GenomeCarver: harvesting genetic parts from genomes to support biological design automation

Emily Scher*
School of Informatics
University of Edinburgh
Edinburgh EH9 3JR, UK

Yisha Luo*
School of Biological Sciences
University of Edinburgh
Edinburgh EH9 3JR, UK

Aaron Berliner
Autodesk Research
San Francisco, California
94111, USA

Jacqueline Quinn
Autodesk Research
San Francisco, California
94111, USA

Carlos Olguin
Autodesk Research
San Francisco, California
94111, USA

Dr. Yizhi Cai†
School of Biological Sciences
University of Edinburgh
Edinburgh EH9 3JR, UK

ABSTRACT

Concept

The advance of genome sequencing and annotation has provided a "gold mine" of genetic parts, which all synthetic biologists wish to include in their toolbox of parts with which to build synthetic biological systems. The currently available computer assisted design systems (CADs) focus heavily, if not exclusively, on composing biological systems using genetic parts [3][7][9], however, how a user obtains parts in the first place remains an open question. To make matters worse, there are a few dozen part standards being proposed and used in the synthetic biology community (104 RFCs on part standard as of today). Even though one can extract a few parts from the genome manually, there is no software to ensure the standard compatibility of parts, and it is also very difficult to scale up the design of parts.

With these problems in mind, we present GenomeCarver, a computational tool for the harvesting and packaging of biological parts from model genomes. GenomeCarver interfaces with various genomes, identifies regions of interest according to user specification (*e.g.,* promoters, open reading frames and terminators) and extraction rules (*e.g.,* a promoter is defined as 500bp upstream of the ATG start codon or last gene boundary, which comes shorter), extracts corresponding DNA sequences from the genome feature files (GFFs), checks the sequence's compatibility with the selected standard (*e.g.,* whether the given sequence includes the forbidden restriction sites of certain parts standards), and finally outputs optimized primer sequences to amplify the parts from genomic DNA, adding necessary flanking sequences to standardize the parts.

Through its compatibility with multiple genomes and multiple parts standards, GenomeCarver bridges the fields of systems biology and synthetic biology, and greatly enriches synthetic biologists' design toolbox. It complements many parts-based design tools which currently exist by supporting the Synthetic Biology Open Language standard [6].

Implementation

GenomeCarver can be accessed as an application built on Autodesk's Project Cyborg (`http://autodeskresearch.com/projects/cyborg`). Project Cyborg is a cloud-based platform for computational tools in the life sciences and programmable matter space, supporting design and engineering across domains and scales. Cyborg enables elastic computing through a node framework that natively provides support for simulation, optimization, and visualization. Being built on Cyborg, GenomeCarver is comprised of nodes for each step of the workflow connected to form a cohesive user experience that guides the user through the tool.

GenomeCarver currently supports three model organisms: yeast *Saccharomyces cerevisiae*, bacterial *Escherichia coli*, and plant *Arabidopsis thaliana*. However, GenomeCarver is flexible enough to be extended to interface a variety of organisms, which we plan to do in the near future. Similarly, GenomeCarver currently supports a finite number of mainstream parts standards such as the BioBrick 1.0[1] and yeast Golden Gate standards[2], but new standards could easily be incorporated. In a future implementation, we even plan to allow users to import their own, custom standards. While it's interfacing with multiple genomes and standards has made GenomeCarver flexible, it's being built on top of Cyborg further's the tool's flexibility, as GenomeCarver will be able to be used in conjunction with the other tools currently being developed on the same platform.

Figure 1 shows the application's workflow. First, a user chooses a genome, a category and the loci of the part. For instance, a user may choose the promoter of *Gal* loci from *Saccharomyces cerevisiae*. Optionally, the user can then define the preferred promoter and terminator lengths, or specify that they would like gene boundaries to be ignored. The default maximum promoter length is 500 base pairs, and the default maximum terminator length is 200 base pairs. If the user does not specify that gene boundaries should be ignored, then GenomeCarver will identify a gene's promoter as the upstream (5' to 3') sequence of a maximum length of 500 (or the specified maximum length) which does not overlap another gene. It will identify the terminator as the following sequence of a maximum length of 200 bases which does not overlap another gene. GenomeCarver then returns the specified sequence(s). The user can then assign the sequence(s) to a standard using another drop down menu. Once selected, the sequence will then be checked

*These authors contributed equally to this work.
†Corresponding author. E-mail:yizhi.cai@ed.ac.uk

IWBDA 2014, June 11-12, 2014, Boston, Massachusetts, USA.

