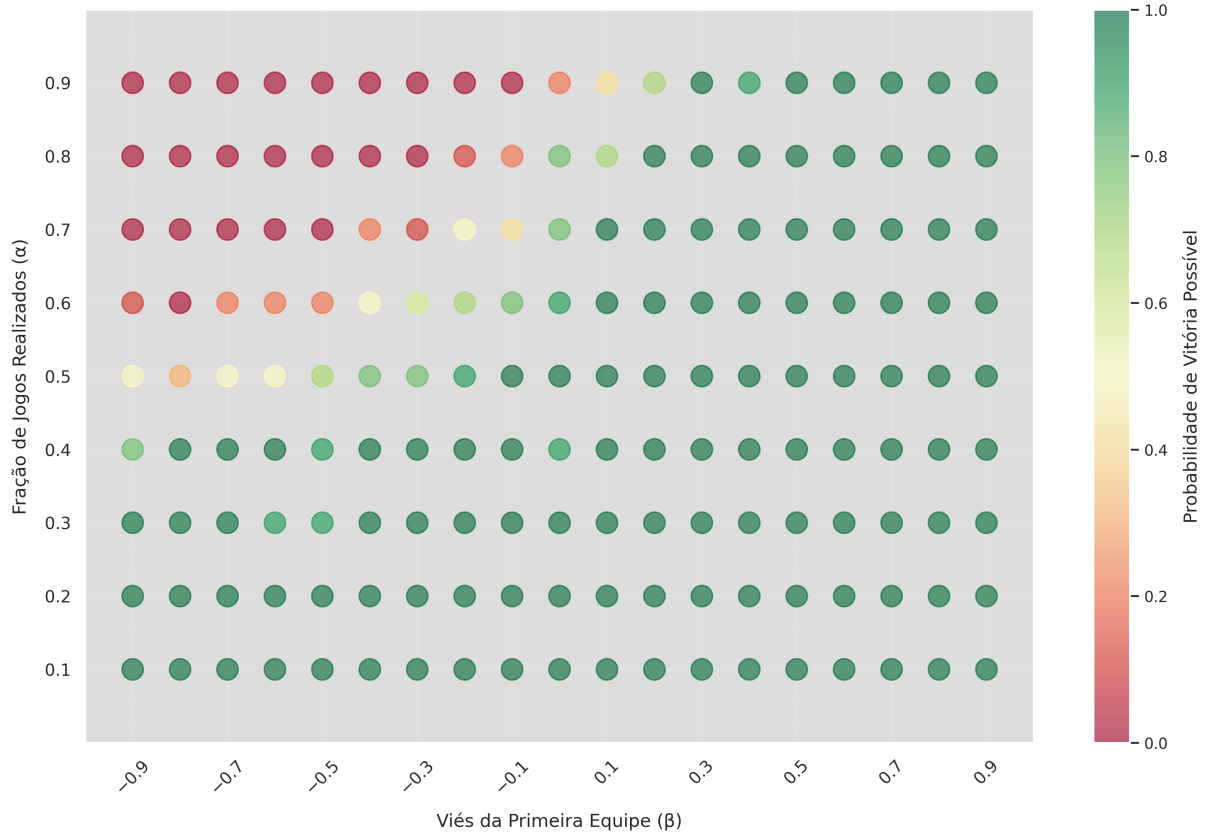


Figura 4: Mapa de calor da probabilidade de vitória em função do viés e da fração de jogos realizados

O mapa de calor revela padrões interessantes na interação entre estes dois fatores. Para valores extremos de viés, a fração de jogos realizados tem impacto reduzido na probabilidade de vitória. Contudo, para valores intermediários de viés, a fração de jogos realizados torna-se um fator crítico, determinando se a equipe ainda tem chances significativas de vencer.

5 Discussão

Os resultados experimentais obtidos revelam insights importantes sobre o problema do vencedor do torneio:

5.1 Eficiência do Algoritmo

O algoritmo baseado em fluxo máximo demonstra-se eficiente mesmo para torneios com grande número de equipes. O tempo de execução cresce de forma previsível e compatível com a complexidade teórica esperada para o algoritmo Ford-Fulkerson em grafos esparsos. A verificação inicial das condições matemáticas também contribui significativamente para a eficiência, permitindo descartar rapidamente casos impossíveis sem necessidade de construir e processar o grafo completo.

5.2 Fatores que Afetam a Probabilidade de Vitória

Os experimentos mostram claramente que tanto o viés de vitórias quanto a fração de jogos realizados têm impacto significativo na probabilidade de uma equipe ainda poder vencer o torneio:

- **Viés de Vitórias:** Como esperado, quanto maior a vantagem inicial da equipe 1 (viés positivo), maior sua probabilidade de poder vencer o torneio. A relação apresenta um comportamento sigmoidal, com uma região de transição relativamente estreita onde pequenas variações no viés causam grandes mudanças na probabilidade.
- **Fração de Jogos Realizados:** Quanto mais avançado o torneio (maior fração de jogos realizados), mais definida se torna a situação e menor a probabilidade de uma equipe que não está na liderança poder vencer. Este comportamento também é intuitivo: com menos jogos restantes, há menos oportunidades para mudar o status quo.
- **Interação entre Fatores:** O mapa de calor revela uma interessante interação entre estes fatores. Para uma equipe com viés de vitórias fortemente negativo, mesmo com poucos jogos realizados, a probabilidade de vitória é baixa. Inversamente, para uma equipe com viés fortemente positivo, mesmo com muitos jogos realizados, a probabilidade de vitória permanece alta.

6 Ambiente de Teste

Os testes foram executados em um processador **Intel Core i5-10210U** de 1.60 GHz com 12 GB de RAM, em um sistema **Windows 11** utilizando **WSL**. O código foi desenvolvido em **C++** utilizando a IDE **Visual Studio Code**.

7 Conclusão

Este trabalho apresentou uma análise detalhada do problema do vencedor do torneio, utilizando uma abordagem baseada em fluxo máximo em redes. Os principais resultados e contribuições incluem:

- A implementação eficiente de uma solução baseada no algoritmo Ford-Fulkerson, capaz de determinar rapidamente se uma equipe específica ainda pode vencer um torneio.
- Uma análise experimental abrangente dos fatores que influenciam a probabilidade de vitória, incluindo o número de equipes, o viés de vitórias e a fração de jogos realizados.
- A identificação de padrões e interações entre esses fatores, revelando como eles afetam conjuntamente a probabilidade de uma equipe poder vencer.

Os resultados demonstram a elegância e eficácia da abordagem de fluxo máximo para resolver este problema combinatório aparentemente complexo. A solução não apenas determina se uma equipe ainda pode vencer, mas também fornece insights valiosos sobre as condições que favorecem ou dificultam essa possibilidade.

Trabalhos futuros poderiam explorar extensões do problema, como determinar a probabilidade exata de vitória sob distribuições aleatórias de resultados, ou considerar regras mais complexas de pontuação (como empates ou pontos diferentes por tipo de vitória).

Referências

- [1] Schwartz, B. L. (1966). Possible winners in partially completed tournaments. *SIAM Review*, 8(3):302–308.
- [2] Ford, L. R. and Fulkerson, D. R. (1956). Maximal flow through a network. *Canadian Journal of Mathematics*, 8:399–404.
- [3] Wayne, K. (2013). The baseball elimination problem. *Princeton University, Computer Science Department*.
- [4] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2009). *Introduction to Algorithms*. MIT Press, 3rd edition.