Journal of Computer-Aided Molecular Design, 11 (1997) 29–38
ESCOM

J-CAMD 380

Optimizing doped libraries by using genetic algorithms

Dirk Tomandl, Andreas Schober and Andreas Schwienhorst*

Department of Molecular Evolution Biology, Institute for Molecular Biotechnology, Beutenbergstrasse 11, D-07745 Jena, Germany

Received 8 May 1996
Accepted 20 September 1996

Keywords: Doped libraries; Molecular evolution; Protein design; Codon usage; Genetic algorithm; Local optimization

Summary

The insertion of random sequences into protein-encoding genes in combination with biological selection techniques has become a valuable tool in the design of molecules that have useful and possibly novel properties. By employing highly effective screening protocols, a functional and unique structure that had not been anticipated can be distinguished among a huge collection of inactive molecules that together represent all possible amino acid combinations. This technique is severely limited by its restriction to a library of manageable size. One approach for limiting the size of a mutant library relies on 'doping schemes', where subsets of amino acids are generated that reveal only certain combinations of amino acids in a protein sequence. Three mononucleotide mixtures for each codon concerned must be designed, such that the resulting codons that are assembled during chemical gene synthesis represent the desired amino acid mixture on the level of the translated protein. In this paper we present a doping algorithm that 'reverse translates' a desired mixture of certain amino acids into three mixtures of mononucleotides. The algorithm is designed to optimally bias these mixtures towards the codons of choice. This approach combines a genetic algorithm with local optimization strategies based on the downhill simplex method. Disparate relative representations of all amino acids (and stop codons) within a target set can be generated. Optional weighing factors are employed to emphasize the frequencies of certain amino acids and their codon usage, and to compensate for reaction rates of different mononucleotide building blocks (synthons) during chemical DNA synthesis. The effect of statistical errors that accompany an experimental realization of calculated nucleotide mixtures on the generated mixtures of amino acids is simulated. These simulations show that the robustness of different optima with respect to small deviations from calculated values depends on their concomitant fitness. Furthermore, the calculations probe the fitness landscape locally and allow a preliminary assessment of its structure.

Introduction

The revolution of molecular biology has led to an intensive study of proteins as well as their technological exploitation. We are no longer restricted to those proteins produced by living organisms. There are various approaches to modulate natural proteins and their functions, or even to design entirely new ones. Among the most recent developments is the so-called irrational design; in its extreme this method requires no a priori information about the protein structure. It relies solely on the desired function of a protein and a way of screening molecule populations for this function. Starting with a combinatorial library of peptides or proteins, the desired molecules are enriched by subsequent rounds of selection and (error-prone) amplification.

However, combinatorial mutagenesis may easily produce a molecular diversity that greatly exceeds the number of different proteins that can be produced in a single experiment. As the number n of randomized positions grows, the number of possible combinations increases by 20^n. Although there are now selection systems, such as the phage display systems [1,2], that allow the screening of extremely large libraries, these methods are still limited to about 10^{11}, or fewer, molecules. This corresponds to the complete coverage of a library with less than nine randomized positions. In a standard type of 'random' library, functional molecules are usually highly diluted in a large

*To whom correspondence should be addressed at: MPI for Biophysical Chemistry, Dept. 081, Am Fassberg 11, D-37077 Göttingen, Germany.

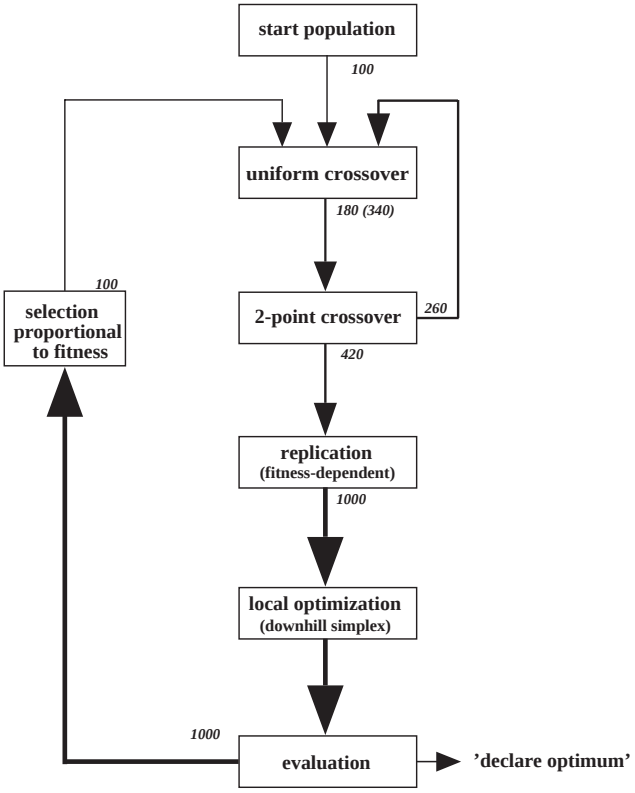

Fig. 1. Schematic representation of the algorithm.

background of nonfunctional molecules. One way to increase the fraction of functional molecules is to employ physical parameters, or to make use of phylogenetic information by comparing with either natural or artificially selected sequences in order to restrict the randomization to doping; that is, the (site-specific) variation is strongly biased to only a certain subset of amino acids per position. Within the scope of such doping strategies, it is important to avoid stop codons [3,4]. Alternatively, it is possible to preferentially generate molecules with only a limited number of mutations as compared to a known, functional wild-type molecule. Here, very small amounts of (all) non-wild-type amino acid codons are 'spiked-in' at certain positions [5–7].

