A Sequential Guillotine Cut Heuristic to Design Insulating Envelopes

Andrés F. Barco
Lidis Research Group
Universidad de San Buenaventura
Santiago de Cali, Colombia, South America
anfelbar@usbcali.edu.co

Michel Aldanondo, Élise Vareilles
Céntre de Génie Indutriel
Université de Toulouse, Mines Albi
Albi, France
michel.aldanondo@mines-albi.fr, elise.vareilles@mines-albi.fr

Abstract

Two-dimensional Cutting & Packing problems consist in allocating a set of two-dimensional items into a set (possible singleton) of two-dimensional larger objects in such a way that items are completely contained in the objects without overlapping. A particular instance of two-dimensional Cutting & Packing problem arises in the context of thermal building renovation. This special case deals with the design of insulating envelopes by packing a set of rectangular and parameterizable panels (items) over rectangular facades (objects). Taking into account the particularities inherited from the industrial problem and the need of aesthetic results, we propose in this paper a constraint-based heuristic, named CaSyE, based on guillotine cuts technique. First, we introduce the industrial problem and its particularities. Then, the constraint-definition of the problem is presented as well as the scheme of the CaSyE solution. At least, the results of the heuristic over realistic facades are shown before the conclusion.

Keywords
Cutting and packing, guillotine cuts, building renovation, decision support system, constraint-based heuristic

1. Introduction

Two-dimensional Cutting & Packing problems consist in allocating a set of two-dimensional items into a set (possible singleton) of two-dimensional larger objects in such a way that items are completely contained in the objects without overlapping. A particular instance of two-dimensional Cutting & Packing problem arises in the context of thermal building renovation. This special case deals with the design of an insulating envelope by packing a set of rectangular and parameterizable (in terms of dimension and position) panels (items) over rectangular facade (objects). The envelope is used to reduce the thermal transfer between the interior and the exterior of the building in the aim of reducing the building's energy consumption.

In two previous reports, the authors have developed heuristic solutions to solve this problem: The first using an on-the-fly greedy approach (Barco et al., 2014) and the second using a filtering-based approach (Barco et al., 2015). However, the generated envelope solutions from these heuristics do not count with aesthetic aspects needed for the retrofit (such as symmetry). Indeed, tacit knowledge as aesthetics is a major challenge for both the modeling and implementation of computer-based solutions. It is known that aesthetics flair has no universally agreed standard. Nonetheless, properties like symmetry are well accepted as aesthetic concept in different domains such as human beauty (Jacobsen and Hofel, 2003), web design (Tuch et al., 2010), computer interfaces (Bauerly and Liu, 2008) and art (McManus, 2005). In consequence, we have adopted symmetry as a reference point for envelopes aesthetics and considered it as the alignment of panel junction.

One interesting direction to tackle the design problem is Cutting (Wascher, 2007). This is a technique from operations research widely used to solve different industrial problems (Bennell et al., 2013; Christofides and

Hadjiconstantinou., 1995) and it has been applied on problems where symmetry is relevant for the final output. It has been used, for instance, for arranging items in a newspaper (Strecker and Hennig, 2009), automatic mosaic generation (Battiato, et al., 2013) and aesthetics photo post-processing (Greco and Cascia, 2013). We have developed a constraint-based heuristic, called CaSyE for Cutting Algorithm for Symmetrical Envelopes, based on the technique of guillotine cuts (Christofides and Hadjiconstantinou., 1995) and aimed at designing envelopes with symmetric (alignment of junctions) appearance.
The remaining of the paper is divided as follows. In Section 2, the facade-layout elements are introduced. In Section 3, the constraint-based definition of the problem is presented. In Section 4, the general scheme of the solution based in guillotine cuts is discussed. In Section 5, the heuristic CaSyE is introduced. Afterwards, in Section 6, solution illustrations are shown. Finally, some conclusions are discussed in Section 7.

2. Building Renovation

The addressed problem appears in a large French multi-partner project called CRIBA (for its acronym in French of Construction and Renovation in Industrialized Wood Steel) that aims to industrialize building renovation in order to reduce energy consumption (Vareilles et al., 2013; Aldanondo et al., 2014). In this section, the problem from the industrial point of view is presented.
A facade is represented by a two-dimensional coordinate plane with origin of coordinates (0,0) at the bottom-left corner of the facade, and contains rectangular zones defining:

- Perimeter of facade with it size (height and width in meters).
- Frames (existing windows and doors over the facade) play an important role as they are meant to be completely overlapped by one and only one panel. Frames are defined with:
  - Origin point (x,y) with respect to origin of facade.
  - Width and height (in meters).
- 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 facade (slabs and shear walls) have load bearing capabilities that allow us to attach panels. Supporting areas have well-defined:
  - Origin point (x,y) with respect to origin of facade.
  - Width and height (in meters).
  - Load bearing capability (in kg/m^2).

Panels are rectangular, of varying sizes and may include new frames (replacing the existing ones). These panels are designed one at a time in the process of layout synthesis and manufactured in the factory prior to shipment and installation on the building site. These panels have a well-defined:

- Size (height and width in meters). Height and width are constrained by a given lower and upper bound related to manufacturing, environmental and transportation limitations.
- New frames (replacing the existing ones). Given internal structure of rectangular panels, new frames must respect a parameterizable minimum distance (d) with panel's borders.

