



**HAL**  
open science

## Building renovation adopts mass customization

Andres Felipe Barco Santa, Élise Vareilles, Paul Gaborit, Michel Aldanondo

► **To cite this version:**

Andres Felipe Barco Santa, Élise Vareilles, Paul Gaborit, Michel Aldanondo. Building renovation adopts mass customization: Configuring insulating envelopes. *Journal of Intelligent Information Systems*, 2017, 49 (1), pp.119-146. 10.1007/s10844-016-0431-6 . hal-01599431

**HAL Id: hal-01599431**

**<https://hal.science/hal-01599431>**

Submitted on 17 Oct 2017

**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.

# Building renovation adopts mass customization

## Configuring insulating envelopes

Andrés F. Barco<sup>1</sup>    Élise Vareilles<sup>1</sup>    Paul Gaborit<sup>1</sup>  
Michel Aldanondo<sup>1</sup>

**Abstract** This work is motivated by an industrial need of manufacturing façades insulating envelopes in order to reduce energy consumption in residential buildings. An insulating envelope is a configuration of a set of rectangular panels that respects a set of limitations. Due to the number of façades to be renovated and the number of possible configurations for a single façade, the envelope configuration is both a *mass customization* problem as well as a *combinatorial* one. The paper then introduces a decision support system based on the framework of constraint satisfaction, as it fits neatly the constrained nature of the problem. Two configuration tasks have been identified as prerequisite to envelopes configurations: (1) the configuration of a questionnaire for information inputs and (2) the configuration of a constraint satisfaction problem for each one of the façades to be renovated. The system architecture promotes maintenance, modularity and efficiency as different configuration tasks are divided into web-services. Conception and implementation of the massive building thermal renovation are then supported.

**Keywords** Building thermal renovation · Massive product configuration · Decision support system · Constraint satisfaction · Web-service architecture

## 1 Introduction

Product configuration and mass customization, i.e., the art of building products guided by customer needs under mass production conditions (Soininen et al. 1998; Yang et al. 2008),

---

✉ Andrés F. Barco  
andrez.fbs@gmail.com

Élise Vareilles  
elise.vareilles@mines-albi.fr

<sup>1</sup> Centre Génie Industriel, Université de Toulouse - Mines d'Albi, Campus Jarlard, 81013 Albi Cedex 09, France

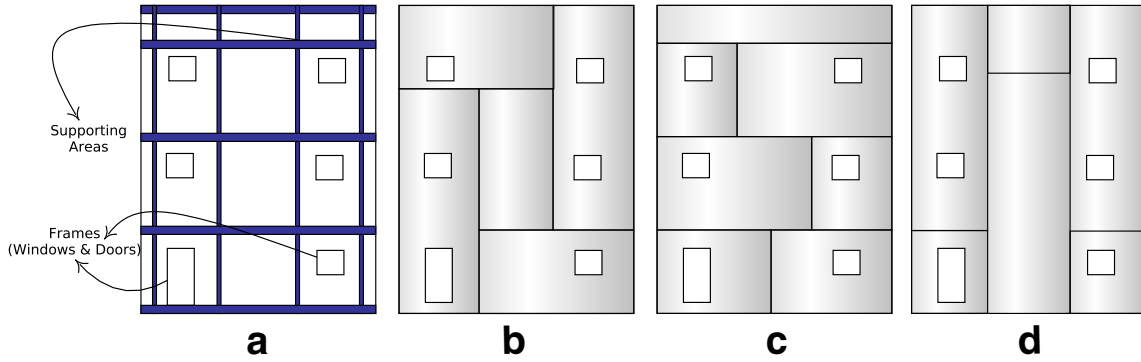
are tasks that have been increasingly supported by intelligent systems as they have been identified as highly complex (Blecker and Abdelkafi 2006; Soininen et al. 2000). Solving a mass customization problem is complex in such an extent that specialized techniques from *Artificial Intelligence* (AI) and *Operation Research* (OR) have been used, often extended, to handle these ubiquitous industrial problems (Felfernig et al. 2014). AI and OR applications to mass customization found their roots in automotive industry and computer industries (see surveys of MacCarthy et al. 2003 or Fogliatto et al. 2012) but quickly spread towards engineering to order systems as for example: railways interlocking systems (Falkner and Schreiner 2014), equipment and services on cement and minerals industries (Höfling 2014), sales and marketing (Höfling 2014) or mobile networks (Nica et al. 2014).

A particular scenario of mass customization arises from the context of *building thermal renovation* as an effort to reduce current energetic consumption levels of buildings (Center TEC 2011; Council UGB 2013; Pérez-Lombard et al. 2008). Here, the problem lies in the configuration of rectangular parameterizable panels, and their attaching devices called fasteners, that must be allocated over the façade area in order to provide an insulating envelope (also referred to as *layout plan*) (Jelle 2011; Aldanondo et al. 2014). An insulating envelope is composed of a set of insulating panels without any holes. Each of the panels is configurable in the sense that:

- Its size (width and height) must be assigned respecting manufacturing, transportation and installation limitations.
- If the panel covers frames (windows or doors), the position of each frame w.r.t. the panel must be determined for its manufacturing and installation.
- Its position over the façade must be accurately set for its installation. Panels must be attached over specific areas strong enough to support their weight.
- The position and number of attaching devices, called fasteners, must be determined for its installation.

Panels are configured at design time, then manufactured, shipped and installed on the working site. The configuration of panels is limited by a set of conditions, such as manufacturing and environmental conditions, that begin with design and go through all the renovation process until panels' installation. These limitations make the configuration of insulating envelopes a complex and error prone task. By now, hand-made methods need to be replaced by an industrialized process and supported by computer-based solutions.

Computer support for this thermal renovation process, and its conception by means of decision support system, is motivated by the following facts. First, the diversity of existing façades is very large as a result of history, building's technical aspects, or locations. Second the expectations of the architects or building owners about the façade style to renovate are also numerous. These two points lead to an extreme diversity of façade renovation problems. Traditional methods of course are able to handle them but for a high cost. The challenge is to be able to achieve the façade renovation problem with all its diversity at a much lower cost within industrialized approaches and methods. Thus, the façade renovation problem inherits the key particularities of mass customization: allows tailored or specific façade renovation at the cost of a mass produced one. Furthermore, façades diversity and renovation expectations diversity are associated with solution diversity, meaning that for a given façade and given expectations many solutions can be identified, each of them with a given thermal performance and a given cost (an illustration is presented in Fig. 1). It is because of these three diversities that the envelopes configuration is a complex combinatorial problem. The computer support of the renovation process allows (1) to efficiently introduce the renovation specifications (geometrical, structural, etc), (2) to generate different valid configuration of



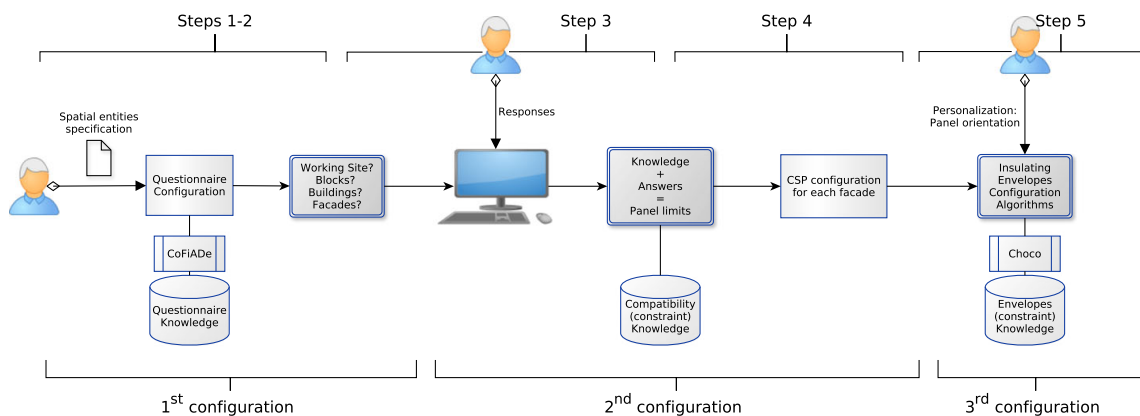
**Fig. 1** Façade and layout configuration examples

panels and envelopes, (3) to rank the different solutions based on several criteria and (4) to present them to the person in charge of the renovation, such as an architect, also called “user” in the rest of the paper.

The aim of this article is to present a decision support system for the configuration of panels-made envelopes in the context of building thermal renovation. The knowledge extracted from different stakeholders (building owner, architect, panels’ manufacturers, etc) have been stated as requirements and then mapped into the mathematical framework of constraint satisfaction problems (CSPs). Constraint satisfaction is used because its declarative model allows a straightforward and clear knowledge representation, and it is well-suited for configuration problems (Junker 2006). Also, the development of a prototype is greatly enhanced by the state-of-the-art constraint-based tools. Finally, the monotonic properties of CSP make it possible to add requirements (constraints) from every stage of the renovation process to create a complete and sound model of the problem.

The support system here introduced is a web-oriented application where a web-service architecture allows to solve all configuration tasks within a five-step renovation process, presented in Fig. 2:

- The first two steps supports the configuration of a questionnaire relating to the renovation model needed in the renovation process and used to determine limits for panels’ size and weight for each façade.



**Fig. 2** Configuration process overview

- The two following steps support the configuration of a constraint satisfaction problem for each of the façades to be renovated with specific parameters coming from the previous steps.
- The fifth step supports the configuration of façades insulating envelopes.

The article focuses only on the first four steps supporting two configuration tasks: questionnaire and CSP model. The system couples two constraint-based technological tools, CoFiADe (Vareilles et al. 2012) and Choco (Prud'homme and Fages 2013), to gain efficiency and modularity.<sup>1</sup>

The article is divided as follows. In Section 2, some related work on constraint satisfaction and product configuration are presented. In Section 3, the context, the problem description and the renovation process are discussed in detail. Then, in Section 4, the configuration tasks needed to support this massive renovation of buildings is introduced. Details of the underlying services in charge of the questionnaire configuration and of CSP instances configuration for each façade are presented in Section 5. In Section 6, the benefits of the support system, under the constraint satisfaction framework, are discussed. Finally, conclusions are drawn in Section 7. Relevant screen-shots of the support system interfaces have been chosen for illustrative purpose.

## 2 Related work

The framework of constraint satisfaction is used as it suits the constrained nature of product configuration (Junker 2006). On one hand, the knowledge (constraints) that restricts possible configuration of elements (variables) is easily modeled under the declarative framework of constraint satisfaction problems (Smith 2006). On the other hand, constraint-based configurators are able to present different solutions to users, often optimal, even when they do not provide all configuration parameters leaving their preferences unknown. Actually, CSP have been used in product configuration problems in such an extent that there is an entire chapter to product configuration in the *Handbook of Constraint Programming* (Junker 2006). The framework is known to solve efficiently packing problems (Beldiceanu et al. 2011) having, among other abstractions, the geometrical constraint GEOST (Beldiceanu et al. 2007). However, as we only deal with rectangular shapes, the constraint GEOST seems too complex for our need and would bring an unnecessary risk from a software maintenance point of view. In this section we present here a brief overview of the use of CSP for configuration problems, as well as some related support systems.

