MONOTONICITY OF OPTIMAL POLICIES IN A ZERO SUM GAME: A FLOW CONTROL MODEL

Eitan ALTMAN*
INRIA
Centre Sophia Antipolis
06565 Valbonne Cedex, France
altman@martingale.inria.fr, tel. 93 65 76 73

January 1993

Abstract

The purpose of this paper is to illustrate how value iteration can be used in a zero-sum game to obtain structural results on the optimal (equilibrium) value and policy. This is done through the following example. We consider the problem of dynamic flow control of arriving customers into a finite buffer. The service rate may depend on the state of the system, may change in time and is unknown to the controller. The goal of the controller is to design a policy that guarantees the best performance under the worst case service conditions. The cost is composed of a holding cost, a cost for rejecting customers and a cost that depends on the quality of the service. We consider both discounted and expected average cost. The problem is studied in the framework of zero-sum Markov games where the server, called player 1, is assumed to play against the flow controller, called player 2. Each player is assumed to have the information of all previous actions of both players as well as the current and past states of the system. We show that there exists an optimal policy for both players which is stationary (that does not depend on the time). A value iteration algorithm is used to obtain monotonicity properties of the optimal policies. For the case that only two actions are available to one of the players, we show that his optimal policy is of a threshold type, and optimal policies exist for both players that may need randomization in at most one state.

*The work of this author was supported by the Chateaubriand fellowship from the French embassy in Israel

Keywords: zero-sum stochastic games, value iteration, monotonicity of optimal policies, control of queueing networks, flow control.

1 Introduction

There are many known computational methods for solving stochastic games, see e.g. the survey paper of Raghavan and Filar [10] and references therein. An alternative method that could reduce computations would be to prove in some way that the optimal strategies for all players are restricted to a small class of policies that have some structural properties. If we are lucky, this class of policies may depend only on one parameter (e.g. some real number), and the calculation of the performance under these policies might be easily done. In that case the original dynamic game may be reduced to a simpler optimization problem over the parameter space.

Several methods have been used in different stochastic games to obtain structural results. Altman and Shimkin considered in [3] a non zero-sum game with an infinite number of players, to solve the problem of choosing between the use of an individual personal computer, and a central computer whose capacity is simultaneously shared between different users. Using coupling and sample-path methods, all Nash optimal policies were shown to be of threshold type. This then enabled the computation of an optimal threshold. McNamara et al. [8] obtained a threshold type equilibrium policy for a dynamic version of the Hawk-Dove game using dynamic programming arguments. Hsiao and Lazar [4] obtained threshold equilibrium policies for a decentralized flow control into a network using the product form of the network as well as Northon's equivalent. The threshold policy is then obtained through a Linear Program. Küenle [7] used dynamic programming and especially the value iteration to solve an inventory control problem under worst case demand conditions. He modeled the problem as a stochastic zero-sum game with full information and identified the structure of an optimal policy of the controller, known as the (s,S) policy. Other results on worst case control in queueing models with routing and scheduling have recently been obtained by Altman and Koole [2], using again tools from stochastic games. Their results include as special case the optimality of "join the shortest queue" for the routing problem, and the well known μc-type policy for the problem of scheduling of the server.

In the present paper we consider a system where one user controls dynamically the flow of arriving customers into a finite buffer. The presence of other users as well as congestion phenomena is modeled by allowing the service rate to depend on the state of the system, and to change in time in a way that is unknown to the controller. Our goal is to design a control strategy that guarantees the best performance under the worst case service conditions. We formulate this problem as a zero-sum stochastic game, where the server is player one and the flow controller is player two. Using the value iteration technique, we are able to establish monotonicity properties of the policies that are optimal for both players, i.e. we provide not only the structure of an optimal policy for the flow controller, but identify also the structure of a worse case service conditions. We show that both policies are monotone decreasing in the state; the flow controller decreases the input flow as the number of customers increases, and under the worst service conditions, the service quality decreases with the number of customers in the queue. If one of the players has only two available actions, then this monotonicity is shown to imply that his optimal policy is of a threshold type, with randomization for both players in at most one state. Related results for the case of an infinite buffer with only two actions to each player were obtained by Altman in [1]. The case of finite buffer seems more delicate since additional boundary conditions appear in the value iteration.

The structure of the paper is as follows: in Section 2 we describe the model. Then basic tools for solving the problem with discounted cost are described in Section 3, and the properties of the optimal policies and value are derived in Section 4. The results are extended to the expected average cost in Section 5. In Section 6 we restrict to the case where two actions are available to each player; we use the structural results from Section 5 to reduce the original dynamic game into a simpler static one.

2 The model

Considered is a discrete-time single-server queue with a buffer of size L. We assume that at most one customer may join the system in a time slot. This arrival (if any) is assumed to occur at the beginning of the time slot.

Let X_t denote the number of customers in the system at time t, $t = 0, 1, \ldots$; the state space is denoted by $\mathbf{X} = \{0, 1, \ldots, L\}$.

Let g_{max} be a real number satisfying $0 < g_{max} < 1$. At the beginning of each time slot, if the state is x then the flow control mechanism, called player 2, chooses in a finite set $\mathbf{G}_x \subset [0, g_{max}]$ an action, which is interpreted as the probability of having one arrival in this time slot. Therefore, if action g is chosen at time t then a customer will enter the system in $[t, t+1)$ with the probability g. We assume that when the buffer is full, no arrivals are possible, and thus $\mathbf{G}_L = \{0\}$. In all states other than L we assume that the available actions are the same, and we denote them by $\mathbf{G}_x = \mathbf{G}$. We further assume that $0 \in \mathbf{G}$.

Let b_{min} and b_{max} be two real numbers satisfying $0 < b_{min} \leq b_{max} < 1$. At the end of each slot, if the state at the beginning of the slot is x, a successful service of a customer occurs, with some probability $b \in \mathbf{B}$ where $\mathbf{B}$ is a finite subset of $[b_{min}, b_{max}]$. If the service fails the customer remains in the queue, and if it succeeds then the customer leaves the system. The value of b, which may represent the quality of service, may change in each time slot, and is not known to player 2. The objective of player 2, to be described below, is to find a best strategy under the worst case service conditions. We model the system as a zero-sum Markov game, where player 1 controls the service quality.

Actions b and g are assumed to be taken independently, based on the information on the current state as well as the information of all past states and actions of both players.

We assume that a customer that enters an empty system may leave the system (with the probability b) at the end of this same time slot.

