

AutoESD: a web tool for automatic editing sequence design for genetic manipulation of microorganisms

Yi Yang^{1,2,3,†}, Yufeng Mao^{1,2,†}, Ruoyu Wang^{1,2,†}, Haoran Li^{1,2}, Ye Liu², Haijiao Cheng², Zhenkun Shi^{1,2}, Yu Wang², Meng Wang², Ping Zheng², Xiaoping Liao^{1,2,*} and Hongwu Ma^{1,2,3,*}

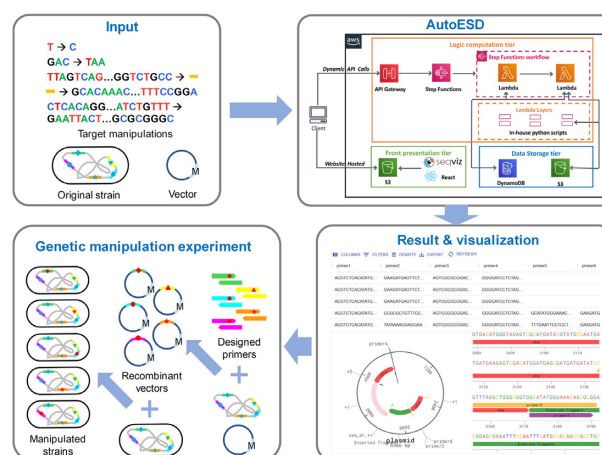
¹Biodesign Center, Key Laboratory of Systems Microbial Biotechnology, Tianjin Institute of Industrial Biotechnology, Chinese Academy of Sciences, Tianjin 300308, China, ²Tianjin Institute of Industrial Biotechnology, Chinese Academy of Sciences, Tianjin 300308, China and ³University of Chinese Academy of Sciences, Beijing 100049, China

Received February 27, 2022; Revised April 20, 2022; Editorial Decision May 05, 2022; Accepted May 09, 2022

ABSTRACT

Advances in genetic manipulation and genome engineering techniques have enabled on-demand targeted deletion, insertion, and substitution of DNA sequences. One important step in these techniques is the design of editing sequences (e.g. primers, homologous arms) to precisely target and manipulate DNA sequences of interest. Experimental biologists can employ multiple tools in a stepwise manner to assist editing sequence design (ESD), but this requires various software involving non-standardized data exchange and input/output formats. Moreover, necessary quality control steps might be overlooked by non-expert users. This approach is low-throughput and can be error-prone, which illustrates the need for an automated ESD system. In this paper, we introduce AutoESD (<https://autoesd.biodesign.ac.cn/>), which designs editing sequences for all steps of genetic manipulation of many common homologous-recombination techniques based on screening-markers. Notably, multiple types of manipulations for different targets (CDS or intergenic region) can be processed in one submission. Moreover, AutoESD has an entirely cloud-based serverless architecture, offering high reliability, robustness and scalability which is capable of parallelly processing hundreds of design tasks each having thousands of targets in minutes. To our knowledge, AutoESD is the first cloud platform enabling precise, automated, and high-throughput ESD across species, at any genomic locus for all manipulation types.

GRAPHICAL ABSTRACT



INTRODUCTION

Genetic engineering of organisms with desirable novel functionalities such as high productivity of valuable biochemicals is a key research objective of synthetic biology (1). Various genetic manipulation and genome engineering techniques have been developed for targeted deletion, insertion and substitution of DNA sequences in the genome of an existing organism (2,3). Among them, the scarless screening-marker-based homologous-recombination (HR) system and the RNA-guided CRISPR/Cas-based system are the most widely used ones (4,5). One common and important step in these techniques is the design of editing sequences to precisely target and edit DNA sequences of interest, such as the design of primers to locate and amplify the DNA fragment to be modified, or sgRNAs to guide the Cas enzyme to the exact site for cutting DNA in CRISPR/Cas-

*To whom correspondence should be addressed. Tel: +86 22 24828758; Email: liao_xp@tib.cas.cn
Correspondence may also be addressed to Hongwu Ma. Tel: +86 22 24828735; Email: ma_hw@tib.cas.cn

†The authors wish it to be known that, in their opinion, the first three authors should be regarded as Joint First Authors.

based system. Editing sequences design (ESD) is not trivial and can be error-prone for a number of reasons. First, due to the existence of various technological variants suitable for different organisms with diverse parameter configurations (e.g. lengths of homologous arms, preferences for carriers of donor DNA), different researchers might design different editing sequences even for the same manipulation objective, potentially leading to failure. Second, quality control steps, such as primers for verification of HR events and off-target risk assessment, are often overlooked, possibly resulting in editing sequences with low efficiency. Third, certain complex steps (e.g. DNA assembly for heterogenous fragment insertion) might introduce mistakes if multiple interdependent pairs of primers are required.

