ACS Synthetic Biology

Research Article

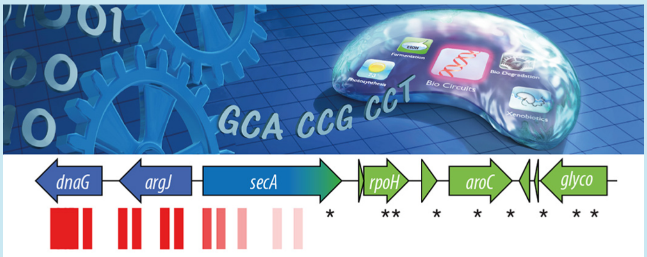
pubs.acs.org/synthbio

Genome Calligrapher: A Web Tool for Refactoring Bacterial Genome Sequences for *de Novo* DNA Synthesis

Matthias Christen,[†] Samuel Deutsch,[‡] and Beat Christen*,[†]

[†]Institute of Molecular Systems Biology, Eidgenössische Technische Hochschule (ETH) Zürich, CH-8093 Zürich, Switzerland
[‡]Joint Genome Institute, Walnut Creek, California 94598, United States

Supporting Information



ABSTRACT: Recent advances in synthetic biology have resulted in an increasing demand for the *de novo* synthesis of large-scale DNA constructs. Any process improvement that enables fast and cost-effective streamlining of digitized genetic information into fabricable DNA sequences holds great promise to study, mine, and engineer genomes. Here, we present Genome Calligrapher, a computer-aided design web tool intended for whole genome refactoring of bacterial chromosomes for *de novo* DNA synthesis. By applying a neutral recoding algorithm, Genome Calligrapher optimizes GC content and removes obstructive DNA features known to interfere with the synthesis of double-stranded DNA and the higher order assembly into large DNA constructs. Subsequent bioinformatics analysis revealed that synthesis constraints are prevalent among bacterial genomes. However, a low level of codon replacement is sufficient for refactoring bacterial genomes into easy-to-synthesize DNA sequences. To test the algorithm, 168 kb of synthetic DNA comprising approximately 20 percent of the synthetic essential genome of the cell-cycle bacterium *Caulobacter crescentus* was streamlined and then ordered from a commercial supplier of low-cost *de novo* DNA synthesis. The successful assembly into eight 20 kb segments indicates that Genome Calligrapher algorithm can be efficiently used to refactor difficult-to-synthesize DNA. Genome Calligrapher is broadly applicable to recode biosynthetic pathways, DNA sequences, and whole bacterial genomes, thus offering new opportunities to use synthetic biology tools to explore the functionality of microbial diversity. The Genome Calligrapher web tool can be accessed at https://christenlab.ethz.ch/GenomeCalligrapher .

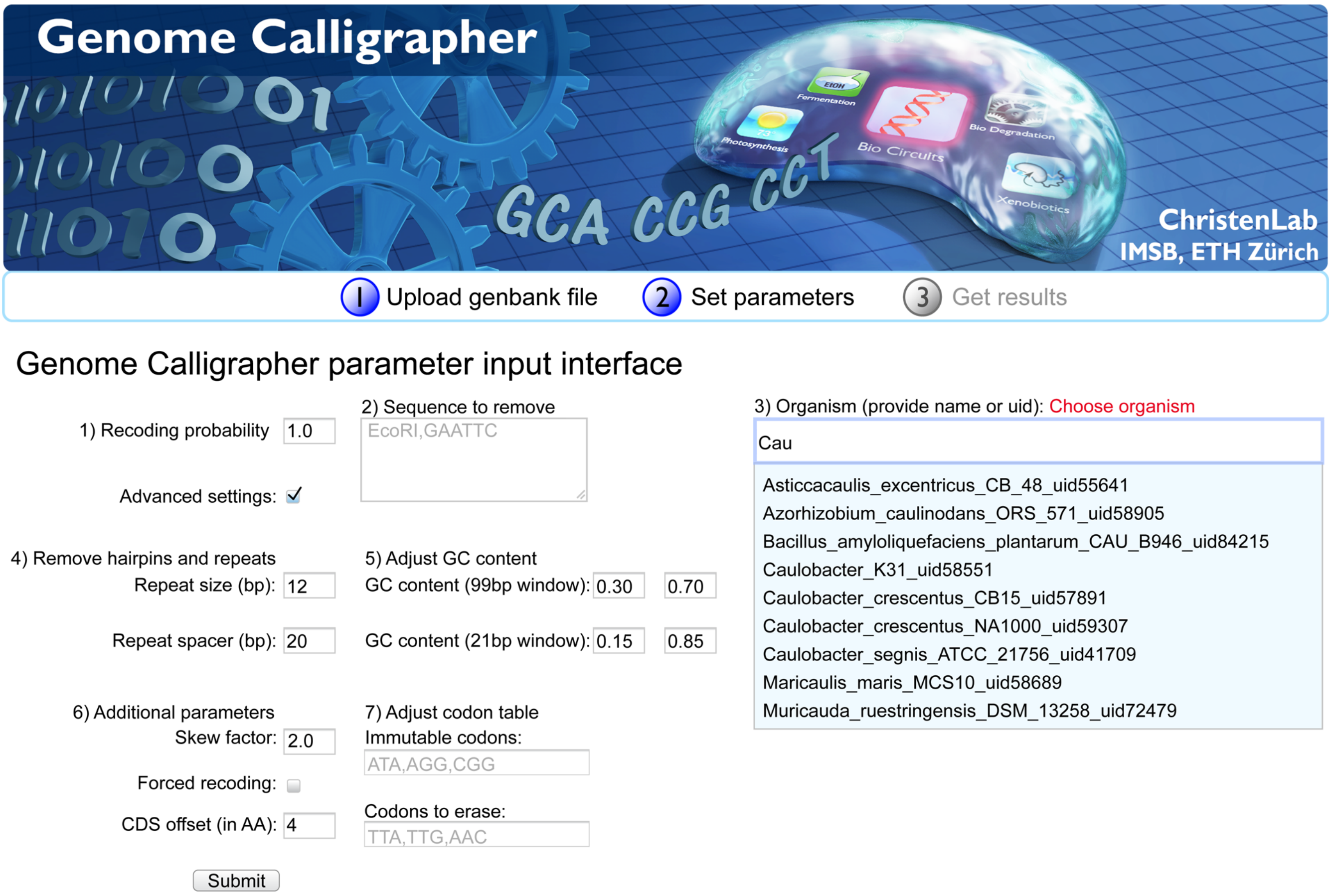
KEYWORDS: *DNA refactoring software, synthetic biology, de novo DNA synthesis, synthetic genome design*

