

## ESCALA DE JOGOS DE TORNEIOS ESPORTIVOS: UMA ABORDAGEM VIA *SIMULATED ANNEALING*

### **Fabrcio Lacerda Biajoli**

Universidade Federal de Ouro Preto  
R. Artur Vitorino Coelho, 153 - Bauxita  
35.400-000 Ouro Preto, MG  
[fabriciolacerd@bol.com.br](mailto:fabriciolacerd@bol.com.br)

### **Otvio Massashi Mine**

Universidade Federal de Ouro Preto  
R. Artur Vitorino Coelho, 153 - Bauxita  
35.400-000 Ouro Preto, MG  
[massashi.mine@bol.com.br](mailto:massashi.mine@bol.com.br)

### **Antnio Augusto Chaves**

Universidade Federal de Ouro Preto  
R. Artur Vitorino Coelho, 153 - Bauxita  
35.400-000 Ouro Preto, MG  
[guto.chaves@bol.com.br](mailto:guto.chaves@bol.com.br)

### **Marcone Jamilson Freitas Souza**

Departamento de Computao  
Universidade Federal de Ouro Preto  
35.400-000 Ouro Preto, MG  
[marcone@iceb.ufop.br](mailto:marcone@iceb.ufop.br)

### **Resumo**

Esse trabalho aborda o problema de programao de jogos de torneios esportivos. Relata-se uma experincia com a utilizao da tcnica metaheurstica *Simulated Annealing* aplicada na montagem da tabela de jogos da primeira diviso do Campeonato Brasileiro de Futebol de 2002, tendo como objetivo a minimizao dos deslocamentos de todos os times durante o campeonato e a satisfao de um determinado conjunto de restries.

Palavras-chave: Programao de jogos de torneios esportivos, *Simulated Annealing*, Otimizao Combinatria.

### **Abstract**

This work deals with the sports timetabling problem. It relates an experience with the application of Simulated Annealing metaheuristic in the elaboration of the Brazilian Soccer Championship's schedule of 2002. The objective is to minimize the total distance traveled by the teams during the championship satisfying a given set of constraints.

Keywords: Sports Timetabling, Simulated Annealing, Combinatorial Optimization.

## 1. Introdução

Montar uma tabela de jogos de uma competição esportiva como o Campeonato Brasileiro de Futebol é uma tarefa muito complexa. É necessário atender a uma série de requisitos envolvendo vários aspectos como, por exemplo, a quantidade de vezes que um time pode jogar consecutivamente fora de sua sede, a minimização dos custos com o deslocamento dos times, o atendimento às redes de televisão etc. A dificuldade de solução desse problema é devida ao grande número de possibilidades a serem analisadas. De acordo com Concílio & Zuben (2002), para uma competição envolvendo  $n$  times, com  $n$  par, confrontando-se entre si em turnos completos (onde todos os times jogam em cada rodada), o número de combinações possíveis é dado por:

$$(n-1)!(n-3)!(n-5)! \dots (n-(n-1))! \times 2^{\frac{(n-1) \times n}{2}}$$

Para exemplificar a magnitude do espaço de soluções, para um torneio com 20 participantes há  $2,9062 \times 10^{130}$  combinações possíveis.

Em vista desse fato, este problema é normalmente abordado por técnicas heurísticas de solução. Concílio & Zuben (2002) abordam o problema da montagem da tabela de jogos do Campeonato Paulista de Futebol de 1997 utilizando programação genética. As restrições abordadas nesse trabalho são: (a) cada participante deve enfrentar todos os seus adversários uma única vez durante o torneio; (b) todos os participantes devem jogar a cada rodada; (c) a diferença do número de jogos de cada participante em seu domínio e fora dele é no máximo um; (d) o número máximo de jogos consecutivos no domínio do participante (ou fora) não é superior a um número  $r$ . Na função de aptidão de uma escala leva-se em consideração se todos os participantes estão percorrendo uma distância parecida ao longo das rodadas, de forma a não privilegiar este ou aquele participante em termos de custo e tempo de deslocamento. Para avaliar esta parcela da função de aptidão toma-se o inverso da diferença entre as distâncias máxima e mínima percorridas pelos participantes do torneio, considerando todas as rodadas. Costa (1995) desenvolve um método híbrido combinando as técnicas Algoritmos Genéticos e Busca Tabu para resolver um problema de escalonamento de jogos da Liga Americana de Hóquei. As principais restrições consideradas nesse trabalho são as seguintes: (a) cada time aponta 56 datas para poderem jogar em casa, sendo que apenas 41 dessas datas serão de fato escolhidas; (b) a distância total percorrida pelos times ao longo da competição deve ser minimizada; (c) um time não pode jogar mais do que dois dias e nem mais que três jogos em 5 dias consecutivos; (d) um time não pode ficar fora de seu domínio mais do que 14 dias seguidos. Nemhauser & Trick (1998) tratam o problema da alocação de jogos de um torneio de basquete envolvendo 9 times de diferentes localidades dos Estados Unidos utilizando uma combinação de uma técnica de programação inteira e uma técnica enumerativa. O procedimento proposto contém 3 fases. A primeira fase consiste em gerar um conjunto de padrões de cardinalidade igual ao número  $n$  de times. Cada padrão é uma sequência de  $n-1$  strings, sendo que cada string pode assumir um dentre dois valores, a saber, casa (C) ou fora (F). Na segunda fase os jogos são designados aos padrões, sem se estabelecer quais são os times envolvidos em cada jogo. Finalmente, na última fase, os times são designados aos padrões, produzindo uma escala.

