An Intelligent Decision Support System for Layout Design

Abdul Rahim Ahmad (arahim@uwaterloo.ca)
Systems Design Engineering, University of Waterloo, Waterloo, Ontario, Canada

Otman Basir[1] (obasir@uwaterloo.ca)
Electrical & Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada

Khaled Hassanein (hassank@mcmaster.ca)
Michael G. DeGroote School of Business, McMaster University, Hamilton, Ontario, Canada

Abstract. Layout Design is a complex, vague and ill-structured problem that requires intelligent and sophisticated decision support. Inadequate information availability, cognitive biases and competing preferences hamper appropriation of a superior outcome. It is therefore desirable to develop an intelligent system for layout design that deals with such challenging issues and provides efficient means of generating, analyzing and manipulating superior alternative layouts. We present a framework for building an interactive Intelligent Expert System for Decision Support in Layout Design based on Soft Computing tools. The proposed framework and the prototype system contribute to the field of decision support in the layout design by enabling explicit representation of experts' knowledge and formal modeling of fuzzy user preferences. Such a system is expected to facilitate the economic and ergonomic efficiency of layout designers, superiority of outcome, and future research in related areas.

Topic Areas: 2, 12.

Introduction

The layout design process is geared towards seeking some superior outcomes in the spatial arrangement of modules in a given space, satisfying given preferences and constraints, based on the fitness evaluation objectives. It is an important problem with applications in a wide array of domains. However, the layout design problem is so vast in scope, involves such myriad of tangible and intangible factors with such high degree of dynamism and subjectivity that validity of any proclaimed optimal solution could easily be challenged [2]. Consequently, the layout design process requires sophisticated knowledge-based modeling techniques and decision support [3], [12].

The foremost task in automating the layout design process is to generate superior Layout Alternatives for further consideration and manipulation by the designers [3]. The layout designers face a high cognitive overhead in acquiring, remembering, understanding, and applying the vast body of subjective and uncertain information/preferences available. Consequently, the usefulness of available algorithms and guidelines is largely limited by the inability of layout designers to appreciate, comprehend and quantify the system related characteristics without an elaborate methodology for applying those [3]. Such barriers mean that the layout design is not easily amenable to automation. Nevertheless, the continuous development of such sophisticated and ubiquitous applications as Facilities, VLSI, and the Web Page layout design has created an intense interest in formulating and automating the layout design algorithms and guidelines.

[1] Corresponding Author

An interactive Decision Support System designed specifically to mitigate the cognitive overload by providing fast and easy ways of generating, analyzing, and manipulating superior layout alternatives seems to be a logical choice in this direction. In this paper, an architectural framework and prototype for an Intelligent Layout design Decision Support System is presented. This framework is a part of an ongoing research project aimed at formulating a generic framework for addressing major issues involved in using subjective and vague layout design preferences.

Limitations of Existing Approaches

A range of formulations for the layout design problem has been proposed in the literature. The most popular of such formulations include the Quadratic Assignment Problem, the Two-Dimensional Bin-packing Problem and the Quadratic Set-covering Problem. Various heuristic and analytical techniques have been published for finding solution to this NP-Hard problem [8]. Recent heuristics that have shown good results are Simulated Annealing, Genetic Algorithms, Naive Evolution, as well as various hybrids. Other solution approaches include Tree Search algorithm, Binary Mixed Integer-Programming, Network Decomposition etc. In addition, various analytical techniques are also reported, which deal with continuous design space with minimal computational requirements [10].

The research in the layout design area has resulted in several Automated Layout Design systems. Here we will discuss limitations of existing systems in three most commonly encountered applications, namely, Facility Layout, Circuit Layout (e.g. VLSI) and the Visual Interface Layout (e.g. the Web Page Layout). Despite all the assertions of flexibility and relative superiority, the existing software have considerable limitations from various standpoints [2]. Indeed, a majority of such automated layout design aids are simply CAD-based documentation and drawing tools [12].

The development and application of automated Facility Layout Design systems started decades ago. A representative, nonetheless, non-exhaustive list of such software includes CORELAP [15], ALDEP [20], SPACECRAFT [17], FLING [4], MOCRAFT [18], FactoryOpt [14], VIP-PlanOpt [7], etc. However, deterministic techniques are used for incorporating preferences in the design process, which are not suitable for such a subjective problem domain [2], [12]. Moreover, many existing systems do not treat a module as a whole and generate layouts with irregular and unrealistic shapes of modules entailing manual adjustment. Furthermore, the inability of existing systems to consider a large number of modules for placement decisions is of important concern. Our experience shows that the VIP-PlanOpt is likely to be the fastest and most robust system for handling more than 40 modules.

In addition, the research in automating the Visual Interface Layout Design acquired eminence with the advent of such influential applications as GUI and the Web. Some examples of existing systems are ADDI [6], UIDE [19]. However, such interface builders typically furnish a set of widgets to facilitate the design process and little support is afforded for incorporating the domain-specific preferences. The process of mapping the domain objects and their properties into corresponding visual properties in the layout design is left to the user. Some work has been done towards creating some visual knowledge-base to ameliorate this problem [6]. However, incorporation of subjective, vague as well as ambiguous preferences and properties is still elusive.

