FROM QUANTITIES TO QUALITIES IN EARLY-STAGE HOSPITAL SIMULATION

Gabriel Wurzer[(a)], Wolfgang E. Lorenz[(b)]

[(a)] Vienna University of Technology
[(b)] Vienna University of Technology

[(a)]gabriel.wurzer@tuwien.ac.at, [(b)]wolfgang.lorenz@tuwien.ac.at

ABSTRACT
Simulation in early stages of hospital planning might not live up to its full potential: Results come as quantitative (crisp) values, whereas early conception specifies requirements in a more qualitative (think: linguistic) fashion. Because of this gap, it is no wonder when planners cannot easily interpret what is computed. In our most recent work, we have addressed this issue through fuzzy analysis: Crisp simulation results are mapped back into linguistic terms, which can then be compared to the requirements set by the planning team. On the one hand, this enables us to communicate "in the right language", on the other hand, we may use this comparison for automatically raising warnings, in case there is a mismatch between the two.

Keywords: Fuzzy Logic, Hospital Simulation, Early-Stage Design.

1. INTRODUCTION

"Linguistic" descriptions of architectural space are ubiquitous in early stages of planning: Requirements are formulated both narratively and with the help of diagrams showing adjacency relationships, preliminary space layout ("schema") and so on. Simulation acting at this stage can compute flows of material and building users, resource utilization, space usage and so on, as has previously been shown in Wurzer, Lorenz, and Pferzinger 2012. However, its utility is sort of limited *as a design tool*, since what is computed is quantitative, but what planners want is *qualitative*. Consider, for example, the adjacency matrix shown in Fig.1a, especially the relation between "Trauma" and "OT" that is specified as "near". Let us also assume we let a simulation act in the accompanying space layout, by recording the average path length of agents crossing between these two spaces (resulting in a number). The problem for planners is now *how to connect this to the specified requirements* ("near"). Which leads us to our

List of Contributions:

- Through fuzzy analysis, we may map quantitative values computed by the simulation back into linguistic terms, which are easily

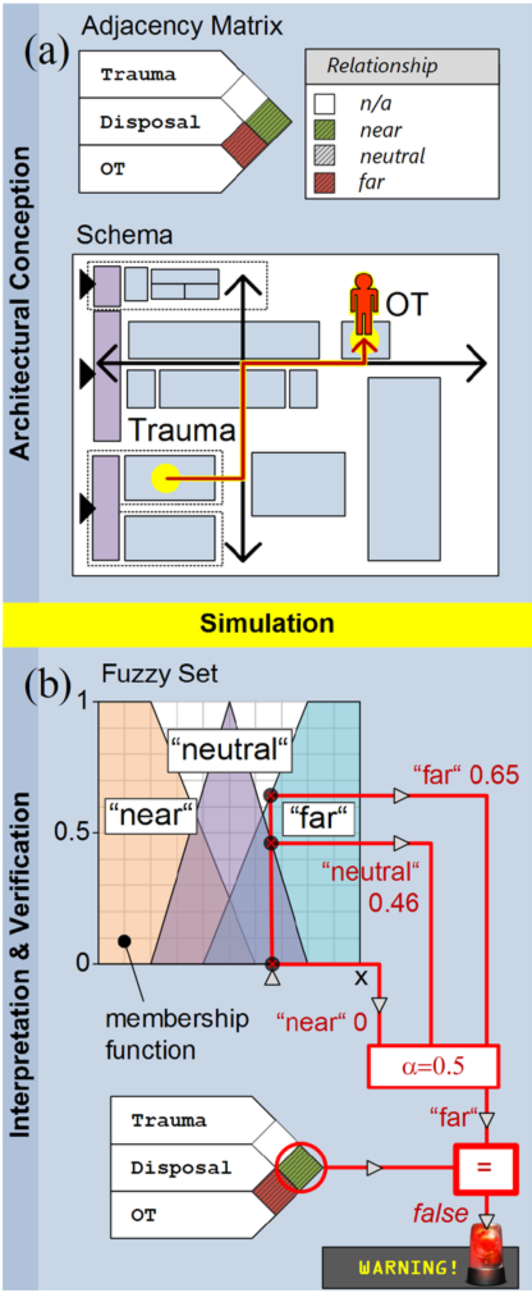

Figure 1: (a) Early-stage hospital concept as input for a simulation, which produces crisp results that are (b) fuzzified and compared with requirements of concept.

understood and compared to the requirements (see Section 3.1.1: Mapping Quantitative to Qualitative).

