Systematic Generation of Algorithms for Iterative Methods

Henrik Barthels

April 17, 2015

Introduction: The FLAME Project

- FLAME: Formal Linear Algebra Methods Environment.
- Goal: Automatic development of linear algebra libraries.
  - Automatic generation of algorithms (direct methods).

My thesis:

- Iterative methods.
- Preliminary results by Eijkhout et al., 2010.

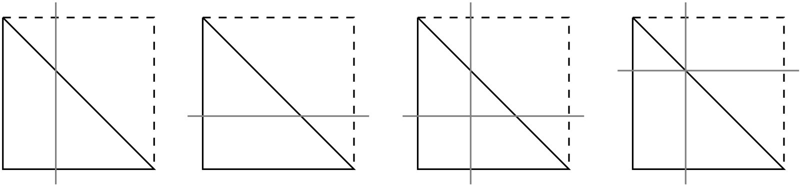
PME Generation: Input

Operation: Lower triangular linear system.

$$x := \Phi(A, b) \equiv \begin{cases} P_{\text{pre}} : \{\texttt{Input}[A] \wedge \texttt{Matrix}[A] \wedge \\ \qquad \texttt{NonSingular}[A] \wedge \\ \qquad \texttt{LowerTriangular}[A] \wedge \\ \qquad \texttt{Input}[b] \wedge \texttt{Vector}[b] \wedge \\ \qquad \texttt{Output}[x] \wedge \texttt{Vector}[x]\} \\ P_{\text{post}} : \{Ax = b\} \end{cases}$$

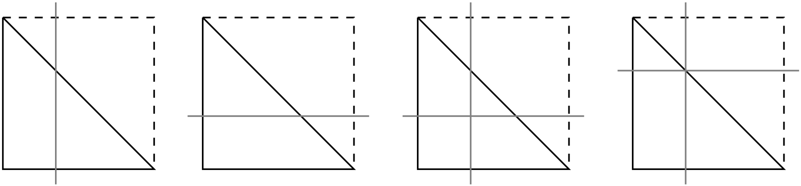
PME Generation: Initial Partitioning

$$P_{\text{post}} : \{Ax = b\}$$



PME Generation: Initial Partitioning

$$P_{\text{post}} : \{Ax = b\}$$



$$\left(\begin{array}{c|c} A_{TL} & 0 \\ \hline A_{BL} & A_{BR} \end{array}\right) \left(\begin{array}{c} x_T \\ \hline x_B \end{array}\right) = \left(\begin{array}{c} b_T \\ \hline b_B \end{array}\right)$$

PME Generation

$$\left(\begin{array}{c|c} A_{TL} & 0 \\ \hline A_{BL} & A_{BR} \end{array}\right)\left(\begin{array}{c} x_T \\ \hline x_B \end{array}\right) = \left(\begin{array}{c} b_T \\ \hline b_B \end{array}\right)$$

PME Generation

$$\left(\frac{A_{TL}x_T}{A_{BL}x_T + A_{BR}x_B}\right) = \left(\frac{b_T}{b_B}\right)$$

PME Generation

$$\left(\frac{A_{TL} x_T = b_T}{A_{BL} x_T + A_{BR} x_B = b_B} \right)$$

PME Generation

$$\left(\frac{x_T := \Phi\left(A_{TL}, b_T\right)}{A_{BL}x_T + A_{BR}x_B = b_B} \right)$$

PME Generation

$$\left(\frac{x_T := \Phi\left(A_{TL}, b_T\right)}{A_{BR} x_B = b_B - A_{BL} x_T} \right)$$

PME Generation

Partitioned Matrix Expression (PME):

$$\left(\frac{x_T := \Phi\left(A_{TL}, b_T\right)}{x_B := \Phi\left(A_{BR}, b_B - A_{BL} x_T\right)} \right)$$

Loop Invariant Identification

$\ldots$

$\{P_{inv}\}$

While G do

    $\{P_{inv}\}$

    $\ldots$

    $\{P_{inv}\}$

endwhile

$\{P_{inv} \wedge \neg G\}$

$\ldots$

Loop Invariant Identification

$$\left(\frac{x_T := \Phi\left(A_{TL}, b_T\right)}{x_B := \Phi\left(A_{BR}, b_B - A_{BL} x_T\right)} \right)$$