With the advent of high-throughput DNA sequencing, massive sequence information has become available across all kingdoms of life. Synthetic biology holds great promise to study and mine this enormous wealth of genetic information. Of particular interest is the use of synthetic DNA to reprogram biosynthetic pathways and entire cells.[1] Using *de novo* DNA synthesis, chemically synthesized oligonucleotides can be assembled into double-stranded DNA (dsDNA) that serve as building blocks for the hierarchical assembly of multikilobase pair plasmids, chromosomes,[1,2] and whole genomes.[3,4] Despite impressive achievements in large-scale DNA synthesis, not every DNA sequence can be synthesized and ordered through commercial DNA synthesis providers in a cost- and time-effective manner. Secondary structure formation, polymerase slippage, and mispriming of oligonucleotides severely impede the oligonucleotide-based assembly of dsDNA.[5,6] In particular, sequences with high GC content impair proper annealing of single-stranded DNA molecules due to complex inter- and intramolecular folding within neighboring guanines.[7,8] This is a major issue for synthesizing DNA sequences derived from organisms with elevated GC content. In addition, homopolymeric DNA stretches and di- and trinucelotide repeats also impair annealing and proper assembly of single-stranded DNA into longer double-stranded fragments.[9,10] As a consequence, not every sequence deposited in genomic databases and DNA part repositories can actually be cost-efficiently manufactured by *de novo* DNA synthesis.

Alternative synthesis strategies such as the use of specially modified and purified oligonucleotides in combination with more sophisticated gene assembly methods do exist to facilitate *de novo* synthesis even in the presence of elevated GC content and other sequence complexities. However, these approaches need individual optimization for every fragment synthesized, are cost-intensive, and offer low throughput in gene synthesis with no guarantee of success. Especially in the case of biosynthetic pathway engineering and construction of entire genomes with hundreds to thousands of genes to be synthesized and assembled, it is necessary that every DNA part can be reliably manufactured in order to complete the desired outcome.

Special Issue: IWBDA 2014

Received: May 4, 2015
Published: June 24, 2015



Genome Calligrapher parameter input interface

1) Recoding probability: 1.0

2) Sequence to remove: EcoRI,GAATTC

3) Organism (provide name or uid): Choose organism

Cau

- Asticcacaulis_excentricus_CB_48_uid55641
- Azorhizobium_caulinodans_ORS_571_uid58905
- Bacillus_amyloliquefaciens_plantarum_CAU_B946_uid84215
- Caulobacter_K31_uid58551
- Caulobacter_crescentus_CB15_uid57891
- Caulobacter_crescentus_NA1000_uid59307
- Caulobacter_segnis_ATCC_21756_uid41709
- Maricaulis_maris_MCS10_uid58689
- Muricauda_ruestringensis_DSM_13258_uid72479

Advanced settings: ☑

4) Remove hairpins and repeats
Repeat size (bp): 12
Repeat spacer (bp): 20

5) Adjust GC content
GC content (99bp window): 0.30 0.70
GC content (21bp window): 0.15 0.85

6) Additional parameters
Skew factor: 2.0
Forced recoding: ☐
CDS offset (in AA): 4

7) Adjust codon table
Immutable codons: ATA,AGG,CGG
Codons to erase: TTA,TTG,AAC

Submit

Figure 1. Genome Calligrapher web interface. (A) The Genome Calligrapher parameter interface allows users to customize synthesis optimization criteria, to select precomputed codon tables, and to adjust particular recoding parameters. In addition to streamlining sequences for DNA synthesis, Genome Calligrapher supports the biological design of DNA and adaptation to the codon usage table. Users can choose to streamline sequences using preset parameters or define customized criteria for specific sequence optimization needs, including specification of disallowed sequence patterns (endonuclease sites and other biologically active sequences). Lower and upper GC content limits can be set for two different sliding windows (99 and 21 bp in size). Parameters related to repeat structures such as the repeat size and spacer length between repeat sequences can be adjusted. Precomputed codon usage tables can be conveniently selected from a total of 2776 sequenced bacterial genomes. Users can specify certain codons as immutable or define forbidden codons to erase from the codon table. Furthermore, a global recoding probability can be specified to introduce a low level of neutral nucleotide substitutions for seeding watermarks into sequences or, when set to high recoding probabilities, to perform gene taming and erase any additional genetic features beyond the encoded proteins.

Sequence-optimization algorithms can be used to eliminate sequence features that interfere with DNA synthesis. The polypeptide sequence information within each protein coding sequence is encoded by a series of 61 nucleotide triplets for 20 amino acids. This redundancy of the genetic code allows a particular codon to be replaced by synonymous ones that still code for the same amino acid. Such codon optimization has been used mainly for improving the expression of individual genes in heterologous host organisms,[11] with only a few algorithms focusing on the optimization of *de novo* DNA synthesis constraints.[12−15] For biosynthetic pathway engineering and whole genome synthesis projects with hundreds of genes to be synthesized, manual processing of individual protein coding sequences in bespoke manner is not feasible. In addition, sequence optimization of whole bacterial genomes is a far more complex task involving correctly handling different types of genetic elements such that biological functionality is maintained. Several design factors need to be considered simultaneously, and optimization of hundreds of DNA parts needs to be performed in a computationally efficient manner. Furthermore, sequence optimization should facilitate chemical DNA synthesis while maintaining the biological functionality of the features encoded. To address these needs, we have developed Genome Calligrapher, implemented as a web-based interface, that provides a collection of tunable sequence optimization features. Genome Calligrapher is, to our knowledge, the first algorithm specifically intended for genome-wide refactoring of bacterial genomes.

