From: AAAI Technical Report SS-99-07. Compilation copyright © 1999, AAAI (www.aaai.org). All rights reserved.

Interval Processing for Search Control Under Incomplete Data

Pham Hong Hanh

Intelligent Systems Laboratory
Department of Computer Science, Faculty of Science & Technology, Assumption University
Ramkhamhaeng Road, Soi 24, Huamark, Bangkok 10240, Thailand.
phhanh@s-t.au.ac.th

Abstract

In searching under incomplete information, the utilities or probabilities of the alternative candidates and branches may be available only in intervals instead of precise numbers. Thus interval comparison becomes a key point in search control. This paper introduces, reviews and examines the techniques which compare the intervals exploiting the available information. We also discuss when these techniques can and should be used and give an example of applying a specialized interval ranking mechanism for a scheduling problem. Through this paper we would like to link the interval computation and especially interval comparison into the search community as a considerable issue and helpful techniques for search control under uncertainty and incomplete information.

1. Introduction

Computation time in the problems that involve search such as optimization, planning, or scheduling is always a subject to minimize. When the given data is incomplete that not only enlarges the search space and therefore increases the searching time but also causes a new problem of estimating indefinite quantities. In many cases, it is the problem of ranking intervals (18,23,34,35,16,22).

When the incomplete data is given initially in intervals instead of precise numbers, the interval comparison techniques are needed for comparing the inexact values – intervals, defining the "best" ones. Thus interval processing including calculation and comparison can help to embed the existing search methods, which have been designed for the precise numbers, into the environment with incomplete information .

The interval processing techniques remain useful even for many searching problems with complete data. In order to decrease searching time, the search is carried out at abstract or high levels (8-19,15,21,26,27,36), where search space is divided into regions and each alternative is a branch or group of candidates instead of a single variant. Consequently, the utility or probability of an alternative extends to some interval. If we are able to rank these intervals then we can refine the search, lead it to the "good" region or prune it from the "bad" branches. Therefore, interval comparison becomes one of the helpful control techniques to improve the search.

The idea of dividing the search space into parts with estimation in intervals has been proposed in many papers. However, in those works, which address directly search problems, not many interval comparison schemes have been discovered and applied except the traditional methods such as optimistic: maximax (larger if upper bound is larger) and pessimistic: maximin (larger if lower bound is larger). Even less work is done in studying how to use the other available information besides the two bounders of the intervals to sort them. While, indeed, the works on how to estimate and order intervals are hidden or mixed in the papers of different areas such as fuzzy numbers (7,2,3), multi-value algebra (20), continuous logic (32,33), control in real-time systems (6,22), and interval computation (14,17,19).

Therefore, this paper attempts to gather the techniques of interval comparison from the outside fields and to represent them as a unique issue in searching under uncertainty and incomplete data.

The next section describes how intervals derive from incomplete data and uncertainty giving a classification of intervals based on the kind and amount of available information. Section 3 introduces, reviews and analyzes the available interval techniques which compare the intervals exploiting the accessible data. Then, in section 4 we discus when and how these techniques can and should be used. In order to show how much interval processing can help search we examine an example which uses a specialized interval comparison mechanism for a scheduling problem in section 5. Finally, challenges and further research are discussed in section 6.

2. From Incompleteness and Uncertainty to Intervals

In human conversations we usually say "I'll see you after about 2 weeks", meaning 14 days with some tolerance Δ, 2 days for example, or "the bus arrives between 7h00-8h00"

Copyright ©1998, American Association for Artificial Intelligence (www.aaai.org). All rights reserved.

rather than "... after exactly 14 days" or "...precisely at 7h30". Even though we could have a careful plan for the next two weeks and there could be a schedule for the bus, instead of the two precise numbers X=14 and Y=7h30 we have to use the intervals [X-Δ,X+Δ]=[12-16] and [Ya,Yb]= [7,8] to represent the outcomes of our plan and of the schedule. In the first case, that may be because we are not certain on some details in our plan. In the second case, it might be caused by having not complete information about the bus arrive time.

In engineering systems and scientific computation, when the information is incomplete or damaged, intervals are the best way to represent the data with noise or the roughly estimated data. When the computation models involve uncertainty, intervals are used to bound the enlarged outcomes.

In reasoning with uncertainty, the probability takes advantage in representing the degrees of our belief. However, in many cases, even probability itself is given or available only in intervals instead of precise numbers (34,35).

