A STRUCTURAL DIAGNOSIS OF SOME IC ORDERINGS*

ROBERT BRIDSON† AND WEI-PAI TANG‡

Abstract. We present a novel analysis of the potential effectiveness of a matrix ordering for IC in terms of just the sparsity structure. By looking at the structure of the approximate inverse implicitly created by IC we can help to explain the success of Reverse Cuthill-McKee orderings, the problems IC(0) has under Red-Black orderings that disappear when extra fill is included, and where fill must be added to make fill-reducing orderings such as Minimum Degree effective.

Key words. ordering, incomplete factorization, incomplete Cholesky, preconditioner, sparse matrix, graph theory

AMS subject classifications. 05C78, 65F10, 65F50

1. Introduction.

The ILU class of preconditioners have proven to be very competitive. However, like any algorithm that involves factoring a matrix into triangular parts (exactly or only approximately), the ordering of the rows and columns can have a crucial effect. The issue of how best to choose that ordering is an important area of research continuing up to today[1, 3, 4, 5, 6, 8, 9].

A natural first choice for an ordering might be the best fill-reducing ordering available—maybe a Minimum Degree variant or some kind of Nested Dissection. This follows the intuition that if there are less fill entries to drop then the incomplete factorization will be more accurate.

However, it often turns out that envelope orderings like Reverse Cuthill-McKee orderings do better despite allowing more fill, at least for level-of-fill based incomplete factorizations and for fairly symmetric problems (e.g. see [1] for discussion of highly nonsymmetric matrices). Another puzzling observation is that Red-Black orderings usually give poor performance for ILU(0) but competitive performance for higher levels of fill.[1] These and many other orderings were considered in depth in [8], but the reasons for the varying performance are still not clear. Other papers have since further explored the effect of ordering on ILU, usually with a view to explaining (and minimizing) the typical trade-off between parallelism and convergence speed, but have generally stuck to model PDE problems on uniform rectangular grids[6, 9].

The following analysis provides some justification for these phenomena that works with the sparsity structure of the matrix, without considering the numerical values. The new results may be used as a guide for the construction of high-quality orderings, particularly in the common case where matrix entries may change from solve to solve but the sparsity structure and ordering stays the same.

We restrict our attention to symmetric positive definite matrices A with an incomplete Cholesky factorization $\bar{L}\bar{L}^T$ approximating the exact factorization LL^T. Our analysis is purely structural however, so the results are equally applicable to matrices with just symmetric structure and whose ILU factors have identical structure after

* The work was supported by the Natural Sciences and Engineering Council of Canada, and Communications and Information Technology Ontario (CITO), funded by the Province of Ontario.

†SCCM Program, Gates 2B, Stanford University, Stanford, CA 94305, USA. email: rbridson@stanford.edu

‡Department of Computer Science, University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada. email: wptang@elora.uwaterloo.ca

[1] Gene Golub has recently suggested that ILU shouldn't be applied directly to a Red-Black ordered matrix, but rather to the Schur complement formed after diagonal scaling and pre-elimination of the red nodes; our analysis could then be applied to the structure and ordering of the Schur complement.

transposition, as is the case with ILU(k). Extensions to the fully unsymmetric case are possible, but we have yet to find satisfying, intuitive results there.

2. Inverse Structure. The goal of an incomplete factorization is that the application of the preconditioner is close to the application of the true inverse of A. In other words, we want $(\bar{L}\bar{L}^T)^{-1}$ close to A^{-1}, viewing IC as a kind of implicit approximate inverse. This study focusses just on what the sparsity structure of the matrix can tell us; for orderings that consider numerical values, see [3, 4, 5, 6, 9] for example.

One obvious desirable property for the incomplete factorization is that $(\bar{L}\bar{L}^T)^{-1}$ at least should have the same nonzero structure as A^{-1} (e.g. fully dense for irreducible matrices, such as those arising from elliptic PDE's). If $(\bar{L}\bar{L}^T)^{-1}$ is constrained to have zeros where A^{-1} has non-negligible entries, the approximation cannot be good—the required coupling between nodes is absent.

