Pivot-Free Block Matrix Inversion

Stephen M. Watt

Ontario Research Centre for Computer Algebra
University of Western Ontario, London, Canada.

Maple Conference 2006, July 23-26, Waterloo Canada.

Motivation

- Matrices may be represented as quad-trees, using a recursive 2×2 block structure with all-zero matrices given by a null pointer. This representation has been studied earlier in the context of computer algebra [1].
- This representation is convenient for reasonably efficient storage in diverse cases, when it is not known whether matrices will be dense, sparse or structured.
- It is also convenient for reasonably efficient communication in diverse cases, when matrices may be accessed by row, column or randomly.
- This representation naturally supports asymptotically fast block algorithms, such as Strassen's matrix multiplication [2].
- We are interested in measuring the performance of this representation for use in generic libraries [3, 4].

Block Matrix Inversion

An elegant way to express inversion of a 2×2 matrix over coefficient ring R is

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}, \qquad M^{-1} = \begin{bmatrix} (A - BD^{-1}C)^{-1} & (C - DB^{-1}A)^{-1} \\ (B - AC^{-1}D)^{-1} & (D - CA^{-1}B)^{-1} \end{bmatrix}.$$

To avoid inverting all of A, B, C, D, we may require only A be invertible:

$$M^{-1} = \begin{bmatrix} I & -A^{-1}B \\ 0 & I \end{bmatrix} \begin{bmatrix} A^{-1} & 0 \\ 0 & S_A^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -CA^{-1} & I \end{bmatrix}$$

$$= \begin{bmatrix} A^{-1} + A^{-1}BS_A^{-1}CA^{-1} & -A^{-1}BS_A^{-1} \\ -S_A^{-1}CA^{-1} & S_A^{-1} \end{bmatrix} \tag{1}$$

where $S_A = D - CA^{-1}B$ is the Schur complement of A in M. Alternatively,

$$M^{-1} = \begin{bmatrix} S_D^{-1} & -S_D^{-1}BD^{-1} \\ -D^{-1}CS_D^{-1} & D^{-1} + D^{-1}CS_D^{-1}BD^{-1} \end{bmatrix}. \tag{2}$$

These still require that either A or D be invertible, but for invertible M we may have that all of A, B, C, D are non-invertible, e.g.

$$\begin{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \end{bmatrix}$$

In this case, the usual approach is to abandon block matrix inversion and use a row-oriented divide and conquer pivoting algorithm for PLU decomposition [5].

Objective

We ask whether it is possible to formulate block matrix inversion in such a way that

1. only operations on entire blocks are used,
2. no case-based branching is required,
3. inverses are required only when they are guaranteed to exist and
4. applied recursively, the method gives asymptotically fast inversion?

As with other block-oriented methods, we do not require numerical stability.
We are able to answer this question positively. We show such a block matrix inversion that is applicable in a wide variety of settings.

Pivot-Free Inversion

To apply the recursive block methods given by (1) or (2), we must find a way to guarantee that the required blocks will be invertible. We note that, for invertible M and N,

$$M^{-1} = (NM)^{-1}N.$$

Thus, finding suitable N can give a recursive block algorithm at the cost of two additional matrix multiplications.

The Real Case

We first consider the case when R is a *formally real ring* with invertible elements $\mathrm{Inv}(R)$, i.e.

$$\forall a_1, \ldots, a_n \in R \qquad \bigvee_{i=1}^{n} a_i \in \mathrm{Inv}(R) \quad \Rightarrow \quad \sum_{i=1}^{n} a_i^2 \in \mathrm{Inv}(R)$$

Examples of formally real rings are the integers, rational and real numbers, and polynomials, rational functions and power series (possibly in non-commuting variables) over formally real rings.

In this case we choose $N = M^T$, giving

$$M^{-1} = (M^TM)^{-1}M^T,$$

which is a special case of the Moore-Penrose inverse [6, 7].

In computing $(M^TM)^{-1}$, the required inverses of A or D, at any recursive level, in (1) and (2) are of principal minors of M^TM, which by the Cauchy-Binet theorem are invertible. Note the diagonal elements of these principal minors will be sums of squares. The required Schur complements are also invertible because they occur within invertible blocks.