Thus, intervals are the most natural and popular way to represent the incompleteness and uncertainty not only in practice but also in theory. In the papers that involve intervals, the definitions of interval are different and vary from a crust set, a fuzzy number, a fuzzy interval, a probabilistic interval, a real number Xo with some possible error, to a range of a variable.

In search, the heart of any AI engines and systems, we are interested in intervals not only as a form of representing incompleteness and uncertainty but also as tools to organize or control the search in a better way. An interval is known with only the two bounders and an interval defined with some additional information should be treated in different ways. In other words, when the data is incomplete and the environment is uncertain we should maximally use all kind of available information.

Therefore, it would be useful to classify intervals based on the kind and amount of the additional information that they have. Deriving from various situations of uncertainty and at different levels of incompleteness, intervals can be grouped with a fuzzy style as in (29). Given a quantity X those possible values are expected in some interval [a,b]. Then, this interval [a,b] can be:

- ***Pure intervals (PI)***, which are known without any additional information, all we know are the two bounders a, b, and $a \le X \le b$;
- ***Oriented intervals (OI)***, which are known with "a little" additional information such as the direction: starting and finishing points and may also including the rate of how the values of X is changed within [a,b];
- ***Statistical intervals (SI)***, which are known with "more" additional information such as statistical values of X observed in the past.
- ***Complete intervals (CI)***, which are known with "complete" information such as the function or rule by which X receives its value in the domain [a,b].

3. Available Interval Techniques

According to the surveys in (14,19), interval computation started in 1924 with a new breath in 1964. In theory, it should include interval arithmetic and interval comparison. However the interval comparison branch received attention from researchers only since 70s to 80s with many theoretical investigation and analysis. Nonetheless, after that there is not much results have found on this topic.

We focus on reviewing and examining the interval comparison techniques as the direct tools for search control, even though the interval arithmetic may also be needed in preprocessing the incomplete data, for this purpose, there are many resources and tools available online at (12-14,31).

In this section we provide a general survey and an examination emphasized on the recent obtained results with fuzzy approach.

3.1 General Survey

With emphasis on the measurement systems that used in comparing intervals, the works related to interval comparison can be divided in to groups as the following:

- Use ***simple estimation (SE)*** {definite state: less, more for non-overlapped intervals; indefinite state: for overlapped intervals} as in [10,11];
- Use ***traditional estimations (TE)*** {<,=,>} as for precise numbers with assumption that an interval [a,b] can be represented by a (pessimistic approach), b (optimistic approach) or (a+b)/2 (average approach) [24,17,23];
- Use ***abstract estimation (AE)*** with words such as building the set of possible order relations between pairs of intervals {before, meets, overlaps, starts, during, finishes, equal} as in [1,6] ;
- Use ***probabilistic estimations (PE)***, for instance by the four criteria PD, PSD, ND, and NSD for ranking the intervals as in [7,8].
- Use some ***intermediate functions (IE)*** f on each interval with different criteria such as Hurwicz ones [12,18] and then sort the intervals.

A careful review in (29) showed us that the most problematic case is when the intervals overlap. Then, any rigid estimation in TE can not be always true, while estimations in SE and AE are not sufficient for ranking intervals of the similar types. The approach by IE sounds good however sometimes to find an appropriate function f is even more problematic. Only PE provided adequate theoretical basis for estimations intervals in general form, however the comparison schemes with the four fuzzy relations on the indices (7) were very complex and therefore are difficult to be applied in practice.

The roadblock in defining and representing the order of the overlapping intervals using the traditional estimation degrees (>,<,=) can be explained by the uncertainty of the problem. When the intervals overlap such as (3,6) and (2,8), any traditional statement, for instance X1 > X2, can not be always true, as shown in (28).

3.2 Fuzzy Measurement

In order to overcome this barrier we propose *to fight fire with fire* and use a fuzzy flexible measurement: more or less with degree D, which are uncertain in some sense, to describe the irregular or inconsistent relationship between the intervals, instead of the traditional estimation measurement with only 3 choices {<,=,>}. Thus, we have X1 >(D) X2 instead of X1 > X2.

This idea was first proposed in [] with 3 comparison mechanisms for PI. Then one of these methods was modified, improved, and applied in a planning problem []. Thus, for PI we have:

- In Extended Comparison EC (28) we add three degrees "precise", "relative", and "probably" to the classical signs {<,=,>}.
- In Multi-Level Comparison MLC (28) each of the signs {<,=,>}is extended to n levels and we have, for instance, X1 >(n) X2 where n is a positive integer.
- In Continuous Comparison CC(28) we use the notation X1 >(G) X2, where G is a real number and 0<G<1 when the intervals are overlapped, otherwise G=1 or G=0.
- In Complete Continuous Comparison CCC(25) we extend the computation of degree G for non-overlapped cases. Then G measures not the degree of our belief that X1>X2 but "how much X1 is more than X2".