The aforementioned approaches have pros and cons; however, in general, the existing systems are inflexible, slow, and incapable of tackling large-scale problems. The inflexibility emanates from various sources including the rigidity of the fitness function(s) as the superiority of a layout is determined by a multitude of competing formal/informal criteria. Furthermore, the use of largely predefined steps in generating layout alternatives has resulted in various unwarranted simplifying assumptions that hamper a diversified search of the solution space. In addition, the data handling methods are suitable only when reliable deterministic or crisp interaction data is available and assignable to specific processes. Nonetheless, such data may only exist for some designated unknown/unrealistic modeling conditions. Authors' personal experience and extensive literature examination show that existing techniques usually follow an 'optimization' approach, instead of adopting a 'decision-making' paradigm, in layout design. Thus, effective ways of analysis and revision through the incorporation of subjective preferences as well as designers' intuition, creativity and expertise are virtually nonexistent.

Various possible means for considering subjective preferences in fast generation and manipulation of superior layout alternatives include flexible revisions of multi-criteria fitness functions. Such flexibility is expected to furnish efficiency and efficacy to an automated layout design system by tapping the creativity, intuition, and other sub-cognitive abilities of the designer. It would also sanction simulation of the process through analysis, comparison, and revision of various alternatives. It should be noted that the computer-based algorithms cannot replace human judgment and experience as those algorithms do not always capture the qualitative aspects of the domain. Nevertheless, it is often effortlessly easy for experts to look at an alternative and endorse whether it is acceptable or not. It points towards the possibility that some incomplete models could assist layout designers in addressing the dynamics of the problem in an efficient manner. As such, the emphasis of our research is not on the pursuit of some perfect methods but rather on development of a tool that could supplement the knowledge, experience, and design intuition of the layout designers.

Proposed Framework

Irrespective of substance and efficacy of automated tools, decision makers always need to make fuzzy and difficult choices without comprehensive knowledge of the principal determinants of the domain as well as the capacity to envisage the results precisely. Recent developments in the field of Soft Computing have rendered powerful tools for tackling with such complex problems. The principal components of soft computing are fuzzy logic, neural networks, genetic algorithms, probabilistic reasoning, chaos theory etc. Such tools deal effectively with complex structuring and ill-defined dynamics of the layout design problem. The underlying paradigm is concerned with modes of computing where precision is traded for tractability, robustness, and ease of implementation. Furthermore, such tools provide means for mimicking the powerful capability of humans in expressing knowledge linguistically. Such approaches are gaining acceptance for modeling cognition, intelligent systems, and artificial intelligence because the procedures involved are most analogous to human reasoning [2]. The knowledge of strengths and weaknesses of these technologies results in synergistic hybrid systems that produce more powerful and robust systems.

Consequently, we have chosen to tap on these tools for tackling the layout design problem.

Our approach to layout design problem is to build an Intelligent Layout design Decision Support System (IL-DSS) using an Expert System paradigm as depicted in Fig-1. The inherent characteristics of the expert system paradigm including the separation of the domain knowledge from the control knowledge, ability to reason under uncertainty, explanation facility, etc. make it our choice of technology. The goal is to enhance the productivity of layout designers by addressing major issues in the layout design process. An efficient algorithm for generating superior layout alternatives is an important step in this regard. Consequently, we have used a hybrid fuzzy-genetic Intelligent Layout Generator for this purpose. The 'intelligence' aspect comes from the use of fuzzy rules/preferences for obtaining penalties and rewards in the evaluation of a genetic fitness function. Accordingly, a Fuzzy Preference Agent is a core component for such a system. However, layout design rules and preferences are dynamic in nature, as people learn new concepts and outgrow old ideas, pronouncing the need for the designers to update the design rules. Such dynamic nature of rules suggests that an online artificial neural network based Pattern Discovery and Validation Agent would be of immense value by providing pattern of design rules and preferences in an automated and self-updated manner.

Fig-1 and 2 appear about here.

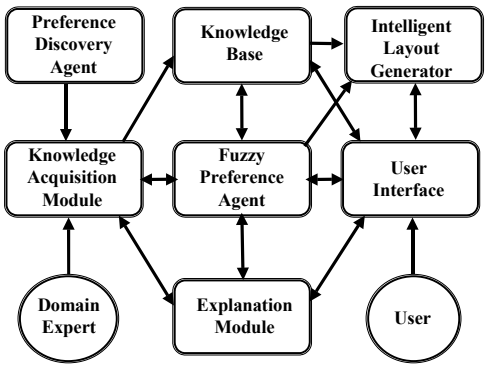

Fig-1: The Proposed Framework.

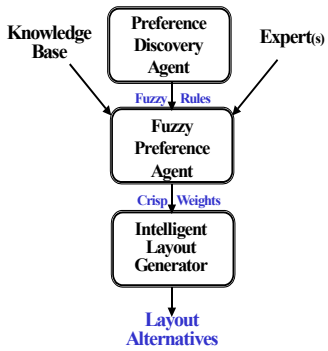

Fig-2: Synergy of Intelligent Modules.

