

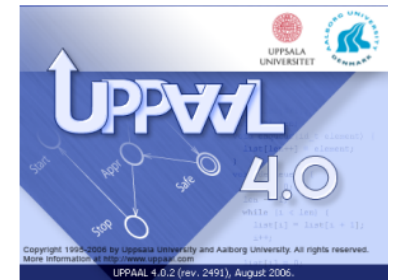
ENERGY AWARE BUILDINGS

Modelling, Analysis & Optimization using
Model Checking & Machine Learning

Kim G Larsen

Jiri Srba, Marco Muniz, Peter Gjøøl, Jakob Taankvist, Mads Kronborg Agesen,
Søren Enevoldsen, Thibaut Le Guilly, Anders Mariegaard,
Marius Mikucionis, Petur Olsen, Arne Skou

Aalborg Universitet





CAV Award 13

UPPAAL Tool Suite

The image displays several windows from the UPPAAL tool suite. The top-left window shows a state machine diagram with states like 'Safe', 'Cross', 'Appr', and 'Start', and transitions labeled with actions and guards. The top-right window shows a grid of state machines for different train instances (Train(0) to Train(5)) and a 'Gate' component. The bottom-left window shows simulation results with a 'Simulations (1)' plot and a 'Probability Density Distribution' histogram. The bottom-right window shows a 'Verifier' interface with an 'Overview' section containing a query: `A[] forall (i : id_t) forall (j : id_t) Train(i).Cross && Train(j).Cross imp...` and a 'Query' section with the query: `A[] forall (i : id_t) forall (j : id_t) Train(i).Cross && Train(j).Cross imply i == j`. A comment below the query states: 'There is never more than one train crossing the bridge (at any time instance).'

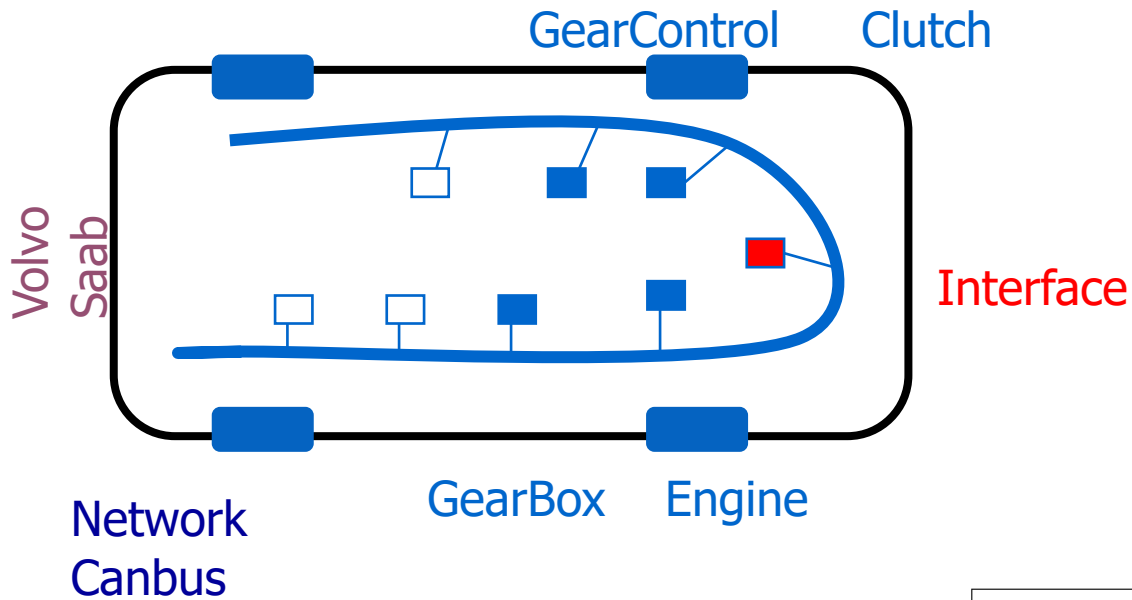
Verification	CLASSIC	1995
Optimization	CORA	2001
Testing	TRON	2004
Synthesis	TIGA	2005
Component	ECDAR	2010
Performance Analysis	SMC	2011
Machine Learning	STRATEGO	2014

