Object Oriented Process Modeling with Fuzzy Logic

Luigi Benedicenti[¥], Giancarlo Succi[§], Tullio Vernazza[¥], Andrea Valerio[¥]

[¥]Universita' di Genova – Dipartimento di Informatica, sistemistica e telematica
[§]University of Calgary, ENEL
Luigi Benedicenti
Università di Genova, DIST
Via Opera Pia 13, 16145 Genova, Italy
Tel: +39-10-3532173 Fax: +39-10-3532154
E-Mail: Luigi.Benedicenti@dist.unige.it

ABSTRACT

Fuzzy Logic has traditionally found an application in control theory, systems analysis, and artificial intelligence [9] [7]. However, it is very difficult to apply it to traditional software engineering. There is evidence that fuzzy logic can help in avoiding early assignment errors in SW engineering methods [1]. The authors have developed a technique that allows object oriented business process modeling. In this paper they propose a fuzzy logic extension to the method. Fuzzy sets can depict the uncertainty on the cost driver. Thus cost driver uncertainties have less impact on the model. The approach is new since it applies fuzzy logic to process modeling, and not to control or expert systems.

INTRODUCTION

Business process modeling is an activity that seldom relies on clear, unambiguous results. In fact, it involves close interactions with human beings. Humans are a possible source of error. In most cases, only subjective measures are available. Subjective measures are the most error-prone. It is necessary to accept the possibility of errors, and to find a way to minimize their effects on the model.

The authors have developed a business process modeling system called Gertrude [2] [3]. Gertrude relies on objet orientation to provide a smooth modeling environment. The model is also connected with cost information by means of the activity based costing technique (ABC) [5]. Thus there can be a measure of the cost for each activity that constitutes the process.

ABC relies on measures of the utilization of a given resource to determine its cost in a certain activity. Then, the cost is distributed among the products that involved the activity. ABC also accounts for infrastructures. An infrastructure is a non-human resource. For example, a computer server is an infrastructure, but an office can also be an infrastructure.

The inclusion of non-homogeneous infrastructures, however, is not trivial. First of all, it is difficult to maintain a single unit of measure for each cost per activity. For example, the cost of a telephone call is expressed in time units, while the cost of a printer grows with the number of paper sheets it has printed. To account for this, the modeler uses *cost drivers.* Each cost driver links a particular activity with a particular product. The units of measure of cost drivers can be different, so that the division of the cost measured for a particular resource results in a uniform cost that can be compared with all the other final costs.

However, very often there is uncertainty on the determination of the cost driver itself. This is why the adoption of the fuzzy set theory is helpful. This paper is organized as follows: the next section introduces fuzzy sets and fuzzy logic. Then there is a section on Gertrude, including a small example. After that, there is a section on the application of fuzzy logic to Gertrude, that extends the previous example with fuzzy logic. Finally there are some conclusions and plans for future work.

FUZZY SETS AND FUZZY LOGIC

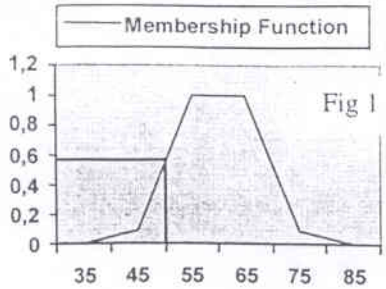
The introduction of fuzzy sets traces back to 1965. Prof. Zadeh [9] has illustrated the concept of fuzzy set as opposed to conventional (crisp) sets. To compare the two concepts, it is necessary to define some terms.

