Paying Less for More? Combo Plans for Edge-Computing Services

Shizhe Zang, Wei Bao, Phee Lep Yeoh, Branka Vucetic, Yonghui Li

Faculty of Engineering and IT,

University of Sydney

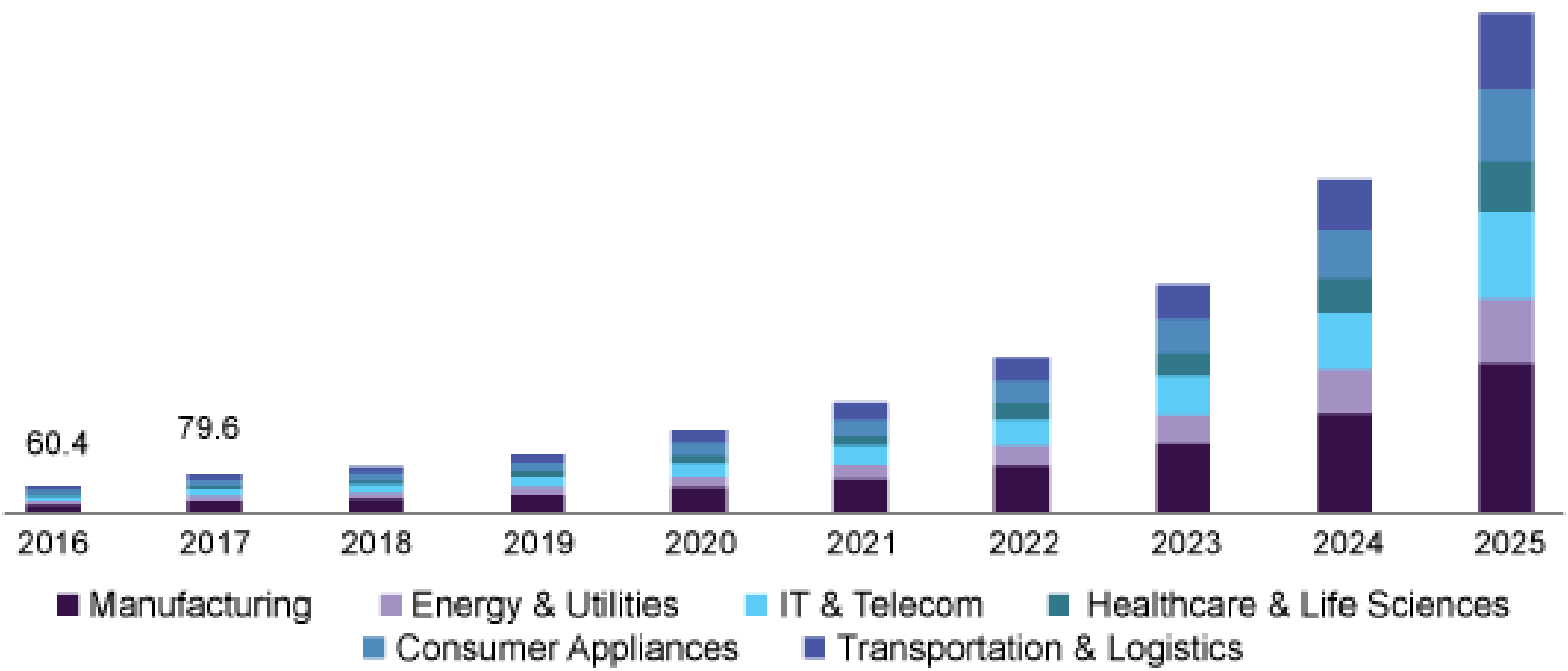
THE UNIVERSITY OF SYDNEY

Outline

- Background and Motivation
- System Model and Problem Formulation
- Problem Solution: Smart Online Reservation Algorithm
- Simulation Result
- Conclusion

Background and Motivation

Edge-computing market opportunity and potential



Background and Motivation

- Vodafone and Continental are using edge computing to make Germany's roads safer.
- AT&T is using edge computing for smart city IoT.
- Amazon, HPE and Saguna are building the edge-computing platforms.

Background and Motivation

Pricing: 1. How do edge-computing service providers design the plans?

