

Abordagens Heurísticas para Problemas de Otimização Combinatória

Attilio Provedel

attilio@ccje.ufes.br

Departamento de Ciência da Informação
CCJE/UFES

Tópicos

Visão geral

Aplicações

Abordagens heurísticas

Estudo de caso: o Problema do Corte de Peças

Estudo de caso: o Problema do Escalonamento

Escalonamento em uma Organização Virtual

Setembro de 2006

Abordagens Heurísticas para Problemas de Otimização Combinatória

1. Visão geral

OTIMIZAÇÃO COMBINATÓRIA

Processo de tentar encontrar a **melhor solução possível** para um problema de otimização (em geral, dentro de um limite de tempo).

Problema para o qual temos **diferentes soluções** possíveis. Tais soluções podem ser **avaliadas** e comparadas.

1. Considere um conjunto de **itens** e uma série de regras que podem ser usadas para selecionar alguns itens do conjunto.

2. Usando as regras, temos várias maneiras de selecionar os itens, criando subconjuntos.

3. Se a cada item estiver associado um **valor**, os subconjuntos criados também terão um valor (função dos valores de seus itens).

4. O problema de **Otimização Combinatória** consiste em encontrar, dentre todos os possíveis subconjuntos, aquele cujo valor seja o **melhor possível**.

2. Aplicações

O Problema das Atribuições:

- Um conjunto de n pessoas disponíveis para executar n tarefas;
- $C_{i,j}$: custo para execução da tarefa j pela pessoa i ;
- O objetivo é encontrar uma atribuição $\{p_1, p_2, \dots, p_n\}$ de custo mínimo;
- A solução é representada pela permutação $\{p_1, p_2, \dots, p_n\}$ dos números $\{1, 2, \dots, n\}$.

$$\sum_{i=1}^n c_{i,p_i}$$

O Problema da Mochila:

- Um conjunto de n itens disponíveis para serem colocados numa mochila de capacidade C ;
- Cada item i possui um valor v_i e ocupa c_i unidades de capacidade;
- O objetivo é determinar o subconjunto I (de itens) de valor máximo, considerando a capacidade da mochila;

$$\sum_{i \in I} c_i \leq C$$

$$\sum_{i \in I} v_i$$

- A solução é representada pelo subconjunto I de $\{1, 2, \dots, n\}$.

Abordagens Heurísticas para Problemas de Otimização Combinatória

O Problema do Roteamento de Veículos:

- Um depósito possui m veículos para realizar entregas aos seus n clientes;
- A capacidade do veículo k é C_k ;
- O pedido do cliente i ocupa c_i unidades de capacidade;
- A distância entre cada cliente i e j é dada por $d_{i,j}$;
- Cada veículo não pode viajar mais do que D unidades de distância;
- O objetivo é determinar a ordem das visitas de cada veículo, minimizando a distância total percorrida, considerando as restrições de capacidade e de distância e observando o atendimento a todos clientes;
- Considere que o conjunto dos n_k clientes visitados pelo veículo k é denotado por $\{p_{1,k}, p_{2,k}, \dots, p_{n_k,k}\}$;
- Subentende-se que o depósito seja representado pelos “clientes fictícios” $p_{0,k}$ e $p_{n_k+1,k}$;
- Formulação:

$$\text{Minimizar } \sum_{k=1}^m \sum_{i=0}^{n_k} d_{p_{i,k}, p_{i+1,k}}$$

$$\text{Sujeito a } \sum_{i=1}^{n_k} c_{p_{i,k}} \leq C_k, \quad k = 1, 2, \dots, m$$
$$\sum_{i=0}^{n_k} d_{p_{i,k}, p_{i+1,k}} \leq D, \quad k = 1, 2, \dots, m$$
$$\sum_{k=1}^m n_k = n$$

Abordagens Heurísticas para Problemas de Otimização Combinatória

O Problema MAX-SAT Ponderado:

- Dada uma coleção de cláusulas, onde cada cláusula i possui um peso w_i , determinar uma interpretação que maximize a soma dos pesos das cláusulas satisfeitas.

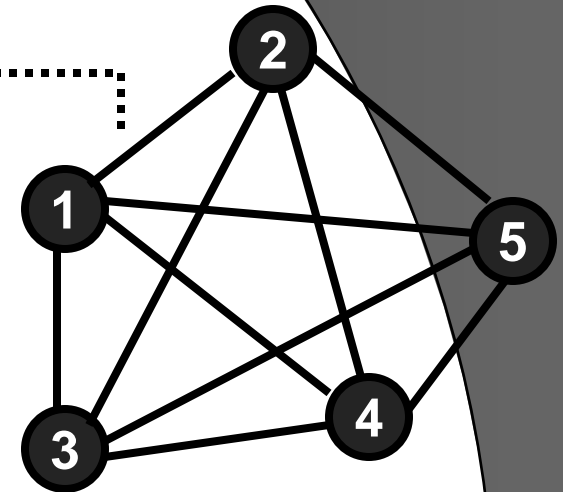
(\mathbf{V}_{X_1} , \mathbf{F}_{X_2} , \mathbf{F}_{X_3} , \mathbf{V}_{X_4} , ...)

$$\begin{aligned} C1 &= \mathbf{x}_1 \vee \sim\mathbf{x}_2 \vee \mathbf{x}_3 \dots \\ C2 &= \mathbf{x}_2 \vee \sim\mathbf{x}_3 \dots \\ C3 &= \sim\mathbf{x}_1 \vee \mathbf{x}_3 \dots \\ C4 &= \mathbf{x}_1 \vee \mathbf{x}_2 \vee \mathbf{x}_3 \dots \\ &\dots \end{aligned}$$

$w_1 = 14, w_2 = 27, w_3 = 82, w_4 = 3, \dots$

O Problema do Caixeiro Viajante:

