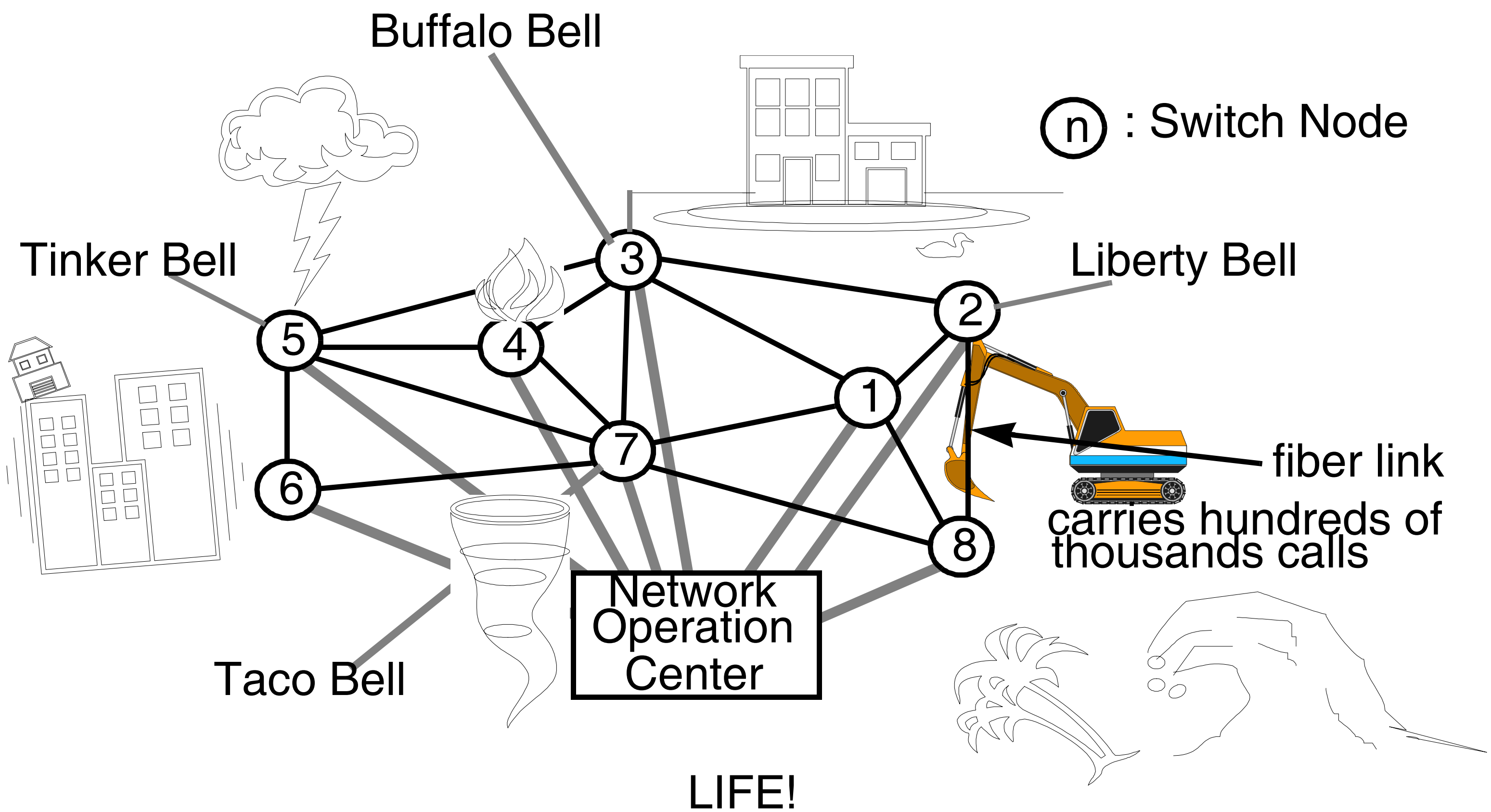
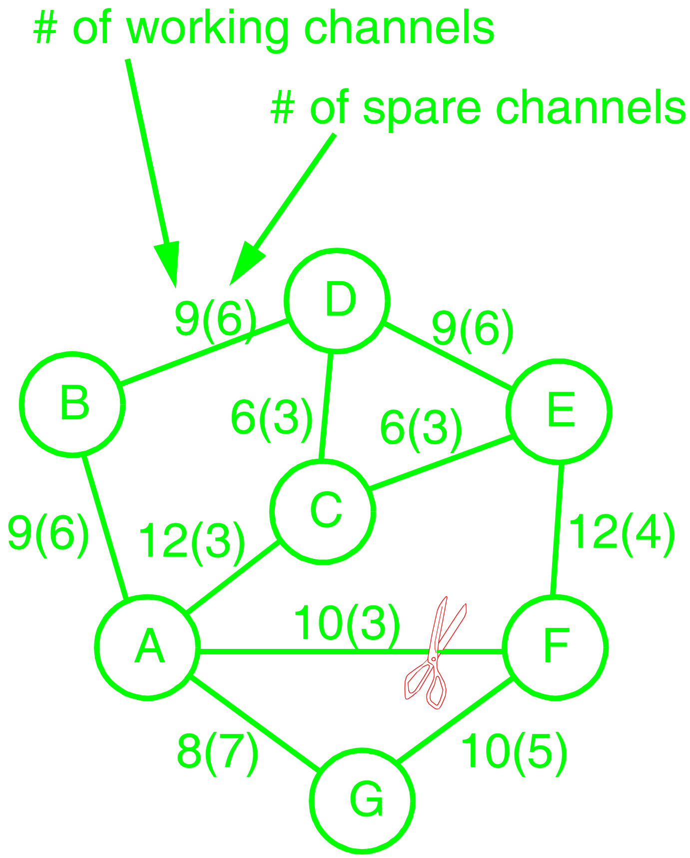
What is the Network Restoration Problem?



- Types of Failures: Channel, Link, Node, Area, and Transient Failures.
- Network Restoration: Process of recovering from network failures.
- Types of Restoration Approaches: Centralized, Distributed, Hierarchical.

Network Model and Performance Metrics

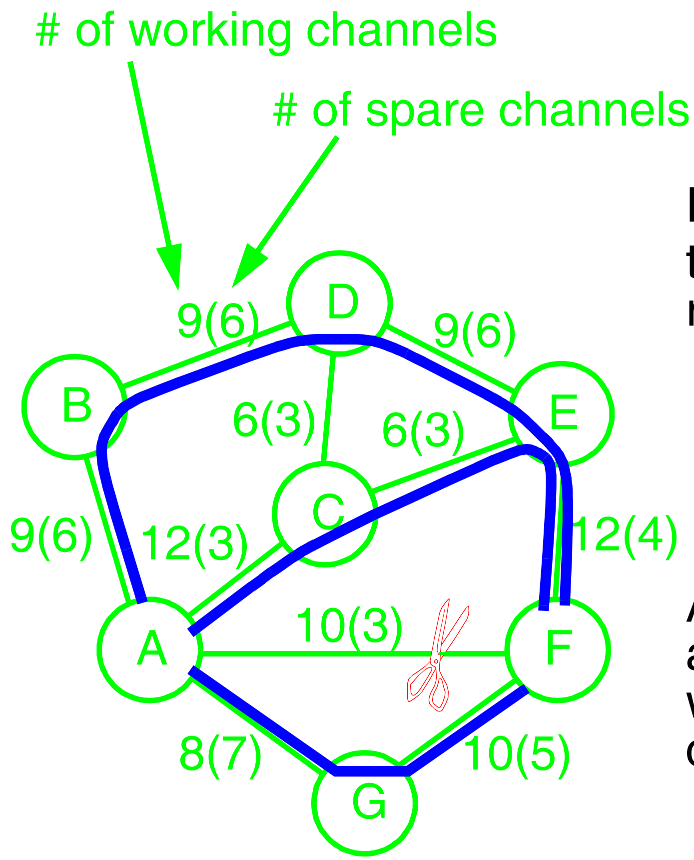


Performance metrics for evaluating network restoration algorithms:

- Time to restoration
- Restoration level
- Spare channel usage
  For path-based approach, this is defined to be the no.of spares used in the restoration paths minus the no. of working channels in the disrupted paths that were released
- Message Volume

Transmission delay⟵Message length/Transmission speed
Propagation delay⟵Node distance/Signal propagation speed
Queueing delay⟵Messages in front*Msg processing time
DCS connection delay⟵DCS connection time*# of channels to be connected

