Block matrix formulations for evolving networks

Caterina Fenu

AICES - RWTH Aachen University (Germany)

joint work with:

Desmond Higham (Strathclyde University, Glasgow (UK))

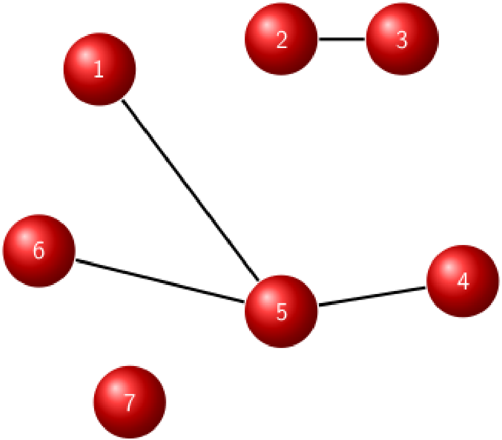
Numerical Linear Algebra and Applications (NL2A)

October 24 – 28, 2016

Outline

- Review some dynamic indices
- Introduce a new block matrix formulation
- Show some numerical results

Evolving networks



Day 1

Evolving networks

- We have:
  - M ordered time points $t_1 < t_2 < \cdots < t_M$.
  - M networks $G^{[k]} = \left(V, E^{[k]}\right)$ with associated adjacency matrices $A^{[k]}$, with $k = 1, \ldots, M$.
  - Dynamic indices already defined.

Evolving networks

- We have:
  - M ordered time points $t_1 < t_2 < \cdots < t_M$.
  - M networks $G^{[k]} = \left(V, E^{[k]}\right)$ with associated adjacency matrices $A^{[k]}$, with $k = 1, \ldots, M$.
  - Dynamic indices already defined.
- We want:
  - Unify the definitions already given.
  - Find a better way to compute them.

Dynamic walks

A dynamic walk of length ℓ from node i_1 to node $i_{\ell+1}$ consists of a sequence of edges $i_1 \to i_2$, $i_2 \to i_3$, ..., $i_\ell \to i_{\ell+1}$ and a nondecreasing sequence of times $t_{r_1} \leq t_{r_2} \leq \cdots \leq t_{r_\ell}$ such that $A^{[r_m]}_{i_m, i_{m+1}} \neq 0$.

Dynamic walks

A dynamic walk of length ℓ from node i_1 to node $i_{\ell+1}$ consists of a sequence of edges $i_1 \rightarrow i_2$, $i_2 \rightarrow i_3$, $\ldots$, $i_\ell \rightarrow i_{\ell+1}$ and a nondecreasing sequence of times $t_{r_1} \leq t_{r_2} \leq \cdots \leq t_{r_\ell}$ such that $A^{[r_m]}_{i_m, i_{m+1}} \neq 0$.

The (i,j)th element of the product $A^{[r_1]}A^{[r_2]} \cdots A^{[r_\ell]}$ counts the number of dynamic walks of length ℓ from node i to node j, where the mth step takes place at time t_{r_m}.

Dynamic Communicability Matrix

Penalizing each walk of length ℓ by a factor a^ℓ and summing for all the possible length we obtain the (i, j)th element of the matrix product

$$\mathcal{Q}^{[\ell]} = (I - aA^{[1]})^{-1}(I - aA^{[2]})^{-1} \cdots (I - aA^{[\ell]})^{-1}.$$

Thus, $\mathcal{Q}_{ij}^{[\ell]}$ is a summary on how well information can be passed from node i to node j.

Row and column sums

$$C_{\mathrm{broadcast}}^{[\ell]} = \mathcal{Q}^{[\ell]}\mathbf{1} \qquad \text{and} \qquad C_{\mathrm{receive}}^{[\ell]} = \mathcal{Q}^{[\ell]\,T}\mathbf{1}$$

are called the broadcast and receive communicabilies.

[Grindrod, Parsons, Higham, Estrada (2011)]

Running Dynamic Communicability Matrix

We can recursively define the matrix $\mathcal{S}^{[j]}$, starting from $\mathcal{S}^{[0]} = 0$, as

$$\mathcal{S}^{[j]} = \left(I + e^{-b\Delta t_j}\mathcal{S}^{[j-1]}\right)\left(I - aA^{[j]}\right)^{-1} - I, \qquad j = 1, \ldots, M,$$

where $\Delta t_j = t_j - t_{j-1}$.