Various tasks involved in layout design include selection of the layout space(s). However, for the purpose of this research, it is assumed that designers have a general structure of the design space and the rest of the process requires spatial arrangements of rectangular modules in the module placement areas. These rectangular modules could be buildings in facility layout, macrocells in VLSI, or product impressions in an E-store. At a macro level, the system accepts a collection of modules, which were categorized and assigned a utility value based on suitable criteria. The system then produces a series of superior layout alternatives for further consideration and manipulation by the layout designers.

This paper is a part of an ongoing research endeavor directed at providing intelligent decision aids to the layout designers [1], [3]. The two of the three 'intelligent' components (namely, Intelligent Layout Generator and Fuzzy Preference Agent) have been the primary focus of this research. This is because of our belief that these two components furnish significant amount of automation realizable in generating and manipulating superior layout alternatives and address the core issues in building the

whole system. Furthermore, these two components furnish a vehicle for carrying out further research in this direction. A somewhat detailed discussion of each component of the system is provided here.

Intelligent Layout Generator (ILG)

Genetic Algorithms have been applied to the layout design problem in various ways. However, much of research in this direction deals with problems consisting of identical modules so that the problem could be treated as simple assignment of identical modules to given cells or when the aspect ratio of modules could be varied. Nevertheless, Genetic Algorithms are known to be a promising approach for layout design through generation of diverse and superior layout alternatives [1], [3]. The core of such approaches is quite simple and involves treating layout as a packing problem by defining an 'ordering of modules' and a 'placement heuristic' for placing modules in the order determined. Consequently, an efficient and efficant placement algorithm is critical for generating superior layout alternatives in reasonable time.

Various fast placement heuristics are available in literature such as Bottom-Left (BL) strategy and Difference Process (DP) strategy etc. [5], [16]. Nevertheless, the existing placement algorithms have several laggings such as poor space utilization as depicted in Fig-3 and 4. Furthermore, these heuristics lack the power to generate superior layouts that are more symmetric and coherent, a key determinant of the layout quality. Consequently, we developed specialized placement heuristics that are superior to existing strategies in space utilization as well as quality of layout [3].

Fig-3 and 4 appear about here.

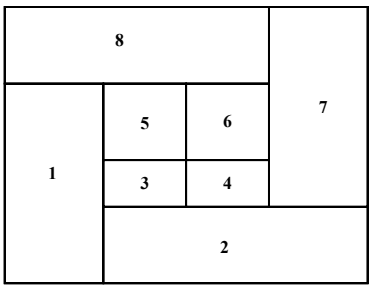

Fig-3: Optimal Layout Not Possible with BL.

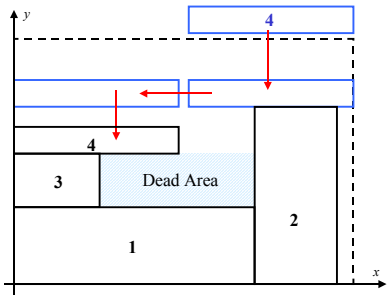

Fig-4: Poor Space Utilization of BL.

It should be noted that flexibility requirements calls for more than one efficient heuristics at the disposal of decision maker. However, here we describe one such heuristic, Minimization of Enclosing Rectangle Area (MERA), developed for our system. The proposed module placement heuristic is motivated by the fact that for any given packing space the number of modules available for placement is a small integer. Consequently, the combinatorial explosion should not become intractable problem if some fast pseudo-exhaustive search is employed for improving space utilization and layout quality in a hierarchical manner. This pseudo-exhaustive technique is summarized here:

Step 1: Place module 1 at the bottom-left corner of the page
Step 2: Set *OBJ* to a big value
Step 3: FOR $K = 2$ to Blocks (Blocks = no. of modules available for placement)

```
FOR L = 1 to Nplaced        (NPlaced = no. of modules already placed)
    FOR A = 1 to 4              (select each corner of module L)
  FOR B = 1 to 4              (select each corner of module K)
            Place corner B of module M_K on corner A of module M_L
            Check Overlap conditions
            Check Boundary conditions
            IF No_Overlap_Violation AND No_Boundary_Violation
THEN
                Calculate the newOBJ
            IF newOBJ is less than OBJ THEN
             OBJ = newOBJ; SAVE placement of module M_K
  END B
END A
  END L
END K
```

Step 4: Stop when there is no room for more modules.

Our studies have demonstrated the superiority of MERA over BL in terms of space utilization as well as quality of outcome. For instance, a run of 100 random sequences for a randomly generated problem of 50 modules showed that MERA resulted in superior outcome in terms of packing height (lower the better) and module tightness (the difference between enclosing rectangle area and the sum of areas of placed modules – higher the better). Furthermore, a couple of facility layout design researchers and practitioners were to provide subjective ranking of layout alternatives in terms of symmetry on a scale of 1-10. Once again, MERA received significantly higher scores than BL. The summary of these results is provided in Table-1. An instance from our experiments with eight-module problems is shown in Fig-5 and Fig-6 for visual comparison purposes. However, this improved quality comes at the cost of computational time with MERA consuming several orders of the time needed by BL. Nevertheless, the resulting quality and diversity of layout alternatives sanction it as a worthwhile trade-off. Detailed discussions on computational complexity are reported elsewhere [3].

