A Compiler for Linear Algebra Operations

Henrik Barthels, M.Sc.

HPAC High Performance and Automatic Computing
RWTH AACHEN UNIVERSITY

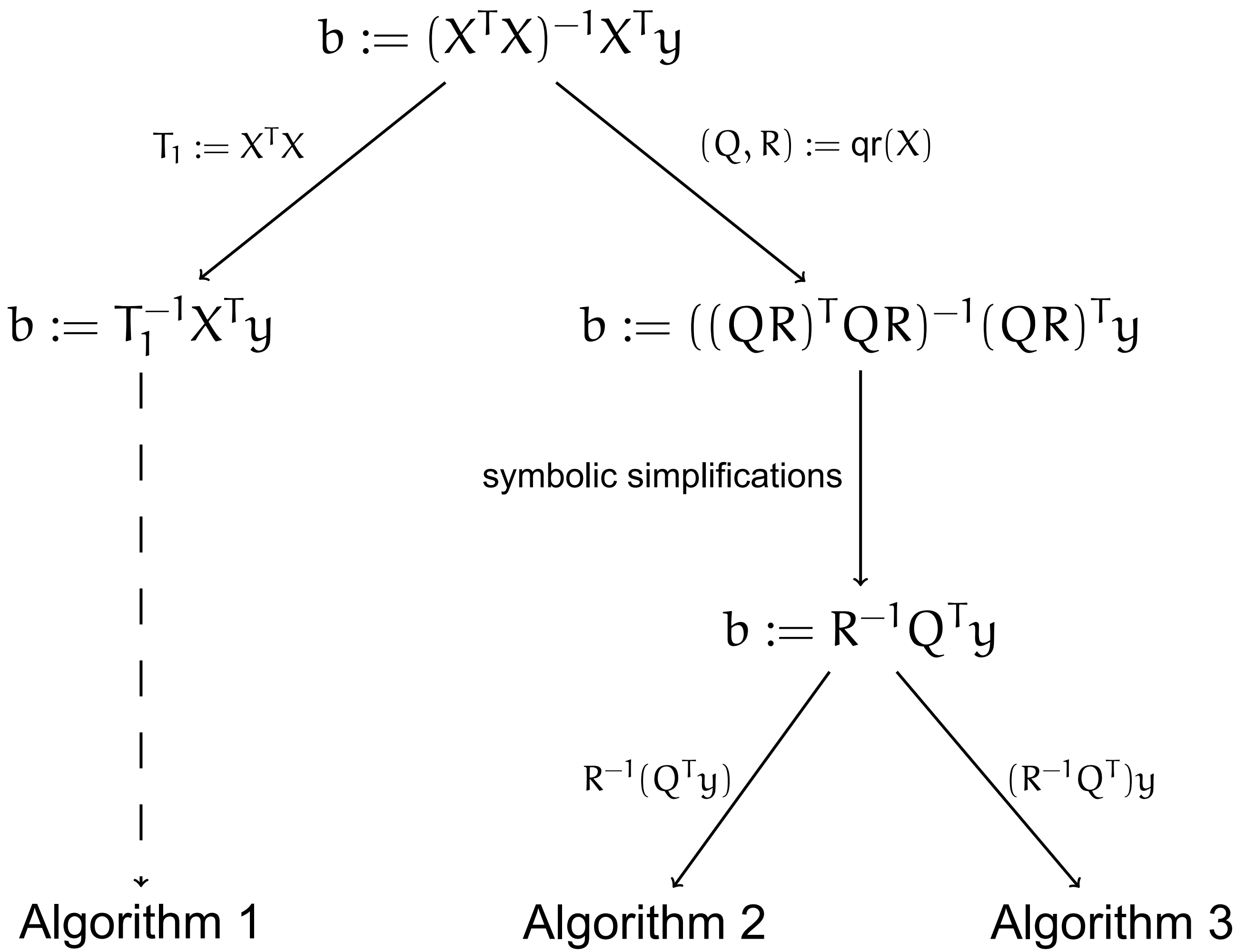
Introduction

- How to compute the following expressions?

$$b := (X^T X)^{-1} X^T y$$
$$x := \left(A^{-T} B^T B A^{-1} + R^T [\Lambda(Rz)] R\right)^{-1} A^{-T} B^T B A^{-1} y$$
$$x_{ij} := A_i B c_j$$

- Matlab is easy to use, but performance is usually suboptimal.

