NASH-BASED STRATEGIES FOR THE CONTROL OF EXTENDED COMPLEX SYSTEMS

by

Yong Liu

B.S., Xi Dian University, 1996

M.S., Xi Dian University, 1999

Submitted to the Graduate Faculty of

School of Engineering in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2003

UNIVERSITY OF PITTSBURGH

SCHOOL OF ENGINEERING

This dissertation was presented

by

Yong Liu

It was defended on

October 24, 2003

and approved by

James F. Antaki, Associate Professor, Department of Biomedical Engineering, Carnegie Mellon University

J. Robert Boston, Professor, Department of Electrical Engineering

Luis F. Chaparro, Associate Professor, Department of Electrical Engineering

Ching-Chung Li, Professor, Department of Electrical Engineering

Dissertation Director: Marwan A. Simaan, Professor, Department of Electrical Engineering

ABSTRACT

NASH-BASED STRATEGIES FOR THE CONTROL OF EXTENDED COMPLEX SYSTEMS

Yong Liu, PhD

University of Pittsburgh, 2003

An extended complex system is a large scale hierarchical system controlled by two or more teams of decision-makers. The teams may have different objective functions, and often can be in direct conflict with each other. Within each team, the decision-makers must cooperate for the collective benefit of the team, but outside the team each member must compete with the decision-makers in the other teams. Decision-making in the context of such an extended complex system can be modeled as a new framework within the theory of games, called multi-team games. A multi-team game is a decision-making structure consisting of several interacting teams of cooperating decision-makers that are simultaneously in conflict with the other teams. In this dissertation, a new strategy, called Noninferior Nash strategy, is proposed to define optimal cooperative decisions for members of non-cooperative teams in an extended complex system. This strategy represents an equilibrium for the teams characterized by the property that no team has an incentive to unilaterally deviate, while maintaining cooperation among its members, in order to improve its overall team performance. The Noninferior Nash strategy in both static and dynamic systems is developed and its properties are investigated. In order to deal with the issue of non-uniqueness of the solution, a team structure that allows for a leader to oversee the overall performance of the team is introduced. The Noninferior Nash strategy with a Leader is formulated so as to select the particular Noninferior Nash strategy that is best for the team. To illustrate these concepts on a realistic system, we consider a practical example of a military air operation modeled as an extended complex system. The Nash Noninferior Strategies are investigated as possible solution concepts for dynamic teaming, team tasking, and unit task assignments and reassignments in the process of optimally planning of shared responsibilities and roles in the hierarchical deployment of the units in the combat. Simulation examples are presented to illustrate the effectiveness of these strategies in preserving the friendly force while destroying the defending enemy units.

DESCRIPTORS

Cooperative Control

Dynamic Task Assignment

Extended Complex System

Game Theory

Military Air Operation

Multi-Team Game

Nash Reassignment Strategy

Noncooperative Control

Noninferior Nash Strategy

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENTS

I would like to express my deeply gratitude to my advisor Dr. Marwan A. Simaan for his invaluable contributions toward my professional growth and for his guidance and advices on my doctoral study.

I express my truthful thanks to Dr. Ching-Chung Li, Dr. J.Robert Boston, Dr. Luis F. Chaparro and Dr. James F. Antaki for serving as my dissertation committee members and for their valuable comments and suggestion. I am very grateful to Dr. Jose B. Cruz, Jr. from the Ohio State University for his valuable advice, and to my colleague graduate students David G. Galati and Aca Gacic for many fruitful discussions on the topic of this thesis. I am very thankful to my friends in Pittsburgh for making this rough ride fun.

I am grateful to my parents, my sister and brother for their constant encouragement throughout my academic career.

My deepest appreciation goes to my husband, Qiang, for his great love and support for me. Without his encouragement, understanding and help, it would have been impossible to complete this dissertation.

I would like to express my gratitude to the Defense Advanced Research Projects Agency (DARPA) and the Air Force Research Laboratory (AFRL) for support of this work through grants number F30602-99-2-0549 and F33615-01-C-3151. The views and conclusions contained in this thesis are those of the author and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the DARPA, or the U.S. government.

1.0 INTRODUCTION

1.1 COMPLEX SYSTEMS IN THE PRESENCE OF AN ADVERSARY

Modern control systems must meet increasingly demanding requirements stemming from the need to cope with significant degrees of uncertainty, as well as more dynamic environment, and to provide greater flexibility [1]. This, in turn, means that a complex dynamic system invariably has a large number of interacting decision-making units and sometimes is vulnerable to various types of disturbances. A general objective of the complex system control is to maximize economic efficiency over a long time horizon. The optimal control to meet this objective is often accomplished by employing a multilevel hierarchical structure. Such control is known as hierarchical control. On the higher level, longer-term goals such as mission planning are considered; and on the lower lever, more specific operations such as mission execution are implemented. In order to implement the hierarchical control efficiently at any level of the system, the decision-makers are often grouped into a team. The main control efforts therefore become to tackle coordination and collaboration problems among these team members to achieve a common goal.

The processes and events that affect the performance of the complex system comprise the operational environment of the system. Usually, such an operational environment is viewed as an external part with uncertainty and noise. Considering these effects outside the system, the power of the control methodology becomes more and more necessary to enable the parts of the complex system to remain operational or even to automatically reconfigure themselves in the event of a threat or other potentially destabilizing disturbance. Management of disturbance in all such systems requires a basic understanding of the true system dynamics, as well as the resource and properties of the disturbance. If the nature of the disturbance is not very clear, it is often thought of as noise, or a random signal, with certain statistical properties. However, some of the disturbances may come from non-random sources such as another system with its own dynamics controlled by another team of decision-makers. We refer to this team of decision-makers as an adversary to the team controlling the original system. In such an extended view, the decision-makers in the adversarial system are treated the same way as independent decision-making units in the original system. It is clear that the relationship between the adversarial team and the original team is not necessarily cooperative, but may be more competitive since they generally have conflicting benefits. The overall system is known as an extended complex system [2].

1.2 MOTIVATION OF THE DISSERTATION

Within an extended complex system, it is apparent that most problems require multiple teams to represent the decentralized nature of the system, the multiple local controls, the multiple perspectives, or the competing interests. An extended complex system, therefore, can be best analyzed in the framework of the game theory. A game is controlled by a group of individuals such that the fate of an individual depends not only on his actions but also on the actions of the others in the group. In an extended complex system, the outcome is determined by the control actions of both the original team and its adversary. In this situation, the control problem for each decision-maker is: what choice should he make in order that his partial influence over the outcome benefits him most. Game theory provides possible answers to this question. It deals with choices that the decision-makers may make to reach an equilibrium outcome and in some cases with aspects related to the communication and collusion which may occur among the decision-makers in their attempts to improve their outcomes. During the past few years, many research fields such as in economics, telecommunication and military planning where conflicts or the cooperation between decision-makers arise have benefited from the introduction of game theoretic tools.

Noncooperative and cooperative game theories are the main two components of game theory. As their names suggest, noncooperative game theory provides decision-makers with strategies if they pursue their own interests which are completely or partly conflicting with others, whereas the cooperative game theory mainly works out the cooperative strategies among the decision-makers having common objectives. Thus, each component of game theories can deal with those systems where there is one relationship among the decision-makers: either noncooperative or cooperative. However, in an extended complex system, the requirements of noncooperation (between the original team and the adversary) and cooperation (within the original team or the adversary) must exist at the same time. The control design in either case is now required to coordinate its own decisions in ways consonant with the established global goals and also to minimize the adverse influence enforced by the adversary.

In this research, our objective is to develop a new game theoretic strategy to design optimal controllers for extended complex systems where cooperation and competition coexist. We will also investigate the problems of team composition and task assignment. That is, how to group the units in a system into cooperative teams and how to allocate these teams to accomplish the systemic tasks and meet the system objectives efficiently. These problems are very important especially in a large-scale complex system. In particular, we will demonstrate the traditional and newly developed game theoretic approaches on a practical example in a future combat system which involves optimal planning of shared responsibilities and roles in the hierarchical deployment and operation of teams of distributed cooperative semiautonomous entities and human operators.

1.3 ORGANIZATION OF THE DISSERTATION

The dissertation comprises seven chapters. The present chapter serves the purpose of introducing motivation of this dissertation. In the next chapter, we review the basic concepts and background material regarding the game-theoretical strategies in noncooperative games and cooperative games. This includes discussion and related literature on the properties of Nash strategies and Pareto strategies in static and dynamic games.

In Chapter 3, we formulate a new game framework, called a multi-team game, and develop a Noninferior Nash strategy to deal with cooperative control problems within one team while having an adversarial relationship with other teams. In this chapter, we discuss the properties of the Noninferior Nash strategies in finite and infinite games, and we investigate mathematical conditions for its existence. We also derive the corresponding analytical expressions for these strategies in both static and dynamic linear quadratic multi-team games.

In Chapter 4, we address the problem of how to select a specific strategy from the set of Noninferior Nash Strategies by introducing the concept of team leaders. We present two examples to demonstrate this strategy: one is a microeconomics problem and the other is routing control problems in the telecommunication network systems.

In Chapter 5, in order to apply our results to a practical problem in the military planning and decision-making area, we derive a mathematical attrition model for a military air operation involving two forces in combat. This model is used as a main test bed for analyzing our theoretical results. It is a discrete-time deterministic state space model with two opposing forces, labeled Blue and Red, each with multiple decision-makers and a two-level hierarchical control structure. In section 5.1, we describe the state variables, the control variables, the state difference equations and the objective functions in the model in details. In section 5.2, we apply a moving-horizon optimization scheme to this finite dynamic game with a single objective function defined for each side. The results of one-step and two-step look-ahead Nash controls are presented for comparison purposes.

In Chapter 6, we present several Nash-based strategies for cooperative teaming and dynamic task assignments that are an integral part of the military planning process. Nash reassignment strategies are introduced in section 6.1. We apply this strategy to address the dynamic resource allocation mechanism during the course of a military operation so as to improve the overall performance of the system. In section 6.2, Nash ordinal strategy is used to determine the initial team composition or mission plan for the top commanders based on their subjective experiences. In section 6.3, we investigate the effects of the strength of the two forces on teaming and tasking problems. In the last section, Noninferior Nash strategies are presented to deal with the cooperative control among the teams in the Blue force.

Finally, conclusions are given in Chapter 7 which summarizes the accomplishments of this research.

2.0 NONCOOPERATIVE AND COOPERATIVE GAME THEORY

The foundations of game theory were laid by John von Neumann with the publication of the book Theory of Games and Economic Behavior in 1944 [3]. They introduced the ideas of the extensive-form and normal-form (or strategic-form) representations of a game, defined the minimax solution, and showed that this solution exists in all two-player zero-sum games, which are noncooperative in nature. Nash (1950) [4] proposed what came to be known as "Nash equilibrium" as a way of extending game-theoretic analysis to noncooperative nonzero-sum games. Nash equilibrium is a natural generalization of the equilibria studied in specific models by Cournot [5], and it is the starting point for most economic analysis. The theory of dynamic games was introduced since the study of differential games was initiated by Isaacs in 1954 [6]. Minimax controls and Nash and Stackelberg open-loop and closed-loop controls were considered respectively by Starr and Ho [7] and [8], and Simaan and Cruz [9] in the general nonzero-sum differential games. In addition, the noninferior controls for cooperative players in a differential game were proposed in [7] and obtained from the pareto solutions to a multi-criterion (or vector-value) optimization problems [10],[11]. In this chapter, we will review the basic concepts and the strategies used in noncooperative games and cooperative games in details.

In general, a static game has three elements: (1) a set of Decision-Makers (DMs), also called players, denoted by $\mathrm{P} = \{P_1, P_2, \cdots, P_N\}$ where N is the number of DMs; (2) a strategy space for each DM U_i, $i = 1, \cdots, N$; (3) and a payoff function, $J_i(u_1, u_2, \cdots, u_N)$, for each DM to minimize where $u_i \in U_i$ ($i = 1, \cdots, N$). If $N = 2$, such a game is called a two-DM game. For example, economic competition by two companies or combat carried out by two forces against each other can be thought of as a two-DM game. If $\sum_{i=1}^{N} J_i(u_1, \cdots, u_N) = 0$, such game is called a zero-sum game; otherwise, it is called a nonzero-sum game. When the strategies in the strategy spaces U_i for $i = 1, \cdots, N$ form a finite set, we called such a game a finite game; otherwise, it is called an infinite game.

2.1 FINITE GAMES

An elementary way to represent a static finite game is in the normal (or matrix) form. A static game represented by a matrix is called a matrix game. Suppose the DMs P_1 and P_2 have m_1 and m_2 strategies to choose from, respectively. Thus, the dimension of the matrix is $m_1 \times m_2$. For P_1, the possible choices are the m_1 rows of the matrix, while for P_2 the possible choices are the m_2 columns of the matrix. Each entry of the matrix is a pair of outcomes of the payoff functions, i.e., $\left(J_1(u_1^i, u_2^j), J_2(u_1^i, u_2^j)\right)$ when P_1 and P_2 choose u_1^i and u_2^j as their strategies, respectively.

In two-DM zero-sum games, what is good for one DM is absolutely harmful to the other because their objectives are opposite. In this case, cooperation is not possible, which may be too restricted in some practical systems where the decision-makers, more or less, may have some common interests. Moreover, no one in the zero-sum games can gain from announcing his strategy in advance of his opponent. Thus, there is no hierarchical structure in zero-sum games. A more widely applied theory is that of nonzero-sum games. In a two-DM nonzero-sum game, as its name implies, the sum of the two payoff functions is not necessarily equal to zero or a constant. In other words, their objectives are not directly opposite. Thus, a possible collusion among the DMs is allowed. A little more complicated information structure occurs here: one DM may not know the other's payoff function, which is not the case in a two-DM zero-sum game. The strategies currently used in nonzero-sum games include the minimax strategy, the Nash strategy, the Stackelberg strategy and the noninferior strategy. In the view of different levels of cooperation between the two DMs, we may divide the strategies in nonzero-sum games into the following several categories.

2.1.1 Strategies with no cooperation

In a hostile environment, the DMs in a game do not have any prior information on any other DM's payoff. One could assume that the others want to do maximum harm to him only and thus takes a strategy to secure his losses against any (rational or irrational) action taken by the others. This assumption is pessimistic and the corresponding solution, called the minimax strategy, is also thought of as a pessimistic strategy. Its definition is given as

Definition 2.1 [5] A strategy u_i^* is a minimax strategy for the i^{th} DM (minimizer) if, for any admissible control $u_i \in U_i, \;\; i = 1, \cdots, N$,

$$\max_{u_1, \cdots, u_{i-1}, u_{i+1}, \cdots, u_N} J_i(u_1, \cdots, u_i^*, \cdots, u_N) \leq \max_{u_1, \cdots, u_{i-1}, u_{i+1}, \cdots, u_N} J_i(u_1, \cdots, u_i, \cdots, u_N) \tag{2.1}$$

Denote $\max_{u_1, \cdots, u_{i-1}, u_{i+1}, \cdots, u_N} J_i(u_1, \cdots, u_i^*, \cdots, u_N)$ by J_i^* $(i = 1, \cdots, N)$ where J_i^* is known as the security level of the i^{th} DM. $\left(J_1^*, J_2^*, \cdots, J_N^*\right)$ is called the minimax value of the games.

By using this strategy, each DM will achieve the smallest from all the largest possible values of its objective functions. For example, in Figure 2.1, the maximum cost incurred to P_1 is 3 if he chooses the strategy A or C. If he selects the strategy B, the maximum cost for him is 2. Thus, the minimax strategy for P_1 is the strategy B. This strategy guarantees that he will not suffer any loss greater than 2. Similarly, P_2 also has one minimax strategy a. If P_2 holds fast to the strategy a, his payoff will be no more than 1. The minimax value of this game is (2,1). However, if the strategy pair (B,a) is implemented, the outcome of the game is (1,-1), which is less than the minimax value of this game.

When a DM does not know the payoffs, or even the rationality, of the other DMs, the minimax strategy provides a useful solution concept to such a game. However, as we can see, since this strategy is so pessimistic, it is not widely used in practice.

		P_2			
		a	b	c	d
P_1	A	(1,1)	(3,3)	(3,1)	(-2,-1)
	B	(1,-1)	(-3,2)	(2,0)	(1,2)
	C	(-2,-1)	(-2,-3)	(3,2)	(1,0)

Figure 2.1 Minimax strategies for a two-DM nonzero-sum game

2.1.2 Strategies with limited cooperation

In some cases, the DMs in a nonzero-sum game may decide to make an agreement to some extent. This requires cooperation between the two DMs. How much cooperation they can achieve depends on how much information is available to them. In general, there are two cases of interest: absolutely equivalent information available to the two DMs and unequivalent information available to them. Here, we consider the strategies for the former case only.

If all the DMs in a nonzero-sum game know the exact information about each other's payoff functions, and they announce their strategies at the same time, then the strategy they use in this situation is called the Nash strategy. We give the definition of the Nash strategy in an *N*-DM nonzero-sum game as follows:

Definition 2.2 [7] The strategy set $\left(u_1^*,...,u_N^*\right)$ is a Nash equilibrium strategy set if,

$$J_i(u_1^*,\cdots,u_i^*,\cdots,u_N^*) \le J_i(u_1^*,\cdots,u_{i-1}^*,u_i,u_{i+1}^*,\cdots,u_N^*) \qquad \text{for } i=1,\cdots,N \quad (2.2)$$

where $u_i \in U_i$.

If the Nash strategy exists, it gives all the DMs a fair solution where any one of them cannot get a more satisfactory solution by refusing to use this strategy if the others stick to this strategy. For example, in Figure 2.1, the strategy pair (A,d) is a Nash solution. It should be noted that the minimax value in a game are definitely not lower (in an ordered way) than the values of any Nash equilibrium outcome. For example, in this game, the minimax value (2,1) is greater than the value of the Nash strategy (A,d), i.e., (-2,-1). Even when the unique Nash equilibrium strategies correspond to the minimax strategies, the minimax values could be higher than the values of the Nash equilibrium outcome. For example, in Figure 2.2, we can easily know that the minimax strategies for P_1 and P_2 are A and d, respectively. Clearly, the unique Nash strategy is (A,d) also. However, the value for the minimax strategy (A,d) is (2,2) which is greater than the value of the Nash strategy of (-2,-1).

Let us consider a three-DM nonzero-sum matrix game in which each DM has two alternatives to choose from. That is, $N = 3$ and the i^{th} DM is denoted by P_i (i=1,2,3). Suppose $U_1 = \{A,B\}$, $U_2 = \{C,D\}$ and $U_3=\{E,F\}$. The outcomes of the game can be displayed in the following two 2 × 2 matrices as shown in Figures 2.3 (a) and (b). The i^{th} component in each entry is the value of the payoff function for the i^{th} DM. The entries of the matrix (a) and (b) are the outcomes of the game if P_3 fixes his control at u_3=E and at u_3=F, respectively. We now claim that (B,D,F) is the Nash equilibrium strategy for this game. To check this, we can use the definition of the Nash strategy. If P_1 deviates from this equilibrium strategy u_1^*=B, then his loss becomes 3 which is not favorable. If P_2 deviates from u_2^*=D, his loss becomes 2 which is not favorable either. Finally, if P_3 deviates from u_3^*=F, his loss becomes 1 which is higher than his equilibrium loss 0. Consequently, (B,D,F) indeed provides a Nash equilibrium outcome, i.e., (0,1,0). By checking every possibility of strategy combinations, we note that this is the only Nash equilibrium solution of this 3-DM game.

		P_2			
		a	b	c	d
P_1	A	(1,3)	(2,3)	(2,1)	(-2,-1)
	B	(3,-1)	(-3,2)	(2,0)	(1,2)
	C	(-2,-1)	(-2,-3)	(3,4)	(1,0)

Figure 2.2 Nonzero-sum game where the Nash strategy is same as the minimax strategy

$P_3 : u_3$ =E		P_2	
		C	D
P_1	A	(0,-1,2)	(2,3,-1)
	B	(1,1,0)	(1,2,1)

(a)

$P_3 : u_3$ =F		P_2	
		C	D
P_1	A	(2,5,1)	(3,2,-1)
	B	(-1,2,-1)	(0,1,0)

(b)

Figure 2.3 A three-DM nonzero-sum game

One way to easily determine the Nash solution is to make use of the concept of reaction sets, which is given as

Definition 2.3 In a *N*-DM nonzero-sum finite game, let $u_{\bar{i}} = \left(u_1, \cdots, u_{i-1}, u_{i+1}, \cdots, u_N\right)$ and $U_{\bar{i}} = U_1 \times \cdots \times U_{i-1} \times U_{i+1} \times \cdots \times U_N$. The set $R_i(u_{\bar{i}}) \subset U_i$, defined for each $u_{\bar{i}} \in U_{\bar{i}}$ by

$$R_i(u_{\bar{i}}) = \left\{ u_i^* \in U_i : J_i(u_i^*, u_{\bar{i}}) = \min_{u_i \in U_i} J_i(u_i, u_{\bar{i}}) \right\} , \tag{2.3}$$

is the rational reaction (optimal response) set of DM i to the strategy $u_{\bar{i}} \in U_{\bar{i}}$ of other DMs.

For each DM, the reaction set should be nonempty in a finite game. It is well known that the Nash equilibrium solution can be obtained by taking the intersection of the reaction sets of all the DMs [7],[8] and [9]. Let $u^* = \left(u_1^*, \cdots, u_i^*, \cdots, u_N^*\right)$ and $u_{\bar{i}}^* = \left(u_1^*, \cdots, u_{i-1}^*, u_{i+1}^*, \cdots, u_N^*\right)$. u^* is a Nash equilibrium strategy if and only if $u_i^* \in R_i(u_{\bar{i}}^*)$ for $i = 1, \cdots, N$. All the Nash equilibrium solutions in the previous examples can be computed by taking the intersection elements of the reaction sets. If the set of intersection elements is empty, then there is no Nash strategy. Here, we give another example to illustrate this approach. Figure 2.4 gives a two-DM nonzero-sum game.

If P_1 chooses A, P_2 will choose a or c to obtain a minimum payoff of –1, i.e., R_2(A)={a,c}. If P_1 chooses B, P_2 will choose a or d with the corresponding the minimum cost of 2. Similarly, we can get all the reaction sets for P_1 and P_2, which are displayed in Table 2.1.

		P_2			
		a	b	c	d
P_1	A	(0,-1)	(-2,1)	(0,-1)	(-2,1)
	B	(3, 2)	(0,3)	(0,3)	(3,2)
	C	(2,1)	(-1,0)	(-1,0)	(2,1)

Figure 2.4 Application of the concept of reaction sets in a game

Table 2.1 Reaction sets for the example in Figure 2.4

(a)

u_1	$R_2(u_1)$
A	{a,c}
B	{a,d}
C	{b,c}

(b)

u_2	$R_1(u_2)$
a	{A}
b	{A}
c	{C}
d	{A}

We observe that A $\in R_1$(a) and a $\in R_2$(A). Thus, (A,a) is a Nash equilibrium solution. We also note that the reaction set of the strategy A is not unique, which includes the strategies a and c. However, (A,c) is not a Nash equilibrium solution because A $\notin R_1$(c). In other words, if P_1

chooses the strategy A and P_2 chooses the strategy c instead of the strategy a, P_2 seems not hurt by this selection because the cost incurred to him is still -1. However, P_1 may notice this possibility and will choose the strategy C if P_2 switches his control to c. Thus, (A,c) cannot be an equilibrium strategy pair. We may also check this by observing that J_1(A,c)> J_1(C,c), which doesn't satisfy the inequality expression (2.2) in the definition 2.2. In addition, we find that C$\in R_1$(c) and c$\in R_2$(C). Therefore, we know that (C,c) is another Nash equilibrium solution. Clearly, the Nash equilibrium solution is not unique in this example.

For the example in Figure 2.3, we may obtain the Nash solution by taking the intersection of the reaction sets of all three DMs, which is given in Table 2.2. Clearly, the intersected strategy is (B,D,F), which has been shown as the Nash solution to this example.

Table 2.2 Reaction set for the example in Figure 2.3

(a) normal form

P_3	E	F	P_2	C	D	P_1	A	B
A C	2	1	A E	-1	3	C E	0	1
A D	-1	-1	A F	5	2	C F	2	-1
B C	0	-1	B E	1	2	D E	2	1
B D	1	0	B F	2	1	D F	3	0

(b) reaction set

	R_3		R_2		R_1
A C	F	A E	C	C E	A
A D	E,F	A F	D	C F	B
B C	F	B E	C	D E	B
B D	F	B F	D	D F	B

2.1.3 Strategies under complete cooperation

A nonzero-sum game, in which all DMs have common interests and desire to improve their payoffs, if they cooperate, is called a cooperative game. In this situation, the noninferior strategy provides a mechanism in which the common benefits of all DMs can be optimized. Its formal definition is given as

Definition 2.4 [7] The strategy $\left(u_1^*,\cdots,u_N^*\right)$ belongs to the noninferior (or pareto) set if, for any other strategy $\left(u_1,\cdots,u_N\right)$,

$$\left\{J_i(u_1,\cdots,u_N) \le J_i(u_1^*,\cdots,u_N^*), i=1,\cdots,N\right\}$$

$$\text{only if} \quad \left\{J_i(u_1,\cdots,u_N) = J_i(u_1^*,\cdots,u_N^*), i=1,\cdots,N\right\}.$$

In the definition 2.4, we note that any strategy from the noninferior set, also called a noninferior strategy, is attempting to minimize the values of the payoff functions of all DMs. To agree on the noninferior strategy means that no other feasible choice of strategies could decrease the costs incurred to some DMs without increasing the costs incurred to the others. For example, in Figure 2.2, (B,b) and (C,b) are noninferior strategies. Note that (A,d) is a Nash strategy in this example, which is not better than (C,b). As we explained before, the Nash strategy is applicable in a hostile environment where both sides do not want to fully cooperate with each other. However, if they can reach an agreement on their interests, i.e., choosing a noninferior solution, they may get better results than by using their Nash strategies.

In general, there are more than one noninferior strategy that satisfy the definition 2.4 as seen from the above example. The values of the components in the noninferior set such as (-3,2) and (-2,-3) are not ordered by the vector criterion for the example in Figure 2.2. Thus, in order to implement this strategy, all the DMs need to share the information and agree on the final solution.

2.2 INFINITE GAMES

Games where at least one of the DMs has an infinite number of control choices form a class of infinite games. Infinite static games cannot be represented by matrices as the finite games. We still use the terminology as in the previous section, for example, there are N DMs, denoted by $P_1, P_2, \cdots, P_N$, where the payoff function of P_i is J_i and the action variable for P_i is u_i which is in the admissible control set U_i. The differences are that the admissible control set U_i is supposed to be a compact metric space and the payoff function $J_i(u_1, u_2, \cdots, u_N)$ is supposed to be continuous, differentiable and strictly convex on the product space $U_1 \times U_2 \times \cdots \times U_N$. As we know, by introducing the concept of reaction sets, we can easily determine the Nash solution in finite games. The notion of reaction sets is still important to infinite games. Particularly, if, for any $u_{\bar{i}} \in U_{\bar{i}}$, $R_i(u_{\bar{i}})$ is a singleton, $R_i(\cdot)$ is called the reaction curve or reaction function of P_i. When J_i is continuous, differentiable and strictly convex with respect to its arguments, the reaction curve of P_i, denoted by $C_i(u_{\bar{i}})$, can be obtained by taking the partial derivative of J_i with respect to his own control variable u_i and setting it to zero, i.e.,

$$\frac{\partial J_i(u_1, \cdots, u_N)}{\partial u_i} = 0 \Rightarrow u_i = C_i(u_{\bar{i}})$$

Using the vector notation, these relations can be written in the compact form as follows:

$$\begin{cases} u_1 = C_1(u_2, u_3, \cdots, u_N) \\ u_2 = C_2(u_1, u_3, \cdots, u_N) \\ \cdots \\ u_N = C_N(u_1, u_2, \cdots, u_{N-1}) \end{cases}$$

or,

$$u = C(u)$$

where $u = \left(u_1, \cdots, u_N\right)'$ and $C = \left(C_1, \cdots, C_N\right)'$.

The Nash strategies are the intersection points of the reaction curves of all the DMs. In other words, if $u^* = \left(u_1^*, u_2^*, \cdots, u_N^*\right)'$ is a Nash strategy, then it should satisfy that

$$u^* = C(u^*) .$$

If, for the i^{th} DM in the game, the cost function J_i is jointly continuous in all its arguments and strictly convex in u_i for every $u_j \in U_j \ (j \neq i)$ and U_i is a compact convex set, then the associated *N*-DM nonzero-sum game admits a Nash equilibrium strategy. One example for a two-DM nonzero-sum game is shown in Figure 2.5 [7]. The intersection point N is a Nash equilibrium solution. Neither of the DMs can improve its payoff if it decides to deviate from this point. If the two reaction curves do not intersect, a Nash solution will not exist. If the two reaction curves have more than one intersection points, each of them is a Nash equilibrium solution.

In a cooperative game, the noninferior strategies can be computed as solving a multi-objective optimal problem [13], if J_i's, $i = 1, \cdots, N$, are convex functions on a convex set

$U_1 \times U_2 \times \cdots \times U_N$. All the objective functions can be summed up and multiplied by stipulated weights $\alpha_1, \alpha_2, \ldots, \alpha_N$ to form one objective J, which is given by

$$\min_{(u_1, \cdots, u_N)} J = \sum_{i=1}^{N} \alpha_i J_i \tag{2.4}$$

where $\sum_{i=1}^{N} \alpha_i = 1,\ \alpha_i \geq 0$. Thus, after solving this optimal problem (2.4), the result is the noninferior strategy that the DMs are concerned with. In Figure 2.5, the dashed curve is the noninferior set of strategies. It is clear that we cannot find any other point with lower levels for both sides simultaneously than those points on the dashed curve. The selection of a specific solution in the noninferior set is generally done subjectively among all the DMs. Agreement on implementing the solution must also be reached. Without the convexity assumption, the solutions to the problem (2.4) provide a subset of noninferior solutions only [14],[15]. In other words, some noninferior solutions may never be discovered by solving the problem (2.4).

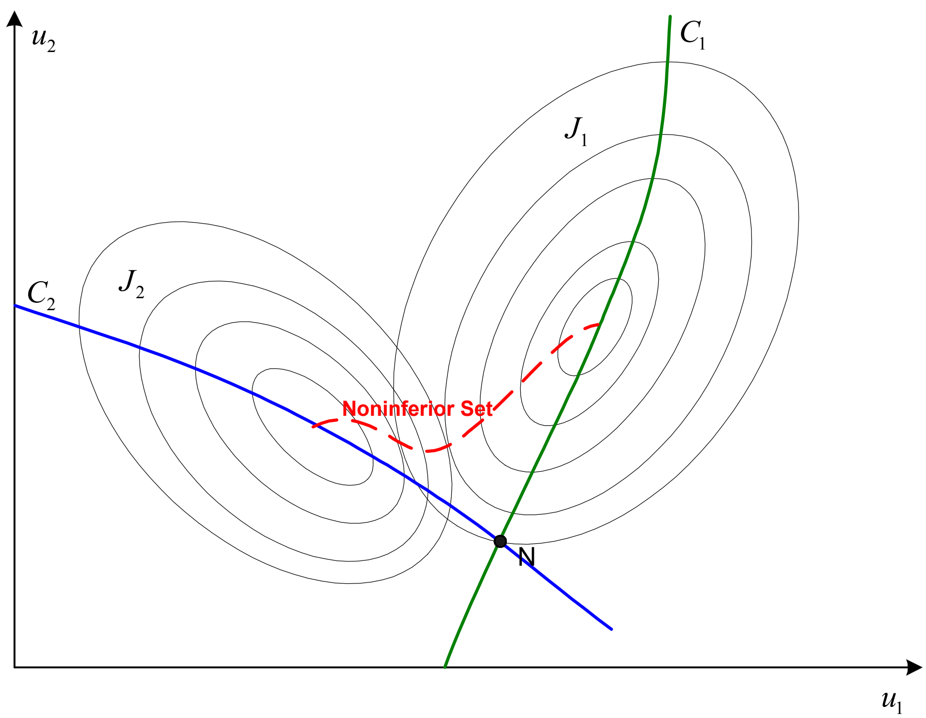


Figure 2.5 Reaction curves in a two-DM nonzero-sum infinite game

3.0 NONINFERIOR NASH STRATEGIES FOR EXTENDED COMPLEX SYSTEMS

Systems controlled by a large number of decision-makers with conflicting objectives are best analyzed using the traditional theory of games as reviewed in the previous chapter. In these systems, each decision-maker acts independently taking into account decisions made by all other decision-makers. The Nash and Stackelberg strategies [7],[18] are very powerful solution concepts for optimizing such systems. On the other hand, systems where all the decision-makers are willing to cooperate are best analyzed using concepts from team theory [19]. In these systems, each decision-maker must operate within the framework of the team, and the Noninferior (or Pareto) strategy [7] is a very powerful solution concept for optimizing such systems. Figure 3.1 shows a block diagram illustrating a system with individual non-cooperating decision-makers and Figure 3.2 shows a block diagram of a system with one team of cooperating decision-makers. In these diagrams, there are N decision-makers, denoted by $DM_1,....,DM_N$, whose control variables are expressed as $u_1,u_2,...,u_N$ respectively. The ith decision-maker has an objective function $J_i(u)$ to minimize, which is generally influenced not only by its own control variables but also by the control variables of all other decision-makers. That is $u=(u_1,u_2,...,u_N)'$.

An issue that arises in the optimization of systems that are controlled by one team is that, in general, the noninferior solution consists of a set and the decision-makers have to mutually agree and select one specific noninferior strategy from this set. An alternative mechanism is to assume that the team has a Leader decision-maker, DM_L, who selects from the set of noninferior solutions, a strategy that optimizes a mutually agreed-upon Leader objective function $J_L(u)$.

In the team optimization problem, because of the cooperative nature of the decision environment, all the decision-makers are included as members in one team. However, in a larger and more complicated organization, the decision environment may be such that some decision-makers may have compatible objectives with other decision-makers while at the same time having incompatible objectives with other individual, or other groups of, decision-makers. It is therefore reasonable to consider systems that are controlled by several competing teams of decision-makers, with each team consisting of several cooperating decision-makers. We refer to these types of systems as extended complex systems, or multi-team systems. The optimization of an extended complex system must be done within a framework that combines team theory with game theory. We refer to this framework as **nonzero-sum multi-team games (MTGs)**. Compared with the optimization schemes of Figure 3.1 and Figure 3.2, a block diagram illustrating the architecture of an extended complex system is shown in Figure 3.3. Zero-sum multi-team games where all the decision-makers in each team have the same objective function have been studied in [20]. Similarly, systems controlled by more than two decision-makers where there exists the possibility of a subset of decision-makers forming a coalition (team) so that the worst performing member in the coalition cannot be improved with another decision without degrading the worst performance of another member in the coalition, have been studied in [21] and [22].

The solution framework of multi-team systems is inherently large and complex due to the introduction of both complicated relationships among the decision-makers and team objective functions. In this chapter, we will first develop a strategy that provides for cooperation among all members within each team and insures a non-cooperative Nash equilibrium among all teams. We refer to this strategy as the Noninferior Nash Strategy (NNS). We show that for systems with continuous control variables, the NNS for each team belongs to a set of solutions.

This chapter is organized as follows. In section 3.1, we formulate the multi-team game problem, define the NNS, and discuss its properties. In section 3.2, we obtain conditions for existence of the NNS in static continuous systems and derive analytical expressions for these strategies for a class of systems with linear quadratic objective functions. In section 3.3, we obtain the conditions for the existence of open-loop and closed-loop NNS solutions in linear quadratic differential multi-team games. Finally, in section 3.4 we present some concluding remarks.

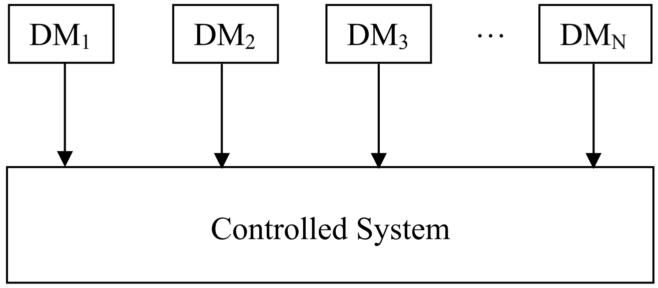


Figure 3.1 System with individual non-cooperative decision-makers

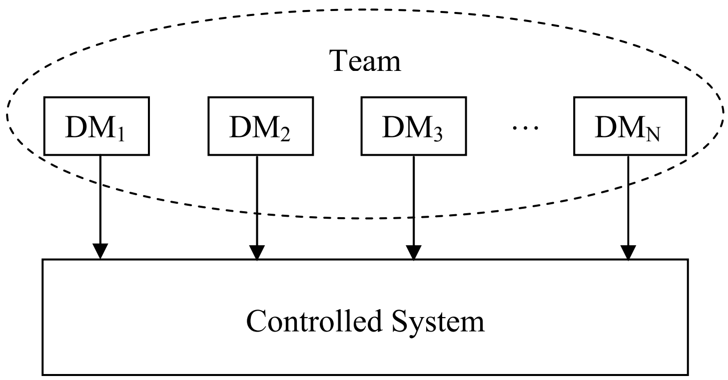


Figure 3.2 System with one team of cooperative decision-makers

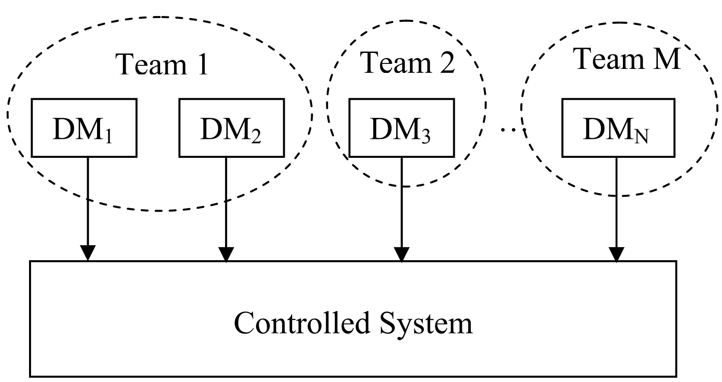


Figure 3.3 System with multiple teams of decision-makers

3.1 NONINFERIOR NASH STRATEGIES IN FINITE STATIC MTGS

Without loss of generality, and for the sake of simplicity of notation, in this thesis we will consider multi-team systems where there are only two teams: **Team 1** and **Team 2**. Systems with more than two teams can be treated in a very similar manner. Let team X have m_X members of decision-makers ($X=1, 2$) and let the control variable of the i^{th} member u_i^X be a vector of dimension k_i^X. Let $u^X = \left(u_1^X, \cdots, u_{m_X}^X\right)'$ denote the overall control vector for team X. Let U_i^X be the admissible control set for the i^{th} member in team X. Thus $U^X = U_1^X \times U_2^X \times \cdots \times U_{m_X}^X$ is the admissible set for the overall control vector u^X of team X. Assume that the i^{th} member in team X wishes to minimize an objective function $J_i^X(u^1, u^2)$ by choosing u_i^X. Note that the cost function $J_i^X(u^1, u^2)$ depends on the control variables of all decision-makers in both teams. The optimization of such a system can be formulated as a pair of vector-valued minimizations of the form:

$$\min_{u^X \in U^X} \begin{pmatrix} J_1^X(u^1, u^2) \\ J_2^X(u^1, u^2) \\ . \\ . \\ J_{m_X}^X(u^1, u^2) \end{pmatrix}, \quad \text{for } X = 1, 2$$

In these systems, we stipulate that the relationship between the two teams is completely adversarial and that cooperation between them is not permissible. In other words, both cooperation within each team and competition between the teams must coexist. An optimum solution $\{\hat{u}^1, \hat{u}^2\}$ with $\hat{u}^1 = \left(\hat{u}_1^1, \hat{u}_2^1, \cdots, \hat{u}_{m_1}^1\right)'$ and $\hat{u}^2 = \left(\hat{u}_1^2, \hat{u}_2^2, \cdots, \hat{u}_{m_2}^2\right)'$, if it exists, must possess the following two properties:

Property 3.1 Within each team *X*, the control vector $\hat{u}^X$ is a noninferior (or Pareto) strategy for team *X*, and

Property 3.2 Between the two teams, the pair of control vectors $\{\hat{u}^1, \hat{u}^2\}$ is a Nash equilibrium strategy.

Thus, with this pair of strategies $\{\hat{u}^1, \hat{u}^2\}$ there is no incentive for the members in one team to collectively deviate, since this will not improve the objective functions of all members of that team simultaneously, but instead will cause a deterioration in the overall team's performance. We will refer to this strategy as the **Noninferior Nash Strategy** (NNS), and its formal definition is given by:

Definition 3.1 The pair of control vectors $\{\hat{u}^1, \hat{u}^2\} \in U^1 \times U^2$ is a NNS if, for any other $u^1 \in U^1$ and $u^2 \in U^2$,

$$\left\{J_i^1(u^1, \hat{u}^2) \le J_i^1(\hat{u}^1, \hat{u}^2),\ i = 1, \ldots, m_1\right\} \text{ only if } \left\{J_i^1(u^1, \hat{u}^2) = J_i^1(\hat{u}^1, \hat{u}^2),\ i = 1, \ldots, m_1\right\}, \quad (3.1)$$

and

$$\left\{J_i^2(\hat{u}^1, u^2) \le J_i^2(\hat{u}^1, \hat{u}^2),\ i = 1, \ldots, m_2\right\} \text{ only if } \left\{J_i^2(\hat{u}^1, u^2) = J_i^2(\hat{u}^1, \hat{u}^2),\ i = 1, \ldots, m_2\right\}. \quad (3.2)$$

Each condition in the above definition requires that the control vector chosen by one team (say $\hat{u}^1$ for team 1) be a noninferior solution against the control vector chosen by the other team ($\hat{u}^2$ for team 2). Additionally, a pair of control vectors $\{\hat{u}^1, \hat{u}^2\}$ that satisfies conditions (3.1) and (3.2) simultaneously will also represent a Nash equilibrium solution between the two teams.

In order to illustrate the general idea behind this solution concept, let us consider the simple two-team matrix game shown in Figure 3.4, and the team composition as given in Table 3.1. The first team has two decision-makers, denoted by DM_1^1 and DM_2^1, respectively. Decision-maker DM_1^1 has a control variable u_1^1 with two choices: A and B; and decision-maker DM_2^1 has another control variable u_2^1 with two choices: C and D. The second team also has two decision-makers, denoted by DM_1^2 and DM_2^2, respectively. Decision-maker DM_1^2 has a control variable u_1^2 with two choices: a and b; and decision-maker DM_2^2 has another control variable u_2^2 with also two choices: c and d. For each pair of choices $\{u^1, u^2\}$ with $u^1 = \begin{pmatrix} u_1^1 \\ u_2^1 \end{pmatrix}$ and $u^2 = \begin{pmatrix} u_1^2 \\ u_2^2 \end{pmatrix}$, the corresponding entries in the matrix shown in Figure 3.4 are the pair of vector-valued objective functions $\begin{pmatrix} J_1^1(u^1, u^2) \\ J_2^1(u^1, u^2) \end{pmatrix}$ for team 1 and $\begin{pmatrix} J_1^2(u^1, u^2) \\ J_2^2(u^1, u^2) \end{pmatrix}$ for team 2. Each decision-maker wants to cooperate with the other member in its team while at the same time insuring that a Nash equilibrium exists between the collective choices of the two teams.

