arquiteturarevista - Vol. 6, nº 2:136-161 (julho/dezembro 2010) ISSN 1808-5741
doi: 10.4013/arq.2010.62.05

The problem of space layout in architecture: A survey and reflections

O problema da distribuição espacial na arquitetura: pesquisa e reflexões

Danny Lobos
danny.lobos@uni-weimar.de
Bauhaus Uni-Weimar. Belvederer Allee 1, Raum 201, 99425 Weimar, Germany

Dirk Donath
dirk.donath@uni-weimar.de
Bauhaus Uni-Weimar. Belvederer Allee 1, Raum 201, 99425 Weimar, Germany

Abstract

This work is about the problem of Space Layout Planning (SLP) applied to Architectural Design. The creation of floor-plan layouts is key to architecture, and it is a problem that, because of its complexity, does not have a precise general method for its resolution. An overview and a wide description of the "Architectural Design" process are presented to understand this phenomenon. Secondly we present a brief history and development of the SLP field, and then a description of the state-of-the-art (techniques, approaches and strategies) to implement layout solutions. Concepts such as optimization, generative systems, artificial intelligence, genetic algorithms, physically-based modelling are reviewed and discussed. Commercial CAAD and BIM software is also reviewed. Conclusions and a discussion about the results and absence of this type of software for architectural practice are presented.

Key words: space layout planning, computer-aided design, automated architectural floor layout, design methodologies.

Resumo

O artigo trata do problema da distribuição espacial em planta (SLP, por sua sigla em Inglês) aplicada ao projeto arquitetônico. A criação de desenhos de plantas é uma questão fundamental e, devido à sua complexidade, nenhum método preciso para ser resolvido. Apresentamos uma visão geral e uma descrição detalhada do processo de projeto arquitetônico para compreender esse fenômeno. Em segundo lugar, apresentamos uma breve história e o desenvolvimento do SLP. Em seguida, trataremos sobre o estado da arte (técnicas, abordagens e estratégias) para implementar soluções de design. Os conceitos de otimização, sistemas generativos, inteligência artificial, algoritmos genéticos e modelagem com base na física também são revistos e discutidos. Também são analisados os softwares comerciais CAAD e BIM. Ao final, apresentamos as conclusões e uma discussão sobre os resultados e o impacto deste tipo de software na prática da arquitetura.

Palavras-chave: planejamento de distribuição espacial, design criado por computador, projetos arquitetônicos automatizados, metodologias de design.

Automated floor layout planning: A utopia?

One of the most important tasks of architects is the creation of a floor-plan layout for a building. The automation of the creation of floor-plan layouts deals with the use of computers to generate designs, and it is called Space Layout Planning. Research in this field started around 50 years ago. Many solutions have been presented and discussed: prototypes, tests, depth computer programming, and optimization formulas embedded in the architectural field, as well as hundreds of publications with big promises for the future. However, in the end, we, the architects, have nothing on our desks to support our daily task in the office, not a single tool that generates a useful architectural floor-plan layout.

No matter how fast, optimal, automated, or parameterized the solutions have been, architects do not use them, and it seems that they will never meet the long list of architectural criteria. This piece of research

Hypotheses space layout planning

We put forward several hypotheses as well as provide and discuss supporting evidence on the following titles:

(i) Architects have to fit a wide space program (clients' needs) into a shape (building's envelope).
(ii) Architects must consider many variables when designing floor-plan layouts.
(iii) At least thirteen variables influence the creation of a floor-plan layout.
(iv) The result of this process is a drawing that contains an arrangement of rectangles.
(v) The Space Layout Planning (SLP) field has taken some of these variables and has tried to generate floor layouts automatically. Four trends are identified: Generative, Constraint-Based, Shape Grammar and Expert Systems.
(vi) The results from SLP have not been accepted in architecture.
(vii) Useful SLP can be achieved by means of a deep understanding of a specific case in architecture.

Floor Plans and Architectural Layout

The design of a floor-plan is a stage within the architectural design process that takes place between the "schematic design" phase and the "design development" phase. Figure 1 shows a single house, its architectural floor-plan layout and the abstraction of relationships between spaces.

Figure 1. Casa Poli in Coliumo, Pezo von Ellrichshausen Architects (Chile, 2005).
Source: Personal communication with authors (2008).

A floor-plan consists of a drawing that shows the location of the different areas and rooms required by the client, as well as the sizes, names, walls and floor limits of each of them.

There are hundreds of design methods (Loemker, 2006); however, the building as an "object that contains rooms and spaces" is, and it has always been, the trend in architecture.

Projects built by some of the most famous architects in history such as Le Corbusier (*Chapel of Notre Dame du Haut in Ronchamp),* Mies Van der Rohe (*Farnsworth House* in Illinois), and Frank Lloyd Wright (*Fallingwater)* show this trend. The phenomenon is also observed in the current results of big international competitions, where Norman Foster, Zaha Hadid, Rem Koolhaas (OMA) and Frank Gehry lead (see Figure 2). In all these works, we can see the predominance of an exterior shape/form that contains the rooms.

Figure 2. Norman Foster (1), Frank Gehry (2), Zaha Hadid (3), Rem Koolhaas OMA (4), Renzo Piano (5). Source: Architects' websites, visited on June 21, 2010.

The complexity of this process can be observed in this example by OMA Architects (see Figure 3) of the China Central Television Headquarters building in Beijing. Here the space program is "forced" to be inside the curious shape of the building (a geometric dialogue between two Z). One can see here, in colour, how the different areas are located, following complex functional requirements and constraints; however, they always fit the exterior form. Another example is the new Shard London Bridge by Renzo Piano (current construction, 2010), where all the areas and rooms of the building are fitted *inside an iceberg-like sculpture emerging from the River Thames* (see Figure 4).

How to put rooms into a shape is something seldom described in architecture books or during university studies. A famous statement in architecture by Louis H. Sullivan says "form follows function", that is to say, the shape of a building or object should be based upon its intended function or purpose. Ching (1979) and Neufert (1935) have made big and famous efforts to explain some "techniques," rather than steps or rules, to distribute the rooms in a shape. Therefore, we conclude that this process is still a black box: the result appears suddenly, like magic.

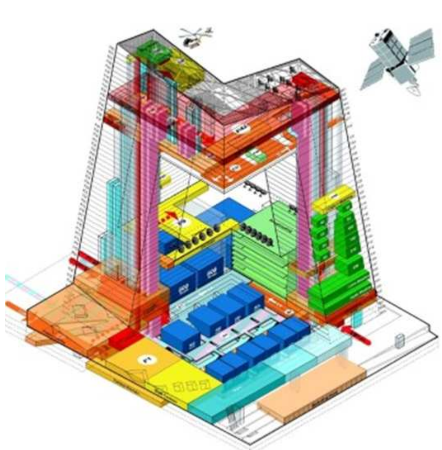


Figure 3. China Central Television Headquarters building in Beijing. Source: Architect website, visited on June 21, 2010.

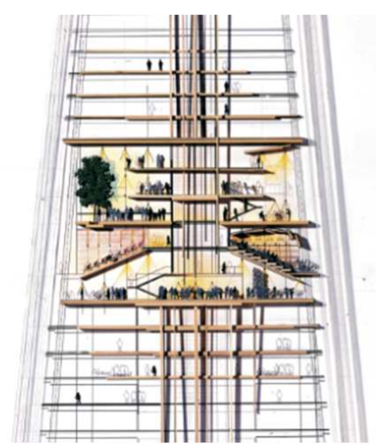


Figure 4. The London Bridge Tower by Renzo Piano. Source: www.plataformarquitectura.cl, visited on June 25, 2010.

Foundations of Floor-Plan Layout in Architectural Design

The creation of an architectural floor layout corresponds to the assignment of relationships to a function and then a geometry (and size) to this function. When a client decides for a new house or a new building, he normally presents the site to the architect and declares what he needs there. This declaration of needs is called Space Program, and it is a transcription or a translation of the needs (human activities) of the client into an architectural programmatic language, that is to say, words and numbers that can be transformed by the architect into rooms, sizes and some relationships between these rooms.

Criteria and Variables

When the architect starts to refine and organize this Space Program, he has to fulfil some criteria. Depending on each case, one criterion could be more important than others. Based on our own experience as practitioners and several bibliographic resources (Neufert, Ching, Hsu, Elezkurtaj), next we describe the most common criteria grouped into rational criteria (possible to be measured by rational procedures) and general design criteria (impossible to be measured by rational procedures).

Rational criteria for architectural floor-plan layout

(i) Solar: To place the rooms in the optimum place and orientation in relation to the sun. Objective: To assure natural illumination to the largest amount of rooms of long stay. Unit: vector.
(ii) Views: To place the spaces in the optimum place and orientation in relation to the views. Objective: To assure the best views of the landscape or the surroundings to the long-stay rooms. Unit: vector.
(iii) Accessibility: Referred to the distance between the main street (or street of access) and the entrance of the building. Objective: To minimize the distance to access the building. Unit: m (meters).
(iv) Related Functions: Some functions of the Space Program are more related to each other than others. Objective: To establish which rooms are highly, fairly or scarcely related, and which room should be away from the rest. Unit: factor.
(v) Minimum distance: The objective is to establish the minimum distance between rooms to optimize the circulation spaces. Unit: m (meters).
(vi) Efficiency (Circulation/Usable Ratio - Circulation): The result of comparing the circulation surface with the usable surface. Objective: To keep most of the surface for use and little surface for circulation. Unit: percentage.
(vii) Efficiency (Volume/Usable): The result of comparing the volume of each space with the usable volume of it. Objective: To keep most of the volume for use and less unusable volume. Other aspects such as the sun and the ventilation could have an influence on this variable. Unit: Percentage.
(viii) Size: The size of the room (width x length x height) refers to some size pattern or standards based on the building type (hotels, schools, housing).

