Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Optimal modification of dynamical network topology

Yoonsoo Kim

Department of Aerospace and System Engineering,
Research Center for Aircraft Parts Technology,
Gyeongsang National University,
Jinju 660-701, Republic of Korea
(e-mail: yoonsoo@gnu.ac.kr).

Abstract: For a given network topology of linking dynamical systems, determining the least or the most important link(s) or edge(s) in the network in terms of a sensitivity or robustness measure is a complex combinatorial optimization problem. The purpose of solving this problem is to modify the given network topology, in the hope of using a less number of costly communication links while keeping or improving the network's performance. In this paper, this so-called network topology optimization (NTO) problem is approached via finding the least or the most sensitive edge(s) by analytically obtaining the sensitivity of each edge or numerically solving LMIs (linear matrix inequalities). Vehicle formation control simulations are given to support the merit of the proposed approach.

Keywords: Network topology; optimization; sensitivity; robustness.

1. PROBLEM STATEMENT AND INTRODUCTION

Consider a network (graph) $\mathcal{G}$ of n dynamical systems (nodes) connected via communication links (edges), [1] and assume that the network dynamics is described by a typical state model:

$$\dot{x} = Ax + Bu; \quad y = Cx, \tag{1.1}$$

where x and u are the vector with entries of x_i and u_i ($i = 1, 2, \cdots, n$) representing the state and control input of individual node, respectively; and $A = [a_{ij}]$, $B = [b_{ij}]$ and $C = [c_{ij}]$ are constant matrices with appropriate dimension and have a structure associated with the network topology which defines *who-talks-to-whom* between the nodes. Suppose a signal v is injected through one of the nodes. According to the signal's characteristics, v can be a reference input r that must be tracked by all the nodes on the graph; or the signal can be an unwanted disturbance w that must be rejected by the nodes. [2]. For the sake of reducing the inter-communication cost, one is now interested in modifying the graph, so that a smaller number of edges is used to track or reject the signal. In this regard, when $v = r$, it is desired to remove an edge(s) so that the network dynamics (transfer function from r to y; G_{ry}) is affected as little as possible. In contrast, when $v = w$, it is desired to remove an edge(s) so that the network dynamics keeps or gains robustness as much as possible, i.e. the H_∞-norm of G_{wy}, $\|G_{wy}\|_\infty$, is minimized.

The aforementioned so-called *network topology optimization* (NTO) problem is basically a combinatorial optimization problem. For a certain measure which correctly reflects the network dynamics, the NTO problem boils down to determining which edge(s) removal affects the measure the most/least, which certainly has a combinatorial nature. The most relevant works to the present NTO problem are found in Kim (2010); Zelazo and Mesbahi (2011). In Kim (2010), a bisection algorithm is proposed to determine the edge(s) affecting the network's algebraic connectivity $\lambda_2(\mathcal{G})$ the most/least. This bisection algorithm quickly computes $\lambda_2(\mathcal{G})$ after each edge removal by solving the so-called *secular equation*. The measure of $\lambda_2(\mathcal{G})$, however, represents how well *static* nodes are connected, rather than how sensitive or robust *dynamic* nodes are with respect to disturbances. Thus, this bisection algorithm is not directly applicable to NTO of present interest. In Zelazo and Mesbahi (2011), a network topology design (NTD) problem for relative sensing networks is discussed. Note that NTD is concerned with designing a complete network in order to meet a given performance requirement, whereas NTO of present interest is concerned with selecting (and then removing) an edge(s) while minimizing the original network's performance degradation. NTD may be formulated as a computationally challenging *mixed-integer semi-definite program*, and requires a relaxation technique (e.g. ignoring the integer constraint on decision variables as tried in Zelazo and Mesbahi (2011)) to be solved for large network cases.

Some other relevant works are also available in the recent literature. In Kim et al (2010); Zelazo and Allgöwer (2012), the network topology is optimized in terms of formation rigidity. In Klaus (2011), a probabilistic search technique is proposed to find a network topology which minimizes the expected time to detect a target. In Okuyama and Tsumura (2011), a given network topology is modified to minimize the network's diameter by adding a path. In the course of minimization, a so-called tree network system is used to approximate the network dynamics. In Shafi et al (2012), several conditions that characterize Laplacian spectral bounds are studied and used for a network topology design. As already mentioned, all these relevant works (except Zelazo and Mesbahi (2011)) try to design a network

[1] Network and graph, dynamical system and node, and communication link and edge shall be used interchangeably in this paper.