The state X_t denotes the number of customers in the system at time t, $t \in \mathbb{N}$, and B_t and G_t denote the actions of players 1 and 2 respectively. Let $M(Y)$ be the set of probability measures on a set Y.

The transition law q is:

$$q(y \,|\, x; b; g) := \begin{cases} \bar{g}b, & \text{if } L \geq x \geq 1,\ y = x - 1; \\ gb + \bar{g}\bar{b}, & \text{if } L \geq x \geq 1,\ y = x; \\ g\bar{b}, & \text{if } L > x \geq 0,\ y = x + 1; \\ 1 - g\bar{b}, & \text{if } y = x = 0; \end{cases}$$

(for any number $\xi \in [0, 1]$, $\bar{\xi} := 1 - \xi$).

We define an immediate payoff

$$C(x,b,g) := c(x) + \theta(b) + \rho(g) \tag{2.1}$$

for all $x \in \mathbf{X}$, $b \in \mathbf{B}$ and $g \in \mathbf{G}$. $C(x,b,g)$ is the cost that player 2 pays to player 1 when the state is x, and the actions of the players are b and g. C generalizes a cost frequently encountered in the literature on flow control models. In (2.1) $c(x)$ is any real-valued *increasing convex* function on $\mathbf{X}$, θ is a real function on $\mathbf{B}$ and ρ is a real function on $\mathbf{G}$. It is natural to assume that θ is increasing in b and $\theta \geq 0$ whereas ρ is decreasing in g and $\rho \leq 0$. $c(x)$ can be interpreted as a holding cost per unit time, ρ as a reward related to the acceptance of incoming customer, and θ as a cost per quality of service.

Let U (V) be the class of policies of player 1 (player 2 resp.). A policy $u \in U$ $(v \in V)$ is a sequence $u = (u_0, u_1, ...)$ $(v = (v_0, v_1, ...)$ resp.) where u_n (resp. v_n) is a conditional probability on $\mathbf{B}$ (resp. $\mathbf{G}$) given the history of all states and actions of both players as well as the current state. Thus each player is assumed to have the information of all last actions of both players as well as the current and past states of the system. Both players know the action sets, the immediate cost C, the initial state and the transition probabilities q.

Let u be a policy of player 1 and v a policy of player 2. Let ξ be a fixed number in $[0,1)$. Define the discounted cost:

$$V_\xi(x,u,v) := E^{u,v}\left[\sum_{t=0}^{\infty} \xi^t\, C(X_t, B_t, G_t) \,|\, X_0 = x\right], \tag{2.2}$$

Define the following problem $(\mathbf{Q}_\xi)$: Find u, v that achieve

$$V_\xi(x) := \sup_{u\in U} \inf_{v \in V} V_\xi(x,u,v), \qquad \forall x \in \mathbf{X}. \tag{2.3}$$

We know ([10] Section 2) that there exists a pair of stationary policies (u^*, v^*) that achieves (2.3), and

$$\sup_{u\in U}\inf_{v\in V} V_\xi(x,u,v) = \inf_{v\in V}\sup_{u\in U} V_\xi(x,u,v) = \sup_{u\in U} V_\xi(x,u,v^*) = \inf_{v\in V} V_\xi(x,u^*,v) = V_\xi(x,u^*,v^*).$$

$V_\xi(x)$ is called the ξ-discounted *value* of the game, and the policies (u^*, v^*) are called optimal policies.

A pair of policies (u,v) are said to be stationary if they depend only on the current state. In that case, we use the notation $u = \{u_x, x \in \mathbf{X}\}$, $u_x \in M(\mathbf{B})$, where $u_x(b)$ is the probability of choosing b when in state x; similarly we use the notation $v = \{v_x, x \in \mathbf{X}\}$, $v_x \in M(\mathbf{G})$, where $v_x(g)$ is the probability of choosing g in state x.

For any $\beta \in M(\mathbf{B})$ we denote $b(\beta) := E_\beta[b] = \sum_{b\in\mathbf{B}} b \cdot \beta(b)$. Similarly, for any $\gamma \in M(\mathbf{G})$ we denote $g(\gamma) := E_\gamma[g] = \sum_{g\in\mathbf{G}} g \cdot \gamma(g)$.

3 Preliminary Results

Let $\mathcal{K}$ be the set of all real-valued functions on $\mathbf{X}$. Define the operator $R : \mathbf{X} \times \mathbf{B} \times \mathbf{G} \times \mathcal{K} \to \mathbb{R}$ as

$$R(x,b,g,f) := E\left[f(X_{t+1})|X_t = x, B_t = b, G_t = g\right], \qquad g \in \mathbf{G}_x$$

we get:

$$R(x,b,g,f) = \begin{cases} (1-g\bar{b})\, f(x) + g\bar{b} f(x+1) & x = 0 \\ \bar{g} b f(x-1) + (gb + \bar{g}\bar{b}) f(x) + g\bar{b} f(x+1) & 1 \le x \le L \end{cases} \tag{3.1}$$

(in the above equation we shall understand $0 \cdot f(L+1) := 0$). Let $R(x,f)$ denote the matrix whose entries are $R(x,b,g,f)$.

Define the operator $S : \mathbf{X} \times \mathbf{B} \times \mathbf{G} \times \mathcal{K} \to \mathbb{R}$ as

$$S(x,b,g,f) := C(x,b,g) + \xi R(x,b,g,f), \qquad g \in \mathbf{G}_x$$

and let $S(x,f)$ denote the matrix whose entries are $S(x,b,g,f)$.

For any x, and functions $D : \mathbf{B} \times \mathbf{G}_x \to \mathbb{R}$, $\beta \in M(\mathbf{B})$ and $\gamma \in M(\mathbf{G}_x)$ define

$$\beta D \gamma := \sum_{b\in\mathbf{B}} \sum_{g\in\mathbf{G}_x} \beta(b)\gamma(g) D(b,g)$$

The value of the "matrix game D" is defined as $val(D) := \sup_{\beta\in M(\mathbf{B})} \inf_{\gamma\in M(\mathbf{G}_x)} \beta D\gamma$. It is known to satisfy $val(D) = \inf_{\gamma\in M(\mathbf{G}_x)} \sup_{\beta\in M(\mathbf{B})} \beta D\gamma$, and there are measures $\beta^* \in M(\mathbf{B})$, $\gamma^* \in M(\mathbf{G}_x)$ such that

$$val(D) = \inf_{\gamma\in M(\mathbf{G}_x)} \beta^* D\gamma = \sup_{\beta\in M(\mathbf{B})} \beta D\gamma^* = \beta^* D \gamma^*$$

β^* and γ^* are said to be optimal for the matrix game D. We shall use the following properties of matrix games. Given some $\gamma \in M(\mathbf{G}_x)$, let $supp(\gamma)$ be the set of actions in the support of γ, i.e. actions that are chosen with positive probability by γ. Define similarly $supp(\beta)$.

