KU KONKUK UNIVERSITY

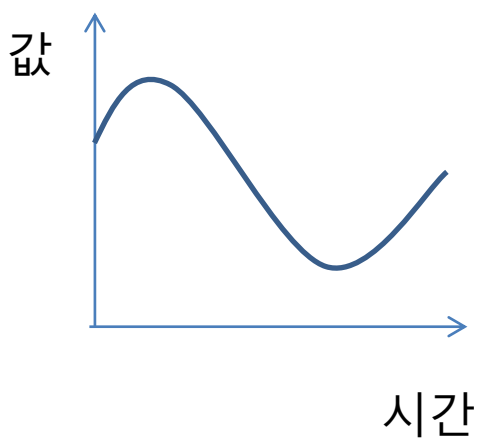
Case Study – HyTech를 이용하여 자동 차간 거리 조절

Jaeyeon Jo

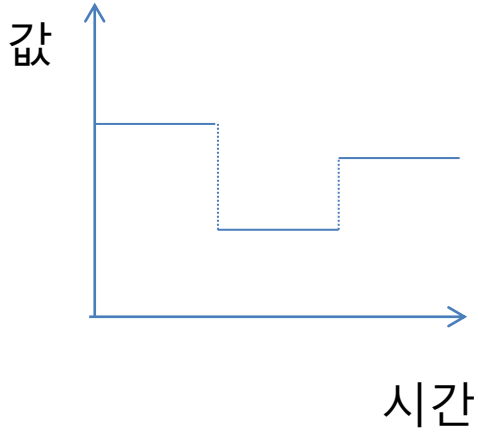
Dependable Software Laboratory

Background

- Hybrid System
  - Continuous element – Physical, Electric, Analog
  - Discrete Element – Digital electronic, Computing, Software