O presente trabalho relata uma experiência com a utilização da técnica *Simulated Annealing* aplicada à resolução do problema. A estrutura de vizinhança considerada é diferente daquela proposta em Costa (1995) e os resultados obtidos comprovam a eficiência desta técnica na abordagem do problema. Esta técnica foi escolhida em função dos seguintes fatos, entre outros: (a) eficiência na abordagem de problemas altamente combinatórios como os de programação de horários (*timetabling*) (DOWSLAND, 1998); (b) não é dependente de solução inicial viável, a qual, para o problema abordado é muito difícil de se conseguir; (c) não é fortemente dependente de uma boa solução inicial; (d) facilidade de implementação. Comparações com outros trabalhos correlatos não foram possíveis em virtude de os problemas abordados terem requisitos diferentes. Acreditamos que a abordagem através dessa técnica produza soluções de boa qualidade em menor tempo do que as produzidas usando Algoritmos Genéticos (método usado em Concílio & Zuben (2002) e Costa(1995)), em vista desse último método requerer um maior esforço computacional, principalmente devido à necessidade de implementação de busca local e refinamentos para tratar de problemas de escalonamento (CONCÍLIO & ZUBEN, 2002).

Este trabalho está organizado como segue. Na seção 2 descreve-se o problema abordado. A

seção 3 apresenta a metodologia abordada, incluindo a forma de representação adotada e de geração de uma solução inicial, a estrutura de vizinhança, a função de avaliação e a adaptação da técnica heurística *Simulated Annealing* ao problema. A seção 4 apresenta detalhes de implementação e a seção 5 mostra os resultados encontrados. A seção 6 conclui o trabalho e aponta os trabalhos futuros.

## 2. Descrição do Problema Abordado

O problema abordado é o da montagem da tabela de jogos da primeira divisão do campeonato brasileiro de futebol de 2002. Neste problema há 26 times que devem se confrontar em 29 rodadas, satisfazendo a uma série de requisitos impostos pela Confederação Brasileira de Futebol (CBF), entre os quais:

- (a) Um time não pode jogar mais de uma vez na mesma rodada;
- (b) O campeonato deve ser realizado em turno único, isto é, cada time só pode jogar uma única vez contra outro time durante o campeonato, seja em sua sede ou fora dela;
- (c) Nas duas primeiras rodadas que cada time participar, um jogo deve ser realizado fora de sua sede e outro em sua sede. Exemplificando, se na primeira rodada o primeiro confronto de um time for fora de casa, então seu próximo confronto deve ser realizado em casa, ou vice versa;
- (d) As duas últimas rodadas que cada time participar devem ter a mesma configuração de seus dois primeiros confrontos. Exemplificando, se os dois primeiros confrontos foram jogar fora de casa e depois em casa, respectivamente, então os dois últimos confrontos devem ser, respectivamente, jogar fora de casa e depois em casa;
- (e) O número de jogos fora de casa deve ser igual ao número de jogos em casa. Quando o número de jogos for ímpar, o time com melhor posição no *ranking* da CBF joga uma partida a mais em casa;
- (f) Não pode haver jogos entre clubes do mesmo estado na última rodada;
- (g) Evitar que um time jogue mais de duas vezes consecutivas em casa;
- (h) Evitar que um time jogue mais de duas vezes consecutivas fora de casa.

Os seis primeiros requisitos ((a) ... (f)) são considerados essenciais, isto é, se eles não forem satisfeitos a tabela é inviável e não pode ser praticada na competição. Os dois últimos requisitos ((g) e (h)) são considerados não-essenciais, isto é, devem ser satisfeitos sempre que possível.

Além de procurar atender a esses requisitos, a CBF procura na montagem da tabela minimizar os gastos dos clubes com seus deslocamentos ao longo das rodadas. Nesse sentido, este trabalho considera, em caráter preliminar, a minimização da soma das distâncias percorridas pelos times durante o campeonato. Um objetivo mais realista, que se encontra em fase de levantamento, é considerar as despesas financeiras de cada time com cada deslocamento, bem como com as despesas diárias de manutenção fora de casa. Também, em caráter preliminar, tratou-se apenas os dois primeiros requisitos ((a) e (b)), bem como o último requisito listado anteriormente.

## 3. Metodologia

A metodologia considerada neste trabalho para tratar o problema baseia-se na metaheurística *Simulated Annealing*. Sendo assim, requer-se a definição da forma de representação de uma solução do problema (seção 3.1), da geração de uma solução inicial (seção 3.2), do estabelecimento de estrutura(s) de vizinhança (seção 3.3), de uma função de avaliação (seção 3.4), bem como da adaptação do método para abordar o problema (seção 3.5).