General design criteria for architectural floor-plan layout

(ix) Geometric Composition: The rooms must be inside a larger geometric form (square, circle, arc, rectangles, etc.), sized and grouped following aesthetic intentions.
(x) The Divine Proportion/Golden Ratio: The wall distribution follows the sides of a rectangle, and the lengths of both sides of the rectangle follow a fixed numerical relationship (1.6180339887). It is possible to measure it.
(xi) 3d Shape to Fill: Ching (1979) defines the possible configurations of space distributions as: Linear, Central, Yard, U Shape, L Shape, Organic Shape, Religious Shapes.
(xii) Sustainable Criteria: The space distribution should be optimum regarding sustainable criteria such as minimal surface in perimeter walls, energy consumption, solar gain in surfaces, material quantification, room light load, etc.
(xiii) Others: Economical, structural and spiritual meanings (FengShui, Mapuches); construction techniques.

Conclusions

At least thirteen requirements should be considered during the architectural floor layout creation, so now our problem is to put together the rooms following at least some of them. The creation of floor-plan layouts in architecture is a problem that, because of its complexity, does not have a precise method for its resolution. In the floor-plan design, it is not possible to fulfil all the criteria, but just some of them.

This stage can be represented through some abstract type of drawings or sketches called: Matrix/Schemas (see Figure 5).

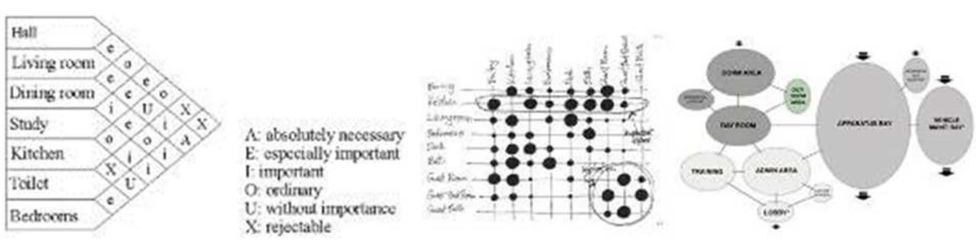


Figure 5. Some abstract drawings or sketches called: Matrix / Schemas.

Finally, the conversion of these abstract drawings into a real design (a final floor layout) is neither clear nor ruled, and it remains in the "black box" paradigm.

Automated space layout planning

In general, a layout deals with objects and their relationships, no matter whether these objects are rectangles, nodes, train stations, motherboard pieces, facilities, or rooms. For designers, the problem is the following: given a set of objects and some relationships between them, making a layout that contains such

In architectural design, there is a lack of tools for space planning. Therefore, in order to find approaches that support the solution of the problem, architects have had to look to the other fields mentioned; however, when we do this, we lose some variables and gain some new ones.

Brief history and development of space layout planning

The oldest approach used to face the problem is called “Floor-plan design for industry”. The study of methodologies was carried out intensively in the ’50s (Immer and Buffa). Then SLP (Systematic Layout Planning) was included. After that, no other important pieces of research were carried out in this field due to the wide acceptance of the SLP method. During the last years, the pieces of research have gone deeply into layout generation and solution optimization.

Naturally, many other authors have tried to use this and other approaches; next a small list/history of authors that have researched into the field of Space Planning (see Table 1):

Table 1. List of authors in the field of Space Planning.

Author	Title	Publication / Reference	Year
Buffa, E.S	*Sequence analysis for functional layouts.*	J. Ind. Eng.	1955
Buffa, E.S., Armour, G.L., Vollman, T.E.	*Allocating Facilities with Craft*	Harvard Business Review	1964
Johnson, T.E., Weinzapfel, G.E., Perkins, J. I., et al.	*IMAGE: An Interactive Graphics-Based Computer System for Multi-Constrained Spatial Synthesis.*	M.I.T.	1970
Mitchell, W.J.:	*A Computer-Aided Approach to Complex Building Layout Problems*	EDRA2 Conference	1970
Miller, W.R.	*Computer-Aided Space Planning*	Workshop on Design Automation	1970
Eastman, C.E.	*A System for Computer Assisted Space Planning*	Workshop on Design Automation	1971
Al Banna, S. and Spillers, W.R.:	*An Interactive Computer Graphics Space Allocation*	Workshop on Design Automation	1972
Mitchell, W.J. and Dillon, R.	*A Polynomial Assembly Procedure for Architectural Floor Planning*	Third Environmental Design Research Association Conference	1972
Stiny G, Gips J	*"Shape Grammars and the Generative Specification of Painting and Sculpture"*	C V Freiman (ed) Information Processing 71	1972
Krawczyk, R.J.	*SPACE PLAN: a User Oriented Package for the Evaluation and the Generation of Spatial Inter-Relationship*	10th Design Automation Workshop	1973
Gero, J. S.	*A System for Computer-Aided Design in Architecture*	Principles of Computer-Aided Design	1973
Teicholz, E.	*The Computer in the Space Planning Process*	12thDesign Automation Conference	1975
Gero, J. S	*Computer Aids to Design And Architecture*	N. Negroponte (ed.)	1975
Weinzapfel, G., Negroponte, N.	*Architecture-by-yourself. An Experiment with Computer Graphics for House Design*	Siggraph	1976
Fortin, G.	*BUBBLE: Relationship Diagram using Iterative Vector Approximation*	15th Design Automation Conference	1978
Ruch, J.	*Interactive Space Layout: A Graphic Theoretical Approach*	Conference on Design Automation	1978
Shaviv, E.	*Automatic Generation of Optimal or Quasioptimal Building Layout*	CAAD Futures	1985

One can see here that the beginning of the studies in this field was more than fifty years ago. In addition, Mitchell and Dillon were the first researchers that brought the problem to the architectural design field. Since then, four main approaches have been developed. Therefore, from a historical and architectural point of view, the four main trends and their precursors are:

(i) Expert Systems (Flemming, Coyne)
(ii) Shape Grammar (Stiny, Mitchell, Duarte)
(iii) Generative (Gero, Fraser)
(iv) Constraint-Based (Gross, Medjdoub and Yannou, Hsu).

State-of-the-art review

There are three main reviewed categories to group the literature: prototypes and research, literature related to academic activities, and commercial CAAD software (packages from companies available in the market, which we divide into BIM and non-BIM packages).

Prototypes and Research

We review the last ten years of research. We present a descriptive survey of each approach. They will be described thus: Author (year), techniques utilized, description of procedures, screenshots, and, finally, an evaluation of each one (the pros and cons). They are sorted by year.

(i) Arvin and House (1999). Physically-Based Modelling Techniques: Forces and elastic band concepts applied to a functional Space Program. Use of "Dynamic Physic Simulation". Adjacency is modelled as a spring (elastic) connection. It transforms the designer's "intention" of "moving a space" into forces. Pros: A comparison between objective design and constrained design is shown. It allows users to interact as in the "real world". Detailed description of the implementation. Cons: Complex definition of the relationship between spaces/mass and vice-versa.

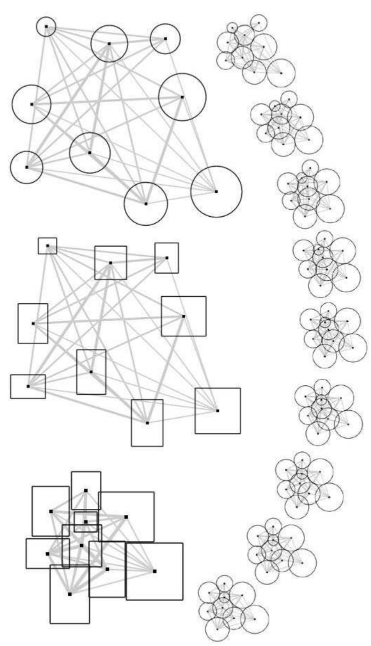


Figure 6. Arvin and House (1999). Physically-Based Modeling Techniques.

Figure 7. Hsu (2000). Constraint-Based.

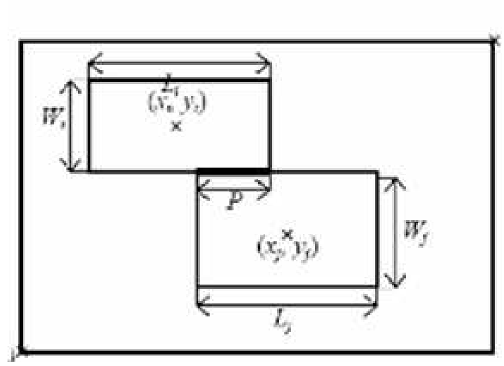


Figure 8. Li et al. (2000) Constraint-Based Generative System.