Gear Controller

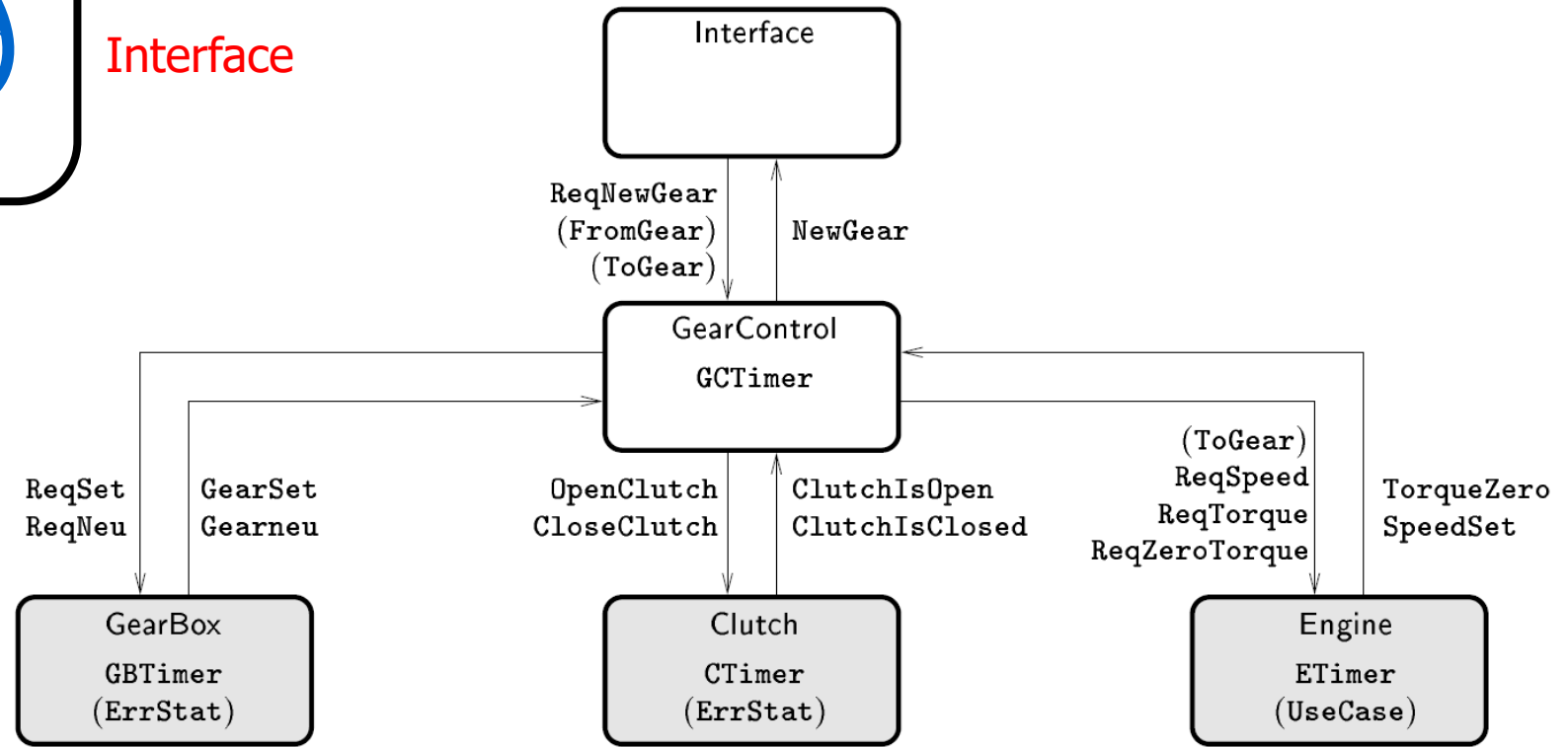
with MECEL AB



AALBORG UNIVERSITY
DENMARK



Flowgraph



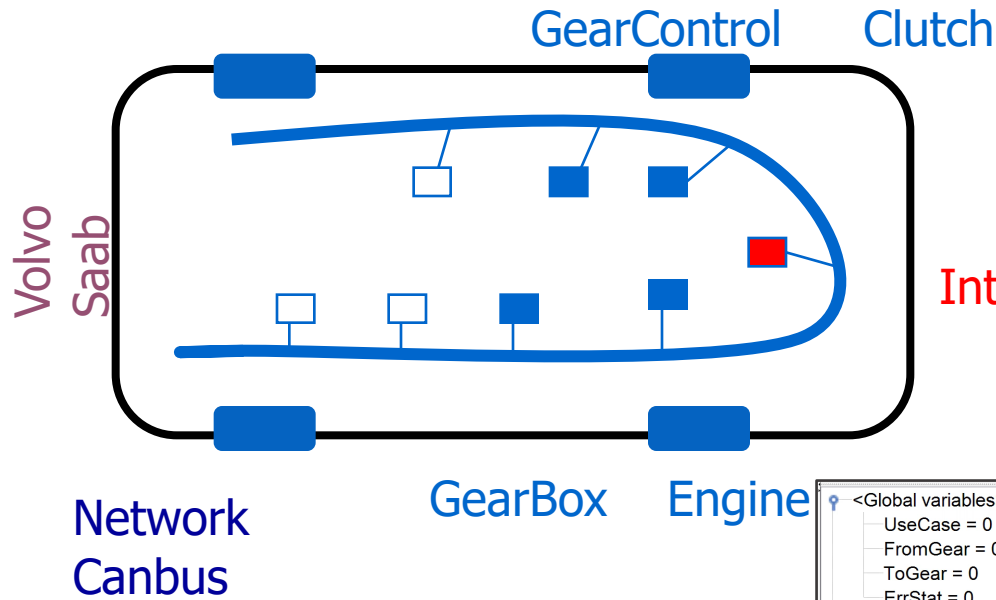
Magnus Lindahl
Paul Pettersson
Wang Yi
2001

Gear Controller

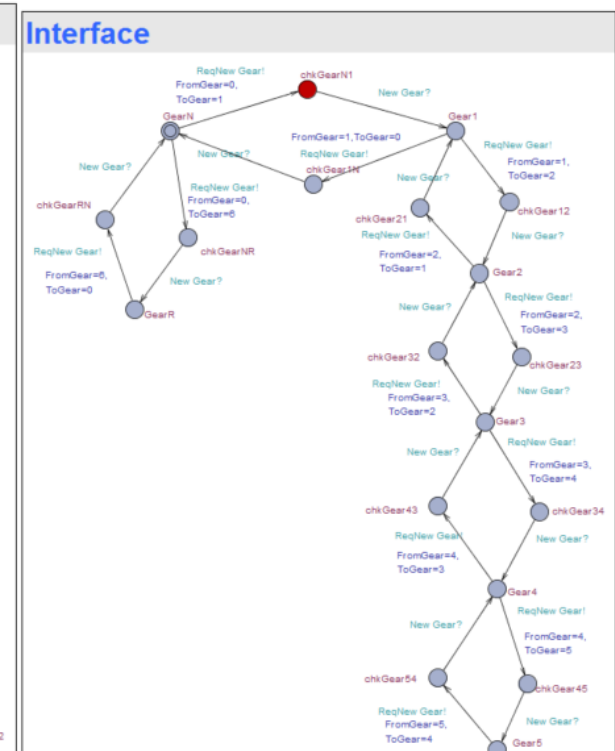
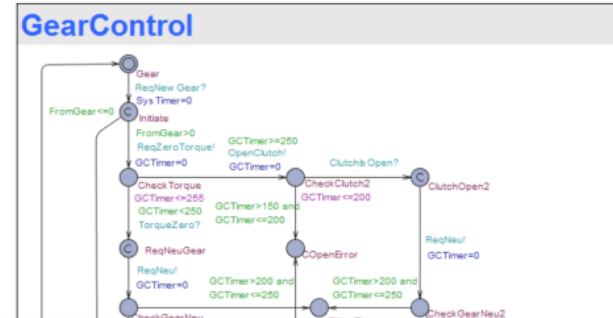
with MECEL AB



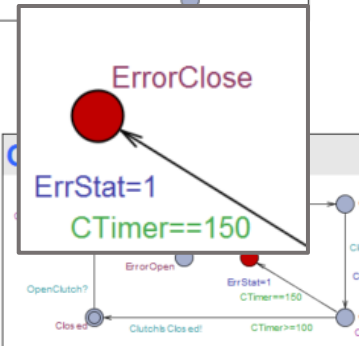
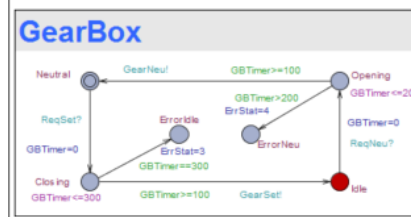
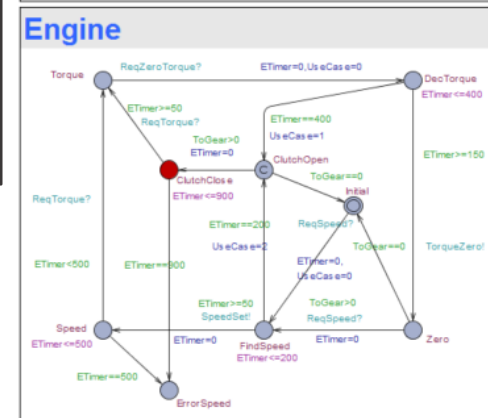
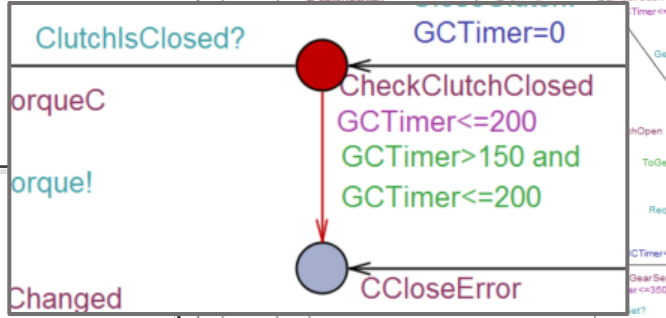
AALBORG UNIVERSITY
DENMARK



Interface



- <Global variables>
 - UseCase = 0
 - FromGear = 0
 - ToGear = 0
 - ErrStat = 0
- <Constraints>
 - CTimer ≥ 0
 - ETimer ≥ 0
 - GBTimer ≥ 0
 - GCTimer ≥ 0
 - SysTimer ≥ 0
 - GearControl.GCTimer ≥ 0
 - CTimer = ETimer
 - ETimer = GBTimer
 - GBTimer = GCTimer
 - SysTimer = GearControl.GCTimer
 - GearControl.GCTimer = CTimer



