The Minimum Degree Heuristic and the Minimal Triangulation Process

Anne Berry[1], Pinar Heggernes[2], and Geneviève Simonet[3]

[1] LIMOS UMR CNRS 6158, Isima, 63173 Aubière, France.
berry@isima.fr
[2] Informatics, University of Bergen, N-5020 Bergen, Norway.
pinar@ii.uib.no
[3] LIRMM, 161 Rue Ada, 34392 Montpellier, France.
simonet@lirmm.fr

Abstract. The Minimum Degree Algorithm, one of the classical algorithms of sparse matrix computations, is a heuristic for computing a minimum triangulation of a graph. It is widely used as a component in every sparse matrix package, and it is known to produce triangulations with few fill edges in practice, although no theoretical bound or guarantee has been shown on its quality. Another interesting behavior of Minimum Degree observed in practice is that it often results in a minimal triangulation. Our goal in this paper is to examine the theoretical reasons behind this good performance. We give new invariants which partially explain the mechanisms underlying this heuristic. We show that Minimum Degree is in fact resilient to error, as even when an undesirable triangulating edge with respect to minimal triangulation is added at some step of the algorithm, at later steps the chances of adding only desirable edges remain intact. We also use our new insight to propose an improvement of this heuristic, which introduces at most as many fill edges as Minimum Degree but is guaranteed to yield a minimal triangulation.

1 Introduction

For the past forty years, problems arising from applications have given rise to challenges for graph theorists, and thus also to a wealth of graph-theoretic results. One of these is computing a minimum triangulation. Although the problem originally comes from the field of sparse matrix computations [20], it has applications in various areas of computer science.

Large sparse symmetric systems of equations arise in many areas of engineering, like the structural analysis of a car body, or the modeling of air flow around an airplane wing. The physical structure can often be thought of as covered by a mesh where each point is connected to a few other points, and the related sparse matrix can simply be regarded as an adjacency matrix of this mesh. Such systems are solved through standard methods of linear algebra, like Gaussian elimination, and during this process non-zero entries are inserted into cells of the matrix that originally held zeros, which increases both the storage

H.L. Bodlaender (Ed.): WG 2003, LNCS 2880, pp. 58–70, 2003.
© Springer-Verlag Berlin Heidelberg 2003

requirement and the time needed to solve the system. It was observed early that finding a good pivotal ordering of the matrix can reduce the amount of *fill* thus introduced: in 1957, Markowitz [14] introduced the idea behind the algorithm known today as Minimum Degree, choosing a pivot row and column at each step of the Gaussian elimination to locally minimize the product of the number of corresponding off-diagonal non-zeros. Tinney and Walker [22] later applied this idea to symmetric matrices, and Rose [20] developed a graph theoretical model of it.

As early as 1961, Parter [18] presented an algorithm, known as Elimination Game (EG), which simulates Gaussian elimination on graphs by repeatedly choosing a vertex and adding edges to make its neighborhood into a clique before removing it, thus introducing the connection between sparse matrices and graphs. In view of the results of [8], the class of graphs produced by EG is exactly the class of chordal graphs. Thus when the given graph is not chordal, Gaussian elimination and EG correspond to embedding it into a chordal graph by adding edges, a process called *triangulation*. As can be observed on the example in Figure 1, the number of fill edges in the resulting triangulation is heavily dependent on the order in which EG processes the vertices. This ordering of the graph corresponds to the pivotal ordering of the rows and columns in Gaussian elimination.

As mentioned above, it is of primary importance to add as few edges as possible when running EG. The corresponding problem is that of computing a *minimum triangulation*, which is NP-hard [23]. It is possible to compute in polynomial time a triangulation which is *minimal*, meaning that an inclusion-minimal set of edges is added [16], [21]. However, such a triangulation can be far from minimum, as can be seen from the example of Figure 1(b). As a result, researchers have resorted to heuristics, of which one of the most universally used and studied is *Minimum Degree (MD)*: this runs EG by choosing at each step a vertex of minimum degree in the transitory elimination graph, as illustrated by Figure 1(d). This algorithm is widely used in practice, and it is known to produce low fill triangulations. In addition, MD is also observed [5] to produce triangulations which are often minimal or close to minimal.

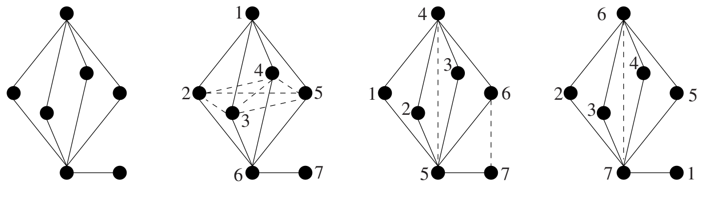

Fig. 1. (a) A graph G, and various triangulations of G by EG through the given orderings: (b) A minimal triangulation with $O(n^2)$ fill edges. (c) A non-minimal triangulation of G with less fill. (d) A minimum triangulation of G.

