Optimizing Selection of Competing Services with Probabilistic Hierarchical Refinement

Tan Tian Huat

Competing Services – Example 1

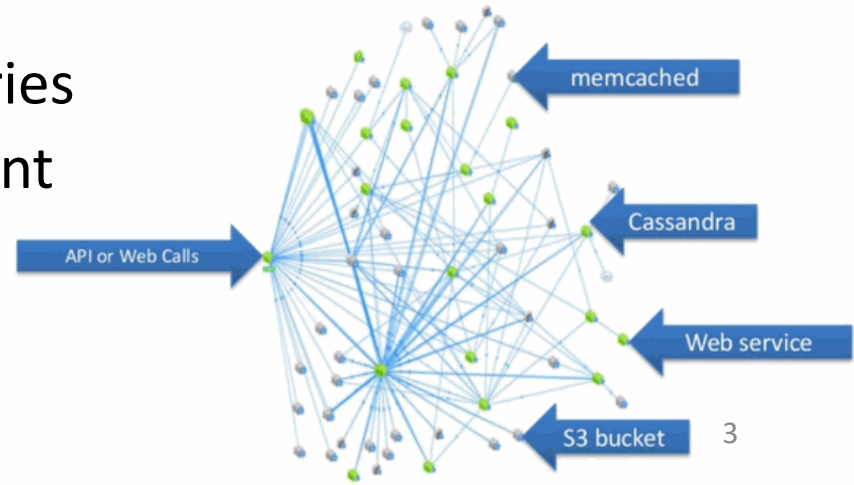
Car Booking Services

Hotel Booking Services

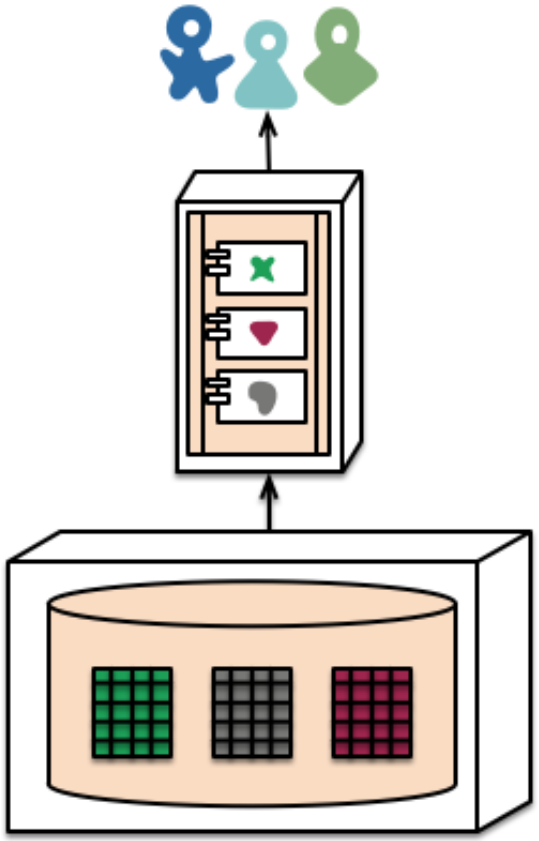
Competing Services – Example 2

- Netflix
  - A global provider of streaming movies and TV series
  - leverages **Microservice Architecture**

