Operations Research Letters 43 (2015) 96–98

Contents lists available at ScienceDirect

Operations Research Letters

journal homepage: www.elsevier.com/locate/orl

A note on the optimal stopping parking problem

Nguyen L. Truong

Department of Industrial Engineering and Operations Research, University of California Berkeley, United States

ARTICLE INFO

Article history:
Received 28 November 2012
Received in revised form
6 December 2014
Accepted 8 December 2014
Available online 18 December 2014

Keywords:
Optimal stopping
Optimal parking problem
Linear programming

ABSTRACT

We consider the classic optimal parking problem in DeGroot (1970) [1] and Puterman (2005), but in which the driver has a general parking utility function. If the utility function is k-modal, we show that the driver's optimal policy involves at most k sequences of skipping and parking in the first available spot.

1. Introduction

The classic optimal parking problem as described in DeGroot [1] and Puterman [2] involves someone *driving down a long street seeking to find a parking spot as close as possible to a specified destination. Assume that*

- *one may park on one side of the street and travel in one direction only,*
- *one can only inspect one spot at a time,*
- *the probability that a spot is vacant is* $\alpha > 0$, *and*
- *the occupancy status of a spot is independent of all other spots.*

We make several modifications to this classical setting. In particular, in our model we assume that

- there are n parking slots along the street, numbered from 1 to n,
- the probability that spot i is vacant is $\alpha_i > 0$,
- the utility derived by parking in spot i is $U_i > 0$,
- the utility for not being able to park in any spot is 0.

As such, in our model, we seek to maximize the expected parking utility, as opposed to minimizing the expected loss in [2]. Our main result in this paper, Theorem 3.2, is a generalization of Tamaki [3, Theorem 1], and is also an extension of the well-known result that when the utility function is unimodal, a threshold policy is optimal.

E-mail address: truongln@gmail.com.

http://dx.doi.org/10.1016/j.orl.2014.12.006

2. Linear programming setup

Let V_i denote the optimal expected parking utility given the driver starts from spot i. Then the following dynamic programming formulation easily presents itself:

$$V_i = \max\{\alpha_i U_i + (1-\alpha_i)V_{i+1},\ V_{i+1}\}, \quad i = 1, 2, \dots, n$$
$$V_{n+1} = 0.$$

Here, $\alpha_i U_i + (1-\alpha_i)V_{i+1}$ denotes the expected utility for stopping to park at spot i, and V_{i+1} denotes the expected utility for continuing on. Our objective is to determine the optimal policy resulting from the above recursion. We shall do so by first presenting this problem as a linear program in appropriate form. It is well-known that the above can be posed as the following linear program:

$$\begin{array}{lll} \min & V_1 & \\ \text{s.t.} & V_i \geq \alpha_i U_i + (1-\alpha_i)V_{i+1}, & 1 \leq i \leq n \\ & V_i \geq V_{i+1}, & 1 \leq i \leq n \\ & V_{n+1} = 0. & \end{array}$$

Let us go one step further, and make the substitution $x_i = V_i - V_{i+1}$ to obtain the new, equivalent linear program:

$$\begin{array}{lll} \min & \displaystyle\sum_{i=1}^{n} x_i & \\ \text{s.t.} & \displaystyle\frac{1}{\alpha_i} x_i + \sum_{l=i+1}^{n} x_l \geq U_i, & 1 \leq i \leq n \\ & x_i \geq 0, & 1 \leq i \leq n. \end{array}$$

It is this specific form of linear program in which we will analyze our parking problem given general utilities U_i's.

3. k-modal utility function

Given a function $U(\cdot)$ over a discrete set of real values $\{z_1, z_2, \dots, z_n\}$, where $z_1 < \cdots < z_i < z_{i+1} < \cdots < z_n$, we say the interval $[z_i, z_j]$ is a local maximum when $U(z_{i-1}) < U(z_i) = \cdots = U(z_j) > U(z_{j+1})$. If $i = 1$ and $j \neq n$, then only the right inequality needs to be satisfied. If $i \neq 1$ and $j = n$, then only the left inequality needs to be satisfied. If $i = 1$ and $j = n$, then we say the interval $[z_1, z_n]$ is a local (interval) maximum. When $i = j$, our definition reduces to the usual notion of a local maximum. In our setting, $z_i = i$ for $1 \le i \le n$, so that $U(z_i) = U(i) \equiv U_i$.