[2] See (2.2) for an example of injecting r into the system, and (2.8) for injecting w into the system

topology using graph-theoretical measures which do not fully accommodate the network dynamics. In this work, however, the modification of a given network topology (not a complex design of the entire network like Zelazo and Mesbahi (2011)) is tried to explicitly incorporate the network dynamics into the edge removal which leads to keeping or improving the network's dynamical performance.

The rest of paper is organized as follows. In §2, two kinds of NTO are presented: one is for $v = r$; and the other is for $v = w$. For the first kind, an edge-removal scheme is developed in such a way that the network dynamics (transfer function from r to y) G_{ry} changes as little as possible, or the peak sensitivity norm $\|\mathcal{S}\|_\infty$ is minimized. For the second kind, another edge-removal scheme is developed in such a way that the network dynamics keeps or gains robustness as much as possible, or the H_∞-norm of the network dynamics (transfer function from w to y) $\|G_{wy}\|_\infty$ is minimized. Concluding remarks follow in §3.

2. NETWORK TOPOLOGY OPTIMIZATION (NTO)

2.1 *For reference tracking: $v = r$*

One of the main control objectives for networked systems is to let all the dynamical systems in the network track a reference signal. For example, a group of flying vehicles may need to be controlled to chase a single target together or to maintain a fixed pattern of formation. Suppose a reference signal r is injected through some of the nodes, and the following form of control input in (1.1)

$$u = K_1 x + K_2 r \tag{2.2}$$

is applied to accomplish a control objective, where K_1 and K_2 are constant matrices with appropriate dimension. After combining (1.1) and (2.2), one can obtain the transfer function G_{ry} from r to y: $G_{ry} = C(sI - (A + BK_1))^{-1}BK_2$, where $A + BK_1$ is assumed to be Hurwitz.

In order to see the effect of edge removal (no communication between two dynamical systems) on the r-to-y relation, the sensitivity function $\mathcal{S} = [\mathcal{S}_{ij}]$ is defined as $\mathcal{S}_{ij} = \partial G_{ry}/\partial e_{ij}$ for some edge weight e_{ij}. Suppose each entry, say a_{ij}, of the state and control matrices, is a differentiable function of e_{ij}. Then, one can easily arrive at the following result after some straightforward calculus.

Theorem 1. For given i, j

$$\begin{aligned}\mathcal{S}_{ij} = &\frac{\partial C}{\partial e_{ij}}(sI - (A + BK_1))^{-1}BK_2 \\ &+ C(sI - (A + BK_1))^{-1}\left(\frac{\partial B}{\partial e_{ij}}K_2 + B\frac{\partial K_2}{\partial e_{ij}}\right) \\ &+ C(sI - (A + BK_1))^{-1}\left(\frac{\partial A}{\partial e_{ij}} + \frac{\partial B}{\partial e_{ij}}K_1 + B\frac{\partial K_1}{\partial e_{ij}}\right) \\ &\times (sI - (A + BK_1))^{-1}BK_2,\end{aligned} \tag{2.3}$$

where $\partial M/\partial e_{ij}$ ($M = [m_{ij}] = A, B, C, K_1$ or K_2) is an all-zero matrix except $\partial m_{ij}/\partial e_{ij}$ at (i, j) position. If $a_{ij} = e_{ij}$ and all the other matrices are insensitive to e_{ij}, (2.3) reduces to

$$\begin{aligned}\mathcal{S}_{ij} = C(sI - (A + BK_1))^{-1}E_{ij} \times \\ (sI - (A + BK_1))^{-1}BK_2,\end{aligned} \tag{2.4}$$

where E_{ij} is an all-zero matrix except an '1' at (i, j) position.