### 3.1. Representação

Uma tabela, ou solução  $s = (s_{ij})$  do problema, é representada por uma matriz com tantas linhas quantas forem as rodadas do torneio e com tantas colunas quantos forem os confrontos por rodada. Como o campeonato de 2002 contou com a participação de 26 times que se confrontaram em 29 rodadas, então uma solução  $s$  para este problema é uma matriz com 29 linhas e 13 colunas. Na realidade bastavam 25 rodadas para a realização de todos os jogos. No entanto, optou-se por

considerar 29 rodadas para efeito de comparação de resultados. Nesta matriz cada célula  $s_{ij}$  representa um possível jogo. Um jogo do tipo A X B (onde A tem mando de campo) ou B X A (onde B tem mando de campo) é representado unicamente por “(sinal) A X B”, sendo o mando de campo definido pelo sinal. Quando positivo, o mando de campo é do time A e, quando negativo, o mando é do time B. Ou seja, o jogo A X B é representado por (+A X B) e o jogo B X A, por (-A X B). A Figura 1 ilustra um fragmento de uma escala.

Rodadas/Jogo	1	...	13
1	+ A X B		- C X D
...	...		...
29	- C X B		+ A X E

Figura 1 – Representação de uma escala de jogos

Deve ser observado que, como há um máximo de 13 jogos a serem realizados em cada uma das 29 rodadas, tem-se 377 células  $s_{ij}$  disponíveis para representar os 325 jogos possíveis. Em outras palavras, nem todas as rodadas serão completas, isto é, haverá rodadas em que alguns times não participarão.

### 3.2. Geração de uma solução inicial

A solução inicial é obtida de forma aleatória, conforme a seguir se descreve. Inicialmente constrói-se uma lista L de todos os possíveis confrontos, com cada confronto envolvendo dois times distintos aparecendo uma única vez, e sendo o mando de campo definido por um sinal aleatoriamente escolhido. A seguir, para cada jogo de cada rodada escolhe-se aleatoriamente um confronto da lista L, alocando-o à rodada. A lista L é, então, atualizada para que os confrontos sorteados não sejam escolhidos novamente. Este procedimento prossegue enquanto a lista L for não vazia. Como parâmetros este procedimento necessita da lista L de confrontos, da matriz Tabela que representa uma solução e da variável rMax, que é o número máximo de rodadas a serem preenchidas. Para o problema abordado o rMax tem valor 29. A Figura 2 apresenta o pseudo-código do procedimento de geração de uma solução inicial.

```

Procedimento Constroi_Solucao_Inicial(L[.], Tabela[.][.], rMax)
1. r ← 1;           {Representa a rodada a ser preenchida}
2. j ← 1;           {Representa o jogo a ser preenchido}
3. enquanto ( L ≠ ∅ ) faça
4.   jogo ← confronto sorteado da lista L de confrontos;
5.   Tabela[r][j] ← jogo;
6.   Atualizar lista L;
7.   if ( r < rMax ) então r ← r + 1;
8.   senão
9.     j ← j + 1;
10.    r ← 1;
11.  fim-se
12. fim-enquanto
13. Retorne Tabela[.][.];
Fim Constroi_Solucao_Inicial;

```

Figura 2 – Procedimento de geração de uma solução inicial

É importante salientar que em virtude da forma de geração da lista de confrontos e da construção da solução inicial, o requisito (b) estabelecido na seção 2 é sempre verificado, isto é, cada time confronta cada um dos outros uma única vez.

### 3.3. Estrutura de Vizinhança

Foram definidos dois tipos de movimentos para definir a vizinhança  $N(s)$  de uma dada escala  $s$ . O primeiro caracteriza-se pela mudança do mando de campo, ou seja, pela troca de sinal do valor que representa o primeiro time de um dado jogo.

A Figura 3 ilustra este tipo de movimento. Como pode ser observado na escala  $s$ , no jogo 1 da primeira rodada o mando de campo é do time A, enquanto que na escala  $s'$  o mando de campo passou para o time B.

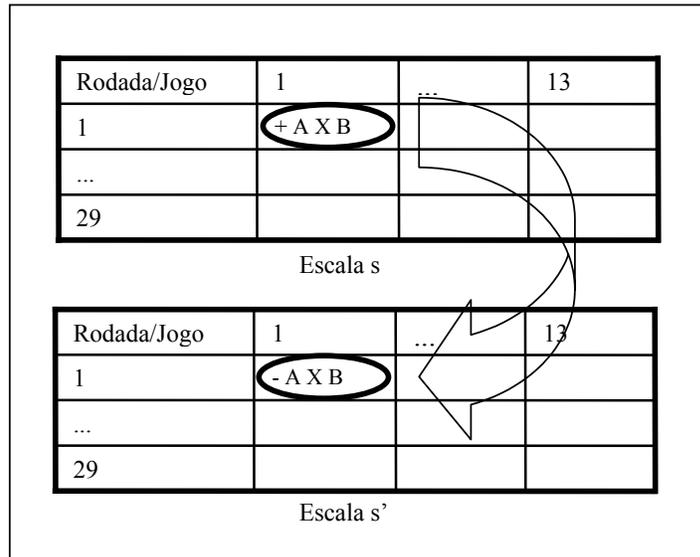


Figura 3 – Representação do movimento “mudança de mando de campo”

O segundo movimento é caracterizado pela troca de jogos entre as rodadas. A Figura 4 ilustra tal movimento. Como pode ser observado, o confronto entre os times A e B, realizado na escala  $s$  na primeira rodada, passa a ser realizado na escala  $s'$  na última rodada. Situação inversa ocorre com o confronto entre os times C e D.

