The Pennsylvania State University

The Graduate School

College of Engineering

MODELING AND OPTIMIZATION IN DIRECTED EVOLUTION PROTOCOLS AND PROTEIN ENGINEERING

A Thesis in

Chemical Engineering

by

Gregory L. Moore

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 2005

The thesis of Gregory L. Moore was reviewed and approved* by the following:

Costas D. Maranas
Professor of Chemical Engineering
Thesis Advisor
Chair of Committee

Ali Borhan
Professor of Chemical Engineering

Patrick Cirino
Assistant Professor of Chemical Engineering

Stephen J. Benkovic
Evan Pugh Professor and Eberly Chair in Chemistry

Andrew L. Zydney
Interim Department Head
Endowed Bio-Chair and Professor of Chemical Engineering

* Signatures are on file in the Graduate School.

Abstract

The central theme of this thesis aims toward the systematic development of integrated approaches for proactively designing protein libraries with focused diversity for directed evolution and protein engineering experiments. Experimental paradigms for library generation rely on the generation of point mutations, recombination of parental sequence segments, and *ab initio* library design using degenerate oligonucleotides. In response to these library design paradigms, this thesis presents a computational toolbox for quantifying how diversity is generated and allocated in the combinatorial DNA library and what sequence permutations are the most promising in terms of preserving protein structure and activity.

Table of Contents

List of Figures

List of Tables

Acknowledgements

This thesis would not have been possible without guidance from my advisor, Costas Maranas. The assistance of our collaborators, including Marc Ostermeier, Stefan Lutz, Steve Benkovic, Kevin Gutshall, Jean Brenchley, James Watney, and Sharon Hammes-Schiffer, is gratefully acknowledged as well. Finally, the remainder of the C. Maranas group was also instrumental to this work.

Chapter 1: Literature Review and Motivation

Section 1.1: Introduction

Through the processes of natural selection and co-option, nature has crafted an astounding array of proteins with a remarkable repertoire ranging from catalysis, signaling, recognition and regulation to compartmentalization and repair. Despite this plethora of functionalities and exquisite specialization, many biotechnological tasks require proteins to operate under conditions that were not selected for in nature, such as enhanced thermostability, altered substrate specificity, different cofactor (*i.e.*, NADH, ATP, *etc.*) dependence, non-aqueous environments and often combinations of the above. Unlike many of the systems engineered by people, proteins through evolution had to acquire the inherent ability to change and assume over time subtly or even dramatically different roles in living organisms. This amazing plasticity has enabled bioengineers to design or more often redesign proteins more attuned to specific tasks. Protein engineering, however, remains a formidable challenge. Proteins are much larger (*i.e.*, over 50 residues) than non-biological catalysts, and exhibit complex networks of dynamic interaction necessary for function. Given the residue composition of a protein, the task of *de novo* identifying its three-dimensional structure is non-trivial and only limited successes [1] are currently available. On top of this, even complete structure resolution does not mean that function is always truly elucidated. In many cases, functionality and non-functionality are separated by differences of only fractions of Angstroms in the position of certain key atoms, an accuracy threshold well beyond the current modeling state-of-the-art. These daunting challenges have led to protein engineering paradigms that involve the synthesis and subsequent screening of multiple protein candidates (from tens to billions) as a way of hedging against the imprecise knowledge of sequence-structure-function relations.

This juxtaposition of repeated library generation and screening has emerged as the *directed evolution* design paradigm. Directed evolution methods mimic the process of Darwinian evolution and selection to produce proteins or even entire metabolic pathways with improved properties. These methods (see Figure 1.1) typically begin with the infusion of diversity into a small set of parental nucleotide sequences through mutagenesis and/or DNA recombination. The resulting combinatorial DNA library is transformed into an appropriate host (*e.g.*, *E. coli*) and then is subjected to a high-throughput screening or selection procedure. The best variants are isolated for another round of mutagenesis or recombination. The cycles of mutagenesis/recombination, screening and isolation continue until a protein with the desired level of improvement is found.

In the last few years, a wide range of success stories of directed evolution for many different applications has been reported [2-6]. For example, Buchholz and co-workers [7] reengineered retroviruses used in gene therapy to greatly enhance their spreading efficiency through human fibrosarcoma cells. Arnold and co-workers [8] used directed evolution to engineer a novel biosynthetic pathway in *E. coli* for the production of carotenoids, a diverse class of natural pigments that are of interest for pharmaceuticals and food colorants while also playing a role in the prevention of cancer and chronic disease. Wittrup and co-workers [9] generated single-chain antibodies that bind essentially irreversibly (femtomolar binding constant) with potential future implications for improved cancer and viral therapeutics. Schmid and co-workers [10] enhanced the alkaline pH activity of an α-amylase that can be used to improve the starch removal capability of household detergents. Briefly, other successes include many-fold improvements in enzyme activity and thermostability [11, 12], improved enantioselectivity [13-15], enhanced bioremediation [16-18], and even the design of genetic circuits [19] and vaccines [20-22]. It is increasingly becoming apparent, however, that it is vital to be able to assess and then "steer" diversity toward the most promising regions of sequence space [23]. This is because only an infinitesimally small fraction of the diversity afforded by DNA and protein sequences can be examined regardless of the efficiency of the screening procedure. For example, a 500-nucleotide gene implies $4^{500} \approx 10^{301}$ alternatives, but even the most efficient screening methods can query only up to 10^{12} sequences [24-26]. Therefore, it is desirable to know how diversity is generated (see Section 1.2) and allocated (see Section 1.3) in the combinatorial DNA library and what sequence permutations are the most promising in terms of preserving protein structure and activity (see Section 1.4).

This chapter describes different ways of generating library diversity through DNA manipulation, discuss the advantages and disadvantages of various mutagenesis and recombination methods (including recent developments in nonhomologous and synthetic oligonucleotide recombination), highlight the computational challenges and progress at the level of combinatorial library generation, and describe efforts to discern sequence composition versus functionality trends at the protein level.

Section 1.2: Experimental Techniques for DNA Library Generation

Methods for combinatorial library generation in directed evolution can be broadly classified depending on whether they utilize *mutagenesis* or *recombination* (see Figure

1.1) as the primary mechanism for generating diversity. Mutagenesis-based methods are deployed to (i) randomly distribute nucleotide mutations throughout the length of the parental DNA sequence(s) (*random mutagenesis*), (ii) exhaustively generate all possible mutations at a particular sequence locus (*saturation mutagenesis*), or (iii) produce specific nucleotide substitutions at predetermined locations (*site-directed mutagenesis*). Because it is often unclear which residues should be mutated (*i.e.*, counterintuitive mutations distal to the active site frequently enhance activity/stability), the successful use of saturation and site-directed mutagenesis has so far been infrequent. More commonly, random mutagenesis has been used to generate libraries of mutated DNA sequences. It is typically performed by amplifying the initial parental DNA sequence(s) via the error-prone PCR reaction [27-29], which involves spiking the PCR reaction mixture with $MnCl_2$ to increase the mutation rate (other similar methods are described by ref. [30]). Another way to generate randomly distributed mutations is by transforming the parental DNA sequence(s) into one of many commercially available bacterial mutator strains [31]. In all cases, the mutation rate must be carefully tuned to achieve a balance between progressing through sequence space at a “snail’s pace” (low mutation rate) and a widespread loss of function in the library through a build-up of deleterious mutations (high mutation rate). Typically, an average rate of one to two amino acid changes per directed evolution cycle has been found to allow steady experimental progress [32]. Random mutagenesis methods are relatively inexpensive and easy to set up in the laboratory and have produced improved variants with non-obvious mutations absent from any known homologous sequences [14]. However, it is important to remember that only sequence diversity adjacent to the parental sequence(s) is probed (see Figure 1.2).

Functioning distant sequence diversity is unlikely to be encountered given that this requires the sampling of an unbroken chain of continually improving point mutations. Moreover, after a few directed evolution cycles, mutational bias could be a factor in the sequence library. Due to redundancies in the codon representation (*i.e.*, 64 codons for only 20 amino acids), a mutated nucleotide may not necessarily code for a different amino acid (silent mutations). Thus, amino acids with larger codon sets tend to mutate less often.

