Receding horizon estimation and control with structured noise blocking for mobile robot slip compensation

Nathan D. Wallace, He Kong, Andrew J. Hill and Salah Sukkarieh

Abstract—The control of field robots in varying and uncertain terrain conditions presents a challenge for autonomous navigation. Online estimation of the wheel-terrain slip characteristics is essential for generating the accurate control predictions necessary for tracking trajectories in off-road environments. Receding horizon estimation (RHE) provides a powerful framework for constrained estimation, and when combined with receding horizon control (RHC), yields an adaptive optimisation-based control method. Presently, such methods assume slip to be constant over the estimation horizon, while our proposed structured blocking approach relaxes this assumption, resulting in improved state and parameter estimation. We demonstrate and compare the performance of this method in simulation, and propose an overlapping-block strategy to ameliorate some of the limitations encountered in applying noise-blocking in a receding horizon estimation and control (RHEC) context.

I. INTRODUCTION

In this paper, we implement a nonlinear receding horizon estimation and control (RHEC) approach for tracking trajectories in unknown and variable slip environments. Building upon insights in our prior work in RHEC [1]–[3] and especially the technique in [4], we investigate the impact of enforcing a noise-blocking structure on the estimated slip parameter sequence, in terms of both the estimation and tracking performance within the RHEC framework. We then propose an extension to this structured blocking (SB) strategy that reduces the periodic variation in parameter estimation quality that emerges from the enforced blocking by way of overlapping blocks. Alternate strategies for addressing this issue may also be found in our complementary work [5].

We compare our modified RHEC framework with the methodology in [6]–[9] which adopts an RHEC approach that fixes slip parameters to a single value over the estimation horizon—henceforth termed full-horizon blocking (FHB). We show that the proposed SB method exhibits both improved state and parameter estimation performance across a variety of different speeds (2–10 m/s). The impact of block size and overlap extent on performance is also evaluated.

II. RELATED WORK

Achieving robust control of wheeled mobile robots (WMR) in the presence of modelling errors and external disturbances is a long-standing challenge in robotics. Early works in this area study control in the presence of bounded additive disturbances [10]–[13], and while these techniques can be specialised for scenarios with slip, deriving the bounds for the uncertainties presents a challenge.

The recent literature on WMR control with slip can be classified into three main approaches; bounded uncertainty models, extended kinematic models (KM) which account for slip effects, and machine learning (ML) and dynamic models.

A. Bounded uncertainty models

Kinematic representations relating perturbations to vehicle slip are presented in [14] for four general WMR configurations, classifying slip perturbations as input-additive, input-multiplicative or matched/unmatched. These are utilised in [15] to design path following controllers which, assuming no other disturbances, apply backstepping techniques and prove the ultimate boundedness of the tracking errors.

In [16], the effects of wheel slip for a tractor are treated as bounded uncertainties, and a robust sliding-mode controller is designed to achieve stability with steering angle constraints. Similar ideas are adopted in [17]–[18] using tube-based RHC techniques for vehicle speeds of <1 m/s.

B. Extended kinematic models

Extended Kalman filters (EKF) with kinematic constraints have been used for slip estimation, both in [19], which models slip as a white-noise velocity perturbation, and in [20], which models longitudinal slip as a multiplicative factor within the KM for a skid-steered WMR. In [21] a generalized bicycle KM with an additional multiplicative factor models a tractor-trailer system with side slipping effects, using an EKF to estimate state and slip parameters, and a nonlinear RHC for path tracking. Lateral tracking errors of ∼15 cm on straight and gently curved paths are achieved at 3.3 m/s.

A comparable method for modelling slip is adopted in [6] for a tractor, augmenting a bicycle KM with two multiplicative factors to capture longitudinal and side slip, which are assumed constant over the estimation horizon. State and parameter constraints, and a nonlinear measurement function, necessitated use of the more advanced RHE strategy. An RHC controls the wheel velocity and steering rate, and this RHC framework is tested experimentally on wet and bumpy grass, with average tracking errors of 0.26 m on gentle curves, and 0.6 m in headland turns at ∼2 m/s.