Timed Automata Models

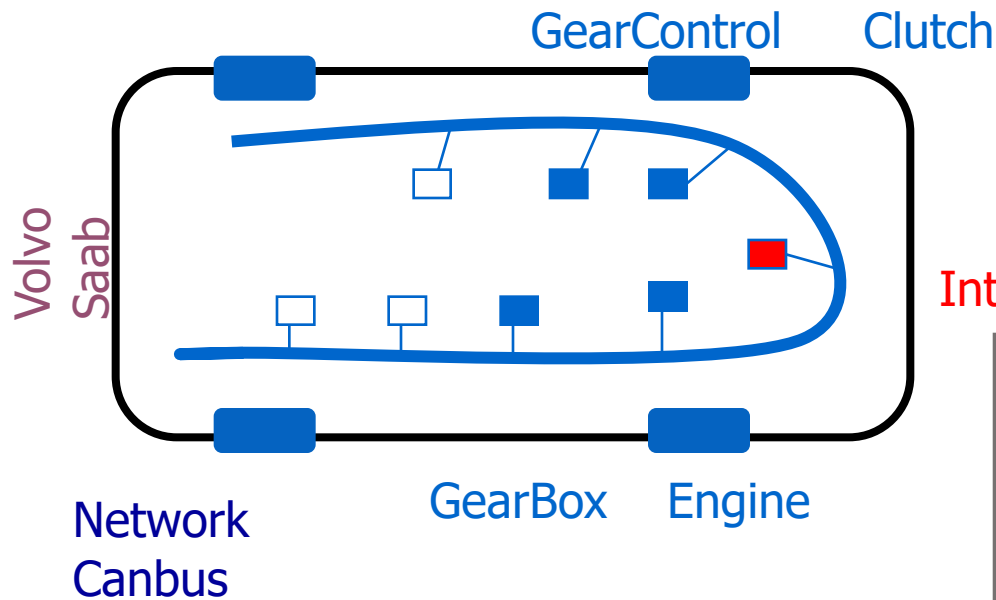
Magnus Lindahl
Paul Pettersson
Wang Yi
2001

Gear Controller

with MECEL AB



AALBORG UNIVERSITY
DENMARK



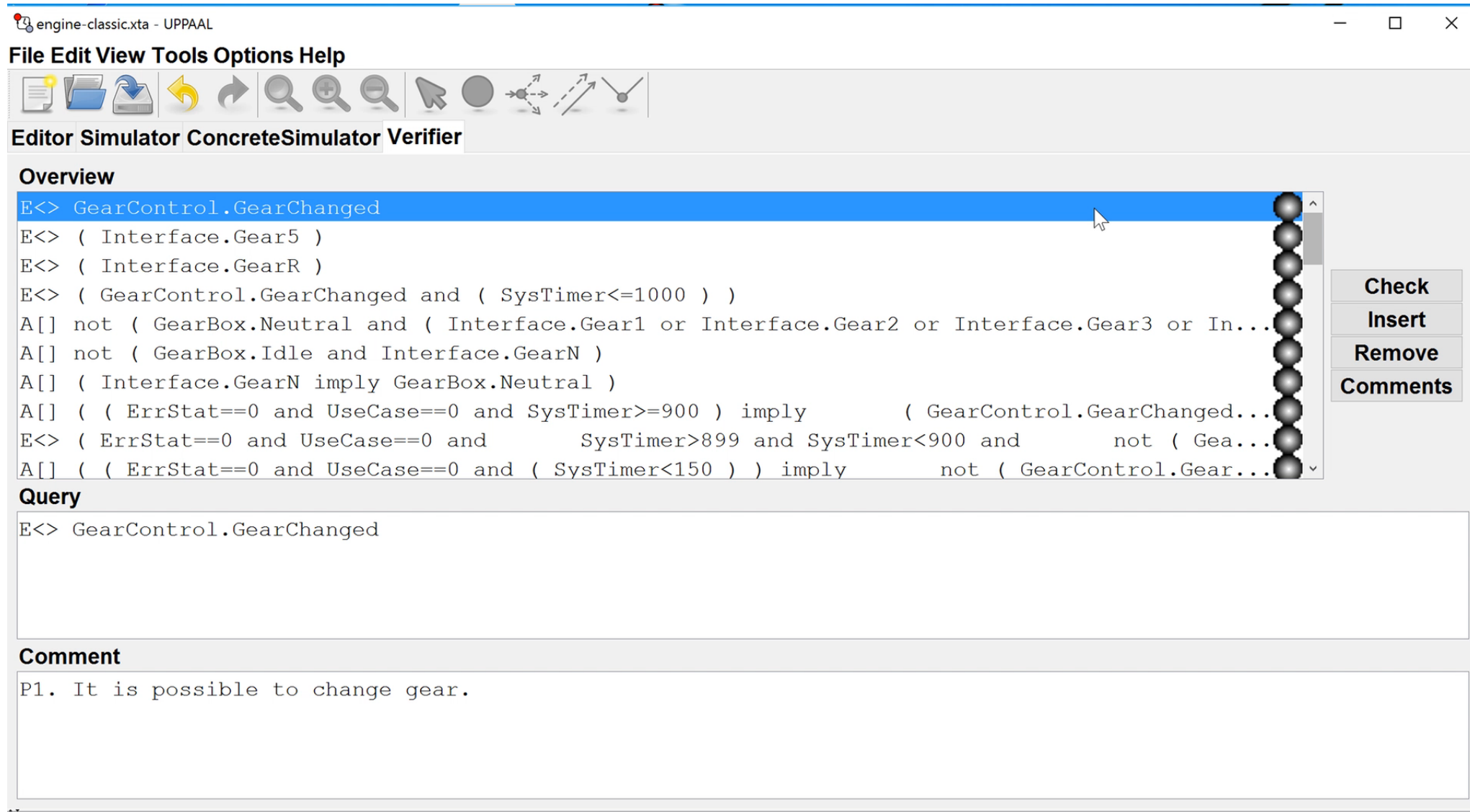
Requirements

Interface

```
GearControl@Initiate  $\rightsquigarrow_{\leq 1500}$  ( ( ErrStat = 0 )  $\Rightarrow$  GearControl@GearChanged )
GearControl@Initiate  $\rightsquigarrow_{\leq 1000}$ 
    ( ( ErrStat = 0  $\wedge$  UseCase = 0 )  $\Rightarrow$  GearControl@GearChanged )
Clutch@ErrorClose  $\rightsquigarrow_{\leq 200}$  GearControl@CCloseError
Clutch@ErrorOpen  $\rightsquigarrow_{\leq 200}$  GearControl@COpenError
GearBox@ErrorIdle  $\rightsquigarrow_{\leq 350}$  GearControl@GSetError
GearBox@ErrorNeu  $\rightsquigarrow_{\leq 200}$  GearControl@GNeuError
Inv ( GearControl@CCloseError  $\Rightarrow$  Clutch@ErrorClose )
Inv ( GearControl@COpenError  $\Rightarrow$  Clutch@ErrorOpen )
Inv ( GearControl@GSetError  $\Rightarrow$  GearBox@ErrorIdle )
Inv ( GearControl@GNeuError  $\Rightarrow$  GearBox@ErrorNeu )
Inv ( Engine@ErrorSpeed  $\Rightarrow$  ErrStat  $\neq$  0 )
Inv ( Engine@Torque  $\Rightarrow$  Clutch@Closed )
```

