From Offline Long-Run to Online Short-Run: Exploring A New Approach of Hybrid Systems Model Checking for MDPnP

Tao Li*, Qixin Wang*, Feng Tan*, Lei Bu, Jian-nong Cao*,
Xue Liu, Yufei Wang*, Rong Zheng

*The Hong Kong Polytechnic Univ.

CPS Week 2011

THE HONG KONG
POLYTECHNIC UNIVERSITY
香港理工大學

Content

Demand

Background

Challenge

Solution

Evaluation

Related Work

Content

Demand

Background

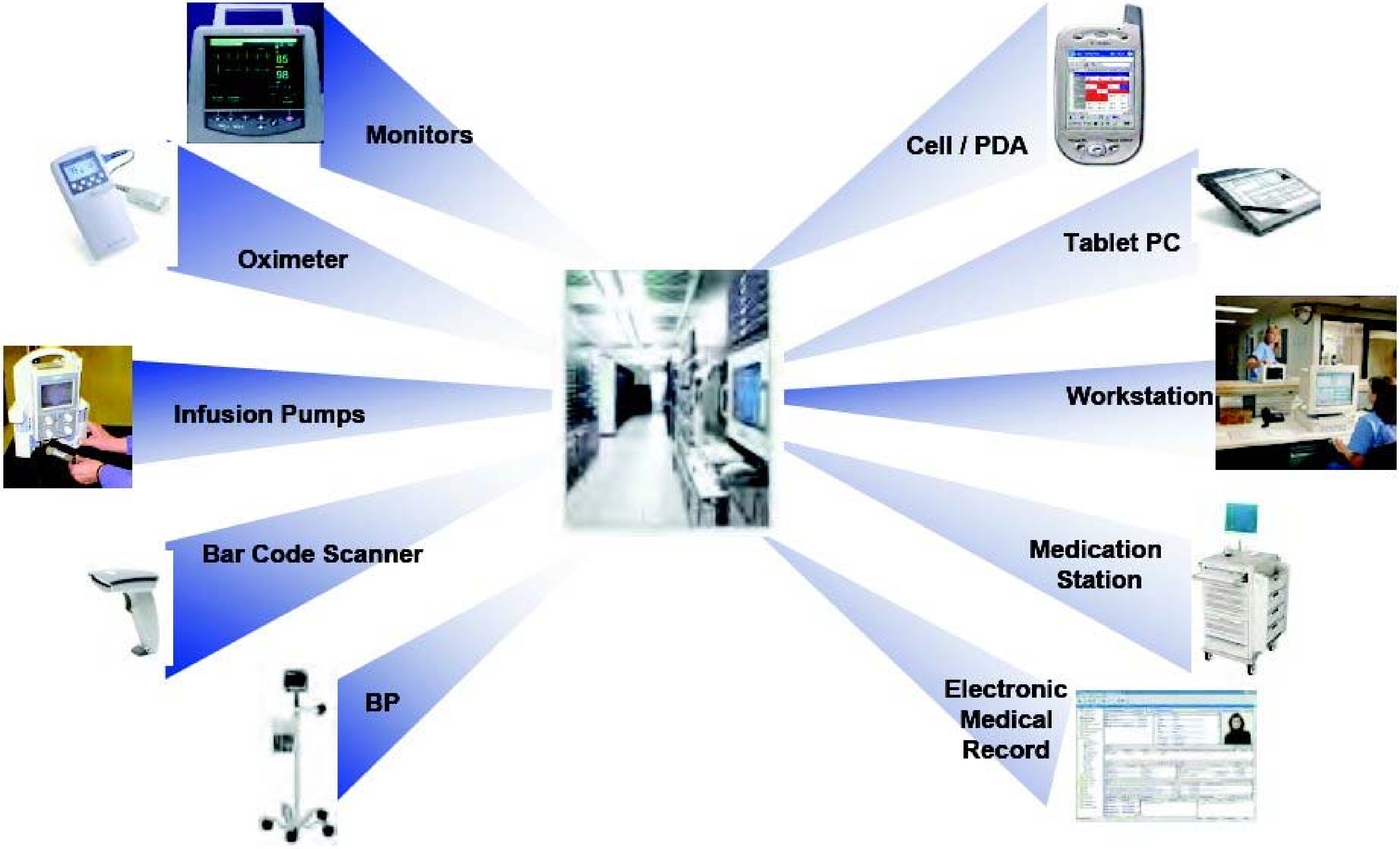
Challenge

Solution

Evaluation

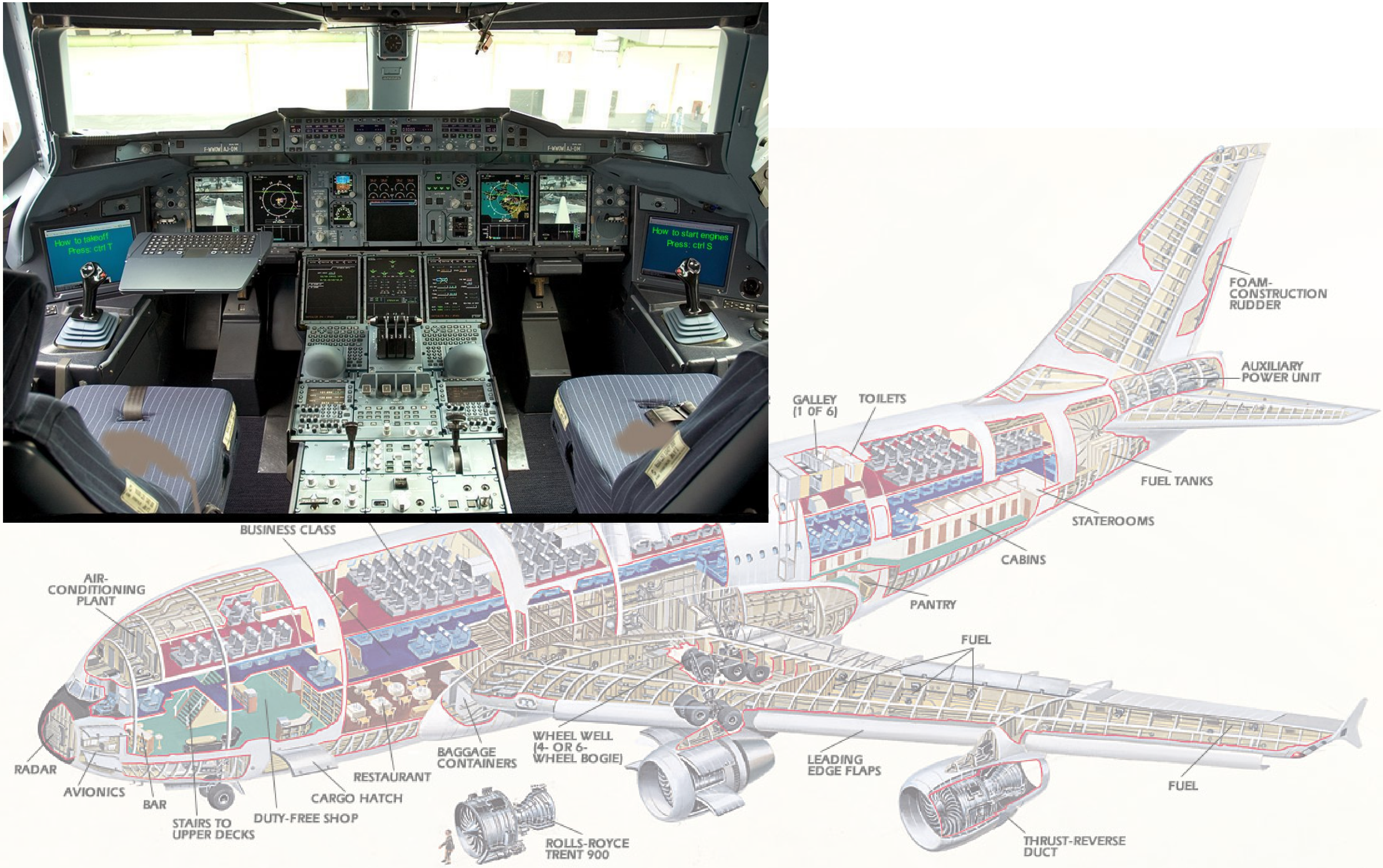
Related Work

MDPnP leads to better safety, capability, and convenience of medical settings.



MDPnP can help prevent many serious/lethal accidents in medical settings.

Following the success of requiring avionics to be verifiably safe → MDPnP to be verifiably safe.



Content

Demand

Background

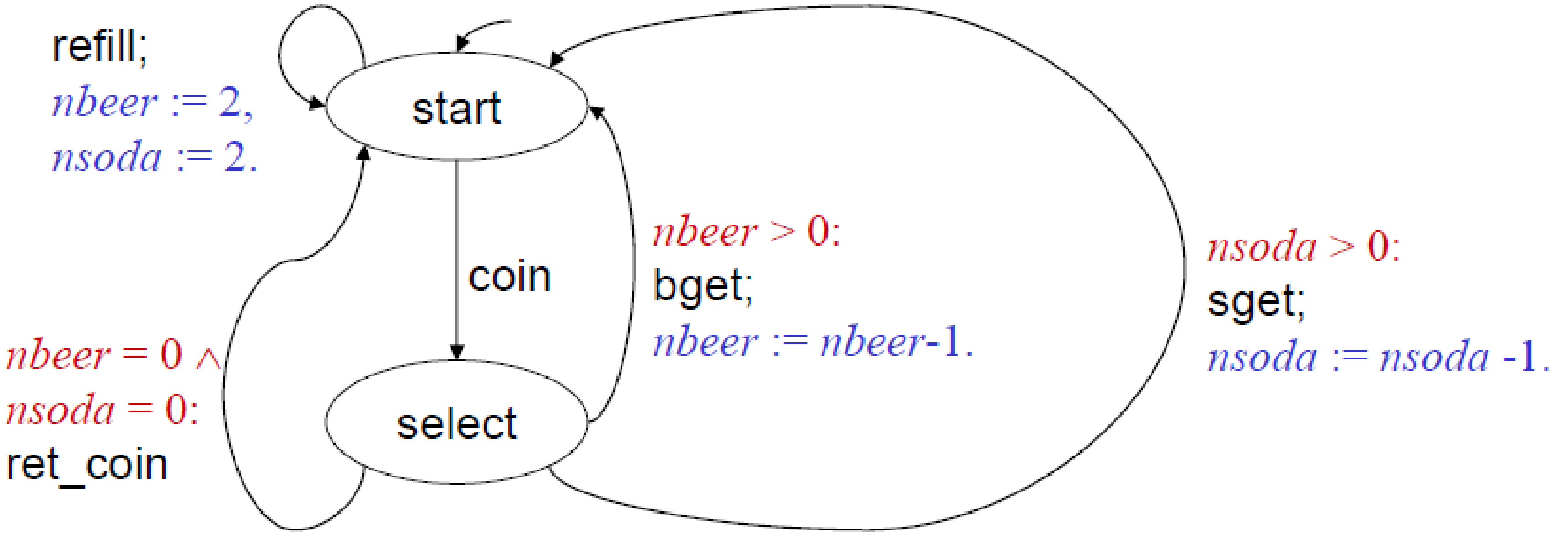
Challenge

Solution

Evaluation

Related Work

A key tool for traditional computer systems verification is model checking.



refill;
$nbeer := 2,$
$nsoda := 2.$

start

coin

select

$nbeer > 0:$
bget;
$nbeer := nbeer\text{-}1.$

$nsoda > 0:$
sget;
$nsoda := nsoda\ \text{-}1.$

$nbeer = 0 \wedge$
$nsoda = 0:$
ret_coin

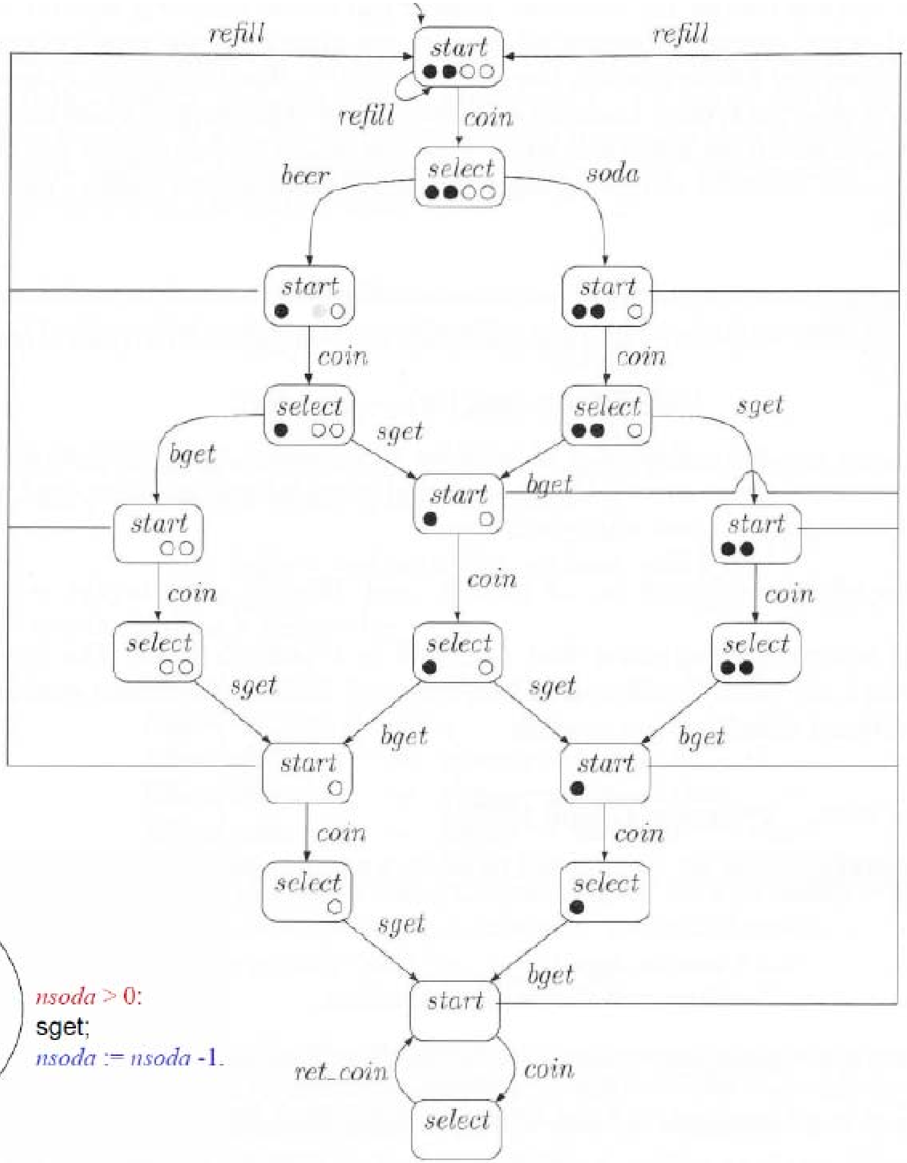
$Var = \{nbeer, nsoda\}$, $\text{domain}(nbeer) = \{0, 1, 2\}$, $\text{domain}(nsoda) = \{0, 1, 2\}$

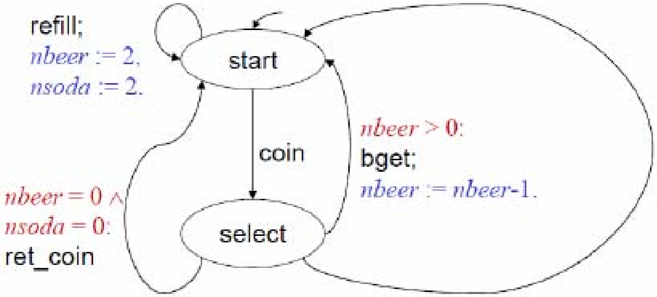
$$PG = (Loc, Act, Effect, :\rightarrow, Loc_0, g_0)$$

Computer systems model checking verifies safety, liveliness, persistence, and other properties.

