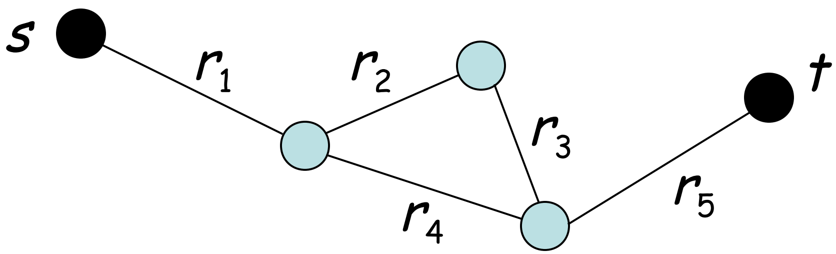
Bounded relative error and Vanishing relative error in Monte Carlo evaluation of static Network Reliability measures

Héctor Cancela
Depto. de Investigación Operativa
Instituto de Computación, Facultad de Ingeniería
Universidad de la República, Uruguay

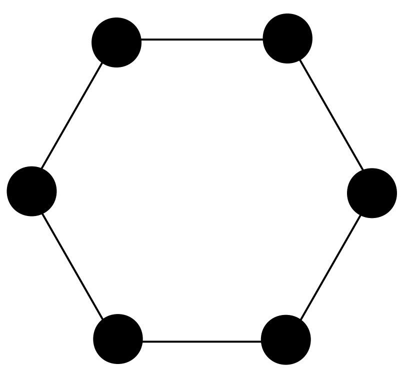
AGENDA

- Classical network reliability model.
- Monte Carlo Simulation
- Variance Reduction methods
- Recursive Variance Reduction
- Asymptotic Zero Variance
- Conclusions

