Universidade Estadual de Campinas FCA - Faculdade de Ciência Aplicadas Curso de Engenharia de Produção e Manufatura

SCHEDULING

CHAVEAMENTO DE COMPETIÇÃO ESPORTIVA

<u>Disciplina:</u> ER 250A - Scheduling

<u>Data de Entrega:</u> 29.11.19

<u>Componentes do grupo:</u>

Daniel Lock Miletto…………………………………………………...RA:141631

Gabriel Barros Dávila……………………………………………….RA:171870

Gabriel Yamato Freitas……………………………………………...RA:172555

Guilherme Lima Correa…………………………………….….....RA:173811

1. INTRODUÇÃO

O gerenciamento de atividades esportivas é uma área promissora e bastante explorada para aplicações de Pesquisa Operacional. As competições esportivas envolvem muitos aspectos econômicos e logísticos. Os problemas desta área em geral possuem formulações simples, porém são problemas difíceis de serem resolvidos computacionalmente, visto que, possuem diversas restrições e múltiplos objetivos.

A programação de tabelas é uma das principais aplicações de Pesquisa Operacional em esportes. As ligas esportivas precisam de tabelas que satisfaçam diferentes tipos de restrições e que otimizem objetivos tais como a distância viajada pelas equipes durante o campeonato ou, em uma competição de menor porte, objetivos que evite conflito de horários entre jogos.

O problema mais abordado de programação de tabelas consiste em decidir quando e onde os jogos de um determinado campeonato serão realizados. Este problema pode ser abordado como sequenciamento de tarefas, usualmente, estas são representadas pelos jogos e os recursos pelas equipes que participam do torneio.

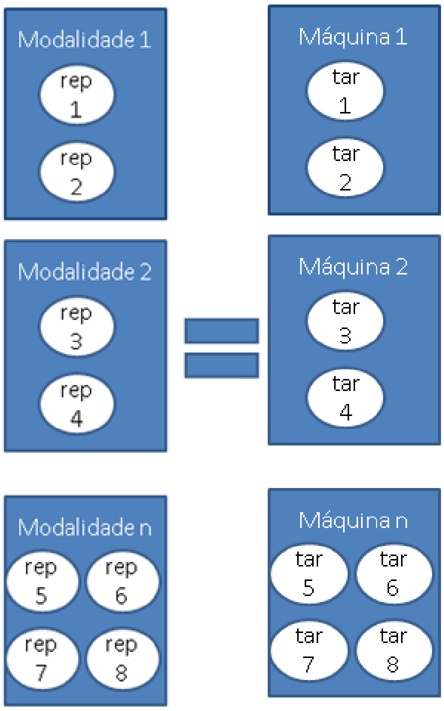
O objetivo do trabalho será analisar a tabela do campeonato *Arulíadas* organizado pela Associação de Repúblicas da Unicamp de Limeira (ARULI). Esta é uma organização estudantil sem fins lucrativos, reconhecida pela UNICAMP. Ela surgiu em 2014 como iniciativa de alunos de ambos os campi da UNICAMP de Limeira em busca de melhores condições de vida e segurança para as moradias estudantis. Atualmente possuem 33 repúblicas associadas que representam em torno de 240 pessoas. A organização conta com alunos de todos os cursos de graduação de Limeira, tanto da FCA (Faculdade de Ciências Aplicadas) quanto da FT (Faculdade de Tecnologia). No Aulíadas jogam às 9 modalidades somente 16 repúblicas masculinas e 8 femininas.

2. DESCRIÇÃO DO PROBLEMA

O Arulíadas é um campeonato com diversas modalidades entre as Repúblicas Universitárias que acontece anualmente, porém todo ano há problemas relacionados com o gerenciamento das partidas, ou seja, há muitas partidas que não são realizadas devido à falta de organização. As partidas masculinas acontecem em um local diferente das femininas, sendo assim cada sexo possui um único local reservado pela organização para os jogos, iniciando-se na sexta e terminando no domingo. Todas as repúblicas jogam todas as modalidades no estilo de jogo, conhecido popularmente como, “mata-mata”, isto é, a república derrotada é eliminada na modalidade e a vencedora segue na competição.

