

# Um Método Para Planejamento de Produção em Sistemas de Manufatura Flexível

Gustavo Silva Paiva  
guustavo.paiva@gmail.com  
Marco Antonio Moreira de Carvalho  
mamc@iceb.ufop.br

Universidade Federal de Ouro Preto

10 de outubro de 2016

## Contextualização

Recentemente, Sistemas de Manufatura Flexível (SMF) têm sido adotado por empresas para aumentar a flexibilidade e diversificação da produção:

- Permite que uma maior gama de produtos seja produzida;
- Com um baixo custo adicional é possível a produção de novos produtos;
- Permitir uma rápida adequação à produção de novos produtos, ou imprevistos.

## Sistemas de Manufatura Flexível

O tipo mais comum de SMF, principalmente em empresas metalúrgicas, é caracterizado por utilizar **máquinas flexíveis**.

- A flexibilidade é referente à capacidade de configuração das ferramentas utilizadas na máquina;
- Cada produto requer um diferente conjunto de ferramentas (e.g. lâminas de corte, brocas de perfuração, etc) para sua fabricação.

## Máquina Flexível

Uma máquina flexível possui as seguintes características e consequências:

- Há um compartimento de capacidade limitada para o carregamento de ferramentas;
- Para se fabricar um produto é necessário carregar todas as ferramentas necessárias com antecedência;
- Entre a fabricação de diferentes produtos em sequência, eventualmente serão necessárias **trocas de ferramentas**;
- Para efetuar estas trocas, é necessário interromper a linha de produção dado que a máquina deve ser desligada.



Figura 1: Foto de uma máquina flexível com compartimento para 5 ferramentas.

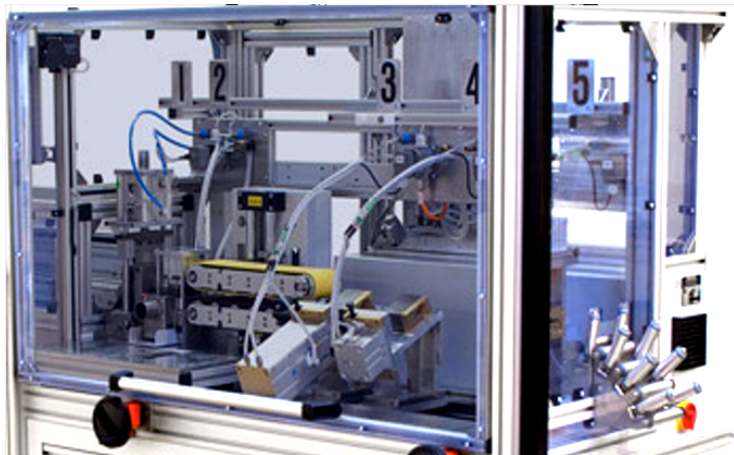


Figura 2: Foto de uma máquina flexível com compartimento para 5 ferramentas.



Figura 3: Foto de uma ferramenta de uma máquina flexível.



Figura 4: Foto de uma ferramenta de uma máquina flexível.



## Descrição do Problema

A partir de uma demanda por produtos, predeterminada, é necessário a criação de um plano de produção para que uma máquina flexível cumpra esta demanda:

- Este plano tem como objetivo minimizar o tempo ocioso da máquina, maximizando a produtividade;
- Considera-se a manufatura de um produto como sendo uma tarefa a ser executada.

## Plano de Produção

O plano de produção consiste em:

- 1 Determinar a ordem em que as tarefas serão executadas; e
- 2 Decidir quando realizar cada troca de ferramentas e quais ferramentas serão trocadas, de maneira a viabilizar a produção.

A primeira etapa é o **Problema de Minimização de Trocas de Ferramentas (MTSP)**;

Já a segunda pode ser determinada tempo polinomial determinístico pelo algoritmo *Keep Tool Needed Soonest* (KTNS).

## Instância

Uma instância do MTSP pode ser descrita pelos seguintes dados:

- O conjunto  $T = \{1 \cdots n\}$  de tarefas que devem ser processadas;
- O conjunto  $F = \{1 \cdots m\}$  de ferramentas disponíveis;
- Para cada tarefa  $j \in T$ , um conjunto de ferramentas  $F_j \in F$  necessárias para processamento da mesma; e
- A capacidade  $C$  do compartimento de ferramentas da máquina.

## Instância e Representação Computacional

Tabela 1: Instância do MTSP.

Tarefas	1	2	3	4	5
Ferramentas	1	1	3	2	1
	2	3	4	3	4
	4		5	5	6
Capacidade do compartimento = 3					

Tabela 2: Matriz binária.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1

- Crama et al. (1994) definiram *1-block*: um conjunto de entradas consecutivas em uma mesma linha da matriz com o valor igual a 1.

## Solução

Uma solução do MTSP é obtida pela permutação  $\phi$  das colunas da matriz  $A$ , resultando na matriz permutação  $A^\phi$ ;

O número de trocas de ferramentas em uma solução  $A^\phi$  é equivalente ao número de inversões em cada uma das linhas, e pode ser calculado utilizando o algoritmo KTNS.

