Phoenix
A Systems Engineering Approach to Genetic Circuit Synthesis

Curtis Madsen[1], Evan Appleton[2], Prashant Vaidyanathan[1], Katherine Elkind[1], Rizki Mardian[1], Nicholas Roehner[1], Alan Pacheco[1], Calin Belta[1], and Douglas Densmore[1]

[1]Boston University Biological Design Center, [2]Harvard Medical School, Massachusetts, USA

Motivation

Systems engineering is an interdisciplinary field that focuses on building and maintaining complex engineering systems over their entire life cycles. Applying standards and core concepts of related sub-fields such as **performance engineering** (ensuring that a system meets its expected performance requirements) and **reliability engineering** (a system does not fail more than expected during its life cycle) is a promising way to advance synthetic biology's potential applications in research areas.

Approach

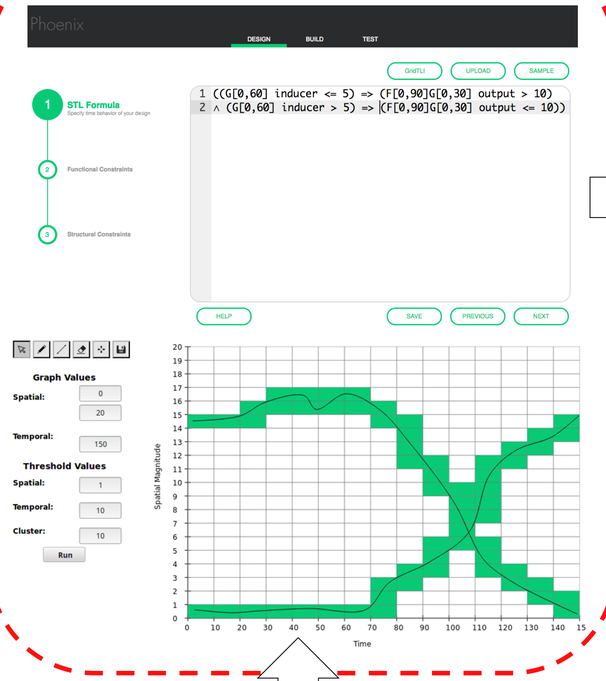
We have created a software platform (Phoenix) that encapsulates the procedures required during the **specify**-**design**-**build**-**test** workflow to enable iterative design of complex genetic systems. The framework is designed to ensure that the **functional**, **performance,** and **structural** specifications of the genetic system are well defined, reproducible, and reliable. The iterative process will use principles of **reinforcement learning** to improve the performance of the synthesis and assignment algorithms.

Acknowledgments

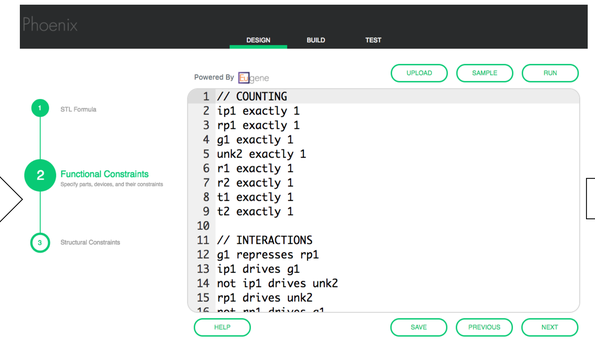
Research reported on this poster was supported by the National Science Foundation under grant CPS Frontier 1446607 and the Office of Naval Research MURI Award. The authors would also like to acknowledge Ernst Oberortner, Zachary Chapsko, Cristian-Ioan Vasile, Iman Haghighi, Masakazu Nagata, Yash Agarwal for their valuable feedback, suggestions, and contributions.