Restoration Paths and Spare Channel Usage



For Link A-F failure,

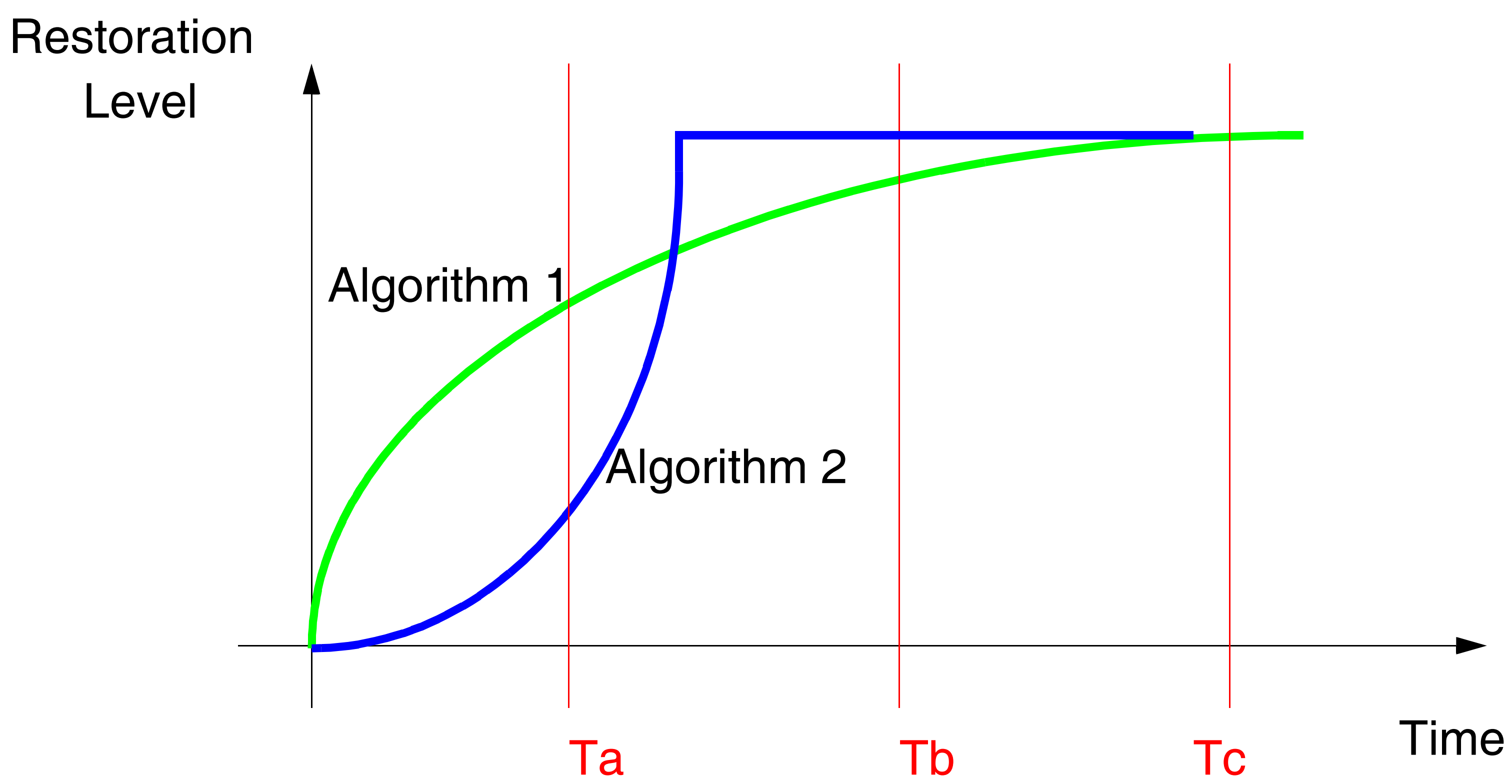
three restoration paths can be found to restore 9 of the 10 disrupted channels.

- Restoration Level-90%
- Spare channel usage-23 spares used

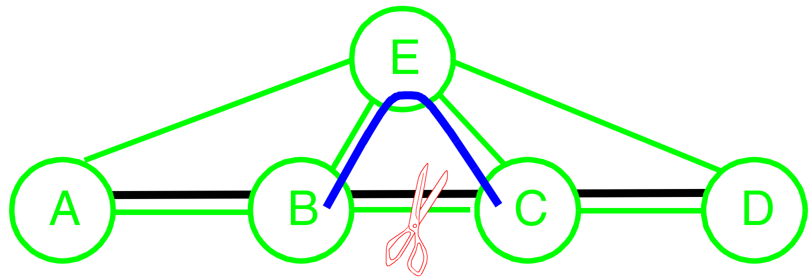
An optimal algorithm based on RELAX-IIIT algorithm (courtesy of Prof. Bersekas of MIT) was implemented to obtain the optimal spare channel usage for a given link failure.

Restoration Level and Time to Restore

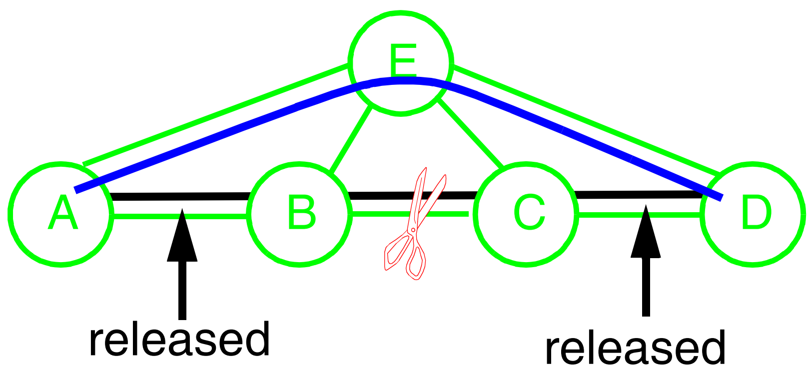
The selection of network restoration algorithms depends on the restoration curves of the algorithms, the real-time requirements, and the existence of priority traffic.



Link Based vs. Path Based Network Restoration Approaches

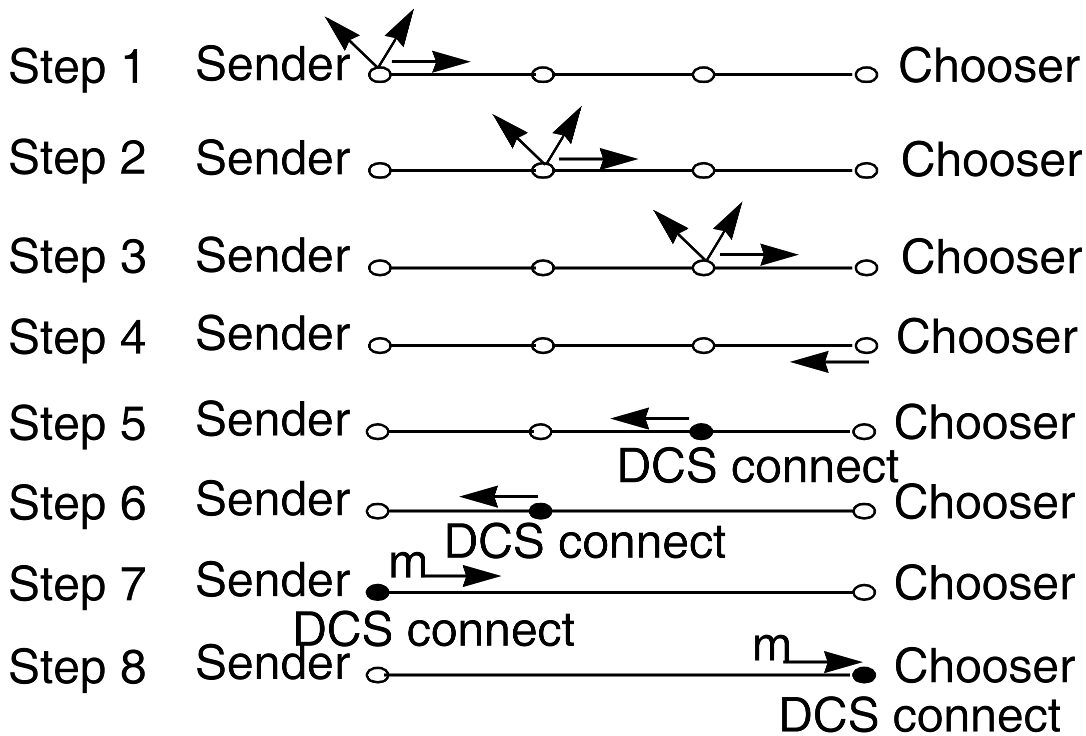


Link based restoration approaches try to find paths around the disrupted areas and keep other working channels intact.



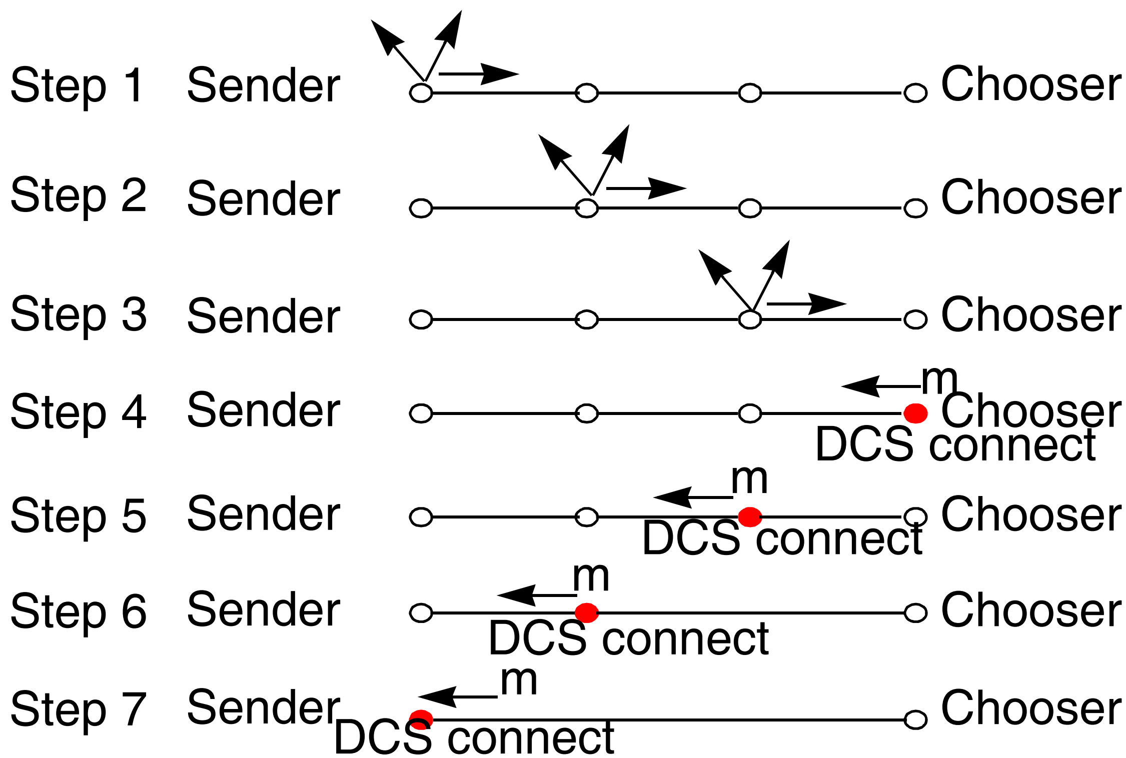
Path based restoration approaches release the working channels of the disrupted paths and have original sources re-establish connections.