(ii) Hsu (2000). Constraint-Based: It creates a database with the relationships between spaces and the surrounding (site, sun, light, wind). Features: AutoCAD + LISP. Generation of several options. Use of colours and 3D diagrams. It generates a 3D wall model. *Pros:* It considers the architectural input, such as relationships, site and natural conditions, for the database. The application follows these restrictions (constraints). It works in a well-known environment (AutoCAD). *Cons:* No description of implementation (just the language: AutoLISP + AutoCAD). No use of non-rectangular shapes. No accuracy in spatial orientation. Difficulty to check spatial relationships.

(iii) Elezkurtaj and Franck (1999). Generative Design: System that supports architectural floor-plan design interactively. Approach: New AI (Artificial Intelligence), Evolutionary Strategy (ES) and Genetic Algorithm (GA). It deals with the definition of the function in architecture, allowing some proportions for the room and some topological relationships between them. The solution is created automatically by GA. *Pros:* Wide description of GA function and optimal and of how to search for the optimum. Description of mathematical operations. The interface emulates architectural design environments, it is of simple use, and it allows an easy and intuitive interaction with the user as well as a fast answer in real time. The result is really acceptable in terms of architectural design. Description and critics of AI, New AI, Shape grammars and GA are a contribution to the discussion and theory of this sub-field. *Cons:* No description of implementation. Images of samples are not clear. No real case study (only from students). There is no fixed room list. It does not consider specific cases (standard cases). Missing links to web in papers. The use of boundaries to contain spaces is not clear.

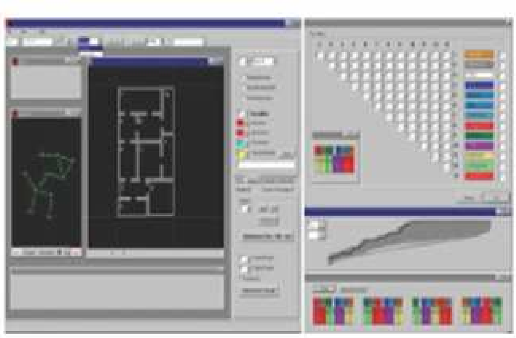


Figure 9. Elezkurtaj and Franck. Generative Design.

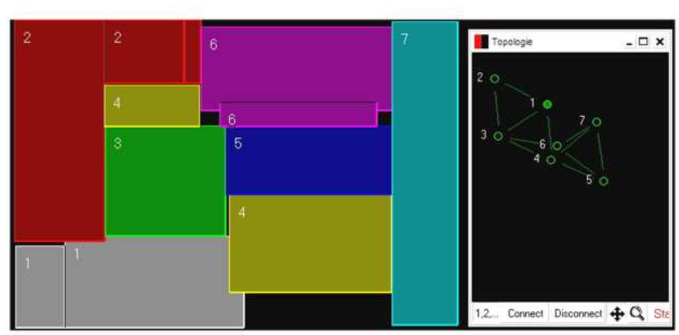


Figure 10. Elezkurtaj and Franck.

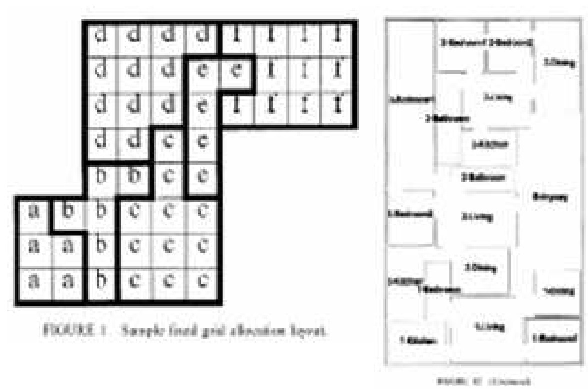


Figure 11. Michalek et al. (2002). Gradient Based and Evolutionary Algorithms.

(iv) Li *et al.* (2000) Constraint-Based Generative System: Non-Linear Programming that provides multiple solutions inside a rectangular floor plate. It uses LINGO, a Non Linear Solver mixed with SLP (Successive Linear Programming) and GRG (Generalized Reduced Gradient Alg.). Visual implementation in Microstation. 10 solutions in 4 minutes. Optimal and sub-optimal solutions for giving designers "other inspirations". *Pros:* The Constrained-Based Approach is similar to the architectural practice. It is a "real" solution that uses "real" data. Detailed explanation of implementation and techniques (SLP and GRG). *Cons:* Irregular boundaries are not included. The complex mix of SLP and GRG implementation solvers is hard to understand for non-expert users.

(v) Michalek *et al.* (2002) Gradient Based and Evolutionary Algorithms: Optimization Model and a method for integrating mathematical optimization and subjective decision during conceptual design. Use of gradient-based algorithms and evolutionary algorithms for discrete decisions and global search. It defines the available space as a set of grid squares and uses an algorithm to allocate each square to a room activity. *Pros:* It takes into consideration the aesthetic and other subjective aspects of design. Mathematical optimization allows the user to interact in the design process without worrying about the background complex operations through an "object-oriented representation" of it. Designers can change objects and constraints during the process. *Cons:* Very complex description of how each variable responds to another. The language of the methodology and the process is not related to an architectural environment.

(vi) Hsu and Krawczyk (2003). CAD In Space Planning Methods: It presents the state-of-the-art of CAD in Space Planning as well as a description of techniques such as: Neighbor Searching techniques, Switching techniques, Random techniques, Zoning Clustering, Virtual Grid Searching Methods, Bubble Diagram simulation, Interactive Space Layout and Physically-Based Space modification. Finally, it shows the novel SPDA tool: Space Planning Design Assistant. *Pros:* Qualification of the spatial character in residential buildings, firms, banks, theaters, etc. Division between fragmental and solid forms. *Cons:* It does not consider the volume or shape that architects use. Complex description of how to use the application. It does not describe the user's input.

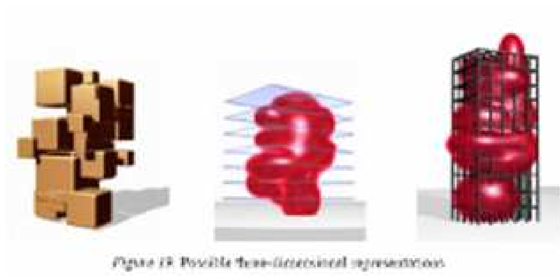


Figure 12. Hsu and Krawczyk (2003, 2004). Computer-Aided Design In Space Planning Methods.

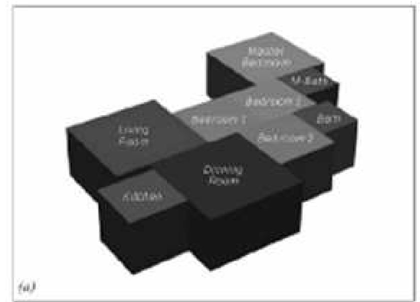


Figure 13. Hsu and Krawczyk (2003, 2004). Computer-Aided Design In Space Planning Methods.

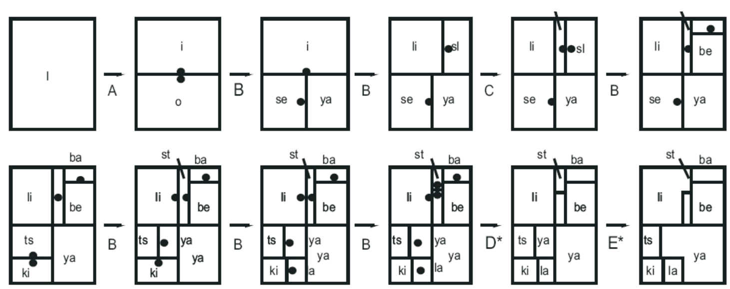


Figure 14. Duarte (2003) Discursive Grammar.

(vii) Duarte (2003). Discursive Grammar: Process for mass customizing housing based on computer-aided design and production systems. Development of an interactive system for generating solutions on the Web based on a "discursive grammar" (programming grammar and designing grammar). It provides the rules for generating designs in a particular style. It describes the designing grammar using Alvaro Siza´s houses at Malagueira as a case study. *Pros:* The use of computer-driven shape grammars came close to passing an architectural Turing test (Elezkurtaj and Frank, 2000). *Cons:* Plans are meaningful only if they are well-formed, which means that the elements are defined in a clear-cut way and manipulated according to syntactical rules (Elezkurtaj and Frank, 2002). Architectural design cannot be reduced to the production of graphics and the imitation of styles (Elezkurtaj and Frank, 2000).