■ RESULTS AND DISCUSSION

The Genome Calligrapher DNA Synthesis Optimization Web Tool. The Genome Calligrapher web tool automates the DNA sequence optimization of large multipart DNA constructs including multikilobase pair plasmids and entire synthetic genomes. The Genome Calligrapher algorithm is written in the Python programming language and is accessible across computer platforms through a PHP-based web browser interface (see Figure 1). Input and output sequences of Genome Calligrapher are in community standard GenBank file format and permit seamless integration into various synthetic biology applications. The Genome Calligrapher web tool provides detailed online documentation, with explanations of

Table 1. List of the Preset Sequence Optimization Parameters Used by Genome Calligrapher

		impaired oligo assembly due to				
Genome Calligrapher parameter	oligo synthesis	T_m	secondary structure	mispriming	parameter type	standard parameter values
GC 99 upper limit	low yield[a]	high		yes	variable	0.70
GC 21 upper limit	low yield[a]	high	yes	yes	variable	0.85
GC 99 lower limit		low			variable	0.30
GC 21 lower limit		low			variable	0.15
direct repeat				yes	variable	size 8–20, spacer 0–20 bp
inverted repeat			yes	yes	variable	size 8–20, spacer 0–20 bp
homopolymeric				yes	fix	listed in Table S1
dinucelotide				yes	fix	listed in Table S1
trinucleotide repeats				yes	fix	listed in Table S1

[a] Guanine-rich oligonucleotides are less efficiently synthesized due to aggregation and solubility issues.

parameter settings and embedded functionalities and descriptions of log files and output files.

To begin the DNA synthesis optimization process, the user first specifies a GenBank sequence for server upload. All GenBank files consisting of a single record of up to 5 Mb in file size are accepted for upload. Splitting of the sequence is recommended for larger genome input files. Genome Calligrapher is intended for refactoring prokaryotic sequences. However, eukaryotic sequences can also be processed as long as no discontinuous CDS features or out-of-phase CDS structures are present. During the upload process, Genome Calligrapher evaluates the input sequence file to conform to GenBank file format specifications (full description of the conformity tests performed are listed in Supporting Information and Methods). Next, after uploading the input sequence file, a new GUI window appears where the user defines organism-specific codon usage tables and recoding parameters (see Figure 1). The Genome Calligrapher algorithm uses built-in rules for removal of homopolymeric sequences and di- and trinucleotide repeats (Table S1). These sequences can lead to misaligned oligonucleotide upon dsDNA assembly that result in short insertion or deletion events. Furthermore, the user can specify additional parameters to customize the sequence optimization process. An overview on optimization parameters and their consequences on oligonucleotide synthesis and assembly performance is shown in Table 1. The recoding probability specified in input field 1 defines the frequency of synonymous codon swapping. The user should consider setting global recoding probability to zero if the source and recipient organism are identical. In this case, the recoding algorithm performs the fewest number of changes while facilitating DNA synthesis optimization. Increasing the recoding probability allows neutral watermarks to be seeded into sequences. High recoding probabilities are used to exchange codon tables or to erase any additional genetic features beyond the encoded protein sequence (i.e., perform gene taming). In input field 2, the user can specify disallowed sequences such as type IIs endonuclease sites or other biologically active sequences to be removed by the recoding algorithm. The Genome Calligrapher web tool contains precomputed codon tables from 2776 bacterial genomes (Methods). To enable sequence refactoring for synthetic biology applications in yeast, Genome Calligrapher also includes the codon usage table of *Saccharomyces cerevisiae*. From this list, the user specifies a codon table for recoding by typing the organism's name or its unique NCBI identifier number (uid) into an autocomplete assisted input field (input field 3).

At this stage, the user can submit the recoding request or, alternatively, fine-tune additional recoding parameters. Among them are parameters for removing hairpins and repeats (input field 4) and upper and lower thresholds for GC content (input field 5). Hairpins and repeats interfere with annealing and extension of overlapping oligonucleotides into dsDNA fragments. GC-content limits are specified for two separate sliding windows of 99 and 21 bp in length. The GC-content threshold for the 99 bp window is used to restrict the melting temperatures of oligonucleotide, whereas the 21 bp window is used to detect short sequences that potentially form secondary structures (for further details on GC window parameters, see Supporting Information and Methods). Additional recoding parameters can be specified in input field 6. The skew factor adjusts codon frequencies to balance GC content. Optionally, the user can set the checkbox "forced recoding" to force the algorithm to replace every codon selected for recoding with a synonymous codon, which is particularly useful for seeding defined levels of watermarks into synthetic sequences. Furthermore, because the first few codons of a CDS contain important signals for translation initiation, the user can protect 5′ sequences from recoding (5′ CDS offset, input field). In input field 7, the user can customize the codon table by specifying immutable codons or erase certain codons from the genetic code. Immutable codons allow the user to preserve rare codons that represent important pause sites for ribosomes.[16] The input field "codons to erase" can be used to free up orthogonal sets of tRNA and edit the genetic code.[17,18]

After the advanced parameter settings have been specified, the job can be submitted to the web server. Before entering the main loop for sequence optimization, the Genome Calligrapher algorithm tests all CDSs from the GenBank file for reading-frame integrity, excludes overlapping CDS segments from recoding, and corrects the reading frame of CDS remnants that might have been split at DNA part boundaries. Next, the Genome Calligrapher algorithm iterates through each CDS and adjusts the GC content of the target sequence, replaces hairpins and direct repeats, and removes disallowed sequence patterns. (For a detailed description of the algorithm, see Supporting Information and Methods.) While sequences of tens of kilobase pairs in size are usually processed within seconds, larger GenBank files composed of complete bacterial genomes may take a few minutes to process (stated performance relates to a single core process). During this process, a progress bar will show the percentage and number of CDS for which recoding has been completed.

After completion of the algorithm, the user is guided to a graphical output interface where a GenBank output file of the optimized sequence can be downloaded (see Figure S1). In addition, log files from the recoding process, statistics, and parameter files are provided. A graphic output of the GC-content adjustment is shown together with a table summarizing codon frequencies before and after the recoding process (Figure S1). The downloaded sequence file can be fed into DNA partitioning and oligonucleotide design tools or directly submitted to a commercial DNA synthesis provider. In summary, we have implemented our Genome Calligrapher algorithm as a user-friendly synthetic biology web tool that enables fast and efficient multipart and genome-scale sequence optimization for *de novo* DNA synthesis at scales only accessible with computer-aided automation.

