

Сравнительный анализ реализаций модификации Рона в методах дробления параметров

Д. Ю. ЛЮДВИН, С. П. ШАРЫЙ

Институт вычислительных технологий СО РАН, Новосибирск, Россия

e-mail: lyudvin@ngs.ru, shary@ict.nsc.ru

Работа посвящена сравнению различных реализаций методов дробления параметров для оптимального внешнего оценивания множеств решений интервальных линейных систем уравнений. Главное внимание уделено анализу модификации на основе методики Рона как наиболее сложной, трудозатратной, но и наиболее эффективной. Представлены результаты вычислительных экспериментов с различными способами организации вычислений, их анализ и обсуждение. Даны практические рекомендации по оптимизации вычислительных схем методов дробления параметров.

Ключевые слова: интервал, интервальный анализ, интервальная система линейных алгебраических уравнений, метод дробления параметров, модификация Рона.

Введение и постановка задачи

Рассматривается задача внешнего интервального оценивания объединенного множества решений интервальных систем линейных алгебраических уравнений (ИСЛАУ) вида

$$\begin{cases} \mathbf{a}_{11}x_1 + \mathbf{a}_{12}x_2 + \dots + \mathbf{a}_{1n}x_n = \mathbf{b}_1, \\ \mathbf{a}_{21}x_1 + \mathbf{a}_{22}x_2 + \dots + \mathbf{a}_{2n}x_n = \mathbf{b}_2, \\ \vdots \quad \quad \quad \ddots \quad \quad \quad \vdots \quad \quad \quad \vdots \\ \mathbf{a}_{n1}x_1 + \mathbf{a}_{n2}x_2 + \dots + \mathbf{a}_{nn}x_n = \mathbf{b}_n \end{cases} \quad (1)$$

с интервальными коэффициентами \mathbf{a}_{ij} и интервальными правыми частями \mathbf{b}_i , $i, j = 1, 2, \dots, n$, или, кратко,

$$\mathbf{A}x = \mathbf{b}, \quad (2)$$

где $\mathbf{A} = (\mathbf{a}_{ij})$ — интервальная $(n \times n)$ -матрица, $\mathbf{b} = (\mathbf{b}_i)$ — интервальный n -вектор (брус). Системы (1)–(2) представляются как семейства обычных точечных (неинтервальных) линейных систем $Ax = b$ той же структуры с матрицами $A \in \mathbf{A}$ и векторами $b \in \mathbf{b}$.

Объединенным множеством решений интервальной линейной системы уравнений будем называть множество

$$\Xi(\mathbf{A}, \mathbf{b}) = \{x \in \mathbb{R}^n \mid (\exists A \in \mathbf{A})(\exists b \in \mathbf{b})(Ax = b)\}, \quad (3)$$

образованное всевозможными решениями точечных систем $Ax = b$ с $A \in \mathbf{A}$, $b \in \mathbf{b}$. Для интервальных систем уравнений существуют также другие множества решений, соответствующие тем или иным практическим ситуациям. Поскольку в настоящей работе они не рассматриваются, то далее (3) будем называть множеством решений.

Далее интервальная матрица \mathbf{A} предполагается неособенной, т. е. содержащей только неособенные (невырожденные) точечные матрицы. Тогда множество решений $\Xi(\mathbf{A}, \mathbf{b})$ системы (1)–(2) ограничено. Нас будет интересовать задача оптимального внешнего интервального оценивания этого множества решений:

Найти интервальный вектор $\mathbf{U} \subset \mathbb{R}^n$, имеющий наименьшую возможную ширину и содержащий множество решений $\Xi(\mathbf{A}, \mathbf{b})$ интервальной системы уравнений $\mathbf{A}x = \mathbf{b}$.

(4)

Вычисление для множества решений внешних покоординатных оценок с заданной точностью является NP-трудной задачей [1–3]. В силу ее труднорешаемости соответствующие численные методы являются экспоненциально трудными и по своей структуре близки переборным алгоритмам дискретной оптимизации.

В работе рассматривается класс эффективных вычислительных алгоритмов для нахождения решений задачи (4), называемый *методами дробления параметров* (или PPS-методами¹) [4–6]. Их основная идея — представить “внешнюю задачу” как оптимизационную и применить для ее решения интервальные методы глобальной оптимизации. Задача поиска нижних оценок множества решений ИСЛАУ эквивалентна задаче глобальной оптимизации величины

$$\min\{x_\nu \mid Ax = b\}$$

как функции параметров $A \in \mathbf{A}$ и $b \in \mathbf{b}$.

Задачу поиска нижних оценок множества решений ИСЛАУ можно переформулировать в виде

Найти $\min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$, $\nu = 1, 2, \dots, n$, либо как можно более точные оценки снизу.

(5)

При поиске внешних оценок множества решений ИСЛАУ будем учитывать, что

$$\max\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\} = -\min\{x_\nu \mid x \in \Xi(\mathbf{A}, -\mathbf{b})\}.$$

Метод дробления параметров заключается в последовательном улучшении оценки $\min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$ посредством дробления системы $\mathbf{A}x = \mathbf{b}$ на две ИСЛАУ-потомка, получающиеся из этой системы расщеплением на концы одного интервального элемента либо в матрице \mathbf{A} , либо в векторе \mathbf{b} . Это возможно в силу теоремы Бека—Никеля, согласно которой точные значения $\min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$ и $\max\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$, $\nu = 1, 2, \dots, n$, достигаются на решениях точечных систем уравнений $Ax = b$ таких, что матрица A и вектор b образованы концами интервальных элементов из \mathbf{A} и \mathbf{b} . Итерационная процедура улучшения оценки строится в соответствии с хорошо известным методом “ветвей и границ”.

Результат теоремы Бека—Никеля был значительно усилен И. Роном [7], уточнившим множество вершин матрицы $A \in \mathbf{A}$ и вектора $b \in \mathbf{b}$, на которых достигаются минимальное и максимальное значения компонент точек из множества $\Xi(\mathbf{A}, \mathbf{b})$ решений интервальной системы $\mathbf{A}x = \mathbf{b}$.

Цель данной работы — реализация различных версий методов дробления параметров, в частности, с использованием модификации Рона, их апробация на тестовых примерах и сравнительный анализ полученных результатов.

¹PPS — Partitioning Parameter Set.

1. Метод дробления параметров для интервальных линейных систем уравнений

Существует ряд интервальных методов (например, интервальный метод Гаусса, интервальный метод Гаусса — Зейделя, метод Кравчика и т. д.), позволяющих получить внешнюю интервальную оценку множества $\Xi(\mathbf{A}, \mathbf{b})$. Данные методы позволяют вычислить интервальный вектор, гарантированно содержащий множество решений ИСЛАУ, но не обеспечивают его оптимальность.

Обозначим через $Encl$ какой-нибудь фиксированный метод внешнего оценивания множества решений ИСЛАУ и далее будем называть его *базовым*. Пусть $Encl(\mathbf{A}, \mathbf{b}) \in \mathbb{IR}^n$ — интервальный вектор, получаемый с помощью этого метода, т. е.

$$Encl(\mathbf{A}, \mathbf{b}) \supseteq \Xi(\mathbf{A}, \mathbf{b}).$$

Пусть $\Upsilon(\mathbf{A}, \mathbf{b})$ — нижний конец ν -й компоненты ($\nu = 1, 2, \dots, n$) внешней интервальной оценки множества решений, получаемой методом $Encl$, т. е. $\Upsilon(\mathbf{A}, \mathbf{b}) := (Encl(\mathbf{A}, \mathbf{b}))_\nu$. Основа PPS-методов — адаптивное дробление области параметров интервальной системы. В случае решения ИСЛАУ (1)–(2) процесс измельчения элементов матрицы \mathbf{A} и вектора правых частей \mathbf{b} можно значительно упростить, если учесть следующий результат.

Теорема Бека — Никеля [8, 9]. *Для любого $\nu \in \{1, 2, \dots, n\}$ точные по координатным оценки точек из объединенного множества решений — экстремальные значения $\min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$ и $\max\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$ — достигаются на решениях крайних точечных систем уравнений $Ax = b$, т. е. таких, что матрица A и вектор b образованы концами интервальных элементов из \mathbf{A} и \mathbf{b} соответственно.*

Потребуем от базового метода $Encl$ удовлетворения условию: оценка $\Upsilon(\mathbf{A}, \mathbf{b})$ монотонна по включению относительно матрицы \mathbf{A} и вектора \mathbf{b} , т. е. для всех $\mathbf{A}', \mathbf{A}'' \in \mathbb{IR}^{n \times n}$ и $\mathbf{b}', \mathbf{b}'' \in \mathbb{IR}^n$ при $\mathbf{A}' \subseteq \mathbf{A}''$ и $\mathbf{b}' \subseteq \mathbf{b}''$ верно неравенство

$$\Upsilon(\mathbf{A}', \mathbf{b}') \geq \Upsilon(\mathbf{A}'', \mathbf{b}''). \quad (6)$$

В дальнейшем будем обозначать ИСЛАУ-потомки, получающиеся из интервальной системы $\mathbf{Q}x = \mathbf{r}$ расчлениением на концы одного интервального элемента либо в матрице \mathbf{Q} , либо в векторе \mathbf{r} , через $\mathbf{Q}'x = \mathbf{r}'$ и $\mathbf{Q}''x = \mathbf{r}''$. Здесь \mathbf{Q}' и \mathbf{Q}'' — матрицы, полученные из \mathbf{Q} заменой элемента q_{ij} на \underline{q}_{ij} и \bar{q}_{ij} соответственно, а \mathbf{r}' и \mathbf{r}'' — векторы, полученные из \mathbf{r} заменой элемента r_i на \underline{r}_i и \bar{r}_i соответственно.

