UNSYS digital

Journal of Unmanned System Technology

Agregation Scheme of Collision Avoidance Control and Formation Forming & Keeping by Topology Switching

Tata Sudiyanto[†], Bambang Riyanto Trilaksono[†], Agus Budiyono[‡], Widyawardana Adiprawita[†]

[†]Institut Teknologi Bandung, Bandung, Indonesia

[‡]Parametrik Research and Development, Bandung, Indonesia

Abstract— A control agregation scheme that combines a collision avoidance controller into a consensus-based formation controller is proposed. The collision avoidance controller part is designed using kinematic model approach, while the consensus-based formation controller part uses consensus algorithm with switching topology to achive non-conflicting control solution between the collision avoidance control part and the formation control part. We also investigate how the change in communication topology affects system's capability to achieve consensus.

Keyphrases—control system agregation, switching topology, formation control, consensus algorithm, collision avidance, kinematic model, multi-vehicle control system, UAV

DOI: 10.21535/just.v8i1.1042

I. INTRODUCTION

FORMATION forming and formation keeping (FFK) task, when considered at individual level, is essentially a tracking task (tracking or following a reference state, or reference trajectory, with its position in the formation as reference state/trajectory). The task performed by an individual UAV is coordinated with similar task performed by other individual UAVs in the team. The result is an emergence of a swarm behavior that fulfills the FFK task. On the other hand, collision avoidance (CA) task is, by definition, the opposite of tracking task (the reference trajectory is to be avoided). When certain condition occurs to an individual UAV in the team that the reference trajectory of both tasks allign with each other, an undesired behavior occurs. If the control system of that particular individual is more sensitive to tracking error, that individual tends to stay close to its postion in the formation, almost inaffected by the collision threat, which is a dangerously undesirable behavior. Or, if the control system is more sensitive to the collision threat, that individual's reaction to collision threat is better but may take longer time to return to its position in formation when the collision threat dissipates. Therefore, any design of swarm control system that seeks the capability to perform both FFK task and CA task must deal with with this problem.

There are many approaches in dealing with this undesired behaviour problem, but they can mainly be categorized into two methods: blending method, and hybrid method. The blending method blends the solution from the CA part with the FFK part according to certain optimized proportion. The hybrid method uses the CA part and the FFK part as a solution set on the individual level. Which part it executes is determined by specific situation. For example, when there is no Intruder detected, the controller execute the FFK part, and vice versa.

A work by Paul *et al* (2007) uses blended potential fields to produce the control system of a team of unmanned rotorcrafts[2]. There is an explicit remark about local minima in the blended potential fields but there is no solution proposed. Another work by Mastellone *et al* (2008) uses blended Lyapunov function in order to obtain a stable and collision-free control system for a team of nonholonomic land robot[3]. An implicit remark about local minima effect is mentioned on the report on the experiment on dynamic formation control where the collision avoidance controller is not activated. However, solution is not offered. Wang & Xin (2011) uses blended cost function to obtain an optimal FFK controller for use in obstacle-laden environment[5]. Kuriki & Namerikawa (2014) uses blended control protocol: The FFK part is a two-dimensional consensus-based controller, the CA part is a potential-based two-dimensional controller. To address local minima problem, Kuriki & Namerikawa introduce a third controller that works on vertical dimension. Whenever local minima is about to occur, the third controller is activated and drive the UAV to 'escape' via vertical motion[6]. Mondal *et al* (2017) proposed a blended potential field scheme where the CA part is adaptive in order to obtain a behavior that maintains a desired distance between team members[8]. Wen *et al* (2017) also uses blended potential field. However, Wen treats the formation and collision avoidance blending problem as a stochastic process, and treats the collision avoidance part as disturbance in order to mitigate the resulting undesired behavior whenever there is no risk of collision[9]. Lou & Yang (2018) uses blended solution: the FFK

Corresponding author: Tata Sudiyanto (e-mail: tatasudiyanto@yahoo.co.id)
This paper was submitted on May 7, 2020; and accepted on November 2, 2020.

part is consensus-based and the CA part is potential-based. The CA part is designed to prevent collision among team members. The agregated FFKCA control design is tested using simulations, among which the controller is tested under switching topology. However, the effect of switching topology to system's stability is not investigated.

Listmann *et al* (2009) uses a rule-based regulator is used to regulate controller output between FFK mode and CA mode[4]. Choutri *et al* (2019) uses hybrid method where the topology of the team is switched based on whether CA task is initiated or not. Team topology switching allows the team to break the formation into several groups of two members in leader-follower formations when CA controller is activated[12]. Choutri uses a leader-follower scheme for their FFK controller but does not report much about their CA controller[12].

In this paper, we propose a control agregation scheme that merges/combines a formation forming & keeping (FFK) controller and a collision avoidance (CA) controller to obtain a control system for a team of UAVs that is capable of performing CA task while also maintaining FFK task. CA controller part is designed using the kinematic-based modeling approach proposed in [10]. The FFK controller part adopts the transformation scheme in [7] that translates a formation control problem into the standard consensus algorithm. Our approach in the designs of CA part, FFK part and the agregation method are presented in Section II. In Section III, the simulation to test the design's performance is presented. Conclusion is presented in Section IV.

II. METHOD

A. The CA controller

In this research we propose a kinematic-based collision avoidance control scheme using the governing kinematic relations between UAV ego and the collision threat source. For each pair of UAV ego $i \in \{1,\cdots,n\}$ and a threat source $j \in \{1,\cdots,n,n+1,\cdots,n+n_{\text{intr}}\}$, we can quantify the collision threat imposed by the presence of an intruder to the UAV ego as CTS (collision threat state), which depends on its relative radial distance and its angular distance (1).

$$CTS_{i,j} := \frac{\|\vec{v}_{j,i}\|}{\|\vec{r}_{i,j}\| - r_{\text{fsph}i,j}} \cdot \left(\sin\psi_{\text{miss}i,j} - \sin\psi_{\text{enc}i,j}\right) \tag{1}$$

n is the number of individuals in the team. n_{intr} is the number of intruders that are not member of the team. If $j \le n$, the threat source is a member of the team. If $j > n$, the threat source is a foreign body that is not a member of the team; the threat source is called Intruder.

$\vec{r}_{i,j}$ is the vector of UAV ego's relative position to the threat source. $\vec{v}_{j,i}$ is the vector of threat source's relative velocity to UAV ego. $\psi_{\text{enc}i,j}$ is the encounter angle. $\psi_{\text{miss}i,j}$ is the miss angle. $r_{\text{fsph}i,j}$ is the radius of the forbidden sphere, that is, the sphere around threat source's body that must not be intruded by UAV ego. The radius of the forbidden sphere is the sum of UAV ego's sphere radius, the threat source's sphere radius, and a prescribe clearance distance r_{clear}. The description of parameters appears in (1) is shown in Figure 1.

In the presence of multiple collision threat source, each member will have to deal with the one that gives the greateset threat value.

$$\forall i \in \{1,\cdots,n\},\quad CTS_i = \max\left(\{CTS_{i,j}\}\right),\quad j = 1,\cdots,n_{\text{intr}} \tag{2}$$

Then, the collision avoidance control law is given by

$$\dot{\vec{v}}_{i\,\text{CA}} = \vec{\omega}_{i,j^*\,\text{CA}} \times \vec{v}_i \tag{3}$$

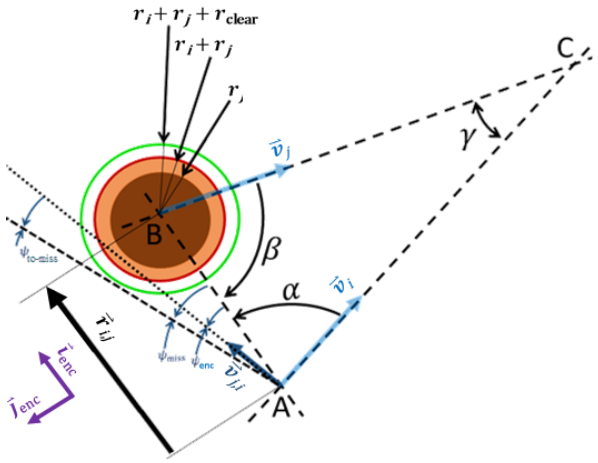


Figure 1 A geometric representation of encounter event

j^* is the identification value of the threat source that gives the greatest threat value. $\vec{\omega}_{\text{CA}}$ is the vector of the required minimum angular velocity of both bodies (UAV ego and the collision threat source) relative to each other to achieve miss event.

$$j^* = \arg\max\left(\left\{CTS_{i,j}\right\}\right) \tag{4}$$

$$\vec{\omega}_{i,j^*\,\text{CA}} = k_{\text{CA}} \cdot CTS \cdot \frac{\vec{r}_{i,j^*} \times \vec{v}_{j^*,i}}{\left\|\vec{r}_{i,j^*} \times \vec{v}_{j^*,i}\right\|} \tag{5}$$

B. The FFK controller

1) Consensus-based formation control problem

The multi-agent system regarding formation control problem is modeled as described in Figure 2. This multi-agent control problem can be formulized as a consensus-based control law as expressed in (6).

