Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Letter to the Editor—A Proof of the Optimality of the Shortest Remaining Processing Time Discipline

Linus Schrage,

and its cost is 1. If we want to find $\boldsymbol{a}(3)$ then we have to partition M_4 by $\boldsymbol{a}(2)$ and continue the algorithm as before.

REFERENCES

1. H. W. KUHN, "The Hungarian Method for Solving the Assignment Problem," *Nav. Res. Log. Quart.* **2**, 83–97 (1955).
2. K. G. MURTY, CAROLINE KAREL, AND JOHN D. C. LITTLE, "The Traveling Salesman Problem: Solution by a Method of Ranking Assignments," Case Institute of Technology, (unpublished) (1962).
3. L. R. FORD, JR., AND D. R. FULKERSON, "Notes on Linear Programming—Part XXXII: Solving the Transportation Problem," Rand Corporation. RM-1736 (June 1956).
4. M. L. BALINSKI AND R. E. GOMORY, "A Primal Method for the Assignment and Transportation Problems," *Management Sci.* **10,** No. 3 (April 1964).

A PROOF OF THE OPTIMALITY OF THE SHORTEST REMAINING PROCESSING TIME DISCIPLINE

Linus Schrage

University of Chicago, Chicago, Illinois

(Received January 17, 1968)

IN A RECENT paper,[1] "The Queue M/G/1 with the Shortest Remaining Processing Time Discipline," it is stated:

> It can be proved inductively that with the SRPT discipline the number of jobs in system at any point in time is less than or equal to the number of jobs in system for any other rule simultaneously acting on the same sequence of arrivals and processing times. Note that the optimality of SRPT does not depend on any assumptions about the distribution of either the interarrival times or processing times.

That paper did not include a proof of the above statement; however, a number of people have expressed interest in the proof, so it is the purpose of this letter to give such a proof.

The Shortest Remaining Processing Time (SRPT) rule states that the processor should at all times process that job of those available, which has the shortest remaining processing time.

DEFINITIONS

AN ARRIVAL stream is a collection of 2-tuples $\{A(n), P(n)\}$, $n = 1, 2, \cdots$ where $A(n)$ is known as the arrival time of the nth job and $P(n)$ is known as the processing time of the nth job.

A queuing discipline is a rule for assigning subintervals of time to the nth job from the interval $(A(n), \infty)$, $n = 1, 2, \cdots$ subject to the restriction that a subinterval of time can be devoted to at most one job.

Define the following notation:

$$\delta(n, t) = \begin{cases} 1 \text{ if the processor is devoted to the } n\text{th job at time } t, \\ 0 \text{ for } t < A(n), \\ 0 \text{ otherwise.} \end{cases}$$

We have the constraint that the processor can be devoted to at most one job at a time, i.e.,

$$\textstyle\sum_{n=1}^{n=\infty} \delta(n, t) \leqq 1 \text{ for all } t.$$

Job n completes when the sum of all intervals of time devoted to n equals $P(n)$; thus, the completion time $C(n)$ is given by

$$C(n) = \min\left\{x: \int_{A(n)}^{x} \delta(n, t)\, dt \geqq P(n)\right\}.$$

The processing time remaining for job n at time t is denoted by $S(n, t)$ and is defined by:

$$S(n, t) = P(n) - \int_{A(n)}^{t} \delta(n, x)\, dx.$$

It follows that

$$C(n) = \min\left\{x: \int_{t}^{x} \delta(n, y)\, dy \geqq S(n, t)\right\}.$$

Denote the set of jobs in system at time t by $\theta(t)$.
Denote the number in system at time t by $N(t)$.

A specification of values for $\delta(n, t)$ for all n and t is referred to as a schedule.

Following the SRPT discipline implies that $[S(k, t) > S(j, t)$ for $j \epsilon \theta(t)$ implies that $\delta(k, t) = 0]$.

ASSUMPTIONS

THE FOLLOWING assumptions are important.

Assumption (I): The arrival stream is independent of the discipline used in processing the jobs. Assumption (II): The total amount of processing required over-all or by a job is independent of any interruptions of processing, i.e., a preemption implies no wasted processing or setup.

OUTLINE OF PROOF

THE PROOF is based on an interchange argument similar to those found in reference 2. The outline of the proof is as follows. Consider a schedule that does not follow the SRPT discipline over some nonzero interval of time. It will be shown that the schedule can be improved by practicing the SRPT discipline over that interval of time. Therefore, non-SRPT disciplines cannot be optimal. Therefore, the SRPT discipline is optimal by default.

PROOF

THE SUBSCRIPT 'o' will be used to identify measures related to the original, non-SRPT schedule and the 'r' subscript will be used to identify measures associated with the revised schedule. No subscript is used for a statement that is true for both schedules.

The original schedule is assumed to be non-SRPT, which implies that we can find an interval $(t, t+v), v>o$ such that

(a) there is some job j contained in $\theta_o(t)$ such that $[S_o(j, x)<S_o(k, x)$ for any $k\epsilon\theta_o(x)$, and $\delta_o(j, x)=0]$ for $t\leqq x\leqq t+v$,

(b) and either: ($b1$) $\delta_o(k, x)=0$ for all $k\epsilon\theta_o(x)$ for $t\leqq x\leqq t+v$, or ($b2$) there is a unique k such that $\delta_o(k, x)=1$ and $k\epsilon\theta_o(x)$ for $t\leqq x\leqq t+v$.

