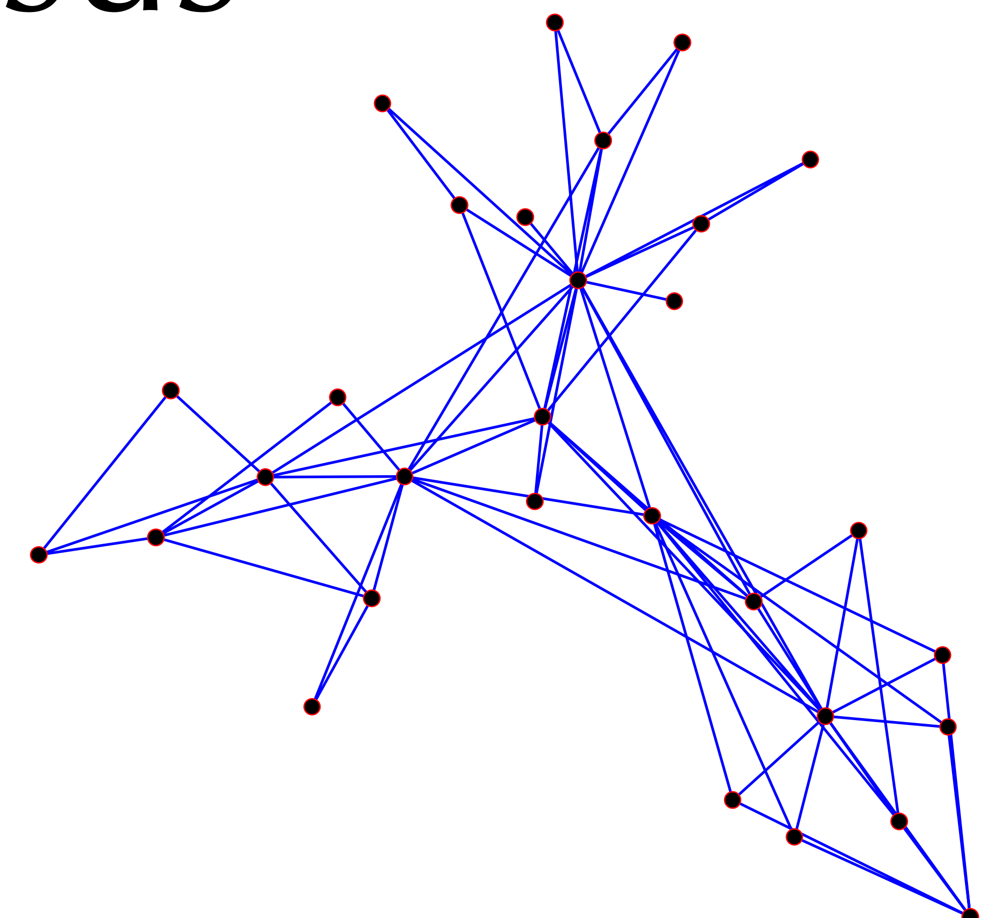
Technion Israel Institute of Technology

Cycles and Sparse Design of Consensus Networks



Daniel Zelazo

Faculty of Aerospace Engineering

Technion-Israel Institute of Technology

Simone Schuler, Frank Allgöwer

Institute for Systems Theory and Automatic Control

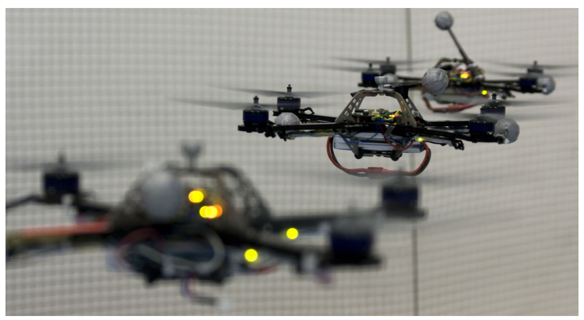
University of Stuttgart

Conference on Decision and Control 2012

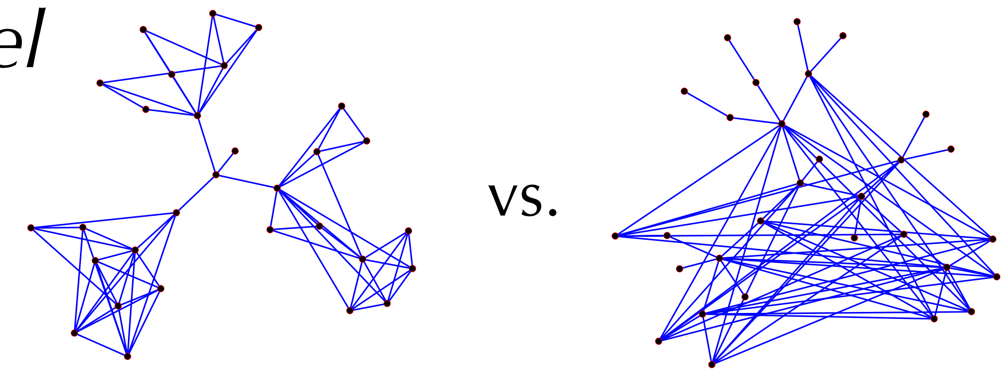
Maui, Hawaii

Consensus-Seeking Networks

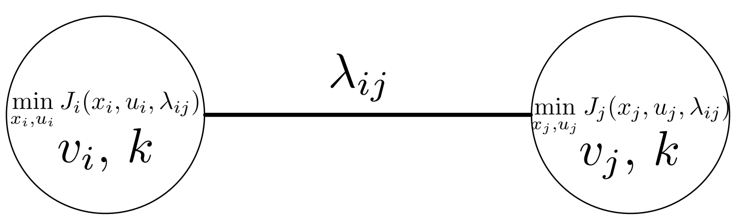
The consensus protocol is a *canonical model* for studying complex networked systems



formation control



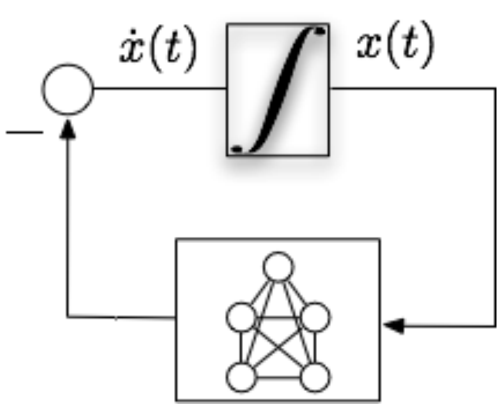
Are certain information structures more favorable to others?



distributed optimization

$\mathcal{H}_2$, $\mathcal{H}_\infty$, $\vdots$ $\propto$ cycle lengths, node degree, $\vdots$

Can notions of *dynamic system performance* be explained in terms of *properties of the graph?*



