Time varying Formation Control Using Differential Game Approach

Shamsi F.*, Abdollahi F.*, Nikravesh K.Y.*

*Department of Electrical Engineering

Amirkabir University of Technology, Tehran, Iran

e-mail: f.shamsi@aut.ac.ir, f_abollahi@aut.ac.ir , nikrvash@aut.ac.ir

Abstract: This paper deals with time varying formation control of mobile robots i.e. the formation can change during the mission. To achieve this goal differential game approach is proposed to control the formation. A time varying incidence matrix is defined to determine the interconnection between robots. An open loop Nash equilibrium solution is employed to satisfy the defined cost function for each robot (the players in differential game) and control the formation. The convergence of the proposed approach to the desired formation is studied. By minor modifications, the introduced approach can be applied for a wide range of fully decentralized to leader follower formation. The effectiveness of the proposed method is verified by simulations.

Keywords: time varying formation, incidence matrix, mobile robots, differential game, open loop Nash equilibrium

1. INTRODUCTION

In the last few years, the problem of formation control in multi agent systems has received much attentions. Applications of formation control are in many areas such as robotics, satellites, spacecrafts, underwater vehicles, automated highway systems and so on. In many applications, using groups of agents rather than a single agent is an urgent to improve redundancy, sensitivity and robustness or to deal with the dangerous or distributed tasks. For instance they can be employed for surveillance, assist human in dangerous environments, exploration and so on.


In the leader follower approach, an agent is considered to be the leader and moves along a specified trajectory. Other agents have to keep the formation by maintaining a desired distance from the leader. The advantage of this method is that it is very easy to implement, however its reliability is poor because there is over reliance on leader agent and no feedback is considered from followers to leader.

The behavior based method integrates several behaviors and assign them to individual agents. It is applicable for a large group of agents. Disadvantage of this approach is its complexity.

Finally, in virtual structure approach, a virtual leader is considered to represent the desired position to agents to keep their formation. This method is not applicable to formations which are time varying. The advantage of this method is its robustness to perturbations on agents.

Dynamic game theory is a useful method to model dynamics among multiple agents (Ba¸sar, T. and Olsder, G.J. 1998). Tomlin et al. (1998) used dynamic game theory to air traffic management. Gu (2008) presented a differential game approach to control formation of mobile robots. In this research, formation control of mobile robots with double integrator dynamics and linear quadratic cost function were modelled as a linear quadratic Nash differential game. In this research fixed formation is considered. This research is the base of our work in this paper.

Most research on formation control assumed that the shape of formation is fixed. However, in some applications, the structure of formation should change due to some constraints such as obstacles in the environment. For example, assume a team of robots in a triangle formation. During their mission they face with some obstacles which make their path too narrow to keep the formation and the formation should be changed to enable them to traverse in this path. When the formation changes, the interconnections and neighbourhood of robots will also change. Some robots may have new neighbours and some of them may lose their neighbours. In this research, an incidence matrix is considered to represent the interconnection between robots. Changing formation causes some variations in the elements of this matrix. To show these variations, an exponential function is used to avoid discontinuity.

In our research, double integrator dynamics is used to model the robots. After determination of time varying formation structure, a cost function is defined for each robot to be minimized. Using open loop Nash equilibrium concept (Engwerda 2005), a controller for each robot is proposed to keep the formation and track a predefined trajectory. The
convergence of the proposed controller is studied in provided theorems. The paper is organized as follow. In Section 2, the problem of time varying formation is formulated. Section 3 proposes the control for the defined problem. Section 4 provides an illustrative example to evaluate the performance of the proposed approach and the Section 5 concludes the paper and gives the direction of future works.