O problema analisado foi modelado como um sequenciamento de jogos, visto que, este pertence à área de otimização combinatória. O desafio principal do trabalho é alocar recursos limitados a tarefas ao longo do tempo de forma a otimizar um ou mais critérios. Neste trabalho, diferente da modelagem mais usual, as tarefas são representadas pelas equipes e as máquinas (recursos) são representadas por modalidades esportivas. Diante disso, o problema foi tratado como um open shop, pois todas as tarefas devem ser processada em cada uma das máquinas sem uma rota pré-estabelecida, ou melhor, todas as repúblicas devem disputar todas as modalidades sem uma ordem de precedência fixada. Cada modalidade é composta por duas ou mais equipes, esta quantidade de times por modalidade é um parâmetro conhecido. Para modelar uma partida considerou-se que duas ou mais atividades podem estar na mesma máquina.

Figura 1. Relação modelo realidade



O objetivo do trabalho é elaborar um chaveamento automatizado do campeonato respeitando as restrições, visto que, atualmente este processo é feito manualmente.

3. REVISÃO LITERÁRIA

O problema estudado neste trabalho trata da programação da tabela (Sports timetabling) do Arulíadas. Portanto, nesta seção serão abordados alguns trabalhos realizados na área de programação de jogos, tabelas e de sequenciamento (open shop) os quais nortearam a modelagem do desafio. Vale ressaltar, que este tipo de modelagem não foi o único analisado pelo grupo para resolver o problema.

O problema de programação de tabelas tem sido tratado de várias formas na Pesquisa Operacional usando técnicas diferentes como programação inteira, busca tabu, algoritmos genéticos e *simulated annealing.* A resolução deste tipo de problemas por métodos exatos é limitada, visto que, só é possível para pequenas dimensões. Para dimensões mais elevadas, a abordagem mais comum é através de técnicas aproximadas e heurísticas. A principal desvantagem destas técnicas é a impossibilidade de garantir a otimalidade da solução. Entretanto, as chamadas metaheurísticas, tais como Busca Tabu (Glover e Laguna, 1997), Simulated Annealing (Kirkpatrick et al., 1983) e Algoritmo Genético (Goldberg, 1989) possuem mecanismos para tentar escapar de ótimos locais ainda distante de um ótimo global. Gerando soluções de boa qualidade.

Costa (1995) desenvolveu um método híbrido combinando as técnicas de algoritmo genético e busca tabu para resolver um problema de escalonamento de jogos da Liga Americana de Hóquei. Crauwels e Van Oudheusen (2003) desenvolveram um procedimento em colônia de Formigas para tratar o problema de Programação de Jogos não espelhado (este impede apenas que os dois confrontos entre dois times não sejam consecutivos em um campeonato de dois turnos).

Além das referências de programação de tabelas, analisou-se referências bibliográficas de sequenciamentos de tarefas. Pinedo (2002, p.15), define Flow Shop da seguinte forma: Existem m máquinas em série. Cada trabalho tem que ser processado em cada uma destas máquinas. Todos os trabalhos têm que seguir o mesmo caminho (ex: todos têm que ser processados primeiro na máquina 1, depois na máquina 2...). Depois de ser concluído em uma máquina, um trabalho é inserido na fila da próxima máquina.

Já Pinedo (2002, p.16), define Open shop: Existem m máquinas. Cada atividade tem que ser processado em cada uma destas m máquinas. Mesmo assim, alguns destes tempos de processamento podem ser zero. Não existe restrição com relação ao caminho de cada atividade através das máquinas. A seqüência permite que o caminho seja determinado por cada atividade e diferentes atividades podem ter diferentes caminhos.

Com base nas pesquisas literárias, identificamos que o problema de programação de tabelas pode ser definido como um open shop. Portanto, o grupo realizou duas modelagens para verificar a eficiência das mesma. A primeira modelagem possui uma variável binária a mais que a segunda modelagem que armazena no programa se o slot de tempo está ocupado. Para isso, identifica-se quando a tarefa se inicia e soma o tempo de processamento, todos os slots ocupados são armazenados em ($O_{i,j,t}$).

3. MODELAGEM

3.1. MODELAGEM MATEMÁTICA 1

Índices:

$i = 1,..., I$ Representa cada república associadas.

$j = 1,..., J$ Representa as modalidades jogadas (esportes).

$t = 1, …, T$ Representa o tempo ao longo da competição.

$s = 1,..., S$ Representa o tempo que está sendo ocupado.

Parâmetros:

Q_j Quantidade de repúblicas que disputam a modalidade (j).

P_j Tempo médio de duração da modalidade (j).

Variáveis de decisão:

A solução do modelo deve resultar nos valores do seguinte conjunto de variáveis:

$X_{i,j,t}$

Variável binária que se igual a 1, indica que a república (*i)* iniciou o jogo da modalidade (*j)* no período (*t)*. Caso contrário se iguala a 0.

