IIJ Internet Initiative Japan

An Autonomous Resource Management Model towards Cloud Morphing

Kenjiro Cho, Jean-François Baffier (IIJ)

EdgeSys2023

An Autonomous Resource Management Model towards Cloud Morphing

Kenjiro Cho, Jean-François Baffier (IIJ)

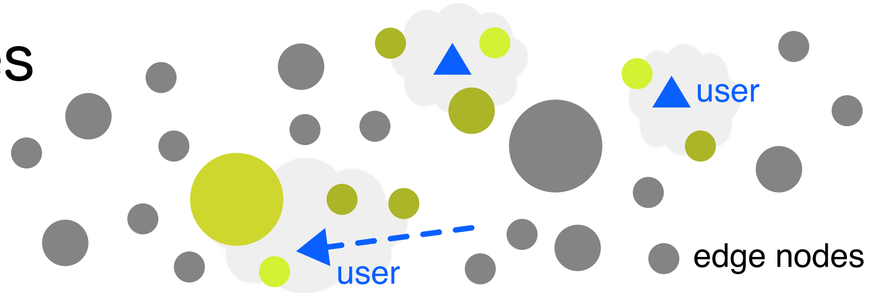
EdgeSys2023

Future Edge Clouds

- will utilize diverse and geographically scattered resources
  - edge resources would be abundant, if easily exploited
- (ephemeral stateless) micro services can be an enabler
  - enabling efficient use of underlying resources
  - making resource management much simpler
- impact similar to time-sharing or packet switching?

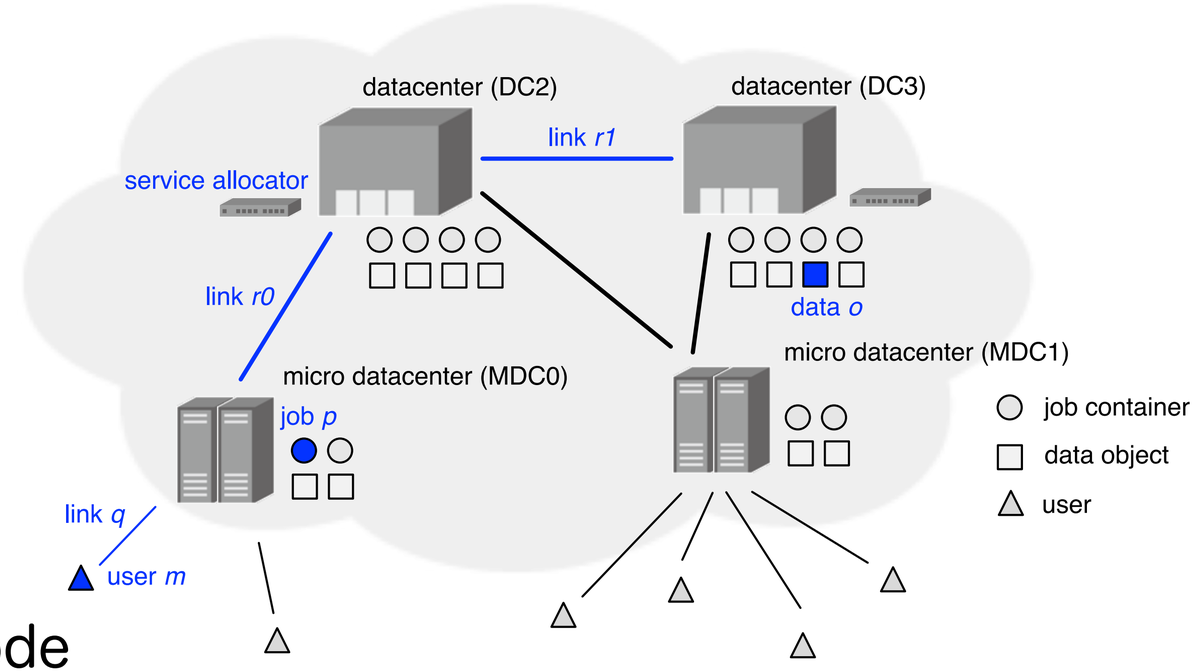
Cloud Morphing Vision

- dynamically morphing clouds, following usage patterns
  - micro services over distributed heterogeneous resources
    - an interactive job follows the user
    - a data-intensive job stays close to data
    - services are inherently fault-tolerant, disaster-resilient
- simpler operation: loose resource management
- we need a new autonomous resource management model!
  - energy saving crucial to edge resources



Simple Resource Management Model

- example: a simple system with 2 DCs and 2 MDCs
- a user requests a service to a nearby service server
- service server instantiates micro jobs
  - obtains the resource info for the job
  - identifies the user and data
  - asks nearby resource agents
    - for resource info and pseudo costs
  - assigns job to the cost-minimizing node

