How to model a TCP/IP network using only 20 parameters

visual hash of *MesoNet* source code from http://www.wordle.net/

K. Mills (NIST), E. Schwartz (CMU) & J. Yuan (Tsinghau U)

Winter Simulation Conference – Dec 8, 2010

Outline

- Goal – Problem – Solution
- Scale Reduction: Theory and Practice
- Overview of the 20 *MesoNet* Parameters
- Parameter Explanations in 5 Categories
  - Network (4 parameters)
  - Sources & Receivers (4 parameters)
  - User Behavior (6 parameters)
  - Protocols (3 parameters)
  - Simulation & Measurement Control (3 parameters)
- Describe Sample Use of Model
- Discuss Simulation Resource Requirements
- Conclusions

Goal – Problem – Solution

- **Goal** – compare proposed Internet congestion control algorithms under a wide range of controlled, repeatable conditions

- **Problem** – real network not controllable and repeatable; test beds *currently* too small; *most* network simulation models have large search space and require infeasible memory and processing resources for large, fast networks; more tractable fluid-flow simulators are *currently* inaccurate

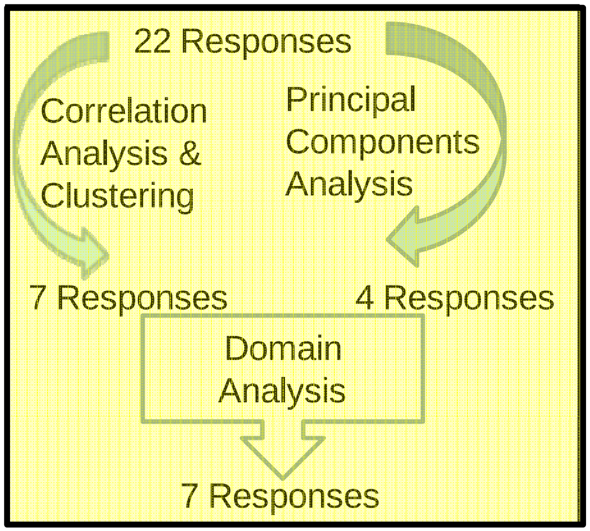
- **Solution** – design a reduced scale network simulation model (***MesoNet***) that is easy to configure and tractable to compute

Scale Reduction: Theory & Practice

Simulating large, fast networks across many conditions and congestion control algorithms requires scale reduction in both model parameters & responses

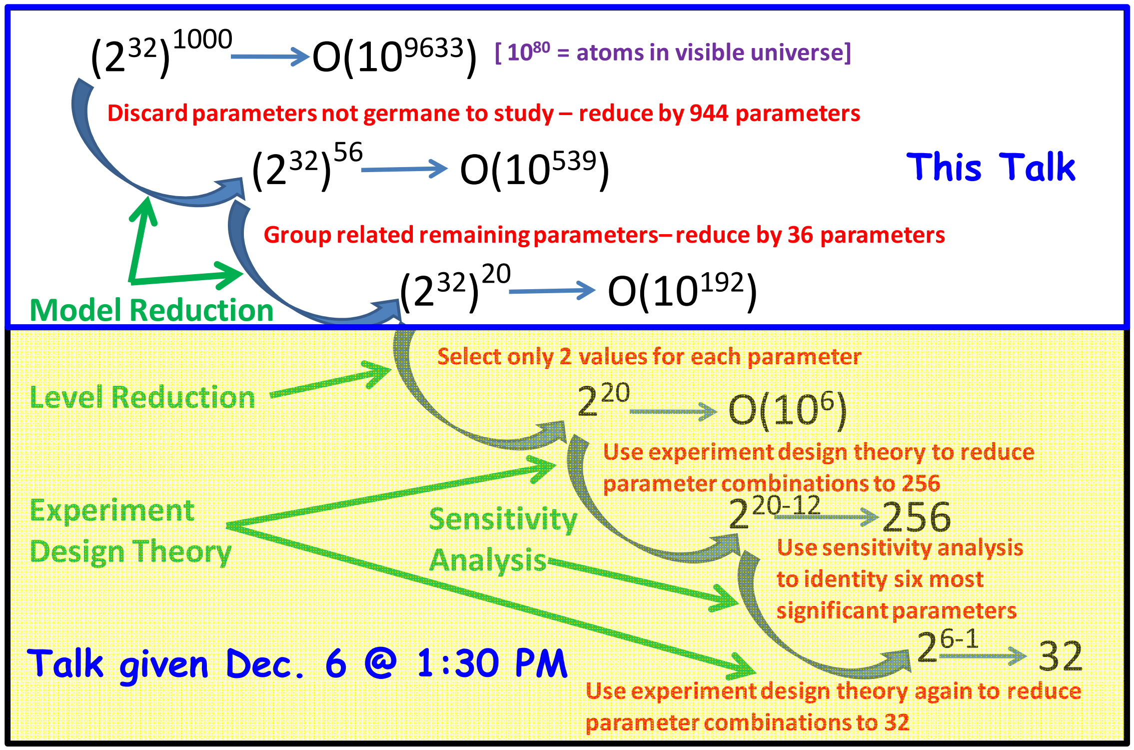
$$\underbrace{y_1, \ldots, y_z}_{\text{Response State-Space}} = f(\underbrace{x_{1|[1,\ldots,\ell]} \ldots, x_{p|[1,\ldots,\ell]}}_{\text{Stimulus State-Space}})$$