2. How do customers reserve the plans?

Background and Motivation

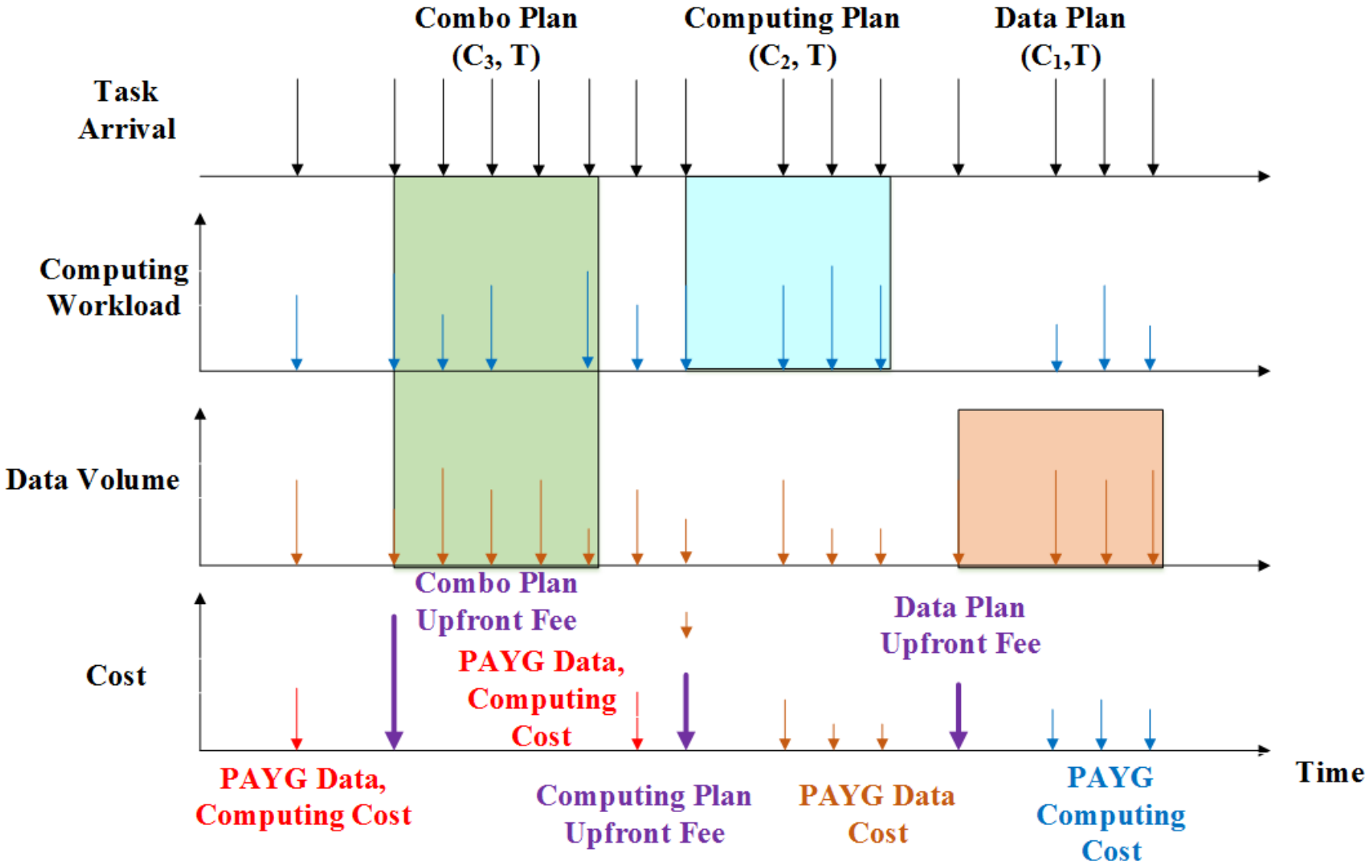
- Individual plans for computing workload and data volume.

  Computing plan (edge computing) <-> Cloud computing plan

  Data plan (edge computing) <-> mobile phone data plan

- Combo plan for edge-computing services covering both data volume and computing workload.

System Model and Problem Formulation



System Model and Problem Formulation

Comparison

- PAYG: no long term commitment, pay for what you use.

- Plans: high upfront fee cost, save money if many tasks come in the following reservation period.