- We may also automate the checking of computed results against the requirements, establishing *simulation as a design tool* rather than only for analysis and optimization (see Section 3.1.2, “Automated Requirements Checking”).
- There are situations in which there is no 1:1 mapping between computed results and a linguistic term, in which case we may use Fuzzy Inference to take multiple results into account. A discussion of this, together with some background on why such cases can be quite common, is given in Section 3.1.3 (“Mapping Multiple Quantities”).

In the current state of the research, we are applying this approach to an early design of an 800-bed clinic, where we are involved in the pre-planning phase. We are convinced, however, that the ideas we bring forward are a contribution also beyond the borders of hospital design and simulation. For example, buildings which have a comparable degree of formalization - airports, schools, even prisons - necessitate an intensive and constant communication between “planners” and “the client”. Like many other approaches that seek to reduce the complexity of this task, our ultimate goal really lies in the reduction of the communication gap in between the involved disciplines.

2. BACKGROUND

Fuzzy Architectural Spatial Analysis is an upcoming field that describes spatial organization through membership to fuzzy sets - i.e. memberships to linguistic terms in the range [0,1]: For example, a room might be "spacious" (0.8), "average" (0.6) and "small" (0.2) at the same time, albeit to a different degree.

Such fuzzy memberships have been previously used for mapping the perceived qualities of architectural space (Ciftcioglu and Durmisevic 2003), most notably: the formation of spatiality within a given building layout (Arabacioglu 2010). Interestingly, all of the approaches dealing with fuzziness in architecture have so far concentrated on form as primary driver for analysis. In early-stage planning, however, there is no such thing as a definite form; the given schemata and relationship diagrams are rather conceptual: They describe the rules that govern space layout, not the actual geometry that will eventually replace every “rectangle” that stands for a space (see Fig. 1a). This, however, comes in a very later stage called “Form Finding”.

2.1. Related Work

In the architectural domain, Fuzzy Logic (Zadeh 1965) has been widely used for evaluation of the perceived qualities of a given design (De Vries and Steins 2008; Holicky 1999; Durmisevic, Ciftcioglu and Sariyildiz 2001; Durmisevic 2002). However, hardly any work on early-stage applications exists, one notable exception being the fuzzy form generation approach presented Koutamanis (2007). To us, it seems that the obsession with form distracts from early-stage applications, most notable, functional design (often also called conceptual design or pre-design).

2.2. Concepts used in Fuzzy Analysis

Fuzzy analysis is especially good at handling the linguistic terms. In fact, it is one of the only methods that can comfortably cope with their uncertainty, building on expert knowledge entered as *membership functions*. More formally, these regard the set of crisp input values X that come from simulation - e.g. average distances that agents have to cross between elements of the schema - and map these to a set A of linguistic terms (e.g. distance={"near","neutral","far"}, also see Fig.1b) that are represented by a membership function

$$\mu_{A_i}: X \rightarrow [0,1] \qquad (1)$$

to determine the degree of membership for that linguistic term, e.g. "near" (0.6), see Fig.1b. Trivially, membership 0 means no member and 1 full member of a linguistic term. A full elaboration on the application of this is given in Section 3.1.1, “Mapping Quantitative to Qualitative”.

We can use a threshold α above which regard the linguistic term satisfied (α-*cut*). In this case, the membership degree can be omitted (see lower part of Fig.1b, in which only “far” survives the α-*cut*). The so-obtained linguistic term can be compared to a linguistic constraint, e.g. an adjacency relation that was defined between two functional areas such as “near” defined in Fig.1a. A further elaboration of this is given in Section 3.1.2, “Automated Requirements Checking”.