MD has given rise to a large amount of research with respect to improving the running time of its practical implementations, and the number of papers written on this subject is in the hundreds [1], [9]. However, very little is proved about its quality. It has in fact been analyzed theoretically only to a limited extent, which makes it difficult to gain control over this heuristic in order to improve it yet further, although recent research has been done on algorithms for low fill minimal triangulations [5], [6], [19].

In this paper, we use recent graph theoretical results on minimal triangulation and minimal separation to explain, at least in part, why MD yields such good results. In fact, it turns out that one of the reasons why MD works so well is that the EG algorithm is remarkably *robust*, in the sense that it is resilient to error: if at some step of the process an undesirable edge with respect to minimal triangulation is added, at later steps the chances of adding only desirable edges remain intact. One of our contributions here is that we use the insight we have gained on the mechanisms which govern EG and MD to propose an algorithmic process which improves the results obtained by MD, giving triangulations that are both minimal and have low fill.

The remainder of this extended abstract is organized as follows: Section 2 gives the graph theoretic background, introduces EG formally, and gives previous results on minimal separation and minimal triangulation. In Section 3, we give some new invariants for EG and MD, and explain why these algorithms are resilient to error, and why in many cases MD computes a minimal triangulation. Section 4 proposes a new algorithmic process in view of using the results of the paper to improve MD.

2 Preliminaries

Given a graph $G = (V, E)$, we denote $n = |V|$ and $m = |E|$. For any subset S of V, $G(S)$ denotes the subgraph of G induced by S, and $\mathrm{C}_G(S)$ denotes the set of connected components of $G(V - S)$. For the sake of simplicity, we will use informal notations such as $H = G + \{e\} + \{x\}$ when H is obtained from G by adding edge e and vertex x. For any vertex v of G, $N_G(v)$ denotes the neighborhood of v in G, and $N_G[v]$ denotes the set $N_G(v) \cup \{v\}$. For a given set of vertices $X \subset V$, $N_G(X) = \cup_{v \in X} N_G(v) - X$, and $N_G[X] = \cup_{v \in X} N_G(v) \cup X$. We will omit the subscripts when there is no ambiguity.

A vertex is *simplicial* if its neighborhood is a clique. We will say that we *saturate* a set of vertices X when we add to the graph all the edges necessary to make X into a clique. A graph is *chordal*, or *triangulated*, if it contains no chordless cycle of length ≥ 4. The set F of edges added to an arbitrary graph $G = (V, E)$ to obtain a triangulation $H = (V, E + F)$ of G is called a *fill*.

A function $\alpha : V \rightarrow \{1, 2, ..., n\}$ is called an *ordering* of the vertices of $G = (V, E)$, and (G, α) will denote a graph G the vertices of which are ordered according to α. We will use $\alpha = (v_1, v_2, ..., v_n)$, where $\alpha(v_i) = i$.

The algorithmic description of Elimination Game (EG) given below defines the notations we will use in the rest of this paper:

```
Algorithm Elimination Game
Input: A graph $G = (V, E)$, and an ordering $\alpha$ of the vertices in $G$.
Output: A triangulation $G^+_\alpha$ of $G$.
$G^1_\alpha \leftarrow G$;   $G^+_\alpha \leftarrow G$;
for $k = 1$ to $n$ do
    Let $F$ be the set of edges necessary to saturate $N_{G^k_\alpha}(v_k)$ in $G^k_\alpha$;
    $G^{k+1}_\alpha \leftarrow G^k_\alpha + F - \{v_k\}$;   $G^+_\alpha \leftarrow G^+_\alpha + F$;
```

According to the definition used in [16], we will call $G^k_\alpha(\{v_k, ..., v_n\} - N_{G^k_\alpha}[v_k])$ the *section graph at step k*. It is immediate from the description of EG that for any distinct $i, j \in [k, n]$, edge $v_i v_j$ is present in G^k_α iff there is a path in G between v_i and v_j (the path may have only one edge) all intermediate vertices of which are numbered $< k$ (i.e. do not belong to G^k_α). Similarly, the edges added during an execution of EG are well defined as $v_i v_j$ is an edge of G^+_α iff $v_i v_j$ is an edge of G or there is a path in G between v_i and v_j, all intermediate vertices of which have a number which is strictly smaller than $\min\{i, j\}$ [21].

The Minimum Degree (MD) heuristic is based on EG: it takes as input an unordered graph G, and computes an ordering α along with the corresponding triangulation G^+_α, by choosing at each step a vertex of minimum degree in G^k_α and numbering it as v_k.

