Double Dutch: A Tool for Designing Libraries of Variant Metabolic Pathways

Nicholas Roehner
Boston University, US
nroehner@bu.edu

Douglas Densmore
Boston University, US
dougd@bu.edu

1. INTRODUCTION

The development of technologies for designing novel DNA components [2, 1] has enabled the design of large combinatorial libraries of variant metabolic pathways and genetic circuits. Since it can be difficult to physically construct and screen these libraries in their entirety, new tools are needed to design these libraries for efficient testing. To meet this need, we have developed Double Dutch, a web application that tailors libraries of variant pathways for use in a design of experiments (DOE) framework.

In the context of synthetic biology, DOE techniques can be used to restrict testing to only those variants that are statistically relevant for determining the relationship between variant parameters and measured performance. Despite this potential, there are few published instances of applying DOE methods to synthetic biology [4]. Currently, there exist general purpose DOE software tools, such as JMP [3], that service the biological sciences with varying degrees of specificity, but none have been explicitly developed for synthetic biology. To bridge the gap between biological and experimental design, Double Dutch automates the process of mapping from the coding sequences (CDS) and other characterized DNA components that make up variant pathways to the factors and levels that define the conditions of a full factorial experiment. The end result is a library of variant pathways that can be used in a DOE framework. Figure 1 presents an overview of this mapping process.

2. GRAMMAR

In order to determine which DNA components are eligible for mapping to the factors and levels of an experimental design, Double Dutch implements a formal grammar. The rules of this grammar specify that experimental factors must be implemented as partial genes that include at least one CDS, while the levels that each factor takes on must be implemented as parameterized DNA components that regulate gene expression, such as promoters, ribosome binding sites (RBS), and terminators. While the examples in this abstract focus on mapping RBS-CDS pairs to factors and mapping promoter-terminator pairs with REU measures of their transcription strengths to levels, Double Dutch is capable of supporting other use cases through its grammar. These include mapping promoter-CDS-terminator combinations to factors and mapping RBSs with REU measures of their translation strengths to levels, or mapping promoter-RBS-CDS combinations to factors and mapping terminators with measures of their relative efficiencies to levels.

3. LEVEL ASSIGNMENT

Once uploaded DNA components and parameters are classified as candidate factors or levels via a grammar, a Double Dutch user only needs to select the partial gene factors in their pathway of interest and choose a desired number of levels per factor. Double Dutch then uses heuristic algorithms, most notably k-means clustering and simulated annealing, to automate the process of assigning candidate DNA components to the levels of the experimental design. Prior to assignment, all candidate components are partitioned into k clusters based on their parameter values, where k is the chosen number of levels per factor. The mean parameter values of these clusters set the target values for each level, while the clusters themselves filter the candidates available for assignment to each level. Double Dutch also allows users to manually set these target values if desired.

Level assignments are costed as the weighted sum of three concerns: level matching, pathway homology, and component reuse. Double Dutch attempts to manage these conflicting concerns according to user-defined weights and find the level assignment with the smallest cost by randomly changing which DNA components are assigned to each experimental level. Each change is accepted or rejected in accordance with a simulated annealing heuristic. Under this heuristic, changes that increase the cost by a large amount are more likely to be accepted early on, which can help prevent entrapment in a local minimum.

In the case of level matching, Double Dutch attempts to minimize the quantitative differences between the parameters of the assigned DNA components and the target values of the experimental levels. In the case of pathway homology, Double Dutch attempts to minimize the number of homologous DNA components within each variant pathway of the resultant library, so as to reduce the risk of homologous recombination during pathway construction. Finally, in the case of component reuse, Double attempts to maximize the reuse of DNA components across variant pathways and thereby minimize the costs associated with modular cloning.

4. PRELIMINARY RESULTS

As a demonstration of Double Dutch's level assignment capability, Figure 2 shows the results of designing pathway libraries for experiments containing five to nine factors and two to five levels. In particular, the top of Figure 2 displays the costs of the best level assignments found by Double Dutch after 500 trials, while the bottom compares these assignments with the best found by a purely random approach. In this example, all three assignment concerns are

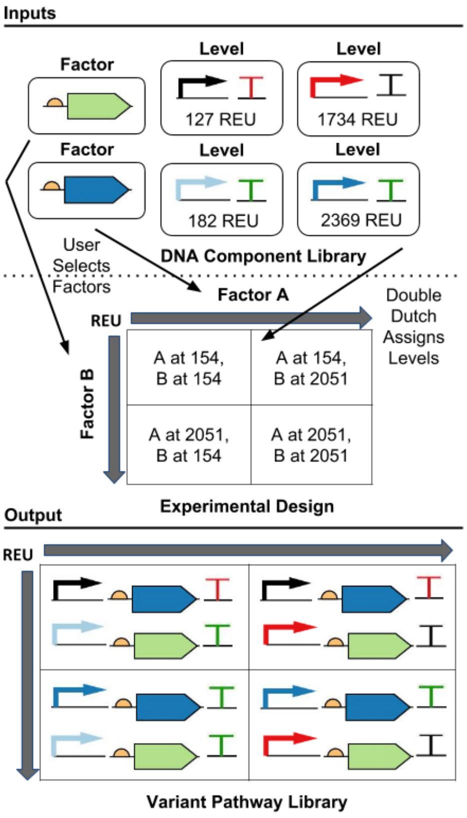

Figure 1: Overview of Double Dutch library design. DNA components are assigned to the factors and levels of a full factorial experimental design to produce a library of variant metabolic pathways.

