5-2006

Growth Rates under Interval Uncertainty

Janos Hajagos

Vladik Kreinovich
The University of Texas at El Paso, vladik@utep.edu

Comments:
UTEP-CS-06-20.
Published in the Proceedings of the *Second International Conference on Fuzzy Sets and Soft Computing in Economics and Finance FSSCEF'2006*, St. Petersburg, Russia, June 28 - July 1, 2006, pp. 31-38.

Growth Rates under Interval Uncertainty

Janos Hajagos[1] and Vladik Kreinovich[2]

[1] Applied Biomathematics,
100 North Country Road
Setauket, New York 11733, USA
logistic@sdf-eu.org
[2] Department of Computer Science,
University of Texas at El Paso,
El Paso, TX 79968, USA
vladik@utep.edu
http://www.cs.utep.edu/vladik

Abstract. For many real-life systems ranging from financial to population-related to medical, dynamics is described by a system of linear equations. For such systems, the growth rate λ can be determined as the largest eigenvalue of the corresponding matrix a_{ij}. In many practical situations, we only know the components of the matrix a_{ij} with interval (or fuzzy) uncertainty. In such situations, it is desirable to find the range of possible values of λ. In this paper, we propose an efficient algorithm for computing λ for a practically important case when all the components a_{ij} of the matrix are non-negative.

1 Growth Rates: A Linear Approximation to the Description of a General System

General description. In general, the state of a real-life complex system can be described by listing the current values of its parameters $x_1, \ldots, x_n$.

- For continuous-time systems, their dynamic can be described as

$$\dot{x}_i = f_i(x_1, \ldots, x_n)$$

  for some functions $f_1, \ldots, f_n$.
- For discrete-time systems, their dynamic can be described as

$$x_i(t+1) = f_i(x_1(t), \ldots, x_n(t))$$

  for some functions $f_1, \ldots, f_n$.

Linearized description. The dependencies $f_i(x_1, \ldots, x_n)$ are usually smooth, so within a reasonable range of values x_i, we can approximate each of these functions by a linear expression:

$$f_i(x_1, \ldots, x_n) = b_i + \sum_{j=1}^{n} a_{ij} \cdot x_j.$$

By applying an appropriate shift $x_i \to x_i - s_i$, we can simplify this system even further, into

$$f_i(x_1, \ldots, x_n) = \sum_{j=1}^{n} a_{ij} \cdot x_j.$$

Thus:

- the dynamic of a continuous-time system can be described by the equation

$$\dot{x}_i = \sum_{j=1}^{n} a_{ij} \cdot x_j;$$

- the dynamic of a discrete-time systems can be described by the equation

$$x_i(t+1) = \sum_{j=1}^{n} a_{ij} \cdot x_j(t).$$

The notion of a growth rate. By using eigenvectors of the matrix $A = (a_{ij})$ as a new base, we get a yet simpler expression for the new variables y_i – the coefficients in the expansion of $x_i(t)$ in this new base.

In the generic case when all eigenvalues are different, the dynamic equations take the simplest possible form:

- for a continuous-time system,

$$\dot{y}_i = \lambda_i \cdot y_i,$$

  where λ_i is the corresponding eigenvalue;
- for a discrete-time system system,

$$y_i(t+1) = \lambda_i \cdot y_i(t).$$

These equations have an explicit solution:

- for a continuous-time system, we get

$$y_i(t) = y_i(0) \cdot \exp(\lambda_i \cdot t);$$

- for a discrete-time system, we get

$$y_i(t) = y_i(0) \cdot \lambda_i^t.$$

This decomposition into simple solutions $y_i(t)$ is one of the main ideas behind the Principle Component Analysis.

