DM826 – Spring 2012

Modeling and Solving Constrained Optimization Problems

# Lecture 11
# Search

Marco Chiarandini

Department of Mathematics & Computer Science
University of Southern Denmark

# Overview

- Random restarts
- Implementation issues
- Search in gecode-python
- Filtering algorithms in Scheduling

# Outline

1. Random Restart

2. Implementation Issues

3. Scheduling

# Algorithm Survival Analysis

Run time distributions

- $T \in [0, \infty]$ time to find a solution on an instance
- $F(t) = \Pr\{T \leq t\}$ $F : [0, \infty] \mapsto [0, 1]$ cdf
- $f(t) = \frac{dF(t)}{dt}$ pdf
- $S(t) = \Pr\{T > t\} = 1 - F(t)$ survival function
- $h(t) = \lim_{\Delta t \to 0} \Pr\{t \leq T < t + \Delta t \mid T \geq t\} \Delta t$ hazard function
- $H(t) = \int_0^t h(s) ds$ $h(s) \frac{f(t)}{S(t)}$ $H(t) = -\log S(t)$ cumulative hazrd function
- $E[T] = \int_0^\infty t f(t) dt = \int_0^1 t dF(t) = \int_0^\infty S(t) dt$ expected run time

# Heavy Tails

$$F(t) \rightarrow_{t\rightarrow\infty} 1 - C^{t^{-\alpha}} \qquad \text{(Pareto like distr.)}$$

In practice, this means that

- most runs are relatively short, but the remaining few can take a very long time.
- Depending on $C$, $\alpha$, the mean of a heavy-tailed distribution can be finite or not, while higher moments are always infinite.
- Gomes and Selman [2005] explain that the length of a single run depends on the order with which randomized backtracking assigns values to the variables.

  In some runs, backtracking has to search very deep branches in the tree of possible solutions before finding a contradiction.

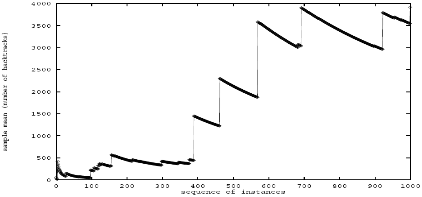
  The same instance may be very easy if solved with a different random reordering of the variables.

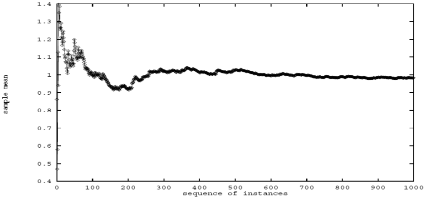
  This is an example phenomenon which is difficult to study based on

# Characterization of Run-time
## Heavy Tails

Gomes et al. [2000] analyze the mean computational cost to find a solution on a single instance

On the left, the observed behavior calculated over an increasing number of runs.
On the right, the case of data drawn from normal or gamma distributions




- The use of the median instead of the mean is recommended
- The existence of the moments (*e.g.*, mean, variance) is determined by the tails behavior: a case like the left one arises in presence of long tails

# Characterization of runtime

Parametric models used in the analysis of run-times to
exploit the properties of the model (eg, the character of tails and completion rate)

Procedure:

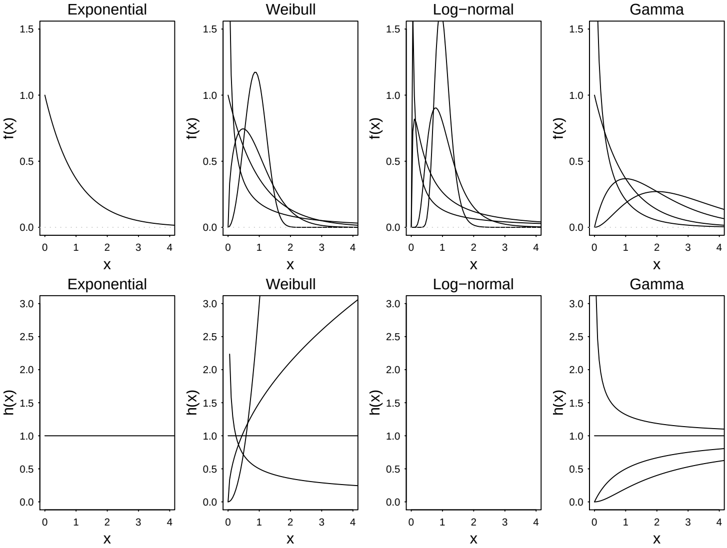
- choose a model
- apply fitting method
  maximum likelihood estimation method:

$$\max_{\theta \in \Theta} \log \prod_{i=1}^{n} p(X_i, \theta)$$

- test the model

# Parametric models

The distributions used are [Frost et al., 1997; Gomes et al., 2000]:



# Characterization of Run-time

Motivations for these distributions:

- qualitative information on the completion rate (= hazard function)
- empirical good fitting

To check whether a parametric family of models is reasonable the idea is to make plots that should be linear. Departures from linearity of the data can be easily appreciated by eye.

Example: for an exponential distribution:

$$\log S(t) = -\lambda t \qquad S(t) = 1 - F(t) \;\; \text{is the survivor function}$$

$\leadsto$ the plot of $\log S(t)$ against $t$ should be linear.

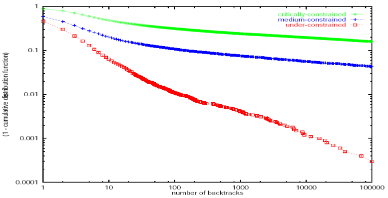
Similarly, for the Weibull the cumulative hazard function is linear on a log-log plot

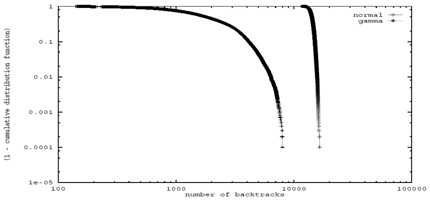
$\leadsto$ heavy tail if $S(t)$ in log-log plot is linear with slope $-\alpha$

# Characterization of Run-time
## Heavy Tails

Graphical check using a log-log plot:

- heavy tail distributions approximate linear decay,
- exponentially decreasing tail has faster-than linear decay




Long tails explain the goodness of random restart. Determining the cutoff time is however not trivial.

# Extreme Value Statistics

- Extreme value statistics focuses on characteristics related to the tails of a distribution function
  1. extreme quantiles (*e.g.*, minima)
  2. indices describing tail decay

- 'Classical' statistical theory: analysis of means.
  Central limit theorem: $X_1, \ldots, X_n$ i.i.d. with $F_X$

$$\sqrt{n}\frac{\bar{X}-\mu}{\sqrt{Var(X)}} \xrightarrow{D} N(0,1), \qquad \text{as } n \to \infty$$

Heavy tailed distributions: mean and/or variance may not be finite!

# Extreme Value Statistics

## Extreme values theory

- $X_1, X_2, \ldots, X_n$ i.i.d. $F_X$
  Ascending order statistics $X_n^{(1)} \leq \ldots \leq X_n^{(n)}$
- For the minimum $X_n^{(1)}$ it is $F_{X_n^{(1)}} = 1 - [1 - F_X^{(1)}]^n$ but not very useful in practice as $F_X$ unknown
- Theorem of [Fisher and Tippett, 1928]:
  "almost always" the normalized extreme tends in distribution to a generalized extreme distribution (GEV) as $n \to \infty$.