The problem subject of our study has five particularities explained in what follows. It is worth noticing that facades, panels, windows, doors and supporting are all rectangular and their edges are parallel to the facade edges.

1. Number of panels and their size incognito. Unlike most Cutting & Packing problems, the number of panels used to create an insulating envelope is not known before the design process starts. In addition, the size of panels is bounded to a given interval product of manufacturing and transportation conditions.
2. Mandatory overlapping (of frames). The first side constraint of the problem is that new frames must be completely included, and therefore overlapped, by panels. Any of these frames must be covered with only one panel, meaning that the partial overlapping of frames is not allowed.
3. Panels' installation (over supporting areas). The second side constraint involves the installation of panels over the facades. In fact, due to the added weight of the panels and given the vertical orientation of facades, panels can only be attached in supporting areas, that will uniformly distribute their weight thus preventing them to fall and the facades to collapse.
4. No overlapping, no holes. Likewise most Cutting & Packing problems, panels overlapping are forbidden. In addition, given the renovation context, the existence of holes in a solution is infeasible (holes are impractical for the thermal insulation). In consequence, panels must be adjacent to each other.
5. Solutions' Performance. The ranking of solutions is made with the number of panels: Junctions between panels introduce a thermal leak that is minimized when minimizing the number of panels. For two envelopes with the same number of panels, the one with minimum length of junctions is preferred.

3. CSP Modeling

In this section the Cutting & Packing problem is formalized as a CSP. Now, in order to give limit to variables, it is needed a set of parameters containing all geometrical and structural information linked to the facade and needed to establish the relation among panels, and panels and facade. Each renovation being unique, the problem has to be adjusted considering some parameters:

- Height fac_h and width fac_w in meters.
- Set F of frames and for each frame $fr^j \in F$:
  - Origin point (fr^j_x, fr^j_y) with respect to origin of facade (fac_{xo}, fac_{yo}).
  - Width fr^j_w and height fr^j_h in meters.
- Set Sa of supporting areas and for each supporting area $sa^k \in Sa$:
  - Origin point (sa^k_x, sa^k_y) with respect to origin of facade (fac_{xo}, fac_{yo}).
  - Width sa^k_w and height sa^k_h in meters.
  - Load bearing capability sa^k_l in kg/m^2.

Decision variables are linked to the position and size of each panel in an insulating envelope. Let us assume that N represents the number of panels in a given insulating envelope. Then, each panel p_i with $i \in [1, N]$ is described by its origin and size attributes

- Its width $p^i_w \in [p_{wl}, p_{wu}]$, lower and upper bounds for panels' width.
- Its height $p^i_h \in [p_{hl}, p_{hu}]$, lower and upper bounds for panels' height.
- Its coordinates (p^i_{x0}, p^{iy0}), bottom-left corner of the panel p^i.

The knowledge extracted from the problem domain by stakeholders (e.g., architects) has been mapped into the constraints in Figure 1. They state the properties a well-designed envelope must possess.

C$_1$ Size constraint

$$\forall p^i, 1 \leq i \leq N: \quad p_{wl} \leq p^i_w \leq p_{wu} \wedge p_{hl} \leq p^i_h \leq p_{hu}$$

C$_2$ Panels and frames constraints

$$\forall_{fr^j} \in F, \exists_{p^i}, 1 \leq i \leq N \mid p^i_{x0} + d \leq fr_x \wedge fr_x + fr_w \leq p^i_{x0} + p^i_w + d$$
$$\wedge\ p^i_{y0} + d \leq fr_y \wedge fr_y + fr_h \leq p^i_{y0} + p^i_h + d$$

C$_3$ Installation constraint

$$\forall p^i, 1 \leq i \leq N:$$
$$\exists_{sa^k} \mid (sa^k_x \leq p^i_{x0} \wedge p^i_{x0} \leq sa^k_x + sa^k_w \wedge\ sa^k_y \leq p^i_{y0} \wedge p^i_{y0} \leq sa^k_y + sa^k_h)$$
$$\exists_{sa^l} \mid (sa^l_x \leq p^i_{x0} \wedge p^i_{x0} \leq sa^l_x + sa^l_w \wedge\ sa^l_y \leq p^i_{y1} \wedge p^i_{y1} \leq sa^l_y + sa^l_h)$$
$$\exists_{sa^m} \mid (sa^m_x \leq p^i_{x1} \wedge p^i_{x1} \leq sa^m_x + sa^m_w \wedge\ sa^m_y \leq p^i_{y0} \wedge p^i_{y0} \leq sa^m_y + sa^m_h)$$
$$\exists_{sa^n} \mid (sa^n_x \leq p^i_{x1} \wedge p^i_{x1} \leq sa^n_x + sa^n_w \wedge\ sa^n_y \leq p^i_{y1} \wedge p^i_{y1} \leq sa^n_y + sa^n_h)$$

C$_4$ Non-overlapping constraint

$$\forall_{p^u, q^v} \mid p^u_{x0} \geq p^v_{x0} + p^v_w \vee p^v_{x0} \geq p^u_{x0} + p^u_w$$
$$\vee\ p^u_{y0} \geq p^v_{y0} + p^v_h \vee p^v_{h0} \geq p^u_{h0} + p^u_h$$

C$_5$ Area (no holes) constraint

$$\sum_{i=1}^{N} \left(p^i_w \times p^i_h\right) = fac_w \times fac_h$$

Figure 1. Constraints in disjunctive normal form.