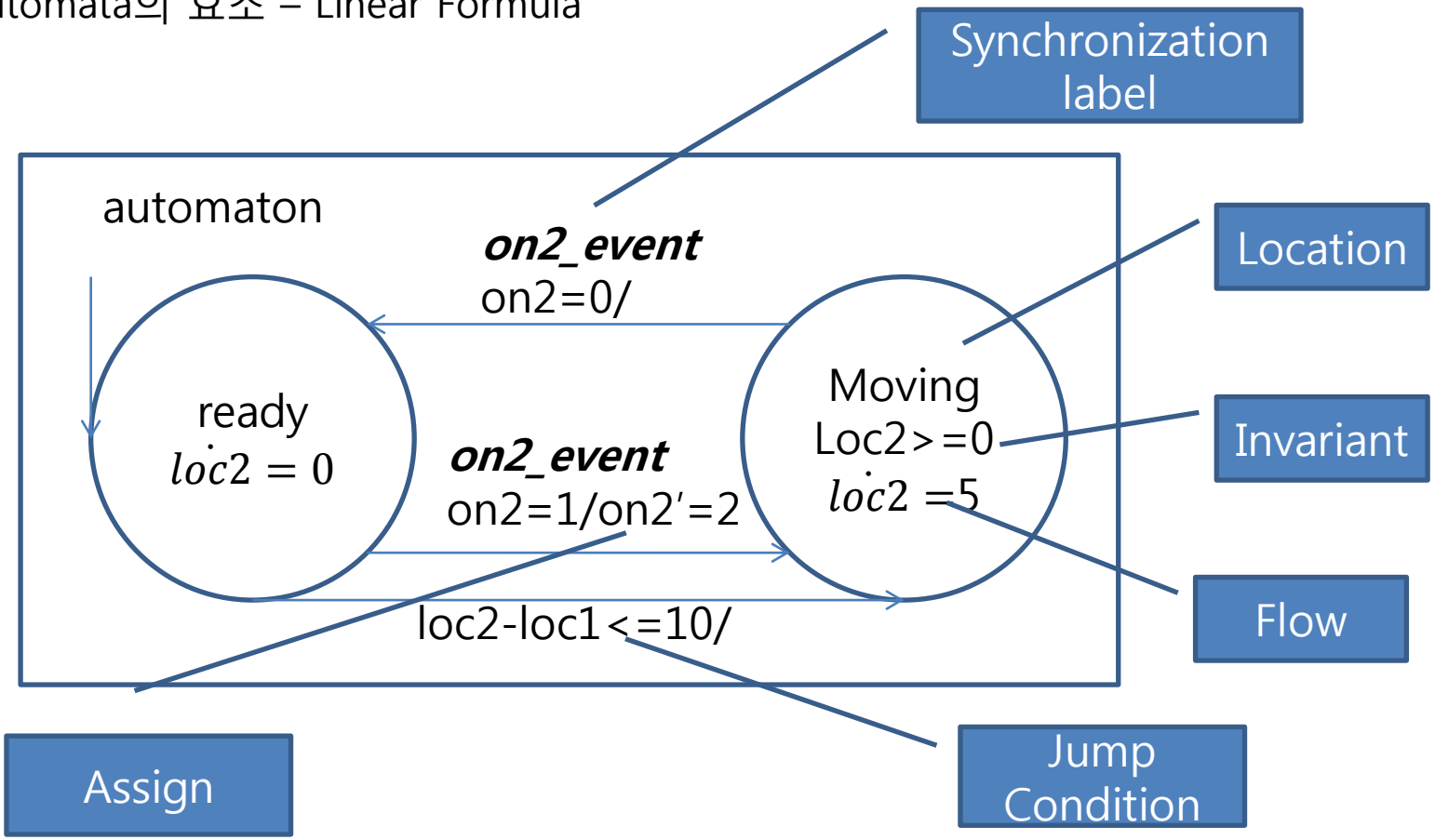
Analog Variable



Discrete Variable

Linear Hybrid Automata

- Linear Hybrid Automata
  - Hybrid System을 Automata의 형식으로 표현
  - Automata의 요소 – Linear Formula