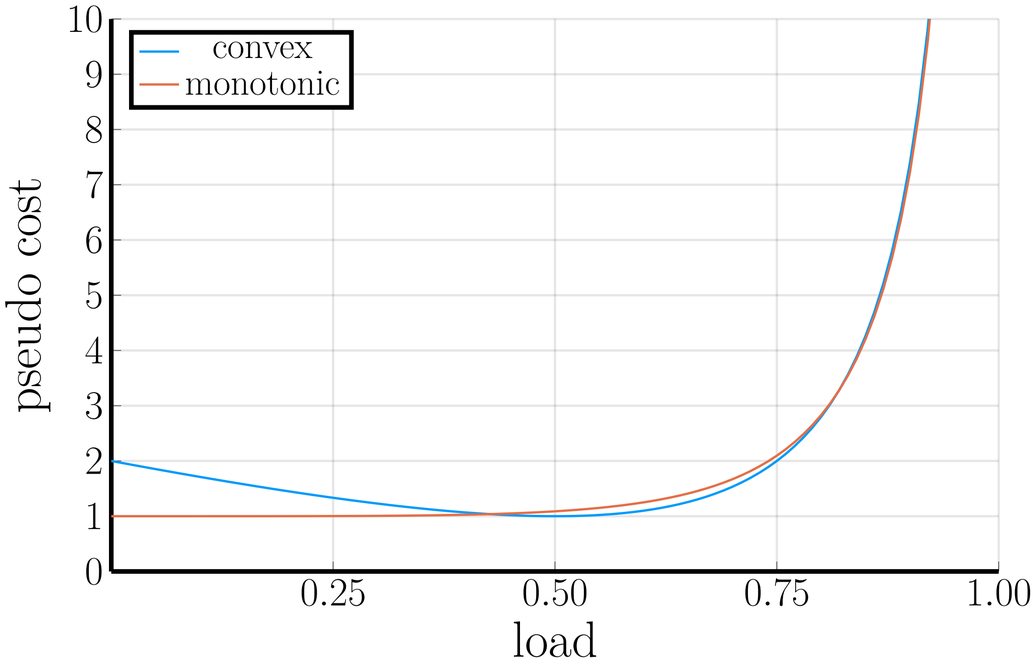


Micro-job assignment

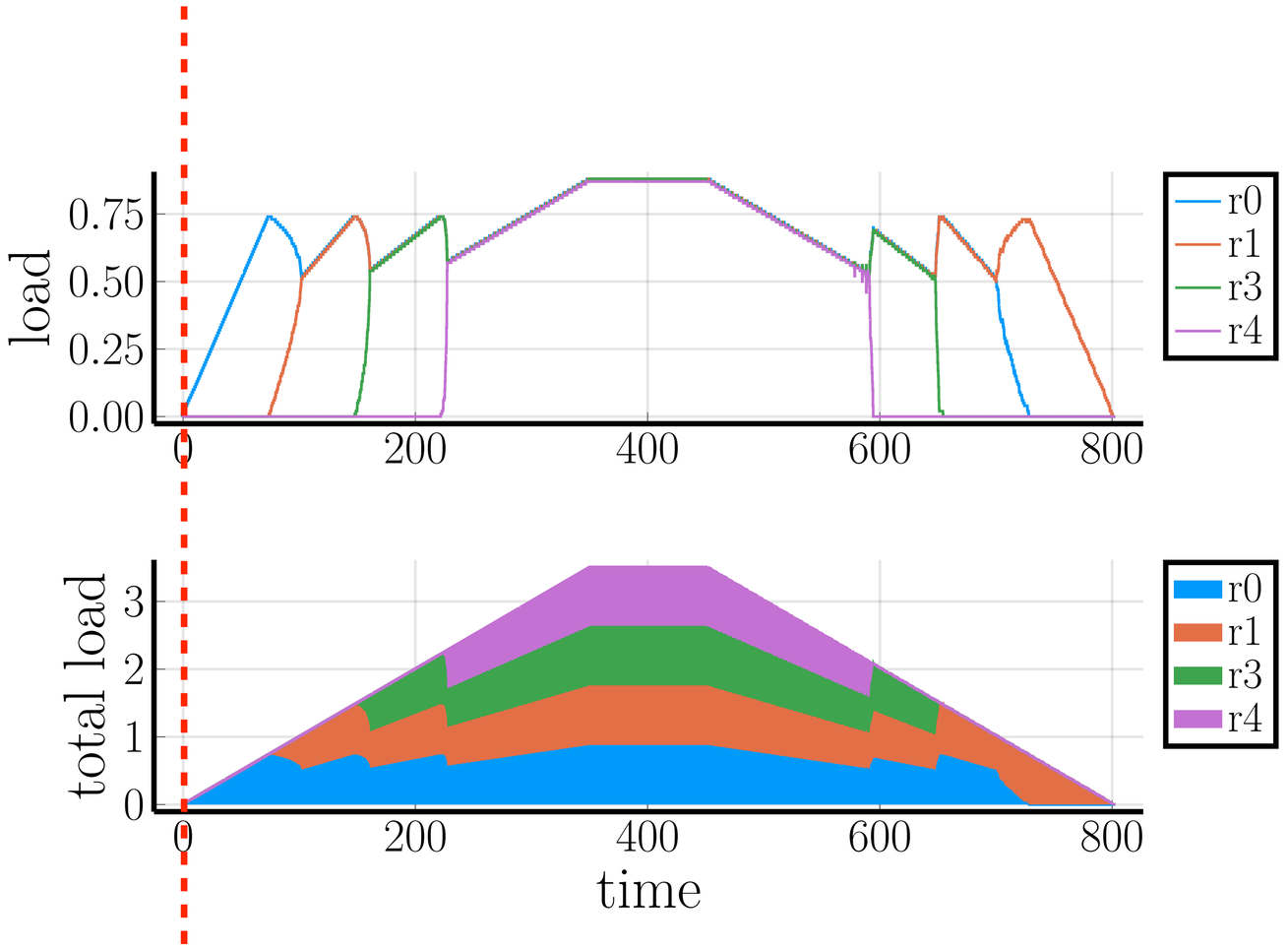
- a micro-job as $J(p,q,r,s)$
  - p: number of micro containers, q: frontend communication with user
  - r: backend communication with data, s: number of time slots
- peudo cost E to host micro-job j at node i for user m and data o
  - computing cost H and communication cost G with $f(\rho)$ cost function of load
  - $E(j,i) = H(j,i) + G(j,i,m,o)$
  - $H(j,i) = p \cdot f(\rho_i)$
  - $G(j,i,m,o) = q \cdot \Sigma f(\rho_l) + r \cdot \Sigma f(\rho_l)$
- to assign job j, the server finds the cost-minimizing node: $argmin_i E(j,i)$