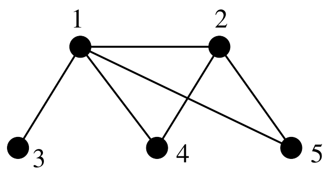

Fig. 1. Example of network topology

To illustrate the use of Theorem 1, consider a consensus network whose topology is given in Fig. 1. Assume that

$$A = -L_{\mathcal{G}} = \begin{bmatrix} -4 & 1 & 1 & 1 & 1 \\ 1 & -3 & 0 & 1 & 1 \\ 1 & 0 & -1 & 0 & 0 \\ 1 & 1 & 0 & -2 & 0 \\ 1 & 1 & 0 & 0 & -2 \end{bmatrix}$$

(see Godsil and Royle (2001) for how to construct the Laplacian matrix $L_{\mathcal{G}}$ for a graph $\mathcal{G}$), $B = [1, 0, 0, 0, 0]^T$, $C = [0, 0, 0, 0, 1]$, $K_1 = -C$ and $K_2 = 1$, $r = 1$, and the initial value of $x = [1, 2, 3, 4, 5]^T$. Note that the given data yield that all the nodes have the reference value of 1 at steady state. Then, (2.4) becomes

$$\begin{aligned}\mathcal{S}_{ij} &= -\underbrace{C(sI - (A - BC))^{-1}}_{\bar{c}} F_{ij} \underbrace{(sI - (A - BC))^{-1}B}_{\bar{b}} \\ &= -(\bar{c}_i - \bar{c}_j)(\bar{b}_i - \bar{b}_j),\end{aligned} \tag{2.5}$$

where F_{ij} is an all-zero matrix except an '1' at (i, i) and (j, j) positions, and a '-1' at (i, j) and (j, i) positions (due to the structure of $L_{\mathcal{G}}$), and $\bar{b}$ and $\bar{c}$ are the vectors with entries of $\bar{b}_i$ and $\bar{c}_i$, respectively. Noting that $A - BC$ is Hurwitz for all possible connected graphs obtained after removing an edge, $\|\mathcal{S}_{ij}\|_\infty$ based on (2.5) can be calculated and shown in Fig. 2-(a). As seen in the figure, e_{24} yields the smallest $\|\mathcal{S}_{ij}\|_\infty$ among *feasible* edges.[3] Fig. 2-(b) demonstrates that the removal of e_{24} does not affect the reference tracking performance much, whereas the removal of e_{15} yielding a large value of $\|\mathcal{S}_{15}\|_\infty$ does affect the transient behaviour much.

More results on the proposed scheme, e.g. performance test on random graphs, shall be included in a journal version of this paper.

2.2 *For disturbance rejection: $v = w$*

When an unwanted signal w comes into the network, edge removal can be done in such a way that $\|G_{wy}\|_\infty$ is minimized to reject w (again, this edge removal must not lead to an unstable or disconnected network). For this purpose, instead of checking the value of $\|G_{wy}\|_\infty$ after each edge removal, the following theorem on LMI (linear matrix inequality) can be utilized to devise a computationally tractable edge-removal rule.

Theorem 2. Let $\gamma > 0$, and G_{wy} is a stable transfer function from w to y, where $\dot{x} = Ax + Bw$ and $y = Cx$. Then, $\|G_{wy}\|_\infty < \gamma$ if and only if

$$\begin{bmatrix} XA + A^TX & XB & C^T \\ B^TX & -\gamma I & \mathbf{0} \\ C & \mathbf{0} & -\gamma I \end{bmatrix} < 0 \tag{2.6}$$

with a positive definite matrix $X > 0$, an identity matrix I, and a zero matrix $\mathbf{0}$ with appropriate dimension.

[3] For instance, e_{34} is not feasible, as it does not exist in the graph.