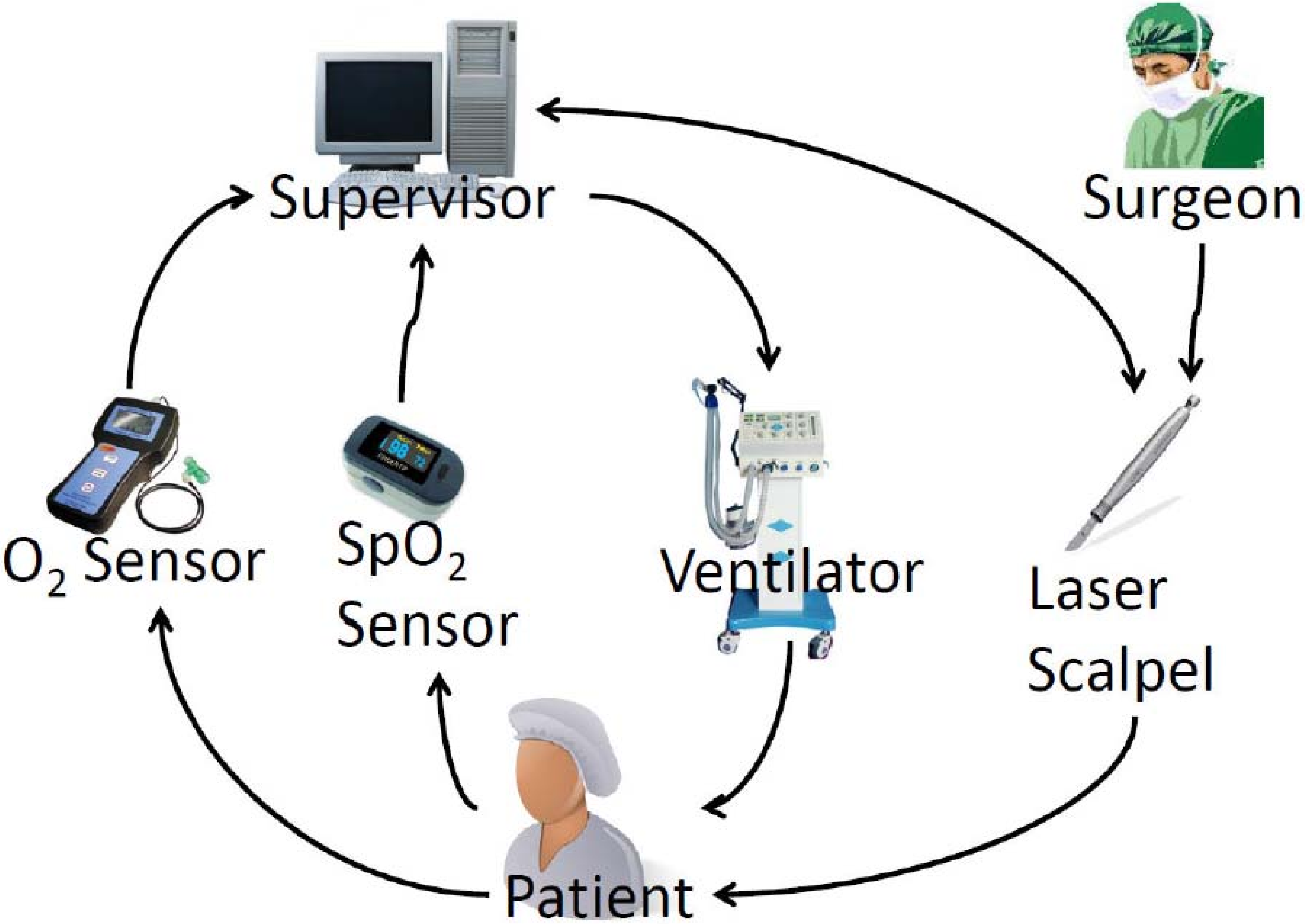
Transition System of a Program Graph Example

Note the combinatorial explosion of size.





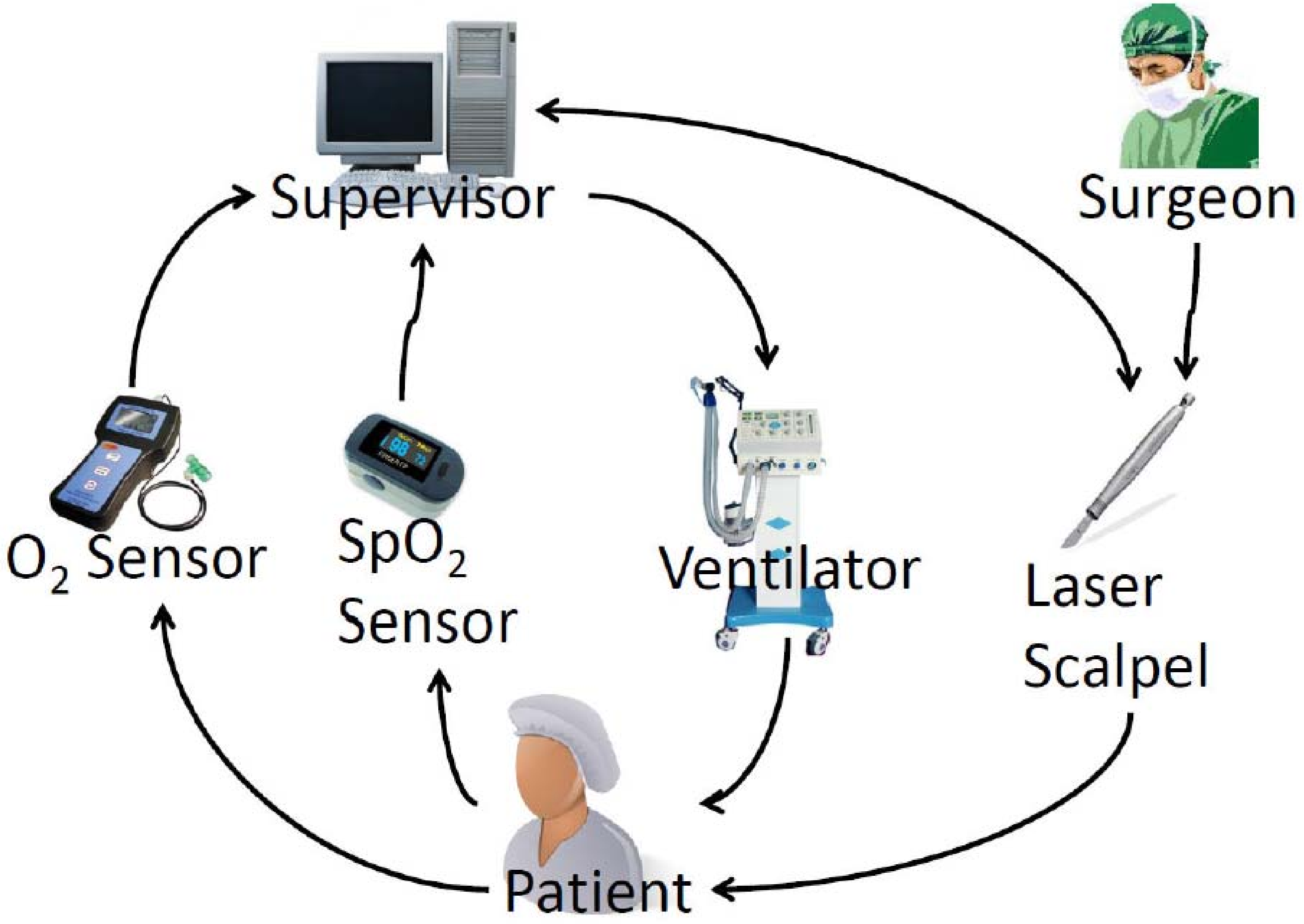
MDPnP is not just a computer system, it is a hybrid of computer & other systems, i.e., CPS.



Laser Tracheotomy MDPnP

MDPnP is not just a computer system, it is a hybrid of computer & other systems, i.e., CPS.

Computer

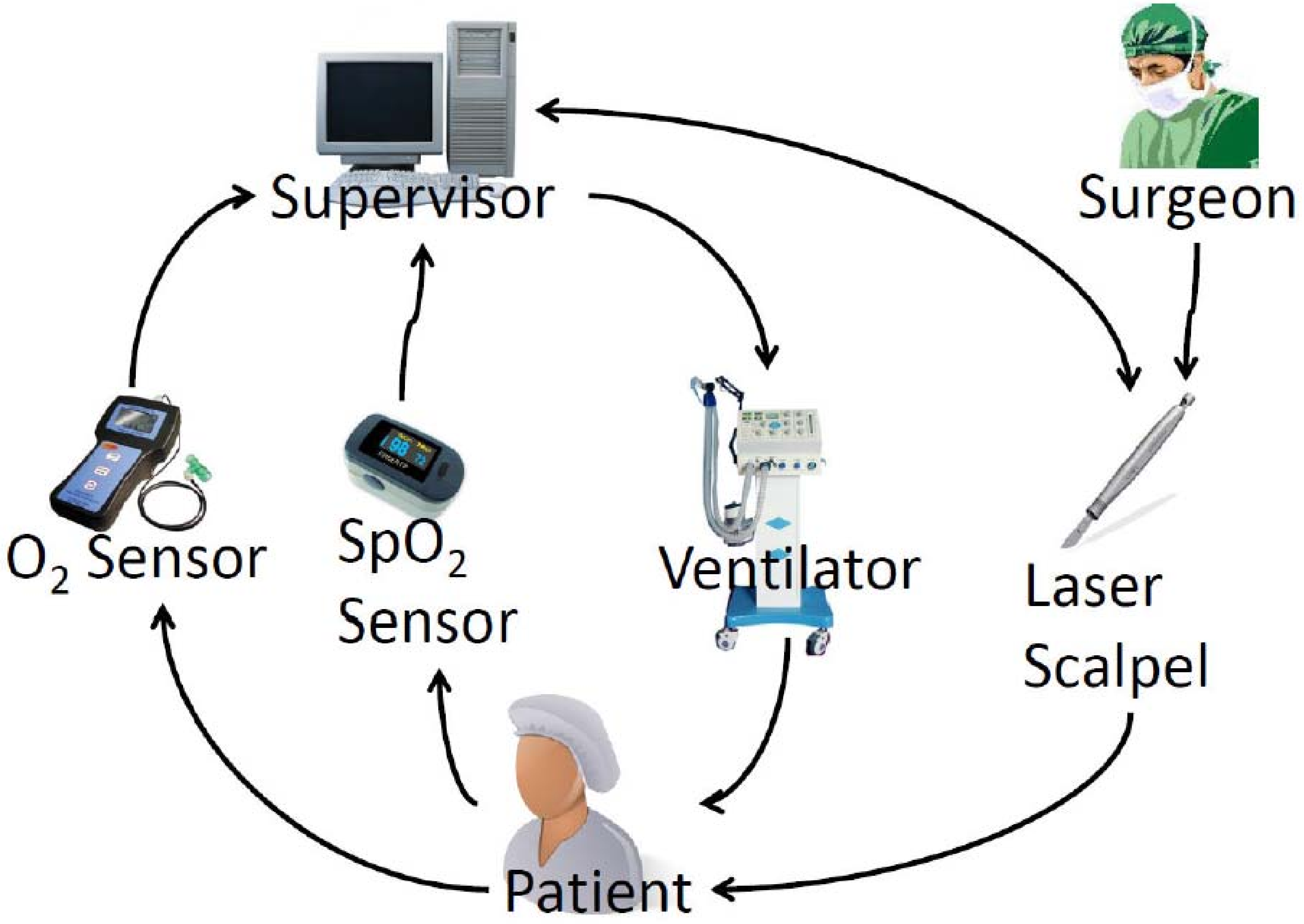


Laser Tracheotomy MDPnP

MDPnP is not just a computer system, it is a hybrid of computer & other systems, i.e., CPS.

Computer

Biochemical



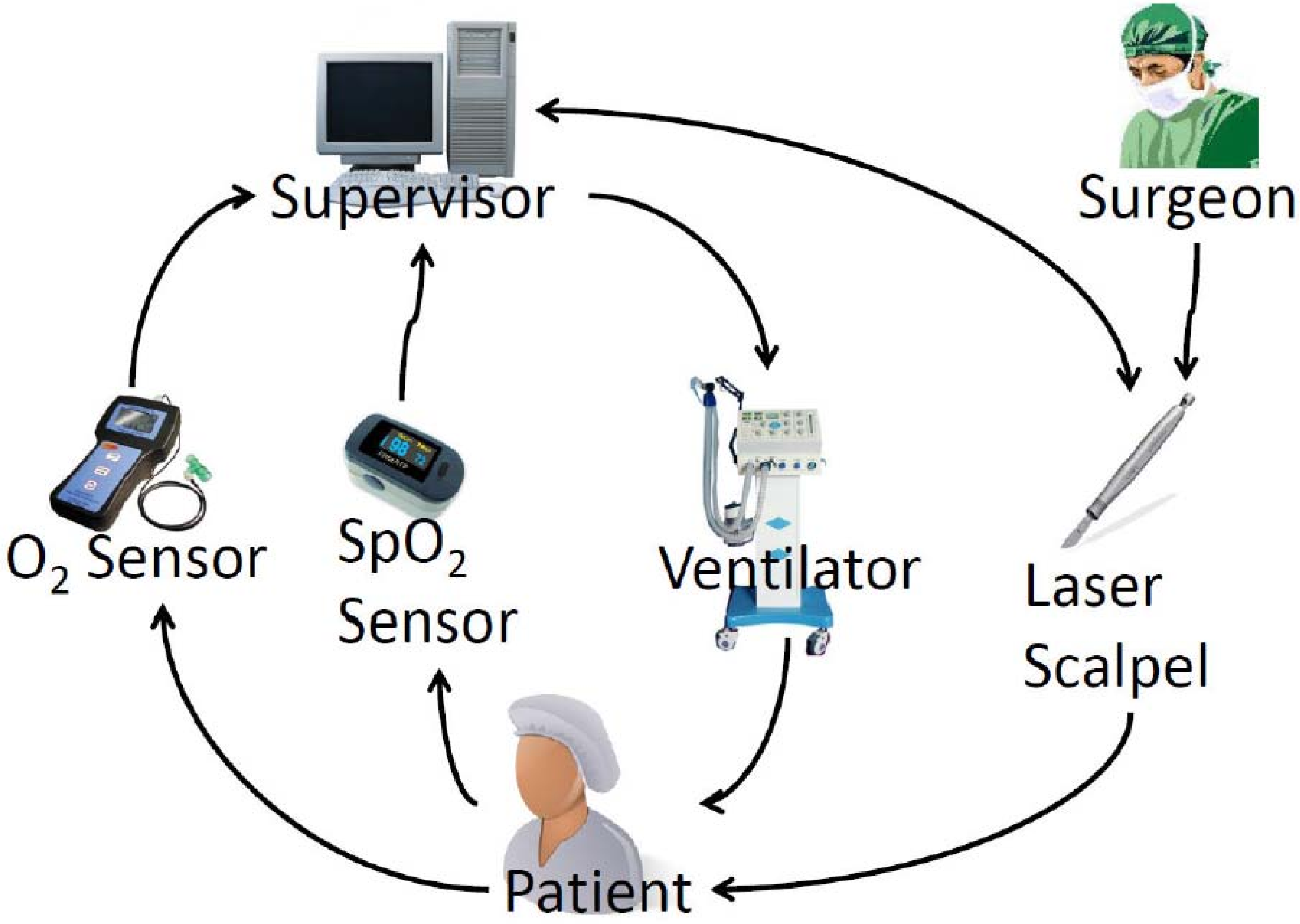
Laser Tracheotomy MDPnP

MDPnP is not just a computer system, it is a hybrid of computer & other systems, i.e., CPS.

