Sell your weakness: a low-cost high-efficiency active pose-graph SLAM method for multiple robots
Supplementary Material

Yongbo Chen*, Liang Zhao*, Ki Myung Brian Lee*,
Chanyeol Yoo*, Shoudong Huang* and Robert Fitch*
*

September 16, 2019

This document provides supplementary material to the paper [1]. Therefore, it should not be considered as a self-contained document, but instead regarded as an appendix of [1], and cited as:

"Y. Chen, L. Zhao, S. Huang, R. Fitch, K. M. B. Lee and C. Yoo, Sell your weakness: a low-cost high-efficiency active pose-graph SLAM method for multiple robots, (Supplementary Material), 2019."

Throughout this document, notations used are the same as those in [1].

1 Problem 1 Explanation

In this part, we focus on Problem 1 in [1]. In order to explore this problem, we first discuss the situation, whose weight value of the added edge is 1, and then discuss the common situation.

For a pose graph $\mathcal{G}^0$, if I add an edge (new measurement), named 1-ESP, to it, can I find a equal operation using a node with 2 edges, named 1-NESP, to replace it or can I estimate their differences on the information aspect? Two different structures are shown in Fig. 1. Their base pose graph parts are the same.

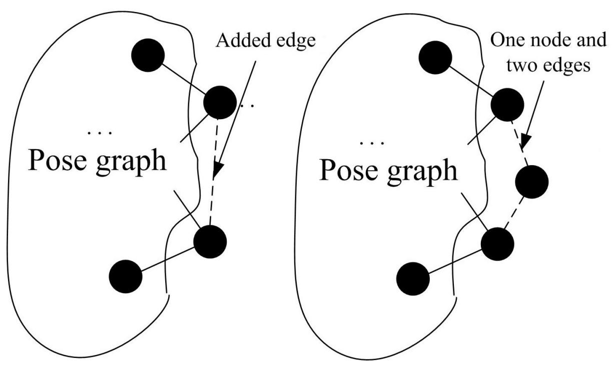

Figure 1: 1-ESP and 1-NESP problems

1.1 Problem 1 Solution

1.1.1 Weight value is equal to 1

Based on Eq.(9) in [1], we can get the D-optimality metric of 1-ESP is almost equal to the tree-connectivity of the whole graph. So we can directly discuss the tree-connectivity instead of the origi-

*All authors are with Faculty of Engineering and Information Technology, University of Technology Sydney, Ultimo, NSW, 2007 Australia, e-mail: Yongbo.Chen@student.uts.edu.au.

nal metric. Set the tree-connectivity of the original pose graph is $n\log(t_w(\mathcal{G}^0_{\mathbb{R}^n}))+d\log(t_w(\mathcal{G}^0_{SO(n)}))$, where $\mathcal{G}^0_{\mathbb{R}^n}$ and $\mathcal{G}^0_{SO(n)}$ are the rotation graph and translation graph obtained by $\mathcal{G}^0$. Then we add an edge to graph $\mathcal{G}^0$. Firstly, let's set the weighted value of the added edge is 1, then its new tree-connectivity of the new pose graph $\mathcal{G}^e$ can be obtain by the following steps:

In order to count the new tree-connectivity, we can consider two different situations for the spanning tree by adding an edge, shown in Fig.2.

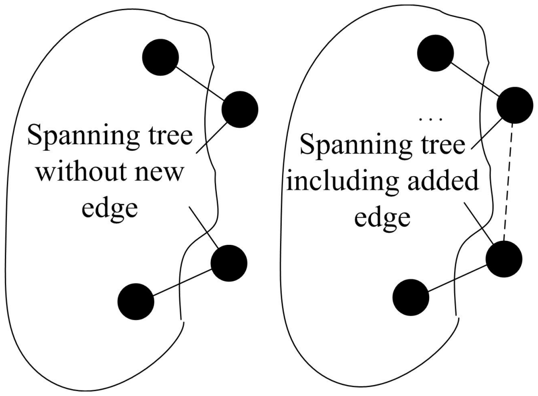

Figure 2: Two kinds of spanning trees in 1-ESP

The first kind of spanning trees does not include the new edge. The number of spanning tree of this situation is equal to the one of the original graph: $t_w(\mathcal{G}^0_{\mathbb{R}^n}) + t_w(\mathcal{G}^0_{SO(n)})$. The second one includes the added edge. The number of spanning tree of this situation is $t_w(\mathcal{G}^1_{\mathbb{R}^n}) + t_w(\mathcal{G}^1_{SO(n)})$. So we can get the tree connectivity of the graph $\mathcal{G}_e$ added a edge is:

