



HAL
open science

Towards an association of product configuration with production planning

Michel Aldanondo, Élise Vareilles, Meriem Djefel

► **To cite this version:**

Michel Aldanondo, Élise Vareilles, Meriem Djefel. Towards an association of product configuration with production planning. *International Journal of Mass Customisation*, 2010, 3 (4), p.316-332. 10.1504/IJMASSC.2010.037648 . hal-01773928

HAL Id: hal-01773928

<https://hal.science/hal-01773928>

Submitted on 28 Nov 2019

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.

Towards an Association of Product Configuration with Production Planning

Michel Aldanondo*, Elise Vareilles*^c and Meriem Djefel *^{&***}

*Toulouse University - Mines Albi - CGI, Albi France
(Tel: 33 5 63 49 32 34; e-mail: first_name.last_name@enstimac.fr).

** Toulouse University - INSA - LATTIS, Toulouse France

^c Corresponding author

Abstract: This communication presents the first ideas relevant to the possibility of coupling together, thanks to constraints modelling, product configuration tools with process planning tools in an interactive and simultaneous way, in order to pass decisions made from one to the other. The first section introduces the problem and the general ideas of the proposed solution. Two constraints based models, relevant to product configuration and process planning, are presented. Then first investigations for coupling these two models and associated problems are discussed. An example illustrates our proposal through out the paper.

Keywords: Product configuration, Process planning, Coupling of domains, Knowledge Based Reasoning, Constraint satisfaction problem, Filtering techniques, Interactive Aiding Design tool, Application

Reference:

Dr. Michel Aldanondo has been a professor at the école des mines d'Albi-Carmaux, France, since 1998. He had his PhD in 1992 after a degree in mechanical field in 1983. He worked for some industries abroad in the 80ies. He concentrates his researches on the development of interactive aiding design tools based on experts'knowledge and applies this approach to industrial problems (design of products, heat treatment operations, of supply chains, ...).

Dr. Elise Vareilles has been an assistant professor at the école des mines d'Albi-Carmaux, France since 2005. She had her PhD in 2005 and won the price of the best PhD Thesis of INP Toulouse in the same year. She works on the development of interactive aiding design tools based on knowledge and is part of the development and the improvement of CoFiADe.

Meriem Djefel is a PhD student at the école des mines d'Albi-Carmaux and the university of Toulouse, France. She has reached the team in 2007. She works in the adaptation of CoFiade on the particular problem of ATLAS.

1 Introduction

The aim of this communication is to present the first results of a study dealing with the development of an aiding system based on constraints that will simultaneously allow the configuration of a product and the planning of its production process in an interactive and automated way. It seems rather logical that on the one hand, product configuration decisions have strong consequences on the planning of its production process and that on the other hand, planning decisions provide hard constraints to product configuration. Therefore, we propose to associate these two problems in order to allow the propagation of the consequences of each product configuration decision toward the planning of its production process and the propagation of the consequences of each process planning decision towards the product configuration. This should reduce or avoid planning impossibilities due to product configuration and configuration impossibilities due to production planning.

This problem originates from a French national funded project called ATLAS whose purpose is the development of an open source software able to help industrialists to design (and not only configure) a product and its associated project simultaneously, by passing decisions made from one to the other.

A literature reviews showed us that this kind of problem has not been addressed in detail yet. In the design community, for more than fifteen years, many studies around Axiomatic Design (Suh 1990), Design Structure Matrix (Steward 1981), or Function Behaviour Structure (Gero 1990), have proposed different domains (customer, functions, requirements, behaviour, physical, process, resource) in order to characterize product development. A recent paper from the DSM community (Lindemann 2007) proposed a mapping of the four domains: functions, components, process and resources as shown in Fig.1. Our proposition is based on these four domains and we associate the function and component views within product configuration, and the process and resource views within the planning of the production process.

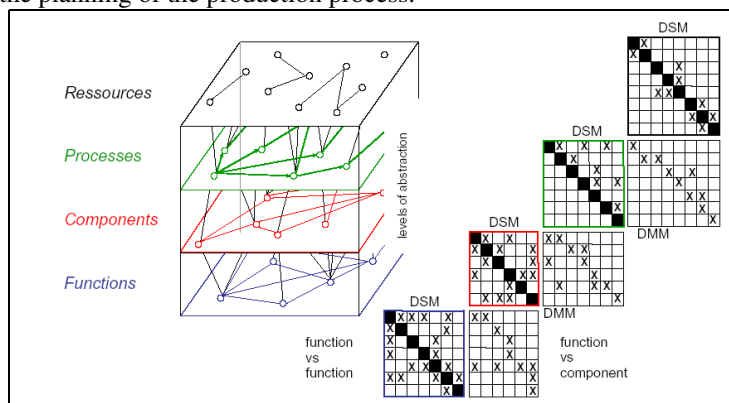


Figure 1 The four domains of our study from (Lindemann 2007)

