

AD2D: Herramienta para el dibujo automático de distribuciones en planta adimensionales 2D

AD2D: A tool for a automatic drawing of a-dimensional 2D floor plan

M. Álvarez^(*), O. Río^(**), A. Recuero^(***), M. S. Romero^(****)

RESUMEN

En este trabajo se presenta una herramienta integrada y modular, formada por tres módulos, para el dibujo automático de plantas arquitectónicas. El trabajo se centra en el algoritmo en dos pasos que forma parte del método heurístico que permite resolver el problema.

El primer paso consiste en el dibujo de los espacios exteriores y se basa en la equi-distribución de los lados de la planta. El segundo consistente en el dibujo de los espacios interiores se basa en un algoritmo de búsqueda en profundidad con evaluación de nodos intermedios y retroceso, lo que permite evitar procesos aleatorios a la vez que optimizar el proceso a partir de la determinación de las adyacencias entre locales del grafo inicial y tomando como punto de partida las soluciones topológicamente correctas previamente determinadas.

403-22

Palabras clave: Métodos heurísticos, Modelos de Grafos, Diseño arquitectónico, Soluciones topológicas, CAD inteligente.

SUMMARY

An integrated tool with a three modules architecture, for automatic drawing of a-dimensional 2D distributions has been developed and presented in this article. We concentrate on the two-step algorithm by introducing a heuristically method.

As a first step, that deals with the drawing of the external spaces and is based in the equal part division of the plan sides. The second step deals with the drawing of the internal spaces and is based on a heuristic search in depth with backtracking. The search algorithm presented here manages to filter and drive the search. The search then is not a random process, but is driven by the adjacencies between the premises given in the initial graph. The topological structure of solutions for rectangular plans is given as input of process.

Keywords: Heuristics methods, Graph models, Architectural design, Topological solutions, Intelligent CAD.

^(*) Facultad de Informatica (U.P.M), Madrid. España.

^(**) Centro de Seguridad y Durabilidad Estructural y de Materiales (U.P.M-C.S.I.C).

^(***) Instituto de Ciencias de la Construcción Eduardo Torroja (C.S.I.C.), Madrid. España.

^(****) Facultad de Ciencias Economicas y Empresariales (U.N.E.D.), Madrid. España.

1. Grafo asociado con una distribución y diferente distribución asociada con el mismo grafo.

1. INTRODUCCIÓN

Una de las tareas más arduas durante el diseño arquitectónico es crear un esquema básico, una distribución en planta, que muestre la primera concreción espacial y formal de un programa arquitectónico previamente establecido. El proceso es tedioso y consume mucho tiempo debido a que es necesario dibujar muchas alternativas de solución que permitan después de una evaluación seleccionar la óptima. Actualmente existen diversas herramientas CAD que actúan como meros colaboradores no como un agente para automatizar del proceso de diseño, pero no existen herramientas informáticas que automaticen el proceso proporcionando alternativas de distribuciones en planta.

Una distribución en planta es una representación gráfica de un plano arquitectónico con sus espacios y elaciones entre ellos. La teoría de grafos (1-7) es el instrumento matemático utilizado usualmente para representar de forma abstracta los distintos tipos de relaciones funcionales entre las diferentes habitaciones o espacios que constituyen la planta, lo que se denomina grafo de adyacencia. A cada distribución le corresponde un único grafo, pero a cada grafo le pueden corresponder distintas distribuciones sobre una planta, Figura 1.

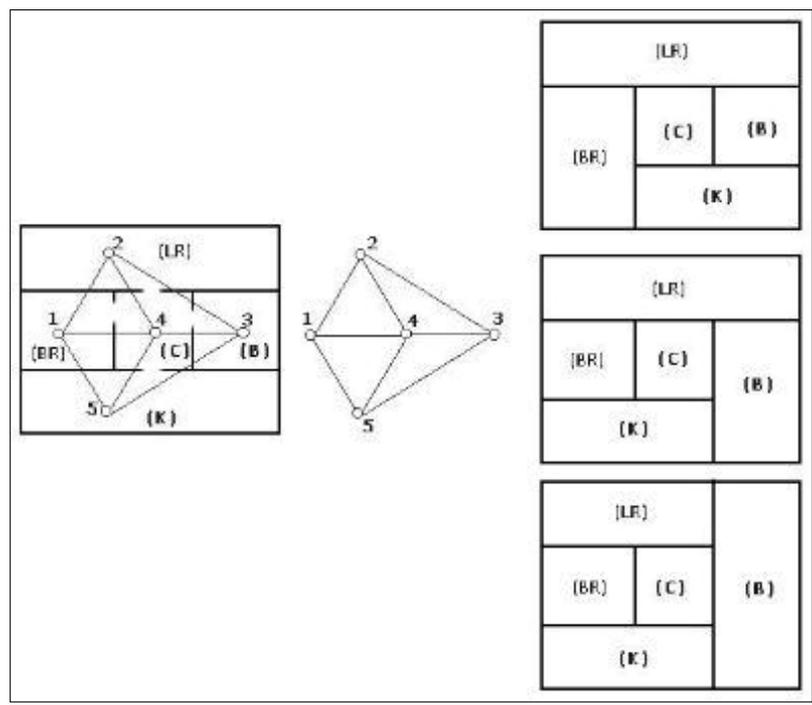
Los autores han desarrollado un proceso (8-12) para la generación automática de distribuciones en planta basado en modelos matemáticos y algoritmos heurísticos de Inteligencia Artificial. Es un método general, que no impone condiciones a priori sobre

las posiciones relativas entre espacios, válido para plantas rectangulares, que son las únicas convexas que pueden ser cubiertas por rectángulos. El proceso se desarrolla en 4 etapas sucesivas:

1. Determinar la realizabilidad de los grafos o grafo sobre una planta elegida.
2. En caso afirmativo, generar todas las posibles soluciones topológicamente correctas.
3. Generar la realización adimensional de las soluciones topológicamente correctas.
4. Imponer el cumplimiento de las condiciones dimensionales, etc.

A su vez se ha estructurado en las siguientes fases:

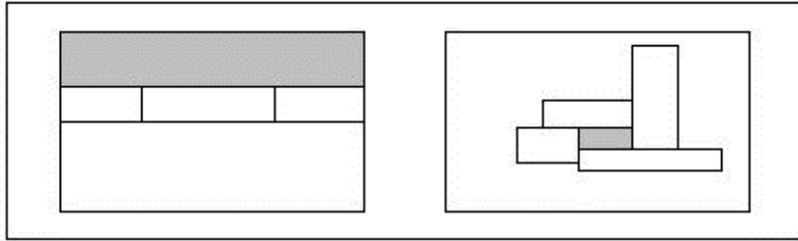
- a) En caso de planta no rectangular reducir la misma a planta rectangular mediante la adición de espacios ficticios adecuados, que a su vez deben ser incluidos en los grafos a comprobar.
- b) Comprobar la realizabilidad del grafo sobre la planta rectangular.
- c) Generar las posibles soluciones topológicamente correctas sobre plantas rectangulares.
- d) Dibujar en forma adimensional las soluciones topológicamente correctas.
- e) Imponer las condiciones dimensionales, de orientación, etc. para dibujar las plantas dimensionadas.



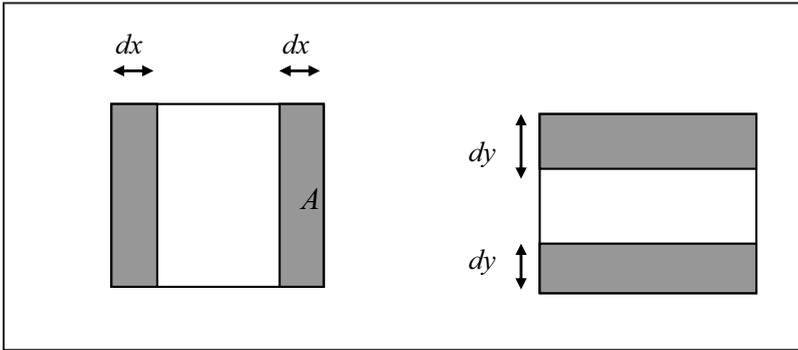
La descripción de las fases a-c ya ha sido presentada de forma teórica en (10-11). En el presente trabajo se aborda la solución de la etapa 3 (fase d del proceso) desarrollando una herramienta informática con un arquitectura de tres módulos. El segundo modulo, el corazón de la herramienta, se ha realizado mediante el desarrollo de dos algoritmos:

1. Para el trazado de los locales exteriores, basado en la división en partes iguales de los lados del rectángulo principal (a partir de ahora la planta).
2. Para el trazado de locales interiores, basado en una técnica de búsqueda heurística en profundidad, con evaluación de nodos intermedios y backtracking, una característica del proceso de resolución de problemas llevada a cabo por el hombre habitualmente.

Están basados en la idea desarrollada por F. Y. Young y D. F. Wong (13-16), un mecanismo de inferencia de proceso



3



4

3. Tipos envoltura real.

4. Tipos de locales.

decisiones sobre los datos de entrada a la herramienta y el lenguaje y software a utilizar para su desarrollo.

2. Módulo general de generación: se puede considerar como el núcleo o corazón del sistema. Se dibujan en forma adimensional las soluciones topológicamente correctas generadas.

3. Módulo de explotación: donde se realizan y se visualizan las consultas realizadas a la herramienta.

A continuación se describe cada uno de los módulos que forman parte de la herramienta.

3.1. El módulo esquema

A su vez se descompone en dos fases: Selección de los datos de entrada y del Software a utilizar en el desarrollo de la aplicación.

