University of Basel

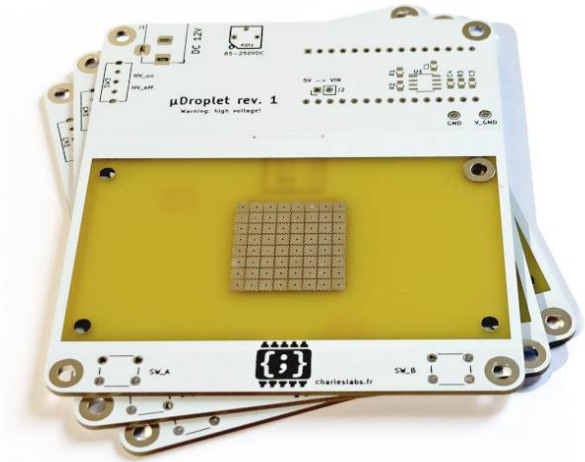
Temporal Planning for Droplet Routing on Microfluidic Biochips - Master Thesis Presentation

Fabian Burch <f.burch@unibas.ch>

DMI, University of Basel

12.12.2022

DMFB: Digital Microfluidic Biochip



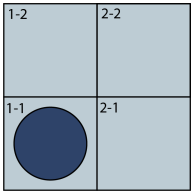
DMFB problems

- Scheduling
- Module Placement
- Droplet Routing
- Pin Assignment
- Escape Routing

Classical Planning & Temporal Planning

Classical Planning

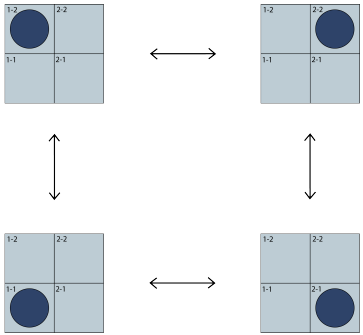
- sets of state variables describe states



- s(*droplet-at* 1-1) = T
- s(*droplet-at* 1-2) = F
- s(*droplet-at* 2-1) = F
- s(*droplet-at* 2-2) = F

Classical Planning

- sets of state variables describe states
- actions allow transitions from one state to another



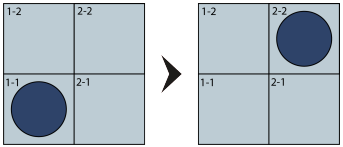
Classical Planning

- sets of state variables describe states
- actions allow transitions from one state to another
- actions have preconditions, effects and a cost

```
(:action move_1-1_1-2
    :precondition
        (droplet-at 1-1)
    :effect (and
        (not (droplet-at 1-1))
        (droplet-at 1-2)
    )
)
```

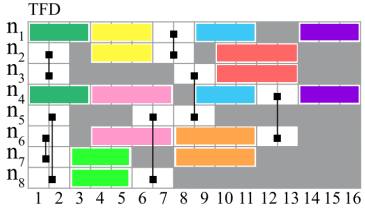
Classical Planning

- sets of state variables describe states
- actions allow transitions from one state to another
- actions have preconditions, effects and a cost
- a planner tries to find a sequence of actions from an initial state to a goal state
- an optimal plan has a minimal sum of costs



Temporal Planning

- actions have a duration instead of being executed instantly
- actions can be executed in parallel
- an optimal plan has a minimal makespan



makespan = 16

Classical vs. Temporal Planning

Classical Planning

- actions are sequential
- effects happen instantly
- cost is minimized

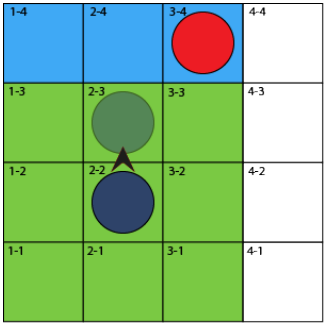
Temporal Planning

- actions can be parallel
- actions have a duration
- makespan is minimized

Droplet Routing Setup

PDDL Model for Droplet Routing

```
(:action move_2-2_2-3
    :parameters (?d - ?droplet)
    :precondition (and
        (droplet-at ?droplet x2 y2)
        (not (occupied x1 y4))
        (not (occupied x2 y4))
        (not (occupied x3 y4))
    )
    :effect (and
        (not (droplet-at ?droplet x2 y2))
        (droplet-at ?droplet x2 y3)
        (not (occupied x2 y2))
        (occupied x2 y3)
    )
)
```



Testset

side length	9	9	9	9	9	10	11	12	13	14	15	15	15	15
droplets	5	6	7	8	9	7	7	7	7	7	7	7	7	7
blockages	3	3	3	3	3	3	3	3	3	3	3	4	5	6

Table of the test set parameters.