### 2.1 Constraint satisfaction & configuration

Despite the fact that CSP modeling is actually very simple and intuitive, product configuration problems involve a complexity that sometimes exceeds the capabilities of CSP framework. Thus, it was argued that CSP paradigm as original conceived was not well suited for addressing most of the configuration problems, due to the lack of mechanisms to handle variables that in some cases are relevant and in other cases are not. Mittal et al. in their seminal work, enhanced CSP with a dynamic view (Mittal and Falkenhainer 1990): constraint variables may be either activated or deactivated. In the first case variables take part of the

---

<sup>1</sup>A preliminary work on the system architecture has been presented in Barco et al. (2015b)

problem and hence in the solving process. The work started a cascade effect of research on product configuration and CSP. For instance, in Sabin and Freuder (1996), the authors developed Composite CSP, incorporating into CSP the possibility to model complex configuration relations such as *whole-part*, *is-a* and *part-of*. Thus, if a sub-problem variable is activated, all the problem is dynamically changed to handle all information in the problem and sub-problem. Among the same lines of work is Gelle and Weigel (1996) in which CSP is enhanced with the manipulation of continuous variables in order to address a wide range of real-life configuration problems. Studies for treating constraints as formulae and computable procedures using a constraint modeling tool (Xie et al. 2005) and to handle advanced structural relationships (Yang et al. 2012), are also found in the literature.

Along with these configuration capabilities, researchers have developed robust techniques on supply chain management thus improving the support for processes industrialization. For instance, CSP techniques are used in the elaboration of the planning and scheduling of the assembly process (Barták et al. 2010). It has been proved that assembly process for mass production is effectively and efficiently addressed by CSP given that most of the precedence relations can be expressed as constraints (e.g. first order formulae such as  $startX > startY + 20$ ) (Topaloglu et al. 2012; Peng et al. 2014). Also, there is a diverse set of CSP software (OpenRules, Inc 2013) implementing scheduling constraints (Gent and Walsh 1999) that deal with time slots and resources.

Given the efforts commented above, CSPs is now a mature technique to address combinatorial problems including configuration problems. More extensions and applications of CSP to product configuration have been proposed and are proposed constantly. However, regardless the considerable body of literature on configuration and constraint satisfaction, no generic solution is capable of addressing all configuration problems that rise in the industry. It is necessary, then, to develop a dedicated computer-based solution for the configuration problem of insulating envelopes.

## 2.2 Computer-based support

Regarding dedicated computer-based solutions for these kinds of industrial problems, there is no general approach to solve them all. Here we name a few tools implemented with different underlying models tackling similar problems. Most of these works are found in the field of *Space Planning* (also known as *Layout Synthesis*). Observe that these works, although supporting architects design of two-dimensional entities, do not pay attention to configuration or customization but rather to space planning problems.

In Shikder et al. (2010), authors present a prototype for the interactive layout configuration of apartment buildings including design information and an iterative design process. In Baykan and Fox (1992) is introduced Wright, a constraint-based layout generation system that exploits disjunctions of constraints to manage the possibilities on positioning two-dimensional objects in a two-dimensional space. Another system, Loos (Flemming 1990), is able to configure spaces using rectangles that cannot be overlapped but that may have holes. It uses test rules applied by steps to the rectangles in order to reach a good configuration based on its orientation and relation with other rectangles. The same authors have developed Seed (Flemming and Woodbury 1995): a system based on Loos used for early stages on architectural design. The system Hege1 (for Heuristic Generation of Layouts) is yet another space planning tool that simulates human design based on experimental cases (Akin et al. 1992). Finally, Medjdoub et al. present the system Archiplan which integrates geometrical and topological constraints to apartment layout planning (Medjdoub and Yannou 2000). These tools, however, are not well conceived for the massive configuration

of insulating envelopes. In fact, despite the great body of literature on configuration, works focusing on façades using any kind of configurable entity is scarce. Works relating façades and their configuration focus over predefined solutions and not in the actual composition or structure of the façade. For instance, in Teboul et al. (2010) authors use shape grammars and supervised classification to generate a segmentation of the façades in order to model buildings and understand patterns as part of urban planning. Among the same line of research we find in Wu et al. (2014) an approach based on inverse modeling to built knowledge on how façade configuration is generated and how elements are related. Also, research has been done using the available partial structure of some buildings to determine the total configuration of façades and thus enable image capturing software and automatic configuration design to capture urban reconstruction (Fan et al. 2014). None of these works intent to customize insulating envelopes w.r.t. the façade and thus we consider them interesting from the architectural point of view but unsuitable for the problem at hand.

### 3 Building renovation industrialization

The challenge of renovating buildings is currently been tackled in a French project called C.R.I.B.A, for its acronym in French of *Construction and Renovation in Industrialized Wood Steel* (Vareilles et al. 2013; Aldanondo et al. 2014): a joint effort between academics at École des Mines d'Albi - Carmaux and several French companies. Here, whereas academics work on the envelopes configuration problem and its performance impact, companies work on how to extract and formalize the information on buildings (gathered in a Building Information Model), how to design panels, how to manufacture them, how to transport them to the working site, how to install them on the façade and how to manage the renovation projects well. In this section we provided details of the renovation industrialization.

#### 3.1 Renovation industrialization process

To get a clear understanding about the limitations and constraints imposed on configuration by the renovation industrialization process, a global view of it is needed. The following six stages describe the renovation industrialization process.

Stage 1. Information collection. In this stage range-finder devices, along with pattern recognition software, is used to create a building information model (BIM) describing the building geometry (Aldanondo et al. 2014). In essence, the BIM includes the size of each façade on the building, the accurate size and position of frames (windows and doors) and any other element visually accessible to the devices.



#### **Limitation** *Façade size and panels' size*

Each façade is potentially different from the others. Thus, to be consistent, panels' size must be set according to the façade it will be allocated on (e.g. a given panel's width cannot be longer than the façade width it belongs to).

Stage 2. Semantic enrichment. Once the BIM has been built, crucial information has to be added with at least, the position and strength (load bearing capabilities) of specific areas





**Fig. 3** Façade: frames and load bearing areas

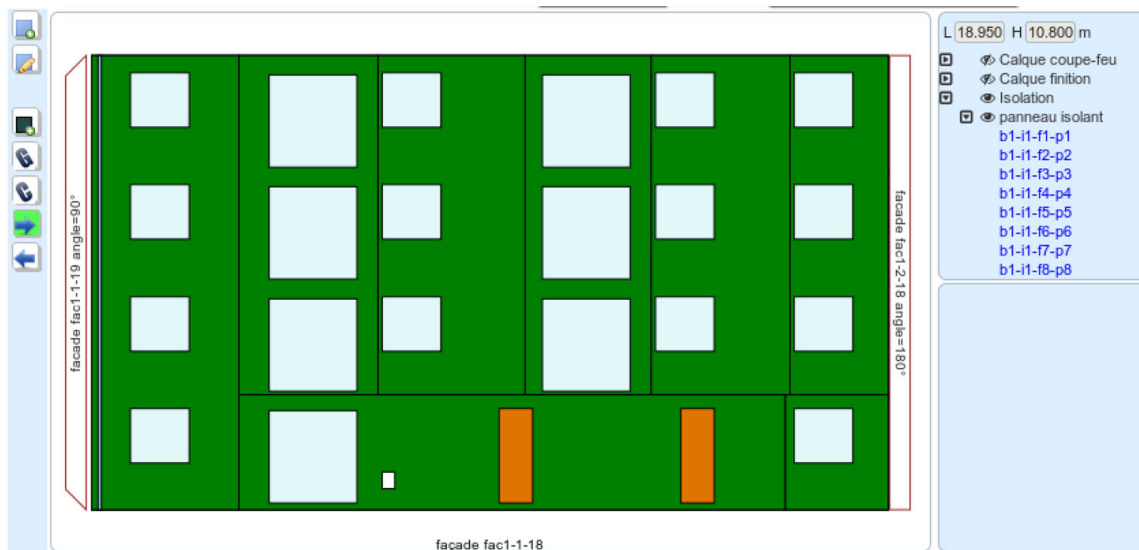
where panels can be attached and the frames to be removed from or added to the façade, as illustrated in Fig. 3.



**Limitation** *Load bearing capability and panel's limits*

Assuming that the bigger a panel the heavier, panels' size are limited by the load bearing capabilities of the façade they will be attached onto and frames that they have to cover.

Stage 3. Envelope configuration. Using the enriched BIM (geometry and load capabilities), the configuration of insulating envelopes is executed, as presented in Fig. 2 with a result example shown in Fig. 4. In this stage, each one of the façades to be renovated is



**Fig. 4** Façade: one insulating envelope configuration



synthesized using cutting and packing algorithms (Barco et al. 2014) and (Barco et al. 2015a).

Stage 4. Manufacturing. Once the insulating envelope has been configured, the panels are manufactured in the factories with respect to their specific size, frames, and with every detail for their allocations.



**Limitation** *Manufacturing and panels' size*

Panels' manufacturers have their own manufacturing limitations due to their plant layout or used machine tools. Thus, each manufacturer have to provide panels' width and height possible sizes: one bound indicating the minimum panel's size and another bound indicating the maximum panel's size. Panels' width and height can be linked by constraints in order to express more manufacturing limitations.

Stage 5. In this stage panels are shipped to the working site for their installation. Additionally, support for installing the panels is decided. This support include cranes, harnesses and any additional machinery needed by workers.



**Limitation** *Support and panels' size*

Panels' size may be limited by the available machinery. For instance, the size of panels is constrained w.r.t. the size of trucks available for their transportation (e.g., the smaller the trucks the smaller the panels).

Stage 6. On-site installation. The final stage is the installation of panels on each of the façades.



**Limitation** *Installation and panels' size*

Accessibility conditions may restrict the use of transportation, cranes or other supporting machinery. As a bandwagon effect, supporting machinery affects panels' size.

The aforementioned stages describe the global process of the renovation. All these limitations are concentrated in a single information system that allows to conserve consistency among the different stages. From a configuration viewpoint, limitation on each stage has an impact on the limits for panels' size and weight. The paper focuses on **Stage 3** of the renovation industrialization process, presented in Section 3.1: envelope configuration.

### 3.2 Façades & hierarchy

Although the core elements of the renovation are façades, panels and fasteners, all spatial entities are important for the configuration of panels, their manufacturing, shipment and installation. In this section how spatial entities, for instance a building, impose limitations over the configurable components, i.e., panels is described. This set of limitations generated by spatial entities is defined in the **Stage 1** and **Stage 2** of the renovation industrialization process, presented in Section 3.1. We begin with the definition of façades.