Similarly, the interval $[z_k, z_l]$ is said to be a local minimum for $U(\cdot)$ when $U(z_{k-1}) > U(z_k) = \cdots = U(z_l) < U(z_{l+1})$. If $k = 1$ and $l \neq n$, then only the right inequality needs to be satisfied. If $k \neq 1$ and $l = n$, then only the left inequality needs to be satisfied. If $k = 1$ and $l = n$, then we say the interval $[z_1, z_n]$ is a local (interval) minimum, e.g. it is considered as both a maximum and minimum. When $k = l$, our definition reduces to the usual notion of a local minimum. We observe that the interval $[z_k, z_l]$ is a local minimum for $U(\cdot)$ if and only if it is a local maximum for the function $-U(\cdot)$.

For our purpose, we will assume that a function $U(\cdot)$ is k-modal if it has k local (interval) maxima. We shall show here that when $U(\cdot)$ is k-modal, the optimal policy involves at most k sequences of skipping some number of spots, and parking in the first available spot thereafter. The proof is by induction on k.

Lemma 3.1. *Let $k = 1$, i.e. the utility function is unimodal, then a threshold policy is optimal. In other words, the optimal solution has the form*

$$x_i^* = \begin{cases} 0, & i = 1, \dots, \pi_1 - 1 \\ +, & i = \pi_1, \dots, n \end{cases}$$

for some appropriate π_1. Here, $+$ denotes a non-negative number. This linear program optimal solution corresponds to the policy where we skip the first $\pi_1 - 1$ spots, and park as soon as we encounter an available spot afterward.

Proof. Let h_1 denote the left most point of the local interval maxima (if the interval consists of a single point, then let h_1 be this point).

Due to the structure of the linear program, the optimal solution can be solved recursively as follows:

$$x_n^* = \alpha_n U_n,$$

$$x_i^* = \max\left\{0,\ \alpha_i\left(U_i - \sum_{l=i+1}^{n} x_l^*\right)\right\}, \quad 1 \le i \le n-1.$$

Let $1 \le \pi_1 \le n$ denote the largest integer for which:

$$(1) \quad \alpha_{\pi_1}\left(U_{\pi_1} - \sum_{l=\pi_1+1}^{n} x_l^*\right) \ge 0, \quad \text{but}$$

$$(2) \quad \alpha_{\pi_1-1}\left(U_{\pi_1-1} - \sum_{l=\pi_1}^{n} x_l^*\right) < 0.$$

This clearly implies $x_{\pi_1-1}^* = 0$. From the definition of π_1, we can also deduce $U_{\pi_1-1} < U_{\pi_1}$, and thus $\pi_1 - 1 < h_1$.

Next, observe that

$$\left(U_{\pi_1-2} - \sum_{l=\pi_1-1}^{n} x_l^*\right) \le U_{\pi_1-1} - \sum_{l=\pi_1-1}^{n} x_l^* = U_{\pi_1-1} - \sum_{l=\pi_1}^{n} x_l^* < 0,$$

where the first inequality holds due to the observation that $\pi_1 - 1 < h_1$, and hence $U_{\pi_1-2} \le U_{\pi_1-1}$; the equality is correct due to the observation that $x_{\pi_1-1}^* = 0$; and the last inequality follows from the definition of $x_{\pi_1-1}^*$. Hence we can conclude $x_{\pi_1-2}^* = 0$.

Continuing in this way, we can deduce $x_i^* = 0$ for all $1 \le i \le \pi_1 - 1$.

To map the optimal linear programming solution to a parking policy, observe that $x_i^* = 0$ corresponds to $V_i = V_{i+1}$, i.e. skipping to the next parking slot. Whereas $x_i^* > 0$ corresponds to $V_i = \alpha_i U_i + (1 - \alpha_i)V_{i+1}$, i.e. stopping and parking if the spot is available. □

Theorem 3.2. *Let the utility function be k-modal, then the optimal solution has the form*

$$x_i^* = \begin{cases} 0, & i = 1, \dots, \pi_1 - 1 \\ +, & i = \pi_1, \dots, \sigma_1 - 1 \\ 0, & i = \sigma_1, \dots, \pi_2 - 1 \\ +, & i = \pi_2, \dots, \sigma_2 - 1 \\ \vdots & \\ 0, & i = \sigma_{k-1}, \dots, \pi_k - 1 \\ +, & i = \pi_k, \dots, n \end{cases}$$

for some appropriate $1 \le \pi_1 \le \sigma_1 \le \cdots \le \sigma_{k-1} \le \pi_k \le n$. Here, $+$ denotes a non-negative number. This linear program optimal solution corresponds to the policy where we skip spots in the interval $[1, \pi_1 - 1]$, park as soon as we encounter an available spot in the interval $[\pi_1, \sigma_1 - 1]$, skip spots in the interval $[\sigma_1, \pi_2 - 1]$, park as soon as we encounter an available spot in the interval $[\pi_2, \sigma_2 - 1]$, and so forth. As such, the optimal parking policy involves at most k sequences of skipping and parking in the first available spot.

Proof. The proof is by induction on k, the number of local (interval) maxima exhibits by the utility function $U(\cdot)$.

