Resource pooling and cost allocation among independent service providers

Beta
Research School for Operations
Management and Logistics

Resource pooling and cost allocation among independent service providers

Frank Karsten, Marco Slikker, Geert-Jan van Houtum

Beta Working Paper series 352

BETA publicatie	WP 352 (working paper)
ISBN	
ISSN	
NUR	982
Eindhoven	August 2011

Resource pooling and cost allocation among independent service providers

Frank Karsten[*], Marco Slikker, Geert-Jan van Houtum
School of Industrial Engineering, Eindhoven University of Technology,
P.O. Box 513, 5600 MB, Eindhoven, The Netherlands

August 1, 2011

Abstract

We study a situation where several independent service providers collaborate by complete pooling of their resources and customer streams into a joint service system. These service providers may represent such diverse organizations as hospitals that pool beds, call centers that share telephone operators, or maintenance firms that pool repairmen. We model the service systems as Erlang delay systems (M/M/s queues) that face a fixed cost rate per server and homogeneous delay costs for waiting customers. We examine rules to fairly allocate the collective costs of the pooled system amongst the participants by applying concepts from cooperative game theory. We consider both the case where players' numbers of servers are exogenously given and the scenario where any coalition picks an optimal number of servers. By exploiting new analytical properties of the continuous extension of the classic Erlang delay function, we provide sufficient conditions for the games under consideration to possess a core allocation (i.e., an allocation that gives no group of players an incentive to split off and form a separate pool) and to admit a population monotonic allocation scheme (whereby adding extra players does not make anyone worse off). This is not guaranteed in general, as illustrated via examples.

Keywords: game theory, queuing theory, service operations.

[*] Corresponding author. Tel.: +31-40-247-4432. E-mail address: f.j.p.karsten@tue.nl.

1 Introduction

Resource pooling is an efficient strategy for dealing with uncertainty in service industries. It refers to an arrangement in which a group of common resources or servers is held for multiple customer streams rather than dedicated, separate resources for each individual customer stream. The main benefit of resource pooling is reduced congestion, as measured by the time spent by customers waiting to be served (Smith and Whitt, 1981). This reduction occurs because with service systems working separately a customer may have to wait for one server while another server is idle — a situation that does not occur in the pooled system. The efficiency benefits of resource pooling are commonly exploited in case multiple customer streams are served by one common service provider. But these benefits can also be obtained if the customer streams belong to several independent service providers.

There are numerous real-life examples in various sectors of independent service providers who may collaborate by pooling their resources into a joint service system. For instance, several manufacturers of advanced technical equipment may employ a number of non-branded repairmen to maintain and repair machines at their customer's sites. Similarly, business units of a large insurance firm may operate a common call center with cross-trained telephone agents. One can also think of airline companies pooling check-in counters. Further, a hospital is often comprised of clinical departments that share operating rooms, hospital beds, and medical staff. Another example is found in manufacturing facilities, where flexible production equipment is shared between several job types. As a final example, consider a number of university faculties that are empowered to make independent decisions. They may collaborate by setting up a common computer cluster and obtain resource pooling benefits. On top of this, they might also be able to buy their ICT-systems at a reduced price due to increased bargaining power. In general, collaboration among service providers enables more efficient use of their resources, offers the opportunity to benefit from large economies of scale, and enhances their negotiation power: benefits aplenty!

But how should these independent entities allocate the total costs of the pooled system among them? A fair cost division is an essential prerequisite for a successful cooperation, but the construction of such an allocation tends to be challenging — in fact, it often is a severe impediment for cooperation (Cruijssen et al., 2007). Cooperative game theory offers a natural paradigm to tackle this problem. In this paradigm, participants or players draw up binding agreements and make side payments to each other. A main notion of fairness from cooperative game theory is the core, which is the set of all stable allocations of the joint costs that give no group of players an incentive to secede and act separately. Under such a stable allocation, each player will feel motivated to collaborate; indeed, no group of players is paying to subsidize the others.

In this paper, we consider cooperative games where an arbitrary number of service providers, the players, face exogenous Poisson streams of customer arrivals and are allowed to collaborate by completely sharing their servers and individual customer streams. We model the service system of any coalition as an Erlang delay system, i.e., an M/M/$s(/\infty)$ queue. Costs consist of linear resource costs for servers and linear delay costs for customers that have to wait before being served. Our modeling approach differs from the approach taken in most of the previous work on cooperative queueing games (e.g., González and Herrero, 2004; Anily and Haviv, 2010). These authors consider cooperative games where each coalition operates an M/M/1 queue. Although such a model is applicable if service capacity can be easily consolidated into a single server (e.g., by choice of material or technology), the M/M/1 model is not appropriate when service facilities consist of *multiple* servers whose service speeds are given, as in all our real-life examples mentioned above. Despite that, some of these examples have been previously used to motivate the study of M/M/1 games: for instance, González and Herrero (2004) are motivated by shared medical services, and Anily and Haviv (2010) mention pooled service technicians in their introduction. By more accurately modeling these settings as M/M/s queues, we obtain more precise results and insights for these settings. Our analysis reveals fundamental differences between cooperative behavior in M/M/1 and M/M/s contexts (see Sections 4.3 and 5.3) and expands our understanding of resource sharing in queueing systems.

We distinguish two cases: fixed numbers of servers and optimal numbers of servers. In the former case, each player possesses a predetermined number of servers, which he brings to any coalition. This setting captures short-term collaborations that arise from an existing situation wherein adjustment of the number of servers is prohibitively expensive or practically unfeasible. In the latter case, each coalition picks a cost-minimizing number of servers. This setting is appropriate when parties are setting up a new, long-term collaborative project. It is also applicable for existing situations if the number of servers can be easily adjusted against negligible costs. In both cases, we assume that customers are homogenous in waiting time costs and in service requirements across players (in line with Anily and Haviv, 2010). Altogether, our analysis of cooperative games for both cases provides insights for a variety of practically relevant collaborative arrangements.

We mainly focus on the existence of stable allocations and, for settings where a stable allocation exists, on the selection of an appropriate, transparant allocation mechanism. Studying existence is important because the core of a game may be empty in general, even if collaboration leads to an overall cost reduction. We investigate under what conditions a core allocation is guaranteed to exist for our queueing games. To deal with the possible multiplicity of stable allocations, we consider the refinement of a population monotonic allocation scheme. An allocation *scheme* deals with partial cooperation: it proposes a cost division not only for the grand coalition of all players but also for every possible subcoalition. Such a scheme is called population monotonic (cf. Sprumont, 1990) if no member of any coalition is assigned more costs after an extra player joins in. We will investigate whether core allocations can be reached through a population monotonic allocation scheme (PMAS).

The main contributions and results of this paper are as follows:

- To the best of our knowledge, we are the first to introduce the class of cooperative games arising from resource pooling in *multi-server* queueing systems where the total number of servers is exogenously determined, and we are the first to consider the exact problem for the case with optimal numbers of servers in detail. We derive new insights which differ from the ones previously obtained for M/M/1 games.

- For the case with a fixed number of servers, we prove that cooperation is always beneficial and supportable by (infinitely many) stable cost allocations. Counterexamples indicate that, in general, the cooperative games in this setting may lack a PMAS and, moreover, some players may be assigned a negative cost in every core allocation. Nevertheless, under a natural assumption on the ratios between players' servers and arrival rates, we identify a simple, positive, proportional core allocation that can be reached through a PMAS.

- For the case in which each coalition picks an optimal number of servers, we show that the existence of a stable allocation and a PMAS is dependent on the domain over which optimization takes place. If each coalition is required to choose an *integer* number of servers, this existence is not guaranteed. But by comparison to a relaxed problem, we show that this nonexistence is purely attributable to the integrality requirement. We introduce approximate core and PMAS concepts, which may be relevant beyond the context of this study, to describe an upper bound on the impact of integrality.

- To obtain these structural results, we derive several new analytical properties of the (standard) continuous extension of the classic Erlang delay function. As a side benefit, our approach generalizes and strengthens well-known characteristics of key performance measures in the M/M/s model.

The remainder of this paper is organized as follows. We start in Section 2 with a brief review of related literature. Then, in Section 3, we provide some preliminaries on cooperative game theory and the continuous extension of the Erlang delay function. In Section 4, we analyze the case in which the number of servers is fixed. In Section 5, we treat the case with optimal numbers of servers. Finally, we draw conclusions and suggest directions for future research in Section 6. All proofs are given in the Appendix.

2 Related literature

There is a rich literature on resource pooling in queueing systems. Smith and Whitt (1981) were the first to prove that sharing the servers of multiple Erlang delay systems with identical service time distributions into an aggregate system is always beneficial. Calabrese (1992) found that combining servers into larger groups, while keeping server utilization constant, leads to reduced congestion, and Benjaafar (1995) provides performance bounds on the effectiveness of resource pooling. These studies assume that the system is owned by a single entity who decides whether or not to pool. In contrast, the present paper considers resource pooling arrangements between independent service providers, each with their own interests, and explicitly addresses the issue of fair cost allocation.

There are several papers that apply cooperative game theory to analyze resource pooling in queueing facilities. The papers in this stream of research can be classified along several dimensions. Tables 1 and 2 position the existing literature according to various modeling assumptions. We first review the previous literature in the area of *single*-server queueing games and compare our research to this previous work. Afterwards, we will discuss the existing research on *multi*-server queueing games.

In single-server queuing games with optimal service capacity, each player is associated with a customer arrival stream and any coalition of players operates an M/M/1 queue that serves the union of its members' arrival streams. González and Herrero (2004) deal with the problem of fairly allocating the cost of the server, which is assumed to be proportional to its service rate. In their model, each coalition optimally chooses the service rate such that certain exogenously given constraints on customers' mean sojourn time are satisfied. García-Sanz et al. (2008) analyze three variations of the model in González and Herrero (2004): they allow more generic sojourn time constraints, consider constraints on the mean waiting time in the queue, and investigate a preemptive priority queueing discipline. Yu et al. (2009) enrich the setting by introducing delay costs and consider a game where coalitions optimize their service rate to minimize the sum of delay and capacity costs. All

	Optimal service capacity	Fixed service capacity
Waiting in queue	González and Herrero (2004) García-Sanz et al. (2008) Yu et al. (2009)	Anily and Haviv (2010) Timmer and Scheinhardt (2010) Anily and Haviv (2011)

Table 1: *Classification of literature on **single**-server cooperative queueing games*

	Optimal number of servers	Fixed number of servers
Waiting in queue	Yu et al. (2007) This paper	This paper
No waiting allowed	Karsten et al. (2011) Özen et al. (2011)	Karsten et al. (2009)

Table 2: *Classification of literature on **multi**-server cooperative queueing games*

papers in this stream of literature assume that a coalition picks an optimal service rate, which can be viewed as the analogue of our case with optimal numbers of servers.

Conversely, Anily and Haviv (2010) study a model in which each player has its own capacity endowment, modeled as a potential service rate. To reduce congestion costs, each coalition may cooperate by pooling these endowments and their individual customer streams into a single M/M/1 system whose service rate is the sum of the potential service rates of its members. This setting parallels our case with fixed numbers of servers. Timmer and Scheinhardt (2010) and Anily and Haviv (2011) analyze models with several single-server stations in a certain network structure. In these models, the network structure is kept intact (as opposed to pooling capacities into a single station) and the total network capacity is predetermined; the various stations may cooperate by redistributing their combined service capacity or by re-routing arrivals, resulting in a network of M/M/1 queues.

In contrast to the cooperative games associated with M/M/1 queues, our work considers service systems with multiple servers in parallel. Inspired by the real-life examples described in the introduction where servers represent human operators, we assume that a server's speed is fixed rather than variable. At first glance, an M/M/s system where each server provides service at a rate μ may seem to behave similar to an M/M/1 system with service rate $s\mu$. Indeed, as long as all servers are busy, the group of customers as a whole is served at the same rate in both systems. However, if less than s customers are present, the single-server system can use its total service capacity, whereas the multi-server system has idle capacity.[1] Thus, the behavior of the two systems is fundamentally different. A second difference is that the number of servers (i.e., total capacity in a multi-server system) can typically only be varied in discrete amounts, a limitation that is absent in single-server systems where the service speed can be set at arbitrary levels.

Other previous literature has tackled multi-server cooperative queueing games, both for systems where waiting is allowed and for systems where waiting is not possible. For the latter setting, Karsten et al. (2009, 2011) study situations where several independent service providers collaborate by pooling their resources into a joint service system for, respectively, the case with fixed numbers of servers and the case with optimal numbers of servers. In these two papers, the service facilities are modeled as Erlang loss systems in which customers that find no free server upon arrival are lost and redirected elsewhere, which differs — both from a modeling and an application perspective — from the setting considered in the present paper where waiting in a queue is allowed. Özen et al. (2011) independently derived the same conclusion as Karsten et al. (2011): games corresponding to server optimization in Erlang loss systems have a nonempty core. Karsten et al. (2011) derive structural properties of a novel extension of the Erlang loss function to arrive at this conclusion, whereas Özen et al. (2011) posit a new framework of single-attribute games to derive this result.

Shifting our attention to multi-server settings where waiting is allowed, we remark that Özen et al. (2011) also used their framework to study the core of various cooperative games arising from a given[2] Erlang delay system wherein cooperating parties optimize the service rate or the amount of demand to serve, and each coalition uses the same amount of servers. The games considered in the present paper are fundamentally different: we consider exogenously given service speeds and arrival rates, and we allow a coalition to possibly optimize the number of servers instead. Finally, we mention that Yu et al. (2007), a previous version of Yu et al. (2009), briefly considers a setting similar to ours where each coalition operates an M/M/s queue with an adjustable number of servers. They only show nonemptiness of the core in a heavy traffic limit under the assumption that the number of servers is chosen via the square-root safety staffing principe, a close-to-optimal rule of

[1] To allow a formal comparison, we fix the arrival rate λ; additionally, we set $x = 1/(s\mu - \lambda)$ and $a = \lambda/\mu$. Then, consider the well-known expressions for expected waiting time: $x \cdot a/s$ in the single-server system with rate $s\mu$ and $x \cdot \hat{C}(s, a)$ in the s-server system with rate μ, with $\hat{C}(s, a)$ the Erlang delay function (see Section 3.2). It is easily verified that a/s is a grossly inaccurate approximation of $\hat{C}(s, a)$, especially when the number of servers s is large.

