Low dimensional approximation of weak constraint variational data assimilation

Melina Freitag

Universität Potsdam
Institut für Mathematik

SFB Colloquium
13th December 2019

joint work with Daniel Green (University of Bath)

Data assimilation setting

Denote $x_k \in \mathbb{R}^n$ state of a system at time t_k.

- numerical (physical) model $\mathcal{M}_k \colon \mathbb{R}^n \to \mathbb{R}^n$ such that

$$x_{k+1} = \mathcal{M}_k(x_k) + \eta_k.$$

- prior estimate x_0^b of the initial condition x_0,

$$x_0 = x_0^b + e_0.$$

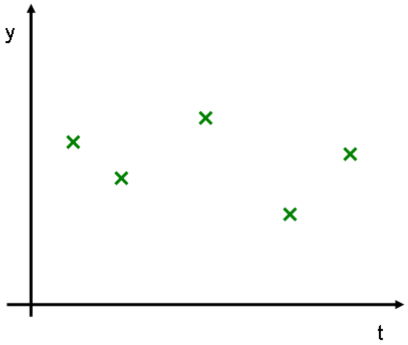
- observations $y_k \in \mathbb{R}^{p_k}$ of the state:

$$y_k = \mathcal{H}_k(x_k) + \epsilon_k,$$

  where $\mathcal{H}_k : \mathbb{R}^n \to \mathbb{R}^{p_k}$ is an observation operator.

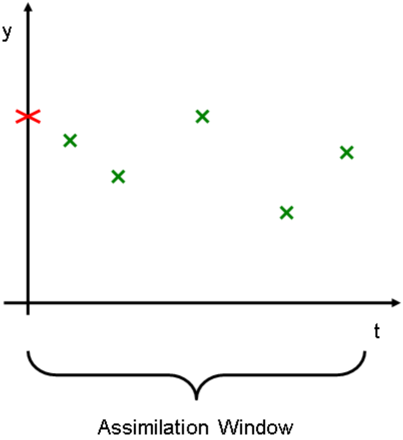
The errors η_k, e_0, ϵ_k are Gaussian with zero mean and covariances $Q_k \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times n}$, $R_k \in \mathbb{R}^{p_k \times p_k}$ respectively.

Schematics of 4D-Var data assimilation



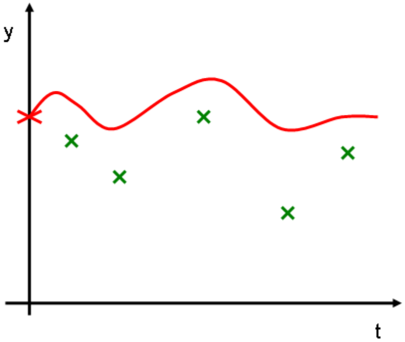
- Take observations y_k of the true dynamical system.

Schematics of 4D-Var data assimilation



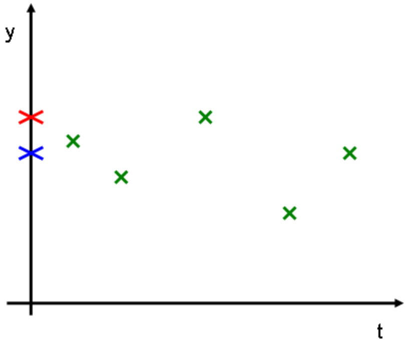
- Use a priori information x_0^b for the initial condition for the numerical model $x_{k+1} = \mathcal{M}_{k+1,k}(x_k)$, approximating the ("true") dynamical system.

Schematics of 4D-Var data assimilation



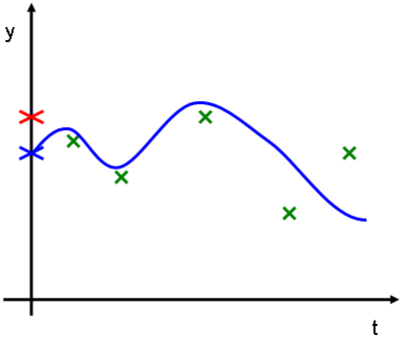
- Run the numerical model using the estimated initial condition.

Schematics of 4D-Var data assimilation



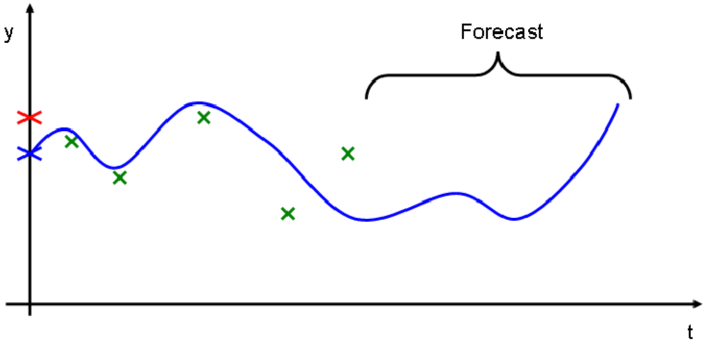
- Minimise a cost function $J(x)$ to find an improved initial condition x_0^a.

Schematics of 4D-Var data assimilation



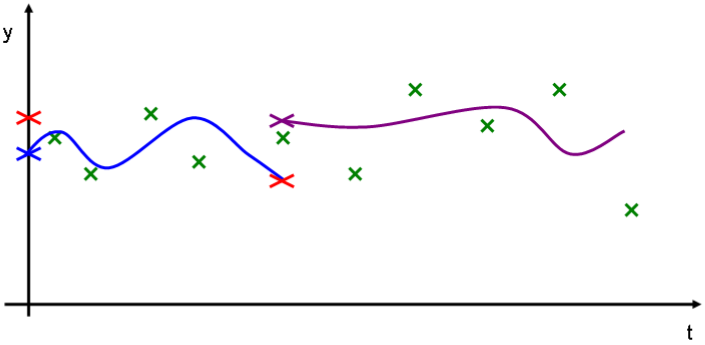
- The numerical model is run using x_0^a as an initial condition.

Schematics of 4D-Var data assimilation



- The simulation is continued to create a forecast.

Schematics of 4D-Var data assimilation



- The process is repeated for new observations.

Part I: A low-rank approach to the solution of weak constraint variational data assimilation problems

Part II: Balanced truncation within weak constraint 4D-Var

Part I

A low-rank approach to the solution of weak constraint variational data assimilation problems

Outline

Weak Constraint 4D-Var

4D-Var cost function

$$J(x) = \frac{1}{2}\|x_0 - x_0^b\|^2_{B^{-1}} + \frac{1}{2}\sum_{k=0}^{N}\|y_k - \mathcal{H}_k(x_k)\|^2_{R_k^{-1}} + \frac{1}{2}\sum_{k=1}^{N}\|x_k - \mathcal{M}_k(x_{k-1})\|^2_{Q_k^{-1}}.$$

where

- $x = \left[x_0^T, x_1^T, \ldots, x_N^T\right]^T$
- B, R_k, Q_k postitive definite error covariance matrices
- y_k observation vector
- $\mathcal{H}_k$ maps state vector x_k from model space to observation space
- $\mathcal{M}_k$ model integration

Incremental 4D-Var - Gauss-Newton method

4D-Var cost function

$$J(x) = \frac{1}{2}\|x_0 - x_0^b\|^2_{B^{-1}} + \frac{1}{2}\sum_{k=0}^{N}\|y_k - \mathcal{H}_k(x_k)\|^2_{R_k^{-1}} + \frac{1}{2}\sum_{k=1}^{N}\|x_k - \mathcal{M}_k(x_{k-1})\|^2_{Q_k^{-1}}.$$

Minimisation using Gauss-Newton method:

- linearise $\mathcal{M}_k$ and $\mathcal{H}_k$ about $x^{(\ell)}$ at each step
- (approximately) minimise quadratic cost function $\tilde{J}(\delta x^{(\ell)})$.
- Increment at iterate ℓ,

$$\delta x^{(\ell)} = \left[(\delta x_0^{(\ell)})^T, (\delta x_1^{(\ell)})^T, \dots, (\delta x_N^{(\ell)})^T\right]^T.$$

$$x^{(\ell+1)} = x^{(\ell)} + \delta x^{(\ell)}$$

Incremental 4D-Var - Gauss-Newton method

Incremental 4D-Var cost function

$$\tilde{J}(\delta x^{(\ell)}) = \frac{1}{2}\|\delta x_0^{(\ell)} - b_0^{(\ell)}\|_{B^{-1}} + \frac{1}{2}\sum_{k=0}^{N}\|d_k^{(\ell)} - H_k \delta x_k^{(\ell)}\|_{R_k^{-1}}$$

$$+ \frac{1}{2}\sum_{k=1}^{N}\|\delta x_k^{(\ell)} - M_k \delta x_{k-1}^{(\ell)} - c_k^{(\ell)}\|_{Q_k^{-1}}.$$