Before proceeding, we will introduce some graph theory notation (see [10, 11] for more details). The (directed) graph of an $n \times n$ matrix B is a graph with vertices $1, \ldots, n$, and an arc $i \to j$ if and only if $B_{ij} \neq 0$. We will just write B instead of "the graph of B" where it is clear. A dipath is an ordered set $\langle i_1, i_2, \ldots i_p \rangle$ such that $i_1 \to i_2$, $i_2 \to i_3$, ..., and $i_{p-1} \to i_p$, often written $i_1 \to i_2 \to \cdots \to i_p$ or simply $i_1 \rightsquigarrow i_p$. The transitive closure of a graph $\mathcal{G}$ is a graph $\mathcal{G}^*$ on the same vertices but with an arc $i \to j$ in $\mathcal{G}^*$ whenever $i \rightsquigarrow j$ in $\mathcal{G}$.

In [11] the nonzero structure of the inverse of a matrix, assuming no fortuitous cancellation, is characterized in terms of its graph: the structure of B^{-1} is the transitive closure of B. In other words, $(B^{-1})_{ij} \neq 0$ if and only if there is a dipath $i \rightsquigarrow j$ in B.

We can immediately determine the structure of A^{-1} then. Assuming A is connected, and making use of its symmetric structure, there is a dipath between any two nodes. Hence A^{-1} is completely dense (first shown in [7]). We are thus interested in having $(\bar{L}\bar{L}^T)^{-1}_{ij} \neq 0$ for all i, j. Observe that:

$$(\bar{L}\bar{L}^T)^{-1}_{ij} = \sum_{k=1}^{n} (\bar{L}^{-1})_{ki} (\bar{L}^{-1})_{kj}$$

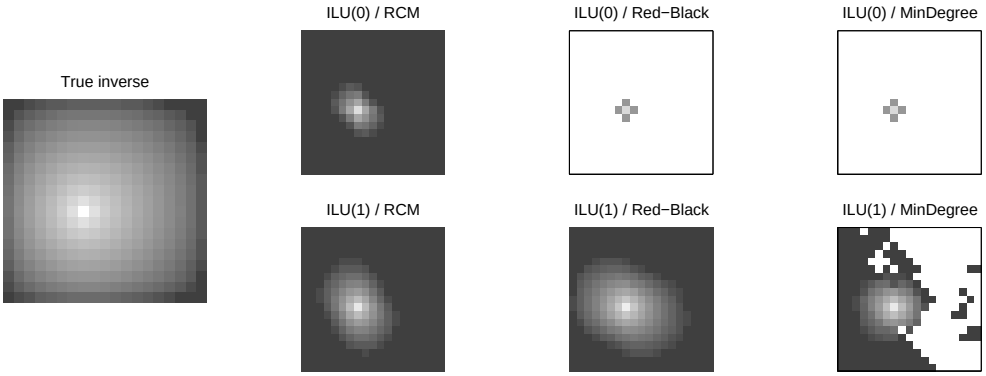
Thus $(\bar{L}\bar{L}^T)^{-1}_{ij} \neq 0$ if and only if there is some k such that $k \rightsquigarrow i$ and $k \rightsquigarrow j$ in $\bar{L}$ (assuming no fortuitous cancellation). Notice that since $\bar{L}$ is lower triangular, $u \to v$ in $\bar{L}$ implies that $u \geq v$, and so similarly $u \rightsquigarrow v$ implies $u \geq v$ (and in fact the the nodes on the dipath are monotonically decreasing). Therefore it is necessary that $n \rightsquigarrow i$ monotonically for all i since no $k > n$—and this is clearly sufficient too, since then $n \rightsquigarrow i$ and $n \rightsquigarrow j$ for all i, j, which implies $(\bar{L}\bar{L}^T)^{-1}_{ij} \neq 0$ for all i, j.

