Analysis of resource pooling games via a new extension of the Erlang loss function

Document status and date:
Published: 01/01/2011

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Analysis of resource pooling games via
a new extension of the Erlang loss function

Frank Karsten, Marco Slikker, Geert-Jan van Houtum

Analysis of resource pooling games via a new extension of the Erlang loss function

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

March 29, 2011

Abstract

We study a situation where several independent service providers collaborate by pooling their resources into a joint service system. These service providers may represent such diverse organizations as hospitals that pool intensive care beds and ambulances, airline companies that share spare parts, or car rental agencies that pool rental cars. We model the service systems as Erlang loss systems that face a fixed cost rate per server and penalty costs for lost customers. We examine the allocation of costs of the pooled system amongst the participants by formulating a cooperative cost game in which each coalition optimizes the number of servers. We identify a cost allocation that is in the core of this game, giving no subset of players an incentive to split off and form a separate pooling group. Moreover, we axiomatically characterize this allocation rule and show that it can be reached through a population monotonic allocation scheme. To obtain these results, we introduce a new extension of the classic Erlang loss function to non-integral numbers of servers and establish several of its structural properties.

Keywords: Games/group decisions: cooperative; probability: stochastic model applications; inventory/production: uncertainty.

---

[*]Corresponding author. Tel.: +31-40-247-5163. E-mail address: g.j.v.houtum@tue.nl.

1 Introduction

In today's business world, service providers are increasingly recognizing the effectiveness of resource pooling to drastically reduce the costs of maintaining their service systems. Resource pooling refers to an arrangement in which a group of common resources is held for several customer classes rather than dedicated resources for each individual customer class. An important benefit of resource pooling is a reduction in the number of customers who are blocked due to unavailability of the requested resources (Smith and Whitt, 1981). Another important benefit is that, in setting up a cluster of pooled resources, economies of scale can be exploited that typically prevail in acquiring and maintaining resources. These benefits translate to substantial cost savings. Such savings are seen in many places in the society in case multiple customer streams are served by one common service provider. But they 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 that collaborate by pooling their resources into a joint service system. For instance, consider a hospital that is composed of several independent clinical departments, each with their own patient populations. Such departments commonly share operating rooms, intensive care beds, and surgeons. Another example is that of airline companies pooling inventories of critical spare components. One can also think of independent car rental agencies that collaborate by holding a common set of rental cars for all their customers. A final example is that of several cities operating a joint fire department. In all these cases, collaboration among service providers enables more efficient use of their resources, which is beneficial to the system as a whole.

The next natural question to be asked is how these service providers should allocate the joint costs of the pooled system amongst them. Indeed, the problem of constructing fair allocation mechanisms is one of the most severe impediments for horizontal cooperation in the logistics industry (Cruijssen et al., 2007). In this paper, we will apply concepts from cooperative game theory, which allows for enforceable binding agreements and transfer payments between players, to identify a stable cost allocation scheme. In cooperative game theory, stability is one of the the most prominent notions of fairness. Under our stable allocation, each group of service providers will feel that pooling their resources with others leads to a cost saving for themselves and, moreover, parties with a large customer load will pay a larger share of the costs.

We model the service systems as Erlang loss systems (M/G/c/c systems) that face resource costs for servers and penalty costs for customers that find all servers occupied

upon arrival. The participating parties, which we will simply refer to as players, each face their own Poisson stream of customers. The players may collaborate by complete pooling of their servers and individual customer streams. We formulate a cooperative cost game with transferable utility in which each coalition optimizes their expected joint costs over an infinite time horizon by choosing a cost-minimizing number of common servers.

The study of cooperation in operations research problems is a fruitful and interesting topic that is drawing growing attention. The applications of cooperative game theory to operations research problems were surveyed at the start of this century by Borm et al. (2001). Various other categories of cooperative games have recently been considered as well; we mention inventory management games (reviewed by Fiestras-Janeiro et al., 2010) and cooperation involving combinatorial structures (see, e.g., Bogomolnaia et al., 2010, and references therein). The literature on the application of cooperative game theory to queueing situations, characterized by stochastic uncertainty over an infinite time horizon, is relatively sparse. This literature can roughly be divided in single-server queueing games and multi-server queueing games. We now survey each category separately.

In single-server queueing games, a coalition of players operates an M/M/1 queueing system with an adjustable service rate. González and Herrero (2004) deal with a cost allocation problem in which each player has a pre-specified service criterion on the mean sojourn time of their customers. A coalition of players chooses its service rate to satisfy these sojourn time constraints. 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. Anily and Haviv (2010) study a model in which each player has its own potential service rate. A coalition of players sets its service rate at a weighted sum of the potential service rates of its members and subsequently has to divide congestion costs. Finally, Yu et al. (2009) consider a situation where a coalition of players optimizes its service rate to minimize delay and capacity costs, possibly subject to service level constraints.

There are two key differences between our work in the context of Erlang loss systems and the cooperative games associated with pooled M/M/1 queues. The first difference is that we focus on situations in which a customer who finds all servers busy upon arrival is served elsewhere at additional costs because waiting for service is not an acceptable option. Secondly, we consider service systems with possibly more than one server. In contrast to the studies on M/M/1 games, we do not assume that the service rate of a single server can be set at arbitrary levels. Instead, we assume that additional servers have to be acquired to increase the service capacity. In fact, the real life examples mentioned

above can be accurately modeled as Erlang loss systems, whereas an M/M/1 model would not fit. Thus, our work provides novel insights that are both practically relevant and of theoretical importance.

In multi-server queueing games, a coalition of players operates a stochastic system with possibly more than one server. Wong et al. (2007) describe cost allocation policies in the context of spare parts inventory pooling. The spare parts take the role of the servers in their Markovian model. They analyze their policies by means of a small numerical experiment. McLean and Sharkey (1993) consider a multi-channel telecommunication model. They introduce several cost allocation methodologies and illustrate them with numerical examples. However, neither of these two papers investigate analytical properties of the underlying game or stability of allocation rules, whereas the main focus of our present study is to derive such structural results. Timmer and Scheinhardt (2010) analyze a queueing model with several single-server stations in tandem. The various stations may cooperate by redistributing their combined server capacity. Their setting with multiple servers in series is different, however, from our setting with multiple parallel servers. Finally, Karsten et al. (2009) consider a cooperative game that is similar to ours: they model service systems as Erlang loss systems as well. In the model considered in Karsten et al. (2009), each player is associated with a fixed number of servers, and each coalition uses the sum of the servers of its members. In contrast, we consider a setting where a coalition picks an *optimal* number of servers. This allows additional benefits of collaboration and leads to a different type of game, which we will call resource pooling game.

In our analysis of resource pooling games, we consider the cost allocation rule that divides joint costs proportional to the arrival rate of customers belonging to each player. This rule is easy to understand and would be easy to implement in practice. We prove that this cost allocation rule has several appealing properties. First, it accomplishes a stable cost division as it gives no subset of players an incentive to split off and form a separate pooling group. Second, it can be reached through a population monotonic allocation scheme under which the costs faced by a player do not increase when the coalition to which he belongs grows. Finally, our rule can be axiomatically characterized as the unique cost allocation rule on resource pooling situations that is immune to manipulations of the players via artificial splitting or merging, while still satisfying a weak continuity property.

To obtain these structural results, we first prove several analytical characteristics of a new extension of the classic Erlang loss function to non-integral numbers of servers. As such, our work is related to the literature on performance analysis and optimization of loss systems with overflow layers. Traditional approximation techniques for such systems, such as the equivalent random method (Wilkinson, 1956) and Hayward's approximation

4

(Fredericks, 1980), also employ functions that extend the domain of the Erlang loss function. However, the extension that is commonly used in these approximation techniques, the integral representation in terms of the Gamma function (see, e.g., Jagerman, 1974), does not satisfy certain characteristics, while existence of an extension of the Erlang loss function satisfying those characteristics is sufficient to prove the aforementioned properties of the proportional allocation. With this purpose in mind, we develop a new extension that does satisfy the desired characteristics.

The model, results and analysis we present in this paper make several primary contributions. Firstly, we introduce a new class of operations research games, called resource pooling games. This contributes to the literature on cooperative games in the form of a rich modeling framework. Secondly, for this game, we identify a simple, yet stable cost allocation. Thirdly, we show that this allocation can be reached through a population monotonic allocation scheme. Fourthly, to derive this, we introduce a new extension of the Erlang loss function and establish several of its properties, which contributes to the literature on the mathematics of loss systems. Finally, we prove that there is only one allocation rule for our model that satisfies natural axioms of non-manipulability and continuity: our proportional rule.

The remainder of this paper is organized as follows. We start in Section 2 with some preliminaries on cooperative game theory and the Erlang loss function. Then, in Section 3, we describe our model, introduce resource pooling games, and propose the proportional cost allocation rule. Section 4 describes our approach to prove that resulting allocations are stable, which requires our new extension of the Erlang loss function. Afterwards, we consider population monotonicity in Section 5 and the axiomatic characterization in Section 6. Finally, we draw conclusions in Section 7.

2 Preliminaries

For reasons of self-containedness, we first give a brief introduction to cooperative game theory. Subsequently, we present the well-known Erlang loss function along with several of its properties, and we introduce the concept of scalability.

2.1 Preliminaries on cooperative game theory

A cooperative cost game with transferable utility, which we will simply refer to as *game*, is a pair (N, c) where N is a nonempty finite set of *players* and $c : 2^N \to \mathbb{R}$ is the *characteristic cost function*, which assigns to every *coalition* $M \subseteq N$ its costs $c(M)$ with

$c(\emptyset) = 0$. The value $c(M)$ is interpreted as the total costs of the joint cooperative effort if only the players in coalition M are involved in it. In particular, $c(N)$ represents the total costs for the *grand coalition* N when all players agree on working together. In this paper, we will consider so-called subadditive games in which it is always beneficial to combine coalitions: a game is called *subadditive* if for any two disjoint coalitions M and R it holds that $c(M) + c(R) \geq c(M \cup R)$. In a subadditive game, cooperation by the grand coalition is socially optimal.

A central problem in cooperative game theory is to allocate $c(N)$ to the individual players in a fair way. Formally, an *allocation* is a vector $x \in \mathbb{R}^N$ satisfying $\sum_{i \in N} x_i = c(N)$. The value x_i is interpreted as the costs assigned to player i. An allocation is called *stable* if $\sum_{i \in M} x_i \leq c(M)$ for all nonempty coalitions $M \in 2^N_-$, where $2^N_- = 2^N \setminus \{\emptyset\}$ denotes the power set consisting of all nonempty subsets of 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 set of all stable allocations is called the *core*, introduced by Gillies (1959). A game with a nonempty core is called *balanced*. Moreover, a game (N, c) is said to be *totally balanced* if for each nonempty coalition $R \in 2^N_-$ the subgame (R, c^R), where $c^R(M) = c(M)$ for all $M \subseteq R$, is balanced. A notion that is even stronger than total balancedness is concavity (sometimes referred to as submodularity), which is concerned with marginal utility for coalition membership (Shapley, 1971). A game is called *concave* if for each $i \in N$ and for all $M, R \subseteq N \setminus \{i\}$ with $M \subseteq R$, it holds that $c(M \cup \{i\}) - c(M) \geq c(R \cup \{i\}) - c(R)$.

