Reliable Computing (2006) **12:** 239–243
DOI: 10.1007/s11155-006-7221-8

Solving Overdetermined Systems of Interval Linear Equations *

ELDON HANSEN
654 Paco Drive, Los Altos, CA 94024, USA, e-mail: eldonh@ispwest.com

and

G. WILLIAM WALSTER
Sun Microsystems, Mailstop UMPK16–160, 16 Network Circle, Menlo Park, CA 94025, USA, e-mail: Bill.Walster@sun.com

(Received: 16 September 2002; accepted: 8 July 2005)

Abstract. An algorithm is developed to compute interval bounds on the set of all solutions to an overdetermined system of interval linear equations.

1. Introduction

Given the real $(n \times n)$ matrix $\mathbf{A}$ and the $(n \times 1)$ column vector $\mathbf{b}$, the linear system of equations

$$\mathbf{Ax} = \mathbf{b} \tag{1.1}$$

is consistent if there is a unique $(n \times 1)$ vector $\mathbf{x}$ for which the system in (1.1) is satisfied. If the number of rows in $\mathbf{A}$ and elements in $\mathbf{b}$ is $m \neq n$, then the system is said to be either under- or overdetermined depending on whether $m < n$ or $n < m$. In the overdetermined case, if $m - n$ equations are not linearly dependent, there is no solution vector $\mathbf{x}$ that satisfies the system. Solutions in the underdetermined case are not unique.

In the point (non-interval) case, there is no generally reliable way to decide if an overdetermined system is consistent or not. Instead a least squares approximate solution is generally sought. In the interval case, it is possible to delete inconsistent cases and bound the set of solutions to the remaining consistent equations. In this note, we consider the problem of solving overdetermined systems of equations in which the coefficients are intervals. That is, we consider a system of the form

$$\mathbf{A}^{\mathbf{I}}\mathbf{x} = \mathbf{b}^{\mathbf{I}}, \tag{1.2}$$

where $\mathbf{A}^{\mathbf{I}}$ is an interval matrix of m rows and n columns with $m > n$. The interval vector $\mathbf{b}^{\mathbf{I}}$ has m components. Such a system might arise directly or by linearizing an overdetermined system of nonlinear equations, see [6]. The width of intervals in (1.2) generally represent measurement errors. The goal is to eliminate all inconsistent solutions. Least squares approximate solutions do not eliminate inconsistent solutions and therefore do not solve the present problem. Bentbib [1] provides a least squares approximate solution to overdetermined interval linear systems.

The solution set of (1.2) is the set of vectors $\mathbf{x}$ for which there exists a real matrix $\mathbf{A} \in \mathbf{A}^{\mathbf{I}}$ and a real vector $\mathbf{b} \in \mathbf{b}^{\mathbf{I}}$ such that (1.1) is satisfied. The set of systems of equations defined in (1.2) is inconsistent if its solution set is empty. When the data in $\mathbf{A}^{\mathbf{I}}$ and $\mathbf{b}^{\mathbf{I}}$ are fallible, at least one $\mathbf{A} \in \mathbf{A}^{\mathbf{I}}$ and/or $\mathbf{b} \in \mathbf{b}^{\mathbf{I}}$ will generally exist such that (1.1) is *inconsistent*. Our goal is to implicitly exclude at least some of these cases. For example, the redundancy resulting from the fact that there are more equations than variables might be deliberately introduced to sharpen the interval bound on the set of solutions to (1.1). A simple example of how this can come about is discussed in [5]. In Section 3, we show how this sharpening is accomplished.

We shall simplify the system using Gaussian elimination. In the point case, it is good practice to avoid forming normal equations from the *original* system. Instead one performs elimination using orthogonal operation matrices to triangularize the coefficient matrix. See [4]. After this first phase, the normal equations of this simpler system can be formed and solved. Our procedure begins with a phase similar to the first phase just described. However, we do not quite complete the usual procedure. We have no motivation to use orthogonal operations because we do not form the normal equations. This is just as well because interval orthogonal matrices do not exist.