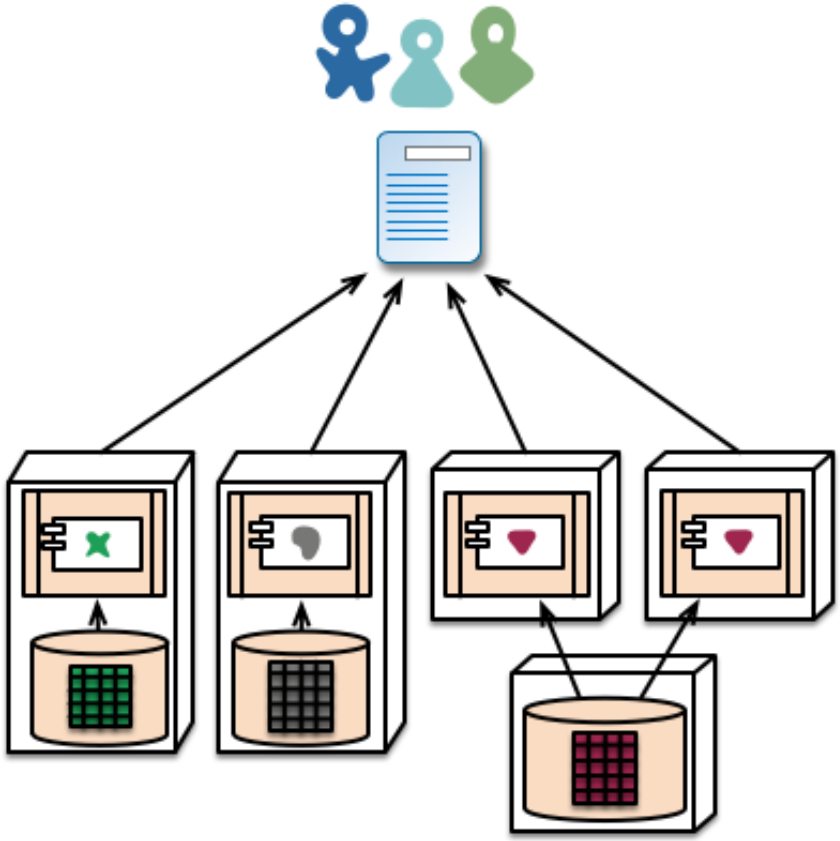
    (opposed to monolithic architecture)
  - Advantages:
    - Strong module boundaries
    - Independent Deployment
    - Technology Diversity



Competing Services – Example 2



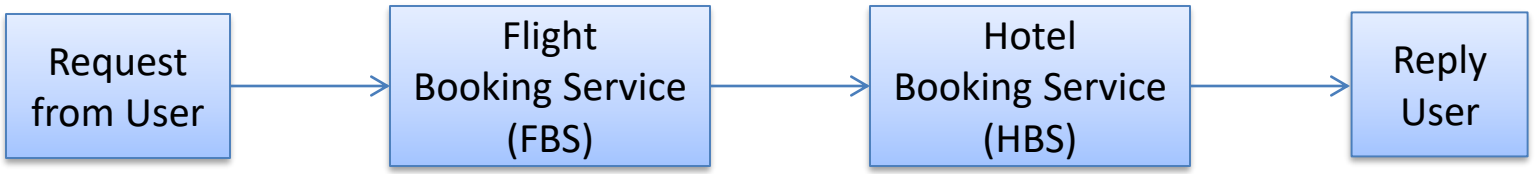
monolith - single database



microservices - application databases

Composite Service

Composite Service:
A service that leverages other existing services for achieving a business goal.



Travel Agency Composite Service (TAS)

Abstract service (e.g., Hotel Booking Service)

Concrete service (e.g., the Hilton Hotel booking service)

Abstract Composite Service
Concrete Composite Service

Quality of Service (QoS)

- **Type of QoS**
  - **Positive**
    - Availability
  - **Negative**
    - Cost, Response Time

Concrete Services	Response Times (ms)	Cost
f_1	200	10
f_2	100	20
f_3	50	30

- **QoS Constraints (can be due to Service Level Agreement)**
  - Response time < 50 ms, Cost < $20
- **QoS Optimality:** The best QoS based on user preference.

Optimal Service Selection

Given a composite service:

- For each **abstract service** (e.g., a hotel booking service)
- Select a **concrete service** (e.g., the Hilton Hotel booking service) for the abstract service at runtime.
- Maximize the QoS optimality.
- Satisfy all QoS constraints.

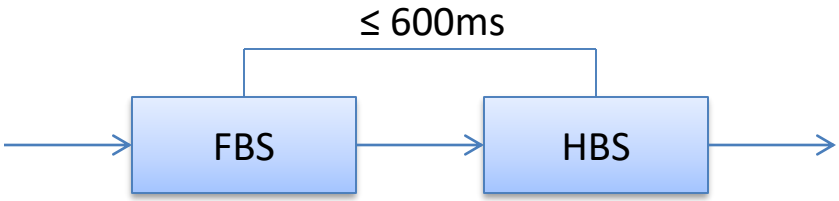
An NP Hard Problem!