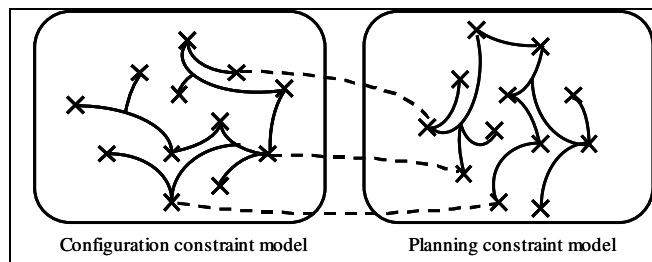
In the configuration community, many authors among them (Sabin and Wegel 1998 or Soininen et al. 1998) have shown that product configuration could be efficiently modelled and aided when considered as a Constraints Satisfaction Problem (CSP). It has also been shown in (Aldanondo and Vareilles 2008) that the four domains of (Lindemann

2007) could be considered in a constraint problem. In a same way, authors interested in scheduling as (Dechter et al. 1991 or Laborie 2003) have shown that project planning could be also modelled and aided when considered as a Temporal CSP. A CSP is a triplet $\{X, D, C\}$ where X is a set of variables, D a set of domains (one for each variable) and C a set of constraints linking the variables (Montanari 1974 and Tsang 1993). The variables can be discrete or continuous. The constraints restrict the possible combinations of values of a variable set.

As our goal is to set up a system that allows interactive and automated assistance for configuration and planning, we are only interested by the filtering capability of CSP and not solving or optimising ones. Each time a user reduces the domain of a variable (either in the configuration or planning domains) a filtering or constraint propagation process is launched (either for configuration or planning) in order to reduce the domain of other variables of the problem.

Once given these elements, we propose to consider configuration and planning problems as two constraint satisfaction problems. In order to propagate decision consequences between the two problems, we suggest linking these two constraint models with what we call coupling constraints (dotted lines) as shown in Fig. 2.

A question raised by this proposition is relevant to the possibility of considering a single constraint problem encompassing the four domains. From a theoretical point of view, there is of course no problem to consider a single problem. But, differences between configuration and planning when considered as a CSP can be pointed out. Firstly, knowledge relevant to configuration and planning is different: dealing with product diversity or process diversity. Secondly, people who provide the knowledge are different: design and marketing services or process and manufacturing departments. Thirdly, users might be different and require the possibility to operate or to process separately and sometimes independently configuration and planning assistance (especially during set up or model tuning). And finally propagation techniques are rather different, discrete arc consistency is used for configuration while temporal or numerical propagation is



necessary for planning.

Figure 2 Proposed coupling approach

The paper is therefore organized as follow: sections 2 and 3 address product configuration and planning issues. The association of the two constraints models is then investigated and discussed in section 4. A detailed example runs through out the paper.

2. Product configuration

This section concerns the definition of the configuration problem that we address. The problem is first defined, then the constraint model is described with propagation techniques. The description of the example finishes this section.

2.1 Configuration Basics

From previous studies (Sabin et al. 1998 or Soinenen et al. 1998), it seems that some common features defining configuration exist. First of all, a product is described by a set of components. Secondly, a generic model of configurable products is able to represent a family of products with all possible variants and options. This model is created by identifying a set of components, a set of product properties and a set of constraints, that restrict possible combinations of components and property values.

Thirdly, customers' requirements can be expressed by a selection or a domain restriction of a component or a property value. Then, we can define the configuration of a product as "finding at least one component set that satisfies all the constraints and the customer's requirements".

We associate previous properties with some kind of product description that matches the "function" domain while components and component groups are associated with the physical or "component" domain presented in section 1.

2.2 Constraint Model and Propagation Techniques

Each group of components and each product property is associated with a configuration variable defined on a symbolic domain. Each component and each property value corresponds with one value of the variable. The constraints represent the allowed or excluded (solid or dot lines of Fig. 3) combinations of components and property values. As far as all component groups and product properties are always present in any configured product (meaning that any component group or property always exist), the variables belonging to the constraint problem always exist and standard CSP propagation techniques, relying on arc consistency, can be used to interactively assist configuration.

However, in most of the cases as explained in (Aldanondo et al. 2003) some component groups or product properties must be added during interactive configuration. Therefore variables must be added to the CSP during constraint propagation. Thus, the Dynamic extension of the CSP, DCSP proposed by (Mittal and Falkenhainer 1990, frequently called Conditional-CSP), is used in order to control the variable existence. DCSP adds to a CSP: a set of initial variables, variables that are called active and that are part of any configured product, a set of potential active variables, variables that are called inactive and that can appear or not in a solution, a set of activity constraints allowing the control of the existence of inactive variables. Activity constraints establish a variable's activity based on an assignment of values to a set of active variables: a specific value of a variable "X" implies the existence of variable "Y"

(arrow on Fig. 3). This is a key point that will have consequences on planning with the same kind of requirement relevant to the control of the existence of tasks.

2.3 Example relevant to Product Configuration.

The figure 3 shows the configuration model of our example. The product taken for example is a very simple crane. In Fig. 3, each box is a variable with its values and each line and arrow represent constraints. In its functional (or descriptive) view (left part of Fig. 3), the product can be defined with:

- four product properties:
 - C_height , the height of the crane (two possible values: “4m”, “8m”),
 - C_width , the width of the crane (two values: “2m”, “4m”),
 - M_load , the maximum load (two values less than one ton :“<1t”, between 1 and 2 tons :“1t<<2t”),
 - $Ctr-Cab$, specifying the presence of a control cabin in the configured crane with two values “yes” and “no”.
- three constraints that reduce the solution space and exclude combinations of values:
 - ($C_height = “4m”$) incompatible with ($C_width = “4m”$), a crane cannot have the same height and width,
 - ($M_load = “1t << 2t”$) incompatible with ($C_width = “4m”$), the maximum load is not compatible with the larger width,
 - ($C_height = “4m”$) incompatible with ($Ctr-Cab = “yes”$), a height of four meters forbids the presence of a control cabin.

