ENGINEERING MECHANICS 2009
National Conference with International Participation
Svratka, Czech Republic, May 11 – 14, 2009

pp. 929–935
Paper #137

PERFORMANCE TESTS OF THE MINIMUM DEGREE ORDERING IMPLEMENTATION

P. Pařík*

Summary: *The minimum degree ordering is one of the most widely used algorithms to preorder a symmetric sparse matrix prior to numerical factorization. There are number of variants which try to reduce the computational complexity of the original algorithm while maintaining a reasonable ordering quality. An in-house finite element solver is used to test several minimum degree ordering algorithm variants to find the most suitable configuration for the use in the Finite Element Method. The performance results obtained and their assessments are presented along with the minimum degree ordering algorithm overview.*

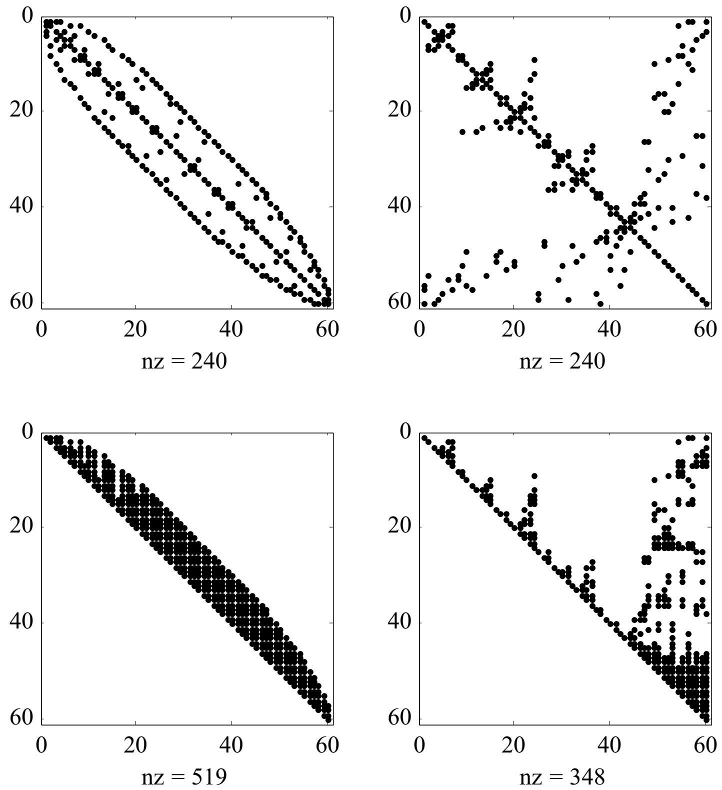
1. Úvod

Matice získané diskretizací metodou konečných prvků mohou být u složitých modelů velmi rozměrné. Tyto matice jsou naštěstí symetrické a řídké, tj. pouze malá část prvků matice je nenulová – často méně než jedno procento. Proto není nutné ukládat a zpracovávat všechny prvky matice (či přibližně polovinu prvků v případě symetrických matic), což je i pro současné nejvýkonnější počítače prakticky nemožné (např. relativně „malá“ symetrická matice řádu 10^5 by zabrala přibližně 40 gigabajtů paměťového prostoru).

Dalším problémem je tzv. zaplnění (fill-in) – změna původně nulových prvků matice na nenulové v průběhu řešení. V nejhorším případě je možné dosáhnout i úplného zaplnění všech původně nulových prvků, tj. faktorizace (dekompozice, triangularizace, eliminace) vede na plnou trojúhelníkovou matici (faktor). Proto je nezbytně nutné změnit pořadí řádků a sloupců matice (tedy prakticky přečíslovat odpovídající systém lineárních rovnic) tak, aby zaplnění po faktorizaci matice bylo co nejmenší a řídkost matice tak byla v co největší míře zachována.

Minimum degree ordering (Tinney & Walker, 1967) je jeden z nejrozšířenějších algoritmů pro přečíslování matic, protože vede na faktory s relativně nízkým zaplněním pro širokou škálu matic. Na obr. 1 je uveden příklad struktury matice a jejího faktoru po aplikování dvou přečíslovacích algoritmů (za povšimnutí stojí i počet nenulových prvků faktoru, který demonstruje výše uvedené tvrzení o kvalitě minimum degree algoritmu).