Sometimes, it is not enough to interpret membership functions directly. Cases where multiple linguistic terms are combined via rules to derive a conclusion are comfortably handled via *fuzzy inference*: For example, a mechanism commonly known as *Mamdani's direct method* can consecutively apply a set of rules of the form *IF condition THEN conclusion*, or, in mathematical terms:

$$\begin{aligned} &Rule_i: \qquad (2)\\ &\quad IF\ x_i\ is\ term\ in\ A_i\ [\ AND\ y_i\ is\ term\ in\ B_i\ |\\ &\qquad\qquad\qquad\qquad OR\ y_i\ is\ term\ in\ B_i\ ]\\ &\quad THEN\ z_i\ is\ term\ in\ C_i \end{aligned}$$

where x_i, y_i and z_i are variables and *term* is a membership function. A_i, B_i and C_i are sets of linguistic terms as described above. Also note that the *AND* and *OR* parts are optional and mutually exclusive.

The derivation of the rules proceeds as follows: In the first step, called fuzzification, the membership degree of every condition term $\mu_{A_i}(x_i)$ and $\mu_{B_i}(y_i)$ is computed. The second step then evaluates the *IF-THEN*-rules using the computed membership degrees:

- Rules consisting of a single condition $IF\ x_i\ is\ term\ in\ A_i$ simply evaluate to their membership degree $m_i = \mu_{A_i}(x_i)$.
- Rules containing a logical *AND*, i.e. $IF\ x_i\ is\ term\ in\ A_i\ AND\ y_i\ is\ term\ in\ B_i$, are evaluated to the minimum of the left and right part of the condition, i.e. $m_i = \min\left(\mu_{A_i}(x_i),\ \mu_{B_i}(y_i)\right)$.
- Rules containing a logical *OR* are evaluated to the maximum, i.e. $m_i = \max\left(\mu_{A_i}(x_i),\ \mu_{B_i}(y_i)\right)$.

For every rule, the computed membership degree m_i is then used to clip the membership function $\mu_{C_i}(z_i)$ of the rule's conclusion part. All values above the line given by $y = m_i$ are cut off, giving the new membership function $\mu_{C_i'}(z)$:

$$\mu_{C_i'}(z) = min\left(m_i, \mu_{C_i}(z)\right)\ \forall z \in C_i \quad (3)$$

As third step, all cut membership functions are aggregated by determining their union:

$$\mu_C(z) = max\left(\mu_{C_1'}(z), \mu_{C_2'}(z), \ldots, \mu_{C_n'}(z)\right) \quad (4)$$

The last step, called *defuzzyfication*, generates a single crisp number out of the aggregated function $\mu_C(z)$ by finding its center of gravity, i.e. the point on the x-axis where a vertical line would slice the aggregate function into two equal areas:

$$COG = \frac{\int_a^b \mu_A(x)x dx}{\int_a^b \mu_A(x) dx} \quad (5)$$

3. ELABORATION

Examples of qualitative requirements that are specified in early planning are found in the adjacency matrices or bubble diagrams (White 1986):

- **Adjacency matrices** (see Fig.1a) specify the relations between functional areas. The semantics of this are manyfold, as architects have always used these in a variety of conotations, for example: Spatial proximity (“near”, “neutral”, “far”) or interaction between two units (“high”, “average”, “low”). Observe also that such relations may be either positive or negative, apart from having a degree.
- **Bubble Diagrams** are a variation of adjacency matrices, in which each function becomes a node of a graph and each relation becomes an edge between two nodes. Therefore, what has been said earlier also applies to this type of diagrams.

The question is now, how to take these into account when interpreting the results of a simulation, which (in our case) runs within the **architectural schema** (also refer to Fig.1a): This diagram shows the preliminary space layout, consisting of an arrangement of spaces. The area that these spaces take up is typically given in the *space allocation program* (which is yet another form of requirement - albeit quantitative) that comes with the competition or tender documentation.

