A New Local Search Procedure for the Minimization of Tool Switches

Gustavo Silva Paiva
Marco Antonio M. Carvalho

Departamento de Computação
Instituto de Ciências Exatas e Biológicas
Universidade Federal de Ouro Preto

March 22, 2016

Conteúdo

Introduction

- Flexible Manufacturing Systems (FMS) are known for their great flexibility and adaptability on production planning
  - It allows a wide range of item types to be produced;
  - e.g., the same machinery present on a production line can be quickly adjusted to produce an unrelated new set of items.
  - This characteristic is also important when the production must the replanned.

Introduction

- A common type of FMS, mostly employed by the automotive and metallurgical industries, uses **flexible machines**
  - Machines able to perform different types of operations requiring only a few changes on their configuration;
  - Cutting, drilling, etc.

A New Local Search Procedure for the MTSP

Introduction

- Each item requires a set of such tools to be produced and the magazine is able to store all tools for a single item
  - All available tools cannot be loaded at the same time on the magazine;
- Between the production of two different item types it may be neccessary to switch tools on the magazine;
- The tool switching imply the interruption of the production line as the machine should be turned off in order to be configured
  - Thus increasing the cost of production.

Introduction

- From a predetermined demand for products, it becomes neccessary to determine a production plan such that the production line meets this demand;
- The plan is divided into jobs and aims to minimize the downtime of the production machines, in order to maximize productivity and reduce related costs;

A New Local Search Procedure for the MTSP

Introduction

- The production plan consists of two parts:
  1. Determine the order in which the jobs are processed; and
  2. Decide when to switch tools and what tools will be switched.
- The first one is the **Minimization of Tool Switches Problem**;
- The second one is deterministic polynomial time solvable, using the *Keep Tools Needed Soonest* algorithm.

A New Local Search Procedure for the MTSP

Problem Data

- The set of jobs $T = \{1, \ldots, n\}$;
- The of set tools $F = \{1, \ldots, m\}$;
- The subset of tools T_j needed to process job $j \in T$;
- The magazine capacity C.

A New Local Search Procedure for the MTSP

Formal Definition

- The *Minimization of Tool Switches Problem* (MTSP) is defined on a binary sparse matrix M:
  - The n rows correspond to the tools and the m columns correspond to each job;
  - Entry $m_{ij} = 1$ if tool i is loaded in the machine during job j processing and $m_{ij} = 0$ otherwise.
- A solution is represented by a permutation π of the elements of T and a tool switching plan, resulting in M_π;
- The number of inversions in M_π is the number of tool switches.

An Example

Jobs	1	2	3	4	5
Tools	1	1	3	2	1
	2	3	4	3	4
	4	5	7	5	6
	7			6	
Capacity = 4					

A Solution Example

π	1	3	5	2	4
Tools	1	$\underline{1}$	1	1	0
	1	0	0	0	1
	0	1	1	1	1
	1	1	$\underline{1}$	0	0
	0	0	0	1	1
	0	0	1	$\underline{1}$	1
	1	1	0	0	0

- Underlined tools denote they are loaded in advance;
- 8 tool switches (the initial four also counts).

A New Local Search Procedure for the MTSP

Motivation

- It is NP-hard;
- Its practical application in VLSI design industry
  - Printed circuit board assembly;
  - The assembly tasks are jobs and components are tools.

A New Local Search Procedure for the MTSP

Proposed Methods

- A graph search for generating an initial solution;
- A new local search method;
- An Iterated Local Search.

Current State of the Art

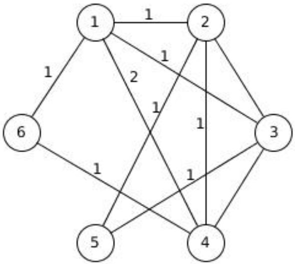
- Metaheuristic: Clustering Search + Variable Neighborhood Search + Biased Random Key Genetic Algorithm (2016);
- Exact: Branch & Bound (2009), Integer Linear Programming (2015).

A New Local Search Procedure for the MTSP

Graph Model

- Nodes represent tools;
- Edges connects tools used in a same job;
- No loops or parallel edges;
- Weights are defined by the frequency of tools;
- Each job induces a clique.

Jobs	1	2	3	4	5
Tools	1	1	3	2	1
	2	3	4	3	4
	4			5	6



A New Local Search Procedure for the MTSP

Graph Search

- The main idea is to search for tools that must be kept together in the solution, avoiding switches;
- Breadth-first search (BFS) in non-decreasing weight of edges order.