Probabilistic Hierarchical Refinement (ProHR)

Probabilistic Hierarchical Refinement (ProHR)

Preprocessing

❖Unsatisfiable Services Pruning



Concrete Service for FBS
f_1, f_2, f_3, f_4

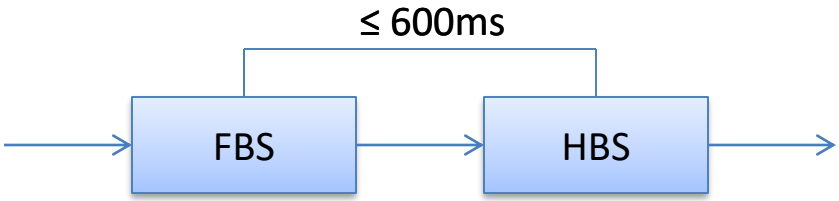
Concrete Service for HBS
h_1, h_2, h_3, h_4

Concrete Services	Response Times (ms)	Availability
f_1/h_1	100	0.85
f_2/h_2	300	0.92
f_3/h_3	500	0.95

Concrete Services for TAS

Preprocessing

❖Non-Skyline Services Pruning



Concrete Services	Response Times (ms)	Availability
f_1/h_1	100	0.85
f_2/h_2	300	0.92
f_3/h_3	500	0.95
f_4	600	0.94

Concrete Services for TAS

Concrete Service for FBS
f_1,f_2,f_3,f_4

Concrete Service for HBS
h_1,h_2,h_3,h_4

Probabilistic Hierarchical Refinement (ProHR)

Probabilistic Ranking

- Ranked the candidate concrete services for an abstract service according their
  - Local Optimality (L(s))
    - The local QoS optimality of a service
  - Constraint Satisfaction Probability (P(s))
    - How likely a service can satisfy the global constraints.

Concrete Services	Response Times (ms)	Availability	L(s)	P(s)	L(s)*P(s)
f_1/h_1	100	0.85	0.5	0.25	0.125
f_2/h_2	300	0.92	0.6	0.5	0.3
f_3/h_3	500	0.95	0.5	0.25	0.125

Probabilistic Ranking – Local Optimality

Concrete Services	Response Times (ms)	Availability	L(s)	P(s)	L(s)*P(s)
f_1/h_1	100	0.85	0.5	0.25	0.125
f_2/h_2	300	0.92	0.6	0.5	0.3
f_3/h_3	500	0.95	0.5	0.25	0.125

Service Ranking:

FBS	HBS
f_2	h_2
f_1	h_1
f_3	h_3

Why f_2/h_2 ranks the highest in L(s):

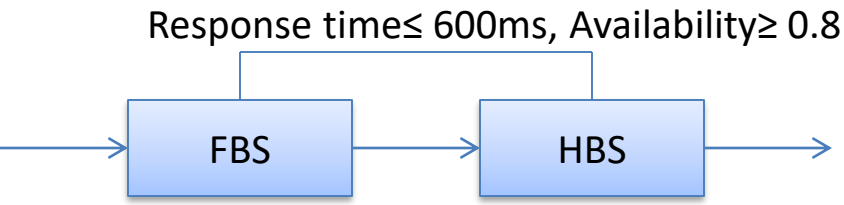
- f_3/h_3 has the worst response time
- f_1/h_1 has the worst availability
- f_2/h_2 is "just nice"

Probabilistic Ranking – Constraint Satisfaction Probability

Concrete Services	Response Times (ms)	Availability	L(s)	P(s)	L(s)*P(s)
f_1/h_1	100	0.85	0.5	0.25	0.125
f_2/h_2	300	0.92	0.6	0.5	0.3
f_3/h_3	500	0.95	0.5	0.25	0.125

Service Ranking:

FBS	HBS
f_2	h_2
f_1	h_1
f_3	h_3



Global Constraints

Response time: 300ms for each abstract service
Availability: 0.9 for each abstract service