In addition to the use of point mutations for generating library diversity, DNA recombination is used to construct hybrids containing *crossovers*, defined as the junction points at which the sequence switches from one parent to another (see Figure 1.1). This allows, in principle, the sampling of sequences contained within the convex polytope defined by the vertices representing the parental sequences (see Figure 1.2). The key idea of recombination is to exchange proven diversity present in existing sequences. The use of DNA recombination for directed evolution was pioneered with the development of DNA shuffling [33], which relies on a PCR-like reaction for the reassembly of randomly fragmented parental sequences. Later, family DNA shuffling [34, 35] was demonstrated by recombining large sets of parental sequences simultaneously. A large number of related protocols such as StEP [36], RACHITT [37], and single-stranded shuffling [38] have also been developed. In all of these methods, crossover generation relies on the annealing and extension of complementary single-stranded fragments originating from different parental sequences (*i.e.*, heteroduplex formation), which tends to bias crossover positions toward stretches of near perfect sequence identity. This, in turn, tends to give rise to biased combinatorial DNA libraries or, even worse, libraries with no additional diversity over the parental one.

In general, a severe bias toward the reassembly of parental sequences (*i.e.*, no recombination) is observed when sequences with less than 60% sequence identity are recombined with annealing-based protocols [33, 39]. Given the fact that protein structure is more frequently conserved than DNA homology, annealing-based methods for recombining genes may potentially exclude solutions to protein engineering problems. The need for a recombination protocol capable of freely exchanging genetic diversity without sequence identity limitations motivated the development of the ITCHY (Incremental Truncation for the Creation of Hybrid enzYmes [40]) and SHIPREC (Sequence Homology-Independent Protein RECombination [41]) protocols. These protocols are capable of generating libraries from low sequence identity parents with crossovers evenly distributed along the length of the sequence (see analysis in ref. [42]). However, ITCHY and SHIPREC are limited to constructing single crossover hybrids between only two parental sequences. Recent protocol design efforts have concentrated on overcoming this limitation by generating multiple crossovers per sequence without homology restrictions. The SCRATCHY protocol [43] generates multiple crossovers by applying DNA shuffling to ITCHY libraries, redistributing the prepositioned ITCHY crossovers throughout the newly reassembled sequences. The number of crossovers generated by SCRATCHY can be boosted even further by enriching the library via PCR amplification of crossover-containing sequence sections [44]. The recently developed SISDC (Sequence-Independent Site-Directed Chimeragenesis [45]), GeneReassembly [46], and SCOPE (Structure-based Combinatorial Protein Engineering [47]) protocols are fundamentally different from ITCHY/SCRATCHY and SHIPREC in that the crossover points must be predetermined prior to the recombination step. For these protocols, fragments have been shown to recombine independently without any sequence bias. A key advantage is the flexibility that they afford to predetermine the number and positions of "smart" crossover sites [48] that hopefully preserve functionality throughout the library.

All DNA recombination methods described so far involve the swapping and concurrent reassembly of parental nucleotide *segments* either obtained through DNA fragmentation or synthesis (GeneReassembly, SCOPE). However, using only nucleotide segments for diversity generation causes blocks of closely spaced polymorphisms to be swapped as a group, limiting library diversity [49]. Synthetic oligonucleotide (nucleotide fragments with lengths of about 20-100 bases) recombination methods overcome this restriction by incorporating degenerate oligonucleotides into the reassembly procedure. The term *degenerate* refers to the synthesis of a mixture of oligonucleotides with different nucleotides (*i.e.*, degeneracies) at certain prespecified positions. The oligonucleotides are designed to include coding information for the polymorphisms present in the parental set, while also including "customized" sequence identity enabling annealing-based recombination between the oligonucleotides. So far, degenerate oligonucleotides have been reassembled by PCR-based reactions (synthetic shuffling [50] and Assembly of Designed Oligonucleotides, ADO [51]) as well as a single sequence of annealing, gap-filling, and ligation steps (degenerate homoduplex recombination, DHR [52]). In all these methods, the occurrence of rare mutations can be boosted by increasing the corresponding oligonucleotide population in the mixture. Furthermore, the oligonucleotides can be designed to be consistent with the codon usage of a specific host organism. Synthetic oligonucleotide recombination can yield a very high crossover density (up to 1 crossover per 12.4 bp [52]); however, there is some concern that the high crossover density may disrupt vital interactions throughout the structure. In fact, a lower average library activity has been observed when comparing a synthetic shuffling library with one generated by family DNA shuffling [50]. In general, the use of synthetic oligonucleotides has been more expensive and time-consuming than the recombination of parental DNA sequences.

Table 1.1 summarizes some of the advantages and disadvantages of each of the protocol types discussed. Recent developments in experimental techniques have made it clear that given sufficient resources, a protocol can be set up to create the desired level of diversity. However, what is less clear is what is the optimal level and type of diversity for a given protein engineering task. Although diversity is required to discover new variants, the average activity of the library tends to drop off as diversity increases [49, 50]. Ultimately, screening capacity limits and defines the optimal library diversity that needs to be considered. Recently many exciting advances in high-throughput screening technologies have been made (see excellent reviews in refs. [24-26]). For instance, phage display [53] and ribosome display [54] systems can be used to screen libraries with as many as 10^{12} members. The use of FACS (Fluorescence-Activated Cell Sorting) coupled with the cell-surface display of proteins and customized, FRET-enabled (Fluorescence Resonance Energy Transfer) substrates can be used to sort library members on the basis of k_{cat} or K_m at a rate of 10^9 per hour [55].

Section 1.3: Computational Challenges at the DNA Level

Although the screening step in directed evolution probes for enhanced protein variants, the diversity generation step (*i.e.*, combinatorialization) is performed via DNA manipulation. Without sufficient diversity in the underlying combinatorial DNA library, the encoded diversity within the protein library will be lacking as well, and the often expensive and labor-intensive screening step will underperform. Thus, being able to predict how alternate protocol setups affect the level and type of diversity generated can ultimately determine the success or failure of a directed evolution project. In this section, we describe efforts at developing predictive modeling frameworks for error-prone PCR and DNA shuffling protocols, followed by methods for optimizing combinatorial DNA library generation to target desired regions of sequence space.

Models for error-prone PCR have focused on predicting mutation rate for a given PCR setup (*e.g.*, cycle number, annealing temperature, primer/template concentrations). This requires the consideration of (i) the plateau effect (where replication efficiency diminishes as the cycle number increases), (ii) the propagation of mutations over a number of PCR cycles with nucleotide-dependent frequencies, and (iii) the ability of nucleotides to back mutate to their original identity given that mutation rates are typically high in error-prone PCR. Some success has been achieved in modeling the plateau effect using kinetic parameters [56-60]. Reference [61] tracked mutations from cycle to cycle considering nucleotide-dependent mutation rates while allowing back mutation, but only with constant replication efficiency. Reference [62] tracked mutations in combination with the plateau effect but did not include back mutation. Reference [63] developed a model that utilizes a branching process to track mutations and incorporates empirical information on the plateau effect. While quite a bit of progress has been achieved towards modeling error-prone PCR, a truly predictive model is still lacking.

Moving next to DNA recombination, Sun first considered models for DNA shuffling of parental sequences with single [64] and multiple [65] point mutations. However, these models did not consider sequence information, and their applicability was limited. Work presented in Chapters 2 and 3 examined for the first time how fragmentation length, annealing temperature, sequence identity and number of shuffled parental sequences affect the number, type and distribution of crossovers along the length of full-length reassembled sequences. In the ***e***Shuffle framework, annealing events during reassembly were modeled as a network of reactions, and equilibrium thermodynamics along with complete nucleotide sequence information was employed to quantify their conversions and selectivities. Comparisons of ***e***Shuffle predictions against experimental data revealed good agreement [39], particularly in light of the fact that there were no adjustable parameters. Specifically we found that crossover numbers were boosted by reducing fragmentation length and annealing temperature and that crossovers tend to aggregate in regions of near perfect sequence identity. As presented in Chapter 5, the customization of ***e***Shuffle for the SCRATCHY protocol led to the ***e***SCRATCHY framework [43]. Using ***e***SCRATCHY we found that in SCRATCHY libraries (i) fragmentation length used for reassembly does not influence the number or location of crossovers generated in full-length sequences, (ii) the crossover distribution is shaped by the crossover statistics of the ITCHY library, and (iii) crossovers are spread evenly throughout the crossover region. The need to safeguard against the formation of reassembled sequences with either truncated or duplicated domains motivated us to further extend the ***e***Shuffle framework to consider out-of-sequence annealing events [66]. Instead of "locking" fragments into their alignment positions, the annealing free energy change was used to determine the likelihood of duplex formation, allowing the prediction of the relative frequency that fragments from different sequence regions will anneal during reassembly.