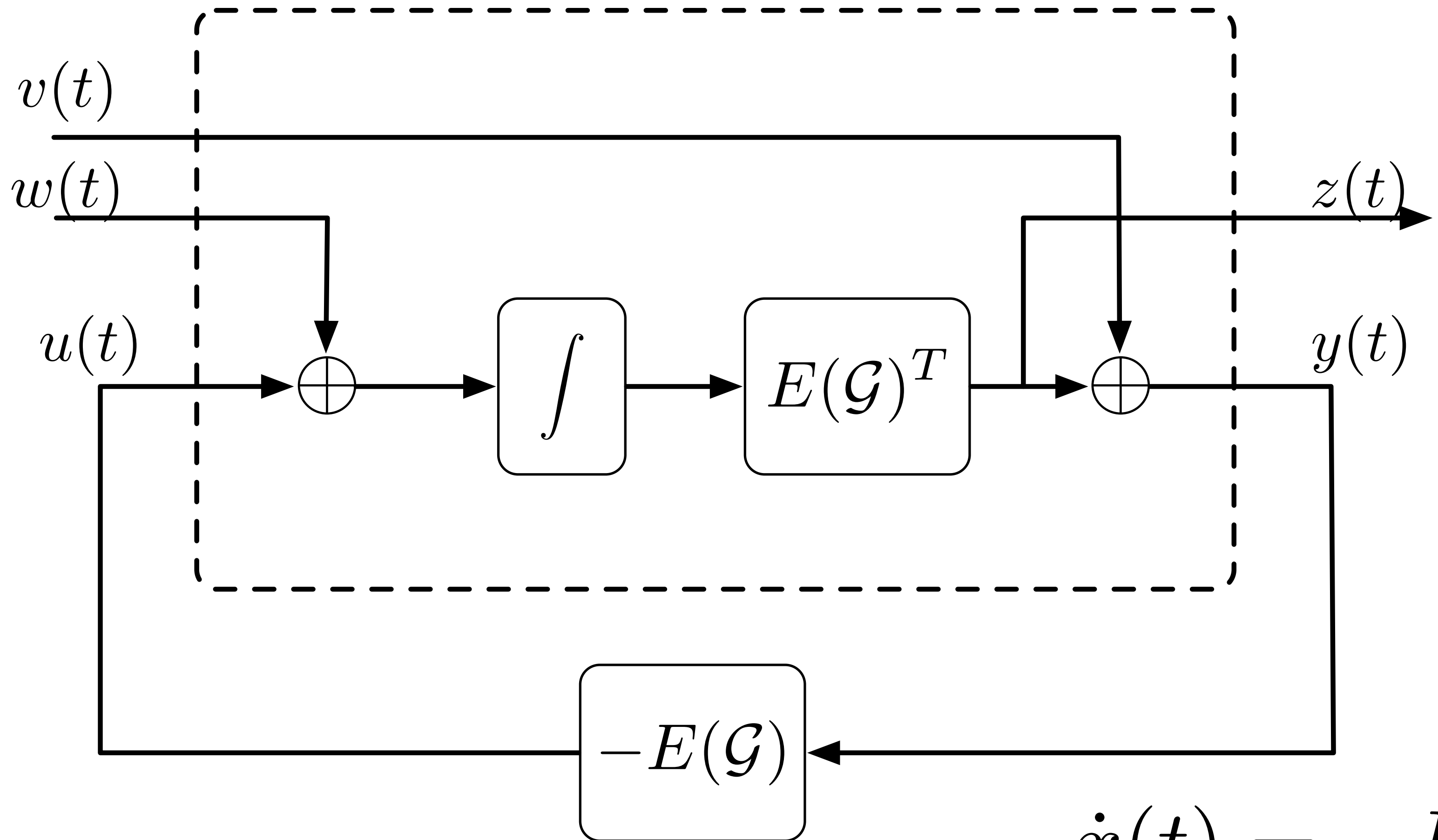
systems theory over graphs

$$\min_{\mathcal{G}} \|\Sigma(\mathcal{G})\|_p$$

How do we *synthesize* good information structures?

The Consensus Protocol

An 'input-output' consensus model

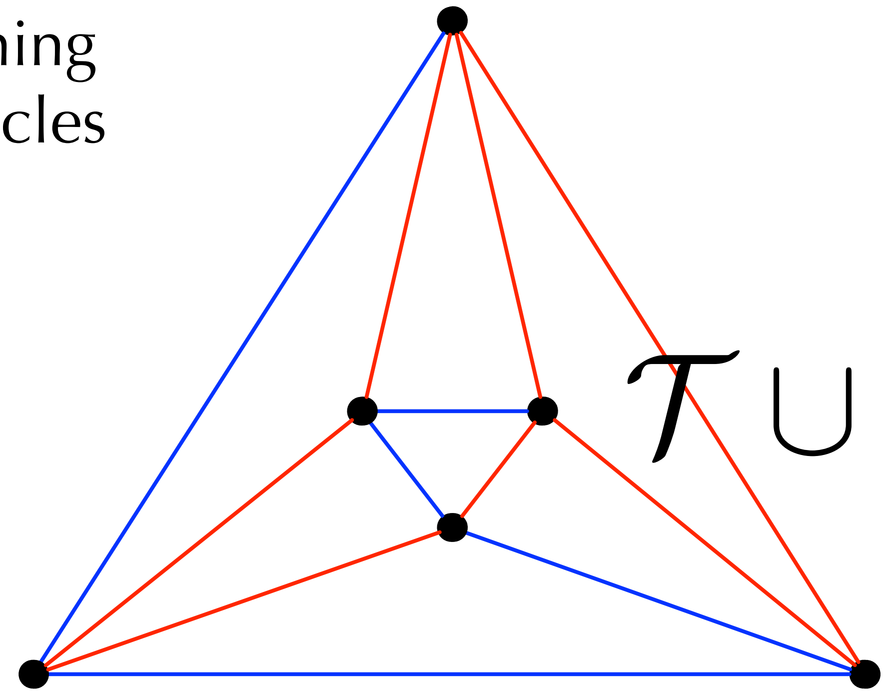


$$\dot{x}(t) = -L(\mathcal{G})x(t)$$

How do disturbances and noises affect the performance of the consensus protocol?

Spanning Trees and Cycles

A graph as the union of a spanning tree and edges that complete cycles



$\mathcal{T} \cup \mathcal{C}$

a spanning tree

remaining edges "complete cycles"

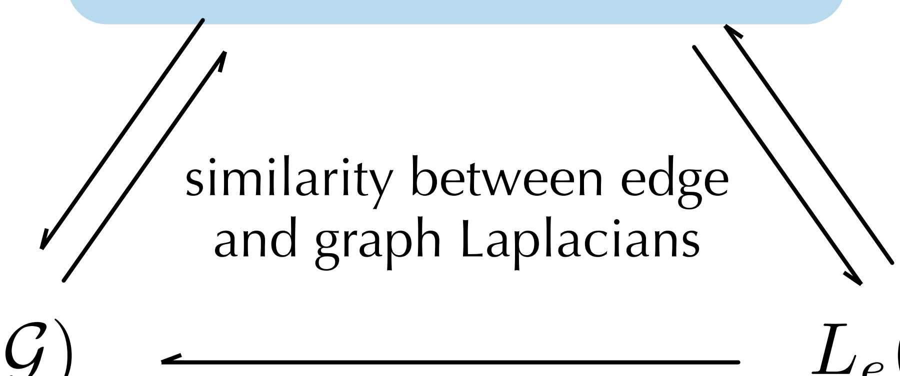
Edge Laplacian

$$L_e(\mathcal{G}) = E(\mathcal{G})^T E(\mathcal{G})$$

$\mathcal{R}_{(\mathcal{T},\mathcal{C})}$ rows form a basis for the *cut space* of the graph

Essential Edge Laplacian

$$L_e(\mathcal{T})\mathcal{R}_{(\mathcal{T},\mathcal{C})}\mathcal{R}^T_{(\mathcal{T},\mathcal{C})}$$