Introduction



Input Grammar

$$z := (X^T M^{-1} X)^{-1} X^T M^{-1} y$$

$M \in \mathbb{R}^{2000 \times 2000}$

M is symmetric positive definite.

$X \in \mathbb{R}^{2000 \times 1000}$

$y \in \mathbb{R}^{2000}$

Instruction Set

BLAS [DDC+90]

- $y \leftarrow Ax + y$
- $C \leftarrow AB + C$
- $B \leftarrow A^{-1}B$
- $\dots$

LAPACK [AB+99]

- Matrix factorizations.
- Eigensolvers.
- Solvers for linear systems with specific properties.

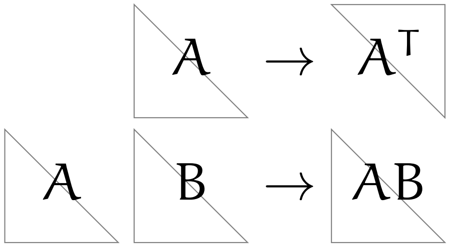
Encoded Linear Algebra Knowledge

Properties

Operation	Cost
$A^{-1}B$ (A lower triangular)	n^3
$A^{-1}B$	$2.7n^3$

Encoded Linear Algebra Knowledge

Inference of Properties



$$\text{expr} = \text{expr}^T \rightarrow \text{Symmetric}(\text{expr})$$

Simplifications

$$(AB)^T \rightarrow B^T A^T$$

$$A^T \rightarrow A \qquad \text{if Symmetric}(A)$$

$$Q^T Q \rightarrow I \qquad \text{if Orthogonal}(Q)$$

Optimizations

Common Subexpression Elimination

$$AB^{-T}B^{-1}A^{T} = CC^{T}$$

$$AB^{-T} = \left(B^{-1}A^{T}\right)^{T} = C$$

Generalized Matrix Chain Problem

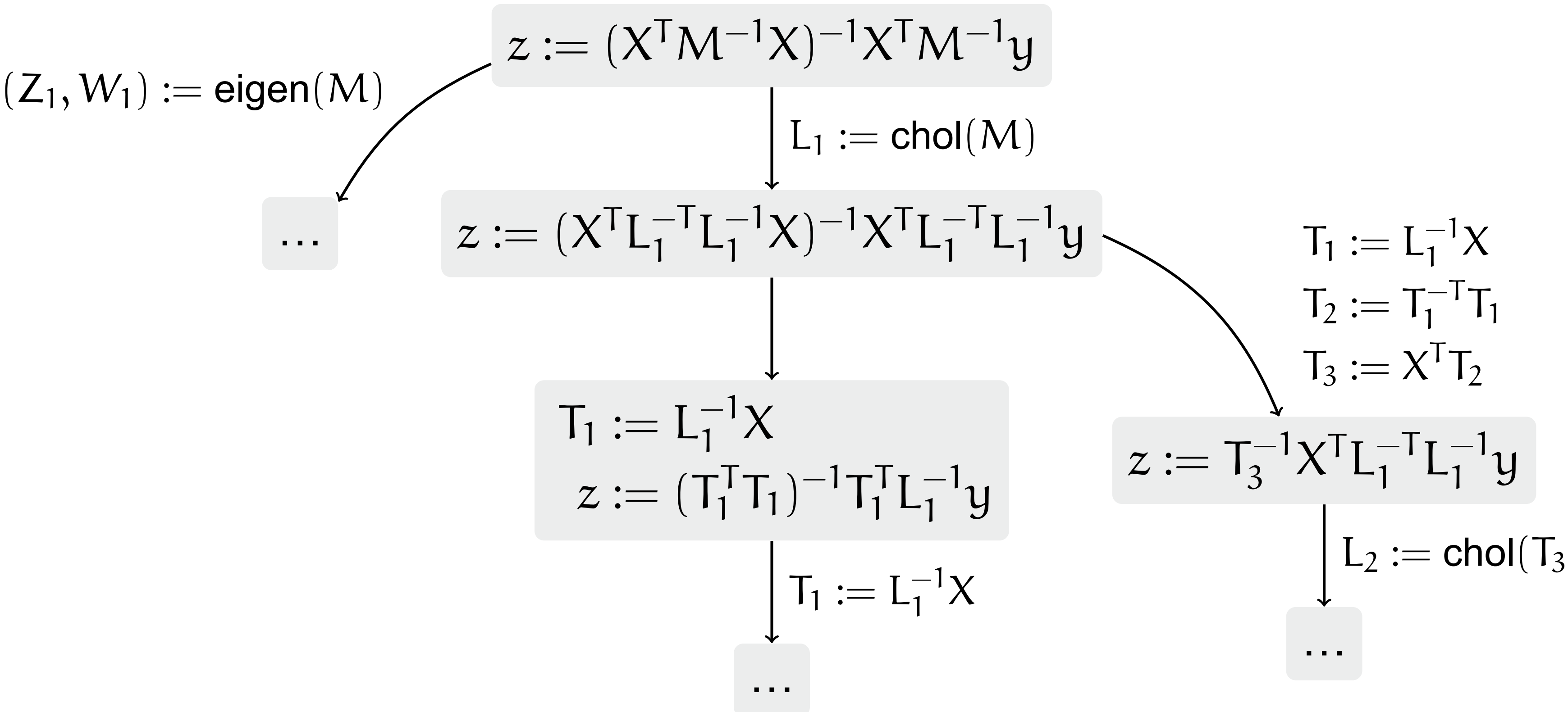
$$A \quad B \quad c$$

$$(AB)c \quad \mathcal{O}(n^3) \qquad\qquad A(Bc) \quad \mathcal{O}(n^2)$$

In practice:

$$X := AB^{T}C^{-T}D$$

Derivation Graph



Results

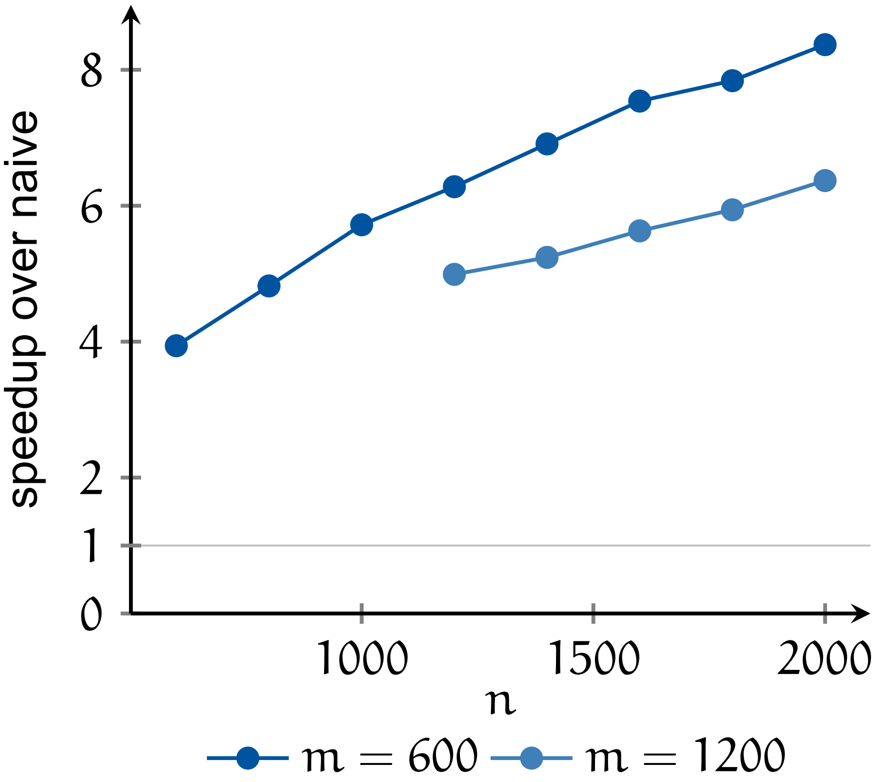
$$z := (X^T M^{-1} X)^{-1} X^T M^{-1} y, X \in \mathbb{R}^{n \times m}.$$