* Ing. Petr Pařík: Ústav termomechaniky Akademie věd ČR, v.v.i.; Dolejškova 5; 182 00 Praha 8; tel.: +420 266 053 441; e-mail: parik@it.cas.cz



Obr. 1 Příklad struktury matice po přečíslování (nahoře) a po následné faktorizaci (dole) Reverse Cuthill-McKee Ordering (vlevo) and Minimum Degree Ordering (vpravo)

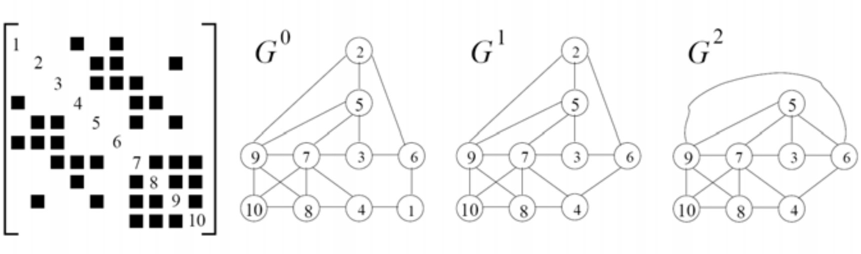
2. Minimum Degree Ordering

Účelem minimum degree ordering algoritmu je najít takové pořadí řádků a sloupců matice, aby následná numerická faktorizace matice vedla na faktor s nejmenším zaplněním, čímž se minimalizují nároky jak na uložení prvků matice, tak na čas výpočtu. Je třeba poznamenat, že nalezení nejmenšího zaplnění je NP-úplný problém, a proto je minimum degree ordering algoritmus založen na heuristice.

Původní minimum degree ordering algoritmus (Tinney & Walker, 1967) měl několik nevýhod, které byly časem různými způsoby překonány (George & Liu, 1989; Amestoy, Davis & Duff, 1996). Nejdůležitější zdokonalení jsou zběžně popsána dále.

2.1. Původní algoritmus

Původní minimum degree algoritmus (Tinney & Walker, 1967; George & Liu, 1989; Amestoy, Davis & Duff, 1996) je založen na tzv. eliminačním grafu (viz obr. 2), kde hrany grafu reprezentují nenulové prvky matice, tj. vrcholy grafu i a j jsou spojeny pouze je-li prvek na i-tém řádku a j-tém sloupci matice nenulový. Každý vrchol má navíc tzv. stupeň (degree), který je definován jako počet hran přilehlých k vrcholu (počet sousedících vrcholů).



Obr. 2 Příklad matice a jejího eliminačního grafu
(počáteční stav a stav po prvních dvou krocích algoritmu)

Na začátku algoritmu se podle nenulové struktury matice sestaví počáteční graf G^0, který je pak v průběhu algoritmu postupně redukován, dokud není prázdný. V každém kroku algoritmu je odebrán jeden vrchol včetně všech jeho přilehlých hran. Odebraný vrchol s hranami se poté nahradí doplněním hran mezi všemi dvojicemi vrcholů původně sousedícími s odebraným vrcholem. Počet přidaných hran, které představují zaplnění při jenom kroku numerické faktorizace, je úměrný stupni odebraného vrcholu, a proto se vždy odebírá vrchol s nejmenším stupněm (odtud jméno algoritmu). V každém kroku k této symbolické eliminace reprezentuje graf G^k nenulovou strukturu částečně faktorizované matice. Příklad matice a jejího eliminačního grafu je na obr. 2, kompletní grafy s vysvětlením lze najít v článku Amestoy, Davis & Duff, 1996.

2.2. Kvocienční graf

Přidávání nových hran reprezentujících zaplnění do eliminačního grafu znamená, že paměťové nároky grafu během symbolické eliminace rostou a nelze je předem odhadnout. To bylo vyřešeno zavedením tzv. kvocienčního grafu, jehož paměťové nároky nikdy nepřekročí velikost počátečního grafu G^0 (George & Liu, 1989; Amestoy, Davis & Duff, 1996).