$O_{i,j,t}$

Variável binária que se igual a 1, indica que a república (*i)* está jogando a modalidade (*j)* no período (*t)*. Caso contrário se iguala a 0.

Há, ainda, restrições que demandam a utilização de variáveis auxiliares. No caso das restrições que garantem que um lote está sendo ocupado por duas repúblicas, temos a seguinte variável:

$W_{j,t}$

Variável binária que para cada modalidade (j) e tempo (t) assume valor 1 se é o slot de tempo está ocupado por duas repúblicas. 0 caso contrário.

Formulação matemática do problema:

$$Minimizar\ z = Cmax \qquad (1)$$

sujeito a:

$$\sum_{i=1}^{I} O_{i,j,t} = Q_j\,(1 - W_{j,t}) \qquad j = 1, \ldots, J \qquad t = 1, \ldots, T \qquad (2)$$

$$\sum_{i=1}^{I} O_{i,j,t} \geq (W_{j,t} - 1) \qquad j = 1, \ldots, J \qquad t = 1, \ldots, T \qquad (3)$$

$$\sum_{j=1}^{J} O_{i,j,t} \leq 3 \qquad i = 1, \ldots, I \qquad j = 1, \ldots, J \qquad (4)$$

$$\sum_{s = t + P_j - 1}^{S} O_{i,j,s} \geq P_j .\ X_{i,j,t} \qquad t = 1, \ldots, T \qquad i = 1, \ldots, I \qquad j = 1, \ldots, J \qquad (5)$$

$$\sum_{t=1}^{T} O_{i,j,t} = P_j \qquad i = 1, \ldots, I \qquad j = 1, \ldots, J \qquad (6)$$

$$\sum_{t=1}^{T} X_{i,j,t} = 1 \qquad i = 1, \ldots, I \qquad j = 1, \ldots, J \qquad (7)$$

$$Cmax >= X_{i,j,t} * (t + P_j) \qquad i = 1, \ldots, I \qquad j = 1, \ldots, J \qquad t = 1, \ldots, T \qquad (8)$$

Xi, j, t , Oi, j, t , $W j, t \in [0, 1]$ $i = 1, \ldots, I$ $j = 1, \ldots, J$ $t = 1, \ldots, T$ (9)

A função objetivo (1) consiste em minimizar o makespan. A primeira (2) e segunda restrição (3) garantem que cada slot de tempo seja ocupado por duas repúblicas de uma vez ou por nenhuma república (ocioso). Vale ressaltar que o *Qj* é a quantidade de repúblicas (*i*) que disputam uma partida da modalidade (*j*). Já a restrição (4) obriga que o número máximo de modalidades por república no mesmo tempo tem que ser menor ou igual a três (considera-se que uma república pode disputar no máximo três partidas ao mesmo tempo devido a quantidade de integrantes inscritos no campeonato). A restrição (5) é bem importante nesta modelagem, pois é nela que cria-se uma variável binária dependente (*Oi,j,t).* Nota-se que o tempo analisado no somatório é o $s = t + Pj - 1.$ É nessa restrição que armazena-se os slots que deverão ser ocupados. A partir de um início da modalidade (Xi,j,t = 1) adiciona-se o tempo de processamento e todos esses slots subsequentes são alocados nas variáveis Oi,j,t. A restrição (6) limita que uma república jogue todas modalidades necessariamente apenas uma vez. A última restrição (7) obriga que cada república inicie uma vez uma partida em uma modalidades, ou seja, junto com a restrição (6) garante que a república jogue todas as modalidades. A restrição (8) designa os valores do makespan (Cmax). Restrição (9) para as variáveis binárias do problema.

3.2. MODELAGEM MATEMÁTICA 2

Índices:

i = 1,..., I   Representa todas as repúblicas associadas.

j = 1,..., J   Representa as modalidades (esportes).

t = 1, …, T   Representa o tempo ao longo da competição..

s = 1,..., S   Representa o tempo que está sendo ocupado.

Parâmetros:

Qj Quantidade de repúblicas (i) que disputam a modalidade (j).

Pj Tempo médio de duração da modalidade (j).

$Modj$ Quantidades total de modalidades (j).

Variáveis de decisão:

A solução do modelo deve resultar nos valores do seguinte conjunto de variáveis:

$X\ i,j,t$

Variável binária que se igual a 1, indica que a república *(i)* iniciou o jogo da modalidade *(j)* no período *(t)*. Caso contrário se iguala a 0.