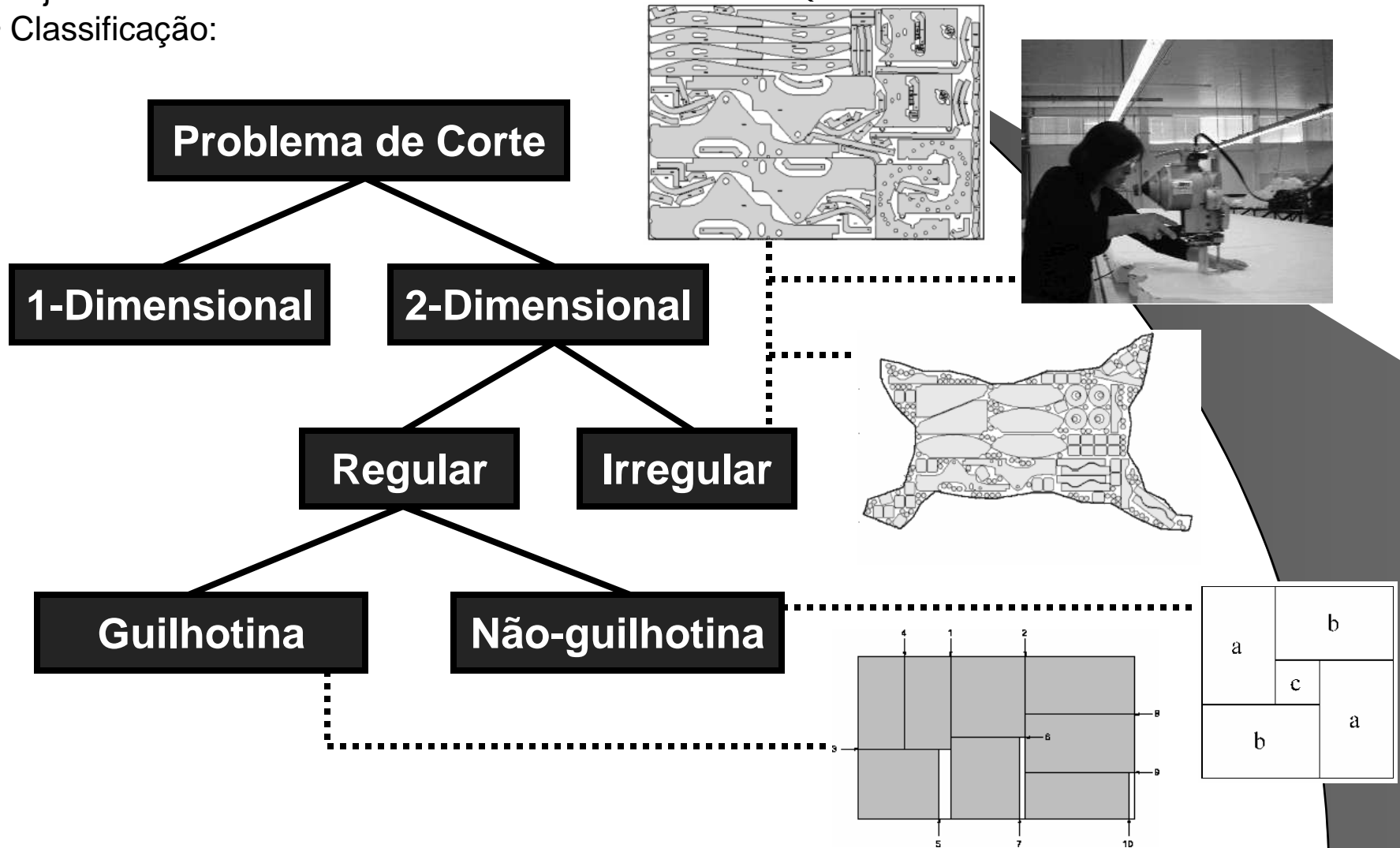
- Dado um conjunto de n idades, um caixeiro viajante tem que encontrar uma rota em que visite cada cidade uma única vez, iniciando e terminando a sua viagem na mesma cidade;
- Não importa a ordem com que as cidades são visitadas;
- De cada cidade pode-se ir diretamente a qualquer outra;
- O objetivo é minimizar a distância total percorrida;
- Total de possíveis soluções: $(n-1)!$.



Abordagens Heurísticas para Problemas de Otimização Combinatória

O Problema do Corte de Peças:

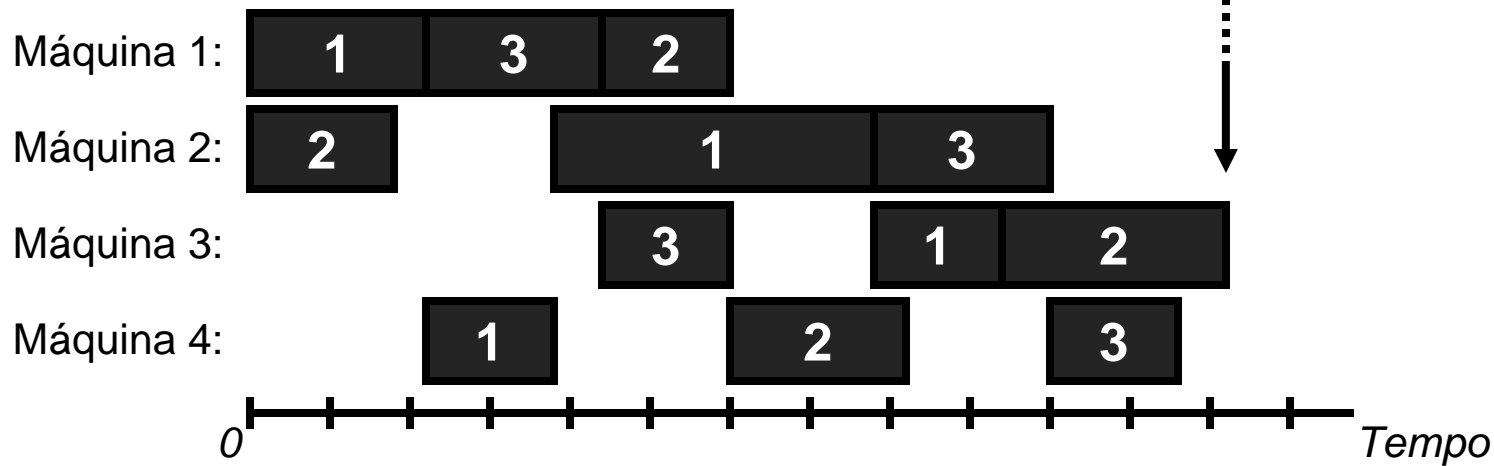
- Dado um conjunto de n peças, o objetivo é determinar como as peças devem ser cortadas a partir de uma peça maior, de maneira que a perda de material envolvida no processo de corte seja minimizada.
- Classificação:



Abordagens Heurísticas para Problemas de Otimização Combinatória

O **Problema do Escalonamento** (ambiente *job shop*):

- Considere um ambiente com m máquinas e n jobs;
- Cada tarefa de um *job* é realizada em uma única máquina;
- Cada *job* possui seu próprio roteiro a seguir, nas visitas às máquinas;
- O *job j* consome $p_{i,j}$ unidades de tempo de processamento na máquina i ;
- O objetivo é minimizar o **tempo de conclusão** do último *job*;
- Exemplo:
 - 4 máquinas e 3 jobs;
 - Visitas:
 - Job 1*: 1 - 4 - 2 - 3;
 - Job 2*: 2 - 1 - 4 - 3;
 - Job 3*: 1 - 3 - 2 - 4.
 - Uma solução:



3. Abordagens heurísticas

Considere o Problema do Caixeiro Viajante. Suponha que um computador seja capaz de listar todas as possíveis soluções para um problema com 20 cidades em 1 hora. Assim, com 21 cidades teríamos 20 horas de computação. Com 22 cidades, 17.5 dias. Com 25 cidades, 6 séculos. Devido ao crescimento exponencial do tempo de computação em função do tamanho do problema, obviamente a “força-bruta” não é uma abordagem interessante.

Solução:

- Métodos **exatos**;
- Métodos **heurísticos**.

Métodos que não garantem a produção de soluções ótimas, mas que tentam encontrar **boas soluções** (sub-ótimas) dentro de um tempo de computação razoável.

Abordagens:

- **Algoritmo A*** ;
- **Recozimento Simulado** (*Simulated Annealing*);
- **Algoritmo Genético**.

4. Estudo de caso: o Problema do Corte de Peças

O problema de corte aqui considerado consiste no problema 2-Dimensional irregular sobre placa retangular.

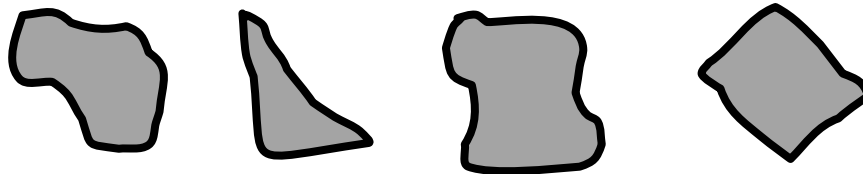
4.1. Abordagem baseada no Algoritmo A*

Placa:



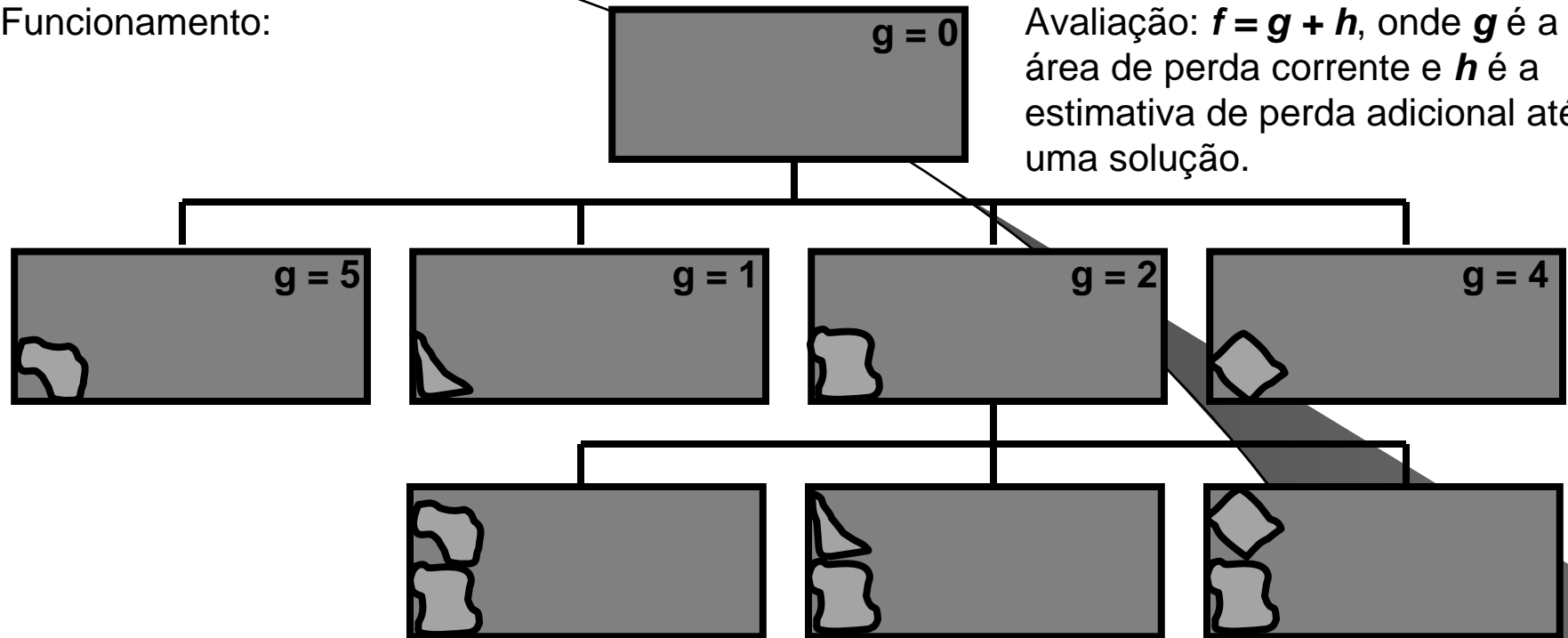
...

