2023 62nd IEEE Conference on Decision and Control (CDC)
December 13-15, 2023. Marina Bay Sands, Singapore

Robust Meta-Learning of Vehicle Yaw Rate Dynamics via Conditional Neural Processes*

Lars Ullrich, Andreas Völz and Knut Graichen

Abstract— **Trajectory planners of autonomous vehicles usually rely on physical models to predict the vehicle behavior. However, despite their suitability, physical models have some shortcomings. On the one hand, simple models suffer from larger model errors and more restrictive assumptions. On the other hand, complex models are computationally more demanding and depend on environmental and operational parameters. In each case, the drawbacks can be associated to a certain degree to the physical modeling of the yaw rate dynamics. Therefore, this paper investigates the yaw rate prediction based on conditional neural processes (CNP), a data-driven meta-learning approach, to simultaneously achieve low errors, adequate complexity and robustness to varying parameters. Thus, physical models can be enhanced in a targeted manner to provide accurate and computationally efficient predictions to enable safe planning in autonomous vehicles. High fidelity simulations for a variety of driving scenarios and different types of cars show that CNP makes it possible to employ and transfer knowledge about the yaw rate based on current driving dynamics in a human-like manner, yielding robustness against changing environmental and operational conditions.**

I. INTRODUCTION

Vehicle models are key components for trajectory planning in highly automated [1], [2], [3], [4] or even autonomous driving [5], [6], [7] that allow to predict the vehicle behavior as a function of the model inputs. Thus, physically feasible trajectories can be generated and evaluated [8]. However, the complexity of the model has a decisive impact. Simple models benefit from lower computational costs but have more restrictive assumptions and a higher model error [9], [10]. For instance, the most simple point-mass model [11], [12] abstracts the vehicle to a large extent and therefore neglects the non-holonomic behaviour. Accordingly, already parking maneuver planning requires a more complex model such as the kinematic single-track model [5]. For evasive maneuver planning, however, a kinematic model is not sufficient and dynamic single-track models are typically used for this purpose [13], [14]. When it comes to highly dynamic maneuvers, single-track drift models or even multi-body models are required [15], [16].

More sophisticated models reveal a higher level of detail in the tire model and the lateral dynamics. The less restrictive assumptions and reduced systematic model error implies not only a higher complexity and computational effort, but also increases the number of environmental and operating dependent parameters such as friction or vehicle mass [17]. Since such parameters have a significant impact on driving dynamics, variations of these parameters cannot be neglected in the trajectory planning of highly autonomous vehicles.

One way to improve the robustness to changing conditions is the use of sensor systems [18], [19], [20], which leads to higher costs and more potential failure sources. Another way might be software-based virtual sensor systems that estimate these parameters from already existing sensor systems [21], [22]. This methodology offers the possibility to create a vehicle model with fewer assumptions and a larger operational domain of validity, which is of particular importance for highly automated driving.

Numerical investigations of a kinematic single-track model with ground truth yaw angle show that the model is robust to varying parameters and the error is very small even in dynamic situations [23], [24], [25]. Thus, if the yaw angle could be determined more accurately and robustly without extended modeling of tires and lateral dynamics, prediction speed could be improved. Since the yaw rate is orientation invariant in contrast to the yaw angle, but leads to the angle by integration, the focus is on the modeling of the yaw rate.

An alternative to detailed physical description of the yaw rate is data-driven modeling [26], [25]. Likewise, there is the possibility to describe dynamic systems by means of continuous neural ODEs [27] or to supplement them by Gaussian processes [28]. Ultimately, the majority of approaches result in one final model. However, vehicle dynamics are subject to different effects depending on the use, thus several specific models and a proper approach for model switching or blending are required for widely accurate predictions [29]. In this area, meta learning could provide a remedy.

The key principle of meta-learning is learning-to-learn [30]. This is achieved, for example, by a dual exploitation of the supervised learning methodology. On the meta level, the mapping from a data set to a dedicated predictor is realized. In this paper, the baseline meta-learning approach of conditional neural processes (CNP) [31] is used to predict the yaw rate. A detailed review of the work that has been done in the area of neural processes in general is provided by [32]. To the best of the authors knowledge, this is the first time that yaw rate predictions are realized a) without detailed physical modeling of tires or lateral dynamics, b) without additional sensors, c) without limiting assumptions on dynamic, environmental, and operational domains, d) thus providing robustness to changing operational and environmental conditions, e) while providing fast predictions through meta-learning.

The paper is structured as follows: physical models of different complexity are introduced in Section II. The method-

*This research is accomplished within the project "AUTOtechagil" (FKZ 01IS22088Y). We acknowledge the financial support for the project by the Federal Ministry of Education and Research of Germany (BMBF).

The authors are with the Chair of Automatic Control, Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), Germany {lars.ullrich, andreas.voelz, knut.graichen}@fau.de

ology of CNP and the application to yaw rate prediction are described in Section III. Finally, CNP are evaluated in different scenarios and the results are compared with the various physical models in Section IV.

II. PHYSICAL MODELS

Physical models represent the state of the art in modeling yaw rate dynamics and form the basis to compare against CNP. The equations describing the yaw rate of the kinematic single-track model (KST), the dynamic single-track model (DST), and the single-track drift model (STD) are outlined below and represent a concise summary of the benchmark CommonRoad [33].