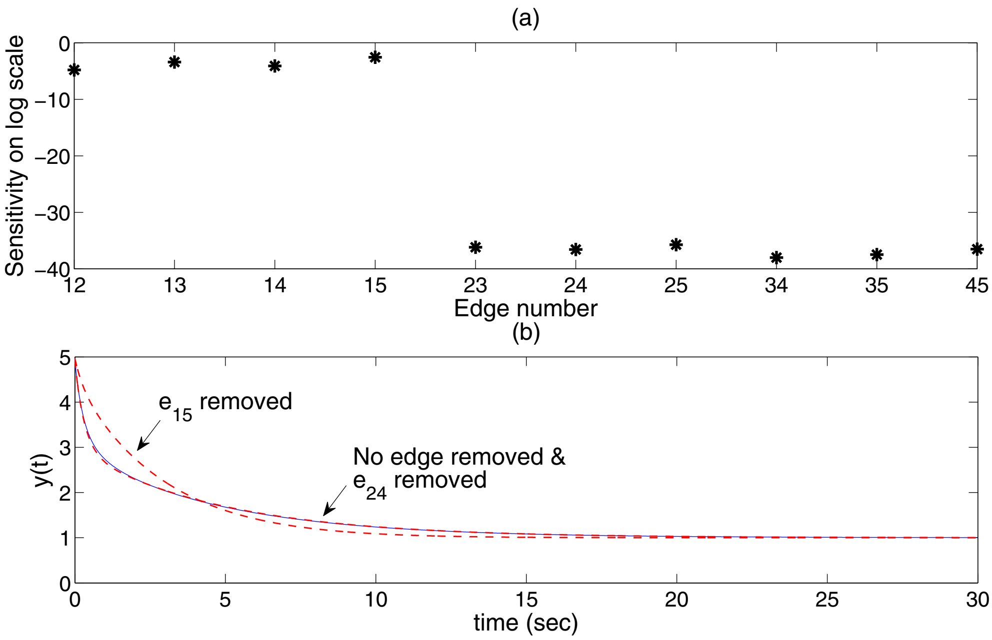

Fig. 2. $\|\mathcal{S}\|_\infty$ and reference tracking performance after edge removal

Proof. See the proof of Corollary 12.3 in Zhou and Doyle (1998).

Based on Theorem 2, one obvious solution strategy is to find an edge(s) such that the edge removal yields the smallest γ along with a feasible X. As choosing such an edge can be a time-consuming combinatorial problem, just a small change in the edge weight is considered instead as in §2.1.

Considering the effect of small changes δe_{ij}'s in e_{ij}'s on A, B and C, e.g. $\Delta A = \sum_{i,j}(\partial A/\partial e_{ij})\delta e_{ij}$, the best edge(s) may be found by solving the following optimization problem $\mathcal{P}$ which employs a modified version of (2.6) for δe_{ij}'s and ΔX with sufficiently small $|\delta e_{ij}|$ and $\|\Delta X\|$: minimize β subject to $X + \Delta X > 0$ and

$$M = [M_{11}, M_{12}, M_{13}; M_{21}, M_{22}, M_{23}; M_{31}, M_{32}, M_{33}] > 0,$$

where

$$\begin{aligned}
&M_{11} = (X+\Delta X)A + X\Delta A + (A+\Delta A)^T X + A^T \Delta X;\\
&M_{12} = (X+\Delta X)B + X\Delta B;\ M_{13} = (C+\Delta C)^T;\\
&M_{21} = B^T(X+\Delta X) + \Delta B^T X;\ M_{22} = -\beta I;\ M_{23} = \mathbf{0};\\
&M_{31} = C + \Delta C;\ M_{32} = \mathbf{0};\ M_{33} = -\beta I. \qquad (2.7)
\end{aligned}$$

After performing this optimization to obtain δe_{ij}'s, ΔX and β, the best edge can be found by identifying the largest entry of δe_{ij}'s.

To illustrate the use of $\mathcal{P}$, consider the 2-dimensional vehicle formation control problem in Lafferrire et al (2005). Each vehicle is assumed to have the following dynamics: $\dot{x}_i = A_{veh}x_i + B_{veh}u_i,\ i = 1, 2, \ldots, n,\ x_i \in \mathbf{R}^4$, where the entries of x_i represent the 2-dimensional position and the velocity of vehicle i, u_i represents the control inputs, and $A_{veh} = I_2 \otimes [0, 1; a_{21}, a_{22}]$, $B_{veh} = I_2 \otimes [0; 1]$. Here, I_2 is a 2-by-2 identity matrix, and $\otimes$ denotes the Kronecker product. Then, using a *decentralized* static feedback control law $u = FL(x - h)$, where u and x are vectors of u_i's and x_i's, the n-vehicle dynamics can be represented as follows: $\dot{x} = Ax + BFL(x-h)$ or $\dot{x} = (A + BFL)x - BFLh$. Here, $A = I_n \otimes A_{veh}$, $B = I_n \otimes B_{veh}$, $L = L_{\mathcal{G}} \otimes I_4$ and h is a constant vector associated with a desired formation (see Lafferrire et al (2005) for details). In Lafferrire et al (2005), it is shown that $a_{21} = 0$ and $F = I_n \otimes F_{veh}$ with $F_{veh} = I_n \otimes [-f_1, -f_2]$ with sufficiently large positive constants f_1 and f_2, can guarantee achieving any formation h for any connected graph.

