# GOODE: A Gaussian Off-The-Shelf Ordinary Differential Equation Solver - Supplements

**David N. John** [1 2] **Vincent Heuveline** [2] **Michael Schober** [3]

## Abstract

We present supplementary material for the paper *GOODE: A Gaussian Off-The-Shelf Ordinary Differential Equation Solver* (John et al., 2019).

## 1. Extension on Solver Comparison

This section presents additional material for comparison of the algorithms used in Section 5 of John et al. (2019). GOODE is used with a squared exponential kernel. An important criterion on selecting a numerical method is the computational effort. Figure 1 displays the run times of the different algorithms compared in Section 5. For GOODE the overall runtimes including hyperparameter optimization via grid search with $M$ grid points are plotted. Additionally, we display these runtimes scaled by $1/M$. The scaled runtimes of GOODE are comparable to the other algorithms. As the overall runtime scales linearly with $M$, it is by magnitude $M$ larger as the runtimes of the other solvers, but still within the range of a few seconds. Note that overall runtime for GOODE might be reduced by a smaller $M$ or another hyperparameter optimization method, in particular if hyperparameters are adapted online.

Note that the comparison of methods is *not* on equal footing. The other codes are highly tuned and feature mesh adaptation strategies tailored to the specific method. GOODE, as presented in the main text, is a proof-of-concept implementation of an elementary method without these features. Therefore, if the runtimes are on the same order of magnitude, it can be expected that future improvements to GOODE will achieve comparable runtimes at higher accuracy on adaptive meshes.

Thus, a comparison of the mesh, i.e. number of mesh points $N$ and minimum mesh width $\delta := \min\{|t_1 - t_2| : t_1, t_2 \in \Delta, t_1 \neq t_2\}$ are presented in Figure 2 and Figure 3. In general, we use a larger number of mesh points $N$ for GOODE in comparison to the other solvers, but with an equidistant mesh. Now, comparing with respect to $\delta$, GOODE operates in general within the same order of magnitude as the other solvers.

Remark: For Problem 23 all other solver have a significantly smaller stepsize than we use for GOODE, that explains why GOODE performs worse for problem 23 considering the relative error.

## 2. GOODE with Matérn 5/2 kernel

Here, we extend the comparison of GOODE with other solvers, by using a Matérn 5/2 kernel (consult Rasmussen & Williams (2006) for the definition). A comparison of GOODE, with Matérn 5/2, to the other solvers, with respect to the relative error, is displayed in Figure 4. The results are obtained in the default setting, but with $N = 71$ and grid search for the hyper-parameter of Matérn 5/2 in $[1.5h, 150h]$. Figure 5 presents the difference of selecting the length scale as global optimum based on the reference or based on maximizing the log likelihood.

## 3. GOODE with the **bvp5c** mesh

As a preliminary study for the performance of GOODE with a non-equidistant mesh $\Delta$, we use the resulting mesh from `bvp5c` for a specific problem and approximate the same problem with GOODE with exactly the same mesh. However, for a low number $N$ of points in the `bvp5c` mesh, hyper-parameter optimization for the squared exponential kernel with global length scale seems to be problematic. Thus, we increase for each solver the relative and absolute tolerance to 1e-5 and 1e-8, respectively, in order to obtain a finer mesh. Everything else is in the default setting.

A comparison of GOODE, with the `bvp5c` mesh, to the other solvers, with respect to the relative error, is displayed in Figure 6. Figure 7 presents the difference of selecting the length scale as global optimum based on the reference or based on maximizing the log likelihood.

[1]Corporate Research, Robert Bosch GmbH, Renningen, Germany [2]Engineering Mathematics and Computing Lab, Interdisciplinary Center for Scientific Computing, Heidelberg University, Germany [3]Bosch Center for Artificial Intelligence, Renningen, Germany. Correspondence to: David N. John <david.john@de.bosch.com>.

*Proceedings of the $36^{th}$ International Conference on Machine Learning*, Long Beach, California, PMLR 97, 2019. 

