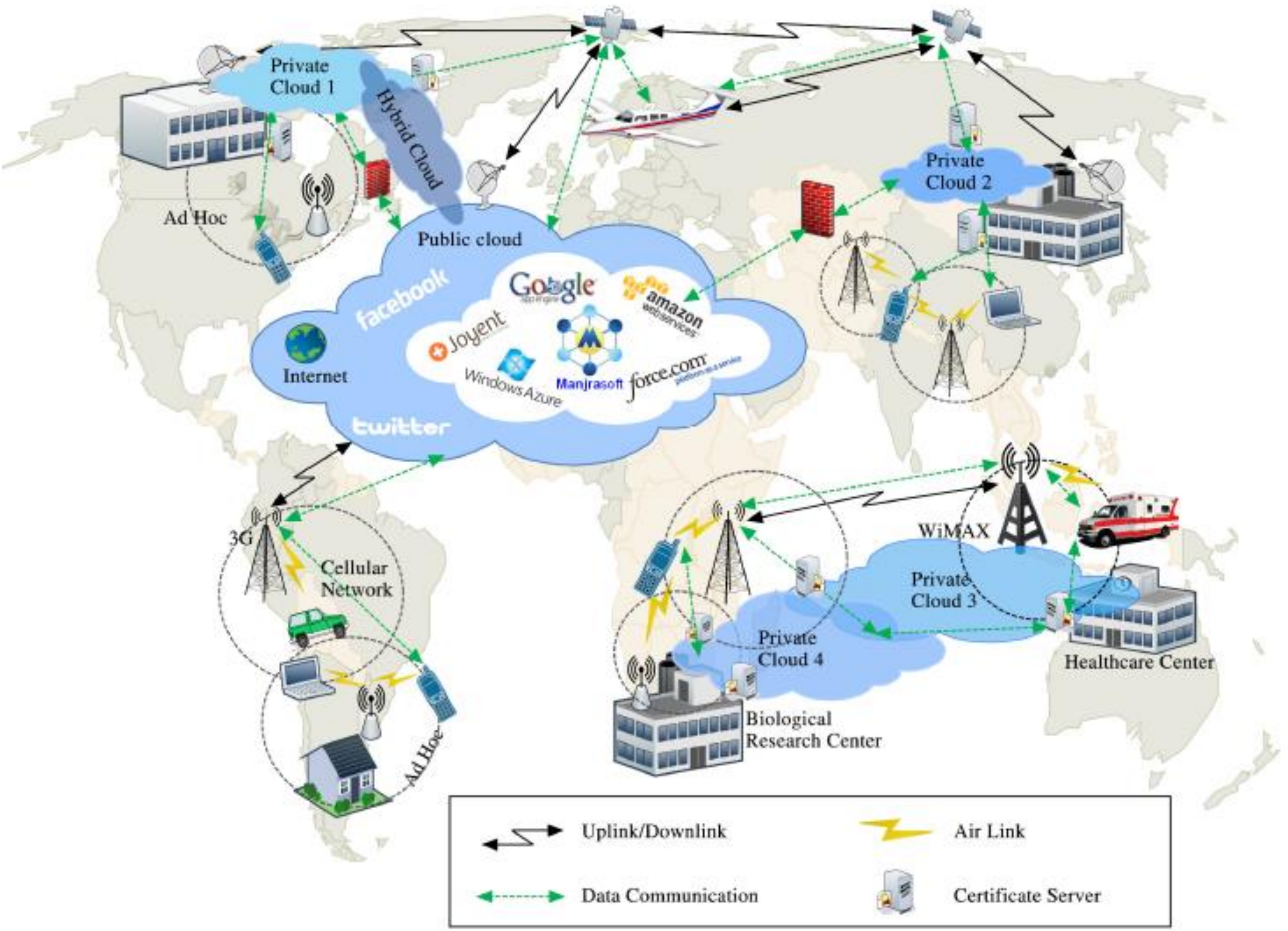
HOT Topics in Computer Science (HOT-T-CS)

Mobile Cloud Computing
Algorithms

Pradipta De
pradipta.de@sunykorea.ac.kr

Big Picture of MCC



Key Problem to Solve

- At its core, MCC framework must solve how to **partition a program for execution on heterogeneous computing resources**
- This is a classic "Task Partitioning Problem"
- Widely studied in processor resource scheduling as "job scheduling problem"
    - But in MCC the assumptions change and often makes it more challenging to solve

Task Partitioning Problem in MCC

Input:

- A call graph representing an application's method call sequence
- Attributes for each node in the graph denotes (a) energy consumed to execute the method on the mobile device, (b) energy consumed to transfer the program states to a remote server

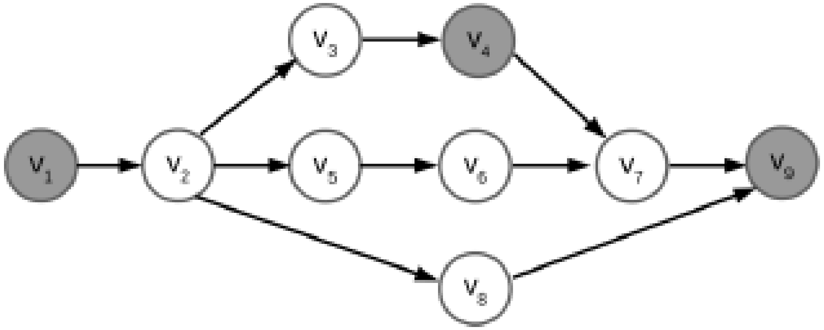
Output:

- Partition the methods into two sets – one set marks the methods to execute on the mobile device, and the second set marks the methods to execute on cloud

Goals and Constraints:

- The energy consumed must be minimized
- There is a limit on the execution time of the application
- Other constraints could be – some methods must be executed on mobile device, total monetary cost, etc.

Mathematical Formulation



Directed Acyclic Graph represents an application Call Graph

- Highlighted nodes must be executed on the mobile device ➔ called native tasks (v1, v4, v9)
- Edges represent the sequence of execution
- Any non-highlighted node can be executed either locally on the mobile device or on cloud

Integer Linear Program to solve the Task Partitioning Problem (as used in MAUI)

$$\text{maximize} \sum_{v \in V} I_v \times E_v^l - \sum_{(u,v) \in E} |I_u - I_v| \times C_{u,v}$$

$$\text{such that:} \sum_{v \in V} ((1 - I_v) \times T_v^l) + (I_v \times T_v^r))$$

$$+ \sum_{(u,v) \in E} (|I_u - I_v| \times B_{u,v}) \leq L$$

$$\text{and} \qquad I_v \leq r_v, \forall v \in V$$

- A 0-1 integer linear program, where

I_v = 0 if method executed locally,

= 1 if method executed remotely

- E is the energy cost to execute method v locally
- $C_{u,v}$ is the cost of data transfer
- T is the time to execute the method
- B is the time to transfer program state

Use of the Formulation

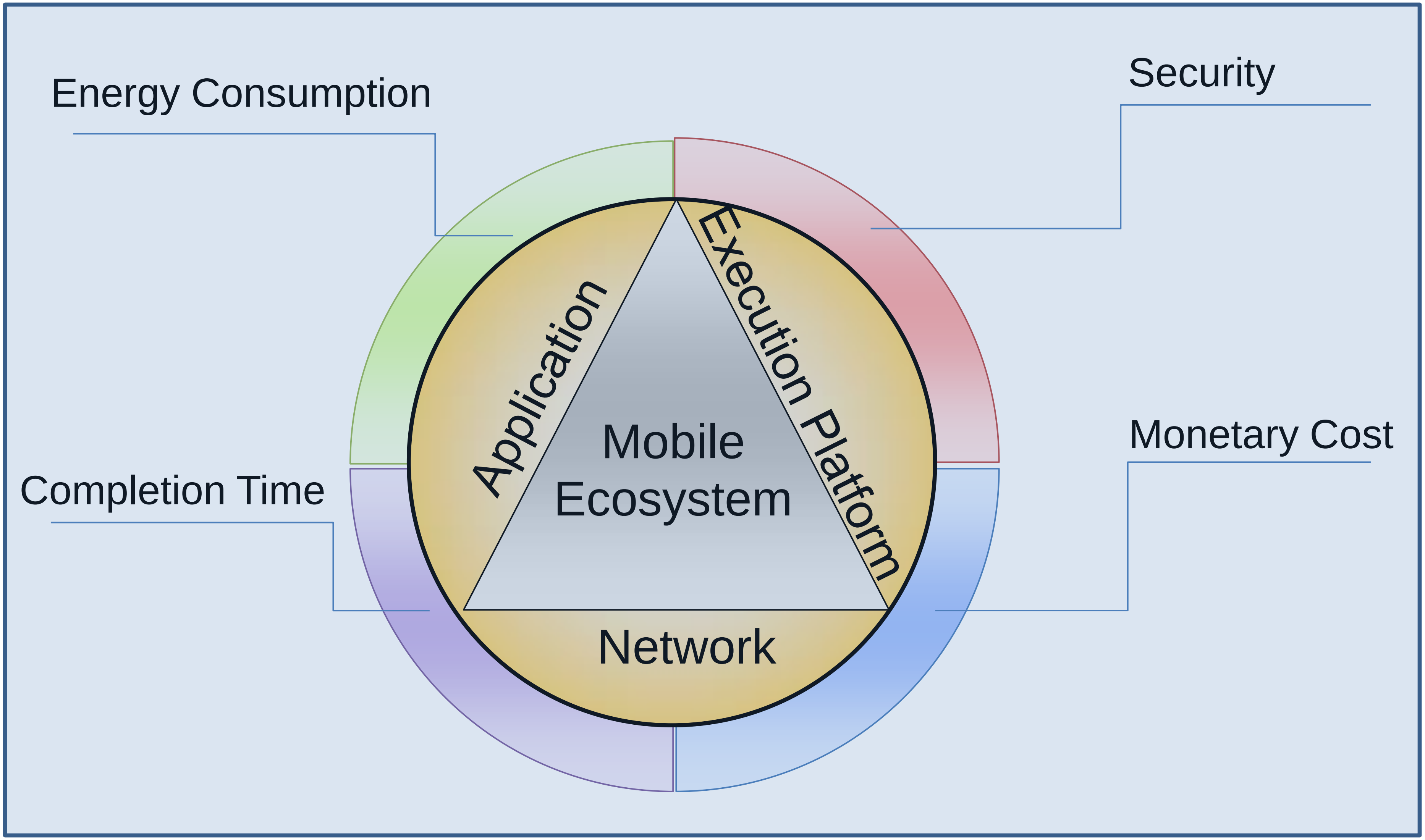
- Static Partitioning
  - When an application is launched, invoke an ILP solver which will tell where each method should be executed
  - There are also heuristics to find solutions faster

