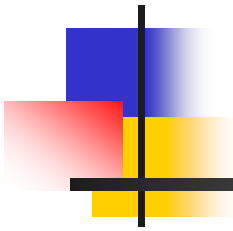
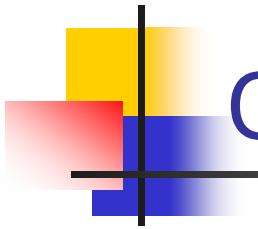


CMPT 886 course project presentation

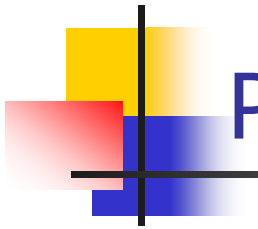
P2VoD: analysis and improvements

Savio Lau
saviol@cs.sfu.ca
September 15, 2004



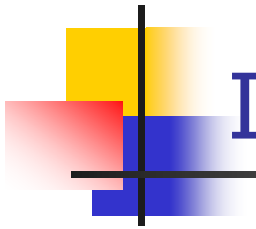
Outline

- Introduction and motivations
- Project goals
- Implementation and design
- Analysis and results
- Future work
- Conclusion



Project introduction

- CMPT 886 project (04-2)
- Based on ICC ’04 paper:
  - P2VoD: Providing Fault Tolerant Video-on-Demand Streaming in Peer-to-Peer Environment by Do, Hua, and Tantaoui



Introduction

- Network I/O bandwidth is VoD's main operational constraint
- Queuing requests results in reneging
- Broadcasting is not always possible
- One solution is to use pipelining, by using client outgoing bandwidth to distribute media
  - Sheu, Hua, and Tavanapong introduced a pipelining idea called Chaining in '97

Chaining

- Chaining:
  - a technique to pipeline the distribution of media in an unicast environment
  - a receiver acts as a server to other receivers down the chain