We present an algorithm which converts target sets of amino acids into mixtures of nucleotide synthons. This method is applicable to either type of doping. The optimization procedure (GALO) employs a hybrid of Genetic Algorithms [8,9] and Local Optimizations based on the downhill simplex method [10] which does not depend on gradient information of the underlying fitness landscape. We assume that synthon mixtures can be prepared with fractions that are multiples of 1‰ of the total number of synthon molecules contained in the mixture, i.e. a fractional resolution of 1‰. Therefore, our algorithm provides solutions at a fractional resolution of 1‰ in nucleotide probabilities. This is an improvement over already existing methods [11–13]. The algorithm can generate very different relative frequencies of occurrence of all amino acids (and stop codons) within a target set. Optional weighting factors can bias solutions to a certain desired codon usage. Furthermore, optional correction factors can be included to compensate for the (possible) differences in the chemical coupling efficiencies of the nucleotide synthons. Simulating statistical errors arising from processes such as synthon mixing during an experimental realization of calculated solutions, probes the landscape of suboptimal solutions surrounding the optima. This information sheds some light on the local structure of the fitness landscape [14] of the optimization problem.

Results

The algorithm (GALO)

For each of the three codon positions, four fractions of nucleotides (A,T,C,G) have to be optimized. This results in a set of $3 \times 4 = 12$ variables. Since for each codon position the four nucleotide fractions add up to 1, the values for three nucleotide fractions, e.g. the A, T and C fractions, are sufficient to determine the fourth fraction (G). Therefore, only $3 \times 3 = 9$ variables have to be optimized. The fact that all amino acids may be represented by codons containing only G and C at the third position reduces the number of variables to $3 + 3 + 1 = 7$. Nevertheless, there might be doping situations that reveal significantly better solutions with the mixture of all four nucleotides at the third position. Due to the properties of the genetic code, certain amino acid mixtures cannot be generated by doped synthesis in a single synthesis column. Therefore, we also consider a mixed resin approach [15] analogous to that used in peptide synthesis [16–18]. Here, the resin for the solid-phase DNA synthesis, carrying the nascent population of molecules, is split into p portions. Each portion consists of a fraction of M_p of the total amount of resin. The fractions do not have to be identical for all portions. The split portions of resin filled into p synthesis columns are treated differently in subsequent elongation steps, thereby adding the next (doped) codon. Afterwards the split resin is recombined in one column. $P_{p,\mu}$ represents the fraction of the μth amino acid in the pth synthesis column. $\sum_{p=1}^{n} M_p P_{p,\mu}$ is the mean actual fractional representation of the μth amino acid over all the p different synthesis columns. Every possible solution is encoded with a string of 10p–1 or 8p–1 real numbers (for mixtures of A,T,C,G or G,C, respectively, at the third codon position), representing all the variables mentioned above. Four additional variables are included that determine the mutation rate in amplification steps which is also subjected to the selection process.

The fitness function for evaluating the nucleotide concentrations per codon position with respect to the fractions of encoded amino acids employs the criterion of a sum of the squares of differences:

$$F(\text{string}) = -\sum_{\mu=1}^{21} W_\mu \left[\left(\sum_{p=1}^{n} M_p P_{p,\mu} \right) - S_{p,\mu} \right]^2 \times \left(1 + \frac{0.1}{n} \sum_{p=1}^{n} \text{penalty}(M_p) \right) \quad (1)$$

S_μ is the desired fractional representation of the μth amino acid in the target set and W_μ is the corresponding weighting factor with a default value of 1 for each amino acid as well as the termination codons. The latter allows the optimization of fractions of certain amino acids (e.g. stop signals) that are superior to others. The 'penalty' term is a means of keeping the number of synthesis columns as small as possible because the split synthesis technique becomes experimentally less feasible as more synthesis columns are used in parallel. In the current version of the algorithm, the penalty term is defined in the range from 0 to 1. Thus, the maximal loss of fitness due to the penalty term is set at 10%.

The general outline of the algorithm GALO (Fig. 1) is as follows:

(1) Start with a (small) population of random strings (100).

(2) Choose two members of the population as parents and perform a uniform crossover. Add the product of crossover to the population. Repeat this step 80 times.

(3) Choose two members of the population as parents and perform a two-point crossover. Add the product of crossover to the population. Repeat this step 80 times.

(4) Repeat steps 2 and 3 once.

(5) Replicate the strings in the population according to their individual fitness values until a given maximal population size (1000) is reached. Let the replication be error-prone.

(6) Use the downhill simplex method to optimize the fittest string of the generation and replace the least fittest string.

(7) If the criterion for the goal of optimization is not fulfilled, proceed to step 8; otherwise 'declare optimum'.

(8) Keep the best strings of a generation and reject the others on the basis of the fitness function until the (small) initial population size is reached.

In genetic algorithms (GAs) it is essential to maintain diversity. Therefore, we specify that 10% of the initial strings be 'immortal'. To achieve a one-to-one correspondence between string parameters and decision variables, we choose real-coded GAs; these are comparatively small and easy to handle. Since GAs in a large number of cases converge best with strings that use alphabets of low cardinality ('building block hypothesis' [9,19]), e.g. binary strings, the use of an indefinite alphabet, i.e. real numbers, could lead to suboptimal performance of the algorithm. However, it has been reported that alphabets of low cardinality can produce certain mutational artifacts ('hamming cliff problem' [19]). We try to avoid both problems posed by extreme cardinality by using only three digits and applying local optimization. The ultimate goal of optimization may be defined in different ways