	Packing Height		**Module Tightness**		**Layout Quality**	
	BL	MERA	BL	MERA	BL	MERA
Wins	4	96	3	97	11	89
Best	66.01	60.61	80.79	86.51	6	7.5
Worst	93.89	76.51	51.73	69.91	1.5	3.5
Mean	77.76	68.35	68.78	79.33	3.41	5.45
Std. Dev.	9.51	7.01	5.08	3.75	1.54	1.25

Table 1: Comparison of BL and MERA

Table-1 appears about here.

Fig-5 and 6 appear about here.

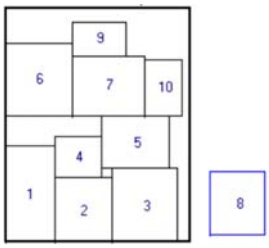

Fig-5: Packing generated by BL

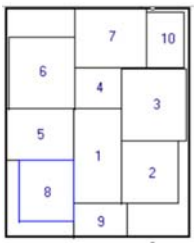

Fig-6: Packing generated by MERA

In addition, the Intelligent Layout Generator incorporates expert knowledge and user preferences in the design process through composite fitness functions comprising of such objectives as contiguous remainder, packing height, module tightness, aspect ratio, symmetry, balance, density, etc. These fitness functions utilize crisp weights furnished by the Fuzzy Preference Agent. The users are given option to manually control these weights as well as fuzzy rules and membership functions, if desired. Such an approach results in a more encompassing and flexible layout optimization process.

Fuzzy Preference Agent (FPA)

The brain of an expert system is the Inference Engine that contains general algorithms capable of manipulating and reasoning with, and about, the knowledge stored in the knowledge base [13]. It should be noted that the knowledge involved in solving the layout design problem is usually imprecise, incomplete, inconsistent and uncertain. Imprecision refers to values that cannot be measure accurately or are vaguely defined; Incompleteness refers to the unavailability of some or all of the values of an attribute; Inconsistency refers to the difference or even conflict in the knowledge elicited from experts; Uncertainty refers to a subjective estimate about the value or validity of a fact or rule [2].

Typically, ad hoc approaches, such as Certainty Theory, are used for managing uncertainties in traditional expert systems employing approximate reasoning [11], [13]. It uses certainty factors to represent the level of belief in a hypothesis given that a particular event has been observed [2]. Other typical statistical approaches for uncertainty management are based on Bayesian rule of evidence that revises the probabilities involved with certain events under the manifestation of new evidence, propagating uncertainty throughout the system. However, such an approach has inherent limitations including the need for repeated experimentation in order to come up with the viable probability values. Relative merits as well as inherent limitations of such traditional approaches can be found in literature [2], [11].

Fuzzy Logic has also been successfully exploited for representation of, and reasoning with, the available knowledge in an expert system [11]. It furnishes a very natural representation of human conceptualization and partial matching. The human decision-making process inherently relies on common sense as well as the use of vague and ambiguous terms. Conceivably, most design guidelines and rules are inherently vague, competing, and even conflicting in nature rendering fuzzy technology an excellent candidate for the implementation of the inference engine. In addition, fuzzy logic focuses on the imprecision of the event itself; whereas, Certainty Factors and Bayesian Probabilities are concerned with the imprecision associated with the outcome of a well-defined event [11].

The core idea is to use a Fuzzy Preference Agent comprising of fuzzy sets, rules and preferences for obtaining penalties and rewards for hybrid fitness evaluation functions. Such multi-criteria fitness functions are more appropriate for automatic generation, evaluation, and comparison of layout alternatives than traditional rigid fitness functions. Consequently, it delivers a good deal of flexibility in the automated layout design process.

The underlying concept of fuzzy set theory is that an element belongs to a set exhibiting fuzzy boundaries with a certain degree of membership. As such, a proposition is neither True nor False, but may be true or false to some degree. This degree is usually taken as a real number in the interval $[0, 1]$. As an example, experts can describe preferences for spacing desired between modules (white space) in layout in fuzzy terms as 'small', 'medium' or 'large'. A fuzzy set A of universe X defined by a function $\mu_A(x)$ is known as the Membership Function (MF): $\mu_A(x) : X \rightarrow [0, 1]$. Where $\mu_A(x) = 1$ if x is totally in A, $\mu_A(x) = 0$ if x is not in A and $0 < \mu_A(x) < 1$ if x is partly in A.

One of the foremost requirements in application of FL is the determination of MFs through experts' knowledge. The typical MFs used in fuzzy knowledge-based systems are the triangular and trapezoidal functions as those provide an adequate representation of experts' knowledge and significantly simplify the computational process [11]. As such, we have employed triangular and trapezoidal MFs in our prototype. It should be noted that certain parameters could have significant interaction with one another affecting more than one value of crisp weights used subsequently in the layout evaluation phase. Consequently, as a future research direction, we intend to develop some mechanism through which FPA can handle such interactions and interdependencies. The ability of FL to realize a complex non-linear input-output relation as a synthesis of multiple simple input-output relations can prove invaluable in this regard [2].