$$\begin{aligned}&\log(\det(\boldsymbol{\mathcal{I}_{nD}}(\mathcal{G}^e))) \approx n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^e_{SO(n)}))\\&= n\log(t_w(\mathcal{G}^0_{\mathbb{R}^n}) + t_w(\mathcal{G}^1_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^0_{SO(n)}) + t_w(\mathcal{G}^1_{SO(n)}))\end{aligned} \tag{1}$$

For 1-NSP, the new tree-connectivity of the new pose graph includes three cases, which is shown in Fig.3.

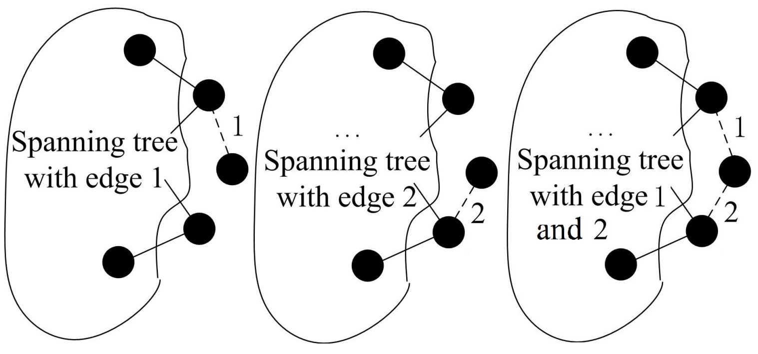

Figure 3: Two kinds of spanning trees in 1-NESP

Because of the added node, at least one measurement between edge 1 and 2 is required. The first two cases means that the spanning tree only includes one of these two edges. In this situation, the weighted numbers of spanning trees of these two cases are equal to the original graph $\mathcal{G}^0$. The rest case means that the spanning tree includes both two edges. By this way, we can see these node and two edges as one edge, which leads that its weighted numbers of spanning trees is equal to the one of the second case of 1-ESP. So we can get the weighted number of spanning trees of the pose graph adding one node and two edges $\mathcal{G}^n$ is:

$$\begin{aligned}
&\log(\det(\boldsymbol{\mathcal{I}}_{\boldsymbol{nD}}(\mathcal{G}^n))) \approx n\log(t_w(\mathcal{G}^n_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^n_{SO(n)})) \\
&= n\log(\underbrace{2t_w(\mathcal{G}^0_{\mathbb{R}^n})}_{\text{add edge 1 or edge 2}} + \underbrace{t_w(\mathcal{G}^1_{\mathbb{R}^n})}_{\text{add edge 1 and 2}}) \\
&+ d\log(\underbrace{2t_w(\mathcal{G}^0_{SO(n)})}_{\text{add edge 1 or edge 2}} + \underbrace{t_w(\mathcal{G}^1_{SO(n)})}_{\text{add edge 1 and 2}})
\end{aligned} \tag{2}$$

Because $t_w(\mathcal{G}^0_{\mathbb{R}^n}) > 0$, $t_w(\mathcal{G}^0_{SO(n)}) > 0$, $t_w(\mathcal{G}^1_{\mathbb{R}^n}) > 0$ and $t_w(\mathcal{G}^1_{SO(n)}) > 0$, based on (10), we have:

$$\begin{aligned}
&n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^e_{SO(n)})) \\
&= n\log(t_w(\mathcal{G}^0_{\mathbb{R}^n}) + t_w(\mathcal{G}^1_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^0_{SO(n)}) + t_w(\mathcal{G}^1_{SO(n)})) \\
&< n\log(t_w(\mathcal{G}^n_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^n_{SO(n)})) \\
&= n\log(2t_w(\mathcal{G}^0_{\mathbb{R}^n}) + t_w(\mathcal{G}^1_{\mathbb{R}^n})) + d\log(2t_w(\mathcal{G}^0_{SO(n)}) + t_w(\mathcal{G}^1_{SO(n)})) \\
&< n\log(2t_w(\mathcal{G}^0_{\mathbb{R}^n}) + 2t_w(\mathcal{G}^1_{\mathbb{R}^n})) + d\log(2t_w(\mathcal{G}^0_{SO(n)}) + 2t_w(\mathcal{G}^1_{SO(n)})) \\
&= n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n}) + d\log(t_w(\mathcal{G}^e_{SO(n)})) + n\log 2 + d\log 2
\end{aligned} \tag{3}$$