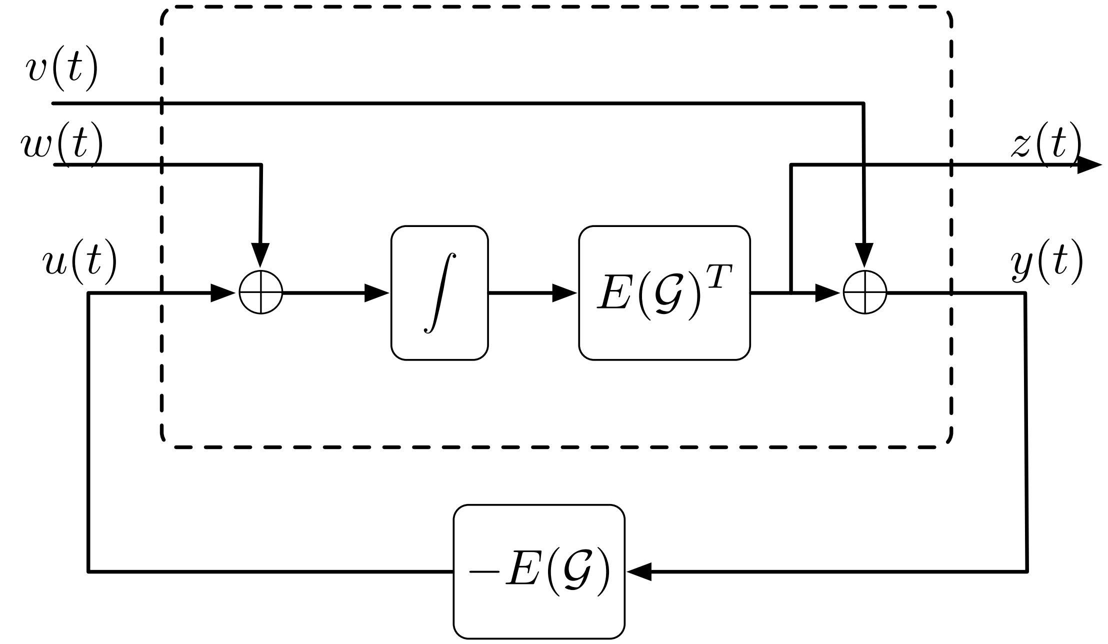


similarity between edge and graph Laplacians

$L(\mathcal{G})$ $L_e(\mathcal{G})$

The Edge Agreement Problem

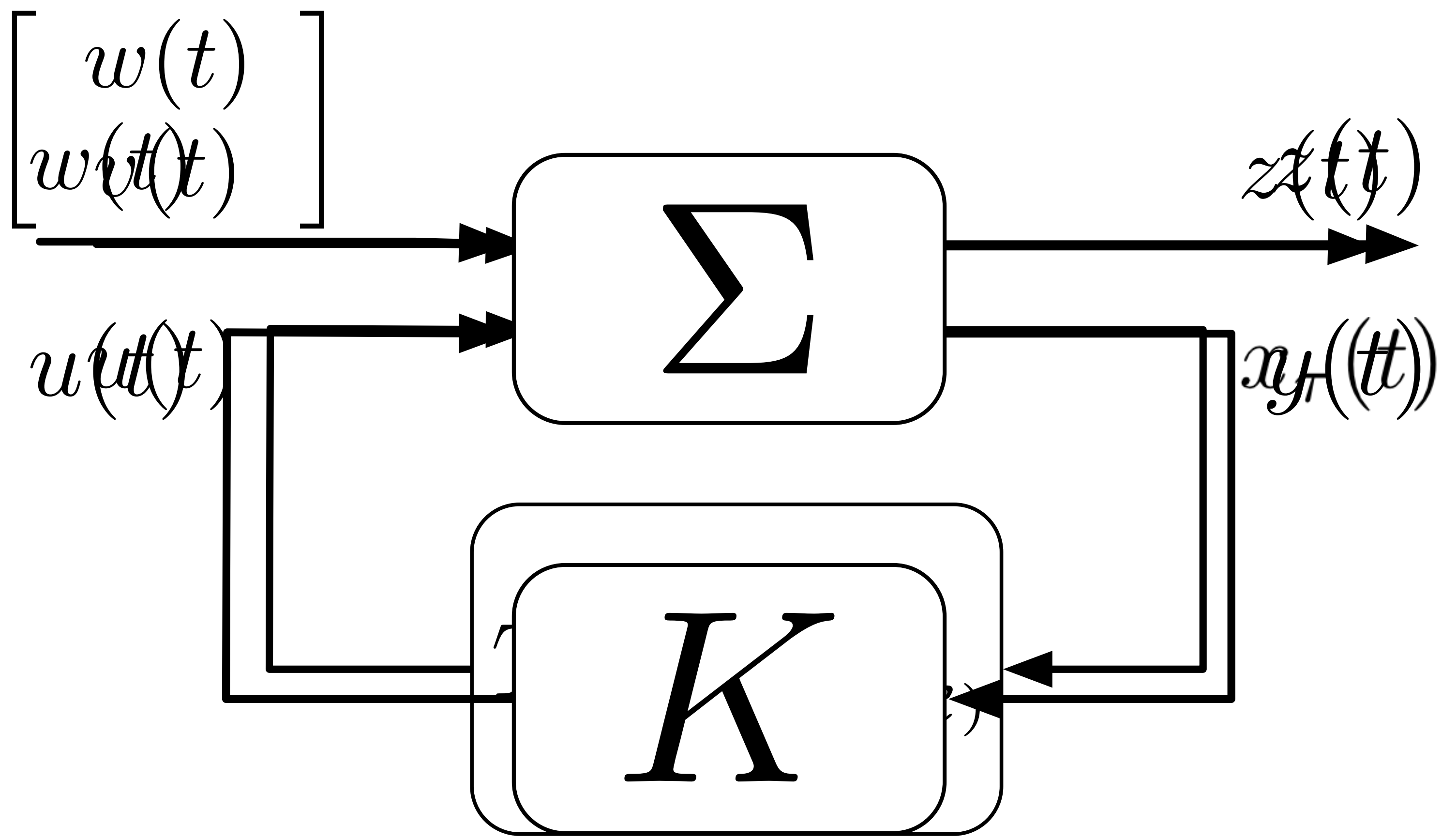
$$\Sigma(\mathcal{G}) : \begin{cases} \dot{x}(t) &= -L(\mathcal{G})x(t) + \begin{bmatrix} I & -E(\mathcal{G}) \end{bmatrix} \begin{bmatrix} w(t) \\ v(t) \end{bmatrix} \\ z(t) &= E(\mathcal{G})^T x(t). \end{cases}$$



$$\Sigma_e(\mathcal{G}) : \begin{cases} \dot{x}_{\mathcal{T}}(t) &= -L_e(\mathcal{T})R_{(\mathcal{T},\mathcal{C})}R^T_{(\mathcal{T},\mathcal{C})}x_{\mathcal{T}}(t) + \\ & \begin{bmatrix} E(\mathcal{T})^T & -L_e(\mathcal{T})R_{(\mathcal{T},\mathcal{C})} \end{bmatrix} \begin{bmatrix} w(t) \\ v(t) \end{bmatrix} \\ z(t) &= x_{\mathcal{T}}(t). \end{cases}$$

stable and minimal realization of consensus protocol

Cycles as Feedback



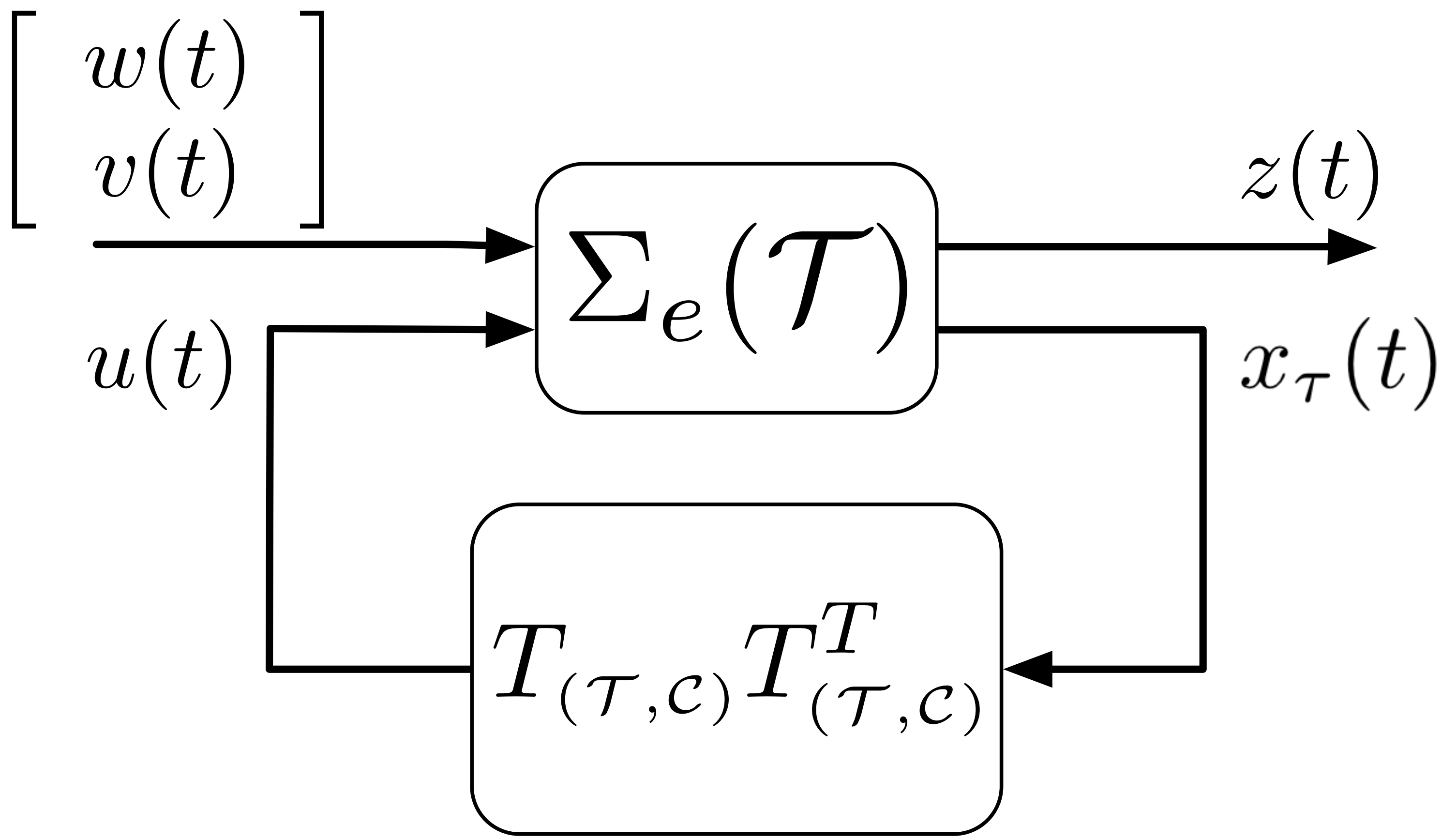
$$R_{(\mathcal{T},\mathcal{C})} = \left[\begin{array}{cc} I & T_{(\mathcal{T},\mathcal{C})} \end{array} \right]$$

$$E(\mathcal{T})T_{(\mathcal{T},\mathcal{C})} = E(\mathcal{C})$$

Design of consensus networks can be viewed as a *state-feedback problem*

$$L_e(\mathcal{T})R_{(\mathcal{T},\mathcal{C})}R^T_{(\mathcal{T},\mathcal{C})} = L_e(\mathcal{T}) + L_e(\mathcal{T})\underline{T_{(\mathcal{T},\mathcal{C})}T^T_{(\mathcal{T},\mathcal{C})}}$$

Cycles as Feedback



A synthesis problem

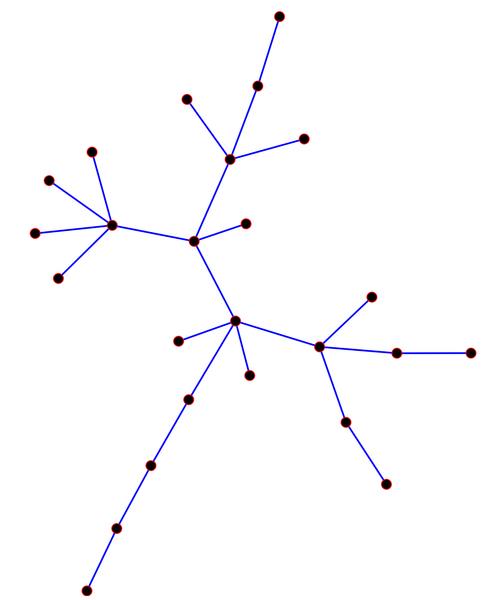
$$\min_{T_{(\mathcal{T},\mathcal{C})} \in \mathbb{R}^{|\mathcal{V}| \times k}} \|\Sigma_e(\mathcal{G})\|_2,$$

