HAL open science

Assembly Line Performance Analysis Based on Aircraft Preliminary Design: a Scheduling Approach

Anouck Chan, Stéphanie Roussel, Thomas Polacsek

To cite this version:

Anouck Chan, Stéphanie Roussel, Thomas Polacsek. Assembly Line Performance Analysis Based on Aircraft Preliminary Design: a Scheduling Approach. 18th International Workdhop on Project Managment and Scheduling PMS 2022, Apr 2022, Gand, Belgium. hal-03771512

HAL Id: hal-03771512

https://hal.science/hal-03771512v1

Submitted on 20 Sep 2022

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.

Assembly Line Performance Analysis Based on Aircraft Preliminary Design: a Scheduling Approach

Anouck Chan[1], Stéphanie Roussel[1] and Thomas Polacsek[1]

ONERA/DTIS, Université de Toulouse, F-31055 Toulouse, France
{anouck.chan,stephanie.roussel,thomas.polacsek}@onera.fr

Keywords: aircraft design, assembly line, RCPSP.

1 Introduction

Conceiving and building a complex product, such as an aircraft, is not only about its specifications but also about how it is built. Because of its complexity, each aircraft has a dedicated industrial system. Thus, when a new aircraft is conceived, so is its associated industrial system. Both designs are interdependent, and design choice of one may impact the design, the performance and the construction of the other.

In the aeronautical domain, the current development cycle is mainly sequential: the aircraft is conceived first, then its means of production is. As described in Polacsek et al. (2017), such an approach raises several issues, and, following the industry 4.0 trend, many recent works follow a simultaneous engineering approach in which both systems (the aircraft and its industrial system) are designed simultaneously. Among others, one challenge raised by such an approach is to allow aircraft designers to estimate the consequences of their choices at different stages of the process with respect to the industrial system.

Recent works address this problem. In Sanchidrián (2019), the author lists several publications that focus on conceiving aircraft manufacturing systems at different stages of the development process. Among those references, in Pralet et al. (2018), the authors evaluate aircraft designs with respect to an existing assembly line.

In this work, we follow the latter approach in order to allow aircraft designers to compare aircraft designs with respect to their assembly line performances. However, we are interested in earlier steps of the development process. We propose a Constraint Programming model of the industrial system at early stages of the process, along with its performance features, and a model of the aircraft designs in terms of high level assembly tasks (Section 2). Then, we show the results associated with two real industrial aircraft designs (Section 3). Finally, in Section 4, we briefly conclude and describe future works.

2 Problem model

2.1 Assembly line presentation

We first describe the main features of an assembly line. We focus on pulsed lines, which are composed of several *workstations*. The aircraft moves from a station to the next at a regular time interval called *takt-time*. Therefore, the total time spent by the aircraft in the assembly line, or *makespan*, is equal to takt-time multiplied by the total number of stations. An assembly line is efficient if it has a short takt-time and a small number of stations (*i.e.* a short makespan). A set of actions (or *activities*) are performed on each station by technicians. These activities are located in specific areas of the aircraft (or *zones*) in which only a maximum number of people can work simultaneously.

In early design stages, the assembly line does not yet exist. More precisely, the number of stations, the takt-time and the makespan are not known. In this work, we focus on the best takt-time and makespan (the smallest ones) that can be reached for a given design.

We also have a limited knowledge about tools and machines that will be available on the assembly line. Indeed, we are only aware of the existence of *dimensioning resources*, *i.e.* resources in limited supply because of their size, their high price or their uniqueness. We also consider possible *incompatibilities* between resources, meaning that some resources cannot belong to the same station.

Note that in this work, as done in Pralet et al. (2018), we do not directly address aircraft architecture design, but consider instead a set of assembly operations that allow to produce the aircraft.

2.2 Constraint Programming approach

In this part, we describe the modelling of the problem, which could be seen as a kind of Resource-Constrained Project Scheduling Problem (RCPSP) extended with resource incompatibilities, neutralisation, hard and simple precedence relations and for which we consider specific criteria.

The assembly line is composed of a set of dimensioning *resources*, denoted $\mathcal{R}$. Each resource $r \in \mathcal{R}$ has a capacity cap_r that represents the number of tasks that can performed simultaneously by r. Then, the aircraft is divided in several *zones*, denoted $\mathcal{Z}$. A *zone* $z \in \mathcal{Z}$ is a physical area of the aircraft. It has a *capacity* cap_z that represents the maximum number of technicians who can work in the zone at the same time.