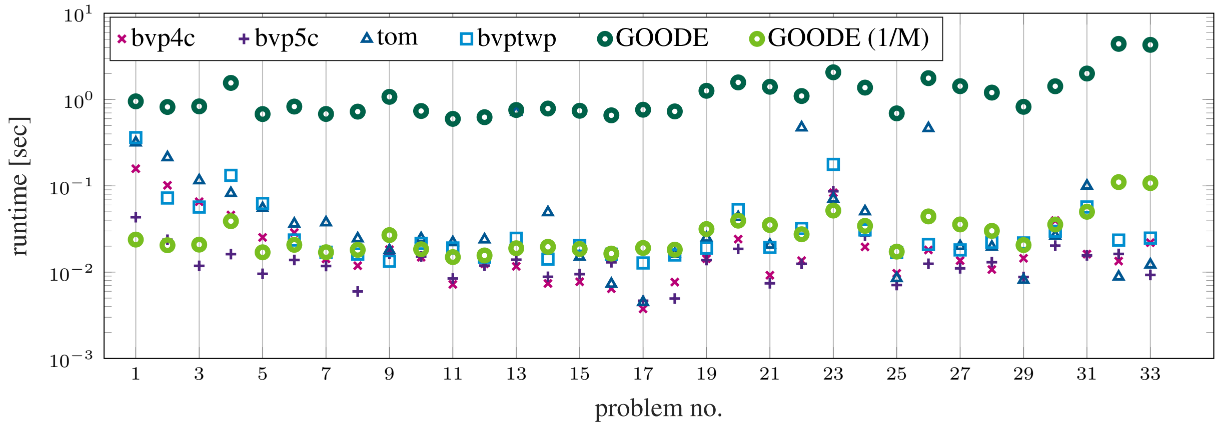


*Figure 1.* Computation times in seconds of several BVP solver to obtain approximative solutions for the test set.

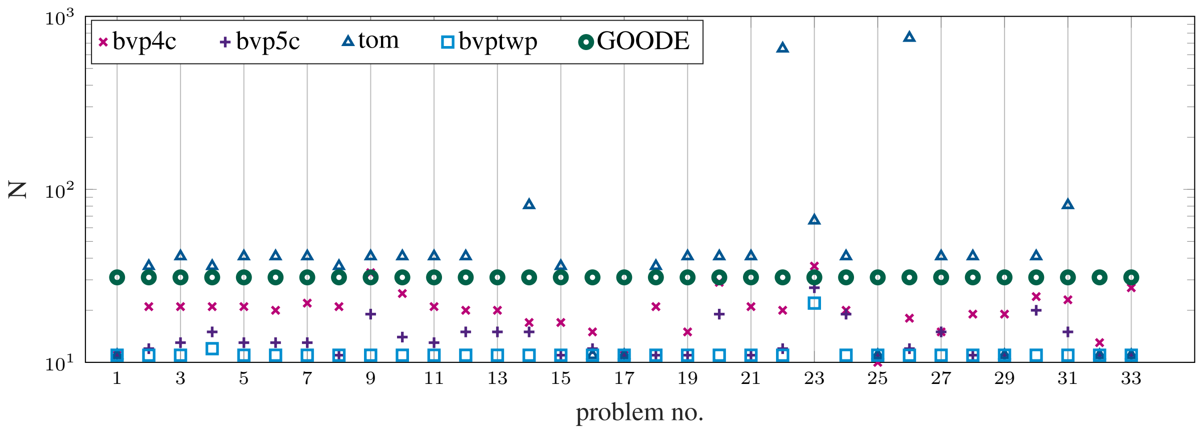


*Figure 2.* Number of mesh points $N$ used by several BVP solver for the test set.

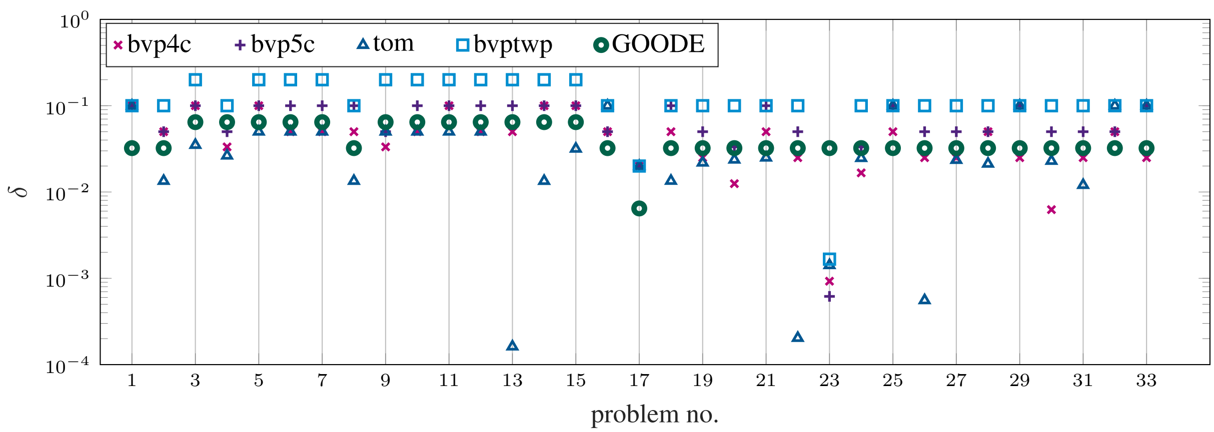


*Figure 3.* Minimum mesh width $\delta$ of the mesh $\Delta$ used by several BVP solver for the test set.