3.1.1. Selección de los datos de entrada

Los datos de entrada al problema que deben ser correctos, es decir poder tener representación en un espacio bidimensional, son ver Figura 2:

- Las soluciones topológicamente correctas, desarrolladas por los autores en la etapa c del proceso en base a:
 - *Programa arquitectónico*, un conjunto de locales rectangulares de distinto tamaño, Figura 1 de manera que tapizan una planta también de forma rectangular. Están agrupados siguiendo unos criterios de accesibilidad entre los mismos, los cuales dependen a su vez de la funcionalidad. Los locales deben numerarse de exteriores

a interiores en el sentido de las agujas del reloj empezando por la esquina 1-2.

– *Grafo de Adyacencia*, donde a cada local se le asigna un vértice y las relaciones entre locales son consideradas como ejes del grafo.

- Propiedades de los locales, se debe definir por cada local:

– *Envoltura real de un local*, es el conjunto de locales que tienen contacto con él ya sea en parte de un lado o en un vértice, Figura 3.

– *El tipo de local*

Interior, si los vértices de su envoltura forman un ciclo (*envoltura cerrada*), local E de la Figura 4.

Exterior, si los vértices de su envoltura no forman un ciclo (*envoltura abierta*), locales A, B, C, y D de la Figura 4.

– *Los lados exteriores*, si el local es exterior, 1, 2, 3, 4 de la Figura 4.

– *Si el local es vertical*, el local o locales verticales a él conectados por un vértice.

– *La envoltura ficticia* si el local es interior. Se calcula recursivamente sobre su envoltura real hasta que exista un local tipo D o un local interior previo.

3.1.2. Selección del Software a utilizar

Las características de trabajo, sugieren el uso del paradigma de la programación orientada a objetos como una manera fácil y directa de representar los componentes del espacio de diseño y las relaciones entre ellos. En la implementación de los algoritmos del modulo de generación de la herramienta se ha utilizado el lenguaje C++.

Aunque no se describen en este trabajo se ha implementado una base de datos de locales en MySQL 5.0 para almacenar la información de las plantas generadas. También se ha implementado en C#, lenguaje incluido entre los lenguajes del entorno Visual Studio. NET, una interfaz para la visualización de las soluciones generadas por los algoritmos descritos que se conecta con la base de datos y extrae la información. Debido al uso de C++ en la implementación de los algoritmos, su traslado a C# y a los entornos .NET resultò sencillo por la similitud de sintaxis existente entre estos lenguajes de alto nivel.

3.2. Módulo de generación de locales

Este módulo donde se generan todas las soluciones posibles, se desarrolla a su vez en dos

fases: Tratamiento de los datos de entrada y Trazado de los locales que constituyen la planta.

3.2.1. Tratamiento de los datos de entrada

El input es la información suministrada por el algoritmo de generación de soluciones topológicamente correctas, paso c del proceso general:

- La lista con todos los locales
- El tipo de local, interior o exterior, y si es exterior los lados exteriores
- La envoltura real del local

3.2.1.1. Clasificación de los locales

Dependiendo del número de lados exteriores de una local, número y consecutividad de los mismos se generan cuatro listas de locales de los diferentes tipos ya vistos:

- Locales Tipo A, tres lados exteriores consecutivos: 1-2-3, 2-3-4, 3-4-1, 4-1-2
- Locales Tipo B, dos lados exteriores consecutivos: 1-2, 2-3, 3-4, 4-1
- Locales Tipo D, un lado exterior: 1, 2, 3 y 4
- Locales Tipo E, ningún lado exterior

Los locales tipo C no van a aparecer como parte del problema y no se tienen en cuenta ya que son poco comunes y dividen en dos partes independientes la planta. Debido a esto en pasos previos del proceso se divide el grafo por cada vértice tipo C, dividiendo así el problema inicial en varios subproblemas independientes.

3.2.1.2. Cálculo del número de locales por cada lado exterior

Se calculan las siguientes constantes que serán utilizadas en el trazado de locales exteriores:

- N1, número de locales con un lado exterior en el lado 1.
- N2, número de locales con un lado exterior en el lado 2.
- N3, número de locales con un lado exterior en el lado 3.
- N4, número de locales con un lado exterior en el lado 4.

3.2.2. Trazado de los locales

El método seguido consiste en trazar ordenadamente, primero los locales exteriores y a continuación los interiores.

3.2.2.1. Trazado de los locales exteriores

El principio básico para el trazado de locales exteriores es "colocar los locales exteriores dividiendo los lados exteriores de la planta en partes iguales" que es la forma más sen-

cilla computacionalmente de llevarlo a cabo cuando se desconocen las dimensiones que tiene que tener cada una de las partes.

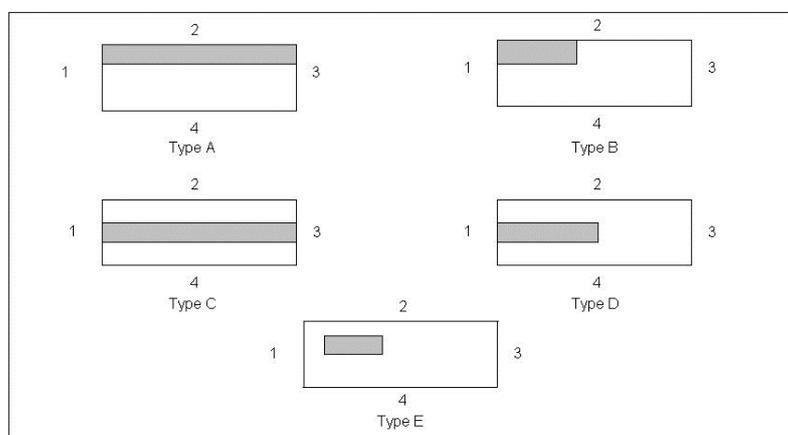
El algoritmo consta de 5 pasos que se realizan de manera consecutiva:

1. Trazado de locales Tipo A
2. Trazado de locales Tipo B
3. Trazado de locales Tipo D
4. Generación de raíces
5. Detección y corrección de huecos y solapamientos

1. Trazado de locales tipo A

Para trazar locales tipo A únicamente es necesario fijar una coordenada, una dimensión del local viene fijada por dos de los vértices de la planta y sólo es necesario fijar una de las dimensiones de cada local (alto o ancho), debido a que ocupan un lado exterior completo de la planta.

Generalizando el procedimiento para cualquier local tipo A, si se supone el local j un local tipo A con lados exteriores l, m, n consecutivos y siendo m el lado exterior que ocupa por completo el local j , se tiene, Figura 5:



5

$$d \text{ coordenada_libre} = \frac{MAX_coordenada_libre}{MAX(Nl, Nn)}$$

Donde:

$coordenada_libre = x$, si m es un lado exterior vertical $| y$, si m es un lado exterior horizontal, según indica la Figura 5.

$dx =$ ancho del local

$dy =$ altura del local

$Nl =$ número de locales que tienen lado exterior en el lado l

$Nn =$ número de locales que tienen lado exterior en el lado n

6. Huecos y solapamientos entre locales exteriores.

7. Trazado de locales tipo B.

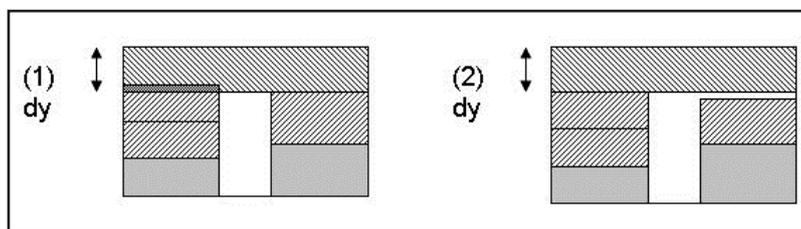
8. Trazado de locales tipo D.

$MAX(Nl, Nn)$ = Máximo entre los dos números anteriores

$MAX_coordenada_libre = (MAX_X \text{ si } dx \text{ o } MAX_Y \text{ si } dy)$ Eq (1).

Como este tipo de locales ocupa parte de dos lados exteriores opuestos, se complica la división en partes iguales en ambos lados, Figura 6:

- Si se hace la división según el lado que menor número de locales tiene, se produce un solapamiento entre locales del lado opuesto.
- Si se realiza según el lado de mayor número de locales, se genera un hueco entre locales exteriores del lado opuesto. Esta solución es la elegida debido a que es más sencilla de tratar.



6

2. TRAZADO DE LOCALES TIPO B

El proceso es más sencillo que el trazado de locales tipo A. En los locales tipo B es necesario fijar solo un valor para la coordenada x y un valor para la coordenada y de cada vértice del local en base a la división en partes iguales de cada uno de los lados exteriores de los que forma parte el local. El resto de coordenadas está predeterminado por la situación del local, son esquinas.

Generalizando el procedimiento para cualquier local tipo B, se supone el local j un local tipo B con lados exteriores m (lado horizontal), n (lado vertical) consecutivos; Figura 7.

$$dx = MAX_X / Nm \quad \text{Eq (2)}$$

$$dy = MAX_Y / Nn, \quad \text{Eq (3)}$$

Donde:

MAX_X = máxima coordenada x en la planta

MAX_Y = máxima coordenada y en la planta

Nm = número de locales con lado exterior en m

Nn = número de locales con lado exterior en n

dx = ancho del local, j

dy = altura del local, j

Por ejemplo, el local de la esquina 1-2, Figura 7:

dx = longitud máxima horizontal partido el número de locales del lado exterior 2

dy = longitud máxima vertical partido entre el número de locales de lado exterior 1.