3.1.1. Mapping Quantitative to Qualitative

We simulate the flow of patients, staff and material through the preliminary hospital schema, computing the utilization of resources (e.g. waiting seats) and spaces (e.g. the whole waiting area). In more detail, patient arrivals and functional areas visited are identified via the Hospital Information System (HIS) of the current hospital, in order to be able to allocate agents within planned new schema of the same hospital (an 800-bed clinic close to Vienna/Austria). In the new schema, every function can be limited in capacity, so that we can simulate resource allocation and queueing. As results, we gain:

- The utilization of each function over time. We can fuzzify this as “low”, “average”, “high”, even though also comparing against the capacity might be a better measure (see 3.1.3, “Mapping Multiple Quantities”).
- The number of patients crossing between each two functional areas (taking intermediate goals into account). Fuzzyfied, this gives the interaction (“high”, “average”, “low”), which can be compared with the adjacency matrix.
- The average path length for a transition between two functional areas can be used in its fuzzified form for obtaining proximity (“near”, “average”, “far”) *as experienced by a patient*. Note that, again, intermediate goals are taken into account.

The examples stated above are only examples for quantities to be mapped into qualities which can be compared to the preset requirements. For a much more elaborate list of these in the context of planning, we forward interested readers to Durmisevic, Ciftcioglu and Sariyildiz (2001).

What still needs to be discussed, however, is how we come to the membership functions in the first place. We have observed that there is no “standard” guidance on this topic, since for example the meaning of “near” or “far” varies even among hospitals. For example, an acute hospital will likely be very strict in what it means by “near” and “far”, while a general hospital might balance out the categories a bit. In line with literature, we thus argue that this is really expert knowledge, which must be sampled per project (which is what we did, even though this was not easy and certainly not a topic of common agreement among the professionals either - we had to find a balance).

3.1.2. Automated Requirements Checking

In addition to producing linguistic results, a consistency checker can automatically compare what stands in an adjacency matrix versus the results obtained as described in Section 3.1.1. However, this comparison must take the membership degree into account (or, it may use $\alpha - cuts$, as described earlier, for filtering). It must also know the semantics of the linguistic terms. Consider, for example, the four example cases example given in Table 1 concerning proximity between two functional areas: In the first case, the required proximity is not stated; hence, it does not matter what the simulation result is - now warning is raised. In the second case, “near” is required but “far” was computed - clear case for a warning. In the third case, proximity “near” versus “average”, the case is not so clear: It really depends on the project if a warning should be raised. The same goes for case four, in which “average” is demanded but n/a is produced (e.g. because the number of agents crossing between the two functional areas is too insignificant to conclude anything).

Spatial Proximity	
Required	*Simulation Result*
n/a	“near”
“near”	“far”
“near”	“average”
“average”	n/a

Table 1: Example of linguistic comparison

3.1.3. Mapping Multiple Quantities

There are cases where one quantity alone is not sufficient for obtaining a quality. Consider, for example, the newly introduced quality “collaboration” in Figure 2: Let us say there are only two membership functions for this, “good” and “bad”, where only “bad” is shown in the example. Then, one rule for obtaining “bad” out of values for proximity and interaction would be:

IF proximity is far AND interaction is high THEN collaboration is bad (6)

In practice, all combinations of proximity and interaction would be specified as rules in this fashion, before evaluated by a fuzzy inference system as presented in the fashion described under Section 2.1. As has been noted in literature (Kosko 1996), however, combining all input variables with each other leads to an exponential growth in fuzzy rules, also known as the *curse of dimensionality* (Bellman 1961). Furthermore, a verification concerning the consistency of the rules becomes an issue, not only because of the high number of these but also because of the information loss that occurs through the application of the AND operator (i.e. minimum function). If several rules need to be satisfied at the same time, the accumulated imprecisions can