The last concept that we wish to introduce is a population monotonic allocation scheme. 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, $y_{i,M} \geq y_{i,R}$ for all coalitions $M, R \in 2^N_-$ with $M \subseteq R$ and $i \in M$. If the game admits a PMAS, say y, then the game is totally balanced and $(y_{i,N})_{i \in N}$ is an element of its core (Sprumont, 1990).

2.2 Preliminaries on the Erlang loss function and scalability

The Erlang loss function is a classic in queueing theory. To describe this function, let $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$ denote the set of all nonnegative integers and let $\mathbb{R}_{++} = \mathbb{R}_+ \setminus \{0\}$ denote the set of all positive real numbers. Then, for any $s \in \mathbb{N}_0$ and $a \in \mathbb{R}_{++}$, the *Erlang loss*

function $B : \mathbb{N}_0 \times \mathbb{R}_{++} \to [0, 1]$ is defined by

$$B(s, a) = \frac{a^s/s!}{\sum_{y=0}^{s} a^y/y!}. \tag{1}$$

$B(s, a)$ may be interpreted as the steady-state probability that an arriving customer finds no free server in an *Erlang loss system* (i.e., an $M/G/s/s$ system) with s servers and offered load a. In such a system, the mean service time is $\tau > 0$ and customers arrive according to a Poisson process with rate $a/\tau > 0$. Since customers that find no free server upon arrival are often called *blocked* customers, $B(s, a)$ is also sometimes referred to as the *blocking probability*.

Smith and Whitt (1981) were the first to show that pooling the resources of several Erlang loss systems is beneficial due to an economies-of-scale effect satisfied by the Erlang loss function. That is, when we increase the offered load a and the number of servers s with the same relative amount t – for instance, by combining the servers and arrival streams of two symmetric Erlang loss systems into a single joint system – the blocking probability does not increase. Note that this is only meaningful when the scaling factor t is chosen such that the number of servers in the scaled system, ts, is in $\mathbb{N}_0$. We describe this scalability property more precisely and in more general terms in the following definition.

Definition 2.1. A function $f : D \to \mathbb{R}$, with $D \subseteq \mathbb{R}_+^n$ for some $n \in \mathbb{N}$, is said to be *upward scalable* if, for each element of its domain $x \in D$, the associated scaled function $f^{(x)} : \{t \in \mathbb{R} : tx \in D\} \to \mathbb{R}$, defined by $f^{(x)}(t) = f(tx)$, is non-increasing. $\diamondsuit$

In particular, the Erlang loss function B is upward scalable; a proof of which can be found in the appendix of Smith and Whitt (1981). The Erlang loss function satisfies several other useful properties as well. The properties that we will use in this paper's analysis are collected in the following theorem.

Theorem 2.1. *Fix $s, s' \in \mathbb{N}$ and $a, \tau, \lambda, \lambda' \in \mathbb{R}_{++}$. Then:*
(i) $B(s + s', (\lambda + \lambda')\tau) \cdot (\lambda + \lambda') \le B(s, \lambda\tau) \cdot \lambda + B(s', \lambda'\tau) \cdot \lambda'$.
(ii) $B(s, \lambda/\mu)$ *is decreasing and convex in μ on domain $\mathbb{R}_{++}$ for μ.*
(iii) $\dfrac{\partial B}{\partial a}(s, a) = [B(s, a) - 1 + s/a] \cdot B(s, a)$.
(iv) $B(s, a) = \dfrac{aB(s - 1, a)}{(aB(s - 1, a) + s)}$.

The subadditivity property, (i), is due to Theorem 1 in Smith and Whitt (1981). Property (ii) corresponds to Proposition 3 in Harel (1990). The expression for the partial derivative in (iii) is due to Theorem 15 in Jagerman (1974). Finally, the recursive relation of property (iv) is well-known (see, e.g., Jagerman, 1974, p. 531).

3 Model

In this section, we introduce resource pooling situations and define the associated games, called resource pooling games.

3.1 Resource pooling situations

Consider a situation where several players require a number of servers for their customer populations. Each player's customers arrive to a service system according to mutually independent Poisson processes. Service times are independent and identically distributed with some general distribution function (with finite mean). Each newly arriving customer immediately goes into service if there is an unoccupied server available. Conversely, if that customer finds no free server, he is lost and some penalty costs have to be paid.

We assume throughout that players are interested in their long-term average costs per time unit. Naturally, penalty costs can be reduced by increasing the number of servers. However, holding a large cluster of servers is costly as well. Thus, a natural trade-off arises between resource costs for those servers and penalty costs. The following example illustrates this problem in a medical context.

Example 3.1. Consider a single clinical department (player) in a hospital. New patients (customers) arrive with rate $\bar{\lambda} = \frac{1}{10}$ and their average length of stay (service time) is $\tau = 1$. If there is no bed (server) available when a patient arrives, he is transferred to another hospital and treated there. The costs charged to the clinical department for this emergency transferral (penalty costs) are $p = 15$, which may additionally encompass goodwill loss and/or fines imposed by the government. The maintenance and capital costs for one bed (resource costs) are $h = 1$ per unit time.

If this department would operate a system with $s \in \mathbb{N}_0$ beds, then the steady-state blocking probability would be described by the Erlang loss function $B(s, \bar{\lambda}\tau)$. Hence, the expected relevant costs per unit time in steady state would add up to $hs + B(s, \bar{\lambda}\tau) \cdot \bar{\lambda}p$. Given that $B(0, \frac{1}{10}) = 1$, $B(1, \frac{1}{10}) = \frac{1}{11}$, and $B(2, \frac{1}{10}) = \frac{1}{221}$, it is easily verified that the optimal number of servers is $S^* = 1$ with associated minimal costs equal to $1 \cdot 1 + \frac{1}{11} \cdot \frac{1}{10} \cdot 15 = 1\frac{3}{22}$. $\diamond$

In a setting with more than one player, the players could reduce their penalty costs by pooling their resources. Furthermore, by acquiring and maintaining resources collaboratively, economies of scale can be exploited. Moreover, the players could benefit by re-optimizing the number of servers in their joint system. To analyze this, we define a *resource pooling situation* as a tuple (N, λ, τ, H, p), where

- N is the nonempty finite set of players;

- $\lambda \in \mathbb{R}^N_{++}$ is the vector of arrival rates, i.e., $\lambda_i > 0$ denotes the arrival rate of customers that belong to player $i \in N$;

- $\tau > 0$ is the mean service time for an arbitrary customer of any player;

- $H : \mathbb{N}_0 \to \mathbb{R}_+$ is a concave increasing unbounded function specifying that the resource costs for holding s servers are $H(s)$ per unit time; and

- $p > 0$ is the expected penalty costs that are incurred whenever a customer is blocked.

This model covers a broad range of situations in which resource sharing can occur between independent service providers, i.e., situations where collaboration between separate Erlang loss systems is possible. In our analysis later on, it will be convenient to start by considering situations with linear resource cost functions. To accommodate this, we call a resource pooling situation (N, λ, τ, H, p) a *linear* resource pooling situation if there exists an $h \in \mathbb{R}_+$ such that $H(s) = hs$ for all $s \in \mathbb{N}_0$. For convenience, we will represent such a situation directly via the tuple (N, λ, τ, h, p).

3.2 Resource pooling games

The cooperation between players takes place as follows. Let $\varphi = (N, \lambda, \tau, H, p)$ be a resource pooling situation and consider a coalition $M \in 2^N_-$. The players in this coalition collaborate by complete pooling of their 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 Poisson arrival process with merged rate $\lambda_M = \sum_{i \in M} \lambda_i$. We assume that the common servers can handle all types of customers with equal ease.

Based on these assumptions, this joint service facility behaves as an Erlang loss system. For any particular choice of the number of servers in the joint system $s \in \mathbb{N}_0$, we can calculate the expected relevant costs per unit time in steady state as

$$K_M(s) = H(s) + B(s, \lambda_M \tau) \cdot \lambda_M p. \tag{2}$$

Since the value of the Erlang loss function is confined to the interval $(0, 1]$, whereas the resource cost function H increases unboundedly, there exists an optimal number of servers, which can be found by an enumerative search procedure. Assuming that any coalition indeed picks a cost minimizing number of servers, we can formulate a game corresponding to resource pooling situation φ. We call the game (N, c^φ), with

$$c^\varphi(M) = \min_{s \in \mathbb{N}_0} K_M(s) \tag{3}$$

for all coalitions $M \in 2^N_-$ and $c^\varphi(\emptyset) = 0$, the associated *resource pooling game*.

For any coalition $M \in 2^N_-$, there may be multiple cost minimizing numbers of servers, so to avoid ambiguity we define a specific optimal number of servers by

$$S^*_M = \min\{s \in \mathbb{N}_0 : K_M(s) = c^\varphi(M)\}. \tag{4}$$

The following theorem shows that cooperation in the context of resource pooling always leads to a reduction in costs.

Theorem 3.1. *Resource pooling games are subadditive.*

Proof. Let $\varphi = (N, \lambda, \tau, H, p)$ be a resource pooling situation. Let $M, R \in 2^N_-$ with $M \cap R = \emptyset$. Then,

$$
\begin{aligned}
c^\varphi(M \cup R) &= K_M(S^*_{M \cup R}) \\
&\leq K_M(S^*_M + S^*_R) \\
&= H(S^*_M + S^*_R) + B\left(S^*_M + S^*_R, (\lambda_M + \lambda_R)\tau\right) \cdot (\lambda_M + \lambda_R)p \\
&\leq H(S^*_M) + H(S^*_R) + B\left(S^*_M + S^*_R, (\lambda_M + \lambda_R)\tau\right) \cdot (\lambda_M + \lambda_R)p \\
&\leq H(S^*_M) + H(S^*_R) + B\left(S^*_M, \lambda_M\tau\right) \cdot \lambda_M p + B\left(S^*_R, \lambda_R\tau\right) \cdot \lambda_R p \\
&= c^\varphi(M) + c^\varphi(R).
\end{aligned}
$$

The first inequality holds because $S^*_{M \cup R}$ is a cost minimizing base stock level for coalition $M \cup R$. The second inequality is valid since H is a concave increasing function. The third inequality holds by part (i) of Theorem 2.1. This completes the proof. $\square$

Although Theorem 3.1 shows that collaboration among all players is beneficial, it does not answer the question of how the total costs of the grand coalition should be allocated to the individual players. To study this, we define a *cost allocation rule on resource pooling situations* (or *rule* for short) as a mapping, say f, that assigns to any resource pooling situation $\varphi = (N, \lambda, \tau, H, p)$ a vector in $\mathbb{R}^N$ such that $\sum_{i \in N} f_i(\varphi) = c^\varphi(N)$. A simple rule would be to divide joint costs proportional to the arrival rate of each player. Formally, we define for any resource pooling situation $\varphi = (N, \lambda, \tau, H, p)$ this rule $\mathcal{A}$ by

$$\mathcal{A}_i(\varphi) = c^\varphi(N) \cdot \lambda_i/\lambda_N \tag{5}$$

for each player $i \in N$. If costs are shared according to this rule, then a player with more frequent customer arrivals pays a greater share of the costs, which seems reasonable. The following example illustrates this rule numerically and shows that resource pooling games need not be concave.