	Flexibility	Upfront Fee	Coverage
PAYG Data	High	0	Small
Data Plan	Medium	C_1	Medium
Computing Plan	Medium	C_2	Medium
Combo Plan	Low	C_3	High

System Model and Problem Formulation

Online plan reservation problem

- To reserve not to reserve a plan? To reserve which plan? (cost minimization)

- Online reservation: the future information is incomplete or unreliable.

- Once the edge-computing service provider knows users' strategy, it can adjust the combo plan price to maximize its profit.

System Model and Problem Formulation

Competitive ratio: measure of online algorithms

› Compare the online strategy with the optimal offline algorithm.

› $\boldsymbol{S}$: any edge-computing service request sequence

$C_A(\boldsymbol{S})$: cost of online reservation algorithm A

$C_{OPT}(\boldsymbol{S})$: cost of optimal offline reservation algorithm

$$C_A(\boldsymbol{S}) \leq c \cdot C_{OPT}(\boldsymbol{S})$$

› An optimal online algorithm gives minimal c for all possible $\boldsymbol{S}$.

Problem Solution: SOR Algorithm

Outline of Smart Online Reservation Algorithm

› Input

Typical data cost: sum of PAYG data cost during $(t-T, T]$

Typical computing cost: sum of PAYG computing cost during $(t-T, T]$

› Output

Plan reservation decisions for the sequence of edge-computing tasks.

{PAYG, data plan, computing plan and combo plan}

Problem Solution: SOR Algorithm

Algorithm 1 SOR Case 1: Inexpensive Combo Plan

1: Proceed if there is a new task.
2: **if** there is no on-going combo plan **then**
3: | Calculate x and y as the typical data cost and computing cost.
4: | **if** $x+y \geq C_3$ **or** $y \geq C_2$ **or** $x \geq C_1$ **then**
5: | | Reserve a combo plan.
6: | **else**
7: | | Satisfy the task by PAYG.
8: | **end if**
9: **end if**
10: Repeat from step 1.

Problem Solution: SOR Algorithm

Working principle

› Main idea: minimize the competitive ratio at the worst case (most adverse edge-computing service request sequence)

› Worst case: after the algorithm reserves a plan, no corresponding request comes.

$$\min_{x,y,\boldsymbol{p}} \frac{x + y + u(\boldsymbol{p})}{c_{opt}(x, y)}$$

Online algorithm cost

Optimal offline cost

x: typical data cost, y: typical computing cost, $u(\boldsymbol{p})$: upfront fees with plan choices $\boldsymbol{p}$

Problem Solution: SOR Algorithm

Findings

- Competitive ratio of SOR: $1 + \frac{C_1+C_2}{C_3}$ (minimum among all online deterministic algorithms).

- SOR only reserves the combo plan when

$$C_3 < \min(\sqrt{C_1(C_1 + C_2)}, \sqrt{C_2(C_1 + C_2)})$$

- Critical price point: combo plan is more attractive compared with individual plans below this point.

Simulation Result

Trace Driven Performance Comparison

› Compare the overall cost performance with the following benchmark strategies.

PAYG only (P): All tasks are satisfied by PAYG.

Combo only (C): A combo plan is purchased when a new task arrives and the previous combo plan expires.

