BIOINFORMATICS APPLICATIONS NOTE Vol. 30 no. 17 2014, pages 2508–2510
doi:10.1093/bioinformatics/btu335

Structural bioinformatics

Advance Access publication May 14, 2014

RiboMaker: computational design of conformation-based riboregulation

Guillermo Rodrigo[1,*] and Alfonso Jaramillo[1,2,*]

[1]Institute of Systems and Synthetic Biology, CNRS - Université d'Évry Val d'Essonne, F-91000 Évry, France and [2]School of Life Sciences, University of Warwick, Coventry CV4 7AL, United Kingdom

Associate Editor: Anna Tramontano

ABSTRACT

Motivation: The ability to engineer control systems of gene expression is instrumental for synthetic biology. Thus, bioinformatic methods that assist such engineering are appealing because they can guide the sequence design and prevent costly experimental screening. In particular, RNA is an ideal substrate to *de novo* design regulators of protein expression by following sequence-to-function models.

Results: We have implemented a novel algorithm, RiboMaker, aimed at the computational, automated design of bacterial riboregulation. RiboMaker reads the sequence and structure specifications, which codify for a gene regulatory behaviour, and optimizes the sequences of a small regulatory RNA and a 5′-untranslated region for an efficient intermolecular interaction. To this end, it implements an evolutionary design strategy, where random mutations are selected according to a physicochemical model based on free energies. The resulting sequences can then be tested experimentally, providing a new tool for synthetic biology, and also for investigating the riboregulation principles in natural systems.

Availability and implementation: Web server is available at http://ribomaker.jaramillolab.org/. Source code, instructions and examples are freely available for download at http://sourceforge.net/projects/ribomaker/.

Contact: Guillermo.Rodrigo@issb.genopole.fr or Alfonso.Jaramillo@warwick.ac.uk

Received on October 23, 2013; revised on April 10, 2014; accepted on May 8, 2014

1 INTRODUCTION

Regulatory RNAs have been identified at the core of many cellular processes in both prokaryotes and eukaryotes (Prasanth and Spector, 2007; Waters and Storz, 2009). Among all possible mechanisms, here we focus on regulatory RNAs that interact with a given mRNA to control protein expression. By applying engineering principles, we can design synthetic regulatory RNAs to create genetic modules from which to program novel functions in the cell (Isaacs *et al*., 2006). In bacteria, this can be achieved by means of a small RNA (sRNA), whose length varies from tens to hundreds nucleotides and which is highly structured. The sRNA interacts with the 5′-untranslated region (5′-UTR) of a given mRNA, as this region controls the ability to interact with the ribosome (Salis *et al*., 2009). This control of gene expression only relies on structural changes in the 5′-UTR, without any RNA processing, which makes it independent of any additional cellular machinery.

This riboregulatory mechanism allows harnessing physicochemical models to predict RNA interaction and function. We take advantage of RNA folding prediction methods (Hofacker *et al*., 1994; Mathews *et al*., 1999), as well as methods for predicting RNA interactions (Busch *et al*., 2008; Mückstein *et al*., 2006), to perform the *de novo* design of sRNAs and 5′-UTRs with a base-pair energy model. Then, the designed RNA systems can be exploited, for instance, to control metabolic pathways in bioproduction applications (Na *et al*., 2013).

In this work, we present an evolutionary method to design nucleic acid sequences implementing riboregulatory modules. For that, a multi-objective, combinatorial optimization algorithm is implemented, where the free energies and secondary structures of the system are considered. Importantly, fully synthetic sequences obtained by computational design have been already verified for functionality *in vitro* and in bacterial cells (Rodrigo *et al*., 2012; Yin *et al*., 2008). In contrast to our previous method (Rodrigo *et al*., 2012, 2013a), the present is not restricted to a neutral evolution within the inverse folding of predefined RNA secondary structures (Fontana and Schuster, 1998). Therefore, the full specification of the intramolecular structures of the species is not essential. This dramatically speeds up the optimization.

2 APPLICATION

RiboMaker is devised for the computational design of conformation-based riboregulation, and in particular of bacterial riboregulation. It designs the riboregulator (sRNA) and the targeted 5′-UTR of a given mRNA (Fig. 1). Moreover, sRNAs can be designed to activate or repress protein expression. It can also design only one element (5′-UTR or sRNA) provided the other is fixed. The resulting sequences can then be tested experimentally in a living cell, provided promoter sequences. RiboMaker could be expanded to account for the computational design of further riboregulatory mechanisms (Rodrigo *et al*., 2013b).