Jobs Sequencing

- The BFS returns a list of tools;
- The jobs sequence is determined in a greedy fashion:
  - The list of tools is traversed, and once all tools of a specific job are found, it is sequenced.
- Two tie breakers:
  1. Higher number of tools (if no job sequenced yet);
  2. Lower number of switches.

A New Local Search Procedure for the MTSP

Local Search

- Each consecutive pair of 1-blocks in a same row denotes an inversion
  - Minimizing 1-blocks number is a different problem.
- 1-blocks columns are inserted before/after the other 1-blocks;
- The best move is performed.

A New Local Search Procedure for the MTSP

Iterated Local Search

- Classic version;
- Local search – only non deteriorating movements;
- Perturbation: 2-opt of columns
  - Randomly selected, 20% of the columns.
- All parameters tuned using the irace package.

Methods

- Iterated Local Search (ILS);
- Biased Random Key Genetic Algorithm + VND + Clustering Search (BRKGA + CS).

A New Local Search Procedure for the MTSP

ILS

- Intel Core i5 3.2 GHz processor;
- 8 GB RAM;
- Ubuntu 15.10 LTS;
- Codes written in C++, compiled with gcc 4.8.4 and the -O3 optimization option.

BRKGA + CS

- Intel Core i7 3.4 GHz processor;
- 16 GB RAM;

A New Local Search Procedure for the MTSP

Instances

Two different data sets from the literature were considered, a total of 1670 instances.

- Yanasse (A, B, C, D, E): 1510 artificial instances;
- Crama (C_1, C_2, C_3, C_4): 160 artificial instances.

A New Local Search Procedure for the MTSP

Average gap from Optimal Solutions (20 runs)

Method	A	B	C	D[a]	E
ILS	0.00%	0.00%	0.00%	0.00%	0.00%
BRKGA + CS	0.00%	0.00%	0.00%	0.05%	0.00%

[a] Considering only known optimal solutions.

A New Local Search Procedure for the MTSP

Average Running Times (20 runs)

Method	A	B	C	D	E
ILS	0.15s	0.26s	2.14s	6.45s	0.70s
BRKGA + CS	3.71s	4.05s	9.83s	27.66s	6.54s

A New Local Search Procedure for the MTSP

Average gap from Best Known Results (20 runs)

Method	C_1	C_2	C_3	C_4
ILS	0.00%	0.00%	0.031%	0.00%
BRKGA + CS	0.00%	0.00%	0.00%	0.34%

A New Local Search Procedure for the MTSP

Average Running Times (20 runs)

Method	C_1	C_2	C_3	C_4
ILS	0.09s	1.11s	168.71s	1063.47s
BRKGA + CS	2.42s	11.58s	123.15s	541.10s

Conclusion

- The proposed ILS was able to match a large part of the best results available on the literature and to improve some of them;
- Running time increased on large instances;
- Future work includes:
  - TSP model + solver;
  - New formulations;
  - New lower bounds.

Acknowledgments

This research has been supported by the National Council for Scientific and Technological Development (CNPq) and by Universidade Federal de Ouro Preto (UFOP).

Questions?

рахмат
danke
謝謝
spas
ngiyabonga
Баярлалаа
mersi
kia ora
barka
welalin
tack
teşekkür ederim
mahalo
tapadh leat
спасибо
faafetai lava
vinaka
спасибі
blagodaram
dank je
misaotra
matondo
paldies
grazzi
хвала
kiitos
dankie
nanni
nandri
dhanyavad
maururu
thank you
gracias
asante
manana
obrigada
tenki
enkosi
bedankt
bayarlalaa
gracie
hvala
koszonom
dankon
aciu
akun
djiere dieuf
tau
mochchakkeram
murakoze
mamnun
дякую
dziękuje
chnorakaloutioun
gratias ago
gracies
sulpay
taiku
go raibh maith agat
chokrane
sobodi
dėkuji
sagolun
sukriya
kop khun krap
obrigado
mèsi
didi madloba
ありがとう
tanemirt
grazie
arigatô
takk
dakujem
trugarez
najis tuke
kam sah hamnida
rahmat
terima kasih
rahmet
ευχαριστώ
diolch
dhanyavadagalu
shukriya
merce
мерси
তোমাকে ধন্যবাদ
감사합니다
xiexie
merci