Running versions of the broadcast and receive communicabilities are then given by the row/column sums of the matrix $\mathcal{S}^{[j]}$:

$$\mathcal{S}^{[j]}\mathbf{1} \qquad \text{and} \qquad {\mathcal{S}^{[j]}}^T\mathbf{1}.$$

[Grindrod, Higham, (2012)]

Nodal betweenness

The nodal betweenness of node r is defined as

$$\mathrm{NB}_r := \frac{1}{(n-1)^2-(n-1)} \sum_{i \neq j \neq r} \frac{(\mathcal{Q}^{[M]})_{ij} - (\bar{\mathcal{Q}}_r^{[M]})_{ij}}{(\mathcal{Q}^{[M]})_{ij}},$$

where

$$\bar{\mathcal{Q}}_r^{[M]} = \prod_{s=1}^{M} \left(I - a\bar{A}_r^{[s]} \right)^{-1}$$

and $\bar{A}_r^{[k]}$ denote the matrix obtained from $A^{[k]}$ by removing all the edges involving node r.

[Alsayed, Higham, (2015)]

Temporal betweenness

The temporal betweenness of time point q is defined as

$$\mathrm{TB}^{[M,q]} := \frac{1}{(n-1)^2-(n-1)} \sum_{i \neq j} \frac{(\mathcal{Q}^{[M]})_{ij} - (\widehat{\mathcal{Q}}^{[M,q]})_{ij}}{(\mathcal{Q}^{[M]})_{ij}},$$

where

$$\widehat{\mathcal{Q}}^{[M,q]} = \prod_{s=1}^{M} \left(I - a\widehat{A}^{[s,q]} \right)^{-1} \qquad \text{and} \qquad \widehat{A}^{[k,q]} = (1-\delta_{kq})A^{[k]}.$$

[Alsayed, Higham, (2015)]

To update or not to update?

Can we update the running receive and broadcast communicabilities at time step t_j using the same information at time step t_{j-1}?

To update or not to update?

Can we update the running receive and broadcast communicabilities at time step t_j using the same information at time step t_{j-1}?

$$\mathbf{1}^T \mathcal{S}^{[j]} = \mathbf{1}^T \left[\left(I + e^{-b\Delta t_j} \mathcal{S}^{[j-1]} \right) \left(I - aA^{[j]} \right)^{-1} - I \right]$$

$$\mathcal{S}^{[j]} \mathbf{1} = \left[\left(I + e^{-b\Delta t_j} \mathcal{S}^{[j-1]} \right) \left(I - aA^{[j]} \right)^{-1} - I \right] \mathbf{1}$$

To update or not to update?

Can we update the running receive and broadcast communicabilities at time step t_j using the same information at time step t_{j-1}?

$$\mathbf{1}^T \mathcal{S}^{[j]} = \left(\mathbf{1}^T + e^{-b\Delta t_j} \mathbf{1}^T \mathcal{S}^{[j-1]}\right) \left(I - aA^{[j]}\right)^{-1} - \mathbf{1}^T$$

$$\mathcal{S}^{[j]} \mathbf{1} = \left(I + e^{-b\Delta t_j} \mathcal{S}^{[j-1]}\right) \left(I - aA^{[j]}\right)^{-1} \mathbf{1}^T - \mathbf{1}^T$$

To update or not to update?

Can we update the running receive and broadcast communicabilities at time step t_j using the same information at time step t_{j-1}?

$$\mathcal{S}^{[j]}\mathbf{1} = \left(\mathbf{1}^T + e^{-b\Delta t_j}\mathbf{1}^T\mathcal{S}^{[j-1]}\right)\left(I - aA^{[j]}\right)^{-1} - \mathbf{1}^T$$

$$\mathcal{S}^{[j]}\mathbf{1} = \left(I + e^{-b\Delta t_j}\mathcal{S}^{[j-1]}\right)\left(I - aA^{[j]}\right)^{-1}\mathbf{1}^T - \mathbf{1}^T$$

First result

Lemma

For the running dynamic communicability matrix $\mathcal{S}^{[j]}$ we have

$$\mathcal{S}^{[j]} = \sum_{i=1}^{j} \left(1 - e^{-b\Delta t_i}\right) e^{-b\sum_{\ell=i+1}^{j} \Delta t_\ell} \mathcal{Q}^{[i,j]} - I,$$

where $\mathcal{Q}^{[i,j]} = \prod_{s=i}^{j} \left(I - aA^{[s]}\right)^{-1}$ and $\Delta t_1 = \infty$.