Thus $(\bar{L}\bar{L}^T)^{-1}$ is fully dense if and only if node $n-1$ is adjacent to n (in $\bar{L}$), node $n-2$ is adjacent to the set $\{n-1, n\}$, node $n-3$ is adjacent to $\{n-2, n-1, n\}$, etc. Imagine selecting the ordering (the labelling of the nodes) in reverse order: choosing which node will be n, then which will be $n-1$, etc. Recall that a graph traversal is a step-by-step selection of the graph's nodes such that at each step, the next node selected is adjacent to some previously selected node. Then what we have is:

THEOREM 2.1. *The implicit approximate inverse $(\bar{L}\bar{L}^T)^{-1}$ is fully dense if and only if the ordering is a "reversed graph traversal" (RGT) of $\bar{L}$, labelling the root n, the next node visited $n-1$, then the next one $n-2$, and so on.*

For IC(0), where the structure of $\bar{L}$ is the same as the structure of the lower triangle of A, we then are looking for a reversed graph traversal of A. A prime

FIG. 2.1. *True inverse and implicit approximate inverses for IC(0) and IC(1) of the 5-point Laplacian under different orderings. Lighter grey indicates larger entries, except pure white indicates exact zeros.*



example of this is Reverse Cuthill-McKee ordering, where a pseudo-peripheral node is chosen as the root, a special breadth-first traversal is made from the root, and the ordering is taken by reversing the traversal so the root is numbered n.

Parenthetically, this helps to explain why ILU can often give faster convergence than an (explicit) approximate inverse with the same number of nonzeros. A non-factored sparse approximate inverse is by definition far from dense—for an elliptic PDE this means that the preconditioner can't resolve low-frequency components of the error and is bound to have slow convergence. Factored sparse approximate inverses may have a dense product, but under more restrictive conditions than ILU: the last row/column of the lower/upper triangular factors respectively must be dense, which is often not the case.

Another result concerns Red-Black ordering on a 5-point grid. For IC(0) this is far from being an RGT—every black node is numbered higher than all its neighbours (which are red), and so there are no dipaths leading to a black node. However, for IC(1) each black node is connected in $\bar{L}$ to the eight nearest black nodes, and assuming the usual row-ordering of the black nodes, this does give an RGT. This is one reason why Red-Black orderings are only competitive at higher fill levels.

Unless special measures are taken, Minimum Degree, Nested Dissection, and similar orderings are typically not RGT's for low fill ILU. Figure 2.1 shows an example from a 5-point Laplacian on a 31×31 square grid. Each plot is of a column of the implicit approximate inverse (or the true inverse) appropriately matched to the PDE's domain—i.e. a 31^2 column vector reshaped into a grid function on the 31×31 mesh. These are in essence discretized views of "slices" of the Green's function or its IC approximations. The shading of the squares show how big the nonzero entries are. It's clear that even if the numerical factorization were optimal, Red-Black ordered IC(0) and the displayed low fill Minimum Degree ordered preconditioner must have serious difficulties, whereas the RCM ordered factorization and Red-Black ordered ILU(1) at least have the structural potential to be very effective.

However, orderings like Minimum Degree and Nested Dissection sometimes can give RGT's for low fill ILU. In particular, on square grids it is easy to determine a Nested Dissection which is a reversed graph traversal as well—figure 2.2 gives an example for a 7×7 grid. The key point is that each separator is itself ordered as

FIG. 2.2. *An example ordering on a 7×7 grid which is both a Nested Dissection and an RGT.*

1	7	4	49	22	28	25
3	8	6	48	24	29	27
2	9	5	47	23	30	26
19	20	21	46	42	41	40
10	18	13	45	31	39	34
12	17	15	44	33	38	36
11	16	14	43	32	37	35

an RGT with its root adjacent to the highest numbered node from the highest level separator possible. In this way there always exists a path in the underlying graph between any node and node n that does not pass through a lower numbered node than the starting node—in fact, a monotonic path that goes along the separators.

3. Inverse Factor Structure. For an even better simulation of the application of $A^{-1} = L^{-T}L^{-1}$, one could demand that the structure of the inverses of the incomplete factors match those of the true factors. Our intuition is that this will allow increased global coupling; it can be seen as an intermediate step between the full coupling of the exact factorization and the weakest global coupling discussed in the previous section.

Requiring that $\bar{L}^{-1}$ is as full as L^{-1} is a stronger condition than requiring just that $\bar{L}^{-T}\bar{L}^{-1}$ is as full as A^{-1}, and so this narrows down which graph traversals can be used. In particular, we will observe that this condition is not satisfied by RGT Nested Dissection orderings with low fill-in, providing some explanation why they are typically still not as effective as other orderings.