Before proceeding to the next section, we will need some results on minimal separation. The notion of a minimal separator was introduced by Dirac [7] to characterize chordal graphs. Given a graph $G = (V, E)$, a vertex set $S \subset V$ is a *minimal separator* if $G(V - S)$ has at least two connected components C_1 and C_2 such that $N_G(C_1) = N_G(C_2) = S$ (C_1 and C_2 are called *full components*).

Characterization 1. ([7]) *A graph is chordal iff all its minimal separators are cliques.*

Recent research [11], [17], [2] has shown that minimal separators are central to minimal triangulations. The idea behind this is that forcing a graph into respecting Dirac's characterization will result into a minimal triangulation, by repeatedly choosing a not yet processed minimal separator and saturating it. We will need the definition of crossing separators, which characterize the separators that disappear when a saturation step of this process is executed:

Definition 1. ([11]) *Let S and S' be two minimal separators of G. S and S' are said to be* crossing *if there exist two connected components C_1, C_2 of $G(V - S)$, such that $S' \cap C_1 \neq \emptyset$ and $S' \cap C_2 \neq \emptyset$ (the crossing relation is symmetric).*

The saturation process described above can be generalized by choosing and simultaneously saturating a set of pairwise non-crossing minimal separators instead of a single minimal separator at each step, until a chordal graph is obtained. We will refer to this generalized process as the *Saturation Algorithm.* Given a set S of minimal separators of G, we will denote G_{S} the graph obtained from G by saturating all the separators belonging to S.

The following results from the works of Kloks, Kratsch and Spinrad [11] and Parra and Scheffler [17] provide a proof of this algorithm and will be used in Sections 3 and 4.

Theorem 1. ([11], [17]) *A graph $H = (V, E + F)$ is a minimal triangulation of $G = (V, E)$ iff there is a maximal set* S *of pairwise non-crossing minimal separators of G such that $H = G_{\mathrm{S}}$.*

Corollary 1. *A graph $H = (V, E+F)$ is a minimal triangulation of $G = (V, E)$ iff H is chordal and there is a set* S *of pairwise non-crossing minimal separators of G such that $H = G_{\mathrm{S}}$.*

Lemma 1. ([17]) *Let $G = (V, E)$ be a graph, let* S *and* S' *be sets of pairwise non-crossing minimal separators of G and G_{S}, respectively. Then* $\mathrm{S} \cup \mathrm{S}'$ *is a set of pairwise non-crossing minimal separators of G.*

Lemma 2. ([17]) *Let $G = (V, E)$, be a graph and* S *a set of pairwise non-crossing minimal separators of G. Then any minimal triangulation of G_{S} is a minimal triangulation of G.*

We will also use the notion of *substar*, which was introduced by Lekkerkerker and Boland [12] in connection with their characterization of chordal graphs.

Definition 2. ([12]) *Given a graph $G = (V, E)$ and a vertex x of G, the* substars *of x in G are the neighborhoods in G of the connected components of $G(V - N[x])$.*

In fact, although Lekkerkerker and Boland seemed not to be aware of this, the set of substars of some vertex x is exactly the set of minimal separators included in the neighborhood of x. LB-simpliciality of a vertex was defined in [3] in the following way for more convenient terminology.

Definition 3. *A vertex x is* LB-simplicial *if every substar of x is a clique.*

This was implicitly used by [12] to characterize chordal graphs as graphs such that every vertex is LB-simplicial, but the notion of substar is also very useful in the context of minimal triangulation, because it provides a fast and easy way of repeatedly finding sets of pairwise non-crossing minimal separators when running the Saturation Algorithm. This is fully described in [4], with in particular the following lemma:

Lemma 3. ([4]) *The substars of a vertex x in a graph G are pairwise non-crossing in G.*

3 Properties of EG Related to Minimal Triangulation

We will now examine how EG behaves with respect to the minimal separators of the graph which is to be triangulated.

3.1 EG and Partial Minimal Triangulation

We will first extend the definition of substar given in Section 2 to that of *substars* of (G, α).

Definition 4. *Given $(G = (V, E), \alpha)$, we will say that a set $S \subset V$ of vertices is a* substar of (G, α) *if there is some step k of EG such that S is a substar of v_k in G^k_α, which will be referred to as a* substar defined at step k *of EG.*

Clearly, during the execution of the EG, at each step k, making the currently processed vertex v_k simplicial will saturate these substars, and may also add some extraneous edges which do not have both endpoints in some common substar, so that two kinds of edges can be added:

- Edges which have both endpoints in some common substar defined at step k. We will refer to these edges as *substar fill edges.*
- Edges which do not have both endpoints in some common substar defined at step k. We will refer to these as *extraneous edges.*

In Section 2, we mentioned that in G and for a given vertex v, the substars of v are the minimal separators included in the neighborhood of v. One of our most interesting discoveries is that, in fact, *all* the substars defined by EG are minimal separators of the input graph, whether or not extraneous edges have been added at earlier steps. This fact is stated in Theorem 2, and its proof is based on the following lemma, which is interesting in its own right, as it describes a strong correspondence between the structures of G and G^k_α.