A. Kinematic Single-Track Model (KST)

In the case of the simplest physical baseline model, the yaw angle Ψ is modeled directly [33]. Therefore, the yaw rate dynamics

$$\ddot{\Psi}_{\mathrm{KST}} = \frac{\mathrm{d}}{\mathrm{d}t}\Big(\dot{\Psi}\Big) = \frac{\mathrm{d}}{\mathrm{d}t}\Big(\frac{v}{l_{\mathrm{wb}}}\tan(\delta)\Big), \tag{1}$$

is determined by the derivative of the yaw angle dynamics $\dot{\Psi}$, where v denotes the velocity, δ the steering angle, and l_{wb} the wheelbase.

B. Dynamic Single-Track Model (DST)

In contrast, the yaw rate dynamics $\ddot{\Psi}_{\mathrm{DST}}$ of the dynamic single-track model [33] is given by

$$\ddot{\Psi}_{\mathrm{DST}} = \frac{1}{I_z}\Big(l_f C_f \delta + (l_r C_r - l_f C_f)\beta - (l_f^2 C_f + l_r^2 C_r)\frac{\dot{\Psi}}{v}\Big) \tag{2}$$

with the moment of inertia I_z of the vehicle about the z-axis, the slip angle β at the center of gravity and the distances l_f, l_r from the center of gravity to the front and rear axle, respectively. The cornering stiffness C_f, C_r for front and rear

$$C_i = \mu C_{S,i} F_{z,i}, \quad i \in \{f, r\} \tag{3}$$

depends on the friction coefficient μ, specific cornering stiffness coefficients $C_{S,f}, C_{S,r}$, and the vertical forces

$$F_{z,f} = m\frac{g l_r - a_{\mathrm{long}} h_{\mathrm{cg}}}{l_r + l_f}, \quad F_{z,r} = m\frac{g l_f + a_{\mathrm{long}} h_{\mathrm{cg}}}{l_r + l_f} \tag{4}$$

that take into account the load transfer caused by the longitudinal acceleration a_{long} as a function of the mass m and the height of the center of gravity h_{cg}.

C. Single-Track Drift Model (STD)

The single-track drift model [33] extends the consideration of lateral dynamics and the complexity of the tire model. The yaw rate dynamics $\ddot{\Psi}_{\mathrm{STD}}$ of the single-track drift model

$$\ddot{\Psi}_{\mathrm{STD}} = \frac{1}{I_z}\Big(F_{y,f}\cos(\delta) l_f - F_{y,r} l_r + F_{x,f}\sin(\delta) l_f\Big) \tag{5}$$

is computed by means of the longitudinal tire forces $F_{y,f}, F_{y,r}$ for front and rear as well as the lateral tire force $F_{x,f}$ for front. The longitudinal and lateral tire forces are calculated via the Pacejka magic tire formula for combined slip [34]. This formula, which is omitted due to space constraints, takes into account the vertical tire forces $F_{z,f}, F_{z,r}$ like in (4) as well as the lateral tire slip α_f, α_r for front and rear

$$\begin{aligned} \alpha_f &= \arctan\Big(\frac{v\sin(\beta) + \dot{\Psi} l_f}{v\cos(\beta)}\Big) - \delta \\ \alpha_r &= \arctan\Big(\frac{v\sin(\beta) - \dot{\Psi} l_r}{v\cos(\beta)}\Big), \end{aligned} \tag{6}$$

and the longitudinal tire slip s_f, s_r for front and rear

$$s_f = 1 - \frac{R_\omega \omega_f}{u_{\omega,f}}, \quad s_r = 1 - \frac{R_\omega \omega_r}{u_{\omega,r}}, \tag{7}$$

with the effective tire radius R_ω and the front and rear tire velocities $u_{\omega,f}, u_{\omega,r}$. The tire velocities can be computed by

$$\begin{aligned} u_{\omega,f} &= v\cos(\beta)\cos(\delta) + (v\sin(\beta) + l_f\dot{\Psi})\sin(\delta) \\ u_{\omega,r} &= v\cos(\beta). \end{aligned} \tag{8}$$

Since the dynamic single-track model as well as the single-track drift model become singular at low velocities, a distinction is mandatory. In this paper, the simulations of the physical models are based on the implementation of CommonRoad [33], where the dynamic single-track model is switched for velocities smaller than $0.1\ \mathrm{m\,s^{-1}}$. In comparison, the single-track drift model utilizes a more complex model blending [33]. The above equations describe only the pure modeling in the cases without singularity issues.

III. CONDITIONAL NEURAL PROCESSES (CNP)

While supervised learning operates on single datasets of single tasks and provides a task-specific predictor, meta-learning operates on datasets of multiple related tasks and provides an advanced predictor that shares knowledge across tasks to be able to predict even in unseen tasks. Since predictions represent a foundation for decision-making in trajectory planning, the consideration of model uncertainty is of critical interest. The neural process family [32] represents a group of models that belong to the Bayesian meta-learning domain and constitute an adequate solution. In the following, the conditional neural processes [31] as basic model of this group are described.