- Dynamic or Adaptive Partitioning
  - For a long running program, the environmental conditions can vary
  - Depending on the input, the energy consumption of a method can vary
  - Should we adapt the partition as the program executes?

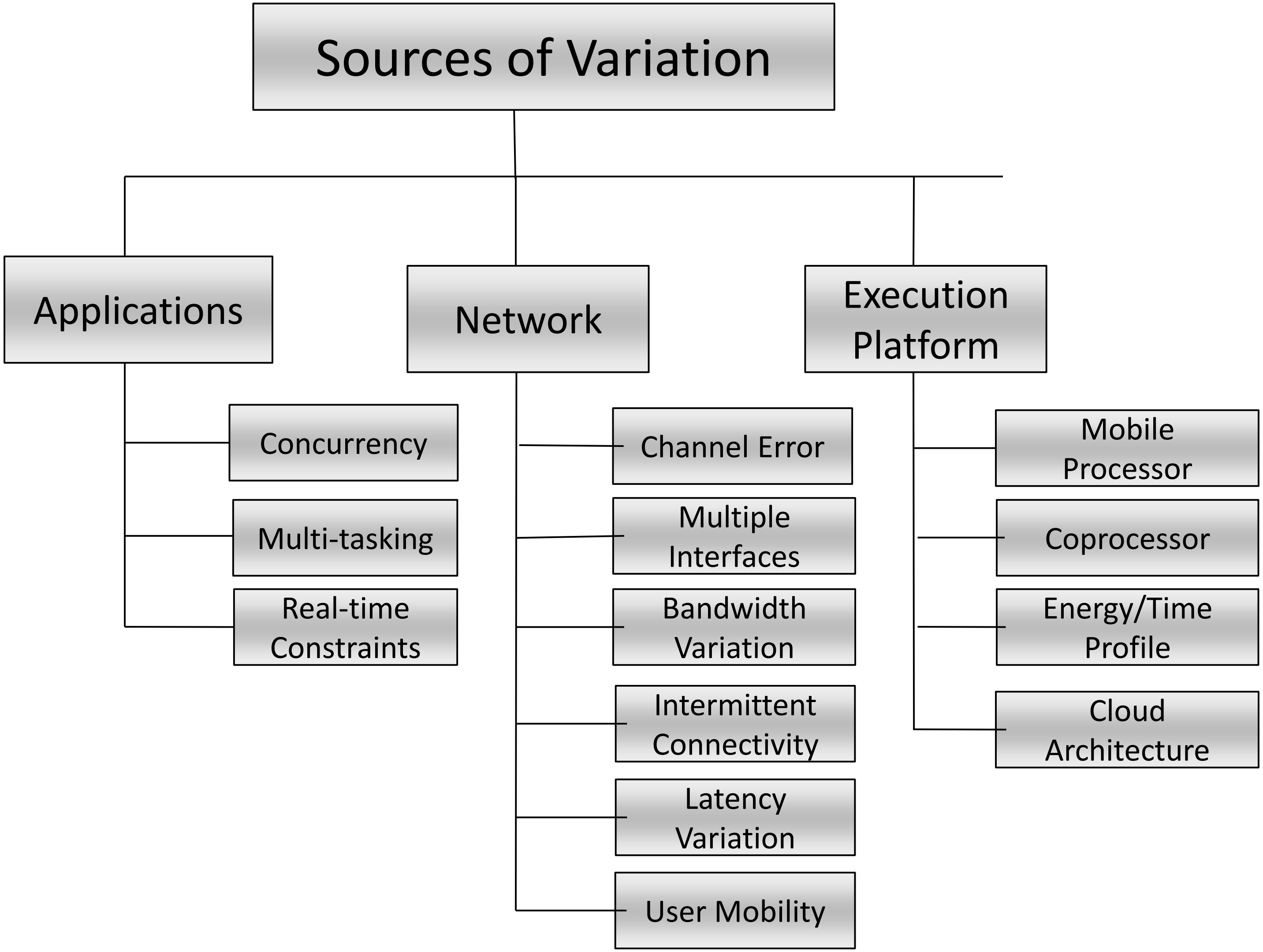
Scenarios of Variations

- User moves while using smartphone → network conditions can change
- Another app starts executing → residual energy to execute application changes
- Another user starts using cloud service → load on the cloud server increases slowing down the app
- Connectivity to network is lost → how to handle failures
- Support for variety of mobile devices → energy and time to execute profiles will change

Sources of Variation and QoE



Mobile Ecosystem



Sources of Variation in Operating Environment

- App Concurrency:
  - Smartphone apps are concurrent – task-level (independent tasks) or data-level (streaming apps)
- Simultaneous apps:
  - Presence of foreground and background apps
- Real-time constraints:
  - Imposes constraints on task completion

- Network variations:
  - Bandwidth: signal quality varies leading to changing bandwidth
  - Connectivity: cannot have guaranteed connectivity
  - Latency: The routes to the server can change, as well as, congestion on the path
  - User mobility: Can lead to change in network conditions
  - Multiple interfaces: Choice of interface can change communication properties