Coalition M	S_M^*	$B(S_M^*, \lambda_M \tau)$	$c^\varphi(M)$
$\{1\}, \{2\}$	1	$\frac{1}{11}$	$1\frac{3}{22}$
$\{3\}$	3	$\frac{32}{827}$	$3\frac{384}{827}$
$\{1, 2\}$	1	$\frac{1}{6}$	$1\frac{1}{2}$
$\{1, 3\}, \{2, 3\}$	3	$\frac{243}{4853}$	$3\frac{6561}{9706}$
$\{1, 2, 3\}$	3	$\frac{1}{16}$	$3\frac{15}{16}$

Table 1: *Parameter values and characteristic cost function for each coalition in Example 3.2.*

Example 3.2. Consider $\varphi = (N, \lambda, \tau, H, p)$, a 3-player resource pooling situation, with $N = \{1, 2, 3\}$, $\lambda_1 = \frac{1}{10}$, $\lambda_2 = \frac{1}{10}$, $\lambda_3 = \frac{8}{10}$, $\tau = 1$, $H(s) = s$ for all $s \in \mathbb{N}_0$, and $p = 15$. As player 1 alone faces the same problem as in Example 3.1, $S_{\{1\}}^* = 1$, $B(S_{\{1\}}^*, \frac{1}{10}) = \frac{1}{11}$, and $c^\varphi(\{1\}) = 1\frac{3}{22}$. We can similarly determine the optimal number of servers, associated blocking probability, and characteristic costs for all other coalitions; all values are shown in Table 1. The cost allocation proportional to arrival rates $\mathcal{A}(\varphi)$ assigns $\frac{63}{160}$ to both players 1 and 2 and $3\frac{24}{160}$ to player 3. It is easy to verify that this cost allocation is stable. Hence, the resource pooling game (N, c^φ) is balanced. However, this game is not concave, since $c^\varphi(\{1, 3\}) - c^\varphi(\{3\}) = 3\frac{6561}{9706} - 3\frac{384}{827} < 0.212 < 0.261 < 3\frac{15}{16} - 3\frac{6561}{9706} = c^\varphi(\{1, 2, 3\}) - c^\varphi(\{2, 3\})$. In other words, player 1's marginal cost contribution may increase if he joins a larger coalition. $\Diamond$

4 Analysis of allocation stability and extensions of the Erlang loss function

Rule $\mathcal{A}$ resulted in a core element in Example 3.2. In this section, we describe our approach to prove that this holds in general for linear resource pooling situations, i.e., situations with linear resource cost functions. In Section 5, we will generalize this result to resource pooling situations, i.e., situations that allow general concave resource cost functions. We focus on linear situations in the current section for notational convenience; this focus allows us to emphasize and intuitively describe the key challenge of finding an extension of the Erlang loss function with certain properties.

4.1 Proof approach

The proof that $\mathcal{A}$, the cost allocation rule proportional to arrival rates, results in core elements is based on the upward scalability property that was introduced in Section 2.2. The main step in the proof can be described as follows. Consider a linear resource pooling situation $\varphi = (N, \lambda, \tau, h, p)$ such that $\frac{\lambda_N}{\lambda_M} S_M^* \in \mathbb{N}_0$ for some coalition M. Then, by scaling down the number of servers and the offered load by the same fraction $\frac{\lambda_M}{\lambda_N}$, we obtain

$$B\left(\frac{\lambda_N}{\lambda_M} S_M^*, \lambda_N \tau\right) \leq B\left(S_M^*, \lambda_M \tau\right). \tag{6}$$

This inequality, which holds because the Erlang loss function B is upward scalable, is crucial in our proof. It is predicated, however, on the assumption that $\frac{\lambda_N}{\lambda_M} S_M^* \in \mathbb{N}_0$, which need not be true. Indeed, if $\frac{\lambda_N}{\lambda_M} S_M^* \notin \mathbb{N}_0$, then inequality (6) is not valid since $\frac{\lambda_N}{\lambda_M} S_M^*$ is not an admissible number of servers and, moreover, $B\left(\frac{\lambda_N}{\lambda_M} S_M^*, \lambda_N \tau\right)$ is not defined. A seemingly obvious way to deal with this problem is to consider a function that extends the domain of the Erlang loss function to non-integral numbers of servers and that coincides with the Erlang loss function whenever the number of servers is an integer. Formally, any function $E : \mathbb{R}_+ \times \mathbb{R}_{++} \to [0, 1]$ is called an *extension of the Erlang loss function* (or *extension* for short) if $E(s, a) = B(s, a)$ for all $s \in \mathbb{N}_0$ and $a \in \mathbb{R}_{++}$.

However, for such an extension to be helpful, it needs to satisfy several properties, which need not be satisfied by all extensions. To describe one such requirement, we define for any extension E and coalition $M \in 2_-^N$ the corresponding cost function $K_M^E : \mathbb{R}_+ \to \mathbb{R}_+$ by

$$K_M^E(s) = hs + E(s, \lambda_M \tau) \cdot \lambda_M p. \tag{7}$$

The following lemma states that existence of an extension that satisfies certain properties is sufficient to prove stability of the allocation proportional to arrival rates.

Lemma 4.1. *Let $\varphi = (N, \lambda, \tau, h, p)$ be a linear resource pooling situation. If there exists an extension of the Erlang loss function, say E, satisfying (i) upward scalability and (ii) $K_N^E(S_N^*) \leq K_N^E(\frac{\lambda_N}{\lambda_M} S_M^*)$ for all $M \in 2_-^N$, then allocation $\mathcal{A}(\varphi)$ is in the core of the associated resource pooling game (N, c^φ).*

Proof. Let E be an extension satisfying properties (i) and (ii), and let $M \in 2_-^N$ be an

arbitrary coalition. Then,

$$\sum_{i \in M} \mathcal{A}_i(\varphi) = K_N^E(S_N^*) \cdot \frac{\lambda_M}{\lambda_N}$$

$$\leq K_N^E\left(\frac{\lambda_N}{\lambda_M} S_M^*\right) \cdot \frac{\lambda_M}{\lambda_N}$$

$$= h \cdot S_M^* + E\left(\frac{\lambda_N}{\lambda_M} S_M^*, \lambda_N \tau\right) \cdot \lambda_M p$$

$$\leq h \cdot S_M^* + E\left(S_M^*, \lambda_M \tau\right) \cdot \lambda_M p$$

$$= K_M^E(S_M^*) = c^\varphi(M),$$

where the first inequality holds by property (ii) and the second inequality holds by property (i). We conclude that $\mathcal{A}(\varphi)$ is indeed a stable allocation. $\qquad\square$

Intuitively, if $\frac{\lambda_N}{\lambda_M} S_M^* \in \mathbb{N}_0$ for all coalitions M, then requirement (ii) in Lemma 4.1 should certainly hold because S_N^* is the optimal number of servers for the grand coalition, and thus the costs when using S_N^* servers cannot exceed the costs under $\frac{\lambda_N}{\lambda_M} S_M^*$ servers. However, if $\frac{\lambda_N}{\lambda_M} S_M^* \notin \mathbb{N}_0$ for some M, then requirement (ii) need not be satisfied by an arbitrary extension. The following example illustrates that a well-known extension in fact does not satisfy this requirement.

Example 4.1. Consider the function $\hat{B} : \mathbb{R}_+ \times \mathbb{R}_{++} \to [0, 1]$, defined by

$$\hat{B}(s, a) = \left(a \int_0^\infty e^{-ax}(1 + x)^s dx\right)^{-1}. \tag{8}$$

This function, which is related to the Erlang loss function via the Gamma function, is an extension (Jagerman, 1974) and is upward scalable (Smith and Whitt, 1981). In fact, it is one of the most commonly used extensions in the literature (see, e.g., Fredericks, 1980 and Jagers and Van Doorn, 1986). However, we cannot use it in an obvious way to prove stability of $\mathcal{A}$ because it fails to satisfy requirement (ii) in Lemma 4.1.

To see this, suppose that $N = \{1, 2\}$, $\lambda_1 = 0.6$, $\lambda_2 = 0.4$, $\tau = 1$, $h = 0.85$, and $p = 3$. The optimal number of servers are $S_{\{1\}}^* = 1$, $S_{\{2\}}^* = 1$, and $S_N^* = 2$. The associated optimal costs are $K_{\{1\}}(S_{\{1\}}^*) = 1\frac{21}{40}$, $K_{\{2\}}(S_{\{2\}}^*) = 1\frac{27}{140}$, and $K_N(S_N^*) = 2.3$. Then, taking $M = \{1\}$, we have $\frac{\lambda_N}{\lambda_M} S_M^* = 1\frac{2}{3}$. As such, we obtain $K_M^{\hat{B}}(\frac{\lambda_N}{\lambda_M} S_M^*) < 2.26 < 2.3 = K_N(S_N^*)$. This is graphically illustrated in Figure 1. So, the costs for the grand coalition when using $\frac{\lambda_N}{\lambda_M} S_M^*$ servers, where the associated blocking probability is interpreted via the extension $\hat{B}$, is actually smaller than the costs under the optimal integer numbers of servers. $\qquad\Diamond$
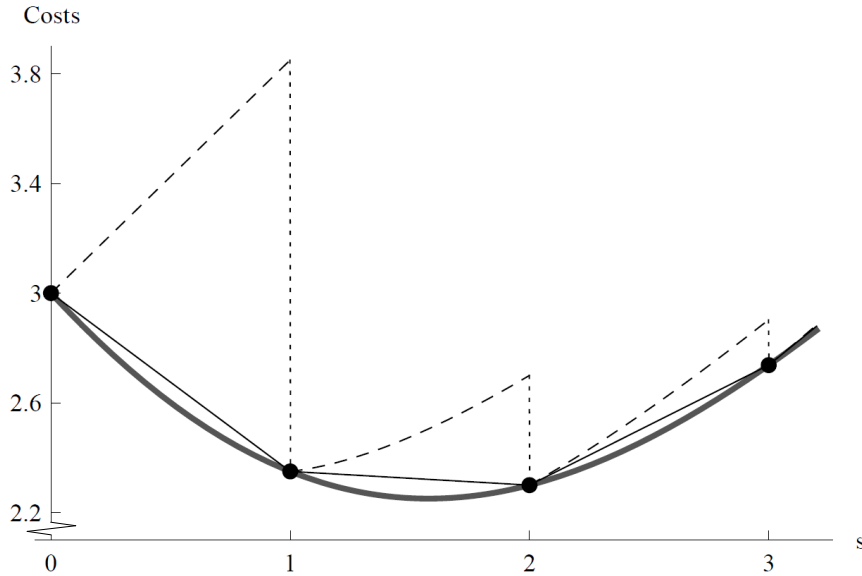
Figure 1: *The cost functions $K_N^{\hat{B}}$ (thick curve), K_N^L (straight line segments), K_N^X (dashed, discontinuous), and K_N (big dots) for the resource pooling situation described in Example 4.1.*

