# CCOF[1] and CompuP2P[2]

slides by Gary Jackson

1 D. Zhou and V. Lo Cluster. **Computing on the Fly: Resource Discovery in a Cycle Sharing Peer-to-Peer System** In *Proceedings of IEEE International Symposium on Cluster Computing and the Grid (CCGrid)*, April 2004.

2 R. Gupta, V. Sekhri, and A. K. Somani. **CompuP2P: An Architecture for Internet Computing Using Peer-to-Peer Networks** *IEEE Transactions on Parallel and Distributed Systems*, Vol. 17, No. 11, November 2006.

# CCOF: Claim

- Need for resource discovery for cycle-sharing in P2P
- Why not DHT?
  - Resource discovery cannot be hashed
  - So CCOF uses an unstructured topology and various flooding mechanisms instead
- Idea: compare a number of discovery mechanisms on a simulated network

# Dynamic Hosts

- Peers leave & join
- Can withdraw access, but still route
- Peer might crash and disappear
- Migration acceptable, but needs care
- Attempt to model real-world availability and usage

# Search Algorithms

- Centralized Search: ideal (?) for comparison
- Expanding Ring: breadth-first flood
- Random Walk: depth-first flood
- Advertisement: flood with capability instead of request
- Rendezvous Point: centralized collection
  - Optimal placement is an open problem

# Scheduling Strategies

- Reschedule at night
- Give up

# Power-Law Topology

- $f(d) \propto d^{\eta}$
  - d is the degree of a node
  - f(d) is the fraction of nodes of degree d
  - $\eta$ is a constant, measured to be -1.4 in Gnutella

# Parameters

- Random Job Size
- Variable client/donor ratio: 0.1 or 0.7
- Variable workload distributions: uniform or normal
- Variable withdrawal probability
- Cherry-picked search parameters

# Results

- Measured versus withdrawal probability:
  - Job Completion Rate
  - First submission success rate
  - Migration Failure Rate
  - Message Overhead
  - Average Distance

