# A Clique-Based Approach to Find Binding and Scheduling Result in Flow-Based Microfluidic Biochips

Trung Anh Dinh[†], Shigeru Yamashita[†], Tsung-Yi Ho[‡], Yuko Hara-Azumi[*]
[†] Ritsumeikan University, Japan
[‡] National Cheng Kung University, Taiwan
[*] Nara Institute of Science and Technology, Japan

# Agenda

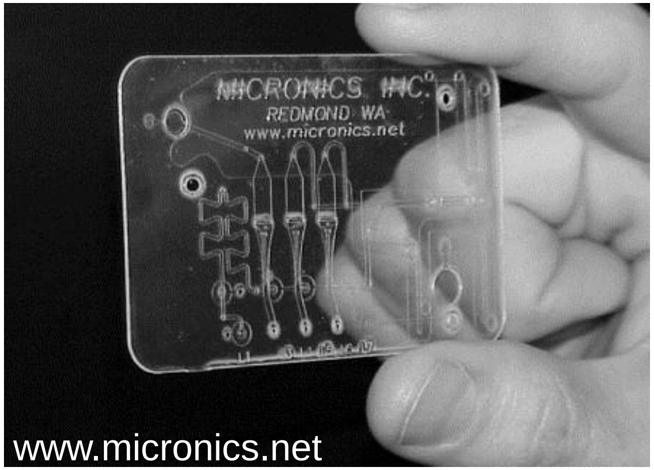
- Introduction
- Problem formulation
- Proposed method
- Experimental results
- Conclusions and future work

# Agenda

- Introduction
  - Flow-Based Microfluidic Biochips
    - Definition
    - Architecture model
  - Biochemical application model
- Problem formulation
- Proposed method
- Experimental results
- Conclusions and future work

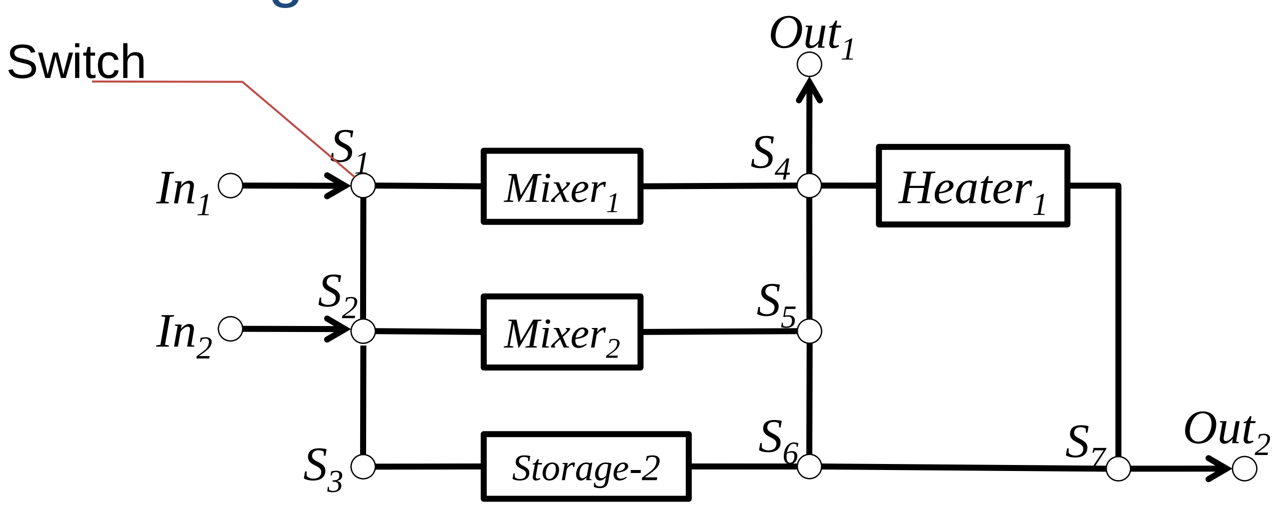
# Flow-Based Microfluidic Biochips

- Flow-Based Microfluidic biochips:
  - Based on the principle of continuous fluid flows in microchannels
  - Microvalves are used to manipulate fluid flows
  - Combination of multiple valves makes more complex units (switches, mixers, etc)



# Architecture Model

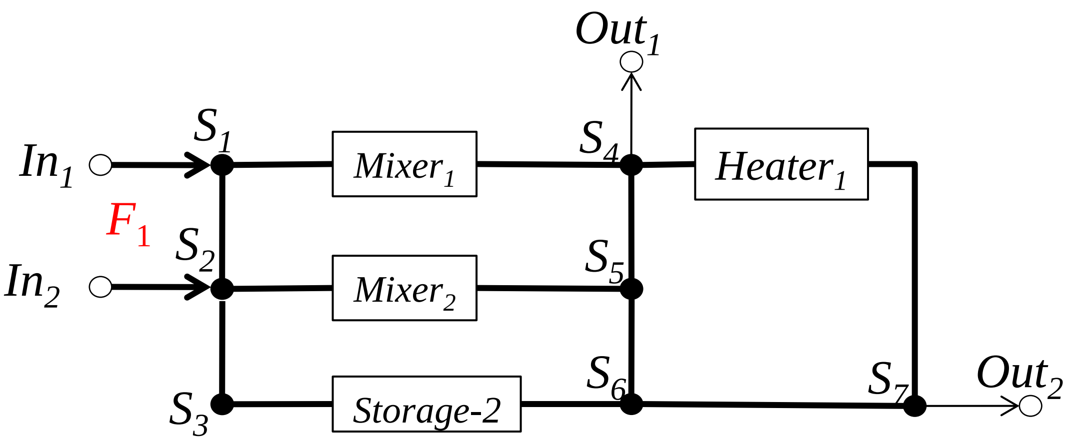
- A system-level model was formerly proposed [1]:
  - Topology graph
  - Set of vertices: microfluidic components $C_i$ (switches, mixers, storage reservoirs, etc)
  - Set of edges: directed channels between components




[1] W. H. Minhass, P. Pop and J. Madsen CASES' 2011

# Flow Paths

- A flow path $F$: a single path between two components. E.g., $F_1 = (In_1, S_1, Mixer_1)$
- However, not all routes between two arbitrary components are acceptable
  - E.g., there is no path from $Heater_1$ to $Mixer_1$
  - A storage is utilized as an intermediate transit