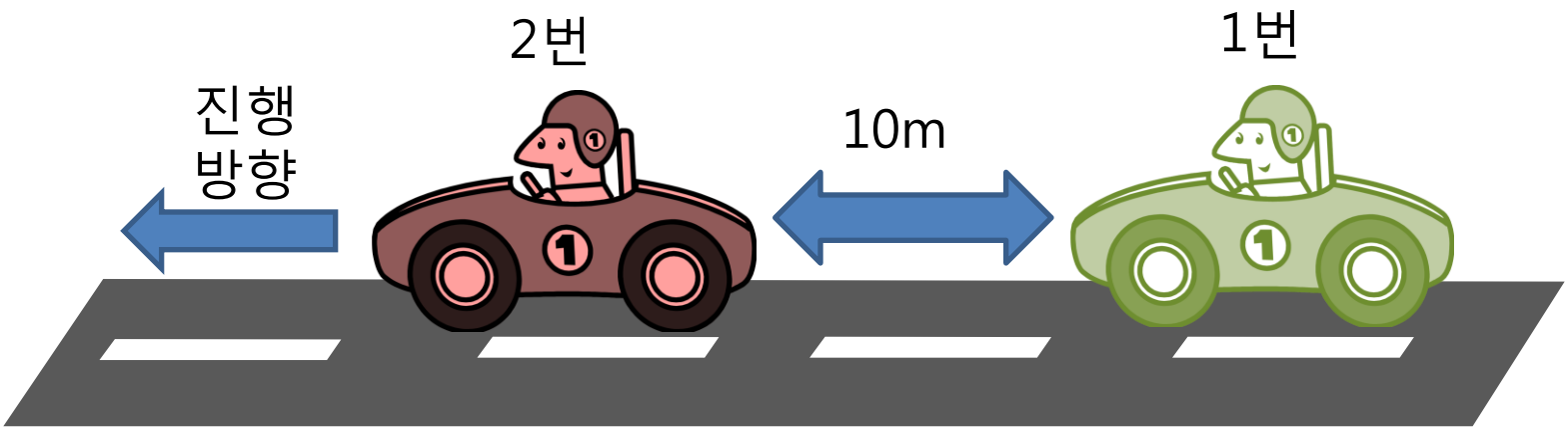


HyTech

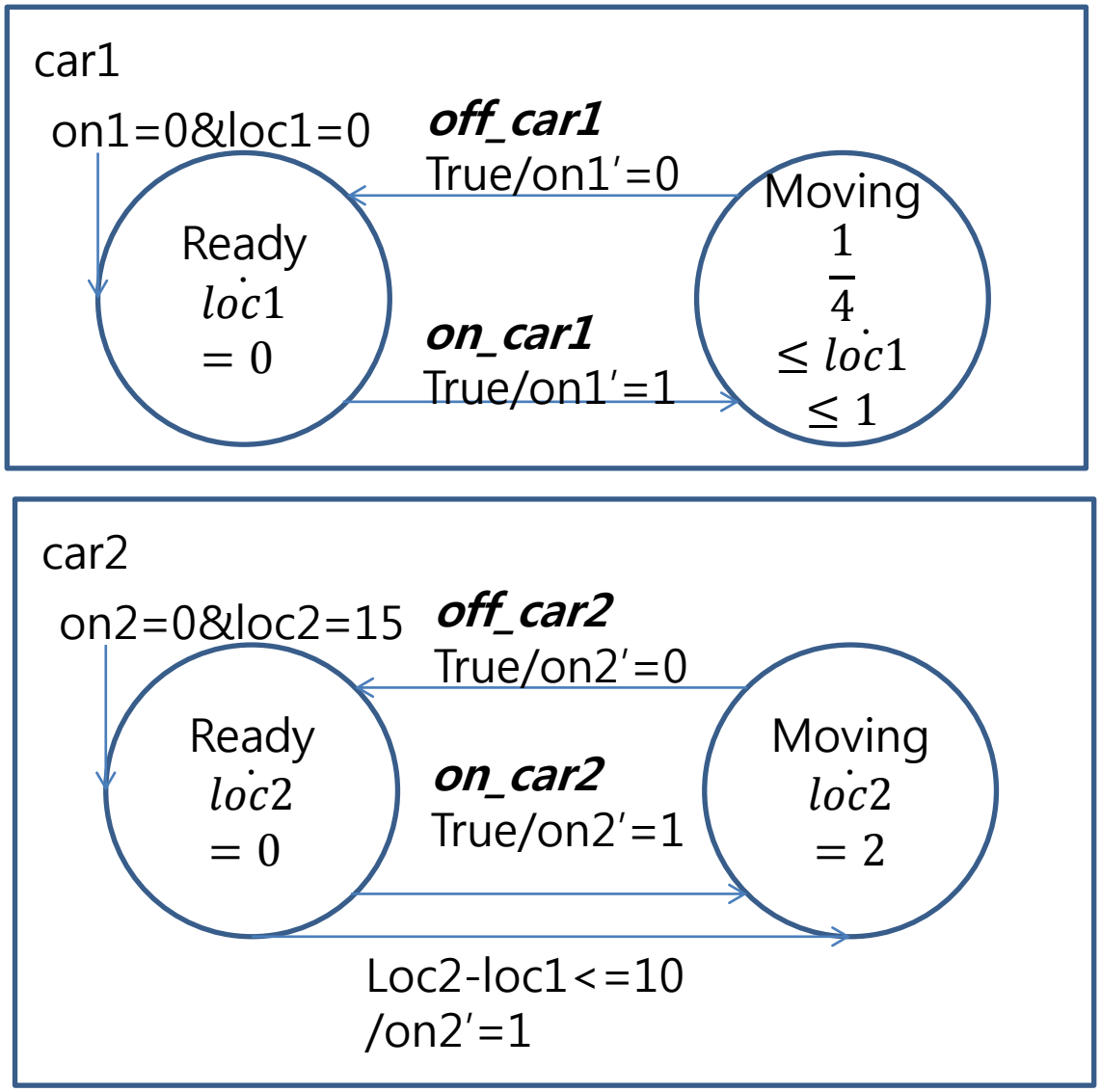
- Hybrid System을 검증, 분석하는 도구
- 입력 언어 – Linear Hybrid Automata
  - Variables, Locations, Initial Condition, Invariant Conditions, Transitions, Rate Conditions
- Parallel composition
  - Synchronizing Hybrid Automata
- Reachability and safety verification
  - Forward/Backward reachability
- Parametric analysis

Model : Car control model

- Example – Car Controller 예제
  - 안전성 위반 조건 : car1과 car2의 거리가 10 이하
  - 안전성 위반 조건 검사 : 초기 상태로부터 해당 상태까지 갈수 있는가?
    - 해당 상태까지 갈 수 있음 : 안전성 위반
    - 해당 상태까지 갈 수 없음 : 안전성 유지
  - 초기 상태를 설정하지 않는다면 해당 상태까지 제약 없이 갈 수 있음
    - 초기상태를 설정할 필요가 있음



Automata


car1
on1=0&loc1=0
off_car1
True/on1′=0
Ready
$\dot{loc1}$
$= 0$
on_car1
True/on1′=1
Moving
$\frac{1}{4} \leq \dot{loc1} \leq 1$
car2
on2=0&loc2=15
off_car2
True/on2′=0
Ready
$\dot{loc2}$
$= 0$
on_car2
True/on2′=1
Moving
$\dot{loc2}$
$= 2$
Loc2-loc1<=10
/on2′=1