## Exemplo de Solução

Tabela 3: Matriz  $A^\phi$  com  $\phi = [3, 4, 1, 5, 2]$ .

$F \backslash \phi$	3	4	1	5	2
1	0	0	<b>1</b>	1	1
2	0	<b>1</b>	1	0	0
3	<b>1</b>	1	0	0	<b>1</b>
4	<b>1</b>	0	<b>1</b>	1	<u>1</u>
5	<b>1</b>	1	0	0	0
6	0	0	0	<b>1</b>	0

- Para executar essa sequência de tarefas são necessárias 8 trocas de ferramentas;
- Ferramentas em negrito indicam a inserção das mesmas;
- Ferramentas sublinhadas não são necessárias para execução da tarefa em questão mas são mantidas para, eventualmente, evitar trocas.

## Motivação

- O MTSP é um problema NP-Difícil;
- Aplicabilidade prática do problema em empresas metalúrgicas e produção de placas de circuitos impressos;
- Foi constatado que os planos de produção das empresas podem ser facilmente melhorados.

## Contribuições

- 1 Nova representação do problema, baseada em Teoria dos Grafos;
- 2 Uma nova heurística construtiva;
- 3 Um novo método de busca local; e
- 4 Incorporação das Contribuições anteriores em uma metaheurística.



## Contribuição 1 – Grafo de Ferramentas

Um *Grafo de Ferramentas* é definido como  $G = (V, E)$ , tal que:

- $V$  é o conjunto de vértices representando as ferramentas;
- $E$  é o conjunto de arestas  $\{i, j\}$  representando a utilização das ferramentas  $i$  e  $j$  por uma mesma tarefa;
- Não existem *loops* e arestas paralelas;
- É proposto um peso para cada aresta  $\{i, j\}$  representando o número de vezes em que as ferramentas  $i$  e  $j$  aparecem juntas em uma mesma tarefa.

## Contribuição 1 – Grafo de Ferramentas

Tabela 4: Instância do MTSP.

Tarefas	1	2	3	4	5
Ferramentas	1	1	3	2	1
	2	3	4	3	4
	4		5	5	6
Capacidade do compartimento = 3					

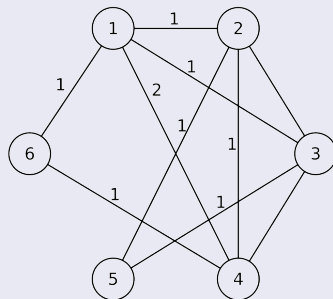


Figura 5: Grafo de ferramentas da instância apresentada anteriormente.

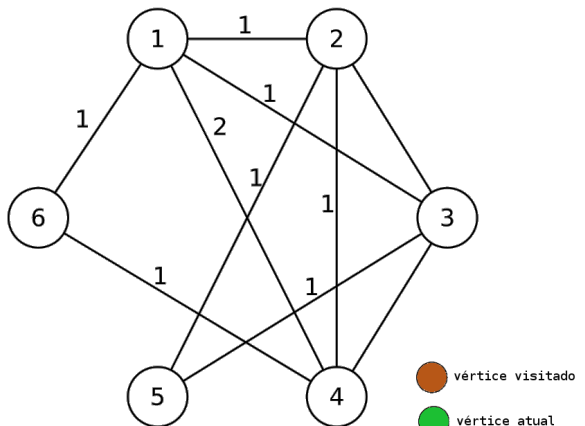
## Contribuição 2 – Heurística Construtiva

A ideia é procurar entre as ferramentas aquelas que são mais utilizadas em conjunto, baseando-se na frequência dessas ocorrências;

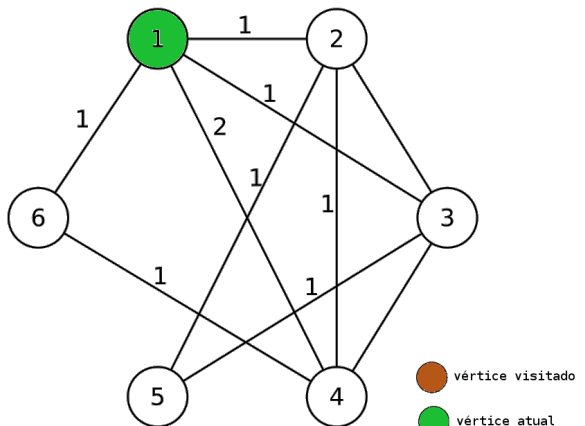
A partir de um vértice inicial, executa-se uma Busca em Largura (ou *Breadth-First Search* – BFS):

- A busca é guiada pela aresta de maior peso.