[2] Özen et al. (2011) assumed in this particular model that each coalition, irrespective of its size, uses the *same* number of servers. This differs significantly from the setting in which coalitions combine the servers of their members; hence, Özen et al. (2011) is not classified under "fixed number of servers" in Table 2.

thumb, whereas we are interested in the exact optimization problem.

3 Preliminaries

For reasons of self-containedness, we introduce in this section several concepts from cooperative game theory that are relevant to our work. Subsequently, we present the continuous extension of the classic Erlang delay function and derive several of its properties.

3.1 Cooperative game theory

Let N be a nonempty finite set of *players*. A subset $M \subseteq N$ is called a coalition, and the set N of all players is referred to as the *grand coalition*. We let $2^N_- = \{M \subseteq N \mid M \neq \emptyset\}$ denote the power set consisting of all nonempty coalitions. For any two sets M and L, we write $M \subset L$ if M is a *proper subset* of L, i.e., if $M \subseteq L$ and $M \neq L$. The function c that assigns to every coalition $M \subseteq N$ its costs $c(M)$ is called the *characteristic cost function*. The value $c(M)$ is interpreted as the total costs of the joint cooperative effort if only the players in M are involved in it. By convention, $c(\emptyset) = 0$. We assume that the costs of any coalition M are freely transferable between the players of M. Then, the pair (N, c) constitutes a cooperative cost game with transferable utility. In the remainder, we will simply refer to this as a *game*.

An interesting property that a game might satisfy is (strict) subadditivity. A game is called *subadditive* if it is always beneficial to combine coalitions, i.e., if for any two coalitions $M, L \subseteq N$ with $M \cap L = \emptyset$ it holds that $c(M) + c(L) \geq c(M \cup L)$. If this inequality is strict for each two disjoint *nonempty* coalitions M and L, we call the game *strictly subadditive*. In a subadditive game, cooperation by the grand coalition is socially optimal. In a strictly subadditive game, each other partition is worse.

A central problem in cooperative game theory is to allocate $c(N)$ to the individual players in a fair way. Formally, an *allocation* for a game (N, c) is a vector $x = (x_i)_{i \in N} \in \mathbb{R}^N$ satisfying $\sum_{i \in N} x_i = c(N)$. The latter requirement is often called *efficiency*. The value x_i is interpreted as the costs assigned to player i. Two well-known allocation rules are the *Shapley value* (Shapley, 1953) and the *nucleolus* (Schmeidler, 1969). The Shapley value Φ of game (N, c) is defined, for all players $i \in N$, by

$$\Phi_i(N, c) = \sum_{M \subseteq N : i \in M} \frac{(|M| - 1)!(|N| - |M|)!}{|N|!} \cdot [c(M) - c(M \setminus \{i\})].$$

An allocation x for a game (N, c) is called *stable* if $\sum_{i \in M} x_i \leq c(M)$ for all $M \in 2^N_-$.

Under a stable allocation, each group of players has to pay no more collectively than what they would face by acting independently. Hence, if the costs of the grand coalition are assigned according to a stable allocation, no coalition has an incentive to split off and establish cooperation on its own. The (convex) set of all stable allocations is called the *core*, introduced by Gillies (1959). The core of a game may be empty, even if the game is subadditive. The nucleolus always results in a core element whenever the core is nonempty, but the Shapley value does not. One class of games for which the Shapley value is guaranteed to be in the core, however, is the class of concave games (Shapley, 1971). A game is called *concave* if any player's marginal cost contribution is smaller for large coalitions, i.e., if for each $i \in N$ and for all $M, L \subseteq N \setminus \{i\}$ with $M \subseteq L$ it holds that $c(M \cup \{i\}) - c(M) \geq c(L \cup \{i\}) - c(L)$.

The last concept that we wish to introduce is a population monotonic allocation scheme (cf. Sprumont, 1990). An allocation scheme for a game (N, c) is a vector $y = (y_{i,M})_{i \in M, M \in 2^N_-}$, with $\sum_{i \in M} y_{i,M} = c(M)$ for all coalitions $M \in 2^N_-$, which specifies how to allocate the costs of every coalition to its members. This scheme is called a *population monotonic allocation scheme* (PMAS) if the amount that a player has to pay does not increase when the coalition to which he belongs grows. That is, $y_{i,M} \geq y_{i,L}$ for all coalitions $M, L \in 2^N_-$ with $M \subseteq L$ and $i \in M$. If a game (N, c) admits a PMAS, say y, then its core is nonempty, $(y_{i,N})_{i \in N}$ is an element of its core, and for each nonempty coalition $L \in 2^N_-$ the *sub-game* (L, c^L), where $c^L(M) = c(M)$ for all $M \subseteq L$, has a nonempty core.

3.2 New properties of the continuous extension of the Erlang delay function

Consider an Erlang delay system, i.e., an M/M/s queue. In such a system, customers arrive according to a Poisson process with rate $\lambda > 0$. They are served by a group of $s \in \mathbb{N}$ homogeneous parallel servers. Service times are independent and exponentially distributed with rate $\mu > 0$. Customers who find all servers busy wait in an infinite capacity queue until served by the first available server. We let $a = \lambda/\mu$ denote the offered load.

The steady-state probability of delay (the probability that an arrival must wait before beginning service) in such a system is described by the classic *Erlang delay function*, first published by Erlang (1917). This function is defined, for each $a > 0$ and $s \in \mathbb{N}$ with $s > a$ (to guarantee stability of the queueing system), by

$$\hat{C}(s, a) = \left(1 + \sum_{y=0}^{s-1} \frac{s!(1 - a/s)}{y! a^{s-y}}\right)^{-1}. \tag{1}$$

Another interesting performance measure, also derived by Erlang (1917), is the expected waiting time (delay before beginning service) experienced by an arbitrary customer in steady state. For any $\lambda > 0$, $\mu > 0$, and $s \in \mathbb{N}$ with $s > \lambda/\mu$, this waiting time equals

$$\hat{W}_q(s, \lambda, \mu) = \frac{\hat{C}(s, \lambda/\mu)}{s\mu - \lambda}. \tag{2}$$

Equations (1) and (2) are valid for any *non-biased* service discipline, i.e., a service discipline that selects the next customer to be served without taking the waiting customers' actual service lengths into account (cf. Cooper, 1981, pp. 95–98). Examples of non-biased service disciplines are service on a first-come first-serve basis, service in random order, or service on a last-come first-serve basis.

For analytical purposes, it will be convenient to extend the domain of the Erlang delay function to non-integral values of s. Jagers and Van Doorn (1991) have suggested a confluent hypergeometric function as a natural continuous extension. This function is defined, for each $a > 0$ and $s \in \mathbb{R}$ with $s > a$, by

$$C(s, a) = \left(\int_0^\infty a e^{-ax} (1 + x)^{s-1} x dx \right)^{-1}. \tag{3}$$

For fixed $a > 0$, $C(s, a)$ is non-increasing and convex in s for $s \in \mathbb{R}$ (Jagers and Van Doorn, 1991). This analytic extension of the Erlang delay function enables a natural way to define the expected waiting time in an (artificial) queueing system with a non-integral number of servers: for any $\lambda > 0$, $\mu > 0$, and $s \in \mathbb{R}$ with $s > \lambda/\mu$, we define

$$W_q(s, \lambda, \mu) = \frac{C(s, \lambda/\mu)}{s\mu - \lambda}. \tag{4}$$

As observed by Jagers and Van Doorn (1991), Equations (1) and (3) coincide for integer values of s, i.e., $\hat{C}(s, a) = C(s, a)$ for all $s \in \mathbb{N}$ and $a \in (0, s)$. Accordingly, Equations (2) and (4) coincide for those cases as well.

The performance measures described above satisfy various interesting structural properties. The literature dealing with these properties is rich (an excellent overview is provided in Whitt, 2002), but most research has focused on $\hat{C}$ and $\hat{W}_q$, thereby restricting the analysis to integer numbers of servers. In what follows, we will show that various well-known monotonicity, convexity, subadditivity, and other properties of $\hat{C}$ and $\hat{W}_q$ are also valid for C and W_q. Thus, we extend the analysis to non-integral numbers of servers by means of (3).

But given that all real-life queueing systems operate under an integral number of servers, why the fuss of this extended analysis? First there is the mathematical appeal of a generalization of known results; in fact, our analysis of the continuous extension (3) will provide simple alternative proofs of classic results in the M/M/s model. But more importantly, the ensuing properties of the continuous extensions C and W_q will allow us to derive interesting results for queueing games.

To obtain new structural results for the extensions C and W_q, we exploit a relation between the continuous extension of the Erlang delay function and the continuous extension of the Erlang *loss* function (for the M/G/s/s model), and we use a result that has already been established for the latter. Following Jagers and Van Doorn (1991), the continuous extension of the classic Erlang loss function is defined for any $s \in [0, \infty)$ and $a > 0$ by

$$B(s,a) = \left(\int_0^\infty a e^{-ax} (1+x)^s dx \right)^{-1}. \tag{5}$$

The following lemma shows that the Erlang delay function can be expressed in terms of the Erlang loss function, and vice versa. For integer s, this relation is well known (see, e.g., Cooper, 1981, p. 92). It appears that this relation remains valid for the continuous extensions (3) and (5).

Lemma 3.1. *Let $a > 0$ and $s \in \mathbb{R}$ with $s > a$. Then,*

$$C(s,a) = \frac{B(s,a)}{1 - (a/s)(1 - B(s,a))}.$$

The proof of this and subsequent results is given in the Appendix. Next, we show that when the load per server is held constant, the probability of delay is decreased by adding servers. (We will use, throughout, "decreasing" in the strict sense.) For integer s, this has already been proven by Calabrese (1992, Proposition 1).

Lemma 3.2. *Fix $a > 0$ and $s \in \mathbb{R}$ with $s > a$. Then, $C(ts, ta)$ is decreasing in t for $t > 0$.*

The following theorem states that when the load per server is held constant again, the expected waiting time is decreased by adding servers. Benjaafar (1995, p. 377) provides a proof of this result for integer s.

Theorem 3.3. *Fix $\lambda, \mu > 0$ and $s \in \mathbb{R}$ with $s > \lambda/\mu$. Then, $W_q(ts, t\lambda, \mu)$ is decreasing in t for $t > 0$.*

The following theorem says that the expected waiting time is decreasing and strictly convex in the number of servers. For integer s, these properties have already been proven by Dyer and Proll (1977).

Theorem 3.4. *Let $\lambda, \mu > 0$. Then, $W_q(s, \lambda, \mu)$ is a decreasing and strictly convex function of s for $s \in \mathbb{R}$ with $s > \lambda/\mu$.*

We conclude this section with a subadditivity property that describes the economy-of-scale effect associated with larger service systems. Specifically, the following theorem says that combining two separate M/M/s queues with common service rates into a joint system will lead to a reduction in the average (per-arrival) delay. Smith and Whitt (1981) provide a proof of this result for integer s, although not with strict inequality.

Theorem 3.5. *Let $\lambda_1, \lambda_2, \mu > 0$. Then, for all $s_1 \in \mathbb{R}$ with $s_1 > \lambda_1/\mu$ and for all $s_2 \in \mathbb{R}$ with $s_2 > \lambda_2/\mu$, it holds that*

$$W_q(s_1 + s_2, \lambda_1 + \lambda_2, \mu) \cdot (\lambda_1 + \lambda_2) < W_q(s_1, \lambda_1, \mu) \cdot \lambda_1 + W_q(s_2, \lambda_2, \mu) \cdot \lambda_2.$$

4 Fixed numbers of servers

In this section, we consider a setting in which each player brings a predetermined number of servers to any coalition. This is a reasonable model for situations where adjusting the number of servers is too expensive or practically impossible. We first introduce the situation in more detail and define the associated game. Subsequently, we analyze structural properties of this game and identify stable and population monotonic cost allocations.

4.1 Situation

Consider several service organizations, which we will simply refer to as players. Each player witnesses a Poisson arrival process of customers, and the arrival processes of the players are assumed to be independent. Each player has an exogenously given number of servers to provide service to their customer streams. The number of servers cannot be easily adjusted and is therefore considered to be fixed. Service times for an arbitrary customer of any player are independent and identically exponentially distributed. Customers who find all servers busy upon arrival wait in a queue, incurring delay costs that are proportional to their waiting time. These delay costs, which are symmetrical across players, represent customer dissatisfaction, lost goodwill, and/or contractual penalties; they are borne by the player to whom the customer belongs.

Players are interested in their long-term average costs per unit time, which they can reduce by collaborating, i.e., pooling their resources to serve their customer streams together. Our aim is to determine (existence of) fair allocations of costs to support the collaboration. To analyze this, we formally define a multi-server queueing situation with a fixed number of servers (FIX-queueing situation for short) as a tuple $(N, (\lambda_i)_{i \in N}, \mu, (s_i)_{i \in N}, h, d)$, where

- N is the nonempty finite set of players;
- $\lambda_i > 0$ is the arrival rate of customers that belong to player $i \in N$;
- $\mu > 0$ is the rate of the exponential service time distribution;
- $s_i > 0$ is the amount of servers that player $i \in N$ brings to any coalition[3];
- $h \geq 0$ is the resource cost incurred for each server per unit time;
- $d > 0$ is the delay cost incurred by any customer for waiting one unit of time in the queue.

With Ψ we denote the set of such situations for which $s_i > \lambda_i/\mu$ for all $i \in N$, i.e., for which each player possesses enough servers to ensure that the expected waiting time in his own service facility is finite[4]. For each coalition $M \in 2^N_-$, we denote $\lambda_M = \sum_{i \in M} \lambda_i$ and $s_M = \sum_{i \in M} s_i$.