weighted equally. In addition, all DNA components belong to a library containing 1,069 promoter-terminator pairs that have been characterized for transcription strength in yeast.

As shown in Figure 2, the cost of the level assignment found by Double Dutch generally increases as the size of the experimental design increases. In addition, the percentage by which Double Dutch outperforms random assignment generally decreases, though not to less than 25 percent for the largest designs. One cause of these effects is that, as the size of the experimental design approaches the limit of what the DNA component library can implement without introducing pathway homology, the level matching and component reuse costs are outweighed by a large pathway homology cost that dominates the total. Finally, the time taken by Double Dutch to perform 500 trials of level assignment increases with design size, but it scales tolerably and is less than six minutes for the largest designs in this example.

5. CONCLUSIONS

Double Dutch is among the first software tools capable of designing combinatorial libraries of variant metabolic pathways that are tailored for use in a DOE framework. While this framework relies on generic statistics software to prune variants for testing and fit the resulting data to an empirical model, we are currently implementing the same techniques in Double Dutch and seeking to customize them for use in a synthetic biological context. Ultimately, Double Dutch can be used to design libraries that provide better coverage of pathway design spaces, minimize the risk of homologous recombination, and reduce the monetary cost of modular cloning. Double Dutch is currently closed source, but the application can be accessed at `www.doubledutchcad.org`.

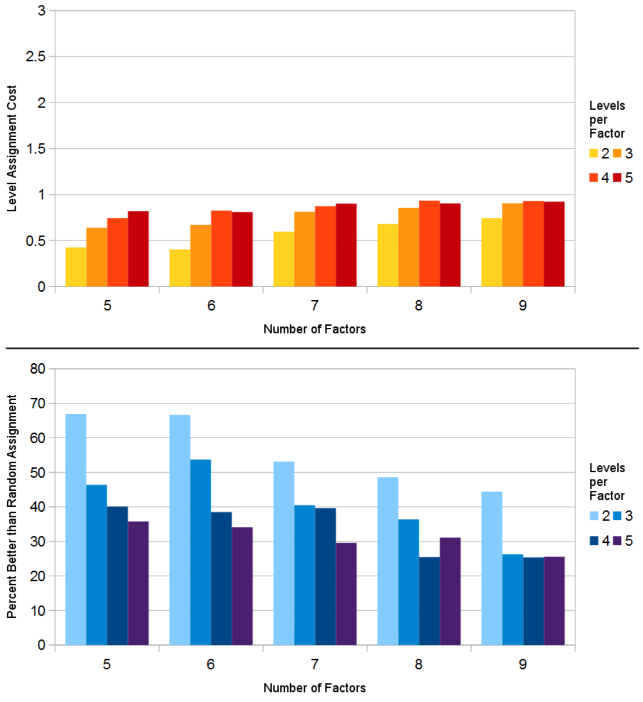

Figure 2: Costs of best level assignments found by Double Dutch (top) and the percentages by which Double Dutch outperforms random level assignment (bottom). The worst possible cost is three.

6. ACKNOWLEDGMENTS

We thank Benjamin Gordon, Eric Young, and the other members of the MIT-Broad Foundry for their aid in conceiving this project and providing characterization data for testing purposes.

7. REFERENCES

[1] Y.-J. Chen, P. Liu, A. A. Nielsen, J. A. Brophy, K. Clancy, T. Peterson, and C. A. Voigt. Characterization of 582 natural and synthetic terminators and quantification of their design constraints. *Nature Methods*, 10:659–664, 2013.

[2] H. M. Salis, E. A. Mirsky, and C. A. Voigt. Automated design of synthetic ribosome binding sites to control protein expression. *Nature Biotechnology*, 27:946–950, 2009.

[3] J. Sall, A. Lehman, M. L. Stephens, and L. Creighton. *JMP start statistics: a guide to statistics and data analysis using JMP*. SAS Institute, 2012.

[4] M. Welch, S. Govindarajan, J. E. Ness, A. Villalobos, A. Gurney, J. Minshull, and C. Gustafsson. Design parameters to control synthetic gene expression in Escherichia coli. *PLoS ONE*, 4, 2009.