Spring 1-1-2015

Agent Objective Function Design in Distributed Engineering Systems

Matther J. Phillips
University of Colorado at Boulder, matthew.phillips-1@colorado.edu

Agent Objective Function Design in Distributed Engineering Systems

by

Matthew J. Phillips

B.S., Harvey Mudd College, 2011

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Master of Science

Department of Electrical, Computer, and Energy Engineering

2015

This thesis entitled:
Agent Objective Function Design in Distributed Engineering Systems
written by Matthew J. Phillips
has been approved for the Department of Electrical, Computer, and Energy Engineering

Dr. Jason Marden

Dr. Berouz Touri

Dr. Eric Keller

Date

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

Phillips, Matthew J. (M.S.)

Agent Objective Function Design in Distributed Engineering Systems

Thesis directed by Dr. Jason Marden

A distributed engineering system requires the design of agent objective functions to control the system sub-entities. These agent functions can cause non-optimal and inefficient system dynamics to occur, and the quality of the design can influence the degree of these undesirable responses. The creation of these agent objective functions is investigated, and an existing method of bounding the induced inefficiencies is expanded to encompass a wider range of games by considering a *budget condition*. Following, a novel method of designing player objective functions via a convex combination of existing functions is introduced and fully characterized. The significance of these contributions are demonstrated in two theoretic models, and the shortcomings of the expansion are discussed. A final contribution demonstrates that one agent objective function is optimal for some models.

Dedication

"Adventure is Curiosity; the willingness to embrace uncertainty . . . wondering about the possibility of doing just one thing differently than before." –Bruce Kirkby

To you.

Acknowledgements

First and foremost, I would like to thank my advisor, Dr. Jason Marden. I am sincerely appreciative of the time that he spent working with me, the feedback and insights he provided me, and the patience that he gave me, all of which are paramount to where I am today.

Along with my advisor, I wish to thank those that served on my defense committee: Dr. Eric Keller and Dr. Behrouz Touri. Dr. Keller was one of the first people I met when I visited CU for graduate school, and he and others demonstrated the passion for research and bonhomie that led me to selecting CU over others. Dr. Touri has provided me with excellent guidance and advice on aspects unrelated to school, and I will always be appreciative of his mentorship.

I would also like to thank Dr. Shannon Stott. She has been an inspiration to me in my pursuit to be the best that I can, and has positively influenced my own counseling of others.

Additionally, I thank Dr. Li Shang for initially bringing me on as a student, and for leading me in my first year of graduate school. His ideas are ever rich, and his spirit is always contagious!

I also thank my lab mates! Holly, Philip, Andi, Yassmin, Raga, Yilan, Dalton, Vinod, Saeid, and Tatiana, you were all monumental in commending my good research ideas, and suggesting alternatives to my poor ones.

Lastly, I thank my family and friends (including those above). Each of you are so important and valuable, and my searching and researching for happiness would be incomplete without you.

Contents

Appendix

Tables

Table

Figures

Figure

Chapter 1

Introduction

1.1 Overview

It is often desired to optimize some form of a global objective in engineering systems. Many of these systems are comprised of numerous sub-entities. Consider, for example, a search system that is composed of a collection of multiple unmanned aerial vehicles (UAVs) searching over a wilderness area for a lost hiker. Ideally, a control algorithm could be implemented with full information about the search area, the location of each UAV, and the area that remains to be investigated. However, factors such as communication limitations, computation constraints, search space size, system complexity, etc., may limit the feasibility to implement a centrally designed control algorithm. Despite these limitations, one way to realize the global objective of finding the hiker is to create individual control algorithms for each UAV so that, given the information available to them, the UAV is controlled by this algorithm to perform in a manner that is globally desired. These individual control algorithms are comprised of an *agent objective function* coupled with a learning algorithm. Hence, the overall goal is to control these multi-agent systems by designing optimal agent objective functions and optimal learning algorithms so that, even though each sub-entity is being controlled independently, collectively they are able to accomplish the global objective. The agent objective functions are designed with the knowledge that each agent has a set of available actions that they may take (*i.e.* turn left/right, go straight). Then the individual UAV agents are programmed to select the action that optimizes their agent function. Thus, two UAVs searching over the same space might have respective agent objective functions that are optimized when the the UAVs are instead searching for the hiker in separate locations (so as to cover more area). In recognizing this, the UAVs would be controlled by their respective agent objective functions to separate.

Historically, these agent objective functions, also known as *player objective functions*, *utility functions*, *cost functions*, or simply *objective functions*, have been approached from a social sciences perspective, where research has focused largely on how to model individual players. However, when thinking about distributed engineering systems, these objective functions are actually *designed* for the agents that utilize them. And whereas social systems are inherently distributed, the underlying structure of engineering systems requires them to become increasingly distributed as they grow in complexity; that is, as some systems grow in size, the amount of total information may be too large for efficient computations to be possible, or the total information available may be limited to each sub-entity, or the space in which the sub-entities are operating might have inherent environmental communication limitations among the entities.

It is the goal of the designer of a distributed engineering system to create these agent objective functions so that the sub-entities operate in a way desired from a global perspective. Problems arise when player objective functions force agents to make decisions in a way not optimal from the global perspective. A designer can quantify and assess agent objective functions by comparing the globally desired outcome to the global outcome that results from the induced dynamics of the agent functions – in particular, the global outcomes for when every agent has locally optimized their own objective function, and each agent cannot further improve their result by unilaterally modifying their action. These "equilibria" are often the focus in player objective design, as there are many dynamics that can reach them in engineering systems.

This thesis focuses on how to assess and quantify agent objective functions in distributed and multi-agent systems, and discusses novel approaches in player objective function design. Although we center on engineering systems, many of the results presented can also be applied to social sciences settings.

1.2 Motivation

These engineering systems can be modeled as a game, where informally, a game is a set of players, each with an available set of decisions or actions that they may select. These agents make choices based on their individual objective functions which maps their decisions to a real number. With their individual objective functions as the control, the agents make decisions that will maximize or minimize their function (depending on the system design). In social science settings, these individual objective functions may be unknown to an outsider, and therefore may require modeling. Yet for engineering systems, these objective functions may be specified by a designer. A centralized engineering system is one where the sub-entities are controlled from a central location via a single control algorithm, and the optimal control is centrally solved. In contrast, distributed systems are ones where the sub-entities are each controlled separately via their own control algorithm. The UAV example system from Section 1.1 is one of many engineering systems that can be viewed in a game theoretic setting where the player objective functions are designed. Network routing provides another example, where it is often globally desired to minimize power consumption or maximize transmission rates [12]. The quality of agent objective functions is important as it directly influences the behavior that can result in a system [10, 12].

1.3 Thesis Outline

The structure of this thesis is as follows: in Chapter 2 we will provide thorough background information on the thesis topic; we will introduce the model of interest, as well as the metrics that will be used to assess designed agent objective functions. In Chapter 3, we will expand the bounds of worst-case inefficiencies that may result in an engineering system for a wider class of games called "smooth" games, and will present (existing) bounds on the best-case inefficiencies that may occur in a certain game model called "potential games." In Chapter 4, we focus on a specific type of network game, called "reverse carpooling," that has been thoroughly characterized in [12], and use it to demonstrate the practicality of the contributions from Chapter 3. Following, in Chapter 5 we discuss the novel method of creating new agent objective functions by convexly combining existing ones, and bound the resulting inefficiencies from these new designs. In Chapter 6, we demonstrate the utility of the results of the previous chapters on a model called a "resource allocation game." In this chapter, we also demonstrate some shortcomings of the previous contributions, and prove that a specific agent objective function is optimal. We conclude with a discussion on the significance of the thesis contributions and explore future research directions (Chapter 7).

1.4 Thesis Contributions

The major contributions from this thesis are the following:

- Existing worst-case system inefficiency bounds for a specific game, known as "smooth games" from [20], is expanded to a wider set of games. The expansion is demonstrated to parallel a similar bound that exists for the best-case system inefficiencies for games known as "potential games." This contribution extends upon prior work presented in [**?**], and applies for all budget conditions and all equilibria types.

- A bound on the worst-case system inefficiencies for a "tunable" player objective function for the reverse carpooling games in [12] is proven and is used to demonstrate the usefulness of the previous contribution.

- A novel design method of convexly combining existing agent objective functions is introduced, and bounds on the worst-case system inefficiencies of the resulting design are proven.

- A specific game model is introduced, and the significance of the previous contributions are demonstrated in agent objective function designs for this game. The results are also used to demonstrate limitations of the previous contributions. Additionally, an optimal agent cost function is proven for these games.

- Additional insights on agent objective function design is provided throughout, and remaining unanswered research questions are discussed.

This concludes the introductory chapter. In the next chapter we will present the background information relevant to the rest of this thesis.

Chapter 2

Background

2.1 Outline

In this chapter, we provide thorough background information pertinent to the contents of this thesis. Following a motivating example that demonstrates the inherent difficulties in agent objective function design, we formally define cost minimization games and introduce the types of equilibria considered in this thesis.

2.2 Example

We consider the following motivating example to illustrate the hardships that designers face when making agent objective functions:

The Model:

- Consider two robots, 1 and 2, hiding from an enemy. Each robot only has the ability to see nearby robots, but cannot communicate with them.

- A set of locations $R = \{1, 2, \ldots, m\}$,[1] where each location, r, has an associated value, $\beta_r \in [0, 1]$, representing the independent probability of being found by the enemy in that

[1] Each location can be thought of as a resource; this is why we use R to represent this set.

location.

- Each robot (or agent), 1 and 2, has a set of locations, $A_1 \subseteq R$ and $A_2 \subseteq R$, respectively, that they may hide in. They may hide in only one location (single selection).

  * We define action profiles as $a = \{a_1, a_2\} \in A$, with $a_1 \in A_1$ and $A = A_1 \times A_2$.

  * We require that $\sum_{r \in R} \beta_r = 1$.

- The global objective is to not be detected by the enemy; that is, to not have *any* of the robots be detected by the enemy.[1]

To realize the global objective, a designer would seek to minimize the total probability of any agent being found by the enemy. This can be formulated by attempting to minimize the summed values of the selected resources. We can express this using a global cost function, $C : A \to \mathcal{R}$, where

$$C(a) = \sum_{r \in \cup_{i \in N} a_i} \beta_r. \tag{2.1}$$

Alternatively, we could express this as

$$C(a) = \sum_{r \in R} \beta_r \cdot 1(|a|_r). \tag{2.2}$$

where $|a|_r$ represents the integer number of players choosing location r, and where the indicator is defined as

$$1(|a|_r) = \begin{cases} 1 & |a|_r > 0 \\ 0 & |a|_r = 0. \end{cases} \tag{2.3}$$

We note that the global objective is not binary, but rather is a gradient across all joint action profiles $a \in A$. We now consider what is known to a designer for this example:

[1] Motivation behind this example stems from the movie *Star Wars IV: A New Hope*, where in the opening scene, the robots C3P0 and R2D2 attempt to hide from the evil Darth Vader.

The Known Information:

- The designer knows the assigned probabilities, β_r, for all locations $r \in R$.
- The designer knows that there are only two robots.
- The designer knows that the robots cannot communicate with each other.
- The designer knows that each robot knows if the other robot is in a location that's in it's own location set.

In a centralized system, it would be straightforward control agents to select locations that jointly minimizes the global objective. However, additional constraints on the designer might make a centralized control impossible. In this example, we consider two constraints to the designer:

The Constraints:

- The designer is unsure about the possible hiding locations that are available to each robot.
- The designer cannot communicate with the robots when they are in this unknown environment.

The Design:

Given these constraint, a designer is tasked with designing a control algorithm to control the individual robots based on (1) the location probability values, and (2) the number of other robots in a location that is available to that robot. For this example, we consider the agent objective functions that are designed to be minimized, and denote this as $J : A \to R$. Clearly a designer wishes to make an objective function that encourages agents to select locations which will minimize the global cost function.

The Assessment Metrics:

Ideally, the designer would like to ensure desirable guarantees on the outcome of robots' decisions for each agent objective function that they design. These metrics are as follows:[1] We consider only the joint actions where both robots have finished minimizing their own agent objective function, and where neither robot can further decrease their function outcome by selecting an alternative location from their location set.

- *Metric #1: Price of Anarchy:* Out of these joint actions, we look at the resulting value of the action that maximizes the global cost function, C. We divide this value by the lowest value of the global cost function that is achievable out of any action profile in the joint action set, A. We call this ratio of the worst-case inefficiency that might result in this distributed system the "Price of Anarchy."

- *Metric #2: Price of Stability:* Out of these joint actions, we look at the resulting value of the action that minimizes the global cost function, C. We divide this value by the lowest value of the global cost function that is achievable out of any action profile in the joint action set, A. We call this ratio of the best-case inefficiency that might result in this distributed system the "Price of Stability."

For this thesis, we consider Metric #1 to be of greater importance than Metric #2. Hence, a designer would like to make agent objective functions than minimize the resulting price of anarchy and price of stability.

Suppose the set of locations for available to each player, and the (normalized)[2] probability values are those shown in Fig. 2.1. That is, agent 1 has access to locations 1 and 2, while agent 2 has access to locations 1 and 3.

[1] These assessment metrics are more rigorously defined in Section 2.6.

[2] That is, normalized by $\sum_{r \in \{\cup_{i \in N} A_i\}} p_r$.

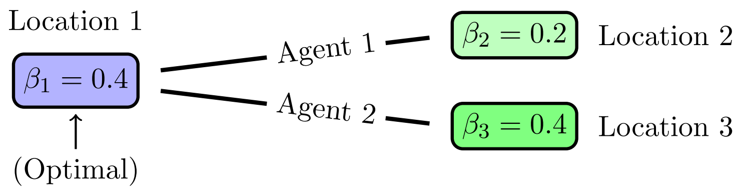


Figure 2.1: Example game illustrating the difficulties in designing player objective functions.

Possible Agent Objective Functions:

From a global perspective, the global cost function is minimized with a value of $\beta_2 = 0.4$ when both players select location 1. Two possible agent cost functions that might be implemented for both agents are the following:

- The *fixed cost function*: this agent cost function is a direct relationship to the location value. That is, $J_i(a) = \beta_{a_i}$ for each player $i \in \{1, 2\}$. We remark in this design, each player is controlled without regard to the action of the other player.

  ∗ *Assessment:* Given the agent cost function, player 2 is indifferent to either location, as both locations have the same value of 0.4. On the other hand, agent 1 will select location 2 since this is a cost of 0.2, and therefore their individual objective function is minimized by selecting resource 2. We see that the resulting global cost for action profiles $a = (2, 1)$ and $a' = (2, 3)$ are the same: $C(a) = C(a') = 0.2+0.4 = 0.6$. Hence, the price of anarchy and price of stability are equal and have a value of $0.6/0.4 = 1.5$.

- The *equal share cost function*: this agent cost function is a direct relationship to the location value, divided equally among the number of players selecting that resource. That is, $J_i(a) = \sum_{r \in \cup_{i \in N} a_i} \frac{\beta_r}{|a|_r}$, where $|a|_r$ is the number of players that select location r.

  ∗ *Assessment:* Given this agent cost function, there are two possible scenarios: If agent 1 selects location 2, and agent 2 selects location 3, then neither will be controlled to deviate from their decision, as player 1 will have maximized their agent cost function, and agent 2 is indifferent between the locations in its action set. Alternatively, if both players select location 1, then neither will have incentive to deviate, as the alternative location in their action sets would strictly increase their agent objective function. We see that the resulting global cost for these two action profiles $a = (2, 3)$ and $a' = (1, 1)$ are $C(a) = 0.2 + 0.4 = 0.6$ and $C(a') = 0.4$. Hence, the price of anarchy is valued as $0.6/0.4 = 1.5$, and the price of stability has a value of $0.4/0.4 = 1$.

Given these two possible agent cost functions implemented in the setup of Fig. 2.1, we see that the equal share cost function is more optimal in this game as, although it affords an equally worst-case inefficiency to that of the fixed function, it affords a more optimal best-case result that matches the globally desired output. Regardless, neither player cost function could successfully guarantee the globally desired output. This simple example demonstrates the non-trivial problem of player objective function design, and it illustrates how designs can lead to non-optimal outcomes in a distributed system. It is the aim of the designer to assess and limit the inefficiencies that may arise from these designs.

2.3 Control Algorithms: The Complete Picture

We recall the remarks in 1.1 that a control algorithm is comprised of an agent objective function and a *learning algorithm*. The agent objective functions form the basis for the underlying interactions for players in a given system, and may be designed so that an individual is influenced by the decisions of other agents. For functions that are designed to be minimized, these functions are represented with $J_i : A \to \mathcal{R}$.

The learning algorithm defines how agents process the information that is available to them in order to make a decision on the action that it will take (given this agent objective function). Many learning designs are well studied and include *Cournot adjustment process*, *fictitious play*, *log-linear learning*, and others. These designs consider a sequence of one-shot games over time steps

$t = \{1, 2, \dots\}$, and (possibly) relies on the information from previous games to dynamically adjust their effect on the control algorithm. Hence, an agent's objective function outcome for action a_i at time t may be be different that the outcome at time $t+1$. Therefore each players' action $a_i(t)$ in the action profile $a(t) = (a_1(t), a_2(t), \dots, a_n(t))$ is selected using a designed learning algorithm structured as

$$a_i(t) = \Pi_i \left(\{a(1), a(2), \dots, a(t)\}; J_i(\cdot) \right). \tag{2.4}$$

A final control algorithm can then be formulated, and additional assessments can be made on the completed control design. With this fundamental picture in mind, the thesis contributions herein focus solely on the design of the agent objective function.

2.4 Cost Minimization Games

Before we detail the metrics used to assess and compare player objective functions, we must first formally define the model that much of this thesis is based upon: cost minimization games.[1] Except when specifically noted, this thesis focuses on systems with global cost functions, and explores agent objective functions that are designed to be minimized by each agent.

Definition 1 (Cost Minimization Games)**.** *A family of cost minimization games, $\mathcal{G}$, consists of:*

- *a set of agents $\mathcal{N} = \{1, \dots, m\}$*[2]
- *a collection of action sets $\mathcal{A}_i = \{A_i', A_i'', \dots\}$ for each agent $i \in \mathcal{N}$ where*
    - ∗ *a collection of joint action sets is $\mathcal{A} = \{A^{(1)}, A^{(2)}, \dots\}$ where $A = \Pi_{i \in \mathcal{N}} A_i$ for any joint action set.*

[1] Parallel definitions exist for games in which the player objective is to maximize their control function, and where the global objective is to maximize a global function.

[2] We sometimes refer to these agents as *players* or *sub-entities* in this thesis.

  * *an action profile is* $a = \{a_1, \ldots, a_m\} \in A$

- *a set of player cost functions,* $J_i : A \rightarrow \mathcal{R}$,[1] *for all* $i \in \mathcal{N}$ *and* $A \in \mathcal{A}$.

- *a global cost function,* $C : A \rightarrow \mathcal{R}$ *for all* $A \in \mathcal{A}$

We denote a family of cost minimization games as $\mathcal{G} = \{\mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i\}_{i \in N}, C\}$, *and define a specific cost minimization game as* $G = \{N \subseteq \mathcal{N}, \{A_i \in \mathcal{A}_i\}_{i \in N}, \{J_i\}_{i \in N}, C\} \in \mathcal{G}$[2].

We represent the joint action $a = (a_1, \ldots, a_n) \in A$ in the conventionally accepted manner as $a = (a_i, a_{-i})$, where a_i indicates the action selected by player i, and a_{-i} denotes the joint action selected all players *other* than player i.

2.5 Equilibria Concepts

Player cost functions are commonly assessed by looking at the types of equilibria that might occur in a game, as there are often many ways in which these stationary equilibria can be reached via the dynamics that result from the individual control algorithms of each agent. This section defines the three most common types of equilibria concepts that are considered in this thesis.[3]

2.5.1 Pure Nash Equilibrium

The most notorious equilibria concept in Game Theory is the pure Nash equilibrium (or more commonly referred to as a *Nash equilibrium*, or pNE) originally presented by Nobel laureate Dr. John F. Nash[4] in his 28 page Princeton University Ph.D. mathematics thesis [16].

[1]For games in which the agent objective function is designed to be minimized, we use the notation, J_i to represent the resulting *cost* to agent i.

[2]We sometimes denote $\{A_i\}_{i \in N}$ using the joint action set A.

[3]Parallel equilibria concepts exist for players with utility functions in games with global welfare functions.

[4]June 13, 1928 – May 23, 2015

Definition 2 (Pure Nash Equilibrium). *An action profile $a = (a_i, a_{-i}) \in A$ is a pure Nash equilibrium action if, for all $a_i' \in A_i$ and for all players $i \in N$,*

$$J_i(a_i, a_{-i}) \leq J_i(a_i', a_{-i}). \tag{2.5}$$

A pure Nash equilibrium represents an action profile where no player $i \in N$ can decrease their agent cost function via a unilateral deviation. Since no individual player can decrease their cost by selecting an alternative action, therefore no player has any incentive to unilaterally deviate from their selected action. Thus, an equilibrium is achieved.

2.5.2 Mixed Nash Equilibrium

Almost equally famous is the mixed Nash equilibrium (mNE), a slightly more generalized equilibria concept that is based on the probability that other players select a given action. We consider each of n players $i \in N$ to have a probability of selecting an action, $a_i \in A_i$, with independent probability $p_i^{(a_i)}$ such that $\sum_{a_i \in A_i} p_i^{(a_i)} = 1$. We define a *strategy* to be a probability distribution for player i over their action set A_i to be $\sigma_i = \{p_i^{(a_1)}, p_i^{(a_2)}, \dots\} \in \Delta(A_i)$. We define $\sigma = \sigma_1 \times \cdots \times \sigma_n$ to be the joint probability distribution that is generated from these independent probability distributions of each player, and use the notation $\sigma_i^{a_i}$ to represent the probability that agent i will play action a_i. Then the expected outcome of the player cost function for the joint strategy $s = (\sigma_1, \dots, \sigma_n)$, given the joint probability distribution σ is denoted as $E_{s\sim\sigma}[J_i(a)]$, where

$$E_{s\sim\sigma}[J_i(s)] = \sum_{a \in A} J_i(a) \cdot \sigma_1^{a_1} \sigma_2^{a_2} \cdots \sigma_n^{a_n}. \tag{2.6}$$

Definition 3 (Mixed Nash Equilibrium). *A set of independent probability distributions $s = (\sigma_1, \dots, \sigma_n)$ is a mixed Nash equilibrium if the product of these distributions, $\sigma = \sigma_1 \times \cdots \times \sigma_n$, satisfies*

$$E_{s\sim\sigma}[J_i(s)] \leq E_{s_{-i}\sim\sigma_{-i}}[J_i(s_i', s_{-i})] \tag{2.7}$$

for all $s_i' \in \Delta(A_i)$ *and for all players* $i \in N$*, where* $s_{-i} = \{s_i, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n\}$ *represents the strategy set of all players other than player* i*, and where* $\sigma_{-i} = \sigma_1 \times \cdots \sigma_{i-1} \times \sigma_{i+1} \times \cdots \times \sigma_n$*.*

The motivation behind the definition for a mixed Nash equilibrium stems from the notion that the actions of other agents may be a probability distribution over their action sets.

2.5.3 Coarse Correlated Equilibria

A coarse correlated equilibrium (CCE) is the broadest equilibria concept that we are concerned with in this thesis. We look at probability distribution σ over the joint action set.

Definition 4 (Coarse Correlated Equilibrium)**.** *A joint probability distribution* σ *is a coarse correlated equilibrium if*

$$E_{a \sim \sigma}[J_i(a)] \leq E_{a \sim \sigma}[J_i(a_i', a_{-i})] \tag{2.8}$$

for all $a_i' \in A_i$ *and for all players* $i \in N$*.*

The common motivation for this equilibrium concept is to consider the scenario where each player $i \in N$ is given a choice to opt out and select any action, $a_i \in A_i$, or to opt in and play the action a_i that is assigned to them when an action $a \in A$ is drawn randomly according to the joint probability distribution σ. The coarse correlated equilibria is therefore when no player $i \in N$ has a unilateral incentive to opt out, given that all other players opt in.

