Optimizing the Scheduling of a Pick and Place Robotic System

Tommaso Schettini

tommaso.schettini@polimi.it

POLITECNICO
MILANO 1863

Master Thesis Supervised by: Prof F. Malucelli
and Prof H. R. Loureço

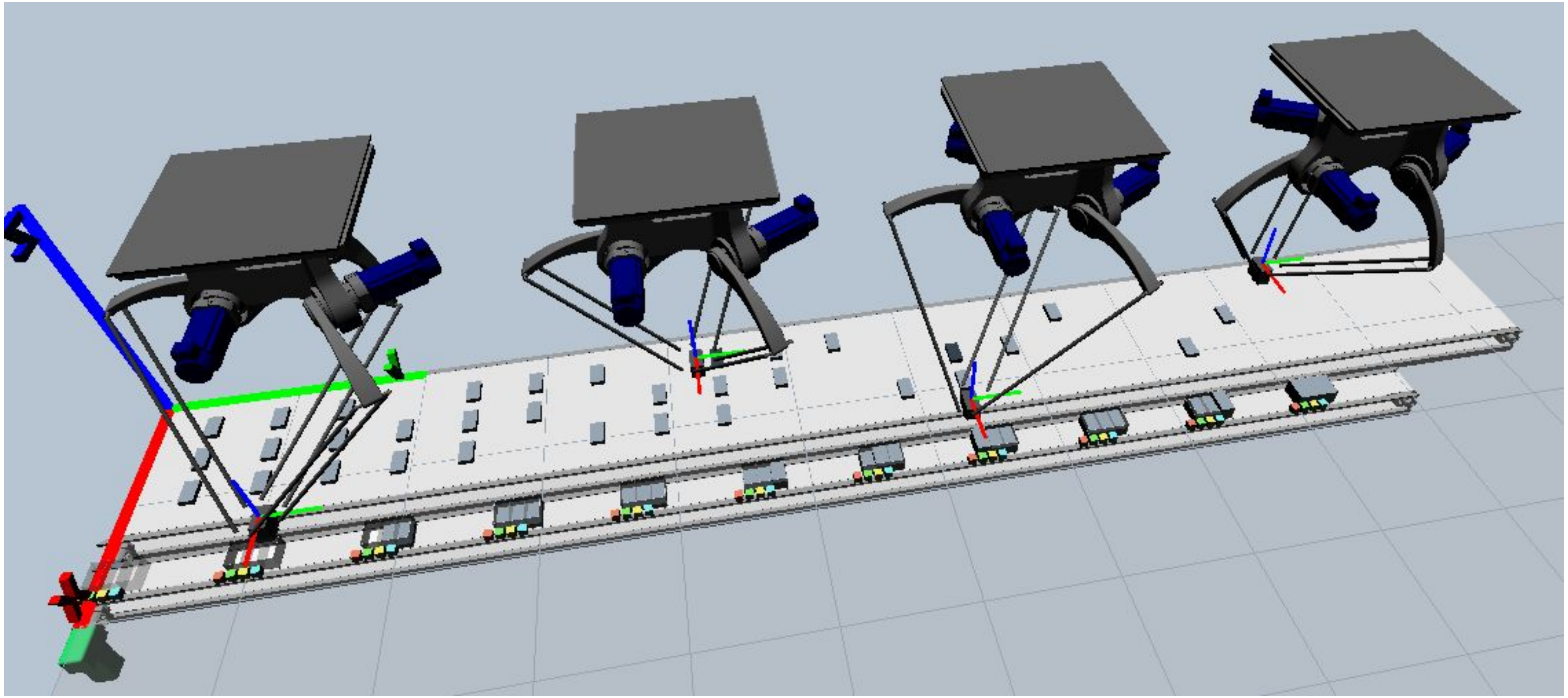
30/11/2017 - Politecnico di Milano

Optimizing the Scheduling of a Pick and Place Robotic System

- Introduction
- Exact Methods
- Metaheuristic Methods
- Results

Introduction

The Pick and Place Scheduling Problem



Exact Methods

Exact Methods

Basic Formulation

Decision Variables

- $x^r_{p,b}$ Assignment of the missions performed by the robots in the scheduling
- $\Pi^r_{p,b}$ Starting time of the missions performed in the scheduling
- λ_p θ_b $y^r_{p,b}$ Auxiliary Variables

$$min \sum_p \lambda_p$$

$S.T.$

$$a_p^r \leq \Pi_{p,b}^r \leq b_p^r - \delta(1 - x_{p,b}^r) + M(1 - y_{p,b}^r) \qquad \forall p, b, r$$

$$\alpha_b^r \leq \Pi_{p,b}^r \leq \beta_b^r - \delta(1 - x_{p,b}^r) + M(1 - y_{p,b}^r) \qquad \forall p, b, r$$

$$y_{p,b}^r + y_{p',b'}^r \leq 1 \qquad \forall p, b, r, p' > p, b' \leq b$$

$$\Pi_{p,b}^r + \delta y_{p,b}^r \leq \Pi_{p',b'}^r + M(1 - y_{p,b}^r) \qquad \forall p, b, r, p' > p, b' \geq b$$

