A Compositional Framework for Convex Model Predictive Control*

Tyler Hanks[1] Baike She[1] Evan Patterson[2] Matthew Hale[1]
Matthew Klawonn[3] James Fairbanks[1]

[1]University of Florida
[2]Topos Institute
[3]Air Force Research Lab

Model-predictive control (MPC) originated in process control in chemical engineering [7], and it has found success in many applications, including autonomous driving [4], battery charging [3], path planning [5], and energy systems [6]. MPC consists of specifying and repeatedly solving constrained optimization problems. These problems are designed to model the response of a controlled system to inputs while satisfying operating constraints and minimizing a cost function over a finite prediction horizon. The generality of MPC gives it several benefits, including the ability to incorporate constraints on a system's state and control inputs, and increased robustness to sensor noise and/or system perturbations.

Conventionally, control engineers implement MPC for specific applications in ad-hoc ways using libraries such as CasADi [1] or CVX [2], which provide tools for nonlinear optimization and algorithmic differentiation, but are not specifically intended for MPC problems. As a result, control engineers using such tools must take their control-theoretic problem statement and transform it by hand into an optimization problem that can be solved by a generic solver.

There is reason to think that this approach can be improved. Classic MPC optimization problems exhibit a specific structure consistent across applications. For a prediction horizon of one time-step, the goal is to select an action that will yield the best subsequent state according to some objective function, state dynamics, and constraints. When extended to a prediction horizon with N steps, the goal is to visit a sequence of N states whose collective utility is given by applying the *same* objective function to each state, and summing the results. Similarly, the dynamics and constraint functions of the problem remain the same at each time step, and the dynamics link one step to another, in that the state reached after one time-step is the initial state for the next step. Intuitively then, MPC optimization problems can be seen as a composite problem resulting from N copies of some generating optimization problem that are coupled by way of the state dynamics. We refer to this as the *compositional structure* of the MPC optimization problem. We emphasize that *solutions* of optimization problems do not compose in general, i.e. there is no general way to build up an optimal solution to an N-step MPC problem by solving N 1-step MPC problems. Thus the compositional structure of MPC must be captured in a different way. To this end, we propose a category theoretic framework that captures the syntactic elements of this structure.

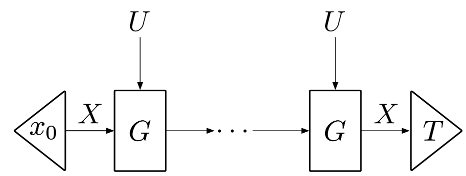

Figure 1: The string diagram representing the MPC problem structure.

Results. To develop this framework, we leverage the theory of convex bifunctions introduced in the seminal work of Rockefellar [8]. A con-

*This talk abstract is from a preprint of the same title to appear on the Arxiv.

vex bifunction subsumes a convex program into a single function, and two bifunctions can be composed in a way that uses the decision variables of one problem to perturb the constraints of the other. Our main contributions are the following.

1. We construct a symmetric monoidal category called **Conv** whose objects are Euclidean spaces and whose morphisms are convex bifunctions representing constrained convex optimization problems. This category provides a formal setting for composing optimization problems.

2. From **Conv**, we construct another category Para(**Conv**) of *parameterized* convex programs. This category formalizes the important semantic distinction between state variables and control variables in MPC optimization problems.

3. We show how convex MPC optimization problems are represented by composite morphisms in Para(**Conv**). Specifically, we show how a model of an affine open-loop control system together with a convex cost function and convex constraints produce a *generating* morphism in **Conv** and that the associated MPC optimization problem over a prediction horizon N is equivalent to the N-fold composition of this generating morphism with itself.

4. We demonstrate how to model initial and terminal state constraints and costs in our framework, which arise often in practice and are generally necessary for stability.

5. We implement this framework in the Julia programming language as a package called AlgebraicControl.jl.

Our Julia implementation provides the following benefits to control engineers.

- The automatic generation of MPC optimization problems in a form directly passable to commercial solvers such as IPopt [9] requires only a description of system dynamics, one-step cost function, and one-step constraints. This saves engineering time.

- The generated code is *correct by construction*, eliminating the possibility of implementation errors.

- Explicit modeling of the structure of MPC optimization problems allows rapid and correct implementation of high-level modeling changes, e.g. modification of system dynamics and constraints, modification of cost functions, switching between receding and moving horizon formulations, etc.

Discussion. The developed categorical framework formally links model predictive control problem formulations with convex optimization problem formulations in way that captures the intuitive, yet previously informal compositional structure of MPC problems. This framework applies the parameterized category construction to the category of convex bifunctions to properly account for the semantic differences between state variables and control inputs in MPC. We construct a convex program for every convex MPC problem such that the N-step horizon for MPC is given by the N-fold composition of a single parameterized bifunction. This insight has led to a software implementation for simulation of MPC controllers in the AlgebraicJulia ecosystem that integrates with existing state of the art solvers, via a code-generation approach. This implementation leverages our theoretical results to guarantee that the generated code is correct-by-construction. In addition to correctness, we argue that implementing software based on category theoretic abstractions eases the implementation burden for both the developer and the user.

Future work will address the categorical construction of specialized convex optimization routines that can exploit the compositional structure of MPC problems as identified in this paper. Additionally, although we focused specifically on MPC problems for linear dynamical systems, this framework can be extended to the non-linear case in future work.

References

[1] Joel A E Andersson, Joris Gillis, Greg Horn, James B Rawlings, and Moritz Diehl. CasADi – A software framework for nonlinear optimization and optimal control. *Math. Prog. Comp.*, 11(1):1–36, 2019.

[2] Michael Grant and Stephen Boyd. CVX: Matlab software for disciplined convex programming, Version 2.1. `http://cvxr.com/cvx`, March 2014.

[3] Branislav Hredzak, Vassilios G Agelidis, and Minsoo Jang. A model predictive control system for a hybrid battery-ultracapacitor power source. *IEEE Trans. on Power Elec.*, 29(3):1469–1479, 2013.

[4] Juraj Kabzan, Lukas Hewing, Alexander Liniger, and Melanie N Zeilinger. Learning-based model predictive control for autonomous racing. *IEEE Rob. and Autom. Lett.*, 4(4):3363–3370, 2019.

[5] Carlos E Luis, Marijan Vukosavljev, and Angela P Schoellig. Online trajectory generation with distributed model predictive control for multi-robot motion planning. *IEEE Robo. and Autom. Lett.*, 5(2):604–611, 2020.

[6] D Mariano-Hernández, L Hernández-Callejo, A Zorita-Lamadrid, O Duque-Pérez, and F Santos García. A review of strategies for building energy management system: Model predictive control, demand side management, optimization, and fault detect & diagnosis. *J. of Bui. Eng.*, 33:101692, 2021.

[7] S Joe Qin and Thomas A Badgwell. An overview of industrial model predictive control technology. In *AIChE symposium series*, volume 93, pages 232–256. New York, NY: American Institute of Chemical Engineers, 1971-c2002., 1997.

[8] R. Tyrrell Rockafellar. *Convex Analysis.* Princeton University Press, 1970.

[9] Andreas Wächter and Lorenz T Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Prog.*, 106:25–57, 2006.