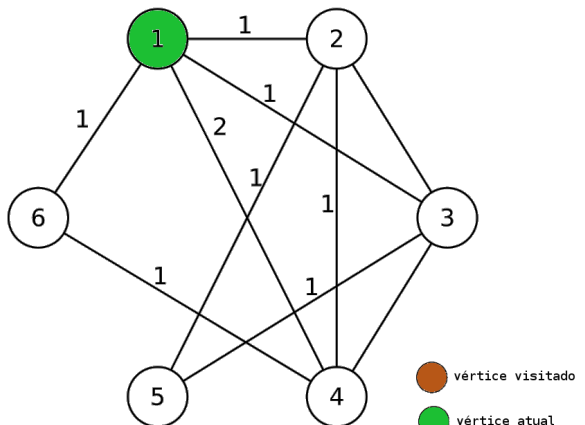
Ao final, a ordem de exploração  $F_\phi$  é obtida e representa a sequência de ferramentas.



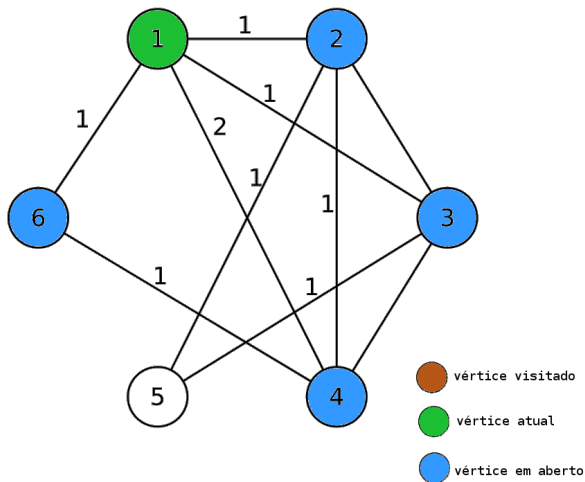
$$Q = \{1\}$$
$$F_\phi = \{\}$$



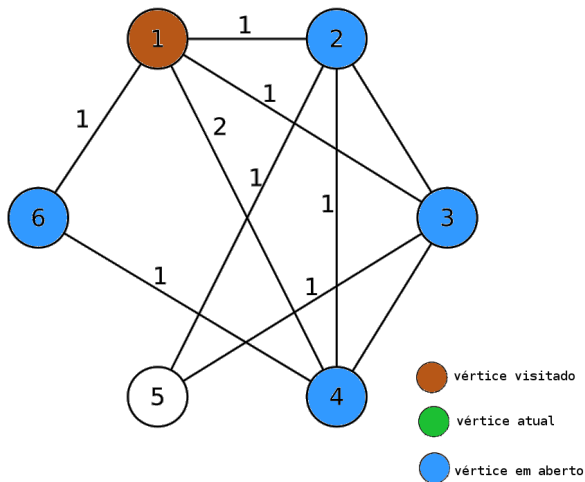
$$Q = \{\}$$
$$F_{\phi} = \{1\}$$



$$Q = \{\}$$
$$F_\phi = \{1\}$$



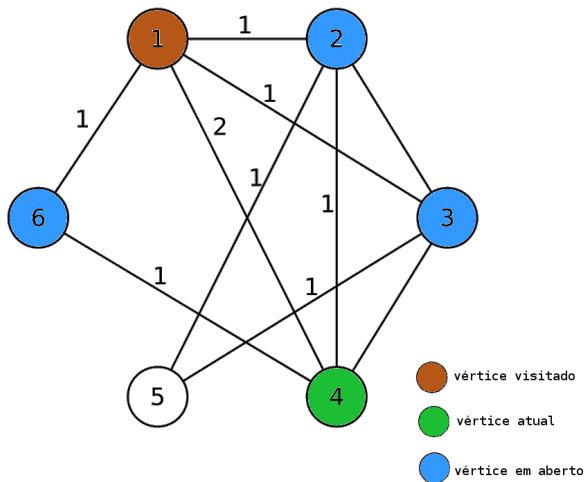
$$Q = \{4, 2, 3, 6\}$$
$$F_\phi = \{1\}$$