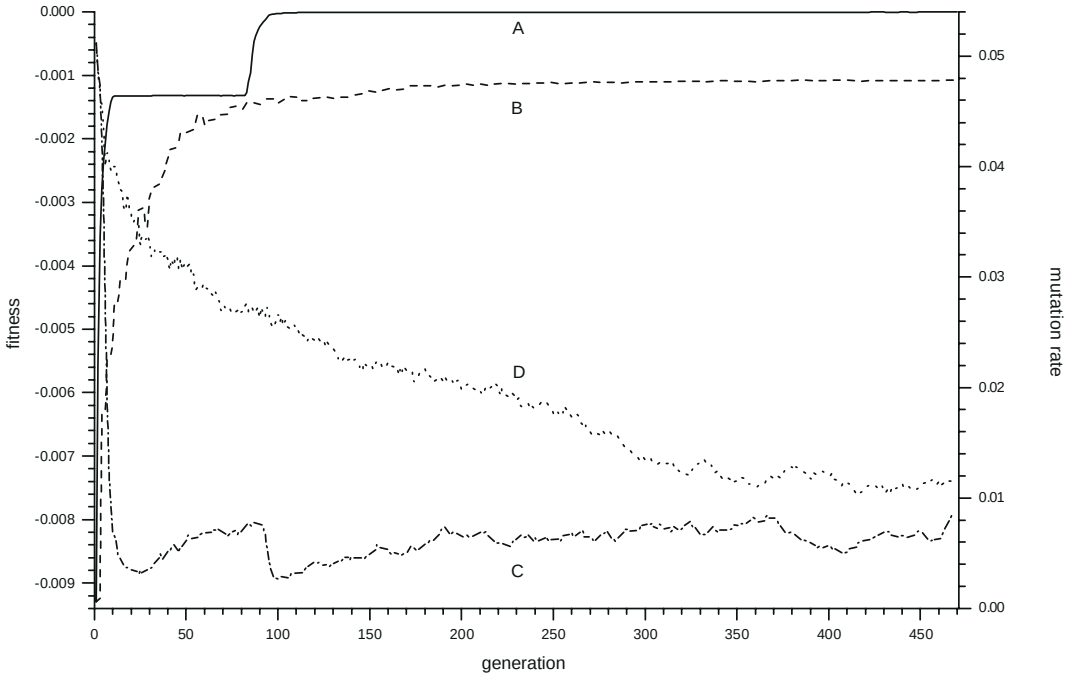

Fig. 2. Performance of the algorithm (problem of equimolar mixtures of all 20 amino acids). Single, typical runs using the GA with (A) and without (B) local optimization, respectively, are compared in terms of the fitness of the best individual and the average mutation rate of the population. The latter decreases steadily with time for the pure GA (D). For the GA with local optimization (C), however, an increase in mutation rate preceding major gains in fitness can be traced.

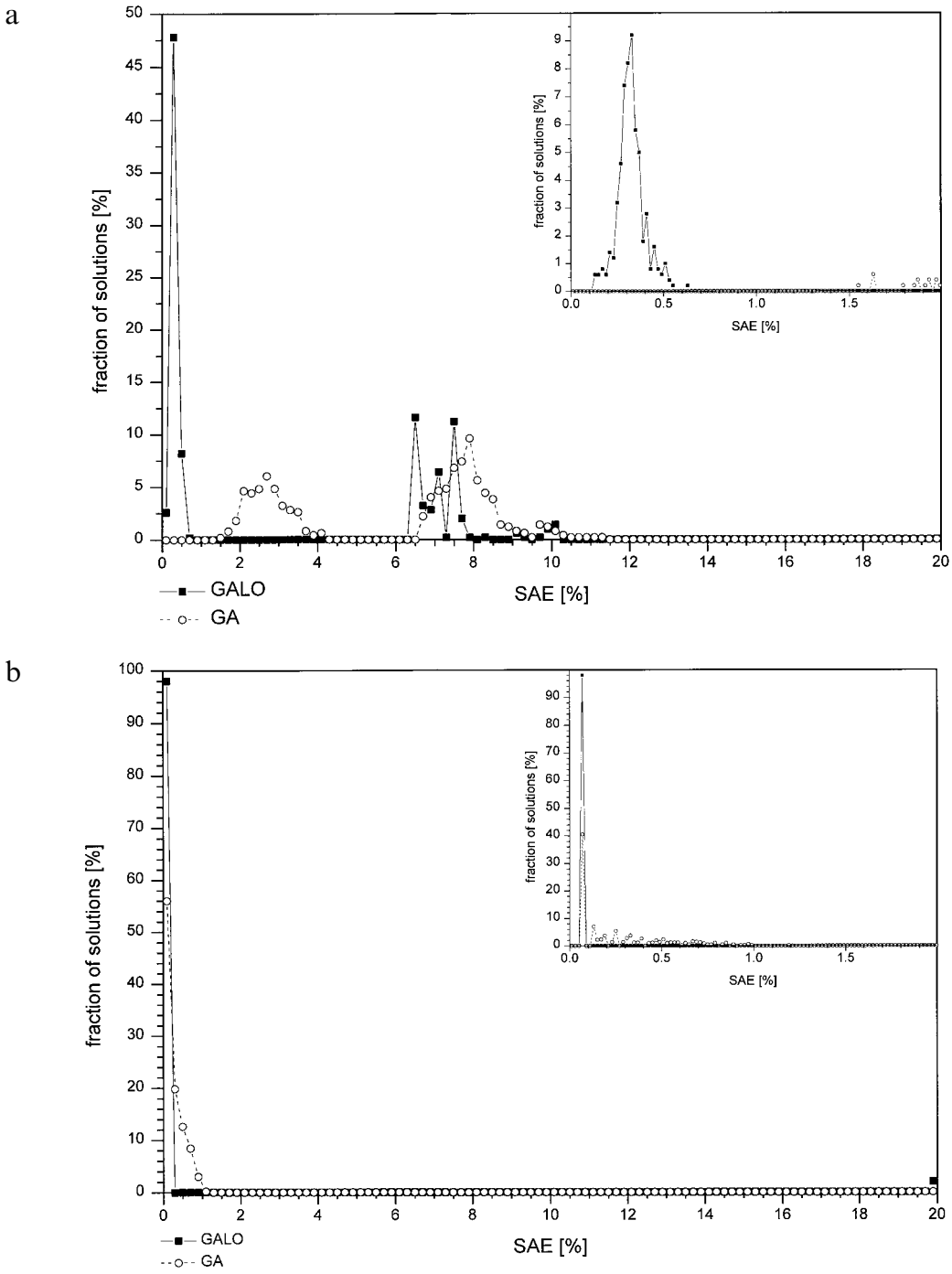

Fig. 3. Reproducibility of optima. For the problem of (a) equimolar mixtures of all 20 amino acids and (b) a mixture of 50% glutamine, 40% leucine and 10% arginine [22,23], 1000 runs were performed respectively. Solutions are concentrated in a few clusters of similar fitness or SAE. In general, the probability to generate a solution in the cluster of best solutions is much lower for the pure GA as compared to the GA with local optimization.

because the landscape is generally not known. We define an 'optimum' solution when the fittest string remains the same for at least 10 iterations. Alternative criteria for an optimum solution are the absolute number of generations or the requirement that a certain fraction of the population must contain the best equal string.