This example shows that we need to be careful in choosing an extension. One way to proceed would be to take an extension that trivially satisfies requirement (ii) in Lemma 4.1 and to subsequently show that it is also upward scalable. Since the linear interpolation of K_N is never below $K_N(S_N^*)$, as illustrated in Figure 1, the linear interpolation of B is an extension satisfying requirement (ii). For all $s \in \mathbb{R}_+$ and $a \in \mathbb{R}_{++}$, this piecewise linear function L is defined by

$$L(s,a) = (1 - (s - \lfloor s \rfloor)) \cdot B(\lfloor s \rfloor, a) + (s - \lfloor s \rfloor) \cdot B(\lceil s \rceil, a), \qquad (9)$$

where $\lceil s \rceil$ denotes the smallest integer larger than or equal to s, and $\lfloor s \rfloor$ denotes the largest integer smaller than or equal to s.

To obtain an extension satisfying requirements (i) and (ii) in Lemma 4.1, we recognize two possible approaches: either showing that L is upward scalable, or proving that another extension of the Erlang loss function satisfies both requirements. In the next section, we will follow the second approach, by introducing a new extension that satisfies both properties. Whether or not L is upward scalable is left for further research.
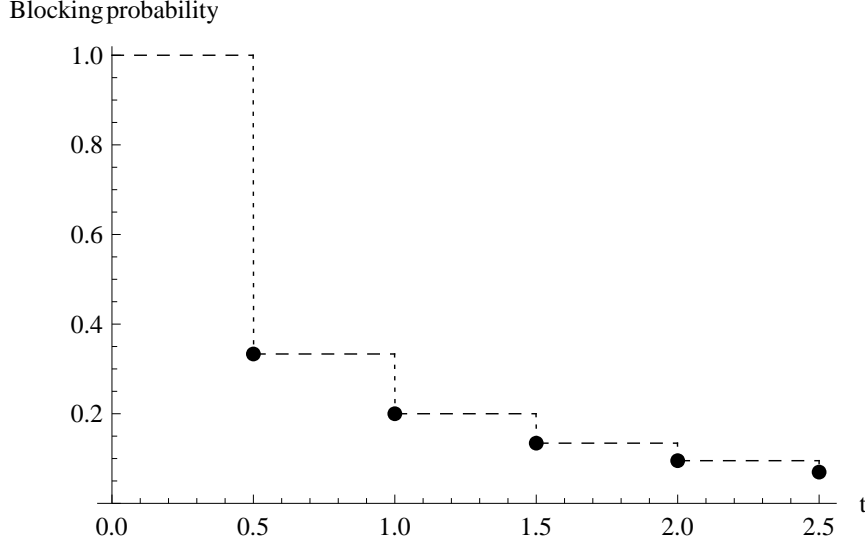
Figure 2: *The scaled functions $X^{(s,a)}(t)$ (dashed, discontinuous) and $B^{(s,a)}(t)$ (big dots) for $s = 2$ and $a = 1$ fixed.*

4.2 The new extension X and its properties

In this section, we define an extension that satisfies requirements (i) and (ii) in Lemma 4.1. More precisely, we introduce a new extension of the Erlang loss function that trivially satisfies upward scalability and subsequently show that it also adheres to requirement (ii). This new extension, $X : \mathbb{R}_+ \times \mathbb{R}_{++} \to [0,1]$, is defined by

$$X(s,a) = \begin{cases} B(\lfloor s \rfloor, a \cdot \lfloor s \rfloor / s) & \text{if } s \geq 1 \text{ and } a \in \mathbb{R}_{++}; \\ 1 & \text{if } s \in [0,1) \text{ and } a \in \mathbb{R}_{++}. \end{cases} \tag{10}$$

Figure 2 depicts (for $s = 2$ and $a = 1$ fixed) the scaled function $X^{(s,a)} : \mathbb{R}_{++} \to [0,1]$ which, by Definition 2.1, is defined by $X^{(s,a)}(t) = X(ts, ta)$. One can infer from Figure 2 that X is indeed an upward scalable extension of the Erlang loss function. Notice that this scaled function is stepwise constant and non-increasing. More precisely, for t between two successive values $t^- \in \{\frac{1}{s}, \frac{2}{s}, \ldots\}$ and $t^+ = t^- + \frac{1}{s}$, $X^{(s,a)}(t)$ equals $B(st^-, at^-)$. As such, we have the following lemma.

Lemma 4.2. *The function X is an extension and upward scalable.*

Proof. The extension part holds by definition. To show upward scalability, let $(s,a) \in \mathbb{R}_+ \times \mathbb{R}_{++}$ and consider the scaled function $X^{(s,a)}$. Let $t_1, t_2 \in \mathbb{R}_{++}$ with $t_1 \leq t_2$. If

15

$t_1 s \in [0, 1)$, then $X^{(s,a)}(t_1) = 1 \geq X^{(s,a)}(t_2)$. Otherwise, if $t_1 s \geq 1$, then

$$X^{(s,a)}(t_1) = B(\lfloor t_1 s \rfloor, a \cdot \lfloor t_1 s \rfloor / s)$$
$$\geq B(\lfloor t_2 s \rfloor, a \cdot \lfloor t_2 s \rfloor / s) = X^{(s,a)}(t_2),$$

where the inequality holds because the Erlang loss function B is upward scalable. We conclude that $X^{(s,a)}$ is a non-increasing function, which completes the proof. $\square$

It remains to show that X satisfies requirement (ii) in Lemma 4.1, i.e., $K_N^X(\frac{\lambda_N}{\lambda_M} S_M^*) \geq K_N^X(S_N^*)$ for all $M \in 2^N$. To this end, it is convenient to show the stronger result that $K_N^X(s) \geq K_N(S_N^*)$ for all $s \in \mathbb{R}_+$. We remark that this result is not obvious: although S_N^* is an optimal number of servers, it was obtained by optimizing over $\mathbb{N}_0$ only, and we now have to consider $K_N^X(s)$ for possibly non-integer s as well. In the process of proving this result, we first establish several structural properties of the new extension X and the linear interpolation L. We will present three lemmas that consider X and L as a function of the number of servers on a domain restricted to an interval between two consecutive integers. To describe this formally, it will be convenient to introduce two restricted functions: for any fixed $r = (\underline{S}, a) \in \mathbb{N} \times \mathbb{R}_{++}$, we define the functions L_r and X_r, both mapping $[\underline{S}, \underline{S}+1)$ to $[0, 1]$, by $L_r(s) = L(s, a)$ and $X_r(s) = X(s, a)$ for all $s \in [\underline{S}, \underline{S}+1)$. So, these functions are described on their domain by

$$X_r(s) = B(\underline{S}, a\underline{S}/s); \tag{11}$$
$$L_r(s) = (1 + \underline{S} - s) \cdot B(\underline{S}, a) + (s - \underline{S}) \cdot B(\underline{S}+1, a). \tag{12}$$

The following lemma states two properties of X that are graphically illustrated in Figure 3.

Lemma 4.3. *Let $r = (\underline{S}, a) \in \mathbb{N} \times \mathbb{R}_{++}$ be fixed. Then X_r is decreasing and convex on its domain $[\underline{S}, \underline{S}+1)$.*

Proof. By part *(ii)* in Theorem 2.1, it holds for each fixed $\hat{s} \in \mathbb{N}$ and $\lambda \in \mathbb{R}_{++}$ that $B(\hat{s}, \lambda/\mu)$ is decreasing and convex in μ for $\mu \in [\underline{S}, \underline{S}+1)$. By substituting $\hat{s} = \underline{S}$, $\lambda = a\underline{S}$, and $\mu = s$, we conclude that $X_r(s) = B(\underline{S}, a\underline{S}/s)$ is decreasing and convex in s on $[\underline{S}, \underline{S}+1)$. $\square$

In proving the next property of the function X, we make use of the following lemma.

