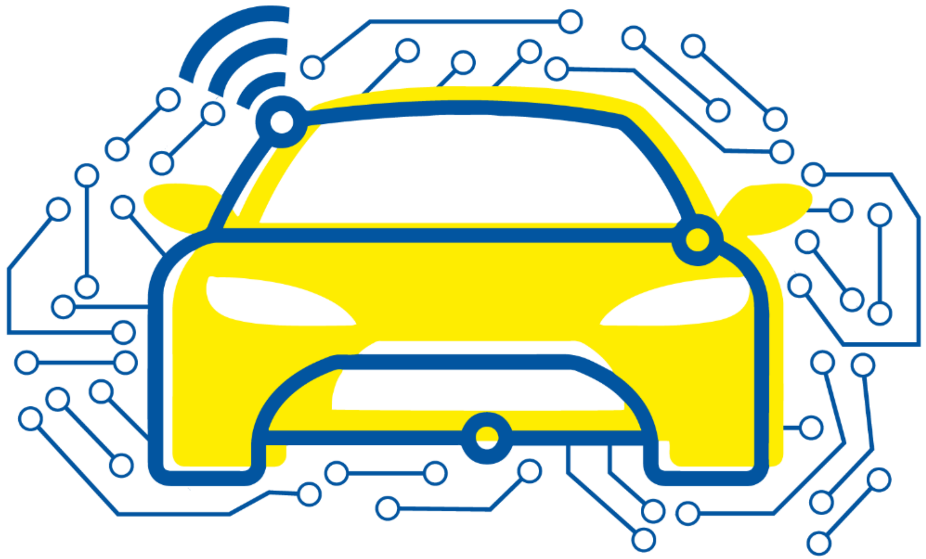
Automated and Connected Driving Challenges

Section 4 – Vehicle Guidance

Vehicle Guidance on Guidance Level

Direct Multiple Shooting



Bastian Lampe, M.Sc.

Institute for Automotive Engineering

Vehicle Guidance on Guidance Level

The direct Multiple-Shooting approach

Dynamic optimization problem is reduced to a static one:

→ finite-dimensional discrete parameterization of the piece-wise continuous control-function

$$\mathbf{u}(t) \text{ with } t \in [t_0, t_N]; N \in \mathbb{N}$$
$$\mathbf{u}(t) = \mathbf{u}_i \text{ with } t \in [t_i, t_{i+1}]$$

and

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}_i(t), \mathbf{u}_i) \text{ with } t \in [t_i, t_{i+1}]$$

States will be calculated based on **forward integration** on each **interval** from an artificial **initial value** $\boldsymbol{s_i}$.
Optimization through specific **change of control variables** $\boldsymbol{u_i}$ and **initial states** $\boldsymbol{s_i}$ on interval i.
Consider continuity condition.

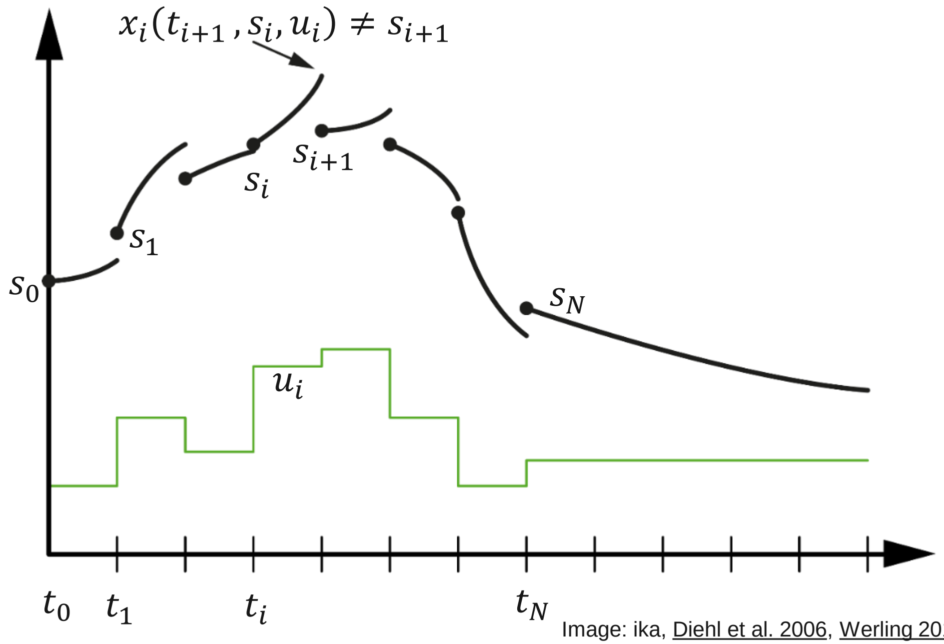


Image: ika, Diehl et al. 2006, Werling 2017
Source: Diehl et al. 2006, Werling 2017

Vehicle Guidance on Guidance Level

The direct Multiple-Shooting approach

Dynamic optimization problem is reduced to a static one:

→ finite-dimensional discrete parameterization of the piece-wise continuous control-function

$$\mathbf{u}(t) \text{ with } t \in [t_0, t_N]; N \in \mathbb{N}$$
$$\mathbf{u}(t) = \mathbf{u}_i \text{ with } t \in [t_i, t_{i+1}]$$

and

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}_i(t), \mathbf{u}_i) \text{ with } t \in [t_i, t_{i+1}]$$
$$\mathbf{x}(t_{i+1}, \mathbf{s}_i, \mathbf{u}_i) = \mathbf{s}_{i+1}$$

States will be calculated based on **forward integration** on each **interval** from an artificial **initial value** $\boldsymbol{s_i}$.
Optimization through specific **change of control variables** $\boldsymbol{u_i}$ and **initial states** $\boldsymbol{s_i}$ on interval i.
Consider continuity condition.