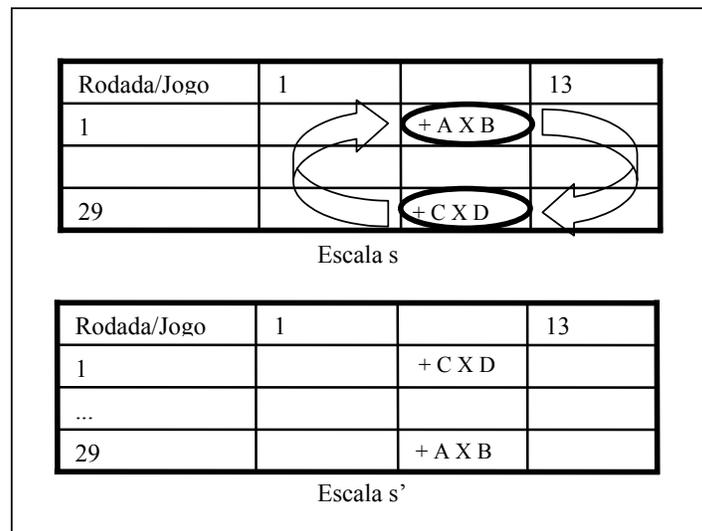


Figura 4 – Representação do movimento “troca de jogos entre rodadas”

Uma escala, ou solução  $s'$ , é dita vizinha de  $s$  se for obtida desta a partir de um movimento ou de mudança de mando de campo ou de troca de rodadas.

### 3.4. Função de Avaliação

Uma escala  $s$  é avaliada por uma função  $f : s \in S \rightarrow \mathfrak{R}$ , onde  $S$  é o conjunto de todas as possíveis escalas. Esta função, a qual deve ser minimizada, é baseada em penalidade e é composta por três partes distintas. A primeira trata da distância percorrida pelos times na seqüência de seus confrontos. As outras duas penalizam o não atendimento ao primeiro e último dos requisitos definidos na seção 2. A função de avaliação considerada, inspirada no trabalho de Costa (1995), é a seguinte:

$$f(s) = \sum_{(i,j) \in A} D_{ij} + \alpha_1 \times \left( \sum_{i=1}^{Ntimes} ExcessoRodadas_i \right) + \alpha_2 \times \left( \sum_{i=1}^{Ntimes} ConsecutivasFora_i \right)$$

Onde:

- $A$  = conjunto de todos os deslocamentos realizados pelos times na escala corrente  $s$ ;
- $D_{ij}$  = Distância entre as sedes dos times  $i$  e  $j$ ; (Ver Anexo B)
- $ExcessoRodadas_i$  = Número de vezes que o time  $i$  joga mais de uma vez na mesma rodada;
- $ConsecutivasFora_i$  = Número de vezes que o time  $i$  joga mais de duas vezes consecutivas fora de casa;
- $Ntimes$  = Número de times do torneio;
- $\alpha_1$  = Penalidade para  $ExcessoRodadas_i$ ;
- $\alpha_2$  = Penalidade para  $ConsecutivasFora_i$ ;

### 3.5. Simulated Annealing

*Simulated Annealing* é um método de busca local proposta por Kirkpatrick et al. (1983) que admite soluções de piora para tentar escapar de ótimos locais. Assim como os métodos tradicionais de busca local, esta técnica requer que seja definida uma estrutura de vizinhança e uma função de avaliação  $f$  que associa a cada solução um valor numérico representando o custo da solução. O processo inicia com um membro  $s$  qualquer do espaço de soluções, normalmente gerado aleatoriamente, e seleciona aleatoriamente um de seus vizinhos  $s'$ . Se esse vizinho  $s'$  for melhor que  $s$ , ele é aceito e passa a ser a nova solução corrente. Se o vizinho  $s'$  for pior que  $s$  por um valor  $\Delta$ , ele pode ser aceito com uma probabilidade  $e^{-\Delta/T}$ , onde  $T$  decresce gradualmente com o progresso do algoritmo. Esse procedimento é repetido até que  $T$  seja tão pequeno que não ocorra mais movimentos de melhora. A melhor solução encontrada durante a busca é considerada como uma aproximação para a solução ótima do problema. Esta técnica foi originalmente derivada de simulações em termodinâmica e, por esta razão, o parâmetro  $T$  é referido como a temperatura e a maneira como ela é reduzida é conhecida como esquema de resfriamento (DOWSLAND, 1993).

O esquema de resfriamento adotado neste trabalho foi o geométrico, isto é, a temperatura  $T$  é atualizada pela fórmula  $T \leftarrow \alpha \times T$ , onde  $\alpha$  representa a razão de resfriamento.