4. Scheme of CaSyE Heuristic

The main idea behind the heuristic CaSyE is to generate envelopes "aesthetically pleasant" by considering their junctions alignment (symmetry) while respecting the industrial conditions commented above. To do this, the set of panels in an envelope is configured according to a given ration between p w and p_h, i.e., a given panel orientation (horizontal for $p_w > p_h$ and vertical otherwise). If due to the geometry of the facade, a portion of the envelope cannot be designed using the chosen orientation, the heuristic creates a partition of the facade, called here sub-facade, and tries to design that partition by changing the panel's orientation. This means that insulating envelopes may contain only vertical panels, only horizontal panels or a combination of both. The CaSyE heuristic is divided in three phases in which guillotine cuts are executed. As an invariant, a guillotine cut in a given facade or sub-facade is done only if no frames conflict exists.

4.1 Phase 1: Free Zones and Sub-facades

The goal of the phase 1 is to know whether the envelope can be designed using only vertical panels, only horizontal panels, both or to find the sub-facades and characterize them within the facade. Intuitively, when using guillotine cuts to design an envelope with vertical panels, for instance, the facade horizontal axis is traversed looking for points with absence of frames. When no frames are present in a given horizontal point, a vertical guillotine cut may trace from the bottom to the top of the facade. A set of consecutive cuts makes an interval (see Figure 2.a). These lines or intervals, termed as Zones Free of Conflicts (ZoFCo), are used latter to place panels' borders. Then, the heuristic of the phase 1 tries to deduce if the facade insulating envelope can be designed using vertical panels by checking the position and dimension of the ZoFCos. If portions of the facade (sub-facades) cannot be covered with panels in the chosen orientation, the heuristic makes a division into sub-facades and marks each of them with a type for further processing. In particular, sub-facade types are; Vertical, the envelope can be designed using vertical panels; Horizontal, the envelope can be designed using horizontal panels; NotVertical, the envelope cannot be designed using vertical panels; NotHorizontal, the envelope cannot be designed using horizontal panels.
A sub-facade may be marked sequentially with the following combinations: NotVertical-Horizontal, NotHorizontal-Vertical, NotVertical-NotHorizontal and NotHorizontal-NotVertical. In the last case the insulating envelope cannot be designed using the proposed heuristic. For instance, let us study the facade in Figure 2.b. Here, let us assume panels' upper bounds of 3 meters for one dimension and 10 meters for the other dimension (i.e., 10 × 3 for a horizontal panel or 3 × 10 for a vertical panel). Envelopes for the sub-facades sub 1 and sub 2 may be designed using vertically and horizontally oriented panels. But, the sub-facade sub 3 at the right cannot be designed: One dimension (width or height) may be successfully covered with 10 meters but the other dimension, that can only takes as maximum size 3 meters, cannot be covered. A non-designed sub-facade implies a hole in the insulating envelope, which is forbidden.

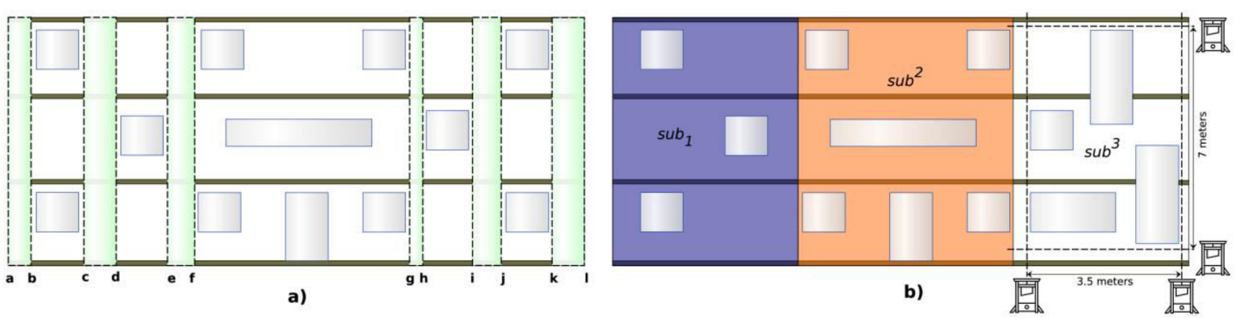


Figure 2. a) Intervals and sub-facades definition and b) Sub-facade with no solution with guillotine cuts.

To clearly understand the scheme previously discussed, let us study the steps for the phase 1 using as example vertical orientation.

Step 1: Let $x_1 = 0$ (i.e., left border of facade) iterate to the right until the end of facade and do:

i. Let x_2 be equal to the origin point of the first frame found.

ii. Define a ZoFCo from x_1 to x_2.
iii. Set x_1 equals to the right edge of the frame. If another frame is blocking the guillotine cut at x_1, then update x_1 to the end point of that blocking frame and repeat until no blocking frames are found. For example, in Figure 2.a, after adding the ZoFCo defined by [e,f], it should start the new iteration with x_1=g as in this point a vertical non-conflictive cut is possible.