1 NETWORK RELIABILITY



$$R_{st} = r_1(r_2 r_3 + r_4 - r_2 r_3 r_4) r_5$$

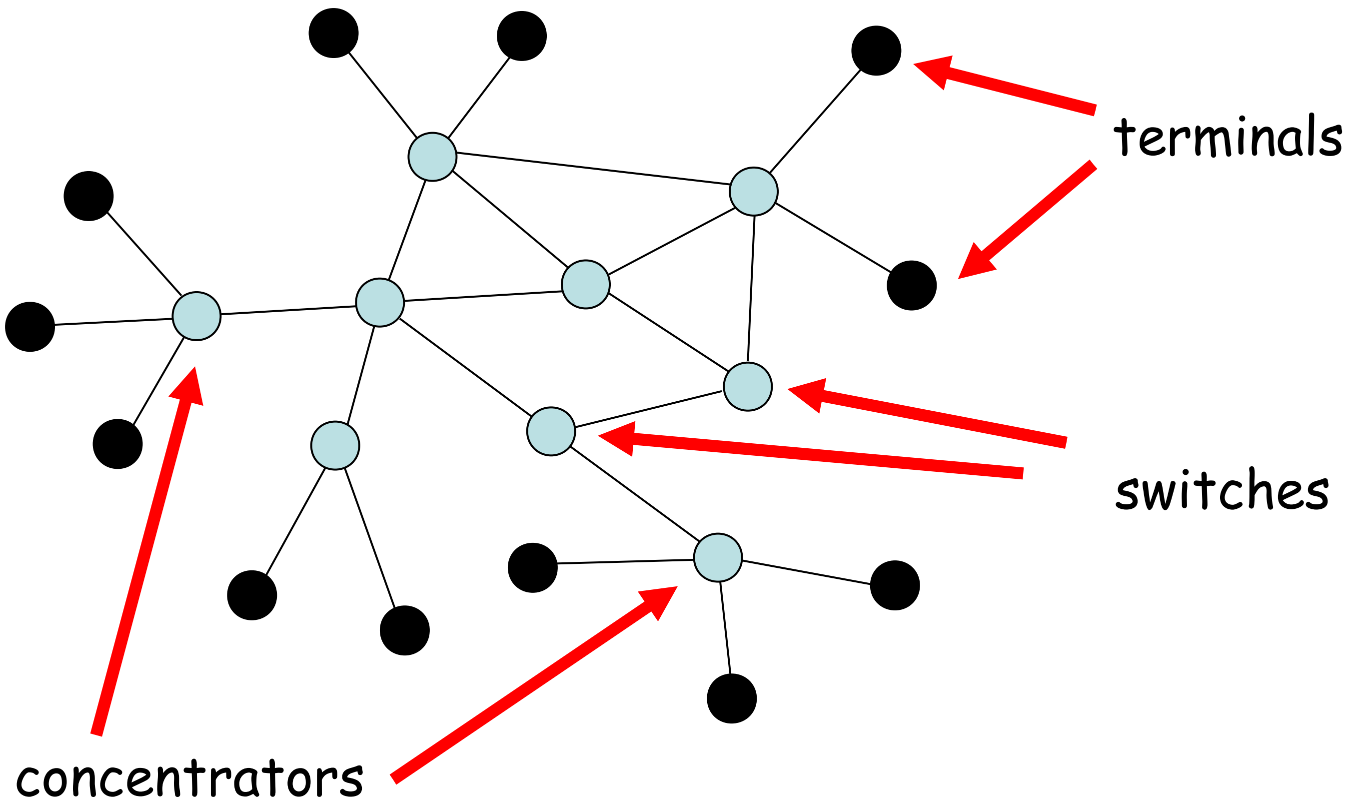


$$r_i = r$$

$$R_{all} = r^5 (6 - 5r)$$

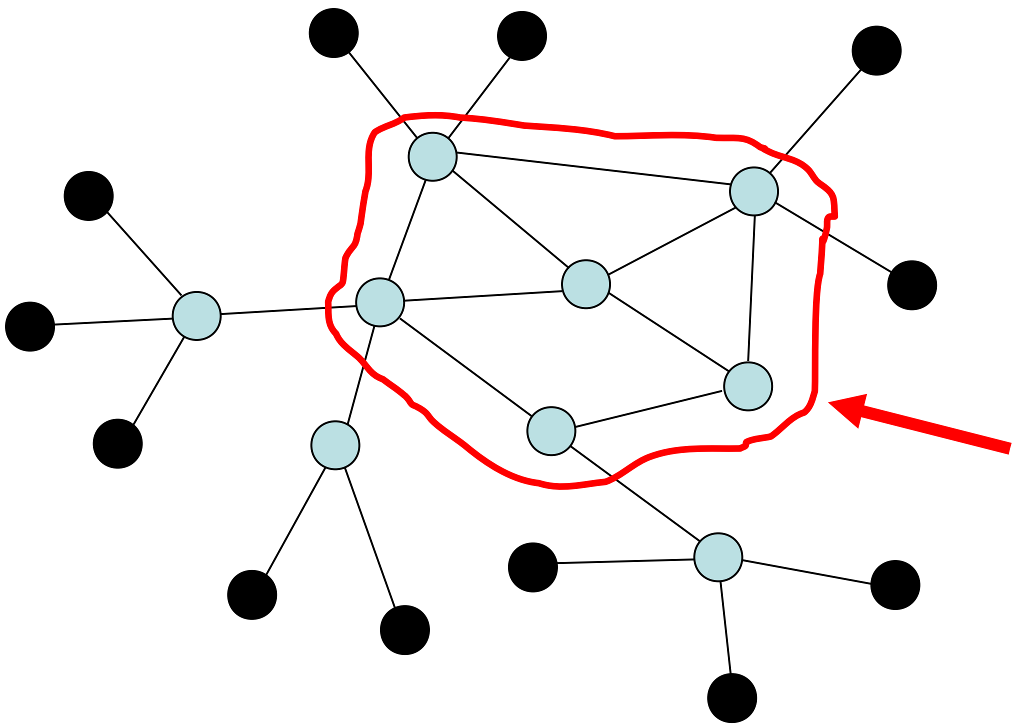
REFERENCE MODEL

A COMMUNICATION NETWORK



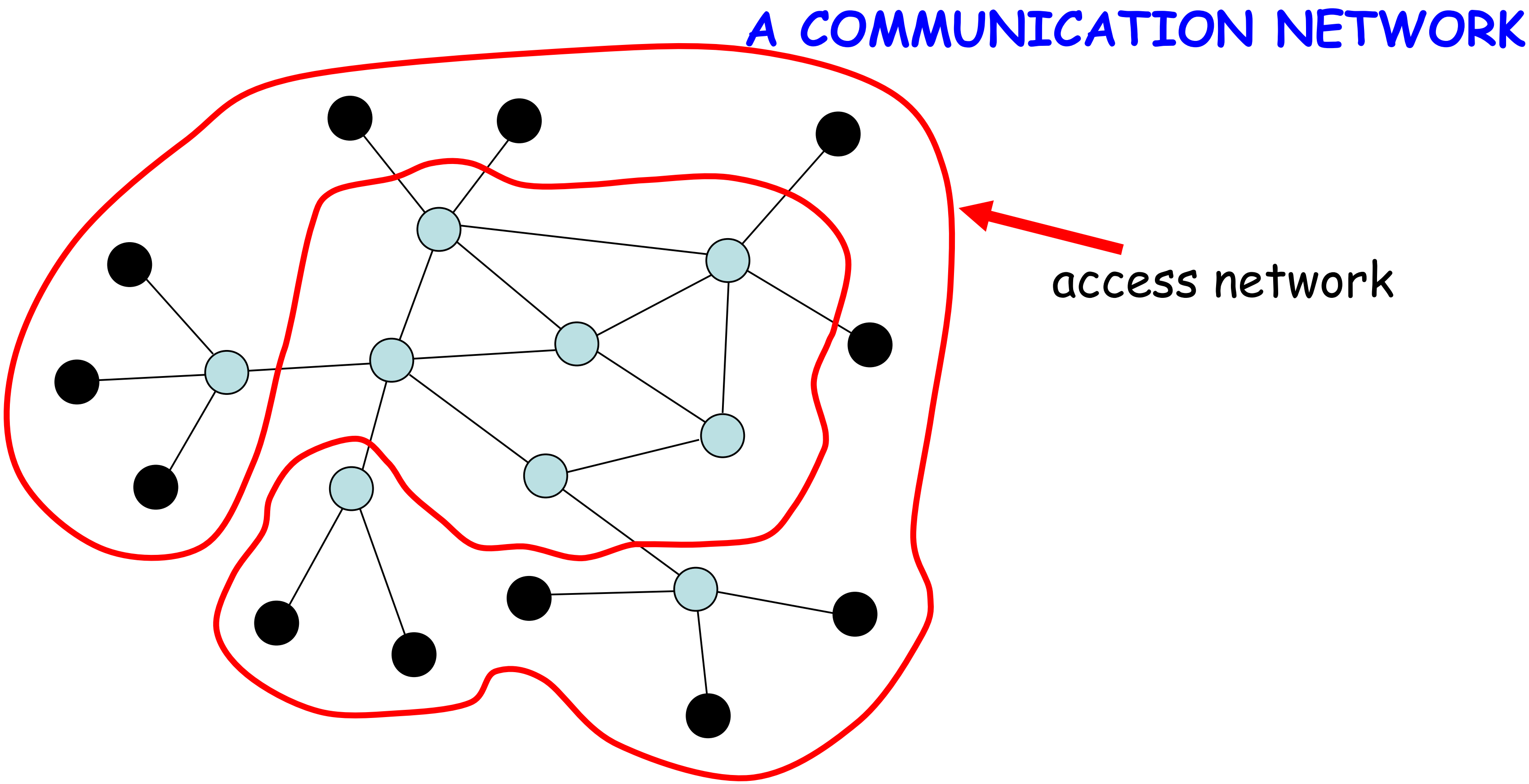
REFERENCE MODEL

A COMMUNICATION NETWORK

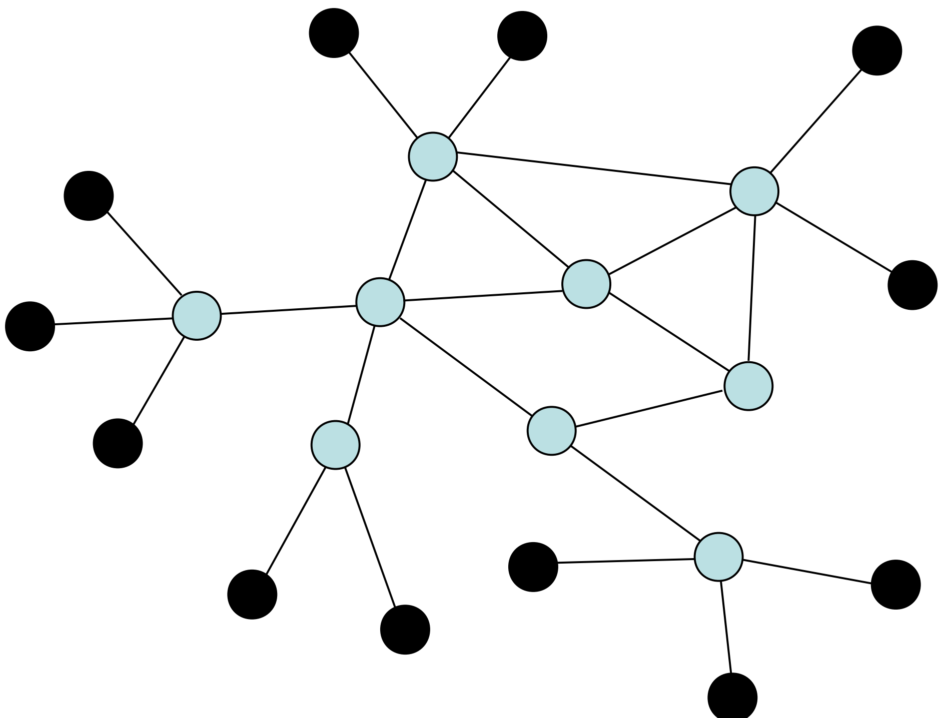


backbone, or core

REFERENCE MODEL



REFERENCE MODEL



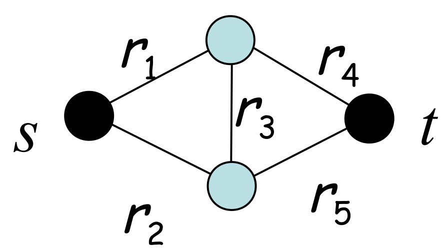
- nodes are perfect
- lines behave independently
- lines are up or down
- for each line i,
  $r_i = \Pr(\text{line } i \text{ is up})$

Associated key-words:

- **reliability diagrams, fault-trees…**
- **graph theory, coherent binary structure theory**

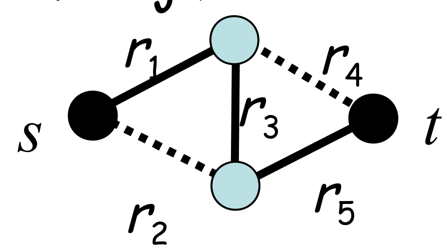
MATHEMATICAL MODEL

- V: the nodes
  K: the *terminals*, or target-set, $K \subseteq V$
  E: the *lines* or edges
  $\{r_i\}_{i \text{ in } E}$: the *elementary reliabilities*
- $N = (V, E)$: (the underlying) undirected graph
- Simple example: the "bridge", $K=\{s,t\}$



