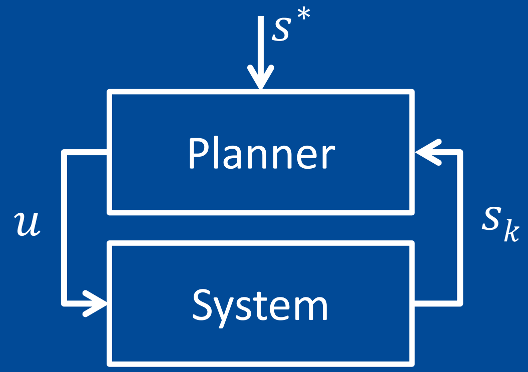
A Planning Based Framework for Controlling Hybrid Systems



Albert-Ludwigs-Universität Freiburg

Johannes Löhr

Patrick Eyerich

Thomas Keller

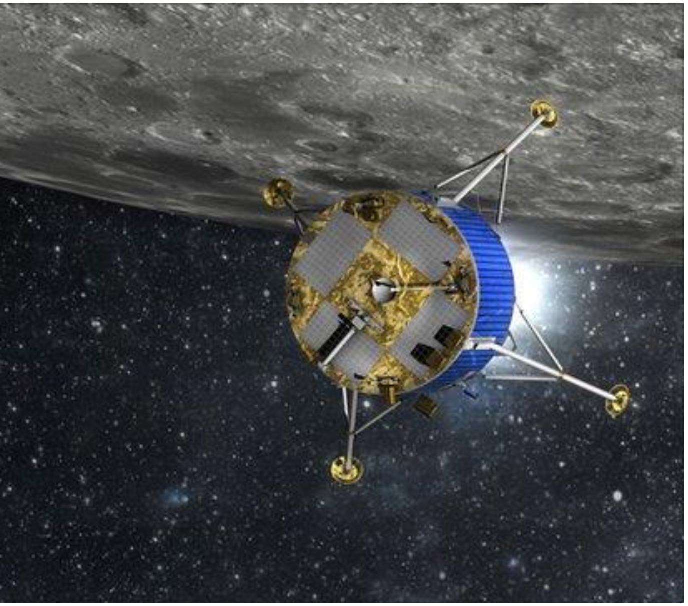
Bernhard Nebel

{loehr, eyerich, tkeller, nebel}@informatik.uni-freiburg.de

Outline

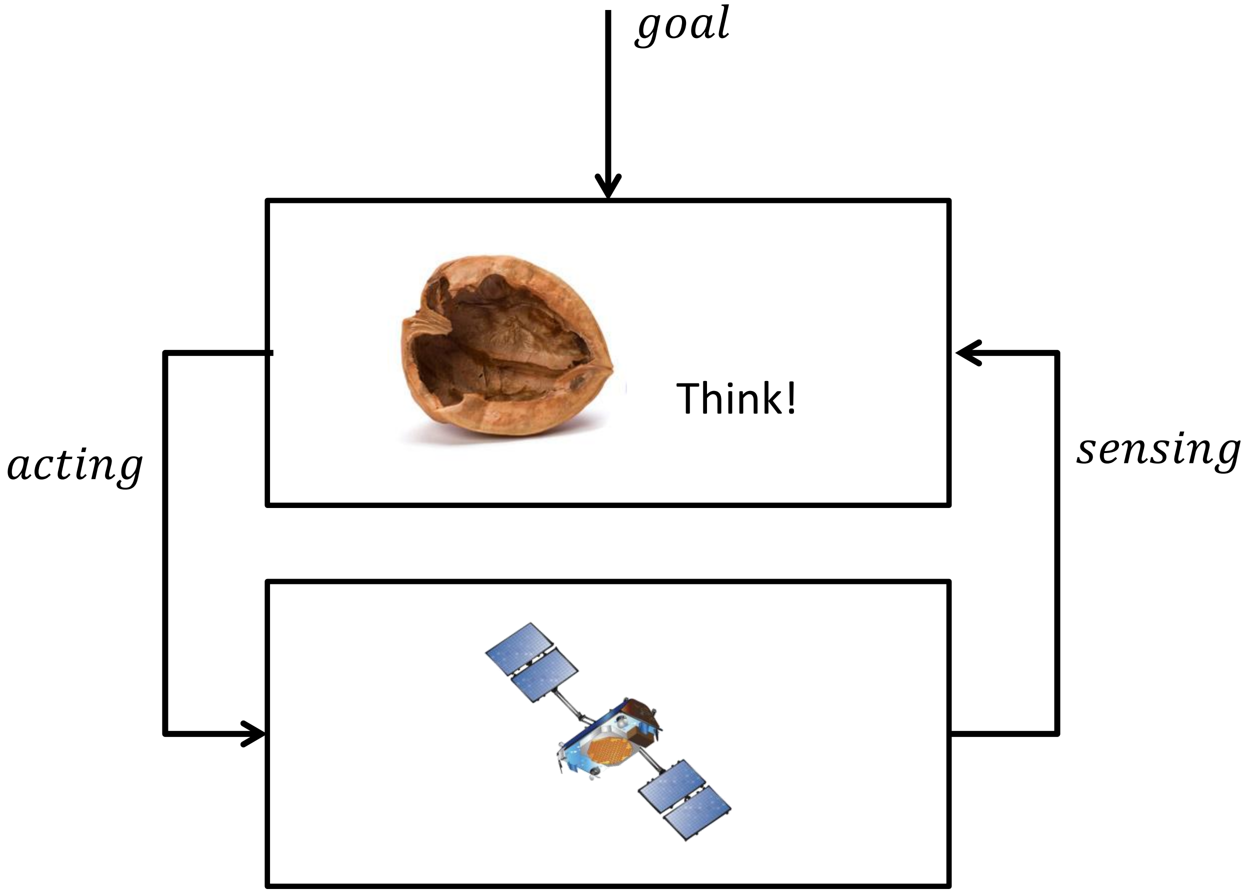
- Motivation
- From Continuous Dynamics…
- … to a Domain Model
- Domain Predictive Control
- Exemplary Simulation
- Discussion
- Outlook

Motivation



Lunar Lander ExoMars

Motivation



Motivation

- Key Aspects

Hybrid Systems
Continuous Dynamics
Boolean State Variables

Autonomy
Reduction of Computational Effort
Quick Decision Generation

Exogenous Events
Obstacles
Reactivity

Outline

- Motivation
- From Continuous Dynamics…
- … to a Domain Model
- Domain Predictive Control
- Exemplary Simulations
- Discussion
- Outlook

From a Hybrid System…

Hybrid System

$$\dot{\boldsymbol{x}}_n(t) = A(\boldsymbol{x}_l)\,\boldsymbol{x}_n(t) + B(\boldsymbol{x}_l)\,\boldsymbol{u}\,(t)$$

Planning Task

Find $\boldsymbol{u}(t),\ t \in [t_a, t_b]$ such that:

$$\mathbf{x}_n(t_a) \longrightarrow \mathbf{x}_n(t_b)$$

Initial State → Desired State

…to a Planning Action…

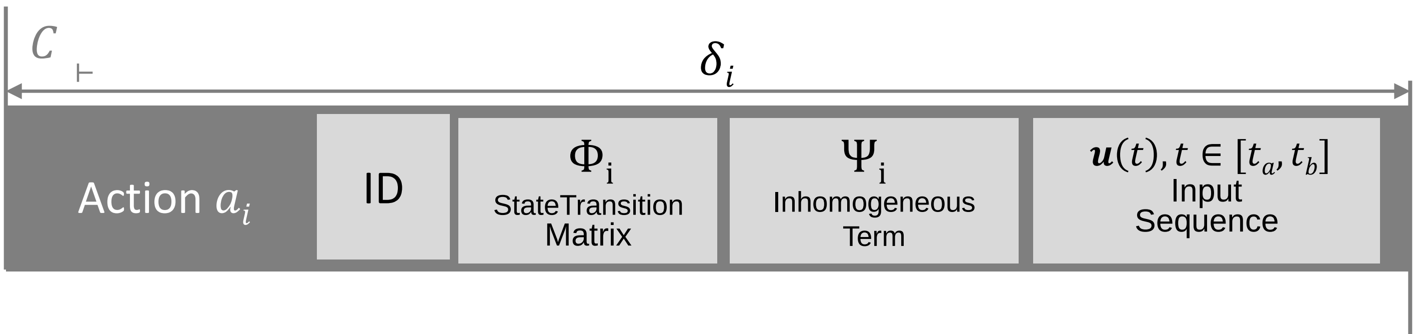
1. Anticipate some input fragments $\boldsymbol{u}_i(t), t \in [0, \delta_i]$

2. Solve the differential equations (preprocessing step)

$$\Phi = e^{A(\boldsymbol{x}_l)\,\delta_i}$$

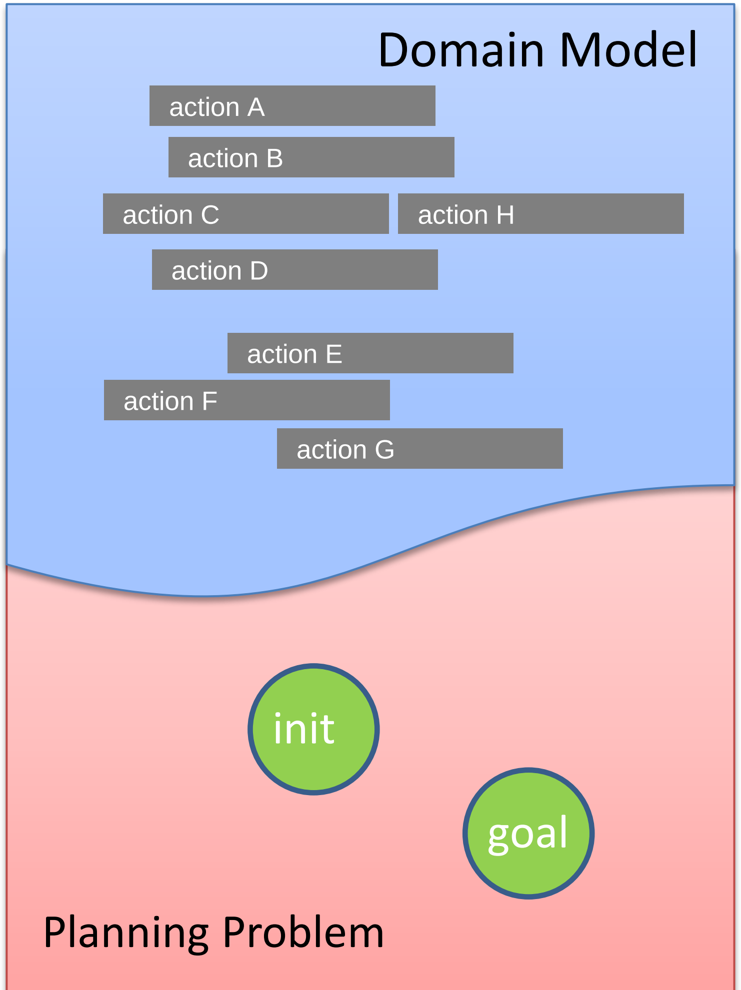
$$\Psi = \int_{t_0}^{t_0+\delta_i} e^{A(x_l)\cdot(t_0+\delta_i-\tau)}\, B(\boldsymbol{x}_l)\, \boldsymbol{u}(\tau)\, d\tau$$