Computer

Biochemical

Mechanical



Laser Tracheotomy MDPnP

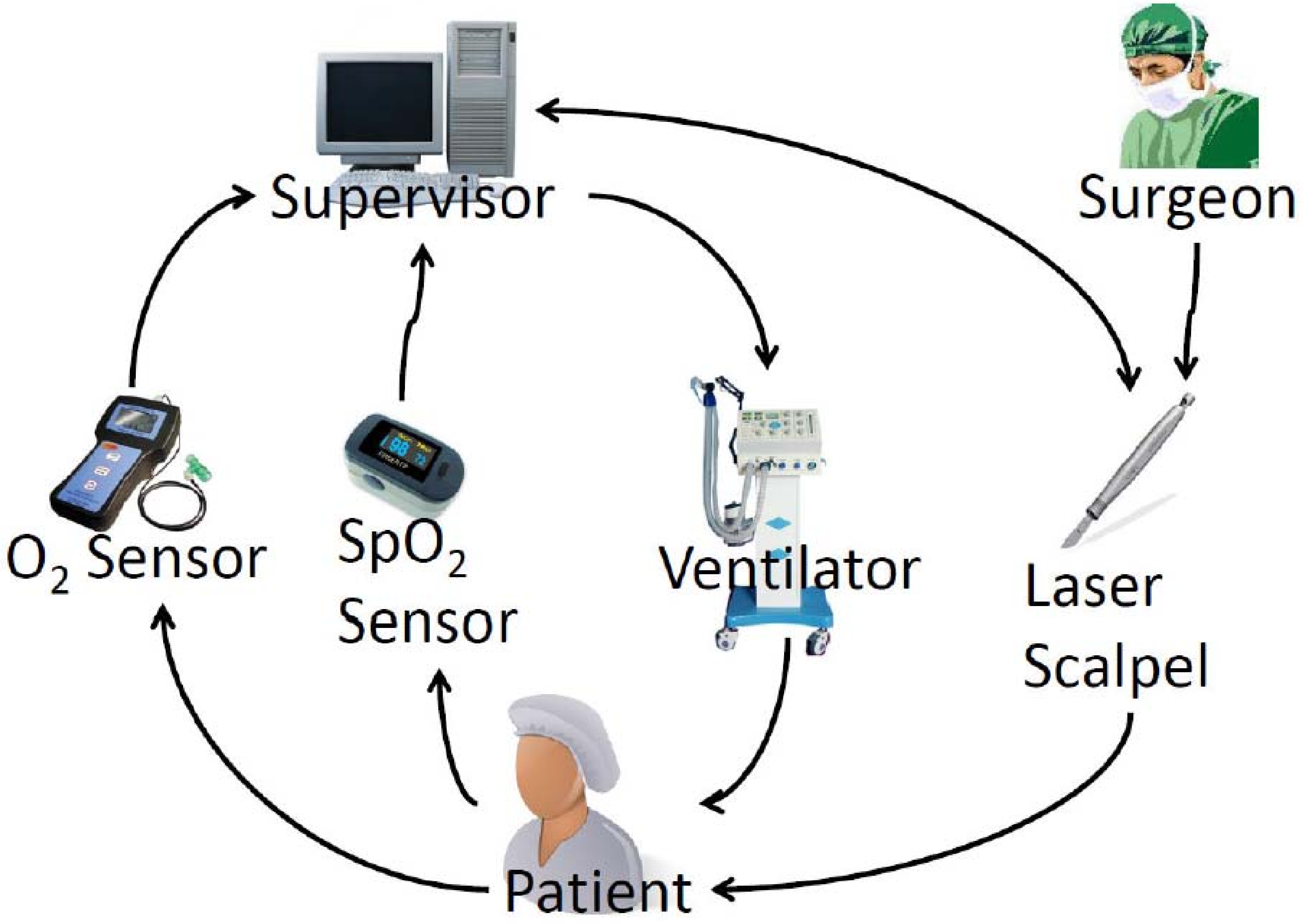
MDPnP is not just a computer system, it is a hybrid of computer & other systems, i.e., CPS.

Computer

Biochemical

Mechanical

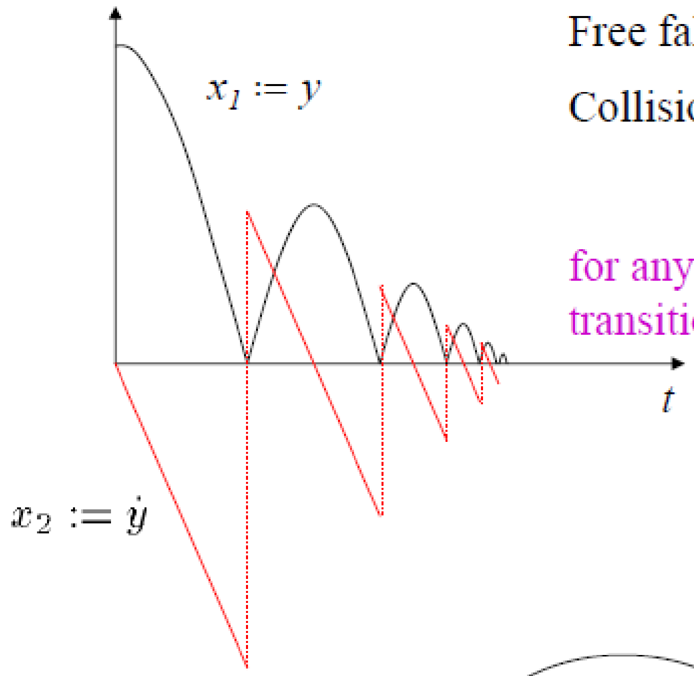
Communication



Laser Tracheotomy MDPnP

A state-of-the-art CPS model checking is Hybrid Systems Model Checking: Comp + Fdbk Ctrl.

Bouncing Ball Example



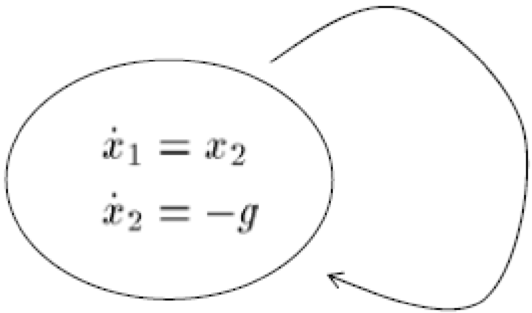
Free fall $\equiv \quad \ddot{y} = -g$

$$\text{Collision} \equiv \quad \begin{aligned} y^+(t) &= y^-(t) = 0 \\ \dot{y}^+(t) &= -c\dot{y}^-(t) \end{aligned}$$

for any c < 1, there are infinitely many transitions in finite time (Zeno phenomena)

guard or jump condition

$x_1 = 0$ & $x_2 < 0$?



$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -g$$

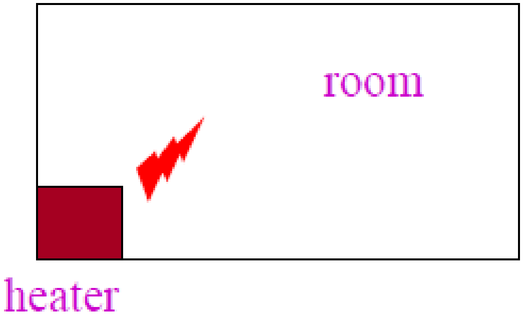
transition

$X_2 := - \; c \; X_2^-$

state reset

The state-of-the-art CPS model checking is Hybrid Systems Model Checking: Comp + Fdbk Ctrl.

Thermostat Example



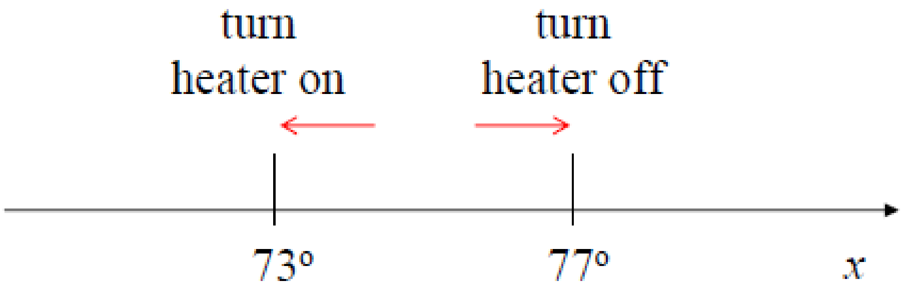
goal ≡ regulate temperature around 75°

$x \equiv$ mean temperature

when heater is off: $\dot{x} \approx -x + 50$ $(x \to 50^\circ)$

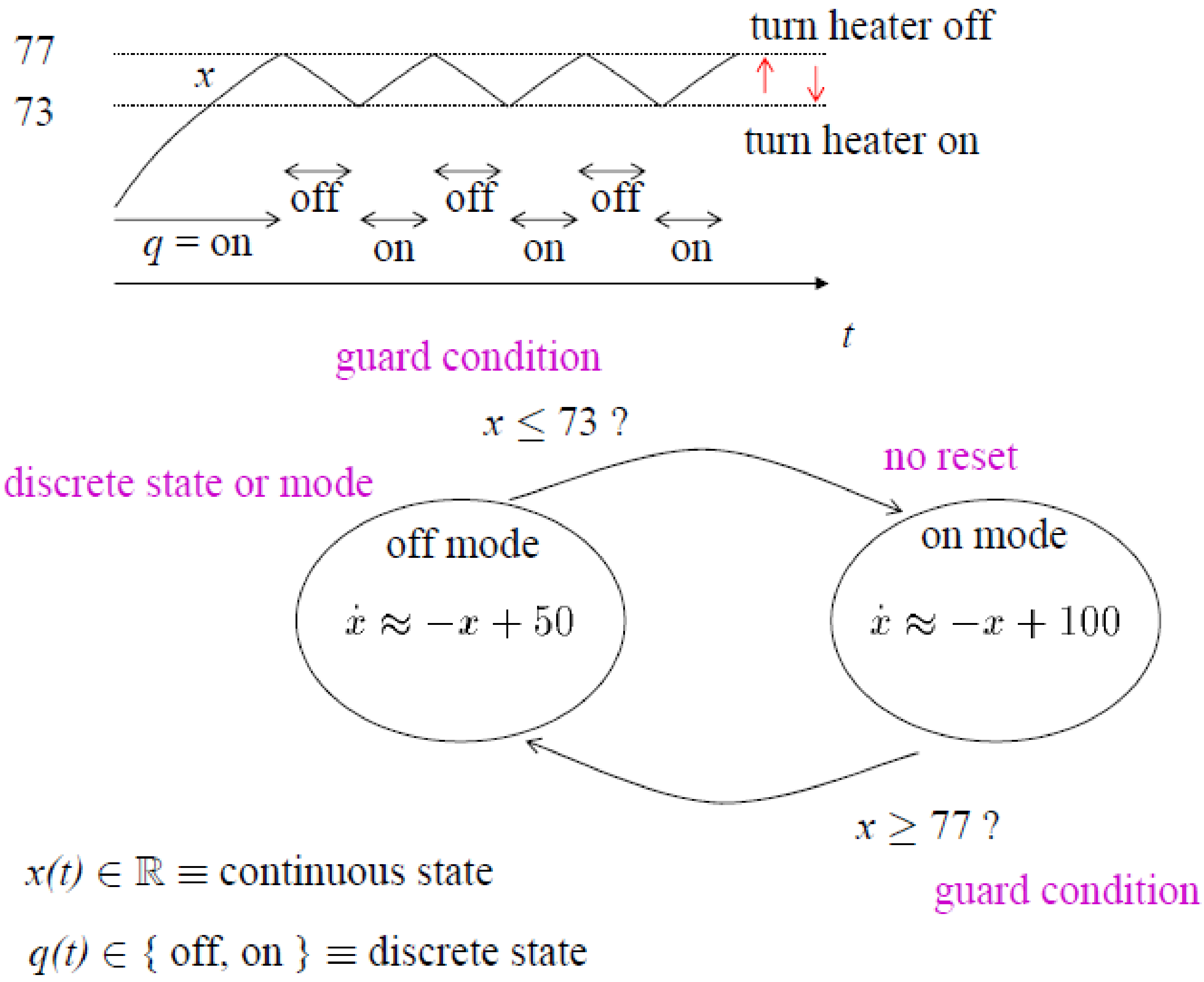
when heater is on: $\dot{x} \approx -x + 100$ $(x \to 100^\circ)$

event-based control



The state-of-the-art CPS model checking is Hybrid Systems Model Checking: Comp + Fdbk Ctrl.

Thermostat Example



$x(t) \in \mathbb{R} \equiv$ continuous state

$q(t) \in \{$ off, on $\} \equiv$ discrete state

Content

Demand

Background

Challenge

Solution

Evaluation

Related Work

However, existing hybrid systems model checking (computer + fdbk ctrl) doesn’t very well fit MDPnP.

However, existing hybrid systems model checking (computer + fdbk ctrl) doesn’t very well fit MDPnP.

Existing model checking:

Offline (partly due to lack of time cost bound),

Time-Unbounded Behavior (Long-Run Future)

However, existing hybrid systems model checking (computer + fdbk ctrl) doesn’t very well fit MDPnP.

Existing model checking:

Offline (partly due to lack of time cost bound),

Time-Unbounded Behavior (Long-Run Future)

Challenge 1: No good offline models for complex biomedical systems of human body.

However, existing hybrid systems model checking (computer + fdbk ctrl) doesn’t very well fit MDPnP.

Existing model checking:

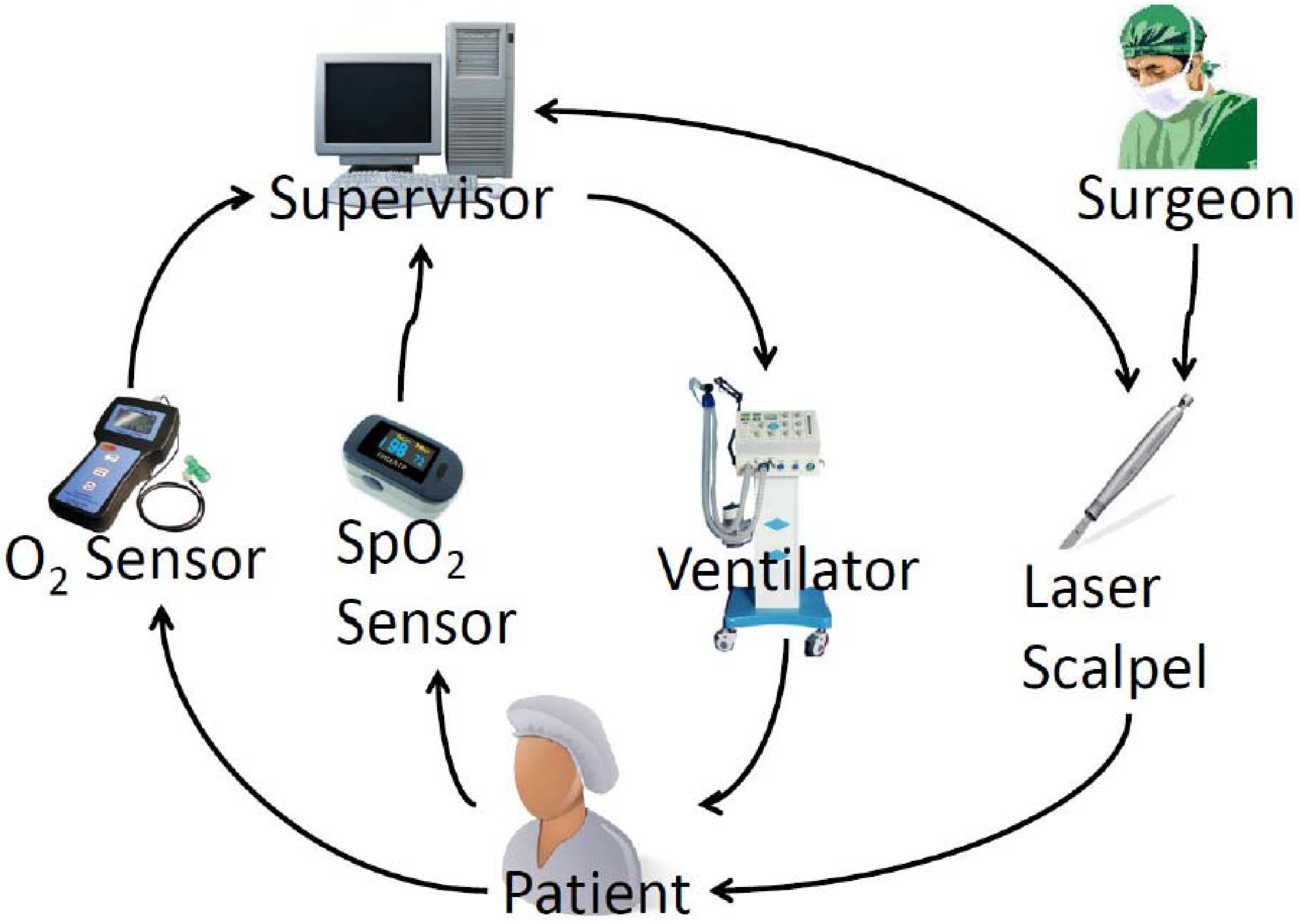
Offline (partly due to lack of time cost bound),

Time-Unbounded Behavior (Long-Run Future)

