

DESIGNAÇÃO DE CORREDORES EM PERCURSOS DE UMA CORRIDA DE REVEZAMENTO

Omero F. Bertol^{1,2}, Eden R. Dosciatti^{1,2}, Percy Nohama¹, Leandro Magatão¹

¹ UTFPR - Universidade Tecnológica Federal do Paraná, Sede Curitiba.
CPGEI - Pós-Graduação em Engenharia Elétrica e Informática Industrial.
Av. Sete de Setembro 3165, 80230-901, Curitiba, PR, Brasil.

² UTFPR - Universidade Tecnológica Federal do Paraná, Câmpus Pato Branco.
GETIC - Grupo de Estudos e Pesquisa em Tecnologias de Informação e Comunicação.
Via do Conhecimento, Km 01, Caixa Postal 571, 85501-970, Pato Branco, PR, Brasil.

{omero, edenrd, nohama, magatao}@utfpr.edu.br

Resumo. *A Pesquisa Operacional é um ramo interdisciplinar da matemática aplicada que objetiva fornecer ferramentas quantitativas como forma de apoiar o processo de tomada de decisões. A Programação Linear Inteira Mista, no campo da Programação Matemática, é uma subárea da Pesquisa Operacional que abrange a análise de sistemas complexos do mundo real, tipicamente na busca de uma alocação ótima dos recursos envolvidos para melhorar o desempenho para atingir um objetivo. Este trabalho apresenta a implementação da solução de um problema de Programação Linear Inteira Mista, que tem o objetivo de designar corredores em percursos de uma corrida de revezamento, usando o ambiente de modelagem GUSEK.*

1. Introdução

A Pesquisa Operacional (PO) é um ramo interdisciplinar da matemática aplicada que faz uso de modelos matemáticos, estatísticos e de algoritmos, fornecendo ferramentas quantitativas como forma de apoiar o processo de tomada de decisões. É usada, sobretudo, para analisar sistemas complexos do mundo real, tipicamente com o objetivo de melhorar ou aperfeiçoar o desempenho [Magatão 2013].

Na PO, a subárea denominada Programação Matemática, emprega “símbolos matemáticos” na construção de modelos para, a partir da idealização da realidade, representar as variáveis do sistema real [Puccini e Pizzolato 1990].

Os modelos de Programação Matemática podem ser entendidos como um conjunto de equações, inequações e dependências lógicas que correspondem a relacionamentos apresentados por estruturas reais. Em um modelo matemático, são incluídos três conjuntos principais de elementos [Magatão 2013]:

- (1) Variáveis de decisão e parâmetros: as variáveis de decisão são as incógnitas a serem determinadas pela solução do modelo e os parâmetros são valores fixos do problema (dados de entrada);
- (2) Restrições: de modo a levar em conta as limitações físicas do sistema, o modelo deve incluir restrições que limitam os valores possíveis (ou viáveis) das variáveis;
- (3) Função objetivo: é uma função matemática que define a qualidade da solução em função das variáveis de decisão empregadas. Engloba considerações de “maximização” de lucros ou “minimização” de custos, por exemplo.

2. Problema

A Corrida de Revezamento das Nascentes do Iguaçu, organizada pela Secretaria do Esporte, Laser e Juventude da Prefeitura de Curitiba/PR, tem por objetivo, além de incentivar a prática esportiva e a integração metropolitana, divulgar as peculiaridades turísticas, naturais e ambientais de nove municípios da Região Metropolitana de Curitiba, cortados pelos afluentes do Rio Iguaçu [Smelj 2013].

O Revezamento das Nascentes trata-se de uma corrida de revezamento a pé, com distância de 108 km, divididos em 12 percursos que deverão ser realizados por uma equipe com no máximo 12 corredores. Em 2013 a prova, em sua 9ª edição, realizou-se no dia 25 de agosto com largada no município de Piraquara e a chegada na cidade de Curitiba, no bosque São Cristóvão, em Santa Felicidade [Smelj 2013].

3. Metodologia

No problema para designação de corredores em percursos de uma corrida de revezamento, as variáveis de decisão devem indicar se o corredor vai realizar ou não um determinado percurso (variáveis binárias). Os parâmetros fornecidos são o número de percursos na corrida, o número de corredores por equipe e os tempos médios de cada corredor em cada um dos percursos. A primeira restrição deve garantir que cada corredor deverá realizar uma quantidade de percursos proporcional ao número de corredores da equipe, por exemplo, se 12 corredores compõem a equipe, então haverá 1 percurso por corredor; se 6 corredores compõem a equipe, então haverá 2 percursos por corredor; se 4 corredores compõem a equipe, então haverá 3 percursos por corredor, ou seja, a quantidade de percursos por corredor será definida pelo resultado da expressão “número de percursos dividido pelo número de corredores”. A segunda restrição trabalha com a limitação de que cada percurso deverá ser realizado por um e somente um corredor. A função objetivo será o somatório dos tempos dos corredores em seus respectivos percursos determinando o “menor tempo de conclusão da prova”.