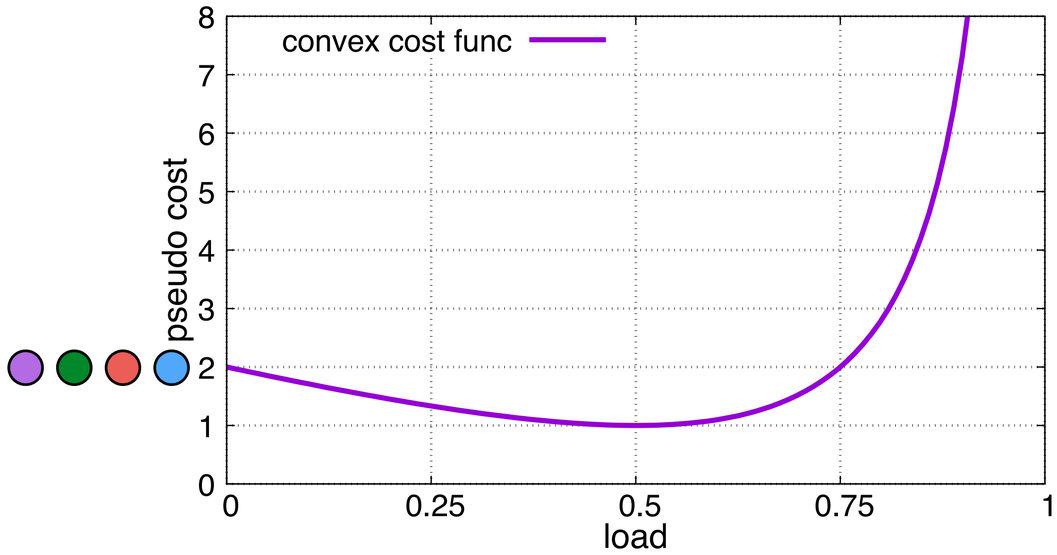
Pseudo Cost Functions

- pseudo cost: a function of the resource load, used for resource assignment
  - a barrier function for enforcing the capacity constraint
- key idea: convex pseudo-cost function
  - convex func: tries to keep the load in the target working load range
  - idle-resource pooling for keeping resources in idle-state when possible
- convex cost function: barrier func + idle-resource pooling
  - $f(\rho) = (2\rho - 1)^2/(1-\rho) + 1$
  - properties: $min\ f(\rho) = f(.5) = 1$, $f(0) = f(.75) = 2$.
- monotonic cost function: barrier func
  - $f(\rho) = \rho^{4.5}/(1-\rho) + 1$
  - to match the convex func in [.5, .75]
  - used for network link costs