Naive implementation

```
z = inv(X'*inv(M)*X)*X'*inv(M)*y;
```

Compiler implementation

```
L1 = chol(M, 'lower');
t1 = linsolve(L1, X, opts1);
t2 = t1'*t1;
L2 = chol(t2, 'lower');
t3 = linsolve(L1, y, opts1);
t4 = t1'*t3;
t5 = linsolve(L2, t4, opts1);
z = linsolve(L2, t5, opts2);
```



Results

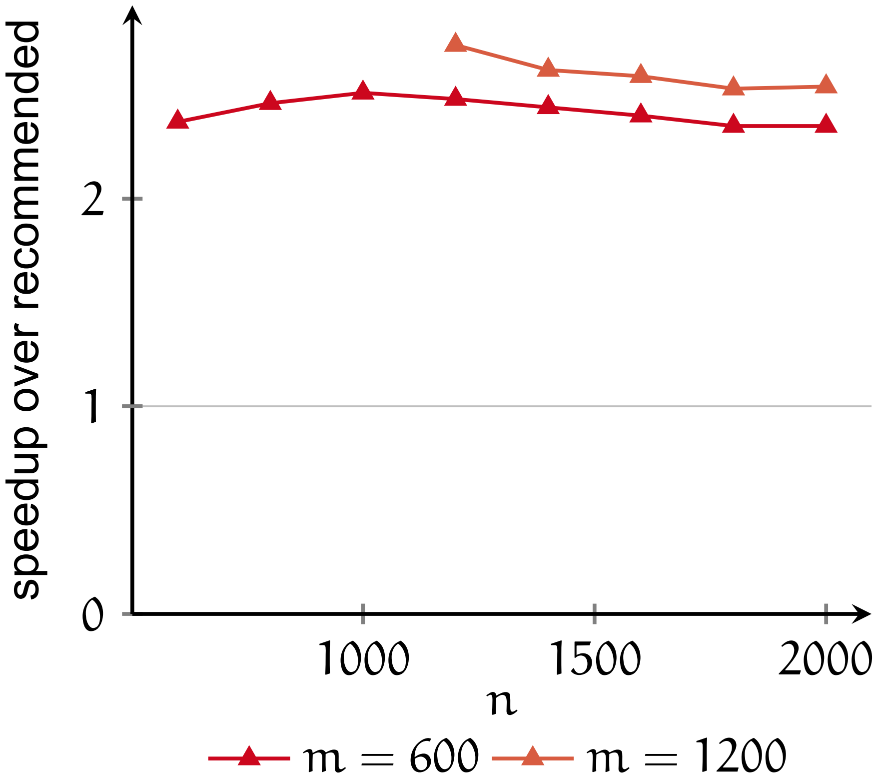
$$z := (X^T M^{-1} X)^{-1} X^T M^{-1} y, \; X \in \mathbb{R}^{n \times m}.$$

Recommended implementation

```
z = (X'*(M\X))\X'*(M\y);
```

Compiler implementation

```
L1 = chol(M, 'lower');
t1 = linsolve(L1, X, opts1);
t2 = t1'*t1;
L2 = chol(t2, 'lower');
t3 = linsolve(L1, y, opts1);
t4 = t1'*t3;
t5 = linsolve(L2, t4, opts1);
z = linsolve(L2, t5, opts2);
```



References

[AB+99] Edward Anderson, Zhaojun Bai, et al. *LAPACK Users' guide*, volume 9. SIAM, 1999.

[DDC+90] Jack J. Dongarra, Jeremy Du Croz, et al. A set of Level 3 Basic Linear Algebra Subprograms. *ACM TOMS*, 16(1):1–17, 1990.