Kvocienční graf je ale složitější než eliminační graf a výpočet stupňů je v něm poněkud komplikovaný. Pro výpočet stupně vrcholu v eliminačním grafu stačí spočítat velikost množiny jeho hran (tedy jejich počet); v eliminačním grafu je však často třeba pro výpočet stupně jednoho vrcholu nejprve najít sjednocení množin hran několika dalších vrcholů (s vyloučením duplicitních hran), což dělá výpočet stupňů zdaleka nejnáročnější částí algoritmu. S tímto problémem se lze vypořádat několika způsoby.

2.3. Supervrcholy

Počet výpočtů stupňů ve kvocienčním grafu je možné snížit zavedením tzv. supervrcholů (George & Liu, 1989; Amestoy, Davis & Duff, 1996). Vrcholy, které tvoří tzv. kliku (tj. úplně propojený podgraf), mají ekvivalentní množiny hran a také stejný stupeň, a proto kterýkoliv z těchto vrcholů může být vybrán jako další pivot. Tyto vrcholy tedy mohou být spojeny do jednoho „supervrcholu", při jehož odebrání z grafu se eliminují všechny vrcholy v něm obsažené najednou, což značně snižuje počet kroků symbolické eliminace.

Čas navíc potřebný pro detekci vrcholů vhodných pro spojení a vytváření supervrcholů je zanedbatelný v porovnání s časem ušetřeným redukcí počtu výpočtů stupňů.

2.4. Přibližné stupně

Složitost výpočtu stupňů ve kvocienčním grafu je možné snížit zavedením tzv. přibližných stupňů. Stupeň vrcholu může být aproximován jeho horní mezí, jejíž výpočet je jednodušší a rychlejší než přesný výpočet. Přečíslování získané za použití přibližných stupňů je obvykle méně kvalitní, tj. vede na větší zaplnění než při použití přesných stupňů.

Pro přibližný výpočet stupně vrcholu ve kvocienčním grafu existuje několik metod; v této práci byly implementovány a testovány tři metody (Gilbert, Moler & Schreiber, 1992; Amestoy, Davis & Duff, 1996).

3. Numerické testy

Numerické testy byly provedeny na finitním řešiči XFES systému PMD (Package for Machine Design – VAMET & ÚT AV, 2003), napsaném ve FORTRANu 77. Tento řešič využívá minimum degree ordering algoritmus a speciální maticové uložení pro efektivní řešení velkých konečněprvkových úloh.

Cílem testů bylo ověřit účinek dvou zlepšení algoritmu – metody přibližného výpočtu stupňů a zavedení supervrcholů – na rychlost a kvalitu přečíslování a stanovit nejlepší konfiguraci pro výpočty reálných konečněprvkových úloh. Je třeba poznamenat, že efektivita minimum degree ordering algoritmu je značně závislá na konkrétní implementaci.

Testy byly provedeny na souboru různých konečněprvkových úloh řádu od cca 10^2 neznámých do cca $2 \cdot 10^6$ neznámých.