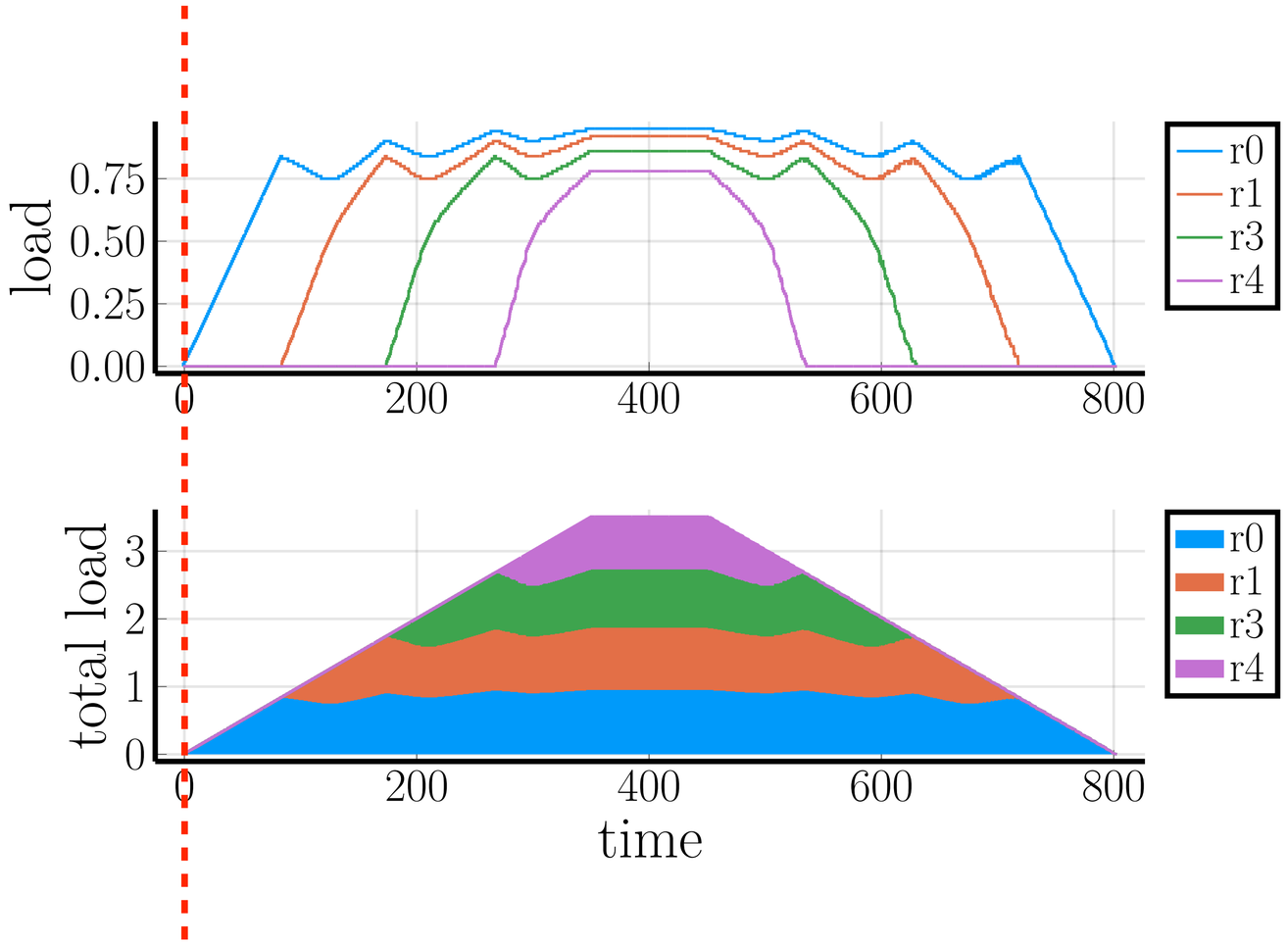
Idle-Resource Pooling with 4 Equivalent Nodes