So we can get the lower bound LB and the upper bound UB of the tree-connectivity of the pose graph $\mathcal{G}^n$ is:

$$\begin{aligned}
LB &= n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^e_{SO(n)})) \\
UB &= n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^e_{SO(n)})) + (n+d)\log 2
\end{aligned} \tag{4}$$

1.1.2 Normal weighted value

In this section, the weighted values of the added edge is set to be $\omega_1 > 1$ and the replaced edge values of the added node with two edges are set to be both equal to ω_2, meeting $2\omega_1 \geq \omega_2 \geq \omega_1 > 1$. Then, we aim to solve the same question corresponding to the previous section.

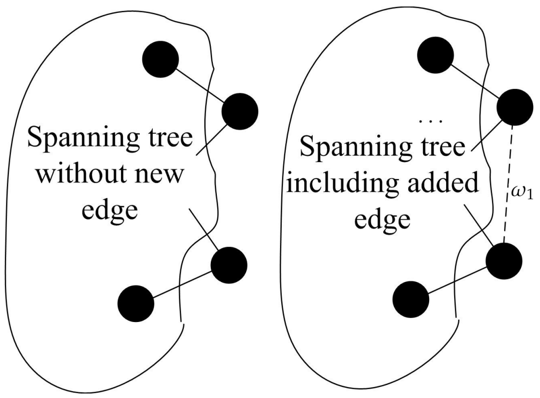

Figure 4: Two kinds of spanning trees with non-one edge in 1-ESP

Based on the definition 3.1 in [1] and (1), we can get the tree connectivity of the graph $\mathcal{G}_0$ added a non-one edge is:

$$\begin{aligned}
&\log(\det(\boldsymbol{\mathcal{I}}_{\boldsymbol{nD}}(\mathcal{G}^e))) \approx n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^e_{SO(n)})) \\
&= n\log(t_w(\mathcal{G}^0_{\mathbb{R}^n}) + \omega_1 t_w(\mathcal{G}^1_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^0_{SO(n)}) + \omega_1 t_w(\mathcal{G}^1_{SO(n)}))
\end{aligned} \tag{5}$$

The tree-connectivity of the corresponding graph adding one node and two edges is:

$$
\begin{aligned}
&\log(\det(\boldsymbol{\mathcal{I}_{nD}}(\mathcal{G}^e))) \approx n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^e_{SO(n)}))\\
&= n\log(t_w(\mathcal{G}^0_{\mathbb{R}^n}) + \omega_1 t_w(\mathcal{G}^1_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^0_{SO(n)}) + \omega_1 t_w(\mathcal{G}^1_{SO(n)}))\\
&\le \log(\det(\boldsymbol{\mathcal{I}_{nD}}(\mathcal{G}^n))) \approx n\log(t_w(\mathcal{G}^n_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^n_{SO(n)}))\\
&= n\log(2\omega_2 t_w(\mathcal{G}^0_{\mathbb{R}^n}) + \omega_2^2 t_w(\mathcal{G}^1_{\mathbb{R}^n})) + d\log(2\omega_2 t_w(\mathcal{G}^0_{SO(n)}) + \omega_2^2 t_w(\mathcal{G}^1_{SO(n)}))\\
&= n\log(\omega_2) + n\log(2) + n\log(t_w(\mathcal{G}^0_{\mathbb{R}^n}) + \frac{\omega_2}{2} t_w(\mathcal{G}^1_{\mathbb{R}^n}))\\
&+ d\log(\omega_2) + d\log(2) + d\log(t_w(\mathcal{G}^0_{SO(n)}) + \frac{\omega_2}{2} t_w(\mathcal{G}^1_{SO(n)}))\\
&\le n\log(\omega_2) + n\log(2) + n\log(t_w(\mathcal{G}^0_{\mathbb{R}^n}) + \omega_1 t_w(\mathcal{G}^1_{\mathbb{R}^n}))\\
&+ d\log(\omega_2) + d\log(2) + d\log(t_w(\mathcal{G}^0_{SO(n)}) + \omega_1 t_w(\mathcal{G}^1_{SO(n)}))\\
&= n\log(\omega_2) + n\log(2) + d\log(\omega_2) + d\log(2) + n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^e_{SO(n)}))\\
&\approx n\log(\omega_2) + n\log(2) + d\log(\omega_2) + d\log(2) + \log(\det(\boldsymbol{\mathcal{I}_{nD}}(\mathcal{G}^e)))
\end{aligned} \tag{6}
$$