This model is sufficiently general to cover a wide variety of situations in which a resource pooling arrangement can arise between independent service providers that already have existing facilities. Our model is simple, yet it has all the necessary ingredients to capture a concrete setting. To illustrate this, we recall the real-life medical example described in the introduction. Modeling this health-care context as a FIX-queueing situation, we can let the players correspond to clinical departments in a hospital, each with their own patient arrival streams. The servers can be represented by hospital beds; the amount of beds is fixed for the duration of the envisioned collaboration. Service time corresponds to a patient's length of stay. Finally, maintenance and capital costs for beds represent the resource costs, and legal regulations and governmental fines determine the delay costs.

4.2 Game

Consider any FIX-queueing situation $\psi = (N, (\lambda_i)_{i \in N}, \mu, (s_i)_{i \in N}, h, d) \in \Psi$ and an arbitrary coalition $M \in 2^N_-$. The players in this coalition collaborate by complete pooling of their

[3] Although this situation only has a natural interpretation when each player has an integer number of servers, our formulation does allow a player to possess a non-integral number of servers. While we could have restricted ourselves to situations with integral numbers of servers, we chose to consider the more general setting for convenience (it allows shorter proofs) and for a better fit with Section 5.

[4] This assumption is not essential; it merely allows a clear exposition. Our results would remain valid under the weaker assumption of $s_N > \lambda_N/\mu$, but analysis of possibly unstable queueing systems for some coalitions would require inconvenient notation.

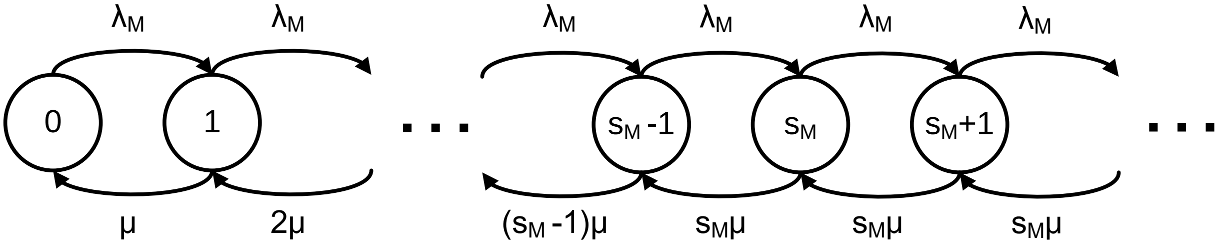

Figure 1: *A Markov chain representation of the joint service facility of coalition M. A state is defined by the number of customers in the system (in service and in queue).*

respective arrival streams and servers into a joint system. Since the superposition of independent Poisson processes is also a Poisson process, this coalition now faces a combined Poisson arrival process with aggregate rate λ_M. The coalition has s_M servers at their disposal. We assume that each server can handle all types of customers with equal ease and that all customers can effortlessly access the joint service facility. A non-biased service discipline, such as service in order of arrival, is used.

Based on these assumptions, the pooled system behaves as an Erlang delay system. Figure 1 illustrates this Markovian queue. The expected waiting time that an arbitrary customer spends in the queue before starting service is equal to $W_q(s_M, \lambda_M, \mu)$. We can now can formulate a game corresponding to FIX-queueing situation ψ. We call the game (N, c^ψ) with

$$c^\psi(M) = hs_M + W_q(s_M, \lambda_M, \mu) \cdot \lambda_M d. \tag{6}$$

for all $M \in 2^N_-$ and $c^\psi(\emptyset) = 0$ the associated *FIX-queueing game*. The first term in (6) represents the (additive) resource cost per unit time faced by coalition M, and the second term corresponds to the delay costs per unit time in steady state.

The following proposition shows that, consistent with intuition, cooperation in the context of resource pooling always leads to a reduction in costs. Its proof is given in the Appendix; it directly follows from Theorem 3.5.

Proposition 4.1. *FIX-queueing games are strictly subadditive.*

Although Proposition 4.1 affirms that collaboration among all players is beneficial, it does not imply the existence of a stable cost allocation or a PMAS. If FIX-queueing games would be concave, existence of both would have been guaranteed. The following example, however, shows that FIX-queueing games need not be concave. The example also illustrates possible allocation rules and shows that the Shapley value is not necessarily in the core.

Coalition M	$\{1\}$	$\{2\}$	$\{3\}$	$\{1,2\}$	$\{1,3\}$	$\{2,3\}$	$\{1,2,3\}$
$W_q(s_M, \lambda_M, \mu)$	1	3	$\frac{1}{1243}$	$\frac{25}{39}$	$\frac{27}{11453}$	$\frac{1}{147}$	$\frac{81}{14077}$
$c^{\psi}(M)$	5	$22\frac{1}{2}$	$\frac{5}{2486}$	$8\frac{1}{78}$	$\frac{405}{22906}$	$\frac{10}{147}$	$\frac{1215}{14077}$

Table 3: *The FIX-queueing game and expected delay times of Example 4.1.*

Example 4.1. Consider the FIX-queueing situation $\psi = (N, (\lambda_i)_{i \in N}, \mu, (s_i)_{i \in N}, h, d) \in \Psi$ with player set $N = \{1, 2, 3\}$, service rate $\mu = 1$, resource cost rate $h = 0$, delay cost rate $d = 10$, and

$$\lambda_1 = 1/2; \quad \lambda_2 = 3/4; \quad \lambda_3 = 1/4;$$
$$s_1 = 1; \qquad s_2 = 1; \qquad s_3 = 3.$$

The characteristic cost function c^{ψ} of the associated FIX-queueing game (N, c^{ψ}) is represented in Table 3, along with the expected waiting time of an arbitrary customer in any coalition's service system.

This game has a nonempty core: for example, the allocation x given by $x_1 = 2$, $x_2 = 3$, and $x_3 = -4\frac{12862}{14077}$ is stable. However, this game is not concave since $c^{\psi}(\{1,2\}) - c^{\psi}(\{2\}) = -14\frac{19}{39} < \frac{5405}{295617} = c^{\phi}(\{1,2,3\}) - c^{\phi}(\{2,3\})$. In other words, player 1's marginal cost contribution may increase if he joins a larger coalition. Accordingly, the game's Shapley value $\Phi(N, c^{\psi})$, which is approximately equal to $\Phi_1(N, c^{\psi}) \approx -0.74$, $\Phi_2(N, c^{\psi}) \approx 8.04$, and $\Phi_3(N, c^{\psi}) \approx -7.21$ (rounded to 2 decimals), is not in the core of this game since $\Phi_2(N, c^{\psi}) + \Phi_3(N, c^{\psi}) > c^{\psi}(\{2,3\})$. $\Diamond$

We remark that the characteristic cost function in the preceding example is not monotonically decreasing. In fact, expected waiting times may increase when a new player joins! Consider coalition $\{2,3\}$ in the example. When player 2 joins player 3, the expected delay experienced by a customer of player 2 reduces from 3 to 1/147. But player 3, on the other hand, observes an increase in expected delays when player 2 joins him; this is because player 3 possesses a relatively large number of servers and, as a result, observes few delays when acting independently. Nevertheless, player 2 can motivate player 3 to collaborate by means of side payments, and the average (per-arrival) delay experienced is lower under a resource pooling arrangement than under an arrangement where both players operate separate systems, in line with Theorem 3.5.

4.3 Cost allocation: stability and population monotonicity

Recall that the FIX-queueing game in Example 4.1 admitted a stable allocation. The following theorem, which presents our general results on the existence and multiplicity of stable cost allocations for FIX-queueing games, shows that this is not a coincidence. In particular, we show that any FIX-queueing game, as well as each of its sub-games, has a stable allocation. Moreover, unless there is only one player, the core is never a singleton. We remark that these properties are also satisfied by M/M/1 games with fixed numbers of servers (cf. Anily and Haviv, 2010).

Theorem 4.2. *Let $\psi = (N, (\lambda_i)_{i \in N}, \mu, (s_i)_{i \in N}, h, d) \in \Psi$ be a FIX-queueing situation.*

(i) The associated game (N, c^{ψ}) and each of its sub-games possess a non-empty core.

(ii) If $|N| > 1$, there are infinitely many core allocations for the game (N, c^{ψ}).

The proof of this theorem is based on the powerful characterization of balanced games, due to Bondareva (1963) and Shapley (1967), and on several properties derived in Section 3.2. Part (i) of Theorem 4.2 implies that the nucleolus is always a stable cost allocation for FIX-queueing games. Since the nucleolus satisfies appealing fairness properties (cf. Snijders, 1995), it would be a suitable method to allocate the total costs of the grand coalition. Nevertheless, computation of the nucleolus may be difficult (see, e.g., Leng and Parlar, 2010). In light of this downside and part (ii) of Theorem 4.2, one may well ask whether the core contains a simple proportional type of cost allocation, e.g., proportional with respect to arrival rates or to numbers of servers. The following example shows that such proportional allocations will not necessarily be in the core, as there are instances in which a player is assigned a negative cost (i.e., a reward) in every core allocation.

Example 4.2. Consider the FIX-queueing game (N, c^{ψ}) of Example 4.1 again. For any allocation x in the core of (N, c^{ψ}), it holds that $x_1 + x_3 \leq c^{\psi}(\{1, 3\})$, $x_2 + x_3 \leq c^{\psi}(\{2, 3\})$, and $x_1 + x_2 + x_3 = c^{\psi}(\{1, 2, 3\})$. Hence,

$$x_3 = x_3 + x_1 + x_2 + x_3 - c^{\psi}(N) \leq c^{\psi}(\{1, 3\}) + c^{\psi}(\{2, 3\}) - c^{\psi}(N) = \frac{405}{22906} + \frac{10}{147} - \frac{1215}{14077} < 0.$$

Thus, player 3 is assigned a negative cost in every core allocation. The intuition behind this is that player 3 should be compensated for the relatively large number of servers that he adds to any coalition. ◊

Interestingly, in the corresponding M/M/1 queueing game (cf. Anily and Haviv, 2010), nonnegative core allocations always existed; thus, in this respect, the multi-server models exhibit different behavior than their single-server counterparts.

We next introduce an allocation scheme under which the expected waiting cost of any coalition is allocated proportional to the arrival rates of its members and, in addition, each player pays the resource costs for its own servers. Under a natural assumption on the ratios between players' servers and arrival rates, this allocation scheme will turn out to be population monotonic. We remark that Anily and Haviv (2010) did not consider population monotonicity or symmetry conditions for their M/M/1 modeling.

Allocation scheme $\mathcal{P}$ for FIX-queueing situation $\psi = (N, (\lambda_i)_{i\in N}, \mu, (s_i)_{i\in N}, h, d) \in \Psi$ is defined, for all $M \in 2^N_-$ and all $i \in M$, by

$$\mathcal{P}_{i,M}(\psi) = hs_i + W_q(s_M, \lambda_M, \mu) \cdot \lambda_i d. \tag{7}$$

Now, suppose that the ratio of the number of servers to arrival rates is symmetric among players. This is a reasonable symmetry condition, as it implies that players with larger arrival rates possess more servers. This symmetry is also in place when players represent equally sized service providers, all with the same number of servers and arrival rates. The following theorem states various properties exhibited by FIX-queueing games under this symmetry condition.

Theorem 4.3. *Let $\psi = (N, (\lambda_i)_{i\in N}, \mu, (s_i)_{i\in N}, h, d) \in \Psi$ be a FIX-queueing situation with $s_i/\lambda_i = s_j/\lambda_j$ for all $i, j \in N$.*

(i) For any two coalitions $M, L \in 2^N_-$ with $M \subset L$, $W_q(s_M, \lambda_M, \mu) > W_q(s_L, \lambda_L, \mu)$.

(ii) The proportional scheme $\mathcal{P}(\psi)$ is a PMAS for the FIX-queueing game (N, c^ψ).

(iii) The allocation that assigns $\mathcal{P}_{i,N}(\psi)$ to each player $i \in N$ is a stable allocation for the FIX-queueing game (N, c^ψ).

(iv) For any partition $\mathscr{Z}$ of N containing $z > 1$ nonempty disjoint coalitions covering N, it holds that $W_q(s_N, \lambda_N, \mu) \cdot z < \sum_{M\in\mathscr{Z}} W_q(s_M, \lambda_M, \mu) \cdot \lambda_M/\lambda_N$.

Part (i) says that the average delay experienced by an arbitrary customer decreases as a coalition grows larger. Part (ii) states that the amount a player has to pay under $\mathcal{P}(\psi)$ does not increase when the coalition to which he belongs grows, and part (iii) identifies a core element, cf. Sprumont (1990). Part (iv) is a direct corollary of Theorem 1 in Benjaafar (1995); it states that pooling among z groups results in a reduction of expected waiting time by at least a factor of z.

The following example shows that, in general, FIX-queueing games with four or more players need not admit a PMAS. This contrasts with results that we will obtain in Section 5 for queueing games with optimal (real) numbers of servers. Absence of a PMAS may complicate coalition formation: it implies that under some sequence of adding players one-by-one to a pooling group, there is at least one player who, at a certain point, becomes worse off when another player is added.

Example 4.3. Consider the FIX-queueing situation $\psi = (N, (\lambda_i)_{i\in N}, \mu, (s_i)_{i\in N}, h, d) \in \Psi$ with $N = \{1, 2, 3, 4\}$, $\mu = 1$, $h = 0$, $d = 10$, and

$$\begin{aligned} &\lambda_1 = \lambda_2 = 1/10; \quad && \lambda_3 = \lambda_4 = 9/10; \\ &s_1 = s_2 = 2; && s_3 = s_4 = 1. \end{aligned}$$

To show that the associated FIX-queueing game (N, c^ψ) game does not admit a PMAS, we use the dual description of the class of games with a PMAS, introduced in Norde and Reijnierse (2002). They provide a set of necessary conditions (their Theorem 8) to determine whether a game has a PMAS or not. For our 4-player game, one of these conditions[5] is given by (cf. p. 331 of their paper):

$$c^\psi(1, 2, 3) + c^\psi(2, 3, 4) \le c^\psi(1, 3) + c^\psi(2, 3) + c^\psi(2, 4). \tag{8}$$

Here, we have dropped the curly set brackets for notational ease. Inequality (8) may be interpreted as stating that an arrangement in which players 1, 2, and 3 work one unit of time together, incurring costs $c^\psi(1, 2, 3)$ per time unit, and in which player 2, 3, and 4 work one unit of time together, incurring costs $c^\psi(2, 3, 4)$ per time unit, generates lower costs than an alternative schedule in which players work the same amount of time as before, but in smaller coalitions. As in our game it holds that

$$c^\psi(1, 2, 3) + c^\psi(2, 3, 4) = \frac{1771561}{109696119} + 1\frac{655000}{1821099} > \frac{5}{11} \cdot 3 = c^\psi(1, 3) + c^\psi(2, 3) + c^\psi(2, 4),$$

Inequality (8) is not satisfied. We conclude that this game lacks a PMAS.