Performance of Consensus

Theorem

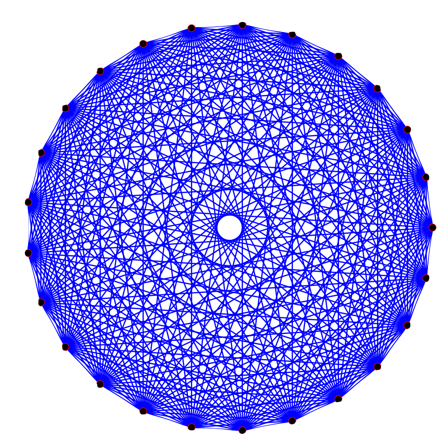
$$\|\Sigma_e(\mathcal{G})\|_2^2 = \frac{1}{2}\mathbf{tr}\left[(R_{(\mathcal{T},\mathcal{C})}R_{(\mathcal{T},\mathcal{C})}^T)^{-1}\right] + (n-1)$$

some immediate bounds...



$$\|\Sigma_e(\mathcal{G})\|_2^2 \leq \|\Sigma_e(\mathcal{T})\|_2^2 = \frac{3}{2}(n-1)$$

all trees are the same

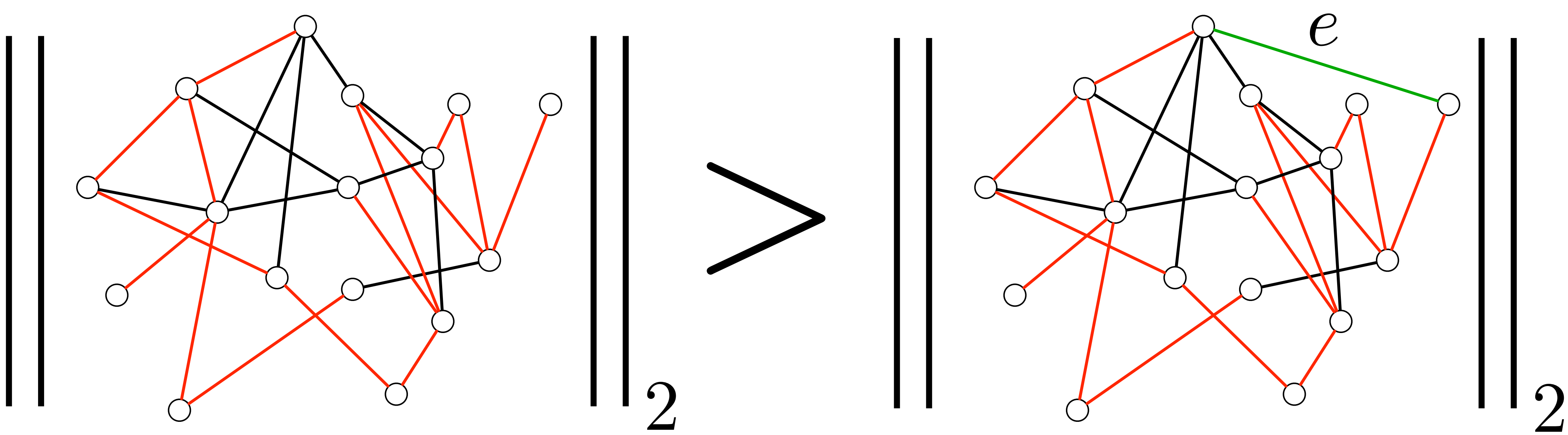


$$\|\Sigma_e(\mathcal{G})\|_2^2 \geq \|\Sigma_e(K_n)\|_2^2 = \frac{n^2-1}{n}$$

Performance and Cycles

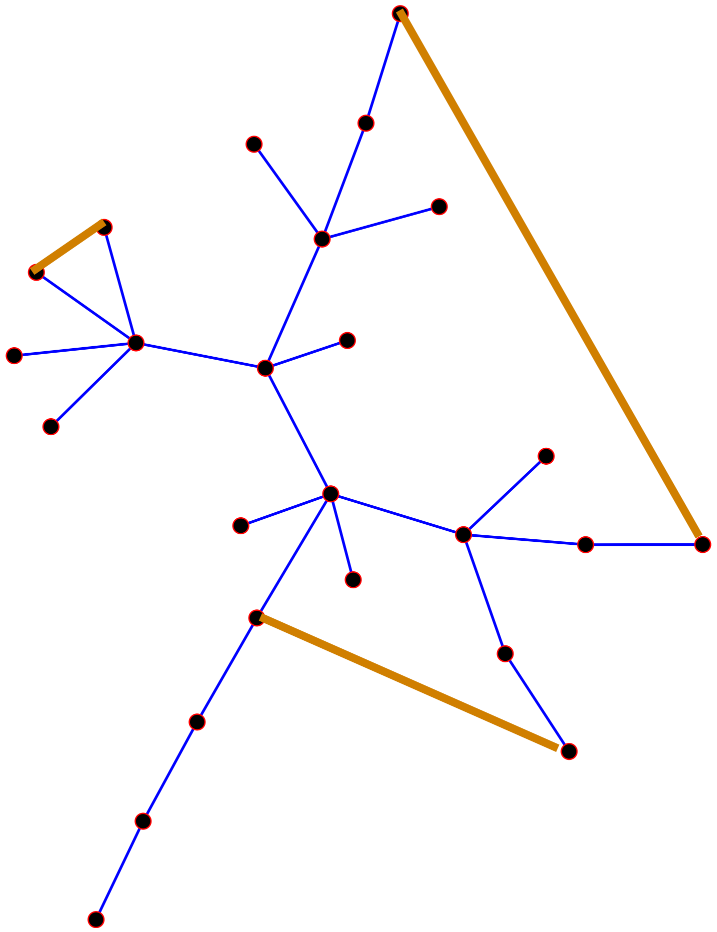
Theorem: Adding cycles always improves the performance.

$$\|\Sigma_e(\mathcal{G}\cup e)\|_2^2 = \|\Sigma_e(\mathcal{G})\|_2^2 - \frac{\left(R_{(\mathcal{T},\mathcal{C})}R_{(\mathcal{T},\mathcal{C})}^T\right)^{-1} cc^T \left(R_{(\mathcal{T},\mathcal{C})}R_{(\mathcal{T},\mathcal{C})}^T\right)^{-1}}{2(1+c^T\left(R_{(\mathcal{T},\mathcal{C})}R_{(\mathcal{T},\mathcal{C})}^T\right)^{-1}c)}$$



Performance and Cycles

Is there a *combinatorial* feature that affects the performance?



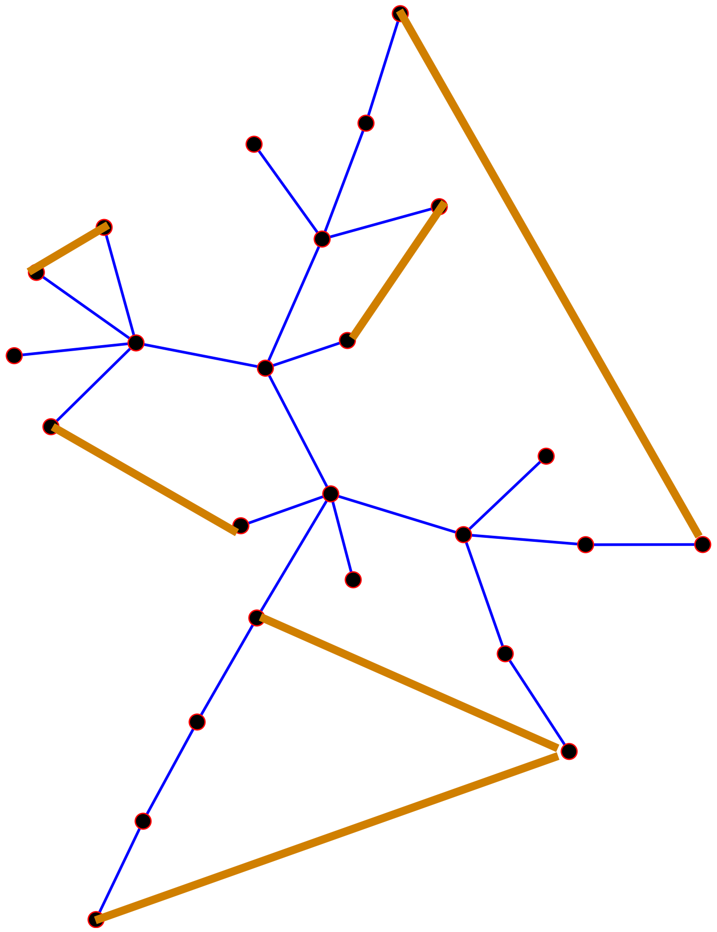
Corollary

$$\|\Sigma_e(\mathcal{T} \cup e)\|_2^2 = \|\Sigma_e(\mathcal{T})\|_2^2 - \frac{1}{2}(1 - l(c)^{-1})$$

long cycles are "better"

Performance and Cycles

Is there a *combinatorial* feature that affects the performance?