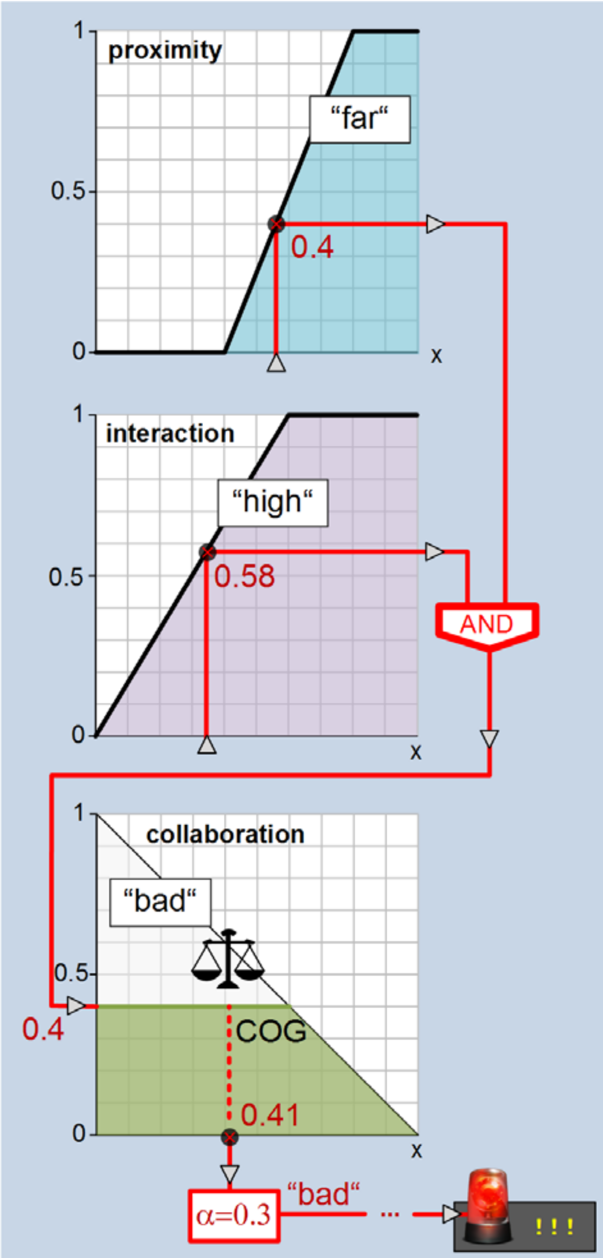

Figure 2: Mapping multiple quantities to a quality. In a first step, the membership function(s) of the composite quality are set up. Through combination (AND - minimum, OR - maximum) of the membership degree of the memberships that are to be combined, we get a combined degree that is subsequently used to clip the membership function of the composite quality. The center of gravity of the resulting function is the composite degree.

effectively outweigh the inference model, making it effectiveless. Another limitation of Mamdani inference is that it does not allow for making a difference in relative importance among rule premises. For example, one notes that the degree by which adjacencies are deemed important varies - White (1986) terms some demands as ‘vitally important,’ and others as ‘desirable,’ alternatively he refers to them as ‘crucial’ or ‘helpful,’ indicating this common difference;

performance assessment should take this information into account, which is a challenging issue in general, and remains unresolved with respect to Mamdani approach.

Several alternative approaches have reported: For example, neuro-fuzzy networks introduce machine learning as means to select appropriate rules, at the cost of transparency. Supplemental to this, Genetic Algorithms (GAs) can also help in this respect (Ciftcioglu, Sariyildiz and Bittermann 2007). Second, using the alternative Takagi-Sugeno inference model (Takagi and Sugeno 1985) can decrease (but does not alleviate) the mentioned shortcoming. Third, the shortcomings in relative importance among the linguistic terms have been put into the linguistic comparison (Section 3.1.2.) rather than being part of the inference, for pragmatic reasons: One might also introduce weights (as in tree-based neuro-fuzzy approaches).