Решив две интервальных “системы-потомка” $\mathbf{Q}'x = \mathbf{r}'$ и $\mathbf{Q}''x = \mathbf{r}''$, в силу свойства (6) можно получить более точную оценку снизу для искомого $\min\{x_\nu \mid x \in \Xi(\mathbf{Q}, \mathbf{r})\}$ в виде

$$\min\{\Upsilon(\mathbf{Q}', \mathbf{r}'), \Upsilon(\mathbf{Q}'', \mathbf{r}'')\}.$$

Доказательство данного утверждения имеется в [4]. Аналогичный эффект дает и расчленение в векторе правых частей \mathbf{r} какого-нибудь интервального элемента r_i на концы \underline{r}_i и \bar{r}_i .

Процедуру улучшения оценки для $\min\{x_\nu \mid x \in \Xi(\mathbf{Q}, \mathbf{r})\}$ посредством дробления параметров можно повторить по отношению к системам-потомкам $\mathbf{Q}'x = \mathbf{r}'$ и $\mathbf{Q}''x = \mathbf{r}''$, затем снова разбить потомков и улучшить оценку и т. д. Данную процедуру можно оформить в соответствии с методом “ветвей и границ”.

2. Модификация Рона в методе дробления параметров

Известно, что объединенное множество решений интервальной линейной алгебраической системы имеет следующую элегантную характеристику Оеттли—Прагера.

Теорема Оеттли—Прагера [10]. *Точка $x \in \mathbb{R}^n$ принадлежит множеству $\Xi(\mathbf{A}, \mathbf{b})$ решений системы $\mathbf{A}x = \mathbf{b}$ тогда и только тогда, когда*

$$|(\text{mid } \mathbf{A})x - \text{mid } \mathbf{b}| \leq \text{rad } \mathbf{A} \cdot |x| + \text{rad } \mathbf{b}. \quad (7)$$

Суть теоремы Оеттли—Прагера состоит в том, что множество решений $\Xi(\mathbf{A}, \mathbf{b})$ описывается простым векторным неравенством (7). Значения x , обращающие (7) в равенство, будем называть *экстремальными решениями* неравенства Оеттли—Прагера.

Обозначим через \mathcal{E} множество n -векторов с компонентами ± 1 . Очевидно, что множество \mathcal{E} имеет мощность 2^n . Для заданных векторов $\sigma, \tau \in \mathcal{E}$ определим матрицы $T_\sigma, A^{\sigma\tau} = \{a_{ij}^{\sigma\tau}\}$ и вектор $b^\sigma = \{b_i^\sigma\}$ следующим образом:

$$T_\sigma = \text{diag } \{\sigma_1, \dots, \sigma_n\},$$

$$A^{\sigma\tau} = \text{mid } \mathbf{A} - T_\sigma \cdot \text{rad } \mathbf{A} \cdot T_\tau, \quad b^\sigma = \text{mid } \mathbf{b} + T_\sigma \cdot \text{rad } \mathbf{b}.$$

Из этого следует, что

$$a_{ij}^{\sigma\tau} = \begin{cases} \bar{a}_{ij}, & \text{если } \sigma_i \tau_j = -1, \\ \underline{a}_{ij}, & \text{если } \sigma_i \tau_j = 1, \end{cases} \quad b_i^\sigma = \begin{cases} \bar{b}_i, & \text{если } \sigma_i = 1, \\ \underline{b}_i, & \text{если } \sigma_i = -1. \end{cases} \quad (8)$$

Теорема Рона об экстремальных решениях [11]. *Пусть $(n \times n)$ -матрица \mathbf{A} неособенна и \mathbf{b} — интервальный n -вектор. Тогда для каждого $\sigma \in \mathcal{E}$ уравнение*

$$\text{mid } \mathbf{A} \cdot x - T_\sigma \cdot \text{rad } \mathbf{A} \cdot |x| = b^\sigma \quad (9)$$

имеет единственное решение x^σ , принадлежащее $\Xi(\mathbf{A}, \mathbf{b})$, и справедливо равенство

$$\text{conv } \Xi(\mathbf{A}, \mathbf{b}) = \text{conv } \{x^\sigma \mid \sigma \in \mathcal{E}\}, \quad (10)$$

где conv — выпуклая оболочка.

Заметим, что вследствие (8) матрица $A^{\sigma\tau}$ и вектор b^σ образованы наборами концов элементов \mathbf{A} и \mathbf{b} соответственно, поэтому система (9) является крайней точечной системой для исходной интервальной системы $\mathbf{A}x = \mathbf{b}$. Для неособенной матрицы \mathbf{A} экстремальные решения могут достигаться лишь на множестве 4^n матриц $A^{\sigma\tau}$ и ассоциированных с ними векторов b^σ , т. е.

$$\min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\} = \min_{\sigma, \tau \in \mathcal{E}} ((A^{\sigma\tau})^{-1} b^\sigma)_\nu, \quad (11)$$

$$\max\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\} = \max_{\sigma, \tau \in \mathcal{E}} ((A^{\sigma\tau})^{-1} b^\sigma)_\nu \quad (12)$$

для каждого $\nu = 1, 2, \dots, n$.

Описанный в разделе 1 метод дробления параметров можно модифицировать (см. [4]), если использовать результаты И. Рона, уточнившие множество вершин матрицы \mathbf{A}

и вектора \mathbf{b} , на которых достигаются оптимальные внешние оценки множества решений ИСЛАУ.

Из (11) и (12) следует, что в процессе дробления исходной ИСЛАУ необходимо отслеживать концы задействованных интервальных элементов матрицы системы и вектора правых частей. Для этого с каждой интервальной линейной системой $\mathbf{Q}x = \mathbf{r}$ мы связываем

- 1) вспомогательную целочисленную $(n \times n)$ -матрицу $W = \{w_{ij}\}$, элементы которой равны ± 1 или 0, такую, что

$$w_{ij} = \begin{cases} -1, & \text{если } \mathbf{q}_{ij} = \bar{\mathbf{a}}_{ij}, \\ 0, & \text{если } \mathbf{q}_{ij} = \mathbf{a}_{ij}, \\ 1, & \text{если } \mathbf{q}_{ij} = \underline{\mathbf{a}}_{ij}, \end{cases} \quad (13)$$

- 2) вспомогательные целочисленные n -векторы $s = \{s_i\}$ и $t = \{t_j\}$ с компонентами ± 1 или 0 такие, что

$$w_{ij} = s_i t_j \quad (14)$$

для всех $i, j = 1, 2, \dots, n$ и

$$s_i = \begin{cases} -1, & \text{если } \mathbf{r}_i = \underline{\mathbf{b}}_i, \\ 0, & \text{если } \mathbf{r}_i = \mathbf{b}_i, \\ 1, & \text{если } \mathbf{r}_i = \bar{\mathbf{b}}_i. \end{cases} \quad (15)$$

Матрицу W и векторы s, t будем называть контрольными. Значения t_j определяются через матрицу W и вектор s посредством соотношения (14). Контрольные векторы s и t являются “приближениями” к векторам σ и τ , входящим в равенства (11) и (12), а контрольная матрица $W = st^T$ является “приближением” к матрице $(\sigma \tau^T)$. В начале работы алгоритма элементы W, s и t инициализируем нулями, далее в процессе дробления исходной ИСЛАУ они переычисляются.

Возникающие в процессе дробления системы-потомки организуем в список \mathcal{L} , состоящий из записей вида

$$(\mathbf{Q}, \mathbf{r}, \Upsilon(\mathbf{Q}, \mathbf{r}), W, s, t, \mathbf{Y}, \mathbf{x}), \quad (16)$$

где \mathbf{Q} — интервальная $(n \times n)$ -матрица, $\mathbf{Q} \subseteq \mathbf{A}$; \mathbf{r} — интервальный n -вектор, $\mathbf{r} \subseteq \mathbf{b}$; $\Upsilon(\mathbf{Q}, \mathbf{r})$ — нижний конец ν -й компоненты внешней интервальной оценки множества решений $\Xi(\mathbf{Q}, \mathbf{r})$; W, s, t — контрольные матрица и векторы; \mathbf{Y} — обратная интервальная матрица такая, что $\mathbf{Y} \supseteq \{\mathbf{Q}^{-1} \mid \mathbf{Q} \in \mathbf{Q}\}$; \mathbf{x} — интервальный n -вектор такой, что $\mathbf{x} \supseteq \Xi(\mathbf{Q}, \mathbf{r})$.

Образующие список \mathcal{L} записи упорядочим по возрастанию значений оценки $\Upsilon(\mathbf{Q}, \mathbf{r})$. Первую запись списка, а также соответствующую ИСЛАУ $\mathbf{Q}x = \mathbf{r}$ и наименьшую в списке оценку $\Upsilon(\mathbf{Q}, \mathbf{r})$ будем называть *ведущими* на данном шаге.

