

Model Predictive Control for Path Tracking in Autonomous Racing

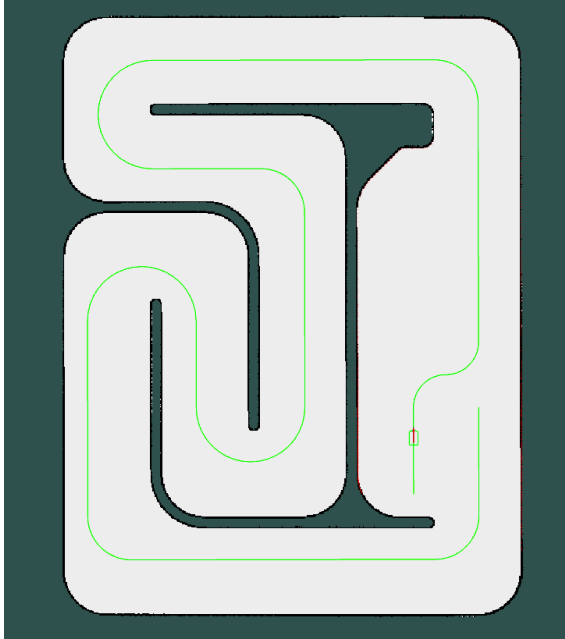
Final presentation

D. Kosak, S. Cheaib, J. Huang, L. Freisinger

09. Februar 2021

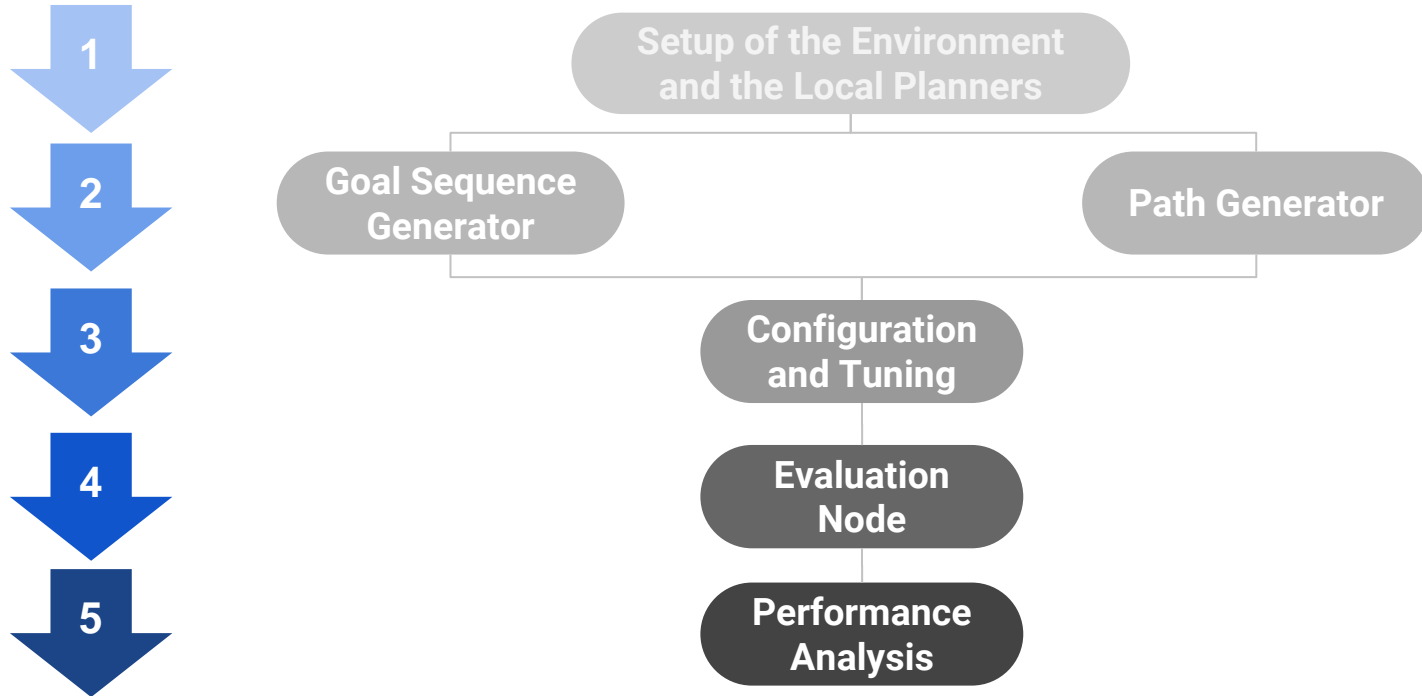


Recap - project objective



1. **Implement** a **controller** in order to achieve a **race track lap** at a given car velocity as **accurate as possible** by using **MPC**.
2. **Prevent** the car from **crashing** into the tracks' borders by implementing **suitable tuning**.
3. **Evaluate** the **performance** compared to:
 - base_local_planner
 - teb_local_planner
 - mpc_local_planner