For elaboration purposes, we consider a simple example where the amount of 'white space' and the 'size of bin' affect the maximum number of 'bin modules' that could possibly be placed in a single bin. This is an important parameter to be determined for the efficiency and efficacy of the whole process. For instance, it would affect the length of chromosome chosen in a Genetic Algorithm (GA) designed to automatically generate superior alternatives. It has dramatic effect on efficiency and quality of results as it determines the search space in a GA. We considered a simple bin-packing problem in which the limited size of the bin might leave some modules outside the resultant layout. In our example, we let *x, y,* and *z* (*white_space, bin_size,* and *bin_modules* respectively) be the linguistic variables; *A1, A2,* and *A3 (small, medium,* and *large*) be the linguistic values determined by fuzzy sets on the universe of discourse *X (white_space)*; *B1, B2, B3* and *B4 (small, medium, large* and *ex-large*) be the linguistic values determined by fuzzy sets on the universe of discourse *Y (bin_size)*; *C1, C2,* and C*3 (small, medium,* and *large*) be the linguistic values determined by fuzzy sets on the universe of discourse *Z (bin_modules)*. The MFs for these linguistic variables are shown in Fig-7. We considered a simple two-input one-output scenario involving the following two rules:

Rule 1: If x is $A2$ (*white_space* is *medium*)
Or y is $B3$ (*bin_size* is *large*)
Then z is $C2$ (*bin_modules* is *medium*)

Rule 2: If x is $A3$ (*white_space* is *large*)
Or y is $B4$ (*bin_size* is *ex-large*)
Then z is $C3$ (*bin_modules* is *large*)

Fig-7 appears about here.

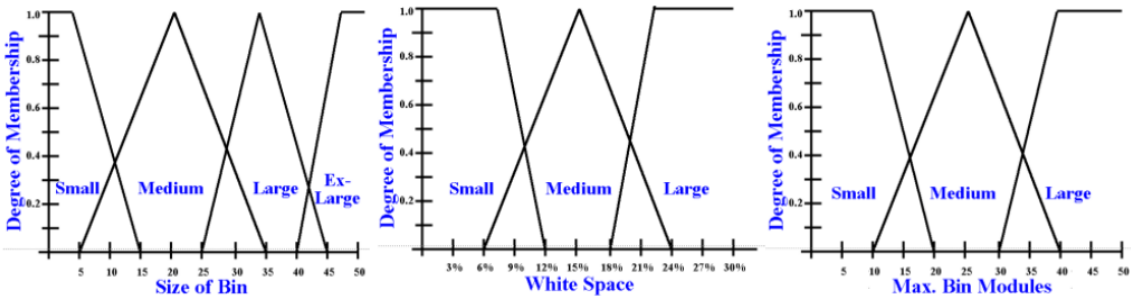

Fig-7: Typical Fuzzy Sets for 'White Space', 'Bin Size' and 'Bin Modules'.

Fig-8 appears about here.

