Methods of solving sparse linear systems.

Soldatenko Oleg
SPbSU, Department of Computational Physics

Methods of solving sparse linear systems.

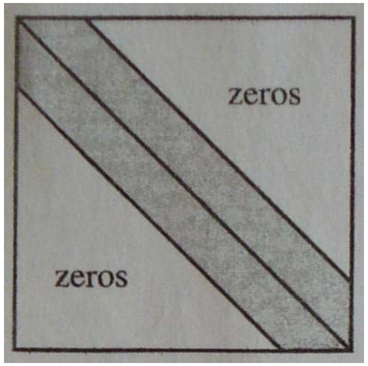
Outline

- Introduction
- Sherman-Morrison formula
- Woodbury formula
- Indexed storage of sparse matrices

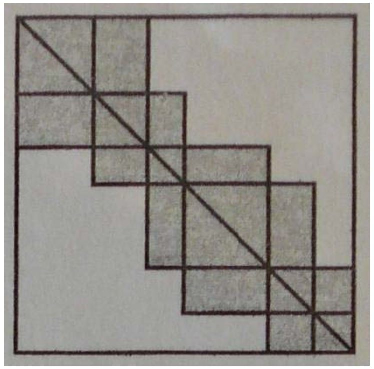
Methods of solving sparse linear systems.

Types of sparse matrixes

Band diagonal with bandwidth M



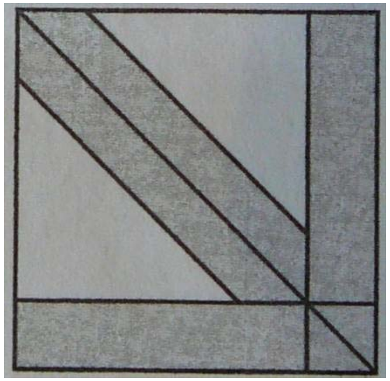
Block tridiagonal



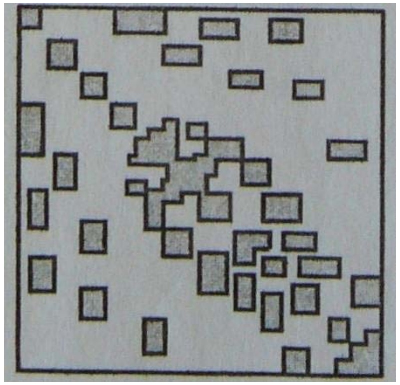
Methods of solving sparse linear systems.

Types of sparse matrixes

Doubly bordered block diagonal



Other…



Methods of solving sparse linear systems.

Sherman-Morrison Formula

- The general idea of Sherman-Morrison formula is replacement of the original matrix in the sum of matrix A and the product of vectors u and v.

$$A \longrightarrow (A + u \otimes v)$$

Methods of solving sparse linear systems.

Sherman-Morrison Formula

$$
\begin{aligned}
(A + u \otimes v)^{-1} &= (1 + A^{-1} \cdot u \otimes v)^{-1} \cdot A^{-1} \\
&= (1 - A^{-1} \cdot u \otimes v + A^{-1} \cdot u \otimes v \cdot A^{-1} \cdot u \\
&\otimes v - \dots) \cdot A^{-1} \\
&= A^{-1} - A^{-1} \cdot u \otimes v \cdot A^{-1} (1 - \lambda + \lambda^2 - \dots) \\
&= A^{-1} - \frac{(A^{-1} \cdot u) \otimes (v \cdot A^{-1})}{1 + \lambda}
\end{aligned}
$$

Where $\lambda \equiv v \cdot A^{-1} \cdot u$

Methods of solving sparse linear systems.

Sherman-Morrison Formula

Given A^{-1} and the vectors $\boldsymbol{u}$ and $\boldsymbol{v}$, we need only to perform matrix multiplication and a vector dot product,

$$\boldsymbol{z} \equiv A^{-1} \cdot \boldsymbol{u} \qquad \boldsymbol{w} \equiv (A^{-1})^T \cdot \boldsymbol{v}$$

$$\lambda \equiv \boldsymbol{v} \cdot \boldsymbol{z}$$

to find inverse matrix

$$A^{-1} \longrightarrow A^{-1} - \frac{\boldsymbol{z} \otimes \boldsymbol{w}}{1 + \lambda}$$

Methods of solving sparse linear systems.

Sherman-Morrison Formula

The Sherman-Morrison formula can be directly applied to a class of sparse problems. If you already have a fast way of calculating the inverse of A , then this method allows you to build up a method for more complicated matrices, adding for example a row or a column at a time. Notice that you can apply the Sherman-Morrison formula more than once successively, using at each stage the most recent update of A^{-1}. Of course, if you have to modify every row, then you are back to an N^3 method.

Methods of solving sparse linear systems.