A Figura 5 apresenta o pseudocódigo do algoritmo *Simulated Annealing* usado para minimizar uma função  $f$ . Este algoritmo tem como parâmetros de entrada uma solução inicial  $s$  (obtida conforme seção 3.2), uma temperatura inicial  $T_0$ , uma estrutura de vizinhança  $N(\cdot)$ , um número máximo de iterações em uma dada temperatura ( $SAmx$ ) e uma razão de resfriamento  $\alpha$ .

```

procedimento SA( $f(\cdot), N(\cdot), \alpha, SAmax, T_0, s$ )
1  $s^* \leftarrow s;$       {Melhor solução obtida até então}
2  $IterT \leftarrow 0;$  {Número de iterações na temperatura T}
3  $T \leftarrow T_0;$    {Temperatura corrente}
4 enquanto ( $T > 0$ ) faça
5   enquanto ( $IterT < SAmax$ ) faça
6      $IterT \leftarrow IterT + 1;$ 
7     Escolha aleatoriamente um tipo de movimento; ( Seção 3.3)
8     Gere um vizinho qualquer  $s' \in N(s);$ 
9      $\Delta = f(s') - f(s);$ 
10    se ( $\Delta < 0$ )
11      então
12         $s \leftarrow s';$ 
13        se ( $f(s') < f(s^*)$ ) então  $s^* \leftarrow s';$ 
14      senão
15        Tome  $x \in [0,1];$ 
16        se ( $x < e^{-\Delta/T}$ ) então  $s \leftarrow s';$ 
17      fim-se;
18    fim-enquanto;
19     $T \leftarrow \alpha x T;$ 
20     $IterT \leftarrow 0;$ 
21  fim-enquanto;
22   $s \leftarrow s^*;$ 
23  Retorne  $s;$ 
fim SA;

```

Figura 5 – Algoritmo *Simulated Annealing*

#### 4. Implementação

Para desenvolvimento do sistema computacional foi utilizada a linguagem Delphi. Uma escala de jogos foi implementada como uma matriz composta por 29 linhas (representando as rodadas) e 13 colunas (representando o número máximo de confrontos por rodada) de células do tipo Tjogo. Tjogo é um registro composto por dois campos inteiros t1 e t2 representando os times envolvidos no confronto:

```

TJogo = record
    t1,
    t2 : integer;
end;
Escala: array[1..29, 1..13] of TJogo;

```

Para quantificar o não atendimento aos dois requisitos tratados, bem como as distâncias percorridas por cada time foi criada uma Matriz de Controle, composta por  $27 \times 31 = 837$  células de dois campos: casa e fora. Em cada uma das 26 primeiras linhas dessa matriz, que correspondem aos 26 times que participam do campeonato, são armazenados no campo casa (respectivamente, fora de casa) o número de vezes que cada time joga em casa (respectivamente, fora de casa) em uma das 29 rodadas do campeonato. A 30ª coluna armazena a distância percorrida por cada time. Na 31ª coluna armazena-se o número de vezes que um time joga mais de duas vezes fora de casa (último dos requisitos listados na seção 2). A 27ª linha contabiliza o número de vezes que os times jogam mais de uma vez em uma mesma rodada (primeiro dos requisitos listados na seção 2). Uma ilustração dessa matriz é dada a seguir:

Times/Rodadas	1		...		31	
1	Casa	Fora				
...						
27						

Figura 6 – Representação dos jogos em casa e fora de casa para cada time

## 5. Resultados

O algoritmo desenvolvido foi implementado na linguagem Delphi usando o compilador Borland Delphi 5.0 e testado em um microcomputador PC AMD Athlon, 1,5 GHz, com 256 MB de RAM sob sistema operacional Windows ME.

Inicialmente, a metaheurística *Simulated Annealing* passou por uma bateria preliminar de testes visando à calibragem de seus parâmetros. Foram os seguintes os parâmetros e pesos utilizados, conforme notação adotada na seção 3.4:  $S_{Amax} = 3300$  (número de iterações em uma dada temperatura),  $T_0 = 1600$  (temperatura inicial),  $\alpha = 0.95$  (taxa de resfriamento),  $\alpha_1 = 10^8$  e  $\alpha_2 = 10^7$ .

Foram realizados 12 testes, cada qual partindo de uma semente diferente de números aleatórios. As características de cada um desses testes encontram-se relatadas na Tabela 1.

Teste	Valor inicial de $f(s)$	Distância	ExcessoRodadas	ConsecutivasFora
1	17320504681	504681	166	72
2	18120495672	495672	175	72
3	5020555525	555525	47	32
4	7630502826	502826	72	43
5	11420527890	507890	108	62
6	19410491585	491585	487	71
7	19850523020	523020	192	65
8	18860466840	513258	170	73
9	17730513258	513258	170	73
10	4930547280	547380	46	33
11	5340525752	525752	50	34
12	19670508842	508842	190	67

Tabela 1 – Dados relativos à solução inicial aleatória

Após a aplicação do algoritmo proposto, foram obtidas as soluções finais com as características expostas na Tabela 2.

Teste	Valor final de $f(s)$	Distância	ExcessoRodadas	ConsecutivasFora
1	350305	350305	0	0
2	303143	303143	0	0
3	353821	353821	0	0
4	343045	343045	0	0
5	359267	359267	0	0
6	340919	340919	0	0
7	335612	335612	0	0
8	351493	351493	0	0
9	331169	331169	0	0
10	338710	338710	0	0
11	331217	331217	0	0
12	349702	349702	0	0

Tabela 2 – Dados das soluções finais obtidas pelo algoritmo proposto

O valor médio obtido foi de 340700,3 Km, com a melhor solução de valor 303143 Km e a pior de 359267 Km. O desvio em relação ao melhor valor encontrado pelo algoritmo, isto é, desvio =  $((\text{Valor Médio} - \text{Melhor Valor}) \times 100) / (\text{Melhor Valor})$  foi de 12,39%.