The n-vehicle dynamics is now perturbed by a disturbance $w \in \mathbf{R}$ with a constant B_w, and the output y is defined as the vector of each vehicle's relative position with respect to the centre of n vehicles' positions, i.e.

$$\dot{x} = (A + BFL)x - BFLh + B_w w; \quad y = Cx \quad (2.8)$$

with $C = -L_{\mathcal{K}} \otimes [1/n, 0, 0, 0; 0, 0, 1/n, 0]$, where $L_{\mathcal{K}}$ denotes the Laplacian matrix corresponding to the complete graph $\mathcal{K}$ on n nodes. Note that L is the only matrix dependent on edge weights e_{ij}'s, and a small change in this matrix's entry is denoted by δl_{ij}. For a properly chosen F, $A + BFL$ has *stable* eigenvalues with negative real parts and two *unstable* eigenvalues with non-negative real parts. In fact, these unstable eigenvalues are those of A_{veh} and do not contribute to achieving a desired formation.[4] Also, they may yield an unstable transfer function G_{wy}, and so must be removed in order to use Theorem 2. For this purpose, a non-square P matrix can be introduced (see Kim and Mesbahi (2006) for how to construct such a P matrix) such that $x = Pz$, $P^T P = I$, and eigenvalues of $P^T(A + BFL)P$ are the same as the stable eigenvalues of $A + BFL$. Then, (2.8) becomes $\dot{z} = P^T(A + BFL)Pz - P^T BFLh + P^T B_w w = \bar{A}z - P^T BFLh + \bar{B}w$ and $y = CPz = \bar{C}z$.

[4] Due to these unstable eigenvalues, vehicles do not normally reach a steady state but move together in a certain direction after a desired formation is achieved. The speed of convergence to a desired formation is determined by the stable eigenvalue with the smallest real part in magnitude.

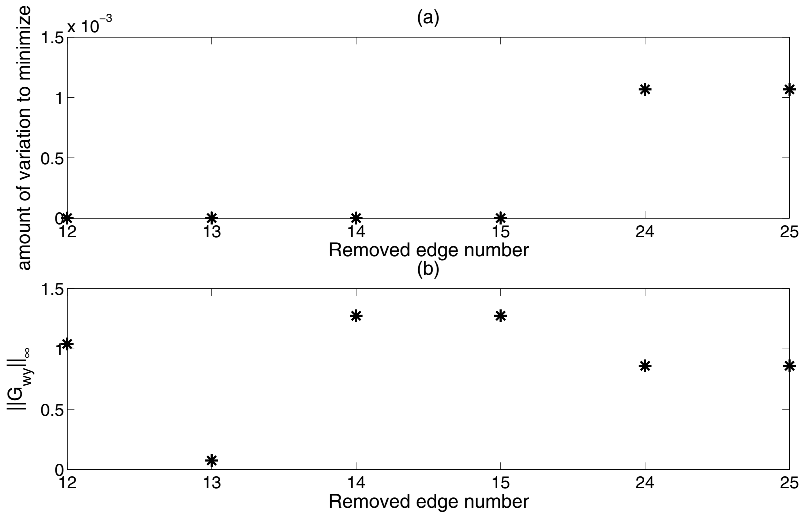

Fig. 3. $|\delta l_{ij}|$ and $\|G_{wy}\|_\infty$ after edge removal: (a) is obtained by solving $\mathcal{P}$, and indicates that e_{25} contributes to the minimization of β the most; (b) is obtained by calculating $\|G_{wy}\|_\infty$ after each edge removal, and indicates that e_{13} yields the smallest $\|G_{wy}\|_\infty$ and e_{25} yields the second smallest $\|G_{wy}\|_\infty$; since e_{13} renders the network disconnected, e_{25} is the best *feasible* one yielding the smallest $\|G_{wy}\|_\infty = \gamma_{\text{min}}$, which agrees with the result from (a), i.e. $\gamma_{\text{min}} = \gamma_{\text{actual}}$.

