Antfarm: Offense

Aaron Ballew and Patrick Wong
User Behavior

- P2P users like *anonymity*
- Tokens are unique IDs, time-stamped, and associated with specific data
- Even if Antfarm performs better, users may still not use it
- (Pg 112): Assumption that receivers are non-malicious...
  - Why not assume the same with senders and have them issue their own tokens?
  - Related note: no incentive for receivers to pay up the tokens (since it's only administered post-transfer)
Goals

• Introduced the problem as a minimization of average delay

• Yet all the work shows Antfarm maximizing average throughput. Assumes direct correlation!

• Optimizing to the average does not prevent high variance in the treatment of peers.

• Paper washes over enforcing minimum service
Goals (cont’d)

• System is dependent on the coordinator

• What if a network problem interferes with token delivery?

• Unfair bias against a peer who otherwise has a good connection with other peers or the seeder

• Clarification: **Measured** max average throughput vs. **True** minimum average delay
Art of simplification

- Linear optimization is much simpler than non-linear
- Figure 1: real-world plot not monotonic - but the local minimas/maximas are dismissed as “artifacts”!
- The reasoning seems intuitive, but not conclusively proven.
Robust?

- Bandwidth allocation calculations every 5 minutes, or if 10% change of area under the curve
- 10% change? Is an oscillation of ±10% for 5 minutes acceptable?
- >10% change, rapid oscillations - triggers frequent reallocations. Can the algorithm keep up?
- Most other network feedback problems are very sensitive to time-scale!
Awesome results though!

- Duh.
- Obviously, taking a p2p network and adding new content servers seeders increase “aggregate bandwidth.”
- Exactly how much of the improvement is attributed to new network resources vs. efficiency of Antfarm?
- Keep in mind: efficiency is the contribution.
A closer look at efficiency...

• Is the improvement due to efficiency optimal?

• Unlikely due to arbitrarily defined constants:
  • Token expiration: 5 minutes
  • 30 tokens/peer
  • Data points flush after 30 minutes
  • Bandwidth allocations every 5 minutes
Conclusion

- Results had potential, ended up only obvious.
- Some unrealistic expectations about users’ behavior (cooperation, anonymity)
- Too optimistic about feedback details