In its physical (or component) view (right part of Fig. 3) the model shows:

- four groups of components:
 - V_Struct , the vertical structure gathering four physical components combining different acceptable loads and lengths: “V4_n”, “V4_s”, “V8_n”, “V8_s”,
 - H_Struct , the horizontal structure gathering two physical components according to its length “H2_n”, “H4_n”,
 - $Engine$, the crane engine gathering 2 physical components with two different powers “E_lp”, “E_hp”,
 - Cab , the control cabin gathering two physical components according to the engine power “Cab_lp”, “Cab_hp”.
 - in this physical view, one constraint associates the possible engines $Engine$ and the control cabins Cab : $(Engine, Cab) = \{ (“E_lp”, “Cab_lp”), (“E_hp”, “Cab_hp”) \}$.
-

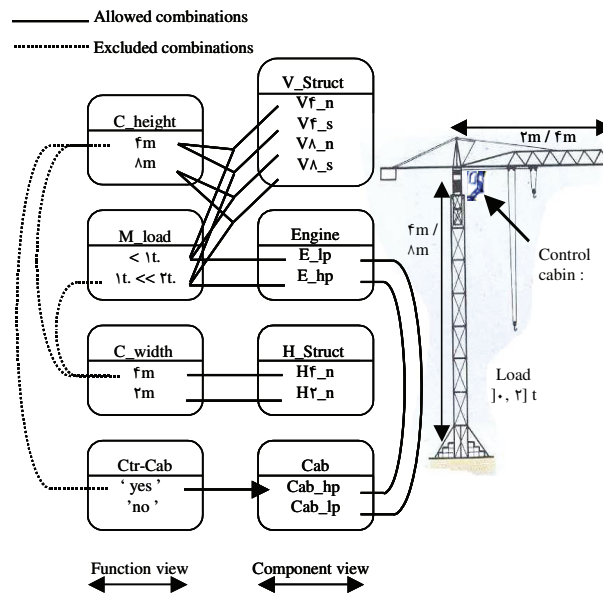


Figure 3 Product configuration model

The functional and physical views are linked with constraints that show how the components can fulfil the properties of the functional view. Three constraints show the following possible combinations:

- the height of the crane (C_height) and the maximum load (M_load) are linked with the vertical structure (V_Struct),
- the maximum load (M_load) impacts the engine selection ($Engine$),
- the width of the crane (C_width) is associated with the horizontal structure (H_Struct).

One activity constraint allows to control the existence of the variable (Cab): $Ctr-Cab = 'yes' \rightarrow Cab$ becomes active.

3. Production planning

This section is concerned by the definition of the planning problem we address. The problem is first defined, then the constraint model is described and propagation techniques are discussed. The description of the example finishes this section.

3.1 Planning Problem Description

As we address production planning, we consider that the production process is a set of task entities. A task entity is defined with:

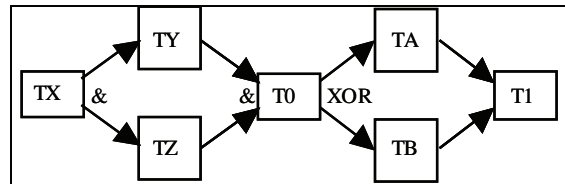
- temporal parameters: possible start time (pst), possible finishing time (pft), possible duration (pdt),
- resource parameters: required resource (rrs), quantity of required resource (qrs).
- compatibility constraints can link possible duration (pdt), with required resource (rrs) and/or quantity of required resource (qrs).

In this paper we only consider planning with unlimited resource capacity. Therefore resource parameters (rrs and qrs) are just used to modulate the possible duration of a task (pdt) and not as added constraints to the scheduling.

The three temporal parameters are numerically defined with intervals while resource parameters can remain symbolic (as we consider unlimited resource capacity). The production process is defined with a set of tasks and a set of precedence constraints.

A precedence constraint expresses that the processing of task Y is after the processing of task X : $Y.pst > X.pft$. This kind of constraints has certain similarities with primitives proposed by (Allen 1983): before, after, starts, finishes....

With these elements the production process can gather sequential tasks and parallel (or unconstrained) tasks as shown with the “AND” node (&) in the left part of Fig. 4 with



tasks TY and TZ . A planning decision corresponds with a value selection or domain restriction of either a temporal parameter or resource parameter. The goal of the interactive assistance is therefore to propagate each decision consequence on the planning model.

Figure 4 Planning process model

When planning is associated to a configuration problem, a specific requirement is added: the need to control the existence of some tasks or to select a task path during planning. The reason can be either to take into account the production process of a component group that does not belong to any configured product (section 2.2) or to be able to take into account process variants as make, subcontract or buy for example. Therefore some kind of exclusive “XOR” node, as shown in the right part of Fig.4, between the tasks TA and TB , must be considered.