Loop Invariant Identification

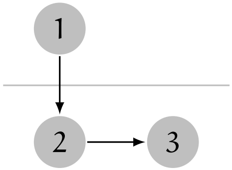
$$\left(\frac{x_T := \Phi\left(A_{TL}, b_T\right)}{x_B := \Phi\left(A_{BR}, b_B - A_{BL}x_T\right)}\right)$$

1. $x_T := \Phi\left(A_{TL}, b_T\right)$
2. $b_B := b_B - A_{BL}x_T$
3. $x_B := \Phi\left(A_{BR}, b_B\right)$

Loop Invariant Identification

$$\left(\frac{x_T := \Phi\left(A_{TL}, b_T\right)}{x_B := \Phi\left(A_{BR}, b_B - A_{BL} x_T\right)} \right)$$

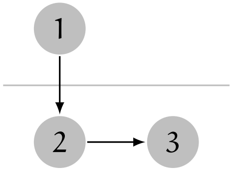
1. $x_T := \Phi\left(A_{TL}, b_T\right)$
2. $b_B := b_B - A_{BL} x_T$
3. $x_B := \Phi\left(A_{BR}, b_B\right)$



Loop Invariant Identification

$$\left(\frac{x_T := \Phi\left(A_{TL}, b_T\right)}{x_B := \Phi\left(A_{BR}, b_B - A_{BL} x_T\right)} \right)$$

1. $x_T := \Phi\left(A_{TL}, b_T\right)$
2. $b_B := b_B - A_{BL} x_T$
3. $x_B := \Phi\left(A_{BR}, b_B\right)$

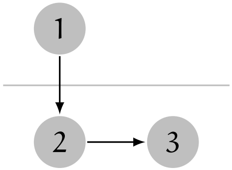


Loop invariant candidates: $\varnothing, \{1\}, \{1, 2\}, \{1, 2, 3\}$

Loop Invariant Identification

$$\left(\frac{x_T := \Phi\left(A_{TL}, b_T\right)}{x_B := \Phi\left(A_{BR}, b_B - A_{BL} x_T\right)}\right)$$

1. $x_T := \Phi\left(A_{TL}, b_T\right)$
2. $b_B := b_B - A_{BL} x_T$
3. $x_B := \Phi\left(A_{BR}, b_B\right)$



Loop invariant candidates: $\varnothing, \{1\}, \{1,2\}, \{1,2,3\}$
Feasible loop invariants: $\{1\}, \{1,2\}$

Algorithm Construction: Worksheet

Algorithm: ...
$\{ P_{\text{pre}} \}$
Partition
$\{ P_{\text{inv}} \}$
While G **do**
$\{(P_{\text{inv}}) \wedge (G)\}$
Repartition
$\{ P_{\text{before}} \}$
Update
$\{ P_{\text{after}} \}$
Continue with
$\{ P_{\text{inv}} \}$
endwhile
$\{(P_{\text{inv}}) \wedge \neg (G)\}$
$\{ P_{\text{post}} \}$

Algorithm Construction: Repartitioning

"Repartition" rule:

$$\left(\begin{array}{c|c} A_{TL} & 0 \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & 0 & 0 \\ \hline A_{10} & A_{11} & 0 \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

"Continue with" rule:

$$\left(\begin{array}{c|c} A_{TL} & 0 \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & 0 & 0 \\ \hline A_{10} & A_{11} & 0 \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

Algorithm Construction: Repartitioning

"Repartition" rule:

$$\left(\begin{array}{c|c} A_{TL} & 0 \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & 0 & 0 \\ \hline A_{10} & A_{11} & 0 \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

"Continue with" rule:

$$\left(\begin{array}{c|c} A_{TL} & 0 \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & 0 & 0 \\ \hline A_{10} & A_{11} & 0 \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

P_{inv}:

$$\{x_T := \Phi\left(A_{TL}, b_T\right) \wedge b_B := b_B - A_{BL}x_T\}$$

Algorithm Construction

P_{before}

$$x_0 := \Phi\left(A_{00}, b_0\right)$$
$$b_1 := b_1 - A_{10}x_0$$
$$b_2 := b_2 - A_{20}x_0$$