Lemma 3.1 *(i) Let (γ^*, β^*) be optimal for a matrix game D. Then for any $g \in supp(\gamma^*)$ and any $b \in supp(\beta^*)$,*

$$\sum_{g \in \mathbf{G}} \gamma^*(g) D(b, g) = val(D) = \sum_{b \in \mathbf{B}} \beta^*(b) D(b, g).$$

(ii) Let (γ^*, β^*) and $(\hat{\gamma}^*, \hat{\beta}^*)$ be optimal solutions for a matrix game D. Then $(\hat{\gamma}^*, \beta^*)$ is also optimal.

Proof. (i) follows from [12] p. 36. (ii) is straight forward. ∎

For any $\beta \in M(\mathbf{B}), \gamma \in M(\mathbf{G}_x)$ we shall understand with some abuse of notation

$$R(x, \beta, \gamma, f) := \beta R(x, f) \gamma, \qquad S(x, \beta, \gamma, f) := \beta S(x, f) \gamma.$$

Recall the definitions of $b(\beta)$ and $g(\gamma)$. We extend $f : \mathbf{X} \to \mathbb{R}$ to $\mathbf{X} \cup \{-1\} \to \mathbb{R}$, and set $f(-1) = f(0)$. With these definitions we have for any $0 \le x \le L$ and stationary u and v,

$$\begin{aligned} R(x, u_x, v_x, f) &= u_x R(x, f) v_x & (3.2) \\ &= \bar{g}(v_x) b(u_x) f(x-1) + (g(v_x) b(u_x) + \bar{g}(v_x) \bar{b}(u_x)) f(x) + g(v_x) \bar{b}(u_x) f(x+1). \end{aligned}$$

Let $T_\xi : \mathcal{K} \to \mathcal{K}$ be the DP (Dynamic Programming) operator associated with $\mathbf{Q}_\xi$:

$$T_\xi f(x) := val\, S(x, f), \qquad x \in \mathbf{X}. \tag{3.3}$$

Let $(u(f), v(f))$ be stationary policies such that the probability measures $(u_x(f), v_x(f))$ are optimal for the matrix game $S(x, f)$ for all $x \in \mathbf{X}$. We shall use the following tools for solving $\mathbf{Q}_\xi$:

Proposition 3.1 *(i) V_ξ satisfies $V_\xi(x) = T_\xi V_\xi(x)$.*

(ii) Let (u^*, v^*) be stationary policies of player 1 and 2 respectively such that for each $x \in \mathbf{X}$, the probability measures u_x and v_x are optimal for the matrix game $S(x, V_\xi)$. Then (u^*, v^*) are optimal

for $\mathbf{Q}_\xi$.

(iii) For every $f \in \mathcal{K}$, $\lim_{n\to\infty} T_\xi^n f = V_\xi$.

Proof. See Shapely [11]. ∎

Remark 3.1 *If either* u_x^* *or* v_x^* *do not randomize in some state* x*, then neither of them needs randomization in that state. This follows from the fact that* u_x^* *and* v_x^* *are solutions for the matrix game* $S(x, V_\xi)$ *(we make use of [12] Theorem 1.16.3 p. 26).*

4 Monotonicity of the optimal policies

We begin by defining the monotonicity of policies. Let u, v be stationary. Denote $b_x^{sup}(u) :=$ the greatest b in the support of u_x, i.e. the greatest $b \in \mathbf{B}$ that is chosen by u with positive probability when in state x. Denote $b_x^{inf}(u) :=$ the smallest b in the support of u_x, and define similarly $g_x^{sup}(v)$ and $g_x^{inf}(v)$.

We say that a stationary policy $u \in U$ is **strongly monotone** if for any $x \le L$ and y with $y < x$, $b_y^{inf}(u) \ge b_x^{sup}(u)$. We say that a stationary policy $v \in V$ is **strongly monotone** if for any $x \le L$ and y with $y < x$, $g_y^{inf}(v) \ge g_x^{sup}(v)$.

The monotonicity of a policy u means that the service quality is nonincreasing (in a probabilistic sense) as the number of customers in the buffer becomes larger. The monotonicity of a policy v means that the input flow is nonincreasing as the number of customers in the buffer becomes larger.

The following is a straight forward consequence from the definition of strongly monotone policies.

Lemma 4.1 *If* u *is strongly monotone then it randomizes in at most* $|\mathbf{B}| - 1$ *states. If* v *is strongly monotone then it randomizes in at most* $|\mathbf{G}| - 1$ *states.*

We shall say that $f \in \mathcal{K}$ satisfies assumption:

WC (weakly convex) if for all $0 \leq x < L-1$,

$$f(x+2) - f(x+1) \geq f(x+1) - f(x). \tag{4.1}$$

SC(x) (strongly convex) if for x given,

$$f(x+2) - f(x+1) > f(x+1) - f(x). \tag{4.2}$$

MI if $f(x)$ is monotone increasing in x, i.e. for any $0 \leq x < L$,

$$f(x+1) \geq f(x) \tag{4.3}$$

Let U^* be the set of stationary policies for the service controller such that $u \in U^*$ if and only if for any $x \in \mathbf{X}$ u_x is optimal for player 1 in the matrix game $S(x, V_\xi)$. Let V^* be the set of stationary policies for the flow controller such that $v \in V^*$ if and only if for any $x \in \mathbf{X}$ v_x is optimal for player 2 in the matrix game $S(x, V_\xi)$. It follows from Proposition 3.1 (ii) and Lemma 3.1 (ii) that any pair (u, v) such that $u \in U^*$ and $v \in V^*$ is optimal for problem $\mathbf{Q}_\xi$.

We are ready to present the main result.

Theorem 4.1 *If the holding cost c satisfies* **MI**, **WC** *and either $c(1) > c(0)$ or* **SC(0)** *then any of the optimal policies $u \in U^*$ and $v \in V^*$ are strongly monotone.*

In order to prove Theorem 4.1 we need first to introduce the following two technical lemmas.