$M_k \in \mathbb{R}^{n\times n}$, $H_k \in \mathbb{R}^{p_k\times n}$ linearisations of $\mathcal{M}_k$ and $\mathcal{H}_k$ about $x^{(\ell)}$.

$$b_0^{(\ell)} = x_0^b - x_0^{(\ell)}, \quad d_k^{(\ell)} = y_k - \mathcal{H}_k(x_k^{(\ell)}), \quad c_k^{(\ell)} = \mathcal{M}_k(x_{k-1}^{(\ell)}) - x_k^{(\ell)}.$$

Concise notation for incremental 4D-Var (all-at-once approach)

Minimise (inner iteration)

$$\tilde{J}(\delta x) = \frac{1}{2}\|L\delta x - b\|^2_{D^{-1}} + \frac{1}{2}\|\mathrm{H}\delta x - d\|^2_{\mathrm{R}^{-1}}$$

with

$$L = \begin{bmatrix} I & & & \\ -M_1 & I & & \\ & \ddots & \ddots & \\ & & -M_N & I \end{bmatrix}$$

$$D = \begin{bmatrix} B & & & \\ & Q_1 & & \\ & & \ddots & \\ & & & Q_N \end{bmatrix}, \quad \mathrm{R} = \begin{bmatrix} R_0 & & & \\ & R_1 & & \\ & & \ddots & \\ & & & R_N \end{bmatrix}$$

$$\mathrm{H} = \begin{bmatrix} H_0 & & & \\ & H_1 & & \\ & & \ddots & \\ & & & H_N \end{bmatrix}, b = \begin{bmatrix} x_0^b - x_0 \\ \mathcal{M}_1(x_0) - x_1 \\ \vdots \\ \mathcal{M}_N(x_{N-1}) - x_N \end{bmatrix}, d = \begin{bmatrix} y_0 - \mathcal{H}_0(x_0) \\ y_1 - \mathcal{H}_1(x_1) \\ \vdots \\ y_N - \mathcal{H}_N(x_N) \end{bmatrix}.$$

State formulation and saddle formulation

$$\tilde{J}(\delta x) = \frac{1}{2}\|L\delta x - b\|^2_{D^{-1}} + \frac{1}{2}\|\mathrm{H}\delta x - d\|^2_{\mathrm{R}^{-1}}$$

Minimise

$$\nabla\tilde{J}(\delta x) = L^T D^{-1}(L\delta x - b) + \mathrm{H}^T\mathrm{R}^{-1}(\mathrm{H}\delta x - d) = 0.$$

$$(L^T D^{-1} L + \mathrm{H}^T\mathrm{R}^{-1}\mathrm{H})\delta x = L^T D^{-1} b + \mathrm{H}^T\mathrm{R}^{-1} d$$

with $\lambda = D^{-1}(b - L\delta x)$, $\mu = \mathrm{R}^{-1}(d - \mathrm{H}\delta x)$ (or **writing the problem with equality constraints and using KKT conditions**) we obtain

$$\begin{aligned} \nabla\tilde{J} = L^T\lambda + \mathrm{H}^T\mu &= 0, \\ D\lambda + L\delta x &= b, \\ \mathrm{R}\mu + \mathrm{H}\delta x &= d. \end{aligned}$$

Saddle Point Formulation

$$\begin{aligned} \nabla \tilde{J} = L^T \lambda + \mathrm{H}^T \mu &= 0, \\ D\lambda + L\delta x &= b, \\ \mathrm{R}\mu + \mathrm{H}\delta x &= d. \end{aligned}$$

Saddle point formulation of 4D-Var

$$\begin{bmatrix} D & 0 & L \\ 0 & \mathrm{R} & \mathrm{H} \\ L^T & \mathrm{H}^T & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ \mu \\ \delta x \end{bmatrix} = \begin{bmatrix} b \\ d \\ 0 \end{bmatrix}$$

Saddle Point Formulation

$$\begin{aligned} \nabla \tilde{J} = L^T \lambda + \mathrm{H}^T \mu &= 0, \\ D\lambda + L\delta x &= b, \\ \mathrm{R}\mu + \mathrm{H}\delta x &= d. \end{aligned}$$

Saddle point formulation of 4D-Var

$$\begin{bmatrix} D & 0 & L \\ 0 & \mathrm{R} & \mathrm{H} \\ L^T & \mathrm{H}^T & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ \mu \\ \delta x \end{bmatrix} = \begin{bmatrix} b \\ d \\ 0 \end{bmatrix}$$

- L integration of a numerical model, L^T its adjoint
- H, L computationally expensive!
- D, R are large, but cheaper to apply than a model evaluation
- saddle point matrix is symmetric indefinite
- preconditioned MINRES or GMRES.

Outline

The Kronecker product

Let $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$ be matrices of appropriate size.
Properties of the Kronecker product and $\mathrm{vec}\,(\cdot)$ operator:

$$\mathcal{A} \otimes \mathcal{B} = \begin{bmatrix} a_{11}\mathcal{B} & \cdots & a_{1n}\mathcal{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\mathcal{B} & \cdots & a_{mn}\mathcal{B} \end{bmatrix} \quad \mathrm{vec}\,(\mathcal{C}) = \begin{bmatrix} c_{11} \\ \vdots \\ c_{1n} \\ \vdots \\ c_{mn} \end{bmatrix}.$$

Moreover

$$(\mathcal{B}^T \otimes \mathcal{A})\mathrm{vec}\,(\mathcal{C}) = \mathrm{vec}\,(\mathcal{A}\mathcal{C}\mathcal{B})\,.$$

Kronecker formulation

Saddle point formulation of 4D-Var

$$\begin{bmatrix} D & 0 & L \\ 0 & \mathrm{R} & \mathrm{H} \\ L^T & \mathrm{H}^T & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ \mu \\ \delta x \end{bmatrix} = \begin{bmatrix} b \\ d \\ 0 \end{bmatrix}$$

Assume $Q_k = Q$, $R_k = R$, $H_k = H$, $M_k = M$, and number of observations $p_k = p$ for each k. Define

$$C = \begin{bmatrix} 0 & & & \\ -1 & 0 & & \\ & \ddots & \ddots & \\ & & -1 & 0 \end{bmatrix}, \quad E_1 = \begin{bmatrix} 1 & & & \\ & 0 & & \\ & & \ddots & \\ & & & 0 \end{bmatrix}, \quad E_2 = \begin{bmatrix} 0 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}.$$

Kronecker saddle point formulation of 4D-Var

$$\begin{bmatrix} E_1 \otimes B + E_2 \otimes Q & 0 & I_{N+1} \otimes I_n + C \otimes M \\ 0 & I_{N+1} \otimes R & I_{N+1} \otimes H \\ I_{N+1} \otimes I_n + C^T \otimes M^T & I_{N+1} \otimes H^T & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ \mu \\ \delta x \end{bmatrix} = \begin{bmatrix} b \\ d \\ 0 \end{bmatrix},$$

Simultaneous matrix equations

Kronecker saddle point formulation of 4D-Var

$$\begin{bmatrix} E_1 \otimes B + E_2 \otimes Q & 0 & I_{N+1} \otimes I_n + C \otimes M \\ 0 & I_{N+1} \otimes R & I_{N+1} \otimes H \\ I_{N+1} \otimes I_n + C^T \otimes M^T & I_{N+1} \otimes H^T & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ \mu \\ \delta x \end{bmatrix} = \begin{bmatrix} b \\ d \\ 0 \end{bmatrix},$$

Using $(\mathcal{B}^T \otimes \mathcal{A})\mathrm{vec}\,(\mathcal{C}) = \mathrm{vec}\,(\mathcal{A}\mathcal{C}\mathcal{B})$:

Simultaneous matrix equations

Kronecker saddle point formulation of 4D-Var

$$\begin{bmatrix} E_1 \otimes B + E_2 \otimes Q & 0 & I_{N+1} \otimes I_n + C \otimes M \\ 0 & I_{N+1} \otimes R & I_{N+1} \otimes H \\ I_{N+1} \otimes I_n + C^T \otimes M^T & I_{N+1} \otimes H^T & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ \mu \\ \delta x \end{bmatrix} = \begin{bmatrix} b \\ d \\ 0 \end{bmatrix},$$

Using $(\mathcal{B}^T \otimes \mathcal{A})\text{vec}\,(\mathcal{C}) = \text{vec}\,(\mathcal{ACB})$:

Simultaneous matrix equations

$$\begin{aligned} B\Lambda E_1 + Q\Lambda E_2 + X + MXC^T &= \mathbb{b}, \\ RU + HX &= \mathbb{d}, \\ \Lambda + M^T\Lambda C + H^T U &= 0. \end{aligned}$$

where $\lambda, \delta x, b, \mu$ and d are vectorised forms of the matrices $\Lambda, X, \mathbb{b} \in \mathbb{R}^{n\times N+1}$ and $U, \mathbb{d} \in \mathbb{R}^{p\times N+1}$ respectively.

Simultaneous matrix equations

Kronecker saddle point formulation of 4D-Var

$$\begin{bmatrix} E_1 \otimes B + E_2 \otimes Q & 0 & I_{N+1} \otimes I_n + C \otimes M \\ 0 & I_{N+1} \otimes R & I_{N+1} \otimes H \\ I_{N+1} \otimes I_n + C^T \otimes M^T & I_{N+1} \otimes H^T & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ \mu \\ \delta x \end{bmatrix} = \begin{bmatrix} b \\ d \\ 0 \end{bmatrix},$$

Using $(\mathcal{B}^T \otimes \mathcal{A})\mathrm{vec}\,(\mathcal{C}) = \mathrm{vec}\,(\mathcal{A}\mathcal{C}\mathcal{B})$:

Simultaneous matrix equations

$$\begin{aligned} B\Lambda E_1 + Q\Lambda E_2 + X + MXC^T &= \mathbb{b}, \\ RU + HX &= \mathbb{d}, \\ \Lambda + M^T\Lambda C + H^T U &= 0. \end{aligned}$$

where $\lambda, \delta x, b, \mu$ and d are vectorised forms of the matrices $\Lambda, X, \mathbb{b} \in \mathbb{R}^{n \times N+1}$ and $U, \mathbb{d} \in \mathbb{R}^{p \times N+1}$ respectively.

Suppose that the matrices Λ, U, X have low-rank representations,

$$\Lambda = W_\Lambda V_\Lambda^T, \quad U = W_U V_U^T, \quad X = W_X V_X^T.$$

Low-Rank GMRES (LR-GMRES)

GMRES for solving a linear system $Ax = b$

- Krylov subspace $\mathcal{K}_k(A, b) = \mathrm{span}\{b, Ab, \cdots, A^{k-1}b\}$
- Gram-Schmidt orthogonalisation

We need:

- Vector addition,
- Matrix vector products,
- Inner products.

Input: Choose x_0, compute $r_0 = b - Ax_0$ and $v_1 = r_0/\|r_0\|$;
Output: Solution of linear system $Ax = b$.

1. for $j = 1, 2, \ldots, k$ do
2. Compute $h_{ij} = \langle Av_j, v_i \rangle$ for $i = 1, 2, \ldots, j$
3. Compute $\tilde{v}_{j+1} = Av_j - \Sigma_{i=1}^{j} h_{ij} v_i$
4. Compute $h_{j+1,j} = \|\tilde{v}_{j+1}\|_2$
5. $v_{j+1} = \tilde{v}_{j+1}/h_{j+1,j}$
6. end for
7. $x_k = x_0 + V_k y_k$.

Low-Rank GMRES (LR-GMRES)

GMRES for solving a linear system $Ax = b$

- Krylov subspace $\mathcal{K}_k(A, b) = \mathrm{span}\{b, Ab, \cdots, A^{k-1}b\}$
- Gram-Schmidt orthogonalisation

We need:

- Vector addition,
- Matrix vector products,
- Inner products.

Input: Choose x_0, compute $r_0 = b - Ax_0$ and $v_1 = r_0/\|r_0\|$;
Output: Solution of linear system $Ax = b$.

1. for $j = 1, 2, \ldots, k$ do
2. Compute $h_{ij} = \langle Av_j, v_i \rangle$ for $i = 1, 2, \ldots, j$
3. Compute $\tilde{v}_{j+1} = Av_j - \Sigma_{i=1}^{j} h_{ij} v_i$
4. Compute $h_{j+1,j} = \|\tilde{v}_{j+1}\|_2$
5. $v_{j+1} = \tilde{v}_{j+1}/h_{j+1,j}$
6. end for
7. $x_k = x_0 + V_k y_k$.

Low-Rank GMRES (LR-GMRES)

$$\Lambda = W_\Lambda V_\Lambda^T, \quad U = W_U V_U^T, \quad X = W_X V_X^T.$$

Matrix vector products

$$\begin{aligned} B\Lambda E_1 + Q\Lambda E_2 + X + MXC^T &= \mathbb{b}, \\ RU + HX &= \mathbb{d}, \\ \Lambda + M^T\Lambda C + H^T U &= 0. \end{aligned}$$

becomes

$$\begin{aligned} \begin{bmatrix} BW_\Lambda & QW_\Lambda & W_X & MW_X \end{bmatrix} \begin{bmatrix} E_1V_\Lambda & E_2V_\Lambda & V_X & CV_X \end{bmatrix}^T &= \mathbb{b}, \\ \begin{bmatrix} RW_U & HW_X \end{bmatrix} \begin{bmatrix} V_U & W_X \end{bmatrix}^T &= \mathbb{d}, \\ \begin{bmatrix} W_\Lambda & M^TW_\Lambda & H^TW_U \end{bmatrix} \begin{bmatrix} V_\Lambda & C^TV_\Lambda & V_U \end{bmatrix}^T &= 0. \end{aligned}$$

Low-Rank GMRES (LR-GMRES)

$$\Lambda = W_\Lambda V_\Lambda^T, \quad U = W_U V_U^T, \quad X = W_X V_X^T.$$

Matrix vector products

Algorithm 2 Matrix multiplication (`Amult`)

Input: $W_{11}, W_{12}, W_{21}, W_{22}, W_{31}, W_{32}$

Output: $Z_{11}, Z_{12}, Z_{21}, Z_{22}, Z_{31}, Z_{32}$

$$Z_{11} = [BW_{11}, \quad QW_{11}, \quad W_{31}, \quad MW_{31}],$$
$$Z_{12} = [E_1W_{12}, \quad E_2W_{12}, \quad W_{32}, \quad CW_{32}],$$
$$Z_{21} = [RW_{21}, \quad HW_{31}],$$
$$Z_{21} = [W_{22}, \quad W_{32}],$$
$$Z_{31} = [W_{11}, \quad M^TW_{11}, \quad H^TW_{21}],$$
$$Z_{32} = [W_{12}, \quad C^TW_{12}, \quad W_{22}]$$

Low-Rank GMRES (LR-GMRES)

Suppose that the matrices Λ, U, X have low-rank representations,

$$\Lambda = W_\Lambda V_\Lambda^T, \quad U = W_U V_U^T, \quad X = W_X V_X^T.$$

Vectors z in GMRES become:

$$\mathrm{vec}\left(\begin{bmatrix} W_\Lambda V_\Lambda^T \\ W_U V_U^T \\ W_X V_X^T \end{bmatrix}\right) = \mathrm{vec}\left(\begin{bmatrix} Z_{11} Z_{12}^T \\ Z_{21} Z_{22}^T \\ Z_{31} Z_{32}^T \end{bmatrix}\right) = z.$$