Subsequent work by ref. [67] further advanced the level of detail of DNA shuffling computational models with the development of a simulation-based model using nucleotide annealing kinetics and thermodynamics. This simulation approach has the advantage of tracking and recording the sequences of a computational ensemble of fragments through multiple rounds of shuffling, and tracks the fate of all reassembled fragments whether or not they are of parental length. A three-step reassembly process was used: (i) single-stranded fragments randomly collide; (ii) on collision, a decision is made whether the molecules will hybridize and, if so, in what arrangement; and (iii) duplexes are extended. This process is repeated until the fraction of unhybridized fragments remains unchanged; this constitutes a round of shuffling. Tracking the entire fragment pool allowed for the quantification of the trade-off between reassembly efficiency (*i.e.*, the fraction of fragments that have reached parental length) and crossover frequency while simultaneously following the production of sequences with missing or repetitive regions. This work represented an important step in optimizing the recovery of diverse, full-length reassembled sequences from a DNA shuffling reaction mixture.

In addition to predictive frameworks for quantifying the allocated library diversity for a given protocol setup, a number of approaches have focused on the inverse problem. Specifically, how should we adjust the protocol setup to achieve the desired statistics of parental composition in the combinatorial libraries? Chapter 4 presents work done exploring the possibility of boosting or even specifically redirecting the formation of crossovers in DNA shuffling by exploiting the inherent redundancy in the codon representation (*e.g.*, isoleucine has the following three synonymous codon representations: ATA, ATC and ATT) while complying with host preferences for specific patterns of codon usage [68]. The key motivation here is that it is possible to optimize the underlying parental DNA sequence codon representation for increasing and/or shaping diversity while at the same time preserving the parental amino acid encodings in the generated combinatorial protein libraries. To this end, the framework named ***e***CodonOpt was developed for exploring the limits of performance that can be achieved through codon optimization.

While in ***e***CodonOpt the objective was to find a *single* codon representation for each of the parental protein sequences, ref. [69] designed instead an *ensemble* of nucleotide sequences that best "matches" a given set of amino acid probabilities. These probabilities can be derived from a multiple sequence alignment of protein family members (*e.g.*, Pfam database [70]) or statistical mechanics approaches that identify protein sequences likely to fit a given protein backbone (discussed in the next section). A two-term objective function was used to score the degree of correlation between the desired amino acid probability distribution and the distribution expected from the nucleotide ensemble. This objective accounts for (i) the absolute difference between desired and designed probabilities (based on the χ^2 function) and (ii) a relative entropy term for quantifying the "distance" between the two distributions [69]. The formulation can also be adapted to generate solutions in accordance with a particular host organism's codon preferences. Significant progress towards predicting and subsequently steering the statistics of unselected combinatorial DNA libraries has been achieved in the last few years. Additional improvements will require a more accurate description of hybridization kinetics and rates of polymerase mediated DNA extensions.

Section 1.4: Computational Challenges at the Protein Level

Currently, two different paradigms are being pursued to computationally aid the design and composition of combinatorial protein libraries. The first involves the *a priori* design of a protein or collection of proteins that best fits a given protein fold. In this case, protein(s) are designed "from scratch" with little guidance from protein family sequence data. The second paradigm aims at elucidating what combinations of parental sequence fragments to include or exclude from the recombination mixture to create a combinatorial library that is both diverse and highly active. Here proven diversity encoded in the form of functional parental sequences is used to assess how well hybrid sequences fit the fold of interest.

Ab initio design of a protein or collection of proteins involves finding the amino acid sequence that best fits a given protein fold. The protein fold is represented by the Cartesian coordinates of its backbone atoms, which are usually fixed in space so that the degrees of freedom associated with backbone movement are neglected (some notable exceptions to the "fixed backbone" design paradigm include refs. [71-76]). Candidate protein designs are generated by selecting amino acid side chains (at atomistic detail) along the backbone design scaffold. For simplicity, side chains are usually only permitted to assume a discrete set of statistically preferred conformations called *rotamers* (see ref. [77] for a review of current rotamer libraries). Thus, a protein design consists of both a residue *and* rotamer assignment. To evaluate how well a possible design fits a given fold, rotamer/backbone and rotamer/rotamer interaction energies for all of the rotamers in the chosen library are tabulated. These potential energies can then be approximated using any of many standard force fields (*e.g.*, CHARMM [78], DREIDING [79], AMBER [80], GROMOS [81]). Alternatively, energy/scoring functions that have been customized for protein design [82-84] are used. Protein design potentials (see ref. [85] for a review) typically include van der Waals interactions, hydrogen bonding, electrostatics, solvation, and even entropy-based penalties for flexible side-chains (*e.g.*, arginine).

Even for a small 50-residue protein, an enormous number (*i.e.*, $153^{50} \approx 10^{109}$ assuming a 153-rotamer library [86]) of designs are possible. Both stochastic and deterministic search strategies have been used to tackle the computational challenge of finding the best design within this vast search space. Because activity level is very difficult to assess computationally, an alternative surrogate for hybrid fitness, namely stability, is employed in most studies. The key justification here is that stability is a prerequisite though not necessarily a monotonic descriptor of functionality. Use of this indirect objective further necessitates the need of designing a combinatorial library rather a single design to improve the chances of success. Stochastic strategies search through the space of feasible designs by making a series of random and/or directed moves. Monte Carlo [83, 87, 88], genetic algorithms [89-91], simulated annealing [92, 93], and many other heuristics [94-96] have been used in protein design with various levels of success. Although stochastic techniques can be used for problems of very large complexity with relatively small CPU/memory requirements, they are not guaranteed to converge to the optimal solution and require extensive tuning of parameters controlling the convergence rate [97, 98].

Conversely, deterministic algorithms are guaranteed to converge to the global minimum energy conformation; however, they tend to be long-running and become intractable for large-scale design problems. The most frequently used deterministic technique is dead-end elimination [99], a pruning method in which rotamers and rotamer pairs that cannot be part of the optimal protein design are eliminated over a number of computational cycles. Recent innovations to accelerate rotamer elimination include the use of upper-bounding information [100], conformational splitting [101], the "magic bullet" metric [102], and background optimization [84]. Dead-end elimination has been used to design the full sequence of a 28-residue zinc finger [103]; the cores of T4 lysozyme (26 residues) [104], thioredoxin (32 residues) [105], and the αMβ2 integrin I domain (45 residues) [106]; small molecule receptors based on periplasmic binding proteins [107]; and metal binding proteins [108].

In practice, more important than finding the mathematical solution to the protein design problem is the ability to generate *in silico* an ensemble of computational designs that subsequently will form the basis for constructing the combinatorial protein library. Furthermore, because the most active proteins are often only marginally stable, examining sub-optimal designs can yield greater insight into a fold's plasticity. Sub-optimal designs may be collected by storing intermediate steps of stochastic searches (*e.g.*, Monte Carlo as in ref. [109]); however, the top 10^5 or even 10^6 designs are not sufficient to completely characterize the vast sequence space associated with large proteins. Alternatively, statistical mechanics based methods can be used to construct, equilibrate, and query ensembles of all possible residue/rotamer states (see ref. [110] for a review). Mean-field theory allows the extraction of individual rotamer site probabilities (first-order [111-114]) or rotamer-rotamer joint probabilities (second-order [115]) after the free energy of the ensemble is minimized. The probabilities represent how well a particular rotamer (or rotamer pair) fits at a particular sequence position (or pair of positions). Equivalently, Saven and co-workers have introduced a method for extracting rotamer site probabilities from a maximal-entropy ensemble [116, 117].

The methods described so far allow followed the first paradigm that aims to design proteins and/or libraries "from scratch" that best fit the fold of interest. However, directed evolution experiments have a natural starting point – the original parental sequences. Following the second paradigm, a number of strategies have been developed that utilize the sequence and structure information encoded in the parental sequences to guide the design of combinatorial protein libraries. Typically, this involves the scoring of libraries of hybrid protein sequences against the parental sequences. This idea was first demonstrated with the SCHEMA algorithm [118], which hypothesized structural disruption whenever a contacting residue pair (within 4.5 Å) in a hybrid has differing parental origins. Hybrids are scored for stability by counting the number of disruptions. SCHEMA also uses the information on residue pair disruptions to partition the protein into blocks that should not be interrupted by crossovers (analogous to the schema theory of genetic algorithms [119]). The algorithm was then used to show that crossover distributions in a number of experiments were preferentially allocated to avoid disrupting these blocks [118]. Though quite successful so far, this approach cannot differentiate between hybrids with different directionality also known as "mirror" chimeras (*i.e.*, A-B vs. B-A arrangement of segments), which have been shown to often have very different functional crossover profiles [43].