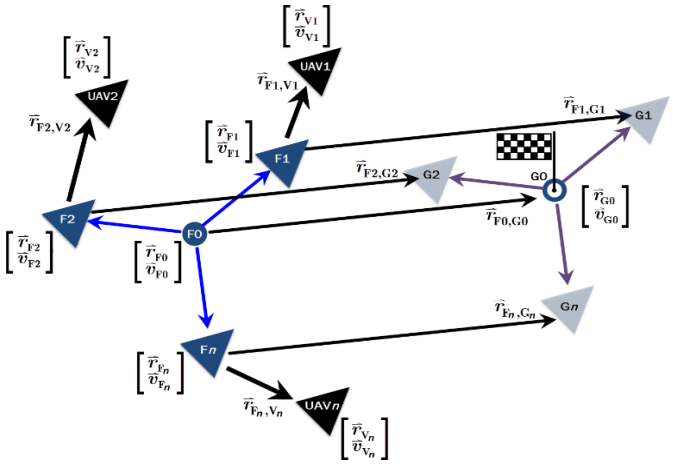


Figure 2 Formation flight control problem

$$\left.\begin{aligned}\dot{\vec{r}}_{\mathrm{Fi,V}i(t)} &= \vec{v}_{\mathrm{Fi,V}i(t)} \\ \dot{\vec{v}}_{\mathrm{Fi,V}i(t)} &= -a_{i,j_{\mathrm{G}}(t)}\cdot\left(k_{\mathrm{tr},\vec{r}}\cdot\left(\vec{r}_{\mathrm{Fi,V}i(t)}-\vec{r}_{\mathrm{G}0(t)}\right)+k_{\mathrm{tr},\vec{v}}\cdot\left(\vec{v}_{\mathrm{Fi,V}i(t)}-\vec{v}_{\mathrm{G}0(t)}\right)\right) \\ &\quad -\sum_{\substack{j=1\\ j\neq i}}^{n} a_{i,j(t)}\cdot\left(\gamma_{\vec{r}}\cdot\left(\vec{r}_{\mathrm{Fi,V}i(t)}-\vec{r}_{\mathrm{Fj,V}j(t)}\right)+\gamma_{\vec{v}}\cdot\left(\vec{v}_{\mathrm{Fi,V}i(t)}-\vec{v}_{\mathrm{Fj,V}j(t)}\right)\right)\end{aligned}\right\} \quad (6)$$

$$\vec{r}\in\mathbb{R}^3,\ \vec{v}\in\mathbb{R}^3,\ k_{\mathrm{tr}}\in\mathbb{R},\ \gamma\in\mathbb{R}$$

$$a_{i,j}\in\{0,1\},\ a_{i,j_{\mathrm{G}}}\in\{0,1\},\ \forall j\neq i$$

$$a_{i,j}\in\mathbb{R},\ a_{i,j_{\mathrm{G}}}\in\mathbb{R},\ \forall j=i$$

A stable consensus-based control system is achieved for all $k_{\mathrm{tr}\vec{r}}>0$, for all $k_{\mathrm{tr}\vec{v}}>0$, for all $\gamma_{\vec{r}}>0$, and for all $\gamma_{\vec{v}}>0$ by ensuring all $a_{i,j}$ correspond to a topology that has spanning tree, that is, all the corresponding Laplacian matrices to those topology must have exactly one zero-eigenvalue and the other eigenvalues have positive real part.

$$\begin{aligned}\lambda_{1(\boldsymbol{L}_n)} &= 0 \\ \mathrm{Re}\left(\lambda_{n(\boldsymbol{L}_n)}\right)\geq \mathrm{Re}\left(\lambda_{n-1(\boldsymbol{L}_n)}\right)\geq\cdots\geq \mathrm{Re}\left(\lambda_{2(\boldsymbol{L}_n)}\right)&> \mathrm{Re}\left(\lambda_{1(\boldsymbol{L}_n)}\right)\end{aligned} \quad (7)$$

where λ is the eigenvalue of Laplacian matrix $\boldsymbol{\mathcal{L}}$ whose elements are

$$\left.\begin{aligned}\boldsymbol{\mathcal{L}}_{n(t)} &= \left\{l_{i,j}\right\},\ \ \boldsymbol{\mathcal{A}}_{n(t)} = \left\{a_{i,j}\right\} \\ l_{i,j} &:= \begin{cases} -a_{i,j}, & j\neq i \\ \sum_{j=1,j\neq i}^{n} a_{i,j}, & j=i \end{cases}\end{aligned}\right\} \quad (8)$$

$$l_{i,j}\in\mathbb{R}$$

$\boldsymbol{\mathcal{A}}$ is the adjacency matrix.

Formation control system (6) can be expressed in compact form

$$\dot{\vec{\xi}}_{(t)} =: \boldsymbol{A}_{\mathrm{FFK}}\cdot\vec{\xi}_{(t)} + \boldsymbol{B}_{\mathrm{FFK}}\cdot\vec{\xi}_{\mathrm{G}(t)} \quad (9)$$

$$\boldsymbol{A}_{\mathrm{FFK}}\in\mathbb{R}^{6n},\ \ \boldsymbol{B}_{\mathrm{FFK}}\in\mathbb{R}^{6n}$$

where

$$\vec{\xi} := \begin{bmatrix}\vec{r}_1^{\mathrm{T}} & \vec{r}_2^{\mathrm{T}} & \cdots & \vec{r}_n^{\mathrm{T}} & \vec{v}_1^{\mathrm{T}} & \vec{v}_2^{\mathrm{T}} & \cdots & \vec{v}_n^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}},\ \ \vec{\xi}_{\mathrm{G}} := \begin{bmatrix}\vec{1}_{n\times1}\otimes\vec{r}_{\mathrm{G}_0} \\ \vec{1}_{n\times1}\otimes\vec{v}_{\mathrm{G}_0}\end{bmatrix} \quad (10)$$

$$\vec{r}_i := \vec{r}_{\mathrm{Fi,V}i},\quad \vec{v}_i := \vec{v}_{\mathrm{Fi,V}i},\quad i=1,2,\cdots,n \quad (11)$$

$$\boldsymbol{A}_{\mathrm{FFK}} := \left[\begin{array}{c:c}\boldsymbol{A}_{11} & \boldsymbol{A}_{12} \\ \hdashline \boldsymbol{A}_{21} & \boldsymbol{A}_{22}\end{array}\right],\qquad \boldsymbol{B}_{\mathrm{FFK}} := \left[\begin{array}{c:c}\boldsymbol{B}_{11} & \boldsymbol{B}_{12} \\ \hdashline \boldsymbol{B}_{21} & \boldsymbol{B}_{22}\end{array}\right] \quad (12)$$

$$\begin{aligned}\boldsymbol{A}_{11} &= \boldsymbol{0}_{n\times n}\otimes\boldsymbol{I}_3, & \boldsymbol{A}_{12} &= \boldsymbol{I}_n\otimes\boldsymbol{I}_3 \\ \boldsymbol{A}_{21} &= -\left(\gamma_{\vec{r}}\cdot\boldsymbol{L}_n + k_{\vec{r}}\cdot\boldsymbol{Q}_n\right)\otimes\boldsymbol{I}_3, & \boldsymbol{A}_{22} &= -\left(\gamma_{\vec{v}}\cdot\boldsymbol{L}_n + k_{\vec{v}}\cdot\boldsymbol{Q}_n\right)\otimes\boldsymbol{I}_3\end{aligned} \quad (13)$$

$$\begin{aligned}\boldsymbol{B}_{11} &= \boldsymbol{0}_{n\times n}\otimes\boldsymbol{I}_3, & \boldsymbol{B}_{12} &= \boldsymbol{0}_{n\times n}\otimes\boldsymbol{I}_3 \\ \boldsymbol{B}_{21} &= k_{\vec{r}}\cdot\boldsymbol{Q}_n\otimes\boldsymbol{I}_3, & \boldsymbol{B}_{22} &= k_{\vec{v}}\cdot\boldsymbol{Q}_n\otimes\boldsymbol{I}_3\end{aligned} \quad (14)$$

$$\boldsymbol{Q}_n = \begin{bmatrix} a_{1,j_{\mathrm{G}}} & 0 & \cdots & 0 \\ 0 & a_{2,j_{\mathrm{G}}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{n,j_{\mathrm{G}}}\end{bmatrix} \quad (15)$$

The value of $\vec{\xi}$ at any future time instant t_{k+1} is related to its value at present time t_k by a state transition matrix $\boldsymbol{\Phi}\in\mathbb{R}^{\boldsymbol{6n}}$.

$$\vec{\xi}_{(t_{k+1})} = \boldsymbol{\Phi}_{(t_{k+1},t_k)}\cdot\vec{\xi}_{(t_k)} \quad (16)$$

$$\boldsymbol{\Phi}_{(t_{k+1},t_k)} = e^{\boldsymbol{A}_{\mathrm{FFK}}\cdot(t_{k+1}-t_k)} = \boldsymbol{I} + \sum_{n=1}^{\infty}\frac{1}{n!}\cdot\left(\boldsymbol{A}_{\mathrm{FFK}}\cdot\left(t_{k+1}-t_k\right)\right)^n \quad (17)$$

If (7) is satisfied, $\vec{\xi}_{t_{k+1}}$ will converge to a steady state value as $(t_{k+1}-t_k)$ approaches infinity for any $\vec{\xi}_{t_k}$, *i.e.*, $\boldsymbol{\Phi}_{(t_{k+1},t_k)}$ will approach a rank-one matrix as $(t_{k+1}-t_k)\to\infty$.

2) Formation stability under switching topology

When the interaction topology is fixed, the existing spanning tree is maintained, and all the eigenvalues of its Laplacian matrix satisfy (7) for all time.

