Relatório 3º projecto ASA 2025/2026

Grupo: AL130

Aluno(s): André Sá (ist1109904) e Joana Melo (ist1114255)

Descrição do Problema e da Solução

O algoritmo executa filtros de otimização para casos triviais, onde, $i \in n$ equipas, c_i representando os pontos atuais de uma equipa e $max(X)$ e os pontos máximos possíveis da equipa alvo:

- Impossível de ganhar se $\exists\ i \in n : c_i > max(X)$
- Ganha sem vitórias se $\forall\ i : c_X > c_i$

O problema é resolvido na sua maioria no modelo linear sobre o conjunto de confrontos pendentes k_{ij}, entre as equipas (i,j) onde $R = \{(i,j): k_{ij} > 0\}$, $(i>j)$ $\forall$ $i,j \in n$.

- Identificação das variáveis do problema:
  - Para cada par de equipas $(i,j) \in R$:
    - w_{ij}: vitórias de i sobre j ($0 \leq w_{ij} \leq k_{ij}$)
    - w_{ji}: vitórias de j sobre i ($0 \leq w_{ji} \leq k_{ij}$)
    - d_{ij}: empates ($0 \leq d_{ij} \leq k_{ij}$)
  - Para cada equipa $i \in n$:
    - p_i: pontos obtidos por i nos jogos restantes
- Especificações do programa linear:
  - Função Objetivo: Minimizar($\sum_{(i,j)\in R: X=i} w_{ij} + \sum_{(i,j)\in R: X=j} w_{ji}$) : minimizar vitórias da equipa alvo X
  - Restrições do problema:
    - $w_{ij} + w_{ji} + d_{ij} = k_{ij}$: todos os jogos são disputados
    - $p_i = 3x(\sum_{(i,j)\in R: X=i} w_{ij} + \sum_{(i,j)\in R: X=j} w_{ji}) + \sum_{(i,j)\in R: X=i \vee X=i} d_{ij}$: cálculo de pontos para cada equipa
    - $c_i + p_X \geq c_i + p_i$, $i \neq X$: X termina em 1º lugar

Análise Teórica

- Pré-processamento(pasing e filtros) é O($n^2 + m$)
- Loop principal vai multiplicar por n: aplica o modelo a todas as equipas
  - Construção do modelo é O(n^3)
  - O número de variáveis do programa linear é $n^2 - m$
  - O número de restrições do programa linear é $n^2 - m$
  - Complexidade do programa linear exponencial(número de restrições + número de variáveis) é O($c^{n^2-m+n^2-m}$)=O(c^{n^2-m}), c-constante

Relatório 3º projecto ASA 2025/2026

Grupo: AL130

Aluno(s): André Sá (ist1109904) e Joana Melo (ist1114255)

T(n,m) = n · (O(n^3)+O(c^{n^2-m}))+ O($n^2 + m$) = O(n^4)+O($n \cdot c^{n^2-m}$)+O($n^2 + m$)=
= O($n \cdot c^{n^2-m}$)=O(c^{n^2-m}), no pior cenário onde m=0 a complexidade é O(c^{n^2})

Avaliação Experimental da Solução Proposta

Foram geradas 20 instâncias com n entre 2 e 47 (p=50%).

n	2	5	7	10	13	15	16	19	22	25	28	31	34	39	42	43	45	47
Tempo (s)	0.2252	0.1737	0.1869	0.3057	0.3868	0.5636	0.8357	0.8383	1.9091	2.1852	5.1984	11.4500	7.6912	22.2550	18.5279	20.3467	39.7388	62.2866

Os resultados confirmam empiricamente a complexidade O(2^(n²-m)), mostrando crescimento exponencial duplo com ~7× de aumento por unidade de n, tornando o algoritmo impraticável para n>50. Desvios residuais são atribuídos a fatores experimentais (sistema, cache, etc.) e não invalidam a tendência principal