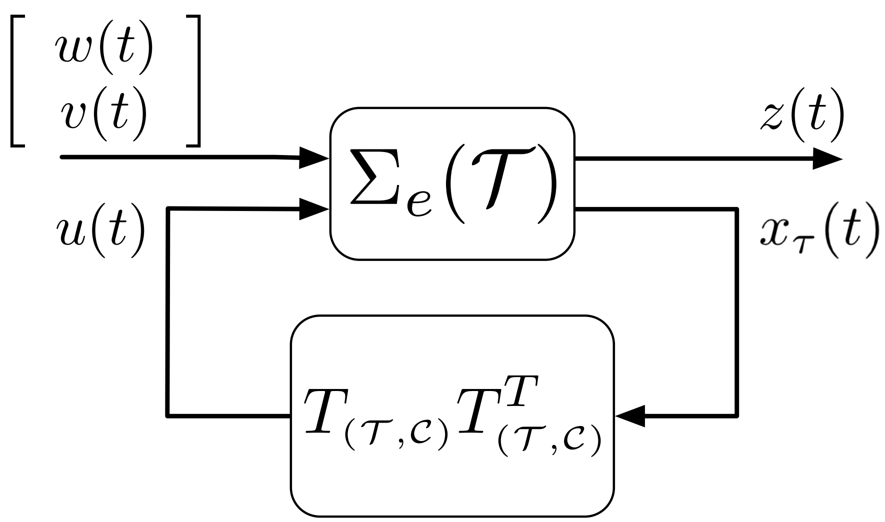


Corollary

$$\|\Sigma_e(\mathcal{T} \cup \{e_1, e_2\})\|_2^2 = \|\Sigma_e(\mathcal{T})\|_2^2 - \left(1 - \frac{l(c_1) + l(c_2)}{2(l(c_1)l(c_2) - s_{12}^2)}\right)$$

"edge disjoint" cycles are better

Design of Cycles



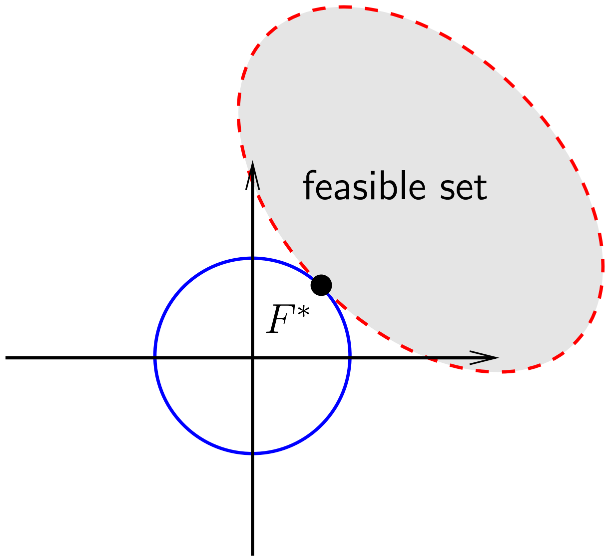
$$\min_{T_{(\mathcal{T},\mathcal{C})}\in\mathbb{R}^{|\mathcal{V}|\times k}} \|\Sigma_e(\mathcal{G})\|_2,$$

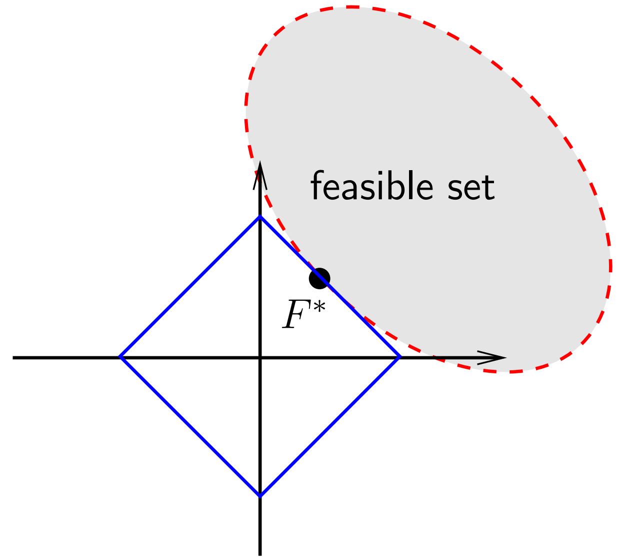
Given a spanning tree, add **k** edges that maximize the performance improvement

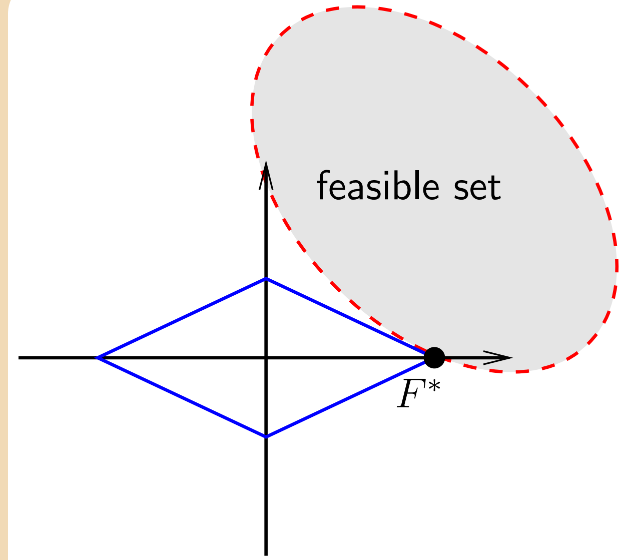
a mixed-integer SDP

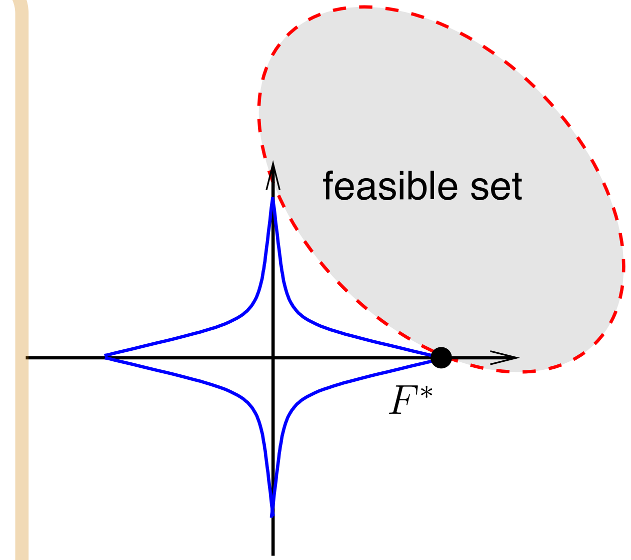
$$\begin{aligned} \min_{M, w_i} \quad & \mathbf{trace}\,[M] \\ \text{s.t.} \quad & \begin{bmatrix} M & I \\ I & I + T_{(\mathcal{T},\overline{\mathcal{T}})} W T_{(\mathcal{T},\overline{\mathcal{T}})} \end{bmatrix} \geq 0 \\ & \sum_i w_i = k,\ w_i \in \{0,1\} \end{aligned}$$

Sparsity Promoting Optimization









$$\min_{x \in \mathcal{X}} \ \|x\|_2$$

∗convex optimization
∗not sparse

$$\min_{x \in \mathcal{X}} \ \|x\|_1$$

∗convex optimization
∗sparse for LP

$$\min_{x \in \mathcal{X}} \sum_i m_i |x_i|$$

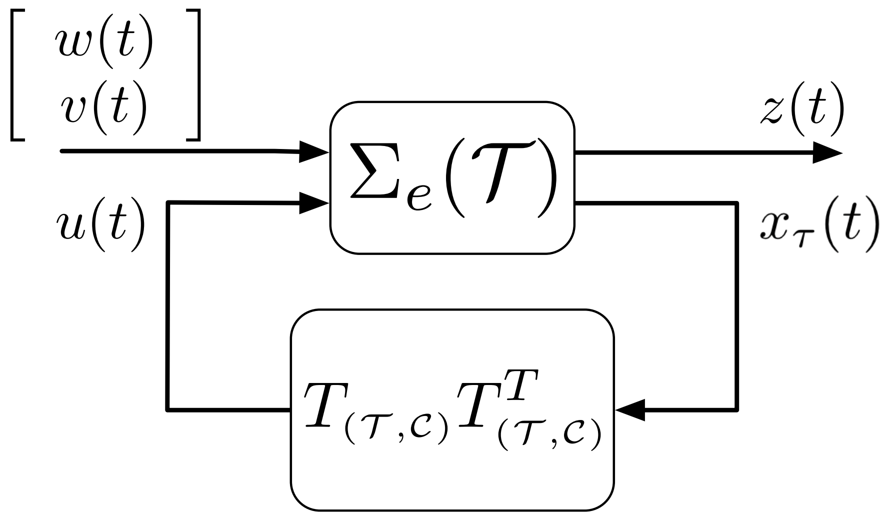
∗convex optimization
∗sparse for SDP

$$\min_{x \in \mathcal{X}} \ \|x\|_p$$

∗non-convex
∗sparse

re-weighted *l*-1 minimization algorithm
[Candes 2008]

Design of Cycles



$$\min_{T_{(\mathcal{T},\mathcal{C})}\in\mathbb{R}^{|\mathcal{V}|\times k}} \|\Sigma_e(\mathcal{G})\|_2,$$