# Flow Paths

- *Routing constraints*: two paths which share common components as sources/sinks or common switches are *exclusive*
  - Exclusive paths cannot be executed simultaneously
  - E.g., $F_1$: $F_2 \vee F_3 \vee F_5 \vee F_{15}$

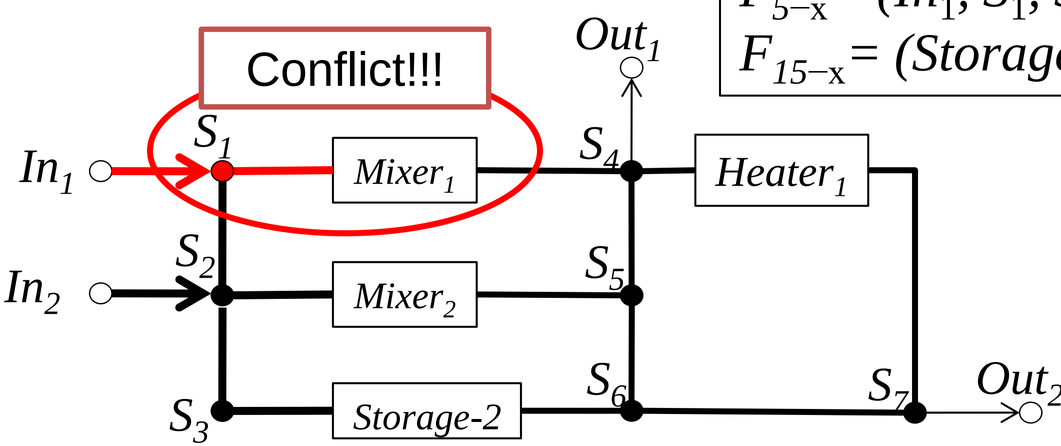
**$F_1$= ($In_1$, $S_1$, $Mixer_1$)**

$F_2$ = ($In_1$, $S_1$, $S_2$, $Mixer_2$)

$F_3$= ($In_2$, $S_2$, $S_1$, $Mixer_1$)

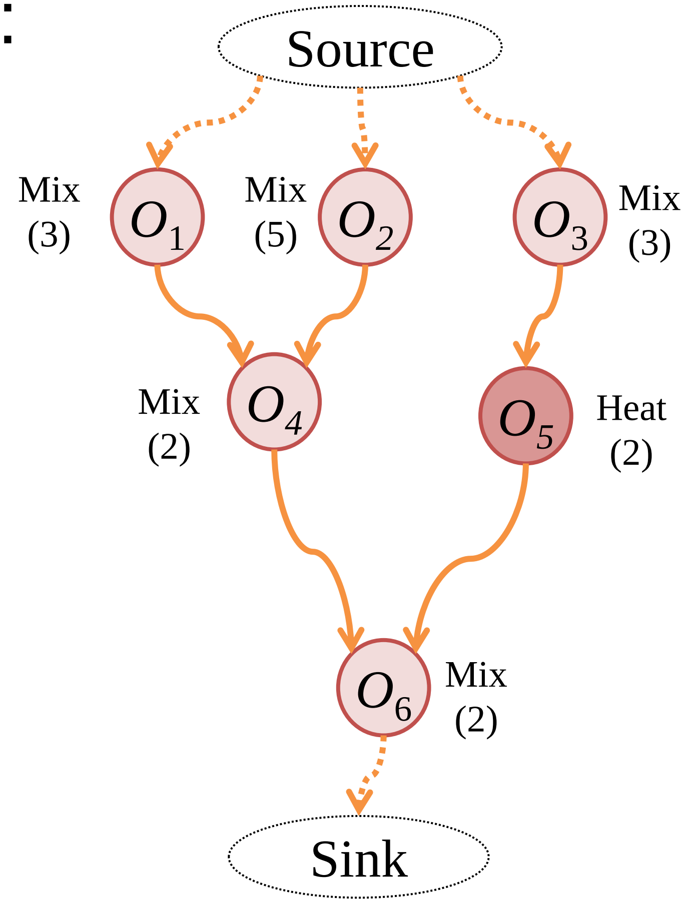
$F_{5-x}$= ($In_1$, $S_1$, $S_2$, $S_3$, Storage-2)

$F_{15-x}$= (Storage-2, $S_3$, $S_2$, $S_1$, $Mixer_1$)



# Biochemical Application Model

- Biochemical Application Model:
  - Sequencing graph
  - Each vertex $O_i$: represents one bioassay operation
  - Edges represent execution constraints
    - E.g., $O_1$ & $O_2$ must be completed before $O_4$

