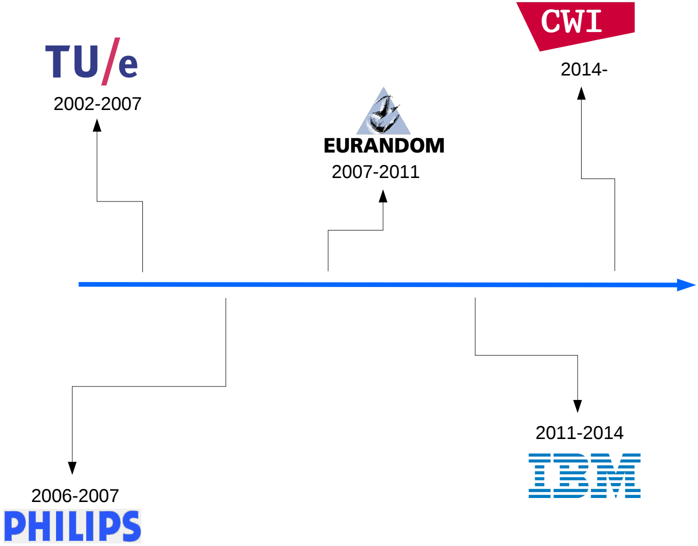
Peak load reduction for distributed backup scheduling

Peter van de Ven

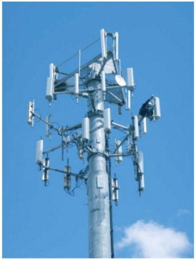
joint work with

Angela Schörgendorfer (Google) and Bo Zhang (IBM Research)


TU/e
2002-2007
CWI
2014-
EURANDOM
2007-2011
2006-2007
PHILIPS
2011-2014
IBM

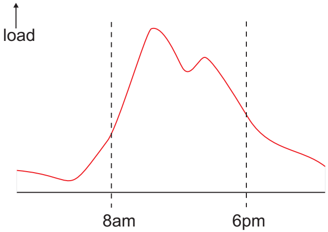
My research interests are in the design and analysis of large (distributed) stochastic systems:

- performance analysis; scheduling; stochastic optimization and control



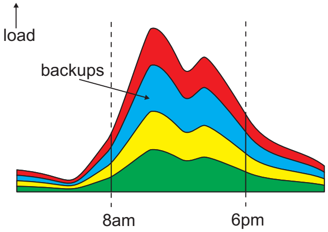
Part I:

Peak load reduction for distributed backup scheduling

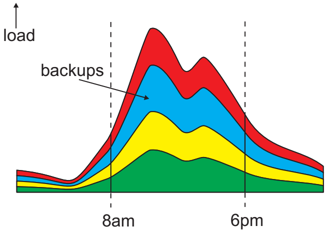


Load is highly varying throughout the day

- network costs are non-linear in the hourly load
- aim to reduce network costs by shifting peak load

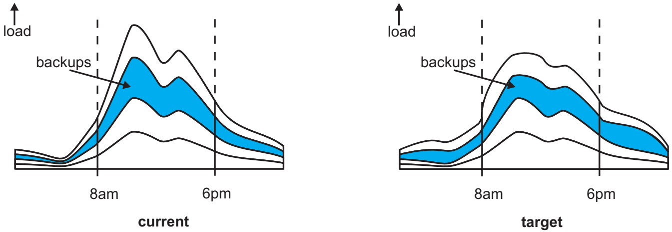


- the load is comprised of different types of traffic
- we focus on the backup traffic



Users (workstations) aim to do daily backups

- the backup transmitted across the network to a central server
- backup traffic can be shifted because of delay-tolerance