We note that the two extra matrix multiplications required by the Moore-Penrose inverse are for special forms of matrices. One multiplication is of a matrix with its own transpose, and the other is the multiplication of a symmetric matrix with a general matrix. For a matrix $M \in R^{n \times n}$, the product M^TM requires *at most*

$$\frac{n^2(2n^{\omega-2} + 2^\omega - 6)}{2^\omega - 4}$$

and *at least* $n^\omega/2^\omega$ multiplications in R, where n^ω is the cost of general matrix multiplication. The upper bound arises from the fact that

$$M^TM = \begin{bmatrix} A^T & C^T \\ B^T & D^T \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} A^TA + B^TB & A^TB + C^TD \\ (A^TB + C^TD)^T & C^TC + D^TD \end{bmatrix}.$$

For the best asymptotic algorithms the upper bound from this lemma is worse than the cost of general matrix multiplication and so is not useful. When Strassen matrix multiplication is used, however, general multiplication costs $n^{\log_2 7}$ and we require only $1/3n^2 + 2/3n^{\log_2 7}$.

The Complex Case

If R is formally real, let $C = R[i]/\langle i^2 + 1 \rangle$. All elements $c \in C$ may be written as $a + bi, a, b \in R$ and we may define conjugation as $\overline{a + bi} = a - bi$. Examples of rings of this form would be the Gaussian integers and the complex numbers.

In this setting, the arguments presented for the formally real case all follow when M^T is replaced by M^*, the conjugate transpose. $M^{-1} = (M^*M)^{-1}M^*$. The diagonal elements of the principal minors of M^*M will be real sums of the form $\sum_i a_i^* a_i$.

This argument holds for any ring with a suitable involution, for example the quaternions over formally real R and conjugation $\overline{a + bi + cj + dk} = a - bi - cj - dk$.

The Finite Field Case

For a field F of finite characteristic, we follow [8, 9] to construct a generalized Moore-Penrose inverse using a conjugation M° in the matrices of rational functions, $F(t)^{n \times n}$:

$$M^\circ = \mathrm{diag}(1, t^{-1}, \ldots, t^{-(n-1)})\ M^*\ \mathrm{diag}(1, t^1, \ldots, t^{n-1})$$

In this case we write $M^{-1} = (M^\circ M)^{-1}M^\circ$. The $k \times k$ principal minors of $M^\circ M$ will be invertible, with entries being Laurent polynomials in t with k terms.

In contrast to the real and complex cases, here computing a block Moore-Penrose inverse requires more expensive element arithmetic. Therefore, if only a single matrix inverse is required, then other methods will be more efficient. When working with parameters, however, using this method can avoid repeated calculations.
In large finite fields, it would be possible to start inversion with (1) and (2) and switch to this method only if a singular block is encountered.

Conclusion

When a matrix ring has a suitable notion of conjugate transpose, we may compute a deterministic, asymptotically fast, block-matrix inverse without pivots and without destroying the block abstraction.

Acknowledgements

The author would like to thank Wayne Eberly, Laureano Gonzalez Vega, Victoria Powers, B. David Saunders and Arne Storjohann for useful conversations.

References

[1] Abdali, S.K. and S. Wise, D.S.: *Experiments with quadtree representation of matrices.* In P. Gianni (ed.), Proc. ISSAC 88, Lecture Notes in Computer Science 358, Springer Verlag (1989), 96–108.

[2] Strassen, V.: *Gaussian Elimination is not Optimal.* Numerische Mathematik Vol. 13, 354-356 (1969).

[3] Dragan, L. and Watt, S.M.: *Performance Analysis of Generics for Scientific Computing.* Proc. 7th Int'l Symp. on Symbolic and Numeric Algorithms in Scientific Computing, (SYNASC 2005), Timişoara Romania, IEEE Press, 93-100.

[4] Watt, S.M.: *Aldor.* In *Handbook of Computer Algebra*, Grabmeier, Kaltofen, Weispfenning (eds) , Springer Verlag (2003), 265–270.

[5] Aho, A., Hopcroft, J. and Ullman, J.: *The Design and Analysis of Computer Algorithms.* Addison-Wesley, 1974.

[6] Moore, E.H.: *On the reciprocal of the general algebraic matrix.* Bull. American Math. Society 26, 394-395 (1920).

[7] Penrose, R.: *A generalized inverse for matrices.* Proc. of the Cambridge Philosophical Society 51, 406-413 (1955).

[8] Mulmuley, K.: *A fast parallel algorithm to compute the rank of a matrix over an arbitrary field*, Combinatorica, Vol. 7 (1987), 101-104.

[9] Diaz Toca, G.M, Gonzalez Vega, L., Lombardi, H.: *Generalizing Cramers rule: Solving uniformly linear systems of equations.* SIAM J. Matrix Anal. Appl. Vol. 27, No. 3, 621-637.