На каждом шаге алгоритма с учетом значений контрольных матрицы W и вектора s выполняется процедура дробления интервального элемента \mathbf{h} ведущей ИСЛАУ $\mathbf{Q}x = \mathbf{r}$. Если \mathbf{h} есть \mathbf{q}_{kl} из матрицы \mathbf{Q} (\mathbf{r}_k из вектора правой части \mathbf{r}), то в случае $w_{kl} = 0$ ($s_k = 0$) порождаются две системы-потомка $\mathbf{Q}'x = \mathbf{r}'$ и $\mathbf{Q}''x = \mathbf{r}''$. Если $w_{kl} = \pm 1$ ($s_k = \pm 1$), то оставляем в рабочем списке только одного потомка (какого именно, зависит от знака элемента w_{kl} или s_k).

Заметим, что в процедуре дробления отбрасываются лишь те системы, которые либо не принадлежат множеству точечных систем $\{A^{\sigma\tau}x = b^\sigma \mid \sigma, \tau \in \mathcal{E}\}$, либо не содержат такие системы. Поэтому отбрасывание систем-потомков не нарушает свойство ведущих оценок $\Upsilon(\mathbf{Q}, \mathbf{r})$ приближать искомым $\min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$ снизу (предполагается, что оценка $\Upsilon(\mathbf{Q}, \mathbf{r})$ монотонна по включению относительно матрицы системы и вектора правых частей).

После дробления ведущей ИСЛАУ необходимо вычислить контрольные матрицы и векторы для систем-потомков. Пусть в результате дробления порождаются две системы-потомка: $\mathbf{Q}'x = \mathbf{r}'$ и $\mathbf{Q}''x = \mathbf{r}''$. Обозначим соответствующие им новые контрольные матрицы через W' и W'' , а пары новых контрольных векторов — через s', t' и s'', t'' . Новые контрольные матрицы и векторы вычислим следующим образом:

1) присваиваем $W'' \leftarrow W' \leftarrow W$, $s'' \leftarrow s' \leftarrow s$, $t'' \leftarrow t' \leftarrow t$;

2) a — если рассеченным элементом был \mathbf{q}_{kl} из матрицы \mathbf{Q} , то присваиваем $w'_{kl} \leftarrow 1$ и $w''_{kl} \leftarrow -1$; b — если рассеченным элементом был \mathbf{r}_k из вектора \mathbf{r} , то присваиваем $s'_k \leftarrow -1$ и $s''_k \leftarrow 1$;

3) если хотя бы один из объектов семейства (W', s', t') изменился, перевычисляем остальные два, используя соотношение (14); эту процедуру повторяем, пока изменения не прекратятся; аналогично поступаем с (W'', s'', t'') .

Если ведущая система порождает одного потомка $\mathbf{Q}'x = \mathbf{r}'$, то контрольные матрица и векторы остаются неизменными, поэтому присваиваем $W' \leftarrow W$, $s' \leftarrow s$, $t' \leftarrow t$.

Отметим одно замечательное свойство контрольной матрицы W , вытекающее из соотношения (14). В каждой ее (2×2) -подматрице любой элемент равен произведению остальных трех элементов

$$w_{i_1j_1}w_{i_1j_2}w_{i_2j_2} = w_{i_2j_1}, \quad (17)$$

$$w_{i_1j_2}w_{i_2j_1}w_{i_2j_2} = w_{i_1j_1}, \quad (18)$$

$$w_{i_1j_1}w_{i_2j_1}w_{i_2j_2} = w_{i_1j_2}. \quad (19)$$

Данные равенства могут быть полезными для уточнения контрольной матрицы W . Например, пусть мы намереваемся рассечь ведущую интервальную систему $\mathbf{Q}x = \mathbf{r}$ по элементу \mathbf{q}_{kl} , в то время как соответствующий элемент w_{kl} матрицы W равен нулю. При этом согласно обычному правилу дробления необходимо породить две системы-потомка. Но логично сделать попытку доопределить w_{kl} , поискав в W (2×2) -подматрицы, имеющие ненулевыми все элементы за исключением w_{kl} . Если такая подматрица найдется, то присваиваем элементу w_{kl} значение произведения остальных трех элементов.

Заметим, что запись $(\mathbf{Q}, \mathbf{r}, \Upsilon(\mathbf{Q}, \mathbf{r}), W, s, t, \mathbf{Y}, \mathbf{x})$, удовлетворяющая на некотором шаге условию

$$\Upsilon(\mathbf{Q}, \mathbf{r}) > \omega,$$

при дальнейшем выполнении алгоритма уже никогда не станет ведущей, поэтому либо вообще не вносится в список, либо удаляется при чистке списка, которая проводится после изменения (уменьшения) параметра ω .

На каждом шаге алгоритма проверяем монотонную зависимость решения системы от ее коэффициентов в соответствии с процедурой, описанной в [4–6]. На основе соотношений (13), (15) изменяем с 0 на ± 1 значения элементов контрольных матрицы W и вектора s , соответствующие тем элементам в \mathbf{Q} и \mathbf{r} , на которых была выявлена монотонность.

В целях обеспечения наиболее быстрой сходимости алгоритма рекомендуется рассекать ведущие ИСЛАУ по элементам, на которых достигается максимум величин

$$\left| \frac{\partial x_\nu(\mathbf{Q}, \mathbf{r})}{\partial q_{ij}} \right| \cdot \text{wid } q_{ij}, \quad \left| \frac{\partial x_\nu(\mathbf{Q}, \mathbf{r})}{\partial r_i} \right| \cdot \text{wid } r_i, \quad i, j \in \{1, 2, \dots, n\}, \quad (20)$$

т. е. максимум произведения модуля интервальной оценки производной решения на ширину соответствующего интервала.

Достаточным условием оптимальности вычисленной оценки $\Upsilon(\mathbf{Q}, \mathbf{r})$, т. е. ее совпадения с $\min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$, является выполнение “условия точности”: $\Upsilon(\mathbf{Q}, \mathbf{r}) = (Q^{-1}r)_\nu$ для всех $Q \in \mathbb{R}^{n \times n}$ и $r \in \mathbb{R}^n$.

Наконец, опишем механизм контроля точности ведущих оценок, порождаемых алгоритмом. Наряду с оценкой $\Upsilon(\mathbf{Q}, \mathbf{r})$ для интервальных систем $\mathbf{Q}x = \mathbf{r}$ вычисляются также и величины $\Upsilon(\text{mid } \mathbf{Q}, \text{mid } \mathbf{r})$. Очевидно, что

$$\Upsilon(\text{mid } \mathbf{Q}, \text{mid } \mathbf{r}) \geq \Upsilon(\mathbf{Q}, \mathbf{r}),$$

и значения $\Upsilon(\text{mid } \mathbf{Q}, \text{mid } \mathbf{r})$ приближают $\min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$ сверху: если для каждого шага алгоритма определять

$$\omega = \min \Upsilon(\text{mid } \mathbf{Q}, \text{mid } \mathbf{r}) \quad (21)$$

по всем интервальным линейным системам $\mathbf{Q}x = \mathbf{r}$, когда-либо побывавшим в списке \mathcal{L} до этого шага, то

$$\min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\} \leq \omega.$$

С другой стороны, если $\mathbf{Q}x = \mathbf{r}$ — ведущая ИСЛАУ, то

$$\Upsilon(\mathbf{Q}, \mathbf{r}) \leq \min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\},$$

и потому одним из критериев остановки алгоритма может быть достижение требуемой малости величины $(\omega - \Upsilon(\mathbf{Q}, \mathbf{r}))$.

Итоговая схема алгоритма метода дробления параметров с модификацией Рона приведена ниже (см. с. 76).

3. Реализация алгоритмов и численные эксперименты

Представленный в предыдущем разделе алгоритм был реализован в программной системе МАТЛАВ, дополненной пакетом интервального расширения INTLAB 5.5. Помимо основной схемы алгоритма внешнего оценивания множества решений ИСЛАУ методом дробления параметров с модификацией Рона были реализованы его модификации, различающиеся базовым алгоритмом *Encl*, а также способом организации и обработки списка. Базовый алгоритм *Encl* реализован на основе

- метода Кравчика [12],
- модифицированного метода Кравчика с использованием “эпсилон-раздутия” [13, 14],
- интервального метода Гаусса [4, 15],
- интервального метода Гаусса — Зейделя [4],
- процедуры Хансена — Блика — Рона (изложена в [11, 16]).

