EXPLOITING IDLENESS IN MULTI-TIER DATA CENTERS
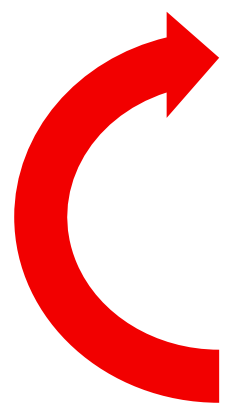
Timothy Zhu, Anshul Gandhi, Mor Harchol-Balter, Mike Kozuch* (CMU, *Intel)

OVERVIEW

- Power is an expensive resource for data centers
- Data center workloads are time-varying and unpredictable
- Solution: dynamic capacity management
  - Stateless servers (app) can be dynamically provisioned
  - Stateful servers (mem) are always left on
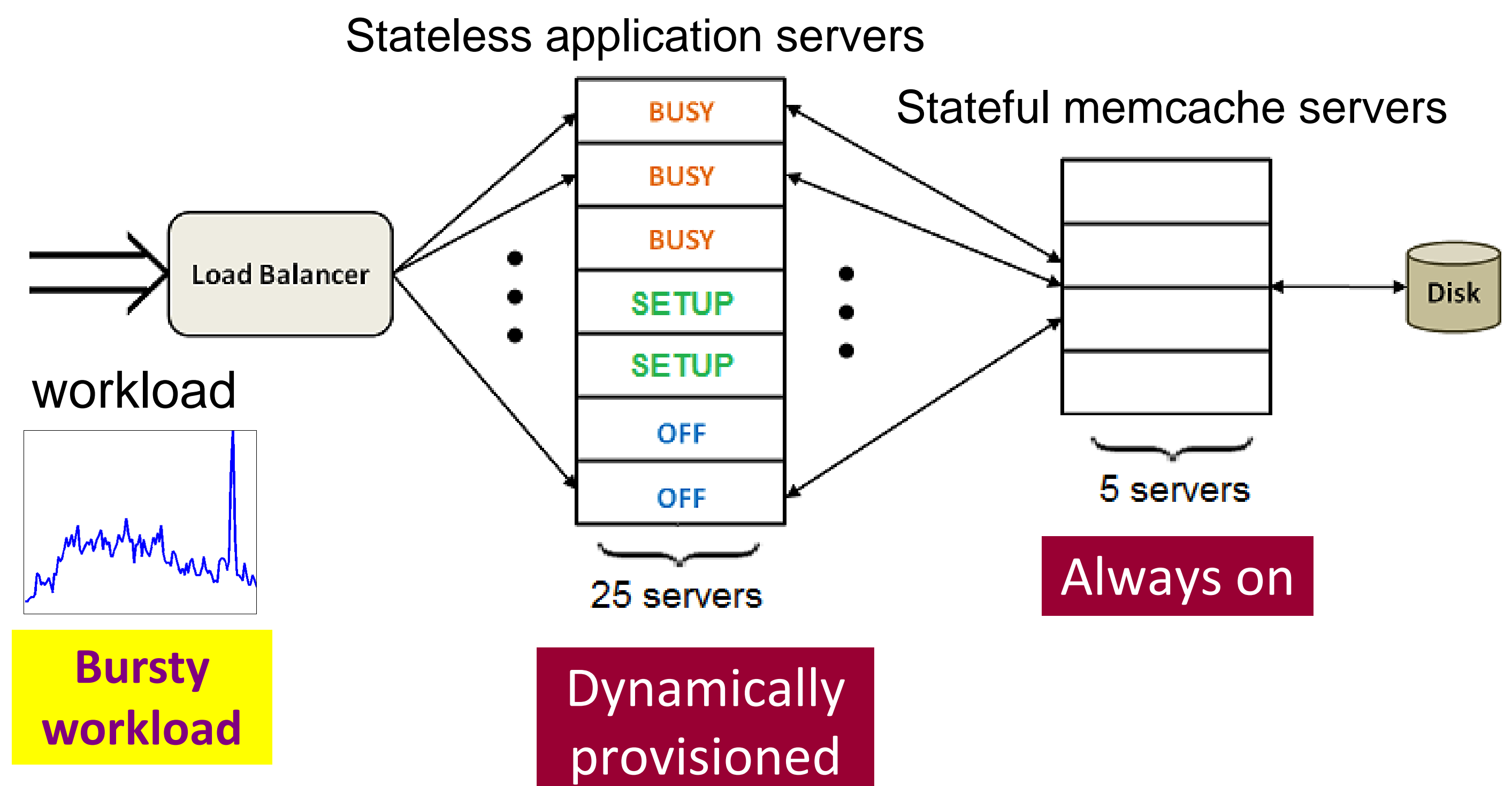 - However, provisioning is not free