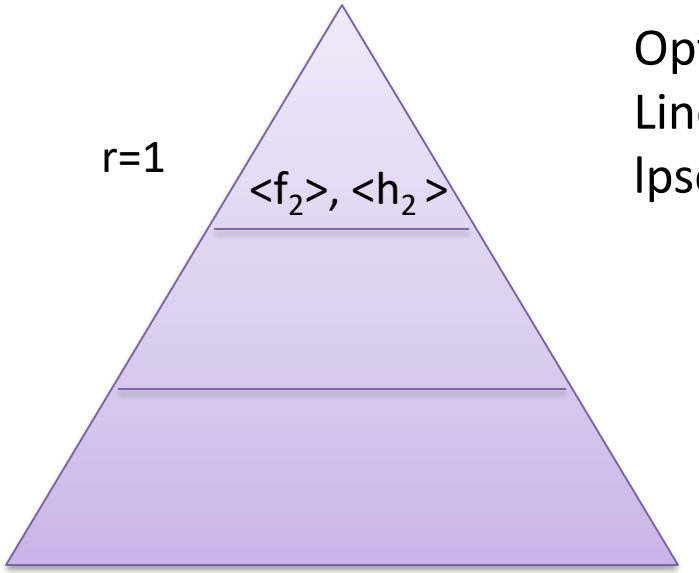
Local Constraints

Why f_2/h_2 ranks the highest in P(s):

- It is the only one that fit the local constraint well
- f_1/h_1 does not match for availability
- f_3/h_3 does not match for response time

Probabilistic Hierarchical Refinement (ProHR)

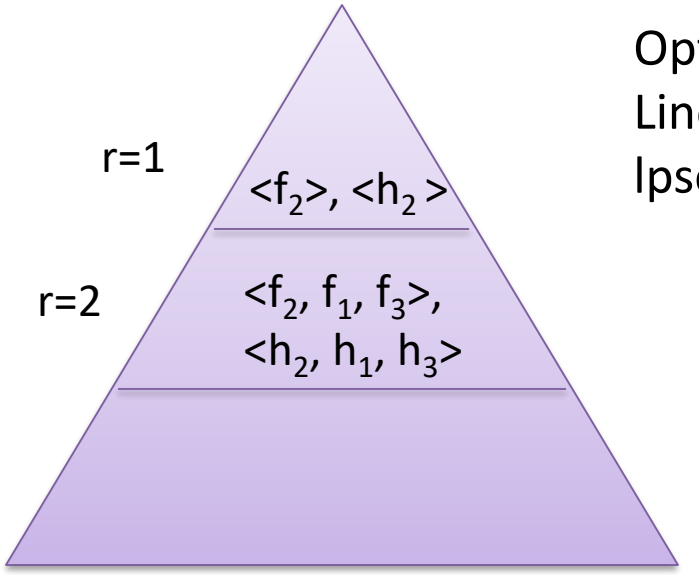
Hierarchical Refinement



Optimal selection using Mixed Integer Linear Programming (e.g., Gurobi, lpsolver)

FBS	HBS
f_2	h_2
f_1	h_1
f_3	h_3

Hierarchical Refinement



Optimal selection using Mixed Integer Linear Programming (e.g., Gurobi, lpsolver)

FBS	HBS
f_2	h_2
f_1	h_1
f_3	h_3

How many services to choose at each round?

- $\mathbb{P}(\mathbb{S})$ is the probability that given an abstract service, at least one concrete service successfully satisfies the global constraints.
- $\mathbb{P}(\mathbb{S})$> Threshold
- Threshold is increased with the number of round.