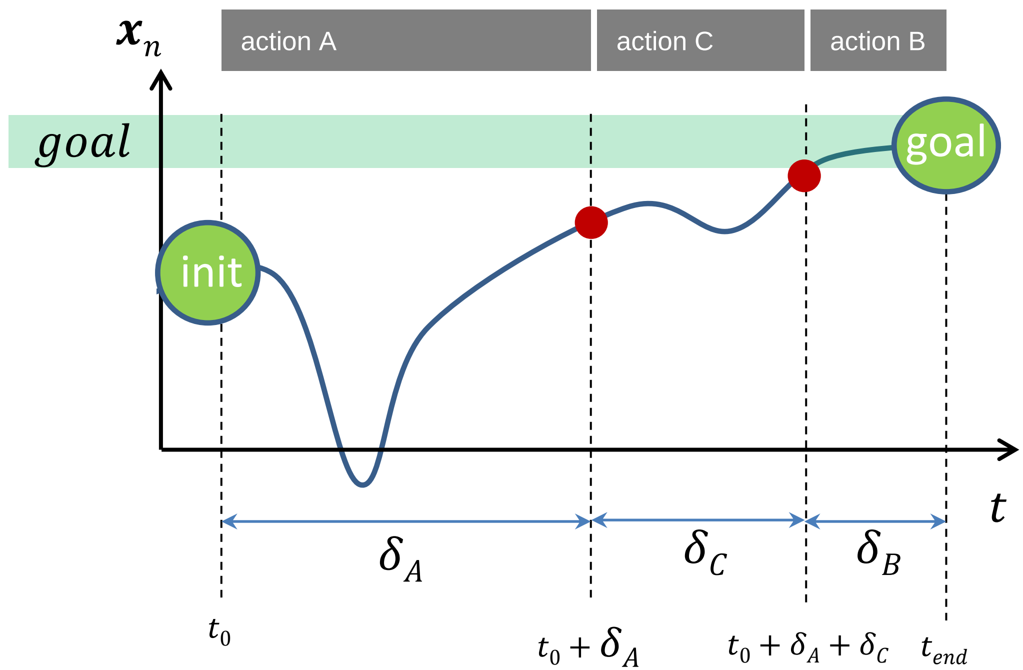
3. Generate planning action



$$E_{\dashv}: \quad \boldsymbol{x}_n(t_0 + \delta_i) = \Phi_i\, \boldsymbol{x}_n(t_0) + \Psi_i$$

… to the Domain Model





Outline

- Motivation
- From Continuous Dynamics...
- ... to a Domain Model
- **Domain Predictive Control**
- Exemplary Simulations
- Discussion
- Outlook

Remember the Key Aspects

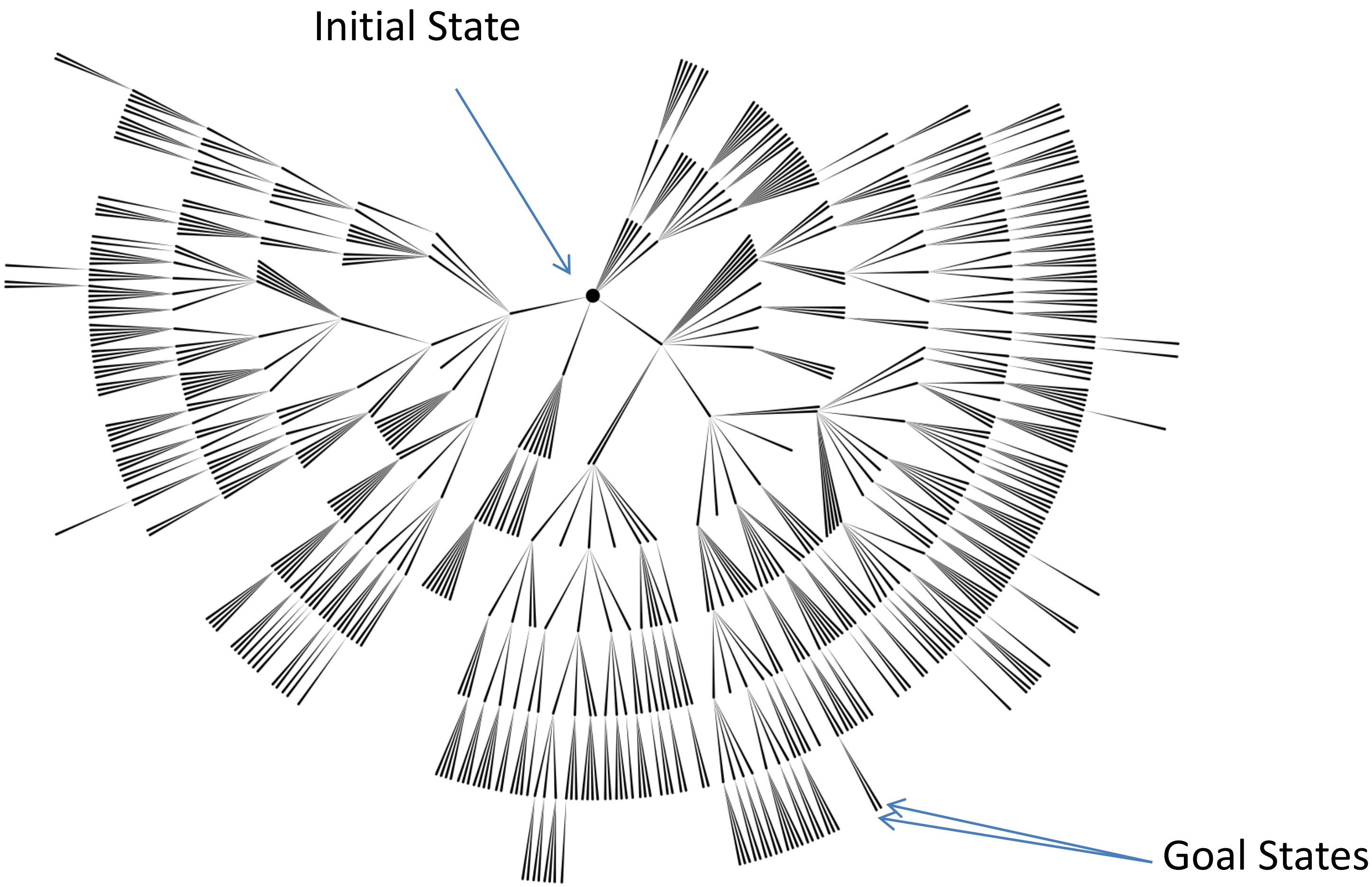
- Key Aspects

Hybrid Systems
Continuous Dynamics
Boolean State Variables

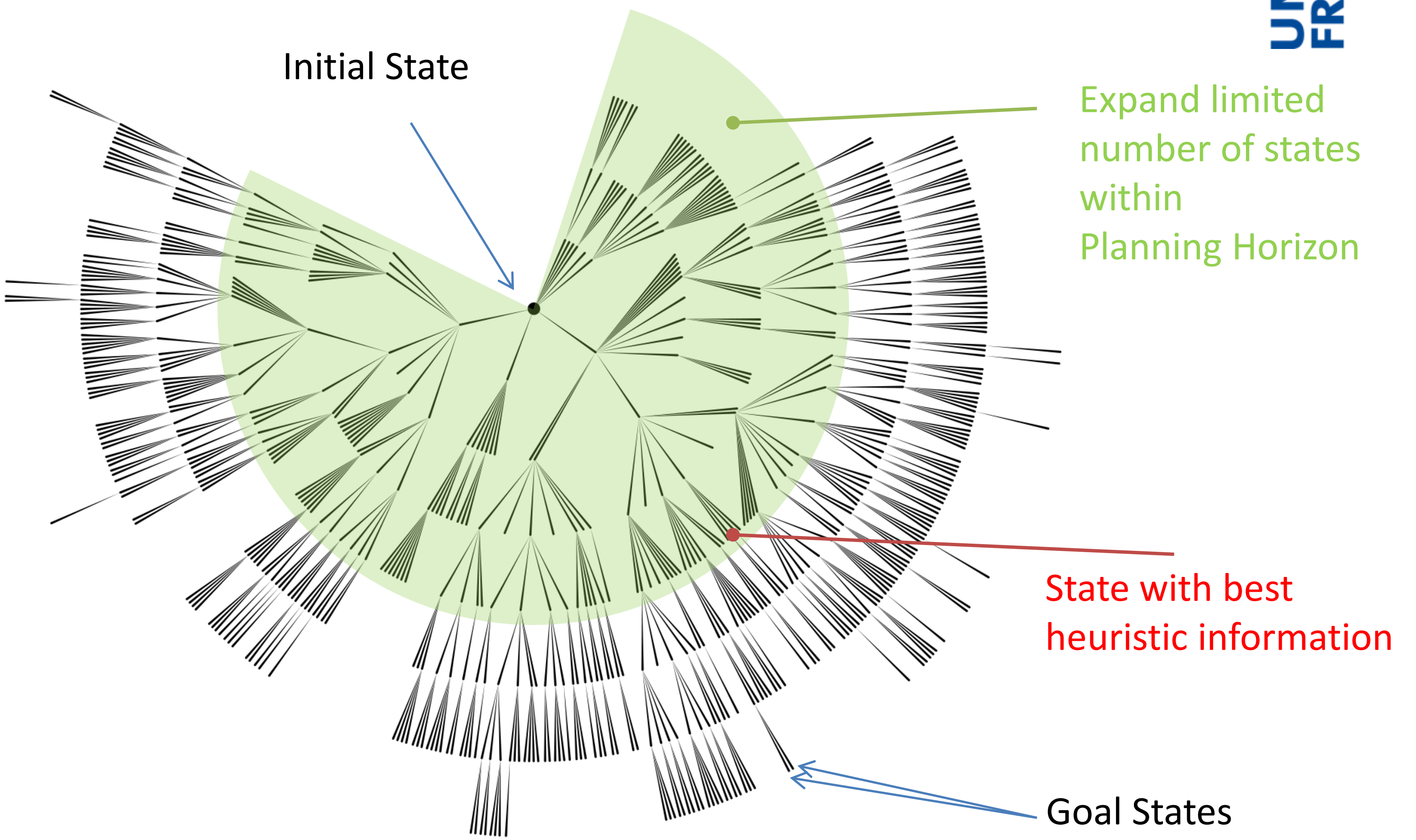
Autonomy
Reduction of Computational Effort
Quick Decision Generation