GroverÕs Self Healing Network [Grover87,89]



- Use Sender Chooser Paradigm.
- Restore disrupted connects on channel by channel basis, one msg per spare
- Perform good on small networks or low spare networks.

BellcoreÕs FITNESS [Yang88]

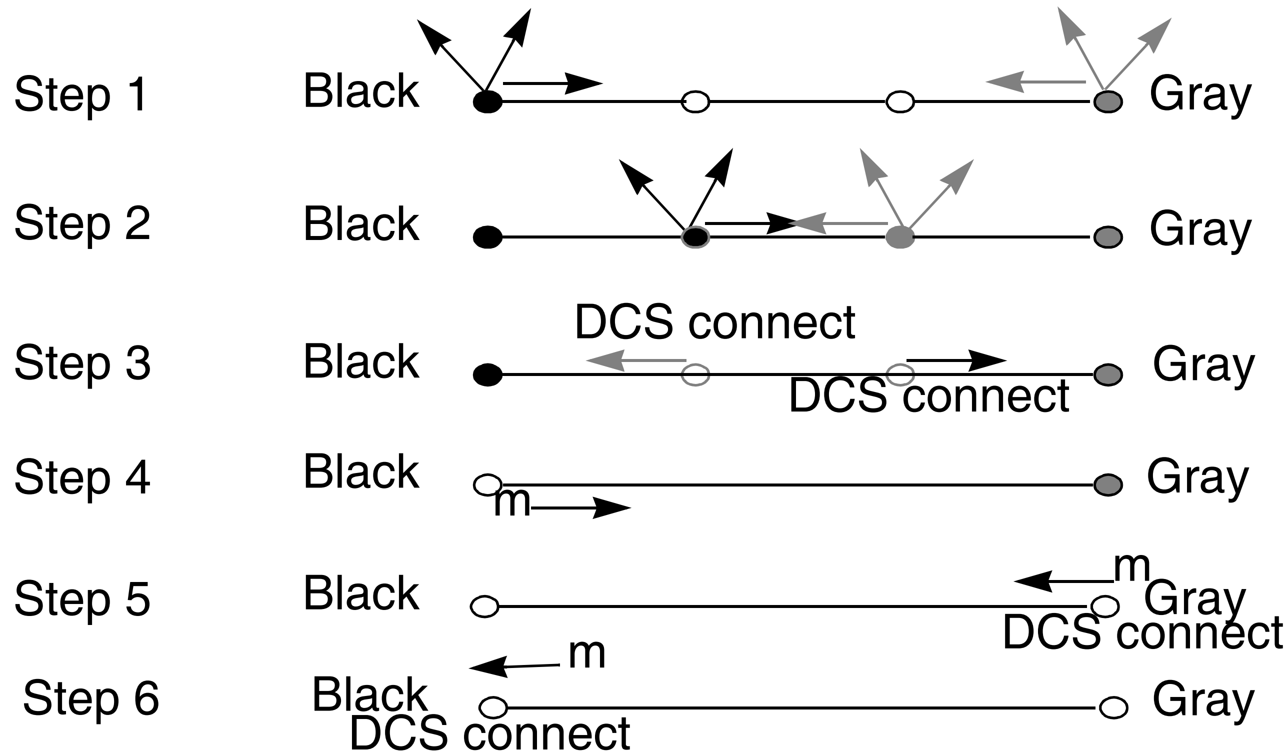


- Use Sender-Chooser Paradigm
- Request Bandwidth on aggregated basis, one request on one outgoing link.
- Use Time-out and wave mechanisms to choose largest bandwidth in a wave.

UCCSÕ RREACT[Chow93a]

- Use Sender-Chooser Control Paradigm
- Use aggregate request similar to FITNESS.
- Attach traverse path info in the request messages.
- Request messages explore all possible paths between Sender and Chooser.
- Chooser builds ÒsmartÓ current network topology based on these path info.
- Chooser allocate bandwidth on FIFS basis.
- Consistently find paths with low spare usage.
- Very reliable.
- Drawback: Message volume increases exponentially with network size.

UCCSÕ Two Prong [Chow93b]



- Two disrupted end nodes broadcast request simultaneously.
- Tandem nodes on receiving msgs from both ends start connections.
- Request messages forward (not broadcast) to other nodes after mid-way
- Fast concurrent connections and low message volume.

Path-based Two Prong

- In this CASI project, we designed a path-based Two Prong algorithm based on link-based Two Prong.
- The network model needs to include path information.
- The protocol handles the release of working capacity in the disrupted paths.
- The protocol resolves the spare contention problem among requests of different disrupted paths. Use priority scheme based on Path ID.
- Retry and backtrack mechanisms implemented to increase
- It can handle both link and node failure cases.
- Message volume increases dramatically (2~8 times) compared with link-based approaches.
- Restoration levels are close to those achieved by link-based approaches.
- Restoration is slower (2~10 times on NJ test network) compared with link-based Two Prong.

Path-based Two Prong: Detecting/Flooding Phase

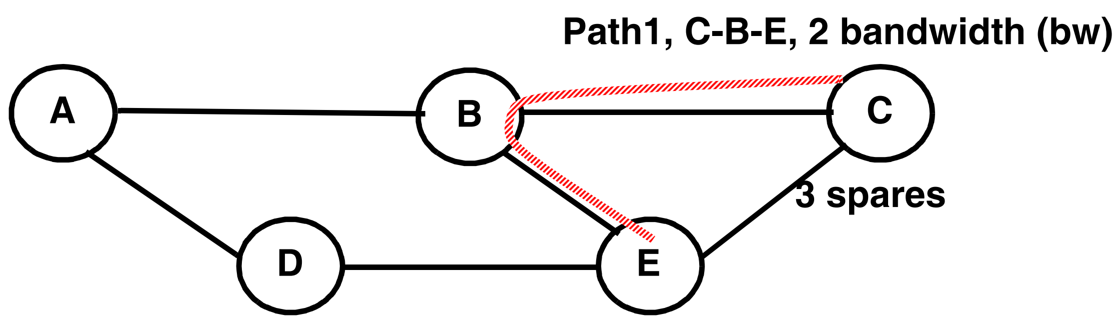


Figure 3.1a. Prior to Node B failure.

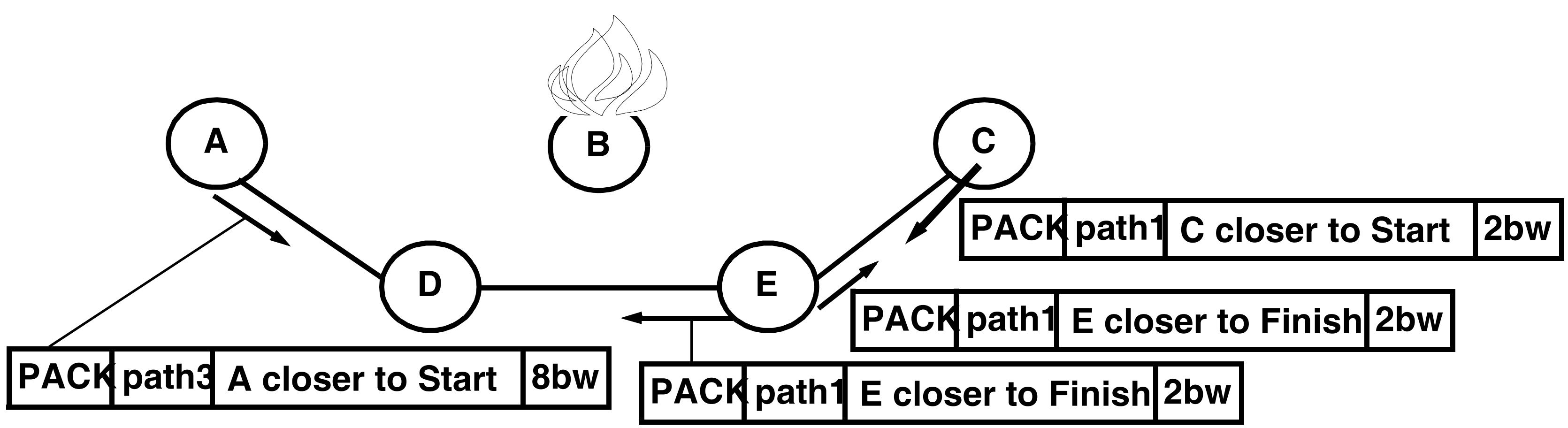


Figure 3.1b. Node B fails.

Path-base Two-Prong: Request/Carving Phase

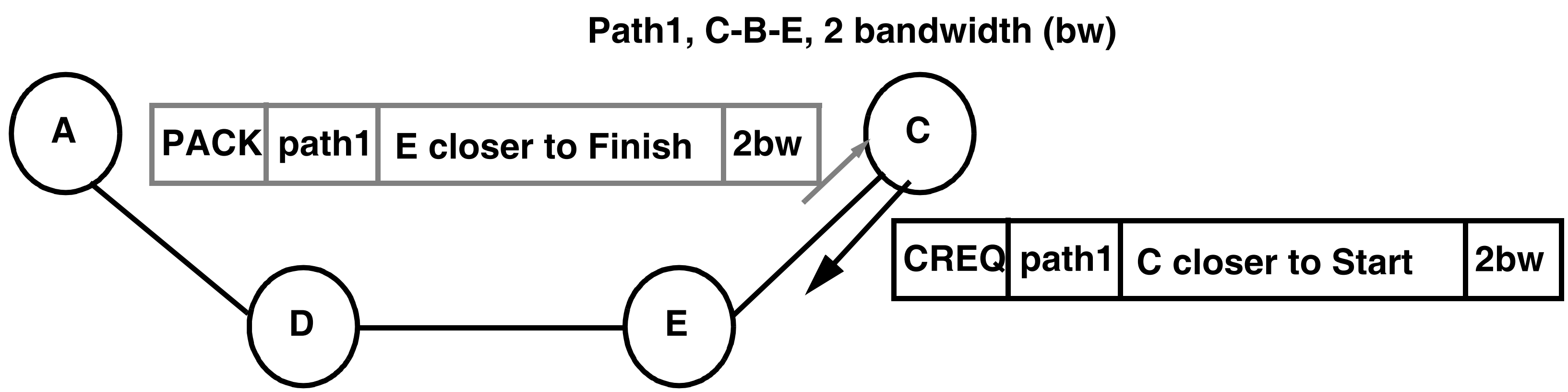


Figure 3.1c. Requesting a connection.