Multidimensional Response Reduction



Talk given Dec. 6 @ 2:00 PM

Parameter Reduction



Model Reduction for ***MesoNet*** Simulator

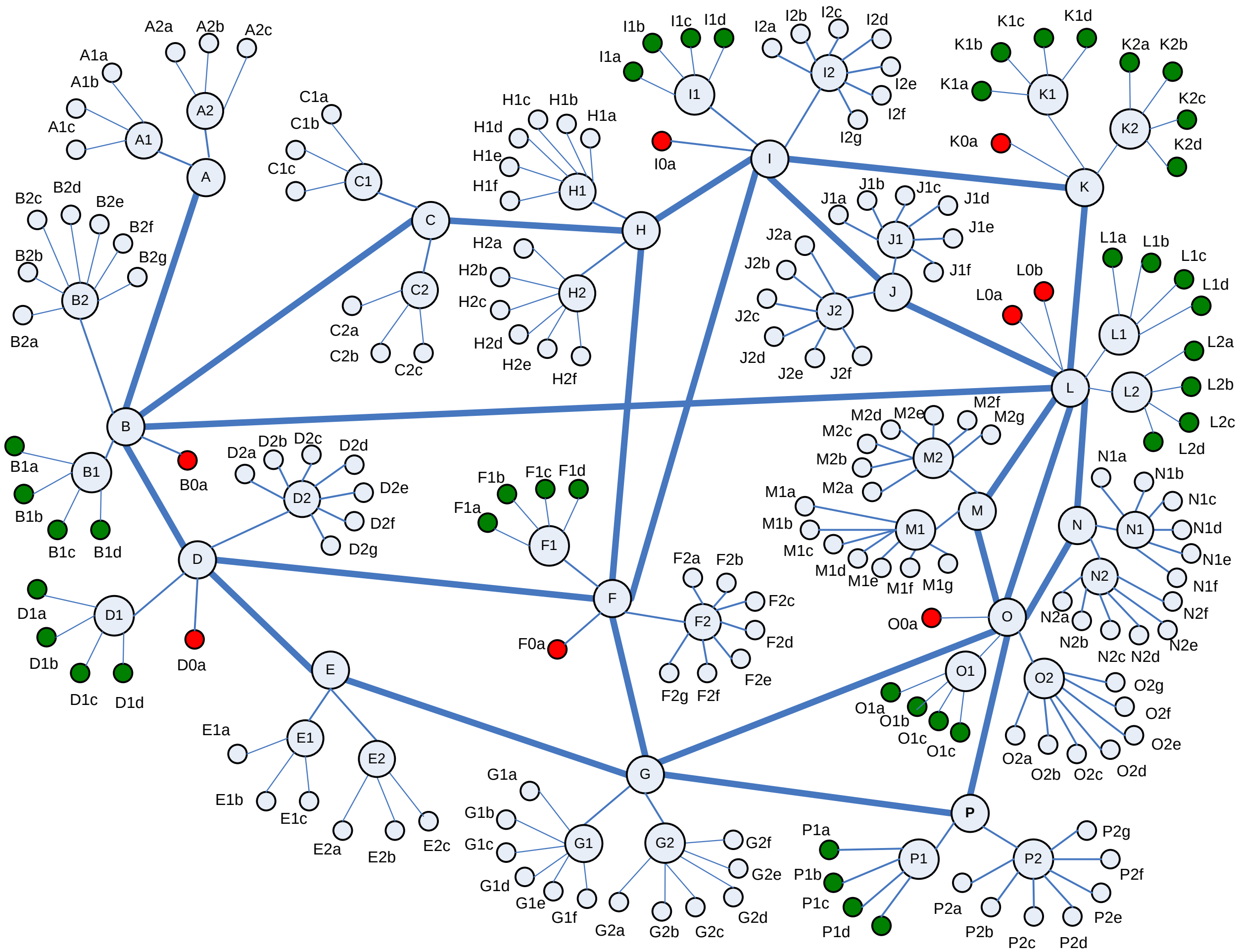
- Need to identify and retain only parameters germane to topic being studied (we identified 56 such parameters)
- Need to examine retained parameters to identify groups of related parameters defining aspects of same macro-parameter (after grouping we identified 20 parameters relevant to a study of Internet congestion control)