3. Trazado de locales tipo D

Se determinan los valores de las coordenadas x, y de los vértices del local situados sobre el lado exterior y de la coordenada x de los vértices restantes en el caso de los locales con lado exterior 2 ó 4, o de la coordenada y en los restantes de los locales con lado exterior 1 ó 3. La coordenada que queda por definir se determina más adelante, Figura 8.

Generalizando el procedimiento para cualquier local tipo D, se supone el local j un local tipo D con lado exterior m:

$$d \text{ coordenada_lado} = MAX_coordenada_lado / Nm, \quad \text{Eq (4)}$$

Donde:

$coordenada_lado = x$, si m es un lado exterior horizontal | y , si m es un lado exterior vertical.

dx = ancho del local j | dy = altura del local j

Nm = número de locales que tienen lado exterior en el lado m

$MAX_coordenada_libre = \text{Máximo valor para } x \text{ o } y \text{ dentro de las dimensiones de la planta (MAX_X o MAX_Y respectivamente).}$

4. Generación de raíces

El diseño en planta, englobado dentro del diseño espacial, pertenece a un grupo de problemas para los cuales una búsqueda heurística en espacios de estados, llamados en este caso "raíces-plantas", resulta una solución aceptable.

Para generar un número de distribuciones en planta es necesario generar las raíces donde colocar los locales interiores. Estas raíces se determinan en función del número de los locales tipo D y deben ser iguales al número de estos locales más uno. Los locales tipo D deben ocupar el espacio interior vacío. La ocupación de este espacio se realiza para facilitar la colocación de los locales interiores.

En este caso para el ejemplo de la Figura 1 se deben generar cinco raíces. Este espacio puede ampliarse considerando una sexta raíz generada de la misma forma que la primera, pero tomando como referencia los dos locales restantes del lado exterior de camino mínimo.

1. Generación de la raíz del primer árbol
Se trata de que los locales D del lado exterior de camino mínimo y los locales D de su opuesto ocupen la mayor parte del espacio interior de la planta. El proceso es el siguiente:

1. Se calcula el lado exterior de camino mínimo, que es aquel separado de su opuesto por un menor número de locales.

2. Para los locales de lado exterior el de camino mínimo y para los locales del lado exterior opuesto al de camino mínimo se divide la dimensión total entre 2, Figura 9:

$$d \text{ coordenada_libre} = \text{MAX_coordenada_libre} / 2 \text{ Eq (5)}$$

3. Para el resto de locales se opera igual que para la dimensión libre de los locales tipo B, los locales de las esquinas, de forma que ocupen el menor espacio de entre los 2 posibles. De ahí que se divida entre el máximo de sus lados exteriores consecutivos (anterior, m , y posterior, n).

$$d \text{ coordenada_libre} = \text{MAX_coordenada_libre} / \text{MAX}(Nm, Nn) \text{ Eq (6)}$$

De las ecuaciones se tiene:

$\text{coordenada_libre} = y$ si lado exterior de camino mínimo horizontal (2 ó 4) | x si lado exterior de camino mínimo vertical (1 ó 3).

Por ejemplo, si el lado exterior de camino mínimo es 1, entonces los locales de éste y del lado exterior 3 ocuparan el espacio interior de la planta Figura 9a. Si el lado exterior de camino mínimo es 2, entonces los locales de este y del lado exterior 4, ocuparan el espacio interior de la planta, como se observa en la Figura 9b.

2. Generación de la raíz del segundo árbol

Se pretende que los locales tipo D del lado exterior de camino mínimo ocupen la mayor parte del espacio interior de la planta, Figura 10.

Es decir, se le da la dimensión completa menos la parte correspondiente a los locales del lado exterior opuesto. Así, suponiendo m lado exterior de camino mínimo de local j y, n , l lados exteriores consecutivos a m :

$$d \text{ coo} = dd \text{ coordenada_libre} = \text{MAX_coordenada_libre} - (\text{MAX_coordenada_libre} / \text{MAX}(Nn, Nl)) \text{ Eq (7)}$$

Para el resto de locales se opera de la forma usual dándole la menor dimensión, Figura 10. Es decir, suponiendo el lado exterior distinto de m (n , l o p opuesto de m):

$$d \text{ coordenada_libre} = \text{MAX_coordenada_libre} / \text{MAX}(Nm, Np) \text{ xEq (8)}$$

3. Generación de la raíz del tercer, cuarto y quinto árbol

Para generar el resto de raíces de los tres árboles restantes de soluciones se procede de igual forma que para generar la segunda

raíz, tomando como referencia los siguientes al camino mínimo sucesivamente. Así para la tercera ocuparán el espacio interior los locales del lado exterior siguiente al de camino mínimo, para la cuarta el siguiente al lado tomado para la tercera y para la quinta el siguiente al lado tomado para la cuarta.

Suponiendo m el lado exterior de camino mínimo:

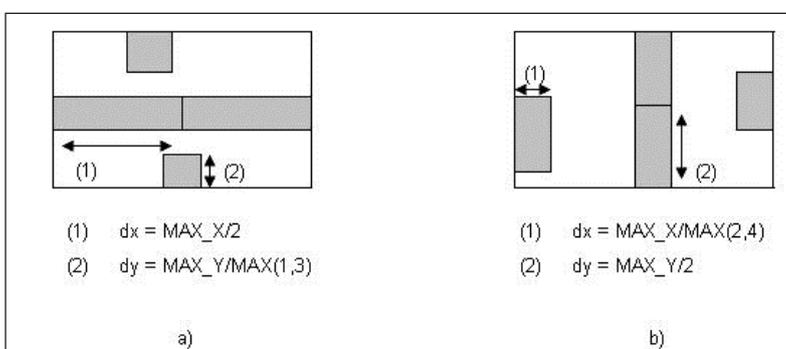
- Tercer árbol: lado exterior referencia = $\text{Sig}(m)$.
- Cuarto árbol: lado exterior referencia = $\text{Sig}(\text{Sig}(m))$.
- Quinto árbol: lado exterior referencia = $\text{Sig}(\text{Sig}(\text{Sig}(m)))$

9. Generación de la primera raíz:

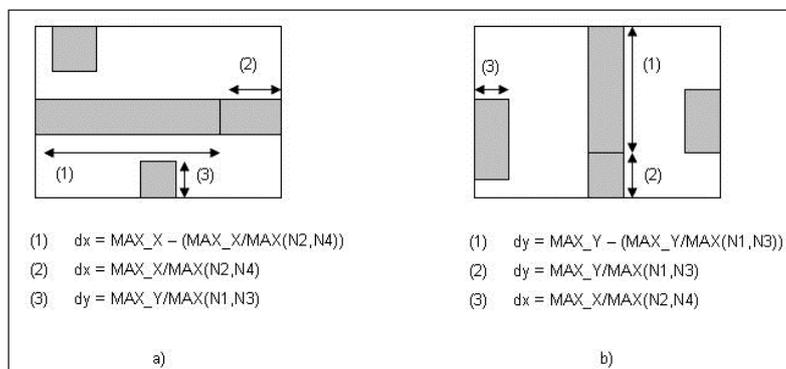
- (a) si lado exterior de camino mínimo 1 ó 3.
(b) si lado exterior de camino mínimo 2 ó 4.

10. Generación de la segunda raíz:

- (a) si 2 lado exterior de camino mínimo 2.
(b) si 1 lado exterior de camino mínimo 1.



9



10

Se consigue que consecutivamente todos los locales tipo D participen en la ocupación del espacio interior.

4. Detección y Corrección de huecos y solapamientos

Se trata de corregir algún hueco o solapamiento entre locales al trazar locales tipo B y tipo D. Para poder detectar los huecos se deben seguir los pasos siguientes:

1. Estudiar la consecutividad de los vértices exteriores de los locales.
2. Corregir el hueco que se forma si se encuentra algún vértice exterior que pertenece a un único local y no es vértice de la planta.

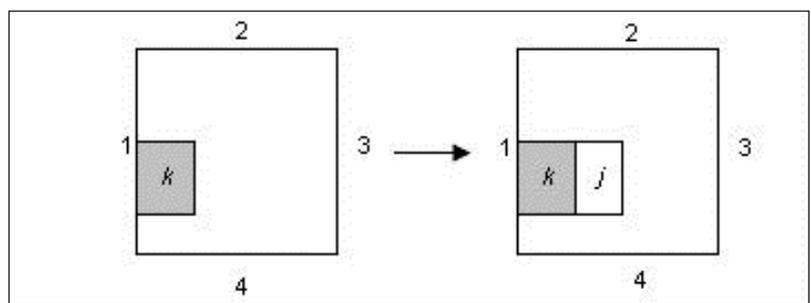
11. Colocación de local interior en local exterior tipo D de lado exterior 1.

12. Colocación del local interior j dentro del local interior k.

Para ello debe aumentarse la dimensión correspondiente al local del vértice detectado hasta alcanzar un lado del siguiente local. La detección de solapamientos es un proceso más complicado, sólo puede darse entre locales tipo D con locales tipo B de lado exterior opuesto.

Para poder detectar los solapamientos se debe seguir los pasos siguientes:

1. Estudiar la colocación de los vértices interiores de cada local. Si existe algún vértice interior que cae dentro del área de otro local entonces hay solapamiento.
2. Corregir el solapamiento disminuyendo la dimensión del local hasta conseguir salir del área del otro local.