Step 2: For all ZoFCo found, make an ordered check: If the distance between the end of ZoFCo i and start of ZoFCo i+1 is bigger than the maximum panel width p_{wu}, then at least two sub-facades and at most three sub-facades have been found (more sub-facades may be found when checking the remaining ZoFCos). This means that the space to be covered has a larger width than the maximum panel width. In our example, the first sub-facade sub 1, labeled as Vertical, goes from a until the point f (see Figure 3.b sub-facade at the left). The second sub-facade sub 2, labeled as NotVertical, goes from f until the point g (see Figure 3.b sub-facade in the middle). The third sub-facade sub3, labeled as Vertical, goes from g until the point l (see Figure 3.b sub-facade at the right).
Step 3: For each sub-facade try failure detection. If its width is less than p_{wl}, then; a) merge it with an adjacent sub-facade already marked as NotVertical (if any) or; b) mark the sub-facade as NotVertical.
Step 4: For any sub-facade marked as NotVertical, try to design the sub-facade envelope using horizontal orientation.

At the end of the process, every sub-facade has been marked.

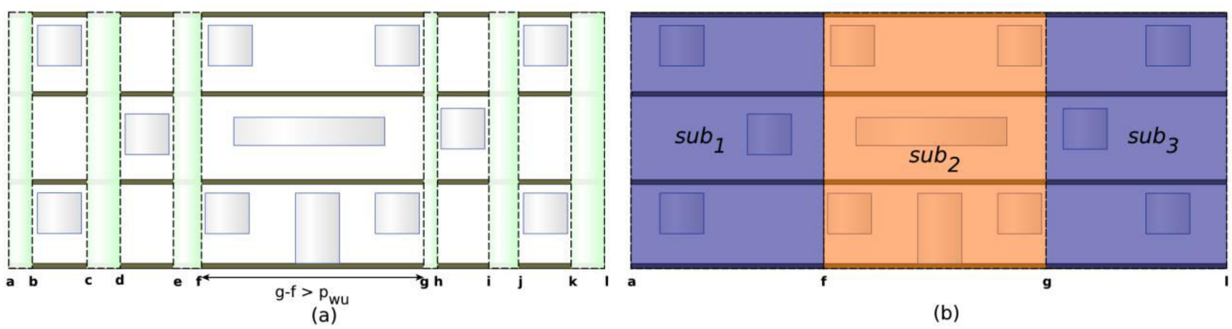


Figure 3. Definition of sub-facades according to ZoFCos.

4.2 Phase 2: Defining Columns and Rows

After the phase 1, the second phase is executed for each sub-facade. In this phase, the columns (respectively rows) for the vertically (respectively horizontally) designed envelope must be defined. It is at these columns (rows) where panels are going to be placed, the edges of the panels matching the edges of the columns. For convenience, let us continue our solution description using vertical panels. The first task is to determine where the left border and the right border of columns will be placed. To do so, the phase 2 uses the ZoFCos intervals that have been found in the previous phase as they are free of frames conflicts. Then, it suffices to select a point within the ZoFCos to define the columns. As an invariant, given that the entire sub-facade must be covered (no holes in the envelope) and considering the column definition from the left to the right of the sub-facade, the end of the column i must be equal to the start of column i + 1.
Taking into account that envelopes should be composed of the minimum number of panels, the definition of the columns, for instance, is made using the upper bounds for panels' width (respectively height). The idea is to place the left and right edge of the column over one or two ZoFCo, in such a way that the width is maximal (see Figure 4.a). If using the width upper bound makes the column enter in a frame conflict, meaning that the point in which the width is maximal does not match a ZoFCo, then the upper bound cannot be used for the current column width (see Figure 4.b). In consequence, the width of the column is reduced, as less as possible, while solving the conflict (see Figure 5.a). Potential inconsistent sizes are handled as well. The heuristic for the phase 2 is applied to every sub-facade independently and behaves as follows. Let x_1 and x_2 define the start and end of the column i. Both variables are used to iterate until the end of the sub-facade. Then do:

Step 1: Let x_1 be equal to the origin (left border) of the sub-facade.

Step 2: Let $x_2=x_1 + p_{wu}$.
Step 3: If the point x_2 does not belong to any ZoFCo, then move x_2 to be the end (right edge) of the previous ZoFCo (the previous at the left). As an illustration, the second column in Figure 4.b is redefined to match the first ZoFCo on the left, as shown in Figure 5.a.

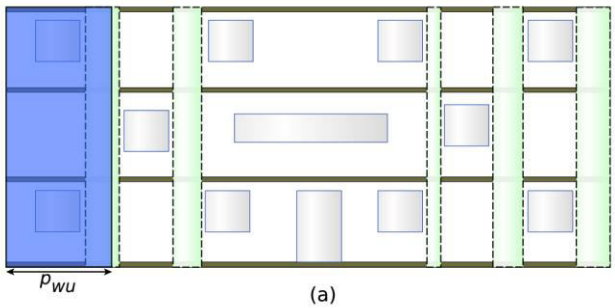


Figure 4. Guillotine cut for columns definition.