Há, ainda, restrições que demandam a utilização de variáveis auxiliares. No caso das restrições que garante que um lote está sendo ocupado ou começando por duas repúblicas, temos a seguintes variáveis:

Wj,t

Variável binária que para cada modalidade (j) e tempo (t) assume valor 1 se é o slot de tempo está ocupado por duas repúblicas. 0 caso contrário.

Vj,t

Variável binária que para cada modalidade (j) e tempo (t) assume valor 1 se é o slot começa com duas repúblicas. 0 caso contrário.

Formulação matemática do problema:

$$Minimizar\ z\ =\ Cmax \tag{8}$$

sujeito a:

$$\sum_{j=1}^{J} Xi,j,t = Modj \qquad i = 1, \ldots, I \qquad t = 1, \ldots, T \qquad (9)$$

$$\sum_{t=1}^{T} Xi,j,t = 1 \qquad i = 1, \ldots, I \qquad j = 1, \ldots, J \qquad (10)$$

$$\sum_{i=1}^{I} \sum_{s = máx\,(0,\, t-Pj+1)}^{S} Xi,j,s = 2\,(1 - Wj,t) \qquad j = 1, \ldots, J \qquad t = 1, \ldots, T$$

(11)

$$\sum_{i=1}^{I} \sum_{s=1}^{S} Xi,j,s \geq (Wj,t - 1) \qquad j = 1, \ldots, J \qquad t = 1, \ldots, T$$

(12)

$$\sum_{i=1}^{I} Xi,j,t = 2\,(1 - Vj,t) \qquad j = 1, \ldots, J \qquad t = 1, \ldots, T$$

(13)

$$\sum_{i=1}^{I} Xi,j,t \geq (Vj,t - 1) \qquad j = 1, \ldots, J \qquad t = 1, \ldots, T \qquad (14)$$

$$\sum_{j=1}^{J} \sum_{s=1}^{S} Xi,j,s \leq 3 \qquad i = 1, \ldots, I \qquad t = 1, \ldots, T \qquad (15)$$

$$Cmax >= Xi,j,t * (t + Pj) \qquad i = 1, \ldots, I \qquad j = 1, \ldots, J \qquad t = 1, \ldots, T \qquad (16)$$

$$Xi,j,t\,, \quad Oi,j,t\,, \quad W\,j,t \in [0, 1] \qquad i = 1, \ldots, I \qquad j = 1, \ldots, J \qquad t = 1, \ldots, T \qquad (17)$$

A função objetivo (8) consiste em minimizar o makespan. A primeira restrição (9) da segunda modelagem matemática garante que as repúblicas joguem todas as modalidades e a restrição (10) obriga que as repúblicas joguem todas as modalidades uma vez. As restrições (11 e 12) limitam que cada slot de tempo é ocupado por duas repúblicas ou nenhuma, caso hajam duas, garante a partida. As restrições (13 e 14) asseguram essa que ou duas repúblicas iniciam uma partida por slot, ou nenhuma república inicia uma partida neste slot. Por último, a restrição (15) obriga que o número máximo de modalidades por república no mesmo tempo tem que ser menor ou igual a três. A restrição (16) designa os valores do makespan (Cmax). Restrição (17) para as variáveis binárias do problema.

4. RESULTADOS OBTIDOS

A modelagem matemática 1 por apresentar mais variáveis binárias só consegue achar a solução ótima para problemas de pequenas dimensões. Concluímos que a variável $O_{i,j,t}$ assume muitos valores assim tornando o problema insolucionável computacionalmente. Para exemplificar isso, comparamos a solução da primeira modelagem com as soluções da segunda modelagem. Além do makespan ser menor na segunda modelagem, é possível encontrar a solução para problemas com dimensões maiores. E para problemas de mesma dimensão, a segunda modelagem continúa mais eficiente. Sendo assim, evitar a criação de variáveis de decisão pode definir se o problema é solucionável através de um método exato ou não. Segue os dados comparativos:

Utilizamos a modelagem matemática 1 para analisar um campeonato o qual , primeiramente, tem 2 repúblicas e ,em seguida, 4 repúblicas disputando 9 modalidades. Esta última dimensão foi considerada a limitante visto que dimensões maiores não possuem um tempo hábil de análise.

Figura 2: Gantt para 2 repúblicas e 9 modalidades.

