Combinatorial Problems in Scientific Computing

Introduction

Uwe Naumann

Informatik 12:
Software and Tools for Computational Engineering (STCE)

RWTH Aachen University

Contents

Outline

Motivation

Newton's Method

SuiteSparse Matrix Collection

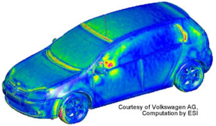
Admin

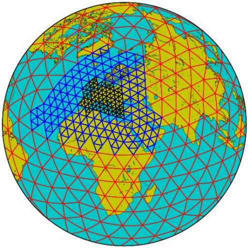
Summary and Next Steps

Motivation

Sparsity and Structure in Numerical Simulations

Scientific computing uses mathematical modelling to numerically simulate / predict / optimize reality, e.g, automotive engineering, geophysics, ...





Numerical simulations are implemented as computer programs.
Sparsity is induced by missing data dependences, e.g, today's temperature in Hong Kong is probably independent of yesterday's temperature in Aachen, ...
Numerical programs exhibit structure, e.g, iteration, recursion, ...

Efficiency and robustness requires solution of various combinatrial problems (ordering, coloring, elimination, ...) on sparse matrices and graphs.

Key ingredients of scientific computing are the numerical approximation of solutions of

- systems of nonlinear equations

$$F(\mathbf{x}) = 0, \quad F : \boldsymbol{R}^n \rightarrow \boldsymbol{R}^n$$

  for given implementations of the residual $\mathbf{y} = F(\mathbf{x})$ and

- convex unconstrained nonlinear optimization problems

$$\text{argmin}_{\mathbf{x}} f(\mathbf{x}), \quad f : \boldsymbol{R}^n \rightarrow \boldsymbol{R}$$

  for given implementations of the objective $y = f(\mathbf{x})$

by Newton's method. The method uses differentiation and linear algebra $\Rightarrow$ plenty of structure and sparsity to exploit.

Motivation

Ingredients of Newton's Method

The content of this course inspired by the ingredients of Newton's method, namely

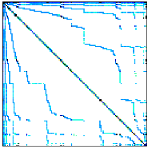
- solution of systems of linear equations involving
  - sparse matrix-vector products
  - sparse matrix chain products
  - sparse direct linear solvers
- algorithmic differentiation involving
  - accumulation of sparse Jacobians and Hessians
  - data-flow reversal

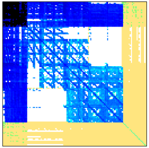
The discussion of these topics will be based on introductions to essential background of Newton's method and of algorithmic differentiation.

Motivation

SuiteSparse Matrix Collection

We use generic residuals $F(\mathbf{x})[=0]$ and objectives $[\min_{\mathbf{x}}]f(\mathbf{x})$ replicating given sparsity patterns, e.g, provided by the SuiteSparse Matrix Collection; see sparse.tamu.edu.





... not a virus!

A dense matrix has n^2 nonzero entries (obviously, ...).

A matrix is sparse if the number of nonzero entries is $O(n)$.

Consider linear system $A \cdot \mathbf{x} = \mathbf{b}$ with regular system matrix $A \in \mathbf{R}^{n \times n}$ and solved as

$$L \cdot \underbrace{U \cdot \mathbf{x}}_{:=\mathbf{z}} = \mathbf{b}$$

by

- $A = L \cdot U$ decomposition at cost of $O(n^3)$
- forward substitution for $L \cdot \mathbf{z} = \mathbf{b}$ at cost of $O(n^2)$
- backward substitution for $U \cdot \mathbf{x} = \mathbf{z}$ at cost of $O(n^2)$

Obviously, treatment of a lower triangular A as dense, e.g,

$$\begin{pmatrix} 1 & 0 \\ 2 & 3 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 2 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 3 \end{pmatrix}$$

is suboptimal.

Householder Reflection of $\mathbf{a} \in \boldsymbol{R}^n$:

$$\mathbf{v} = \mathbf{a} + \text{sign}(a_0) \cdot \|\mathbf{a}\| \cdot \mathbf{e}_0$$
$$H = I - 2 \cdot \frac{\mathbf{v} \cdot \mathbf{v}^T}{\mathbf{v}^T \cdot \mathbf{v}}$$
$$\mathbf{a} = H \cdot \mathbf{a}$$

better at cost of $O(n)$ (instead of $O(n^2)$)

$$\mathbf{v} = \mathbf{a} + \text{sign}(a_0) \cdot \|\mathbf{a}\| \cdot \mathbf{e}_0$$
$$\mathbf{a} = \mathbf{a} - 2 \cdot \frac{\mathbf{v}^T \cdot \mathbf{a}}{\mathbf{v}^T \cdot \mathbf{v}} \cdot \mathbf{v}$$

- associativity of matrix multiplication

$$\left(\begin{pmatrix}1 & 2\\3 & 4\end{pmatrix}\cdot\begin{pmatrix}1 & 2\\3 & 4\end{pmatrix}\right)\cdot\begin{pmatrix}1\\3\end{pmatrix}=\begin{pmatrix}1 & 2\\3 & 4\end{pmatrix}\cdot\left(\begin{pmatrix}1 & 2\\3 & 4\end{pmatrix}\cdot\begin{pmatrix}1\\3\end{pmatrix}\right)$$

- distributivity of real arithmetic

$$a_1\cdot b_1\cdot c\cdot d+a_2\cdot b_2\cdot c\cdot d=(a_1\cdot b_1+a_2\cdot b_2)\cdot c\cdot d$$

- structural orthogonality of vectors

$$\left(\begin{pmatrix}1\\0\end{pmatrix}+\begin{pmatrix}0\\2\end{pmatrix}\right)\cdot 3$$

carries same information as

$$\left(\begin{pmatrix}1\\0\end{pmatrix}\cdot 3,\begin{pmatrix}0\\2\end{pmatrix}\cdot 3\right)$$

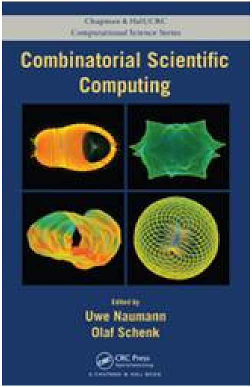
in less storage (and potentially at lower computational cost).

Outline

Admin

Further Details on RWTH Moodle

- The lecture (one per week) covers general story including theory and algorithms.
- The tutorial (one per week) supports the lecture by discussing both pen&paper as well as programming exercises.
- Learning material consists of
  - videos, slides, code, reference solutions for tutorial exercises, pointers into literature, e.g, ...



- Registration (RWTH online) “buys” you access to the RWTH Moodle page of the course.

Summary

... of this lecture

- Scientific computing is an essential element of mankind's toolbox for understanding the world.
- Combinatorial problems occur in the context of many scientific computing methods.
- For example, they can be posed as ordering, coloring or elimination problems on sparse matrices and/or graphs.
- Knowledge of algorithms for the solution of these problems enables efficient and robust implementation of numerical simulation methods.
- You should attend this course ... ;-)

Outline

Summary

-

Next Steps

-
- Continue the course to find out more ...