Exogenous Events
Obstacles
Reactivity

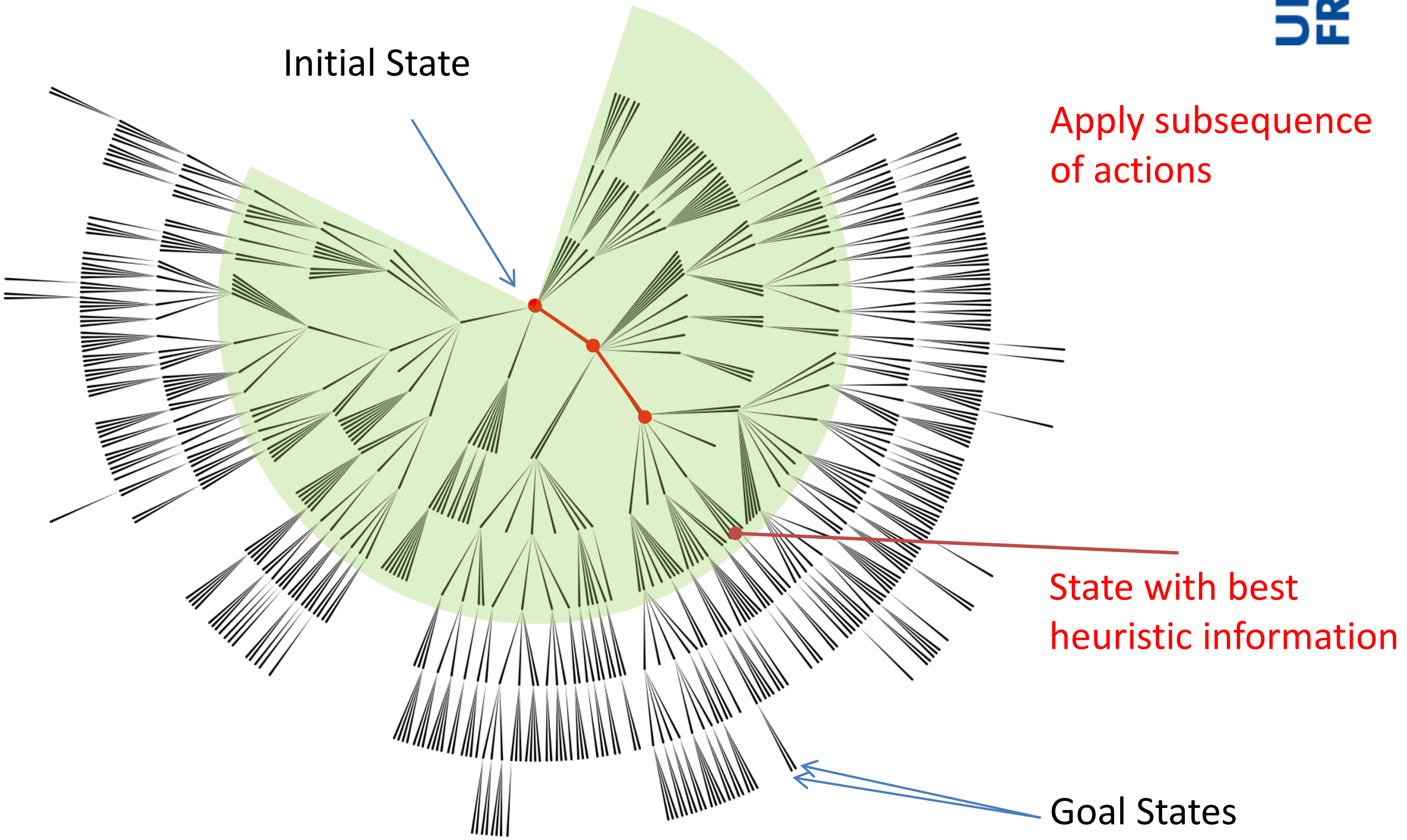
Domain Predictive Control



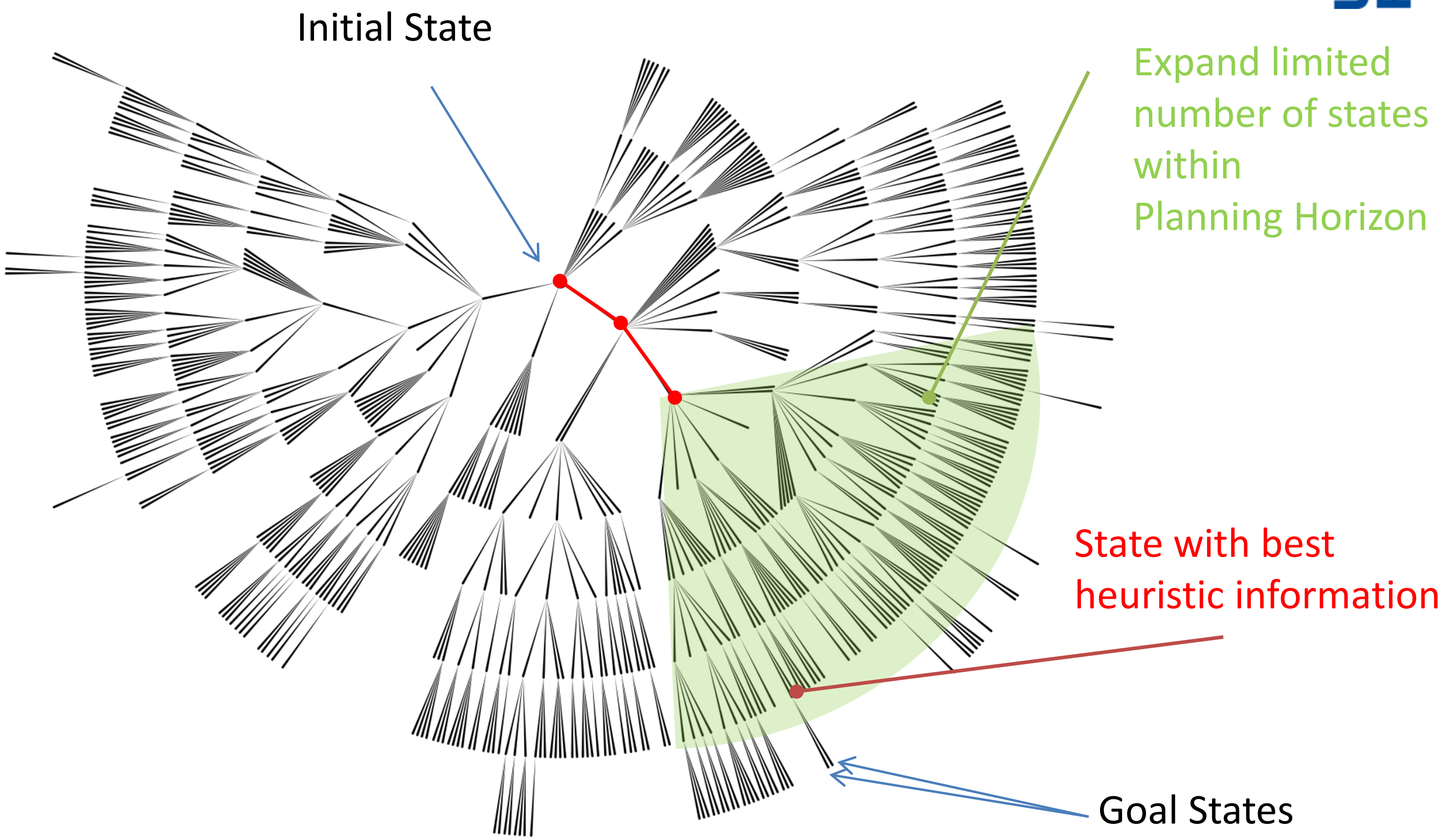
Domain Predictive Control



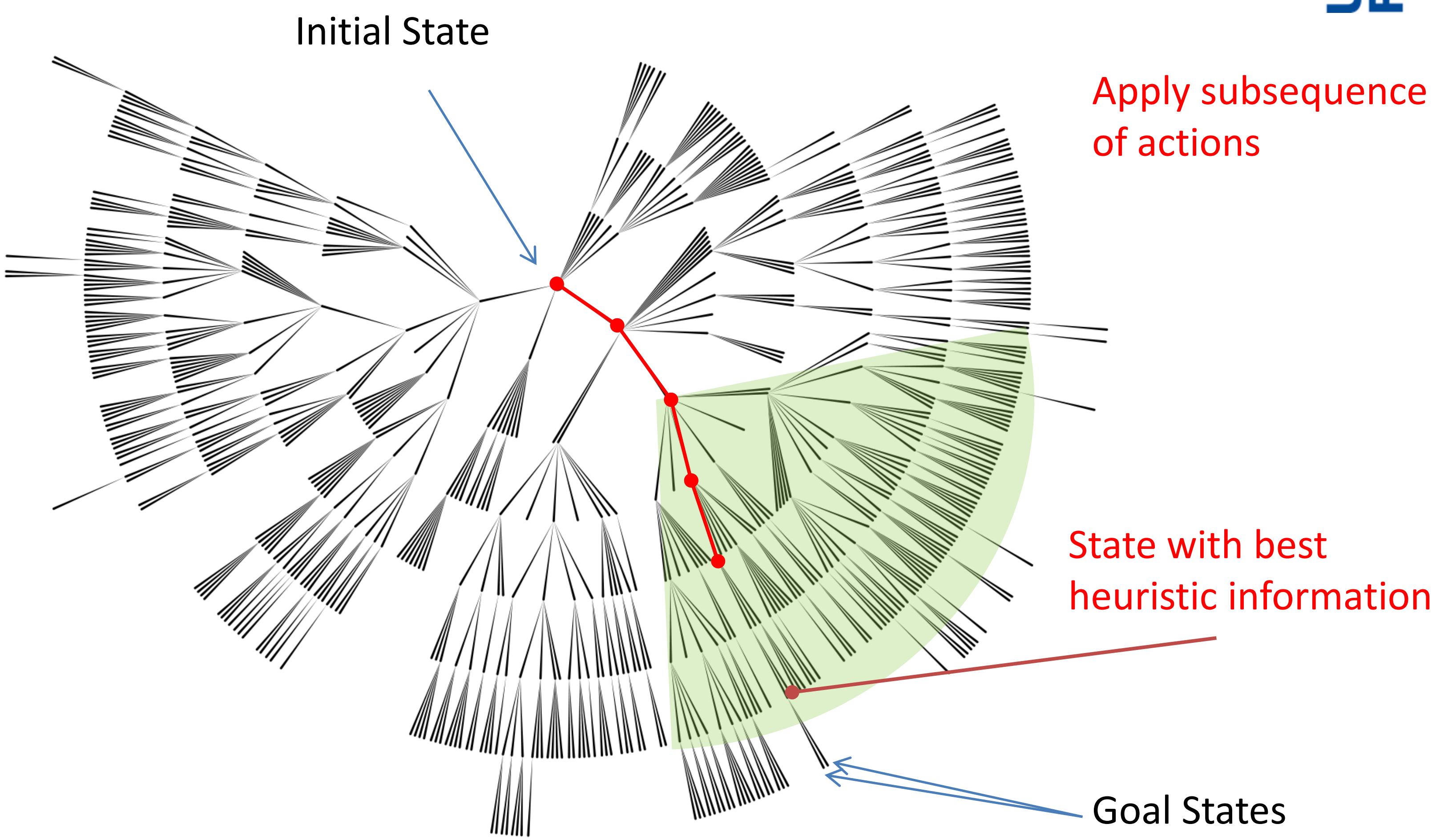
Domain Predictive Control



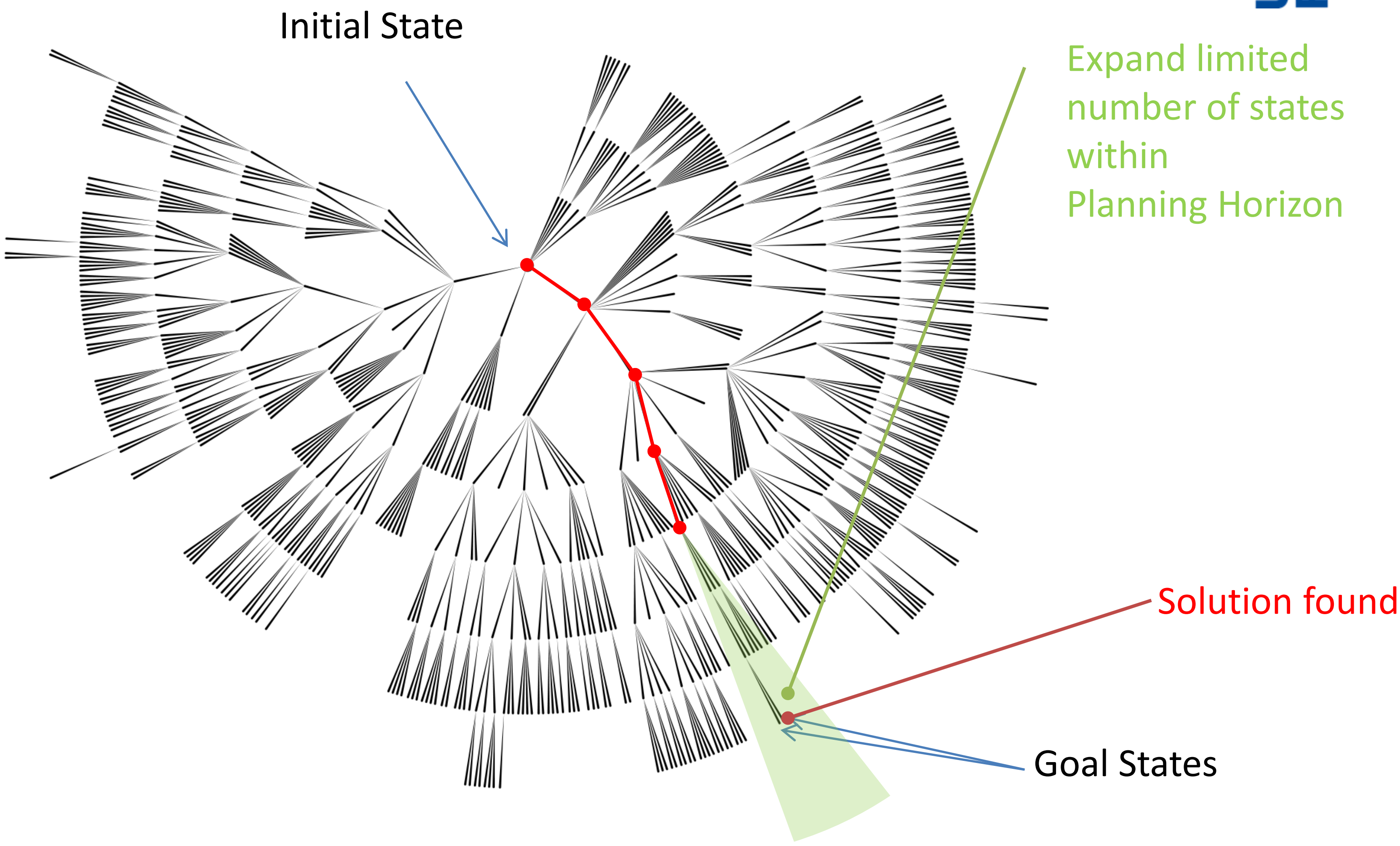
Domain Predictive Control



