



HAL
open science

A MILP formulation for multi-robot pick-and-place scheduling

Cyrille Briand

► **To cite this version:**

Cyrille Briand. A MILP formulation for multi-robot pick-and-place scheduling. 16th International Conference on Project Management and Scheduling (PMS 2018), Apr 2018, Roma, Italy. pp.58–61. hal-04250992

HAL Id: hal-04250992

<https://hal.science/hal-04250992>

Submitted on 20 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A MILP formulation for multi-robot pick-and-place scheduling

Briand C.¹, Ndiaye J.C.¹ and Parlouar R.²

¹ LAAS-CNRS, Université de Toulouse, UPS, Toulouse, France
email: briand@laas.fr, ndiaye@laas.fr

² NOVALYNX, 35 Bis Route de Bessières, 31240 L'Union, France
email: remi.parlouar@novalynx.fr

Keywords: Assignment and Scheduling, Mixed Integer Linear Programming, Multi-robot pick-and-place.

1 Introduction

The 21st century is marked by the fourth industrial revolution, which embraces many technologies and concepts. Among them, robotization is often viewed as the most promising avenue of progress in the field of automated production. Indeed the use of more and more sophisticated machines and robots is able to bring improvements in production costs, rates, quality and operators safety.

Among the various types of robots frequently used in production systems are the handling robots, which basically pick parts somewhere in the shop and place them elsewhere. More specifically, this study focuses on automated packaging systems involving several handling robots. A packaging system is generally composed of two conveyor belts conveying products and boxes, respectively. A handling robot picks one or several products on the former conveyor and places them in a box on the latter. The conveyor belts may have several possible shapes: parallel, perpendicular or circular. The parallel one is the most common and product and box flows can go in the same or opposite direction, as illustrated in Figure 1 (taken from (Blanco Rendon 2013)). A pick/place task can only be carried out by one robot when the corresponding product/box is present inside the working area of this robot. The normal speed of each conveyor being assumed known, a time window can be associated with each robot task. Moreover, the pick/place task duration can vary and depends either on the product or the robot.

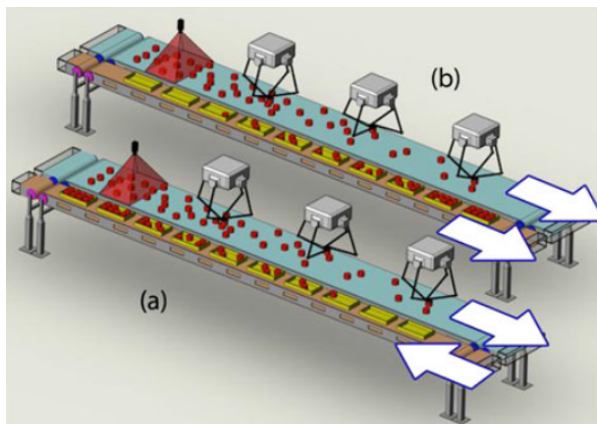


Fig. 1. Conveyor belts configurations

The problem considered in this paper, further referred to as the Multi-robot-Pick-and-Place Scheduling Problem (MPPSP), consists in i) assigning products and boxes to robots and ii) defining a *consistent* starting time for each pick/place task, so that the filling rate is maximized (or equivalently, the number of filled boxes over a given time horizon is maximized). In the case of a pick-task, the starting-time consistency only requires that the task is performed by the robot during its time execution window (i.e., when the product is present inside the robot workspace). In the case of a place-task, the previous condition should obviously hold *and*, additionally, there are flow constraints: if k products should be placed inside each box in a one-shot operation, one has to ensure that k pick tasks have been achieved before the place task can be carried out. Finally, note that in the case the conveyor speed can be controlled (which is assumed impossible in the present study), the filling rate can be further improved, which gives rise to a third MPPSP dimension consisting in the determination of the optimal conveyor speed profiles.

In many existing systems, a vision system is integrated in front of the conveyor entries to locate the various parts, which allows predicting the working-area entry or exit events a few seconds before their occurrence. Moreover, in the context of the industry 4.0, all information about production and packaging processes may be known in advance so that execution windows of pick/place tasks could be either predicted earlier. Under the assumption of predictability of the product/box flows, the MPPSP is studied in its offline version in this paper and a compact Mixed-Integer Linear Programming (MILP) formulation is proposed.

The paper is structured as follows. First, a brief literature overview is made that particularly put into evidence some relationships between MPPSP and some other well-known problems of the scheduling literature. Then, our MILP formulation is established that takes benefits from specific dominance rules, which allows characterizing all the dominant solutions on a robot within a single *master-sequence*. Some conclusions are drawn in the last section.