2. PROBLEM FORMULATION

Consider a team of $m$ mobile robots with double integrator dynamics. The state equation of each robot with $n$-dimensional coordinates is defined as follow

$$\dot{x}_i = ax_i + bu_i$$

where $x_i = [q_i^T, \dot{q}_i^T]^T$, $a = \begin{bmatrix} 0 & I_n \\ 0 & 0 \end{bmatrix}$, $b = \begin{bmatrix} 0 \\ I_n \end{bmatrix}$, $u_i \in \mathbb{R}^n$ is the control signal. The dynamics of all team can be defined as

$$\dot{x} = Ax + \sum_{i=1}^{m} B_i u_i$$

where $x = [x_1^T, \ldots, x_m^T]^T$, $A = a \otimes I_m$, $B_i = [0, \ldots, 1, \ldots, 0]^T \otimes b$. It is assumed that the $u_i$ belongs to a compact and convex subset of $\mathbb{R}^n$ that contains the origin.

In this paper, each robot is considered as a player and each player has a goal. Keeping formation and tracking a specified trajectory are the players’ goals. In differential game theory, players’ objectives are formulated as linear quadratic cost functions that should be minimized. For each player, the value of cost function depends on other players’ decisions.

To describe the cost functions of players, first we should define the structure of formation. A common way for defining the structure of formation is using graph theory in which a directed graph is used to represent the interconnection (edges) between vertices (robots). The incidence matrix $D$ of a directed graph $\mathcal{G}$ is a matrix whose elements are $1, -1,$ or $0$. Its rows and columns are indexed by vertices of $v$ and edges of $e$, respectively. The $w_{th}$ entry of $D$ is equal to $1$ if the vertex $v$ is the head of the edge and in the neighbour of $v$, $-1$ if the vertex $u$ is the tail of the edge and in the neighbour of $v$, and $0$, otherwise. If graph $\mathcal{G}$ has $m$ vertices and $e$ edges, then incidence matrix $D$ of the graph $\mathcal{G}$ has order $m \times e$ (Gu 2008, Godsil and Royle 2001). It is assumed that the graph $\mathcal{G}$ is connected to ensure that the formation is keeping.

The main contribution of this paper is to obtain a time varying formation structure. When formation changes the neighbourhoods of robots will also change and changing the neighbourhoods of robots, in turn will cause varying some of the elements of incidence matrix $D$ from $\pm 1$ to $0$ and vice versa. Assume the formation is changed after $t_1$ seconds from the time that formation is started. To avoid discontinuity, the corresponding elements of incidence matrix that should change due to changing the formation are defined by sigmoid function as follows

$$\pm 1 \rightarrow 0 : g(t) = \frac{\pm 1}{1 + e^{a(t-t_1)}}$$

where $a$ is a positive number that determines the slope of changing the element.

Time varying formation control with a known time of changing formation, is applicable in situations where the robots’ path is known or the obstacles are fixed and known. The formation can be defined by keeping a specified distance between positions and velocities of robots that are adjacent $(d_{ij}^d = x_i^d - x_j^d)$. Considering this assumption, the cost function of each player (robot) includes a term that indicates the distance error between the states of each robot and its neighbours and the desired distances. The formation error for $i$th robot can be defined as bellow

$$\sum_{i,j \in e} w_{ij}(t)\|x_i(t) - x_j(t) - d_{ij}^d\|^2$$

where the robot $j$ is in vicinity of robot $i$ and $w_{ij}(t)$ is the weight of this error. If the formation varies with time, these weights can also be time varying. Because in new formation, weight of some errors should become zero as a result of removing the interconnection between some robots and vice versa. It should be noted that in time varying formation, the desired value of distances between robots are also time varying.

Let us define (5) in matrix form

$$\sum_{i,j \in e} w_{ij}(t)\|x_i(t) - x_j(t) - d_{ij}^d\|^2 =$$

$$\sum_{i,j \in e} w_{ij}(t)\|Q_i(t)x_i(t) - Q_j(t)x_j(t)\|^2$$

where $Q_i(t) = \bar{D}(t)W_i(t)\bar{D}(t)^T$, $\bar{D}(t) = D(t) \otimes I_{2n}$, $W_i(t) = diag(w_{ij}(t))$ and $Q_i(t)$ is real symmetric positive semi-definite.

Again it is assumed that $x_i$ belongs to a compact and convex subset of $\mathbb{R}^{2n}$ that contains the $x_i^d$.