Step 4: Try failure detection. If $x_2 - x_1 < p_{wl}$, then reduce width of previous column $i - 1$ by $p_{wl} - (x_2 - x_1)$ and update x_1 and x_2. This process must be done iteratively as a width reduction in any previous column may generate new size conflicts. Lastly, if there is no previous column then fail and exit.

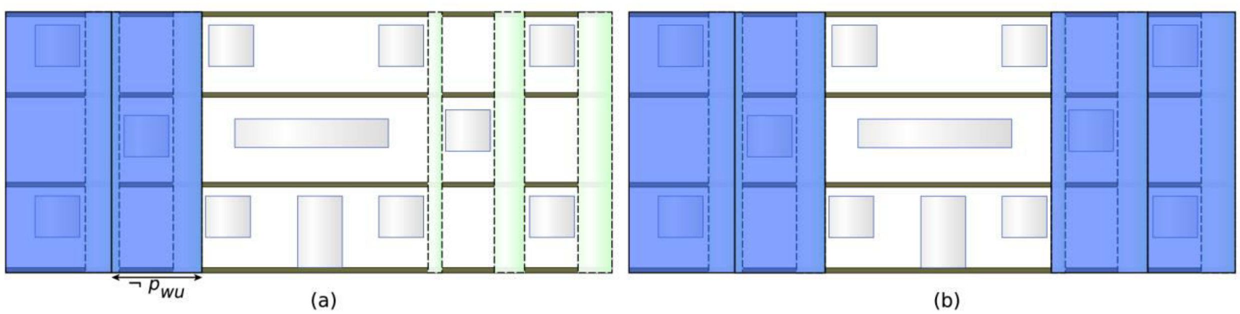


Figure 5. Guillotine cut for columns definition (continuation).

Step 5: Define the column i from x_1 to x_2.
Step 6: Set $x_1 = x_2$ and iterate again from Step 2 until all the sub-facade has been processed.

At the end of the process, every sub-facade has been divided in columns (respectively rows) where panels' borders will be located. The last process then sets the final position and size of panels over these columns.

4.3 Phase 3: Panels Packing

The packing of panel is executed for each of the columns and rows generated in the phase 2. As commented before, this last phase has as objective to set the final position and size of panels. Also, this phase handles potential conflicts with frames by executing guillotine cuts in non-conflictive zones. Likewise the previous phases, the packing starts by an extreme of the sub-facade until its end. In the case of an envelope designed using vertical panels, the packing process starts in the bottom of each column by extending the panel in its maximum allowed height (see Figure 6.a). If a horizontal cut is not possible, the panel height must be reduced to the first place in which no conflict exists (see Figure 6.b).

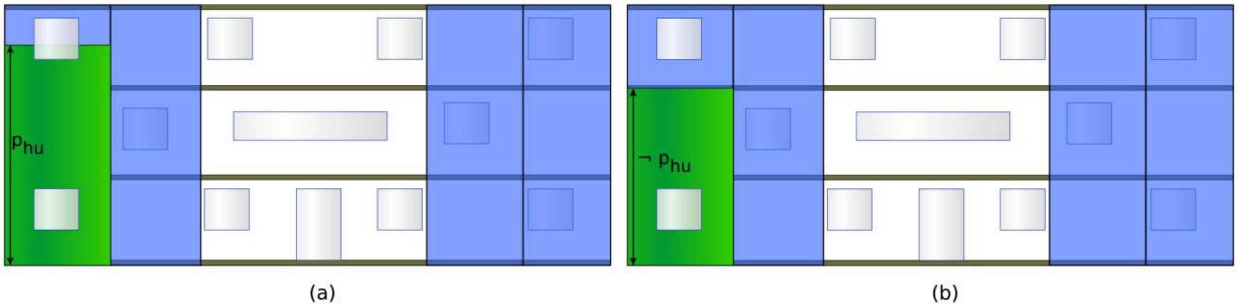


Figure 6. Guillotine cuts for packing panels.

The packed panels in a given column will have the width of the column and the height defined by the horizontal guillotine cut. Again, a similar process is carried on rows when designing envelopes with horizontally oriented panels. The heuristic solution of the third phase, applied to each column independently, behaves as follows.

Let y_1 and y_2 represent, respectively, the bottom edge and top edge of the panel i. Both variables are used to iterate until the end of the column. Then do:

Step 1: Let y_1 equals the bottom of the column.
Step 2: Let $y_2 = y_1 + p_{hu}$.
Step 3: If horizontal guillotine cut in y_2 has conflicts with frames or supporting areas, then move (reduce) y_2 in order to solve the conflicts (as illustrated in Figure 6.b).
Step 4: Check if both top corners are included in supporting areas. If this is not the case, reduce the panel's height until both top-left and top-right match supporting areas.
Step 5: Using the current position and size, compute the panel's weight and check if it is successfully supported in the supporting areas at the bottom corners. If the weight cannot be supported then reduce the panel size in 10% and iterate again from Step 3.
Step 6: Try failure detection. If the $y_2 - y_1 < p_{hl}$, then reduce height of previous panel $i - 1$ by $p_{hl} - (y_2 - y_1)$ and update y_1 and y_2. If there is no previous panel then fail and exit.
Step 7: Define a panel from y_1 to y_2 in the current column.
Step 8: Set $y_1 = y_2$ and iterate from Step 2 until the column has been processed.

