On Reliability Hardening in Cyber-Physical Digital-Microfluidic Biochips

Speaker: Guan-Ruei Lu

Authors: Guan-Ruei Lu*, Guan-Ming Huang †, Ansuman Banerjee §, Bhargab B. Bhattacharya §, Tsung-Yi Ho ‡, Hung-Ming Chen ∗

∗ Institute of Electronics, National Chiao Tung University, Hsinchu, Taiwan

† Institute of Electrical and Computer Engineering, National Chiao Tung University, Hsinchu, Taiwan

‡ Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan

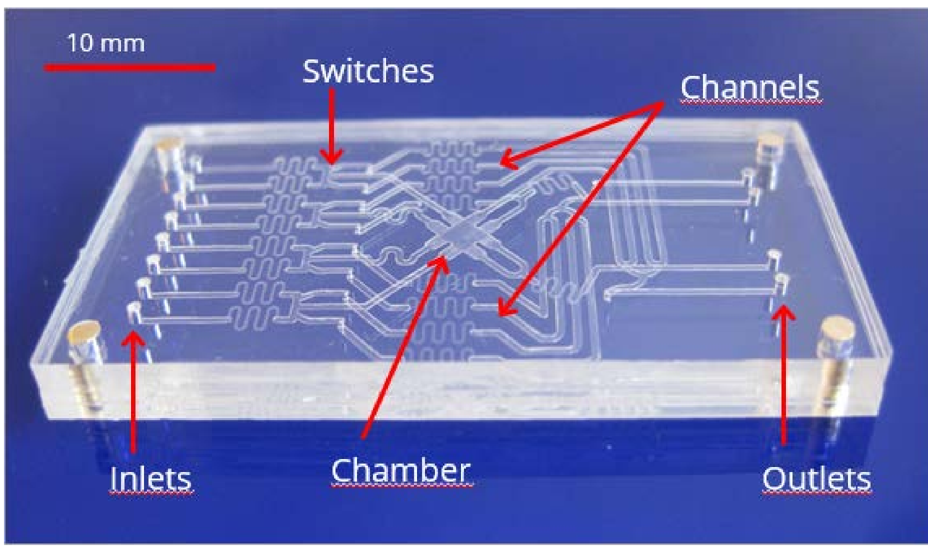
§ Advanced Computing & Microelectronics Unit, Indian Statistical Institute, Kolkata, India

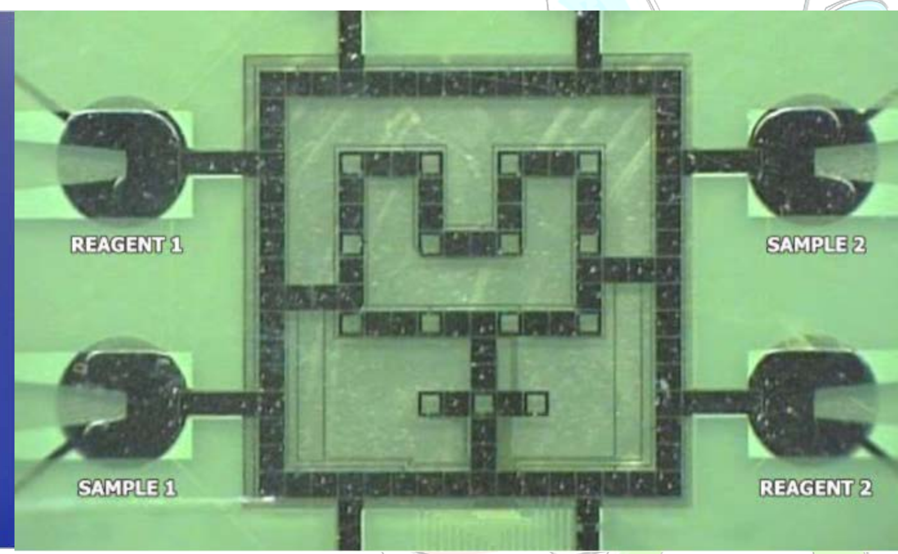
Outline

- Introduction
  - Microfluidic Biochip
  - Reliability Challenges
  - Cyber-Physical System (Self-Recovery & Checkpoints)
  - Summary of Checkpoint Problems
- Problem Formulations
- Proposed Algorithm
- Experimental Result

Microfluidic Biochips

- Flow-based biochips
  - Manipulation of continuous liquid through permanently-etched micro-channels
- Digital Biochips (DMFBs)
  - Manipulation of discrete droplets on an array of electrodes





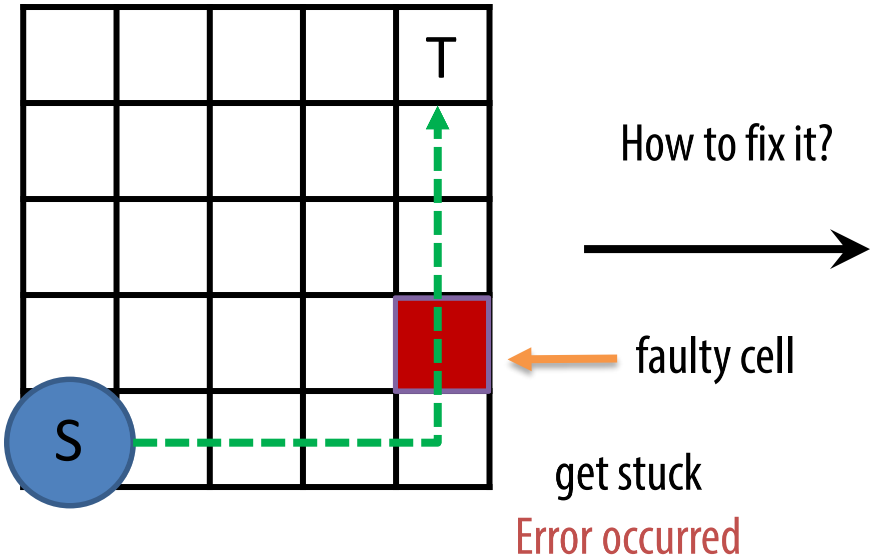
Digital Microfluidic Biochips (DMFBs)

- Two design types of digital microfluidic biochips
  - Application-specific biochips
  - General-purpose biochips

	Application-Specific	General-Purpose
*Hardware Resource (detector, heater, …so on)	User-defined (Customized)	Pre-defined, limited
Advantages	High performance, unit cost	Short design cycle, reusability, short time-to-market
Disadvantages	More complex design flow, long time-to-market	Performance, unit cost

Reliability Issues on DMFBs

- Unexpected errors on DMFBs
  - If an unexpected error occurs during the experiment, the outcome of the entire experiment will be incorrect.