According to the definition given above, for the game in Figure 3.4, we can determine that the pair $\{\hat{u}^1, \hat{u}^2\}$ with $\hat{u}^1 = \begin{pmatrix} A \\ C \end{pmatrix}$ and $\hat{u}^2 = \begin{pmatrix} b \\ c \end{pmatrix}$ is a Noninferior Nash strategy. If the decision-makers in team 1 stick to the strategy $\hat{u}^1$, then the decision-makers in team 2 cannot improve both values of their objective functions by changing the strategy $\hat{u}^2$. Similarly, if the strategy of the decision-makers in team 2 remains fixed at $\hat{u}^2$, then the decision-makers in team 1 have no incentive to choose a strategy different from $\hat{u}^1$ because this will not improve the benefits for both decision-makers in that team simultaneously. In other words, the strategies $\hat{u}^1 = \begin{pmatrix} A \\ C \end{pmatrix}$ and $\hat{u}^2 = \begin{pmatrix} b \\ c \end{pmatrix}$

		Team 2			
	$\begin{pmatrix}J_1^2\\J_2^2\end{pmatrix}$ $\begin{pmatrix}J_1^1\\J_2^1\end{pmatrix}$	$\begin{pmatrix}a\\c\end{pmatrix}$	$\begin{pmatrix}a\\d\end{pmatrix}$	$\begin{pmatrix}b\\c\end{pmatrix}$	$\begin{pmatrix}b\\d\end{pmatrix}$
Team 1	$\begin{pmatrix}A\\C\end{pmatrix}$	$\begin{pmatrix}4\\4\end{pmatrix},\begin{pmatrix}3\\4\end{pmatrix}$	$\begin{pmatrix}6\\6\end{pmatrix},\begin{pmatrix}6\\5\end{pmatrix}$	$\begin{pmatrix}0\\1\end{pmatrix},\begin{pmatrix}3\\3\end{pmatrix}$	$\begin{pmatrix}3\\3\end{pmatrix},\begin{pmatrix}9\\9\end{pmatrix}$
	$\begin{pmatrix}A\\D\end{pmatrix}$	$\begin{pmatrix}3\\3\end{pmatrix},\begin{pmatrix}1\\1\end{pmatrix}$	$\begin{pmatrix}5\\5\end{pmatrix},\begin{pmatrix}0\\0\end{pmatrix}$	$\begin{pmatrix}2\\2\end{pmatrix},\begin{pmatrix}3\\4\end{pmatrix}$	$\begin{pmatrix}1\\1\end{pmatrix},\begin{pmatrix}4\\3\end{pmatrix}$
	$\begin{pmatrix}B\\C\end{pmatrix}$	$\begin{pmatrix}3\\4\end{pmatrix},\begin{pmatrix}2\\1\end{pmatrix}$	$\begin{pmatrix}5\\4\end{pmatrix},\begin{pmatrix}2\\4\end{pmatrix}$	$\begin{pmatrix}1\\0\end{pmatrix},\begin{pmatrix}2\\2\end{pmatrix}$	$\begin{pmatrix}2\\4\end{pmatrix},\begin{pmatrix}3\\4\end{pmatrix}$
	$\begin{pmatrix}B\\D\end{pmatrix}$	$\begin{pmatrix}4\\3\end{pmatrix},\begin{pmatrix}6\\0\end{pmatrix}$	$\begin{pmatrix}5\\5\end{pmatrix},\begin{pmatrix}3\\4\end{pmatrix}$	$\begin{pmatrix}2\\4\end{pmatrix},\begin{pmatrix}3\\3\end{pmatrix}$	$\begin{pmatrix}2\\3\end{pmatrix},\begin{pmatrix}9\\9\end{pmatrix}$

Figure 3.4 A two-team game in matrix form

Table 3.1 Team composition and decision variables

Teams	Team composition	Decision Choices
Team 1	DM_1^1	A, B
	DM_2^1	C, D
Team 2	DM_1^2	a, b
	DM_2^2	c, d

satisfy both conditions (3.1) and (3.2) of the above definition simultaneously and thus constitute an NNS.

The counterpart of the traditional reaction set of game theory when figuring out a Noninferior Nash strategy is called the **Noninferior Reaction Set (NRS)** and is defined as follows:

Definition 3.2 The map $R_{NRS}^2[u^1]: U^1 \to U^2$ is defined as the Noninferior Reaction Set for team 2 if given any arbitrary control vector $u^1 \in U^1$ for team 1, the control vector $u^2 \in R_{NRS}^2[u^1]$ satisfies:

$$\left\{ J_i^2(u^1, u^{2\#}) \le J_i^2(u^1, u^2),\ i = 1, \dots, m_2 \right\} \text{ only if } \left\{ J_i^2(u^1, u^{2\#}) = J_i^2(u^1, u^2),\ i = 1, \dots, m_2 \right\} \quad (3.3)$$

for all $u^{2\#} \in U^2$.

In a similar way, we can define $R_{NRS}^1[u^2]: U^2 \to U^1$ as the Noninferior Reaction set for team 1. Thus, the noninferior reaction set for team 2 is equivalent to the collection of all noninferior control sets for team 2 for all possible choices of control vectors by the members of team 1. For the above example, Figure 3.5 illustrates how this is done when team 2 chooses $u^2 = \begin{pmatrix} b \\ c \end{pmatrix}$. In this situation, the matrix game shown in Figure 3.5 describes the options available for the two decision-makers DM_1^1 and DM_2^1 in team 1. The noninferior solution set in this case consists of the two pairs of controls $\begin{pmatrix} A \\ C \end{pmatrix}$ and $\begin{pmatrix} B \\ C \end{pmatrix}$. Thus, $R_{NRS}^1[\begin{pmatrix} b \\ c \end{pmatrix}] = \{\begin{pmatrix} A \\ C \end{pmatrix}, \begin{pmatrix} B \\ C \end{pmatrix}\}$.

Team 2: $\begin{pmatrix} b \\ c \end{pmatrix}$		Decision-maker DM_2^1 of Team 1	
		C	D
Decision-maker DM_1^1 of Team 1	A	$\begin{vmatrix} 0 \\ 1 \end{vmatrix}$	$\begin{pmatrix} 2 \\ 2 \end{pmatrix}$
	B	$\begin{pmatrix} 1 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 2 \\ 4 \end{pmatrix}$

Figure 3.5 Matrix game for DM_1^1 and DM_2^1 when DM_1^2 and DM_2^2 select b and c

With the introduction of the concept of noninferior reaction sets, it is clear that a strategy $\{\hat{u}^1, \hat{u}^2\} \in U^1 \times U^2$ is a NNS if

$$\hat{u}^1 \in R_{NRS}^1[\hat{u}^2] \text{ and } \hat{u}^2 \in R_{NRS}^2[\hat{u}^1] \tag{3.4}$$

That is, a NNS must lie in the intersection of the noninferior reaction sets of the two teams. In order to illustrate this approach for finding the NNS, let us determine the noninferior reaction sets for the two teams of Figure 3.4. These sets are shown in Table 3.2.

The unique intersection of both reaction sets is the pair $\{\begin{pmatrix}A\\C\end{pmatrix},\begin{pmatrix}b\\c\end{pmatrix}\}$ which is the NNS for this game. Note that $R^1_{NRS}[\begin{pmatrix}b\\c\end{pmatrix}]$ has two elements which are: $\begin{pmatrix}A\\C\end{pmatrix}$ and $\begin{pmatrix}B\\C\end{pmatrix}$ with corresponding outcomes for team 1 of $\begin{pmatrix}0\\1\end{pmatrix}$ and $\begin{pmatrix}1\\0\end{pmatrix}$, respectively. However, the pair $\{\begin{pmatrix}B\\C\end{pmatrix},\begin{pmatrix}b\\c\end{pmatrix}\}$ is not a NNS. If team 2 chooses $\begin{pmatrix}b\\c\end{pmatrix}$ and team 1 chooses $\begin{pmatrix}B\\C\end{pmatrix}$, only decision-maker DM^1_2 in team 1 obtains a better outcome. On the other hand, if team 2 knows that team 1 may choose $\begin{pmatrix}B\\C\end{pmatrix}$, it will choose $\begin{pmatrix}a\\c\end{pmatrix}$ instead of $\begin{pmatrix}b\\c\end{pmatrix}$ since $R^2_{NRS}[\begin{pmatrix}B\\C\end{pmatrix}]=\begin{pmatrix}a\\c\end{pmatrix}$. Hence, the control pair $\{\begin{pmatrix}B\\C\end{pmatrix},\begin{pmatrix}b\\c\end{pmatrix}\}$ is not an equilibrium solution.

Table 3.2 Noninferior reaction sets for the game in Figure 3.4

u^1	$R^2_{NRS}[u^1]$	u^2	$R^1_{NRS}[u^2]$
$\begin{pmatrix}A\\C\end{pmatrix}$	$\begin{pmatrix}b\\c\end{pmatrix}$	$\begin{pmatrix}a\\c\end{pmatrix}$	$\begin{pmatrix}A\\D\end{pmatrix}$
$\begin{pmatrix}A\\D\end{pmatrix}$	$\begin{pmatrix}a\\d\end{pmatrix}$	$\begin{pmatrix}a\\d\end{pmatrix}$	$\begin{pmatrix}B\\C\end{pmatrix}$
$\begin{pmatrix}B\\C\end{pmatrix}$	$\begin{pmatrix}a\\c\end{pmatrix}$	$\begin{pmatrix}b\\c\end{pmatrix}$	$\{\begin{pmatrix}A\\C\end{pmatrix},\begin{pmatrix}B\\C\end{pmatrix}\}$
$\begin{pmatrix}B\\D\end{pmatrix}$	$\{\begin{pmatrix}a\\c\end{pmatrix},\begin{pmatrix}b\\c\end{pmatrix}\}$	$\begin{pmatrix}b\\d\end{pmatrix}$	$\begin{pmatrix}A\\D\end{pmatrix}$

3.2 CONTINUOUS STATIC MTGS

3.2.1 Noninferior Nash strategies in continuous static multi-team games

Consider a two-team game, with m_1 decision-makers in team 1 and m_2 decision-makers in team 2. Let the control vectors u_i^X for the members $i = 1, \cdots, m_X$ in each team be grouped into a team control vector $u^X = (u_1^X, u_2^X, ..., u_{m_X}^X)$ that belong to compact and convex admissible sets of the form $U^X = U_1^X \times U_2^X \times \cdots \times U_{m_X}^X$ where X =1, 2. Let the objective function of the i^{th} decision-maker in team X, $J_i^X(u^1, u^2)$, be a real-valued continuous and strictly convex function on $U^1 \times U^2$. For the purpose of simplifying the notation, when one team is denoted by X, we will use $\bar{X}$ to denote the other team, and vice versa. That is,

$$\bar{X} = \begin{cases} 2 & \text{when } X = 1 \\ 1 & \text{when } X = 2 \end{cases}.$$

Now let us assume that team $\bar{X}$ has chosen a team control $u^{\bar{X}}$, then the corresponding noninferior reaction set for team X can be determined by minimizing the function:

$$J^{X,\xi^X}(u^1, u^2) = \sum_{i=1}^{m_X} \xi_i^X J_i^X(u^1, u^2) \tag{3.5}$$

with respect to u^X for every vector of parameters $\xi^X = (\xi_1^X, \xi_2^X, ..., \xi_{m_X}^X)' \in W^X$ where W^X is given by

$$W^X = \left\{ \xi^X \in \mathbb{R}^{m_X} : \sum_{i=1}^{m_X} \xi_i^X = 1, \quad 0 \le \xi_i^X \le 1, \quad \xi^X = (\xi_1^X, \xi_2^X, \cdots, \xi_{m_X}^X)' \right\} \tag{3.6}$$

Let $C_{NRS}^{X}(\xi^{X}, u^{\bar{X}})$ denote the set of solutions u^{X,ξ^X} to the optimization problem given in (3.5) and parameterized by ξ^{X}. We now give a definition of the NNS in terms of the vector $\xi = \begin{pmatrix} \xi^1 \\ \xi^2 \end{pmatrix}$, followed by a theorem that provides necessary conditions for its existence.

Definition 3.3 For a given vector ξ, the pair of team control vectors $\{\hat{u}^{1,\xi}, \hat{u}^{2,\xi}\}$ is a Noninferior Nash strategy if

$$\hat{u}^{1,\xi} \in C_{NRS}^{1}(\xi^1, \hat{u}^{2,\xi}) \text{ and } \hat{u}^{2,\xi} \in C_{NRS}^{2}(\xi^2, \hat{u}^{1,\xi}) \tag{3.7}$$

Theorem 3.1 (Existence of NNS in Two-Team Games) For each team $X \in \{1,2\}$, let $U^1 \times U^2$ be a compact and convex subset of $\mathbb{R}^{\sum_{i=1}^{m_1} k_i + \sum_{j=1}^{m_2} k_j}$. Let the cost functional $J_i^X(u^1,u^2): U^1 \times U^2 \to R$ for $i = 1,....,m_X$ be jointly continuous in u^1 and u^2 $\in U^1 \times U^2$, and strictly convex in u^X for every $u^{\bar{X}} \in U^{\bar{X}}$. Then, for every vector of weights $\xi = \begin{pmatrix} \xi^1 \\ \xi^2 \end{pmatrix}$ there exists a Noninferior Nash solution.

Proof. Without loss of generality let us consider the reaction of team 1 for a specific choice $u^2 \in U^2$ by team 2. The noninferior control vector $u^{1,\xi^1} \in U^1$ can be determined by minimizing with respect to u^1 the function $J^{1,\xi^1}(u^1,u^2)$ defined in (3.5). Since J_i^1 for $i = 1,2,\cdots,m_1$ are strictly convex for all $u^1 \in U^1$ it follows that $J^{1,\xi^1}(u^1,u^2)$ is also strictly convex for all $u^1 \in U^1$. Hence there exists a unique mapping $f_{\xi^1}^1 : U^2 \to U^1$ such that $u^{1,\xi^1} = f_{\xi^1}^1(u^2)$ uniquely minimizes $J^{1,\xi^1}(u^1,u^2)$ for the given $u^2 \in U^2$. The mapping $f_{\xi^1}^1$ represents a noninferior reaction solution for team 1 when it uses a weight vector ξ^1. Similarly, the noninferior reaction solution for team 2 when it uses a weight vector ξ^2, given that team 1 chooses $u^1 \in U^1$, can be determined as the unique mapping $f^2_{\xi^2} : U^1 \to U^2$, i.e., $u^{2,\xi^2} = f^2_{\xi^2}(u^1)$. Using a vector notation, these two mappings can be combined in a compact form as $\overline{u}^{\xi} = F_{\xi}(\overline{u}^{\xi})$ where $\overline{u}^{\xi} = \begin{pmatrix} u^{1,\xi^1} \\ u^{2,\xi^2} \end{pmatrix}$, $F_{\xi} = \begin{pmatrix} f^1_{\xi^1} \\ f^2_{\xi^2} \end{pmatrix}$ and $\xi = \begin{pmatrix} \xi^1 \\ \xi^2 \end{pmatrix}$. Clearly, the functions $f^1_{\xi^1}$ and $f^2_{\xi^2}$ are continuous in their arguments, and hence F_{ξ} is a continuous mapping. Since F_{ξ} maps $U^1 \times U^2$ into itself, and because of the compactness of U^1 and U^2, by using Kakutani fixed point theorem [23], there exists a unique $\hat{u}^{\xi} = \begin{pmatrix} \hat{u}^{1,\xi} \\ \hat{u}^{2,\xi} \end{pmatrix}$ such that $\hat{u}^{\xi} = F^{\xi}(\hat{u}^{\xi})$. The pair $\{\hat{u}^{1,\xi}, \hat{u}^{2,\xi}\} \in U^1 \times U^2$ belongs to the intersection of both reaction sets and hence it constitutes a Noninferior Nash Strategy for the given weight vectors ξ^1 and ξ^2. □

3.2.2 Noninferior Nash strategies in quadratic multi-team games

Quadratic games with quadratic cost functions are of particular interest in the game theory. In a quadratic multi-team system each decision-maker has a quadratic cost function. For decision-maker *i* in team *X* (*X*=1, 2), let the objective function be of the form:

$$J_i^X(u^1,u^2) = \frac{1}{2}\begin{pmatrix}(u^1)' & (u^2)'\end{pmatrix}\begin{pmatrix} R^{Xi,11} & R^{Xi,12} \\ R^{Xi,21} & R^{Xi,22} \end{pmatrix}\begin{pmatrix} u^1 \\ u^2 \end{pmatrix} + \begin{pmatrix}(r^{Xi,1})' & (r^{Xi,2})'\end{pmatrix}\begin{pmatrix} u^1 \\ u^2 \end{pmatrix} + c^{Xi} \tag{3.8}$$

where $u^X = \begin{pmatrix} u_1^X \\ \cdot \\ \cdot \\ u_{m_X}^X \end{pmatrix}$ and $u_i^X \in \mathbb{R}^{k_i^X}$ with k_i^X being the dimension of the control vector for member i in team X. In (3.8), $R^{Xi,21} = \left(R^{Xi,12}\right)'$ and the matrices $R^{Xi,pq}$ for $X = 1,2; p = 1,2;$ and $q = 1,2$ are partitioned as follows:

$$R^{Xi,pq} = \begin{pmatrix} R_{11}^{Xi,pq} & R_{12}^{Xi,pq} & \cdots & R_{1m_q}^{Xi,pq} \\ R_{21}^{Xi,pq} & R_{22}^{Xi,pq} & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ R_{m_p1}^{Xi,pq} & \cdots & \cdots & R_{m_pm_q}^{Xi,pq} \end{pmatrix} \tag{3.9}$$

and the sub-matrices $R_{sl}^{Xi,pq}$ have dimensions $(k_s^p \times k_l^q)$ for $s = 1,\cdots,m_p$; $l = 1,\cdots,m_q$. The vectors $r^{Xi,p}$ for $X = 1,2$ and $p = 1,2$ are partitioned as follows:

$$r^{Xi,p} = \begin{pmatrix} r_1^{Xi,p} \\ r_2^{Xi,p} \\ \vdots \\ r_{m_p}^{Xi,p} \end{pmatrix}, \tag{3.10}$$

and the subvector $r_s^{Xi,p}$ has dimension k_s^p for $s = 1,\cdots,m_p$. The term c^{Xi} is a constant. Without loss of generality, we assume that the matrices $R^{Xi,pp}$ for $X = 1,2;$ $p = 1,2;$ and $i = 1,...,k_i^X$ are symmetric and positive definite.

Theorem 3.2 Two-team games with quadratic objective functions as defined by (3.8)-(3.10) with $R^{Xi,pp}$>0, admits a unique Noninferior Nash solution $\hat{u}^\xi = \begin{pmatrix} \hat{u}^{1,\xi} \\ \hat{u}^{2,\xi} \end{pmatrix}$ for the given weight vectors $\xi^1 = \left(\xi_1^1,\cdots,\xi_{m_1}^1\right)' \in W^1$ and $\xi^2 = \left(\xi_1^2,\cdots,\xi_{m_2}^2\right)' \in W^2$ if the matrix

$$R^{\xi} = \begin{pmatrix} R^{1\xi^1,11} & R^{1\xi^1,12} \\ R^{2\xi^2,21} & R^{2\xi^2,22} \end{pmatrix} \tag{3.11}$$

where

$$R^{X\xi^X,pq} = \sum_{i=1}^{m_X} \xi_i^X R^{Xi,pq} \quad , \tag{3.12}$$

is nonsingular. This Noninferior Nash solution is unique if the matrix defined by (3.11) is invertible, in which case it is given by

$$\hat{u}^{\xi} = -\left(R^{\xi}\right)^{-1} r^{\xi} \tag{3.13}$$

where $r^{\xi} = \begin{pmatrix} r^{X\xi^X,1} \\ r^{X\xi^X,2} \end{pmatrix}$, and

$$r^{X\xi^X,p} = \sum_{i=1}^{m_X} \xi_i^X r^{Xi,p} . \tag{3.14}$$

Proof. For each team $X \in \{1,2\}$ the noninferior set of solutions can be determined by minimizing the objective function:

$$\begin{aligned} J^{X\xi^X}(u^1,u^2) &= \sum_{i=1}^{m_X} \xi_i^X J_i^X(u^1,u^2) \\ &= \frac{1}{2}\left((u^1)' \quad (u^2)'\right)\begin{pmatrix} R^{X\xi^X,11} & R^{X\xi^X,12} \\ R^{X\xi^X,21} & R^{X\xi^X,22} \end{pmatrix}\begin{pmatrix} u^1 \\ u^2 \end{pmatrix} + \left((r^{X\xi^X,1})' \quad (r^{X\xi^X,2})'\right)\begin{pmatrix} u^1 \\ u^2 \end{pmatrix} + c^{X\xi^X} \end{aligned} \quad . \tag{3.15}$$

We note that in (3.15) $R^{X\xi^X,21} = \left(R^{X\xi^X,12}\right)'$. The Nash solution is now easily derived by setting $\nabla_{u^X} J^{X\xi^X}(u^1,u^2) = 0$ for $X = 1,2$, i.e.,

$$\frac{\partial J^{X\xi^X}(u^X,u^{\bar{X}})}{\partial u^X} = R^{X\xi^X,XX}u^X + R^{X\xi^X,X\bar{X}}u^{\bar{X}} + r^{X\xi^X,X} = 0 \tag{3.16}$$

which yield the linear matrix equation:

$$\begin{pmatrix} R^{1\xi^1,11} & R^{1\xi^1,12} \\ R^{2\xi^2,21} & R^{2\xi^2,22} \end{pmatrix}\begin{pmatrix} \hat{u}^{1,\xi} \\ \hat{u}^{2,\xi} \end{pmatrix} = -\begin{pmatrix} r^{1\xi^1,1} \\ r^{2\xi^2,2} \end{pmatrix} \tag{3.17}$$

Therefore, the necessary and sufficient conditions for the solution given in (3.17) to be an NNS are as follows. For each pair of weight vectors $\{\xi^1,\xi^2\} \in W^1 \times W^2$:

(a) The matrices $R^{1\xi^1,11}$ and $R^{2\xi^2,22}$ are positive definite, (3.18a)

(b) The matrix $\hat{R}^{\xi} = \begin{pmatrix} R^{1\xi^1,11} & R^{1\xi^1,12} \\ R^{2\xi^2,21} & R^{2\xi^2,22} \end{pmatrix}$ is nonsingular. (3.18b)

We note that the matrix in (b) is not necessarily symmetric. □

Example 3.1 Consider two households, each consisting of a husband and wife, in conflict. The team members of household 1 are H1 and W1 and the team members of household 2 are H2 and W2. Let the decision variables of H1 and W1 be $u_1^1 = x$ and $u_2^1 = y$ respectively and the decision variables of household 2 be $u_1^2 = u$ and $u_2^2 = v$ respectively as shown in Table 3.3.

Table 3.3 A quadratic two-team example

Teams	Team Composition	Decision Variables	Objective Function (Minimize)
Household 1	DM_1^1 (H1)	x	$J_{H1} = \frac{1}{2}[(x-v)^2 + (y-u)^2]$
	DM_2^1 (W1)	y	$J_{W1} = \frac{1}{2}[(x-u-1)^2 + (y-v)^2]$
Household 2	DM_1^2 (H2)	u	$J_{H2} = \frac{1}{2}[v^2 + (u-x)^2]$
	DM_2^2 (W2)	v	$J_{W2} = \frac{1}{2}[(v-1)^2 + (u-y)^2]$

Consider the weight vectors $\xi^1 = \begin{pmatrix} \xi_1^1 \\ \xi_2^1 \end{pmatrix} = \begin{pmatrix} \alpha \\ 1-\alpha \end{pmatrix}$ with ($0 \le \alpha \le 1$) for household 1, and $\xi^2 = \begin{pmatrix} \xi_1^2 \\ \xi_2^2 \end{pmatrix} = \begin{pmatrix} \beta \\ 1-\beta \end{pmatrix}$ with ($0 \le \beta \le 1$) for household 2. With these parameters, equation (3.17) can be written as:

$$\begin{pmatrix} 1 & 0 & -(1-\alpha) & -\alpha \\ 0 & 1 & -\alpha & -(1-\alpha) \\ -\beta & -(1-\beta) & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ u \\ v \end{pmatrix} = \begin{pmatrix} 1-\alpha \\ 0 \\ 0 \\ 1-\beta \end{pmatrix}. \tag{3.19}$$

The above matrix satisfies the necessary and sufficient conditions (3.18) for existence of the NNS provided its determinant is not equal to zero. That is: Det ($\hat{R}^{\xi}$)= $[1-\alpha(1-\beta)-\beta(1-\alpha)] \neq 0$. Under this condition, (3.19) can be solved for the NNS to get:

$$\begin{cases} \hat{x}^{\xi^1} = (1-\alpha)\left[\dfrac{(1-\beta)^2(1-\alpha)+\beta(1-\alpha\beta)}{1-\alpha(1-\beta)-\beta(1-\alpha)}\right]+(1-\alpha\beta) \\ \hat{y}^{\xi^1} = \alpha\left[\dfrac{(1-\beta)^2(1-\alpha)+\beta(1-\alpha\beta)}{1-\alpha(1-\beta)-\beta(1-\alpha)}\right]+(1-\alpha)(1-\beta) \\ \hat{u}^{\xi^2} = \dfrac{(1-\beta)^2(1-\alpha)+\beta(1-\alpha\beta)}{1-\alpha(1-\beta)-\beta(1-\alpha)} \\ \hat{v}^{\xi^2} = (1-\beta) \end{cases} \tag{3.20}$$

Clearly, in this example the NNS is not unique and depends on the values of α and β. Table 3.4 illustrates several NNSs corresponding to the several combinations of values for α and β. For $0 \le \alpha, \beta \le 1$, contour curves of the determinant Det $(\hat{R}^{\xi})$ are shown in Figure 3.6. It is clear that the value of Det $(\hat{R}^{\xi})$ equal to zero is at only two points: A ($\alpha = 0, \beta = 1$) and B ($\alpha = 1, \beta = 0$). At point B, we still can find such a Noninferior Nash solution as given by

$$\left(\hat{x}^{\xi^1}, \hat{y}^{\xi^1}, \hat{u}^{\xi^2}, \hat{v}^{\xi^2}\right)' = (1, 1, 1, 1)' .$$

Therefore, The existence conditions for NNS are not satisfied at only point A.

Table 3.4 Several possible Noninferior Nash solutions for different values of α and β

(α, β)	$\hat{x}$	$\hat{y}$	$\hat{u}$	$\hat{v}$	$\hat{J}_{H1}$	$\hat{J}_{W1}$	$\hat{J}_{H2}$	$\hat{J}_{W2}$
(0.5,0.5)	1.2500	0.7500	1.0000	0.5000	0.3125	0.3125	0.1563	0.1563
(0.6,0.2)	1.2727	0.9091	0.9818	0.8000	0.1144	0.2574	0.3623	0.0226
(0.1,0.8)	3.5923	0.4769	2.9692	0.2000	8.8597	0.1094	0.2141	3.4258
(0.2,0.2)	1.7882	0.8471	1.0353	0.8000	0.5060	0.0316	0.6035	0.0377
(0.8,0.8)	0.4471	0.3882	0.4353	0.2000	0.0316	0.5060	0.0201	0.3211
(0.6,1)	0.6667	0.4000	0.6667	0	0.2578	0.5800	0	0.5356
(1,0)	1	1	1	1	0	0.5	0.5	0

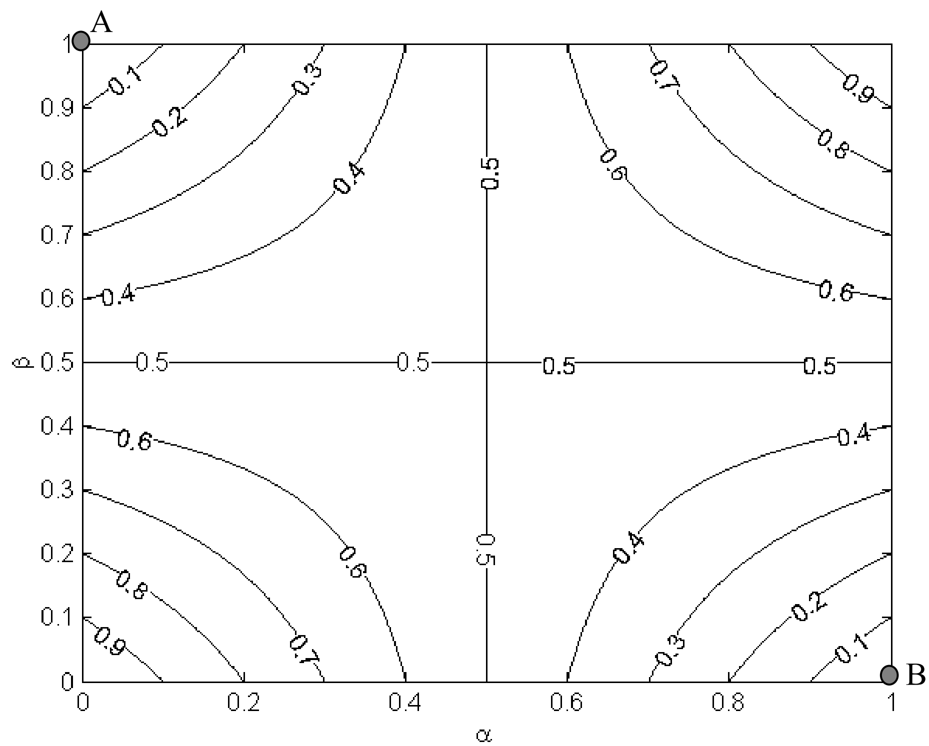


Figure 3.6 Contour curve of $\mathrm{Det}(\hat{R}^{\xi}) = 1 - \alpha(1-\beta) - \beta(1-\alpha)$ in Example 3.1

3.3 CONTINUOUS-TIME INFINITE DYNAMIC MTGS

In this section, we mainly concern with a special class of dynamic multi-team games, i.e., linear quadratic differential games where the system is linear and the payoff functions are quadratic functions of states and controls.

For simplicity, we still consider a two-team dynamic system with m_1 decision-makers in team 1 and m_2 decision-makers in team 2. Let the control vectors u_i^X for the members $i = 1, \cdots, m_X$ in each team be grouped into a team control vector $u^X = (u_1^X, u_2^X, ..., u_{m_X}^X)$ that belongs to compact and convex admissible sets of the form $U^X = U_1^X \times U_2^X \times \cdots \times U_{m_X}^X$ where X =1, 2.

The overall linear dynamic system is described by the following state equations:

$$\dot{x}(t) = Ax(t) + \sum_{j=1}^{m_1} B_j^1 u_j^1 + \sum_{j=1}^{m_2} B_j^2 u_j^2 \qquad\qquad x(t_0) = x_0 \tag{3.21}$$

where state variable $x(t) \in \mathbb{R}^n$, $A(\cdot)$ and $B_j^X(\cdot)$ $(X = 1,2; j = 1, \cdots m_X)$ are matrices of appropriate dimensions. x_0 is the initial state. For the i^{th} decision-maker in team X, the cost function is given by

$$J_i^X(x,t,u^1,u^2) = \frac{1}{2}x(t_f)^T S_{if}^X x(t_f) + \frac{1}{2}\int_{t_0}^{t_f}\left(x^T Q_i^X x + \sum_{j=1}^{m_1}\left(u_j^1\right)^T R_{ij}^{X1} u_j^1 + \sum_{j=1}^{m_2}\left(u_j^2\right)^T R_{ij}^{X2} u_j^2 \right)dt \tag{3.22}$$

where $S_{if}^X(\cdot), Q_i^X(\cdot)$, $R_{ij}^{X1}(\cdot)$ and $R_{ij}^{X2}(\cdot)$ are matrices of appropriate dimensions, defined on $[t_0, t_f]$, and with continuous entries. Furthermore, $S_{if}^X(\cdot)$ and $Q_i^X(\cdot)$ are symmetric, and $R_{ij}^{X1}(\cdot)$ and $R_{ij}^{X2}(\cdot)$ are positive definite over $[t_0, t_f]$.

In optimal control problem [16], open-loop controls, functions of time, are the optimal controls for a trajectory through a specified initial state, and closed-loop (or feedback) controls, are the optimal controls which are the functions of the state and time everywhere. It is well known that, in deterministic optimal control problems, the open-loop solution can be generated from the closed-loop control simply by integrating the state equation forward from the given initial points. Alternatively, a closed-loop control can be generated by successively solving the open-loop problem for each initial point. Therefore, in a deterministic optimal control problem, open-loop controls and closed-loop controls are only the different ways of describing the same result. However, this conclusion is not true for the dynamic games any longer [8]. Generally speaking, open-loop Nash controls are not identical to closed-loop Nash controls in a dynamic game. One reason for the difference between the open-loop controls and the closed-loop controls is that, for the closed-loop controls, several control sequences are eliminated from consideration at the initial instant (t=0) by the assumption that, for optimizing the transition from the current states to the remaining part of trajectory based on the current states, DMs always attempt to use the same game rules. Thus, it is not always safe to apply this assumption to the nonzero-sum games. In addition, the closed-loop controls do not always give us the better results than the open-loop controls [8].

3.3.1 Open-loop noninferior Nash control

For the decision-makers in team X, they minimize the following aggregated objective function under the given weight vector $\xi^X \in W^X$ (where W^X is given by (3.6)):

$$J^X(x,t,u^1,u^2;\xi^X) = \sum_{i=1}^{m_X} \xi_i^X J_i^X$$

$$= \frac{1}{2} x(t_f)^T \left(\sum_{i=1}^{m_X} \xi_i^X S_{if}^X\right) x(t_f) +$$

$$\frac{1}{2}\int_{t_0}^{t_f} \left(x^T \left(\sum_{i=1}^{m_X} \xi_i^X Q_i^X\right) x + \sum_{j=1}^{m_1} \left(u_j^1\right)^T \sum_{i=1}^{m_X} \left(\xi_i^X R_{ij}^{X1}\right) u_j^1 + \sum_{j=1}^{m_2} \left(u_j^2\right)^T \sum_{i=1}^{m_X} \left(\xi_i^X R_{ij}^{X2}\right) u_j^2 \right) dt \qquad (3.23)$$

Let $S_f^X = \sum_{i=1}^{m_X} \xi_i^X S_{if}^X$, $Q^X = \sum_{i=1}^{m_X} \xi_i^X Q_i^X$, $R_j^{X1} = \sum_{i=1}^{m_X} \xi_i^X R_{ij}^{X1}$ and $R_j^{X2} = \sum_{i=1}^{m_X} \xi_i^X R_{ij}^{X2}$. The expression (3.23) can be rewritten as:

$$J^X(x,t,u^1,u^2;\xi^X) = \frac{1}{2} x(t_f)^T S_f^X x(t_f) +$$

$$\frac{1}{2}\int_{t_0}^{t_f} \left(x^T Q^X x + \sum_{j=1}^{m_1} \left(u_j^1\right)^T R_j^{X1} u_j^1 + \sum_{j=1}^{m_2} \left(u_j^2\right)^T R_j^{X2} u_j^2 \right) dt \ . \qquad (3.24)$$

In the view of restriction $S_{if}^X(\cdot)$ and $Q_i^X(\cdot) \geq 0$, $J^X(x,t,u^1,u^2;\xi^X)$ is a strictly convex function of u^X for all permissible control functions $u^{\bar{X}}$. According to Theorem A-1 [12], it is a sufficient condition and every solution set of the first order conditions provides an open-loop NNS with the given ξ^1 and ξ^2. Next, we will derive the analytical expressions for the open-loop NNS.

The Hamiltonian is given by

$$H^X(x,t,u^1,u^2;\lambda^X) = \frac{1}{2}\left(x^T Q^X x + \sum_{j=1}^{m_1} \left(u_j^1\right)^T R_j^{X1} u_j^1 + \sum_{j=1}^{m_2} \left(u_j^2\right)^T R_j^{X2} u_j^2 \right)$$

$$+ (\lambda^X)^T \left(Ax + \sum_{j=1}^{m_1} B_j^1 u_j^1 + \sum_{j=1}^{m_2} B_j^2 u_j^2 \right) \qquad (3.25)$$

Let $R^{X1} = \begin{bmatrix} R_1^{X1} & & & \\ & R_2^{X1} & & \\ & & \ddots & \\ & & & R_{m_X}^{X1} \end{bmatrix}$, $R^{X2} = \begin{bmatrix} R_1^{X2} & & & \\ & R_2^{X2} & & \\ & & \ddots & \\ & & & R_{m_X}^{X2} \end{bmatrix}$ and $B^X = \begin{bmatrix} B_1^X \\ B_2^X \\ \cdots \\ B_{m_X}^X \end{bmatrix}$. The Hamiltonian can be rewritten as

$$H^X(x,t,u^1,u^2;\lambda^X) = \frac{1}{2}\left(x^T Q^X x + \left(u^1\right)^T R^{X1}u^1 + \left(u^2\right)^T R^{X2}u^2\right) + (\lambda^X)^T\left(Ax + B^1u^1 + B^2u^2\right) \quad (3.26)$$

Now, we can write out the necessary condition for NNS as follows:

$$\begin{cases} \dfrac{\partial H^X}{\partial u^X} = R^{XX}u^X + \left(B^X\right)^T \lambda^X = 0 & X = 1,2 \\ \dot{\lambda}^X = -\dfrac{\partial H^X}{\partial x} = -(Q^X x + A^T\lambda^X) & \lambda^X(t_f) = \dfrac{\partial\left(\frac{1}{2}x(t_f)^T S_f^X x(t_f)\right)}{\partial x(t_f)} \\ \dot{x} = Ax + B^1u^1 + B^2u^2 & x(t_0) = x_0 \end{cases} \quad (3.27)$$

Furthermore, we can obtain the NNS $\left(\hat{u}^{1,o}, \hat{u}^{2,o}\right)$ under the given weight vector $\xi^X \in W^X$ as

$$\hat{u}^{X,o} = -\left(R^{XX}(\xi^X)\right)^{-1}\left(B^X(t)\right)^T \lambda^X(t) \qquad X=1,2. \quad (3.28)$$

The costate equation is given by

$$\dot{\lambda}^X = -(Q^X x + A^T\lambda^X) \qquad \lambda^X(t_f) = S_f^X x(t_f),\ X=1,2. \quad (3.29)$$

The optimal trajectory $\left\{\hat{x}(t), t \in [t_0, t_f]\right\}$ can be obtained as:

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + \sum_{X=1}^{2} B^X \hat{u}^{X,o} \\ &= A\hat{x} - \sum_{X=1}^{2} B^X(t)\left(R^{XX}(\xi^X)\right)^{-1}\left(B^X(t)\right)^T \lambda^X(t) \qquad \hat{x}(t_0) = x_0. \end{aligned} \quad (3.30)$$

The set of differential equations constitutes a two-point boundary value problem, the solution of which can be written, without loss of generality, as

$$\lambda^X(t) = M^X(t)\hat{x}(t) \qquad X = 1,2;\ t \in [t_0, t_f]. \tag{3.31}$$

Substituting (3.31) into the costate equation (3.29), we got

$$M^X(t)\dot{\hat{x}}(t) + \dot{M}^X(t)\hat{x}(t) = -Q^X\hat{x}(t) - A^T M^X(t)\hat{x}(t)\ . \tag{3.32}$$

From (3.30), we have

$$M^X(t)\left(A\hat{x} - \sum_{i=1}^{2} B^i(t)\left(R^{ii}(\xi^i)\right)^{-1}\left(B^i(t)\right)^T M^i\hat{x}(t)\right) + \dot{M}^X(t)\hat{x}(t) = -Q^X\hat{x}(t) - A^T M^X(t)\hat{x}(t)\ .$$

As a result, we got the coupled matrix Riccati differential equations as follows:

$$\begin{cases} \dot{M}^X(t) + M^X(t)A + A^T M^X(t) + Q^X - M^X(t)\left(\sum_{i=1}^{2} B^i(t)\left(R^{ii}(\xi^i)\right)^{-1}\left(B^i(t)\right)^T M^i(t)\right) = 0 \\ M^X(t_f) = S_f^X \qquad\qquad X = 1,2 \end{cases} \tag{3.33}$$

The unique open-loop NNS under the given weight vector $\xi^X \in W^X$ is given by

$$\hat{u}^{X,o}(t,\xi^X) = -\left(R^{XX}(\xi^X)\right)^{-1}\left(B^X(t)\right)^T M^X(t)\Phi(t,t_0)x_0 \qquad X = 1,2 \tag{3.34}$$

where $\Phi(t,t_0)$ is the state transition matrix of the system satisfying:

$$\dot{\Phi}(t,t_0) = \left(A - \sum_{i=1}^{2} B^i(t)\left(R^{ii}(\xi^i)\right)^{-1}\left(B^i(t)\right)^T M^i(t)\right)\Phi(t,t_0) \qquad \Phi(t,t) = I\ . \tag{3.35}$$

3.3.2 Closed-loop noninferior Nash control

We use dynamic programming method to derive the closed-loop NNS for linear quadratic differential multi-team systems. In the expression (3.28), let $\lambda^X(t) = \dfrac{\partial \hat{J}^X(x,t)}{\partial \hat{x}}$ and we have

$$\hat{u}^X = -\left(R^{XX}(\xi^X)\right)^{-1}\left(B^X(t)\right)^T \frac{\partial \hat{J}^X(x,t)}{\partial \hat{x}}. \tag{3.36}$$

Suppose that $J^X(x,t)=\dfrac{1}{2}x^T S^X(t)x$, thus we got

$$\frac{\partial J^X(x,t)}{\partial x}=S^X(t)x=\lambda^X(t)\,, \tag{3.37a}$$

$$\frac{\partial J^X(x,t)}{\partial t}=\frac{1}{2}x^T\dot{S}^X(t)x\,. \tag{3.37b}$$

We also know that

$$\frac{\partial J^X(x,t)}{\partial t}=-\hat{H}^X(x,t,\hat{u}^1,\hat{u}^2,\frac{\partial\hat{J}^X}{\partial x}) \tag{3.38}$$

Substituting (3.37b), (3.26) and (3.28) into (3.38), we have

$$\begin{aligned}\frac{1}{2}x^T\dot{S}^X(t)x=-\frac{1}{2}&\left(x^TQ^Xx+\sum_{i=1}^{2}(-R^{ii}(\xi^i)^{-1}(B^i)^T\lambda^i)^T R^{Xi}(-R^{ii}(\xi^i)^{-1}(B^i)^T\lambda^i)\right)\\ &-(\lambda^X)^T\left(Ax+\sum_{i=1}^{2}B^i(-R^{ii}(\xi^i)^{-1}(B^i)^T\lambda^i)\right)\end{aligned} \tag{3.39}$$

Substituting (3.37a) into the above equation (3.39), we got

$$\begin{aligned}\frac{1}{2}x^T\dot{S}^X(t)x=-\frac{1}{2}x^TQ^Xx-\frac{1}{2}x^T&\left(\sum_{i=1}^{2}S^iB^iR^{ii}(\xi^i)^{-1}R^{Xi}R^{ii}(\xi^i)^{-1}(B^i)^TS^i)\right)x\\ -x^TS^XAx+x^T&\left((S^X)^T\sum_{i=1}^{2}B^iR^{ii}(\xi^i)^{-1}(B^i)^TS^i\right)x.\end{aligned} \tag{3.40}$$

Considering the symmetry of the matrix S^X, we can write it into

$$S^XA=\frac{1}{2}S^XA+\frac{1}{2}A^TS^X. \tag{3.41}$$

Replacing S^X in (3.40) using (3.41), we have the equation for $S^X(t)$ as:

$$\begin{cases} \dot{S}^X(t) = -Q^X - S^X A - A^T S^X - \left(\sum_{i=1}^{2} S^i B^i R^{ii}(\xi^i)^{-1} R^{Xi} R^{ii}(\xi^i)^{-1}(B^i)^T S^i \right) \\ \qquad\qquad + (S^X)^T \left(\sum_{i=1}^{2} B^i R^{ii}(\xi^i)^{-1}(B^i)^T S^i \right) + \left(\sum_{i=1}^{2} B^i R^{ii}(\xi^i)^{-1}(B^i)^T S^i \right) S^X \\ S^X(t_f) = S_f^X, \qquad\qquad X = 1,2. \end{cases} \tag{3.42}$$

The closed-loop NNS $\left(\hat{u}^{1,c}, \hat{u}^{2,c}\right)$ under the given weight vector $\xi^X \in W^X$ can be calculated as

$$\hat{u}^{X,c} = -\left(R^{XX}(\xi^X)\right)^{-1}\left(B^X\right)^T S^X(t)\hat{x}(t) \qquad X = 1,2\,. \tag{3.43}$$

The optimal state trajectory is calculated as:

$$\dot{\hat{x}} = \left(A - \sum_{X=1}^{2} B^X(t)\left(R^{XX}(\xi^X)\right)^{-1}\left(B^X(t)\right)^T S^X(t) \right)\hat{x} \qquad \hat{x}(t_0) = x_0\,. \tag{3.44}$$

3.4 CONCLUSIONS

In this section, we presented a new framework for optimizing extended complex systems that involve multiple teams of decision-makers. We developed a new solution concept, called the Noninferior Nash Strategy (NNS), which combines the properties of the cooperative noninferior (or Pareto) solution from team theory and the noncooperative Nash solution from game theory. Such a strategy insures cooperation within each team and competition among the various teams. We investigated the properties of the Noninferior Nash Strategy in matrix and static multi-team games and provided necessary conditions for its existence. We have shown that, in general, there is a set of Noninferior Nash Strategies. Therefore, how to select an appropriate Noninferior Nash Strategy is a critical issue. Several examples to illustrate the various solution concepts introduced in this section were also presented. We also presented the analytical expressions for open-loop and closed-loop Noninferior Nash controls to a class of linear quadratic differential multi-team games.

4.0 NONINFERIOR NASH STRATEGIES WITH A LEADER FOR EXTENDED COMPLEX SYSTEMS

As can be seen from the analysis in the previous chapter, how each team chooses a specific solution from a set of Noninferior Nash strategies (or how it chooses the weight vector ξ^X) is critical in determining the resulting NNS. While this choice can be done by a mutual (enforceable) agreement among all team members, in some cases there may exist a team Leader whose responsibility is to make that choice. Furthermore, the team Leader may use a different objective function as a criterion for making this choice. If all the team Leaders' objective functions depend on the control variables of all decision-makers, then a game situation will also exist among the team Leaders and the specific choices of noninferior Nash solutions will have to be made based on a game theoretic approach. This situation actually occurs in many real applications such as in cooperative teaming of autonomous entities such as unmanned aerial vehicles, robots, etc., in the control of ancillary services in future energy distribution grids, as well as in the management of computer communication networks.