If the processor is idle (condition $b1$) during $(t, t+v)$ then certainly $C(j)$ can be reduced without altering any other $C(i)$ by processing j during the interval, i.e., setting $\delta_r(j, x)=1$ for $t\leqq x\leqq t+v$.

Suppose condition $b2$ has been found. Let σ be the set of all subintervals in (t, ∞) during which $\delta_o(k, t)+\delta_o(j, t)=1$, i.e., those subintervals during which processing is devoted to either j or k.

Clearly $C(k)+C(j)=\min[C(k), C(j)]+\max[C(k), C(j)]$. We will consider a reassignment of $\delta_o(k, x)$ and $\delta_o(j, x)$ for $x\epsilon\sigma$ according to the SRPT discipline.

From the definition of $C(\cdot)$ it follows that

$$\max\{C_o(k), C_o(j)\}=\min\left\{x{:}\int_t^x[\delta_o(k, y)+\delta_o(j, y)]\,dy\geqq S_o(k, t)+S_o(j, t)\right\}$$
$$=\min\left\{x{:}\int_t^x[\delta_r(k, y)+\delta_r(j, y)]\,dy\geqq S_o(k, t)+S_o(j, t)\right\}=\max\{C_r(k), C_r(j)\},$$

i.e., regardless of how j and k are scheduled over σ, the last job will always finish at the same time.

It follows from the same definition that $\min[C(k), C(j)]$ is minimized if we schedule k and j according to SRPT, i.e., the function

$$\min\left\{x{:}\int_t^x\delta(i, y)\,dy\geqq S(i, t)\right\}$$

is minimized by choosing $i=j$. Therefore,

$$\min[C_r(k), C_r(j)]<\min[C_o(k), C_o(j)].$$

It now follows that:

(1) $C_o(i)=C_r(i)$ for $i\neq j$ or k
(2) $\min[C_r(k), C_r(j)]<\min[C_o(k), C_o(j)]$,
(3) $\max[C_r(k), C_r(j)]=\max[C_o(k), C_o(j)]$.

Thus, $N_r(y)=N_o(y)-1$ for $C_r(j)\leqq y\leqq\min[C_o(k), C_o(j)]$, and $N_r(y)=N_o(y)$ otherwise. Therefore, any non-SRPT schedule can be improved in the above sense; and therefore the number in system at any point in time under the SRPT discipline is always less than or equal to the number in system at the same point in time for the same arrival stream under a non-SRPT discipline.

REFERENCES

1. LOUIS W. MILLER AND LINUS SCHRAGE, "The Queue M/G/1 with the Shortest Remaining Processing Time Discipline," *Opns. Res.* **14,** 670–683 (1966).
2. RICHARD W. CONWAY, WILLIAM L. MAXWELL, AND LOUIS W. MILLER, *The Theory of Scheduling*, Addison-Wesley Publishing Co., New York, 1967.

A CLOSED FORM SOLUTION OF CERTAIN PROGRAMMING PROBLEMS*

Martin Pincus

Polytechnic Institute of Brooklyn, Brooklyn, New York

(Received February 13, 1967)

A formula is given for the coordinates of the point that maximizes a given function $F(x_1, \cdots, x_n)$ over the closure of a bounded domain S in n-dimensional Euclidean space. The principal assumption made in deriving the formula is that F attains a global maximum at exactly one point of S. In certain cases the formula may be used to discuss the maximization problem as a function of the parameters involved. Some simple examples are given.

WE CONSIDER the problem of maximizing a given function $F(x_1, \cdots, x_n)$ over a given set S. Under the assumption that F attains a global maximum at exactly one point of S and other not too restrictive conditions we obtain a closed form expression for the coordinates of the maximizing point.

In certain cases, for instance a linear program, we can evaluate the expression exactly. This allows a discussion of the maximization problem in terms of the parameters involved. We carry this out for a two-dimensional linear programming problem and a one-dimensional quadratic programming problem.

The formula obtained is motivated by the classical Laplace asymptotic evaluation of certain integrals.[1] For similar ideas in function space that originated in the work of M. DONSKER, *see* SCHILDER[2] and PINCUS.[3]

Throughout this paper we let S denote the closure of a bounded domain in n-dimensional Euclidean space R^n. The points of R^n are denoted by $X=(x_1, \cdots, x_n)$ and the Euclidean distance by $\|X\|=\sqrt{\sum_{i=1}^{i=n} x_i^2}$. We are now ready to state our main result.

THEOREM. *Let $F(x_1, \cdots, x_n)=(FX)$ be a continuous function on S. Assume that F attains a global maximum at exactly one point $Z=(z_1,\cdots,z_n)$ of S. Then*

$$\lim_{\lambda\to\infty} \frac{\int\cdots_S\int x_i \exp[\lambda F(x_1, \cdots, x_n)]dx_1\cdots dx_n}{\int\cdots_S\int \exp[\lambda F(x_1, \cdots, x_n)]dx_1\cdots dx_n} = z_i. \qquad (i=1, \cdots, n)$$

* The author thanks A. YASPAN for helpful suggestions.