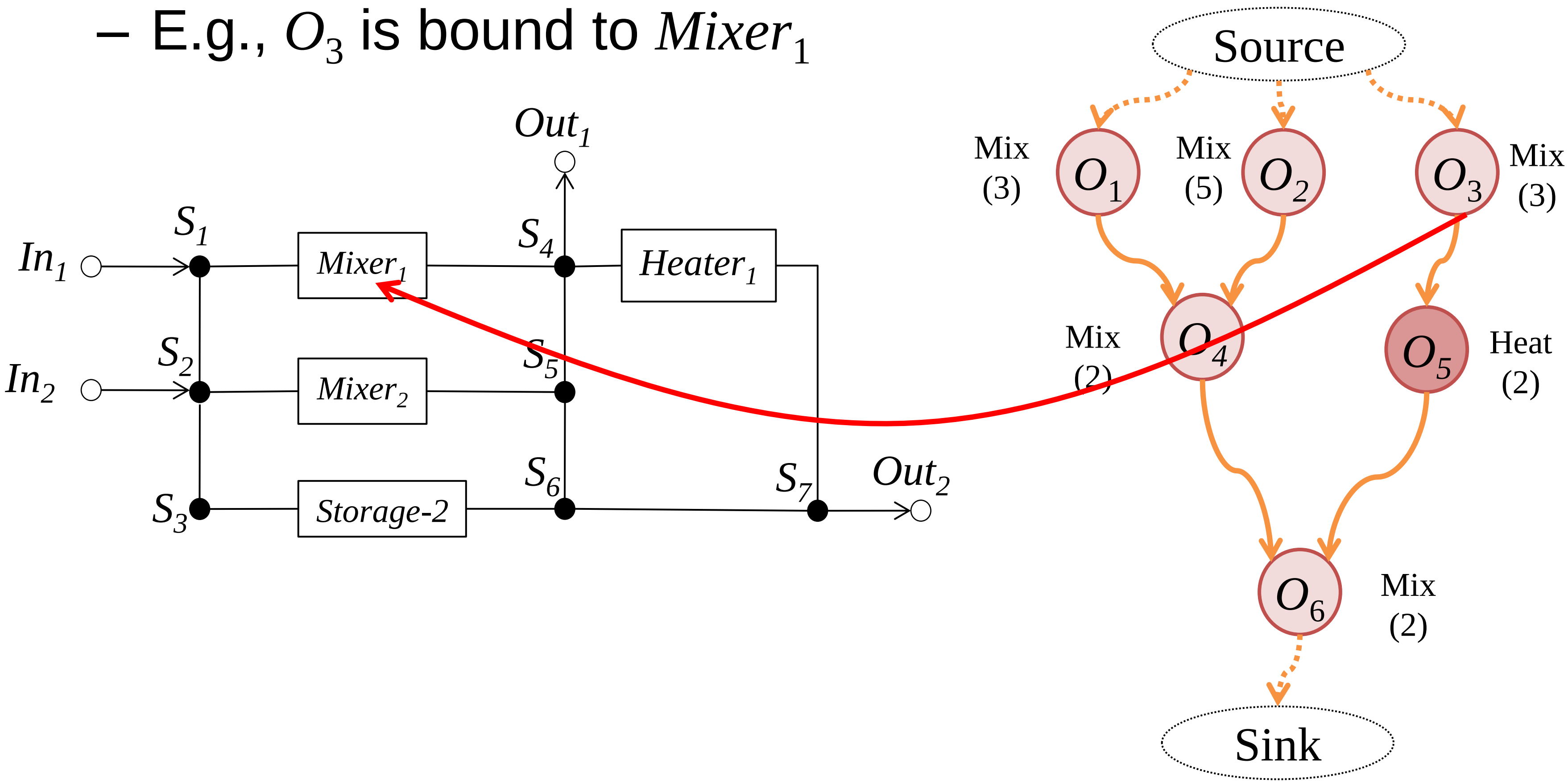


# Agenda

- Introduction
- Problem formulation
- Proposed method
- Experimental results
- Conclusions and future work

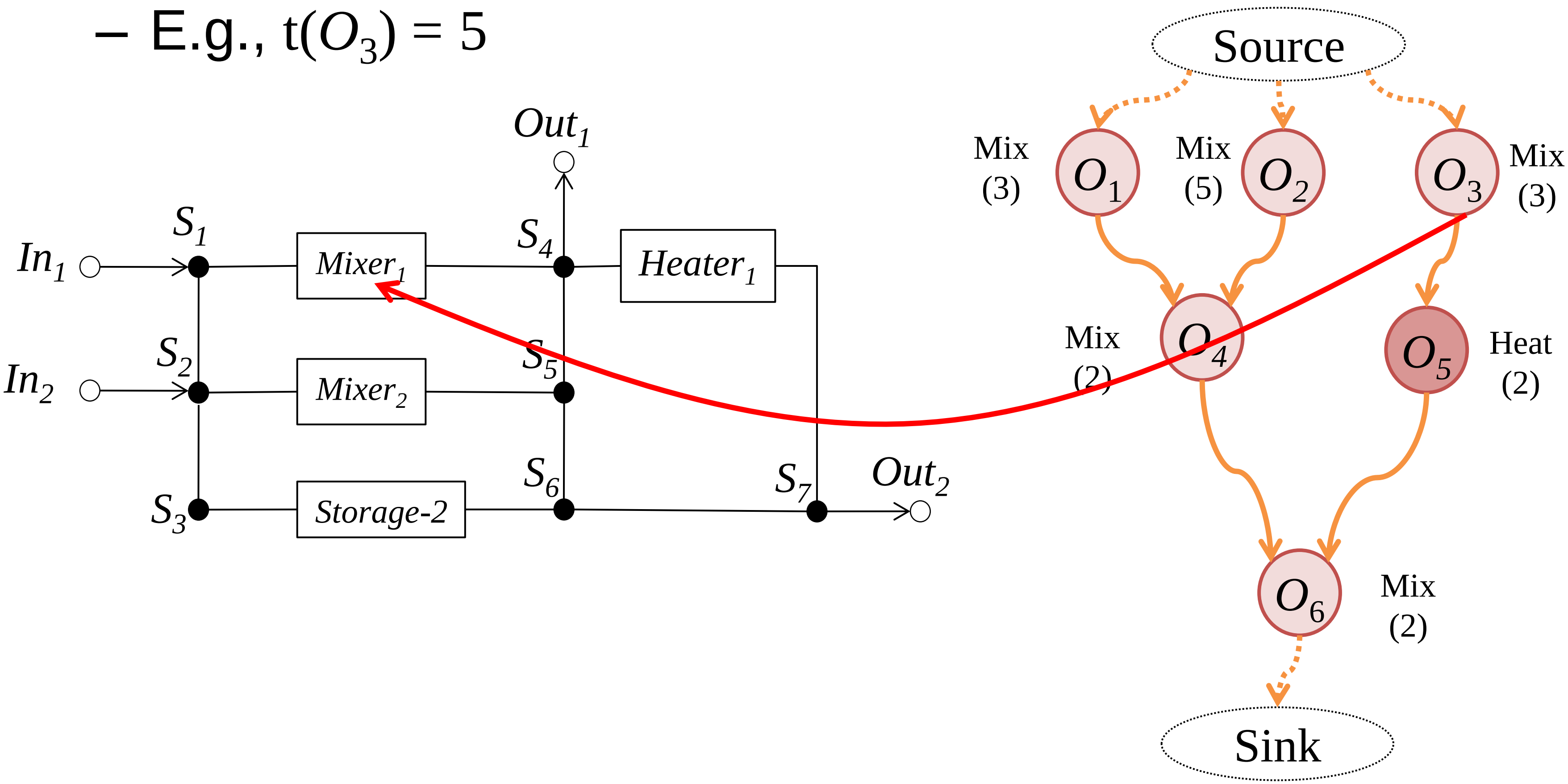
# Problem Formulation

- Binding: decides by which component an operation should be executed
  - E.g., $O_3$ is bound to $Mixer_1$