3.2 Constraint Model and Propagation Techniques

A task entity (noted Tt , for task “t”) is created and is always composed of three real variables corresponding with the three temporal parameters. If necessary, resource parameters ($Tt.rrs$ and $Tt.qrs$) can be added. Temporal parameters are numerical parameters defined with intervals:

- pst: possible start time, $Tt.pst = [\text{optimistic start time, pessimistic start time}]$,
- pdt: possible duration, $Tt.pdt = [\text{optimistic duration, pessimistic duration}]$,
- pft: possible finishing time, $Tt.pft = [\text{optimistic finish time, pessimistic finish time}]$,

When all the variables are numerical and each constraint written with a mathematical formula, $f(x_1, x_2, \dots, x_n) (= > <) 0$, bound consistency, proposed by (Lhomme 1993) and based on interval arithmetic (Moore 1996), proposes filtering techniques that operate fine if the following conditions are fulfilled:

- $f(x_1, x_2, \dots, x_n) = 0$ can be projected on any variable x_i meaning that a function f_i exists as: $x_i = f_i(x_1, x_2, \dots, x_{i-1}, x_{i+1}, \dots, x_n)$,
- any projections f_i is continuous and monotonous.

We consider only the two following kinds of mathematical formulae as planning constraints:

- $TY.pst > TX.fst$, expressing that task Y is after task X
- $Tt.pft = Tt.pst + Tt.pdt$, expressing the relation between starting, finishing times and duration for any task.

We can therefore use bound consistency filtering algorithm with the projections of the two previous kinds of constraints. Then, interactive planning is aided by the propagation of all the constraints in order to reduce the definition domain of all temporal parameters and resource parameters. This propagation technique works perfectly as far as there is no exclusive “XOR” node.

In order to deal with the “XOR” node or to modulate the existence of some tasks or sets of tasks, it is necessary to be able to express that some of them have their existence conditioned. A CSP extension called Conditional and Composite Temporal Constraint Satisfaction Problems or CCTCSP has been proposed in (Mouhoub and Sukpan 2005). But, CCTCSP does not allow the tasks (represented by events) to have an interval for their duration. We have therefore defined a meta-task entity XOR that includes at least two tasks that are in exclusion. This meta-task has also three temporal parameters: $XOR.pst$, $XOR.pft$ and $XOR.pdt$.

Assume that we have a process TO followed by two exclusive tasks, TA and TB , and then TI (as in the right part of Fig. 4). The tasks TA and TB are in exclusion, meaning that only one of the two must be part of the solution.

In order to select which task between TA and TB belongs to the solution, we encompass them in a meta-task called XOR_{AB} , as shown in Fig. 5, and we add :

- the particular value 0 in the definition domain of the possible duration of TA and TB , meaning that if the duration of the task equals 0 the task does not belong to the

solution,

- a constraint that forces at least one of the task to be part of the solution: ($TA.pdt = 0$, $TB.pdt \neq 0$) and ($TA.pdt \neq 0$, $TB.pdt = 0$).

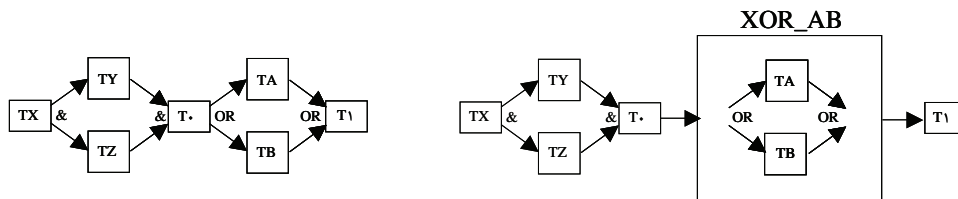


Figure 5 XOR node including two tasks

The possible starting and finishing times of XOR_AB are constrained by:

- $XOR_AB.pst > T0.pft$, meaning that the meta-task XOR_AB starts after $T0$,
- $XOR_AB.pst \leq TA.pst$ and $XOR_AB.pst \leq TB.pst$, meaning that the meta-task XOR_AB constrains the beginning of TA and TB ,
- $TA.pft \leq XOR_AB.pft$ and $TB.pft \leq XOR_AB.pft$, meaning that the meta-task XOR_AB constrains the end of TA and TB ,
- $T1.pst > XOR_AB.pft$, meaning that the meta-task XOR_AB finishes before $T1$.

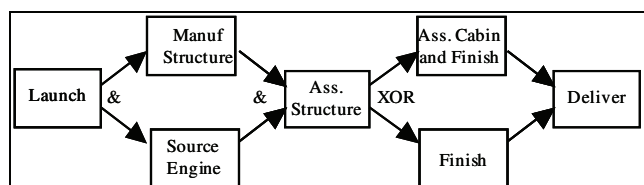
The possible duration of the meta-task XOR_AB equals the union of the possible duration of the tasks that are included in itself: $XOR_AB.pdt = \text{Union}(TA.pdt, TB.pdt)$. But its possible duration must strictly be greater than 0 because at least, one of the two tasks is part of the solution.