Lemma 4. *Given $(G = (V, E), \alpha)$ and an integer $k \in [1, n]$, let $G^k_\alpha = (V^k_\alpha, E^k_\alpha)$ and $S \subseteq V^k_\alpha$. The connected components of $G^k_\alpha(V^k_\alpha - S)$ are the sets $C \cap V^k_\alpha$ where C is a connected component of $G(V - S)$ such that $C \cap V^k_\alpha \neq \emptyset$, with the same neighborhoods, i.e. $N_{G^k_\alpha}(C \cap V^k_\alpha) = N_G(C)$.*

Proof. Let $S \subseteq V^k_\alpha$. We have to prove that $\mathrm{C}_{G^k_\alpha}(S) = \{C \cap V^k_\alpha,\ C \in \mathrm{C}_G(S) \mid C \cap V^k_\alpha \neq \emptyset\}$ and $\forall C \in \mathrm{C}_G(S)$ such that $C \cap V^k_\alpha \neq \emptyset$, $N_{G^k_\alpha}(C \cap V^k_\alpha) = N_G(C)$. Let $C \in \mathrm{C}_G(S)$ such that $C \cap V^k_\alpha \neq \emptyset$ and let $C' = C \cap V^k_\alpha$. Let us show that $C' \in \mathrm{C}_{G^k_\alpha}(S)$ and $N_{G^k_\alpha}(C') = N_G(C)$. $G^k_\alpha(C')$ is connected because for any vertices x and y in C', there is a path P in $G(C)$ between x and y, and by the properties of EG, the sub-sequence of P containing only the vertices belonging to V^k_α is a path in $G^k_\alpha(C')$ between x and y. Let us show that $N_{G^k_\alpha}(C') \subseteq N_G(C)$. Let $x \in N_{G^k_\alpha}(C')$ and $y \in C'$ such that $xy \in E^k_\alpha$. By the description of EG, there is a path in G between x and y all intermediate vertices of which belong to $V - V^k_\alpha$, and therefore belong to $V - S$ and consequently belong to C, so $x \in N_G(C)$. Let us show that $N_G(C) \subseteq N_{G^k_\alpha}(C')$. Let $x \in N_G(C)$ and $y \in C$ such that $xy \in E$. As $C' \neq \emptyset$, we may choose $z \in C'$. Let P be a path in $G(C)$ between y and z and let z' be the first vertex of P from y belonging to V^k_α. Vertex $z' \in C'$, and due to EG, $xz' \in E^k_\alpha$, so $x \in N_{G^k_\alpha}(C')$. Thus $N_{G^k_\alpha}(C') = N_G(C)$. As $C' \neq \emptyset$, $C' \subseteq V^k_\alpha - S$, $G^k_\alpha(C')$ is connected and $N_{G^k_\alpha}(C') = N_G(C) \subseteq S$, it follows that $C' \in \mathrm{C}_{G^k_\alpha}(S)$. Therefore, $\{C \cap V^k_\alpha,\ C \in \mathrm{C}_G(S) \mid C \cap V^k_\alpha \neq \emptyset\} \subseteq \mathrm{C}_{G^k_\alpha}(S)$. As $\cup_{C \in \mathrm{C}_G(S)}(C \cap V^k_\alpha) = V^k_\alpha - S$, the reverse inclusion holds too.

Theorem 2. *Every substar of* (G, α) *is a minimal separator of* G.

Proof. Let S be a substar defined at step k. S is a minimal separator of G^k_α, and by Lemma 4, there are at least as many full components of $G(V - S)$ as of $G^k_\alpha(V^k_\alpha - S)$. So S is also a minimal separator of G.

We are now ready to state our Main Theorem:

Theorem 3. *The set of substars of* (G, α) *forms a set of pairwise non-crossing minimal separators of* G.

Proof. Let S and S' be two substars of (G, α) defined at steps k and k' respectively, with $k \leq k'$. By Theorem 2, they are both minimal separators of G. Let us show that they are non-crossing in G. If $k = k'$ then they are non-crossing in G^k_α by Lemma 3, so they are non-crossing in G by Lemma 4. We suppose now that $k < k'$. S is a clique of G^{k+1}_α and $S' \subseteq V^{k+1}_\alpha$, so there is a connected component C of $G^{k+1}_\alpha(V^{k+1}_\alpha - S')$ such that $S \subseteq S' \cup C$. By Lemma 4, there is a connected component C' of $G(V - S')$ containing C, so $S \subseteq S' \cup C'$. Hence S and S' are non-crossing in G.

