Appendix

A. Factor-Graph Formulation of HF [19]

As described in Section II-C2 the utilized state estimator builds on HF [19], which formulates estimation as a factor-graph Maximum a Posteriori (MAP) problem

$$\mathcal{X}^\star \; = \; \arg\max_{\mathcal{X}} \, p(\mathcal{X} \mid \mathcal{Z}) \; \propto \; p(\mathcal{Z} \mid \mathcal{X}) \, p(\mathcal{X}), \tag{1}$$

where $\mathcal{Z}$ collects all sensor measurements and priors, and $\mathcal{X}$ comprises the robot and alignment variables

$$\mathcal{X} \; = \; \{ {}^{I}\mathcal{X}_{N_I}, \; {}^{G}\mathcal{X}_{N_G}, \; {}^{R}\mathcal{X}_{N_R} \}. \tag{2}$$

Here, ${}^{I}\mathcal{X}_{N_I}$ denotes the time-indexed robot state in the world frame, while ${}^{G}\mathcal{X}_{N_G}$ and ${}^{R}\mathcal{X}_{N_R}$ parameterize the global and reference-frame alignments that couple the local map with the global reference frame. The two design choices introduced in Section II-C2, namely absolute LiDAR-SLAM pose factors and explicit estimation of the map-to-world alignment, act on ${}^{R}\mathcal{X}_{N_R}$ and on the corresponding factors in this graph. This makes the factor graph robust to drifts in LiDAR-SLAM and misalignments arising from measurement uncertainties.

B. Curvature-Adaptive Lookahead Sampling

This appendix gives the full explanation of the lookahead sampling scheme summarized in Section II-C4. The procedure is illustrated in Figure 7.

The robot position $\mathbf{x}_{\text{robot}}$ is first projected onto the global path to obtain the closest anchor point $\mathbf{x}_i$ at index i. Two secant vectors $\mathbf{v}_i^{(p)}$ and $\mathbf{v}_i^{(n)}$ are then constructed, pointing forward and backward along the path from the anchor. Their normalized directions $\hat{\mathbf{v}}_i^{(p)}$ and $\hat{\mathbf{v}}_i^{(n)}$ define the local curvature κ_i as

$$\kappa_i = \frac{\theta_i}{\|\hat{\mathbf{v}}_i^{(n)}\|}, \qquad \text{with } \theta_i = \arccos\big(\hat{\mathbf{v}}_i^{(p)} \cdot \hat{\mathbf{v}}_i^{(n)}\big), \tag{3}$$

where θ_i is the angular difference between the two vectors. The adaptive lookahead distance follows as

$$L(\kappa_i) = L_{\min} + \frac{L_{\max} - L_{\min}}{1 + |\kappa_i|/\kappa_{\text{ref}}}, \tag{4}$$

with $L_{\min}$ and $L_{\max}$ the minimum and maximum lookahead distances, and κ_{ref} controlling the sensitivity to curvature changes. Equation (4) decreases monotonically in $|\kappa_i|$, so that straight path segments yield a long lookahead and tight turns a short one.

Given $L(\kappa_i)$, we place N equally spaced samples along the path ahead of the anchor point. The lookahead waypoint $\mathbf{x}_{\text{goal}}$ is obtained as an exponentially weighted average of these samples with respect to their arc-length distance from the anchor, which emphasizes the nearby path geometry while still accounting for the structure of the upcoming trajectory.

C. Autonomy Metrics

The two autonomy metrics reported in Table I are defined as follows. The Robot Attention Demand [24] is

$$\text{RAD} = \frac{IE}{IE + NT}, \tag{5}$$

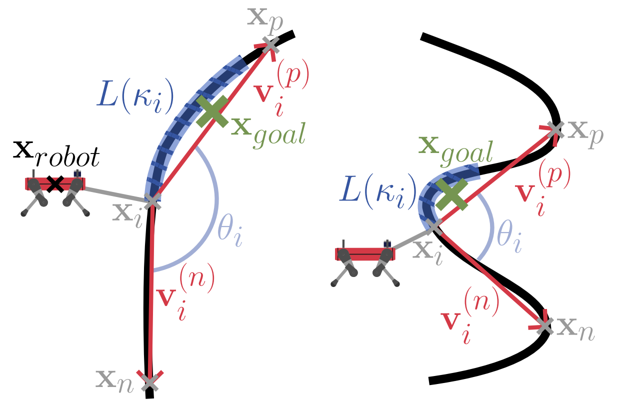

Fig. 7. Computation of the curvature-based adaptive lookahead. The robot position ($\mathbf{x}_{robot}$) is projected onto the path to obtain the nearest anchor point ($\mathbf{x}_i$), from which forward and backward secant vectors $\big(\mathbf{v}_i^{(p)}, \mathbf{v}_i^{(n)}\big)$ define the local curvature (κ_i). The lookahead waypoint ($\mathbf{x}_{\text{goal}}$) is then placed according to the curvature-dependent lookahead distance ($L(\kappa_i)$).

where the interaction effort IE denotes the average duration of operator interventions and the neglect tolerance NT the average interval between them. Lower values indicate that the robot required less operator attention per unit of autonomous operation. The Autonomy Rate expresses the fraction of total mission time during which the robot operated without human input,

$$\text{AR} = 100 \times \left(1 - \frac{T_{\text{intervention}}}{T_{\text{total}}}\right), \tag{6}$$

given in percentage.