Similar approaches appear in [7] and [8] for tracked and skid-steered WMRs, using a unicycle KM with factors for longitudinal and lateral slip, achieving tracking errors of ∼0.2 m/s for the tracked, and <0.3 m/s for the skid-steered robot. A tractor-trailer system was modelled in [9] using a generalised tricycle KM with three factors for longitudinal and slide slip. Centralised RHE was used, and the proposed distributed RHC framework is compared...
against a centralised strategy, achieving comparable tracking errors of \(<10\) cm with faster solution times.

Recently, the above RHEC framework was compared against a linear RHC approach in [22], using input-state linearisation (ISL) with an EKF estimator to compensate for system state nonlinearity. It is shown that RHEC produces better trajectory tracking performance than the ISL approach, albeit with a slight increase in computation cost.

In [23] a tractor is modelled by an extended bicycle KM with additive side slip effects on both wheels, ignoring longitudinal slip, and a nonlinear adaptive control law embedded with an RHC algorithm was designed to avoid tracking overshoot. A mixed backstepping kinematic and dynamic observer was later developed in [24] to improve slip observation at higher speeds—up to 5 m/s.

C. Machine learning or more complex dynamic models

ML methods learn slip models from prior experience and apply them predictively. Remote prediction using visual information [25]–[27] can learn nonlinear disturbance models which, combined with simple vehicle models, can achieve high performance path tracking at up to 2 m/s. However, learning controller robustness remains a largely unanswered question, though some works attempt to address this [28].

Reinforcement learning methods have been applied [29], extended to scenarios using more complex models as in [30]–[32], with great progress made in recent years in slip modelling and compensation for planetary exploration robots, such as the analysis and experimental validations in [31], [32]. ML methods learn slip models from prior experience and apply them predictively. Remote prediction using visual information [25]–[27] can learn nonlinear disturbance models which, combined with simple vehicle models, can achieve high performance path tracking at up to 2 m/s. However, learning controller robustness remains a largely unanswered question, though some works attempt to address this [28].

Reinforcement learning methods have been applied [29], extended to scenarios using more complex models as in [30]–[32], with great progress made in recent years in slip modelling and compensation for planetary exploration robots, such as the analysis and experimental validations in [31], [32].

There is also a large and rich literature on tire road interaction, dynamics and performance analysis (see e.g., [30]), with great progress made in recent years in slip modelling and compensation for planetary exploration robots, such as the analysis and experimental validations in [31], [32]. We remark that the proposed framework could be extended to scenarios using more complex models as in [30]–[32], although they require prohibitively large amounts of training data to sufficiently generalise the learnt model for operation.

There is also a large and rich literature on tire road interaction, dynamics and performance analysis (see e.g., [30]), with great progress made in recent years in slip modelling and compensation for planetary exploration robots, such as the analysis and experimental validations in [31], [32]. We remark that the proposed framework could be extended to scenarios using more complex models as in [30]–[32], although they require prohibitively large amounts of training data to sufficiently generalise the learnt model for operation.

III. PRELIMINARIES

A. Notation

Throughout this paper, \([a_1, \ldots, a_n]^T\) denotes \(\begin{bmatrix} a^1 & \cdots & a^n \end{bmatrix}^T\), where \(a_1, \ldots, a_n\) are appropriately dimensioned scalars/vectors/matrices. The weighted Euclidean norm is denoted by \(\|a\|_R^2 = a^T R a\).

B. System Model

For simplicity, the vehicle model used in this study is identical to that used in [6]; a slight adaptation of the conventional bicycle KM, which is known to approximate car-like vehicle behaviour quite accurately [33]–[35]. Slip is incorporated via two multiplicative parameters; one for longitudinal slip and another for side slip. The equations describing the motion are as follows:

\[
\begin{bmatrix}
\dot{x}_{pos} \\
\dot{y}_{pos} \\
\dot{\beta} \\
\dot{\delta}
\end{bmatrix} =
\begin{bmatrix}
\kappa u_1 \cos \beta \\
\kappa u_1 \sin \beta \\
\kappa \nu I \tan (\mu \cdot \delta) \\
u_2
\end{bmatrix},
\]

where \(x_{pos} [m]\) and \(y_{pos} [m]\) represent the position of the ‘rear wheel’—or centre of the rear axle in the case of a 4-wheeled vehicle—in global cartesian coordinates, \(\beta [\text{rad}]\) is the yaw angle or heading, and \(\delta [\text{rad}]\) is the steering angle; shown in Figure 1. The control variables are wheel speed with respect to the ground, \(u_1 [\text{m/s}]\), and the steering rate \(u_2 [\text{rad/s}]\). The parameter \(l [\text{m}]\) represents the wheel base.