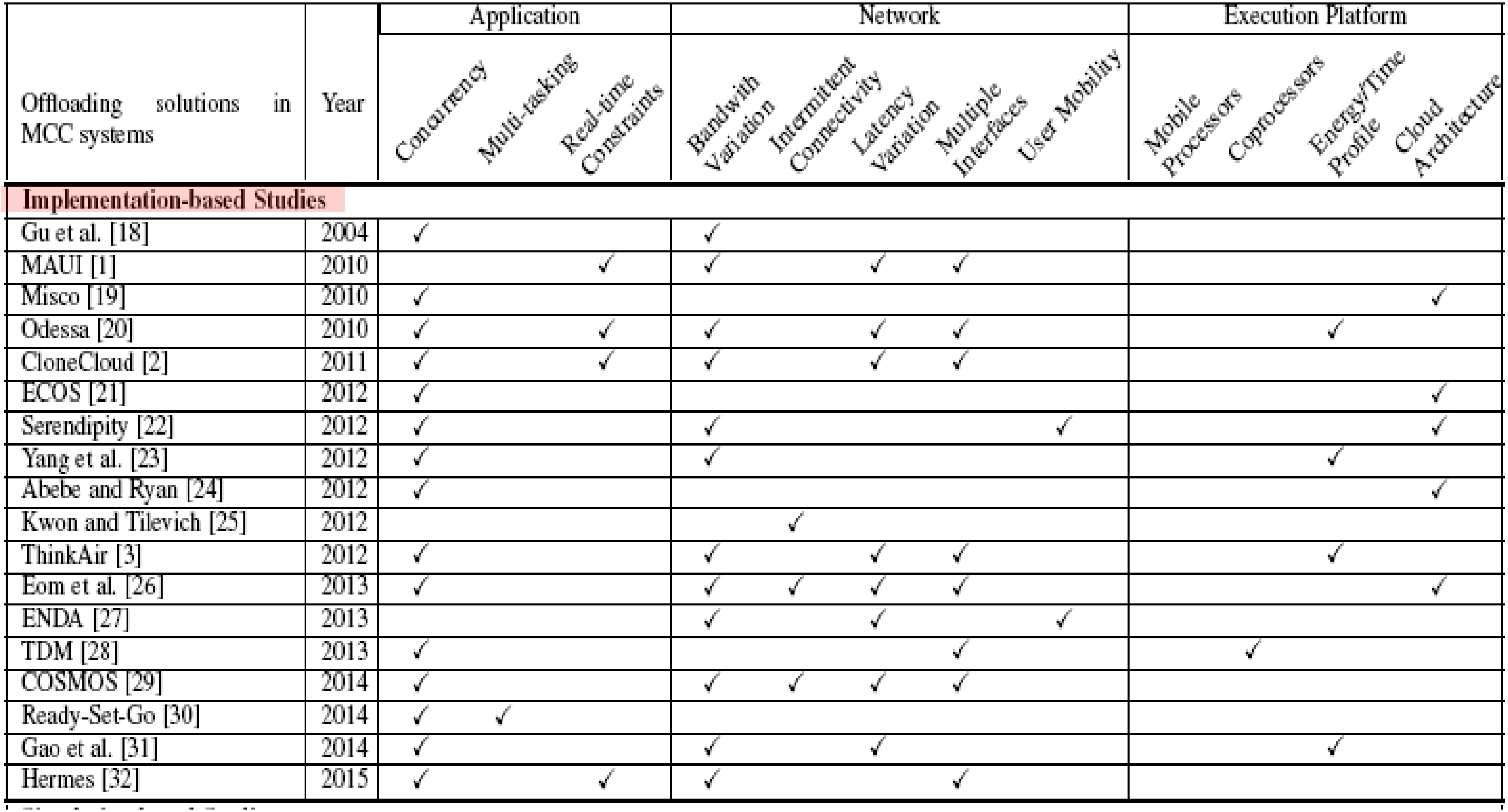
Sources of Variation in Operating Environment

- Heterogeneous Operating Platforms
  - Mobile Processors: Smartphones can have 2 to 8 processors, can implement dynamic voltage and frequency scaling
  - Coprocessors: GPUs are present in many smartphones ➔ does it help to use the GPUs
  - Energy and Time Profile: Hardware components vary in characteristics ➔ individual profiles are required
  - Cloud Architecture: Different hierarchy of cloud architecture is possible