In practice, the distribution of extremes is approximated by a GEV:

$$F_{X_n^{(1)}}(x) \sim \begin{cases} \exp(-1(1 - \gamma \frac{x-\mu}{\sigma})^{-1/\gamma}, & 1 - \gamma \frac{x-\mu}{\sigma} > 0, \gamma \neq 0 \\ \exp(-\exp(\frac{x-\mu}{\sigma})), & x \in \mathbf{R}, \gamma = 0 \end{cases}$$

Parameters estimated by simulation by repeatedly sampling $k$ values $X_{1n}, \ldots, X_{kn}$, taking the extremes $X_{kn}^{(1)}$, and fitting the distribution.
$\gamma$ determines the type of distribution: Weibull, Fréchet, Gumbel, ...

# Extreme Value Statistics

## Tail theory

- Work with data exceeding a high threshold.
- Conditional distribution of exceedances over threshold $\tau$

$$1 - F_\tau(y) = P(X - \tau > y \,|\, X > \tau) = \frac{P(X > \tau + y)}{P(X > \tau)}$$

- If the distribution of extremes tends to GEV distribution then there exist a Pareto-type function such that for some $\gamma > 0$

$$1 - F_X(x) = x^{-\frac{1}{\gamma}} \ell_F(x), \qquad x > 0,$$

  with $\ell_F(x)$ a slowly varying function at infinity.

In practice, fit a function $Cx^{-\frac{1}{\gamma}}$ to the exceedances:
$Y_j = X_i - \tau$, provided $X_i > \tau$, $j = 1, \ldots, N_\tau$.
$\gamma$ determines the nature of the tail

# Characterization of Run-time

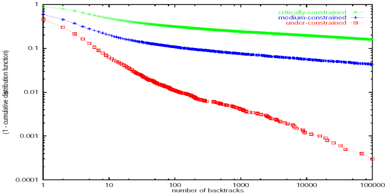
## Heavy Tails

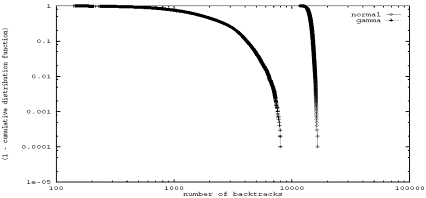
The values estimated for $\gamma$ give indication on the tails:

- $\gamma > 1$: long tails hyperbolic decay (the completion rate decreases with $t$) and mean not finite
- $\gamma < 1$: tails exhibit exponential decay

Graphical check using a log-log plot:

- heavy tail distributions approximate linear decay,
- exponentially decreasing tail has faster-than linear decay




Long tails explain the goodness of random restart. Determining the cutoff time is however not trivial.

# Restart strategies

- Restart strategy: execute a sequence of runs of a randomized algorithm, to solve a single problem instance, stopping the $r$-th run after a time $\tau(r)$ if no solution is found, and restarting the algorithm with a different random seed

- defined by a function $\tau : \mathbb{N} \to \mathbb{R}^+$ producing the sequence of thresholds $\tau r$ employed.

- origins in the field of communication networks
(Fayolle et al., 1978) derive the optimal timeout for a simple "send and wait" communication protocol, maximizing the transmission rate.

- It can be proved that restarts is beneficial under two conditions: if the survival function decreases less fast than an exponential, and if the RTD is improper.

Luby et al. [1993] prove that:

- if $F(t)$ is known:
  the optimal restart strategy is uniform, i.e., $\tau(r) = \tau$.
  Optimal cutoff time $\tau^*$ can be evaluated minimizing the expected value of the total run-time $T_\tau$:

$$E\{T_\tau\} = \frac{\tau - \int_0^\tau F(t)dt}{F(\tau)}$$

- if $F(t)$ is not known, suggested a universal, non-uniform restart strategy, whose cutoff sequence is composed of powers of 2:
  $r = 1, 2, \ldots, \tau(r) := 2^{j-1}$ if $r = 2^j - 1; \tau(r) := \tau(r - 2^{j-1} + 1)$ if $2^{j-1} \leq r < 2^j - 1$.
  (when $2^{j-1}$ is used twice, $2^j$ is the next.)
  whose performance $t^U$ is bounded with high probability with respect to the expected run-time $E\{T_{\tau^*}\}$ of the optimal uniform strategy, as
  $t_U \leq 192E\{T_{\tau^*}\}(\log E\{T_{\tau^*}\} + 5)$
  Eg. Sequence $\{1, 1, 2, 1, 1, 2, 4, 1, \ldots\}$,

# Outline

1. Random Restart

2. Implementation Issues

3. Scheduling

# Architecture

- Detecting failure and entailment.
- Implementing domains.
- Finding dependent propagators.
- State restoration.
- Variables for propagators.
- Multiple variable occurrences and unification.
- Private state.