A general solution $x_i(t)$ is a linear combination of such terms. Thus, in the general case, asymptotically,

– for a continuous-time system,

$$x_i(t) \sim \exp(\lambda \cdot t),$$

where λ is the largest of these eigenvalues;
– for a discrete-time system,

$$x_i(t) \sim \lambda^t.$$

When the largest eigenvalue is degenerate, we have $x_i(t) \sim x^k \cdot \exp(\lambda \cdot t)$ or $x_i(t) \sim x_k \cdot \lambda^t$ for some integer k, i.e., modulo polynomial terms, still the same asymptotic.

Because of this fact, the largest eigenvalue λ is called the *growth rate* of a system.

This formula indeed describes a growth rate. This asymptotic behavior well describes different types of growth (see, e.g., [3]):

– the population growth,
– the growth in animals and plants,
– the growth rates of number of affected people under an epidemic,
– financial growth,
– etc.

For example, in population growth, different variables x_i describe the number of people of i-th age group, etc.

2 Growth under Interval Uncertainty: A Computational Problem

Computing the growth rate is important. In view of the above applications, it is important to compute the growth rate for a given system.

Idealized case: exactly known coefficients. In general, we never known the *exact* values of parameters of real-life systems, these parameters are always known with some *uncertainty*.

In many real-life situations, however, this uncertainty is small. In such situations, we can safely assume that we know the exact values a_{ij} of all the coefficients. In such situations, we can use known algorithms to find the eigenvalues [3, 7, 14] and thus, find the largest of these eigenvalues – the growth rate.

Often, we cannot ignore the uncertainty. In many real-life situations, however, we cannot ignore the uncertainty. In such situations, we have to take into account the fact that the coefficients a_{ij} are only known with uncertainty.

Case of interval uncertainty. Often, in addition to the approximate values $\widetilde{a}_{ij}$ of the corresponding coefficients, we also know the upper bounds Δ_{ij} on the approximation error $|\widetilde{a}_{ij} - a_{ij}|$. In such situations, we know that the actual (unknown) value of each coefficient a_{ij} belongs to the interval

$$\mathbf{a}_{ij} = [\underline{a}_{ij}, \overline{a}_{ij}] \stackrel{\text{def}}{=} [\widetilde{a}_{ij} - \Delta_{ij}, \widetilde{a}_{ij} + \Delta_{ij}].$$

Computing the growth rate under interval uncertainty: a computational problem. In case of interval uncertainty, different values $a_{ij} \in \mathbf{a}_{ij}$ lead to different growth rates λ. In such situations, it is desirable to find the interval $[\underline{\lambda}, \overline{\lambda}]$ of possible values of λ – or at least an interval that guarantees to contain this interval.

Of special importance is the upper endpoint $\overline{\lambda}$ of the desired interval, because this upper endpoint indicates how fast a population can grow, or how fast a disease can spread.

The need for computing such an interval has been known for a few decades; see, e.g., [2].

Case of small uncertainty: sensitivity analysis. When the uncertainty is relatively small, i.e., when the uncertainty $\Delta a_{ij} \stackrel{\text{def}}{=} \widetilde{a}_{ij} - a_{ij}$ is much smaller than the approximate value $\widetilde{a}_{ij}$, we can linearize the equations for describing the eigenvalues in terms of a_{ij} and use the sensitivity analysis techniques to get reasonable estimates for $[\underline{\lambda}, \overline{\lambda}]$; see, e.g., [2].

General case: the problem is computationally intractable (NP-hard). In many real-life situations, e.g., in many financial and biological systems, the uncertainty is not small, so we can no longer use the linearized techniques to find $\overline{\lambda}$.

In general, we this face a problem of finding the range of possible values of λ for all matrices a_{ij} within a given interval matrix $\mathbf{a}_{ij}$, i.e., for all matrices for which $a_{ij} \in \mathbf{a}_{ij}$. It is known that in general, this problem is NP-hard; see [9] and references therein. This means, crudely speaking, that it is not possible top have an algorithm that would always compute the desired range for λ in physically reasonable time.

Moreover, it is also known [9] that even the problem of computing the eigenvalues with a given accuracy is NP-hard.