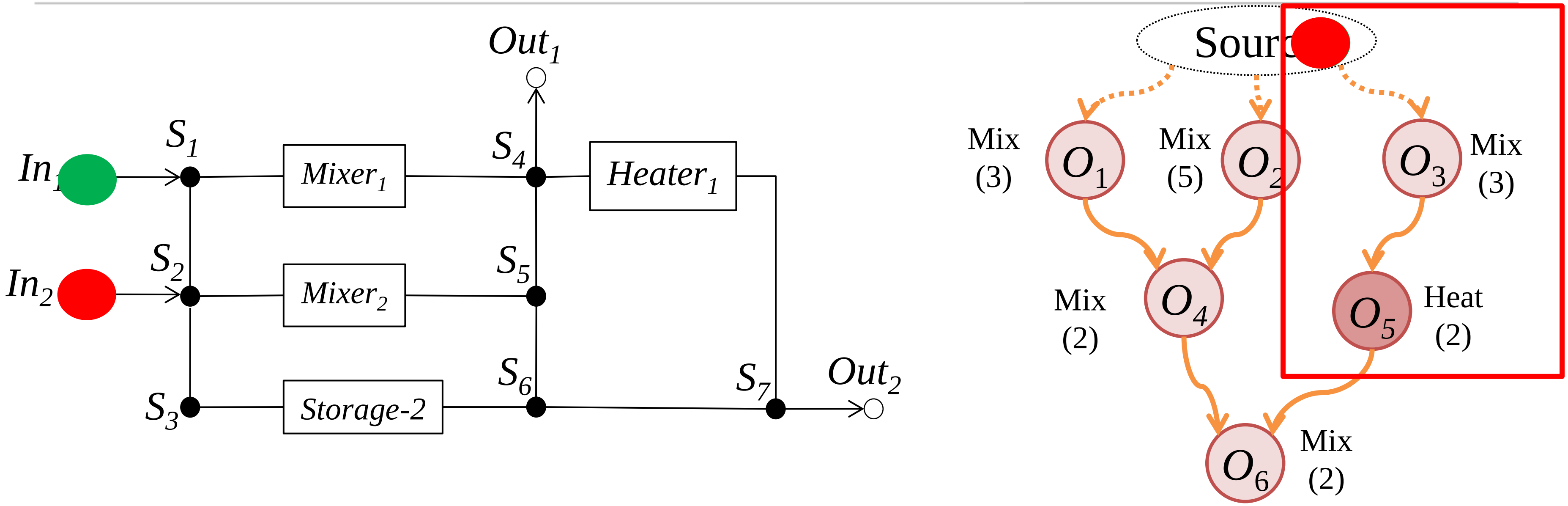


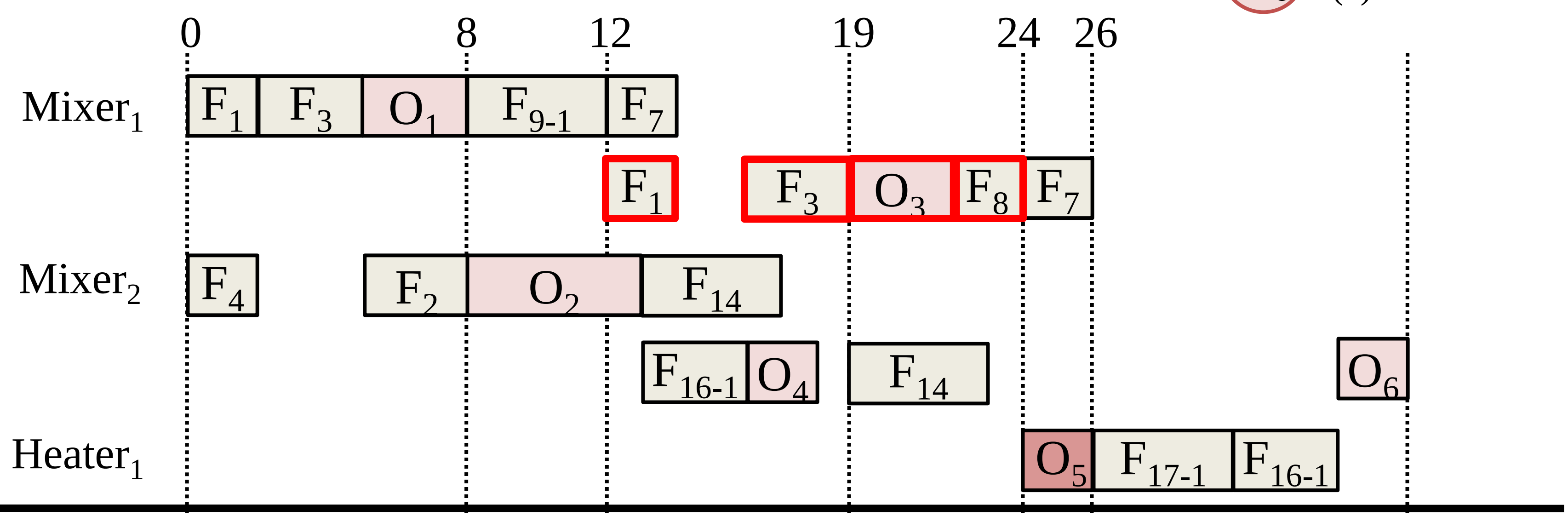
# Problem Formulation

- Scheduling: decides when each operation should be executed
  - E.g., t($O_3$) = 5



# Problem Formulation



# Problem Formulation

- Objective: find a binding and scheduling result which minimizes the execution time of a biochemical application
- Constraints: routing constraints and execution constraints
- Related work:
  - W. H. Minhass, P. Pop and J. Madsen CASES' 2011
    - List-based scheduling algorithm (LS algorithm)
  - W. H. Minhass, P. Pop and J. Madsen DTIP' 2012
    - Constraint Programming (routing constraints are not considered)

# Agenda

- Introduction
- Problem formulation
- Proposed method
- Experimental results
- Conclusions and future work