Simulation Result

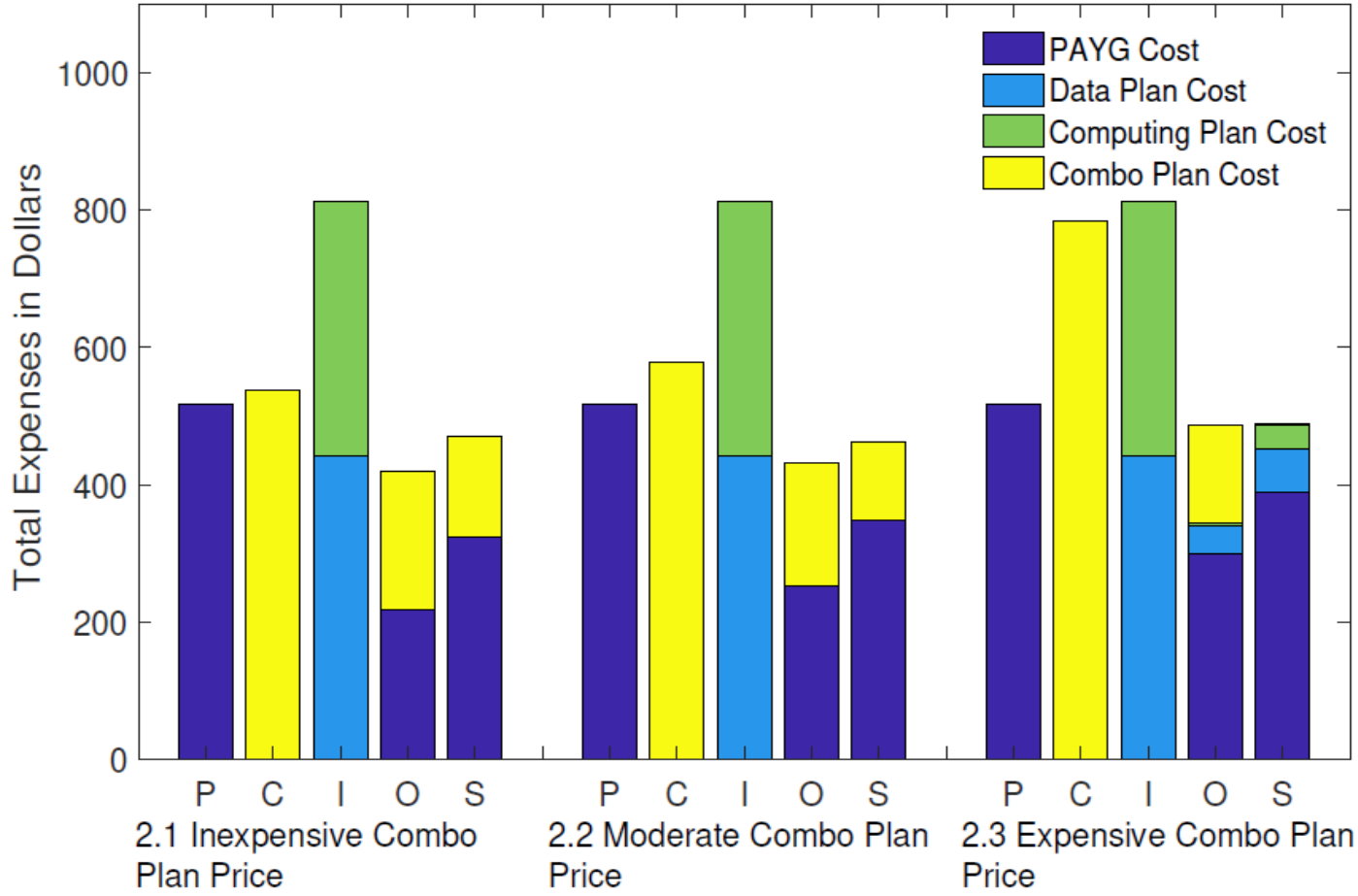
Trace Driven Performance Comparison

Individual plan (I): A data or computing plan is purchased when new data volume or computing workload arrives and the previous plan expires.

Offline (O): The user knows all task arrivals in advance and purchase a plan to make sure that the sum of further PAYG cost it will cover is greater than the plan upfront fee.