DNA Synthesis Constraints Across Bacterial Genomes. To globally examine the degree of synthesis constraints across all sequenced bacterial genomes and to determine the amount of recoding required to transform these sequences into fabricable DNA parts, we carried out bioinformatics analyses of all completed bacterial genomes deposited at NCBI GenBank database. Overall, we analyzed 2.6 Gb of sequences from 4720 microbial chromosomes and plasmids and identified the fraction of bacterial genes that could be synthesized as wild-type sequences (see Figure 2A and Tables 2 and S2). For this analysis, conservative synthesis constrain criteria designed to pass sequence screens from all major vendors were applied (see Supporting Information, Table S2). The outcome was quite surprising: according to our analysis, only 24.84% of all wild-type protein-coding sequences (CDS) are amenable to synthesis, whereas a large fraction of 75.16% of the sequences deposited at the GenBank database are excluded from standard *de novo* DNA synthesis because of sequence feature violations, as specified by commercial DNA synthesis vendors (Table S1). Even for *Escherichia coli* K12, with a well-balanced GC content of 52.0%, only 57.6% (2489/4319 CDS) of the annotated CDS passed our synthesis criteria (see Figure 2A and Table 2). An even more dramatic picture is observed for the 54.3% of all bacterial species with GC content below 40 or higher than 60 percent. For these genomes, typically less than one-third of the protein-coding genes are amenable to *de novo* DNA synthesis. Several important prokaryotic model organisms for synthetic biology possess difficult-to-synthesize genomes (see Table 2). The cell-cycle model organism *Caulobacter crescentus*[19] displays an extremely low predicted synthesis success rate of 5.05% (196/3885 CDS). Similarly, for the methanol-degrading organism *Methylobacterium extorquens* AM1,[20] only 4.22% (209/4947 CDS) can be commercially synthesized without recoding. On average across all bacteria species, each CDS contains more than eight GC-content violations. Sequence patterns impeding *de novo* DNA synthesis occur in 1 out of 10 CDS, and hairpins or inverted repeats, in 1 out of 100 CDS. Cumulatively, in a 4 Mb bacterial genome, there are, on average, 31 000 GC violations, 400 sequence exceptions, and 60 hairpins detected. In sum, we estimate that roughly 5.89 million bacterial genes (out of a total of 7.84 million genes deposited at NCBI) would be rejected for synthesis by commercial providers of standard double-stranded DNA synthesis due to a high risk of synthesis failure.

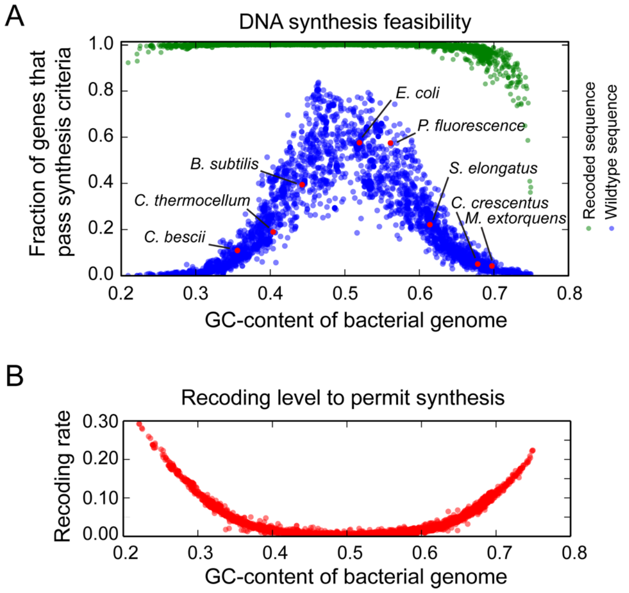


Figure 2. Occurrence and frequency of different types of *de novo* synthesis constraints across sequenced bacterial genomes and plasmid sequence (>100 kbp) deposited at NCBI. (A) The fraction of protein-coding genes amenable to low-cost *de novo* DNA synthesis is shown for the original sequences (blue) and after refactoring with the Genome Calligrapher algorithm (green). Commonly used synthetic biology model organisms are highlighted (red). (B) Level of recoding needed to remove synthesis constraints within protein-coding sequences is plotted as a function of the GC content of the original genome sequence (red). Detailed synthesis feasibility statistics assessed by the Genome Calligrapher algorithm for 4720 sequenced bacterial chromosomes and plasmids are listed in Supporting Information, Table S2.

Level of Recoding Required for Removal of DNA Synthesis Constraints. To assess the level of recoding needed to optimize distinct bacterial genomes for *de novo* DNA synthesis, we processed all bacterial GenBank files (downloaded from NCBI) and streamlined a total of 7.8 million CDS with the Genome Calligrapher algorithm. We used standard synthesis optimization parameters and maintained codon usage tables for each organism (see Methods). After sequence optimization, 98.72% (7.74 million out of 7.84 million) of all CDS passed synthesis criteria and sequence constraints were resolved with 99.8% efficiency (Table S2). We then assessed the degree of recoding needed as a function of the GC content. For genomes with a balanced GC content between 40 and 60 percent, less than 0.91% of recoding was required to overcome synthesis constraints (see Figure 2B and Tables 2 and S2). For the remaining genomes that display a more dramatic GC skew, on average a codon replacement rate of 6.87% was sufficient for sequence optimization. Given these moderate levels of recoding, it is likely that the Genome Calligrapher algorithm preserves the biological functionality of the recoded CDS.[21]

Design, Streamlining, and Partial Synthesis of a Synthetic Essential Genome. To test the feasibility and utility of our sequence optimization approach for *de novo* synthesis of whole bacterial genomes, we designed a synthetic minimal genome based on all essential and fitness-relevant sequences of the cell-cycle organism *C. crescents*. A total of 567 single and multipart sequences, identified by an ultradense transposon mutagenesis approach (TnSeq),[22] were compiled into a 766 828 bp long genome construct. An overview of the design process is shown in Figure 3. Cumulatively, the designed genome sequence contains 3920 annotated genetic features including 657 protein coding genes, 3 rRNAs, 48 tRNAs, 43 ncRNAs, 2757 promoters, and 334 stem loop terminators as well as 79 small essential features (Tables S2). From a

Table 2. Occurrence of *de Novo* DNA Synthesis Constraints Across Protein Coding Genes of Different Bacterial Genomes

			no. of synthesis constraints			
organism (genome size)	GC content (%)	DNA synthesis rate[a] (%)	GC content violations[b]	exception sequences[c]	hairpins and repeats[d]	recoding required[e] (%)
Bacillus subtilis (4.2 Mb)	44.3	39.4	7061	168	15	0.85
Caldicellulosiruptor bescii (4.2 Mb)	37.1	10.9	23 628	76	34	4.19
Caulobacter crescentus (4.0 Mb)	67.8	5.0	60 751	359	48	7.63
Clostridium thermocellum (3.8 Mb)	41.0	19.0	14 107	149	34	1.96
Escherichia coli (4.6 Mb)	52.0	57.6	3794	245	18	0.42
Pseudomonas fluorescence (6.4 Mb)	61.4	22.1	21 380	296	53	1.67
Synechococcus elongatus (2.7 Mb)	56.1	57.5	1566	519	13	0.39