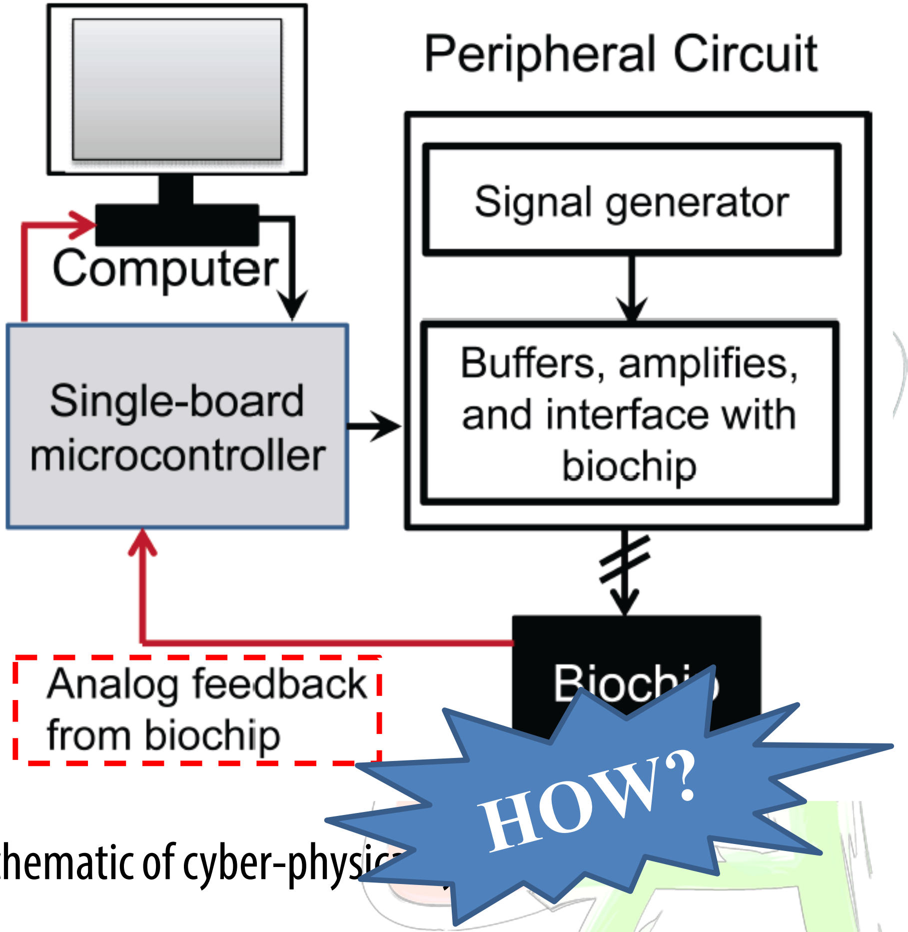
Cyber-physical system has been presented to fix this issue.

A case about unexpected error

Cyber-Physical System

- What can it do?
  - Real-time **monitoring**
  - Real-time **recovering errors**

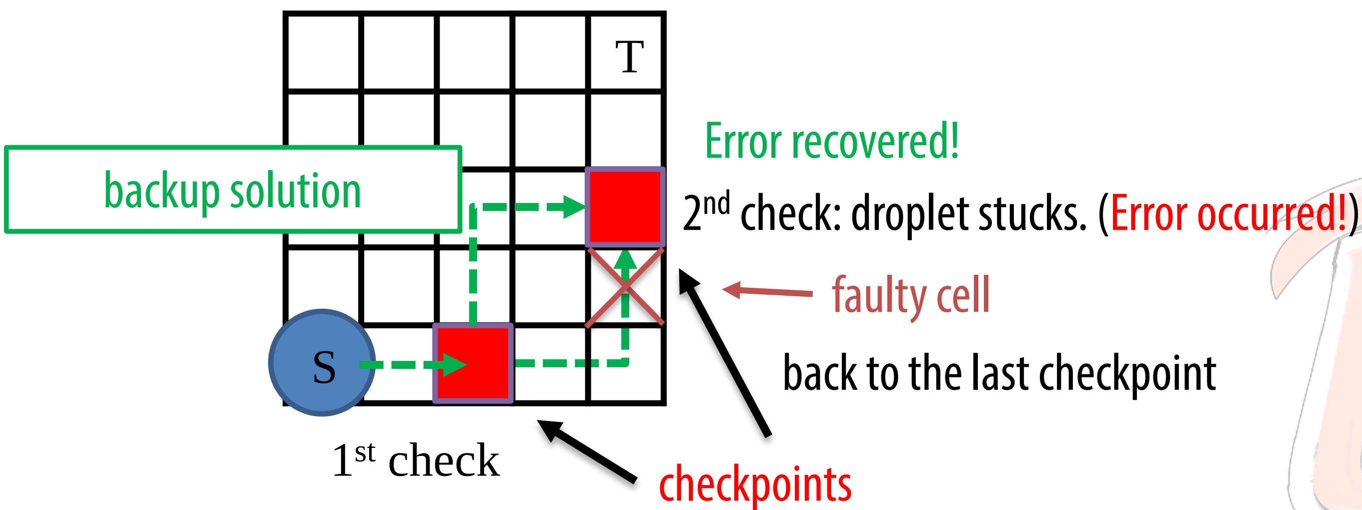
It can recovery the error by referring the feedback signal from the sensor on biochip (checkpoint).