This means that while there exist computationally efficient methods of computing an enclosure for the desired interval $[\underline{\lambda}, \overline{\lambda}]$, but these methods sometimes lead to a drastic excess width.

Important case: a non-negative matrix. In many real-life situations, the matrix a_{ij} is *non-negative* in the sense that all its coefficients are non-negative. Such non-negative matrices describe population growth, spread of disease, financial situations, etc. [3].

What we propose. We propose a new algorithm that, for non-negative matrices, exactly computes the upper bound $\overline{\lambda}$ on λ in feasible computation time.

Comment. In this paper, we concentrated on the computation of the largest eigenvalue, a practically useful characteristic of an interval matrix. If, in addition to describing the asymptotic growth rate, we want to find a more detailed description of a growth, then we need to find not only the largest eigenvalue, but also other eigenvalues and the corresponding eigenvectors. Algorithms for solving this problem under interval uncertainty are presented, e.g., in [10].

3 New Algorithm

This algorithm is based on known algorithms for the case of the exact matrix. Our algorithm assumes that already have an algorithm $\mathcal{A}$ for computing the largest eigenvalue $\lambda(A)$ of a given non-negative matrix $A = \|a_{ij}\|$. Such algorithms are described, e.g., in [3, 7, 14].

Input to the new algorithm. Let us assume that instead of the exact non-negative matrix $A = \|a_{ij}\|$, we are given the interval-valued matrix $\mathbf{A} = \|\mathbf{a}_{ij}\|$, where $\mathbf{a}_{ij} = [\underline{a}_{ij}, \overline{a}_{ij}]$.

In other words, for each i and each j, instead of a single value a_{ij}, we have an interval $[\underline{a}_{ij}, \overline{a}_{ij}]$ – i.e., in effect, two values $\underline{a}_{ij}$ and $\overline{a}_{ij}$. We can alternatively describe this situation by saying that:

- for each i and j, we know the lower endpoint $\underline{a}_{ij}$, and
- for each i and j, we know the upper endpoint $\overline{a}_{ij}$.

In other words, instead of a single matrix $A = \|a_{ij}\|$, we are given two matrices: $\underline{A} \stackrel{\text{def}}{=} \|\underline{a}_{ij}\|$ and $\overline{A} \stackrel{\text{def}}{=} \|\overline{a}_{ij}\|$.

Description of the new algorithm.

- First, we apply the algorithm $\mathcal{A}$ for computing $\lambda(A)$ to the matrix $\underline{A}$; the resulting value is returned as $\underline{\lambda}$.
- Then, we apply the algorithm $\mathcal{A}$ for computing $\lambda(A)$ to the matrix $\overline{A}$; the resulting value is returned as $\overline{\lambda}$.

Comment. This simple idea does not work for general interval matrices, only for non-negative ones. In the following text, we provide a proof that for non-negative matrices, this algorithm indeed works well.

Justification of the new algorithm. In order to provide the desired justification, let us introduce some notations. We have defined a matrix A to be non-negative if all its components are non-negative, i.e., as $a_{ij} \geq 0$ for all i and j. It is natural to denote this non-negativity in the usual way, as $A \geq 0$.

We can similarly define a vector $x = (x_1, \ldots, x_n)$ to be non-negative if all its components are non-negative, i.e., if $x_i \geq 0$ for all i. This relation will also be denoted by $x \geq 0$.

We can also define the relation $A \le B$ for two matrices A and B as $B - A \ge 0$. In other words, for matrices $A = \|a_{ij}\|$ and $B = \|b_{ij}\|$, the order $A \le B$ is defined component-wise, as $a_{ij} \le b_{ij}$ for all i and j.

Now, we are ready for the justification. This justification is based on the two known facts (described below in detail):

- the representation of the largest eigenvalue as a solution to an auxiliary optimization problem, and
- the Perron-Frobenius Theorem about the eigenvectors of non-negative matrices.