Chaining illustration

b_d=4

d(a,b)=1

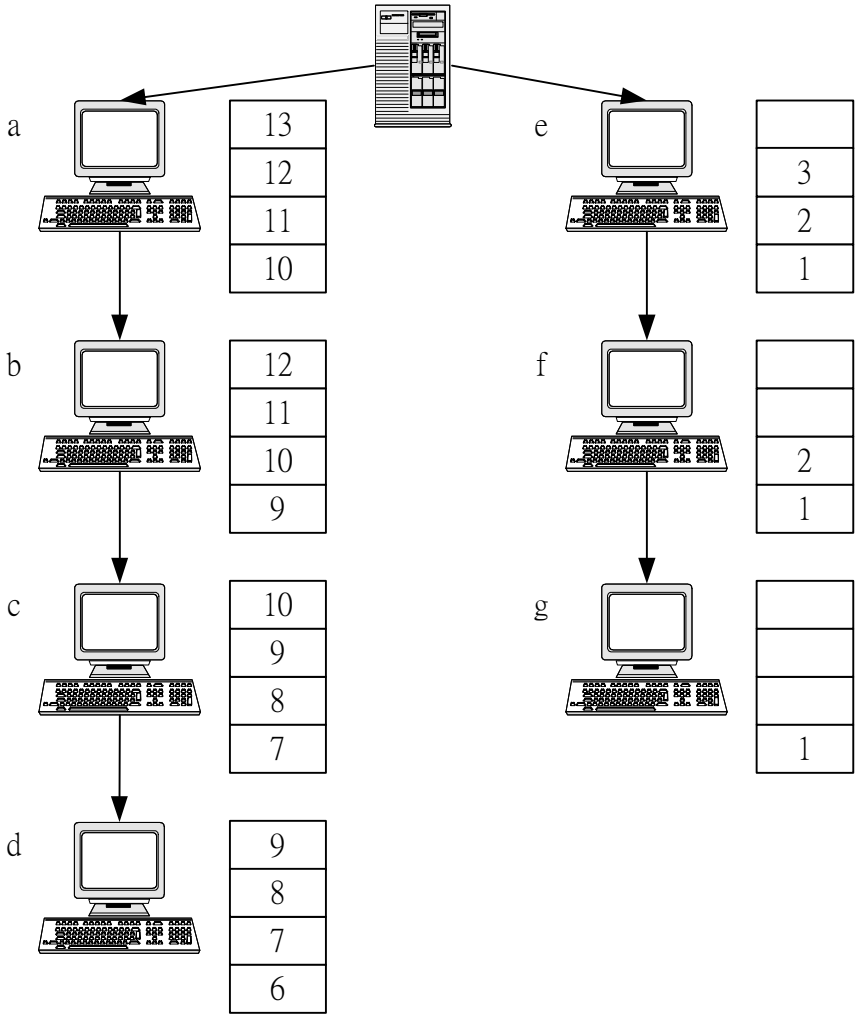
d(b,c)=2

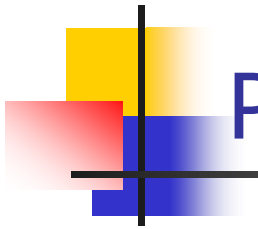
d(c,d)=1

d(d,e)=6

d(e,f)=1

d(f,g)=1





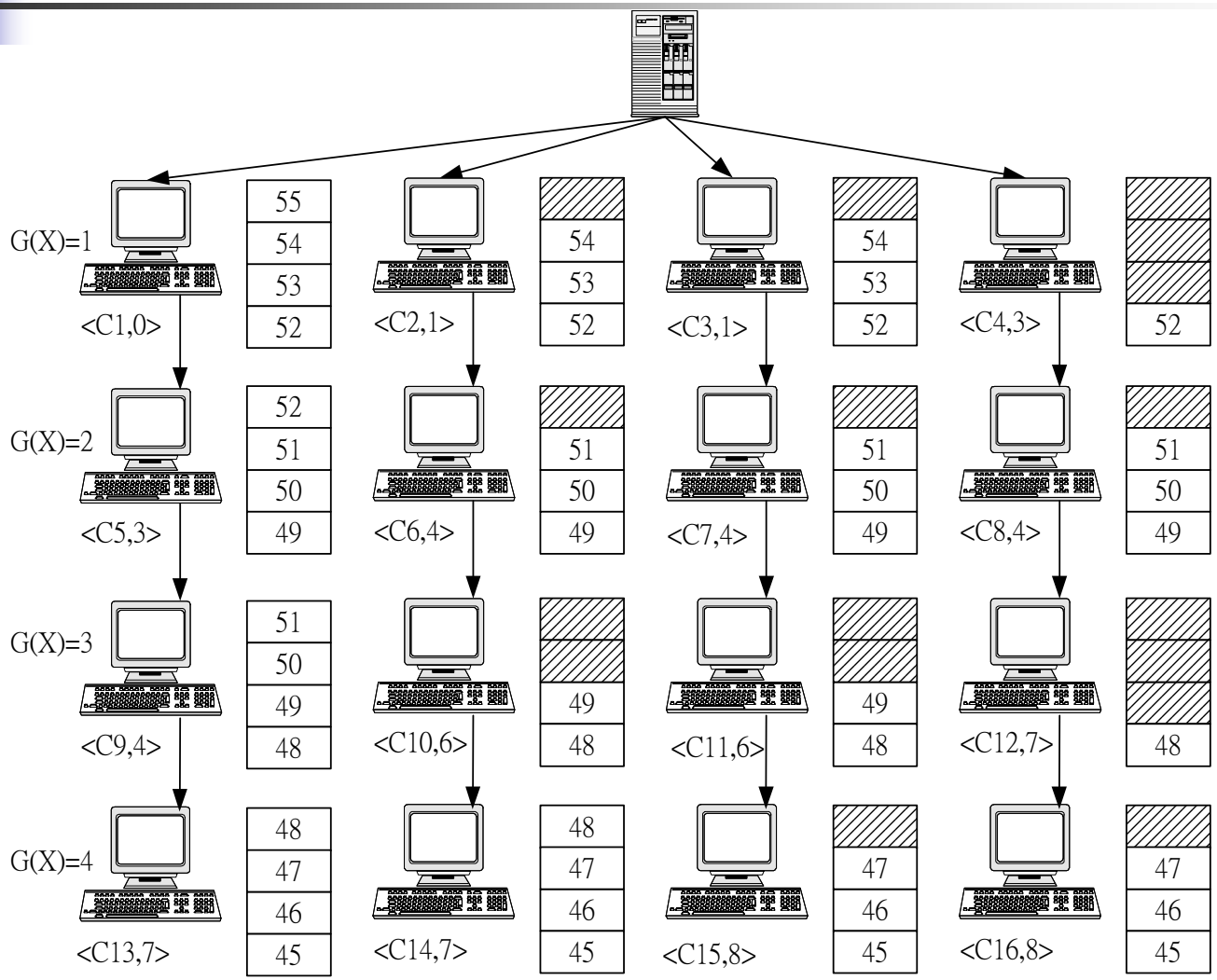
P2VoD

- P2VoD
  - extended the idea of chaining into the concept of generations
  - each generation's peer has the same oldest block cached
  - provides localized recovery
  - quick join: only youngest generation is contacted

P2VoD generations

- Clients in generation G(X)=g will have the same oldest R-block in their cache
- Not all clients use the full size of the cache
  - client cache usage is determined by the time difference between the arrival time of client Y and the arrival time of client X which is the oldest member in a generation
  - $ab_X - ab_Y = t_{jY} - t_{jX}$
- First generation G(X)=1 has at most K members