P_{after}

$$x_0 := \Phi(A_{00}, b_0)$$
$$x_1 := \Phi(A_{11}, b_1 - A_{10}x_0)$$
$$b_2 := b_2 - A_{20}x_0 - A_{21}x_1$$

Algorithm Construction

P_{before}

$$x_0 := \Phi\left(A_{00}, b_0\right)$$
$$b_1 := b_1 - A_{10}x_0$$
$$b_2 := b_2 - A_{20}x_0$$

P_{after}

$$x_0 := \Phi(A_{00}, b_0)$$
$$x_1 := \Phi(A_{11}, b_1 - A_{10}x_0)$$
$$b_2 := b_2 - A_{20}x_0 - A_{21}x_1$$

Update:

$$x_1 := \Phi\left(A_{11}, b_1\right)$$

Algorithm Construction

P_{before}

$$
\begin{aligned}
x_0 &:= \Phi\left(A_{00}, b_0\right) \\
b_1 &:= b_1 - A_{10}x_0 \\
b_2 &:= b_2 - A_{20}x_0
\end{aligned}
$$

P_{after}

$$
\begin{aligned}
x_0 &:= \Phi(A_{00}, b_0) \\
x_1 &:= \Phi(A_{11}, b_1 - A_{10}x_0) \\
b_2 &:= b_2 - A_{20}x_0 - A_{21}x_1
\end{aligned}
$$

Update:

$$
\begin{aligned}
x_1 &:= \Phi\left(A_{11}, b_1\right) \\
b_2 &:= b_2 - A_{21}x_1
\end{aligned}
$$

Algorithm Construction

Algorithm: $x := \Phi(A, b)$

Partition $A \rightarrow \left(\begin{array}{c|c} A_{TL} & 0 \\ \hline A_{BL} & A_{BR} \end{array}\right)$

where A_{TL} is 0×0

While size$(A_{TL}) <$ size(A) **do**

Repartition

$$\left(\begin{array}{c|c} A_{TL} & 0 \\ \hline A_{BL} & A_{BR} \end{array}\right) \rightarrow \left(\begin{array}{c|c|c} A_{00} & 0 & 0 \\ \hline A_{10} & A_{11} & 0 \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

where A is $k \times k$

Variant 1	Variant 2
$b_1 := b_1 - A_{10}x_0$	$x_1 := \Phi(A_{11}, b_1)$
$x_1 := \Phi(A_{11}, b_1)$	$b_2 := b_2 - A_{21}x_1$

Continue with

$$\left(\begin{array}{c|c} A_{TL} & 0 \\ \hline A_{BL} & A_{BR} \end{array}\right) \leftarrow \left(\begin{array}{c|c|c} A_{00} & 0 & 0 \\ \hline A_{10} & A_{11} & 0 \\ \hline A_{20} & A_{21} & A_{22} \end{array}\right)$$

endwhile

Iterative Methods

- Iterative methods for solving linear systems $Ax = b$.
- Initial guess x_0.
- Successively better solutions $x_1, x_2, \ldots$
- Convergence: $\|Ax_i - b\| < \varepsilon$.

Iterative Methods

- Iterative methods for solving linear systems $Ax = b$.
- Initial guess x_0.
- Successively better solutions $x_1, x_2, \ldots$
- Convergence: $\|Ax_i - b\| < \varepsilon$.

Matrix representation:

$$X = (x_0 \; x_1 \; x_2 \ldots)$$
$$R = (r_0 \; r_1 \; r_2 \ldots)$$

Matrix Representation

Operation: Symmetric conjugate gradient method.

P_{post} :

$$
\begin{aligned}
APD &= R\left(\underline{I} - \underline{J}\right) \\
P\left(I - U\right) &= \underline{R} \\
PD &= X\left(\underline{I} - \underline{J}\right) \\
\|Re_r\| &< \varepsilon
\end{aligned}
$$

Source: Eijkhout et al., 2010.

Matrix Representation

Operation: Symmetric conjugate gradient method.