We remark that every pNE is also a mNE, and every mNE is also a CCE; the reverse is not necessarily true. Additionally, while not all games may have a pure Nash equilibrium, all finite games have at least one mNE [17], and therefore also have at least one CCE.

2.6 Player Cost Function Metrics

With the equilibria concepts defined, we now present the metrics that we will use to the assess player objective functions. The primary and secondary quantification techniques are the Price of Anarchy and the Price of Stability.[1]

2.6.1 Price of Anarchy

The Price of Anarchy (PoA) is the ratio of the worst-case equilibrium of a global objective function to that of a global optimal outcome, and is a method of quantifying the effectiveness of a system when considering the selfish behavior of agents [11, 20]. Although often considered in terms of the pNE, it also extends to the other equilibria concepts as well.

Definition 5 (Price of Anarchy)**.** *Consider a family of games $\mathcal{G} = \{\mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{\mathcal{J}_i\}_{i\in N}, C\}$. For a game $G \in \mathcal{G}$, let $\varepsilon(G)$ be the set of equilibria, and let $a^* \in A$ be the action that minimizes the global objective function. The Price of Anarchy, $PoA(G)$, is*

$$PoA(G) := \max_{a\in\varepsilon(G)} \frac{C(a)}{C(a^*)}, \tag{2.9}$$

and the Price of Anarchy for the family of games $G \in \mathcal{G}$, $PoA(\mathcal{G})$, is

$$PoA(\mathcal{G}) := \sup_{G\in\mathcal{G}} PoA(G). \tag{2.10}$$

2.6.2 Price of Stability

The Price of Stability (PoS) is the ratio of the best-case equilibrium of a global objective function to that of a global optimal outcome, and is a method of quantifying the best-case inefficiencies of a system when considering the selfish behavior of agents [20, 24]. Similarly to the

[1] Parallel metrics exist for games with global welfare functions.

PoA, while the PoS is often considered in terms of the pNE, it also extends to the other equilibria concepts.

Definition 6 (Price of Stability)**.** *Consider a family of games* $\mathcal{G} = \{\mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{\mathcal{J}_i\}_{i\in N}, C\}$*. For a game* $G \in \mathcal{G}$*, let* $\varepsilon(G)$ *be the set of equilibria, and let* $a^* \in A$ *be the action that minimizes the global objective function. The Price of Stability,* $PoS(G)$*, is*

$$PoS(G) := \min_{a\in\varepsilon(G)} \frac{C(a)}{C(a^*)}, \tag{2.11}$$

and the Price of Stability for the family of games $G \in \mathcal{G}$*,* $PoS(\mathcal{G})$*, is*

$$PoS := \sup_{G\in\mathcal{G}} PoS(G). \tag{2.12}$$

From these definitions, we remark that

$$1 \leq PoS(G) \leq PoA(G) \tag{2.13}$$

for all $G \in \mathcal{G}$, and

$$1 \leq PoS(\mathcal{G}) \leq PoA(\mathcal{G}) \tag{2.14}$$

We are interested in designing player objective functions that primarily minimize the $PoA(\mathcal{G})$, and secondarily minimize the $PoS(\mathcal{G})$.

2.7 Summary

In Chapter 2 we introduced a brief overview and motivation behind the thesis topic. Following, we defined cost minimization games, and reviewed the three equilibria concepts pertinent to this thesis. We then defined the Price of Anarchy and the Price of Stability, the two agent cost function assessment metrics. The contents of this chapter are by no means complete, and readers are encouraged to review [2, 7, 9, 18, 19, 21, 22, 23, 26] and related works for more information on game theory and multi-agent systems.

Chapter 3

Smooth Games, Potential Games, and the Budget

3.1 Outline

Chapter 2 introduced methods to quantify agent objective functions. However, finding the PoS and PoA may be non-trivial, as it requires a deep understanding of how the game structure and agent objective functions coalesce. Thus, it may instead be desired (and easier) to find a *bound* on the PoA and PoS for a family of games. It therefore becomes a question of whether methods exist to easily bound the resulting PoS and PoA of a family of games. Incredibly, for many game sets, the answer is yes. This chapter presents two such methods that correspond to two games known as smooth games and potential games. The significance of smooth games is that, given minor constraints, they imply an upper bound on the PoA for all equilibria types, many cases in which this bound is tight.[1] The significance of potential games is that they imply an upper bound on the PoS for all equilibria types. In this chapter, we will expand upon smooth games to (1) decrease the implied bounds, and (2) grant bounds for a family of games. A significant topic in this chapter focuses on the *budget*, which is a rigorous way of coupling the summation of agent objective function outcomes to the global objective function outcome for the set of joint actions.

We begin by introducing smooth games, and reviewing the upper PoA bounds. Following,

[1]By *tight* we mean that for a family of games, $\mathcal{G}$, there exists a game $G \in \mathcal{G}$ where the upper bound is strictly equal to $PoA(\mathcal{G})$

we define the budget. Then we combine the budget with smooth games to this to expand the PoA bounds currently afforded for smooth games. We then present bounds for the PoS for potential games that are similar to those bounds for smooth games. The relevance of the results from this chapter will be illustrated using example games in Chapters 4 and 6.

3.2 Smooth Games

A smooth game is a cost minimization game that satisfies a defined constraint. The motivation for introducing smooth games is that they admit an upper bound on the PoA; this bound is equivalent across all equilibria types discussed in Chapter 2 [20].[1]

Definition 7 (Smooth Games [20])**.** *A smooth game is a cost minimization game,* $G = \{N, \{A_i\}_{i\in N}, \{J_i\}_{i\in N}, C\}$ *that satisfies the smoothness argument:*

$$\sum_{i\in N} J_i(a_i^*, a_{-i}) \leq \lambda \cdot C(a^*) + \mu \cdot C(a) \tag{3.1}$$

for all actions $a, a^* \in A$*. A smooth game,* G*, is said to be* (λ, μ)*-smooth with smoothness parameters* λ *and* μ*.*

3.2.1 Smooth Game Upper PoA Bound

We present the PoA upper bound for a (constrained)[2] smooth game now:

Theorem 1 (Smooth Game Upper PoA Bound (Constrained) [20])**.** *Let a cost minimization game,* $G = \{N, A, \{J_i\}_{i\in N}, C\}$*, be* (λ, μ)*-smooth, where* $\sum_{i\in N} J_i(a) = C(a)$*, and where* $\lambda > 0$ *and* $\mu < 1$*.*

[1] An additional property exist that expands the set of smoothness parameters of a smooth game. However, as the contribution of this property to player objective function design is uncertain, it is provided in Appendix C for the more curious readers.

[2] By *constrained* we mean a cost minimization game $G = \{N, A, \{J_i\}_{i\in N}, C\}$, with the constraint that $\sum_{i\in N} J_i(a) = C(a)$.

Then for all equilibria types,

$$PoA(G) \leq \frac{\lambda}{1-\mu}. \tag{3.2}$$

We remark that there may exist a set of smoothness parameters, $\{(\lambda_1, \mu_1), \ldots, (\lambda_m, \mu_m)\}$, that satisfy the smoothness argument for a given game. A notion called the *robust PoA* was introduced to define the lowest bound on the PoA that could be proven using this set.

Definition 8 (Robust PoA (Constrained) [20])**.** *For a (constrained) cost minimization game that is* (λ, μ)*-smooth, the robust PoA is:*

$$\inf\left\{\frac{\lambda}{1-\mu} : (\lambda, \mu) \text{ such that the game is } (\lambda, \mu) - \text{smooth and } \lambda > 0 \text{ and } \mu < 1\right\}. \tag{3.3}$$

Fig. 3.1 plots the resulting upper PoA bound that is implied through Eq. (3.2) for some (λ, μ) pairs satisfying $\lambda > 0$ and $\mu < 1$.

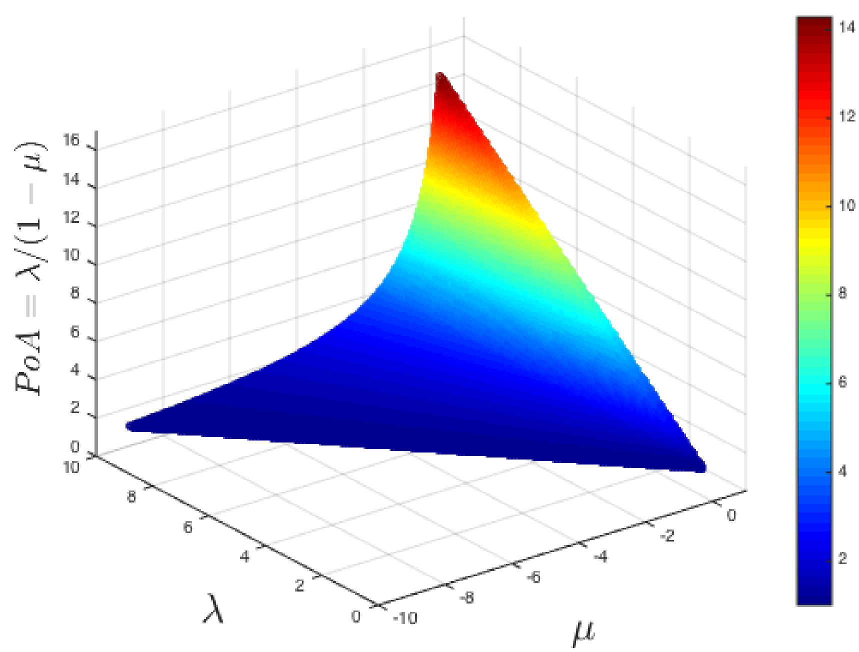


Figure 3.1: The resulting upper PoA bound that is implied through Eq. (3.1) for some (λ, μ) pairs satisfying $\lambda > 0$ and $\mu < 1$.

3.2.2 The Budget

To upper bound the PoA for smooth games,[1] we must first introduce the *budget*.

Definition 9 (Budget and Lower Budget)**.** *Consider a cost minimization game* $G = \{N, A, \{J_i\}_{i \in N}, C\}$*.*

- *The budget,* $B(G)$*, is*

$$B(G) := \max_{a \in A} \left[\frac{C(a)}{\sum_{i \in N} J_i(a)} \right]. \tag{3.4}$$

For the family of games $G \in \mathcal{G}$*, the budget,* $B(\mathcal{G})$*, is*

$$B(\mathcal{G}) := \sup_{G \in \mathcal{G}} B(G). \tag{3.5}$$

- *The lower budget,* $\overline{B}(G)$*, is*

$$\overline{B}(G) := \max_{a \in A} \left[\frac{C(a)}{\sum_{i \in N} J_i(a)} \right]. \tag{3.6}$$

For the family of games $G \in \mathcal{G}$*, the lower budget,* $\overline{B}(\mathcal{G})$*, is*

$$\overline{B}(\mathcal{G}) := \inf_{G \in \mathcal{G}} B(G). \tag{3.7}$$

We say that the budget is in one of three states:[2]

(1) *budget deficit* when $B(G) > 1$;

(2) *budget balanced* when $B(G) = 1$;

(3) *budget surplus* when $B(G) < 1$.

[1] We omit the term *constrained* to mean that $\sum_{i \in N} J_i(a)$ does not necessarily equal $C(a)$.

[2] We remark that these budget conditions are strictly greater than zero, and strictly less than infinity.

3.2.3 Smooth Game Upper PoA Bound

The PoA upper bounds for smooth games are now expanded for a game with any budget condition, and for all equilibria types.

Theorem 2 (Smooth Game PoA Upper Bound)**.** *Let a cost minimization game, $G = \{N, A, \{J_i\}_{i \in N}, C\}$, be (λ, μ)-smooth with budget $B(G)$ satisfying $\lambda > 0$ and $\mu B(G) < 1$. Then for all equilibria types,*

$$PoA(G) \leq \frac{\lambda B(G)}{1 - \mu B(G)}. \tag{3.8}$$

Proof. Assume a cost minimization game $G = \{N, A, \{J_i\}_{i \in N}, C\}$ that is (λ, μ)-smooth with budget $B(G)$, coarse correlated equilibrium σ, and where $\lambda > 0$ and $\mu B(G) < 1$. Consider any action $a = (a_i, a_{-i})$ selected according to the joint probability distribution σ, and let $a^* = (a_i^*, a_{-i}^*)$ be the action that minimizes C. Then by Eq. (3.1) we have:

$$\mathbf{E}_{a \sim \sigma}[C(a)] \leq \mathbf{E}_{a \sim \sigma}\left[B(G)\left(\sum_{i \in N} J_i(a)\right)\right] \tag{3.9}$$

$$= B(G)\left(\sum_{i \in N} \mathbf{E}_{a \sim \sigma}\left[J_i(a)\right]\right) \tag{3.10}$$

$$\leq B(G)\left(\sum_{i \in N} \mathbf{E}_{a \sim \sigma}\left[J_i(a_i^*, a_{-i})\right]\right) \tag{3.11}$$

$$= B(G) \cdot \mathbf{E}_{a \sim \sigma}\left[\sum_{i \in N} J_i(a_i^*, a_{-i})\right] \tag{3.12}$$

$$\leq B(G) \cdot \mathbf{E}_{a \sim \sigma}\left[\lambda C(a^*) + \mu C(a)\right] \tag{3.13}$$

$$= \lambda B(G) \cdot C(a^*) + \mu B(G) \cdot \mathbf{E}_{a \sim \sigma}\left[C(a)\right] \tag{3.14}$$

$$\Rightarrow \frac{\mathbf{E}_{a \sim \sigma}[C(a)]}{C(a^*)} \leq \frac{\lambda B(G)}{1 - \mu B(G)} \tag{3.15}$$

$$\Rightarrow PoA(G) \leq \frac{\lambda B(G)}{1 - \mu B(G)}. \tag{3.16}$$

□

Eq. (3.9) follows from the budget condition. The importance of Theorem 2 should not be overlooked: incorporating the budget condition affords a bound on the PoA for smooth games for all budget conditions, *even* in budget deficit scenarios, and tightens the PoA bound implied in Section 3.2 for strictly budget relaxed scenarios. From a design perspective, this suggests that the PoA bound for smooth games is less optimal when the budget is in deficit.

We similarly expand the definition for the robust PoA.[1]

Definition 10 (Robust PoA)**.** *The robust PoA for a cost minimization game, G, with budget $B(G)$, that is (λ, μ)-smooth is:*

$$\inf\left\{\frac{\lambda}{1-\mu} : (\lambda, \mu) \text{ such that the game is } (\lambda, \mu) - \text{smooth and } \lambda > 0 \text{ and } \mu B(G) < 1\right\}. \quad (3.17)$$

3.3 Welfare Maximization Equivalence

We take a brief interlude on cost minimization games to present the equivalent theorems for welfare maximization games. As the theorems in this section parallel those already presented, we omit the proofs. We first present welfare maximization games, then we introduce welfare maximization smooth games, define the budget conditions, and conclude with a theorem to bound the PoA for this family of games. Following this section, we will focus entirely on cost minimization game models.

Informally we define the family of welfare maximization games as the family of cost minimization games $\mathcal{G} = \{\mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i\}_{i\in N}, C\}$ with two modifications:

- Instead of agent cost functions $\{J_i\}_{i\in N}$ that each agent aims to minimize, we have agent utility functions, $\{U_i : \mathcal{A} \to \mathcal{R}\}_{i\in N}$ that each agent aims to maximize.

[1] Additional theorems from [20] for repeated play and no-regret sequences are peripheral to the main contents of this thesis topic; for completeness, the expansions for these theorems with the budget taken into account are provided in Appendix A.

- Instead of a global cost function, C, that is desired to be minimized, we have a global welfare function, $W : \mathcal{A} \to \mathcal{R}$ that is desired to be maximized.

Definition 11 (Smooth Games (Welfare Maximization) [20]). *A (welfare maximization) smooth game is a welfare maximization game, $G = \{N, \{A_i\}_{i\in N}, \{U_i\}_{i\in N}, W\}$ that satisfies the smoothness argument:*

$$\sum_{i\in N} U_i(a_i^*, a_{-i}) \leq \lambda \cdot W(a^*) - \mu \cdot W(a) \tag{3.18}$$

for all actions $a, a^* \in A$. A smooth game, G, is said to be (λ, μ)-smooth with smoothness parameters λ and μ.

To upper bound the PoA for welfare maximization smooth games, we first introduce the *budget*.

Definition 12 (Budget and Lower Budget (Welfare Maximization)). *Consider a welfare maximization game $G = \{N, A, \{J_i\}_{i\in N}, C\}$. The (lower) budget, $B_W(G)$, is*

$$B_W(G) := \min_{a\in A} \left[\frac{C(a)}{\sum_{i\in N} J_i(a)} \right]. \tag{3.19}$$

For the family of games $G \in \mathcal{G}$, the budget, $B_W(\mathcal{G})$, is

$$B_W(\mathcal{G}) := \inf_{G\in\mathcal{G}} B_W(G). \tag{3.20}$$

For welfare maximization games, we consider the *lower* PoA bound, as we define the PoA to be the welfare resulting from the worst-case equilibrium over the resulting welfare for the action that maximizes this function. Hence, the resulting PoA will be no greater than one. The PoA lower bounds for welfare maximization smooth games are now expanded for a game with any budget condition, and for all equilibria types.

Theorem 3 (Smooth Game PoA Upper Bound). *Let a welfare maximization game, $G = \{N, A, \{U_i\}_{i\in N}, W\}$, be (λ, μ)-smooth with budget $B_W(G)$ satisfying $\lambda > 0$ and $\mu B_W(G) > -1$.*

Then for all equilibria types,

$$PoA(G) \leq \frac{\lambda B_W(G)}{1 + \mu B_W(G)}. \tag{3.21}$$

This concludes the section on welfare maximization games, and we now return to cost minimizatoin games for the remainder of this thesis.

3.4 Potential Games

Potential games are games in which the change in every individual's cost function can be globally expressed by a *potential function.* These games are important in this thesis, as they are useful in upper bounding the best-case game inefficiencies that might result in a system. We first present the definition for an (exact) potential game, then we present the PoS upper bounds that can be inferred by them.

Definition 13 (Exact Potential Game [12, 15])**.** *An exact potential game is a cost minimization game, $G = \{N, A, \{J_i\}_{i \in N}, C\}$, with a global function, $\phi : A \rightarrow \mathcal{R}$ such that for every agent $i \in N$, every action $a_i', a_i'' \in A_i$, and every action $a_{-i} \in A_{-i}$,*

$$J_i(a_i', a_{-i}) - J_i(a_i'', a_{-i}) = \phi(a_i', a_{-i}) - \phi(a_i'', a_{-i}). \tag{3.22}$$

For cost minimization games, the action(s) that minimize the potential function is also a pure Nash equilibrium [12]. Thus, potential games have at least one pure Nash equilibrium. We also remark that the pure Nash equilibrium action that defines the PoS, a^{NE}, results in a potential at least as low as any other action, $a \in A$, including the action that optimally minimizes the global objective function, a^*. That is,

$$\phi(a^{NE}) \leq \phi(a^*) \tag{3.23}$$

Using Eq. (3.23) to bound the Price of Stability is a well known technique referred as the *potential function method* [21].

3.4.1 Potential Game Upper PoS Bound

We now aim to bound the resulting PoS. We first define upper and lower potential budgets for the potential game, and use this the upper bound the PoS. It is remarkable how a budget is used to bound both the PoA in Section 3.2.3 and the PoS in this section, and it draws attention to a possible link between smooth and potential games.

Definition 14 (Potential Budget)**.** *Let a cost minimization game, $G = \{N, A, \{J_i\}_{i\in N}, C\}$, be a potential game with potential function ϕ.*

- *The upper potential budget, $B_\phi(G)$, is defined as*

$$B_\phi(G) := \max_{a\in A}\left[\frac{C(a)}{\phi_i(a)}\right] \tag{3.24}$$

  For the family of games $G \in \mathcal{G}$, the upper potential budget, $B_\phi(\mathcal{G})$, is defined as

$$B_\phi(\mathcal{G}) := \sup_{G\in\mathcal{G}} B_\phi(G) \tag{3.25}$$

- *The lower potential budget, $\overline{B}_\phi(G)$, is defined as*

$$\overline{B}_\phi(G) := \min_{a\in A}\left[\frac{C(a)}{\phi(a)}\right] \tag{3.26}$$

  For the family of games $G \in \mathcal{G}$, the lower potential budget, $\overline{B}_\phi(\mathcal{G})$, is defined as

$$\overline{B}_\phi(\mathcal{G}) := \inf_{G\in\mathcal{G}} \overline{B}_\phi(G) \tag{3.27}$$

Using Definition 14, we now introduce a bound on the PoS for potential games. The theorem is not original, but is a more exact version of the theorem originally presented in [21] (*see* Appendix B for original theorem).

Theorem 4 (Smoothness pNE PoS Bound[1])**.** [2] *Let a cost minimization game $G = \{N, A, \{J_i\}_{i\in N}, C\}$ be a potential game with potential function ϕ. Suppose the upper and lower*

[1] *See* Appendix B for original theorem.

[2] We remind the reader that for a game G, $PoS(G) \geq 1$.

potential budgets are $B_\phi(G)$ *and* $\overline{B}_\phi(G)$*, respectively, in the family of games,* $\mathcal{G}$*. Then*

$$PoS(G) \leq \frac{B_\phi(G)}{\overline{B}_\phi(G)} \leq \frac{B_\phi(\mathcal{G})}{\overline{B}_\phi(\mathcal{G})}. \tag{3.28}$$

Proof. Let a cost minimization game $G = \{N, A, \{J_i\}_{i\in N}, C\}$ be a potential game with potential function ϕ, with upper potential budget $B_\phi(G)$ and lower potential budget $\overline{B}_\phi(G)$. Let action profile $a = (a_i, a_{-i})$ be the pure Nash equilibrium that minimizes the potential function (and is therefore the action that defines $PoS(G)$), and let action profile $a^* = (a_i^*, a_{-i}^*)$ be the action that minimizes C. Then

$$C(a) \leq B_\phi(G) \cdot \phi(a) \tag{3.29}$$

$$\leq B_\phi(G) \cdot \phi(a^*) \tag{3.30}$$

$$\leq B_\phi(G) \cdot \frac{C(a^*)}{\overline{B}_\phi(G)} \tag{3.31}$$

$$\Rightarrow \frac{C(a)}{C(a^*)} \leq \frac{B_\phi(G)}{\overline{B}_\phi(G)} \tag{3.32}$$

$$\leq \frac{B_\phi(\mathcal{G})}{\overline{B}_\phi(\mathcal{G})}. \tag{3.33}$$

□

In regards to agent cost function design, Theorem 4 reveals how the price of stability of a potential game is upper bounded more optimally[1] when the upper and lower potential budgets have minimal difference. Since the potential function is a direct result of the agent objective functions of a game, a designer should consider this when creating these agent functions.

Additionally, re-framing Theorem 30 into a similar framework as that of the smooth game PoA bound (Theorem 2) suggests a connection between smooth games and potential games, as the role of a budget in each is similar.

[1] That is, the upper bound on the PoS is closer to 1.

3.5 Chapter Summary

In Chapter 3, we introduced smooth games. We then defined a budget for cost minimization games, and used this budget to expand the implied bounds on the PoA for smooth games.

Although hypothesized, it remains an open question as to whether the budget always affects the actual PoA for all smooth games (as the implied bound is not necessarily tight), as well as the PoA for games that are not smooth.

One unanswered question is whether it would be useful in defining a *player specific budget* for each player, rather than for the set of players. That is, considering a cost minimization game $G = \{N, A, \{J_i\}_{i \in N}, C\}$, a player budget, $B_i(G)$, for each player $i \in N$ could be defined as

$$B_i(G) = \max_{a \in A} \frac{C(a)}{J_i(a)}, \tag{3.34}$$

and would specify more precisely the role of budget for each player.

As noted in Section 3.4.1, Theorem 4 demonstrates how the PoS of a potential game will have a more optimal upper bound when the upper and lower potential budgets have a smaller difference. Designers that agent objective functions for cost minimization games that induce a potential function should keep this in mind. Additionally, the similar roles of the budget in Sections 3.2 and 3.4.1 suggests a connection between smooth games and potential games, and hints at a tradeoff between the PoA and PoS upper bounds. Future research should investigate this postulation.