$$Q = \{4, 2, 3, 6\}$$

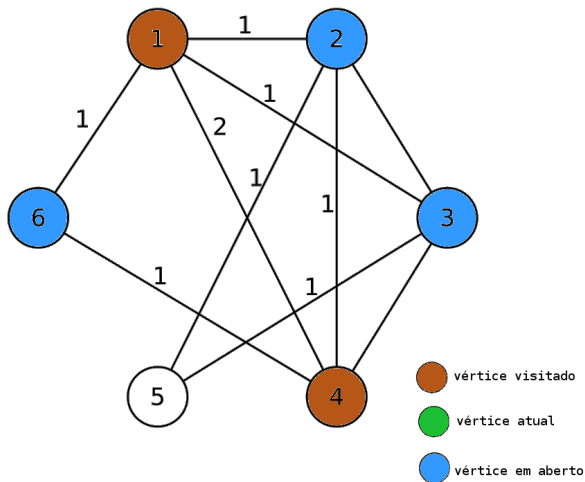
$$F_\phi = \{1\}$$





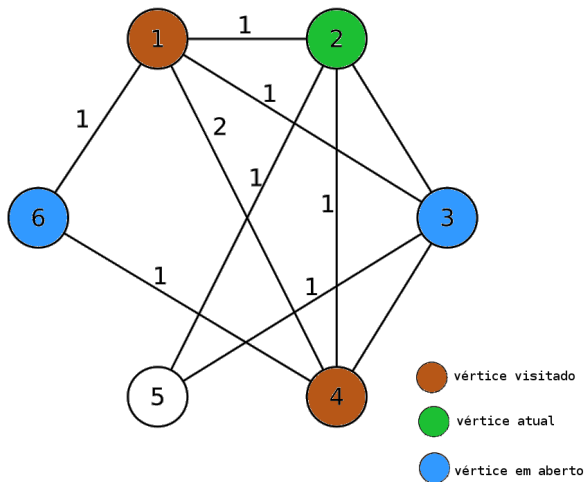
$$Q = \{2, 3, 6\}$$

$$F_\phi = \{1, 4\}$$

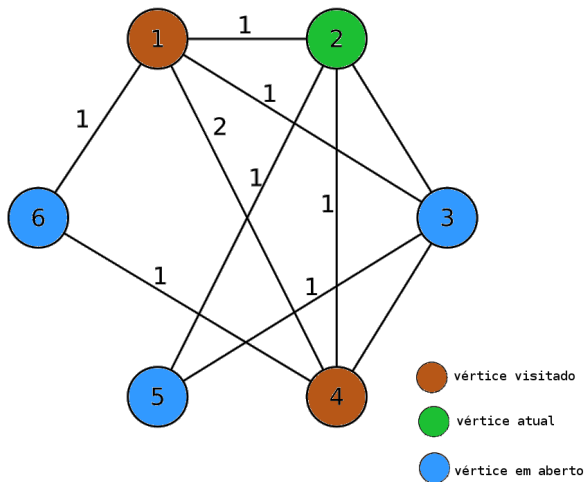


$$Q = \{2, 3, 6\}$$

$$F_\phi = \{1, 4\}$$

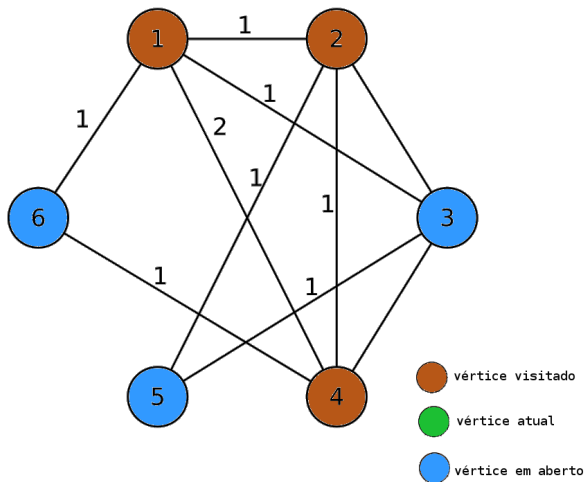


$$Q = \{3, 6\}$$
$$F_\phi = \{1, 4, 2\}$$



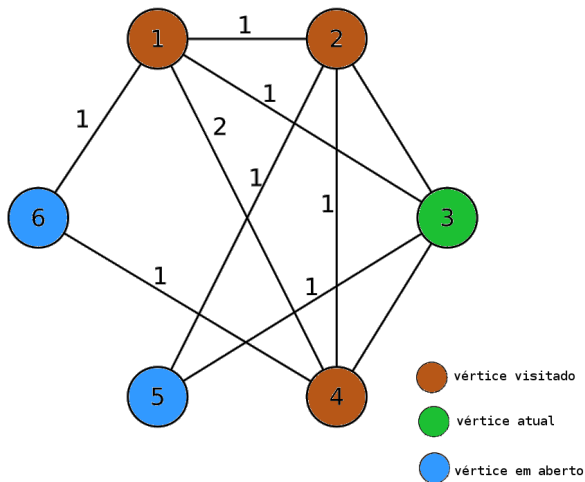
$$Q = \{3, 6, 5\}$$

$$F_\phi = \{1, 4, 2\}$$

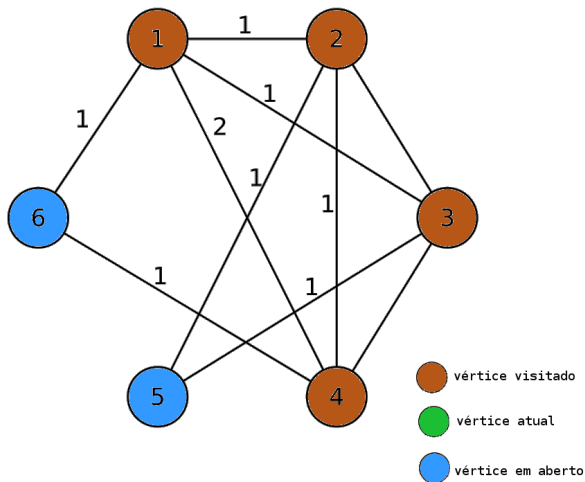


$$Q = \{3, 6, 5\}$$

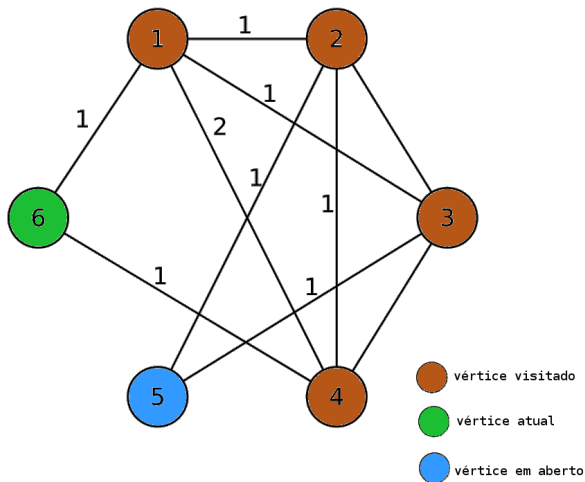
$$F_\phi = \{1, 4, 2\}$$



$$Q = \{6, 5\}$$
$$F_\phi = \{1, 4, 2, 3\}$$

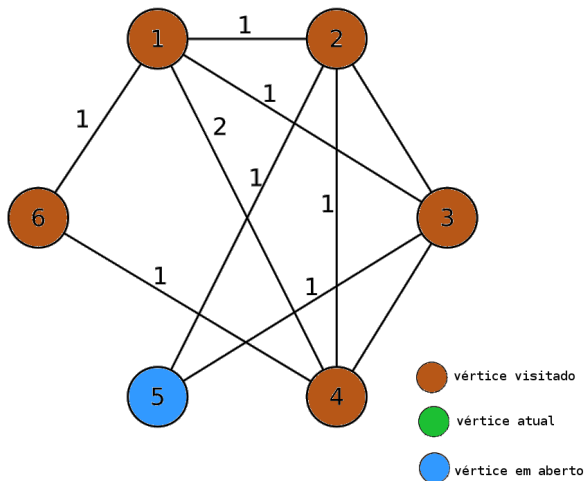


$$Q = \{6, 5\}$$
$$F_\phi = \{1, 4, 2, 3\}$$

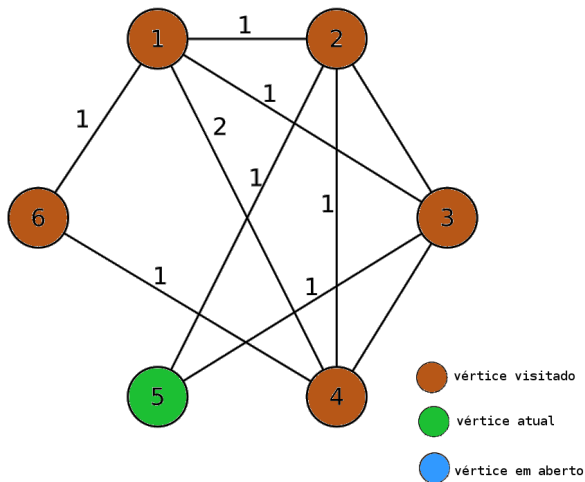


$$Q = \{5\}$$
$$F_\phi = \{1, 4, 2, 3, 6\}$$