The schematic of cyber-physic

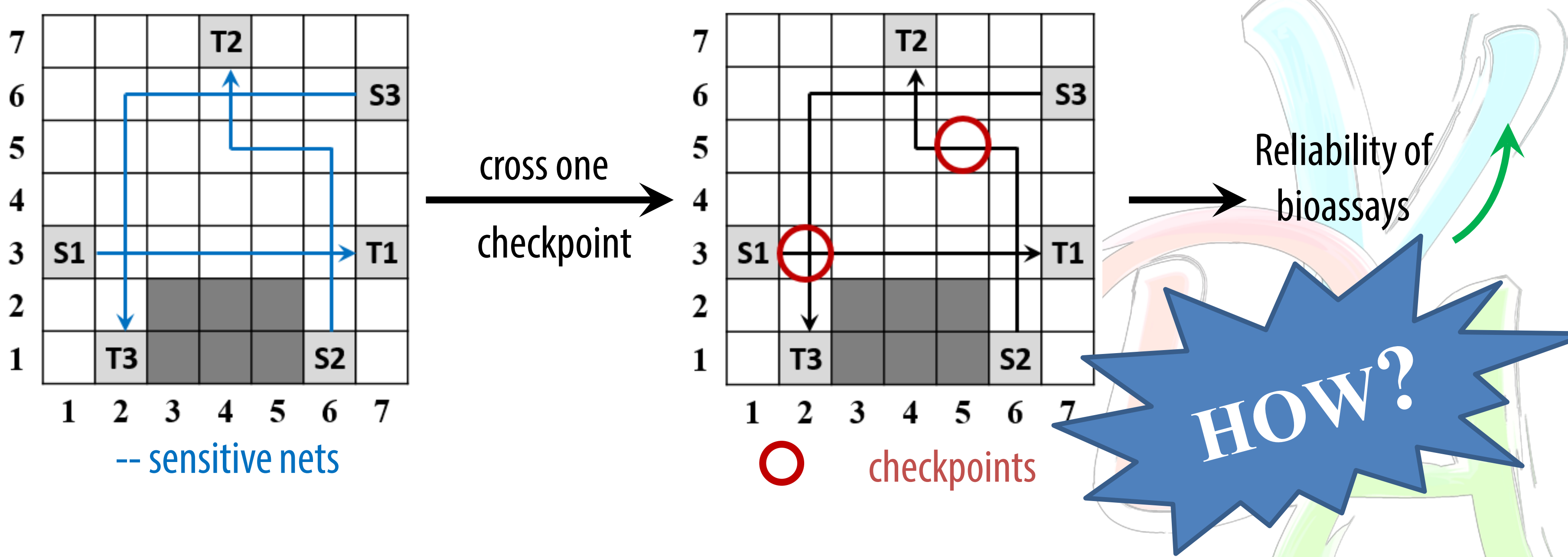
Checkpoint and Self-Recovery

- **Checkpoint** is a specific electrode on biochip, which can provide sensor data (size, concentration, ...etc).
- **Self-recovery** is to provide a backup plan to bypass the faulty cells.
  - Increase the **reliability** of bioassays



Motivation

- Each <u>highly sensitive</u> net should have <u>self-recovery ability</u> to increase the reliability of bioassay
  - Each sensitive droplet should cross at least one checkpoint.
  - Finally, the reliability of bioassays can be increased.



Checkpoint Problems on Application-Specific DMFBs (AS-DMFBs)

- More checkpoints lead to more cost.
  - How to minimize the number of checkpoints?
  - How to decide the location of checkpoints?