Phoenix Workflow

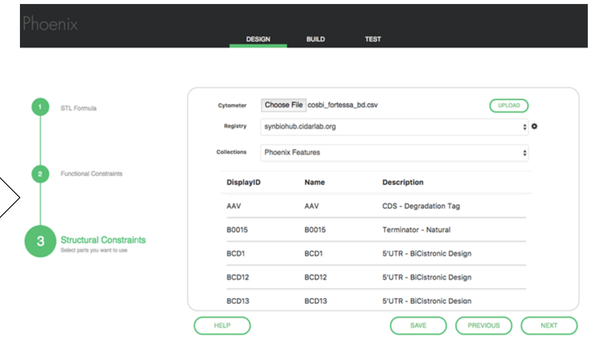
Specify STL Formula



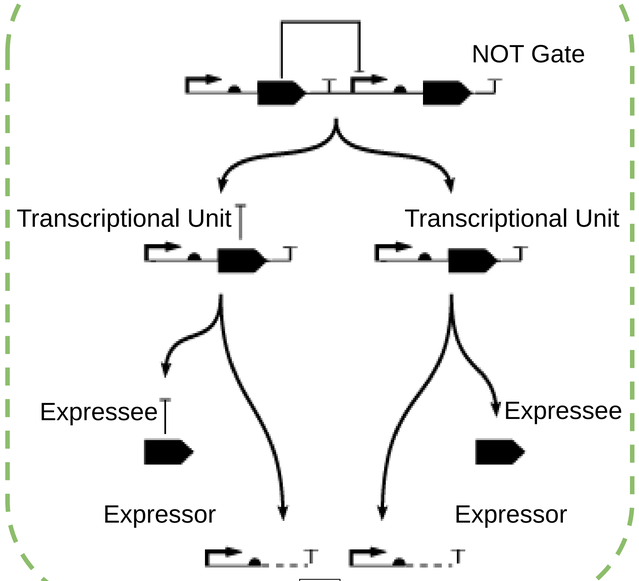
Structural Constraints



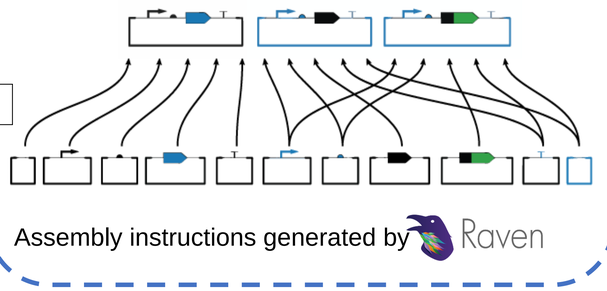
Library of Components



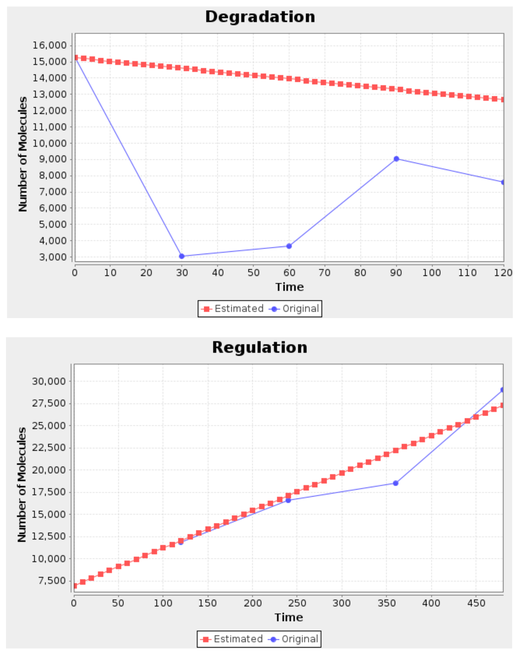
Decompose Structure



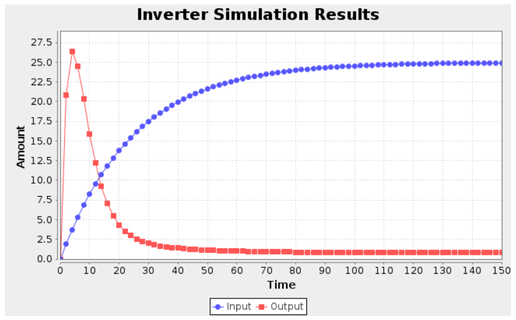
Generate Assembly Instructions



Estimate Parameters



Simulate



Test & Verify