2 Literature overview

A vast majority of the paper of the literature tackles the online version of the problem, taking interest in designing efficient rules or cooperation mechanisms between robots that maximize the filling rate, while balancing the working load between robots, e.g., (Blanco Rendon 2013, Bouchrit 2016, Huang *et al.* 2015, Humbert *et al.* 2015). In the OR literature, Daoud *et al.* (2014) took interest in designing pick-and-place robotic systems and propose fast metaheuristics to determine the best schedule rule to be applied to each robot.

For the offline version of MPPSP, the literature is scarcer. In (A. Bouchrit 2016), a network-based MILP formulation is proposed to solve the offline MPPSP in the case of a homogeneous product/box flow (each product/box is separated from the next one on the conveyor by a constant distance). Products are considered as nodes within a network and the problem amounts to find for each robot the best path to collect the maximum possible number of products, which gives a pick-and-place task sequence. Many constraints are taken into account such as conveyor belt velocities, robot load balance, time windows and flow constraints. Nevertheless, the implementation of this formulation on commercial solver does not provide satisfying performances as finding optimal solution turns out to be too time-demanding.

In the scheduling literature, MPPSP is sharing some similarities with the parallel machine problem with time windows that aims at minimizing the number of tardy jobs, (denoted as $P|r_j|\sum U_j$ in (Pinedo 2008)). This problem is known to be NP-Hard in the strong sense even for one single machine. Nevertheless, still under the assumption of a single machine environment, it is polynomially solvable when execution windows have a

staircase structure. As a specific feature of MPPSP, we observe that there are several possible time windows for the execution of a task (depending on the robot implementing it), which tends to indicate that MPPSP is also related to the Runway Scheduling Problem (RSP) (Artiouchine *et. al.* 2008) that consists in sequencing aircraft landing. Note that RSP is also NP-hard.

3 MILP formulation

This chapter takes an interest in finding a job sequence that maximizes the number of filled boxes assuming the product and box flows predictable. We consider three sets \mathcal{B} , \mathcal{P} and \mathcal{R} of B boxes, P products and R robots, respectively. In the notations used below, index b (p and r , respectively) refers to a box $b \in \mathcal{B}$ (a product $p \in \mathcal{P}$ and a robot $r \in \mathcal{R}$, respectively). The pick and place processing times are denoted D_{pr} and D_{br} , which depend on robot r . We refer to $[S_{pr}, F_{pr}]$ and $[S_{br}, F_{br}]$ as the execution windows of product p (box b , respectively) on robot r .

In the remainder of this paper, as the conveyor speed is assumed constant, we set $F_{pr} - S_{pr} = \Delta_{pr}$, $\forall (p, r) \in \mathcal{P} \times \mathcal{R}$ ($F_{br} - S_{br} = \Delta_{br}$, $\forall (b, r) \in \mathcal{B} \times \mathcal{R}$, resp.). Moreover, without loss of generality, we assume that $\Delta_{pr} > \Delta_{br}$ (products stay longer in the robot working area than boxes) but, as explained below, it could be the reverse.

Once an assignment of products and boxes to robots is decided (note that a product/box can possibly not be assigned), the problem left is to find a pick-and-place sequence on each robot that i) is time feasible and ii) respects the constraint that k picks should always precede any *place* operation. For ensuring time feasibility, following the idea proposed by Briand and Ourari (2013), a *master sequence* can be considered that characterizes a set of dominant sequences. This master sequence uses the notion of a *top-job*, i.e. a job such that its execution window does not (strictly) include the execution window of any other job. In our case, as there are only two kinds of time intervals (the pick and place ones) and because $\Delta_{pr} > \Delta_{br}$, any place operation is a top job. Therefore, a master sequence Θ_r having the form below can be defined for each robot r .

$$\Theta_r = \sigma_{1r}^- \quad 1 \quad \sigma_{1r}^+ \quad \sigma_{12} \quad \sigma_{2r}^- \quad 2 \quad \sigma_{2r}^+ \cdots i-1 \quad \underbrace{\sigma_{i-1r}^+ \quad \sigma_{i-1,i} \quad \sigma_{ir}^+}_{\theta_{i-1r}} \quad i \quad \sigma_{ir}^+ \cdots$$