When using interval Gaussian elimination, it is generally necessary to precondition the system to avoid excessive widening of intervals due to dependence. In Section 2, we show how preconditioning can be done in the present case where $\mathbf{A}^{\mathbf{I}}$ is not square. See [2] for a more thorough discussion of this topic.

2. Preconditioning

We now describe one way to obtain the preconditioning matrix. Let $\mathbf{A}_c$ denote the center of the interval matrix $\mathbf{A}^{\mathbf{I}}$. Note that $\mathbf{A}_c$ can be computed using rounded arithmetic since only an approximation is needed. Again using approximate arithmetic, perform Gaussian elimination to transform $\mathbf{A}_c$ into upper trapezoidal form. Elements in positions (i, j) with $i < j$ should be eliminated when possible. Row pivoting is to be used. Column pivoting can also be used.

The operations to transform $\mathbf{A}_c$ are also performed (approximately) on an identity matrix of order m. The matrix resulting from these operations is the desired preconditioning matrix $\mathbf{B}$. To precondition (1.2) we multiply by $\mathbf{B}$. We obtain

$$\mathbf{M}^{\mathbf{I}}\mathbf{x} = \mathbf{r}^{\mathbf{I}}, \tag{2.1}$$

where $\mathbf{M}^{\mathbf{I}} = \mathbf{B}\mathbf{A}^{\mathbf{I}}$ is an m by n interval matrix and $\mathbf{r}^{\mathbf{I}} = \mathbf{B}\mathbf{b}^{\mathbf{I}}$ is an interval vector of m components. When computing $\mathbf{M}^{\mathbf{I}}$ and $\mathbf{r}^{\mathbf{I}}$, we use interval arithmetic to bound rounding errors.

3. Elimination

We now perform elimination on the preconditioned equations. We apply an interval version of Gaussian elimination to the system $\mathbf{M}^{\mathbf{I}}\mathbf{x} = \mathbf{r}^{\mathbf{I}}$ thereby transforming $\mathbf{M}^{\mathbf{I}}$ into almost (see below) upper trapezoidal form. This procedure only fails when all possible pivot elements contain zero. Note that after preconditioning, no pivot selection is performed during the elimination to obtain a result with the form

$$\begin{bmatrix} \mathbf{T}^{\mathbf{I}} \\ \mathbf{W}^{\mathbf{I}} \end{bmatrix} \mathbf{x} = \begin{bmatrix} \mathbf{u}^{\mathbf{I}} \\ \mathbf{v}^{\mathbf{I}} \end{bmatrix}, \tag{3.1}$$

where $\mathbf{T}^{\mathbf{I}}$ is a square upper triangular interval matrix of order n, and $\mathbf{u}^{\mathbf{I}}$ and $\mathbf{v}^{\mathbf{I}}$ are interval vectors of n and $m-n$ components, respectively. The submatrix $\mathbf{W}^{\mathbf{I}}$ is a matrix of $m-n$ rows and n columns. It is zero except in the last column. Therefore, we can represent it in the form

$$\mathbf{W}^{\mathbf{I}} = [\, \mathbf{0} \;\; \mathbf{z}^{\mathbf{I}} \,],$$

where $\mathbf{0}$ denotes an $m-n$ by $n-1$ block of zeros, and $\mathbf{z}^{\mathbf{I}}$ is a vector of $m-n$ intervals.

We now have a set of equations

$$Z_i x_n = V_i \;(i = 1, \ldots, m-n). \tag{3.2}$$

Also,

$$T_{nn} x_n = U_n. \tag{3.3}$$

Therefore, the unknown value x_n is contained in the interval

$$X_n = \frac{U_n}{T_{nn}} \bigcap_{i=1}^{m-n} \frac{V_i}{Z_i}. \tag{3.4}$$

Taking this intersection is what implicitly eliminates fallible data from $\mathbf{A}^{\mathbf{I}}$ and $\mathbf{b}^{\mathbf{I}}$. It is this operation that allows us to get a sharper bound on the set of solutions to the original system in (1.2) than might otherwise be obtained.