Several computer-aided design (CAD) tools have been developed for ESD (6–8). For example, PrecisePrimer (9) and Primer3 (10) can perform primer design for DNA fragment amplification. CHOPCHOP (11) and gBIG (12) are developed to design precise guide RNAs for CRISPR/Cas system. J5 (13) and Raven (14) can assist the primer design for vector DNA assembly. Experimental biologists can employ multiple individual CAD tools in a stepwise manner to assist ESD, but this requires multiple software packages involving non-standardized data exchange and input/output formats. Instead, a one-stop service covering all the experimental steps is a better alternative and was shown to be effective with some recently developed tools. For example, GeneTargeter (15) and Yestriction v0.1 (16) were developed for whole-process ESD in the case of gene knock-out using the CRISPR/Cas HR system in *Plasmodium falciparum* and *Saccharomyces cerevisiae*, respectively. GEDpmcg (17) provides one-stop ESD for the construction of point mutations in the genome of *Corynebacterium glutamicum* based on two rounds of single crossover HR. However, these tools were developed explicitly for limited genetic manipulation types within particular species. Moreover, these tools can only choose genes as editing targets, limiting their application potential, since non-coding sequences such as promoters or ribosomal binding sites (RBSs) are also very important manipulation targets.

To achieve target editing with any manipulation type (deletion, insertion or substitution) at any genomic locus across species, we chose the classical screening-marker-based HR (SMHR) system, which is a reliable genetic manipulation technique applicable in most microorganisms (5,18–20). For example, for two rounds of single crossover HR mediated by the suicide plasmid-mediated is the most widely-used HR system in *C. glutamicum* (20,21), while two rounds of cascade double/single crossover are preferred in *Bacillus subtilis* (5). For species with high transform efficiency, such as *Saccharomyces cerevisiae* (18) and *Escherichia coli* (19), donor DNA is often provided in the form of PCR products to avoid the costs of plasmid construction. Moreover, with the rapid development of automated strain engineering systems and emerging biofoundries (22–24), it has become physically feasible to construct thousands of strains within months or even weeks (25,26), increasing the demand for precise, automated and high-throughput design of editing sequences (7,27). In this paper, we introduce AutoESD (<https://autoesd.biodesign.ac.cn/>), the first cloud-based CAD tool for editing sequence design. Specif-

ically, we developed flexible workflows to support various technological variants of SMHR. We integrated all editing sequence design steps covering the whole process of genetic manipulation using many techniques into a standardized one-stop service, resulting in an automated high-throughput *in silico* design platform. Moreover, AutoESD has an entirely cloud-based serverless architecture, offering high reliability, robustness and scalability. We believe that AutoESD combined with automated genetic engineering platforms will pave the way for an ultrafast design-build-test-learn synthetic biology cycle in the future.

RESULTS

Genetic manipulation principles and computational design workflows

In most microorganisms, the SMHR system is a reliable genetic manipulation technique in which two rounds of HR events occur with the help of homologous arms. The screening markers will be integrated into the genome in the first HR event for screening, then removed in the second HR event, enabling the scarless editing of the target genome (4,5). According to the type of HR events (single or double crossover) and the form in which the homologous arms are provided (plasmid, PCR fragment, or oligonucleotide), we included five major technical variants in this study, namely plasmid-mediated single/single crossover, plasmid-mediated double/single crossover, fragment-mediated double/single crossover, fragment-mediated double/double crossover and oligonucleotide-mediated double/double crossover HR (Figure 1A, Supplementary Figures S1–S4 and S6–S12).

Taking the plasmid-mediated double/single crossover as an example (Figure 1A), if a 500 bp fragment is to be inserted into the genome of *Bacillus subtilis*, the whole procedure can be divided into three major steps. First, we need to assemble the intermediate sequence, which contains the upstream homologous arm (UHA), inserted sequence, screening marker, and downstream homologous arm (DHA). This requires the design of primers for UHA (P1/P2), DHA (P3/P4), screening marker (P5/P6), and inserted sequence (P7/P8). Second, the screening marker and the inserted sequence will be integrated into the genome in the first HR event for screening, followed by the use of testing primers (TP1/TP2) to verify the first HR event and confirm that the sequences were actually integrated into the genome. Third, the fragment with screening markers will be removed from the edited genome, after which another pair of testing primers (TP3/TP4) is required to confirm the final editing result.

In this study, we standardized the experimental process into a computational design workflow and coordinated functions corresponding to the dependencies in the experimental process as shown in Figure 1B:

Input: The workflow starts with an input handling model that processes the input file for target manipulation in a format similar to VCF (28), enabling users to integrate different types of sequence manipulations in a single file. Within this file, ‘Sequence upstream of the manipulation site (>100bp)’ rather than exact position or gene ID is used to locate the manipulation site in the given genome due

