SWINBURNE UNIVERSITY OF TECHNOLOGY

Further Optimising Online FPS Game Server Discovery

Christopher Leong

Overview

- What is game server discovery?
- Current discovery process
- Previously proposed discovery process[1]
- Further improvements
- Conclusion

[1] G. Armitage, Optimising Online FPS Game Server Discovery through Clustering Servers by Origin Autonomous System. ACM NOSSDAV 2008, May 2008.

Game server discovery

- Generating a list of available game servers
- Players use a 'game browser' to find a suitable server in the list to play on
- 'Phase 1'
  - Query master server for list of addresses of game servers
- 'Phase 2'
  - Query game servers returned by master server sequentially

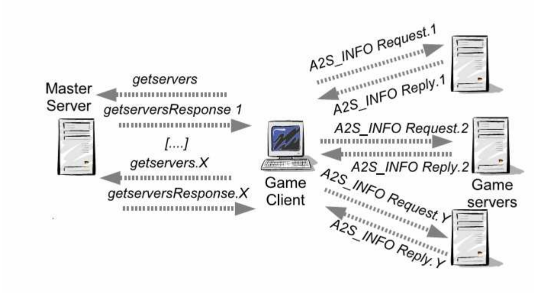
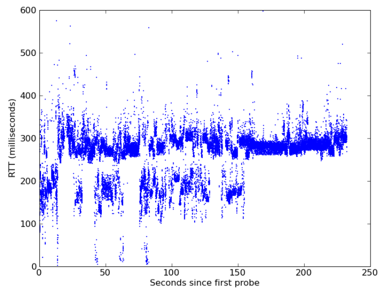


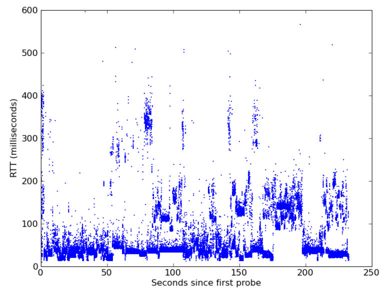
Image source: G. Armitage, Optimising Online FPS Game Server Discovery through Clustering Servers by Origin Autonomous System. ACM NOSSDAV 2008, May 2008.

What's wrong with the current process?

- Slow
  - Nearly 4 minutes to complete on 'DSL>256k' connection
- Resource intensive
  - ~5MB network traffic generated to effectively join a single server
- 'Brute force'
  - Probes all servers regardless of potential latency to player