Vector addition

$X_{k1} = [Y_{k1}, \quad Z_{k1}]$, $X_{k2} = [Y_{k2}, \quad Z_{k2}]$ for $k = 1, 2, 3$:

$$x = \mathrm{vec}\left(\begin{bmatrix} X_{11} X_{12}^T \\ X_{21} X_{22}^T \\ X_{31} X_{32}^T \end{bmatrix}\right) = \mathrm{vec}\left(\begin{bmatrix} Y_{11} Y_{12}^T + Z_{11} Z_{12}^T \\ Y_{21} Y_{22}^T + Z_{21} Z_{22}^T \\ Y_{31} Y_{32}^T + Z_{31} Z_{32}^T \end{bmatrix}\right) = y + z.$$

Low-Rank GMRES (LR-GMRES)

$$\mathrm{vec}\left(\begin{bmatrix} W_{11}W_{12}^T \\ W_{21}W_{22}^T \\ W_{31}W_{32}^T \end{bmatrix}\right) = w \quad \text{and} \quad \mathrm{vec}\left(\begin{bmatrix} V_{11}(V_{12})^T \\ V_{21}(V_{22})^T \\ V_{31}(V_{32})^T \end{bmatrix}\right) = v,$$

To compute the inner product $\langle w, v \rangle$ we use the trace:

$$\mathrm{vec}\,(\mathcal{A})^T\,\mathrm{vec}\,(\mathcal{B}) = \mathrm{trace}(\mathcal{A}^T\mathcal{B})$$

Low-Rank GMRES (LR-GMRES)

$$\operatorname{vec}\left(\begin{bmatrix} W_{11}W_{12}^T \\ W_{21}W_{22}^T \\ W_{31}W_{32}^T \end{bmatrix}\right) = w \quad \text{and} \quad \operatorname{vec}\left(\begin{bmatrix} V_{11}(V_{12})^T \\ V_{21}(V_{22})^T \\ V_{31}(V_{32})^T \end{bmatrix}\right) = v,$$

To compute the inner product $\langle w, v \rangle$ we use the trace:

$$\operatorname{vec}(\mathcal{A})^T \operatorname{vec}(\mathcal{B}) = \operatorname{trace}(\mathcal{A}^T \mathcal{B})$$

Inner products $\langle w, v \rangle$

$$\begin{aligned} \langle w, v \rangle \; &= \operatorname{trace}\left(W_{11}^T V_{11}(V_{12})^T W_{12}\right) + \operatorname{trace}\left(W_{21}^T V_{21}(V_{22})^T W_{22}\right) \\ &+ \operatorname{trace}\left(W_{31}^T V_{31}(V_{32})^T W_{32}\right). \end{aligned}$$

Low-Rank GMRES (LR-GMRES)

$$\mathrm{vec}\left(\begin{bmatrix} W_{11}W_{12}^T \\ W_{21}W_{22}^T \\ W_{31}W_{32}^T \end{bmatrix}\right) = w \quad \text{and} \quad \mathrm{vec}\left(\begin{bmatrix} V_{11}(V_{12})^T \\ V_{21}(V_{22})^T \\ V_{31}(V_{32})^T \end{bmatrix}\right) = v,$$

To compute the inner product $\langle w, v \rangle$ we use the trace:

$$\mathrm{vec}\,(\mathcal{A})^T\,\mathrm{vec}\,(\mathcal{B}) = \mathrm{trace}(\mathcal{A}^T\mathcal{B})$$

Inner products $\langle w, v \rangle$

$$\begin{aligned}\langle w, v \rangle \; &= \mathrm{trace}\left(W_{11}^T V_{11} (V_{12})^T W_{12}\right) + \mathrm{trace}\left(W_{21}^T V_{21} (V_{22})^T W_{22}\right) \\ &+ \mathrm{trace}\left(W_{31}^T V_{31} (V_{32})^T W_{32}\right).\end{aligned}$$

Truncating after concatenation, gives a low-rank implementation of GMRES.

Existence of a low-rank solution

Tensor rank

Let $x = \mathrm{vec}\,(X) \in \mathbb{R}^{n^2}$. The minimal number r such that

$$x = \sum_{i=1}^{r} u_i \otimes v_i,$$

where $u_i, v_i \in \mathbb{R}^n$ is called the *tensor rank* of the vector x.

Tensor rank and standard rank

Let $x \in \mathbb{R}^{n^2}$ be the vectorisation of $X \in \mathbb{R}^{n \times n}$, such that $x = \mathrm{vec}\,(X)$. The tensor rank of the vector x is equal to the rank of the matrix X.

Existence of a low-rank solution

Tensor rank

Let $x = \text{vec}(X) \in \mathbb{R}^{n^2}$. The minimal number r such that

$$x = \sum_{i=1}^{r} u_i \otimes v_i,$$

where $u_i, v_i \in \mathbb{R}^n$ is called the *tensor rank* of the vector x.

Tensor rank and standard rank

Let $x \in \mathbb{R}^{n^2}$ be the vectorisation of $X \in \mathbb{R}^{n \times n}$, such that $x = \text{vec}(X)$. The tensor rank of the vector x is equal to the rank of the matrix X.

Theorem (Existence of low-rank solution)

$$\tilde{J}(\delta x) = \frac{1}{2}(L\delta x - b)^T D^{-1}(L\delta x - b) + \frac{1}{2}(H\delta x - d)^T R^{-1}(H\delta x - d).$$

- *M is invertible*
- *spectrum of $(-C \otimes I + I \otimes -M^{-1})$ is contained in a rectangle in $\mathbb{C}_-$*

Then δx can be approximated by a vector of tensor rank at most $4(2r+1)^2(\text{rank}(b) + p + 1)$. Here r arises from the quadrature approximation of L^{-1}, and p is the number of observations in the data assimilation problem.

Outline

One-dimensional advection-diffusion system

We consider the 1D-advection-diffusion problem:

$$\frac{\partial}{\partial t}u(x,t) = 0.1\,\frac{\partial^2}{\partial x^2}u(x,t) + 1.4\,\frac{\partial}{\partial x}u(x,t)$$

for $x \in [0,1]$, $t \in (0,T)$, subject to the boundary and initial conditions

$$\begin{aligned} u(0,t) &= 0, && t \in (0,T) \\ u(1,t) &= 0, && t \in (0,T) \\ u(x,0) &= \sin(\pi x), && x \in [0,1]. \end{aligned}$$

Crank-Nicolson scheme, $n = 100$, $\Delta t = 10^{-3}$. Assimilation window 200 time steps.

One-dimensional advection-diffusion system

Partial, noisy observations, $p = 20$, $B_{i,j} = 0.1 \exp(\frac{-|i-j|}{50})$, $Q = 10^{-4} I_{100}$, $R = 0.01 I_p$, saddle point matrix size $= 44,000$.

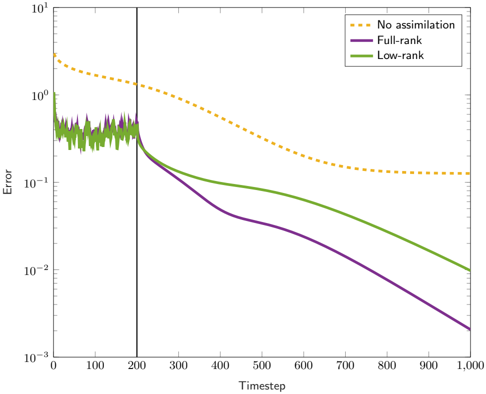


Figure: Root mean squared error for 1D advection-diffusion problem with partial, noisy observations ($r = 20$).

One-dimensional advection-diffusion system

Partial, noisy observations, $p = 20$, $B_{i,j} = 0.1\exp(\frac{-|i-j|}{50})$, $Q = 10^{-4} I_{100}$, $R = 0.01 I_p$, saddle point matrix size $= 44,000$.

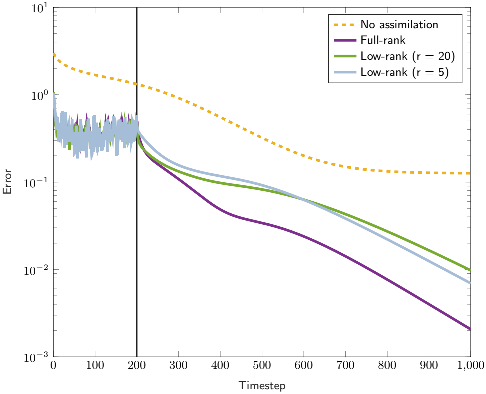


Figure: Root mean squared error for 1D advection-diffusion problem with partial, noisy observations $(r = 20, 5)$.