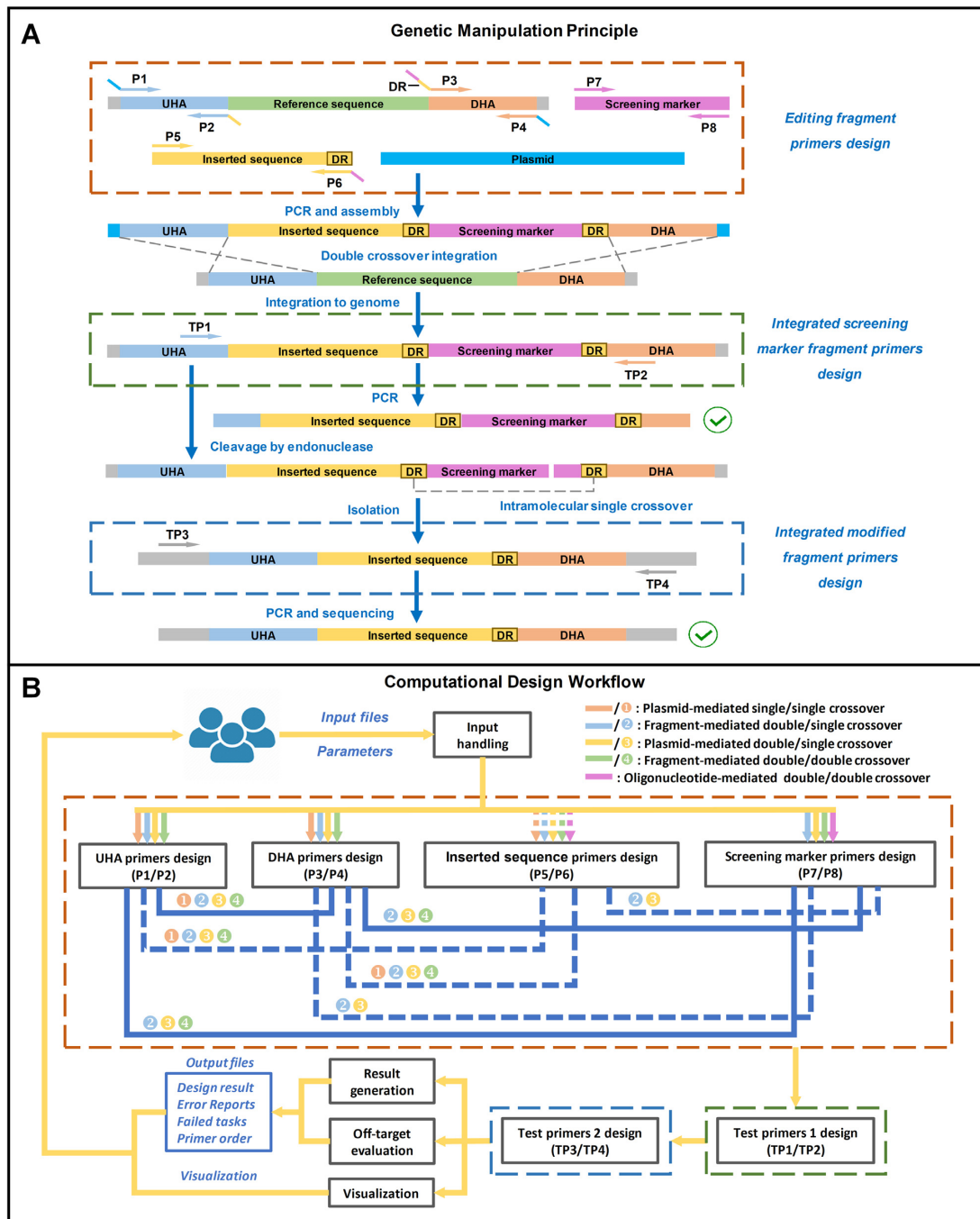


Figure 1. (A) Genetic manipulation principle; (B) computational design workflow of AutoESD.

to the following two advantages. First, for non-standard genome (e.g. a laboratory evolved strain) or unsequenced genomes, it is difficult to determine the position, but it is often easy to find the sequences around the site. Second, using the sequence instead of gene ID allows us to manipulate at any site, including noncoding regions such as promoters as well as coding sequences. Similarly, we also use sequences in the input file to indicate reference sequences and inserted sequences. An empty inserted sequence (represented as ‘-’ in the input file) implies a deletion, and an empty reference sequence implies an insertion. In this way, we can include

different types of sequence manipulations (deletion, insertion or substitution) in a single input file. Moreover, for some cases, such as deletion of a gene’s complete CDS, it is easy to obtain the exact chromosomal position from the genome annotation file (e.g. in GFF or GTF format). AutoESD also supports input files which include the exact position of manipulation site (a combination of chromosome, position and strand).

Assembly: Four primer design modules are needed for designing P1–P8 to obtain the assembled plasmid (Figure 1A). There are interdependencies between the primers (e.g.

20 bp overlaps between P2 and P5, 20 bp overlaps between P3 and P6). The major difference among these technical variants is that the required modules/parameters are not the same and the interdependencies between modules are slightly different (Figure 1B), making it quite difficult for non-expert users. For example, the screening marker primer design is unnecessary for plasmid-mediated single/single crossover, while the UHA/DHA primer design modules are not required for oligonucleotide-mediated double/double crossover.

First HR for integration: This step integrates the foreign sequences into the target site in the genome. Two technical variants: single crossover or double crossover, can be chosen according to their suitability for the chassis organism. Testing primers (TP1/TP2) need to be designed to verify the first HR event and confirm that the sequences were actually integrated into the genome. This step is often overlooked by inexperienced biologists, leading to failure at a later stage and a waste of time.

Second HR for marker removal: This step removes the screening marker sequences from the edited genome for a scarless genome manipulation. Similar to the first HR, single or double crossover can be chosen. However, the options may be restricted by the first HR event. Another pair of testing primers (TP3/TP4) is required to be designed to confirm the final editing result.

After designing all the editing sequences, off-target evaluation of the designed homologous arms is performed to help end-users assess the potential risk and adjust parameter settings if needed. In addition, detailed visualization of editing sequences is provided for the convenience of end-users doing quality checks online before placing the primer order. Although there are many differences in terms of technical variants, the modularized algorithm design approach allows us to assemble logic functions like building LEGO, generating a similar overall workflow (Figure 1B). By designing the underlying data structure and modules, we construct separate functions for all modules. Notably, one single function is created for each module to support all five different technical variants by different parameter settings. In this way, the whole workflow enables on-demand one-stop editing sequence design across technical variants, supporting various genetic manipulations across species. The core logical code of AutoESD is available on GitHub (<https://github.com/tibbdc/autoesd>).