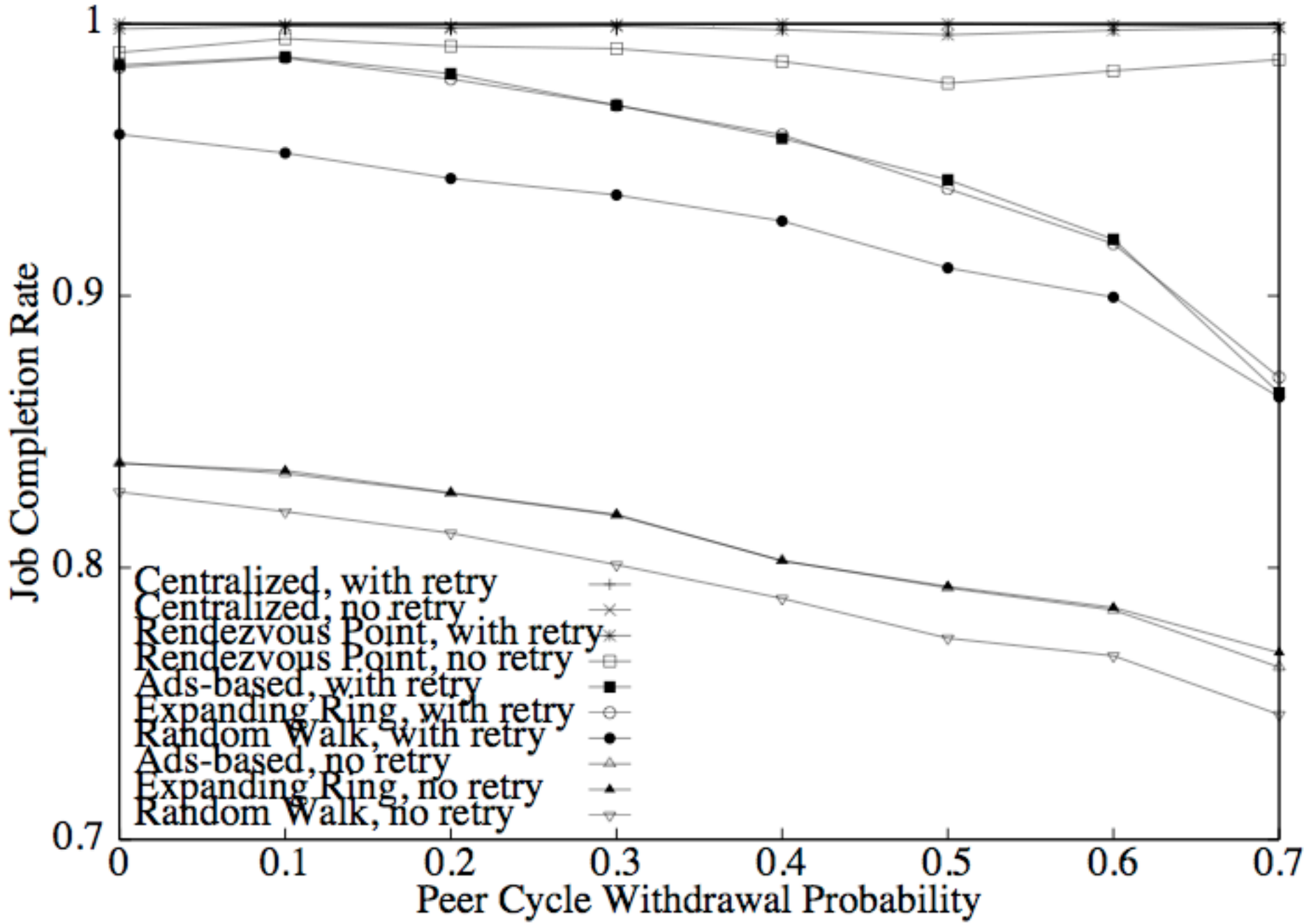


Fig. 2. Job completion rate under uniform workload, when the ratio of clients to donors is 0.1.