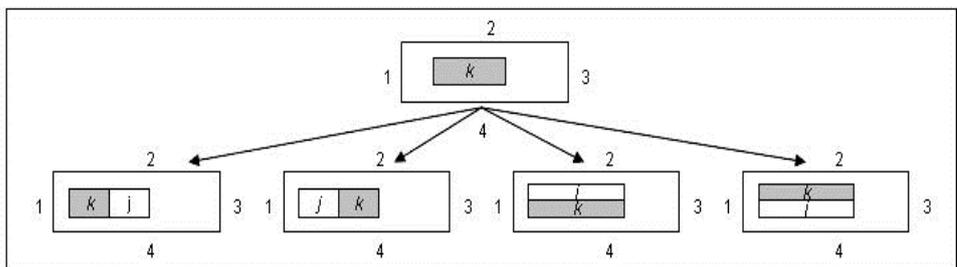


11

exterior 2 ocupará la mitad inferior, de lado exterior 3 la mitad izquierda y lado exterior 4 la mitad superior.

Por el contrario cuando se coloca un local interior en otro local interior puede ocupar cualquiera de las posiciones dentro de él: izquierda, derecha, inferior o superior, Figura 12.

En este trabajo para trazar los locales interiores, se ha desarrollado un algoritmo de dos pasos empleado procedimientos heurísticos. Es decir, se busca la solución o soluciones a través de una secuencia de estados, con una evaluación en cada paso, basada en el estado parcial de los datos especificados. El proceso se representa en un árbol estado-acción, donde el vértice raíz representa el estado inicial de los datos, las ramas son operaciones alternativas que pueden llevarse a cabo, los nodos intermedios son soluciones parciales y los nodos finales soluciones definitivas. Se ha usado el *backtracking* con el objeto de solucionar situaciones de bloqueo en el diseño, esto es, situaciones en las que, a partir del diseño alcanzado, sea imposible avanzar mediante refinamientos hasta conseguir un diseño que cumpla las restricciones.



12

3.2.2.2. Trazado de locales interiores

El principio básico para el trazado de los locales interiores o locales tipo E es que se tratará de "colocar los locales interiores en locales exteriores tipo D o en locales interiores previamente trazados" siempre que no sean locales verticales al local interior a colocar. Por este motivo se ha ocupado el espacio interior de la planta con los locales tipo D en la generación de raíces.

Antes de entrar a desarrollar el algoritmo para el trazado de locales interiores es necesario conocer las diferentes posiciones que puede ocupar un local interior en un local exterior tipo D o en otro local interior.

En un local exterior tipo D solo puede colocarse un local tipo E en la mitad opuesta a su lado exterior. Es decir, si el local es de lado exterior 1 el local interior ocupará su mitad derecha Figura 11, si es de lado

El algoritmo consta a su vez de tres pasos:

1. Búsqueda, que va generando los distintos nodos del árbol.
2. Corrección, que opera sobre los nodos generados en la búsqueda.
3. Evaluación de cada uno de los nodos, de cada uno de los nodos. Poda y retroceso, *backtracking* cuando proceda.

1. Búsqueda

El algoritmo de búsqueda empleado, basado en búsqueda heurística en profundidad va generando los diferentes caminos de cada árbol de manera recursiva, a partir de las cinco raíces generadas, obteniéndose cinco árboles estado-acción.

Por cada local tipo D y por cada local interior E pertenecientes a la envoltura real del local interior a colocar se obtiene un nodo

dentro del nivel del árbol correspondiente a dicho local. En el caso de que la envoltura real del local interior no tenga locales tipo D o locales interiores ya trazados en la planta debe calcularse su envoltura ficticia y seguir el mismo procedimiento.

Lo que se hace es aplicar recursividad sobre la lista de locales interiores a colocar, de modo que para cada local de la lista se llevan a cabo los pasos indicados en el diagrama de flujo de la Figura 13. La raíz será la planta con el trazado de los locales exteriores, parte de los cuales ocupan el espacio interior. Los nodos intermedios se van generando a medida que se colocan los locales interiores, por lo que cada nivel intermedio del árbol se corresponde con la colocación de un local interior en la planta. Los nodos finales de nivel *n* son las soluciones finales obtenidas tras la corrección de la planta final, con todos los locales ya trazados, de tal forma que cumpla con los requisitos de entrada, adyacencias entre locales.

Existe una excepción a la hora de realizar la búsqueda y colocar el primer local interior, cuando no existen locales tipo D en la planta y este caso se reduce a la configuración mostrada en la Figura 14.

En este caso, la raíz con los locales exteriores que se genera, es la planta dividida en cuatro partes ocupando todo el espacio, Figura 14 (caso A). Para resolver esto, se hace un hueco interior rectangular sobre el que se colocará el primer local interior. A partir de aquí el procedimiento de búsqueda se normaliza.

El hueco interior se logra:

- disminuyendo el área de la esquina 1-2 y aumentando la 2-3 (Caso b)
- disminuyendo el área de la esquina 2aumentando la 3-4 (Caso c)
- disminuyendo el área de la esquina 3-4 y aumentando la 4-1 (Caso d)
- disminuyendo el área de la esquina 4-1 y aumentando la 1-2 (Caso e)

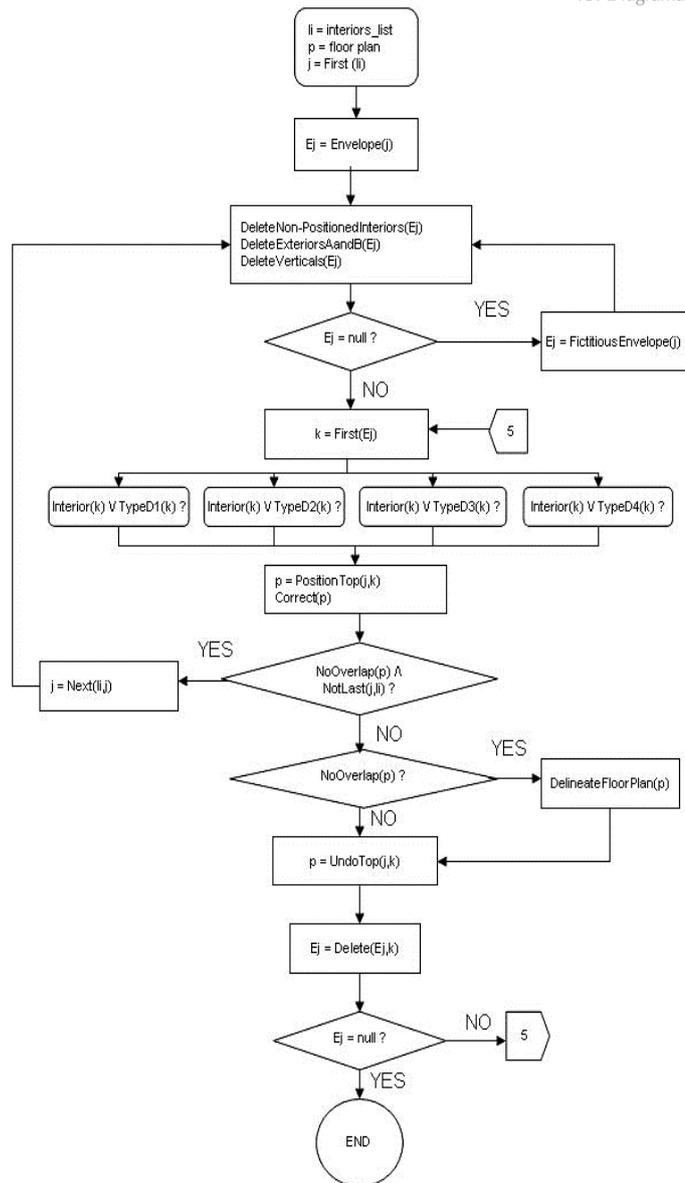
En este caso solo se genera un árbol de soluciones de profundidad:

$$profundidad = n^{\circ} \text{ locales interiores} + 1$$

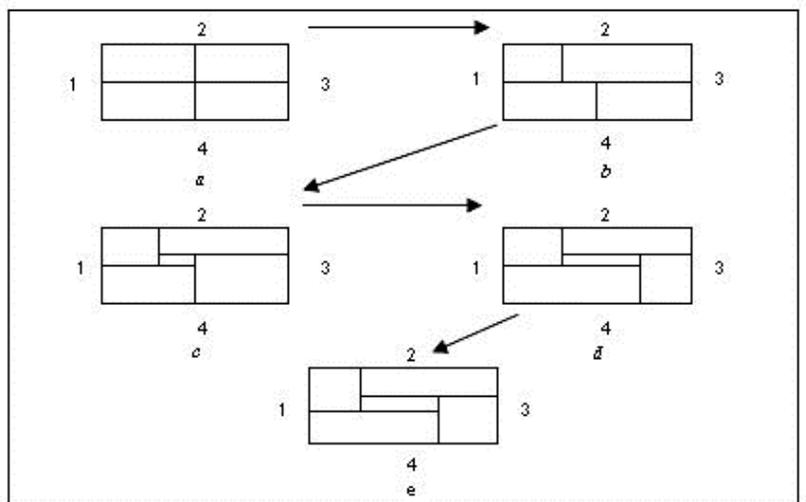
En este caso excepcional solamente una raíz es la entrada al proceso. En los demás casos el árbol de soluciones tiene una profundidad:

$$profundidad = n^{\circ} \text{ locales interiores} + 2(\text{raíz} + \text{hojas del árbol})$$

13. Diagrama de flujo.



14. Caso de excepción.



15. Caso de estudio.

2. Corrección

La distintas configuraciones que se van formando mediante el algoritmo de búsqueda, no llegan por si solas a generar una solución. El algoritmo de corrección trata de encaminar las soluciones potenciales hacia una solución final, de modo que cumpla el mayor número de adyacencias de entrada entre los locales ya colocados. Para ello se deben corregir las envolturas de los locales colocados en cada una de ellas, y decidir posteriormente si se continúa o no con la búsqueda de solución por cada rama (aplicación de la condición de evaluación).