Note that this theorem does not guarantee that the set of substars defines a set of pairwise non-crossing minimal separators which is maximal. For instance, for any non complete graph G, if v_1 is a universal vertex of G then there is no substar of (G, α) whereas G has at least one minimal separator. A less trivial counterexample is given in Figure 2(b) of Example 1.

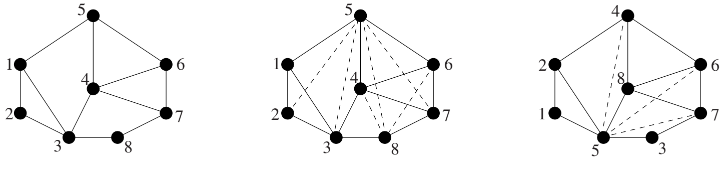

Fig. 2. Two executions of EG on the same graph with (b) an arbitrary ordering, and (c) an MD ordering.

Example 1. Figure 3.1 shows two executions of EG on graph G. A graph G and an ordering α are given in (a). The minimal separators of G are: $\{1, 3\}$, $\{3, 5\}$, $\{3, 7\}$, $\{1, 4, 6\}$, $\{1, 4, 7\}$, $\{1, 4, 8\}$, $\{3, 5, 7\}$, $\{4, 5, 7\}$, $\{4, 5, 8\}$, $\{4, 6, 8\}$, $\{3, 4, 6\}$.

We now demonstrate the execution of EG on (G, α) resulting in the graph shown in (b). **Step 1**: $N(1) = \{2, 3, 5\}$, $C_1 = \{4, 6, 7, 8\}, N(C_1) = \{3, 5\}$; substar fill edge 35 and extraneous edge 25 are added. **Step 2**: $N(2) = \{3, 5\}$; $C_2 = \{4, 6, 7, 8\}, N(C_2) = \{3, 5\}$; 2 is simplicial, so no edge is added. **Step 3**: $N(3) = \{4, 5, 8\}$, $C_3 = \{6, 7\}, N(C_3) = \{4, 5, 8\}$; substar fill edges 48 and 58 are added. **Step 4**: $N(4) = \{5, 6, 7, 8\}$; 4 is universal, so no component is defined; extraneous edges 57 and 68 are added; the remaining graph becomes a clique; no further edge is added. The set of substars of (G, α) is thus $\{\{3, 5\}, \{4, 5, 8\}\}$, which is a set of pairwise non-crossing minimal separators of G, but not a maximal one as $\{\{3, 5\}, \{4, 5, 8\}, \{4, 5, 7\}\}$ and $\{\{3, 5\}, \{4, 5, 8\}, \{4, 6, 8\}\}$ are also sets of pairwise non-crossing minimal separators of G. If only substar fill edges are preserved, a chordless cycle 5678 remains in the graph thus obtained. In order to saturate a maximal set of pairwise non-crossing minimal separators of G, $\{4, 5, 7\}$ or $\{4, 6, 8\}$ should also be saturated.

On the same graph, MD yields a minimal (and even minimum) triangulation, as shown in (c).

We would like to end this subsection with a discussion on the robustness of EG and MD regarding the process of defining non-crossing minimal separators of G. If, during the EG process, no extraneous edge is added, then the triangulation which is computed is minimal. However, due to Theorem 3, even when extraneous edges have been added, all substar fill edges added later belong to a set of pairwise non-crossing minimal separators of G, and therefore to a minimal triangulation of G. Thus if only a few extraneous edges are added during EG process, they will not destroy the property that all the substar fill edges are "useful" edges and that these few extraneous edges are the only "unnecessary" edges introduced. This makes EG a fault-tolerant procedure.

3.2 Conditions for EG and MD to Produce Minimal Triangulations

We have seen that EG does not necessarily compute a minimal triangulation. However, as mentioned earlier, MD is observed in practice to often produce orderings that are minimal or close to minimal, in addition to low fill. We will now give an explanation of this good behavior of MD through Lemma 5.

Lemma 5. *Let v_k be a vertex of minimum degree in G^k_α such that the union of all substars defined at step k is equal to a substar S defined at step k. Then only substar fill edges are added at step k.*

Proof. Assume by contradiction that there is an extraneous edge yz added at step k. Then y or z, say y, is not in S. $N_{G^k_\alpha}(y) \subseteq N_{G^k_\alpha}[v_k] - \{y, z\}$, so that $|N_{G^k_\alpha}(y)| < |N_{G^k_\alpha}(v_k)|$, which contradicts the fact that v_k is a vertex of minimum degree in G^k_α.

This result shows in particular that when the section graph is connected at some step of MD, then fill edges are added at that step only within the single substar defined. As is clear from the proof of this lemma, this is not the case in general for EG. Moreover, in most practical applications [15], the input graph is sparse; although no statistical result has been established on this, intuitively, a vertex x of minimum degree quite often defines only one substar, which usually corresponds to a connected section graph. MD run on sparse graphs thus stands a significantly higher chance of generating minimal triangulations than EG.