4. IMPLEMENTATION RESULTS

Currently, we have two implementations: The basic flow simulation (written in Microsoft Visio) that simulates the movement of patients between functional units. Each functional unit can consists of multiple functions, which can either be unbounded (resource with unlimited capacity) or bounded - in which case we simulate queuing. In all cases, functions record their utilization, agents remember their agent history and so on; we can draw a lot of data already from this model, alas, there is no notion of actual circulation or way system within the building. Thus, this simulation cannot offer insights into the path lengths used by each individual, which is the basis e.g. for estimating transitioning lengths. We have opted to write a second implementation of a more academical nature, which tries to present the ideas given in this paper and extends on the missing circulation/way network part of our base implementation. The program (which was written in NetLogo) is given in Figure 3:

- Schemata can be generated automatically or imported from a sketch (sic!). The first functionality is purely exploratory - it helps in finding a basic decomposition of a clinic. Once this is known, a manually drawn sketch can capture the essence of what an architect wants to have better, and thus we have provided a vectorizer/tracer that can infer schemata from hand-drawings. Similar techniques have been employed in Architecture for some time now, see for example Koutamanis (1992).
- Functions are attributed to spaces randomly from a preset pool, in order to be able to vary not only the basic spatial structure but also the functional assignments within.
- After manual positioning of entrances, a flow simulation is performed. Agents are given a chain of functions to visit (goals). Crossing a function increases its utilization. Secondly, if coming from a previous function, we record an interaction between both units. These interactions can then be checked against a preset adjacency matrix (see middle part of Figure 3), highlighting areas where the linguistic specification ("low", "average", "high") is not met.
- The interaction check uses the presented fuzzy concept for three single membership functions. We map the quantitative value for interaction between each two functions. For example, we might have an interaction of 10 between OT and ED, which gives low=0.3, average=1.0 and high = 0.2, depending on the choice of membership functions used. The preset is "low", thus we take the low membership of 0.3 and apply a threshold=0.5. Since 0.3 < 0.5, we may not conclude that the OT and ED have a "low" interaction. Thus, we highlight the corresponding entry in the adjacency matrix red (middle-left part of Figure 3).

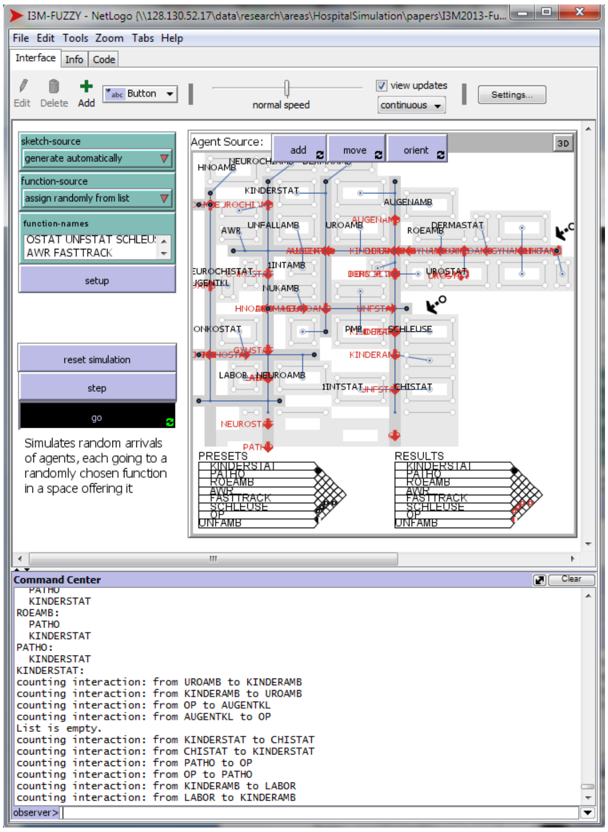

Figure 3: Screenshot of the NetLogo implementation featuring generation of floor plans and evaluation of interaction by comparing simulation results with the preset adjacency matrix.

5. SUMMARY AND FUTURE WORK

In this paper, we have argued for benefits of fuzzy logic as means to map the results of a simulation performed in the context of early-stage hospital design back into the language of the requirements, in our case: adjacency matrix and bubble diagram. As contributions, we have targeted three fields: (1.) The mapping of quantitative values to qualitative statements, (2.) automated checking of qualitative results against requirements and (3.) mapping multiple quantities to qualities.

Summing up, our approach works fine in the case presented - adjacency requirements versus flow simulation within the architectural schema, but is clearly limited in handling a larger set of quantities being combined, because of inherent complexity issues inherent to our underlying inference system.