PROBABILITY STRUCTURE

- Ω: set of all partial sub-graphs of N (same nodes, part of the edges)
- $G = (V, F)$: a random graph on Ω; probabilistic structure:
  for any $H \subseteq E$
  $$\Pr(G = (V, H)) = \prod_{i \in H} r_i \prod_{j \notin H} (1 - r_j)$$



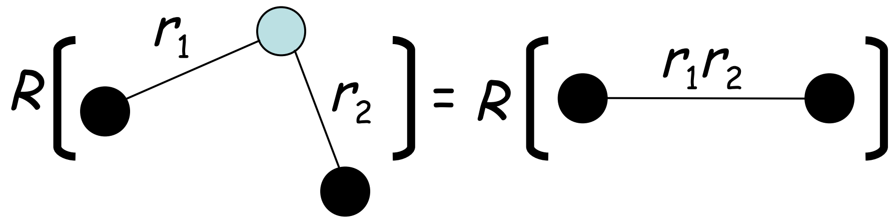
- Example: $H = \{1,3,5\}$,
  $\Pr(G = (V, H)) =$
  $r_1 r_3 r_5 (1 - r_2) (1 - r_4)$

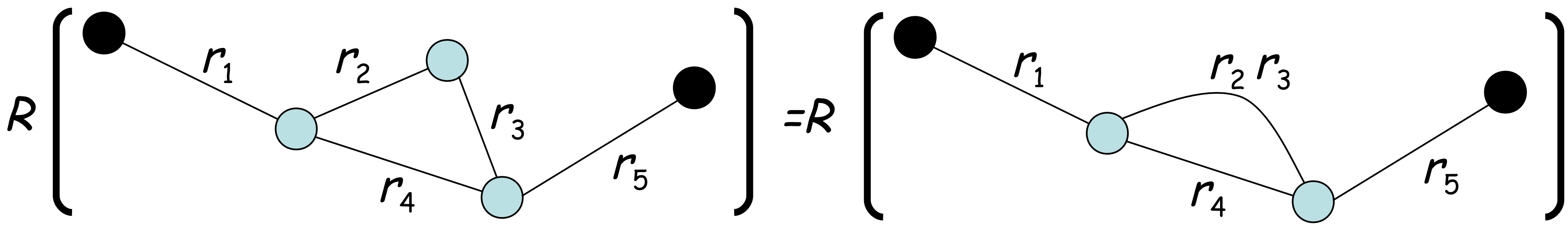
RELIABILITY METRIC

- Goal: R = K-network reliability,
  = Pr(the nodes in K are connected)
  (or equivalently $Q = 1 - R$)
- U: set of all partial sub-graphs of N where all nodes in K are connected; thus,
  $$R = \Pr(G \in U).$$

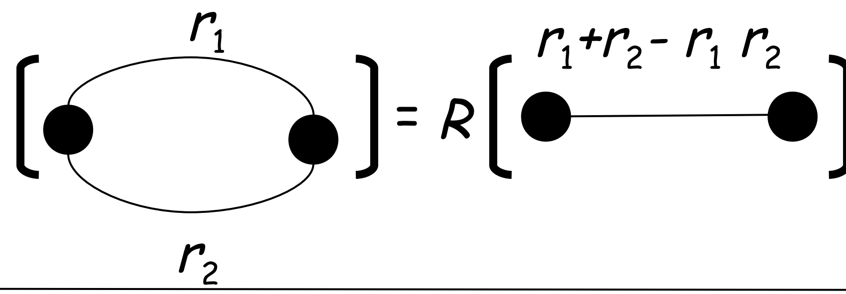
SERIES-PARALLEL REDUCTIONS

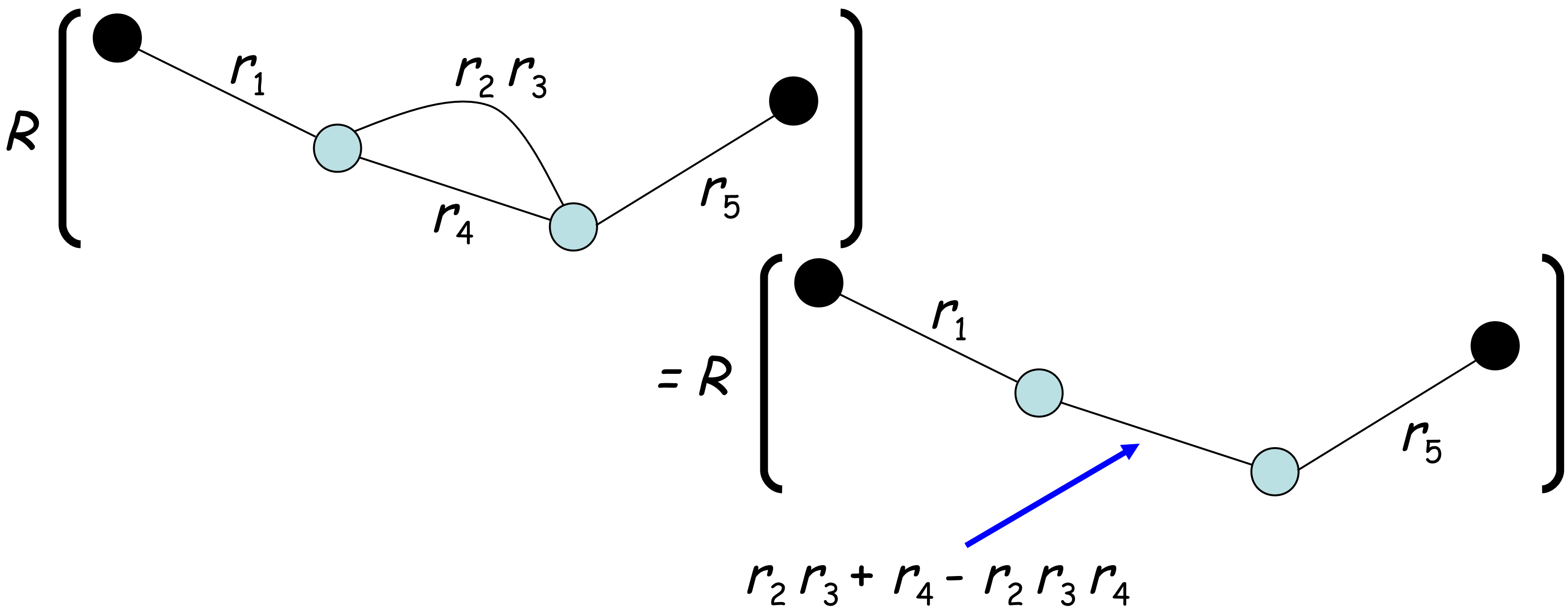
Series reduction:

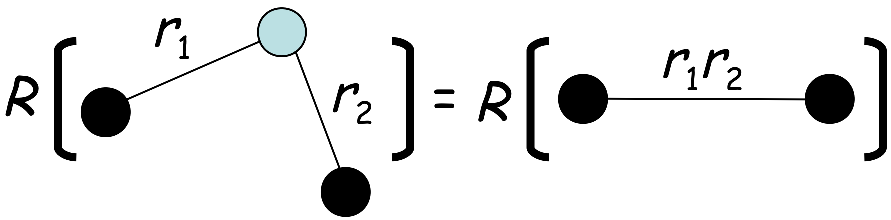


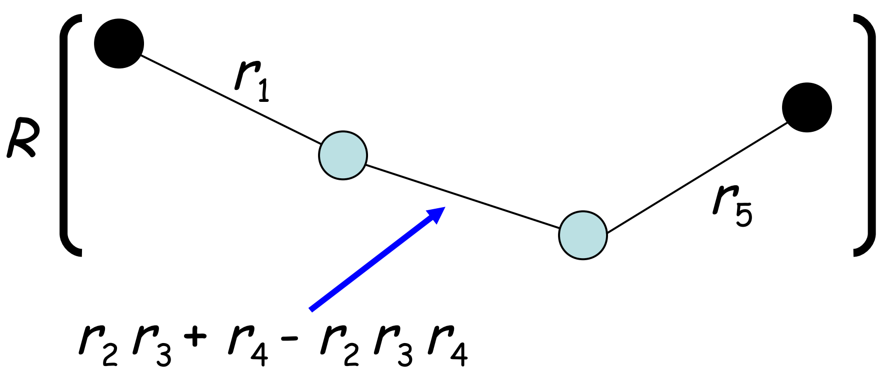


Parallel reduction: R





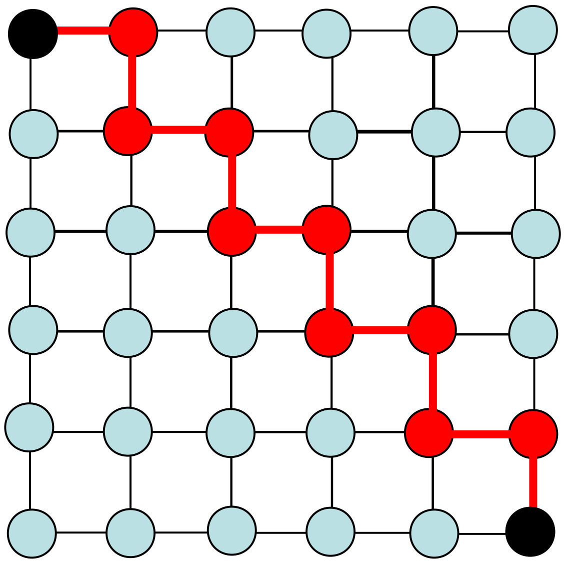




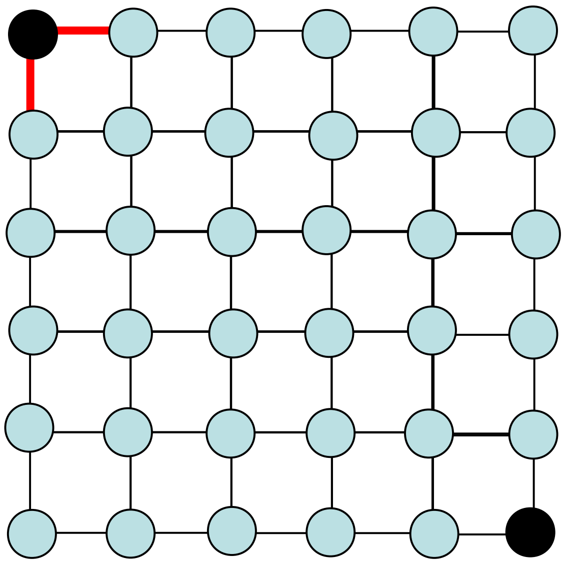
$= r_1(r_2 r_3 + r_4 - r_2 r_3 r_4)\ r_5$

Series-parallel reductions have polynomial cost, but they are not always applicable (f.e, the "bridge" can not be reduced).

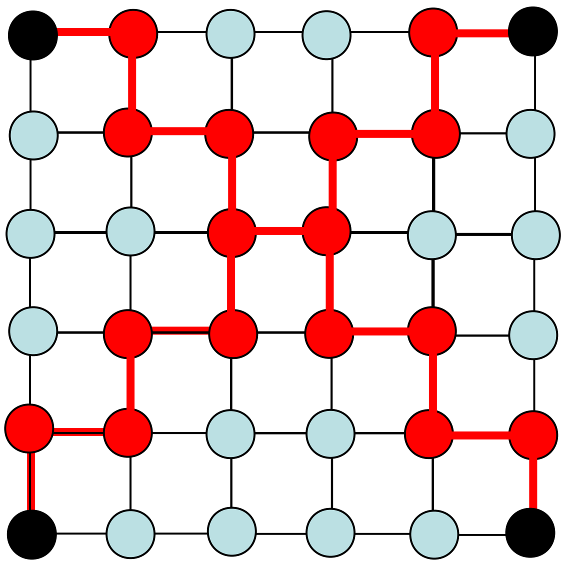
PATHSETS AND CUTSETS



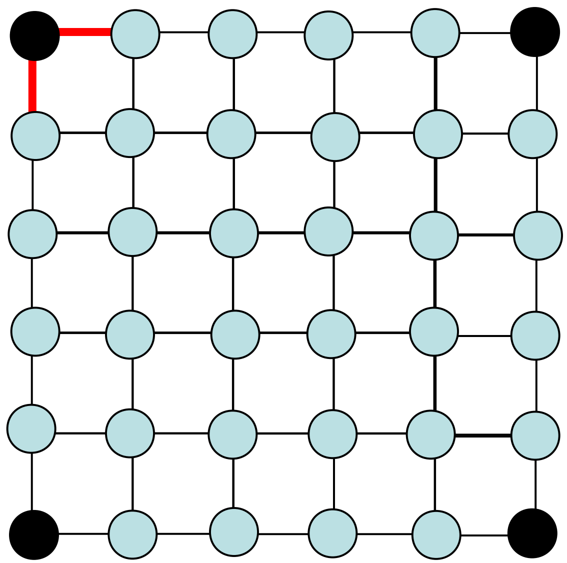
A minimal pathset, or *minpath* ($|K| = 2$)



A minimal cutset, or *mincut* (|*K*| = 2)



A minpath ($|K| = 4$)



A mincut ($|K| = 4$)

- Let P be a pathset.
- Let P-up denote the event
  P-up = "all links in P are up",
  $\Pr(P\text{-up}) = \prod_{\text{link } i \text{ is in } P} r_i$
- Since P-up $\Rightarrow$ system is up,
  $\Pr(P\text{-up}) \leq R$

- Let C be a cutset.
- Let C-down denote the event
  C-down = "all links in C are down",
  $\Pr(C\text{-down}) = \prod_{\text{link } i \text{ is in } C} (1 - r_i)$
- Since C-down $\Rightarrow$ system is down,
  $\Pr(C\text{-down}) \leq Q = 1 - R$

EXACT EVALUATION

- Computational complexity
  - General case : #P-hard.
  - For K={s,t} : #P-hard
  - For K=V: #P-hard
  - #P-hard for planar graphs.
  - Polynomial complexity for s-p reducible networks
  - Polynomial complexity for complete topologies and equireliable links, and K={s,t}.

OTHER ALTERNATIVES TO EXACT EVALUATION

- Upper and lower bounds
  - Can be used in place of exact results, for evaluating or designing a network.
  - Should be tight and computed efficiently
- Monte Carlo simulation.
  - Computational complexity grows moderately with network size (usually linearly or quadratically).
  - Compromise between precision and computation time.
  - Precision depends on value of edge reliabilities; problems when the reliabilities are very small (rare event situation).