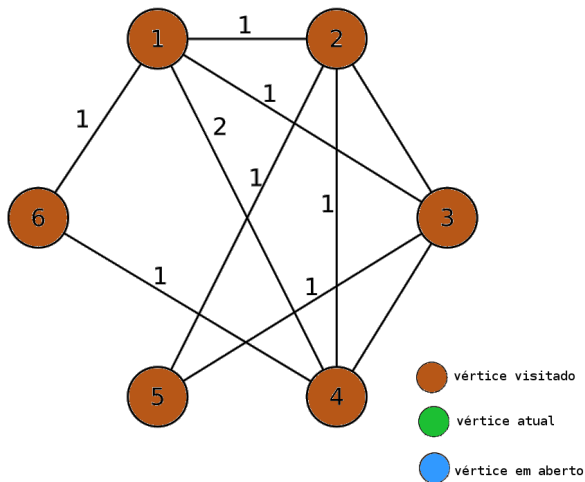




$$Q = \{5\}$$
$$F_\phi = \{1, 4, 2, 3, 6\}$$



$$Q = \{\}$$
$$F_\phi = \{1, 4, 2, 3, 6, 5\}$$



$$Q = \{\}$$
$$F_\phi = \{1, 4, 2, 3, 6, 5\}$$

## Contribuição 2 – Heurística Construtiva

O sequenciamento das tarefas  $\phi$  é obtido a partir de  $F_\phi$ , do seguinte modo:

- Iterativamente, é simulado a disponibilização das ferramentas de  $F_\phi$ , uma a uma;
- A cada iteração, verifica se existem tarefas aptas, isto é, tarefas que possuem todas ferramentas disponibilizadas;
- De acordo com os critérios de desempate, adiciona as tarefas aptas em  $\phi$  da seguinte maneira:
  - Se  $\phi = \emptyset$ , escolha a tarefa com o maior número de ferramentas;
  - Caso contrário, escolha a tarefa que adiciona o menor número de trocas na solução parcial.