It may be the case that the defined columns defined in the previous phase do not traverse supporting areas, in which case the envelope cannot be designed with the CaSyE heuristic using the current panel size bounds. Otherwise, at the end of this phase, each column (respectively row) has been covered by panels. The resulting insulating envelope for the facade is the union of all panels of every sub-facade (if any). Now, let us discuss how these phases are merged together to assist architects design.

5. Heuristic Implementation

The phases are executed sequentially to generate a given envelope. This sequential process is executed twice in order to generate two different solutions (if the facade geometry allows it): One setting the ratio p w > p h (horizontally oriented panels) and one setting the ratio p w < p h (vertically oriented panels). Figure 7 presents the pseudo-code of the CaSyE heuristic based on guillotine cuts.
The heuristic contains two blocks: A block for designing vertical panels (lines 6-17) and another for horizontal ones (lines 21-30). Initially, the envelope for the inputted facade (line 2) is meant to be designed using vertical panels only (lines 6-17). Thus, the phase 1 finds the set of ZoFCo to design the envelope and it will find and mark the partitions (sub-facade) that cannot be designed using vertical panels (line 6) if any. For all sub-facades found that are marked as Vertical, execute the phase 2 and phase 3 (lines 9-13). Note here that if the whole envelope for the inputted facade can be designed using vertical panels, with no partitions, then there is only one sub-facade to be processed; only one sub-facade marked as Vertical. In the case that there exists sub-facade marked as NotVertical, these ones are saved in the unknown list (lines 16-17) for a further processing using horizontal panels only. Once the first block is executed, the heuristic tries to design the envelopes for the sub-facade that are label as NotVertical

Algorithm 1: CaSyE - Cutting Algorithm for Symmetrical Envelopes

```
1  def Casye(façade):
2      unknown ← [ façade ];
3      solution ← [];
4      subfaçades ← [];
5      do
           /* Get subfaçades to be Vertical and NotVertical                    */
6          subfaçades ← phase1(unknown, "vertical");
7          unknown ← [];
8          for (∀ sf ∈ subfaçades) do
9              if (isMarked(sf, "vertical")) then
10                 columns ← phase2(sf, "vertical");
11                 panels ← phase3(columns, "vertical");
12                 if (panels ≠ ∅) then
13                     solution.add(panels);
14             else if (isMarked(sf, "notVertical") and isMarked(sf, "notHorizontal")) then
15                 return []; /* No solution found                                 */
16             else
                   /* Add the non-vertical subfaçades for further processing       */
17                 unknown.add(sf);
           /* Get subfaçades to be Horizontal and NotHorizontal.                */
18         subfaçades ← [];
19         subfaçades ← phase1(unknown, "horizontal");
20         unknown ← [];
21         for (∀ sf ∈ subfaçades) do
22             if (isMarked(sf, "horizontal")) then
23                 rows ← phase2(sf, "horizontal");
24                 panels ← phase3(rows, "horizontal");
25                 if (panels ≠ ∅) then
26                     solution.add(panels);
27             else if (isMarked(sf, "notVertical") and isMarked(sf, "notHorizontal")) then
28                 return []; /* No solution found                                 */
29             else
                   /* Add the non-horizontal subfaçades for further processing     */
30                 unknown.add(sf);
31     while (unknown ≠ ∅);
32     return solution;
```

Figure 7. CaSyE - Cutting Heuristic for Symmetrical Envelopes.

(lines 21-30). The behavior of the second block is then similar as the first block only that the sub-facade in the list unknown are all marked as NotVertical. A solution with all designed panels is returned at the end of the loop when no more sub-facades are left (line 31).