P_{post} :

$$
\begin{aligned}
APD &= R\left(\underline{I} - \underline{J}\right) \\
P\left(I - U\right) &= \underline{R} \\
PD &= X\left(\underline{I} - \underline{J}\right) \\
\|Re_r\| &< \varepsilon
\end{aligned}
$$

$$
\begin{aligned}
X &= (x_0 \; x_1 \; x_2 \ldots) \\
R &= (r_0 \; r_1 \; r_2 \ldots)
\end{aligned}
$$

Source: Eijkhout et al., 2010.

Matrix Representation

Operation: Symmetric conjugate gradient method.

P_{post} :

$$
\begin{aligned}
APD &= R\left(\underline{I} - \underline{J}\right) \\
P\left(I - U\right) &= \underline{R} \\
PD &= X\left(\underline{I} - \underline{J}\right) \\
\|Re_r\| &< \varepsilon
\end{aligned}
$$

$$\{R, U, P, D, X\} := \mathsf{CG}\left(A, Re_0, Xe_0\right)$$

Source: Eijkhout et al., 2010.

PME Generation: Initial Partitioning

$$\begin{aligned} APD &= R\left(\underline{I} - \underline{J}\right) \\ P\left(I - U\right) &= \underline{R} \\ PD &= X\left(\underline{I} - \underline{J}\right) \end{aligned}$$

PME Generation: Initial Partitioning

$$\begin{aligned} APD &= R\left(\underline{I} - \underline{J}\right) \\ P\left(I - U\right) &= \underline{R} \\ PD &= X\left(\underline{I} - \underline{J}\right) \end{aligned}$$

$$A \left(\begin{array}{c|c} P_L & p_R \end{array} \right) \left(\begin{array}{c|c} D_{TL} & 0 \\ \hline 0 & \delta_{BR} \end{array} \right) = \left(\begin{array}{c|c|c} R_L & r_R & r_+ \end{array} \right) \left(\begin{array}{c|c} I - J & 0 \\ \hline -e_r^T & 1 \\ \hline 0 & -1 \end{array} \right)$$

$$\left(\begin{array}{c|c} P_L & p_R \end{array} \right) \left(\begin{array}{c|c} I - U_{TL} & -u_{TR} \\ \hline 0 & 1 \end{array} \right) = \left(\begin{array}{c|c} R_L & r_R \end{array} \right)$$

$$\left(\begin{array}{c|c} P_L & p_R \end{array} \right) \left(\begin{array}{c|c} D_{TL} & 0 \\ \hline 0 & \delta_{BR} \end{array} \right) = \left(\begin{array}{c|c|c} X_L & x_R & x_+ \end{array} \right) \left(\begin{array}{c|c} I - J & 0 \\ \hline -e_r^T & 1 \\ \hline 0 & -1 \end{array} \right)$$

PME Generation

$$\left(\begin{array}{c|c} AP_L D_{TL} = R_L \left(I - J \right) - r_R e_r^T & Ap_R \delta_{BR} = r_R - r_+ \end{array} \right)$$

$$\left(\begin{array}{c|c} P_L \left(I - U_{TL} \right) = R_L & -P_L u_{TR} + p_R = r_R \end{array} \right)$$

$$\left(\begin{array}{c|c} P_L D_{TL} = X_L \left(I - J \right) - x_R e_r^T & p_R \delta_{BR} = x_R - x_+ \end{array} \right)$$

$$\begin{aligned} P_{\text{post}} : \\ APD &= R \left(\underline{I} - \underline{J} \right) \\ P \left(I - U \right) &= \underline{R} \\ PD &= X \left(\underline{I} - \underline{J} \right) \end{aligned}$$

PME Generation

$$\left(\begin{array}{c|c} AP_L D_{TL} = R_L (I - J) - r_R e_r^T & Ap_R \delta_{BR} = r_R - r_+ \end{array} \right)$$

$$\left(\begin{array}{c|c} P_L (I - U_{TL}) = R_L & -P_L u_{TR} + p_R = r_R \end{array} \right)$$

$$\left(\begin{array}{c|c} P_L D_{TL} = X_L (I - J) - x_R e_r^T & p_R \delta_{BR} = x_R - x_+ \end{array} \right)$$

$$AP_L D_{TL} = \left(\begin{array}{c|c} R_L & r_R \end{array} \right) \left(\frac{I - J}{-e_r^T} \right)$$

$$P_L (I - U_{TL}) = R_L$$

$$P_L D_{TL} = \left(\begin{array}{c|c} X_L & x_R \end{array} \right) \left(\frac{I - J}{-e_r^T} \right)$$