The parameters \(\kappa, \mu \in [0, 1]\) represent the longitudinal and side slip respectively. \(\kappa\) relates commanded wheel speed to actual ground speed, and \(\mu\) results from inertia of the vehicle, which is approximately relative to the steering angle. The percentage slip experienced is therefore \(1 - \kappa\) and \(1 - \mu\).

IV. RECEIVING HORIZON EST. AND CONTROL

A RHEC strategy is adopted in this work, the structure of which is outlined in Figure 2. The nonlinear RHEC problem is formulated as follows. Let \(x_k \in \mathbb{R}^{n_x}\) denote the state of the system, \(y_k \in \mathbb{R}^{n_y}\) an observation, \(u_k \in \mathbb{R}^{n_u}\) the control actions and \(p_k \in \mathbb{R}^{n_p}\) the system parameters we wish to estimate. The system is assumed to evolve in accordance to the given dynamic model \(x_{k+1} = f(x_k, u_k, p_k)\), and observations are taken in accordance with the measurement model function \(y_k = h(x_k, u_k, p_k)\). Let variables \((x_k, y_k, u_k, p_k)\) refer to the real process. These each have associated decision and optimal decision variables in the optimisation, which we denote \((\chi_k, \eta_k, \rho_k, \hat{p}_k)\) and \((\hat{x}_k, \hat{y}_k, \hat{u}_k, \hat{p}_k)\) respectively.

At each sampling time \(t_k\), let the real system state be \(x_k\). A sensor measurement \(y_k\) is taken, and the RHE then uses the past \(N_e\) measurements \(y_j, \ j \in \mathcal{J}_k^{-N_e}\) to estimate \((x_k, p_k)\). This estimate \((\hat{x}_k, \hat{p}_k)\) is passed to the RHC module along with a time-based reference trajectory \(\lambda_j = [\lambda_j^x, \lambda_j^u]\), \(\lambda_j^x \subseteq \mathbb{R}^{n_x}, \lambda_j^u \subseteq \mathbb{R}^{n_u}, \ j \in \mathcal{J}_k^{k+1}\). The RHC module then computes the optimal control actions over the horizon \(N_c\), and the next action \(\hat{u}_k\) is sent to the robot for execution. This process repeats at each sampling time (\(\Delta t\) seconds apart) until the end of the trajectory is reached.

A. Nonlinear Receding Horizon Estimation

Receding horizon estimation (RHE) is a powerful optimisation-based estimation technique that provides a systematic framework for handling constraints. In contrast to full
information estimation (FIE), which utilises the entire history of available information, RHE uses only measurements taken within a finite time-frame, and captures the information of prior measurements in an arrival cost term.

Given \( N_e \) past measurements \( y_j, j \in \mathbb{J}_{k-N_e+1} \) and a measurement model function \( h \), the constrained estimation problem to be solved at time \( t_k \) is

\[
\min_{\chi(.), \nu(.), \rho(.)} \left\{ \Gamma_{k-N_e+1} + \sum_{i=k-N_e+1}^{k} \| \eta_i - y_i \|_{R_k}^2 \right\} \tag{2}
\]

\[
\text{s.t.} \quad \dot{x}_j = f(x_j, \nu_j, \rho_j) \quad \eta_j = h(x_j, \nu_j, \rho_j) \quad \chi_{\min} \leq x_j \leq \chi_{\max} \quad \rho_{\min,j} \leq \rho_j \leq \rho_{\max,j}
\]

where

\[
\Gamma_{k-N_e+1} = \left\| \chi_{k-N_e+1} - \hat{x}_{k-N_e+1} \right\|_{\Pi_{k-N_e+1}^{-1}}^2 \tag{3}
\]

is the arrival cost function, \( \hat{x}_{k-N_e+1} = \hat{x}_{k-N_e+1|k-N_e} \) is the optimal state prediction at \( t_{k-N_e+1} \), \( R_k \) is the symmetric positive semi-definite weighting matrix equal to the inverse of the measurement noise covariance matrix [36], and \( \Pi_{k-N_e+1}^{-1} \) is the inverted covariance matrix of a smoothed EKF [37].