### 3.2.1 Façade

A façade is represented by a 2D coordinate plane, with origin of coordinates (0,0) at the bottom-left corner of the façade, containing rectangular zones defining:

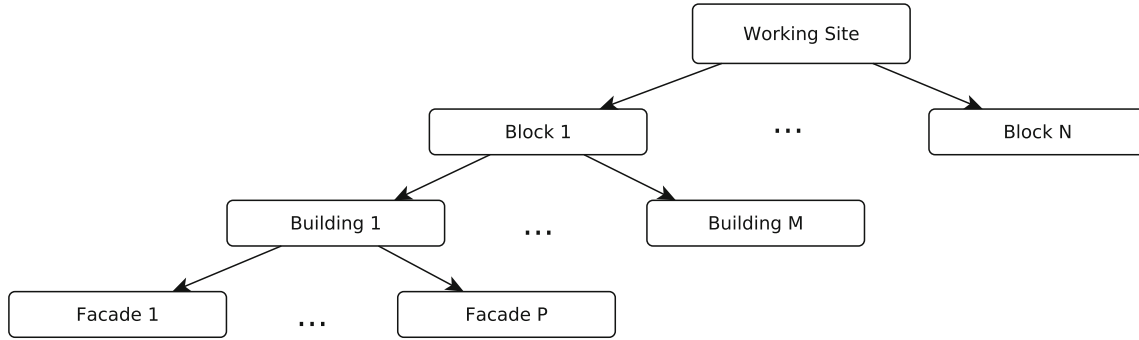
- Perimeter of façade with its dimensions (height and width in meters).
- Frames. Windows and doors over the façade play an important role as they are meant to be completely overlapped by panels. Frames are defined by:
  - Origin point with respect to the origin of a façade.
  - Width and height (in meters).
  - Constraint over position w.r.t. façade. To renovate the entire façade, frames borders should maintain a minimal distance to the façade borders.
  - Constraint over position w.r.t. panels. Given panels' internal structure, frames borders must keep a minimal distance to panels' borders.
- Supporting areas. As the layout problem must deal with a perpendicular space plan, gravity must be considered. It turns out that some areas over the façade have load bearing capabilities that allow panels to be attached. There may be any number of supporting areas each one of them providing (possibly different) load bearing capabilities. Supporting areas have well-defined:
  - Origin point with respect to the origin of a façade.
  - Width and height (in meters).
  - Load bearing capability (in  $\text{kg/m}^2$ ).
  - Constraint over positions of supporting areas. Façades are meant to be surrounded by supporting areas in order to be able to attach panels.

### 3.2.2 Façades habitats

Façades are not isolated structures. Spatial entities always belong to another bigger entity: a building to a block, a block to a neighborhood, a neighborhood to a city, a city to a country, etc. This *part-of* relation may be described as a hierarchical relation, the smaller part being lower in the hierarchy. This holds true in the renovation process for its spatial entities and is crucial for defining panels' limits. The hierarchy of the renovation is as follows. A façade is *part-of* a building, a building is *part-of* a block, a block is *part-of* a working site. An instance of such hierarchical structure is presented in Fig. 5.

Now, each node in Fig. 5 is described by environmental properties. These environmental properties, for now assumed to be independent between nodes, impose some limitations on the size and weight of panels. These environmental properties are mainly related to spatial entities accessibility and weather conditions.

The accessibility conditions refer to the surroundings circumstances that allow to get in and out of a spatial entity. For instance, a working site may have wide streets and gates, railways and even a seaport. The more possibilities and the more easy to access the spatial entity the less limitations it imposes over the panels. Let us consider two buildings *A* and *B* belonging to the working site *W* with the following accessibility conditions. The building *A* has big dual carriageway whereas the building *B* has a small one-way road. In addition, *A* has no trees in its surroundings whereas *B* has one garden filled with many centuries-old trees in its proximity. Clearly, the building *B* is restricted in the size and number of trucks that can access it as well as the kind of supporting machinery to attach panels. As expected, the smaller the trucks the smaller the panels.



**Fig. 5** Hierarchical view of renovation

At the other end of the spectrum are the weather conditions. These conditions refer to the climatological circumstances in which the whole renovation takes part. From these circumstances the wind and the season are taken into account. On the one hand, if a given working site is subject to strong wind then either panels must be specially attached or their size must be further limited (reduced) due to the installation process. In this particular renovation the panels' size are limited. Conversely, if the season in which the on-site work takes place is winter, then it is physically harder for working, as well as technically harder, to move and install bigger panels than small ones. Moreover, the cost of the renovation may increase because more workforce is needed or because non-working days may come up.

### 3.2.3 Cascade effect

Spatial entities inherit environmental conditions of spatial entities on superior levels of the hierarchy. This fact is decisive for setting the panels' limits for a given façade, which is at the bottom of the hierarchy.

Let us consider the same buildings *A* and *B* as in Section 3.2.2. Suppose three accessibility conditions levels (*easy*, *medium*, *hard*), three types of trucks (*big*, *medium*, *small*) and three types of panels' size (*big*, *medium*, *small*). Now let us instantiate the accessibility of the working site *W* to *medium*. This valuation implies that only *medium* and *small* trucks carrying *medium* and *small* panels can access the working site *W*. As the access to the buildings *A* and *B* is conditioned by the access to *W* as they are *part-of W*, their types of trucks and panels' sizes are therefore limited to *medium* and *small* for trucks and *medium* and *small* for panels. This property of inheritance is a monotonic operation, i.e., it can only reduced the possibilities.

Then, the accessibility of the building *A* and building *B* can be specified. Let us instantiate the accessibility of *A* to *easy*, meaning that focusing only on the building, every type of trucks carrying every size of panels are allowed. This valuation has no impact on the type of trucks neither panels' sizes available for *A*, still *medium* and *small* for trucks and *medium* and *small* for panels. However, if the accessibility of *B* is valuated to *hard*, this valuation implies that only *small* trucks carrying *small* panels can access it. This is a rather logical consequence that is straightforward modeled by CSP under entailment relations.

## 3.3 Configurable components

### 3.3.1 Rectangular panels

Panels are rectangular, of varying sizes and may include different equipment (frames, shutters, solar modules, etc.). These panels are designed one at a time in the process of layout

configuration, **Stage 3** of the renovation industrialization process presented in Section 3.1, and manufactured in the factory prior to shipment and installation on the building site. These panels have a well-defined:

- Size (height  $p_h$  and width  $p_w$  in meters).
- Constraint over height  $p_h$  and width  $p_w$  by mathematical formulae in order to express manufacturing limits.
- Constraint over dimensions by given lower and upper bounds consequence of environmental and installation limitations.
- Orientation: if the ratio  $\frac{p_w}{p_h}$  is less than one, the panel is vertical, otherwise it is horizontal.
- Thickness and insulation type chosen for the whole renovation.
- Set of frames with accurate size and position with respect to the bottom-left corner of the panel.
- Constraint over frames borders. Given internal structure of rectangular panels, frames must respect a minimum distance w.r.t. panels' borders.
- Weight  $p_{we}$  (in  $kg$ ) depending on several factors: size ( $p_w, p_h$ ), thickness, insulation type and frames. Up-to-now, we compute a panel's weight with

$$p_{we} = (p_w \times p_h) \times \alpha \quad (1)$$

where  $\alpha$  ( $kg.m^{-2}$ ) is the weight of  $1m^2$  of panels taking into account panels' material, internal structure, chosen insulation thickness and type, and frames.

- Cost (in €) depending mainly on size ( $p_w, p_h$ ) and frames. The cost  $p_c$  of a given panel  $p$  is computed with the next formula

$$p_c = (p_w \times p_h) * \beta \quad (2)$$

where  $\beta$  ( $€.m^{-2}$ ) is the cost of  $1m^2$  of panels taking into account panels' material, internal structure, chosen thickness and insulation type and frames.

- Thermal performance (in  $w.m^{-2}.k^{-1}$ ) depending on size ( $p_w, p_h$ ), thickness ( $p_t$ ) and insulation type ( $p_{is}$ ).

In the remaining of the article it is assumed a constant panels' thickness and a unique type of insulation. Considering that all panels belonging to the same façade-layout plan have the same thickness and insulation type, and in order to have a good thermal insulation, the less panels' junctions the better: it is at panels' junctions that it exists thermal transfers. In consequence, façades should wear the minimum number of panels, each one of them as large as possible while respecting the architectural constraints, supporting areas, manufacturing, installation and accessibility limitations. Well configured panels meet the following criteria:

- a) Cover the greatest possible area given the geometric position of frames and supporting areas,
- b) no overlapping with any other panel and no holes allowed,
- c) no interference with the definition and placement of other panels.

### 3.3.2 Fasteners

Fasteners are metallic devices used to attach panels over the façade. These elements consist of two parts: one fixed directly onto the façade (support bracket) and one installed on the panel at the factory (panel's bracket). At the level of the panels, fasteners are fixed on three possible locations: along the bottom edge of the panel, along the top edge of the panel or

along the lateral edges of the panel. The position is determined with respect to panels' size and weight (e.g. if a panel's height is bigger than its width then more stability is achieved attaching the panel along the lateral edges). Ergo, defining in which of these edges panels are attached, as well as the number of fasteners, is an output of the configuration.

## 4 Support system configuration tasks

The main configuration need of the industrial case is the configuration of insulating envelopes, last step of **Stage 3** of the renovation industrialization process, presented in Section 3.1: envelope configuration. To achieve façades configuration, two prerequisite configuration tasks must be performed: (1) the configuration of a questionnaire to be filled by the user and (2) the configuration of a constraint satisfaction model for each façade to be renovated. In this section we discuss these two prerequisite tasks. To do so, a description of the user-system interaction is presented.

### 4.1 Configuration process (Stage 3)

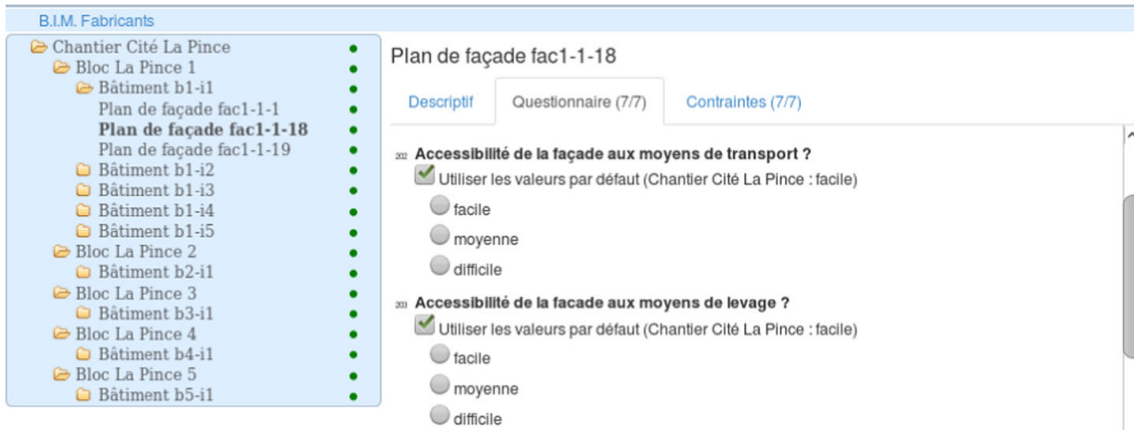
Information about spatial entities is acquired by the support system by means of an input file describing all geometrical and structural properties, and by means of a web-based questionnaire for each spatial entity in the input file. After questionnaire completion, the lower bound and upper bound of panels' size and weight are deduced. Also, given that several instances of façades have to be solved once the questioning stage over, the systems creates a constraint satisfaction model for each façade using the information in the input file and the deduced limits for panels' size and weight. Here, each constraint satisfaction model instance is parameterized according to the façade information (e.g. environmental conditions) and the particular deduced panels' limits.