Peças:



Abordagens Heurísticas para Problemas de Otimização Combinatória

Funcionamento:



Parcela heurística:

- **Nula;**
- **Percentual;**
- **Retangular.**

Procedimento básico:

- Em cada passo, o nó mais promissor é expandido (geração de sucessores);
- O processo continua até que o nó mais promissor seja uma solução (um estado final).

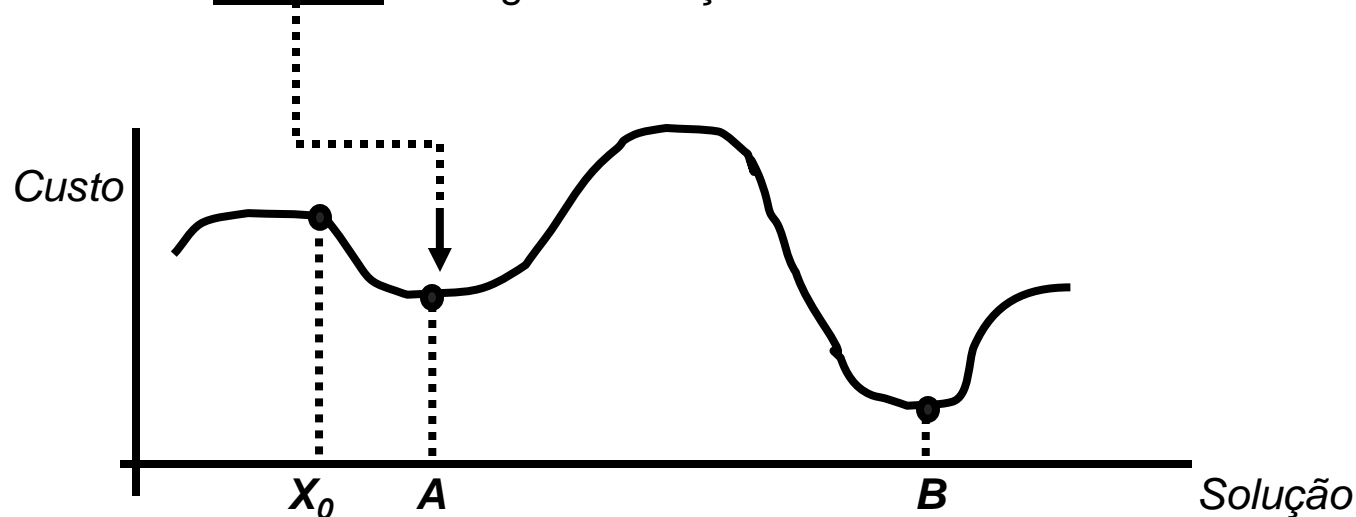
4.2. Abordagem baseada em Recozimento Simulado

Método de Melhoramento Iterativo:

- Espaço solução;
- Função custo;
- Conjunto de movimento que podem ser utilizados para modificar uma solução.

Funcionamento:

- Seja uma solução inicial X_0 ;
- Dada uma solução corrente X_i , suas vizinhas são examinadas até que uma solução X_{i+1} de **melhor valor** seja encontrada;
- O procedimento termina quando é alcançada uma solução que não possua nenhuma vizinha com melhor valor;
- Tendência: **prisão** em alguma solução não satisfatória.



Abordagens Heurísticas para Problemas de Otimização Combinatória

Recozimento Simulado (*Simulated Annealing*):

- Analogia: **mecânica estatística**;
- Questão fundamental: o que acontece a uma dada amostra de matéria no limite da baixa temperatura?
- Experimentos que determinam o estado de **baixa temperatura** de um sistema que forneça **energia mínima** são feitos por um método chamado **recozimento** (*annealing*);
- Procedimento:
 - Inicialmente, o sistema é elevado a uma **alta temperatura**;
 - **Lentamente**, a temperatura é **reduzida** e o sistema consome um **longo tempo** em cada valor de temperatura (se a temperatura for reduzida muito rapidamente, o sistema não teria tempo suficiente para alcançar o equilíbrio térmico);
 - Situação final: **baixa temperatura** e **baixa energia** do sistema.
- Analogia com **problemas de otimização**:

Disciplina central da física da matéria condensada, uma massa de métodos para analisar as propriedades agregadas do grande número de átomos encontrados em amostras de matéria sólida ou líquida.

ESTADO	SOLUÇÃO
ENERGIA	FUNÇÃO CUSTO
ESTADO FUNDAMENTAL	SOLUÇÃO ÓTIMA
RESFRIAMENTO RÁPIDO	RESFRIAMENTO ITERATIVO
RESFRIAMENTO CUIDADOSO	RECOZIMENTO SIMULADO

Abordagens Heurísticas para Problemas de Otimização Combinatória

Algoritmo Recozimento Simulado;

Início

S := Solução Inicial;

T := Temperatura Inicial;

Enquanto não(**Critério de Parada**) faça

 Enquanto não(**Equilíbrio**) faça

S' := Vizinha de **S**;

D := Custo(**S'**) - Custo(**S**);

Prob := MIN (1, $\exp(-D/T)$);

 Se (random(0, 1) \leq **Prob**) então

S := **S'**;

 Fim-Enquanto;

 Atualizar **T**;

Fim-Enquanto;

Fim.