*To whom correspondence should be addressed.

3 METHOD

3.1 Model of bacterial riboregulation

Our model accounts for several energetic and structural terms. The objective function (F_{obj}), to be minimized, reads

$$F_{obj} = \Delta G_{5'UTR:sRNA} + \Delta G^{seed}_{5'UTR:sRNA} - \Delta G_{5'UTR:5'UTR} - \Delta G^{seed}_{5'UTR:5'UTR} - \Delta G_{sRNA:sRNA} - \Delta G^{seed}_{sRNA:sRNA} + \Delta G^{structure}_{5'UTR} + \Delta G^{structure}_{sRNA} + \Delta G^{structure}_{5'UTR:sRNA} + G_{sRNA}. \quad (1)$$

$\Delta G_{5'UTR:sRNA}$ is the free energy release after complete hybridization between the 5′-UTR and the sRNA. $\Delta G_{5'UTR:5'UTR}$ and $\Delta G_{sRNA:sRNA}$ are the same but for homodimer formation. $\Delta G^{seed}_{5'UTR:sRNA}$ is the free energy release after seed pairing, here approached by the length of the seed region (Rodrigo *et al.*, 2012, 2013a). $\Delta G^{seed}_{5'UTR:5'UTR}$ and $\Delta G^{seed}_{sRNA:sRNA}$ are the same but, as before, for homodimer formation. $\Delta G^{structure}_{5'UTR}$, $\Delta G^{structure}_{sRNA}$ and $\Delta G^{structure}_{5'UTR:sRNA}$ are the works required to fold the 5′-UTR, the sRNA and the complex 5′-UTR:sRNA according to the structure specifications. For that, we calculate the Hamming distance between the actual and target structures, rescaling it in terms of free energy with an empirical parameter (Rodrigo *et al.*, 2013a). Finally, G_{sRNA} is the free energy of the sRNA.

In Equation (1), positive terms correspond to objectives for a positive design strategy, whereas negatives ones correspond to objectives for a negative design (Dirks *et al.*, 2004). We give the same importance to accessibility as to hybridization (Busch *et al.*, 2008), and, for simplicity, we do not consider weighting parameters to construct F_{obj}. Accordingly, we will maximize the interaction between the 5′-UTR and the sRNA, while minimizing the two interactions to form homodimers (Rodrigo *et al.*, 2012). This is important to shift the equilibrium towards the appropriate complex. In addition, we will minimize the structural distance between the actual and target systems (for the 5′-UTR and complex 5′-UTR:sRNA) to reach the intended regulatory behaviour.

3.2 Sequence and structure specifications

RiboMaker starts from random sequences or can read initial ones in case the user wants to feed it. As sequence specifications, the user only needs to detail the Shine-Dalgarno box and the start codon for the 5′-UTR, which will be kept fixed during the optimization. For the sRNA, the user can specify the sequence of the transcription terminator, although it is not mandatory and the sRNA can be designed without. In addition, as structure specifications, RiboMaker only requires the specification of the structure of the Shine-Dalgarno box (and neighbouring nucleotides) in both intra- and intermolecular folding states. To compute $\Delta G^{structure}_{5'UTR}$ (intramolecular specification), the partition function is used to avoid high ensemble defects (Zadeh *et al.*, 2011). RiboMaker does not require the specification of the full intramolecular structures of all single species.

3.3 Optimization scheme

We used Monte Carlo Simulated Annealing (MCSA) as an optimization scheme (Kirkpatrick *et al.*, 1983). Initial sequences are iteratively mutated, subjected to sequence constraints, towards a solution that satisfies the specified behaviour. Single-point mutations in one strand are applied every iteration. After that, F_{obj} is evaluated to decide the acceptance or rejection of that mutation, following a Metropolis criterion. The MCSA temperature is continuously adjusted during the process following an exponential cooling scheme. Accordingly, the initial temperature is usually chosen to be high, hence moving during evolution from random to adaptive walk. To avoid undesired sequences, a quality check operator is applied, for instance, to avoid repeats.

3.4 Fitness landscape and convergence

The vast search space for RNA design can be explored by means of combinatorial optimization methods (Dirks *et al.*, 2004). For the problem of riboregulation, Figure 2 shows the energy landscape, together with an optimized trajectory, in terms of three significant components of F_{obj}, which account for interaction and structural specifications of regulation. There, each point represents a different sequence, and different solutions can be reached from the same specifications.