Block representation

We want to define a block-matrix representation of the data $\{A^{[k]}\}_{k=1}^{M}$ that transform the network sequence into an "equivalent" large, static network with adjacency matrix of dimension Mn.

We have two main requirements for such a representation.

- We would like to be able to interpret this static network in terms of the interactions represented by the original data.
- We would like to be able to recover the dynamic centrality measures already discussed by applying standard matrix functions to this larger network.

Block representation

We can define a matrix $B \in \mathbb{R}^{Mn \times Mn}$ as

$$B := \begin{bmatrix} \alpha A^{[1]} & \beta_2 I & & & \\ & \alpha A^{[2]} & \beta_3 I & & \\ & & \ddots & \ddots & \\ & & & \alpha A^{[M-1]} & \beta_M I \\ & & & & \alpha A^{[M]} \end{bmatrix}.$$

Where $\{\beta_\ell\}_{\ell=2,M}$ and α are parameters.

Block representation

Theorem

The dynamic communicability matrices $\mathcal{Q}^{[i,j]}$ and the running dynamic communicability matrices $\mathcal{S}^{[j]}$ can be computed by applying the function $f(x) = (1-x)^{-1}$ to the matrix B.

In particular:

- the dynamic communicability matrices $\mathcal{Q}^{[i,j]}$ can be obtained from the block $[f(B)]_{ij}$ setting $\beta_\ell = 1$, $\ell = 2, \ldots, M$ and $\alpha = a$;

- the running dynamic communicability matrices $\mathcal{S}^{[j]}$ are obtained starting from the blocks on the jth block-column $[f(B)]_{.j}$ setting $\beta_\ell = e^{-b\Delta t_\ell}$, $\alpha = a$.

Block representation

Theorem

Let $\bar{A}_r^{[k]}$ denote the matrix obtained from $A^{[k]}$ by removing all the edges involving node r and let $\{\widehat{A}^{[k,q]}\}_{k=1}^{M}$ be the adjacency matrix sequence obtained replacing $A^{[q]}$ with 0. Then, for $f(x) = (1-x)^{-1}$,

$$\mathrm{NB}_r = \frac{1}{(n-1)^2 - (n-1)} \sum_{i \neq j \neq r} \sum \frac{[f(B)]_{1M}^{ij} - \left[f(\bar{B}_r)\right]_{1M}^{ij}}{[f(B)]_{1M}^{ij}},$$

$$\mathrm{TB}^{[M,q]} = \frac{1}{(n-1)^2 - (n-1)} \sum_{i \neq j} \sum \frac{[(f(B)]_{1M}^{ij} - [(f(\widehat{B}^{[q]})]_{1,M-1}^{ij}}{[f(B)]_{1M}^{ij}}.$$

where $[(f(B)]_{\ell,k}^{ij}$ denotes the (i,j)th element of the (ℓ,k)th block of the matrix $f(B)$.

Block representation

Theorem

The matrices $\bar{B}_r$ and $\widehat{B}^{[q]}$ are given by

$$\bar{B}_r = \bigoplus_{i=1}^{M} (\alpha \bar{A}^{[i]}) + diag([I, \ldots, I], 1)$$

$$\widehat{B}^{[q]} = \bigoplus_{q \neq i=1}^{M} (\alpha A^{[i]}) + diag([I, \ldots, I], 1)$$

Block representation

We will focus on the computation of running broadcast and receive communicabilities since we need to store the whole matrix $\mathcal{S}^{[j-1]}$ in order to obtain the running dynamic communicability matrix $\mathcal{S}^{[j]}$.