+ Complex vehicle models can be used; output can be directly feed-forward to the controller; Good convergence behaviour for unstable and badly non-linear systems

- local minimum; runtime increases exponentially with number of states; early abort of the optimization will lead to invalid results

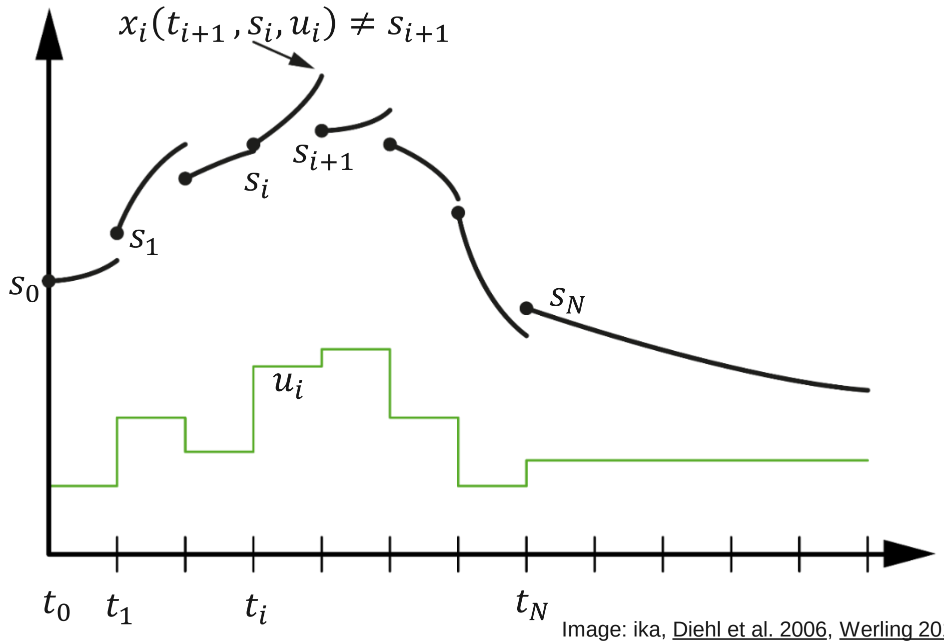


Image: ika, Diehl et al. 2006, Werling 2017
Source: Diehl et al. 2006, Werling 2017

Vehicle Guidance on Guidance Level

Direct Multiple-Shooting - Optimization process

Initialization:

$$\mathbf{q}^0 = [\mathbf{u}_0, \dots, \mathbf{u}_N, \mathbf{s}_0, \dots, \mathbf{s}_N]$$

Optimization

$$\min_{\mathbf{q}} \sum_{i=0}^{N-1} J_i(x(t_i), u(t_i))$$

with respect to

$$g_i = 0 \quad \forall\, i \in [0, N]$$
$$g_f = 0$$
$$h_i \leq 0 \quad \forall\, i \in [0, N]$$
$$\mathbf{x}_i(t_{i+1}, \mathbf{s}_i, \mathbf{u}_i) = s_{i+1} \quad \forall\, i \in [0, N)$$

Integration

$$\dot{\mathbf{x}}_i(t) = f(\mathbf{x}_i(t), \mathbf{u}_i) \quad t \in [t_i, t_{i+1}]$$
$$\mathbf{x}_i(t_i) = \mathbf{s}_i$$

$\mathbf{x}_{0\dots N}(t)$,
$\mathbf{u}_{0\dots N}$

Evaluation

$$J_i = J(\mathbf{x}_i(t), \mathbf{u}_i)$$
$$g_f = g(\mathbf{x}_f, \mathbf{u}_f)$$
$$g_i = g(\mathbf{x}_i(t), \mathbf{u}_i)$$
$$h_i = h(\mathbf{x}_i(t), \mathbf{u}_i)$$

$J_{0\dots N}$,
g_f,
$g_{0\dots N}$,
$h_{0\dots N}$

Source: Diehl et al. 2006, Werling 2017

Vehicle Guidance on Guidance Level

Direct Multiple-Shooting - Optimization process

Optimization

$$\min_{\mathbf{q}} \sum_{i=0}^{N-1} J_i(x(t_i), u(t_i))$$

with respect to

$$g_i = 0 \quad \forall\, i \in [0, N]$$
$$g_f = 0$$
$$h_i \leq 0 \quad \forall\, i \in [0, N]$$
$$\mathbf{x}_i(t_{i+1}, \mathbf{s}_i, \mathbf{u}_i) = s_{i+1} \quad \forall\, i \in [0, N)$$

Initialization:

$$\mathbf{q}^0 = [\mathbf{u_0}, \dots, \mathbf{u}_N, \mathbf{s_0}, \dots, \mathbf{s}_N]$$

$\mathbf{q}^k$

k: Number of Iteration

Integration

$$\dot{\mathbf{x}}_i(t) = f(\mathbf{x}_i(t), \mathbf{u}_i) \quad t \in [t_i, t_{i+1}]$$
$$\mathbf{x}_i(t_i) = \mathbf{s}_i$$

$\mathbf{x}_{0\dots N}(t)$,
$\mathbf{u}_{0\dots N}$

Evaluation

$$J_i = J(\mathbf{x}_i(t), \mathbf{u}_i)$$
$$g_f = g(\mathbf{x}_f, \mathbf{u}_f)$$
$$g_i = g(\mathbf{x}_i(t), \mathbf{u}_i)$$
$$h_i = h(\mathbf{x}_i(t), \mathbf{u}_i)$$

$J_{0\dots N}$,
g_f,
$g_{0\dots N}$,
$h_{0\dots N}$

Source: Diehl et al. 2006, Werling 2017