Magnus Lindahl
Paul Pettersson
Wang Yi
2001

UPPAAL Model Checking - Demo



engine-classic.xta - UPPAAL

File Edit View Tools Options Help

Editor Simulator ConcreteSimulator Verifier

Overview

```
E<> GearControl.GearChanged
E<> ( Interface.Gear5 )
E<> ( Interface.GearR )
E<> ( GearControl.GearChanged and ( SysTimer<=1000 ) )
A[] not ( GearBox.Neutral and ( Interface.Gear1 or Interface.Gear2 or Interface.Gear3 or In...
A[] not ( GearBox.Idle and Interface.GearN )
A[] ( Interface.GearN imply GearBox.Neutral )
A[] ( ( ErrStat==0 and UseCase==0 and SysTimer>=900 ) imply ( GearControl.GearChanged...
E<> ( ErrStat==0 and UseCase==0 and SysTimer>899 and SysTimer<900 and not ( Gea...
A[] ( ( ErrStat==0 and UseCase==0 and ( SysTimer<150 ) ) imply not ( GearControl.Gear...
```

Check
Insert
Remove
Comments

Query

```
E<> GearControl.GearChanged
```

Comment

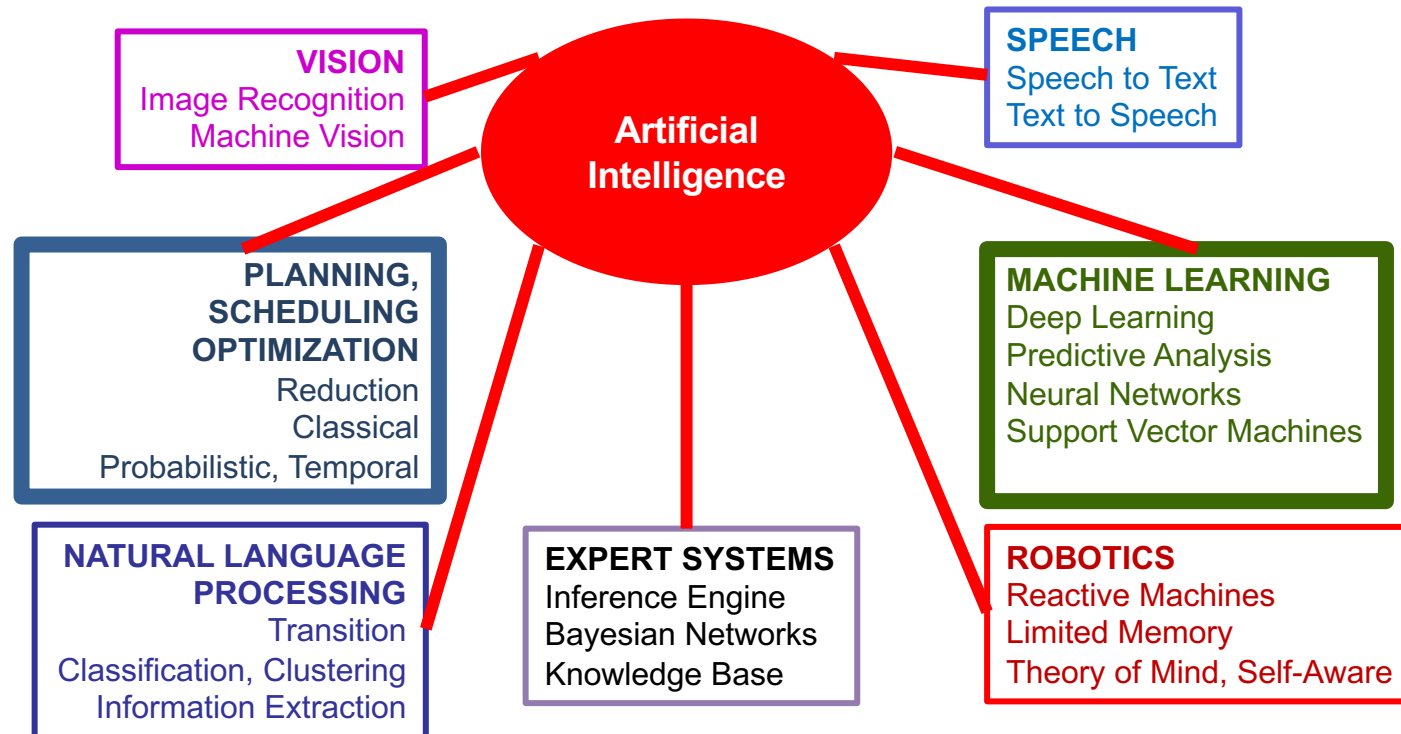
```
P1. It is possible to change gear.
```

SMART HOUSES

- Automatic control of a number of functions in a house.
- Formulated as a **game** between **environment** and the **controller**.
- Automatic synthesis of improved, optimal and personalized control strategy.
- From **abstract game strategies** to **concrete code** running on **real hardware**.

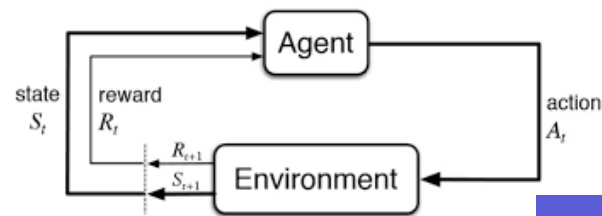
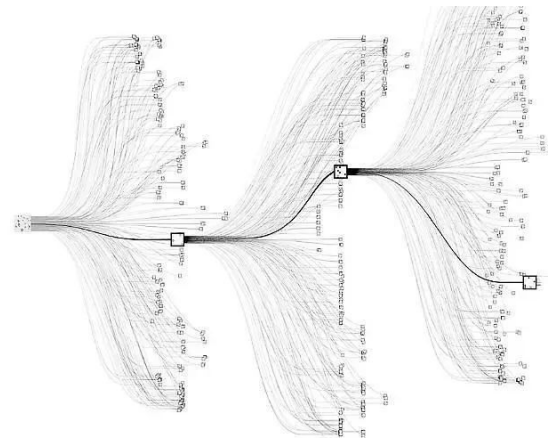
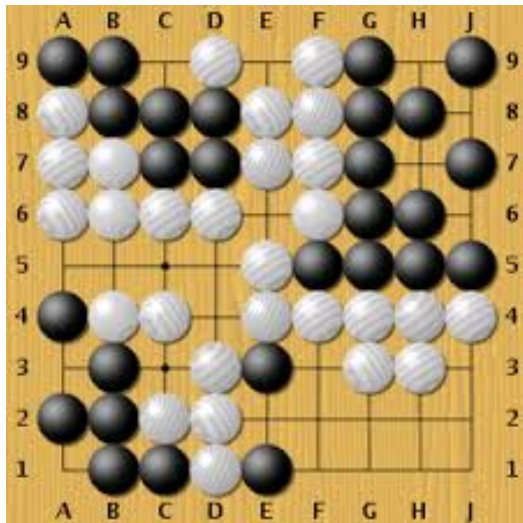