Implementation process



Goal sequence generator

requirement: fast implementation of the test frame

function

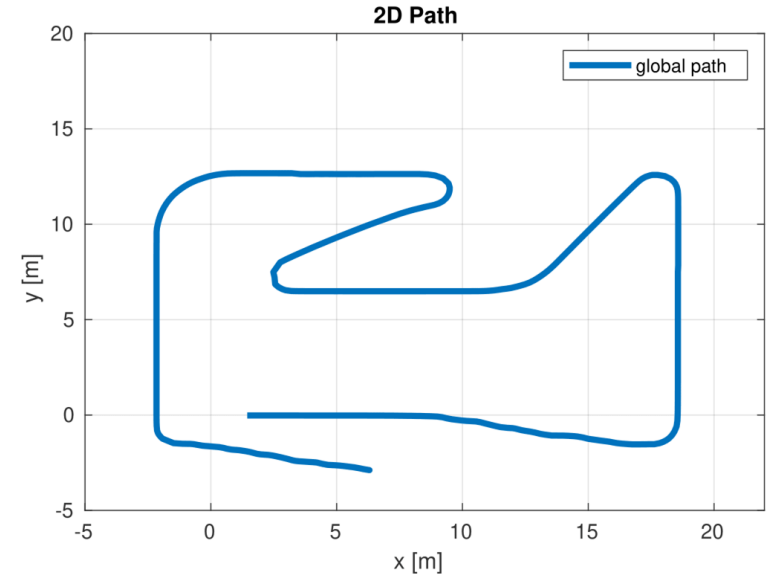
- several goal poses to guide car through track
- at least one turning between car and active goal

advantages

- ⊕ benefits early implementation process
- ⊕ race-realistic global path
- ⊕ reveals behaviour when new goal set

disadvantages

- bound to specific map
- not 100% reproducible



Path generator

Path Generator Plugin

Path Generator
Source Code

Plugin Registration

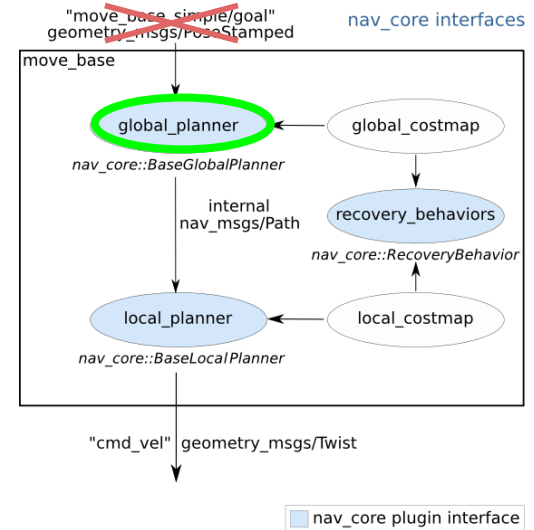
Path Drawing
Functions

Plan Publishing
Function

Plugin Description File

package.xml File

CMakeLists.txt File



Path generator: source code

Plugin description:

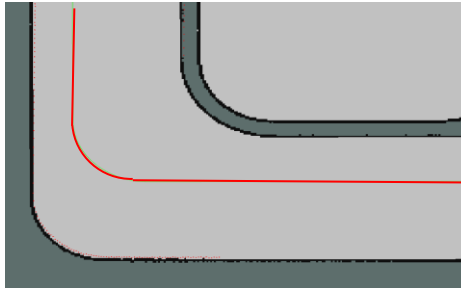
```
<!-- SadekCh: Plugin Description File for the Global Planner definition.
The name specified here will be used in the launch file to use this planner -->

<library path="lib/libpath-generator_lib">
  <class name="path_generator/PathGenerator" type="path_generator::PathGenerator"
    base_class_type="nav_core::BaseGlobalPlanner">
    <description>This is a simple path generator global planner plugin by SadekCh.
    The generated global path can only be edited by changing the used drawing
    functions in the source code</description>

  </class>
</library>
```

