Towards Scalable Coverage-Based Testing of Autonomous Vehicles - Supplementary Materials

Anonymous Author(s)
Affiliation
Address
`email`

1 Additional Experiments

Ablation on GP Kernels: While GUARD requires minimal tuning of GP kernel parameters due to online kernel learning, the *type* of kernel is still a design choice. In GUARD we employ Matern32 kernels to model each parameter and take the product of individual kernels to obtain the final kernel. In Table 1 we conduct an ablation that also considers 1) RBF kernels, 2) Using sum instead of product, and 3) no composition, using one RBF or Matern32 kernel for all parameters. We first find that without modeling each parameter separately, the results are poor. Also, using a sum operation to aggregate kernels is ineffective as well. The best results are obtained from a product of RBF or Matern32 kernels. We choose product of Matern32 as the lower false positive rate is critical for ensuring safety. Furthermore, this choice also exhibits much lower variance in the metrics, making it more reliable for testing.

BO Acquisition Functions: As mentioned in Section 2 of the main manuscript, many adversarial example generation methods use Bayesian Optimization(BO) [1] and employ acquisition functions that minimize some adversarial objective. Since BO also leverages GPs and active learning similar to GUARD, we can adopt these acquisition functions. In particular we use lower confidence bound (LCB), expected improvement (EI), and probability of improvement (PI). For EI and PI we adjust them to decrease the objective (expected decrease and probability of decrease). The results in Table 2 show that these alternatives can achieve strong results but is not as effective as using the Straddle acquisition function. This is because cost minimization oversample very negative points and Straddle oversamples boundary points, but the former is more useful in partioning pass / fail regions. The performance is however very close, and we hypothesize this is because the parameter space is dominated by positive (passes). With very small negative regions, severe negatives are close to boundary points.

Runtime Comparison A major implementation detail regarding our baselines is the discretization resolution. For grid search and t-way testing, there is a direct tradeoff between coverage and balanced accuracy, error recall, false positive rate. Thus, we select a resolution which can cover most of the parameter space. For HiddenGems [2], the resolution does not affect coverage since it uses a levelset estimation [3] algorithm to determine which bins are covered. The limitation for our choice of resolution of 6 is due to how the computation budget increases exponentially with resolution. This is because the method requires running the GP on all discretized bins at every iteration. And while running the GP a single time is neglible compared to running simulation, the exponential scaling makes the GP queries a bottleneck at higher resolutions. As we show in Figure 1, at this resolution the runtime roughly equals GUARD.

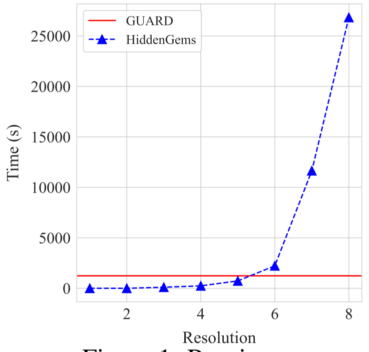

Figure 1: Runtime

2 Algorithm Details

We provide a more detailed overview of the algorithm used for GUARD in Algorithm 1. The $\arg\max$ operations are done via gradient based optimization as described in the main manuscript.

Kernel Type	Composition	Coverage(%)	Bal. Acc(%)	Pos Acc(%)	Neg Acc(%)	Err. Recall(%)	FPR(%)
RBF	-	96.1 ± 0.8	80.2 ± 5.7	98.1 ± 0.5	62.4 ± 11.2	56.0 ± 10.6	10.6 ± 3.3
Matern32	-	87.3 ± 1.5	78.5 ± 4.2	**99.3 ± 0.2**	57.7 ± 8.4	36.8 ± 7.7	6.15 ± 1.1
RBF	Sum	88.0 ± 6.7	58.6 ± 3.2	89.1 ± 11.4	28.2 ± 11.5	21.4 ± 12.0	20.2 ± 5.7
Matern32	Sum	90.4 ± 3.3	58.4 ± 3.3	88.0 ± 10.7	28.7 ± 13.7	23.6 ± 12.7	21.0 ± 3.1
RBF	Product	**96.8 ± 0.7**	83.3 ± 7.0	97.9 ± 0.6	68.7 ± 13.9	**62.6 ± 13.1**	7.83 ± 4.0
Matern32	Product	94.3 ± 0.9	**84.2 ± 5.3**	99.2 ± 0.2	**70.3 ± 10.6**	58.9 ± 5.3	**5.89 ± 3.1**