Other terms that should be considered in the players’ cost functions are the desired values of states at final time, keeping input bounded, and tracking the trajectory. Considering these terms, the cost function of each player can be expressed as follows

$$\int_{t_0}^{t_1} \left( x(t_f) - x^d(t_f) \right)^T Q_{if} \left( x(t_f) - x^d(t_f) \right) +$$

$$\int_{t_0}^{t_1} \left( x_i(t) - x_i^d(t) \right)^T Q_{ii} \left( x_i(t) - x_i^d(t) \right) +$$

$$u_i^T R_i u_i$$

where $Q_{if} = diag(w_{ij})$, $Q_{ii} = diag(w_i)$, and they are $2n \times 2n$ matrices and $R_i$ is a diagonal $n \times n$ matrix. It is assumed that $Q_{if}$ is real symmetric positive semi-definite matrix and $R_i$ is real symmetric positive definite matrix.

For robots which should track a trajectory named as leaders, the cost function can be represented as

$$\int_{t_0}^{t_1} \left( x(t_f) - x^d(t_f) \right)^T Q_{if} \left( x(t_f) - x^d(t_f) \right) +$$

$$\int_{t_0}^{t_1} \left( x(t) - x^d(t) \right)^T Q_i(t) \left( x(t) - x^d(t) \right) +$$

$$u_i^T R_i u_i$$

(8)
where $Q_{i}(t) = Q_{i}(t_{f}) + diag(0, ..., q_{i_{f}}, ..., 0)$. The index "l" stands for "leader robot".

The presented approach is applicable to different types of scenarios. For example, each robot can track the trajectory while keeping the desired distance for formation. In leader follower scenario, the leader robot may contribute in keeping formation when leading the group to follow the trajectory or just track the trajectory and leave the formation keeping to the followers. Depending on the task that is defined for each robot, the cost function can be modified properly.

It should be noted that time varying formation causes time varying incidence matrix $D$ and therefore time varying $Q_{i}$ and $Q_l$ which makes the control of formation challenging. Next section provides a proper control for predefined time varying formation.

3. CONTROL DESIGN

Gu (2008) used the error state and solve the problem as a linear quadratic regulator problem. In our paper, the desired states are included in solving procedure. Indeed, tracking problem is solved instead of regulator problem.

Generalizing the result of Engwerda (2005) for the problem of finding open loop Nash equilibrium for a linear quadratic regulator problem, the following theorem can be concluded.

**Theorem 1.** Assume a system with $m$ players and state equation (2). The problem is to find the control vectors $u_i$ ($i = 1, ..., m$) for each player such that the cost function (8) is minimized. Let us assume that there exist a set of solution $K_i$ ($i = 1, ..., m$) and $v_i$ $(i = 1, ..., m)$ to the coupled differential equations

$$
\dot{K_i}(t) = -K_i(t)A - A^{T}K_i(t) - Q_i(t) + K_i(t) \sum_{j=1}^{m} S_j K_j(t) \\
K_i(t_f) = Q_{i_{f}}
$$

(9)

$$
\dot{v_i}(t) = K_i(t) \sum_{j=1}^{m} S_j v_j(t) - A^T v_i(t) + Q_i(t)x_d(t) \\
v_i(t_f) = -Q_{i_{f}}x_d(t_f)
$$

(10)

where $S_i = B_i R^{-1} B_i^T$.

Then the problem has a unique open loop Nash equilibrium for every initial state as below

$$
u_i(t) = -R_i A_i^{T} B_i^T (K_i(\phi(t, 0) x_0 + f(t)) + v_i(t))
$$

(11)

where

$$
\phi(t, 0) = \left( A - \sum_{i=1}^{m} S_i K_i(t) \right) \phi(t, 0)
$$

(12)

and the close loop system is

$$
\dot{x}(t) = \left( A - \sum_{i=1}^{m} S_i K_i(t) \right) x(t) - \sum_{i=1}^{m} S_i v_i(t)
$$

(13)

The proof is given in Appendix A.

The result of Theorem 1, can be used for controlling time varying formation. The next theorem demonstrates this issue.