Chapter 4

Reverse Carpooling

4.1 Outline

Chapter 3 expanded the PoA bounds for smooth games to allow for any budget condition (Theorem 2). This chapter will demonstrate the utility of those results by applying them to a specific type of network coding resource allocation game known as *reverse carpooling.* After defining this game, we introduce a generic agent function and bound the PoA. This specific game is used to motivate the theory of Chapter 3 because it is already extensively studied in terms of the PoA and PoS. We compare the results to functions designed in [12] and [**?**], and discuss the tightness of our proof.

4.2 Reverse Carpooling

Research on network coding aims to maximize transmitted information, minimize power consumption, and improve signal integrity [12]. A reverse carpooling game is one that aims to reduce power consumption by transmitting two signals that are being transmitted in a network in opposite directions at a given point with a single signal. The method, called reverse carpooling, allows for any intermediate node to send the bitwise sum (XOR) of the two received signals, as the recipients will then be able to decode their own message by subtracting away their own original message. Fig. 4.1 illustrates this where two transmissions, a and b, are sent from nodes v_1 and v_3 via intermediate node v_2. The intermediate node then broadcasts the XOR of these two messages to the connected nodes, whereby each message can by deciphered. By doing this, the amount of power required to send two separate signals in opposite directions on a network is equivalent to sending a single signal in only one direction.

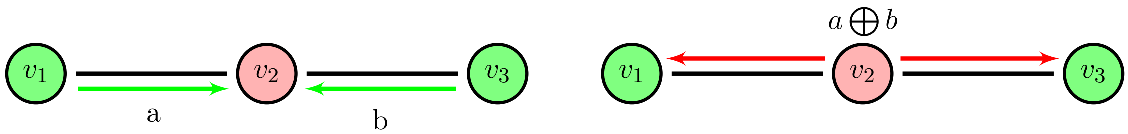


Figure 4.1: Illustration of reverse carpooling, as depicted in [12].

4.2.1 Game Definition

The reverse carpooling network is defined by the set of edges, $E = \{e_1, \dots e_k\}$, and the set of vertices, $V = \{v_1, \dots, v_m\}$, The reverse carpooling game is defined by the following components:[1]

- a set of nodes $V = \{v_1, \dots, v_m\}$,
- a set of edges $E = \{e_1, \dots e_k\}$, which connect two nodes to one another and is denoted as $e_l = (v_k, v_{k+1})$. The set of nodes that are connected via a single vertex to node v_i are defined as the neighbors of v_i, and denoted as $\mathcal{N}(v_i)$
- a finite set of agents $N = \{1, \dots, n\}$, where each agent is a source, s_i, and destination, t_i, pair, $(s_i, t_i) \in V^2$
- action sets A_i for all agents $i \in N$, that is a set of transmission paths from s_i to t_i that agent i may take
  - ∗ each action is a set of nodes; that is, action $a_i = \{s_i, v_2, \dots, v_{|a_i|}\}$, where $|a_i|$ is the number of nodes in action a_i, and $v_{k+1} \in \mathcal{N}(v_k)$

[1]We define this game from [12, **?**].

- a unique detailed path for action path a_i can be denoted in a reverse carpooling game from a source node to a destination node as

$$I(a_i) := \{v_1[\varnothing, v_2], v_2[v_1, v_3], \ldots, v_{|a_i|-1}[v_{|a_i|-2}, t_i]\}, \tag{4.1}$$

where, for agent $i \in N$, $|a_i|$ represents the length of action path a_i, and $v_x[v_{x-1}, v_{x+1}]$ represents the path from node v_{x-1} to node v_{x+1} through intermediate node v_x; hence, a agent's path is comprised of a set of nodes, with directionality

* the joint action set $A = \Pi_{i \in N} A_i$ represents the set of all transmission paths that can be taken by all agents

  - for any $a \in A$, we use $|a|_{v[v_x,v_y]}$ to represent the number of agents sending a transmission from node v_x to v_y via node v, *i.e.*

$$|a|_{v[v_x,v_y]} := |\{i \in N : v[v_x, v_y] \in I(a_i)\}| \tag{4.2}$$

- agent cost functions, $J_i : A \to \mathcal{R}$, that is a function of the number of agents that send a transmission in the same direction as agent i at a given node, and the set of agents $T \subseteq N$ that send a transmission in the opposite direction as agent i at a given edge; that is, $J_i(a) = \sum_{v[v_x,v_y] \in I(a_i)} f(i, \{a\}_{v[v_x,v_y]}, \{a\}_{v[v_x,v_y]})$; we remark that this agent cost function is designed to be oblivious to the network topology

- a system cost $C : A \to \mathcal{R}$ which is the total number of unique transmissions sent in the network; we elaborate on this cost now:

  * the cost for transmission at a node $v \in V$ for an action profile $a \in A$ is

$$C_v(a) := \sum_{(v_x,v_y) \in \mathcal{N}(v)^2 : x > y} \max\{|a|_{v[v_x,v_y]}, |a|_{v[v_y,v_x]}\} \tag{4.3}$$

* the system cost is defined as $C(a) := \sum_{v \in V} C_v(a)$

A reverse carpooling games is denoted as $G = \{N, A, \{J_i\}_{i \in N}, C\}$[1]

We introduce alternative notation to facilitate agent cost function design. We represent the number of nodes on player i's path where the number of transmissions going in the same direction as player i's transmission is strictly *greater* than the number of transmissions going in the opposite direction as $N_i^{(>)}$, and define it using the indicator function $1\{\cdot\}$ as

$$N_i^{(>)}(a) = \sum_{v[v_x, v_y] \in I(a_i)} 1\{v[v_x, v_y] > v[v_x, v_y]\}. \tag{4.4}$$

Likewise we define $N_i^{(<)}(a)$ and $N_i^{(=)}(a)$ using similar definitions, where $N_i^{(=)}(a)$ represents the number of nodes with an equal number of transmissions going in either direction, and where $N_i^{(<)}(a)$ is the number of nodes on player i's path such that the number of transmissions going in the same direction is strictly less than the number of transmissions going in the opposite direction.

Then the fraction that an individual player, $i \in N$, contributes to the system cost can be denoted as $V_i : A \to \mathcal{R}$, where

$$V_i(a) := N_i^{(>)}(a) + \frac{1}{2} N_i^{(=)}(a), \tag{4.5}$$

And the system cost, C, is [12]

$$C(a) = \sum_{i \in N} V_i(a). \tag{4.6}$$

4.2.2 Generic Agent Cost Function

Using the previous notation, we can broadly define an agent cost function as

$$J_i^{(g)}(a; \alpha, \beta, \gamma) = \alpha N_i^{(>)}(a) + \beta N_i^{(=)}(a) + \gamma N_i^{(<)}(a), \tag{4.7}$$

[1] A specific reverse carpooling game is denoted as $G = \{N, A, \{J_i\}_{i \in N}, C, \{E, V\}\}$, with the game network defined by the edge and node sets.

for any constants, α, β, and γ that are each non-negative real numbers. We call this the *generic agent cost function.* We use the term *generic* to imply that a designer of this agent cost function would need to specify the constants before implementing it in a reverse carpooling setting.

4.3 PoA Bound on the Generic Agent Cost Function

Using the results from Chapter 3, the PoA of a broadly defined agent cost function can be bounded.

Theorem 5 (Reverse Carpooling: Generic Player Cost Function PoA Bound)**.** *Let any reverse carpooling game $G = \{N, A, \{J_i^{(g)}\}_{i\in N}, C\} \in G$ implement player cost function*

$$J_i^{(g)}(a;\alpha,\beta,\gamma) = \alpha N_i^{(>)}(a) + \beta N_i^{(=)}(a) + \gamma N_i^{(<)}(a), \tag{4.8}$$

for all $i \in N$, with constants, $\alpha, \beta, \gamma \in \mathcal{R}^+$. Then the game is a (λ, μ)-smooth game with

$$\lambda = 2 \cdot \max\{\alpha, \beta, \gamma\} \tag{4.9}$$

$$\mu = 0. \tag{4.10}$$

Furthermore, the game has a (upper) budget, $B(G)$, bound above by

$$B(G;\alpha,\beta,\gamma) \leq \frac{1}{\min\{\alpha, 2\beta\}}, \tag{4.11}$$

and has a PoA upper bounded by

$$1 \leq PoA(G) \leq 2 \cdot \frac{\max\{\alpha,\beta,\gamma\}}{\min\{\alpha, 2\beta\}}. \tag{4.12}$$

Proof. Consider a reverse carpooling game $G = \{N, A, \{J_i^{(g)}\}_{i\in N}, C\} \in G$ that implements player cost function

$$J_i^{(g)}(a;\alpha,\beta,\gamma) = \alpha N_i^{(>)}(a) + \beta N_i^{(=)}(a) + \gamma N_i^{(<)}(a), \tag{4.13}$$

for all $i \in N$, and with constants, $\alpha, \beta, \gamma \in \mathcal{R}^+$. The smoothness parameters for this player cost function are derived for any actions $a = (a_i, a_{-i}),\ a^* = (a_i^*, a_{-i}^*) \in A$ as follows:

$$\sum_{i \in N} J_i^{(g)}(a_i^*, a_{-i}; \alpha, \beta, \gamma) = \sum_{i \in N} \left[\alpha N_i^{(>)}(a_i^*, a_{-i}) + \beta N_i^{(=)}(a_i^*, a_{-i}) + \gamma N_i^{(<)}(a_i^*, a_{-i})\right] \tag{4.14}$$

$$\leq \sum_{i \in N} \left[\max\{\alpha, \beta, \gamma\} \cdot |I(a_i^*)|\right] \tag{4.15}$$

$$= \max\{\alpha, \beta, \gamma\} \cdot \sum_{i \in N} \left[N_i^{(>)}(a_i^*) + N_i^{(=)}(a_i^*) + N_i^{(<)}(a_i^*)\right] \tag{4.16}$$

$$\leq \max\{\alpha, \beta, \gamma\} \cdot \sum_{i \in N} \left[2N_i^{(>)}(a_i^*) + N_i^{(=)}(a_i^*)\right] \tag{4.17}$$

$$= 2 \cdot \max\{\alpha, \beta, \gamma\} \cdot \sum_{i \in N} \left[N_i^{(>)}(a_i^*) + \frac{1}{2} N_i^{(=)}(a_i^*)\right] \tag{4.18}$$

$$= 2 \cdot \max\{\alpha, \beta, \gamma\} \cdot C(a^*) \tag{4.19}$$

Comparing this to Eq. (3.1), we have that the family of games is smooth with smooth parameters

$$\lambda = 2 \cdot \max\{\alpha, \beta, \gamma\} \tag{4.20}$$

$$\mu = 0. \tag{4.21}$$

Note that $|I(a_i^*)|$ in Eq. (4.15) is the total number of nodes that are on player i's path. Eq. (4.17) follows since $\sum_{i \in N} N_i^{(>)}(a) \geq \sum_{i \in N} N_i^{(<)}(a)$. By Theorem 2, the PoA is bound above by

$$PoA(G; \alpha, \beta, \gamma) \leq \frac{\lambda B(G; \alpha, \beta, \gamma)}{1 - \mu B(G; \alpha, \beta, \gamma)} \tag{4.22}$$

where the budget, $B(G; \alpha, \beta, \gamma)$, is defined as

$$B(G; \alpha, \beta, \gamma) = \max_{a \in \mathcal{A}} \left(\frac{C(a)}{\sum_{i \in N} J_i(a; \alpha, \beta, \gamma)} \right). \tag{4.23}$$

Finding an upper bound on $B(G; \alpha, \beta, \gamma)$, we have:

$$B(G; \alpha, \beta, \gamma) = \max_{a \in \mathcal{A}} \left(\frac{C(a)}{\sum_{i \in N} J_i(a)} \right) \tag{4.24}$$

$$= \max_{a \in \mathcal{A}} \left(\frac{C(a)}{\sum_{i \in N} \left(\alpha N_i^{(>)}(a) + \beta N_i^{(=)}(a) + \gamma N_i^{(<)}(a)\right)} \right) \tag{4.25}$$

$$\leq \max_{a \in \mathcal{A}} \left(\frac{C(a)}{\sum_{i \in N} \left(\alpha N_i^{(>)}(a) + \beta N_i^{(=)}(a) \right)} \right) \tag{4.26}$$

$$\leq \max_{a \in \mathcal{A}} \left(\frac{C(a)}{\min\{\alpha, 2\beta\} \cdot \sum_{i \in N} \left(N_i^{(>)}(a) + \frac{1}{2} N_i^{(=)}(a) \right)} \right) \tag{4.27}$$

$$= \max_{a \in \mathcal{A}} \left(\frac{C(a)}{\min\{\alpha, 2\beta\} \cdot C(a)} \right) \tag{4.28}$$

$$= \max_{a \in \mathcal{A}} \left(\frac{1}{\min\{\alpha, 2\beta\}} \right) \tag{4.29}$$

$$= \frac{1}{\min\{\alpha, 2\beta\}}. \tag{4.30}$$

Then from Eq. (4.22),

$$PoA(G; \alpha, \beta, \gamma) \leq 2 \cdot \frac{\max\{\alpha, \beta, \gamma\}}{\min\{\alpha, 2\beta\}}. \tag{4.31}$$

□

Corollary 1. *The upper bound on the price of anarchy from Theorem. 5 is minimized when*

$$2\beta \geq \alpha \geq \max\{\beta, \gamma\} \geq \min\{\beta, \gamma\}. \tag{4.32}$$

This result motivates the findings from Chapter 3, as it provides a generic player cost function that can be "tuned" by a designer so that an upper bound on the PoA can be guaranteed.

We remark that the authors of [12] proved that the price of anarchy for the set of reverse carpooling games which implement any player cost function as constrained by Section 4.2.1 was no less than two [12].

4.4 Summary

In this chapter we reviewed reverse carpooling, and used it as a way to demonstrate how Theorem 2 from Chapter 3 can be applied to a broadly defined agent cost function to substantially decrease the set of optimal agent cost functions. This game, combined with the broadly defined generic agent cost function, beautifully demonstrates the simplicity in affording worst-case efficiency guarantees across a broad spectrum of agent functions. We remark that no effort was even spent in determining the actual budget; rather the budget was only bounded.

We remark that this example does not demonstrate the utility of the potential PoS bound as discussed in Section 3.4.1. The reason is that there does not exist an exact potential function for the generic agent cost function.[1]

The next chapter will seek to design player cost functions by convexly combining multiple player cost functions, and will prove bounds on the resulting PoA and PoS of their respective games.

[1]We do remark that the PoS for reverse carpooling games that utilizes an agent cost function defined in [**?**] called the z cost function, can be found using the theorems introduced in Section 3.4.1. (Note that the z cost function in [**?**] is a translation of the α cost function originally presented in [12].)

Chapter 5

Convexly Combining Player Objective Functions

5.1 Outline

This chapter delves into the novel method of designing new player cost functions by convexly combining existing ones. In the following chapter, we will present an example game showing this method has the potential to result in an improved player objective function.

This chapter seeks to bound the resulting PoA for games with a player objective function defined from a convex combination. We focus on smooth games, and use the theorems presented in Chapter 3. While we present the ideas in context of games with a global cost function, we remark that parallel theorems exist for games with global welfare functions. With the goal of bounding the PoA in a game with a convexly combined cost function, we work sequentially: first we show that two smooth games with specific player objective functions can be convexly combined to produce a smooth game; then we show bounds on the resulting budget; following we show that artifacts from individual smooth games will continue existing in a convexly combined smooth game; lastly we show PoA bounds on the resulting smooth game. Although we don't make broad claims on the tightness of the resulting bounds for all smooth games, we will show in Chapter 6 that for some game types, the bounds in this Chapter are tight. Table 5.1 presents a tabulated organization of the theorems in this chapter, and summarizes the results that precede the main objective of this chapter; that is, Theorem 10 (and Corollary 2).

Table 5.1: Roadmap to Theorem 10

Theorem	Significance
Theorem 6	This shows that the convex combination of two smooth cost minimization games that are equal in all aspects except for their agent cost functions results in a game, G^*, that is also smooth.
Theorem 7	$B(G^*;z) \leq \frac{B(G)B(G')}{(1-z)B(G')+zB(G)}$
Theorem 8	$\overline{B}(G^*;z) \geq \frac{\overline{B}(G)\overline{B}(G')}{(1-z)\overline{B}(G')+z\overline{B}(G)}$
Theorem 9	$\lambda^{(CC)}B(G^*;z) > 0$ and $\mu^{(CC)}B(G^*;z) < 0$
Theorem 10	$PoA(G^*;z) \leq \frac{\lambda^{(CC)}(z)\cdot B(G^*;z)}{1-\mu^{(CC)}(z)\cdot B(G^*;z)}$
Corollary 2	$\frac{\lambda^{(CC)}(z)\cdot B(G^*;z)}{1-\mu^{(CC)}(z)\cdot B(G^*;z)} \geq \frac{\lambda^{(CC)}(z)\cdot\overline{B}(G)\overline{B}(G')}{((1-z)\overline{B}(G')+z\overline{B}(G))-\mu^{(CC)}(z)\cdot\overline{B}(G)\overline{B}(G')}$ $\frac{\lambda^{(CC)}(z)\cdot B(G^*;z)}{1-\mu^{(CC)}(z)\cdot B(G^*;z)} \leq \frac{\lambda^{(CC)}(z)\cdot B(G)B(G')}{((1-z)B(G')+zB(G))-\mu^{(CC)}(z)\cdot B(G)B(G')}$

It is important to recognize that the results from this chapter can be applied to an arbitrary number of player objective functions. For example, we can design a new player objective function from three player objective functions by first convexly combining two of them, and then convexly combining the result with the third; however, we focus on the foundational idea of combining two, and leave it to a designer to utilize the tools as fitting to their circumstances.

5.2 Convex Combinations of Player Objective Functions in Smooth Games

We begin by showing that a convex combination of any two player cost functions of two smooth games will result in a smooth game with convexly combined smoothness parameters.

Theorem 6. *Let two smooth games be $G = \{N, A, \{J_i\}_{i\in N}, C\}$ and $G' = \{N, A, \{J_i'\}_{i\in N}, C\}$, that are identical aside from their player cost functions. Suppose each game, G and G', is (λ, μ)-smooth and (λ', μ')-smooth. Then a convex combination of the player cost functions, $J_i^{(CC)} : A \to \mathcal{R}$, where*

$$J_i^{(CC)}(a;z) = (1-z)\cdot J_i(a) + z\cdot J_i'(a) \tag{5.1}$$

with $z \in [0,1]$, results in a $\big((1-z)\cdot\lambda + z\cdot\lambda', (1-z)\cdot\mu + z\cdot\mu'\big)$-smooth game, G^*.

Proof. Suppose there exist two smooth games, $G = \{N, A, \{J_i\}_{i\in N}, C\}$ and $G' = \{N, A, \{J_i'\}_{i\in N}, C\}$, that are identical aside from their (possibly) unique player cost functions, and that are (λ, μ)-smooth and (λ', μ')-smooth, respectively.

Consider a new player cost function for all $i \in N$, $J_i^{(CC)} : A \to \mathcal{R}; z$, that is a convex combination of the two player cost functions from each game:

$$J_i^{(CC)}(a; z) = (1 - z) \cdot J_i(a) + z \cdot J_i'(a) \tag{5.2}$$

for $z \in [0, 1]$. From the definition of a smooth game (Definition 7) we know that

$$\sum_{i \in N} J_i(a_i^*, a_{-i}) \leq \lambda \cdot C(a^*) + \mu \cdot C(a) \tag{5.3}$$

and

$$\sum_{i \in N} J_i'(a_i^*, a_{-i}) \leq \lambda' \cdot C(a^*) + \mu' \cdot C(a) \tag{5.4}$$

$\forall a, a^* \in \mathcal{A}$. Then it is trivially true that

$$(1 - z) \cdot \sum_{i \in N} J_i(a_i^*, a_{-i}) \leq (1 - z) \cdot \lambda \cdot C(a^*) + (1 - z) \cdot \mu \cdot C(a) \tag{5.5}$$

and

$$z \cdot \sum_{i \in N} J_i'(a_i^*, a_{-i}) \leq z \cdot \lambda' \cdot C(a^*) + z \cdot \mu' \cdot C(a) \tag{5.6}$$

$\forall a, a^* \in \mathcal{A}$. By combining Eqs. (5.2), (5.5), and (5.6), we have that

$$\begin{aligned} \sum_{i \in N} J_i^{(CC)}(a_i^*, a_{-i}; z) &= (1 - z) \cdot \sum_{i \in N} J_i(a_i^*, a_{-i}) + z \cdot \sum_{i \in N} J_i'(a_i^*, a_{-i}) \\ &\leq (1 - z) \cdot \lambda \cdot C(a^*) + (1 - z) \cdot \mu \cdot C(a) + z \cdot \lambda' \cdot C(a^*) + z \cdot \mu' \cdot C(a) \end{aligned} \tag{5.7}$$

$$\Rightarrow \sum_{i \in N} J_i^{(CC)}(a_i^*, a_{-i}; z) \leq (1 - z) \cdot \lambda \cdot C(a^*) + (1 - z) \cdot \mu \cdot C(a) + z \cdot \lambda' \cdot C(a^*) + z \cdot \mu' \cdot C(a) \tag{5.8}$$

$$\Rightarrow \sum_{i \in N} J_i^{(CC)}(a_i^*, a_{-i}; z) \leq \big((1 - z) \cdot \lambda + z \cdot \lambda'\big) \cdot C(a^*) + \big((1 - z) \cdot \mu + z \cdot \mu'\big) \cdot C(a) \tag{5.9}$$

By comparing Eq. (5.9) to Definition 7 and Eq. (3.1), we see that the resulting game is a $\big((1 - z) \cdot \lambda + z \cdot \lambda', (1 - z) \cdot \mu + z \cdot \mu'\big)$-smooth game, G^*. □

5.3 Budget Bounds for Convexly Designed Player Cost Functions

We now prove upper and lower bounds on the budget of a smooth game that results from a convex combination of two player cost functions in associated smooth games.