The experimental realization of the calculated doping is subject to statistical errors, e.g. pipetting errors. Therefore, it is useful to characterize how stable an optimized dope is to small deviations from the calculated concentrations of the four nucleotide synthons at all three codon positions. It was assumed that the experimentally realized synthon concentrations form a Gaussian distribution around the theoretically calculated concentrations. The

standard deviation, defining the Gaussian distribution as a measure of the significance of statistical errors, was varied stepwise from 0.5% to 10% for each of the theoretical concentrations. After renormalization of the relative synthon concentrations, the sum of absolute errors (SAE) of the single amino acid fractions was calculated. For each standard deviation, the SAE was averaged over 100 runs. If better solutions were detected accidentally, they were used to replace the original ones. Therefore, the procedure of error simulation can also be regarded as a control in order to evaluate the quality of the (local) optimization.

Finally, we have included optional correction factors to compensate for (possible) differences in the coupling efficiencies of the synthons used in DNA synthesis. Weighting factors are also included to account for a desired codon usage; this can be crucial, for example, for recombinant protein expression [20]. As a default all these factors are set to 1, assuming that no correction is needed.

Numerical results

All calculations were done on an SGI Indy R 4400 (150 MHz). Generating an equimolar mixture of all 20 amino acids and, at the same time, minimizing the frequency of stop codons on the level of DNA [21] was chosen as the first test case. Optimization was usually terminated after 2 s for the one-column, and after 1 min for the three-column 'portioning-and-mixing' GA with local optimization (GALO1 and GALO3). Figure 2 compares the performance of the GA with and without local optimization. The time courses of two typical experiments are shown. They focus on the fitness of the best individual as well as on the average mutation rate of the entire pool. It is obvious that the downhill simplex method is a major improvement in optimization. The algorithm without local optimization clearly requires more generations to reach the optimum. However, the GA cannot be omitted because local optimization alone would certainly lead to early trapping in local optima. Nevertheless, the absolute fitness of the optima found by the GA with and without local optimization was usually the same; however, there are examples where the GALO clearly excels (see Fig. 3a). As the number of generations increases, the average mutation rate for the pure GA decreases (Fig. 2). The GA with local optimization shows a marked adaptation of the mutation rate; it increases slightly whenever the process of optimization tends to become trapped in a local optimum.

To check the reproducibility of optimal solutions, 1000 runs were calculated for each problem. In general, solutions are distributed throughout very few clusters of comparable fitness. Figure 3 shows the distributions for solutions of different classes of the sums of absolute errors (SAEs). Solutions for two different problems at two different resolutions are displayed. In the case of equimolar mixtures of all 20 amino acids (Fig. 3a), approximately 30% of the resulting end points of optimization were found within the cluster of optimal solutions

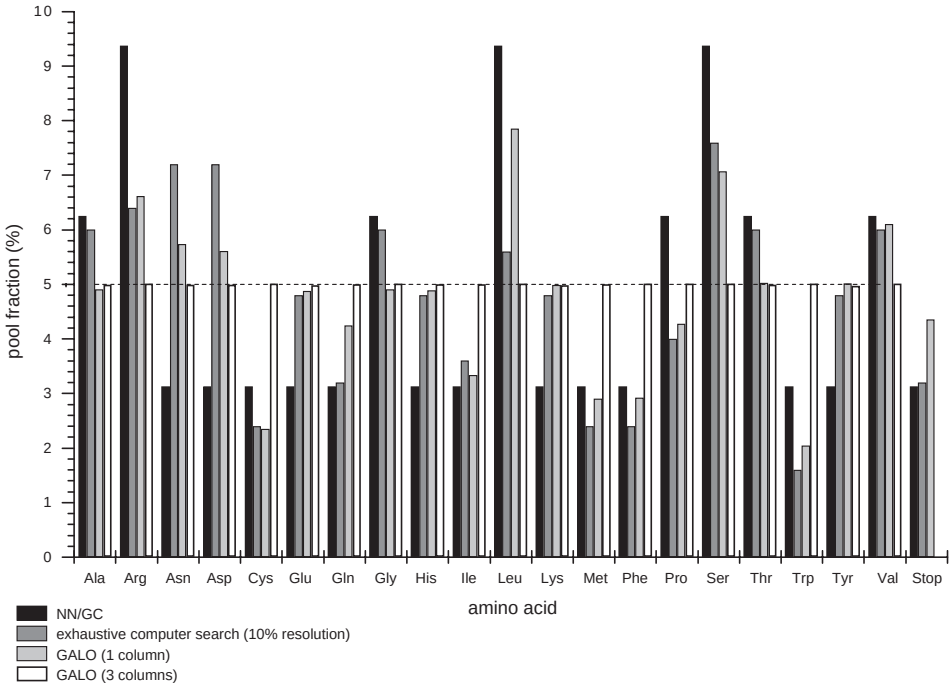

Fig. 4. Comparison of solutions generated by different methods. For the problem of an equimolar representation of all 20 amino acids, solutions were generated by the NN(G/C) approach, an exhaustive computer search of all possible single codon dopes with a fractional resolution of 10% [11,12], a one-column variant and a three-column portioning-and-mixing type of the GALO. The individual SAEs are 45.00%, 32.40%, 26.74% and 0.19%, respectively.