One-dimensional advection-diffusion system

Partial, noisy observations, $p = 20$, $B_{i,j} = 0.1\exp(\frac{-|i-j|}{50})$, $Q = 10^{-4}I_{100}$, $R = 0.01I_p$, saddle point matrix size $= 44,000$.

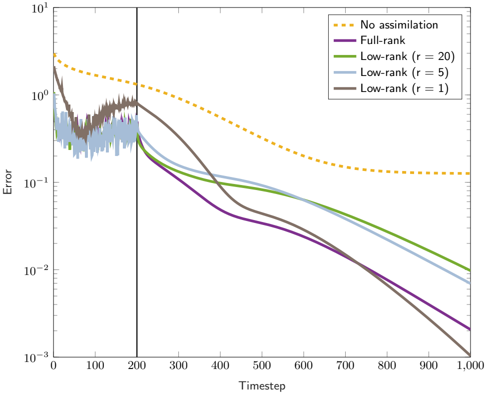


Figure: Root mean squared error for 1D advection-diffusion problem with partial, noisy observations $(r = 20, 5, 1)$.

One-dimensional advection-diffusion system

n	N	p	rank	# of matrix elements in full-rank solution	low-rank solution	storage reduction
100	199	100	20	20,000	6,000	70%
500	199	500	20	100,000	14,000	86%
500	199	100	20	100,000	14,000	86%
500	199	100	5	100,000	3,500	96.5%
500	199	100	1	100,000	700	99.3%

Table: Storage requirements for full- and low-rank methods in the advection-diffusion equation examples.

Solver	runtime (s)
GMRES	9.0055
LR-GMRES (rank 50)	12.9397
LR-GMRES (rank 20)	2.5673
LR-GMRES (rank 5)	0.5909
LR-GMRES (rank 1)	0.3127

Table: Comparison of computation time for low-rank GMRES for advection-diffusion.

Extension to time-dependent systems

Kronecker saddle point formulation of 4D-Var

$$\begin{bmatrix} E_1 \otimes B + E_2 \otimes Q & 0 & I_{N+1} \otimes I_n + C \otimes M \\ 0 & I_{N+1} \otimes R & I_{N+1} \otimes H \\ I_{N+1} \otimes I_n + C^T \otimes M^T & I_{N+1} \otimes H^T & 0 \end{bmatrix} \begin{bmatrix} \lambda \\ \mu \\ \delta x \end{bmatrix} = \begin{bmatrix} b \\ d \\ 0 \end{bmatrix},$$

Extension to time-dependent systems

For time-dependent operators we can rewrite the Kronecker saddle point matrix as

Time-dependent Kronecker saddle point formulation

$$\begin{bmatrix} F_1 \otimes B + \sum_{i=1}^{N} F_{i+1} \otimes Q_i & 0 & I_{N+1} \otimes I_n + \sum_{i=1}^{N} C_i \otimes M_i \\ 0 & \sum_{i=0}^{N} F_{i+1} \otimes R_i & \sum_{i=0}^{N} F_{i+1} \otimes H_i \\ I_{N+1} \otimes I_n + \sum_{i=1}^{N} C_i^T \otimes M_i^T & \sum_{i=0}^{N} F_{i+1} \otimes H_i^T & 0 \end{bmatrix},$$

Here

- F_i only has 1 on the ith entry of the diagonal,
- C_i only has -1 on the ith column of the subdiagonal.

Lorenz-95 example

The model is defined by a system of n non-linear ODEs

$$\frac{\mathrm{d}x^i}{\mathrm{d}t} = -x^{i-2}x^{i-1} + x^{i-1}x^{i+1} - x^i + f,$$

where $x = [x^1, x^2, \ldots, x^n]^T$ is the state, and f is a forcing term.
We take $n = 150$, with noisy observations at each point, over 150 timesteps.

Lorenz-95 example

Noisy observations, $p = 150$, $B_{i,j} = 0.1\exp(\frac{-|i-j|}{50})$, $Q = 10^{-4} I_{150}$, $R = 0.01 I_p$, saddle point matrix size $= 67,500$.

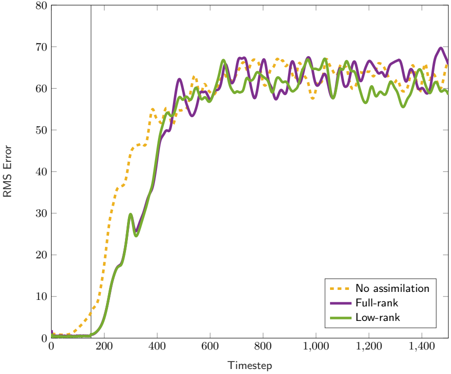


Figure: Root mean squared error for 150-dimensional Lorenz-95 system with noisy observations $(r = 20)$.

Lorenz-95 example

Noisy observations, $p = 150$, $B_{i,j} = 0.1\exp(\frac{-|i-j|}{50})$, $Q = 10^{-4}I_{150}$, $R = 0.01I_p$, saddle point matrix size $= 67,500$.

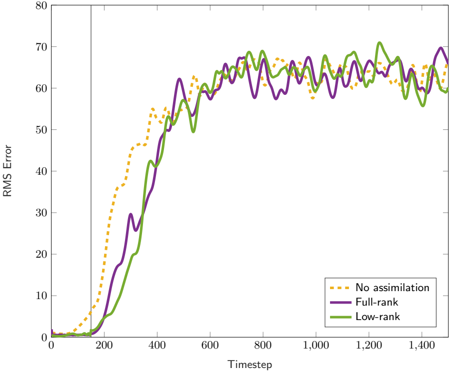


Figure: Root mean squared error for 150-dimensional Lorenz-95 system with noisy observations $(r = 5)$.

Lorenz-95 example

Experimenting with different rank choices, we have achieved the following reductions:

n	N	p	rank	# of matrix elements in full-rank solution	low-rank solution	storage reduction
40	199	40	20	8,000	4,800	40%
40	199	8	20	8,000	4,800	40%
500	199	500	20	100,000	14,000	86%
500	199	500	5	100,000	3,500	96.5%

Table: Storage requirements for full- and low-rank methods in the Lorenz-95 examples.

Outline

Conclusions and future work

Conclusions

- Weak constraint 4D-Var is a very large optimisation problem.
- It can be shown that under certain assumptions low-rank solutions exist.
- Preconditioning may not be necessary, with the low-rank approach acting like a regularisation.
- Very large reduction in storage and computing time.

Conclusions and future work

Conclusions

- Weak constraint 4D-Var is a very large optimisation problem.
- It can be shown that under certain assumptions low-rank solutions exist.
- Preconditioning may not be necessary, with the low-rank approach acting like a regularisation.
- Very large reduction in storage and computing time.

Future work

- Higher dimensional examples
- Better theoretical foundation (inexact GMRES theory)

References

M. Stoll and T. Breiten, *A low-rank in time approach to PDE-constrained optimization*, SIAM J. Sci. Comput., 37 (2015), pp. B1–B29.

L. Grasedyck, *Existence of a low rank or $\mathcal{H}$-matrix approximant to the solution of a Sylvester equation*, Numer. Linear Algebra Appl., 11 (2004), pp. 371–389.

Y. Saad and M. H. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Comput., 7 (1986), pp. 856–869.

M.A. Freitag and D.L.H. Green, *A low-rank approach to the solution of weak constraint variational data assimilation problems*, J. Comput. Phys., 357 (2018), pp. 263–281.

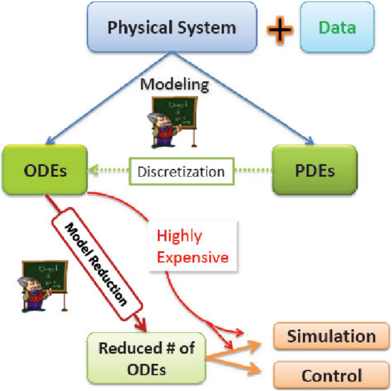
M. Fisher, S. Gratton, S. Gürol, Y. Trémolet, and X. Vasseur, *Low rank updates in preconditioning the saddle point systems arising from data assimilation problems*, Optimization Methods and Software, 33(1), (2018), pp. 45–69.