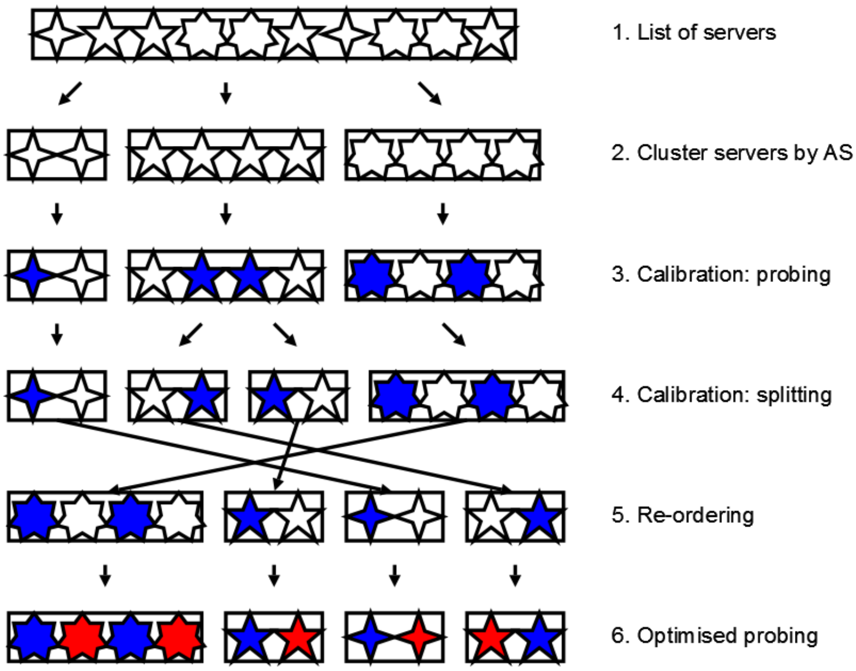
Illustration of the issue



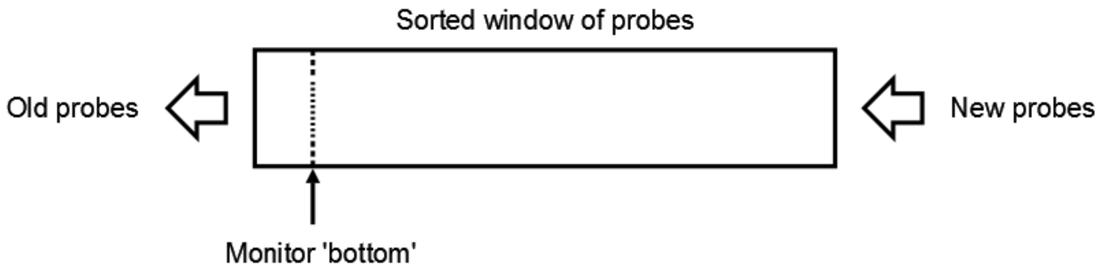


- Japan (left) is **distant** to many low RTT servers
- UK (right) is **close** to many low RTT servers
- RTT fluctuates across entire discovery period
- Previously proposed solution aims to
  - Probe low RTT servers before high RTT servers (Algorithm 1)
  - Provide automatic early termination for a given RTT threshold (Algorithm 2)

Algorithm 1: 'Re-ordering'

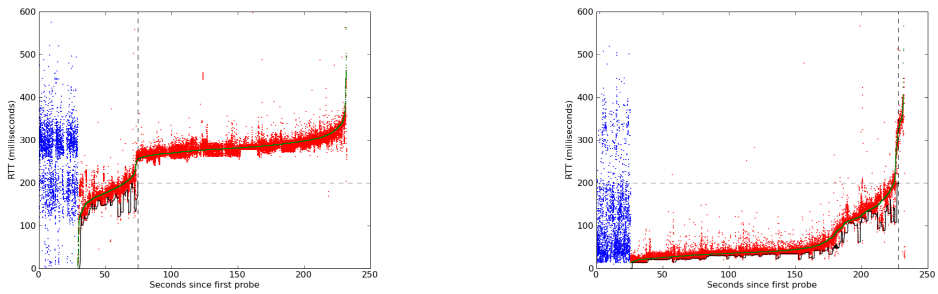


Algorithm 2: 'Auto-stop'



- When 'bottom' > maximum tolerated latency (RTT_{stop}), stop discovery process
- Why? Re-ordering not perfect

Illustration of Algorithms 1 & 2



- Illustrates previously proposed discovery algorithms (benchmark)
- Further improvements?
  - Alternative sub-clustering algorithm
  - Alternative choices in the number of calibration probes

Alternative sub-clustering algorithm

- Existing
  - Split into **/16 subnets** if the variance within an AS cluster is large
- Proposed
  - Split using **dynamic network prefix** choice if the variance within an AS cluster is large

Algorithm: ‘Alternative sub-clustering’

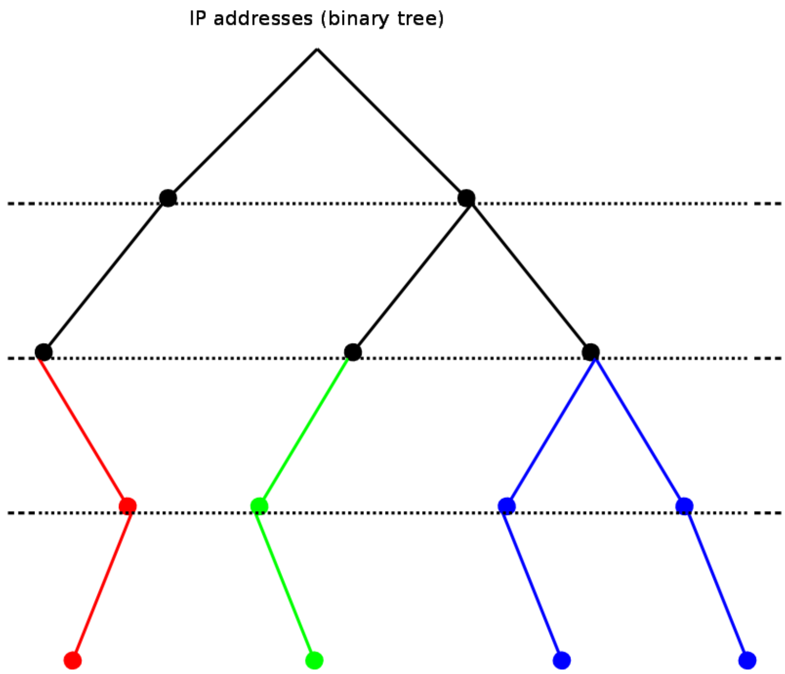
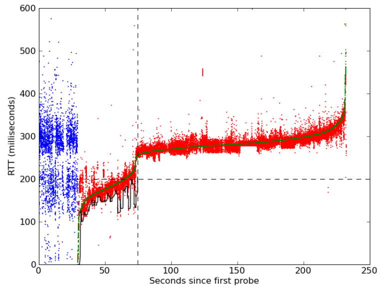
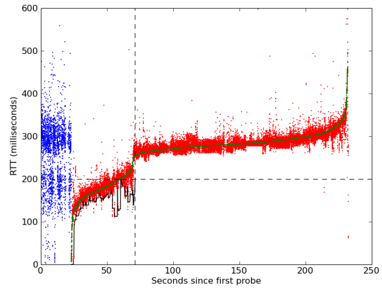