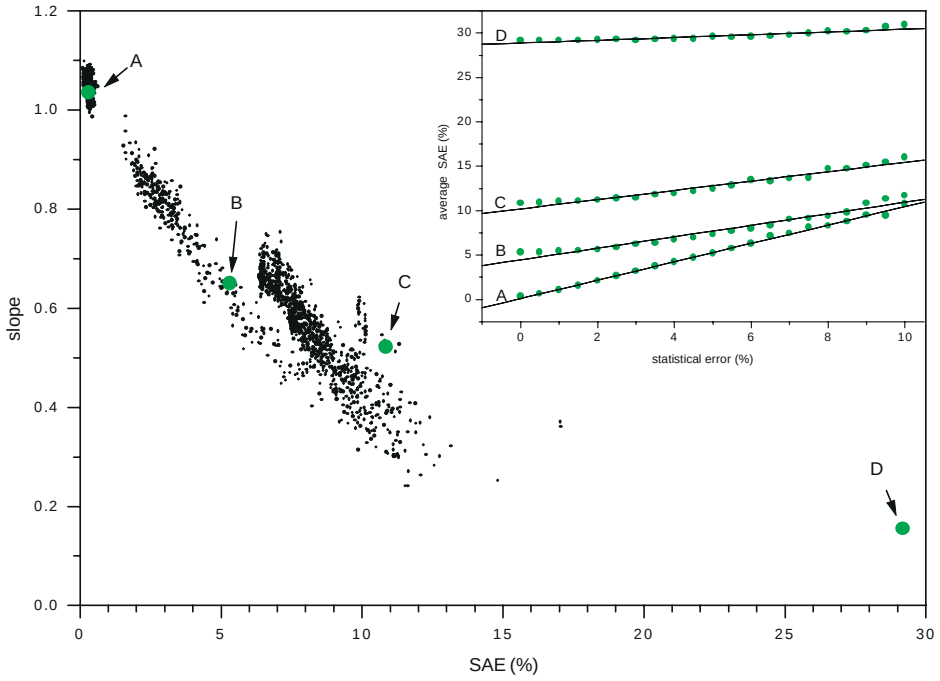

Fig. 5. Stabilities of solutions against experimental errors. Different optima from Fig. 3a were analyzed. It is assumed that experimentally realized synthon concentrations follow a Gaussian distribution around the theoretically obtained optimal values, defined by a certain standard deviation. With the standard deviation growing, the SAE concerning all single amino acid fractions increases linearly. The slope defined in this way can be taken as a measure of the steepness or stability of optima. It is dependent on the absolute fitness values of the solution.

for the GALO. The pure GA will only eventually produce a solution in the same cluster of best fitness values.

For the relatively simple mixture of 50% glutamine, 40% leucine and 10% arginine [22,23] (Fig. 3b), which is among the few doping schemes that can be solved analytically, practically all solutions cluster narrowly around the global optimum, provided that local optimization is applied. Without local optimization, about 50% of the solutions are distributed among clusters of lower fitness.

Figure 4 shows a comparison of solutions for the problem of an equimolar mixture of all 20 amino acids generated by different methods. The portioning-and-mixing type of the algorithm presented here is able to generate nearly perfect solutions with only three synthesis columns. It is noteworthy that even the one-column variant of the genetic algorithm surpasses an exhaustive computer search of all possible singly doped codons with a fractional resolution of 10% [11,12] as well as the common NN(G/C) library. Due to the properties of the genetic code, the calculated mixtures are enriched in amino acids such as arginine, leucine and serine that are represented by six codons each. However, this bias is most pronounced in the case of the NN(G/C) library and almost negligible for the portioning-and-mixing type of the algorithm. Only in the latter approach are stop codons almost completely avoided. All other methods produce nearly the same fraction of 3–4% stop codons.

Since the preparation of synthon mixtures is subject to statistical errors, we analyzed their influence on the target composition of amino acids. The problem of equimolar mixtures of all 20 amino acids served as an example. As expected, the average error for the target composition of amino acids increases as the statistical error in synthon mixing increases. The ranking of the methods compared in Fig. 4 remains the same within the statistical errors defined by a Gaussian distribution with a standard deviation of up to 10% of the theoretically obtained solutions for the optimized single synthon fractions. The method proposed in this paper yields better results even in the case of relatively high statistical errors (data not shown). In Fig. 5 (inset) the stability of four arbitrarily chosen solutions generated by the three-column 'portioning-and-mixing' GALO is compared in regard to the stability against Gaussian distributed statistical errors. When the standard deviation which defines the Gaussian distribution increases, the SAE of the single amino acid fractions becomes larger as well. The better solutions have a steeper optimum, i.e. they are more susceptible to small statistical errors during chemical DNA synthesis (Fig. 5).

The design of a combinatorial cassette for the active site of thioredoxin serves as a practical example for demonstrating the utility of the GA-based method proposed in this paper. Thioredoxin acts as a hydrogen donor for reductive enzymes such as ribonucleotide reductase [24, 25]. It functions as a general reductant for disulfides in proteins, including insulin and fibrinogen [26], and serves as a regulatory factor for enzymes or receptors in photosynthetic systems [27]. *E. coli* thioredoxin (108 amino

TABLE 1
CALCULATED DOPES FOR AN ACTIVE SITE POSITION OF THIOREDOXIN USUALLY OCCUPIED BY LYSINE (K)

	Desired	One column	Two columns
Ala	0.00	0.34	0.00
Arg	22.20	23.02	22.21
Asn	5.60	7.08	5.61
Asp	5.60	0.35	5.61
Cys	0.00	0.13	0.00
Glu	0.00	2.27	0.00
Gln	0.00	3.32	0.00
Gly	0.00	1.23	0.00
His	11.10	0.52	11.10
Ile	0.00	0.93	0.00
Leu	0.00	0.26	0.00
Lys	44.30	45.36	44.27
Met	0.00	1.48	0.00
Phe	0.00	0.01	0.00
Pro	0.00	0.49	0.00
Ser	0.00	3.57	0.00
Thr	5.60	6.72	5.62
Trp	0.00	0.58	0.00
Tyr	5.60	0.27	5.58
Val	0.00	0.12	0.00
End	0.00	1.98	0.00
Sae		42.31	0.10