4 Improving the Results of Minimum Degree

Based on the results of the previous sections, we now present a variant of MD that produces a minimal triangulation and yields at most the same amount of fill, because it computes a minimal triangulation which is a subgraph of the graph computed by MD.

We have seen in Section 3 that the set S of non-crossing minimal separators of G defined by the set of all substars of (G, α) is not necessarily maximal. Thus if we remove all extraneous edges from G^+_α (or equivalently, add to G only the substar fill edges), the graph G_{S} thus obtained might fail to be triangulated. However, by Lemma 2, we know that any minimal triangulation H of G_{S} is a minimal triangulation of G. We thus propose a process which computes a minimal triangulation of the graph G_{S} obtained by running MD on G and removing the extraneous edges. In order to give the MD approach its chances, we will repeat the process of running MD on the partially triangulated graph obtained and removing the extraneous edges, until a chordal graph is obtained. However, at the beginning of each iteration, we will remove all LB-simplicial vertices, according to the following result:

Lemma 6. *Let $G = (V, E)$ be a graph, X the set of LB-simplicial vertices of G, and $G' = G(V - X) = (V', E')$. For any minimal triangulation $H' = (V', E' + F')$ of G', the graph $H = (V, E + F')$ is a a minimal triangulation of G.*

Proof. H is chordal because for any cycle C in H of length ≥ 4, either C is in H' and then C has at least one chord, or C contains a vertex x of X and then the neighbors of x in C induce a chord of C, as they belong to the substar of x in H defined by the component containing the other vertices of C, which is also substar of x in G (otherwise there would be an edge yz in F' such that y belongs to a connected component D of $G(V - N_G[x])$ and z does not belong to $N_G[D]$; by Theorem 1, y and z would be in a minimal separator of G', so they would be vertices of a chordless cycle C' in G', C' would have to contain two distinct vertices in $N_G(D)$ inducing a chord of C', a contradiction). So H is a a triangulation of G. It is a minimal one because for any chordal graph $H_1 = (V, E + F'_1)$ with $F'_1 \subseteq F'$, the graph $H'_1 = (V', E' + F'_1)$ is chordal too, so that $F'_1 = F'$.

Thus the LB-simplicial vertices can only cause MD to add extraneous edges, as well as unnecessarily increasing some vertex degrees, which justifies our systematically eliminating them from the graph at each step. Note that any step of the MD process tends to create LB-simplicial vertices, so removing these can make a significant difference regarding the quality of the fill obtained. We now present the new algorithm.

```
Algorithm Minimal Minimum Degree (MMD)
Input: A graph $G$.
Output: A minimal triangulation $H$ of $G$.
Run MD on $G$, which defines an ordering $\alpha$ and a set of substars S of $(G, \alpha)$;
$G' \leftarrow G_{\mathrm{S}}$; $H \leftarrow G_{\mathrm{S}}$;
while $G'$ is not chordal do
    Remove all LB-simplicial vertices from $G'$;
    Run MD on $G'$, which defines an ordering $\alpha$ and a set of substars S of $(G', \alpha)$;
    $G' \leftarrow G'_{\mathrm{S}}$; $H \leftarrow H_{\mathrm{S}}$;
```

It should be noted that graphs can be constructed such that no execution of MD can produce a minimal triangulation. Such an example is a graph consisting of two large cliques, connected by a single path of length ≥ 2. The graph is chordal, but vertices on the path will be chosen by MD at first steps, introducing unnecessary fill. MMD will not encounter any problem with that kind of graph, since the only minimal triangulation of a chordal graph is the graph itself.

We now prove that MMD gives a fill which is as least as good a that of MD, by showing that MMD yields a subgraph of the graph computed by MD.

Theorem 4. *Let* S *be the set of substars of* (G, α)*. Then any minimal triangulation* H *of* G_{S} *is a minimal triangulation of* G *which is a subgraph of* G_α^+*.*