A. CNP Framework

Classical supervised learning approaches have a dataset $\mathcal{D} = \{(\boldsymbol{x}_n, \boldsymbol{y}_n)\}_{n=1}^N$ consisting of inputs $\boldsymbol{x}_n \in \mathcal{X} \subseteq \mathbb{R}^{d_{\boldsymbol{X}}}$ and outputs $\boldsymbol{y}_n \in \mathcal{Y} \subseteq \mathbb{R}^{d_{\boldsymbol{Y}}}$ and learn the input-output mapping function $f : \mathcal{X} \to \mathcal{Y}$ such that the final weights result in a minimum error over the entire dataset. Thus, a model is trained which approximates the process best on average. However, in the case that the data generating process itself is subject to variations due to changing conditions, the model would neglect them. To counteract this issue, a meta-learning dataset considers a finite set of input-output mapping functions $f : \mathcal{X} \to \mathcal{Y}$, sampled from a probability distribution P over functions. Each sampled function f_j leads to a task dataset $\mathcal{D}_j = (\mathcal{C}_j, \mathcal{T}_j)$ containing a labeled context set $\mathcal{C}_j = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^N$ and an unlabeled target set $\mathcal{T}_j = \{\boldsymbol{x}_i\}_{i=N+1}^{N+M}$. Thus, the meta-learning dataset $\mathcal{D}_{\mathcal{M}} = \{\mathcal{D}_j\}_{j=1}^K$ emerges as a set of K task-specific datasets.

The objective of the meta-learning procedure is to capture the stochastic process that generated a given context set $\mathcal{C}_j$ to predict the outputs corresponding to the target set $\mathcal{T}_j$. For this purpose, CNP maps the context set into embeddings

$$\boldsymbol{e}_{j,i} = \boldsymbol{\phi}_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{y}_i) \quad \forall (\boldsymbol{x}_i, \boldsymbol{y}_i) \in \mathcal{C}_j \tag{9}$$

using a learnable encoder neural network $\boldsymbol{\phi}_{\boldsymbol{\theta}} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^{d_{\boldsymbol{e}}}$. Here, the dimension of the embeddings $d_{\boldsymbol{e}}$ is a design parameter of fixed size. Furthermore, the task-agnostic embedding set $\mathcal{E}_j = \{\boldsymbol{e}_{j,i}\}_{i=1}^{N}$ is converted into a single representation

$$\boldsymbol{e}_j = \boldsymbol{a}(\mathcal{E}_j) = \boldsymbol{e}_{j,1} \oplus \boldsymbol{e}_{j,2} \oplus \cdots \oplus \boldsymbol{e}_{j,N-1} \oplus \boldsymbol{e}_{j,N} \tag{10}$$

employing an aggregation function. For better generalization, CNP require that the aggregation function performs commutative operations $\oplus$ to guarantee a single permutation invariant representation $\boldsymbol{e}_j$, according to Deep Sets [35]. In order to consider context datasets of flexible size, the average is usually taken. The encoded context is decoded together with the target set

$$\boldsymbol{d}_{j,i} = \boldsymbol{\rho}_{\boldsymbol{\theta}}(\boldsymbol{x}_i, \boldsymbol{e}_j) \quad \forall \boldsymbol{x}_i \in \mathcal{T}_j \tag{11}$$

by means of a learnable decoder neural network $\boldsymbol{\rho}_{\boldsymbol{\theta}} : \mathcal{X} \times \mathbb{R}^{d_{\boldsymbol{e}}} \to \mathbb{R}^{d_{\boldsymbol{d}}}$. In regression tasks, the decoding $\boldsymbol{d}_{j,i}$ is used to parameterize the mean $\mu_{j,i}(\boldsymbol{d}_{j,i})$ and variance $\sigma^2_{j,i}(\boldsymbol{d}_{j,i})$ of a Gaussian distribution for each $\boldsymbol{x}_i$ in $\mathcal{T}_j$. The simplest parameterization is the definition of a two-dimensional decoder output ($d_{\boldsymbol{d}} = 2$), which directly assigns a value to μ and σ^2. In addition, CNP are characterized by the fact that multilayer perceptrons (MLP) are used for the encoder-decoder structure.

More generally and compactly, the CNP models the conditional predictive distribution

$$p(f(\mathcal{T})|\mathcal{T}, \mathcal{C}) = p(f(\mathcal{T})|\boldsymbol{\rho}(\mathcal{T}, \boldsymbol{E}(\mathcal{C}), \boldsymbol{\theta}) \tag{12}$$

for a given context and target set by means of a decoder ρ and an encoding composition $\boldsymbol{E} = \boldsymbol{a} \circ \boldsymbol{\phi}$, which is a joint distribution over the random variables $f(\boldsymbol{x}_i)_{i=1}^{N+M}\}$. For regression tasks, the predictive distribution is modeled as a factorized Gaussian over the target set

$$p(f(\mathcal{T})|\mathcal{T}, \mathcal{C}) = \prod_{i=1}^{N+M} \mathcal{N}(\mu_i, \sigma_i^2). \tag{13}$$

Under this direct parameterization, mathematical guarantees of stochastic processes are traded off in favor of higher flexibility and scalability for conditional predictions, resulting in a complexity of $\mathcal{O}(N + M)$ for M predictions given N context pairs.