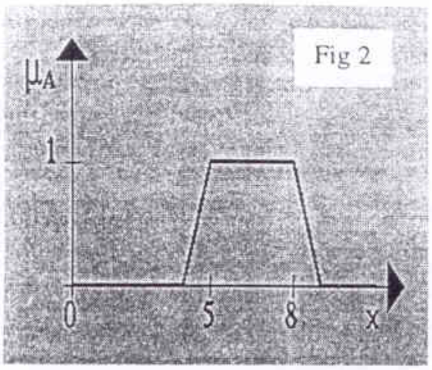
Let the *universe of discourse* X be the set of all possible elements in a formal system. Then a *crisp subset* A_c is any subset of X in the sense of the traditional definition. A_c either contains or does not contain a particular element $x \in X$. Conversely, a *fuzzy set* A couples each element $x \in X$ with a *membership function* $\mu_A(x)$ that indicates the *membership degree* of the element x for the set A. $\mu_A(x)$ maps X into the *membership space* M.

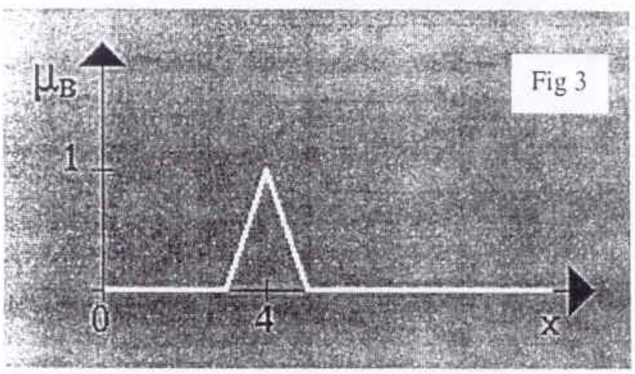
If M includes only the two values 0 and 1, then the set A is a crisp set. For fuzzy sets, M is the interval [0,1]. If X is a continuous set then A is represented by its membership function. Fig. 1 shows the membership function for the temperature x of a thermometer X. For x = 60° F, the temperature is in A.

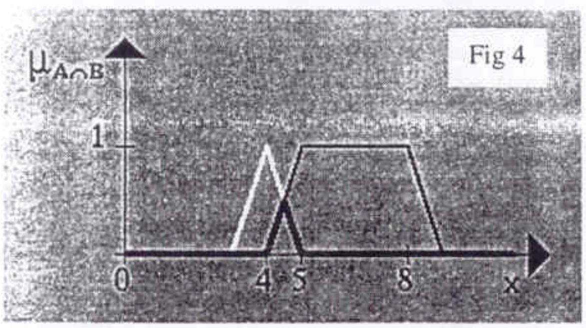
For x = 50 °F, the temperature is only partially in A. It belongs to A with a degree of 58%.

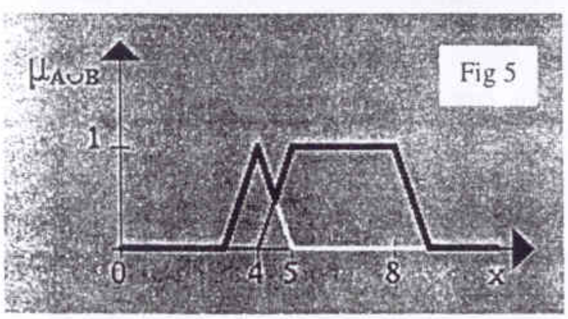
Consider the two sets A and B defined by their respective membership functions in Fig. 2 and Fig. 3. Fig. 4 depicts the results of a fuzzy intersection. In this case, the operator used for the intersection is the minimum of the two functions. Fig. 5 shows the result of a fuzzy union. Here the operator employed is the maximum of the two functions.





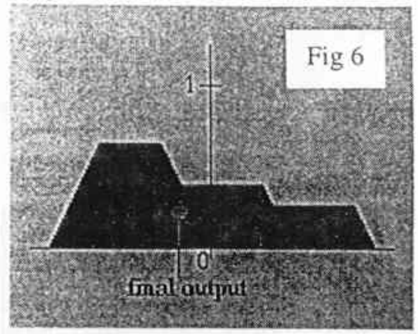






Fuzzy control systems employ a set of rules that trigger single actions. Then they combine the actions and come out with a single value for the control. Each rule refers to one or more fuzzy sets and provides a result that can be another fuzzy set. Fuzzy values are expressed in a conventional linguistic form, such as "low", "high", "medium", "tall", or "open". The corresponding variable holding the *linguistic terms* is a *linguistic variable*. The process of mapping an input variable to a linguistic variable is called *fuzzification*. The inverse process is called *defuzzification*.