P_{post} :

$$APD = R \left(\underline{I} - \underline{J} \right)$$

$$P (I - U) = \underline{R}$$

$$PD = X \left(\underline{I} - \underline{J} \right)$$

PME Generation

$$\left(\; AP_L D_{TL} = R_L \left(I - J \right) - r_R e_r^T \;\middle|\; Ap_R \delta_{BR} = r_R - r_+ \; \right)$$

$$\left(\; P_L \left(I - U_{TL} \right) = R_L \;\middle|\; -P_L u_{TR} + p_R = r_R \; \right)$$

$$\left(\; P_L D_{TL} = X_L \left(I - J \right) - x_R e_r^T \;\middle|\; p_R \delta_{BR} = x_R - x_+ \; \right)$$

$$\{ \left(R_L \mid r_R \right), U_{TL}, P_L, D_{TL}, \left(X_L \mid x_R \right) \} := \mathbf{CG} \left(A, R_L e_0, X_L e_0 \right)$$

PME Generation

$$\left(\; A P_L D_{TL} = R_L\,(I - J) - r_R e_r^T \;\middle|\; A p_R \delta_{BR} = r_R - r_+ \; \right)$$

$$\left(\; P_L\,(I - U_{TL}) = R_L \;\middle|\; -P_L u_{TR} + p_R = r_R \; \right)$$

$$\left(\; P_L D_{TL} = X_L\,(I - J) - x_R e_r^T \;\middle|\; p_R \delta_{BR} = x_R - x_+ \; \right)$$

$$\{(R_L \mid r_R)\,, U_{TL}, P_L, D_{TL}, (X_L \mid x_R)\} := \mathbf{CG}\,(A, R_L e_0, X_L e_0)$$

$$-P_L u_{TR} + p_R = r_R$$

PME Generation

$$\left(\; AP_L D_{TL} = R_L \left(I - J \right) - r_R e_r^T \;\middle|\; A p_R \delta_{BR} = r_R - r_+ \; \right)$$

$$\left(\; P_L \left(I - U_{TL} \right) = R_L \;\middle|\; -P_L u_{TR} + p_R = r_R \; \right)$$

$$\left(\; P_L D_{TL} = X_L \left(I - J \right) - x_R e_r^T \;\middle|\; p_R \delta_{BR} = x_R - x_+ \; \right)$$

$$\left\{ \left(R_L \mid r_R \right), U_{TL}, P_L, D_{TL}, \left(X_L \mid x_R \right) \right\} := \mathbf{CG} \left(A, R_L e_0, X_L e_0 \right)$$

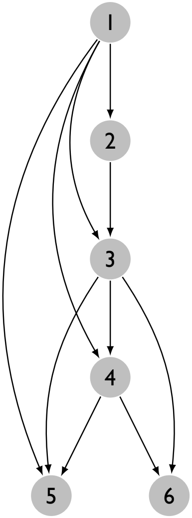
$$-P_L^T A P_L u_{TR} + P_L^T A p_R = P_L^T A r_R$$

PME Generation

PME:

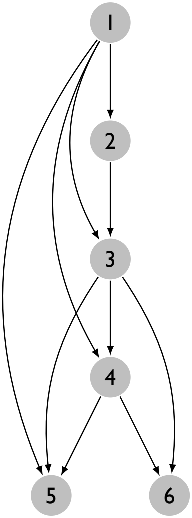
$$\begin{aligned}
\{(R_L \mid r_R), U_{TL}, P_L, D_{TL}, (X_L \mid x_R)\} &:= \mathbf{CG}(A, R_L e_0, X_L e_0) \\
u_{TR} &:= -\left(P_L^T A P_L\right)^{-1} P_L^T A r_R \\
p_R &:= r_R + P_L u_{TR} \\
\delta_{BR} &:= \frac{r_R^T r_R}{r_R^T A p_R} \\
r_+ &:= r_R - A p_R \delta_{BR} \\
x_+ &:= x_R - p_R \delta_{BR}
\end{aligned}$$