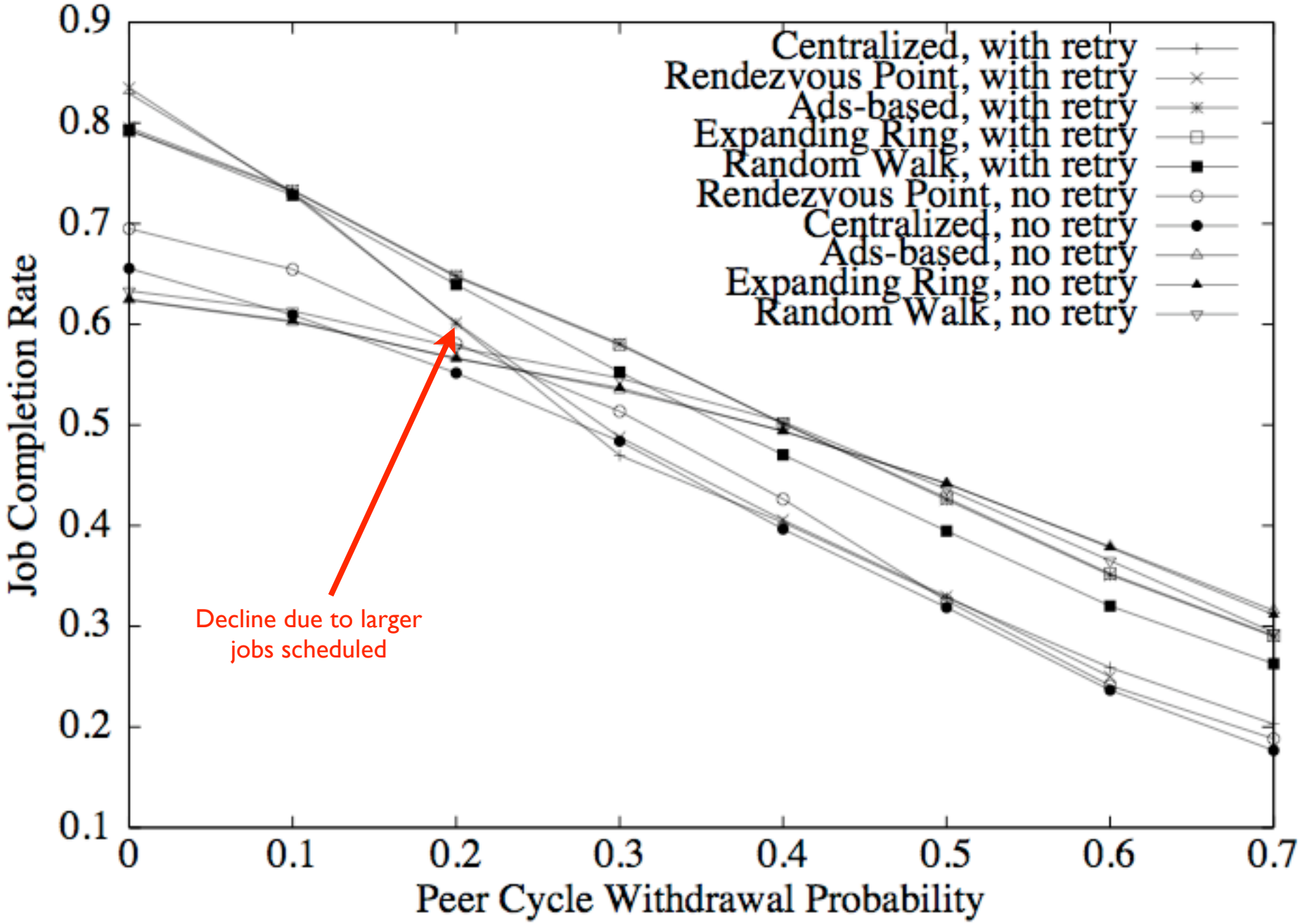


Fig. 5. Job completion rate under uniform workload, when the ratio of clients to donors is 0.7.