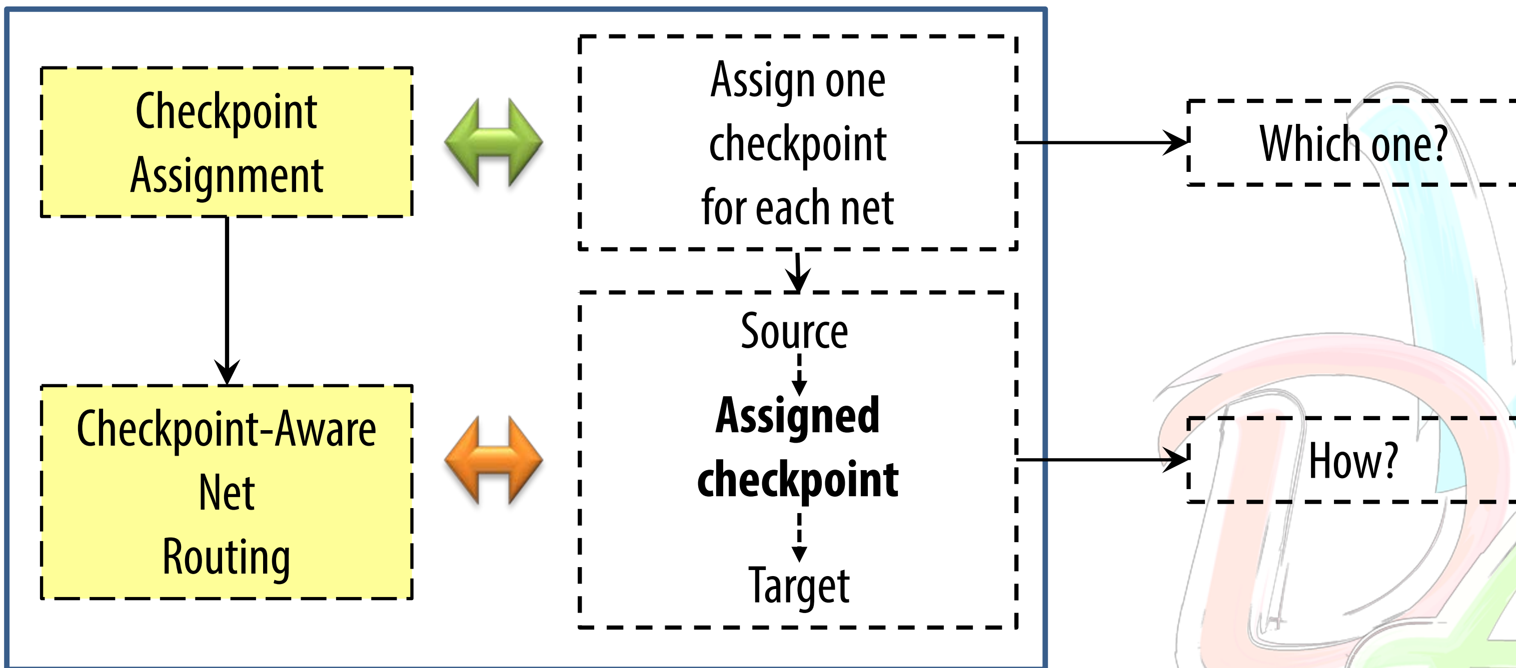
Design Flow

Checkpoint Minimization	⟺	Minimize **#checkpoint**
↓		↓
Checkpoint Allocation (placement)	⟺	Determine the **locations of checkpoints**

Checkpoint Problems on General-Purpose DMFBs (GP-DMFBs)

- There is a limited number/pre-located checkpoint.

Design Flow



Comparison between Two Types of Biochip Design

- Our contribution
  - Checkpoint minimization & checkpoint allocation.
  - Checkpoint assignment & checkpoint routing

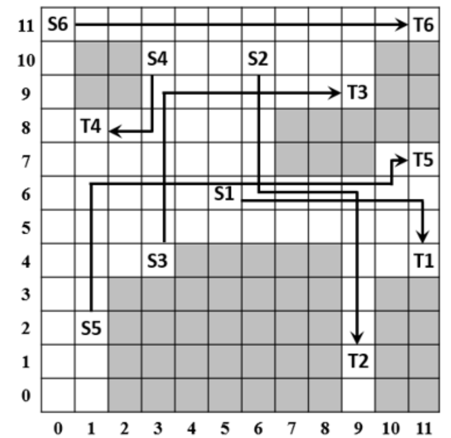
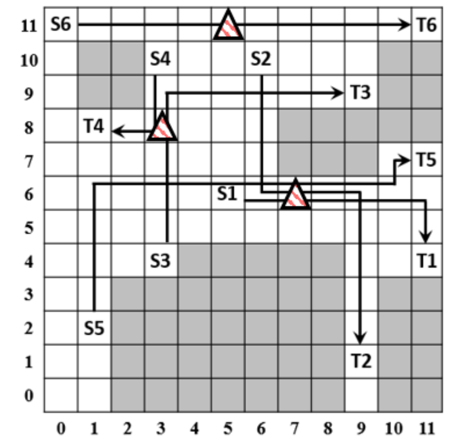
		Application-Specific	General-Purposed
Checkpoint Resource		Customized	Pre-defined
Goals	#Checkpoints Minimization	V	
	Checkpoint Allocation (Placement)	V	
	Checkpoint Assignment		V
	Checkpoint Routing		V

Outline

- Introduction
- Problem Formulations
  - Application-Specific DMFBs
  - General-Purpose DMFBs
- Proposed Algorithm
- Experimental Result

Problems in Application-Specific DMFBs

- **Features:** Checkpoints can be arranged.
- **Objective:** Minimize the number of required checkpoints.
- **Constraint:** Each sensitive droplet needs to cross at least one checkpoint.

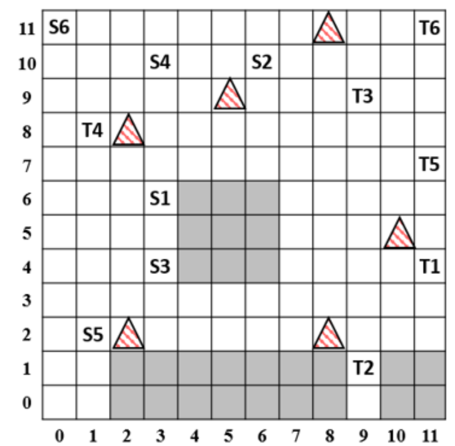
Input	Output
Routing solution	#Checkpoints & Locations of checkpoints
	

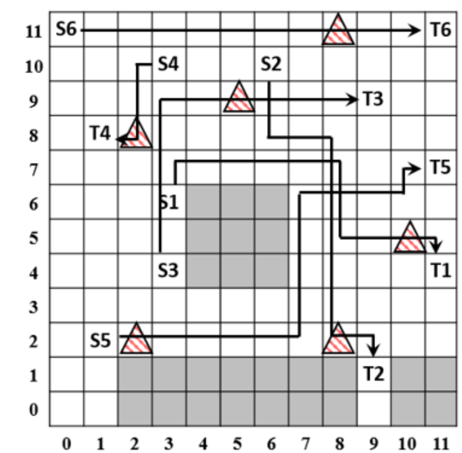
checkpoint

Problems in General-Purpose DMFBs

- **Features:** Checkpoints are <u>pre-located</u>.
- **Objective:** Minimize the Max. arrival time T_{max}.
- **Constraint:** Each sensitive droplet needs to cross at least one checkpoint.