How to evaluate MCC performance

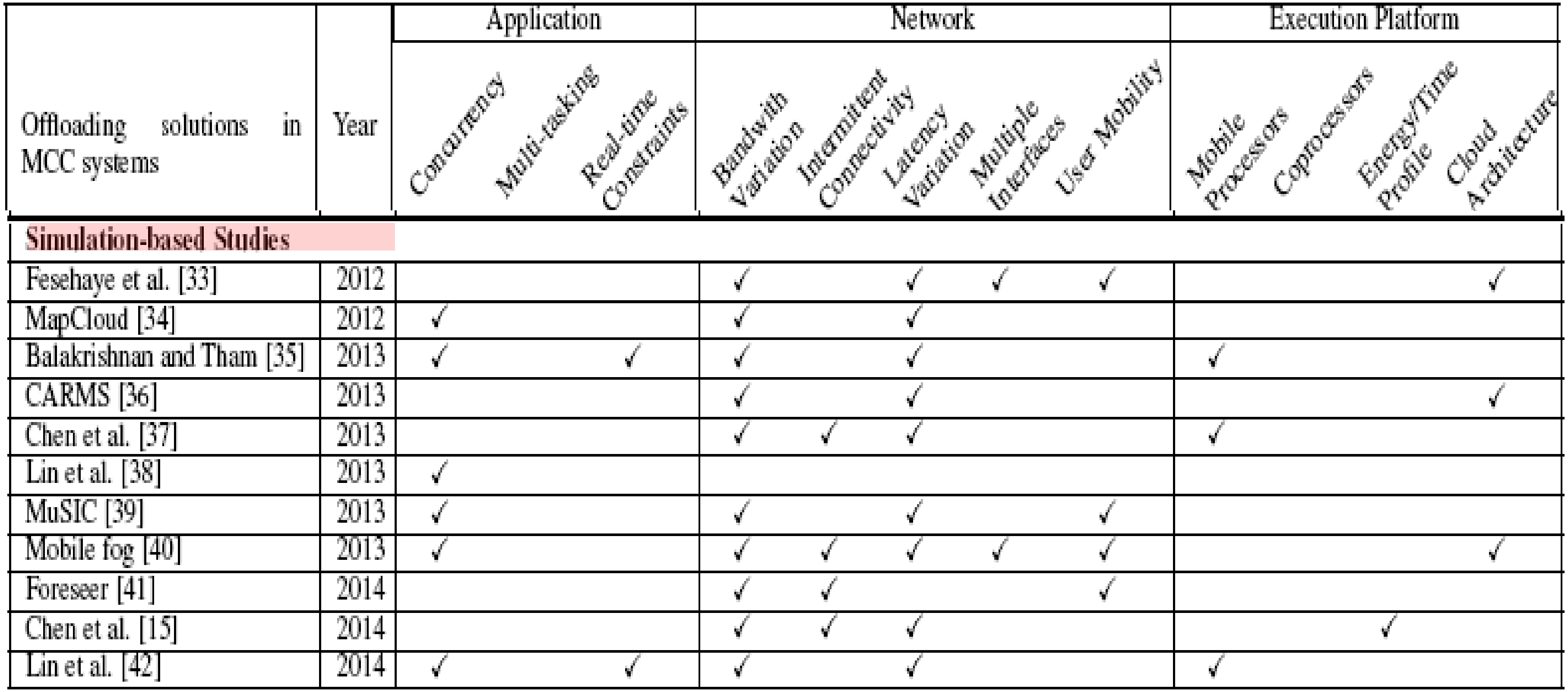
- Energy Consumption
  - Must reduce energy usage and extend battery life
- Time to Completion
  - Should not take longer to finish the application compared to local execution
- Monetary Cost
  - Cost of network usage and server usage must be optimized
- Security
  - As offloading transfers data to the servers, ensure confidentiality and privacy of data ➔ how to identify methods which process confidential data