But if the team's interaction topology is evolving discretely, the evolution can be expressed as a set of graphs representing a sequence of occuring topologies along a set of certain intervals.

$$\left\{\boldsymbol{\mathcal{G}}_{n(t_1\leq t<t_2)},\boldsymbol{\mathcal{G}}_{n(t_2\leq t<t_3)},\cdots,\boldsymbol{\mathcal{G}}_{n(t_{p-1}\leq t<t_p)}\right\},\ \ t_p>t_{p-1}>\cdots>t_2>t_1 \quad (18)$$

with the corresponding set of Laplacian matrices,

$$\left\{\boldsymbol{\mathcal{L}}_{n(t_1\leq t<t_2)},\boldsymbol{\mathcal{L}}_{n(t_2\leq t<t_3)},\cdots,\boldsymbol{\mathcal{L}}_{n(t_{p-1}\leq t<t_p)}\right\},\ \ t_p>t_{p-1}>\cdots>t_2>t_1 \quad (19)$$

Also, if each topology in sequence (18) has spanning tree, then each Laplacian matrix in (19) satisfies (7), thus the team control system is stable along the given interval. Following this, it is implied that when p approaches infinity, the state transition matrix has a limit which equals a rank-one matrix.

$$\lim_{p\to\infty}\boldsymbol{\Phi}_{(t_p,t_1)} = \lim_{p\to\infty}\prod_{k=1}^{p-1}\boldsymbol{\Phi}_{(t_{k+1},t_k)} = [1\ \ \cdots\ \ 1]^{\mathrm{T}}\cdot w^{\mathrm{T}} \quad (20)$$

$$w = [w_1\ \ w_2\ \ \cdots\ \ w_{6n}]^{\mathrm{T}},\ \ w_i\in\mathbb{R}$$

However, according to Ren & Beard (2005), if each topology in sequence (18) does not have spanning tree, formation stability is still achievable if the union of all directed graphs of the occuring topologies across some time interval has a spanning tree frequently enough as the system evolves[1]. The statement can be elaborated as the following:

Taking sequence (18) as our case, the union graph is expressed as

$$\boldsymbol{\mathcal{G}}_{n(t_1\leq t<t_p)} := \bigcup_{k=1}^{p-1}\boldsymbol{\mathcal{G}}_{n(t_k\leq t<t_{k+1})} \quad (21)$$

The corresponding adjacency matrix for (21) is

$$\mathcal{A}_{n(\bigcup \mathcal{G})} = \{a_{i,j(\bigcup \mathcal{G})}\} := \sum_{k=1}^{p-1} \mathcal{A}_{n(t_k \le t < t_{k+1})} \tag{22}$$

$$a_{i,j(\bigcup \mathcal{G})} \in \{0,1\},\ \forall j \neq i$$

Since $a_{i,j} \in \{0,1\}$ for all $j \neq i$, the addition operations contained in the sigma operator in (22) are inclusive-or operations. Therefore, $a_{i,j(\cup \mathcal{G})} \in \{0,1\}$ for all $j \neq i$. The corresponding Laplacian matrix follows using (8).

$$\mathcal{L}_{n(\bigcup \mathcal{G})} = \{l_{i,j(\bigcup \mathcal{G})}\}$$
$$l_{i,j(\bigcup \mathcal{G})} := \begin{cases} -a_{i,j(\bigcup \mathcal{G})}, & j \neq i \\ \sum_{j=1, j\neq i}^{n} a_{i,j(\bigcup \mathcal{G})}, & j = i \end{cases} \tag{23}$$

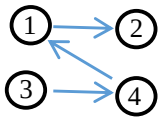
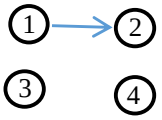
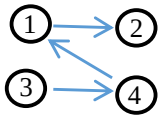
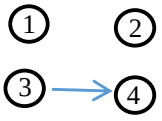
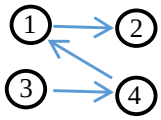
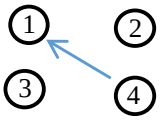
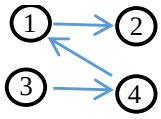
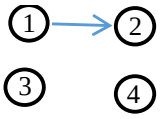
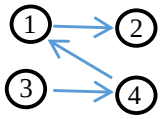
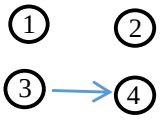
Sequence (18) will stabilize formation control system (6) if the Laplacian matrix corresponding to the union graph of all occuring graphs in sequence (18) has exactly one zero-eigenvalue and the other eigenvalues have positive real parts.

$$\lambda_{1\left(\mathcal{L}_{n(\cup \mathcal{G})}\right)} = 0$$
$$\mathrm{Re}\left(\lambda_{n\left(\mathcal{L}_{n(\cup \mathcal{G})}\right)}\right) \ge \mathrm{Re}\left(\lambda_{n-1\left(\mathcal{L}_{n(\cup \mathcal{G})}\right)}\right) \ge \cdots \ge \mathrm{Re}\left(\lambda_{2\left(\mathcal{L}_{n(\cup \mathcal{G})}\right)}\right) > \mathrm{Re}\left(\lambda_{1\left(\mathcal{L}_{n(\cup \mathcal{G})}\right)}\right) \tag{24}$$

The statement can then be proven by evaluating the evolution of the state transition matrix of the system in the state evolution equation (16). For system with switching topology by a sequence given by (18), the state transition matrix across the whole interval $[t_p, t_1)$ can be expressed as product of all state transition matrices that corresponds to the occuring topology graphs in the sequence.

$$\boldsymbol{\Phi}_{(t_p,t_1)} = \prod_{k=1}^{p-1} \boldsymbol{\Phi}_{(t_{k+1},t_k)} \tag{25}$$

Table 1 Comparison between sequence of evolving graph of: constant topology with spanning tree (Case A) and switching non-spanning tree topology (Case B)

	Case A	Case B
$t = 0$	 $L=\begin{bmatrix}1&0&0&-1\\-1&1&0&0\\0&0&0&0\\0&0&-1&1\end{bmatrix}$, $\boldsymbol{\phi}_{(0,0)}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}$	 $L=\begin{bmatrix}0&0&0&0\\-1&1&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}$, $\boldsymbol{\phi}_{(0,0)}=\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}$
$0 \le t < 1$	 $L=\begin{bmatrix}1&0&0&-1\\-1&1&0&0\\0&0&0&0\\0&0&-1&1\end{bmatrix}$, $\boldsymbol{\phi}_{(1,0)}=\begin{bmatrix}0.3679&0&0.2642&0.3679\\0.3679&0.3679&0.0803&0.1839\\0&0&1&0\\0&0&0.6321&0.3679\end{bmatrix}$	 $L=\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&-1&1\end{bmatrix}$, $\boldsymbol{\phi}_{(1,0)}=\begin{bmatrix}1&0&0&0\\0.6321&0.3679&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}$
$1 \le t < 2$	 $L=\begin{bmatrix}1&0&0&-1\\-1&1&0&0\\0&0&0&0\\0&0&-1&1\end{bmatrix}$, $\boldsymbol{\phi}_{(2,0)}=\begin{bmatrix}0.1353&0&0.5940&0.2707\\0.2707&0.1353&0.3233&0.2707\\0&0&1&0\\0&0&0.8647&0.1353\end{bmatrix}$	 $L=\begin{bmatrix}1&0&0&-1\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}$, $\boldsymbol{\phi}_{(2,0)}=\begin{bmatrix}0.1353&0&0.5940&0.2707\\0.2707&0.1353&0.3233&0.2707\\0&0&1&0\\0&0&0.8647&0.1353\end{bmatrix}$
$2 \le t < 3$	 $L=\begin{bmatrix}1&0&0&-1\\-1&1&0&0\\0&0&0&0\\0&0&-1&1\end{bmatrix}$, $\boldsymbol{\phi}_{(3,0)}=\begin{bmatrix}0.0498&0&0.8009&0.1494\\0.1494&0.0498&0.5768&0.2240\\0&0&1&0\\0&0&0.9502&0.0498\end{bmatrix}$	Sequence repeats  $L=\begin{bmatrix}0&0&0&0\\-1&1&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix}$, $\boldsymbol{\phi}_{(3,0)}=\begin{bmatrix}0.3679&0&0.3996&0.2325\\0.6321&0.3679&0&0\\0&0&1&0\\0&0&0.6321&0.3679\end{bmatrix}$
$3 \le t < 4$	 $L=\begin{bmatrix}1&0&0&-1\\-1&1&0&0\\0&0&0&0\\0&0&-1&1\end{bmatrix}$, $\boldsymbol{\phi}_{(4,0)}=\begin{bmatrix}0.0183&0&0.9084&0.0733\\0.0733&0.0183&0.7619&0.1465\\0&0&1&0\\0&0&0.9817&0.0183\end{bmatrix}$	 $L=\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&-1&1\end{bmatrix}$, $\boldsymbol{\phi}_{(4,0)}=\begin{bmatrix}0.3679&0&0.3996&0.2325\\0.4651&0.1353&0.2526&0.1470\\0&0&1&0\\0&0&0.6321&0.3679\end{bmatrix}$
⋮	⋮ ⋮	⋮ ⋮
$t \to \infty$	$L=\begin{bmatrix}1&0&0&-1\\-1&1&0&0\\0&0&0&0\\0&0&-1&1\end{bmatrix}$, $\lim_{t\to\infty}\boldsymbol{\phi}_{(t,0)}=\begin{bmatrix}0&0&1&0\\0&0&1&0\\0&0&1&0\\0&0&1&0\end{bmatrix}$	$L=\begin{bmatrix}0&0&0&0\\0&0&0&0\\0&0&0&0\\0&0&-1&1\end{bmatrix}$, $\lim_{t\to\infty}\boldsymbol{\phi}_{(t,0)}=\begin{bmatrix}0&0&1&0\\0&0&1&0\\0&0&1&0\\0&0&1&0\end{bmatrix}$