Алгоритм оптимального внешнего оценивания множества решений ИСЛАУ методом дробления параметров с модификацией Рона

```

DO WHILE ((ведущая оценка  $\Upsilon(\mathbf{Q}, \mathbf{r})$  неточна) и  $(\omega - \Upsilon(\mathbf{Q}, \mathbf{r}) > \varepsilon)$ )
    вычисляем интервальные расширения производных  $\frac{\partial x_\nu(\mathbf{Q}, \mathbf{r})}{\partial q_{ij}}$  и
         $\frac{\partial x_\nu(\mathbf{Q}, \mathbf{r})}{\partial r_i}$  по элементам  $q_{ij}$  и  $r_i$  с ненулевой шириной;
    сжимаем в  $\mathbf{Q}$  и  $\mathbf{r}$  элементы, на которых выявлена монотонная
        зависимость  $x_\nu$  от  $q_{ij}$  и  $r_i$ ;
    соответствующие элементы в контрольных матрице  $W$  и векторе  $s$ 
        изменяем с 0 на  $\pm 1$  на основе (13), (15);
    ищем в ведущей ИСЛАУ  $\mathbf{Q}x = \mathbf{r}$  интервальный элемент  $h$ ,
        которому соответствует наибольшее из произведений (20);
    пытаемся уточнить матрицу  $W$  на основе равенств (17)–(19);
    порождаем одну или две ИСЛАУ-потомка  $\mathbf{Q}'x = \mathbf{r}'$  и  $\mathbf{Q}''x = \mathbf{r}''$ ;
    если порождены две системы-потомка, перевычисляем матрицы
         $W', W''$  и векторы  $s', s'', t', t''$  согласно (14),
        иначе присваиваем  $W' \leftarrow W, s' \leftarrow s, t' \leftarrow t$ ;
    вычисляем вектор  $\mathbf{x}' \leftarrow \text{Encl}(\mathbf{Q}', \mathbf{r}')$  и, возможно,  $\mathbf{x}'' \leftarrow \text{Encl}(\mathbf{Q}'', \mathbf{r}'')$ ;
    присваиваем оценку  $v' \leftarrow \Upsilon(\mathbf{Q}', \mathbf{r}')$  и, возможно,  $v'' \leftarrow \Upsilon(\mathbf{Q}'', \mathbf{r}'')$ ;
    вычисляем оценку обратной интервальной матрицы  $\mathbf{Y}' \supseteq (\mathbf{Q}')^{-1}$ 
        и, возможно,  $\mathbf{Y}'' \supseteq (\mathbf{Q}'')^{-1}$ ;
    вычисляем оценку  $\Upsilon(\text{mid } \mathbf{Q}', \text{mid } \mathbf{r}')$  и, возможно,  $\Upsilon(\text{mid } \mathbf{Q}'', \text{mid } \mathbf{r}'')$ 
        и присваиваем  $\mu \leftarrow \min\{\Upsilon(\text{mid } \mathbf{Q}', \text{mid } \mathbf{r}'), \Upsilon(\text{mid } \mathbf{Q}'', \text{mid } \mathbf{r}'')\}$ ;
    удаляем из списка  $\mathcal{L}$  бывшую ведущую запись  $(\mathbf{Q}, \mathbf{r}, v, W, s, t, \mathbf{Y}, \mathbf{x})$ ;
    если  $v' \leq \omega$ , то помещаем запись  $(\mathbf{Q}', \mathbf{r}', v', W', s', t', \mathbf{Y}', \mathbf{x}')$  в список  $\mathcal{L}$ 
        в порядке возрастания значений переменной  $v$  в третьем поле;
    если ведущая ИСЛАУ породила две системы-потомка и  $v'' \leq \omega$ ,
        то помещаем запись  $(\mathbf{Q}'', \mathbf{r}'', v'', W'', s'', t'', \mathbf{Y}'', \mathbf{x}'')$  в список  $\mathcal{L}$ 
        в порядке возрастания значений переменной  $v$  в третьем поле;
    если  $\omega > \mu$ , то полагаем  $\omega \leftarrow \mu$  и чистим список  $\mathcal{L}$ : удаляем из него
        все такие записи  $(\mathbf{Q}, \mathbf{r}, v, W, s, t, \mathbf{Y}, \mathbf{x})$ , что  $v > \omega$ ;
END DO

```

В качестве алгоритма *Encl* была использована также процедура `verifylss` из пакета INTLAB [17]. Процедура $\mathbf{x} = \text{verifylss}(\mathbf{A}, \mathbf{b})$ предназначена для внешнего оценивания множества решений интервальной системы $\mathbf{A}x = \mathbf{b}$ и состоит из двух этапов, первый из которых реализует алгоритм, основанный на операторе Кравчика, второй — процедуру Хансена — Блика — Рона. Если на первом этапе интервальная оценка \mathbf{x} решения ИСЛАУ в течение семи итераций не находится, то выполняется второй этап, т. е. решение будет получено с помощью процедуры Хансена — Блика — Рона.

Следует отметить, что во всех базовых алгоритмах применялось преобуславливание ИСЛАУ обратной средней матрицей. Соответствующие базовые алгоритмы использовались также для внешнего оценивания “обратной интервальной матрицы”, которая применяется в тесте на монотонность и при выборе рассекаемого интервального элемента.

Реализованы следующие способы организации и обработки списка:

— список \mathcal{L} упорядочен по возрастанию оценки $\Upsilon(\mathbf{Q}, \mathbf{r})$ таким образом, что первая его запись является одновременно и ведущей. Чистка бесперспективных записей производится только при изменении параметра ω ;

— список \mathcal{L} организован в виде кучи [18]. Ведущая запись, для которой оценка $\Upsilon(\mathbf{Q}, \mathbf{r})$ минимальна, определяется при просмотре всего списка. Чистка бесперспективных записей осуществляется только при изменении параметра ω ;

— в списке \mathcal{L} выделяется упорядоченный подсписок $\mathcal{L}_\gamma = \{(\mathbf{Q}, \mathbf{r}, \Upsilon(\mathbf{Q}, \mathbf{r}), W, s, t, \mathbf{Y}, \mathbf{x}) \mid \Upsilon(\mathbf{Q}, \mathbf{r}) < \gamma\}$ записей, которые будем называть активными. Здесь γ — некоторая “пороговая константа” такая, что $\Upsilon(\mathbf{Q}, \mathbf{r}) < \gamma < \omega$. Дополнение $\mathcal{L} \setminus \mathcal{L}_\gamma$ организовано в виде кучи. Если в процессе работы алгоритма подмножество \mathcal{L}_γ станет пустым, то пороговая константа γ перевычисляется. Например, можно взять $\gamma = \frac{1}{1+\lambda}(\lambda \cdot \omega + \Upsilon(\mathbf{Q}, \mathbf{r}))$, где $0 < \lambda < 1$, и из \mathcal{L} снова выделить упорядоченный подсписок \mathcal{L}_γ . Производится чистка не всего списка, а только подписка активных записей. Данный способ обработки списка предложен П.С. Панковым [19];

— в списке \mathcal{L} выделяется упорядоченный подсписок \mathcal{L}_l активных записей, имеющий некоторую фиксированную максимальную длину l ; дополнение $\bar{\mathcal{L}} = \mathcal{L} \setminus \mathcal{L}_l$ организовано в виде кучи. Ведущей записью считается первая запись \mathcal{L}_l .

Если на некотором шаге алгоритма длина \mathcal{L}_l станет равной l , т.е. подсписок активных записей полностью заполнится, то новая запись в зависимости от содержащейся в ней оценки $\Upsilon(\mathbf{Q}, \mathbf{r})$ может быть внесена либо в \mathcal{L}_l , либо в подсписок $\bar{\mathcal{L}}$, организованный в виде кучи. Обозначим через Υ_{\max} максимальную оценку $\Upsilon(\mathbf{Q}, \mathbf{r})$, содержащуюся в записях подписка \mathcal{L}_l . В случае, когда длина \mathcal{L}_l равна l , новая запись с оценкой $\Upsilon(\mathbf{Q}, \mathbf{r}) < \Upsilon_{\max}$ вносится в подсписок активных записей. При этом происходит его переполнение и лишняя запись помещается в кучу $\bar{\mathcal{L}}$. Записи-потомки с оценками $\Upsilon(\mathbf{Q}, \mathbf{r}) \geq \Upsilon_{\max}$ при полностью заполненном подписке \mathcal{L}_l вносятся в $\bar{\mathcal{L}}$.

Если подсписок \mathcal{L}_l активных записей становится пустым, то из $\bar{\mathcal{L}}$ отбираются не более l записей с наименьшими оценками $\Upsilon(\mathbf{Q}, \mathbf{r})$ и помещаются упорядоченно в \mathcal{L}_l .

Чистка подписка $\bar{\mathcal{L}}$ производится при изменении параметра ω , чистка подписка \mathcal{L}_l выполняется только при $\bar{\mathcal{L}} = \emptyset$.

Представленные алгоритмы были апробированы на следующих тестовых интервальных линейных системах уравнений.

Пример 1. Интервальная линейная система Ноймайера [20] имеет вид

$$\begin{pmatrix} \theta & [0, 2] & \cdots & [0, 2] \\ [0, 2] & \theta & \cdots & [0, 2] \\ \vdots & \vdots & \ddots & \vdots \\ [0, 2] & [0, 2] & \cdots & \theta \end{pmatrix} x = \begin{pmatrix} [-1, 1] \\ [-1, 1] \\ \vdots \\ [-1, 1] \end{pmatrix}, \quad (22)$$

где θ — неотрицательный вещественный параметр. Отметим, что матрица системы четного порядка n неособенна при $\theta > n$, нечетного порядка n — при $\theta > \sqrt{n^2 - 1}$. При

приближении θ к границам неособенности размеры объединенного множества решений ИСЛАУ (22) неограниченно возрастают. Варьируя θ , можно получить набор тестов для проверки рассмотренных алгоритмов оптимального решения “внешней задачи” ИСЛАУ.

Пример 2. Интервальная линейная система уравнений, впервые рассмотренная С.П. Шарым в [4], имеет вид

$$\begin{pmatrix} [n-1, N] & [\alpha-1, 1-\beta] & \cdots & [\alpha-1, 1-\beta] \\ [\alpha-1, 1-\beta] & [n-1, N] & \cdots & [\alpha-1, 1-\beta] \\ \vdots & \vdots & \ddots & \vdots \\ [\alpha-1, 1-\beta] & [\alpha-1, 1-\beta] & \cdots & [n-1, N] \end{pmatrix} x = \begin{pmatrix} [1-n, n-1] \\ [1-n, n-1] \\ \vdots \\ [1-n, n-1] \end{pmatrix}, \quad (23)$$

где n — размерность системы ($n \geq 2$), $0 < \alpha \leq \beta \leq 1$, N — вещественное число, не меньшее $n-1$. Варьируя значения α , β , n и N , получим широкий набор ИСЛАУ для тестирования разработанных алгоритмов. Когда β уменьшается, приближаясь к нулю, матрица системы (23) становится все более близкой к особенной, а множество решений неограниченно увеличивается в размерах. Изменяя отношение α к β , можно модифицировать форму множества решений.