Input	Output
Locations of checkpoints & Netlist	Routing solution





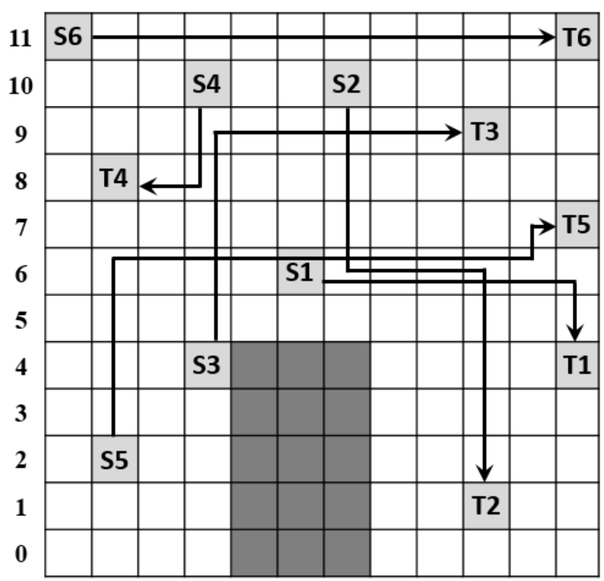
checkpoint

Outline

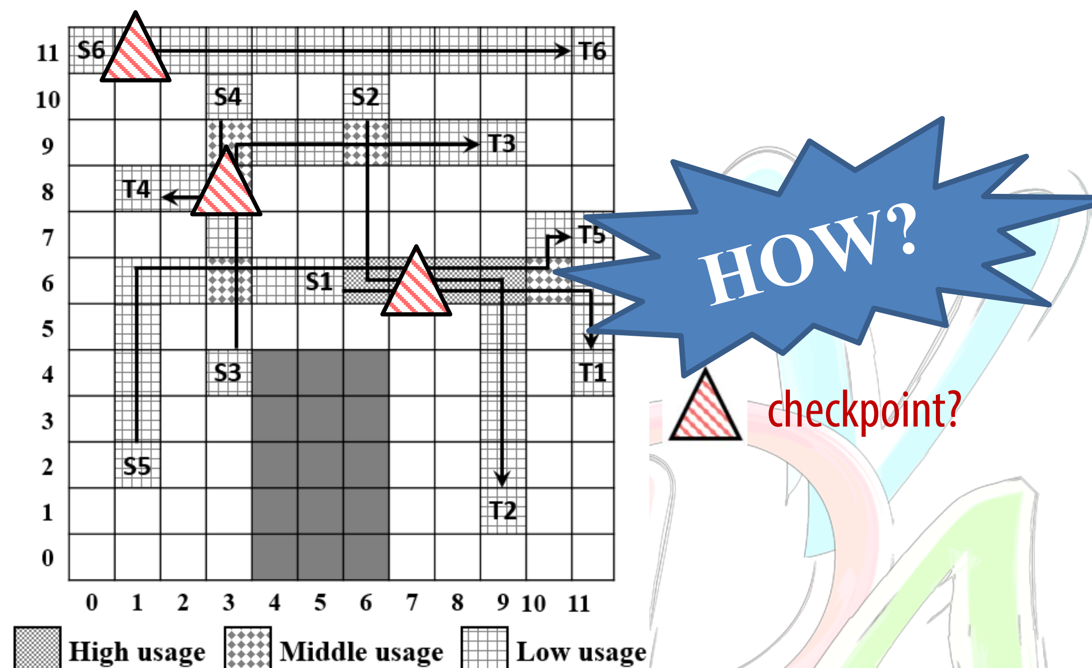
- Introduction
- Problem Formulations
- Proposed Algorithm
  - Checkpoint Minimization for Application-Specific DMFBs
  - Checkpoint-Aware Routing for General-Purpose DMFBs
- Experimental Result

Checkpoint Minimization

- Intuitively, we might assign highly used cells to be checkpoints.