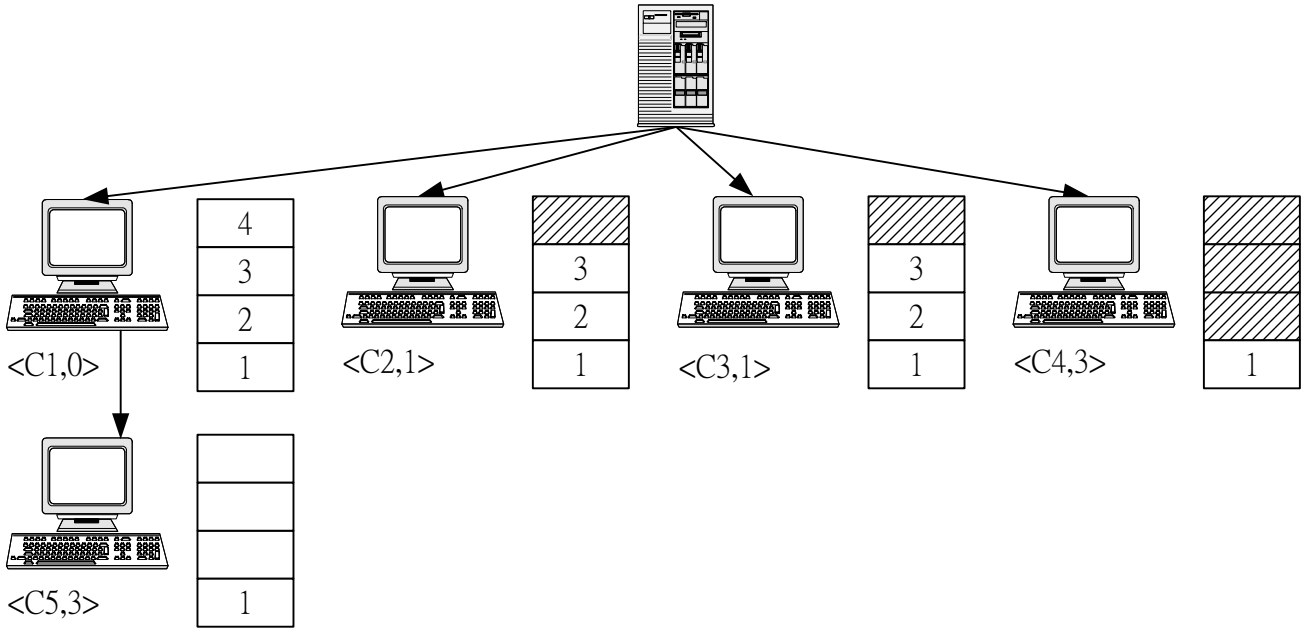
P2VoD illustration



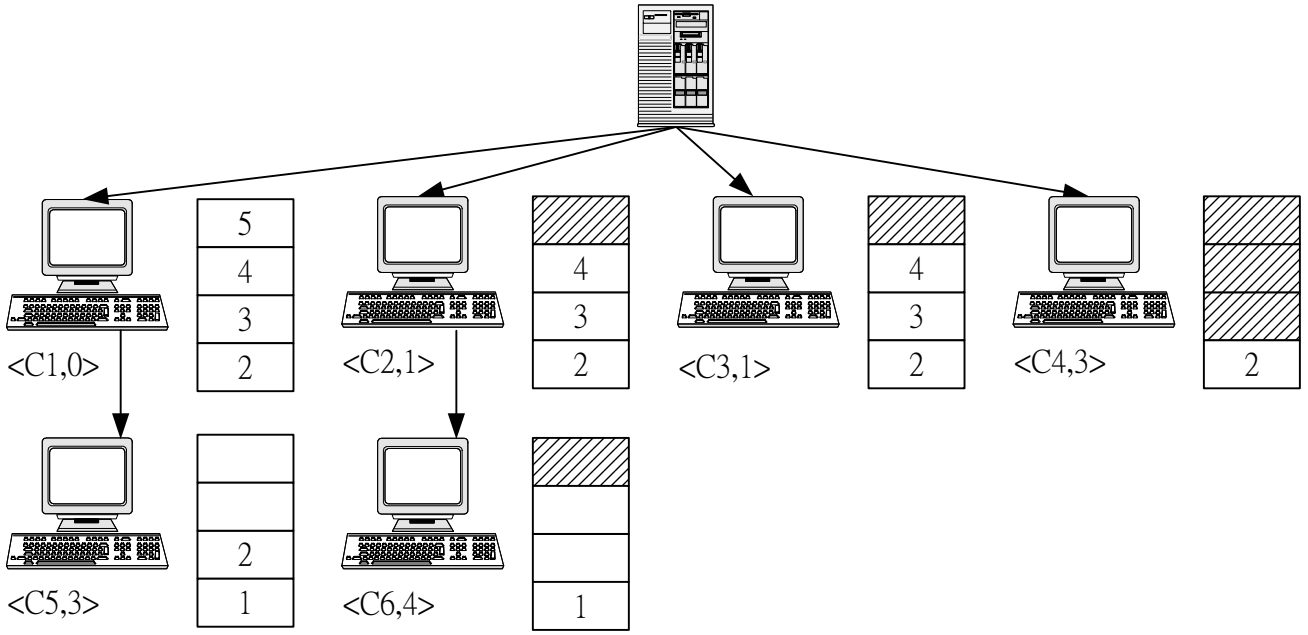
P2VoD joining

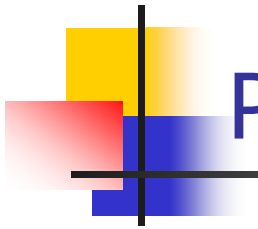
- Parent selection:
  - round robin
  - smallest delay selection
  - smallest distance selection

P2VoD joining



P2VoD joining

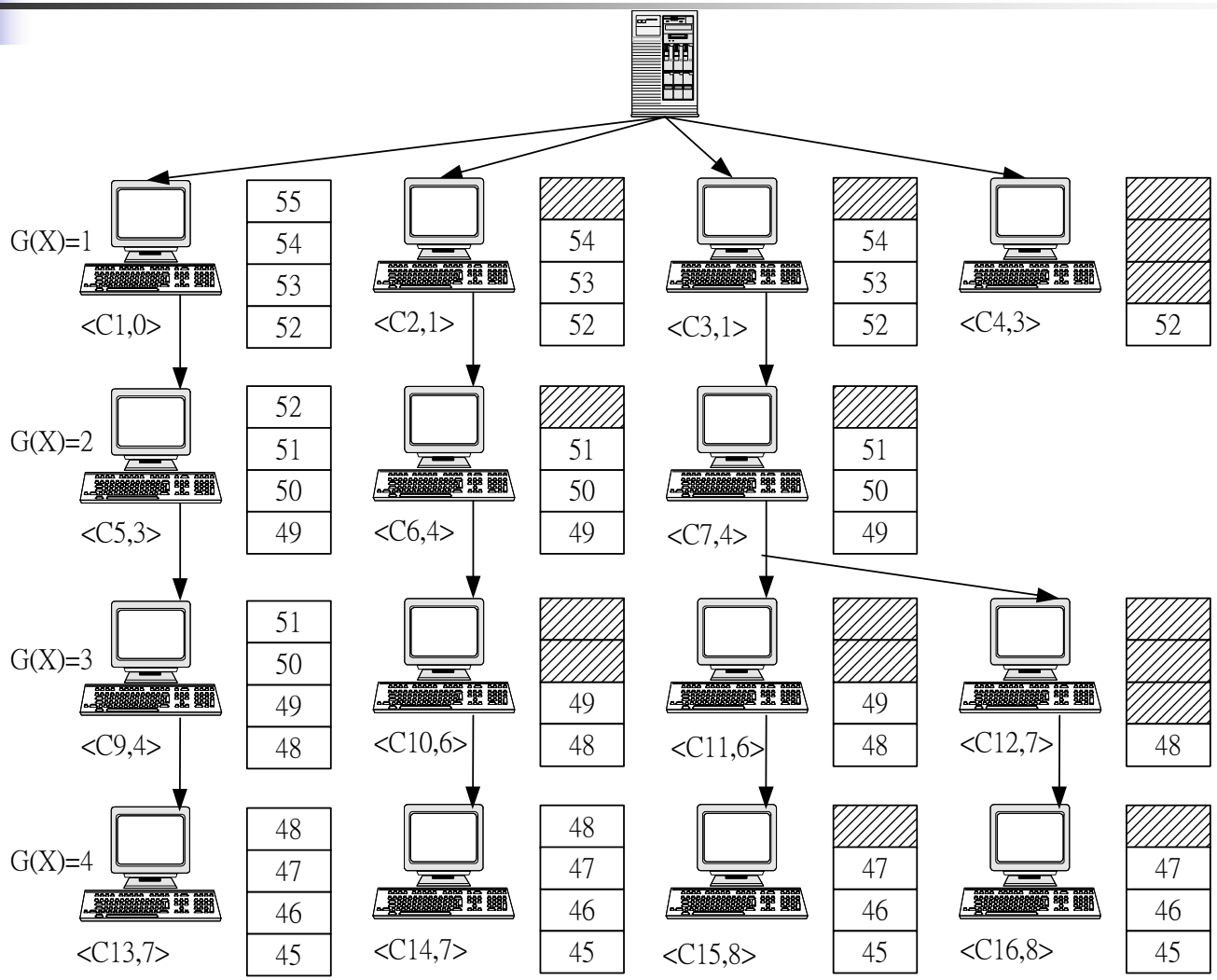


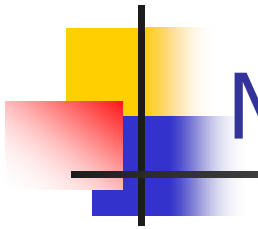


P2VoD failure recovery

- Detect failure through heartbeat
- Recovery starts oldest orphaned client:
  - 1. Orphan client contacts sibling of parents and if one has bandwidth available switchover to that parent. Failed->2
  - 2. Orphan client contacts server and connect directly to server if bandwidth exists. Otherwise session terminated.
  - repeat for each orphaned client