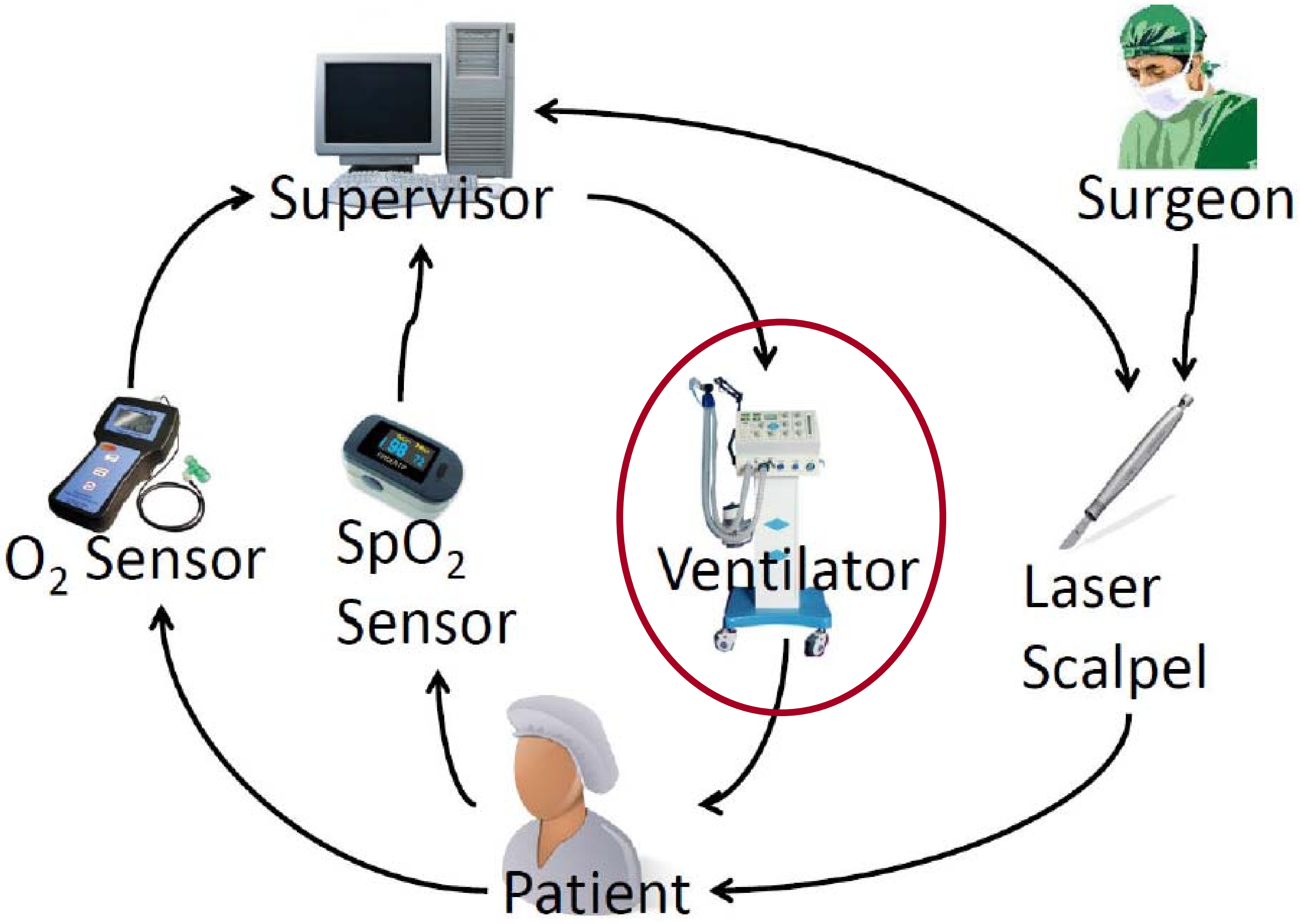
Challenge 1: No good offline models for complex biomedical systems of human body.

Challenge 2: Verification state space easily explode.

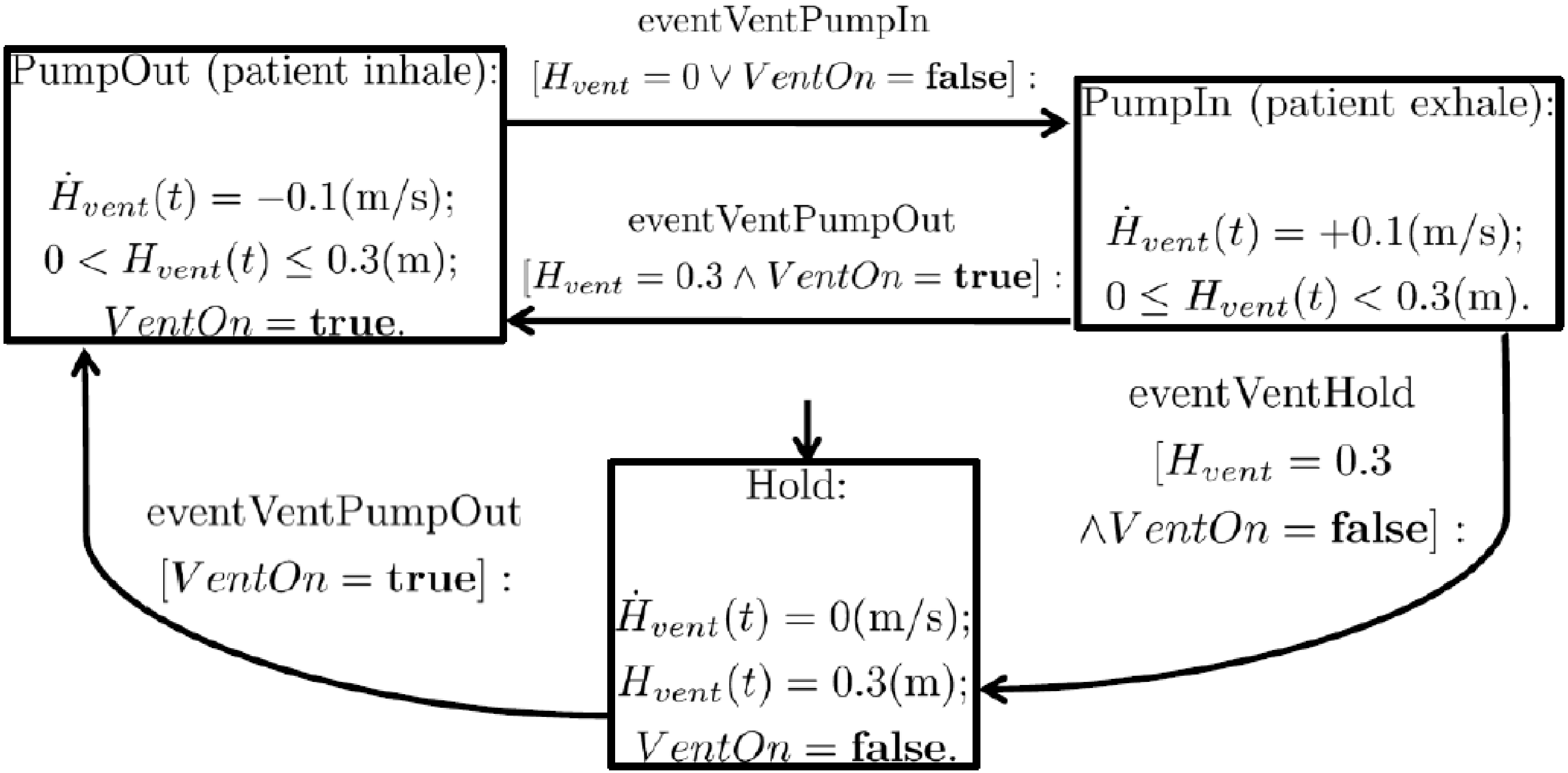
Take laser tracheotomy offline hybrid systems modeling as an example.



Take laser tracheotomy offline hybrid systems modeling as an example.



Take laser tracheotomy offline hybrid systems modeling as an example.



PumpOut (patient inhale):

$\dot{H}_{vent}(t) = -0.1(\mathrm{m/s})$;

$0 < H_{vent}(t) \leq 0.3(\mathrm{m})$;

$VentOn = \textbf{true}$.

eventVentPumpIn

$[H_{vent} = 0 \vee VentOn = \textbf{false}]$:

PumpIn (patient exhale):

$\dot{H}_{vent}(t) = +0.1(\mathrm{m/s})$;

$0 \leq H_{vent}(t) < 0.3(\mathrm{m})$.

eventVentPumpOut

$[H_{vent} = 0.3 \wedge VentOn = \textbf{true}]$:

eventVentHold

$[H_{vent} = 0.3$

$\wedge VentOn = \textbf{false}]$:

Hold:

$\dot{H}_{vent}(t) = 0(\mathrm{m/s})$;

$H_{vent}(t) = 0.3(\mathrm{m})$;

$VentOn = \textbf{false}$.

eventVentPumpOut

$[VentOn = \textbf{true}]$:

Legend:

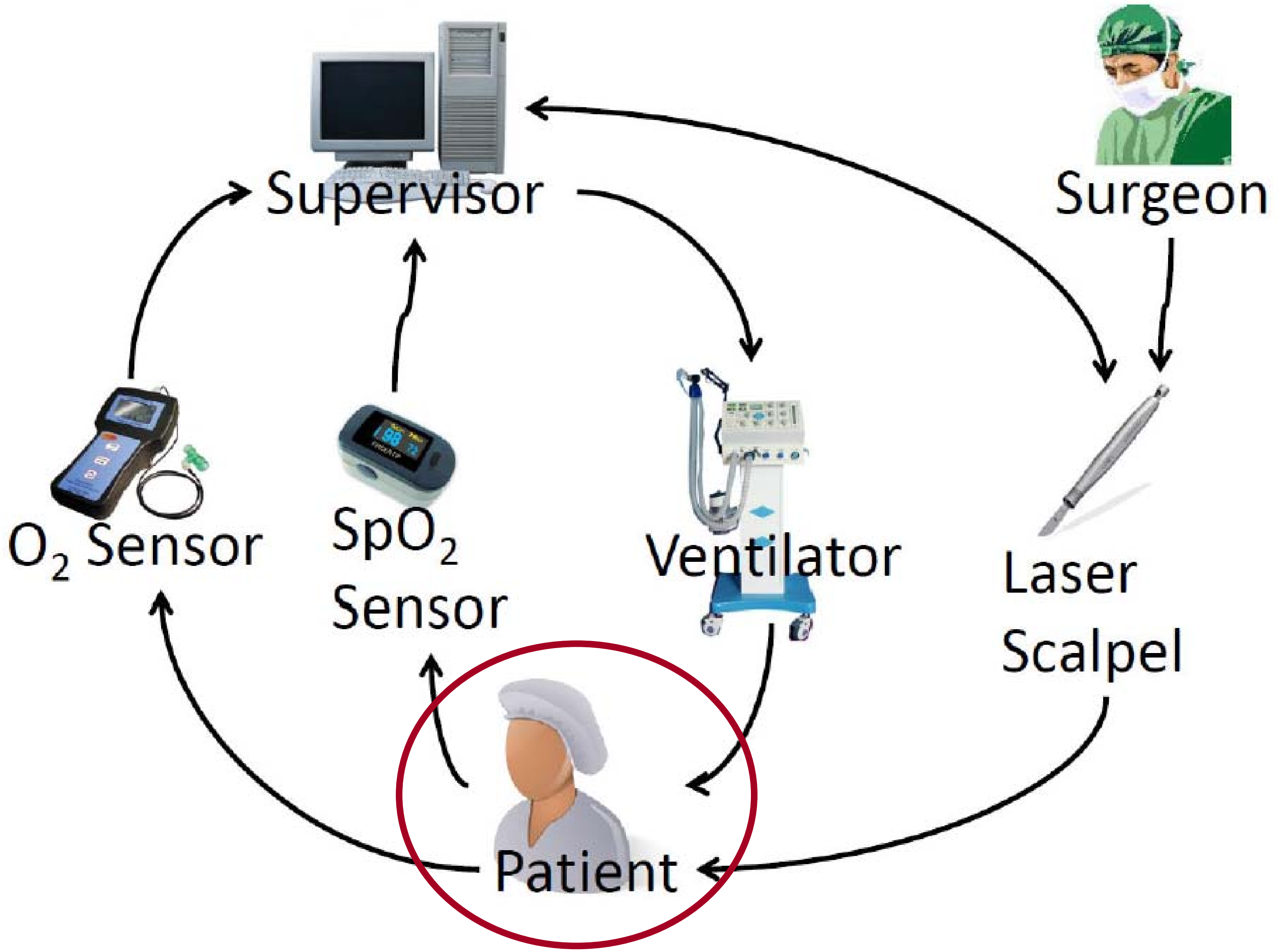
→ (w/ source location) Event; (w/o source location) Initial location indicator

□ Location

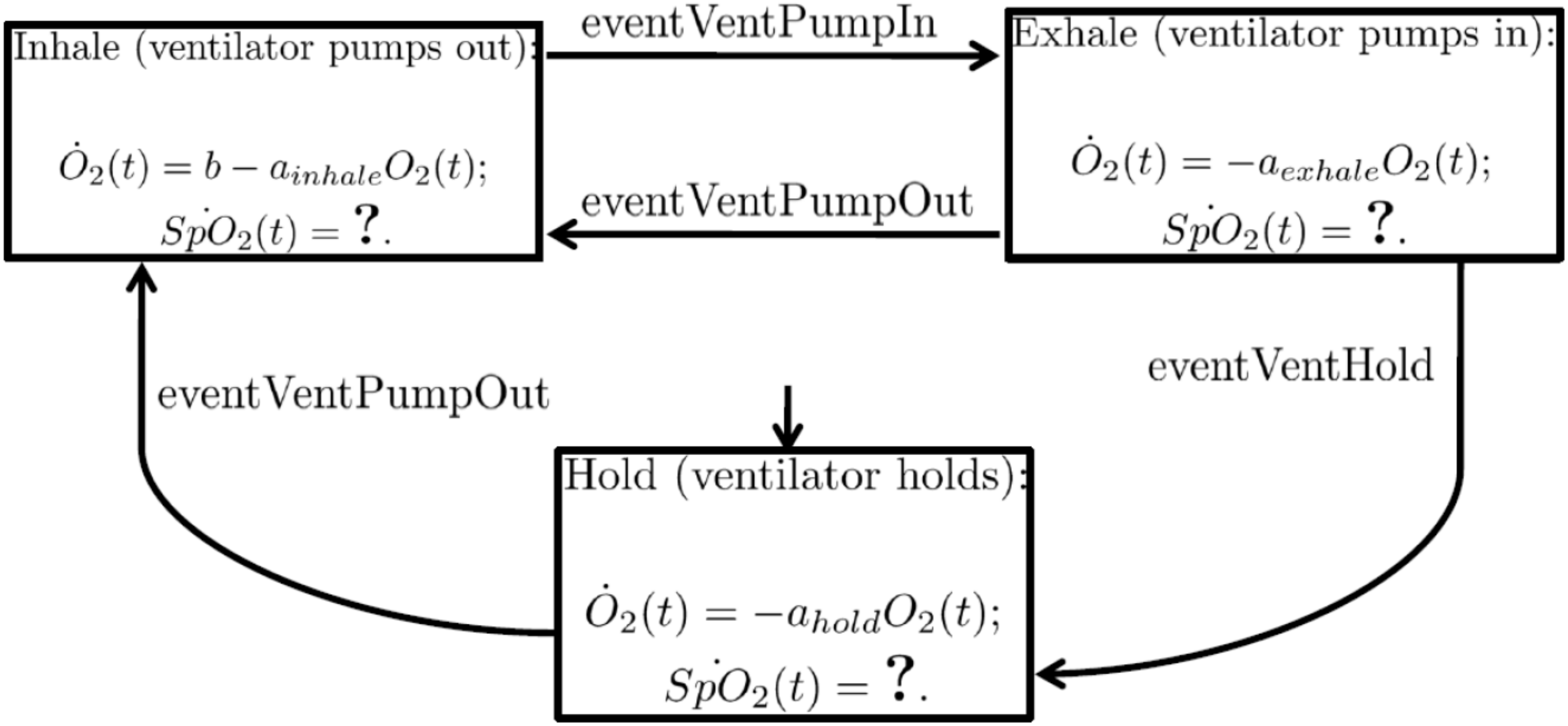
[ ] Event guard (event triggering condition)

:= Variable value update

Take laser tracheotomy offline hybrid systems modeling as an example.



Take laser tracheotomy offline hybrid systems modeling as an example: model SpO_2 offline?