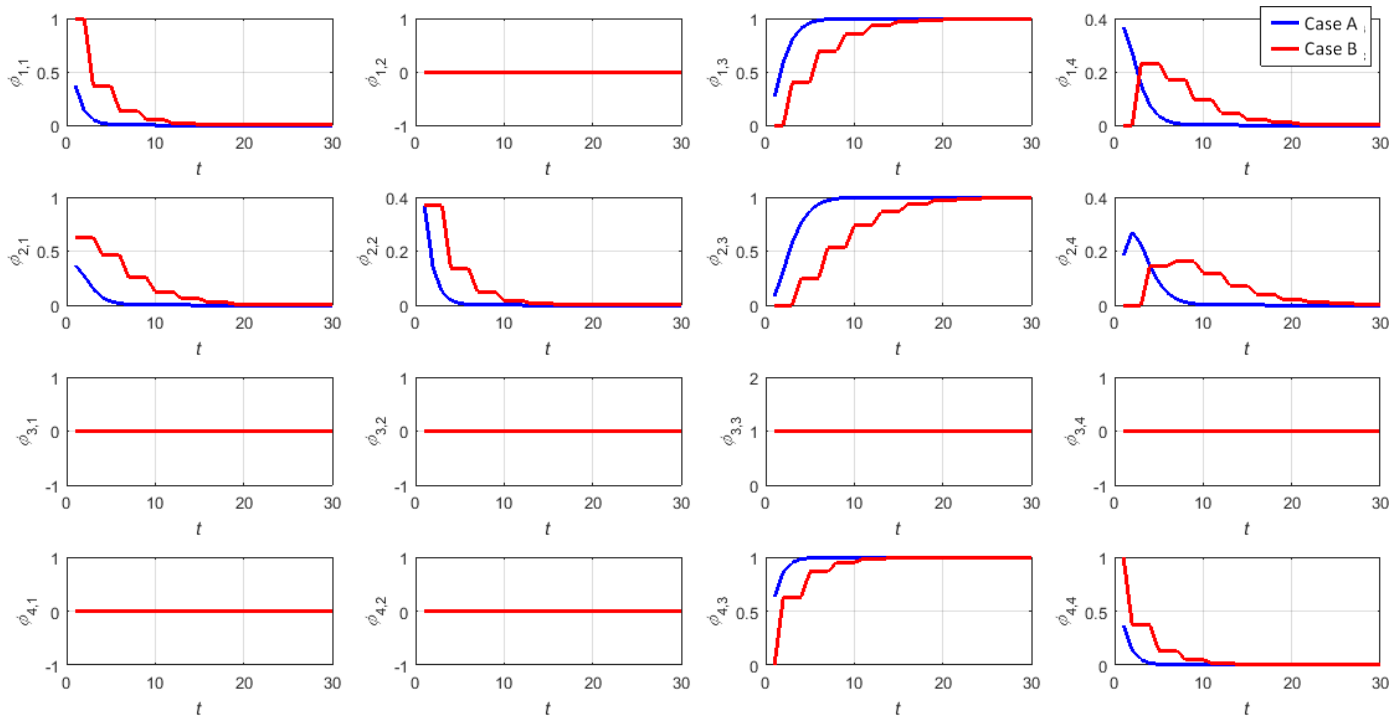


Figure 3 Comparison on state transition matrix evolution between: fixed topology with spanning tree (Case A) and switching no-spanning-tree topology (Case B)

Next, we present a comparison of two cases: Case A vs. Case B. Case A is a team of four members and has contant topology with spanning tree. Case B is also a team with the same size, but it has topology that consists of only one edge so it has no spanning tree. Which pair that the edge connects also changes from time to time, effectively changing the topology of the whole team. The sequence of the switching topology in Case B repeats every third switch. Also, the sequence is so chosen that the union graph of the occuring graph equals the topology graph in Case A. The team size used in both cases can be generalized for team with different sizes. From these two cases, their respective state transition matrices are calculated and evaluated. Comparison data is presented in Table 1 up to several switch intants. Comparison for the extended time interval is presented in Figure 3 where it can be seen that the transition matrix in both cases converges to a rank-one matrix whose rows are identical ([0 0 1 0]). The difference between Case A and Case B is that convergence occur at a longer time in Case B. Therefore we can reformulate (20) for sequences of non-spanning tree (NST) graphs:

Supposing all the occuring graph in sequence (20) are NST graphs, and supposing sequence (20) is periodic with p is its period. Then, the graph union in (21) is also periodic with period p. System state evolution during each period can then be written as

$$\vec{\xi}_{(t_p)} = \boldsymbol{\Phi}_{(t_p,t_1)} \cdot \vec{\xi}_{(t_1)} = \left(\prod_{k=1}^{p-1} \boldsymbol{\Phi}_{(t_{k+1},t_k)} \right) \cdot \vec{\xi}_{(t_1)} \tag{26}$$

If sequence (20) occurs repetitively for q times, then

$$\vec{\xi}_{(t_{qp})} = \boldsymbol{\Phi}_{(t_{qp},t_1)} \cdot \vec{\xi}_{(t_1)} = \boldsymbol{\Phi}^{q}_{(t_p,t_1)} \cdot \vec{\xi}_{(t_1)} = \left(\prod_{k=1}^{p-1} \boldsymbol{\Phi}_{(t_{k+1},t_k)} \right)^{q} \cdot \vec{\xi}_{(t_1)} \tag{27}$$

State evolution (27) will converge to a limit value if a rank-one matrix (with appropriate size) is the limit value of $\boldsymbol{\Phi}^{q}_{(t_p,t_1)}$ as $q \to \infty$. This requirement is satisfied if the union graph $\cup \mathcal{G}$ that $\boldsymbol{\Phi}_{(t_p,t_1)}$ corresponds to has spanning tree.

$$\left.\begin{aligned} & \lambda_1\left(\mathcal{L}_{\mathcal{G}\left(\boldsymbol{\Phi}_{(t_p,t_1)}\right)}\right) = 0 \\ & \mathrm{Re}\left(\lambda_n\left(\mathcal{L}_{\mathcal{G}\left(\boldsymbol{\Phi}_{(t_p,t_1)}\right)}\right)\right) \geq \mathrm{Re}\left(\lambda_{n-1}\left(\mathcal{L}_{\mathcal{G}\left(\boldsymbol{\Phi}_{(t_p,t_1)}\right)}\right)\right) \geq \cdots \geq \mathrm{Re}\left(\lambda_2\left(\mathcal{L}_{\mathcal{G}\left(\boldsymbol{\Phi}_{(t_p,t_1)}\right)}\right)\right) > \mathrm{Re}\left(\lambda_1\left(\mathcal{L}_{\mathcal{G}\left(\boldsymbol{\Phi}_{(t_p,t_1)}\right)}\right)\right) \end{aligned}\right\} \tag{28}$$

C. The aggregated control system

For a team of UAV that is performing FFK task using consensus-based formatioin flight control system (6), an incoming collision threat can occur in two possible scenarios. The first scenario is the threat is imposed on the individual that is the root of the spanning tree, and the second scenario is where the threat is imposed on any other individual than the root.

1) Collision threat imposing the root individual

Let us now consider a team of four UAVs. While performing FFK task, an Intruder (a foreign UAV who does not belong to the team) approaches and imposes UAV V1 who is the root of spanning tree ST1. Before UAV V1 begins an attempt to avoid the Intruder, V1 must first acquire Intruder's relative position and relative velocity to itself, *i.e.*, there must be a form of information flow taking place from the Intruder to V1. The information flow between Intruder and V1 is not different to those between team members, with the exception that the information flow is for CA purpose, not FFK. Hence, the initial

team topology graph representation $\mathcal{G}_4$ consisting four vertices representing the four team members with V1 as the root of the spanning tree ST1 (Figure 4(a)) gets extended to include the Intruder. The extended graph $\mathcal{G}_{4+1}$ consists of four team member vertices plus one Intruder vertex with the Intruder as the root of the extended spanning tree extST1 (Figure 4(b)). With $\mathcal{G}_{4+1}$, V1 can perform the CA task while the rest of the team (V2, V3, V4) maintain the team topology $\mathcal{G}_4$ as is to perform FFK task. The corresponding Laplacian matrices for these graphs are expressed in (29).

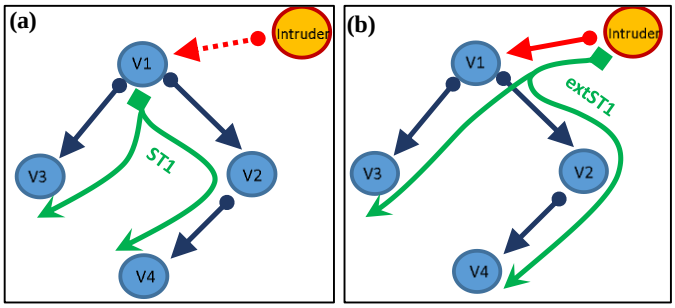


Figure 4 Collision threat on root individual:
(a) Graph representation which excludes Intruder,
(b) Extended graph representation which includes Intruder

$$\mathcal{L}_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}, \quad \mathcal{L}_{4+1} = \left[\begin{array}{cccc:c} 1 & 0 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ \hdashline 0 & 0 & 0 & 0 & 0 \end{array}\right] \tag{29}$$

The first row of $\mathcal{L}_4$ corresponds to V1. Since V1 is the root of a (spanning) tree, all entries in that row are zeros. The (4+1)$^{\text{th}}$ row of $\mathcal{L}_{4+1}$ corresponds to Intruder. All entries of it are also zeros since Intruder is the root of a (spanning) tree.