For SI, we have proposed a specialized Statistic-Oriented Estimation method based on a pseudo-distribution measurement (30). We have built three alternative comparison schemes, correspondingly adapts to the three from of distribution discovered from the observed statistic values in the past:

- Value-Based Scheme VBS for intervals with chaotic pseudo-distribution form;
- Frequency-Based Scheme FBS for intervals with one-peak pseudo-distribution form;
- Density-Based Scheme DBS for intervals with non-peak pseudo-distribution form.

For OI, we have built (29) a namely Direction-Oriented-Comparison - DOC method for pairwise comparison of monotonic intervals and then have shown how DOC can be extended to sort the set of OI as well as of multi-oriented intervals (which can be built from oriented intervals). The proposed comparison method is based on using the available additional information such as starting and finishing points or direction of the intervals to fuzzyfy the comparison measurement and therefore allows us to give the estimation more accurate with more details.

4. Interval Comparison for Search Control

In this section we discuss when and how the reviewed techniques can and should be used.

In order to choose the suitable techniques to a particular problem it is useful to examine the techniques from the view of a search engine. That is we will be interested in the output of the comparison schemes rather than their organization, which has been investigated in the previous section. Focusing on the time spent for comparison: complexity, on how the obtained order reflects the real order: accuracy, and on the type of the intervals (the object or the input of a technique), we have examined the techniques for interval comparison by three characteristics: complexity, accuracy, and interval types. The summary is provided in table 1.

Table 1

Techniques	*Complexity*	*Accuracy*	*Interval Type*
SE	0	/	PI,-
TE	*	///	PI,-
AE	**	///	PI,OI,
PE	****	/////	SI,PI,-
IE	(****)	(////)	CI,PI,-
FE	***	////	SI,OI,PI,

- ✓ FE: for the techniques based on fuzzy measurement which have been described in section 3.2
- ✓ " ( ) " means uncertain because it depends on the kind and the complexity of the intermediate function.
- ✓ "-" means the given technique can also be applied for other kind of intervals but it would not be recommended because then the other available information will not be used.
- ✓ "*" or "/" is an abstract unit, the more "*" the more complex a technique is; the more "/", the more accurate the comparison is.

4.1 When ?

From the analysis in table 1 and the knowledge about comparison mechanisms in section 3, it is obvious the follows.

When we know only about the two bounders a and b and nothing else, but we have to rank the intervals with a low cost, the SE and TE techniques should be used. The searching time then is short. However we have to be satisfied with a low accuracy. AE and FE can also be applied for the given case but they will not be very efficient. This is usually the case of searching for dynamic systems.

When the problems involve real-time systems or concerning control issues, AE and FE (DOS) should be used because they use the additional information such as the direction of intervals, starting and finishing points (which are available in the given kind of systems) to make the interval order be more accurate.

When search is carried out with the fixed parameters and a fixed model (the uncertainty and incompleteness still can take their place!) and the information about the past behavior of the systems is available, we should use PE and FE (VBS,FBS,DBS). While PE could achieve high accuracy by an accurate theoretical judgement if the environment is perfect, FE focus on a practical and flexible comparison method with consideration about the possible difference between theory and practice, the time limitation, and the diversity of the available data.

Finally when the search is for scientific or complex engineering problem, where at the first look, everything is defined and known including the utility function by which a quantity receives its precise value. However because of the time constraints, we can not calculate the utilities for all possible variants when the utility function is too complex. Then we have to estimate the domains of the outcome utilities of groups of variants. These domain are the intervals which we have to rank. Here, the IE, FE, and PE can be used.

4.2 How ?

The next step, after defining the suitable interval comparison techniques, is to implement and embed them in a given search procedure. Usually it is easy to embed because the interval technique just plays the role of a sorting tool. In order to use the techniques we have two choices:

1. Learn the algorithm and implement it by ourselve
2. Use the built-in interval engines as the library functions

In the first case, not all but some algorithms are available in (14). For the second choice, the offers are much more with numbers of software interval packages for free at (12,13,31). However most of them are the implementation of SE and TE. The reason is that the more accurate interval comparison methods usually have large running time and different format for input according to each particular problem. Therefore if we need such kind of interval ranking engines such as PE,IE, and some of FE it is better to implement it together with search engine, the theoretical basis can be obtained in (7,24-30,18).