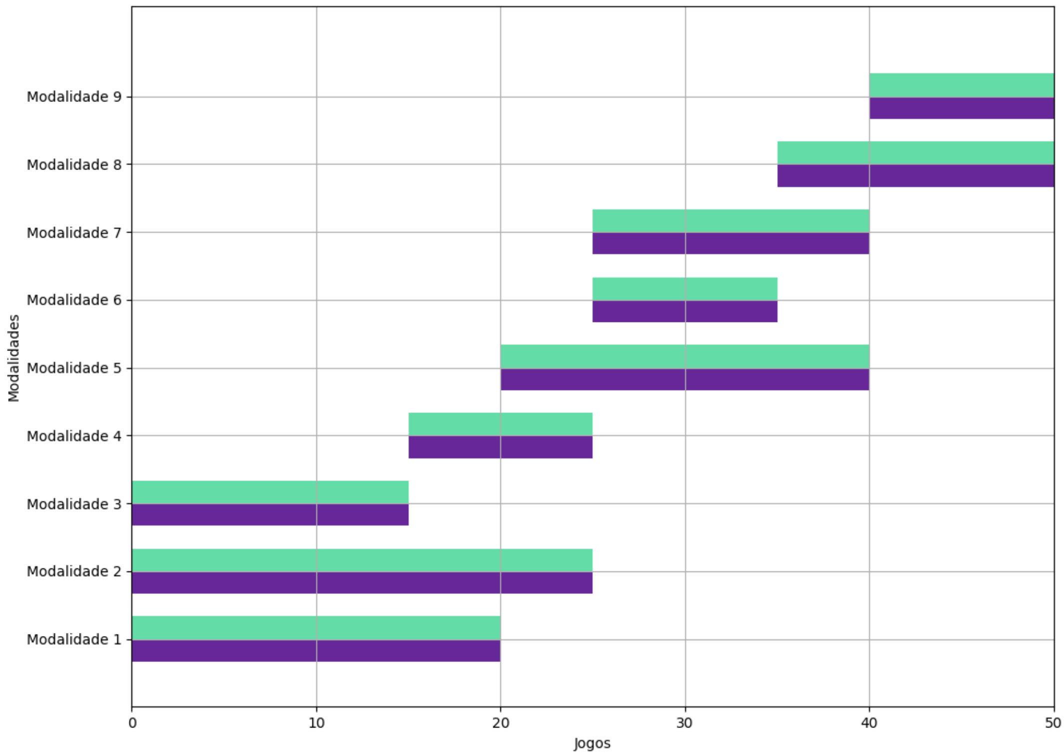
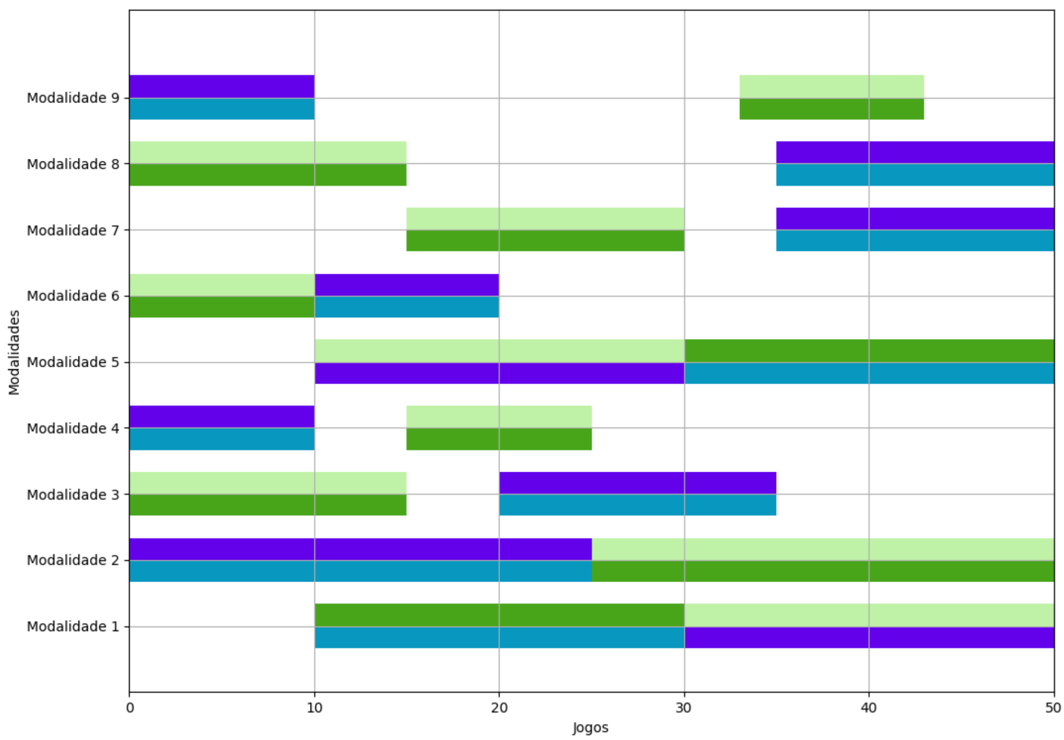


Figura 3: Gantt para 4 repúblicas e 9 modalidades.