Lemma 4.4. *Let $\underline{S} \in \mathbb{N}$ and $a \in \mathbb{R}_{++}$. Then,*

$$a[B(\underline{S}, a)]^2 - 1 - (a - \underline{S} - 1)B(\underline{S}, a) \leq 0. \tag{13}$$
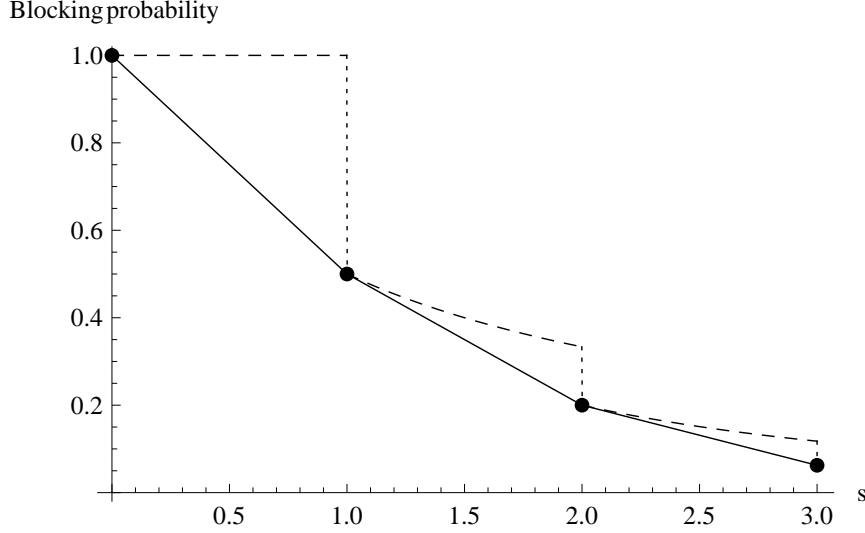
Figure 3: *The functions $L(s, a)$ (straight line segments), $X(s, a)$ (dashed, discontinuous), and $B(s, a)$ (big dots) for $a = 1$ fixed. The graph of $X_{(1,a)}$, for instance, corresponds to the dashed curve on $[1, 2)$.*

Proof. We start by rewriting inequality (13). By filling in the definition of the Erlang loss function (1) in inequality (13), subsequently combining the terms into a single fraction, and finally multiplying both sides with $-\left(\sum_{y=0}^{\underline{S}} a^y/y!\right)^2 < 0$, we obtain that inequality (13) is equivalent to

$$\left(\sum_{y=0}^{\underline{S}} \frac{a^y}{y!}\right)^2 - a \cdot \left(\frac{a^{\underline{S}}}{\underline{S}!}\right)^2 + (a - \underline{S} - 1) \cdot \left(\sum_{y=0}^{\underline{S}} \frac{a^y}{y!}\right) \cdot \frac{a^{\underline{S}}}{\underline{S}!} \geq 0. \tag{14}$$

Next, let the function $f : \mathbb{N} \to \mathbb{R}$ be defined by

$$f(s) = \left(\sum_{y=0}^{s} \frac{a^y}{y!}\right)^2 - \frac{a^{2s+1}}{s! \cdot s!} + \sum_{y=0}^{s} \frac{a^{y+s} \cdot (a - s - 1)}{y! \cdot s!}. \tag{15}$$

Notice that inequality (14) corresponds to $f(\underline{S}) \geq 0$. To complete the proof, we show that $f(s) \geq 0$ for all $s \in \mathbb{N}$ by induction. Firstly,

$$f(1) = (1 + a)^2 - a^3 + (a + a^2)(a - 2) = (1 + a)^2 - a^3 + a^2 - 2a + a^3 - 2a^2 = 1 \geq 0.$$

17

To avoid empty summations later on, it is convenient to treat the case $s = 2$ separately:

$$f(2) = \left(\frac{1}{2}a^2 + a + 1\right)^2 - \frac{1}{4}a^5 + \left(\frac{1}{4}a^4 + \frac{1}{2}a^3 + \frac{1}{2}a^2\right) \cdot (a - 3)$$

$$= \frac{1}{4}a^4 + a^3 + 2a^2 + 2a + 1 - \frac{1}{4}a^5 + \frac{1}{4}a^5 + \frac{1}{2}a^4 + \frac{1}{2}a^3 - \frac{3}{4}a^4 - \frac{3}{2}a^3 - \frac{3}{2}a^2$$

$$= \frac{1}{2}a^2 + 2a + 1 \geq 0.$$

For the induction step, let $s \in \{3, 4, \ldots\}$ and assume that $f(s-1) \geq 0$. Then,

$$f(s) = \left(\sum_{y=0}^{s} \frac{a^y}{y!}\right)^2 - \frac{a^{2s+1}}{s! \cdot s!} + \sum_{y=0}^{s} \frac{a^{y+s} \cdot (a - s - 1)}{y! \cdot s!}$$

$$= \left(\sum_{y=0}^{s} \frac{a^y}{y!}\right)^2 - \frac{a^{2s+1}}{s! \cdot s!} + \frac{a^{2s+1}}{s! \cdot s!} - \frac{a^{2s} \cdot s}{s! \cdot s!} - \frac{a^{2s}}{s! \cdot s!} + \sum_{y=0}^{s-1} \frac{a^{y+s} \cdot (a - s - 1)}{y! \cdot s!}$$

$$= \frac{a^{2s}}{s! \cdot s!} + 2 \cdot \sum_{y=0}^{s-1} \frac{a^{y+s}}{y! \cdot s!} + \left(\sum_{y=0}^{s-1} \frac{a^y}{y!}\right)^2 - \frac{a^{2s} \cdot s}{s! \cdot s!} - \frac{a^{2s}}{s! \cdot s!} + \sum_{y=0}^{s-1} \frac{a^{y+s} \cdot (a - s - 1)}{y! \cdot s!}$$

$$= \left(\sum_{y=0}^{s-1} \frac{a^y}{y!}\right)^2 - \frac{a^{2s} \cdot s}{s! \cdot s!} + \sum_{y=0}^{s-1} \frac{a^{y+s} \cdot (a - s + 1)}{y! \cdot s!}$$

$$= \left(\sum_{y=0}^{s-1} \frac{a^y}{y!}\right)^2 + \left[-\frac{a^s \cdot s}{s!} + \sum_{y=1}^{s} \frac{a^y}{(y-1)!} + \sum_{y=0}^{s-1} \frac{a^y \cdot (-s+1)}{y!}\right] \cdot \frac{a^s}{s!}$$

$$= \left(\sum_{y=0}^{s-1} \frac{a^y}{y!}\right)^2 + \left[\sum_{y=1}^{s-1} \frac{a^y}{(y-1)!} + \sum_{y=0}^{s-1} \frac{a^y \cdot (-s+1)}{y!}\right] \cdot \frac{a^s}{s!}$$

$$= \left(\sum_{y=0}^{s-1} \frac{a^y}{y!}\right)^2 + \left[\sum_{y=1}^{s-2} \frac{a^y}{(y-1)!} + \sum_{y=0}^{s-2} \frac{a^y \cdot (-s+1)}{y!}\right] \cdot \frac{a^s}{s!}$$

$$= \left(\sum_{y=0}^{s-1} \frac{a^y}{y!}\right)^2 + \left[\left(\sum_{y=1}^{s-2} \frac{a^y}{y!} \cdot (y - s + 1)\right) - s + 1\right] \cdot \frac{a^s}{s!}$$

$$\geq \left(\sum_{y=0}^{s-1} \frac{a^y}{y!}\right)^2 + \left[\sum_{y=1}^{s-2} \frac{a^y}{y!} \cdot \left(\frac{-s^2}{y+1} + s\right) - s + 1\right] \cdot \frac{a^s}{s!}$$

$$\geq \left(\sum_{y=0}^{s-1} \frac{a^y}{y!}\right)^2 + \left[\sum_{y=1}^{s-2} \frac{a^y}{y!} \cdot \left(\frac{-s^2}{y+1} + s\right) - s^2 + s - \frac{s^2}{a}\right] \cdot \frac{a^s}{s!}$$

$$= \left(\sum_{y=0}^{s-1} \frac{a^y}{y!}\right)^2 + \left[\sum_{y=0}^{s-2} \frac{a^y}{y!} \cdot \left(\frac{-s^2}{y+1} + s\right) - \frac{s^2}{a}\right] \cdot \frac{a^s}{s!}$$

18

$$= \left(\sum_{y=0}^{s-1} \frac{a^y}{y!} \right)^2 + \left[-\frac{a^{s-1} \cdot s}{(s-1)!} - \sum_{y=0}^{s-2} \frac{a^y \cdot s^2}{(y+1)!} - \frac{s^2}{a} + \sum_{y=0}^{s-2} \frac{a^{y-1} \cdot a \cdot s}{y!} + \frac{a^{s-1} \cdot s}{(s-1)!} \right] \cdot \frac{a^s}{s!}$$

$$= \left(\sum_{y=0}^{s-1} \frac{a^y}{y!} \right)^2 + \left[-\frac{a^{s-1} \cdot s}{(s-1)!} - \sum_{y=1}^{s-1} \frac{a^{y-1} \cdot s^2}{y!} - \frac{s^2}{a} + \sum_{y=0}^{s-1} \frac{a^{y-1} \cdot a \cdot s}{y!} \right] \cdot \frac{a^s}{s!}$$

$$= \left(\sum_{y=0}^{s-1} \frac{a^y}{y!} \right)^2 + \left[-\frac{a^{s-1} \cdot s}{(s-1)!} + \sum_{y=0}^{s-1} \frac{a^{y-1} \cdot (a-s) \cdot s}{y!} \right] \cdot \frac{a^s}{s!}$$

$$= \left(\sum_{y=0}^{s-1} \frac{a^y}{y!} \right)^2 - \frac{a^{2(s-1)+1}}{(s-1)! \cdot (s-1)!} + \sum_{y=0}^{s-1} \frac{a^{y+s-1} \cdot (a-(s-1)-1)}{y!(s-1)!}$$

$$= f(s-1) \geq 0.$$

In the first couple of steps, we split up summations, split off terms from summations, and cancel out common terms. The first inequality is valid because, for all $y \in \mathbb{N}_0$, it holds that $s^2 - 2s(y+1) + (y+1)^2 = (s-(y+1))^2 \geq 0$, and thus $y - s + 1 \geq -s^2/(y+1) + s$. The second inequality holds because $-s + 1 \geq -s^2 + s \geq -s^2 + s - s^2/a$, since $s > 1$. In the final steps, we rearrange our expression to show that it is equal to $f(s-1)$, which was non-negative by the induction hypothesis. Hence, by the principle of mathematical induction we have for all $s \in \mathbb{N}$ that $f(s) \geq 0$. This completes the proof. $\square$

Now, in contrast to X and L, the functions X_r and L_r are differentiable, which allows us to compare their derivatives, evaluated at $\underline{S}$, in the following lemma.