Each place task i has two sets σ_{ir}^- and σ_{ir}^+ of pick tasks at its left and its right, respectively. More specifically, σ_{i-1r}^+ represents products which intervals overlap place interval $i-1$ but not place interval i . Similarly, σ_{ir}^- gathers pick tasks such that their intervals overlap box interval i but not box interval $i-1$. Eventually, $\sigma_{i-1,i}$ gathers product intervals which overlap both box intervals $i-1$ and i . We refer to θ_{i-1r} as the subset of pick tasks located between place task $i-1$ and i , with θ_{0r} (θ_{Br}) the subset located at the left (the right) of box 1 (of box B , resp.). Note that the same pick task can belong to several sets θ and one has to decide whether the task is performed and, if it is performed, in which set θ . One advantage of a master sequence lies in the fact that, once the previous decisions made, the time feasibility of the resulting pick-and-place sequence can easily be assessed.

The following formulation takes benefit of the master sequence notion and introduces the following binary variables. A box b is filled by robot r if binary variable $y_{br} = 1$ (0 otherwise). A product p is picked in subset θ_{br} if $x_{bpr} = 1$.

$$\max z = \sum_b \sum_r y_{br}$$

$$\sum_b \sum_{p \in \theta_{br}} x_{bpr} \leq 1 \quad , \quad \forall p \quad \forall r \quad (1)$$

$$\sum_r y_{br} \leq 1 \quad , \quad \forall b \quad (2)$$

$$ky_{br} \leq -k \sum_{i < b} y_{ir} + \sum_{i < b} \sum_{p \in \theta_{ir}} x_{pir} \leq k \quad , \quad \forall b \quad \forall r \quad (3)$$

$$\text{The master sequence } \Theta_r \text{ is time feasible} \quad , \quad \forall r \quad (4)$$

$$\begin{aligned} x_{bpr} &\in \{0, 1\} \quad , \quad \forall b \quad \forall p \quad \forall r \\ y_{br} &\in \{0, 1\} \quad , \quad \forall p \quad \forall r \end{aligned}$$

The formulation aims at maximizing the number of filled boxes. Constraints (1-2) enforce any product/box to be picked/filled once at the most. Constraints (3) aim at satisfying the (flow) constraint, i.e. k product at the most should be picked before any place operation. As in (C. Briand and S. Ourari 2013), high level constraints (4) can be implemented using a set of big-M linear constraints (not stated here for matter of conciseness) that use integer variables s_{br} and f_{br} . These variables refer to as the earliest starting time and the latest finishing time, respectively, of place task b on robot r (this value linearly depending on the values of other binary variables), provided that $s_{br} + D_{br} \leq f_{br}$.

4 Conclusion

This paper sketches a formulation for solving the offline MPPSP. This formulation has been tested and validated using some academic instances. A more systematic experimental study is currently in progress to assess the efficiency of our approach. The special case where the processing times of the pick/place tasks are identical (i.e., $D_{pr} = D_{pick_r}$ and $D_{br} = D_{place_r}$) will also be considered.

References

- Artiouchine K., P. Baptiste, C. Durr, 2008, "Runway Sequencing with Holding Patterns", *European Journal of Operational Research*, Vol. 189(3), pp.1254-1266.
- Blanco Rendon D.P., 2013, "Modelling and Simulation of a Scheduling Algorithm for a Pick-and-Place Packaging System", MastersThesis, Polytechnic of Milan.
- Bouchrit A., 2016, "Optimal Scheduling for Robotized Pick and Place Packaging Systems", MastersThesis, Polytechnic of Milan.
- Briand C. and S. Ourari, , 2013, "Minimizing the number of tardy jobs for the single machine scheduling problem: MIP-based lower and upper bounds", *RAIRO - Operations Research*, Vol. 47, pp. 33-46.
- Daoud S., H. Chehade, F. Yalaoui and L. Amodeo, 2014, "Efficient metaheuristics for pick and place robotic systems optimization", *Journal of Intelligent Manufacturing*, Volume 25, pp. 27-41.
- Huang Y., R. Chiba, T. Arai, T. Ueyama, J. Ota, 2015, "Robust multi-robot coordination in pick-and-place tasks based on part-dispatching rules", *Robotics and Autonomous Systems*, Volume 64, 2015, pp. 70-83.
- Humbert G., M.T. Pham, X. Brun, M. Guillemot and D. Noterman, 2015, "Comparative analysis of pick & place strategies for a multi-robot application", *Proc. IEEE 20th Conference on Emerging Technologies and Factory Automation (ETFA)*, Luxembourg.
- Pinedo, M.L., 2008, "Scheduling: Theory, Algorithm and Systems.", 3rd Edition, Springer-Verlag, New York.