$\mathcal{A}$ is the set of assembly operations of the aircraft. Each activity $a \in \mathcal{A}$ has a duration d_a and occupy one or more zones of the aircraft. For each activity $a \in \mathcal{A}$ and each zone $z \in \mathcal{Z}$, occ_a^z is the number of technicians required for performing a in z. An activity may *neutralise* a set of zones $neutr_a$. For example, safety inspection operations must be made without any technician passing through the inspected zone, and thus neutralises it. An activity may also require specific machines and tools resources. For all $a \in \mathcal{A}$ and $r \in \mathcal{R}$, $consu_a^r$ represents the consumption of the resources r by the activity a.

We consider two types of *precedence* relation between activities. The first one is $\mathcal{P}_s \in \mathcal{A}^2$ and is called *simple precedence*. If a_i and a_j are two activities in $\mathcal{A}$, $(a_i, a_j) \in \mathcal{P}_s$ means that a_j must start after the end of the a_i. The second type, *hard precedence* is denoted $\mathcal{P}_h \in \mathcal{A}^2$. If (a_i, a_j) belongs to $\mathcal{P}_h$ then a_j has to start at the end of a_i. The precedence graph induced by $\mathcal{P}_s$ and $\mathcal{P}_h$ is assumed to be acyclic.

Finally, we consider an *incompatibility* symmetric relation $\chi \subset \mathcal{R}^2$ that models resources that cannot belong to the same station. Formally, (r_1, r_2) belongs to χ expresses that resources spans of r_1 and r_2 cannot overlap. Resource span of $r \in \mathcal{R}$ is the smallest time interval that contains all intervals of activities consuming r.

In order to formalise this problem with a Constraint Programming approach, for each activity $a \in \mathcal{A}$, we consider a variable $\texttt{start}_a$ whose domain is $[\![0, H]\!] \subset \mathbb{N}$ (where H is a given upper bound of the makespan) and that represents the start date of a. We do not detail here the formalisation of the constraints described along with the model description.

Example 1 (Toy example). Consider the following build process: activities, resources and zones along with their capacities are respectively $\mathcal{A} = \{a_1, \ldots, a_5\}$, $\mathcal{Z} = \{z_1, z_2\}$, $\mathcal{R} = \{r_1, r_2, r_3\}$, $cap_{z_2} = 2$ and $cap_{z_1} = cap_{r_1} = cap_{r_2} = 1$, $cap_{r_3} = 2$. Precedence relations are $P_s = \{(a_2, a_5), (a_3, a_4), (a_1, a_3)\}$, $P_h = \{(a_1, a_2)\}$ and $\chi = \{(r_1, r_3)\}$. Consumption of zones and resources by activities are described on Figure 1. Figure 2 represents a possible activities configuration satisfying all the constraints. Dashed rectangles represent zone neutralisation, for instance, a_2 and a_3 both neutralise z_1 at the same time. Activity a_5 has to wait until the end of a_4 because of incompatibility between resources r_3 and r_1.

	d	z_1	z_2	r_1	r_2	r_3
a_1	2	1	n	1		
a_2	1	n	1	1		
a_3	2	n	1		1	
a_4	1		1	1		
a_5	2	1				2

n stands for zone neutralisation.

Fig. 1: Resources consumption, zones occupation and neutralisation activities in Example 1.

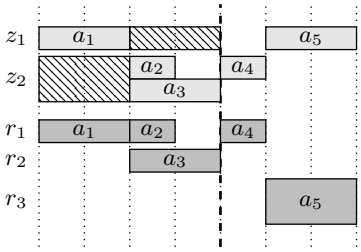

Fig. 2: A build process scheduling

The objective function should take into account the performance elements mentioned earlier, that is minimising makespan and minimising takt-time. We choose to consider the projection of scheduling onto stations by influencing, among other things, the takt-time, first, then the makespan (lexicographic objective function). In order to model the takt-time, we first consider the span interval of each resource. If a resource is attached to a unique station, resources spans will be a lower bound of takt-time. But only considering resource spans is not representative enough as all stations must have the same duration. This is why we fix an arbitrary number of stations and try to dispatch all resources span intervals in at most one station. This corresponds to minimising the number of cuts of resource spans by stations start/end dates. Because of the different relations between elements and the fact that all intervals have the same duration, the existence of a problem schedule without cut for a given scheduling horizon H is not guaranteed. For example, in Figure 2, consider $H = 8$ and a division in 2 stations, which means that takt-time is at most 4. A cut minimisation would put the end of the first station at time 4 (dashed line). In fact, the resource-span of r_1 is cut by this line as it starts at time 0 and ends at time 5. The other spans are not cut, thus the criterion has a value of 1. Once the number of cuts is minimised, we minimise the makespan. In the example, the makespan with one cut is 7.