A modelagem matemática 2 consegue analisar dimensões maiores. Ela foi capaz de analisar o programa em sua totalidade, ou seja, 16 repúblicas e 9 modalidades. Além disso, a modelagem é mais eficiente apresentando um makespan e um tempo de execução menor.

Figura 4: Gantt para 2 repúblicas e 9 modalidades.

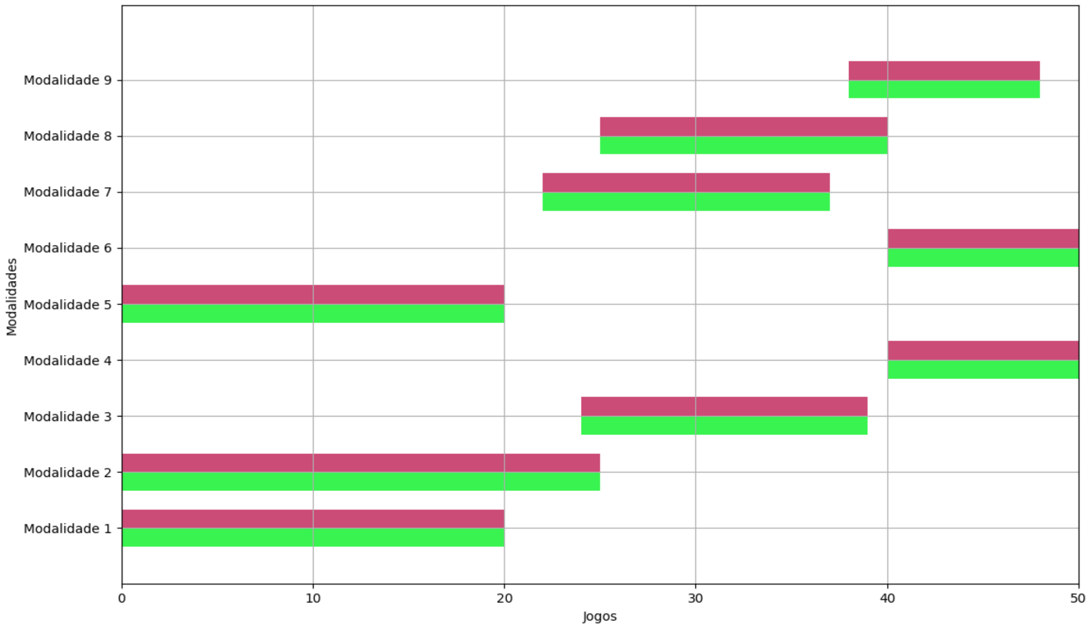


Figura 5: Gantt para 4 repúblicas e 9 modalidades.