**Corollary 2.** Consider a team of $m$ mobile robots with dynamics defined by (2) and cost function (8) for each robot. To control a defined time varying formation of robots, if there exist a set of solutions to the coupled differential equations (9) and (10), the unique open loop Nash equilibrium for every initial state can be obtained from (11) and the resulting closed loop system is (14).

The proof is obtained directly from Theorem 1.

The necessary and sufficient condition for solvability of coupled Riccati differential equation is provided in (Engwerda 2005 and Gu 2008) It should be noted that the boundary conditions of differential equations (9) and (10) are terminal values. Therefore the differential equations should be solved backward in time.

4. SIMULATION

In this section, an illustrative example is provided. Consider a team of four robots. At first, they should form a triangle formation (Fig. 1) and after 8 seconds they should change their formation to a line (Fig. 2).

![Fig. 1. First formation](image1)

![Fig. 2. Second formation](image2)

For the first formation the incidence matrix and weight matrices are

$$
D_1 = \begin{bmatrix}
-1 & 0 & -1 \\
1 & -1 & 0 \\
0 & 1 & 0
\end{bmatrix}
$$

$$
W_1 = \begin{bmatrix}
10 & 0 & 0 \\
0 & 0 & 0 \\
0 & 10 & 0
\end{bmatrix} \quad W_2 = \begin{bmatrix}
10 & 0 & 0 \\
0 & 10 & 0 \\
0 & 0 & 0
\end{bmatrix}
$$

and for the second formation we have

$$
D_2 = \begin{bmatrix}
-1 & 0 & 0 \\
1 & -1 & 0 \\
0 & 1 & -1
\end{bmatrix}
$$

$$
W_3 = \begin{bmatrix}
10 & 0 & 0 \\
0 & 10 & 0 \\
0 & 0 & 0
\end{bmatrix} \quad W_4 = \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 10 & 0
\end{bmatrix}
$$
To describe the variation of incidence matrix and weight matrices due to variable formation, let us employ the proposed sigmoidal function. Therefore, the incidence and weight matrices are redefined as follows:

\[
D(t) = \begin{bmatrix}
-1 & 0 & 0 & 0 \\
1 & -1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
\end{bmatrix}
\]

\[
W_1 = \begin{bmatrix}
10 & 0 & 0 & 0 \\
0 & 0 & 0 & 10 \\
10 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
\end{bmatrix},
W_2 = \begin{bmatrix}
10 & 0 & 0 & 0 \\
0 & 10 & 0 & 0 \\
0 & 0 & 0 & 10 \\
0 & 0 & 10 & 0 \\
\end{bmatrix},
W_3 = \begin{bmatrix}
10 & 0 & 0 & 0 \\
0 & 0 & 0 & 10 \\
0 & 10 & 0 & 0 \\
0 & 0 & 0 & 0 \\
\end{bmatrix},
W_4 = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 10 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 10 & 0 \\
\end{bmatrix}
\]

Robot 1 is considered as the leader robot that should track a sinusoid trajectory. Other robots only keep the desired distance with their neighbours. Input weights are considered as follows:

\[
R_{ii} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, i = 1, ..., m
\]

Result of simulation for \( t_f = 10 \) sec is shown in Fig. 3.

Fig. 3. Time varying formation tracking a sinusoid path in given example.

As we can see in Fig. 3, at first robots form a triangle formation and after 8 second, the formation changes and they form a line formation. Meanwhile they track the sinusoid trajectory.

It should be noted that the second formation can also be achieved by time varying desired distance between robot 4 and robot 1. However, due to restriction on communication between robots, when the distance between two robots becomes more than a specific value, the formation graph (interconnection and communication between robots) should be changed. Changing the formation graph leads to changes in desired distances \( \sigma^d \).

5. CONCLUSION

In this paper, a differential game approach was proposed to control a time varying formation. The introduced formation control can be used in situations where we need to change the formation because of environmental constraints, occurring faults and changing neighbours. The convergence of the proposed approach is proved mathematically. The effectiveness of the proposed approach was confirmed by simulation results. In next step, we plan to deal with the design of state feedback Nash equilibrium solution to control

the time varying formation problem.