We have shown that, in general, there is a set of Noninferior Nash Strategies for multi-team systems. How to determine the NNS to implement is a critical issue. In this chapter, we involve the team Leaders' objective functions as a mechanism for selecting a strategy from this set. We call this strategy the Noninferior Nash Strategy with a Leader (NNSL). In section 4.1, we present the definition of NNSL. In section 4.2, we present a simple microeconomic example to illustrate the properties of NNSL. In section 4.3, we apply the NNSL to the routing control problems in two-node parallel-link network system. In section 4.4, we give some concluding remarks.

4.1 NONINFERIOR NASH STRATEGIES WITH A LEADER

In this section, we will address the issue of selecting a specific solution from the set of NNS in the context of continuous games where each team Leader has the task of choosing its team weight vector ξ^X. As before, we will consider only the case of two teams. We will assume that the Leader of team X chooses $\xi^X \in S^X \subseteq W^X$ so as to minimize an objective function $J_L^X(\hat{u}^{X,\xi}) = \hat{J}_L^X(\xi^1, \xi^2)$. Note that the form of $J_L^X(\cdot)$ for the team leader may be different from those of decision-makers in the corresponding team. The subset S^X corresponds to the values of parameters $\xi^X \in W^X$ for which NNS solutions exist. Within this new structure, we give the following definition of the Noninferior Nash Strategy with a Leader (NNSL).

Definition 4.1 The pair of strategies $\{\hat{u}^{1,\hat{\xi}^1}, \hat{u}^{2,\hat{\xi}^2}\}$ is NNSL if there exists a pair $\{\hat{\xi}^1, \hat{\xi}^2\}$ such that:

$$\begin{aligned} &\hat{J}_L^1(\hat{\xi}^1, \hat{\xi}^2) \le \hat{J}_L^1(\xi^1, \hat{\xi}^2) \quad \text{for all } \xi^1 \in S^1 \\ &\hat{J}_L^2(\hat{\xi}^1, \hat{\xi}^2) \le \hat{J}_L^2(\hat{\xi}^1, \xi^2) \quad \text{for all } \xi^2 \in S^2 \end{aligned} . \tag{4.1}$$

In other words, the pair $\{\hat{u}^{1,\hat{\xi}^1}, \hat{u}^{2,\hat{\xi}^2}\}$ is an NNSL if the pair of weight vectors $\{\hat{\xi}^1, \hat{\xi}^2\}$ results in a Nash equilibrium between the objective functions of the two leaders.

As we can see from the definition above, when each Leader's objective function is expressed in terms of the weight vectors, the resulting functions may end up depending on the weight vector of that Leader's team only. In other words, on the higher level, a noncooperative game exists between two leaders who select the appropriate control variables $\xi^X \in S^X$ in order to improve their own objectives. Nash strategy is a reasonable solution to such a game. Since it is not easy to obtain the analytical expression of NNSL to such a complicated hierarchical decision-making system, in the following sections, we present several examples to illustrate the properties of NNSL.

4.2 NNSL TO MICROECONOMICS PROBLEMS

One particular situation will occur in a two-team system where one team has only one decision-maker as will be illustrated in the following simple example from duopoly microeconomics [24] and [25]. In this case, each Leader will be faced with a simple optimization problem rather than a game with the other Leader. In this example, we consider one team with two decision-makers ($m_1 = 2$) and a Leader and the other team with only one decision-maker ($m_2 = 1$). In the case of a team with one member, that decision-maker will also be the Leader, and the NNS for that team will be a function of only the other team's weight vector.

Consider two firms A and B that produce and sell the same product in a competitive market. Firm A has two divisions (for example West Coast and East Coast divisions) A_1 and A_2 each having an independent decision-maker. Firm B has only one division and one decision-maker. Table 4.1 describes the production variables and profit functions for each of the three divisions involved in the market.

Table 4.1 Description of market competition example

Teams	Division	Decision -Maker	Production Output	Production Constraints	Production Cost	Profit Function
Firm A	Division A_1	DA_1	x_1	$0 \le x_1 \le 50$	$a_1 x_1^{\,2}$ $(a_1 > 0)$	$J_1^1 = px_1 - a_1 x_1^{\,2}$
	Division A_2	DA_2	x_2	$0 \le x_2 \le 50$	$a_2 x_2$ $(a_2 > 0)$	$J_2^1 = px_2 - a_2 x_2$
Firm B		DB	y	$0 \le y \le 50$	$by^{\,2}$ $(b > 0)$	$J^2 = py - by^2$

Assume that the product is sold at a market price p which is determined based on a demand function of the form [26]:

$$p = p_0 - (x_1 + x_2 + y) \quad (\, p_0 > 0\,) \tag{4.2}$$

The objective of each decision-maker is to maximize the profits of his/her division. We note that since Firm B has only one decision variable, its weight vector will be a fixed scalar $\hat{\xi}^2 = 1$. The Leader of Firm A, however, has to decide on a solution in the noninferior set of its two divisions. The noninferior set for Firm A is determined by considering an objective function of the form $J^1 = \alpha J_1^1 + (1-\alpha) J_2^1$ where $0 \le \alpha \le 1$. Using the results in (3.17) with $\xi^1 = \begin{pmatrix} \alpha \\ 1-\alpha \end{pmatrix}$ and $\xi^2 = 1$ we have

$$\begin{pmatrix} 2\alpha(1+a_1) & 1 & \alpha \\ 1 & 2(1-\alpha) & (1-\alpha) \\ 1 & 1 & 2(1+b) \end{pmatrix} \begin{pmatrix} \hat{x}_1^{\xi^1} \\ \hat{x}_2^{\xi^1} \\ y^{\xi^2} \end{pmatrix} = \begin{pmatrix} \alpha p_0 \\ (1-\alpha)(p_0 - a_2) \\ p_0 \end{pmatrix}. \tag{4.3}$$

The necessary and sufficient conditions (3.18a) and (3.18b) for an NNS to exist are satisfied provided:

$$\text{a) } \left(\frac{1}{2}-\frac{1}{2}\sqrt{\frac{a_1}{1+a_1}}\right) < \alpha < \left(\frac{1}{2}+\frac{1}{2}\sqrt{\frac{a_1}{1+a_1}}\right), \text{ and} \tag{4.4a}$$

$$\text{b) } \operatorname{Det}(\hat{R}^{\xi}) = 2\alpha(1-\alpha)(4a_1b+3a_1+4b+2)-(2b+1) \neq 0\,. \tag{4.4b}$$

Under these conditions, we get the following NNS solutions:

$$\begin{cases} \hat{x}_1(\alpha) = \dfrac{v_1\alpha^2+v_2\alpha+v_3}{2\alpha(1-\alpha)(4a_1b+3a_1+4b+2)-(2b+1)} \\ \hat{x}_2(\alpha) = \dfrac{u_1\alpha^2+u_2\alpha}{2\alpha(1-\alpha)(4a_1b+3a_1+4b+2)-(2b+1)} \\ \hat{y}(\alpha) = \dfrac{h_1\alpha^2+h_2\alpha+h_3}{2\alpha(1-\alpha)(4a_1b+3a_1+4b+2)-(2b+1)} \end{cases} \tag{4.5}$$

where

$v_1 = a_2 - 2(2b+1)p_0, \quad v_2 = -a_2(2b+3)+3(2b+1)p_0, \quad v_3 = (-1-2b)p_0 + 2a_2(1+b)$,

$u_1 = (3+a_1+b+a_1b)a_2 - 2(1+a_1)(1+2b)p_0, \quad u_2 = -(3+a_1+b+a_1b)a_2 + (2a_1+1)(2b+1)p_0$,

$h_1 = -2(1+a_1)a_2 - 2a_1p_0, \quad h_2 = (2a_1+3)a_2 + 2a_1p_0, \quad h_3 = -a_2$.

As we can see from (4.5), the Noninferior Nash strategies are functions of the weight parameter α provided α satisfies conditions (4.4). Now, let us suppose that the Leader (or CEO) of Firm A wants to choose a weight α so as to maximize his firm's market share. That is, he wishes to maximize the objective function:

$$J_L^1 = x_1 + x_2 \tag{4.6}$$

Considering the results in (4.5), the Leader objective function (4.6) can now be expressed as

$$\hat{J}_L^1(\alpha) = \frac{(v_1+u_1)\alpha^2+(v_2+u_2)\alpha+v_3}{2\alpha(1-\alpha)(4a_1b+3a_1+4b+2)-(2b+1)} \tag{4.7}$$

Since maximizing this function analytically with respect to α is not practical, we will illustrate the results using the following numerical values for the various parameters in the problem. Let $a_1 = 1$, $a_2 = 50$, $b = 0.8$ and $p_0 = 200$. For these values, conditions (4.4a-4.4b) are satisfied provided $0.15 \le \alpha \le 0.85$. Plots of $\hat{x}_1(\alpha)$, $\hat{x}_2(\alpha)$, $\hat{y}(\alpha)$, and $\hat{J}_L^1(\alpha) = \hat{x}_1(\alpha) + \hat{x}_2(\alpha)$ and plots of the resulting product price $\hat{p}(\alpha)$ and profits $\hat{J}_1^1(\alpha)$ and $\hat{J}_2^1(\alpha)$ for divisions A_1 and A_2, $\hat{J}_1^1(\alpha) + \hat{J}_2^1(\alpha)$ for Firm A, and $\hat{J}^2(\alpha)$ for Firm B, for values of α in this range are shown in Figures 4.1-4.3, respectively.

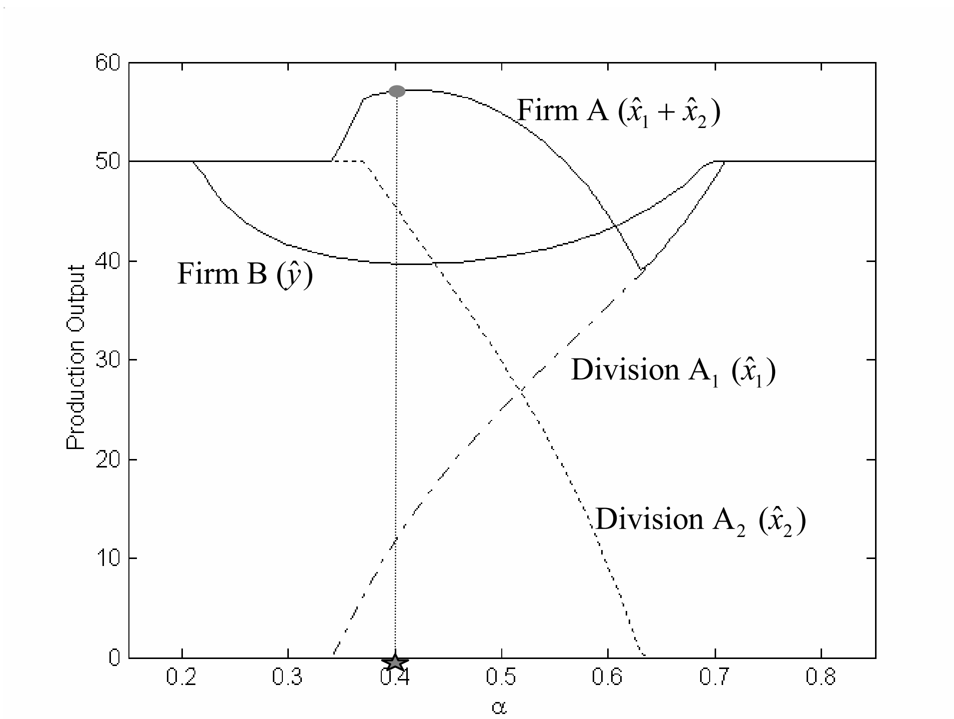


Figure 4.1 Production outputs $\hat{x}_1(\alpha)$, $\hat{x}_2(\alpha)$, $\hat{y}(\alpha)$, and $\hat{J}_L^1(\alpha) = \hat{x}_1(\alpha) + \hat{x}_2(\alpha)$ as functions of α

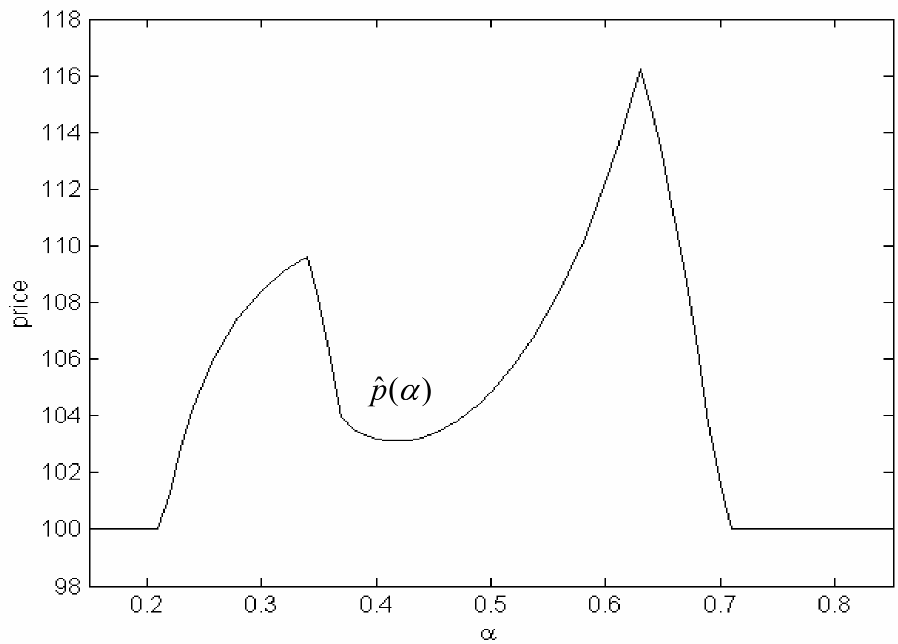


Figure 4.2 Price $\hat{p}(\alpha)$ as a function of α

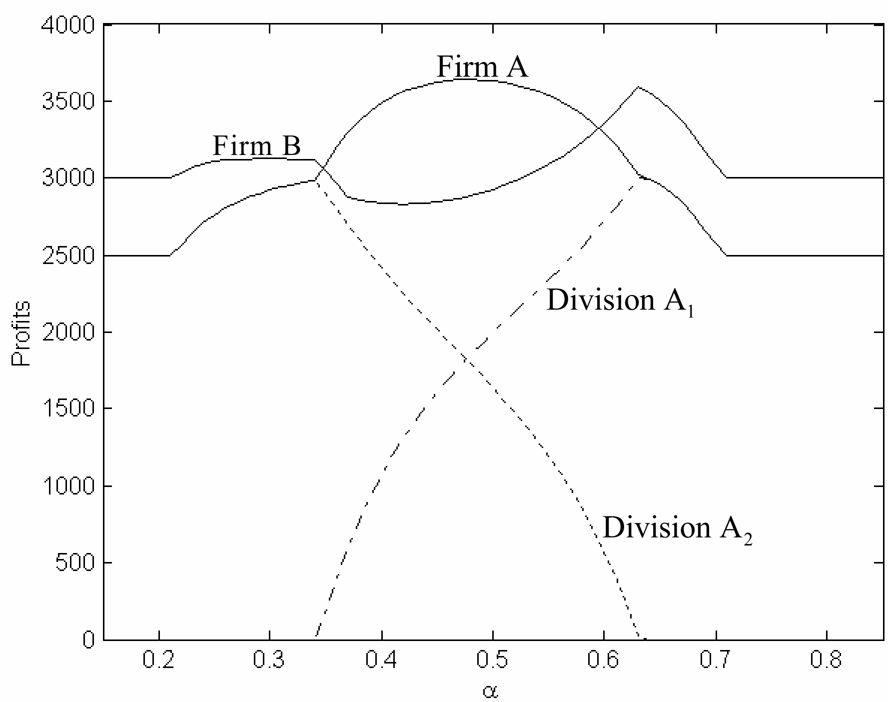


Figure 4.3 Profits of Firm A and Firm B as functions of α

It is clear from Figure 4.1 that the maximum of $\hat{J}_L^1(\alpha) = \hat{x}_1(\alpha) + \hat{x}_2(\alpha)$ occurs when $\hat{\alpha} = 0.4$, i.e., $\hat{\xi}^1 = \begin{pmatrix} 0.4 \\ 0.6 \end{pmatrix}$ and has a value of $\hat{J}_L^1(\hat{\alpha}) = 57.10$. In other words, with this choice of $\hat{\xi}^1$ the Leader (CEO) of Firm A is able to maximize his firm's total market share while at the same time keeping a Nash equilibrium between his and the other firm. We note that if the Leader's objective is to maximize the total profits of Firm A, the choice of $\hat{\alpha}$ should be at 0.5 instead of 0.4 as is clear from Figure 4.3.

Plots of the reaction sets for both divisions in Firm A, and for both firms, are shown in Figure 4.4 and Figure 4.5 respectively. We note that the reaction set of Firm B in actuality should be a three-dimensional plot representing the reaction y for all possible choices of x_1 and x_2. Since producing this plot would require considerable effort in visualizing three dimensional surfaces, we only produced a subset of this plot, shown in Figure 4.5, representing the reaction y for all possible choices of $x_1 + x_2$. From this plot, it is clear that all the NNSs are concentrated in a small region bounded in the range $35 \leq y \leq 50$ and $40 \leq x_1 + x_2 \leq 60$. We zoom in on this region in Figure 4.6.

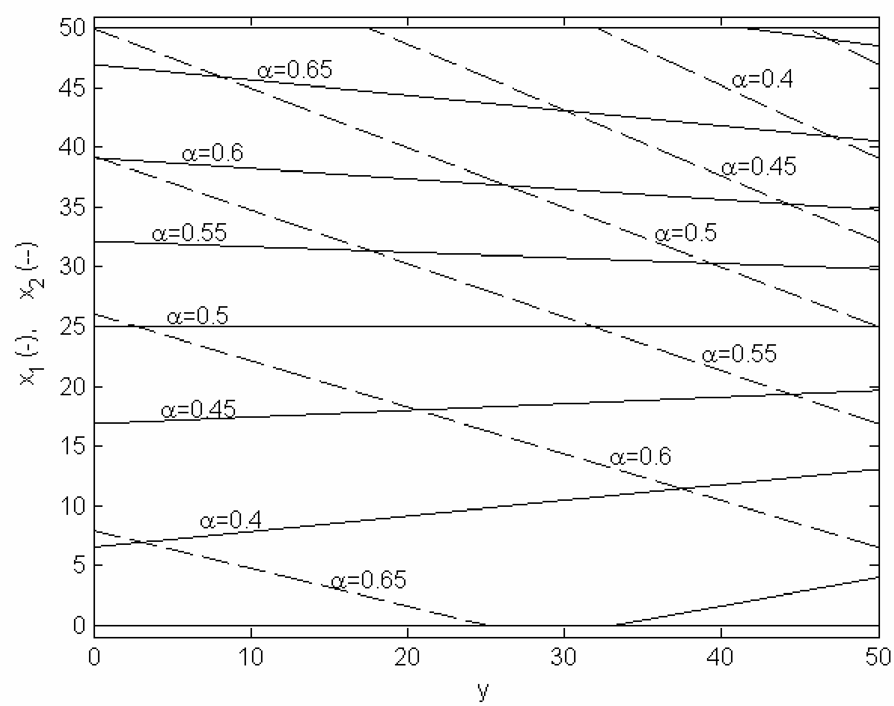


Figure 4.4 Reaction sets of Division A_1 and Division A_2 of Firm A

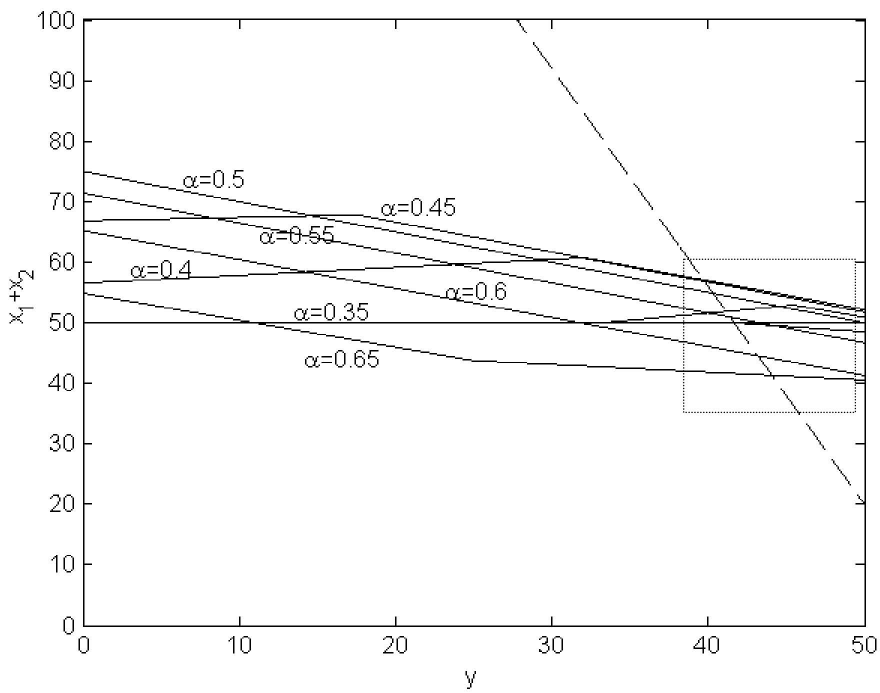


Figure 4.5 Reaction sets of Firm A and Firm B

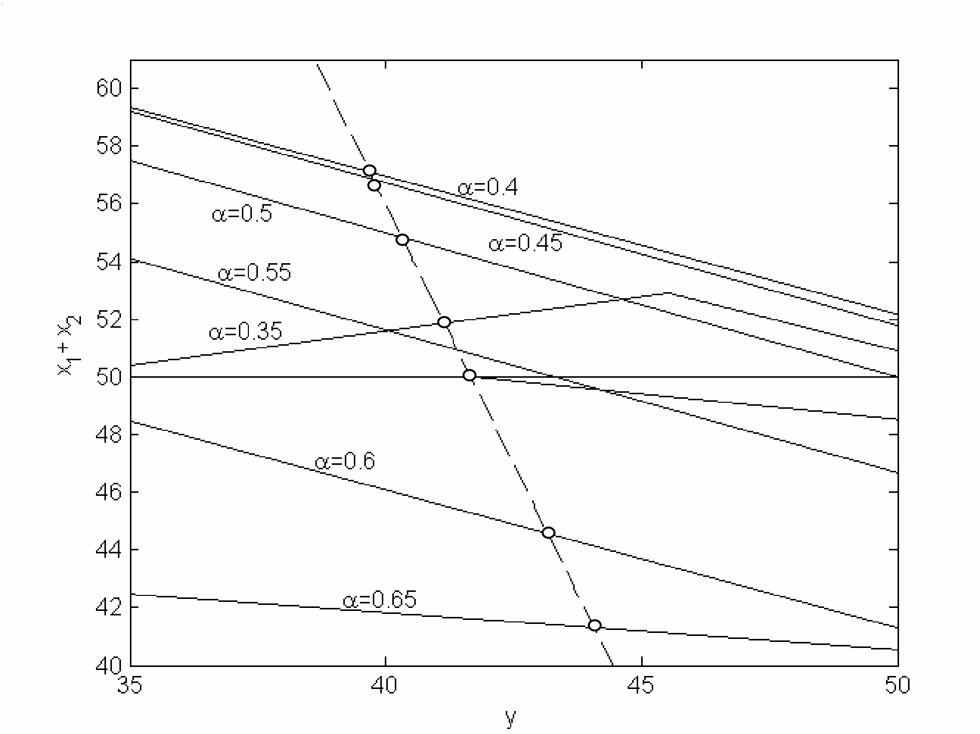


Figure 4.6 Noninferior Nash solutions (marked as circles) for different values of α

4.3 NNSL TO ROUTING CONTROL PROBLEMS

The problem of routing is encountered in all and every network shared by a large number of users. It is necessary to multiplex the resources of communication traffic in order to assign dedicated links of sufficient capacity to all the users to meet their needs. Traditionally, such networks are designed and built as a single entity with a single administration and objective under the assumption that users are passive and would cooperate to enhance the overall performance of the network. In modern communication networking, however, this assumption of a single administration is no longer valid since the users now have various, even contradictory, performance measures and demands. One possible way of managing such a network is to let the individual users compete with each other in a way that allows each of them to reach its optimal working state. In such a situation, users may change their control strategies based on the state of the network. A change in the control by one user is likely to affect other users' performance and cause them to change their control strategies as well, resulting in a dynamic system. At the end, the outcome of the network is heavily dependent on the actions taken by all the users, and consequently the problem for each user to determine its optimum control actions, is be best analyzed within the framework of game theory.

The literature on the analysis of equilibria in competitive routing control problems using game theory is very rich. Routing problems in communication networks shared by selfish users are modeled as noncooperative games in [27],[28],[29],[30] and [31] or as noncooperative repeated games in [32]. The concept of a Nash equilibrium, a main concern in [27],[28],[29],[30] and [31], ensures that no user find it beneficial to change its behavior unilaterally. Conditions for the existence and uniqueness of Nash equilibria are presented based on various types of cost functions for the users, such as polynomial link holding functions [27], utility functions in the form of “throughput/delay” [28], utility function in the form of “throughput-delay” [29], communication quality functions [30], and average delay functions [31],[32]. However, Nash equilibria are generically inefficient and exhibit suboptimal network performance. This deficiency can be overcome with the intervention of a network agent, say a network manager or a team leader. Stackelberg strategies are applied to address this issue [33],[34]. Considering a network manager who acts as a Stackelberg leader and controls a portion of the network flow, Korilis, *et al*. [33] derived necessary and sufficient conditions for the existence of a maximally strategy for manager to drive the system into a global optimum. Note that the leader considered in [33] is a special user in the system, and hence the problem is not formulated within a hierarchical structure. Basar, *et al*. [34] introduced a hierarchical network between a Stackelberg leader, who sets the price per unit of bandwidth, and multiple Nash followers, who decide on their flow rates. The leader’s objective is to maximize the total revenue it collects and the followers choose their levels of flow so as to maximizing an objective function that represents a tradeoff between the disutility of the payment to the leader and congestion costs on the link they use. They observed that the revenue-incentive for the network increases the available capacity (or decreases the delay) in the network in proportion to the number of users in the network. In [33],[34], however, only one team leader is considered.

In a network with more complicated organization, control may be shared by competing teams of users rather than single users. Teams are groups of users that share a common objective. With this structure, it is possible to envision the existence of a leader (or manager) for each team, whose function is to centralize all decisions for that team. Each type of entity can be considered as a user. For example, a set of different companies, each with different classes of traffics such as data, audio, image or video, each class of traffic controlled by a user, in the same neighborhood use wireless local area networks and share the same internet resource to send their traffics. The network manager (or team leader) for each company attempts to optimize the performance of all the traffics sent from his company. Obviously, team leaders usually have no choice but compete with each other to try to gain their own users over the network. One natural way of managing such a resource is allowing the users belonging to identical team leader to cooperate with each other and letting those team leaders compete with each other and settle to an equilibrium where each of them reaches its optimum operation point. The diagram of such a hierarchical structure is given in Figure 4.7. A similar structure is considered in [35]. However, each team leader only considers the average performance of all his entities (or users) as his objective in [35]. Of practical interest, each user in a team may have its own objective to meet. Team leaders may also have their own objectives different from those of their users.

In previous section, when using NNSL, we note that an optimization problem rather than a game is considered on the higher level. In this section, we will apply NNSL to a simple network consisting of a common source node and a common destination node interconnected by a number of parallel links. This network is shared by several teams of users and each team has a Team Leader (TL) to coordinate the actions of his team members. The users within each team cooperate for the benefit of their team. The teams, on the other hand, compete among themselves in order to achieve an objective that relates to the overall performance of the network. Our main goal is to devise a control scheme for the modern parallel-link networks and investigate the effectiveness of NNSL in the problem of splitting load among those link resources, i.e., routing problem.

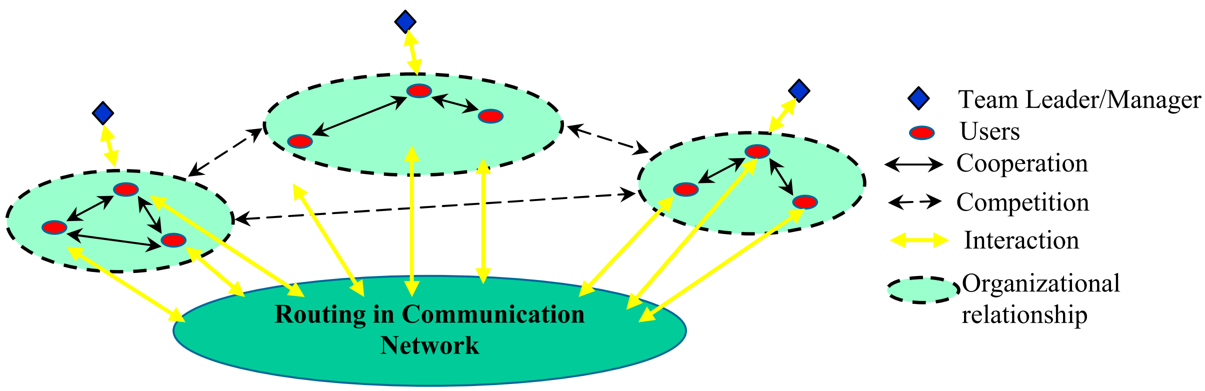


Figure 4.7 Diagram of hierarchical structure in network routing

4.3.1 Model and problem formulation

We present a generic parallel-link network model and formulate the routing problem within the framework of a multi-team system. We consider a set $N = \{1, \cdots, \mathrm{N}\}$ of teams and a set of Team Leaders (TLs) $TL = \{TL_1, \cdots, TL_N\}$. Each team consists of several users that share a set $L = \{1, \cdots, \mathrm{L}\}$ of communication links, interconnecting a common source and a common destination node. Let c_l be the capacity (or service rate) of link l, $c = (c_1, \cdots, c_{\mathrm{L}})$ the capacity configuration, and $C = \sum_{l=1}^{\mathrm{L}} c_l$ the total capacity of the system of parallel links. Suppose that $c_1 < c_2 < \cdots < c_L$. The i^{th} $(i = 1, \cdots, n_X)$ user has a throughput demand that is Poisson process with average rate λ_i^X>0. Let $\lambda = \sum_{X=1}^{\mathrm{N}} \sum_{i=1}^{n_X} \lambda_i^X$ be the total throughput demand of all users in the networks. Furthermore, for stability reasons it is supposed that the total throughput demand is less than the total capacity of the parallel links, i.e., $\lambda < C$. The i^{th} user in team X ships its flow by splitting its demand λ_i^X over the set of parallel links. Let $f_i^X(l)$ denote the expected fraction of flow that user i in team X sends on link l. The user flow fraction configuration

$$\mathrm{f}_i^X = \left(f_i^X(1), \cdots, f_i^X(\mathrm{L}) \right) \tag{4.8}$$

is called a routing strategy of user i in team X and the set

$$\mathrm{F}_i^X = \left\{ \mathrm{f}_i^X \in \mathbb{R}^{\mathrm{L}} : 0 \le \lambda_i^X f_i^X(l) \le c_l, \sum_{l=1}^{\mathrm{L}} f_i^X(l) = 1, 0 \le f_i^X(l) \le 1, l \in L \right\} \tag{4.9}$$

of strategies that satisfy the user's demand is called the strategy space of user i in team X. The routing control profile for the users from team X is denoted by

$$\mathrm{f}^X = \left(\mathrm{f}_1^X, \cdots, \mathrm{f}_{n_X}^X \right) \tag{4.10}$$

and takes values in the product strategy space

$$\mathrm{F}^X = \otimes_{i=1}^{n_X} \mathrm{F}_i^X . \tag{4.11}$$

The system routing control profile is given by

$$\mathrm{f} = \left(\mathrm{f}^1, \cdots, \mathrm{f}^{\mathrm{N}} \right) \tag{4.12}$$

and takes values in the overall product strategy space

$$\mathrm{F} = \otimes_{X=1}^{\mathrm{N}} \mathrm{F}^X . \tag{4.13}$$

Such a system is shown in Figure 4.8.

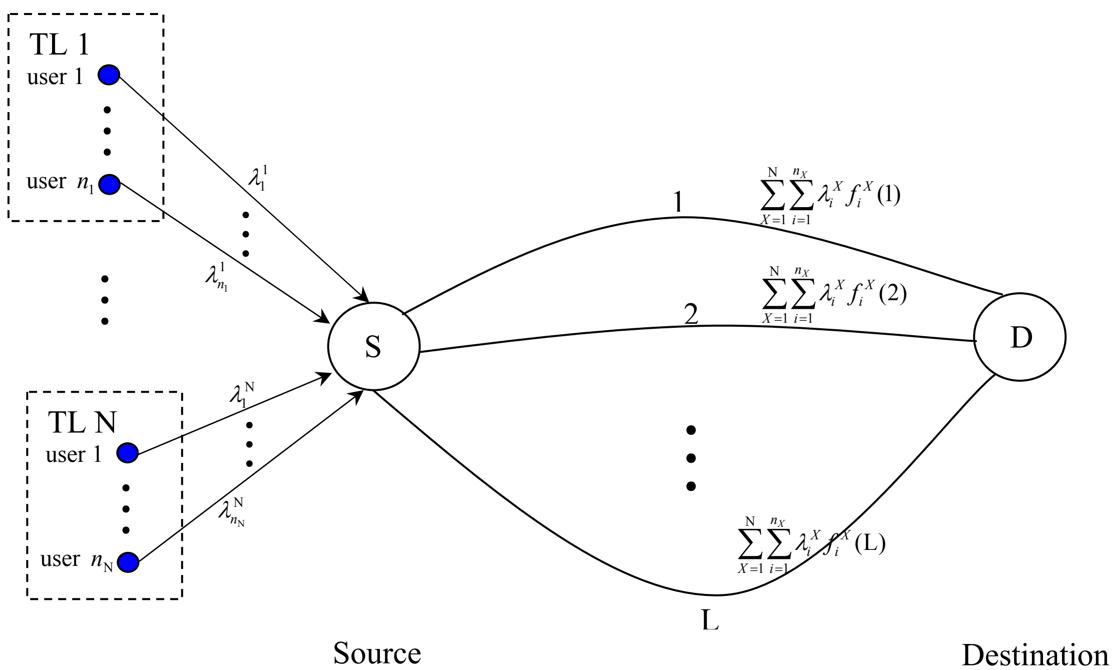


Figure 4.8 Two-node parallel-link communication network with multiple teams of users

The user i from team X has certain routing decision $\mathrm{f}_i^X = \left(f_i^X(1), \cdots, f_i^X(\mathrm{L}) \right)$ to make for the purpose of, for example, minimizing their average delay time. In this research, we consider the average delay as a cost function for each user. In particular, we let the service requirement of each user be exponentially distributed with mean 1, without loss of generality. We concentrate on the $M/M/1$ delay function [36] $d(l)$ on link l ($l \in \mathrm{L}$) :

$$d(l) = \begin{cases} \dfrac{1}{c_l - \sum_{X=1}^{\mathrm{N}} \sum_{i=1}^{n_X} \lambda_i^X f_i^X(l)} & \sum_{X=1}^{\mathrm{N}} \sum_{i=1}^{n_X} \lambda_i^X f_i^X(l) < c_l \\ \infty & \sum_{X=1}^{\mathrm{N}} \sum_{i=1}^{n_X} \lambda_i^X f_i^X(l) \geq c_l \end{cases} \tag{4.14}$$

Thus, the total delay for user i from M_X is:

$$d_i^X = \sum_{l=1}^{\mathrm{L}} \lambda_i^X f_i^X(l) d(l) . \tag{4.15}$$

The average delay, i.e., the cost function, for user i in team X under control strategy profile f_j^i to be minimized is given by

$$J_i^X(\mathrm{f}) = \frac{d_i^X}{\lambda_i^X} = \sum_{l=1}^{\mathrm{L}} f_i^X(l) d(l) \tag{4.16}$$

where $J_i^X : \mathrm{F} \to \mathbb{R}$ and, obviously, this cost function depends on the control strategies of other users also.

Team leaders may have various forms of objective functions, denoted by $P^X : \mathrm{F}^X \to \mathbb{R}$, at a higher level. In this section, we consider two types of objective functions for team leaders: efficiency objective function (Type 1) and flow cost function (Type 2). Team leader with the objective function of Type 1 wants to maximize the efficient utilization of highest capacity link. This objective function is given by

Type 1: $$P^X(\mathrm{f}^X) = \sum_{i=1}^{n_X} \lambda_i^X f_i^X(L) \tag{4.17a}$$

Team leader with the objective function of Type 2 is to minimize the total cost of flow for his users. Let $p^X(l)$ be the cost paid by users from team X for their flow on link l, and TL_X wishes to minimize the total cost of the flow. This objective function is given by

Type 2: $$P^X(\mathrm{f}^X) = \sum_{l=1}^{\mathrm{L}} \left(p^X(l) \sum_{i=1}^{n_X} \lambda_i^X f_i^X(l) \right) \tag{4.17b}$$

where $P^X : \mathrm{F}^X \to \mathbb{R}$.

The optimal routing problem is formulated as

$$\left.\begin{array}{l} \max\limits_{\mathrm{f}^X} P^X(\mathrm{f}^X) \text{ for Type 1} \quad \mathrm{f}^X \in \mathrm{F}^X, X \in N \\ \text{or } \min\limits_{\mathrm{f}^X} P^X(\mathrm{f}^X) \text{ for Type 2} \quad \mathrm{f}^X \in \mathrm{F}^X, X \in N \end{array}\right. \quad \text{for each TL in the system;} \tag{4.18a}$$

$$s.t. \quad \min_{\mathrm{f}_i^X} J_i^X(\mathrm{f}) \quad \mathrm{f} \in \mathrm{F}, \mathrm{f}_i^X \in \mathrm{F}_i^X, X \in N, i = 1, \cdots, n_X \quad \text{for each user from TL } X. \tag{4.18b}$$

4.3.2 Team optimization for single-team routing control problems

Before applying NNSL, let us consider the team optimization problem [19] in routing control, i.e., N=1. For simplicity, we consider two users with the throughput demand of λ_1 and λ_2, respectively, and two parallel links in the system with capacities of c_1 and c_2, respectively. Let x and y denote the fraction of flow demand of user 1 and user 2 will be assigned to link 1, respectively. According to constraints in (4.9), 1-x (or 1-y) is the fraction of flow demand of the user 1 (or user 2) will be assigned to link 2. The system is illustrated in Figure 4.9.

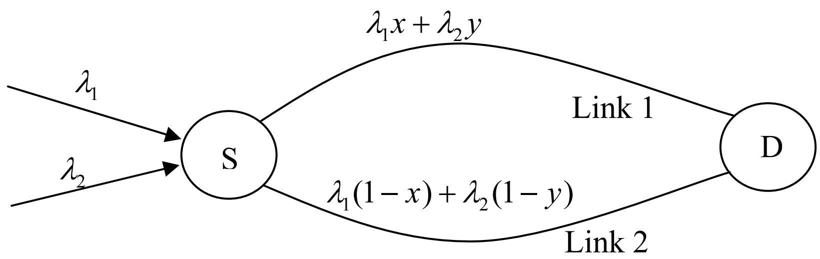


Figure 4.9 Single-team routing problem

As expressed in (4.16), the cost function J_i for user i is given by

$$J_1(x,y) = \frac{x}{c_1 - \lambda_1 x - \lambda_2 y} + \frac{1-x}{c_2 - \lambda_1(1-x) - \lambda_2(1-y)} \tag{4.19}$$

and

$$J_2(x,y) = \frac{y}{c_1 - \lambda_1 x - \lambda_2 y} + \frac{1-y}{c_2 - \lambda_1(1-x) - \lambda_2(1-y)} \tag{4.20}$$

In the team optimization problem, both users can cooperate with each other and there is a leader for the system, whose objective is to maximize the efficient usage of the link with high capacity (objective function of Type 1). The objective function for the team leader is given by

$$J_L(x,y) = \lambda_1(1-x) + \lambda_2(1-y) \tag{4.21}$$

The team optimization problem can be formulated as:

$$\max_{x,y} J_L(x,y) \tag{4.22}$$

$$s.t.\ \min_x J_1(x,y) \text{ and } \min_y J_2(x,y) \tag{4.23}$$

$$c_1 - \lambda_1 x - \lambda_2 y > 0 \text{ and } c_2 - \lambda_1(1-x) - \lambda_2(1-y) > 0 \tag{4.24}$$

$$0 \le x, y \le 1 \tag{4.25}$$

The cost functions $J_1(x,y)$ and $J_2(x,y)$ are convex with respect to x and y over the convex space given by (4.24) and (4.25). Thus, the optimal solution for (4.23) can be figured out by minimizing a weighted scalar-valued cost function $J(x,y;\alpha)$ as follows:

$$\min_{x,y} J(x,y;\alpha) = \alpha J_1(x,y) + (1-\alpha) J_2(x,y) \tag{4.26}$$

where α is a weight factor satisfying $0 \le \alpha \le 1$.

As we know, for each α, there exists an optimal solution $(x^*(\alpha), y^*(\alpha))$. Therefore, the infinity number of optimal solutions results from the infinity number of candidates of α. Since the cost function of team leader on the higher level is also determined by the optimal controls x by user 1 and y by user 2, $J_L(x,y)$ becomes a function of weight factor α. In other words, the objective function of team leader is used to decide the optimal choice of α.

For example, let p_1=400, p_2=100, $\lambda_1 = 1, \lambda_2 = 3, c_1 = 3$ and $c_2 = 6$. The convex set given by (4.24) and (4.25) is expressed as the blue-shaded area shown in Figure 4.10. The cost function of user 1, $J_1(x,y)$, is given in Figure 4.11. We observe that $J_1(x,y)$ is convex with respect convex set given by (4.24) and (4.25). However, the objective function for user 1, the average delay, is extremely large with respect to the decisions around the boundaries $c_1 - \lambda_1 x - \lambda_2 y$=0 and $c_2 - \lambda_1(1-x) - \lambda_2(1-y)$=0. Therefore, in practice, user 1 has to avoid the use of those decision choices. The objective functions $J_1(x,y)$ and $J_2(x,y)$ in reasonable areas are given in Figure 4.12. After figuring out all the possible cooperative controls for both users, i.e., $(x^*(\alpha), y^*(\alpha))$ for all α 's, we substitute these solutions to (4.21) to calculate the optimal value of J_L. The result is shown in Figure 4.13: $\alpha^* = 0.25$, $x^*(\alpha^*) = 0.03, y^*(\alpha^*) = 0.3$, $J_1^* = 0.3456$, $J_2^* = 0.3838$ and $J_L^* = 3.07$.