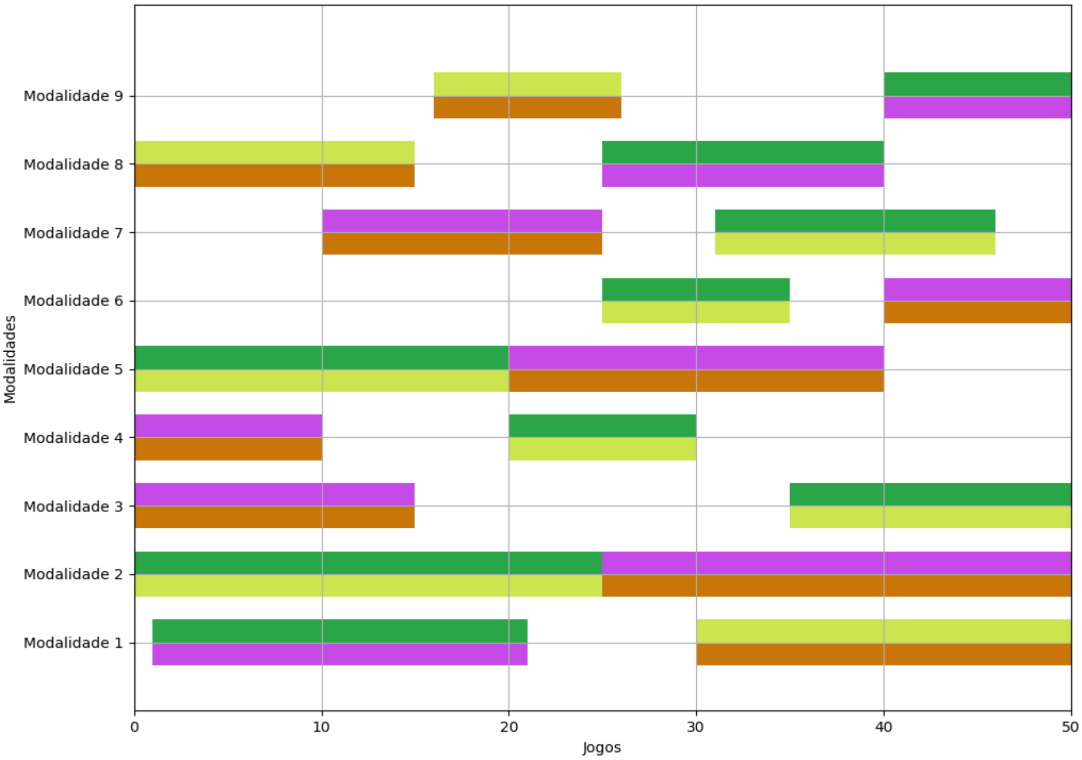


Figura 6: Gantt para 8 repúblicas e 9 modalidades.

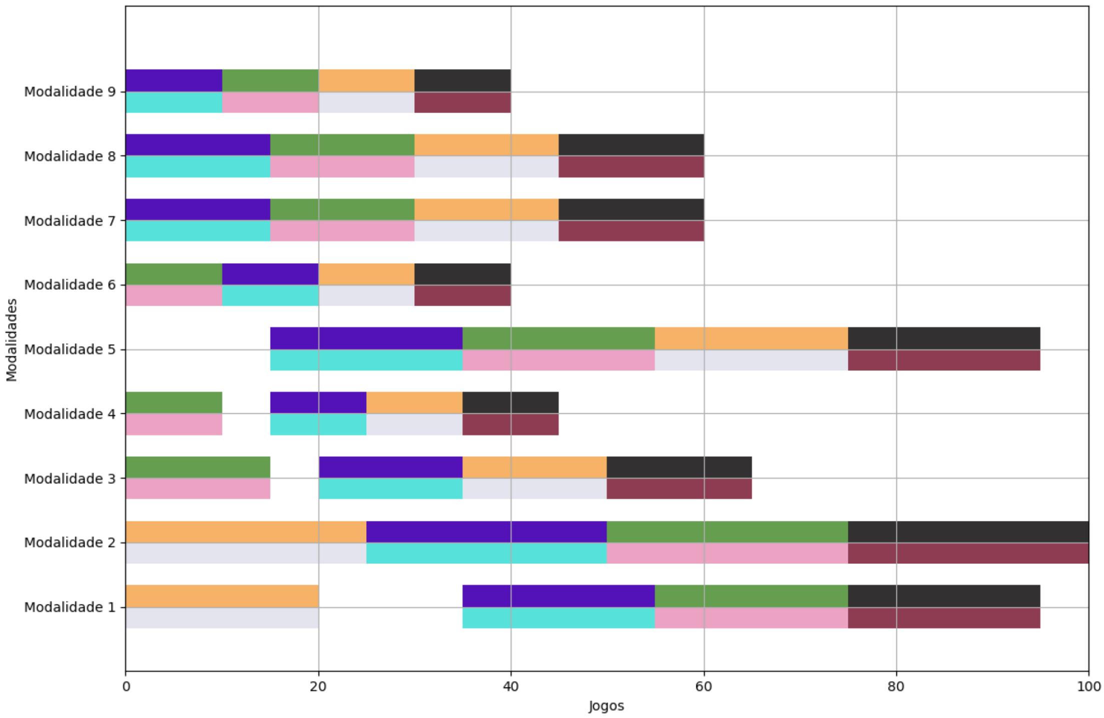
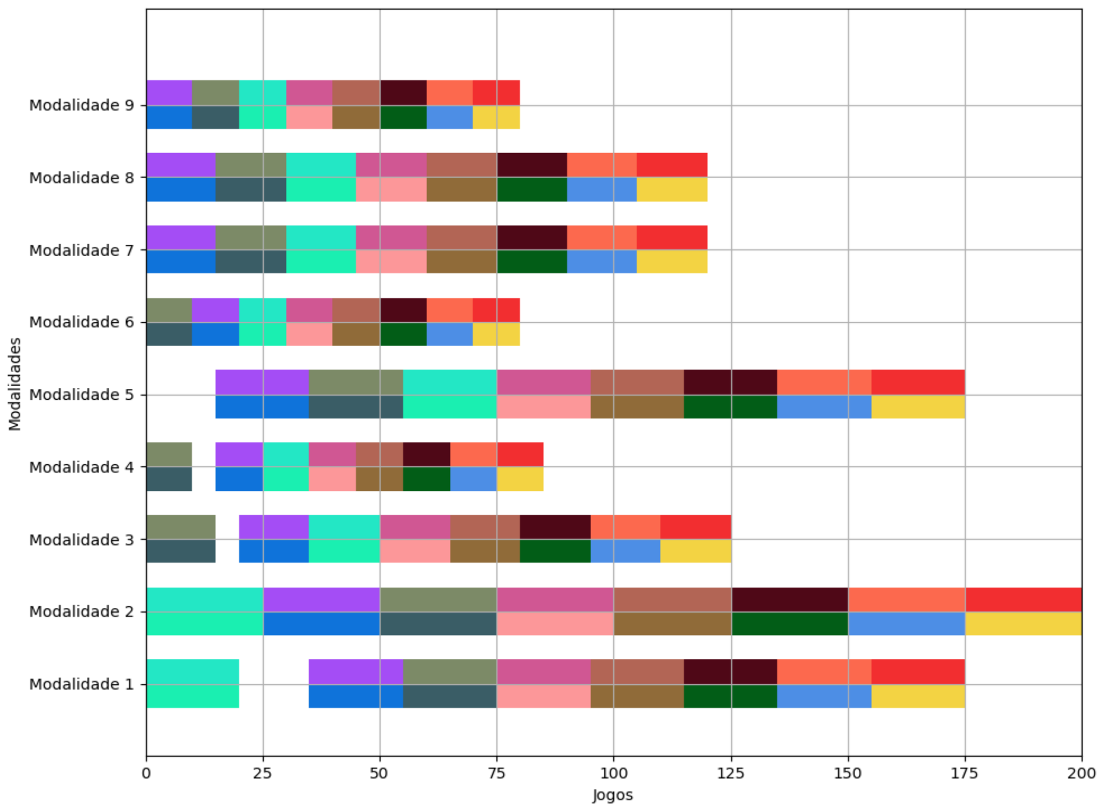