Part II

Balanced truncation within weak constraint 4D-Var

Outline

Model order reduction



- Given a physical model with dynamics described by states $x \in \mathbb{R}^n$ where n is large.
- Describe the dynamics of the system using a reduced number of states ($\ll n$).
- Should be available at significantly lower cost/storage.
- Can be used for simulation, prediction, optimisation, data assimilation, ...

Linear time invariant systems

Linear time invariant system

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

Coefficient matrices

- system matrix $A \in \mathbb{R}^{n \times n}$,
- input matrix $B \in \mathbb{R}^{n \times m}$,
- output matrix $C \in \mathbb{R}^{p \times n}$.

Input/output/state vectors

- state vector $x(t) \in \mathbb{R}^n$ with $x(t_0) = x_0$
- input vector/control $u(t) \in \mathbb{R}^n$
- output $y(t) \in \mathbb{R}^p$

Properties

- n is the order of the system

Linear time invariant systems

Linear time invariant system

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

Coefficient matrices

- system matrix $A \in \mathbb{R}^{n \times n}$,
- input matrix $B \in \mathbb{R}^{n \times m}$,
- output matrix $C \in \mathbb{R}^{p \times n}$.

Input/output/state vectors

- state vector $x(t) \in \mathbb{R}^n$ with $x(t_0) = x_0$
- input vector/control $u(t) \in \mathbb{R}^n$
- output $y(t) \in \mathbb{R}^p$

Properties

- n is the order of the system

Problem

Many modern applications lead to large systems orders n, e.g. $n \approx 10^6$ or higher $\Rightarrow$ very high computations costs!

Linear time invariant systems

Linear time invariant system

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

$A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$,
$x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^p$.

Linear time invariant systems

Linear time invariant system

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

$A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$, $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^p$.

$\longrightarrow$

Model order reduction

$$\dot{\tilde{x}}(t) = \tilde{A}\tilde{x}(t) + \tilde{B}u(t)$$
$$\tilde{y}(t) = \tilde{C}\tilde{x}(t)$$

$\tilde{A} \in \mathbb{R}^{r \times r}$, $\tilde{B} \in \mathbb{R}^{r \times m}$, $\tilde{C} \in \mathbb{R}^{p \times r}$, $\tilde{x}(t) \in \mathbb{R}^r$, $u(t) \in \mathbb{R}^m$ and $\tilde{y}(t) \in \mathbb{R}^p$ such that

$$\tilde{y}(t) \approx y(t)$$

and

$$r \ll n.$$

Model reduction by projection

Approximate state variable $x(t)$ in a reduced basis, e.g. $x(t) \approx V\tilde{x}(t)$ for some $V \in \mathbb{R}^{n\times r}$ and $r \ll n$:

$$V\dot{\tilde{x}}(t) \approx AV\tilde{x}(t) + Bu(t)$$
$$\tilde{y}(t) = CV\tilde{x}(t)$$

Model reduction by projection

Approximate state variable $x(t)$ in a reduced basis, e.g. $x(t) \approx V\tilde{x}(t)$ for some $V \in \mathbb{R}^{n\times r}$ and $r \ll n$:

$$V\dot{\tilde{x}}(t) \approx AV\tilde{x}(t) + Bu(t)$$
$$\tilde{y}(t) = CV\tilde{x}(t)$$

Let $W^TV = I \in \mathbb{R}^{r\times r}$, $W \in \mathbb{R}^{n\times r}$ and require Petrov-Galerkin condition:

$$W^T \left(V\dot{\tilde{x}}(t) - (AV\tilde{x}(t) + Bu(t))\right) = 0.$$

Model reduction by projection

Approximate state variable $x(t)$ in a reduced basis, e.g. $x(t) \approx V\tilde{x}(t)$ for some $V \in \mathbb{R}^{n \times r}$ and $r \ll n$:

$$V\dot{\tilde{x}}(t) \approx AV\tilde{x}(t) + Bu(t)$$
$$\tilde{y}(t) = CV\tilde{x}(t)$$

Let $W^TV = I \in \mathbb{R}^{r \times r}$, $W \in \mathbb{R}^{n \times r}$ and require Petrov-Galerkin condition:

$$W^T\left(V\dot{\tilde{x}}(t) - (AV\tilde{x}(t) + Bu(t))\right) = 0.$$

Projection methods

$$\dot{\tilde{x}}(t) = \tilde{A}\tilde{x}(t) + \tilde{B}u(t)$$
$$\tilde{y}(t) = \tilde{C}\tilde{x}(t)$$

where $\tilde{A} = W^TAV \in \mathbb{R}^{r \times r}$, $\tilde{B} = W^TB \in \mathbb{R}^{r \times m}$ and $\tilde{C} = CV \in \mathbb{R}^{p \times r}$

Model reduction by projection

Approximate state variable $x(t)$ in a reduced basis, e.g. $x(t) \approx V\tilde{x}(t)$ for some $V \in \mathbb{R}^{n \times r}$ and $r \ll n$:

$$V\dot{\tilde{x}}(t) \approx AV\tilde{x}(t) + Bu(t)$$
$$\tilde{y}(t) = CV\tilde{x}(t)$$

Let $W^T V = I \in \mathbb{R}^{r \times r}$, $W \in \mathbb{R}^{n \times r}$ and require Petrov-Galerkin condition:

$$W^T \left(V\dot{\tilde{x}}(t) - (AV\tilde{x}(t) + Bu(t))\right) = 0.$$

Projection methods

$$\dot{\tilde{x}}(t) = \tilde{A}\tilde{x}(t) + \tilde{B}u(t)$$
$$\tilde{y}(t) = \tilde{C}\tilde{x}(t)$$

where $\tilde{A} = W^T AV \in \mathbb{R}^{r \times r}$, $\tilde{B} = W^T B \in \mathbb{R}^{r \times m}$ and $\tilde{C} = CV \in \mathbb{R}^{p \times r}$

Need to find projection matrices V and W!

Balanced Truncation - controllability/observability for deterministic case

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0$$
$$y(t) = Cx(t)$$

Observability

- suppose $u(t) = 0$ for all $t \in [0; T]$

Balanced Truncation - controllability/observability for deterministic case

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0$$
$$y(t) = Cx(t)$$

Observability

- suppose $u(t) = 0$ for all $t \in [0; T] \Rightarrow y(t) = Ce^{tA}x_0$

Balanced Truncation - controllability/observability for deterministic case

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0$$
$$y(t) = Cx(t)$$

Observability

- suppose $u(t) = 0$ for all $t \in [0;T] \Rightarrow y(t) = Ce^{tA}x_0$
- gauge how easy the initial state x_0 can be observed by the energy that state produces (output) over the interval $[0;T]$: the more energy the state produces, the easier it is to observe:

$$\int_0^T \|y(t)\|^2 dt = \int_0^T x_0^T e^{tA^T} C^T C e^{tA} x_0 dt = x_0^T Q_T x_0$$

where $Q_T = \int_0^T e^{tA^T} C^T C e^{tA} dt$

Balanced Truncation - controllability/observability for deterministic case

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0$$
$$y(t) = Cx(t)$$

Observability

- suppose $u(t) = 0$ for all $t \in [0;T] \Rightarrow y(t) = Ce^{tA}x_0$
- gauge how easy the initial state x_0 can be observed by the energy that state produces (output) over the interval $[0;T]$: the more energy the state produces, the easier it is to observe:

$$\int_0^T \|y(t)\|^2 dt = \int_0^T x_0^T e^{tA^T} C^T C e^{tA} x_0 dt = x_0^T Q_T x_0$$

where $Q_T = \int_0^T e^{tA^T} C^T C e^{tA} dt$

Controllability/Reachability

- amount of energy required (by input) to steer x_0 to the target x_T.