Hierarchical Refinement

r = 1

r = 2

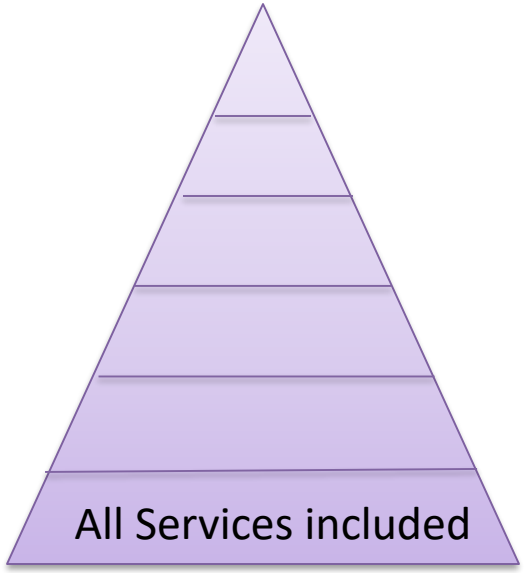
.

.

.

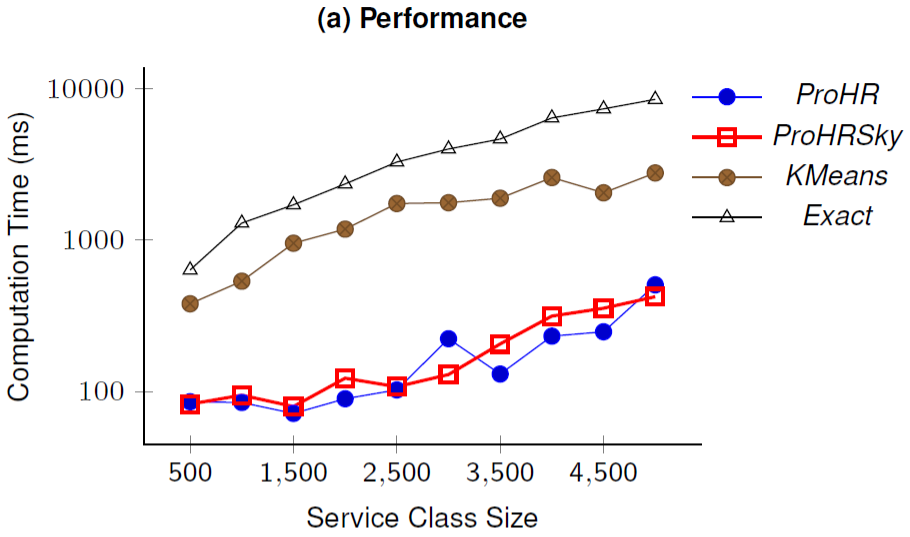
.

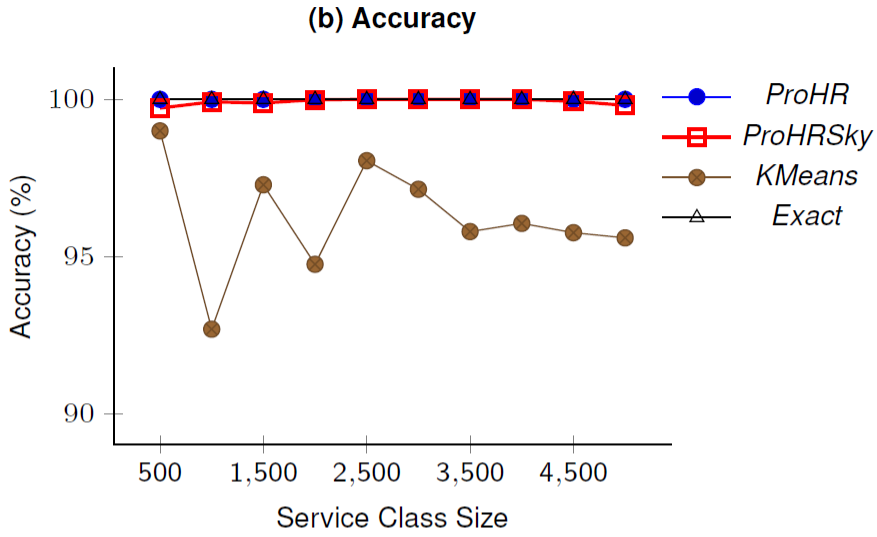
.