$$\sum_b \sum_r x_{p,b}^r = sP_p - \lambda_p \qquad \forall p$$

$$\sum_p \sum_r x_{p,b}^r = sB_b - \theta_b \qquad \forall b$$

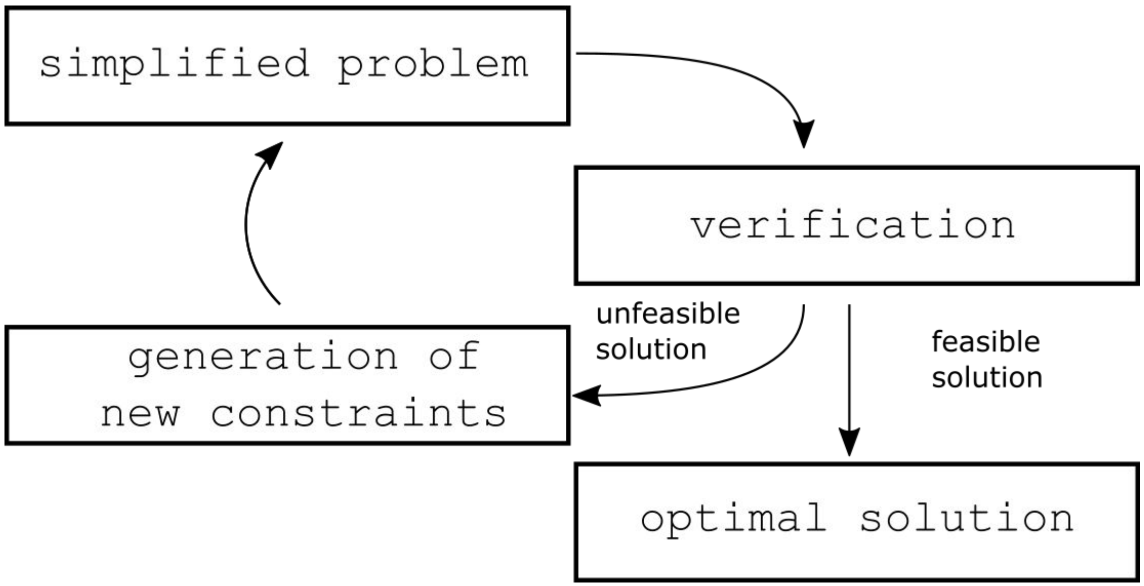
$$sP_p * x_{p,b}^r \geq y_{p,b}^r \qquad \forall p, b, r$$

