Evaluating Matrix Functions by Resummations on Graphs

P.-L. Giscard, S. Thwaite, D. Jaksch

Numerical Analysis Seminar
13th November 2012
Goal: compute $f(M)$

1. From Matrices to Graphs

2. Factoring Walks on Graphs

3. Back to Matrices
   - Live examples

4. Summary & Outlook
Goal: compute $f(M)$

1. From Matrices to Graphs

2. Factoring Walks on Graphs

3. Back to Matrices
   - Live examples

4. Summary & Outlook
From Matrices to Graphs

- Adjacency matrix $A^n =$ number of walks on graph

*What about arbitrary matrices?*
From Matrices to Graphs

- Adjacency matrix $A^n = \text{number of walks on graph}$

What about arbitrary matrices?

Matrix partition
Division of a matrix into blocks & mapping on graph
From Matrices to Graphs

Product of blocks $\iff$ walk

$$\begin{bmatrix} \text{Product of blocks} \end{bmatrix} \iff \begin{bmatrix} \text{walk} \end{bmatrix}$$

Edge weights

$$(\mathcal{M}^4) \iff \mathcal{M} \cdot \mathcal{M} \cdot \mathcal{M} \cdot \mathcal{M} + \ldots$$

Walk weight

Arbitrary matrix $\mathcal{M}^n = \text{sum of walk weights}$

P.-L. Giscard

Numerical Analysis Seminar, 13th November 2013
From Matrices to Graphs

Taylor series are recast into walk-sums

\[
\left( \sum_n M^n \right) = \sum_{w} c(w) = + +
\]

Accelerate convergence via graph theory
Goal: compute $f(M)$

1. From Matrices to Graphs
2. Factoring Walks on Graphs
3. Back to Matrices
   - Live examples
4. Summary & Outlook
Factoring Walks on Graphs

Observation: walk = simple path & simple cycle

Walks factor into simple paths, simple cycles

Prime factorization on graphs arXiv:1202.5523
Factoring Walks on Graphs

Observation: walk = simple path & simple cycle

Walks factor into simple paths, simple cycles

$\text{Prime factorization on graphs } \text{arXiv:1202.5523}$
Factoring Walks on Graphs

Why?

- Factorized forms ...

\[ p \circ c_1 \circ (c_2)^0 + p \circ c_1 \circ (c_2)^1 + p \circ c_1 \circ (c_2)^2 + p \circ c_1 \circ (c_2)^3 + \cdots \]

... are easy to sum !

\[ p \circ c_1 \circ (I - c_2)^{-1} \]

- We can sum all walks this way

Prime factorization on graphs arXiv:1202.5523
Factoring Walks on Graphs

*How does it work?*

- Factor sums of walks

\[
\sum_{w: \alpha \rightarrow \beta} w = \sum_{p: \alpha \rightarrow \beta} (\alpha)(\alpha \nu_1)(\nu_1)(\nu_1 \nu_2) \cdots (\nu_p \beta)(\beta)
\]

Ensemble of simple paths
Factoring Walks on Graphs

*How does it works?*

- Factor sums of walks

\[
\sum_{w: \alpha \rightarrow \beta} w = \sum_{p: \alpha \rightarrow \beta} (\alpha)(\alpha \nu_1)(\nu_1)(\nu_1 \nu_2) \cdots (\nu_p \beta)(\beta)
\]

Insert simple cycles: \(G, G\{\alpha\}, G\{\alpha, \nu_1, \cdots \nu_p\}\)
Factoring Walks on Graphs

How does it works?

- Factor sums of walks

\[
\sum_{w: \alpha \rightarrow \beta} w = \sum_{p: \alpha \rightarrow \beta} (\alpha)(\alpha \nu_1)(\nu_1)(\nu_1 \nu_2) \cdots (\nu_p \beta)(\beta)
\]

Insert simple cycles:

\[
(\alpha) \rightarrow [I - \sum_{c: \alpha \rightarrow \alpha} (\alpha \mu_1)(\mu_1) \cdots (\mu_\ell)(\mu_\ell \alpha)]^{-1}
\]
Factoring Walks on Graphs

How does it works?