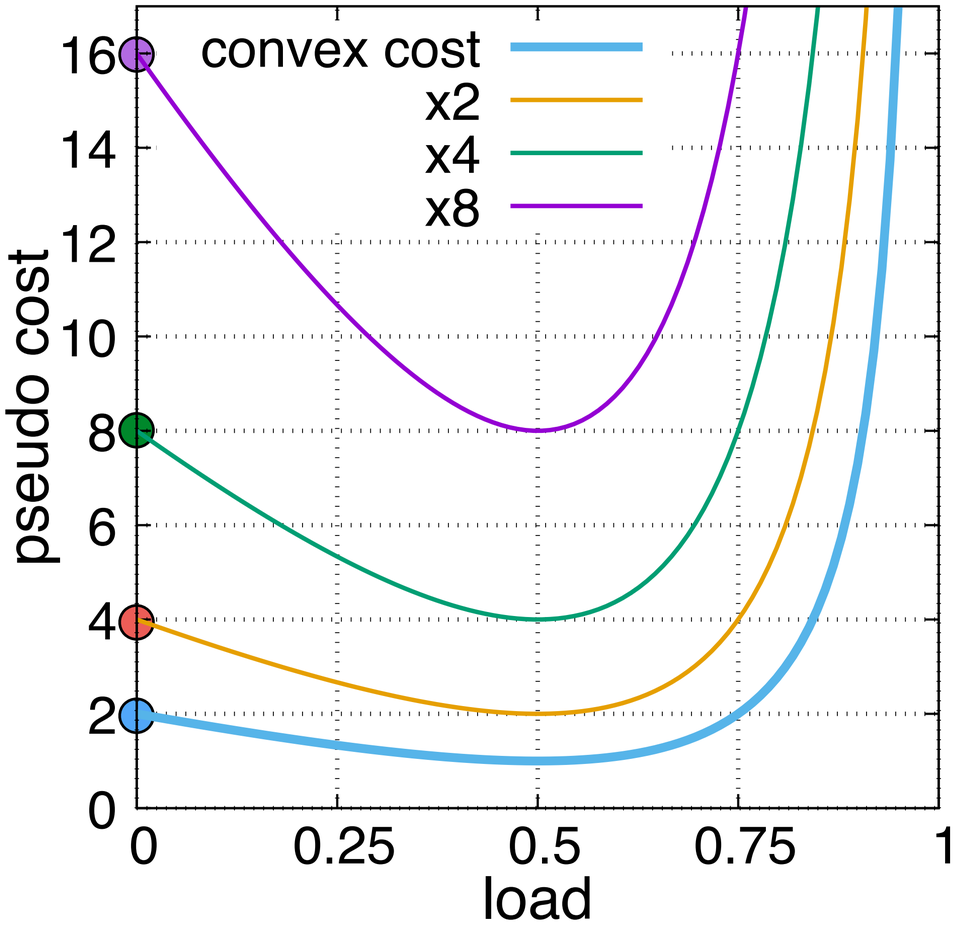




the loads of active nodes are kept in [0.5, 0.75]