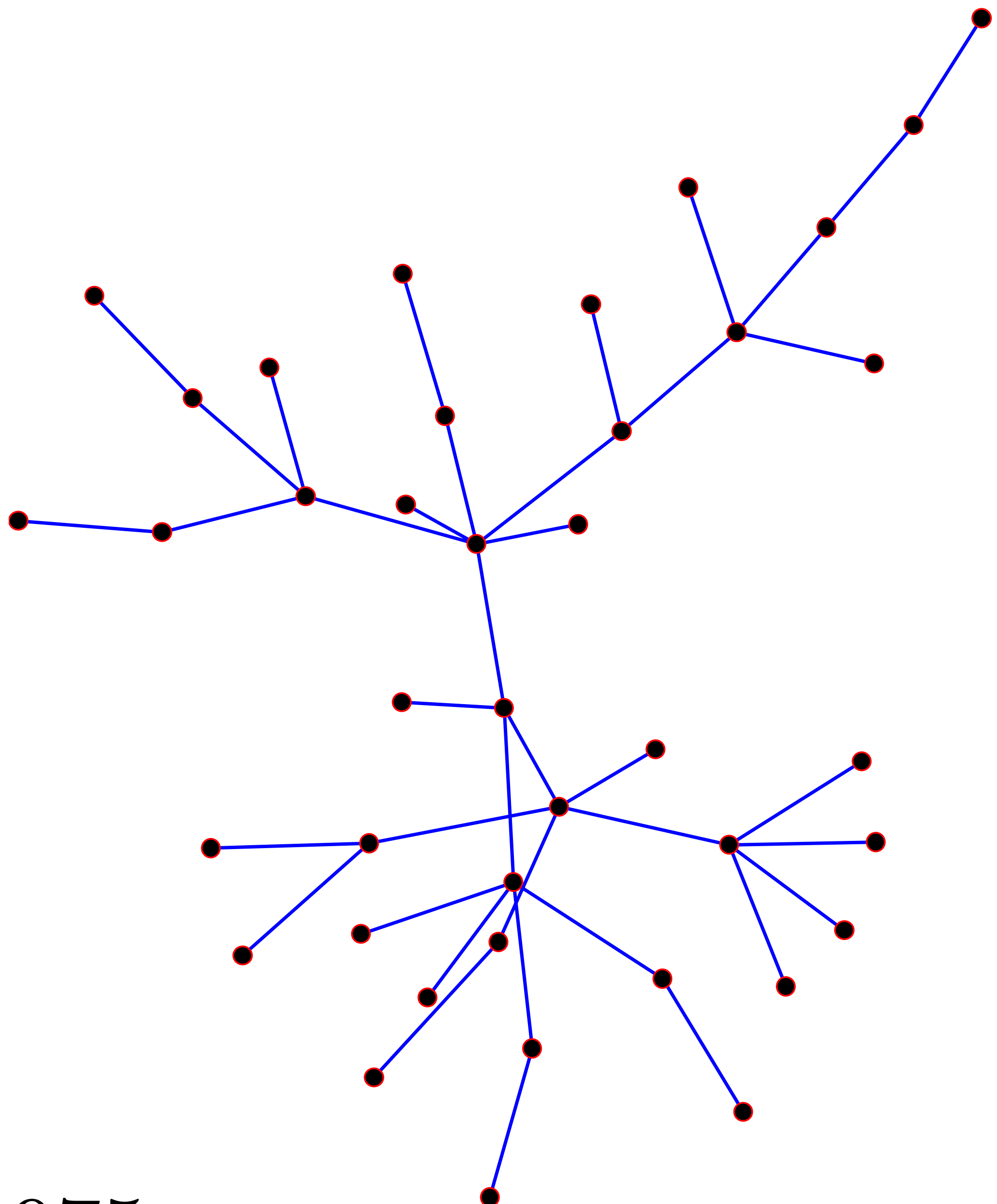
Given a spanning tree, add **k** edges that maximize the performance improvement

$$\min_{M,w_i} \quad \alpha\,\mathbf{trace}\,[M] + (1-\alpha)\sum_i m_i w_i$$

$$\text{s.t.} \quad \begin{bmatrix} M & I \\ I & I + T_{(\mathcal{T},\overline{\mathcal{T}})} W T_{(\mathcal{T},\overline{\mathcal{T}})} \end{bmatrix} \geq 0$$