The method can reach a good solution in few minutes (in a standard computer with a 2 GHz processor). Because all structural specifications, intra- and intermolecular, are introduced into the objection function (following the penalty method), all sequences generated by mutations are evaluated, which allows preventing traps and reaching many different adaptive paths (10^3 iterations can be enough to obtain good designs for the riboregulatory models shown in Figure 1).

3.5 Designability and limitations

Of relevance, F_{obj} has been shown to correlate with riboregulatory activity in natural and synthetic systems (Mutalik *et al.*, 2012; Rodrigo *et al.*, 2013a). To increase riboregulatory activity, the user can also specify certain recognition element of the RNA chaperon Hfq as a part of the

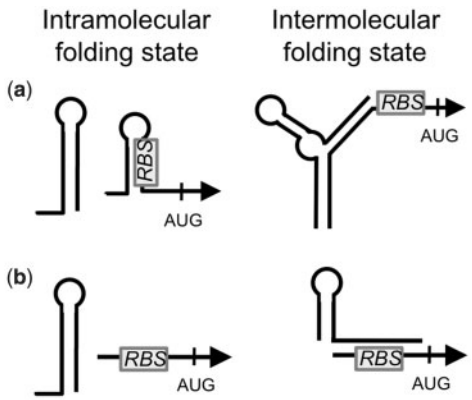

Fig. 1. Scheme of the two riboregulatory models considered. (**a**) Positive regulation, the sRNA activates gene expression, and (**b**) negative, the sRNA represses gene expression. The degree of exposition or blockage of the RBS serves as a control variable

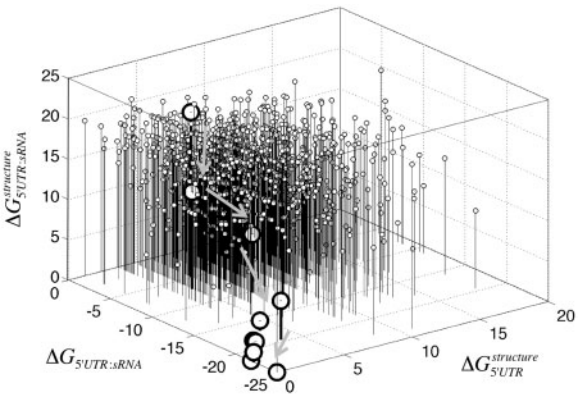

Fig. 2. Illustration of the energy landscape in terms of three components of the objective function (>1000 points representing random sequences). Big circles represent different states during an optimization run, showing the convergence of the algorithm (arrows)

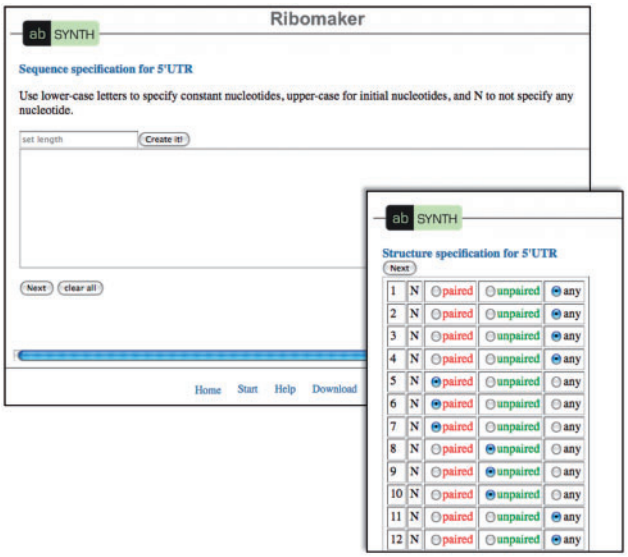

Fig. 3. Snapshots of the web server, where the user specifies the sequence and structure

designed sRNA (Na *et al*., 2013). It is expected that by starting from random sequences, the designed ones be sufficiently dissimilar to others to make cross-talk unlikely (Rodrigo *et al*., 2012), and, in turn, to be able to regulate chromosomal genes too (Na *et al*., 2013). In addition, orthogonal systems could be obtained afterwards by introducing few key mutations (Mutalik *et al*., 2012).

The main limitation of our approach, which is otherwise general for RNA design, is the use of secondary structure to model RNA, which prevents obtaining designs with pseudoknot interactions or even non-canonical base pairing (Bida and Das, 2012). This is not critical, as it only has the effect of restricting the space of available sequences, already too large to explore it exhaustively. Moreover, the control over conformations, although simple, is useful to obtain functional designs in bacteria that regulate ribosome binding (Fig. 1), but it would require new regulatory models to get designs for mammals or plants.