Recall that a depth-first search is a graph traversal where at each step the next node to be visited is chosen to be adjacent to the most recently visited node possible. The subgraph induced by the edges linking each successive node (the child) to the most recently visited adjacent node (its parent) is the associated depth-first search tree.

Define a deepening search to be a topological (or post-order) traversal of a depth-first search tree, i.e. a traversal of a depth-first search tree beginning at the root and visiting each parent node before any of its children. Alternatively put, at each step of a deepening search consider the connected components of unvisited nodes, and select a node from any of those components that is adjacent to the most recently visited node possible for that component. (A depth-first search is a particular type of deepening search.) This leads to the following result:

THEOREM 3.1. *Assume that there is no numerical cancellation and that the structure of the lower triangle of A is a subset of the structure of $\bar{L}$, which is in turn a subset of the structure of the exact lower triangular factor L. Then the structure of $\bar{L}^{-1}$ matches the structure of L^{-1} if and only if A has been ordered according to a reversed deepening search (RDS) of $\bar{L}$. In particular, any RDS of A fulfills this condition.*

Proof. Recall that the structure of the inverse B^{-1} of some matrix B is given by the transitive closure of the graph of B, which is the same as the transitive closure of the transitive reduction of B [11]. Since the structure of $\bar{L}$ is a subset of the structure of L, the structure of $\bar{L}^{-1}$ is a subset of the structure of L^{-1}, and is equal to it if and only if $\bar{L}$ contains the transitive reduction of L.

Further recall that the transitive reduction of L is the elimination tree of A[12]. Thus a necessary and sufficient condition for the result is that $\bar{L}$ contain the elimination tree of A.

Note that augmenting the structure of A with any subset of edges from L (i.e. adding "nonzeros" that are actually numerically zero) does not effect the structures of its triangular factors, and hence also does not effect its elimination tree. Thus we can simplify our argument by considering the structure of A augmented with the edges from $\bar{L}$ and $\bar{L}^T$, denoted by $\bar{A}$, instead of A. The necessary and sufficient condition is then reduced to $\bar{A}$ containing its own elimination tree.

Let $E(i)$ denote the set of nodes reachable from i through paths in $\bar{A}$ using only nodes $\{1, \ldots, i-1\}$, or in other words, the nodes reachable from i after all higher numbered nodes have been removed from the graph of $\bar{A}$. Partition each $E(i)$ according to the connected components it induces in the graph of $\bar{A}$, giving subsets $E_1(i), \ldots, E_{f(i)}(i)$.[2] So each $E_k(i)$ is a set of nodes numbered less than i that induces a connected subgraph of $\bar{A}$, with at least one node adjacent to i.

As characterized in [2] for example, the nonzeros in row i of L^{-1} correspond to the set $E(i)$. Observe that the elimination tree, the transitive reduction of L^{-1}, has an edge between i and the highest numbered node in each $E_k(i)$, by the following reasoning. Certainly these edges must be contained in the transitive reduction, since there is no equivalent path to them in L^{-1}: L^{-1} is triangular, so there can't be a path from i to the highest numbered node going through lower numbered nodes. These edges also suffice to give the full transitive closure, since it is clear that for each such edge (i, j) with $j = \max(E_k(i))$, we have $E(j) = E_k(i) \setminus \{j\}$, so inductively a path using just these exists from i to any node in $E(i)$.

Therefore the necessary and sufficient condition becomes: for each i, there is an edge in $\bar{A}$ from i to each of $\max(E_1(i)), \ldots, \max(E_{f(i)}(i))$. Imagine visiting the nodes in reverse order, starting at n and ending at 1. The condition means that within a connected component of unvisited nodes, the next node to visit must be a neighbour of the most recently visited node adjacent to the component. In other words, the condition is simply that the ordering must be a reversed deepening search of $\bar{L}$ (and the associated depth-first search tree is the elimination tree, consisting of the edges between each i and $\max(E_1(i)), \ldots, \max(E_{f(i)}(i))$.