Table 1: Ablation of different kernel choices for GUARD

Aquisition Fn.	Coverage(%)	Bal. Acc(%)	Pos Acc(%)	Neg Acc(%)	Err. Recall(%)	FPR(%)
PI	90.7 ± 1.6	82.2 ± 3.9	99.4 ± 0.1	64.9 ± 7.8	49.0 ± 7.5	8.23 ± 2.4
EI	91.8 ± 1.3	82.6 ± 2.3	99.4 ± 0.2	65.7 ± 0.5	50.7 ± 5.2	6.99 ± 1.4
LCB	91.5 ± 1.5	83.4 ± 1.8	**99.6 ± 0.1**	67.3 ± 3.6	52.4 ± 4.7	5.99 ± 1.4
Straddle	**94.3 ± 0.9**	**84.2 ± 5.3**	99.2 ± 0.2	**70.3 ± 10.6**	**58.9 ± 5.3**	**5.89 ± 3.1**

Table 2: Comparison of cost minimization acquisition functions versus Straddle acquistion function.

Algorithm 1: Detailed Algorithm for GUARD

```
Input: Test function $f^*$, test Threshold $\gamma$, budget $N$, initial batch size $M$, initial kernel parameters $\boldsymbol{\kappa}_0$, kernel update frequency $K$, Gaussian CDF $\Phi(\cdot)$
$G \leftarrow GP(\boldsymbol{\kappa}_0)$                                   ▷ initialize GP
$\mu(\cdot), \sigma(\cdot) \leftarrow$ mean of $G$, standard deviation of $G$
for $t$ in 1, ..., $M$ do
    $\boldsymbol{\theta}_t \leftarrow \text{argmax}_{\boldsymbol{\theta}}\, \sigma(\boldsymbol{\theta})$                ▷ Initial batch for exploration
    $G \leftarrow \textsc{UpdateGPSamples}(G, \boldsymbol{\theta}_t, f^*(\boldsymbol{\theta}_t))$
end
$\boldsymbol{\kappa}_M \leftarrow \text{argmax}_{\boldsymbol{\kappa}}\, P(f(\boldsymbol{\theta}_1)\dots f(\boldsymbol{\theta}_t) \mid \boldsymbol{\theta}_1 \dots \boldsymbol{\theta}_t; \boldsymbol{\kappa})$   ▷ Probability under GP prior
$G \leftarrow \textsc{UpdateGPKernel}(G, \boldsymbol{\kappa}_M)$
for $t$ in M+1, ..., $N$ do
    $\boldsymbol{\theta}_t \leftarrow \text{argmax}_{\boldsymbol{\theta}}\, \beta\sigma(\boldsymbol{\theta}) + |\mu(\boldsymbol{\theta}) - \gamma|$      ▷ Explore or be close to boundary
    $G \leftarrow \textsc{UpdateGP}(G, \boldsymbol{\theta}_t, f^*(\boldsymbol{\theta}_t))$
    if $t \equiv 0 \mod K$ then
        $\boldsymbol{\kappa}_t \leftarrow \text{argmax}_{\boldsymbol{\kappa}}\, P(f(\boldsymbol{\theta}_1)\dots f(\boldsymbol{\theta}_t) \mid \boldsymbol{\theta}_1 \dots \boldsymbol{\theta}_t; \boldsymbol{\kappa})$ $G \leftarrow \textsc{UpdateGPKernel}(G, \boldsymbol{\kappa}_t)$
    end
end
function $\textsc{PassFailEstimation}(\boldsymbol{\theta}, \alpha)$ is
    $p^+ \leftarrow \Phi\left(\frac{\mu(\boldsymbol{\theta})-\gamma}{\sigma(\boldsymbol{\theta})}\right)$        ▷ Probability value at $\boldsymbol{\theta}$ is greater than threshold
    $p^- \leftarrow \Phi\left(\frac{\gamma-\mu(\boldsymbol{\theta})}{\sigma(\boldsymbol{\theta})}\right)$         ▷ Probability value at $\boldsymbol{\theta}$ is less than threshold
    if $p^+ \geq \alpha$ then
        return $1$                                                  ▷ Pass
    else if $p^- \geq \alpha$ then
        return $-1$                                                 ▷ Fail
    else
        return $0$                                                  ▷ Uncertain
end
return $\textsc{PassFailEstimation}$
```

3 Test Scenarios

Details on our parameterized scenarios are provided in Table 3 where we give a high level description of each scenario, associated parameters configuring the traffic agents or road, and parameter bounds.