The fuzzy logic "and" corresponds to an intersection of the sets being compared. The fuzzy logic "or" corresponds to a union. So the result is a fuzzy set with a membership value given by the union or intersection of some fuzzy sets. The final set is the union of all the fuzzy sets created by rules firing. Fig. 6 shows the final set, and the final value for the output (control): the center of gravity of the resulting composite shape.



BUSINESS PROCESS MODELING WITH GERTRUDE

Gertrude is a business process modeling methodology based on object orientation and ABC [3]. Object orientation is necessary for flexible modeling and customizable abstraction level. The link with Activity Based Costing permits a seamless connection between costs and activities. [5]

Gertrude is an empirical modeling technique based on Ivar Jacobson's Use Cases [6]. It employs the OMT [8] notation for most of the diagrams since they are well known.

The empirical approach in Gertrude is useful to model highly variable processes [4]. In fact, many software firms lack a well defined process.

Gertude relies on four types of objects: activities, people, roles, and infrastructures. The modeling process is straightforward. Starting from the information collected via interviews, modelers construct and refine a model. Eventually, the model converges to a stable version that realistically depicts the firm. Gertrude adopts two different languages: one for the modelers, and the other for the rest of the firm.

The internal language is:

- Technical, (modelers have to be experienced).
- Complete, because it expresses the model.
- Formally defined, for easier transportation into automated graphic tools.

The external language is:

- Simple, because everyone must understand it.
- Synthetic, to cope with abstractions.

The modeler handles the following objects:

- Use cases: the textual description of the instances of the process derived from the interviews.
- Activities interaction diagrams: the representation of the interactions among activities over time.
- People interaction diagrams: the representation of the interactions among people over time.

- Activities Roles People snapshot: the overview of the objects and their main relations.
- Role, process, and infrastructure hierarchy: the static representation of the firm.
- Activities Roles People diagram: detailed analysis of each activity and its requirements.
- Allocation diagrams: the description of the allocation of people to roles, roles to activities, Infrastructures to people, and infrastructures to activities.

The modeler also produces external deliverables (that use the external language):

- Allocation matrices: the description of the potential and actual allocation of people to roles, roles to activities, and people to activities.
- Subparts of the object oriented diagram. These are views over the entire process most useful to project managers and people that do not really need the entire picture of the firm.

The entire view of the process is available for everyone to consult on the basis that it is vital that no information is hidden from personnel.

The collection of ABC data is a continuous process. Therefore, it is the means to evaluate the model and provide feedback from every employee. ABC is not only the technique used to collect alternate cost data for parallel accounting, but is also the herald of poor process design. In fact, ABC can show that people spend too much time switching among different activities (*thrashing*). It can also check the validity of the model. If employees perceive that the activities they perform are different from the ones in the ABC forms, they are free to neglect them and write new activities they feel more pertaining to their job. This enacts a continuous improvement process that makes the model better and better.

The following example depicts one small Italian firm which was modeled when Gertrude was under beta-test. For privacy reasons, personal data has been edited.

Freedom Thread is a small enterprise of four people. They also employ, from time to time, a consultant, for specific tasks. This example presents a subset of the product lines and a high-level abstraction of the production process. The ABC side of Freedom Thread is simplified here and for sake of clarity the number and type of resources have been greatly reduced.

Fig. 7 shows the ARP Snapshot of the firm. It is apparent that the enterprise is centralized and uses a collaborative approach only in the development, testing, and problem solving phases. This is very effective in case of small firms, because responsibilities are clear.

Table I lists the cost per resource. Table II shows the weekly resource consumption breakdown diagram edited from the ABC forms. Table III is the activity distribution profile, depicting the cost per activity in the resource unit of measure.