$$\sum_i w_i = k, \quad 0 \leq w_i \leq 1.$$

Simulation Examples



spanning tree
30 nodes

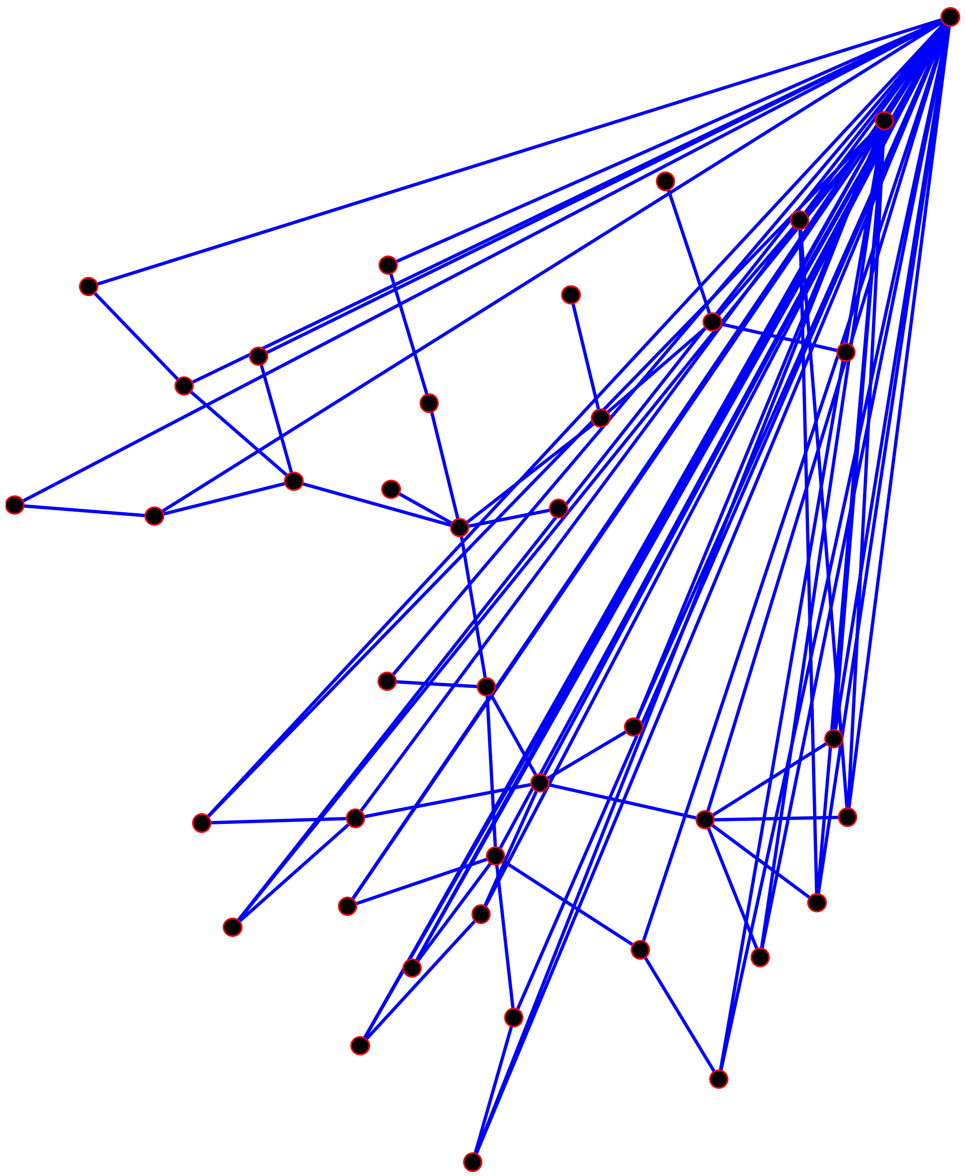
741 candidate edges

add 40 new edges

$$\|\Sigma(\mathcal{T})\|_2^2 = 58.5$$

$$\|\Sigma(K_n)\|_2^2 = 39.975$$

Simulation Examples



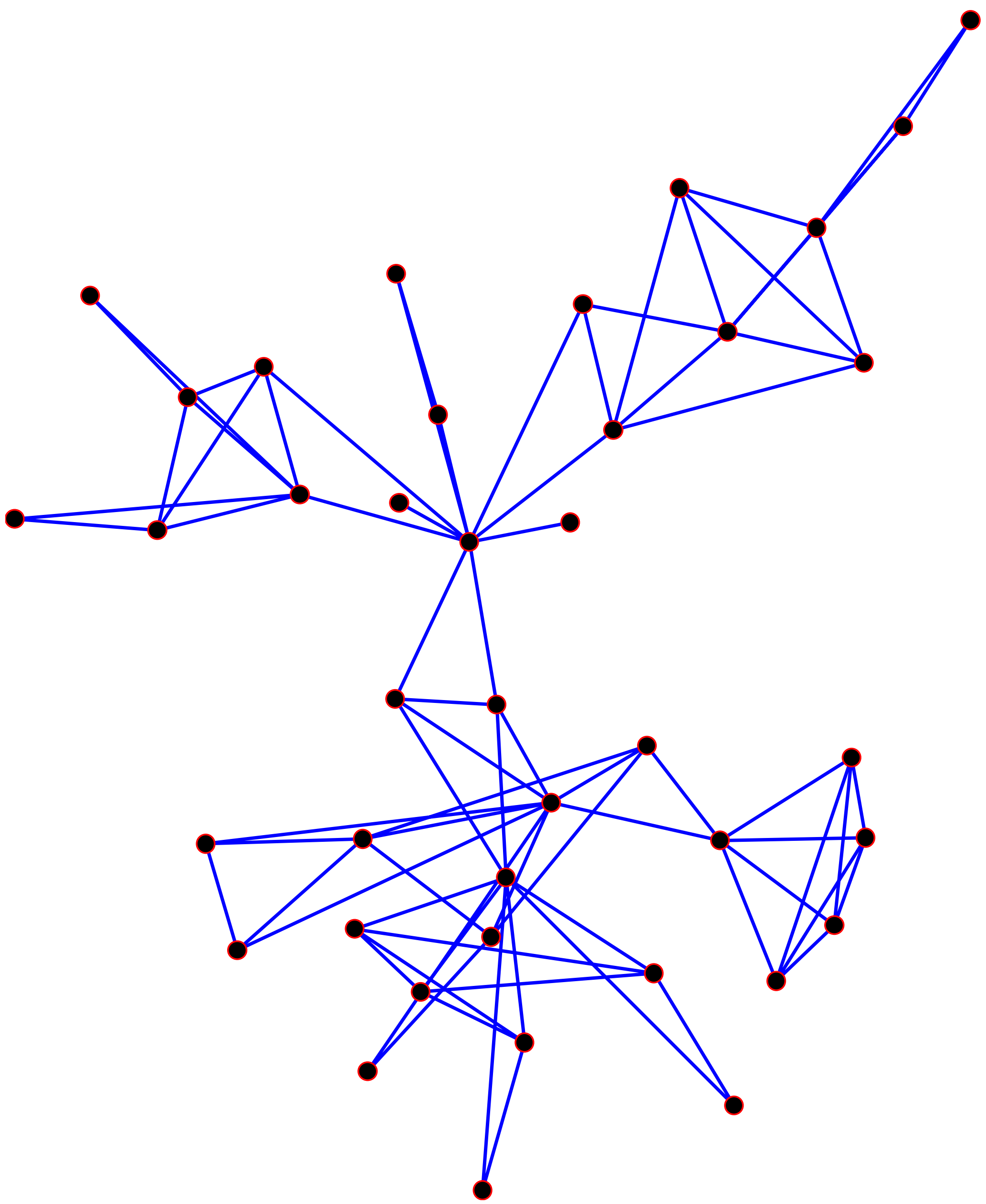
weights can be used to promote certain graph properties

"long cycle weights"

$$m_i = \mathbf{diam}(\mathcal{G}) - \|c_i\|_1 + 1$$

$$\|\Sigma(\mathcal{G})\|_2^2 = 50.233$$

Simulation Examples



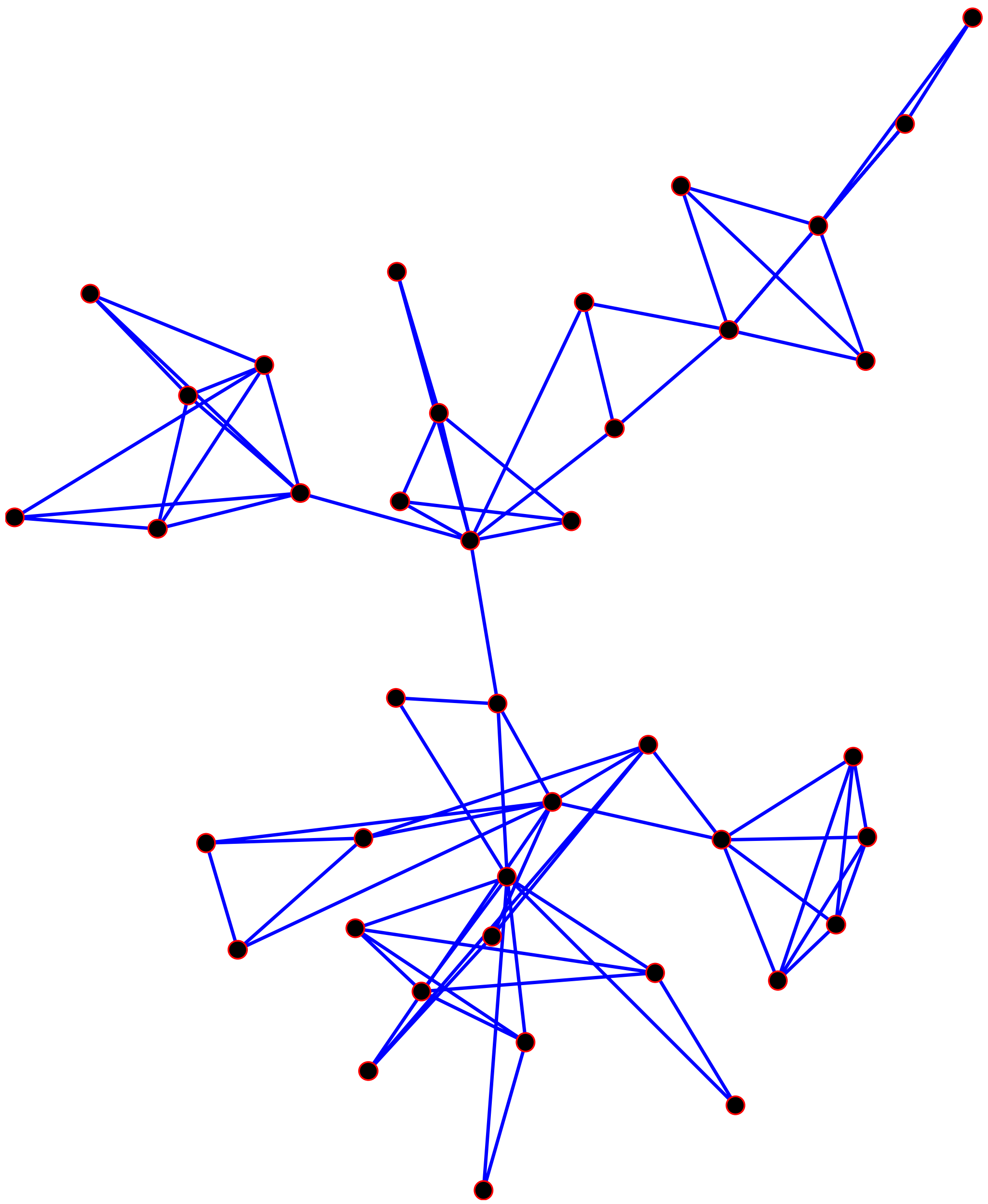
weights can be used to promote certain graph properties

"short cycle weights"

$$m_i = \|c_i\|_1$$

$$\|\Sigma(\mathcal{G})\|_2^2 = 48.704$$

Simulation Examples



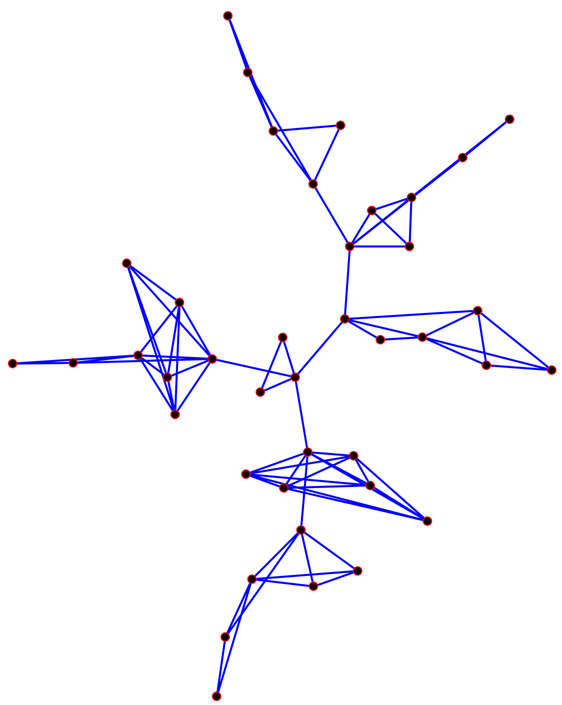
weights can be used to promote certain graph properties

"cycle correlation weights"

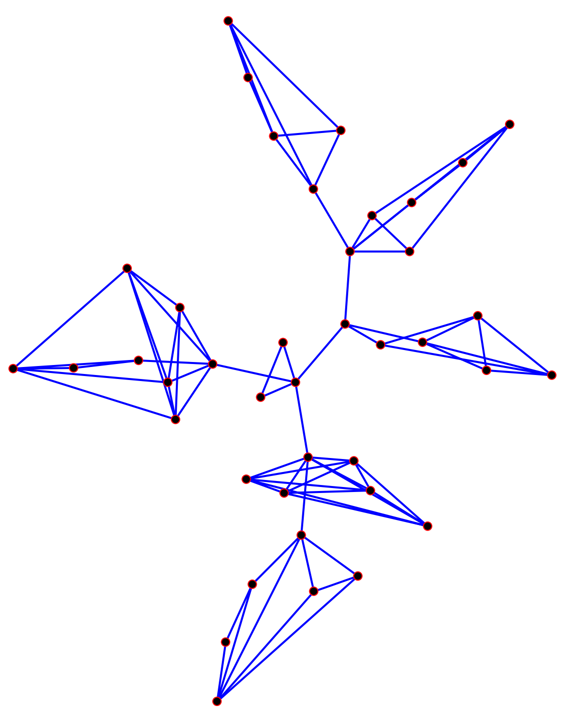
$$m_i = \frac{1}{|\mathcal{E}_c|} \sum_{j \neq i} \left| \left[T_{(\mathcal{T},c)} T_{(\mathcal{T},c)}^T \right]_{ij} \right|$$