Idle-Resource Pooling with 4 Proportional Nodes

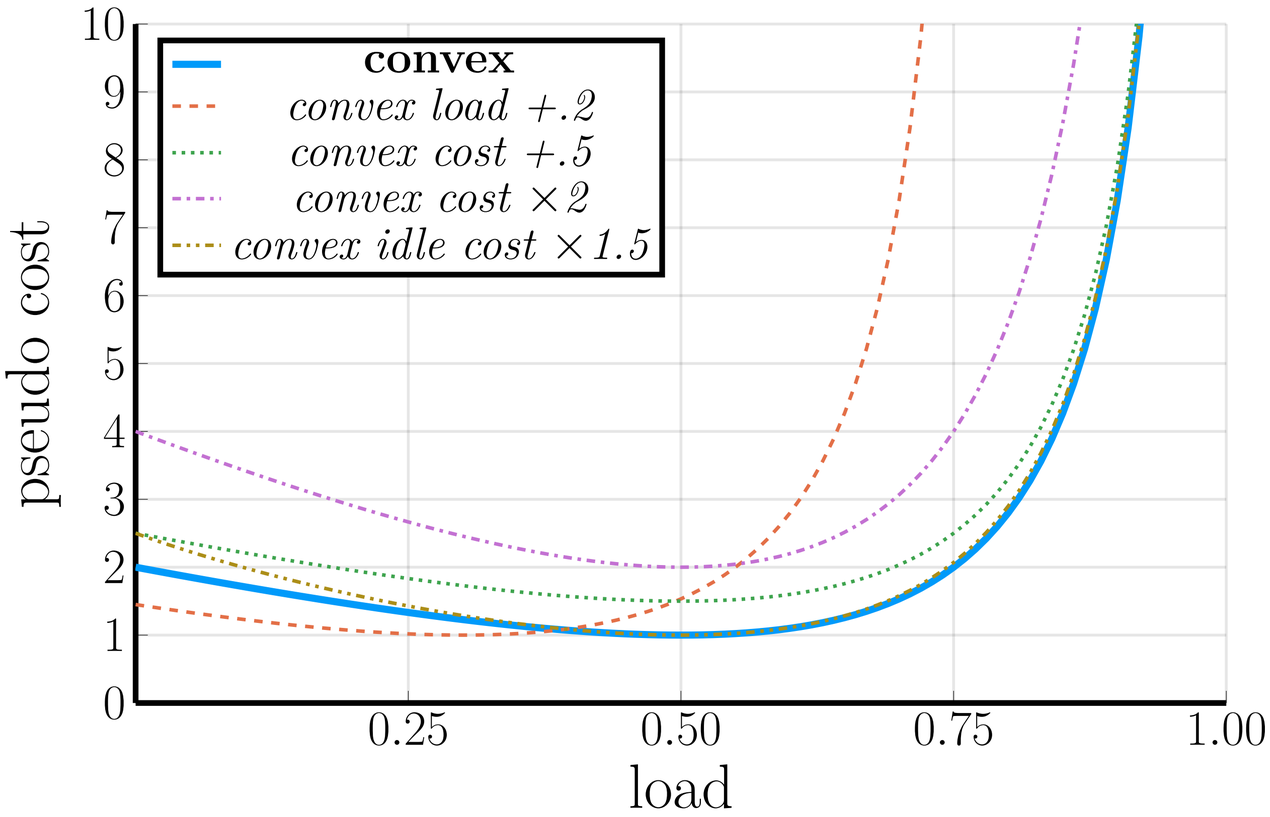




the loads go up beyond 0.75
to activate more expensive nodes

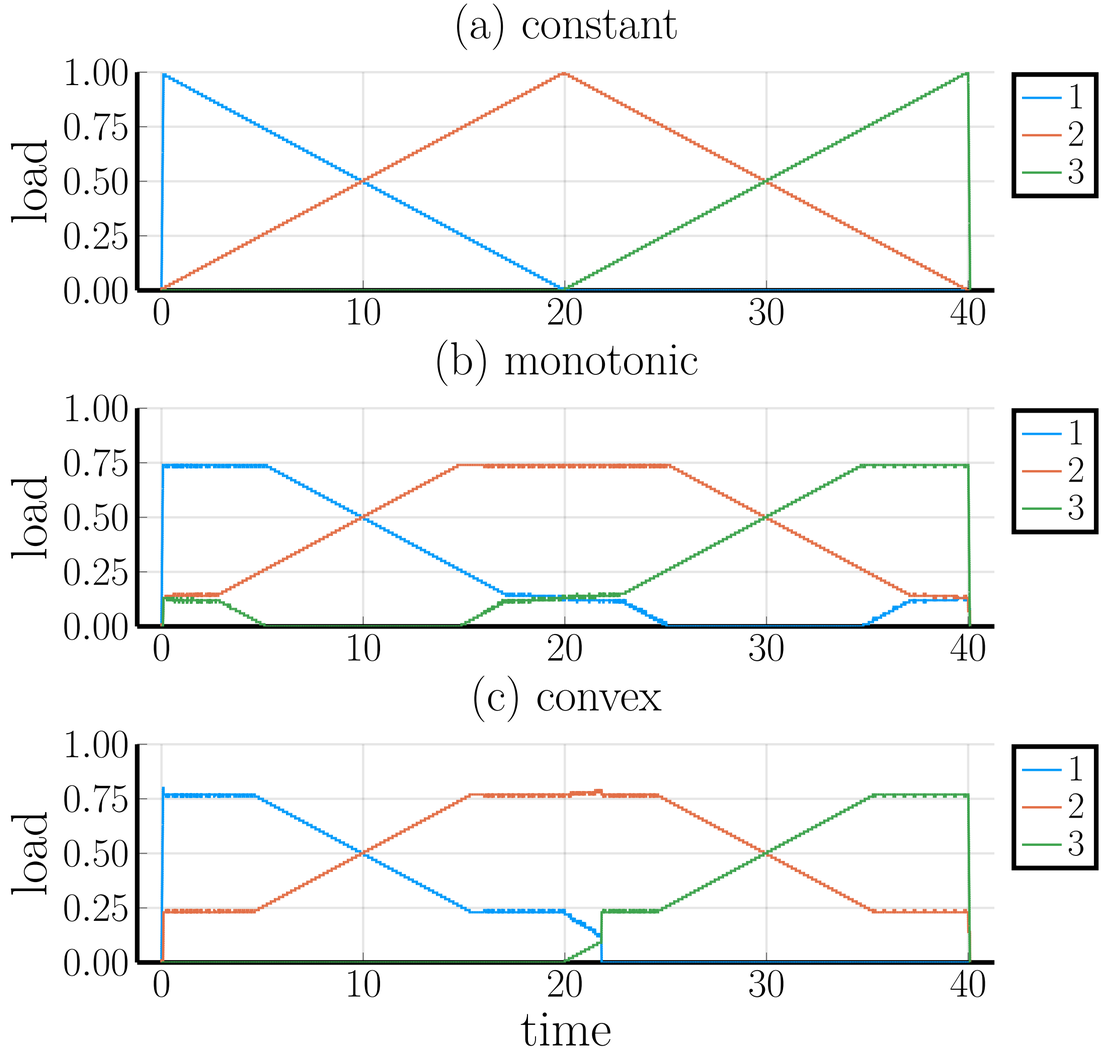
Manipulating Cost Functions

- resource usage can be controlled by manipulating cost functions
- lower or raise the load level:
  - $f'(\rho)=f(\rho+\Delta)$
- change the activation order:
  - $f'(\rho) = nf(\rho)$
  - $f'(\rho) = f(\rho) + \Delta$
- make idle state stickier:
  - $f'(\rho)=n(2\rho-1)^2/(1-\rho^n)+1$
- premium and economy services
  - same as lowering the load level
  - but for the user (not for the resource)