Theorem 7. *Consider two smooth games, $G = \{N, A, \{J_i\}_{i\in N}, C\}$ and $G' = \{N, A, \{J'_i\}_{i\in N}, C\}$, that are identical aside from their player cost functions. Suppose each game, G and G', is (λ, μ)-smooth and (λ', μ')-smooth with budget conditions, $B(G)$ and $B(G')$. Then a convex combination of these two player cost functions, $J_i^{(CC)} : A \to \mathcal{R}$, where*

$$J_i^{(CC)}(a; z) = (1 - z) \cdot J_i(a) + z \cdot J'_i(a) \tag{5.10}$$

for $z \in [0, 1]$, will produce a smooth game G^* with budget, $B(G^*; z)$, bounded above by

$$B(G^*; z) \leq \frac{B(G)B(G')}{(1 - z)B(G') + zB(G)}. \tag{5.11}$$

Proof. Suppose there exist two smooth games, $G = \{N, A, \{J_i\}_{i\in N}, C\}$ and $G' = \{N, A, \{J'_i\}_{i\in N}, C\}$, that are identical aside from their (possibly) unique player cost functions, and that are (λ, μ)-smooth and (λ', μ')-smooth, respectively. Suppose also that they have associated budget conditions, $B(G)$ and $B(G')$, respectively. Define a convex combination of these two player cost functions, $J_i^{(CC)} : A \to \mathcal{R}$, as

$$J_i^{(CC)}(a; z) = (1 - z) \cdot J_i(a) + z \cdot J'_i(a), \tag{5.12}$$

for $z \in [0, 1]$, and denote the resulting smooth game as G^*. Then the budget that we desire to bound above is $B(G^*; z)$. Trivially the theorem holds with equality for $z = 0$ and $z = 1$. Henceforth, let $z \in (0, 1)$. Without loss of generality, let actions $\widehat{a}$ and a' be the actions that define $B(G)$ and $B(G')$. That is,

$$B(G) = \max_{a\in A} \frac{C(a)}{\sum_{i\in N} J_i(a)} = \frac{C(\widehat{a})}{\sum_{i\in N} J_i(\widehat{a})} \tag{5.13}$$

and

$$B(G') = \max_{a \in A} \frac{C(a)}{\sum\limits_{i \in N} J_i(a)} = \frac{C(a')}{\sum\limits_{i \in N} J_i(a')}. \tag{5.14}$$

We now consider

$$B(G^*; z) = \max_{a \in A} \frac{C(a)}{\sum\limits_{i \in N} J_i(a)} = \frac{C(a^{(CC)})}{\sum\limits_{i \in N} J_i(a^{(CC)})} \tag{5.15}$$

where $a^{(CC)}$ may or may not be equal to $\widehat{a}$ or a'. By definition, we see that

$$B(G) = \frac{C(\widehat{a})}{\sum\limits_{i \in N} J_i(\widehat{a})} \geq \frac{C(a^{(CC)})}{\sum\limits_{i \in N} J_i(a^{(CC)})} \tag{5.16}$$

and

$$B(G') = \frac{C(a')}{\sum\limits_{i \in N} J'_i(a')} \geq \frac{C(a^{(CC)})}{\sum\limits_{i \in N} J'_i(a^{(CC)})} \tag{5.17}$$

That is to say, the action that defines the budget for each respective player cost function is not less than the action that defines the budget for the convexly defined player cost function. Looking at the inverses, we have that

$$\frac{\sum\limits_{i \in N} J_i(\widehat{a})}{C(\widehat{a})} \leq \frac{\sum\limits_{i \in N} J_i(a^{(CC)})}{C(a^{(CC)})} \tag{5.18}$$

and

$$\frac{\sum\limits_{i \in N} J'_i(a')}{C(a')} \leq \frac{\sum\limits_{i \in N} J'_i(a^{(CC)})}{C(a^{(CC)})} \tag{5.19}$$

Then trivially, the inequality of a convex combination of Eqs. (5.18) and (5.19) holds. That is, for $z \in (0, 1)$,

$$(1-z)\frac{\sum\limits_{i \in N} J_i(\widehat{a})}{C(\widehat{a})} + z\frac{\sum\limits_{i \in N} J'_i(a')}{C(a')} \leq (1-z)\frac{\sum\limits_{i \in N} J_i(a^{(CC)})}{C(a^{(CC)})} + z\frac{\sum\limits_{i \in N} J'_i(a^{(CC)})}{C(a^{(CC)})} \tag{5.20}$$

$$= \frac{(1-z)\sum\limits_{i \in N} J_i(a^{(CC)}) + z\sum\limits_{i \in N} J'_i(a^{(CC)})}{C(a^{(CC)})} \tag{5.21}$$

$$= \frac{1}{B(G^*; z)} \tag{5.22}$$

$$\Rightarrow (1-z)\frac{1}{B} + z\frac{1}{B'} \leq \frac{1}{B(G^*; z)} \tag{5.23}$$

$$\Rightarrow B(G^*; z) \leq \frac{B(G)B(G')}{(1-z)B(G') + zB(G)}. \tag{5.24}$$

□

The significance of Theorem 7 is that it bounds the resulting budget of a smooth game that results from a convex combination of two player cost functions. In the next section, this theorem will aid in bounding the PoA for this resulting smooth game. As an example illustrating this upper bound on $B(G^*; z)$ is shown in Fig. 5.1 for $B(G) = 1$ and $B(G') = 10$.

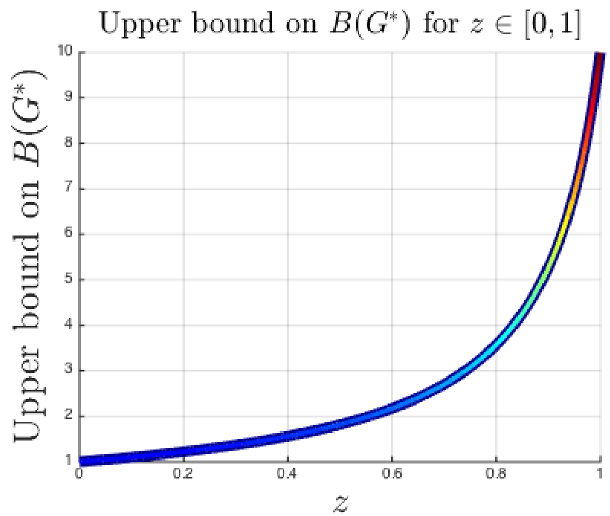


Figure 5.1: An example scenario of the upper bound afforded by Theorem 7 showing the upper bound on $B(G^*; z)$ for when $B(G) = 1$ and $B(G') = 10$.

A similar theorem to Theorem 7 defines a lower bound on $B(G^*; z)$:

Theorem 8. *Consider two smooth games,* $G = \{N, A, \{J_i\}_{i \in N}, C\}$ *and* $G' = \{N, A, \{J'_i\}_{i \in N}, C\}$*, that are identical aside from their player cost functions. Suppose each game,* G *and* G'*, is* (λ, μ)*-smooth and* (λ', μ')*-smooth and have associated lower budget conditions,* $\overline{B}(G)$ *and* $\overline{B}(G')$*. Then a convex combination of these two player cost functions,* $J_i^{(CC)} : A \to \mathcal{R}$*, where*

$$J_i^{(CC)}(a; z) = (1 - z) \cdot J_i(a) + z \cdot J'_i(a), \tag{5.25}$$

for $z \in [0, 1]$*, will produce a smooth game,* G^**, with a lower budget,* $\overline{B}(G^*; z)$*, bounded below by*

$$\overline{B}(G^*; z) \geq \frac{\overline{B}(G)\overline{B}(G')}{(1 - z)\overline{B}(G') + z\overline{B}(G)}. \tag{5.26}$$

Proof. Suppose there exist two smooth games, $G = \{N, A, \{J_i\}_{i\in N}, C\}$ and $G' = \{N, A, \{J'_i\}_{i\in N}, C\}$, that are identical aside from their (possibly) unique player cost functions, and that are (λ, μ)-smooth and (λ', μ')-smooth, respectively. Suppose also that they have associated lower budget conditions, $\overline{B}(G)$ and $\overline{B}(G')$, respectively. Define a convex combination of these two player cost functions, $J_i^{(CC)} : A \to \mathcal{R}$, as

$$J_i^{(CC)}(a; z) = (1 - z) \cdot J_i(a) + z \cdot J'_i(a), \tag{5.27}$$

for $z \in [0, 1]$. Denote the budget of $J_i^{(CC)}$ that we desire to bound below as $\overline{B}(G^*; z)$. Trivially the theorem holds with equality for $z = 0$ and $z = 1$. Henceforth, let $z \in (0, 1)$. Without loss of generality, let actions $\widehat{a}$ and a' be the actions that define $\overline{B}(G)$ and $\overline{B}(G')$. That is,

$$\overline{B}(G) = \min_{a\in A} \frac{C(a)}{\sum\limits_{i\in N} J_i(a)} = \frac{C(\widehat{a})}{\sum\limits_{i\in N} J_i(\widehat{a})} \tag{5.28}$$

and

$$\overline{B}(G') = \min_{a\in A} \frac{C(a)}{\sum\limits_{i\in N} J_i(a)} = \frac{C(a')}{\sum\limits_{i\in N} J_i(a')}. \tag{5.29}$$

We now consider

$$\overline{B}(G^*; z) = \min_{a\in A} \frac{C(a)}{\sum\limits_{i\in N} J_i(a)} = \frac{C(a^{(CC)})}{\sum\limits_{i\in N} J_i(a^{(CC)})} \tag{5.30}$$

where $a^{(CC)}$ may or may not be equal to $\widehat{a}$ or a'. By definition, we see that

$$\overline{B}(G) = \frac{C(\widehat{a})}{\sum\limits_{i\in N} J_i(\widehat{a})} \le \frac{C(a^{(CC)})}{\sum\limits_{i\in N} J_i(a^{(CC)})} \tag{5.31}$$

and

$$\overline{B}(G') = \frac{C(a')}{\sum\limits_{i\in N} J'_i(a')} \le \frac{C(a^{(CC)})}{\sum\limits_{i\in N} J'_i(a^{(CC)})} \tag{5.32}$$

That is to say, the action that defines the minimum budget for each respective player cost function is not more than the action that defines the budget for the convex combination of the two. Looking at the inverse, we have that

$$\frac{\sum\limits_{i\in N} J_i(\widehat{a})}{C(\widehat{a})} \ge \frac{\sum\limits_{i\in N} J_i(a^{(CC)})}{C(a^{(CC)})} \tag{5.33}$$

and

$$\frac{\sum_{i\in N} J_i'(a')}{C(a')} \geq \frac{\sum_{i\in N} J_i'(a^{(CC)})}{C(a^{(CC)})} \tag{5.34}$$

Then trivially, the inequality of a convex combination of Eqs. (5.33) and (5.34) holds. That is, for $z \in (0,1)$,

$$(1-z)\frac{\sum_{i\in N} J_i(\widehat{a})}{C(\widehat{a})} + z\frac{\sum_{i\in N} J_i'(a')}{C(a')} \geq (1-z)\frac{\sum_{i\in N} J_i(a^{(CC)})}{C(a^{(CC)})} + z\frac{\sum_{i\in N} J_i'(a^{(CC)})}{C(a^{(CC)})} \tag{5.35}$$

$$= \frac{(1-z)\sum_{i\in N} J_i(a^{(CC)}) + z\sum_{i\in N} J_i'(a^{(CC)})}{C(a^{(CC)})} \tag{5.36}$$

$$= \frac{1}{\overline{B}(G^*; z)} \tag{5.37}$$

$$\Rightarrow (1-z)\frac{1}{\overline{B}(G)} + z\frac{1}{\overline{B}(G')} \geq \frac{1}{\overline{B}(G^*; z)} \tag{5.38}$$

$$\Rightarrow \overline{B}(G^*; z) \geq \frac{\overline{B}(G)\overline{B}(G')}{(1-z)\overline{B}(G') + z\overline{B}(G)}. \tag{5.39}$$

□

The importance of Theorem 8 is this: it limits the lowest upper bound provable for a smooth game that results from a convexly designed player cost function.

5.4 Smooth Game PoA Upper Bound of a Convex Design

The following theorem will help in establishing a bound on the PoA for a game resulting from a convexly designed player cost function; theorem 7 will be used to prove it.

Theorem 9. *Consider two smooth games, $G = \{N, A, \{J_i\}_{i\in N}, C\}$ and $G' = \{N, A, \{J_i'\}_{i\in N}, C\}$, that are identical aside from their player cost functions. Suppose each game, G and G', is (λ,μ)-smooth and (λ',μ')-smooth with budget conditions, $B(G)$ and $B(G')$, such that $\lambda B(G) > 0$ and $\mu B(G) < 1$, and $\lambda' B(G') > 0$ and $\mu' B(G') < 1$. Then a player cost function that is a convex combination of that from each respective game will result in a $(\lambda^{(CC)}, \mu^{(CC)})$-smooth game, G^*, with a budget, $B(G^*; z)$, where $\lambda^{(CC)} B(G^*; z) > 0$ and $\mu^{(CC)} B(G^*; z) < 1$.*

Proof. Suppose there exist two smooth games, $G = \{N, A, \{J_i\}_{i \in N}, C\}$ and $G' = \{N, A, \{J_i'\}_{i \in N}, C\}$, that are identical aside from their (possibly) unique player cost functions, and that are (λ, μ)-smooth and (λ', μ')-smooth, respectively. Suppose also that they have associated budget conditions, $B(G)$ and $B(G')$, such that $\lambda B(G) > 0$ and $\mu B(G) < 1$, and $\lambda' B(G') > 0$ and $\mu' B(G') < 1$.

Consider the player cost function, $J_i^{(CC)} : A \to \mathcal{R}$, that is a convex combination of the two player cost functions from each game; that is,

$$J_i^{(CC)}(a; z) = (1 - z) \cdot J_i(a) + z \cdot J_i'(a) \tag{5.40}$$

for $z \in [0, 1]$. From Theorem 6, we have that a player cost function which is a convex combination of that from each respective game will result in a $(\lambda^{(CC)}, \mu^{(CC)})$-smooth game, where

$$\lambda^{(CC)}(z) = (1 - z) \cdot \lambda + z \cdot \lambda' \tag{5.41}$$

$$\mu^{(CC)}(z) = (1 - z) \cdot \mu + z \cdot \mu'. \tag{5.42}$$

From remarks in Section 3.2.2, we note that a budget $B(G^*; z)$ is strictly greater than zero for all $z \in [0, 1]$. When looking at $z = 0$ and $z = 1$, we see that since $\lambda B(G) > 0$ and $\lambda' B(G') > 0$, therefore $\lambda > 0$ and $\lambda' > 0$. Hence, for $z \in [0, 1]$, $(1-z) \cdot \lambda + z \cdot \lambda' > 0$. Therefore, $\lambda^{(CC)} B(G^*; z) > 0$.

From Theorem 7, we have that

$$B(G^*; z) \leq \frac{B(G)B(G')}{(1 - z)B(G') + zB(G)}. \tag{5.43}$$

Finding an upper bound on $\mu^{(CC)}B(G^*;z)$ we have:

$$\mu^{(CC)}B(G^*;z) \leq \mu^{(CC)}\left(\frac{B(G)B(G')}{(1-z)B(G')+zB(G)}\right) \tag{5.44}$$

$$= \left((1-z)\cdot\mu + z\cdot\mu'\right)\left(\frac{B(G)B(G')}{(1-z)B(G')+zB(G)}\right) \tag{5.45}$$

$$= \left(\frac{(1-z)\cdot\mu B(G)B(G') + z\cdot\mu' B(G)B(G')}{(1-z)B(G')+zB(G)}\right) \tag{5.46}$$

$$= \left(\frac{(\mu B)\cdot(1-z)B(G') + \mu'\,(B')\cdot zB(G)}{(1-z)B(G')+zB(G)}\right) \tag{5.47}$$

$$< \left(\frac{1\cdot(1-z)B(G') + 1\cdot zB(G)}{(1-z)B(G')+zB(G)}\right) \tag{5.48}$$

$$= \left(\frac{(1-z)B(G')+zB(G)}{(1-z)B(G')+zB(G)}\right) \tag{5.49}$$

$$= 1 \tag{5.50}$$

$$\Rightarrow \mu^{(CC)}B(G^*;z) < 1. \tag{5.51}$$

□

Theorem 10. *Consider two smooth games, $G = \{N, A, \{J_i\}_{i\in N}, C\}$ and $G' = \{N, A, \{J_i'\}_{i\in N}, C\}$, that are identical aside from their player cost functions. Suppose each game, G and G', is (λ,μ)-smooth and (λ',μ')-smooth with budgets $B(G)$ and $B(G')$, such that $\lambda B(G) > 0$ and $\mu B(G) < 1$, and $\lambda' B(G') > 0$ and $\mu' B(G') < 1$. Then a player cost function that is a convex combination of that from each respective game will result in a $(\lambda^{(CC)}, \mu^{(CC)})$-smooth game, G^* with a budget, $B(G^*;z)$, where $PoA(G^*;z)$ for all equilibria types is bound above by*

$$PoA(G^*;z) \leq \frac{\lambda^{(CC)}(z)\cdot B(G^*;z)}{1-\mu^{(CC)}(z)\cdot B(G^*;z)} \tag{5.52}$$

for all $z \in [0,1]$, where

$$\lambda^{(CC)}(z) = (1-z)\cdot\lambda + z\cdot\lambda' \tag{5.53}$$

$$\mu^{(CC)}(z) = (1-z)\cdot\mu + z\cdot\mu'. \tag{5.54}$$

Proof. The proof follows by combining Theorems 2, 6, and 9. □

Corollary 2. *The right side of the inequality of Eq. (5.52) in Theorem 10 is lower bounded such that*

$$\frac{\lambda^{(CC)}(z)\cdot B(G^*;z)}{1-\mu^{(CC)}(z)\cdot B(G^*;z)} \geq \frac{\lambda^{(CC)}(z)\cdot \frac{\overline{B}(G)\overline{B}(G')}{(1-z)\overline{B}(G')+z\overline{B}(G)}}{1-\mu^{(CC)}(z)\cdot \frac{\overline{B}(G)\overline{B}(G')}{(1-z)\overline{B}(G')+z\overline{B}(G)}} \tag{5.55}$$

$$= \frac{\lambda^{(CC)}(z)\cdot \overline{B}(G)\overline{B}(G')}{((1-z)\overline{B}(G')+z\overline{B}(G))-\mu^{(CC)}(z)\cdot \overline{B}(G)\overline{B}(G')} \tag{5.56}$$

and upper bounded such that

$$\frac{\lambda^{(CC)}(z)\cdot B(G^*;z)}{1-\mu^{(CC)}(z)\cdot B(G^*;z)} \leq \frac{\lambda^{(CC)}(z)\cdot \frac{B(G)B(G')}{(1-z)B(G')+zB(G)}}{1-\mu^{(CC)}(z)\cdot \frac{B(G)B(G')}{(1-z)B(G')+zB(G)}} \tag{5.57}$$

$$= \frac{\lambda^{(CC)}(z)\cdot B(G)B(G')}{((1-z)B(G')+zB(G))-\mu^{(CC)}(z)\cdot B(G)B(G')} \tag{5.58}$$

for all $z \in [0,1]$, *where*

$$\lambda^{(CC)}(z) = (1-z)\cdot\lambda + z\cdot\lambda' \tag{5.59}$$

$$\mu^{(CC)}(z) = (1-z)\cdot\mu + z\cdot\mu'. \tag{5.60}$$

Proof. The proof for the lower bound presented in Eq. (5.56) follows by combining Theorem 8 with Theorem 10. The proof for the upper bound presented in Eq. (5.58) follows by combining Theorem 7 with Theorem 10. □

The significance of Theorem 10 and Corollary 2 is that the smooth game which results from the convex combination of two player cost functions of two separate smooth games has a PoA that is non-trivially bounded for all equilibria types.

5.5 Summary

In this chapter, we considered the method of designing player cost functions by convexly combining existing functions for their respective smooth games, and showed an upper bound on the PoA of the resulting game. We showed the following properties:

- the resulting game is also smooth, and has smoothness parameters that are a convex combination of the smoothness parameters of each respective game (Theorem 6);

- the (upper) budget of the resulting game is bounded above by Eq. (5.11) (Theorem 7);

- the lower budget of the resulting game is bounded below by Eq. (5.26) (Theorem 8);

- if each respective (λ, μ)-smooth game with budget B satisfies $\lambda > 0$ and $\mu B < 1$, then the $(\lambda^{(CC)}, \mu^{(CC)})$-smooth game with a budget $B(G^*; z)$ will also satisfy $\lambda^{(CC)} > 0$ and $\mu^{(CC)} B^{CC} < 1$ (Theorem 9);

- given the previous facts, the resulting smooth game will have a PoA that is upper bounded by Eqs. (5.52) and (5.58) (Theorem 10 and Corollary 2), and that has a PoA bound that is not less than Eq. (5.56) (Corollary 2).

The following chapter will look at a specific resource allocation game, and will heavily focus on player cost functions that are designed using these convex combination tools presented in this chapter, and using the PoA upper bounds devised in Chapter 3.[1]

[1] As mentioned in Section 5.1, similar proofs (omitted) exist for smooth games with global welfare functions and player utility functions. The omissions are made to simplify the presented content, as the parallel theorems are proven equally.

Chapter 6

A More Specific Resource Allocation Game

6.1 Outline

In this chapter, we introduce a specific type of cost minimization game called a resource allocation game, and show how the results from Chapters 3 and 5 can aid in designing and assessing agent cost functions for this game. Additionally, we demonstrate how a smoothness argument may not always afford a tight PoA bound.

Resource allocation games are finite games where it is desired to allocate some set of resources among a group of agents. The UAV example in Section 1.1 is an example of such a game, as the UAV agents were selecting from a land space to search over to find a lost hiker.

We begin by introducing a specific resource allocation game, coined *generic allocation games*, that will be the focus of this chapter. We follow by introducing four player cost functions used in this chapter, and show that the game is smooth for all four. Then we upper bound the PoA implied for these player cost functions, and show how smoothness does not always tightly upper bound the PoA. We also look at the tight PoS upper bounds for these cost functions, and prove that one of the functions is optimal. We end with concluding remarks and remaining research questions. Although we focus on games with a global cost function and player cost functions, the results of this chapter can easily be re-worked in terms of a global welfare function with only slight modifications to the game structure (*see* Section 6.11).

6.2 Generic Allocation Games

We define the class of generic allocation games as follows:

Definition 15 (Generic Allocation Games)**.** *A class of generic allocation game has the following components:*

- *a set of agents* $\mathcal{N} = \{1, \ldots, m\}$
- *a set of resources* $R = \{r_1, \ldots, r_l\}$*, each with assigned value* $\beta_r \in (0, 1]$
- *a collection of action sets* $\mathcal{A}_i$ *where* $\mathcal{A}_i = \{A'_i, A''_i, \ldots\}$ *for each agent* $i \in \mathcal{N}$ [1]
  - ∗ *a collection of joint action sets is* $\mathcal{A} = \{A^{(1)}, A^{(2)}, \ldots\}$ *where* $A = \Pi_{i \in \mathcal{N}} A_i$ *for any joint action set.*
  - ∗ *an action profile is* $a = \{a_1, \ldots, a_m\} \in A \subseteq R$
- *a set of* ***local***[2] *agent cost functions,* $J_i : A \to \mathcal{R}$*, for all* $i \in \mathcal{N}$ *and for all* $A \in \mathcal{A}$ *defined as*

$$J_i(a; d) = \sum_{r \in a_i} \beta_r \cdot f(i, |a|_r; d) \tag{6.1}$$

  where

  - ∗ *the agent function is* ***scalable****; that is, the agent function does not change given the set of agents in the system, and therefore for any* $N \subseteq \mathcal{N}$ *satisfies:*

$$J_i(a; d, N) = \sum_{r \in a_i} \beta_r \cdot f(i, |a|_r; d, N) = \sum_{r \in a_i} \beta_r \cdot f(i, |a|_r; d, N') = J_i(a; d, N') \tag{6.2}$$

[1] The existence of these subsets is implicit.

[2] By *local* we mean that the cost is dependent only on the resource value and the number of players selecting that resource.

* $*$ *constant* $d \in [0, 1)$,[1]

* $*$ $|a|_r$ *denotes the number of players that selected resource* r *for their action, i.e.,*

$$|a|_r = I\{i \in \mathcal{N} : a_i = r\}, \tag{6.3}$$

* $*$ $f(i, |a|_r; d) = 1$ *when* $|a|_r = 1$

- *local*[2] *cost functions,* $C_r : \mathcal{A} \to \mathcal{R}$*, for all* $r \in R$ *defined by*

$$C_r(|a|_r; d) = \beta_r \cdot |a|_r^d \tag{6.4}$$

that are submodular[3]

- *global cost function,* $C : A \to \mathcal{R}$ *for all* $A \in \mathcal{A}$*, which is the sum of the separate local cost functions over all resources:*

$$C(a; d) = \sum_{r \in R} C_r(|a|_r; d) \tag{6.5}$$

The class of generic allocation games is denoted as $\mathcal{G} = \{\mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i\}_{i \in N}, R, \{C_r\}_{r \in R}\}$*. An* N*-family of generic allocation games (sometimes simply referred to as a 'game') is denoted as* $G = \{\{1, \ldots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i\}_{i \in N}, R, \{C_r\}_{r \in R}\}$.[4]

6.3 Agent Cost Functions

We now discuss the agent cost functions of interest in this chapter. Table 6.1 summarizes the four cost functions that will be discussed.

---

[1](1) We note that the trivial case is when $d = 1$ for this family of games; (2) We assume the constant to be implicitly assumed when referring to any game.

[2]By *local* we mean that the function depends only on the resource value and the number of players selecting it.

[3]By *submodular* we mean the local cost function satisfies $C_r(|S|)+C_r(|T|) \leq C_r(|S|\cup|T|)+C_r(|S|\cap|T|)\ \forall S, T \subseteq R$.