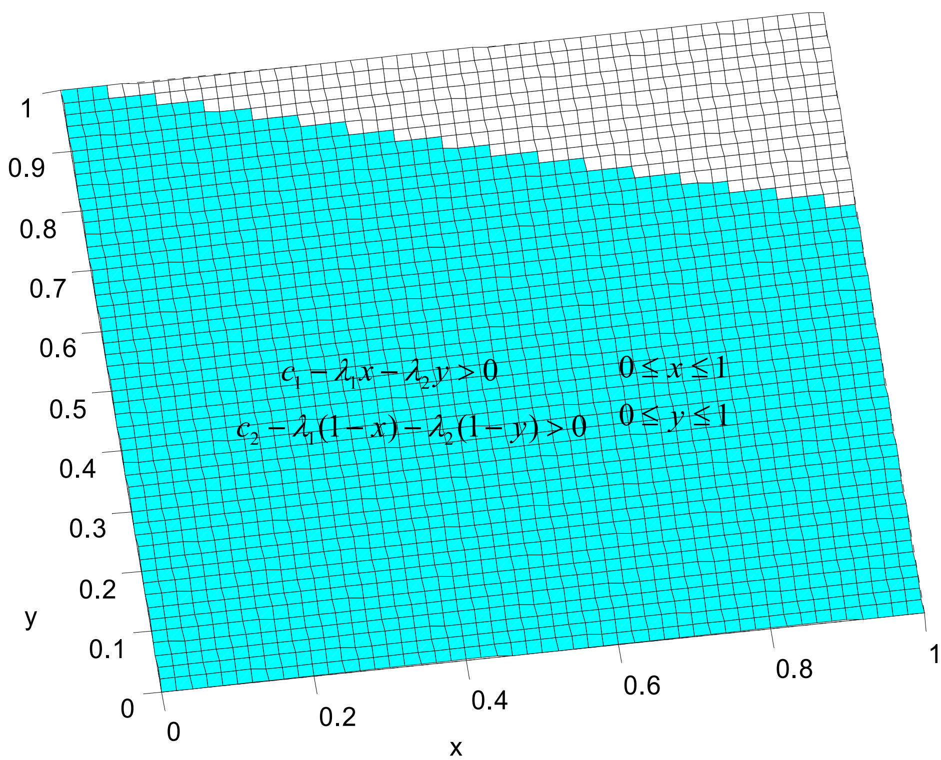


Figure 4.10 Convex set of the given example

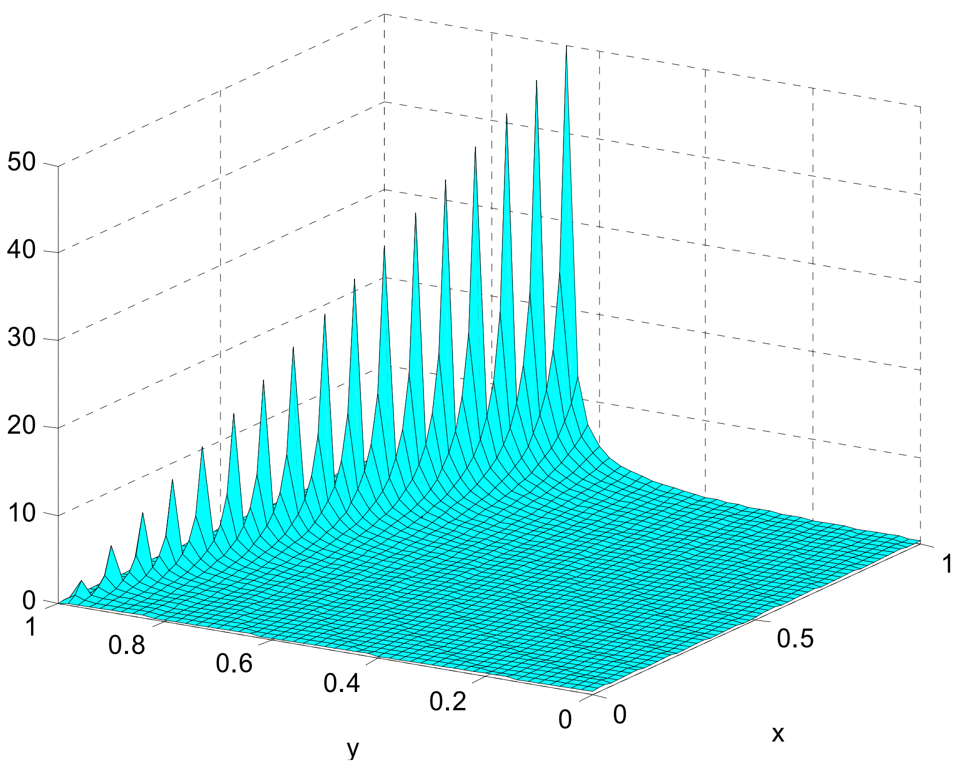


Figure 4.11 Convex cost function $J_1(x, y)$

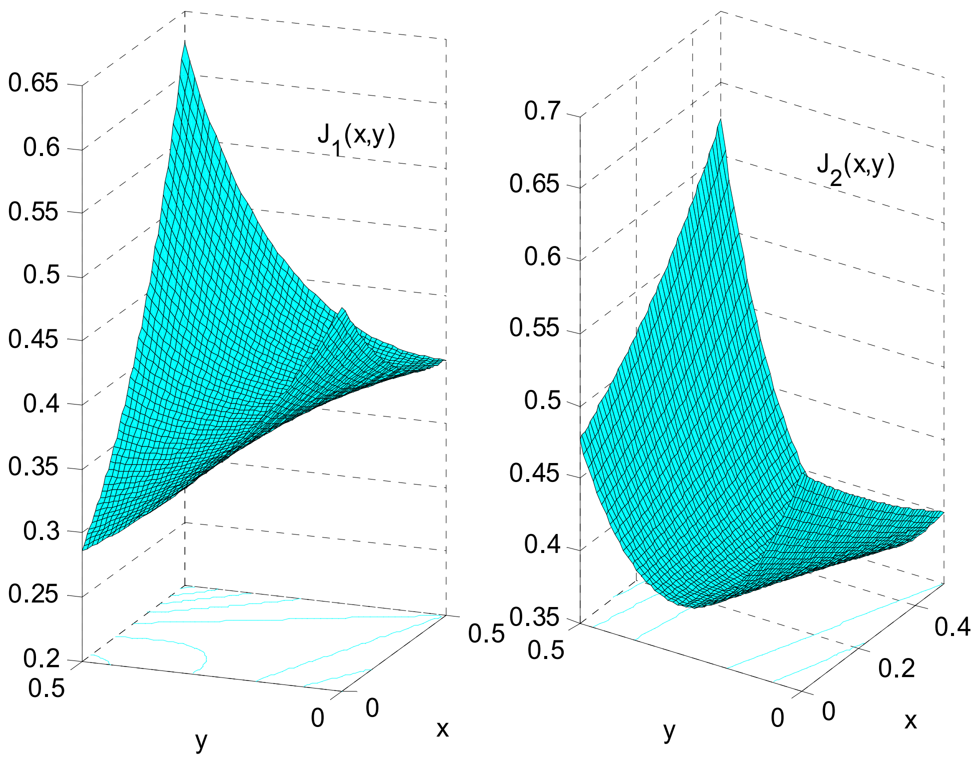


Figure 4.12 Cost functions $J_1(x, y)$ and $J_2(x, y)$ in reasonable areas

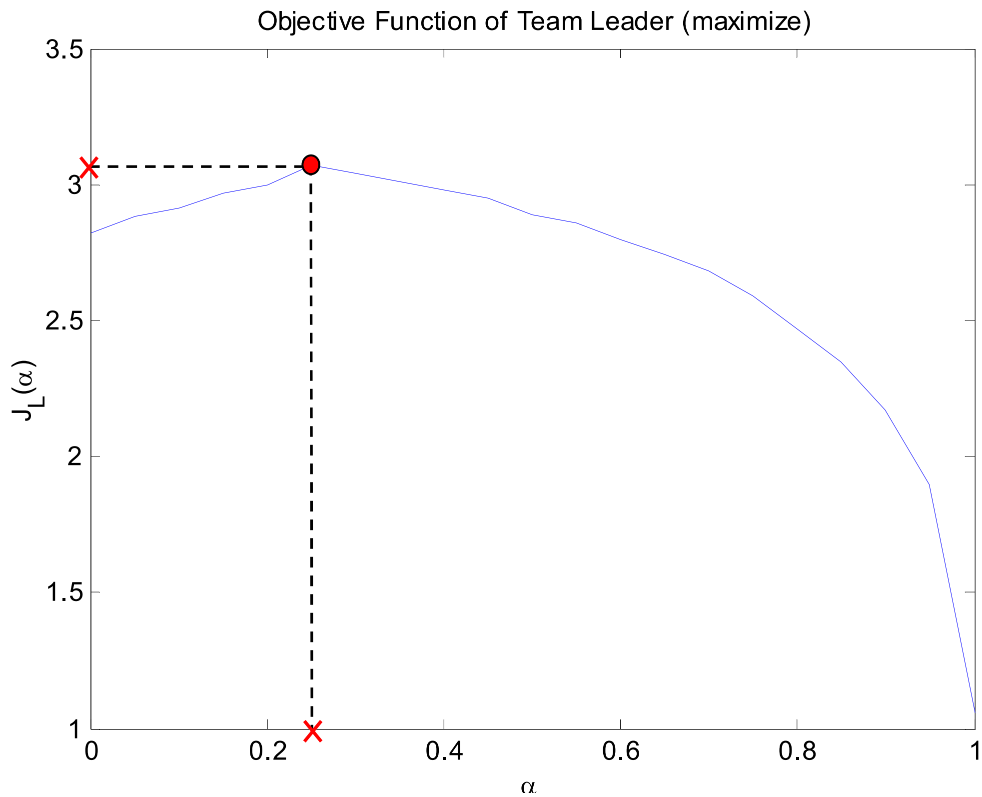


Figure 4.13 Objective function for team leader w.r.t. different values of weight factor

4.3.3 NNSL for multi-team routing control problems

In the team optimization problem as explained above, one assumption is that all the users in the system want to cooperate with each other. Naturally, it should be assumed that the users in the same team can cooperate with each other for the socially optimum and will compete for the limited common resources with the users in other teams.

For simplicity, assume there are two teams, called HET and TELE, respectively, and each team has two users also. We still consider a two-node parallel-link communication network as before. The total system capacity is $C = \sum_{X=1}^{2} c_X$. Let the throughput demand of user *i* from HET arrives at the system with rate λ_i^H (*i*=1,2). The total throughput demand for the users from HET is $\lambda^H = \sum_{i=1}^{2} \lambda_i^H$. The fractions of flow of user 1 and user 2 from HET assigned to link 1 are x ($\in [0,1]$) and y ($\in [0,1]$), respectively. Let the throughput demand of user *j* served by TELE arrives at the system with rate λ_j^T (*j*=1,2). The total throughput demand for TELE customers is $\lambda^T = \sum_{j=1}^{2} \lambda_j^T$. The fractions of flow of user 1 and user 2 from TELE assigned to link 1 are u ($\in [0,1]$) and v ($\in [0,1]$), respectively. Furthermore, we only consider the total capacity can accommodate the total user demand, that is, $\lambda^N + \lambda^T \le C$. The whole system is illustrated in Figure 4.14.

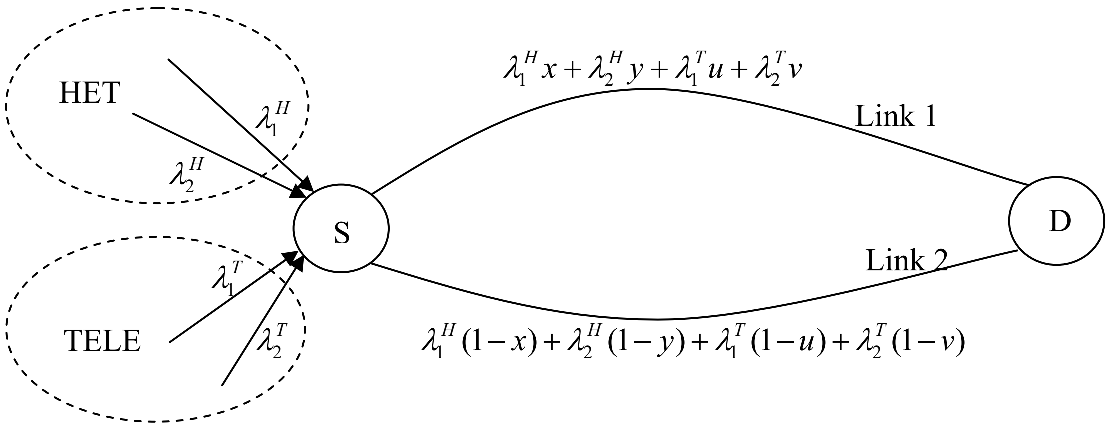


Figure 4.14 Two-team routing problem

As before, each user wants to minimize its average delay in the system. It can be formulated as the following optimal problem:

$$\min_x J_1^H(x,y,u,v) = \frac{x}{g(x,y,u,v)} + \frac{1-x}{h(x,y,u,v)} \tag{4.27}$$

$$\min_y J_2^H(x,y,u,v) = \frac{y}{g(x,y,u,v)} + \frac{1-y}{h(x,y,u,v)} \tag{4.28}$$

for the users from HET, and

$$\min_u J_1^T(x,y,u,v) = \frac{u}{g(x,y,u,v)} + \frac{1-u}{h(x,y,u,v)} \tag{4.29}$$

$$\min_v J_2^T(x,y,u,v) = \frac{v}{g(x,y,u,v)} + \frac{1-v}{h(x,y,u,v)} \tag{4.30}$$

for the users from TELE.

$$s.t. \qquad g(x,y,u,v) > 0 \text{ and } h(x,y,u,v) > 0 \tag{4.31}$$

$$0 \le x,y,u,v \le 1 \tag{4.32}$$

where

$$g(x, y, u, v) = C_1 - \lambda_1^H x - \lambda_2^H y - \lambda_1^T u - \lambda_1^T v,$$

and

$$h(x, y, u, v) = C_2 - \lambda_1^H (1-x) - \lambda_2^H (1-y) - \lambda_1^T (1-u) - \lambda_1^T (1-v).$$

Clearly, this optimal problem can be formulated as a multi-team system with N=2 and $n_1 = n_2 = 2$. The solution to this problem is a noninferior Nash strategy. The average delay objective functions J_i^H and J_j^T (i, j=1,2) are strictly convex over the convex space given by (4.31) and (4.32). According to Theorem 3.1, there exists a noninferior Nash strategy under a given weight vector $\xi = \left[\xi^N = (\alpha, 1-\alpha), \xi^T = (\beta, 1-\beta)\right]$ to the routing problem for the users served by two managers. The linear combinational weighted objective functions for the users are given by

$$J^H(\alpha) = \alpha J_1^H + (1-\alpha) J_2^H \tag{4.33}$$

$$J^T(\beta) = \beta J_1^T + (1-\beta) J_2^T \tag{4.34}$$

Note that the noninferior Nash strategies are the functions of α and β, i.e., $x^* = x^*(\alpha,\beta), y^* = y^*(\alpha,\beta), u^* = u^*(\alpha,\beta)$ and $v^* = v^*(\alpha,\beta)$. Since there are infinite combinations of α and β, we still need to decide the optimal weight vector ξ^*. We introduce different types of objective function for the two TLs:

$$\text{(Type 1)} \ \max_{(x^*, y^*)} J_L^H(x^*, y^*) = \lambda_1^H (1-x^*) + \lambda_2^H (1-y^*) \tag{4.35}$$

and

$$\text{(Type 2)} \quad \min_{(u^*, v^*)} J_L^T(u^*, v^*) = p_1^T(\lambda_1^T u^* + \lambda_2^T v^*) + p_2^T(\lambda_1^T (1-u^*) + \lambda_2^T (1-v^*)) \tag{4.36}$$

The leader from HET wants to maximize the throughput on the link with highest capacity ($C_2 > C_1$), and the leader from TELE wishes to minimize the total cost of usage of different links. Let p_1^T and p_2^T be the price per flow for link 1 and link 2, respectively. It is clear that $J_L^H(\cdot)$ and $J_L^T(\cdot)$ are the functions of α and β as well. The optimal choices of α and β can be determined by figuring out a Nash solution to a noncooperative game between two leaders with respective to the objective functions $J_L^H(\alpha,\beta)$ and $J_L^T(\alpha,\beta)$. Since it is not easy to obtain the analytical expression of NNSL to such a complicated hierarchical decision-making system, we use a numerical example to illustrate the properties and effectiveness of NNSL.

Let $c_1 = 3$, $c_2 = 6$, $\lambda_1^H = 1$, $\lambda_2^H = 3$, $\lambda_1^T = 0.5$, $\lambda_2^T = 1$, $p_1^T = 10$ and $p_2^T = 30$. The corresponding NNSL (optimal routing fractions) under the managers' objective functions are given in Table 4.2.

Table 4.2 Noninferior Nash strategies under the team leaders' objective functions

α^*	β^*	x^*	y^*	u^*	v^*	J_1^{H*}	J_2^{H*}	J_1^{T*}	J_2^{T*}	J_L^{H*}	J_L^{T*}
0.25	0.8,0.85, 0.9,0.95,1	0.7	0	0	1	0.6748	0.4545	0.4545	0.7692	3.3	25

For the purpose of comparison, we consider the situation where all the users choose the best strategies and they don't consider the corresponding manager's objective function. Clearly, these strategies among four users compose a Nash equilibrium strategy, which is given in Table 4.3. In other words, no user has a rational motive to unilaterally deviate from its equilibrium strategy.

Table 4.3 Nash strategies for four users

x^*	y^*	u^*	v^*	J_1^{H*}	J_2^{H*}	J_1^{T*}	J_2^{T*}	J_L^{H*}	J_L^{T*}
0.2	0.3	0.02	0.2	0.5424	0.5574	0.5194	0.5424	2.78	40

Comparing the results in Table 4.2 and Table 4.3, we observe that, using Nash strategy, some, but not all, users may gain better in reducing average delay time. However, considering the team leaders' cost functions and using NNSL, the total flow through link 2 from HET is 3.3, which is greater than 2.78 resulting from Nash strategy, and the cost paid by the manager from TELE is 25, which is less than that when implementing Nash solution. In other words, the objectives for both team leaders are improved by using NNSL.

4.4 CONCLUSIONS

In this chapter, we developed a new control strategy NNSL for the multi-team systems where each team has a leader with an objective usually different from those of the team members. This strategy is extended from Noninferior Nash Strategy. The team leader's objective function is used as a criterion for selecting a particular solution from the set of NNS for that team. Because each team's collective choice of control variable, in general, will depend on the choice of control variables by all other teams, each leader's objective function will also depend on all the other leaders' control variables. The team leaders' optimization problems, therefore, will need to be solved within the context of game theory as well. We use the examples of duopoly microeconomics and routing control in a two-node parallel-link network to illustrate the effectiveness of NNSL in improving the overall system performance.

5.0 GAME-THEORETIC MODELING AND CONTROL OF AN EXTENDED COMPLEX SYSTEM: MILITARY AIR OPERATION

As we know, a large-scale multi-team system involving many complex relationships such as cooperation among the members of a team and competition among the different groups. A military air operation is a good example of such an extended complex system. The schematic diagram of this system is shown in Figure 5.1. The military system usually includes two adversaries: an attacking force labeled as *Blue* and a defending force labeled as *Red*. Each force often has one top commander, and several fighting units (FUs), which are grouped and directed by unit commanders. In order to win a battle, two forces have to use the resources available to them to carry out a campaign against each other. In addition, the fighting units in either force must coordinate with each other in order to accomplish the assigned tasks efficiently. Obviously, the military operation system reflects important features of a large-scale multi-team system. Thus, optimizing the coordination for such system allows us to investigate how to control a complex multi-team system efficiently.

Model-based control of a military operation system provides us a convenient way to study the properties and performance of the extended complex system at a theoretical level. This kind of control is based on a dynamic attrition model of the military operation. A state space dynamic model of an extended complex military operation that involves two opposing forces is recently developed in [37]. Instead of only calculating the attrition for forces in an air combat, this model is expressed using the game theoretic approach and the state space approach, and hence is amenable for the application of results from modern optimal control [16] and dynamic game theories [12]. The model considers an attacking Blue force and a defending Red force as shown in Figure 5.1. The model is dynamic in nature with state variables whose evolution with time depends on the choice of control actions by both forces. It is extended in the sense that the effect of each opposing force, and the environment if any, are explicitly included in the model. The Blue force is composed of semi-autonomous aerial vehicles that consist of Blue Fighters (BFs) and Blue Bombers (BBs). The fighters are essentially SEAD (i.e., Suppressing Enemy Air Defenses) fighter planes whose purpose is to attack and suppress the Red air defenses, and the bombers are planes whose purpose is to destroy the Red Fixed Targets (FTs) such as bridges, refineries, air bases, etc. The Red force is composed of ground units that consist of Troops (RTs), such as tanks and mobile vehicles, and Air Defenses (RDs) such as SAM (i.e., Surface to Air Missile) batteries and radars. In addition, FTs are the units that the Blue force is planning to attack and the Red force is planning to defend. For each force, the command and control decisions are made at two levels: a top-level commander, followed by lower level unit commanders. The controls for a unit include relocation control, target selection, and fire control. The roles of a top commander involve mission planning, initial resource allocation and corridor assignment, etc. Each commander has an associated objective function, and these objective functions, even for the same force, may differ from each other for a variety of reasons. These objective function models will be used for investigating a range of possible game-theoretic control strategies.

In this chapter, we will introduce the state space dynamic model of this military air operation and the moving-horizon Nash strategies mainly used in controlling this system.

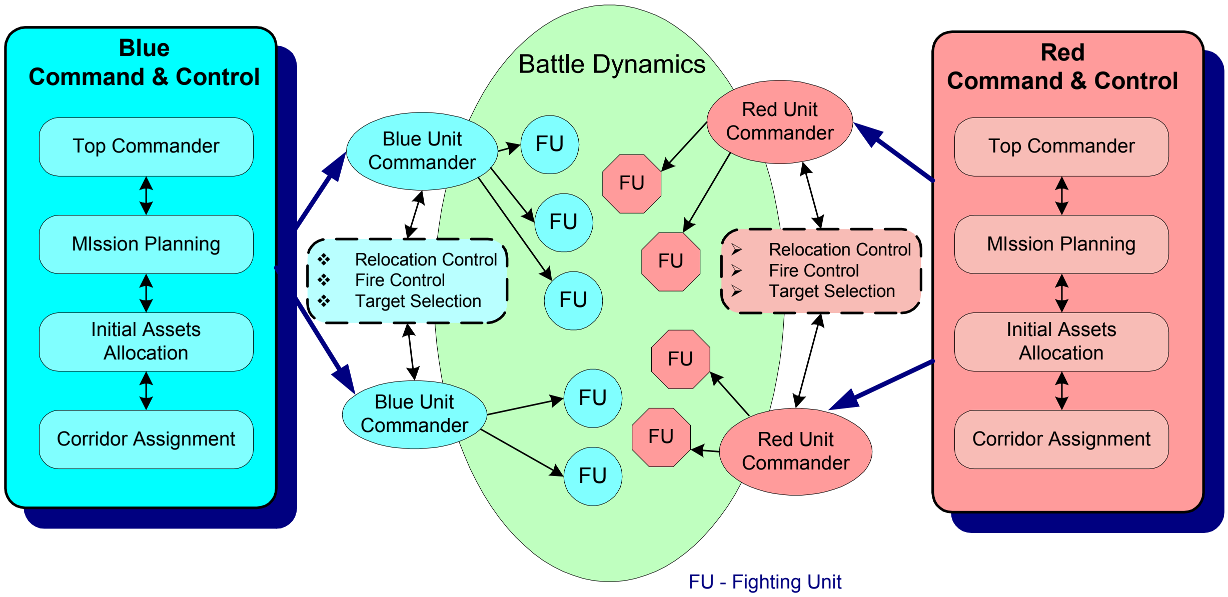


Figure 5.1 A military air operation system

5.1 DESCRIPTION OF MODEL

5.1.1 The unit's state variables

Let $N^{BB}, N^{BF}, N^{RT}, N^{RD}$, and N^{FT} denote the number of units of each type involved in the operation. Although the model can be derived in the continuous time-space domain, we will initially assume that time is sampled into stages $k = 0,1,2,\cdots,K$ where K is the total number of stages, and that the scenario is taking place on a two-dimensional terrain sampled in the x-y directions into a square grid. Continuous time and three-dimensional continuous space will be considered as an extension of this work at a later time.

Consider the i^{th} unit of type X, where $X = \{BB,\ BF,\ RT, \text{or } RD\}$. Let the vector $\eta_i^X(k) = \begin{bmatrix} x_i^X(k) \\ y_i^X(k) \end{bmatrix}$ denote its location at time k, where x is the horizontal coordinate and y is the vertical coordinate. In each force, the individual elements are grouped into units, and the elements in each unit are referred to as platforms. Thus a unit of BBs with ten platforms is a group of ten Blue Bombers acting as a unified entity. Each platform in a unit is carrying a certain number of weapons. Instead of considering individual weapons, we will characterize each unit by the average number of weapons per platform that it possesses. Let $p_i^X(k)$ denote the number of platforms and let $w_i^X(k)$ denote the average number of weapons per platform at time k in that unit. We use the word platform as a generic description of the type of force in each of the units in the model (e.g. fighters, bombers, troops, etc.). We assume that the platforms of a given unit carry only one type of weapons. Thus, for each moving unit in the theatre of operations, we will define a 4-dimensional state variable:

$$z_i^X(k) = \begin{bmatrix} x_i^X(k) \\ y_i^X(k) \\ p_i^X(k) \\ w_i^X(k) \end{bmatrix}, \quad X = \{BB, BF, RT, RD\}, \quad i = 1, 2, \cdots, N^X, \quad k = 0, 1, 2, 3, \cdots, K. \tag{5.1}$$

Combining all the state variables for each type of forces into one vector, we can write:

$$z^X(k) = \begin{bmatrix} z_1^X(k) \\ \vdots \\ z_{N^X}^X(k) \end{bmatrix}. \tag{5.2}$$

The overall state vectors corresponding to the Blue and Red forces are therefore defined as:

$$z^B(k) = \begin{bmatrix} z^{BB}(k) \\ z^{BF}(k) \end{bmatrix}, \qquad \text{and} \qquad z^R(k) = \begin{bmatrix} z^{RT}(k) \\ z^{RD}(k) \end{bmatrix} \tag{5.3}$$

Now, for the fixed targets, let their fixed positions be determined by the vectors $\eta_i^{FT} = \begin{bmatrix} x_i^{FT} \\ y_i^{FT} \end{bmatrix}, i = 1, 2, \cdots, N^{FT}$. Let $p_i^{FT}(k)$ denote the number of platforms in the i^{th} fixed target at time *k*. These platforms carry no weapons and are subject to attack by the Blue forces. We can define a state vector for the fixed targets as:

$$z^{FT}(k) = \begin{bmatrix} z_1^{FT}(k) \\ \vdots \\ z_{N^{FT}}^{FT}(k) \end{bmatrix}, \quad k = 0, 1, 2, 3, \cdots, K \tag{5.4}$$

where $z_i^{FT}(k) = \begin{bmatrix} x_i^{FT} \\ y_i^{FT} \\ p_i^{FT}(k) \\ 0 \end{bmatrix}, \qquad i = 1, 2, \cdots, N^{FT}$.

Combining the state vectors for the Blue and Red forces as well as the state vector for the fixed targets, we can define a $4 \times (N^{BB} + N^{BF} + N^{RT} + N^{RD}) + N^{FT}$ dimensional state vector for the entire operation as:

$$z(k) = \begin{bmatrix} z^B(k) \\ z^R(k) \\ z^{FT}(k) \end{bmatrix} \tag{5.5}$$

5.1.2 Two-level hierarchical controls and control constraints

(1) Unit Commander Controls

We will assume that each moving unit commander in Figure 5.1 has the following control (or command) variables at each time k:

- Relocate control: A unit can decide to relocate (move) to another adjacent point on the grid. The corresponding control command is:

$$r_i^X(k) = \begin{bmatrix} a_i^X(k) \\ b_i^X(k) \end{bmatrix}, \quad \text{where } a_i^X(k) \in \{-1, 0, +1\} \text{ and } b_i^X(k) \in \{-1, 0, +1\} \tag{5.6}$$

where a corresponds to a move in the x-direction and b corresponds to a move in the y-direction. There are eight neighboring locations that each unit can relocate to, as illustrated in Figure 5.2. The $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$ option corresponds to the unit deciding to remain in its current location.

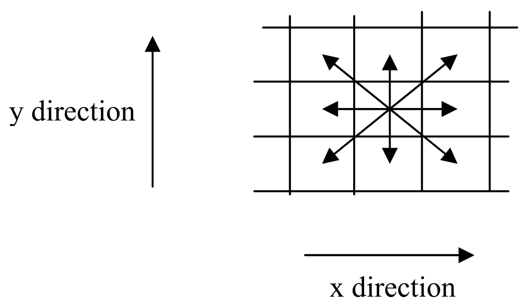


Figure 5.2 Relocate commands

- Fire Control: Each unit has an option to fire or not to fire. When a unit decides to fire, it must decide on the salvo size $c_i^X(k)$. There is a finite range from which the average salvo size at each time k can be chosen. That is

$$c_i^X(k) \in [0,\ C_i^X(k)] \tag{5.7}$$

where $C_i^X(k)$ is the largest salvo size that can be fired at time k. Note that if a unit decides not to fire, then $c_i^X(k) = 0$.

- Choice of Target: Each unit can fire only at one target of the opposing forces. If $d_i^X(k)$ denotes the choice of target for unit i at time $k,$ then

$$d_i^{BB}(k) = \{RT_j, RD_j, or\, FT_j \text{ for some j}\} \tag{5.8}$$

$$d_i^{BF}(k) = \{RT_j, RD_j, or\, FT_j \text{ for some j}\} \tag{5.9}$$

$$d_i^{RT}(k) = \{BB_j, or\, BF_j \text{ for some j}\} \tag{5.10}$$

$$d_i^{RD}(k) = \{BB_j, or\, BF_j \text{ for some j}\} \tag{5.11}$$

Combining all the command variables into one 4-dimensional control vector, we have the following control vector for each unit:

$$u_i^X(k) = \begin{bmatrix} a_i^X(k) \\ b_i^X(k) \\ c_i^X(k) \\ d_i^X(k) \end{bmatrix}. \tag{5.12}$$

We will now define a composite control vector for each type of forces:

$$u^{BB}(k) = \begin{bmatrix} u_1^{BB}(k) \\ u_2^{BB}(k) \\ \vdots \\ u_{N^{BB}}^{BB}(k) \end{bmatrix}, \qquad \text{and } u^{BF}(k) = \begin{bmatrix} u_1^{BF}(k) \\ u_2^{BF}(k) \\ \vdots \\ u_{N^{BF}}^{BF}(k) \end{bmatrix} \tag{5.13}$$

for the Blue units and

$$u^{RT}(k) = \begin{bmatrix} u_1^{RT}(k) \\ u_2^{RT}(k) \\ \vdots \\ u_{N^{RT}}^{RT}(k) \end{bmatrix}, \quad \text{and} \qquad u^{RD}(k) = \begin{bmatrix} u_1^{RD}(k) \\ u_2^{RD}(k) \\ \vdots \\ u_{N^{RD}}^{RD}(k) \end{bmatrix} \tag{5.14}$$

for the Red units.

The overall control vectors for the Blue and Red forces can be represented as:

$$u^B(k) = \begin{bmatrix} u^{BB}(k) \\ u^{BF}(k) \end{bmatrix}, \quad \text{and} \quad u^R(k) = \begin{bmatrix} u^{RT}(k) \\ u^{RD}(k) \end{bmatrix}. \tag{5.15}$$

The dimensionality of these vectors will be $4 \times (N^{BB} + N^{BF})$ and $4 \times (N^{RT} + N^{RD})$ respectively.

There are numerous constraints that the above control variables must satisfy. These are

- Relocate-Fire constraint: For simplicity, we will assume that a unit cannot relocate and fire at the same time. That is, a unit can fire its weapons only if its relocate command is $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$. This constraint can be expressed as:

$$\left\| r_i^X(k) \right\|_\infty + u[c_i^X(k) - 1] \leq 1. \tag{5.16}$$

In the above expression the infinite norm $\left\| r_i^X(k) \right\|_\infty$ is defined as follows:

$$\left\| r_i^X(k) \right\|_\infty = \begin{cases} 0, & \text{if } a_i^X(k) = 0 \text{ and } b_i^X(k) = 0 \\ 1, & \text{Otherwise} \end{cases} \tag{5.17}$$

That is, $\left\| r_i^X(k) \right\|_\infty$ is equal to 0 if the unit decides to stay in the same location and is equal to 1 if the unit decides to relocate. The notation $\mathrm{u}[n]$ is the standard discrete-time unit step sequence defined according to

$$\mathrm{u}[n] = \begin{cases} 0, & n < 0 \\ 1, & n \geq 0 \end{cases} \tag{5.18}$$

Thus $\mathrm{u}[c_i^X(k) - 1]$ is equal to 0 if the unit decides not to fire and is equal to 1 if the unit decides to fire. The unit step notation u[.] should not be confused with the control notation u(k) used throughout this thesis.

- Fire-Target constraint: We will assume that no two units of the same force can fire at the same target of the opposing force. The corresponding constraint is

$$\sum_{i=1}^{N^{BB}} \mathrm{u}[c_i^{BB}(k)-1] + \sum_{i=1}^{N^{BF}} \mathrm{u}[c_i^{BF}(k)-1] \le 1 \quad \text{for each Red target } j \tag{5.19}$$

and

$$\sum_{i=1}^{N^{RT}} \mathrm{u}[c_i^{RT}(k)-1] + \sum_{i=1}^{N^{RD}} \mathrm{u}[c_i^{RD}(k)-1] \le 1 \quad \text{for each Blue target } j \tag{5.20}$$

- Salvo size constraint: We will assume that ammunitions are not being replenished during the course of the operation. Thus, the largest salvo size that a unit can fire is constrained by

$$C_i^X(k+1) \le w_i^X(k) \tag{5.21}$$

- Speed constraints: In order to account for different entities moving at different speeds, we will choose the interval between consecutive steps (i.e. k to k+1) as the time that it takes the fastest unit to move one position on the grid. The relocate control of slower units can then be constrained to be activated only after a certain number of steps have elapsed, which corresponds to the time it will take that unit to move one position on the grid.

(2) Top Commander Control

The highest-level commands are the controls of the top commander of each force. As shown in Figure 5.1, its purpose is to define:

- The initial allocation of assets: That is, the numbers of platforms $p_i^X(0)$, and weapons $w_i^X(0)$ for each moving unit.

- The mission planning: That includes initial team composition, initial task assignment and resource re-allocation.
- The corridor assignment: Any constraints on the paths of each unit.

5.1.3 State difference equations

As mentioned earlier, the state vector for each moving unit is a 4-dimensional vector consisting of the position sub-vector η_i^X, the number of platforms p_i^X, and the number of weapons per platforms w_i^X in that unit. The state vector for each fixed target consists only of the position sub-vector η_i^{FT} and the number of platforms p_i^{FT}. We will now derive equations that describe the dynamics of the engagement between the forces. These equations relate the state variables at time *k+1* to the state and control variables at time *k*.

- The position sub-vectors for all moving units in $X = \{BB, BF, RT, RD\}$ change according to the equation of motions:

$$\eta_i^X(k+1) = \eta_i^X(k) + r_i^X(k) \tag{5.22}$$

- The number of platforms for the moving units change according to the following attrition equations:

$$\begin{aligned} p_i^{BB}(k+1) = \quad & p_i^{BB}(k)\Bigg[1 - \sum_{j=1}^{N^{RT}} Q_{ij}^{BBRT}(k)P_{ij}^{BBRT}(k)\delta(\eta_i^{BB}(k), \eta_j^{RT}(k))\delta(BB_i, d_j^{RT}(k)) \\ & - \sum_{j=1}^{N^{RD}} Q_{ij}^{BBRD}(k)P_{ij}^{BBRD}(k)\delta(\eta_i^{BB}(k), \eta_j^{RD}(k))\delta(BB_i, d_j^{RD}(k))\Bigg] \end{aligned} \tag{5.23}$$

$$\begin{aligned} p_i^{BF}(k+1) = \quad & p_i^{BF}(k)\Bigg[1 - \sum_{j=1}^{N^{RT}} Q_{ij}^{BFRT}(k)P_{ij}^{BFRT}(k)\delta(\eta_i^{BF}(k), \eta_j^{RT}(k))\delta(BF_i, d_j^{RT}(k)) \\ & - \sum_{j=1}^{N^{RD}} Q_{ij}^{BFRD}(k)P_{ij}^{BFRD}(k)\delta(\eta_i^{BF}(k), \eta_j^{RD}(k))\delta(BF_i, d_j^{RD}(k))\Bigg] \end{aligned} \tag{5.24}$$

$$p_i^{RT}(k+1) = p_i^{RT}(k)\left[1-\sum_{j=1}^{N^{BB}} Q_{ij}^{RTBB}(k)P_{ij}^{RTBB}(k)\delta(\eta_i^{RT}(k),\eta_j^{BB}(k))\delta(RT_i,d_j^{BB}(k)) \right. \\ \left. -\sum_{j=1}^{N^{BF}} Q_{ij}^{RTBF}(k)P_{ij}^{RTBF}(k)\delta(\eta_i^{RT}(k),\eta_j^{BF}(k))\delta(RT_i,d_j^{BF}(k))\right] \tag{5.25}$$

$$p_i^{RD}(k+1) = p_i^{RD}(k)\left[1-\sum_{j=1}^{N^{BB}} Q_{ij}^{RDBB}(k)P_{ij}^{RDBB}(k)\delta(\eta_i^{RD}(k),\eta_j^{BB}(k))\delta(RD_i,d_j^{BB}(k)) \right. \\ \left. -\sum_{j=1}^{N^{BF}} Q_{ij}^{RDBF}(k)P_{ij}^{RDBF}(k)\delta(\eta_i^{RD}(k),\eta_j^{BF}(k))\delta(RD_i,d_j^{BF}(k))\right] \tag{5.26}$$

and

- The number of platforms for the Fixed Targets change according to the attrition equation:

$$p_i^{FT}(k+1) = p_i^{FT}(k)\left[1-\sum_{j=1}^{N^{BB}} Q_{ij}^{FTBB}(k)P_{ij}^{FTBB}(k)\delta(\eta_i^{FT}(k),\eta_j^{BB}(k))\delta(FT_i,d_j^{BB}(k)) \right. \\ \left. -\sum_{j=1}^{N^{BF}} Q_{ij}^{FTBF}(k)P_{ij}^{FTBF}(k)\delta(\eta_i^{FT}(k),\eta_j^{BF}(k))\delta(FT_i,d_j^{BF}(k))\right] \tag{5.27}$$

In expressions (5.23) to (5.27), the Kronecker delta, is defined as

$$\delta(V,W) = \begin{cases} 0 & \text{if } V \neq W \\ 1 & \text{if } V = W \end{cases}, \tag{5.28}$$

and the terms $Q_{ij}^{XY}(k)$ and $P_{ij}^{XY}(k)$ represent the *engagement* and *attrition* factors between the attacking unit (j^{th} unit of Y) and the unit being attacked (i^{th} unit of X). These two factors are determined according to the following expressions:

$$Q_{ij}^{XY}(k) = \beta_{pij}^{XY}(1-e^{-\mu_{pij}^{XY}\frac{p_j^Y(k)}{p_i^X(k)}}) \tag{5.29}$$

and

$$P_{ij}^{XY}(k) = 1-(1-\beta_w PK_{ij}^{XY})^{s_j^Y(k)} \tag{5.30}$$

In expression (5.29), $p_i^X(k)$ and $p_j^Y(k)$ are the number of platforms in the i^{th} unit of X and j^{th} unit of Y respectively and β_{pij}^{XY} represents the probability that the j^{th} unit of Y acquires the i^{th} unit of X as a target. This probability is modified by an exponential factor that starts at 0 if the size of the attacking unit is much smaller than that of the unit being attacked and increases exponentially towards 1 as the size of the unit being attacked decreases in relation to the size of the attacking unit. This can be seen in Figure 5.3 with $\beta^{XY} = 1$ and $\mu^{XY} = 1$. The term μ_{pij}^{XY} is a normalizing factor that uniformly scales the units of these platforms if they are of different types.

In expression (5.30), β_w is a weather dependent modification factor ($0 \le \beta_w \le 1$), PK_{ij}^{XY} represents the probability of kill under ideal weather conditions for a single weapon (i.e. an effective salvo size of 1) for the type of weapon used by unit *j* against the type of platform in unit *i*, and $s_j^Y(k)$ represents the average effective salvo size of the weapons fired by the j^{th} unit of *Y* that reach the i^{th} unit of X at time *k*. Mathematically, $s_j^Y(k)$ is computed according to:

$$s_j^Y(k) = \frac{c_j^Y(k)p_j^Y(k)}{p_i^X(k)} E\left(\frac{p_j^Y(k)}{p_i^X(k)}\right) \tag{5.31}$$

where $E(\cdot)$ is a factor that models the inefficiencies of scale that may exist when two forces of unequal sizes are engaged in combat and modifies the average salvo size that reaches the target accordingly. This factor was first introduced by Helmbold [38],[39] as a modification of Lanchester's equations, and was labeled as the effective *firing modification factor*. In essence, this factor takes into account the fact that the larger the size of the attacking force with respect to the force being attacked the less effective their weapons will be. In other words, $E(\cdot)$ should be a decreasing function of its argument. In our model, we will use the following expression for $E(\cdot)$ as was suggested by Helmbold [38],[39]:

$$E\left(\frac{p_j^Y(k)}{p_i^X(k)}\right) = \left(\mu_{pij}^{XY} \frac{p_j^Y(k)}{p_i^X(k)}\right)^{\omega-1} \tag{5.32}$$

where the factor $0 \le \omega \le 1$ is referred to as the Weiss parameter. If the attacking unit is much larger than the unit being attacked, the firing modification factor will decrease the effectiveness of the average salvo size that reaches the target. This is so, because of the physical constraints on space and timing that limit the capability of the larger attacking force. On the other hand, if the force being attacked is much larger than the attacking force, then the firing modification factor will increase the effectiveness of the average salvo size that reaches the target. This is so because the attacking force will have more targets to choose from. One example of the size effect factors is given in Figure 5.4 where $\omega = \frac{1}{2}$.

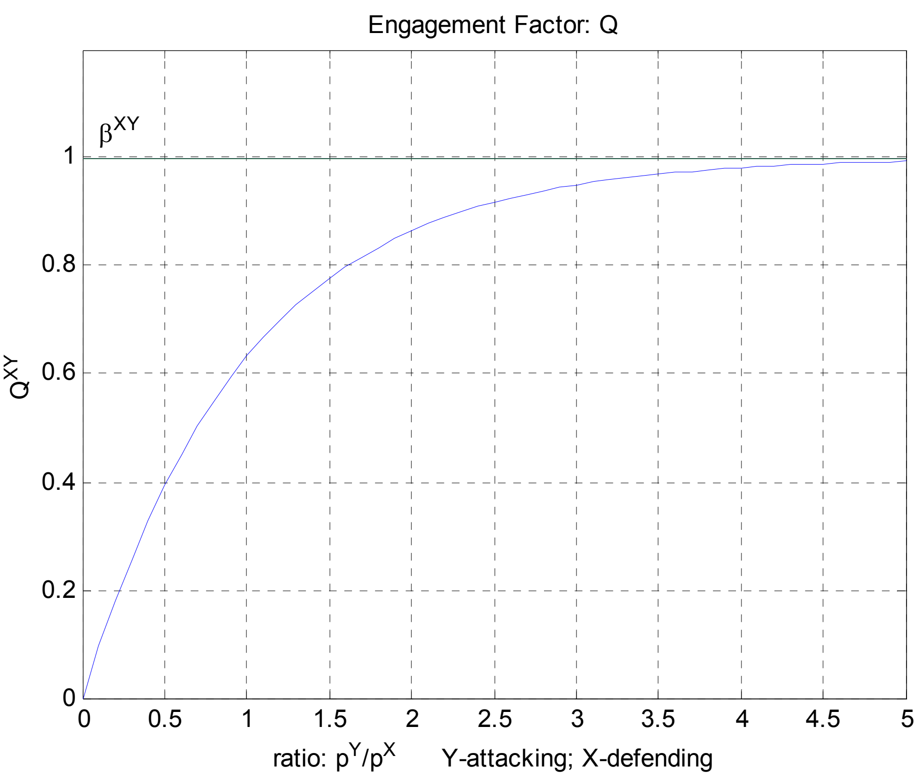


Figure 5.3 Engagement factor

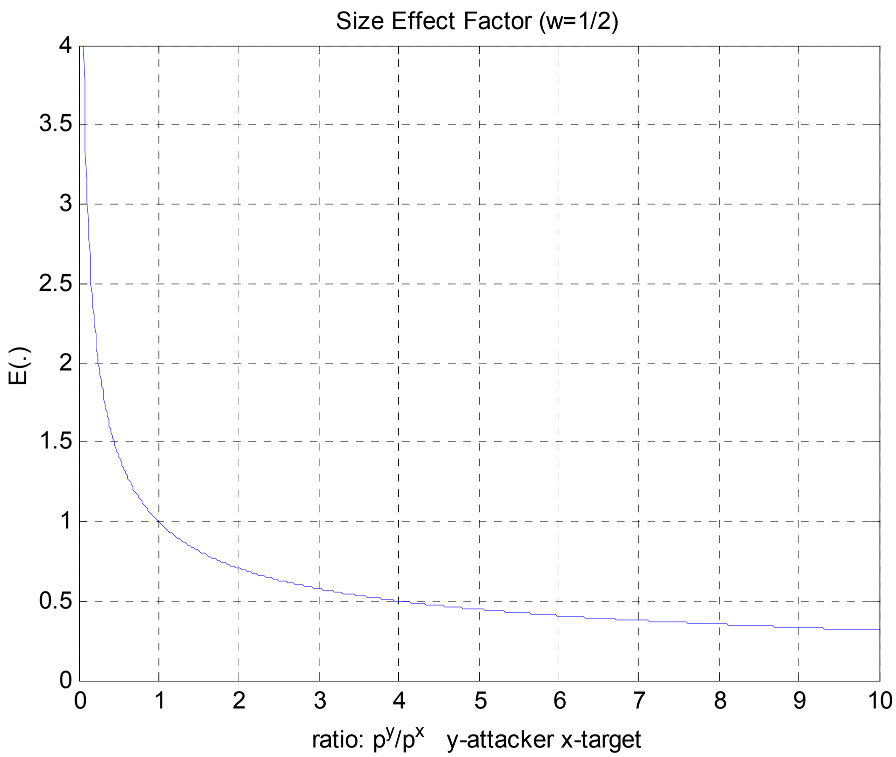


Figure 5.4 Size effect factor

Finally, we should mention that the form of equations (5.23)-(5.27) clearly assumes that acquisition of the target on the part of the attacking unit always occurs before weapons release. The number of weapons per platform for all moving units in $X = \{BB,\ BF,\ RT,\ RD\}$ changes according to the following expressions:

$$w_i^X(k+1) = w_i^X(k) - c_i^X(k) \tag{5.33}$$

Now, combining the position state equations (5.22), the number of platforms state equations (5.23)-(5.32), and the number of weapons state equations (5.33) for all forces into one vector, we get the final expression for the state equations

$$z(k+1) = f(z(k), u^B(k), u^R(k), k) \tag{5.34}$$

where z is a $4\times(N^{BB} + N^{BF} + N^{RT} + N^{RD}) + N^{FT}$ dimensional state vector, u^B is an $4\times(N^{BB} + N^{BF})$ dimensional control vector of the Blue forces and u^R is an $4\times(N^{RT} + N^{RD})$ dimensional control vector of the Red forces. The function f is a $4\times(N^{BB} + N^{BF} + N^{RT} + N^{RD}) + N^{FT}$ vector of functions determined from equations (5.22)-(5.33) as described above.