[a] Percentage of protein coding sequences without sequence constraints impeding low-cost *de novo* DNA synthesis. [b] Sum of violations detected within protein coding sequences using 99 and 21 bp sliding windows. [c] Total number of homopolymeric, di- and trinucleotide repeat sequences detected. [d] Total number of hairpins and direct repeat sequences detected by the Genome Calligrapher algorithm. [e] Specifies the amount of synonymous codon substitution required for removal of DNA synthesis constraints.

commercial supplier of *de novo* DNA synthesis (Integrated DNA Technologies), we requested a feasibility analysis for the manufacturing of the entire genome constructs out of 700 bp dsDNA bricks. The sequences tested for synthesis feasibility were once the original minimal genome based on wild-type sequence parts and a synthetic sequence version of the genome with protein-coding stretches optimized through our Genome Calligrapher algorithm (see Figure 4 and Tables S4−S6). Out of the wild-type sequence version, 50.22% of the 700 bp dsDNA blocks (561/1117 gBlocks) fail synthesis criteria, most often because of GC content violations. In contrast, after sequence refactoring with the Genome Calligrapher algorithm, only 4.12% of all dsDNA building blocks (46/1117 gBlocks) are not accepted for synthesis, most likely because they contain noncoding sequences not targeted by the optimization algorithm. These results indicate that sequence optimization of protein-coding sequences alone is sufficient for polishing bacterial genome sequences for DNA synthesis.

Next, to test the synthesis feasibility of the optimized sequence version of the synthetic minimal genome, we decided to synthesize a set of eight 20 kb long segments (total synthesis effort of 168 kb) covering approximately 20 percent of the complete synthetic minimal genome construct. The genome segments were partitioned into 61 overlapping 3 kb DNA building blocks that were ordered from a commercial supplier of low-cost synthetic DNA (Gen9). Whereas 54/61 wild-type sequences contained sequence feature violations, all of the recoded sequences passed the feature screen. The overall synthesis success rate was 93% (57 building blocks out of 61 ordered were successfully produced) (Table 3). The remaining 4 building blocks that failed synthesis in the first round were reordered and successfully manufactured from a different supplier (Life-Technologies, GeneArt). Using higher-order assembly in yeast, these 61 synthesized 3 kb DNA building blocks were then successfully assembled into the desired eight 20 kb genome segments (Supporting Information and Methods). In sum, these results indicate that Genome Calligrapher can be efficiently used to render difficult-to-synthesize sequences, up to the size of entire bacterial genomes, into fabricable synthetic DNA.

Summary and Conclusions. Synthetic Biology holds great promise for solving global challenges. Of particular interest is the prospect to engineer pathways and entire cells to produce food, fuels, or chemical compounds in a more sustainable way.[23−26] Recently, large biosynthetic pathways[27−30] and even synthetic copies of whole genomes have been successfully assembled and transplanted into cells.[1,3] Despite these impressive achievements, most synthetic genomes maintain gene organization and sequences from wild-type templates. However, the real potential of *de novo* DNA synthesis resides in the engineering of DNA molecules that lack biological counterparts. As the price of synthesis drops further, *de novo* DNA synthesis will play an increasingly large role as an enabling technology for the fabrication of artificial genetic programs and their introduction into biological systems. For this vision to become a reality, software tools and algorithms that aid in the design of artificial genomes and biosynthetic pathways to enable their successful synthesis, assembly, and expression will be of crucial importance.

Currently, the majority of available software tools streamline FASTA sequences of individual CDS,[13,14,31,32] with most of them focusing on optimization of recombinant protein expression rather than DNA synthesis optimization. Here, we present Genome Calligrapher as a web tool to process large multipart assembly constructs up to whole bacterial genomes for optimized DNA synthesis. With the successful manufacturing of 168 kb of synthetic DNA, representing 20% of the essential *Collaborate crescentus* genome sequence, we provide experimental evidence that computational streamlining of DNA sequences is a fast and cost-effective approach toward the manufacturing of large-scale DNA constructs. Studies that probed the genetic limits of recoding within essential genes revealed remarkable degrees of codon replacement tolerated even within CDS encoding cellular core functions.[21] Given the minimal level of recoding introduced upon sequence streamlining by Genome Calligrapher, it is likely that the algorithm maintains the biological functionality of the genetic features and programs encoded. From a synthetic biology perspective, it will be exciting to further test these synthetic constructs for functionality in a cellular system and define the conditions where recoding impairs biological function.

Finally, the Genome Calligrapher algorithm is not intended to optimize protein expression levels or to assist in the biological design process of individual DNA parts to be assembled into functional biosynthetic pathways. However, the standardized GenBank file format used by Genome Calligrapher facilitates compatibility within independent synthetic biology software tools that do support such biological function