The meta-task XOR temporal parameters are propagated with the same bound consistency filtering algorithm. When two sequences of tasks are considered, it is first necessary to aggregate each path then to define the meta-task XOR on the aggregated task.

It is important to note that the meta-task XOR permits us to select a task or a path in the planning by simulating a deactivation of tasks. These selection and deactivation can be the result of a user's selection, the result of a propagation of a reduction of the duration of the meta-task XOR or as we will see in next section, the propagation of a coupling constraint. Once a task is selected and the others deactivated, the temporal parameters of the meta-task XOR and the selected task are equals. This specific piece of model is a key proposition in order to fulfil the coupling requirement.

3.3 Example relevant to Process planning

The figure 6 shows the planning model relevant to the production of the crane example composed of seven tasks. The possible duration (pdt) for each task is provided, while possible start and finish time are considered as a result of planning. A required resource



with a quantity is defined only for the two tasks (*Manuf Structure* and *Deliver*). A launching task has been added.

Figure 6 Production process of the crane

The seven tasks are defined as follows:

- *Launch*, notated L , allows to set a launching time: $L.pst = L.pdt = L.pft = 0$.
- *Manuf Structure*, notated MS , corresponds with the manufacturing of the two structures, the vertical and the horizontal ones (V_struct and H_Struct), a small and a large machine can be used with a quantity of 1: $MS.pdt = [3, 6]$, $MS.rrs = \{large_mach, small_mach\}$, $MS.qrs = "1"$.
- *Source Engine*, notated SE , corresponds with the sourcing of the engine (E_lp and E_hp): $SE.pdt = [2, 4]$.
- *Ass Structure*, notated AS , corresponds to the assembly of the structures with the engine, whose duration always equals to 2: $AS.pdt = 2$.
- *Ass Cabin and Finish*, notated ACF , corresponds to the assembly of the cabin on the crane and to the finish. This task exists if and only if a control cabin is present in the configuration: $ACF.pdt = [3, 4]$.
- *Finish*, notated F , corresponds to the finish of the crane without a control cabin. This task exists if and only if there is no control cabin in the final product: $F.pdt = [1, 2]$.
- *Deliver*, notated D , corresponds with the delivering of the complete crane. Two different transportation resources can be used (slow or fast transport) and modulate its duration: $D.pdt = [1, 2]$, $D.rrs = \{fast_trans, slow_trans\}$, $D.qrs = "1"$. The following constraint links the duration and the transportation resources: $(D.rrs, D.pdt): \{ (slow_trans, [1, 1.5]), (fast_trans, [1.5, 2]) \}$

As two tasks are in exclusion (*Ass Cabin and Finish* and *Finish*), a meta-task *XOR* must be added to the model. This *XOR*, notated XOR_ACFF , includes the two tasks in exclusion (see the double box in Fig. 7). Its duration equals the union of the possible duration of the tasks that are encompassed in itself: $XOR_ACFF.pdt = U(ACF.pdt, F.pdt) = U([3, 4], [1, 2]) = \{[1, 2], [3, 4]\}$

Given these elements, the figure 7 shows the planning state after constraint propagation when starting and finishing dates of all tasks had an initial definition domain equal to $[0, 20]$.

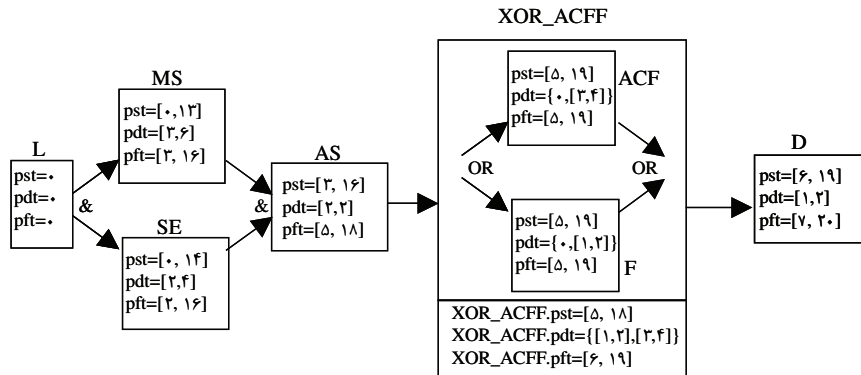


Figure 7 Planning model of the crane process

4. Coupling Configuration and Planning

Coupling principles and relevant coupling constraints are introduced and some illustrations with the crane example are provided and discussed.

4.1 Coupling Constraints

As explained in the introduction, our goal is to be able to propagate decisions consequences from product configuration to process planning and also, in the opposite way, from process planning to product configuration. Therefore we define a coupling constraint as a compatibility constraint that links variables of the configuration model with variables of the planning model.

Any variable of the configuration model, belonging either to the function or the component view, can belong to a coupling constraint. On the planning model side, resource parameters (required resource, rrs, quantity of required resource, qrs) or temporal parameters of a task (pdt, pst, pft) can belong to a coupling constraint. A variable belonging to a coupling constraint is called a coupling variable. Given these elements three coupling cases can be derived:

- when the coupling variable is a resource parameter (rrs or qrs). This allows to propagate the impact of a configuration decision on the selection of the required resource and/or resource quantity, reverse behaviour from resource selection to product configuration is of course possible.
- when the coupling variable is a temporal parameters (pst, pdt, pft). This allows to propagate the impact of a configuration decision on the modulation of the duration, of the beginning and of the end of a task, reverse behaviour from temporal parameters to product configuration can also of course be considered.
- when the coupling variable is the possible duration (pdt) of a meta-task *XOR*. The previous behaviour is completed with the possible selection of one task or path that

is in exclusion.