Setup cost
260 s
200 W
(+more)

- ❑ BUSY server: 200 Watts
- ❑ IDLE server: 140 Watts
- ❑ OFF server: 0 Watts

IMPLEMENTATION TESTBED

Stateless application servers

Stateful memcache servers

workload

Bursty workload

25 servers — **Dynamically provisioned**

5 servers — **Always on**
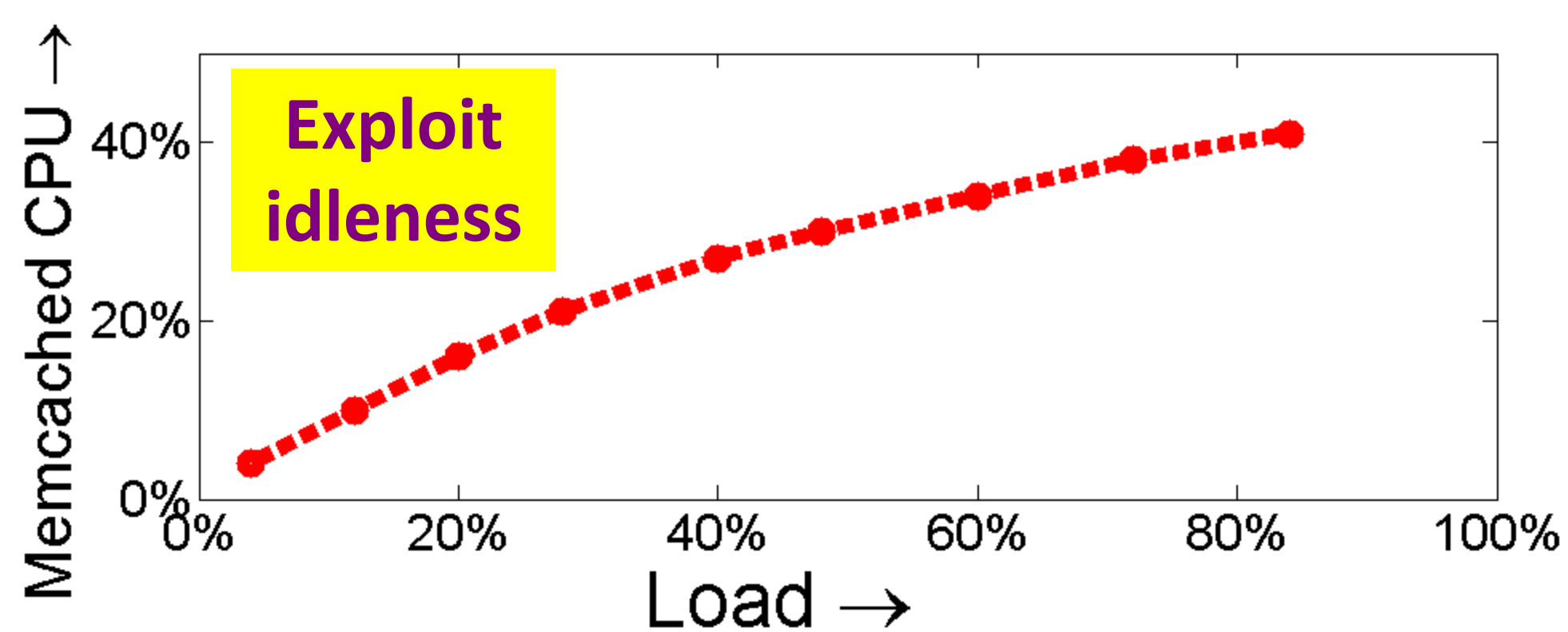
PROBLEM STATEMENT

- SLA: 95%-tile response time guarantees
  - Typically around 500ms (<< setup time)
- Workload fluctuations can't be handled by app servers
- Example: flash crowds, heavy-tailed jobs, service outages
- Obvious solution: maintain (costly) spare app capacity
- Anything else that can be done?