Path Drawing Function:

linear path
curved path



Path Publishing Function:

```
iters per sec: 18 possible: 20
[ INFO] [1612681265.343971044, 177.212000000]: global_path is being published for 6 more seconds.
iters per sec: 18 possible: 21
iters per sec: 20 possible: 22
[ INFO] [1612681266.841383454, 178.212000000]: global_path is being published for 5 more seconds.
iters per sec: 18 possible: 21
iters per sec: 16 possible: 19
[ INFO] [1612681268.486062624, 179.212000000]: global_path is being published for 4 more seconds.
iters per sec: 18 possible: 20
iters per sec: 18 possible: 20
iters per sec: 18 possible: 20
[ INFO] [1612681269.973946573, 180.212000000]: global_path is being published for 3 more seconds.
iters per sec: 16 possible: 19
iters per sec: 17 possible: 20
iters per sec: 16 possible: 18
[ INFO] [1612681271.689696160, 181.212000000]: global_path is being published for 2 more seconds.
iters per sec: 17 possible: 21
iters per sec: 16 possible: 17
iters per sec: 15 possible: 17
iters per sec: 13 possible: 16
```

Path generator: pros and cons

Advantages:

- ⊕ Easily customizable and reproducible path
- ⊕ Precalculated before local planner starts acting
- ⊕ Does not depend on current state and performance of local planner

Disadvantages:

- ⊖ “makePlan” function not very flexible to changes
- ⊖ Odometry data not easily integrated into the plugin (restricted resources)

Configuration and tuning

Objective

- alignment to the simulation environment
- robust and stable behaviour
- ensure comparability

Challenges

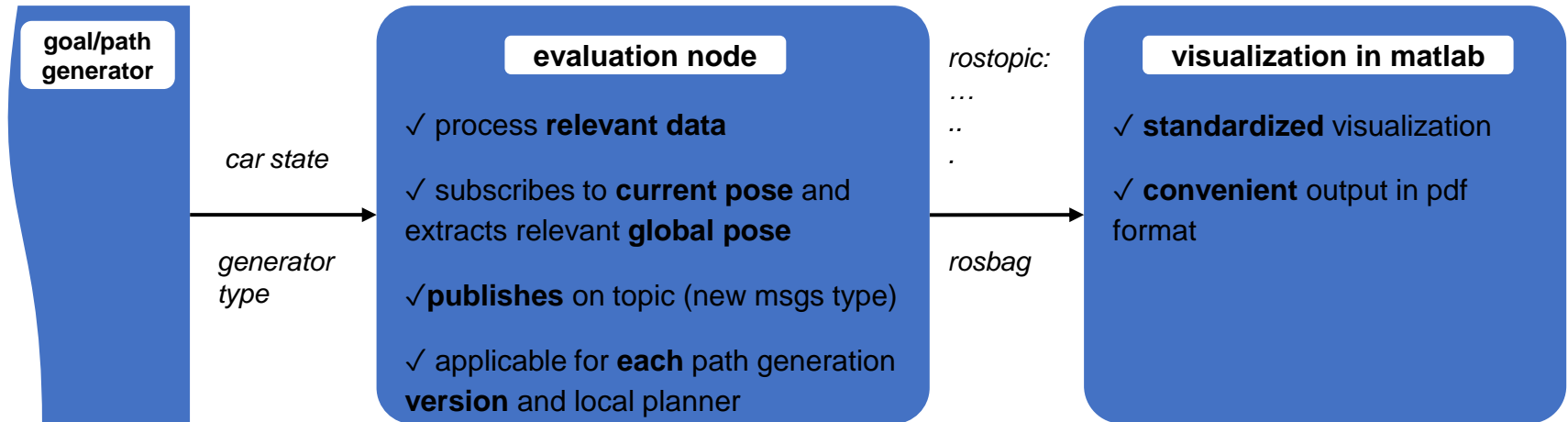
- limited documentation
- understanding effect of parameters
- large number of parameters
- sporadic failures
- reconciliation of both global path versions

Achievement

- comparability relies on robustness
- trade-off between. comparability & robustness
→ prio robustness

Evaluation Strategy

requirements: one overall **evaluation method**, **fast** evaluation, data **available** for **external** system