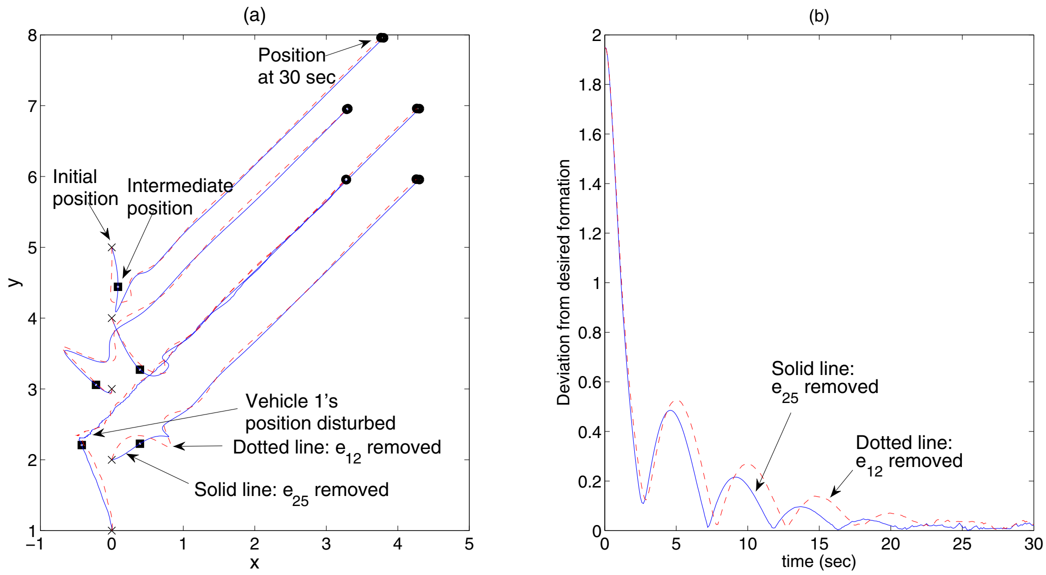

Fig. 4. Performance comparison between different edge removals for $\mathcal{G}$ in Fig. 1: (a) five vehicles initially aligned in a line (marked as 'x') converge to a desired pentagon formation (marked as 'o') via an intermediate formation (marked as 'square'); (b) deviation from desired formation: removal of e_{25} yields better transient performance than removal of e_{12} does (in terms of $\int_0^\infty \|y - Ch\|_2\, dt$).

Since $\bar{A}$ is now Hurwitz and the only matrix dependent on e_{ij}'s, (2.7) can be used with $\bar{A}$, $\overline{\Delta A}$, $\bar{B}$ and $\bar{C}$ in place of A, ΔA, B and C, respectively, and $\Delta B = \Delta C = 0$. Note that $\overline{\Delta A} = P^T BF \Delta LP = P^T BF(\Delta L_{\mathcal{G}} \otimes I_4)P$. For $A_{veh} = [0, 1; 0, 0.1]$, $B_w = [1, 0, \cdots, 0]^T$, $F = [-1/2, -1/2]$ and $L_{\mathcal{G}}$ corresponding to the graph in Fig. 1, the magnitudes of entries of $\Delta L_{\mathcal{G}}$, $|\delta l_{ij}|$, are shown in Fig. 3-(a) after solving $\mathcal{P}$ (along with move limits on $\Delta L_{\mathcal{G}}$ and ΔX) via SeDuMi (Sturm (1999)) and YALMIP (Löfberg (2004)). The figure indicates that e_{25}, contributes to the minimization of β the most. In order to check if the selected edge indeed yields the best performance, $\|G_{wy}\|_\infty$ is calculated by minimizing γ subject to (2.6) after each edge removal and the result is shown in Fig. 3-(b). The figure shows that e_{13} yields the smallest $\|G_{wy}\|_\infty$. However, this edge is not a feasible one as it renders the graph disconnected. Clearly, the second best edge or the best feasible edge matches to the one found by solving $\mathcal{P}$. Fig. 4 shows how different edge removals affect the performance when a random signal ($w(t) \in [-0.1, 0.1]$ for each t) is injected into the first vehicle's position. The figure demonstrates that the removal of the best e_{25} lets the vehicles converge to a desired pentagon formation (marked as 'o' in Fig. 4-(a)) from an initial line formation (marked as 'x' in Fig. 4-(a)), with a less deviation from the desired formation than when e_{12} is removed (see Fig.

4-(b)). Here, the deviation is defined as $d(t) = \|y(t) - Ch\|_2$, [5] where $h = h_p \otimes h_v = [0, 0, 1, 0, 0, 1, 1, 1, 1/2, 2]^T \otimes [1, 0]^T$ defines the desired formation. [6] In order to see the quantitative effect of w on d, the comparison of d_{12} ($d(t)$ with no w and removal of e_{12}), d_{12}^w ($d(t)$ with w and removal of e_{12}), d_{25} and d_{25}^w is performed. As expected, $\|d_{25} - d_{25}^w\|_2 = 0.00984 \leq \|d_{12} - d_{12}^w\|_2 = 0.0108$ implies that the removal of e_{25} attenuates the disturbance effect better than the removal of e_{12} does.