To make this nonexistence intuitively plausible, notice that there are two types of players: on the one hand there are players 1 and 2, both with low arrival rates and many servers, and on the other hand there are players 3 and 4, both with high arrival rates and few servers. Any two-player coalition containing one player of each type can already attain most of the benefits of pooling. Combining this fact for three of those two-player coalitions leads to an incompatibility with the costs that should be paid in two related three-player coalitions, which have relatively high costs. ♢

5 Optimal numbers of servers

In this section, we consider a setting in which the number of servers can be jointly optimized by each coalition. This is a reasonable model for situations where the number of servers

[5] Note that this condition follows from $y_{2,\{1,2,3\}} \le y_{2,\{2,3\}}$ and five similar monotonicity inequalities, combined with efficiency.

can be easily adjusted against negligible costs. We will define and analyze two games associated with such a situation, which differ in the domain on which this optimization takes place.

5.1 Situation

Consider a setting as previously described in Section 4.1, but now with an additional aspect: we allow any coalition of players (including singletons) to re-optimize the number of servers in their joint system. This is equivalent to a setting where players do not possess a number of servers a priori, but instead jointly purchase or develop a cluster of new servers after cooperation is established. Taking this into account, we further allow larger coalitions to exploit stronger bargaining power and/or economies of scale; as a result, larger coalitions can acquire and maintain servers at a reduced cost rate.

Clearly, if collaboration in this fashion is allowed, then the grand coalition will be no worse off than in the case with a fixed number of servers. After all, due to the resource pooling effect, fewer servers may suffice to jointly serve all customer streams in a cost-effective way. This reduction in the costs of the grand coalition would make it easier to find a stable allocation. Yet, sub-coalitions will also choose a cost-minimizing number of servers, reducing their costs and shrinking the core. Given these two opposite effects, we will investigate whether the properties obtained for FIX-queueing games — such as the existence of a stable cost allocation — remain valid for this new setting.

To analyze this, we introduce a multi-server queueing situation with optimal numbers of servers (OPT-queueing situation for short) as a tuple $(N, (\lambda_i)_{i\in N}, \mu, (h_M)_{M\in 2^N_-}, d)$, where N, λ, μ, and d are as in Section 4, and h_M is the resource cost incurred per unit time for each server operated by coalition M. Note that players are not associated with a number of servers anymore. With Γ we shall denote the set of such situations for which $h_M \geq h_L > 0$ for all $M, L \in 2^N_-$ with $M \subseteq L$, i.e., for which the resource cost rate does not increase as a coalition grows and always remains positive.[6]

OPT-queueing situations are often applicable if customers are served by human operators who are easily hired or fired — as opposed to service by technically advanced, expensive machines, in which case a FIX-queueing situation may be more appropriate. To illustrate how the OPT-queueing framework accommodates various known settings, one may think of several copy machines manufacturers (players) that use technicians (servers)

[6] This situation could be extended by allowing concave increasing unbounded resource cost functions rather than linear costs (cf. Karsten et al., 2011). This extended model, however, does not provide new insights.

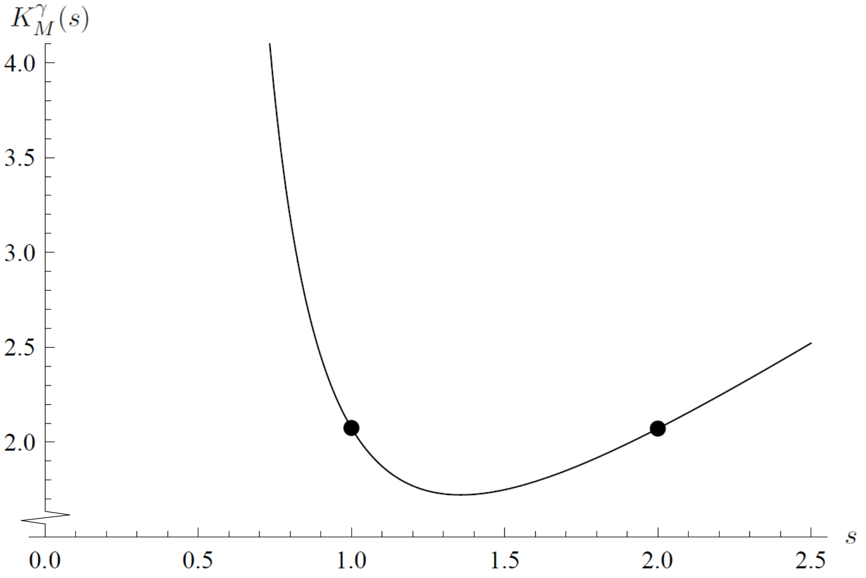

Figure 2: *The cost function $K_M^\gamma(s)$ as a function of the number of servers s, for $\lambda_M = 0.5$, $\mu = 1$, $h_M = 1$, and $d = 2.15$. This function is not defined for $s \leq \lambda_M/\mu$. The values $K_M^\gamma(1) = 2\frac{3}{40}$ and $K_M^\gamma(2) = 2\frac{43}{600}$ are identified by black dots.*

to deal with machine breakdowns (arrivals). Alternatively, one may think of various business units of a large insurance firm (players) that employ telephone agents (servers) to quickly respond to incoming customer calls (arrivals).

5.2 Games

Let $\gamma = (N, (\lambda_i)_{i \in N}, \mu, (h_M)_{M \in 2^N_-}, d) \in \Gamma$ be an OPT-queueing situation, and consider a coalition $M \in 2^N_-$. As before, this coalition will jointly serve customers that arrive according to a Poisson process with combined rate $\lambda_M = \sum_{i \in M} \lambda_i$. Suppose that this coalition would pick $s > \lambda_M/\mu$ common servers. Then, this coalition's joint service facility behaves as an Erlang delay system, and the expected costs per unit of time in steady state incurred by coalition M are equal to

$$K_M^\gamma(s) = h_M s + W_q(s, \lambda_M, \mu) \cdot \lambda_M d. \tag{9}$$

Figure 2 illustrates this cost function. Next, we formulate two different games corresponding to OPT-queueing situation γ. In the first game, any coalition M optimizes $K_M^\gamma(s)$ over integer numbers of servers, i.e., over domain $\mathscr{N}_M^\gamma = \{s \in \mathbb{N} \mid s > \lambda_M/\mu\}$. In the second game, the optimization is taken over real numbers of servers, i.e., over domain

$\mathscr{R}_M^\gamma = \{s \in \mathbb{R} \mid s > \lambda_M/\mu\}$. In both cases, due to this optimization, the resource costs represent a non-additive part of the characteristic cost function, in contrast to FIX-queueing games.

We emphasize that our main interest lies in the former game; it represents the exact discrete optimization problem. The latter game will help in understanding the discrete game: we will use the nice, mathematical results that we can obtain for the setting where the optimization is taken over the real numbers to derive results for the setting where each coalition picks an optimal integer number of servers.

Although a system with a non-integral number of servers does not lend itself to a natural interpretation, we remark that one might view it as, e.g., an approximation of a system with a part-time worker. We also point out that Borst et al. (2004), in dealing with with the staffing problem of large call centers, have approximated costs based on the continuous extension (3) to find an approximately optimal number of servers.

Now, we call the game $(N, \hat{c}^\gamma)$ with

$$\hat{c}^\gamma(M) = \min_{s \in \mathscr{N}_M^\gamma} K_M^\gamma(s) \tag{10}$$

for all $M \in 2^N_-$ and $\hat{c}^\gamma(\emptyset) = 0$ the associated *$OPT^{\mathbb{N}}$ queueing game*. Consider any coalition $M \in 2^N_-$. On domain $\mathscr{N}_M^\gamma$, the cost function $K_M^\gamma(s)$ is strictly convex (due to Theorem 3.4) and it achieves a minimum (since the costs grow unboundedly as the number of servers tends to infinity). Hence, an optimal integer number of servers is given by the smallest $s \in \mathscr{N}_M^\gamma$ satisfying $K_M^\gamma(s+1) \geq K_M^\gamma(s)$, and we denote this optimizer by $\hat{s}_M^*$.

Further, we call the game (N, c^γ) with

$$c^\gamma(M) = \min_{s \in \mathscr{R}_M^\gamma} K_M^\gamma(s) \tag{11}$$

for all $M \in 2^N_-$ and $c^\gamma(\emptyset) = 0$ the associated *$OPT^{\mathbb{R}}$ queueing game.* Consider again any coalition $M \in 2^N_-$. By strict convexity of $K_M^\gamma(s)$ in s and because $\lim_{s\to\infty} K_M^\gamma(s) = \lim_{s \downarrow \lambda_M/\mu} K_M^\gamma(s) = \infty$, the cost function $K_M^\gamma(s)$ achieves a minimum on domain $\mathscr{R}_M^\gamma$, implying that the game is well-defined. Now, the optimal real number of servers is unique (due to the strict convexity), and we denote it by s_M^*.

The following proposition states that cooperation is beneficial and establishes a link between the two games.

Proposition 5.1. *Let $\gamma = (N, (\lambda_i)_{i\in N}, \mu, (h_M)_{M\in 2^N_-}, d) \in \Gamma$ be an OPT-queueing situation, with associated $OPT^{\mathbb{N}}$ queueing game $(N, \hat{c}^\gamma)$ and $OPT^{\mathbb{R}}$ queueing game (N, c^γ).*

(i) Both games $(N, \hat{c}^\gamma)$ and (N, c^γ) are strictly subadditive.

(ii) Let $M \in 2^N_-$ be a coalition. Then, either $\hat{s}_M^* = \lfloor s_M^* \rfloor$ or $\hat{s}_M^* = \lceil s_M^* \rceil$. Furthermore, $\hat{c}^\gamma(M) \geq c^\gamma(M)$, with equality if and only if $s_M^* \in \mathbb{N}$.

5.3 Cost allocation: stability and population monotonicity

In this section, we investigate whether or not cost allocation can be carried out in a stable and population monotonic way. We start by introducing two simple rules that allocate costs proportional to arrival rates. The first rule, $\hat{\mathscr{P}}$, divides the costs of the grand coalition in $OPT^{\mathbb{N}}$ queueing games. The second rule, $\mathscr{P}$, does this for $OPT^{\mathbb{R}}$ queueing games.

Formally, they are defined, for any $\gamma = (N, (\lambda_i)_{i \in N}, \mu, (h_M)_{M \in 2^N_{-}}, d) \in \Gamma$ and $i \in N$, by $\hat{\mathscr{P}}_i(\gamma) = \hat{c}^\gamma(N)\lambda_i/\lambda_N$ and by $\mathscr{P}_i(\gamma) = c^\gamma(N)\lambda_i/\lambda_N$, respectively. Extending this idea to every coalition, we define the proportional allocation scheme rules $\hat{\mathcal{P}}$ and $\mathcal{P}$, for any $\gamma = (N, (\lambda_i)_{i \in N}, \mu, (h_M)_{M \in 2^N_{-}}, d) \in \Gamma$, $M \in 2^N_{-}$, and $i \in M$, by $\hat{\mathcal{P}}_{i,M}(\gamma) = \hat{c}^\gamma(M)\lambda_i/\lambda_M$ and by $\mathcal{P}_{i,M}(\gamma) = c^\gamma(M)\lambda_i/\lambda_M$, respectively. Note that these rules result in an efficient, genuine allocation (scheme) for their respective games. The following example illustrates the rules for $OPT^{\mathbb{R}}$ queueing games and simultaneously shows that $OPT^{\mathbb{N}}$ queueing games need not admit a stable cost allocation.

Example 5.1. Consider the OPT-queueing situation $\gamma = (N, (\lambda_i)_{i \in N}, \mu, (h_M)_{M \in 2^N_{-}}, d) \in \Gamma$ with player set $N = \{1, 2, 3\}$, arrival rates $\lambda_1 = \lambda_2 = 0.2$ and $\lambda_3 = 0.1$, service rate $\mu = 1$, resource cost rate $h_M = 1$ for all $M \subseteq N$, and delay cost rate $d = 2.15$. The cost function for the grand coalition corresponds to the cost function displayed in Figure 2 on page 20. The characteristic cost functions $\hat{c}^\gamma$ of the associated $OPT^{\mathbb{N}}$ queueing game $(N, \hat{c}^\gamma)$ and c^γ of the associated $OPT^{\mathbb{R}}$ queueing game (N, c^γ) are represented in Table 4, along with the optimal numbers of servers for each coalition in both settings. The table also specifies the allocation scheme $\mathcal{P}(\gamma)$ for game (N, c^γ).

Coalition M	$\hat{s}^*_M$	$\hat{c}^\gamma(M)$	s^*_M	$c^\gamma(M)$	$\mathcal{P}_{1,M}(\gamma)$	$\mathcal{P}_{2,M}(\gamma)$	$\mathcal{P}_{3,M}(\gamma)$
$\{1\}$	1	$1\,\frac{43}{400}$	0.75878	1.01675	1.01675	*	*
$\{2\}$	1	$1\,\frac{43}{400}$	0.75878	1.01675	*	1.01675	*
$\{3\}$	1	$1\,\frac{43}{1800}$	0.50511	0.70489	*	*	0.70489
$\{1,2\}$	1	$1\,\frac{43}{75}$	1.17171	1.50706	0.75353	0.75353	*
$\{1,3\}$	1	$1\,\frac{387}{1400}$	0.97478	1.27524	0.85016	*	0.42508
$\{2,3\}$	1	$1\,\frac{387}{1400}$	0.97478	1.27524	*	0.85016	0.42508
N	2	$2\,\frac{43}{600}$	1.35662	1.72219	0.68887	0.68887	0.34444

Table 4: *The OPT-queueing games, optimal numbers of servers, and proportional allocation scheme of Example 5.1. For the setting where the optimization is taken over the real numbers, all values are rounded to 5 decimals.*

Notice that $\mathcal{P}_{1,\{1,2\}}(\gamma) > \mathcal{P}_{1,N}(\gamma)$ and similarly $\mathcal{P}_{2,\{1,2\}}(\gamma) > \mathcal{P}_{2,N}(\gamma)$, i.e., the amount that player 1 or 2 has to pay does not increase when player 3 joins them. This population monotonicity can be verified for the members of all other nested pairs of coalitions as well, implying that $\mathcal{P}(\gamma)$ is population monotonic. Accordingly, the $OPT^{\mathbb{R}}$ queueing game (N, c^γ) has a nonempty core containing $\mathscr{P}(\gamma)$.