- 100 randomly generated instances for each column

Configuration

Classical Planning

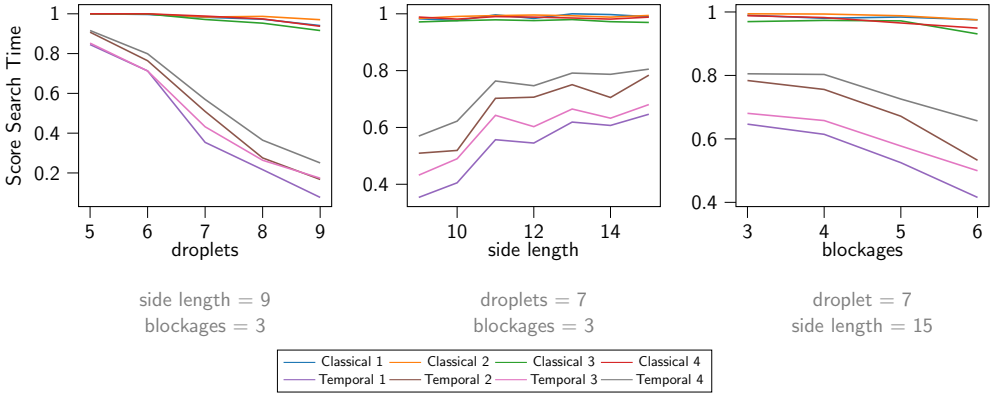
- Fast Downward
- first iteration of Lama
- time limit: 5 minutes

Temporal Planning

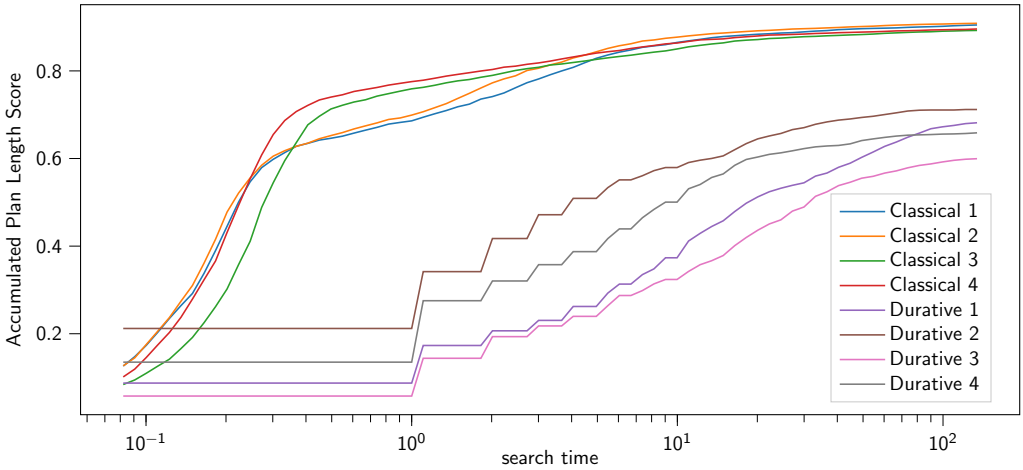
- Temporal Fast Downward (TFD)
- anytime search disabled
- time limit: 5 minutes