Domain Predictive Control



Domain Predictive Control



Domain Predictive Control

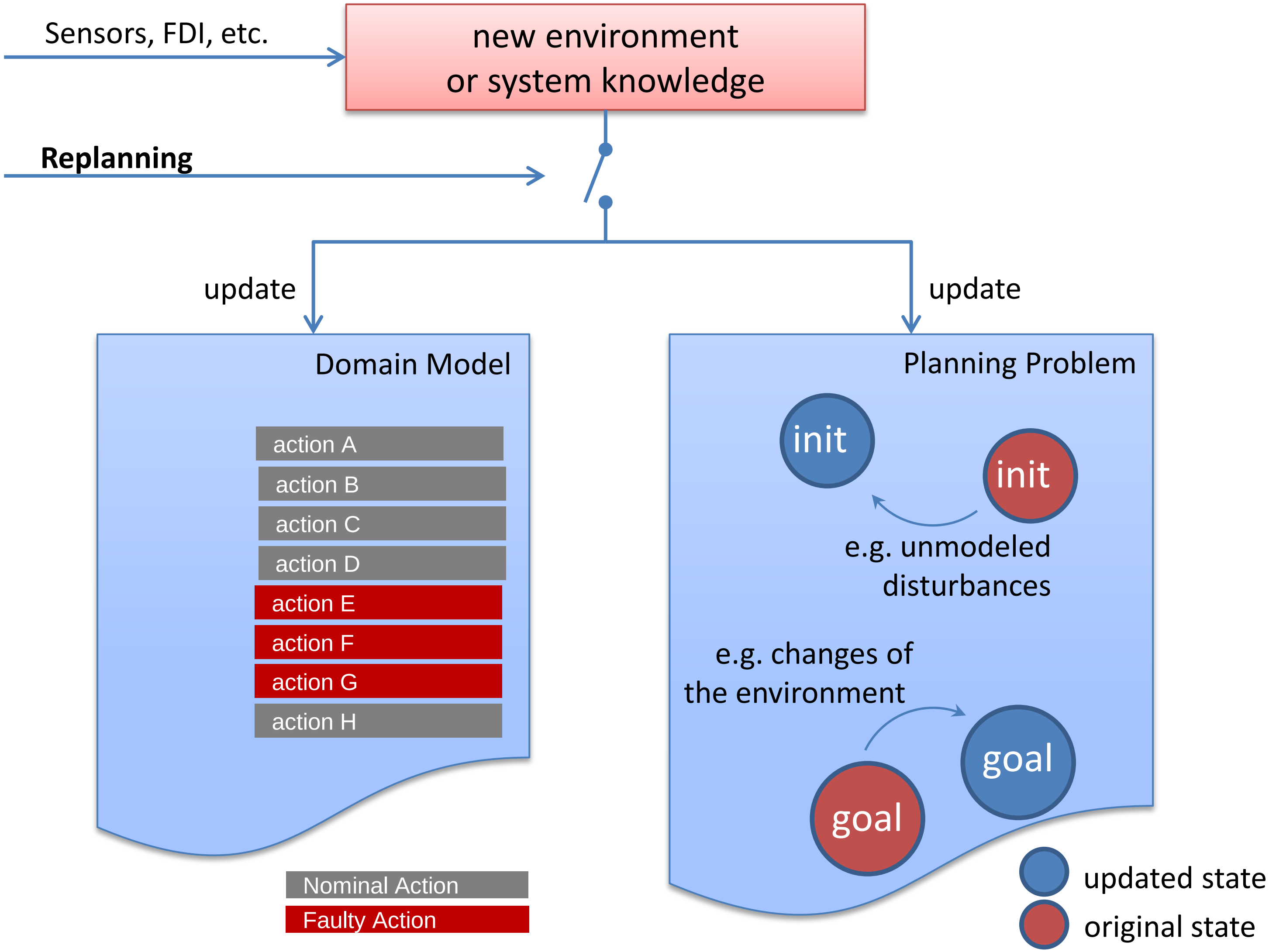
- Key Aspects

Hybrid Systems
Continuous Dynamics
Boolean State Variables

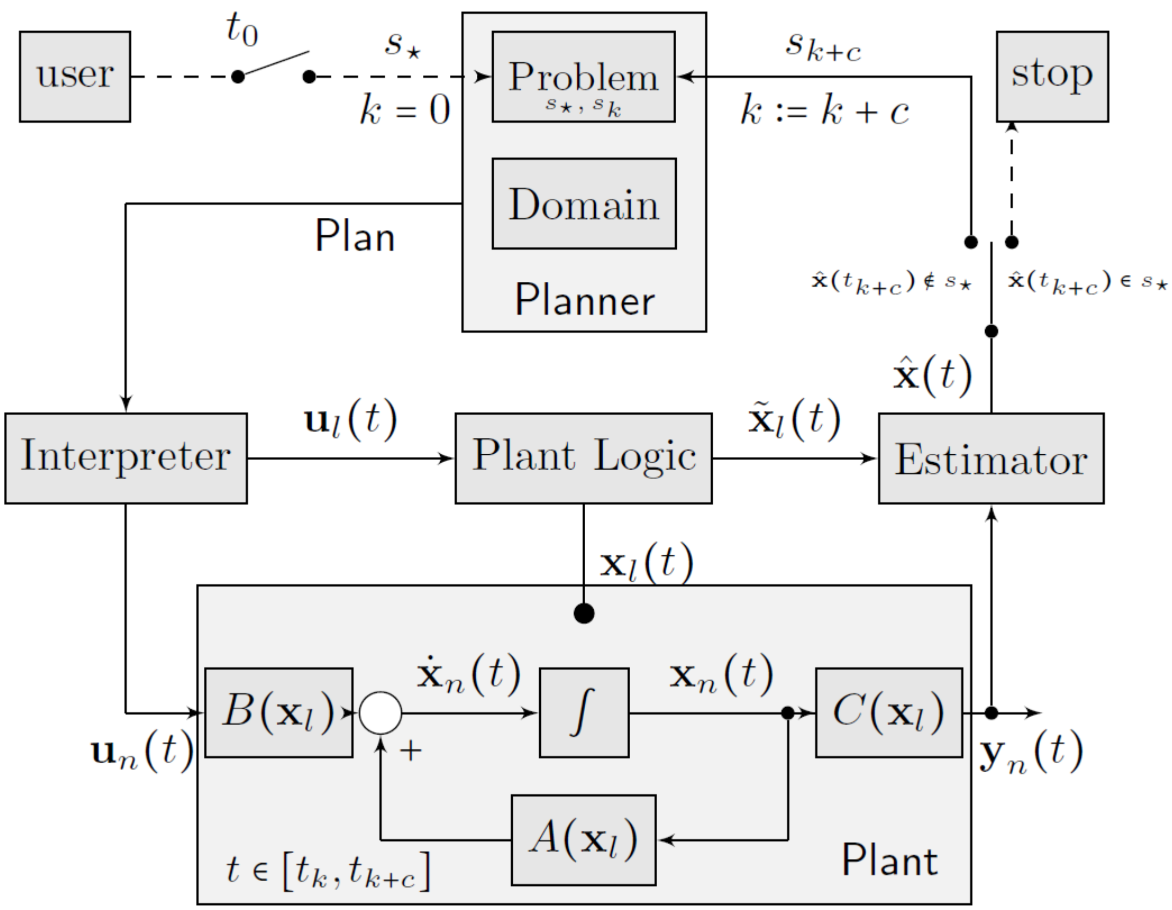
Autonomy
Reduction of Computational Effort
Quick Decision Generation

