Scheduling Subset Tests:

One-time, Continuous, and how they Relate

Edith Cohen
Microsoft Research SVC
Tel Aviv University

Haim Kaplan
Tel Aviv University

Yishay Mansour
Tel Aviv University

Test Scheduling

Set of ***elements*** E

Set of ***tests*** S where $s \in S$ is a subset of E

Priorities $p_e \in [0,1]$ associated with elements

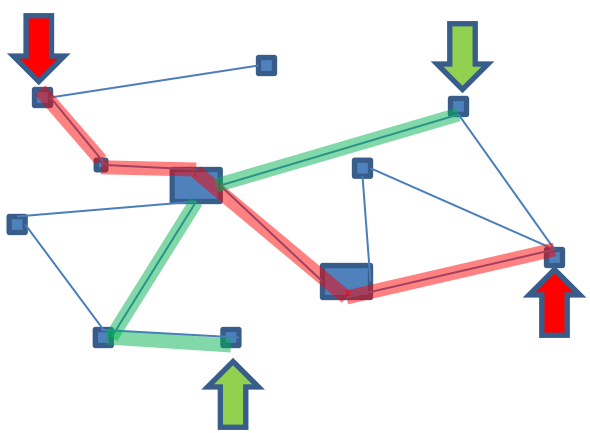
A ***schedule*** is a sequence of tests.

$$\sigma_1, \sigma_2 , \sigma_3 , \sigma_4 , \ldots$$

We are interested in a schedule that covers the elements as "efficiently" as possible

Network Management Application:
Detecting Silent failures

- Network elements: physical (links) or logical (forwarding rules), are subject to failures
- Some elements are more critical than others (have different priorities)
- Presence of failures can be detected by sending probe packets (tests)



Search Application

- Elements are topics / properties /problems
- Priorities are a probability distribution over elements
- Tests are the possible “solutions” (technician that can solve the set of problems)

Test can be applied sequentially

Goal: to find a solution as quickly as possible

One-time versus Continuous

$$E = \{a, b, c, d, e, f, g\}$$

$\boldsymbol{A} = \{a, c, f\}$ $\boldsymbol{B} = \{a, d, e, g\}$ $\boldsymbol{C} = \{b, e, g\}$ $\boldsymbol{D} = \{e, f, g\}$

- **One-time schedule**: discover if a failure is present at time $t = 0$ by applying a sequence of tests.

$$\boldsymbol{B}\ \boldsymbol{A}\ \boldsymbol{C}$$

- **Continuous schedule**: Continuously monitor the system by issuing tests.

$$\boldsymbol{B}\ \boldsymbol{A}\ \boldsymbol{B}\ \boldsymbol{C}\ \boldsymbol{B}\ \boldsymbol{A}\ \boldsymbol{B}\ \boldsymbol{C}\ \boldsymbol{B}\ \boldsymbol{A}\ \boldsymbol{B}\ \boldsymbol{C}\ \ldots..$$

SUM versus MAX objectives

Detection time of failure of element e occurring at time t :

$$T(e,t) = \text{argmin}_i \; e \in \sigma_{t+i}$$

- SUM: Minimize average detection time (weighted by priority)

  One-time: $\sum_{e \in E} p_e$ T(e,0)

  Continuous: $max_t \sum_{e \in E} p_e$ T(e,t)

- MAX: Minimize worst-case detection time (weighted by priority)

  One-time: $max_{e \in E}\, p_e$ T(e,0)

  Continuous: $max_t\, max_{e \in E}\, p_e$ T(e,t)

Rich history: one-time/continuous, max/sum, deterministic/stochastic

- Max, one-time and continuous, with uniform priorities, is classic set cover
- Singletons (element=set) continuous schedules: broadcast disks, teletexts Ammar/Wong 1987 Acharya et al (sigmod 1995) Bar-Noy et al… 2002, 2004
- Kleinrock square root law (continuous, singletons, SUM, stochastic) [Cohen/Shenker Sigcomm 2002]: p2p search
- One-time SUM 4-approximation (tight) [Feige, Lovasz, Tetali 2002]. [Cohen Fiat Kaplan 2003] application to associative search
- Network failure detection (continuous, MAX, stochastic and deterministic): [Kompella et al Infocom 2007] [Nguyen et.al INFOCOM 2009] [Zeng et al CONEXT 2012] [CHKMRYT 2013]….
- Coninuous Stochastic memoryless (aka "fractional") solution: LP for MAX convex program for SUM, reduction from memoryless to deterministic [CHKMRYT 2013]