# Proposed Method

- The main reason which makes the problem different from conventional LSI binding and scheduling problems is routing constraints
- Previous work: LS algorithm (heuristic)
- Proposed method: transform to a maximum clique problem (MCP)
  - Routing constraints are formulated naturally
  - Satisfies the optimality of the problem

# Proposed Method

- Key idea:
  - For an upper bound of execution time, verify the existence of a binding and scheduling result
  - Binary search the upper bound to find the optimal execution time
- The existence of a valid binding and scheduling result $\leftrightarrow$ the validity of the maximum clique's cardinality in $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

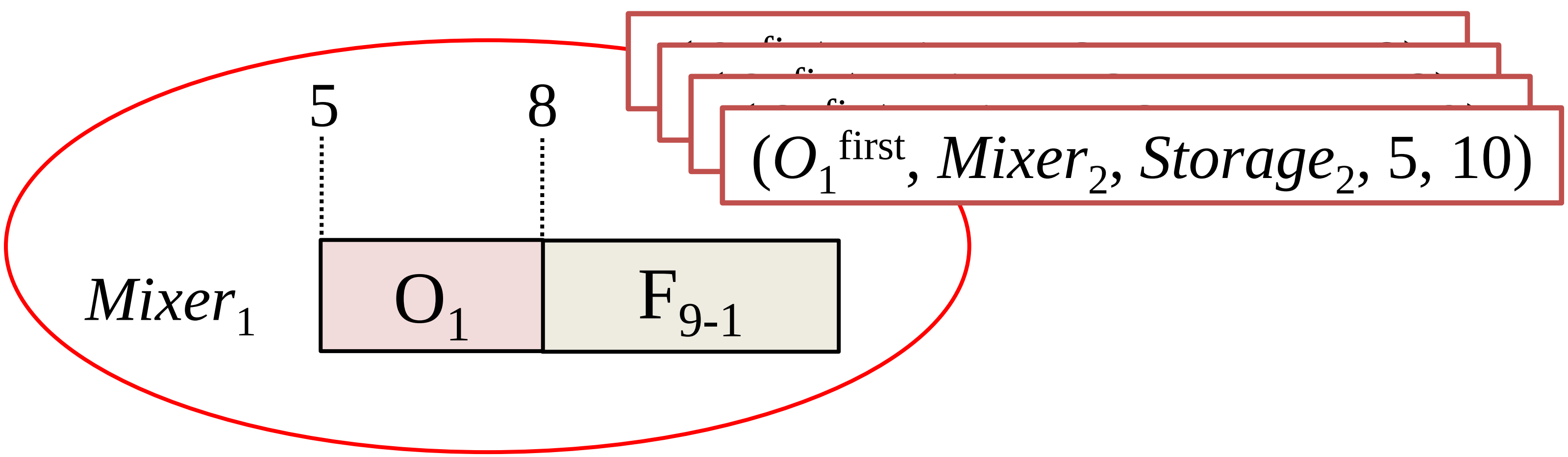
# Type 1-vertices

- For an operation $O_i$, what information do we need ?
  - E.g., $O_1$:
    - Which component to be bound ?
    - When to execute ?
    - When/where to go ?

E.g. $Mixer_1$

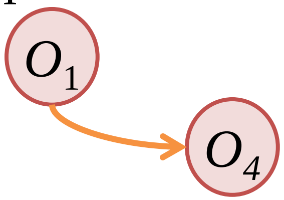
E.g. 5

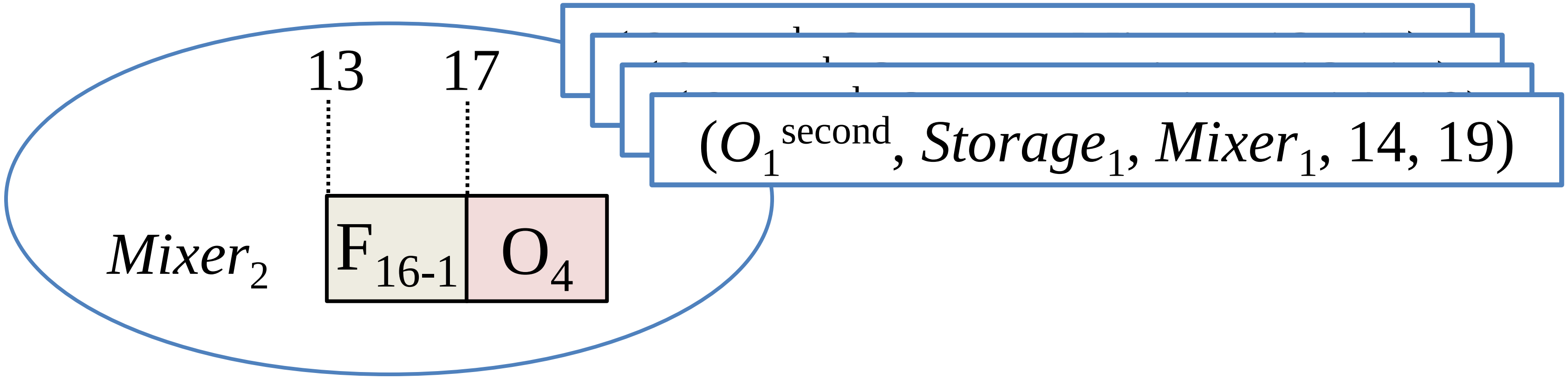
E.g. $8/Storage_1\ (F_9)$



# Type 2-vertices

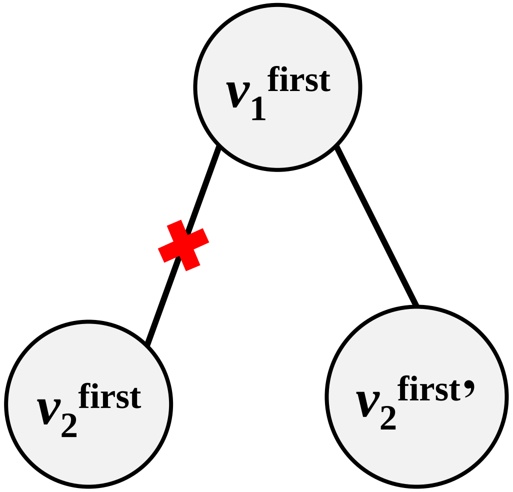
- For the output of an operation $O_i$, what information do we need ?
  - E.g., the output of $O_1$
    - Where is the output of $O_1$ now ? E.g. $Storage_1$
    - When/Where to go ? E.g. 13/$Mixer_2$ ($F_{16\text{-}1}$)
    - When to execute the next operation ? E.g. 17