Exogenous Events
Obstacles
Reactivity

Domain Predictive Control



Domain Predictive Control Arcitecture



Outline

Motivation

From Continuous Dynamics…

… to a Domain Model

Domain Predictive Control

Exemplary Simulations

Discussion

Outlook

Exemplary Simulation

action 0	$\alpha_y = 0°$	$\alpha_x = 0°$
action 1	$\alpha_y = 0°$	$\alpha_x = 2.92°$
action 2	$\alpha_y = 0°$	$\alpha_x = -2.92°$
action 3	$\alpha_y = 2.92°$	$\alpha_x = 0°$
action 4	$\alpha_y = -2.92°$	$\alpha_x = 0°$
action 5	$\alpha_y = 2.92°$	$\alpha_x = -2.92°$
action 6	$\alpha_y = 2{,}92°$	$\alpha_x = 2.92°$
action 7	$\alpha_y = 2.92°$	$\alpha_x = -2.92°$
action 8	$\alpha_y = -2.92°$	$\alpha_x = -2.92°$

$$\delta_i = 0.5\, s, \forall\, i \in [0,8]$$

$p_{\vdash}$ =

$[(x > 8) \lor (x < 2) \lor (y > 3) \lor (y < 2)] \land$

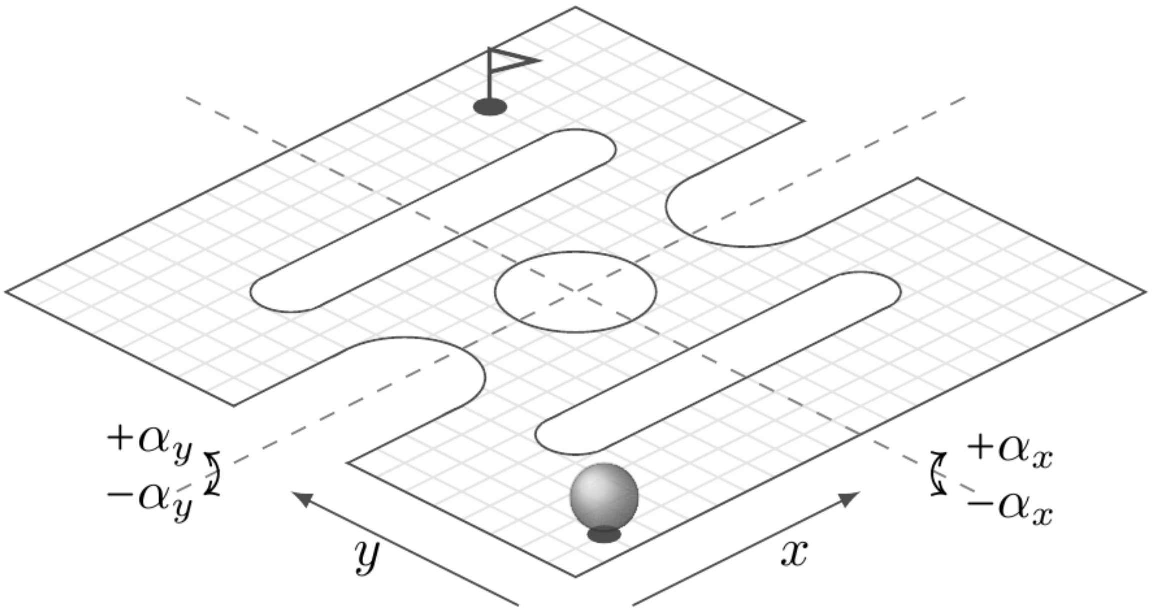
$[(x > 6) \lor (x < 4) \lor (y > 6) \lor (y < 4)] \land$

$[(x > 4) \lor (x < 0) \lor (y > 6) \lor (y < 4)] \land$

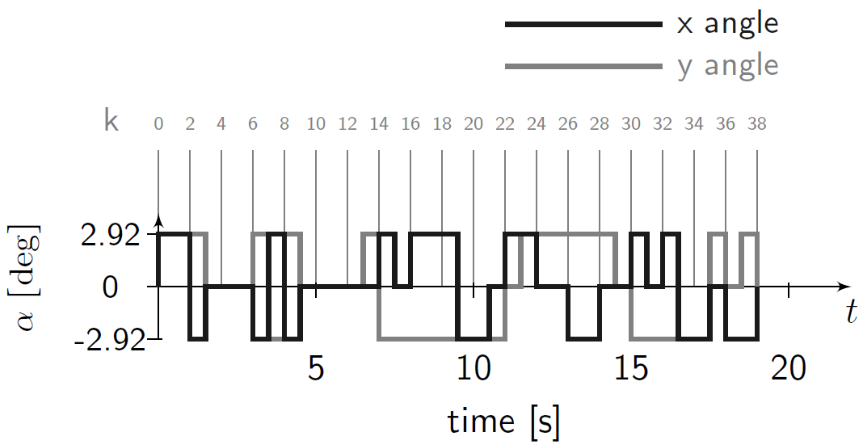
$[(x > 10) \lor (x < 6) \lor (y > 6) \lor (y < 4)] \land$

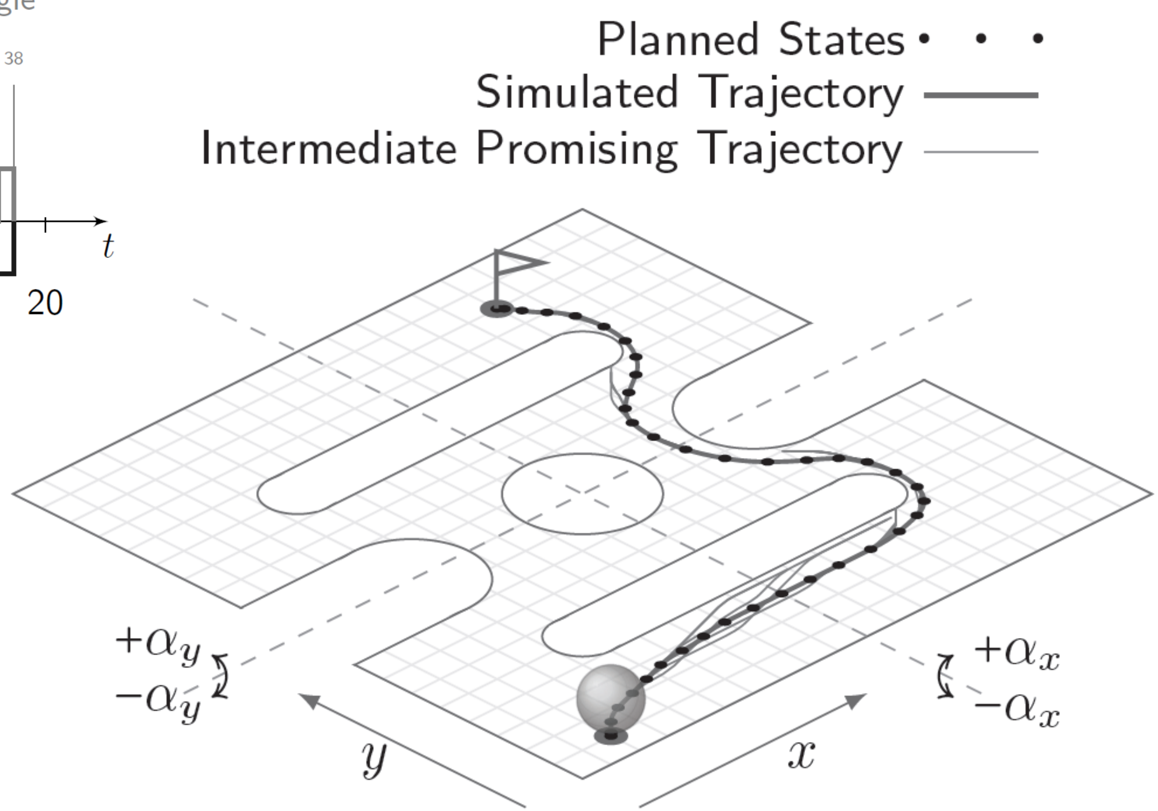
$[(x > 8) \lor (x < 2) \lor (y > 8) \lor (y < 7)] \land$

$[(x > 0) \land (x < 10) \land (y > 0) \land (y < 10)]$



Exemplary Simulation

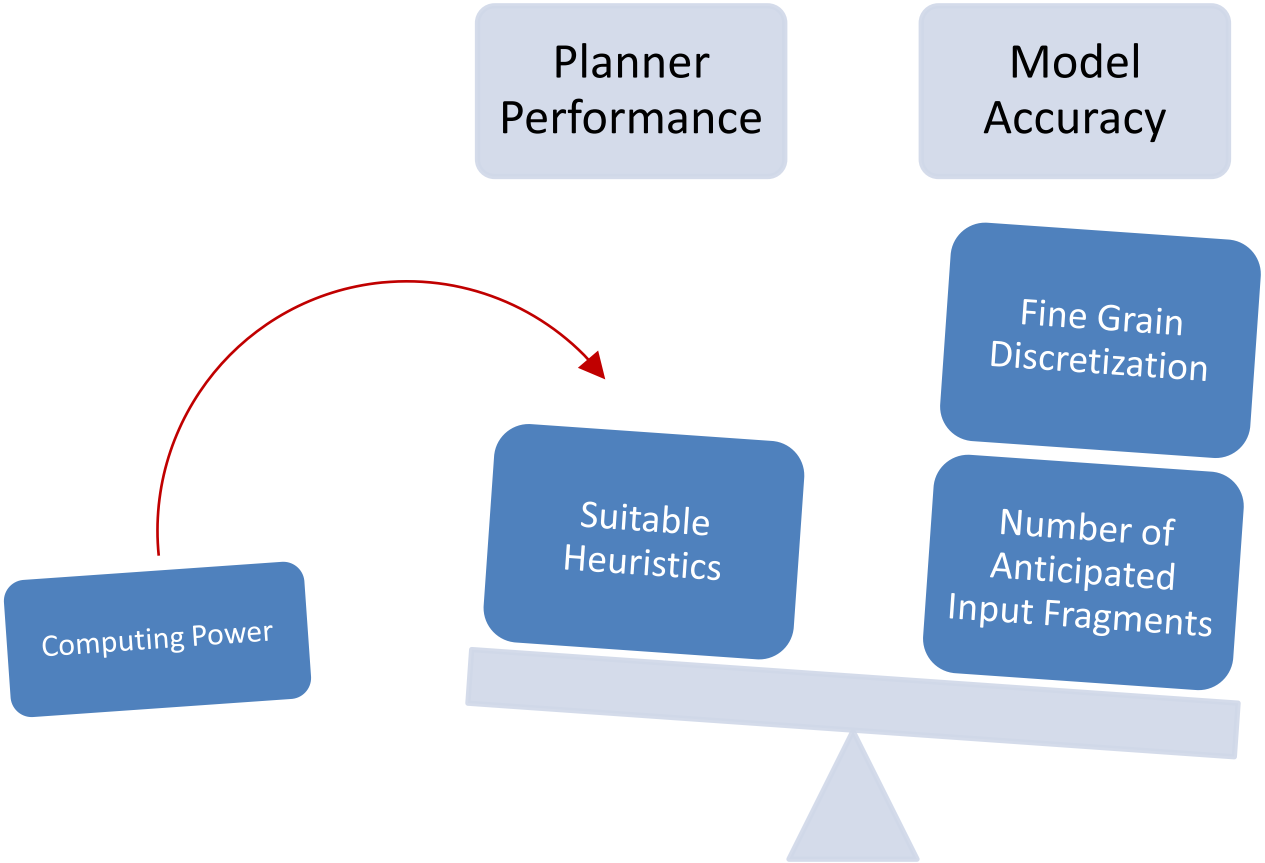




Outline

- Motivation
- From Continuous Dynamics…
- … to a Domain Model
- Domain Predictive Control
- Exemplary Simulations
- Discussion
- Outlook