Table I

Number	Resource	Cost	Unit of Measure
3	Personal Computer	2	US$ / Hour
1	Server	5	US$ / Hour
1	Packet Line	0.01	US$ / Packet

Table II

Person Name	PC1	PC2	PC3	Server	Packet Line
Daniel	1	35	0	20	5000
Mara	10	0	0	1	300
Luke	10	0	10	1	10
Alfred	6	0	20	1	900
John	13	5	5	10	1800
Idle Time	0	0	5	7	NA

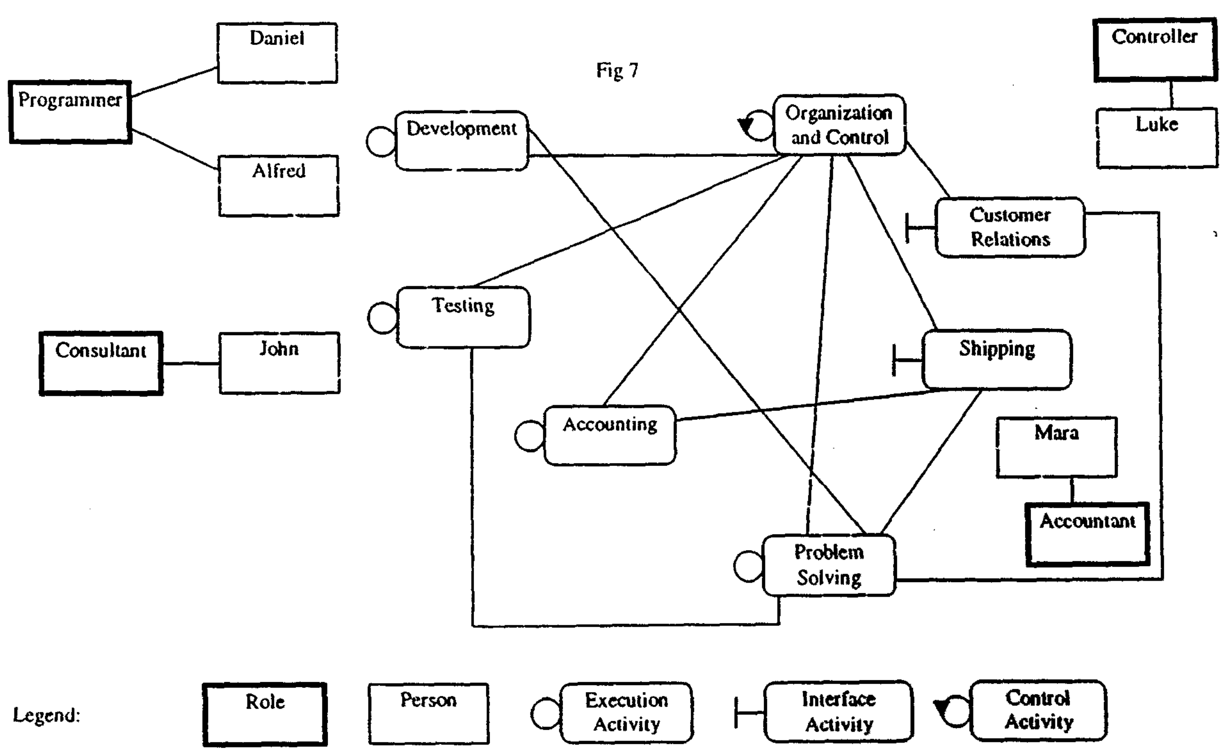


Fig 7

Total	40	40	40	40	8010

Table III

Activity	PC1	PC2	PC3	Server	Packet Line
Accounting	8	0	0	0	1000
Problem Solving	10	0	3	0	3000
Organization and Control	10	0	10	1	10
Shipping	0	0	0	1	1300
Customer Relations	1	0	1	0	1000
Development	1	35	13	16	700
Testing	10	5	8	15	1000
Total	**40**	**40**	**35**	**33**	**8010**

EXTENSION OF GERTRUDE WITH FUZZY LOGIC

The Activity Based Costing requires that costs be assigned to each activity. Then, each activity spans its cost on the different product lines (or singleton products in case of customized software for example).