HyTech Code

```
var
        loc1:analog;
        loc2:analog;

        on1:discrete;
        on2:discrete;
```

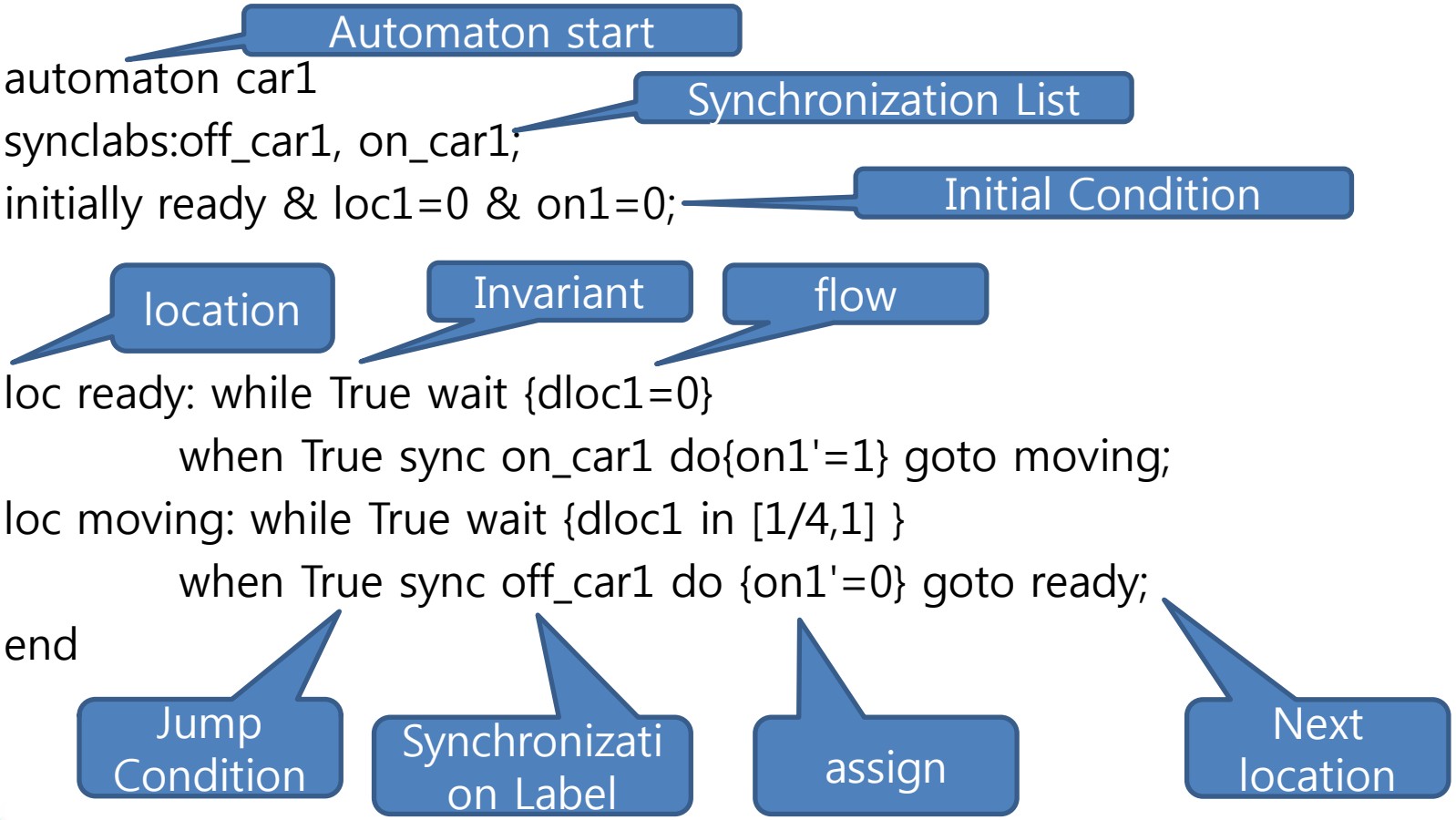
HyTech에서 검증할 Automata가 사용할 Variable 선언

Rate에 따른 variable 분류

analog : flow에서 정할 수 있음

discrete : rate가 0이고 transition에서만 값 변경 가능

HyTech Code


Automaton start
automaton car1
Synchronization List
synclabs:off_car1, on_car1;
Initial Condition
initially ready & loc1=0 & on1=0;
location
Invariant
flow
loc ready: while True wait {dloc1=0}
when True sync on_car1 do{on1'=1} goto moving;
loc moving: while True wait {dloc1 in [1/4,1] }
when True sync off_car1 do {on1'=0} goto ready;
end
Jump Condition
Synchronization Label
assign
Next location