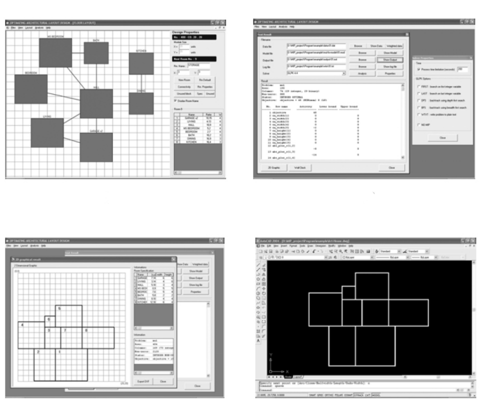


Figure 15. Keatruangkamala and Sinapiromsaram (2005). Mixed Integer

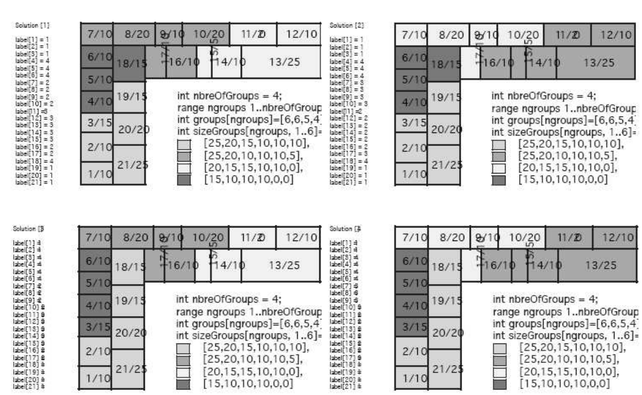


Figure 16. Loemker (2006). Operations Research: Allocation Techniques + Scheduling Algorithms.

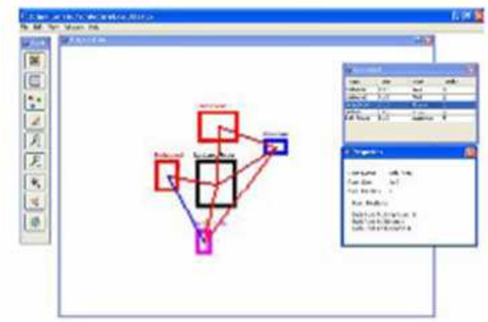


Figure 17. Nilkaew (2006). Genetic Algorithm.

(viii) Keatruangkamala and Sinapiromsaram (2005). Mixed Programming: Several houses with 4, 5, 6, 7 and 8 rooms are designed. Use of solvers: GLPK, CPLEX, DICOPT. Definition of variables and parameters: functional and constraint dimension. Constraint and objective functions: minimize the distance between rooms and maximize room spaces. Use of GLPK (GNU Linear programming Kit) from Moscow Aviation Institute (Russia). *Pros:* Clear interface, fast and with the promise of an optimal layout solution with multi-objectives. It continues the ideas from Li *et al.* (2000). *Cons:* Complex geometry description instead of goals and multi-objectives. No test with architects. Complex understanding of formulae for non-experts.

(ix) Loemker (2006). Operations Research: Allocation Techniques + Scheduling Algorithms: Architectural Layout Planning is described in the form of mathematical rules. It demonstrates that "design" is a combinatorial problem in principle, i.e. a constraint-based search for an overall optimal solution of a design problem. It is applied to the design of new buildings, as well as to the revitalization of existing buildings. Planning task approach from Operation Research is taken to prepare optimal decisions through the use of mathematical methods, where the understanding of design is related to the search for solutions that fulfil specific criteria. Use of scheduling algorithms. It allows non-destructive optimization of existing floor-plans. *Pros:* It allows distributing a space program into an existing building. The use of non-rectangular boundary is allowed. Ten results are obtained in a few minutes. The "non-destructive" approach contributes to the creation of a concept of "Sustainable Renovation of Buildings." *Cons:* The adaptation of the Operation Research Approach to solve the re-allocation is complex for non-expert users. The interface is in progress and user interaction (input and constraints) is not described in this paper.

(x) Nilkaew (2006). Genetic Algorithm: It studies the House Design problem. Analysis Process: Room Space to Room Relation. Qualitative: Topological (Architectural Space and Relations), and Quantitative: alternate schematic plan options. Made by GA process and computational optimization algorithms. *Pros:* Easy understanding of concepts: mix of qualitative and quantitative variables. It brings a real, logical and architectural way of thinking. It captures information from architectural knowledge: function schemes, sizes, and relationships. It generates several solutions that fit this knowledge. *Cons:* No use of boundary. No variety of shapes (only rectangles). No more information about GA implementation and objective function. No description of time consumption.

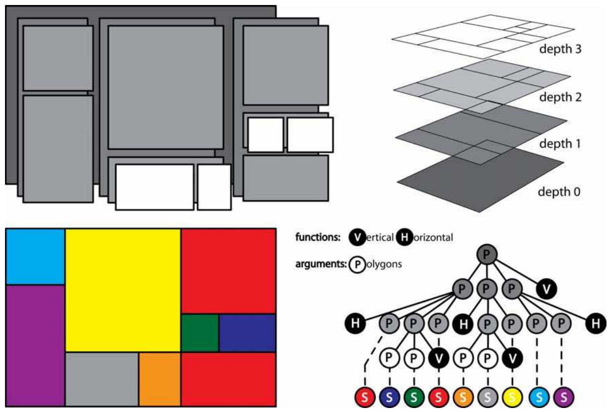


Figure 18. Doulgerakis (2007). Genetic Programming + Unfolding Embryology.

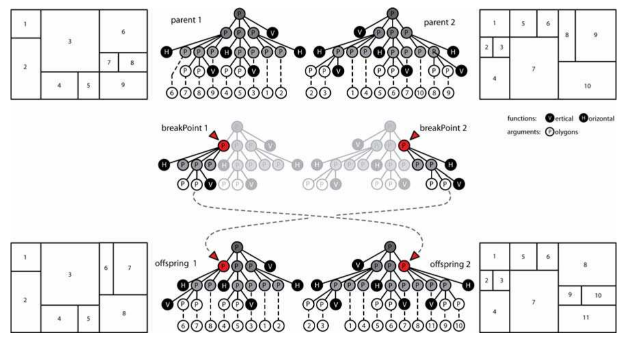


Figure 19. Doulgerakis (2007).

(xi) Doulgerakis (2007). Genetic Programming + Unfolding Embryology: Implementation of computational methods for the generation and optimization of floor plans, considering the spatial configuration and the assignment of activities. A co-operative system was created, which is composed of a Genetic Programming (GP) algorithm and an agent-based unfolding embryology procedure that assigns activities to the spaces generated by the GP algorithm. The Ranking Sum Fitness evaluation method is proposed and applied for the achievement of multi-objective optimization. *Pros:* It provides a complete literature review and classification of Space Layout Planning. A co-operative system (Genetic Programming algorithm + agent-based unfolding embryology procedure) assigns activities to the spaces generated by the GP algorithm in a natural way for designers. *Cons:* Arbitrary mix of the social and cultural generative forces of the layout with evolutionary systems. The Ranking Sum Fitness evaluation method and concepts are not close to architectural practices.

(xii) Donath and Gonzalez (2006, 2008). Constraint-Based Design: A participatory housing planning process in Latin America supported by a constraint-based design strategy implemented in two different and separated subdomains of the problem. First of all, we have a Maxon´s Xpresso prototype for the building bulk design problem that considers variables (constraints) such as floor area of the unit, lot coverage, constructability, and building height, and that works in a semi-automated way, giving the user the chance to interact with volumes and to write the input. It starts from a predefined set of objects, and it finally shows a new 3d layout configuration. The second prototype, developed in OPL Studio, is capable, after the input of constraints, of finding and displaying a 2D sequence of solutions represented by a combination of rectangles. *Pros:* It allows the user to interact with the application and see the results in 3D mode in real-time. It considers the constraints of maximum lot coverage and constructability. It describes the process of generation of a layout for the specific kinds of houses in depth. *Cons:* Both prototypes solve different parts of the problem in different environments. The results of the first part are not proved to be applicable either to real praxis (architects), or to the rest of the users (planning offices, dwellings, government) because of the complex definition behind it and because of its interface. The method described in the second part does not consider all the applicable building codes to a site.

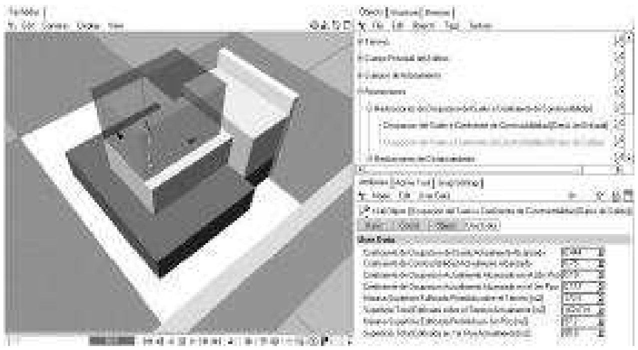


Figure 20. Donath and Gonzalez (2006). Constraint-Based Design in Participatory Housing Planning.

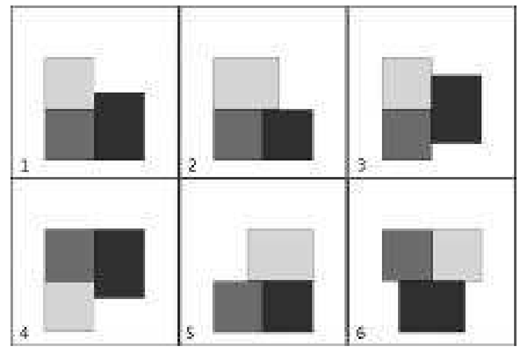


Figure 21. Donath and Gonzalez (2008).