Cloud platform implementation based on Amazon web services

Based on the workflow shown in Figure 1B, we have written several functions in python for different design modules and built a user-friendly web server for the complete design workflow using cloud-based serverless architecture, named AutoESD. This enables biologists without coding experience to use it for reliable, robust, and automatic editing sequence design. We used a three-tier architecture (the front presentation tier, logic computation tier, and data storage tier), as shown in Figure 2, to build our web server on Amazon Web Services (AWS). The data storage tier manages persistent storage of our platform, including AWS DynamoDB and AWS S3 for storing user-uploaded input files,

parameters and jobs. The front presentation tier represents the components users directly interact with, which is hosted by the AWS S3 static website functionality and accelerated by AWS CloudFront. The seqviz package (<https://github.com/Lattice-Automation/seqviz>) was used to provide customizable visualization of design results, which can help end-users check the design results online. The logic computation tier manages requests from external systems and contains the core services such as AWS Lambda, AWS API Gateway and AWS Step Functions. The API Gateway handles the HTTP requests and routes them to the correct backend. AWS Step Functions orchestrate the serverless workflow by processing messages from the API Gateway and invoking AWS Lambda asynchronously. AWS Lambda provides core computation functionality with main functions to handle different requests. More specifically, the Lambda function gets the corresponding input files and parameters from AWS S3, and calls the workflow function in python to obtain the design results, which will be uploaded to AWS S3 as well.

In our platform, uploaded input files will be stored on an AWS S3 bucket. When the submit button is clicked after all parameters are set, a request is sent to the API Gateway, which passes the parameters to the AWS Step Function, and all parameters are stored on AWS DynamoDB. Then, the browser gets the response, jumps to the results page, and waits for the computing results. AWS Lambda is invoked asynchronously by AWS Step Function event sources, which runs the logic code and uploads the result files to AWS S3. Each submission will trigger a parallel computing process, regardless of how much demand there is on the website, showcasing the great scalability of serverless computing. To further show the robustness of our platform, we perform a large-scale simulation test. We simulated a group of *in silico* experiments, including 100, 200, 300, 400, 500, 800, 1000 and 1500 concurrent design requirements, each containing 2000 genetic manipulation targets. The results showed that the overall running time of the platform is not correlated with the number of tasks from 100 to 800 (Supplementary Figure S5). The total time starts to increase as the number of tasks exceeds 1000, and this is because AWS sets a concurrency limit of 1000 for each user-defined Lambda function. Moreover, it is worth noting that user inputs and outputs are only visible to users themselves, and will be stored on AWS server for no more than one week.

Case study

B. subtilis is one of the most important industrial protein producers and a promising generally recognized as safe (GRAS) chassis for the biosynthesis of various biochemicals (29,30). Here, we show an example on how AutoESD can be used to design editing sequences for engineering *B. subtilis* to produce meso-2,3-butanediol (2,3-BD), a platform chemical and promising alternative biofuel (31). Six manipulations (deletion of *bdhA*, *acoA*, *pta*, *ldh*; substitution of the promoter for the precursor biosynthetic operon *alsSD* with a strong promoter P_{43} , and insertion of the heterologous *budC* gene under the control of P_{43} at the *amyE* locus) have been reported as effective engineering strategies for 2,3-BD production (31). To design editing sequences for

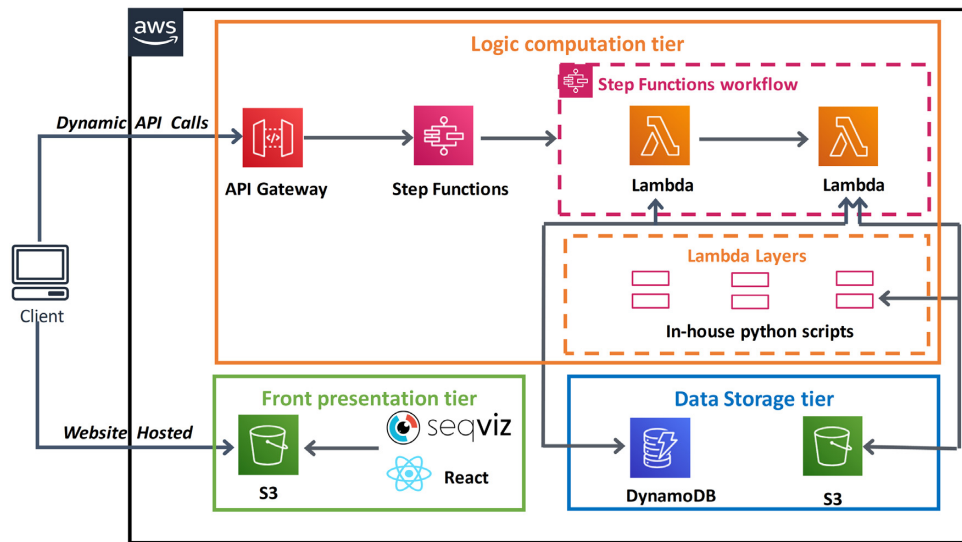


Figure 2. The software architecture of AutoESD.