AI and Machine Learning



AI and Machine Learning

Monte Carlo Tree Search



Reinforcement Learning



CENTER FOR DATA-INTENSIVE CYBER-PHYSICAL SYSTEMS



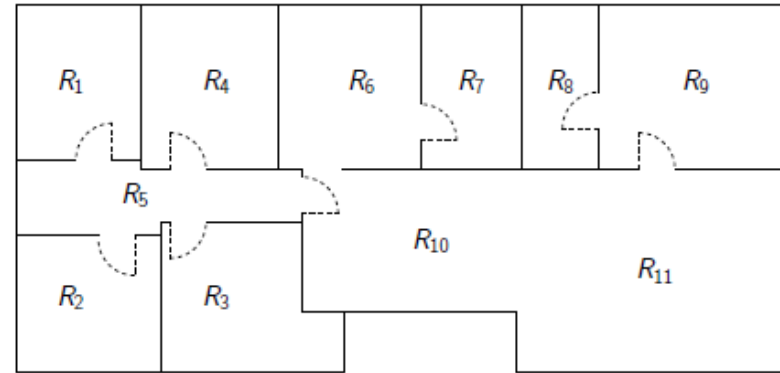
NEOGRID
TECHNOLOGIES

OPTIMAL FLOOR HEATING



Floor Heating Scenario

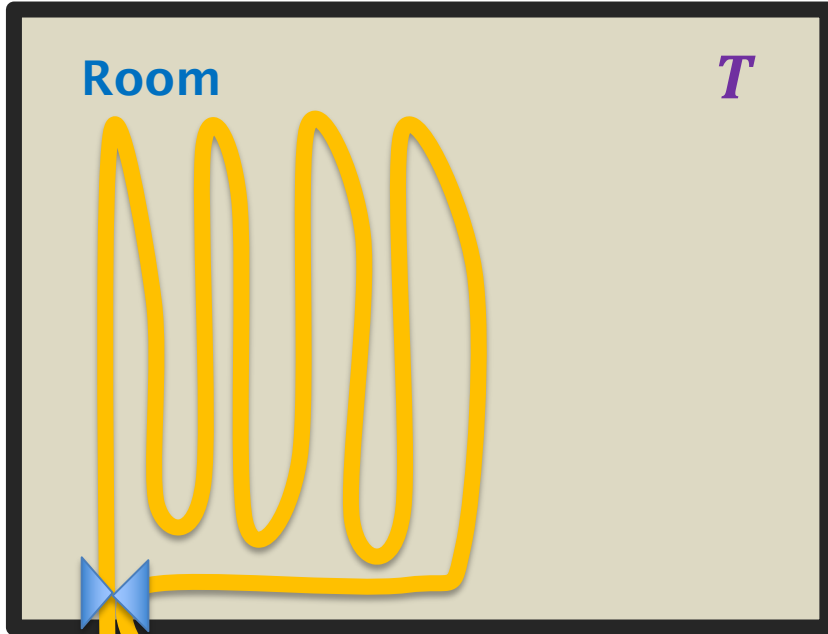
- Each room has a hot water loop that can be opened/closed
- Loops are controlled via activating / deactivating valves.
- Rooms equipped with wireless temperature sensors (report every 15 minutes).
- Each room has its user-defined target temperature.



Control Task:

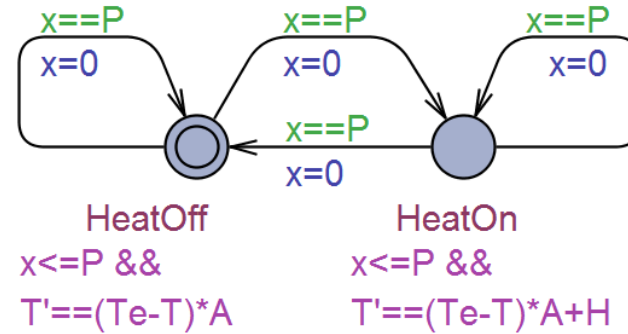
maintain room temperatures as close as possible to target temperatures

1-Room / 1-Window Game



T_e

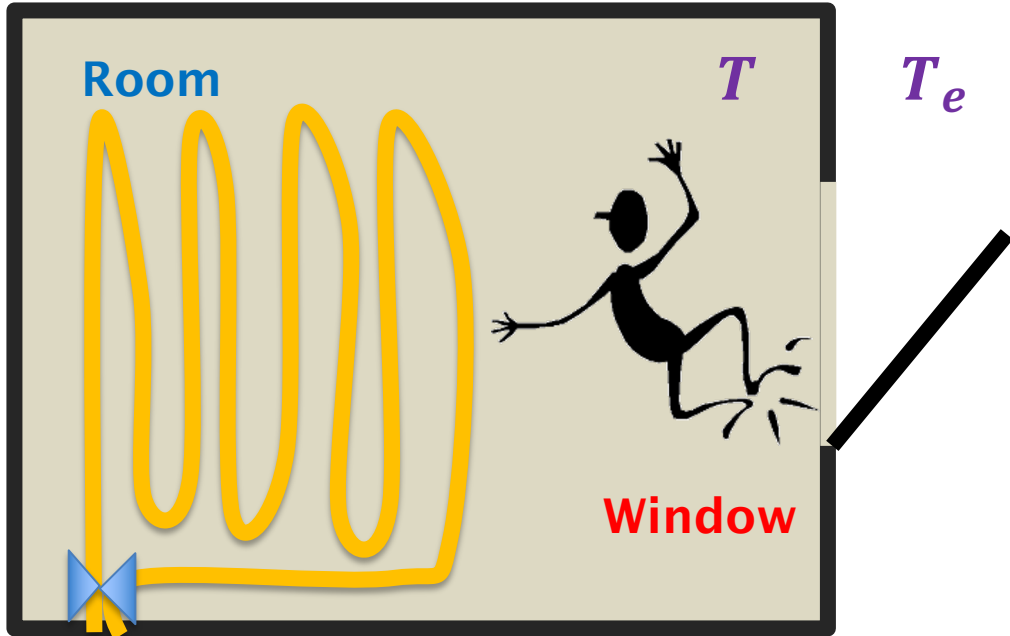
Room



```

const double Tg = 21.0; // room temp. goal
const double Te = 15.0; // environment temp.
const double H = 0.04; // power of heater
const double Aclosed = 0.002; // heat loss when window closed
const double Aopen = 0.004; // heat loss when window open
const int P = 15; // heater switching period
const int h = 60; // 1 hour = 60 time units
    
```

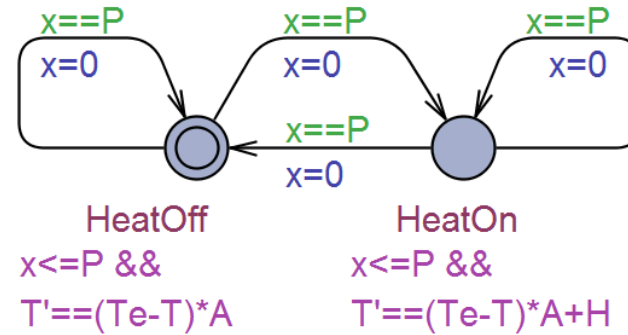
1-Room / 1-Window Game



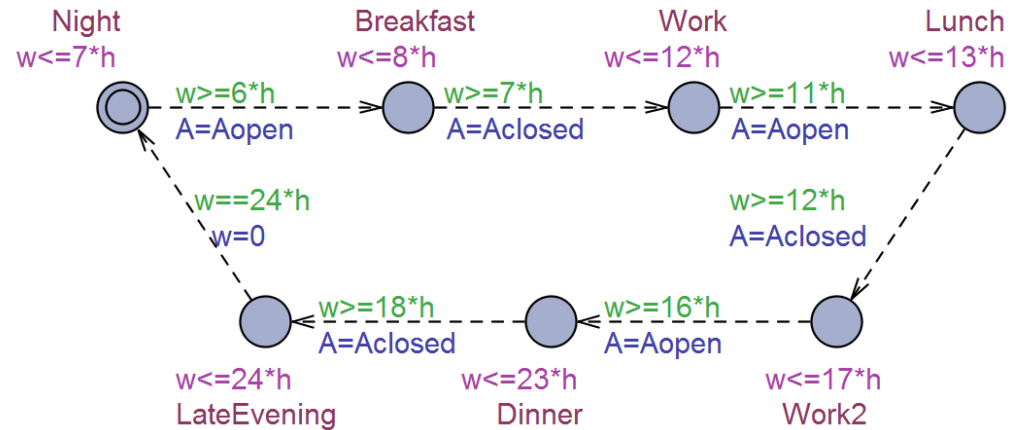
Find **strategy** that minimizes expected **discomfort**:

$$D(H) = \int_{t=0}^{t=H} (T(t) - T_g(t))^2 dt$$

Room



Window



heatedroom.xml - UPPAAL

File Edit View Tools Options Help

Editor Simulator ConcreteSimulator Verifier

Overview

```
// Optimal Control Strategy for 15 Periods
strategy opt = minE (D) [<=15*P]: <> t==15*P
simulate 10 [<=15*P] { T, Window.Open+15, Room.HeatOn+17 } under opt
simulate 10 [<=15*P] { D } under opt
E[<=15*P; 10](max:D) under opt
```