Lemma 4.5. *Let $r = (\underline{S}, a) \in \mathbb{N} \times \mathbb{R}_{++}$ be fixed. Then $L'_r(\underline{S}) \leq X'_r(\underline{S})$.*

Proof. First of all, the derivative of L_r for any $s \in [\underline{S}, \underline{S}+1)$ is

$$L'_r(s) = B(\underline{S}+1, a) - B(\underline{S}, a). \tag{16}$$

To obtain the derivative of X_r, we combine part (iii) of Theorem 2.1 with Equation (11) to derive that for any $s \in [\underline{S}, \underline{S}+1)$:

$$\begin{aligned} X'_r(s) &= [B(\underline{S}, a \cdot \underline{S}/s) - 1 + \underline{S}/(a \cdot \underline{S}/s)] \cdot B(\underline{S}, a \cdot \underline{S}/s) \cdot \left(-a\underline{S}/s^2\right) \\ &= [B(\underline{S}, a \cdot \underline{S}/s) - 1 + s/a] \cdot B(\underline{S}, a \cdot \underline{S}/s) \cdot \left(-a\underline{S}/s^2\right). \end{aligned} \tag{17}$$

Evaluating the derivatives (16) and (17) at $s = \underline{S}$, we obtain

$$L_r'(\underline{S}) - X_r'(\underline{S}) = B(\underline{S}+1, a) - B(\underline{S}, a) - \left[B(\underline{S}, a) - 1 + \frac{\underline{S}}{a}\right] \cdot B(\underline{S}, a) \cdot \frac{-a}{\underline{S}}$$

$$= \frac{aB(\underline{S}, a)}{aB(\underline{S}, a) + \underline{S} + 1} - B(\underline{S}, a) - \left[B(\underline{S}, a) - 1 + \frac{\underline{S}}{a}\right] \cdot B(\underline{S}, a) \cdot \frac{-a}{\underline{S}}$$

$$= \frac{aB(\underline{S}, a)}{aB(\underline{S}, a) + \underline{S} + 1} + B(\underline{S}, a) \cdot \left[-1 + B(\underline{S}, a) \cdot \frac{a}{\underline{S}} - \frac{a}{\underline{S}} + 1\right]$$

$$= B(\underline{S}, a) \cdot \left[\frac{a}{aB(\underline{S}, a) + \underline{S} + 1} + (B(\underline{S}, a) - 1) \cdot \frac{a}{\underline{S}}\right]$$

$$= B(\underline{S}, a) \cdot \frac{a}{\underline{S}(aB(\underline{S}, a) + \underline{S} + 1)} \cdot \left[\underline{S} + (B(\underline{S}, a) - 1) \cdot (aB(\underline{S}, a) + \underline{S} + 1)\right]$$

$$= B(\underline{S}, a) \cdot \frac{a}{\underline{S}(aB(\underline{S}, a) + \underline{S} + 1)} \cdot \left[a[B(\underline{S}, a)]^2 - 1 - (a - \underline{S} - 1)B(\underline{S}, a)\right]$$

$$\leq 0.$$

The second equality holds by part (iv) in Theorem 2.1. The other equalities hold by rewriting. The inequality holds because $B(\underline{S}, a) > 0$, $a > 0$, $\underline{S}(aB(\underline{S}, a) + \underline{S} + 1) > 0$, and $a[B(\underline{S}, a)]^2 - 1 - (a - \underline{S} - 1)B(\underline{S}, a) \leq 0$, where the last-named inequality holds by Lemma 4.4. We conclude that $L_r'(\underline{S}) \leq X_r'(\underline{S})$. $\qquad\square$

The following lemma states that the graph of X always lies above the graph of L, as illustrated in Figure 3.

Lemma 4.6. $X(s, a) \geq L(s, a)$ for all $s \in \mathbb{R}_+$ and $a \in \mathbb{R}_{++}$.

Proof. Let $(s, a) \in \mathbb{R}_+ \times \mathbb{R}_{++}$. We distinguish two cases.

Case 1: $s < 1$. Then, by definition $X(s, a) = 1$, whereas $L(s, a) \leq 1$.

Case 2: $s \geq 1$. Then, we denote $\underline{S} = \lfloor s \rfloor$ and consider the functions $X_{(\underline{S}, a)}$ and $L_{(\underline{S}, a)}$, which are described on their domain $[\underline{S}, \underline{S} + 1)$ by Equations (11) and (12). First of all, observe that $X_{(\underline{S}, a)}(\underline{S}) = L_{(\underline{S}, a)}(\underline{S})$ since both X and L are extensions of the Erlang loss function. Secondly, by Lemma 4.5, we observe that the derivative of $L_{(\underline{S}, a)}$ at $\underline{S}$ does not exceed the derivative of $X_{(\underline{S}, a)}$ at $\underline{S}$. Thirdly, $X_{(\underline{S}, a)}$ is convex by Lemma 4.3, whereas $L_{(\underline{S}, a)}$ is by definition a linear function, which (together with the second observation) implies that $X_{(\underline{S}, a)}' \geq L_{(\underline{S}, a)}'$ on $[\underline{S}, \underline{S} + 1)$. Combining these three observations yields $X_{(\underline{S}, a)}(s) \geq L_{(\underline{S}, a)}(s)$. We conclude that $X(s, a) \geq L(s, a)$. $\qquad\square$

As a result of Lemma 4.6, we have the following lemma.

Lemma 4.7. Let $\varphi = (N, \lambda, \tau, h, p)$ be a linear resource pooling situation. Then $K_N^X(s) \geq K_N(S_N^*)$ for all $s \in \mathbb{R}_+$.

Proof. Let $s \in \mathbb{R}_+$. The result follows upon comparison with the linear interpolation L:

$$K_N^X(s) \geq K_N^L(s) \geq \min_{\underline{S} \in \mathbb{R}_+} K_N^L(\underline{S}) = K_N(S_N^*),$$

where the first inequality holds by Lemma 4.6 and the first equality holds because K_N^L linearly interpolates between consecutive points of K_N. $\qquad\square$

Recall Lemma 4.1: existence of an extension of the Erlang loss function satisfying two properties is sufficient to conclude that the allocation proportional to arrival rates is stable in general. Lemma 4.2 showed that the extension X satisfies upward scalability – requirement (i) in Lemma 4.1. Lemma 4.7 immediately implies that $K_N^X(S_N^*) \geq K_N^X(\frac{\lambda_N}{\lambda_M} S_M^*)$ for all $M \in 2_-^N$ – requirement (ii) in Lemma 4.1. Thus, we can now state the following theorem, which is our first main result regarding resource pooling games.

Theorem 4.8. *Let $\varphi = (N, \lambda, \tau, h, p)$ be a linear resource pooling situation. Then, allocation $\mathcal{A}(\varphi)$ is stable, i.e., it is in the core of the associated resource pooling game (N, c^φ).*

5 Population monotonicity

In this section, we extend Theorem 4.8 by generalizing it to (possibly non-linear) resource pooling situations and by showing that $\mathcal{A}$, the cost allocation rule proportional to the arrival rates, can be reached via a population monotonic allocation scheme (PMAS). The idea behind the rule $\mathcal{A}$, which allocates the costs of the grand coalition proportional to player's arrival rates, can be naturally extended to an allocation scheme for each coalition. Accordingly, we define for any resource pooling situation $\varphi = (N, \lambda, \tau, H, p)$, coalition $M \in 2_-^N$, and player $i \in M$ this allocation scheme by

$$\bar{\mathcal{A}}_{i,M}(\varphi) = c^\varphi(M) \cdot \lambda_i / \lambda_M. \tag{18}$$

The following example illustrates this allocation scheme.

Example 5.1. Consider the resource pooling situation φ of Example 3.2 again. The corresponding allocation scheme $\bar{\mathcal{A}}(\varphi)$ is shown in Table 2. Notice that $\bar{\mathcal{A}}_{1,\{1,2\}}(\varphi) = \frac{3}{4} > \frac{63}{160} = \bar{\mathcal{A}}_{1,N}(\varphi)$ and similarly $\bar{\mathcal{A}}_{2,\{1,2\}}(\varphi) > \bar{\mathcal{A}}_{2,N}(\varphi)$, i.e., the amount that player 1 or 2 has to pay does not increase when player 3 joins them. This can be verified for all other coalitions as well, implying that $\bar{\mathcal{A}}(\varphi)$ is population monotonic. Finally, note that $\bar{\mathcal{A}}_{i,N}(\varphi) = \mathcal{A}_i(\varphi)$ for all players $i \in N$; thus, $(\bar{\mathcal{A}}_{i,N}(\varphi))_{i \in N}$ is a stable cost allocation. $\qquad\Diamond$

	Player 1	Player 2	Player 3
$\{1\}$	$1\frac{3}{22}$	*	*
$\{2\}$	*	$1\frac{3}{22}$	*
$\{3\}$	*	*	$3\frac{384}{827}$
$\{1,2\}$	$\frac{3}{4}$	$\frac{3}{4}$	*
$\{1,3\}$	$\frac{11893}{29118}$	*	$3\frac{3895}{14559}$
$\{2,3\}$	*	$\frac{11893}{29118}$	$3\frac{3895}{14559}$
N	$\frac{63}{160}$	$\frac{63}{160}$	$3\frac{24}{160}$

Table 2: *The values for the allocation scheme $\bar{\mathcal{A}}_{i,M}(\varphi)$ in Example 5.1.*

The following theorem states that resource pooling games always admit a PMAS. The proof of this theorem is similar to the proof of Theorem 4.8, based on the properties of the new extension X, while two new aspects are simultaneously tackled: possibly non-linear resource cost functions and population monotonicity.

Theorem 5.1. *Let $\varphi = (N, \lambda, \tau, H, p)$ be a resource pooling situation. Then $\bar{\mathcal{A}}(\varphi)$ is a PMAS of the associated resource pooling game (N, c^φ).*

Proof. We begin by extending the domain of the resource cost function H to all nonnegative reals by a linear interpolation, i.e., we define the function $H^{lin} : \mathbb{R}_+ \to \mathbb{R}_+$ by

$$H^{lin}(s) = (1 - (s - \lfloor s \rfloor)) \cdot H(\lfloor s \rfloor) + (s - \lfloor s \rfloor) \cdot H(\lceil s \rceil). \tag{19}$$

For any coalition $M \in 2^N_-$, we extend the domain of the cost function K_M (introduced in Equation (2)) in similar fashion, i.e., we define the function $K_M^{lin} : \mathbb{R}_+ \to \mathbb{R}_+$ by

$$\begin{aligned} K_M^{lin}(s) &= (1 - (s - \lfloor s \rfloor)) \cdot K_M(\lfloor s \rfloor) + (s - \lfloor s \rfloor) \cdot K_M(\lceil s \rceil) \\ &= H^{lin}(s) + L(s, \lambda_M \tau) \cdot \lambda_M p. \end{aligned} \tag{20}$$

We now fix two arbitrary coalitions $M, R \in 2^N_-$ with $M \subseteq R$, and we let $i \in M$. We show that $\bar{\mathcal{A}}(\varphi)$ is population monotonic as follows:

$$\begin{aligned} \bar{\mathcal{A}}_{i,R}(\varphi) &= \frac{\lambda_i}{\lambda_R} \cdot K_R^{lin}(S_R^*) \\ &\leq \frac{\lambda_i}{\lambda_R} \cdot K_R^{lin}\left(\frac{\lambda_R}{\lambda_M} S_M^*\right) \\ &= \frac{\lambda_i}{\lambda_R} \cdot H^{lin}\left(\frac{\lambda_R}{\lambda_M} S_M^*\right) + L\left(\frac{\lambda_R}{\lambda_M} S_M^*, \lambda_R \tau\right) \cdot \lambda_i p \end{aligned}$$