B. CNP based Yaw Rate Prediction

The framework of CNP offers the possibility to consider information of the current driving dynamics in the form of the context set $\mathcal{C}$ and to provide a distribution of predictions $p(f(\mathcal{T})|\mathcal{T}, \mathcal{C})$ depending on this context. Therefore, the dynamics of the yaw rate is interpreted as a stochastic process and the measured data from various circumstances, e.g. different dynamic ranges or weather conditions, as sampled functions $f \sim P$ from the distribution P of this stochastic process.

In this paper CarMaker[1] (CM) is used to generate the meta-learning dataset. The simulation tool CarMaker is widely used in the automotive industry due to its high-quality multi-body model and tire models [36], [37]. Data are selected in two urban cases, two interurban cases, two longitudinal dynamic cases and fourteen lateral dynamic cases to yield a balanced data set, which is a general requirement for accurate predictions [38]. All twenty cases are simulated in different weather conditions, simulated by different friction values ($\mu_{\mathrm{dry}} = 1.0, \mu_{\mathrm{wet}} = 0.5, \mu_{\mathrm{icy}} = 0.2$). Furthermore, the velocity is varied between $0\,\mathrm{and}\ 120\,\mathrm{km\,h^{-1}}$. As with real-world data, time series of different lengths occur depending on the test run. The used model builds up on the neural process family implementation[2] and extends the framework to use datasets of different length without any additional data selection procedure.

An overview of the architecture of the used CNP is listed in the Tab. I. There are two encoders listed, which represent the single encoder in Sec. III-A. Here, the inputs are encoded first which are further encoded with the targets in the second encoder. As the architecture reveals, an embedding size $d_{\boldsymbol{e}}$ of 64 is used. Inspired by the physical models, different

TABLE I: CNP Architecture.

General Modules	Network Architecture	Hidden Layers	Hidden Size	Activation Function
Feature Encoder	MLP	1	64	ReLU
Context Encoder	MLP	2	128	ReLU
Decoder	MLP	4	64	ReLU

input vectors $\boldsymbol{x}_i$ have been investigated for the prediction of the scalar output variable $y_i = \dot{\Psi}$. As a result, an input vector consisting of the steering angle δ, velocity v and the longitudinal acceleration a_{long} has emerged. The CNP is trained by randomly sampling task data sets from the metadata set, selecting context and target splits, passing the data through the architecture, calculating the loss, and performing a stochastic gradient update until it converges. As a loss function, the negative conditional log likelihood

$$\mathcal{L}(\boldsymbol{\theta}) = -\mathbb{E}_{f\sim P}\Big[\mathbb{E}_S[\log p_{\boldsymbol{\theta}}(\{\boldsymbol{y}_i\}_{i=1}^{S}|\{\boldsymbol{x}_i\}_{i=1}^{S}, \mathcal{C}_S)]\Big] \tag{14}$$

is minimized, where S denotes a subset of the sampled task-set. Since the CNP is trained offline, no gradient updates are necessary during the deployment, which enables fast predictions. The overall deployment workflow of the trained model is outlined in Fig. 1.

The learning-to-learn approach makes it possible to transfer the knowledge to previously unseen cases and thus offers a computationally efficient alternative to online learning. Due to the strict separation of learning and deployment, no improvement or adaptation can be achieved online, but at the same time catastrophic forgetting [39] is counteracted by having a temporally unchangeable system.

[1] https://ipg-automotive.com/en/products-solutions/software/carmaker/

[2] http://yanndubs.github.io/Neural-Process-Family/.

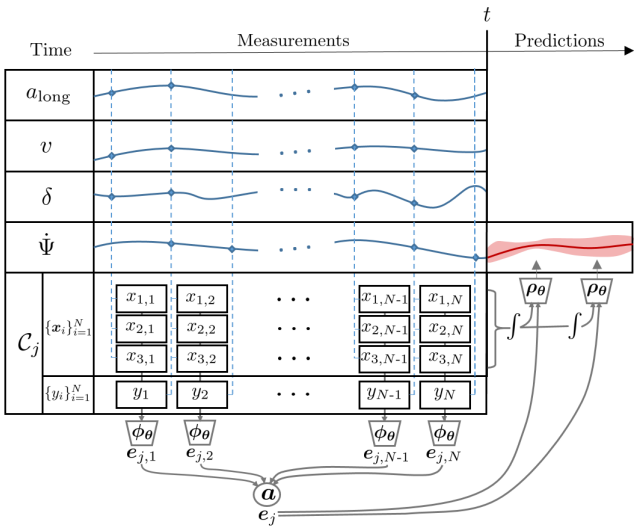


Fig. 1: CNP for yaw rate predictions.

IV. EVALUATION RESULTS

The methodology is evaluated on the basis of the wide range of scenarios that are also used for the training, but under different environmental and usage conditions. Due to modified conditions, these scenarios represent new samples of the underlying stochastic processes.

In order to investigate the adaptation to changing environmental conditions, variations of the friction coefficient are considered to simulate different road and weather conditions. With regard to changed operating conditions, a changed load (additional four persons, each 80 kg) is selected. For the purposes of generalization, further scenarios as well as the transfer to other vehicles are considered. The evaluation compares the CNP against the physical models, which are parameterized using the parameters of the CarMaker simulation software, which is used as reference.