Summary of when and how to apply interval comparison techniques is provided in table 2.

Table 2

Problem	Techniques	Implementation & Sources
C1: PI	SE,TE	12,13,31,9
C2: OI	AE,FE	1,6,24,25,28-30
C3: SI	PE,FE	7, 24,25,28-30
C4: CI	IE	18

5. A Scheduling Example with Fuzzy Interval Comparison

In this section we examine a solution for a scheduling problem, which use a fuzzy interval comparison mechanism FE (VBS) to rank the concurrent regions in the search space. We also implement two other alternative comparison schemes SE, TE, and compare with FE. The experimental results show us that FE provide more accurate order than the other and therefore increase the utility of the obtained schedule.

The scheduling problem is stated as the follows:

Given:

- N concurrent actions $A_1,A_2,...A_N$ and N is a very large number.
- A set of N utility function F1(t),F2(t),..FN(t) which define the utility that action Ai takes place at t moment time is Fi(t).
- Data base that contains the values of the utilities of $A_1,A_2,...A_N$ at different time moments which are observed in the past.

Question:

Find a scheduler of executing N actions in sequence so that the total utility is maximum.

Solution

This problem falls to the category C4. Therefore, according to 4.1 we will apply the given strategy dividing the search space into regions with utility in intervals as following:

Suppose we have a time series of N virtual time moments: $T_1 < T_2 < ... < T_N$ representing the beginning times for the N actions $A_1,A_2,...A_N$ in the plans. MU[i,j]=Ui(j) is the utility of action A_i that it takes place at time moment Tj , where j=1..N. In this matrix MU[N,N], each cell MU[i,j] is the utility of assigning Ai to Tj and is some interval [a[i,j],b [i,j]]. With such kind of information we need to find the best plans with the maximum total utility.

	T1	< T2	< ..	< TN
A_1	$[a_{1,1},b_{1,1}]$	$[a_{1,2},b_{1,2}]$	[ ]	$[a_{1,N},b_{1,N}]$
A_2	$[a_{2,1},b_{2,1}]$	$[a_{2,2},b_{2,2}]$	[ ]	$[a_{2,N},b_{2,N}]$
...	[...]	[...]	[ ]	[...]
A_N	$[a_{N,1},b_{N,1}]$	$[a_{N,2},b_{N,2}]$	[ ]	$[a_{N,N},b_{N,N}]$

Figure 1. Matrix of Utilities MU

Using the interval arithmetic knowing the form of the given utilities functions, we can calculate the bound of the N virtual time moments T1< T2 < ...< TN . Suppose they are: T1=[0, Δmax]; T2=[Δmin, 2 × Δmax]; ...; TN=[(N-1) × Δmin, N × Δmax]; where Δmin, Δmax are the low and upper bounds of the processing time for an action.

Then, the scheduling problem becomes as the follows:

Given:

- N concurrent actions $A_1, A_2, ... A_N$ and
- a time series of N time moments: $T_1 < T_2 < ... < T_N$
- the utility of assigning Ai to Tj is some interval [a[i,j],b [i,j]] given in each cell MU[i,j] of the matrix MU[N,N].

Question: Find an assignment of N actions to the time series (given not as precise number but in intervals) so that the total utility is maximum.

Now, using interval processing techniques, we can process the incomplete information given in intervals to control the search by leading it to the region with the "best" interval. We choose the FE (VBS) to implement this job.

Simulation and Experimental Results

In order to examine the efficiency of the given interval technique we implement three methods: Greedy (GD), Genetic Algorithm (GA), and our FE(VBS).