Propagation Services

- Events
- Selecting the next propagator
- Value operations

Variable Domains

- Iterators
- Domain operations
- Subscriptions
- Domain representation

Propagators

- Life cycle.
- Idempotent propagators
- Multiple value removals
- Amount of information available
- Use of private state (auxiliary data structures, incrementality, domain information, fixed parameters)
- Multiple variable occurrences

Search

- Branching
- State Restoration
  - copying
  - trailing – time-stamping, multiple-value trail
  - recomputation
- Expressiveness
- Predefined exploration strategies

# Outline

# Edge Finding

$$\text{If } L_J - E_{J\cup\{i\}} < p_i + p_J, \text{ then } i \gg J \quad (a)$$
$$\text{If } L_{J\cup\{i\}} - E_J < p_i + p_J, \text{ then } i \ll J \quad (b)$$

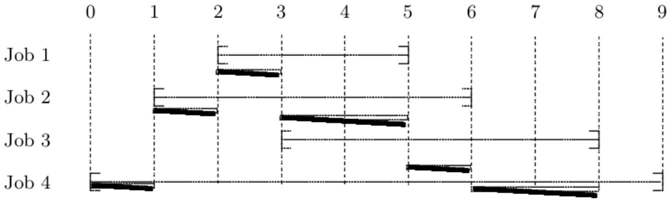
$$\text{If } i \gg J, \text{ then update } E_i \text{ to } \max\left\{E_i, \max_{J'\subset J}\{E_{J'} + p_{J'}\}\right\}.$$

$$\text{If } i \ll J, \text{ then update } L_i \text{ to } \min\left\{L_i, \min_{J'\subset J}\{L_{J'} - p_{J'}\}\right\}.$$

| $j$ | $p_j$ | $E_j$ | $L_j$ |
|---|---|---|---|
| 1 | 1 | 2 | 5 |
| 2 | 3 | 1 | 6 |
| 3 | 1 | 3 | 8 |
| 4 | 3 | 0 | 9 |

# $O(n^2)$ algorithm

| $j$ | $p_j$ | $E_j$ | $L_j$ |
|---|---|---|---|
| 1 | 1 | 2 | 5 |
| 2 | 3 | 1 | 6 |
| 3 | 1 | 3 | 8 |
| 4 | 3 | 0 | 9 |




| $i$ | $J_i$ | $\bar{p}$ | $k$ | $J_{ik}$ | $L_k - E_i$ | $p_i + \bar{p}_{J_{ik}}$ |
|---|---|---|---|---|---|---|
| 1 | $\{1,2,3,4\}$ | $(1,2,1,2)$ | 4 | $\{2,3,4\}$ | $9-2$ | $1+5$ |
| | | | 3 | $\{2,3\}$ | $8-2$ | $1+3$ |
| | | | 2 | $\{2\}$ | $6-2$ | $1+3$ |
| 2 | $\{1,2,3,4\}$ | $(1,3,1,2)$ | 4 | $\{1,3,4\}$ | $9-1$ | $3+4$ |
| | | | 3 | $\{1,3\}$ | $8-1$ | $3+2$ |
| | | | 1 | $\{3\}$ | $5-1$ | $3+1$ |
| 3 | $\{2,3,4\}$ | $(0,2,1,2)$ | 4 | $\{2,4\}$ | $9-3$ | $1+4$ |
| | | | 2 | $\{2\}$ | $6-3$ | $1+2$ |
| 4 | $\{1,2,3,4\}$ | $(1,3,1,3)$ | 3 | $\{1,2,3\}$ | $8-0$ | $3+5$ |
| | | | 2 | $\{1,2\}$ | $6-0$ | $3+4$ |