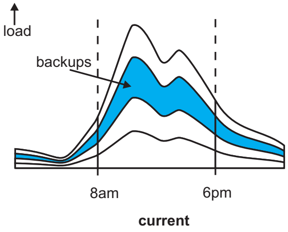


Backups are initiated by the users, based on local information only

- backup policy is characterized by hourly backup probabilities

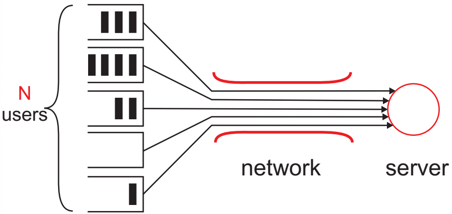
However

- users are not always connected
- the server may have some capacity constraint



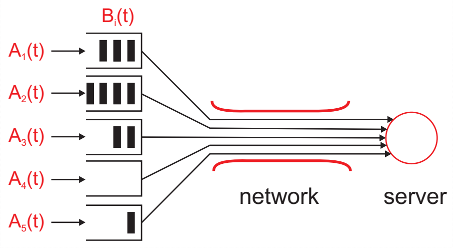
Question 1: How much traffic can the network handle for given backup probabilities?
Question 2: How to choose the backup probabilities to minimize the network costs while ensuring regular backups?

Model outline

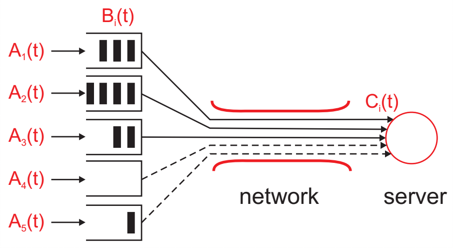


Consider N users that generate data (bits) over time

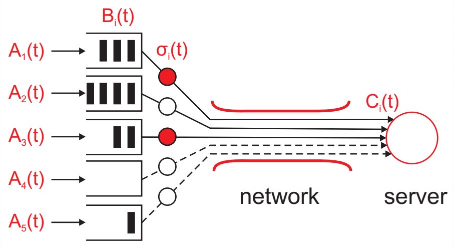
- user backup to a central server
- access to server over a data network



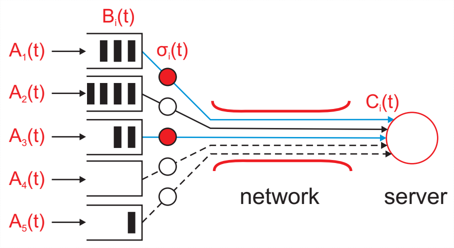
- time is slotted
- user i generates $A_i(t)$ bits in slot t
- data is added to the backlog $B_i(t)$



- users can only do a backup when connected to the network
- connectivity is represented by $C_i(t) \in \{0, 1\}$, $i = 1, \ldots, N$



Connected users may decide to do a backup ($\sigma_i(t) = 1$) or not ($\sigma_i(t) = 0$)



Connected users may decide to do a backup ($\sigma_i(t) = 1$) or not ($\sigma_i(t) = 0$)

The backlog evolution and offered backup load are given by

$$B_i(t+1) = (A_i(t) + B_i(t))(1 - C_i(t)\sigma_i(t)),$$

$$H(t) = \sum_{i=1}^{N} (A_i(t) + B_i(t)) C_i(t)\sigma_i(t)$$

User type & Backup probability

Consider the number of days since backup $W_i(t) \in \{0, 1, \dots\}$ (user type):

$$W_i(t+1) = W_i(t)\big(1 - \sigma_i(t) C_i(t)\big) + \mathbb{1}_{\{\eta(t+1)=0\}},$$

with $\eta(t) := t \mod T$ the hour of the day

User type & Backup probability

Consider the number of days since backup $W_i(t) \in \{0, 1, \dots\}$ (user type):

$$W_i(t+1) = W_i(t)\big(1 - \sigma_i(t) C_i(t)\big) + \mathbb{1}_{\{\eta(t+1)=0\}},$$

with $\eta(t) := t \mod T$ the hour of the day

The backup decision is random and may depend on $W_i(t)$ and the hour of the day $u(t) \in \{0, \dots, T-1\}$:

$$\sigma_i(t) = \begin{cases} 1 & \text{w.p. } \nu(u(t), W_i(t)) \\ 0 & \text{otherwise} \end{cases}$$

Capacity & network cost minimization

Let $\mathbf{W}(t) = (W_1(t), \ldots, W_N(t))$ and consider the Markov process $U := \{(\eta(t), \mathbf{W}(t)), t = 0, 1, \ldots\}$