[4]Although unused in this chapter, a generic allocation game can be denoted as $\widehat{G} = \{\{1, \ldots, n\} = N \subseteq \mathcal{N}, \{A_i \in \mathcal{A}_i\}_{i \in N}, \{J_i\}_{i \in N}, \widehat{R} \subseteq R, \{C_r\}_{r \in \widehat{R}}\} \in G$

Table 6.1: Agent cost functions discussed for the generic allocation game.

Name	Function
Fixed	$J_i^{(F)}(a;d) = \sum_{r \in a_i} C_r(\lvert(a_i,\varnothing)\rvert_r; d)$
Marginal Contribution	$J_i^{(MC)}(a;d) = \sum_{r \in a_i} (C_r(\lvert a\rvert_r; d) - C_r(\lvert(\varnothing, a_{-i})\rvert_r; d))$
z	$J_i^{(z)}(a_i, a_{-i}; d, z) = \sum_{r \in a_i} [(1-z) \cdot C_r(a_i, \varnothing) + z \cdot (C_r(\lvert a\rvert_r; d) - C_r(a \backslash \{i\}))]$
Shapley	$J_i^{(S)}(a_i, a_{-i}; d) = \sum_{r \in a_i} \frac{C_R(\lvert a\rvert_r; d)}{\lvert a\rvert_r}$

6.3.1 Fixed Cost Function

Let the fixed (F) agent cost function, $J_i^{(F)} : \mathcal{A} \to \mathcal{R}$, be defined as

$$J_i^{(F)}(a_i, a_{-i}; d) = \sum_{r \in a_i} \beta_r \cdot f^{(F)}(i, |a|_r; d) \tag{6.6}$$

$$= \sum_{r \in a_i} (C_r(|(a_i, \varnothing)|_r; d) - C_r(|(\varnothing)|; d))$$

$$= \sum_{r \in a_i} C_r(|(a_i, \varnothing)|_r; d) \tag{6.7}$$

for player $i \in \mathcal{N}$, where $C_r(|(a_i, \varnothing)|_r; d)$ represents the local cost of resource r when only player i selects r (*i.e.* $C_r(|(a_i, \varnothing)|_r; d) = \beta_r$), and where $C_r(|(\varnothing)|; d)$ is the local cost when no agent selects resource r (*i.e.* $C_r(|(\varnothing)|; d) = 0$). In words, this agent function charges players the cost of the resource they select in a manner that is independent of the actions of the other players. Here we note that

$$f^{(F)}(i, k; d) = 1 \tag{6.8}$$

for all $k \in N$.

6.3.2 Marginal Contribution Cost Function

Let the marginal contribution (MC) agent cost function, $J_i^{(MC)} : A \to \mathcal{R}$, be defined as

$$J_i^{(MC)}(a_i, a_{-i}; d) = \sum_{r \in a_i} \left(C_r(|a|_r; d) - C_r(|(\varnothing, a_{-i})|_r; d)\right) \tag{6.9}$$

for any agent $i \in \mathcal{N}$, where $C_r(|(\varnothing, a_{-i})|_r)$ local cost of resource r when player i does *not* select resource r. This agent cost function is contingent on the actions of other players.

6.3.3 The z Cost Function

Let the z player cost function, $J_i^{(z)} : A \to \mathcal{R}$, be a convex combination of the fixed and the MC cost functions. That is,

$$\begin{aligned} J_i^{(z)}(a_i, a_{-i}; d, z) &= (1 - z) \cdot J_i^{(F)}(a_i, a_{-i}; d, z) + z \cdot J_i^{(MC)}(a_i, a_{-i}; d, z) \\ &= \sum_{r \in a_i} \big((1 - z) \cdot C_r(|(a_i, \varnothing)|_r; d) + z \cdot (C_r(|a|_r; d) - C_r(|(\varnothing, a_{-i})|_r; d))\big) \end{aligned} \tag{6.10}$$

for $z \in [0, 1]$. This cost function will be used to motivate many results from Chapter 5.

6.3.4 The Shapley Cost Function

Let the Shapley (S) player cost function, $J_i^{(S)} : A \to \mathcal{R}$, be defined as

$$J_i^{(S)}(a_i, a_{-i}; d) = \sum_{r \in a_i} \frac{C_r(|a|_r; d)}{|a|_r} \tag{6.11}$$

for player i. That is, the local cost of each resource r is split equally among the players that select that resource.

6.4 Proving Smoothness for Generic Allocation Games

We now prove that every generic allocation games that utilizes one of the player cost functions presented in Section 6.3 (or any convex combination of these, by Theorem 6) are all smooth games with identical smoothness parameters. We note that the proofs for each are similar. Table 6.2 summarizes the smoothness parameters proven in this section.

Table 6.2: Smoothness parameters for an N-family generic allocation game, $G = \{\{1,\ldots,n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i\}_{i\in N}, R, \{C_r\}_{r\in R}\}$.

Agent Cost Function	**Smoothness Parameters** (λ, μ)	**Theorem**
Fixed	$(n^{1-d}, 0)$	Theorem 11
Marginal Contribution	$(n^{1-d}, 0)$	Theorem 12
z	$(n^{1-d}, 0)$	Theorem 13
Shapley	$(n^{1-d}, 0)$	Theorem 14

6.4.1 Smoothness Proof for the Fixed Player Cost Function

Theorem 11. *Let an N-family generic allocation game be $G = \{\{1,\ldots,n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i^{(F)}\}_{i\in N}, R, \{C_r\}_{r\in R}\}$. Then game G is $(n^{1-d}, 0)$-smooth.*

Proof. Assume the N-family generic allocation game G. Consider any joint action profiles $a = (a_i, a_{-i}), a^* = (a_i^*, a_{-i}^*) \in A$ for any $A \in \mathcal{A}$. A summation over each player for playing action a_i^* when all other players play their action in action profile a_{-i} is

$$\sum_{i\in N} J_i^{(F)}(a_i^*, a_{-i}; d) = \sum_{i\in N}\sum_{r\in a_i^*} \left[C_r(|(a_i^*, \varnothing)|_r; d) - C_r(|(\varnothing)|_r; d)\right] \tag{6.12}$$

$$= \sum_{i\in N}\sum_{r\in a_i^*} C_r(|(a_i^*, \varnothing)|_r; d) \tag{6.13}$$

$$= \sum_{i\in N}\sum_{r\in a_i^*} \beta_r \tag{6.14}$$

$$= \sum_{r\in R} \beta_r |a^*|_r \tag{6.15}$$

$$= \sum_{r\in R} \beta_r |a^*|_r^d \cdot |a^*|_r^{1-d} \tag{6.16}$$

$$\leq n^{1-d} \cdot \sum_{r\in R} \beta_r |a^*|_r^d \tag{6.17}$$

$$= n^{1-d} \cdot C(a^*; d) \tag{6.18}$$

Comparing this to Eq. (3.1), we see that the generic allocation game G is $(n^{1-d}, 0)$-smooth. □

6.4.2 Smoothness Proof for the MC Player Cost Function

Theorem 12. *Let an N-family generic allocation game be* $G = \{\{1,\ldots,n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i^{(MC)}\}_{i\in N}, R, \{C_r\}_{r\in R}\}$. *Then game G is $(n^{1-d}, 0)$-smooth.*

Proof. Assume the N-family generic allocation game G. Consider any joint action profiles $a = (a_i, a_{-i}), a^* = (a_i^*, a_{-i}^*) \in A$ for any $A \in \mathcal{A}$. A summation over each player for playing action a_i^* when all other players play their action in action profile a_{-i} is

$$\sum_{i\in N} J_i^{(MC)}(a_i^*, a_{-i}; d) = \sum_{i\in N}\sum_{r\in a_i^*} \left(C_r(|(a_i^*, a_{-i})|_r; d) - C_r(|(\varnothing, a_{-i})|_r; d)\right) \tag{6.19}$$

$$= \sum_{i\in N}\sum_{r\in a_i^*} \left(\beta_r |(a_i^*, a_{-i})|_r^d - \beta_r(|(a_i^*, a_{-i})|_r - 1)^d\right) \tag{6.20}$$

$$= \sum_{i\in N}\sum_{r\in a_i^*} \beta_r \left(|(a_i^*, a_{-i})|_r^d - (|(a_i^*, a_{-i})|_r - 1)^d\right) \tag{6.21}$$

$$\leq \sum_{i\in N}\sum_{r\in a_i^*} \beta_r \tag{6.22}$$

$$= n^{1-d} \cdot C(a^*; d) \tag{6.23}$$

where Eq. (6.22) follows since $|\hat{a}|_r^d - (|\hat{a}|_r - 1)^d \leq 1$ for $d \in [0,1]$ and for any $\hat{a} \in \Pi_{i\in N} A_i$ such that $|\hat{a}|_r \geq 1$. Eq. (6.23) follows using intermediate steps already presented in Eqs. (6.15 – 6.18). Comparing this to Eq. (3.1), we see that the generic allocation game G is $(n^{1-d}, 0)$-smooth. □

6.4.3 Smoothness Proof for the z Player Cost Function

Theorem 13. *Let an N-family generic allocation game be* $G = \{\{1,\ldots,n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i^{(z)}\}_{i\in N}, R, \{C_r\}_{r\in R}\}$. *Then game G is $(n^{1-d}, 0)$-smooth.*

Proof. The proof follows by applying Theorem 6 from Chapter 5. □

Before we show the game with the Shapley cost function is smooth, we will first prove that the smoothness parameters defined in Theorem 12 are the lowest that can be afforded.

Corollary 3. *Let an N family generic allocation game be*
$G = \{\{1, \ldots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i^{(z)}\}_{i\in N}, R, \{C_r\}_{r\in R}\}$. *Then the smoothness parameters afforded by Theorem 13, are the minimum smoothness values provable.*

Proof. Assume the N-family generic allocation game G. Suppose the action set available to every agent $i \in N$ is $A_i = \{i, n+1\}$. That is, every agent has access to two resources, one of which can be selected by any player, and one that is selectable only by themselves. Define action a to be the action profile where all agents select their unique resource, i. Define action a^* to be the joint action profile when all players select resource $n+1$.

- We begin by showing that the λ smoothness parameter must be at least n^{1-d}. Suppose that all resources $\{1, 2, \ldots, n\}$ have values ε where $\varepsilon > 0$ and $\varepsilon \to 0$. Let the cost of resource $n+1$ be one. Fig. 6.1 depicts this scenario.

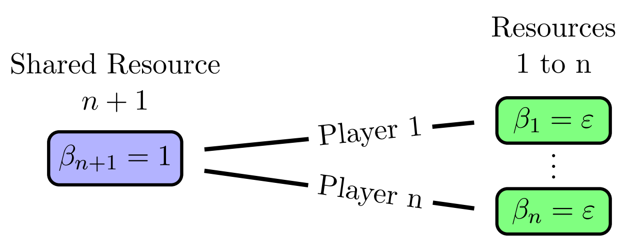


Figure 6.1: Action profile a is when all players select the **green** resources. Action profile a^* is when all players select the shared **blue** resource.

We first calculate the left side of Eq. (3.1):

$$\sum_{i\in N} J_i^{(z)}(a_i^*, a_{-i}; d, z) = \sum_{i\in N} \beta_{n+1} \cdot ((1-z)+z) = n \tag{6.24}$$

Then we calculate $C(a^*; d)$:

$$C(a^*; d) = \sum_{r \in R} \beta_r \cdot |a^*|_r^d \tag{6.25}$$

$$= n^d \tag{6.26}$$

Lastly, we calculate $C(a; d)$:

$$C(a; d) = \sum_{r \in R} \beta_r \cdot |a|_r^d \tag{6.27}$$

$$= \sum_{i \in N} \beta_i \tag{6.28}$$

$$= n \cdot \varepsilon \to 0 \tag{6.29}$$

Hence by smoothness,

$$\sum_{i \in N} J_i^{(z)}(a_i^*, a_{-i}; d, z) \leq \lambda C(a^*; d) + \mu C(a; d) \tag{6.30}$$

$$\Rightarrow n \leq \lambda \cdot n^d \tag{6.31}$$

$$\Rightarrow \lambda \geq n^{1-d} \tag{6.32}$$

- Similarly we show that the μ smoothness parameter must be at least zero. Consider the same two action profiles, a and a^*. Suppose now that all resources $\{1, 2, \ldots, n\}$ have value one, and let the value of resource $n+1$ be ε where $\varepsilon > 0$ and $\varepsilon \to 0$. Fig. 6.2 depicts this scenario.

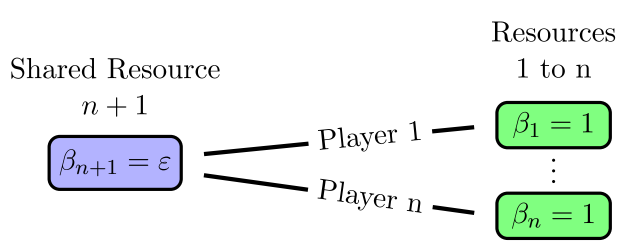


Figure 6.2: Action profile a is when all players select the **green** resources. Action profile a^* is when all players select the shared **blue** resource.

We first calculate the left side of Eq. (3.1):

$$\sum_{i \in N} J_i^{(z)}(a_i^*, a_{-i}; d, z) = \sum_{i \in N} \beta_{n+1} \cdot ((1-z)+z) \tag{6.33}$$

$$= 0 \tag{6.34}$$

Calculating $C(a^*; d)$ we have:

$$C(a^*; d) = \sum_{r \in R} \beta_{n+1} \cdot k_{n+1}^d \tag{6.35}$$

$$= n \cdot \varepsilon \to 0 \tag{6.36}$$

Calculating $C(a; d)$ we have:

$$C(a; d) = \sum_{i \in N} \beta_i = n \tag{6.37}$$

Hence by smoothness,

$$\sum_{i \in N} J_i^{(z)}(a_i^*, a_{-i}; d, z) \leq \lambda C(a^*; d) + \mu C(a; d) \tag{6.38}$$

$$\Rightarrow 0 \leq \lambda \cdot 0 + \mu \cdot n \tag{6.39}$$

$$\Rightarrow \mu \geq 0 \tag{6.40}$$

□

6.4.4 Smoothness Proof for the Shapley Agent Cost Function

Theorem 14. *Let an N-family generic allocation game be $G = \{\{1, \ldots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i^{(S)}\}_{i \in N}, R, \{C_r\}_{r \in R}\}$. Then game G is $(n^{1-d}, 0)$-smooth.*

Proof. Assume the N-family generic allocation game G. Consider any joint action profiles $a = (a_i, a_{-i}), a^* = (a_i^*, a_{-i}^*) \in A$ for any $A \in \mathcal{A}$. A summation over each player for playing action a_i^*

when all other players play their action in action profile a_{-i} is

$$\sum_{i \in N} J_i^{(S)}(a_i^*, a_{-i}; d) = \sum_{i \in N} \sum_{r \in a_i^*} \frac{C_R(|a|_r; d)}{|a|_r} \tag{6.41}$$

$$= \sum_{i \in N} \sum_{r \in a_i^*} \frac{\beta_r |a|_r^d}{|a|_r} \tag{6.42}$$

$$\leq \sum_{i \in N} \sum_{r \in a_i^*} \beta_r \tag{6.43}$$

$$= n^{1-d} \cdot C(a^*; d) \tag{6.44}$$

where Eq. (6.43) follows since $\frac{|a|_r^d}{|a|_r} \leq 1$ for $d \in [0, 1)$ and $|a|_r \geq 1$, and where Eq. (6.44) follows using intermediate steps already presented in Eqs. (6.15 – 6.18). Comparing this to Eq. (3.1), we see that the generic allocation game G is $(n^{1-d}, 0)$-smooth. □

6.5 Proving Budget Bounds for Generic Allocation Games

We now prove the budget condition for generic allocation games that utilize each of the player cost functions presented in Section 6.3. We note that the proofs for each are similar. Table 6.3 summarizes the budget bounds proven in this section.

Table 6.3: The budget conditions for an N-family generic allocation game, $G = \{\{1, \dots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i\}_{i \in N}, R, \{C_r\}_{r \in R}\}$.

Agent Cost Function	**Budget (Upper Bound)**	**Theorem**
Fixed	1	Theorem 15
Marginal Contribution	$\frac{1}{n^{1-d} \cdot (n^d - (n-1)^d)}$	Theorem 16
z	$\max_{\hat{n}: \hat{n} \leq n} \frac{1}{\hat{n}^{1-d}(\hat{n}^d - (\hat{n}-1)^d)}$	Theorem 17
Shapley	1	Theorem 18

6.5.1 Budget Proof for the Fixed Player Cost Function

Theorem 15. *Let an N-family generic allocation game be*

$G = \{\{1, \dots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i^{(F)}\}_{i \in N}, R, \{C_r\}_{r \in R}\}$. *Then the budget is* $B^{(F)}(G) = 1$.

Proof. Assume the N-family generic allocation game G. The budget, $B^{(F)}(G)$, is defined as

$$B^{(F)}(G) = \max_{a \in A} \frac{C(a;d)}{\sum\limits_{i \in N} J_i^{(F)}(a;d)} \tag{6.45}$$

$$= \max_{a \in A} \frac{C(a)}{\sum\limits_{i \in N} \sum\limits_{r \in a_i} C_r(|(a_i, \varnothing)|_r; d)} \tag{6.46}$$

$$= \max_{a \in A} \frac{\sum\limits_{r \in R} \beta_r |a|_r^d}{\sum\limits_{i \in N} \sum\limits_{r \in a_i} \beta_r} \tag{6.47}$$

$$= \max_{a \in A} \frac{\sum\limits_{r \in R} \beta_r |a|_r^d}{\sum\limits_{r \in R} \beta_r |a|_r} \tag{6.48}$$

$$\leq \max_{a \in A} \frac{\sum\limits_{i \in N} \sum\limits_{r \in a_i} \beta_r}{\sum\limits_{i \in N} \sum\limits_{r \in a_i} \beta_r} \tag{6.49}$$

$$= 1 \tag{6.50}$$

where it is strictly equal when all players select a unique resource. Hence, $B^{(F)}(G) = 1$. □

Fig. 6.3 illustrates the game scenario which defines the budget, $B^{(F)}(G)$, from Theorem 15.

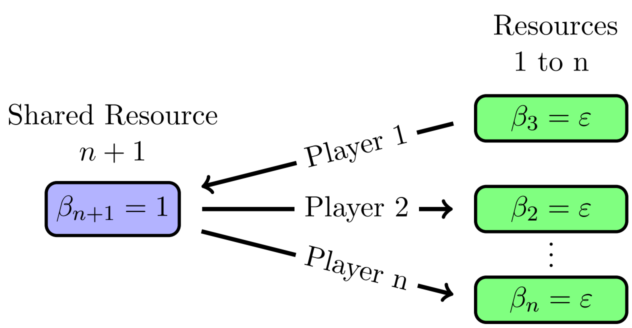


Figure 6.3: The generic allocation game scenario that defines the budget, $B^{(F)}(G)$, proven in Theorem 15, is when player 1 selects resource $n+1$, while all other players select their unique resource.

6.5.2 Budget Proof for the MC Player Cost Function

Theorem 16. *Let an N-family generic allocation game be*

$G = \{\{1, \ldots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i^{(MC)}\}_{i \in N}, R, \{C_r\}_{r \in R}\}$. *Then the budget is*

$B^{(MC)}(G) = \frac{1}{n^{1-d} \cdot (n^d - (n-1)^d)}$.

Proof. Assume the N-family generic allocation game G. The budget, $B^{(MC)}(G)$, is defined as

$$B^{(MC)}(G) = \max_{a \in A} \frac{C(a;d)}{\sum_{i \in N} J_i^{(MC)}(a;d)} \tag{6.51}$$

$$= \max_{a \in A} \frac{C(a;d)}{\sum_{i \in N} \sum_{r \in a_i} [C_r(|a|_r;d) - C_r(|(\varnothing, a_{-i})|_r;d)]} \tag{6.52}$$

$$= \max_{a \in A} \frac{\sum_{r \in R} \beta_r |a|_r^d}{\sum_{i \in N} \sum_{r \in a_i} [\beta_r |a|_r^d - \beta_r (|a|_r - 1)^d]} \tag{6.53}$$

$$\leq \frac{n^d}{n \cdot (n^d - (n-1)^d)} \tag{6.54}$$

$$= \frac{1}{n^{1-d} \cdot (n^d - (n-1)^d)} \tag{6.55}$$

where inequality Eq. (6.54) is strictly equal when all players select the same resource. We remark that since $\frac{1}{n^d - (n-1)^d} > n^{1-d}$, therefore $B^{(MC)}(G) > 1$ for $d \in [0,1)$[1]. □

Fig. 6.4 illustrates the game scenario which defines the budget, $B^{(MC)}(G)$, from Theorem 16.

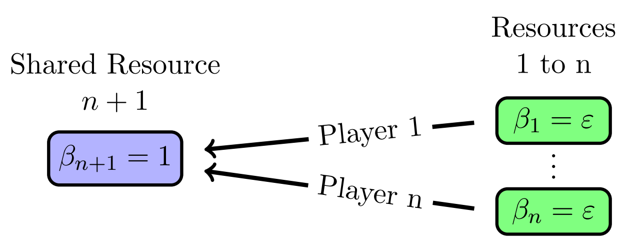


Figure 6.4: The generic allocation game scenario that defines the budget, $B^{(MC)}(G)$, proven in Theorem 16, is when all players select the shared resource $n+1$.

6.5.3 Budget Proof for the z Player Cost Function

Theorem 17. *Let an N-family generic allocation game be*

$G = \{\{1,\ldots,n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i^{(z)}\}_{i \in N}, R, \{C_r\}_{r \in R}\}$.

[1] *See* Theorem 31 in Appendix D for proof.

Then the budget $B^{(z)}(G) = \max_{\hat{n}:\hat{n}\leq n} \frac{1}{\hat{n}^{1-d}(\hat{n}^d-(\hat{n}-1)^d)}$, *where integer* $\hat{n} \geq 1$.

Proof. Assume the N-family generic allocation game G. The budget, $B^{(z)}(G)$, is defined as

$$B^{(z)}(G) = \max_{a\in\mathcal{A}} \frac{C(a;d)}{\sum\limits_{i\in N} J_i^{(z)}(a;d)} \tag{6.56}$$

$$= \max_{a\in\mathcal{A}} \frac{\sum\limits_{r\in R} \beta_r |a|_r^d}{\sum\limits_{i\in N}\sum\limits_{r\in a_i} \Big((1-z)\cdot J_i^{(F)}(a_i,a_{-i};d) + z\cdot J_i^{(MC)}(a_i,a_{-i};d)\Big)} \tag{6.57}$$

$$\leq \max_{\hat{n}:\hat{n}\leq n} \frac{1}{\hat{n}^{1-d}(\hat{n}^d-(\hat{n}-1)^d)} \tag{6.58}$$

An exhaustive search over the possible player set sizes reveals that Eq. (6.58) follows. Considering game scenarios where agents might select individual resources of non-negligible value, it is clear that this action would strictly decrease the budget bound. Considering all other game scenarios we see that the budget is defined when some set of players selects the same resource, $n+1$, and the remaining players select a unique resource, each with negligible value (dependent on d and z). We use the game scenario to motivate understanding for this proof shown in Fig. 6.5, which is a budget that mimics the the convex nature of the z agent cost function design. That is, the model that defines the budget is a transitioning model between the fixed cost function budget model (Fig. 6.3) and the MC cost function budget model (Fig. 6.4).