The first known fact is that the largest eigenvalue $\lambda(A)$ of a matrix A can be described as

$$\lambda(A) = \max_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2},$$

where

$$\|x\|_2 \stackrel{\text{def}}{=} \sqrt{x_1^2 + \ldots + x_n^2}$$

denotes the length of a vector x; see, e.g., [3, 7, 14].

For non-negative matrices A, the Perron-Frobenius Theorem [3, 4, 7, 14] states that at least one of the eigenvectors $x = (x_1, \ldots, x_n)$ corresponding to the largest eigenvalue $\lambda(A)$ is also non-negative: $x \ge 0$. Thus, the maximum in the above definition of $\lambda(A)$ is attained on a non-negative vector. Therefore, when computing this maximum, we can restrict ourselves only to non-negative vectors:

$$\lambda(A) = \max_{x \ge 0 \,\&\, x \neq 0} \frac{\|Ax\|_2}{\|x\|_2}.$$

When $0 \le A$ and $x \ge 0$, then, by definition of the matrix multiplication, we get $Ax \ge 0$. If $0 \le A \le B$ and $x \ge 0$, then similarly $0 \le Ax \le Bx$. When we have two vectors a and b for which $0 \le a_i \le b_i$ for all i, then, of course,

$$a_1^2 + \ldots + a_n^2 \le b_1^2 + \ldots + b_n^2,$$

hence $\|a\|_2 \le \|b\|_2$. Thus, if $0 \le A \le B$, then for every vector $x \ge 0$, we get $\|Ax\|_2 \le \|Bx\|_2$ and therefore,

$$\frac{\|Ax\|_2}{\|x\|_2} \le \frac{\|Bx\|_2}{\|x\|_2}.$$

Since this inequality holds for every vector $x \neq 0$, the maximum $\lambda(A)$ of its left-hand side is smaller than or equal than the maximum $\lambda(B)$ of its right-hand side. In other words, if $0 \le A \le B$, then $\lambda(A) \le \lambda(B)$.

By definition of an interval-valued matrix, all possible matrices $A \in [\underline{A}, \overline{A}]$ satisfy the inequality $\underline{A} \le A \le \overline{A}$. Since we assumed that our matrices are non-negative, we conclude that $0 \le \underline{A} \le A \le \overline{A}$. Thus, for every possible matrix $A \in [\underline{A}, \overline{A}]$, we get $\lambda(\underline{A}) \le \lambda(A) \le \lambda(\overline{A})$.

Hence, all the values $\lambda(A)$ lie within the interval $[\lambda(\underline{A}), \lambda(\overline{A})]$. Since both endpoints of this interval are attained for some matrices from the matrix interval $[\underline{A}, \overline{A}]$, we thus conclude that the interval $[\lambda(\underline{A}), \lambda(\overline{A})]$ is the actual range of $\lambda(A)$. Thus, for non-negative interval matrices, the above algorithm is indeed justified.

Practical applications. In [5, 6], we apply our ideas to the dynamics of real-life ecological systems for which we only know the components a_{ij} with interval uncertainty.

4 Case of Fuzzy Uncertainty

Case of fuzzy uncertainty. Often, knowledge comes in terms of uncertain expert estimates. To describe this uncertainty, for each possible value of a_{ij}, we describe the degree $\mu_{ij}(a_{ij}) \in [0, 1]$ to which this value is possible. These degrees form a *fuzzy set.*

Processing fuzzy uncertainty can be reduced to processing interval uncertainty. For each degree of certainty α, we can determine the set of values of a_{ij} that are possible with at least this degree of certainty – the *α-cut*

$$\{a_{ij} \,|\, \mu_{ij}(a_{ij}) \geq \alpha\}$$

of the original fuzzy set. In most cases, this α-cut is an interval.

Vice versa, if we know α-cuts for every α, then, for each value a_{ij}, we can determine the degree of possibility that a_{ij} belongs to the original fuzzy set [1, 8, 11–13]. A fuzzy set can be thus viewed as a nested family of its α-cuts.