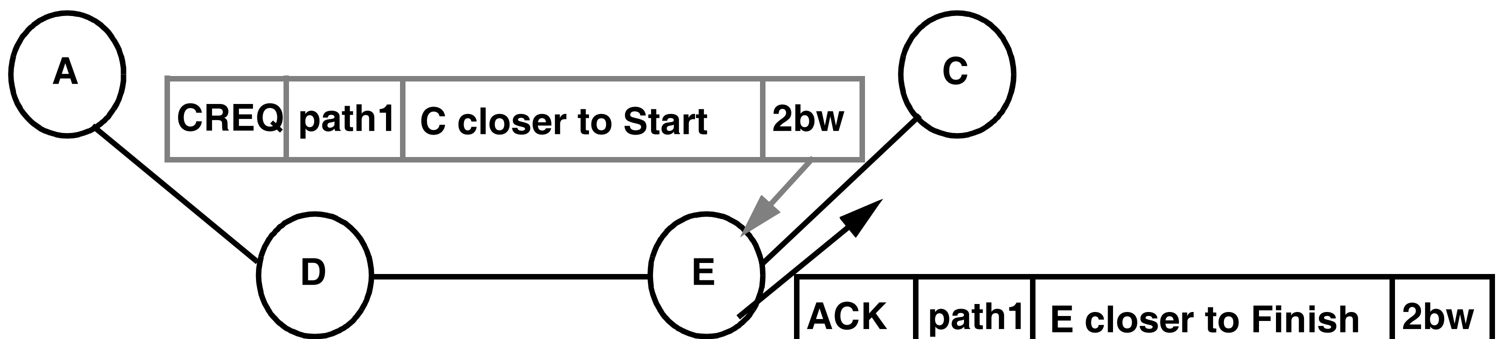


Figure 3.1d. Exploring the restoration path.

Path-base Two-Prong: Confirmation Phase

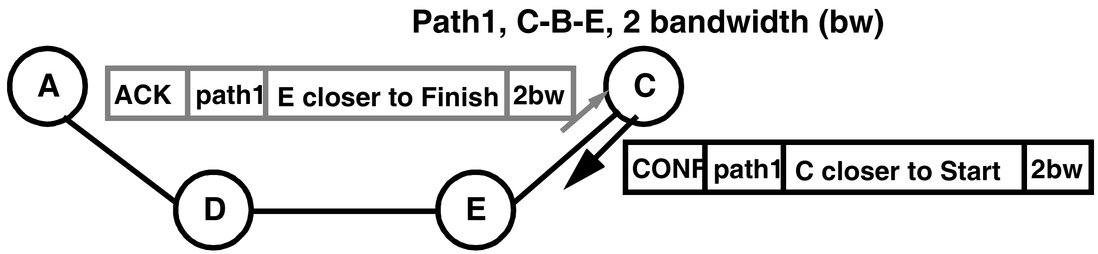


Figure 3.1e. Starting confirmation phase.

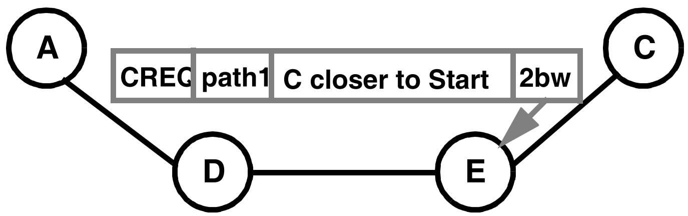


Figure 3.1f. Ignoring the connection request.

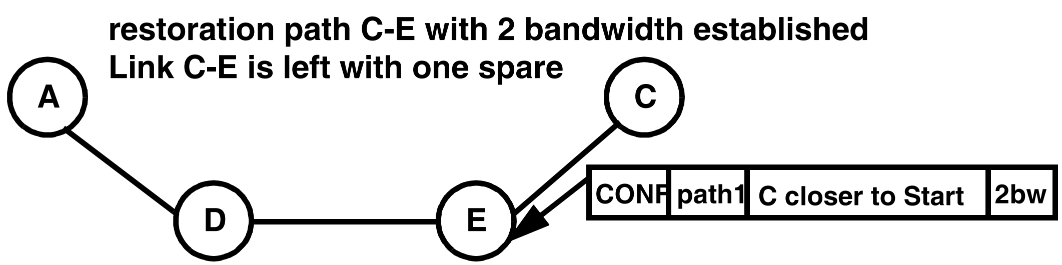
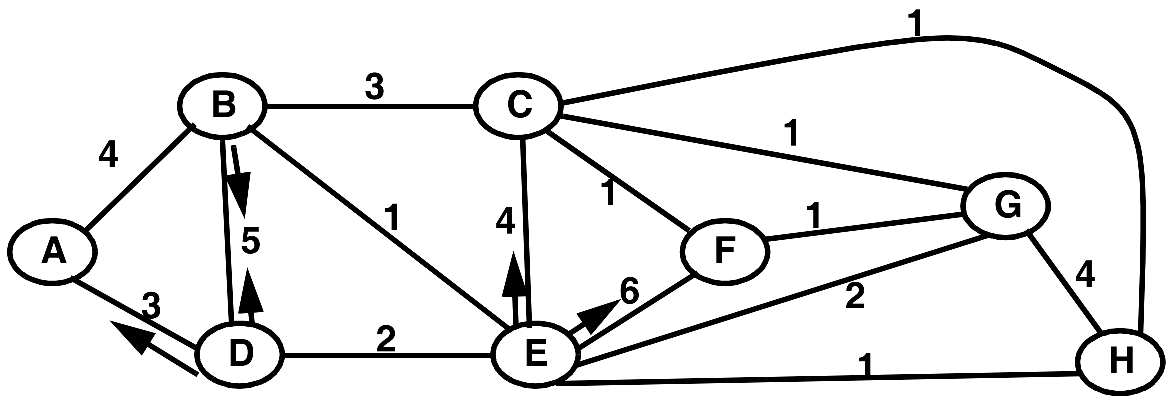


Figure 3.1g. Restoration path completed.

Problems Encountered in Path-based Two-Prong

- Lingering of old reservation requests after the path is restored.



Disrupted path: A-H with 6 bw.

Restoration paths found: A-B-C-H, 1bw
A-B-C-G-H, 1bw
A-B-C-F-G-H, 1bw
A-B-E-G-H, 1bw
A-D-E-H, 1bw
A-D-E-G-H, 1bw

Figure 3.2. Nodes B, D, and E still explore restoration paths.

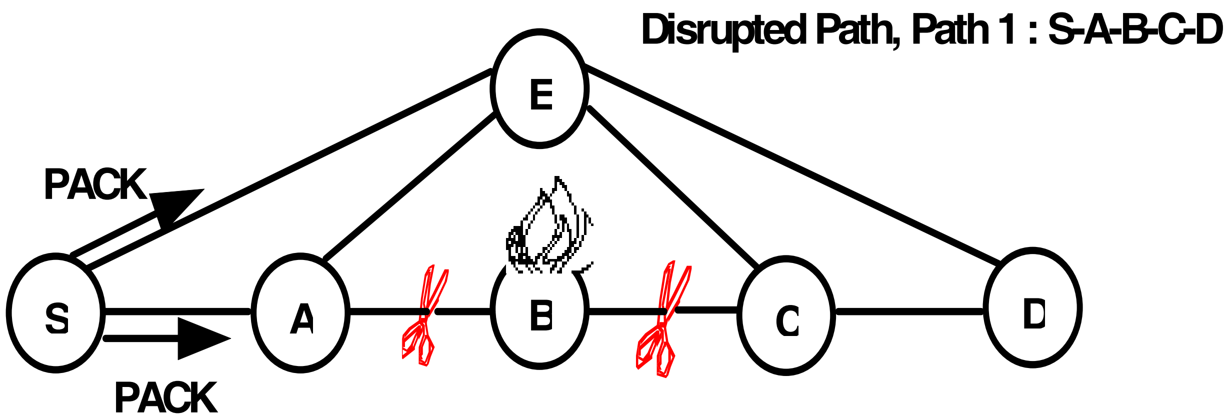
Sometimes the end nodes can detect the path is fully/maximally restored.