Comparative View



Offloading solutions in MCC systems	Year	Application			Network					Execution Platform			
		Concurrency	Multi-tasking	Real-time Constraints	Bandwith Variation	Intermittent Connectivity	Latency Variation	Multiple Interfaces	User Mobility	Mobile Processors	Coprocessors	Energy/Time Profile	Cloud Architecture
Implementation-based Studies													
Gu et al. [18]	2004	✓			✓								
MAUI [1]	2010			✓	✓		✓	✓					
Misco [19]	2010	✓											✓
Odessa [20]	2010	✓		✓	✓		✓	✓				✓	
CloneCloud [2]	2011	✓		✓	✓		✓	✓					
ECOS [21]	2012	✓											✓
Serendipity [22]	2012	✓			✓				✓				✓
Yang et al. [23]	2012	✓			✓							✓	
Abebe and Ryan [24]	2012	✓											✓
Kwon and Tilevich [25]	2012					✓							
ThinkAir [3]	2012	✓			✓		✓	✓				✓	
Eom et al. [26]	2013	✓			✓	✓	✓	✓					✓
ENDA [27]	2013				✓		✓		✓				
TDM [28]	2013	✓						✓			✓		
COSMOS [29]	2014	✓			✓	✓	✓	✓					
Ready-Set-Go [30]	2014	✓	✓										
Gao et al. [31]	2014	✓			✓		✓					✓	
Hermes [32]	2015	✓		✓	✓			✓					