For a full explanation of our reasoning and our entire study report see NIST Special Publication 500-282: <u>Study of Proposed Internet Congestion Control Mechanisms</u>
Available online at http://www.nist.gov/itl/antd/Congestion_Control_Study.cfm

MesoNet – a TCP/IP network model using only 20 parameters

Category	Identifier	Name
Network Configuration	X1	Topology
	X2	Propagation Delay
	X3	Network Speed
	X4	Buffer Provisioning
Sources & Receivers	X5	Number of Sources & Receivers
	X6	Distribution of Sources
	X7	Distribution of Receivers
	X8	Source & Receiver Interface Speeds
User Behavior	X9	Think Time
	X10	Patience
	X11	Web Object Size for Browsing
	X12	Proportion & Size of Larger File Downloads
	X13	Selected Spatiotemporal Congestion
	X14	Long-lived Flows
Protocols	X15	Congestion Control Algorithms
	X16	Initial Congestion Window Size
	X17	Initial Slow Start Threshold
Simulation & Measurement Control	X18	Measurement Interval Size
	X19	Simulation Duration
	X20	Startup Pattern

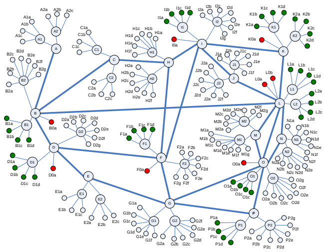
Parameter X1 is the Topology =
Routers + Links + Routes + Propagation Delays



3 Router Tiers: Backbone – Point of Presence (PoP) – Access

3 Access Router Classes: Typical – Fast – Directly Connected

1 ingress/egress path from access routers to backbone routers



Topology Link Characteristics and Scaling Propagation Delay with Parameter X2

Link#	Endpoints	Cost Metric	Prop. Delay (ms)	X2 = 0.5	X2 = 2
1	A-B	50	21	10.5	42
2	B-C	10	25	12.5	50
3	B-D	50	8	4	16
4	B-L	223	75	37.5	150
5	C-H	100	12	6	24
6	D-E	10	10	5	20
7	D-F	108	33	16.5	66
8	E-G	100	33	16.5	66
9	F-G	10	7	3.5	14
10	F-H	50	12	6	24
11	F-I	55	22	11	44
12	G-O	104	23	11.5	46
13	G-P	110	19	9.5	38
14	I-H	10	14	7	28
15	I-J	50	8	4	16
16	I-K	147	22	11	44
17	J-L	60	20	10	40
18	K-L	50	7	3.5	14
19	L-M	50	12	6	24
20	L-N	39	6	3	12
21	L-O	10	14	7	28
22	M-O	10	6	3	12
23	N-O	10	8	4	16
24	O-P	10	14	7	28

- Packets incur propagation delay when transiting a link
- Cost metric used to compute routes from source backbone router to destination backbone router

Defined Speed Relationships among Router Classes used to Scale Router Speeds with Parameter X3

(*MesoNet* simplification – only routers have speeds)

Parameter	Value	Speed Relationships		Speed Scaling with X3	
		Router Class	Speed	X3 = 800	X3 = 1600
*s*1	X3	Backbone	*s*1 x *BBspeedup*	1600	3200
*s*2	4	PoP	*s*1/ *s*2	400	800
*s*3	10	N-Class	*s*1/ *s*2/ *s*3	40	80
BBspeedup	2	F-Class	*s*1/ *s*2/ *s*3 x *Bfast*	80	160
Bfast	2	D-Class	*s*1/ *s*2/ *s*3 x *Bdirect*	400	800
Bdirect	10				

Parameter X4 selects the Buffer Provisioning Algorithm, which generally interacts with network speed and propagation delay

Three Parameters Determine Number (X5) and Distribution of Sources (X6) and Receivers (X7)

Combination of parameters X5, X6, and X7 determine distribution of flows in the topology during the simulation