To account for infrastructures, each activity must bear the cost for each resource it consumed. Each single resource cost must then be transferred on the product lines.

Thus, an activity has many different costs: the cost of the human re-ources and the cost of the resources. Each of these costs must be divided for the product lines. The divisors of each cost are called *cost drivers*.

If every piece of information is known, there is no need for fuzzy logic. However, it is often very unlikely that the data is accurate. Human error, out-of-norm procedures, and the level of abstraction that the model requires to work, sometimes do not allow accurate measurements. In these cases, fuzzy logic is the choice.

It is possible to employ fuzzy logic in the following three cases:

- Attribution of resources to activities
- Determination of activity cost driver
- Determination of resource (per activity) cost driver

The first case occurs when there is a lack of knowledge of resource consumption per activity. The second case occurs when the costs of the activities are known, but there is no precise knowledge of the contribution of some or all of the activities to each product line. This is the most unlikely to occur. Good ABC forms should prevent the occurrence of this problem. The last case occurs when there is not a clear understanding of how each resource consumed during each activity contributed to each product line. This happens often, because it is more difficult to trace the consumption of some resources.

For example, if a developer receives a lot of telephone calls while working, it can be difficult to determine how many of them related to the actual product developed, how many have been help calls on previous projects, and how many personal calls interrupted the workflow. In many cases, only subjective measures such as "half and half", "many calls", or "a few calls" are available. This is an ideal setting for fuzzy logic.

The application of fuzzy logic to any of the previous cases always follows the following procedure:

- Determine the membership functions and linguistic variables for the input and output variables (fuzzification)
- Determine the set of rules
- Fire the rules
- Find the results (defuzzification)
- Normalize the results to the cost drivers

The last step is crucial for cost drivers. The normalization assures that no inflated costs can propagate through the activities to the product lines: in fact, given a single activity and a single cost, the sum of the cost drivers for each product line will yield 1.

The following example is the continuation of the example presented in the previous section. It is the determination of resource cost driver for the small firm depicted in the previous section. This example presents a very simple set of rules for sake of simplicity. However, the number of rules is theoretically unlimited, and so is the number of fuzzy inputs. Modelers are free to choose whichever rules they find useful according to the contingency of the problem.

During the ABC form acquisition, Freedom Thread was developing mainly 4 products:

- P1: a network tunneling protocol
- P2: a proprictary communication protocol
- P3: a human interface component
- P4: a client/server database

Consider Table III. Suppose the goal is to find the cost drivers for the Packet Line consumption in the Customers Relation activity. The input variables are the number of packets consumed for each product, and the global confidence in this value. The output variable is the estimated value of the number of packets consumed for each product. The outputs will be normalized to find the cost drivers for each product line.

Table IV summarizes the values of the linguistic variables for input and output variables.

Table IV

Variables			
Name	Packet Number (Input)	Confidence (Input)	Cost Driver (Output)
Linguistic Values	Minimal	Low	Low
	Typical	Medium	Medium
	Maximum	High	High

Fig. 8 shows the membership functions for each variable. Table V summarizes the rules for this example. The application of the rules with 108 packets with confidence 4 yields the result in Fig. 9.

Its defuzzification results in CD = 150.

Table V

	Confidence = Low	Confidence = Medium	Confidence = High
Packet Number = Minimum	Cost Driver = Medium	Cost Driver = Medium	Cost Driver =Low
Packet Number = Typical	Cost Driver = Medium	Cost Driver = Medium	Cost Driver =Medium
Packet Number = Maximum	Cost Driver = Medium	Cost Driver = High	Cost Driver =High

When all the cost drivers are known, the normalization takes place. This ends the procedure.