Inhale (ventilator pumps out):
$\dot{O}_2(t) = b - a_{inhale}O_2(t);$
$S\dot{p}O_2(t) = ?.$
eventVentPumpIn
Exhale (ventilator pumps in):
$\dot{O}_2(t) = -a_{exhale}O_2(t);$
$S\dot{p}O_2(t) = ?.$
eventVentPumpOut
eventVentHold
eventVentPumpOut
Hold (ventilator holds):
$\dot{O}_2(t) = -a_{hold}O_2(t);$
$S\dot{p}O_2(t) = ?.$

Content

Demand

Background

Challenge

Solution

Evaluation

Related Work

Online periodical real-time hybrid systems model checking of time-bounded (i.e., short-run) future!

Online periodical real-time hybrid systems model checking of time-bounded (i.e., short-run) future!

Traditional model checking vs. Ours:

Offline ←→ Online Periodical Real-Time

Long-Run Future ←→ Short-Run Future

Online periodical real-time hybrid systems model checking of time-bounded (i.e., short-run) future!

Traditional model checking vs. Ours:

Offline ←→ Online Periodical Real-Time

Long-Run Future ←→ Short-Run Future

Challenge 1: No good offline models for complex biomedical systems of human body.

Online periodical real-time hybrid systems model checking of time-bounded (i.e., short-run) future!

Traditional model checking vs. Ours:

Offline ←→ Online Periodical Real-Time

Long-Run Future ←→ Short-Run Future

Challenge 1: No good offline models for complex biomedical systems of human body.

Most vital signs' online short-run behavior is easy to predict.

Online periodical real-time hybrid systems model checking of time-bounded (i.e., short-run) future!

Traditional model checking vs. Ours:

Offline ←→ Online Periodical Real-Time

Long-Run Future ←→ Short-Run Future

Challenge 1: No good offline models for complex biomedical systems of human body.

Most vital signs' online short-run behavior is easy to predict.

Challenge 2: Verification state space easily explode.

Online periodical real-time hybrid systems model checking of time-bounded (i.e., short-run) future!

Traditional model checking vs. Ours:

Offline ←→ Online Periodical Real-Time

Long-Run Future ←→ Short-Run Future

Challenge 1: No good offline models for complex biomedical systems of human body.

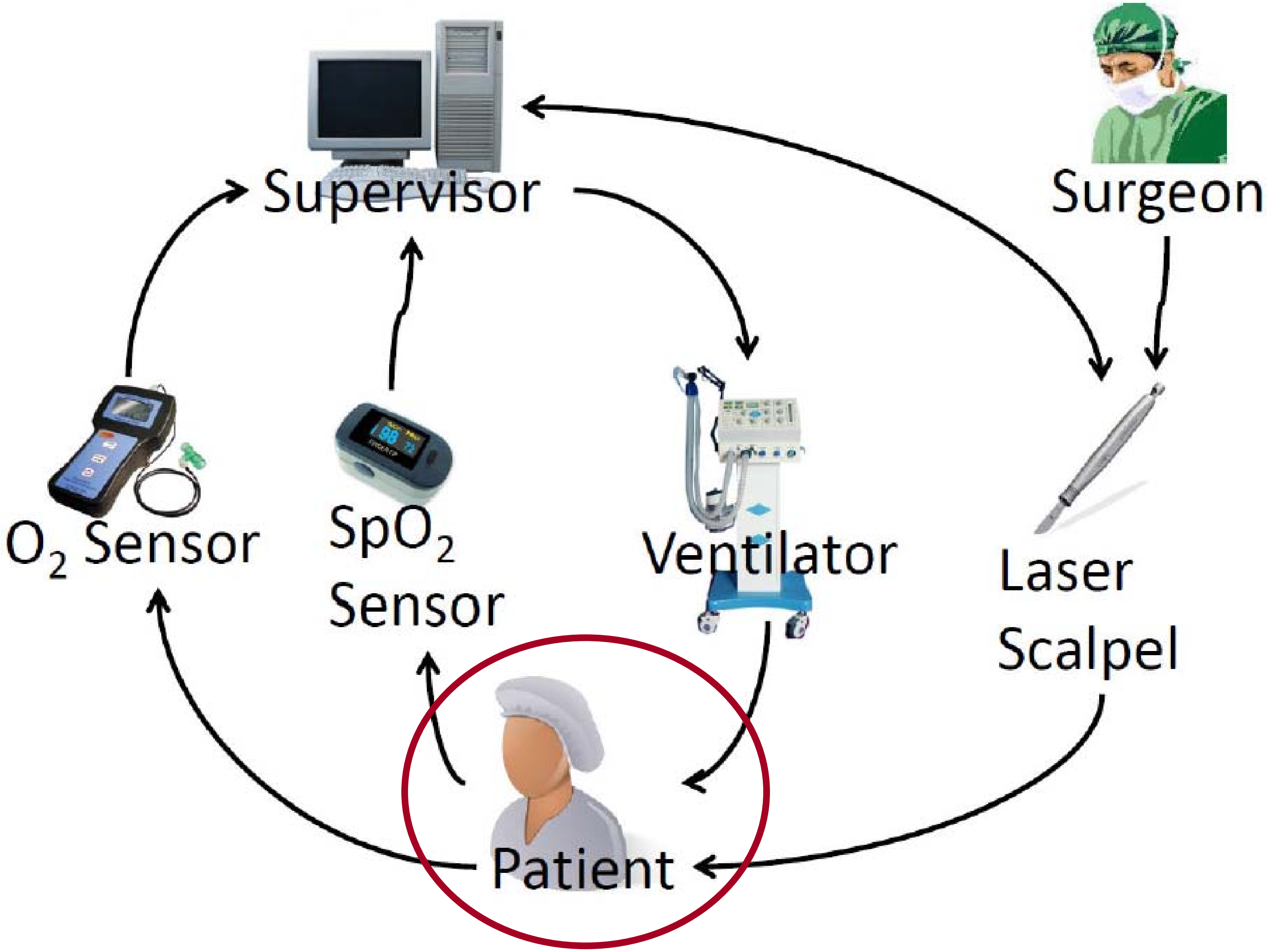
Most vital signs' online short-run behavior is easy to predict.

Challenge 2: Verification state space easily explode.

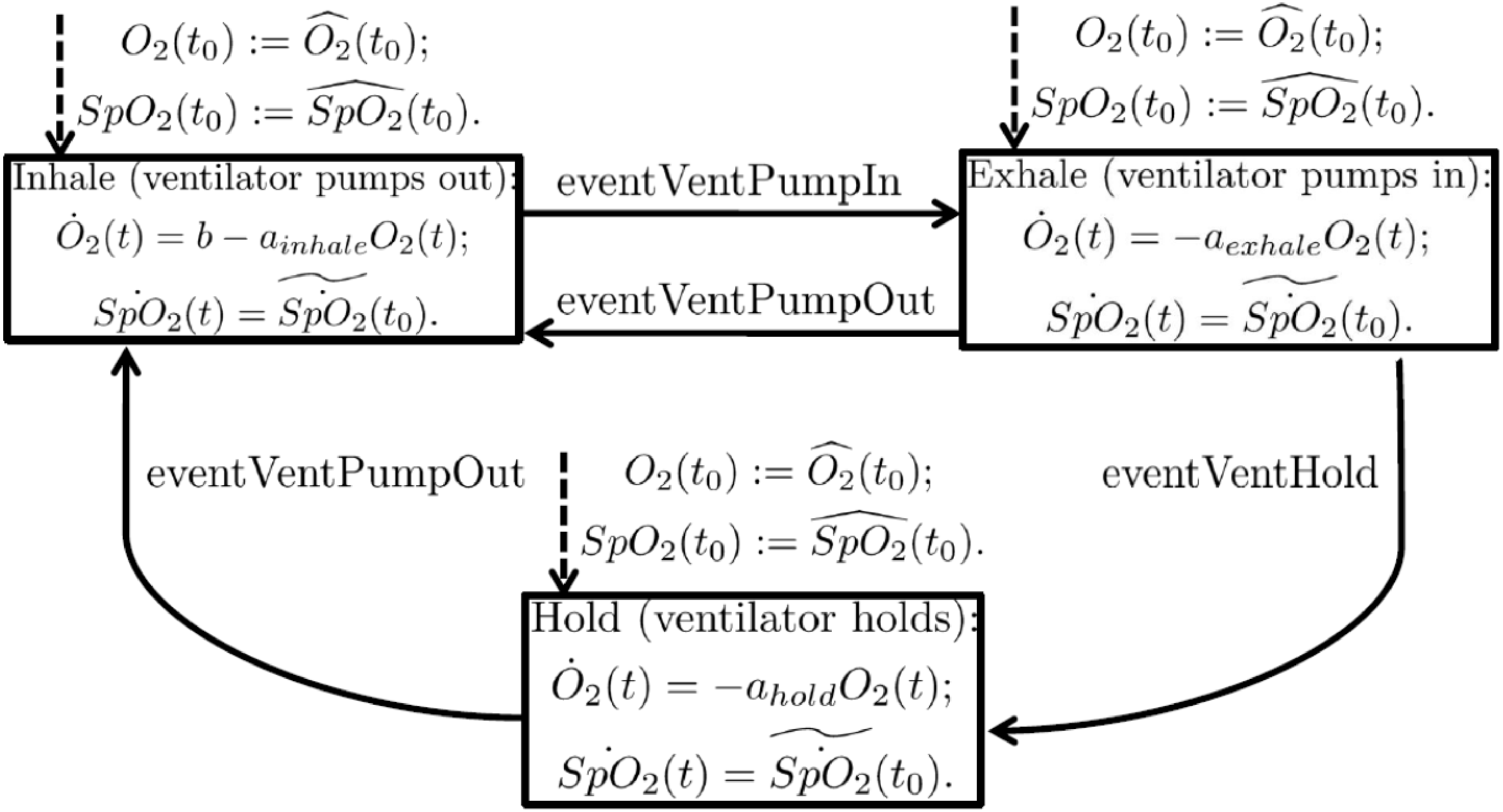
Online → Fixes Many Parameters

Short-Run → Shrink State Space

Let's model the patient again, now online and short-run, with period T.



Let’s model the patient again, now online and short-run, with period T.



$O_2(t_0) := \widehat{O_2}(t_0);$

$SpO_2(t_0) := \widehat{SpO_2}(t_0).$

Inhale (ventilator pumps out):

$\dot{O}_2(t) = b - a_{inhale}O_2(t);$

$\dot{SpO_2}(t) = \widetilde{\dot{SpO_2}}(t_0).$

eventVentPumpIn

eventVentPumpOut

$O_2(t_0) := \widehat{O_2}(t_0);$

$SpO_2(t_0) := \widehat{SpO_2}(t_0).$

Exhale (ventilator pumps in):

$\dot{O}_2(t) = -a_{exhale}O_2(t);$

$\dot{SpO_2}(t) = \widetilde{\dot{SpO_2}}(t_0).$

eventVentPumpOut

$O_2(t_0) := \widehat{O_2}(t_0);$

$SpO_2(t_0) := \widehat{SpO_2}(t_0).$

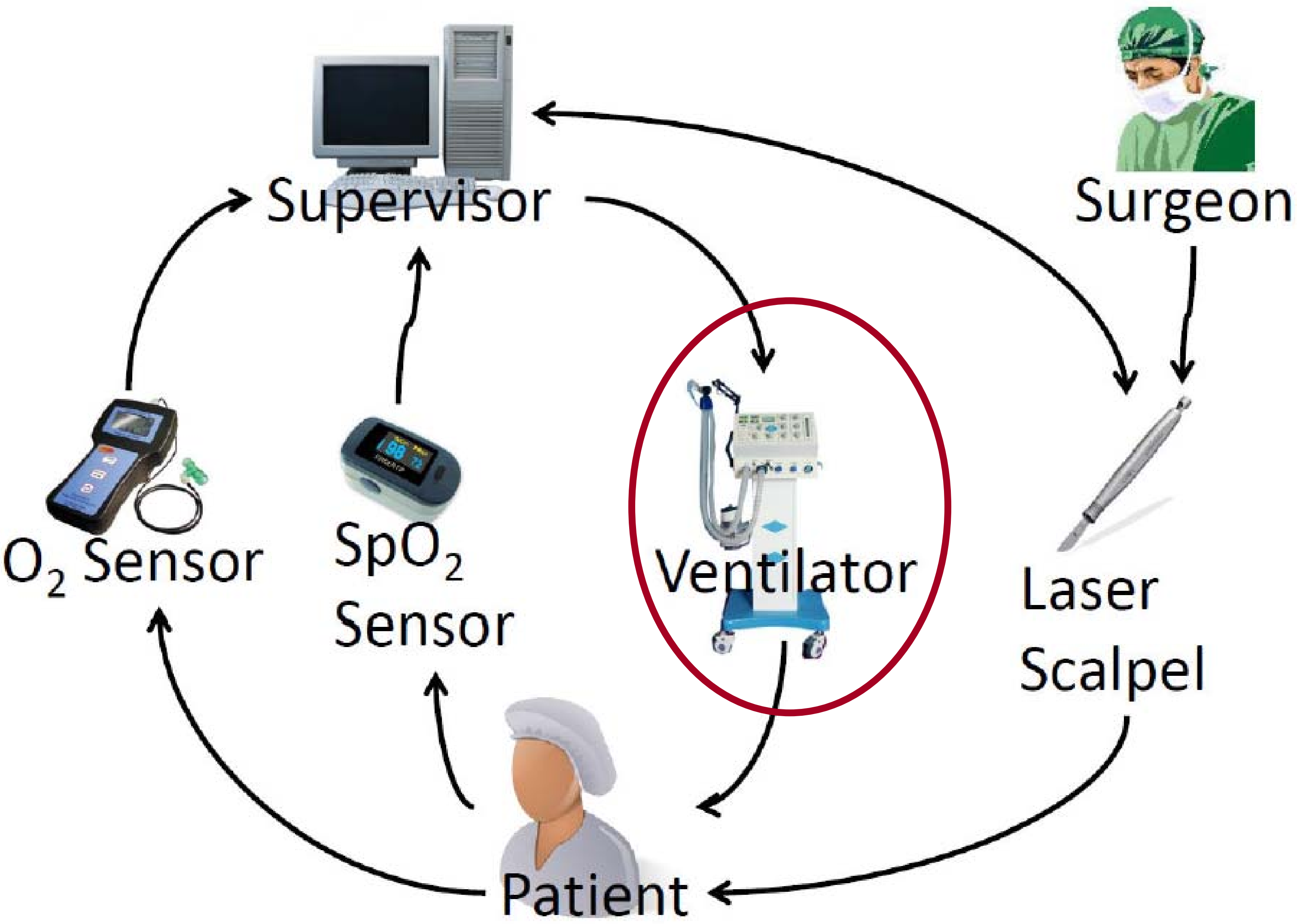
eventVentHold

Hold (ventilator holds):

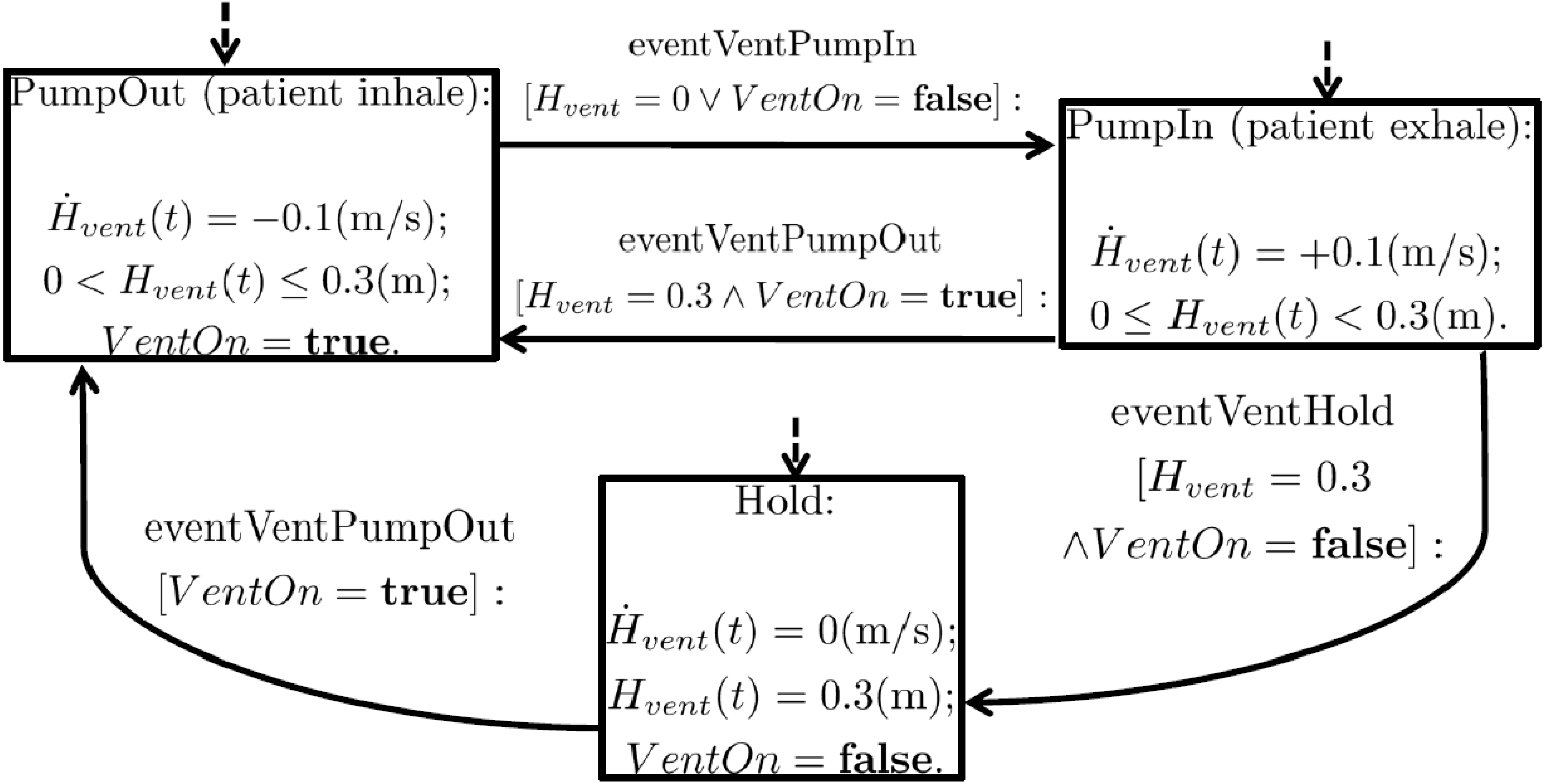
$\dot{O}_2(t) = -a_{hold}O_2(t);$

$\dot{SpO_2}(t) = \widetilde{\dot{SpO_2}}(t_0).$

The online short-run model for ventilator.