3.1. Testované konfigurace

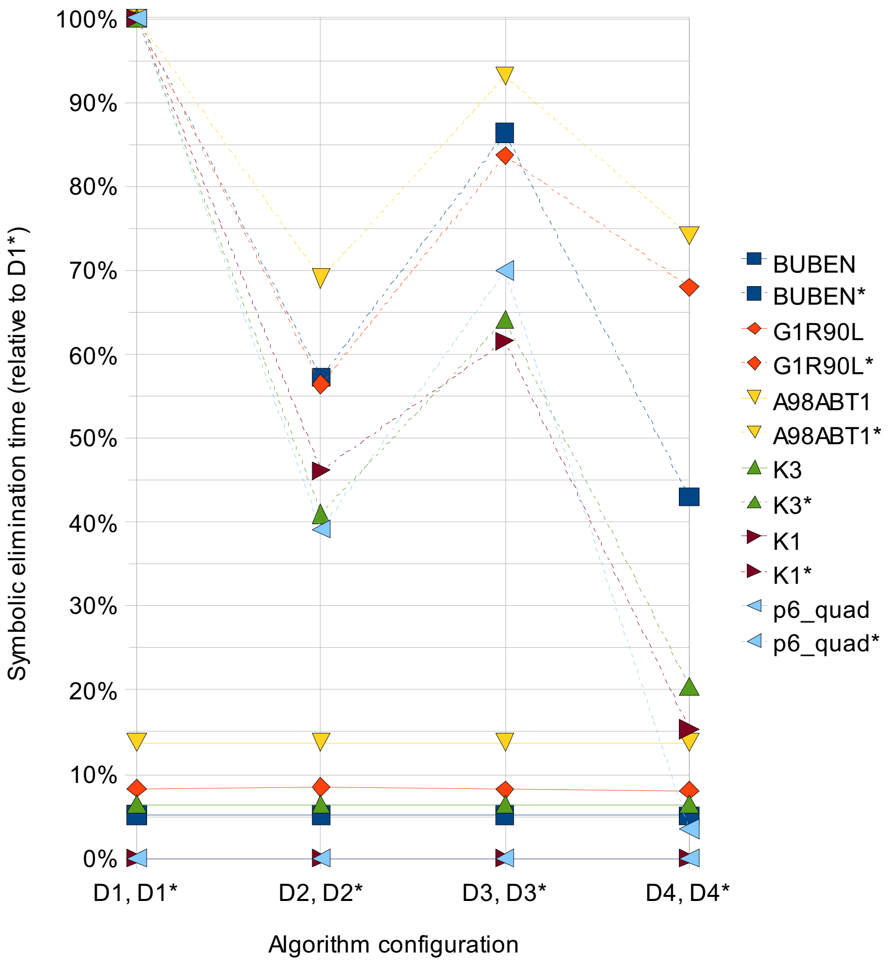
Celkem bylo testováno 8 konfigurací minimum degree ordering algoritmu: čtyři metody výpočtu stupňů s vypnutými nebo zapnutými supervrcholy. Testovaná implementace algoritmu je založena výhradně na kvocienčním grafu.

Konfigurace algoritmu – metody výpočtu stupňů:

- D1 – přesný výpočet (Tinney & Walker, 1967; George & Liu, 1989; Amestoy, Davis & Duff, 1996). Tato varianta by měla být časově nejnáročnější, ale produkovat nejlepší přečíslování.
- D2 – přibližný výpočet, který navrhli Amestoy, Davis a Duff (Amestoy, Davis & Duff, 1996). Tato varianta by měla produkovat téměř stejně kvalitní přečíslování jako D1, ale přitom být podstatně rychlejší.
- D3 – přibližný výpočet, který navrhli Ashcraft a Eisenstat (Amestoy, Davis & Duff, 1996). Tato varianta je ve skutečnosti speciální kombinací D1 a D4, měla by produkovat lepší přečíslování než D4 a přitom být rychlejší než D1.
- D4 – přibližný výpočet, který navrhli Gilbert, Moler a Schreiber (Gilbert, Moler & Schreiber, 1992; Amestoy, Davis & Duff, 1996). Tato varianta je nejjednodušší a měla by být nejrychlejší.

3.2. Rychlost přečíslování

Naměřená rychlost minimum degree ordering algoritmu je vynesena na obr. 3. Na ose x je konfigurace algoritmu (D1-D4, D1*-D4*) a na ose y výpočetní čas algoritmu relativně k referenční konfiguraci D1*, tj. přesný výpočet stupňů bez supervrcholů. Čáry spojující body náležící ke stejné úloze slouží pouze pro zlepšení čitelnosti a nemají žádný reálný význam. Výsledky získané bez supervrcholů jsou označeny hvězdičkou.