De este modo será necesario:

- Hacer que locales que NO deben ser adyacentes en la planta, y se han trazado como adyacentes dejen de serlo. Para ello será necesario disminuir una dimensión de uno de los locales implicados. En este caso se va a procurar modificar, siempre que se pueda, el local objeto de estudio.
- Corregir los locales que deben ser adyacentes y se hayan trazado como no adyacentes. Para ello se debe aumentar la dimensión o dimensiones, en algunos casos, del local que no es objeto de estudio, en este caso, para evitar caer en un bucle infinito.

3. Condición de evaluación. poda o *backtracing*

Es necesario acotar el espacio generado por los distintos árboles estado-acción, de forma que se reduzca la complejidad del sistema. Es necesario un algoritmo de evaluación, que sea capaz de podar aquellas ramas del árbol que no generen solución al problema inicial tras el estudio de sus nodos intermedio.

El proceso que debe seguirse implica:

- Evaluar cada nodo obtenido como resultado de la aplicación del Algoritmo de Corrección para determinar si es potencialmente una solución o no.
- Podar y eliminar del proceso de búsqueda la rama del árbol correspondiente, cuando

en uno de los trazados (intermedios o final) de los obtenidos durante el proceso de corrección, existe un local completamente solapado.

- Evaluar después de la corrección parcial de cada nodo intermedio estudiando, “*si durante el proceso de corrección de la envoltura en planta de un local, ha sido necesario ampliar el alto o ancho de uno de los implicados, de forma que solape por completo a otro de los ya trazados*”, podando esa rama.
- Hacer *backtracking* para retroceder en la búsqueda comenzando la exploración de una nueva rama.

Así se obtienen 5 espacios de soluciones formados por los nodos hoja de los 5 árboles estado-acción.

3.3. Modulo de explotación

Hasta que la herramienta se integre con las realizadas anteriormente a ella en el desarrollo de los pasos del proceso completo de trazado automático de plantas, de las que tomara los datos de entrada, es necesario introducir los datos desde teclado en muestra la interfaz de usuario, una ventana de petición de datos. Una vez introducidos los datos no se le permite más interactividad con el sistema.

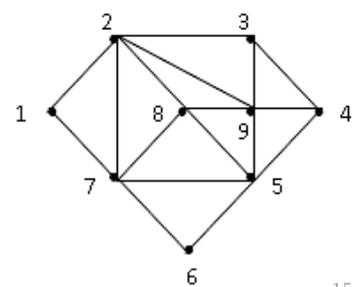
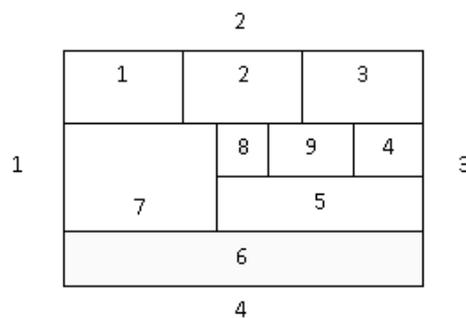
Para finalizar la introducción de datos de un local o indicar que un local es interior basta pulsar 0.

4. EJEMPLO DE EJECUCIÓN

A continuación se explica el funcionamiento de la aplicación de generación de distribuciones en plantas adimensionales para el ejemplo de la Figura 15.

4.1. Modulo Esquema

El usuario solicitará al sistema que genere los trazados de las distribuciones en planta correspondiente al grafo representado en la Figura 15. La Figura 15a muestra el



número de locales y la Figura 15b el grafo de adyacencia con la relación entre los locales.

4.1.1. Datos de Partida

En la en la Tabla 1 se muestran los datos de partida. Se deben indicar los locales que son verticales. Como en este caso no hay locales verticales en las plantas soluciones, no se ha reflejado información sobre la propiedad de verticalidad entre locales. Para indicar que un local es interior, se fija el 0 como lado exterior.

Locales	Lados exteriores	Envoltura
1	1-2	2-7
2	2	1-3-7-8-9
3	2-3	2-4-9
4	3	3-5-9
5	3	4-6-7-8-9
6	3-4-1	5-7
7	1	1-2-5-6-8
8	0	2-5-7-9
9	0	2-3-4-5-8

4.1.2. Modulo de generación de locales

Una vez entrados los datos se procede a realizar las siguientes operaciones:

4.1.2.1. Clasificación de locales

La clasificación de los datos sobre locales, en función del número y consecutividad de los lados exteriores de cada local, es la siguiente:

Tipo A:

3 lados exteriores consecutivos: 3-4-1, local 6

Tipo B:

2 lados exteriores consecutivos: 1-2 local 1
2 lados exteriores consecutivos: 2-3 local 3,

Tipo D:

1 lado exterior: 2 local 2,
1 lado exterior: 3 local 4
1 lado exterior: 3 local 5
1 lado exterior: 1 local 7

Locales Tipo E:

0 lados exteriores: local 8 y 9

Por lo tanto el orden de colocación de locales en la planta será: 6, 1, 3, 2, 4, 5, 7, 8 y 9.

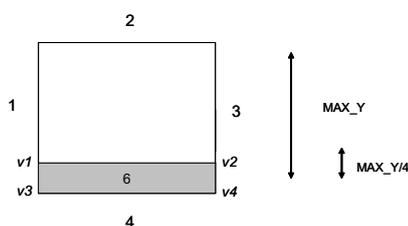
4.1.2.2. Cálculo del número de locales por cada lado exterior

La clasificación del número de locales por lado exterior es la siguiente:

$N1 = 3$, los locales 1, 6 y 7 tienen como lado exterior el 1

$N2 = 3$, los locales 1, 2 y 3 tienen como lado exterior el 2

$N3 = 4$, los locales 3, 4, 5 y 6 tienen como lado exterior el 3



16

$N4 = 1$, sólo el local 6 tiene como lado exterior el 4

4.1.2.3. Trazado de locales exteriores

Son los locales 6, 1, 3, 2, 4, 5 y 7.

a) Local 6 de lados exteriores 2, 4, 1.

Su trazado se muestra en la, Figura 16.

Las coordenadas iniciales de los cuatro vértices son:

$v1.x = v3.x = 0.0$, por la propiedad de rectangularidad del local
 $v2.x = v4.x = MAX_X$, por la propiedad de rectangularidad del local

$v3.y = MAX_Y$

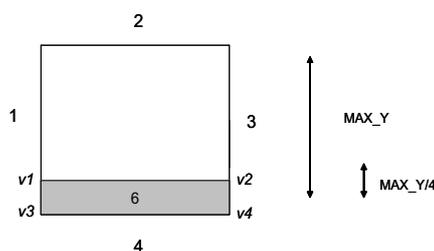
$v4.x = MAX_X$

$v4.y = MAX_Y$

La coordenada $v1.y$ es:

$v1.y = MAX_Y - (MAX_Y / MAX(N1, N3))$

Como $N1 = 3$, $N3 = 4 \rightarrow MAX(N1, N3) = N3 = 4$



17

18. Trazado de los locales tipo D (2, 4, 5 y 7).

$$v1.y = MAX_Y - (MAX_Y / N3)$$

$$v1.y = MAX_Y - (MAX_Y / 4)$$

$$v2.y = v1.y, \text{ por la propiedad de rectangularidad del local}$$

4.1.2.4. Trazado de locales tipo B

Son los locales 1y 3. Sus trazados se muestran en la Figura17.

a) Local 1de lados exteriores 1-2.

Las coordenadas iniciales de los cuatro vértices son:

Para los lados exteriores 1 y 2, N1 = 3 y N2 = 3

$$v1.x = v3.x = 0.0$$

$$v1.y = v2.y = 0.0$$

$$v2.x = v4.x = MAX_X / 3$$

$$v3.y = v4.y = MAX_Y / 3$$

b) Local 3 con lados exteriores 2-3

Las coordenadas iniciales de los cuatro vértices son:

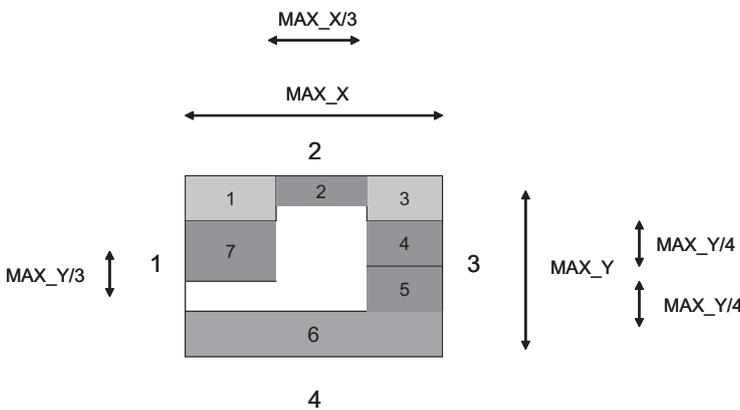
Para los lados exteriores 2 y 3, N2 = 3 y N3 = 4

$$v1.x = v3.x = MAX_X - (MAX_X / 3)$$

$$v1.y = v2.y = 0.0$$

$$v2.x = v4.x = MAX_X$$

$$v3.y = v4.y = MAX_Y / 4$$



18

4.1.2.5. Trazado de locales Tipo D

Son los locales 2, 4, 5 y 7. Su trazado se muestra en la Figura 18. Se determina una sola de las dos dimensiones, la otra se calculara posteriormente.

a) Local 2 con lado exterior 2.

Se coloca en primer lugar. Las coordenadas iniciales son:

$$v1.x = v3.x = v2(\text{local } 1).x$$

$$v1.y = v2.y = v2(\text{local } 1).y = 0.0$$

$$v2.x = v4.x = v2(\text{local } 1).x + MAX_X/N2 = v2(\text{local } 1).x + MAX_X/3$$

b) Local 4 con lado exterior 3.