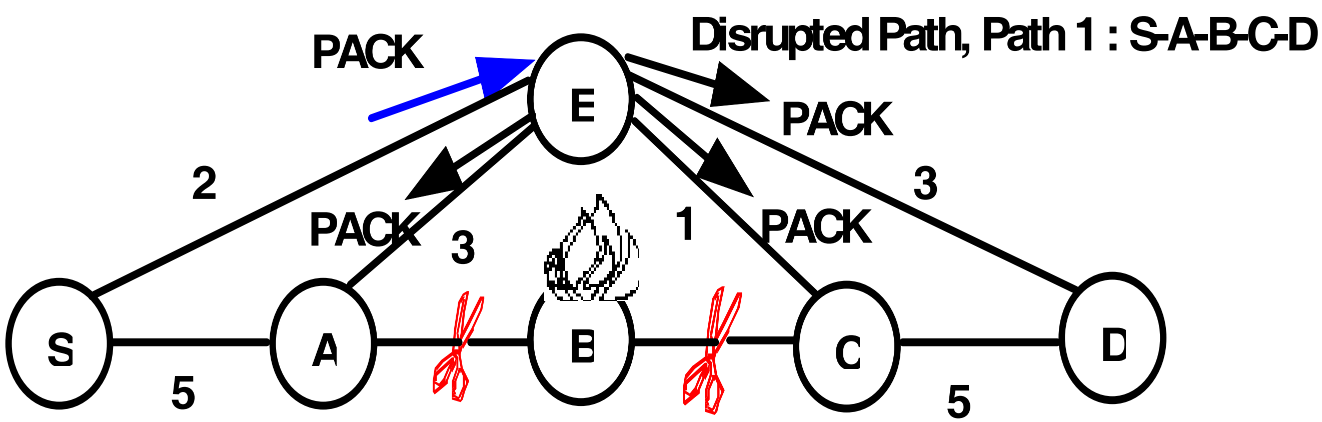
- Detecting the request cycle. (use trail fields)

Path based One Prong

- Use Sender Chooser control paradigm. (Multiple sender-chooser pairs.)
- Simplify Two Prong logic.
- Preliminary implementation is operational.
- On NJ test network, the restoration level is very close to Two Prong.
- The restoration time seems to be shorter.
- Message volume almost cut in half.
- There are bugs in the simulation program for some of the link cut, possibly due to the overflow on the message buffer.
- The additional simulation control parameters need to be incorporated and simulation results verified.

Adaptive One Prong: Broadcast Phase





Selective Message Forwarding

When a node receives a duplicate copy of the PACK message from another neighboring node, it will only forward the PACK message to the sending node of the first PACK message.

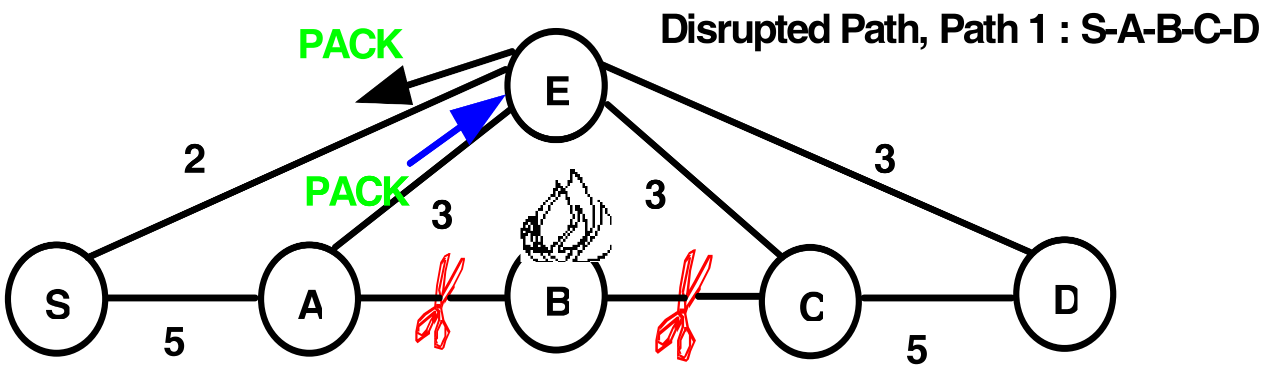
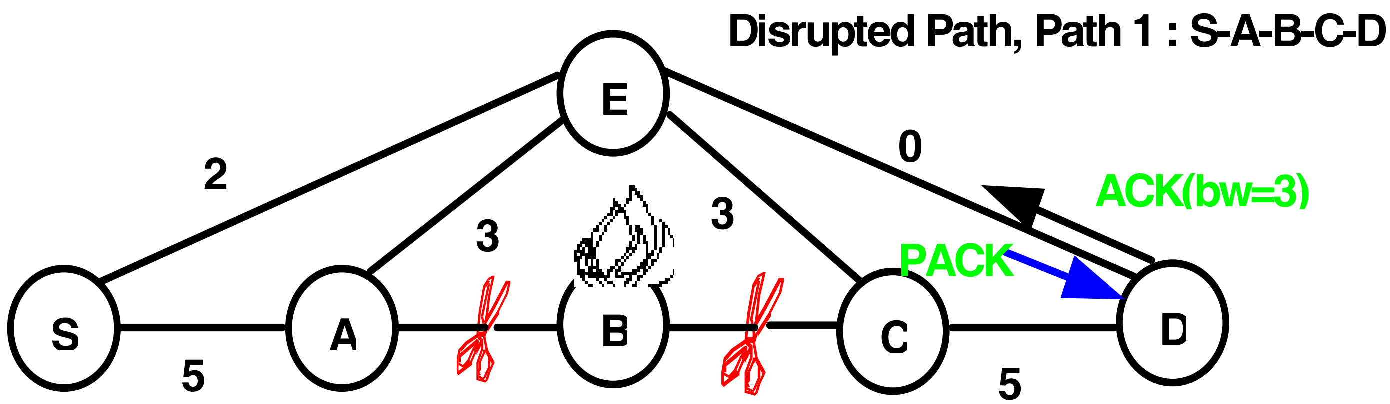


Table 1: Restoration Table in Node E

Path ID	UP	HOP	BW	DOWN	ACKHOP	Confirmed	Part	Msg_Send	Msg_Recv
1	S	1	5			0	M	0	0
1	A	2	5			0	M	0	0

Path Trace out Phase



Path Trace-out Phase

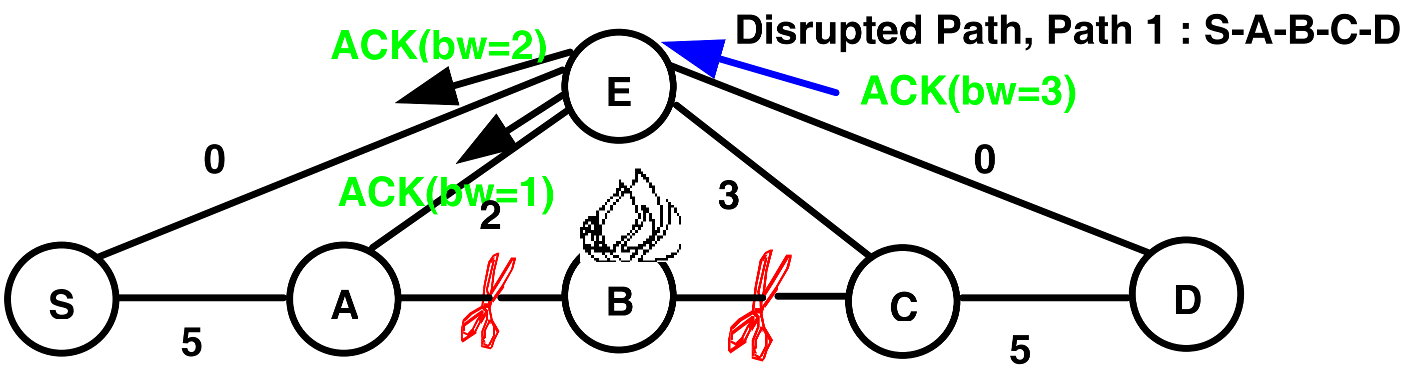


Table 3: Restoration Table in Node E after sending ACK(1bw) to A.

Path ID	UP	HOP	BW	DOWN	ACKHOP	Confirmed	Part	Msg_Send	Msg_Recv
1	S	1	2	D	1	ACKED	M	1	1
1	S	1	3			0	M	0	0
1	A	2	1	D	1	ACKED	M	2	1
1	A	2	4			0	M	0	0

Path Trace-out Phase

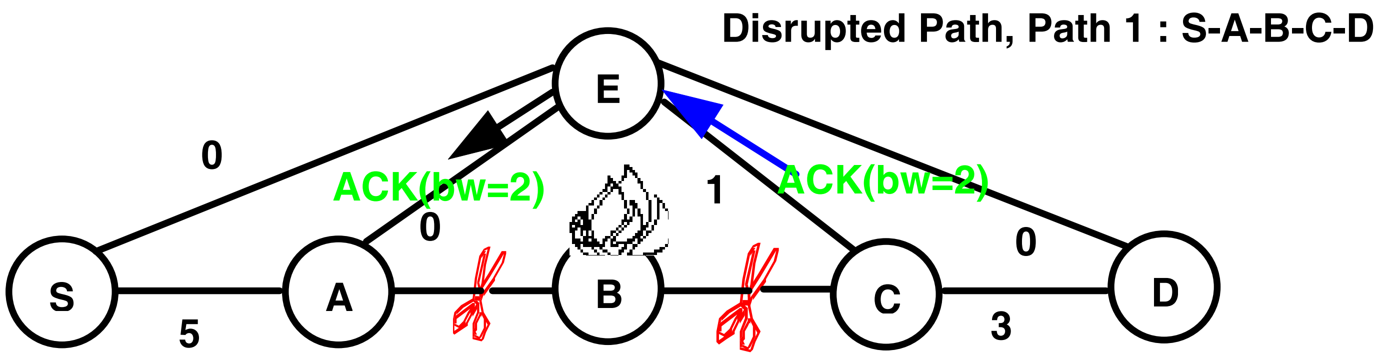


Table 4: Restoration Table in Node E after sending ACK(2bw) to A.