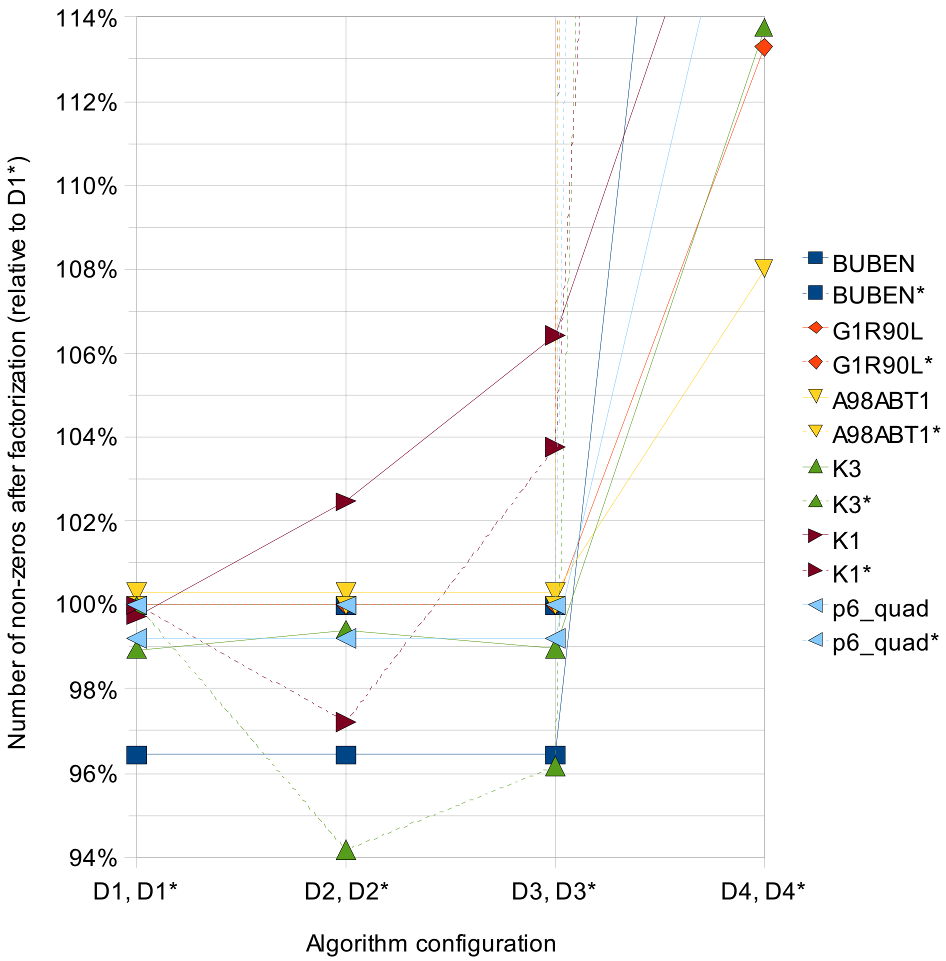
Obr. 3 Rychlost minimum degree ordering algoritmu (vybrané výsledky)

Rozdíly v rychlosti mezi konfiguracemi se supervrcholy (plné čáry) je pouze v řádech jednotek procent a rychlost je zhruba o 75% vyšší než u konfigurací bez supervrcholů. Při bližším zkoumání hodnot jsou konfigurace D3 a D4 o něco rychlejší než D1; konfigurace D4 je nejrychlejší. Konfigurace D2 se překvapivě jeví o něco pomalejší než D1, což je patrně způsobeno implementací algoritmu.

Rozdíly v rychlosti mezi konfiguracemi bez supervrcholů (čárkované čáry) je mnohem lépe viditelný. Přibližné stupně (konfigurace D2*-D4*) jsou o cca 10-90% rychlejší než přesné stupně (konfigurace D1*). Konfigurace D4* je většinou nejrychlejší, následuje D2*, potom D3*.

3.3. Kvalita přečíslování

Kvalita přečíslování získaná minimum degree ordering algoritmem je vynesena na obr. 4. Na ose x je konfigurace algoritmu (D1-D4, D1*-D4*) a na ose y kvalita přečíslování (měřená paměťovými nároky – počtem nenulových bloků matice po faktorizaci) relativně k referenční konfiguraci D1*, tj. přesný výpočet stupňů bez supervrcholů. Čáry spojující body náležící ke stejné úloze slouží pouze pro zlepšení čitelnosti a nemají žádný reálný význam. Výsledky získané bez supervrcholů jsou označeny hvězdičkou.