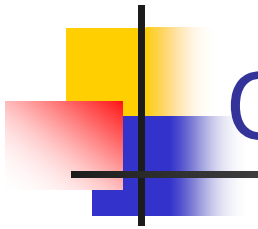
P2VoD failure recovery





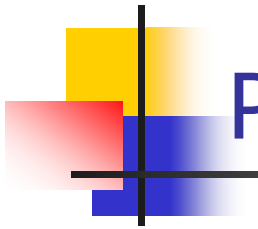
Motivations

- mathematical results are often not achievable in distributed systems like P2VoD
- Simulations provides insights into the performance of a system
- Simulation can quickly illustrate performance of a protocol in different scenarios



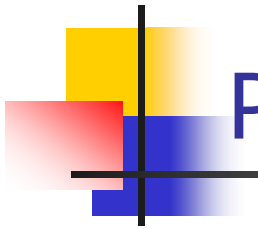
Outline

- Introduction and motivations
- Project goals
- Implementation and design
- Analysis and results
- Future work
- Conclusion



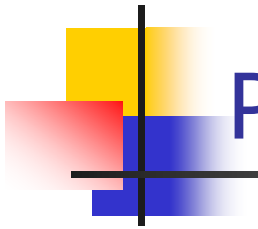
Project goals

- Examine relationship between MB, arrival rate and chain continuity
  - approximating the relationship in the presence of client failures
  - minimum chain continuity is achieved when an arrival exists before the first page is removed from cache
  - find minimum arrival rate for 90, 95 and 99% confidence for given MB



Project goals

- Examine parent selection algorithms
  - Smallest distance, random, round robin:
    - number of possible concurrent clients
    - number of peers contacted with each join algorithm
  - Smallest distance should be best for concurrent clients but most join connects
  - Random and round robin are similar but round robin handles failures better

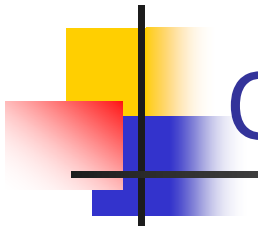


Project goals

- If best-delay joining is used:
  - High arrival rate implies wide generations - results in large number of join requests
  - Limit each generation to L peers, where L>=K

Project goals

- Only generation G(X)=1 has limit of K peers
  - High arrival rate implies full bandwidth usage during growth – no recovery bandwidth
  - Reserve bandwidth for failure recovery
    - reserve bandwidth equal to the failure probability
  - Expects deterministic failure cost upper bound

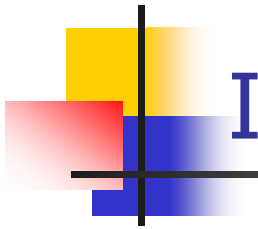


Outline

- Introduction and motivations
- Project goals
- Implementation and design
- Analysis and results
- Future work
- Conclusion

Implementation and design: GT-ITM

- Paper author used GT-ITM topology generator to generate random topology
- GT-ITM is one of the hierarchical network generators used with simulators
  - transit-stub graphs
  - one transit domains (4 nodes), 12 stub domains (96 nodes)
  - ts2ns tool is used to import the GT-ITM topology into ns-2 format

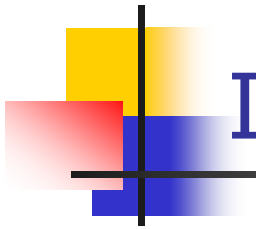


Implementation and design: ns-2

- ns-2 is a open source network simulator used widely in academia
- Programming is done in both C/C++ and Otcl
  - C/C++ are faster than Otcl
  - Otcl does not require recompiling on changes
- For the project, ns-2 release 2.27 is used

