E ... incidence matrix

L ... Laplacian

\[ L(k) = E K E^T = \sum_{\ell=1}^{m} k_{\ell} e_{\ell} e_{\ell}^T \]

Structured feedback gain \( K = \begin{bmatrix} k_1 & \cdots & k_m \end{bmatrix} \)

Arrive at a structured optimal control problem.

Let's first consider graphs that do not have loops. (trees)

Coordinate transformation:

\[
\begin{bmatrix}
\dot{y}(t) \\
\dot{x}(t)
\end{bmatrix} = \begin{bmatrix}
E^T \\
\frac{1}{N} I^T
\end{bmatrix} \begin{bmatrix}
x(t) \\
T
\end{bmatrix}
\]

\( y(t) \) ... relative difference between adjacent nodes.

\( x(t) \) ... average node.

Then,

\[
\begin{bmatrix}
\dot{y}(t) \\
\dot{x}(t)
\end{bmatrix} = \begin{bmatrix}
-E_t^T E_t & K & 0 \\
O & 0 & 0
\end{bmatrix} \begin{bmatrix}
y(t) \\
x(t)
\end{bmatrix} + \begin{bmatrix}
E_t^T \\
\frac{1}{N} I^T
\end{bmatrix} d(t)
\]

- \( \bar{x}(t) \) is preserved when \( d(t) = 0 \), otherwise it drifts with random walk.
\[ \dot{y}(t) = -E_t^T E_t K_y \dot{y}(t) + E_t^T d(t) \]

\[ z(t) = \begin{bmatrix} \dot{E}_t \left( E_t^T E_t \right)^{-1} \\ -E_t K \end{bmatrix} y(t) \]

$H_2$-norm from $d$ to $z$:

\[ J(K) = \frac{1}{2} \text{trace} \left( G^\top K^{-1} + G K \right) \]

where \( G = E_t^T E_t \)

\[ J(K) = \frac{1}{2} \sum_{n=1}^{N-1} \left( \frac{1}{k_n g_n} + k_n g_n \right) \]

\[ = \frac{1}{2} \sum_{n=1}^{N-1} \frac{1 + \left( k_n g_n \right)^2}{k_n g_n} \]

Can minimize \( J(K) \) by minimizing each term

\[ \frac{1 + \left( k_n g_n \right)^2}{k_n g_n} \]

because we have separability between the index 'n' or between nodes.

So, if we use incidence matrix of a tree graph, we can separate the effect of nodes on the objective function, if \( J \) is the difference between the values of each node, and then we can solve the optimal control problem.
General undirected graphs.

Incidence matrix \[ E = \begin{bmatrix} E_t & E_c \end{bmatrix} \]
\[ \downarrow \]
Part of the incidence matrix where there is a loop (cycle).

Columns of \( E_c \) are linear combination of columns of \( E_t \).

Equality-constrained convex optimization problem

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{s.t.} & \quad Ax - b = 0
\end{align*}
\]

\[ \ell(x, y) = f(x) + y^T(Ax - b) \]

If \( f \) is differentiable,

\[ \nabla_x \ell(x, y) = \nabla f(x) + A^T y = 0 \]

\[ \begin{cases}
E \nabla f(x) = \frac{1}{2} x^T Q x \quad ; \quad Q = Q^T > 0 \\
Q x + A^T y = 0
\end{cases} \]

\[ \iff \begin{cases}
x = -Q^{-1} A^T y \\
\frac{x^{k+1}}{y^k} = -Q^{-1} A^T y^k \\
y^{k+1} = y^k + s^k (Ax^{k+1} - b)
\end{cases} \]

Advantage it may lead to distributed implementation.