Properties

```
var init_reg, final_reg, reached_reg : region;

init_reg := loc[car1]=ready & loc[car2]=ready & loc1=0 & loc2=15 & on1=0
& on2=0 & t=0;
final_reg := loc2-loc1<10;

reached_reg := reach forward from init_reg endreach;

print trace to final_reg using reached_reg;
```

Region definition

Initial region

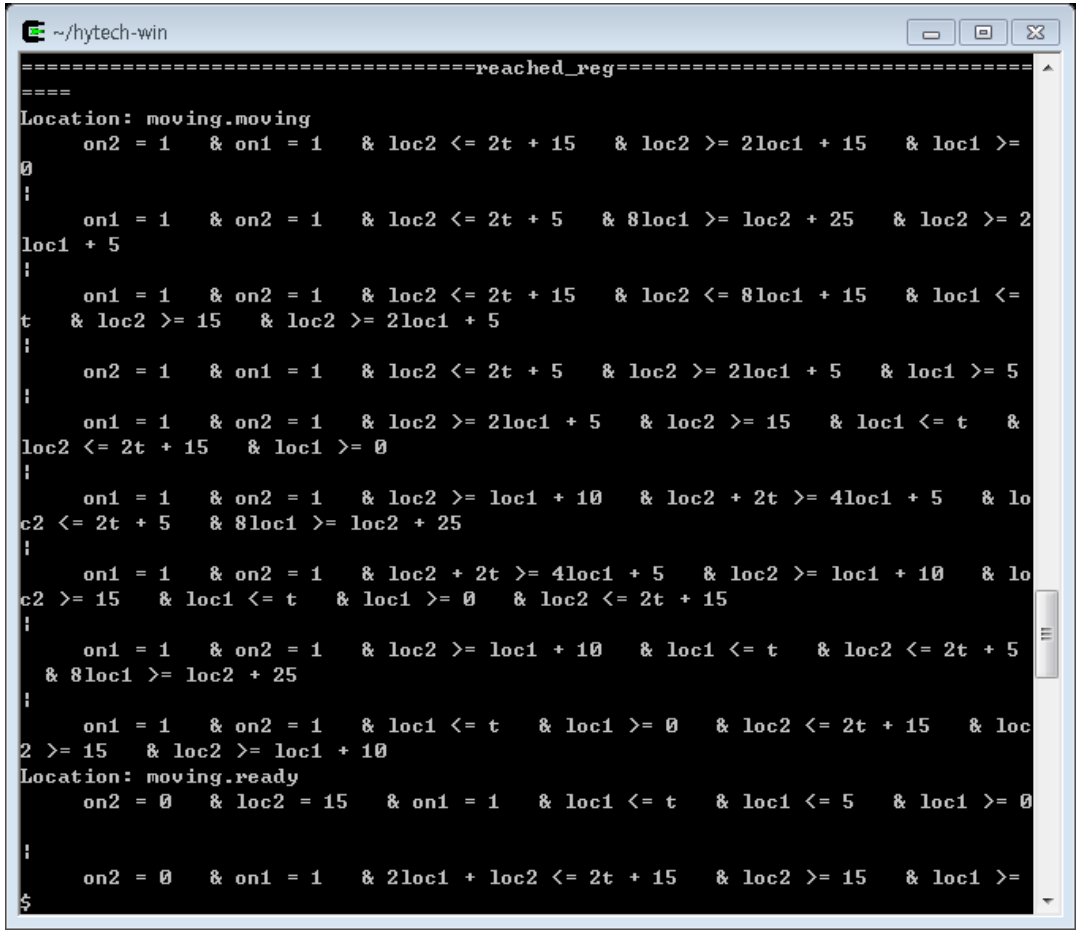
Verification property

All reachable region from init_reg

Verification Code

Verification

- Reached State List
  - Initial condition으로부터 도달할 수 있는 모든 상태를 나타냄
  - Moving, Moving상태에서 가능한 모든 변수의 값들을 나타냄



Verification

- Final_reg
  - 검증할 속성을 나타냄
  - 이 상태에 도달할 수 있다면, 검증속성을 만족하지 못함

- No path to….
  - 초기 상태로부터 Final_reg까지 도달할 수 없을 나타냄
  - 만약 Final_reg까지 도달 가능하다면 도달하는 시나리오 중 하나를 보여줌 (Counter Example)