Figura 7: Gantt para 16 repúblicas e 9 modalidades.



Para analisar os resultados de maneira mais didática elaborou-se um tabela com os dados:

Tabela 1: Comparação de Makespan e tempo de execução em relação ao número de repúblicas entre modelagens.

	Repúblicas	Modalidade	Makespan (min)	Tempo de Execução (s)
Modelagem 1	2	9	50	5,56
	4	9	50	428,22
	8	9	-	-
	16	9	-	-
	32	9	-	-
Modelagem 2	2	9	50	0,56
	4	9	50	8,45
	8	9	100	2,26
	16	9	200	9,46
	32	9	400	120,54

A partir dos dados, os resultados estão de acordo com o esperado. A modelagem matemática 2 é superior a 1 em todos os aspectos. Portanto, para esses parâmetros e restrições é possível terminar o Arulíadas em um final de semana. Porém, o problema foi relaxado. Existe um problema relacionado a quantidades de repúblicas, o chaveamento só é perfeito caso os número de repúblicas (*i*) seja igual a 2^n. Se isso não acontecer, o chaveamento precisará que algum time pule alguma fase do campeonato (também conhecido popularmente como chapéu). Além desse relaxamento, podemos observar que as mesmas repúblicas se enfrentam em todas as modalidades, isso não é interessante para o campeonato, pois cria uma rivalidade entre os times. Há relaxamento em relação a restrição de disponibilidade de horários. Acreditamos que com todas essas restrições a mais o modelo exato não seria capaz de resolver o problema, sendo assim, a melhor opção seria uma heurística.

5. REFERÊNCIAS BIBLIOGRÁFICAS

GLOVER, F. e LAGUNA, M. (1997) Tabu Search. Kluwer Academic Publishers,Norwell, MA.

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

GOLDBERG, D. E. (1989) Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Berkeley.

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

CRAUWELS, H.; VAN OUDHEUSDEN, D. (2003) Ant Colony Optimization and Local Improvement. Proceedings of CP’AI’OR’03, Montreal, Canada.

PINEDO, Michael, 2002, Scheduling: Theory, algorithms, and systems – 2ª ed., New Jersey, Prentice Hall.