A. Friction variation

The model is trained using data from 20 scenarios, each at different speeds and for friction coefficients 1.0, 0.5 and 0.2. In this evaluation, the trained model is executed in the same 20 scenarios, each across different velocities for friction coefficients 0.75, 0.35 and 0.1. The context set is based on measurements over a length of 10 % of each scenario. The target set is determined by Euler integration of the control variables over the length of the scenario. The CNP model is compared against the physical models, which also employ Euler integration for yaw rate prediction. We simulate on the one hand models without online parameter adaption (KST, DST, STD) and models with an ideal online parameter update (DST(μ), STD(μ)). The results in Fig. 2 show that the physical models perform slightly better than the CNP at a friction coefficient close to dry roads. However, the CNP performs better at lower friction coefficients. The DST model does not benefit from the knowledge of the friction coefficient, but the STD model does. Presumably, this can be attributed to drift behavior of the vehicle in highly dynamic scenarios. The CNP demonstrates consistent performance over the different friction coefficients. On average, only the ideal STD(μ) is superior, which has full knowledge about the current friction coefficient.

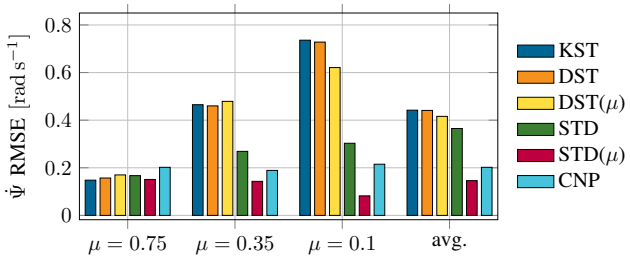


Fig. 2: Evaluation under changed friction coefficients.

B. Mass Variation

The effect of changed mass is evaluated similarly to the effect of changed friction. The mass was increased for this purpose, the friction remains the same as in the training. The ideal physical models (DST(μ, m), STD(μ, m)) use ground truth mass and friction values in this evaluation. Figure 3 shows that in the case of dry roads, simple models are slightly better. In the wet and icy road case, more complex models indicate significant advantages. Overall, it can be seen that mass is a less influential factor than friction, but this may be due to uniform load distribution. Again, it becomes apparent that only the ideal STD(μ, m) can benefit from the actual ground truth values. Similar to the first evaluation, the error of the CNP is relatively constant over the variations compared to other models. Again, on average only the STD(μ, m) model performs comparably.

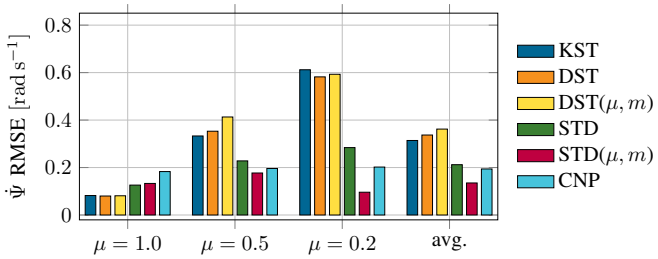


Fig. 3: Evaluation of prediction under varied mass. Results separated according to different road (friction) conditions.

C. Scenario variation

Unlike previous evaluations, this evaluation is based on scenarios that are not part of the training. Two urban scenarios, two interurban driving scenarios, two racetrack scenarios (Nordschleife Nürburgring, Hockenheimring) and one mountain pass scenario serve as the test cases. In all scenarios, velocity is varied appropriately. Friction parameters are likewise varied to simulate dry, wet and icy road surfaces. The numerical evaluation in Fig. 4 illustrates a similar pattern as in the previous evaluations. While CNP receives only the acceleration as an additional input compared to the simplest KST model, the error is comparable to complex physical models, which furthermore have the true parameters given. To complete the analysis, Fig. 5 shows the

prediction of the yaw rate for various friction coefficients for the Hockenheimring racetrack scenario. In the dry road domain, the CNP performs lowest while showing the highest uncertainty. In the area of wet and icy roads, the prediction is better while the uncertainty is quite low. The scenario on the icy road is significantly shorter, since the same driving style over different road parameters causes the vehicle to leave the track in this icy condition.

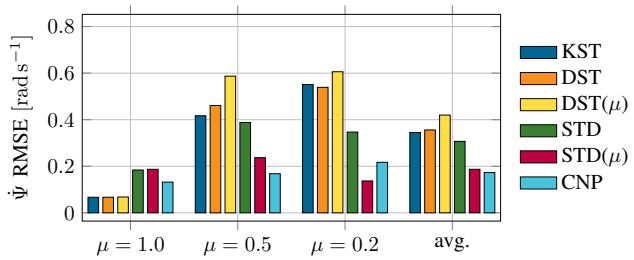


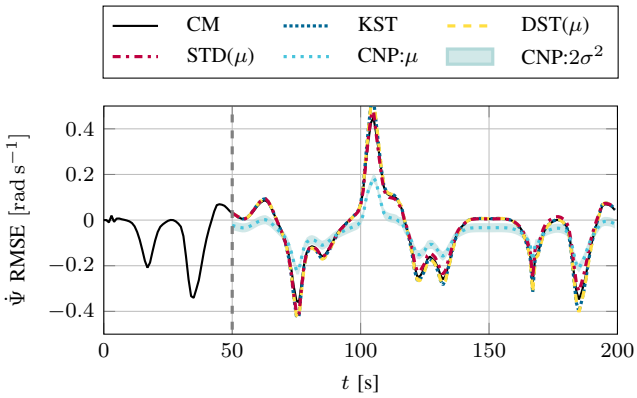
Fig. 4: Prediction evaluation under varied scenarios. Results separated according to different road (friction) conditions.