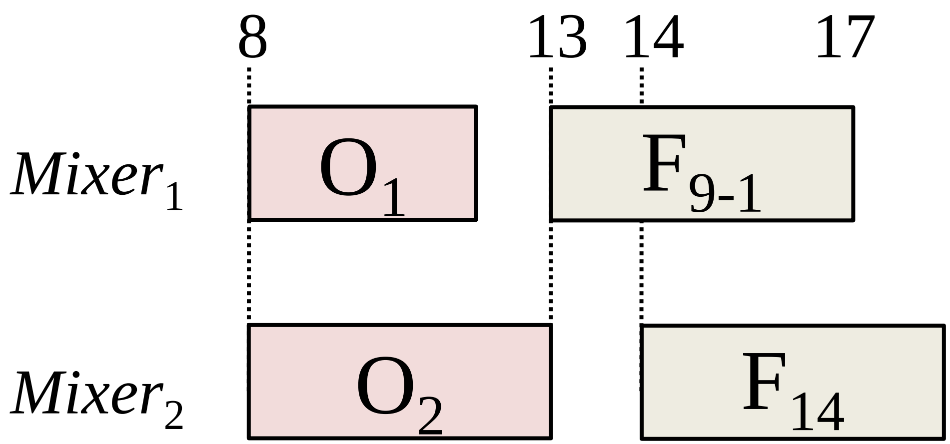



# Set of edges $\mathcal{E}$

**Routing constraints**




$v_1^{first} = (O_1^{first}, Mixer_1, Storage_1, 8, 13)$



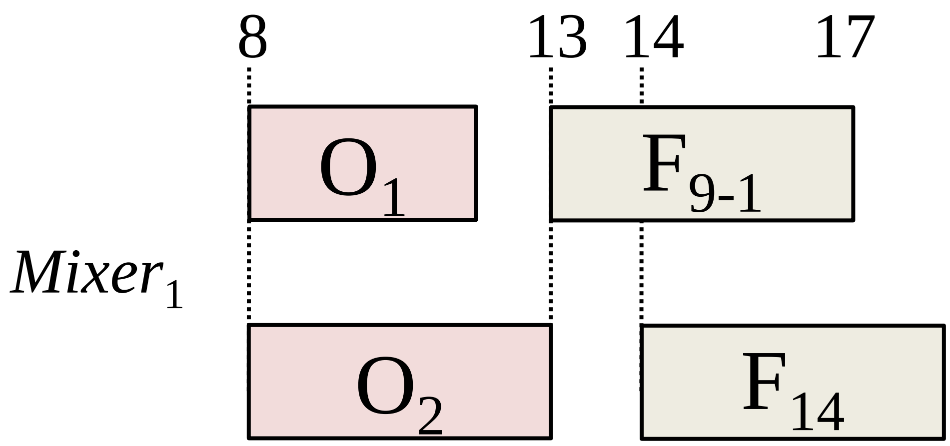

$v_2^{first\,'} = (O_2^{first}, Mixer_2, Out_2, 8, 17)$

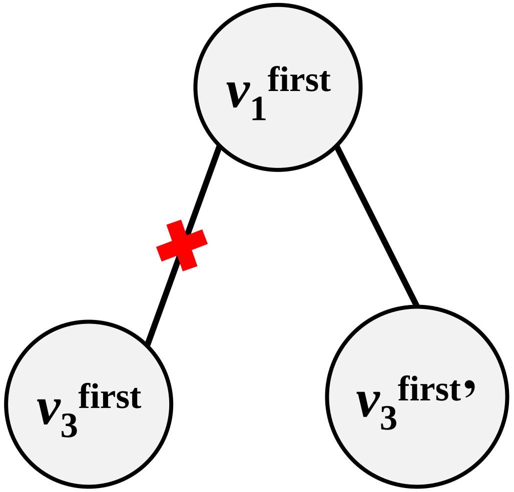
# Set of edges $\mathcal{E}$

Routing constraints

Resource constraints

$v_1^{\text{first}} = (O_1^{\text{first}}, Mixer_1, Storage_1, 8, 13)$



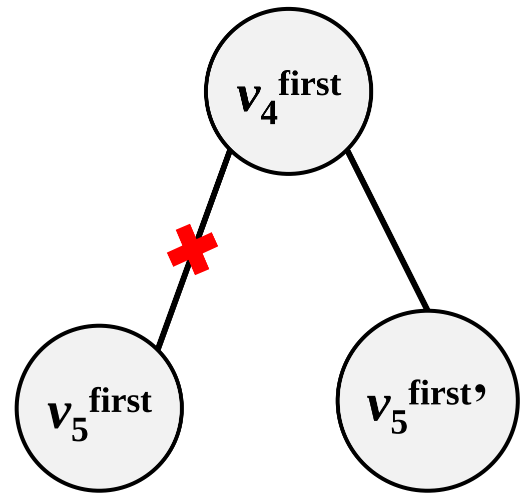

$v_3^{\text{first}\,'} = (O_2^{\text{first}}, Mixer_1, Out_2, 14, 20)$

# Set of edges $\mathcal{E}$

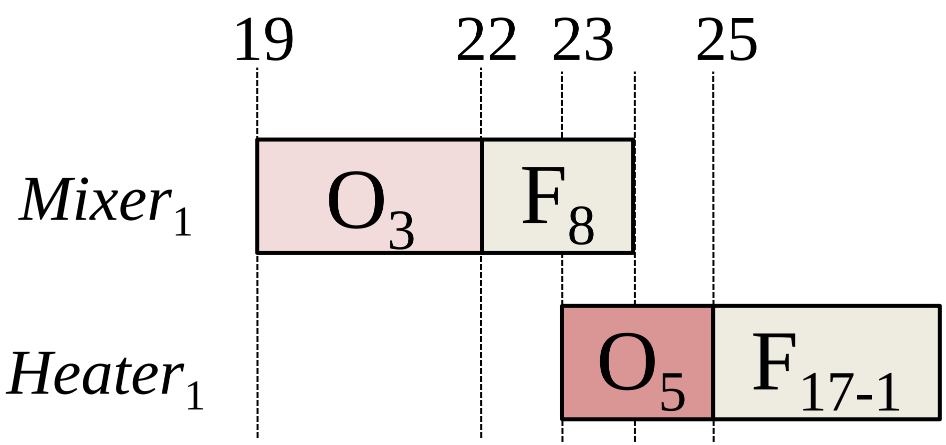
Routing constraints

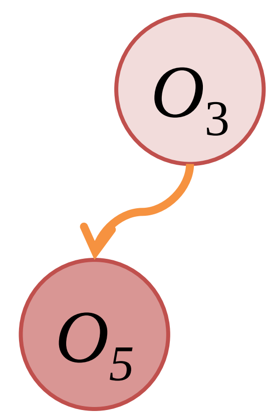
Resource constraints

Execution constraints



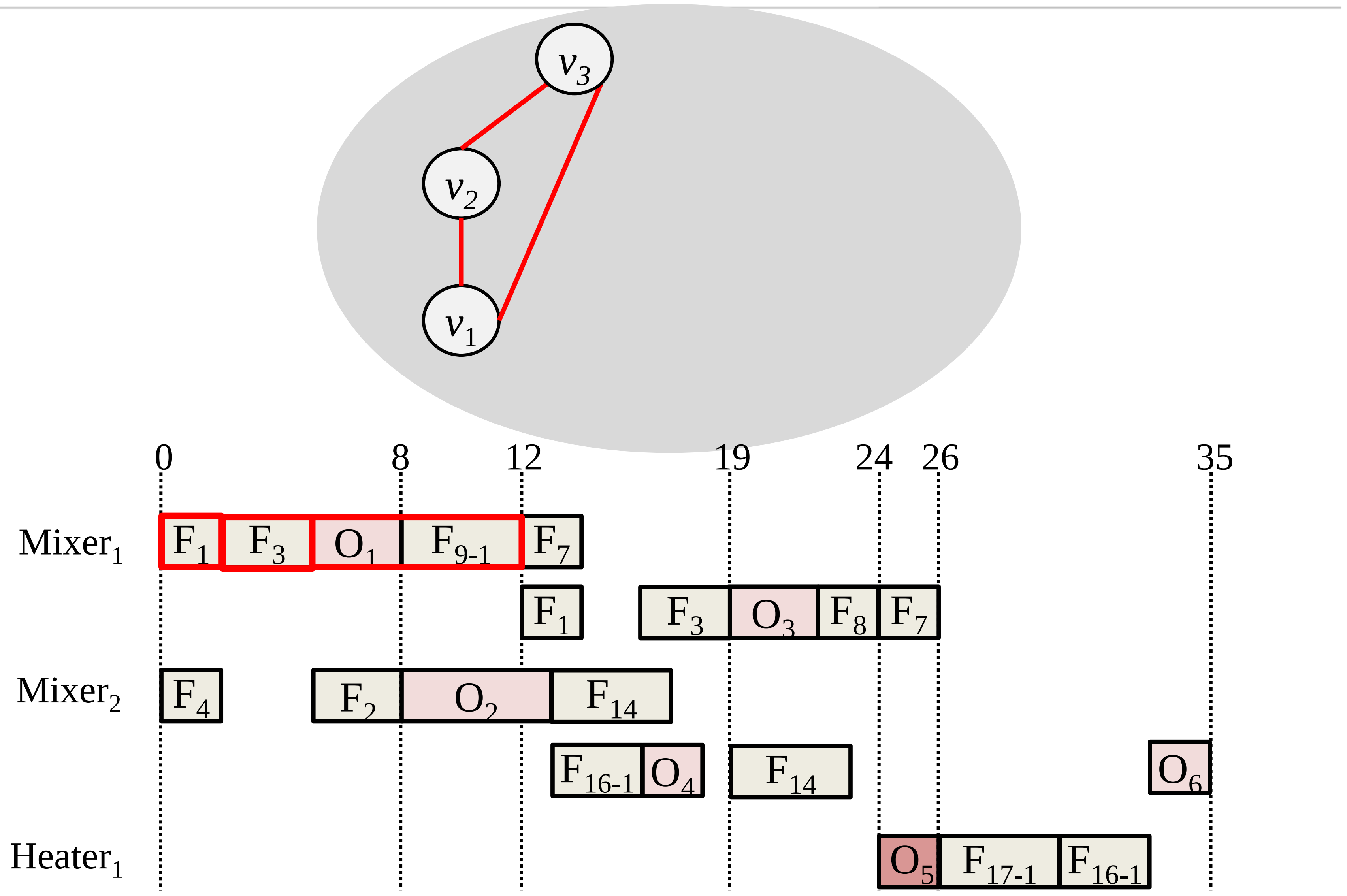

$v_4^{\text{first}} = (O_3^{\text{first}}, Mixer_1, Heater_1, 19, 22)$