Lemma 4.2 *Let $h : \mathbf{X} \cup \{-1\} \to \mathbb{R}$ be a nondecreasing function. Let $\zeta_1, \zeta_2 \in [0,1]$. Then, for all $0 \leq x < L$,*

$$F(x) := \zeta_2 h(x+1) + \bar{\zeta}_2 h(x) - \zeta_1 h(x) - \bar{\zeta}_1 h(x-1) \geq 0 \tag{4.4}$$

Moreover, if (i) $h(x+1) > h(x)$ and $\zeta_2 \neq 0$, or (ii) if $h(x) > h(x-1)$ and $\zeta_1 \neq 1$, then $F(x) > 0$.

Proof.

$$F(x) \geq h(x) - \zeta_1 h(x) - \bar{\zeta}_1 h(x-1) = \bar{\zeta}_1 [h(x) - h(x-1)] \geq 0,$$

and the second claim follows similarly. ∎

Lemma 4.3 *Assume that the holding cost c satisfies* **WC** *and* **MI**.

(i) Assume that f satisfies **WC** *and* **MI**. *Then $T_\xi f$ satisfies* **WC** *and* **MI**.

(ii) The value function V_ξ satisfies **WC** *and* **MI**.

(iii) If V_ξ satsifies **SC(x)** *in one state $x < L-1$, then it satisfies* **SC(y)** *for all $y \geq x$. If $V_\xi(1) - V_\xi(0) > 0$ then V_ξ satisfies* **SC(y)** *for all states y, $0 \leq y < L-1$.*

Finally, assume that the holding cost c satisfies **WC**, **MI** *and either $c(1) > c(0)$ or* **SC(0)**. *Then*

(iv) V_ξ satisfies **SC(y)** *for all states y, $0 \leq y < L-1$.*

Proof. (i) Choose arbitrary $u \in U^*$ and $v \in V^*$. We begin by establishing **MI**. Recall the definitions of $u_x(f)$ and $v_x(f)$ before Proposition 3.1. Choose any $0 \leq x \leq L-1$; let $b := b(u_x(f))$ and $g := g(v_{x+1}(f))$ ($g(\gamma)$ and $b(\beta)$ were defined in the end of Section 2).

$$
\begin{aligned}
& T_\xi f(x+1) - T_\xi f(x) && (4.5)\\
&= valS(x+1, f) - valS(x, f)\\
&\geq S(x+1, u_x(f), v_{x+1}(f), f) - S(x, u_x(f), v_{x+1}(f), f)\\
&= c(x+1) - c(x)\\
&\quad + \xi \left\{ \bar{g}b[f(x) - f(x-1)] + (gb + \bar{g}\bar{b})[f(x+1) - f(x)] + g\bar{b}[f(x+2) - f(x+1)] \right\}\\
&\geq c(1) - c(0) \geq 0 && (4.6)
\end{aligned}
$$

(The equation above holds indeed for $x = L-1$ too since in that case $g = 0$; in that case, we shall understand $gf(x+2) := 0$).

Next we check **WC**. Choose any $0 \leq x \leq L-2$; let $b := b(u_{x+1}(f))$, $g_1 := g(v_x(f))$. and $g_2 := g(v_{x+2}(f))$. Denote

$$F(x) = valS(x+2, f) - valS(x+1, f) - [valS(x+1, f) - valS(x, f)],$$

We have

$$F(x) \quad \geq \quad S(x+2, u_{x+1}(f), v_{x+2}(f), f) - S(x+1, u_{x+1}(f), v_x(f), f) \qquad (4.7)$$

$$
\begin{aligned}
& -[S(x+1,u_{x+1}(f),v_{x+2}(f),f)-S(x,u_{x+1}(f),v_x(f),f)] \\
= \quad & c(x+2)-c(x+1)-c(x+1)+c(x)+ \\
& \xi\left\{b\left[g_2f(x+2)+\bar{g}_2f(x+1)-g_1f(x+1)-\bar{g}_1f(x)\right]\right. \\
& +\bar{b}\left[g_2f(x+3)+\bar{g}_2f(x+2)-g_1f(x+2)-\bar{g}_1f(x+1)\right] \\
& -b\left[g_2f(x+1)+\bar{g}_2f(x)-g_1f(x)-\bar{g}_1f(x-1)\right] \\
& \left.-\bar{b}\left[g_2f(x+2)+\bar{g}_2f(x+1)-g_1f(x+1)-\bar{g}_1f(x)\right]\right\} \\
\geq \quad & \xi\left\{b\left[g_2(f(x+2)-f(x+1))+\bar{g}_2(f(x+1)-f(x))\right.\right. \qquad (4.8) \\
& \left.\left.-\ g_1(f(x+1)-f(x))-\bar{g}_1(f(x)-f(x-1))\right]\right\} \\
& +\xi\left\{\bar{b}\left[g_2(f(x+3)-f(x+2))+\bar{g}_2(f(x+2)-f(x+1))\right.\right. \\
& \left.\left.-g_1(f(x+2)-f(x+1))-\bar{g}_1(f(x+1)-f(x))\right]\right\} \\
\geq 0 & \qquad (4.9)
\end{aligned}
$$

which follows by applying twice Lemma 4.2 with $\zeta_i = g_i$, once with $h(x) = f(x+1) - f(x)$ for the term in the first curly brackets, and once with $h(x) = f(x+2) - f(x-1)$ for the term in the second curly brackets. (The equation above holds indeed for $x = L-2$ too since in that case $g_2 = 0$; in that case, we shall understand $g_2 f(x+3) := 0$).

(ii) Choose $f(x) = 0, \forall x \in \mathbf{X}$. By repeated application of Lemma 4.3 (i), it follows that $T_\xi^n f$ satisfies **MI** and **WC** for $n = 1, 2, \ldots$; moreover, $\lim_{n\to\infty} T_\xi^n f$ satisfies **MI** and **WC**. Hence by Proposition 3.1 (iii), V_ξ satisfies **MI** and **WC**.

(iii) Suppose that V_ξ satsifies **SC(x-1)** for some fixed $0 < x < L-1$. By substituting V_ξ instead of f in (4.7) and applying again Lemma 4.2 (this time we apply the second part of the Lemma;

indeed condition (ii) there holds since g_1 cannot be equal to one, and $h(x) = V_\xi(x+1) - V_\xi(x)$ satisfies $h(x) > h(x-1)$ by the assumption). We thus get strict inequality in (4.9). Hence V_ξ satisfies **SC(x)** as well, and similarly we conclude that it satisfies **SC(y)** for any $y \geq x$.