4 IMPLEMENTATION

4.1 Source distribution

The program is implemented in C++, and it has been compiled and executed under Linux and Mac OS X environments. The Vienna RNA package v1.8 (Hofacker *et al*., 1994) is used as a library to calculate RNA secondary structures and free energies. The program reads the design specifications from a text file and writes the designed sequences and scoring terms into another text file. The program can run distributed in high-performance computing clusters.

4.2 Web server

The site was developed in HyperText Markup Language and PHP HyperText Preprocessor. For creating the web structure (header, body and footer), we used Cascading Style Sheets. To initialize the values and check that all fields are filled, we created a JavaScript function. The user needs to fill the fields for sequence and structure specifications. Session variables are employed to hold all information (sequences and structures for all species) during a single-user session. Then, our C++ program is executed for optimization of sequences. The designed sequences, together with the corresponding structures and free energies, are shown on the screen (Fig. 3).

ACKNOWLEDGEMENT

We thank Ricardo Marco for the development of the website and Joan Hérisson for his support with the abSYNTH web platform.

Funding: This work was supported by the grants FP7-KBBE-613745 (PROMYS) and FP7-ICT-610730 (EVOPROG) to A.J., and G.R. acknowledges the EMBO long-term fellowship co-funded by Marie Curie actions (ALTF-1177-2011).

Conflict of Interest: none declared.

REFERENCES

Bida,J.P. and Das,R. (2012) Squaring theory with practice in RNA design. *Curr. Opin. Struct. Biol.*, **22**, 457–466.

Busch,A. *et al.* (2008) IntaRNA: efficient prediction of bacterial sRNA targets incorporating target site accessibility and seed regions. *Bioinformatics*, **24**, 2849–2856.

Dirks,R.M. *et al.* (2004) Paradigms for computational nucleic acid design. *Nucleic Acids Res.*, **32**, 1392–1403.

Fontana,W. and Schuster,P. (1998) Shaping space: the possible and the attainable in RNA genotype-phenotype mapping. *J. Theor. Biol.*, **194**, 491–515.

Hofacker,I.L. *et al.* (1994) Fast folding and comparison of RNA secondary structures. *Monatsch. Chem.*, **125**, 167–188.

Isaacs,F.J. *et al.* (2006) RNA synthetic biology. *Nat. Biotechnol.*, **24**, 545–554.

Kirkpatrick,S. *et al.* (1983) Optimization by simulated annealing. *Science*, **220**, 671–680.

Mathews,D.H. *et al.* (1999) Expanded sequence dependence of thermodynamic parameters provides improved prediction of RNA secondary structure. *J. Mol. Biol.*, **288**, 911–940.

Mückstein,U. *et al.* (2006) Thermodynamics of RNA-RNA binding. *Bioinformatics*, **22**, 1177–1182.

Mutalik,V.K. *et al.* (2012) Rationally designed families of orthogonal RNA regulators of translation. *Nat. Chem. Biol.*, **8**, 447–454.

Na,D. *et al.* (2013) Metabolic engineering of Escherichia coli using synthetic small regulatory RNAs. *Nat. Biotechnol.*, **31**, 170–174.

Prasanth,K.V. and Spector,D.L. (2007) Eukaryotic regulatory RNAs: an answer to the 'genome complexity' conundrum. *Genes. Dev.*, **21**, 11–42.

Rodrigo,G. *et al.* (2012) De novo automated design of small RNA circuits for engineering synthetic riboregulation in living cells. *Proc. Natl Acad. Sci. USA*, **109**, 15271–15276.

Rodrigo,G. *et al.* (2013a) Full design automation of multi-state RNA devices to program gene expression using energy-based optimization. *PLoS Comput. Biol.*, **9**, e1003172.

Rodrigo,G. *et al.* (2013b) A new frontier in synthetic biology: automated design of small RNA devices in bacteria. *Trends Genet.*, **29**, 529–536.

Salis,H.M. *et al.* (2009) Automated design of synthetic ribosome binding sites to control protein expression. *Nat. Biotechnol.*, **27**, 946–950.

Waters,L.S. and Storz,G. (2009) Regulatory RNAs in bacteria. *Cell*, **136**, 615–628.

Yin,P. *et al.* (2008) Programming biomolecular self-assembly pathways. *Nature*, **451**, 318–322.

Zadeh,J.N. *et al.* (2011) Nucleic acid sequence design via efficient ensemble defect optimization. *J. Comput. Chem.*, **32**, 439–452.