In contrast, the $OPT^{\mathbb{N}}$ queueing game $(N, \hat{c}^\gamma)$ has an empty core! To see this, suppose x is a stable allocation for game $(N, \hat{c}^\gamma)$. Thus, x satisfies $x_1 + x_2 + x_3 = \hat{c}^\gamma(N)$, $x_1 + x_2 \leq \hat{c}^\gamma(\{1,2\})$, $x_1 + x_3 \leq \hat{c}^\gamma(\{1,3\})$, and $x_2 + x_3 \leq \hat{c}^\gamma(\{2,3\})$, which implies $2\hat{c}^\gamma(N) \leq \hat{c}^\gamma(\{1,2\}) + \hat{c}^\gamma(\{1,3\}) + \hat{c}^\gamma(\{2,3\})$. However, $2\hat{c}^\gamma(N) = 4\frac{43}{300} > 4\frac{53}{420} = \hat{c}^\gamma(\{1,2\}) + \hat{c}^\gamma(\{1,3\}) + \hat{c}^\gamma(\{2,3\})$. This yields a contradiction. We conclude that no stable allocation exists. $\Diamond$

It is worth pointing out that Yu et al. (2007) gave a (2-player) counterexample indicating that their games corresponding to server optimization in Erlang delay systems need not have a nonempty core. However, their counterexample included players with asymmetrical delay costs, and as a result their game was not subadditive, i.e., complete pooling was detrimental. In contrast, in our subadditive game $(N, \hat{c}^\gamma)$, all customers are homogenous in delay costs and full pooling is beneficial. Despite the subadditivity, a stable allocation is lacking.

In Example 5.1, we observed that the proportional allocation scheme rule $\mathcal{P}$ accomplished a population monotonic allocation scheme for the $OPT^{\mathbb{R}}$ queueing game. The following theorem shows that this is not a coincidence.

Theorem 5.2. *Let $\gamma = (N, (\lambda_i)_{i\in N}, \mu, (h_M)_{M\in 2^N_-}, d) \in \Gamma$ be an OPT-queueing situation. Then, $\mathcal{P}(\gamma)$ is a PMAS for the $OPT^{\mathbb{R}}$ queueing game (N, c^γ). Moreover, each sub-game of (N, c^γ) has a non-empty core, and $\mathscr{P}(\gamma)$ is in the core of (N, c^γ).*

The following theorem states sufficient conditions for $OPT^{\mathbb{N}}$ queueing games to possess a core allocation and to admit a PMAS.

Theorem 5.3. *Let $\gamma = (N, (\lambda_i)_{i\in N}, \mu, (h_M)_{M\in 2^N_-}, d) \in \Gamma$ be an OPT-queueing situation.*

(i) If $s^*_N \in \mathbb{N}$, then the game $(N, \hat{c}^\gamma)$ has a non-empty core that contains $\hat{\mathscr{P}}(\gamma)$.

(ii) If $s^*_M \in \mathbb{N}$ for all $M \in 2^N_-$, then $\hat{\mathcal{P}}(\gamma)$ is a PMAS for game $(N, \hat{c}^\gamma)$.

Several insights emerge from our analysis thus far. First, $OPT^{\mathbb{R}}$ queueing games show nice properties: they have nonempty cores and admit a population monotonic allocation scheme. These properties are not satisfied by $OPT^{\mathbb{N}}$ queueing games in general; the integrality requirement is the sole culprit, as clearly illustrated by the preceding theorem.

Interestingly, the M/M/1 games where coalitions optimize service capacity to reduce their customers' system sojourn times — considered in González and Herrero (2004), García-Sanz et al. (2008), and Yu et al. (2009) — as well as the M/G/s/s games where coalitions pick optimal numbers of servers — analyzed by Karsten et al. (2011) and Özen et al. (2011) — all had nonempty cores. As such, $OPT^{\mathbb{N}}$ queuing games exhibit fundamentally different behavior.

5.4 Approximate stability and population monotonicity

In the previous section, we showed that $OPT^{\mathbb{N}}$ queueing games exhibit different behavior than $OPT^{\mathbb{R}}$ queueing games: games in the latter class always have a nonempty core, whereas games in the former class may possess an empty core. Yet, the only difference between the two games is the domain over which the number of servers is optimized. To expand our understanding of potential instability in $OPT^{\mathbb{N}}$ queueing games, we will introduce approximate core and PMAS concepts, and we will use these concepts to derive insights regarding the impact of integrality.

The first general concept for cooperative games that we introduce in this section can be seen a generalization of the core. For any vector $\epsilon = (\epsilon_i) \in \mathbb{R}^N$, we define the (vector) *ϵ-core* of game (N, c) as

$$Core_\epsilon(N, c) = \{x \in \mathbb{R}^N \mid \sum_{i \in N} x_i = c(N) \text{ and } \sum_{i \in M} x_i \leq c(M) + \sum_{i \in M} \epsilon_i \text{ for all } M \in 2^N_-\}.$$

This ϵ-core is the set of all cost allocations where no coalition M can obtain lower costs by leaving the grand coalition, if upon leaving it must pay a penalty of ϵ_i for member i. Naturally, the core of a game coincides with its $\mathbf{0}$-core. For any game (N, c), an allocation in $Core_\epsilon(N, c)$ for some vector ϵ is called an *ϵ-stable* allocation. Our vector *ϵ-core* is reminiscent of the weak ϵ-core introduced by Shapley and Shubik (1966), but differs from it by associating a number with each player rather than a single number with all players.

We next introduce another new concept analogous to the (vector) ϵ-core. For any vector $\epsilon = (\epsilon_i) \in \mathbb{R}^N$, we say that an allocation scheme y for a game (N, c) is an *$\epsilon-PMAS$* if $y_{i,M} + \epsilon_i \geq y_{i,L}$ for all coalitions $M, L \in 2^N_-$ with $M \subset L$ and $i \in M$. Here, ϵ_i may be interpreted as an exogenous bonus received by player i if the coalition to which he belongs grows.

The following theorem uses these newly defined notions to capture the influence of the integrality requirement.

Theorem 5.4. *Let $\gamma = (N, (\lambda_i)_{i \in N}, \mu, (h_M)_{M \in 2^N_-}, d) \in \Gamma$ be an OPT-queueing situation.*

(i) Fix ϵ by $\epsilon_i = h_N \lambda_i / \lambda_N$ for all $i \in N$. Then, the game $(N, \hat{c}^\gamma)$ has a non-empty (vector) ϵ-core, and $\hat{\mathscr{P}}(\gamma)$ is an ϵ-stable allocation.

(ii) Fix ϵ by $\epsilon_i = h_{\{i\}}$ for all $i \in N$. Then, $\hat{\mathcal{P}}(\gamma)$ is a ϵ-PMAS for the game $(N, \hat{c}^\gamma)$.[7]

Part (i) of Theorem 5.4 constructively shows that any possible instability disappears if coalitions would have to pay an amount, no greater than h_N, to leave the grand coalition. Part (ii) of this theorem states a similar conclusion regarding the effect of discrete service capacity on population monotonicity. Altogether, our analysis suggests that $OPT^{\mathbb{N}}$ queueing games associated with realistic large service facilities have nonempty ϵ-cores and an ϵ-PMAS for relatively small ϵ, as the resource cost incurred for a single server is small relative to the total costs faced by any coalition if the optimal number of servers is large. This is in line with the observation of Yu et al. (2007) that the core of $OPT^{\mathbb{N}}$ queueing games would be nonempty if the characteristic costs are approximated via the Halfin-Whitt heavy traffic regime — an approximation that is asymptotically exact as the arrival rate tends to infinity.

6 Concluding remarks

In this paper, we have applied concepts from cooperative game theory to study the problem of fair allocation of shared costs in multi-server queueing systems with infinite waiting room. Our model features several independent service providers, each associated with their own customer populations. They can collaborate by sharing servers, which is beneficial from the whole system point of view. In both cases considered, fixed and optimal numbers of servers, we studied (existence of) stable allocations of the resource costs for common servers and delay costs for waiting customers — stable in the sense that no subset of players has an incentive to split off and form a separate pooling group. Our analysis reveals that collaboration is always supportable by a stable allocation if players' numbers of servers are exogenously given (in line with the corresponding M/M/1 game of Anily and Haviv, 2010), whereas a stable allocation need not exist in general if each coalition optimizes over *integer* number of servers (in contrast to M/M/1 games considered in, e.g., Yu et al., 2009).

[7] If $|N| > 1$, our proof approach immediately reveals that a stronger, but less crisp, bound is possible: $\hat{\mathcal{P}}(\gamma)$ is also an $\tilde{\epsilon}$-PMAS for game $(N, \hat{c}^\gamma)$ if we set, for all $i \in N$, $\tilde{\epsilon}_i$ to be equal to $\max\{h_M \lambda_i / \lambda_M : M \in 2^N_-, |M| = 2, \text{and } i \in M\}$. Note that $\tilde{\epsilon}_i \leq h_{\{i\}}$.

6.1 Proportional cost allocations

A common theme in our study is that a proportional allocation, which simply divides joint delay costs proportional to players' arrival rates, is often stable and reachable through a population monotonic allocation scheme. This is true for the case with fixed numbers of servers under a symmetry condition. For the case with optimal numbers of servers, the corresponding proportional allocation is close-to-stable in general and stable in absence of the integrality requirement. These results imply that in a broad range of realistic settings, especially for large service facilities, the allocation proportional to arrival rates is a "fair" (or at least "close-to-fair") way to divide joint costs (in accordance with Yu et al., 2009).

This is an important result, because a proportional allocation is easy to understand, is computationally attractive, and would be easy to implement in practice. In fact, it could even be implemented via a simple cost division per realization. Although our games have been formulated in expected terms to investigate a priori attractiveness of resource pooling, fair assignments of *realized* delay costs in any finite time period are needed to sustain support for the cooperation in practice. For the specific case where the number of servers is optimized, one seemingly fair process to fully assign actually realized costs in any time period would be for every player to incur the actual delay costs for their own customers and, upon arrival of any of its customers, to pay the resource costs incurred for all servers until the next customer arrival of any player. Notice that the long-term average costs assigned to each player under this process coincide with the proportional allocation of expected costs! Additionally, a player may appreciate that, under the proposed process of assigning realized costs, few customer arrivals for this player over some period of time imply a correspondingly low cost charge. Moreover, the process eliminates the need for transfer payments of delay costs, thereby avoiding disputes about the exact magnitude of delay costs.

6.2 Future research

There are various directions in which our work may be extended. One interesting avenue is to relax the assumption that delay costs and service times are symmetric across players. Opposed to the setting considered in this paper, complete pooling of servers need not be superior anymore if such asymmetries are allowed (see, e.g., Smith and Whitt, 1981). This issue may be circumvented by preferential treatment of more critical classes via priority disciplines, although this is far from a trivial extension.

Finally, we concede that completely pooling service systems is not always feasible, especially if service facilities of the players are operated at geographically dispersed locations. Nevertheless, some degree of partial pooling may still be feasible. To study such a setting, challenging as it may be, one may consider a model variation in which players partition themselves into separate service groups, in line with Whitt (1999).

References

S. Anily and M. Haviv. Cooperation in Service Systems. *Operations Research*, 58(3): 660–673, 2010.

S. Anily and M. Haviv. Homogeneous of degree one games are balanced with applications to service systems. Working Paper, School of Business, Tel Aviv University, 2011.

S. Benjaafar. Performance bounds for the effectiveness of pooling in multi-processing systems. *European Journal of Operational Research*, 87(2):375–388, 1995.

O. Bondareva. Certain applications of the methods of linear programming to the theory of cooperative games (in Russian). *Problemy Kibernetiki*, 10:119–139, 1963.

S. Borst, A. Mandelbaum, and M.I. Reiman. Dimensioning large call centers. *Operations Research*, 52(1):17–34, 2004.

J.M. Calabrese. Optimal workload allocation in open networks of multiserver queues. *Management Science*, 38(12):1792–1802, 1992.

R.B. Cooper. *Introduction to queueing theory.* North-Holland, 1981.

F. Cruijssen, M. Cools, and W. Dullaert. Horizontal cooperation in logistics: Opportunities and impediments. *Transportation Research Part E*, 43(2):129–142, 2007.

M.E. Dyer and L.G. Proll. On the validity of marginal analysis for allocating servers in M/M/c queues. *Management Science*, 23(9):1019–1022, 1977.

A.K. Erlang. Løsning af nogle problemer fra sandsynlighedsregningen af betydning for de automatiske telefoncentraler. *Electroteknikeren*, 13:5–13, 1917. Translation: Solution of some problems in the theory of probabilities of significance in automatic telephone exchanges. In: E. Brockmeyer, H.L. Halstrøm, and A. Jensen, editors, *The Life and Works of A.K. Erlang*, pages 138–155. Transactions of the Danish Academy of Technical Sciences, 1948.

M.D. García-Sanz, F.R. Fernández, M.G. Fiestras-Janeiro, I. García-Jurado, and J. Puerto. Cooperation in Markovian queueing models. *European Journal of Operational Research*, 188(2):485–495, 2008.

D.B. Gillies. Solutions to general non-zero-sum games. In A. Tucker and R. Luce, editors, *Contribution to the theory of games IV, Volume 40 of Annals of Mathematics Studies*, pages 47–85. Princeton University Press, 1959.