Las coordenadas iniciales:

$$v2.x = v4.x = v4(\text{local } 1).x = MAX_X$$

$$v2.y = v1.y = v4(\text{local } 1).y$$

$$v4.y = v3.y = v2.y + MAX_Y/N3 = v2.y + MAX_Y/4$$

c) Local 5

Se conocen las coordenadas del vértice 2, que serán iguales a las coordenadas del vértice 4 del local 4.

Las coordenadas iniciales son:

$$v2.x = v4.x = v4(\text{local } 4).x = MAX_X$$

$$v2.y = v1.y = v4(\text{local } 4).y$$

$$v4.y = v3.y = v2.y + MAX_Y/N3 = v2.y + MAX_Y/4$$

d) Local 7 con lado exterior 1

Las coordenadas iniciales son:

$$v3.x = v1.x = v1(\text{local } 6).x = 0.0$$

$$v3.y = v4.y = v1(\text{local } 6).y$$

$$v1.y = v2.y = v3.y - MAX_Y/N1 = v3.y - MAX_Y/3$$

4.1.2.6. Generación de Raíces

Falta por determinar la dimensión y del local 2 y la dimensión x de los locales tipo D (4, 5 y 7). El dar una dimensión u otra a cada uno de estos locales determina las 5 raíces de cada uno de los árboles de soluciones.

a) Raíz del primer árbol de soluciones

Determina la coordenada libre de los locales de lado exterior de camino mínimo y los del opuesto. Para ello se debe determinar el lado exterior de camino mínimo estudiando las envolturas de cada uno de los locales . En este caso se tiene:

- 1) Lado exterior 1
 - distancia entre local 1 y local 3: 2 (1-2 y 2-3)
 - distancia entre local 1 y local 4: 3
 - distancia entre local 1 y local 5: 3
 - distancia entre local 7 y local 3: 2
 - distancia entre local 7 y local 4: 3
 - distancia entre local 7 y local 5: 1

- 2) Lado exterior 2
 - distancia entre local 1 y local 6: 2
 - distancia entre local 2 y local 6: 2
 - distancia entre local 3 y local 6: 3

- 3) Lados exteriores 3 y 4

El estudio sería equivalente a la de los lados exteriores 1 y 2, pero como se va a tomar el menor, no es necesario llevarlo a cabo.

Se obtiene así que el lado exterior de camino mínimo es el lado exterior 1 al que

pertenece el local 7 y su opuesto es el lado exterior 3, ya que la distancia mínima entre dos locales es 1, que es la distancia entre el local 7 y el local 5.

Aplicando el algoritmo las coordenadas resultantes son:

$$\begin{aligned} v_{3,2}.y &= v_{4,2}.y = MAX_Y/N3 = MAX_Y/4 \\ v_{1,4}.x &= v_{3,4}.x = MAX_X/2 \\ v_{1,5}.x &= v_{3,5}.x = MAX_X/2 \\ v_{2,7}.x &= v_{4,7}.x = MAX_X/2 \end{aligned}$$

En la Figura 19I se puede ver la raíz inicial del primer árbol de soluciones para el ejemplo dado en la Figura 15.

b) Generación de la raíz del segundo árbol de soluciones

Las coordenadas resultantes de aplicar el algoritmo son:

$$\begin{aligned} v_{3,2}.y &= v_{4,2}.y = MAX_Y/N3 = MAX_Y/4 \\ v_{1,4}.x &= v_{3,4}.x = MAX_X - (MAX_X/N2) = MAX_X - (MAX_X/3) \\ v_{1,5}.x &= v_{3,5}.x = MAX_X - (MAX_X/N2) = MAX_X - (MAX_X/3) \\ v_{2,7}.x &= v_{4,7}.x = MAX_X - (MAX_X/N2) = MAX_X - (MAX_X/3) \end{aligned}$$

En la Figura 19II se puede ver la raíz inicial del segundo árbol de soluciones para el ejemplo dado.

Al igual que en la primera raíz se debe corregir el hueco existente entre los locales 6 y 7, para generar definitivamente la raíz del segundo árbol de soluciones.

c) Generación de la raíz del tercer árbol de soluciones

Las coordenadas resultantes de aplicar el algoritmo son:

$$\begin{aligned} v_{3,2}.y &= v_{4,2}.y = MAX_Y - (MAX_Y/N3) = MAX_Y - (MAX_Y/4) \\ v_{1,4}.x &= v_{3,4}.x = MAX_X - (MAX_X/N2) = MAX_X - (MAX_X/3) \\ v_{1,5}.x &= v_{3,5}.x = MAX_X - (MAX_X/N2) = MAX_X - (MAX_X/3) \\ v_{2,7}.x &= v_{4,7}.x = MAX_X/N2 = MAX_X/3 \end{aligned}$$

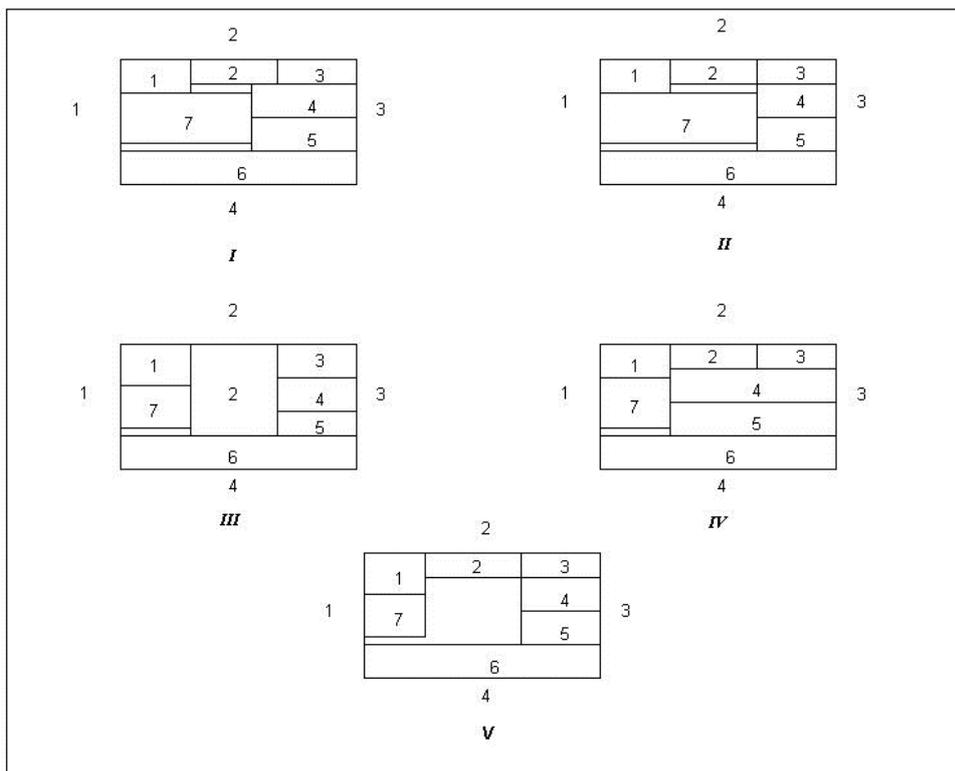
En la Figura 19III se puede ver la raíz inicial del tercer árbol de soluciones para el ejemplo dado.

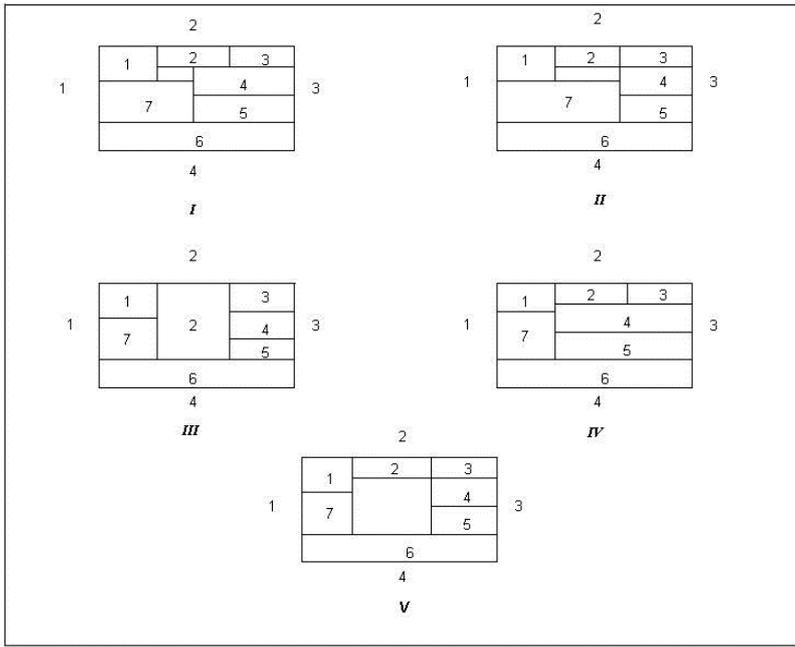
d) Generación de la raíz del cuarto árbol de soluciones

Las coordenadas resultantes de aplicar el algoritmo son:

$$\begin{aligned} v_{3,2}.y &= v_{4,2}.y = MAX_Y/N3 = MAX_Y/4 \\ v_{1,4}.x &= v_{3,4}.x = MAX_X/N2 = MAX_X/3 \\ v_{1,5}.x &= v_{3,5}.x = MAX_X/N2 = MAX_X/3 \\ v_{2,7}.x &= v_{4,7}.x = MAX_X/N2 = MAX_X/3 \end{aligned}$$

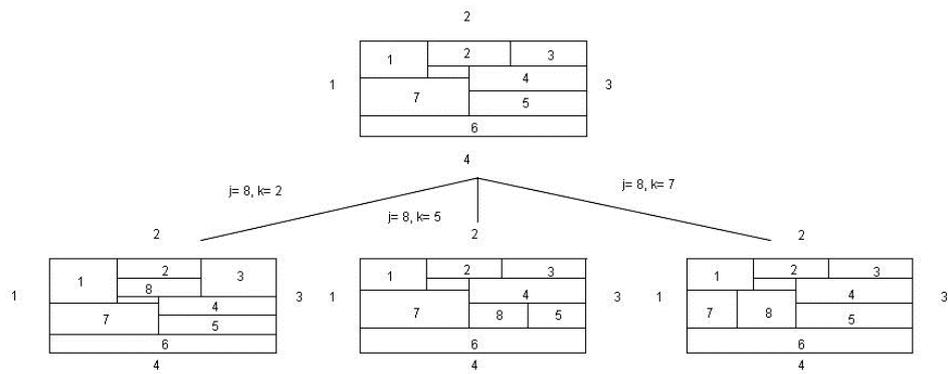
19. Raíces iniciales de los 5 árboles.