Можно показать [4], что оптимальными покомпонентными оценками множества $\tilde{\Xi}$ решений системы (23) являются

$$\min\{x_i \mid x \in \tilde{\Xi}\} = -1/\alpha,$$

$$\max\{x_i \mid x \in \tilde{\Xi}\} = 1/\alpha, \quad i = 1, 2, \dots, n,$$

причем эти оценки не зависят от конкретного значения N .

Пример 3. Интервальная линейная система Тофта [21] имеет вид

$$Ax = b, \quad (24)$$

где матрица системы

$$A = \begin{pmatrix} [1-r, 1+r] & 0 & \cdots & 0 & [1-r, 1+r] \\ 0 & [1-r, 1+r] & \cdots & 0 & [2-r, 2+r] \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & [1-r, 1+r] & [n-1-r, n-1+r] \\ [1-r, 1+r] & [2-r, 2+r] & \cdots & [n-1-r, n-1+r] & [n-r, n+r] \end{pmatrix}$$

и вектор правой части

$$b = \begin{pmatrix} [1-R, 1+R] \\ [1-R, 1+R] \\ \vdots \\ [1-R, 1+R] \end{pmatrix}.$$

Здесь r, R — положительные вещественные числа. Матрица данной системы является интервализацией известной тестовой матрицы вычислительной линейной алгебры из справочника [22].

4. Сравнительный анализ результатов

4.1. Влияние базового алгоритма

Мы провели ряд тестовых расчетов с интервальной системой Ноймайера (см. пример 1), меняя размерность системы n , значение ее параметра θ и базовый алгоритм *Encl* внешней оценки множества решений интервальной системы. В табл. 1 представлены результаты численных экспериментов — время нахождения оптимальной нижней оценки первой компоненты множества решений ИСЛАУ (см. пример 1). Базовые алгоритмы обозначены следующим образом: К — метод Кравчика, МК — модифицированный метод Кравчика, G — интервальный метод Гаусса, GS — интервальный метод Гаусса — Зейделя, HBR — процедура Хансена — Блика — Рона, V — процедура *verifylss* из пакета INTLAB.

Проанализировав полученные результаты, можно сделать следующие выводы.

Метод Кравчика, являясь итерационным, в качестве базового алгоритма оказался наименее эффективным. Несколько быстрее работает программа, если базовый алгоритм основан на модификации метода Кравчика и на интервальных методах Гаусса

Т а б л и ц а 1. Сравнение базовых алгоритмов (см. пример 1)

Параметр θ матрицы системы	Время работы программы на основе базовых алгоритмов, с					
	К	МК	G	GS	HBR	V
$n = 5$						
10	3.314	2.847	2.814	2.818	1.046	2.009
14	0.386	0.326	0.700	0.714	0.284	0.338
18	0.269	0.259	0.569	0.614	0.221	0.270
22	0.210	0.229	0.500	0.571	0.195	0.238
26	0.212	0.199	0.499	0.539	0.189	0.207
30	0.214	0.197	0.499	0.547	0.188	0.204
$n = 8$						
16	390.255	103.098	303.903	289.908	65.099	89.632
20	26.002	7.016	17.196	17.372	5.432	6.144
24	3.811	2.604	6.305	6.544	2.004	1.881
28	1.977	1.744	4.730	4.817	1.500	1.408
32	1.384	1.357	3.479	3.578	1.104	1.038
36	1.088	1.301	2.736	2.791	0.838	0.789
40	0.894	0.962	2.265	2.283	0.688	0.649
44	0.844	0.813	2.107	2.091	0.641	0.604
48	0.743	0.763	1.948	1.971	0.589	0.557
$n = 10$						
22	3282.707	610.170	1142.017	1028.602	166.874	520.371
26	100.611	30.148	84.395	82.148	25.425	26.211
30	20.640	12.953	34.359	34.145	10.293	9.4704
34	11.968	6.359	18.369	18.371	5.392	4.9562
38	6.561	4.994	14.252	14.230	4.185	3.852
42	5.435	3.719	11.098	11.075	3.256	3.003
46	3.865	3.182	8.694	8.661	2.555	2.423
50	2.836	2.595	8.459	8.408	2.474	2.290
54	2.681	2.516	7.017	6.969	2.058	1.898
58	2.286	2.311	5.813	5.750	1.704	1.573

и Гаусса — Зейделя. Применение этих алгоритмов как базовых, скорее всего, нежелательно.

Наилучшую скорость работы показала программа, использующая процедуру Хансена — Блика — Рона. Эффективность применения в качестве базового алгоритма процедуры `verifylss` из пакета INTLAB оказалась несколько хуже за счет того, что на первом этапе ее работы, основанном на методе Кравчика, решение не достигается, после чего используется процедура Хансена — Блика — Рона.

Таким образом, при выборе базового алгоритма следует предпочесть процедуру Хансена — Блика — Рона.

При анализе примера 1, изменяя значение параметра θ , было исследовано влияние свойств интервальной матрицы системы — ее близость к границам неособенности — на работу программы. В табл. 1 приведены результаты численных экспериментов с интервальной системой Ноймайера при различных значениях параметра θ (приведены те из них, при которых время работы программы не слишком велико).

Для характеристики свойств интервальной матрицы системы использованы величины:

- 1) $\rho = \rho(|(\text{mid } \mathbf{A})^{-1}| \cdot \text{rad } \mathbf{A})$ — спектральный радиус матрицы $|(\text{mid } \mathbf{A})^{-1}| \cdot \text{rad } \mathbf{A}$;
- 2) $\Delta\sigma = \sigma_{\min}(\text{mid } \mathbf{A}) - \sigma_{\max}(\text{rad } \mathbf{A})$ — разность между наименьшим сингулярным числом матрицы $\text{mid } \mathbf{A}$ и наибольшим сингулярным числом матрицы $\text{rad } \mathbf{A}$.

Напомним, что сингулярными числами матрицы $\mathbf{A} \in \mathbb{R}^{n \times n}$ называются неотрицательные квадратные корни из собственных чисел матрицы $\mathbf{A}\mathbf{A}^T$.

Выбор указанных выше величин мотивируется следующими признаками неособенности интервальных матриц.

Признак Риса — Бека. Пусть интервальная $(n \times n)$ -матрица \mathbf{A} такова, что $\text{mid } \mathbf{A}$ неособенная и $\rho(|(\text{mid } \mathbf{A})^{-1}| \cdot \text{rad } \mathbf{A}) < 1$. Тогда \mathbf{A} неособенная.

Теорема. Пусть интервальная $(n \times n)$ -матрица \mathbf{A} такова, что $\text{mid } \mathbf{A}$ неособенная и $\max_{1 \leq j \leq n} (\text{rad } \mathbf{A} \cdot |(\text{mid } \mathbf{A})^{-1}|)_{jj} \geq 1$. Тогда \mathbf{A} особенная.

Признак Румпа. Пусть для интервальной $(n \times n)$ -матрицы \mathbf{A} имеет место неравенство $\sigma_{\max}(\text{rad } \mathbf{A}) < \sigma_{\min}(\text{mid } \mathbf{A})$. Тогда \mathbf{A} неособенная.

Признак Рона — Рекса. Если для интервальной $(n \times n)$ -матрицы \mathbf{A} имеет место $\sigma_{\min}(\text{rad } \mathbf{A}) \geq \sigma_{\max}(\text{mid } \mathbf{A})$, то она особенная.

В табл. 2 приведены значения рассматриваемых характеристик ρ и $\Delta\sigma$ матрицы Ноймайера при различных значениях ее диагонального параметра θ и размерности n .

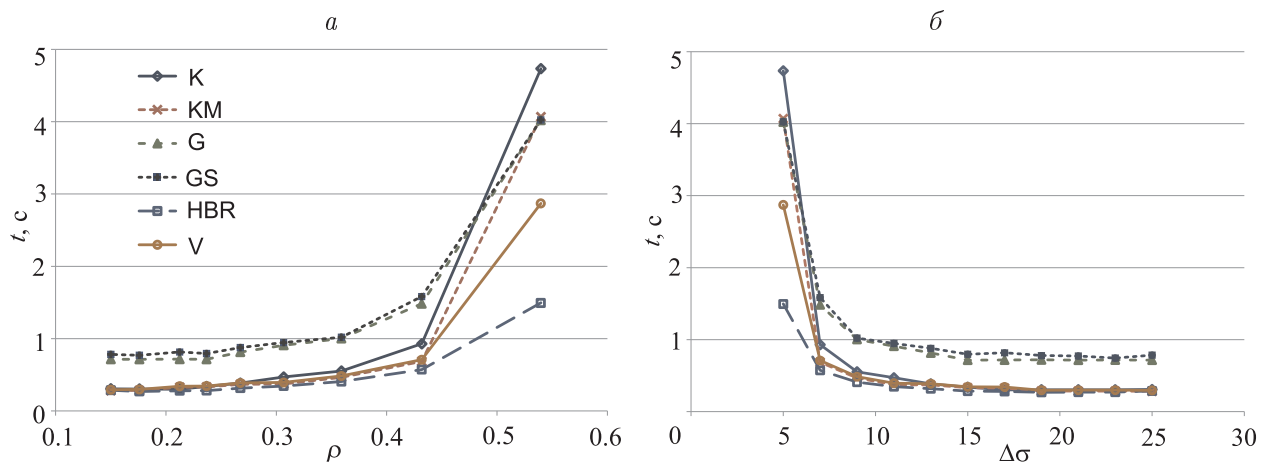
Отметим, что при близких к нулю значениях ρ и достаточно больших $\Delta\sigma$ время работы программы мало. Однако оно экспоненциально возрастает при приближении матрицы системы к границам неособенности, т. е. при $\rho \rightarrow 1$, $\Delta\sigma \rightarrow 0$.

