

Miniprojekt: Ekvationslösning

Stora matriser

I många realistiska tillämpningar måste man lösa *stora* linjära ekvationsystem, med tusentals obekanta. Det är i dessa fall som effektiva algoritmer blir viktiga att använda.

Som exempel ska ni här räkna på ett komplicerat fackverk: en modell av Eiffeltornet. Ett fackverk består av stänger förenade genom leder i ett antal noder. Ni ska beräkna deformationen av fackverket när noderna belastas av yttre krafter. Ekvationerna för deformationen härleds i hållfastläran, och baseras på att förskjutningarna i varje nod är små, och att Hookes lag gäller för förlängningen av varje stång.

I slutändan får man ett linjärt ekvationssystem på formen $Ax = b$. När antalet noder i fackverket är N kommer antalet obekanta vara $2N$ och $A \in \mathbb{R}^{2N \times 2N}$. Matrisen A brukar kallas *styhetsmatrisen*. Högerledet b innehåller de givna yttre krafterna som verkar på noderna,

$$b = \left(F_1^x, F_1^y, F_2^x, F_2^y, \dots, F_N^x, F_N^y \right)^T, \quad b \in \mathbb{R}^{2N},$$

där $F_j = (F_j^x, F_j^y)^T$ är kraften i nod j . Lösningen x innehåller de resulterande (obekanta) förskjutningarna,

$$x = \left(\Delta x_1, \Delta y_1, \Delta x_2, \Delta y_2, \dots, \Delta x_N, \Delta y_N \right)^T, \quad x \in \mathbb{R}^{2N}.$$

Här är alltså $(\Delta x_j, \Delta y_j)^T$ förskjutningen av nod j när fackverket belastas med krafterna i b .

På hemsidan finns filerna `eiffel1.mat`, `eiffel2.mat`, `eiffel3.mat` och `eiffel4.mat`. De innehåller fyra olika modeller av Eiffeltornet med växande detaljrikedom ($N = 261, 399, 561, 1592$). Varje modell består av nodkoordinater i vektorerna `xnod`, `ynod`, stångindex i matrisen `bars` (används bara för plotningen) och styvhetsmatrisen `A`.

a) Ladda in en av modellerna i MATLAB med kommandot `load`. Hämta funktionsfilen `trussplot.m` från kurshemsidan och anropa den med `trussplot(xnod, ynod, bars)` för att plotta tornet. Välj nu en av noderna och belast den med en kraft rakt högerut med beloppet ett. (Sätt $F_j^x = 1$ för något j , och resten av elementen i b lika med noll, dvs i MATLAB tex: `j=58; b=zeros(2*N,1); b(j*2-1)=1;`) Lös systemet $Ax = b$ med backslash för att få fram förskjutningarna i alla punkter. Beräkna de nya koordinaterna för det belastade tornet, $x_j^{ny} = x_j + \Delta x_j$, etc.:

```
xny = xnod + x(1:2:end); yny = ynod + x(2:2:end);
```

Plotta det belastade tornet. Använd `hold on` för att plotta de två tornen ovanpå varandra i samma figur. Markera vilken nod ni valt.

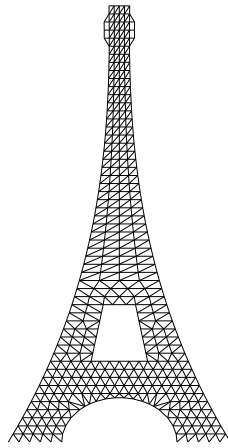


Figure 1: Modellen i `eiffel3.mat`, med 399 noder (798 obekanta).

b) Ni ska räkna ut i vilka noder fackverket är mest respektive minst känslig för horisontell belastning. Börja med den minsta modellen, `eiffel1.mat`. Tag en nod i taget. Belasta den med samma kraft som i a) ovan och räkna ut resulterande förskjutningar x . Notera storleken på förskjutningen, dvs $\|x\|$. Fortsätt med nästa nod, etc. Systematisera beräkningarna med en forslunga i ert Matlab-program och spara storleken på förskjutningen för varje nod. Ta sedan reda på vilken som nod ger störst respektive minst förskjutning.

Plotta tornet med `trussplot` och markera de mest och minst känsliga noderna i figuren.

Ni kommer alltså att behöva lösa samma stora linjära ekvationssystem med många olika högerled (N stycken). När matrisen är stor, vilket är fallet för de större modellerna, blir detta mycket tidskrävande. Optimera programmet genom att använda LU-faktorisering av A (MATLAB-kommandot `lu`). Uppskatta tidsvinsten av detta. Kan ni köra även de större modellerna?

c) När en matris är gles kan betydligt effektivare metoder än vanlig gausseliminering användas för att lösa ekvationssystemet. Använd `spy(A)` för att studera styvhetsmatrisens struktur. Vad kan man säga om den?

Genom att tala om för MATLAB att matrisen är gles kommer bättre metoder automatiskt användas när backslash anropas. Detta kan ni enkelt göra här genom att skriva `A=sparse(A)`. Gå igenom beräkningarna i c) igen. Hur stor tidsvinst gör man i detta fall genom att låta MATLAB använda metoder för glesa matriser? Prova med och utan LU-faktorisering.