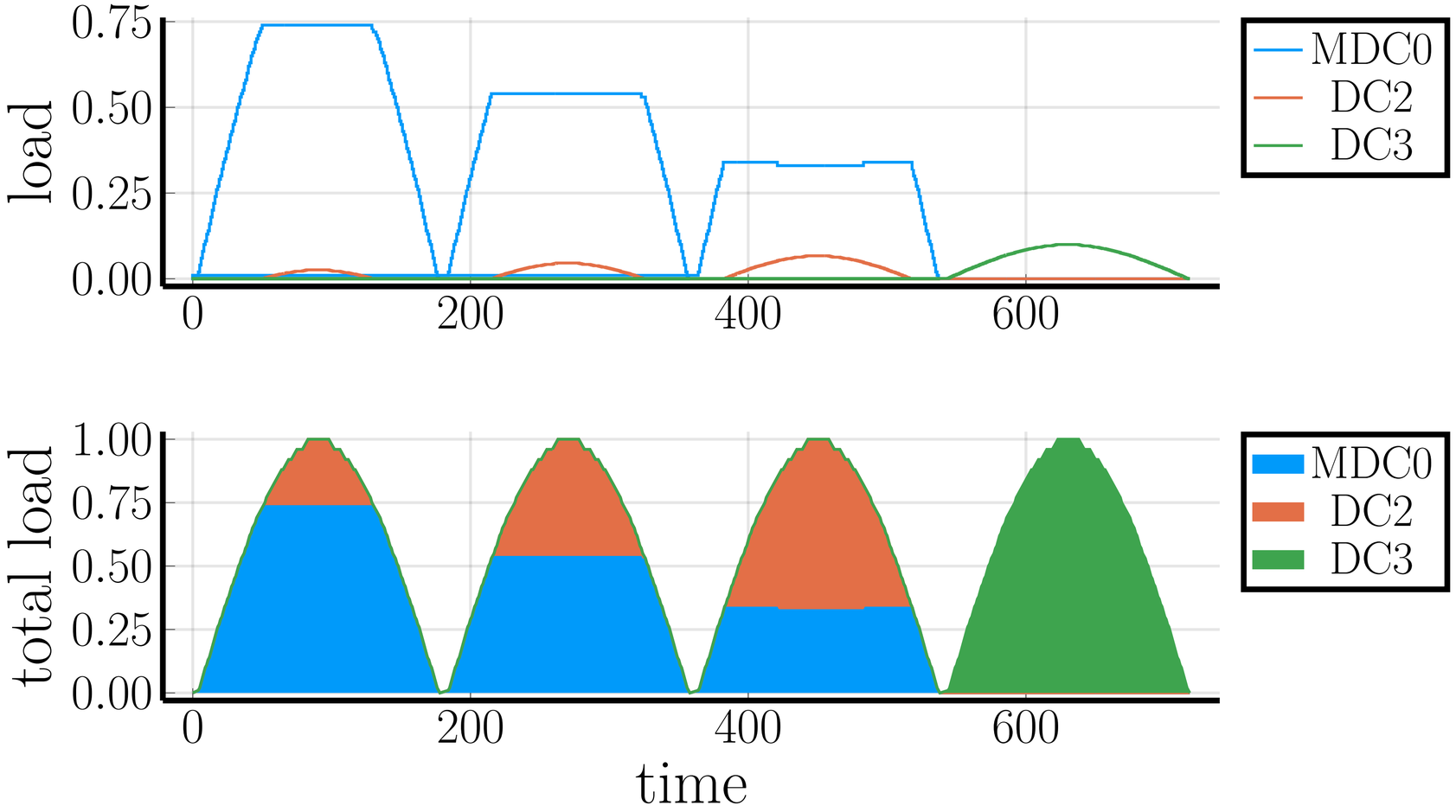


Effects of Different Cost Functions

- scenario: flock of drones over 3 nodes
  - (a) constant (baseline)
  - (b) monotonic cost function
  - (c) convex cost function
- (b)(c) keep the load under 0.75
- remaining load:
  - (b) equally shared
  - (c) shifted to one