Illustration of alternative sub-clustering





- Japanese client using existing (left) and alternative (right) sub-clustering algorithms
- **Effect by itself on discovery time/traffic appears marginal**

Alternative number of calibration probes

- Existing
  - $N_{sample} = \sqrt{N_{cluster}}$
  - N_{sample} = Number of calibration probes
  - $N_{cluster}$ = Number of servers in the AS cluster
- Proposed
  - Scaling $N_{cluster}$
  - Prioritised sampling

Method 1: Scaling $N_{cluster}$

N_{sample}	Calibration probes	Autostop (time and % worst case)		% all probes <RTT_{stop} found
$(N_{cluster})^{0.5}$	3185	74.5s	32.1%	100.0%
$(N_{cluster}/2)^{0.5}$	2391	70.9s	30.5%	99.9%
$(N_{cluster}/4)^{0.5}$	1908	68.3s	29.4%	99.7%
$(N_{cluster}/8)^{0.5}$	1623	66.1s	28.5%	99.6%

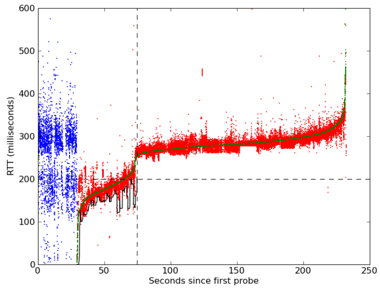
- **Negligible trade-off in 'playable' servers found for relatively large reductions in time (and hence network traffic too)**

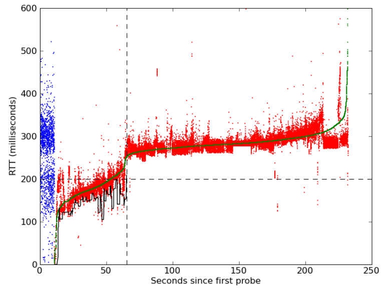
Method 2: Prioritised sampling

- Concentrate calibration resources on larger, possibly more diverse AS clusters → Use N_{sample} probes
- Servers within smaller AS clusters less likely to exhibit large variations → Use single probe

N_{sample}	Calibration probes	Autostop (time and % worst case)		% all probes <RTT_{stop} found
$(N_{cluster})^{0.5}$	1911	68.6s	29.5%	99.7%
$(N_{cluster}/2)^{0.5}$	1680	67.1s	28.9%	99.7%
$(N_{cluster}/4)^{0.5}$	1524	66.1s	28.5%	99.7%
$(N_{cluster}/8)^{0.5}$	1413	65.3s	28.1%	99.6%

Issue with using fewer calibration probes





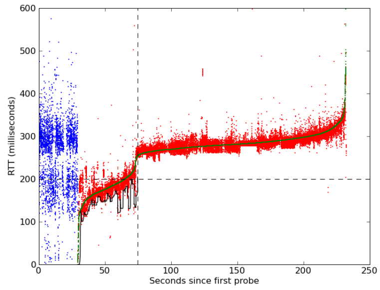
- Japanese client using original (left) choice of N_{sample} and using prioritised sampling (right)
- **Undesired effect of poorer re-ordering (not obvious from previous tables)**

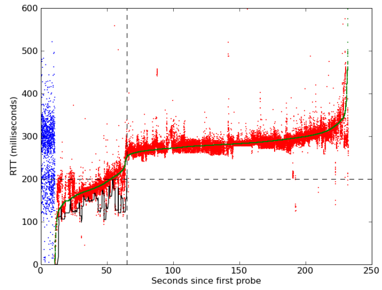
Combined optimisations

Algorithm	Calibration probes	Autostop (time and % worst case)		% all probes <RTT_{stop} found
Unmodified	3185	74.5s	32.1%	100%
Modified	1362	65.1s	28.1%	99.6%
1 probe per AS	1176	55.8s	24.0%	84.8%

- ‘Modified’ combines both alternative sub-clustering and alternative choice of N_{sample}
  - 43% of original calibration probes (15% more than 1 per AS)
  - 13% less time than original
  - Negligible loss in ‘playable’ servers found

Illustration of combined optimisations





- Japanese client using original ‘unmodified’ algorithm (left) and ‘modified’ algorithm (right)
- Reasonable re-ordering fidelity

Conclusions

- Alternative choice in sub-clustering can improve re-ordering of servers
  - Marginal effects on time/traffic for discovery process
- Alternative choice in N_{sample} can reduce time/traffic for discovery process
  - Undesired effect of poorer re-ordering
- Alternatives complement each other’s weaknesses
  - Appreciable reductions in time/traffic at negligible cost to accuracy

Thanks

- Grenville Armitage
- Jason But
- Philip Branch