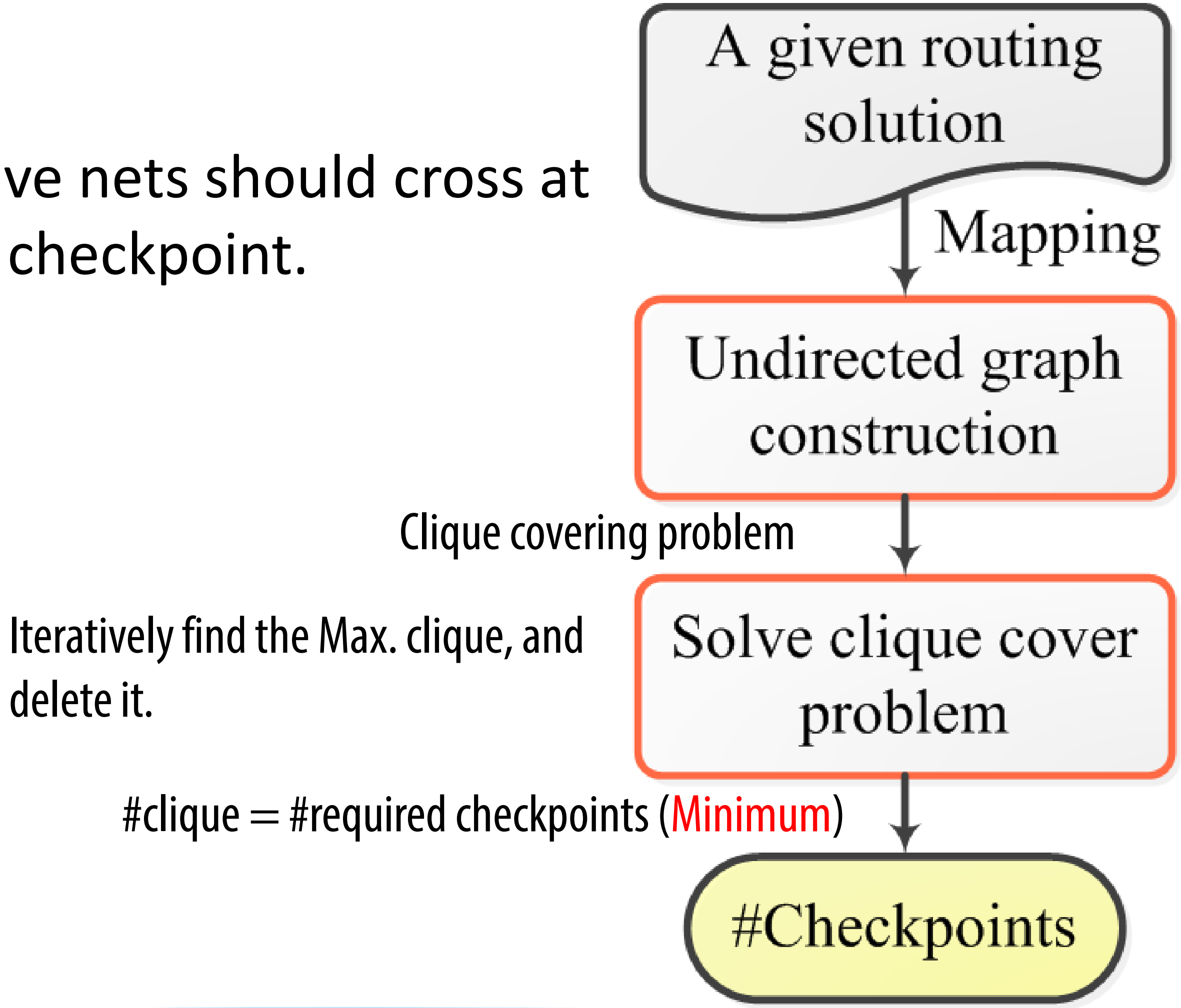
Routing solution



Usage map

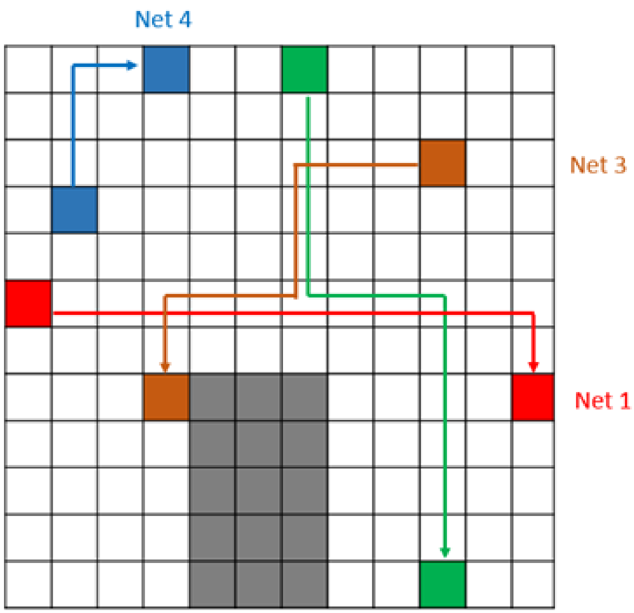
Proposed Minimization Flow

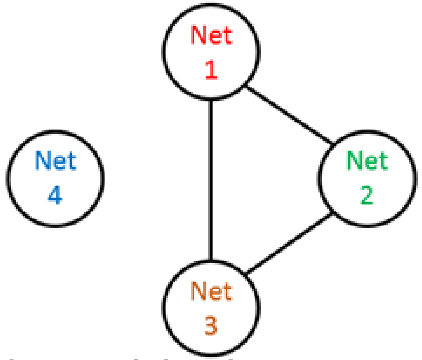
- Constraint:
  - All sensitive nets should cross at least one checkpoint.



Compatibility Graph Construction

- Construct compatibility graph, where
  - A node represents a net
  - An edge represents the paths of two nets are overlapping



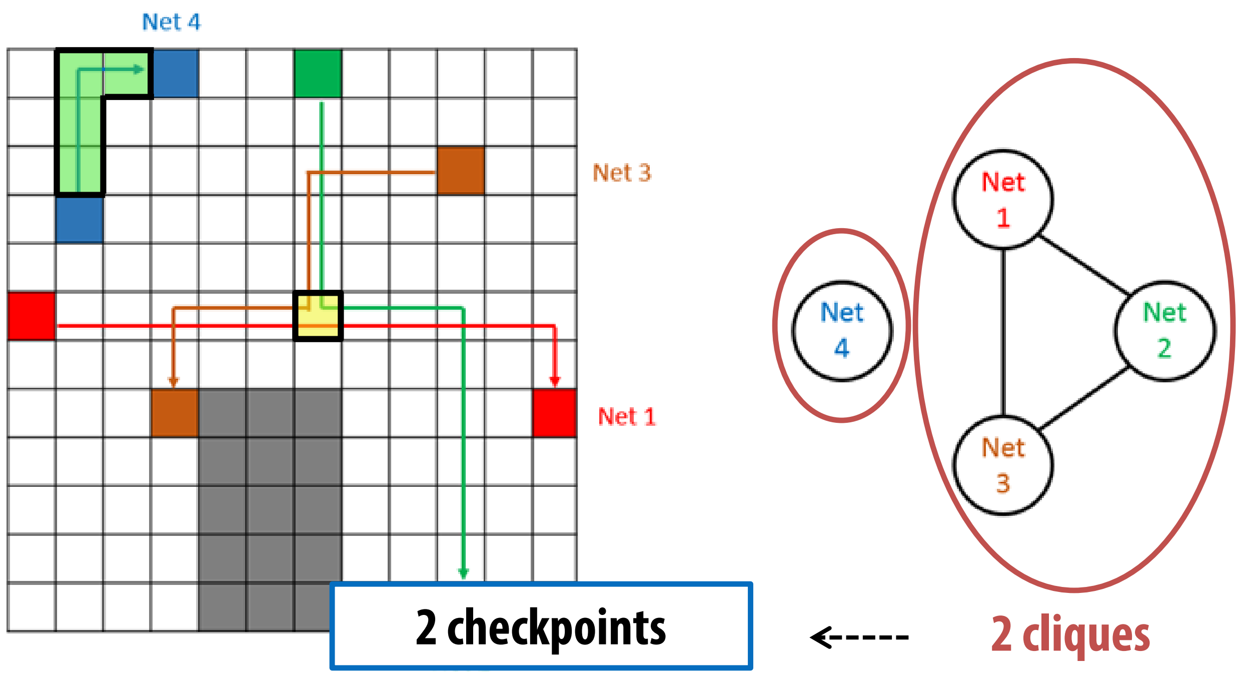


After the graph has been constructed

original minimization problem ------> **clique cover problem**

Clique Cover Problem

- Find a minimal number of non-overlapping cliques which cover all nodes in graph.



