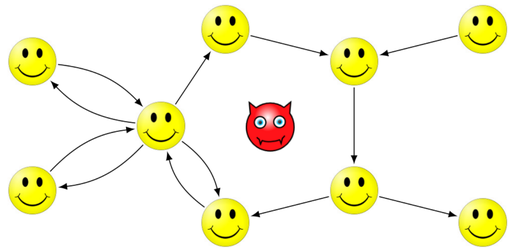
The Adversarial Noise Threshold for Distributed Protocols

William M. Hoza and Leonard J. Schulman
California Institute of Technology

The model

- n parties (represented by vertices in directed graph)
- m communication channels (edges in directed graph)
- One adversary, who can observe and modify all transmissions
- In every *round*, each party transmits one bit on each outgoing edge (synchronously)
- But the adversary flips bits mid-flight as she sees fit, subject to an *error budget*



Question: How much error can be tolerated?

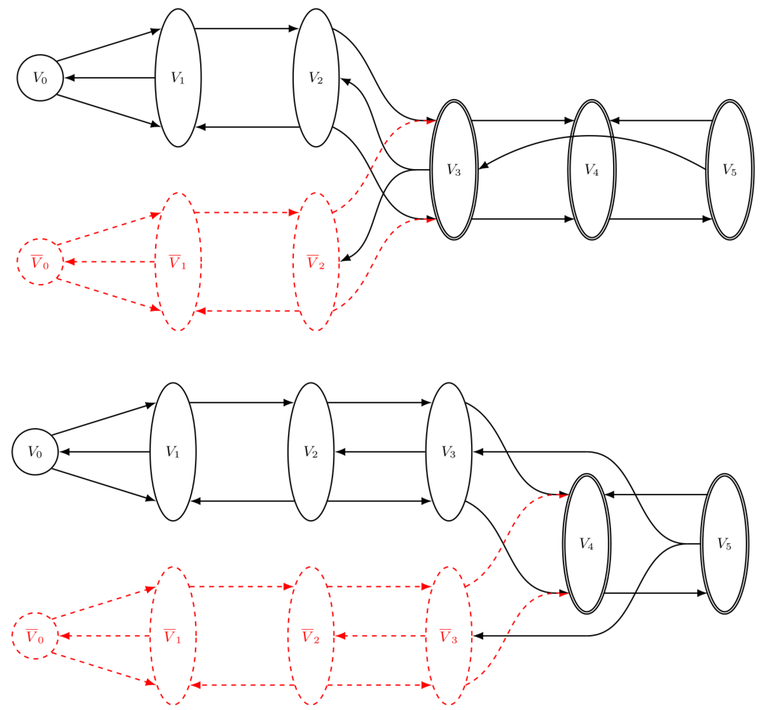
- *Protocol* π: collection of algorithms the parties use to decide which bits to send
- *Compiler* C: transforms a protocol π into a *simulation protocol* $C(\pi)$, to deal with noise
- Goal: After executing $C(\pi)$ in the presence of any adversary with sufficiently small error budget, parties give same outputs as if they had executed π on noiseless network

Global error budget

- We allow the adversary to corrupt some fraction of *all* transmissions
- **Theorem**: There is a compiler for protocols on *undirected* networks such that:
  - Simulation protocol runs on *subnetwork* (some edges removed)
  - Tolerates error rate $\Omega\left(\frac{1}{n}\right)$
  - Number of rounds of communication blows up by a factor of $\mathcal{O}\left(\frac{m \log n}{n}\right)$
- The error rate is within a constant factor of optimal
- The round complexity is within a factor of $\mathcal{O}(k \log n)$ of optimal, where $k =$ the edge connectivity of original network
- For general directed graphs: **Theorem**: The optimal tolerable error rate is $\Theta\left(\frac{1}{s}\right)$, where s is the minimum number of edges in any subgraph with the same reachability relation
- Proof ideas for positive results:
  - Compiler from (Rajagopalan & Schulman '94) was designed for stochastic noise, but can be adapted to tolerate adversarial error rate $\Omega\left(\frac{1}{m}\right)$
  - Precompose with "sparsifying compiler" which simulates original protocol on sparsest possible subnetwork. Immediately tolerates optimal error rate
  - For undirected case, use a *cut sparsifier*. Can be chosen so that all cut-sets are reduced by factor of $\mathcal{O}\left(\frac{m}{n}\right)$ (adapted from de Carli Silva, Harvey, Sato '11)
  - By approximate max-flow min-cut results for *multicommodity flow*, obtain paths for bits to travel on with congestion $\mathcal{O}\left(\frac{m \log n}{n}\right)$ (adapted from Linial, London, Rabinovich '95)
  - Restore a few ($\mathcal{O}(n)$) edges so that paths can have length $\mathcal{O}\left(\frac{m \log n}{n}\right)$ with the same congestion
  - Can schedule transmissions so bits arrive in $\mathcal{O}($length + congestion$)$ rounds (Leighton, Maggs, Rao '94)

Per-edge error budget

- Restricted adversary: separate budget of errors for each edge
- **Theorem**: The optimal tolerable error rate is between $\frac{1}{4D}$ and $\frac{1}{2D}$, where D is the maximum finite directed distance between any two parties
- Proof idea for negative result:
  - Say the shortest path from P_0 to P_D has length D
  - The protocol requires P_0 to send a message to P_D along this path
  - The adversary simulates an "alternate reality," where P_0 is trying to send a different message
  - V_i is the set of parties at distance i from P_0
  - Black indicates actual parties and channels controlled by them. Red indicates imaginary parties are channels controlled by them. Double borders indicate parties who don't know which "possible world" is the real world
  - In the first figure, the adversary is attacking all channels from V_2 to V_3
  - In the second figure, later in the simulation, the adversary attacks all channels from V_3 to V_4
  - As time progresses, more and more parties figure out which of the two "possible worlds" is the real world. But P_D never figures it out



- Compiler used to prove positive result has exponential round complexity blowup (involves each party sending the list of all messages she would ever send in any circumstance, all encoded in a good error correcting code)
- **Theorem**: The optimal tolerable error rate for *polynomial-query black-box compilers* is $\Theta\left(\frac{1}{R}\right)$, where R is the maximum number of distinct vertices visited in any directed walk through the network

Acknowledgements

Thanks to Nellie Bergen and Adrian Foster Tillotson, the ARCS Los Angeles Founder Chapter, and the Caltech SURF office!