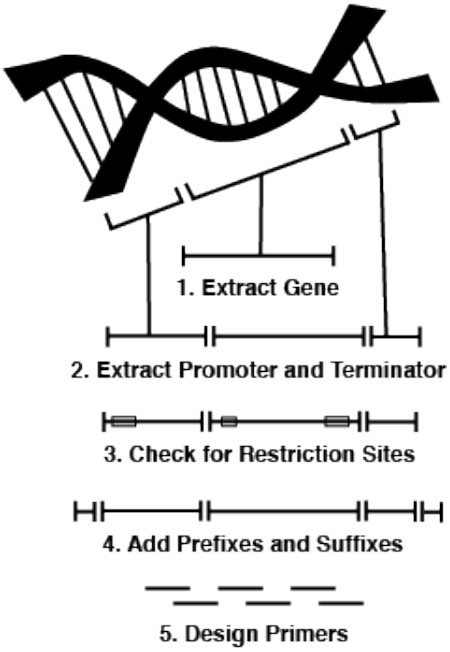

Figure 1: The workflow of GenomeCarver

for restriction sites, returning a warning if an incompatible restriction site is found. The part will then be packaged by adding the appropriate prefix and suffix. GenomeCarver then allows the packaged parts to be exported in CSV and SBOL formats [6].

Experimental verification
GenomeCarver has been used extensively in several labs in the USA, UK and China to systematically design thousands of yeast parts of each category conforming to the yeast Golden Gate standard. We used the designed primers to amplify parts from genomic DNA in a high throughput fashion, cloning the parts onto Topo vector backbone, and sequence verifying them all (data not shown in the abstract). We also demonstrated high efficient assembly of various genetic switches using these parts and standard Golden Gate reaction, and transformed these assembled switches into yeast for functional assays. Most recently, GenomeCarver has been used to design all the 6000 yeast promoters and 6000 yeast terminators, which demonstrates that we can scale up the design automation easily.

Future plan
In the next version of GenomeCarver, we are planing to include additional genomes, such as mammalian ones, as well as to support user-customized standards. Batch design functionality will also be developed to support large projects, such as BioFab (http://biofab.synberc.org/) type projects for various genomes. We will also develop better primer design strategies [8] to maximize the parts amplification success rate. We are also planning to support codon optimization for gene parts, so that a user can carve out a gene from one species and codon optimize it for another species, and GenomeCarve will output oligonucleotides for *de novo* DNA synthesis. Finally, a better integration with existing parts-based design tools will be needed for a better user design experience.

Conclusion
GenomeCarver has been built to fill a gap left by existing Synthetic Biology computational tools. It allows users to extract parts directly from genomes, and to package them into standardized formats for parts synthesis. We have used this tool to design over 12,000 parts, and constructed and verified several hundred of them. This tool, along with the parts repository we created using it, will be a useful and important addition to the synthetic biology community.

Acknowledgement
ES is supported by an Autodesk research fellowship. The project is funded by an Edinburgh Chancellor's Fellowship to YC. We thank Drs. Jef D. Boeke (New York University, USA) and Junbiao Dai (Tsinghua University, China) for helpful discussions to initiate the project.

1. REFERENCES

[1] Technical report, August 2005.
[2] Technical report, Johns Hopkins University, October 2012.
[3] Yizhi Cai, Brian Hartnett, Claes Gustafsson, and Jean Peccoud. A syntactic model to design and verify synthetic genetic constructs derived from standard biological parts. *Bioinformatics*, 2007.
[4] Carola Engler, Ramona Gruetzner, Romy Kandzia, and Sylvestre Marillonnet. Golden gate shuffling: A one-pot dna shuffling method based on type iis restriction enzymes. *PLoS ONE*, 4(5):e5553, 05 2009.
[5] Carola Engler, Romy Kandzia, and Sylvestre Marillonnet. A one pot, one step, precision cloning method with high throughput capability. *PLoS ONE*, 3(11):e3647, 11 2008.
[6] Michal Galdzicki, Cesar Rodriguez, Deepak Chandran, Herbert M. Sauro, and John H. Gennari. Standard biological parts knowledgebase. *PLoS ONE*, 6(2):e17005, 02 2011.
[7] Nathan J. Hillson, Rafael D. Rosengarten, and Jay D. Keasling. j5 dna assembly design automation software. *ACS Synthetic Biology*, 1(1):14–21, 2012.
[8] Andreas Untergasser, Ioana Cutcutache, Triinu Koressaar, Jian Ye, Brant C. Faircloth, Maido Remm, and Steven G. Rozen. Primer3âĂŤnew capabilities and interfaces. *Nucleic Acids Research*, 40(15):e115, 2012.
[9] Bing Xia, Swapnil Bhatia, Ben Bubenheim, Maisam Dadgar, Douglas Densmore, and J. Christopher Anderson. Chapter five - developer's and user's guide to clotho v2.0: A software platform for the creation of synthetic biological systems. In Christopher Voigt, editor, *Synthetic Biology, Part B Computer Aided Design and DNA Assembly*, volume 498 of *Methods in Enzymology*, pages 97 – 135. Academic Press, 2011.