Phylogenetic data from 18 sequences of the thioredoxin family were used to determine the target dope at this position.

acids) also has several non-redox related functions [28–30]. The tertiary structure reveals that the active site sequence (-WCGPCK-) forms a tight, disulfide-constrained loop on the protein's surface [31]; both cysteines are required for the redox activity. The selection of active thioredoxin variants can easily be performed in bacteria using selective media [32]. Phylogenetic data from 18 sequences [33–39] of the thioredoxin family reveal other functional residues at the six active site positions. For example, lysine (K) at the last position of the active site sequence appears only 8 times (or 44.4%) among this phylogeny of 18 sequences. It can be replaced by arginine (four sequences or 22.2%), histidine (two sequences or 11.1%) or asparagine, aspartic acid, threonine and tyrosine (one sequence or 5.6% each). Hence, doping based on this phylogeny would ideally produce a mixture of lysine, arginine, histidine, asparagine, aspartic acid, threonine and tyrosine; the fractions at this position correspond to those given in the parentheses above. Doping algorithms 'reverse translate' this desired mixture by designing three nucleotide mixtures of the concomitant (doped) codon (Table 1). The ranking of the different methods for calculating the optimal doping is identical to that for an equimolar mixture of all amino acids. To obtain an almost perfect solution, a GALO with two columns is required.

Discussion

Doping increases the probability of finding 'positive' mutants in a random library by reducing the search space

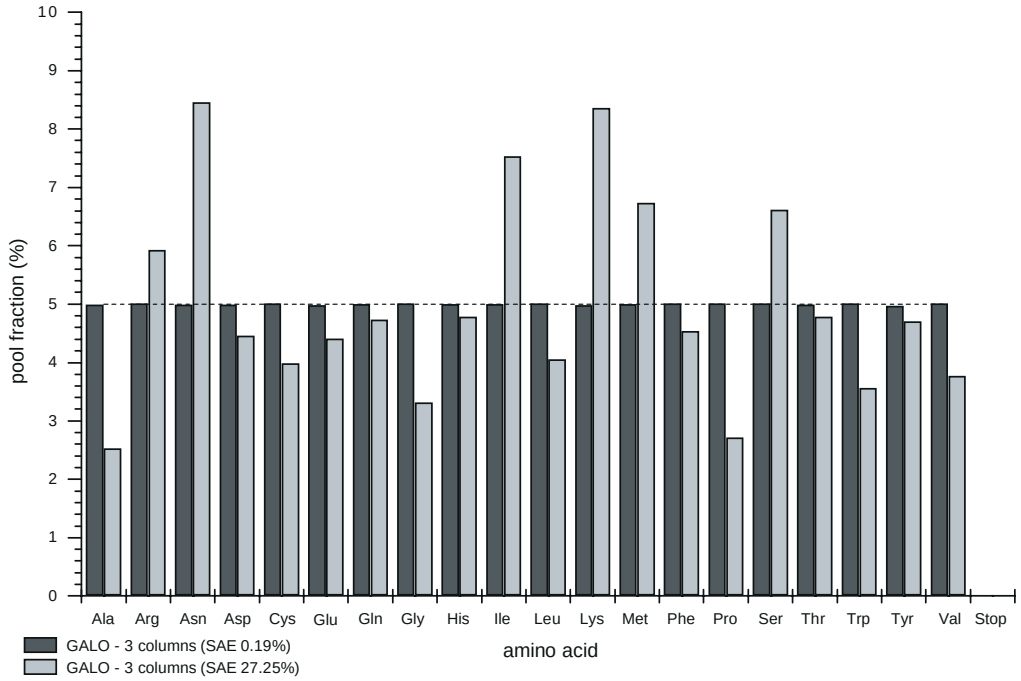

Fig. 6. Effect of unbalanced synthon reaction rates on the target dope. Experimentally generated dopes based on the same three-column portioning-and-mixing solution are significantly different depending on which synthon reactivities are assumed. If the A phosphoramidite is indeed incorporated at higher frequency than C, G or T [47], the algorithm without correction factors, i.e. based on the assumption of equal reaction rates, generates large deviations from the calculated optimum.

to a 'promising' subset of sequences, e.g. the wild-type and its related phylogeny. Alternatively, positional subsets of amino acids can be determined by calculating structurally compatible amino acids within the framework of structure prediction methods (see Ref. 40 for a review). Unfortunately, the synthesis of mixed oligonucleotides using trinucleotide phosphoramidites [41–45] as a possible way of completely avoiding undesired amino acids and stop codons at any given position is not yet available commercially. Thus, it is desirable to have a method for converting target sets of amino acids into nucleotide mixtures that is amenable to standard DNA synthesis based on mononucleotide synthons.

We present an optimization procedure that is a hybrid of a genetic algorithm and a local optimization which is based on the downhill simplex method. This procedure is capable of solving the doping problem for subsets of amino acids at any given relative frequency of occurrence and with a fractional resolution of 1‰. The error function that is used for evaluating the nucleotide concentrations per codon position with respect to the fractions of encoded amino acids uses the criterion of the sum of the squares of the differences. The criterion of group probabilities [11] is disadvantageous at this point because it optimizes only subsets with equally probable amino acids. The resolution of doping greatly exceeds that obtained by other methods [11,12]. Apparently the step size of the other methods is too coarse-grained and this makes it difficult to find a great number of very good local optima. Methods like the exhaustive computer search for all possible single codon dopes can only achieve comparable resolutions at the expense of unacceptably high computer time.

State-of-the-art DNA synthesizers are capable of managing synthon mixtures at very high resolutions if these mixtures are prepared externally. If synthons are mixed directly in the synthesis column prior to the reaction, resolutions of the order of 2.5% can be achieved. Both types of mixing synthons are subject to experimental errors. The optimization procedure presented here consistently evaluates the stability of the optimized doping by checking the values for small deviations from the calculated optimal concentrations of the four nucleotide synthons. As the fitness of the optima increases, the steepness with which the fitness value decreases from its optimum value becomes more pronounced. This means that good solutions are more susceptible to statistical errors in DNA synthesis. Solutions found by this procedure are often very similar, as has been shown for the QLR mixture (i.e. different optima cluster in the solution landscape). There are also examples where optima populate a large part of the total solution landscape; this has been observed for an equimolar mixture of all 20 amino acids. As the resolution of doping increases, these degenerate solutions are found more frequently.