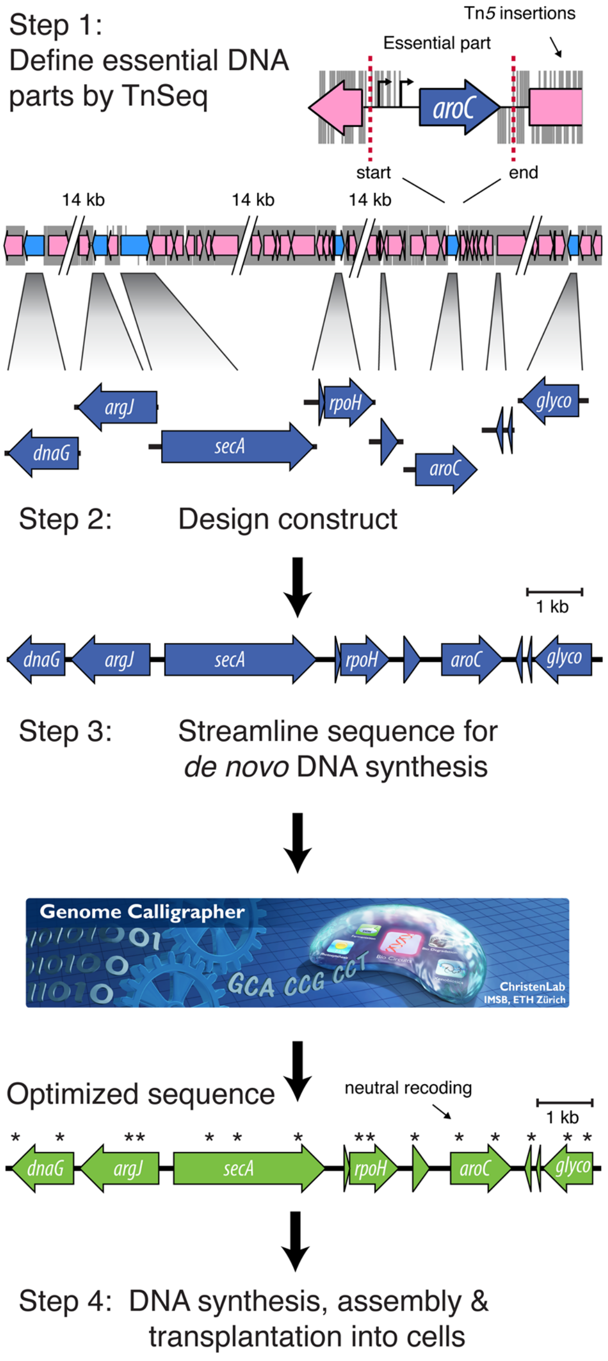


Figure 3. Implementation of the Genome Calligrapher web tool into the design−refactor−synthesis workflow of synthetic biology to compile synthetic genome constructs. (Step 1) Experimental systems biology approaches are used in conjunction with bioinformatics approaches to identify DNA parts. Hypersaturated transposon mutagenesis coupled to high-throughput sequencing (TnSeq) is used to identify the entire list of essential and high-fitness DNA parts required for rich media growth of the model bacterium *Caulobacter crescentus*. (Step 2) Parts are concatenated, with order and orientation maintained as found on the original wild-type genome, and compiled into a synthetic essential genome construct (GenBank file). (Step 3) The synthetic essential genome sequence file is then refactored for *de novo* DNA synthesis using the Genome Calligrapher algorithm. (Step 4) The optimized sequence is synthesized by low cost *de novo* DNA synthesis. Refactored sequences, where codons have been optimized to meet synthesis criteria, are highlighted with (*).

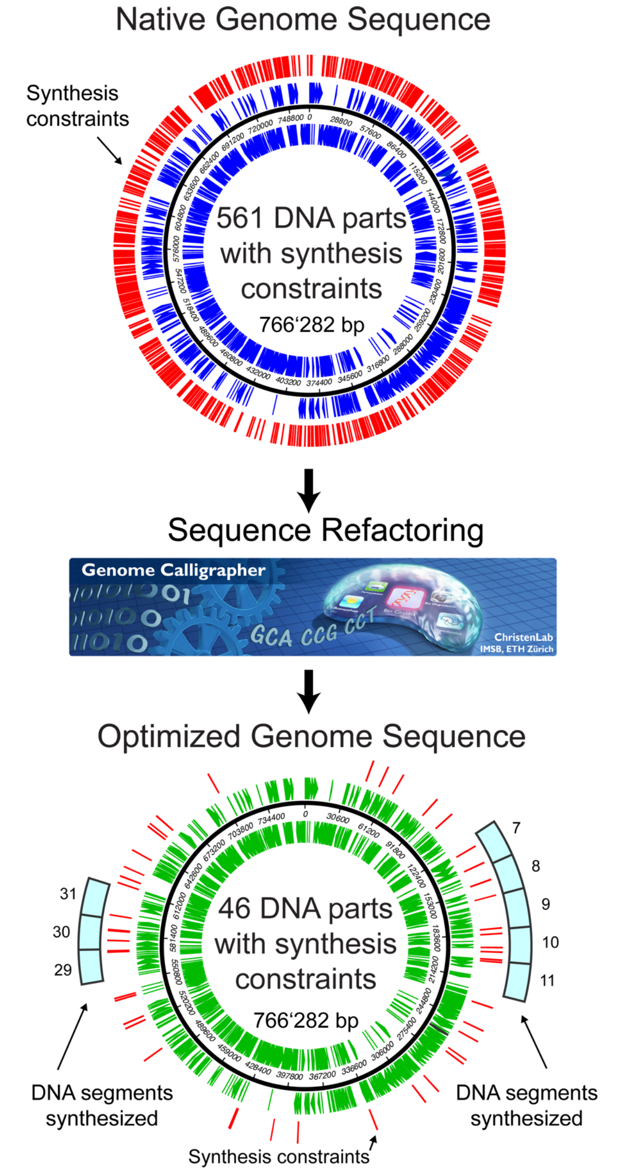


Figure 4. Synthesis feasibility analysis of native and sequence-optimized essential *Caulobacter crescentus* genomes. The upper panel shows the native genome sequence with CDS on the reverse and forward strands plotted in blue. The 561 (out of 1017) DNA parts that failed DNA synthesis criteria are plotted in red. The lower panel shows the genome sequence after refactoring with the Genome Calligrapher algorithm. Sequence optimized CDS are plotted in green on the inner and outer circles, respectively. The 46 DNA remaining DNA parts that fail *de novo* synthesis constraints are plotted in red.

design.[33,34] Furthermore, the freely accessible web tool interface provides connection points to integrate Genome Calligrapher into the computer-aided design−built−test cycle of synthetic biology.

■ METHODS

Web Tool Availability and License. The Genome Calligrapher web tool is available free-of-charge for non-commercial (e.g., academic, nonprofit, or government) use under an ETH Zürich end-user license agreement. The Genome Calligrapher software can be accessed through the public Genome Calligrapher web site (https://christenlab.ethz.ch/GenomeCalligrapher) and is also available for download

Table 3. Results from the Sequence Refactoring of the Eight 20 kb Synthetic Essential Genome Test Segments and Their Performance in *de Novo* DNA Synthesis

		GC content violations[a]				no. of				recoding[b]		
name	size (bp)	99 bp		21 bp		exceptions[c]		repeats[d]		codons	(%)	synthesis success rate[e]
seg_7	20 593	0	(153)	13	(65)	0	(5)	0	(0)	305/5683	5.37	8/8
seg_8	19 317	0	(162)	1	(78)	0	(3)	0	(1)	372/5184	7.18	7/7
seg_9	21 367	0	(236)	3	(106)	0	(1)	0	(2)	530/5965	8.89	7/8[f]
seg_10	20 370	0	(279)	0	(86)	0	(0)	0	(1)	533/5264	10.13	8/8
seg_11	20 828	0	(266)	3	(115)	0	(0)	0	(0)	570/6349	8.98	8/8
seg_29	19 025	0	(219)	1	(98)	0	(2)	0	(0)	493/4924	10.01	7/7
seg_30	19 403	0	(273)	2	(99)	0	(3)	0	(0)	571/5505	10.37	5/7[f]
seg_31	20 247	0	(138)	0	(56)	0	(1)	0	(0)	295/5750	5.13	7/8[f]
sum	161 150	0	(1726)	23	(703)	0	(15)	0	(4)	3369/44623	8.22	57/61