In our group, we have reevaluated the effect of having contacting residue pairs with different parental origins. Instead of always counting them as unfavorable, we view such pairs as places where potential clashes may occur between contacting residues. In Chapter 6, the SIRCH (Second-order mean-field Identification of Residue-residue Clashes in protein Hybrids [115]) procedure for evaluating protein hybrids is presented. In SIRCH, an extended, *second-order* mean-field description is used to elucidate the probabilities of all possible residue-residue combinations in a minimum Helmholtz free energy ensemble. The pairwise substitution patterns uncovered by the second-order mean-field description are then used to detect clashes in potential hybrids. SIRCH has been used to analyze pairwise substitution patterns in the dihydrofolate reductase (DHFR) enzyme and to assess the result of the recombination of *E. coli* and human glycinamide ribonucleotide (GAR) transformylases [43, 120, 121]. Results demonstrate that experimentally determined functional crossover positions for the GAR transformylases are consistent with the predicted residue-residue clashes. Analysis of these predicted clashes revealed that they primarily arise due to (i) the introduction of repulsive residue pairs such as +/+ or -/-, (ii) the disruption of hydrogen bonds due to the formation of donor/donor or acceptor/acceptor pairs, and (iii) the generation of steric clashes or cavities [122].

SCHEMA, SIRCH, and residue clash maps are increasingly being used to predict “smart” crossover sites [123] for experimental protocols that require preset crossover positions, such as SISDC, GeneReassembly, and synthetic oligonucleotide recombination methods. In addition, clash map information can be used in conjunction with protein design algorithms to suggest site-directed mutagenesis strategies for alleviating clashes in either parental sequences (upstream) or promising hybrids (downstream).

Section 1.5: Thesis Overview

The remainder of this thesis is organized as follows. Chapter 2 introduces quantitative models for the generation (error prone PCR) and recombination (DNA shuffling) of point mutations. Chapter 3 presents the ***e***Shuffle modeling framework for predicting the number, type, and location of crossovers in DNA shuffling experiments. Chapter 4 establishes the ***e***CodonOpt computational framework for designing parental DNA sequences for recombination through codon usage optimization. Chapter 5 develops the ***e***SCRATCHY model for the crossover statistics of the homology-independent SCRATCHY protocol. Chapter 6 offers a second-order mean-field based approach (SIRCH) for characterizing the complete set of residue-residue couplings consistent with a given protein structure. Finally, Chapter 7 concludes by offering some perspectives on the future of computational tools in directed evolution and protein engineering.

Section 1.6: References

1. Bradley, P., Chivian, D., Meiler, J., Misura, K.M., Rohl, C.A., Schief, W.R., Wedemeyer, W.J., Schueler-Furman, O., Murphy, P., Schonbrun, J., Strauss, C.E. & Baker, D. (2003), "Rosetta predictions in CASP5: successes, failures, and prospects for complete automation," *Proteins* **53 Suppl 6**: 457-468.
2. Dalby, P.A. (2003), "Optimising enzyme function by directed evolution," *Curr Opin Struct Biol* **13**(4): 500-505.
3. Bacher, J.M., Reiss, B.D. & Ellington, A.D. (2002), "Anticipatory evolution and DNA shuffling," *Genome Biol* **3**(8): REVIEWS1021.
4. Brakmann, S. (2001), "Discovery of superior enzymes by directed molecular evolution," *ChemBioChem* **2**(12): 865-871.
5. Petrounia, I.P. & Arnold, F.H. (2000), "Designed evolution of enzymatic properties," *Curr Opin Biotechnol* **11**(4): 325-330.
6. Schmidt-Dannert, C. (2001), "Directed evolution of single proteins, metabolic pathways, and viruses," *Biochemistry* **40**(44): 13125-13136.
7. Schneider, R.M., Medvedovska, Y., Hartl, I., Voelker, B., Chadwick, M.P., Russell, S.J., Cichutek, K. & Buchholz, C.J. (2003), "Directed evolution of retroviruses activatable by tumour-associated matrix metalloproteases," *Gene Ther* **10**(16): 1370-1380.
8. Schmidt-Dannert, C., Umeno, D. & Arnold, F.H. (2000), "Molecular breeding of carotenoid biosynthetic pathways," *Nat Biotechnol* **18**(7): 750-753.

9. Boder, E.T., Midelfort, K.S. & Wittrup, K.D. (2000), "Directed evolution of antibody fragments with monovalent femtomolar antigen-binding affinity," *Proc Natl Acad Sci USA* **97**(20): 10701-10705.

10. Bessler, C., Schmitt, J., Maurer, K.H. & Schmid, R.D. (2003), "Directed evolution of a bacterial alpha-amylase: toward enhanced pH-performance and higher specific activity," *Protein Sci* **12**(10): 2141-2149.

11. Miyazaki, K., Wintrode, P.L., Grayling, R.A., Rubingh, D.N. & Arnold, F.H. (2000), "Directed evolution study of temperature adaptation in a psychrophilic enzyme," *J Mol Biol* **297**(4): 1015-1026.

12. Baik, S.H., Ide, T., Yoshida, H., Kagami, O. & Harayama, S. (2003), "Significantly enhanced stability of glucose dehydrogenase by directed evolution," *Appl Microbiol Biotechnol* **61**(4): 329-335.

13. Reetz, M.T., Wilensek, S., Zha, D. & Jaeger, K.E. (2001), "Directed Evolution of an Enantioselective Enzyme through Combinatorial Multiple-Cassette Mutagenesis," *Angew Chem Int Ed Engl* **40**(19): 3589-3591.

14. Horsman, G.P., Liu, A.M., Henke, E., Bornscheuer, U.T. & Kazlauskas, R.J. (2003), "Mutations in distant residues moderately increase the enantioselectivity of Pseudomonas fluorescens esterase towards methyl 3bromo-2-methylpropanoate and ethyl 3phenylbutyrate," *Chemistry* **9**(9): 1933-1939.

15. Carr, R., Alexeeva, M., Enright, A., Eve, T.S., Dawson, M.J. & Turner, N.J. (2003), "Directed Evolution of an Amine Oxidase Possessing both Broad Substrate Specificity and High Enantioselectivity," *Angew Chem Int Ed Engl* **42**(39): 4807-4810.

16. Furukawa, K. (2000), "Engineering dioxygenases for efficient degradation of environmental pollutants," *Curr Opin Biotechnol* **11**: 244-249.

17. Wackett, L.P. (1998), "Directed evolution of new enzymes and pathways for environmental catalysis," *Ann NY Acac Sci* **864**: 142-152.

18. Bruhlmann, F. & Chen, W. (1999), "Tuning biphenyl dioxygenase for extended substrate specificity," *Biotechnol Bioeng* **63**(5): 544-551.

19. Yokobayashi, Y., Weiss, R. & Arnold, F.H. (2002), "Directed evolution of a genetic circuit," *Proc Natl Acad Sci USA* **99**(26): 16587-16591.

20. Whalen, R.G., Kaiwar, R., Soong, N.W. & Punnonen, J. (2001), "DNA shuffling and vaccines," *Curr Opin Mol Ther* **3**(1): 31-36.

21. Patten, P.A., Howard, R.J. & Stemmer, W.P. (1997), "Applications of DNA shuffling to pharmaceuticals and vaccines," *Curr Opin Biotechnol* **8**(6): 724-733.

22. Marzio, G., Verhoef, K., Vink, M. & Berkhout, B. (2001), "In vitro evolution of a highly replicating, doxycycline-dependent HIV for applications in vaccine studies," *Proc Natl Acad Sci USA* **98**(11): 6342-6347.

23. Moore, J.C., Jin, H., Kuchner, O. & Arnold, F.H. (1997), "Strategies for the in vitro evolution of protein function: enzyme evolution by random recombination of improved sequences," *J Mol Biol* **272**: 336-347.

24. Lin, H. & Cornish, V.W. (2002), "Screening and selection methods for large-scale analysis of protein function," *Angew Chem Int Ed Engl* **41**(23): 4402-4425.

25. Chen, W. & Georgiou, G. (2002), "Cell-Surface display of heterologous proteins: From high-throughput screening to environmental applications," *Biotechnol Bioeng* **79**(5): 496-503.