Sherman-Morrison Formula

For some other sparse problems, the Sherman-Morrison formula cannot be directly applied for the simple reason that storage of the whole inverse matrix A^{-1} is not feasible. If you want to add only a single correction of the form $u \otimes v$, and solve the linear system

$$(A + u \otimes v) \cdot x = b$$

Then you proceed as follows. Using the fast method that is presumed available for the matrix , solve the two auxiliary problems

$$A \cdot y = b \qquad\qquad A \cdot z = u$$

Methods of solving sparse linear systems.

Sherman-Morrison Formula

In terms of these solutions,

$$x = y - \left(\frac{v \cdot y}{1 + (v \cdot z)}\right) z$$

Sherman-Morrison Formula

Example

Cyclic tridiagonal system:

$$\begin{bmatrix} b_0 & c_0 & 0 & \dots & & & \beta \\ a_1 & b_1 & c_1 & \dots & & & \\ & & & \dots & a_{N-2} & b_{N-2} & c_{N-2} \\ \alpha & & & \dots & 0 & a_{N-1} & b_{N-1} \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ \dots \\ x_{N-2} \\ x_{N-1} \end{bmatrix} = \begin{bmatrix} r_0 \\ r_1 \\ \dots \\ r_{N-2} \\ r_{N-1} \end{bmatrix}$$

Methods of solving sparse linear systems.

Sherman-Morrison Formula

- This is a tridiagonal system, except for the matrix elements α and β in the corners.
- We use the Sherman-Morrison formula, treating the system as a tridiagonal plus a correction.

$$u = \begin{bmatrix} \gamma \\ 0 \\ \vdots \\ 0 \\ \alpha \end{bmatrix} \qquad v = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ \beta/\gamma \end{bmatrix}$$

Methods of solving sparse linear systems.

Sherman-Morrison Formula

- Then the matrix is the tridiagonal part of initial matrix, with two terms modified:

$$b'_0 = b_0 - \gamma, \qquad b'_{N-1} = b_{N-1} - \frac{\alpha\beta}{\gamma}$$

We solve equations

$$A \cdot y = b \qquad A \cdot z = u$$

With the standard tridiagonal algorithm, for example sweep method.

Methods of solving sparse linear systems.

Sherman-Morrison Formula

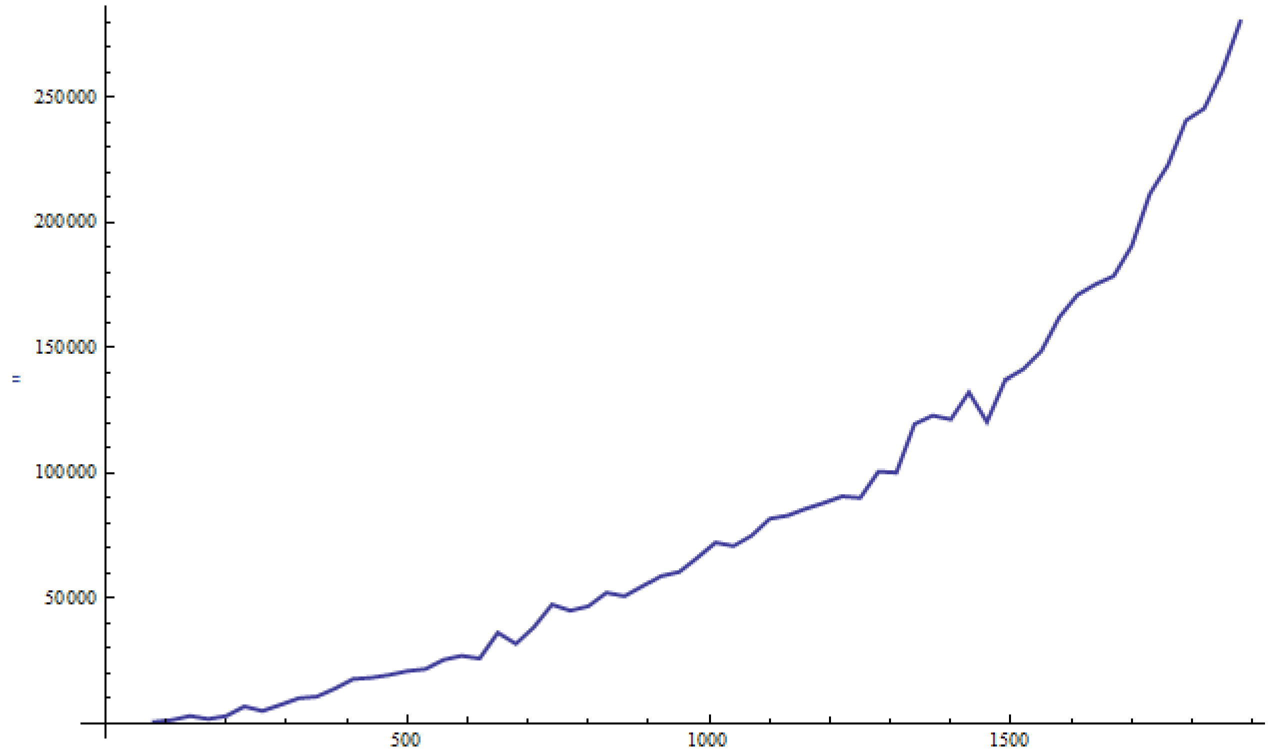
And get the solution by the formula

$$x = y - \left(\frac{v \cdot y}{1 + (v \cdot z)}\right) z$$

Methods of solving sparse linear systems.