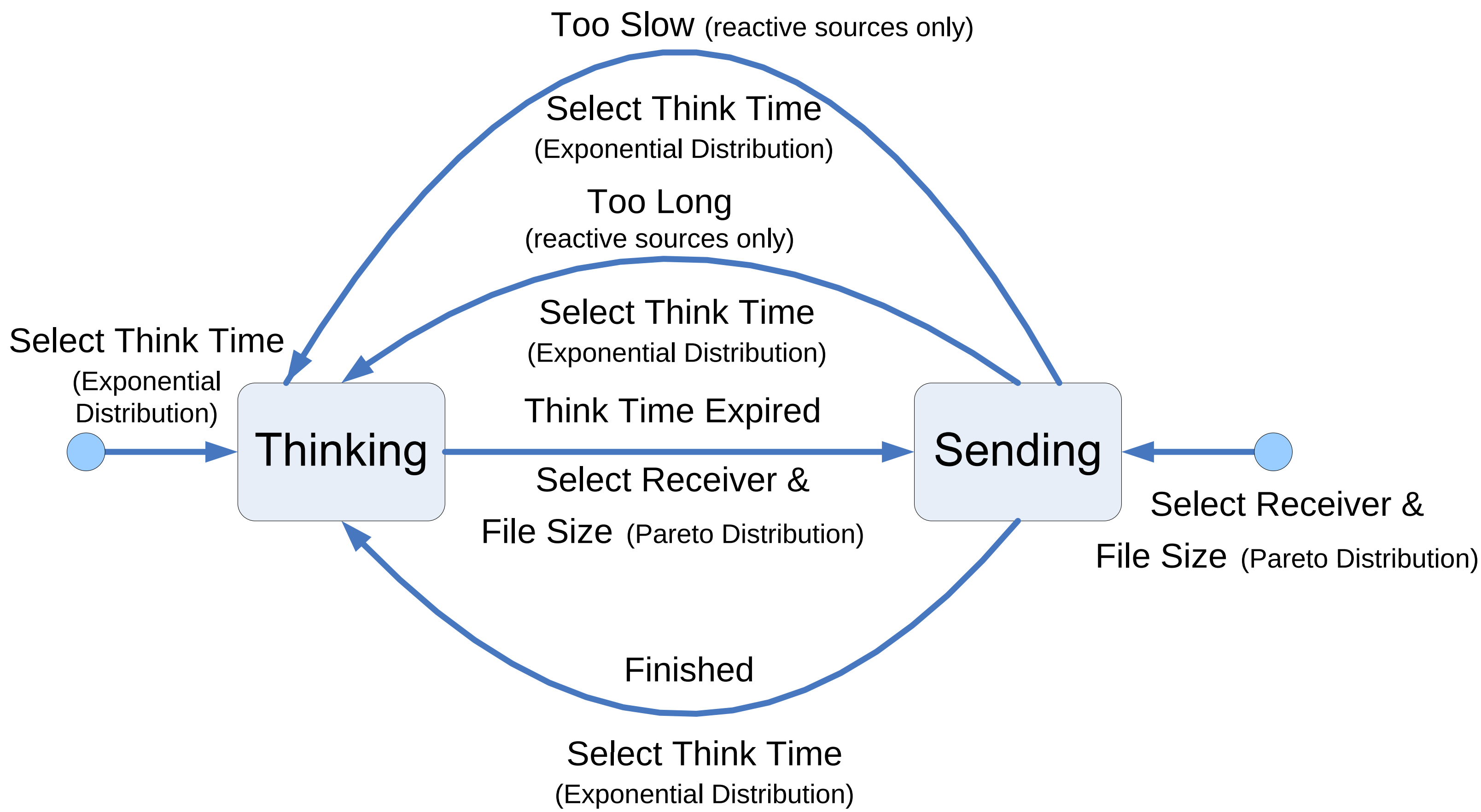
Sample Computation of Number and Distribution of Sources and Receivers
(given Topology on Slide 7 and *base # Sources* = 100, X5 = 3, *probNs* = 0.1, *probNsf* = 0.6, *probNr* = 0.8, *probNrf* = 0.1)

Class	#routers	srcs/router	#srcs	%srcs	rcvrs/router	#rcvrs	%rcvrs
N-class	122	90	10,980	31.6	960	117,120	95.3
F-class	40	540	21,600	62.2	120	4,800	3.9
D-class	8	270	2,160	6.2	120	960	0.8

Flow class	%flows
NN-flows	30.1
FN-flows	60.5
FF-flows	2.4
DN-flows	6.1
DF-flows	0.74
DD-flows	0.05