26. Olsen, M., Iverson, B. & Georgiou, G. (2000), "High-throughput screening of enzyme libraries," *Curr Opin Biotechnol* **11**(4): 331-337.

27. Leung, D.W., Chen, E. & Goeddel, D.V. (1989), "A method for random mutagenesis of a defined DNA segment using a modified polymerase chain reaction," *Technique* **1**: 11-15.

28. Lin-Goerke, J.L., Robbins, D.J. & Burczak, J.D. (1997), "PCR-based random mutagenesis using manganese and reduced dNTP concentration," *Biotechniques* **23**(3): 409-412.

29. Cadwell, R.C. & Joyce, G.F. (1994), "Mutagenic PCR," *PCR Methods Appl* **3**(6): S136-140.

30. Matsumura, I. & Ellington, A.D. (2002), "Mutagenic polymerase chain reaction of protein-coding genes for in vitro evolution," *Meth Mol Biol* **182**: 259-267.

31. Greener, A., Callahan, M. & Jerpseth, B. (1996), "An efficient random mutagenesis technique using an E. coli mutator strain," *Methods Mol Biol* **57**: 375-385.

32. Arnold, F.H. & Moore, J.C. (1997), "Optimizing industrial enzymes by directed evolution," *Adv Biochem Eng Biotechnol* **58**: 1-14.

33. Stemmer, W.P. (1994), "DNA shuffling by random fragmentation and reassembly: in vitro recombination for molecular evolution," *Proc Natl Acad Sci USA* **91**(22): 10747-10751.

34. Crameri, A., Raillard, S.A., Bermudez, E. & Stemmer, W.P. (1998), "DNA shuffling of a family of genes from diverse species accelerates directed evolution," *Nature* **391**(6664): 288-291.

35. Ness, J.E., Welch, M., Giver, L., Bueno, M., Cherry, J.R., Borchert, T.V., Stemmer, W.P. & Minshull, J. (1999), "DNA shuffling of subgenomic sequences of subtilisin," *Nat Biotechnol* **17**(9): 893-896.

36. Zhao, H., Giver, L., Shao, Z., Affholter, J.A. & Arnold, F.H. (1998), "Molecular evolution by staggered extension process (StEP) in vitro recombination," *Nat Biotechnol* **16**(3): 258-261.

37. Coco, W.M., Levinson, W.E., Crist, M.J., Hektor, H.J., Darzins, A., Pienkos, P.T., Squires, C.H. & Monticello, D.J. (2001), "DNA shuffling method for generating highly recombined genes and evolved enzymes," *Nat Biotechnol* **19**(4): 354-359.

38. Kikuchi, M., Ohnishi, K. & Harayama, S. (2000), "An effective family shuffling method using single-stranded DNA," *Gene* **243**(1-2): 133-137.

39. Moore, G.L., Maranas, C.D., Lutz, S. & Benkovic, S.J. (2001), "Predicting crossover generation in DNA shuffling," *Proc Natl Acad Sci USA* **98**(6): 3226-3231.

40. Ostermeier, M., Nixon, A.E., Shim, J.H. & Benkovic, S.J. (1999), "Combinatorial protein engineering by incremental truncation," *Proc Natl Acad Sci USA* **96**(7): 3562-3567.

41. Sieber, V., Martinez, C.A. & Arnold, F.H. (2001), "Libraries of hybrid proteins from distantly related sequences," *Nat Biotechnol* **19**(5): 456-460.

42. Ostermeier, M. (2003), "Theoretical distribution of truncation lengths in incremental truncation libraries," *Biotechnol Bioeng* **82**(5): 564-577.

43. Lutz, S., Ostermeier, M., Moore, G.L., Maranas, C.D. & Benkovic, S.J. (2001), "Creating multiple crossover libraries independent of sequence identity," *Proc Natl Acad Sci USA* **98**: 11248-11253.

44. Kawarasaki, Y., Griswold, K.E., Stevenson, J.D., Selzer, T., Benkovic, S.J., Iverson, B.L. & Georgiou, G. (2003), "Enhanced crossover SCRATCHY: construction and high-throughput screening of a combinatorial library containing multiple non-homologous crossovers," *Nucleic Acids Res* **31**(21): e126.

45. Hiraga, K. & Arnold, F.H. (2003), "General method for sequence-independent site-directed chimeragenesis," *J Mol Biol* **330**(2): 287-296.

46. Richardson, T.H., Tan, X., Frey, G., Callen, W., Cabell, M., Lam, D., Macomber, J., Short, J.M., Robertson, D.E. & Miller, C. (2002), "A novel, high performance enzyme for starch liquefaction. Discovery and optimization of a low pH, thermostable alpha-amylase," *J Biol Chem* **277**(29): 26501-26507.

47. O'Maille, P.E., Bakhtina, M. & Tsai, M.D. (2002), "Structure-based combinatorial protein engineering (SCOPE)," *J Mol Biol* **321**(4): 677-691.

48. Bogarad, L.D. & Deem, M.W. (1999), "A hierarchical approach to protein molecular evolution," *Proc Natl Acad Sci USA* **96**(6): 2591-2595.

49. Ostermeier, M. (2003), "Synthetic gene libraries: in search of the optimal diversity," *Trends Biotechnol* **21**(6): 244-247.

50. Ness, J.E., Kim, S., Gottman, A., Pak, R., Krebber, A., Borchert, T.V., Govindarajan, S., Mundorff, E.C. & Minshull, J. (2002), "Synthetic shuffling expands functional protein diversity by allowing amino acids to recombine independently," *Nat Biotechnol* **20**(12): 1251-1255.

51. Zha, D., Eipper, A. & Reetz, M.T. (2003), "Assembly of designed oligonucleotides as an efficient method for gene recombination: a new tool in directed evolution," *Chembiochem* **4**(1): 34-39.

52. Coco, W.M., Encell, L.P., Levinson, W.E., Crist, M.J., Loomis, A.K., Licato, L.L., Arensdorf, J.J., Sica, N., Pienkos, P.T. & Monticello, D.J. (2002), "Growth factor engineering by degenerate homoduplex gene family recombination," *Nat Biotechnol* **20**(12): 1246-1250.

53. Fernandez-Gacio, A., Uguen, M. & Fastrez, J. (2003), "Phage display as a tool for the directed evolution of enzymes," *Trends Biotechnol* **21**(9): 408-414.

54. Dower, W.J. & Mattheakis, L.C. (2002), "In vitro selection as a powerful tool for the applied evolution of proteins and peptides," *Curr Opin Chem Biol* **6**(3): 390-398.

55. Olsen, M.J., Stephens, D., Griffiths, D., Daugherty, P., Georgiou, G. & Iverson, B.L. (2000), "Function-based isolation of novel enzymes from a large library," *Nat Biotechnol* **18**(10): 1071-1074.

56. Weiss, G. & von Haeseler, A. (1995), "Modeling the polymerase chain reaction," *J Comput Biol* **2**(1): 49-61.

57. Stolovitzky, G. & Cecchi, G. (1996), "Efficiency of DNA replication in the polymerase chain reaction," *Proc Natl Acad Sci USA* **93**: 12947-12952.

58. Schnell, S. & Mendoza, C. (1997), "Theoretical description of the polymerase chain reaction," *J Theor Biol* **188**: 313-318.

59. Schnell, S. & Mendoza, C. (1997), "Enzymological considerations for a theoretical description of the Quantitative Competitive Polymerase Chain Reaction (QC-PCR)," *J Theor Biol* **184**: 433-440.

60. Valikanov, M.V. & Kapral, R. (1999), "Polymerase chain reaction: a Markov process approach," *J Theor Biol* **201**: 239-249.

61. Moore, G.L. & Maranas, C.D. (2000), "Modeling DNA mutation and recombination for directed evolution experiments," *J Theor Biol* **205**(3): 483-503.

62. Weiss, G. & von Haeseler, A. (1997), "A coalescent approach to the polymerase chain reaction," *Nucleic Acids Res* **25**(15): 3082-3087.

63. Wang, D., Zhao, C., Cheng, R. & Sun, F. (2000), "Estimation of the mutation rate during error-prone polymerase chain reaction," *J Comput Biol* **7**: 143-158.

64. Sun, F. (1998), "Modeling DNA Shuffling," *RECOMB '98, Proceedings of the second annual international conference on Computational molecular biology*: 251-257.