## Passo-a-passo

Tabela 5: Matriz binária.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1

Tabela 6: Sequência das Ferramentas

Iteração	Ferramentas Disponíveis
1	[1]
2	[1, 4]
3	[1, 4, 2]
4	[1, 4, 2, 3]
5	[1, 4, 2, 3, 6]
6	[1, 4, 2, 3, 6, 5]

## Passo-a-passo

Tabela 5: Matriz binária.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1

Tabela 6: Sequência das Ferramentas

Iteração	Ferramentas Disponíveis
1	[1]
2	[1, 4]
3	[1, 4, 2]
4	[1, 4, 2, 3]
5	[1, 4, 2, 3, 6]
6	[1, 4, 2, 3, 6, 5]

$$\phi = \{\}$$

## Passo-a-passo

Tabela 5: Matriz binária.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1

Tabela 6: Sequência das Ferramentas

Iteração	Ferramentas Disponíveis
1	[1]
2	[1, 4]
3	[1, 4, 2]
4	[1, 4, 2, 3]
5	[1, 4, 2, 3, 6]
6	[1, 4, 2, 3, 6, 5]

$$\phi = \{\}$$

## Passo-a-passo

Tabela 5: Matriz binária.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1

Tabela 6: Sequência das Ferramentas

Iteração	Ferramentas Disponíveis
1	[1]
2	[1, 4]
3	[1, 4, 2]
4	[1, 4, 2, 3]
5	[1, 4, 2, 3, 6]
6	[1, 4, 2, 3, 6, 5]

$$\phi = \{1\}$$



## Passo-a-passo

Tabela 5: Matriz binária.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1

Tabela 6: Sequência das Ferramentas

Iteração	Ferramentas Disponíveis
1	[1]
2	[1, 4]
3	[1, 4, 2]
4	[1, 4, 2, 3]
5	[1, 4, 2, 3, 6]
6	[1, 4, 2, 3, 6, 5]

$$\phi = \{1, 2\}$$

## Passo-a-passo

Tabela 5: Matriz binária.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1

Tabela 6: Sequência das Ferramentas

Iteração	Ferramentas Disponíveis
1	[1]
2	[1, 4]
3	[1, 4, 2]
4	[1, 4, 2, 3]
5	[1, 4, 2, 3, 6]
6	[1, 4, 2, 3, 6, 5]

$$\phi = \{1, 2, 5\}$$

## Passo-a-passo

Tabela 5: Matriz binária.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1

Tabela 6: Sequência das Ferramentas

Iteração	Ferramentas Disponíveis
1	[1]
2	[1, 4]
3	[1, 4, 2]
4	[1, 4, 2, 3]
5	[1, 4, 2, 3, 6]
6	[1, 4, 2, 3, 6, 5]

$$\phi = \{1, 2, 5, 3\}$$

$$\phi = \{1, 2, 5, 4\}$$

## Passo-a-passo

Tabela 5: Matriz binária.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1

Tabela 6: Sequência das Ferramentas

Iteração	Ferramentas Disponíveis
1	[1]
2	[1, 4]
3	[1, 4, 2]
4	[1, 4, 2, 3]
5	[1, 4, 2, 3, 6]
6	[1, 4, 2, 3, 6, 5]

$$\phi = \{1, 2, 5, 3\} \rightarrow 6 \text{ trocas}$$

$$\phi = \{1, 2, 5, 4\} \rightarrow 7 \text{ trocas}$$

## Passo-a-passo

Tabela 5: Matriz binária.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1

Tabela 6: Sequência das Ferramentas

Iteração	Ferramentas Disponíveis
1	[1]
2	[1, 4]
3	[1, 4, 2]
4	[1, 4, 2, 3]
5	[1, 4, 2, 3, 6]
6	[1, 4, 2, 3, 6, 5]

$$\phi = \{1, 2, 5, 3\} \rightarrow 6 \text{ trocas}$$

$$\cancel{\phi = \{1, 2, 5, 4\}} \rightarrow 7 \text{ trocas}$$

## Passo-a-passo

Tabela 5: Matriz binária.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1

Tabela 6: Sequência das Ferramentas

Iteração	Ferramentas Disponíveis
1	[1]
2	[1, 4]
3	[1, 4, 2]
4	[1, 4, 2, 3]
5	[1, 4, 2, 3, 6]
6	[1, 4, 2, 3, 6, 5]

$$\phi = \{1, 2, 5, 3, 4\} \rightarrow 8 \text{ trocas}$$

## Contribuição 3 – Busca Local de Agrupamento de 1-*blocks*

A partir da definição de 1-*block* (Crama et al., 1994), podemos perceber que:

- Cada 1-*block* representa uma troca de ferramentas que deve ser efetuada.