На рис. 1 приведена зависимость времени работы программы от характеристик ρ и $\Delta\sigma$ интервальной матрицы \mathbf{A} размерности $n = 5$ для различных базовых алгоритмов.

В табл. 3 представлены результаты численных экспериментов с интервальной системой Шарого (пример 2) при разных значениях размерности n матрицы системы, ее параметров α , β , N и базовых методов внешней оценки множества решений интервальной системы. Здесь же приведены характеристики ρ и $\Delta\sigma$ интервальной матрицы системы и время нахождения оптимальной нижней оценки первой компоненты множества решений ИСЛАУ на основе разных базовых алгоритмов. На рис. 2 изображены диаграммы эффективности работы модификаций программы при разных значениях параметров и размерности интервальной системы Шарого.

Т а б л и ц а 2. Характеристики ρ и $\Delta\sigma$ матрицы Ноймайера

Параметр θ	$\rho((\text{mid } \mathbf{A})^{-1} \cdot \text{rad } \mathbf{A})$	$\sigma_{\min}(\text{mid } \mathbf{A}) - \sigma_{\max}(\text{rad } \mathbf{A})$
$n = 5$		
10	0.5397	5
14	0.3590	9
18	0.2674	13
22	0.2125	17
26	0.1760	21
30	0.1501	25
$n = 8$		
16	0.5884	8
20	0.4503	12
24	0.3633	16
28	0.3037	20
32	0.2605	24
36	0.2279	28
40	0.2024	32
44	0.1819	36
48	0.1652	40
$n = 10$		
24	0.5392	14
28	0.4423	18
32	0.3740	22
36	0.3235	26
40	0.2846	30
44	0.2539	34
48	0.2291	38
52	0.2086	42
56	0.1914	46
60	0.1767	50

Рис. 1. Зависимость времени работы программы t от характеристик ρ (а) и $\Delta\sigma$ (б) матрицы Ноймайера

Анализируя полученные результаты, можно сделать выводы, аналогичные изложенным выше. Наименее эффективной оказалась программа, использующая в качестве базового метод Кравчика и его модификацию (эти данные в табл. 3 не приводятся). Наилучшую скорость работы имеем в случае, когда базовым методом является процедура Хансена — Блика — Рона. Скорость работы программы на основе процедуры `verifylss` из пакета INTLAB значительно падает при возрастании размерности интервальной си-

Т а б л и ц а 3. Сравнение базовых алгоритмов (пример 2)

Параметр N интервальной матрицы системы	Характеристика матрицы системы		Время работы программы на основе базовых алгоритмов, с			
	ρ	$\Delta\sigma$	G	GS	HBR	V
$\alpha = 0.4, \beta = 0.6, n = 10$						
15	0.6757	3.6	11.225	9.054	2.356	3.578
20	0.7353	3.6	10.823	9.322	2.341	3.853
25	0.7764	3.6	10.811	9.211	2.322	3.855
$\alpha = 0.4, \beta = 0.6, n = 20$						
25	0.6219	7.6	105.509	93.826	21.935	39.846
30	0.6637	7.6	105.558	97.270	21.959	25.755
35	0.6972	7.6	105.554	95.336	21.880	77.288
$\alpha = 0.4, \beta = 0.6, n = 30$						
35	0.6014	11.6	386.129	407.671	91.435	456.992
40	0.6329	11.6	386.522	426.977	91.432	358.437
45	0.6598	11.6	385.823	430.413	92.363	676.331
$\alpha = 0.6, \beta = 0.8, n = 10$						
15	0.5150	5.4	3.659	3.582	1.081	2.313
20	0.6029	5.4	7.915	6.450	1.622	1.876
25	0.6646	5.4	7.904	6.377	1.626	1.919
$\alpha = 0.6, \beta = 0.8, n = 20$						
25	0.4328	11.4	23.631	21.467	5.309	17.544
30	0.4956	11.4	23.892	21.540	5.310	47.364
35	0.5458	11.4	27.784	24.573	6.299	97.143
$\alpha = 0.6, \beta = 0.8, n = 30$						
35	0.4021	17.4	74.710	68.064	16.188	216.368
40	0.4494	17.4	74.648	67.998	16.192	441.926
45	0.4897	17.4	74.671	67.643	16.188	1616.238
$\alpha = 0.3, \beta = 0.7, n = 10$						
15	0.7353	2.7	13.542	12.483	3.373	5.410
20	0.7874	2.7	13.435	12.864	3.259	4.170
25	0.8224	2.7	13.457	12.525	3.360	4.186
$\alpha = 0.3, \beta = 0.7, n = 20$						
25	0.6868	5.7	157.602	151.104	34.394	34.016
30	0.7246	5.7	159.019	144.843	34.416	34.065
35	0.7543	5.7	161.887	141.492	34.346	37.167
$\alpha = 0.3, \beta = 0.7, n = 30$						
35	0.6679	8.7	712.694	656.571	149.929	145.106
40	0.6969	8.7	704.724	670.453	150.615	146.732
45	0.7212	8.7	712.297	647.965	151.154	147.519

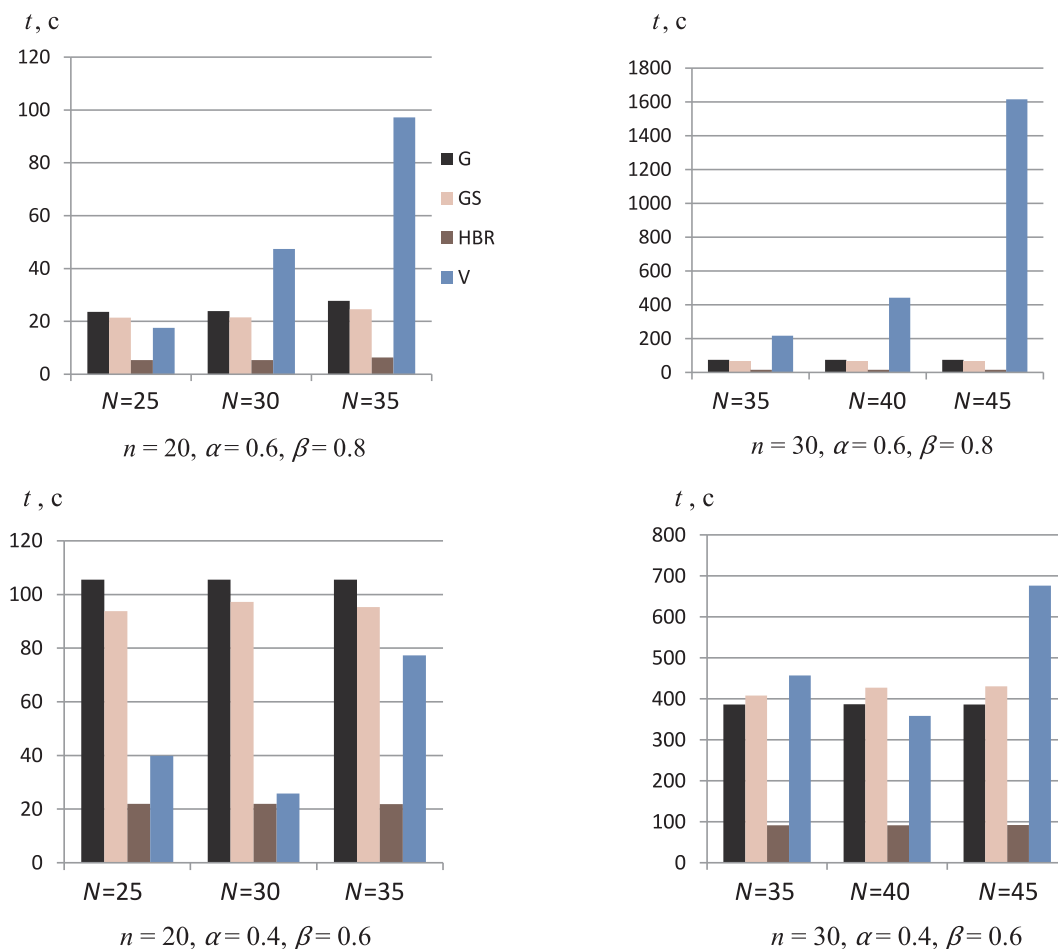


Рис. 2. Эффективность алгоритма при разных значениях параметров и размерности интервальной системы Шарого

стемы за счет увеличения времени выполнения итераций метода Кравчика на первом этапе. Модификации на основе базовых методов Гаусса и Гаусса — Зейделя показывают достаточно близкие по эффективности результаты.