The criteria for comparing are:

- Total utility of the obtained schedules
- Scheduling times

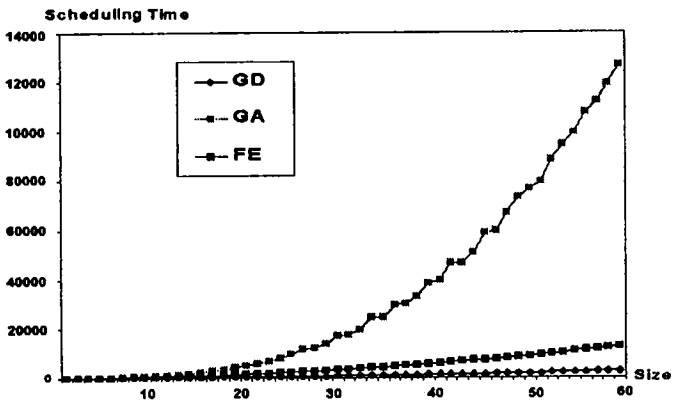

Figure 2

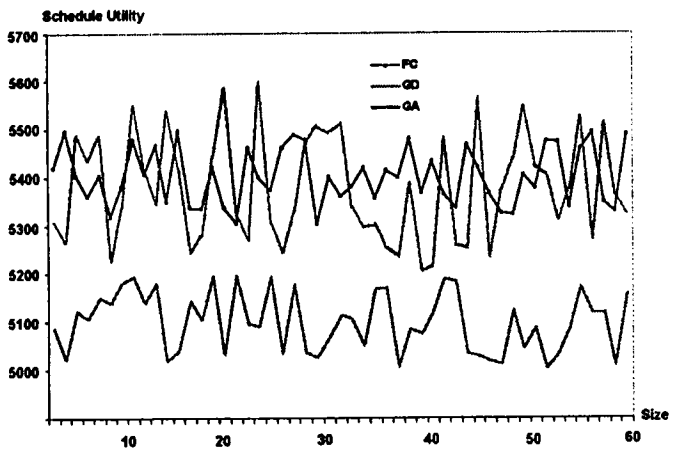

Figure 3

The experimental results in Figures 2,3 show us that FE (VBS) achieves a schedule with better utility while the planning time is more than GD but much less than GA.

6. Discussion

In this paper we have briefly reviewed different interval processing and especially interval comparison techniques. As shown in the analysis and the scheduling example, interval comparison techniques can help us not only to represent the incompleteness and the uncertainty but also improve the search quality and/or the searching time.

Theoretical judgements and experiments also show us that the more we can use the other available information besides the two interval bounders, the better the search result is. However, it is difficult to represent the additional information because the information sources are very different. Another challenge is the time complexity of analyzing this information. In some cases we have to use a specialized hardware implementation instead of software to meet the time constraints (24). Therefore further research direction is to find the ways to deal with these dares.

Interval computation is growing up with number of investigations and applications, they are available at (14). Through this paper we would like to link the interval computation and especially interval comparison into the search community as a considerable issue and helpful techniques for search control under uncertainty and incomplete information.

References

1. Allen J.F. et al. 1991. *Reasoning about Plans.* Morgan Kaufmann.
2. Baas , Kwakernaak , Rating and Ranking of multiple aspect of alternatives using fuzzy sets, Automatica 13 (1), pp47-58, 1977.
3. Bouchon-Meunier B., Nguyen H.T., Kreinovich V., Kosheleva, "Optimization with Soft Constraints: Case of Fuzzy Intervals", Proc. of NAFIP,pp 177-179, 1994.
4. Cesta A. el al.. 1998. Profile-Based Algorithms to solve Multiple Capacitated Metric Scheduling Problems, In *Proceedings of AIPS'98*, 214-223.
5. Clancey, W. J. 1984. Classification Problem Solving. In *Proceedings of the Fourth National Conference on Artificial Intelligence,* 49-54. Menlo Park, Calif.: AAAI Press.
6. Dean T. and Wellman M. 1991. *Planning and Control,* Morgan Kaufmann.
7. Dubois D. , Prade H., "Rangking fuzzy numbers in the Setting of possibility Theory ", Information Sciences 30, pp 183-224, 1983.