Setting $\beta_\ell = e^{-b\Delta t_\ell}$ and $\alpha = a$, we obtain

$$\mathcal{S}^{[j]}\mathbf{1}_{\mathrm{n}} = (\mathbf{d} \otimes I_{\mathrm{n}})^T f(B)(\mathbf{e}_j \otimes \mathbf{1}_{\mathrm{M}}) - \mathbf{1}_{\mathrm{n}}$$
$$\mathcal{S}^{[j]^T}\mathbf{1}_{\mathrm{n}} = (\mathbf{e}_j \otimes I_{\mathrm{n}})^T f(B)^T(\mathbf{d} \otimes \mathbf{1}_{\mathrm{n}}) - \mathbf{1}_{\mathrm{n}},$$

where $\mathbf{d} = [1, 1-\beta_2, \ldots, 1-\beta_M]^T$, $\mathbf{1}_{\mathrm{M}}$ and $\mathbf{1}_{\mathrm{n}}$ are vectors of all ones in $\mathbb{R}^M$ and $\mathbb{R}^n$, respectively.

Bilinear form

We are interested in computation of quantities of the form

$$\mathbf{u}^T f(B)\mathbf{v}, \qquad \mathbf{u}, \mathbf{v} \in \mathbb{R}^{Mn},$$

with $\mathbf{u}, \mathbf{v}$ unit vectors and $f(B) = (I - B)^{-1}$ nonsymmetric.
In particular, $(\mathbf{u} = \mathbf{d} \otimes \mathbf{e}_i, \mathbf{v} = \mathbf{e}_j \otimes \mathbf{1}_{\mathrm{M}})$ and $(\mathbf{u} = \mathbf{e}_j \otimes \mathbf{e}_i, \mathbf{v} = \mathbf{d} \otimes \mathbf{1}_{\mathrm{n}})$, $i = 1, \ldots, n$, for the broadcast and receive running communicabilities of node i, respectively.

In particular, we use the nonsymmetric block Lanczos algorithm and pairs of block Gauss and anti-Gauss quadrature rules.

If $U = [\mathbf{u}\ \mathbf{1}]$ and $V = [\mathbf{v}\ \mathbf{1}]$, then we want to approximate the quantities

$$U^T f(B) V, \qquad U, V \in \mathbb{R}^{Mn \times 2}.$$

Resolution of a sparse linear system

We need to compute the quantities

$$\mathbf{u}^T (I - B)^{-1} \mathbf{v}, \qquad \mathbf{u}, \mathbf{v} \in \mathbb{R}^{Mn}.$$

This can be done by solving the sparse linear system $(I - B)\mathbf{x} = \mathbf{v}$ and then computing the scalar product $\mathbf{u}^T \mathbf{x}$.

The linear system can be solved either directly or iteratively. The peculiarity of the block formulation allows us to have at hand a regular matrix splitting. In fact, we have $I \geq 0$, $B \geq 0$ and $\rho(B) < 1$. Therefore, the iterative method

$$\mathbf{x}^{(k+1)} = B\mathbf{x}^{(k)} + \mathbf{v},$$

with given starting vector $\mathbf{x}^{(0)}$, converges to the solution $\mathbf{x}$.

Computational tasks

We analyze the following methods:

- **original** is the original approach proposed in [Grindrod, Higham (2012)].
- **quadrules** is the approach based on the Gauss and anti-Gauss quadrature rules.
- **linsolv** is the method that solves the big linear system $(I - B)\mathbf{x} = \mathbf{v}$ using the MATLAB "backslash".
- **iterative** is the iterative approach based on the regular matrix splitting $I - B$.
- **lsqr** is the method based on the solution of the linear system obtained by using the lsqr MATLAB function.

Computational tasks

We want to test the performance of the methods when the size of the matrix B increases.

Computational tasks

We want to test the performance of the methods when the size of the matrix B increases.

