Engin, Int J Biotechnol Bioeng 2014, 1: 102
http://dx.doi.org/10.15344/2014/ijbb/102

International Journal of
Biotechnology & Bioengineering

Commentary Open Access

Life Re-engineered with "Gene Compilers" – The Impact of Synthetic Biology on Biotechnology

Evren Doruk Engin*

Assistant Professor, Biotechnology Institute, Ankara University, Turkey

Publication History:
Received: February 06, 2014
Accepted: February 08, 2014
Published: February 10, 2014

The advent of recombinant DNA has started in early seventies, with the discovery and use of DNA manipulating enzymes, namely restriction endonucleases, terminal transferases and ligases. Cohen, Berg and Boyer performed the first "genetic engineering" experiments, by transferring gene segments to bacterial cells in 1973 [1,2]. Successful expression of fully functional human somatostatin in bacterial cells was followed by production of synthetic human insulin, which eventually gave rise to a new class of drugs and a huge industry. From engineering perspective, materials knowledgebase is crucial for a successful design. Hence, the pioneering works of these scientists and their successors involved more "black magic" than rational design, in the lack of DNA sequence data [3].

The invention of DNA sequencing with chain terminating inhibitors by Frederich Sanger, 24 years after the discovery of double helix structure of DNA, was one of the most notable advancements that accelerated the expansion of our knowledgebase regarding the "programming" of life [4]. The analysis of sequence data from numerous genome projects reshaped our understanding of the cell in a way that, the "operating" cell itself was not a hardcoded device. Indeed, it is now clear that, genomic organization of the cell blends two highly interleaved contexts: the protein coding blocks which store the encoded peptide sequences, and the regulatory blocks which decides the temporal and spatial abundance of all the protein machinery needed to carry out / execute the biochemical reactions to decrease the entropy and generate an ordered structure, so called "life"[5]. Interestingly, the genomic organization of the cell denotes considerable resemblance to "numerical machine code" which is directly executed by a computer's central processing unit. In this analogy model, protein coding sequences represent the data blocks that define the shape and form of the functonal objects to be created in the real world, and the regulatory sequences serve as the logic behind the scene, which performs background housekeeping tasks, takes inputs from the environment, make decisions and execute the creation of appropriate functional objects, in response [6]. We may further extend this analogy to apprehend the challenges we encounter in "deciphering genomes". Essentially, functional genomic studies can be considered as reverse engineering of a low level machine coded "closed source software", in which researchers use data mining algorithms to capture and annotate the coding and regulatory sequences, in the context of metabolic activities [7]. Likewise, the design of an expression vector construct is done much in the same way, by using low level machine coding like paradigm – that is, the designer should take care of the sequences and relative alignments of individual sequence elements as well as the feasibility of the cloning experiments to build that construct. Nonetheless, there are numerous examples where "low level coding paradigm" has worked seamlesly for molecular cloning of single genes and production of many recombinant proteins [8]. What if one wanted to implement and construct highly complex designs?

In modern software engineering, coding directly in numerical machine code is virtually never done, because the complexity of the code may extend far beyond the human perception for even small sized projects. Instead, it is industy standart to use "autocode" or "compiler" systems. A compiler is a sort of computer program that translates a high level programming language into executable numerical machine code. High level programming languages usually provide strong abstraction to hide away the unintelligible numerical machine code details and may even partly enable the programmer to use neutral language to describe the computational algorithm [9].

There is surely an impact of biology on computer science. Nature inspired computational models and hardware designs have gained popularity over years (10). In turn, the use of computational methods in biology has contributed much to our understanding of life [11]. With the arrival of new generation of technologies, DNA sequencing has become highly automated and several orders of magnitude cheaper process. This permitted affordable personalized genome projects start thriving, which greatly expands our "material knowledgebase" [12]. Improvements in the accuracy of solid phase oligonucleotide synthesis, along with novel automated strategies to assemble multiple overlapping single stranded fragments into hundreds to thousands base pair long DNA segments, obsoleted the strict requirement for biological gene source in cloning experiments, while enabling much straightforward protein engineering strategies [13]. Keeping Moore's Law of electronic hardware in mind, the synthesis of partial or complete custom chromosomes will shortly take their place in price lists of suppliers. When the time comes, designing the construct by using old school of low level machine coding paradigm will fall short to embrace the full power of synthetic biology. In case we were building our synthetic organism, we would consider reshaping the metabolism to direct the cells to our specified manufacturing or biotransformation process, which would in turn, involve transfer of an array of enzyme coding genes along with their regulatory circuitry. In our designer platform, the user (programmer / genetic engineer) would be able to describe the desired synthetic cell, in abstract terms of the metabolites, peptides, enzymes or biotransformation reactions, instead of plasmid, insert and oligonuclotide primer sequences or restriction enzyme choices. Once the design goals, key intermediate steps and constraints were set, the software component of the platform would be resposible for the low level tasks like retrieving sequence data from both general purpose databases (Genbank, EMBL,..) or specialized databases (Registry of Parts) [14], proper positioning of