Из табл. 3 также видно, что при фиксированных значениях n , α и β увеличение параметра N не приводит к изменению характеристики $\Delta\sigma$ интервальной матрицы системы (при этом ρ варьирует в небольших пределах). Поэтому скорость работы программы, основанной на базовых методах Гаусса, Гаусса — Зейделя и процедуре Хансена — Блика — Рона, меняется незначительно. Однако модификация программы, использующая процедуру `verifylss`, оказывается весьма чувствительной к изменению N (см. рис. 2). Это объясняется тем, что интервальная оценка решения, как правило, вычисляется на первом этапе работы процедуры, т. е. с помощью модифицированного метода Кравчика, а точность полученной оценки существенно зависит от значения N .

Далее была проведена апробация разработанных алгоритмов и их модификаций на тестовой интервальной системе, описанной в примере 3 (табл. 4). Сравнивая результаты работы алгоритмов, основанных на разных базовых методах, можно сделать выводы, аналогичные полученным ранее.

Таким образом, на ряде тестовых примеров апробированы шесть базовых методов для использования их в разработанном алгоритме дробления параметров с модифика-

Т а б л и ц а 4. Сравнение базовых алгоритмов (пример 3)

Параметры матрицы системы $r = R$	Характеристика матрицы системы ρ	Время работы программы на основе базовых алгоритмов, с			
		G	GS	HBR	V
$n = 5$					
0.1	0.2281	0.480	0.311	0.221	0.378
0.2	0.4562	0.422	0.375	0.116	0.199
0.3	0.6844	1.161	1.151	0.307	0.688
$n = 10$					
0.1	0.2028	0.813	0.810	0.200	0.216
0.2	0.4056	2.185	1.921	0.475	0.435
0.3	0.6083	4.083	5.345	0.820	1.399
$n = 15$					
0.1	0.1998	1.771	1.769	0.419	0.410
0.2	0.3997	5.861	5.101	1.154	0.838
0.3	0.5995	13.834	16.335	2.594	4.690
$n = 20$					
0.1	0.1991	3.107	3.094	0.696	0.678
0.2	0.3983	9.126	7.789	1.571	1.728
0.3	0.5974	41.162	43.119	7.576	11.403
$n = 25$					
0.1	0.199	4.863	4.805	1.062	1.592
0.2	0.3979	14.129	12.084	1.694	2.016
0.3	0.5969	90.467	104.553	16.273	27.129
$n = 30$					
0.1	0.1989	10.969	10.546	1.557	2.238
0.2	0.3979	22.807	19.828	3.450	4.110
0.3	0.5968	216.787	231.572	39.057	60.433

цией Рона. Среди них наиболее эффективной оказалась процедура Хансена — Блика — Рона. Время работы алгоритма нахождения оптимальной внешней оценки множества решений ИСЛАУ зависит как от ее размерности, так и от близости к границам неосо-бенности.

4.2. Эффективность процедуры работы со списком

В разделе 3 были описаны следующие четыре способа организации и обработки списка записей, в которых хранится информация о системах-потомках, полученных в результате дробления параметров:

- 1) список \mathcal{L} организован в виде кучи;
- 2) записи списка \mathcal{L} упорядочены по возрастанию оценки $\Upsilon(\mathbf{Q}, \mathbf{r})$;
- 3) в списке \mathcal{L} выделяется упорядоченный подсписок \mathcal{L}_l активных записей, имеющий некоторую фиксированную максимальную длину, а остальные записи организованы в виде кучи;
- 4) способ, предложенный Панковым.

Соответствующие модификации алгоритма оптимального внешнего оценивания множества решений ИСЛАУ, реализующие данные способы работы со списком, апробиро-

Т а б л и ц а 5. Эффективность алгоритмов организации и обработки списка

Параметры и размерность интервальной системы		Время работы программы, с			
		способ 1	способ 2	способ 3	способ 4
<i>Интервальная система Ноймайера</i>					
$n = 6$	$\theta = 12$	4.347	3.013	2.082	2.943
	$\theta = 16$	1.000	0.677	0.667	0.670
	$\theta = 20$	0.596	0.404	0.399	0.400
$n = 8$	$\theta = 16$	119.767	73.602	65.039	72.844
	$\theta = 20$	10.241	5.452	5.432	5.427
	$\theta = 24$	3.859	2.036	2.004	2.031
$n = 10$	$\theta = 20$	5815.058	2323.042	1407.043	2235.154
	$\theta = 22$	336.505	185.925	166.874	184.065
	$\theta = 24$	67.605	41.349	40.886	41.154
$n = 12$	$\theta = 28$	853.403	461.624	420.766	460.146
	$\theta = 30$	301.189	178.815	175.215	176.626
	$\theta = 35$	84.595	48.643	48.003	48.689
<i>Интервальная система Шарого</i>					
$n = 6$	$N = 20, n = 10$	5.739	3.428	3.259	3.275
	$N = 30, n = 20$	67.014	35.038	34.416	35.523
	$N = 40, n = 30$	297.970	151.301	150.615	151.303
<i>Интервальная система Тофта</i>					
$n = 8$	$n = 10$	1.340	0.856	0.820	0.857
	$n = 20$	13.358	7.772	7.576	7.784
	$n = 30$	71.625	39.921	39.057	39.930

ваны на тестовых системах (см. примеры 1–3). Результаты численных экспериментов представлены в табл. 5. В качестве базового алгоритма использовалась процедура Хансена — Блика — Рона.

Наименее эффективным оказался способ организации списка в виде кучи (способ 1). Очевидно, на каждом шаге алгоритма достаточно много времени требуется на просмотр всего списка с целью поиска ведущей записи. Скорость обработки списка значительно повышается, если список упорядочивается по возрастанию оценки $\Upsilon(\mathbf{Q}, \mathbf{r})$ (способ 2). Однако процесс сортировки списка, добавления и удаления записей из него требует определенных временных затрат. Поэтому имеет смысл поддерживать упорядоченность, а также производить чистку не всего списка \mathcal{L} , а некоторой его части — подсписка активных записей — способы 3 и 4. Если последние сравнивать по временным затратам, то следует отдать предпочтение способу 3, при котором фиксируется максимальная длина подсписка активных записей. В способе 4, предложенном Панковым, длина подсписка активных записей определяется пороговой константой γ и на некоторых шагах алгоритма может быть достаточно большой, что приводит к снижению его быстродействия, особенно в случае, когда матрица системы близка к границам неособенности.

При организации списка по способу 4 на каждом шаге алгоритма ведущая запись содержит наименьшую оценку $\Upsilon(\mathbf{Q}, \mathbf{r})$. Данное свойство может нарушаться при организации списка по способу 3, что приводит к увеличению количества итераций, необходимых для получения оптимального решения ИСЛАУ.

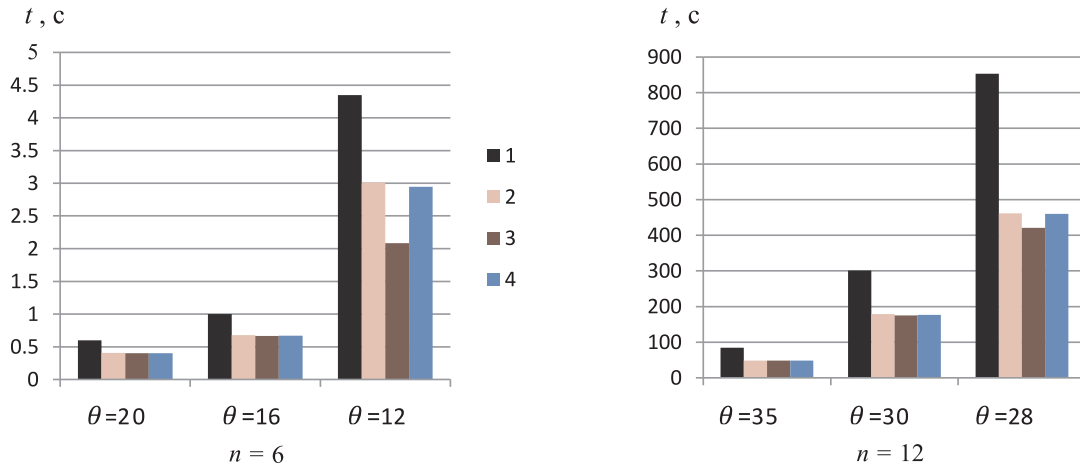


Рис. 3. Эффективность алгоритмов организации и обработки списка

По результатам тестовых экспериментов с системой Ноймайера при разных размерностях ($n = 6$ и $n = 12$) и значениях параметра θ построены диаграммы (рис. 3). Напомним, что при возрастании θ увеличивается характеристика матрицы системы ρ и уменьшается характеристика $\Delta\sigma$ (см. табл. 2), т. е. матрица ИСЛАУ приближается к границам неособенности. При этом, как видно из рис. 3, разница в быстродействии различных способов обработки списка становится более заметной.

4.3. Сравнение процедур оптимального внешнего оценивания множества решений ИСЛАУ

Приведем результаты сравнительного анализа работы двух алгоритмов оптимального внешнего оценивания множества решений интервальной линейной системы уравнений $\mathbf{Ax} = \mathbf{b}$:

- процедуры `verintervalhull` из пакета VERSOFT [23], основанной на методе Рона (см. раздел 2) и написанной им самим;
- процедуры `linppsr`, реализующей метод дробления параметров с модификацией Рона, где в качестве базового метода используется процедура Хансена — Блика — Рона, а список записей организован и обрабатывается по способу 3 (см. раздел 4.2).