Discussion


Planner Performance
Model Accuracy
Fine Grain Discretization
Suitable Heuristics
Number of Anticipated Input Fragments
Computing Power

Summary and Conclusion

Pros

- **general** problem formulation
- **changing system configurations**
- restricted state space
- **failure tolerance**

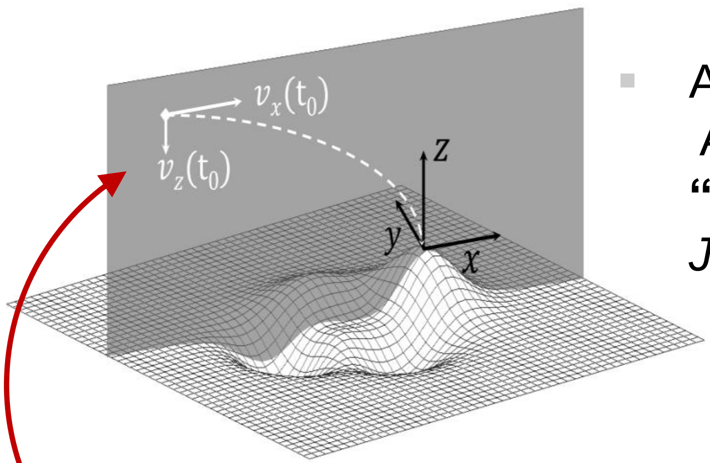
Cons

- anticipated input fragments
- discrete system dynamics
- computational effort
- currently: linear systems only

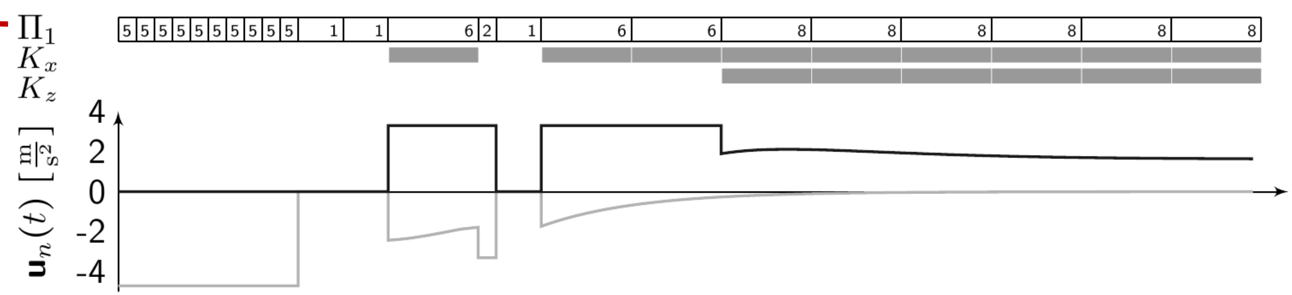
Outline

- Motivation
- From Continuous Dynamics…
- … to a Domain Model
- Domain Predictive Control
- Exemplary Simulations
- Discussion
- **Outlook**

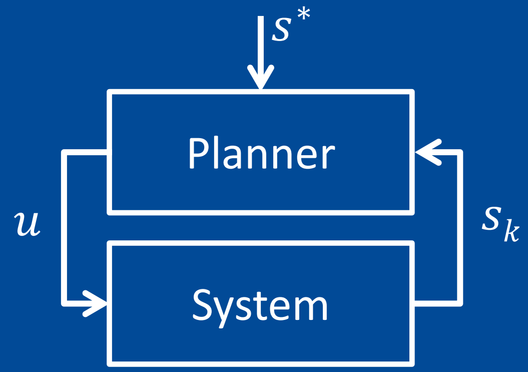
Outlook



- American Institute of Aeronautics and Astronautics Astrodynamics Specialist Conference (August 2012)
"Planning-based Autonomous Lander Control"
J. Löhr, B. Nebel, and S. Winkler



Planning Based Framework for Controlling Hybrid Systems



Albert-Ludwigs-Universität Freiburg

UNI FREIBURG

Johannes Löhr
loehr@informatik.uni-freiburg.de

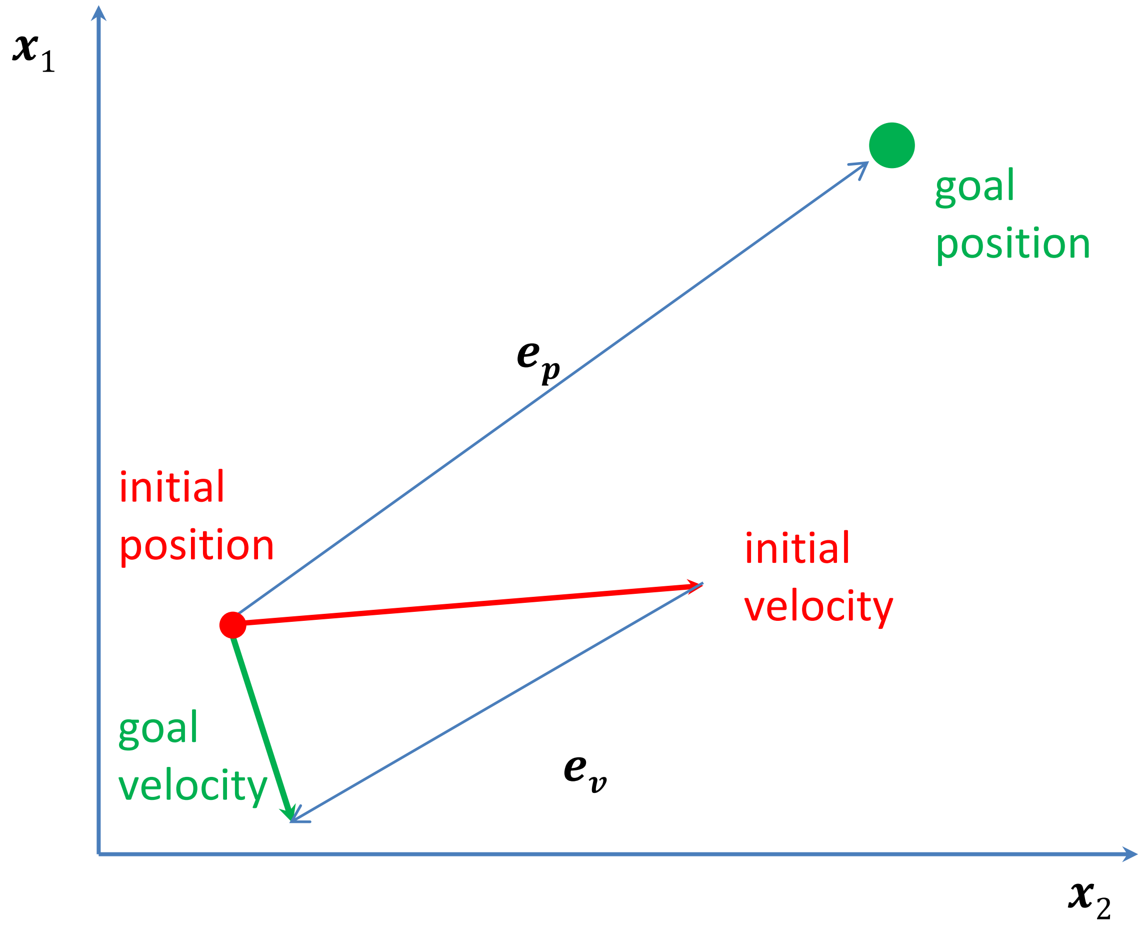
Heuristic

Error State Heuristic

$$h = \|\boldsymbol{e}_p + \boldsymbol{e}_v\|_1$$

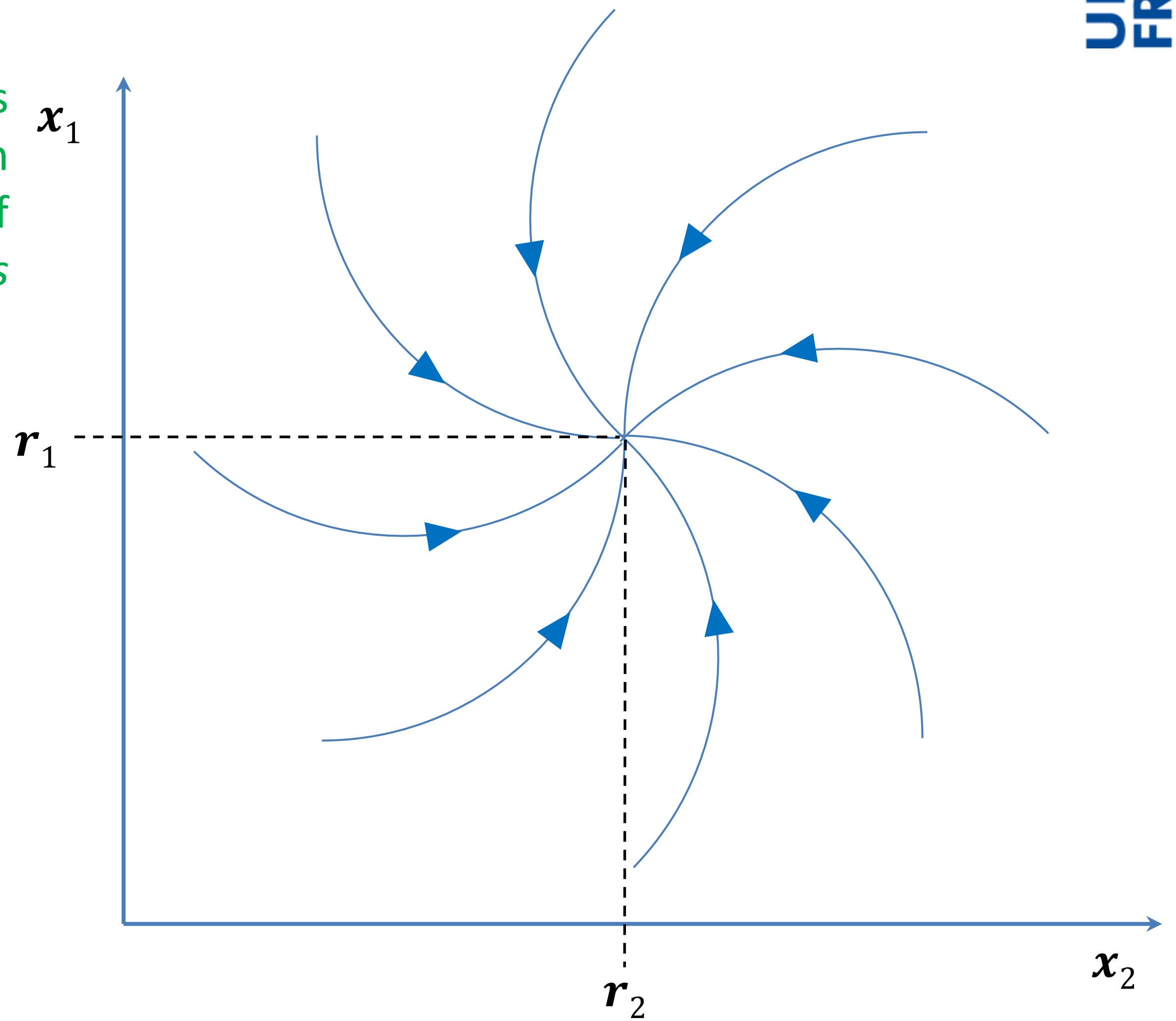
Open Heuristic Issues:

1. Weighting of numerical variables
2. Weighting between numerical and logical variables



Stable Dynamics

Homogeneous
Solution
of
Stable Dynamics



$\dot{x}_n(t) = A_{cl}\,x_n(t) + B u_n(t)$

A_{cl} state feedback controlled
$u_n(t) = -K\,x_n(t)$

closed loop dynamics
$A_{cl} = A - BK$

choose controller K such that
$Re(eig|A - B\,K|) < 0$

Stable Dynamics

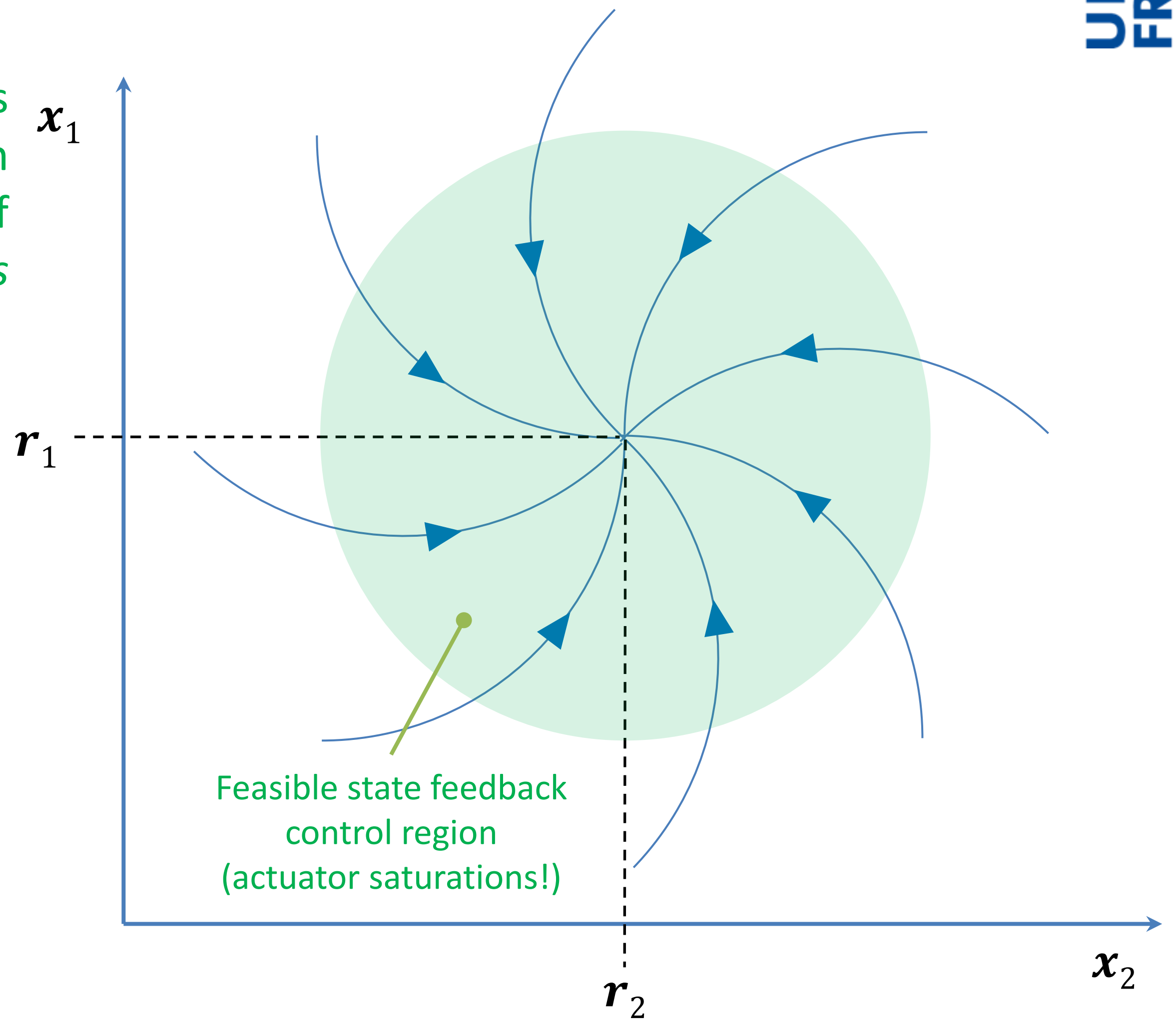
Homogeneous
Solution
of
Stable Dynamics

$\dot{\boldsymbol{x}}_n(t) = A_{\boldsymbol{cl}}\,\boldsymbol{x}_n(t) + B\boldsymbol{u}_n(t)$

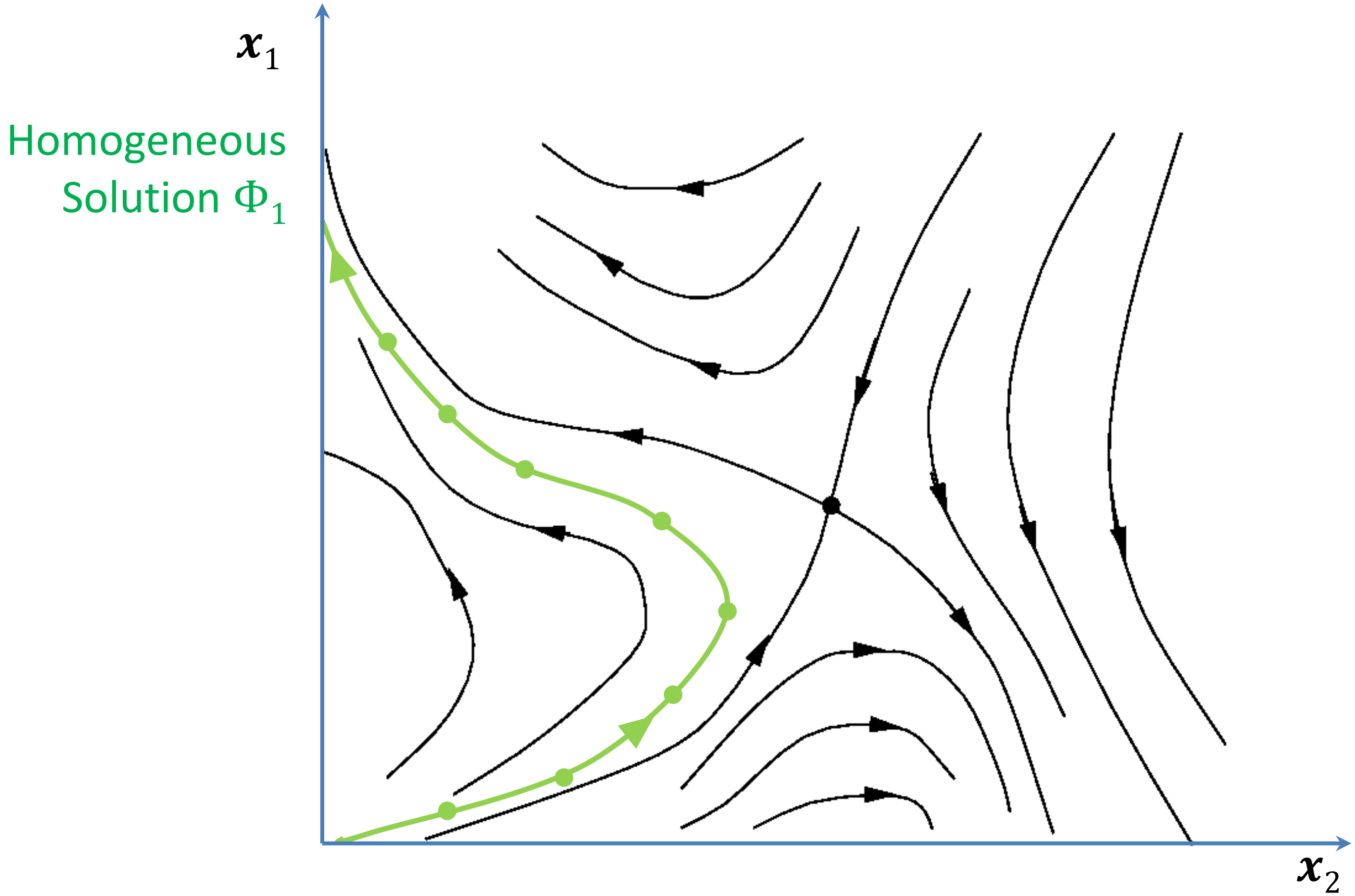
$A_{\boldsymbol{cl}}$ state feedback controlled
$\boldsymbol{u}_n(t) = -\mathrm{K}\,\boldsymbol{x}_n(t)$

closed loop dynamics
$A_{\boldsymbol{cl}} = \mathrm{A} - \mathrm{BK}$

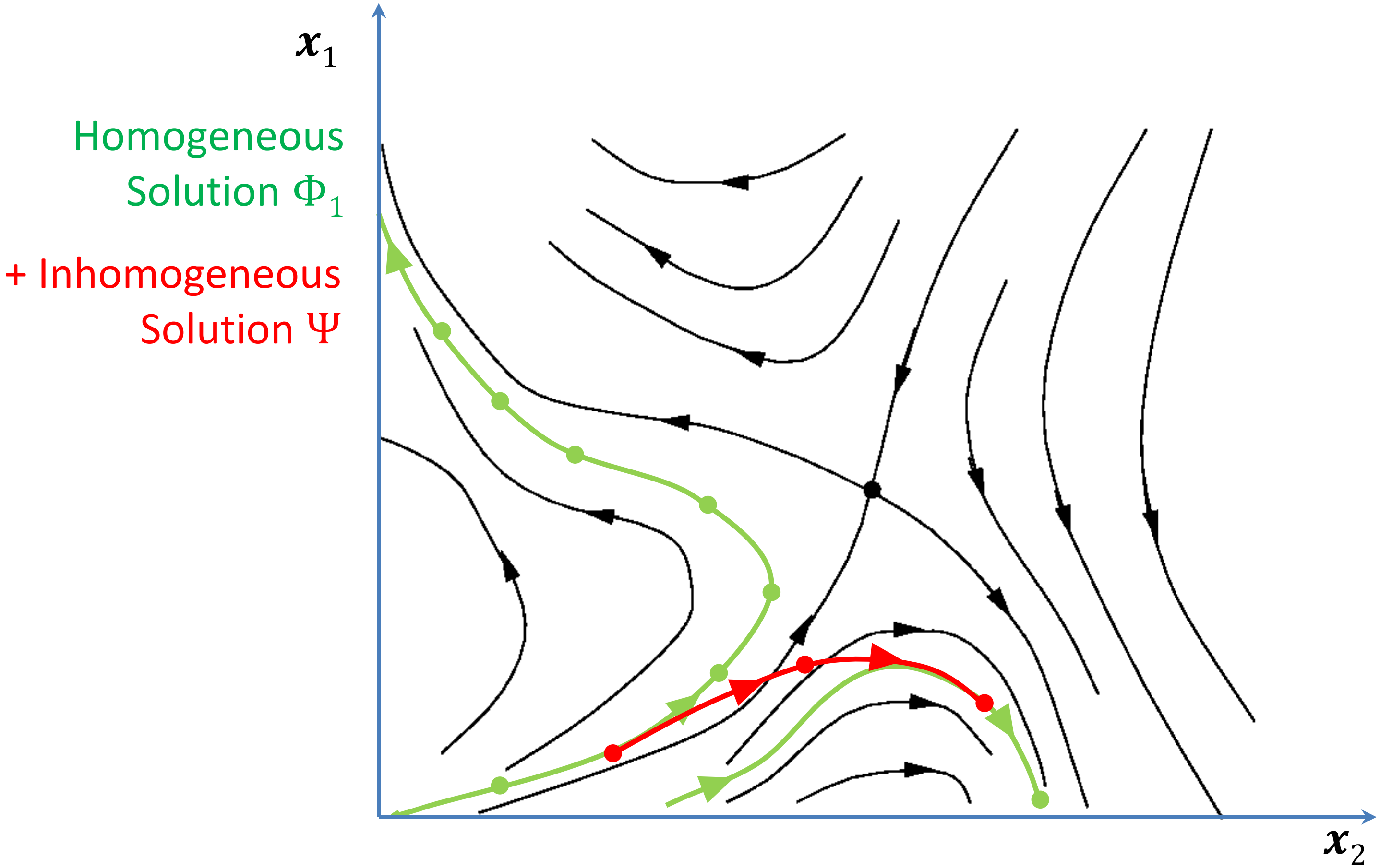
choose controller K such that
$Re(eig|A - \mathrm{B}\,K|) < 0$