5.1.4 Two-level objective functions

As mentioned earlier, our model considers two levels of command for each force; the top-level commander control and the lower level unit controls.

(1) Top Commanders' Objective Functions

We will assume that the objective of each top commander is to allocate the least amount of initial resources to its forces while at the same time insuring that:

1. The total number of platforms and the total number of weapons of its own forces remaining at the end of the battle are maximized, and

2. The total number of platforms and the total number of weapons of the adversary's forces remaining at the end of the battle are minimized.

Thus, the Blue force top commander must decide on the allocation of the initial assets $p_i^{BF}(0)$, $w_i^{BF}(0)$, $p_i^{BB}(0)$, and $w_i^{BB}(0)$ to maximize the objective function:

$$
\begin{aligned}
J^B \quad = \quad & \alpha_1^B \sum_{i=1}^{N^{BB}} p_i^{BB}(K) w_i^{BB}(K) + \alpha_2^B \sum_{i=1}^{N^{BB}} p_i^{BB}(K) \\
& + \alpha_3^B \sum_{i=1}^{N^{BF}} p_i^{BF}(K) w_i^{BF}(K) + \alpha_4^B \sum_{i=1}^{N^{BF}} p_i^{BF}(K) \\
& - \alpha_5^B \sum_{i=1}^{N^{RT}} p_i^{RT}(K) w_i^{RT}(K) - \alpha_6^B \sum_{i=1}^{N^{RT}} p_i^{RT}(K) \\
& - \alpha_7^B \sum_{i=1}^{N^{RD}} p_i^{RD}(K) w_i^{RD}(K) - \alpha_8^B \sum_{i=1}^{N^{RD}} p_i^{RD}(K) - \alpha_9^B \sum_{i=1}^{N^{FT}} p^{FT}(K) \\
& - \alpha_{10}^B \sum_{i=1}^{N^{BB}} p_i^{BB}(0) w_i^{BB}(0) - \alpha_{11}^B \sum_{i=1}^{N^{BB}} p_i^{BB}(0) \\
& - \alpha_{12}^B \sum_{i=1}^{N^{BF}} p_i^{BF}(0) w_i^{BF}(0) - \alpha_{13}^B \sum_{i=1}^{N^{BF}} p_i^{BF}(0)
\end{aligned}
\tag{5.35}
$$

Additionally, the top Blue commander may require the Blue Forces to remain within a prescribed corridor

$$
\psi^B(\eta_i^{BB}(k), \eta_j^{BF}(k)) = 0 \tag{5.36}
$$

in the state space. Similarly, the Red force top commander must decide on the allocation of the initial assets $p_i^{RT}(0)$, $w_i^{RT}(0)$, $p_i^{RD}(0)$, and $w_i^{RD}(0)$ to maximize the objective function:

$$
\begin{aligned}
J^R \quad = \quad & \alpha_1^R \sum_{i=1}^{N^{RT}} p_i^{RT}(K) w_i^{RT}(K) + \alpha_2^R \sum_{i=1}^{N^{RT}} p_i^{RT}(K) \\
& + \alpha_3^R \sum_{i=1}^{N^{RD}} p_i^{RD}(K) w_i^{RD}(K) + \alpha_4^R \sum_{i=1}^{N^{RD}} p_i^{RD}(K) \\
& - \alpha_5^R \sum_{i=1}^{N^{BB}} p_i^{BB}(K) w_i^{BB}(K) - \alpha_6^R \sum_{i=1}^{N^{BB}} p_i^{BB}(K) \\
& - \alpha_7^R \sum_{i=1}^{N^{BF}} p_i^{BF}(K) w_i^{BF}(K) - \alpha_8^R \sum_{i=1}^{N^{BF}} p_i^{BF}(K) + \alpha_9^R \sum_{i=1}^{N^{FT}} p_i^{FT}(K) \\
& - \alpha_{10}^R \sum_{i=1}^{N^{RT}} p_i^{RT}(0) w_i^{RT}(0) - \alpha_{11}^R \sum_{i=1}^{N^{RT}} p_i^{RT}(0) \\
& - \alpha_{12}^R \sum_{i=1}^{N^{RD}} p_i^{RD}(0) w_i^{RD}(0) - \alpha_{13}^R \sum_{i=1}^{N^{RD}} p_i^{RD}(0)
\end{aligned}
\tag{5.37}
$$

and may require the Red Forces to remain within a prescribed corridor

$$\psi^R(\eta_i^{RT}(k), \eta_j^{RD}(k)) = 0 \tag{5.38}$$

in the state space. In each of the previous expressions, we will assume that the alpha's are all non-negative coefficients that account for normalization of the various terms in the objective function as well as the distribution of weights to assign relative importance to the terms in the objective function.

(2) Unit's Objective Functions

Once the top commander's decisions have been made, the various units must at each time k thereafter decide on their control vectors $u^{BB}(k)$, $u^{BF}(k)$, $u^{RT}(k)$, and $u^{RD}(k)$. We will assume that each unit's objective is to preserve as much as possible its own forces (platforms and total weapons) and destroy as much as possible the forces of its adversary. Thus, for the objective of the Blue Fighters is to maximize

$$J^{BF} = \sum_{k=1}^{K} J^{BF}(k) \tag{5.39}$$

where

$$
\begin{aligned}
J^{BF}(k) \;&=\; \alpha_1^{BF}\sum_{i=1}^{N^{BF}} p_i^{BF}(k)w_i^{BF}(k)+\alpha_2^{BF}\sum_{i=1}^{N^{BF}} p_i^{BF}(k)+\alpha_3^{BF}\sum_{i=1}^{N^{BB}} p_i^{BB}(k)\\
&\quad -\alpha_4^{BF}\sum_{i=1}^{N^{RT}} p_i^{RT}(k)w_i^{RT}(k)-\alpha_5^{BF}\sum_{i=1}^{N^{RT}} p_i^{RT}(k)\\
&\quad -\alpha_6^{BF}\sum_{i=1}^{N^{RD}} p_i^{RD}(k)w_i^{RD}(k)-\alpha_7^{BF}\sum_{i=1}^{N^{RD}} p_i^{RD}(k)
\end{aligned} \tag{5.40}
$$

and the objective of the Blue Bombers is to maximize

$$J^{BB} = \sum_{k=1}^{K} J^{BB}(k) \tag{5.41}$$

where

$$
\begin{aligned}
J^{BB}(k) \;&=\; \alpha_1^{BB}\sum_{i=1}^{N^{BB}} p_i^{BB}(k)w_i^{BB}(k)+\alpha_2^{BB}\sum_{i=1}^{N^{BB}} p_i^{BB}(k)-\alpha_3^{BB}\sum_{i=1}^{N^{FT}} p_i^{FT}(k)\\
&\quad -\alpha_4^{BB}\sum_{i=1}^{N^{RT}} p_i^{RT}(k)w_i^{RT}(k)-\alpha_5^{BB}\sum_{i=1}^{N^{RT}} p_i^{RT}(k)\\
&\quad -\alpha_6^{BB}\sum_{i=1}^{N^{RD}} p_i^{RD}(k)w_i^{RD}(k)-\alpha_7^{BB}\sum_{i=1}^{N^{RD}} p_i^{RD}(k)
\end{aligned} \tag{5.42}
$$

In J^{BB} we have assumed that an additional objective of the Blue Bombers is to minimize the dispositions (i.e. destroy) of the fixed targets. In a similar fashion, we will define objective functions for the Red Troops and Red Defenses as:

$$J^{RT} = \sum_{k=1}^{K} J^{RT}(k) \tag{5.43}$$

where

$$
\begin{aligned}
J^{RT}(k) \;&=\; \alpha_1^{RT}\sum_{i=1}^{N^{RT}} p_i^{RT}(k)w_i^{RT}(k)+\alpha_2^{RT}\sum_{i=1}^{N^{RT}} p_i^{RT}(k)+\alpha_3^{RT}\sum_{i=1}^{N^{FT}} p_i^{FT}(k)\\
&\quad -\alpha_4^{RT}\sum_{i=1}^{N^{BF}} p_i^{BF}(k)w_i^{BF}(k)-\alpha_5^{RT}\sum_{i=1}^{N^{BF}} p_i^{BF}(k)\\
&\quad -\alpha_6^{RT}\sum_{i=1}^{N^{BB}} p_i^{BB}(k)w_i^{BB}(k)-\alpha_7^{RT}\sum_{i=1}^{N^{BB}} p_i^{BB}(k)
\end{aligned} \tag{5.44}
$$

and

$$J^{RD} = \sum_{k=1}^{K} J^{RD}(k) \tag{5.45}$$

where

$$\begin{aligned} J^{RD}(k) \quad = \quad & \alpha_1^{RD} \sum_{i=1}^{N^{RD}} p_i^{RD}(k) w_i^{RD}(k) + \alpha_2^{RD} \sum_{i=1}^{N^{RD}} p_i^{RD}(k) + \alpha_3^{RD} \sum_{i=1}^{N^{FT}} p_i^{FT}(k) \\ & - \alpha_4^{RD} \sum_{i=1}^{N^{BF}} p_i^{BF}(k) w_i^{BF}(k) - \alpha_5^{RD} \sum_{i=1}^{N^{BF}} p_i^{BF}(k) \\ & - \alpha_6^{RD} \sum_{i=1}^{N^{BB}} p_i^{BB}(k) w_i^{BB}(k) - \alpha_7^{RD} \sum_{i=1}^{N^{BB}} p_i^{BB}(k) \end{aligned} \tag{5.46}$$

In each of the previous expressions, we will assume that the alpha's are all non-negative coefficients that account for normalization of the various terms in the objective function as well as the distribution of weights to assign relative importance to the terms in the objective function. A sensitivity analysis of the effects of changing the distribution of weights [2] can provide a useful guide to a commander in planning for a battle, depending on the importance of various targets, availability of assets, and other mission constraints.

5.2 MOVING-HORIZON NASH CONTROLS

As described in the last section, the nonlinear model encompasses different types of controls that may generate many different control choices. Depending on the richness of the control space, the number of possible states in which the system can be found can grow faster or slower, but always exponentially as a function of time. Even in problems of reasonable size, deriving the optimal control solution for both forces may not be feasible for more than a couple of time steps. In other words, the optimization of the overall system, especially involving two opponent groups of controllers, is almost impossible. Moving-horizon controls with finite steps are therefore considered since, as we know, such type of control is particularly useful in the process where the dynamic system to be controlled is complicate and often subject to control and state constraints.

5.2.1 K-step moving-horizon optimal controls

Consider a discrete-time system controlled by two independent decision-makers whose state vector evolves according to the equation:

$$x_{k+1} = f_k(x_k, u_k^1, u_k^2), \qquad k=0,\cdots,N-1 \tag{5.47}$$

where x_k is the state vector, u_k^1 and u_k^2 are independent control sequences of the two decision-makers and x_0 is the initial condition at $k=0$. Suppose that each decision-maker wishes to optimize a performance index over the interval $[0,\ N]$ of the form

$$J^i = \phi_N^i\left(N, x_N\right) + \sum_{k=0}^{N-1} L^i\left(x_k, u_k^1, u_k^2\right), \qquad i = 1,2\,. \tag{5.48}$$

Obtaining a game theoretic optimal solution for such a system may be extremely complex [12], and its complexity may rise exponentially with the length of the time horizon *N*. In order to overcome these difficulties, we will consider an optimal solution over a short moving horizon of steps, which in general will require much less computational effort. We can formulate this problem in the following form.

Given the description of the dynamic system (5.47) and the performance indices (5.48) for both decision-makers, we can obtain a solution $\left\{u_k^{1^*}, u_k^{2^*}\right\}$ at time k by considering performance indices over the reduced interval of K look-ahead steps:

$$J_{k,k+K}^i = \phi_{k+K}^i\left(k+K, x_{k+K}\right) + \sum_{j=k}^{k+K-1} L^i\left(x_j, u_j^1, u_j^2\right), \qquad i=1,2 \quad k < N-K \tag{5.49}$$

Once the end of the horizon is reached, we retain the original form of the performance indices for $k \ge N-K$ and calculate the optimal sequence $\left\{u_k^{1^*}, u_k^{2^*}\right\}_{N-K}^{N}$ using

$$J_{N-K,N}^{i} = \phi_N^i\left(N, x_N\right) + \sum_{k=N-K}^{N-1} L^i\left(x_k, u_k^1, u_k^2\right), \quad i = 1,2\,. \tag{5.50}$$

We called such an optimal control sequence as a *K*-step moving-horizon control.

5.2.2 One-step and two-step look ahead Nash controls

One-step and two-step moving-horizon Nash controls using dynamic programming methods have been developed by J.B Cruz *et al.* in [40]. For the purpose of simplicity, in the initial stage, they ignored the hierarchical control structures inside the military model and assume that each force is looked as an entity or a group. Thus, for each of the two forces, an aggregate objective function is defined at every stage k that each force wishes to maximize. These functions are in the form:

$$J^B(k) = \sum_{i=1}^{N^{BB}} \alpha_{BBi}\hat{p}_i^{BB}(k) + \sum_{i=1}^{N^{BF}} \alpha_{BFi}\hat{p}_i^{BF}(k) - \sum_{i=1}^{N^{RT}} \alpha_{RTi}\hat{p}_i^{RT}(k) - \sum_{i=1}^{N^{RD}} \alpha_{RDi}\hat{p}_i^{RD}(k) - \sum_{i=1}^{N^{FT}} \alpha_{FTi}\hat{p}_i^{FT}(k) \tag{5.51a}$$

$$J^R(k) = -\sum_{i=1}^{N^{BB}} \beta_{BBi}\hat{p}_i^{BB}(k) - \sum_{i=1}^{N^{BF}} \beta_{BFi}\hat{p}_i^{BF}(k) + \sum_{i=1}^{N^{RT}} \beta_{RTi}\hat{p}_i^{RT}(k) + \sum_{i=1}^{N^{RD}} \beta_{RDi}\hat{p}_i^{RD}(k) + \sum_{i=1}^{N^{FT}} \beta_{FTi}\hat{p}_i^{FT}(k) \tag{5.51b}$$

where $\hat{p}_i^X(k)$ is a normalized number of platforms:

$$\hat{p}_i^X(k) = \frac{p_i^X(k)}{p_i^X(0)} \quad k = 0,1,2,3,\cdots,K\,. \tag{5.52}$$

The expressions in (5.51) are linear combinations of normalized platforms and express the objective of each force to maximize its own platforms and minimize the platforms of the opposing force. Now we call those unit commanders and top commanders in each force simply as a force leader. The controls at each stage k are chosen so as to maximize the above objective functions at stage k+1. In our model we have a finite set of control variables for every unit in the battle. This set is determined by the allowable relocate controls, the choices of targets, and the salvo size controls. In the one-step and two-step look-ahead methods, at each time k the forces consider their control options over only the next time step and the next two time steps, respectively.

(1) One-step looking ahead Nash control

In the one-step looking ahead Nash control, the Blue force will seek a control vector $u_k^{B^*}$ at time k that will maximize its objective function $J_{k,k+1}^B$ only at time $k+1$. Similarly, the Red force will seek a control vector $u_k^{R^*}$ that will also maximize its objective function $J_{k,k+1}^R$ only at time $k+1$. The Nash equilibrium strategies $u_k^{B^*}$ and $u_k^{R^*}$ for such a solution must therefore satisfy:

$$\begin{aligned} J_{k,k+1}^B(u_k^{B^*}, u_k^{R^*}) &\geq J_{k,k+1}^B(u_k^B, u_k^{R^*}) \qquad \forall \quad u_k^B \in U_k^B \\ J_{k,k+1}^R(u_k^{B^*}, u_k^{R^*}) &\geq J_{k,k+1}^R(u_k^{B^*}, u_k^R) \qquad \forall \quad u_k^R \in U_k^R \end{aligned}. \tag{5.53}$$

where U_k^B and U_k^R are sets of all available control choices at time k for Blue and Red, respectively. Following the expression used in (5.49), with $K = 1$, we see that expression (5.51a) rewritten as $J_{k,k+1}^B(u_k^B, u_k^R) = J_{k+1}^B(u_k^B, u_k^R)$ will become:

$$J_{k,k+1}^B(u_k^B, u_k^R) = \sum_{i=1}^{N^{BB}} \alpha_{BBi} \hat{p}_i^{BB}(k+1) + \sum_{i=1}^{N^{BF}} \alpha_{BFi} \hat{p}_i^{BF}(k+1) - \sum_{i=1}^{N^{RT}} \alpha_{RTi} \hat{p}_i^{RT}(k+1) - \sum_{i=1}^{N^{RD}} \alpha_{RDi} \hat{p}_i^{RD}(k+1) - \sum_{i=1}^{N^{FT}} \alpha_{FTi} \hat{p}_i^{FT}(k+1) \tag{5.54}$$

The objective function $J_{k,k+1}^R(u_k^B, u_k^R)$ is determined in the same fashion. Since the forces will seek to optimize only for one step at a time, this type of solution may be interpreted as a sequence of finite static game solutions.

(2) Two-step looking ahead Nash control

A more interesting solution is to let $K=2$ which corresponds to a two-step look-ahead problem. In this case the Blue and Red forces determine their control variables $u_k^{B^*}$ and $u_k^{R^*}$ at time k by maximizing the objective functions given by the expressions:

$$J_{k,k+2}^B = J_{k+1}^B + J_{k+2}^B \tag{5.55a}$$

$$J_{k,k+2}^R = J_{k+1}^R + J_{k+2}^R \,. \tag{5.55b}$$

where the right hand side terms are obtained from (5.51). In the case of the Blue force, this corresponds to:

$$\begin{aligned} J_{k,k+2}^B(u_k^B, u_{k+1}^B, u_k^R, u_{k+1}^R) = & \sum_{i=1}^{N^{BB}} \alpha_{BBi}\hat{p}_i^{BB}(k+1) + \sum_{i=1}^{N^{BF}} \alpha_{BFi}\hat{p}_i^{BF}(k+1) \\ & - \sum_{i=1}^{N^{RT}} \alpha_{RTi}\hat{p}_i^{RT}(k+1) - \sum_{i=1}^{N^{RD}} \alpha_{RDi}\hat{p}_i^{RD}(k+1) - \sum_{i=1}^{N^{FT}} \alpha_{FTi}\hat{p}_i^{FT}(k+1) \\ & + \sum_{i=1}^{N^{BB}} \alpha_{BBi}\hat{p}_i^{BB}(k+2) + \sum_{i=1}^{N^{BF}} \alpha_{BFi}\hat{p}_i^{BF}(k+2) \\ & - \sum_{i=1}^{N^{RT}} \alpha_{RTi}\hat{p}_i^{RT}(k+2) - \sum_{i=1}^{N^{RD}} \alpha_{RDi}\hat{p}_i^{RD}(k+2) - \sum_{i=1}^{N^{FT}} \alpha_{FTi}\hat{p}_i^{FT}(k+2) \end{aligned} \tag{5.56}$$

A similar expression can be derived for the Red force. In the two-step Nash approach, both sides look for sequences of two consecutive controls $\left(u_k^{B^*}, u_{k+1}^{B^*}\right)$ and $\left(u_k^{R^*}, u_{k+1}^{R^*}\right)$ that will satisfy the Nash equilibrium:

$$J_{k,k+2}^B(u_k^{B^*}, u_{k+1}^{B^*}, u_k^{R^*}, u_{k+1}^{R^*}) \geq J_{k,k+2}^B(u_k^B, u_{k+1}^B, u_k^{R^*}, u_{k+1}^{R^*}) \quad \forall \left(u_k^B, u_{k+1}^B\right) \in U_k^B \times U_{k+1}^B \tag{5.57a}$$

and

$$J_{k,k+2}^R(u_k^{B^*}, u_{k+1}^{B^*}, u_k^{R^*}, u_{k+1}^{R^*}) \geq J_{k,k+2}^R(u_k^{B^*}, u_{k+1}^{B^*}, u_k^R, u_{k+1}^R) \;\; \forall \left(u_k^R, u_{k+1}^R\right) \in U_k^R \times U_{k+1}^R \tag{5.57b}$$

where U_k^X is the set of all admissible controls for force X at time step k. After such sequences of control choices are found, only the controls at time k are actually implemented. The controls at time *k+1* are obtained by considering the same problem at the next step, i.e., for performance functions in the form $J^B_{k+1,k+3} = J^B_{k+2} + J^B_{k+3}$ and $J^R_{k+1,k+3} = J^R_{k+2} + J^R_{k+3}$, and so on. As such, this is a two-step moving horizon Nash solution.

Since the sets of all possible choices for the controls are finite, each of the one-step and two-step look-ahead Nash solutions, if it exists, can be determined from the corresponding bimatrix game representations. The Nash solution for bimatrix games does not always exist in pure strategies. If this situation occurs, the forces might then consider using a different solution strategy such as the Stackelberg solution [18] which is known to always exist in pure strategies. We should note that the one-step look-ahead approach does not really capture the dynamics of the operation and, hence, eliminates any possibility of using the relocate command. That is, the units will not be able to initiate a movement as a result of the optimization process. We rectify this by assigning a corridor for the Blue units, which guides each unit to a predetermined target. The two-step look-ahead approach includes some movement dynamics, but the units still have to be guided to the vicinity of their assigned engagement areas. In the next section, we will illustrate these concepts with an example.

5.2.3 Illustrative example

We consider a scenario that is taking place on a 10×10 square grid. Each square on the grid corresponds to roughly 40×40 square miles in dimensions. The Blue force consists of a group of three airborne units: one Blue Bomber (BB) unit and two Blue Fighters (BFs). The mission of the Blue force is to destroy one fixed target (FT) that is heavily defended by three Red Air Defense units (RDs). Two Red Troop units are also available in the area. The mission of the Blue force is considered accomplished when the fixed target is damaged by more than 40%.

After a successful mission, the Blue airplanes return to base. The Blue base is located in the upper right hand corner of the grid at coordinated (10, 10). We will assume that the Blue mission is planned for a maximum duration of 2 hours. For the type of airplanes used, and grid dimensions, we will use time steps of 5 minutes each in real time. The maximum duration of the mission will therefore correspond to 24 time steps. We will assume that when engagement occurs, the forces will continue optimizing their controls until the goal of the Blue force is accomplished or until the Blue units spend all available weapons before accomplishing the mission.

Table 5.1 summarizes the initial conditions (coordinates and force strength) for the scenario considered in the example. On the Blue side, the Bomber unit consists of 10 F4 bomber planes each equipped with 4 MK2 guided bombs, and the Fighter unit consists of 6 F2-E fighter planes each and each plane equipped with 4 air-to-ground missiles. On the Red side, each of the 3 Air Defense units consists of 7 platforms: one radar system and 6 SAM launchers. Each SAM launchers is equipped with 3 surface-to-air missiles. Thus, the average number of weapons per unit is 18/7=2.57. We also assume a maximum salvo size of one missile per launcher or 6/7=0.86 missile per unit. The troop units consist of 50 armored vehicles each and equipped with 3 shoulder-launched SAMs per vehicle. Finally, the fixed target is an airport with a total of 10 platforms (such as runways, command center, control tower, hangars, etc.) to be destroyed. As mentioned earlier, the mission of the Blue force is considered accomplished when at least 4 of the 10 airport platforms have been destroyed. The probabilities of kill PK_{ij}^{XY} for each unit on one side against units from the other side are given in Table 5.2. The values are given for the case when a "row" unit fires at a "column" unit. In our simulations, we will assume ideal weather conditions ($\beta_w = 1$).

Table 5.1 Initial conditions for the example

Unit	Type	Coordinates on the grid	Number of Platforms	Number of Weapons	Max. Salvo size
BB	F4 bombers	(8, 7)	10.0	4.0	1.0
BF_1	F2-E fighters	(8, 7)	6.0	4.0	1.0
BF_2	F2-E fighters	(8, 7)	6.0	4.0	1.0
RT_1	Armored vehicles	(5, 5)	50.0	3.0	0.5
RT_2	Armored vehicles	(5, 4)	50.0	3.0	0.5
RD_1	Fixed SAM & radar	(2, 2)	7.0	2.57	0.86
RD_2	Fixed SAM & radar	(2, 2)	7.0	2.57	0.86
RD_3	Fixed SAM & radar	(2, 2)	7.0	2.57	0.86
FT	Airport	(2, 2)	10.0	N/A	N/A

Table 5.2 Probabilities of kill for the example

	BB	BF_1	BF_2	RT_1	RT_2	RD_1	RD_2	RD_3	FT
BB	0	0	0	0.6	0.6	0.6	0.5	0.4	0.3
BF_1	0	0	0	0	0	0.8	0.7	0.7	0
BF_2	0	0	0	0	0	0.8	0.7	0.6	0
RT_1	0.2	0.1	0.1	0	0	0	0	0	0
RT_2	0.2	0.1	0.1	0	0	0	0	0	0
RD_1	0.7	0.3	0.3	0	0	0	0	0	0
RD_2	0.5	0.3	0.2	0	0	0	0	0	0
RD_3	0.5	0.2	0.2	0	0	0	0	0	0
FT	0	0	0	0	0	0	0	0	0

The objective functions are specified in the form of equation (5.51), where the weighting coefficients ($\alpha's$ and $\beta's$) for both Blue and Red forces are given in Table 5.3. From the coefficients in the objective function of the Blue force we can see that the highest priority for the Blue force ($\alpha = 1$) is given to damaging the fixed target. A high priority ($\alpha = 0.8$) is also assigned for the preservation of the bombers. On the other hand, for the Red force the highest priority ($\beta = 1$) is assigned for protecting the fixed target. High priorities are also given for preserving the RD_1 unit ($\beta = 0.7$) and destroying as many of the Blue bombers as possible ($\beta = 0.7$). Clearly, the (α and β) weighting coefficients in the objective functions can be adjusted by the top commander to investigate the outcome for any given set of mission priorities, and assumptions of priorities on the other side.

Table 5.3 Weighting coefficients in the objective functions for the example

	BB	BF_1	BF_2	RT_1	RT_2	RD_1	RD_2	RD_3	FT
Blue α_{Xi}	0.8	0.5	0.5	0.1	0.1	0.3	0.3	0.2	1.0
Red β_{Xi}	0.7	0.4	0.3	0.1	0.1	0.7	0.5	0.5	1.0

The initial conditions are summarized in Figure 5.5. The left hand side of the figure shows the location of the units on the two-dimensional grid, and the right hand side shows the number of platforms for each unit in bar chart form. We will show a few snapshots at specific time instants. In the one-step look-ahead simulation, the Blue units travel along a specified corridor towards the location of the fixed target and engage the Red defense units in that location. The controls that govern this engagement are calculated for both sides using the one-step look-ahead

Nash strategy described above. In Figure 5.6, we observe that all the Blue units enter the engage area together. After several time steps of engagement, the Blue group manages to inflict more than 40% damage to the fixed target and returns to base. Figure 5.7 shows the outcome at the end of the operation. As can be seen, on the Blue side 8 bombers and 4 fighters have been lost. On the Red side the third air defense unit was left undamaged while the first two have been almost completely destroyed. The Red troops are left intact since the Blue force decided to completely avoid them. We note that in Figures 5.5 through 5.10, the scale for RT1 and RT2 on the bar charts should be multiplied by a factor of 10.

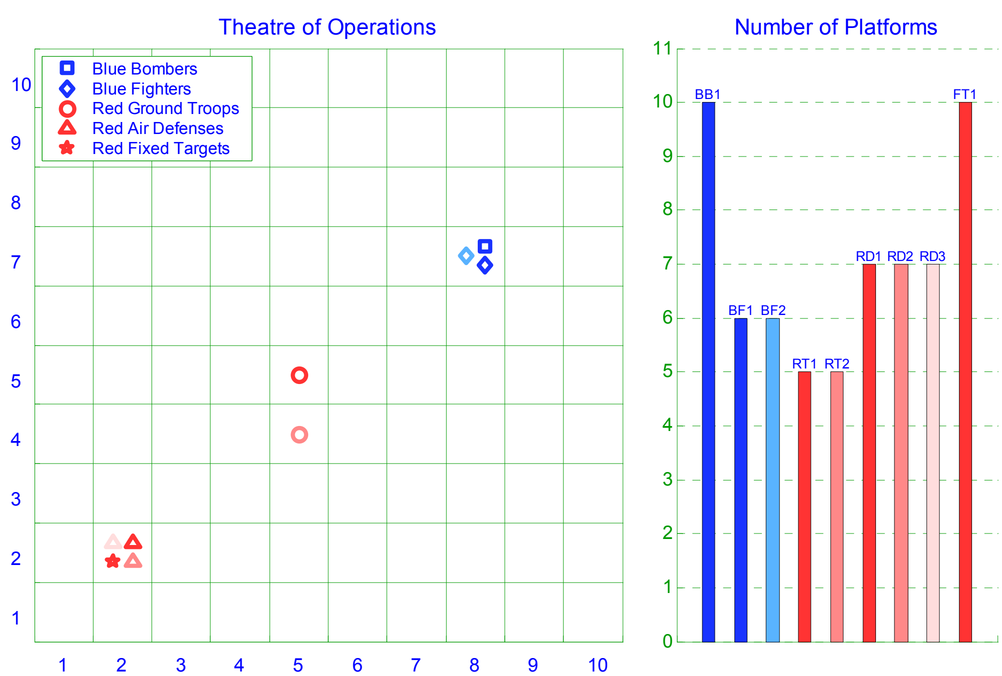


Figure 5.5 Initial states at k=0

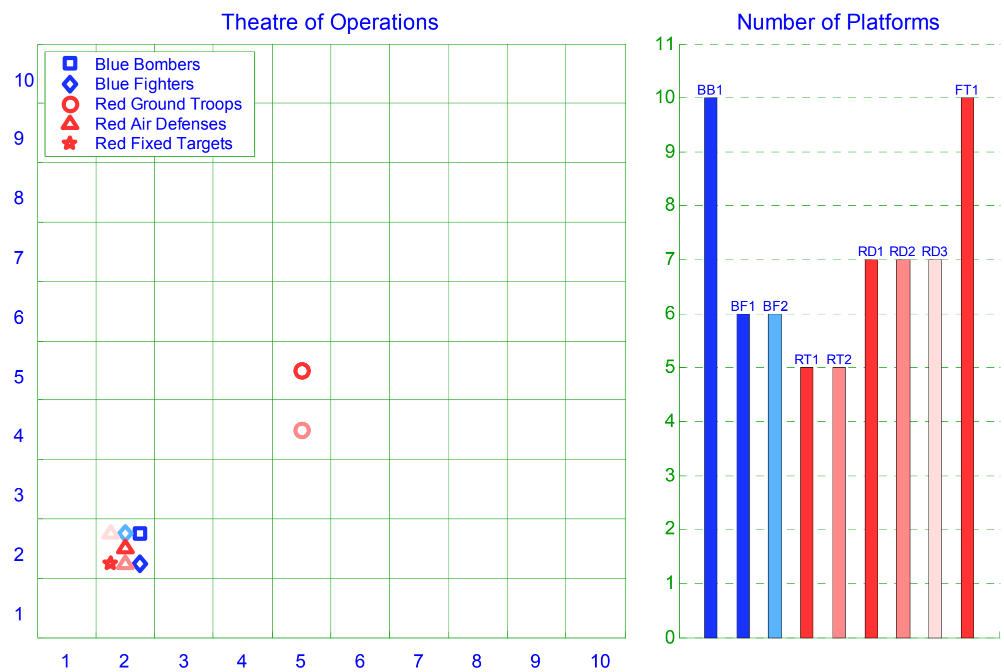


Figure 5.6 Attrition during full engagement at k=7

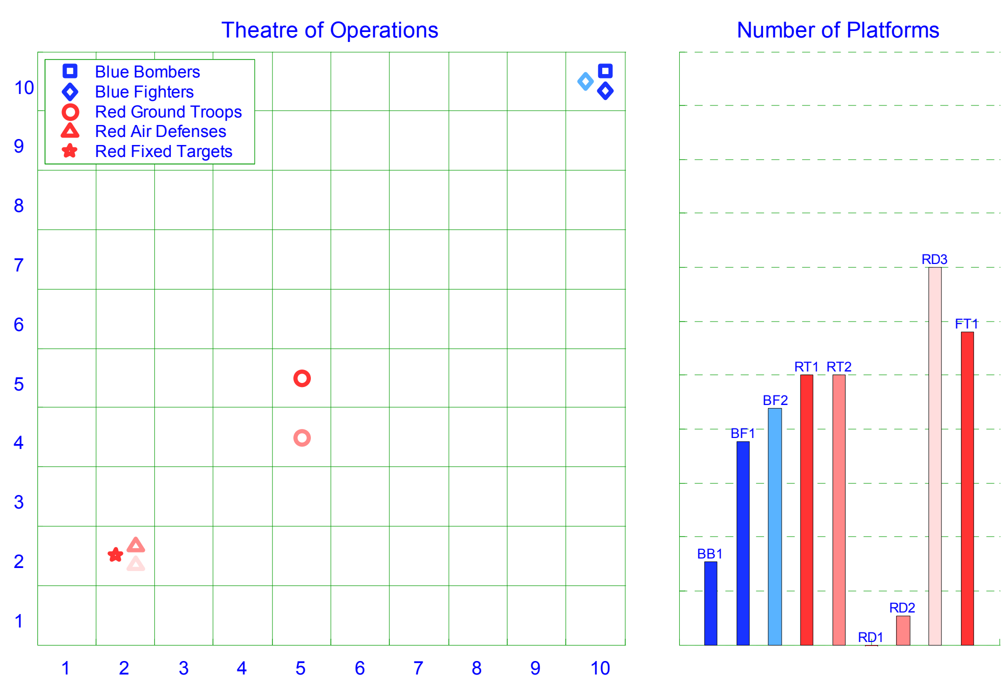


Figure 5.7 Final outcome at k=24 for the one-step look-ahead solution

We then consider the two-step look-ahead case, which was solved using dynamic programming. In this approach, at time step k, we determine all possible control options for both sides and compute all possible feasible states at time $k+1$. Then for each of these states, we repeat and compute all possible feasible states at time $k+2$. For each state at time $k+1$ that leads to a feasible state at time $k+2$, we then determine the Nash solution and the Nash costs-to-go for both forces. These costs are added to the values of the objective functions for the transition from time k to $k+1$, and the Nash solution recomputed at time k, considering all possible control options available at that time. We should note that even though this two-step look-ahead process yields control actions for the next two consecutive time steps, only the control actions for the first time step are implemented. The dynamic programming process is then repeated at the next time step.

In the simulation, as in the one-step case, the Blue airplanes follow a pre-specified corridor up to just one unit on the grid away from the location of the fixed target. At that point, as the results of the solution reveal, the Blue force uses the opportunity of optimizing for the next two time steps (i.e., next 10 minutes) and finds that the Nash optimal strategy is to send only the two fighter units first to engage the Red force and weaken its air defenses before sending in the bombers. This is consistent with what is known as the SEAD (Suppressing the Enemy Air Defenses) scenario. A snapshot of this can be seen in Figure 5.8. Clearly, there is an advantage for the Blue force to do so since the blue objective function J^B includes a high weight on preserving the bombers. Note that this is in contrast to the one-step look-ahead solution in which all Blue units (fighters and bombers) decided to engage the Red units at the same time, thus risking losing a large number of bombers since the Red air defenses have not yet been weakened, and they have a high priority towards destroying the Blue bombers. Figure 5.9 is a snapshot at the next time step. Here we see that the Blue bombers join the attack only after the Red defense units have been weakened. After several additional time steps the mission is accomplished when the fixed target is damaged by more than 40% and the Blue airplanes return to base. Figure 5.10 shows the outcome at the end of the operation. As can be seen, on the Blue side 6 bombers and 4 fighters have been lost. On the Red side the first two air defense units have been almost completely destroyed as in the one step look-ahead case, and the third unit was considerably more damaged than in the one step look-ahead case. The Red troops are still left intact since in this case the Blue force decided to also completely avoid them.

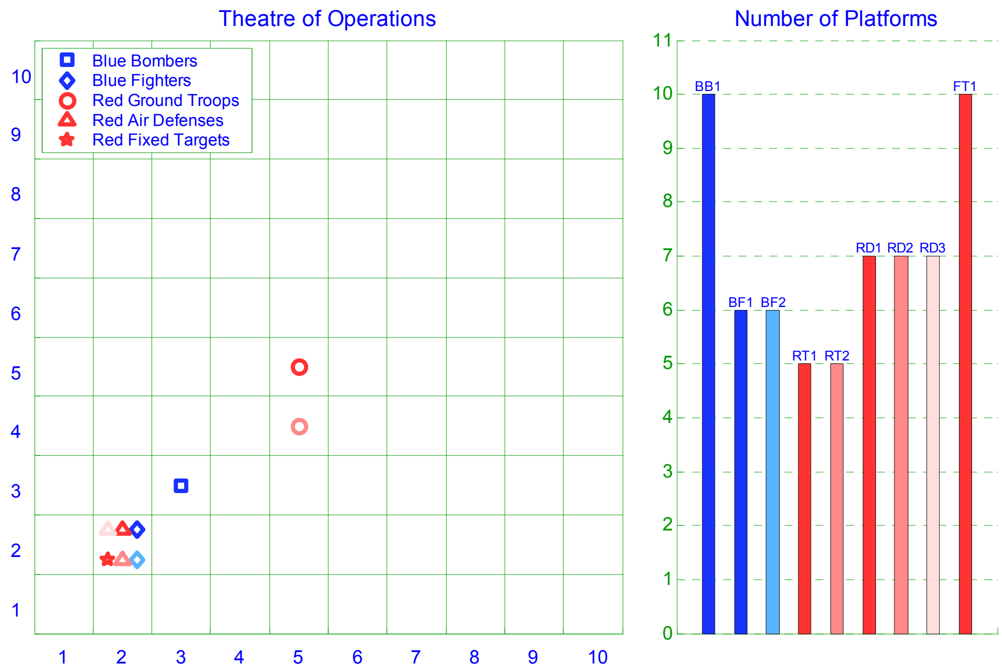


Figure 5.8 Fighters attack first at step $k=7$ while bombers wait in the two-step look-ahead solution

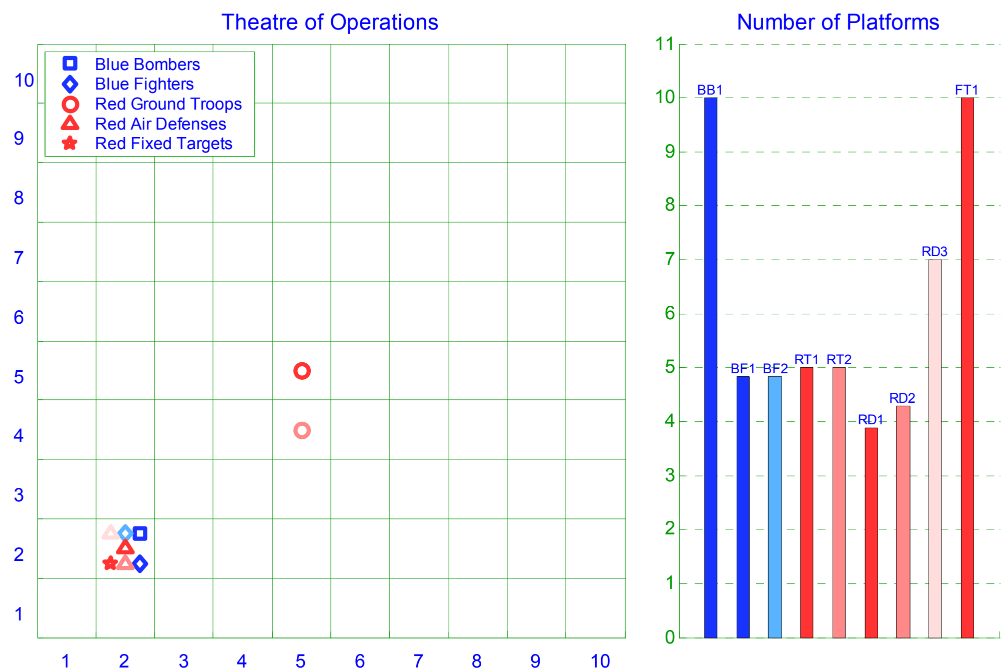


Figure 5.9 Bombers join the attack at step $k=8$ in the two-step look-ahead case

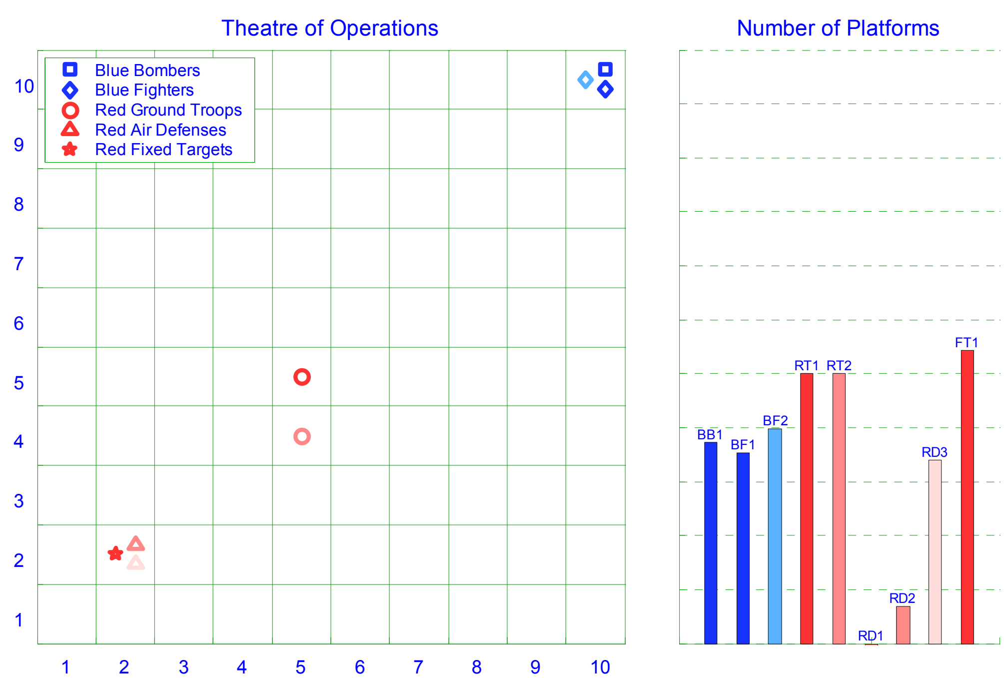


Figure 5.10 Final outcome at $k=24$ for the two-step look-ahead solution

A comparison of the final outcomes for the one-step and two-step approaches is given in Figure 5.11. From the perspective of the Blue side, the improvement in the two-step approach is obvious. The BF units were damaged a bit more in the two-step case, but the BB unit preserved considerably more platforms as a result of this better planned two-step look-ahead strategy. At the same time the Red air defenses suffered substantially more damage. The third RD unit suffered almost 50% damage in the two-step case. In the one-step case the RD_3 unit was left undamaged and still capable of doing significant damage to the Blue airplanes. Overall, it appears that the two-step look-ahead approach is a better strategy for the Blue force.

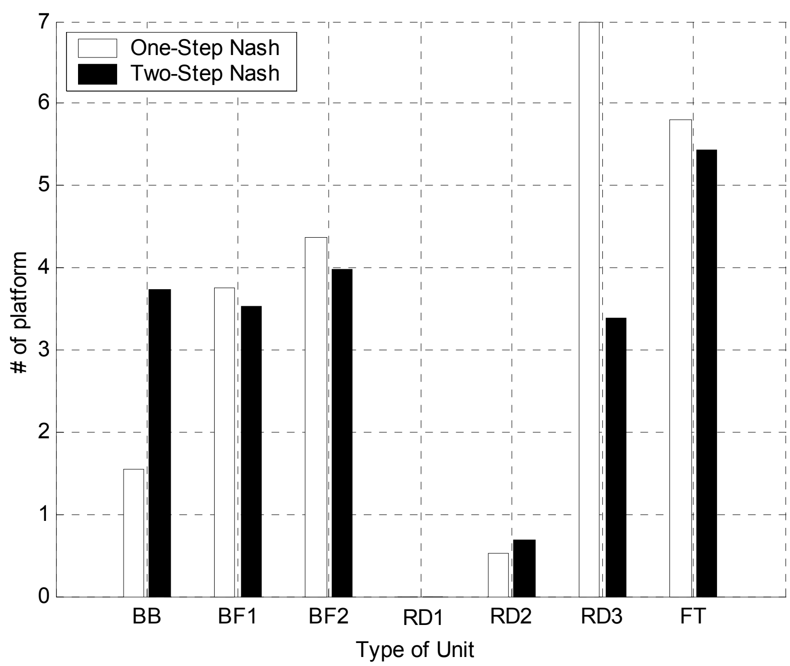


Figure 5.11 Comparison of the remaining platforms for all units in the one-step and two-step look-ahead approaches

The Nash solution is generally a balanced solution that does not favor one force over the other. In our case, the solution is not supposed to favor the Blue force over the Red force, and the outcome should in general depend only on the relative strengths of the forces. In this example, however, it is important to point out that in almost all cases there is a tendency for the Blue force to gain an advantage over the Red force when the optimizing window is extended from one to two steps. This appears to be more problem-specific than a general behavior. Clearly, in our example, the Blue airborne force is far more agile than the slow moving Red ground force. It is therefore reasonable to expect that the force with better moving capabilities be able to benefit from the dynamic nature of the optimization.