□

6.5.3.1 Example

Consider an N-family generic allocation game
$G = \{\{1,\ldots,n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i^{(z)}\}_{i\in N}, R, \{C_r\}_{r\in R}\}$. A plot showing the number of players that defines $B^{(z)}(G)$ for this 25 agent generic allocation game is shown in Figs. 6.6a and 6.6b. Fig. 6.7a depicts $B^{(z)}(G)$ as a function of z and d for this same 25 player game. Lastly, Fig. 6.7b depicts the upper bound on $B^{(z)}(G)$ as afforded by Theorem 7, Eq. (5.11), for a range of z and d

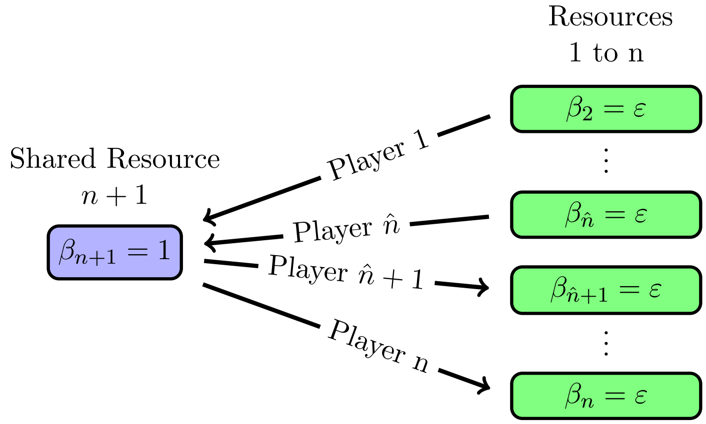


Figure 6.5: The generic allocation game scenario that defines the budget, $B^{(z)}(G)$, proven in Theorem 17, is when a subset of players (of size $\hat{n}$, where $\hat{n} \geq 1$) selects the shared resource, while all other players select their unique resource.

values. We remark that Theorem 7 presents a budget upper bound that adheres to Corollary 2. We remark that $B^{(z)}(G; z \in [0, 0.5]) = 1$.

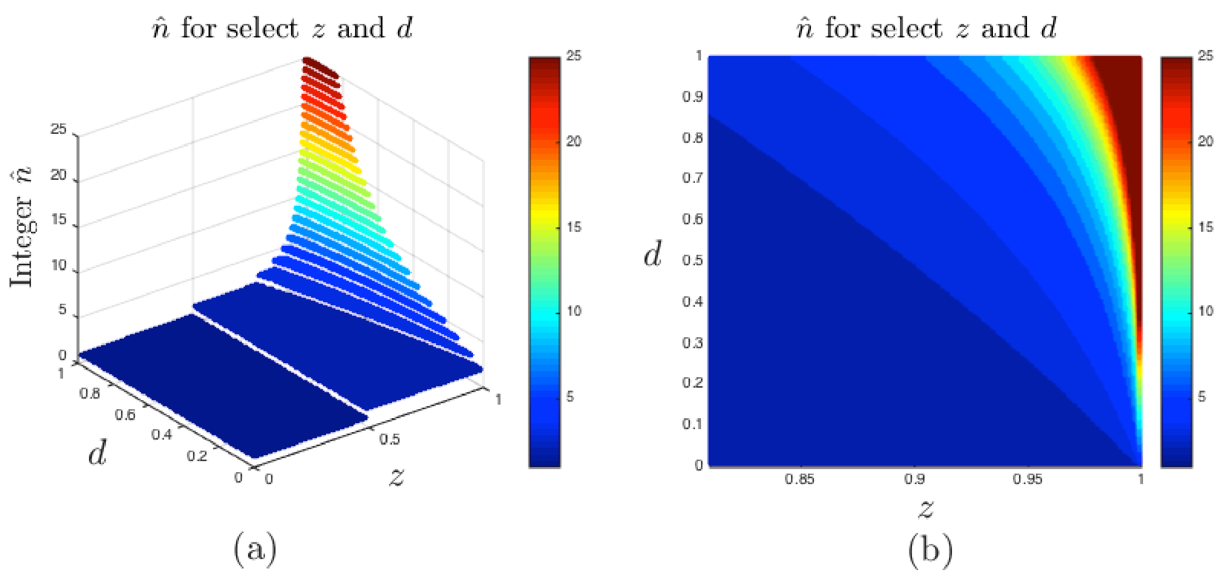
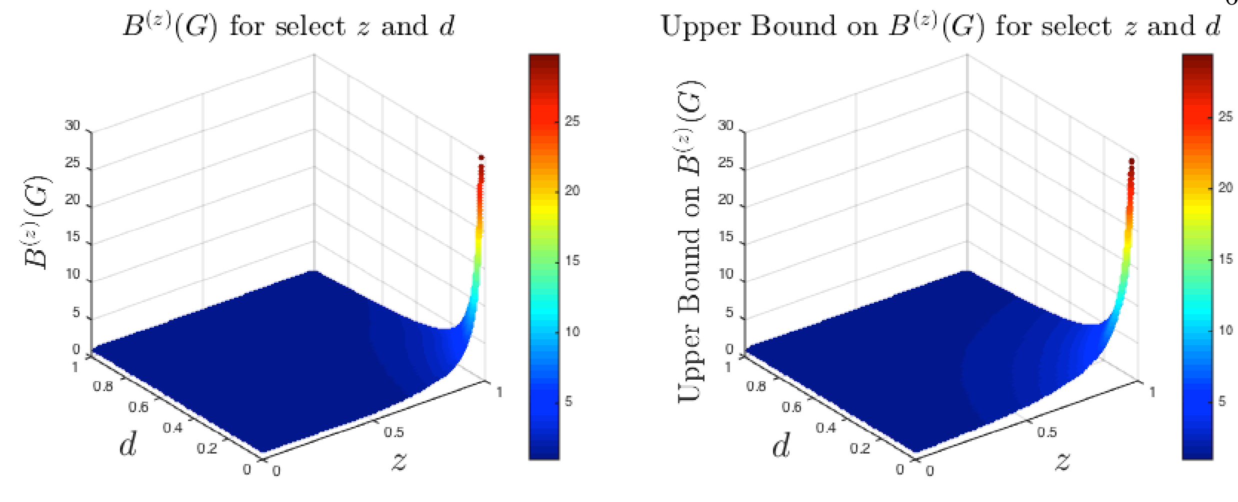


Figure 6.6: (a) A plot showing the size of $\hat{n}$ that defines $B^{(z)}(G)$ for select values of $z \in (0, 0.99)$ and $d \in (0, 1)$ for a 25 player game (note the consecutive steps); (b) top view of (a) for $d \in (0, 1)$ and $z \in (0.8, 1)$.



(a) (b)

Figure 6.7: (a) A plot of $B^{(z)}(G)$ for select $z \in (0,1)$ and $d \in (0,1)$ for a 25 player game that implements the z cost function for the generic allocation game; (b) a plot of the upper bound on $B^{(z)}(G)$ as afforded by Theorem 7, Eq. (5.11), for select $z \in (0,1)$ and $d \in (0,1)$ for the same 25 player game.

6.5.4 Budget Proof for the Shapley Agent Cost Function

Theorem 18. *Let an N-family generic allocation game be*

$G = \{\{1,\ldots,n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i^{(S)}\}_{i\in N}, R, \{C_r\}_{r\in R}\}$. *Then the budget* $B^{(S)}(G) = 1$.

Proof. Assume the N-family generic allocation game G. The budget, $B^{(S)}(G)$, is defined as

$$B^{(S)}(G) = \max_{a\in A} \frac{C(a;d)}{\sum_{i\in N} J_i^{(S)}(a;d)} \tag{6.59}$$

$$= \max_{a\in A} \frac{\sum_{i\in N}\sum_{r\in a_i} \beta_r \frac{|a|_r^d}{|a|_r}}{\sum_{i\in N}\sum_{r\in a_i} \beta_r \frac{|a|_r^d}{|a|_r}} \tag{6.60}$$

$$= 1 \tag{6.61}$$

□

The Shapley agent cost function is budget balanced for all actions $a \in \Pi_{i\in N} A_i$.

6.6 Proving PoA Upper Bound for Generic Allocation Games

We now prove the upper bound on the PoA for generic allocation games that utilize each of the player cost functions presented in Section 6.3. We note that the proofs for each are similar. Table 6.4 summarizes the budget bounds proven in this section.

Table 6.4: The upper bound on the PoA for an N-family generic allocation game, $G = \{\{1,\ldots,n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i\}_{i\in N}, R, \{C_r\}_{r\in R}\}$.

Agent Cost Function	**PoA(G) (Upper Bound)**	**Theorem**	**Is the bound always tight?**
Fixed	n^{1-d}	Theorem 19	Yes
Marginal Contribution	$\frac{1}{n^d-(n-1)^d}$	Theorem 20	Yes
z	$\frac{n^{1-d}}{\min_{\hat{n}:\hat{n}\leq n} \hat{n}^{1-d}(\hat{n}^d-(\hat{n}-1)^d)}$	Theorem 21	$z \in [0, 0.5]$: Yes $z \in (0.5, 1)$: Sometimes $z = 1$: Yes
Shapley	n^{1-d}	Theorem 22	Yes

6.6.1 PoA Upper Bound for the Fixed Player Cost Function

Theorem 19. *Let an N-family generic allocation game be $G = \{\{1,\ldots,n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i^{(F)}\}_{i\in N}, R, \{C_r\}_{r\in R}\}$. Then $1 \leq PoA(G) \leq= n^{1-d}$ and is tight.*

Proof. The proof follows by combining Theorems 2, 11, and 15.

Consider a game where every player $i \in N$ has access to only two resources from a set of $n+1$ resources, where one is unique to their action set and where the other is mutually accessible by all other agents; hence, $\{i, n+1\} \in A_i$ for all $i \in N$. Fig. 6.8 illustrates this game with defined resource values. Hence it is tight.

□

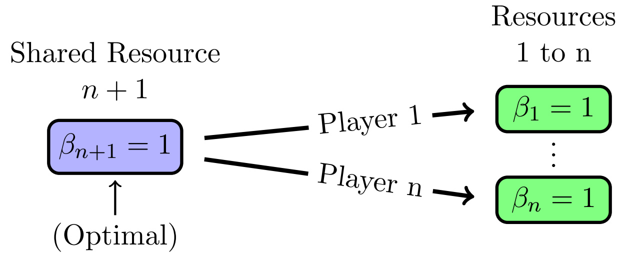


Figure 6.8: This illustrates the generic allocation game that demonstrates the upper PoA bound afforded by Theorem 19 is tight. The worst case equilibrium is when all agents select the **green** resources; the action profile that minimizes the global cost is when all agents select the shared **blue** resource.

6.6.2 PoA Upper Bound for the MC Player Cost Function

Theorem 20. *Let an N-family generic allocation game be*

$G = \{\{1,\ldots,n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i^{(MC)}\}_{i\in N}, R, \{C_r\}_{r\in R}\}$. *Then* $1 \leq PoA(G) \leq \frac{1}{n^d-(n-1)^d}$ *and is tight.*

Proof. The proof follows by combining Theorems 2, 12, and 16. We remark that this upper PoA bound is at least as much as n^{1-d} since $\frac{1}{n^d-(n-1)^d} > n^{1-d}$ (*see* Theorem 31 in Appendix D for proof).

Consider a game where every player $i \in N$ has access to only two resources, where both are mutually accessible by all other agents; hence, $\{1, n+1\} \in A_i$ for all $i \in N$. Fig. 6.9 illustrates this game with defined resource values. Hence it is tight.

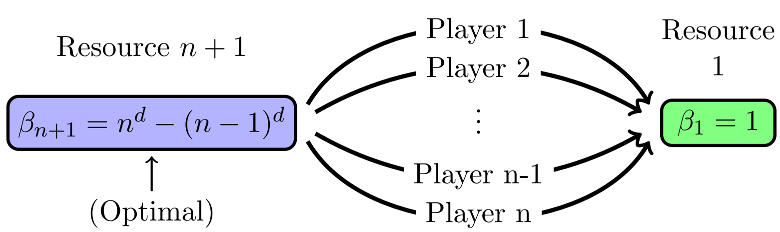


Figure 6.9: This illustrates the generic allocation game that demonstrates the upper PoA bound afforded by Theorem 20 is tight. The worst case equilibrium is when all agents select the shared **green** resource; the action profile that minimizes the global cost is when all agents select the shared **blue** resource.

□

6.6.3 PoA Upper Bound for the z Player Cost Function

Theorem 21. *Let an N-family generic allocation game be*

$G = \{\{1, \ldots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i^{(z)}\}_{i \in N}, R, \{C_r\}_{r \in R}\}$. *Then*

$1 \leq PoA(G) \leq \frac{n^{1-d}}{\min_{\hat{n}:\hat{n} \leq n} \hat{n}^{1-d}(\hat{n}^d - (\hat{n}-1)^d)}$, *and is tight if $n = 2$. If $n > 2$, it is tight for $z \in [0, 0.5]$.*

Proof. The proof follows by combining Theorems 2, 13, and 17.

We first prove tightness for any player set size for $z \in [0, 0.5]$.[1] Consider a game where every player $i \in N$ has access to only two resources from a set of $n + 1$ resources, where one is unique to their action set and where the other is mutually accessible by all other agents; hence, $\{i, n+1\} \in A_i$ for all $i \in N$. Fig. 6.10 illustrates this game with defined resource values. Hence it is tight.

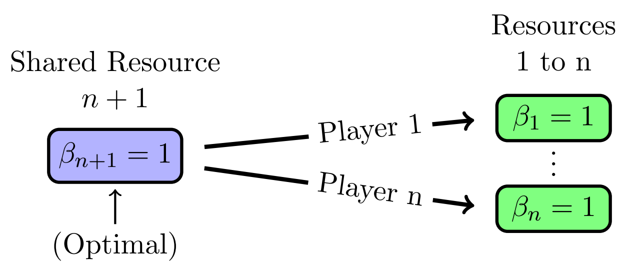


Figure 6.10: This illustrates the generic allocation game that demonstrates the upper PoA bound afforded by Theorem 21 is tight for $z \in [0, 0.5]$. The worst case equilibrium is when all agents select the **green** resources; the action profile that minimizes the global cost is when all agents select the shared **blue** resource. (Denoted as Game (a).)

Now we consider when $n = 2$, for $z \in (0.5, 1]$.[2] Consider a game where every player $i \in N$ has access to only two resources, where both are mutually accessible by all other agents; hence, $\{1, n+1\} \in A_i$ for all $i \in N$. Fig. 6.11 illustrates this game with defined resource values. Hence it is tight.

□

[1] We remark that for Theorem 17, $\hat{n} = 1$ maximizes the expression for $B^{(z)}(G)$ for this interval.

[2] We remark that for Theorem 17, $\hat{n} = 2$ maximizes the expression for $B^{(z)}(G)$ for this interval.

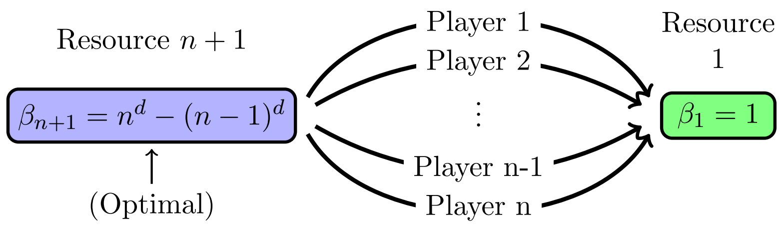


Figure 6.11: This illustrates the generic allocation game that demonstrates the upper PoA bound afforded by Theorem 21 is tight for two player games for $z \in [0.5, 1]$. The worst case equilibrium is when all agents select the shared **green** resource; the action profile that minimizes the global cost is when all agents select the shared **blue** resource. (Denoted as Game (c).)

We specifically do not claim tightness for the z agent cost function for $z \in (0.5, 1]$. Let's now discuss why this is: Consider the scenario where N is a set of three players. For $z \in (0.5, 0.5 + \delta]$, where $\delta < 0.5$ is dependent on the value for d, exhaustive modeling reveals that no game construction will have a PoA equal to the upper bound proven in Theorem 21. We demonstrate instead the game structures that grant the actual PoA.

In a three agent game, consider a game where each player $i \in N$ has access to only two resources from a set of $n+1$ resources, where one is unique to their action set and where the other is mutually accessible by all other agents; hence, $\{i, n+1\} \in A_i$ for all $i \in N$. Already we have shown that for $z \in [0, 0.5]$, this results in a tight game. Now consider $z \in (\delta, 1]$. The game presented for the proof of Theorem 20 (Fig. 6.9) is the one that maximizes this PoA.

The final range is $z \in (0.5, 0.5+\delta)$ for some δ. Consider two players sharing two resources, and the final player sharing one of those two resources, and having the sole access to a third resource. Fig. 6.12 illustrates this game.

When we see the game structures that define the actual PoA for the generic allocation game with the z cost function, a pattern emerges. Fig. 6.13a plots the PoA bounded above by Theorem 21. Fig. 6.13b seeks to clarify Fig. 6.13a by plotting the actual PoA that is realizable for each game. From the gap between the red and black curves in Fig. 6.13a, it's clear that the upper

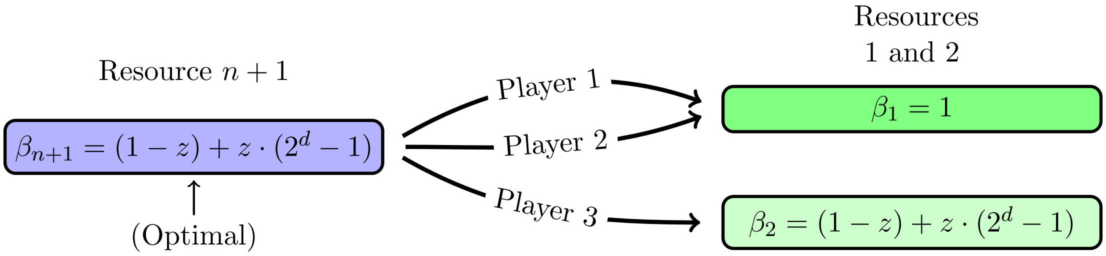
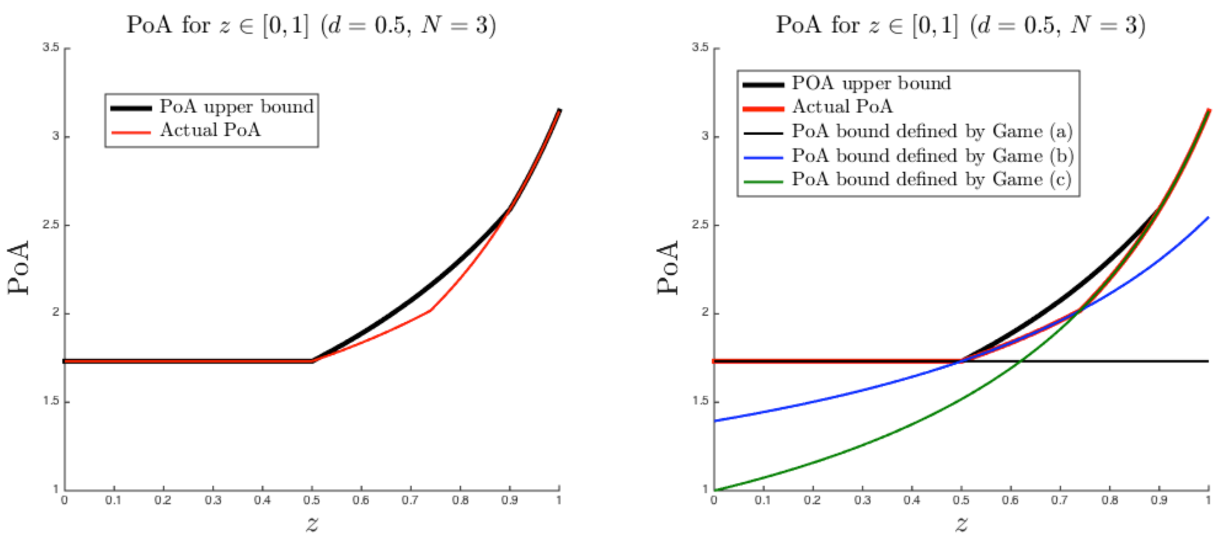


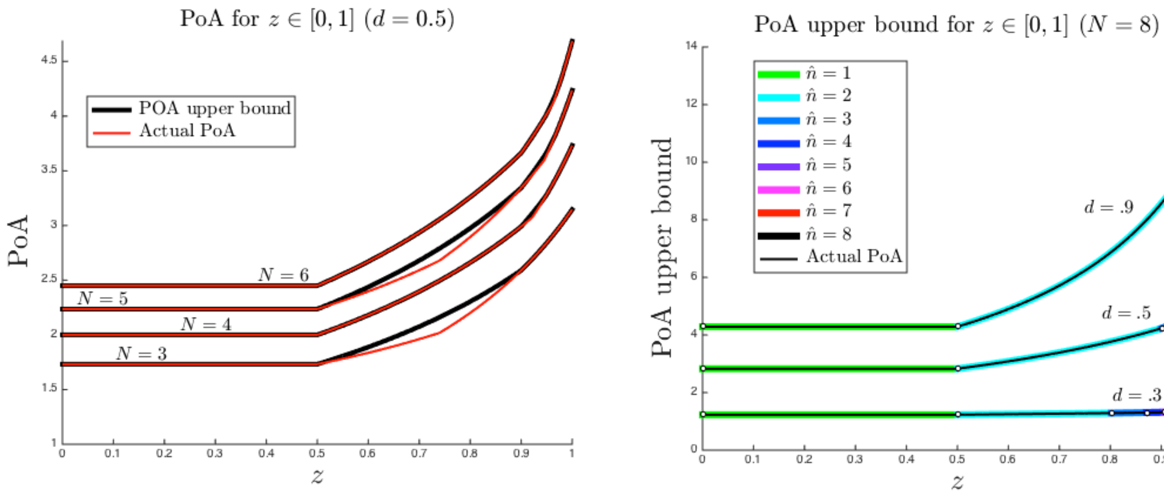
Figure 6.12: This illustrates the generic allocation game that demonstrates the largest PoA that can be reached for a three player game for $z \in (0.5, \delta]$ for some $\delta < 0.5$. The worst case equilibrium is when two agents select the shared **green** resource 1, and the third selects the non-shared resource 2; the action profile that minimizes the global cost is when all agents select the shared **blue** resource. (Denoted as Game (b).)

bound provable via smoothness is not always tight. Additional examples arise when plotting this for larger player sets. Using similar game constructions, Fig. 6.14a reveals how the tightness of the PoA bound varies with respect to the number of agents in the game for $z \in (0.5, 1)$. Fig. 6.14b provides more insight into how the PoA upper bound is crafted.



(a) (b)

Figure 6.13: (a) This illustrates the generic allocation game that demonstrates the largest PoA that can be reached for a three player game for $z \in [0, 1]$; **black** represents the PoA bound afforded by Theorem 21; **red** represents the actual PoA that can be achieved; (b) this plots how the actual PoA is obtained for three players; thin **black** represents the achievable PoA for $z \in [0, 0.5]$ for Game (a) from Fig. 6.10, **blue** represents the PoA realizable for $z \in (0.5, \delta]$ for Game (b) from Fig. 6.12, and **green** represents the reachable PoA for $z \in (\delta, 1]$ from Game (c) in Fig. 6.11.



(a) (b)

Figure 6.14: (a) This illustrates the generic allocation game that demonstrates the largest PoA that can be reached for multiple agent set sizes in a game for $z \in [0, 1]$; **black** represents the PoA bound afforded by Theorem 21; **red** represents the actual PoA that can be achieved; (b) this plot clarifies how the PoA upper bound curve is created; since the bound varies only with respect to the budget, this shows how the budget is created from Theorem 21.

This non-tightness results from the budget definition. Since the budget is defined over the set of *all* actions, $a \in A$, it will include non-equilibria action profiles. If any of those action profiles define the budget condition, then the resulting PoA bound implied by Theorem 2 will not be tight. A straightforward solution would be to define the budget using only the set of CCE equilibria. That is, for a game G, we could define a CCE budget, $B_{CCE}(G)$, as

$$B_{CCE}(G) = \max_{\sigma^{CCE} \in \widehat{\Sigma}(G)} \left\{ \frac{C(a)}{\sum_{i \in N} \mathbf{E}_{a \sim \sigma} \left[J_i(a) \right]} \right\}, \tag{6.62}$$

where $\widehat{\Sigma}(G)$ are the set of all joint probability distributions in game G. With little effort, it's clear that this budget would imply that the PoA for the set of all equilibria types is bounded above by

$$PoA \leq \frac{\lambda B_{CCE}(G)}{1 - \mu B_{CCE}(G)}, \tag{6.63}$$

given the game is (λ, μ)-smooth with $\lambda > 0$ and $\mu B_{CCE}(G) < 1$. However, while this solution is correct, it would force a designer to find all the equilibria before calculating this new budget, which could be an insurmountable problem in complex game settings. Therefore this game aids to demonstrate that finding the robust smoothness parameters does not always result in a tight PoA bound.