Parameter X8 defines the probability that sources and receivers connect to the topology at 1 Gbps or 100 Mbps

User Behavior Represented via Sources^



Parameter x9 specifies average Think Time

Parameter x10 specifies User Patience (probability source is reactive)

^Note: this simplified diagram omits a flow connection phase that occurs before sending and also the potential for the connection phase to fail – after which source enters Thinking

User Traffic Characterization

Parameter X11 characterizes **Web Objects** **(λon, α)**

Size of Web Objects

λon	average size (packets)
α	shape of Pareto distribution

Probability of Web Object
(1 – *Fp* – *Sp* – *Mp*)

Parameter x12 characterizes **Larger Files** **[(*Fx*, *Fp*), (*Sx*, *Sp*), (*Mx*, *Mp*)]**

Larger File Size Multipliers

Fx	documents
Sx	software downloads
Mx	movies

Larger File Probabilities

Fp	documents
Sp	software downloads
Mp	movies

Parameter X13 characterizes **Jumbo Files** **(*Jx*, *Jon*, *Joff*)**

Jumbo File Characteristics

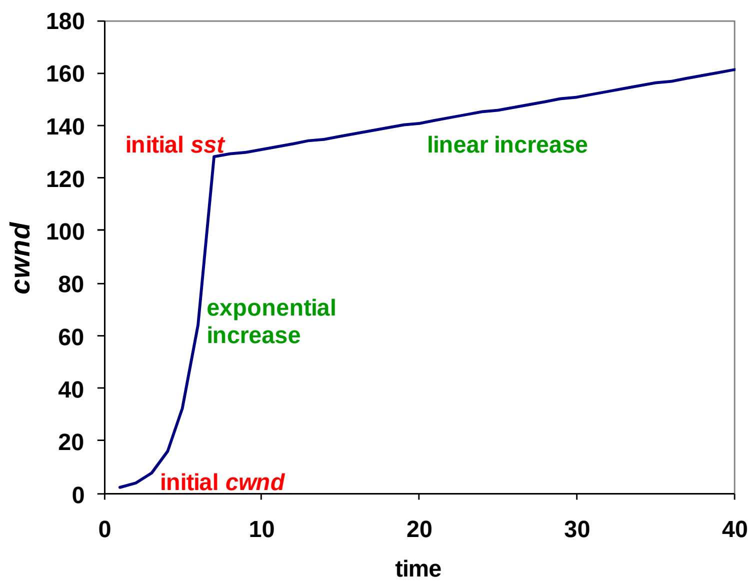
Jx	size multiplier for jumbo files
Jon	fraction of simulated time after which jumbo file transfers begin
Joff	fraction of simulated time after which jumbo file transfers end

Parameter X14 characterizes number, location and start and stop times for **Long-Lived Flows**

Assignment of Three Protocol Parameters

Parameter x15 specifies **(***prTCP***,** *prHSTCP***,** *prCTCP, prSCALABLE, prFAST, prHTCP, prBICTCP***)**

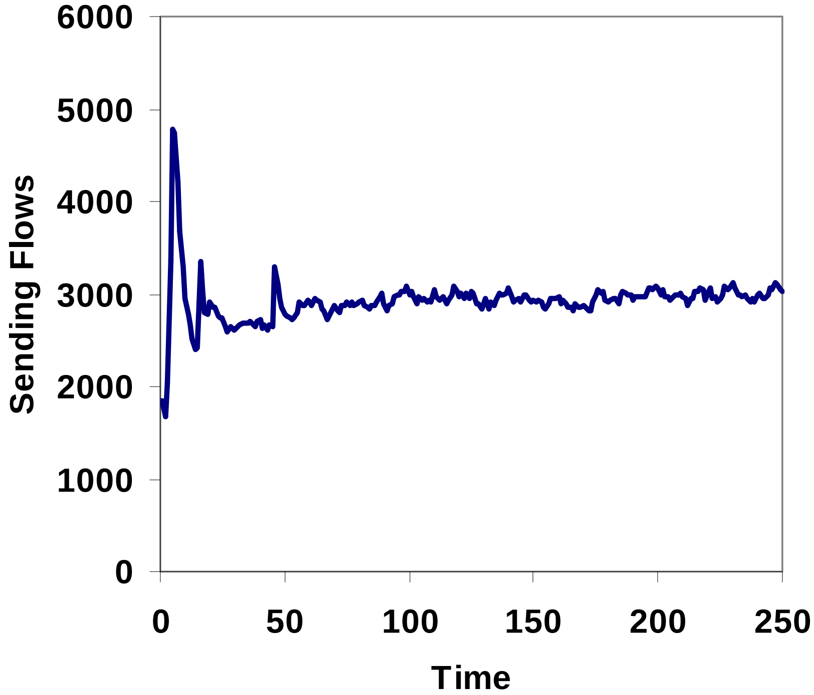
Congestion Control Algorithm	Identifier	Probability of Source Implementation
Transmission Control Protocol (TCP)	**1**	***prTCP***
High Speed TCP (HSTCP)	**2**	***prHSTCP***
Compound TCP (CTCP)	**3**	***prTCP***
Scalable TCP (STCP)	**4**	***prSTCP***
FAST AQM Scalable TCP (FAST)	**5**	***prFAST***
Hamilton TCP (HTCP)	**6**	***prHTCP***
Binary Increase Congestion (BIC)	**7**	***prBIC***