- Factor sums of walks

\[ \sum_{w: \alpha \rightarrow \beta} w = \sum_{p: \alpha \rightarrow \beta} (\alpha)(\alpha \nu_1)(\nu_1)(\nu_1 \nu_2) \cdots (\nu_p \beta)(\beta) \]

Insert simple cycles:

\[ (\alpha) \rightarrow [I - \sum_{c: \alpha \rightarrow \alpha} (\alpha \mu_1)(\mu_1) \cdots (\mu_{\ell})(\mu_{\ell} \alpha)]^{-1} \]

Insert simple cycles again...

\[ \mathcal{G}\backslash\{\alpha\} \quad \mathcal{G}\backslash\{\alpha, \nu_1, \cdots \nu_p\} \]

\[ \mathcal{G}\backslash\{\alpha\} \quad \mathcal{G}\backslash\{\alpha, \mu_1, \cdots \mu_{\ell-1}\} \]
Factoring Walks on Graphs

What does it bring?

• Sums of walks
  \[ \sum_{p: \alpha \rightarrow \beta} [\alpha]_G (\alpha \nu_1) [\nu_1]_{G \{ \alpha \}} (\nu_1 \nu_2) \cdots (\nu_p \beta) [\beta]_{G \{ \alpha, \nu_1, \cdots, \nu_p \}} \]
  \[ \Rightarrow \text{Sum of simple paths} \]

  \[ \Rightarrow \text{Continued fraction of simple cycles} \]

  \[ [\alpha]_G = \left[ \mathcal{I} - \sum_{c: \alpha \rightarrow \alpha} (\alpha \mu_1) [\mu_1]_{G \{ \alpha \}} \cdots [\mu_\ell]_{G \{ \alpha, \mu_1, \cdots, \mu_{\ell-1} \}} (\mu_\ell \alpha) \right]^{-1} \]

• 'Few' simple paths & cycles + vertex removal
  \[ \Rightarrow \text{Finite number of operations} \]

And for matrices?
Goal: compute $f(M)$

1. From Matrices to Graphs

2. Factoring Walks on Graphs

3. Back to Matrices
   - Live examples

4. Summary & Outlook
Back to Matrices

Same, with weights

- Replace edges $\Rightarrow$ weights
  $\longrightarrow$ factorizes $\sum_n \mathcal{M}^n$

$\left( \sum_n \mathcal{M}^n \right)_{\beta\alpha} = \sum_{P_G} [\beta]_{g\{\alpha,\nu_2,\ldots,\nu_p\}} \mathcal{M}_{\beta\nu_p} \cdots [\nu_2]_{g\{\alpha\}} \mathcal{M}_{\nu_2\alpha} [\alpha]_{g}$

Indices of the partition

\[
\begin{pmatrix}
\alpha = \square \\
\beta = \square \\
\end{pmatrix}
\]
Same, with weights

- Replace edges $\Rightarrow$ weights
  $\quad \xrightarrow{}$ factorizes $\sum_n \mathcal{M}^n$

\[
(\sum_n \mathcal{M}^n)_{\beta \alpha} = \sum [\beta]_{g \setminus \{\alpha, \nu_2, \ldots, \nu_p\}} \mathcal{M}_{\beta \nu_p} \cdots [\nu_2]_{g \setminus \{\alpha\}} \mathcal{M}_{\nu_2 \alpha} [\alpha]_g
\]

Sum over the simple paths
Back to Matrices

*Same, with weights*

- Replace edges $\Rightarrow$ weights
  - $\sum_n \mathcal{M}^n$ factorizes

\[
\left( \sum_n \mathcal{M}^n \right)_{\beta\alpha} = \sum_{P_G} [\beta]_{g\setminus\{\alpha,\nu_2,\ldots,\nu_p\}} \mathcal{M}_{\beta\nu_p} \cdots [\nu_2]_{g\setminus\{\alpha\}} \mathcal{M}_{\nu_2\alpha} \ [\alpha]_G
\]

Edge weights
Back to Matrices

Same, with weights

- Replace edges ⇒ weights
  → factorizes $\sum_n M^n$

\[
(\sum_n M^n)_{\beta\alpha} = \sum_{P_G} [\beta]_{G\backslash\{\alpha,\nu_2,\ldots,\nu_p\}} M_{\beta\nu_p} \cdots [\nu_2]_{G\backslash\{\alpha\}} M_{\nu_2\alpha} [\alpha]_G
\]