There is conflicting evidence considering the reaction rates of the four standard phosphoramidites used in automated DNA synthesis. Whereas some experiments suggest that freshly prepared, nominally equimolar, pre-made mixtures of the phosphoramidites yield nearly equimolar (±1–5%) product distributions [21,46], other research groups [47,48] report that the A phosphoramidite is incorporated at a somewhat higher frequency than C, G or T. On the other hand, the User Bulletin of Applied Biosystems [49] lists the A phosphoramidite as the least reactive. The same reference also mentions that the G phosphoramidite degrades faster than the others. This indicates that the age of the phosphoramidite solutions may be critical. Figure 6 shows the experimentally generated doping based on a noncompensating three-column solution with the published reaction rates of Ref. 47. It is essential to quantify carefully the effects of different reaction rates on the synthesizer and on the synthesis protocol because they would greatly influence the experimentally generated dope.

If the protein encoding DNA sequences are to be translated in a particular organism, e.g. *E. coli*, the codon usage can be crucial for good expression yields [20,50–54]. A good guess is obtained by using predominantly the most frequent codons for each amino acid. Table 2 compares the codon usage of optimal solutions, derived by different methods for the problem of equal distribution of all 20 amino acids, to the codon usage in *E. coli* (CUTG Database, GenBank, Release 93; also available for on-line

TABLE 2
CODON USAGE USING THE ARGININE CODONS AS A VALUABLE EXAMPLE

Arg	Relative codon usage in *E. coli* (%)	NN/GC (%)	Exhaustive computer search (10% step size) (%)	GALO1 (%)	GALO3 (%)	GALO4 with codon usage (%)
AGA	4.80	0.00	0.00	0.00	0.00	0.00
AGG	3.02	33.33	37.50	34.85	100.00	0.00
CGA	6.39	0.00	0.00	0.00	0.00	0.00
CGC	38.28	33.33	37.50	9.09	0.00	10.00
CGG	9.57	33.33	25.00	30.30	0.00	50.00
CGT	37.94	0.00	0.00	25.76	0.00	40.00

For the problem of an equimolar mixture of all 20 amino acids, the portioning-and-mixing type of the algorithm exclusively generates arginine codons (AGA/AGG) that are slowly translated in *E. coli* when unconstrained by appropriate weighting factors. Applying a weighting factor of zero for each of these two arginine codons reveals solutions with the fraction of AGA and AGG codons minimized.

access to DDBJ: http://www.ddbj.nig.ac.jp/). None of the methods was originally intended to select a certain codon usage; therefore, it is not surprising that all of the above methods show more or less large deviations from the codon usages of frequently used host organisms, e.g. *E. coli*. Libraries based on nonoptimal codon usage would produce a significantly large number of clones showing low (or even no) expression. Therefore, it is desirable to include a variable set of weighting factors in the codon tables used by the GA which stresses the appropriate codon usage. This seems to be particularly important for the portioning-and-mixing type of algorithm that generates only arginine codons (AGA/AGG) when unconstrained by the appropriate weighting factors (Table 2). Codons (AGA/AGG) are known to be slowly translated in *E. coli* [52]. If these two arginine codons are weighted by a factor of zero, solutions of approximately the same overall fitness as for the unconstrained version are found, but at the expense of an additional synthesis column. In contrast to unconstrained algorithms, the fraction of AGA and AGG codons is now basically negligible. In addition to codon usage there are other factors known to affect translation efficiency, e.g. mRNA structure. A challenge for future developments would be to include features of mRNA based, for example, on its calculated RNA secondary structure [55] as an additional boundary condition in the optimization procedure described above.

Conclusions

We have combined a genetic algorithm with a local optimization procedure based on the downhill simplex method which enables us to convert target sets of amino acids into nucleotide mixtures. The optimization procedure has several advantages. (i) The algorithm produces excellent solutions and requires very little computer time. (ii) The majority of solutions with good fitness cluster close to the best local optimum, as judged by comparison with the results of several optimizations. (iii) Unlike other present methods, the algorithm proposed here produces solutions at 1‰ of the fractional resolution in nucleotide probabilities. (iv) The algorithm also includes a way to generate disparate relative frequencies of occurrence of all amino acids (and stop codons) within a target set. (v) The simulation of possible statistical errors, e.g. pipetting errors, serves as a criterion for the robustness of different optima. (vi) Optional correction factors can be included to compensate for the (possible) differences in the chemical coupling efficiencies of the nucleotide synthons and to account for specific codon usage.

Acknowledgements

This research was supported in part by grants BEO 0310665, BEO 0310701 and BEO 0310713 from the German Bundesministerium für Bildung und Forschung (BMBF) and Evotec Biosystems GmbH. The authors would also like to thank F. Wirsching for discussions and M. Clegg for critically reading the manuscript.