**B. Structured noise blocking in estimation**

One of the main issues encountered with constrained estimation methods such as RHE, and especially FIE, is that an optimisation problem must be solved at each sampling time. This burden can be alleviated via reformulation as a multi-parametric quadratic program [38], or exploiting the optimisation structure [39].

A strategy called move blocking (MB) has seen prior application in RHC for a similar purpose [40]. This strategy involves constraining groups of adjacent-in-time predicted inputs to have the same value, curtailing the number of degrees of freedom in the optimisation problem to reduce complexity at the expense of optimality [41].

Recently, we have adopted a similar concept in FIE and RHE for estimating the process noise sequence (PNS) [4], and more importantly, we show that to ensure stability in RHE one has to enforce the same segment structure of the PNS enforced in MB FIE for the optimization steps within the receding horizon (i.e., steps between \( T-N \) and \( T-1 \)). The resulting MB RHE strategy becomes a dynamic estimator with a periodically varying computational complexity.

An illustrative example of this concept, henceforth termed structured blocking (SB), is shown in Figure 3.

The contribution of this work is to extend the noise-blocking concept to parameter and state estimation, examining the strategy within the context of RHEC for slip parameter estimation.

To that end, we can consider the slip parameters \( \kappa, \mu \) to be conceptually analogous to environmental noise. Thus, the estimator will choose values of these parameters such that the deviation between the estimated and measured state is minimised. However, the optimal choice of value for these parameters from an optimisation perspective is not necessarily reflective of the true value, as in the unblocked case it can result in overfitting to the noisy measurements.

In RHEC, we must be particularly attentive of this behaviour, as the controller relies on the estimated parameter values, which if incorrect may result in sub-optimal control decisions. The FHB approach in [6]–[8], where the slip parameter values are assumed constant over the estimation horizon, avoids this issue and produces good results at the speeds tested in those works—typically below 2.5 m/s. However, we conjecture that at higher speeds the state and parameter estimation accuracy of the FHB method will be compromised, since the rate of change in the observed parameter values increases with vehicle speed.

**Fig. 2.** Block diagram of the RHEC framework.
By applying SB with smaller intervals \((S < N_c)\), it is possible to obtain better parameter estimates than with FHB. However, by enforcing the necessary segment structure, the size of the terminal block varies from \([1, S]\), as seen in Figure 4. When this size is small, the parameter estimate encounters the same overfitting issue as in the unblocked case.

This led us to propose a strategy to address this issue; the virtual extension—or ‘overlapping’—of blocks. Functionally, this means we both suppress the existence of the last block in the horizon unless its size is at least equal to an overlap parameter \(o \in \mathbb{Z} \leq S\), and extend the prior block to encompass the overlap region \(s_j^t = a\Delta t\). An illustration of the SB with overlapping (SB-O) is shown in Figure 3 (c). This strategy reduces the variability in the terminal block’s parameter estimate, providing a more reliable estimate to the RHC module. Some alternative and complementary strategies for addressing this issue have also been presented in [5].

C. Nonlinear Receding Horizon Control

Receding horizon control (RHC) is an approach that seeks to predict the system behaviour over a finite time horizon via minimisation of a cost function composed of states, inputs and references, and similarly to RHE, the RHC framework also supports handling of state and input constraints.

For RHC, at the current time \(t_k\), we wish to predict the state-action-parameter sequence for the next \(N_c\) time steps; thus the constrained optimisation problem to be solved is:

\[
\min \chi(\cdot) \Omega(\cdot) \quad \left\{ \sum_{i=k+1}^{k+N_c} \chi_k^i - \chi_{k+N_c}^i \right\}^2 + \Omega_{k+N_c} \right\}
\]

\[
s.t. \quad \begin{cases} 
\dot{x}_j = f(x_j, u_j, \rho_j) \\
\chi_{\min} \leq x_j \leq \chi_{\max} \\
\nu_{\min} \leq u_j \leq \nu_{\max}
\end{cases} \quad j \in [k+1, N_c]
\]

where \(\Omega_{k+N_c} = \|\chi_k^{k+N_c} - \chi_{k+N_c}\|^2\)

is the terminal cost function, \(\chi_k^i \in \mathbb{R}^{n_x}\) the subset of the state variables \(\chi_k\) with corresponding entries in \(\chi_k\), and \(V_k, V_N\) the symmetric positive-semidefinite weighting matrices.