5.3 CONCLUSIONS

In this chapter, an attrition-type discrete-time nonlinear dynamic model is formulated for two opposing forces engaged in a military air operation, which is known as a good example of extended complex systems. We considered Nash strategies over a short, one or two-step look-ahead, moving horizon as a possible mechanism for overcoming the computational complexity in a practical situation. We performed our simulation tests and demonstrated the advantages of the two-step look-ahead Nash strategies over the one-step look-ahead Nash strategies. Our simulation results also proved that this attrition model is sound and it can be readily used to investigate the effectiveness of various game theoretic control strategies applied to a complex system with an intelligent adversary.

6.0 NASH STRATEGIES FOR DYNAMIC TEAM COMPOSITION AND DYNAMIC TASK ASSIGNMENT IN A MILITARY AIR OPERATION

As we mentioned before, in a large-scale extended complex system, different units may have different resources, and this leads to different capabilities and costs for handling the given tasks. In order to complete the various tasks more efficiently, the leader (or manager) often has to group the units into teams based on certain criteria, and allow them to cooperate with each other in order to enhance overall performance of the system. To organize the units into teams is also a natural way to reduce the complexity of the system from the leader's perspective. In general, dealing with N teams of M agents each may be much simpler than dealing with $N \times M$ agents. As we know, in the presence of an adversary such as the military dynamic system considered in the previous chapter, the situation becomes more complicated. For example, the leader of each force may divide his units into several teams each allocated a specific task. By teaming, the Blue force will organize all the Blue units in an efficient way in order to complete the assigned tasks, and the Red force will deploy all the Red defense parts to effectively protect the Red fixed target. In addition, a team division by one force needs to refer to the team composition by the other one. Thus, teaming is in the context of game. As the operation of the overall system progresses, a leader may reassess his initial task assignment among the teams and may decide that a different assignment could yield better overall performance of the system. In that case a reassignment of tasks and a redeployment of resources will have to be performed. These problems are known as the dynamic resource allocation problems in a complex system.

Dynamic team composition and dynamic task assignment are very important, but very complex, issues of dynamic resource allocation in a multi-team system, and thus need to be considered in any control architecture of large dynamic multi-team systems. We introduce several useful strategies for cooperative teaming and dynamic task assignment in this chapter, including NNS. In the previous chapter, the effects of teaming and tasking are not evident because of the limited number of fixed targets and other units in the scenario set-ups. In this chapter, we will focus on the applications of these strategies to the military operation system.

6.1 NASH REASSIGNMENT STRATEGIES

The problem of allocating resources and assigning tasks in multi-team systems is an extremely important step in insuring that maximum overall performance of the system is achieved. A mechanism that allows for reallocation of resources and reassignment of tasks is important in the control of complex dynamic systems especially when the initial deployment of resources and assignment of tasks appear to be ineffective in yielding satisfactory results. In other words, a reassignment of tasks and a redeployment of resources will have to be performed. In a similar manner, when a specific team completes its initial assignment, the leader may consider two options. He may decide to terminate this team's control activity (i.e. retire the team), or reassign the team to another ongoing task. In the former case, the control of the system will continue, but with fewer teams, and in the latter, the team may be merged with one of the remaining teams to help improve its ability to complete its task. These complicated issues need to be considered in any control architecture that involves a multitude of teams and tasks. In the model of the military operation developed in the previous chapter, there are several tasks that need to be performed on each side of the engagement. For example, a typical task for the attacking force may involve destroying a specific part of a fixed, or moving, target on the defending side. The model allows for the possibility of teaming on each side for the purpose of accomplishing the required tasks. The fighting units on each side can be teamed up and allocated specific tasks to accomplish. In that case, a problem will arise if some of the teams are able to accomplish their tasks successfully and others are not. For example, a situation of this type may occur when a weak team is assigned to a difficult task that it cannot accomplish on its own. It is therefore natural for the commander to consider reassigning those teams that are still capable, after successfully finishing their tasks, to join the remaining teams. In some cases, even if a team is able to complete its task on its own, the associated costs and the overall system performance may vary drastically if those teams that accomplish their tasks first are reassigned to the remaining tasks rather than if they are left inactive afterwards. The commander may therefore consider reassigning teams that have accomplished their tasks first to cooperate with the remaining teams in order to accelerate the accomplishment of the overall mission of the force.

In this section, the reassignment problem in multi-team multi-task dynamic systems, specifically as encountered by a commander in a military operation, is investigated based on the model developed in the previous chapter. We consider the reassignment problem and use the moving-horizon Nash strategies to formulate possible solutions for it. We present two simulation examples to illustrate the advantages of the Nash reassignment strategies.

6.1.1 Problem formulation

We begin by considering a general task reassignment problem for the Blue force. Let us assume that there are m distinct fixed targets, each occupying a specific location on the grid and defended by specific units of the Red force. Destroying a fixed target and weakening its defending units is defined as a task for the Blue force. When there is only one fixed target on the Red side, the Blue commander will assign the entire Blue force to that task. When the number of targets is greater than one, the commander may partition the Blue force into teams and decide which team will be assigned to which task. Let us assume that the Blue force is divided into n teams $\left\{T_1^B, T_2^B, \cdots, T_n^B\right\}$. Each team consists of a combination of Blue units (bombers and fighters). The objective function of team T_i^B at stage *k*, denoted by $J_i^B(k)$, is given by a subset of expression $J^B(k)$ in (5.51). We assume that each team has a pre-assigned task. If some teams accomplish their tasks before others, instead of returning to base, the commander has the option of reassigning them to other, either new or ongoing, tasks. Let $I_c(k)$ denote the set of indices of teams that have accomplished their tasks at stage *k*. For $i \in I_c(k)$, let $t_i^B(k)$ denote the task that team T_i^B can be re-assigned to. The number of possible combinations of assignments of teams who have accomplished their tasks to unaccomplished tasks can grow exponentially as will be explained later. Let $r(i,k) > 0$ be the cost of reassigning team T_i^B ($i \in I_c(k)$) to the new task $t_i^B(k)$ at stage *k*. Thus, the optimal re-assignment problem at stage *k* can be formulated as:

$$\underset{K-1)]}{} \tilde{J}_{k,K}^B \quad \text{where} \quad \tilde{J}_{k,K}^B = \sum_{l=k}^{K} \left[\sum_{i \notin I_c(l)} J_i^B(l) \quad \sum_{i \in I_c(l)} \right. \tag{6.1}$$

In (6.1) the control $u^B(k) =$ for (k), which basically says that the control vector in (5.15) has been augmented by the choice of a new task $t_i^B(k)$. It is clear that the optimal control actions $u \quad k \quad u^*(K - \;]$ taken by the Blue teams also depend on the controls of the Red force and hence the problem will need to be considered within the framework of game theory as will be discussed in the next section. In other words, the solution will continue to be game-theoretic in nature. We will maintain the Nash strategy as the approach to obtain the optimal reassignment controls for any Blue team that has been reassigned. Once $t_i^B(k)$ is determined, the units in team T_i^B will move to the location of the new task.

Let n_a and $n_b(k)$ be the number of teams to be re-assigned and the number of unaccomplished tasks at time k, respectively. The number of task choices $k)$ for the i^{th} re-assigned team is equal to $n_b(k)+1$, i.e., the number of unaccomplished tasks plus the choice of returning to base. Thus, the number of all possible combinations of task choices for the re-assigned teams at time k is $(n_b(k)+1)^{n_a(k)}$. Clearly, this number will grow exponentially with increasing and $n_b(k)$ adding another complexity to the task reassignment problem. To reduce it, one way is to allow those re-assigned teams to select the unaccomplished tasks near their current locations only, and thus the cost of any reassigned path can be ignored in the objective functions.

6.1.2 Moving-horizon Nash reassignment solution

Because of the computational complexity involved, even in cases that do not involve reassignment, determining a solution for problems of this type over the entire time horizon K is not in general numerically feasible. In order to reduce the computational complexity in determining the controls, instead of maximizing the objective functions from stage k to the

vectors $\tilde{u}^{B^*}($ and $u^{R^*}(k)$ at time k that will maximize the objective functions over a reduced look-ahead moving horizon of length K_r steps ($K_r << K$):

$$\tilde{J}^{B}_{k,k+K_r} = \sum_{l=k}^{k+K_r} \left[\sum_{i \notin I_c(l)} J_i^B(l) - \sum_{i \in I_c(l)} r(i,l) \right] \tag{6.2a}$$

$$J^{R}_{k,k+K_r} = \sum_{l=k}^{k+K_r} J^R(l) \tag{6.2b}$$

We should note that the one-step look-ahead approach does not effectively capture the dynamics of the air operation and, hence, eliminates any possibility of optimizing the relocate command of the Blue force. The two-step look-ahead approach, on the other hand, includes some optimization over the relocate command, though limited to only two time intervals ahead of the present time. Clearly, whenever reassignment is necessary the two-step look-ahead strategy enables the Blue commander to make more effective decisions in the sense that the unnecessary losses of the reassigned teams can be reduced.

In this section, we do not intend to address the entire range of issues related to the re-assignment problem. Instead, we will focus on the following two situations that require reassignment [41]:

- Situation 1: Some teams cannot complete their pre-assigned tasks on their own.
- Situation 2: Some teams can complete their pre-assigned tasks but with a heavy cost in time and losses.

In both of these situations, the commander may consider reassigning a team that has completed its task to one or more of these "weaker" teams. We will explore these characteristics and the advantages of the Nash reassignment strategies in the following illustrative examples.

6.1.3 Illustrative examples

We consider a scenario where the Blue force consists of two groups of Blue bombers, BB1 and BB2, and two groups of Blue Fighters, BF1 and BF2. The Red force includes two adjacent fixed targets, FT1 and FT2, (e.g., two bridges) defended by four groups of Red defense units (RD1, …, RD4) and one group of Red troops (RT1). Let us consider an initial assignment, as shown in Table 6.1, where Blue is divided into two teams. Team 1 includes BB1 and BF1 and is assigned FT2, and Team 2 includes BB2 and BF2 and is assigned FT1. The task of a Blue team is considered accomplished when its assigned fixed target loses at least 40% of its platforms. After a task is accomplished, the corresponding team will either be reassigned or will be returned to base (located in the upper right corner of the grid). The initial states are shown in Figure 6.1. To illustrate the results of the Nash Reassignment Strategies based on this scenario, we will discuss two examples, corresponding to the two different situations of reassignment mentioned in section 6.1.2.

Table 6.1 Initial deployment for the example

Unit	Type	location	Platforms	Weapons	Max.Salvo
BB1	F4 bombers	(5,5)	7	4	1
BB2	F4 bombers	(6,10)	7	4	1
BF1	F2-E fighters	(5,5)	8	4	1
BF2	F2-E fighters	(6,10)	6	3	1
RT1	Armored vehicles	(4,5)	50	3	0.5
RD1	Fixed SAM & Radar	(2,4)	6	15/6	5/6
RD2	Fixed SAM & Radar	(2,4)	7	18/7	6/7
RD3	Fixed SAM & Radar	(3,3)	6	15/6	5/6
RD4	Fixed SAM & Radar	(3,3)	7	18/7	6/7
FT1	Bridge	(2,4)	10	N/A	N/A
FT2	Bridge	(3,3)	10	N/A	N/A

In both examples, the simulations are performed in MATLAB using Nash type two-step look-ahead moving controls.

Example 6.1: In this example, we consider probabilities of kill for each pair of units as given in Table 6.2, and weighting coefficients in the objective functions of both Blue and Red force as given in Table 6.3.

a) At first, the simulation is performed without the possibility of reassignment. The final outcome of this simulation is shown in Figure 6.2. We see that Team 1 returned to base after accomplishing its task, but Team 2 exhausted all its weapons and could not accomplish its task since more than 60% of FT1's platforms remain undamaged.

b) We then performed the same simulation except that the top commander now decides to re-assign Team 1, after it accomplishes its task, to join Team 2. Figure 6.3, shows a snapshot of how this is accomplished. We see that in the first step, upon joining Team 2, BF1 is very effective in increasing Team 2's ability to weaken the defense units around FT1. In the next step, we see that BB1 now joins in the attack of FT1. This can be clearly seen in Figure 6.4. In Figure 6.5, we can see that FT1 is damaged to 40% and the task of Team 2 has now been accomplished with help from Team 1.

c) Figure 6.6 gives a comparison of the remaining number of platforms in the two simulations discussed above. It is clear that the reassignment of Team 1, after it finished its task against FT2, to join Team 2, not only helps that Team complete its task against FT1 but also saves more platforms of BB2 and BF2 in Team 2, while BB1 only suffers a little more damages than that in the simulation without using the reassignment strategies.

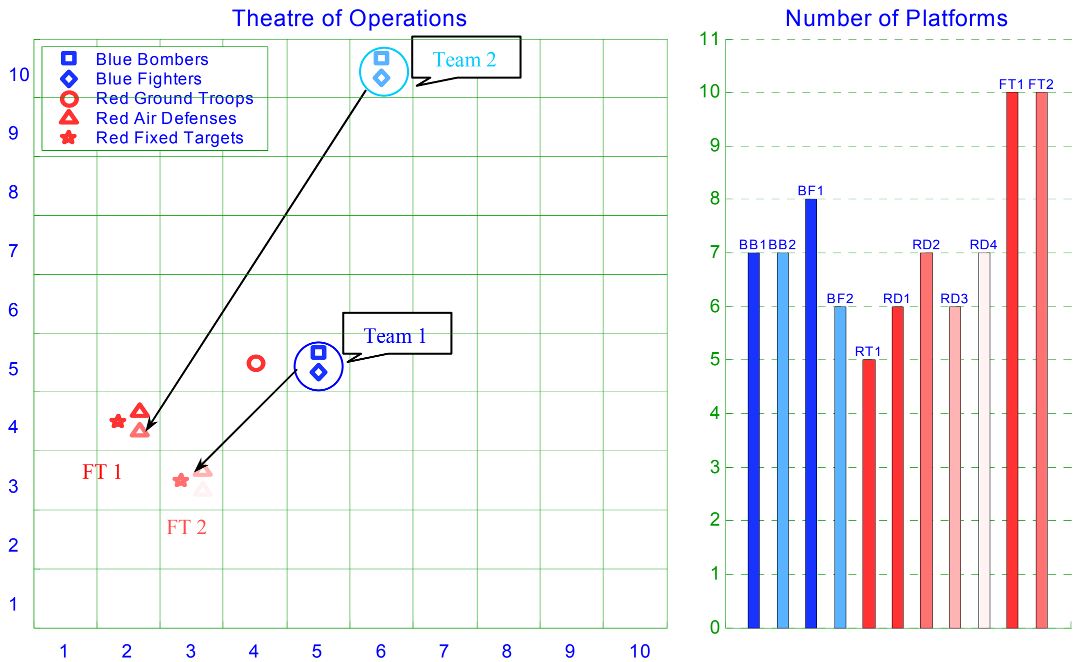


Figure 6.1 Initial states for the example

Table 6.2 Probabilities of kill for Example 6.1

	BB1	BB2	BF1	BF2	RT1	RD1	RD2	RD3	RD4	FT1	FT2
BB1	0	0	0	0	0.6	0.5	0.4	0.6	0.5	0.4	0.6
BB2	0	0	0	0	0.6	0.5	0.4	0.5	0.4	0.3	0.5
BF1	0	0	0	0	0	0.8	0.8	0.8	0.8	0	0
BF2	0	0	0	0	0	0.7	0.7	0.7	0.7	0	0
RT1	0.2	0.2	0.1	0.1	0	0	0	0	0	0	0
RD1	0.7	0.7	0.3	0.3	0	0	0	0	0	0	0
RD2	0.5	0.5	0.2	0.2	0	0	0	0	0	0	0
RD3	0.5	0.5	0.15	0.15	0	0	0	0	0	0	0
RD4	0.6	0.6	0.15	0.15	0	0	0	0	0	0	0
FT1	0	0	0	0	0	0	0	0	0	0	0
FT2	0	0	0	0	0	0	0	0	0	0	0

Table 6.3 Weighting coefficients in the objective functions for Example 6.1

	BB1	BB2	BF1	BF2	RT1	RD1	RD2	RD3	RD4	FT1	FT2
Blue α_{Xi}	0.8	0.4	0.2	0.1	0.1	0.3	0.2	0.3	0.3	1	1
Red β_{Xi}	0.7	0.7	0.4	0.3	0.1	0.7	0.5	0.5	0.5	1	1

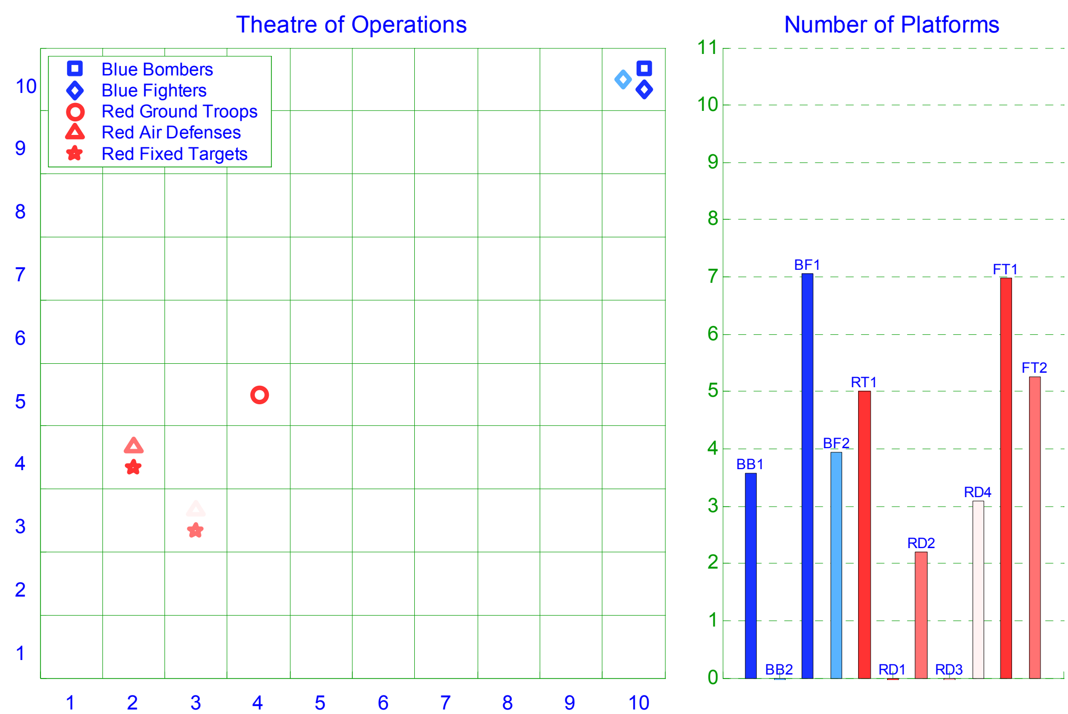


Figure 6.2 Final states without reassignment in Example 6.1

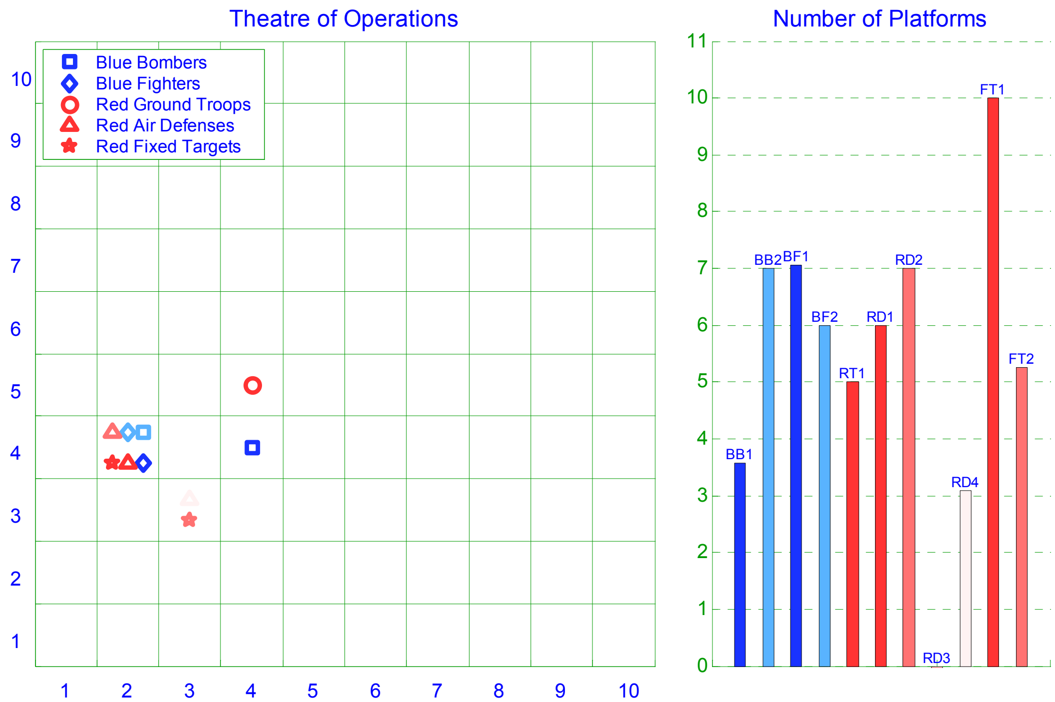


Figure 6.3 Effect of BF1 joining Team 2 in Example 6.1

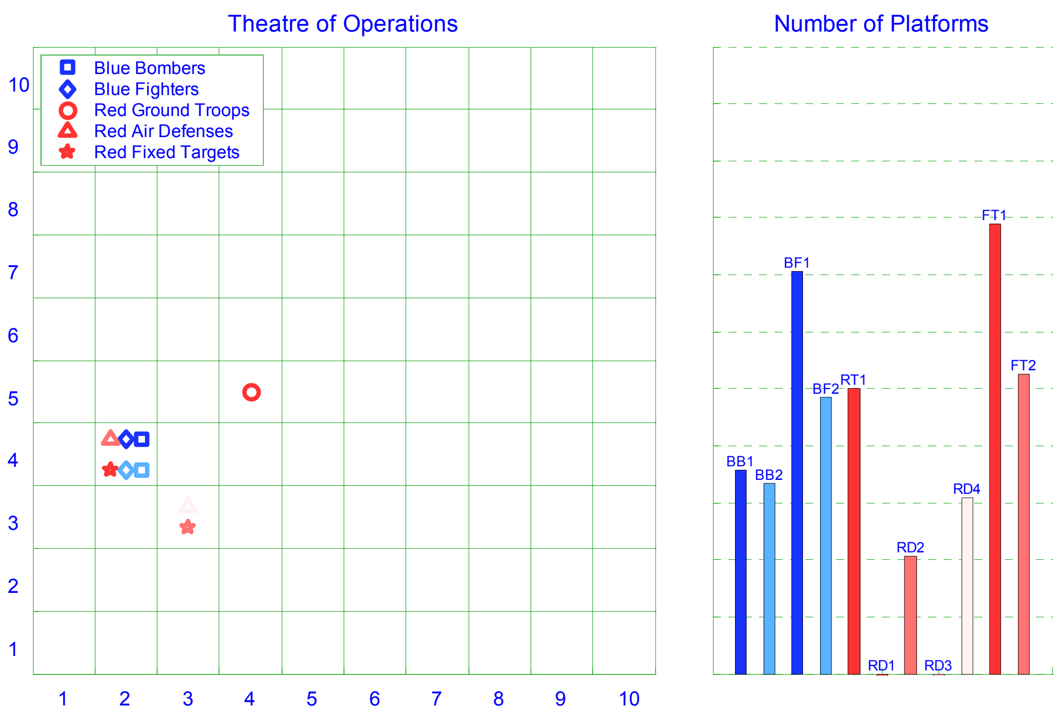


Figure 6.4 Effect of BB1 joining Team 2 in Example 6.1

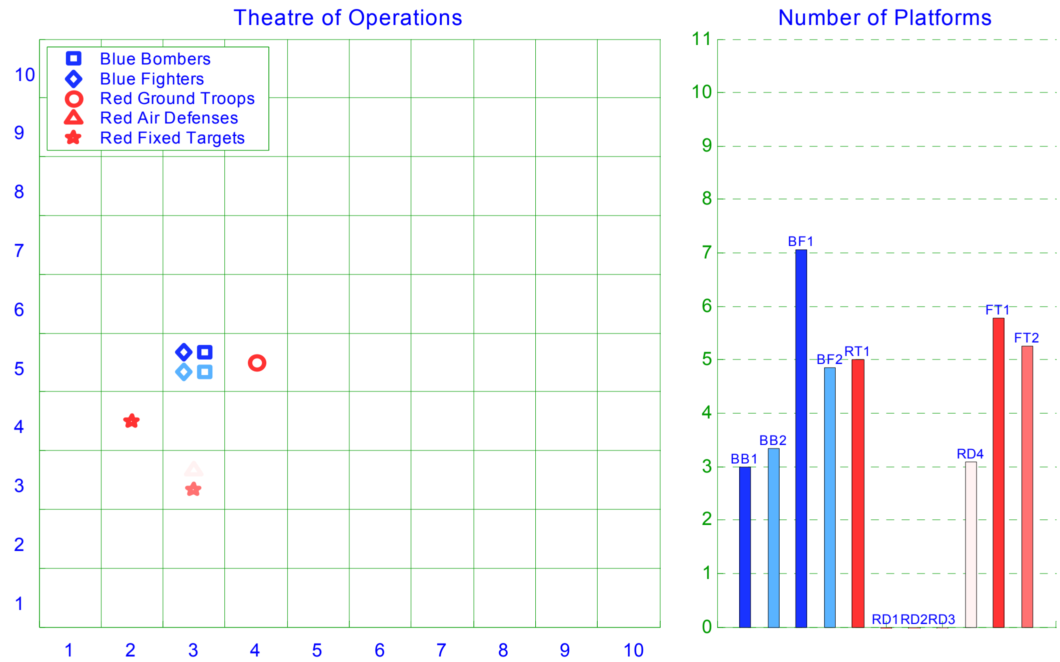


Figure 6.5 Team 1 and Team 2 accomplish Team 2's task in Example 6.1

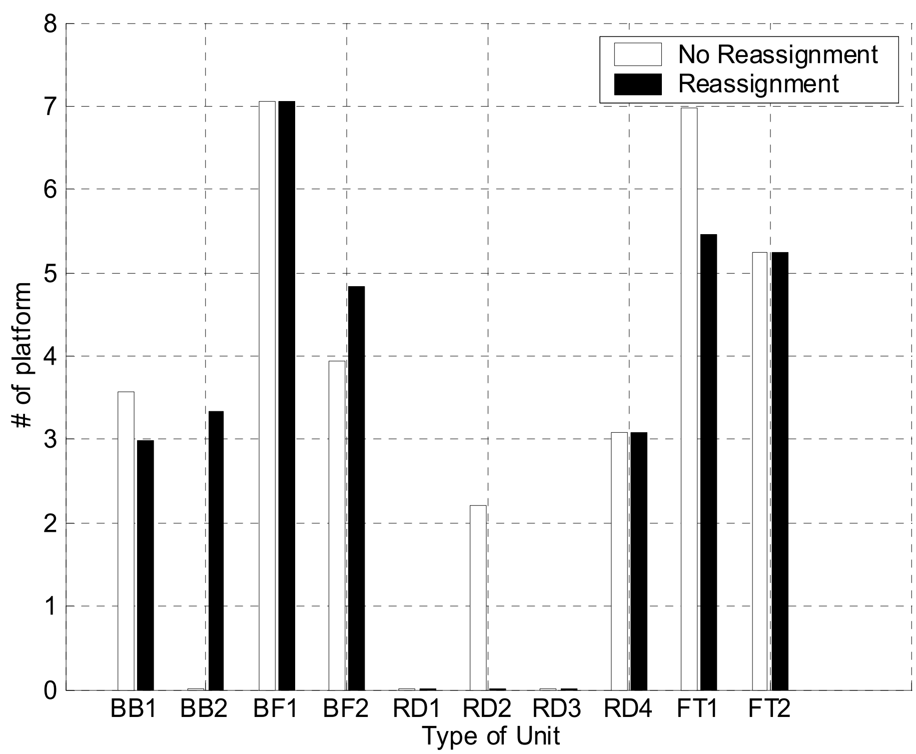


Figure 6.6 Comparison of the remaining platforms in Example 6.1

Example 6.2: In this example, we modify the values of probabilities of kill and weighting coefficients in the objective functions. These are now shown in Tables 6.4 and 6.5, respectively. The reason for doing this is to enhance Team 2's ability to accomplish its task without Team 1's help. When there is no reassignment strategy in the simulation, we indeed see that Team 2 can now finish its task without the help of Team 1. This is illustrated in Figure 6.7. We note, however, that it takes seven time steps for Team 2 to accomplish this task, and this may not be considered satisfactory. The top commander then decides to reassign Team 1, after finishing its task, to join Team 2. In Figure 6.8, we see that upon joining Team 2 BF1 is active first, and Figure 6.9 shows the last step in which FT1 is destroyed. It is interesting to note that, during the entire period when Team 1 is reassigned, the BB1 unit remains inactive since it appears that only BF1 is needed by Team 2 to accomplish its task. Also, only five steps are now required to accomplish Team 1's task resulting in a saving of two time steps. Comparing the results of these two situations in Figure 6.10, we note that, as in the first example, the choice of reassignment also saves more platforms of BB2 and BF2 in Team 2 and destroys more units of RD1 and RD2.

Table 6.4 Probabilites of kill for Example 6.2

	BB1	BB2	BF1	BF2	RT1	RD1	RD2	RD3	RD4	FT1	FT2
BB1	0	0	0	0	0.6	0.5	0.4	0.5	0.5	0.4	0.6
BB2	0	0	0	0	0.6	0.5	0.4	0.5	0.4	0.3	0.5
BF1	0	0	0	0	0	0.8	0.8	0.8	0.8	0	0
BF2	0	0	0	0	0	0.8	0.8	0.8	0.7	0	0
RT1	0.2	0.2	0.1	0.1	0	0	0	0	0	0	0
RD1	0.7	0.7	0.3	0.3	0	0	0	0	0	0	0
RD2	0.5	0.5	0.2	0.2	0	0	0	0	0	0	0
RD3	0.5	0.5	0.15	0.15	0	0	0	0	0	0	0
RD4	0.6	0.6	0.15	0.15	0	0	0	0	0	0	0
FT1	0	0	0	0	0	0	0	0	0	0	0
FT2	0	0	0	0	0	0	0	0	0	0	0

Table 6.5 Weighting coefficients in the objective functions for Example 6.2

	BB1	BB2	BF1	BF2	RT1	RD1	RD2	RD3	RD4	FT1	FT2
Blue α_{Xi}	0.8	0.6	0.2	0.1	0.1	0.4	0.4	0.4	0.4	1	1
Red β_{Xi}	0.7	0.7	0.4	0.3	0.1	0.7	0.5	0.5	0.5	1	1

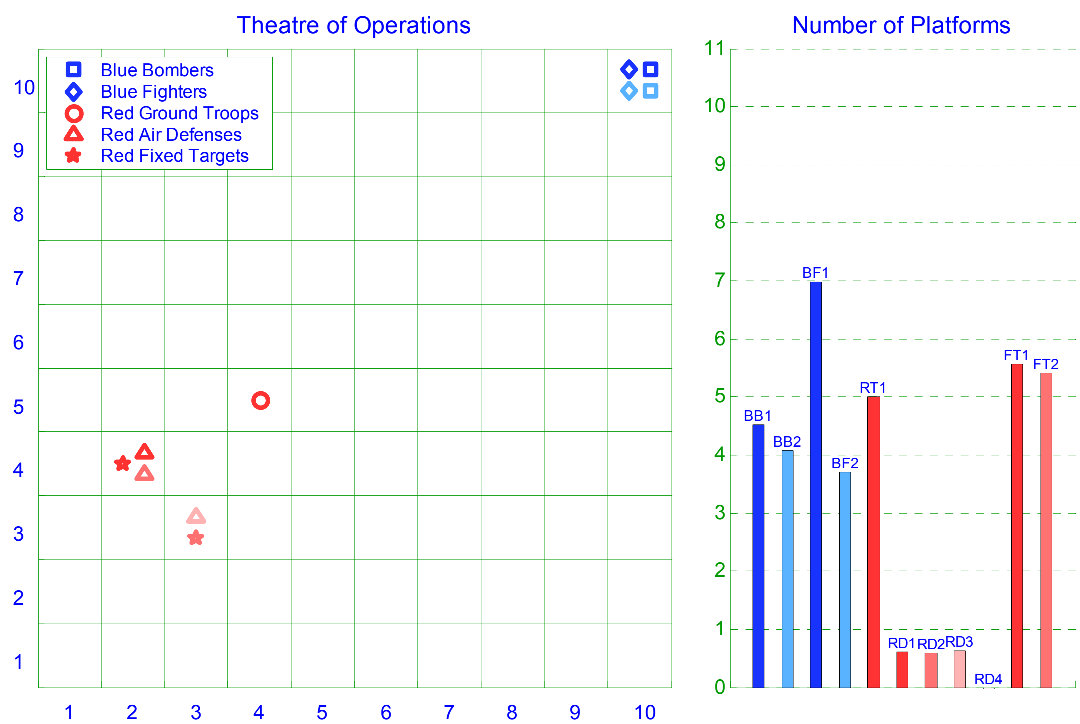


Figure 6.7 Final states without reassignment in Example 6.2

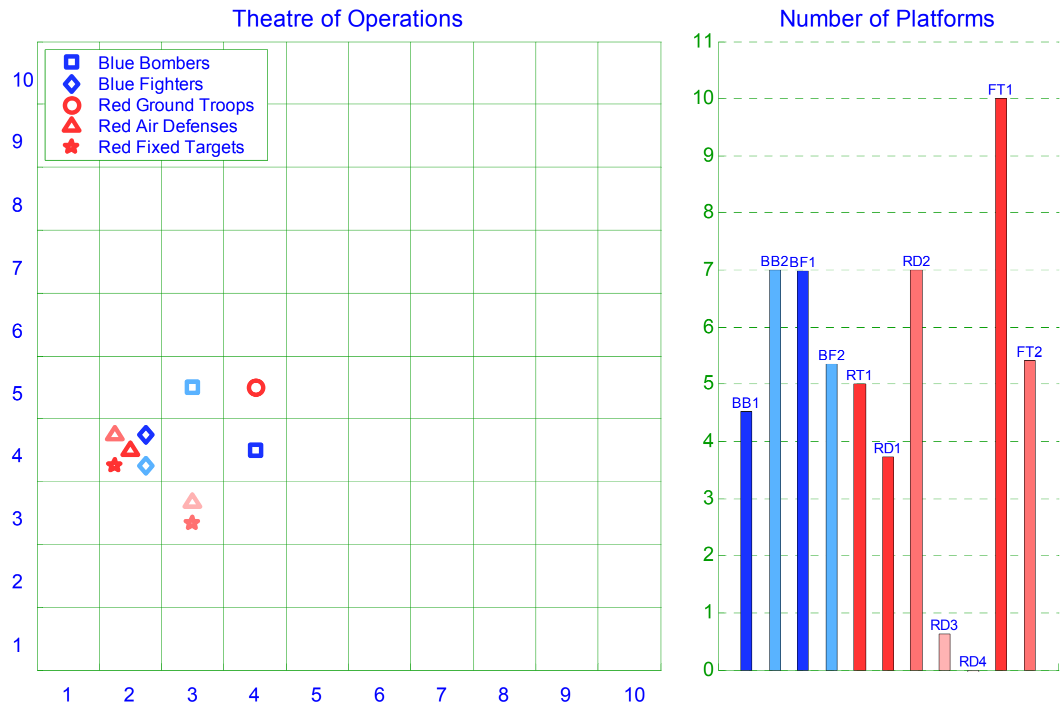


Figure 6.8 Effect of BF1 joining Team 2 in Example 6.2

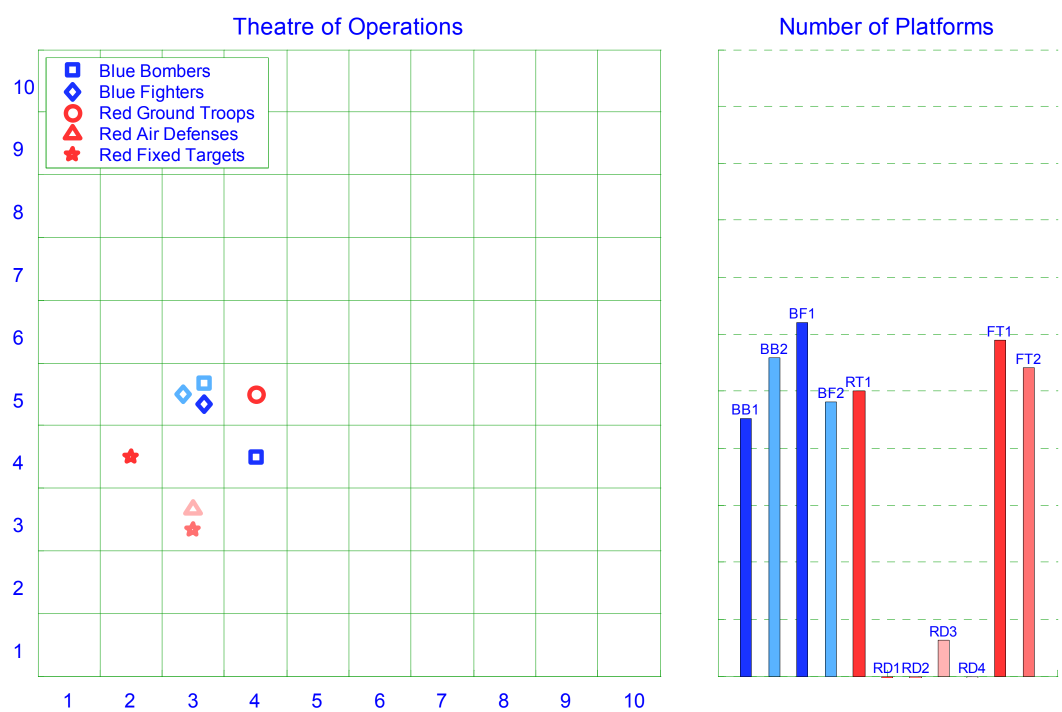


Figure 6.9 Team 2 accomplishes its task only with the help of BF1 in Example 6.2

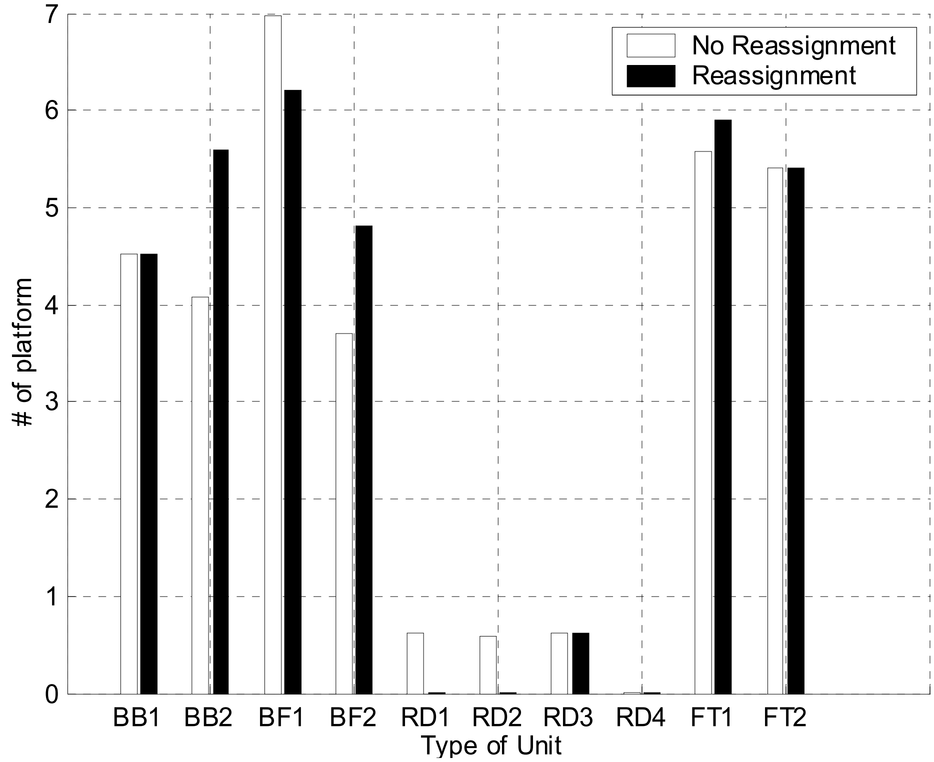


Figure 6.10 Comparison of the remaining platforms in Example 6.2

6.2 NASH ORDINAL STRATEGIES

In the previous discussion, we assume that the initial deployment of Red defense parts and the initial team composition of the Blue force and the initial task assignment of these teams are well known to both the Blue force and the Red force. Under this assumption, game-theoretic approaches such as Nash and Nash reassignment strategies can be applied to optimize the operation procedure. In a real batter, however, there are often several possible ways for the Red force to deploy his parts and for the Blue force to organize his units into teams. Different combinations of the initial choices taken by the top commanders in both forces may result in different outcomes. One force usually may not know the initial decision made by the other one exactly. Thus, both commanders would like to consider the outcomes resulting from all the possible initial situations of both sides at first and then make their decisions. Clearly, the outcome resulting from one top commander's options also depend on the choices of the other one. In other words, the decision of the initial team composition and task assignment made by the Blue top commander is required to consider the decision of the initial deployment made by the Red top commander, and vise versa. The problem here still can be formulated as a game. Such game is a little different from the games we mentioned before because the top commanders may not have evident mathematic expressions for their goals as those objective functions given by (5.51). The top commanders may rank, instead of a concrete calculation, the outcomes from various choices relying on their experiences. These games are known as ordinal games [42]. In this section, we will apply the Nash strategies in the ordinal game theory, called *Nash Ordinal Strategies* (NOS), to determine the initial deployment for the Red defense parts and the task pre-assignment and team composition for the Blue force [43].

We consider a scenario where the Blue force has two groups of Blue bombers, BB1 and BB2, and two groups of Blue fighters, BF1 and BF2. The Red force includes two adjacent fixed targets, FT1 and FT2, (e.g., a refinery and a bridge) defended by four groups of Red defense units (RD1, …, RD4) and one group of Red troops (RT1). The description for the units is shown in Table 6.6. The probabilities of kill for each unit pair are given in Table 6.7.

Table 6.6 Description of units in the example

Unit	Type	Platforms	Weapons	Max.Salvo
BB1	F4 bombers	7	4	1
BB2	F4 bombers	7	4	1
BF1	F2-E fighters	8	4	1
BF2	F2-E fighters	6	3	1
RT1	Armored vehicles	50	3	0.5
RD1	Fixed SAM & Radar	6	15/6	5/6
RD2	Fixed SAM & Radar	7	18/7	6/7
RD3	Fixed SAM & Radar	6	15/6	5/6
RD4	Fixed SAM & Radar	7	18/7	6/7
FT1	Building	10	N/A	N/A
FT2	Bridge	10	N/A	N/A

Table 6.7 Probabilities of kill for the example

	BB1	BB2	BF1	BF2	RT1	RD1	RD2	RD3	RD4	FT1	FT2
BB1	0	0	0	0	0.6	0.5	0.4	0.5	0.5	0.4	0.6
BB2	0	0	0	0	0.6	0.5	0.4	0.5	0.4	0.3	0.5
BF1	0	0	0	0	0	0.8	0.8	0.8	0.8	0	0
BF2	0	0	0	0	0	0.7	0.7	0.7	0.7	0	0
RT1	0.2	0.2	0.1	0.1	0	0	0	0	0	0	0
RD1	0.7	0.7	0.3	0.3	0	0	0	0	0	0	0
RD2	0.6	0.6	0.2	0.2	0	0	0	0	0	0	0
RD3	0.5	0.5	0.15	0.15	0	0	0	0	0	0	0
RD4	0.6	0.6	0.15	0.15	0	0	0	0	0	0	0
FT1	0	0	0	0	0	0	0	0	0	0	0
FT2	0	0	0	0	0	0	0	0	0	0	0

We note that BB1 has stronger capabilities of destroying the red fixed targets than BB2 because the probabilities of FT1 and FT2 being killed by BB1 are greater than those of FT1 and FT2 being killed by BB2, respectively. With a similar reason, BF1 is stronger than BF2 in destroying the red defense units. Moreover, RD1 and RD2 are stronger than RD3 and RD4 in killing the blue units. Let us consider several reasonable initial choices for both forces. The Blue commander has three options of team composition and task pre-assignment, which are listed in Table 6.8. The Red commander has also three options of deployment as shown in Table 6.9.