References

1 Smith, G., Science, 228 (1985) 1315.
2 Makowski, L., Curr. Opin. Struct. Biol., 4 (1994) 225.
3 Little, J.W., Gene, 88 (1990) 113.
4 Siderovski, D.P. and Mak, T.W., Comput. Biol. Med., 23 (1993) 463.
5 Matteucci, M.D. and Heyneker, H.L., Nucleic Acids Res., 11 (1983) 3113.
6 Derbyshire, K.M., Salvo, J.J. and Grindley, N.D.F., Gene, 46 (1986) 145.
7 Ner, S.S., Atkinson, T.C. and Smith, M., Nucleic Acids Res., 17 (1989) 4015.
8 Holland, J.H., Adaptation in Natural and Artificial Systems, The University of Michigan Press, Ann Arbor, MI, U.S.A., 1975.
9 Goldberg, D.E., Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, MA, U.S.A., 1989.
10 Nelder, J.A. and Mead, R., Comput. J., 7 (1965) 308.
11 Arkin, A.P. and Youvan, D.C., Biotechnology, 10 (1992) 297.
12 Goldman, E.R. and Youvan, D.C., Biotechnology, 10 (1992) 1557.
13 Balint, R.F. and Larrick, J.W., Gene, 137 (1993) 109.
14 Kauffman, S.A., J. Theor. Biol., 157 (1992) 1.
15 Glaser, S., Yelton, D. and Huse, W.D., J. Immunol., 149 (1992) 3903.
16 Lam, K.S., Salmon, S.E., Hersh, E.M., Hruby, V.J., Kazmierski, W.M. and Knapp, R.J., Nature, 354 (1991) 82.
17 Lam, K.S., Hruby, V.J., Lebl, M., Knapp, R.J., Kazmierski, W.M., Hersh, E.M. and Salmon, S.E., Bioorg. Med. Chem. Lett., 3 (1993) 419.
18 Sebestyén, F., Dibó, G., Kovács, A. and Furka, A., Bioorg. Med. Chem. Lett., 3 (1993) 413.
19 Goldberg, D.E., Complex Systems, 5 (1991) 139.
20 Robinson, M., Lilley, R., Little, S., Emtage, J.S., Yarranton, G., Stephens, P., Millican, A., Eaton, M. and Humphreys, G., Nucleic Acids Res., 12 (1984) 6663.
21 Jellis, C.L., Cradick, T.J., Rennert, P., Salinas, P., Boyd, J., Amirault, T. and Gray, G.S., Gene, 137 (1993) 63.
22 Davidson, A.R. and Sauer, R.T., Proc. Natl. Acad. Sci. USA, 91 (1994) 2146.
23 Davidson, A.R., Lumb, K.J. and Sauer, R.T., Nat. Struct. Biol., 2 (1995) 856.
24 Laurent, T.C., Moore, E.C. and Reichard, P., J. Biol. Chem., 239 (1994) 3436.
25 Engström, N.-E., Holmgren, A., Larsson, A. and Söderhäll, S., J. Biol. Chem., 249 (1974) 205.
26 Holmgren, A., Annu. Rev. Biochem., 54 (1985) 237.
27 Buchanan, B.B., Arch. Biochem. Biophys., 288 (1991) 1.
28 Lim, C.-J., Haller, B. and Fuchs, J.A., J. Bacteriol., 161 (1985) 799.
29 Huber, H.E., Russel, M., Model, P. and Richardson, C.C., J. Biol. Chem., 261 (1986) 15006.
30 Kunkel, T.A., Patel, S.S. and Johnson, K.A., Proc. Natl. Acad. Sci. USA, 91 (1994) 6830.
31 Katti, S.K., LeMaster, D.M. and Eklund, H., J. Mol. Biol., 212 (1990) 167.

32 Lim, C.-J., Geraghty, D. and Fuchs, J.A., J. Bacteriol., 163 (1985) 311.
33 Holmgren, A., Eur. J. Biochem., 6 (1968) 475.
34 Qin, J., Clore, G.M. and Gronenborn, A.M., Structure, 2 (1994) 503.
35 Beckman, D.L. and Kranz, R.G., Proc. Natl. Acad. Sci. USA, 90 (1993) 2179.
36 Loferer, H., Bott, M. and Hennecke, H., EMBO J., 12 (1993) 3373.
37 Edman, J.C., Ellis, L., Blacher, R.W., Roth, R.A. and Rutter, W.J., Nature, 317 (1985) 267.
38 Martin, J.L., Structure, 3 (1995) 245.
39 Epp, O., Ladenstein, R. and Wendel, A., Eur. J. Biochem., 133 (1983) 51.
40 Sippl, M.J., Weitckus, S. and Flöckner, H., In Merz Jr., K. and LeGrand, S. (Eds.) The Protein Folding Problem and Tertiary Structure Prediction, Birkhäuser, Boston, MA, U.S.A., 1994, pp. 354–407.
41 Chattopadhyaya, J.B. and Reese, C.B., Nucleic Acids Res., 8 (1980) 2039.
42 Sondek, J. and Shortle, D., Proc. Natl. Acad. Sci. USA, 89 (1992) 3581.
43 Virnekäs, B., Ge, L., Plückthun, A., Schneider, K.C., Wellnhofer, G. and Moroney, S., Nucleic Acids Res., 22 (1994) 5600.
44 Lyttle, M.H., Napolitano, E.W., Calio, B.L. and Kauvar, L.M., Biotechniques, 19 (1995) 274.
45 Ono, A., Matsuda, A., Zhao, J. and Santi, D.V., Nucleic Acids Res., 23 (1995) 4677.
46 Zon, G., Gallo, K.A., Samson, C.J., Shao, K.-I., Summers, M.F. and Byrd, R.A., Nucleic Acids Res., 13 (1985) 8181.
47 Hermes, J.D., Parekh, S.M., Blacklow, S.C., Köster, H. and Knowles, J.R., Gene, 84 (1989) 143.
48 Horwitz, B.H. and DiMaio, D., Methods Enzymol., 185 (1990) 599.
49 Andrus, A.W., User Bulletin 13, Applied Biosystems, Foster City, CA, U.S.A., 1987, pp. 2–4.
50 Spanjaard, R.A. and Van Duin, J., Proc. Natl. Acad. Sci. USA, 85 (1988) 7967.
51 Ernst, J.F. and Kawashima, E., J. Biotechnol., 8 (1988) 1.
52 Bonekamp, F. and Jensen, K.F., Nucleic Acids Res., 16 (1988) 3013.
53 Brinkmann, U., Mattes, R.E. and Buckel, P., Gene, 85 (1989) 109.
54 Schenk, P.M., Baumann, S., Mattes, R. and Steinbiss, H.H., Biotechniques, 19 (1995) 196.
55 Hofacker, I.L., Fontana, W. and Stadler, P.F., Monatsh. Chem., 125 (1994) 167.