D. Vehicle variation

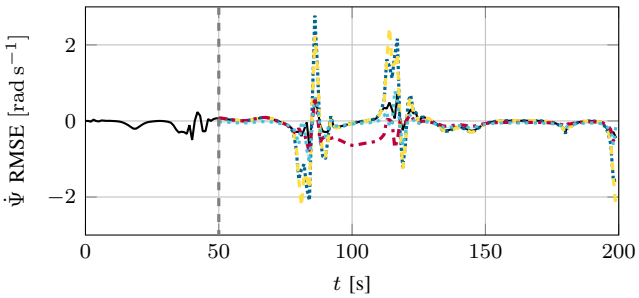
The training and evaluation up to now is based on the use of a VW Beatle, the CarMaker DemoCar. In this evaluation the transferability of the CNP to other vehicles is evaluated. Here, the evaluation from Section IV-C is applied to a small car (Honda Fit), an SUV (BMW X5), a van (VW T6) and a sports car (Porsche 911). The results for dry roads (Tab. II), wet roads (Tab. III), icy roads (Tab. IV) as well as average (Tab. V) demonstrate that the CNP performs similarly on other vehicles. This shows that the CNP is able to extract the characteristics of the present dynamics from a context dataset and to transfer learned knowledge to a large extent.

V. CONCLUSIONS AND FUTURE WORKS

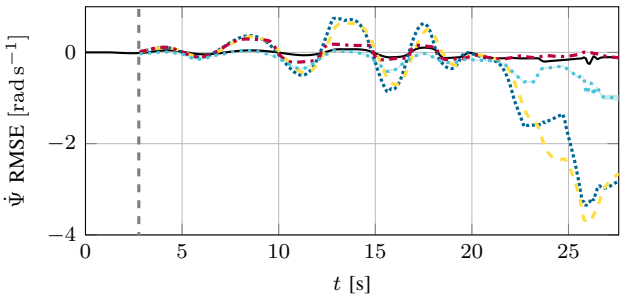
The investigation of the meta-learning approach conditional neural processes in terms of yaw rate prediction shows that the methodology is robust to environmental and operational variations and can even be transferred to other cars. It is also demonstrated that CNP offer a low mean prediction error while simultaneously maintaining a low computational complexity. Thus, a kind of model-order reduction of classical models is feasible. Moreover, the methodology also provides the possibility of uncertainty quantification, and more recent architectures of the neural process family are proven to yield improvement in this area. Future work remains to investigate the performance difference towards more advanced neural processes models, especially with respect to variance, which in general needs to be investigated in more detail. In addition, the impact of stochastic measurement inputs remains to be investigated. Ultimately, the behavior of a resulting stochastic hybrid vehicle model for trajectory planning in autonomous driving is to be investigated. While the methodology offers great advantages, and resolves e.g. out-of-distribution issues in a targeted manner, the safety assurance of AI will be a key research question.



(a) Dry road condtions ($\mu = 1.0$).



(b) Wet road condtions ($\mu = 0.5$).



(c) Icy road condtions ($\mu = 0.2$).

Fig. 5: Qualitative trajectory evaluation over (a) dry, (b) wet and (c) icy roads on the Hockenheim racetrack. The vertical dashed line separates the context set on the left from the predictions on the right. The time horizon in (c) is shorter since, given the same driving behavior, the vehicle leaves the road under icy conditions.

REFERENCES

[1] V. Milanés, S. E. Shladover, J. Spring *et al.*, "Cooperative adaptive cruise control in real traffic situations," *IEEE Trans. Intell. Transp. Syst. (T-ITS)*, vol. 15, no. 1, pp. 296–305, 2013.

[2] J. Funke, M. Brown, S. M. Erlien *et al.*, "Collision avoidance and stabilization for autonomous vehicles in emergency scenarios," *IEEE Trans. Control Syst. Technol. (TCST)*, vol. 25, no. 4, pp. 1204–1216, 2016.

[3] S. Wen, G. Guo, B. Chen *et al.*, "Cooperative adaptive cruise control of vehicles using a resource-efficient communication mechanism," *IEEE Trans. Intell. Veh. (T-IV)*, vol. 4, no. 1, pp. 127–140, 2018.

[4] K. Liu, J. Gong, A. Kurt *et al.*, "Dynamic modeling and control of high-speed automated vehicles for lane change maneuver," *IEEE Trans. Intell. Veh. (T-IV)*, vol. 3, no. 3, pp. 329–339, 2018.

[5] B. Paden, M. Čáp, S. Z. Yong *et al.*, "A survey of motion planning

TABLE II: RMSE Vehicle variation dry roads.