O ponto central do algoritmo localiza-se no processo de **aceitação probabilística** de uma solução vizinha, mesmo que esta apresente um **custo maior**.

*Uma vez descrita a metodologia do Recozimento Simulado, para adaptarmos esta técnica para o Problema do Corte de Peças, precisamos estabelecer: **Solução Inicial, Temperatura Inicial, Cálculo da Função Custo, Geração de Solução Vizinha, Atualização de Temperatura, Sistema em Equilíbrio e Critério de Parada.***

Abordagens Heurísticas para Problemas de Otimização Combinatória

Demanda de peças:

- $P_i, i = 1, 2, \dots, n.$

Solução Inicial:

- Construa uma **permutação aleatória** para definição da seqüência de encaixe das peças;
- As peças devem ser encaixadas o mais **embaixo** e à **esquerda** possível da placa.
- Por exemplo ($n=5$): $P_3 - P_5 - P_2 - P_1 - P_4.$

Temperatura Inicial:

- Este valor deve ser **alto** o bastante para que qualquer solução vizinha possa ser aceita inicialmente, independente do seu custo.

Cálculo da Função Custo:

- A função **Custo(S)** é definida como a **área não utilizada** (área de perda) entre as peças no *layout* correspondente.

Geração de Solução Vizinha:

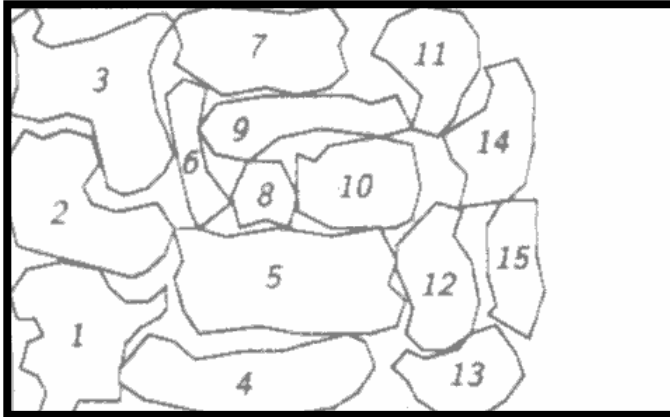
- A partir de uma lista de custos acumulados peça a peça, faça:
 - Selecione uma peça **p** aleatoriamente da seqüência corrente;
 - Fixe a seqüência corrente até a peça anterior à peça **p** e construa uma nova permutação aleatória a partir da peça **p**.

Abordagens Heurísticas para Problemas de Otimização Combinatória

- Visualização (geração de solução vizinha):

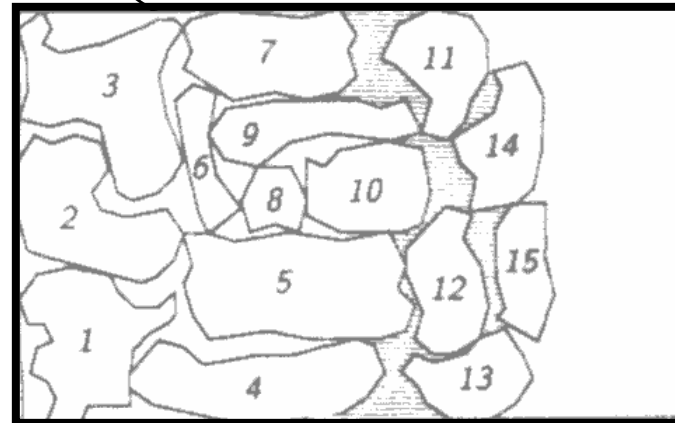
Solução corrente:

$L = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]$

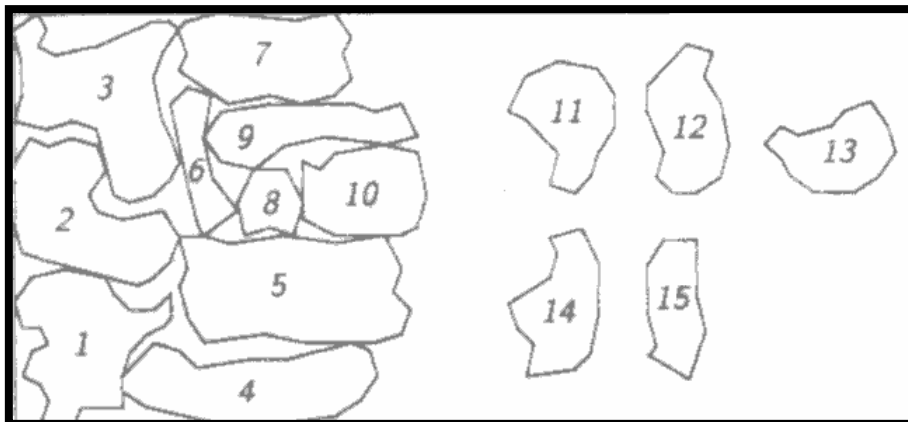


Solução corrente:

$L = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15]$

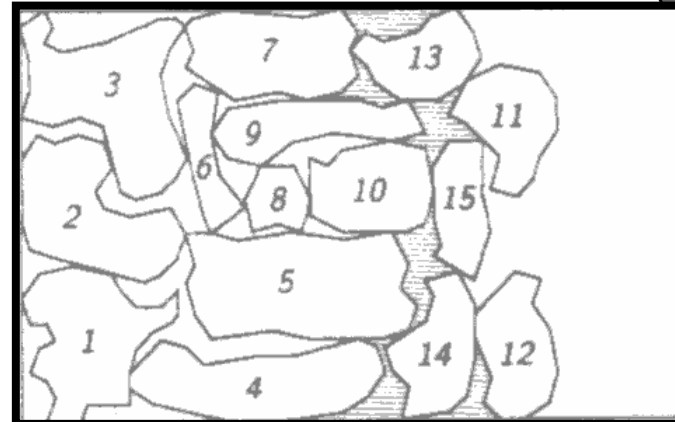


Geração da solução vizinha:



Solução vizinha:

$L' = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 13, 14, 15, 11, 12]$



Abordagens Heurísticas para Problemas de Otimização Combinatória

Atualização de Temperatura:

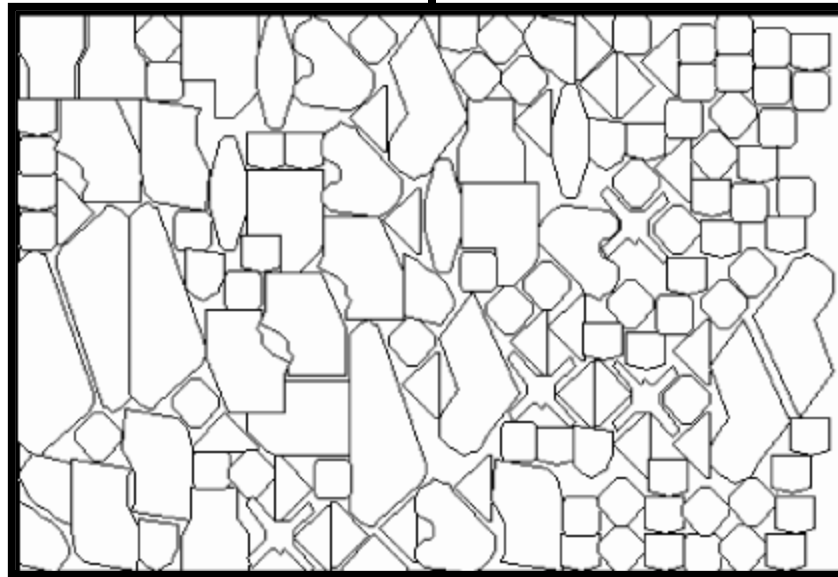
- $T := a \times T$, onde $0.85 < a < 0.95$.

Sistema em Equilíbrio:

- Estipulação de um número de iterações do *loop* interno do algoritmo.

Critério de Parada:

- O algoritmo termina quando nenhum melhoramento é alcançado após um certo número de iterações consecutivas. Então, o algoritmo apresenta a **solução encontrada**.



5. Estudo de caso: o Problema do Escalonamento

O problema de escalonamento aqui considerado consiste no problema Job Shop.

O **Problema do Escalonamento** (ambiente *Job Shop*):

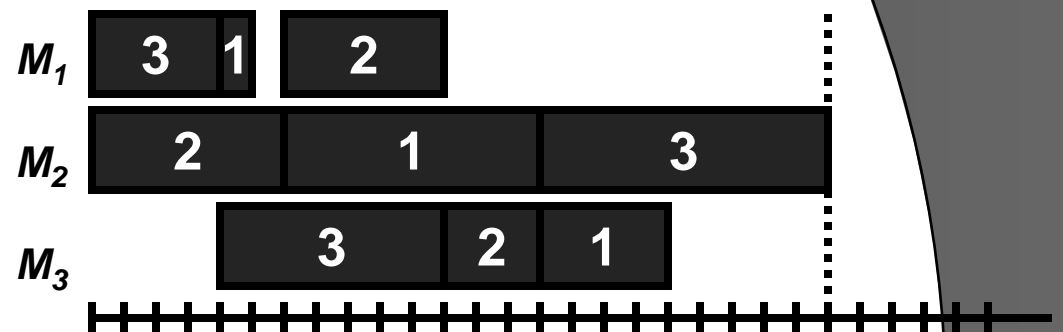
- Um conjunto de n jobs $J = \{J_1, J_2, \dots, J_n\}$ deve ser processado em um conjunto de m máquinas $M = \{M_1, M_2, \dots, M_m\}$ disponíveis. Cada *job* possui uma ordem de execução específica entre as máquinas, ou seja, um *job* é composto de uma lista ordenada de operações, cada uma das quais definida pela máquina requerida e pelo tempo de processamento na mesma. As restrições que devem ser respeitadas são:
 - Operações não podem ser interrompidas e cada máquina pode processar apenas um *job* de cada vez;
 - Cada *job* pode estar sendo processado em uma única máquina de cada vez,
 - Não existe restrição de precedência entre operações de diferentes *jobs*.
- O objetivo é minimizar o tempo de completção do último *job*.

Exemplo: **3 jobs** e **3 máquinas**.

Job	-----	Visitação	-----
1	1 / M_1	8 / M_2	4 / M_3
2	6 / M_2	5 / M_1	3 / M_3
3	4 / M_1	7 / M_3	9 / M_2

Uma solução (aliás, ótima):

23



5.1. Abordagem baseada em Algoritmos Genéticos

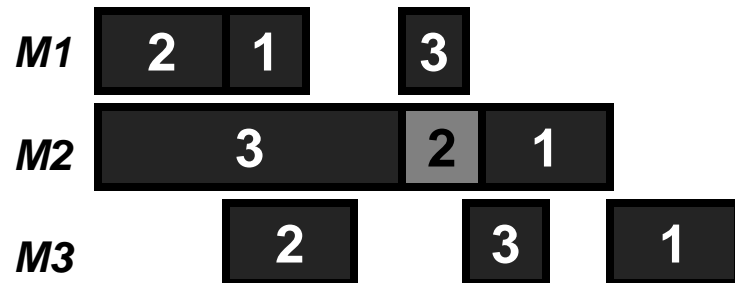
Exercício...