Discretized Dynamics

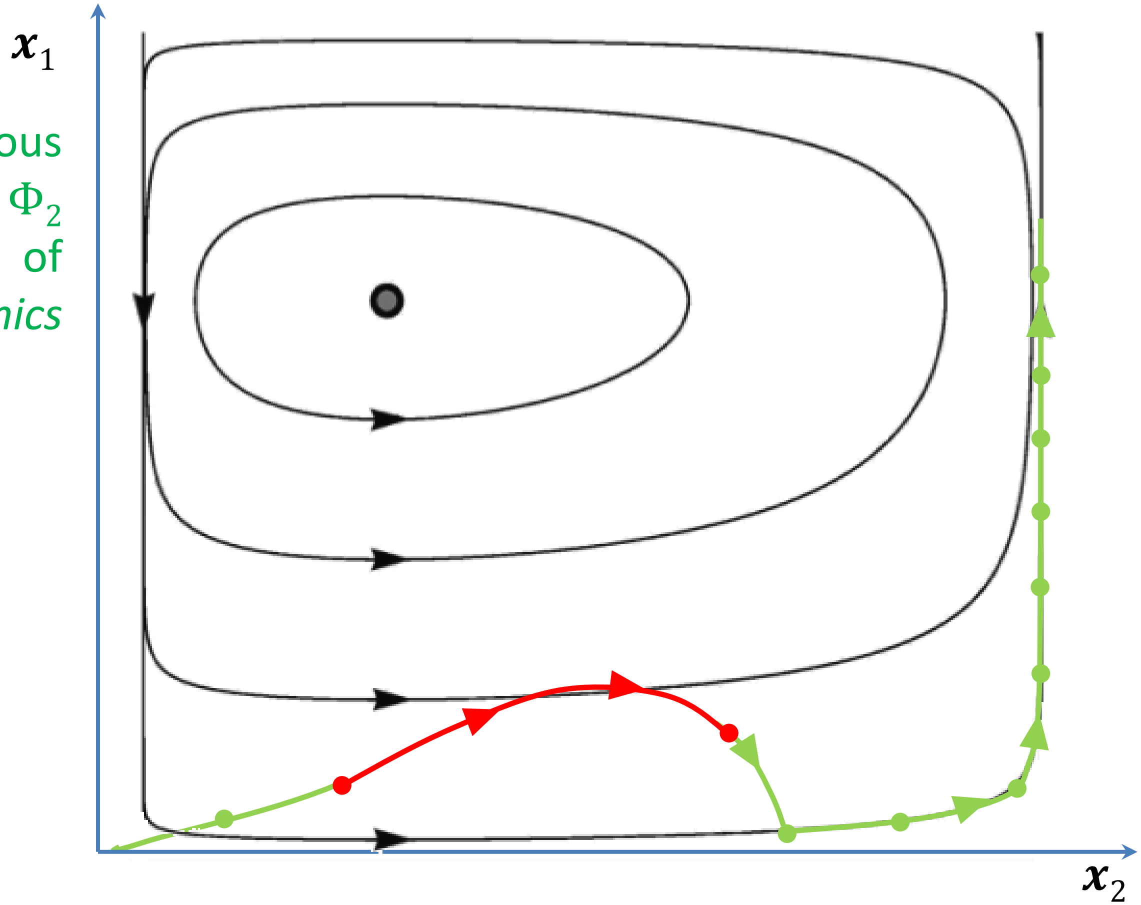


Discretized Dynamics

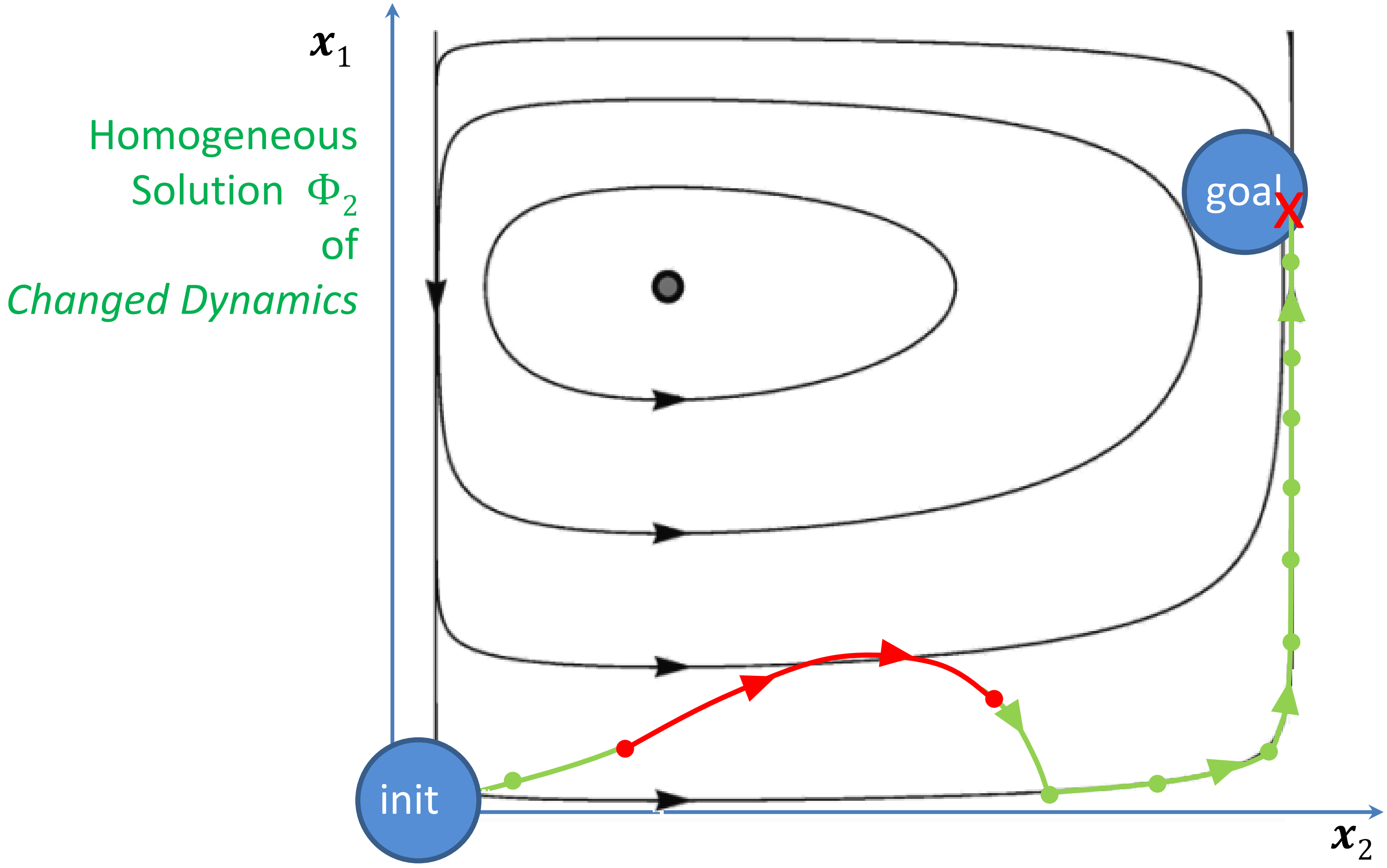


Discretized Dynamics

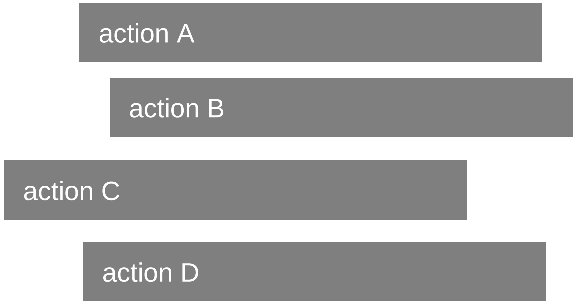
Homogeneous Solution Φ_2 of *Changed Dynamics*



Discretized Dynamics



The Domain Model
A Piecewise Affine System?



$$\boldsymbol{x}_n(k+1) = \Phi_i\, \boldsymbol{x}_n(k) + \Psi_i\;, \boldsymbol{x}_n \in \Omega_i$$

$$\Omega_i \cap \Omega_j = \emptyset, \qquad \forall\, i \neq j$$

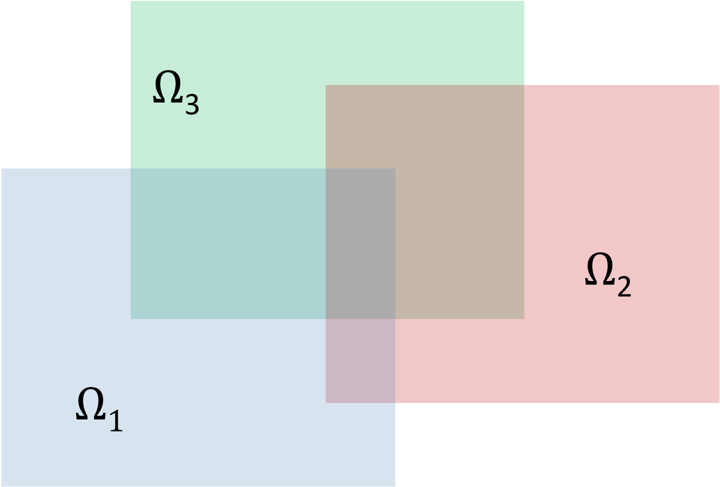
$$\Omega_i \cap \Omega_j = \mathcal{X}, \qquad \forall\, i \neq j$$

$\mathcal{X}$:

is not necessarily empty,

we explicitly allow for overlaps of Ω_i

Depicts the state space we can choose between actions



Time Discretization