Comparative View

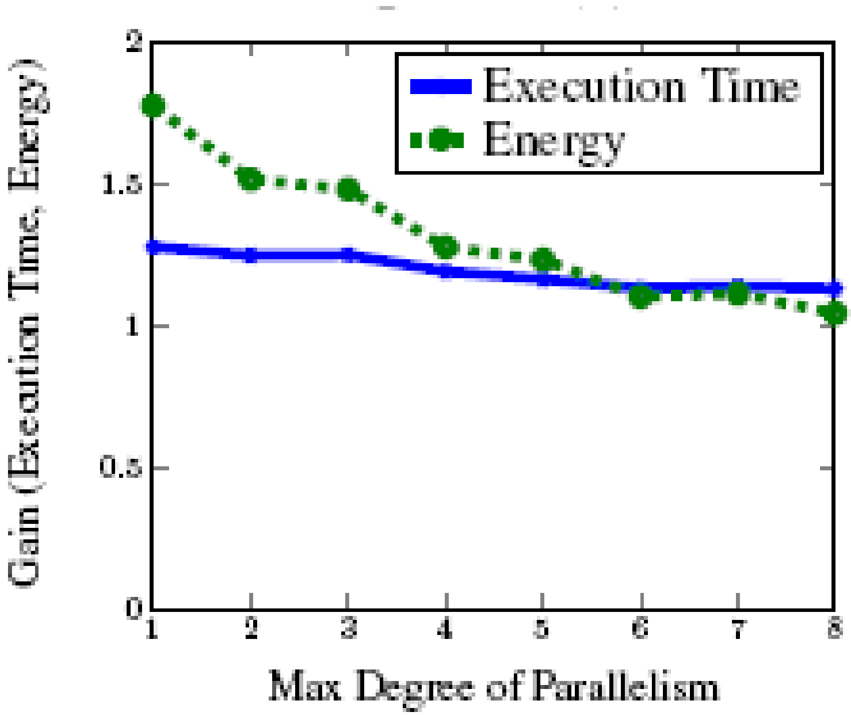


Offloading solutions in MCC systems	Year	Application			Network					Execution Platform			
		Concurrency	Multi-tasking	Real-time Constraints	Bandwith Variation	Intermittent Connectivity	Latency Variation	Multiple Interfaces	User Mobility	Mobile Processors	Coprocessors	Energy/Time Profile	Cloud Architecture
Simulation-based Studies													
Fesehaye et al. [33]	2012				✓		✓	✓	✓				✓
MapCloud [34]	2012	✓			✓		✓						
Balakrishnan and Tham [35]	2013	✓		✓	✓		✓			✓			
CARMS [36]	2013				✓		✓						✓
Chen et al. [37]	2013				✓	✓	✓			✓			
Lin et al. [38]	2013	✓											
MuSIC [39]	2013	✓			✓		✓		✓				
Mobile fog [40]	2013	✓			✓	✓	✓	✓	✓				✓
Foreseer [41]	2014				✓	✓			✓				
Chen et al. [15]	2014				✓	✓	✓					✓	
Lin et al. [42]	2014	✓		✓	✓		✓			✓			

Impact Analysis of the Parameters

- In real implementation, it is difficult to create controlled experiments to study the impact of individual parameters
- Perform a simulation based analysis, but model all the parameters into the simulation model, using different constraints
  - Precedence: maintaining the DAG ordering
  - Concurrency: degree of parallelism
  - Execution time: time budget
  - Deadline: real time restrictions
  - Energy budget: total energy to be saved

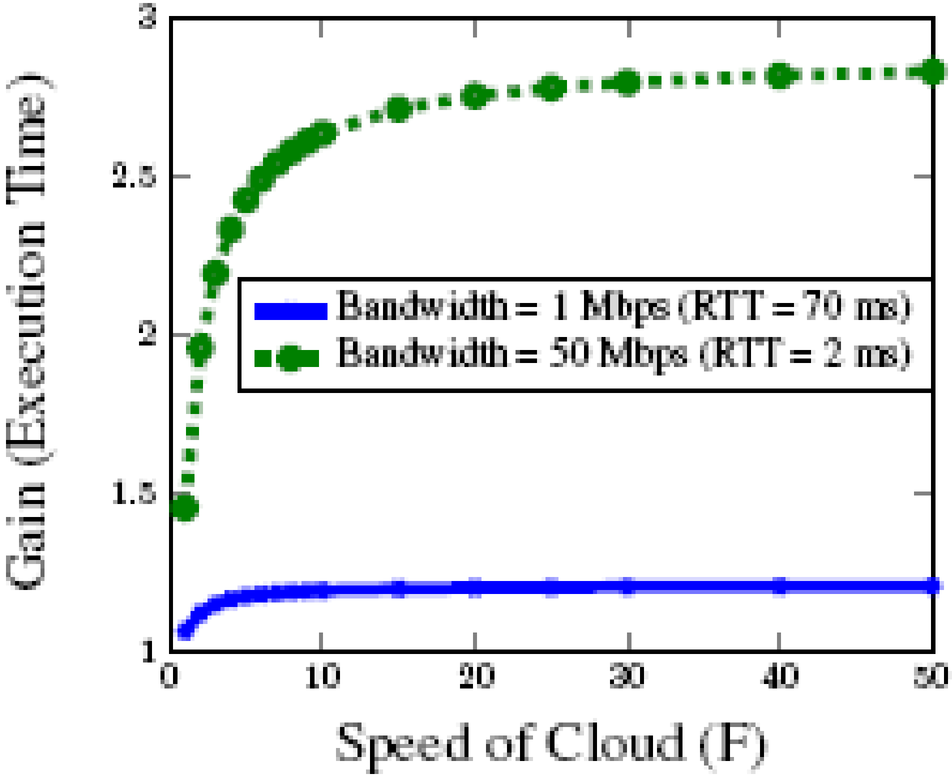
Impact of parallelism



With increasing number of threads, both execution time and energy consumed increases