Sherman-Morrison Formula

Acceleration with respect to Gauss method



Woodbury Formula

If you want to add more than a single correction term, then you cannot use Sharman-Morrison formula repeatedly, since without storing a new A^{-1} you are not able to solve the auxiliary problems efficiently after the first step. Instead you need the Woodbury Formula

$$(A + U \cdot V^T)^{-1} = A^{-1} - [A^{-1} \cdot U \cdot (1 + V^T \cdot A^{-1} \cdot U)^{-1} \cdot V^T \cdot A^{-1}]$$

Here A is, as usual, an $N \times N$ matrix, while U and V are matrices with $P < N$ and usually $P \ll N$.

Woodbury Formula

The relation between the Woodbury formula and successive applications of the Sherman-Morrison formula is now clarified by noting that, if U is the matrix formed by columns with P vectors $u_0, \ldots, u_{P-1}$, and V is the matrix formed by columns with P vectors $v_0, \ldots, v_{P-1}$

$$U \equiv \left[\begin{bmatrix} \\ u_0 \\ \\ \end{bmatrix} \ldots \begin{bmatrix} \\ u_{P-1} \\ \\ \end{bmatrix}\right] \qquad V \equiv \left[\begin{bmatrix} \\ v_0 \\ \\ \end{bmatrix} \ldots \begin{bmatrix} \\ v_{P-1} \\ \\ \end{bmatrix}\right]$$

Methods of solving sparse linear systems.

Woodbury Formula

Then two ways of expressing the same correction to are possible

$$\left(A + \sum_{k=0}^{P-1} u_k \otimes v_k\right) = (A + U \cdot V^T)$$

Note that the subscripts on ***u*** and ***v*** do not denote components, but rather distinguish the different column vectors.

Woodbury Formula

Last equation reveals that, if you have A^{-1} in storage, then you can either make the P corrections at once by using Woodbury formula, inverting a $P \times P$ matrix, or else make them by applying Sherman-Morrison formula P successive times.

If you don't have storage for A^{-1}, then you must use Woodbury formula in the following way: To solve the linear equation

$$\left(A + \sum_{k=0}^{P-1} u_k \otimes v_k \right) \cdot x = b$$

Methods of solving sparse linear systems.

Woodbury Formula

first solve the auxiliary problems

$$A \cdot z_0 = u_0$$
$$A \cdot z_1 = u_1$$
$$\dots$$
$$A \cdot z_{P-1} = u_{P-1}$$

and construct the matrix by columns of the obtained solutions

$$Z \equiv \left[\begin{bmatrix} \\ z_0 \\ \\ \end{bmatrix} \dots \begin{bmatrix} \\ z_{P-1} \\ \\ \end{bmatrix} \right]$$

Methods of solving sparse linear systems.

Woodbury Formula

Next, do the $P \times P$ matrix inversion

$$H \equiv (1 + V^T \cdot Z)^{-1}$$

Finally, solve one more auxiliary problem

$$A \cdot y = b$$

In terms of these quantities, the solution is given by the formula

$$x = y - Z \cdot [H \cdot (V^T \cdot y)]$$

Methods of solving sparse linear systems.

Indexed Storage of Sparse Matrices

- The first N locations of **sa** store diagonal matrix elements of **A**, in their order. (Note that diagonal elements are stored even if they are zeros; this is at most a slight storage inefficiency, since diagonal elements are nonzero in most realistic applications)
- Each of the first N locations of ija stores the index of the array sa that contains the first off-diagonal element of the corresponding row of the matrix. (If there are no off-diagonal elements for that row, it is one greater than the index in sa of the most recently stored element of a previous row)

Methods of solving sparse linear systems.

Indexed Storage of Sparse Matrices

- Location 0 of ija is always equal to N +1. (It can be read (или used) to determine N)
- Location N of ija is one greater than the index in sa of the last off-diagonal element of the last row. (It ca be read to determine the number of nonzero elements in the matrix, or the number of elements in the arrays sa and ija.) Location N of sa is not used and can be set arbitrarily.
- Entries in sa at locations >= N+1 contain off-diagonal elements of A ordered by rows and, within each row, ordered by columns.
- Entries in ija at locations >= N+1 contain the column number (или index) of the corresponding element in sa.

Methods of solving sparse linear systems.

Indexed Storage of Sparse Matrices

$$A = \begin{bmatrix} 3 & 0 & 1 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 \\ 0 & 7 & 5 & 9 & 0 \\ 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0 & 6 & 5 \end{bmatrix}$$

Index k		0	1	2	3	4	5	6	7	8	9	10
ija[k]		6	7	7	9	10	11	2	1	3	4	2
sa[k]		3	4	5	0	5	x	1	7	9	2	6

Thank you