- $k = 1$: Then the utility function $U(\cdot)$ is unimodal, and the result holds due to Lemma 3.1.
- $k \le s - 1$: If the utility function is $(s-1)$-modal, assume that the optimal solution has at most $s - 1$ sequences of skipping and parking in the first available spot.
- $k = s$: Let h_i denote the left most point of the ith (counting from left to right) local interval maxima. If the interval consists of a single point, then let h_i be this point.
  Let l_i denote the left most point of the ith (counting from left to right, beginning at h_1) local interval minima. If the interval consists of a single point, then let l_i be this point.
  Here, we have that $1 \le h_1 < l_1 < \cdots < l_{s-1} < h_s \le n$. The existence of l_s is not guaranteed.
  Due to the structure of the linear program, observe that the optimal solution has the following recursive structure:

  $$x_n^* = \alpha_n U_n,$$

  $$x_i^* = \max\left\{0,\ \alpha_i\left(U_i - \sum_{j=i+1}^{n} x_j^*\right)\right\}.$$

  Let $1 \le \pi_s \le n$ denote the largest number for which

  $$(1') \quad \alpha_{\pi_s}\left(U_{\pi_s} - \sum_{j=\pi_s+1}^{n} x_j^*\right) \ge 0, \quad \text{but}$$

  $$(2') \quad \alpha_{\pi_s-1}\left(U_{\pi_s-1} - \sum_{j=\pi_s}^{n} x_j^*\right) < 0.$$

  If π_s does not exist, then we should stop and park in an available spot as soon as possible. When it does exist, there are two scenarios to consider.
  - $\pi_s < l_{s-1}$: then we know $x_i^* = +$ for $\pi_s \le i \le n$, and also $\pi_s \le h_{s-1}$. Consider the strictly smaller parking problem from 1 to π_s with a *modified* U_{π_s} equal to the expected utility for employing the optimal policy from π_s onward, and a modified $\alpha_{\pi_s} = 1$. This problem contains $s - 1$ local (interval) maxima,

and hence by the induction hypothesis, requires at most $s-1$ sequences of skipping and parking in the first available space. Concatenate this optimal solution for the smaller problem with our observed solution earlier, and we still only have at most $s-1$ sequences of skipping and parking. So that the theorem holds in this scenario.

– $\pi_s \geq l_{s-1}$: by a similar argument as that in Lemma 3.1, we must have $\pi_s - 1 < h_s$. Next, let $1 \leq \sigma_{s-1} \leq \pi_s - 1$ be the largest integer for which

$$(1'') \quad \alpha_{\sigma_{s-1}} \left(U_{\sigma_{s-1}} - \sum_{j=\sigma_{s-1}+1}^{n} x_j^* \right) < 0, \quad \text{but}$$

$$(2'') \quad \alpha_{\sigma_{s-1}-1} \left(U_{\sigma_{s-1}-1} - \sum_{j=\sigma_{s-1}}^{n} x_j^* \right) \geq 0.$$

If such a σ_{s-1} does not exist, then we are done since all x_i^* are 0 for $1 \leq i \leq \pi_s - 1$. That is, we will skip through all spots $1 \leq i \leq \pi_s - 1$, and park as soon as we find an available spot afterward. This optimal policy has only one sequence of skipping and parking in the first available spot.

If such a σ_{s-1} exists, then observe that for $l_{s-1} \leq i \leq \pi_s - 1 < h_s$, we must have $U(i) \leq U(i+1)$. This implies $\alpha_i \left(U_i - \sum_{l=i+1}^{n} x_l^* \right) < 0$ for i's in this range. As such, due to how we define σ_{s-1}, we deduce that $\sigma_{s-1} \leq l_{s-1}$. By using a similar argument as in the above case, and invoking the induction hypothesis, the optimal policy involves at most $s-1$ sequences of skipping and parking in the first available spot before reaching σ_{s-1}. From this point and beyond, we only require skipping in the interval $[\sigma_{s-1}, \pi_s - 1]$, and parking in the first available spot in the interval $[\pi_s, n]$. Thus, the optimal policy involves at most s sequences of skipping and parking. $\square$

We end this note by highlighting the intuitiveness of our result above: the optimal policy dictates the possibility of parking in neighborhoods of the driver's utility peaks, and ignoring all other spots close to the driver's utility valleys.

References

[1] M.H. DeGroot, Optimal Statistical Decisions, McGraw-Hill Book Company, New York, 1970, p. 384.

[2] M.L. Puterman, Markov Decision Processes, Wiley-InterScience, New York, 2005, pp. 319–323.

[3] Mitsushi Tamaki, Adaptive approach to some stopping problems, J. Appl. Probab. 22 (3) (1985) 644–652.