Let us consider the information flow from the user perspective, accompanied by relevant GUI screen-shots.

- Step 1:- The user uploads a file containing the geometry and structural specification of spatial entities. This file comes from **Stage 1** and **Stage 2** of the renovation industrialization process, presented in Section 3.1. Information is stored in a data base.
- Step 2:- The first service builds and presents a questionnaire to the user for each of the spatial entities in the input file.
- Step 3:- The user answers the questions as much as possible (leaving blanks for the questions he cannot answer).
- Step 4:- Using the information about spatial entities (database) and their manufacturing/installation/environmental conditions (user answers), the system deduces panels' lower and upper bounds for width, height and weight for each façade (parametrization of each CSP). The user has the possibility to overwrite these bounds within the deduced limits.
- Step 5:- The user invokes the second service for insulating envelopes configuration.

**Manual.** The user draws each panel one at a time on the client GUI. Each panel is assured to be consistent with the problem requirements and CSP model by sending its information to validate into the service.

**Semi-automatic.** The user draws some well defined panels and then invokes the service to finish the configuration.



**Fig. 6** Questionnaire: working site questions

**Automatic.** The user invokes the service to provide compliant insulating envelopes respecting all stakeholders requirements.

## 4.2 Questionnaire configuration

Once the input file has been processed by the support system, it can proceed by configuring a set of questions for each spatial entity in the file, Step 2 of the configuration process presented in Section 4.1 and illustrated in Fig. 6. Then, after the user has completed the questionnaires (end of Step 3), the system configures, i.e., deduces, the limits of panels' size and weight for each façade. The questionnaire inquires the following information.

**Working site.** This is the bigger spatial entity of the renovation.

1. Values provided by the user are:
  - Is the working site in a windy region? {yes, no}
  - When does the on-site work take place? {summer, fall, winter, spring}
  - What is the cost target? Euros.
  - What is the performance target?  $w.m^{-2}.k^{-1}$
  - Is there some obstacles? {yes, no}
  - How is the accessibility? {easy, medium, hard}
2. Parameters deduced by the first filtering service which can be overwritten by the user:
  - Panels' width ( $w_{ws}$ ) and height ( $h_{ws}$ ) lower bound.
  - Panels' width ( $\overline{w_{ws}}$ ) and height ( $\overline{h_{ws}}$ ) upper bound.
  - Panels' maximum weight ( $\overline{we_{ws}}$ )

**Block.** A block is a set of buildings which are usually attached by a common wall.

1. Values provided by the user are:
  - Is there some obstacles? {yes, no}
  - How is the accessibility? {easy, medium, hard}
2. Parameters deduced by the first filtering service which can be overwritten by the user:

- Panels' width ( $\underline{w}_{bl}$ ) and height ( $\underline{h}_{bl}$ ) lower bound.  
 $\underline{w}_{bl} \in [\underline{w}_{ws}, \overline{w}_{ws}]$  and  $\underline{h}_{bl} \in [\underline{h}_{ws}, \overline{h}_{ws}]$
- Panels' width ( $\overline{w}_{bl}$ ) and height ( $\overline{h}_{bl}$ ) upper bound.  
 $\overline{w}_{bl} \in [\underline{w}_{ws}, \overline{w}_{ws}]$  and  $\overline{h}_{bl} \in [\underline{h}_{ws}, \overline{h}_{ws}]$
- Panels' maximum weight ( $\overline{we}_{bl}$ )  
 $\overline{we}_{bl} \leq \overline{we}_{ws}$

**Building.** A building is the spatial entity where apartments are arranged and is the host of several façades.

1. Values provided by the user are:
  - Is there some obstacles ? {yes, no}
  - How is the accessibility ? {easy, medium, hard}
2. Parameters deduced by the first filtering service which can be overwritten by the user:
  - Panels' width ( $\underline{w}_{bg}$ ) and height ( $\underline{h}_{bg}$ ) lower bound.  
 $\underline{w}_{bg} \in [\underline{w}_{bl}, \overline{w}_{bl}]$  and  $\underline{h}_{bg} \in [\underline{h}_{bl}, \overline{h}_{bl}]$
  - Panels' width ( $\overline{w}_{bg}$ ) and height ( $\overline{h}_{bg}$ ) upper bound.  
 $\overline{w}_{bg} \in [\underline{w}_{bl}, \overline{w}_{bl}]$  and  $\overline{h}_{bg} \in [\underline{h}_{bl}, \overline{h}_{bl}]$
  - Panels' maximum weight ( $\overline{we}_{bg}$ )  
 $\overline{we}_{bg} \leq \overline{we}_{bl}$

**Façade.** A façade is a composition of apartments along with frames.

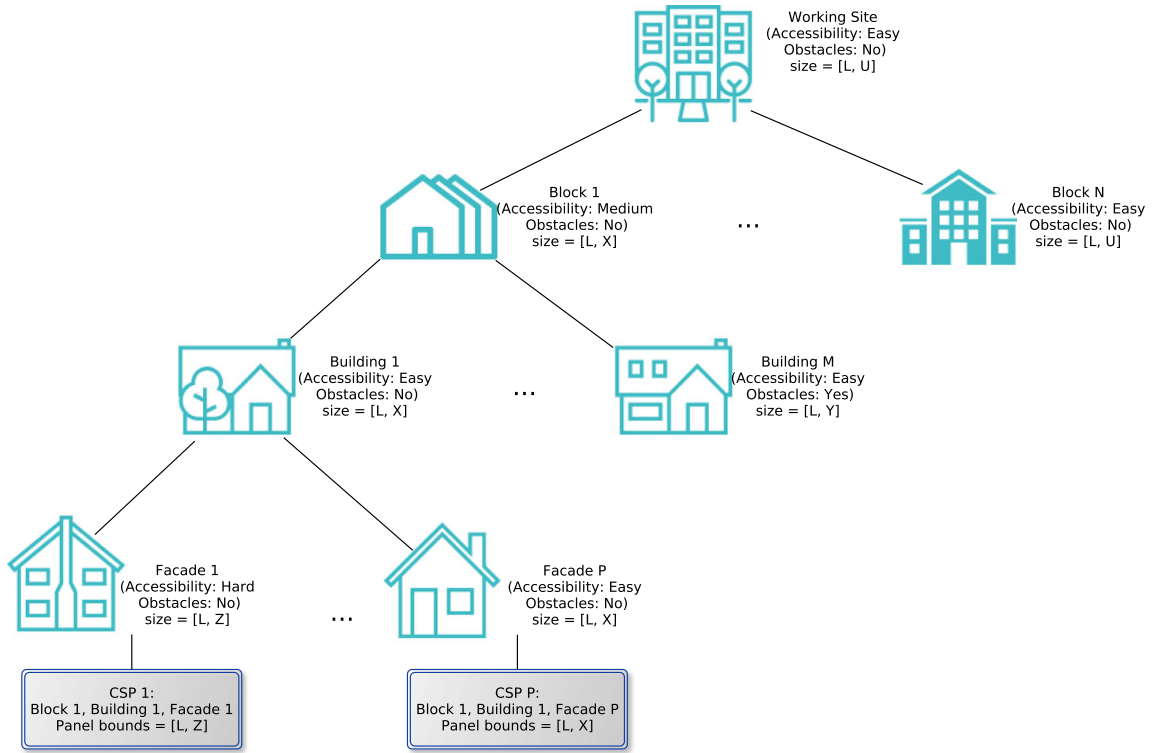
1. Values provided by the user are:
  - Is there some obstacles ? {yes, no}
  - How is the accessibility ? {easy, medium, hard}
2. Parameters deduced by the first filtering service which can be overwritten by the user:
  - Panels' width ( $\underline{w}_{fc}$ ) and height ( $\underline{h}_{fc}$ ) lower bound.  
 $\underline{w}_{fc} \in [\underline{w}_{bg}, \overline{w}_{bg}]$  and  $\underline{h}_{fc} \in [\underline{h}_{bg}, \overline{h}_{bg}]$
  - Panels' width ( $\overline{w}_{fc}$ ) and height ( $\overline{h}_{fc}$ ) upper bound.  
 $\overline{w}_{fc} \in [\underline{w}_{bg}, \overline{w}_{bg}]$  and  $\overline{h}_{fc} \in [\underline{h}_{bg}, \overline{h}_{bg}]$
  - Panels' maximum weight ( $\overline{we}_{fc}$ )  
 $\overline{we}_{fc} \leq \overline{we}_{bg}$

This information collection has two specific goals: (1) to provide details about renovation aspects, such as the performance target, that are needed in the configuration process and (2) to provide upper bound for panels' size and weight for each level of spatial entities.

### 4.3 One Façade, one CSP-based configuration problem

One critical aspect of the support system is the ability to configure a constraint satisfaction model for each façade to renovate. Each façade has (potentially) different size, number of windows, supporting areas, etc. It is important from the point of view of the configuration to assign valid values to panels, for instance, possible positions for panels must lie between





**Fig. 7** Downwards consistency among entities

zero and the façade width and height. Simply put, each façade has its own configuration parameters used in the constraint satisfaction model and in the envelope configuration process. Also, each façade may have different accessibility conditions, obstacles or even user preferences.

When configuring these CSP instances (Step 4 of the configuration process presented in Section 4.1), it is important to conserve downwards consistency. Downwards consistency refers to the fact that information on higher level of the spatial entities hierarchy is propagated to the inferior levels, i.e., working site  $\rightarrow$  blocks  $\rightarrow$  buildings  $\rightarrow$  façades, but it cannot be propagated upwards. As an example consider only accessibility conditions, obstacles presence and panels' size limits, for the specification in Fig. 7.

Note that lower spatial entities on the hierarchy inherit values from upper levels. In Fig. 7, for instance, façade 1 has an accessibility condition valuated to *hard* and thus the upper bound for panels' size is reduced to a given  $Z$ . This upper bound is not propagated upwards to the building 1; it conserves its inherited value  $X$ . Consequently, façade P inherits the value of  $X$  from the building 1 as no further reduction is needed for their panels' configuration. Naturally, it is the case that  $Z < X < U$ . Using this information a CSP is configured for each façade to be renovated. Note that monotonic properties of constraint satisfaction framework make transparent this configuration process.