[a] GC content violations before (in brackets) and after sequence optimization with the Genome Calligrapher algorithm are shown. GC limits applied during recoding were 0.3 and 0.7 for the 99 bp window and 0.15 and 0.85 for the 21 bp window. These GC limits were set to fulfill sequence requirements of most commercial providers of *de novo* DNA synthesis. [b] Number and percentage of recoding needed within each DNA segments for removal of *de novo* DNA synthesis constraints. [c] Number of homopolymeric, di- and trinucleotide repeats. [d] Number of hairpins and direct repeats detected before and after recoding, with a minimal repeat size of 12 and maximal spacer length of 20 base pairs. [e] Success rate of synthesizing 3 kb DNA fragments. [f] Four out of 61 fragments failed synthesis in the first batch but were successfully synthesized in a second synthesis batch.

upon request. For further information, please refer to the public Genome Calligrapher web site.

Genome Calligrapher Web Tool Implementation. The Genome Calligrapher web tool runs on a server cluster with web interface implementation in PHP. The Genome Calligrapher algorithm is written in Python programming language (https://python.org) and utilizes the Biopython package[35] to parse GenBank files and matplotlib (matplotlib.org) to generate graphic output files. A complete description of the algorithm and additional functionalities can be found in the Supporting Information and Methods.

Calculation of Codon Usage Tables. A total of 2776 complete bacterial genomes were downloaded from NCBI (as deposited per November 2014), and codon tables of each bacterial species were calculated by analyzing the codon frequency over all protein-coding genes encoded on the chromosome or on plasmids using a custom Python script. Separate codon tables were calculated for each bacterial species for which genome sequences are available from multiple isolates or serovar types.

Design, Refactoring and Partial Synthesis of the Synthetic Essential Genome Construct. Detailed information on sequence design, essential DNA part list, sequence optimization parameters used for refactoring, DNA partitioning and DNA synthesis success rates are listed in Supporting Information and Methods, Data SI, Tables S1 and S5.

■ ASSOCIATED CONTENT

Ⓢ Supporting Information

Additional materials and methods and data, including Tables S1−S5 and Figure S1. The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acssynbio.5b00087.

■ AUTHOR INFORMATION

Corresponding Author

*Tel: ++41 44 633 64 44; Fax: +4144 633 10 51; E-mail: beat.christen@imsb.biol.ethz.ch.

Author Contributions

B.C. and M.C. designed and developed the software and Web server. B.C and M.C. performed the bioinformatics synthesis feasibility analysis across bacterial genomes; B.C, M.C., and S.D. designed the experiments; S.D. performed DNA partitioning, synthesis, and subsequent assembly into eight 20 kb long fragments; and B.C., M.C., and S.D. wrote the manuscript.

Notes

The authors declare the following competing financial interest(s): B.C. and M.C. have pending patent applications whose value may be affected by the publication of this article.

■ ACKNOWLEDGMENTS

This work was supported by the Eidgenössische Technische Hochschule (ETH) Zürich and a Community Science Program (CSP) DNA synthesis award [JGI CSP-1593 to B.C. and M.C.] from the U.S. Department of Energy Joint Genome Institute in Walnut Creek. CA, USA. The work conducted by the U.S. Department of Energy Joint Genome Institute, a DOE Office of Science User Facility, is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. We would like to thank Dr. Heinz Christen for the Genome Calligrapher algorithm development, Alex Neckelmann and Dr. Sangeeta Nath (Joint Genome Institute) for their help with the DNA synthesis and assembly, Dr. Adam Clore (Integrated DNA Technologies) for providing the synthesis feasibility analysis for wild-type and synthetic genome constructs, Layla Lang (http://laylalang.com) for the design of the Genome Calligrapher logo and illustration, and Dr. Bridget Kulasekara, Dr. Hemantha Kulasekara (University of Washington), and Mirjam Christen for critical reading of the manuscript.

■ REFERENCES

(1) Gibson, D. G., and Venter, J. C. (2014) Synthetic biology: construction of a yeast chromosome. *Nature 509*, 168−169.

(2) Karas, B. J., Molparia, B., Jablanovic, J., Hermann, W. J., Lin, Y.-C., Dupont, C. L., Tagwerker, C., Yonemoto, I. T., Noskov, V. N., Chuang, R.-Y., Allen, A. E., Glass, J. I., Hutchison, C. A., Smith, H. O., Venter, J. C., and Weyman, P. D. (2013) Assembly of eukaryotic algal chromosomes in yeast. *J. Biol. Eng. 7*, 30.

(3) Gibson, D. G., Glass, J. I., Lartigue, C., Noskov, V. N., Chuang, R.-Y., Algire, M. A., Benders, G. A., Montague, M. G., Ma, L., Moodie, M. M., Merryman, C., Vashee, S., Krishnakumar, R., Assad-Garcia, N., Andrews-Pfannkoch, C., Denisova, E. A., Young, L., Qi, Z.-Q., Segall-Shapiro, T. H., Calvey, C. H., Parmar, P. P., Hutchison, C. A., Smith,

H. O., and Venter, J. C. (2010) Creation of a bacterial cell controlled by a chemically synthesized genome. *Science 329*, 52−56.

(4) Gibson, D. G., Smith, H. O., Hutchison, C. A., Venter, J. C., and Merryman, C. (2010) Chemical synthesis of the mouse mitochondrial genome. *Nat. Methods* 7, 901−903.

(5) Czar, M. J., Anderson, J. C., Bader, J. S., and Peccoud, J. (2009) Gene synthesis demystified. *Trends Biotechnol. 27*, 63−72.

(6) Kosuri, S., and Church, G. M. (2014) Large-scale de novo DNA synthesis: technologies and applications. *Nat. Methods 11*, 499−507.

(7) Gellert, M., Lipsett, M. N., and Davies, D. R. (1962) Helix formation by guanylic acid. *Proc. Natl. Acad. Sci. U.S.A. 48*, 2013−2018.