KEY IDEA: USE ALWAYS ON STATEFUL SERVERS

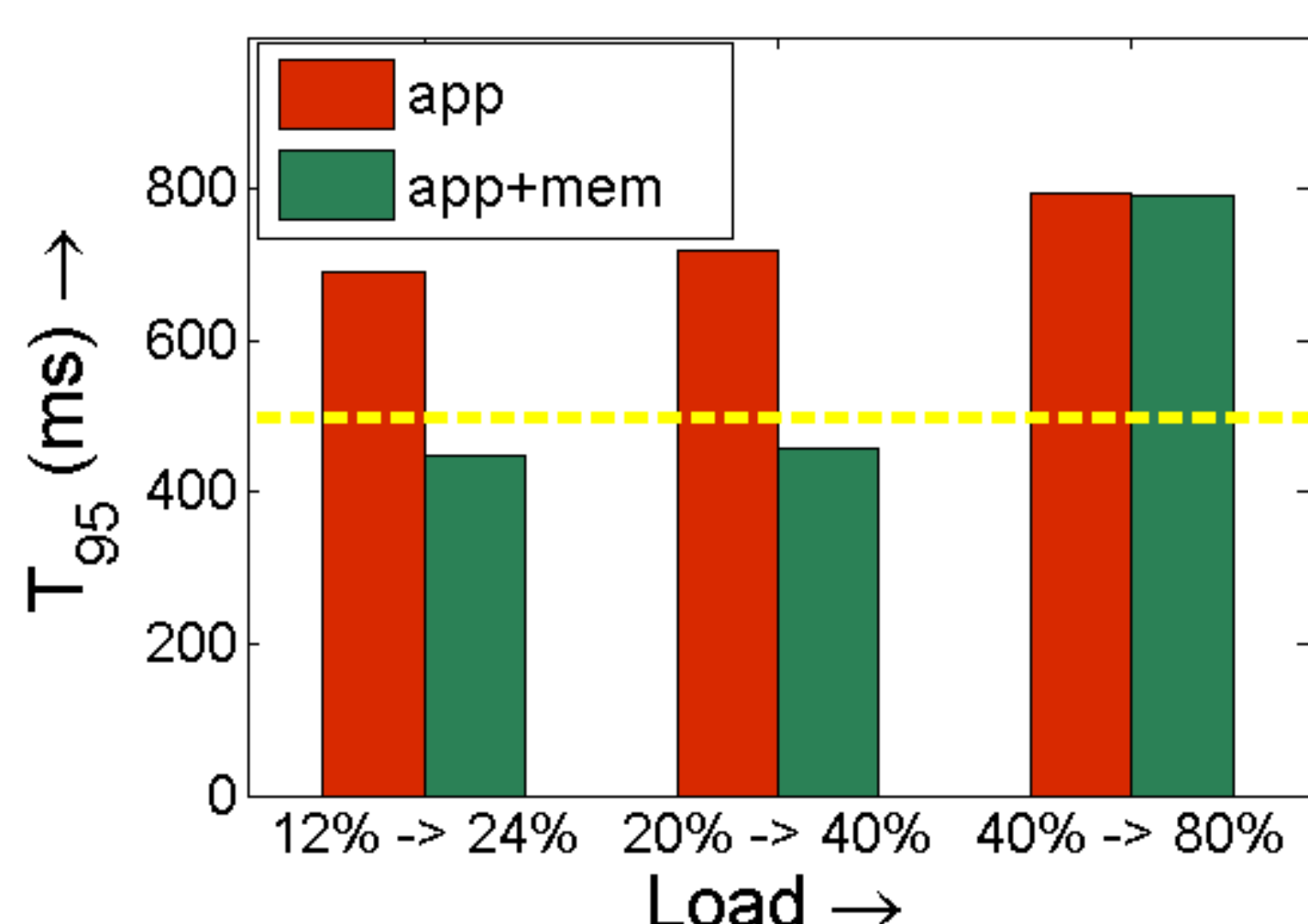**Exploit idleness**

SOLUTION

- A fraction of app load can be sent to mem servers during periods of transient overload
  - Especially while new app servers are setting up