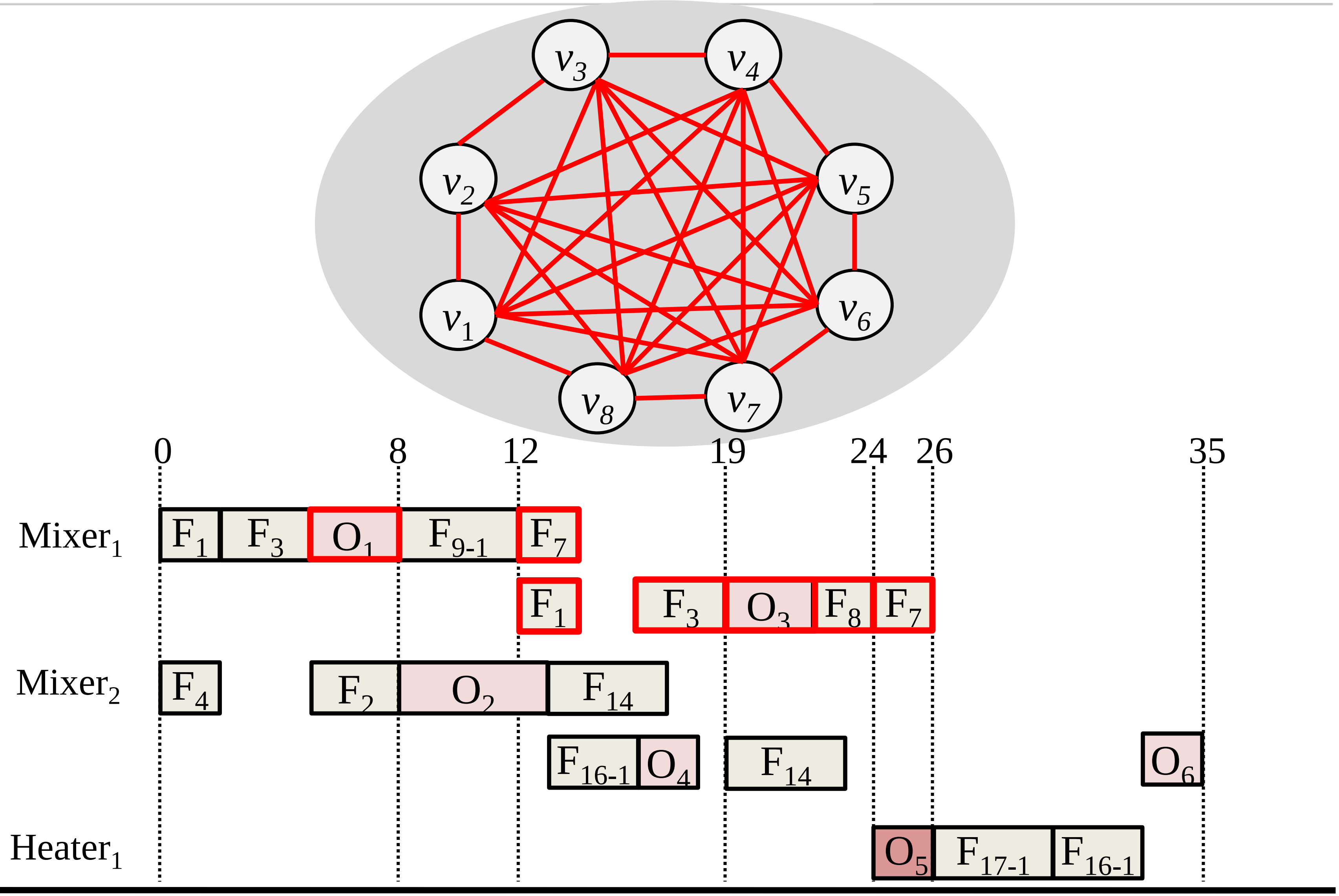




$v_5^{\text{first}'} = (O_5^{\text{first}}, Heater_1, Storage_1, 24, 26)$

# Graph: Example

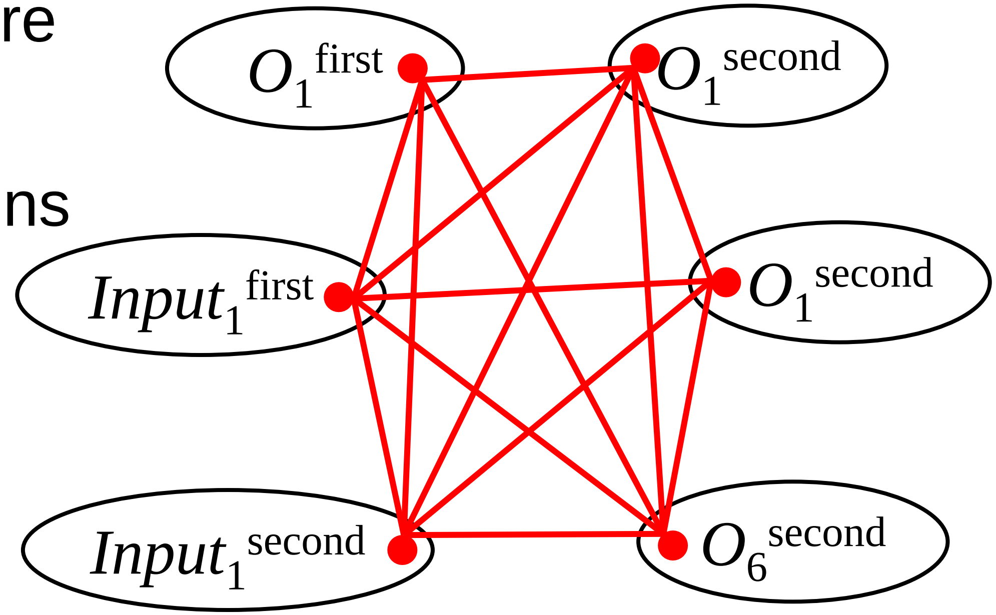


# Graph: Example



# Graph Properties

- A final result is a sub-graph, in which vertices are mutually connected, i.e., a clique
- Vertices are divided into groups:
  - $Input_1^{first}$, $Input_1^{second}$, $O_1^{first}$, $O_1^{second}$, … $O_6^{first}$, $O_6^{second}$
  - Vertices in one group are independent
  - Final result must contains one vertex from each group




Final result is even a maximum clique

# Proposed Method: Summary

- For a fix upper bound of execution time, the existence of a result can be verified
  - By checking the cardinality of the maximum clique
- Binary search the upper bound to find the optimal execution time

# Experimental Results

| | I/O Ports | #Operations | [2] | Ours |
|---|---|---|---|---|
| PCR | 2/2 | 7 | 30.3 | 29.8 (1.6%) |
| IVD | 2/2 | 12 | 31.3 | 30.5 (2.55%) |
| Synthetic Benchmark 1 | 2/2 | 10 | 56 | 51.5 (8.03%) |
| Synthetic Benchmark 2 | 2/2 | 6 | 37* | 35 (5.41%) |

* self-implemented program

[2] W. H. Minhass, P. Pop and J. Madsen CASE'2011 DTIP' 2012

# Conclusions and Future Work

- A new formulation for binding and scheduling problem in flow-based microfluidic biochips is proposed
  - Previous: LS algorithm (heuristic)
  - Ours: successfully transforms original problem to MCP and find optimal results
- Future work: as MCP is NP-Complete, more heuristic improvements are needed to deal with larger designs
- Acknowledgement: Wajid Hassan Minhass and Prof. Paul Pop for providing benchmarks and manuscripts for the comparative studies

THANK YOU FOR YOUR ATTENTION