Balanced Truncation - controllability/observability for deterministic case

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0$$
$$y(t) = Cx(t)$$

Observability

- suppose $u(t) = 0$ for all $t \in [0; T] \Rightarrow y(t) = Ce^{tA}x_0$
- gauge how easy the initial state x_0 can be observed by the energy that state produces (output) over the interval $[0; T]$: the more energy the state produces, the easier it is to observe:

$$\int_0^T \|y(t)\|^2 dt = \int_0^T x_0^T e^{tA^T} C^T C e^{tA} x_0 dt = x_0^T Q_T x_0$$

where $Q_T = \int_0^T e^{tA^T} C^T C e^{tA} dt$

Controllability/Reachability

- amount of energy required (by input) to steer x_0 to the target x_T.
- similar derivation gives

$$\int_0^T \|u(t)\|^2 dt = x_T^T P_T^{-1} x_T \quad \text{where} \quad P_T = \int_0^T e^{tA} B B^T e^{tA^T} dt.$$

Balanced Truncation - controllability and observability

Controllability

Let A be stable. The unique solution P of the Lyapunov equation

$$AP + PA^T = -BB^T$$

is positive definite if and only if the pair (A, B) is controllable.

$$P = \int_0^\infty e^{A\tau} BB^T e^{A^T\tau} \, d\tau \quad \text{Controllability Gramian.}$$

Balanced Truncation - controllability and observability

Controllability

Let A be stable. The unique solution P of the Lyapunov equation

$$AP + PA^T = -BB^T$$

is positive definite if and only if the pair (A, B) is controllable.

$$P = \int_0^\infty e^{A\tau} BB^T e^{A^T\tau}\, d\tau \quad \text{Controllability Gramian.}$$

Observability

Let A be stable. The unique solution Q of the Lyapunov equation

$$A^T Q + QA = -C^T C$$

is positive definite if and only if the pair (A, C) is observable.

$$Q = \int_0^\infty e^{A^T\tau} C^T C e^{A\tau}\, d\tau \quad \text{Observability Gramian.}$$

Balanced Truncation

Idea behind Balanced Truncation

- States that are difficult to reach have large components in the span of the eigenvectors corresponding to small eigenvalues of the reachability Gramian P
- States that are difficult to observe have large components in the span of eigenvectors corresponding to small eigenvalues of the observability Gramian Q

Balanced Truncation

Idea behind Balanced Truncation

- States that are difficult to reach have large components in the span of the eigenvectors corresponding to small eigenvalues of the reachability Gramian P
- States that are difficult to observe have large components in the span of eigenvectors corresponding to small eigenvalues of the observability Gramian Q
- eliminates states that are both difficult to reach and difficult to observe.

Balanced Truncation

Idea behind Balanced Truncation

- States that are difficult to reach have large components in the span of the eigenvectors corresponding to small eigenvalues of the reachability Gramian P
- States that are difficult to observe have large components in the span of eigenvectors corresponding to small eigenvalues of the observability Gramian Q
- eliminates states that are both difficult to reach and difficult to observe.
- find a basis in which the dominant reachable and observable states are the same

Balanced Truncation (BT)

Balanced System

A stable linear system

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

is called balanced if the observability/controllability Gramians P, Q from

$$AP + PA^T = -BB^T, \quad A^TQ + QA = -C^TC$$

satisfy $P = Q = diag(\sigma_1, \dots, \sigma_n)$ with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$ called Hankel Singular Values, given by $\sqrt{\lambda(PQ)} = \{\sigma_1, \dots, \sigma_n\} = \Sigma$.

Balanced Truncation (BT)

Balanced System

A stable linear system

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

is called balanced if the observability/controllability Gramians P, Q from

$$AP + PA^T = -BB^T, \quad A^TQ + QA = -C^TC$$

satisfy $P = Q = diag(\sigma_1, \ldots, \sigma_n)$ with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$ called Hankel Singular Values, given by $\sqrt{\lambda(PQ)} = \{\sigma_1, \ldots, \sigma_n\} = \Sigma$.

Balancing Transformation

Transformation $\tilde{x} = Tx$, $T \in \mathbb{R}^{n \times n}$, always exists if $P, Q > 0$ and can be chosen as

$$T = \Sigma^{-\frac{1}{2}} U^T L^T \quad \text{and} \quad T^{-1} = KV\Sigma^{-\frac{1}{2}},$$

where $\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_n)$ and $P = KK^T$, $Q = LL^T$ and $K^TL = V\Sigma U^T$.

Balanced Truncation (BT)

Balancing Transformation

Transformation $\tilde{x} = Tx$, $T \in \mathbb{R}^{n \times n}$, always exists if $P, Q > 0$ and can be chosen as

$$T = \Sigma^{-\frac{1}{2}} U^T L^T \quad \text{and} \quad T^{-1} = K V \Sigma^{-\frac{1}{2}},$$

where $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$ and $P = KK^T$, $Q = LL^T$ and $K^T L = V \Sigma U^T$ gives the required matrices.

$$(\tilde{A}, \tilde{B}, \tilde{C}) = (TAT^{-1}, TB, CT^{-1})$$

Balanced Gramians $\tilde{P} = TPT^T$ and $\tilde{Q} = T^{-T} Q T^{-1}$ which are equal and diagonal and

$$(\tilde{A}, \tilde{B}, \tilde{C}) = \left(\begin{bmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ \tilde{A}_{21} & \tilde{A}_{22} \end{bmatrix}, \begin{bmatrix} \tilde{B}_1 \\ \tilde{B}_2 \end{bmatrix}, \begin{bmatrix} \tilde{C}_1 & \tilde{C}_2 \end{bmatrix} \right).$$

Balanced Truncation (BT)

Balancing Transformation

Transformation $\tilde{x} = Tx$, $T \in \mathbb{R}^{n \times n}$, always exists if $P, Q > 0$ and can be chosen as

$$T = \Sigma^{-\frac{1}{2}} U^T L^T \quad \text{and} \quad T^{-1} = K V \Sigma^{-\frac{1}{2}},$$

where $\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_n)$ and $P = KK^T$, $Q = LL^T$ and $K^T L = V \Sigma U^T$ gives the required matrices.

$$(\tilde{A}, \tilde{B}, \tilde{C}) = (TAT^{-1}, TB, CT^{-1})$$

Balanced Gramians $\tilde{P} = TPT^T$ and $\tilde{Q} = T^{-T} Q T^{-1}$ which are equal and diagonal and

$$(\tilde{A}, \tilde{B}, \tilde{C}) = \left(\begin{bmatrix} \tilde{A}_{11} & \tilde{A}_{12} \\ \tilde{A}_{21} & \tilde{A}_{22} \end{bmatrix}, \begin{bmatrix} \tilde{B}_1 \\ \tilde{B}_2 \end{bmatrix}, \begin{bmatrix} \tilde{C}_1 & \tilde{C}_2 \end{bmatrix} \right).$$

Truncating

By truncating the discardable states, the truncated reduced system is then given by $(\tilde{A}_{11}, \tilde{B}_1, \tilde{C}_1)$.

Outline

Concise notation for incremental 4D-Var (all-at-once approach)

Minimise

$$\tilde{J}(\delta x) = \frac{1}{2}\|L\delta x - b\|^2_{D^{-1}} + \frac{1}{2}\|\mathrm{H}\delta x - d\|^2_{\mathrm{R}^{-1}}$$

with

$$L = \begin{bmatrix} I & & & \\ -M_1 & I & & \\ & \ddots & \ddots & \\ & & -M_N & I \end{bmatrix}$$

$$\mathrm{H} = \begin{bmatrix} H_0 & & & \\ & H_1 & & \\ & & \ddots & \\ & & & H_N \end{bmatrix}.$$

- L - all-at-once model operator over the assimilation window
- H - all-at-once observation operator

Balanced truncation for weak constraint 4D-Var

Idea: Project M_k and H_k onto lower dimensional subspace

$$\tilde{M}_k = W^T M_k V \in \mathbb{R}^{r \times r}$$
$$\tilde{H}_k = H_k V \in \mathbb{R}^{p_k \times r}$$

where W and V are obtained from Balanced Truncation.

Balanced truncation for weak constraint 4D-Var

Idea: Project M_k and H_k onto lower dimensional subspace

$$\tilde{M}_k = W^T M_k V \in \mathbb{R}^{r\times r}$$
$$\tilde{H}_k = H_k V \in \mathbb{R}^{p_k\times r}$$

where W and V are obtained from Balanced Truncation.
Projection of the covariance matrices:

$$\hat{B} = W^T B W, \quad \hat{Q}_k = W^T Q_k W$$

e.g.

$$\tilde{J}(\delta\hat{x}) = \frac{1}{2}\|\hat{L}\delta\hat{x} - \hat{b}\|^2_{\hat{D}^{-1}} + \frac{1}{2}\|d - \hat{H}\delta\hat{x}\|^2_{R_k^{-1}},$$

where $\delta\hat{x} = W^T\delta x$, $\hat{L}$, $\hat{H}$, etc projected versions of L, H.