If the original system contains at least one consistent set of equations, the intersection in (3.4) must not be empty.

Knowing X_n, we can backsolve (3.1) for $X_{n-1}, \ldots, X_1$. From (3.1), this takes the standard form of backsolving a triangular system $\mathbf{T}^{\mathbf{I}}\mathbf{x} = \mathbf{u}^{\mathbf{I}}$. Sharpening X_n using (3.4) also produces sharper bounds $\mathbf{x}^{\mathbf{I}}$ on other components of $\mathbf{x}$ when we backsolve.

Rohn [3] provides an iterative method for bounding the solution set. With sufficient effort, the optimal bounding set can be computed. However, because of the required computing Rohn recommends his procedure only for moderately sized problems. Our (much faster) procedure will generally not sharply bound the optimal solution because of interval width caused by dependence.

4. Inconsistency

Suppose the initial equations (1.2) are inconsistent. Then equations (3.2) and (3.3) might or might not be consistent. Interval widening due to dependence and roundoff can cause the intersection in (3.4) to be non-empty.

Nevertheless, suppose we find that the intersection in (3.4) is empty. This event *proves* that the original equations in (1.2) are inconsistent. Proving inconsistency might be the signal that a theory is measurably false, which might be an extremely enlightening event. On the other hand, inconsistency might mean no more than invalid measurements have been made. These possibilities are discussed in [5]. If invalid measurements are suspected, it might be important to discover which equation(s) in (1.2) are inconsistent. We might know which equation(s) in the transformed system (3.1) must be eliminated to obtain consistency. However, an equation in (3.1) is generally a linear combination of all the original equations in (1.2). Therefore, to establish consistency in the original system, we generally cannot determine which of its equation(s) to remove.

We might be able to determine a likely removal candidate by using the following steps:

1. Remove enough equations from (3.1) that the intersection in (3.4) is not empty.
2. Solve the modified version of (3.1) for $X_{n-1}, \ldots, X_1$. This process cannot fail if the elimination process to obtain (3.1) does not fail.
3. Substitute the solution into the original system (1.2). Any equation in (1.2) is a removal candidate if its left and right members do not intersect.

Acknowledgement

The authors are grateful to George Corliss for his constructive comments and suggestions.

References

1. Bentbib, A. H.: Solving the Full Rank Interval Least Squares Problem, *Applied Numerical Mathematics* **41** (2002), pp. 283–294.
2. Hansen, E. R. and Walster, G. W.: *Global Optimization Using Interval Analysis—Second Edition, Revised and Expanded*, Marcel Dekker, Inc., New York—Basel, 2004.
3. Rohn, J.: Enclosing Solutions of Overdetermined Systems of Linear Interval Equations, *Reliable Computing* **2** (2) (1996), pp. 167–171.
4. Stewart, G. W.: *Introduction to Matrix Computations*, Academic Press, 1973.

5. Walster, G. W.: Philosophy and Practicalities of Interval Arithmetic, in: Moore, R. E. (ed.), *Reliability in Computing*, Academic Press, Inc., San Diego, 1988, pp. 309–323.
6. Walster, G. W. and Hansen, E. R.: Computing Interval Parameter Bounds from Fallible Measurements Using Overdetermined (Tall) Systems of Nonlinear Equations, in: Bliek, Ch., Jermann, Ch., and Neumaier, A. (eds), *Global Optimization and Constraint Satisfaction, First International Workshop Global Constraint Optimization and Constraint Satisfaction, COCOS 2002, Valbonne-Sophia Antipolis, France, October 2–4, 2002, Revised Selected Papers*, Springer, 2003, pp. 171–177.