65. Sun, F. (1999), "Modeling DNA shuffling," *J Comput Biol* **6**(1): 77-90.

66. Moore, G.L. & Maranas, C.D. (2002), "Predicting out-of-sequence reassembly in DNA shuffling," *J Theor Biol* **219**(1): 9-17.

67. Maheshri, N. & Schaffer, D.V. (2003), "Computational and experimental analysis of DNA shuffling," *Proc Natl Acad Sci U S A* **100**(6): 3071-3076.

68. Moore, G.L. & Maranas, C.D. (2002), "eCodonOpt: a systematic computational framework for optimizing codon usage in directed evolution experiments," *Nucleic Acids Res* **30**(11): 2407-2416.

69. Wang, W. & Saven, J.G. (2002), "Designing gene libraries from protein profiles for combinatorial protein experiments," *Nucleic Acids Res* **30**(21): e120.

70. Bateman, A., Birney, E., Cerruti, L., Durbin, R., Etwiller, L., Eddy, S.R., Griffiths-Jones, S., Howe, K.L., Marshall, M. & Sonnhammer, E.L. (2002), "The Pfam protein families database," *Nucleic Acids Res* **30**(1): 276-280.

71. Harbury, P.B., Tidor, B. & Kim, P.S. (1995), "Repacking protein cores with backbone freedom: structure prediction for coiled coils," *Proc Natl Acad Sci USA* **92**(18): 8408-8412.

72. Harbury, P.B., Plecs, J.J., Tidor, B., Alber, T. & Kim, P.S. (1998), "High-resolution protein design with backbone freedom," *Science* **282**(5393): 1462-1467.

73. Klepeis, J.L., Floudas, C.A., Morikis, D., Tsokos, C.G., Argyropoulos, E., Spruce, L. & Lambris, J.D. (2003), "Integrated computational and experimental approach for lead optimization and design of compstatin variants with improved activity," *J Am Chem Soc* **125**(28): 8422-8423.

74. Keating, A.E., Malashkevich, V.N., Tidor, B. & Kim, P.S. (2001), "Side-chain repacking calculations for predicting structures and stabilities of heterodimeric coiled coils," *Proc Natl Acad Sci USA* **98**(26): 14825-14830.

75. Larson, S.M., England, J.L., Desjarlais, J.R. & Pande, V.S. (2002), "Thoroughly sampling sequence space: large-scale protein design of structural ensembles," *Protein Sci* **11**(12): 2804-2813.

76. Kraemer-Pecore, C.M., Lecomte, J.T. & Desjarlais, J.R. (2003), "A de novo redesign of the WW domain," *Protein Sci* **12**(10): 2194-2205.

77. Dunbrack Jr., R.L. (2002), "Rotamer libraries in the 21st century," *Curr Opin Struct Biol* **12**(4): 431-440.

78. MacKerell, A.D., Brooks, B., Brooks, C.L., Nilsson, L., Roux, B., Won, Y. & Karplus, M., *CHARMM: The energy function and its parameterization with an overview of the program*, in *The Encyclopedia of Computational Chemistry*, Schleyer, R., Editor. 1998, John Wiley & Sons: Chichester. p. 271-277.

79. Mayo, S.L., Olafson, B.D. & Goddard, W.A. (1990), "DREIDING - A Generic Force-Field for Molecular Simulations," *J Phys Chem* **94**(26): 8897-8909.

80. Cornell, W.D., Cieplak, P., Bayly, C.I., Gould, I.R., Merz Jr., K.M., Ferguson, D.M., Spellmeyer, D.C., Fox, T., Caldwell, J.W. & Kollman, P.A. (1995), "A second generation force field for the simulation of proteins, nucleic acids and organic molecules," *J Am Chem Soc* **117**: 5179-5197.

81. Scott, W.R., Hunenberger, P.H., Tironi, I.G., Mark, A.E., Billeter, S.R., Fennen, J., Torda, A.E., Huber, T., Kruger, P. & van Gunsteren, W.F. (1999), "The GROMOS biomolecular simulation program package," *J Phys Chem A* **103**(19): 3596-3607.

82. Chiu, T.L. & Goldstein, R.A. (1998), "Optimizing potentials for the inverse protein folding problem," *Protein Eng* **11**(9): 749-752.

83. Kuhlman, B. & Baker, D. (2000), "Native protein sequences are close to optimal for their structures," *Proc Natl Acad Sci USA* **97**(19): 10383-10388.

84. Looger, L.L. & Hellinga, H.W. (2001), "Generalized dead-end elimination algorithms make large-scale protein side-chain structure prediction tractable:

implications for protein design and structural genomics," *J Mol Biol* **307**(1): 429-445.

85. Gordon, D.B., Marshall, S.A. & Mayo, S.L. (1999), "Energy functions for protein design," *Curr Opin Struct Biol* **9**(4): 509-513.
86. Lovell, S.C., Word, J.M., Richardson, J.S. & Richardson, D.C. (2000), "The penultimate rotamer library," *Proteins* **40**(3): 389-408.
87. Dantas, G., Kuhlman, B., Callender, D., Wong, M. & Baker, D. (2003), "A large scale test of computational protein design: folding and stability of nine completely redesigned globular proteins," *J Mol Biol* **332**(2): 449-460.
88. Kuhlman, B., O'Neill, J.W., Kim, D.E., Zhang, K.Y. & Baker, D. (2002), "Accurate computer-based design of a new backbone conformation in the second turn of protein L," *J Mol Biol* **315**(3): 471-477.
89. Johnson, E.C., Lazar, G.A., Desjarlais, J.R. & Handel, T.M. (1999), "Solution structure and dynamics of a designed hydrophobic core variant of ubiquitin," *Structure Fold Des* **7**(8): 967-976.
90. Desjarlais, J.R. & Handel, T.M. (1995), "De novo design of the hydrophobic cores of proteins," *Protein Sci* **4**(10): 2006-2018.
91. Raha, K., Wollacott, A.M., Italia, M.J. & Desjarlais, J.R. (2000), "Prediction of amino acid sequence from structure," *Protein Sci* **9**(6): 1106-1119.
92. Xu, Z. & Farid, R.S. (2001), "Design, synthesis, and characterization of a novel hemoprotein," *Protein Sci* **10**(2): 236-249.
93. Jiang, X., Farid, H., Pistor, E. & Farid, R.S. (2000), "A new approach to the design of uniquely folded thermally stable proteins," *Protein Sci* **9**(2): 403-416.

94. Ogata, K., Jaramillo, A., Cohen, W., Briand, J.P., Connan, F., Choppin, J., Muller, S. & Wodak, S.J. (2003), "Automatic sequence design of major histocompatibility complex class I binding peptides impairing CD8+ T cell recognition," *J Biol Chem* **278**(2): 1281-1290.

95. Wernisch, L., Hery, S. & Wodak, S.J. (2000), "Automatic protein design with all atom force-fields by exact and heuristic optimization," *J Mol Biol* **301**(3): 713-736.

96. Jaramillo, A., Wernisch, L., Hery, S. & Wodak, S.J. (2002), "Folding free energy function selects native-like protein sequences in the core but not on the surface," *Proc Natl Acad Sci USA* **99**(21): 13554-13559.

97. Desjarlais, J.R. & Clarke, N.D. (1998), "Computer search algorithms in protein modification and design," *Curr Opin Struct Biol* **8**(4): 471-475.

98. Voigt, C.A., Gordon, D.B. & Mayo, S.L. (2000), "Trading accuracy for speed: A quantitative comparison of search algorithms in protein sequence design," *J Mol Biol* **299**(3): 789-803.

99. Desmet, J., Demaeyer, M., Hazes, B. & Lasters, I. (1992), "The dead-end elimination theorem and its use in protein side-chain positioning," *Nature* **356**(6369): 539-542.

100. Gordon, D.B. & Mayo, S.L. (1999), "Branch-and-Terminate: a combinatorial optimization algorithm for protein design," *Structure* **7**: 1089-1098.

101. Pierce, N.A., Spriet, J.A., Desmet, J. & Mayo, S.L. (2000), "Conformational Splitting: A More Powerful Criterion for Dead-End Elimination," *J Comp Chem* **21**(11): 999-1009.

102. Gordon, D.B. & Mayo, S.L. (1998), "Radical Performance Enhancements for Combinatorial Optimization Algorithms Based on the Dead-End Elimination Theorem," *J Comp Chem* **19**(13): 1505-1514.

103. Dahiyat, B.I. & Mayo, S.L. (1997), "De novo protein design: fully automated sequence selection," *Science* **278**(5335): 82-87.