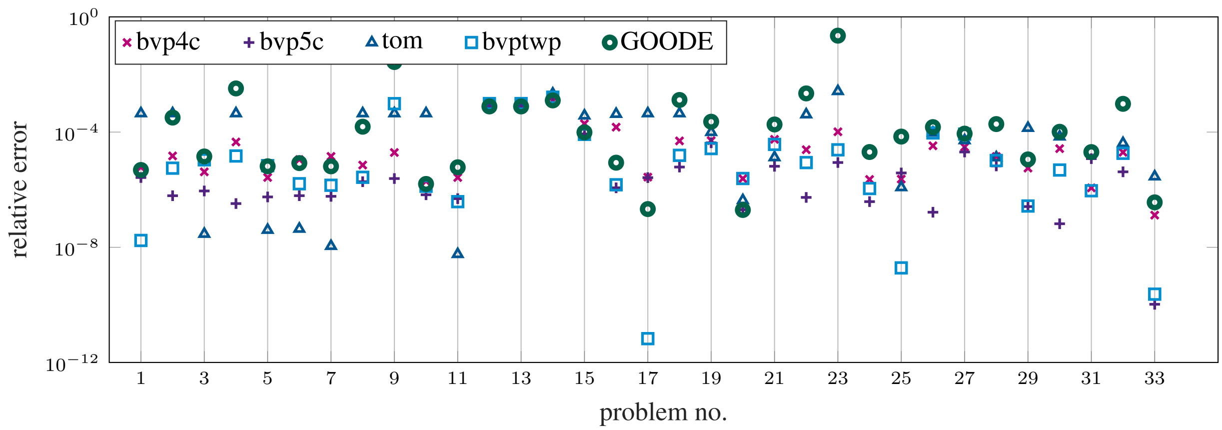


*Figure 4.* Relative errors of the approximations obtained by several BVP solver for the test set. GOODE with Matérn 5/2 kernel.

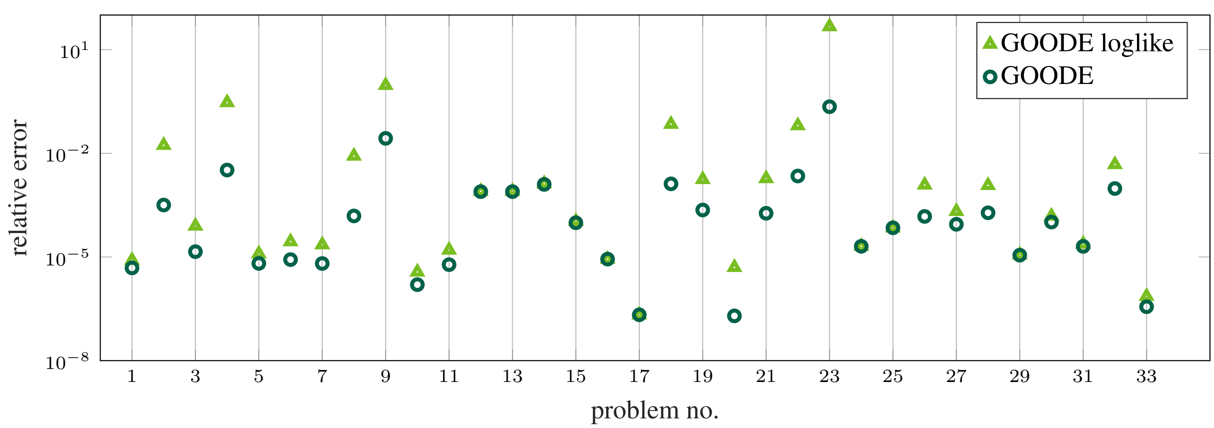


*Figure 5.* GOODE with Matérn 5/2 kernel. Comparison of global optimum (with respect to the reference) vs. log likelihood optimum, each computed on a fine grid for the Matérn 5/2 kernel hyper-parameter.