But, Execution time variation is much lower as the time to migrate data is overlapped with execution

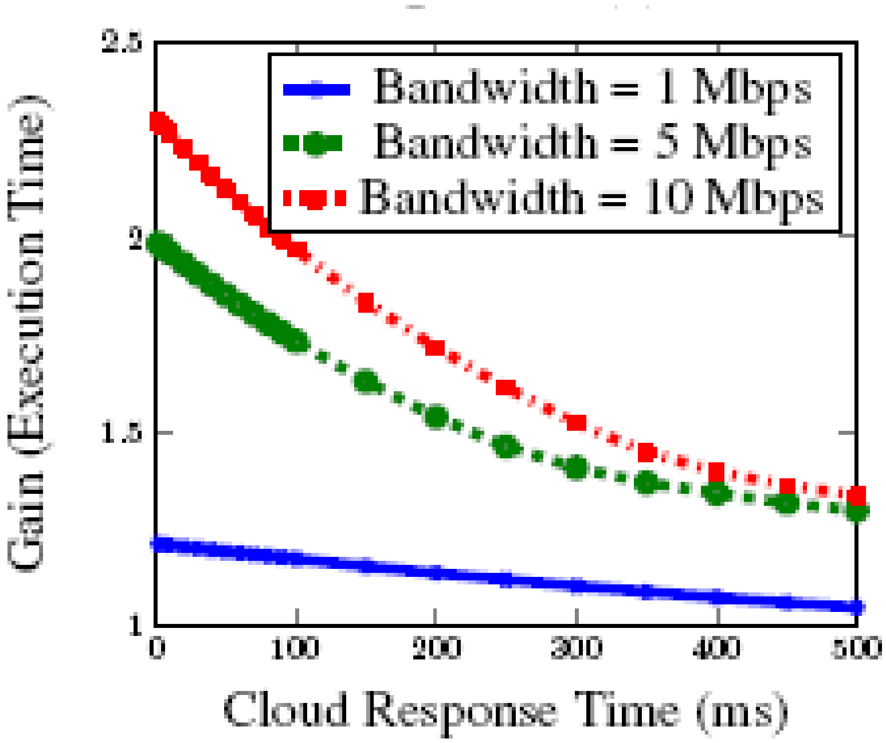
Impact of Cloud Processor Speed



- If bandwidth is low, then there is little impact of a faster cloud server
  - Low network bandwidth leads to longer use of network card, and higher energy consumption

- If the bandwidth is high, then the benefit of faster cloud server is evident

Impact of Round-Trip Delay



At lower bandwidths, RTT delay has relatively low impact on execution time

At high bandwidth, the RTT is higher than the actual transmission time, while it is opposite for low bandwidth → so at low bandwidth, there is relatively low impact of RTT

Open Questions

- How can one design a practical and usable MCC framework
  - System as well as partitioning algorithm

- Is there a scalable algorithm for partitioning
  - Optimization formulations are NP-hard
  - Heuristics fail to give any performance guarantee

- If all parameters cannot be considered, which are the most relevant parameters to consider in design of MCC systems

References

1. Sanaei, Zohreh, et al. "Heterogeneity in mobile cloud computing: taxonomy and open challenges." *Communications Surveys & Tutorials, IEEE* 16.1 (2014): 369-392.
2. Ra, Moo-Ryong, et al. "Odessa: enabling interactive perception applications on mobile devices." *Proceedings of the 9th international conference on Mobile systems, applications, and services*. ACM, 2011.
3. Shi, Cong, et al. "COSMOS: computation offloading as a service for mobile devices." *Proceedings of the 15th ACM international symposium on Mobile ad hoc networking and computing*. ACM, 2014.
4. Xiang, Liyao, et al. "Ready, set, go: Coalesced offloading from mobile devices to the cloud." *INFOCOM, 2014 Proceedings IEEE*. IEEE, 2014.
5. Kao, Yi-Hsuan, et al. "Hermes: Latency Optimal Task Assignment for Resource-constrained Mobile Computing.“, In Infocom 2015.
6. Lin, Xue, et al. "Energy and performance-aware task scheduling in a mobile cloud computing environment." *Cloud Computing (CLOUD), 2014 IEEE 7th International Conference on*. IEEE, 2014.