these manipulations, users first need to prepare the input file including sequence data related to these genes (Figure 3A and Supplementary file 1). As mentioned before, we used sequences to locate the manipulation site. It should also be noted that both the sequences on positive and negative strands are acceptable. Still, the sequences for any single manipulation target should be provided in the same direction. Specifically, although the genes *bdhA* and *pta* are located in the antisense strand, the ‘Sequence upstream of the manipulation site (>100 bp)’ is still the upstream sequence if we follow the direction of the respective open reading frames of *bdhA* and *pta*. Similarly, the ‘Reference sequence’ is also the coding sequence for these two genes. For the substitution of the promoter sequence of the *alsSD* operon, which is also located on the antisense strand, the ‘Inserted sequence’ should be the P_{43} sequence following the same direction as the *alsSD* operon. For the heterologous *budC* gene to be inserted into the *amyE* locus on the positive strand, the ‘inserted sequence’ is the P_{43} sequence and its coding sequence. By including multiple sequence manipulations in one input file, AutoESD allows the users to automatically design editing sequences covering different manipulation types at arbitrary genomic loci, encompassing more than the CDS region.

When using AutoESD, end-users need to clarify the targeted *B. subtilis* strains by selecting/uploading the genome sequence, the specific HR variant by uploading the linear plasmid sequence, and the screening marker sequence (Figure 3A). In this showcase, we selected one variant from two rounds of double/single crossover HR systems, which is dependent on the suicide vector pSS and counter-selectable marker *upp* (encoding uracil phosphoribosyl transferase) (32). For this technical variant, the endogenous gene *upp* is usually deleted in advance to enable its counter-selection. Accordingly, the genome of *B. subtilis* 168 Δupp was chosen as an example. Then, users need to sequentially select ‘Plasmid-mediated’, ‘Double’ and ‘Single’ on the website (Figure 3A) and upload a linear pSS sequence to clarify this

variant (Supplementary file 2) and the ‘Target Manipulations’ file (Supplementary file 1). Once all four inputs are uploaded and submitted, AutoESD will perform an initial input check and then invoke the downstream ESD workflow. The results, including all the designed primers, can be downloaded or checked online (Figure 3B). Notably, if the user prefers a rapid identification of the effect of engineering targets on the 2,3-BD production, they can change the option of ‘Screening marker removal’ from the default ‘Yes’ to ‘No’ to obtain the simplified designed results without those needed in the second HR event (Figure 3A and Supplementary Figure S9). AutoESD provides an interactive sequence visualization interface for checking target manipulation, designed primers, and other detailed information (e.g. assembled recombinant plasmid, two rounds of verified PCR products, the location of all primers) (Figure 3C and D). To make the genetic manipulation more traceable, the genome sequence FASTA file after any single manipulation can also be downloaded. In addition, to avoid potential off-target events, the output file (evaluation result) provides the target with potential off-target risk as the UHA/DHA can also be aligned to other regions in the target genome. The user can try to lower the risks of off-target events by altering the lengths of the UHA/DHA in the parameter settings. Moreover, to improve the reusability of primers, user-specified test primers, which have been typically used in their labs, are allowed to be uploaded (Figure 3A and Supplementary Figure S6). These design and quality check functions show that AutoESD provides reliable automated editing sequence design for all the manipulation types at both coding and intergenic regions.

It is worth mentioning that plasmid-mediated single/single crossover HR is also a widely used technique in *B. subtilis*. This can be implemented using the suicide plasmid pCU, which carries *upp* as the counter-selectable marker and the *cat* (chloramphenicol resistant gene) as a direct selection marker (31). For the same engineering targets for 2,3-BD biosynthesis, end-users only need to