8. Geffner H and Bonet B. 1998. High_Level Planning and Control with Incomplete Information Using POMDPs, In *Proceedings of the* Workshop on IPSEDUE, AIPS'98, 113-120.
9. Goodwill R. and Simmons R, 1998. Search Control of Plan Generation in Decision-Theoretic Planners, In *Proceedings of the AIPS'98*, 94-101.
10. Haddawy P. , Kahn Jr C and Doan A., Decision-theoretic refinement planning in medical decision making: Management of Acute deep venous thrombosis, Medical Decision Making.
11. Haddawy P. and Doan A., Abstracting Probabilistic Actions, Proc. of UAI-94, pp270-277.
12. Hales Thomas C., 1998, Univ. of Michigan, "Interval Package. A C++ library especially designed to prove inequalities arising in the Kepler conjecture"., http://www.math.lsa.umich.edu/~hales/countdown
13. Harlequin Group Limited, 1996, "Common Lisp HyperSpec", http://www.harlequin.com/ education /books /HyperSpec/
14. Kearfortt R.B., Interval Computations: Introduction, Uses, and Resources, 1998, http://cs.utep.edu/interval-comp/main.html
15. Knoblock C.A. 1990. Learning abstraction hierarchies for problem solving, In *Proceedings of the AAAI-90*, 923-928.
16. Koenig S. and Simmons R.G. 1998. Solving Robot Navigation Problems with Initial Pose Uncertainty Using Real-Time Heuristic Search; In *Proceedings of the 4th AIPS.*
17. Kohout LJ and Bandler, Fuzzy interval inference utilizing the checklist paradigm and BK-relational products, Application of Interval Computations, Applied Optimization, V3, pp 291-335,1996.
18. Kosheleva O., Kreinovich V., "Decision Making and Game theory in Case of Interval Uncertainty", TR-UTEP-CS-93-4,1993.
19. Kreinovich V, Data processing Beyond Traditional Statistics: Applications of Interval Computations. A Brief Introductions, APIC'95 Proceeding & Journal of Reliable Computing.
20. Levin V.I. and Trejo R., On Comparison of Interval Quantities, TR-Penza Technological Univ., 1998.
21. Miura Jun and Shirai Y., Scheduling Parallel Execution of Planning and action for a Hierachically-Decomposable Planning Problem, In *Proceedings of the Workshop on IPSEDUE, AIPS'98*, 74-80.
22. Nareyaek A. A Planning model for agents in Dynamic and Uncertain Real-Time Environment, Workshop on IPSEDUE, AIPS'98, pp7-14,1998
23. Nguyen T. Hung, Kreinovich V. and Zuo Qiang, "Interval Valued Degrees of Belief: Application of Interval Computations to Expert systems and Intelligent Control", Int J. of Uncertainty, Fuzziness and Knowledge-Based Systems, Vol. 5, N3, pp317-358,1997.
24. Pham H. and Do Q.M., 1999, Adaptive Fuzzy Controller for Predicting Order of Concurrent Actions in Real-time Systems, Tech. Report ISL-01-99-01, Assumption Univ.
25. Pham H.H. and Nguyen Q. A., Fuzzy Complete Continuous Comparison for Planning Actions with Incomplete Information, in Proc. of the Vietnam-Japan Bilateral Symposium on Fuzzy Systems and Applications, pp209-218, September 1998.
26. Pham H.H. and Simonenko V, "Objective-Oriented Algorithm for Job Scheduling in Parallel Heterogeneous Systems " in "Lecture Notes in Computer Science" LNCS 1291 of Springer P., 1997.
27. Pham H.H., " Intelligent Scheduler with Fuzzy Techniques and Evolutionary Management in Heterogeneous Computing Systems" pp. 122-127, Vol 1, in Proc. of IEEE Int. Con. on Intelligent Processing Systems, Beijing, China, October ,1997.
28. Pham H.H., "Fuzzy Comparisons for Interval-Valued Numbers", in Proc. of the 6th Int. Con. on Fuzzy Theory and Technology, North Caroline, USA, October 1998.
29. Pham H.H.. 1998, Sorting Oriented Interval-Valued Numbers with Fuzzy Measurement, Tech. Report ISL-10-98-02, Dept of CS, Assumption Univ.
30. Pham H.H.. 1998. Fuzzy Estimation for Interval-Valued Numbers with Statistical Information, Tech. Report ISL-10-98-01, Dept of CS, Assumption Univ.
31. PROFIL/BIAS Basic Interval Arithmetic Subroutines, VTT, the Technical Research Centre of Finland http://www.vtt.fi/"
32. Sunaga, Theory of an Interval Algebra and its Application to Numerical Analysis, Gaukutsu Bunken Fukeyu-kai, Tokyo, 1958.
33. Volgin L.I and Levin V.I. , Continous Logic : Theory and Applications , Estonian Academy of Sciences, Talin 1991.
34. Yager R. and V. Kreinovich, Decision making under interval probabilities, Technical Report UTEP-CS-98-12, May 1998
35. Yager R. and V. Kreinovich, Fair division under interval uncertainty, Technical Report UTEP-CS-98-13, May 1998.
36. Zhang Weixiong, Flexibale and Approximate Computation through State-Space Reduction, Uncertainty in Artificial Intelligent '98, Morgan Kaufmann, pp531-538, 1998.