Scenario	Parameter	Bounds
AV follows lane, actor cuts in from neighboring lane	AV initial speed	[20, 40] m/s
	Actor initial speed relative to AV	[-10, 10] m/s
	Cut-in lane change duration	[1, 10] s
	Time to collision at initialization assuming constant speed	[1, 6] s
	Road curvature	[-0.002, 0.002] m^{-1}
AV follows lane, multiple actors merge in from on-ramp	AV time-to-arrival to merge point	[0.5, 5.0] s
	AV initial speed relative to speed limit	[-10.0, 10.0] m/s
	Actors speed relative to AV	[-10.0, 10.0] m/s
	Time interval between actors	[1.0, 4.0] m/s
	Actors time-to-arrival to merge point	[0.5, 5.0] s
AV follows lane, lead actor brakes	Time-to-collision at initialization assuming constant speed	[1.0, 10.0] s
	Actor initial distance relative to AV	[20.0, 200.0] m
	AV initial speed relative to speed limit	[-10.0, 0.0] m/s
	Actor target speed relative to AV initial speed	[-30.0, 10.0] m/s
	Road curvature	[-0.003, 0.003] m^{-1}
AV follows lane, actor overtakes from neighboring lane	Actor initial follow distance	[1.0, 30.0] m/s
	Actor headway for lane change	[-10.0, 10.0] m
	Actor overtake acceleration	[0.1, 5.0] m/s^2
	Actor lane change duration	[0.5, 3.0] s
	Road curvature	[-0.002, 0.002] m^{-1}
AV follows lane, actor cuts in from shoulder	Actor initial speed relative to AV	[-29.0, -15.0] m/s
	Actor initial distance from lane boundary	[0.1, 0.3] m
	Time-to-collision at initialization assuming constant speed	[1.0, 6.0] s
	Actor cut-in duration	[1.0, 8.0] s
	Road curvature	[-0.002, 0.002] m^{-1}
AV merges from on-ramp, with an actor on its target lane	AV time-to-arrival to merge point	[0.5, 6.0] s
	Actor time-to-arrival to merge point	[0.5, 6.0] s
	AV initial speed relative to speed limit	[-10.0, 10.0] m/s
	Actor initial speed relative to AV initial speed	[-10.0, 10.0] m/s
	Lane encroachment threshold for actor to react to AV	[-5.0, 5.0] m
AV merges from parallel on-ramp, with an actor on its target lane	AV time-to-arrival to merge point	[0.5, 6.0] s
	Actor time-to-arrival to merge point	[0.5, 6.0] s
	AV initial speed relative to speed limit	[-10.0, 10.0] m/s
	Actor initial speed relative to AV initial speed	[-10.0, 10.0] m/s
	Lane encroachment threshold for actor to react to AV	[-5.0, 5.0] m
AV merges from on-ramp, with multiple actors on its target lane	AV time-to-arrival to merge point	[0.5, 6.0] s
	Actors time-to-arrival to merge point	[-8.0, 4.0] s
	AV initial speed relative to speed limit	[-10.0, 10.0] m/s
	Actors initial speed relative to AV initial speed	[-10.0, 10.0] m/s
	Time interval between actors	[1.0, 6.0] s
AV changes lane, with an actor on its target lane	Road curvature	[-0.002, 0.002] m^{-1}
	Actor initial speed relative to AV initial speed	[-10.0, 10.0] m/s
	AV initial speed relative to speed limit	[-10.0, 10.0] m/s
	Time-to-collision at initialization assuming constant speed	[1.0, 15.0] s
	Lane encroachment threshold for actor to react to AV	[-2.0, 2.0] m
AV changes lane, with multiple actors merging from on-ramp	AV time-to-arrival to merge point	[0.5, 6.0] s
	Actor time-to-arrival to merge point	[0.5, 6.0] s
	AV initial speed relative to speed limit	[-10.0, 10.0] m/s
	Actor initial speed relative to AV initial speed	[-10.0, 10.0] m/s
	Time interval between actors	[1.0, 6.0] s

Table 3: Test scenario details

References

[1] A. Agnihotri and N. Batra. Exploring bayesian optimization. *Distill*, 2020. doi:10.23915/distill.00026. https://distill.pub/2020/bayesian-optimization.

[2] A. Petrov, C. Fang, K. M. Pham, Y. H. Eng, J. G. M. Fu, and S. D. Pendleton. Hiddengems: Efficient safety boundary detection with active learning. *arXiv preprint arXiv:2210.13956*, 2022.

[3] A. Gotovos. Active learning for level set estimation. Master's thesis, Eidgenössische Technische Hochschule Zürich, Department of Computer Science,, 2013.