Possible locations of the first checkpoint

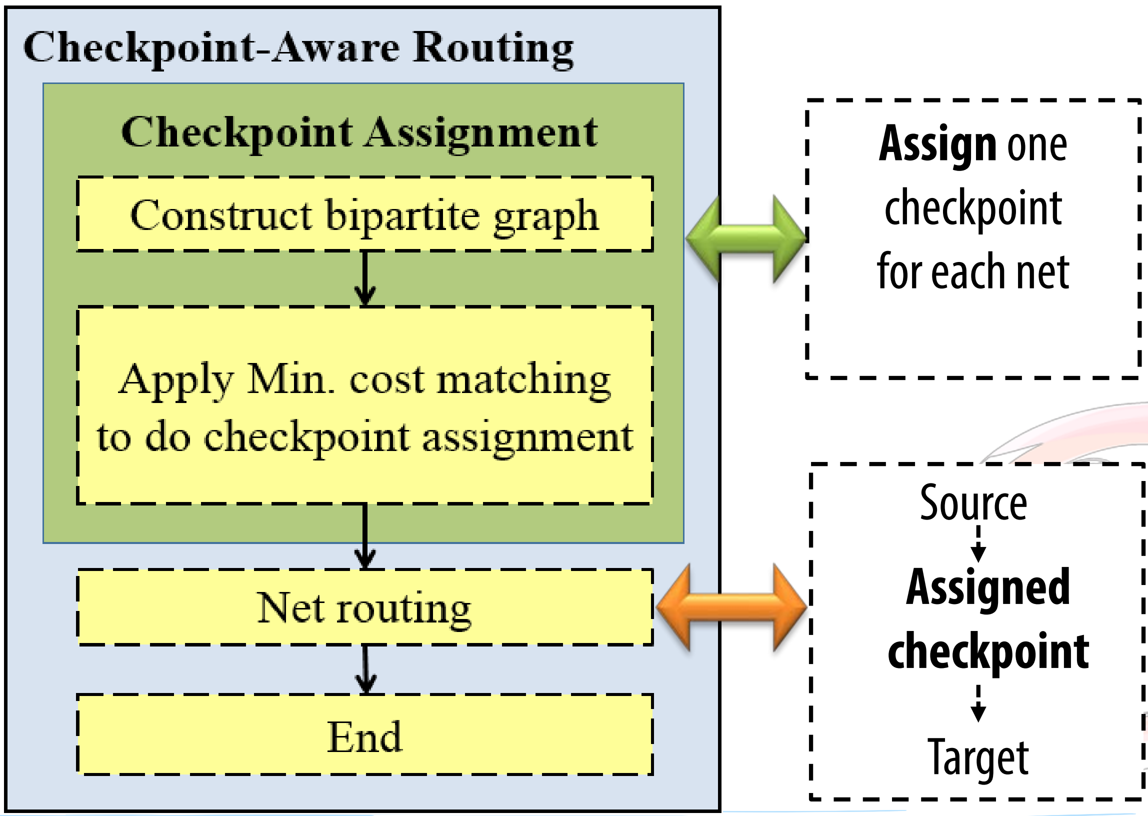
Possible locations of the second checkpoint

Outline

- Introduction
- Problem Formulations
- Proposed Algorithm
  - Checkpoint Minimization for Application-Specific DMFBs
  - Checkpoint-Aware Routing for General-Purpose DMFBs
- Experimental Result

Checkpoint-Aware Routing

- Constraint: All sensitive nets should cross at least one checkpoint.