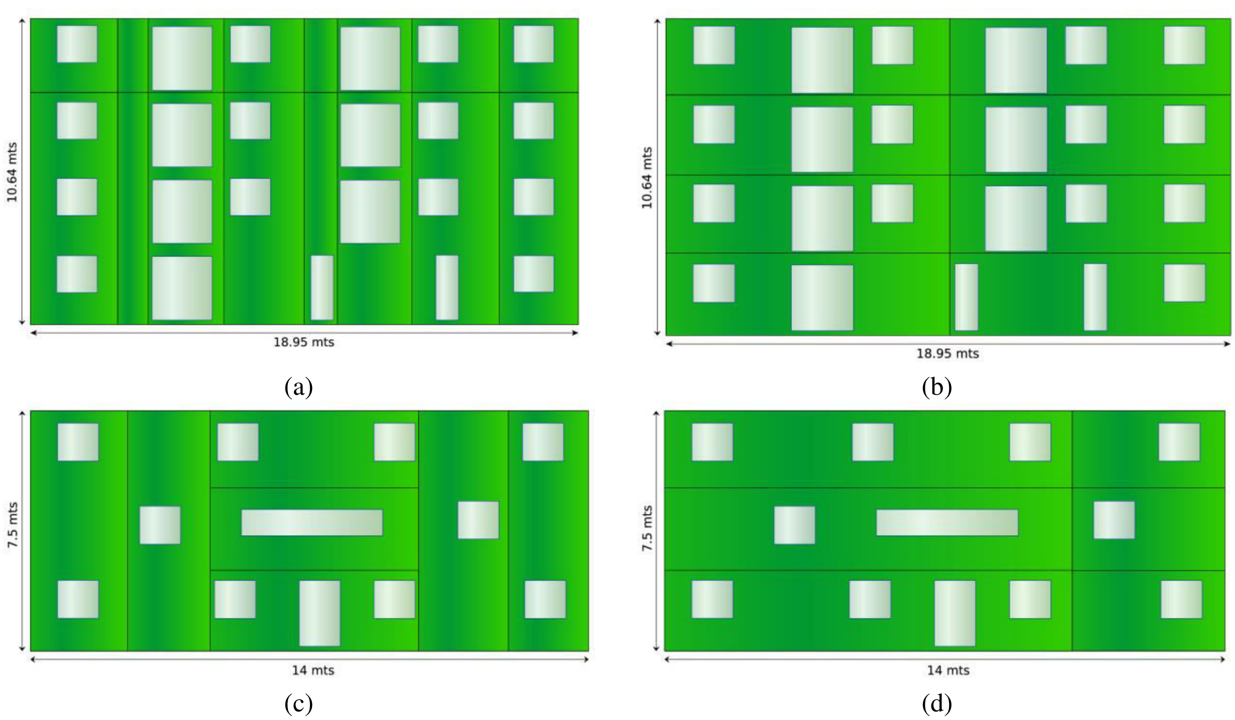
Intuitively, when designing envelopes sub-facades may be created dynamically. Lines 17 and 29 make sure that these new sub-facades are processed later using the corresponding orientation. It does so in the same iteration for sub-facade marked as NotVertical, as they are store in the list unknown and processed in latter in second block (lines 21-30). For the facade marked as NotHorizontal the processing is done in the next iteration by, again, keep them in the unknown list thus preventing getting out of the loop.

However, it may be the case that a given sub-facade cannot be designed using any orientation. If this is the case, i.e., if a given sub-facade has been marked both NotVertical and NotHorizontal, the CaSyE heuristic fails at designing the envelope (lines 14-15 and lines 26-27). This is due to the third condition for envelopes design: No holes are allowed in a solution. Ergo, if a given portion of the inputted facade cannot be designed, then the facade has no solution. Nevertheless, in an attempt to overcome this situation, the heuristic is executed twice: One starting vertically designed envelopes and another starting with horizontally designed envelopes. Indeed, the orientation in which starts an envelope design has an influence on the kind of sub-facade found and consequently in the final output. In other words, swapping the position of the first block (lines 8-17) with the position of the second block (lines 21-30) has an impact on the resulting solution. Thus, a given facade has, potentially, two different solutions.

6. Evaluation

In this section are present envelopes generated by the heuristic. Keep in mind that, although an evaluation of symmetry may be built with respect to the junctions' alignment and panels' sizes, these are not a sufficient condition to consider an envelope aesthetically pleasant. The aesthetics evaluation must be still done visually. Envelopes facades presented in Figure 7 are generated by the CaSyE heuristic.

- Figure 7.a shows an envelope solution made out with p_{wu} = 3 meters as width upper bound and p_{hu} = 10 meters as height upper bound, i.e., vertically designed envelope. The envelope is composed of 16 panels and its length of junctions is 123.02 meters.
- Figure 7.b shows an envelope solution made out with p_{wu} = 10 meters as width upper bound and p_{hu} = 3 meters as height upper bound, i.e., horizontally designed envelope. The envelope is composed of 8 panels and its length of junctions is 97.0 meters.
- Figure 7.c illustrates an envelope solution made out with p_{wu} = 3 meters as width upper bound and p_{hu} = 10 meters as height upper bound, i.e., vertically designed envelope. Here, although the frame in the middle of the facade is blocking the definition of vertical panels, the insulating envelope still has its vertical junctions aligned (symmetric appearance). Further, in the middle only 3 panels were designed thus preventing the length of junctions to increase with small panels. The envelope is composed of 7 panels and its length of junctions is 65.5 meters.
- Figure 7.d illustrates an envelope solution made out with p_{wu} = 10 meters as width upper bound and p_{hu} = 3 meters as height upper bound, i.e., horizontally designed envelope. The envelope is composed of 6 panels and its length of junctions is 57.0 meters.



(a) (b) (c) (d)

Figure 8. Vertical and horizontal insulating envelopes by CaSyE heuristic.

6. Conclusions

In this paper, we have proposed a sequential heuristic, named CaSyE, which exploits facade geometrical structure and takes inspiration from the human aesthetic concept of symmetry to design buildings insulating envelopes. The motivation behind the cutting approach is the need for designing aesthetics insulating envelopes for facade in an automated fashion. Given that no standard definition of aesthetics exists, we have considered the junctions alignment (symmetry) as a good criterion. The heuristic is based on the well-known technique of guillotines cuts and is able to generate up-to two envelopes. Then, it tries to artificially behave as a human being would behave in the sense that conflicts with frames are avoided while attempting to provide a pleasant appearance.
The guillotine cuts technique allows to avoid conflicts of the constraint stated in the model. Also, given that cuts are made from one extreme of the facade to the other, panels junctions are aligned thus given a symmetric appearance. The final position and size of panels are defined according to the non-conflictive areas and panels' size upper bounds in the aim of minimizing the number of used panels. Finally, it is important to highlight that our solution is built under the semantics of each constraint in the model thus guiding the reasoning of the problem and the application development.