Implementation and design: ns-2

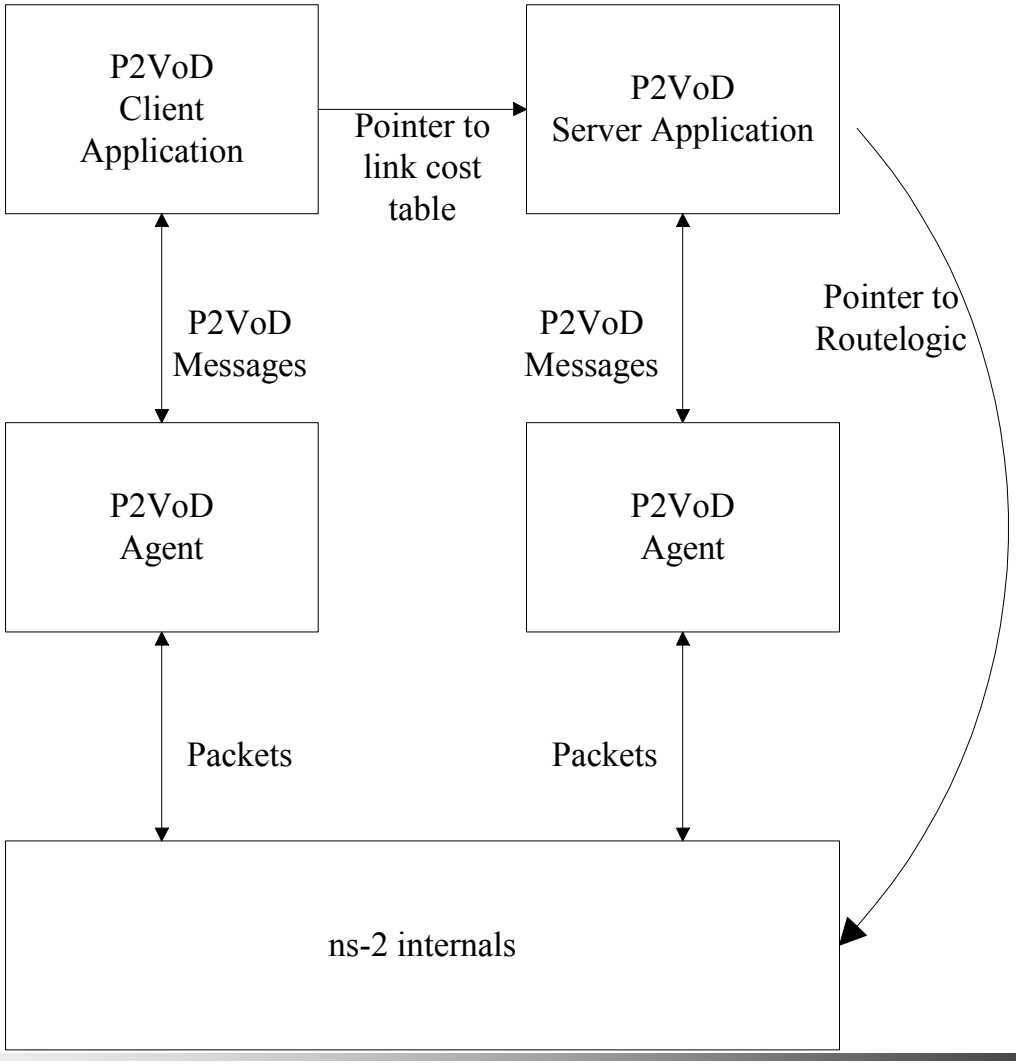
- P2VoD is implemented using 3 classes
  - P2VoD agent:
    - extension of UDP agent responsible for transmitting and receiving messages
  - P2VoD client application:
    - simulating client behavior in joining, failure recovery and updates
  - P2VoD server application
    - maintains list of youngest generation



Implementation and design: ns-2

- Each node is connected to the P2VoD agent
- One of client or server application is connected to the P2VoD agent
- Static routing is used in the topology
- Server applications maintains a link cost table used by clients
- Interarrival times are generated using an external software

Implementation and design: ns-2

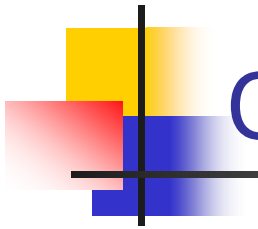


Implementation and design: external C program

- Interarrival times coordination is achieved by generating times external to simulator
  - exponential interarrival times for Poisson distribution are generated and formatted in ns-2 tcl script format
- Based on a given probability, some nodes are failed and failure time is chosen from a uniform distribution

Implementation and design: external C program

- 3 input parameters input:
  - client arrival rate (arrivals per hour)
  - simulation length (seconds)
  - client application failure probability (percentage)
  - “3 14400 5” would indicate 3 arrivals per hours, 4 simulation hours and client application failure probability of 0.05



Outline

- Introduction and motivations
- Project goals
- Implementation and design
- Analysis and results
- Future work
- Conclusion

