Physikbasierte Modellierung und Simulation
Assignment 8

Present your solutions for this sheet in the exercise on Thursday, January 19, 2017.
In this exercise, you will solve a system of linear equations using the conjugate gradient method. This is
not a programming task! Solve it on a piece of paper.

8.1 Solve a system of linear equations (80 points)

Solve the system \( Ax = b \) (see below) for \( x \), starting from the zero vector. You should be able to see the
solution of this system by just looking at it, so you can check whether you are on the right track. You
can stop iterating as soon as the residual falls below \( 10^{-9} \).

\[
A = \begin{pmatrix}
1 & 1 & 4 \\
1 & 1 & 1 \\
4 & 1 & 1
\end{pmatrix}
\]

\[
b = (6 \ 6 \ 6)^T
\]

Solution

\[
\vec{x}_0 = (0 \ 0 \ 0)^T
\]

\[
\vec{r}_0 = (6 \ 6 \ 6)^T
\]

\[
\vec{d}_0 = (6 \ 6 \ 6)^T
\]

Step 1:

\[
\alpha_0 = 2 \cdot 10^{-1}
\]

\[
\vec{x}_1 = (1.2 \ 1.2 \ 1.2)^T
\]

\[
\vec{r}_1 = (-1.2 \ 2.4 \ -1.2)^T
\]

\[
||\vec{r}_1|| = ||Ax_1 - b|| = 2.93938769134
\]

\[
\beta_1 = 8 \cdot 10^{-2}
\]

\[
\vec{d}_1 = (-7.2 \cdot 10^{-1} \ 2.88 \ -7.2 \cdot 10^{-1})^T
\]
Step 2:
\[ \alpha_1 = 1.6666666667 \]
\[ \vec{x}_2 = \begin{pmatrix} -4.4408920985 \cdot 10^{-16} & 6.0 & -4.4408920985 \cdot 10^{-16} \end{pmatrix}^T \]
\[ \vec{r}_2 = \begin{pmatrix} 1.7763568394 \cdot 10^{-15} & 4.4408920985 \cdot 10^{-16} & 1.7763568394 \cdot 10^{-15} \end{pmatrix}^T \]
\[ \|\vec{r}_2\| = \|A\vec{x}_2 - \vec{b}\| = 2.51214793389 \cdot 10^{-15} \]

(All results within numerical precision.)

8.2 Analysis (20 points)

Answer the following questions:

a) What is the maximal number of iterations a conjugate gradient algorithm may take for a 3-by-3 system?

b) What is the principal advantage of the conjugate gradient algorithm compared to, for example, Gauss elimination?

c) What conditions does the matrix \( A \) have to fulfill?

d) Imagine you want to solve a system that does not have these properties. How could you still make use of the conjugate gradient method to solve it?

Solution

a) 3 steps (or \( n \) steps for an \( n \)-by-\( n \) system)

b) The matrix \( A \) does not need to be given explicitly; it suffices to be able to compute matrix-vector products. This is useful when the matrix is sparse or when matrix-vector products can be computed in a more efficient way, as for some geometrical projections (which can be computed efficiently on the GPU) or special transforms like the Fourier transform. Also, because conjugate gradients is an iterative method, intermediate results are available, and early exit is possible as soon as the residual gets small enough.

c) \( A \) has to be symmetric and positive definite. This implies that it is square and has full rank, and the solution is uniquely determined.

d) You can still apply the conjugate gradient algorithm to the “normal equations” \( A^TA\vec{x} = A^T\vec{b} \) to obtain a least-squares approximation to \( A\vec{x} = \vec{b} \).

\[ \text{http://graphics.tu-bs.de/teaching/lectures/ws1617/pbm/} \]