Conclude that $4 \gg \{1,2\}$ and update $E_4$ from 0 to 5

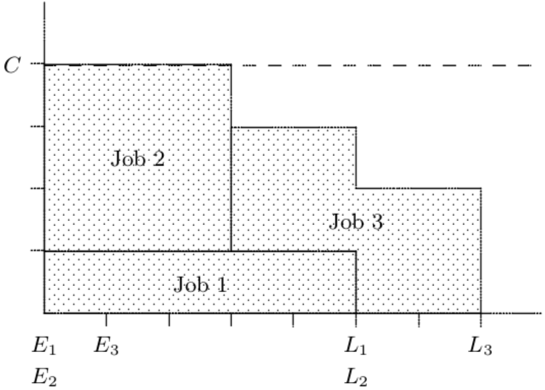
# Not first, Not Last

$$\text{If } L_J - E_i < p_i + p_J \text{, then } \neg(i \ll J). \quad (a)$$
$$\text{If } L_i - E_J < p_i + p_J \text{, then } \neg(i \gg J). \quad (b)$$

$$\text{If } \neg(i \ll J)\text{, then update } E_i \text{ to } \max\left\{E_i, \min_{j \in J}\{E_j + p_j\}\right\} \qquad (a)$$
$$\text{If } \neg(i \gg J)\text{, then update } L_i \text{ to } \min\left\{L_i, \max_{j \in J}\{L_j - p_j\}\right\} \qquad (b)$$

# Cumulative Scheduling

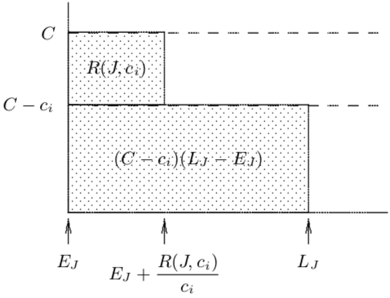
| $j$ | $p_j$ | $c_j$ | $E_j$ | $L_j$ |
|---|---|---|---|---|
| 1 | 5 | 1 | 0 | 5 |
| 2 | 3 | 3 | 0 | 5 |
| 3 | 4 | 2 | 1 | 7 |



# Edge Finding



$$\text{If } e_i + e_J > C \cdot (L_J - E_{J\cup\{i\}}), \text{ then } i > J. \quad (a)$$
$$\text{If } e_i + e_J > C \cdot (L_{J\cup\{i\}} - E_J), \text{ then } i < J. \quad (b)$$




$$\text{If } i > J \text{ and } R(J, c_i) > 0, \text{ update } E_i \text{ to } \max\left\{E_i, E_J + \frac{R(J, c_i)}{c_i}\right\}.$$
$$\text{If } i < J \text{ and } R(J, c_i) > 0, \text{ update } L_i \text{ to } \min\left\{L_i, L_J - \frac{R(J, c_i)}{c_i}\right\}.$$

# References

Frost D., Rish I., and Vila L. (1997). **Summarizing CSP hardness with continuous probability distributions.** In *Proceedings of AAAI/IAAI*, pp. 327–333.

Gagliolo M. (2010). **Online Dynamic Algorithm Portfolios**. Ph.D. thesis, IDSIA/USI.

Gomes C. and Selman B. (2005). **Can get satisfaction**. *Nature*, 435, pp. 751–752.

Gomes C., Selman B., Crato N., and Kautz H. (2000). **Heavy-tailed phenomena in satisfiability and constraint satisfaction problems**. *Journal of Automated Reasoning*, 24(1-2), pp. 67–100.

Luby M., Sinclair A., and Zuckerman D. (1993). **Optimal speedup of las vegas algorithms**. *Information Processing Letters*, 47(4), pp. 173–180.