So we have:

$$
\begin{aligned}
&LB_1 \le n\log(t_w(\mathcal{G}^n_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^n_{SO(n)})) \le UB_1\\
&LB_1 = n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^e_{SO(n)}))\\
&UB_1 = (n+d)\log(\omega_2) + (n+d)\log(2) + n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^e_{SO(n)}))
\end{aligned} \tag{7}
$$

2 Problem 2 Explanation

The second problem is the similar situation, which has more than one node. Can we find a equal operation using m node with at least $m+1$ edges, named multi-NESP, to replace 1-ESP or can we estimate their differences on the information aspect? Two different pose-graph structures are shown in Fig. 5. Their pose graph parts are the same.

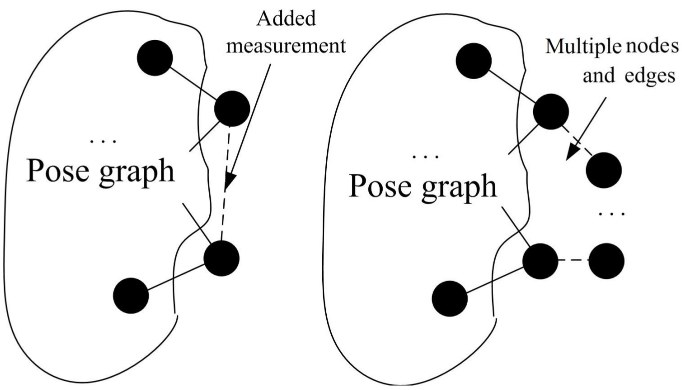

Figure 5: 1-ESP and multi-NESP

2.1 Problem 2 Solution

2.1.1 1 chain strucuture with m nodes and $m+1$ edges

Assume the added m nodes and $m+1$ edges, whose weighted values are equal and defined as ω_3 meeting $C^1_{m+1}\omega_1 \ge \omega_3 \ge \omega_1 \ge 1$, in the m-NESP problem forms a chain structure whose two ending nodes are connected with the corresponding nodes of the 1-ESP problem.

Similar to (2), we can get the weighted number of spanning trees of the pose graph adding m

nodes and $m+1$ edges $\mathcal{G}^m$ is:

$$
\begin{aligned}
&\log(\det(\boldsymbol{\mathcal{I}_{nD}(\mathcal{G}^m)})) \approx n\log(t_w(\mathcal{G}^m_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^m_{SO(n)}))\\
&= n\log(\underbrace{C^1_{m+1}\omega_3^m t_w(\mathcal{G}^0_{\mathbb{R}^n})}_{\text{add a open edge line}} + \underbrace{\omega_3^{m+1} t_w(\mathcal{G}^1_{\mathbb{R}^n})}_{\text{add all edges}})\\
&+ d\log(\underbrace{C^1_{m+1}\omega_3^m t_w(\mathcal{G}^0_{SO(n)})}_{\text{add a open edge line}} + \underbrace{\omega_3^{m+1} t_w(\mathcal{G}^1_{SO(n)})}_{\text{add all edges}})
\end{aligned}
\tag{8}
$$

The tree-connectivity of the corresponding graph adding one node and two edges is:

$$
\begin{aligned}
&\log(\det(\boldsymbol{\mathcal{I}_{nD}}(\mathcal{G}^e))) \approx n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^e_{SO(n)}))\\
&= n\log(t_w(\mathcal{G}^0_{\mathbb{R}^n}) + \omega_1 t_w(\mathcal{G}^1_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^0_{SO(n)}) + \omega_1 t_w(\mathcal{G}^1_{SO(n)}))\\
&\le \log(\det(\boldsymbol{\mathcal{I}_{nD}}(\mathcal{G}^m))) \approx n\log(t_w(\mathcal{G}^m_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^m_{SO(n)}))\\
&= n\log(C^1_{m+1}\omega_3^m t_w(\mathcal{G}^0_{\mathbb{R}^n}) + \omega_3^{n+1} t_w(\mathcal{G}^1_{\mathbb{R}^n}))\\
&+ d\log(C^1_{m+1}\omega_3^m t_w(\mathcal{G}^0_{SO(n)}) + \omega_3^{m+1} t_w(\mathcal{G}^1_{SO(n)}))\\
&= n\log(\omega_3^m) + n\log(C^1_{m+1}) + n\log(t_w(\mathcal{G}^0_{\mathbb{R}^n}) + \frac{\omega_3}{C^1_{m+1}} t_w(\mathcal{G}^e_{SO(n)}))\\
&+ d\log(\omega_3^m) + d\log(C^1_{m+1}) + d\log(t_w(\mathcal{G}^0_{SO(n)}) + \frac{\omega_3}{C^1_{m+1}} t_w(\mathcal{G}^1_{SO(n)}))\\
&\le n\log(\omega_3^m) + n\log(C^1_{m+1}) + n\log(t_w(\mathcal{G}^0_{\mathbb{R}^n}) + \omega_1 t_w(\mathcal{G}^e_{SO(n)}))\\
&+ d\log(\omega_3^m) + d\log(C^1_{m+1}) + d\log(t_w(\mathcal{G}^0_{SO(n)}) + \omega_1 t_w(\mathcal{G}^1_{SO(n)}))\\
&= n\log(\omega_3^m) + n\log(C^1_{m+1}) + n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^e_{SO(n)}))\\
&+ d\log(\omega_3^m) + d\log(C^1_{m+1})\\
&\approx (nm+dm)\log(\omega_3) + (n+d)\log(C^1_{m+1}) + d\log(\det(\boldsymbol{\mathcal{I}_{nD}}(\mathcal{G}^e)))
\end{aligned}
\tag{9}
$$

2.1.2 Complex structure with more than $m+1$ edges

The chain structure shown in Section 2.1.1 is just a part of the complex structure with more than $m+1$ edges. Based on Theorem 4 in [2], we know that the tree-connectivity gain for a connected graph is normalized, monotone, and sub-modular. So the tree-connectivity of the case in Section 2.1.1 is a lower-bound of the complex one. Assume the pose-graphs corresponding this complex case are $\mathcal{G}^{m*}$, $\mathcal{G}^{m*}_{\mathbb{R}^n}$ and $\mathcal{G}^{m*}_{SO(n)}$, we have:

$$
\begin{aligned}
&\log(\det(\boldsymbol{\mathcal{I}_{nD}}(\mathcal{G}^e))) \approx n\log(t_w(\mathcal{G}^e_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^e_{SO(n)}))\\
&\le \log(\det(\boldsymbol{\mathcal{I}_{nD}}(\mathcal{G}^{m*}))) \approx n\log(t_w(\mathcal{G}^{m*}_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^{m*}_{SO(n)}))
\end{aligned}
\tag{10}
$$

which means that the solution of the 1-ESP problem can give a performance guarantee for the solution of the multi-NESP problem.

3 Some discussions about the bound performance

Let's consider the simplest way in Section 1.1.1. Because, for most complex base pose graphs, the weighted number of the spanning trees including both the edge 1 and 2 is obviously smaller than the other case $t_w(\mathcal{G}^1_{\mathbb{R}^n}) << t_w(\mathcal{G}^0_{\mathbb{R}^n}), t_w(\mathcal{G}^1_{SO(n)}) << t_w(\mathcal{G}^0_{SO(n)})$, we have:

$$
\begin{aligned}
&n\log(t_w(\mathcal{G}^n_{\mathbb{R}^n})) + d\log(t_w(\mathcal{G}^n_{SO(n)})) - LB >>\\
&UB - n\log(t_w(\mathcal{G}^n_{\mathbb{R}^n})) - d\log(t_w(\mathcal{G}^n_{SO(n)})) > 0
\end{aligned}
\tag{11}
$$

So it is easy to find that, compared with the lower bound, usually, its tree-connectivity $\log(t_w(\mathcal{G}^n_{\mathbb{R}^n}) + t_w(\mathcal{G}^n_{SO(n)}))$ gets closed to UB. In order to verify this result, we perform several simulations on some well-known datasets by adding two randomly selected nodes, including city10000, intel, CSAIL and manhattan. These two nodes are used to add one edge or a node with two edges. Based on (11),