Evaluation results - Overview

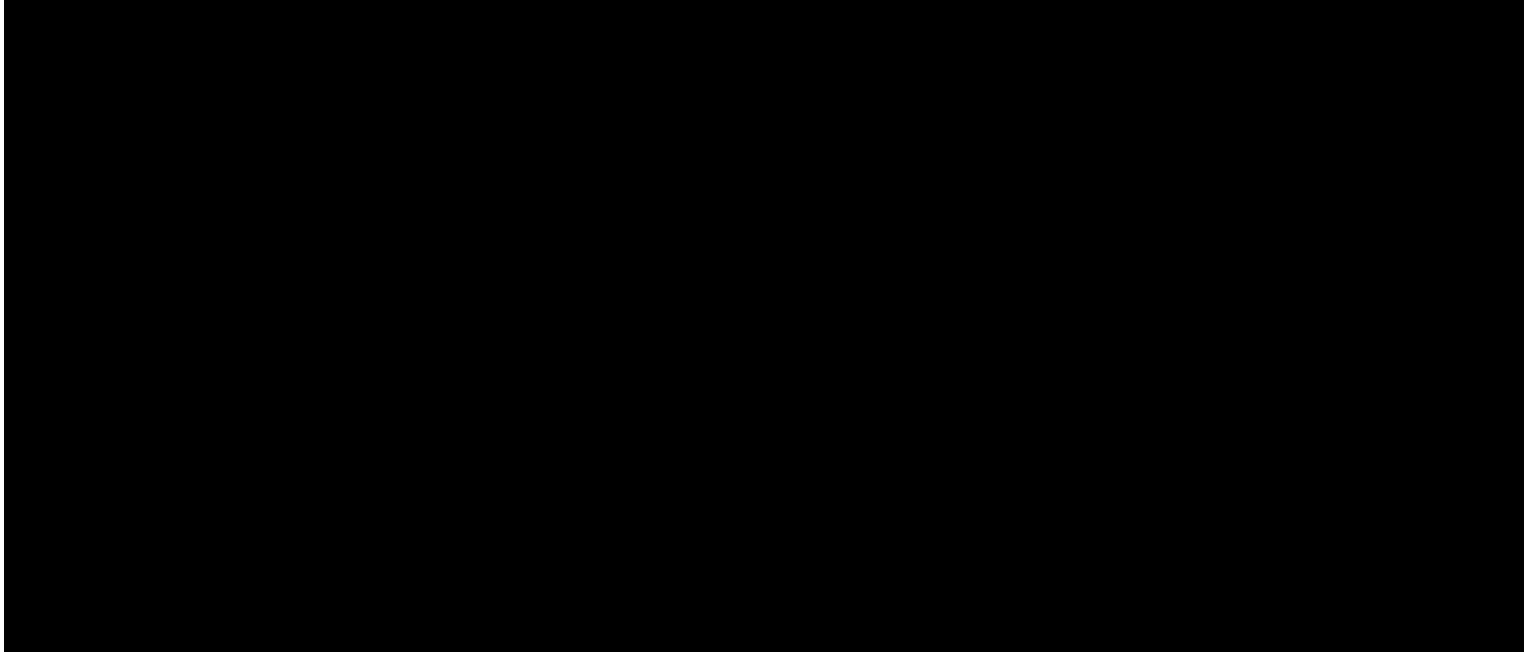
Performance measures:

- lap time
- local and global path length
- euclidean distance (mean, variance, max)
- theta (mean, variance, mse, max)

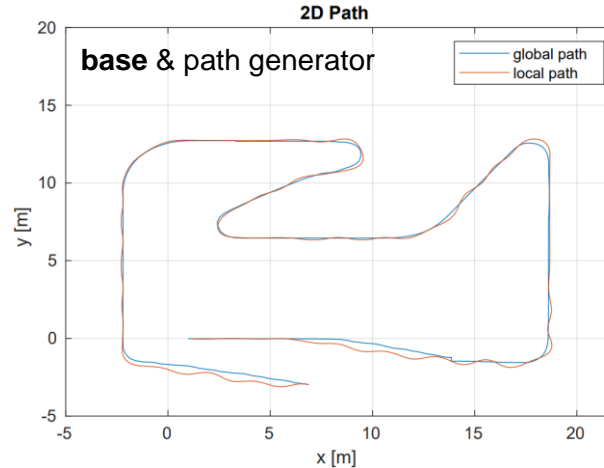
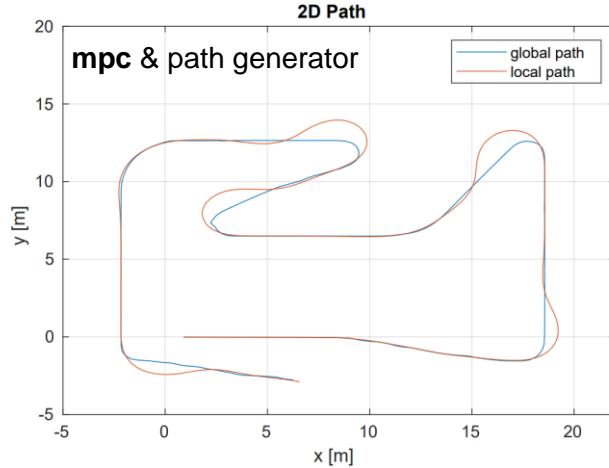
Path generation version	base_local_planner	teb_local_planner	mpc_local_planner
path generator	not stable	++	+
Goal sequence generator	+	-	++