Obr. 4 Kvalita minimum degree ordering algoritmu (vybrané výsledky)

Rozdíly v kvalitě přečíslování mezi konfiguracemi se pohybují v rozmezí ±5%, nicméně u většiny úloh dávají konfigurace se supervrcholy D1-D3 (plné čáry) a bez supervrcholů D1*-D3* (čárkované čáry) prakticky stejně kvalitní přečíslování. U konfigurace D4 značně klesá kvalita přečíslování, a to úměrně velikosti úlohy; u konfigurace D4* je pokles kvality ještě řádově větší. Některé konfigurace mohou u některých úloh dosáhnout lepšího přečíslování, ale jedná se pouze o jednotlivé případy, v řádu jednotek procent, bez vysledovatelného trendu.

4. Závěr

Byla otestována, změřena, porovnána a zhodnocena implementace minimum degree ordering algoritmu na souboru několika desítek konečněprvkových úloh.

Rozšíření kvocienčního grafu o supervrcholy bylo potvrzeno jako velmi důležité zdokonalení minimum degree ordering algoritmu, protože kromě značného zvýšení rychlosti přečíslování také někdy mírně zvyšuje jeho kvalitu. Důvodem je, že hromadná eliminace vrcholů se stejným stupněm vede na menší zaplnění než jejich eliminace v „náhodném" pořadí, jako je tomu u původního minimum degree ordering algoritmu. Z tohoto důvodu bude závěrečné hodnocení provedeno pouze pro konfigurace se supervrcholy.

Metoda přibližného výpočtu stupňů, kterou navrhli Gilbert, Moler a Schreiber (konfigurace D4), byla potvrzena jako nejrychlejší, nicméně kvalita přečíslování byla v porovnání s ostatními konfiguracemi řádově nižší.

Metoda přibližného výpočtu stupňů, kterou navrhli Ashcraft a Eisenstat (konfigurace D3), obvykle dává poměrně dobré přečíslování, nicméně je pomalejší než ostatní přibližné metody (konfigurace D2 a D4).

Metoda přibližného výpočtu stupňů, kterou navrhli Amestoy, Davis a Duff (konfigurace D2), byla potvrzena jako kvalitní a zároveň rychlá aproximace přesných stupňů.

Z uvedených výsledků dále vyplývá, že v případě testované implementace minimum degree ordering algoritmu není nutné na úkor požadované maximální rychlosti přečíslování potencionálně snižovat jeho kvalitu zaváděním přibližného výpočtu stupňů a tím zvyšovat paměťové nároky na řešení. Vysoká optimalizace testované implementace algoritmu (konkrétně konfigurace D1) na programové úrovni umožňuje provádět přečíslování matic, které je rychlé i pro úlohy s 10^6 a více neznámými a zároveň produkuje minimální zaplnění při faktorizaci.

5. Poděkování

Tato práce vznikla za podpory grantu GA ČR 101/09/1630.

6. Literatura

Tinney, W. F. & Walker, J. W. (1967) Direct solutions of sparse network equations by optimally ordered triangular factorization. *Proc. of the IEEE*, Vol. 55, pp. 1801-1809.

George, A. & Liu, J. W. H. (1989) The evolution of the minimum degree-ordering algorithm. *SIAM Review*, Vol. 31, pp. 1-19.

Amestoy, P. R., Davis, T. A. & Duff, I. S. (1996) An approximate minimum degree ordering algorithm. *SIAM J. Matrix Analysis & Application*, Vol. 17, No. 4, pp. 886-905.

Gilbert, J. R., Moler, C. & Schreiber, R. (1992) Sparse matrices in MATLAB: design and implementation. *SIAM J. Matrix Analysis & Application*, Vol. 13, pp. 333-356.

VAMET & Ústav termomechaniky AV ČR (2003) *PMD verze f77.9*.