$$x, \lambda, \theta \in N_0^+ \qquad y \quad binary$$

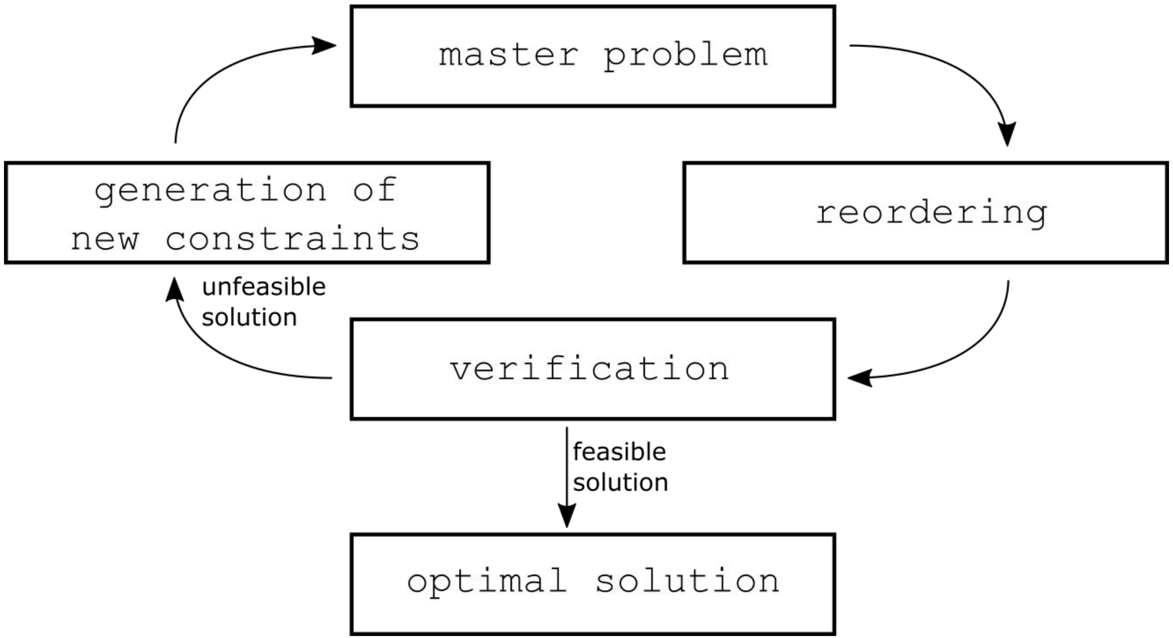
Exact Methods

Row Generation

General Scheme



Implemented Scheme



Metaheuristic Methods

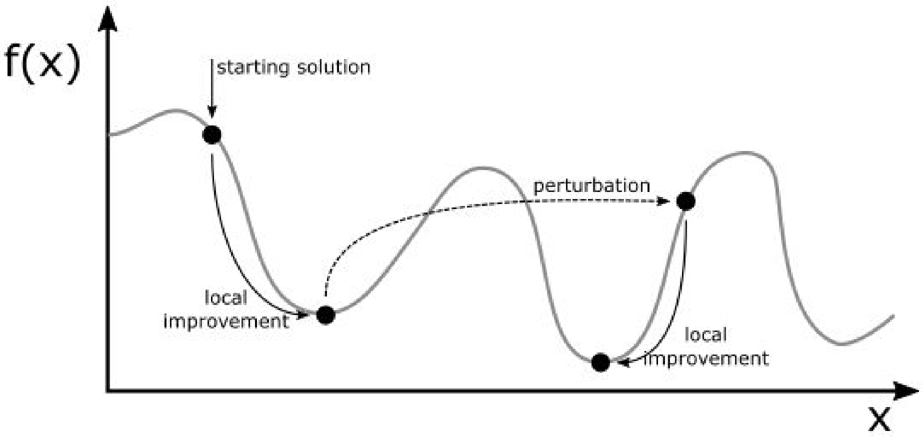
Metaheuristic Methods

Iterated Local Search

Main Idea

Alternate between **local search** and **perturbation** phases.

The idea is to move from one local optima to another.



Results

Testing Instances

The testing of the methods was performed on a number of artificially generated testing instances:

- Start-up conditions
- 3 Testing Configurations
- 11 Instances per configuration

Results

Exact Methods

Fixed Speed - Solution Time (sec)

	Configuration 0	Configuration 1	Configuration 2
45 pcs	0.9	0.63	3.6
90 pcs	6.4	10	90.6
180 pcs	115.7	332.9	1414.2

AMPL implementation on a 3.30GHz Intel Xeon CPU E31245

Controllable Speed - Solution Time (sec)

	Configuration 0	Configuration 1	Configuration 2
45 pcs	1.8	1.3	2.9
90 pcs	39	10.6	90.8
180 pcs	3121.7	196.1	3600*

AMPL implementation on a 3.30GHz Intel Xeon CPU E31245

Pieces Lost - 180 pcs

	Configuration 0	Configuration 1	Configuration 2
Fixed speed	4	36.5	22.3
Controllable speed	0	17.3	0

AMPL implementation on a 3.30GHz Intel Xeon CPU E31245

Results

Metaheuristic Methods

Fixed Speed - 180 pcs

	Configuration 0	Configuration 1	Configuration 2
Pieces lost	5.5	37.5	24.9
Opt*	4	36.5	22.3
Time (s)	0.6	0.4	1

C++ implementation on a 2.40GHz Intel Core i7-3630QM CPU

Controllable Speed - 180 pcs

	Configuration 0	Configuration 1	Configuration 2
Pieces lost	0	18.7	0
Opt*	0	18.2	0
Time (s)	0.02	0.2	0.07

C++ implementation on a 2.40GHz Intel Core i7-3630QM CPU

Many thanks for the attention

Questions?

Glossary I

The pieces enter the system through the “pickup” conveyor.

The pieces entering the system do so in columns, typically with a constant number of pieces. Piece *p* will refer to an individual piece in the p-th column on the pickup conveyor.

The containers enter the system through the “place” conveyor.

The elementary operations of the robots are called “missions”. The missions will be described using the notation:

$$(p, b, r)$$

Glossary II

The set	P = {1..np}	is the set of the columns of pieces entering the system.
The set	B = {1..nb}	is the set of the containers entering the system.
The set	Ro = {1..nr}	is the set of the robots acting on the system.

Glossary III

To define the solutions of the problem we will identify as a “scheduling” the list of the missions to be performed by each robot.

To fully define a solution of the system the scheduling has to be coupled with the starting time of each mission and the speed profile for the “place” conveyor.

Parameters I

- δ is the minimum time between the start of two consecutive missions of the same robot
- a_p^r is the first time instant a mission involving a piece on column p ∈ P can be started by the robot r ∈ Ro.
- b_p^r is the last time instant a mission involving the a piece on column p ∈ P can be started by the robot r ∈ Ro.
- α_b^r is the time first instant a mission involving the container b ∈ B can be started by the robot r ∈ Ro.
- β_b^r is the last time instant a mission involving the container b ∈ B can be started by the robot r ∈ Ro.

Parameters II

- sB_b is the number of pieces in the column p ∈ P at the beginning of the scheduling.
- sP_p is the number of slots in the container b ∈ B that are empty at the beginning of the scheduling.
- M is an arbitrary large number