3. CONCLUDING REMARKS

In this paper, a complex combinatorial NTO (network topology optimization) problem was posed and solved. Unlike similar problems in the literature, this NTO problem is considered using sensitivity and robustness measures to accommodate the network dynamics. Two kinds of the NTO problem were identified: one is for keeping the network dynamics the same as much as possible after removing a communication link(s); and the other is for gaining the network's robustness as much as possible after removing a communication link(s). The first kind was approached by analytically obtaining the sensitivity of the closed-loop transfer function G_{ry} with respect to each edge, then the edge(s) which yields the least sensitivity in terms of peak norm was removed. The second kind was approached by numerically solving LMIs which help to identify the edge(s) yielding the largest decrease in the H_∞-norm of G_{wy}. This proposed scheme was tested on random graphs (not shown in this paper), and the obtained solutions were close (within 1% in terms of $\|G_{wy}\|_\infty$) to the optimal ones with high probability.

A possible extension of the present work is modifying the proposed scheme to account for the multiple-edge removal case. This extension shall also be included in a journal version of this paper.

ACKNOWLEDGEMENTS

This research was supported by Basic Science Research program (Grant No. 2012-1060) and Priority Research Centers Program (Grant No. 2011-0031383) through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology.

REFERENCES

Y. Kim, "Bisection algorithm of increasing algebraic connectivity by adding an edge," *IEEE Transactions on Automatic Control*, vol. 55, no. 1, pp. 170–174, 2010.

D. Zelazo and M. Mesbahi, "Graph-Theoretic Analysis and Synthesis of Relative Sensing Networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 5, pp. 971–982, 2011.

Yanghyun Kim, G. Zhu and J. Hu, "Optimizing formation rigidity under connectivity constraints," *In Proc. 49th IEEE Conference on Decision and Control*, pp. 6590–6595, 2010.

D. Zelazo and F. Allgöwer, "Growing Optimally Rigid Formations," *In Proc. American Control Conf.*, Montreal, Canada, pp. 3901-3906, 2012

[5] $\|\cdot\|_2$ denotes the 2-norm or Euclidean norm of a vector.

[6] h_p represents the relative position of each vehicle with respect to the first vehicle. For example, the fifth vehicle's desired relative position with respect to the first vehicle is $[1/2, 2]$. The '0' in h_v represents zero relative velocity between vehicles.

C. Klaus, "Probabilistic search on optimized graph topologies," *Master's Thesis*, Naval Postgraduate School, 2011.

T. Okuyama and K. Tsumura, "Path choice problem on multi-agent systems: Heterogeneous case," *In Proc. SCIE Annual Conference*, pp. 1355–1360, 2011.

S.Y. Shafi, M. Arcak and L. El Ghaoui, "Graph Weight Allocation to Meet Laplacian Spectral Constraints," *IEEE Transactions on Automatic Control*, vol. 57, no. 7, pp. 1872–1877, 2012.

A. Ghosh and S. Boyd, "Growing well-connected graphs," in *Proc. IEEE Conf. Decision Control*, Dec. 2006, pp. 6605–6611.

B. Korte and J. Vygen, *Combinatorial Optimization: Theory and Algorithms (Second Edition),* Springer, Berlin, 2002.

C. Godsil and G. Royle, *Algebraic Graph Theory,* Springer, New York, 2001.

K. Zhou and J.C. Doyle, *Essentials of robust control,* Prentice-Hall, Inc., New Jersey, 1998.

Y. Kim and M. Mesbahi, "On maximizing the second smallest eigenvalue of a state-dependent graph Laplacian," *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 116–120, 2006.

G. Lafferrire, A. Williams, J. Caughman and J. J. P. Veerman, "Decentralized control of vehicle formations," *Systems & Control Letters*, vol. 54, pp. 899–910, 2005.

J. F. Sturm, Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones, *Optimization Methods and Software*, vol. 11-12, 1999, pp. 625-653.

D. Peaucelle, D. Henrion, Y. Labit and K. Taitz, *Users Guide for SEDUMI INTERFACE 1.04*, LAAS - CNRS, 2002.

J. Löfberg, "YALMIP : A toolbox for modeling and optimization in MATLAB", *In Proc. CACSD Conf.*, 2004.