References

Aldanondo, M., Barco, A., Vareilles, E., Falcon, M., Gaborit, P., and Zhang, L. Towards a bim approach for a high performance renovation of apartment buildings. In Shuichi Fukuda, Alain Bernard, Balan Gurumoorthy, and Abdelaziz Bouras, editors, Product Lifecycle Management for a Global Market, volume 442 of IFIP Advances in Information and Communication Technology, pages 21–30. Springer Berlin Heidelberg, 2014.

Barco, A., Fages, JG., Vareilles, E., Aldanondo, M., and Gaborit, P. Open packing for facade-layout synthesis under a general purpose solver. In Gilles Pesant, editor, Principles and Practice of Constraint Programming, volume 9255 of Lecture Notes in Computer Science, pages 508–523. Springer International Publishing, 2015.

Barco, A., Vareilles, E., Aldanondo, M., and Gaborit, P. A recursive heuristic for building renovation in smart cities. In Foundations of Intelligent Systems, volume 8502 of Lecture Notes in Computer Science, pages 144–153. Springer International Publishing, 2014.

Battiato, S., Milone, A., and Puglisi, G. Artificial mosaic generation with gradient vector flow and tile cutting. JECE, 2013:8:8–8:8, January 2013.

Bauerly M., and Liu, Y. Effects of symmetry and number of compositional elements on interface and design aesthetics. International Journal of Human-Computer Interaction, 24(3):275–287, 2008.

Bennell, J., Oliveira, JF., and Wascher, G. Cutting and packing. International Journal of Production Economics, 145(2):449–450, October 2013.

Christofides N., and Hadjiconstantinou, E. An exact heuristic for orthogonal 2-d cutting problems using guillotine cuts. European Journal of Operational Research, 83(1):21 – 38, 1995.

Greco, L., and Cascia, M. Saliency based aesthetic cut of digital images. In Alfredo Petrosino, editor, Image Analysis and Processing – ICIAP 2013: 17th International Conference, Naples, Italy, September 9-13, 2013, Proceedings, Part II, pages 151–160. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

Jacobsen, Y., and Hofel, L. Descriptive and evaluative judgment processes: Behavioral and electrophysiological indices of processing symmetry and aesthetics. Cognitive, Affective, & Behavioral Neuroscience, 3(4):289–299, 2003.

McMANUS, IC. Symmetry and asymmetry in aesthetics and the arts. European Review, 13:157–180, 10 2005.

Strecker, T., and Hennig, L. Automatic layouting of personalized newspaper pages. In Bernhard Fleischmann, Karl-Heinz Borgwardt, Robert Klein, and Axel Tuma, editors, Operations Research Proceedings 2008: Selected Papers of the Annual International Conference of the German Operations Research Society (GOR) University of Augsburg, September 3-5, 2008, pages 469–474. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

Tuch, A., Bargas-Avila, J., and Opwis, K. Symmetry and aesthetics in website design: It's a man's business. Computers in Human Behavior, 26(6):1831 – 1837, 2010. Online Interactivity: Role of Technology in Behavior Change.

Vareilles, E., Barco, A., Falcon, M., Aldanondo, M., and Gaborit, P. Configuration of high performance apartment buildings renovation: A constraint based approach. In 2013 IEEE International Conference on Industrial Engineering and Engineering Management, pages 684–688, Dec 2013.

Wascher, G., Haußner, H., and Schumann, H. An improved typology of cutting and packing problems. European Journal of Operational Research, 183(3):1109–1130, December 2007.

Biographies

Andrés F. Barco received a Bs. Eng. in Systems Engineering in 2009 at Universidad del Valle, Cali-Colombia, a MSc degree in Computer Science in 2012 at Pontificia Universidad Javeriana-Cali, Cali-Colombia, and a PhD degree in Informatics and Industrial Engineering in 2016 at École Nationale Supérieure des Mines d'Albi-Carmaux, Albi-France. Currently, he is assistant professor in the Multimedia Engineering program at Universidad de San Buenaventura-Cali, Cali-Colombia. He is in charge of the courses data structures, computer graphics, multimedia databases and vector geometry, all of them directed to undergraduate students. His main research interests include constraint satisfaction, decision support systems, human-computer interaction and configuration. ORCID: orcid.org/0000-0001-6751-3458

Michel Aldanondo is a Full Professor, Director of the Industrial Engineering Laboratory of Toulouse University - Mines Albi, France. He teaches design and operation management courses mainly at the graduate level. Michel concentrates his research on the development of interactive knowledge based aiding design tools and more specifically configuration software. He has directed 12 PhD students and more than 50 master students and published more than 150 articles in journals and conference proceedings..

Élise Vareilles is a Tenured Faculty member at IMT Mines Albi, France and Affiliate Researcher at ETS, Montréal, Canada (2017-2020). Élise received a HDR (Accreditation to supervise Research) in 2015, a PhD in Industrial Engineering in 2005 from Université de Toulouse. She is part of the Industrial Engineering Centre (CGI) at Mines Albi and she is the leader of the ORKID branch that focuses supporting decision-making process in different industrial contexts.