To prove the second claim, we substitute again V_ξ instead of f in (4.7) and consider $x = 0$. Again we have the case of the strict inequality in Lemma 4.2 since $h(x) := V_\xi(x+1) - V_\xi(x)$ satisfies indeed $h(x) - h(x-1) = V_\xi(1) - V_\xi(0) > 0$ (recall that $h(0) := 0$ since $V_\xi(-1) := V_\xi(0)$). We thus get again strict inequality in (4.9). It follows that $V_\xi(0)$ satisfies **SC(0)**, and hence by the first claim, it satisfies **SC(y)** for all $0 \leq y < L-1$.

(iv) Fix $x = 0$. Assume $c(1) > c(0)$. It follows that (4.6) holds with strict inequality for any f satisfying **MI** and in particular for $f = V_\xi$. Hence V_ξ which, by Proposition 3.1 (i), is equal to $valS(x, V_\xi)$, satisfies

$$V_\xi(1) - V_\xi(0) \geq c(1) - c(0) > 0.$$

The proof is then established by applying the second part of (iii).

Next assume that c satisfies **SC(0)**. Substituting V_ξ into (4.7) and considering $x = 0$ we get $F(x) > 0$ since we have a strict inequality in (4.8). Hence V_ξ satisfies **SC(0)**. The proof is then established by applying the first part of (iii). ∎

Proof of Theorem 4.1: (i) Choose some $v \in V^*$. In order to establish the monotonicity of v, it suffices to show that for any $x < L$, $g_1 < g_2$ and any $u \in U^*$,

$$\Delta(u,x) := S(x, u_x, g_2, V_\xi) - S(x, u_x, g_1, V_\xi) - [S(x-1, u_{x-1}, g_2, V_\xi) - S(x-1, u_{x-1}, g_1, V_\xi)] \quad (4.10)$$

is positive. Indeed, we show that this implies that $g_x^{sup}(v) \leq g_{x-1}^{inf}(v)$. Suppose $\Delta(x)$ is positive but the latter does not hold, i.e. $g_x^{sup}(v) > g_{x-1}^{inf}(v)$. Set $g_1 = g_{x-1}^{inf}(v)$ and $g_2 = g_x^{sup}(v)$. It follows from Lemma 3.1 (i) that for any $u \in U^*$

$$valS(x, V_\xi) = S(x, u_x, g_2, V_\xi) \leq S(x, u_x, g_1, V_\xi).$$

Hence since $\Delta(x,u)$ is positive, we have by (4.10)

$$S(x-1, u_{x-1}, g_2, V_\xi) < S(x-1, u_{x-1}, g_1, V_\xi) = valS(x-1, V_\xi)$$

where the last equality follows from Lemma 3.1 (i). This however contradicts the definition of the value of the matrix game $S(x-1, V_\xi)$. Hence it is indeed sufficient to show that $\Delta(u,x) > 0$, $\forall u \in U^*$ in order to prove the monotonicity of v. Fix some $u \in U^*$.

$$\begin{aligned}
\Delta(x,u) &= \xi(g_2 - g_1)\Big(b(u_x)[V_\xi(x) - V_\xi(x-1)] + \bar{b}(u_x)[V_\xi(x+1) - V_\xi(x)] \\
&\qquad - \left(b(u_{x-1})[V_\xi(x-1) - V_\xi(x-2)] + \bar{b}(u_{x-1})[V_\xi(x) - V_\xi(x-1)]\right)\Big) \\
&> 0 \qquad (4.11)
\end{aligned}$$

where the last inequality follows from Lemma 4.2 with $\zeta_2 = \bar{b}(u_x)$, $\zeta_1 = \bar{b}(u_{x-1})$ and $h(x) = V_\xi(x) - V_\xi(x-1)$, and since, by Lemma 4.3 (iv), V_ξ satisfies $\mathbf{SC(x)}$ for all x. This establishes the monotonicity of v.

Choose some $u \in U^*$. From similar arguments as in the first part of the proof, it suffices in order to establish the monotonicity of u, to show that for any $x \le L$, $b_2 > b_1$, $v \in V^*$,

$$\hat{\Delta}(x,v) := S(x, b_2, v_x, V_\xi) - S(x, b_1, v_x, V_\xi) - [S(x-1, b_2, v_{x-1}, V_\xi) - S(x-1, b_1, v_{x-1}, V_\xi)] < 0 \qquad (4.12)$$

Indeed, we show that this implies that $b_x^{sup}(u) \le b_{x-1}^{inf}(u)$. Suppose $\hat{\Delta}(x,v)$ is negative but the latter does not hold, i.e. $b_x^{sup}(u) > b_{x-1}^{inf}(u)$. Set $b_1 = b_{x-1}^{inf}(u)$ and $b_2 = b_x^{sup}(u)$.

It follows from Lemma 3.1 (i) that for any $v \in V^*$ we have

$$valS(x, V_\xi) = S(x, b_2, v_x, V_\xi) \ge S(x, b_1, v_x, V_\xi).$$

Hence since $\hat{\Delta}(x)$ is negative, we have by (4.12)

$$S(x-1, b_2, v_{x-1}, V_\xi) > S(x-1, b_1, v_{x-1}, V_\xi) = valS(x-1, V_\xi)$$

where the last equality follows from Lemma 3.1 (i). This however contradicts the definition of the value of the matrix game $S(x-1, V_\xi)$. Hence it is indeed sufficient to show that $\hat{\Delta}(x) < 0$ in order to prove the monotonicity of u. Fix some $v \in V^*$.

$$\hat{\Delta}(x) = -\xi(b_2 - b_1)\Big(\bar{g}(v_x)[V_\xi(x) - V_\xi(x-1)] + g(v_x)[V_\xi(x+1) - V_\xi(x)]$$

$$
\begin{aligned}
&\qquad - \left(\bar{g}(v_{x-1})[V_\xi(x-1) - V_\xi(x-2)] + g(v_{x-1})[V_\xi(x) - V_\xi(x-1)]\right)\Big) \\
&< \quad 0
\end{aligned}
\tag{4.13}
$$

where the last inequality follows from Lemma 4.2 with $\zeta_2 = g(v_x)$, $\zeta_1 = g(v_{x-1})$ and $h(x) = V_\xi(x) - V_\xi(x-1)$, and since V_ξ satisfies $\mathbf{SC(x)}$ for all x. (Recall that $\mathbf{G}_L = \{0\}$. Hence in the above equation, when $x = L$, $g(v_x)[V_\xi(x+1) - V_\xi(x)]$ is understood to be zero). This establishes the monotonicity of u. ∎

Lemma 4.1, Theorem 4.1 and Remark 3.1 imply the following

Corollary 4.1 *If the holding cost c satisfies* **MI**, **WC** *and either $c(1) > c(0)$ or* $\mathbf{SC(0)}$ *then there exist optimal stationary u^* and v^* that require randomization in not more than* $\min(|\mathbf{B}|, |\mathbf{G}|) - 1$ *states.*

In the next Corollary, we specify two cases where one of the players (or both) have in fact a threshold type optimal policy. The proof is a direct application of Theorem the previous Corollary as well as Remark 3.1.