Vehicle	KST	DST	DST(μ)	STD	STD(μ)	CNP
Honda Fit	**0,112**	0,154	0,159	0,196	0,205	0,166
VW Beatle	**0,067**	**0,067**	0,068	0,184	0,187	0,132
BMW X5	0,107	0,172	0,098	0,253	**0,140**	0,150
VW T6	0,062	0,059	**0,028**	0,136	0,120	0,100
Porsche 911	**0,054**	0,077	0,071	0,201	0,151	0,142

TABLE III: RMSE Vehicle variation wet roads.

Vehicle	KST	DST	DST(μ)	STD	STD(μ)	CNP
Honda Fit	0,305	0,396	0,637	0,423	0,289	**0,178**
VW Beatle	0,417	0,461	0,587	0,388	0,237	**0,168**
BMW X5	0,226	0,296	0,283	0,334	**0,165**	0,191
VW T6	0,130	0,138	0,126	0,151	0,125	**0,114**
Porsche 911	0,306	0,364	0,440	0,644	0,331	**0,190**

TABLE IV: RMSE Vehicle variation icy roads.

Vehicle	KST	DST	DST(μ)	STD	STD(μ)	CNP
Honda Fit	0,660	0,620	0,657	0,387	**0,251**	0,260
VW Beatle	0,551	0,539	0,606	0,347	**0,137**	0,217
BMW X5	0,545	0,536	0,477	0,422	0,401	**0,284**
VW T6	0,348	0,339	0,280	0,325	0,244	**0,179**
Porsche 911	0,746	0,781	0,785	0,472	**0,254**	0,356

TABLE V: RMSE Vehicle variation on average.

Vehicle	KST	DST	DST(μ)	STD	STD(μ)	CNP
Honda Fit	0,359	0,390	0,484	0,336	0,248	**0,201**
VW Beatle	0,345	0,356	0,420	0,307	0,187	**0,173**
BMW X5	0,292	0,335	0,286	0,336	0,235	**0,208**
VW T6	0,180	0,179	0,145	0,204	0,163	**0,131**
Porsche 911	0,369	0,407	0,432	0,439	0,245	**0,229**

and control techniques for self-driving urban vehicles," *IEEE Trans. Intell. Veh. (T-IV)*, vol. 1, no. 1, pp. 33–55, 2016.

[6] R. Hult, F. E. Sancar, M. Jalalmaab *et al.*, "Design and experimental validation of a cooperative driving control architecture for the grand cooperative driving challenge 2016," *IEEE Trans. Intell. Transp. Syst. (T-ITS)*, vol. 19, no. 4, pp. 1290–1301, 2018.

[7] J. Betz, A. Wischnewski, A. Heilmeier *et al.*, "A software architecture for the dynamic path planning of an autonomous racecar at the limits of handling," in *Proc. IEEE Int. Conf. Connected Veh. Expo. (ICCVE)*, 2019, pp. 1–8.

[8] C. Pek and M. Althoff, "Computationally efficient fail-safe trajectory planning for self-driving vehicles using convex optimization," in *Proc. 21st IEEE Int. Conf. Intell. Transp. Syst. (ITSC)*, 2018, pp. 1447–1454.

[9] C. M. Kang, S.-H. Lee, and C. C. Chung, "Comparative evaluation of dynamic and kinematic vehicle models," in *Proc. 53rd IEEE Conf. Decis. Control (CDC)*, 2014, pp. 648–653.

[10] J. Kong, M. Pfeiffer, G. Schildbach *et al.*, "Kinematic and dynamic vehicle models for autonomous driving control design," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2015, pp. 1094–1099.

[11] J.-B. Tomas-Gabarron, E. Egea-Lopez, and J. Garcia-Haro, "Vehicular trajectory optimization for cooperative collision avoidance at high speeds," *IEEE Trans. Intell. Transp. Syst. (T-ITS)*, vol. 14, no. 4, pp. 1930–1941, 2013.

[12] D. N. Godbole, V. Hagenmeyer, R. Sengupta *et al.*, "Design of emergency manoeuvres for automated highway system: obstacle avoidance problem," in *Proc. IEEE Conf. Decis. Control (CDC)*, vol. 5, 1997, pp. 4774–4779.

[13] Z. Shiller, Y.-R. Gwo *et al.*, "Dynamic motion planning of autonomous vehicles," *IEEE Trans. Robot. Autom. (TRA)*, vol. 7, no. 2, pp. 241–249, 1991.

[14] J. Hwan Jeon, S. Karaman, and E. Frazzoli, "Anytime computation of time-optimal off-road vehicle maneuvers using the RRT," in *Proc. 50th IEEE Conf. Decis. Control (CDC) & Eur. Control Conf. (ECC)*, 2011, pp. 3276–3282.

[15] A. Chebly, R. Talj, and A. Charara, "Coupled longitudinal and lateral control for an autonomous vehicle dynamics modeled using a robotics formalism," *Proc. 20th IFAC World Congress (IFAC WC)*, vol. 50, no. 1, pp. 12 526–12 532, 2017.

[16] Í. B. Viana, H. Kanchwala, and N. Aouf, "Cooperative trajectory planning for autonomous driving using nonlinear model predictive control," in *Proc. IEEE Int. Conf. Connected Veh. Expo. (ICCVE)*, 2019, pp. 1–6.