#### 4.4 Personalization and recommendation

In order to improve the configuration capabilities of the system, it is useful to allow the personalization of the insulating envelopes. In this section we present the personalization (preferences) set by the user, the features that are crucial for the configuration and the features that may be excluded or completed by the system (our analysis is based on Falkner

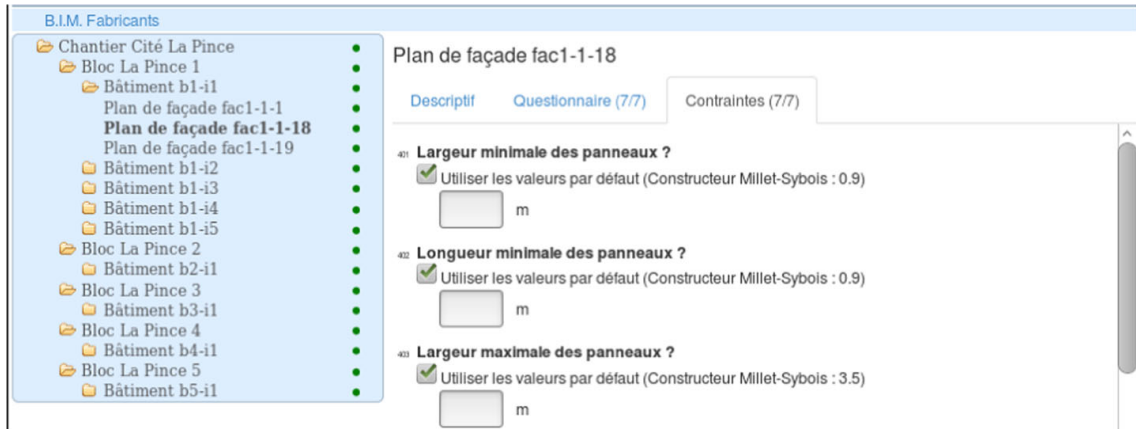


Fig. 8 Questionnaire: overwriting of panels' limits

et al. 2011). Although more work in this sense has to be carried out (discussed as future work), this section provides a glance of how the configuration system may be promoted into a recommender system.

#### 4.4.1 User preferences

As indicated in Section 3.1, in **Stage 3** of the renovation industrialization process, the user has the possibility to overwrite panels' limits within those deduced by the system, as presented in Fig. 8. For instance, the user can enforce panels to be square ( $p_w = p_h$ ) in order to give the façade a specific look. This first feature for personalization allows the user to generate more solution diversity (defined as different solutions Schreiber 2010).

The second feature for personalization of the configuration depends on panels' orientation. Panels' orientation is only a relation between panels' width  $p_w$  and height  $p_h$ , being horizontally oriented when  $p_w > p_h$  and vertically oriented otherwise. This is an important personalization feature as it will affect the aesthetics properties of the envelopes.<sup>2</sup> Nonetheless, the preferred panels' orientation may be not respected due to conflicts in the (constraint) knowledge model; it is therefore considered as a soft-constraint. When the preferred orientation cannot be satisfied, the system recommends to the user the solutions composed of the maximum number of panels in the preferred orientation; an application of the *nearer-is-better* similarity function (Falkner et al. 2011). Choosing a minimum number of panels to be vertical (respectively horizontal) is a personalization feature considered as future work (Section 7).

#### 4.4.2 Features ranking and exclusion

To simplify the process from the user viewpoint, we have selected a set of key features to the configuration as well as the optional ones which can be left to the system. The idea behind this ranking is to avoid overwhelming the final user with the process (Tiihonen and Felfernig 2010). Two of the features are key for the configuration, meaning that they are the most highly ranked (Falkner et al. 2011):

<sup>2</sup>Aesthetics property is not considered in the model or process as this kind of knowledge is hard to formalize and ergo it is out of scope of the present article.

- Orientation: it is important to achieve a pleasant aesthetic envelope. If this feature is not set, the system recommends envelopes configured using only one panel's orientation. Then, it tries to configure envelopes using only vertically or only horizontally oriented panels.
- Panels' size bounds: the limits for the panels' size are mandatory and have to take into account all the renovation industrialization process limitations. Forgetting one of them (manufacturing, shipping or installing limitations) leads to envelopes configuration which cannot be either manufactured, shipped or installed onto the façades.

At the other end of the spectrum are the optional features which can be left in blank or be filled by the support system. In the renovation process, these features concern mainly information that has an impact on the configuration:

- Working site in windy region and season of on-site work: default values are *no wind* in the region and *summer* for on-site work which have no impact on panels' limits.
- Target cost and target performance: ideally, the configuration should respect the target cost and performance. However, if no targets are set, the system generates solutions with minimal number of panels which entail minimum cost and maximum thermal performance.
- Obstacles presence and accessibility conditions: these properties are inherited from the working site information and propagated downwards. If no information is available, default values are *no obstacles* and *easy access* which have no impact on panels' limits.
- Fasteners location: the location of fasteners (bottom, lateral edges, top) is another preference (soft constraint). This is however bypassed by the system if the size of panels demands it. For instance, the system recommends *bottom* fasteners for horizontal panels and *lateral* fasteners for vertical ones.

## 5 Implementing configuration tasks

As commented before, different configuration tasks are assigned to different web-services. The information inputted into the questionnaire is processed by the first service called *Filtering Service*. Once the first service has processed the information, the user invokes the second one to provide compliant envelope configurations. The second service is then called *Solving Service*. The architecture of the on-line support system is presented in Fig. 9.

In order to give a clear understanding on how the system works, let us describe its input and output in a formal way. For each of the services the input is a tuple of the form  $\langle SPEC, \mathcal{V}, \mathcal{D}(\mathcal{V}), \mathcal{C}(\mathcal{V}) \rangle$  with  $|\mathcal{V}| = |\mathcal{D}(\mathcal{V})|$  and

- $SPEC = \langle WS, BK, BG, FAC \rangle$ ; *WS* variables describing the working site, *BK* variables describing blocks, *BG* variables describing buildings and *FAC* variables describing façades.
- $\mathcal{V} = \langle \mathcal{P}, \mathcal{FA} \rangle$ ; *P* variables describing (from the manufacturing viewpoint) a single panel and *FA* variables describing a single fastener.
- $\mathcal{D}(\mathcal{V}) = \langle \mathcal{D}(\mathcal{P}), \mathcal{D}(\mathcal{FA}), \dots \rangle$ ; domain for each one of the variables in  $\mathcal{V}$ .
- $\mathcal{C}(\mathcal{V})$  a set of constraints over variables in  $\mathcal{V}$ .

Information in *SPEC* describes only properties of the spatial entities such as the number of façades, sizes, positions, etc. Variables in  $\mathcal{V}$  and  $\mathcal{D}(\mathcal{V})$  are manufacturer dependent and include the initial panels' domains which depends on the manufacturing process.

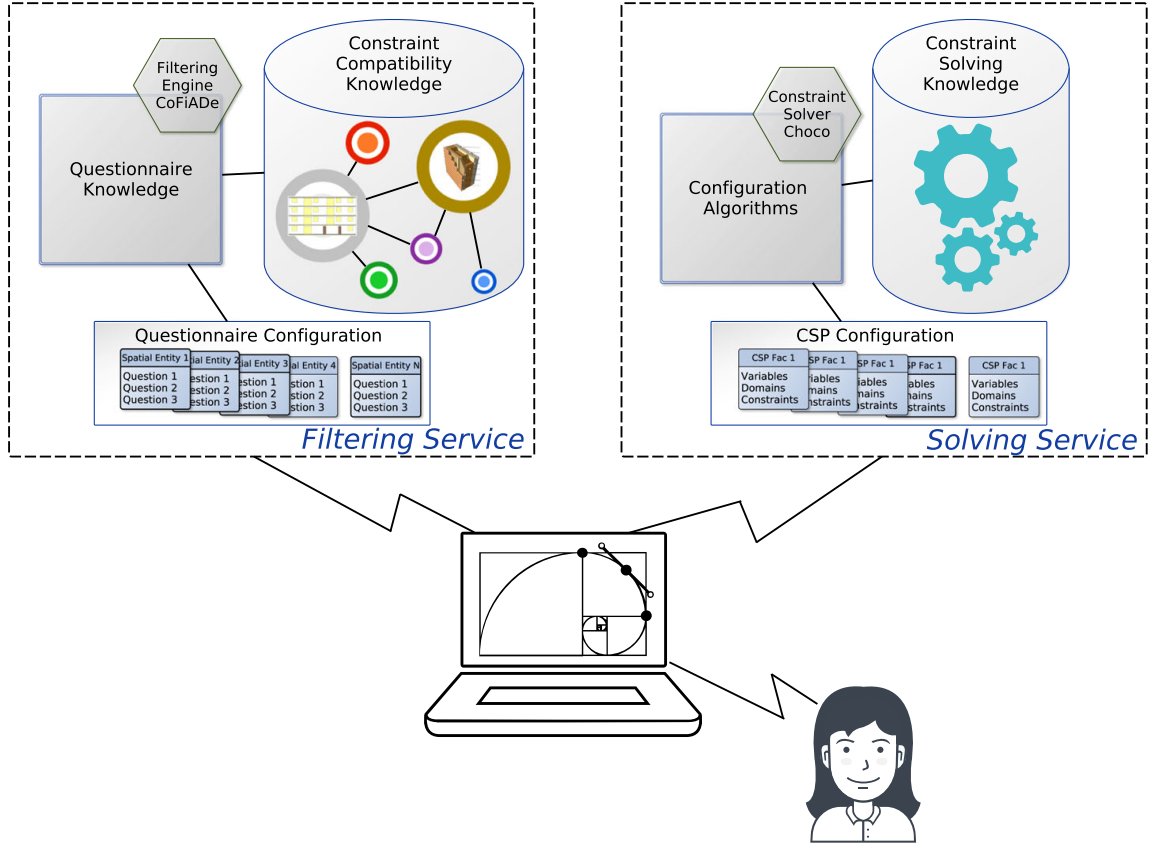


Fig. 9 Service-based architecture for on-line configurator

Constraint in  $\mathcal{C}(\mathcal{V})$  are extracted from the problem domain by stakeholders and, as shown in Sections 5.1 and 5.2, these constraints are different in each service.

## 5.1 Customization with a questionnaire

In this section, details of the filtering service in charge of the questionnaire configuration is introduced. The filtering service exploits the filtering engine CoFiADe whose implementation is based on compatibility tables, i.e., constraints expressed by enumerating all possible valid compatibility of values (these kinds of constraints can also be found, for instance, in the finite domain constraint module of SWI-Prolog Wielemaker et al. 2010; Triska 2012).

### 5.1.1 Mapping

The filtering service is in charge of removing values from elements in  $\mathcal{D}(\mathcal{V})$  that are not allowed by the established constraints. Here, constraints  $\mathcal{C}(\mathcal{V})$  describe valid combination among different arguments in  $\mathcal{SPEC}$  and variables in  $\mathcal{D}(\mathcal{V})$ . We denote this set of constraints  $\mathcal{C}_f(\mathcal{V})$  to distinguish them from the ones used in the solving service. These constraints are formalized as compatibility tables (presented in next section). Formally, the filtering is a mapping  $\mathcal{M}$  from variables and domains to domains

$$\mathcal{M}(\mathcal{SPEC}, \mathcal{V}, \mathcal{D}(\mathcal{V}), \mathcal{C}_f(\mathcal{V})) \rightarrow \mathcal{D}'(\mathcal{V}) \quad (3)$$

The result  $\mathcal{D}'(\mathcal{V})$  contains the new domain for panels and fasteners, where  $\mathcal{D}'(\mathcal{V}) \subseteq \mathcal{D}(\mathcal{V})$ .