Path ID	UP	HOP	BW	DOWN	ACKHOP	Confirmed	Part	Msg_Send	Msg_Recv
1	S	1	2	D	1	ACKED	M	1	1
1	S	1	3			0	M	0	0
1	A	2	1	D	1	ACKED	M	2	1
1	A	2	2	C	2	ACKED	M	3	2
1	A	2	2			0	M	0	0

Handle partially satisfied request

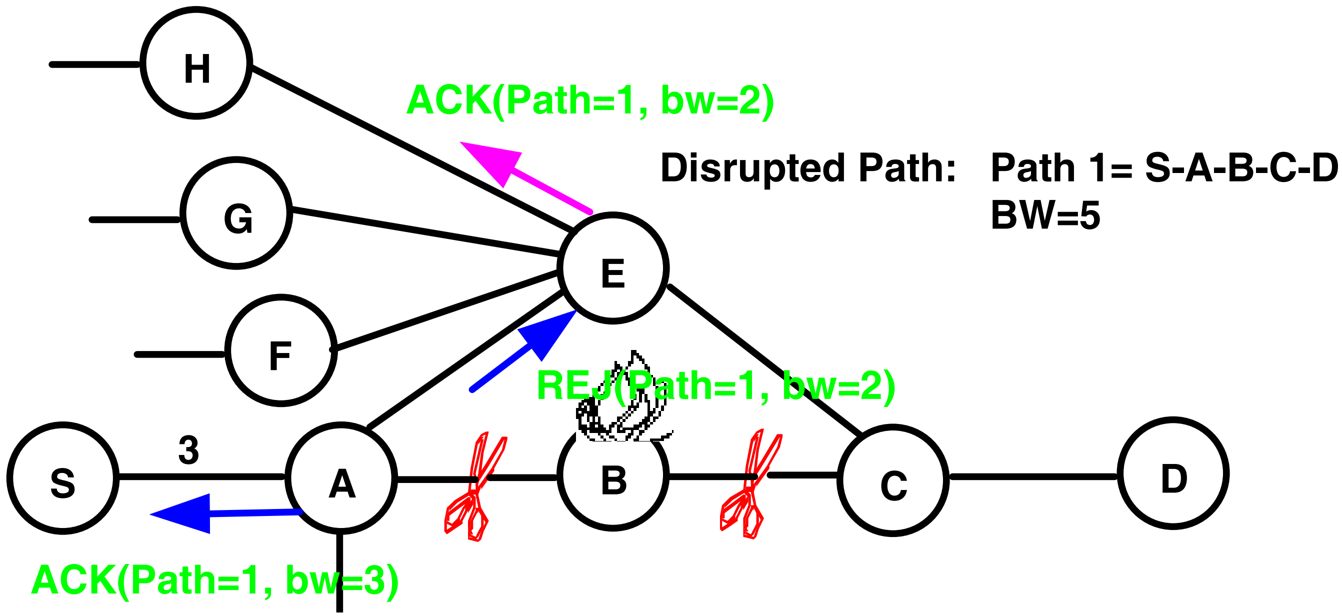


Table 10: Restoration Table in Node E after receiving REJ from A.

Path	UP	Retry	HOP	BW	DOWN	ACKHOP	Confirmed	Part	Msg_Send	Msg_Recv
1	A	1	2	3	D	1	ACKED	M	1	1
1	H	0	3	5	D	1	ACKED	M	2	1
1	H	0	3	5						
1	F	0	5	5						
1	A	1	2	2						

Confirmation Phase

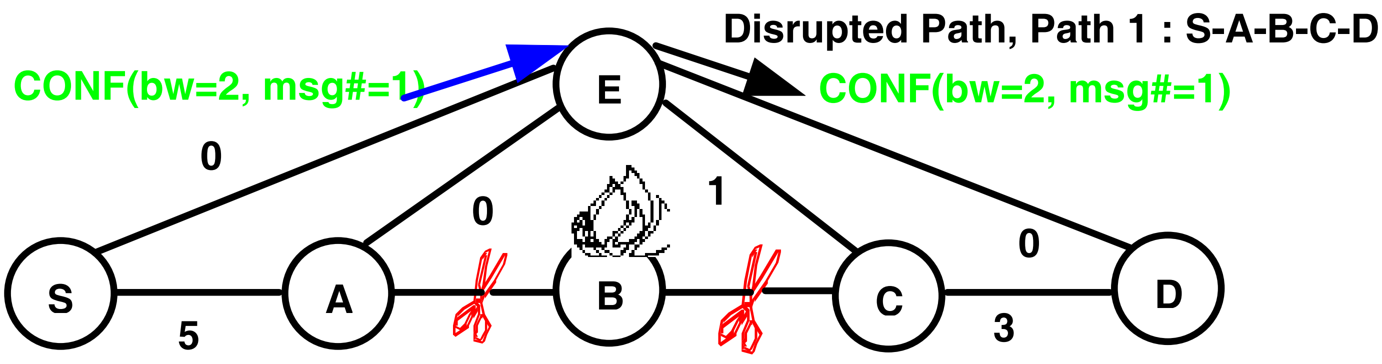
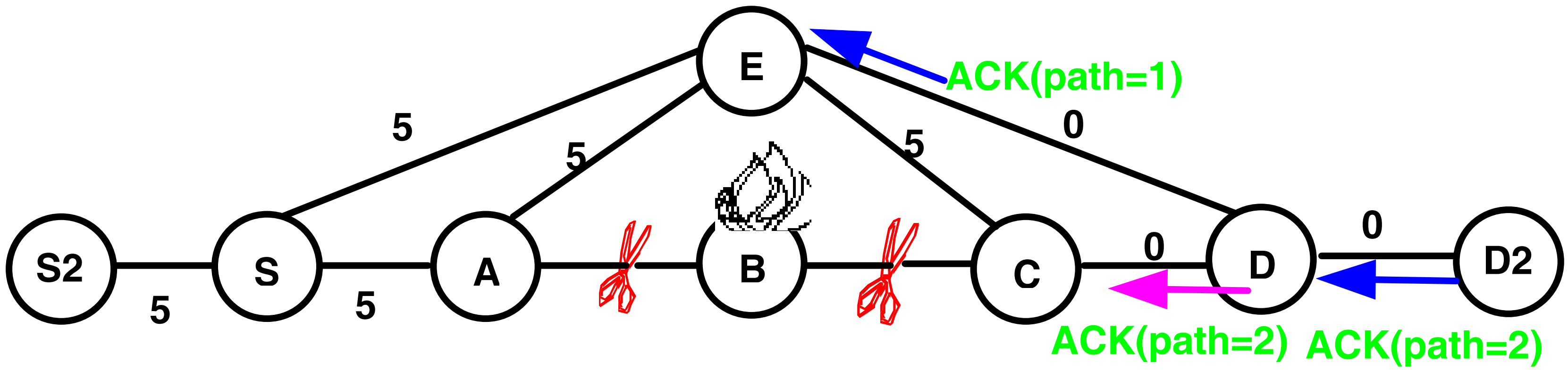
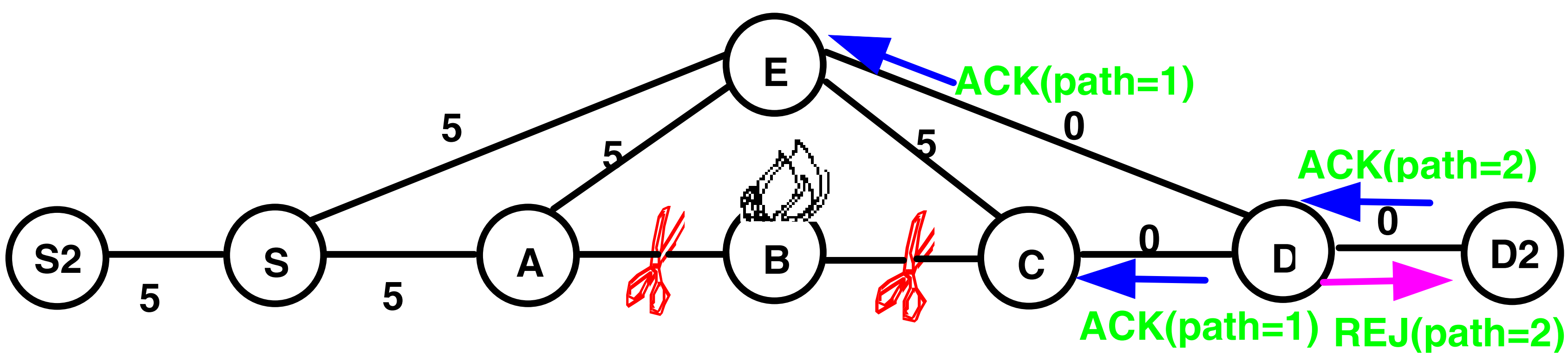


Table 5: Restoration Table in Node E after sending the first CONF to Node D.