A Tabela 3 relaciona os resultados obtidos com os valores apresentados pela tabela oficial do Campeonato Brasileiro de Futebol de 2002.

Tabela Oficial do ano 2002			Resultados obtidos pelo algoritmo proposto				
$f(s)$	Distância	ConsecutivasFora	Solução	$f(s)$	Distância	ConsecutivasFora	Melhora
80.613.556	613.556	8	Melhor	303.143	303.143	0	50,56%
			Pior	359.267	359.267	0	41,45%

Tabela 3 – Comparação entre os resultados oficiais e os obtidos

O tempo médio de resolução do problema foi de aproximadamente 8,5 minutos. A tabela referente à melhor solução obtida é apresentada no Anexo A.

## 6. Conclusão e Trabalhos Futuros

Este trabalho contribui com a apresentação de uma modelagem heurística, baseada em *Simulated Annealing*, para tratar a montagem de uma tabela de jogos entre times participantes de uma competição esportiva realizada em vários locais e ao longo de várias rodadas. Foi desenvolvido um sistema computacional para tratar um conjunto reduzido de requisitos relativos à montagem da escala de jogos do campeonato brasileiro de 2002. Os resultados apresentados demonstram claramente a eficiência do método na abordagem do problema, tanto no atendimento aos requisitos exigidos, quanto na diminuição dos custos envolvidos. Observa-se que, no pior caso, houve uma melhora de até 41,45% em relação à distância total percorrida pelos times na tabela oficial.

É importante ressaltar, entretanto, que apesar de a metodologia proposta resultar em diminuição de custos com deslocamento, ela não é de implementação fácil devido sobretudo a fatores políticos. Por exemplo, muitos dirigentes podem achar que seus times foram prejudicados em relação a outros pela sequência de jogos da tabela implementada. É também oportuno dizer que a abordagem proposta não leva em consideração a participação dos times em outras competições. Sendo assim, otimiza-se tão somente o deslocamento dos times na competição.

Atualmente está-se implementando os requisitos relacionados na seção 2 e não considerados nesta fase preliminar, bem como outros relativos às exigências das redes de televisão, a fim de torná-lo mais próximo da realidade praticada pela CBF.

## Agradecimentos

Os autores agradecem à Confederação Brasileira de Futebol (CBF), particularmente ao Dr. Tânus Jorge Nagem, pela disponibilização das informações necessárias à execução deste trabalho.

## Referências

CONCÍLIO, R. & ZUBEN, F. J. (2002) – Uma Abordagem Evolutiva para Geração Automática de Turnos Completos em Torneios. *Revista Controle & Automação*, v. 13, n. 2, p. 105-122.

COSTA, D. (1995) – An Evolutionary Tabu Search Algorithm and the NHL Scheduling Problem. *INFORMS*, v. 33, n. 3, p. 161-178.

DOWSLAND, K. A. (1993) - Simulated Annealing. In C.R. Reeves, editor, *Modern Heuristic Techniques for Combinatorial Problems*, Advanced Topics in Computer Science Series, chapter 2, p. 20-69. Blackwell Scientific Publications, London.

DOWSLAND, K. A. (1998) – Off-the-Peg or Made-to-Measure? Timetabling and Scheduling with SA and TS. In Practice and Theory of Automated Timeabling II (Burke, E. And Carter, M., eds), *Lecture Notes in Computer Science*, v. 1408, p. 37-52.

KIRKPATRICK, S.; GELLAT, D. C. & VECCHI, M. P. (1983) - Optimization by Simulated Annealing. *Science*, v. 220, p. 671-680.

NEMHAUSER, G. L. & TRICK, M. A. (1998) – Scheduling a Major College Basketball Conference. *Operations Research*, v. 46, n. 1, p. 1-8.

## ANEXO A

## Tabela da Melhor Solução Obtida pelo Trabalho Proposto

Distância Total Percorrida: 303143 km;

(Exemplo de representação de um jogo na tabela abaixo: **21 X 5** ⇔ **Atlético-PR X Corinthians**, sendo o mando de campo do Atlético-PR)