We consider the case where $\mathbf{B} = \{b_1, b_2\}$ (i.e. the server has only two possible actions); we then use the notation $u_x = (u_x(1), u_x(2))$. We show for this case that the server has an optimal stationary policy of a threshold type.

Similarly, we may consider the case where $\mathbf{G} = \{g_1, g_2\}$, i.e. the flow controller has two actions in all states excluding state L (where the only available action is 0), and $g_1 = 0$. We show that v^* is of threshold type. We use the notation $v_x = (v_x(1), v_x(2))$ for this case.

Corollary 4.2 *Assume that the holding cost c satisfies* **MI**, **WC** *and either $c(1) > c(0)$ or* $\mathbf{SC(0)}$.
(i) Assume that $\mathbf{B} = \{b_1, b_2\}$ where $b_1 < b_2$. Then there exists $m_u \in \mathbf{X}$ such that

$$
u_x^* = \begin{cases} (1,0) & \text{if } x > m_u \\ (q_u, \bar{q}_u) & \text{if } x = m_u \\ (0,1) & \text{if } x < m_u \end{cases}
\tag{4.14}
$$

where $q_u \in [0,1]$ is some constant. Moreover, v_x^* needs no randomizations in any state except for (perhaps) $x = m_u$.

(ii) Assume that $\mathbf{G} = \{0, g\}$ *where* $g > 0$. *Then there exists* $m_v \in \mathbf{X}$ *such that*

$$v_x^* = \begin{cases} (1,0) & \text{if } x > m_v \\ (q_v, \bar{q}_v) & \text{if } x = m_v \\ (0,1) & \text{if } x < m_v \end{cases} \tag{4.15}$$

where $q_v \in [0,1]$ *is some constant. Moreover,* u_x^* *needs no randomizations in any state except for (perhaps)* $x = m_v$.

Remark 4.1 *It follows from Corollary 4.2 that if* $\mathbf{B} = \{b_1, b_2\}$ *and* $\mathbf{G} = \{g_1, g_2\}$ *and if* $m_u \neq m_v$ *then no randomization is needed at any state by both* u^* *and* v^*. *If* $m_u = m_v$ *then randomization may be needed only at* $x = m_u = m_v$.

Corollary 4.3 *Assume that the holding cost c satisfies* **MI**, **WC** *and either* $c(1) > c(0)$ *or* $\mathbf{SC(0)}$. *Assume that* $\mathbf{G} = \{0, g\}$. *Let* u^*, v^* *be optimal policies where* v^* *is as in Corollary 4.2. Let* m_v *be the threshold used by* v^* *and assume that* $m_v < L - 2$. *Consider the problem with all parameters the same, except that the buffer size* $\hat{L}$ *satisfies* $m_v < \hat{L} < L$. *Then* $\hat{u}^*, \hat{v}^*$ *are optimal policies for the new problem, where* $\hat{u}_x^* := u_x^*$, $\hat{v}_x^* := v_x^*$ *for all* $0 \leq x \leq \hat{L}$.

Proof. We shall use "hat" to denote quantities that correspond to the buffer $\hat{L}$. Choose any $x,\ 0 \leq x \leq \hat{L}$. Since $m_v < \hat{L}$, it follows that for any stationary u, the policies $\hat{u}$ and $\hat{v}^*$ (that are the restriction of u and v^* to the states $\{0, ..., \hat{L}\}$) satisfy $V_\xi(x, u, v^*) = \hat{V}_\xi(x, \hat{u}, \hat{v}^*)$. Hence

$$V_\xi(x, u^*, v^*) = \sup_{u \in U} V_\xi(x, u, v^*) = \sup_{\hat{u} \in \hat{U}} \hat{V}_\xi(x, \hat{u}, \hat{v}^*) = \hat{V}_\xi(x, \hat{u}^*, \hat{v}^*).$$

Consider the class of policies (for the system L) denoted by V' where player two always chooses 0 at any $x \geq \hat{L}$. Then for any $v \in V'$, $V_\xi(x, u^*, v) = \hat{V}_\xi(x, \hat{u}^*, \hat{v})$. Since $v^* \in V'$, we have for $x \leq \hat{L}$

$$\hat{V}_\xi(x, \hat{u}^*, \hat{v}^*) = V_\xi(x, u^*, v^*) = \inf_{v \in V} V_\xi(x, u^*, v) = \inf_{v \in V'} V_\xi(x, u^*, v) = \inf_{\hat{v} \in \hat{V}} \hat{V}_\xi(x, \hat{u}^*, \hat{v}).$$

Hence $\hat{u}^*, \hat{v}^*$ are optimal when the buffer size is $\hat{L}$. ∎

5 The average cost

Define the expected average cost

$$W(x,u,v) := \overline{\lim_{s\to\infty}} \frac{1}{s} E^{u,v}\left[\sum_{t=0}^{s-1} C(X_t, B_t, G_t) \mid X_0 = x\right], \tag{5.1}$$

Define the problem $\overline{\mathbf{Q}}$: Find u, v that achieve

$$W(x) := \sup_{u\in U} \inf_{v\in V} W(x,u,v), \qquad \forall x \in \mathbf{X}. \tag{5.2}$$

Theorem 5.1 *(i) There exists a pair of stationary policies (u^*, v^*) that achieves (5.2) such that $W = W(x)$ does not depend on x, and*

$$\sup_{u\in U} \inf_{v\in V} W(x,u,v) = \inf_{v\in V} \sup_{u\in U} W(x,u,v) = \sup_{u\in U} W(x,u,v^*) = \inf_{v\in V} W(x,u^*,v) = W(x,u^*,v^*). \tag{5.3}$$

(ii) If the holding cost c satisfies **MI**, **WC** *and either $c(1) > c(0)$ or* $\mathbf{SC(0)}$ *then there exist stationary u^* and v^* satisfying (5.3) which are strongly monotone policies.*

In the above theorem, $W(x)$ is called the expected average *value* of the game and the policies (u^*, v^*) are said to be optimal policies.

Proof. (i) follows from [9] Theorem 2.2. The only condition that should be verified is that there exists some state x_0 such that under any policies u and v, the state process reaches eventually x_0. It is easily seen that this condition is indeed verified, with $x_0 = 0$.