The online short-run model for ventilator.



PumpOut (patient inhale):

$\dot{H}_{vent}(t) = -0.1(\mathrm{m/s});$

$0 < H_{vent}(t) \leq 0.3(\mathrm{m});$

$VentOn = \textbf{true}.$

eventVentPumpIn

$[H_{vent} = 0 \vee VentOn = \textbf{false}]:$

PumpIn (patient exhale):

$\dot{H}_{vent}(t) = +0.1(\mathrm{m/s});$

$0 \leq H_{vent}(t) < 0.3(\mathrm{m}).$

eventVentPumpOut

$[H_{vent} = 0.3 \wedge VentOn = \textbf{true}]:$

eventVentHold

$[H_{vent} = 0.3$

$\wedge VentOn = \textbf{false}]:$

Hold:

$\dot{H}_{vent}(t) = 0(\mathrm{m/s});$

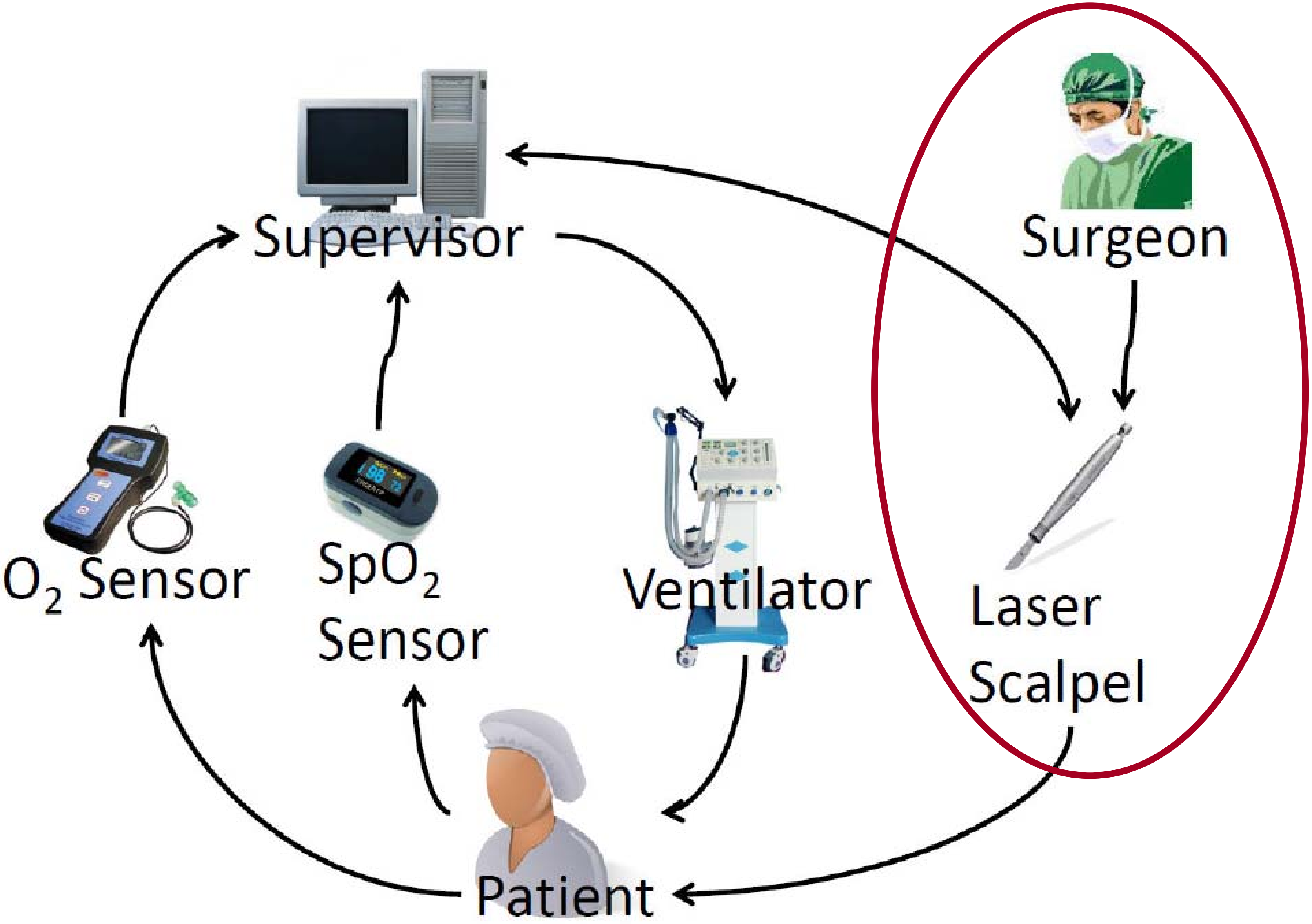
$H_{vent}(t) = 0.3(\mathrm{m});$

$VentOn = \textbf{false}.$

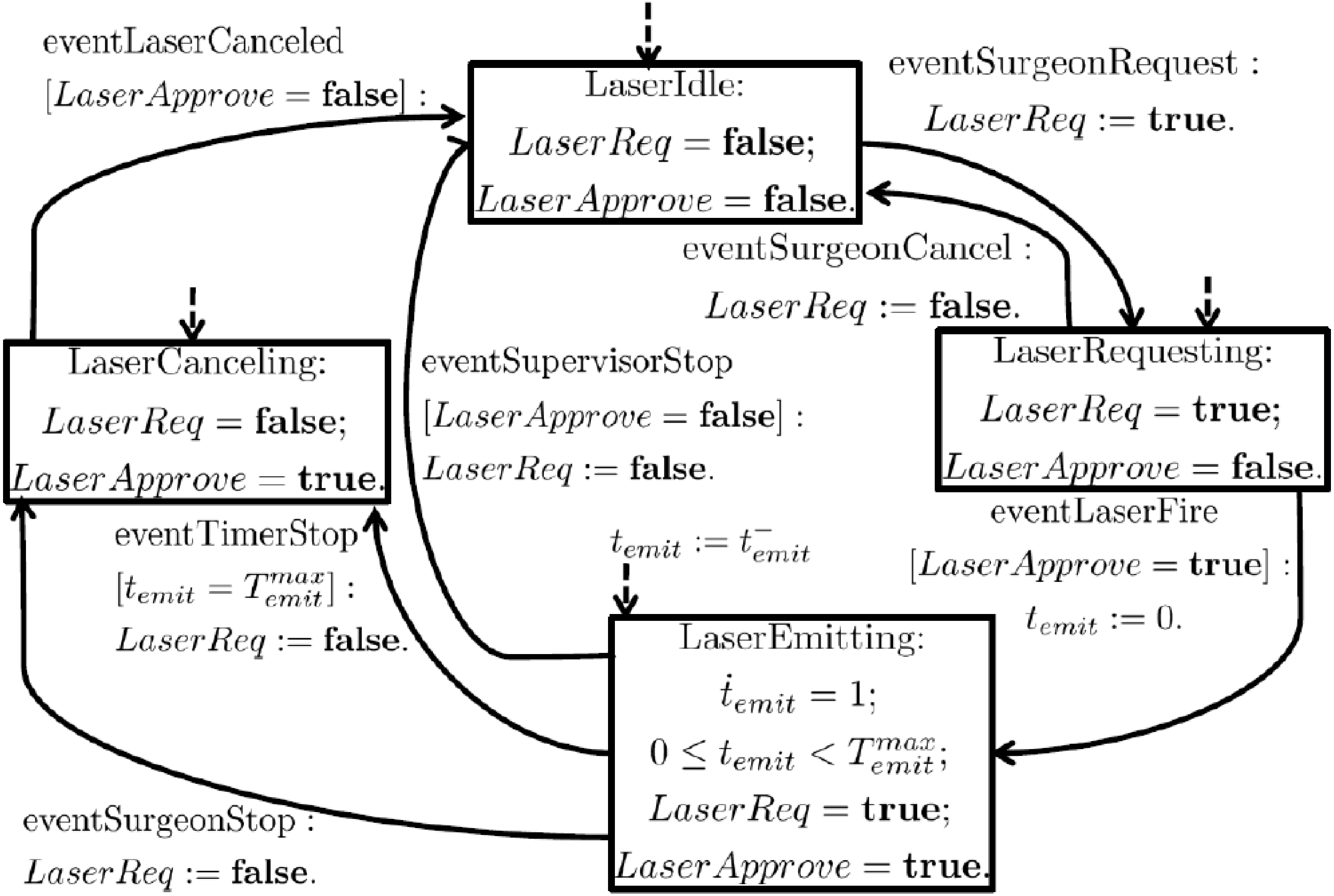
eventVentPumpOut

$[VentOn = \textbf{true}]:$

The online short-run model for laser-scalpel.



The online short-run model for laser-scalpel.



eventLaserCanceled

$[LaserApprove = \textbf{false}]$:

LaserIdle:

$LaserReq = \textbf{false};$

$LaserApprove = \textbf{false}.$

eventSurgeonRequest :

$LaserReq := \textbf{true}.$

eventSurgeonCancel :

$LaserReq := \textbf{false}.$

LaserCanceling:

$LaserReq = \textbf{false};$

$LaserApprove = \textbf{true}.$

eventSupervisorStop

$[LaserApprove = \textbf{false}]$:

$LaserReq := \textbf{false}.$

LaserRequesting:

$LaserReq = \textbf{true};$

$LaserApprove = \textbf{false}.$

eventLaserFire

$[LaserApprove = \textbf{true}]$:

$t_{emit} := 0.$

eventTimerStop

$[t_{emit} = T_{emit}^{max}]$:

$LaserReq := \textbf{false}.$

$t_{emit} := t_{emit}^{-}$

LaserEmitting:

$\dot{t}_{emit} = 1;$

$0 \leq t_{emit} < T_{emit}^{max};$

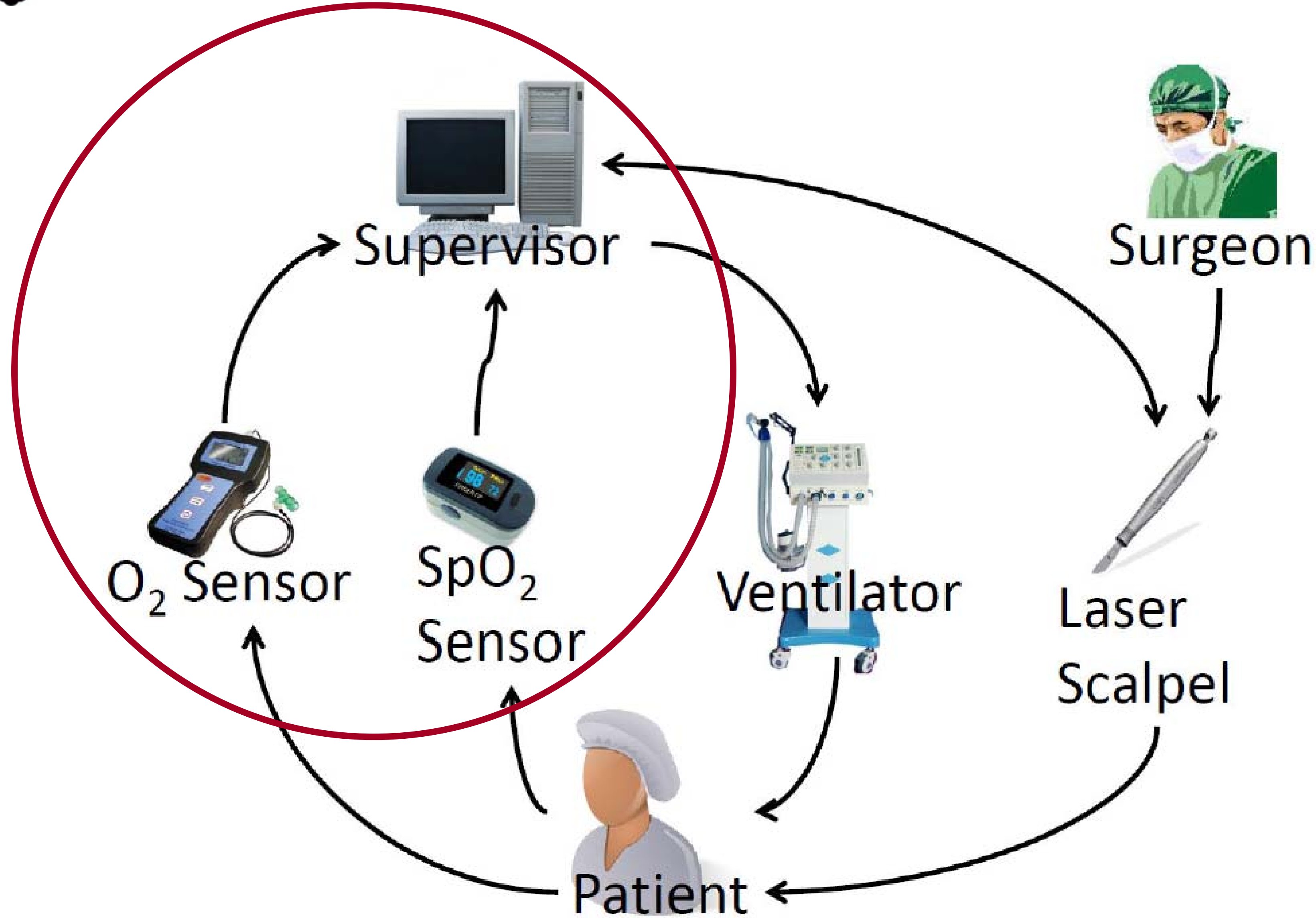
$LaserReq = \textbf{true};$

$LaserApprove = \textbf{true}.$

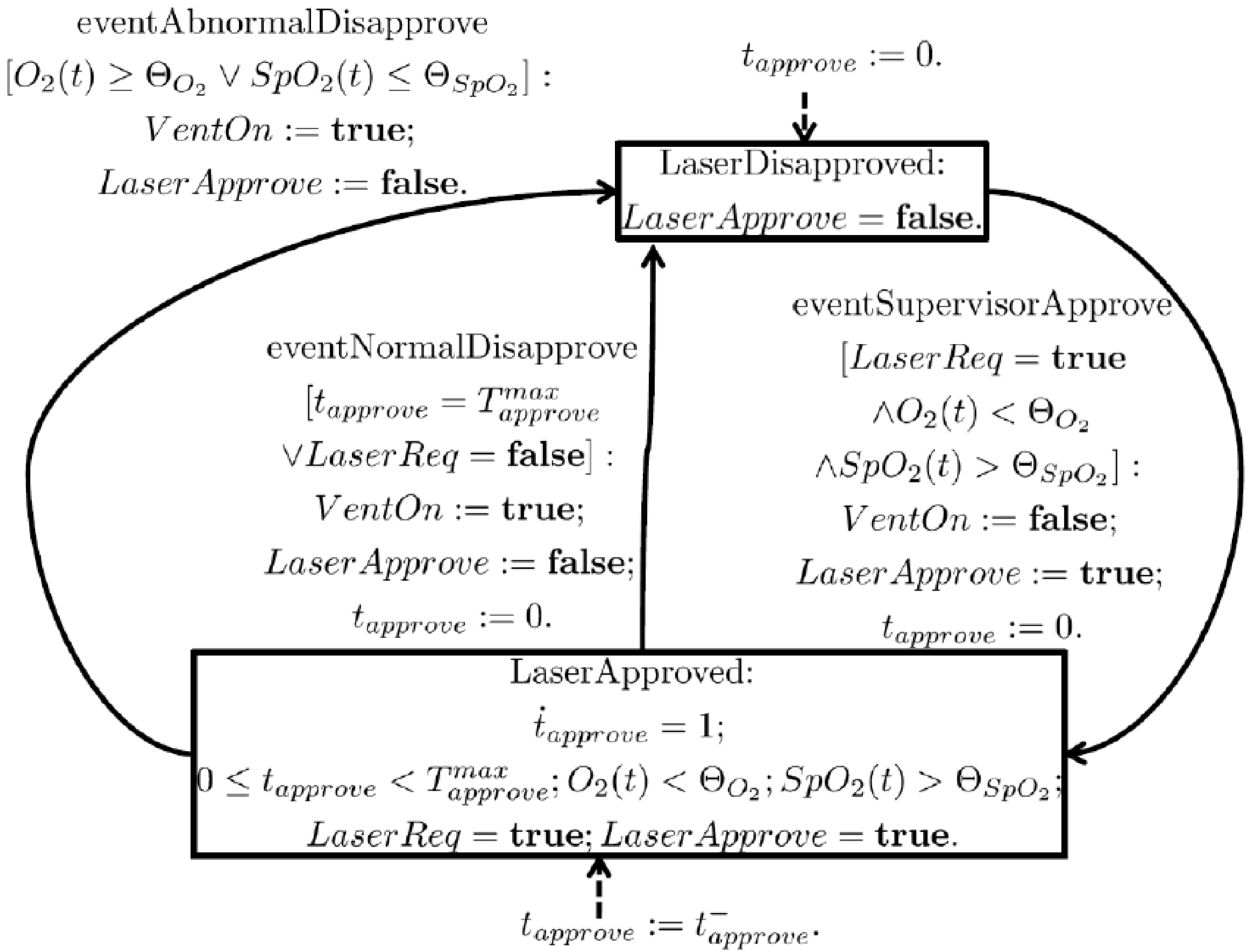
eventSurgeonStop :

$LaserReq := \textbf{false}.$

The online short-run model for supervisor.



The online short-run model for supervisor.



eventAbnormalDisapprove

$[O_2(t) \geq \Theta_{O_2} \vee SpO_2(t) \leq \Theta_{SpO_2}]$:

$VentOn :=$ **true**;

$LaserApprove :=$ **false**.

$t_{approve} := 0.$

LaserDisapproved:

$LaserApprove =$ **false**.

eventNormalDisapprove

$[t_{approve} = T^{max}_{approve}$

$\vee LaserReq =$ **false**$]$:

$VentOn :=$ **true**;

$LaserApprove :=$ **false**;

$t_{approve} := 0.$

eventSupervisorApprove

$[LaserReq =$ **true**

$\wedge O_2(t) < \Theta_{O_2}$

$\wedge SpO_2(t) > \Theta_{SpO_2}]$:

$VentOn :=$ **false**;

$LaserApprove :=$ **true**;

$t_{approve} := 0.$

LaserApproved:

$\dot{t}_{approve} = 1;$

$0 \leq t_{approve} < T^{max}_{approve}; O_2(t) < \Theta_{O_2}; SpO_2(t) > \Theta_{SpO_2};$

$LaserReq =$ **true**; $LaserApprove =$ **true**.

$t_{approve} := t^{-}_{approve}.$

Question: Can the hybrid systems model checking finish (terminate) within period T?

Question: Can the hybrid systems model checking finish (terminate) within period T?

Hybrid Systems Model Checking → undecidable

Question: Can the hybrid systems model checking finish (terminate) within period T ?

Hybrid Systems Model Checking → undecidable

Linear Hybrid Automaton (LHA) model checking → undecidable

Question: Can the hybrid systems model checking finish (terminate) within period T ?

Hybrid Systems Model Checking → undecidable

Linear Hybrid Automaton (LHA) model checking → undecidable

Simple Time-Bounded (STB) LHA model checking →

Question: Can the hybrid systems model checking finish (terminate) within period T?

Hybrid Systems Model Checking → undecidable

Linear Hybrid Automaton (LHA) model checking → undecidable

Simple Time-Bounded (STB) LHA model checking →

We proved a well-known reachability calculation procedure terminates within polynomial time.

Question: Can the hybrid systems model checking finish (terminate) within period T ?

Hybrid Systems Model Checking → undecidable

Linear Hybrid Automaton (LHA) model checking → undecidable

Simple Time-Bounded (STB) LHA model checking →

We proved a well-known reachability calculation procedure terminates within polynomial time.

STB LHA is powerful enough to describe laser tracheotomy scenario, a representative MDPnP application.

Content

Demand

Background

Challenge

Solution

Evaluation

Related Work

Evaluation Setup

Evaluation Setup

Emulated Oxymeter and O2 sensor using NIH PhysioNet real-world patient vital sign traces.

Evaluation Setup

Emulated Oxymeter and O2 sensor using NIH PhysioNet real-world patient vital sign traces.

Sampling/Model-Checking Period: $T = 3$ second.

Evaluation Setup

Emulated Oxymeter and O2 sensor using NIH PhysioNet real-world patient vital sign traces.

Sampling/Model-Checking Period: $T = 3$ second.

Hand written online model generator + PHAVer hybrid systems model checker

Evaluation Setup

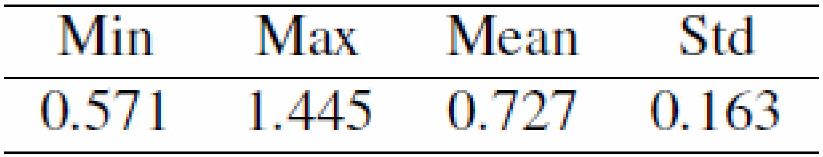
Emulated Oxymeter and O2 sensor using NIH PhysioNet real-world patient vital sign traces.

Sampling/Model-Checking Period: $T = 3$ second.

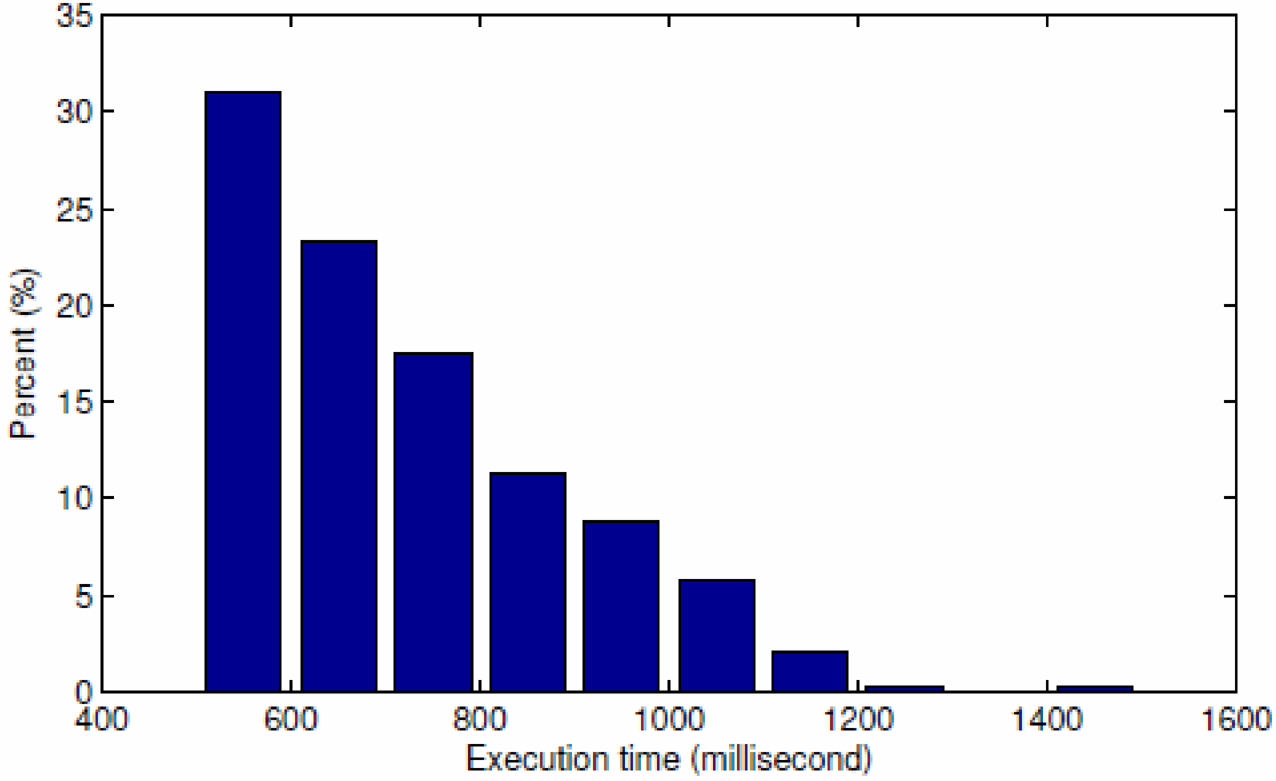
Hand written online model generator + PHAVer hybrid systems model checker

Lenovo Thinkpad X201 + Intel Core i5
+ 2.9G Mem + 32-bit Ubuntu 10.10

Statistics of execution (modeling + checking) time cost: real-time feasible (with pipelining).

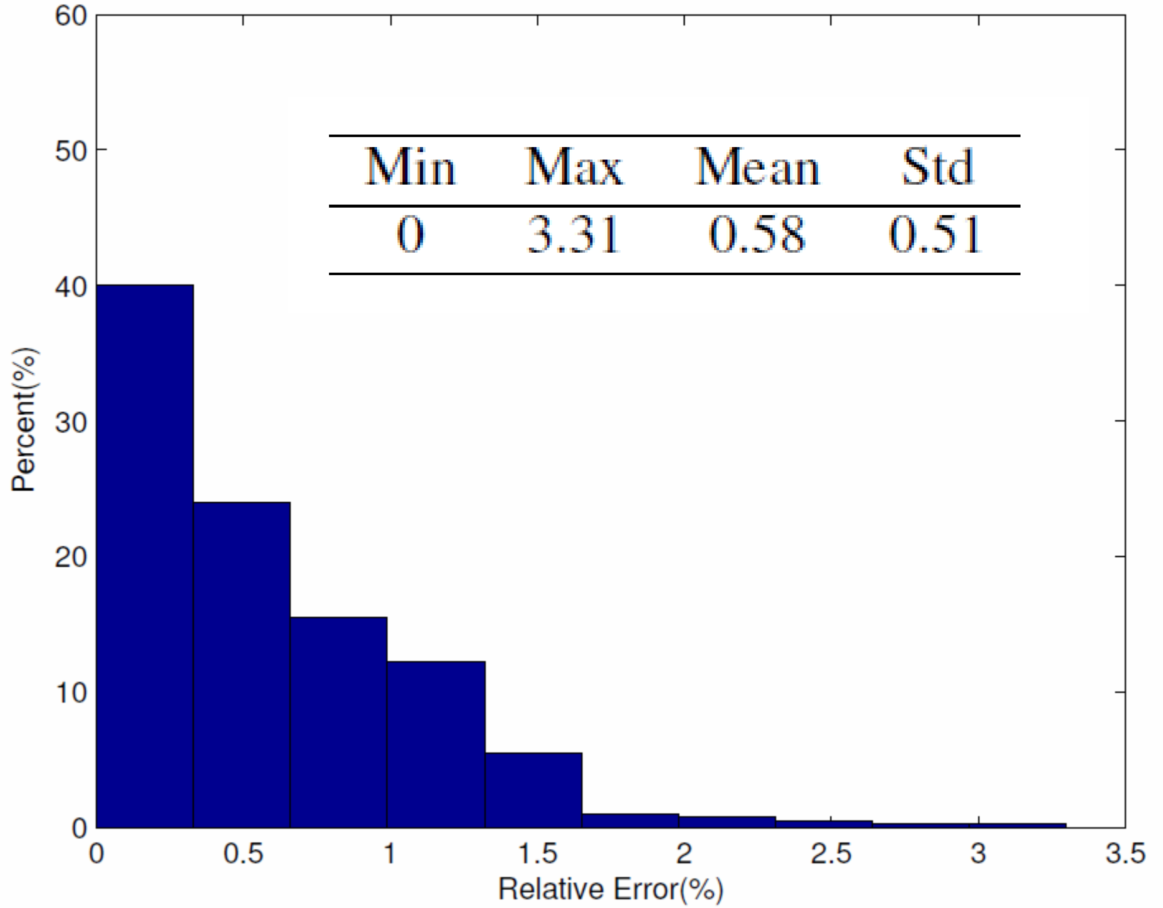


Min	Max	Mean	Std
0.571	1.445	0.727	0.163



Statistics of online SpO_2 prediction accuracy

$$ERR_{SpO_2}(t_0+T) = \frac{|\widehat{SpO}_2(t_0+T) - \widetilde{SpO}_2(t_0+T)|}{\widehat{SpO}_2(t_0+T)}$$



Min	Max	Mean	Std
0	3.31	0.58	0.51

Content

Demand

Background

Challenge

Solution

Evaluation

Related Work

Related Work

Runtime Verification [finkbeiner02]

Online discrete systems model checking [qi09][easwaran06]

Other hybrid systems model checkers [robby03][bartocci08]

Thank You!

References

[bartocci08] E. Bartocci, F. Corradini, E. Entcheva, R. Grosu, and S. A. Smolka, Cellexcite: An efficient simulation environment for excitable cells. BMC Bioinformatics, 9(2):1-13, Mar. 2008.

[easwaran06] Arvind Easwaran, Sampath Kannan, Oleg Sokolsky: Steering of Discrete Event Systems: Control Theory Approach. Workshop on Runtime Verification 2006.

[finkbeiner02] B. Finkbeiner, S. Sankaranarayanan, and H. Sipma, Collecting statistics over runtime executions. ENTCS, 70:4, 2002

[qi09] Z. Qi, A. Liang, H. Guan, M. Wu, and Z. Zhang, A hybrid model checking and runtime monitoring method for c++ web services. Proc. of the Fifth International Joint Conference on INC, IMS and IDC, 2009.

[robby03] Robby, M. B. Dwyer, and J. Hatcliff. Bogor: An extensible and highly-modular software model checking framework. Proc. of the 9th European Software Engineering Conference (ESEC/FSE-11), 2003.