Loop Invariant Identification



1. $\{\ldots\} := \mathsf{CG}\,(A, R_L e_0, X_L e_0)$
2. $u_{TR} := -\left(P_L^T A P_L\right)^{-1} P_L^T A r_R$
3. $p_R := r_R + P_L u_{TR}$
4. $\delta_{BR} := \frac{r_R^T r_R}{r_R^T A p_R}$
5. $r_+ := r_R - A p_R \delta_{BR}$
6. $x_+ := x_R - p_R \delta_{BR}$

Loop Invariant Identification



1. $\{\ldots\} := \mathsf{CG}\,(A, R_L e_0, X_L e_0)$
2. $u_{TR} := -\left(P_L^T A P_L\right)^{-1} P_L^T A r_R$
3. $p_R := r_R + P_L u_{TR}$
4. $\delta_{BR} := \frac{r_R^T r_R}{r_R^T A p_R}$
5. $r_+ := r_R - A p_R \delta_{BR}$
6. $x_+ := x_R - p_R \delta_{BR}$

We use $P_{\text{inv}} = \{1, 2, 3\}$

Loop Invariant Identification: Feasibility

Algorithm: $\ldots$
$\{ P_{pre} \}$
Partition
Preprocessing
$\{ P_{inv} \}$
While G **do**
$\{(P_{inv}) \wedge (G)\}$
Repartition
$\{ P_{before} \}$
Update
$\{ P_{after} \}$
Continue with
$\{ P_{inv} \}$
endwhile
$\{(P_{inv}) \wedge \neg (G)\}$
$\{ P_{post} \}$

Loop Invariant Identification: Feasibility

P_{post} :

$$\begin{aligned} APD &= R\left(\underline{I} - \underline{J}\right) \\ P\left(I - U\right) &= \underline{R} \\ PD &= X\left(\underline{I} - \underline{J}\right) \\ \|Re_r\| &< \varepsilon \end{aligned}$$

Loop Invariant Identification: Feasibility

P_{post} :

$$\begin{aligned} APD &= R\left(\underline{I} - \underline{J}\right) \\ P\,(I - U) &= \underline{R} \\ PD &= X\left(\underline{I} - \underline{J}\right) \\ \|Re_r\| &< \varepsilon \end{aligned}$$

Loop guard G: $\|r_R\| \geqslant \varepsilon$

Loop Invariant Identification: Feasibility

P_{post} :

$$
\begin{aligned}
APD &= R\left(\underline{I} - \underline{J}\right) \\
P\left(I - U\right) &= \underline{R} \\
PD &= X\left(\underline{I} - \underline{J}\right) \\
\|Re_r\| &< \varepsilon
\end{aligned}
$$

Loop guard G: $\|r_R\| \geqslant \varepsilon$

$$
R \rightarrow \left(\; R_L \;\middle|\; r_R \;\middle|\; r_+ \; \right)
$$

Loop Invariant Identification: Feasibility

$$P_{\text{inv}} \wedge \neg G \Rightarrow P_{\text{post}}$$

P_{inv}:

$$\begin{aligned}\{(R_L \mid r_R), U_{TL}, P_L, D_{TL}, (X_L \mid x_R)\} &:= \mathbf{CG}\left(A, R_L e_0, X_L e_0\right)\\ u_{TR} &:= -\left(P_L^T A P_L\right)^{-1} P_L^T A r_R\\ p_R &:= r_R + P_L u_{TR}\end{aligned}$$

$G : \|r_R\| \geqslant \varepsilon$

Loop Invariant Identification: Feasibility

$$P_{\text{inv}} \wedge \neg G \Rightarrow P_{\text{post}}$$

P_{inv}:

$$\begin{aligned} \{(R_L \mid r_R), U_{TL}, P_L, D_{TL}, (X_L \mid x_R)\} &:= \mathbf{CG}\,(A, R_L e_0, X_L e_0) \\ u_{TR} &:= -\left(P_L^T A P_L\right)^{-1} P_L^T A r_R \\ p_R &:= r_R + P_L u_{TR} \end{aligned}$$