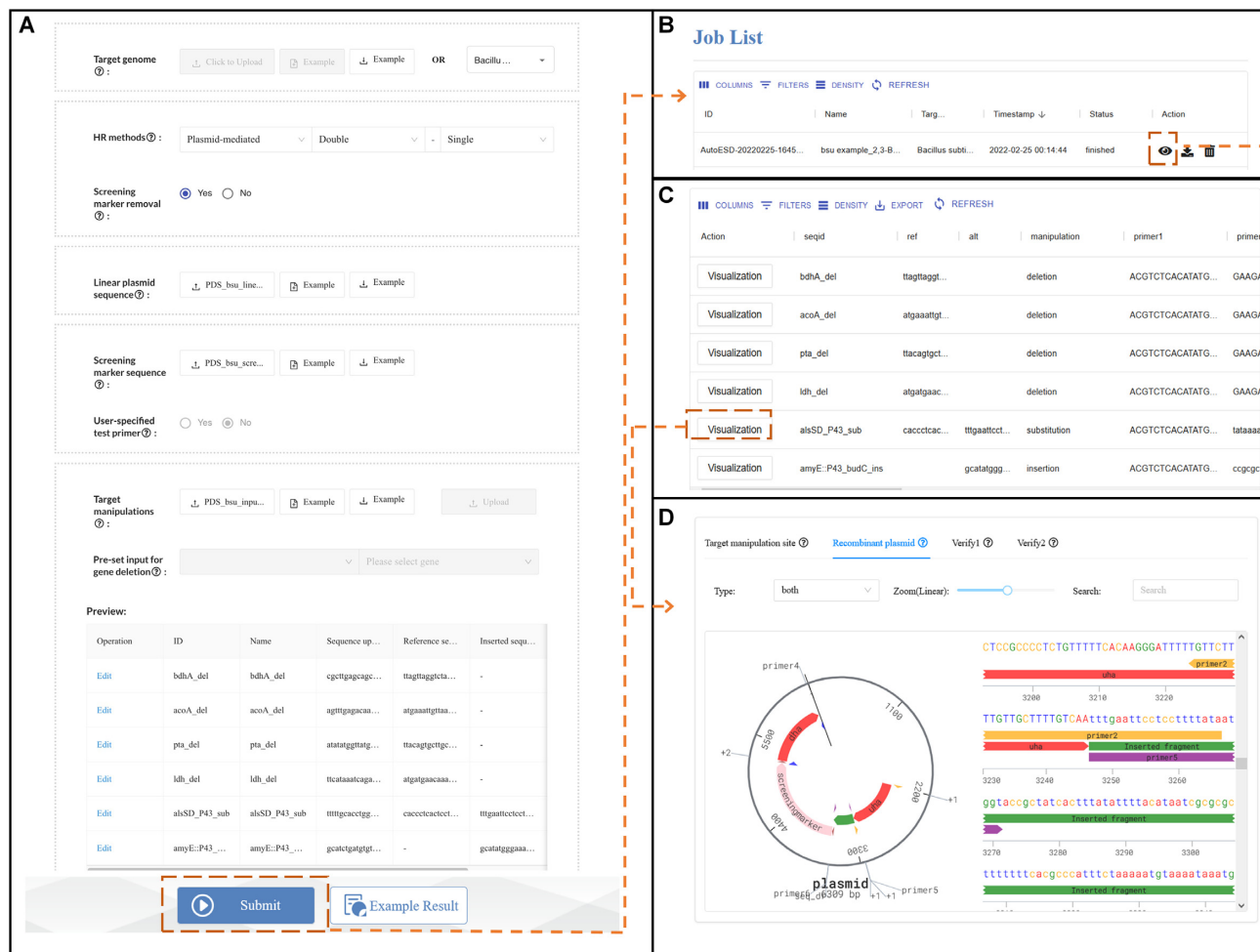


Figure 3. Online operation flow for running AutoESD. (A) Submission; (B) job manager; (C) design results; (D) detailed visualization.

change the first crossover from ‘Double’ to ‘Single’ on the website and change the input file of the plasmid sequence to the linear pCU sequence (Supplementary file 3). The options to choose different crossover techniques allow the user to select suitable HR variants for different species, enabling the application of AutoESD across different organisms. We tested AutoESD in five species using all technical variants with thousands of *in silico* generated genetic manipulations covering all the types of genetic manipulation in the corresponding genomes (all high-throughput input examples are provided on the website). All the design tasks were completed within minutes (Table 1), indicating the high-throughput editing sequence design capacity of AutoESD.

DISCUSSION

In the synthetic biology era, the massive scale of trial-and-error experiments in engineering organisms goes far beyond the traditional labor-intensive research paradigm, motivating the development of revolutionary biofoundries (22,24). ESD automation will play an indispensable role in bridging the model-based system-level genotype design step (33) with robot-assisted high-throughput strain construction steps in

an automated biofoundry (7,15,27,34). In this spirit, we developed the web tool AutoESD for the precise, automated and high-throughput design of editing sequences required for the SMHR mediated genetic manipulation of genome engineering targets. Compared with those ESD tools developed for specific genetic manipulation types in one or a few species (Supplementary Table S1), AutoESD standardized and modularized the experimental process into a computational design workflow (Figure 1B). As a result, AutoESD can easily support one-stop ESD for all the manipulation types, at any genomic loci and across species using various SMHR variants from one input file including information for all required manipulations through a user-friendly web interface (Figure 3A).

Moreover, AutoESD was built based on the cloud-based serverless architecture. In this way, we can build and run applications and services without provisioning or managing servers. This architecture has also been adopted in some other studies (35,36), showing its high reliability, robustness and scalability. AutoESD can support hundreds of end-users simultaneously submitting design jobs with over 2000 genetic manipulation targets per job, and all the jobs can be processed in parallel in minutes. To the best of our knowledge, AutoESD is the first tool that supports ESD cover-

Table 1. Running time of AutoESD for high-throughput tasks across species

Strain	Number of target manipulations in one submission	Technical variant	Total running time (s)
<i>Bacillus subtilis</i> 168	1000	Plasmid-mediated double/single crossover	224
<i>Corynebacterium glutamicum</i> ATCC 13032	1000	Plasmid-mediated single/single crossover	196
<i>Escherichia coli</i> BL21(DE3)	1000	Fragment-mediated double/single crossover	208
<i>Saccharomyces cerevisiae</i> S288c	2000	Fragment-mediated double/double crossover	424
<i>Escherichia coli</i> K-12 MG1655	2000	Oligonucleotide-mediated double/double crossover	387

ing most genetic manipulation demands of synthetic biology and metabolic engineering. We believe that AutoESD combined with robotic genetic engineering platforms will pave the way for an accelerated design-build-test-learn synthetic biology cycle in the future biofoundries.

The SMHR system is chosen due to two major reasons: (i) it is a mature and reliable genetic manipulation technique in most microorganisms, and can be rapidly applied to those newly isolated non-model organisms; (ii) it can theoretically support the editability of targeted genome without the risk of introducing additional mutations/scar, and off-target risks can be avoided with an appropriate length of HR arms. At present, AutoESD can assist most experimental biologists with routine strain construction demands using SMHR variants suitable for their species. What's more, AutoESD can also assist biologists trying new technical variant to further improve editing efficiency. In addition to SMHR, there are also genetic manipulation techniques, such as CRISPR/Cas-based HR (16), base editing (25) and prime editing (37), as well as other DNA assembly methods such as type II endonuclease mediated Golden Gate method (38). Next step, we will include more technical variants, assembly methods in our design tool and eventually help biologists to achieve their functional manipulation goal.