For fixed ν, we can analyze U to obtain insights into the backup system

The stationary distribution π of the process U satisfies

$$\pi(u,w) = \pi(0,1)k_1(u,w), \quad u = 0,\dots,T-1;\ w \geq 1,$$
$$\pi(u,0) = \pi(0,1)k_2(u), \quad u = 1,\dots,T-1,$$

where

$$k_1(u,w) := m(0,u-1,w)\prod_{y=1}^{w-1} m(0,T-1,y),$$
$$k_2(u) := \sum_{v=1}^{u}\sum_{y=1}^{\infty} k_1(v-1,y)c(v-1)\nu(v-1,y),$$

and with normalizing constant

$$\pi(0,1) = \Big[\sum_{w=1}^{\infty}\sum_{u=0}^{T-1} k_1(u,w) + \sum_{u=1}^{T-1} k_2(u)\Big]^{-1}.$$

Let

$$m(u, u', w) := \prod_{v=u}^{u'} (1 - c(v)\nu(v, w)), \;\; u, u' = 0, \ldots, T-1; \;\; w = 0, 1, \ldots,$$

and define

$$\begin{aligned} g(u, w) := 1 - m(u, T-1, w)m(0, u-1, w+1), \\ u, u' = 0, \ldots, T-1; \quad w = 0, 1, \ldots, \end{aligned}$$

Proposition

If

$$\alpha(u) := \lim_{w \to \infty} w\left(g(u, w) - w^{-1}\right)$$

exists and $\alpha(u) \in (0, \infty)$ *for all* $u \in \{0, 1, ..., T-1\}$*, then* U *is positive recurrent.*

Lemma

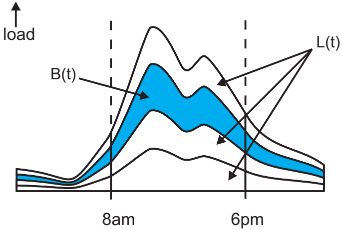
The expected backlog size of a type-w user at hour u:

$$S(u,w)=\sum_{u'=0}^{T-1} p(u',u,w)\mathbb{E}_\pi\left[B(t)\mid \eta(t)=u,\ W(t)=w,\ \eta(\tau^*(t))=u'\right].$$

where

$$\mathbb{E}_\pi\left[B(t)\mid \eta(t)=u,\ W(t)=w,\ \eta(\tau^*(t))=u'\right]$$
$$=\begin{cases}\sum_{v=u'+1}^{u-1} a(v), & w=0,\\ \sum_{v=u'+1}^{T-1} a(v)+(w-1)\hat{a}+\sum_{v=0}^{u-1} a(v), & w\geq 1,\end{cases}$$

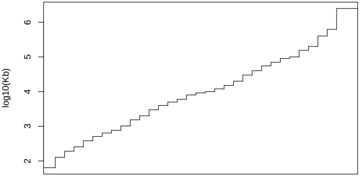
$$p(u',u,w)=\begin{cases}\sum_{w'=0}^{\infty}\frac{\pi(u',w')}{\pi(u,0)}c(u')\nu(u',w')m(u'+1,u-1,0), & w=0,\\ \sum_{w'=0}^{\infty}\frac{\pi(u',w')}{\pi(0,1)}c(u')\nu(u',w')m(u'+1,T-1,0), & w\geq 1.\end{cases}$$



We consider the sum of $H(t)$ (backup load) and $L(t)$ (extraneous load)

- denote $H(t;\nu)$ to reflect the impact of ν

Network costs



The data network is associated with a load-dependent hourly cost function $g : \mathbb{R} \mapsto \mathbb{R}$

We are interested in the average per-day network costs ($T = 24$)

$$G(\nu) = \sum_{u=0}^{T-1} \mathbb{E}\{g\big(L(u) + H(u;\nu)\big)\},$$

We approximate

$$G(\nu) \approx \widetilde{G}(\nu) := \sum_{u=0}^{T-1} g\big(\mathbb{E}\big[L(u)\big] + \mathbb{E}_{\pi}\big[H(u;\nu)\big]\big),$$

The $\mathbb{E}\big[L(u)\big]$ can be obtained from measurements. Exploiting user independence:

$$\mathbb{E}_{\pi}[H(u;\nu)] = NT \sum_{w=0}^{\infty} \pi(u,w)\big(S(u,w) + a(u)\big)c(u)\nu(u,w),$$