Balanced truncation for weak constraint 4D-Var

Consider linear discrete system

$$\begin{aligned} \delta x_{-1} &= 0, \\ \delta x_{k+1} &= M \delta x_k + u_k, \\ d_k &= H \delta x_k, \end{aligned}$$

where, in the weak constraint data assimilation case, the inputs are:

$$u_k \sim \begin{cases} \mathcal{N}(0, B), & \text{for } k = -1 \\ \mathcal{N}(0, Q_k), & \text{for } k \geq 0. \end{cases}$$

Balanced truncation for weak constraint 4D-Var

Consider linear discrete system

$$\begin{aligned} \delta x_{-1} &= 0, \\ \delta x_{k+1} &= M\delta x_k + u_k, \\ d_k &= H\delta x_k, \end{aligned}$$

where, in the weak constraint data assimilation case, the inputs are:

$$u_k \sim \begin{cases} \mathcal{N}(0, B), & \text{for } k = -1 \\ \mathcal{N}(0, Q_k), & \text{for } k \geq 0. \end{cases}$$

Reachability and observability Gramians (for the discrete version)

$$\mathcal{G}_r = B + \sum_{j=1}^{\infty} M^j Q (M^T)^j,$$

$$\mathcal{G}_o = \sum_{j=0}^{\infty} (M^T)^j H^T R H M^j,$$

Balanced truncation for weak constraint 4D-Var

Approach: Solve discrete Lyapunov (or Stein) equations:

$$\mathcal{G}_r = M\mathcal{G}_r M^T + B + M(Q - B)M^T,$$
$$\mathcal{G}_o = M^T\mathcal{G}_o M + H^T R H.$$

Balanced truncation for weak constraint 4D-Var

Approach: Solve discrete Lyapunov (or Stein) equations:

$$\mathcal{G}_r = M\mathcal{G}_r M^T + B + M(Q - B)M^T,$$
$$\mathcal{G}_o = M^T \mathcal{G}_o M + H^T R H.$$

Decompose $\mathcal{G}_r = KK^T$, $\mathcal{G}_o = LL^T$ and compute SVD of

$$K^T L = Z\Sigma Y^T,$$

where $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$ are the Hankel singular values.

Balanced truncation for weak constraint 4D-Var

Approach: Solve discrete Lyapunov (or Stein) equations:

$$\mathcal{G}_r = M\mathcal{G}_r M^T + B + M(Q-B)M^T,$$
$$\mathcal{G}_o = M^T\mathcal{G}_o M + H^T R H.$$

Decompose $\mathcal{G}_r = KK^T$, $\mathcal{G}_o = LL^T$ and compute SVD of

$$K^T L = Z\Sigma Y^T,$$

where $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$ are the Hankel singular values.
Projection matrices are

$$V = KZ_r\Sigma_r^{-\frac{1}{2}} \in \mathbb{R}^{n\times r},$$
$$W = LY_r\Sigma_r^{-\frac{1}{2}} \in \mathbb{R}^{n\times r}.$$

Outline

One-dimensional advection-diffusion system

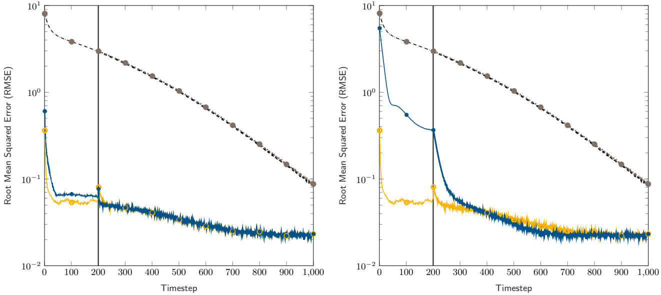
$$\frac{\partial}{\partial t}u(x,t) = 0.1\frac{\partial^2}{\partial x^2}u(x,t) + 1.4\frac{\partial}{\partial x}u(x,t)$$

for $x \in [0,1]$, $t \in (0,T)$, subject to the boundary and initial conditions

$$\begin{aligned} u(0,t) &= 0, && t \in (0,T) \\ u(1,t) &= 0, && t \in (0,T) \\ u(x,0) &= \sin(\pi x), && x \in [0,1]. \end{aligned}$$

Crank-Nicolson scheme, $n = 500$, $\Delta t = 10^{-3}$. Assimilation window 200 time steps.

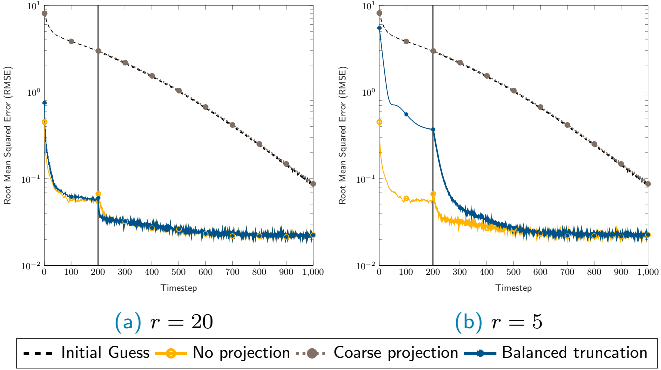
One-dimensional advection-diffusion system



(a) $r = 20$ (b) $r = 5$

Figure: RMS error for the 1D advection-diffusion example with full, noisy observations ($r = 20$, $r = 5$).

One-dimensional advection-diffusion system



(a) $r = 20$ (b) $r = 5$

Figure: RMS error for the 1D advection-diffusion example with partial, noisy observations ($r = 20$, $r = 5$).

One-dimensional advection-diffusion system

Projection method	Forming matrices	CG solve	Total
No proj.	0	5.0049	5.0049
BT ($r = 20$)	1.2271	0.1419	1.3690
Coarse proj. ($r = 20$)	0.0009	0.0208	0.0217
BT ($r = 5$)	1.1778	0.0467	1.2245
Coarse proj. ($r = 5$)	0.0007	0.0125	0.0132

Table: Computation time for 1D advection-diffusion equation example ($r = 20$, $r = 5$).

Outline

Conclusions and future work

Conclusions

- Balanced truncation effective reducing the dimension of forward model
- Expensive offline phase, cheap online computation
- Computable error bounds available
- Reduction in storage and computing time

Conclusions and future work

Conclusions

- Balanced truncation effective reducing the dimension of forward model
- Expensive offline phase, cheap online computation
- Computable error bounds available
- Reduction in storage and computing time

Future work

- Better methods for nonlinear problems (POD-DEIM)
- Online model reduction

References

A. C. Antoulas, *Approximation of large-scale dynamical systems, vol. 6*, SIAM (2005), pp. B1–B29.

A. S. Lawless, N. K. Nichols, C. Boess, A. Bunse-Gerstner, *Approximate Gauss–Newton methods for optimal state estimation using reduced-order models.*, Internat. J. Numer. Methods Fluids, 56(8) (2008), pp. 1367–1373.

C. Boess, A. S. Lawless, N. K. Nichols, A. Bunse-Gerstner, *State estimation using model order reduction for unstable systems.*, Comput. Fluids, 46(1) (2011), pp. 155–160.

M.A. Freitag and D.L.H. Green, *Projection methods for weak constraint variational data assimilation*, Submitted. 2019.

References

- A. C. Antoulas, *Approximation of large-scale dynamical systems, vol. 6*, SIAM (2005), pp. B1–B29.
- A. S. Lawless, N. K. Nichols, C. Boess, A. Bunse-Gerstner, *Approximate Gauss–Newton methods for optimal state estimation using reduced-order models.*, Internat. J. Numer. Methods Fluids, 56(8) (2008), pp. 1367–1373.
- C. Boess, A. S. Lawless, N. K. Nichols, A. Bunse-Gerstner, *State estimation using model order reduction for unstable systems.*, Comput. Fluids, 46(1) (2011), pp. 155–160.
- M.A. Freitag and D.L.H. Green, *Projection methods for weak constraint variational data assimilation*, Submitted. 2019.

Thank You!