(ii) Due to (i), we may restrict the search of u^* and v^* to stationary policies (where the limsup in (5.1) is achieved as a limit). The claim then follows from Theorem 4.1 and [5] Corollary 3.2. that shows that any limit of the discounted optimal policies converges to a policy which is expected average optimal, as the discount factor goes to one. It remains to check Assumption A1 or A2 of that Corollary. However, it is shown in [5] Theorem 3 that the following weaker conditions imply A1:

A5(i) there exists a set $K \subset \mathbf{X}$ and a finite number $\mathcal{B}$ such that under any pair of stationary policies

u and v and any initial state x, the mean first passage time to K is at most $\mathcal{B}$.

A5(ii) For each stationary u and v, the Markov chain has no two disjoint closed sets.

In our model, A5(ii) clearly holds, since from every state we can reach 0 under any stationary u and v. A5(i) is satisfied too by choosing $K = \{0\}$. To see that, define $T(x)$ to be the supremum over all policies in (U, V) of the expected hitting time of state zero starting from state x. Consider the transition probabilities

$$\tilde{q}(y \mid x; g; b) := \begin{cases} q(y \mid x; g; b) & \text{if } x \neq 0 \\ 1\{y = 0\} & \text{if } x = 0 \end{cases}$$

Then $T(x) = \sup_{u,v} \sum_{s=1}^{\infty} P_x^{u,v}(X_s \neq 0)$. It follows by Theorem 3.2.1 in Kallenberg [6] that there exist stationary (u,v) that achieve that sup. However, for any stationary (u,v), 0 is a recurrent state that is reachable from any other state, the expected hitting time of 0 under (u,v) is finite, and hence $T(x) < \infty$. ∎

If one of the player has only two actions, we get in particular the following.

Corollary 5.1 *Assume that the holding cost c satisfies* **MI**, **WC** *and either $c(1) > c(0)$ or* $\mathbf{SC(0)}$. *Assume that $\mathbf{B} = \{b_1, b_2\}$ (or $\mathbf{G} = \{g_1, g_2\}$). Then a stationary optimal policy u^* (or v^* respectively) exists which has the threshold structure described in Corollary 4.2.*

It is easily seen that Corollary 4.1 and 4.3 extend also to the expected average case.

Remark 5.1 *All the results of this Section hold also for the case that the expected average cost (5.1) is defined through a liminf instead of limsup.*

6 Calculating the optimal policies

In this Section we assume that both players have only two actions. In particular, the flow controller may either use action 0 or action $g = g_{max}$. The server may choose between b_1 and b_2, where $b_1 = b_{min}$ and $b_2 = b_{max}$. We show how to use the previous structural results in order to compute the optimal policies. We restrict to the expected average cost criterion and assume that the conditions of Corollary 5.1 hold. We know from Corollary 5.1 that there exist optimal threshold policies for both players (given in Corollary 4.2). Hence we may restrict our problem to searching for optimal stationary policies among all threshold ones. We thus calculate the cost under a pair u, v of threshold policies that are characterized by the parameters m_u, q_u and m_v, q_v respectively (see definitions in Corollary 4.2).

We first calculate the costs for the case that $q_u = q_v = 1$, and then obtain the general case. Clearly, we need only to consider the case that $m_v \leq L$. Note that the policy where the flow controller always chooses g can be identified with $m_v = L$. We note that for a given policy m_v of the flow controller, all the policies such that $m_u > m_v$ have the same cost. In particular, for $q_u = q_v = 1$, all the policies such that $m_u \geq m_v$ have the same cost.

The steady state probabilities

From standard balance arguments we have the following relations between the steady state probabilities $\pi(m_u, 1, m_v, 1)$ (i.e., the steady state probabilities when using $m_u, m_v, q_u = q_v = 1$).

$$
\begin{array}{rcll}
\pi_x(m_u,1,m_v,1) g\bar{b}_2 & = & \pi_{x+1}(m_u,1,m_v,1)\bar{g}b_2 & 0 \leq x < \min(m_u,m_v) - 1 \\
\pi_x(m_u,1,m_v,1) g\bar{b}_2 & = & \pi_{x+1}(m_u,1,m_v,1)\bar{g}b_1 & x = m_u - 1, m_u < m_v \\
\pi_x(m_u,1,m_v,1) g\bar{b}_1 & = & \pi_{x+1}(m_u,1,m_v,1)\bar{g}b_1 & m_u \leq x < m_v - 1 \\
\pi_x(m_u,1,m_v,1) g\bar{b}_1 & = & \pi_{x+1}(m_u,1,m_v,1) b_1 & x = m_v - 1, m_u < m_v \\
\pi_x(m_u,1,m_v,1) g\bar{b}_2 & = & \pi_{x+1}(m_u,1,m_v,1) b_1 & x = m_v - 1, m_u = m_v \\
\pi_x(m_u,1,m_v,1) g\bar{b}_2 & = & \pi_{x+1}(m_u,1,m_v,1) b_2 & x = m_v - 1, m_u > m_v
\end{array}
$$

Denote

$$
\begin{array}{rcl}
\zeta_1 & := & \dfrac{g\bar{b}_2}{\bar{g}b_2}, \\[2ex]
\zeta_2 & := & \dfrac{g\bar{b}_2}{\bar{g}b_1} \text{ if } m_u < m_v, \quad 1 \text{ otherwise;} \\[2ex]
\zeta_3 & := & \dfrac{g\bar{b}_1}{\bar{g}b_1} \text{ if } m_u < m_v - 1, \quad 1 \text{ otherwise;} \\[2ex]
\zeta_4 & := & \dfrac{g\bar{b}_1}{b_1} \text{ if } m_u < m_v, \ \dfrac{g\bar{b}_2}{b_1} \text{ if } m_u = m_v, \ \dfrac{g\bar{b}_2}{b_2} \text{ if } m_u > m_v.
\end{array}
$$

We get

$$\pi_x(m_u,1,m_v,1) = \begin{cases} \pi_0(m_u,1,m_v,1)\zeta_1^x & 0 \le x < \min(m_u,m_v) \\ \pi_0(m_u,1,m_v,1)\zeta_1^{\min(m_u,m_v)-1}\zeta_2\zeta_3^{x-m_u} & m_u \le x < m_v \\ \pi_0(m_u,1,m_v,1)\zeta_1^{\min(m_u,m_v)-1}\zeta_2\zeta_3^{m_v-m_u-1}\zeta_4 & x = m_v \\ 0 & \text{otherwise} \end{cases} \tag{6.1}$$

Since the steady state probabilities sum to one, (6.1) yields

$$\begin{aligned} \pi_0(m_u,1,m_v,1) &= \left[\sum_{x=0}^{\min(m_u,m_v)-1} \zeta_1^x + \zeta_1^{\min(m_u,m_v)-1}\zeta_2\left(\sum_{x=0}^{m_v-m_u-1}\zeta_3^x + \zeta_3^{m_v-m_u-1}\zeta_4\right)\right]^{-1} \\ &= \left[\frac{1-\zeta_1^{\min(m_u,m_v)}}{1-\zeta_1} + \zeta_1^{\min(m_u,m_v)-1}\zeta_2\left(\frac{1-\zeta_3^{m_v-m_u}}{1-\zeta_3} + \zeta_3^{m_v-m_u-1}\zeta_4\right)\right]^{-1} \end{aligned} \tag{6.2}$$

The steady state probabilities are now obtained by substituting (6.2) into (6.1).