We aim to minimize the network costs, while ensuring regular backups

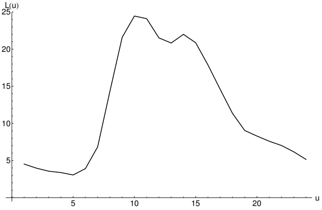
$$\min_{\nu} \widetilde{G}(\nu)$$

$$\text{such that } T \sum_{w=w_j}^{\infty} \pi(0,w) \leq \gamma_j, \quad j = 1, 2, \ldots.$$

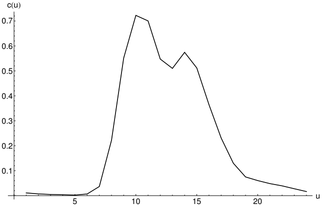
This problem is non-convex.

Implementation

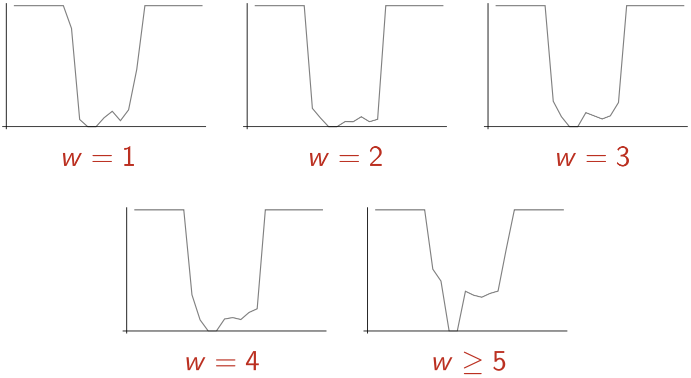
We consider an IBM location with $N = 5397$ users



load $L(u)$



connectivity $c(u)$



We truncate the state space at $w = 5$ and compute the $\nu(u, w)$

- the probability is high at night and low during working hours
- users that have not done a backup in a while are more likely to schedule during peak hours

Comparison to original scheduling table

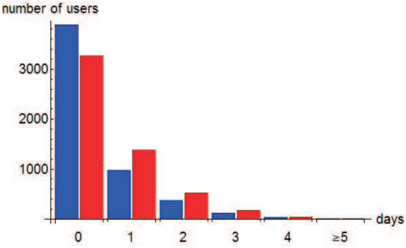
Europe team's schedule:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0.5	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	0.1	0	0	0	0	0	0	0.12	0.5	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	0.1	0	0	0	0.12	0.14	0	0.16	0.5	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	0.1	0	0	0	0.12	0.14	0.2	0.33	0.5	1	1	1	1	1	1	1
=5	1	1	1	1	1	1	1	1	0.1	0.12	0.14	0.14	0.16	0.19	0.25	0.33	0.5	1	1	1	1	1	1	1

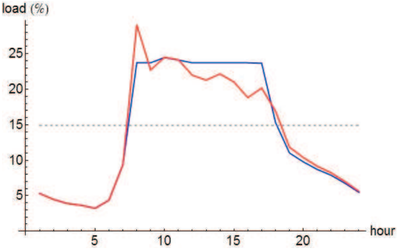
Our schedule:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
1	1	1	1	1	1	1	1	0.81	0.06	0	0	0.07	0.13	0.05	0.14	0.47	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	0.15	0.07	0	0	0.04	0.04	0.08	0.04	0.06	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	0.21	0.08	0	0	0.12	0.09	0.06	0.09	0.2	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	0.3	0.09	0	0	0.1	0.11	0.09	0.15	0.18	1	1	1	1	1	1	1	1
=5	1	1	1	1	1	1	1	0.51	0.41	0	0	0.33	0.3	0.28	0.31	0.33	0.67	1	1	1	1	1	1	1

Comparison to original scheduling table - Performance

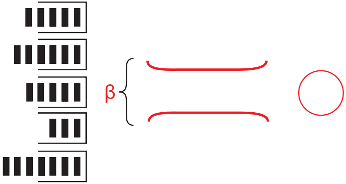


days without backup



total load

Constrained system



Consider a constraint on the total amount of backup traffic:

$$B_i(t+1) = (A_i(t) + B_i(t))(1 - C_i(t)\sigma_i(t)\min\{1, \beta/B(t)\})$$

Users are no longer independent

- stability conditions and proofs are more involved
- use mean field limit to compute long-term average behavior