Effective vertex weights

\[
[\alpha]_G = \left[ I - \sum_{C_{G;\alpha}} M_{\alpha\mu_\ell} [\mu_\ell]_{G\backslash\{\alpha,\mu_2,\ldots,\mu_{\ell-1}\}} \cdots [\mu_2]_{G\backslash\{\alpha\}} M_{\mu_2\alpha} \right]^{-1}
\]

Sum over simple cycles

\[
\text{P.-L. Giscard} \\
19 / 27 \\
\text{Numerical Analysis Seminar, 13th November 2013}
\]
Back to Matrices

Same, with weights

• Replace edges $\Rightarrow$ weights
  $\mapsto$ factorizes $\sum_n M^n$

\[
(\sum_n M^n)_{\beta\alpha} = \sum_{P_G} [\beta]_{G\backslash\{\alpha,\nu_2,\ldots,\nu_p\}} M_{\beta\nu_p} \cdots [\nu_2]_{G\backslash\{\alpha\}} M_{\nu_2\alpha} [\alpha]_G
\]

$[\alpha]_G =$

\[
\left[ I - \sum_{C_{G;\alpha}} M_{\alpha\mu_\ell} [\mu_\ell]_{G\backslash\{\alpha,\mu_2,\ldots,\mu_{\ell-1}\}} \cdots [\mu_2]_{G\backslash\{\alpha\}} M_{\mu_2\alpha} \right]^{-1}
\]

$\Rightarrow$ Builds a continued fraction
Back to Matrices

What did we obtain?

- A factorized form for $\sum_n M^n$
- Valid for any partition
- Valid for any $M$ by analytic continuation
Back to Matrices

What did we obtain?

- A factorized form for $\sum_n M^n$
- Valid for any partition
- Valid for any $M$ by analytic continuation

Did we accelerate the Taylor series?

- Only a finite number of terms left
- Continued fraction

Other $f(M)$?

- Compute resolvent matrix $\mathcal{R}_M$
Step by step procedure:

- Partition $\mathcal{M} \Rightarrow \mathcal{G}$
- Calculate $\mathcal{R}_\mathcal{M}$ with path-sums
- Compute the inverse transform $f(\mathcal{M})$
- Already done for $\mathcal{M}^q$, $\exp(\mathcal{M})$, $\mathcal{M}^{-1}$, $\log(\mathcal{M})$

The method of path-sums arXiv:1112.1588
Back to Matrices

Remarks:

- $\mathcal{M}^q$, $q = -1$, $\mathcal{M}$ singular $\Rightarrow$ Drazin inverse
- Matrices with non-commuting elements
- $K_2 \Rightarrow$ 4 blocks inversion formula

$$
\begin{pmatrix}
A & B \\
C & D
\end{pmatrix}^{-1} = \\
\begin{pmatrix}
(A - BD^{-1}C)^{-1} & -A^{-1}B (D - CA^{-1}B)^{-1} \\
-D^{-1}C (A - BD^{-1}C)^{-1} & (D - CA^{-1}B)^{-1}
\end{pmatrix}
$$

- Block tridiagonal $\Rightarrow$ continued fraction
Live Examples!
Goal: compute $f(M)$

1. From Matrices to Graphs

2. Factoring Walks on Graphs

3. Back to Matrices
   - Live examples

4. Summary & Outlook
Summary & Outlook

Main Results

• Matrix function $\Rightarrow$ sum of simple paths & simple cycles

  Prime factorization on graphs arXiv:1202.5523
  The method of path-sums arXiv:1112.1588

• Successfully implemented in quantum mechanics

  Physical approach arXiv:1204.5087
  Quantum dynamics arXiv:1108.1177

Open questions

• MATLAB implementation
• Works for functions of operators! (Chebfun style)
• Extension to tensors, TNT algorithm
Thank you!

Simón Thwaite       Dieter Jaksch

Work supported by:
Scatcherd European Scholarship
EPSRC
Clarendon Scholarship
Balliol College, Oxford