(xiii) Medjoub and Yannou (2001). Topological Level and Heuristic Algorithms: Space planning application that uses topological solutions and graphs. It applies Heuristic Algorithms for Space Ordering and it allows constraints. It solves topological aspects without presuming dimensions. It is possible to define relationships, orientation, and minimum sizes. *Pros:* Concepts like Space Planning, Topological Solutions, Heuristics, Space Ordering and Constraint-based are well described. It argues the validity of Constraint Programming. It argues that, in preliminary design, topology is more important than geometry (as a criticism of shape grammars, expert systems and others). Clear explanation of variables and constraints. Complete description of searching mechanisms and results. It does not presume dimensions at the beginning. The space program is handled in an "architectural" way. *Cons:* It mentions constraint and restriction equations, but does not give names or descriptions. It is tested with architects and other users, but not tabulated. Time-consuming search in the first steps. Complex resolutions for an architectural context but necessary for the expected solution.

(xiv) Del Río-Cidoncha *et al.* (2007b). Facility Layout Design: The goal of this piece of research was to develop and implement a model for Facility Layout Design in Architecture. It takes approaches and definitions from both the fields of Architecture and Engineering. It mixes Expert System and Artificial Intelligence (AI). Constraints and needs are translated into several algorithms in three stages: Locating (slicing trees), Routing (expert systems based on rules for productions) and Orientation (Computer Assisted Numerical Method) following the FengShui method. It presents the CEF (Cut-Expert FengShui) Methodology that creates floor layouts. Solution based on a mix of three techniques: Slicing Trees, Expert Systems and FengShui. *Pros:* First approach that is close to Architecture and Engineering, using elements from both. Excellent state-of-the-art presentation and comparison between approaches, with a detailed description of techniques. The solution generated "looks like" an architectural floor-plan layout. *Cons:* Definitions of the theoretical background are taken from researchers and not from designers. The problem of routing is not an issue in architecture. The process is divided into three parts, and it is not clear how the user jumps from one to the other. The use of a random existing sample makes it difficult to know the parameters utilized in the original design.

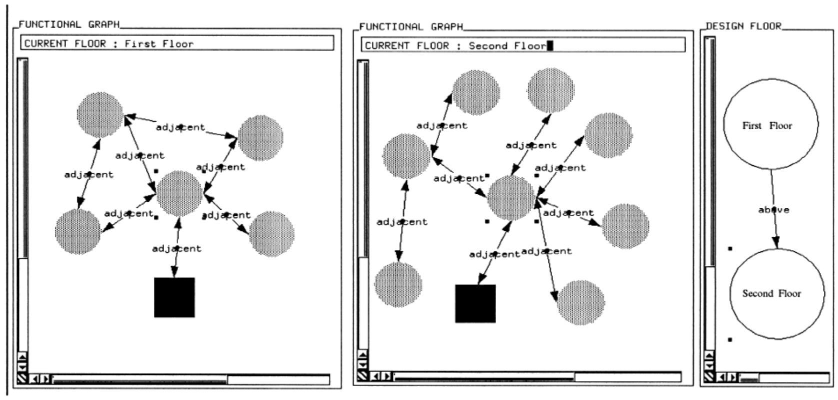


Figure 22. Medjoub and Yannou (2001). Topological Level and Heuristic Algorithms.

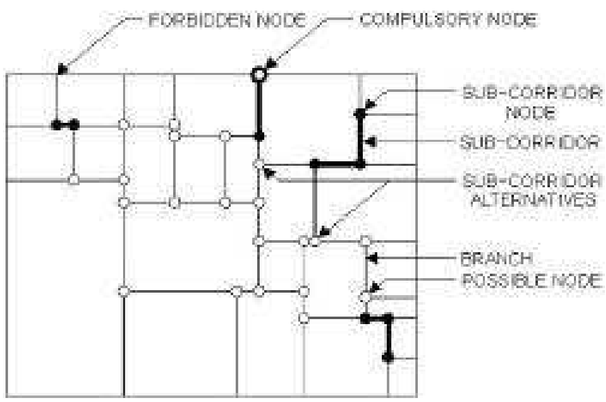


Figure 23. Del Río-Cidoncha et al. (2007b).

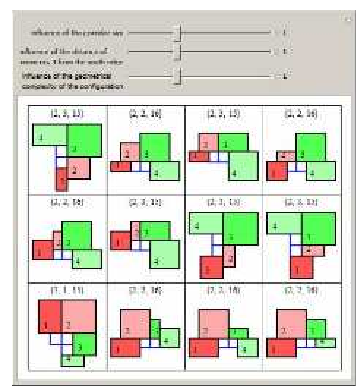


Figure 24. Multi-objective Optimization Tool for Room Configuration.

(xv) Multi-objective Optimization Tool for Room Configuration: Developed by Wolfram Research of Champaign. This case shows a *Multi-objective Optimization of a Room Configuration* of a very simple building layout with two apartments, each having only two rooms and a corridor. The layouts are ranked according to three criteria: the size of the corridor (the blue grid), the distance of room 3 from the southernmost edge of the layout and the geometrical complexity (the total number of unique coordinates of all the corners of each room). These three values are normalized, weighted, and combined into the Aggregate Objective Function (AOF). Users change the weights to alter the importance of a given parameter. The values of these parameters are shown for each layout. There are 247 different room configurations, but only the 12 best (according to a given AOF) are shown. *Pros:* It allows to watch and understand the optimization process step by step. It allows the user to change some parameters by using slide bars. *Cons:* It does not use boundary. The orientation of rooms is hard to understand. Input of graphical information is not allowed.

Comparison of different approaches

Third, we make a classification table of these approaches. The criteria are Science (where they come from), Approach (trend within a scientific field), Implementation (techniques used in the resolution of the problem) and Boundary use (to distribute the space program, yes or no).

Table 2. Comparison table for different approaches.

Comparison Table for Different Space Layout Planning Approaches	Arvin and House	Hsu	Elezkurtaj and Franck	Li, Frazer and Thang	Michalek, Choudhary and Papalambros	Hsu and Krawczyk	Duarte	Keatruangkamala and Sinapiromsaram	Loemker	Nilkaew	Doulgerakis	Donath and Gonzalez	Medjoub and Yannou	Del Rio	Mathematica		Average
Approach Number	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15		
SCIENCE																	
Engineering			x	x	x	x		x	x	x		x	x	x	x		11
Mathematics							x								x		3
Physics	x																1
Medicine											x						1
Architecture		x	x	x		x	x			x		x	x	x	x		10
APPROACH																	
Expert Systems						x		x						x			4
Generative (GA and EA)			x		x					x	x		x	x			6
Shape Grammars							x										1
Constraint Based		x		x	x	x		x	x			x	x		x		9
Graph Approach													x				1
Mixed Approach								x			x						2
Others (Agents, Physic)	x										x						2
IMPLEMENTATION																	
Linear Programming								x	x					x	x		4
Non Linear Programming				x					x								2
Gradient Based Algorithms				x	x									x			3
Genetic Algorithms			x							x	x						3
Integer Programming								x	x								2
Differential Equations	x																1
Mathematical Equations						x			x			x					3
Drawing techniques						x			x								2
BOUNDARY USE																	
Yes			x	x			x		x		x	x	x	x			8
Not	x	x			x	x		x		x						x	7

Conclusions for research and prototypes

As a conclusion, we can say that:

(i) Big efforts have been made for solving this problem using approaches from several disciplines; however, half of the approaches come from Engineering and the other half come from Architecture.
(ii) Most of the approaches use the Constraint-Based Approach and Genetic Algorithms.
(iii) There are many methods for implementation.
(iv) Half of the approaches use boundary and the other half do not.
(v) The use of Graphs is seldom seen.
(vi) There is no strong evidence of using these technologies in practice. Only Medjoub and Jannou, as well as Duarte, declare to use it for real architectural design, but no strong proof is shown.
(vii) Some of the approaches are very close to architectural practice and others are far away. That means

All of the approaches produce a floor-plan design; this consists of a border shape, which represents a flat or a building storey, that contains other interior shapes, which, in turn, represent functions and sizes for human activities (rooms).

Commercial (Non BIM) CAAD Software Review

Next, we present some digital tools from software companies that are/were available in the market. The review was made based on practical experience and knowledge about these software programs as well as on the support documentation provided by the companies.

(i) Alberti (AcadGraph, 1998)
A German company developed a complete package solution for the automatic generation of architectural room layouts. It applies concepts of Neuronal Networks to handle the relationships and constraints between spaces. It needs "real" input such as the structure of building stories, the name of the rooms, the orientation of the rooms (north, south, etc.), and the relationships between the rooms (strong, medium, and weak). Finally, the algorithm produces about one hundred solutions in some seconds and chooses the one that fits the criteria better. Pros: The definition of the rooms and the variables to consider are similar to the real architectural practice in a very clear and simple interface. It works with a real boundary for the building (rectangular and non-rectangular). Cons: The solutions generated by the software were never accepted in practice because they give a non-artistic floor-plan design and generate many new "empty" spaces to fill the complete boundary (see grey areas in the images below). The final result is the creation of "nonsensical" floor plans (see Figure 25). This could happen because of the architects´ inability to clearly specify the space program at the beginning or maybe because in real practice this happens but it is quickly fixed by hand drawing.