6.6.4 PoA Upper Bound for the Shapley Agent Cost Function

Theorem 22. *Let an N-family generic allocation game be*

$G = \{\{1, \ldots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i^{(S)}\}_{i \in N}, R, \{C_r\}_{r \in R}\}$*. Then* $1 \leq PoA(G) \leq n^{1-d}$ *and is tight.*

Proof. The proof follows by combining Theorems 2, 14, and 18.

Consider a game where every player $i \in N$ has access to only two resources from a set of $n+1$ resources, where one is unique to their action set and where the other is mutually accessible by all other agents; hence, $\{i, n+1\} \in A_i$ for all $i \in N$. Fig. 6.15 illustrates this game with defined resource values. Hence it is tight.

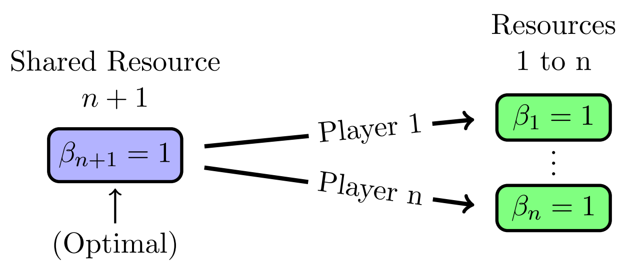


Figure 6.15: This illustrates the generic allocation game that demonstrates the upper PoA bound afforded by Theorem 22 is tight. The worst case equilibrium is when all agents select the **green** resources; the action profile that minimizes the global cost is when all agents select the shared **blue** resource.

□

6.7 Proving PoA Lower Bound for Generic Allocation Games

Now we prove that the optimal player cost function gives a PoA no less than n^{1-d} for the set of all generic allocation games with an agent cost function as defined within Definition 15.[1]

Theorem 23. *Consider any N-family generic allocation game,*
$G = \{\{1, \ldots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i\}_{i \in N}, R, \{C_r\}_{r \in R}\}$*. For every local agent cost function as defined within Definition 15 that is local and scalable, the PoA is at least n^{1-d}.*

Proof. Consider a game where every player $i \in N$ has access to only two resources from a set of $n+1$ resources, where one is unique to their action set and where the other is mutually accessible by all other agents; hence, $\{i, n+1\} \in A_i$ for all $i \in N$ for some $A \in \mathcal{A}$. Fig. 6.8 illustrates this game with defined resource values. Fig. 6.8 illustrates this game with defined resource values. Suppose initially that each player selects their unique resource i. Since the agent cost function is assumed to be local, there is no reason any player would prefer to switch to resource $n+1$. Thus a pure Nash equilibrium is reached, and the $PoA(G) = \frac{N}{n^d} = n^{1-d}$. Hence, any local and scalable agent cost function could result in a PoA for the N-family generic allocation game of at least n^{1-d}. $\square$

Given Theorem 23, we remark that the fixed, MC, z (for $z \in [0, 0.5]$), and Shapley agent cost functions afford the lowest PoA for all equilibria types.

6.8 Proving PoS Upper Bound for Generic Allocation Games

We now find the PoS upper bounds for generic allocation games that utilize each of the player cost functions presented in Section 6.3. We note that the proofs for each are similar. Table 6.5 summarizes the budget bounds proven in this section.

[1]This proof closely follows that of Theorem 6.1 in [12].

Table 6.5: The upper bound on the PoS for a generic allocation game,
$G = \{\{1, \ldots, n\} = N, \{A_i\}_{i \in N}, \{J_i\}_{i \in N}, R, \{C_r\}_{r \in R}\}$.

Agent Cost Function	**PoS(G) (Upper Bound)**	**Theorem**	**Is the bound always tight?**
Fixed	n^{1-d}	Theorem 24	Yes
Marginal Contribution	$(1-z) \cdot n^{1-d} + z$	Theorem 24	Yes
z	$\frac{n^{1-d}}{\min_{\hat{n}:\hat{n} \leq n} \hat{n}^{1-d}(\hat{n}^d - (\hat{n}-1)^d)}$	Theorem 24	Yes
Shapley	$\frac{\mathcal{H}_n^{(1-d)}}{n^d}$	Theorem 25	Yes

6.8.1 PoS Upper Bound for the Fixed, MC, and z Player Cost Functions

Theorem 24. *Let an N-family generic allocation game be*
$G = \{\{1, \ldots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i^{(z)}\}_{i \in N}, R, \{C_r\}_{r \in R}\}$*. Then* $1 \leq PoS(G) \leq (1-z) \cdot n^{1-d} + z$ *and is tight.*

Proof. Assume the N-family generic allocation game G. We first find the potential function. Consider any two actions $a_i', a_i'' \in A_i$ for any $A_i \in \mathcal{A}_i$ and $i \in N$, and any action $a_{-i} \in A_{-i}$ for any $A_{-i} \in \mathcal{A}_{-i}$. The agent cost functions are

$$J_i(a_i', a_{-i}) = (1-z) \cdot C_r(|(a_i', \varnothing)|; d) + z \cdot (C_r(|(a_i', a_{-i})|; d) - C_r(|(\varnothing, a_{-i}|); d) \tag{6.64}$$

$$J_i(a_i'', a_{-i}) = (1-z) \cdot C_r(|(a_i'', \varnothing)|; d) + z \cdot (C_r(|(a_i'', a_{-i})|; d) - C_r(|(\varnothing, a_{-i}|); d) \tag{6.65}$$

Therefore,

$$J_i(a_i', a_{-i}; d) - J_i(a_i'', a_{-i}; d) \tag{6.66}$$

$$= (1-z) \cdot C_r(|(a_i', \varnothing)|; d) + z \cdot (C_r(|(a_i', a_{-i})|; d) - C_r(|(\varnothing, a_{-i})|; d) \tag{6.67}$$

$$- \big((1-z) \cdot C_r(|(a_i'', \varnothing)|; d) + z \cdot (C_r(|(a_i'', a_{-i})|; d) - C_r(|(\varnothing, a_{-i})|; d)\big) \tag{6.68}$$

$$= z \cdot C(|(a_i', a_{-i})|; d) + (1-z) \sum_{i \in N} \sum_{r \in a_i} C_r(|(a_i, \varnothing)|_r; d) \tag{6.69}$$

$$- \big(z \cdot C(|(a_i'', a_{-i})|; d) + (1-z) \sum_{i \in N} \sum_{r \in a_i} C_r(|(a_i, \varnothing)|_r; d)\big) \tag{6.70}$$

Hence it is a potential game with potential function, $\phi(a;d)$, given by

$$\phi(a;d) = z \cdot C(a;d) + (1-z)\sum_{i\in N}\sum_{r\in a_i} C_r(|(a_i,\varnothing)|_r;d). \tag{6.71}$$

and the game therefore has at least one pure Nash equilibrium. We use Theorem 4 to upper bound the PoS(G).[1]

- Using Definition 14, we first find $B_\phi(G)$:

$$B_\phi(G) = \max_{a\in\mathcal{A}} \frac{C(a;d)}{z\cdot C(a;d) + (1-z)\sum_{i\in N}\sum_{r\in a_i} C_r(|(a_i,\varnothing)|_r;d)} \tag{6.72}$$

$$\leq \max_{a\in\mathcal{A}} \frac{C(a;d)}{z\cdot C(a;d) + (1-z)\sum_{r\in R} C_r(|a|_r;d)} \tag{6.73}$$

$$= \max_{a\in\mathcal{A}} \frac{C(a;d)}{z\cdot C(a;d) + (1-z)\cdot C(a;d)} \tag{6.74}$$

$$= 1 \tag{6.75}$$

- Then using Definition 14, we find $\overline{B}_\phi(G)$:

$$\overline{B}_\phi(G) = \min_{a\in A} \frac{C(a;d)}{z\cdot C(a;d) + (1-z)\sum_{i\in N}\sum_{r\in a_i} C_r(|(a_i,\varnothing)|_r;d)} \tag{6.76}$$

$$\geq \min_{a\in A} \frac{C(a;d)}{z\cdot C(a;d) + (1-z)\cdot n^{1-d}C(a;d)} \tag{6.77}$$

$$= \frac{1}{z+(1-z)n^{1-d}} \tag{6.78}$$

$$\Rightarrow \frac{1}{\overline{B}_\phi(G)} \leq z + (1-z)\cdot n^{1-d} \tag{6.79}$$

Then by Theorem 4,

$$PoS(G) \leq \frac{B_\phi(G)}{\overline{B}_\phi(G)} \tag{6.80}$$

$$\leq (1-z)\cdot n^{1-d} + z. \tag{6.81}$$

[1] A similar proof using Theorem 24 exists to upper bound the PoS for the α and z cost functions in reverse carpooling from Chapter 4.

The tightness example follows: consider a game where every player $i \in N$ has access to only two resources from a set of $n+1$ resources, where one is unique to their action set and where the other is mutually accessible by all other agents; hence, $\{i, n+1\} \in A_i$ for all $i \in N$. Fig. 6.16 illustrates this game with defined resource values. Given the resource values in Fig. 6.16, it's clear that the only equilibrium is when all players select their unique resource. Hence it is tight.

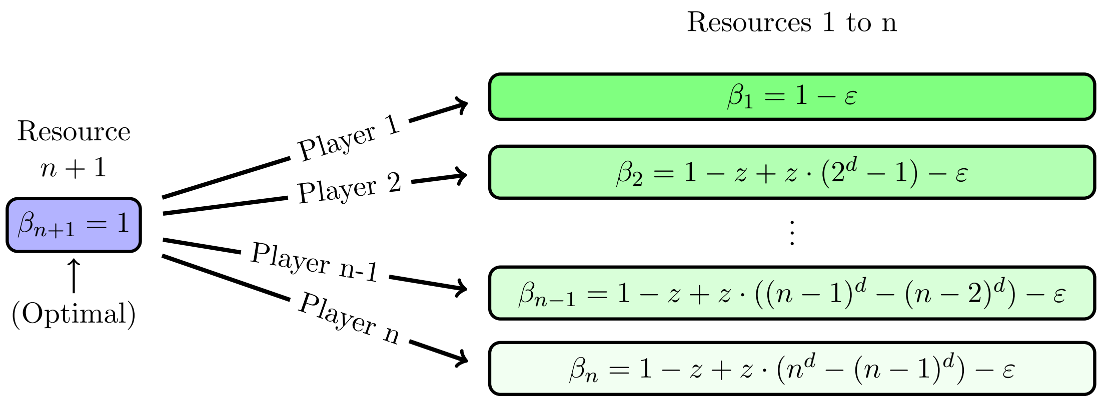


Figure 6.16: This illustrates the generic allocation game that demonstrates the upper PoS bound afforded by Theorem 24 is tight. The only equilibrium is when all agents select the **green** resources; the action profile that minimizes the global cost is when all agents select the shared **blue** resource.

□

6.8.2 PoS Upper Bound for the Shapley Agent Cost Function

Theorem 25. *Let an N-family generic allocation game be* $G = \{\{1, \ldots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i^{(S)}\}_{i \in N}, R, \{C_r\}_{r \in R}\}$. *Then* $1 \leq PoS(G) \leq \frac{\mathcal{H}_n^{(1-d)}}{n^d} = \frac{\sum_{i \in N} i^{d-1}}{n^d}$ *and is tight.*

Proof. Assume the N-family generic allocation game G. We first find the potential function. Consider any two actions profiles $a' = (a'_i, a_{-i}), a'' = (a''_i, a_{-i}) \in A$ for any $A \in \mathcal{A}$ and for any $i \in N$.

Then the agent cost functions are

$$J_i(a_i', a_{-i}; d) = \frac{C_r(|(a_i', a_{-i})|; d)}{|(a_i', a_{-i})|_r} \tag{6.82}$$

$$J_i(a_i'', a_{-i}; d) = \frac{C_r(|(a_i'', a_{-i})|; d)}{|(a_i'', a_{-i})|_r} \tag{6.83}$$

Therefore,

$$J_i(a_i', a_{-i}; d) - J_i(a_i'', a_{-i}; d) = \frac{C_r(|(a_i', a_{-i})|; d)}{|(a_i', a_{-i})|_r} - \frac{C_r(|(a_i'', a_{-i}|); d)}{|(a_i'', a_{-i})|_r} \tag{6.84}$$

$$= \sum_{r\in R} \sum_{x=1}^{|a'|_r} \frac{C_r(x; d)}{x} - \sum_{r\in R} \sum_{x=1}^{|a''|_r} \frac{C_r(x; d)}{x} \tag{6.85}$$

Hence it is a potential game with potential function, $\phi(a; d)$, given by

$$\phi(a; d) = \sum_{r\in R} \sum_{x=1}^{|a|_r} \frac{C_r(x; d)}{x} \tag{6.86}$$

and the game therefore has at least one pure Nash equilibrium. We now continue the proof using a similar method as in [1], although with some significant expansion on their work. Let action a^* be the action that minimizes the global cost function, and let action a be the pure Nash equilibrium that minimizes the potential function. Then we know that

$$\phi(a; d) \leq \phi(a^*; d). \tag{6.87}$$

We first find an upper bound on $\phi(a^*)$:

$$\phi(a^*;d) = \sum_{r\in R}\sum_{x=1}^{|a^*|_r} \frac{C_r(x;d)}{x} \tag{6.88}$$

$$= \sum_{r\in R}\sum_{x=1}^{|a^*|_r} \frac{\beta_r x^d}{x} \tag{6.89}$$

$$= \sum_{r\in R} \beta_r \cdot |a^*|_r^d \frac{\sum_{x=1}^{|a^*|_r} \frac{x^d}{x}}{|a^*|_r^d} \tag{6.90}$$

$$\leq \sum_{r\in R} \beta_r \cdot |a^*|_r^d \frac{\sum_{x=1}^{n} \frac{x^d}{x}}{n^d} \tag{6.91}$$

$$= \sum_{r\in R} \beta_r \cdot |a^*|_r^d \frac{\sum_{x=1}^{n} \frac{1}{x^{(1-d)}}}{n^d} \tag{6.92}$$

$$= \sum_{r\in R} \beta_r \cdot |a^*|_r^d \frac{\mathcal{H}_n^{(1-d)}}{n^d} \tag{6.93}$$

$$= \sum_{r\in R} C_r(a^*;d) \frac{\mathcal{H}_n^{(1-d)}}{n^d} \tag{6.94}$$

$$= C(a^*;d) \frac{\mathcal{H}_n^{(1-d)}}{n^d} \tag{6.95}$$

$$\Rightarrow \phi(a^*;d) \leq C(a^*;d) \frac{\mathcal{H}_n^{(1-d)}}{n^d} \tag{6.96}$$

where Eq. (6.91) follows since $\frac{\sum_{x=1}^{k} \frac{x^d}{x}}{k^d} \leq \frac{\sum_{x=1}^{m} \frac{x^d}{x}}{m^d}$ for all $k, m \geq 1$ and $d \in [0,1)$ such that $k \leq m$, and where $\mathcal{H}_n^{(1-d)}$ is the hyperharmonic series[1] defined as [14]:

$$\mathcal{H}_n^{(1-d)} = \sum_{x=1}^{n} \mathcal{H}_k^{(1-d)} = \sum_{x=1}^{n} \frac{1}{k^{(1-d)}} \tag{6.97}$$

We remark that Eq. (6.97) is a strictly diverging series for $d \in [0,1)$.

[1] Also known as the *p-series*.

Now we find a lower bound on $\phi(a)$:

$$\phi(a;d) = \sum_{r\in R}\sum_{x=1}^{|a|_r}\frac{C_r(x;d)}{x} \tag{6.98}$$

$$= \sum_{r\in R}\sum_{x=1}^{|a|_r}\frac{\beta_r x^d}{x} \tag{6.99}$$

$$= \sum_{r\in R}\beta_r\sum_{x=1}^{|a|_r}\frac{x^d}{x} \tag{6.100}$$

$$\geq \sum_{r\in R}\beta_r\cdot|a|_r \tag{6.101}$$

$$= C(a;d) \tag{6.102}$$

$$\Rightarrow C(a;d) \leq \phi(a;d) \tag{6.103}$$

where Eq. (6.101) follows since $\sum_{x=1}^{m}\frac{x^d}{x} \geq m^d$. Combining Eqs. (6.96), (6.87), and (6.103), we have

$$C(a;d) \leq \phi(a;d) \leq \phi(a^*;d) \leq C(a^*;d)\frac{\mathcal{H}_n^{(1-d)}}{n^d} \tag{6.104}$$

$$\Rightarrow \frac{C(a;d)}{C(a^*;d)} \leq \frac{\mathcal{H}_n^{(1-d)}}{n^d} \tag{6.105}$$

$$\Rightarrow PoS(G) \leq \frac{\mathcal{H}_n^{(1-d)}}{n^d} \tag{6.106}$$

The tightness example follows: consider a game where every player $i \in N$ has access to only two resources from a set of $n+1$ resources, where one is unique to their action set and where the other is mutually accessible by all other agents; hence, $\{i, n+1\} \in A_i$ for all $i \in N$. Fig. 6.17 illustrates this game with defined resource values. Given the resource values in Fig. 6.17, it's clear that the only equilibrium is when all players select their unique resource. It's also clear that the global minimizing action profile is when all players select the resource $n+1$. Hence it is tight.[1]

□

[1]We remark that $n^d - (n-1)^d \leq n^{d-1}$ by Theorem 31.

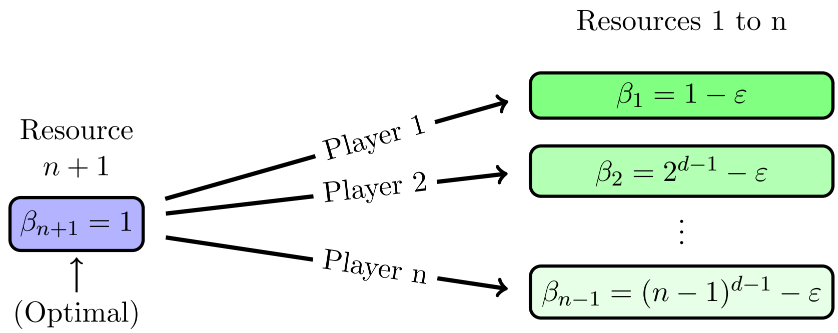


Figure 6.17: This illustrates the generic allocation game that demonstrates the upper PoS bound afforded by Theorem 25 is tight. The only equilibrium is when all agents select the **green** resources; the action profile that minimizes the global cost is when all agents select the shared **blue** resource.

6.9 Proving PoS Lower Bound

Here we show a lower bound on the PoS bound for the set of all player cost functions (the proof follows similarly to Theorem 6.2 in [12]):

Theorem 26. *Fix any local and scalable agent cost function as defined within Definition 15. Let the PoA be γ for the N-family generic allocation game $G = \{\{1, \ldots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i\}_{i \in N}, R, \{C_r\}_{r \in R}\}$ that implement this agent cost function. Then the PoS is lower bounded by*

$$PoS(G) \geq \frac{1 + \gamma \cdot (n-1)^d}{\gamma \cdot n^d}. \tag{6.107}$$

Proof. Consider the N-family generic allocation game G with two resources as illustrated in Fig. 6.18 and with any local agent cost function as defined within Definition 15. Consider the game illustrated in Fig. 6.18 for $\varepsilon > 0$ and $\to 0$. By Theorem 23, we recognize that $\beta_{n+1} \in (0, 1)$, and is therefore a valid value for a resource with a value strictly less than the value of resource 1 (*i.e.* $\beta_1 > \beta_{n+1}$).

We define three actions of interest:

- action $\widehat{a}$ is when all agent $i \in N$ select resource 1.

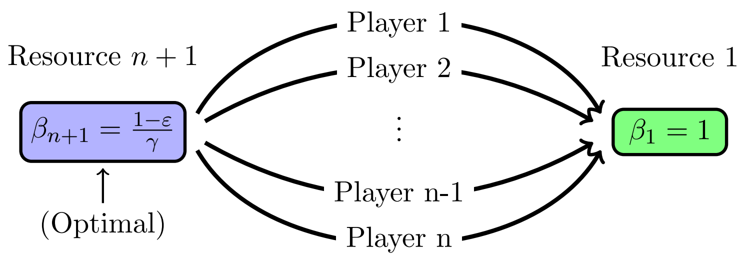


Figure 6.18: This illustrates the generic allocation game that forces a lower bound on the PoS, as a function of the PoA.

- action a is when $n-1$ agents select resource 1; the final agent selects resource $n+1$.

- action profile a^* is when all agents select resource $n+1$.

Given this game, it's obvious that the action which minimizes the global cost function is when all players select action a^*.

The global cost for action $\widehat{a}$ is

$$C(\widehat{a};d) = \sum_{r\in R} C_r(\widehat{a};d) \tag{6.108}$$

$$= n^d \tag{6.109}$$

The global cost for action a^* is

$$C(a^*;d) = \beta_2 \cdot n^d \tag{6.110}$$

$$= \frac{1-\varepsilon}{\gamma} \cdot n^d \tag{6.111}$$

$$\tag{6.112}$$

Hence, the fraction $\frac{C(\widehat{a};d)}{C(a^*;d)}$ is

$$\frac{C(\widehat{a};d)}{C(a^*;d)} = \frac{n^d}{\frac{1-\varepsilon}{\gamma} \cdot n^d} \tag{6.113}$$

$$= \frac{\gamma}{1-\varepsilon} \tag{6.114}$$

which is strictly greater than γ, the assumed PoA of the set of generic allocation games. Therefore action $\widehat{a}$ is strictly not an equilibrium. Hence, given the player cost function that is local and scalable, at least one player has the incentive to deviate to resource $n+1$.

Now, without loss of generality, restrict all players' action sets to only have the option to select resource 1, except for a single player that still can select both resources. Now only actions $\widehat{a}$ and a remain in the joint action set A. We remark that action a is the unique Nash equilibrium, while action $\widehat{a}$ is the optimal action that minimizes the global cost function (so long as $\frac{1-\varepsilon}{\gamma}+(n-1)^d \geq n^d$, which, by Theorem 31, is true for all $d \in [0,1)$).

The global cost for action a is

$$C(a;d) = \beta_2 + \beta_1 \cdot (n-1)^d \tag{6.115}$$

$$= \frac{1-\varepsilon}{\gamma} + (n-1)^d \tag{6.116}$$

and the PoS, $\frac{C(a;d)}{C(\widehat{a};d)}$ is therefore

$$\frac{C(a;d)}{C(\widehat{a};d)} = \frac{\beta_2 + \beta_1 \cdot (n-1)^d}{\beta_1 \cdot n^d} \tag{6.117}$$

$$= \frac{\frac{1-\varepsilon}{\gamma} + (n-1)^d}{n^d} \tag{6.118}$$

□

Theorem 27. *Fix any local and scalable agent cost function as defined within Definition 15 for the N-family generic allocation game $G = \{\{1,\ldots,n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i\}_{i\in N}, R, \{C_r\}_{r\in R}\}$, and that affords $PoA(G) \leq n^{1-d}$. Then the PoS is lower bounded by*

$$PoS(G) \geq \frac{1 + n^{1-d}(n-1)^d}{n}. \tag{6.119}$$

Proof. The proof follows by combining Theorem 23 with Theorem 26. □

Fig. 6.19 plots the PoA upper bound afforded by Theorem 21 for the 3-family of generic allocation games using the z player cost function. It also plots the achievable PoA and PoS. Then it shows the PoS lower bound given by Theorem 27. Lastly, it shows (at the budget balanced location of $z = 0.5$) the tight PoA and PoS points for 3-family games that implement the Shapley cost function.