Chain continuity analysis

- An active P2VoD session needs at least 1 client per generation:

$$P(X>0)=1-\frac{e^{-\lambda}\lambda^{0}}{0!}=1-e^{-\lambda}$$

- Given a certain confidence interval C, and a generation time of MB * movie_length, we arrive at the arrive rate per hour λ′:

$$\lambda'=\frac{\lambda}{MB*movie\_length}=\frac{-\ln(1-C)}{MB*movie\_length}$$

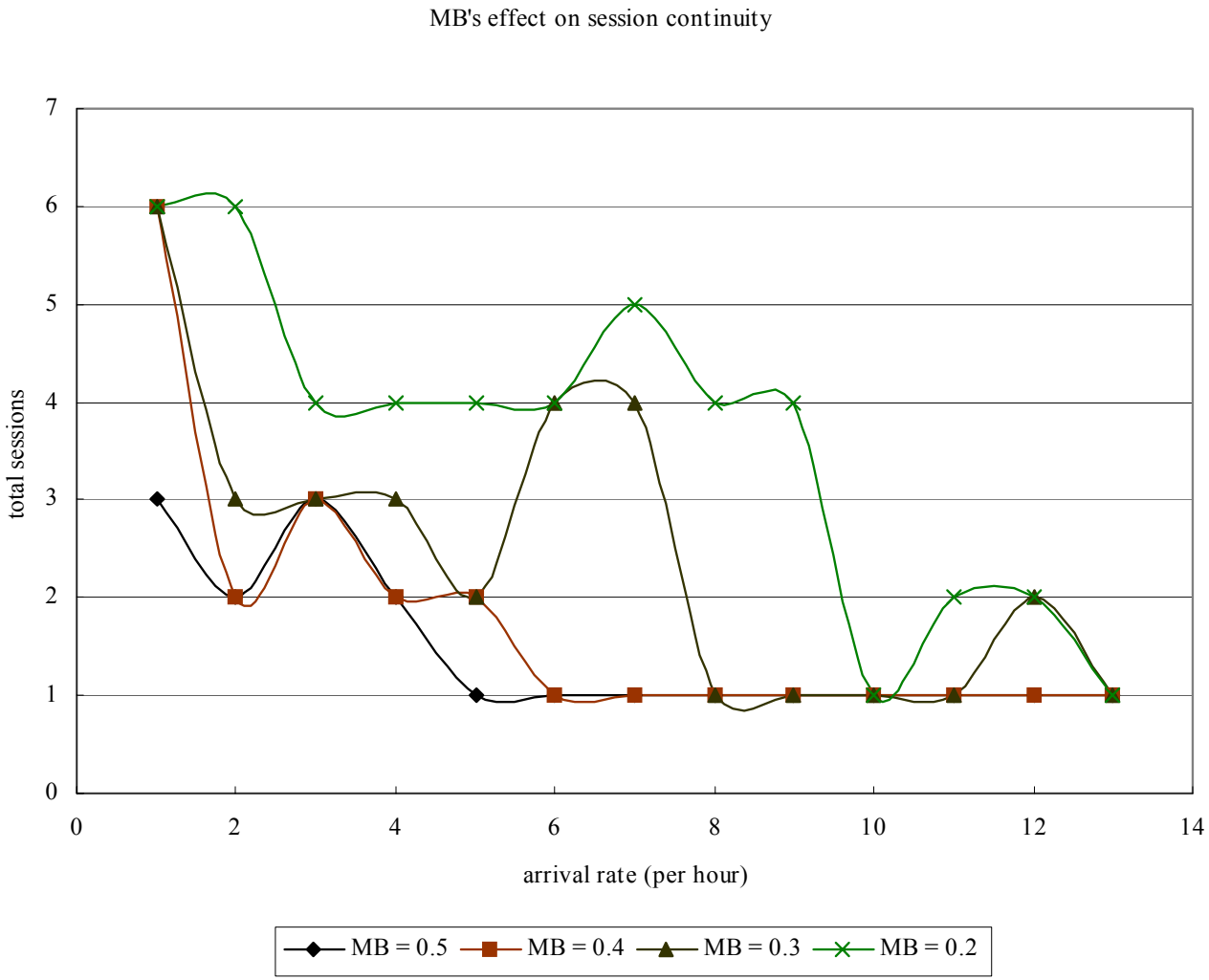
Chain continuity analysis

- The addition of failure probability makes true minimal arrive rate determination difficult.
- An approximation is used where the arrival rate is increased by the factor of the survival rate reciprocal:

$$\lambda'' = \frac{\lambda}{MB * movie\_length}\left(\frac{1}{1-FP}\right) = \frac{-\ln(1-C)}{MB * movie\_length}\left(\frac{1}{1-FP}\right)$$

- λ'' is the arrival rate given certain failure probability FP

Chain continuity results



Chain continuity results

- Effects of failure are not as expected
- Even without additional arrivals, chain continuity is the same for failure probability between 5-20%
- Reason: session can only be broken if
  - a single or more arrival during a generation AND
  - all arrivals fails before video ends