$\neg G : \|r_R\| < \varepsilon$

Loop Invariant Identification: Feasibility

$$P_{\text{inv}} \wedge \neg G \Rightarrow P_{\text{post}}$$

P_{inv}:

$$\begin{aligned} \{(R_L \mid r_R), U_{TL}, P_L, D_{TL}, (X_L \mid x_R)\} &:= \mathbf{CG}\left(A, R_L e_0, X_L e_0\right) \\ u_{TR} &:= -\left(P_L^T A P_L\right)^{-1} P_L^T A r_R \\ p_R &:= r_R + P_L u_{TR} \end{aligned}$$

$\neg G : \|r_R\| < \varepsilon$

P_{post}:

$$\begin{aligned} \{R, U, P, D, X\} &:= \mathbf{CG}\left(A, Re_0, Xe_0\right) \\ \|Re_r\| &< \varepsilon \end{aligned}$$

Algorithm Construction: Repartitioning

"Repartition" rules:

$$\left(R_L \,\middle|\, r_R \,\middle|\, r_+ \right) \rightarrow \left(R_0 \,\middle|\, r_1 \,\middle|\, r_2 \right)$$

$$\left(\begin{array}{c|c} D_{TL} & 0 \\ \hline 0 & \delta_{BR} \end{array} \right) \rightarrow \left(\begin{array}{c|c} D_{00} & 0 \\ \hline 0 & \delta_{11} \end{array} \right)$$

"Continue with" rules:

$$\left(R_L \,\middle|\, r_R \,\middle|\, r_+ \right) \leftarrow \left(R_0 \,|\, r_1 \,\middle|\, r_2 \,\middle|\, r_3 \right)$$

$$\left(\begin{array}{c|c} D_{TL} & 0 \\ \hline 0 & \delta_{BR} \end{array} \right) \leftarrow \left(\begin{array}{c|c|c} D_{00} & 0 & 0 \\ \hline 0 & \delta_{11} & 0 \\ \hline 0 & 0 & \delta_{11} \end{array} \right)$$

Algorithm Construction

P_{before}

$$\begin{aligned}
\{(R_0 \mid r_1), U_{00}, P_0, D_{00}, (X_0 \mid x_1)\} &:= \mathsf{CG}\,(A, R_0 e_0, X_0 e_0) \\
u_{01} &:= -\left(P_0^T A P_0\right)^{-1} P_0^T A r_1 \\
p_1 &:= r_1 + P_0 u_{01}
\end{aligned}$$

Algorithm Construction

P_{after}:

$$\begin{aligned}
\{(R_0 \mid r_1), U_{00}, P_0, D_{00}, (X_0 \mid x_1)\} &:= \mathsf{CG}(A, R_0 e_0, X_0 e_0) \\
u_{01} &:= -\left(P_0^T A P_0\right)^{-1} P_0^T A r_1 \\
p_1 &:= r_1 + P_0 u_{01} \\
\delta_{11} &:= \frac{r_1^T r_1}{r_1^T A p_1} \\
r_2 &:= r_1 - A p_1 \delta_{11} \\
x_2 &:= x_1 - p_1 \delta_{11} \\
\nu_{12} &:= -\frac{p_1^T A r_2}{p_1^T A p_1} \\
p_2 &:= r_2 + p_1 \nu_{12}
\end{aligned}$$

Algorithm Construction

P_{after}:

$$\begin{aligned}
\{(R_0 \mid r_1), U_{00}, P_0, D_{00}, (X_0 \mid x_1)\} &:= \mathsf{CG}\,(A, R_0 e_0, X_0 e_0) \\
u_{01} &:= -\left(P_0^T A P_0\right)^{-1} P_0^T A r_1 \\
p_1 &:= r_1 + P_0 u_{01} \\
\delta_{11} &:= \frac{r_1^T r_1}{r_1^T A p_1} \\
r_2 &:= r_1 - A p_1 \delta_{11} \\
x_2 &:= x_1 - p_1 \delta_{11} \\
\nu_{12} &:= -\frac{p_1^T A r_2}{p_1^T A p_1} \\
p_2 &:= r_2 + p_1 \nu_{12}
\end{aligned}$$

Conclusion

Automatic derivation of algorithms for iterative methods.