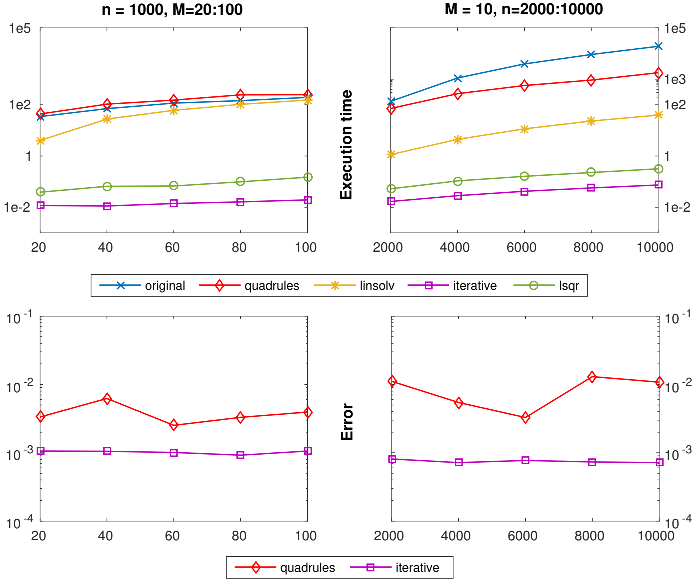
- As a first approach, we independently sample M times from the same static network model with a fixed number of nodes n. This can be done in MATLAB by using the package CONTEST by Taylor and Higham.

Computational tasks

We want to test the performance of the methods when the size of the matrix B increases.

- As a first approach, we independently sample M times from the same static network model with a fixed number of nodes n. This can be done in MATLAB by using the package CONTEST by Taylor and Higham.
- As a second approach, we generate the M matrices by using the evolving network model proposed and analyzed in [Grindrod, Higham, Parsons (2012)]. Here, the network sequence corresponds to the sample path of a discrete time Markov chain.

Computational tests (Pref network model)



Computational tests

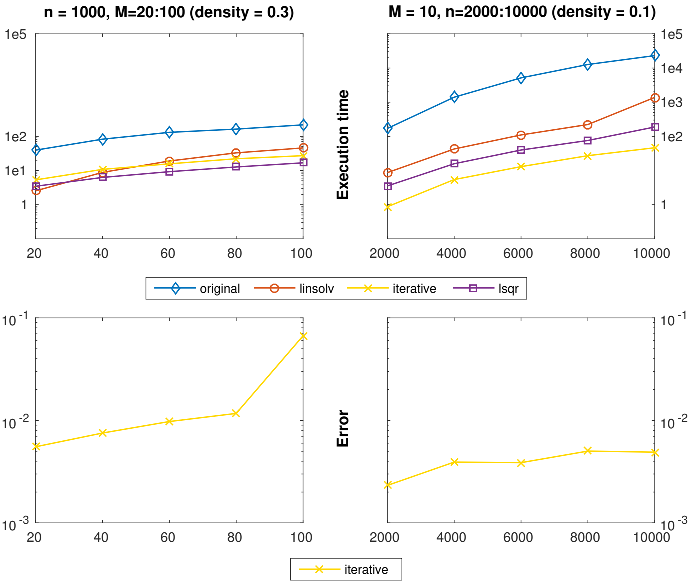
As a second set of numerical experiments we make use of the triadic closure model developed in [Grindrod, Higham, Parsons (2012)].

Starting from an Erdös-Rényi network model with a given edge density, we generate a sequence of M matrices in which the network at time point $k+1$ is built starting from the network at the previous time point. In particular, the expected value of $A^{[k+1]}$ given $A^{[k]}$ is

$$\mathcal{F}(A^{[k]}) = (1-\tilde{\omega})A^{[k]} + (\mathbf{1}^T\mathbf{1} - A^{[k]}) \circ (\delta \mathbf{1}^T\mathbf{1} + \epsilon (A^{[k]})^2),$$

where $\tilde{\omega} \in (0,1)$ is the death rate, $\delta\mathbf{1} + \epsilon(A^{[k]})^2$ is the birth rate, with $0 < \delta \ll 1$ and $0 < \epsilon(n-2) < 1-\delta$, and $\mathbf{1}$ is the vector of all ones.

Computational tests ($\tilde{\omega} = \delta = 20/n^2$ and $\epsilon = 5/n^2$)



Conclusions

We introduced a block-matrix representation that allows us to recover previously defined dynamic centralities and to compute/approximate them faster than the original methods.

Conclusions

We introduced a block-matrix representation that allows us to recover previously defined dynamic centralities and to compute/approximate them faster than the original methods.

Thank you