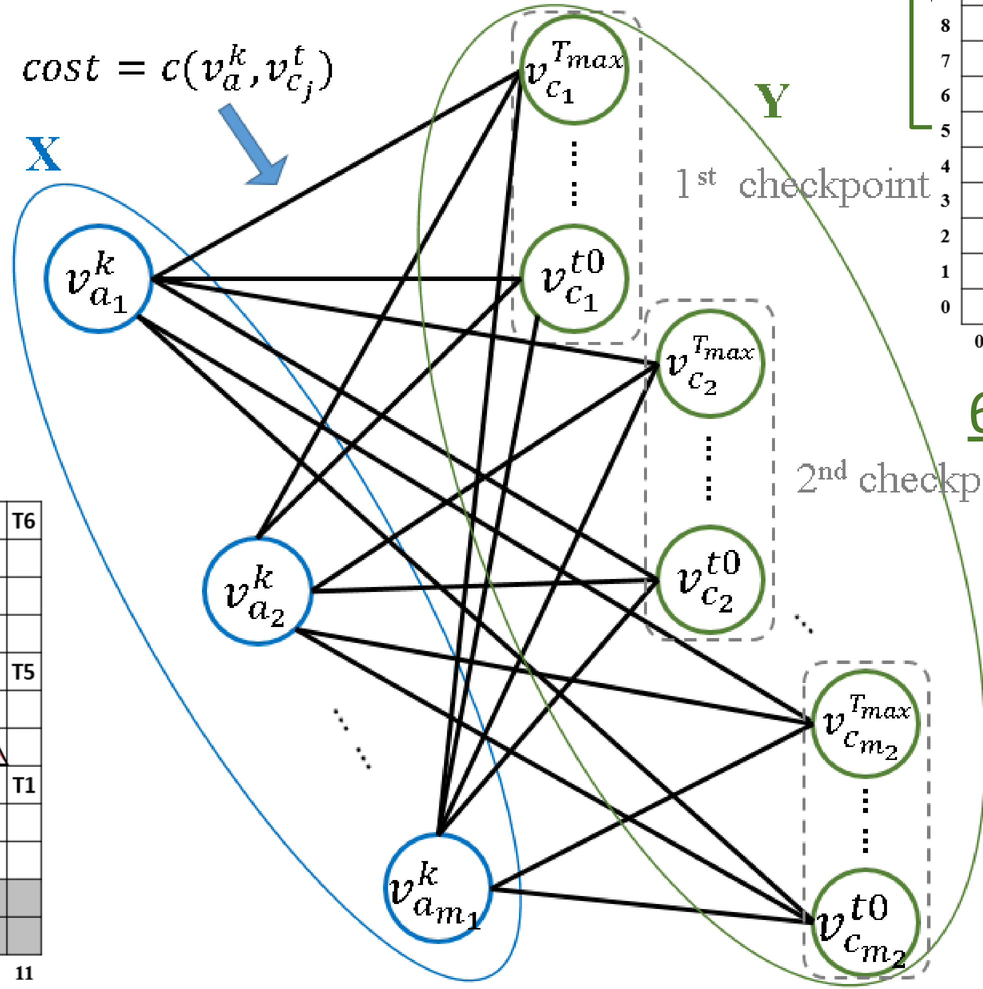


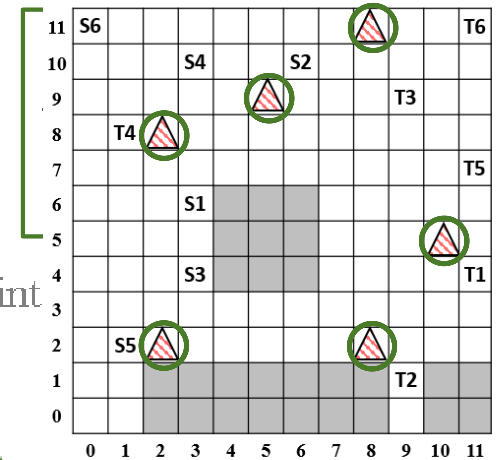
Checkpoint Assignment

- We construct the bi-partite graph to balance two goals:
  - **Reducing (eliminating) <u>checkpoint congestion</u>**,
    (there are several droplets trying to use the same checkpoint at the same time)
    - Some droplets needs to be stalled when checkpoint congestion.
  - **Minimizing the <u>maximum droplet transportation time (max wire length)</u>**.

Bipartite Graph Construction

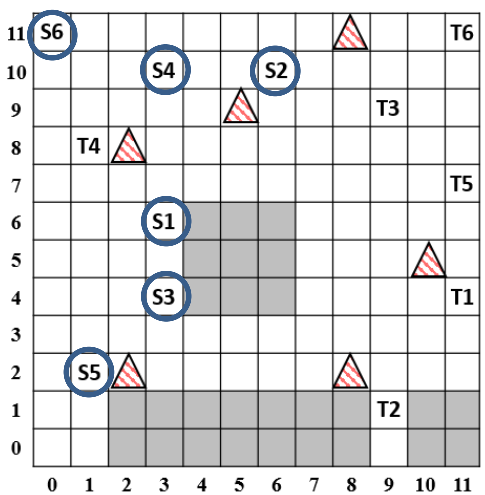
Cost = model (net length, congestion penalty)





6 checkpoints

6 droplets



Cost in Bipartite Graph

- The cost is defined as:

wire length estimation timing penalty for checkpoint congestion

$$c\left(v_a^k, v_{c_j}^t\right) = \begin{cases} \boxed{WL_{a,k}^j} + \boxed{\left(t - T_{shorstest}^{a,k,j}\right)}, if\ t \geq T_{shorstest}^{a,k,j} \\ \infty, if\ t < T_{shorstest}^{a,k,j} \end{cases}$$

where $WL_{a,k}^j$ is the sum of the distance from source to checkpoint and the distance from checkpoint to target, and $T_{shorstest}^{a,k,j}$ is the minimum distance from source to target.

- We use **Hungarian** method [1] to derive the matching in a bipartite graph.

[1]: Harold W. Kuhn, "The Hungarian Method for the assignment problem", Naval Research Logistics Quarterly, 2: 83–97, 1955. Kuhn's original publication.

Outline

- Introduction
- Problem Formulations
- Proposed Algorithm
- Experimental Result

Setting on Checkpoint Minimization

- We choose state-of-the-art routing engine to derive the routing solution [2] as input.
- Assume all droplets are highly sensitive.
  - That is, all droplets need to cross at least one checkpoint.
- **Baseline algorithm** determines the number of required checkpoints in an <u>exhaustive manner</u>.
  - That is, <u>#net is equal to #checkpoint</u>

[2]: O. Keszocze, R. Wille, T.-Y. Ho, and R. Drechsler, "Exact one-pass synthesis of digital microfluidic biochips," in Design Automation Conference, pp. 1–6, 2014.

Result on Checkpoint Minimization

- On average, the proposed minimization algorithm reduces the number of required checkpoint by 17%.

	Exhaustive[1]	Proposed
Name	#Checkpoint	#Checkpoint
in_vitro_1	28	25
in_vitro_2	35	29
protein_1	181	150
protein_2	178	146
Avg.	1.00x	0.83x

#checkpoint: Total number of required checkpoints among

The proposed algorithm can **globally select the most efficient candidates (cells)** as checkpoints. ➔ Less #checkpoint are needed

Setting on Checkpoint-Aware Routing

- Assume the locations of checkpoints are <u>evenly distributed</u>.
- Assume all droplets are highly sensitive.
  - That is, all droplets need to cross at least one checkpoint.
- **Baseline algorithm** is the <u>nearest-neighbor</u> algorithm.
  - All droplets will choose the nearest checkpoint to cross.

Result on Checkpoint-Aware Routing

- The proposed routing algorithm achieves
  - <u>100% routing completion</u> for all test cases.
  - On average, 10% less average latest-arrival-time and 15% less maximum latest-arrival-time.

	Nearest[1]		Proposed	
Name	Max. T_{la}	Avg. T_{la}	Max. T_{la}	Avg. T_{la}
in_vitro_1	21 (fail)	12	18	12
in_vitro_2	20	12	16	10
protein_1	24 (fail)	17	20	16
protein_2	20	9	18	7
Avg.	1.00x	1.00x	0.85x	0.90x

The proposed algorithm can **globally assign the checkpoints to nets.** ➔ Less checkpoint congestions are occurred.

Conclusion

- We introduced the checkpoint problems in the two design flows.
- We proposed
  - The **first checkpoint minimization algorithm** in application-specific DMFBs
  - The **first checkpoint-aware routing algorithm** in general-purpose DMFBs.
- The experimental results on a set of real-life applications demonstrated that the proposed algorithms are effective.

Thank You