Na elaboração do modelo para abordar este cenário foi necessário levantar o desempenho dos componentes da equipe do revezamento em cada um dos percursos da corrida. Para realizar o levantamento, 12 corredores foram pré-selecionados e cada corredor respondeu um formulário *web* [Dados 2013]. No formulário, cada percurso era apresentado com a distância e a altimetria do terreno e o corredor, a partir da análise destes dados, deveria indicar o tempo médio previsto para conclusão do respectivo percurso.

Na fase de implementação da solução os elementos do modelo devem ser convertidos em regras operacionais através de um *software* de computador conhecido como *solver*. Um *solver* é uma ferramenta computacional que permite resolver equações com diversas variáveis desconhecidas. O objetivo do processo realizado no *solver* é localizar os valores de variáveis de uma equação que resultam em um valor otimizado para a função objetivo [Magatão 2013].

Atualmente, várias empresas disponibilizam *solvers* com métodos de solução de modelos de Programação Linear Inteira Mista. Como alternativas de versões comerciais pode-se citar o *IBM-CPLEX* da IBM e o *Solver Excel* da Microsoft. Como opções de *software* livre tem-se o *LP_Solve* da LGPL (*Lesser General Public License*) e o *GLPK* (*GNU Linear Programming*) [Magatão 2013].

4. Resultados

O ambiente de modelagem GUSEK foi utilizado na implementação do modelo matemático desenvolvido para designar corredores em percursos de uma corrida de revezamento. O GUSEK é uma interface de desenvolvimento para modelos de Programação Linear e Programação Linear Inteira Mista [Gusek 2013]. O pacote consiste em uma versão customizada do editor de textos SciTE (*SCIntilla Text Editor*) integrada a uma versão pré-compilada do *solver* GLPK (*GNU Linear Programming Kit*) para a plataforma Win32 [Scite 2013], [Glpk 2013].

Os principais comandos usados no GUSEK para implementar a solução do modelo matemático proposto foram: a) *var*: declaração das variáveis de decisão; b) *param*: declaração de parâmetros de entrada; c) *set*: declaração dos intervalos dos índices; d) *subject to*: declaração das restrições; e) *minimize*: minimizar o valor da função objetivo; f) *solve*: executar o modelo; g) *for*: estrutura de repetição; h) *if*: estrutura de decisão; i) *printf*: apresentar resultados no dispositivo de saída; e, j) *end*: finalizar o modelo.

O código fonte do modelo matemático implementado para resolver o problema da designação de corredores em percursos em uma corrida de revezamento, pode ser observado no Quadro 1.

Quadro 1. Implementação do modelo matemático proposto no GUSEK.

```
param nc 'número de corredores', integer;
param np 'número de percursos', integer;
set I 'índice corredores' := 1..nc;
set J 'índice percursos' := 1..np;
# Indica se o corredor vai realizar o percurso ou não
# cp[i,j] = 0; i-ésimo corredor "não" vai realizar o j-ésimo percurso
# cp[i,j] = 1; i-ésimo corredor "sim", vai realizar o j-ésimo percurso
var cp 'corredor versus percurso' {i in I, j in J} binary;

# Tempo dos Corredores versus Percursos:
# tp[i,j] - tempo do i-ésimo corredor no j-ésimo percurso
param tp 'tempo corredor versus percurso' {i in I, j in J};

# Função Objetivo: minimizando a soma dos tempos (corredores versus percursos)
minimize z: sum {i in I, j in J} ( tp[i,j] * cp[i,j] );

# Cada Corredor realiza (nro de percursos/nro de corredores) Percursos
subject to CorredorXPercurso {i in I}: sum {j in J} cp[i,j] = (np div nc);

# Cada Percurso é realizado por um e somente um Corredor
subject to PercursoXCorredor {j in J}: sum {i in I} cp[i,j] = 1;

solve;

printf "\n";
printf "
          P E R C U R S O S
Cor.  .1.. .2.. .3.. .4.. .5.. .6.. .7.. .8.. .9.. .10. .11. .12.\n";
printf "-----\n";
for {i in I} {
  printf "%3d:", i;
  for {j in J} {
    printf " %4d", tp[i,j];
  }
  printf("\n");
}
printf "-----";
```

Quadro 1 (cont.). Implementação do modelo matemático proposto no GUSEK.

```

printf "\n\n";
printf "
          P E R C U R S O S
Cor.  .1.. .2.. .3.. .4.. .5.. .6.. .7.. .8.. .9.. .10. .11. .12.\n";
printf "-----\n";
for {i in I} {
  printf "%3d:", i;
  for {j in J} {
    printf " %4s", if (cp[i,j] = 1) then tp[i,j] else "....";
  }
  printf "\n";
}
printf "-----\n";

printf "\nTempo de Conclusão da Prova:\n";

# converte o tempo total (z) em horas (h), minutos (m) e segundos (s)
printf "%2.0f horas\n", (z div 3600);
printf "%2.0f minutos\n", (z mod 3600) div 60;
printf "%2.0f segundos\n\n", (z mod 3600) mod 60;

end;

```

A execução da solução implementada, apresentada na Figura 1, mostra os resultados divididos em três partes: **a)** parâmetros de entrada retirados das respostas ao formulário *web* [Dados 2013], os tempos dos corredores *versus* percursos foram informados em segundos; **b)** tempo do corredor no seu respectivo percurso; e, **c)** menor tempo de conclusão da prova calculado em horas, minutos e segundos.