Droplet Routing Results

Droplet Routing Results 1: Search Time for Varying Parameters

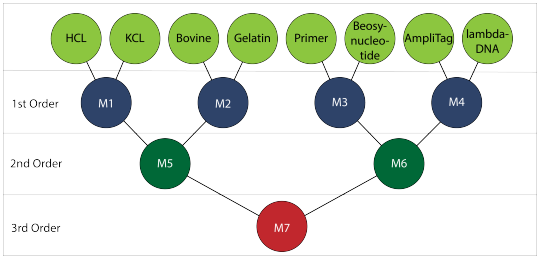


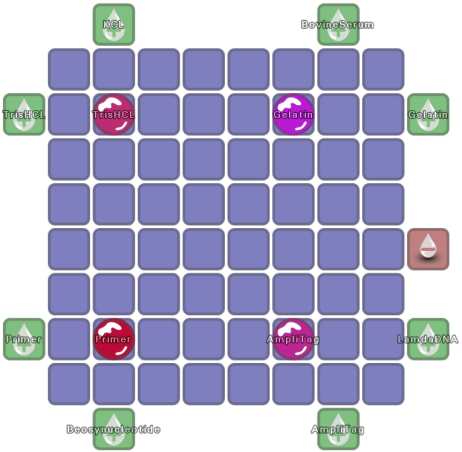
Droplet Routing Results 2: Plan Quality



Extensions

Extensions: The Polymerase Chain Reaction (PCR)





Extensions: Change of Paradigm

With the possibility to add and remove droplets, we can not use the droplet parameter as a unique identifier anymore!

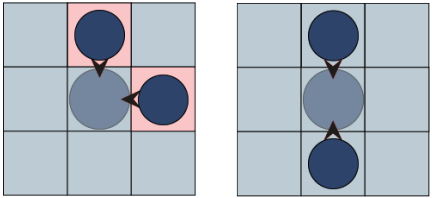
unique identifier: droplet ?d $\longrightarrow$ position ?x ?y

(droplet-at ?d ?x ?y) $\longrightarrow$ (reagent-type ?r ?x ?y)

Any property is attached to the position, for example: (small ?x ?y)