DATA AVAILABILITY

The core logical code and sample dataset of AutoESD are available on GitHub (<https://github.com/tibbdc/autoesd>).

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENTS

We thank Drs Ting Shi, Yong Yu, Dingyu Liu and Wei Yuan (Tianjin Institute of Industrial Biotechnology, Chinese Academy of Sciences) for discussions on genetic manipulation techniques.

FUNDING

National Key Research and Development Program of China [2018YFA0900300]; National Natural Science Foundation of China [32101186, 21908239]; Tianjin Synthetic Biotechnology Innovation Capacity Improvement Project [TSBICIP-PTJS-001, TSBICIP-PTJJ-005]; Youth Innovation Promotion Association CAS; China Postdoctoral Science Foundation [2018M641658]. Funding for open access

charge: Tianjin Synthetic Biotechnology Innovation Capacity Improvement Project [TSBICIP-PTJS-001]

Conflict of interest statement. None declared.

REFERENCES

- Si, T., Chao, R., Min, Y., Wu, Y., Ren, W. and Zhao, H. (2017) Automated multiplex genome-scale engineering in yeast. *Nat. Commun.*, **8**, 15187.
- Nielsen, J. and Keasling, J.D. (2016) Engineering cellular metabolism. *Cell*, **164**, 1185–1197.
- Ding, Q., Diao, W., Gao, C., Chen, X. and Liu, L. (2020) Microbial cell engineering to improve cellular synthetic capacity. *Biotechnol. Adv.*, **45**, 107649.
- Wang, Q., Zhang, J., Al Makishah, N.H., Sun, X., Wen, Z., Jiang, Y. and Yang, S. (2021) Advances and perspectives for genome editing tools of *Corynebacterium glutamicum*. *Front. Microbiol.*, **12**, 654058.
- Dong, H. and Zhang, D. (2014) Current development in genetic engineering strategies of *Bacillus* species. *Microb. Cell Fact.*, **13**, 63.
- Appleton, E., Madsen, C., Roehner, N. and Densmore, D. (2017) Design automation in synthetic biology. *Cold Spring Harb. Perspect. Biol.*, **9**, a023978.
- Appleton, E., Densmore, D., Madsen, C. and Roehner, N. (2017) Needs and opportunities in bio-design automation: four areas for focus. *Curr. Opin. Chem. Biol.*, **40**, 111–118.
- Kalendar, R., Khassenov, B., Ramankulov, Y., Samuilova, O. and Ivanov, K.I. (2017) FastPCR: an *in silico* tool for fast primer and probe design and advanced sequence analysis. *Genomics*, **109**, 312–319.
- Pauthenier, C. and Faulon, J.L. (2014) PrecisePrimer: an easy-to-use web server for designing PCR primers for DNA library cloning and DNA shuffling. *Nucleic Acids Res.*, **42**, W205–W209.
- Untergasser, A., Cutcutache, I., Koressaar, T., Ye, J., Faircloth, B.C., Remm, M. and Rozen, S.G. (2012) Primer3—new capabilities and interfaces. *Nucleic Acids Res.*, **40**, e115.
- Montague, T.G., Cruz, J.M., Gagnon, J.A., Church, G.M. and Valen, E. (2014) CHOPCHOP: a CRISPR/Cas9 and TALEN web tool for genome editing. *Nucleic Acids Res.*, **42**, W401–W407.
- Wang, Y., Liu, Y., Li, J., Yang, Y., Ni, X., Cheng, H., Huang, T., Guo, Y., Ma, H., Zheng, P. *et al.* (2019) Expanding targeting scope, editing window, and base transition capability of base editing in *Corynebacterium glutamicum*. *Biotechnol. Bioeng.*, **116**, 3016–3029.
- Hillson, N.J., Rosengarten, R.D. and Keasling, J.D. (2012) J5 DNA assembly design automation software. *ACS Synth. Biol.*, **1**, 14–21.
- Appleton, E., Tao, J., Haddock, T. and Densmore, D. (2014) Interactive assembly algorithms for molecular cloning. *Nat. Methods*, **11**, 657–662.
- Cárdenas, P., Esherrick, L.Y., Chambonnier, G., Dey, S., Turlo, C.V., Nasamu, A.S. and Niles, J.C. (2022) GeneTargeter: automated *in silico* design for genome editing in the malaria parasite, *Plasmodium falciparum*. *CRISPR J.*, **5**, 155–164.
- Mans, R., van Rossum, H.M., Wijsman, M., Backx, A., Kuijpers, N.G.A., van den Broek, M., Daran-Lapujade, P., Pronk, J.T., van Maris, A.J.A. and Daran, J.-M.G. (2015) CRISPR/Cas9: a molecular Swiss army knife for simultaneous introduction of multiple genetic modifications in *Saccharomyces cerevisiae*. *FEMS Yeast Res.*, **15**, fov004.
- Yang, Y., Mao, Y., Liu, Y., Wang, R., Lu, H., Li, H., Luo, J., Wang, M., Liao, X. and Ma, H. (2021) GEDpm-cg: genome editing automated