As stated previously in Step 4 of the configuration process presented in Section 4.1, the goal of initial filtering is to set domains for configurable components using spatial entities information and constraints to do so. In our on-line support system we use the filtering engine CoFiADe (Vareilles et al. 2012) to perform this filtering. Several reasons support our choice. First, the system is already on-line, making it usable in no time. Second, it is well conceived for supporting decision-making processes. And third, it uses efficient compatibility tables for domain pruning; applying a given compatibility table is made in constant time  $\mathcal{O}(1)$ .

### 5.1.2 Compatibility knowledge

Configurable components of the renovation are panels and fasteners to attach panels. Panels are configurable by fixing their width, height, weight and position over the façade. Fasteners are configurable by determining their number and setting their type {bottom, top, lateral}.

The following material describes the compatibility tables, presented in Table 1, which state the allowed combinations between the user's input values and configurable components values.

- C1 Relation between environmental conditions of spatial entities and panels' size, where  $WW_u$  and  $WH_u$  are upper-bounds for panels' width and height, respectively, when constrained by environmental conditions
- C2 Relation season that on-site work will take place and panels' size, where  $SW_u$  and  $SH_u$  are upper-bounds for panels' width and height, respectively, when constrained by the season.
- C3 Relation between obstacles in spatial entities and panels' size, where  $OW_u$  and  $OH_u$  are upper-bounds for panels' width and height, respectively, when constrained by the presence of obstacles.

**Table 1** Compatibility tables on filtering service

$C_1$		$C_2$	
Wind	Panels' size	Season	Panels' size
yes	$(w_p \leq WW_u) \wedge (h_p \leq WH_u)$	summer $\vee$ spring	$\emptyset$
no	$\emptyset$	fall $\vee$ winter	$(w_p \leq SW_u) \wedge (h_p \leq SH_u)$

$C_3$		$C_4$	
Obstacles	Panels' size	Accessibility	Panels' dimensions
yes	$(w_p \leq OW_u) \wedge (h_p \leq OH_u)$	easy	$\emptyset$
no	$\emptyset$	medium	$(w_p \leq AW_{u1}) \wedge (h_p \leq AH_{u1})$
		hard	$(w_p \leq AW_{u2}) \wedge (h_p \leq AH_{u2})$

$C_5$		$C_6$	
Cost	Insulation	Performance	Insulation
$< 50000$	low	$< 25$	high
$[50000, 100000]$	low, medium	$[25, 50]$	medium, high
$\geq 100000$	low, medium, high	$\geq 50$	low, medium, high

$C_7$		$C_8$	
Weight	Position Fasteners	Position fasteners	# fasteners
$< 500$	{top,bottom,laterals}	{top,bottom}	$[2,4,6]$
$[500,1000]$	{top,bottom}	laterals	$[4,6]$
$> 1000$	bottom		

- C4 Relation between accessibility of spatial entities and panels' size, where  $AW_{u1}$  and  $AH_{u1}$  are upper-bounds for panels' width and height, respectively, when constrained by *medium* level accessibility conditions, and  $AW_{u2}$  and  $AH_{u2}$  are upper-bounds for panels' width and height, respectively, when constrained by *hard* level accessibility conditions.
- C5 Relation between renovation cost and panels' insulation: it reflects the fact that the thickness of the insulation depends on the user budget.
- C6 Relation between desired performance and panels' insulation: it shows the fact that the thickness of the insulation depends on the desired final energetic performance.
- C7 Relation between panels' weight and fasteners positions.
- C8 Relation between fasteners position and number of fasteners.

## 5.2 Customization with a constraint solver

As stated in Step 5 of the configuration process presented in Section 4.1, the solving service configures a CSP given the façade specifications and uses deduced panels' limits to set panels in valid domains. Afterwards, the configuration of envelopes is executed thanks to cutting and packing algorithms. The solving process exploits the constraint solver Choco whose implementation is based on propagators, i.e., dedicated filtering algorithms (these kinds of constraints can be found, for instance, in the constraint programming environments Gecode (Schulte et al. 2010) and ECLiPSe (Schimpf and Shen 2012)).

### 5.2.1 Envelopes configuration

The second service in the support system is in charge of layout-plans configuration. The system uses several algorithms to generate layout plans in an automatic, manual or semi-manual way.

Now, while information of  $SPEC$  and  $\mathcal{V}$  are the same as the filtering services, it is not the case for domains and constraints. To differentiate them let us call the input domains  $\mathcal{D}_s(\mathcal{V})$  and the constraints  $\mathcal{C}_s(\mathcal{V})$ . Intuitively, variable domains  $\mathcal{D}_s(\mathcal{V})$  are provided by the mapping of the filtering service, i.e.,

$$\mathcal{M}(SPEC, \mathcal{V}, \mathcal{D}(\mathcal{V}), \mathcal{C}_f(\mathcal{V})) = \mathcal{D}'(\mathcal{V}) = \mathcal{D}_s(\mathcal{V}) \quad (4)$$

where  $\mathcal{D}(\mathcal{V})$  is the initial domain for panels' size and weight. Constraints in  $\mathcal{C}_s(\mathcal{V})$  are stated as first order formulas and express, not compatibility among elements but, requirements for valid layout plans (see next section for a description of these constraints). All resolution approaches implemented in the solving service respect all constraints of the model.

The output of the solving service is a set of layout-plan solutions, each one characterized by a cost and energetic performance. Formally, this service is a function of the form

$$\mathcal{F}(SPEC, \mathcal{V}, \mathcal{D}_s(\mathcal{V}), \mathcal{C}_s(\mathcal{V}), \mathcal{H}) = \langle \mathcal{PX}, \mathcal{PY}, \mathcal{PDX}, \mathcal{PDY}, \mathcal{FX}, \mathcal{FY} \rangle \quad (5)$$

where  $\mathcal{PX}$  and  $\mathcal{PY}$  represent the origin of coordinates,  $\mathcal{PDX}$  and  $\mathcal{PDY}$  the width and height, respectively, for each panel in the solution, and  $\mathcal{FX}$  and  $\mathcal{FY}$  represent the position for each fastener. Additionally, the function is parameterized by an heuristic  $\mathcal{H}$  stating which algorithm is meant to be used. Available algorithms are *greedy* (Barco et al. 2014), and *global constraints with backtrack search* (Barco et al. 2015a) using Choco solver (Prud'homme and Fages 2013).

### 5.2.2 Envelopes knowledge

Let  $F$  denote the set of frames and  $S$  the set of supporting areas. Let  $o_{e,d}$  and  $l_{e,d}$  denote the origin and length, respectively, of a given entity  $e$  in the dimension  $d$ , with  $d \in [1, 2]$ . For instance,  $o_{fr,1}$  denotes the origin in the horizontal axis and  $l_{fr,1}$  denotes the width of frame  $fr$ . Additionally,  $lb_d$  and  $ub_d$  denote the length lower bound and length upper bound, respectively, in dimension  $d$  for all panels.

Each panel is described by its origin point w.r.t. the façade origin and its size. Let us assume that  $\mathcal{P}$  is the set of panels composing the layout-plan solution. Then, each  $p \in \mathcal{P}$  is defined by  $\langle o, l \rangle$  where

- $o_{p,d} \in [0, o_{fac,d}]$  is the origin of panel  $p$  in dimension  $d$ .
- $l_{p,d} \in [lb_{p,d}, ub_{p,d}]$  is the length of panel  $p$  in dimension  $d$ .

The following six constraints express the relations that must respect a layout solution.

- (a) *Manufacturing and transportation limitations constrain panels' size with a give upper bound  $ub$  in one or both dimensions.*

$$\forall p \in \mathcal{P} l_{p,d} \leq ub_d$$

- (b) *(diffN) For two given panels  $p$  and  $q$  there is at least one dimension where their projections do not overlap.*

$$\forall p \in P, \forall q \in P, p \neq q, \exists d \in [1, 2] | o_{p,d} \geq o_{q,d} + l_{q,d} \vee o_{q,d} \geq o_{p,d} + l_{p,d}$$

- (c) *A given panel  $p$  must either be at the façade edge or ensure that enough space is left to fix another panel.*

$$\forall p \in P o_{p,d} + l_{p,d} \leq l_{fac,d} - lb_k \vee o_{p,d} + l_{p,d} = l_{fac,d}$$

- (d) *Each frame over the façade must be completely overlapped by one and only one panel. Additionally, frames borders and panels' borders must be separated by a minimum distance denoted by  $\Delta$ .*

$$\forall f \in F, \exists p \in P | o_{p,d} + \Delta \leq o_{f,d} \wedge o_{f,d} + l_{f,d} \leq o_{p,d} + l_{p,d} + \Delta$$

- (e) *The entire façade surface must be covered with panels.*

$$\sum_{i \in P} \prod_{d \in [1,2]} (o_{i,2} + l_{i,2}) = \prod_{d \in [1,2]} l_{fac,d}$$

- (f) *Panels' corners must be matched with supporting areas in order to be properly attached onto the façade.*

$$\forall p \in P, \exists s \in S | o_{s,d} \leq o_{p,d} \vee o_{p,d} + l_{p,d} \leq o_{s,d} + l_{s,d}$$

## 6 Discussion

### 6.1 Underlying handling

The division of configuration tasks and its handling by different services is supported by the underlying declarative model of constraint satisfaction. Indeed, the monotonic properties of constraint-based systems make it possible to collect the requirements from the problem domain, state them as constraint and map them into constraint-based implementations in a simple intuitive way. Additionally, as industrial problems tend to evolve, the monotonicity



of constraint satisfaction allows the adaptation of the model to new needs by the addition or deletion of constraints.

Now, note that the user's answers to the questionnaire are sent to the first service which replies with the panels' size bounds for a particular façade. This means that the clients' browser is the one capturing the panels' bounds. Then, upon user's request, the system sends the façades specifications and particular panels' bounds to the solving service which replies with different envelopes solutions. Therefore, there is no need for any intermediary procedures to map information from one service to the other; only values (answer to questions) are sent to the filtering service and only values (panels' bounds) are returned from it whereas only values (façade specification and panels' bounds) are sent to the solving service and only values (bill of material) are returned from it. After uploading a working site specification (steps 1–2), the configuration process is launched and all the information relevant to the working site is taken into account, as shown in Fig. 10. Then, the user has to answer a set of questions about the working site, each of its blocks, buildings and facades in order to limit the panels' size and weight bounds. The facade layout synthesis can now start, as illustrated on Fig. 11. The user has now the possibility either to draw her/him-self the facade layout by placinf panels one by one or to ask the system for an automatic solution, as shown in Fig. 12.