D. Solution Methods

As both RHE (2) and RHC (4) are nonlinear least squares optimisation problems, similar solution methods can be applied to both. Three popular approaches include simultaneous collocation, single-shooting, and multiple-shooting.
We implement and test the RHEC algorithm with the SB-O strategy for a four-wheeled car-like robot in simulation, comparing its performance with the FHB approach. The RHE and RHC modules are implemented in ACADO and MATLAB. For simplicity, we opt to use similar weightings as employed in [6]. The horizon length and step size used are $N_h = N_c = 15$ and $\Delta t = 0.2$ s respectively, and the steering angle is constrained to $-35^\circ \leq \delta \leq 35^\circ$.

A. RHE Configuration

Since the ACADO Code Generation tool does not yet support parameter estimation functionality, we instead augment the state and system model (1) with the slip parameter terms and two corresponding virtual controls, $u_\kappa$ and $u_\mu$.

In the FHB case, $u_\kappa$, $u_\mu$ are zero always, fixing the parameter value over the horizon. To implement SB-O, two blocking parameters $S$ and $o$ are introduced, specifying block size and overlap extent respectively. Each sampling time the bounds for $u_\kappa$, $u_\mu$ are updated to be zero everywhere except at the start of a new block, to both define and enforce the blocking structure. The overlap constraint is enforced by suppressing formation of new blocks after time $t_k - o \Delta t$.

It should be noted that persistent excitation is required to accurately estimate the traction parameters, since the mechanism by which these parameters are observed is proprioceptive in nature. As the robot’s speed and steering angle approach zero, these parameters will become unobservable.

The measurement function used in these simulations is:

\[ h(x,u) = [x_{pos} - d \cos(\delta), y_{pos} - d \sin(\delta), \delta, u_1, u_2]^T, \]

with $d = 0.4$ m being the offset of the GPS behind the rear axle centre, and we assume the same standard deviation in measurements as in [6]; $\sigma_x = \sigma_y = 0.03$ m for the GPS position measurement, $\sigma_\delta = 0.01745$ rad for the steering angle measurement, and $\sigma_{u_1} = 0.1$ m/s and $\sigma_{u_2} = 0.1$ rad/s for the actuation measurements. The RHE weighting matrix is $R_k = \text{diag} (\sigma_x^2, \sigma_y^2, \sigma_\delta^2, \sigma_{u_1}^2, \sigma_{u_2}^2)^{-1} \in \mathbb{R}^{5 \times 5}$.

B. RHC Configuration

Let us denote the ‘global’ space-based reference trajectory as $\Lambda_j = \begin{bmatrix} x_{ref,1}^{(j)}, y_{ref,1}^{(j)}, \kappa_{ref,1}^{(j)}, u_{1,ref}^{(j)}, u_{2,ref}^{(j)} \end{bmatrix}^T$, consisting of a sequence of waypoints augmented with a desired control reference. At each $t_k$, we generate a new ‘local’ time-based reference trajectory $\lambda$ from $\Lambda$, defined as per Section IV.

This is achieved by taking the current position estimate and finding the closest point $(\hat{x}_{ref}, \hat{y}_{ref})$ on the global trajectory. The next $N_c$ points along the trajectory, spaced equally $\Delta t \cdot \hat{a}_{ref}^k$ metres along the path, are then returned as the local trajectory $\lambda_k$.

The weighting matrices for the RHC step are $V_k = \text{diag} (1.0, 1.0, 5.0, 5.0)$ and $V_2 = 10 \cdot V_k$, and the steering rate constraints are $(-35^\circ/s \leq u_2 \leq 35^\circ/s)$.

VI. RESULTS

Simulations were run with the configuration outlined in Section V for a range of block sizes $S \in \mathcal{S}_{10}^{30}$ and overlap extents $o \in \mathcal{S}_{15}^2$ at each velocity $u_{1,ref} \in \mathcal{S}_{10}^{2}$ m/s. The computational effort of solving the estimation problem for the FHB and SB-O cases is compared in Figure 6, where the solution time for SB-O is seen to be competitive with the FHB method, with sub-millisecond scale solutions achieved for all except the unblocked case. While the results presented here focus primarily on higher-speed tracking, the SB-O method at lower speeds performs at least as well as the FHB method, with tracking performance competitive with [6]–[8].