Com base nessa definição, é proposta uma busca local que objetiva a redução do número de 1-*blocks* em cada linha da matriz  $A^\phi$ , que consiste em:

- Examinar iterativamente a matriz  $A^\phi$  em busca de dois 1-*blocks*  $i$  e  $j$  em uma mesma linha;
- Ao encontrar, tenta-se mover as colunas, uma por vez, de  $i$  para antes ou depois de  $j$ .

## Passo-a-passo

Tabela 7: Matriz binária.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1



## Passo-a-passo

Tabela 8: 9 trocas.

$F \backslash T$	1	2	3	4	5
1	1	1	0	0	1
2	1	0	0	1	0
3	0	1	1	1	0
4	1	0	1	0	1
5	0	0	1	1	0
6	0	0	0	0	1

Tabela 9: 8 trocas.

$F \backslash T$	2	3	4	1	5
1	1	0	0	1	1
2	0	0	1	1	0
3	1	1	1	0	0
4	0	1	0	1	1
5	0	1	1	0	0
6	0	0	0	0	1

Tabela 10: 9 trocas.

$F \backslash T$	2	3	4	5	1
1	1	0	0	1	1
2	0	0	1	0	1
3	1	1	1	0	0
4	0	1	0	1	1
5	0	1	1	0	0
6	0	0	0	1	0

$$\phi = [2, 3, 4, 1, 5]$$

## Passo-a-passo

Tabela 11: 8 trocas.

$F \backslash T$	2	3	4	1	5
1	1	0	0	1	1
2	0	0	1	1	0
3	1	1	1	0	0
4	0	1	0	1	1
5	0	1	1	0	0
6	0	0	0	0	1

Tabela 12: 8 trocas.

$F \backslash T$	3	4	1	2	5
1	0	0	1	1	1
2	0	1	1	0	0
3	1	1	0	1	0
4	1	0	1	0	1
5	1	1	0	0	0
6	0	0	0	0	1

Tabela 13: 8 trocas.

$F \backslash T$	3	4	1	5	2
1	0	0	1	1	1
2	0	1	1	0	0
3	1	1	0	0	1
4	1	0	1	1	0
5	1	1	0	0	0
6	0	0	0	1	0

$$\phi = [3, 4, 1, 5, 2]$$

## Contribuição 4 – Busca Local Iterada

Foi proposta uma implementação da metaheurística Busca Local Iterada (ou *Iterated Local Search* – ILS) (Lourenço et al., 2003) com as seguintes especificações:

- Utiliza-se a sequência obtida pela heurística construtiva como solução inicial;
- O conhecido método *2-opt* é utilizado como mecanismo de perturbação, aceitando qualquer troca;
- Como buscas locais de intensificação são utilizados:
  - Novamente o método *2-opt*<sup>a</sup>; e
  - O Método de Agrupamento de *1-blocks*.

---

<sup>a</sup>Apenas um percentual do espaço de busca é explorado

## Métodos

- Busca Local Iterada (**ILS**) proposta;
- *Clustering Search*, Descida em Vizinhança Variável e Algoritmo Genético de Chaves Aleatórias Viciadas (**CS+BRKGA**, Chaves et al. (2016))<sup>a</sup>;

Visto que ambos os métodos possuem componentes aleatórios, são apresentados os resultados de 20 execuções independentes.

---

<sup>a</sup>Este método é considerado o atual estado-da-arte

## ILS

- Processador Intel Core i5 3.2 GHz ;
- 8 GB RAM;
- Ubuntu 15.10 LTS;
- Códigos escritos em C++, compilado com g++ 4.8.4 e com a opção de otimização -O3.

## CS + BRKGA

- Processador Intel Core i7 3.4 GHz;
- 16 GB RAM.
- Códigos escritos em C++.

## Instâncias

Dois grupos de instâncias foram considerados, totalizando 1510 instâncias.

- Yanasse et al. (2009) ( $A$ ,  $B$ ,  $C$ ,  $D$ ,  $E$ ): 1350 instâncias artificiais;
- Crama et al. (1994) ( $C_1$ ,  $C_2$ ,  $C_3$ ,  $C_4$ ): 160 instâncias artificiais.

## Legenda dos Resultados

São utilizadas a seguir as seguintes nomenclaturas:

- $e \rightarrow$  número de exemplares;
- $S^* \rightarrow$  melhor solução obtida pelos métodos;
- $S \rightarrow$  solução média obtida pelos métodos;
- $OPT \rightarrow$  solução ótima;
- $BKS \rightarrow$  melhor solução conhecida;
- $\sigma \rightarrow$  desvio padrão entre as 20 execuções.