- Automatic derivation of matrix properties.
- Provably correct algorithms.
- Families of algorithms.

Future work:

- Matrix representations.
- Stability analysis.
- Performance analysis.
- Implementation.

Bibliography

Paolo Bientinesi.
Mechanical Derivation and Systematic Analysis of Correct Linear Algebra Algorithms.
Ph.D. Thesis, Department of Computer Sciences, University of Texas at Austin, July 2006.

Victor Eijkhout, Paolo Bientinesi, and Robert van de Geijn.
Proof-Driven Derivation of Krylov Solver Libraries.
Technical report, Aachen Institute for Computational Engineering Science, RWTH Aachen, June 2010.
Technical Report AICES-2010/06-3.

Derivation of Properties

$$\begin{aligned} APD &= R\left(I - J\right) \\ P\left(I - U\right) &= R \end{aligned}$$

Derivation of Properties

$$
\begin{aligned}
APD &= R(I - J) \\
P(I - U) &= R \\
AP &= R(I - J)D^{-1} \\
P &= R(I - U)^{-1}
\end{aligned}
$$

Derivation of Properties

$$
\begin{aligned}
APD &= R\,(I - J) \\
P\,(I - U) &= R \\
AP &= R\,(I - J)\,D^{-1} \\
P &= R\,(I - U)^{-1}
\end{aligned}
$$

`Diagonal`$[R^T R]$

Derivation of Properties

$$
\begin{aligned}
APD &= R\,(I - J) \\
P\,(I - U) &= R \\
AP &= R\,(I - J)\,D^{-1} \\
P &= R\,(I - U)^{-1}
\end{aligned}
$$

`Diagonal`$[R^T R]$

$$
\begin{aligned}
R^T AP &= R^T R\,(I - J)\,D^{-1} \\
R^T P &= R^T R\,(I - U)^{-1}
\end{aligned}
$$

Derivation of Properties

$$\begin{aligned}
APD &= R(I - J) \\
P(I - U) &= R \\
AP &= R(I - J)D^{-1} \\
P &= R(I - U)^{-1}
\end{aligned}$$

$\texttt{Diagonal}[R^T R]$

$$\begin{aligned}
R^T AP &= R^T R(I - J)D^{-1} \\
R^T P &= R^T R(I - U)^{-1}
\end{aligned}$$

$\texttt{UpperTriangular}[R^T P] \rightarrow \texttt{LowerTriangular}[P^T R]$

Derivation of Properties

$$\begin{aligned} APD &= R\,(I - J) \\ P\,(I - U) &= R \\ AP &= R\,(I - J)\,D^{-1} \\ P &= R\,(I - U)^{-1} \end{aligned}$$

$\texttt{Diagonal}[R^T R]$

$$\begin{aligned} R^T AP &= R^T R\,(I - J)\,D^{-1} \\ R^T P &= R^T R\,(I - U)^{-1} \end{aligned}$$

$\texttt{UpperTriangular}[R^T P] \rightarrow \texttt{LowerTriangular}[P^T R]$

$$P^T AP = P^T R\,(I - J)\,D^{-1}$$

Loop Invariant Identification: Feasibility

Initially, R_L, P_L,... have size $n \times 0$.

$$
\begin{aligned}
\{(R_L \mid r_R), U_{TL}, P_L, D_{TL}, (X_L \mid x_R)\} &:= \mathsf{CG}(A, R_L e_0, X_L e_0) \\
u_{TR} &:= -\left(P_L^T A P_L\right)^{-1} P_L^T A r_R \\
p_R &:= r_R + P_L u_{TR}
\end{aligned}
$$

Loop Invariant Identification: Feasibility

Initially, R_L, P_L,... have size $n \times 0$.

$$\{r_R, \varnothing, \varnothing, \varnothing, x_R\} = \mathsf{CG}\left(A, r_R, x_R\right)$$

$$p_R = r_R$$

Loop Invariant Identification: Feasibility

Initially, R_L, P_L,... have size $n \times 0$.

$$\{r_R, \varnothing, \varnothing, \varnothing, x_R\} = \mathsf{CG}\left(A, r_R, x_R\right)$$

$$p_R = r_R$$

Preprocessing operation: $p_R := r_R$