The expected average costs

When both u and v are of threshold type, we denote (with some abuse of notation) $W(m_u,q_u,m_v,q_v) := W(x,u,v)$. The expected average cost is given by

$$\begin{aligned} & W(m_u,1,m_v,1) \\ &= \sum_{x=0}^{m_v} c(x)\pi_x(m_u,1,m_v,1) \\ &\quad +\theta(b_2)\sum_{x=0}^{m_u-1}\pi_x(m_u,1,m_v,1) + \theta(b_1)\sum_{x=m_u}^{m_v}\pi_x(m_u,1,m_v,1) \\ &\quad +\rho(g)(1-\pi_{m_v}(m_u,1,m_v,1)) + \rho(0)\pi_{m_v}(m_u,1,m_v,1) \end{aligned}$$

Next we calculate for any q_u, and $q_v = 1$. To do that, we consider the regeneration points as the times that $X_t = m_u$. We call a "cycle" the duration between two consecutive visits to that state. The expected cost is then given by the expected cost per cycle divided by the expected cycle duration. We observe that with probability q_u, (respectively $\bar{q}_u$) the expected cost per cycle is equal to the one obtained if the server uses the policy $(m_u,1)$, (respectively, $(m_u,0) = (m_u+1,1)$); moreover, with probability q_u, (respectively $\bar{q}_u$) the expected cycle duration is equal to the one obtained if the server uses the policy $(m_u, 1)$, (respectively, $(m_u, 0) = (m_u+1, 1)$). Finally, we note that the expected cycle durations are just the inverse of the steady state probabilities of visiting state m_u. This yields

$$W(m_u, q_u, m_v, 1) =$$

$$\frac{q_u W(m_u, 1, m_v, 1)\pi_{m_u}^{-1}(m_u, 1, m_v, 1) + \bar{q}_u W(m_u + 1, 1, m_v, 1)\pi_{m_u}^{-1}(m_u + 1, 1, m_v, 1)}{\pi_{m_u}^{-1}(m_u, q_u, m_v, 1)}$$

where

$$\pi_{m_u}^{-1}(m_u, q_u, m_v, 1) = q_u \pi_{m_u}^{-1}(m_u, 1, m_v, 1) + \bar{q}_u \pi_{m_u}^{-1}(m_u + 1, 1, m_v, 1)$$

Finally, we get by similar arguments the cost for any q_u and q_v. We consider the regeneration points as the times that $X_t = m_v$. We thus get

$$W(m_u, q_u, m_v, q_v) = \tag{6.3}$$

$$\frac{q_v W(m_u, q_u, m_v, 1)\pi_{m_v}^{-1}(m_u, q_u, m_v, 1) + \bar{q}_v W(m_u, q_u, m_v + 1, 1)\pi_{m_v}^{-1}(m_u + 1, q_u, m_v, 1)}{q_v \pi_{m_v}^{-1}(m_u, q_u, m_v, 1) + \bar{q}_v \pi_{m_v}^{-1}(m_u, q_u, m_v + 1, 1)}$$

Thus the original dynamic game reduces to the problem of

$$\max_{m_u, q_u} \min_{m_v, q_v} W(m_u, q_u, m_v, q_v) \tag{6.4}$$

where $W(m_u, q_u, m_v, q_v)$ is given in (6.3). It follows from Corollary 5.1 that in order to solve (6.4) it suffices to restrict to the cases where either $m_u = m_v$ or $q_u = q_v = 1$.

References

[1] E. Altman, "Flow control using the theory of zero-sum Markov games," *Proceedings of the 31st IEEE Conference on Decision and Control*, Tucson, Arizona, pp. 1632-1637, December 1992.

[2] E. Altman and G. Koole, "Stochastic Scheduling Games with Markov Decision Arrival Processes", to appear in *Journal Computers and Mathematics with Appl.*, 3rd special issue on Differential Games, 1993.

[3] E. Altman and N. Shimkin, “Individually Optimal Dynamic Routing in a Processor Sharing System: Stochastic Game Analysis”, EE Pub No. 849, August 1992. Submitted to *IEEE Trans. Automatic Control*, 1992.

[4] M. T. Hsiao and A. A. Lazar, “Optimal Decentralized Flow Control of Markovian queueing Networks with Multiple Controllers”, *CTR Technical Report*, CUCTR-TR-19, Columbia University, 1986.

[5] A. Federgruen, “On N-person stochastic Games with denumerable state space”, *Adv. Appl Prob.* **10**, pp. 452-471, 1978.

[6] L. C. M. Kallenberg, *Linear Programming and Finite Markovian Control Problems*, Math. Centre Tracts 148, Amsterdam, 1983.

[7] H.-U. Küenle, “On the optimality of (s,S)-strategies in a minimax inventory model with average cost criterion”, *Optimization* **22** No. 1, pp. 123-138, 1991.

[8] J. M. McNamara, S. Merad and E. J. Collins, “The Hawk-Dove game as an average-cost problem”, *Adv. Appl. Prob.* **23**, pp. 667-682, 1991.

[9] T. Parthasarathy and M. Stern, “Markov games – a survey”, *Differential Games and Control Theory II*, Roxin, Liu and Sternberg, 1977.

[10] T.E.S. Raghavan and J.A. Filar, “Algorithms for Stochastic Games - A survey”, *Zeitschrift für OR*, vol 35, pp. 437-472, 1991.

[11] L. S. Shapely, “Stochastic games”, *Proceeding of the National Academy of Sciences USA* **39**, pp. 1095-1100, 1953.

[12] N. N. Vorob'ev, *Game Theory*, Lectures for Economists and Systems Scientists, Springer-Verlag, 1977.