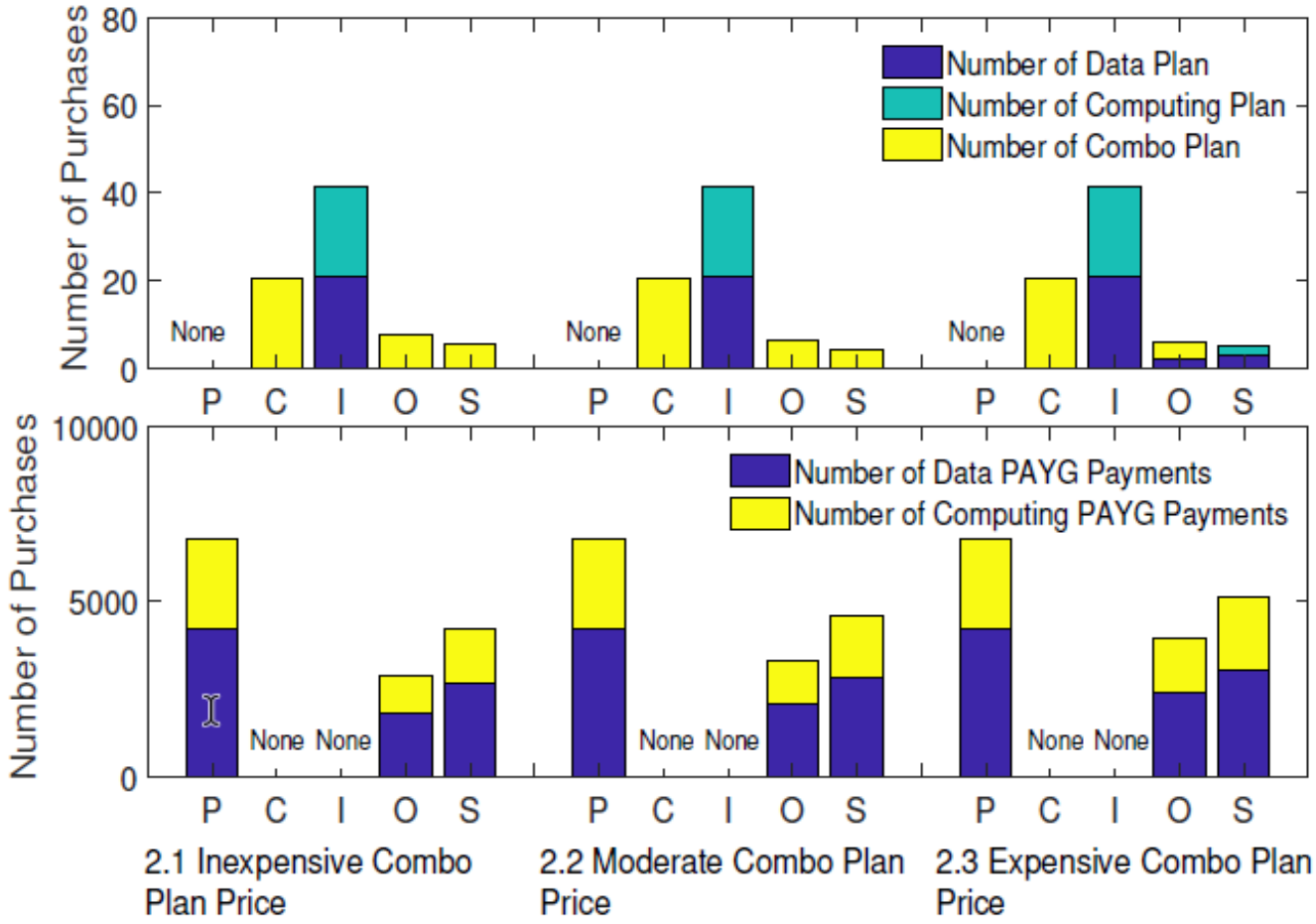
Simulation Result

P: PAYG only, C: Combo only, I: Individual plans, O: Offline, S: Smart online reservation algorithm



Simulation Result

P: PAYG only, C: Combo only, I: Individual Plan, O: Offline, S: Smart online reservation algorithm



Conclusion

- Online plan reservation problem for edge-computing services.
- Smart Online Reservation algorithm gives optimal competitive ratio among all deterministic algorithms.
- Critical price point: combo plan is more attractive compared with individual plans below this point.
- Feedback: combo plan about the edge computing

Shizhe Zang, PhD Candidate, University of Sydney

Email:shizhe.zang@sydney.edu.au

System Model and Problem Formulation

PAYG and Individual Plans

- PAYG (data) : satisfy required data volume a with unit data price p_1 and the cost is $a \times p_1$.
- PAYG (computing) : satisfy required computing workload b with unit computing price p_2 and the cost is $b \times p_2$
- Data plan: cover all the data volume in the following reservation period T with upfront fee C_1
- Computing plan: cover all the computing workload in the following reservation period T with upfront fee C_2.

System Model and Problem Formulation

Combo Plan

- Combo plan: cover all the data volume and computing workload in the following reservation period T with upfront fee C_3.

- Combo plan price C_3 : $C_3 > C_1$(data plan price), $C_3 > C_2$(computing plan price), $C_3 \leq C_1 + C_2$.