Proof. In [4] the following property (P) is proved: Let S be a set of pairwise non-crossing minimal separators of G. Then $\forall S \in \mathrm{S}$, $\mathrm{C}_G(S) = \mathrm{C}_{G_{\mathrm{S}}}(S)$ and $\forall C \in \mathrm{C}_G(S)$, $N_G(C) = N_{G_{\mathrm{S}}}(C)$. We will use (P) in our proof. Let H be a minimal triangulation of G_{S}. By Theorem 3 and Lemma 2, H is a minimal triangulation of G. Let us show that H is a subgraph of G_α^+. By Theorem 1 and the fact that G_{S} is a subgraph of G_α^+, it is sufficient to show that any minimal separator of G_{S} is a clique of G_α^+. Let T be a minimal separator of G_{S}, let u and v be two vertices of T with $\alpha(u) < \alpha(v)$ and let $k = \alpha(u)$ (i.e. $u = v_k$). Let us show that uv is an edge of G_α^+, i.e. $v \in N_{G_\alpha^k}(v_k)$. We assume by contradiction that $v \notin N_{G_\alpha^k}(v_k)$. Let S be the substar of (G, α) defined at step k by the component containing v. S is a minimal separator of G_α^k separating the vertices v_k and v of T. By Lemma 4 S is also a minimal separator of G separating v_k and v, and by Theorem 3 and (P), it is a minimal separator of G_{S} separating v_k and v. So S and T are crossing in G_{S}. Hence S intersects at least two components of $\mathrm{C}_{G_{\mathrm{S}}}(T)$. But as $S \in \mathrm{S}$, S is a clique of G_{S} so that S cannot intersect two components of $\mathrm{C}_{G_{\mathrm{S}}}(T)$, a contradiction.

Theorem 5. *MMD computes a minimal triangulation of* G *which is a subgraph of* G_α^+*, where* α *is the MD ordering computed at the beginning of the algorithm.*

Proof. MMD terminates, because at the beginning of each step, the LB-simplicial vertices are all removed and this process cannot make a non LB-simplicial vertex become LB-simplicial. This is because a vertex is not LB-simplicial iff it belongs to a chordless cycle of length ≥ 4, and such a cycle remains after removing LB-simplicial vertices. Thus the vertex of minimum degree which is chosen first is not LB-simplicial; as a result, making it simplicial cannot fail to add at least one substar fill edge. Let us now prove MMD correctness. Let H be the output graph, let S_0 be the set of substars computed at the beginning of the algorithm and S' be the union of those computed in the **while**-loop. Thus $H = (G_{\mathrm{S}_0})_{\mathrm{S}'}$, G_{S_0} being the input graph of the **while**-loop. By Theorem 4 it is sufficient to show that H is a minimal triangulation of G_{S_0}, or more generally that for any input graph G' of the **while**-loop, the graph $G'_{\mathrm{S}'}$, where S' is the union of the sets of substars computed in the **while**-loop, is a minimal triangulation of G'. Let us prove this property by induction on the number p of iterations of the

while-loop before G' gets chordal. It trivially holds for $p = 0$, as in that case G' is chordal and S' is the empty set. We suppose that it holds when the number of iterations of the **while**-loop before G' gets chordal is p. Let us show that it holds when this number is $p + 1$. Let G'_1 be the graph obtained from G' by removing all its LB-simplicial vertices, let S'_1 be the set of substars computed at the first iteration of the **while**-loop and S'' be the union of those computed at the following iterations, and let G'' be the graph obtained at the end of the first iteration. Thus $G'' = (G'_1)_{\mathrm{S}'_1}$ and $\mathrm{S}' = \mathrm{S}'_1 \cup \mathrm{S}''$. By induction hypothesis, $G''_{\mathrm{S}''}$ is a minimal triangulation of G'', so by Theorem 3 and Lemma 2, it is also a minimal triangulation of G'_1. $G''_{\mathrm{S}''} = ((G'_1)_{\mathrm{S}'_1})_{\mathrm{S}''} = (G'_1)_{\mathrm{S}'_1 \cup \mathrm{S}''} = (G'_1)_{\mathrm{S}'}$. Thus the graphs $G'_{\mathrm{S}'}$ and $G''_{\mathrm{S}''}$ are obtained from G' and G'_1 respectively by adding the same set F' of edges, so by Lemma 6, $G'_{\mathrm{S}'}$ is a minimal triangulation of G', which completes the proof by induction and therefore the proof of MMD correctness.

With practical tests, we have compared MMD against MD with respect to the number of edges in the resulting triangulation. We have done a simple and straightforward implementation of MMD in Matlab, and we have run the tests both on randomly generated graphs of varying density, and on graphs from Matrix Market [15]. On each graph G, we first generated an MD ordering α. Then we compared the number of fill edges in G^+_α to the number of fill edges produced by MMD. As expected by Theorem 5, the number of fill edges resulting from MMD was always less than or equal to the number of fill edges resulting from MD. An interesting point is that on most graphs, MMD required only two iterations of the **while**-loop. The reduction in the number of fill edges achieved by MMD was not very large, because of MD's remarkably good performances. However, this improved algorithm may both give significant results on very large graphs and help researchers gain a better evaluation of how close MD gets to an optimal solution.

Finally, we would like to mention that existing MD codes in use are very fast although the theoretical running time of these implementations is not good [10]. In addition, other iterative procedures for computing minimal triangulations have been implemented to run fast in practice [19]. Thus we believe that, with some effort, MMD can be implemented to run efficiently in practice.