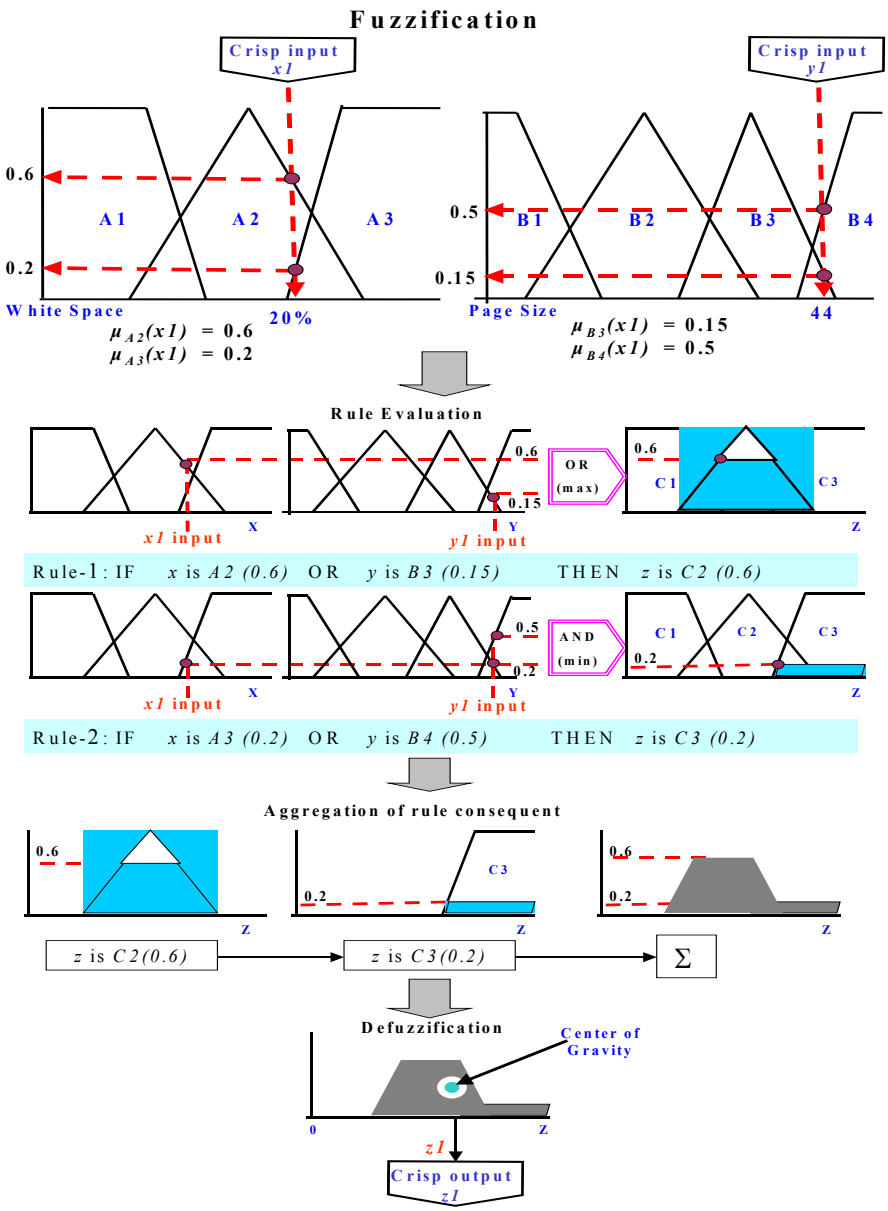

Fig-8: Example of Mamdani style Fuzzy Inferencing in Layout Design.

As a first step, we *fuzzified* all the crisp inputs and determined the degree to which these inputs belong to each of the appropriate fuzzy sets. The crisp input *x1* (*white_space* rated by experts as 20%) corresponds to the MFs *A2* and *A3* (*medium* and *large*) to the degrees of 0.6 and 0.2, respectively. Likewise, the crisp input *y1* (*bin_size* rated as 44 units) corresponds to the MFs *B3* and *B4* (*large* and *ex-large*) to the degrees of 0.15 and 0.5, respectively. The *rule evaluation* involved applying the fuzzified inputs to antecedents in the fuzzy rules. Here we used the *min* operator to evaluate the fuzzy *OR* operation and the *max* operator to evaluate the fuzzy *AND* operation, respectively. This resulted in the following:

$$\mu_{C2}(z) = max[\mu_{A2}(x), \mu_{B3}(y)] = 0.6; \qquad \mu_{C3}(z) = min[\mu_{A3}(x), \mu_{B4}(y)] = 0.2.$$

The result of antecedent evaluation is applied to the MF of the consequent by 'clipping' the consequent MF to the level of the truth-value of the rule antecedent. The *Aggregation* involved unification of the outputs of all rules. In this regard, we used the clipped or scaled consequent MFs. This way we evaluated the fuzzy layout design rules; however, the final output of the FPA needs to be a crisp number for use in the fitness function. The most popular technique for *defuzzification* is the 'centroid' technique where a vertical line carves the aggregate fuzzy set into two equal masses. Using the Mamdani technique in the given example, the crisp value for the *bin_modules* came out to be about 27. This inferencing mechanism is depicted in Fig-8. In this manner, the automatic layout generator adapted itself in terms of 'bin modules' based on preferences furnished by experts/users. It illustrates how vague linguistic rules can be used to derive important and useful crisp values. Likewise, the FPA can be used to furnish other parameters for subsequent use.

Details regarding the implementation of FPA are reported elsewhere [1], [2]. In short, our preliminary studies show that fuzzy logic constitutes an effective tool for inferencing mechanism in layout design. It provided greater flexibility, expressive power, and ability to model vague preferences.

Pattern Discovery Agent (PDA)

The subjective and dynamic nature of the layout preferences usually frustrates the creation of an up-to-date knowledge base. In this regard, the importance of knowing decision-maker's needs and expectations through the quantitative analysis of intangible behavioural data cannot be overemphasized. The ability of Artificial Neural Networks to learn from test cases could furnish rules automatically eluding tedious and expensive processes of knowledge acquisition, validation and revision. Such automatic learning of non-quantifiable and dynamic design rules is possible from superior layouts benchmarks as well as experts' evaluation and manipulation of layout alternatives on an ongoing basis. The integration of such an automated learning mechanism would improve the efficiency and efficacy of the layout design process. However, in the absence of the two aforementioned core components, that would exploit those preferences, an effective Preference Discovery Agent could not be developed and tested. As such, currently, we are working on improving the ILG and FPA.

User Interface

The User Interface defines the way in which an expert system interacts with the user, the environment, and other related systems such as databases. An interactive and user-friendly user interface is deemed to be an essential ingredient in rendering the system easy to learn and easy to use. Towards this end, we developed a graphical user interface for our prototype that offers efficacy to experts and simplicity to novices in the layout design field. It has the capability to enter/override preferences and facilitates fast, easy, and visible knowledge-based manipulation of the layout alternatives. It permits the designer to move modules in and out of the layout or rearrange them in the given layout while observing changes in various modes of fitness displays and provides means for analyzing the same layout from multiple perspectives such as ability to reflect on the intuition of other decision-makers. A screenshot of user interface facilitating such

manipulations is shown in Fig-9. Such an interactive interface is critical to the overall success of our system.