P. González and C. Herrero. Optimal sharing of surgical costs in the presence of queues. *Mathematical Methods of Operations Research*, 59(3):435–446, 2004.

A.A. Jagers and E.A. Van Doorn. Convexity of functions which are generalizations of the Erlang loss function and the Erlang delay function. *SIAM Review*, 33(2):281–282, 1991.

F.J.P. Karsten, M. Slikker, and G.J. Van Houtum. Spare parts inventory pooling games. BETA Working Paper 300, Eindhoven University of Technology, 2009.

F.J.P. Karsten, M. Slikker, and G.J. Van Houtum. Analysis of resource pooling games via a new extension of the Erlang loss function. BETA Working Paper 344, Eindhoven University of Technology, 2011.

M. Leng and M. Parlar. Analytic solution for the nucleolus of a three-player cooperative game. *Naval Research Logistics*, 57(7):667–672, 2010.

H. Norde and H. Reijnierse. A dual description of the class of games with a population monotonic allocation scheme. *Games and Economic Behavior*, 41(2):322–343, 2002.

U. Özen, M.I. Reiman, and Q. Wang. On the Core of Cooperative Queueing Games. To appear in Operations Research Letters., 2011.

D. Schmeidler. The nucleolus of a characteristic function game. *SIAM Journal on Applied Mathematics*, 17(6):1163–1170, 1969.

L.S. Shapley. A value for n-person games. In H. Kuhn and A. Tucker, editors, *Contribution to the theory of games II, Volume 28 of Annals of Mathematics Studies*, pages 307–317. Princeton University Press, 1953.

L.S. Shapley. On balanced sets and cores. *Naval Research Logistics Quarterly*, 14:453–460, 1967.

L.S. Shapley. Cores of convex games. *International Journal of Game Theory*, 1(1):11–26, 1971.

L.S. Shapley and M. Shubik. Quasi-cores in a monetary economy with nonconvex preferences. *Econometrica*, 34(4):805–827, 1966.

D.R. Smith and W. Whitt. Resource sharing for efficiency in traffic systems. *Bell System Technical Journal*, 60(1):39–55, 1981.

C. Snijders. Axiomatization of the nucleolus. *Mathematics of Operations Research*, 20(1): 189–196, 1995.

Y. Sprumont. Population monotonic allocation schemes for cooperative games with transferable utility. *Games and Economic Behavior*, 2(4):378–394, 1990.

J. Timmer and W. Scheinhardt. How to share the cost of cooperating queues in a tandem network? In *Conference proceedings of the 22nd International Teletraffic Congress 2010*, pages 1–7. IEEE, 2010.

W. Whitt. Partitioning customers into service groups. *Management Science*, 45(11):1579–1592, 1999.

W. Whitt. The Erlang B and C formulas: problems and solutions. Unpublished class notes. Available online: `http://www.columbia.edu/~ww2040/ErlangBandCFormulas.pdf`, 2002.

Y. Yu, S. Benjaafar, and Y. Gerchak. Capacity Pooling and Cost Sharing among Independent Firms in the Presence of Congestion. Working paper, University of Minnesota, 2007.

Y. Yu, S. Benjaafar, and Y. Gerchak. Capacity Sharing and Cost Allocation among Independent Firms in the Presence of Congestion. Working paper, University of Minnesota, 2009.

A Appendix

A.1 Proofs for section 3.

Proof of Lemma 3.1. For notational ease, let $C = C(s,a)$, $B = B(s,a)$, $C^{-1} = 1/C$, $B^{-1} = 1/B$, and $\rho = a/s$. (Note that $C > 0$ and $B > 0$.) Then,

$$\begin{aligned} C^{-1} - (1-\rho)B^{-1} &= \int_0^\infty ae^{-ax}(1+x)^{s-1}xdx - \int_0^\infty ae^{-ax}(1+x)^s(1-\rho)dx \\ &= \int_0^\infty ae^{-ax}(1+x)^{s-1}[x-(1+x)(1-\rho)]dx \\ &= \int_0^\infty ae^{-ax}(1+x)^{s-1}[\rho(1+x)-1]dx \\ &= -ae^{-ax}(1+x)^s/s\Big|_{x=0}^{x=\infty} = \rho. \end{aligned}$$

This implies $BC^{-1} - 1 + \rho = B\rho$, which in turn implies $BC^{-1} - 1 = -\rho(1-B)$, and thus $C = B/[1-\rho(1-B)]$. This completes the proof. □

Proof of Lemma 3.2. By Lemma 3.1, we have

$$C(ts,ta) = \frac{1}{\dfrac{1-a/s}{B(ts,ta)} + a/s}. \tag{12}$$

Now, as shown in the appendix of Smith and Whitt (1981), $B(ts,ta)$ is decreasing in t for $t > 0$ (see also the clarifying remark in Karsten et al., 2009, Section 2.2). This result, in combination with Equation (12), implies that $C(ts,ta)$ is also decreasing in t for $t > 0$. □

Proof of Theorem 3.3. Let $t_1, t_2 > 0$ with $t_1 < t_2$. Then,

$$\begin{aligned} W_q(t_1s, t_1\lambda, \mu) &= C(t_1s, t_1\lambda/\mu)/[t_1(s\mu-\lambda)] \\ &> C(t_2s, t_2\lambda/\mu)/[t_1(s\mu-\lambda)] \\ &> C(t_2s, t_2\lambda/\mu)/[t_2(s\mu-\lambda)] = W_q(t_2s, t_2\lambda, \mu), \end{aligned}$$

where the first inequality holds by Lemma 3.2. We conclude that $W_q(ts, t\lambda, \mu)$ is decreasing in t for $t > 0$. □

In the process of proving the next property of the continuous extension of the Erlang delay function, we will use the following lemma.

Lemma A.1. *Let $f : D \to \mathbb{R}$ be a positive, non-increasing, convex function on an interval $D \subseteq \mathbb{R}$. Let $g : D \to \mathbb{R}$ be a positive, decreasing, strictly convex function on the same domain as f. Then the function $h : D \to \mathbb{R}$, defined by $h(s) = f(s)g(s)$ for all $s \in D$, is decreasing and strictly convex.*

Proof. The case $|D| = 1$ is trivial. In the remainder of the proof, we will assume $|D| > 1$. Let $s_1, s_2 \in D$ with $s_1 < s_2$. We first show that h is decreasing. We have

$$h(s_1) = f(s_1)g(s_1) \geq f(s_2)g(s_1) > f(s_2)g(s_2) = h(s_2),$$

where the first inequality holds because f is non-increasing while $g(s_1) > 0$, and the second inequality holds because $f(s_2) > 0$ and g is decreasing. We conclude that h is decreasing.

Next, we show that h is strictly convex. Let $x \in (0, 1)$. Then,

$$\begin{aligned}
h(xs_1 + (1-x)s_2) &= f(xs_1 + (1-x)s_2) \cdot g(xs_1 + (1-x)s_2) \\
&\leq \big[xf(s_1) + (1-x)f(s_2)\big] \cdot g(xs_1 + (1-x)s_2) \\
&< \big[xf(s_1) + (1-x)f(s_2)\big] \cdot \big[xg(s_1) + (1-x)g(s_2)\big] \\
&= x^2 h(s_1) + (1-x)^2 h(s_2) + x(1-x)f(s_1)\big[g(s_2) - g(s_1) + g(s_1)\big] \\
&\qquad + (1-x)xf(s_2)\big[g(s_1) - g(s_2) + g(s_2)\big] \\
&= xh(s_1) + (1-x)h(s_2) \\
&\qquad + x(1-x)\big[f(s_1)[g(s_2) - g(s_1)] + f(s_2)[g(s_1) - g(s_2)]\big] \\
&\leq xh(s_1) + (1-x)h(s_2),
\end{aligned}$$

The first inequality holds because f is convex, while g is a positive function. The second inequality holds because g is strictly convex and f is a positive function. The third inequality holds because $x(1-x) > 0$, $g(s_2) - g(s_1) < 0$, and $f(s_1) - f(s_2) \geq 0$. We conclude that h is indeed strictly convex. □

Proof of Theorem 3.4. For the fixed choice of λ and μ, we denote $a = \lambda/\mu$ and we define, for all $s \in \mathbb{R}$ with $s > a$, $f(s) = C(s, a)$ and $g(s) = 1/(s\mu - \lambda)$. Thus, for all $s \in \mathbb{R}$ with $s > a$, we have $W_q(s, \lambda, \mu) = f(s)g(s)$. Since, as shown by Jagers and Van Doorn (1991), the continuous extension of the Erlang delay function is positive, non-increasing and convex in the number of servers, $f(s)$ has the same properties. Moreover, $g(s)$ is positive, decreasing in s, and convex in s. Hence, the assumptions in Lemma A.1 are satisfied, and the theorem follows. □

Proof of Theorem 3.5. The subadditivity result follows from

$$\begin{aligned}
W_q(s_1+s_2,\lambda_1+\lambda_2,\mu) &< \frac{\lambda_1}{\lambda_1+\lambda_2}W_q\left(\frac{\lambda_1+\lambda_2}{\lambda_1}s_1,\lambda_1+\lambda_2,\mu\right)\\
&\quad+\frac{\lambda_2}{\lambda_1+\lambda_2}W_q\left(\frac{\lambda_1+\lambda_2}{\lambda_2}s_2,\lambda_1+\lambda_2,\mu\right)\\
&< \frac{\lambda_1}{\lambda_1+\lambda_2}W_q(s_1,\lambda_1,\mu)+\frac{\lambda_2}{\lambda_1+\lambda_2}W_q(s_2,\lambda_2,\mu),
\end{aligned}$$

where the first inequality holds by the convexity property of Theorem 3.4 and the second inequality holds by Theorem 3.3. Multiplying both sides with $\lambda_1+\lambda_2>0$ completes the proof. □

A.2 Proofs for Section 4.

Proof of Proposition 4.1. Let $\psi=(N,(\lambda_i)_{i\in N},\mu,(s_i)_{i\in N},h,d)\in\Psi$ be a FIX-queueing situation. Let $M,L\in 2^N_-$ with $M\cap L=\emptyset$. Then,

$$\begin{aligned}
c^\psi(M\cup L) &= h(s_M+s_L)+W_q(s_M+s_L,\lambda_M+\lambda_L,\mu)\cdot(\lambda_M+\lambda_L)d\\
&< hs_M+W_q(s_M,\lambda_M,\mu)\cdot\lambda_M d+hs_L+W_q(s_L,\lambda_L,\mu)\cdot\lambda_L d\\
&= c^\psi(M)+c^\psi(L),
\end{aligned}$$

where the inequality holds by Theorem 3.5. □

To prove Theorem 4.2, we employ the notion of balancedness. For this, we first need to define balanced maps and balanced collections for player set N. A map $\kappa:2^N_-\to[0,1]$ is called *balanced for* N if $\sum_{M\in 2^N_-:i\in M}\kappa(M)=1$ for all $i\in N$. For each balanced map κ for N, we define $\mathbb{B}(\kappa)=\{M\in 2^N_-\mid\kappa(M)>0\}$. Any collection $\mathbb{B}\subseteq 2^N_-$ of coalitions is called *balanced for* N if there is a balanced map κ for N such that $\mathbb{B}=\mathbb{B}(\kappa)$. As an example, for $N=\{1,2,3\}$, $\mathbb{B}=\{\{1,2\},\{1,3\},\{2,3\}\}$ is a balanced collection because the map $\kappa:2^N_-\to[0,1]$, described by $\kappa(M)=\frac{1}{2}$ if $|M|=2$ and $\kappa(M)=0$ otherwise, is a balanced map satisfying $\mathbb{B}(\kappa)=\mathbb{B}$. Thus, balanced maps may be fractional-valued. In the proof of Theorem 4.2, we will use balanced maps to construct queueing systems with possibly fractional numbers of servers, which is a main reason for analyzing the continuous extension of the Erlang delay function in Section 3.2.

A balanced collection $\mathbb{B}$ is called *minimally balanced for* N if $N\notin\mathbb{B}$ and there does not exist another balanced collection for N that is a proper subset of $\mathbb{B}$. Correspondingly, a balanced map κ is called *minimally balanced for* N if $\mathbb{B}(\kappa)$ is a minimally balanced collection for N. Each minimally balanced collection is associated with a unique minimally balanced map (Shapley, 1967). Let $\mathcal{B}^N$ denote the set of minimally balanced maps for N. Every $\kappa \in \mathcal{B}^N$ satisfies $\sum_{M\in\mathbb{B}(\kappa)} \kappa(M)\cdot\sum_{i\in M} f(i) = \sum_{i\in N} f(i)$ for all functions $f : N \to \mathbb{R}$.

Bondareva (1963) and Shapley (1967) showed that the core of a game (N, c) with two or more players is non-empty if and only if for every minimally balanced map $\kappa \in \mathcal{B}^N$ it holds that $\sum_{M\in\mathbb{B}(\kappa)} \kappa(M)c(M) \geq c(N)$. Using this result, we will be able to prove part (i) of Theorem 4.2. To prove part (ii) of that theorem, we will exploit the following lemma.

Lemma A.2. *Let (N, c) be a game with $|N| > 1$ and $c(N) < \sum_{M\in\mathbb{B}(\kappa)} \kappa(M)c(M)$ for each $\kappa \in \mathcal{B}^N$. This game has infinitely many core allocations.*

Proof. Let $\kappa^* \in \operatorname{argmin}_{\kappa\in\mathcal{B}^N}\{\sum_{M\in\mathbb{B}(\kappa)} \kappa(M)c(M)\}$. Such a κ^* exists because the number of minimally balanced collections for N is no larger than the number of subsets of 2^N_-. Let $\epsilon^* = \sum_{M\in\mathbb{B}(\kappa^*)} \kappa^*(M)c(M) - c(N)$; note that $\epsilon^* > 0$. We will aim to identify two different core allocations. To this end, we define the auxiliary game (N, c^*) by $c^*(M) = c(M)$ for all proper subsets $M \subset N$, and $c^*(N) = c(N) + \epsilon^*$. Then, for any $\kappa \in \mathcal{B}^N$, we obtain

$$c^*(N) = c(N) + \epsilon^* = \sum_{M\in\mathbb{B}(\kappa)} \kappa^*(M)c(M) - \epsilon^* + \epsilon^* \leq \sum_{M\in\mathbb{B}(\kappa)} \kappa(M)c(M),$$

where the inequality holds by choice of κ^* as a minimizer. Hence, the game (N, c^*) is balanced as well, so the nucleolus $\nu(N, c^*)$ is in the core of (N, c^*).