Parameter x16 specifies initial congestion window (*cwnd*)

Parameter x17 specifies initial slow start threshold (*sst*)

Simulation Measurement & Control



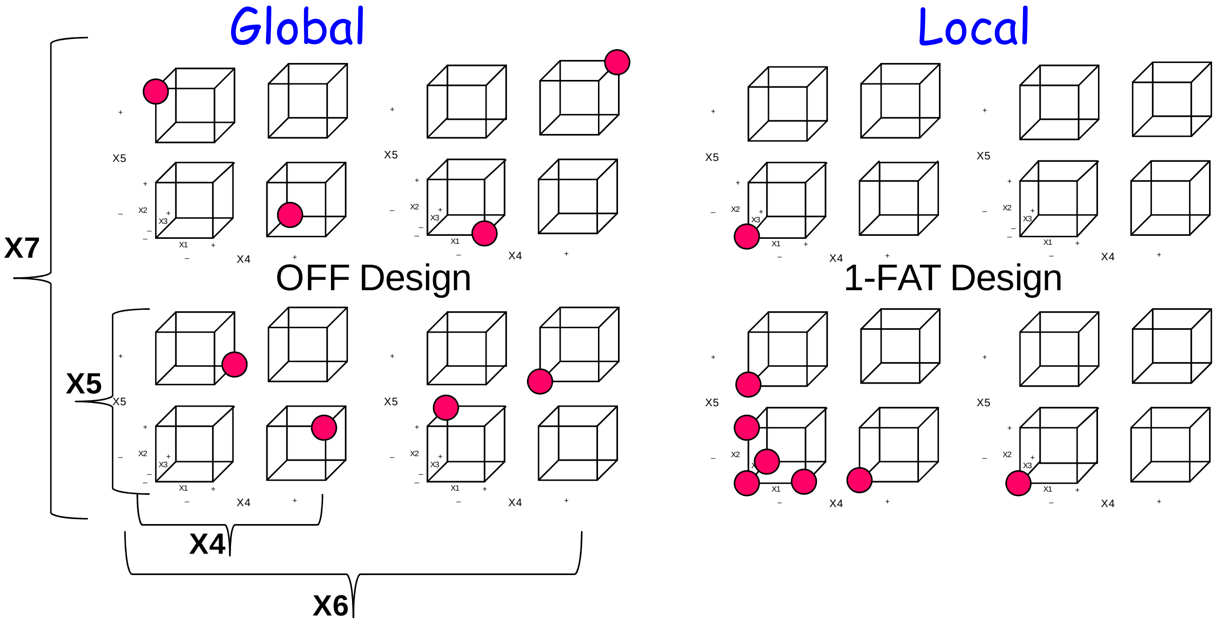
Count of Flows in the Sending State Measured every *M* = 200 ms for *MI* = 250 intervals – Simulation Duration (.2 s x 250 =) 50 s – 0.25 % of sources starting a t=0, 0.08 % of sources starting after an average delay 33 % of think time, 0.17 % of sources starting after an average delay 66 % of think time and remaining sources starting after average delay of think time

Parameter x18 specifies measurement interval size

Parameter x19 specifies number of measurement intervals to simulate

Parameter x20 specifies startup pattern for sources

Combining *MesoNet* Parameters with 2-Level Orthogonal Fractional Factorial (OFF) Experiment Design



Example Comparing OFF Design vs. Factor-at-a-Time (FAT) Design for 7 parameters

Comparing 7 Congestion Control Algorithms with 2-Level design for, 9 *MesoNet* Parameters requires (2^9 x 7 =) 3584 runs

At 28 processor hours per run and with 48 available processors, these runs would require about 2090 hours (87 days)

Adopting a 2^{9-4} OFF experimental design would reduce the resource requirement to only (32 x7) = 224 runs, which could be completed in about 130 hours (1 week)

Cost: misses 2^9 - 2^5 parameter combinations

Two Sample Experiments using 2^{9-4} Orthogonal Fractional Factorial Design

One Experiment Design – Two Experiments

Definition of the 32 Parameter Configurations used to Simulate a Modest Size, Moderate Speed Network in Experiment #1.
For Experiment #2 red values for X3 and # Sources were multiplied by 10 to Simulate a Larger, Faster Network.