Addressing the issues in an extended model is clearly on our wish list, respective future work items having been highlighted in the text; however, these require extensive effort not only with regard to the underlying technical concepts, but also when considering our target audience: Our implementation is currently intended for clients (hospital holdings, clinics) and planners (architects and process planners), such that we can give them a design tool which is simple and offers some (limited) capabilities for process simulation. In the light of this, it is vital for us that what the software does is rather unsurprising, as the users are quite unfamiliar (if not reluctant) with regard to the technical concepts in the background. It is thus no surprise that we have first concentrated on the direct case, where the 1:1 relationship between the simulation's result and the resulting quality is obvious. Adding the possibilities for multiple quantities to be combined must be equally comprehensible.

ACKNOWLEDGMENTS

Siren image in Figs. 1, 2 by Sebastien Durel is being used under cc-by-attribution, cc-non-commercial, cc-no-derivative, cc-share-alike license. We wish to thank our reviewers for their inputs, especially concerning the use of multiple quantities. We further wish to thank the city of Vienna, which financed parts of this research work under the ((MODYPLAN)) project.

REFERENCES

Arabacioglu, B.C., 2010. Using fuzzy inference system for architectural space analysis. *Applied Soft Computing*, 10 (3), 926–937.

Bellman, R.E., 1961. *Adaptive control processes: a guided tour*. Princeton: Princeton University Press.

Ciftcioglu, O., Sariyildiz, I.S. and Bittermann, M.S., 2007. Building performance analysis supported by GA. *Proceedings of the IEEE Congress on Evolutionary Computation 2007*, pp. 859-866. September 25-28, Singapore.

Ciftcioglu, O. and Durmisevic, S., 2003. Fuzzy Logic in Architectural Design. *Proceedings of the 12th International Conference on Fuzzy Systems*, pp. 79-84. May 25-28, Delft.

De Vries, B. and Steins, R.J., 2008. Assessing working conditions using Fuzzy Logic. *Automation in Construction*, 17 (5), 584–591.

Durmisevic, S., Ciftcioglu, Ö. and Sariyildiz, S., 2001. Quantifying the Qualitative Design Aspects. *Proceedings of the eCAADe Conference*, pp. 111-116. August 29-31, Helsinki.

Durmisevic, S., 2002. *Perception aspects in underground spaces using intelligent knowledge modeling*. Thesis (PhD). TU Delft.

Dutta, K. and Sarthak, S., 2011. Architectural space planning using evolutionary computing approaches: a review. *Artificial Intelligence Review*, 36 (4), 311–321.

Holicky, M., 1999. Fuzzy probabilistic optimisation of building performance. *Automation in Construction*, 8 (4), 437–443.

Koutamanis, A., 2007. Fuzzy Modelling for Early Architectural Design. *International Journal of Architectural Computing*, 5 (4), 589–610.

Koutamanis, A. and Mitosi, V., 1992. Architectural computer vision: Automated recognition of architectural drawings. *Proceedings of the Design and Decision Support Systems in Architecture Conference 1993*, pp. 17-25. Merlo (The Netherlands).

Takagi, T. and Sugeno, M., 1985. Fuzzy identification of systems and its application to modeling and control, *IEEE Transaction on Systems, Man and Cybernetics*, 15, 116-132.

Wilensky, U., 1999. *NetLogo*. Software. Center for Connected Learning and Computer-Based Modeling, Northwestern University. Available from: http://ccl.northwestern.edu/netlogo [accessed 7 March 2013]

White, E.T., 1986. *Space Adjacency Analysis - Diagramming Information for Architectural Design*. Tallahassee: Architectural Media.

Wurzer, G., Lorenz, W.E. and Pferzinger M., 2012. Pre-Tender Hospital Simulation Using Naive Diagrams As Models. *Proceedings of the International Workshop on Innovative Simulation for Health Care 2012*, pp. 157-162. September 19-21, Vienna (Austria). Available from: http://publik.tuwien.ac.at/files/PubDat_209962.pdf [accessed 7th March 2013]

Zadeh, L.A., 1965. Fuzzy sets. *Information and Control*, 8 (3), 338–353.