20. Raíces finales de los 5 árboles.

21. Primer árbol de búsqueda.



21

En la Figura 19IV se puede ver la raíz inicial del cuarto árbol de soluciones para el ejemplo dado.

e) *Generación de la raíz del quinto árbol de soluciones*

Las coordenadas resultantes de aplicar el algoritmo son:

$$\begin{aligned}
 v3_{2,y} &= v4_{2,y} = \text{MAX}_Y/N3 = \text{MAX}_Y/4 \\
 v1_{4,x} &= v3_{4,x} = \text{MAX}_X - (\text{MAX}_X/N2) = \\
 &= \text{MAX}_X - (\text{MAX}_X/3) \\
 v1_{5,x} &= v3_{5,x} = \text{MAX}_X - (\text{MAX}_X/N2) = \\
 &= \text{MAX}_X - (\text{MAX}_X/3) \\
 v2_{7,x} &= v4_{7,x} = \text{MAX}_X/N2 = \text{MAX}_X/3
 \end{aligned}$$

En la Figura 19V se puede ver la raíz inicial del quinto árbol de de soluciones para el ejemplo dado.

4.1.2.7. Detección y corrección de huecos y solapamientos

En cada uno de los cinco casos de la Figura 19 existe el mismo hueco entre los locales 6 y 7. El siguiente paso es corregir los huecos

exteriores de las raíces, para generar la forma definitiva las 5 raíces de cada uno de los 5 árboles de soluciones que genera el algoritmo, Figura 20.

En cada una de ellas se tiene:

$$v3_{7,y} = v4_{7,y} = v1_{6,y}$$

Se puede observar que no se han producido solapamientos en ellos y no es preciso corregir solapamientos entre locales exteriores.

4.1.2.8. Trazado de locales interiores

Existen dos locales interiores tipo E, el 8 y el 9. En este trabajo solo se va a exponer el trazado del local 8. De manera análoga se procedería para trazar el local 9.

4.1.2.8.1. Búsqueda aplicada al local 8

Se parte de la envoltura real del local interior 8 que es: 2-5-7-9

Se siguen los pasos del algoritmo para obtener, dentro del nivel del árbol correspondiente a cada local tipo D y por cada local interior perteneciente a la envoltura real del local, los árboles de búsqueda para cada una de las raíces generadas anteriormente. Para simplificar el ejemplo se va a indicar solamente el primer árbol de búsqueda que se muestra en la Figura 21.

4.1.2.8.2. Corrección aplicada al local 8

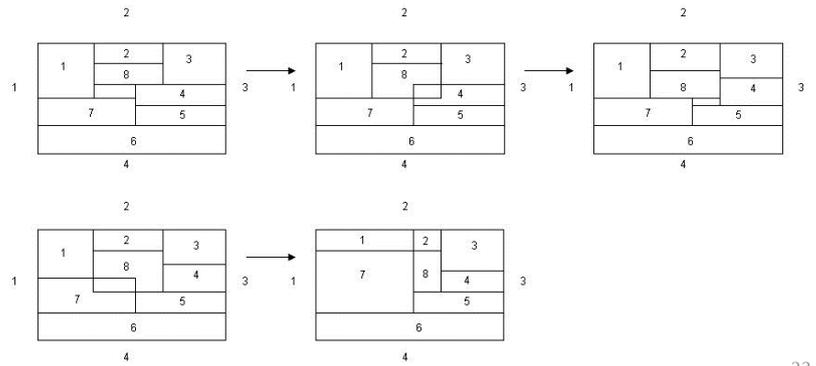
Sobre las plantas generadas en el último nivel de cada árbol, se aplica el algoritmo de corrección.

Para simplificar el ejemplo se va a indicar solamente la corrección de envolturas de la primera rama del primer árbol de búsqueda, Figura 22. El local 8 debe ser adyacente a los locales 2, 5 y 7, según indica la envoltura dada para dicho local en los datos de entrada.

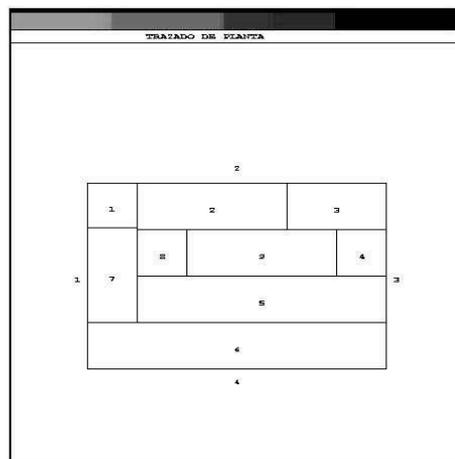
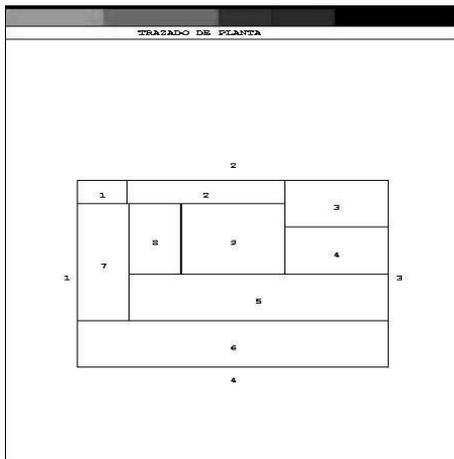
4.1.2.8.3. Condición de evaluación, poda o Backtracing aplicada al local 8

Como en el ejemplo de la Figura 22 no existe ningún local solapado completamente por otro local en ninguna de las fases de corrección, la rama correspondiente no debe ser podada.

Si se siguiera con el proceso sobre todas las ramas de los árboles de búsqueda se podría ver que esta solución no es única, sólo es la primera de ellas en el espacio de soluciones.



22



23

22. Proceso de corrección de la primera rama del primer árbol

23. Trazados generados por el programa

4.3. Visualización de los resultados

Al finalizar todo el proceso de búsqueda heurística, la herramienta devuelve las soluciones encontradas mostrando en pantalla en el orden en que se van obteniendo. En la Figura 23 se muestran los dos primeros trazados de planta obtenidos en la ejecución por el programa para el ejemplo de la Figura 15.

5. DISCUSIÓN DE RESULTADOS

La herramienta presentada en este documento es una primera aproximación para conseguir automatizar el proceso de dibujo de distribuciones en planta. Esto quiere decir que existe la posibilidad de refinar más aún la búsqueda heurística a través de los árboles estado-acción, si se utilizan todas las restricciones impuestas por el usuario en cuanto a dimensión, y condiciones de accesibilidad de cada uno de los locales de la planta en la poda de ramas durante la búsqueda de soluciones. Esto permite acotar más el espacio de soluciones y por lo tanto, facilitar la labor del arquitecto en el proceso de diseño arquitectónico.

Tanto la imposición de las restricciones de dimensiones, como el uso de las condiciones de accesibilidad entre locales, pueden integrarse en la etapa de poda del algoritmo.

Actualmente, durante esta etapa se evalúan condiciones que pueden no llegar a generar ninguna solución a partir de un estado determinado del trazado, o que puede generar soluciones idénticas a otras ya obtenidas por caminos más sencillos en el árbol. Si además de tener en cuenta la generación de solución por la rama de un árbol durante la evaluación de un nodo intermedio, estado intermedio del trazado de planta, se amplía esta evaluación de forma que se estudie si, cada uno de los locales del trazado representado en dicho nodo cumple las restricciones de dimensión y de accesibilidad entre locales impuestas por el usuario, se conseguirá una solución ya no adimensional, si no dimensionada, aunque no sea la solución óptima.

La imposición de restricciones es muy eficaz en el caso de llegar a encontrar soluciones, pero sin embargo, puede llegar a perjudicar al algoritmo de forma que al acotar tanto el espacio de soluciones no se encuentre ninguna solución válida durante la búsqueda heurística.

Se puede optimizar el proceso a fin de conseguir la mejor solución posible de entre los trazados automáticos, resultado de su ejecución usando lenguajes de restricciones como Prolog y otras técnicas, como algoritmos genéticos y técnicas multicriterio.

6. CONCLUSIONES

El proyecto de automatizar el proceso de obtención de trazados de distribuciones en planta es difícil de llevar a cabo debido a la complejidad del mismo. Sin embargo se pueden desarrollar herramientas usando la teoría de grafos y algoritmos de búsqueda heurística, para llegar a obtener diferentes soluciones válidas para un problema de entrada.