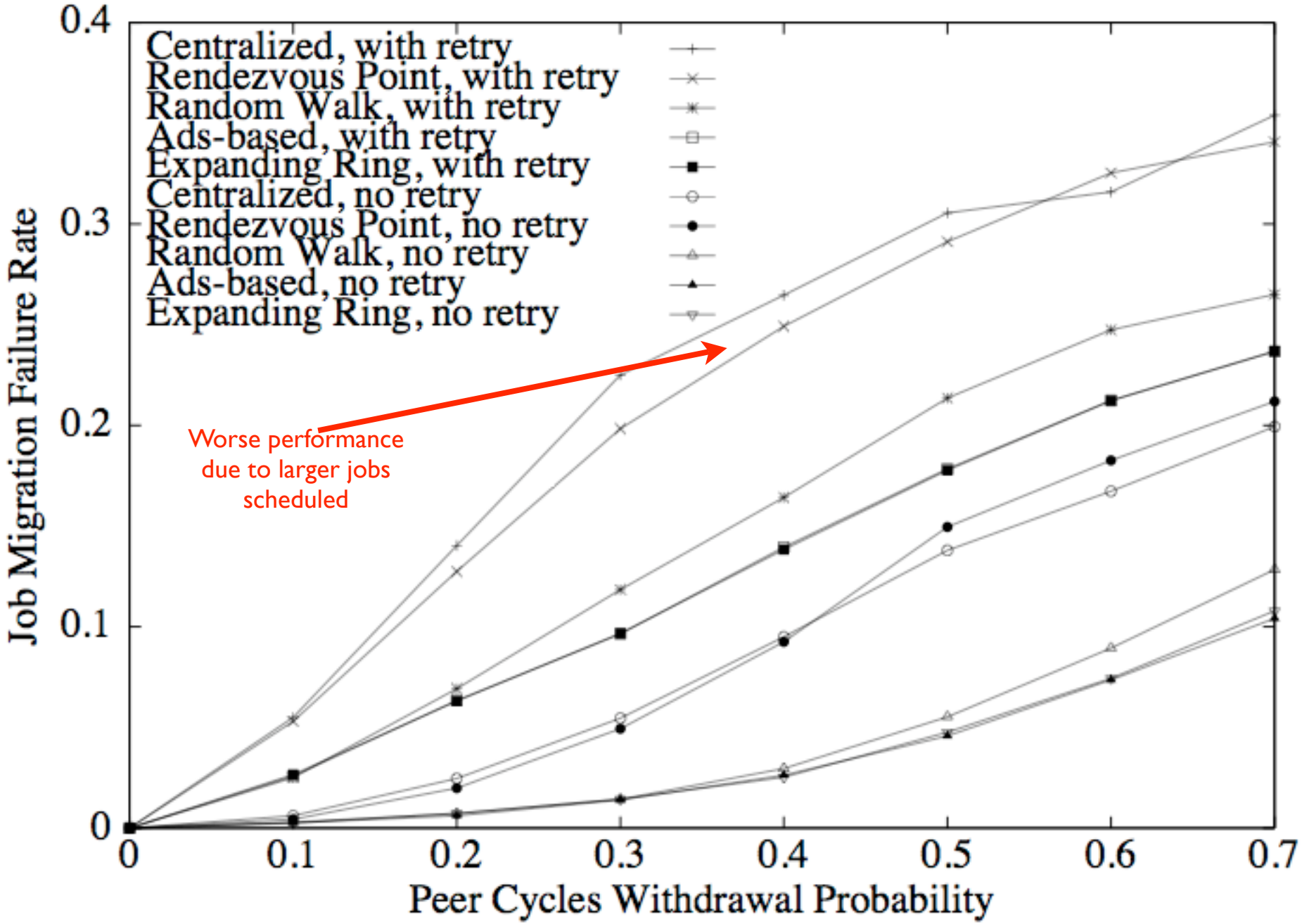


Fig. 7. Job migration failure rate under uniform workload, when the ratio of clients to donors is 0.7

# Conclusion

- Rendezvous Point works the best with low load
- All do about the same as Centralized under heavy load
  - But large jobs seem to be a problem ...
  - Maybe because the other schemes are incapable of scheduling large jobs
- Message overhead is low (numbers per 24h)

# CompuP2P

- Claim:
  - Grid and Public Resource are insufficient for exploiting all available resources
  - Need an architecture for resource sharing on P2P networks

# Idea

- Market economy based system for arbitrating access to resources
  - Market Owners (MO) do the arbitration
  - Basic commodities:
    - C cycles/sec
    - G gigabytes of storage

# Single Overlay Scheme

- C *is* the Chord ID
- Flaw: if C is unevenly distributed, MOs will not be evenly distributed

# Processor Overlay Scheme

- Hash C in to the regular Chord Space
- Then form another overlay of MO nodes in the C space
- MO nodes broadcast themselves to others

# Fixed List Pricing

- MO always gets the same cut
- Reverse Vickrey Auction
  - Lowest bid wins at next-to-lowest price
  - Service profit = next-to-lowest bid - lowest bid

# Variable List Pricing

$$\text{Payoff}_{\text{MO}} = (MC'_S - MC'_1)/(MC'_S)^2 + \delta$$

$$\text{Payoff}_{\text{seller}} = MC'_1 + 1.$$

- Actually, hybrid pricing scheme
- MO payed more for lower cost services to ensure lowest bid wins
- Properties:
  - Payoff covers cost of service
  - Total cost to buyer is bounded

# Prototype Implementation

- XML job description
- KARMA protocol to handle currency exchange
  - Each node is given a starting amount
  - *What if clients use up all their money and there is excess work to do?*

# Handling Failure

- MO Failure: sellers relist, clients go elsewhere
- Seller Failure: MO checks seller, and purges dead ones, clients don't pay for unfinished work
- Client Failure: Not a big deal until delivery, then seller hangs on to results for short time
  - Does the client still pay?

# Applications

- ISTOS: Simulations of optic networks
- Traveling Salesman Problem: Parallel approximation

# Conclusion

- Real structured P2P system for resource arbitration
- Using MOs creates a hierarchy, but this can heal using the normal mechanism for taking over failed nodes
- CompuP2P is pretty cool!

# Questions

- In CCOF, do clients know the times they will be idle?
  - Yes, based on their profile (§III.B)
- How does Expanding Ring work?
  - Basically, make a broadcast to all one-hop neighbors, who make a broadcast to their neighbors, and so forth, up to the radius of the ring

# Questions

- Can Chord handle the type of churn in a manner that is good enough for resource sharing, as they pointed out in the previous paper?
  - I don't think the problem was nodes going offline -- it was nodes withdrawing resources

# Questions

- What's the difference between Rendezvous Point and MO?
  - MOs are chosen automatically
  - MOs have a sophisticated scheme for matching resources
  - Rendezvous Nodes should be placed based on network topology

# Questions

- How are MOs compensated?
  - Fixed compensation, or compensation based on finding the lowest bid in the variable list pricing scheme
- Do both overlay schemes use MOs?
  - Yes. C is the node id in Single, and hash (C) is the node did in Processor

# Questions

- How do sellers profit?
  - Fixed list: their profit is the difference between their low bid and the next highest bid
  - Variable list: low bid + 1