4.2 Illustration with the Crane Example

Some coupling constraints are first proposed and then coupling is illustrated on the crane example. These coupling constraints are illustrated as follow:

- the planning variable is a resource parameter: The resource of the task *Manufacturing Structure*, *MS.rrs*, is linked with the height of the crane, *C_height*, with the following allowed combinations, Constraint (*C_height*, *MS.rrs*): { (“4m”, “small_mach”), (“8m”, “large_mach”) }.
- the planning variable is a duration parameter that does not belong to a meta-task *XOR*. The duration of the task *Manufacturing Structure*, *MS.pdt*, is linked with the maximum load, *M_load*, with the following allowed combinations: Constraint (*M_load*, *MS.pdt*): { (“<1t”, [3 , 4.5]) , (“1t << 2t”, [4.5 , 6]) } . The duration of the task *Source Engine*, *SE.pdt*, is linked with the engine component, *Engine*, with the following allowed combinations: Constraint (*Engine*, *SE.pdt*): { (“E_lp”, [2, 3]) , (“E_hp”, [3, 4]) } .
- the planning variable is a duration parameter that belongs to a meta-task *XOR* in order to select the existence of a task. The duration of the meta-task *XOR_ACFF.pdt*, is linked with the existence of a control cabin, *Ctr-Cab*, with the following allowed combinations: Constraint (*Ctr-Cab*, *XOR_ACFF.pdt*): { “no”, [1, 2]) , (“yes”, [3 , 4]) .

4.2.1 Propagating Product Configuration Decisions on Production Planning.

If we assume the following configuration decisions: *C_height* = “8m”, *C_width* = “2m” and *M_load* = “1t << 2t”, and leave open the decision relevant to the existence of a control cabin.

The filtering of the configuration model provides the component set:

- *V_Struct* = “V8_s”
- *H_Struct* = “H2_n”
- *Engine* = “E_hp”.

The filtering of the coupling constraints provides:

- The selection of resource: *MS.rrs* = “large_mach”
- The modulation of the task duration: *MS.pdt* = [4.5 ,6] and *SE.pdt* = [3, 4]

Planning constraint propagation leads to the reduction of the tasks (as shown in Fig. 8):

- *Launch*: *L.pst* = *L.pdt* = *L.pft* = 0,
 - *Manuf Structure*: *MS.pdt* = [4.5 ,6], *MS.pst* = [0, 11.5] and *MS.pft* = [4.5 , 16]
-

- *Source Engine*: $SE.pdt = [3, 4]$, $SE.pst = [0, 13]$ and $SE.pft = [3, 16]$
- *Ass Structure*: $AS.pdt = [2, 2]$, $AS.pst = [4.5, 16]$ and $AS.pft = [6.5, 18]$
- *XOR_ACFF*: $pdt = \{ [1, 2], [3,4] \}$, $XOR_ACFF.pst = [6.5, 18]$, $XOR_ACFF.pft = [7.5, 19]$
- *Ass Cabin and finish*: $ACF.pdt = \{ 0, [3, 4] \}$, $ACF.pst = [6.5, 19]$ and $ACF.pft = [6.5, 19]$
- *Finish*: $F.pdt = \{ 0, [1, 2] \}$, $F.pst = [6.5, 19]$ and $F.pft = [6.5, 19]$
- *Deliver*: $D.pdt = [1, 2]$, $D.pst = [7.5, 19]$ and $D.pft = [8.5, 20]$

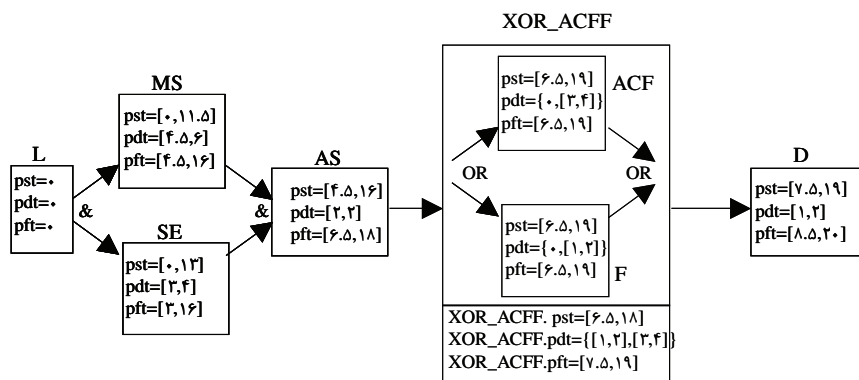


Figure 8. Planning after configuration.

One of the consequences of the coupling lies in the increase of the total finishing time ($D.pft$) from $[7, 20]$ to $[8.5, 20]$. If we assume now that the person in charge of planning wants to secure the manufacturing tasks and selects the maximum duration for them: $MS.pdt = [6,6]$ and $SE.pdt = [4,4]$. Planning propagation gives the planning of Fig. 9 with a total finishing time $D.pft = [10,20]$.