Rodada/Jogo	1	2	3	4	5	6	7	8	9	10	11	12	13	Times	Nº
<b>Rodada 1</b>	21 X 5	6 X 26	20 X 3	22 X 19	10 X 13	12 X 9	7 X 18	25 X 2	8 X 11	24 X 23	1 X 17	-	-	Atlético-MG	<b>1</b>
<b>Rodada 2</b>	22 X 5	8 X 2	3 X 14	16 X 7	9 X 6	17 X 20	26 X 12	19 X 15	18 X 21	23 X 25	4 X 10	-	-	Cruzeiro	<b>2</b>
<b>Rodada 3</b>	25 X 9	15 X 20	5 X 26	10 X 6	1 X 24	16 X 21	11 X 12	13 X 14	8 X 17	22 X 18	-	-	-	São Paulo	<b>3</b>
<b>Rodada 4</b>	5 X 1	2 X 10	16 X 3	19 X 8	21 X 7	24 X 4	12 X 17	15 X 13	20 X 18	9 X 22	26 X 11	-	-	São Caetano	<b>4</b>
<b>Rodada 5</b>	14 X 23	2 X 11	17 X 19	6 X 1	7 X 25	20 X 8	10 X 3	13 X 9	22 X 15	18 X 24	-	-	-	Corinthians	<b>5</b>
<b>Rodada 6</b>	3 X 22	12 X 2	14 X 10	4 X 25	6 X 13	23 X 7	9 X 17	19 X 26	20 X 16	11 X 5	8 X 24	15 X 18	-	Palmeiras	<b>6</b>
<b>Rodada 7</b>	1 X 8	2 X 13	3 X 19	10 X 15	14 X 6	24 X 7	17 X 23	22 X 11	16 X 9	12 X 21	4 X 20	25 X 5	-	Portuguesa	<b>7</b>
<b>Rodada 8</b>	9 X 1	6 X 22	13 X 3	14 X 16	7 X 26	8 X 21	11 X 17	25 X 12	19 X 20	15 X 24	4 X 2	-	-	Figueirense	<b>8</b>
<b>Rodada 9</b>	1 X 10	23 X 2	21 X 3	5 X 7	16 X 6	9 X 24	12 X 19	18 X 11	20 X 13	26 X 15	22 X 4	-	-	Botafogo	<b>9</b>
<b>Rodada 10</b>	21 X 20	13 X 18	5 X 8	6 X 23	3 X 1	15 X 12	19 X 25	10 X 26	17 X 16	11 X 14	-	-	-	Flamengo	<b>10</b>
<b>Rodada 11</b>	1 X 12	2 X 15	23 X 3	5 X 9	8 X 10	24 X 25	20 X 11	4 X 13	16 X 18	26 X 21	14 X 7	-	-	Fluminense	<b>11</b>
<b>Rodada 12</b>	21 X 1	17 X 6	12 X 22	18 X 5	7 X 11	9 X 23	8 X 3	16 X 19	25 X 20	14 X 4	13 X 24	-	-	Vasco	<b>12</b>
<b>Rodada 13</b>	14 X 26	7 X 9	3 X 25	1 X 18	6 X 20	8 X 12	4 X 17	11 X 16	22 X 23	15 X 5	13 X 19	-	-	Juventude	<b>13</b>
<b>Rodada 14</b>	15 X 1	7 X 13	26 X 3	5 X 12	17 X 24	18 X 2	4 X 11	19 X 6	10 X 21	9 X 8	20 X 14	-	-	Bahia	<b>14</b>
<b>Rodada 15</b>	6 X 2	19 X 7	18 X 17	23 X 4	14 X 21	9 X 26	24 X 10	22 X 16	11 X 1	25 X 15	13 X 5	-	-	Vitória	<b>15</b>
<b>Rodada 16</b>	21 X 13	2 X 22	5 X 14	17 X 7	8 X 15	4 X 26	10 X 19	12 X 3	16 X 23	24 X 11	1 X 25	-	-	Grêmio	<b>16</b>
<b>Rodada 17</b>	22 X 13	2 X 14	7 X 4	19 X 11	6 X 24	1 X 16	12 X 10	26 X 23	17 X 25	15 X 21	3 X 18	-	-	Internacional	<b>17</b>
<b>Rodada 18</b>	19 X 1	24 X 2	8 X 4	16 X 5	11 X 6	3 X 7	25 X 21	13 X 12	14 X 17	20 X 26	23 X 18	10 X 22	-	Ponte Preta	<b>18</b>
<b>Rodada 19</b>	1 X 20	19 X 24	4 X 9	17 X 5	7 X 2	18 X 8	25 X 13	12 X 14	15 X 23	16 X 26	21 X 22	6 X 3	-	Guarani	<b>19</b>
<b>Rodada 20</b>	3 X 11	6 X 15	25 X 16	5 X 10	8 X 7	2 X 17	23 X 13	9 X 21	14 X 19	26 X 24	12 X 4	22 X 20	-	Gama	<b>20</b>
<b>Rodada 21</b>	1 X 22	24 X 16	6 X 25	19 X 5	13 X 8	2 X 26	18 X 12	7 X 10	17 X 3	21 X 4	20 X 23	9 X 14	-	Atlético-PR	<b>21</b>
<b>Rodada 22</b>	25 X 18	5 X 3	24 X 20	16 X 2	8 X 26	10 X 17	11 X 13	19 X 9	4 X 6	23 X 21	14 X 15	-	-	Coritiba	<b>22</b>
<b>Rodada 23</b>	20 X 7	6 X 12	26 X 13	23 X 1	4 X 19	3 X 2	18 X 10	8 X 16	14 X 22	17 X 21	25 X 11	15 X 9	-	Paraná	<b>23</b>
<b>Rodada 24</b>	4 X 5	1 X 26	12 X 7	8 X 23	21 X 24	9 X 20	2 X 19	22 X 17	15 X 3	11 X 10	18 X 14	-	-	Santos	<b>24</b>
<b>Rodada 25</b>	11 X 21	4 X 15	5 X 23	7 X 22	26 X 25	10 X 20	14 X 24	9 X 18	16 X 12	8 X 6	1 X 2	-	-	Goiás	<b>25</b>
<b>Rodada 26</b>	1 X 7	9 X 3	4 X 16	24 X 5	20 X 2	6 X 21	18 X 26	25 X 14	23 X 10	22 X 8	11 X 15	-	-	Paysandu	<b>26</b>
<b>Rodada 27</b>	23 X 11	19 X 18	8 X 25	5 X 6	15 X 7	13 X 17	3 X 4	1 X 14	16 X 10	24 X 22	20 X 12	2 X 9	-		
<b>Rodada 28</b>	2 X 5	1 X 13	18 X 4	3 X 24	10 X 9	7 X 6	12 X 23	15 X 16	17 X 26	22 X 25	21 X 19	14 X 8	-		
<b>Rodada 29</b>	19 X 23	4 X 1	5 X 20	9 X 11	13 X 16	12 X 24	15 X 17	25 X 10	26 X 22	6 X 18	21 X 2	-	-		