22

$$\leq \frac{\lambda_i}{\lambda_R} \cdot H^{lin}\left(\frac{\lambda_R}{\lambda_M} S_M^*\right) + X\left(\frac{\lambda_R}{\lambda_M} S_M^*, \lambda_R \tau\right) \cdot \lambda_i p$$

$$\leq \frac{\lambda_i}{\lambda_R} \cdot H^{lin}\left(\frac{\lambda_R}{\lambda_M} S_M^*\right) + X(S_M^*, \lambda_M \tau) \cdot \lambda_i p$$

$$= \frac{\lambda_i}{\lambda_M} \cdot \frac{\lambda_M}{\lambda_R} \cdot H^{lin}\left(\frac{\lambda_R}{\lambda_M} S_M^*\right) + B(S_M^*, \lambda_M \tau) \cdot \lambda_i p$$

$$\leq \frac{\lambda_i}{\lambda_M} \cdot H(S_M^*) + B(S_M^*, \lambda_M \tau) \cdot \lambda_i p$$

$$= \frac{\lambda_i}{\lambda_M} \cdot K_M(S_M^*) = \bar{\mathcal{A}}_{i,M}(\varphi).$$

The first inequality holds because S_R^* is a cost minimizing number of servers for coalition R, so both $K_R(\lfloor \frac{\lambda_R}{\lambda_M} S_M^* \rfloor)$ and $K_R(\lceil \frac{\lambda_R}{\lambda_M} S_M^* \rceil)$ are no smaller than $K_R(S_R^*)$, and the same holds for the associated linear interpolation. The second inequality holds by Lemma 4.6, and the third inequality holds because X is upward scalable (Lemma 4.2). The fourth inequality holds because H is a concave non-negative function, so the same holds for its linear interpolation H^{lin}, and thus

$$\frac{\lambda_M}{\lambda_R} \cdot H^{lin}\left(\frac{\lambda_R}{\lambda_M} S_M^*\right) \leq \frac{\lambda_M}{\lambda_R} \cdot H^{lin}\left(\frac{\lambda_R}{\lambda_M} S_M^*\right) + \frac{\lambda_R - \lambda_M}{\lambda_R} \cdot H^{lin}(0) \leq H^{lin}(S_M^*),$$

where the first inequality holds because $H^{lin}(0) \geq 0$ and the second inequality holds by concavity (since $S_M^* = \frac{\lambda_M}{\lambda_R} \cdot \frac{\lambda_R}{\lambda_M} S_M^* + \frac{\lambda_R - \lambda_M}{\lambda_R} \cdot 0$). We conclude that $\bar{\mathcal{A}}(\varphi)$ is a PMAS. $\square$

Finally, by Sprumont (1990), the following corollary follows immediately from Theorem 5.1.

Corollary 5.2. *Let $\varphi = (N, \lambda, \tau, H, p)$ be a resource pooling situation. Then $(\bar{\mathcal{A}}_{i,N}(\varphi))_{i \in N}$ is in the core of the associated resource pooling game (N, c^φ). Moreover, this game is totally balanced.*

6 Axiomatic characterization

In this section, we show that there is only one allocation rule on resource pooling situations that is immune to manipulations of the players via artificial splitting or merging, while simultaneously satisfying a weak continuity requirement: the proportional rule $\mathcal{A}$. This non-manipulability property means that no group of players will have an incentive to artificially represent themselves together as a single player, or vice versa. The following two definitions describe this concept more formally.

Definition 6.1. Two resource pooling situations $\varphi = (N, \lambda, \tau, H, p)$ and $\bar{\varphi} = (\bar{N}, \bar{\lambda}, \tau, H, p)$ are said to be *manipulations of each other* if there exist $M_{sp} \subseteq N$, $i_{me} \in \bar{N}$, and $G \subseteq N$ such that

- $N = G \cup M_{sp}$ and $\bar{N} = G \cup \{i_{me}\}$;

- $G \cap M_{sp} = \emptyset$, $G \cap \{i_{me}\} = \emptyset$, and $M_{sp} \cap \{i_{me}\} = \emptyset$; and

- $\sum_{j \in M_{sp}} \lambda_j = \bar{\lambda}_{i_{me}}$, in addition to $\lambda_i = \bar{\lambda}_i$ for all $i \in G$.

Then, M_{sp} is called *split-set*, i_{me} is called *merge-player*, and G is called *common set*. ◇

Definition 6.2. A rule f is called *non-manipulable* if, for any two resource pooling situations φ and $\bar{\varphi}$ that are manipulations of each other with split-set M_{sp} and merge-player i_{me}, it holds that $f_{i_{me}}(\bar{\varphi}) = \sum_{j \in M_{sp}} f_j(\varphi)$. ◇

Non-manipulability is very similar to immunity to coalitional manipulation (considered by Mosquera et al., 2008, for inventory situations) and non-advantageous merging or splitting (considered by García-Sanz et al., 2008, for queueing situations). In those concepts, any situation with player set N and coalition $M \subset N$ is associated with a fixed M-manipulation with player set $\{i_M\} \cup (N \setminus M)$, whereas we have defined non-manipulability as a relation between two situations. Our definition circumvents the potential problem where a player with name i_M might already be contained in set N.

Seemingly reasonable rules need not satisfy non-manipulability. For instance, consider the rule Q that assigns costs proportional to the square root of the arrival rates, which is defined for any resource pooling situation $\varphi = (N, \lambda, \tau, H, p)$ by

$$Q_i(\varphi) = c^\varphi(N) \cdot \frac{\sqrt{\lambda_i}}{\sum_{j \in N} \sqrt{\lambda_j}} \tag{21}$$

for each player $i \in N$. This rule assigns less costs per arrival to players with relatively high arrival rates, which may appear fair since players with high arrival rates already have some economies-of-scale benefits by themselves. However, as shown in the following example, this rule need not be stable and, moreover, it gives an incentive for players to merge, implying that Q does not satisfy non-manipulability.

Example 6.1. Consider the resource pooling situation φ of Example 3.2 again. Rule Q assigns $Q_1(\varphi) \approx 0.82$, $Q_2(\varphi) \approx 0.82$, and $Q_3(\varphi) \approx 2.31$. Note that since $Q_1(\varphi) + Q_2(\varphi) > c^\varphi(\{1, 2\})$, $Q(\varphi)$ is not a stable allocation.

Now, take split-set $\{1, 3\}$, merge-player i_{me} and common set $\{2\}$ to obtain the manipulation $\bar{\varphi} = (\bar{N}, \bar{\lambda}, \tau, H, p)$ with $\bar{N} = \{i_{me}, 2\}$, $\bar{\lambda}_{i_{me}} = \frac{9}{10}$, and $\bar{\lambda}_2 = \frac{1}{10}$. Then, $Q_{i_{me}}(\bar{\varphi}) = 2\frac{61}{64}$

and $Q_2(\bar\varphi) = \frac{63}{64}$. Clearly, $Q_{i_{me}}(\bar\varphi) \neq Q_1(\varphi) + Q_3(\varphi)$, and thus Q does not satisfy non-manipulability. $\diamond$

The following lemma states that the costs assigned to any player by a non-manipulable rule can only depend on the vector of arrival rates through that player's own arrival rate and an aggregate parameter. The lemma also states that these costs are independent of the total number and names of participating players. To state this lemma, we let $\mathbb{H}$ denote the set of all resource cost functions (i.e., the set of concave increasing unbounded functions mapping $\mathbb{N}_0$ to $\mathbb{R}_+$).

Lemma 6.1. *Let f be a non-manipulable rule. Then there exists an associated function $g : \mathbb{R}_{++}^3 \times \mathbb{H} \times \mathbb{R}_{++} \to \mathbb{R}$ such that, for all resource pooling situations $\varphi = (N, \lambda, \tau, H, p)$, $f_j(\varphi) = g(\lambda_j, \lambda_N, \tau, H, p)$ for all players $j \in N$.*

Proof. Let $\varphi = (N, \lambda, \tau, H, p)$ and $\hat\varphi = (\hat N, \hat\lambda, \tau, H, p)$ be two resource pooling situations such that player j is both in N and in $\hat N$, $\lambda_j = \hat\lambda_j$, and $\lambda_N = \hat\lambda_{\hat N}$. We now aim to show that $f_j(\varphi) = f_j(\hat\varphi)$, and we distinguish two cases.

Case 1: $N = \{j\}$. Then, $\hat N = \{j\}$ as well, so clearly $\varphi = \hat\varphi$, and thus $f_j(\varphi) = f_j(\hat\varphi)$.

Case 2: $|N| \geq 2$. Then, let $\bar\varphi = (\bar N, \bar\lambda, \tau, H, p)$ be the manipulation of φ via split-set $M_{sp} = N \setminus \{j\}$, merge-player i_{me}, and common set $\{j\}$. Notice that, by the combination of assumptions on φ and $\hat\varphi$, situation $\bar\varphi$ is a manipulation of $\hat\varphi$ as well via split-set $\hat M_{sp} = \hat N \setminus \{j\}$, merge-player i_{me}, and common set $\{j\}$. Now, $c^\varphi(N) = c^{\hat\varphi}(\hat N) = c^{\bar\varphi}(\bar N)$ because the grand coalition in all three situations faces the exact same problem. Then,

$$f_j(\varphi) = c^\varphi(N) - \sum_{i \in M_{sp}} f_i(\varphi) = c^{\bar\varphi}(\bar N) - \sum_{i \in M_{sp}} f_i(\varphi)$$
$$= c^{\bar\varphi}(\bar N) - f_{i_{me}}(\bar\varphi) = f_j(\bar\varphi),$$

where the first and fourth equalities hold because f is an allocation rule (thus fully assigning the costs of the grand coalition over all players), and the third equality holds because f satisfies non-manipulability. Analogously, we can also show $f_j(\hat\varphi) = f_j(\bar\varphi)$, and thus $f_j(\varphi) = f_j(\hat\varphi)$.

Hence, as φ and $\hat\varphi$ were chosen arbitrarily, the value $f_j(\varphi)$ can only depend on the vector λ through λ_j and λ_N and does not depend on the size of N.

To complete the proof, we show that $f_j(\varphi)$ does not depend on the name of player j either. Let φ' be a resource pooling situation that is obtained from φ by merely relabeling player j to j' in the entire tuple. But then φ' is a manipulation of φ with split-set $\{j\}$, merge-player j', and common set $N \setminus \{j\}$, and thus $f_j(\varphi) = f_{j'}(\varphi')$. As φ' was chosen arbitrarily, $f_j(\varphi)$ indeed does not depend on the name of player j. $\square$

To uniquely characterize $\mathcal{A}$, we employ a natural continuity property in addition to non-manipulability.