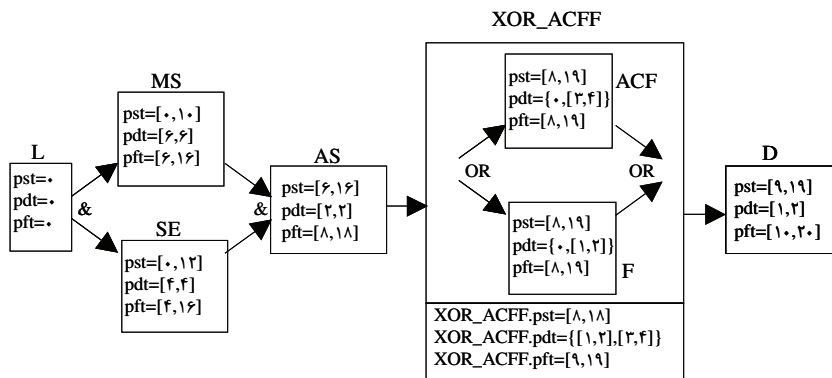


Figure 9. Planning after *Manufacturing* freezing duration.

4.2.2 Propagating production planning decisions toward product configuration.

If we assume that the total finishing time $D.pft$ should be now " ≤ 11 ". This implies:

- For the task D : $D.pft = [10, 11]$ and $D.pst = [9, 10]$,
- For the task XOR_ACFF , $XOR_ACFF.pft = [9, 10]$ and $XOR_ACFF.pst = [8, 9]$,
- For the task F : $F.pft = [8, 10]$ and $F.pst = [8, 10]$,
- For the task ACF : $ACF.pft = [8, 10]$ and $ACF.pst = [8, 10]$.

The propagation of the constraints expressing that finish time equal start time plus duration ($Tt.pft = Tt.pst + Tt.pdt$) and the constraint of the meta-task XOR_ACFF gives:

- For the task XOR_ACFF : $XOR_ACFF.pdt = [1, 2]$
- For the task ACF : $ACF.pdt = [0, 0]$
- For the task F : $F.pdt = [1, 2]$

The planning propagation has therefore selected the task F and deactivated the task ACF . The null value for $ACF.pdt$ is then propagated to the configuration model and forbid the selection of a control cabin by reducing the variable $Ctr-Cab$ to the value "no" as shown on Fig. 10. The remaining decisions concern the selection of the delivering resource and the final tuning of the duration of the delivering and finish tasks.

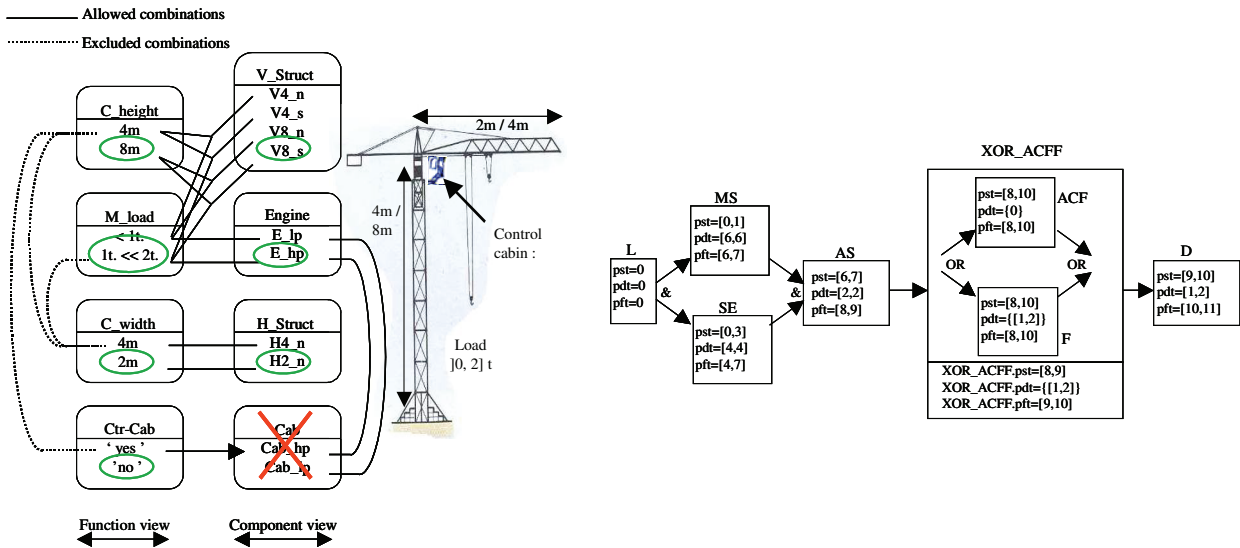


Figure 10 Configuration and planning models after final coupling

5. Conclusion

The aim of this paper has been to present the first results relevant to the development of an interactive aiding system based on constraints, that simultaneously allows product configuration and production planning. The main interest of this system is to be able to take into account product configuration decisions when dealing with production planning and production planning decisions when dealing with product configuration.