Factor->	X2	X3	X4	X5	X7	X9	X11	X12	X15
Condition	--	--	--	--	--	--	--	--	--
1	1	800	0.5	3	0.7	5000	100	0.04/0.004/0.0004	0.7
2	1	1600	0.5	2	0.3	5000	100	0.04/0.004/0.0004	0.3
3	2	800	0.5	2	0.7	5000	100	0.02/0.002/0.0002	0.3
4	2	1600	0.5	3	0.3	5000	100	0.02/0.002/0.0002	0.7
5	1	800	1	2	0.3	5000	100	0.02/0.002/0.0002	0.7
6	1	1600	1	3	0.7	5000	100	0.02/0.002/0.0002	0.3
7	2	800	1	3	0.3	5000	100	0.04/0.004/0.0004	0.3
8	2	1600	1	2	0.7	5000	100	0.04/0.004/0.0004	0.7
9	1	800	0.5	3	0.3	7500	100	0.02/0.002/0.0002	0.3
10	1	1600	0.5	2	0.7	7500	100	0.02/0.002/0.0002	0.7
11	2	800	0.5	2	0.3	7500	100	0.04/0.004/0.0004	0.7
12	2	1600	0.5	3	0.7	7500	100	0.04/0.004/0.0004	0.3
13	1	800	1	2	0.7	7500	100	0.04/0.004/0.0004	0.3
14	1	1600	1	3	0.3	7500	100	0.04/0.004/0.0004	0.7
15	2	800	1	3	0.7	7500	100	0.02/0.002/0.0002	0.7
16	2	1600	1	2	0.3	7500	100	0.02/0.002/0.0002	0.3
17	1	800	0.5	2	0.3	5000	150	0.02/0.002/0.0002	0.3
18	1	1600	0.5	3	0.7	5000	150	0.02/0.002/0.0002	0.7
19	2	800	0.5	3	0.3	5000	150	0.04/0.004/0.0004	0.7
20	2	1600	0.5	2	0.7	5000	150	0.04/0.004/0.0004	0.3
21	1	800	1	3	0.7	5000	150	0.04/0.004/0.0004	0.3
22	1	1600	1	2	0.3	5000	150	0.04/0.004/0.0004	0.7
23	2	800	1	2	0.7	5000	150	0.02/0.002/0.0002	0.7
24	2	1600	1	3	0.3	5000	150	0.02/0.002/0.0002	0.3
25	1	800	0.5	2	0.7	7500	150	0.04/0.004/0.0004	0.7
26	1	1600	0.5	3	0.3	7500	150	0.04/0.004/0.0004	0.3
27	2	800	0.5	3	0.7	7500	150	0.02/0.002/0.0002	0.3
28	2	1600	0.5	2	0.3	7500	150	0.02/0.002/0.0002	0.7
29	1	800	1	3	0.3	7500	150	0.02/0.002/0.0002	0.7
30	1	1600	1	2	0.7	7500	150	0.02/0.002/0.0002	0.3
31	2	800	1	2	0.3	7500	150	0.04/0.004/0.0004	0.3
32	2	1600	1	3	0.7	7500	150	0.04/0.004/0.0004	0.7

Values of 11 Fixed Parameters

Parameter	Assigned Value
X1	Abilene Topology (Backbone: 11 routers and 14 links; 22 PoP routers; 139 Access routers)
X6	*probNs* = 0.1, *probNsf* = 0.6
X7	*probNr* = 0.6, *probNrf* = 0.2
X10	0 (all users have infinite patience)
X13	*Jon* = 1; *Joff* = 1; *Jx* = 1 (no explicit spatiotemporal congestion)
X14	no long-lived flows
X16	initial *cwnd* = 2 (default Microsoft Windows™ value)
X17	initial *sst* = $2^{31}/2$ (arbitrary large value)
X18	*M* = 200 ms
X19	*MI* = 18,000 (x .2 *M* =) 3600 s
X20	*prON* = 0.25, *prONsecond* = 0.08, *prONthird* = 0.17

Each of the 32 parameter combinations were run against 7 congestion control protocols – requiring 7 x 32 = 224 simulations

Is *MesoNet* Computationally Tractable?