Evaluation results

simulation of **mpc_local_planner** and **base_local_planner** combined with **goal sequence generator**



Evaluation results



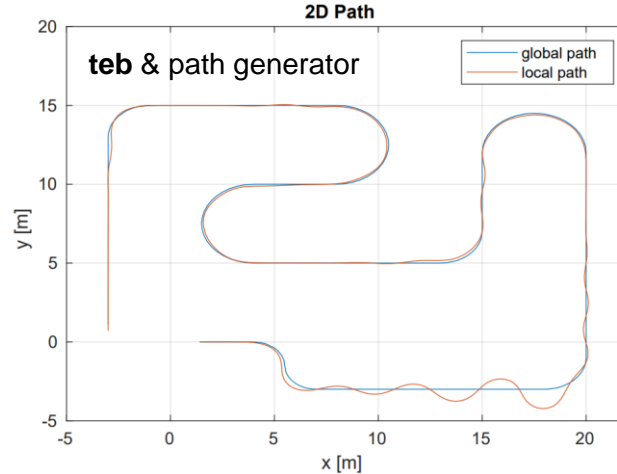
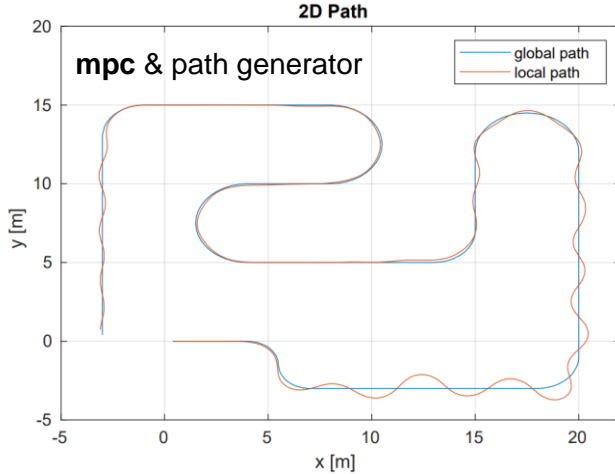
- ⊕ lowest position error in straight segment
- 🔍 high position error in curve

performance measures:

mpc:	path_len_ref	path_len_mea	lap_time	ed_mean	ed_max
	92.4054	98.3428	61.5020	0.2882	1.3232
base:	path_len_ref	path_len_mea	lap_time	ed_mean	ed_max
	93.0589	95.1138	63.5000	0.1489	0.5633

key finding:
mpc shows the **best overall performance**

Evaluation results



- 🔍 teb performs similarly to mpc
- 🔍 position error mainly influenced by oscillation

performance measures:

mpc:	path_len_ref	path_len_mea	lap_time	ed_mean	ed_max	error_mean_theta	err_max_theta
	104.8564	108.2462	73.1010	0.2223	0.8977	0.9613	1.8268
teb:	path_len_ref	path_len_mea	lap_time	ed_mean	ed_max	error_mean_theta	err_max_theta
	104.1514	105.7152	71.0980	0.2103	0.7668	0.8113	1.7871

key finding:

current state of mpc does not outperform teb

Contribution

Name	Setup	Goal Generator	Path Generator	Configuration & Tuning	Evaluation C++ Part	Evaluation Matlab Part	Others, e.g. Researchs
Dominik	P	C		C	A		C
Sadek	C		C		A		C
Jiangnan	C			A	C		C
Leonie	P			C		C	C

Remarks:

“C” - main contributor

“A” - assistant/advisor.

“P” - has participated

Conclusion

Achievements

- ✓ mpc local planner setup in gazebo simulator
- ✓ generation of global path for comparison intention
- ✓ overall evaluation package
- ✓ mpc performance analysis

Outview:

- further parameter optimization of `mpc_local_planner`
- race track including obstacles

Thanks for your attention!