Now, we let $i, j \in N$, $i \neq j$, be two different players, and we define two allocations for the game (N, c): $n^{(i)}$ and $n^{(j)}$. The first allocation, $n^{(i)}$, is defined by $n^{(i)}_k = \nu_k(N, c^*)$ for all $k \in N \setminus \{i\}$ and $n^{(i)}_i = \nu_i(N, c^*) - \epsilon^*$. The second allocation, $n^{(j)}$, is defined analogously, now reducing player j's allocation by ϵ^*. Clearly, $n^{(i)} \neq n^{(j)}$. Note that $n^{(i)}$ is an efficient allocation for game (N, c) since

$$\sum_{k\in N} n^{(i)}_k = \sum_{k\in N\setminus\{i\}} \nu_k(N, c^*) + \nu_i(N, c^*) - \epsilon^* = c^*(N) - \epsilon^* = c(N),$$

where the second equality holds because $\nu(N, c^*)$ is an efficient allocation for the game (N, c^*). Moreover, $n^{(i)}$ is a stable allocation for game (N, c) since, for any proper subset $M \subset N$, it holds that

$$\sum_{k\in M} n^{(i)}_k \leq \sum_{k\in M} \nu_k(N, c^*) \leq c^*(M) = c(M),$$

where the second inequality is valid because $\nu(N, c^*)$ was a stable allocation for the game (N, c^*). We conclude that $\nu^{(i)}$ is in the core of game (N, c). The argument for core inclusion of $\nu^{(j)}$ goes analogously. Finally, as the core is a convex set, the existence of two different core allocations implies that there are infinitely many core allocations. □

Proof of Theorem 4.2. We first prove part (i). If $|N| = 1$, then the proof is trivial. In case $|N| > 1$, we let $\kappa \in \mathcal{B}^N$ be an arbitrary minimally balanced map. Then, we have

$$\begin{aligned}
c^{\psi}(N) &= hs_N + W_q\left(s_N, \lambda_N, \mu\right) \cdot \lambda_N d \\
&= h \sum_{M \in \mathbb{B}(\kappa)} \kappa(M) s_M + W_q\left(\sum_{M \in \mathbb{B}(\kappa)} \kappa(M) s_M \cdot \frac{\lambda_N}{\lambda_M} \cdot \frac{\lambda_M}{\lambda_N}, \lambda_N, \mu\right) \cdot \lambda_N d \\
&\leq h \sum_{M \in \mathbb{B}(\kappa)} \kappa(M) s_M + \sum_{M \in \mathbb{B}(\kappa)} \kappa(M) \frac{\lambda_M}{\lambda_N} \cdot W_q\left(s_M \cdot \frac{\lambda_N}{\lambda_M}, \lambda_N, \mu\right) \cdot \lambda_N d \\
&= \sum_{M \in \mathbb{B}(\kappa)} \kappa(M) \left[hs_M + W_q\left(s_M \cdot \frac{\lambda_N}{\lambda_M}, \lambda_N, \mu\right) \cdot \lambda_M d\right] \\
&< \sum_{M \in \mathbb{B}(\kappa)} \kappa(M) \left[hs_M + W_q\left(s_M, \lambda_M, \mu\right) \cdot \lambda_M d\right] \\
&= \sum_{M \in \mathbb{B}(\kappa)} \kappa(M) c^{\psi}(M).
\end{aligned}$$

The second equality holds because $\sum_{M \in \mathbb{B}(\kappa)} \kappa(M) s_M = s_N$. The first inequality holds by the convexity property of Theorem 3.4, and $\sum_{M \in \mathbb{B}(\kappa)} \kappa(M) \lambda_M / \lambda_N = 1$; that is, we employ this convexity by using that the nonnegative convex weights $\kappa(M) \lambda_M / \lambda_N$ add up to 1. The second inequality holds by Theorem 3.3. This inequality is strict because κ is minimally balanced and thus, by definition, every $M \in \mathbb{B}(\kappa)$ is a proper subset of N.

We conclude that the game (N, c^{ψ}) is balanced. Noting that every sub-game of (N, c^{ψ}) is a game associated with an element of Ψ itself completes the proof of part (i).

To show part (ii), we assume that $|N| > 1$. The inequality $c^{\psi}(N) < \sum_{M \in \mathbb{B}(\kappa)} \kappa(M) c^{\psi}(M)$ is strict for each $\kappa \in \mathcal{B}^N$, as shown above in part (i). Thus, by Lemma A.2, the game (N, c^{ψ}) has infinitely many core allocations. This completes the proof. □

Proof of Theorem 4.3. Part (i). Let $M, L \in 2^N_-$ with $M \subset L$. Then,

$$W_q(s_M, \lambda_M, \mu) > W_q\left(s_M \cdot \frac{s_L}{s_M}, \lambda_M \cdot \frac{s_L}{s_M}, \mu\right) = W_q(s_L, \lambda_L, \mu),$$

where the inequality follows by Theorem 3.3 and the equality holds because $s_M / \lambda_M = s_L / \lambda_L$.

Part (ii). Let $M, L \in 2^N_-$ with $M \subseteq L$, and let $i \in M$. Then,

$$\begin{aligned}
\mathcal{P}_{i,M}(\psi) &= hs_i + W_q(s_M, \lambda_M, \mu) \cdot \lambda_i d \\
&\geq hs_i + W_q(s_L, \lambda_L, \mu) \cdot \lambda_i d = \mathcal{P}_{i,L}(\psi),
\end{aligned}$$

where the inequality follows by part (i). We conclude that $\mathcal{P}(\psi)$ is indeed a PMAS.

Finally, parts (iii) and (iv) immediately follow from Sprumont (1990) and from Theorem 1 in Benjaafar (1995), respectively. $\square$

A.3 Proofs for Section 5.

Proof of Proposition 5.1. Part (i). Let $M, L \in 2^N_-$ with $M \cap L = \emptyset$. To show strict subadditivity of the $OPT^{\mathbb{R}}$ game (N, c^γ), we have

$$\begin{aligned}
c^\gamma(M \cup L) &= h_{M\cup L} \cdot s^*_{M\cup L} + W_q(s^*_{M\cup L}, \lambda_M + \lambda_L, \mu) \cdot (\lambda_M + \lambda_L)d \\
&\leq h_{M\cup L}(s^*_M + s^*_L) + W_q(s^*_M + s^*_L, \lambda_M + \lambda_L, \mu) \cdot (\lambda_M + \lambda_L)d \\
&\leq h_M s^*_M + h_L s^*_L + W_q(s^*_M + s^*_L, \lambda_M + \lambda_L, \mu) \cdot (\lambda_M + \lambda_L)d \\
&< h_M s^*_M + W_q(s^*_M, \lambda_M, \mu) \cdot \lambda_M d + h_L s^*_L + W_q(s^*_L, \lambda_L, \mu) \cdot \lambda_L d \\
&= c^\gamma(M) + c^\gamma(L),
\end{aligned}$$

where the first inequality holds because $s^*_{M\cup L}$ is an optimal number of servers for coalition $M \cup L$, the second inequality is valid because $h_{M\cup L} \leq h_M$ and $h_{M\cup L} \leq h_L$ by assumption on γ, and the third inequality holds by Theorem 3.5. We conclude that (N, c^γ) is strictly subadditive. The proof of strict subadditivity of the $OPT^{\mathbb{N}}$ game $(N, \hat{c}^\gamma)$ goes analogously.

Part (ii). First, the claim that $\hat{s}^*_M = \lfloor s^*_M \rfloor$ or $\hat{s}^*_M = \lceil s^*_M \rceil$ follows immediately from strict convexity of $K^\gamma_M(s)$. The relation between $\hat{c}^\gamma(M)$ and $c^\gamma(M)$ follows from uniqueness of s^*_M and from the observation that in the $OPT^{\mathbb{R}}$ game any coalition $M \in 2^N_-$ optimizes over a larger domain than in the $OPT^{\mathbb{N}}$ game, which implies that $\hat{c}^\gamma(M) = c^\gamma(M)$ if and only if $s^*_M = \hat{s}^*_M$, but that occurs if and only if $s^*_M \in \mathbb{N}$. $\square$

Proof of Theorem 5.2. Let $M, L \in 2^N_-$ with $M \subseteq L$, and let $i \in M$. Then,

$$\begin{aligned}
\mathcal{P}_{i,L}(\gamma) &= \frac{\lambda_i}{\lambda_L} K^\gamma_L(s^*_L) \\
&\leq \frac{\lambda_i}{\lambda_L} K^\gamma_L\left(s^*_M \frac{\lambda_L}{\lambda_M}\right) \\
&= h_L s^*_M \frac{\lambda_i}{\lambda_M} + W_q\left(s^*_M \frac{\lambda_L}{\lambda_M}, \lambda_L, \mu\right) \cdot \lambda_i d \\
&\leq h_L s^*_M \frac{\lambda_i}{\lambda_M} + W_q(s^*_M, \lambda_M, \mu) \cdot \lambda_i d \\
&\leq h_M s^*_M \frac{\lambda_i}{\lambda_M} + W_q(s^*_M, \lambda_M, \mu) \cdot \lambda_i d \\
&= \frac{\lambda_i}{\lambda_M} K^\gamma_M(s^*_M) = \mathcal{P}_{i,M}(\gamma),
\end{aligned}$$

where the first inequality holds because s_L^* is the optimal number of servers for coalition L, the second inequality is valid by Theorem 3.3, and the third inequality holds because $h_L \leq h_M$ by assumption on γ. We conclude that $\mathcal{P}(\gamma)$ is indeed a PMAS for game (N, c^γ). By Sprumont (1990), this immediately implies that each sub-game of (N, c^γ) has a non-empty core and that $\mathscr{P}(\gamma)$ is a core element of game (N, c^γ). □

Proof of Theorem 5.3. Part (i). Let $M \in 2_-^N$ be an arbitrary coalition. Assuming $s_N^* \in \mathbb{N}$, it holds that $\hat{c}^\gamma(N) = c^\gamma(N)$ and $\hat{c}^\gamma(M) \geq c^\gamma(M)$, by part (ii) of Proposition 5.1. Using this in combination with Theorem 5.2, we obtain $\hat{\mathscr{P}}(\gamma) = \mathscr{P}(\gamma) \in Core(N, c^\gamma) \subseteq Core(N, \hat{c}^\gamma)$. Thus, $\hat{\mathscr{P}}(\gamma)$ is a stable allocation for game $(N, \hat{c}^\gamma)$.

Part (ii). Let $M, L \in 2_-^N$ with $M \subseteq L$, and let $i \in M$. Note that s_M^* and s_L^* are, by assumption, integer numbers. Using part (ii) of Proposition 5.1, we obtain

$$\hat{\mathcal{P}}_{i,L}(\gamma) = \hat{c}^\gamma(L)\frac{\lambda_i}{\lambda_L} = c^\gamma(L)\frac{\lambda_i}{\lambda_L} = \mathcal{P}_{i,L}(\gamma) \leq \mathcal{P}_{i,M}(\gamma) = c^\gamma(M)\frac{\lambda_i}{\lambda_M} = \hat{c}^\gamma(M)\frac{\lambda_i}{\lambda_M} = \hat{\mathcal{P}}_{i,M}(\gamma),$$

where the inequality holds by Theorem 5.2. We conclude that $\hat{\mathcal{P}}(\gamma)$ is indeed a PMAS for game $(N, \hat{c}^\gamma)$. □

Proof of Theorem 5.4. Part (i). Let $M \in 2_-^N$. Then,

$$\begin{aligned}
\sum_{i \in M} \hat{\mathscr{P}}_i(\gamma) &= K_N^\gamma(\hat{s}_N^*) \cdot \frac{\lambda_M}{\lambda_N} \\
&\leq K_N^\gamma\Big(\Big\lceil \hat{s}_M^* \frac{\lambda_N}{\lambda_M} \Big\rceil\Big) \cdot \frac{\lambda_M}{\lambda_N} \\
&= \bigg(h_N \Big\lceil \hat{s}_M^* \frac{\lambda_N}{\lambda_M} \Big\rceil + W_q\Big(\Big\lceil \hat{s}_M^* \frac{\lambda_N}{\lambda_M} \Big\rceil, \lambda_N, \mu\Big) \cdot \lambda_N d\bigg) \cdot \frac{\lambda_M}{\lambda_N} \\
&\leq \bigg(h_N \Big\lceil \hat{s}_M^* \frac{\lambda_N}{\lambda_M} \Big\rceil + W_q\Big(\hat{s}_M^* \frac{\lambda_N}{\lambda_M}, \lambda_N, \mu\Big) \cdot \lambda_N d\bigg) \cdot \frac{\lambda_M}{\lambda_N} \\
&\leq \bigg(h_N \Big\lceil \hat{s}_M^* \frac{\lambda_N}{\lambda_M} \Big\rceil + W_q\big(\hat{s}_M^*, \lambda_M, \mu\big) \cdot \lambda_N d\bigg) \cdot \frac{\lambda_M}{\lambda_N} \\
&\leq \bigg(h_N \Big(\hat{s}_M^* \frac{\lambda_N}{\lambda_M} + 1\Big) + W_q\big(\hat{s}_M^*, \lambda_M, \mu\big) \cdot \lambda_N d\bigg) \cdot \frac{\lambda_M}{\lambda_N} \\
&= h_N \hat{s}_M^* + \sum_{i \in M} \epsilon_i + W_q\big(\hat{s}_M^*, \lambda_M, \mu\big) \cdot \lambda_M d \\
&\leq h_M \hat{s}_M^* + \sum_{i \in M} \epsilon_i + W_q\big(\hat{s}_M^*, \lambda_M, \mu\big) \cdot \lambda_M d = \hat{c}^\gamma(M) + \sum_{i \in M} \epsilon_i.
\end{aligned}$$

The first inequality holds because $\hat{s}_N^*$ is an optimal number of servers for the grand coalition. The second inequality holds by the monotonicity property of Theorem 3.4. The third inequality is due to Theorem 3.3. The final inequality holds since $h_N \leq h_M$ by assumption on γ. We conclude that $\hat{\mathscr{P}}(\gamma)$ is an ϵ-stable allocation for game $(N, \hat{c}^\gamma)$.