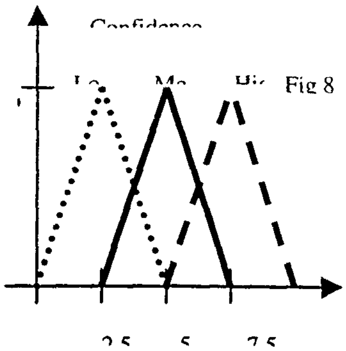
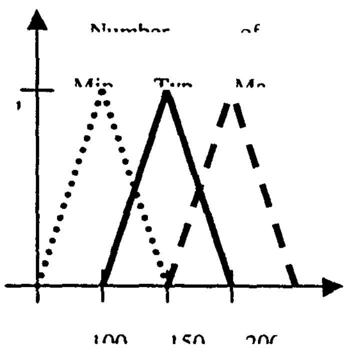
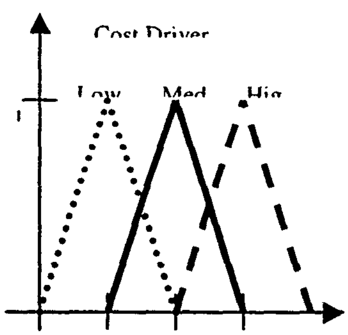


Fig 8





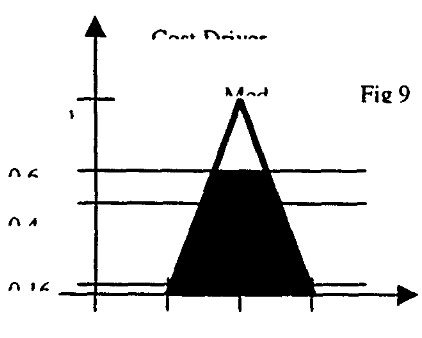


Fig 9

Conclusion and Future Work

This paper presented a method for coping with unreliable data in a business process modeling method. Fuzzy logic techniques can reduce the source of error because they allow for inaccurate data. The modeler defines fuzzy logic rules and then constructs a fuzzy estimating machine.

The application of fuzzy logic has two advantages. First, it allows inaccuracy in measurements. Second, it can effectively cope with uncertain quantities, often expressed by means of linguistic terms such as "little", "some", or "a few". By choosing appropriate rules, moreover, the modeler can incorporate in the model data coming from subjective sources without compromising the validity of the model

The example presented is a very simple case of determination of cost drivers using fuzzy quantities.

In the future, the authors will extend the fuzzy logic method to the sequential ordering of activities. This will permit to execute the model, effectively simulating the process before its execution.

References

[1] Aksit, M. and F. Marcelloni. Reducing Quantization Error and Contextual Bias Problems in Object Oriented Methods by Applying Fuzzy Logic Techniques. In Proceedings of the 10th workshop, ECOOP '97: Modeling Software Processes and Artifacts. June, 1997

[2] Benedicenti, L., P. Predonzani, G. Succi, and T. Vernazza. Gertrude – A methodology for Business Process Modeling (DIST-LIPS-TR-97001). 1997

[3] Benedicenti, L., P. Predonzani, G. Succi, and T. Vernazza. Gertrude: OO for BPR. Proceedings of the IEEE 6th International Conference on Emerging Technologies and Factory Automation. IEEE Press, 9-12 September, 1997

[4] Gruhn, V., W. Schafer. Software and Business Process Technology. Tutorial at ICSE 97, Boston, Ma

[5] Innes, J., F. Mitchell. I costi di struttura - Metodologie di analisi e di gestione. Egea, 1994.

[6] Jacobson, I., M. Ericsson, and A. Jacobson. The object advantage - business process reengineering with object technology. ACM Press, 1995.

[7] Klir G. J., and B. Yuan. Fuzzy Sets and Fuzzy Logic: Theory and Applications. Upper Saddle River, NJ, Prentice Hall, 1995.

[8] Rumbaugh, J., M. Blaha, W. Premerlani, F. Eddy, and W. Lorensen. Object-Oriented Modeling and Design. Prentice-Hall, Englewood Cliffs, NJ., 1991

[9] Zadeh, L. Fuzzy Sets. Information and Control, 8(3), 1965