REFERENCES


Appendix A
Assume that there is a team of \( m \) players and the dynamics of players are described as follows

\[ x(t) = Ax(t) + \sum_{i=1}^{m} B_i u_i(t) \quad i = 1, \ldots, m \quad (15) \]

For each player, a cost function is defined as

\[ J^i(u) = \frac{1}{2} \left\{ \|x(t_f) - x^d(t_f)\|_{K_{if}}^2 \right. \\
+ \int_{t_f}^{t_i} \|x(\tau) - x^d(\tau)\|_{Q_{if}(\tau)}^2 \d\tau \right. \\
+ \|u_i(\tau)\|_{R_{ii}(\tau)}^2 \d\tau \right) \quad (16) \]

The necessary conditions for \( u_i \) to minimize the Hamiltonian are

\[ \dot{q}_i(t) = -\frac{\partial H_i}{\partial x} = -Q_{ii}(t)(x(t) - x^d(t)) - AA^T p_i(t) \quad (18) \]

and

\[ \frac{\partial H_i}{\partial u_i} = 0 \quad (19) \]

which results in

\[ R_{ii} u_i + B_i^T p_i(t) = 0 \quad (20) \]

Equation (20) yields

\[ u_i(t) = -R_{ii}^{-1} B_i^T p_i(t) \quad (21) \]

Substituting (21) in (15)

\[ \dot{x}(t) = Ax(t) - \sum_{i=1}^{m} B_i R_{ii}^{-1} B_i^T p_i(t) \quad (22) \]

Now define

\[ S_i = B_i R_{ii}^{-1} B_i^T \quad (23) \]

Substituting (23) in (22) leads to

\[ \dot{x}(t) = Ax(t) - \sum_{i=1}^{m} S_i p_i(t) \quad (24) \]

Augmenting the (18) and (24) for \( m \) robots results in

\[
\begin{bmatrix}
\dot{x}(t) \\
\dot{p}_1(t) \\
\vdots \\
\dot{p}_m(t)
\end{bmatrix}
= \begin{bmatrix}
A & -S_1 & \ldots & -S_m \\
Q_1(t) & -A^T & 0 & 0 \\
\vdots & 0 & \ddots & 0 \\
Q_m(t) & 0 & 0 & -A^T
\end{bmatrix}
\begin{bmatrix}
x(t) \\
p_1(t) \\
\vdots \\
p_m(t)
\end{bmatrix}

\begin{bmatrix}
0 \\
Q_1 x^d(t) \\
\vdots \\
Q_m x^d(t)
\end{bmatrix}
\quad (25)\]

Equation (25) can be redefined as follows

\[
\begin{bmatrix}
\dot{x}(t) \\
\dot{p}_1(t) \\
\vdots \\
\dot{p}_m(t)
\end{bmatrix}
= \begin{bmatrix}
A_{11} & -A_{12} \\
-A_{21} & -A_{22}
\end{bmatrix}
\begin{bmatrix}
x(t) \\
p_1(t)
\end{bmatrix}
+ \begin{bmatrix}
0 \\
A_{21} x^d(t)
\end{bmatrix}
\quad (26)\]

where

\[ P(t) = \begin{bmatrix}
p_1(t) \\
\vdots \\
p_m(t)
\end{bmatrix}
\]

\[ (A_{11})_{2n \times m} = A \quad (A_{12})_{(2n \times m) \times (2n \times m)} = \begin{bmatrix} S_1 & \ldots & S_m \end{bmatrix}
\]

\[ (A_{21}(t))_{(2n \times m) \times (2n \times m)} = \begin{bmatrix} Q_1(t) \\
\vdots \\
Q_m(t)
\end{bmatrix}
\]

\[ (A_{22}(t))_{(2n \times m) \times (2n \times m)} = \begin{bmatrix} A^T & 0 & 0 \\
0 & \ddots & 0 \\
0 & 0 & A^T
\end{bmatrix}
\]