В табл. 6 представлены результаты тестовых расчетов с интервальными системами (см. примеры 1–3) при различных размерностях и значениях параметров матрицы ИСЛАУ. Следует заметить, что задача вычисления оптимальных внешних оценок множества решений интервальной системы является NP-трудной. Время работы рассмотренных алгоритмов экспоненциально возрастает с увеличением размерности n системы. Из таблицы видно, что скорость работы процедуры `verintervalhull` не зависит от свойств интервальной матрицы \mathbf{A} решаемой системы и остается неизменной при различных значениях ее параметров. Быстродействие алгоритма `linppsr` снижается, если матрица \mathbf{A} близка к границам неособенности. Однако при больших размерностях n процедура `linppsr` намного эффективнее, чем `verintervalhull`.

Как было показано в разделе 2, искомые оценки $\min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$ и $\max\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$, $\nu = 1, 2, \dots, n$, достигаются на множестве не более чем 2^n экстремальных решений, т. е. верхняя оценка сложности метода Рона равна 2^n . Сложность методов дробления параметров без модификации Рона в худшем случае пропорциональна 2^{n^2} .

Т а б л и ц а 6. Эффективность алгоритмов `verintervalhull` и `linppsr`

Параметры и размерность интервальной системы		Время работы алгоритма, с	
		<code>linppsr</code>	<code>verintervalhull</code>
<i>Интервальная система Ноймайера</i>			
$n = 5$	$\theta = 10$	8.012	3.017
	$\theta = 20$	1.642	3.019
	$\theta = 30$	1.641	3.025
$n = 10$	$\theta = 25$	617.238	144.113
	$\theta = 30$	197.024	144.226
	$\theta = 45$	51.422	144.815
	$\theta = 60$	31.065	145.012
$n = 12$	$\theta = 30$	2943.612	1084.556
	$\theta = 50$	246.804	1084.476
	$\theta = 70$	99.308	1085.555
	$\theta = 90$	57.049	1084.001
<i>Интервальная система Шарого</i>			
$\alpha = 0.4, \beta = 0.6$	$n = 10, N = 15$	42.528	144.717
	$n = 20, N = 25$	882.344	Более 8 ч
	$n = 30, N = 35$	5483.106	Более 8 ч
$\alpha = 0.6, \beta = 0.8$	$n = 10, N = 15$	17.477	142.084
	$n = 20, N = 25$	209.250	Более 8 ч
	$n = 30, N = 35$	969.280	Более 8 ч
<i>Интервальная система Тофта</i>			
$r = 0.2$	$n = 10$	5.750	9.115
	$n = 20$	56.524	1321.249
	$n = 30$	161.393	Более 8 ч

Однако менее трудоемким алгоритм дробления параметров становится за счет того, что на каждом его шаге существенно используется информация, полученная в ходе выполнения предыдущих шагов, т. е. алгоритм в данном случае является адаптивным.

Важная особенность метода дробления параметров состоит также в том, что он порождает последовательность приближенных оценок для $\min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$ и $\max\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$, $\nu = 1, 2, \dots, n$, снизу и сверху соответственно. Поэтому в случае труднорешаемой задачи, когда размерность системы достаточно велика или матрица системы близка к границам неособенности, процесс работы алгоритма `linppsr` может быть прерван до своего полного завершения, причем полученные оценки будучи не оптимальными могут служить более или менее точными приближениями искомым $\min\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$ и $\max\{x_\nu \mid x \in \Xi(\mathbf{A}, \mathbf{b})\}$, $\nu = 1, 2, \dots, n$.

Таким образом, среди рассмотренных реализаций методов дробления параметров с модификацией Рона, различающихся базовым алгоритмом (метод Кравчика, модифицированный метод Кравчика с использованием “эпсилон-раздутия”, интервальный метод Гаусса, интервальный метод Гаусса — Зейделя, процедура Хансена — Блика — Рона), наиболее предпочтителен по времени работы алгоритм, использующий процедуру Хансена — Блика — Рона с преобуславливанием обратной средней матрицей. Для достижения наибольшей эффективности алгоритма имеет смысл при работе со списком систем-потомков, полученных в результате дробления параметров, выделять подписание ак-

тивных записей, в котором поддерживается упорядоченность по значениям оценки и производится чистка от “бесперспективных” потомков.

Анализ результатов апробации алгоритма `linpprs`, реализующего метод дробления параметров с модификацией Рона, показал, что время работы программы существенно зависит от свойств интервальной матрицы системы, в частности, от близости этой матрицы к границе множества неособенных (невырожденных) матриц. Если матрица системы “почти особенная”, а также если размерность системы велика, быстродействие алгоритма снижается. В этом случае для получения решения за практически приемлемое время можно воспользоваться свойством последовательной гарантии методов дробления параметров и прервать работу программы для вычисления некоторого приближения искомой внешней оценки. Это свойство алгоритма `linpprs` выгодно отличает его от других подходов к оптимальному внешнему оцениванию множеств решений интервальных линейных систем, например, рассмотренной выше процедуры `verintervalhull`, реализующей метод Рона.

Список литературы

- [1] KREINOVICH V., LAKEYEV A.V., ROHN J., KANL P. Computational Complexity and Feasibility of Data Processing and Interval Computations. Dordrecht: Kluwer, 1997.
- [2] KREINOVICH V., LAKEYEV A.V., NOSKOV S.I. Optimal solution of interval linear systems is intractable (NP-hard) // Interval Comput. 1993. No. 1. P. 6–14.
- [3] ROHN J., KREINOVICH V. Computing exact componentwise bounds on solutions of linear systems with interval data is NP-hard // SIAM J. on Matrix Analysis and Appl. 1995. Vol. 16. P. 415–420.
- [4] ШАРЫЙ С.П. Конечномерный интервальный анализ. Электронная книга <http://www-sbras.nsc.ru/interval/Library/InteBooks/SharyBook.pdf>
- [5] ШАРЫЙ С.П. Оптимальное внешнее оценивание множеств решений интервальных систем уравнений. Часть 1 // Вычисл. технологии. 2002. Т. 7, № 6. С. 90–113.
Шарый С.П. Оптимальное внешнее оценивание множеств решений интервальных систем уравнений. Часть 2 // Там же. 2003. Т. 8, № 6. С. 84–109.
- [6] SHARY S.P. A new class of algorithms for optimal solution of interval linear systems // Interval Comput. 1992. No. 2(4). P. 18–29.
- [7] ROHN J. Systems of linear interval equations // Linear Algebra and its Appl. 1989. Vol. 126. P. 39–78.
- [8] BEECK H. Über die Struktur und Abschätzungen der Lösungsmenge von linearen Gleichungssystemen mit Intervalkoeffizienten // Computing. 1972. Vol. 10. P. 231–244.
- [9] NICKEL K. Die Überschätzung des Wertebereiches einer Funktion in der Intervallrechnung mit Anwendungen auf lineare Gleichungssysteme // Ibid. 1977. Vol. 18. P. 15–36.
- [10] OETTLI W. On the solution set of linear system with inaccurate coefficients // SIAM J. Numer. Anal. 1965. Vol. 2. P. 115–118.
- [11] ЗАДАЧИ линейной оптимизации с неточными данными / М. Фидлер, Й. Недома, Я. Рамик, И. Рон, К. Циммерманн. М. — Ижевск: РХД, 2008.
- [12] KRAWCZYK K.R., NEUMAIER A. An improved interval Newton operator // J. of Math. Analysis and Appl. 1986. Vol. 118. P. 194–201.

- [13] MAYER G. Epsilon-inflation in verification algorithms // J. of Comput. and Appl. Math. 1995. Vol. 60. P. 147–169.
- [14] RUMP S.M A note on epsilon-inflation // Reliable Comput. 1998. Vol. 4. P. 371–375.
- [15] NEUMAIER A. New techniques for the analysis of linear interval equations // Linear Algebra and its Appl. 1984. Vol. 58. P. 273–325.
- [16] NEUMAIER A. A simple derivation of Hansen—Bliiek—Rohn—Ning—Kearfott enclosure for linear interval equations // Reliable Comput. 1999. Vol. 5. P. 131–136.
- [17] HARGREAVES G.I. Interval analysis in MATLAB // Manchester Center for Computational Mathematics, 2002. (<http://old.ict.nsc.ru/interval/Programming/INTLABtutor.pdf>)
- [18] БАУЭР Ф.Л., ГООЗ Г. Информатика. М.: Мир, 1996.
- [19] ПАНКОВ П.С. Алгоритмы доказательства устойчивых утверждений и глобальной оптимизации в ограниченной области. Фрунзе, 1984 (Деп. в ВИНТИ, № 5250-84Деп).
- [20] NEUMAIER A. Interval methods for systems of equations. Cambridge Univ. Press, 1990.
- [21] ТОФТ О. Sequential and parallel solution of linear interval equations // Eksamensproject: NI-E-92-04. Numerisk Institute, Danmarks Tekniske Hojskole. Lyngby, 1992.
- [22] GREGORY R.T. A collection of matrices for testing computational algorithms. N.-Y.: Wiley-Intersci., 1978.
- [23] VERIFICATION software in MATLAB / INTLAB. Электронный ресурс, доступный на <http://www.cs.cas.cz/rohn/matlab>

*Поступила в редакцию 4 мая 2011 г.,
с доработки — 16 сентября 2011 г.*