Our Work

- Unified treatment of all variants, focus on deterministic schedules
- Approximation algorithms
- Understand the relation between the one-time and continuous objectives per-instance

Approximation Ratio Results

	Max	Sum
One-time	$\Theta(\log n)$	$4-\varepsilon$, 4 [Feige,Lovasz, Tetali 2002]
Continuous	$\Omega(\log n)$ $O(\log^2 n)$	O(log n)

n is the number of elements

Upper bounds for continuous improve over previous that depend on the max number of tests containing an element [CHKMRT 2013].

Lower bounds on approximation ratio for MAX via reduction to set cover and using [Feige 1998]

Relating one-time and continuous

Thm: On any scheduling instance $I = (E, S, p)$, for both MAX and SUM objectives

$$\frac{OPT-continuous(I)}{OPT-one-time(I)} = O(\log n)$$

There is a family of instances I_m such that

$$\frac{OPT-continuous(I_m)}{OPT-one-time(I_m)} \geq H_m \quad ,$$

where H_m are the Harmonic numbers

Upper Bound on Ratio: SUM

Build a Continuous Schedule from a One-Time schedule:

$s_1, s_2, s_3, s_4, s_5, s_6, \ldots, s_m$ (each test appears at most once)

Continuous schedule:

- Assign frequency $q_i = \frac{1}{i \; H_m}$ to test s_i
- Construct a deterministic continuous schedule which has test s_i at least once every ${}^2/_{q_i}$ steps [Bar-Boy et al 02]

Analysis: An element e that is first covered by test s_i contributes $i \, p_e$ to the one-time objective and $\leq \; p_e \, {}^2/_{q_i} = 2 \, i \, p_e / H_m$ to the continuous objective

Ratio is at most $2 \, H_m$

Lower Bound on Ratio

Family of singleton schedules with m tests/elements with priorities

$$p_1 \geq p_2 \geq p_3 \geq \cdots \geq p_m$$

MAX: Priorities $p_i \propto \frac{1}{i}$

The optimal one-time schedule is by decreasing priority: cost is $\max_i i\, p_i = 1$

Continuous memoryless (stochastic) schedule with frequencies q_i , $\sum_{i=1}^{m} q_i = 1$, has cost $\max_i p_i / q_i$.

This is minimized for $q_i \propto p_i \rightarrow q_i = \frac{1}{iH_m}$

Therefore optimal cost $\max_i p_i / q_i = H_m$

$\rightarrow$ ratio is H_m

Lower Bound on Ratio

Family of singleton schedules with m tests/elements with priorities

$$p_1 \geq p_2 \geq p_3 \geq \cdots \geq p_m$$

SUM: Priorities $p_i \propto \frac{1}{i^2}$

The optimal one-time schedule is by decreasing priority:

Cost is $\sum_{i=1}^{m} i \; p_i = \sum_{i=1}^{m} \frac{1}{i} = H_m$

Continuous memoryless (stochastic) schedule with frequencies q_i , where $\sum_{i=1}^{m} q_i = 1,$ has cost $\sum_{i=1}^{m} p_i \, / \, q_i$.

This is minimized for $q_i \propto \sqrt{p_i}$ (Kleinrock's square-root law)

$q_i = \frac{1}{iH_m}$ therefore optimal cost is $\sum_{i=1}^{m} p_i \, / \, q_i = {H_m}^2$

$\rightarrow$ ratio is H_m

What is the “typical” ratio ?

Zipf priorities singleton instances

Priorities $p_i \propto \frac{1}{i^{\alpha}}$

- SUM: ratio is constant except when $\alpha = 2$

- MAX: ratio is constant except when $\alpha = 1$

Open problems

Lower bound/Tighter upper bound for Continuous MAX

Continuous SUM

Thank you!