Solving (26), yields

\[ x(t_f) = \phi(t_f,t)x(t) + \phi_{12}(t_f,t) P(t) + f_1(t) \quad (28) \]

\[ P(t_f) = \phi_{21}(t_f,t)x(t) + \phi_{22}(t_f,t) P(t) + f_2(t) \quad (29) \]

On the other hand, the boundary conditions are

\[ P(t_f) = K_f x(t_f) - K_f x^d(t_f), K_f = \begin{bmatrix} K_{1f} \\
K_{2f} \end{bmatrix} \quad (30) \]

Substituting \( x(t_f) \) and \( P(t_f) \) from (28) and (29) to the first boundary condition (30) leads to

\[ \phi_{21}(t_f,t) x(t) + \phi_{22}(t_f,t) P(t) + f_2(t) = K_f \left( \phi_{21}(t_f,t) x(t) + \phi_{12}(t_f,t) P(t) + f_1(t) \right) - K_f x^d(t_f) \]

By considering the above equation \( P(t) \) is obtained as follows

\[ P(t) = \left( \phi_{22}(t_f,t) - K_f \phi_{12}(t_f,t) \right)^{-1} \left( K_f \phi_{11}(t_f,t) - \phi_{21}(t_f,t) x(t) + \phi_{22}(t_f,t) P(t) + f_2(t) \right) \]

The above equation can be simplified as follows

\[ P(t) = K(t) x(t) + v(t) \quad (31) \]

Differentiating (31), yields

\[ \ddot{P}(t) = \dot{K}(t) x(t) + \dot{v}(t) \quad (32) \]

By substituting from (26) in (32), one gets

\[ -A_{21}(t) x(t) - A_{22} P(t) + A_{21}(t) x^d(t) = \dot{K}(t) x(t) + \dot{v}(t) + \left( \dot{v}(t) - K(t) A_{12} x(t) + A_{22} v(t) - A_{21}(t) x^d(t) \right) = 0 \quad (33) \]

Rearranging (33) results in

\[ \left( \dot{K}(t) + K(t) A + A_{22} K(t) + A_{21}(t) - K(t) A_{12} K(t) \right) x(t) + \left( \dot{v}(t) - K(t) A_{12} x(t) + A_{22} v(t) \right) - A_{21}(t) x^d(t) = 0 \quad (34) \]

Equation (34) has to exist for every \( x(t) \) and \( x^d(t) \). Therefore, one should have

\[ K(t) = -K(t) A + A_{22} K(t) - A_{21}(t) + K(t) A_{12} K(t) \quad \dot{v}(t) = (K(t) A_{12} - A_{22}) v(t) + A_{21} x^d(t) \]

In other words:

\[
\begin{bmatrix}
\dot{K}_1(t) \\
\dot{K}_m(t)
\end{bmatrix}
= -\begin{bmatrix} K_1(t) \\
K_m(t)
\end{bmatrix} A - \begin{bmatrix} 0 & 0 \\
0 & A^T\end{bmatrix} \begin{bmatrix} K_1(t) \\
K_m(t)
\end{bmatrix} - \begin{bmatrix} Q_1(t) \\
Q_m(t)
\end{bmatrix} + \begin{bmatrix} K_1(t) \\
K_m(t)
\end{bmatrix} \begin{bmatrix} S_1 & \ldots & S_m \end{bmatrix} \begin{bmatrix} K_1(t) \\
K_m(t)
\end{bmatrix} \quad (35)\]

\[
\begin{align*}
\begin{bmatrix}
v_1(t) \\
v_2(t) \\
\vdots \\
v_m(t)
\end{bmatrix}
&= 
\begin{bmatrix}
K_1(t) \\
K_2(t) \\
\vdots \\
K_m(t)
\end{bmatrix}
\begin{bmatrix}
[S_1 & \ldots & S_m]
\end{bmatrix}
\begin{bmatrix}
v_1(t) \\
v_2(t) \\
\vdots \\
v_m(t)
\end{bmatrix} \\
& \quad - 
\begin{bmatrix}
A^T & 0 & 0 \\
0 & \ddots & 0 \\
0 & 0 & A^T
\end{bmatrix}
\begin{bmatrix}
v_1(t) \\
v_2(t) \\
\vdots \\
v_m(t)
\end{bmatrix} \\
& \quad + 
\begin{bmatrix}
Q_1(t) \\
Q_2(t) \\
\vdots \\
Q_m(t)
\end{bmatrix}
x^d(t) \\
& \quad + 
\begin{bmatrix}
0 \\
0 \\
\vdots \\
0
\end{bmatrix}
\end{align*}
\]