5.2. Abordagem baseada em Recozimento Simulado

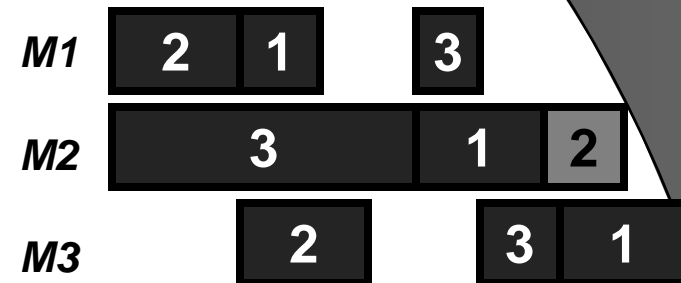
Geração de **solução vizinha**:

- Selecionar um *job* em uma máquina;
- Trocar o *job* de posição na seqüência associada à máquina;
- Exemplo:

Solução corrente



Solução vizinha



Agora, o método irá conter uma variação no que se refere à estrutura de vizinhança.

Nova abordagem:

- Várias **estruturas de vizinhança** poderão ser utilizadas durante a busca.

Abordagens Heurísticas para Problemas de Otimização Combinatória

Algoritmo Recozimento Simulado com Vizinhanças Variáveis;

Início

$S :=$ Solução Inicial;

$k := 1$;

Enquanto ($k < max$) faça

$S_1 :=$ Vizinha de S usando N_k ;

$T :=$ Temperatura Inicial;

Enquanto não(Critério de Parada) faça

Enquanto não(Equilíbrio) faça

$S_2 :=$ Vizinha de S_1 usando N_k ;

$D :=$ Custo(S_2) - Custo(S_1);

$Prob :=$ MIN (1, $\exp(-D/T)$);

Se (random(0, 1) $\leq Prob$) então

$S_1 := S_2$;

Fim-Enquanto;

Atualizar T ;

Fim-Enquanto;

Se Custo(S_1) < Custo (S) então

$S := S_1$;

$k := 1$;

Senão

$k := k + 1$;

Fim-Se;

Fim-Enquanto;

Fim.

N_k : k mudanças de posição de *jobs*.

6. Escalonamento em uma Organização Virtual

6.1. Descrição da Organização Virtual

Estrutura:

- Rede de **pequenas organizações** independentes e geograficamente dispersas.

Objetivo:

- A partir de oportunidades de serviço surgidas no **setor de cortes industriais**, as organizações são unidas de forma cooperativa e complementar para atender a tais oportunidades.

Área de atuação:

- Confecção, calçados, metal-mecânico, vidro, construção civil, indústria naval, etc.

Serviços comuns:

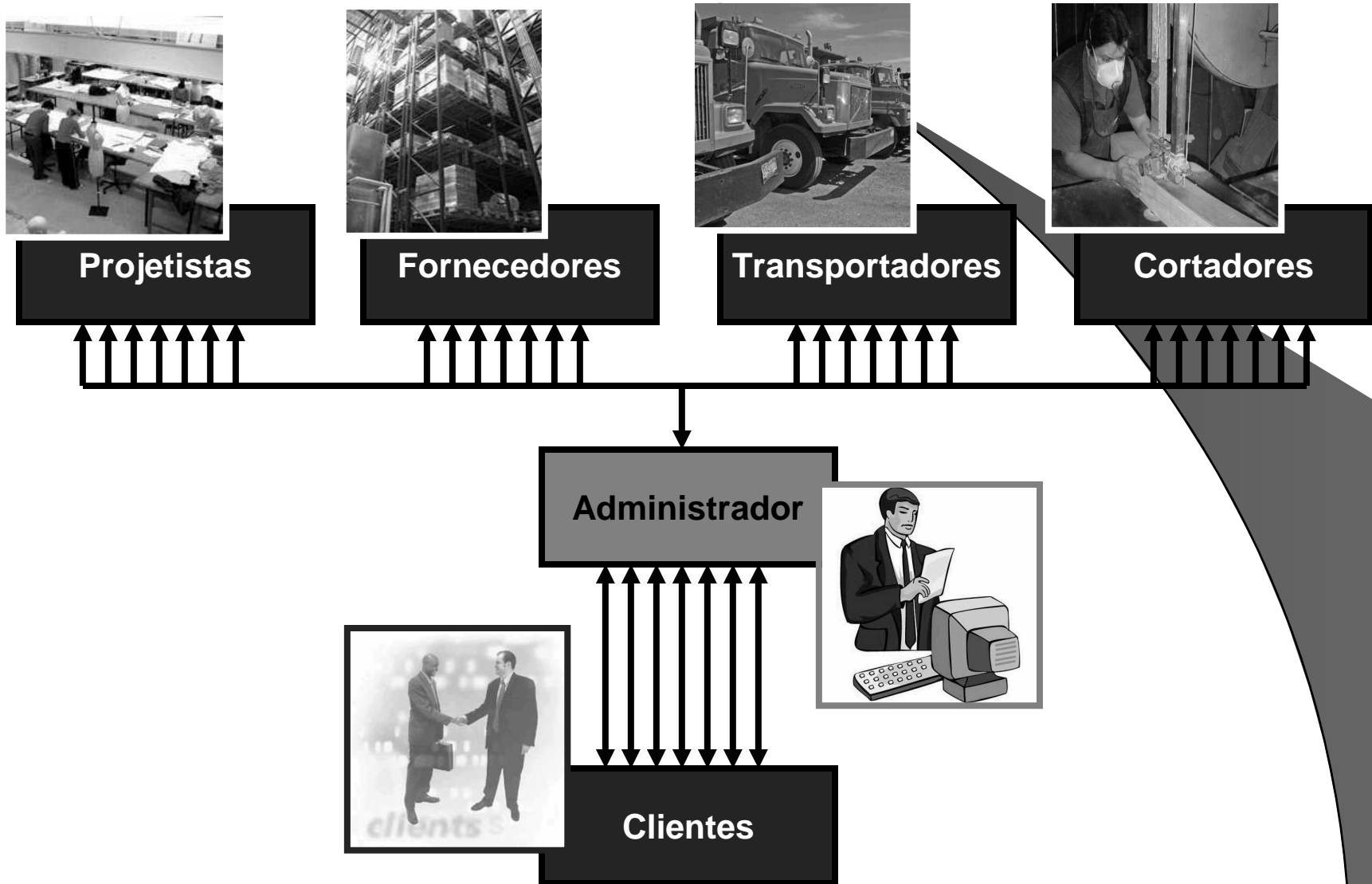
- **Fornecimento** de matéria-prima;
- **Projeto** de *layout* para corte;
- **Transporte** (de matéria-prima e produto cortado);
- **Corte** de matéria-prima.