## Instâncias de Yanasse et al. (2009)

Tabela 14: Resultados dos Grupos A, B, C e E de Yanasse et al. (2009)

Conjunto	e	OPT	CS+BRKGA		Método Proposto		
			$S$	$T$	$S$	$T$	$\sigma$
A	340	<b>24,544</b>	<b>24,544</b>	3,71	<b>24,544</b>	0,11	0,000
B	330	<b>25,216</b>	25,218	4,05	25,217	0,18	0,0026
C	340	<b>28,925</b>	29,079	9,83	28,926	1,67	0,0034
E	80	<b>16,888</b>	16,949	6,54	16,890	0,51	0,0051



## Instâncias de Yanasse et al. (2009)

Tabela 15: Resultados do Grupo D de Yanasse et al. (2009)

Conj.	e	CS+BRKGA			Método Proposto			
		$S^*$	$S$	$T$	$S^*$	$S$	$T$	$\sigma$
<i>D</i>	260	24,258	26,252	27,66s	<b>24,016</b>	24,025	6,05s	0,0113

## Instâncias de Crama et al. (1994)

Tabela 16: Resultados dos Grupos  $C_1$  e  $C_2$  de Crama et al. (1994)

Conjunto	e	OPT	CS+BRKGA		Método Proposto		
			$S$	$T$	$S$	$T$	$\sigma$
$C_1$	40	<b>11,175</b>	11,178	2,418	<b>11,175</b>	0,075	0,000
$C_2$	40	<b>22,000</b>	22,073	11,578	22,025	1,033	0,056

## Instâncias de Crama et al. (1994)

Tabela 17: Resultados dos Grupos  $C_3$  e  $C_4$  de Crama et al. (1994)

Conjunto	e	CS+BRKGA		Método Proposto	
		$S^*$	$S$	$S^*$	$S$
$C_3$	40	79,825	80,958	<b>79,575</b>	79,888
$C_4$	40	159,275	161,455	<b>158,300</b>	158,723

Tabela 18: Resultados estatísticos dos Grupo  $C_3$  e  $C_4$  de Crama et al. (1994)

Conjunto	e	CS+BRKGA		Método Proposto	
		$T$	$\sigma$	$T$	$\sigma$
$C_3$	40	123,15s	0,68	175,10s	0,14
$C_4$	40	541,10s	1,22	1097,28s	0,28

## Conclusões

- O método proposto foi capaz de suplantar os melhores resultados conhecidos e em mais de 3 grupos de instâncias novos melhores resultados foram estabelecidos;
- O método proposto obteve um desvio padrão igual a zero em 380 instâncias e para as demais instâncias o desvio padrão foi extremamente baixo, demonstrando robustez;
- O tempo de execução do método aumentou de acordo com o tamanho da instância, porém são valores aceitáveis na prática.
- Trabalhos futuros incluem:
  - Novas estruturas de vizinhança;
  - Utilização de novas técnicas, como paralelismo.

- Daniele Catanzaro, Luis Gouveia, and Martine Labbé. Improved integer linear programming formulations for the job Sequencing and tool Switching Problem. *European Journal of Operational Research*, 2015. ISSN 0377-2217. doi: 10.1016/j.ejor.2015.02.018.
- A. A. Chaves, L. A. N. Lorena, E. L. F. Senne, and M. G. C. Resende. Hybrid method with CS and BRKGA applied to the minimization of tool switches problem. *Computers & Operations Research*, 67:174–183, March 2016. ISSN 0305-0548. doi: 10.1016/j.cor.2015.10.009.
- Yves Crama, Antoon W. J. Kolen, Alwin G. Oerlemans, and Frits C. R. Spieksma. Minimizing the number of tool switches on a flexible machine. *International Journal of Flexible Manufacturing Systems*, 6(1):33–54, January 1994. ISSN 0920-6299, 1572-9370. doi: 10.1007/BF01324874.
- Helena R Lourenço, Olivier C Martin, and Thomas Stützle. *Iterated local search*. Springer, 2003.
- Horacio Hideki Yanasse, Rita de Cássia Meneses Rodrigues, and Edson Luiz França Senne. Um algoritmo enumerativo baseado em ordenamento parcial para resolução do problema de minimização de trocas de ferramentas. *Gestão & Produção*, 16(3):370–381, 2009. ISSN 0104-530X. doi: 10.1590/S0104-530X2009000300005.

# Agradecimentos



# Agradecimentos



## Instâncias de Catanzaro et al. (2015)

Tabela 19: Resultados dos Grupos A, B, C e D de Catanzaro et al. (2015)

Conjunto	e	BKS <sup>a</sup>	Método Proposto			
			$S^*$	$S$	$T$	$\sigma$
<i>datA</i>	40	<b>10,850</b>	<b>10,850</b>	10,850	0,068	0,000
<i>datB</i>	40	21,975	<b>21,775</b>	21,778	1,005	0,006
<i>datC</i>	40	77,725	<b>74,850</b>	75,065	174,286	0,140
<i>datD</i>	40	163,900	<b>158,325</b>	158,810	1052,259	0,246

<sup>a</sup>Melhor solução conhecida