## ANEXO B

## Tabela das Distâncias Entre as Sedes dos Clubes

(A matriz abaixo é simétrica, ou seja,  $d_{ij} = d_{ji}$ )

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
1	-	0	586	586	586	586	586	1301	434	434	434	434	1585	1372	1372	1712	1712	740	740	716	1004	1004	1004	658	906	2824
2	0	-	586	586	586	586	586	1301	434	434	434	434	1585	1372	1372	1712	1712	740	740	716	1004	1004	1004	658	906	2824
3	586	586	-	0	0	0	0	705	429	429	429	429	982	1962	1962	1119	1119	99	99	1015	408	408	408	72	926	2933
4	586	586	0	-	0	0	0	705	429	429	429	429	982	1962	1962	1119	1119	99	99	1015	408	408	408	72	926	2933
5	586	586	0	0	-	0	0	705	429	429	429	429	982	1962	1962	1119	1119	99	99	1015	408	408	408	72	926	2933
6	586	586	0	0	0	-	0	705	429	429	429	429	982	1962	1962	1119	1119	99	99	1015	408	408	408	72	926	2933
7	586	586	0	0	0	0	-	705	429	429	429	429	982	1962	1962	1119	1119	99	99	1015	408	408	408	72	926	2933
8	1301	1301	705	705	705	705	705	-	1144	1144	1144	1144	478	2682	2682	476	476	773	773	1673	300	300	300	777	1493	3500
9	434	434	429	429	429	429	429	1144	-	0	0	0	1426	1649	1649	1553	1553	511	511	1148	852	852	852	501	1338	3250
10	434	434	429	429	429	429	429	1144	0	-	0	0	1426	1649	1649	1553	1553	511	511	1148	852	852	852	501	1338	3250
11	434	434	429	429	429	429	429	1144	0	0	-	0	1426	1649	1649	1553	1553	511	511	1148	852	852	852	501	1338	3250
12	434	434	429	429	429	429	429	1144	0	0	0	-	1426	1649	1649	1553	1553	511	511	1148	852	852	852	501	1338	3250
13	1585	1585	982	982	982	982	982	478	1426	1426	1426	1426	-	2963	2963	131	131	1057	1057	1900	584	584	584	1054	1720	3727
14	1372	1372	1962	1962	1962	1962	1962	2682	1649	1649	1649	1649	2963	-	0	3090	3090	1982	1982	1446	2385	2385	2385	2034	1643	2100
15	1372	1372	1962	1962	1962	1962	1962	2682	1649	1649	1649	1649	2963	0	-	3090	3090	1982	1982	1446	2385	2385	2385	2034	1643	2100
16	1712	1712	1119	1119	1119	1119	1119	476	1553	1553	1553	1553	131	3090	3090	-	0	1177	1177	2027	711	711	711	1181	1847	3854
17	1712	1712	1119	1119	1119	1119	1119	476	1553	1553	1553	1553	131	3090	3090	0	-	1177	1177	2027	711	711	711	1181	1847	3854
18	740	740	99	99	99	99	99	773	511	511	511	511	1057	1982	1982	1177	1177	-	0	921	476	476	476	171	835	2842
19	740	740	99	99	99	99	99	773	511	511	511	511	1057	1982	1982	1177	1177	0	-	921	476	476	476	171	835	2842
20	716	716	1015	1015	1015	1015	1015	1673	1148	1148	1148	1148	1900	1446	1446	2027	2027	921	921	-	1366	1366	1366	1087	209	2140
21	1004	1004	408	408	408	408	408	300	852	852	852	852	584	2385	2385	711	711	476	476	1366	-	0	0	481	1186	3193
22	1004	1004	408	408	408	408	408	300	852	852	852	852	584	2385	2385	711	711	476	476	1366	0	-	0	481	1186	3193
23	1004	1004	408	408	408	408	408	300	852	852	852	852	584	2385	2385	711	711	476	476	1366	0	0	-	481	1186	3193
24	658	658	72	72	72	72	72	777	501	501	501	501	1054	2034	2034	1181	1181	171	171	1087	481	481	481	-	998	3035
25	906	906	926	926	926	926	926	1493	1338	1338	1338	1338	1720	1643	1643	1874	1874	835	835	209	1186	1186	1186	998	-	2017
26	2824	2824	2933	2933	2933	2933	2933	3500	3250	3250	3250	3250	3727	2100	2100	3854	3854	2842	2842	2140	3193	3193	3193	3035	2017	-

(Obs: As distâncias da tabela acima estão em km de rodovia. A numeração dos times segue a mesma legenda do ANEXO A)