Table 6.8 Options of Blue for team composition and task assignment for the example

Options for Blue Commander	Teaming	Task Assignment
Option X	Team 1: BB1 and BF1	FT1
	Team 2: BB2 and BF2	FT2
Option Y	Team 1: BB1 and BF1	FT2
	Team 2: BB2 and BF2	FT1
Option Z	All Blue units in one Team	FT1 then FT2

Table 6.9 Options of Red deployment for the example

Options for Red Commander	Defending FT1	Defending FT2
Option A	RD1 and RD2	RD3 and RD4
Option B	RD1 and RD3	RD2 and RD4
Option C	RD1, RD2 and RD3	RD4

For example, in option X, the Blue top commander divides Blue units into two teams: team 1 includes BB1 and BF1, assigned to attack FT1, and team 2 includes BB2 and BF2, assigned to attack FT2. After a task is accomplished, the corresponding team will either be reassigned or will be returned to base. In option B, the Red top commander deploys RD1 and RD3 to defend FT1, and assigns RD2 and RD4 to defend FT2. The initial states are shown in Figure 6.11. Nine simulations corresponding to 3×3 combinations are performed using two-step look-ahead Nash and Nash reassignment strategies. Each pair of Blue and Red choices leads to a specific battle damage. Table 6.10 gives the weights coefficients in the objective functions.

Table 6.10 Weighting coefficients in the objective functions for the example

	BB1	BB2	BF1	BF2	RT1	RD1	RD2	RD3	RD4	FT1	FT2
Blue α_{Xi}	0.8	0.6	0.2	0.1	0.1	0.4	0.4	0.4	0.4	1	1
Red β_{Xi}	0.7	0.7	0.4	0.3	0.1	0.7	0.5	0.5	0.5	1	1

Figure 6.12 gives the total remaining platforms of each type of units for the nine simulations. For each Blue choice and each Red choice there is an ordered preferential ranking for the Blue commander and an ordered preferential ranking for the Red Commander, based on a battle damage assessment. Figure 6.13 gives this ranking matrix. The Blue commander ranks the outcome of the option Y as the best choice if the Red force selects the option B because the most blue bombers and blue fighters are preserved in this case. The Red commander prefers the outcome of option C if the Blue force chooses the option X since most of defense units are saved in this situation. We observe that the Nash ordinal strategy in this game is (Y, A), i.e., the Blue

commander will assign the strong team of BB1 and BF1 to attack FT2 and the weaker team of BB2 and BF2 to attack FT1, and the Red commander will deploy the strong units of RD1 and RD2 to defend FT1 and the weaker units of RD3 and RD4 to protect FT2. Figuring out a Nash ordinal strategy can be done simply by only ranking each column choices for the Blue force and ranking each row choices for the Red force [42]. By doing this, the matrix in Figure 6.13 becomes the matrix in Figure 6.14. Note that only 3 options need to be compared and ranked at one time instead of 9 options in the previous way. It is no doubt, that (Y,A) is still the Nash ordinal strategy in this game.

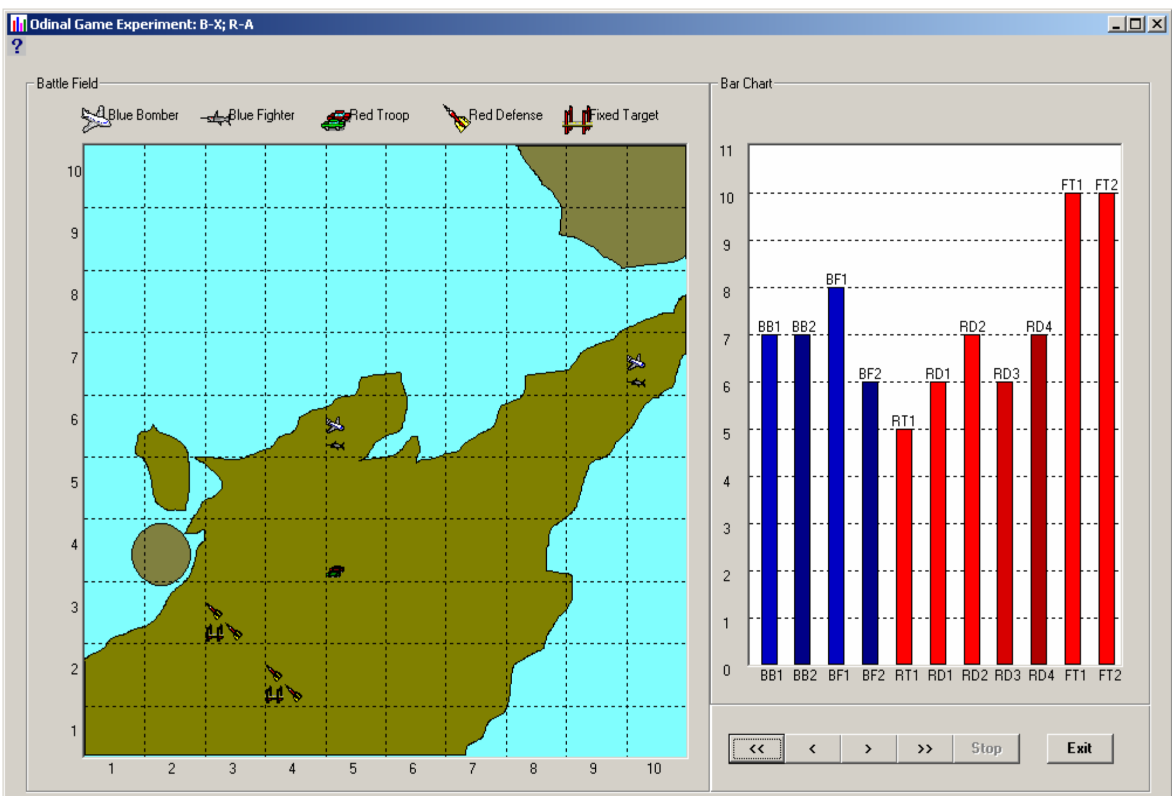


Figure 6.11 Initial situation for the example

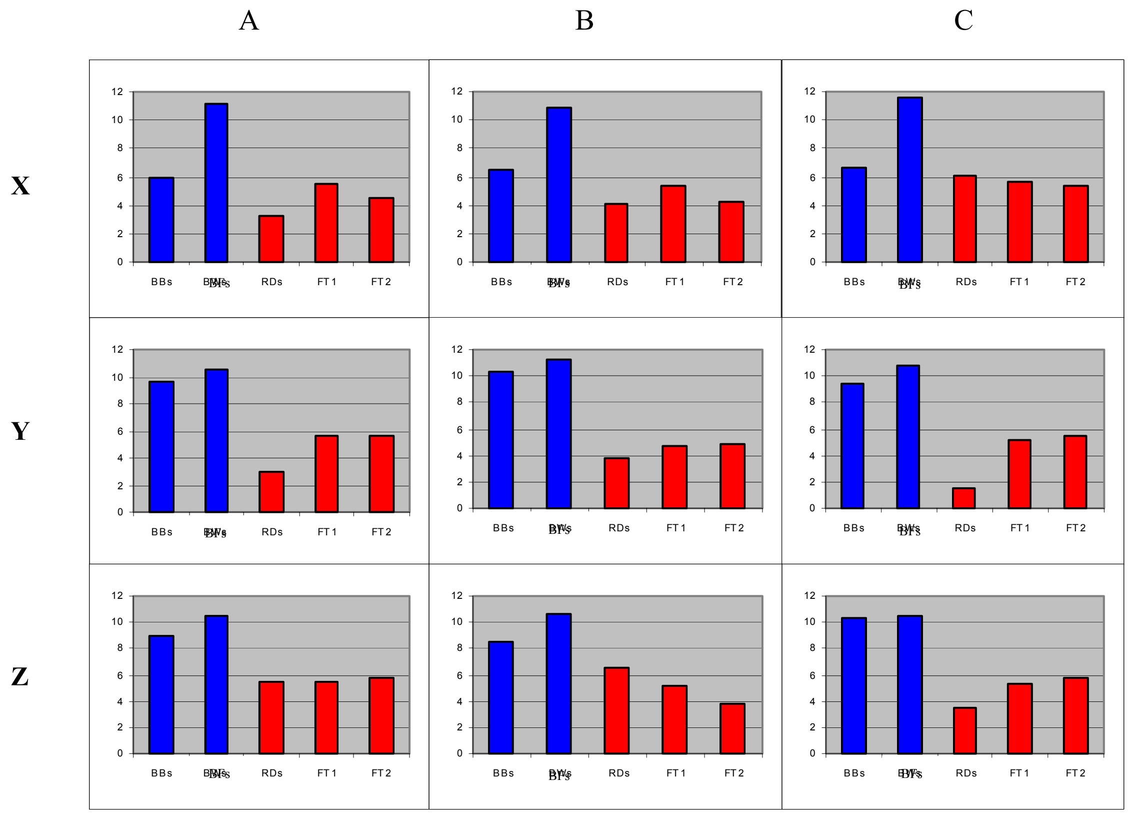


Figure 6.12 Outcomes of 3 X 3 options for the example

		Red Commander		
		A	B	C
Blue Commander	X	(9,3)	(7,2)	(8,1)
	Y	(5,6)	(1,7)	(3,9)
	Z	(6,5)	(4,4)	(2,8)

Figure 6.13 Ranking matrix in an ordinal game

		Red Commander		
		A	B	C
Blue Commander	X	(3,3)	(3,2)	(3,1)
	Y	(1,1)	(1,2)	(2,3)
	Z	(2,2)	(2,1)	(1,3)

Figure 6.14 Simplified ranking in an ordinal game

6.3 EFFECTS OF RESOURCE CAPABILITIES ON COOPERATIVE TEAMING STRATEGIES

Cooperative teaming is a very important issue in the optimization of large-scale multi-team systems, especially when there is an adversary affecting the outcome of the optimization. If the Blue units happen to be much weaker than the Red units and are assigned to destroy a strongly defended Red fixed target first, the Blue force may be completely demolished before it has a chance to move on to other missions. It is therefore natural for the leader of the Blue force to consider efficiently teaming its limited resource to complete its overall goal as much as possible. In that case, as the leader of the attacking entity, the Blue top commander may ask the questions: "What kind of Blue team composition and task assignment is most effective against the Red units?" or "Will there be an advantage for the Blue force in teaming its units against the Red adversary?" In other words, an important question for the Blue commander to ask is: Is it always necessary to group its units into cooperating teaming? In this section, we will attempt to answer these questions in the context of the military air operation model developed in the previous chapter. This section will use several simulations based on the model of an air military operation to illustrate varying the resource capabilities available to each team can considerably influence the effectiveness of the team composition [44].

In this section, we will consider the problem of team composition and task assignments faced by the Blue top commander. That is, how should the Blue force be divided into teams and what should the team tasks be in order for the Blue force to optimally achieve its objectives? We will attempt to answer these questions by considering the following specific scenario: the Blue force consists of two groups of Blue bombers, BB1 and BB2, and two groups of Blue Fighters, BF1 and BF2. The Red force includes two adjacent fixed targets, FT1 and FT2, defended by three groups of Red defense units RD1, RD2 and RD3 and one group of Red troop RT1. The description and initial equipment for each unit are listed in Table 6.11. We consider probabilities of kill for each pair of units as given in Table 6.12. From Table 6.12, we observe that the group of bombers BB1 has stronger capability against the red fixed targets than the group BB2. In addition, the group of fighter planes BF1 is more effective against RD1 – RD3 and RT1 than the group BF2. Also, note that the Red troops (RT1) are not as effective as the Red defense units (RD1- RD3) against the Blue units. In order to test the various teaming options that the Blue top commander may have, we will consider the following specific deployment of the Red forces. Clearly, in a real situation, the Red top commander may also have several options of teaming his forces as well. We will assume that FT1 is defended by the Red defense units RD1 and RD2, and FT2 is defended by the Red troops (RT1) and the Red defense unit RD3. This can be seen in Figure 6.15. For the purpose of simplicity, the deployment for the Red force is kept unchanged with respect to the varying options for the Blue force. By examining Table 6.12, it is clear that FT1 is strongly defended and FT2 is weakly defended since the probabilities of the Blue units being destroyed by RD1 – RD3 are much larger than those by RT1. This deployment is reasonable for the Red commander if FT1 is more important to defend than FT2.

Table 6.11 Description and initial equipment of units for the example

Unit	Type	Platforms	Weapons	Max.Salvo
BB1	F4 bombers	10	4	1
BB2	F4 bombers	10	4	1
BF1	F2-E fighters	2	4	1
BF2	F2-E fighters	2	3	1
RT1	Armored vehicles	50	3	0.5
RD1	Fixed SAM & Radar	7	18/7	6/7
RD2	Fixed SAM & Radar	7	18/7	6/7
RD3	Fixed SAM & Radar	7	18/7	6/7
FT1	Bridge 1	10	N/A	N/A
FT2	Bridge 2	10	N/A	N/A

Table 6.12 Probabilities of kill for the example

	BB1	BB2	BF1	BF2	RT1	RD1	RD2	RD3	FT1	FT2
BB1	0	0	0	0	0.7	0.6	0.6	0.6	0.7	0.7
BB2	0	0	0	0	0.7	0.6	0.6	0.6	0.3	0.3
BF1	0	0	0	0	0.6	0.8	0.8	0.8	0	0
BF2	0	0	0	0	0.5	0.7	0.7	0.7	0	0
RT1	0.1	0.1	0.1	0.1	0	0	0	0	0	0
RD1	0.4	0.4	0.4	0.4	0	0	0	0	0	0
RD2	0.4	0.4	0.4	0.4	0	0	0	0	0	0
RD3	0.4	0.4	0.4	0.4	0	0	0	0	0	0
FT1	0	0	0	0	0	0	0	0	0	0
FT2	0	0	0	0	0	0	0	0	0	0

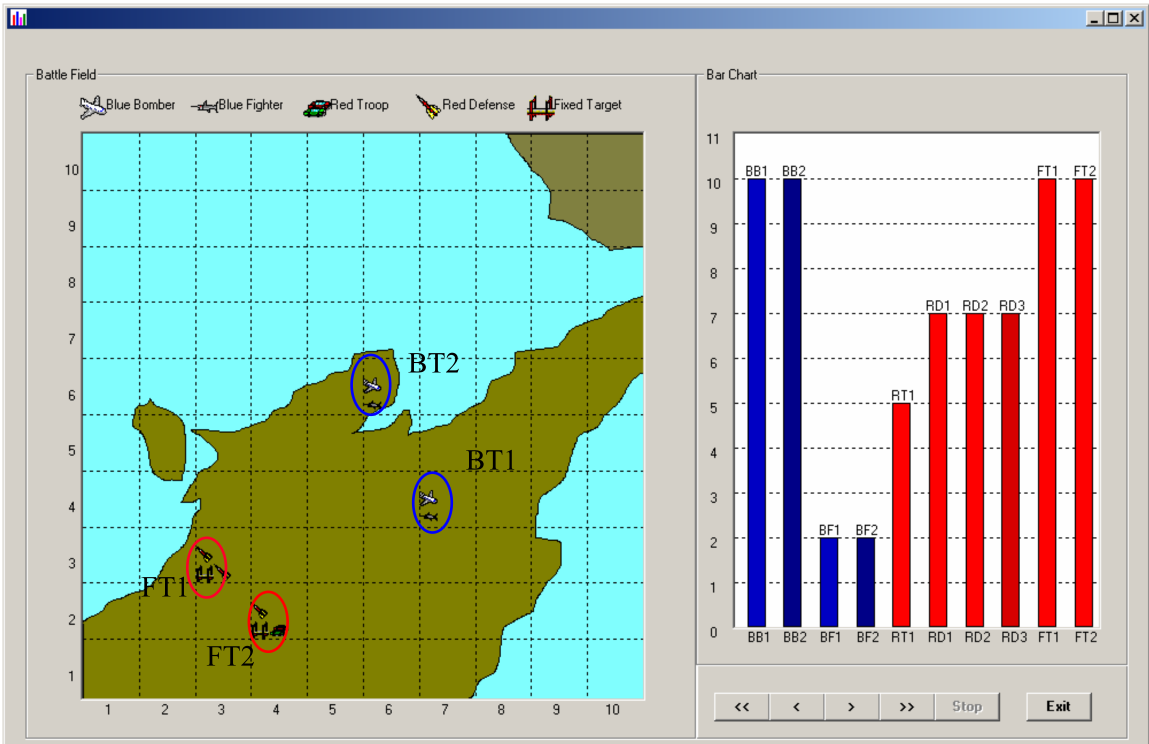


Figure 6.15 Initial states for the example

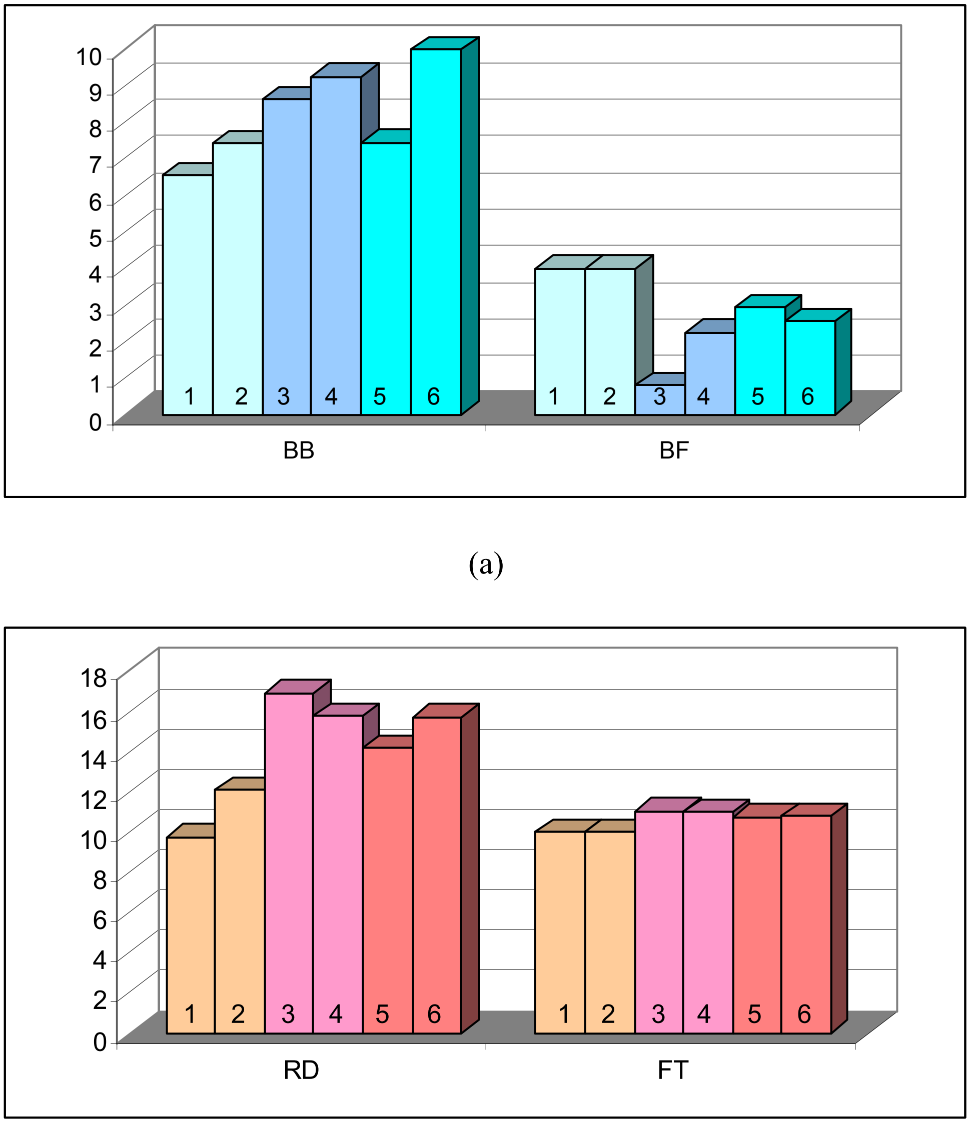
Now let us consider six different team compositions and task assignments for the Blue top commander as shown in Table 6.13. For example, Blue can be divided into two teams. Team 1 consisting of BB1 and BF1 is assigned to fixed target FT1 and Team 2 consisting to BB2 and BF2 is assigned to Fixed target FT2. This particular option (Option 4 or Option 6 in Table 6.13) essentially consists of teaming the strong blue units together and assigning them to the strongly defended target and teaming the two weak Blue units together and assigning them to the weakly defended target. In the different options, the simulations are performed in MATLAB using Nash type two-step look-ahead moving controls. If there is cooperative reassignment, the simulations will also use Nash Reassignment two-step look-ahead strategies. The various weights in the objective functions (5.51) used in the simulations are given in Table 6.14. The comparisons of remaining platforms for the Blue units and the Red units for the six options of Table 6.13 are shown in Figure 6.16.

Table 6.13 Options for Blue commander for the example

Options for Blue Commander	Teaming	Task Assignment	Is cooperative reassignment used?
Option 1	No Teaming	FT2 then FT1	
Option 2	No Teaming	FT1 then FT2	
Option 3	Team 1: BB1 and BF1	FT2	No
	Team 2: BB2 and BF2	FT1	
Option 4	Team 1: BB1 and BF1	FT1	No
	Team 2: BB2 and BF2	FT2	
Option 5	Team 1: BB1 and BF1	FT2	Yes
	Team 2: BB2 and BF2	FT1	
Option 6	Team 1: BB1 and BF1	FT1	Yes
	Team 2: BB2 and BF2	FT2	

Table 6.14 Weighting coefficients in the objective functions for the example

	BB1	BB2	BF1	BF2	RT1	RD1	RD2	RD3	FT1	FT2
Blue α_{Xi}	0.9	0.9	0.1	0.1	0.25	0.25	0.25	0.25	1	1
Red β_{Xi}	0.7	0.7	0.2	0.2	0.1	0.5	0.5	0.5	0.5	0.5



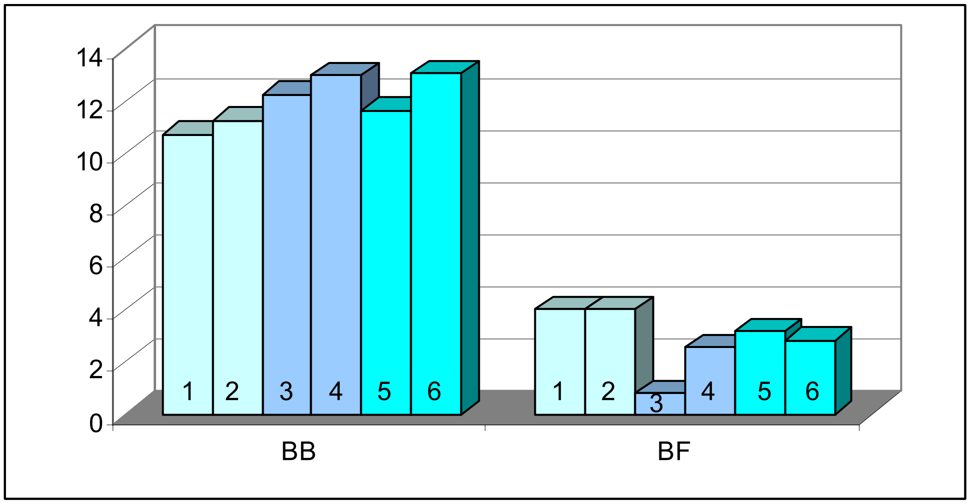
(a)

(b)

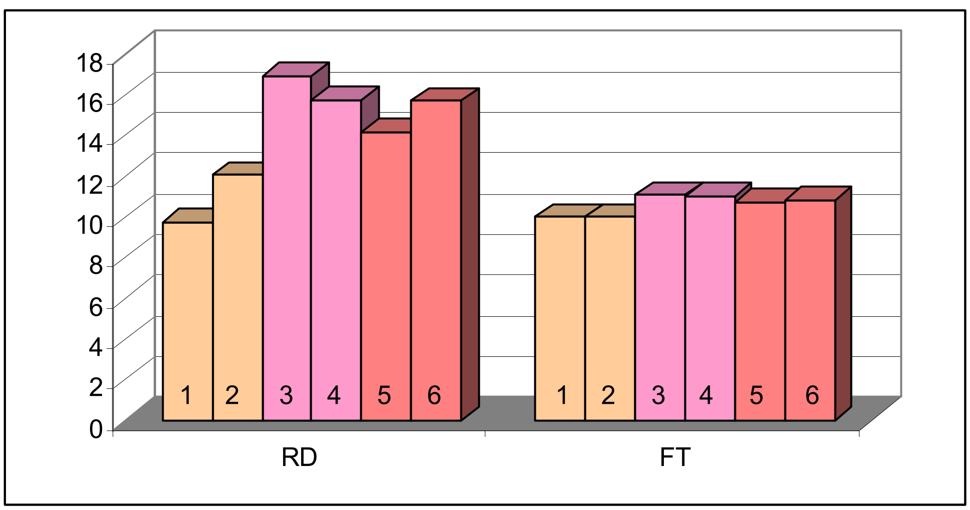
Figure 6.16 Comparison of remaining units for options 1- 6

In each option, the Blue units completed the required mission by destroying more than 40% of the red fixed targets. However, the remaining Blue units and Red units vary considerably among the six options. This can be seen in Figures 6.16. If the Blue commander wishes to preserve more of his bombers, then option 6, i.e., dispatching the stronger team to attack the strongly defended fixed target with reassignment, seems to be the best option since the number of the remaining BB platforms in this option is the greatest for this scenario. If there is no teaming at all such as the situation in option 1, the blue side will lose 4 more blue bombers than that in option 6. In addition, the reassignment strategies used in options 5 and 6 save more Blue fighters than those in option 3 and option 4 without reassignment, respectively. In this situation, cooperative teaming and task assignment such as option 6 would be a wise choice for the Blue commander in terms of preserving more Blue bombers.

In the previous scenario set-up, we note that the group of BB2 has a very small probability of kill against the red fixed targets. In other words, some blue units have very limited capabilities to destroy the targets. With limited resources, the overall planning, involving team division, task assignment and team reallocation, becomes a very important issue for the blue commander in the sense that different plans may lead to significantly different outcomes. Our question now is: is it necessary to consider cooperative teaming and task assignment if the blue units are all very strong? In order to answer this question, let us make the Blue units stronger and run the simulations again. We increased the probabilities of red fixed targets being destroyed by the group of BB2 from 0.3 to 0.5. Comparisons of the remaining platforms for the Blue and Red units are shown in Figure 6.17, respectively.



(a)



(b)

Figure 6.17 Comparison of remaining units when Blue is made stronger

In Figure 6.17, we now observe that the differences among the remaining platforms of the Blue bombers are not as significant as in the previous example. For the purpose of preserving more blue bombers, the team composition in this situation is less important than before. In addition, the no-teaming options 1 and 2 provides the blue commander with better results in the sense that more blue fighters are saved and more red defense units are destroyed, as can be seen in Figure 6.17.

6.4 NONINFERIOR NASH STRATEGIES

In this section, let us investigate the characteristics of NNS [45] in team composition and task assignment for the military air operation.

6.4.1 Problem formulation

The evolution of dynamic system is described as (5.34). The team composition and task assignment is formulated as a multi-team system. We assume that the Blue force and the Red force are divided into m_B and m_R sub-teams, respectively. For the j th Blue sub-team, there are N_j^{BB} BBs and N_j^{BF} BFs, which satisfy

$$\sum_{j=1}^{m_B} N_j^{BB} = N^{BB} \text{ and } \sum_{j=1}^{m_B} N_j^{BW} = N^{BW} . \tag{6.3}$$

For the j th Red sub-team, there are N_j^{RT} RTs , N_j^{RD} RDs and N_j^{FT} FTs, which satisfy

$$\sum_{j=1}^{m_R} N_j^{RT} = N^{RT}, \sum_{j=1}^{m_R} N_j^{RD} = N^{RD} \text{ and } \sum_{j=1}^{m_R} N_j^{FT} = N^{FT} . \tag{6.4}$$

Thus controllers u^B and u^R can be written in the following form:

$$u^B = \left(u_1^B, \cdots, u_{m_B}^B\right)' \text{ and } u^R = \left(u_1^R, \cdots, u_{m_R}^R\right)' \tag{6.5}$$

where the i^{th} Blue sub-team control u_i^B ($i = 1, \cdots, m_B$) and the j^{th} Red sub-team control u_j^R ($j = 1, \cdots, m_R$) are the vectors of appropriate dimensions.

The Blue force and the Red force have conflicting objective functions and cannot cooperate with each other. The objective function of each team at step k is given by

$$\begin{aligned} J^{B_i}(k) = &\sum_{j=1}^{N_i^{BB}} \alpha_{BB_j}^i \hat{p}_{BB_j}^i(k) + \sum_{j=1}^{N_i^{BF}} \alpha_{BF_j}^i \hat{p}_{BF_j}^i(k) \\ &- \sum_{j=1}^{N_i^{RT}} \alpha_{RT_j}^i \hat{p}_{RT_j}^i(k) - \sum_{j=1}^{N_i^{RD}} \alpha_{RD_j}^i \hat{p}_{RD_j}^i(k) - \sum_{j=1}^{N_i^{FT}} \alpha_{FT_j}^i \hat{p}_{FT_j}^i(k), \quad (i = 1, \cdots, m_B) \end{aligned} \tag{6.6a}$$

$$\begin{aligned} J^{R_i}(k) = &-\sum_{j=1}^{N_i^{BB}} \beta_{BB_j}^i \hat{p}_{BB_j}^i(k) - \sum_{j=1}^{N_i^{BF}} \beta_{BF_j}^i \hat{p}_{BF_j}^i(k) \\ &+ \sum_{j=1}^{N_i^{RT}} \beta_{RT_j}^i \hat{p}_{RT_j}^i(k) + \sum_{j=1}^{N_i^{RD}} \beta_{RD_j}^i \hat{p}_{RD_j}^i(k) + \sum_{j=1}^{N_i^{FT}} \beta_{FT_j}^i \hat{p}_{FT_j}^i(k), \quad (i = 1, \cdots, m_R) \end{aligned} \tag{6.6b}$$

In the above expressions, $\hat{p}_{X_j}^i$ is the normalized number of platforms for the j^{th} unit of X in i^{th} team, i.e.,

$$\hat{p}_{X_j}^i(k) = \frac{p_{X_j}^i(k)}{p_{X_j}^i(0)} \qquad k = 0,1,2,3....K\,. \tag{6.7}$$

Clearly, this newly model for military air operation can be formulated as an optimization problem of a two-team system. For example, in the Blue force, the i^{th} sub-team has its own objective function $J^{B_i}(k)$ to be maximized. Also, all the blue sub-teams are required to cooperate with each other to complete pre-assigned tasks. The overall problem can be formulated as:

$$\left(\max_{\left\{u_1^B(0),\cdots,u_1^B(K-1)\right\}} (\sum_{k=0}^{K} J^{B_i}(k)),\cdots, \max_{\left\{u_{m_B}^B(0),\cdots,u_{m_B}^B(K-1)\right\}} (\sum_{k=0}^{K} J^{B_{m_B}}(k)) \right) \tag{6.8a}$$

for the Blue sub-teams and

$$\left(\max_{\left\{u_1^R(0),\cdots,u_1^R(K-1)\right\}} (\sum_{k=0}^{K} J^{R_i}(k)),\cdots, \max_{\left\{u_{m_R}^R(0),\cdots,u_{m_R}^R(K-1)\right\}} (\sum_{k=0}^{K} J^{R_{m_R}}(k)) \right) \tag{6.8b}$$

for the Red sub-teams such that

$$z(k+1) = f(z(k), u^B(k), u^R(k), k)\,. \tag{6.9}$$

Note that the objective function $J^{B_i}(k)$, $i = 1,\cdots, m_B$,(or $J^{R_j}(k)$, $j = 1,\cdots, m_R$) is not only a function of the opposing force control u^R (or u^B), but also a function of the controls of other sub-teams in the same force. Therefore, we can apply Nash Noninferior Strategy to this multi-team dynamic system. The algorithm used to determine this NNS is given as follows:

Step 1. Consider maximizing the sub-team objective function $J_{k,k+2}^{B_i}$ (or $J_{k,k+2}^{R_i}$) over two-step time horizon:

$$J_{k,k+2}^{B_i} = J^{B_i}(k+1) + J^{B_i}(k+2)\,,\, i = 1,\cdots, m_B\,; \tag{6.10a}$$

$$J_{k,k+2}^{R_i} = J^{R_i}(k+1) + J^{R_i}(k+2)\,,\, i = 1,\cdots, m_R\,; \tag{6.10b}$$

Step 2. Construct the scalar objective criteria for each force under a given weight vector $\left(\xi^B, \xi^R\right)$ as:

$$J_{k,k+2}^{B,\xi^B}(u^B(k), u^B(k+1); u^R(k), u^R(k+1)) = \sum_{i=1}^{m_B} \xi_i^B J_{k,k+2}^{B_i} \tag{6.11a}$$

$$J_{k,k+2}^{R,\xi^R}(u^B(k), u^B(k+1); u^R(k), u^R(k+1)) = \sum_{i=1}^{m_R} \xi_i^R J_{k,k+2}^{R_i} \tag{6.11b}$$

where

$$\xi^X = \left(\xi_1^X, \cdots, \xi_{m_X}^X\right), \sum_{i=1}^{m_X} \xi_i^X = 1, \quad \xi_i^X \geq 0, \; i = 1, \cdots, m_X, \; X = B, R. \tag{6.12}$$

Step 3. Solve for Nash Noninferior Strategies $\left\{\left(\hat{u}^B(k), \hat{u}^B(k+1)\right), \left(\hat{u}^R(k), \hat{u}^R(k+1)\right)\right\}$ which satisfy that

$$J_{k,k+2}^{B,\xi^B}(\hat{u}^B(k), \hat{u}^B(k+1); \hat{u}^R(k), \hat{u}^R(k+1)) \geq J_{k,k+2}^{B,\xi^B}(u^B(k), u^B(k+1); \hat{u}^R(k), \hat{u}^R(k+1))$$

$$\forall \left\{u^B(k), u^B(k+1)\right\} \in U^B(k) \times U^B(k+1), \tag{6.13a}$$

$$J_{k,k+2}^{R,\xi^R}(\hat{u}^B(k), \hat{u}^B(k+1); \hat{u}^R(k), \hat{u}^R(k+1)) \geq J_{k,k+2}^{R,\xi^R}(\hat{u}^B(k), \hat{u}^B(k+1); u^R(k), u^R(k+1))$$

$$\forall \left\{u^R(k), u^R(k+1)\right\} \in U^R(k) \times U^R(k+1). \tag{6.13b}$$

where $U^X(k)$ and $U^X(k+1)$ are admissible control sets for the force X at step k and step $k+1$, respectively.

As we discussed before, for each given weight vector, there may exist a Nash Noninferior Strategy. We will use the following example to explain the effect of various choices of weight vectors by force commanders on the outcome of system engagement.

6.4.2 Illustrative example

We consider a scenario where Red fixed target FT1, a bridge, is strongly defended by three groups of fixed SAMs & Radars (RD1~RD3) and two groups of armored vehicles (RT1,RT2). Blue force includes two groups of blue bombers (BB1 and BB2) and two groups of blue fighters (BF1 and BF2). The initial deployment of units is shown in Table 6.15. Probabilities of kill for each pair of units are given in Table 6.16.

For simplicity, we consider only dividing the Blue force into two sub-teams, and keeping the Red force as one team. The team composition is given in Table 6.17. Each Blue sub-team decides to maximize the remaining platforms of its own team members. Furthermore, Team B1 wishes to minimize the remaining platforms of the red defenses and red troops while Team B2 hopes to minimize the remaining platforms of the red fixed target. For each sub-team, the weighting coefficients of team members are listed in Table 6.18.

Table 6.15 Initial situation and equipments of the units for the example

Unit	Type	Location	Platforms	Weapons	Max. Salvo
BB1	F4 bombers	(8,7)	8	4	1
BB2	F4 bombers	(8,7)	2	4	1
BF_1	F2-E fighters	(8,7)	8	4	1
BF_2	F2-E fighters	(8,7)	4	4	1
RT_1;RT_2	Armored vehicles	(5,5); (5,4)	50	3	1
RD_1	Fixed SAM & radar	(2,2)	7	18/7	6/7
RD_2	Fixed SAM & radar	(2,2)	7	18/7	6/7
RD_3	Fixed SAM & radar	(2,2)	7	18/7	6/7
FT	Bridge	(2,2)	10	N/A	N/A

Table 6.16 Probabilities of kill for the example

	BB1	BB2	BF1	BF2	RT1	RT2	RD1	RD2	RD3	FT1
BB1	0	0	0	0	0.6	0.6	0.6	0.5	0.4	0.3
BB2	0	0	0	0	0.6	0.6	0.6	0.5	0.4	0.3
BF1	0	0	0	0	0	0	0.8	0.7	0.7	0
BF2	0	0	0	0	0	0	0.8	0.7	0.6	0
RT1	0.2	0.2	0.1	0.1	0	0	0	0	0	0
RT2	0.2	0.2	0.1	0.1	0	0	0	0	0	0
RD1	0.7	0.7	0.3	0.3	0	0	0	0	0	0
RD2	0.5	0.5	0.3	0.2	0	0	0	0	0	0
RD3	0.5	0.5	0.2	0.2	0	0	0	0	0	0
FT1	0	0	0	0	0	0	0	0	0	0

Table 6.17 Team composition for the example

	Sub-Teams	Units	Objective Function
Blue	Team B1	BB1, BF1	$J_k^{B_1} = \alpha_{BB_1}\hat{p}_{BB_1} + \alpha_{BF_1}\hat{p}_{BF_1} - \alpha_{RT_1}\hat{p}_{RT_1} - \alpha_{RT_2}\hat{p}_{RT_2} - \alpha_{RD_1}\hat{p}_{RD_1} - \alpha_{RD_2}\hat{p}_{RD_2} - \alpha_{RD_3}\hat{p}_{RD_3}$
	Team B2	BB2, BF2	$J_k^{B_2} = \alpha_{BB_2}\hat{p}_{BB_2} + \alpha_{BF_2}\hat{p}_{BF_2} - \alpha_{FT_1}\hat{p}_{FT_1}$
Red	Team R	RT1~2,RD1~3, FT1	$J_k^{R} = -\sum_{i=1}^{2}(\beta_{BB_i}\hat{p}_{BB_i} + \beta_{BF_i}\hat{p}_{BF_i}) + \sum_{i=1}^{2}\beta_{RT_i}\hat{p}_{RT_i} + \sum_{i=1}^{3}\beta_{RD_i}\hat{p}_{RD_i} + \beta_{FT_1}\hat{p}_{FT_1}$

Table 6.18 Weighting coefficients of team members in the team objective function

	BB1	BB2	BF_1	BF_2	RT_1	RT_2	RD_1	RD_2	RD_3	FT
Blue α_{Xi}	0.8	0.8	0.5	0.5	0.1	0.1	0.2	0.2	0.2	1
Red β_{Xi}	0.7	0.7	0.4	0.3	0.1	0.1	0.7	0.5	0.5	1

In order to find the Nash noninferior strategies, we use the following scalar criterion for the blue side:

$$J_{k,k+2}^{B,\xi^B} = \xi_1^B J_{k,k+2}^{B_1} + \xi_2^B J_{k,k+2}^{B_2} \tag{6.14}$$

where $\xi_1^B + \xi_2^B = 1, \quad \xi_1^B, \xi_2^B \geq 0$. We select several options of values of (ξ_1^B, ξ_2^B), which are given in Table 6.19.

Table 6.19 Weighting coefficients in the scalar criterion of the Blue force

Options	Option 1	Option 2	Option 3	Option 4	Option 5
$[\xi_1^B, \xi_2^B]$	[0 , 1]	[0.25 , 0.75]	[0.5 , 0.5]	[0.75 , 0.25]	[1 , 0]

By using two-step looking-ahead dynamic programming method, we solve for the Nash Noninferior Strategies. Figure 6.18 shows the initial scenario of this example. Figure 6.19 gives the simulation results for different options of weighting coefficients listed in Table 6.20. In one extreme case such as option 1, the Blue force doesn't complete its task because the number of the remaining platforms of FT1 is still more than 60%. This is caused by the fact that, with the option 1, i.e $\xi_1^B = 0, \xi_2^B = 1$, Blue commander only cares to destroy the red fixed target and save units in Team B2 as many as possible. However, he ignores the red defenses and the casualty of units in Team B1. Therefore, BB1 will decide to attack the red target directly and BB2 will not enter the red area at all. Also, since Blue fighters are only most effective in destroying red defenses, thus BF1 and BF2 decide not to enter the engagement area either. Option 5, the other extreme case, is also unreasonable since the Blue force doesn't attack the red fixed target (FT) at all, though attacking FT is undoubtedly the most important goal for the Blue force. We called both option 1 and option 5 "blind choices" for the blue commander. It shouldn't occur in the real implementation.

In option 2, for the Blue objective function $J_{k,k+2}^{B,\xi^B}$ in (6.14), Team B1 has a smaller weighting coefficient than Team B2 (0.25<0.75). The Blue commander still pays less attention to the red defense parts than to the red fixed target. The number of BB1 is larger than that of BB2 and thus BB1 is more effective to attack fixed target than BB2. Therefore, Team B1 enters the red area first as shown in Figure 6.20. The fixed target is not attacked enough when BB1 is used up. The Red defense parts are still strong. BF2, at this moment, decides to participate in to weaken the red defense part. This can be seen in Figure 6.21. Then, BB2 at last accomplishes the mission, as shown in Figure 6.22. In this option, we see that the members in Team B1, especially BB1, suffer grievous losses.

In option 4, Blue commander focuses on attacking the red defense parts. Thus the weight assigned to the objective of Team B2 in the scalar criterion is greater than that assigned to the objective of Team B1. BFs in both teams decide to enter the area first to attack RDs, as shown in Figure 6.23. In Figure 6.24, we see that, after the RDs are destroyed to some degree, BB1 joins them to attack the fixed target until the task is completed. The final results are shown in Figure 6.25. BB2 never enters the area in this option since BB1 is more effective than BB2 to attack FT1 and BB1 is enough to finish this task when the defense parts are destroyed. In this option, more BBs are saved than that in option 2 while BFs lose a lot.