AGENDA

- Classical network reliability model.
- Monte Carlo Simulation
- Variance Reduction methods
- Recursive Variance Reduction
- Splitting
- Conclusions

STANDARD MONTE CARLO SIMULATION

- #failed = 0
- **for** m = 1, 2, …, M
  - g = sample(G)
  - **if** $g \notin U$ **then** #failed += 1
- Q^{std} = #failed/M
- $V^{std} = Q^{std}(1 - Q^{std})/(M-1)$

- M – sample size
- Q^{std} – unreliability estimator
- V^{std} – estimator of the variance of Q^{std}

COMPUTATIONAL COMPLEXITY

- Internal loop: sampling a graph state (state of each edge), and verify if belongs or not to set U (DFS search); total complexity of order $O(|E|)$.
- M iterations; initialization time and final computations of $O(1)$.
- Total computation time $O(M|E|)$, linear in nb. of edges and nb. of replications.

PRECISION

- Relative error estimation:
  $RelErr = (V^{std})^{1/2} / Q^{std}$
  $= [(1 - Q^{std})\, Q^{std} / (M-1)]^{1/2} / Q^{std}$

  $= [(1 - Q^{std}) / ((M-1)\, Q^{std})]^{1/2}$
  $\approx 1/(MQ^{std})^{1/2}$
- When $Q \ll 1$, relative error grows, and if $Q \rightarrow 0$, $RelErr \rightarrow \infty$ (Rare Event problem).
- Error does not depend on the network size, but depends on edge reliability; if high reliabilities, a failure of the network has very low probability to be observed (rare event).

- It is possible to improve precision, taking more replications. To obtain a relative error relativo RelErr, we can compute M from the previous formula, obtaining $M \approx 1/(Q(\text{RelErr})^2)$.
- Total computation time of order $O(M\,|E|)$; when Q or RelErr very small, it will be prohibitively large $\Rightarrow$ motivation to develop variants improving the behavior of standard Monte Carlo (“Variance reduction methods”).

VARIANCE REDUCTION METHODS

- Generic methods, applicable to any simulation problem:
  - Importance sampling; cross-entropy.
  - Antithetic variates.
  - Control variates.
  - Stratified sampling.
- Specific for network reliability:
  - Employ structure and properties of the reliability problem to improve variance or computation time.
  - Many times adapt ideas from generic methods and from exact computation methods.

- Many ideas and methods in literature.
- A high level classification:
  - Based on bounds on the reliability (sampling in a subset of Ω , which lowers the variance). Van Slyke and Frank/ Kumamoto, Tanaka and Inoue / Fishman.
  - Based on antithetic sampling or generalizations (improve efficiency in generation of uniform variates and lowers the variance). Kumamoto, Tanaka and Inoue / Rubino and El Khadiri / Wei-Chang Yeh .
  - Based on partitioning state space Ω. , or on reformulating the problem in terms of other random variables with smaller variance. Karp and Luby / Jun and Ross / Cancela and El Khadiri.

- Based on graph evolution models (stochastic processes), with importance sampling to reduce variance. Wong and Easton / Elperin, Gertsbakh and Lomonosov.
- Cross-Entropy based variants to optimize the IS parameters. Hui, Bean, Kraetzl, and Kroese.
- Reformulations of the standard method to improve the computational efficiency. Rubino and El Khadiri.

ALTERNATIVES FOR METHOD EVALUATION

- Computational studies over test sets.
- Study of theoretical properties and asymptotic behavior.

COMPUTATIONAL STUDIES

- Over test sets, comprising different topologies and reliabilities.
- Problems:
  - Absence of standardized test library.
  - Unavailability of methods' implementations..
  - Literature results which only include variances, not running times.
  - Difficulties in normalizing runnng itmes over different computers.

- Performance measures for a method x:
  - Variance $Var(Q^x)$, for a fixed sample size M, or for a given time T.
  - Computation time $T(Q^m)$, for a fixed sample size M, or to obtain a predetermined precision.
  - Comparison against a reference method, standard Monte Carlo.

- How to compute the "speedup" of method x w.r.t. method y?
  - Fixed M, variance ratio $Var(Q^y) / Var(Q^x)$. Problem: does not take into account computing time per iteration.
  - More fair alternative:
    - Fix the precision
    - Run until obtaining this precision, compute
    - $T(Q^y) / T(Q^x)$.
  - Problem: if computing times differ by many orders of magnitude, unfeasible (example: if $T(Q^y) / T(Q^x)=10^9$, and $T(Q^x)=1$ sec, $T(Q^y)=31$ years).

- A better alternative
  - Relative efficiency (or speedup) $Var(Q^y)T(Q^y)/Var(Q^x)T(Q^x)$.
  - Interpretation: if $Var(Q^y)T(Q^y)/Var(Q^x)T(Q^x) = W$, then "method x is W times faster than method y " (i.e, it obtains the same precision with W times smaller effort).
  - Alternatively, for a given computational effort, method x obtains a variance W times smaller than method y.

THEORETICAL STUDIES

- Direct comparison of variance, or upper bound of variance, to the standard Monte Carlo one.
- Asymptotic properties:
  - Bounded relative error.
  - Bounded normal approximation
  - Bounded relative efficiency.

BOUNDED RELATIVE ERROR

- Framework:
  - ε rarity parameter
  - Link reliability : $r_i = 1 - a_i \varepsilon^{b_i}$
  - $Q(\varepsilon) \to 0$ when $\varepsilon \to 0$.
  - Relative error of method x: RelErr = $(V^x)^{1/2} / Q$
- Definition:
  x verifies "Bounded relative error" iff for every network G, and every fixed M, there is E such that $(V^x)^{1/2} / Q < E$ *when* $\varepsilon \to 0$.
- Interpretation: for a given topology and M, method x precision does not depend on ε.

VANISHING RELATIVE ERROR

- Framework:
  - ε rarity parameter
  - Link reliability : $r_i = 1 - a_i \varepsilon^{b_i}$
  - $Q(\varepsilon) \to 0$ when $\varepsilon \to 0$.
  - Relative error of method x: RelErr = $(V^x)^{1/2} / Q$
- Definition:
  x verifies "Vanishing relative error" iff for every network G, and every fixed M, there is E such that $(V^x)^{1/2} / Q \to 0$ *when* $\varepsilon \to 0$.
- Interpretation: for a given topology and M, method precision improves for rare event cases.

BOUNDED NORMAL APPROXIMTION

- Definition:
x verifies "Bounded normal approximation" iff for every G, the distance between distribution of Q^x and a normal distribution is bounded when $\varepsilon \to 0$ (this condition can be expressed using the third moment of Q^x and the variance, employing Berry-Essen theorem).
- Interpretation: employing Q^x and V^x it is possible to build an interval confidence based on the normal law, valid independently of ε.

BOUNDED RELATIVE EFFICIENCY

- Definition:
  x verifies "Bounded relative efficiency" iff for every network G, and fixed M, there exists E s.t. $Q^2 / (V^x T^x) > E$ when $\varepsilon \to 0$.
- Interpretation: given a topology, it is possible to obtain the same precision in the same computing time even when $\varepsilon \to 0$ (the method is robust w.r.t. "rare events")..

Bounded relative error and Vanishing relative error in Monte Carlo evaluation of static Network Reliability measures (part 2)

H. Cancela

Universidad de la República

SBPO 2010,Bentos Gon calves, September 2010

Outline

Outline

Crude Monte Carlo simulation

- Random state-vector of the network:

$$X = (X_1, \ldots, X_m)$$

where X_e Bernoulli r.v. $= 1$ if link e is working, 0 otherwise.
- Structure function Φ of $\{0,1\}^m$ into $\{0,1\}$ such that $\Phi(x) = 1$ if all nodes in $\mathcal{K}$ are connected when the stat $e-$vector is $x = (x_1, \ldots, x_m)$, and $\Phi(x) = 0$ otherwise.
- Searched reliability: $\mathbb{E}[\Phi(X)] = r = r(\mathcal{G})$ and unreliability $q = 1 - r = \mathbb{E}[1 - \Phi(X)]$.