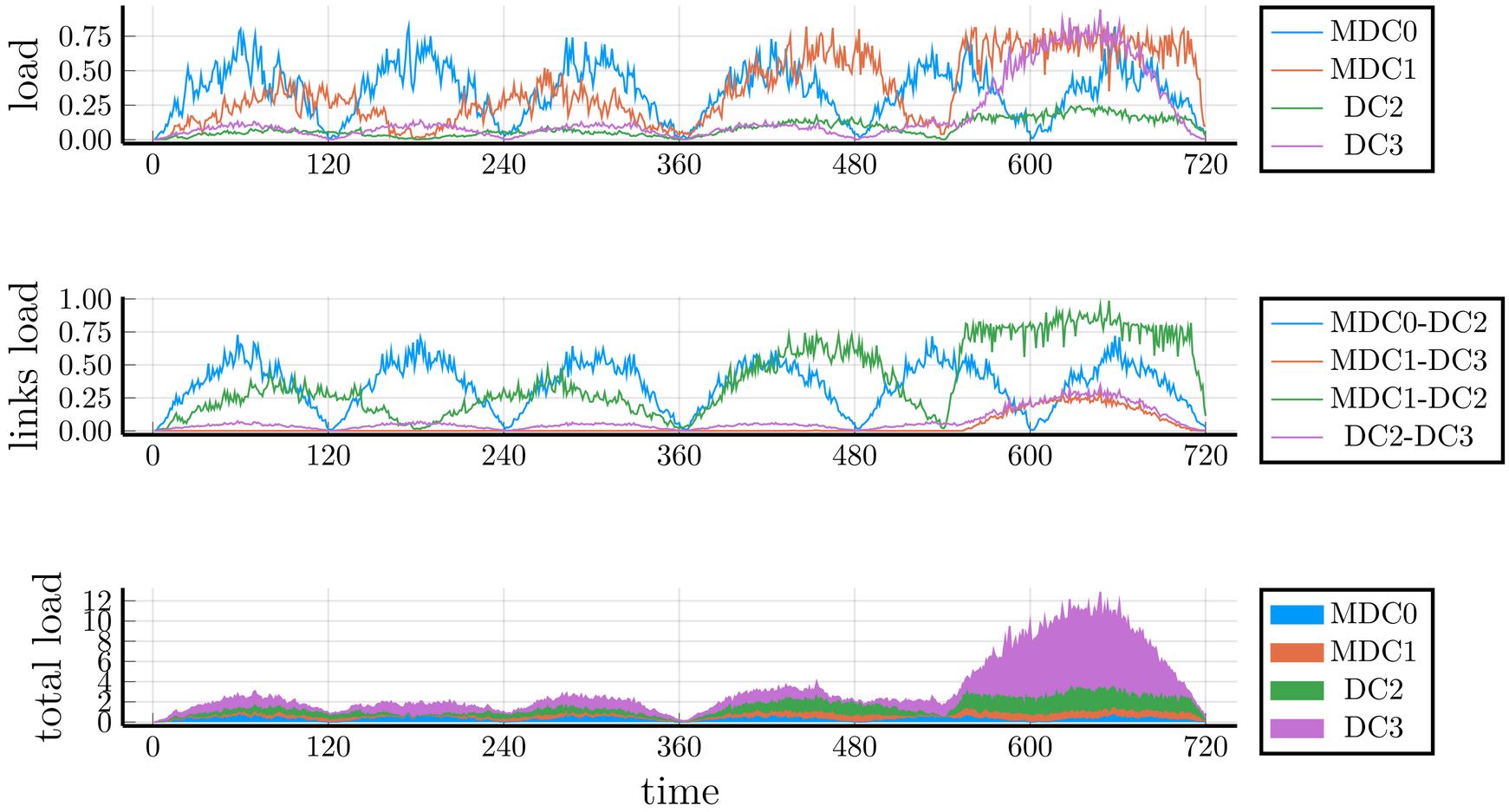


Cost Manipulations



shifting the load +.2, +.4 and raising the weight for data access

Mixed Load with 2 DCs and 2 MDCs



mixed traffic from MDCs, user1's jobs x1, x1, x2, x10

Related Work

- rich collection of work
  - congestion pricing: for access networks, transport layer, cloud computing
  - resource allocation as optimization: for edge computing, micro datacenter
- [Xu et al. 2017] separating the infra service for MDCs
- [Murphy et al. 1994] cost function as barrier func for ATM networks
- [Wagner et al. 2012] game-theoretic resource allocation in resilient military cloud

- inspired by micro services, we apply congestion pricing to edge resource management
- we are unique in using convex cost function for idle-resource pooling

Future Work

- model refinement
- prototype development
- further research
  - hierarchical configurations: inter-DC vs. intra-DC
    - cloud federation with idle resource pooling
  - crowdsourcing the resource supply: charging, authentication, etc
  - data placement: data migration using history, a new storage model?
  - dynamic L2 paths: lightpaths over WDM?

Summary

- cloud morphing vision:
  - micro services over distributed heterogeneous resources
- proposed cost function based resource management model
  - a convex cost function for idle-resource pooling
- many challenges ahead!