[17] J. K. Subosits and J. C. Gerdes, "Impacts of model fidelity on trajectory optimization for autonomous vehicles in extreme maneuvers," *IEEE Trans. Intell. Veh. (T-IV)*, vol. 6, no. 3, pp. 546–558, 2021.

[18] Y. Du, C. Liu, Y. Song *et al.*, "Rapid estimation of road friction for anti-skid autonomous driving," *IEEE Trans. Intell. Transp. Syst. (T-ITS)*, vol. 21, no. 6, pp. 2461–2470, 2019.

[19] R. Matsuzaki, T. Keating, A. Todoroki *et al.*, "Rubber-based strain sensor fabricated using photolithography for intelligent tires," *Sens. Actuat. A*, vol. 148, no. 1, pp. 1–9, 2008.

[20] H. Lee and S. Taheri, "Intelligent tires? A review of tire characterization literature," *IEEE Intell. Transp. Syst. Mag. (ITSM)*, vol. 9, no. 2, pp. 114–135, 2017.

[21] F. Gustafsson, N. Persson, M. Drevö *et al.*, "Virtual sensors of tire pressure and road friction," in *Proc. Society Automotive Engineers World Congress (SAE WC)*, 2001.

[22] T. Nakatsuji, I. Hayashi, P. Ranjitkar *et al.*, "Online estimation of friction coefficients of winter road surfaces using the unscented Kalman filter," *Transp. Res. Rec. (TRR)*, vol. 2015, no. 1, pp. 113–122, 2007.

[23] H. Zhu, L. Li, M. Jin *et al.*, "Real-time yaw rate prediction based on a non-linear model and feedback compensation for vehicle dynamics control," *Proc. IMechE.*, vol. 227, no. 10, pp. 1431–1445, 2013.

[24] T. Lubiniecki and G. Schildbach, "Adaptive mpc for trajectory tracking with online adaption of the vehicle model's yaw intensification," in *Proc. IEEE Eur. Control Conf. (ECC)*, 2023, pp. 1–6.

[25] J. Kontos, B. Kránicz, and Á. Vathy-Fogarassy, "Prediction for future yaw rate values of vehicles using long short-term memory network," *Sensors*, vol. 23, no. 12, p. 5670, 2023.

[26] X. Jin, G. Yin, and N. Chen, "Advanced estimation techniques for vehicle system dynamic state: A survey," *Sensors*, vol. 19, no. 19, p. 4289, 2019.

[27] A. Rahman, J. Drgoňa, A. Tuor *et al.*, "Neural ordinary differential equations for nonlinear system identification," in *Proc. IEEE Am. Control Conf. (ACC)*, 2022, pp. 3979–3984.

[28] L. Hewing, J. Kabzan, and M. N. Zeilinger, "Cautious model predictive control using Gaussian process regression," *IEEE Trans. Control Syst. Technol. (TCST)*, vol. 28, no. 6, pp. 2736–2743, 2019.

[29] J. A. Matute-Peaspan, M. Marcano, S. Diaz *et al.*, "Lateral-acceleration-based vehicle-models-blending for automated driving controllers," *Electronics*, vol. 9, no. 10, p. 1674, 2020.

[30] T. Hospedales, A. Antoniou, P. Micaelli *et al.*, "Meta-learning in neural networks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, vol. 44, no. 9, pp. 5149–5169, 2021.

[31] M. Garnelo, D. Rosenbaum, C. Maddison *et al.*, "Conditional neural processes," in *Proc. 35th Int. Conf. Mach. Learn. (ICML)*. PMLR, 2018, pp. 1704–1713.

[32] S. Jha, D. Gong, X. Wang *et al.*, "The neural process family: Survey, applications and perspectives," *arXiv preprint arXiv:2209.00517*, 2022.

[33] M. Althoff, M. Koschi, and S. Manzinger, "CommonRoad: Composable benchmarks for motion planning on roads," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2017, pp. 719–726.

[34] H. B. Pacejka, *Tire and vehicle dynamics*. Oxfort: Butterworth-Heinemann, 2002.

[35] M. Zaheer, S. Kottur, S. Ravanbakhsh *et al.*, "Deep sets," *Proc. 31st Int. Conf. Neural Inf. Process. Syst. (NIPS)*, vol. 30, 2017.

[36] H. Holzmann and U. Wurster, "King of the HIL," in *Vehicle Dynamics International, Annual Showcase*, 2012, pp. 34–37.

[37] K. Tong, Z. Ajanovic, and G. Stettinger, "Overview of tools supporting planning for automated driving," in *Proc. IEEE 23rd Int. Conf. Intell. Transp. Syst. (ITSC)*, 2020, pp. 2950–2957.

[38] F. Mirus, T. C. Stewart, and J. Conradt, "The importance of balanced data sets: Analyzing a vehicle trajectory prediction model based on neural networks and distributed representations," in *Proc. Int. Jt. Conf. Neural Netw. (IJCNN)*. IEEE, 2020, pp. 1–8.

[39] A. Robins, "Catastrophic forgetting, rehearsal and pseudorehearsal," *Connection Science*, vol. 7, no. 2, pp. 123–146, 1995.