Equation (31) for \( t = t_f \) results
\[
P(t_f) = K(t_f)x(t_f) + v(t_f)
\]
(37)

Substituting \( P(t_f) \) from (30) leads
\[
K_f x(t_f) - K_f x^d(t_f) = K(t_f)x(t_f) + v(t_f)
\]
(38)

Equation (38) has to exist for every \( x(t) \) and \( x^d(t) \). Therefore, one should have
\[
K(t_f) = K_f
\]
(39)
and
\[
v(t_f) = -K_f x^d(t_f)
\]
(40)

In other words
\[
\begin{bmatrix}
K_1(t_f) \\
K_2(t_f) \\
\vdots \\
K_m(t_f)
\end{bmatrix}
= 
\begin{bmatrix}
K_1 f \\
K_2 f \\
\vdots \\
K_m f
\end{bmatrix}
\]
and
\[
\begin{bmatrix}
\hat{v}_1(t_f) \\
\hat{v}_2(t_f) \\
\vdots \\
\hat{v}_m(t_f)
\end{bmatrix}
= 
\begin{bmatrix}
K_{1 f} \\
K_{2 f} \\
\vdots \\
K_{m f}
\end{bmatrix}
\begin{x^d(t_f)}
\end{bmatrix}
\]

Therefore, for each player the following equations should be solve to obtain the \( K_i \) and \( v_i \)
\[
\begin{align*}
K_i(t) &= -K_i(t)A - A^T K_i(t) - Q_i(t) + K_i(t) \sum_{j=1}^{m} S_j K_j(t) \\
\hat{v}_i(t) &= K_i(t) \sum_{j=1}^{m} S_j v_j(t) - A^T v_i(t) + Q_i(t)x^d(t)
\end{align*}
\]

The boundary conditions are
\[
K_i(t_f) = K_{i f}
\]
and
\[
\hat{v}_i(t_f) = -K_{i f}x^d(t_f)
\]

By substituting \( K_i \) and \( v_i \) from the above equations to (31) for each player, the costate \( p_i(t) \) is achieved as follows
\[
p_i(t) = K_i(t)x(t) + v_i(t)
\]
(41)

Considering (21) the control signal for each player is
\[
u_i(t) = -R_{ii}^{-1}B_i^T p_i(t)
\]
\[
= -R_{ii}^{-1}B_i^T(K_i(t)x(t) + v_i(t))
\]
(42)

Substituting \( u_i(t) \) from (42) to (24) leads
\[
\hat{x}(t) = (A - \sum_{i=1}^{m} S_i K_i(t))x(t) - \sum_{i=1}^{m} S_i v_i(t)
\]
(43)

Solving (43) results
\[
x(t) = \phi(t,0)x_0 + f(t)
\]
(44)

Differentiating (44), yields
\[
\dot{x}(t) = \hat{\phi}(t,0)x_0 + \hat{f}(t)
\]
(45)

Considering (43) and (45), and substituting \( x(t) \) from (44), one gets
\[
\left( A - \sum_{i=1}^{m} S_i K_i(t) \right) \dot{x}(t) = \left( A - \sum_{i=1}^{m} S_i K_i(t) \right) f(t) - \sum_{i=1}^{m} S_i v_i(t) = \phi(t,0)x_0 + \hat{f}(t)
\]

Rearranging the above equation results in
\[
\phi(t,0) - \left( A - \sum_{i=1}^{m} S_i K_i(t) \right) \phi(t,0) x_0
\]
\[+ \left( \hat{f}(t) - (A - \sum_{i=1}^{m} S_i K_i(t))f(t) + \sum_{i=1}^{m} S_i v_i(t) \right) = 0
\]