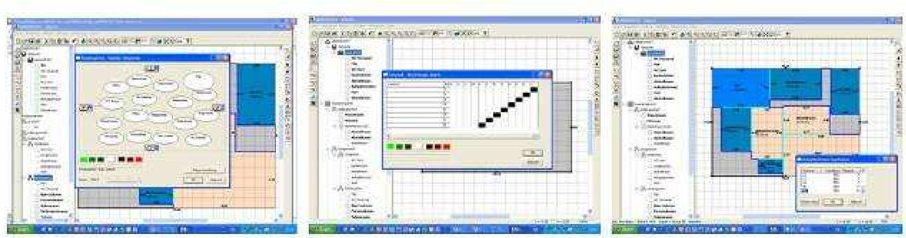
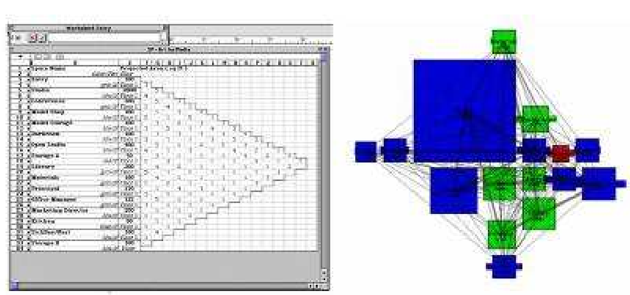


Figure 25. Alberti (AcadGraph, 1998).
Source: Self-elaboration Lobos © 2008.



(ii) Vectorworks10 (Nemetscheck, 2004)

The tool was included only in one version of Vectorworks under the name of "Space Planning Tools". It consisted of three steps: internal or external definition of the Space Program (rooms, names, sizes and relationship between them), import schedule to the software and, finally, automatic creation of space arrangements by the software. The result is a planar and non-overlapping distribution of the rooms on the screen (see Figure 26). Pros: The relationships between spaces are defined in a classical way, so every architect can understand the interface. It shows a small evaluation in a 2D band for space surface comparison. Cons: The creation of the schedule can be made inside or outside the software, but then the import step has some complications related to the extensions and versions of the Microsoft Excel software. The architect must re-locate all the rooms manually following any criteria.

(iii) Affinity 5.0-5.6 (Trelligence, 2006-2009)

It was created to support the architectural business process of a building (plan, design, construction). It consists of different steps: capture of space program (within the software or outside by using spreadsheets), project settings (building site, use, budget and costs to meet, settings of rooms and areas such as sizes, numbers, relationships), schematic design (manual distribution from browser to the screen of each room), 3D visualization (it enables a GDL technology for real-time 3D views of the rooms as 3D blocks), and evaluation and report (the generated solution is compared to the original requirements of the project, and an easy comparison can be made, a red color appearing when the original requirements are not met). Pros: It is possible to set some variables of the spaces and areas in a digital format, and it allows to re-use this information or bring it from other applications. The report is very accurate and refers to data of real needs of the client and other players. It has a wide library of room types and room implements, as well as several templates for building types (house, offices, health, etc.). Cons: It does not create a new solution. Architects must manually move and place the spaces along each story. The setting phase, previous to design, is very long and difficult (if we think about non-expert users) and is completely far from the "3D way" of thinking of architects. Plugins for Revit and Archicad only evaluate layouts after the decisions are made and they do not generate solutions (see Figure 27). Version 5.6 offers a Dynamic Relationship Modelling tool that translates adjacency requirements into relationships, but without generating a design.

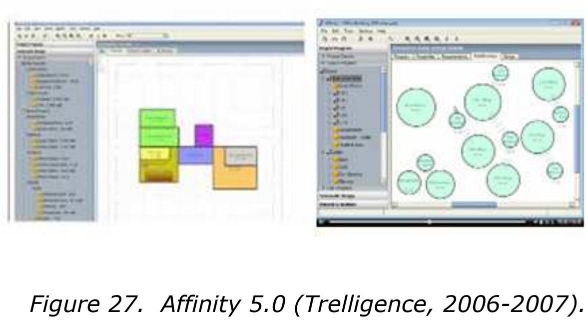


Figure 27. Affinity 5.0 (Trelligence, 2006-2007).

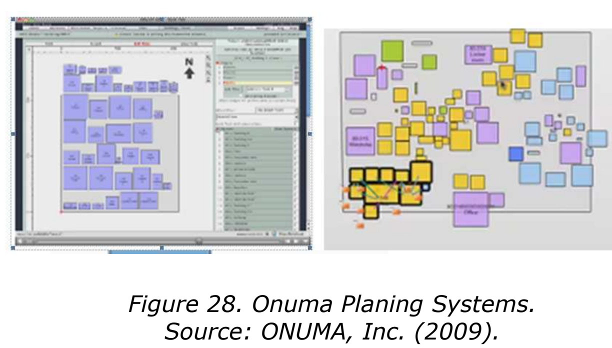


Figure 28. Onuma Planing Systems.
Source: ONUMA, Inc. (2009).

(iv) Onuma Planing Systems (ONUMA, Inc., 2009)
It offers a Web-based planning workflow and allows different teams to be connected and to design collaboratively. It allows to group and visualize the spaces in areas and then share the result in different interfaces through IFC and XML files (SketchUp, Revit, Archicad, IFC Viewer). Onuma starts from Excel spreadsheets with the room; it needs name, number (about story), level, and area or sizes (not both). As a result, it shows all rooms grouped by level, in a single order: one room next to another. Then one must manually move the rooms to create a layout. Pros: Easy transcription of Space Program requirements to the software; it allows teamwork, good quality and web performance of 3D views; finally, it allows IFC format exchange. Cons: It does not create a floor layout.

Conclusions for Commercial (non BIM) CAAD Software

There is no commercial application that supports the creation or automatic generation of an architectural floor-plan layout. Only Alberti creates solutions, but they are not acceptable. Others like Affinity describe the schematic layout in detail but do not generate one. Vectorworks is also a good approach because it generates a solution, but this is not a layout; here the spaces are together and then we need to move them manually.

One can notice here that through the years, the idea of an automatic creation of a final layout has been rejected by architects, and a new trend has emerged that just intends to improve the performance of the layout design process by simply supporting the design, without layout creation. This can be made by handling complex information, giving easy access for non-experts to use this technology, and making plug-in for other platforms (CAD and BIM).

Commercial BIM Software Review

Due to the importance of BIM in the architectural practice, we have decided to include it in our review of the state-of-the-art. Names include version, company and year.

(v) Archicad 9.0 (Graphisoft, 2007, now Nemetschek)
The "Zone" Tool allows to describe the space contained within walls (name, number, story, area, perimeter, etc.) in depth. It allows colour and 3D visualization of the space (without walls and others). It must be done after the design of walls.

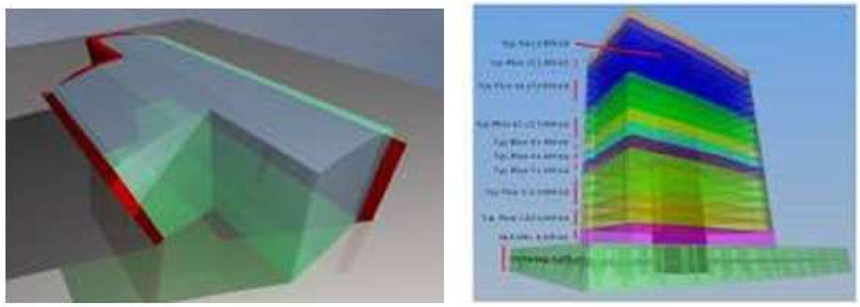


Figure 29. The "Zone" Tool in Archicad 9.0.
Source: Graphisoft (2007).

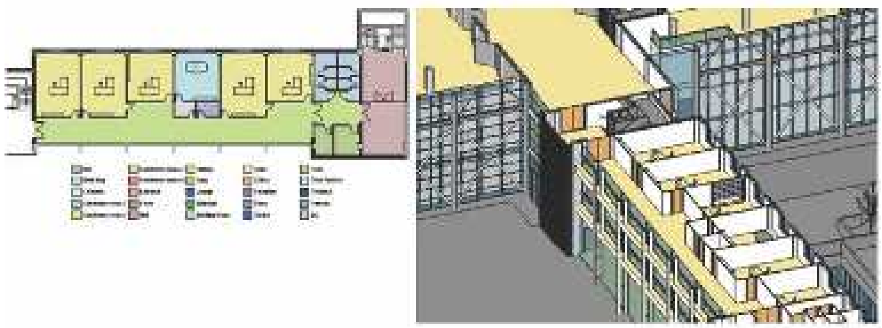


Figure 30. Room/ Area Tool in Autodesk Revit 2008.
Source: Autodesk © (2008).

(vi) Autodesk Revit 2008 (Autodesk, 2007)

The "Room/Area" tool allows automatic and real-time scheduling of the spaces within walls and creation of a colour legend for each story. This allows to visualize the spaces-rooms-areas in each floor plan and section in a customizable way. It is easy to create reports for each storey. It must be done after the design of walls. It is possible to place mass objects and then turn them into rooms with different possibilities for size.

(vii) Bentley Architecture-Microstation v8 (Bentley Systems, 2008)

The tool called "Place Space" allows to put spaces in the workspace. Room and component schedules, quantity and cost calculation, specifications. It can be done after or before the design of walls. Spaces must be dragged manually to create a composition layout.