## 6.2 Advantages

Benefits for configuration tasks division into web-services are rather simple. On the one hand we apply the well-known principle *Divide and conquer*. In our on-line system this principle allows us to add or remove variables, domains and questions in the filtering service, i.e., by means of adding or removing compatibility tables. In addition, as we use CoFiADe, we may mix different variable representations as integer domains, continuous domains and symbolic domains whereas in most constraint systems mixing variable domains is not allowed or is not efficient enough. For instance, given the reduced number of constraints for continuous domains in Choco, the representation has to be changed to integer domains.

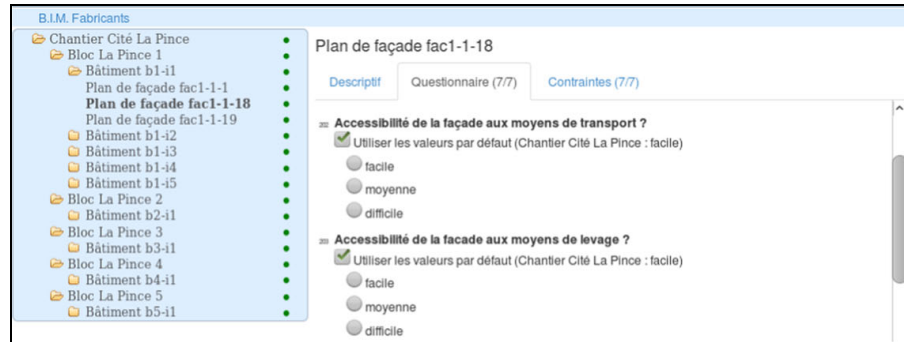
On the other hand, as a benefit of tasks division, we improve performance by avoiding the use of binary equalities and binary inequalities constraints whose computational time is  $\mathcal{O}(n * m)$ , where  $n$  and  $m$  are the number of values in the domain of the two variables involved in the constraint. Thus, at the moment of finding solutions, the

After opening a working site specification, every block, building and façade is parsed and shown in the interface.

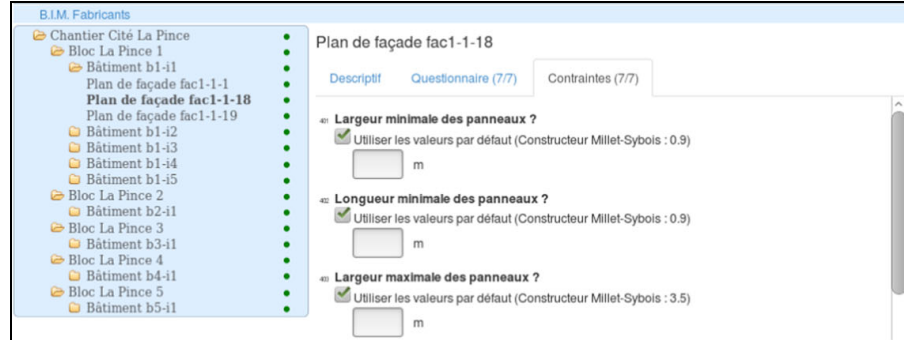


Fig. 10 Decision support system screen-shots (part 1)

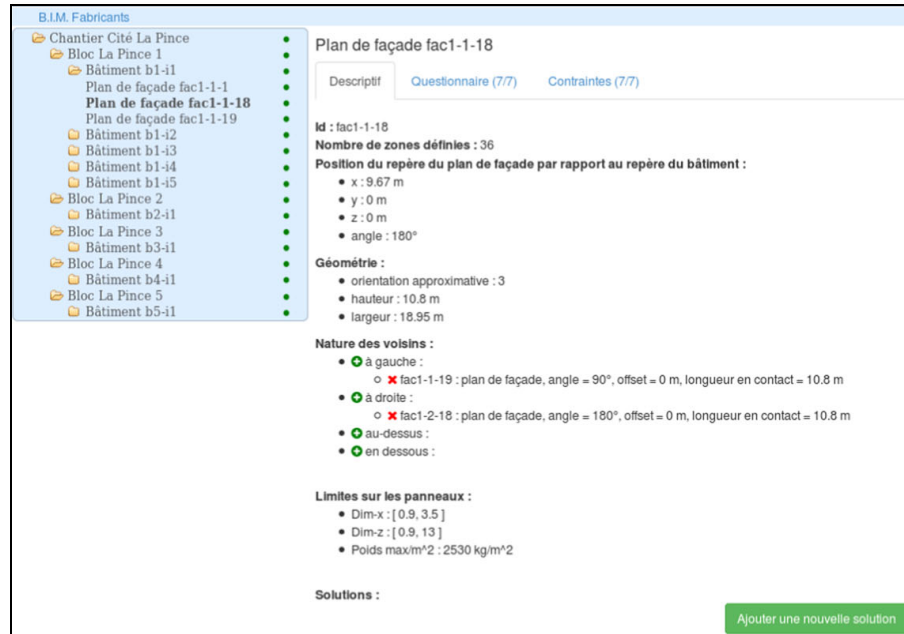
For every spatial entity the system presents a questionnaire about its accessibility conditions.



For every spatial entity the system presents a questionnaire about the panels' constraints (architects preferred limits).



When selecting a façade, the geometrical properties and the panels' bounds are shown. The "add solution" button is enabled.



At pressing the design button, a two-dimensional view of the façade is shown, along with the input/output information and the design tool-bar.



Fig. 11 Decision support system screen-shots (part 2)

Over the two-dimensional view the architect draws panels. The interactive design visually informs of well (green) and ill (red) defined panels.



Fully automatic solutions are also possible (greedy and global constraints algorithms).

**Fig. 12** Decision support system screen-shots (part 3)

underlying constraint solver, in our case Choco, propagates and applies search using only those constraints defining a layout plan.

Regarding the performance, the two configuration tasks must be studied separately. As commented before, applying a given compatibility table in the filtering service is made in constant time. Thus, the time involved in the filtering service depends on the questionnaire and on the number of spatial entities (buildings, façades and so on). On the solving service, by contrast, the performance depends on the underlying solving provided by each algorithmic solution. Execution over façades with size  $40 \times 10$  meters,  $50 \times 12$  meters and  $60 \times 15$  meters lasts between one and two seconds. The use of a dedicated heuristic that exploits the problem structure allows to reach such good performance.

## 7 Conclusions

The reduction of energy consumption in buildings is recognized as an international key issue for the coming years. As the construction of new energy efficient buildings is not enough to face the demand, the renovation of existing ones is a real need. In addition, conception and implementation of such renovation must be supported by intelligent systems if efficiency and harmony are desired.

This work is part of a project that investigates the possibility of automated building renovation based on rectangular panels and assisted by a support system. The aim of this article has been to introduce a constraint-based decision support system for the mass customization of insulating envelopes. The automation of the industrial process is motivated

by the number of buildings to renovate, the number of possible insulating envelopes configuration and the complexity of such configuration. All limitations from the industrial scenario, extracted by stakeholders, are integrated into an on-line support system able to support architects decision-making. In summary, for the case of building thermal renovation we have:

1. Described the renovation process and its impact on the configuration.
2. Described the dependency between the spatial entities of the renovation and its impact over the configuration.
3. Described the configuration process from the user perspective, highlighting the relation between the user's answers and the configuration.
4. Clearly stated two configuration tasks needed by the configuration of envelopes and solved by the support system.
5. Discussed the current personalization and recommendation features of the system (panels' bound and orientation).
6. Clearly formalized the input of the system as well as each of its services: one formalized as compatibility tables whereas the other one as first order formulas.
7. Shown that consistency and integrity of solutions are straightforward modeled and implemented thanks to the monotonic properties of constraint satisfaction problems.
8. Shown that the underlying coupling and communication methods are transparent for the user.

Perspectives of this work are as follow. Firstly, the inclusion of information about transportation companies, workforce companies and supporting machinery suppliers into the support system. Doing so the user can select the specific company to be hired in the renovation process so the impact of their service over the configuration can be further automated. Secondly, the support system must be able to configure additional equipment over the insulating envelopes. This equipment refers to the type of new frames (e.g., transparent, opaque, polarized), the option to add window shutters and solar modules. These possibilities would have an impact on the configuration, cost and performance of the envelopes. Thirdly, the inclusion of aesthetic preferences on the configuration process. This is a difficult challenge as the modeling of subjective criteria is complex in many models and in particular in the constraint satisfaction framework but is necessary for the final result of the renovated buildings. Finally, a deeper analysis of the industrial cases may give us hints on how to cope with personalization and include recommendation features, not considered in the present work. For instance, a user can ask for solutions with a specific ratio of vertical and horizontal panels in order to play with the aesthetics of the envelopes.

**Acknowledgments** The authors wish to acknowledge the TBC Générateur d'Innovation company, the Millet and SyBois companies and all partners in the CRIBA project, for their contributions on recollecting buildings renovation information. Special thanks to the referees for their comments and to Philippe Chantry from École des Mines d'Albi for his contribution to the on-line system graphical interface and additional abstractions.

## References

- Akin, O., Dave, B., & Pithavadian, S. (1992). Heuristic generation of layouts (hegel): based on a paradigm for problem structuring. *Environment and Planning B: Planning and Design*, 19(1), 33–59.
- Aldanondo, M., Barco, A. F., Vareilles, E., Falcon, M., Gaborit, P., & Zhang, L. (2014). Towards a bim approach for a high performance renovation of apartment buildings. In Fukuda, S., Bernard, A.,