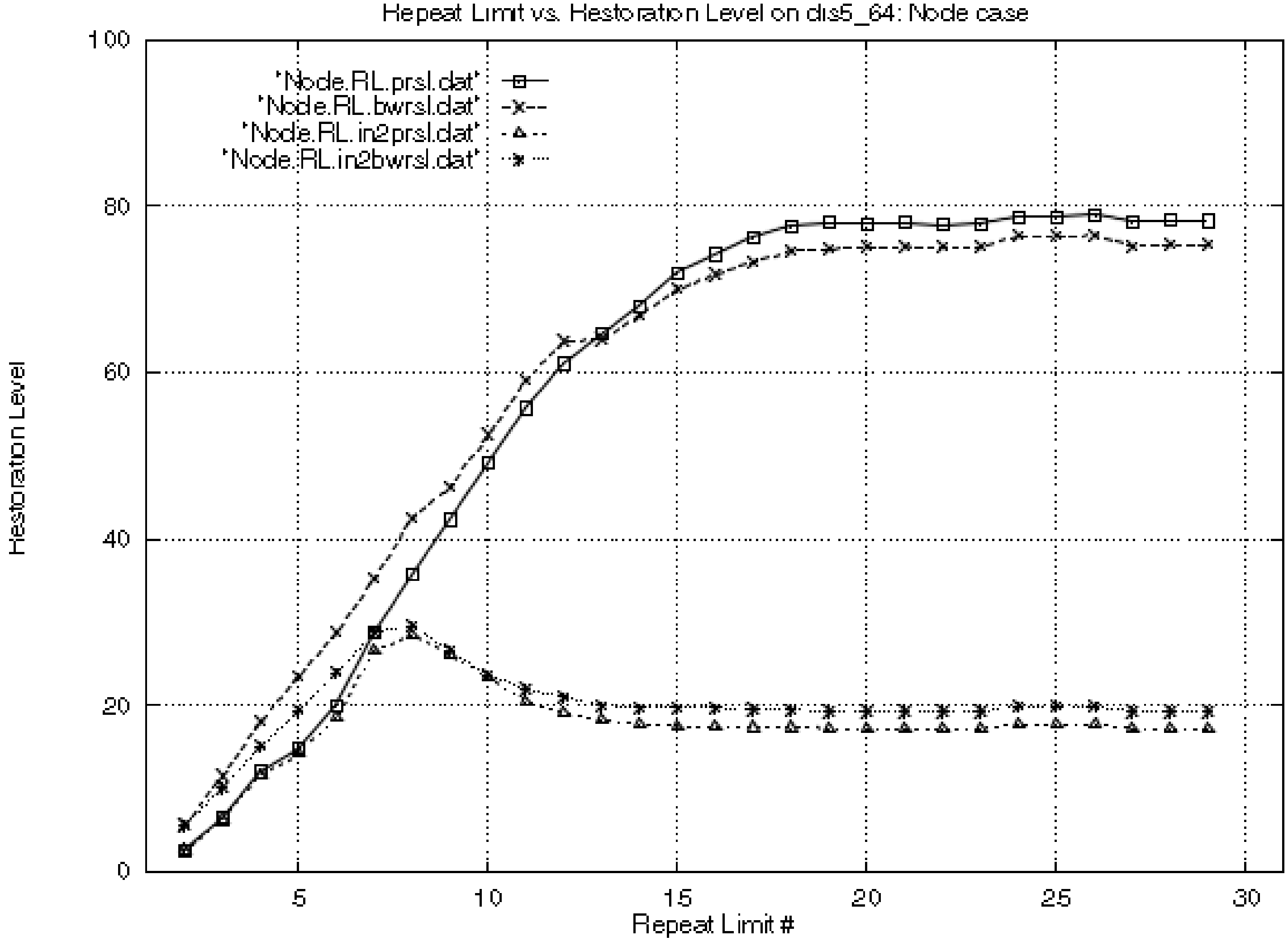
Path ID	UP	HOP	BW	DOWN	ACKHOP	Confirmed	Part	Msg_Send	Msg_Recv
1	S	1	2	D	1	CONF	M	1	1
1	S	1	3			0	M	0	0
1	A	2	1	D	1	ACKED	M	2	1
1	A	2	2	C	2	ACKED	M	3	2
1	A	2	2			0	M	0	0

Conservative vs. Aggressive Bandwidth Request

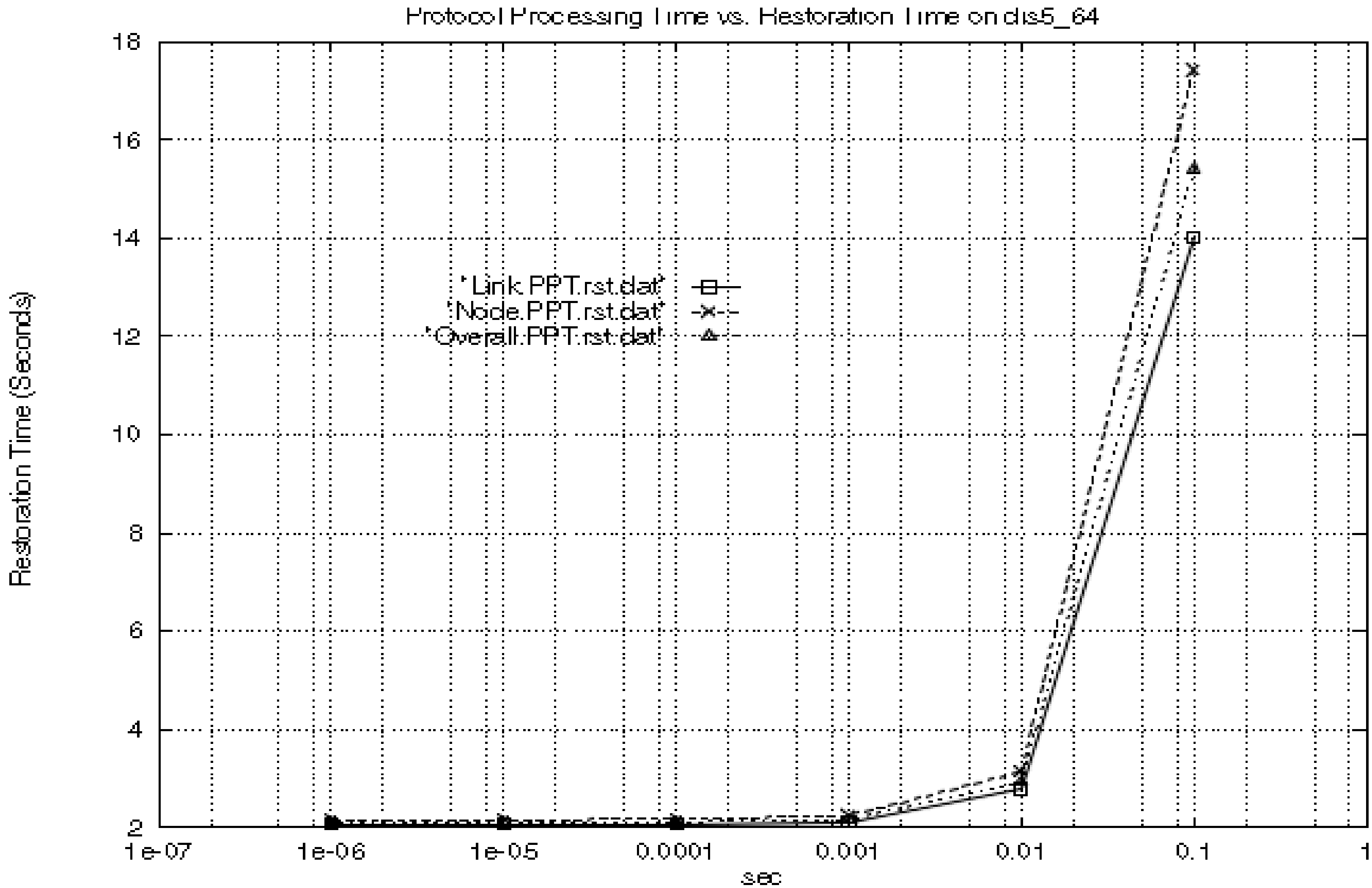




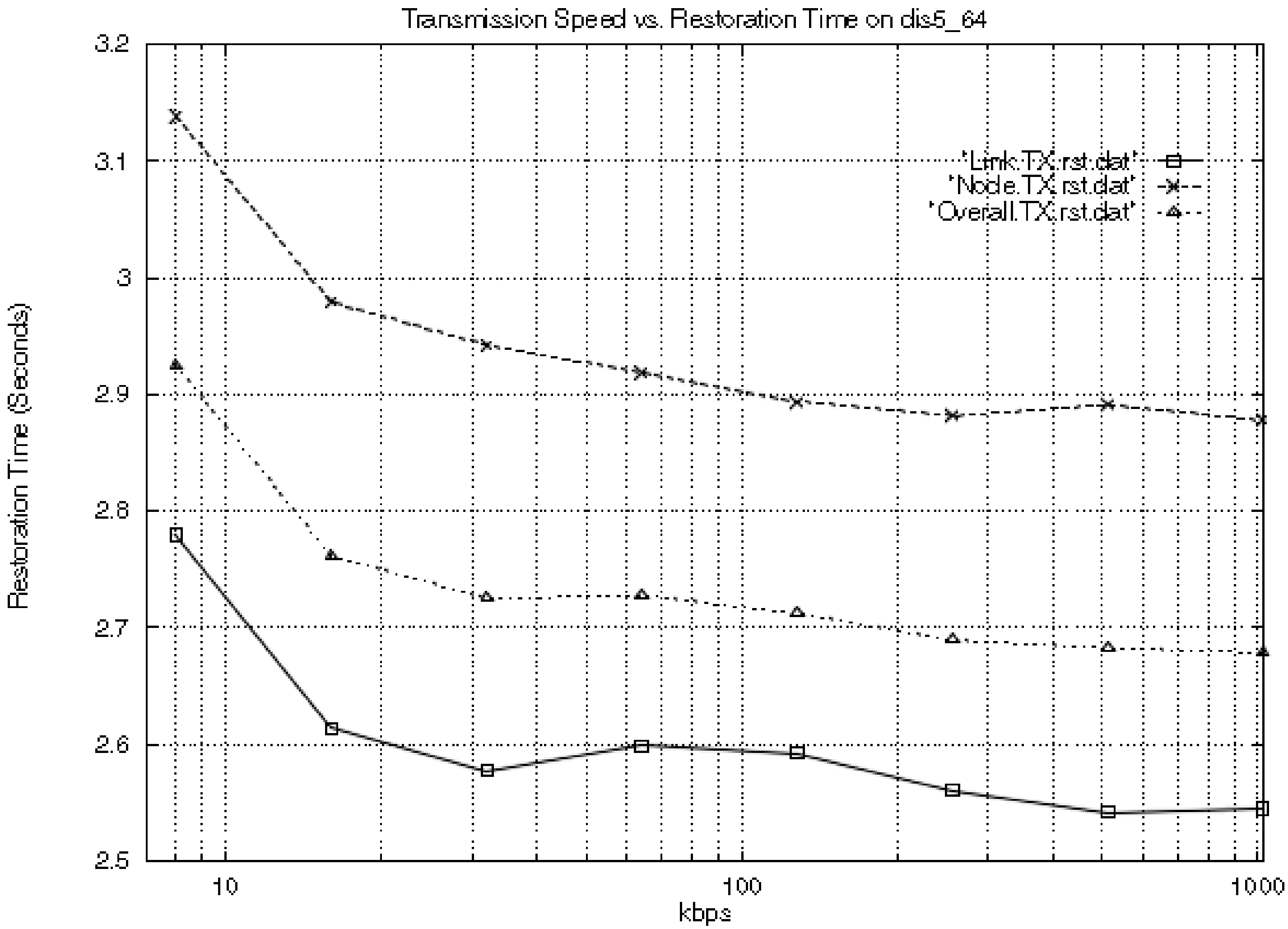
Impact of Repeat limit on Restoration level