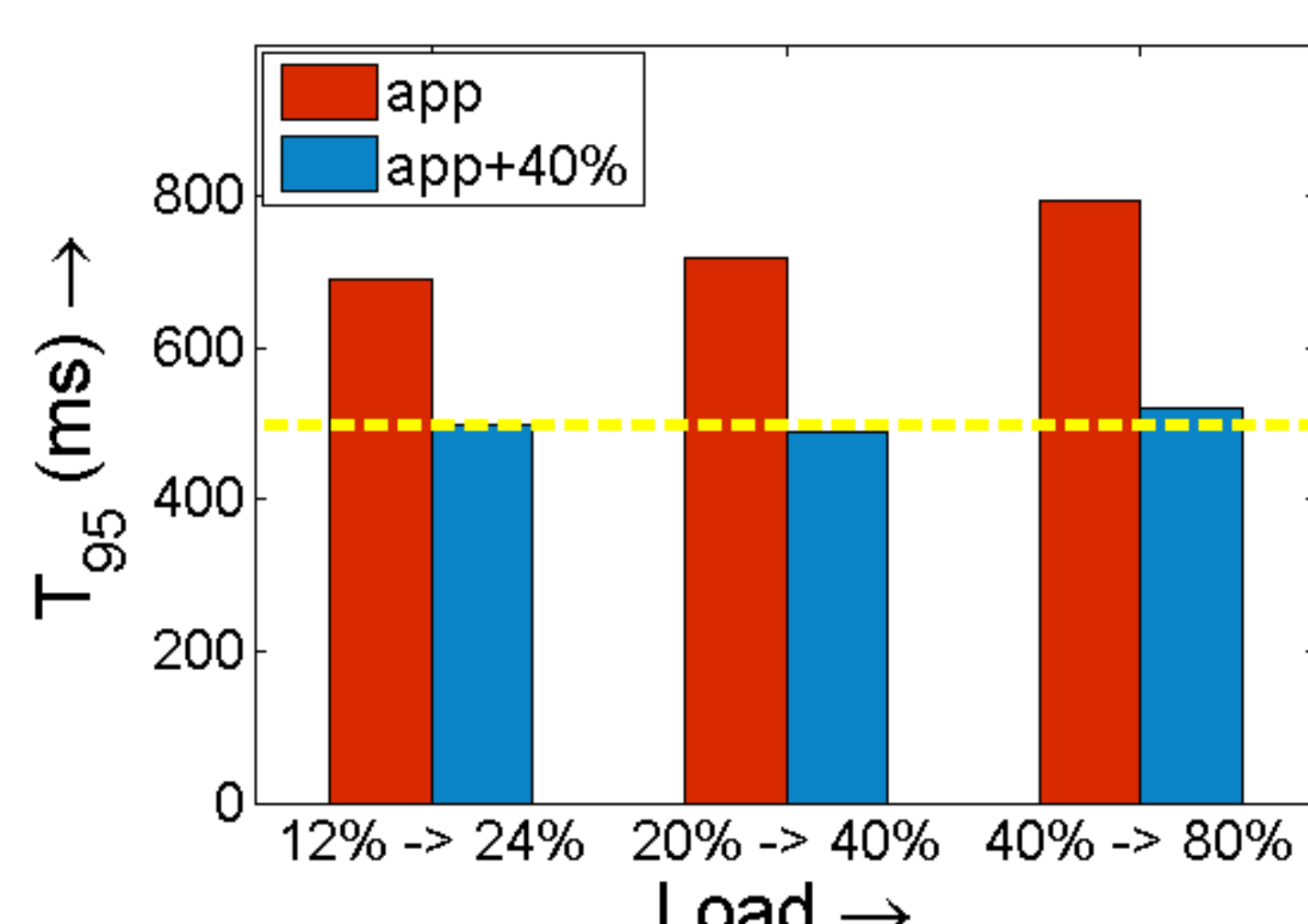- How much app load can mem servers handle?
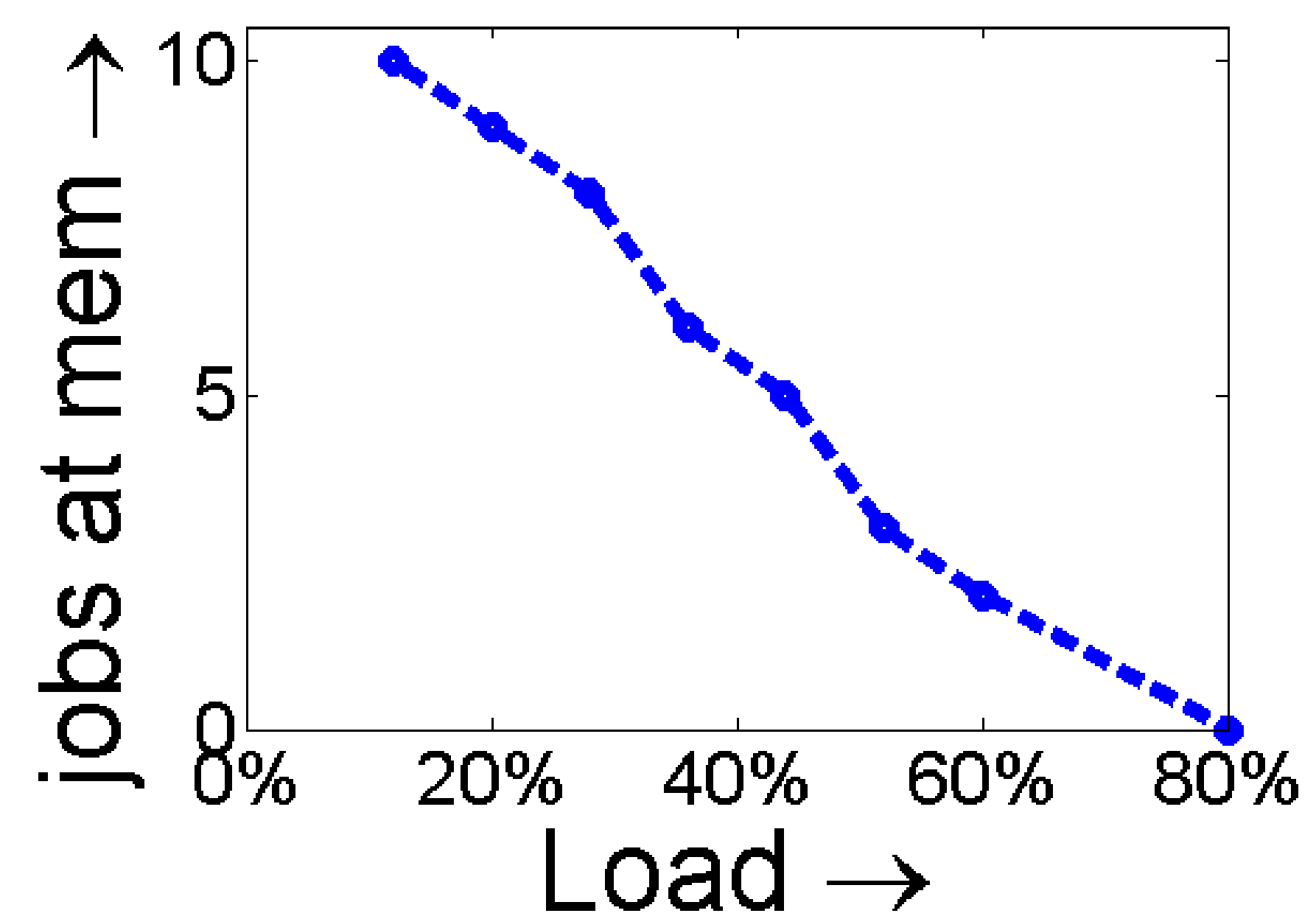
PRELIMINARY RESULTS

Big performance improvement without additional power

exploit mem servers

over provision

CONCLUSION AND FUTURE WORK

Conclusion:
- Exploit idleness in stateful servers
  - Mem servers = 40% app capacity at no cost
- Works great for low loads
- Does not work for high loads

Future work:
- Better isolation for jobs at mem servers
- Sleep states could reduce pressure on mem servers
- Need spare capacity at high loads
  - Use AutoScale to manage spare capacity