***Corresponding Author:** Dr. Evren Doruk Engin, Assistant Professor, Biotechnology Institute, Ankara University, Turkey, E-mail: edengin@ankara.edu.tr

Citation: Engin ED (2014) Life Re-engineered with"Gene Compilers"–The Impact of Synthetic Biology on Biotechnology. Int J Biotechnol Bioeng 1: 102. doi: http://dx.doi.org/10.15344/2014/ijbb/102

the genetic elements, the design and simulation of possible amplification, digestion and ligation reactions. The platform would be able to make use of a hybrid strategy, combining the amplification of biological gene sources from template DNA material and artificial gene synthesis where needed. In the end of the design process, platform would yield a compilation of necessary molecular cloning steps and resultant construct(s) along with the list of required materials. Additionally, a microfludic station would have been incorporated to our synthetic biology platform to enable the partial or complete automated assembly of synthetic DNA molecule [15].

Synthetic biology takes its roots from the discovery of mathematical logic in gene regulation. Genetic engineering has granted us deeper access to the native regulation mechanisms, along with the ability to introduce a new set of heterologous peptides/enzymes or complete pathways to an organism. The impetus towards highly engineered synthetic organisms obviates the traditional genetic engineering approach in design and implementation. Accordingly, revisiting the software engineering – genetic engineering analogy probably offers the most versatile solution: Welcome to the age of gene compilers!

References

1. Jackson DA, Symons RH, Berg P (1972) Biochemical Method for Inserting New Genetic Information into DNA of Simian Virus 40: Circular SV40 DNA Molecules Containing Lambda Phage Genes and the Galactose Operon of Escherichia coli. Proc Natl Acad Sci U S A. 69: 2904-2909.

2. Cohen SN, Chang ACY, Boyer HW, Helling RB (1973) Construction of Biologically Functional Bacterial Plasmids In Vitro. Proc Natl Acad Sci U S A. Nov;70: 3240-3244.

3. De Lorenzo V (2010) Environmental biosafety in the age of synthetic biology: do we really need a radical new approach? Environmental fates of microorganisms bearing synthetic genomes could be predicted from previous data on traditionally engineered bacteria for in situ bioremediation. BioEssays. 32: 926-931.

4. Sanger F, Nicklen S, Coulson AR (1977) DNA sequencing with chain-terminating inhibitors. Proc Natl Acad Sci U S A. 74: 5463-5467.

5. Khalil AS, Collins JJ (2010) Synthetic Biology: Applications Come of Age. Nat Rev Genet. 11: 367-379.

6. Siuti P, Yazbek J, Lu TK (2013) Synthetic circuits integrating logic and memory in living cells. Nat Biotechnol. 31: 448-452.

7. Tatusova T, DiCuccio M, Badretdin A, Chetvernin V, Ciufo S, et al. (2013) Prokaryotic Genome Annotation Pipeline. (2nd edition), National Center for Biotechnology Information, USA.

8. Crea R, Kraszewski A, Hirose T, Itakura K (1978) Chemical synthesis of genes for human insulin. Proc Natl Acad Sci U S A. 75: 5765-5769.

9. http://en.wikipedia.org/w/index.php?title=Compiler&oldid=593569007

10. Jong-Hyun Lee, Chang Wook Ahn, and Jinung An, "An Approach to Self-Assembling Swarm Robots Using Multitree Genetic Programming," The Scientific World Journal, vol. 2013, Article ID 593848, 10 pages, 2013.

11. Klimke W, O'Donovan C, White O, Brister JR, Clark K, et al. (2011) Solving the Problem: Genome Annotation Standards before the Data Deluge. Stand Genomic Sci 5: 168-193.

12. Margulies M, Egholm M, Altman WE, Attiya S, Bader JS, et al. (2005) Genome Sequencing in Open Microfabricated High Density Picoliter Reactors. Nature. 437: 376-380.

13. Gibson DG (2011) Enzymatic assembly of overlapping DNA fragments. Methods Enzymol. 498: 349-361.

14. Constante M, Grunberg R, Isalan M (2011) A Biobrick Library for Cloning Custom Eukaryotic Plasmids. PLoS One. 6: e23685.

15. Gulati S, Rouilly V, Niu X, Chappell J, Kitney RI (2009) Opportunities for microfluidic technologies in synthetic biology. J R Soc Interface. 6: S493-S506.