- design platform for point mutation construction in *Corynebacterium glutamicum*. *Front. Bioeng. Biotechnol.*, **9**, 768289.
18. Wingler, L.M. and Cornish, V.W. (2011) Reiterative recombination for the in vivo assembly of libraries of multigene pathways. *Proc. Natl. Acad. Sci. U.S.A.*, **108**, 15135.
 19. Lin, Z., Xu, Z., Li, Y., Wang, Z., Chen, T. and Zhao, X. (2014) Metabolic engineering of *Escherichia coli* for the production of riboflavin. *Microb. Cell Fact.*, **13**, 104.
 20. Wang, T., Li, Y., Li, J., Zhang, D., Cai, N., Zhao, G., Ma, H., Shang, C., Ma, Q., Xu, Q. *et al.* (2019) An update of the suicide plasmid-mediated genome editing system in *Corynebacterium glutamicum*. *Microb. Biotechnol.*, **12**, 907–919.
 21. Bommareddy, R.R., Chen, Z., Rappert, S. and Zeng, A.-P. (2014) A *de novo* NADPH generation pathway for improving lysine production of *Corynebacterium glutamicum* by rational design of the coenzyme specificity of glyceraldehyde 3-phosphate dehydrogenase. *Metab. Eng.*, **25**, 30–37.
 22. Chao, R., Mishra, S., Si, T. and Zhao, H. (2017) Engineering biological systems using automated biofoundries. *Metab. Eng.*, **42**, 98–108.
 23. Hillson, N., Caddick, M., Cai, Y., Carrasco, J.A., Chang, M.W., Curach, N.C., Bell, D.J., Le Feuvre, R., Friedman, D.C., Fu, X. *et al.* (2019) Building a global alliance of biofoundries. *Nat. Commun.*, **10**, 2040.
 24. Tellechea-Luzardo, J., Otero-Muras, I., Goñi-Moreno, A. and Carbonell, P. (2022) Fast biofoundries: coping with the challenges of biomanufacturing. *Trends Biotechnol.*, <https://doi.org/10.1016/j.tibtech.2021.12.006>.
 25. Wang, Y., Liu, Y., Liu, J., Guo, Y., Fan, L., Ni, X., Zheng, X., Wang, M., Zheng, P., Sun, J. *et al.* (2018) MACBETH: multiplex automated *Corynebacterium glutamicum* base editing method. *Metab. Eng.*, **47**, 200–210.
 26. Tenhaef, N., Stella, R., Frunzke, J. and Noack, S. (2021) Automated rational strain construction based on high-throughput conjugation. *ACS Synth. Biol.*, **10**, 589–599.
 27. Densmore, D.M. and Bhatia, S. (2014) Bio-design automation: software + biology + robots. *Trends Biotechnol.*, **32**, 111–113.
 28. Danecek, P., Auton, A., Abecasis, G., Albers, C.A., Banks, E., DePristo, M.A., Handsaker, R.E., Lunter, G., Marth, G.T., Sherry, S.T. *et al.* (2011) The variant call format and VCFtools. *Bioinformatics*, **27**, 2156–2158.
 29. Gu, Y., Xu, X., Wu, Y., Niu, T., Liu, Y., Li, J., Du, G. and Liu, L. (2018) Advances and prospects of *Bacillus subtilis* cellular factories: from rational design to industrial applications. *Metab. Eng.*, **50**, 109–121.
 30. Park, S.A., Bhatia, S.K., Park, H.A., Kim, S.Y., Sudheer, P.D.V.N., Yang, Y.H. and Choi, K.Y. (2021) *Bacillus subtilis* as a robust host for biochemical production utilizing biomass. *Crit. Rev. Biotechnol.*, **41**, 827–848.
 31. Fu, J., Huo, G., Feng, L., Mao, Y., Wang, Z., Ma, H., Chen, T. and Zhao, X. (2016) Metabolic engineering of *Bacillus subtilis* for chiral pure meso-2,3-butanediol production. *Biotechnol. Biofuels*, **9**, 90.
 32. Shi, T., Wang, G., Wang, Z., Fu, J., Chen, T. and Zhao, X. (2013) Establishment of a markerless mutation delivery system in *Bacillus subtilis* stimulated by a double-strand break in the chromosome. *PLoS One*, **8**, e81370.
 33. Xu, N., Ye, C. and Liu, L. (2018) Genome-scale biological models for industrial microbial systems. *Appl. Microbiol. Biotechnol.*, **102**, 3439–3451.
 34. Karkaria, B.D., Fedorec, A.J.H. and Barnes, C.P. (2021) Automated design of synthetic microbial communities. *Nat. Commun.*, **12**, 672.
 35. Lee, B.D., Timony, M.A. and Ruiz, P. (2019) DNAnVisualization.org: a serverless web tool for DNA sequence visualization. *Nucleic Acids Res.*, **47**, W20–W25.
 36. Grzesik, P., Augustyn, D.R., Wyciślik, Ł. and Mrozek, D. (2022) Serverless computing in omics data analysis and integration. *Brief. Bioinf.*, **23**, bbab349.
 37. Hsu, J.Y., Grünewald, J., Szalay, R., Shih, J., Anzalone, A.V., Lam, K.C., Shen, M.W., Petri, K., Liu, D.R., Keith Joung, J. *et al.* (2021) PrimeDesign software for rapid and simplified design of prime editing guide RNAs. *Nat. Commun.*, **12**, 1034.
 38. Engler, C., Kandzia, R. and Marillonnet, S. (2008) A one pot, one step, precision cloning method with high throughput capability. *PLoS One*, **3**, e3647.