If this problem sounds rather common and natural in industry, our literature survey did not bring any significant contributions from a scientific point of a view. On one side, we found many contributions in design and configuration aided with constraint approaches and, on the other, many dealing with constraint based process planning. But study dealing with both are extremely rare and most of the time corresponds with some specific development as (Steward and Tate 2000).

Therefore, we first briefly presented both configuration and planning problems and proposed to associate each of them with a constraint satisfaction problem and to link them with what we called coupling constraints. Then, a configuration problem was defined and modelled with two views: functions and components. Constraint propagation is achieved thanks to arc consistency techniques. An example dealing with a simple crane was introduced. The planning problem was defined and modelled thanks to a network of tasks with a specific piece of model that allows to take into account exclusive "XOR" nodes. Bound consistency techniques are used to propagate constraints.

For the two problems, it was clearly shown that the hard problem lies in the necessity to be able to control the existence of component in configuration and the existence of tasks in planning. The dynamic extension of CSP (DCSP) was used for configuration while a mix of Conditional and Composite Temporal CSP (CCTCCSP) and DCSP was used for planning. The crane example was extended with production planning entities.

The final section described the coupling constraints that can associate the two previous models in order to pass decisions made from one to the other. The crane example was used to show how configuration decisions were propagated to planning and how planning decisions were propagated to configuration.

One of the interests of the proposed system is that it relies only on the simple assembly of two constraint filtering techniques: arc consistency for discrete CSP and Bound consistency for numerical or mixed CSP. The crane example can be soon consulted at <http://cofiade.enstimac.fr/cgi-bin/cofiade.pl> (chose model Crane-08). The other main interest is that both domains remain independent. Therefore, knowledge gathering, model set up, processing and actors of each domain can work independently and should gather or negotiate only for defining and testing coupling constraints.

These propositions must be considered as first results that need to be consolidated with some study relevant with structured (or multi-level) configuration and planning, finite resource capacity planning and some scalability considerations. We are currently working on the model of an industrial problem in order to test on a real example the limits of our proposition.

Acknowledgement

The authors would like to thank the French National Research Agency (ANR), the 7th Strategic Activity Domain (Architecture and Integration) of Aerospace Valley for their involvement in this project and their partners in the ATLAS project.

References

- Aldanondo, M. Hadj-Hamou, K., Moynard, G. and Lamothe, J. (2003). Mass customization and configuration: Requirement analysis and constraint based modeling propositions. *Journal of Integrated Computer-Aided Engineering*, vol. 10 - n° 2, pp 177-189.
- Aldanondo, M. and Vareilles, E. (2008). Configuration for mass customization: how to extend product configuration towards requirements and process configuration. *Journal of Intelligent Manufacturing*, vol. 19, pp 521-535.
- Allen, J. (1983). Maintening knowledge about temporal intervals. *Communication of the ACM*, vol 26(11), pp 832-843.
- Dechter, R., Meiri, I. And Pearl, J. (1991). Temporal Constraint Satisfaction Problems. *Artificial Intelligence*, n°49, pp 61-95.
- Gero J.S. (1990). Design prototypes: a knowledge representation schema for design. *Artificial Intelligence magazine*, vol. 11 - n°4, pp 26-36
- Laborie, P. (2003). Algorithms for Propagating Resource Constraints in AI Planning and Scheduling: Existing Approaches and New Results. *Artificial Intelligence*, n°143, pp 151-188.
- Lindemann, U. (2007). A vision to overcome “chaotic” design for X processes in early phases. *Proc. of ICED07*, Paris France.
- Lhomme, O. (1993). Consistency techniques for numerical CSPs. *Proc. of IJCAI 93*, Chambéry France, pp 232-238.
- Mittal, S. and Falkenhainer B. (1990). Dynamic Constraint Satisfaction Problems, *Proc. of AAAI90*, Boston USA, pp 25-32.
- Montanari, U. (1974). Networks of constraints: fundamental properties and application to picture processing. *Information sciences*, vol. 7, pp 95-132.
- Moore, R.E. (1966). Intervals Analysis. *Prentice Hall*.
- Mouhoub, M. Sukpan, A. (2005). A New Temporal CSP Framework Handling Composite Variables and Activity Constraints, *Proc. Of the 17th IEEE International Conference on Tools with Artificial Intelligence*, Hong-Kong, pp 143-149.
- Sabin, D. and Weigel, R. (1998). Product Configuration Frameworks – A survey. *IEEE Transactions on Intelligent Systems*, vol. 13 -n° 4, pp 42-49.
- Soininen, T., Tiihonen, T., Männistö, T. and Sulonen. R. (1998). Towards a General Ontology of Configuration. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, vol. 12 - n° 4, pp 357-372.
- Steward, D. (1981). The Design Structure System: A Method for Managing the Design of Complex Systems. *IEEE Transactions on Engineering Management*, vol. 28, pp 71-74.
- Steward D., and Tate D. (2000). Integration of axiomatic design and project planning. *Proc. of first Int Conf. on Axiomatic Design*. Cambridge MA USA, pp 285-289.
-

Suh, N. (1990). The principles of design. *Oxford Series*.

Tsang, E. (1993). Foundations of constraints satisfaction, *Academic Press, London*.