	Experiment #1 (32-bit SLX)	Experiment #2 (64-bit SLX)
CPU hours (224 runs)	**5,857.18**	**94,355.28**
Avg. CPU hours/Run	**26.15**	**421.23**
Min. CPU hours/Run	**12.58**	**203.04**
Max. CPU hours/Run	**43.97**	**739.04**
Avg. Memory Usage (Mbytes)	**196.56**	**2,392.41**

Experiment #1: Required 35 processor weeks

Experiment #2: Required 11 processor years

Parallel simulation of configurations reduced this to:

Experiment #1: 1 week using 48 processors

Experiment #2: 31 days using 48 processors

	Experiment #1 – Slow, Small Network		Experiment #2 – Large, Fast Network	
Statistic	**Flows Completed**	**Data Packets Sent**	**Flows Completed**	**Data Packets Sent**
Avg./Run	**11,466,429**	**3,414,017,482**	**116,317,093**	**33,351,040,358**
Min./Run	**7,258,056**	**2,138,998,764**	**72,944,797**	**21,069,357,409**
Max./Run	**17,390,781**	**5,048,119,166**	**175,947,632**	**50,932,067,100**
Total All Runs	**2,568,480,122**	**764,739,915,978**	**26,055,028,851**	**7,470,633,040,199**

Comparing *MesoNet* with RossNet* Parallel Network Simulator (Throughput/Latency Tradeoff)

MesoNet limited to 1 processor per simulation, using sequential SLX simulator

RossNet can use 2-4 processors per simulation

Simulation Experiment	Event Rate (events/second)
MesoNet Experiment #1 – 32-bit SLX – 1 processor per simulation	**725,359**
MesoNet Experiment #2 – 64-bit SLX – 1 processor per simulation	**439,864**
RossNet Simulation of small topologies – 2 to 4 processors per simulation	**256,244**
RossNet Simulation of AT&T topology – 2 to 4 processors per simulation	**150,720**

RossNet speedups from parallel simulation averaged just under 1.7 (max. 3.2) when using 4 processors

Given 48 processors, *MesoNet* can run 48 simulations in parallel, while RossNet simulations using 4 processors can run only 12 simulations in parallel

RossNet requires a speedup of 4 to equal the throughput of *MesoNet*

If sufficient processors exist to run all RossNet simulations in parallel, then RossNet might provide superior latency to *MesoNet*

*Yaun, G., D. Bauer, H. Bhutada, C. Carothers, M. Yukel and S. Kalyanaraman. 2003. Large-Scale Network Simulation Techniques: Examples of TCP and OSFP Models. In *SIGCOM Computer Communications Review*, 33:3, 27-41.

Conclusions

- Defined a concise TCP/IP model using only 20 parameters
- Showed how the model can be combined with 2-level orthogonal fraction factorial techniques to design efficient experiments
- Demonstrated how to carefully explore a parameter space using parallel instances of a sequential simulator
- Found our model and approach competitive with a parallel TCP/IP simulator, which required additional processors to achieve the same throughput

Related Work

- More Parallel Simulators – (throughput/latency tradeoff)

Riley, G., M. Ammar, F. Fujimoto, A. Park, K. Perumalla and D. Xu. 2004. A Federated Approach to Distributed Network Simulation. In *ACM Transactions on Modeling and Computer Simulation*, 14:2, 116-148.

Zeng. X., R. Bagrodia and M. Gerla. 1998. GloMoSim: a Library for Parallel Simulation of Large-scale Wireless Networks. In Proceedings of the 12th Workshop on Parallel and Distributed Simulations, 154-161.

- Fluid-Flow Simulators – (inaccurate)

Towsley, D., V. Misra and W. Gong. 2000. Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED. In Proceedings of SIGCOMM, 30:4, 151,160.

Yi, Y. and S. Shakkottai. 2007. FluNet: A hybrid internet simulator for fast queue regimes, In Computer Networks: The International Journal of Computer and Telecommunications Networking, 51:18, 4919-4937.

- Hybrid Continuous-Time/Discrete-Event Simulators– (promising)

Lee, J., S. Bohacek, J. Hespanha and K. Obraczka. 2007. Modeling Communication Networks with Hybrid Systems. In IEEE/ACM Transactions on Networking, 15:3, 630-643.