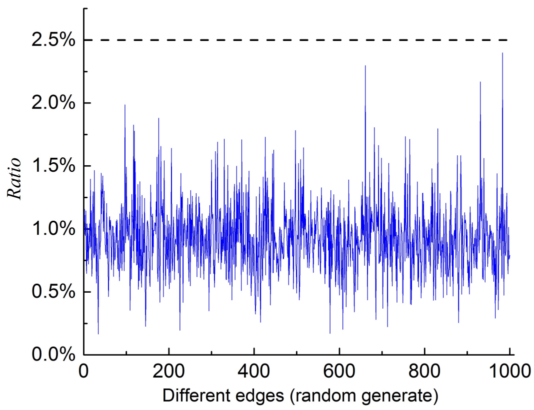

Figure 6: city10000

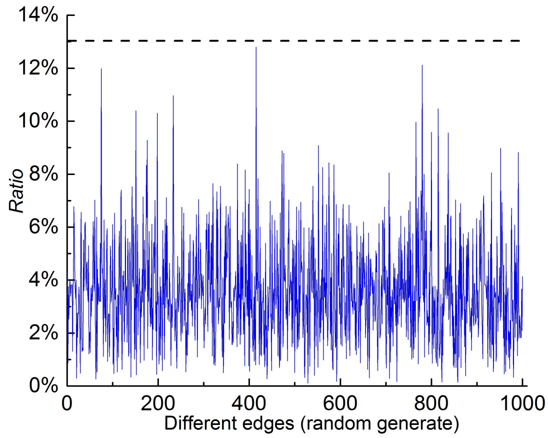

Figure 7: intel

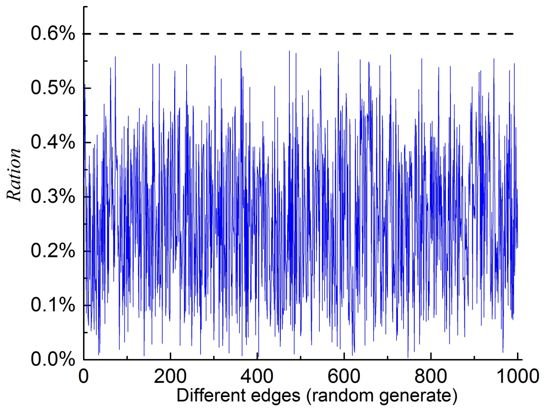

Figure 8: CSAIL

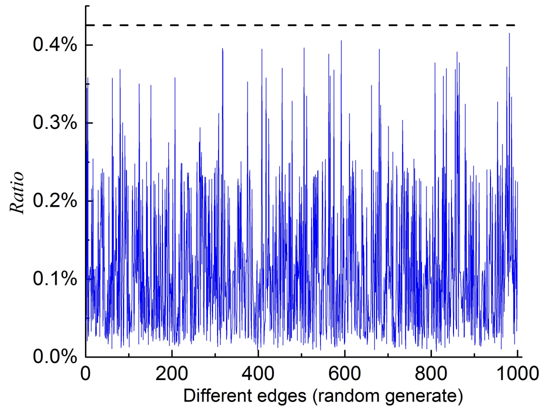

Figure 9: manhattan

the ration $Ratio = \frac{\log(t_w(\mathcal{G}^n_{SO(n)}))+\log 2-\log(t_w(\mathcal{G}^e_{SO(n)}))}{\log(t_w(\mathcal{G}^n_{SO(n)}))-\log(t_w(t_w(\mathcal{G}^n_{SO(n)}))}$ is shown in Fig. 6, Fig. 7, Fig. 8 and Fig. 9. The results show that, as an example, the rotation graph greatly meet our assumptions about $t_w(\mathcal{G}^1_{SO(n)}) << t_w(\mathcal{G}^0_{SO(n)})$, which helps to ensure our conclusion in (11). So even though these bound may not be very tight, in fact, they get closed to their upper bounds for most pose-graphs.

References

[1] Y. Chen, L. Zhao, S. Huang, R. Fitch, K. M. B. Lee and C. Yoo, Sell your weakness: a low-cost high-efficiency active pose-graph SLAM method for multiple robots, IEEE Robotics and Automation Letters (RA-L), 2018, submitted.

[2] K. Khosoussi, M. Giamou, G.S. Sukhatme, S. Huang, G. Dissanayake, and J.P. How, “Reliable graph topologies for SLAM," *The International Journal of Robotics Research*, 2018.