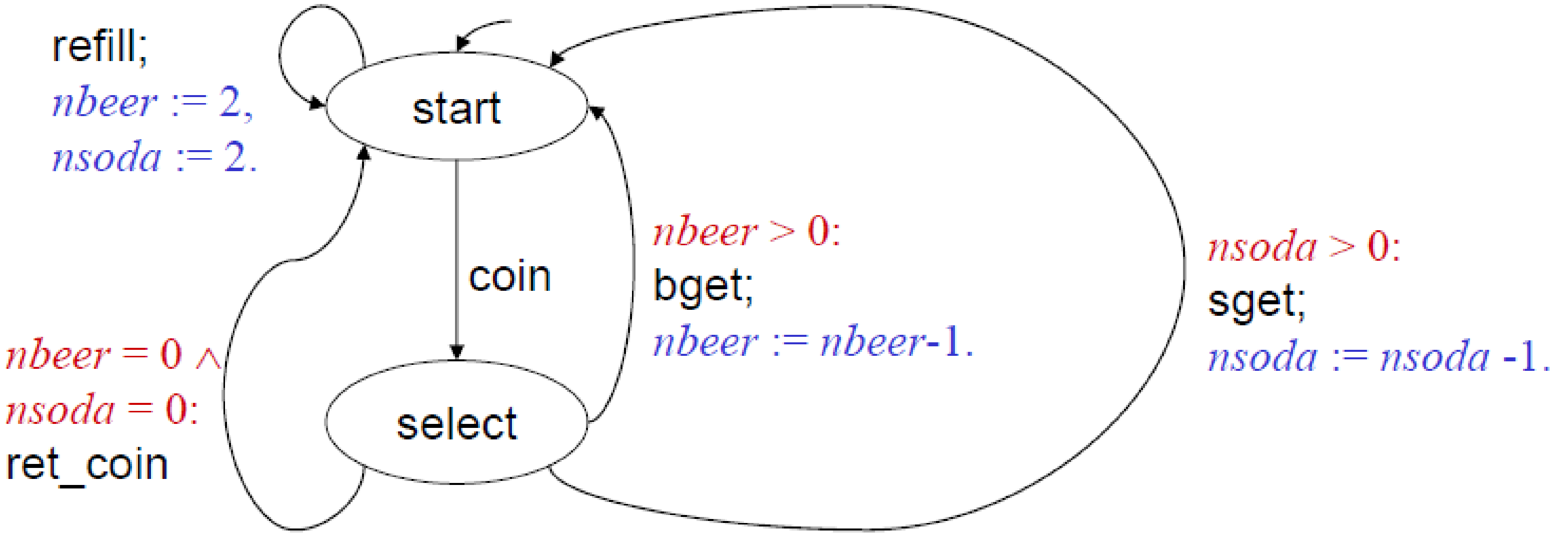
Backup

A key tool for traditional (computer systems) verification is model checking.

$$PG = (Loc, Act, Effect, :\rightarrow, Loc_0, g_0)$$

Set of Locations, e.g., {start, select}

Set of Actions, e.g., {bget, sget, coin, ret_coin, refill}



$$Var = \{nbeer, nsoda\}, \text{domain}(nbeer) = \{0, 1, 2\}, \text{domain}(nsoda) = \{0, 1, 2\}$$

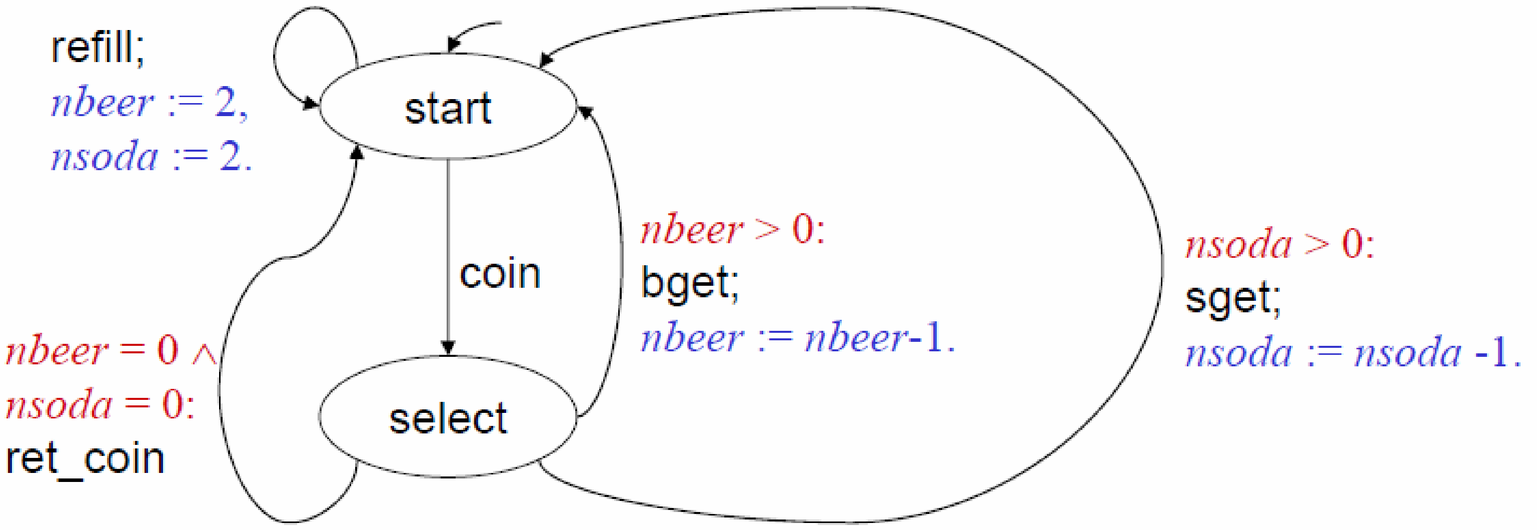
A key tool for traditional (computer systems) verification is model checking.

$$PG = (Loc, Act, Effect, :\rightarrow, Loc_0, g_0)$$

Effect Function : $Act \times Eval(Var) \mapsto Eval(Var)$, e.g.,

$Effect(\text{coin}, \eta) = \eta$, $Effect(\text{ret\_coin}, \eta) = \eta$, $Effect(\text{sget}, \eta) = \eta[nsoda := nsoda - 1]$,

$Effect(\text{bget}, \eta) = \eta[nbeer := nbeer - 1]$, $Effect(\text{refill}, \eta) = [nsoda := 2, nbeer := 2]$.



$Var = \{nbeer, nsoda\}$, domain($nbeer$) = $\{0, 1, 2\}$, domain($nsoda$) = $\{0, 1, 2\}$

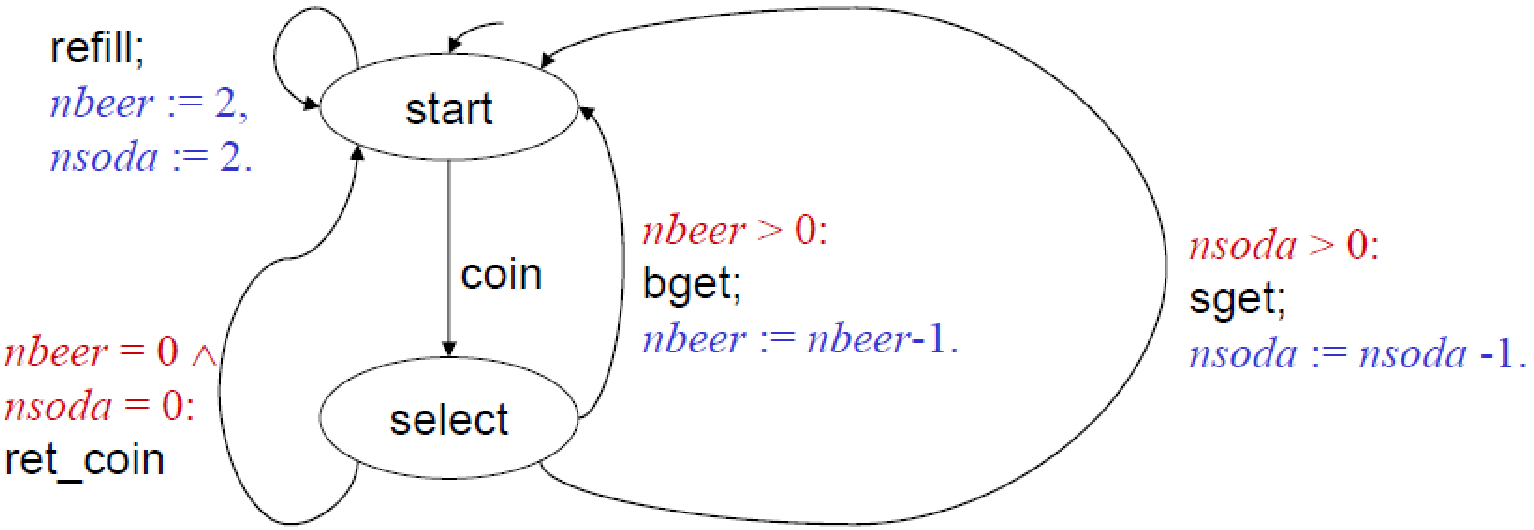
A key tool for traditional (computer systems) verification is model checking.

$$PG = (Loc, Act, Effect, :\rightarrow, Loc_0, g_0)$$

Conditional Transition Relation : $\subseteq Loc \times Cond(Var) \times Act \times Loc$.

Often use shortcut $l : \xrightarrow{g:\alpha} l'$ instead of (l, g, α, l');

in cases where $g = true$, use $l : \xrightarrow{\alpha} l'$.



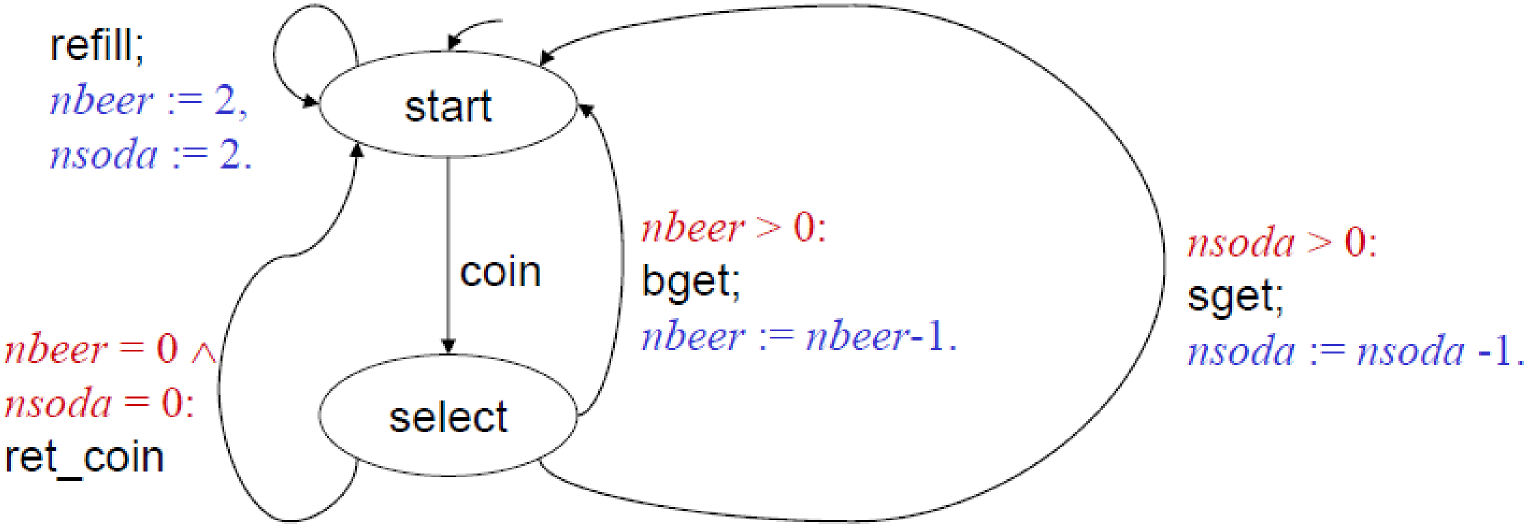
$Var = \{nbeer, nsoda\}$, domain($nbeer$) = {0, 1, 2}, domain($nsoda$) = {0, 1, 2}

A key tool for traditional (computer systems) verification is model checking.

$$PG = (Loc, Act, Effect, :\rightarrow, Loc_0, g_0)$$

$\subseteq Loc$, Set of Initial Locations, e.g., {start}

Initial Condition



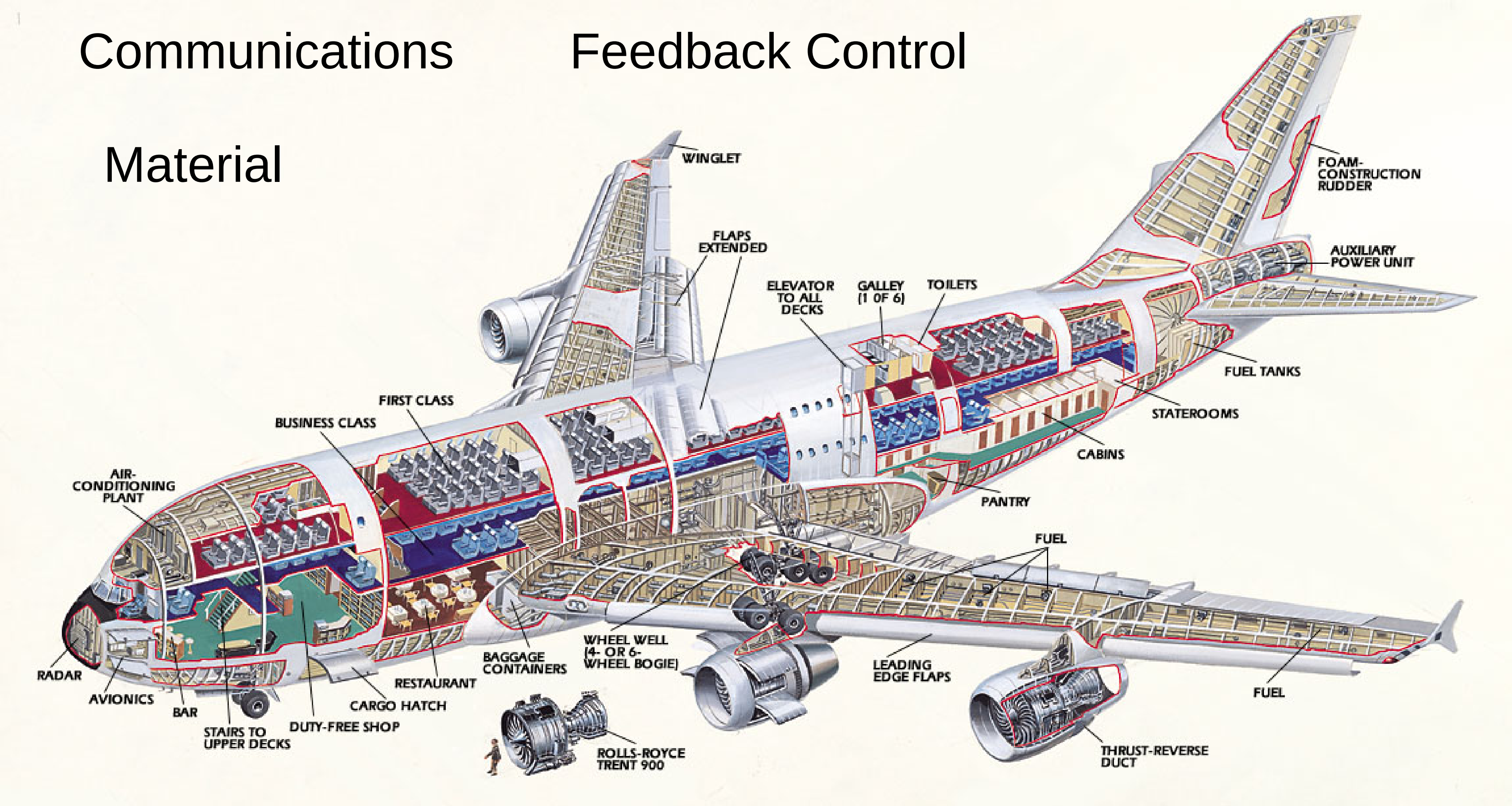
$Var = \{nbeer, nsoda\}$, $\mathrm{domain}(nbeer) = \{0, 1, 2\}$, $\mathrm{domain}(nsoda) = \{0, 1, 2\}$

MDPnP is not just computer systems, it is a hybrid of computer & other systems, i.e., CPS.

Computer Mechanics Aerodynamics

Communications Feedback Control

Material



WINGLET
FOAM-CONSTRUCTION RUDDER
FLAPS EXTENDED
AUXILIARY POWER UNIT
ELEVATOR TO ALL DECKS
GALLEY (1 OF 6)
TOILETS
FUEL TANKS
FIRST CLASS
STATEROOMS
BUSINESS CLASS
CABINS
AIR-CONDITIONING PLANT
PANTRY
FUEL
WHEEL WELL (4- OR 6-WHEEL BOGIE)
BAGGAGE CONTAINERS
LEADING EDGE FLAPS
RADAR
RESTAURANT
FUEL
AVIONICS
CARGO HATCH
BAR
DUTY-FREE SHOP
STAIRS TO UPPER DECKS
ROLLS-ROYCE TRENT 900
THRUST-REVERSE DUCT