Fig-9 appears about here

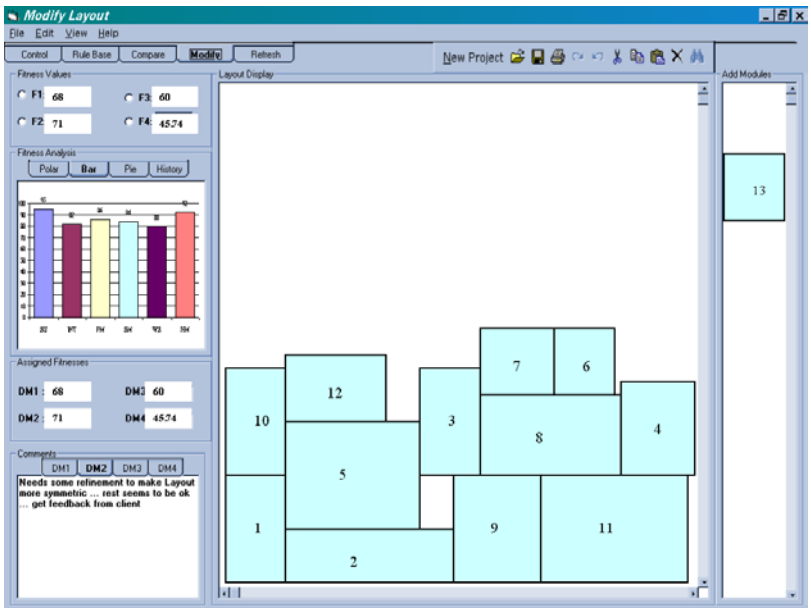

Fig 9: A Screenshot of the User Interface.

Explanation Facility

The ability to trace responsibility for conclusions to its sources is crucial to both transfer of expertise and problem solving [13]. The explanation unit could trace such responsibility and explain the behavior of expert system by interactively answering questions. Indeed, the capability of an expert system to explain the reasoning behind its recommendations is one of the main reasons in choosing this paradigm over other intelligent approaches for the implementation of our concept. Once again, a well-designed, interactive and effective user interface is an important ingredient in enabling a good explanation facility.

Knowledge Base

Knowledge is the primary raw material in any expert system [13]. Typically, knowledge elicitation would continue throughout the lifecycle of the system development and its usage. The conceptual model of the elicited knowledge needs to be converted to a format suitable for computer manipulation through Knowledge Representation. The knowledge of the system consists of facts and heuristics or algorithms. It also contains the relevant domain specific and control knowledge necessary for understanding, formulating and solving problems. There are various ways of storing and retrieving preferences/rules including 'If-Then' production rules. The fact that representing knowledge in form of such traditional production rules enhances the modularity of the system prompted us to adopt this approach. However, traditional rule-based representation cannot easily incorporate new decision alternatives to the expert system, potentially making it inflexible. This furnished another reason for our choice of fuzzy logic for building the inference engine.

Knowledge Acquisition Unit

The knowledge acquisition is the accumulation, transfer, and transformation of problem solving expertise from experts or documented knowledge sources to a computer program for constructing or expanding the knowledge base [13]. It is a major bottleneck in the development of a versatile system mainly because of mental activities happening at the sub-cognitive level are difficult to verbalize, capture, or even become aware of using the cognitive approach of knowledge acquisition from experts. Consequently, the task of extracting knowledge from an expert is extremely tedious and time consuming. Consequently, we envisage our system to have both manual and automated knowledge acquisition capabilities. However, the automated knowledge acquisition is not being tackled during the present course of this research. Currently, the system permits incorporation and manipulation of fuzzy rules and membership functions enabling a more linguistic style knowledge acquisition.

Philosophy and Synergy of Intelligent Components

The IL-DSS differs from traditional expert systems by virtue of three 'intelligent' components namely ILG, FPA, and PDA. We believe, the significance of philosophy and synergy of these intelligent components deserves more discussion. As already mentioned, the ILG generates superior layout alternatives based on pre-specified and user-specified constraints and preferences and the FPA incorporates the soft knowledge in the inference engine. A PDA could complement ILG and FPA by discovering and refining rules/preferences automatically, using an online artificial neural network.

The proposed synergy of ILG, PDA, and FPA is shown in Fig-2. The FPA receives fuzzy preferences and rules from various sources including domain experts, the knowledge base and the PDA. These fuzzy preferences and rules are defuzzified by FPA, furnishing crisp weights for the use in ILG. The ILG, in turn, generates superior layout alternatives for further consideration and manipulation by the designer. The layout alternatives generated by ILG could be validated by the user and/or by the PDA.

As already mentioned, the ILG and FPA are two core components in the IL-DSS due to important roles in automating the layout design process. These two components could work in tandem with PDA to combine the powers of the three main soft computing technologies representing various complementary aspects of human intelligence needed to tackle the problem in hand. The real power is extracted by the synergy; and, a synergistic bliss of expert system with fuzzy logic, genetic algorithms, and neural computing improves adaptability, robustness, fault tolerance, and speed of knowledge based systems [11]. The PDA could not be developed and tested without the underlying core components, like ILG and FPA, being in place. Consequently, we have worked on ILG and FPA first and followed a bottom-up building strategy with ILG being the primary goal followed by FPA. We intend to tackle development and integration of PDA with ILG and FPA later on.