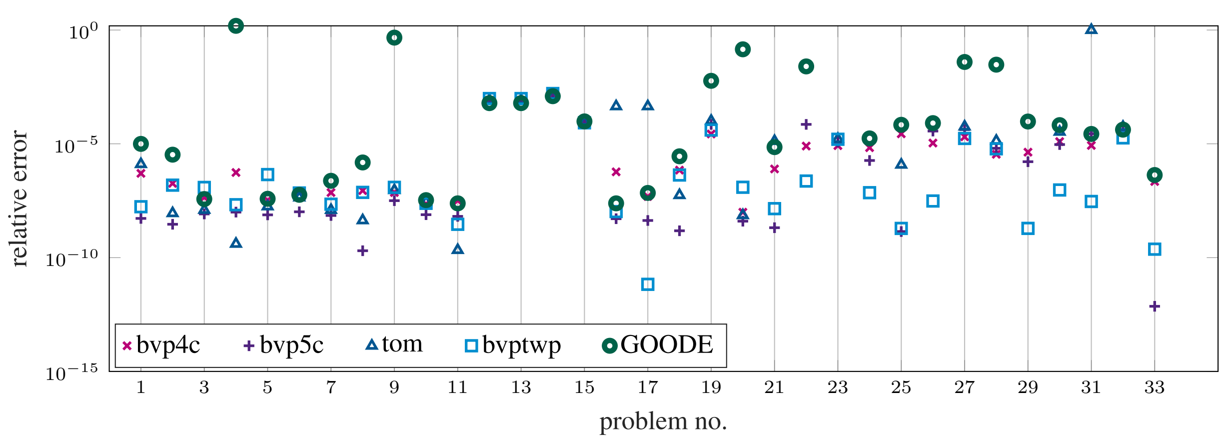


*Figure 6.* Relative errors of the approximations obtained by several BVP solver for the test set. GOODE with the `bvp5c` mesh.

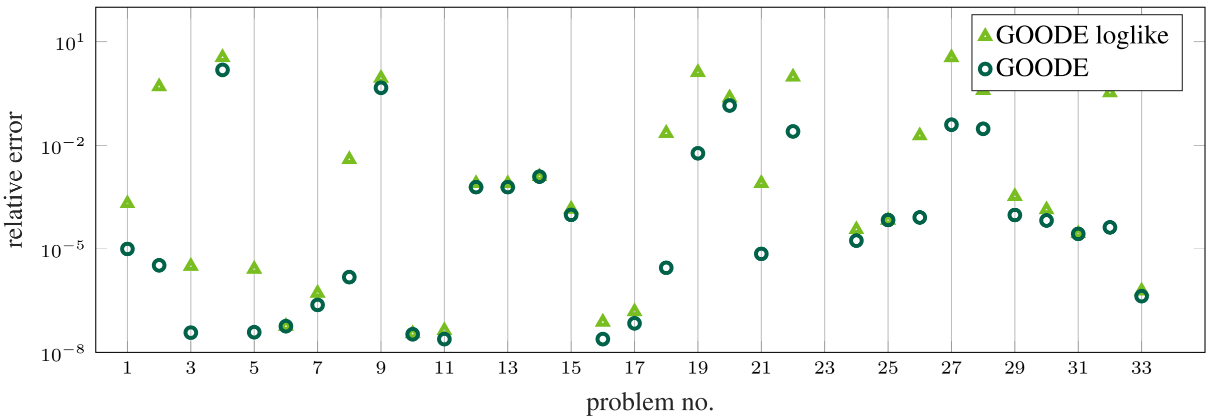


*Figure 7.* GOODE with the `bvp5c` mesh. Comparison of global optimum (with respect to the reference) vs. log likelihood optimum, each computed on a fine grid for $\lambda$.

## 4. Remark on Adaptive Mesh Refinement for GOODE

Adaptive mesh refinement for GOODE based on the local error estimate $\sigma(t)$ would be a useful extension. However, $\sigma(t)$ provides information about the local error of the approximation, but no information where grid points need to be placed to achieve a certain accuracy. Choosing an appropriate mesh is typically an optimization problem in its own right. For other methods various heuristics have been proposed to solve this efficiently. Introducing an ad-hoc heuristic without proper justification and testing would not have done the topic justice. In general, mesh selection in BVP algorithms is a more intricate problem than in IVPs (cf. Ascher et al. (1994, Chap. 9, 28 pages) vs. Hairer et al. (1987, Sect. II.4, 8 pages)) as two questions (number of additional points and locations) need to be solved simultaneously. Note that, to this date, there is only one published probabilistic ODE solver with automatic step size adaptation for IVPs, showcasing the difficulty even in the simpler case of IVPs (Schober et al., 2019).

## References

Ascher, U. M., Mattheij, R. M., and Russell, R. D. *Numerical solution of boundary value problems for ordinary differential equations*. Siam, 1994.

Hairer, E., Nørsett, S., and Wanner, G. *Solving Ordinary Differential Equations I – Nonstiff Problems*. Springer, 1987.

John, D. N., Heuveline, V., and Schober, M. GOODE: A Gaussian Off-the-shelf Ordinary Differential Equation Solver. In *Proceedings of the 36th International Conference on Machine Learning*, Proceedings of Machine Learning Research. PMLR, 2019.

Rasmussen, C. and Williams, C. *Gaussian Processes for Machine Learning*. MIT, 2006.

Schober, M., Särkkä, S., and Hennig, P. A probabilistic model for the numerical solution of initial value problems. *Statistics and Computing*, 29(1):99–122, Jan 2019. ISSN 1573-1375. doi: 10.1007/s11222-017-9798-7. URL https://doi.org/10.1007/s11222-017-9798-7.