Congestion Control Oracle

There can be one or more solutions for the problem or no solution may satisfy the constraint.

INPUT FILE

Mahimahi takes as input a trace file which represents the time-varying capacity of a cellular network as experienced by a mobile user. Each line gives a timestamp in milliseconds (from the beginning of the trace) and represents an opportunity for one 1500-byte packet to be drained from the bottleneck queue and cross the link. If more than one MTU-sized packet can be transmitted in a particular millisecond, the same timestamp is repeated on multiple lines. For example:

---

0 -> denotes opportunity to send packet/packets of upto MTU size.

3

7

11

11

27

28

39

.

.

.

Considerations and Constraints

1. We need to consider the case where the oracle can choose a packet of any size (0-MTU) to be sent through the bottleneck queue. So, for each delivery opportunity the oracle will decide:
    1. Whether it should avail this delivery opportunity or not.
    2. What's the size of packet to be sent?
2. For ith delivery opportunity, we need to consider all previous delivery opportunities because they ultimately affect the current window_size, delay, and throughput.
3. Trivial: Average throughput must be less than average capacity i.e. Avg(Throughput) <= Avg(Capacity)
4. Trivial: There must be some packets in the pipeline all the time(to avoid corner solutions and to make sure that the link is not empty at any moment) i.e. Instantaneous Throughput >= some value

5. We need to maxmize some utility function.

Problem Statement

set = {all delivery opportunities}

subset = {delivery opportunities used}

We need to find a **subset** of a given **set** so that our constraints are satisfied. We would also like to find the correct size of packets to be sent for ith delivery opportunity but for now lets assume for brevity that our oracle only sends MTU sized packets.

We can formulate it as an **optimal subset selection probelm**.

We are given a finite (our trace file is always finite) and discrete (delivery opportunities after 1ms) set. Lets call this set **U** of size **n**(n denotes the size of trace file i.e. total milliseconds). Let F_1, F_2, ... F_k denote some constraints imposed on subsets of **U** i.e.

F_i : 2^n -> {T, F} (we have 2^n different possibilities)

where T and F denote the logical value of true(use opportunity) or false(ignore opportunity) and i = {1, 2, ..., n}

We also need to provide a criterion function (utility function) to find the best subset. Assume **G** denotes a criterial function of the form:

G : X -> R ;

where R is the set of Real Numbers which means that our utility(criterial) function **G** outputs some real value when given a subset **X** as input.

Our goal now is to find the maximal element

Find **X** such that:

X c **U** (X is a subset of U)

and

F_i(X) for i = 1, 2, 3...k (constraints are followed)

such that

G(X) >= G(Y) (so as to maximize utility)

for any Y C U (Y is a subset of U)