A. State and Parameter Estimation Performance

The estimation performance of the SB-O and FHB methods is compared in Figure 7 for $S = 2$ and $o = 4$ on the figure-8 trajectory at 8 m/s. In the comparison shown, we chose to apply SB-O to $\kappa$, but not $\mu$, as the observability of $\mu$ is proportional to the steering angle, which we found did not vary sufficiently over the 3s horizon for SB to yield any improvements in estimation over FHB in the tests conducted. We do see, however, that applying SB-O to $\kappa$ significantly improves the quality of the $\kappa$ estimate, while a competitive estimation quality is maintained for $\beta$ and $\mu$.

The mean average estimation errors for position and the parameter values over the full range of $(S,o)$ values tested on the sawtooth trajectory at 8 m/s are shown in Figure 8 (b)–(d) for the case where SB-O is applied to both $\kappa$ and $\mu$, and in Figure 9 (b)–(d) for the case where SB-O is applied to $\kappa$ while FHB is applied to $\mu$.
Fig. 8. Performance comparison of SB-O method (applied to $\kappa$ and $\mu$) vs FHB as function of block size $S$ and overlap extent $o$ for the sawtooth trajectory at 8 m/s. Mean average error for (a) trajectory tracking, (b) longitudinal slip $\kappa$, (c) position estimate and (d) side slip $\mu$. Note that for $S = \{1, 3\}, o < 2$ the estimator did not converge.

Fig. 9. Performance comparison with setup as per Figure 8, except SB-O is applied to $\kappa$ only ($\mu$ uses FHB in both).

Fig. 10. Performance comparison for figure-8 trajectory at 8 m/s, SB-O is applied to $\kappa$ only ($\mu$ uses FHB in both).

Fig. 11. Percentage reduction in tracking error at each tested velocity.

We observe that position and $\kappa$ estimation quality improved for most choices of $(S, o)$. Increasing $o$ was seen to reduce parameter estimation errors, particularly for smaller block sizes where the periodic variation in estimate quality would have compromised the parameter estimate. Blocking both parameters was observed to reduce estimation quality of $\mu$ for smaller $o$, which we believe is due to its reduced observability. This is supported by the results in Figure 9, where FHB is applied to $\mu$—this unambiguously improved both position and $\kappa$ estimates. Similar results are seen in Figure 10 for the figure-8 trajectory.

B. Tracking Performance

The tracking performance shown in Figures 8–10 (a) indicates that smaller $S$ and $o$ correlates with increased tracking error, which is expected based on the discussions in Section IV-B. The tracking error is also observed to improve proportionally with estimation quality, though with some additional variance, which we suspect is due to the influence of the $\beta$ and $\delta$ estimates, which were not included in these figures. For the figure-8 case, the tracking performance is close, but slightly worse on average, despite better estimation performance, though whether this remains the case in practice will be subject to further investigation.

A summary of the performance improvement of SB-O over FHB is shown in Figure 11, for the best performing $(S, o)$ at each speed. The tracking quality is observed to consistently surpass the FHB method by $\sim 3$–$6\%$ at all tested speeds. For a more detailed performance analysis of both the proposed SB-O method and other alternative SB strategies, we refer the reader to [5].

VII. CONCLUSION

A RHEC framework has been implemented and tested in simulation for a car-like robot using an augmented bicycle kinematic model that accounts for slip. SB has been investigated as an alternative to FHB, better capturing variations in parameter values, and an overlapping-block strategy has been proposed to manage variations in parameter estimate quality due to the enforced blocking structure. The performance of SB-O has been assessed for a range of block-sizes $S$ and overlaps $o$, and compared against FHB methods, where it yielded improved estimation and tracking performance with competitive computational efficiency. Future work will involve applying this strategy with a more complex kinematic model—a 4 wheel independent steer/drive field robot—both in a higher-fidelity dynamic simulation environment, and experimentally on the physical platform. It is also planned to further investigate the relationship between $S, o$ and the temporal frequency of the estimated parameters’ variation.