Crude Monte Carlo simulation

- Random state-vector of the network:

$$X = (X_1, \ldots, X_m)$$

  where X_e Bernoulli r.v. $= 1$ if link e is working, 0 otherwise.
- Structure function Φ of $\{0,1\}^m$ into $\{0,1\}$ such that $\Phi(x) = 1$ if all nodes in $\mathcal{K}$ are connected when the stat $e-$vector is $x = (x_1, \ldots, x_m)$, and $\Phi(x) = 0$ otherwise.
- Searched reliability: $\mathbb{E}[\Phi(X)] = r = r(\mathcal{G})$ and unreliability $q = 1 - r = \mathbb{E}[1 - \Phi(X)]$.

- Consider n independent copies $X^{(i)} = (X_1^{(i)}, \ldots, X_m^{(i)})$ of X, and compute $Y^{(i)} = 1 - \Phi(X^{(i)})$.
- The crude estimator of q is then

$$\hat{Y}_n = \frac{1}{n} \sum_{i=1}^{n} Y^{(i)}.$$

- Confidence interval built from the central limit theorem.

Rarity modeling

- We assume that $q = \mathbb{E}[Y] \ll 1$.
- This can be due to the large number of paths connecting nodes in $\mathcal{K}$ or to large reliabilities of individual links.
- We assume $q_e \to 0 \ \forall e$, so that $q \to 0$.
- The *relative error* is proportional to

$$\frac{\sqrt{\mathrm{Var}[\hat{Y}_n]}}{\mathbb{E}[Y]} = \frac{\sqrt{q(1-q)}}{q\sqrt{n-1}} \to \infty$$

  as $q \to 0$.
- As a consequence, more and more paths are required to get a specified relative error as $q \to 0$.

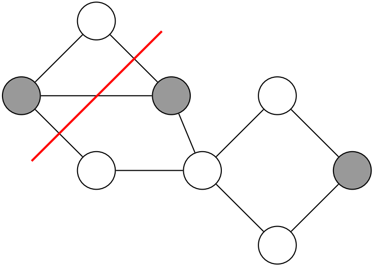
Definition

An estimator $\hat{Y}_n'$ is said to verify *Bounded Relative Error (BRE)* if $\frac{\sqrt{\mathrm{Var}[\hat{Y}_n']}}{\mathbb{E}[Y]}$ is bounded as $\mathbb{E}[Y] \to 0$. Equivalently, if $\frac{\sqrt{\mathbb{E}[(\hat{Y}_n')^2]}}{\mathbb{E}[Y]}$ is bounded as $\mathbb{E}[Y] \to 0$.

Outline

Recursive Variance Reduction (RVR)

- Principle: select a $\mathcal{K}$-cutset, i.e., a set $\mathcal{C}$ of links whose failure ensures the system failure.



- If all links in $\mathcal{C}$ are failed (probability $q_{\mathcal{C}}$), the system is failed. Clearly, $q_{\mathcal{C}} \leq q$.
- B_j= "the $j-1$ first links of $\mathcal{C}$ are down, but the j-th is up"
- $\mathbb{P}[B_j] = \left(\prod_{k=1}^{j-1} q_k\right) r_j$
- Define $p_j = \mathbb{P}[B_j \mid \text{at least one link is working}] = \mathbb{P}[B_j]/(1 - q_{\mathcal{C}})$

Recursive Variance Reduction (RVR)

The RVR estimator:

- Select a cut, and compute $q_{\mathcal{C}}$ and the p_js.
- Pick an edge at random in C according to the probability distribution $(p_j)_{j=1,\cdots,|\mathcal{C}|}$
- Let the chosen edge be the jth. Call $\mathcal{G}_j$ the graph obtained from $\mathcal{G}$ by deleting the first $j-1$ edges of $\mathcal{C}$ and by contracting the jth.
- The value y_{RVR} returned by the RVR estimator of $q(\mathcal{G})$, the unreliability of $\mathcal{G}$, is recursively defined as

$$y_{RVR}(\mathcal{G}) = q_{\mathcal{C}} + (1 - q_{\mathcal{C}}) y_{RVR}(\mathcal{G}_j).$$

RVR estimator

Formally, the RVR estimator of $Q(\mathcal{G})$ is the random variable

$$Y_{RVR} = q_{\mathcal{C}} + (1 - q_{\mathcal{C}}) \sum_{j=1}^{|\mathcal{C}|} \frac{\mathbf{1}_{B_j}}{1 - q_{\mathcal{C}}} Y_{RVR}(\mathcal{G}_j).$$

Theorem

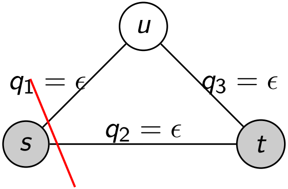
The estimator is unbiased: $\mathbb{E}[Y_{RVR}] = q(\mathcal{G}) = q$.
Second moment computed as

$$\begin{aligned}\mathbb{E}[Y_{RVR}^2] &= q_{\mathcal{C}}^2 + 2q_{\mathcal{C}}(1 - q_{\mathcal{C}}) \left(\sum_{j=1}^{|\mathcal{C}|} \frac{\mathbb{P}[B_j]}{1 - q_{\mathcal{C}}} \mathbb{E}[Y_{RVR}(\mathcal{G}_j)] \right) \\ &\quad + (1 - q_{\mathcal{C}})^2 \left(\sum_{j=1}^{|\mathcal{C}|} \frac{\mathbb{P}[B_j]}{1 - q_{\mathcal{C}}} \mathbb{E}[Y_{RVR}^2(\mathcal{G}_j)] \right).\end{aligned}$$

No Bounded Relative Error for RVR

Proposition

RVR algorithm does not verify Bounded Relative Error property.



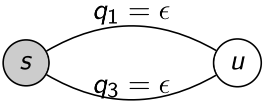
- Cut: the two links starting from node s and ordering them as first the link from s to t.
- $q_{\mathcal{C}} = \epsilon^2$.

$$\begin{aligned}\mathbb{E}[Y_{RVR}^2] &= \epsilon^4 + 2\epsilon^2\Big[(1-\epsilon)\mathbb{E}[Y_{RVR}(\mathcal{G}_1)] + \epsilon(1-\epsilon)\mathbb{E}[Y_{RVR}(\mathcal{G}_2)]\Big] \\ &+ (1-\epsilon^2)\Big[(1-\epsilon)\mathbb{E}[Y_{RVR}^2(\mathcal{G}_1)] + \epsilon(1-\epsilon)\mathbb{E}[Y_{RVR}^2(\mathcal{G}_2)]\Big].\end{aligned}$$

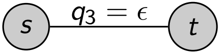
Counter-example for BRE (ctd)

where

- $\mathcal{G}_1$: link from s to t is working $\rightsquigarrow$ s and t are merged (the system is necessarily connected).

  $Y_{RVR}(\mathcal{G}_1) = 0$. Thus $\mathbb{E}[Y_{RVR}(\mathcal{G}_1)] = \mathbb{E}[Y^2_{RVR}(\mathcal{G}_1)] = 0$.
- $\mathcal{G}_2$: link from s to t failed, but the one from s to u is working $\rightsquigarrow$ s and u are merged.