So, if instead of a (crisp) interval $\mathbf{a}_{ij}$ of possible values of the component a_{ij}, we have a fuzzy set $\mu_{ij}(a_{ij})$ of possible values, then we can view this information as a family of nested interval matrices $\mathbf{a}_{ij}(\alpha)$ – α-cuts of the given fuzzy-valued matrices.

Computation under fuzzy uncertainty. Let us consider the case when instead of a (crisp) interval $\mathbf{a}_{ij}$ of possible values of the components, we have a fuzzy set $\mu_{ij}(a_{ij})$ of possible values. In this case, we can view this information as a family of nested interval-valued matrices $\mathbf{a}_{ij}(\alpha)$ – α-cuts of the given fuzzy sets.

Our objective is then to compute the fuzzy number λ – the largest eigenvalue of this fuzzy-valued matrix. In this case, for each level α, to compute the α-cut of this fuzzy number, we can apply interval computations to the α-cuts $\mathbf{a}_{ij}(\alpha)$ of the corresponding fuzzy-valued matrix. The resulting nested intervals form the desired fuzzy set for λ.

So, e.g., if we want to describe 10 different levels of uncertainty, then we must solve 10 interval computation problems – i.e., apply the above algorithm 10 times.

Acknowledgments

This work was supported in part by NASA under cooperative agreement NCC5-209, NSF grants EAR-0225670 and DMS-0532645, Star Award from the University of Texas System, and Texas Department of Transportation grant No. 0-5453.

The authors are thankful to Arnold Neumaier for his valuable advise.

References

1. Bojadziev G., Bojadziev, M.: *Fuzzy Sets, Fuzzy Logic, Applications*, World Scientific, Singapore, 1995.
2. Caswell, H. A.: General formula for the sensitivity of population growth rate to changes in life history parameters. Theoretical Population Biology **14** (1978) 215–230.
3. Caswell, H.: *Matrix Population Models: Construction, Analysis, and Interpretation*, Sinauer Publ., Sunderland, Massachusetts, 2001.
4. Ding, J., Fay, T. H.: The Perron-Frobenius theorem and limits in geometry. American Mathematical Monthly **112**, No. 2 (2005) 171–175.
5. Hajagos, J. G.: Low-dose extrapolation models for reliable human health assessment. In: *Proceedings of the Workshop on Reliable Engineering Computing*, Savannah, Georgia, September 15–17, 2004.
6. Hajagos, J. G.: *Accurately Computing Ecological Risk under Measurement Uncertainty*, Ph.D. Dissertation, Department of Ecology and Evolution, State University of New York at Stony Brook, June 2005.
7. Horn, R. A., Johnson, C. R.: *Matrix Analysis*, Cambridge University Press, Cambridge, 1994.
8. Klir, G., Yuan, B.: *Fuzzy Sets and Fuzzy Logic*, Prentice Hall, New Jersey, 1995.
9. Kreinovich, V., Lakeyev, A., Rohn, J., Kahl, P.: *Computational Complexity and Feasibility of Data Processing and Interval Computations*, Kluwer, Dordrecht, 1998.
10. Lauro, C. N., Palumbo, F.: Principal componwent analysis of interval data: a symbolic data analysis approach. Computational Statistics **15** (2000) 73–87.
11. Moore, R. E., Lodwick, W. A.: Interval analysis and fuzzy set theory. Fuzzy Sets and Systems **135** (2003) 5–9.
12. Nguyen, H. T., Kreinovich, V.: Nested intervals and sets: concepts, relations to fuzzy sets, and applications, In: Kearfott, R. B., Kreinovich, V., Eds.: *Applications of Interval Computations*, Kluwer, Dordrecht, 1996, pp. 245–290.
13. Nguyen, H. T., Walker, E. A.: *First Course in Fuzzy Logic*, CRC Press, Boca Raton, Florida, 2006.
14. Varga, R. S.: *Matrix Iterative Analysis*, Springer Verlag, Berlin, 2000.