The FFKCA controller is formulated as the following.

$$\vec{u}_i = \vec{u}_{i\text{FFK}} + \vec{u}_{i\text{CA}} \tag{30}$$

The FFK part is given by (6)

$$\left.\begin{aligned} \dot{\vec{r}}_{\text{F}i,\text{V}i(t)} &= \vec{v}_{\text{F}i,\text{V}i(t)} \\ \dot{\vec{v}}_{\text{F}i,\text{V}i(t)} &= -a_{i,j_{\text{G}}(t)} \cdot \left(k_{\text{tr},\vec{r}} \cdot \left(\vec{r}_{\text{F}i,\text{V}i(t)} - \vec{r}_{\text{G}0(t)} \right) + k_{\text{tr},\vec{v}} \cdot \left(\vec{v}_{\text{F}i,\text{V}i(t)} - \vec{v}_{\text{G}0(t)} \right) \right) \\ &\quad - \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{i,j(t)} \cdot \left(\gamma_{\vec{r}} \cdot \left(\vec{r}_{\text{F}i,\text{V}i(t)} - \vec{r}_{\text{F}j,\text{V}j(t)} \right) + \gamma_{\vec{v}} \cdot \left(\vec{v}_{\text{F}i,\text{V}i(t)} - \vec{v}_{\text{F}j,\text{V}j(t)} \right) \right) \end{aligned}\right\} \tag{6}$$

$$\{a_{i,j}\} \in \mathcal{A}_{\text{span}} \tag{31}$$

$$a_{i,j} = 0, \text{ for } i = i_{\text{root}}, \ \forall j \neq i \tag{32}$$

with two requirements described in (31) and (32): all $a_{i,j}$ must belong to adjacency matrix $\mathcal{A}_{\text{span}}$ that corresponds to topology with spanning tree, and all $a_{i,j}$ in the row that corresponds to the root vertex have values equal zero.

2) Collsion threat imposing non-root individual

Now we consider a scenario where the Intruder imposes collision threat on V2. Initially, the team topology graph $\mathcal{G}_4$ is similar to that of the previous scenario: it has a spanning tree, with V1 as the root (Figure 5(a)). When V2 attempts to avoid the Intruder, it creates an information flow from Intruder to V2. The extended graph $\mathcal{G}_{4+1}$ now consists of two NSTs (Figure 5(b)). The first tree (NST1) spans on V1 as its root, on V2, on V3, and on V4, with no path for information from Intruder to reach any of them. The second tree (NST2) spans on Intruder as its root, on V2, and on V4, with no path for information from Intruder to reach V1 and V2. Furthermore, the two NSTs in $\mathcal{G}_{4+1}$ converge on V2 and span to V4 with the same path. This means there are always conflicting reference trajectories for V2 to follow: trajectory to avoid Intruder vs. trajectory to track V1. Whenever these conflicting trajectories allign with each other, a local minima effect occurs causing an undesired dangereous behaviour to emerge on V2 and all vertices on its branch (in this case: V4), which becomes a problem since the team cannot perform FFK task safely. This problem can be identified by evaluating the corresponding $\mathcal{L}_{4+1}$ of $\mathcal{G}_{4+1}$ expressed in (33).

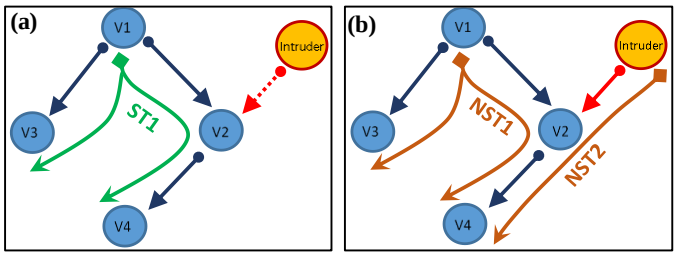


Figure 5 Collision threat on a non-root individual:
(a) Graph representation which excludes Intruder,
(b) Extended graph representation which includes Intruder

$$\mathcal{L}_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}, \quad \mathcal{L}_{4+1} = \left[\begin{array}{cccc:c} 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & 0 & 0 & -1 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ \hdashline 0 & 0 & 0 & 0 & 0 \end{array}\right] \tag{33}$$

In (33), the conflicting reference trajectories are indicated by the presence of more than one zero-row in the Laplacian matrix. There are two zero-rows in matrix $\mathcal{L}_{4+1}$. The first zero-row is the first row, which corresponds to V1. The zero entries in this row indicate that V1 is a root of a tree (NST1). The other zero-row is the last row, which corresponds to the Intruder as the root of NST2.

To solve this problem, the team topology graph $\mathcal{G}_4$ must be changed/switched to a topology $\mathcal{G}_4^*$ that facilitates the existance of a spanning tree in its extended topology $\mathcal{G}_{4+1}^*$. In our case, we can achieve this by first identifying the conflicting path in $\mathcal{G}_{4+1}$, that is the path that is shared by two or more NSTs, and eliminate the path sharing. From Figure 5(b), path V2→V4 is shared by NST1 and NST2. Path sharing elimination can be done either by eliminating edge V1→V2 or edge Intruder→V2. Edge Intruder→V2 is essential for V2 to perform CA task, so edge V1→V2 gets eliminated. By eliminating edge V1→V2, we now have two separated NSTs (Figure 6(a)): NST1* and NST2.

Next, we need to connect one NST to the other so that both NSTs become one (spanning) tree, which spans on all vertices with Intruder vertex as the root. Figure 6(b) illustrates three

possible edges (connections) to connect NST1* to NST2 in our case: edge Intruder→V1, edge V2→V1, and edge V4→V1. Selecting any one of the edge candidates, or any two, or even all three of the edge candidates will result in the creation of at least one new spanning tree. However, in this research, there are at least two constraints that limits our selection option.

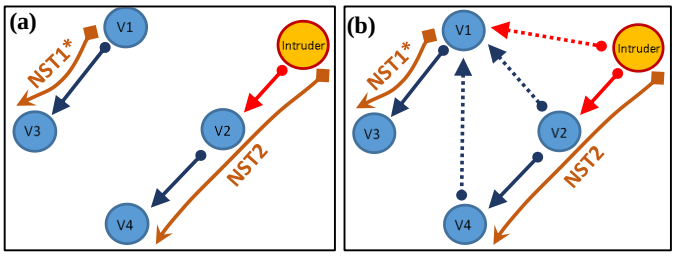


Figure 6 Extended graph after edge V1→V2 is eliminated (a), and edge candidates that will connect NST1* to NST2 (b)

The first constraint is the necessity in using minimum number of connections, so we assume the use of the least number of connection policy, which for our current case, is one connection. The other constraint is the objective contraint, that is, the FFK task. We cannot select edge Intruder→V1 since it will divide the team into two separate teams, topologically, which will cause the FFK task cannot be carried out with the current team size. Satisfying these two constraints leaves us edge V2→V1, and edge V4→V1.

We select edge V2→V1 for the new connection that connects NST1* to NST2. The consideration for this selection is that 'planting' NST1* on V2 create the shortest path from the root of NST2 (Intruder) to the root of NST1*. Shorter path means information takes less time to arrive on all vertices. The resulting extended graph $\mathcal{G}_{4+1}^{*}$ of the new topology is presented in Figure 7(a). Here, we can see a similarity between $\mathcal{G}_{4+1}^{*}$ in this scenario with $\mathcal{G}_{4+1}$ of the previous scenario: on both of them, collision threat is imposed on the root of the spanning tree. Corresponding Laplacian matrices are expressed by (34).

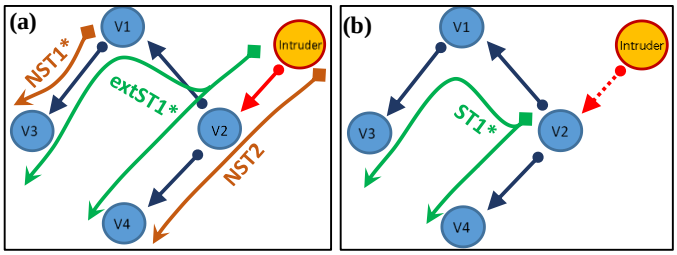


Figure 7 The final stage of the topology switching: Spanning tree is re-established in the extended graph $\mathcal{G}_{4+1}^{*}$ (a), spanning tree is re-established in the graph $\mathcal{G}_{4}^{*}$ (b)

$$\mathcal{L}_4 = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}, \quad \mathcal{L}_{4+1} = \left[\begin{array}{cccc:c} 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ \hdashline 0 & 0 & 0 & 0 & 0 \end{array}\right] \tag{34}$$

The process of switching topology when collision threat imposes a non-root individual can be summarized as the followings.

1. Identifying which individual is the most threatened by an Intruder's presence. The state of being the most threatened can be determined using CTS as the metric. CTS for each team member is calculated using (1).
2. Individual with the highest value of CTS is given exemption from doing any tracking task required by the current team topology. Therefore,

$$\text{for } j = 1,\ 2,\ \cdots,\ n: \quad a_{i,j(t_k)} = \begin{cases} 0, & \text{if} \quad i = i_{CTS\max} \\ a_{i,j(t_k-1)}, & \text{if} \quad i \neq i_{CTS\max} \end{cases} \tag{35}$$

3. And for individual that is formerly the root of the previous spanning tree, must be assigned to track the individual that is the root of the current spanning tree

$$\text{for } \forall i \neq i_{CTS\max}: \quad a_{i,i_{CTS\max}(t_k)} = \begin{cases} 1, & \text{if} \quad \left[a_{i,1(t_k-1)} \ \ a_{i,2(t_k-1)} \ \cdots \ a_{i,n(t_k-1)}\right] = [0\ 0\ \cdots\ 0] \\ a_{i,j(t_k-1)}, & \text{if} \quad \left[a_{i,1(t_k-1)} \ \ a_{i,2(t_k-1)} \ \cdots \ a_{i,n(t_k-1)}\right] \neq [0\ 0\ \cdots\ 0] \end{cases} \tag{36}$$

III. SIMULATION

A. *CA task*

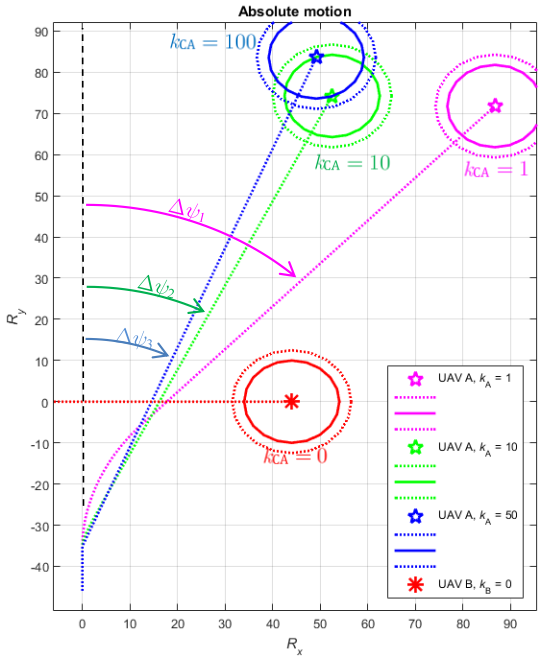


Figure 8 Trajectory deflection comparison

We test our controller design by conducting simulations. Simulation on CA controller is performed at several controller gain (k_{CA}) value: $k_{CA} = 1$, $k_{CA} = 10$, and $k_{CA} = 50$. The simulated scenario is a side encounter with a non-cooperative Intruder where both UAV ego (UAV A) and Intruder (UAV B) start from positions and velocities that lead to a collision. Results from this simulation shows that higher gain value means better performance in doing CA task. Performance is measured using

how much the original trajectory is deflected $\Delta\Psi$ (Figure 8). High gain value means small $\Delta\Psi$, good the performance. Also, high gain value affects how soon the CA system initiate avoiding maneuver: the higher the gain, the sooner the maneuver is initiated (Figure 9). More results on this simulation is published in [10].