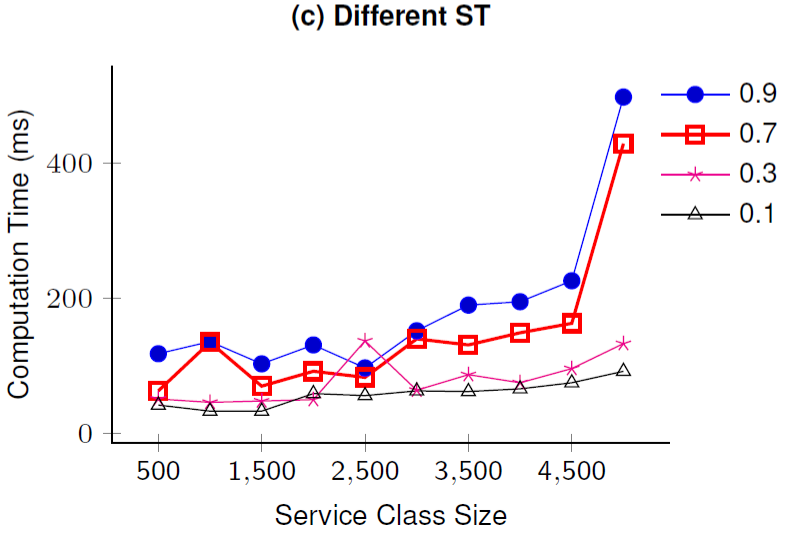


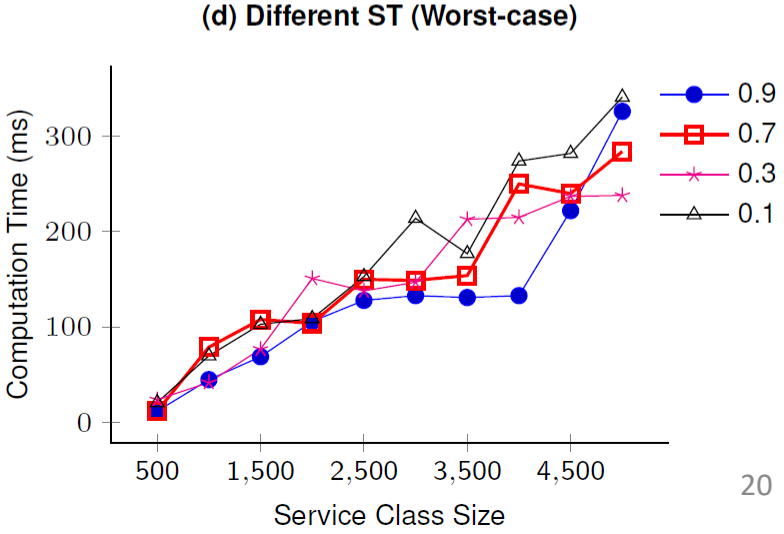
We will find a solution if there is one.

Experiments Result

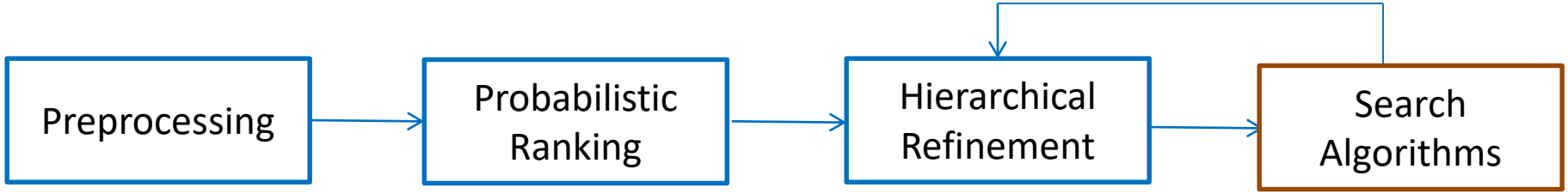








At a Higher Level



On a Higher Perspective

ProHR

1. Preprocessing -> Delete unsuitable candidate
2. Ranking -> Rank the candidates probabilistically
3. Hierarchical Refinement -> Select the ranked candidates probabilistically

Conclusion

❖Conclusion

- We propose Probabilistic Hierarchical Refinement (ProHR)
- On a higher level - an approach that can be integrated with searching techniques (e.g., MIP, EA) for NP-hard problems.

Questions?