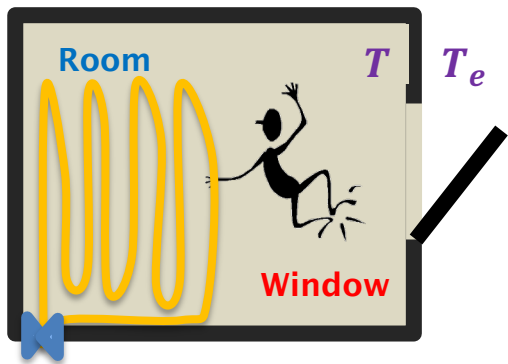
Check
Insert
Remove
Comments

Query

```
simulate 10 [<=30*P] { D }
```

Comment

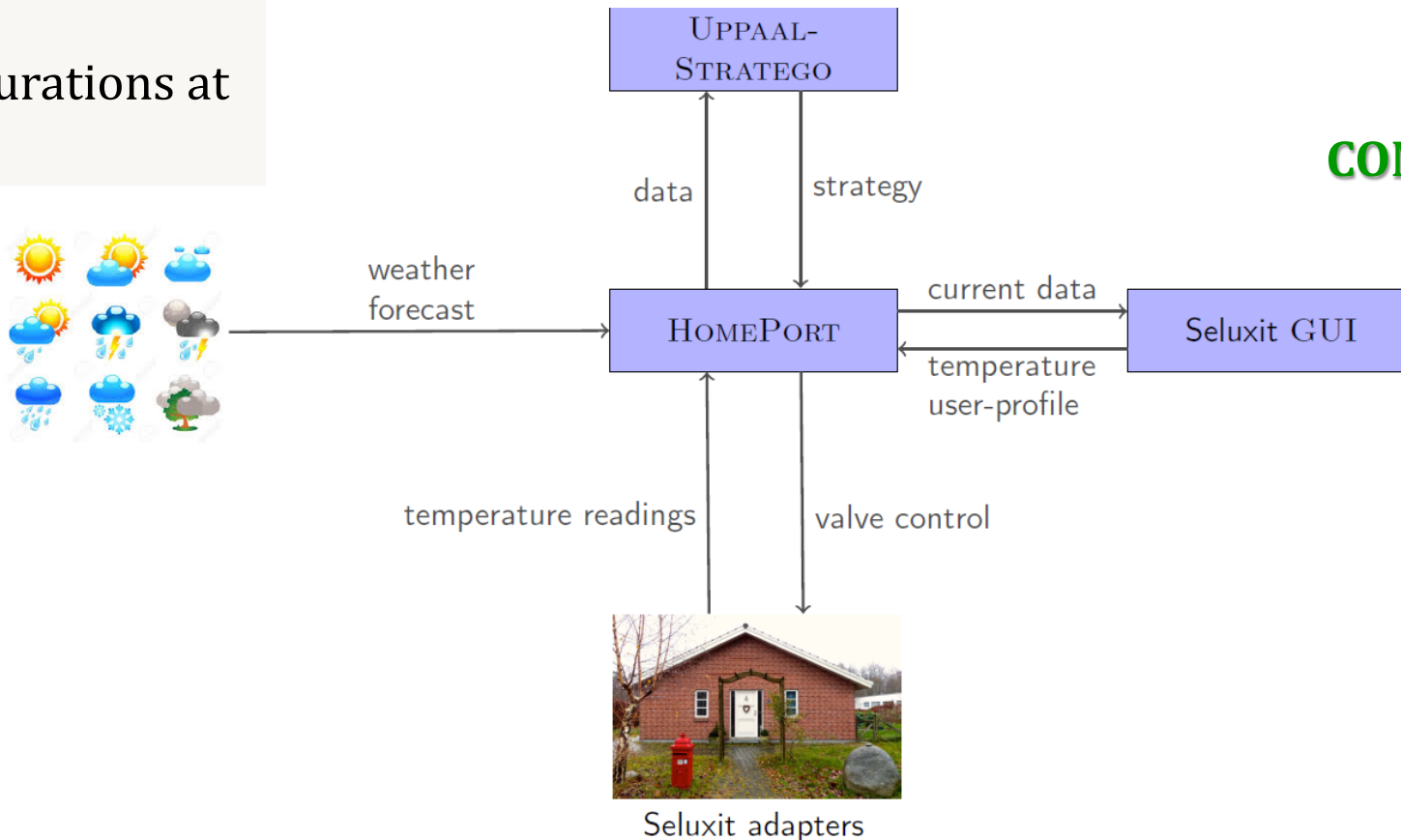
demonstration how the distance function behaves with random controller



Full Floor Heating Case

CHALLENGE

2^{11} valve configurations at each 15 minutes



SYNTHESIS BY LEARNING
ON-LINE SYNTHESIS
COMPOSITIONAL SYNTHESIS

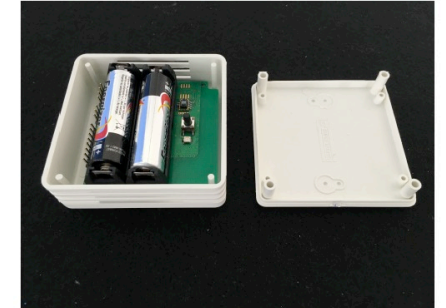
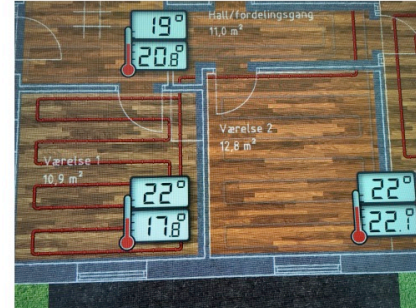


3 day scenario

Full Floor Heating Case

Weather	Distance			Energy		
	Bang-Bang	Stratego	imp.	Bang-Bang	Stratego	imp.
Aalborg	14583	8342	43%	14180	12626	10%
Anadyr	2385515	1483272	37%	23040	22475	2%
Ankara	17985	10464	41%	17468	15684	10%
Minneapolis	22052	12175	44%	18165	15882	12%
Murmansk	399421	187941	52%	22355	21011	6%

Weather	Distance			Energy		
	Bang-Bang	Stratego	imp.	Bang-Bang	Stratego	imp.
Aalborg	14583	8552	41%	14180	12590	11%
Anadyr	2385515	1503448	36%	23040	22371	2%
Ankara	17985	10511	41%	17468	15697	10%
Minneapolis	22052	12725	42%	18165	15837	12%
Murmansk	399421	191441	52%	22355	20923	6%



Evaluation of under modified parameters (0-20%)