104. Mooers, B.H., Datta, D., Baase, W.A., Zollars, E.S., Mayo, S.L. & Matthews, B.W. (2003), "Repacking the Core of T4 lysozyme by automated design," *J Mol Biol* **332**(3): 741-756.

105. Bolon, D.N., Marcus, J.S., Ross, S.A. & Mayo, S.L. (2003), "Prudent modeling of core polar residues in computational protein design," *J Mol Biol* **329**(3): 611-622.

106. Shimaoka, M., Shifman, J.M., Jing, H., Takagi, J., Mayo, S.L. & Springer, T.A. (2000), "Computational design of an integrin I domain stabilized in the open high affinity conformation," *Nat Struct Biol* **7**(8): 674-678.

107. Looger, L.L., Dwyer, M.A., Smith, J.J. & Hellinga, H.W. (2003), "Computational design of receptor and sensor proteins with novel functions," *Nature* **423**(6936): 185-190.

108. Dwyer, M.A., Looger, L.L. & Hellinga, H.W. (2003), "Computational design of a Zn2+ receptor that controls bacterial gene expression," *Proc Natl Acad Sci USA* **100**(20): 11255-11260.

109. Hayes, R.J., Bentzien, J., Ary, M.L., Hwang, M.Y., Jacinto, J.M., Vielmetter, J., Kundu, A. & Dahiyat, B.I. (2002), "Combining computational and experimental screening for rapid optimization of protein properties," *Proc Natl Acad Sci USA* **99**(25): 15926-25931.

110. Saven, J.G. (2001), "Designing protein energy landscapes," *Chem Rev* **101**(10): 3113-3130.

111. Voigt, C.A., Mayo, S.L., Arnold, F.H. & Wang, Z.G. (2001), "Computational method to reduce the search space for directed protein evolution," *Proc Natl Acad Sci USA* **98**(7): 3778-3783.

112. Lee, C. (1994), "Predicting protein mutant energetics by self-consistent ensemble optimization," *J Mol Biol* **236**(3): 918-939.

113. Koehl, P. & Delarue, M. (1994), "Application of a self-consistent mean field theory to predict protein side-chains conformation and estimate their conformational entropy," *J Mol Biol* **239**(2): 249-275.

114. Mendes, J., Soares, C.M. & Carrondo, M.A. (1999), "Improvement of side-chain modeling in proteins with the self-consistent mean field theory method based on an analysis of the factors influencing prediction," *Biopolymers* **50**(2): 111-131.

115. Moore, G.L. & Maranas, C.D. (2003), "Identifying residue-residue clashes in protein hybrids by using a second-order mean-field approach," *Proc Natl Acad Sci USA* **100**(9): 5091-5096.

116. Kono, H. & Saven, J.G. (2001), "Statistical theory for protein combinatorial libraries. Packing interactions, backbone flexibility, and the sequence variability of a main-chain structure," *J Mol Biol* **306**(3): 607-628.

117. Zou, J. & Saven, J.G. (2000), "Statistical theory of combinatorial libraries of folding proteins: energetic discrimination of a target structure," *J Mol Biol* **296**(1): 281-294.

118. Voigt, C.A., Martinez, C., Wang, Z.-G., Mayo, S.L. & Arnold, F.H. (2002), "Protein building blocks preserved by recombination," *Nat Struct Biol* **9**(7): 553-558.

119. Holland, J., *Adaptation in Natural and Artificial Systems*. 1975, Ann Arbor: The University of Michigan Press.

120. Ostermeier, M., Shim, J.H. & Benkovic, S.J. (1999), "A combinatorial approach to hybrid enzymes independent of DNA homology," *Nat Biotechnol* **17**(12): 1205-1209.

121. Lutz, S., Ostermeier, M. & Benkovic, S.J. (2001), "Rapid generation of incremental truncation libraries for protein engineering using alpha-phosphothioate nucleotides," *Nucleic Acids Res* **29**(4): E16.

122. Saraf, M.C. & Maranas, C.D. (2003), "Using a Residue Clash Map to Functionally Characterize Protein Recombination Hybrids," *Protein Eng*: accepted.

123. Meyer, M.M., Silberg, J.J., Voigt, C.A., Endelman, J.B., Mayo, S.L., Wang, Z.G. & Arnold, F.H. (2003), "Library analysis of SCHEMA-guided protein recombination," *Protein Sci* **12**(8): 1686-1693.

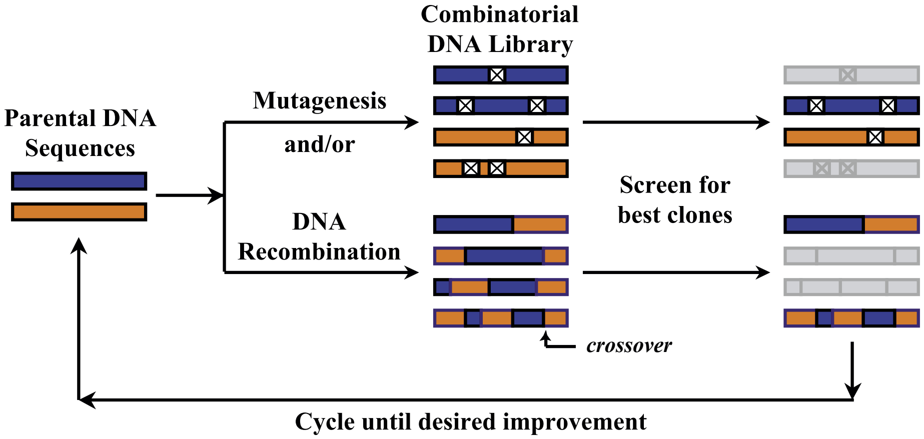


Figure 1.1: Schematic representation of the key steps of directed evolution experiments. *Crossovers* are defined as the junction points between segments from different parental sequences.

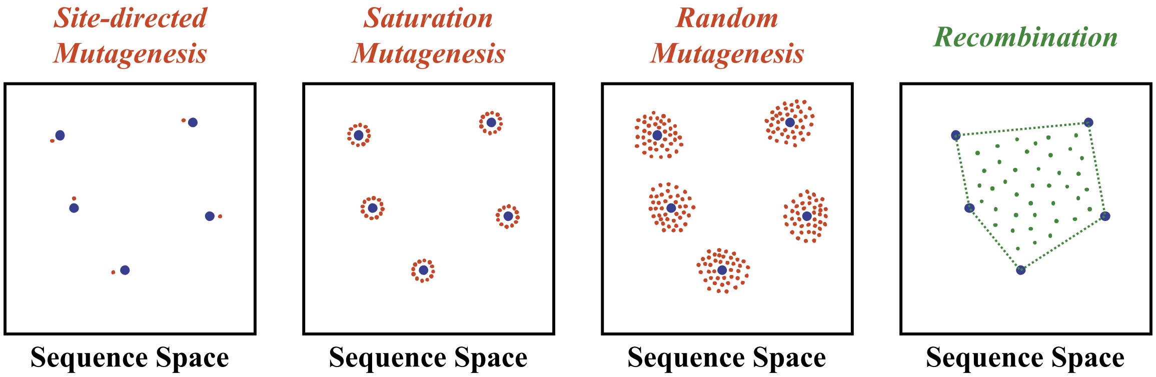


Figure 1.2: Depiction of the sequence space explored by mutagenesis (site-directed, saturation, and random) and recombination. Large blue dots represent parental sequences, while smaller red (mutagenesis) and green (recombination) dots represent combinatorial DNA library members.

Table 1.1: Summary of Methods for Combinatorial DNA Library Generation

Library Generation Method	**Advantages**	**Disadvantages**
Saturation Mutagenesis	✓ Complete assessment of all possible mutations at a particular residue position.	○ Must predetermine residue position. ○ Very limited exploration of sequence diversity.
Random Mutagenesis *Error-prone PCR, mutator strains*	✓ Easy, inexpensive setup.	○ Sequence diversity explored only near parental sequences. ○ Biased mutational frequencies.
Annealing-based Recombination *DNA shuffling, StEP, RACHITT, single-stranded shuffling*	✓ Straightforward PCR-based protocol. ✓ Large sets of parental sequences can be recombined.	○ Crossover positions biased toward stretches of sequence homology. ○ Severe bias toward parental sequence reassembly when parents have less than 60% sequence identity.
Nonhomologous Recombination *ITCHY, SHIPREC, SCRATCHY, SISDC, SCOPE, GeneReassembly*	✓ No bias toward regions of sequence identity. ✓ Multiple crossovers possible with SCRATCHY, SISDC, SCOPE, and GeneReassembly. ✓ Can predetermine crossover sites for SISDC, SCOPE, GeneReassembly.	○ More complicated protocols. ○ Only single-crossover hybrids generated with ITCHY and SHIPREC.
Synthetic Oligonucleotide Recombination *Synthetic shuffling, ADO, DHR*	✓ Crossovers can occur between closely spaced mutations. ✓ Rare mutations can be boosted with added oligonucleotides. ✓ Codon usage can be modified to comply with a particular host.	○ Average library activity can be lower due to broken couplings. ○ Generally more expensive, time-consuming to design oligonucleotides.