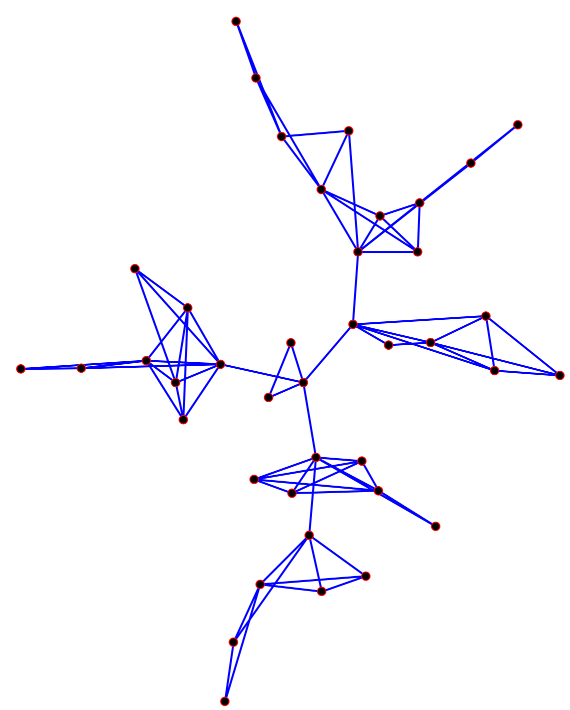
$$\|\Sigma(\mathcal{G})\|_2^2 = 48.939$$

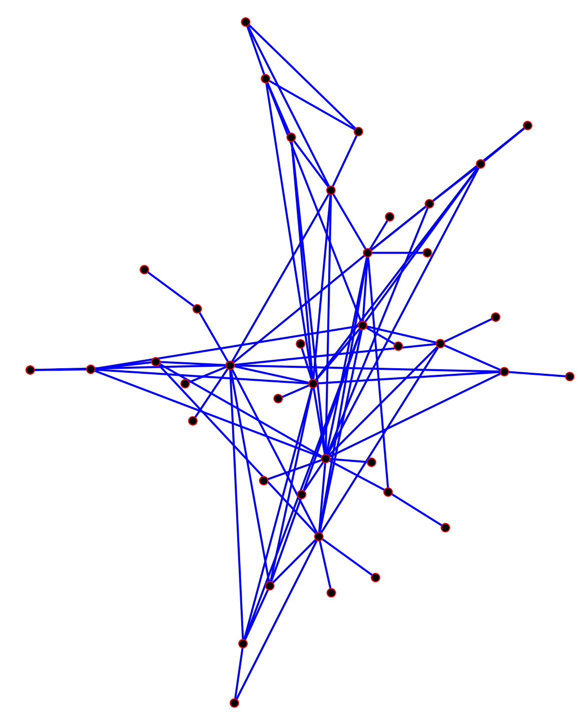
Simulation Examples



weights can be used to promote certain graph properties



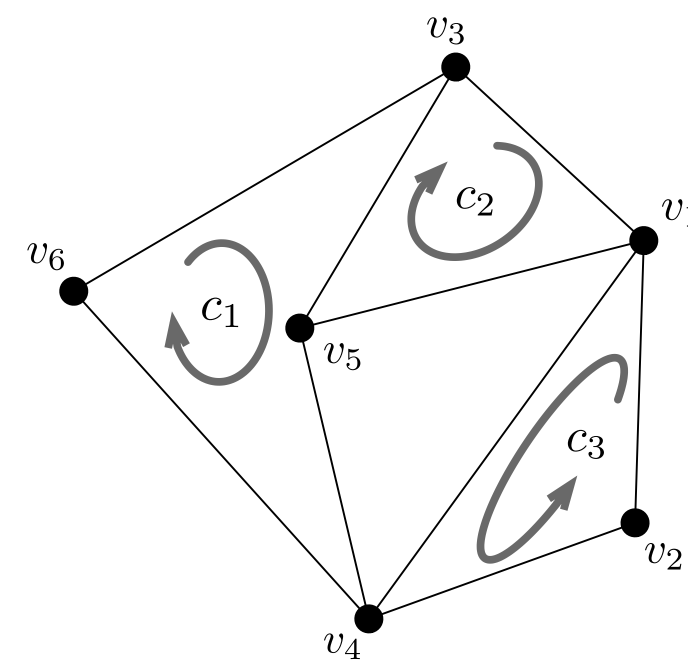




Concluding Remarks

role of cycles in consensus networks

- ✳ internal feedback
- ✳ performance



a tractable design procedure

- ✳ l1 optimization
- ✳ design of multi-agent systems

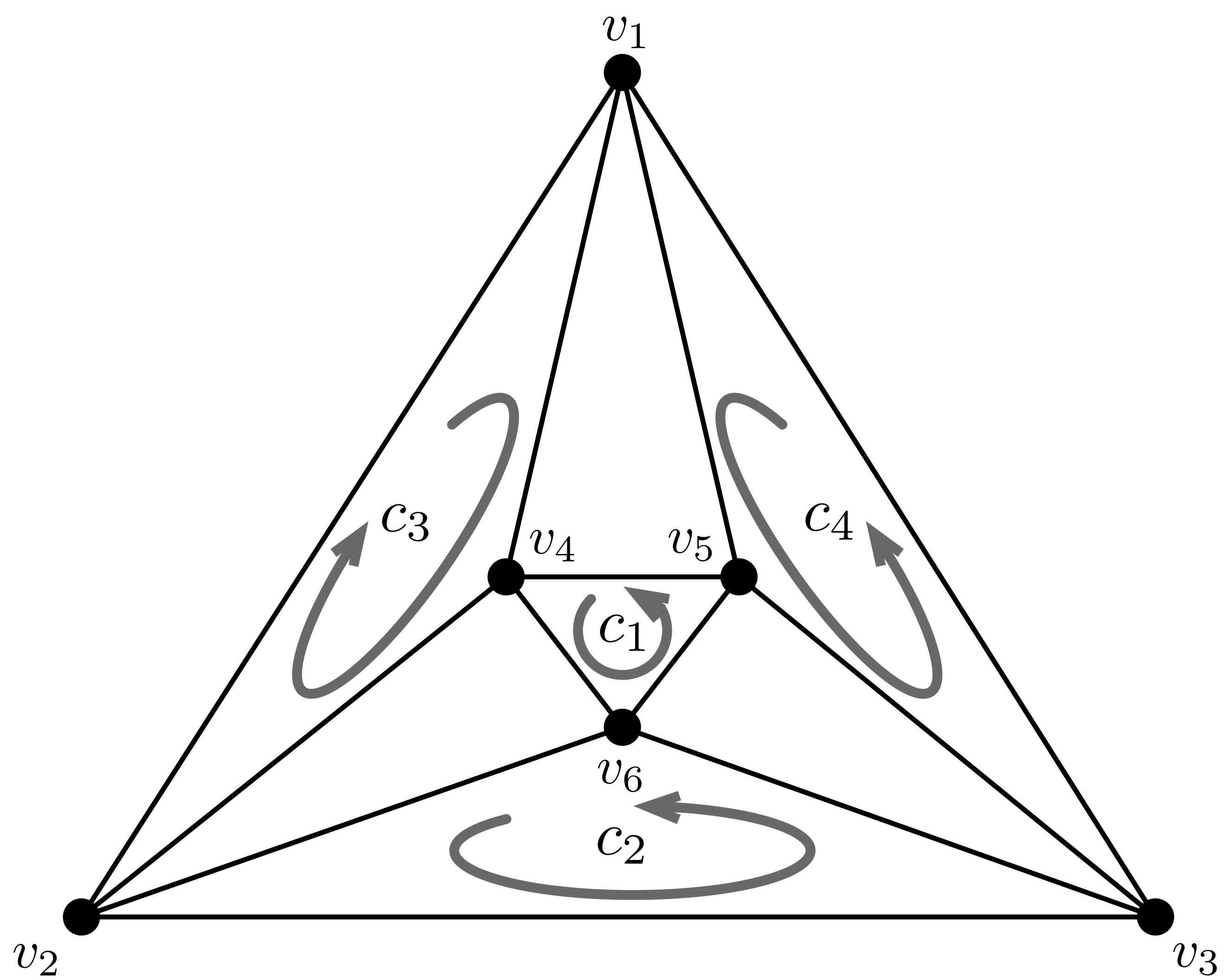
future works

- ✳ additional performance metrics
- ✳ push to large scale

Performance and design of cycles in consensus networks
Systems & Control Letters 62(1) : 85-96, 2013.

Concluding Remarks

Questions?



Eulerian Consensus Networks