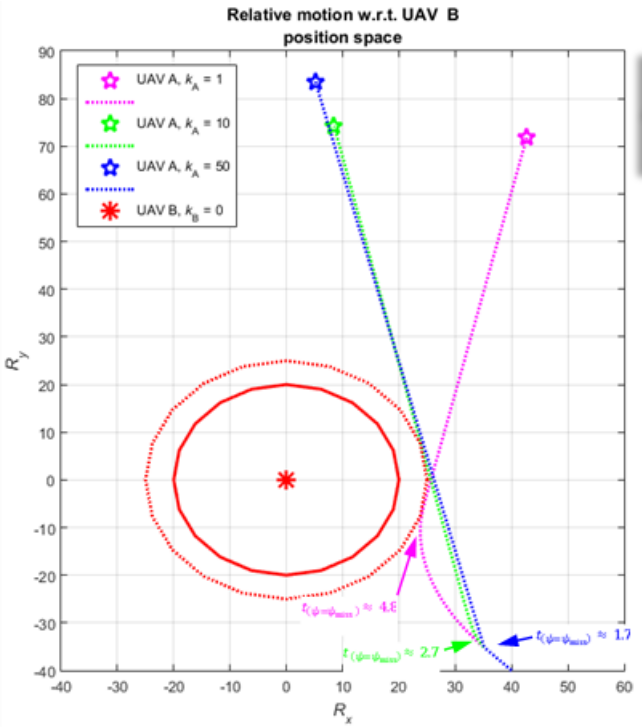


Figure 9 Trajectory comparison in relative position space

B. *FFK task*

The simulation on FFK controller is performed to demonstrate the the capability of the controller to perform formation forming task. In this simulation, each UAV starts from different location before starting to fly to form a certain V-shape formation. Trajectories are plotted in Figure 10.

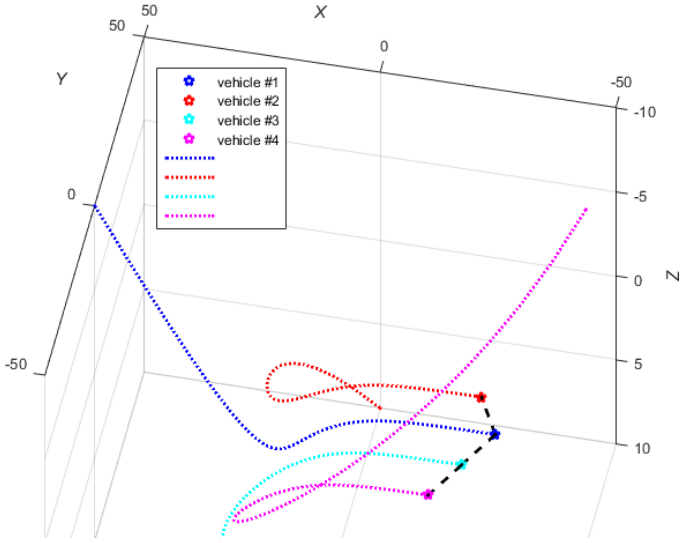


Figure 10 Trajectories when formation forming task is executed

C. *FFKCA task*

Test on FFKCA controller is conducted by two simulations (simulation A and simulation B) of the same scenario. The two simulation differ to each other regarding the topology they use. The team in the first simulation uses a topology with one spanning tree (Figure 11(a)), while in the second simulation the team uses two spanning trees in its topology (Figure 11(b)).

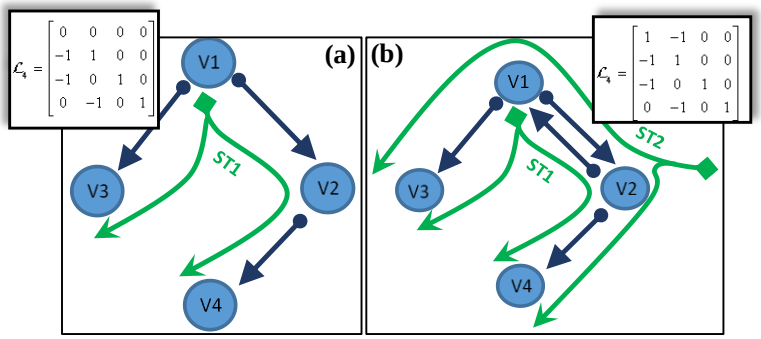


Figure 11 1-spanning tree topology in simulation A (a), 2-spanning tree topology in simulation B (b)

The two simulations run the following scenario:

1. Initially a team of 4 UAV is flying in a diamond formation. Figure describes the formation shape and the position of each individual UAV in that formation (Figure 12(a)). One Intruder is also placed not far from the team.
2. The intruder moves approaching UAV2's position (Figure 12 (b)). This situation is designed to force a collision threat situation where UAV2 gets the highest threat.
3. When the Intruder gets closer to UAV2's position, it changes its trajectory towards UAV4' position (Figure 12 (c)). By this situation, the highest collision threat is shifted from UAV2 to UAV4 while the team is still recovering its formation. This situation is designed to force the team to enter another collision threat situation while still being in transient condition. And since there is no path from UAV4 to the rest of the team, this situation also force the team to operate with no spanning tree in its topology.
4. The intruder maintain its trajectory direction during its approach to UAV4's postion, and leave the team's vicinity (Figure 12 (d)).

Simulation results are hown in Figure 13 to Figure 23.

As can be seen in Figure 14 when the Intruder approaches UAV2, UAV2 begin reacting by stop tracking UAV1 and commit an evasive maneuver to reduce the imposed collision threat by the Intruder. All other UAVs also react but more weakly. As UAV2's evades away from its post causing it to break the formation, other member that tracks UAV2's state according to team topology begin their respective maneuver to compensate.