  $\mathbb{E}[Y_{RVR}(\mathcal{G}_2)] = \epsilon$, $\mathbb{E}[Y^2_{RVR}(\mathcal{G}_2)] = \epsilon^2$.
- Finally, $\mathbb{E}[Y^2_{RVR}] = \Theta(\epsilon^3)$, and $\mathbb{E}[Y^2_{RVR}]/(\mathbb{E}[Y_{RVR}])^2 = \Theta(\epsilon^{-1}) \to \infty$ as $\epsilon \to 0$.

We may have BRE depending on the ordering of the links.

Outline

Balanced RVR

- Non-BRE comes from the crude distribution for sampling the first working link on the cut.
- *Importance Sampling (IS)* used instead; that is, the sampling of the first line up in the cut is not anymore (p_j).
- So far, we built a partition by assigning to the events B_j, for $1 \leq j \leq |\mathcal{C}|$, the conditional probabilities

$$p_j = \mathbb{P}[B_j \mid A],$$

where A is the event "at least one link in cut $\mathcal{C}$ is up".
- Let us write the RVR estimator as

$$Y_{RVR} = q_{\mathcal{C}} + (1 - q_{\mathcal{C}}) \sum_{j=1}^{|\mathcal{C}|} \mathbf{1}_{B'_j} Y_{RVR}(\mathcal{G}_j),$$

where B'_j represents the same event as B_j but has the (conditional) probability p_j.

Balanced RVR

- Now, we change this probability p_j by the uniform distribution on $\{1, 2, \cdots, |\mathcal{C}|\}$, $\tilde{p}_j = 1/|\mathcal{C}|$, for sampling B'_j.
- Let us call Y_{BRVR} the corresponding estimator, but using this uniform distribution, we write

$$\begin{aligned} Y_{BRVR} &= q_{\mathcal{C}} + (1 - q_{\mathcal{C}}) \sum_{j=1}^{|\mathcal{C}|} \mathbf{1}_{B'_j} \frac{p_j}{\tilde{p}_j} Y_{BRVR}(\mathcal{G}_j) \\ &= q_{\mathcal{C}} + |\mathcal{C}| \sum_{j=1}^{|\mathcal{C}|} \mathbf{1}_{B'_j} \mathbb{P}[B_j] Y_{BRVR}(\mathcal{G}_j). \end{aligned}$$

- Estimator: using likelihood ratio $p_j/\tilde{p}_j$ to keep it unbiased.

Results on Balanced RVR

Theorem

The estimator Y_{BRVR} is unbiased: $\mathbb{E}[Y_{BRVR}] = q$.
BRVR algorithm verifies Bounded Relative Error property.

Proof by induction from

$$\begin{aligned}\mathbb{E}[Y_{BRVR}^2] &= q_{\mathcal{C}}^2 + 2q_{\mathcal{C}}|\mathcal{C}|\left(\sum_{j=1}^{|\mathcal{C}|}\mathbb{P}[B_j]\mathbb{E}[Y_{BRVR}(\mathcal{G}_j)]\right)\\ &\quad +|\mathcal{C}|^2\left(\sum_{j=1}^{|\mathcal{C}|}(\mathbb{P}[B_j])^2\mathbb{E}[Y_{BRVR}^2(\mathcal{G}_j)]\right).\end{aligned}$$

Outline

Zero-variance Approximation RVR

- *Zero-variance change of measure*: chooses the appropriate (ideally the best) IS for the first working link on the cut:
- choose B'_j with probability $\tilde{p}_j$ in the IS estimator, with

$$\tilde{p}_j = \frac{\mathbb{P}[B_j] q(\mathcal{G}_j)}{\sum_{k=1}^{|\mathcal{C}|} \mathbb{P}[B_k] q(\mathcal{G}_k)} \tag{1}$$

- Resulting estimator:

$$Y_{ZRVR} = q_{\mathcal{C}} + \left(\sum_{k=1}^{|\mathcal{C}|} \mathbb{P}[B_k] q(\mathcal{G}_k) \right) \sum_{j=1}^{|\mathcal{C}|} \mathbf{1}_{B'_j(\mathcal{G})} \frac{1}{q(\mathcal{G}_j)} Y_{ZRVR}(\mathcal{G}_j).$$

Theorem

Y_{ZRVR} has variance $\mathrm{Var}[Y_{ZRVR}] = 0$.

- Implementing it requires the knowledge of the $q(\mathcal{G}_i)$, but in that case, no need to simulate!

Zero Variance Approximation

- Instead, use some approximation $\hat{q}(\mathcal{G}_i)$ of $q(\mathcal{G}_i)$ plugged into (1).

$$Y_{AZRVR} = q_{\mathcal{C}} + \left(\sum_{k=1}^{|\mathcal{C}|} \mathbb{P}[B_k]\hat{q}(\mathcal{G}_k)\right) \sum_{j=1}^{|\mathcal{C}|} \mathbf{1}_{B'_j(\mathcal{G})} \frac{1}{\hat{q}(\mathcal{G}_j)} Y_{AZRVR}(\mathcal{G}_j).$$

Proposition

If $\forall 1 \leq j \leq |\mathcal{C}|$, $\hat{q}(\mathcal{G}_j) = \Theta(q(\mathcal{G}_j))$ *as* $\epsilon \to 0$, Y_{AZRVR} *verifies BRE property.*

Zero Variance Approximation

- Define the *mincut-maxprob* approximation $\hat{q}(\mathcal{G})$ of $q(\mathcal{G})$ as maximal probability of a mincut of graph $\mathcal{G}$ (computed in polynomial time).

Proposition

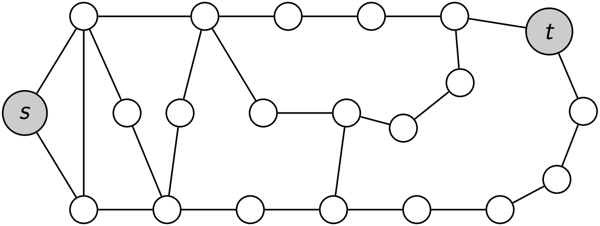
With the mincut-maxprob approximation, $\hat{q}(\mathcal{G}_j) = \Theta(q(\mathcal{G}_j))$ *as* $\epsilon \to 0$, *therefore BRE property is obtained.*

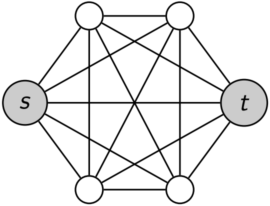
Proposition

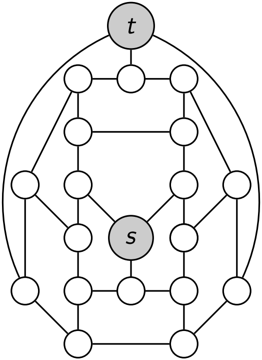
If, $\hat{q}(\mathcal{G}_j) = q(\mathcal{G}_j) + o(q(\mathcal{G}_j))$ *as* $\epsilon \to 0$ *for all* $1 \leq j \leq |\mathcal{C}|$, *the Vanishing relative (VRE) property (the RE tends to 0, stronger than just being bounded) is verified.*