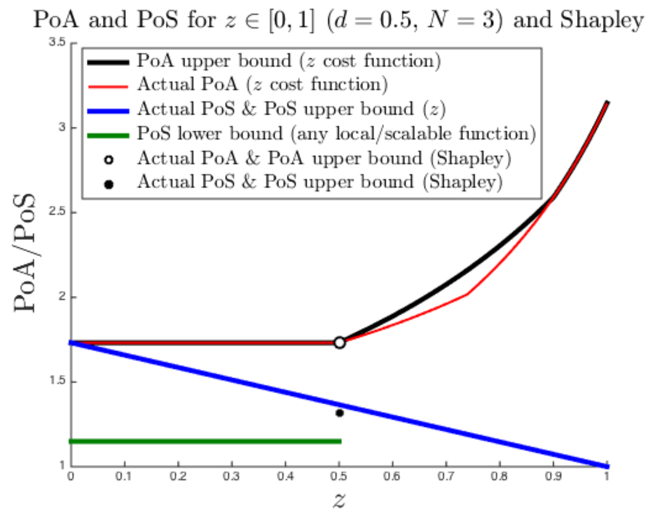


Figure 6.19: A plot showing the PoA (bounds and realizable) and PoS (bounds and realizable) for the z player cost function for the set of generic allocation games. Additionally plotted is the lower PoS bound. The Shapley PoA and PoS is shown for reference at the budget balanced location of $z = 0.5$. ($d = 0.5$, $N = 3$)

The plots raise an obvious question: is there a local and scalable agent cost function that grants the optimal PoA, while also granting the optimal PoS suggested by the lower bound of Theorem 27. The answer is no, as will be proven now.

6.10 The Shapley Cost Function is Optimal

Here we prove the optimal agent cost function that is both local and scalable, and that primarily affords the lowest *PoA* for an N-family of games, and secondarily affords the lowest PoS, is the Shapley cost function.

Theorem 28. *Fix any local and scalable agent cost function for any N-family generic allocation game $G = \{\{1, \ldots, n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i \in N}, \{J_i\}_{i \in N}, R, \{C_r\}_{r \in R}\}$. The agent cost function that* *primarily grants the lowest PoA, and secondarily grants the lowest PoS, is the Shapley cost function.*

Proof. Assume an N-family generic allocation game

$G = \{\{1,\ldots,n\} = N \subseteq \mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i\}_{i\in N}, R, \{C_r\}_{r\in R}\}$, and recall a local and scalable agent cost function is of the form

$$J_i(a;d) = \sum_{r\in a_i} \beta_r \cdot f(i,|a|_r;d) \tag{6.120}$$

with $f(i,1;d)=1$. Assume $J_i^*(a) = \sum_{r\in a_i} \beta_r \cdot f^*(i,|a|_r;d)$ is the local and scalable player cost function exists that is optimal.

- **Step 1:** Given that the Shapley cost function is local and scalable, and has a PoA(G) of n^{1-d}, then the optimal cost function must afford a PoA(G) no greater than n^{1-d}.

  Consider initially the case when $n=2$. We seek to define $f^*(i,2;d)$. Assume initially that $f^*(i,2;d) \leq 1$ for the game depicted in Fig. 6.20.

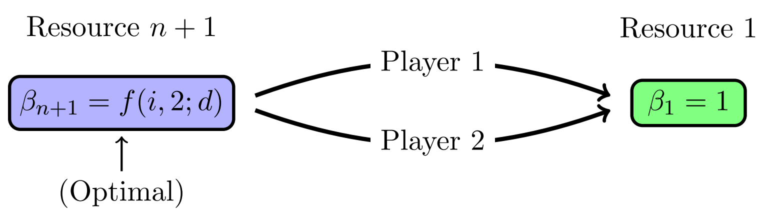


Figure 6.20: This illustrates the generic allocation game that forces a lower bound on the local agent cost function.

  Call action profile $a = (\beta_1, \beta_1)$ and action profile $a^* = (\beta_{n+1}, \beta_{n+1})$. Clearly resource $n+1$ is optimal when both players select it, while resource 1 is an equilibrium that defines the PoA for game G. That is, for player 1,

$$J_i^*(a_1,a_2;d) = \beta_1 \cdot f(i,2;d) \leq \beta_{n+1} \cdot f^*(i,1;d) = J_i^*((a_1^*,a_2);d) \tag{6.121}$$

$$\Rightarrow J_i^*((a_1,a_2);d) \leq J_i^*((a_1^*,a_2);d), \tag{6.122}$$

  and likewise for player 2. Hence, it is an equilibrium.

By assumption, this optimal agent cost function has a PoA no greater than n^{1-d}. Therefore

$$PoA(G) = \frac{\beta_1 \cdot 2^d}{\beta_{n+1} \cdot 2^2} \leq 2^{1-d} \tag{6.123}$$

$$\Rightarrow \frac{1}{f(i,2;d)} \leq 2^{1-d} \tag{6.124}$$

$$\Rightarrow f(i,2;d) \geq 2^{d-1} \tag{6.125}$$

- **Step 2:** Now suppose $f^*(i,2;d) > 1$. We will show that this is not allowed in a proof by contradiction. Consider the game depicted in Fig. 6.21.

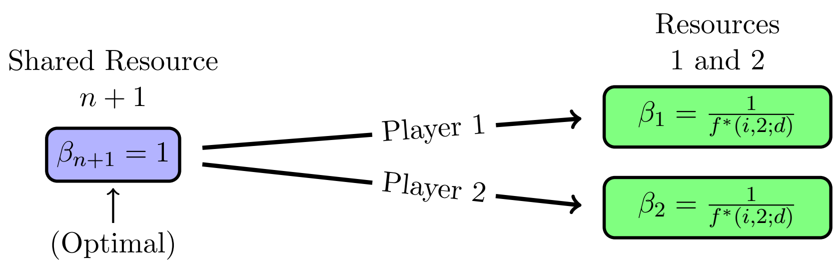


Figure 6.21: This illustrates the generic allocation game that forces an upper bound on the local agent cost function.

Call action profile $a = (\beta_1, \beta_2)$ and action profile $a^* = (\beta_{n+1}, \beta_{n+1})$. Clearly resource $n+1$ is optimal when both players select it, while resource 1 is an equilibrium that defines the PoA for this game G. That is, for player 1,

$$J_i^*(a_1, a_2; d) = \beta_1 \cdot f(i,2;d) \leq \beta_{n+1} \cdot f^*(i,1;d) = J_i^*((a_1^*, a_2); d) \tag{6.126}$$

$$\Rightarrow J_i^*((a_1, a_2); d) \leq J_i^*((a_1^*, a_2); d), \tag{6.127}$$

and likewise for player 2. Hence, it is an equilibrium. Calculating this equilibrium we have:

$$PoA(G) = \frac{\beta_1 + \beta_2}{\beta_{n+1} \cdot 2^2} \tag{6.128}$$

$$= \frac{\frac{1}{f^*(i,2;d)} + \frac{1}{f^*(i,2;d)}}{2^d} \tag{6.129}$$

$$= 2^{1-d} \cdot \frac{1}{f^*(i,2;d)} \tag{6.130}$$

But $\frac{1}{f^*(i,2;d)} > 1$, which leads to a contradiction. Now we know that $f^*(i,2;d) \in [2^{d-1}, 1]$.

- **Step 3:** Consider now the PoS. Given that the Shapley cost function is local and scalable, and affords the lowest allowable PoA for a local and scalable cost function by Theorem 23, therefore the optimal cost function must afford a PoS(G) no greater than $\frac{\mathcal{H}_n^{(1-d)}}{n^d}$ (given Theorem 25). In continuing the case of $n = 2$, this bound is:

$$\frac{\mathcal{H}_n^{(1-d)}}{n^d} = \frac{\frac{2^d}{2} + \frac{1^d}{1}}{2^d} \tag{6.131}$$

$$= \frac{2^{d-1} + 1}{2^d} \tag{6.132}$$

Suppose the game is the one depicted in Fig. 6.22.

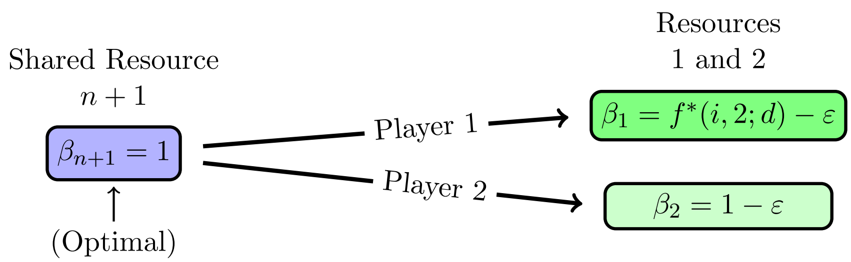


Figure 6.22: This illustrates the generic allocation game that forces an upper bound on the local agent cost function.

From Eqs. (6.125) and (6.135), we see that this is a valid game since all resource values are in $(0, 1]$. Call action profile $a = (\beta_1, \beta_2)$ and action profile $a^* = (\beta_{n+1}, \beta_{n+1})$. Action a is the unique Nash equilibrium in this game: Consider when both players select the shared resource. Given the resource values, player 1 will desire to switch to resource 1 to decrease their cost function. Following, the remaining player will also desire to switch to resource 2. However, the optimal decision is when both agents select resource $n+1$ since $2^d \leq \beta_1 + \beta_2$. Hence, the PoS is

$$PoS(G) = \frac{f(i, 1; d) + f^*(i, 2; d) - 2 \cdot \varepsilon}{2^d} \leq \frac{\mathcal{H}_n^{(1-d)}}{n^d} \tag{6.133}$$

$$\Rightarrow \frac{1 + f^*(i, 2; d) - 2 \cdot \varepsilon}{2^d} \leq \frac{2^{d-1} + 1}{2^d} \tag{6.134}$$

$$\Rightarrow f^*(i, 2; d) \leq 2^{d-1} + 2 \cdot \varepsilon \tag{6.135}$$

where the right side can be made arbitrarily close to 2^{d-1}. Combining Eqs. (6.125) and (6.135) we see that, for $n = 2$, the optimal agent cost function that is local and scalable is

$$f^*(i,2;d) = 2^{d-1}. \tag{6.136}$$

- **Step 5:** Now we have defined $f^*(i,1;d) = 1$ and $f^*(i,2;d) = 2^{d-1}$.

We then repeat this same process for $n = 3, 4, \ldots$, and the result is $f^*(i,k;d) = k^{d-1}$ for all $k \in N$. This is equivalent to the Shapley cost function. □

6.11 Welfare Generic Allocation Game Equivalent

As mentioned in Section 6.1, an equivalent generic resource allocation game exists for cost minimization games with a global welfare function, and parallel results of this chapter can be concocted equivalently. We informally define this game now.

A class of (welfare) generic allocation games is a class of (cost) generic allocation games, $G = \{\mathcal{N}, \{\mathcal{A}_i\}_{i\in N}, \{J_i^{(z)}\}_{i\in N}, R, \{C_r\}_{r\in R}\}$, with the following modifications:

- the constant $d \geq 1$
- each agent is assigned an agent utility function that is designed to be maximized
- the global objective is to maximize the local welfare functions.

6.12 Summary

In Chapter 6 we looked at N-family generic allocation games, and demonstrated how player cost functions can be designed from two separate functions to create an improvement on overall equilibria bounds that might be achieved.[1] We discussed multiple player cost functions, and per-

[1] In particular, the z cost function retained the same PoA on the interval $z \in [0, 0.5]$, and strictly decreased the PoS on that same interval.

fectly bounded these player cost functions. We also demonstrated how the combined smoothness argument and budget condition does not necessarily tightly bound the resulting PoA for all budget conditions, even when the smoothness parameters are optimal.

The analysis on the Shapley cost function in this chapter demonstrated that (1) the Shapley cost function is the optimal local welfare function, and (2) the z cost function is not always an optimal player cost function that affords the lowest inefficiency bounds across multiple types of resource allocation games when compared to the Shapley cost function, when considering the set of all local and scalable agent cost functions. While the focus in this chapter is on a single resource, the results can be expanded to a multiple resource allocation settings with little effort.

In Section 6.9, we considered the possibility of an inherent tradeoff between the PoA and the PoS. The results from this chapter suggest this may be unlikely for a local and scalable cost function.

Chapter 7

Thesis Conclusion

7.1 Discussion

This thesis delved into the question of how to design player cost functions in distributed systems that have a global cost function, and how these agent functions are influenced by the budget. We provided metrics to assess player objective functions, and we expanded on existing methods used to quantify these functions.

Following, we discussed player objective function design in terms of convexly combining multiple player objective functions. We demonstrated how the expanded methods could aid in quantifying these new designs. We use two games to demonstrate the significance of these expansions. In the second example game, we presented shortcomings in the contributions, and discussed where those shortcomings came from. Although the example games were both resource allocation games, the results from Chapters 3 and 5 apply to any cost minimization game.

The significance of this thesis is unquestionable.

- The smoothness PoA bounds presented in Theorem 2 of Chapter 3 provide a method of bounding the worst-case inefficiencies across a broad class of games that may otherwise be inherently difficult to quantify.

- The reverse carpooling example from Chapter 4 soundly demonstrates how a broadly defined agent objective function can be defined and then "tuned" to have a PoA specified by a designer when implementing the previous contribution.

- The design of player cost functions using a convex combination of existing functions provides a novel and useful approach to agent function design in order to bound the PoA and PoS of the resulting design (Chapter 5).

- The theorems from Chapter 6 prove an optimal local and scalable agent cost function for a broad class of allocation games. This result may have a broad impact on design on agent objective functions that fall within this model.

- Lastly, all of these results equally apply to global welfare maximization settings.

7.2 Unanswered Questions

Open questions remain. Firstly, we question the role of the budget in games that are not smooth. Secondly, we question the role of the potential budget in broader types of potential games. In Section 6.9 we considered the possibility of an inherent tradeoff between the PoA and the PoS. Furthermore, it seems there may be a link between smooth an potential games, as their similarities are striking. Finding such a connection might link the PoS and PoA upper bounds. For distributed engineering systems, it also remains an open question how sharing information about action sets with local agents might influence the resulting PoA and PoS for resource games. Lastly, Continued research should focus on gaining insight finding better methods to bound the price of stability, as many agent function designs do not compel a potential function.

Bibliography

[1] E. Anshelevich, A. Dasgupta, J. Kleinberg, E. Tardos, Tom Wexler, and Tim Roughgarden. The price of stability for network design with fair cost allocation. 45th Annual IEEE Symposium on Foundations of Computer Science, 2004.

[2] P. G. Balaji and D. Srinivasan. An introduction to multi-agent systems. Studies in Computational Intelligence, 310:1–27, 2010.

[3] Chandra Chekuri, Julia Chuzhoy, Liane Lewin-Eytan, Joseph Naor, and Ariel Orda. Non-cooperative multicast and facility location games. IEEE Journal on Selected Areas in Communications, 25(6):1193–1206, 2007.

[4] Ho-Lin Chen, Tim Roughgarden, and Gregory Valiant. Designing Network Protocols for Good Equilibria. SIAM Journal on Computing, 39(5):1799–1832, 2010.

[5] Constantinos Daskalakis, Paul W. Goldberg, and Christos H. Papadimitriou. The complexity of computing a Nash equilibrium. Communications of the ACM, 52(2):89, 2009.

[6] Shahar Dobzinski, Aranyak Mehta, Tim Roughgarden, and Mukund Sundararajan. Is shapley cost sharing optimal? Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 4997 LNCS:327–336, 2008.

[7] Simon Gächter. Behavioral Game Theory. In Blackwell Handbook of Judgment and Decision Making, pages 485–503. Blackwell Publishing Ltd, 2008.

[8] Itzhak Gilboa and Eitan Zemel. Nash and correlated equilibria: Some complexity considerations, 1989.

[9] MO Jackson. A Brief Introduction to the Basics of Game Theory. Social Science Research Network, pages 1–21, 2011.

[10] R Johari and J Tsitsiklis. A game theoretic view of efficiency loss in resource allocation. Advances in Control, Communication Networks, and Transportation Systems, pages 203–223, 2005.

[11] Elias Koutsoupias and Christos Papadimitriou. Worst-case equilibria. Computer Science Review, 3(2):65–69, 2009.

[12] Jason R. Marden and Michelle Effros. The price of selfishness in network coding. Information Theory, IEEE Transactions on, 58(4):2349–2361, 2012.

[13] Jason R. Marden, Shalom D. Ruben, and Lucy Y. Pao. A model-free approach to wind farm control using game theoretic methods. IEEE Transactions on Control Systems Technology, 21(4):1207–1214, 2013.

[14] István Mezo and Ayhan Dil. Hyperharmonic series involving Hurwitz zeta function. Journal of Number Theory, 130(2):360–369, 2010.

[15] Dov Monderer and Lloyd S. Shapley. Potential Games. Games and Economic Behavior, 14(1):124–143, 1996.

[16] John F Nash. Non coperative games.pdf. PhD thesis, Princeton University, 1950.

[17] John F Nash. Non-Cooperative Games. Annals of Mathematics, 54:286–295, 1951.

[18] Simon Parsons and Michael Wooldridge. Game theory and decision theory in multi-agent systems, 2002.

[19] Parag C. Pendharkar. Game theoretical applications for multi-agent systems. Expert Systems with Applications, 39(1):273–279, 2012.

[20] Tim Roughgarden. Intrinsic robustness of the price of anarchy. Proceedings of the 41st annual ACM symposium on . . . , 55(7):116, 2009.

[21] Tim Roughgarden. Algorithmic game theory, volume 53. 2010.

[22] Tim Roughgarden and Éva Tardos. Introduction to the inefficiency of equilibria. Algorithmic Game Theory, pages 443–458, 2007.

[23] Larry Samuelson. Evolution and Game Theory, 2002.

[24] Andreas S Schulz and Nicolás E Stier-Moses. On the performance of user equilibria in traffic networks. Proceedings of the fourteenth annual ACMSIAM symposium on Discrete algorithms, (Wardrop 1952):86–87, 2003.

[25] Yassmin Shalaby. Minimizing Price of Anarchy in Resource Allocation Games. PhD thesis, 2014.

[26] Theodore L Turocy and Bernhard Von Stengel. Game Theory. Encyclopedia of Information Systems, pages 403 – 420, 2003.

Appendix A

Additional PoA Bounds

A.1 Repeated Play and No-Regret Sequences

Consider a cost minimization game G. Taking into account the budget condition, the extension theorem applies to certain outcome sequences generated by repeated play as well (*see* discussion on learning algorithms in Section 2.3). Using the illustrated point in [20], consider a sequence $(a^1, a^2, \dots a^T)$ of outcomes of a (λ, μ)-smooth game with minimum cost action a^* of the game. For each i and t define

$$\delta_i(a^t) = f_i(a^t) - f_i(a_i^*, a_{-i}^t) \tag{A.1}$$

as the hypothetical improvement of player i's cost at time t, had they used strategy a_i^* instead of a_i^t. We can produce a similar derivation as is done in [20], taking into the budget condition, as follows:

$$C(a^t) \leq B(G) \cdot \sum_{i=1}^{k} J_i(a^t) \tag{A.2}$$

$$= B(G) \sum_{i=1}^{k} \left(J_i(a_i^*, a_{-i}^t) + \delta_i(a^t) \right) \tag{A.3}$$

$$\leq \lambda B(G) \cdot C(s^*) + \mu B(G) \cdot C(a^t) + B(G) \sum_{i=1}^{k} \delta_i(a^t) \tag{A.4}$$

$$\Rightarrow C(a^t) \leq \frac{\lambda B(G)}{1 - \mu B(G)} C(a^*) + \frac{B(G)}{1 - \mu B(G)} \sum_{i=1}^{k} \delta_i(a^t) \tag{A.5}$$

for each t.

For action sequences where each player experiences vanishing average (external) regret, the cost over time is asymptotically competitive with that of every time-invariant strategy:

$$\frac{1}{T}\sum_{t=1}^{T} J_i(a_i^t) \leq \frac{1}{T}\left[\min \sum_{t=1}^{T} J_i(a_i', a_{-i}^t)\right] + o(1) \tag{A.6}$$

Theorem 29. *For every general cost minimization game G that is smooth with robust PoA $\rho(G)$ that incorporates the budget condition $B(G)$, every outcome sequence a^1, ..., a^T that satisfies Eq. (A.6) for every player, and every outcome a^* of G,*

$$\frac{1}{T}\sum_{t=1}^{T} C(\boldsymbol{a}^t) \leq \left[\frac{\lambda B(G)}{1-\mu B(G)} + o(1)\right] \cdot C(a^*) \tag{A.7}$$

as $T \to \infty$.

A similar theorem (omitted) for global finite games with a global welfare function exists and is left as an exercise for the reader.

Appendix B

PoS Bound for Potential Games: Original Theorem

This theorem (referred to from Section 3.4.1) was originally presented in [21].

Theorem 30. *Let a cost minimization game* $G = \{N, A, \{J_i\}_{i \in N}, C\}$ *be a potential game with potential function* ϕ*, and suppose that for any action* $a \in A$*,*

$$c_1 \cdot C(a) \leq \phi(a) \leq c_2 \cdot C(a) \tag{B.1}$$

for some constants $c_1, c_2 > 0$*. Then the Price of Stability for this game is at most* c_2/c_1*.*

Appendix C

Additional Properties of Smooth Games

This appendix section provides an additional property on smooth games that was suggested from Section 3.2. Here we address the following question: if a cost minimization game is (λ, μ)-smooth, are there any additional pairs of smoothness parameters that ensure the game is smooth? While we recognize that other smoothness parameters may loosen the PoA bounds implied from Theorem 2, we still provide this as a tool that might be utilized when designing player cost functions.

Proposition 1. *Let a cost minimization game, $G = \{N, A, \{J_i\}_{i \in N}, C\}$, be (λ, μ)-smooth with $C(a) \geq 0$ for all $a \in A$. Then the game, G, is also $(\lambda + \delta, \mu + \varepsilon)$-smooth for $\delta, \varepsilon \geq 0$.*

Proof. Let a cost minimization game, $G = \{N, A, \{J_i\}_{i \in N}, C\}$, be (λ, μ)-smooth with $C(a) \geq 0$ for all $a \in A$. By the definition of a smooth game (Definition 7) we have:

$$\sum_{i \in N} J_i(a_i^*, a_{-i}) \leq \lambda \cdot C(a^*) + \mu \cdot C(a) \tag{C.1}$$

$$\leq \lambda \cdot C(a^*) + \delta \cdot C(a^*) + \mu \cdot C(a) + \varepsilon \cdot C(a) \tag{C.2}$$

$$= (\lambda + \delta) \cdot C(a^*) + (\mu + \varepsilon) \cdot C(a) \tag{C.3}$$

for all $a, a^* \in A$. Comparing Eq. (C.3) to Eq. (3.1), we see that the game is also $(\lambda + \delta, \mu + \varepsilon)$-smooth. □

Appendix D

General Remark

We add a theorem here to help understand the results of Theorem 16.

Theorem 31. *For integer* $N \geq 1$*,*

$$\frac{1}{n^d - (n-1)^d} = n^{1-d} \tag{D.1}$$

for $d = 1$*, and*

$$\frac{1}{n^d - (n-1)^d} > n^{1-d} \tag{D.2}$$

for $d \in [0, 1)$*.*

Proof. Let n be an integer greater than or equal to one. Let $d = 1$. Then

$$\frac{1}{n^d - (n-1)^d} = \frac{1}{n - (n-1)} \tag{D.3}$$

$$= 1 \tag{D.4}$$

$$= n^{1-1} \tag{D.5}$$

$$= n^{1-d} \tag{D.6}$$

This proves the first part of the theorem. Henceforth, let $d \in [0, 1)$. For $N = 1$ it is a trivial solution. Henceforth, let n be an integer greater than one. For $d = 1 - \varepsilon$ for $\varepsilon \to 0^+$, we see that

$$\frac{n^{1-d}}{n^d - (n-1)^d} \to 1^+ \tag{D.7}$$

A derivative of $\frac{n^{1-d}}{n^d-(n-1)^d}$ with respect to d shows that $\frac{n^{1-d}}{n^d-(n-1)^d}$ has a negative slope on the interval $d \in [0, 1)$. Hence, since $\frac{n^{1-d}}{n^d-(n-1)^d}$ is continuous on this same interval, the second half of the theorem also holds. $\square$