(viii)Allplan BIM 2008 (Nemetschek 2008)

It is possible to create a room element by using the Room tool to define its boundaries, or to create rooms automatically within a specified area by using the Auto-Room tool (it will detect all the spatial enclosures and create individual rooms within them). It allows the creation of floor space calculations and color-filled plans based on various criteria for space planning and the Facility Management. It must be done after the design of walls.

Figure 31. Place Space Tool in Bentley Architecture (Microstation v8). Source: Bentley © (2008).

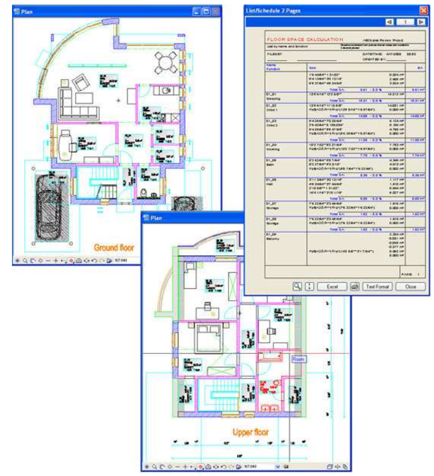


Figure 32. Room Tool in Allplan BIM 2008.

(ix) Architectural Desktop 2007 (Autodesk 2007)

Now the software program is called Autodesk Architecture. Here it is possible to create rooms and areas or to use a template with room standards (i.e.: ansi-boma). The rooms can be moved into an area and maintain some relationship between them (room-area, room-level). The rooms can be placed into walls, and then the user automatically gets a room-boundary or the room object can be placed, and the user generates walls from its boundaries. It generates wide and precise schedules and lists with all the rooms of the project and calculates areas with deduction rules.

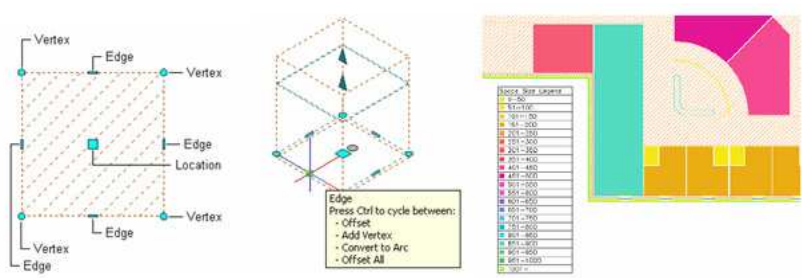


Figure 33. Space and Room tool in Autodesk Architecture (formerly ADT). Source: Autodesk © (2007).

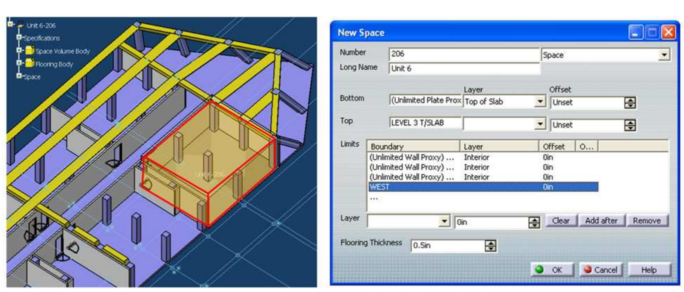


Figure 34. The Space Command in Digital Project Designer V1,R4. Source: http://www.gtwiki.org/mwiki/r4doc/creatingrooms.htm, visited on June 3, 2010.

(x) Digital Project Designer V1-R4 (Gehry Technologies, 2008)

Digital Project Designer (Version 1, Release 4) is a comprehensive 3D modeller with an extensive set of tools for creating and managing building information throughout the building lifecycle. It was developed by Gehry Technologies (USA).

The "Space" command allows to create room objects by clicking walls; it is possible to use an existing slab to define the bottom of such a room. It also allows to create bounded spaces from either physical boundaries such as walls or virtual boundaries such as grid lines, curved surfaces, and the like. Once the spaces are made, the software calculates the Space Volume.

Other software programs

(xi) Microsoft VISIO (MV)

The existing template provides a tool for Facility Managers and Space Planners to show high-level views of building-space occupancy. The template uses the concepts of stacking and blocking to represent visually how much space is both used and unused in a particular building (Figure 35).

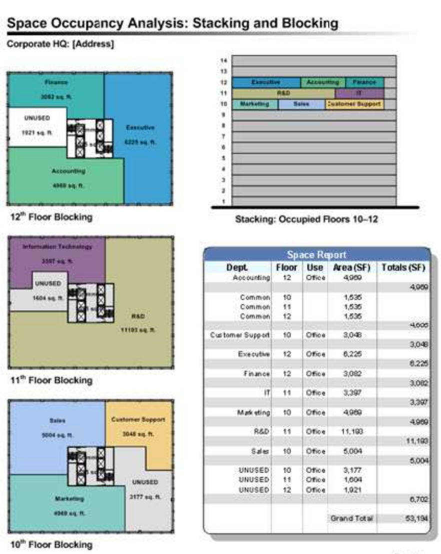


Figure 35. Microsoft VISIO Template for Facility Management.
Source http://office.microsoft.com/en-us/templates, visited August 16, 2010.

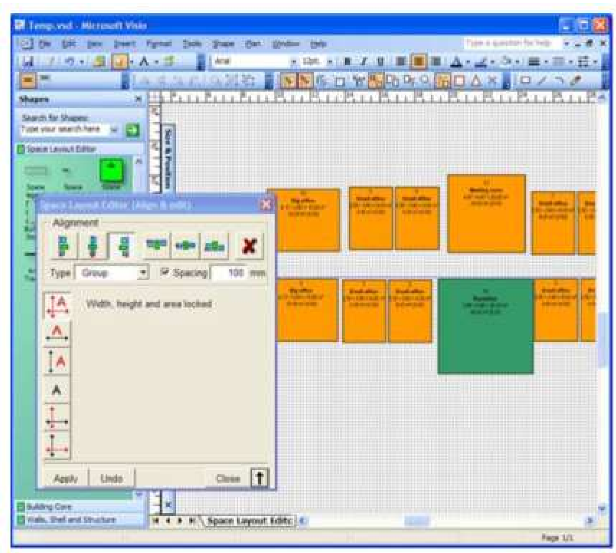


Figure 36. Space Layout Editor for Microsoft Visio.
Source: www.digitalalchemypro.com, visited August 18, 2010.

(xii) Space Layout Editor for Microsoft Visio

A plugin for MV. Automatic space object generation, based on client´s space program in Excel. It generates an IFC BIM spaces model. See Figure 36.

Conclusions for Commercial BIM Software

After this deep inquiry and analysis of the possibilities of the commercial BIM software, we tabulate our observations (see Table 3) and conclude the following:

- None of them generates a floor layout automatically.
- The process of supporting Space Planning tasks begins after the creation of the walls or spaces by the designer.
- These software programs are useful to visualize and communicate decisions.
- It is necessary to provide tools for the kinds of tasks that work within a BIM solution. If a solution is made outside of this environment, i.e. Open Source, then we must figure out how to import it to BIM. Moreover, the IFC exchange format has not proved to be standard or completely feasible.
- BIM software proves to have a big advantage: they can save many information/data like type, position, material, relationship, hierarchy, sizes, families, and story/height, and they allow a powerful visualization.

Table 3. Comparison Table for commercial BIM software.

COMPARISON TABLE FOR COMMERCIAL BIM SOFTWARE

	BOUNDARY	Spaces	Areas	Topology	Sizes parameters	Options	Automatic Generation	Space creation without walls	Space creation from list of rooms
Archicad 9.0	√	√	-	x	x	x	x	x	x
Autodesk Revit 2008	√	√	x	x	√	√	x	√	x
Bentley Architecture (Microstation v8)	-	√	-	x	-	-	x	√	x
Allplan BIM 2008	-	√	-	x	-	-	x	x	x
Architectural Desktop 2007	√	√	√	x	√	x	x	√	x
Digital Project Designer	√	√	-	x	-	-	x	x	x

√ (allowed), - (unknown), x (not allowed)

Summary for automated space layout planning

The most important conclusion from the state-of-the-art review is that after 50 years of research in this field architects do not have a right tool to support the “automatic” creation of floor layouts.

Even though most researchers in the field of Space Layout Planning attempt to support architectural design, none of them makes reference or quotes paradigmatic architects such as Le Corbusier, Mies van der Rohe, Frank Lloyd Wright, and Walter Gropius or any design methods utilized in a real architecture office. They do not mention resources for architectural design practice like the most recognized magazines (Summa+, Croquis, Domus, DETAIL, Architectural Review) or well-known theoretical books for architects (Neufert, Ching, Benevolo). Therefore, it seems that they come up with their own idea of what a floor-plan layout design in Architecture is, and then they develop a solution for this idea. This could lead to a misunderstanding of the architectural problem. Of course, they cannot be expected to read everything about Architecture, but at least they should have a notion of what architects call “good architecture”.

Conclusions about CAAD approaches: Commercial, research and prototypes

Architectural and Engineering Approaches

The problem of floor-plan design is a matter that concerns both architects and engineers, even though there are many differences and similarities (Del Río-Cidoncha *et al.,* 2007a). The similarities are:

- Input data is a contour.
- There are some rooms or activities to be located (with certain dimensional requirements).
- There are some relationships between rooms or activities.
- The results are the positions or distributions of these activities or rooms.