Outline

Three topologies: arpanet, C6, dodecahedron







Network (q_e)	$Q(G)$	$N \times Var(SMC)$	$N \times Var(RVR)$	$N \times Var(Bal)$	$N \times Var(AZV)$
Arp (5.00 $e-01$)	9.63989 e-01	3.47133 e-02	3.71795 e-03	1.60608 e-01	1.69321 e-01
Arp(3.00 e-01)	6.81507 e-01	2.17055 e-01	4.74801 e-02	5.65742 e-01	8.45549 e-01
Arp (1.00 e-01)	9.54229 e-02	8.63174 e-02	1.46865 e-02	3.68529 e-02	9.55806 e-02
Arp (1.00 e-02)	6.54074 e-04	6.53646 e-04	1.63753 e-05	6.71095 e-07	3.06912 e-06
Arp (1.00 e-03)	6.05581 e-06	6.05577 e-06	1.60407 e-08	5.64473 e-11	3.43246 e-11
Arp (1.00 e-04)	6.00560 e-08	6.00560 e-08	1.60041 e-11	5.69261 e-15	3.47090 e-16
Arp (1.00 e-05)	6.00056 e-10	6.00056 e-10	1.60004 e-14	5.69924 e-19	3.47477 e-21
Arp (1.00 e-06)	6.00006 e-12	6.00006 e-12	1.60000 e-17	5.69992 e-23	3.47512 e-26
C6 (5.00 e-01)	7.64160 e-02	7.05766 e-02	7.72612 e-05	6.87599 e-4	7.27858 e-05
C6 (3.00 e-01)	5.26728 e-03	5.23953 e-03	2.56429 e-07	7.86630 e-06	2.27577 e-07
C6 (1.00 e-01)	2.00766 e-05	2.00762 e-05	1.28070 e-13	2.28489 e-10	1.17223 e-13
C6 (1.00 e-02)	2.00001 e-10	2.00001 e-10	1.01244 e-26	2.92080 e-20	1.00225 e-26
C6 (1.00 e-03)	2.00000 e-15	2.00000 e-15	1.00102 e-39	2.99201 e-30	1.00002 e-039
C6 (1.00 e-04)	2.00000 e-20	2.00000 e-20	1.00000 e-52	2.99920 e-40	1.00000 e-52
C6 (1.00 e-05)	2.00000 e-25	2.00000 e-25	1.42434 e-65	2.99992 e-50	1.42434 e-65
C6 (1.00 e-06)	1.99998 e-30	1.99998 e-30	num. pblm.	2.99986 e-60	num. pblm.
Dod (5.00 e-01)	7.09745 e-01	2.06007 e-01	1.57246 e-02	4.23225 e-01	1.34634 e-01
Dod (3.00 e-01)	1.68518 e-01	1.40120 e-01	9.22721 e-03	1.05285 e-01	1.68222 e-02
Dod (1.00 e-01)	2.87960 e-03	2.87131 e-03	5.80985 e-06	7.53573 e-06	6.32871 e-07
Dod (1.00 e-02)	2.06189 e-06	2.06189 e-06	2.17456 e-12	2.06824 e-12	1.12133 e-14
Dod (1.00 e-03)	2.00602 e-09	2.00602 e-09	2.01614 e-18	2.00608 e-18	1.01110 e-21
Dod (1.00 e-04)	2.00060 e-12	2.00060 e-12	2.00160 e-24	2.00060 e-24	1.00110 e-28
Dod (1.00 e-05)	2.00006 e-15	2.00006 e-15	2.00016 e-30	2.00006 e-30	1.00011 e-35
Dod (1.00 e-06)	2.00001 e-18	2.00001 e-18	2.00002 e-36	2.00001 e-36	1.00001 e-42

Network (q_e)	$\sqrt{N}$ $RE(RVR)$ $\times$	$\frac{RE(SMC)}{RE(RVR)}$	$\sqrt{N}$ $RE(Bal)$ $\times$	$\frac{RE(SMC)}{RE(Bal)}$	$\sqrt{N}$ $RE(AZV)$ $\times$	$\frac{RE(SMC)}{RE(AZV)}$
Arp (5.00 e-01)	1.69 e+00	3.06 e+00	1.11 e+01	4.65 e-01	1.14 e+01	4.53 e-01
Arp (3.00 e-01)	6.84 e-01	2.14 e+00	2.36 e+00	6.19 e-01	2.89 e+00	5.07 e-01
Arp (1.00 e-01)	1.27 e+00	2.42 e+00	2.01 e+00	1.53 e+00	3.24 e+00	9.50 e-01
Arp (1.00 e-02)	6.19 e+00	6.32 e+00	1.25 e+00	3.12 e+01	2.68 e+00	1.46 e+01
Arp (1.00 e-03)	2.09 e+01	1.94 e+01	1.24 e+00	3.28 e+02	9.67 e-01	4.20 e+02
Arp (1.00 e-04)	6.66 e+01	6.13 e+01	1.26 e+00	3.25 e+03	3.10 e-01	1.32 e+04
Arp (1.00 e-05)	2.11 e+02	1.94 e+02	1.26 e+00	3.24 e+04	9.82 e-02	4.16 e+05
Arp (1.00 e-06)	6.67 e+02	6.12 e+02	1.26 e+00	3.24 e+05	3.11 e-02	1.31 e+07
C6 (5.00 e-01)	1.15 e-01	3.02 e+01	3.43 e-01	1.01 e+01	1.12 e-01	3.11 e+01
C6 (3.00 e-01)	9.61 e-02	1.43 e+02	5.32 e-01	2.58 e+01	9.06 e-02	1.52 e+02
C6 (1.00 e-01)	1.78 e-02	1.25 e+04	7.53 e-01	2.96 e+02	1.71 e-02	1.31 e+04
C6 (1.00 e-02)	5.03 e-04	1.41 e+08	8.55 e-01	8.27 e+04	5.01 e-04	1.41 e+08
C6 (1.00 e-03)	1.58 e-05	1.41 e+12	8.65 e-01	2.59 e+07	1.58 e-05	1.41 e+12
C6 (1.00 e-04)	5.00 e-07	1.41 e+16	8.66 e-01	8.17 e+09	5.00 e-07	1.41 e+16
C6 (1.00 e-05)	1.89 e-08	1.18 e+20	8.66 e-01	2.58 e+12	1.89 e-08	1.18 e+20
C6 (1.00 e-06)	num. pblm.	num. pblm.	8.66 e-01	8.17 e+14	num. pblm.	num. pblm.
Dod (5.00 e-01)	4.32 e-01	3.62 e+00	2.24 e+00	6.98 e-01	1.26 e+00	1.24 e+00
Dod (3.00 e-01)	5.70 e-01	3.90 e+00	1.93 e+00	1.15 e+00	7.70 e-01	2.89 e+00
Dod (1.00 e-01)	8.37 e-01	2.22 e+01	9.53 e-01	1.95 e+01	2.76 e-01	6.74 e+01
Dod (1.00 e-02)	7.15 e-01	9.74 e+02	6.97 e-01	9.98 e+02	5.14 e-02	1.36 e+04
Dod (1.00 e-03)	7.08 e-01	3.15 e+04	7.06 e-01	3.16 e+04	1.59 e-02	1.41 e+06
Dod (1.00 e-04)	7.07 e-01	1.00 e+06	7.07 e-01	1.00 e+06	5.00 e-03	1.41 e+08
Dod (1.00 e-05)	7.07 e-01	3.16 e+07	7.07 e-01	3.16 e+07	1.58 e-03	1.41 e+10
Dod (1.00 e-06)	7.07 e-01	1.00 e+09	7.07 e-01	1.00 e+09	5.00 e-04	1.41 e+12

Outline

Conclusions

We have

- Standard Monte Carlo method is easy to implement, but has limitations for highly reliable networks, or when a small relative error is needed.
- To improve its efficiency, two main paths:
  - Reduce the variance per iteration
  - Reduce the computing time per iteration.
- Theoretical results establishing desirable properties for the behavior of variance reduction methods; BRE, VRE, etc.
- Only in some cases it has been possible to verify these properties.

Conclusions

We have

- RVR does not always verify BRE;
- RVR balanced version verifies BRE;
- Zero-variance IS approximation verifies BRE, and even VRE;
- Computational results illustrate the gain that can be obtained.