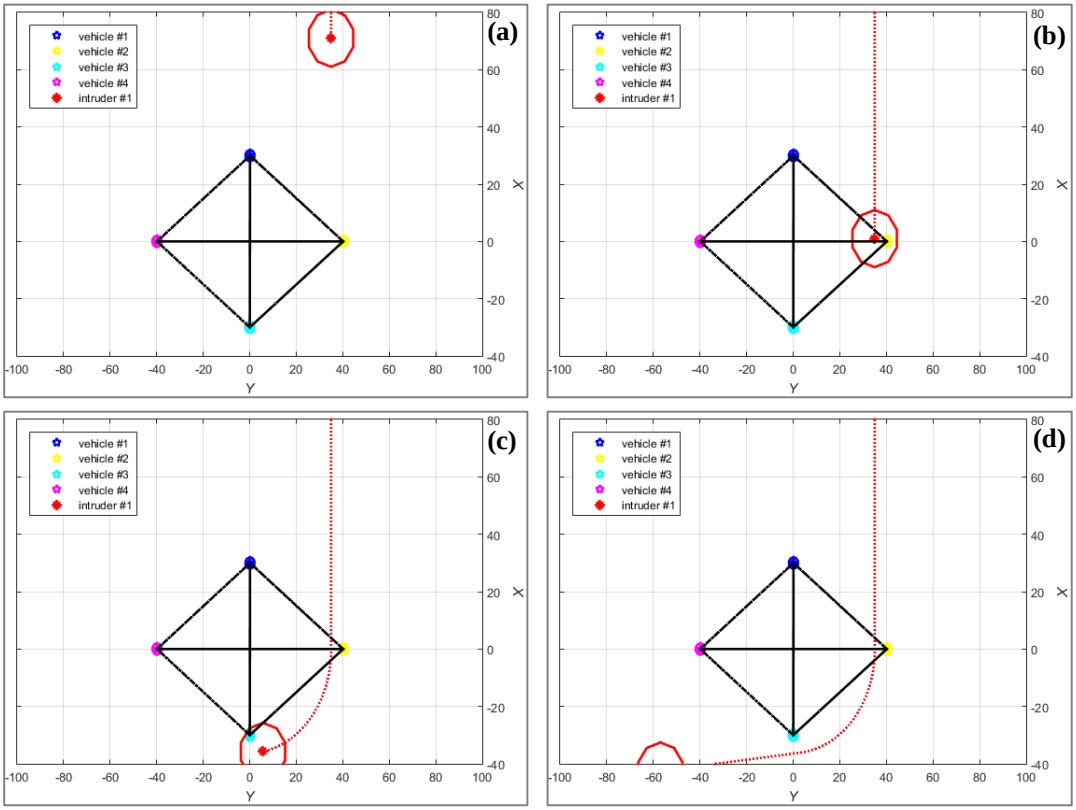
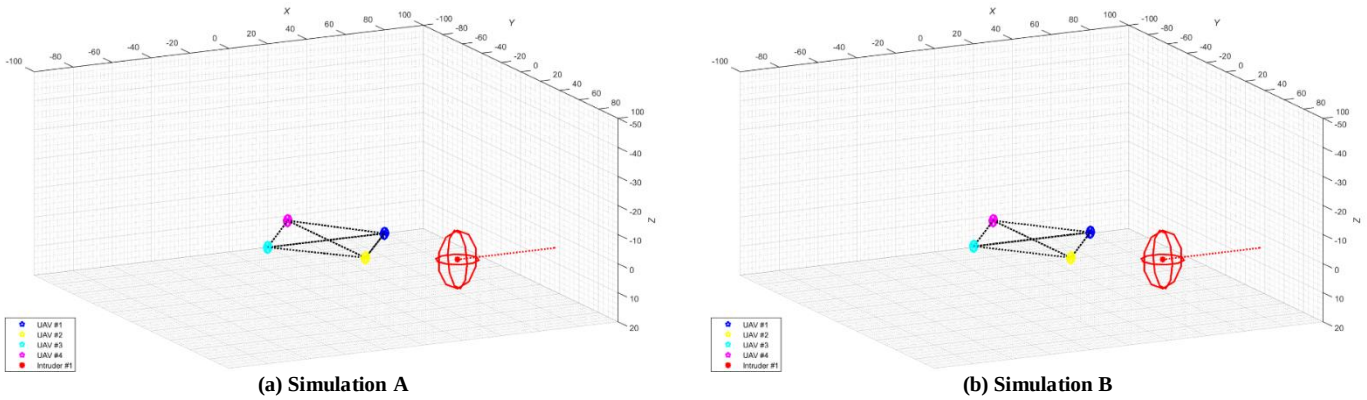
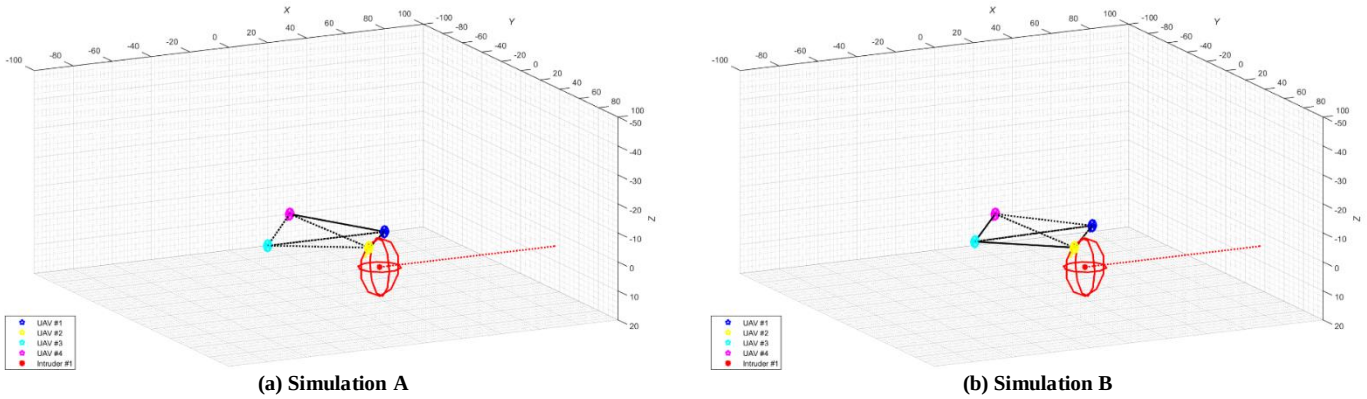