Impact of Protocol Processing time on Performance



Impact of Transmission Speed



Comparison of Network Restoration Algorithms

Link Failure Cases

Scenario	Perf. Metric	Centralized Link-based	Two Prong Link/Path based	One Prong Path Based	GroverÕs SHN	FITNESS	RREACT
New Jersey Single Link Failure N01 - N02	Time msec Level Spares Used # of Msgs	257 100% 312 12	482/2959 100% 318/160 126/674	1287 74% 60 308	3126 100% 312 4269	1096 100% 343 136	582 100% 252 78
New Jersey Single Link Failure N04- N05	Time msec Level Spares Used # of Msgs	127 100% 237 12	107/874 100% 204/196 89/222	862 100% 290 157	3173 100% 212 2753	1151 100% 269 30	275 100% 204 107
New Jersey Single Link Failure N08 - N11	Time msec Level Spares Used # of Msgs	205 100% 301 13	475/4836 100%/98% 215/175 133/914	1363 100% 160 413	2387 93.75% 192 3177	1827 100% 239 197	739 100% 233 114

Restoration Time: centralized methods faster, link-based Two prong close.

Number of Messages: centralized methods have 8-10 times fewer messages.

SHN restoration slows down due to heavy message volume.

Comparison of Network Restoration Algorithms

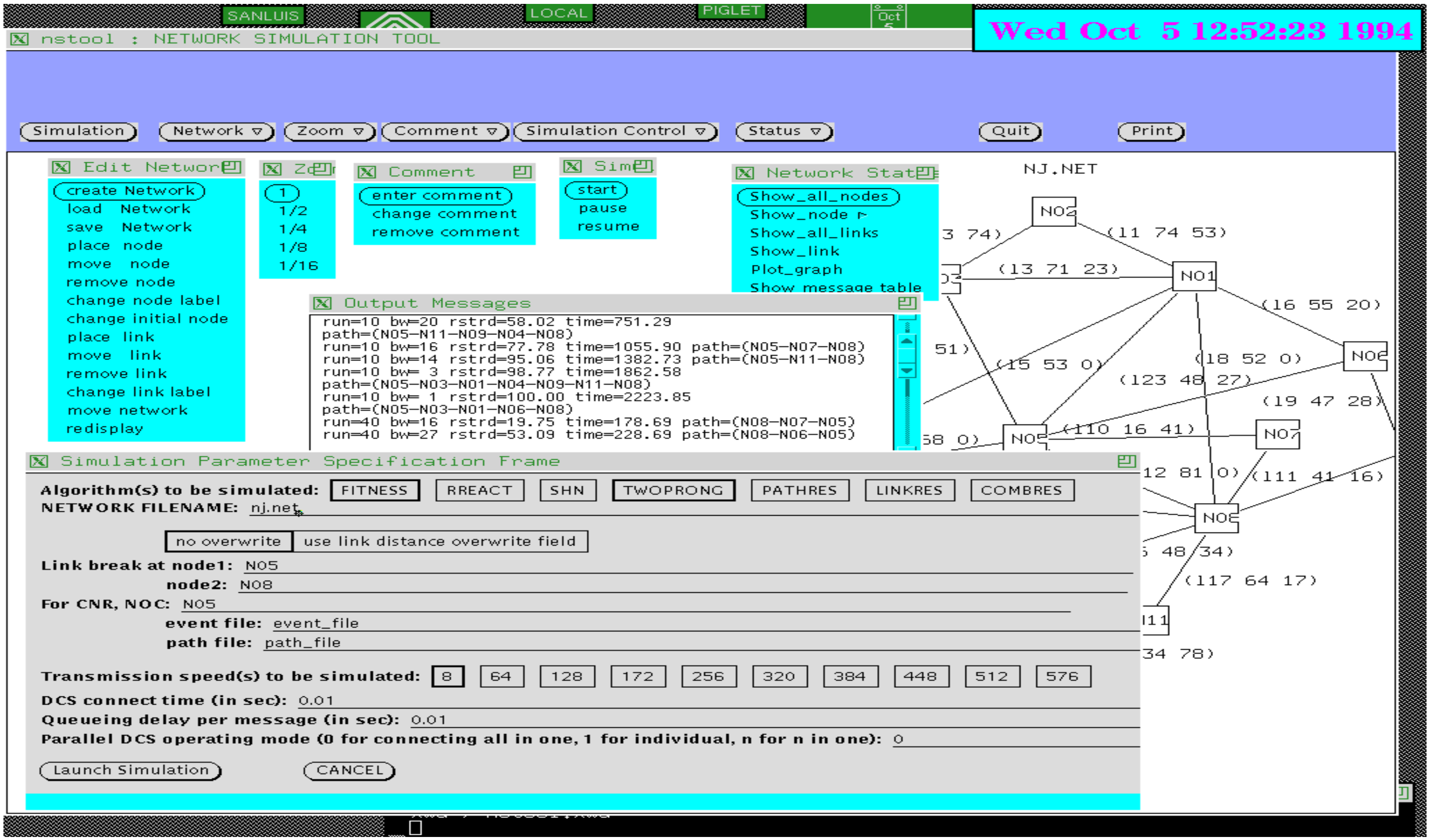
Node Failure Cases

Scenario	Perf. Metric	Centralized Path based	Centralized Link based	Centralized Combined	Komine*	Two Prong Path based	One Prong Path Based
New Jersey Single Node Failure N01	Time msec Level # of Msgs Spare Usage	467 100% 16 -	396 81. 9% 13 -	430 81. 9% 15 -	1445 57% 1413 -	35889 46% 863 3 (r168)	1526 46% 454 -53 (r168)
New Jersey Single Node Failure N04	Time msec Level # of Msgs Spare Usage	191 100% 33 -	153 100% 28 -	167 100% 11 -	2025 100% 1647 -	3175 100% 596 61(r61)	996 100% 209 63(r61)
New Jersey Single Node Failure N05	Time msec Level # of Msgs Spare Usage	416 100% 18 -	370 90. 5% 18 -	426 100% 18 -	2337 91% 1633 -	18954 69% 5246 72(r102)	1940 80% 442 61(r102)

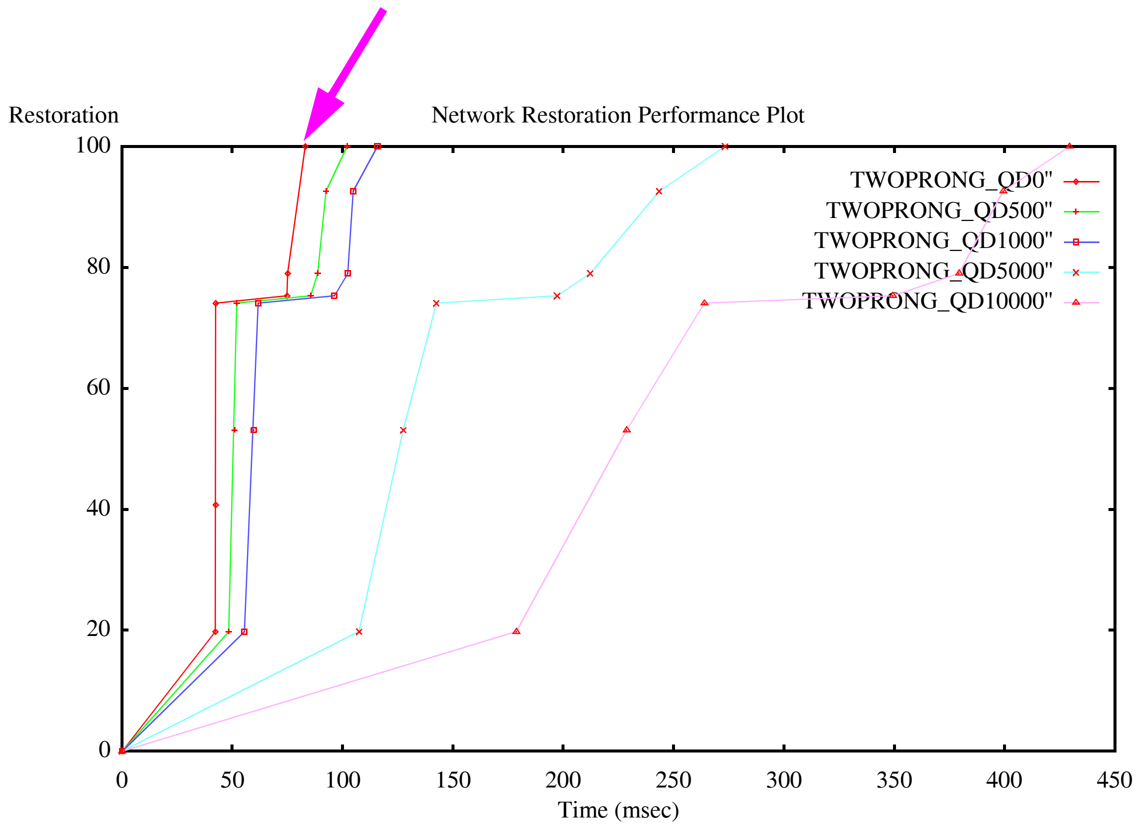
* These are the simulation results of our implementation of FujitsuÕs KOMINE network restoration algorithm.

NSTOOL 1.2

Built a graphical user interface to facilitate the survivable network design.



Exploring the theoretical limitation of survivable network algorithms



Generated by GNUPLOT