Finally, observe that barring numerical cancellation adding more fill to $\bar{L}$ cannot reduce the structure of its inverse. Thus if the ordering is an RDS of the original A, i.e. of an IC(0) factor, then any higher fill incomplete factorizations $\bar{L}\bar{L}^T$ must satisfy the condition. □

An example of an RDS ordering that has already been investigated for ILU (and shown to be fairly effective) is the spiral ordering of [8]. This illustrates a special case of RDS orderings, where the associated depth-first search tree is actually a Hamiltonian path (i.e. it doesn't branch at all) and so the subdiagonal of A is all nonzero, which guarantees that L^{-1} will be a fully dense lower-triangular matrix.

Although Reverse Cuthill-McKee orderings are always RGT orderings, they typically are not RDS orderings, at least for low-fill IC. However, it should be noted that by rearranging each level set appropriately, they may always be converted into RDS orderings—perhaps at the expense of increasing the envelope, but that is generally inconsequential in modern sparse matrix packages for iterative methods.

[2] These may be called eliminated elements at step i of a complete factorization, and would correspond to the quotient nodes adjacent to i in the quotient graph[10].

On the other hand, though we observed in the previous section that seperator-based fill-reducing orderings such as Nested Dissection can be made into RGT orderings by reordering separators, it is in general impossible to similarly convert them to RDS orderings. Lower-level separators generally are adjacent to higher-level ones at the wrong places, namely the middle nodes rather than an end node. This may help to explain why these orderings, even if they are RGT's, generally perform poorly with low fill.[3].

4. Conclusion. The two theorems in this paper provide some understanding of already observed phenomena—the robust performance of RCM orderings, etc. However, we look forward to a constructive use in designing more effective orderings (and fill patterns) for IC. It certainly seems likely that any ordering considered today can be made into an RGT either by adding extra fill or reordering separators, for example, and many can be further made into an RDS. Though this is no guarantee of a good quality preconditioner, this at least removes structural impediments and paves the way for an effective incomplete numerical factorization.

5. Acknowledgements. We would like to thank the anonymous referees for their helpful suggestions in refocussing and clarifying the paper and for some instructive references.

REFERENCES

[1] M. Benzi, D. Szyld, and A. van Duin, *Orderings for Incomplete Factorization Preconditioning of Nonsymmetric Problems*, to appear in SIAM J. Sci. Com., 20 (1999).
[2] R. Bridson and W.-P. Tang, *Ordering, anisotropy and factored sparse approximate inverses*, SIAM J. Sci. Comput., 21 (1999), pp. 867–882.
[3] S. Clift, H. Simon, and W.-P. Tang, *Spectral ordering techniques for incomplete LU preconditioners for CG methods*, manuscript.
[4] S. Clift and W.-P. Tang, *Weighted graph based ordering techniques for preconditioned conjugate gradient methods*, BIT, 35 (1995), pp. 30–47.
[5] E. D'Azevedo, P. Forsyth, W.-P. Tang, *Towards a cost-effective ILU preconditioner with high level fill*, BIT, 32 (1992), pp. 442–463.
[6] S. Doi, *On parallelism and convergence of incomplete LU factorizations*, App. Num. Math., 7 (1991), pp. 417–436.
[7] I. Duff, A. Erisman, C. Gear, and J. Reid, *Sparsity structure and Gaussian elimination*, SIGNUM Newsletter, 23 (1988), pp. 2–8.
[8] I. Duff and G. Meurant, *The effect of ordering on preconditioned conjugate gradients*, BIT, 29 (1989), pp. 635–637.
[9] V. Eijkhout, *Analysis of parallel incomplete point factorizations*, Lin. Alg. Appl., 154–156 (1991), pp. 723–740.
[10] A. George and J. Liu, *Computer Solution of Large Sparse Positive Definite Systems*. Prentice-Hall, Englewood Cliffs, NJ, 1981.
[11] J. Gilbert, *Predicting structure in sparse matrix computations*, SIAM J. Matrix. Anal. Appl., 15 (1994), pp. 62–79.
[12] J. Liu, *The role of elimination trees in sparse factorization*, SIAM J. Matrix Anal. Appl., vol. 11, no. 1, pp. 134–172, January 1990.

[3] We note that for unsymmetric problems, however, the story can be quite different[1]