Figure 12 Encounter scenario for FFKCA test



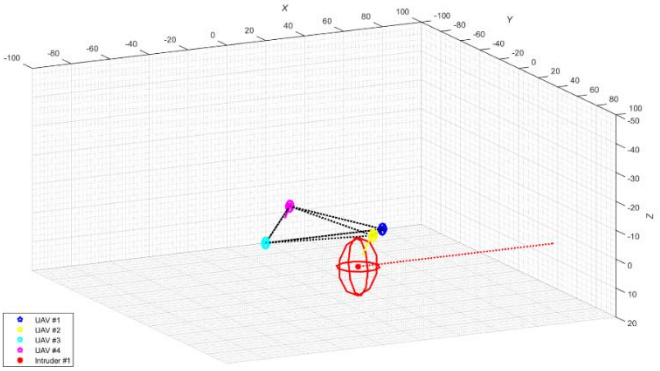
(a) Simulation A **(b) Simulation B**

Figure 13 $k = 51$

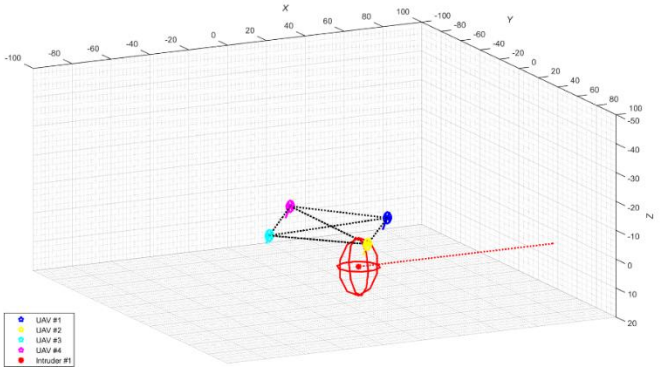


(a) Simulation A **(b) Simulation B**

Figure 14 $k = 91$

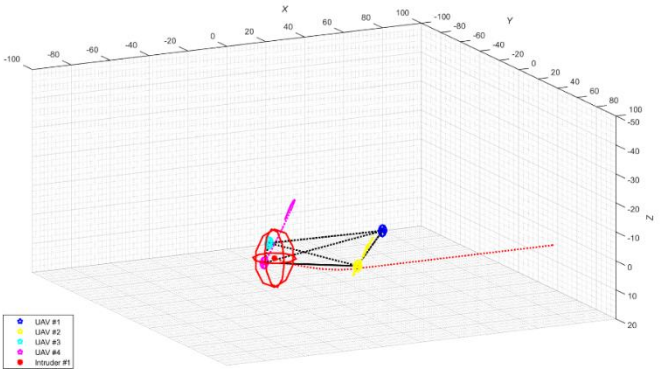


(a) Simulation A

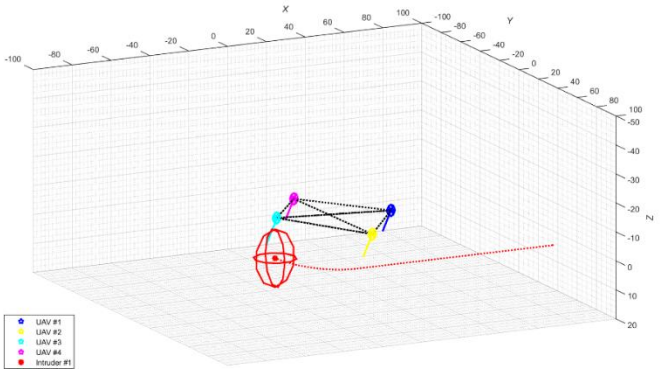


(b) Simulation B

Figure 15 $k = 101$

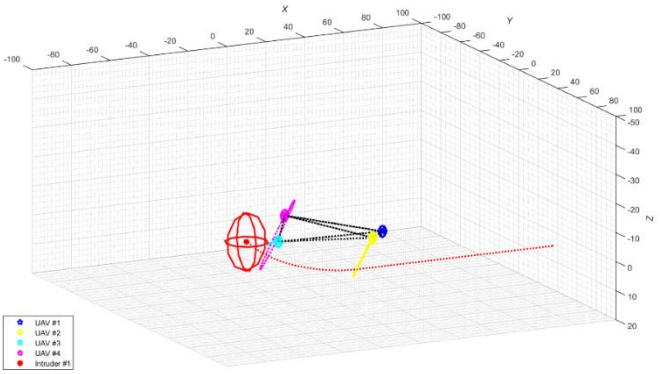


(a) Simulation A

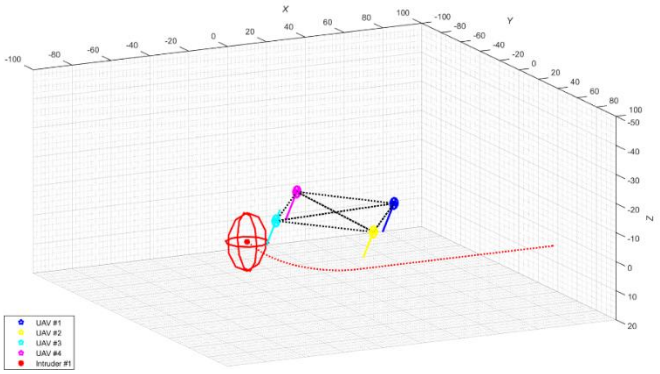


(b) Simulation B

Figure 16 $k = 141$

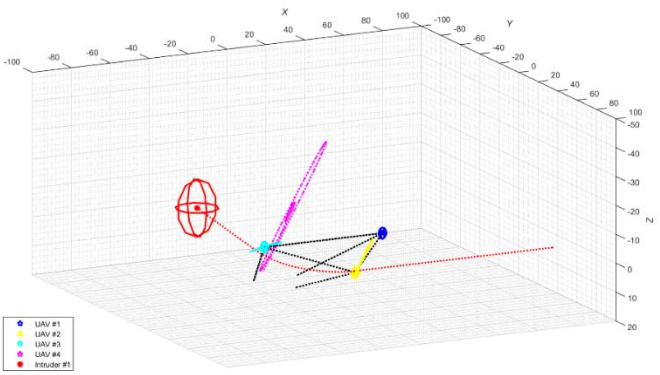


(a) Simulation A

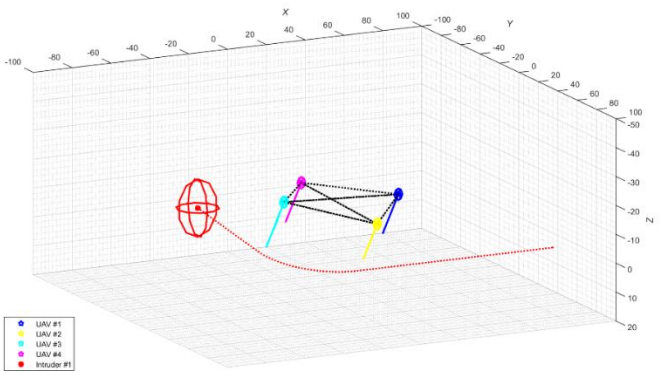


(b) Simulation B

Figure 17 $k = 161$

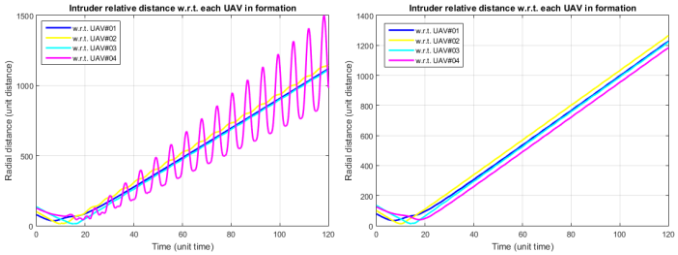


(a) Simulation A



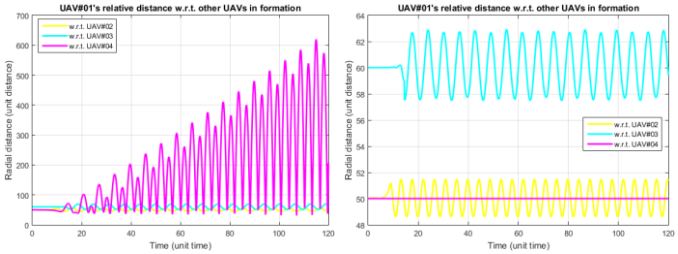
(b) Simulation B

Figure 18 $k = 201$



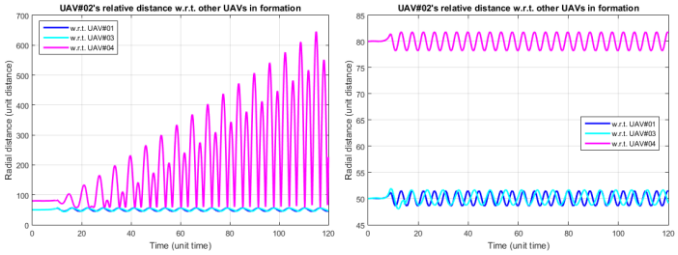
(a) Simulation A **(b) Simulation B**

Figure 19 Comparison on UAV's relative distance to Intruder



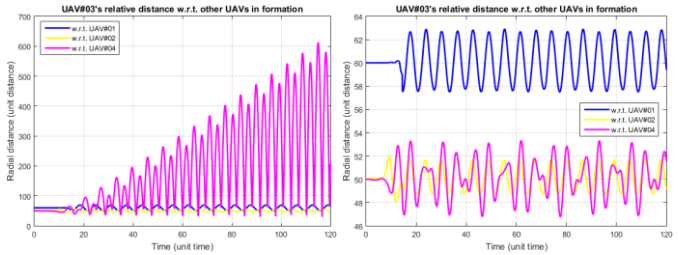
(a) Simulation A **(b) Simulation B**

Figure 20 Comparison on UAV1's relative distance to other team member



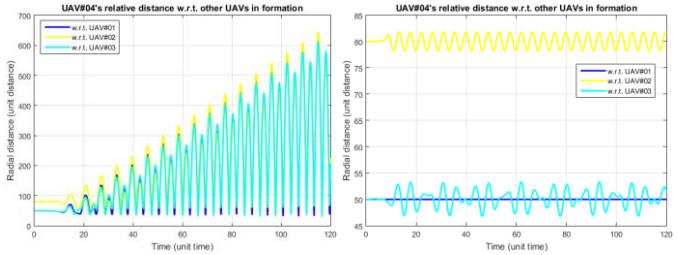
(a) Simulation A **(b) Simulation B**

Figure 21 Comparison on UAV2's relative distance to other team member



(a) Simulation A **(b) Simulation B**

Figure 22 Comparison on UAV3's relative distance to other team member



(a) Simulation A **(b) Simulation B**

Figure 23 Comparison on UAV4's relative distance to other team member

In Simulation A, only UAV4 tracks UAV2. Since no information regarding UAV2's breakaway from the formation cannot reach UAV1 and UAV3, any effort to compensate for UAV2's breakaway to maintain formation geometry occurs in an uncoordinated manner. Team formation geometry begins to deform as the result (Figure 15(a)). The deformation gets worse as the Intruder turn its trajectory to UAV4's position (Figure 18(a)). The animated plot during the entire simulation time is presented in a supplementary file titled "Animated Plot of Simulation A (Figure 13 – Figure 18)".

Manwhile in Simulation B, beside UAV4, UAV1 also tracks UAV2,.which being tracked by UAV3. This allows information regarding UAV2's breakaway to eventually reach all team member. Effort to compensate for UAV2's breakaway occurs in a coordinated manner. Team formation drifts away from its initial position, but its geometry (shape) is maintained (Figure 15(b)). When Intruder's trajectory turns towards UAV4, the effect is minimal since the team has already moving away from the Intruder (Figure 18(b)). The animated plot during the entire simulation time is presented in a supplementary file titled "Animated Plot of Simulation B (Figure 13 – Figure 18)".

IV. Conclusion

The consensus algorithm offers a control paradigm for a team of UAV system that facilitates the effort to combine two opposing control solution (FFK control vs. CA control) through the manipulation of communication topology of the team. By manipulating the topology, we ensure the resulting CA control solution never gets opposed by the resulting FFK control solution, eliminating the possibility for the team to be trapped in local minima situation.

The CA control part is designed using kinematic modeling approach where collission threat is formulated as a function of radial distance, radial velocity, and angular distance of the threat source to UAV ego.

The agregation of CA control into the FFK control is essentially a switching algorithm on the team's topology that is based on a proven lemma about the stability of consensus-based controller whose topology is changing discretely (switching). The algorithm is tested on two simulations, each demonstrating the FFKCA control performance with different communication topology. The test shows satisfactory result and may serve as a proof of consept on our agregation scheme for FFKCA control.

References

[1] Ren, Wei; Beard, Randal W.; "Consensus Seeking in Multiagent Systems under Dynamically Changing Interaction Topologies", IEEE Transactions on Automatic Control vol. 50(5), 2005, pp.655–661.

[2] Paul, Tobias; Krogstad, Thomas R.; Gravdahl, Jan Tommy; "Modelling of UAV Formation Flight using 3D Potential Field", Simulation Modelling Practice and Theory, vol.16, 2008, pp.1453-1462.

[3] Mastellone, Silvia; Stipanovic, Dusan M.; Graunke, Christopher R.; Intlekofer, Koji A.; Spong, Mark W.; "Formation Control and Collision Avoidance for Multi-Agent Non-Holonomic Systems: Theory and Experiments", The International Journal of Robotics Research, 2008.

[4] Listmann, Kim D.; Masalawala, Mohanish V.; Adamy, Jurgen; "Consensus for Formation Control of Nonholonomic Mobile Robots", 2009 IEEE International Conference on Robotics and Automation, Kobe, Japan.

[5] Wang, Jianan; Xin, Ming; "Multi-agent Consensus Algorithm with Obstacle Avoidance via Optimal Control Approach", 2011 American Control Conference, San Fransisco, USA.

[6] Kuriki, Yasuhiro; Namerikawa, Toru; "Consensus-based Cooperative Formation Control with Collision Avoidance for A Multi-UAV System", American Control Conference, 2014.

[7] Dong, Xiwang; Zhou, Yan; Ren, Zhang; Zhong, Yisheng; "Time-Varying Formation Control for Unmanned Aerial Vehicles with Switching Interaction Topologies", Control Engineering Practice, vol. 46, 2016, pp.26-36.

[8] Mondal, Arindam; Behera, Laxmidhar; Sahoo, Soumya R.; Shukla, Anupam; "A Novel Multi-Agent Formation Control Law with Collision Avoidance", IEEE/CAA Journal of Automatica Sinica, vol. 4(3), 2017, pp.558-568. DOI: 10.1109/JAS.2017.7510565

[9] Wen, Guoxing; Chen, C.L. Philip; Liu, Yan-Jun; "Formation Control with Obstacle Avoidance for A Class of Stochastic Multi-Agent Systems", IEEE Transactions on Industrial Electronics, 2017. DOI: 10.1109/TIE.2017.2782229

[10] Sudiyanto, Tata; Trilaksono, Bambang R.; Budiyono, Agus; Adiprawita, Widyawardana; "Three-Dimensional Collision Avoidance Control for UAVs using Kinematic-based Collision Threat Situation Modeling Approach", International Journal on Electrical Engineering and Informatics, vol.10(3), 2018, pp.542-579.

[11] Lou, Gewei; Yang, Wenjing: "Formation Control and Collision Avoidance of Multi-Agent System with Switching Communication Topology", CSAE, 2018. DOI: 10.1145/3207677.3278088

[12] Choutri, Kheireddine; Lagha, Mohand; Dala, Laurent; "Distributed Obstacle Avoidance for UAVs Formation using Consensus-based Switching Topology", International Journal of Computing and Digital Systems, vol. 8(2), 2019. pp.167-178. DOI: 10.12785/ijcds/080208