Extension 1: Spawning, Disposing, Merging and Splitting

Spawns and Waste Outlet

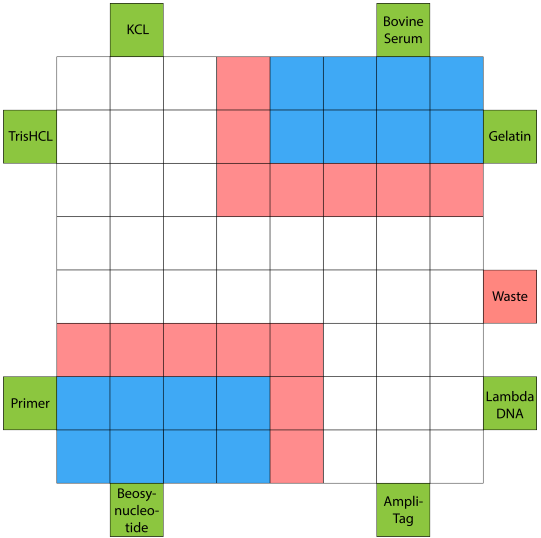


Impossible and Possible Arrangement for Merging

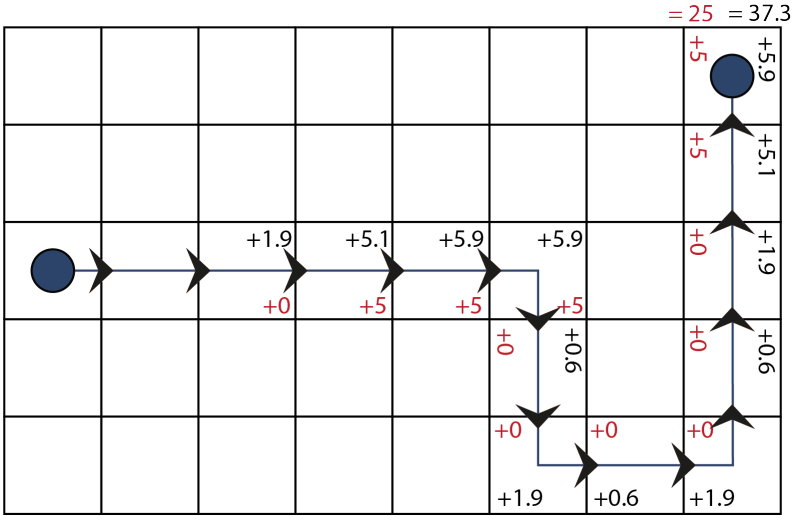
Extension 1: Code for (Horizontal) Merging

```
(:action merge_x
    :parameters(?r1 ?r2 ?r3 - reagent ?x1 ?x2 ?xt - xcoord ?yt - ycoord)
    :precondition (and
        ; check that the correct reagents are present
        (reagent-type ?r1 ?x1 ?yt)
        (reagent-type ?r2 ?x2 ?yt)
        (small ?x1 ?yt)
        (small ?x2 ?yt)
        ; check that three cells are next to each other
        (ISEAST ?x1 ?xt)
        (ISWEST ?x2 ?xt)
        ; ckeck that the ?r1 and ?r2 can be mixed into ?r3
        (MIX ?r1 ?r2 ?r3)
    )
    :effect (and
        ; delete all atoms of the droplets that are merged
        (not (reagent-type ?r1 ?x1 ?yt))
        (not (reagent-type ?r2 ?x2 ?yt))
        (not (small ?x1 ?yt))
        (not (small ?x2 ?yt))
        (not (occupied ?x1 ?yt))
        (not (occupied ?x2 ?yt))
        ; add atoms for the newly merged, big droplet
        (reagent-type ?r3 ?xt ?yt)
        (occupied ?xt ?yt)
    )
)
```

Extension 2: Mixing Modules



Extension 3: Mixing Without Modules

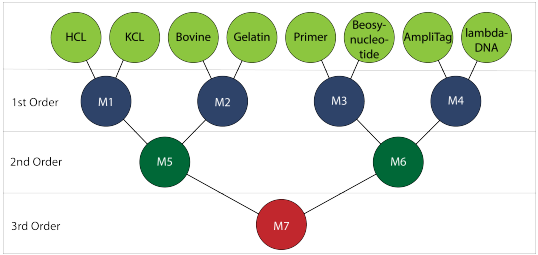


Extensions Results

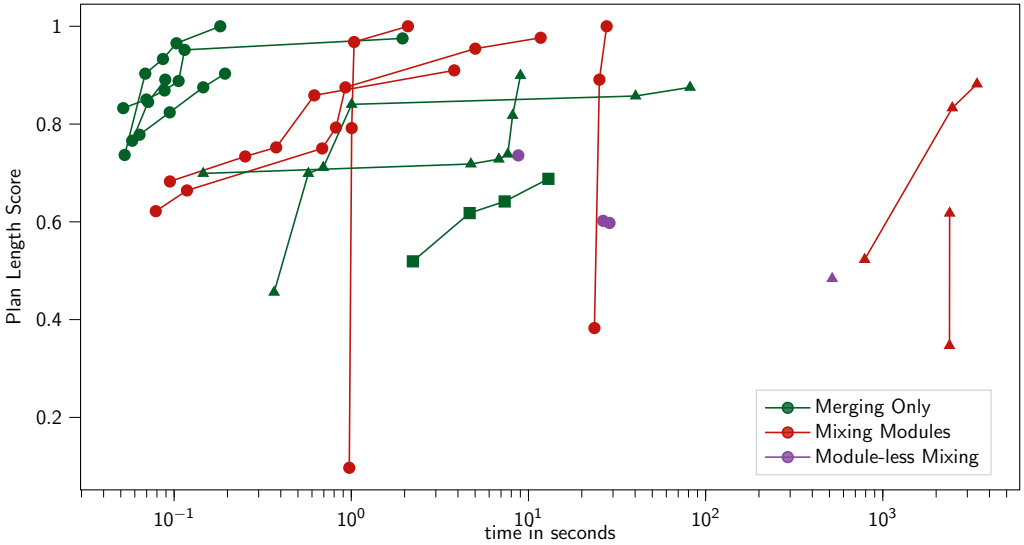
Extensions: Configuration

Classical Planning

- Fast Downward
- Lama
- time limit: 30 minutes



Extensions Results



Conclusions

Classical Planning works very well for droplet routing.

Planning is flexible so we can add functionality.

When scaling the problem size to real life experiments, the model struggles to find plans.

Questions?

f.burch@unibas.ch