- Gurumoorthy, B., & Bouras, A. (Eds.) *Product lifecycle management for a global market, IFIP advances in information and communication technology*, (Vol. 442 pp. 21–30). Berlin Heidelberg: Springer. doi:[10.1007/978-3-662-45937-9\\_3](https://doi.org/10.1007/978-3-662-45937-9_3).
- Barco, A.F., Vareilles, E., Aldanondo, M., & Gaborit, P. (2014). A recursive algorithm for building renovation in smart cities. In *Foundations of intelligent systems, lecture notes in computer science*, (Vol. 8502 pp. 144–153). Springer International Publishing. doi:[10.1007/978-3-319-08326-1\\_15](https://doi.org/10.1007/978-3-319-08326-1_15).
- Barco, A. F., Fages, J. G., Vareilles, E., Aldanondo, M., & Gaborit, P. (2015a). Open packing for facade-layout synthesis under a general purpose solver, In Pesant, G. (Ed.) *Principles and practice of constraint programming, lecture notes in computer science*, (Vol. 9255 pp. 508–523): Springer International Publishing. doi:[10.1007/978-3-319-23219-5\\_36](https://doi.org/10.1007/978-3-319-23219-5_36).
- Barco, A.F., Vareilles, E., Gaborit, P., & Aldanondo, M. (2015b). Industrialized building renovation: Manufacturing through a constraint-based on-line support system. In *2015 IEEE international conference on industrial engineering and engineering management (IEEM)* (pp 947–951) doi:[10.1109/IEEM.2015.7385789](https://doi.org/10.1109/IEEM.2015.7385789).
- Barták, R., Salido, M., & Rossi, F. (2010). Constraint satisfaction techniques in planning and scheduling. *Journal of Intelligent Manufacturing*, 21(1), 5–15. doi:[10.1007/s10845-008-0203-4](https://doi.org/10.1007/s10845-008-0203-4).
- Baykan, C. A., & Fox, M. S. (1992). *Artificial intelligence in engineering design (volume i)*. San Diego: Academic Press Professional, Inc.. chap WRIGHT: A Constraint Based Spatial Layout System, pp. 395–432.
- Beldiceanu, N., Carlsson, M., Poder, E., Sadek, R., & Truchet, C. (2007). A generic geometrical constraint kernel in space and time for handling polymorphic k-dimensional objects, In Bessière, C. (Ed.) *Principles and practice of constraint programming CP 2007, lecture notes in computer science*, (Vol. 4741 pp. 180–194). Berlin Heidelberg: Springer. doi:[10.1007/978-3-540-74970-7\\_15](https://doi.org/10.1007/978-3-540-74970-7_15).
- Beldiceanu, N., Carlsson, M., Demasse, S., & Poder, E. (2011). New filtering for the cumulative constraint in the context of non-overlapping rectangles. *Annals of Operations Research*, 184(1), 27–50. doi:[10.1007/s10479-010-0731-0](https://doi.org/10.1007/s10479-010-0731-0).
- Blecker, T., & Abdelkafi, N. (2006). Complexity and variety in mass customization systems: analysis and recommendations. *Management Decision*, 44(7), 908–929. doi:[10.1108/00251740610680596](https://doi.org/10.1108/00251740610680596).
- Center TEC (2011). Energy conservation handbook. The Energy Conservation Center, Japan.
- Council UGB (2013). New Construction Reference Guide. U.S. Green Building Council.
- Falkner, A., & Schreiner, H. (2014). Chapter 16 - siemens: configuration and reconfiguration in industry, In Felfernig, A., Hotz, L., Bagley, C., & Tiihonen, J. (Eds.) *Knowledge-based configuration* (pp. 199–210). Boston: Morgan Kaufmann. doi:[10.1016/B978-0-12-415817-7.00016-5](https://doi.org/10.1016/B978-0-12-415817-7.00016-5).
- Falkner, A., Felfernig, A., & Haag, A. (2011). Recommendation technologies for configurable products. *AI Magazine*, 32, 99–108.
- Fan, L., Musialski, P., Liu, L., & Wonka, P. (2014). Structure completion for facade layouts. *ACM Transactions on Graphics*, 33(6), 210:1–210:11. doi:[10.1145/2661229.2661265](https://doi.org/10.1145/2661229.2661265).
- Felfernig, A., Hotz, L., Bagley, C., & Tiihonen, J. (2014). Knowledge-based configuration: from research to business cases. 1st edn. San Francisco: Morgan Kaufmann Publishers Inc.
- Flemming, U. (1990). Knowledge representation and acquisition in the LOOS system. *Building and Environment*, 25(3), 209–219. doi:[10.1016/0360-1323\(90\)90047-U](https://doi.org/10.1016/0360-1323(90)90047-U).
- Flemming, U., & Woodbury, R. (1995). Software environment to support early phases in building design (seed): overview. *Journal of Architectural Engineering*, 1(4), 147–152. doi:[10.1061/\(ASCE\)1076-0431\(1995\)1:4\(147\)](https://doi.org/10.1061/(ASCE)1076-0431(1995)1:4(147)).
- Fogliatto, F. S., da Silveira, G. J., & Borenstein, D. (2012). The mass customization decade: an updated review of the literature. *International Journal of Production Economics*, 138(1), 14–25.
- Gelle, E., & Weigel, R. (1996). Interactive configuration using constraint satisfaction techniques. In *Second international conference on practical application of constraint technology, PACT-96* (pp. 37–44). Menlo Park: AAAI Press.
- Gent, I., & Walsh, T. (1999). Csplib: a benchmark library for constraints, In Jaffar, J. (Ed.) *Principles and practice of constraint programming - CP'99. Lecture notes in computer science*, (Vol. 1713 pp. 480–481). Berlin Heidelberg: Springer.
- Höfling, B. (2014). Chapter 18 - encoway: from erp-based to sales-oriented configuration, In Felfernig, A., Hotz, L., Bagley, C., & Tiihonen, J. (Eds.) *Knowledge-based configuration* (pp. 219–227). Boston: Morgan Kaufmann. doi:[10.1016/B978-0-12-415817-7.00018-9](https://doi.org/10.1016/B978-0-12-415817-7.00018-9).
- Jelle, B. P. (2011). Traditional, state-of-the-art and future thermal building insulation materials and solutions - properties, requirements and possibilities. *Energy and Buildings*, 43(10), 2549–2563. doi:[10.1016/j.enbuild.2011.05.015](https://doi.org/10.1016/j.enbuild.2011.05.015).
- Junker, U. (2006). *Configuration. Chapter 24 of handbook of constraint programming (foundations of artificial intelligence)*. New York: Elsevier Science Inc.



- MacCarthy, B., Brabazon, P. G., & Bramham, J. (2003). Fundamental modes of operation for mass customization. *International Journal of Production Economics*, 85(3), 289–304.
- Medjdoub, B., & Yannou, B. (2000). Separating topology and geometry in space planning. *Computer-Aided Design*, 32(1), 39–61. doi:10.1016/S0010-4485(99)00084-6.
- Mittal, S., & Falkenhainer, B. (1990). Dynamic constraint satisfaction problems. In *AAAI* (pp 25–32).
- Nica, I., Wotawa, F., Ochenbauer, R., Schober, C., Hofbauer, H., & Boltek, S. (2014). Chapter 19 - kapsch: reconfiguration of mobile phone networks, In Felfernig, A., Hotz, L., Bagley, C., & Tiihonen, J. (Eds.) *Knowledge-based configuration* (pp. 229–240). Boston: Morgan Kaufmann. doi:10.1016/B978-0-12-415817-7.00019-0.
- OpenRules, Inc (2013). Constraint programming solvers catalog. Available from <http://openjvm.jvmhost.net/CPSolvers/>.
- Peng, Y., A Lu, D., & A Chen, Y. (2014). A constraint programming method for advanced planning and scheduling system with multilevel structured products. *Discrete Dynamics in Nature and Society*, 2014(Article ID 917685). doi:10.1155/2014/917685. 7 pages.
- Pérez-Lombard, L., Ortiz, J., & Pout, C. (2008). A review on buildings energy consumption information. *Energy and Buildings*, 40(3), 394–398. doi:10.1016/j.enbuild.2007.03.007.
- Prud'homme, C., & Fages, J. (2013). An introduction to choco 3.0 an open source java constraint programming library. In *CP solvers: modeling, applications, integration, and standardization*. International workshop, Uppsala Sweden.
- Sabin, D., & Freuder, E. (1996). Configuration as composite constraint satisfaction. In *Proc. artificial intelligence and manufacturing. Research planning workshop* (pp. 153–161). AAAI Press.
- Schimpf, J., & Shen, K. (2012). Eclipse-from lp to clp. *Theory Practice Log Program*, 12(1-2), 127–156. doi:10.1017/S1471068411000469.
- Schreiber, Y. (2010). Principles and practice of constraint programming. In *CP 2010: 16th international conference, CP 2010, St. Andrews, Scotland, September 6-10, 2010. Proceedings, chap value-ordering heuristics: search performance vs. solution diversity* (pp. 429–444). Berlin: Springer. doi:10.1007/978-3-642-15396-9\_35.
- Schulte, C., Tack, G., & Lagerkvist, M.Z. (2010). Modeling and programming with gencode.
- Shikder, S., Price, A., & Mourshed, M. (2010). Interactive constraint-based space layout planning. In *W070-special track 18th CIB world building congress* (p. 112). Salford.
- Smith, B. M. (2006). *Modelling. Chapter 11 of handbook of constraint programming (foundations of artificial intelligence)*. New York: Elsevier Science Inc.
- Soininen, T., Tiihonen, J., Männistö, T., & Sulonen, R. (1998). Towards a general ontology of configuration. *Artificial Intelligence in Engineering Descriptive Analysis Manufacturing*, 12(4), 357–372. doi:10.1017/S0890060498124083.
- Soininen, T., Niemelä, I., Tiihonen, J., & Sulonen, R. (2000). Unified configuration knowledge representation using weight constraint rules. *ECAI-2000 Workshop on Configuration*.
- Teboul, O., Simon, L., Koutsourakis, P., & Paragios, N. (2010). Segmentation of building facades using procedural shape priors. In *2010 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 3105–3112). doi:10.1109/CVPR.2010.5540068.
- Tiihonen, J., & Felfernig, A. (2010). Towards recommending configurable offerings. *International Journal of Mass Customisation*, 3(4), 389–406. doi:10.1504/IJMASSC.2010.037652.
- Topaloglu, S., Salum, L., & Supciller, A. (2012). Rule-based modeling and constraint programming based solution of the assembly line balancing problem. *Expert Systems with Applications*, 39(3), 3484–3493. doi:10.1016/j.eswa.2011.09.038.
- Triska, M. (2012). The finite domain constraint solver of swi-prolog, In Schrijvers, T., & Thiemann, P. (Eds.) *Functional and logic programming, lecture notes in computer science*, (Vol. 7294 pp. 307–316). Berlin: Springer. doi:10.1007/978-3-642-29822-6\_24.
- Vareilles, E., Gaborit, P., Aldanondo, M., Carbonnel, S., & Steffan, L. (2012). Cofiade constraints filtering for aiding design. In *Actes des neuviemes Journées Francophones de Programmation par Contraintes*. Toulouse France.
- Vareilles, E., Barco Santa, A., Falcon, M., Aldanondo, M., & Gaborit, P. (2013). Configuration of high performance apartment buildings renovation: a constraint based approach. In *2013 IEEE international conference on industrial engineering and engineering management (IEEM)* (pp. 684–688). doi:10.1109/IEEM.2013.6962498.
- Wielemaker, J., Schrijvers, T., Triska, M., & Lager, T. (2010). Swi-prolog. Computing Research Repository abs/1011.5332.
- Wu, F., Yan, D. M., Dong, W., Zhang, X., & Wonka, P. (2014). Inverse procedural modeling of facade layouts. *ACM Transactions on Graphics*, 33(4), 121:1–121:10. doi:10.1145/2601097.2601162.

- Xie, H., Henderson, P., & Kernahan, M. (2005). Modelling and solving engineering product configuration problems by constraint satisfaction. *International Journal of Production Research*, 43(20), 4455–4469. doi:[10.1080/00207540500142381](https://doi.org/10.1080/00207540500142381).
- Yang, D., Dong, M., & Miao, R. (2008). Development of a product configuration system with an ontology-based approach. *Computer-Aided Design*, 40(8), 863–878. doi:[10.1016/j.cad.2008.05.004](https://doi.org/10.1016/j.cad.2008.05.004).
- Yang, D., Dong, M., & Chang, X. (2012). A dynamic constraint satisfaction approach for configuring structural products under mass customization. *Engineering Applications of Artificial Intelligence*, 25(8), 1723–1737.