CENTER FOR DATA-INTENSIVE CYBER-PHYSICAL SYSTEMS



FLEXIBILITY & FLEX OFFERS

FLEXIBILITY

START BY CLICKING HERE

POWERED BY UPPAAL v.4

1. PICK A WEATHER TYPE

18°C / 64°F
WIND: 12 M/S / 5 MPH

18°C / 64°F
WIND: 12 M/S / 5 MPH

18°C / 64°F
WIND: 12 M/S / 5 MPH

18°C / 64°F
WIND: 12 M/S / 5 MPH

18°C / 64°F
WIND: 12 M/S / 5 MPH



2. PLACE YOUR DAILY TASKS IN THE TIMELINE

COFFEE

LAUNDRY

DISHWASHER

DRY CLOTHES

CAR RECHARGING

LAPTOP RECHARGING

MOBILE RECHARGING

VACUUMCLEANER RECHARGING

HEATING

FLOOR HEATING

VENTILATION

00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23

YOU



\$37

00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23

AVERAGE CONSUMER



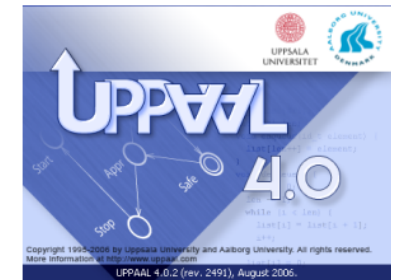
\$58

HOW GREEN DO YOU WANT TO BE?

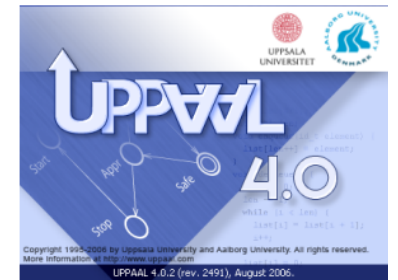
A LOT

A BIT

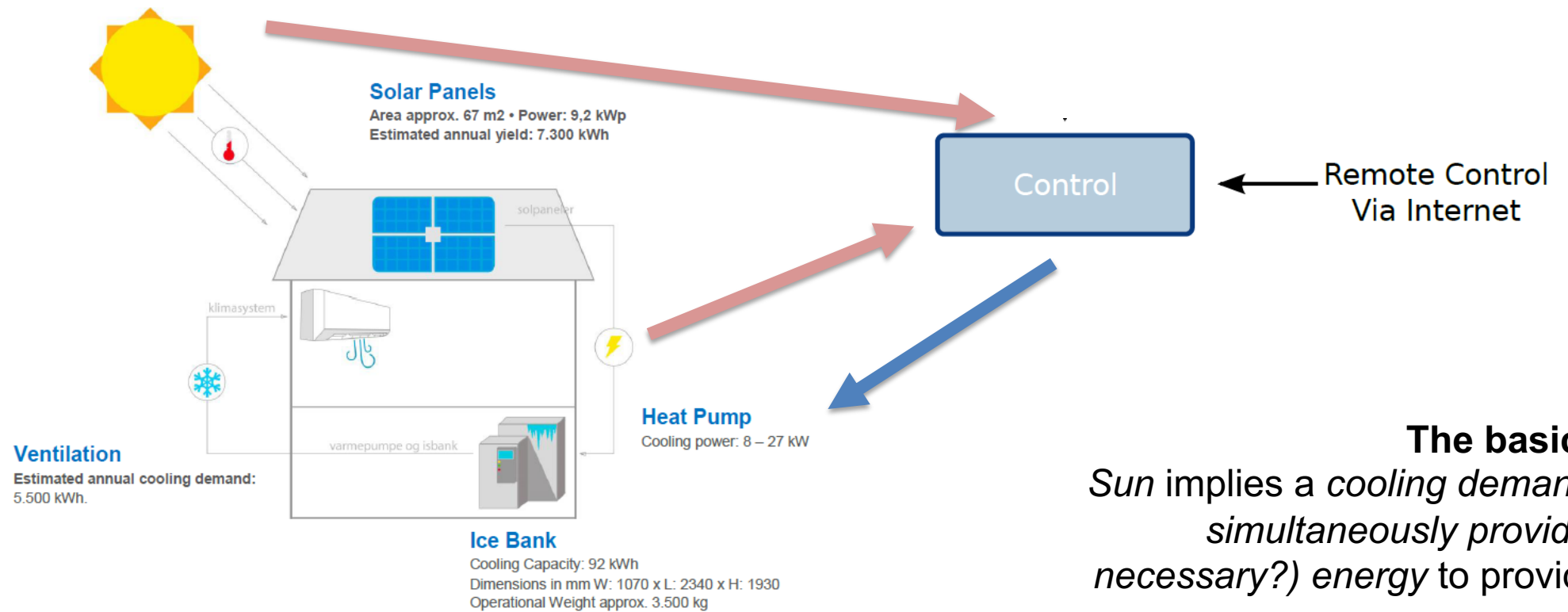
NOT AT ALL



FLEXIBILITY



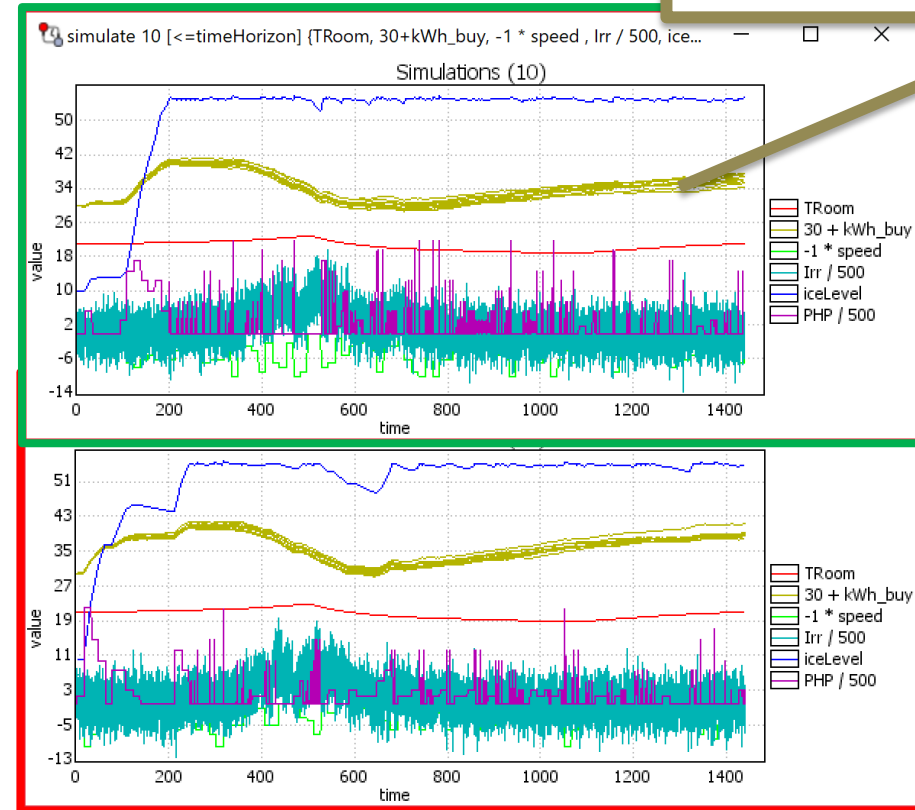
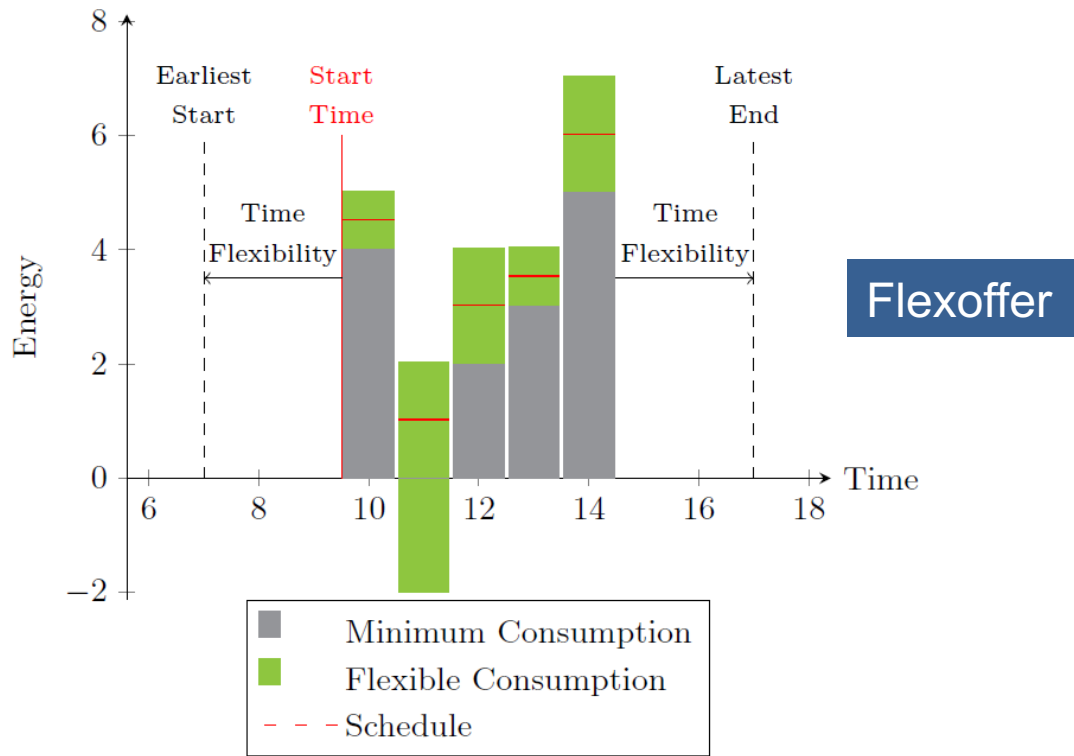
FLEX OFFERS and Reinforcement Learning