5 Conclusion

Our contributions in this paper are threefold. We have found new invariants for the Elimination Game process, proving that it defines and saturates a set of minimal separators of the input graph, with a remarkably fault-tolerant behavior. We have shown that Minimum Degree has additional properties that gives it a high chance of actually producing minimal triangulations, thereby explaining this practical behavior of MD. Finally, we have given a new algorithm that produces minimal triangulations with low fill, based on our findings about EG and MD.

Acknowledgment. We thank Elias Dahlhaus for interesting discussions on minimal triangulations.

References

1. P. Amestoy, T. A. Davis, and I. S. Duff. An approximate minimum degree ordering algorithm. *SIAM J. Matrix Anal. Appl.*, 17:886–905, 1996.
2. A. Berry. Désarticulation d'un graphe. *PhD Dissertation, LIRMM, Montpellier, December 1998.*
3. A. Berry, J. R. S. Blair, and P. Heggernes. Maximum Cardinality Search for Computing Minimal Triangulations. In L. Kucera, editor, *Graph Theoretical Concepts in Computer Science - WG 2002*, Springer Verlag, 2002. Lecture Notes in Computer Science.
4. A. Berry, J.-P. Bordat, P. Heggernes, G. Simonet, and Y. Villanger. A wide-range algorithm for minimal triangulation from an arbitrary ordering. Reports in Informatics 243, University of Bergen, Norway, 2003, and Research Report 02-200, LIRRM, Montpellier, France. *Submitted to Journal of Algorithms, November 2002.*
5. J. R. S. Blair, P. Heggernes, and J. A. Telle. A practical algorithm for making filled graphs minimal. *Theoretical Computer Science*, 250:124–141,2001.
6. E. Dahlhaus. Minimal elimination ordering inside a given chordal graph. In R. H. Möhring, editor, *Graph Theoretical Concepts in Computer Science*, pages 132–143. Springer Verlag, 1997. Lecture Notes in Computer Science 1335.
7. G.A. Dirac. On rigid circuit graphs. *Anh. Math. Sem. Univ. Hamburg*, 25:71–76, 1961.
8. D. R. Fulkerson and O. A. Gross. Incidence matrices and interval graphs. *Pacific Journal of Math.*, 15:835–855, 1965.
9. J. A. George and J. W. H. Liu. The evolution of the minimum degree ordering algorithm. *SIAM Review*, 31:1–19, 1989.
10. P. Heggernes, S. Eisenstat, G. Kumfert, and A. Pothen. The computational complexity of the Minimum Degree algorithm. *Proceedings of 14th Norwegian Computer Science Conference, NIK 2001, University of Tromsø, Norway.* Also available as ICASE Report 2001-42, NASA/CR-2001-211421, NASA Langley Research Center, USA.
11. T. Kloks, D. Kratsch, and J. Spinrad. On treewidth and minimum fill-in of asteroidal triple-free graphs. *Theoretical Computer Science*, 175:309–335, 1997.
12. C. G. Lekkerkerker and J. Ch. Boland. Representation of a finite graph by a set of intervals on the real line. *Fund. Math.*, 51:45–64, 1962.
13. J. W. H. Liu. Equivalent sparse matrix reorderings by elimination tree rotations. *SIAM J. Sci. Stat. Comput.*, 9:424–444, 1988.
14. H. M. Markowitz. The elimination form of the inverse and its application to linear programming. *Management Science*, 3:255–269, 1957.
15. Matrix Market *Web site:* `http://math.nist.gov/MatrixMarket/`
16. T. Ohtsuki, L. K. Cheung, and T. Fujisawa. Minimal triangulation of a graph and optimal pivoting order in a sparse matrix. *J. Math. Anal. Appl.*, 54:622–633, 1976.
17. A. Parra and P. Scheffler. How to use the minimal separators of a graph for its chordal triangulation. *Proceedings of the 22nd International Colloquium on Automata, Languages and Programming (ICALP '95), Lecture Notes in Computer Science*, 944:123–134, 1995.

18. S. Parter. The use of linear graphs in Gauss elimination. *SIAM Review,* 3:119–130, 1961.
19. B. Peyton. Minimal orderings revisited. *SIAM J. Matrix Anal. Appl.*, 23:271–294, 2001.
20. D. J. Rose. A graph-theoretic study of the numerical solution of sparse positive definite systems of linear equations. In R. C. Read, editor, *Graph Theory and Computing*, pages 183–217. Academic Press, 1972.
21. D. J. Rose, R. E. Tarjan, and G. S. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM J. Comput.*, 5:266–283, 1976.
22. W. F. Tinney and J. W. Walker. Direct solutions of sparse network equations by optimally ordered triangular factorization In Proceedings of the IEEE, 55:1801–1809, 1967.
23. M. Yannakakis. Computing the minimum fill-in is NP-complete. *SIAM J. Alg. Disc. Meth.*, 2:77–79, 1981.