All constraints and criteria in this model can be encoded using intervals and modelling structures of IBM ILOG Optimization Programming Language (OPL).

3 Experiments

The proposed model has been applied on two real industrial data sets depicting similar aircrafts, namely Design1 and Design2. In these two data sets, we have a set of 5 dimensioning resources composed of two types of machines: the first type is constituted by robots (R1 and R2) and the second by specific tools (T1, T2 and T3). They all have a capacity equal to 1 but note that it would be possible to increase it (done for some experiments not presented in this paper). Robots and specific tools have an incompatibility relation ($(R_i, T_j) \in \chi$ for all $i \in [1, 2]$ and $j \in [1, 3]$). We have about 180 tasks, 48 zones with capacity 1 by data set. Design1 has about 700 precedence relations and Design2 about 300.

For each experiment, we give the algorithm a maximum time limit of 60 seconds for all experiments except for the division into seven or more stations that were given a limit of 120 seconds. In addition, each result presented in the abstract has an optimal gap lower than 0.15 for cases with less than eight stations, lower than 0.25 for the eight stations case. The scheduling horizon H is equal to the sum of all activities durations. Such a value was large enough to allow the dispatch of resources to stations without any cut by resource span.

The experiments were run with CpOptimizer 20.1 on 20-core Intel(R) Xeon(R) CPU E5-2660 v3 2.60GHz, 64GB RAM, Ubuntu 18.04.5 LTS.

Table 1: Results in *takt-time* by number of stations

k	**4**	**5**	**6**	**7**	**8**
Design1	9h54'12	8h53	6h50'15	5h51	5h30
Design2	9h51	9h7'12	9h7'12	9h7'12	9h7'12

Table 1 represents the results obtained on our data sets by varying the number of stations from 4 to 8. We can notice that with the exception of the 4 stations configuration, Design1's takt-times are better than those associated with Design2. In a previous experiment, not presented in this paper, we have calculated for each Design, the overall takt-time lower bound from resource span. Their values are 5h30 for Design1 and 9h7'12 for Design2. Thus, we can notice that Design1 reaches its lower bound in the 8 stations configuration, while Design2 in the 5 stations configuration. So, unless we manage to reduce these bounds we can not anymore decrease either takt-time or makespan. Depending on the number of stations chosen, the two designs are Pareto optimal when considering takt-time and makespan.

4 Conclusion and future works

In this paper, we have proposed a scheduling-based analysis tool that allows us to evaluate the preliminary aircraft design against its performance on a pulsed assembly line. The experiments along with the industrial feedback show that the approach is promising.

In some experiments not presented in this paper, we have tested different assembly line configurations in which activities could choose among various resources of the same type (*e.g.* two robots of type $R1$). It strongly improves the results of Design2. Thus, a first future work could be to extend the model to add flexibility in the choice of resources *(RCPSP with modes)*. Another way to continue this work would be to consider costs associated with the number of stations and takt-time in order to differentiate solutions that are Pareto optimal in our current approach. Then, we have allowed activities to overlap on more than one station, but in real world instances, some may require to be performed in a single station. It might therefore be necessary to add this constraint in future models. Finally, the addition of an uncertainty on activities' duration could allow us to propose a more robust schedule with respect to the assembly line.

References

Polacsek, T., Roussel, S., Bouissière, F., Cuiller, C., Dereux, P. & Kersuzan, S. (2017), Towards thinking manufacturing and design together: An aeronautical case study, *in* 'Conceptual Modeling - 36th International Conference, ER 2017, Proceedings', Vol. 10650 of *Lecture Notes in Computer Science*, Springer, pp. 340–353.

Pralet, C., Roussel, S., Polacsek, T., Bouissière, F., Cuiller, C., Dereux, P., Kersuzan, S. & Lelay, M. (2018), A scheduling tool for bridging the gap between aircraft design and aircraft manufacturing, *in* 'Proceedings of the Twenty-Eighth International Conference on Automated Planning and Scheduling, ICAPS', AAAI Press, pp. 347–355.

Sanchidrián, T. B. (2019), Scheduling with limited resources along the aeronautical supply chain: from parts manufacturing plants to final assembly lines, PhD thesis, Industriales.