(8) Jensen, M. A., Fukushima, M., Davis, R. W., and Deb, S. (2010) DMSO and betaine greatly improve amplification of GC-rich constructs in de novo synthesis. *PLoS One 5*, e11024.

(9) Hughes, R. A., Miklos, A. E., and Ellington, A. D. (2011) Gene synthesis: methods and applications. *Methods Enzymol. 498*, 277−309.

(10) McDowell, D. G., Burns, N. A., and Parkes, H. C. (1998) Localised sequence regions possessing high melting temperatures prevent the amplification of a DNA mimic in competitive PCR. *Nucleic Acids Res. 26*, 3340−3347.

(11) Gould, N., Hendy, O., and Papamichail, D. (2014) Computational tools and algorithms for designing customized synthetic genes. *Front. Bioeng. Biotechnol.* 2, 41.

(12) Wu, G., Bashir-Bello, N., and Freeland, S. J. (2006) The Synthetic Gene Designer: a flexible web platform to explore sequence manipulation for heterologous expression. *Protein Expression Purif. 47*, 441−445.

(13) Puigbò, P., Guzmán, E., Romeu, A., and Garcia-Vallvé, S. (2007) OPTIMIZER: a web server for optimizing the codon usage of DNA sequences. *Nucleic Acids Res. 35*, W126−31.

(14) Chin, J. X., Chung, B. K.-S., and Lee, D.-Y. (2014) Codon Optimization OnLine (COOL): a web-based multi-objective optimization platform for synthetic gene design. *Bioinformatics 30*, 2210−2212.

(15) Hoover, D. M., and Lubkowski, J. (2002) DNAWorks: an automated method for designing oligonucleotides for PCR-based gene synthesis. *Nucleic Acids Res. 30*, e43.

(16) Komar, A. A. (2009) A pause for thought along the co-translational folding pathway. *Trends Biochem. Sci. 34*, 16−24.

(17) Mehl, R. A., Anderson, J. C., Santoro, S. W., Wang, L., Martin, A. B., King, D. S., Horn, D. M., and Schultz, P. G. (2003) Generation of a bacterium with a 21 amino acid genetic code. *J. Am. Chem. Soc. 125*, 935−939.

(18) Lajoie, M. J., Rovner, A. J., Goodman, D. B., Aerni, H.-R., Haimovich, A. D., Kuznetsov, G., Mercer, J. A., Wang, H. H., Carr, P. A., Mosberg, J. A., Rohland, N., Schultz, P. G., Jacobson, J. M., Rinehart, J., Church, G. M., and Isaacs, F. J. (2013) Genomically recoded organisms expand biological functions. *Science 342*, 357−360.

(19) McAdams, H. H., and Shapiro, L. (2003) A bacterial cell-cycle regulatory network operating in time and space. *Science 301*, 1874−1877.

(20) Ochsner, A. M., Sonntag, F., Buchhaupt, M., Schrader, J., and Vorholt, J. A. (2015) Methylobacterium extorquens: methylotrophy and biotechnological applications. *Appl. Microbiol. Biotechnol. 99*, 517−534.

(21) Lajoie, M. J., Kosuri, S., Mosberg, J. A., Gregg, C. J., Zhang, D., and Church, G. M. (2013) Probing the limits of genetic recoding in essential genes. *Science 342*, 361−363.

(22) Christen, B., Abeliuk, E., Collier, J. M., Kalogeraki, V. S., Passarelli, B., Coller, J. A., Fero, M. J., McAdams, H. H., and Shapiro, L. (2011) The essential genome of a bacterium. *Mol. Syst. Biol. 7*, 528−528.

(23) Keasling, J. D. (2008) Synthetic biology for synthetic chemistry. *ACS Chem. Biol. 3*, 64−76.

(24) Breitling, R., and Takano, E. (2015) Synthetic biology advances for pharmaceutical production. *Curr. Opin. Biotechnol. 35C*, 46−51.

(25) Doroghazi, J. R., Albright, J. C., Goering, A. W., Ju, K.-S., Haines, R. R., Tchalukov, K. A., Labeda, D. P., Kelleher, N. L., and Metcalf, W. W. (2014) A roadmap for natural product discovery based on large-scale genomics and metabolomics. *Nat. Chem. Biol. 10*, 963−968.

(26) Kliebenstein, D. J. (2014) Synthetic biology of metabolism: using natural variation to reverse engineer systems. *Curr. Opin. Plant Biol. 19*, 20−26.

(27) Smanski, M. J., Bhatia, S., Zhao, D., Park, Y., B A Woodruff, L., Giannoukos, G., Ciulla, D., Busby, M., Calderon, J., Nicol, R., Gordon, D. B., Densmore, D., and Voigt, C. A. (2014) Functional optimization of gene clusters by combinatorial design and assembly. *Nat. Biotechnol. 32*, 1241−1249.

(28) Quin, M. B., and Schmidt-Dannert, C. (2014) Designer microbes for biosynthesis. *Curr. Opin. Biotechnol. 29*, 55−61.

(29) Tseng, H.-C., and Prather, K. L. J. (2012) Controlled biosynthesis of odd-chain fuels and chemicals via engineered modular metabolic pathways. *Proc. Natl. Acad. Sci. U.S.A. 109*, 17925−17930.

(30) Frasch, H.-J., Medema, M. H., Takano, E., and Breitling, R. (2013) Design-based re-engineering of biosynthetic gene clusters: plug-and-play in practice. *Curr. Opin. Biotechnol. 24*, 1144−1150.

(31) Richardson, S. M., Wheelan, S. J., Yarrington, R. M., and Boeke, J. D. (2006) GeneDesign: rapid, automated design of multikilobase synthetic genes. *Genome Res. 16*, 550−556.

(32) Villalobos, A., Ness, J. E., Gustafsson, C., Minshull, J., and Govindarajan, S. (2006) Gene Designer: a synthetic biology tool for constructing artificial DNA segments. *BMC Bioinf.* 7, 285.

(33) Hillson, N. J., Rosengarten, R. D., and Keasling, J. D. (2012) j5 DNA assembly design automation software. *ACS Synth. Biol. 1*, 14−21.

(34) Appleton, E., Tao, J., Haddock, T., and Densmore, D. (2014) Interactive assembly algorithms for molecular cloning. *Nat. Methods 11*, 657−662.

(35) Cock, P. J. A., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B., and de Hoon, M. J. L. (2009) Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics 25*, 1422−1423.