Part (ii). Let $M, L \in 2^N_-$ with $M \subset L$, and let $i \in M$. Then, in line with the proof of part (i) of this theorem,

$$\begin{aligned}
\hat{\mathcal{P}}_{i,L}(\gamma) &\leq K_L^\gamma\left(\left\lceil \hat{s}_M^* \frac{\lambda_L}{\lambda_M} \right\rceil\right) \cdot \frac{\lambda_i}{\lambda_L} \\
&\leq \left(h_L \left\lceil \hat{s}_M^* \frac{\lambda_L}{\lambda_M} \right\rceil + W_q(\hat{s}_M^*, \lambda_M, \mu) \cdot \lambda_L d\right) \cdot \frac{\lambda_i}{\lambda_L} \\
&\leq \left(h_L \left(\hat{s}_M^* \frac{\lambda_L}{\lambda_M} + 1\right) + W_q(\hat{s}_M^*, \lambda_M, \mu) \cdot \lambda_L d\right) \cdot \frac{\lambda_i}{\lambda_L} \\
&\leq h_M \hat{s}_M^* \frac{\lambda_i}{\lambda_M} + h_{\{i\}} + W_q(\hat{s}_M^*, \lambda_M, \mu) \cdot \lambda_i d = \hat{\mathcal{P}}_{i,M}(\gamma) + \epsilon_i,
\end{aligned}$$

where the last inequality holds since $h_L \leq h_M \leq h_{\{i\}}$ by assumption on γ and since $\lambda_i/\lambda_L \leq 1$. We conclude that $\hat{\mathcal{P}}(\gamma)$ is an ϵ-PMAS for game $(N, \hat{c}^\gamma)$. □

Working Papers Beta 2009 - 2011

nr.	Year	Title	Author(s)
352	2011	Resource pooling and cost allocation among independent service providers	Frank Karsten, Marco Slikker, Geert-Jan van Houtum
351	2011	A Framework for Business Innovation Directions	E. Lüftenegger, S. Angelov, P. Grefen
350	2011	The Road to a Business Process Architecture: An Overview of Approaches and their Use	Remco Dijkman, Irene Vanderfeesten, Hajo A. Reijers
349	2011	Effect of carbon emission regulations on transport mode selection under stochastic demand	K.M.R. Hoen, T. Tan, J.C. Fransoo G.J. van Houtum
348	2011	An improved MIP-based combinatorial approach for a multi-skill workforce scheduling problem	Murat Firat, Cor Hurkens
347	2011	An approximate approach for the joint problem of level of repair analysis and spare parts stocking	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
346	2011	Joint optimization of level of repair analysis and spare parts stocks	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
345	2011	Inventory control with manufacturing lead time flexibility	Ton G. de Kok
344	2011	Analysis of resource pooling games via a new extenstion of the Erlang loss function	Frank Karsten, Marco Slikker, Geert-Jan van Houtum
343	2011	Vehicle refueling with limited resources	Murat Firat, C.A.J. Hurkens, Gerhard J. Woeginger
342	2011	Optimal Inventory Policies with Non-stationary Supply Disruptions and Advance Supply Information	Bilge Atasoy, Refik Güllü, TarkanTan
341	2011	Redundancy Optimization for Critical Components in High-Availability Capital Goods	Kurtulus Baris Öner, Alan Scheller-Wolf Geert-Jan van Houtum
339	2010	Analysis of a two-echelon inventory system with two supply modes	Joachim Arts, Gudrun Kiesmüller
338	2010	Analysis of the dial-a-ride problem of Hunsaker and Savelsbergh	Murat Firat, Gerhard J. Woeginger

335	2010	Attaining stability in multi-skill workforce scheduling	Murat Firat, Cor Hurkens
334	2010	Flexible Heuristics Miner (FHM)	A.J.M.M. Weijters, J.T.S. Ribeiro
333	2010	An exact approach for relating recovering surgical patient workload to the master surgical schedule	P.T. Vanberkel, R.J. Boucherie, E.W. Hans, J.L. Hurink, W.A.M. van Lent, W.H. van Harten
332	2010	Efficiency evaluation for pooling resources in health care	Peter T. Vanberkel, Richard J. Boucherie, Erwin W. Hans, Johann L. Hurink, Nelly Litvak
331	2010	The Effect of Workload Constraints in Mathematical Programming Models for Production Planning	M.M. Jansen, A.G. de Kok, I.J.B.F. Adan
330	2010	Using pipeline information in a multi-echelon spare parts inventory system	Christian Howard, Ingrid Reijnen, Johan Marklund, Tarkan Tan
329	2010	Reducing costs of repairable spare parts supply systems via dynamic scheduling	H.G.H. Tiemessen, G.J. van Houtum
328	2010	Identification of Employment Concentration and Specialization Areas: Theory and Application	F.P. van den Heuvel, P.W. de Langen, K.H. van Donselaar, J.C. Fransoo
327	2010	A combinatorial approach to multi-skill workforce scheduling	Murat Firat, Cor Hurkens
326	2010	Stability in multi-skill workforce scheduling	Murat Firat, Cor Hurkens, Alexandre Laugier
325	2010	Maintenance spare parts planning and control: A framework for control and agenda for future research	M.A. Driessen, J.J. Arts, G.J. v. Houtum, W.D. Rustenburg, B. Huisman
324	2010	Near-optimal heuristics to set base stock levels in a two-echelon distribution network	R.J.I. Basten, G.J. van Houtum
323	2010	Inventory reduction in spare part networks by selective throughput time reduction	M.C. van der Heijden, E.M. Alvarez, J.M.J. Schutten
322	2010	The selective use of emergency shipments for service-contract differentiation	E.M. Alvarez, M.C. van der Heijden, W.H. Zijm

321	2010	Heuristics for Multi-Item Two-Echelon Spare Parts Inventory Control Problem with Batch Ordering in the Central Warehouse	B. Walrave, K. v. Oorschot, A.G.L. Romme
320	2010	Preventing or escaping the suppression mechanism: intervention conditions	Nico Dellaert, Jully Jeunet.
319	2010	Hospital admission planning to optimize major resources utilization under uncertainty	R. Seguel, R. Eshuis, P. Grefen.
318	2010	Minimal Protocol Adaptors for Interacting Services	Tom Van Woensel, Marshall L. Fisher, Jan C. Fransoo.
317	2010	Teaching Retail Operations in Business and Engineering Schools	Lydie P.M. Smets, Geert-Jan van Houtum, Fred Langerak.
316	2010	Design for Availability: Creating Value for Manufacturers and Customers	Pieter van Gorp, Rik Eshuis.
315	2010	Transforming Process Models: executable rewrite rules versus a formalized Java program	Bob Walrave, Kim E. van Oorschot, A. Georges L. Romme
314	2010	Getting trapped in the suppression of exploration: A simulation model	S. Dabia, T. van Woensel, A.G. de Kok
313	2010	A Dynamic Programming Approach to Multi-Objective Time-Dependent Capacitated Single Vehicle Routing Problems with Time Windows	
	2010		
312	2010	Tales of a So(u)rcerer: Optimal Sourcing Decisions Under Alternative Capacitated Suppliers and General Cost Structures	Osman Alp, Tarkan Tan
311	2010	In-store replenishment procedures for perishable inventory in a retail environment with handling costs and storage constraints	R.A.C.M. Broekmeulen, C.H.M. Bakx
310	2010	The state of the art of innovation-driven business models in the financial services industry	E. Lüftenegger, S. Angelov, E. van der Linden, P. Grefen
309	2010	Design of Complex Architectures Using a Three Dimension Approach: the CrossWork Case	R. Seguel, P. Grefen, R. Eshuis
308	2010	Effect of carbon emission regulations on transport mode selection in supply chains	K.M.R. Hoen, T. Tan, J.C. Fransoo, G.J. van Houtum
307	2010	Interaction between intelligent agent strategies for real-time transportation planning	Martijn Mes, Matthieu van der Heijden, Peter Schuur
306	2010	Internal Slackening Scoring Methods	Marco Slikker, Peter Borm, René van den Brink

305	2010	Vehicle Routing with Traffic Congestion and Drivers' Driving and Working Rules	A.L. Kok, E.W. Hans, J.M.J. Schutten, W.H.M. Zijm
304	2010	Practical extensions to the level of repair analysis	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
303	2010	Ocean Container Transport: An Underestimated and Critical Link in Global Supply Chain Performance	Jan C. Fransoo, Chung-Yee Lee
302	2010	Capacity reservation and utilization for a manufacturer with uncertain capacity and demand	Y. Boulaksil; J.C. Fransoo; T. Tan
300	2009	Spare parts inventory pooling games	F.J.P. Karsten; M. Slikker; G.J. van Houtum
299	2009	Capacity flexibility allocation in an outsourced supply chain with reservation	Y. Boulaksil, M. Grunow, J.C. Fransoo
298	2010	An optimal approach for the joint problem of level of repair analysis and spare parts stocking	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
297	2009	Responding to the Lehman Wave: Sales Forecasting and Supply Management during the Credit Crisis	Robert Peels, Maximiliano Udenio, Jan C. Fransoo, Marcel Wolfs, Tom Hendrikx
296	2009	An exact approach for relating recovering surgical patient workload to the master surgical schedule	Peter T. Vanberkel, Richard J. Boucherie, Erwin W. Hans, Johann L. Hurink, Wineke A.M. van Lent, Wim H. van Harten
295	2009	An iterative method for the simultaneous optimization of repair decisions and spare parts stocks	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
294	2009	Fujaba hits the Wall(-e)	Pieter van Gorp, Ruben Jubeh, Bernhard Grusie, Anne Keller
293	2009	Implementation of a Healthcare Process in Four Different Workflow Systems	R.S. Mans, W.M.P. van der Aalst, N.C. Russell, P.J.M. Bakker
292	2009	Business Process Model Repositories - Framework and Survey	Zhiqiang Yan, Remco Dijkman, Paul Grefen
291	2009	Efficient Optimization of the Dual-Index Policy Using Markov Chains	Joachim Arts, Marcel van Vuuren, Gudrun Kiesmuller
290	2009	Hierarchical Knowledge-Gradient for Sequential Sampling	Martijn R.K. Mes; Warren B. Powell; Peter I. Frazier
289	2009	Analyzing combined vehicle routing and break scheduling from a distributed decision making perspective	C.M. Meyer; A.L. Kok; H. Kopfer; J.M.J. Schutten
288	2009	Anticipation of lead time performance in Supply Chain Operations Planning	Michiel Jansen; Ton G. de Kok; Jan C. Fransoo
287	2009	Inventory Models with Lateral Transshipments: A Review	Colin Paterson; Gudrun Kiesmuller; Ruud Teunter; Kevin Glazebrook
286	2009	Efficiency evaluation for pooling resources in health care	P.T. Vanberkel; R.J. Boucherie; E.W. Hans; J.L. Hurink; N. Litvak
285	2009	A Survey of Health Care Models that Encompass Multiple Departments	P.T. Vanberkel; R.J. Boucherie; E.W. Hans; J.L. Hurink; N. Litvak

284	2009	Supporting Process Control in Business Collaborations	S. Angelov; K. Vidyasankar; J. Vonk; P. Grefen
283	2009	Inventory Control with Partial Batch Ordering	O. Alp; W.T. Huh; T. Tan
282	2009	Translating Safe Petri Nets to Statecharts in a Structure-Preserving Way	R. Eshuis
281	2009	The link between product data model and process model	J.J.C.L. Vogelaar; H.A. Reijers
280	2009	Inventory planning for spare parts networks with delivery time requirements	I.C. Reijnen; T. Tan; G.J. van Houtum
279	2009	Co-Evolution of Demand and Supply under Competition	B. Vermeulen; A.G. de Kok
278	2010	Toward Meso-level Product-Market Network Indices for Strategic Product Selection and (Re)Design Guidelines over the Product Life-Cycle	B. Vermeulen, A.G. de Kok
277	2009	An Efficient Method to Construct Minimal Protocol Adaptors	R. Seguel, R. Eshuis, P. Grefen
276	2009	Coordinating Supply Chains: a Bilevel Programming Approach	Ton G. de Kok, Gabriella Muratore
275	2009	Inventory redistribution for fashion products under demand parameter update	G.P. Kiesmuller, S. Minner
274	2009	Comparing Markov chains: Combining aggregation and precedence relations applied to sets of states	A. Busic, I.M.H. Vliegen, A. Scheller-Wolf
273	2009	Separate tools or tool kits: an exploratory study of engineers' preferences	I.M.H. Vliegen, P.A.M. Kleingeld, G.J. van Houtum
272	2009	An Exact Solution Procedure for Multi-Item Two-Echelon Spare Parts Inventory Control Problem with Batch Ordering	Engin Topan, Z. Pelin Bayindir, Tarkan Tan
271	2009	Distributed Decision Making in Combined Vehicle Routing and Break Scheduling	C.M. Meyer, H. Kopfer, A.L. Kok, M. Schutten
270	2009	Dynamic Programming Algorithm for the Vehicle Routing Problem with Time Windows and EC Social Legislation	A.L. Kok, C.M. Meyer, H. Kopfer, J.M.J. Schutten
269	2009	Similarity of Business Process Models: Metics and Evaluation	Remco Dijkman, Marlon Dumas, Boudewijn van Dongen, Reina Kaarik, Jan Mendling
267	2009	Vehicle routing under time-dependent travel times: the impact of congestion avoidance	A.L. Kok, E.W. Hans, J.M.J. Schutten
266	2009	Restricted dynamic programming: a flexible framework for solving realistic VRPs	J. Gromicho; J.J. van Hoorn; A.L. Kok; J.M.J. Schutten;

Working Papers published before 2009 see: http://beta.ieis.tue.nl