Parent selection results

- Scenarios generated:
  - ~169 arrivals during 4 hour period
  - no application failures
  - each transit link supports 25 concurrent connections
  - each stub link supports 7 concurrent connections

Parent selection results

	Smallest Delay	Round Robin	Random	Smallest Distance
# of failed client joins	34	34	12	0
# of failed connection acks	27	24	0	27

Parent selection results

- Smallest distance has least number of failed joins at the expense of contacting a large number of clients
- Random selection has least number of failed acknowledgments if the network capacity is not full
- Round robin parent selection does not provide large benefits over smallest delay selection

Bandwidth reservation and generation size limit results

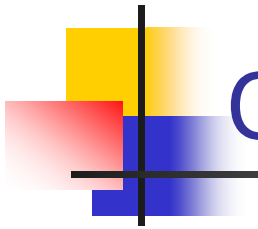
- Given generation size limit L=10 and MB=0.5:
  - smallest distance parent selection needs to contact maximum 10 supergeneration peers at steady state
  - without generation size limit the average number of supergeneration contacts was 42
  - generation size limit has no benefits to smallest delay, round robin nor random parent selection methods

Bandwidth reservation and generation size limit results

- Bandwidth reservation:
  - identical scenario as previous cases with smallest distance parent selection
  - stub node's current connection capacity is reduced from 7 to 3

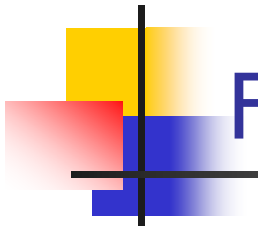
Bandwidth reservation and generation size limit results

- Bandwidth reservation:
  - No noticeable effect until failure probability is greater than 30%
  - Number of orphaned clients during growth phase is reduced by 70% compared to scenario without reservation bandwidth



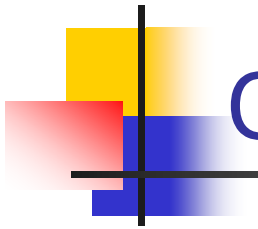
Outline

- Introduction and motivations
- Project goals
- Implementation and design
- Analysis and results
- Future work
- Conclusion



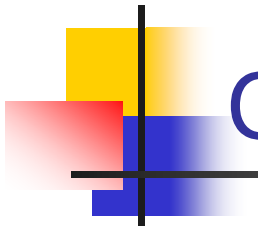
Future work

- Quantify benefits of smallest distance parent selection method in terms of ratio between transit and stub domain size using transit-stub graphs
- Evaluate performance of improved P2VoD in other network topologies



Outline

- Introduction and motivations
- Project goals
- Implementation and design
- Analysis and results
- Future work
- Conclusion



Conclusion

- Implemented P2VoD in ns-2
- Analyzed the minimal requirements for chain continuity
- Evaluated P2VoD given different parent selection methods
- Improved upon the original P2VoD protocol for the growth phase

References

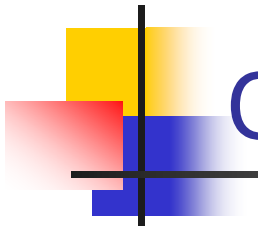
- S. Sheu, K. A. Hua, and W. Tavanapong, "Chaining: a generalized batching technique for video-on- demand," in *Proc. the IEEE Intl. Conf. on Multimedia computing and System*, Ottawa, Canada, June 1997, pp. 110–117.
- S. Sheu, K. A. Hua, and T. Hu, "Virtual batching: a new scheduling technique for video-on-demand servers," in *Proc. Intl. Conf. On Database Systems for Advanced Applications (DASFAA'97)*, Melbourne, Australia, April 1997.
- T. Do, K. A. Hua, and M. Tantaoui, "P2VoD:

References

- A. Dan, D. Sitaram, and P. Shahabuddin, "Dynamic batching policies for an on-demand video server," *Multimedia Systems*, vol 4. pp. 112–121, Nov. 1994.
- C. C. Aggarwal, J. L. Wolf, and P. S. Yu, "On optimal batching policies for video-on-demand storage servers," in Proc. *IEEE Intl. Conf. on Multimedia Computing and Systems*, Hiroshima, Japan, June 1996, pp.253–258.
- Y. Guo, K. Suh, J. Kurose, and D. Towsley, "P2Cast: P2P patching scheme for VoD

References

- ns-2 [Online]. Available: http://www.isi.edu/nsnam/ns.
- K. Fall, K. Varadhan, The ns manual (formerly ns notes and documentation) [Online]. Available: http://www.isi.edu/nsnam/ns/doc/index.html.
- E.W. Zegura, K. Calvert, S. Bhattacharjee, "How to model an Internetwork," in *Proc. IEEE INFOCOM*, San Francisco, CA, Mar. 1996, vol. 2, pp. 594–602.



Questions?