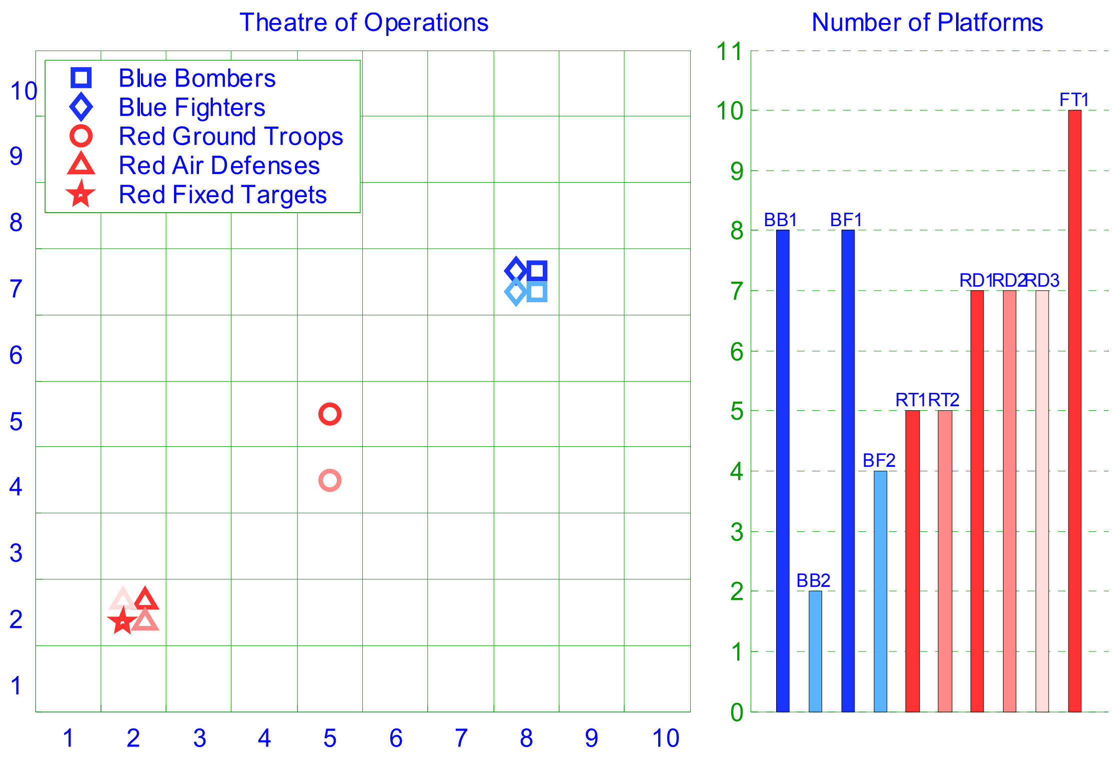


Figure 6.18 Initial situations for the example

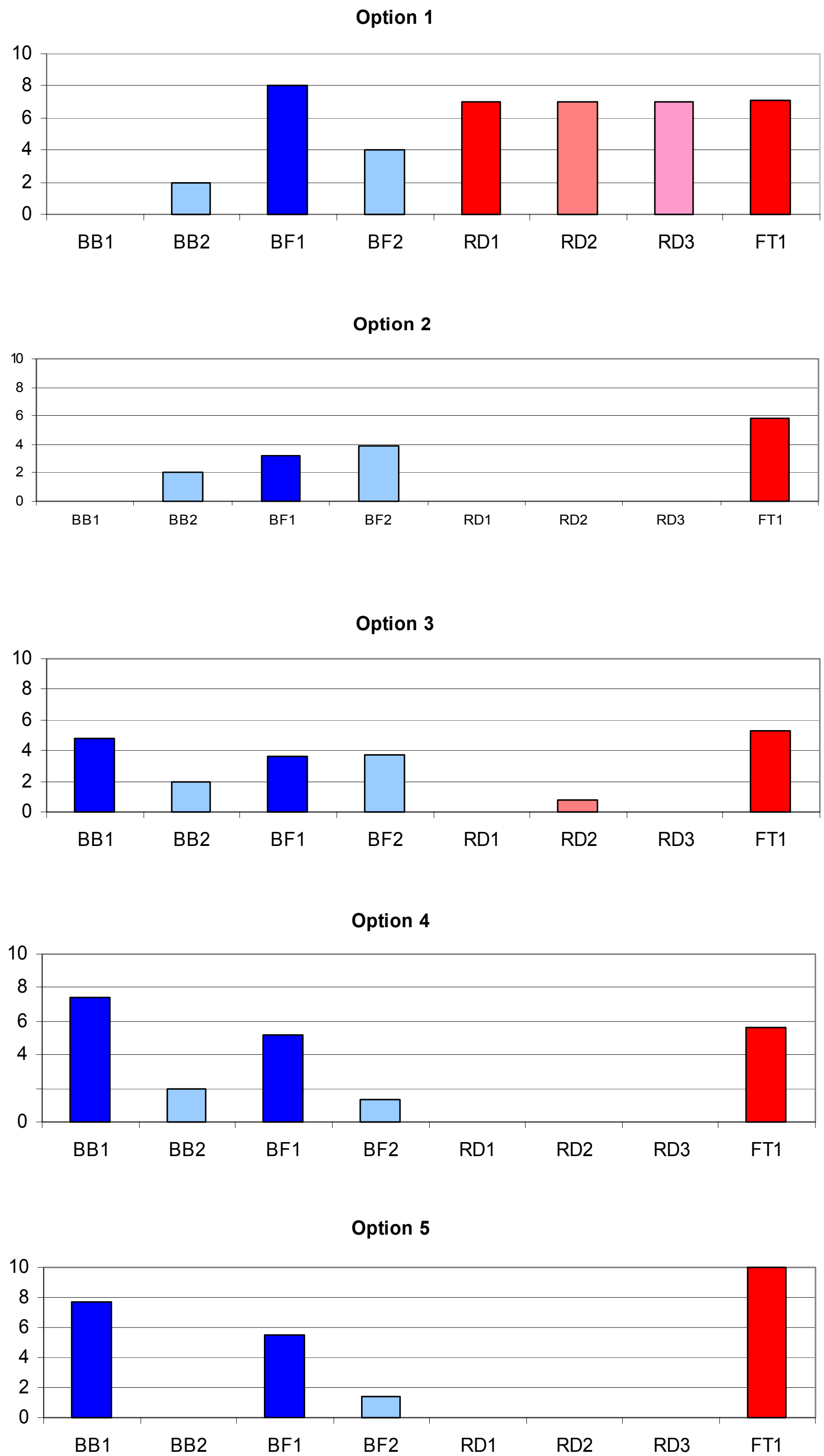


Figure 6.19 The number of remaining platforms for various options

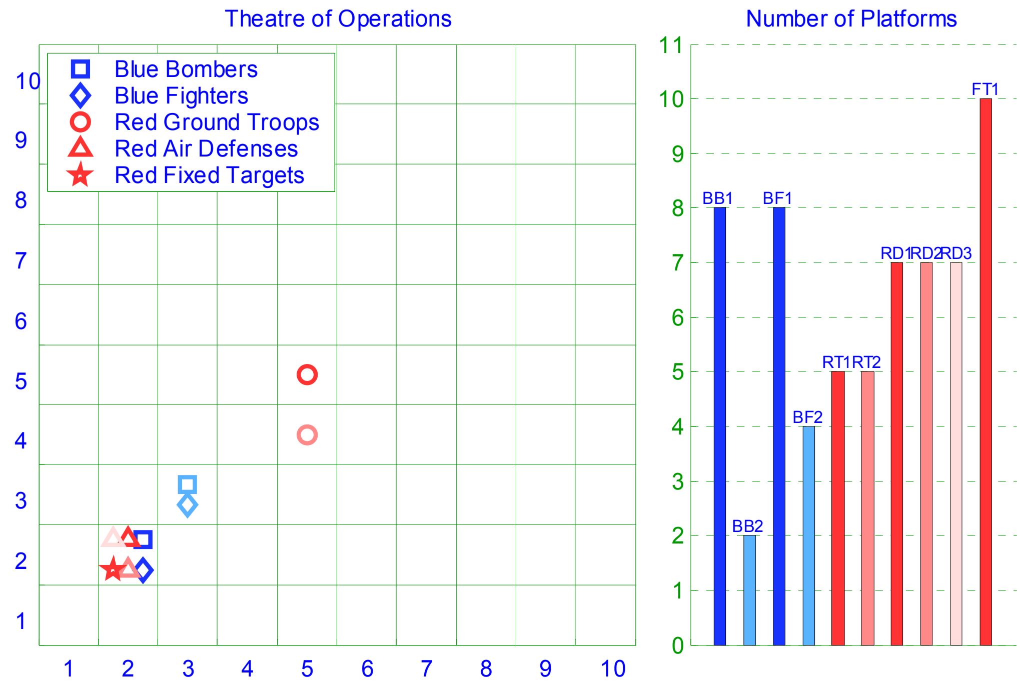


Figure 6.20 BB1 and BF1 enter the area first in option 2

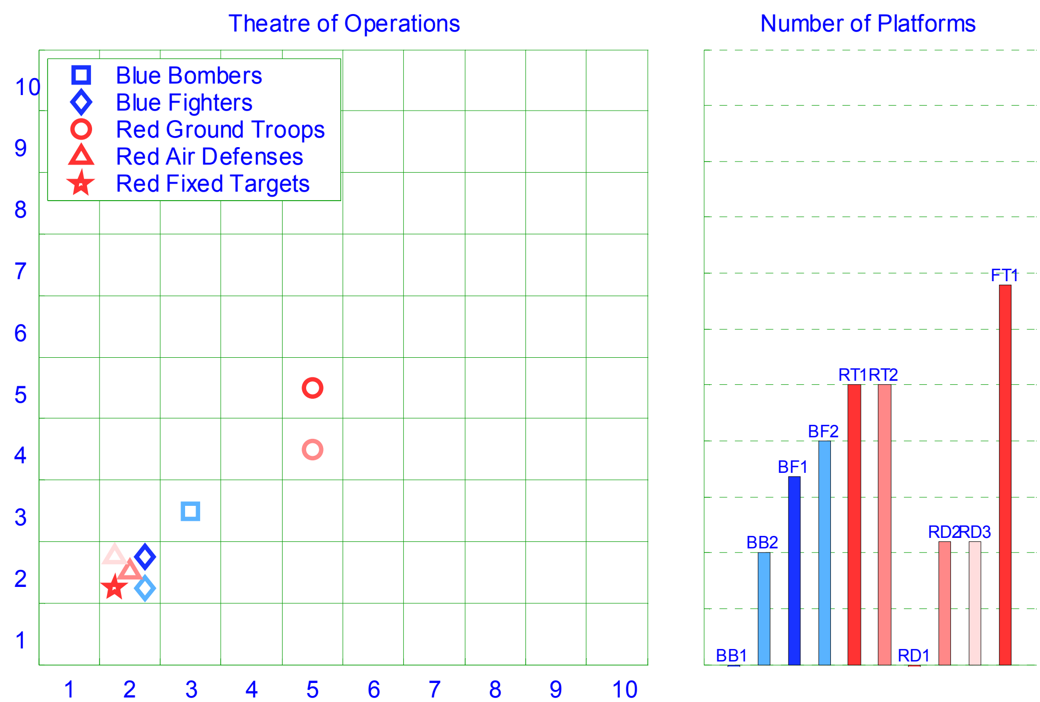


Figure 6.21 BF2 enters the area in option 2

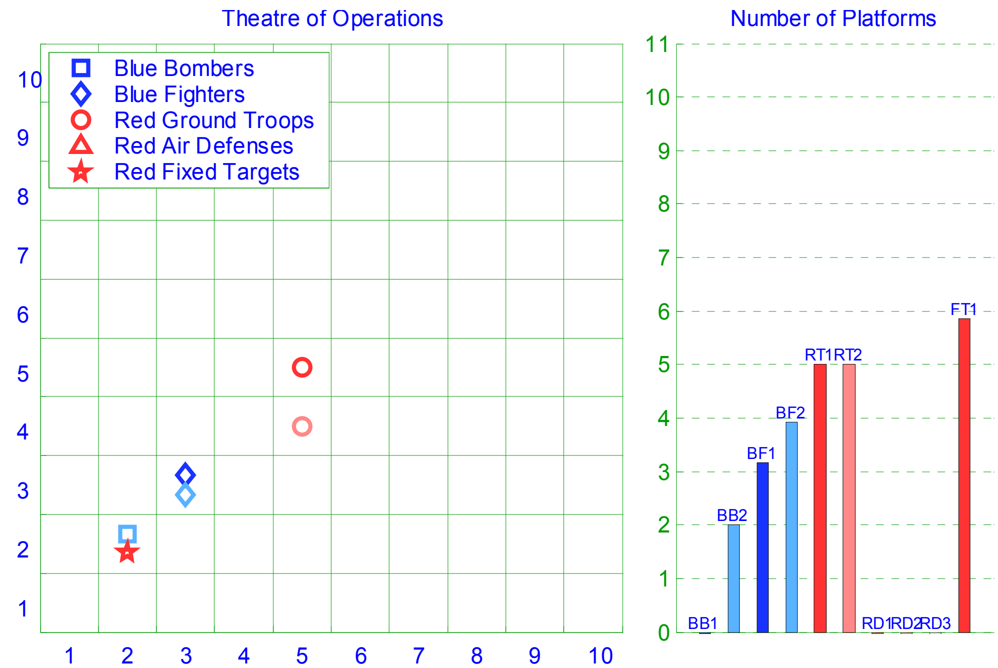


Figure 6.22 BB2 enters to complete the task in option 2

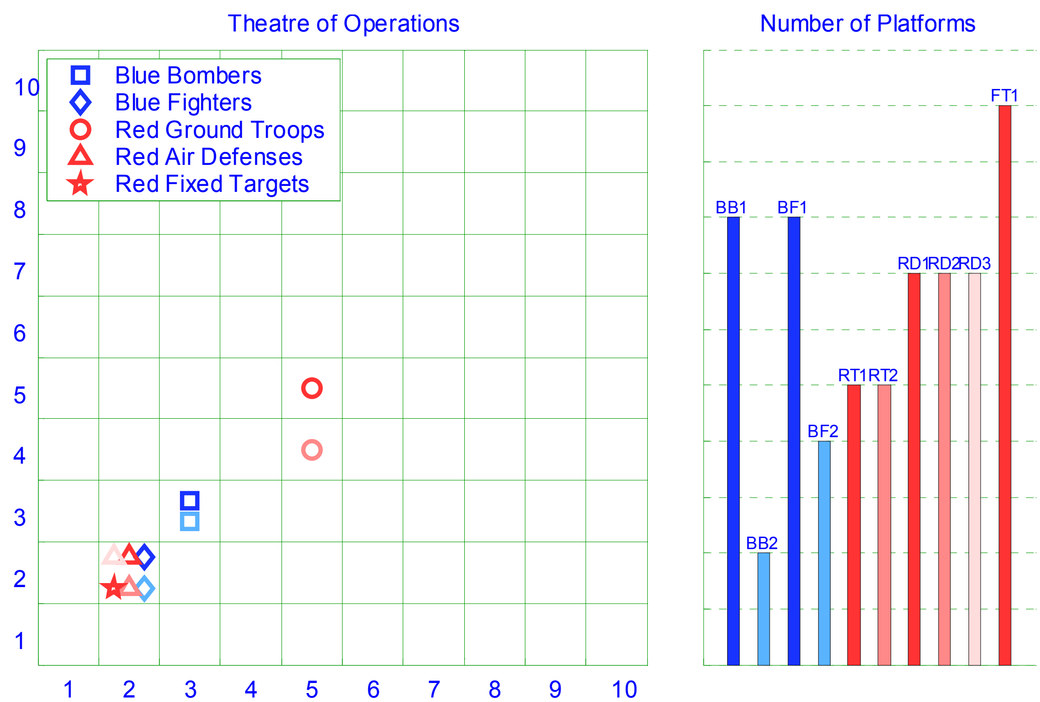


Figure 6.23 BFs enter first to attack the red defense parts in option 4

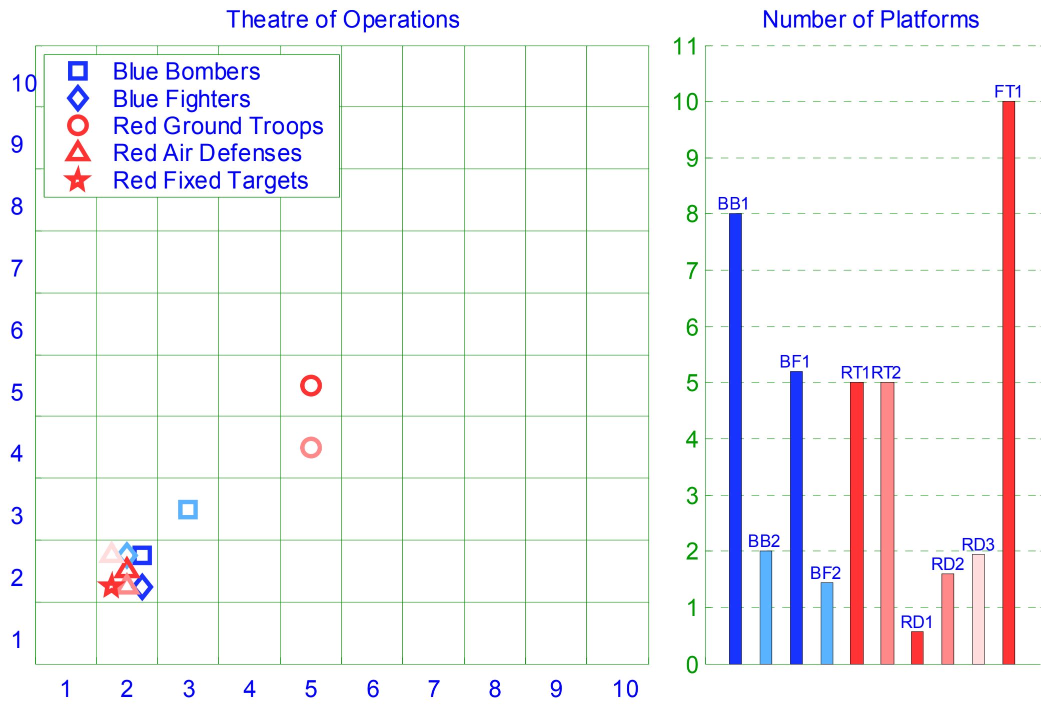


Figure 6.24 BB1 enters to attack FT in option 4

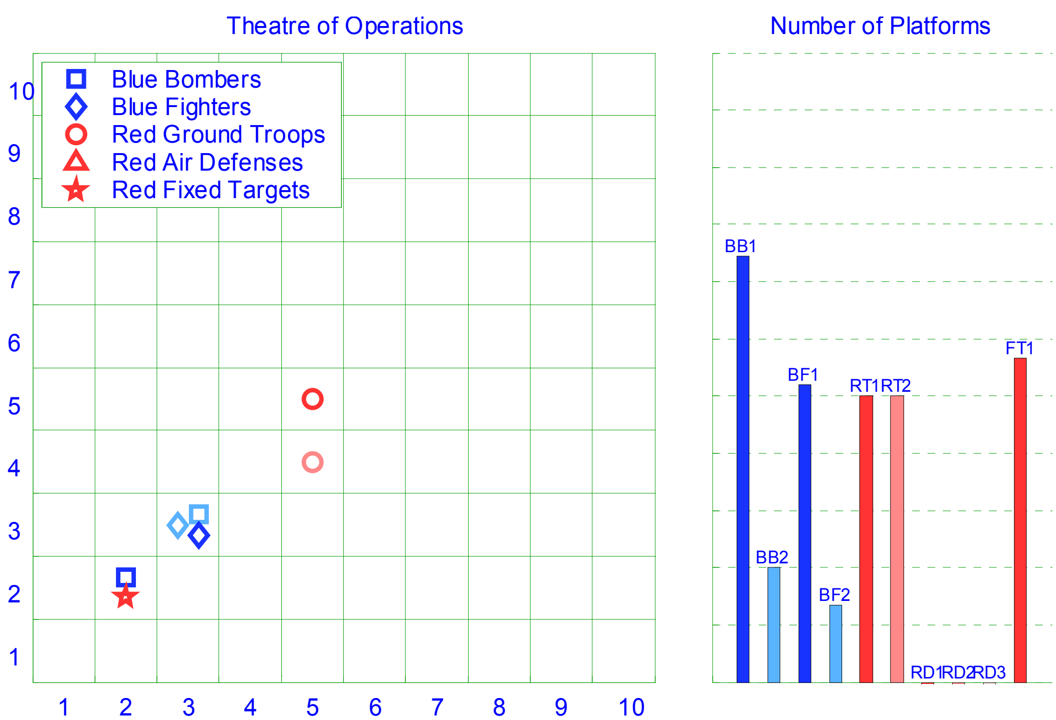


Figure 6.25 BB1 finishes the task without BB2 in option 4

Option 3, in which the weights for the objectives of both Team B1 and Team B2 are equal, is equivalent to that situation where there is no team composition at all. Compared to option 4, only BF1 first enters the area to attack the red defenses, as shown in Figure 6.26. Next, in Figure 6.27, we see that BB1 gets into position to attack the fixed target. We note that the red defense parts are still strong at this time. Therefore, BF2 also moves into the area to help to cripple the red defenses (in Figure 6.28). At last, in Figure 6.29, we see that the BB1 finishes the task and return to the base while there is still red defense alive.

We also compared the total remaining number of BBs' and BFs' platforms for the options 2, 3 and 4 in Figure 6.30 and Figure 6.31, respectively. For the purpose of saving more BBs, option 4 looks best. For the purpose of saving more BFs, option 3 looks better. As we can see in this example, several options have been provided to the Blue commander, from which he can choose one according to some fixed criterion or his own subjective desire.

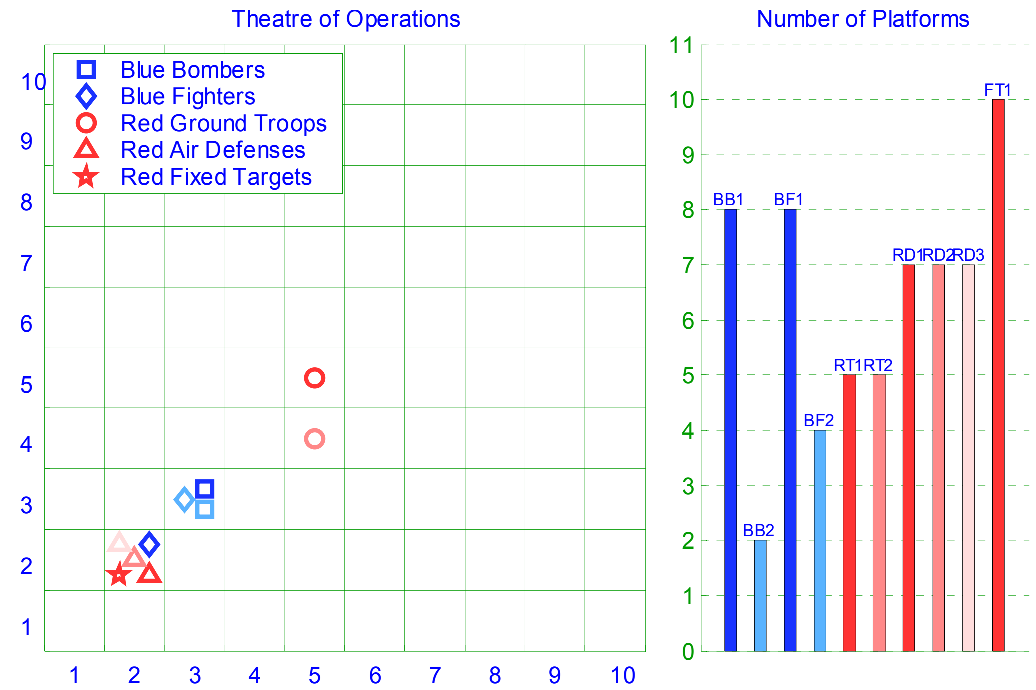


Figure 6.26 BF1 enters to attack RDs in option 3

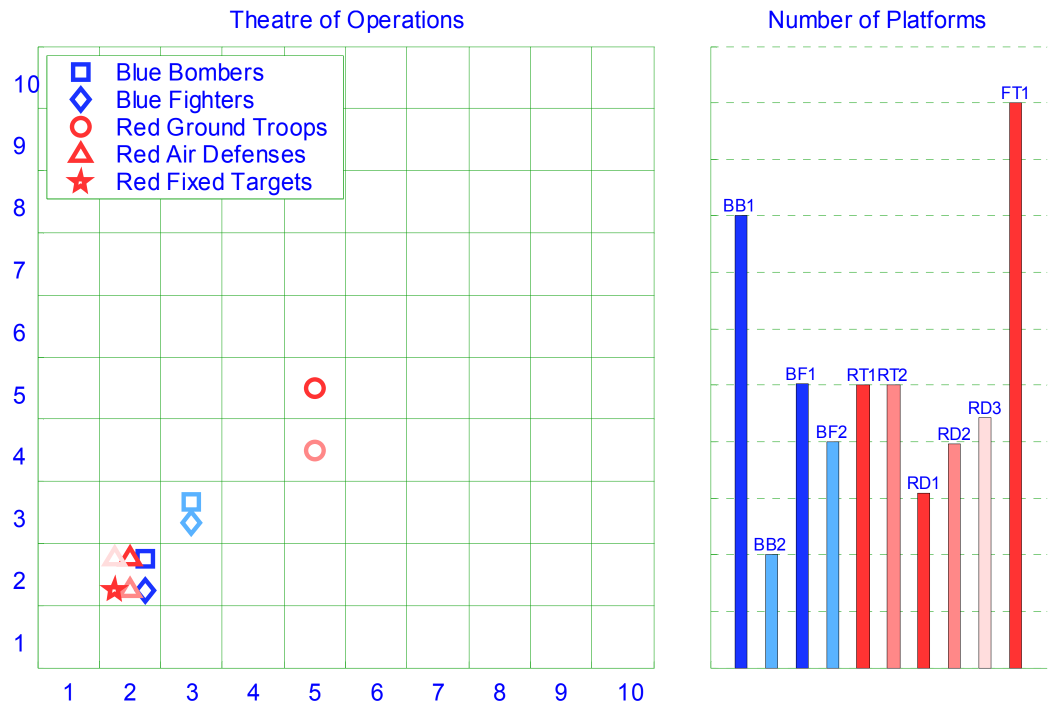


Figure 6.27 BB1 enters the area in option 3 while RDs are still strong

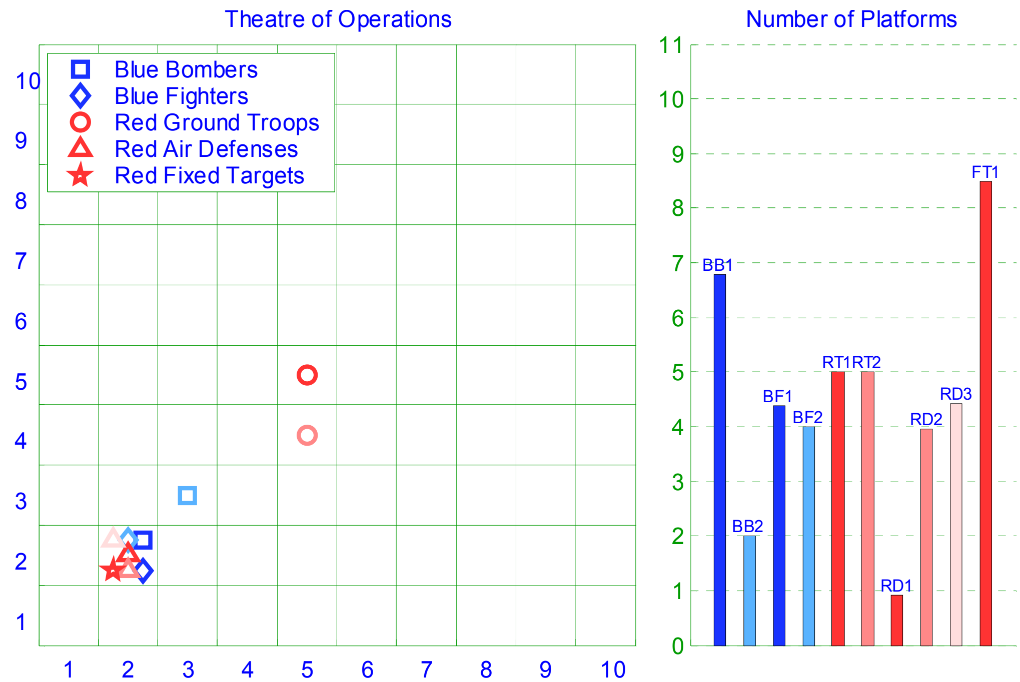


Figure 6.28 BF2 moves into the area to attack RDs in option 3

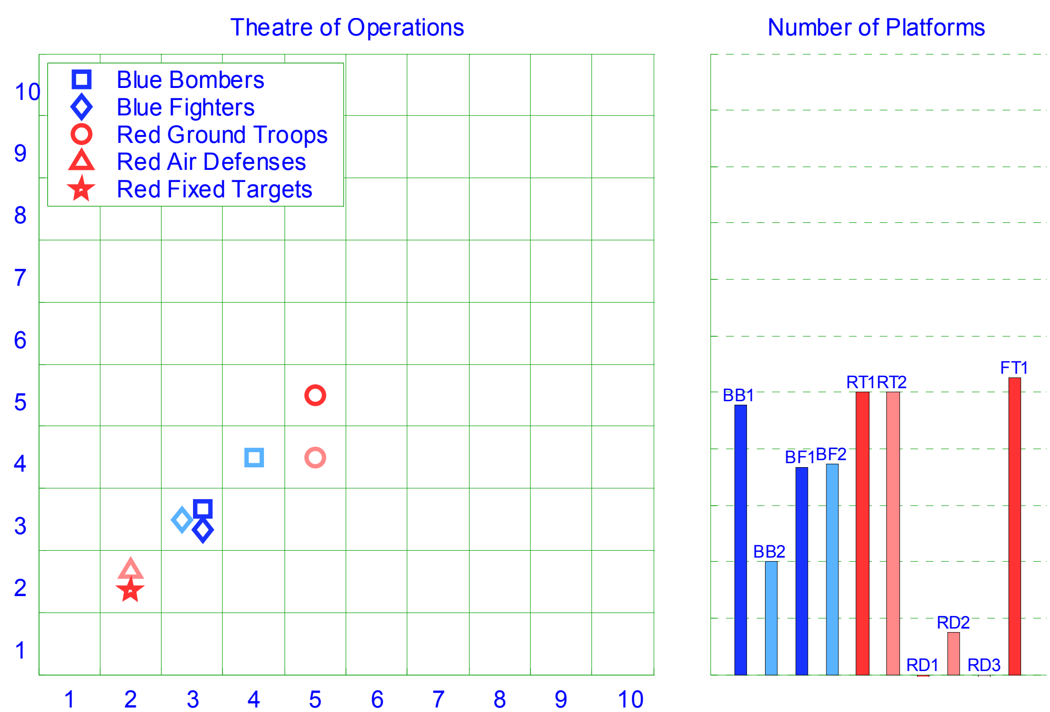


Figure 6.29 The task is completed in option 3

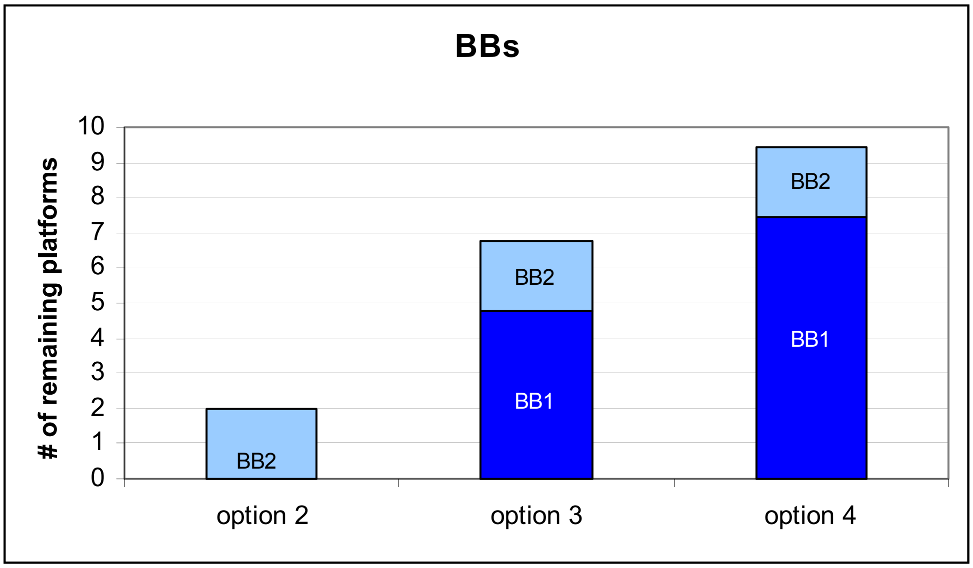


Figure 6.30 Comparison of the total number of BBs' remaining platforms in options 2-4

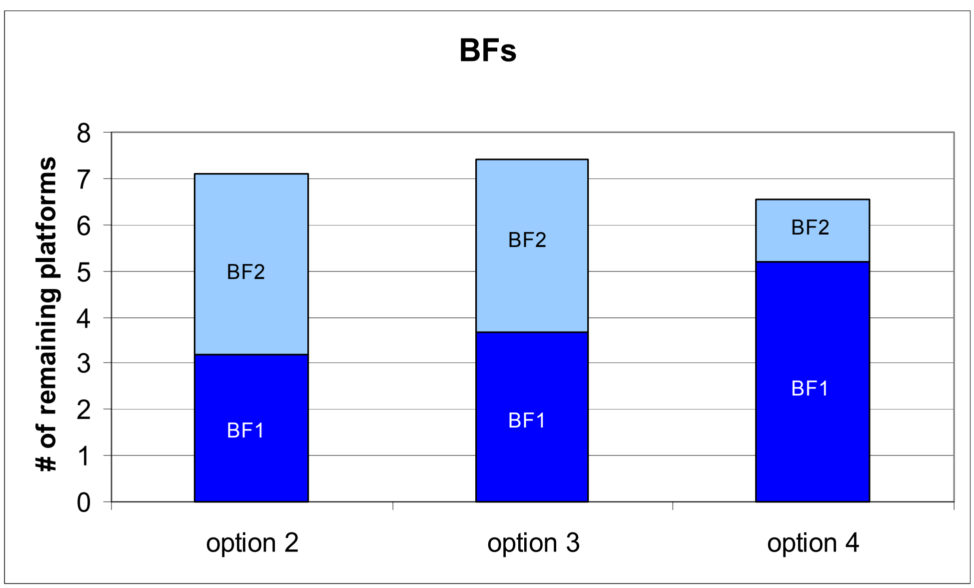


Figure 6.31 Comparison of the total number of BFs' remaining platforms in options 2-4

6.5 CONCLUSIONS

In this chapter, Nash reassignment strategies, Nash ordinal strategies and Noninferior Nash strategies are investigated as possible approaches to determine the optimal dynamic team composition and task assignment in the military air operation.

The reassignment problem in multi-team multi-task dynamic systems is specifically encountered by a commander to reassign some teams successfully accomplishing their tasks to assist other teams which perform their pre-assigned tasks either unsuccessfully or inefficiently. Our simulation examples demonstrated the Nash reassignment strategies can improve the overall performance of the Blue force. A Nash Ordinal strategy is presented for the top commander of each force to make decision on the initial task assignment and team composition. We have shown that Nash ordinal strategies are effective and useful in the decision of the initial resource allocation by the top leaders especially when mathematical expressions for their objectives are not available. We also discussed the effects of cooperative teaming with different set-ups by varying the resource capabilities available to each team. Our simulation results have shown that when one side has limited resources and strength to complete its mission, cooperative teaming among its constituents can improve the overall system's performance. Cooperative teaming in that case would be a wise choice for the leader of that side. However, as that side is made stronger, the difference in outcome between teaming and non-teaming becomes less and less noticeable. In that case, teaming may not be as necessary, and in fact may result in deterioration in performance. In the end, two-step look-ahead Noninferior Nash strategies are presented to investigate the effects of various options of weight vectors by force commanders for team task assignment on the cooperative performance among teams in one force.

7.0 CONCLUSIONS

In this dissertation, we developed a new game theoretic strategy, called Noninferior Nash strategy for an extended complex system consisting of several teams of cooperating decision-makers that are simultaneously in conflict with other adversarial teams controlling the same system. We investigated the properties of the Noninferior Nash strategy in both finite and infinite static games, and presented conditions for its existence in continuous time static games. We also obtained the conditions for existence of this strategy and its analytical solutions for a class of linear quadratic multi-team static games and dynamic games. This strategy has the property that there is no incentive for any one team in the system to deviate unilaterally while at the same time maintaining complete cooperation among team members. The Noninferior Nash strategies are considered as a mechanism for strengthening team cooperation in the presence of an adversary and thus improving the overall performance of the system.

In order to deal with the issue of non-uniqueness of the solution, we introduced the concept of the Noninferior Nash strategy with a team Leader (NNSL). This strategy is an extension of the Noninferior Nash Strategy, and allows for the selection of a particular solution from the set of solutions if each team has a Leader that optimizes a team objective function that may be different from those of the team members. In the general case, obtaining this solution may also involve a game among the team Leaders. Two examples of microeconomics problems and routing problems in parallel-link network are presented to illustrate the effectiveness of NNSL in improving the overall system performance.

A military air operation consisting of two sets of opposing forces is a typical example of an extended complex system. In this thesis, we introduced a dynamic model of the military air operation and investigated various Nash-based strategies for optimal planning of shared responsibilities and roles in the hierarchical deployment of units in the combat. Nash reassignment strategies (NRS) are applied in the situation when a team is not able to accomplish its task or when it can accomplish it in an inefficient manner. The top commander (or system leader in general) may decide to reassign another team to reinforce that team's ability to achieve its objective. Our simulation results showed that it is possible to reallocate resources dynamically and optimally, and thus improve system's performance using reassignment strategies. We also discussed the effects of cooperative teaming with different set-ups for the capabilities of one of the forces. Our experiments have shown that when one side has limited resources to complete its mission, cooperative teaming among its constituents can improve the overall system's performance. We applied the Noninferior Nash strategies (NNS) to determine the cooperative control for the teams on one side by varying the weighting coefficients related to the importance of these teams' strategic objectives.

BIBLIOGAPHY

[1] Nicolas R. Jennings and Stefan Bussmann, "Agent-Based Control Systems," IEEE Control Systems Magazine, Vol.23, No.3, 2003, pp.61-73.

[2] Aca Gacic, Multi-Agent Control of an Enterprise with an Adversary, M.S. Thesis, University of Pittsburgh, 2000.

[3] John Von Neumann and Oskar Morgenstern, Theory of Games and Economic Behavior, 1st Edition; Princeton, New Jersey: Princeton University Press, 1944.

[4] J. F. Nash, "Equilibrium Points in n-person Games," Proceedings of the National Academy of Sciences, U.S.A., 36, 1950, pp.48-49.

[5] Cournot, Recherches sur les Principes Mathematiques de la Theorie des Richesses. English edition (ed. N. Bacon): Researches into the Mathematical Principles of the Theory of Wealth (Macmillan, 1897).

[6] R. Isaacs, Differential Games, John Wiley and Sons, New York, 1965.

[7] W. Starr and Y. C. Ho, "Nonzero-Sum Differential Games," Journal of Optimization Theory and Applications, Vol.3, No.3, 1969, pp.184-206.

[8] W. Starr and Y. C. Ho, "Further Properties of Nonzero-Sum Differential Games," Journal of Optimization Theory and Applications, Vol.3, No.4, 1969, pp.207-219.

[9] M. A. Simaan and J. B. Cruz, Jr., "On the Stackelberg Strategy in Nonzero-Sum Games," Journal of Optimization Theory and Applications, Vol.11, No.5, 1973, pp.535-555.

[10] N. O. Dacunha and E. Polak, "Constrained Minimization Under Vector-Valued Criteria in Finite Dimensional Spaces," Journal of Mathematical Analysis and Applications, Vol.19, 1967, pp.103-124.

[11] Klinger, "Vector-Valued Performance Criteria," IEEE Transactions on Automatic Control, Vol.AC-9, No.1, 1964.

[12] T. Basar and G. J. Olsder, Dynamic Noncooperative Game Theory, Academic Press, , San Diego, CA, 1981.

[13] Vira Chankong and Yacov Y. Haimes, Multiobjective Decision Making: Theory and Methodology, North-Holland, 1983.

[14] P. L. Yu, "A Class of Solutions of group decision problems," Management Sciences, Vol.19, 1974, pp.936-946.

[15] P. L. Yu, "Cone Convexity, Cone Extreme Points, and Nondominated Solutions in Decision Problems with Multiobjectives," Journal of Optimization Theory and Applications, Vol.49, 1974, pp.430-468.

[16] Frank L. Lewis and Vassilis L. Syrmos, "Optimal Control," John Wiley & Son, Inc., 1995.

[17] Marwan A. Simman, Multi-Controller Strategies for Systems with Discrete Acquired and Biased Prior Information, Ph.D dissertation, University of Illinois at Urbana-Champaign, 1972.

[18] M. A. Simaan and J. B. Cruz, Jr., "A Stackelberg Solution for Games with Many Players," IEEE Transactions on Automatic Control, Vol. AC-18, No.3, 1973, pp.322-324.

[19] R. Radner, "Team Decision Problem," Annals of Mathematical Statistics, Vol.33, No.3, 1962, pp.857-881.

[20] Y. C. Ho and F. K. Sun, "Value of Information in Two-Team Zero-Sum Problems, Multicriteria Decision Making and Differential Games," Edited by G. Leitmann, Plenum Press, 1976, pp.305-319.

[21] Haurie, "On Pareto Optimal Decisions for a Coalition of a Subset of Players," IEEE Transactions On Automatic Control, Vol.AC-18, No.2, 1973, pp.144-149.

[22] W. E. Schmitendorf and G. Moriarty, "A Sufficiency Condition for Coalitive Pareto-Optimal Solutions," Multicriteria Decision Making and Differential Games, Edited by G. Leitmann, Plenum Press, 1976, pp.163-172.

[23] J. B. Rosen, "Existence and Uniqueness of Equilibrium Points for Concave N-Person Games," Econometrica, Vol.33, No.3, 1965, pp.520-534.

[24] G. Gaudet and S. W. Salant, "Uniqueness of Cournot Equilibrium: New Results from Old Methods," Reviews of Economic Studies, Vol.58, 1991, pp.399-404.

[25] S. W. Salant and G. Shaffer, "Unequal Treatment of Identical Agents in Cournot Equilibrium," The American Economic Review, Vol.89, No.3, 1999, pp.585-604.

[26] S. W. Salant and G. Shaffer, "Optimal Asymmetric Strategies in Research Joint Ventures," International Journal of Industrial Organization, Vol.16, 1998, pp.195-208.

[27] Eitan Altman, Tamer Basar, Tania Jimenez and Nahum Shimkin, "Competitive Routing in Networks with Polynomial Costs," IEEE Transactions on Automatic Control, Vol.47, No.1, Jan. 2002, pp.92-96.

[28] Eitan Altman, Tamer Basar and R. Srikant, "Nash Equilibria for Combined Flow Control and Routing in Networks: Asymptotic Behavior for a Large Number of Users," IEEE Transactions on Automatic Control, Vol.47, No.6, June. 2002, pp.917-930.

[29] Ismet Sahin and Marwan A. Simaan, "Routing and Flow Control for Parallel Links Communication Networks with Multiple Competing Users", submitted to American Control Conference 2004.

[30] Katsunori Yamaoka, Shinji Sugawara and Yoshinori Sakai, "Connection Oriented Packet Communication Control Method Based on Game Theory," IEEE International Conference on Communications, Vol.2, Vancouver, BC, Canada, 1999, pp.1346-1351.

[31] A. Economides and J. A. Silvester, "A Game Theory Approach to Cooperative and Non-Cooperative Routing Problems," IEEE International Telecommunication Symposium, 1990, pp.597-601.

[32] Richard J. La and Venkat Anantharam, "Optimal Routing Control: Repeated Game Approach," IEEE Transactions on Automatic Control, Vol.47, No.3, March, 2002, pp.437-450.

[33] Yannis A. Korilis, Aurel A. Lazar and Ariel Orda, "Achieving Network Optima Using Stackelberg Routing Strategies," IEEE/ACM Transactions on Networking, Vol.5, No.1,1997, pp.161-173.

[34] T. Basar and R. Srikant, "A Stackelberg Network Game with a Large Number of Followers," Journal of Optimization Theory and Applications, Vol.115, No.3, December 2002, pp.479-490.

[35] Thomas Boulogne, Eitan Altman, Hisao Kameda and Odile Pourtallier, "Mixed Equilibrium (ME) for Multiclass Routing Games," IEEE Transactions on Automatic Control, Vol.47, No.6, 2002, pp.903-916.

[36] John N. Daigle, Queueing Theory for Telecommunications, Addison-Wesley Publishing Company, Inc., 1992.

[37] J. B. Cruz, Jr., M. A. Simaan, A. Gacic, H. Jiang, B. Letellier, M. Li, and Y. Liu, "Game-Theoretic Modeling and Control of Military Operations," IEEE Transactions on Aerospace and Electronic Systems, Vol. 37, No. 4, 2001,pp. 1393-1405.

[38] R. L. Helmbold, "A Modification of Lanchester's Equation," Operations Research, Vol. 13, 1065, pp.857-859.

[39] J. S. Przemieniecki, Mathematical Methods in Defense Analysis, AIAA Education Series, 3rd Edition, 2000.

[40] J. B. Cruz, Jr., M. A. Simaan, A. Gacic and Y. Liu, "Moving Horizon Game Theoretic Approaches for Control Strategies in a Military Operation," IEEE Transactions on Aerospace and Electronic Systems, Vol.38, No.3, 2002, pp. 989-999.

[41] Y. Liu, M. A. Simaan and J. B. Cruz, Jr., "An Application of Dynamic Nash Task Assignment Strategies to Multi-Team Military Air Operations," Automatica, Vol.39, No.8, August 2003, pp.1469-1478.

[42] J. B. Cruz, Jr. and M. A. Simaan, "Ordinal Games and Generalized Nash and Stackelberg Solutions," Journal of Optimization Theory and Applications, Vol.107, No.2, November, 2000, pp.205-222.

[43] Yong Liu, David Galati and Marwan A. Simaan, "Task Assignment in Multi-Team Cooperative Systems," Proceedings of the International Conference on Computer, Communication and Control Technologies, Orlando, Florida, July 31, August 1-2, 2003.

[44] Yong Liu, Marwan A. Simaan, and Jose B. Cruz, Jr., "Game Theoretic Approach to Cooperative Teaming and Tasking in the Presence of an Adversary," American Control Conference, June 4-6, Denver, Colorado, USA, 2003.

[45] Y. Liu and M. A. Simaan, "Noninferior Nash Strategies for Multi-Team Systems," Journal of Optimization Theory and Applications, Vol.120, No.1, January, 2004.