INTEGER OPTIMAL SOLUTION FOUND												
Time used: 0.0 secs												
Memory used: 0.3 Mb (294771 bytes)												
[A]	P E R C U R S O S											
Cor.	.1..	.2..	.3..	.4..	.5..	.6..	.7..	.8..	.9..	.10.	.11.	.12.
1:	2435	3375	2700	4200	2700	3000	2640	3120	4971	3000	1980	2340
2:	2580	3720	3300	4620	3180	3900	3360	3900	6300	3900	2400	3000
3:	2400	3600	3000	4200	3000	3120	3000	3600	5400	3600	2100	2100
4:	2400	3000	3000	3660	2700	2700	2520	2880	4440	2880	1800	2100
5:	2400	3600	4200	3900	2700	3000	3000	3120	5280	3300	2340	2520
6:	1920	2700	2460	4140	3000	2565	2469	2580	4453	2663	1623	1865
7:	2280	2460	2400	3840	2040	2340	2040	2340	3960	2640	1560	1800
8:	2160	3120	2640	3900	2700	3000	2580	3000	4320	3000	1920	2100
9:	2280	3360	2700	4500	2700	2760	2640	3360	5400	3420	2100	2220
10:	1945	2980	2065	3660	2215	2670	2050	2650	4440	2540	1700	1940
11:	1502	2196	1792	2714	1740	1924	1740	2100	3384	2200	1320	1440
12:	1700	2400	2000	3100	2150	2150	2050	2350	3700	2500	1550	1650
[B]	P E R C U R S O S											
Cor.	.1..	.2..	.3..	.4..	.5..	.6..	.7..	.8..	.9..	.10.	.11.	.12.
1:	3000
2:	2580
3:	2100
4:	1800
5:	2700
6:	2700
7:	2340
8:	2580
9:	2760
10:	2065
11:	2714
12:	3700
Tempo de Conclusão da Prova:												
8 horas												
37 minutos [C]												
19 segundos												
Model has been successfully processed												

Figura 1- Resultado da execução do modelo matemático proposto implementado no GUSEK.

5. Discussão e Conclusões

O presente trabalho destaca a importância da Pesquisa Operacional (PO) como ramo interdisciplinar da matemática aplicada que faz uso de modelos matemáticos e algoritmos fornecendo ferramentas quantitativas como forma de apoiar o processo de tomada de decisões.

Particular atenção foi dada à Programação Linear Inteira Mista (PLIM), uma das técnicas de sucesso da PO, que utiliza um conjunto de equações/inequações para representar relacionamentos de estruturas reais. Em um modelo matemático de PLIM, são incluídos três conjuntos principais de elementos: (i) as variáveis de decisão e parâmetros; (ii) um conjunto de restrições; e, (iii) uma função objetivo, ou função de avaliação do sistema.

Um problema para designação de corredores em percursos de uma corrida de revezamento foi apresentado, bem como a sua modelagem em PLIM. Adicionalmente, destacou-se o uso do GUSEK, um *software* livre destinado à modelagem e resolução de modelos de PLIM. Assim, foi apresentada a implementação no ambiente GUSEK do estudo de caso proposto (Quadro 1), evidenciando-se a forma ótima de alocação dos corredores, conforme Figura 1.

Referências

- Dados (2013) “Formulário Web Usado no Levantamento dos Tempos dos Corredores nos Percursos da IX Corrida de Revezamento das Nascentes do Iguaçu”, https://docs.google.com/forms/d/1Rm5ZVuzgHV79LHS5xT_F9zqghV8v_zD4JzqRlpha9fM/viewform, Agosto.
- Glpk (2013) “Solver GNU Linear Programming Kit (GLPK)”, <http://www.gnu.org/software/glpk/>, Junho.
- Gusek (2013) “Página Oficial do Ambiente de Modelagem GUSEK”, http://gusek.sourceforge.net/gusek_ptbr.html, Junho.
- Magatão, L. (2013) “Otimização de Sistemas. Notas de aulas da disciplina”, <http://pessoal.utfpr.edu.br/magatao/osi/>, Junho.
- Puccini, A.L.; Pizzolato, N.D. (1990) “Programação Linear”, 2ª Ed. Livros Técnicos, Rio de Janeiro-RJ.
- Scite (2013) “SCIntilla Text Editor”, <http://www.scintilla.org/SciTE.html>, Junho.
- Smelj (2013) “Secretaria do Esporte, Lazer e Juventude da Prefeitura de Curitiba/PR”, <http://www.curitiba.pr.gov.br/conteudo/equipe-smelj-secretaria-municipal-do-esporte-lazer-e-juventude/110>, Agosto.