(i) *Supporting or Generating?*

Space Layout Planning approaches can be divided into two main categories:

- Computer Supported Space Layout: related to commercial issues.
- Computer Generated Space Layout: related to academic/research issues.

Most of the commercial software programs only *support* the layout creation by manipulating rooms through manual methods (move, rotate, edit boundary), but only a few prototype applications *generate* layouts.

1. Mathematics or Arts?

Now there are some points to enrich the future discussion about the problem. From the point of view of Engineering:

- There are successful approaches in the field of engineering; they meet the criteria of this field.
- Floor planning can be seen as a mathematical problem: Rectangular Polynomial Arrangement (Gero, 1977).
- Space Layout is an Engineering problem that was solved and framed a long time ago (Sequence Analysis and Systematic Layout Planning by Buffa (1955) and now some sub-steps are being improved every year (Del Río-Cidoncha *et al.,* 2007a).
- Optimal or Sub-Optimal results have been achieved (Michalek *et al.*, 2002; Li *et al.*, 2000).
- The approaches follow complex constraints and relationships.
- The approaches solve the problem from the engineer's point of view.

From the point of view of Architecture:

- The approaches have failed.
- Has any architect used these approaches in a real design project?

- Have these approaches been implemented in the architectural practice? (Except Vectorworks10 and Alberti).
- What happens with the interfaces?
- Is there a poor conception and a misunderstanding of the problem?
- Are the researchers/authors architects?
- Do architects use optimal or sub-optimal results?

Recommendations for future research

Future research in this area of Space Layout Planning should be implemented in the practice of architecture. For this aim, we propose:

- To describe in detail all the variables that must be taken into consideration when designing a floor-plan layout, from the architectural point of view.
- To allow user participation; this is crucial to define high constraints. In the approaches reviewed in this article, these constraints must be added through annoying mechanisms.
- Researchers must attempt to decrease the dependence on high-performance algorithms, since this is far from the architectural background and practice.
- By analysing the architectural problem itself in depth, it is possible to find the clues for the solution (like topological structures, space program specifications, building regulations for specific cases, etc.) instead of using approaches from other fields that will lead us to non-sense results.
- Before making an application (either prototype or commercial), it is necessary to know the complete design process and its variables exactly and to know what a good solution is; this knowledge comes only from a deep understanding of architecture.
- It has been proven that architects do not use optimization, ratio for room sizes, level of connection (low-high) between spaces and many other variables that engineers include in their solutions for Space Planning to mathematize the design problem.
- There is no universal tool for space planning; each case (residential buildings, hospitals, schools, etc.) has some special requirements, even though they do have some common points in the early stages.

Conclusions

The capacity of computers to solve the problem of floor-plan layout design in architecture has been demonstrated, but at the same time we have demonstrated that the results are not utilized in the architectural practice.

Facing the lack of methodologies in the architectural field, the trend in the last fifty years has been to take elements from engineering. All the pieces of research and prototypes have a good evaluation from the engineering point of view, but not from the architectural one. The clearest and absolute proof of that is the absence of those results on our desktop, as available tools for the architect’s daily work. On the other hand, (Onuma, Trelligence) in the last years, offering a fast, though not automated, way to get the layout design and its evaluation in the early stages.

BIM (Building Information Modeling) systems do not generate designs; rather they support the visualization and manipulation of vast and complex information about building design from the early stages until the building life cycle is completed.

Engineers' solutions, in the case of Facility Layout Planning and related fields, should look at and learn from well-kwon architects like Mies Van der Rohe, Le Corbusier, and Frank Lloyd Wright. In the same way, architects should learn how engineers resolve design problems.

It is crucial for the future that architecture students continue with the classical Design Courses, but, at the same time, they have to be trained in programming language tools and related courses. In this way, they will resolve the problem of their field without jeopardizing the quality and usability of the solution.

References

ARVIN, S.; HOUSE, D. 1999. Physically Based Modeling Technique in Space Layout Planning. *In:* INTERNATIONAL CONFERENCE ON COMPUTER AIDED ARCHITECTURAL DESIGN FUTURES, 8, Atlanta, 1999. *Proceedings...* Atlanta, 1999, p. 245-262.

BUFFA, E.S. 1955. Sequence analysis for functional layouts. *J. Ind. Eng*., **6**:12-25.

CHING, F. 1979. *Architecture: Form, Space and Order.* New York, Van Nostrand Reinhold.

DEL RÍO-CIDONCHA, M.; IGLESIAS, J.; MARTÍNEZ, J. 2007a. A comparison of floorplan design strategies in architecture and engineering. *Automation in Construction,* **16**:559-568.

DEL RÍO-CIDONCHA, M.; IGLESIAS, J.; MARTÍNEZ-PALACIOS, J. 2007b. A multidisciplinary model for floorplan design. *International Journal of Production Research,* **45**(15):3457-3476.

DONATH, D.; GONZALEZ, L.F. 2006. A constraint based Building Bulk Design Support. *In:* SOCIEDAD IBEROAMERICANA DE GRÁFICA DIGITAL, 10, Santiago de Chile, 2006. Proceedings... Santiago, 2006, 278-282.

DONATH, D.; GONZALEZ, L.F. 2008. Constraint-Based Design in Participatory Housing Planning. *International Journal of Architectural Computing,* **6**(1):97-117.

DOULGERAKIS, A. 2007. *Genetic and Embryology in Layout Planning.* London, UK. Masters Dissertation. University of London, 84 p.

DUARTE, J. 2003. A Discursive Grammar for Customizing Mass Housing. *In:* EUROPEAN COMPUTER AIDED ARCHITECTURAL DESIGN AND EDUCATION, 21, Graz, 2003. Proceedings... Graz, 2003, p. 665-674.

ELEZKURTAJ, T.; FRANCK, G. 1999. Genetic Algorithms in Support of Creative Architectural Design. *In:* EUROPEAN COMPUTER AIDED ARCHITECTURAL DESIGN AND EDUCATION, 17, Liverpool, 1999. Proceedings... Liverpool, 1999, p. 645-651.

ELEZKURTAJ, T.; FRANCK, G. 2000. Algorithmic Support of Creative Architectural Design. *In:* EUROPEAN COMPUTER AIDED ARCHITECTURAL DESIGN AND EDUCATION, 18, *Proceedings…* Weimar, 2000, p. 309-312.

ELEZKURTAJ, T.; FRANCK, G. 2002. Algorithmic Support of Creative Architectural Design. *In:* UMBAU, 19, Vienna, 2002. *Proceedings…* Vienna, 2002, p. 1-16.

GERO, J.S. 1977. Note on "Synthesis and optimization of small rectangular floor plans" of Mitchell, Steadman, and Liggett. *Environment and Planning B*, **4**:81-88.

HSU, Y.C.; KRAWCZYK, R. 2003. New generation of computer aided design in space planning methods: A survey and a proposal. *In:* COMPUTER AIDED ARCHITECTURAL DESIGN AND RESEACRH IN ASIA, Bangkok, 2003. Proceedings... Bangkok, 2003, p. 101-116.

HSU, Y.C.; KRAWCZYK, R. 2004. Form Development with Spatial Character. *In:* EUROPEAN COMPUTER AIDED ARCHITECTURAL DESIGN AND EDUCATION, 18, Copenhagen, 2004. *Proceedings...* Copenhagen, 2004, p. 278-285.

HSU, Y.C. 2000. Constraint Based Space Planning: A Case Study. *ACADIA Quarterly*, **19**(3):2-3.

KEATRUANGKAMALA, K.; SINAPIROMSARAM, K. 2005. Optimizing Architectural Layout Via Mixed Integer Programming. *In:* CAAD FUTURES, 11, Vienna, 2005. *Proceedings...* Viena, 2005, p. 175-184.

LI, S.P.; FRAZER, J.; TANG, M.X. 2000. A Constraint Based Generative System for Floor Layouts*. In:* CONFERENCE ON COMPUTER AIDED ARCHITECTURAL DESIGN RESEARCH IN ASIA, 5, Singapore, 2000. *Proceedings...* Singapore, 2005, p. 441-450.

LOEMKER, T. 2006. Designing with Machines: solving architectural layout planning problems by the use of a constraint programming language and scheduling algorithms. *In:* INTERNATIONAL CONFERENCE OF THE ARAB SOCIETY FOR COMPUTER AIDED ARCHITECTURAL DESIGN, Sharjah, 2006. *Proceedings...* Sharjah, 2006, p. 88-106.

MEDJDOUB, B.; YANNOU, B. 2001. Dynamic space ordering at a topological level in space planning. *Artificial Intelligence in Engineering,* **15**:47-60.

MICHALEK, J.; CHOUDHARY, R.; PAPALAMBROS, P. 2002. Architectural Layout Design Optimization. *Engineering Optimization,* **34**:461-484.

NEUFERT, E. 1935. *Architects' Data.* London, Blackwell Science, 638 p.

NILKAEW, P. 2006. Assistant Tool for Architectural Layout Design by Genetic Algorithm. *In:* CONFERENCE ON COMPUTER AIDED ARCHITECTURAL DESIGN RESEARCH IN ASIA, 11, Kumamoto, 2006. *Proceedings...* Kumamoto, 2006, p. 641-643.

Submetido em: 31/03/2010

Aceito em: 03/10/2010