Especialidades:

- Cada parceiro (servidor) da rede é capaz de executar um, e somente um, serviço.

Abordagens Heurísticas para Problemas de Otimização Combinatória

Ilustração:



Abordagens Heurísticas para Problemas de Otimização Combinatória

Os servidores:

- **Projetistas:**

- A partir do pedido do cliente, o projetista é capaz de produzir bons planos de corte.

- **Fornecedores:**

- A partir do requerimento de matéria-prima informado pelo projetista, o fornecedor especializado tem a função de disponibilizá-la para o transporte até o cortador.

- **Transportadores:**

- O transportador é encarregado de transportar matéria-prima do fornecedor para o cortador e do cortador para o cliente.

- **Cortadores:**

- A partir do plano de corte produzido pelo projetista, o cortador especializado efetivamente executa os cortes sobre a matéria-prima e disponibiliza as peças cortadas para o transporte até o cliente.

Cada pedido é composto de **tipo de corte**, **tipo de matéria-prima**, **dados para corte** e **localização do cliente**.

O administrador:

1. Elaboração de uma **base de pedidos** de clientes;
2. Envio dos pedidos aos servidores para que **orçamentos** sejam propostos;
3. Análise dos orçamentos e atribuição/seqüenciamento dos pedidos junto aos servidores, considerando **dois objetivos**:
 - Minimização do **preço médio** para atendimento aos pedidos;
 - Minimização do **tempo médio** de término dos atendimentos aos pedidos.

Otimização
multi-objetivo

Cada orçamento é composto de **preço** do serviço e **tempo** do serviço.

6.2. Abordagem baseada em Recozimento Simulado

Exemplo:

- Sejam 3 pedidos e 3 estágios (**A**, **B** e **C**) de servidores com 2 servidores em cada estágio;
- Considere os seguintes orçamentos (**preço e tempo**) informados pelos servidores:

Estágio	Servidor	P ₁	P ₂	P ₃
A	S ₁	(23,03)	(17,05)	(26,07)
	S ₂	(19,04)	(22,06)	(20,03)
B	S ₃	(36,03)	(25,02)	(17,06)
	S ₄	(32,04)	(31,03)	(21,04)
C	S ₅	(14,02)	(18,04)	(13,05)
	S ₆	(12,03)	(21,03)	(16,04)

- **Solução inicial:**

PASSO 1. Para cada pedido em cada estágio, selecionar aleatoriamente um servidor;

PASSO 2. Seqüenciar aleatoriamente os pedidos, nos servidores selecionados.

PASSO 1:

P₁:

- S₂ em **A**, S₄ em **B** e S₅ em **C**.

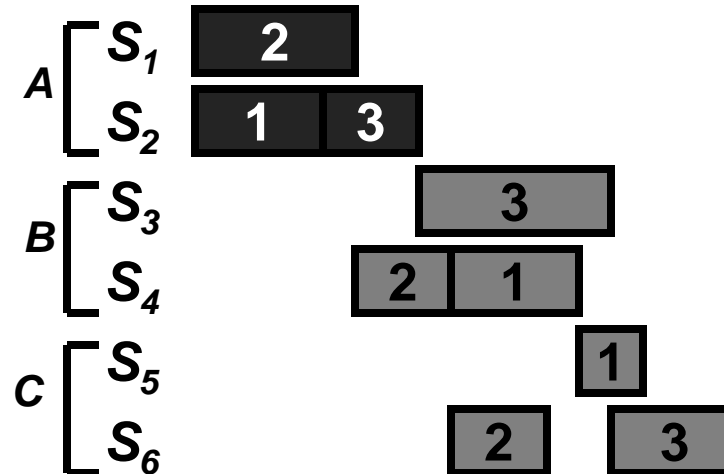
P₂:

- S₁ em **A**, S₄ em **B** e S₆ em **C**.

P₃:

- S₂ em **A**, S₃ em **B** e S₆ em **C**.

PASSO 2:



Preço médio: 65.3

Tempo médio: 14.0

Custo da solução:

(65.3, 14.0)

Abordagens Heurísticas para Problemas de Otimização Combinatória

- **Geração de solução vizinha:**

PASSO 1. Selecionar um estágio;

PASSO 2. Selecionar um pedido em um servidor do estágio selecionado;

PASSO 3. Trocar o pedido de posição na seqüência associada ao servidor ou trocar o pedido de servidor;

Algoritmo **Recozimento Simulado**;

Início

...

$S' :=$ Vizinha de S ;

$D :=$ Custo(S') - Custo(S);

$Prob :=$ MIN (1, $exp(-D/T)$);

...

Como calcular isso?

Por exemplo, se $Custo(S') = (58.7, 16.2)$ e $Custo(S) = (65.3, 14.0)$, como calcular “ $(65.3, 14.0) - (58.7, 16.2)$ ”?

E depois, como calcular isso?

Estratégia:

- Aplicar o algoritmo separadamente para cada objetivo, obtendo (P^*, T) e (P, T^*) ;

- Para uma dada solução S' de custo (P', T') , realizar a combinação linear dos objetivos (transformação em um problema com um único objetivo), atribuindo um peso (w_P e w_T) para cada objetivo:

$$Custo(S') = w_P \times \frac{(P' - P^*)}{(P - P^*)} + w_T \times \frac{(T' - T^*)}{(T - T^*)}$$