Definition 6.3. A rule f is called *continuous in arrival rates* if, for any resource pooling situation $\varphi = (N, \lambda, \tau, H, p)$, it holds that $\lim_{\lambda^* \to \lambda} f(N, \lambda^*, \tau, H, p) = f(N, \lambda, \tau, H, p)$. $\diamondsuit$

Before showing that there is only one rule satisfying both non-manipulability and continuity in arrival rates (the proportional rule $\mathcal{A}$), we prove the following lemma.

Lemma 6.2. *The rule $\mathcal{A}$ is non-manipulable and continuous in arrival rates.*

Proof. First, take φ, $\bar{\varphi}$, M_{sp}, i_{me}, and G as in Definition 6.2. Then,

$$
\begin{aligned}
\mathcal{A}_{i_{me}}(\bar{\varphi}) &= c^{\bar{\varphi}}(\bar{N}) \cdot \bar{\lambda}_{i_{me}} / \bar{\lambda}_{\bar{N}} = c^{\bar{\varphi}}(\bar{N}) \cdot \sum_{i \in M_{sp}} \lambda_i / \bar{\lambda}_{\bar{N}} \\
&= c^{\varphi}(N) \cdot \sum_{i \in M_{sp}} \lambda_i / \lambda_N = \sum_{i \in M_{sp}} \mathcal{A}_i(\varphi),
\end{aligned}
$$

where the third equality holds because the grand coalition in both situations faces the same problem and $\bar{\lambda}_{\bar{N}} = \lambda_N$, which is equal to $\bar{\lambda}_N$. Hence, $\mathcal{A}$ satisfies non-manipulability.

To establish continuity, simply observe that λ_i, λ_N, and $B(S_N^*, \lambda_N \tau)$ are all continuous in λ. Thus, $\mathcal{A}(\varphi) = c^{\varphi}(N) \cdot \lambda_i / \lambda_N$, where $\lambda_N > 0$ as all players have strictly positive arrival rates in resource pooling situations, is obviously continuous in arrival rates as well. $\square$

We are now ready to state the main result of this section in the following theorem.

Theorem 6.3. *$\mathcal{A}$ is the unique allocation rule on resource pooling situations satisfying both non-manipulability and continuity in arrival rates.*

Proof. Let f be a non-manipulable rule that is continuous in arrival rates, which exists by Lemma 6.2, and take an associated function g as in Lemma 6.1. Let $\varphi = (N, \lambda, \tau, H, p)$ be a resource pooling situation. Keeping λ_N, τ, H, and p fixed, we define the function $\hat{g} : (0, \lambda_N] \to \mathbb{R}$ by $\hat{g}(\ell) = g(\ell, \lambda_N, \tau, H, p)$.

Now, we aim to show that $\hat{g}$ is linear. To this end, let $\ell_1, \ell_2 \in (0, \lambda_N]$ be such that $\ell_1 + \ell_2 \le \lambda_N$. Then, there exists a resource pooling situation $\varphi' = (N', \lambda', \tau, H, p)$ such that $N' = \{1, 2, \ldots, n\}$ for some integer $n \ge 3$, $\lambda'_{N'} = \lambda_N$, $\lambda'_1 = \ell_1$, and $\lambda'_2 = \ell_2$. Let $\bar{\varphi}' = (\bar{N}', \bar{\lambda}', \tau, h, p)$ be a manipulation of φ' via split-set $\{1, 2\}$, merge-player i_{me} and common set $\{3, \ldots, n\}$. Then,

$$
\begin{aligned}
\hat{g}(\ell_1 + \ell_2) &= \hat{g}(\lambda'_1 + \lambda'_2) = \hat{g}(\bar{\lambda}_{i_{me}}) = f_{i_{me}}(\bar{\varphi}') = f_1(\varphi') + f_2(\varphi') \\
&= \hat{g}(\lambda'_1) + \hat{g}(\lambda'_2) = \hat{g}(\ell_1) + \hat{g}(\ell_2).
\end{aligned}
$$

26

The second equality holds since $\bar{\varphi}'$ is a manipulation of φ', and hence $\lambda_1' + \lambda_2' = \bar{\lambda}_{i_{me}}$. The third and fifth equalities hold by Lemma 6.1, and the fourth equality holds because f is a non-manipulable allocation rule. Hence, $\hat{g}$ is an additive function. Although we would like to conclude that $\hat{g}$ is a linear function, we need to be careful at this point because there exist additive functions which are not linear (Hamel, 1905).

However, $\hat{g}$ is a continuous function because, for any $\ell \in (0, \lambda_N]$, there exists a resource pooling situation $\hat{\varphi} = (\hat{N}, \hat{\lambda}, \tau, H, p)$ such that $j \in N$, $\hat{\lambda}_j = \ell$, and $\hat{\lambda}_{\hat{N}} = \lambda_N$, while by Lemma 6.1 it holds that $\hat{g}(\lambda_j) = f_j(\varphi)$. Thus, continuity in arrival rates of f implies continuity of $\hat{g}$. As $\hat{g}$ is both continuous and additive, in line with Cauchy (1821, Chapter 5) we conclude that $\hat{g}$ is linear, i.e., $\hat{g}(\ell) = \ell \cdot \hat{C}$ for all $\ell \in (0, \lambda_N]$ and some constant $\hat{C} \in \mathbb{R}$.

To finish the proof, we return to our original situation φ. As $f_i(\varphi) = \hat{g}(\lambda_i)$ for all $i \in N$, it holds that $\sum_{i \in N} f_i(\varphi) = \lambda_N \cdot \hat{C}$. Since f is an allocation rule, $\sum_{i \in N} f_i(\varphi) = c^\varphi(N)$, so $\hat{C} = c^\varphi(N)/\lambda_N$. Hence, $f_i(\varphi) = c^\varphi(N) \cdot \lambda_i/\lambda_N = \mathcal{A}_i(\varphi)$ for all $i \in N$. $\qquad \square$

We finish this section with a discussion. First, we compare our axiomatic characterization to related literature. Mosquera et al. (2008) uniquely characterize a proportional allocation rule for so-called p-additive cost games via an immunity to coalitional manipulation property that is similar to our non-manipulability property, and García-Sanz et al. (2008) give such a characterization for M/M/1 queueing games. Although Theorem 6.3, and in particular the line of its proof, is somewhat similar to the (proof of) these results in Mosquera et al. (2008) and García-Sanz et al. (2008), we have extended and adjusted their work in various ways. First and foremost, resource pooling games do not fit in the class of p-additive cost games or M/M/1 queueing games, so Theorem 6.3 states a new result.

Furthermore, we do not have to assume nonnegativity of cost allocations or independence on (names in and size of) the player set, as is done in Mosquera et al. (2008) and García-Sanz et al. (2008). Instead of nonnegativity, we assume a continuity property. We view exclusion of negative cost allocations a priori as unnatural – in fact, for a related model, Anily and Haviv (2010) show that core allocations with negative entries can exist. Thus, we consider our continuity requirement to be more natural. Our proof of Theorem 6.3 shows that nonnegativity can be easily replaced by continuity in the proof methodology described in Mosquera et al. (2008) and García-Sanz et al. (2008). Finally, by defining manipulations as a relation between two resource pooling situations, as mentioned earlier, we have shown in our proof of Lemma 6.1 that independence on (names in and size of) the player set actually follows from non-manipulability.

Finally, we consider independence of our two axioms. The square root rule Q is clearly

continuous in arrival rates, but does not satisfy non-manipulability, as shown in Example 6.1. Describing a non-manipulable rule that is not continuous in arrival rates seems beyond the scope of this paper because that would require construction of a pathological additive function that is discontinuous everywhere, in line with Hamel (1905).

7 Conclusion

In this paper, we presented a cooperative game corresponding to a situation where several players pool their resources to serve the union of their individual customer arrival streams. We modeled the service system of any coalition as an Erlang loss system in which the number of joint servers is chosen to minimize the sum of server costs and penalty costs.

We identified a cost allocation rule with appealing properties. Firstly, as this rule simply divides joint costs proportional to the arrival rate of each player, it is easy to understand and would be easy to implement in practice. Secondly, the cost allocation is stable, i.e., it gives no subset of players an incentive to split off and form a separate pooling group. Moreover, it can be reached through a population monotonic allocation scheme under which the costs faced by any player do not increase when the coalition to which he belongs grows. Finally, our rule is immune to manipulations of the players via artificial splitting or merging. In proving these properties, we used analytical characteristics of a new extension of the Erlang loss function to non-integral numbers of servers.

Acknowledgements

References

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

A. Bogomolnaia, R. Holzman, and H. Moulin. Sharing the cost of a capacity network. *Mathematics of Operations Research*, 35(1):173–192, 2010.

P. Borm, H. Hamers, and R. Hendrickx. Operations research games: a survey. *TOP*, 9(2): 139–199, 2001.

A.L. Cauchy. *Cours d'analyse de l'Ecole Royale Polytechnique, Part 1: Analyse Algebrique.* Paris, 1821.

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

M.G. Fiestras-Janeiro, I. Garca-Jurado, A. Meca, and M.A. Mosquera. Cooperative game theory and inventory management. *European Journal of Operational Research*, In press, 2010.

A.A. Fredericks. Congestion in blocking systems - a simple approximation technique. *Bell System Technical Journal*, 59(6):805–827, 1980.

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. *Contributions to the Theory of Games*, 4:47–85, 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.

G. Hamel. Eine Basis aller Zahlen und die unstetigen Lösungen der Funktionalgleichung: $f(x + y) = f(x) + f(y)$. *Mathematische Annalen*, 60(3):459–462, 1905.

A. Harel. Convexity properties of the Erlang loss formula. *Operations Research*, 38(3): 499–505, 1990.

D.L. Jagerman. Some properties of the Erlang loss function. *Bell System Technical Journal*, 53(3):525–551, 1974.

A.A. Jagers and E.A. Van Doorn. On the continued Erlang loss function. *Operations Research Letters*, 5(1):43–46, 1986.

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.

R.P. McLean and W.W. Sharkey. An approach to the pricing of broadband telecommunications services. *Telecommunication Systems*, 2(1):159–184, 1993.

M.A. Mosquera, I. García-Jurado, and M.G. Fiestras-Janeiro. A note on coalitional manipulation and centralized inventory management. *Annals of Operations Research*, 158 (1):183–188, 2008.

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

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

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.

R.I. Wilkinson. Theories for toll traffic engineering in the USA. *Bell System Technical Journal*, 35(2):421–514, 1956.

H. Wong, D. Van Oudheusden, and D. Cattrysse. Cost allocation in spare parts inventory pooling. *Transportation Research Part E*, 43(4):370–386, 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.

Working Papers Beta 2009 - 2011

nr.	Year	Title	Author(s)
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

		stocks	
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	A. Busic, I.M.H. Vliegen, A. Scheller-Wolf

		sets of states	
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