La herramienta para el trazado automático de esquemas adimensionales en planta, presentado en este documento es una primera aproximación. Tiene una arquitectura de tres módulos siendo el núcleo o corazón del sistema el módulo segundo, desarrollado mediante un algoritmo en dos pasos que imita el proceder humano.

Por un lado el algoritmo logra filtrar y encaminar la búsqueda de forma que no sea una búsqueda aleatoria o de "fuerza bruta", sino que sea una búsqueda guiada por las adyacencias entre locales dadas en el grafo inicial. Esto hace que sea más potente o más fuerte dentro de los algoritmos heurísticos vistos hasta el momento, sin alejarse de ser un algoritmo general.

Es un algoritmo potente y bastante general, a pesar de que se hace un refinamiento exhaustivo en la búsqueda, pero no está probado que sea válido para cualquier problema inicial que sirva como entrada a los pasos previos. A pesar de esto, deja abierto un camino optimista para retomar los métodos heurísticos como solución al problema arquitectónico.

BIBLIOGRAFÍA

- (1) Balacrishnan, V. K.: "Introductory discrete mathematics". New York: Dover, 1991.
- (2) Chattrand, G.; Oellermann, O. R.: "Applied and algorithmic graph theory", New York: McGraw-Hill, 1993.
- (3) Recuero, A.; Río, O.; Álvarez, M.: "Aplicación de la teoría de grafos a la planificación y programación de proyectos". *Informes de la Construcción*, 46, 431, 1994, 49-60.
- (4) Recuero, A.: "Algorithms for path searching and graph connectivity analysis". *Advances in Engineering Software*, 23, 1995.
- (5) Balacrishnan, V. K.: "Graph Theory". New York: Schaum's Outline Series, 1997.
- (6) Harary, F.: "Graph Theory". New Delhi: Narosa Publishing House, 1999.
- (7) Saidur Rahman, Md.; Takao, N.; Shubhashis, G.: "Rectangular drawings of planar graphs" *J. Algorithms*, 50, 1, 2004, 62-78.
- (8) Recuero, A.; Álvarez, M.; Río O.: "Mapping a graph into a given rectangular plan (Realización de un grafo en recintos rectangulares sobre una planta definida). Published in Spanish only". *Informes de la Construcción*, 47, 437, 1995, 63-85.
- (9) Recuero, A.; Álvarez, M.; Río, O.: "Partition of a rectangle into rectangles. An application to architectural design". Second Workshop EG-SEA-AI, Bergamo, 1995.
- (10) Recuero, A.; Río, O.; Álvarez, M.: "Heuristic method to check the realisability of a graph into a rectangular plan". *Advances in Engineering Software*, 31, 2000, 223-331.
- (11) Álvarez, M.; San José, M.; Río, O.; Recuero, A.: "Use of a Adjacency Graph in Computer Aided drawing of floor plans", *Conference on Construction Applications of Virtual Realty, ADETTI-ISCTE*, Lisboa, 2004.
- (12) Recuero, A.; Río, O., Álvarez M.: "Realisations of a Graph into Rectangles covering a Rectangle: Building floor Plan Design". *7th International Conference on Computational Structures Technology*. Edited by BHV Topping and Z. Bittnar, Lisboa, Portugal Cap. IX, September 2004, 139-140
- (13) Young, F. Y.; Wong, D. F.; Yang H.: "Slicing floor plans with range constraint". *IEEE Transaction of Computer-Aided Design of Integrated Circuits and Systems*, 19, 2000, 272-278
- (14) Young, F. Y.; Wong, D. F.; Yang, H.: "Slicing floor plans with Boundary Constraints", *IEEE Transaction of Computer-Aided Design of Integrated Circuits and Systems*. 18, 1999, 1385-1389.
- (15) Young, F. Y.; Wong, D. F.; Yang, H.: "Slicing floor plans with Pre-placed Modules". *Proceedings IEEE International Conference on Computer-Aided Design*, 1998, 252-258.
- (16) Young, F. Y.; Wong D. F.; Yang H.: "Floor plan Area Minimization using Lagrangian Relaxation". *IEEE Transaction of Computer-Aided Design of Integrated Circuits and Systems*, 20, 2001, 687-692.
- (17) Ungar, P.: "On diagrams representing maps". *Journal of London Mathematical Society*, 28, 1953, 336-42.
- (18) Levin, PH.: "Use of graphs to decide optimum layout of buildings". *Architectural Journal*, 1966, 809-17.

- (19) Nugent, C. E.; Vollmann, T. E.; Ruml, J.: "An experimental Comparison on Techniques for the assignment of Facilities to Locations". *Operation Research*, 16, 1, 1968.
- (20) Mitchell, W.; Steadman, J. P.; Liggett, R. S.: "Synthesis and Optimisation of Small Rectangular Floor Plans". *Environment and Planning B*, 3, 1976.
- (21) Magiera, J.: "Exterior partitions of a rectangle into rectangles and their graphs". Graph Theory Lagow 1981 proceedings, Berlin, Springer Verlag, Berlin, 1981, 125-130.
- (22) Roth, J.; Hashimshony, R.; Wachman, A.: "Turning a graph into rectangular floor plan". *Building and Environment*, 17, 1982, 163-173.
- (23) Akin, O.; Dave, B.; Pithavadian, S.: "Heuristic generation of layouts (HeGel): based on paradigm for problem structuring". *Artificial Intelligence in Engineering Design*. Elsevier, 1987, 413-441.
- (24) Tommelein, I. D.; Vaughann Jhonson, Jr. M.; Hayes-Roth, B.; Levitt, R. E.: "SIGHTPLAN: A blackboard expert system for construction site layout". *Expert Systems in Computer-Aided Design*; North Holland, 1987, 153-167.
- (25) Roth, J.; Hashimshony, R.: "Algorithms in graph theory and their use for solve problems in Architectural design". *Computer Aided Design*, 20, 1988, 373-381.
- (26) Brown, D. C.; Chandrasekaran, B.: "Design problem solving". London: Pitman; 1989.
- (27) Eastman, C.: "Modelling of Buildings: Evolution and Concepts", *Automation in Construction*, 1, 1992, 99-109.
- (28) Schwarz, D. M.; Berry, E.; Shabib, E.: "Representating solving the automatic building design problem". *Computer Aided Design*, 26, 9, 1994, 689-698.
- (29) Schwarz, D. M.; Berry, E.; Shabib, E.: The use of automatic building design system. *Computer Aided Design*, 26, 10, 1994, 747-761.
- (30) Bierdemann, J. D.; Grierson D. E.: "A generic Model for Building Design", *Engineering with Computers*, 11, 3, 1995, 173-184.
- (31) Femming, U.; Chien, S.-F.: "Schematic layout design in the SEED System", *ASCE Journal of architectural Engineering*, 1, 4, 1995, 162-169.
- (32) Hover, W.; Graf, W.: "A bibliographical survey of constraint-based approaches to CAD, graphics, layout, visualization, and related topics", *Knowledge-Based Systems*, 9, 7, 1996, 449-464.
- (33) Gero, J. S.; KazaKov, V. A.: "Envolving design genes in space layout planning problems", *Artificial Intelligence in Engineering*, 12, 3, 1998, 163-76.
- (34) Medjdoub, B.; Yannou, B.: "Topological enumeration heuristics in constraint-based space layout planning". Gero JS, Sudweeks F, Ed. *AI in Design*, 98, 1998, 271-90.
- (35) Jo, J. H.; Gero J. S.: "Space layout planning using an evolutionary approach", *Artificial Intelligence in Engineering* 12, 3, 1998, 149-62.
- (36) Ge, J.-X.; Chou S.-C.; Gao, X.-S.: "Geometric constraint satisfaction using optimization methods", *Computer Aided Design*, 31, 1999, 867-879.
- (37) Grabska, E.; Pelaez, W.: "Floor layout design with the use of graph rewriting system PROGRES", Schellenbach-Held H, Denk H, Ed.; *Proceedings of the Ninth International Workshop of the European Group for Intelligent Computing in Engineering (EG-ICE)*, VDI Verlag, Dusseldorf, 2002, 149-57.
- (38) Bohlen, B.; Jager, D.; Scheleicher, A.; Westfechtel, B.: "UPGRADE: A framework for building Graph-Based Interactive Tools", Corradini A, Ehrig H, Kreowski H-J, Rozenberg G, Ed.; LNCS, 2505, Springer, Barcelona, Spain, 2002, 270-85.
- (39) Szuba, J.; Borkowski, A.: "Graph Transformations in architectural design", *Computer Assisted Mechanics and Engineering Science*, 10, 1, 2003, 93-109.
- (40) Rivard, H.; Fenves, S. J.: "A representation for conceptual design of building" *J. of Computing in Civil Engineering (ASCE)* 14, 3, 2000, 151-159.
- (41) Meniru, K.; Bedard, C.; Rivard, H.: "Early building design using computers", P. Christianson Ed.; *Proceedings of the Conference on Distributing Knowledge in Building (CIB w78 2002)*, Aarhus School of Architecture, Denmark, 2002.
- (42) Kraft, B.; Schneider, G.: "Semantic room objects for conceptual design support: a Knowledge-based approach", Martens B, Brown A, Ed.; *Proceedings of the 11th International Conference on Computer aided Architectural Design Futures (CAAD Futures 05)*, Springer, Heidelberg, 2005, 207-216.
- (43) Cappello, F.; Mancuso, A.: "A genetic algorithm for combined topology and shape optimizations", *Computer Aided Design*, 3, 2003, 1-9.
- (44) Kraft, B.; Nagl, M.: "Visual knowledge specification for conceptual design: Definition and tool support", *Advanced Engineering Informatics*, 21, 2007, 67-83.

* * *