The basic idea
Sun implies a cooling demand and simultaneously provide (the necessary?) energy to provide the cooling.

FLEX OFFERS

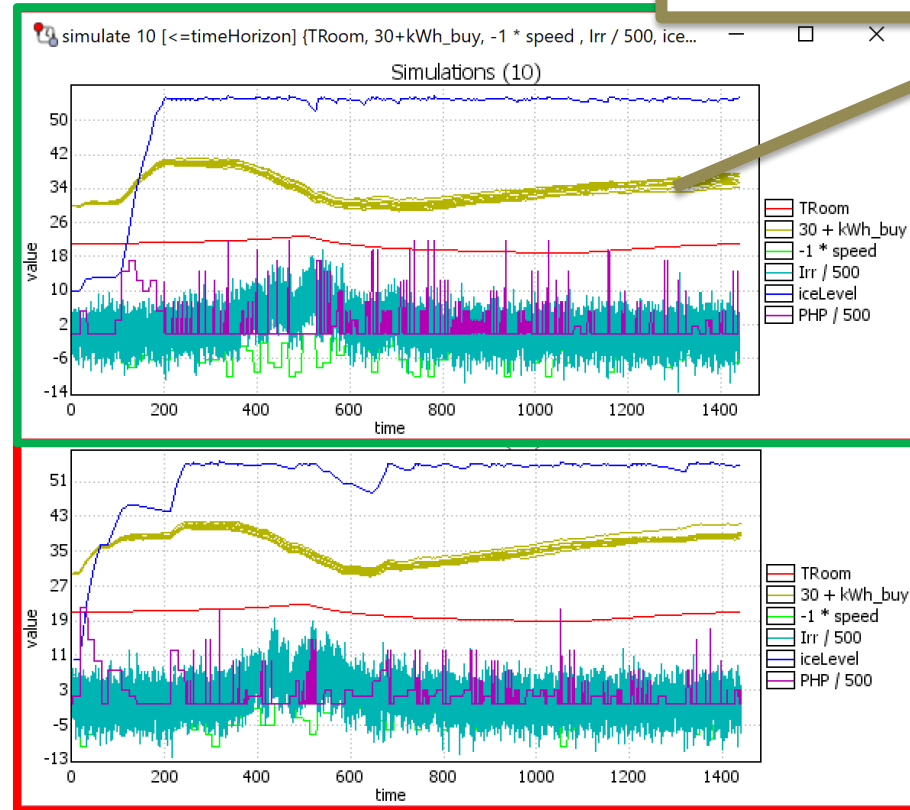
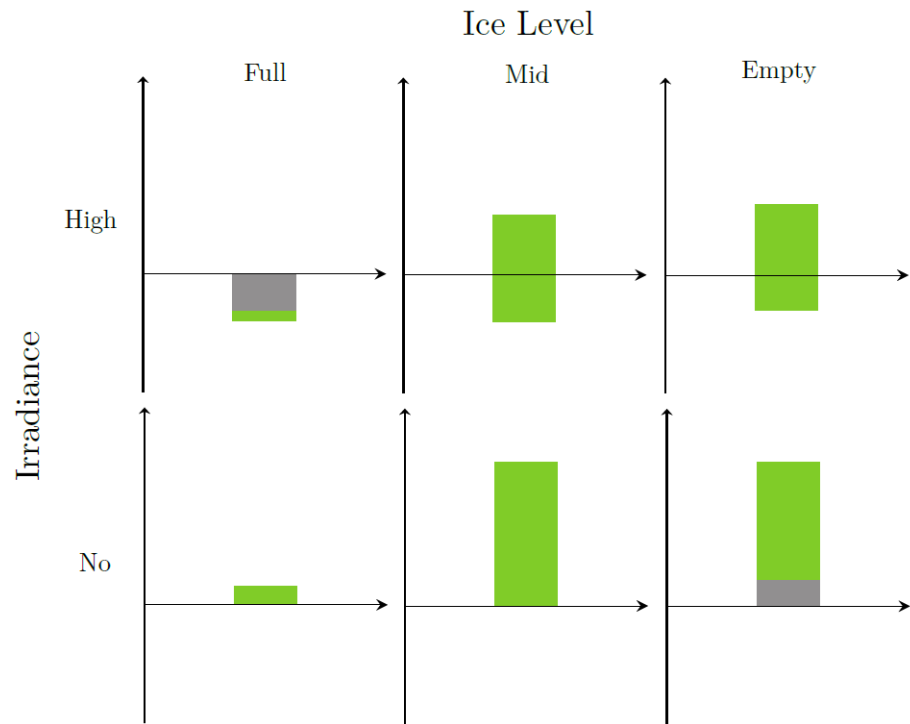
Probabilistic flex-offers found from strategies for minimizing (maximizing) kWh_buy



strategy min(max) kWh =
 min(max) E (kWh_buy) [<=H]: <> (i - offset) == H

FLEX OFFERS

Probabilistic flex-offers found from strategies for minimizing (maximizing) kWh_buy



strategy min(max) kWh =
 $\min(\max) E(\text{kWh_buy}) [\leq H] : \langle \rangle (i - \text{offset}) == H$

MORE GAMES USING UPPAAL

- **Traffic Control**
- Zone-based climate control **pig-stables**
- Profit-optimal, energy-aware schedules for **satellites**
- Optimal control of **heat-pumps**
- Personalized **light control** in home automation
- Safe and energy optimal control of **hydraulic pumps**
- **COVID19**



www.uppaal.org