Summary

This work is primarily motivated due to the inadequacy of decision, design and instructional aids available to layout designers. To address these needs, we presented a

framework for an interactive Intelligent Layout design Decision Support System using an Expert System paradigm. It consists of an Intelligent Layout Design Generator that receives crisp preferences from a Fuzzy Preference Agent that, in turn, obtains subjective rules and preferences from various knowledge sources. Furthermore, it supports knowledge-based generation and manipulation of superior layout alternatives. The usual time constraints and frequency of updates required in procurement of a layout justify the proposed and prototyped system as an indispensable and high priority tool. The emphasis of this research is on development of a tool that can supplement the knowledge, experience, and design intuition of the layout designers. Preliminary tests of our IL-DSS prototype have shown immense promise in terms of speed and quality of outcome. In addition, it provides a vehicle to further the research and instructional efforts in this important direction. However, the caveat is that the IL-DSS can result in 'anchoring effect' – the tendency to make decisions based on inadequate adjustment of subsequent estimates from an initial estimate that serves as an anchor. Nevertheless, the potential benefits of systems like IL-DSS outweigh these shortcomings. Furthermore, such shortcomings can easily be evaded with little prudence and creative thinking, which are essential ingredients of any decision-making processes. This paper should furnish researchers and practitioners in layout design area a better understanding of tools and ideas in tackling the layout design problem. In addition, it is expected to stimulate the research in the layout design field and identify other areas craving for such interdisciplinary solution approaches. As future work, we intend to modify ILG to support layout generation in multiple bin scenarios. The promise of automated preference discovery as well as an ability to account for interdependencies and interactions among various preferences in the inferencing mechanism provide very prolific but challenging research streams

References

1. A.R. Ahmad, O. Basir, K. Hassanein, "Intelligent Expert System for Decision Support in the Layout Design", *Working Paper*, Systems Design Engineering, University of Waterloo, 2004.
2. A.R. Ahmad, O.Basir, K.Hassanein, "Fuzzy Inferencing in the Web Page Layout Design", *Proc. of the 1st Workshop on Web Services: Modeling, Architec. & Infrastructure,* France, pp. 33-41, April 2003a.
3. A.R. Ahmad, O. Basir, K. Hassanein, "Efficient Placement Heuristics for Genetic Algorithm based Layout Optimization", *Working Paper*, Systems Design Engineering, University of Waterloo, 2003b.
4. E. L. Blair, S. Miller, "An interactive approach to facilities design using microcomputers", *Journal of Computers in Industrial Engineering,* Vol. 9, pp. 91-102, 1985.
5. K.A. Dowsland, S. Vaid, W.B. Dowsland, "An algorithm for polygon placement using a bottom-left strategy", *Euro J of Op Res.*, Vol. 141 (Special issue on cutting and packing), pp. 371-381, 2002.
6. M.G. El-Said, G. Fischer, S.A. Gamael-Din, M. Zaki, "ADDI: A tool for automating the design of visual interfaces", *Computers & Graphics*, Vol 21, No. 1, pp. 79-87, 1997.
7. Engineering Optimization Software Inc., "VIP-PLANOPT: A Visual Tool for Automating the Design of Facilities Layout", *www.EnggOptSoftware.Bizland.com*, 2003.
8. M.R. Garey, D.S. Johnson, "*Computers & Intractability*", W.H. Freeman Press, New York, 1979.
9. M.M.D. Hassan, G.L. Hogg, "On constructing the block layout by graph theory", *International Journal of Production Research*, vol. 32, No. 11, pp. 2559-2584, 1994.
10. M. Mir, M.H. Imam, "A hybrid optimization approach for layout design of unequal-area facilities", *Computers and Industrial Engineering*, Vol. 39, 2001.
11. M. Negnevitsky, "*Artificial Intelligence: A Guide to Intelligent Systems*", Pearson, Sydney, 2002.
12. J.A.Tompkins, J.White, Y.A.Bozer, et al., "*Facilities Planning*", 3rd Ed., John Wiley, NY, 2002.
13. E.Turban,J.Aronson,"*Decision Support Systems & Intelligent Systems"*,6th ed.,PrenticeHall,NJ, 2001.
14. Unigraphics Solutions Inc., "VisFactory and FactoryOpt", www.ugs.com, 2003.
15. R. Lee, J-M.Moore, "CORELAP-computerized relationship layout planning", *Journal of Industrial Engineering,* Vol. 18, No. 3, pp.195-200, 1967.

16. T.W. Leung, C.K. Chan, M.D. Troutt, “Application of a mixed simulated annealing-genetic algorithm heuristic for the two-dimensional orthogonal packing”, *Euro.J. of Op. Res*, Vol. 145, pp. 530-542, 2003.
17. D. S. Johnson, M. R. Garey e F. R. K. Chung, “On packing two-dimensional bins”, *SIAM J. Alg. Disc. Meth*. Vol. 3, pp. 66-76, 1982.
18. J.A. Svestka, “MOCRAFT: a professional quality microcomputer implementation of craft with multiple objectives”, *Computers & Industrial Engineering*, 1990, Vol. 18, No. 1, p. 13-22.
19. J. Foley, W. Kim, S. Kovacevic, and K. Murray, “UIDE-An Intelligent User Interface Design Environment”, In J W Sullivan and S.W. Taylor (Eds.), *Intelligent User Interface*, ACM, NY, 1991.
20. J.M. Seehof, and W.O.Evans, Automated layout design programme, *Journal of Industrial Eng.,* Vol. 18, No. 12, pp. 690-695, 1967.