Chapter 2: Modeling the Generation/Recombination of Point Mutations

Section 2.1: Background

A key step in the directed evolution experimental cycle is the introduction of new genetic diversity to the library. There are two basic ways for introducing diversity: error-prone PCR and DNA recombination. Error-prone PCR protocols were used in early directed evolution experiments [1]. Polymerase chain reaction (PCR) is a DNA amplification technique in which an initial small amount of DNA is replicated in consecutive cycles increasing its concentration exponentially (see Figure 2.1).

The error-prone PCR replication process [2-4] intentionally introduces copying errors by imposing mutagenic reaction conditions (*e.g.*, through the addition of Mn^{+2} or Mg^{+2}). The first step of PCR is the denaturization of the DNA into single strands. The second step is the annealing of a primer to the DNA single strands. Primers consist of two DNA oligonucleotides with lengths of 15-30 base pairs complementary to the ends of the amplified region. The third step is primer extension by a polymerase (typically *Taq*). Nucleotides complementary to the single strand template are added by using the original sequence as a template, extending the complementary strands until normal DNA double strands are recovered. Unavoidable mutations occur in this step when non-complementary nucleotides are incorporated into the chain. Reference [5] reports mutation rates for *Taq* ranging from 10^{-7} up to 10^{-3} mutations per nucleotide polymerized. These mutation rates are nucleotide dependent [5, 6]. The control of these highly variable (spanning 4 orders of magnitude) copying errors is vital for mutagenesis since the “right” number of mutations will provide just enough diversity for evolutionary advancement without producing a build-up of deleterious errors. However, the ability of error prone PCR *alone* to successively improve a DNA sequence through continuously improving single point mutations is somewhat limited since the build-up of deleterious mutations typically overwhelms the beneficial ones. This has been recognized by researchers and currently DNA recombination, capable of filtering out deleterious mutations while retaining the improving ones, is almost exclusively being employed in directed evolution experiments.

Unlike error-prone PCR where no exchange of genetic material occurs between parental sequences, DNA recombination methods rely on the mixing and concatenation of genetic material from a number of parental sequences and are currently the preferred mutagenesis method in directed evolution experiments. Recombination protocols include DNA shuffling (sexual PCR) [7, 8], staggered extension process (StEP) [9] and random-priming recombination (RPR) [10]. A thorough review of currently employed DNA recombination protocols can be found in Section 1.2. Directed evolution experiments utilizing DNA recombination (shuffling) as the mutagenesis step are briefly described as follows (see also Figure 2.2).

First an initial set of parental sequences sharing a number of desired traits are selected for recombination. Next, the selected sequences undergo *random fragmentation* typically using DNaseI. Double-stranded fragments within a certain size range (*e.g.*, 100-200 bp) are retained. The retained fragments are then *reassembled* by thermocycling with a DNA polymerase (PCR *without added primers*). As in regular PCR, this involves first the *denaturization* of the double-stranded fragments into single-stranded ones. Denaturing is followed by *annealing* where single-stranded fragments anneal to other fragments overlapping by a sufficiently large number of complementary bases to form 3′ or 5′ overhangs. The third step is *polymerase extension* (see Figure 2.2). Note that the 3′ overhangs are not changed because DNA polymerase only possesses 5′ → 3′ activity. These three steps are repeated and the average fragment length increases after each cycle. After a number of cycles, DNA sequences of the original length are obtained. Finally, regular PCR with primers is utilized to amplify the reassembled strands. The key advantage of DNA shuffling over error-prone PCR is that it can recombine a large number of mutations within a few selection cycles quickly yielding functional blocks with combinations of beneficial mutations.

At the same time that new directed evolution success stories (see Section 1.1) are published and the potential for discovering truly novel biocatalysts is gaining acceptance, it is becoming apparent that the process is limited by key unanswered questions regarding the optimal mix, scheduling and setup of error-prone PCR and DNA recombination steps; the optimal selection of parental sequences for recombination; and the effect of parameters such as recombinatory fragment length, annealing temperature and number of shuffling cycles on the assembly of full length product sequences. To answer these questions, a set of quantitative models are introduced. In this chapter, a model of error-prone PCR is presented, and the predictions are compared to experimental data. Then, three models describing the DNA shuffling process are discussed. The first, (Random Fragmentation Model), describes the fragment size distribution after treatment with DNaseI. The second, (Fragment Assembly Model), predicts the fragment size distribution after each annealing/extension step. The third, (Sequence Matching Model), estimates the fraction of fully-assembled genes whose nucleotide sequence matches a target one. For all models, examples are provided along with comparisons with experimental data.

Section 2.2: Modeling Error-prone PCR

While lately error-prone PCR has been largely replaced by DNA recombination as the mutagenesis step, modeling single point mutations is still important since they will occur within any recombination protocol. Quantitative studies of PCR have so far addressed PCR efficiency [11], reaction kinetics [12], effect of annealing temperatures [13], and primer lengths [14, 15]. Reference [5] proposed the following simple equation, $f = Np / 2$, for predicting the overall error rate f after N PCR cycles given that the per cycle error rate is p. This relation does not account for the fact that copying errors depend on the nucleotide being replicated. For example, A miscopies to C, G or T with different probabilities [3, 6, 16]. This omission thus may yield inaccurate estimates.

In the proposed model, mutations that occur during the extension step when nucleotides are added via polymerase are treated as being nucleotide dependent. A per cycle mutation matrix $\mathcal{M}$ is defined that models these different mutation rates with elements M_{ij} representing the probability of nucleotide i mutating to nucleotide j:

$$\mathcal{M} = \begin{pmatrix} M_{AA} & M_{AT} & M_{AC} & M_{AG} \\ M_{TA} & M_{TT} & M_{TC} & M_{TG} \\ M_{CA} & M_{CT} & M_{CC} & M_{CG} \\ M_{GA} & M_{GT} & M_{GC} & M_{GG} \end{pmatrix}$$

These values depend on the experimental conditions. The per cycle mutation rate matrix $\mathcal{M}$ can then be used to identify the mutation rate matrix $\mathcal{C}^n$ after n extension steps. This matrix measures the mutation rates of a sequence obtained after n extension events starting from the original sequence. Because the occurrence of mutations in one extension step is independent of mutations that occurred in previous extension steps a recursive relation for $\mathcal{C}^n$ is derived as follows:

$$C_{ij}^{n} = \begin{cases} \delta_{ij}, & n = 0 \\ M_{ij}, & n = 1 \\ \sum\limits_{k=A,C,T,G} M_{kj} C_{ik}^{n-1}, & n \geq 2 \end{cases}$$

where δ_{ij} equals one if $i = j$ and zero otherwise.

However, after N PCR cycles not all sequences in the reaction mixture result after exactly N extensions of the original sequence. This is due to the fact that after a sequence is formed, it remains in the mixture to serve as a template in subsequent extension steps. For example, after three PCR cycles (see Figure 2.1), sixteen single strands of DNA are produced, of which two are the original DNA double strand ($n = 0$), six are the result of one extension step ($n = 1$), six are the result of two extension steps ($n = 2$), and two are the result of three extension steps ($n = 3$).

This result is generalized for N PCR cycles (see Figure 2.3). In the appendix of ref. [17], it is proven by induction that after N PCR cycles the number of sequences which are the product of exactly n extensions of the original DNA strand is equal to

$$Z_{N,n} = 2\binom{N}{n}$$

The total number of single-stranded sequences present in the reacting mixture after N PCR cycles is equal to $2 \cdot 2^N$ since every PCR cycle doubles their number. Therefore, the fraction of the sequences present in the reaction mixture after N PCR steps that are the result of n extension events is equal to

$$\frac{1}{2^N}\binom{N}{n}$$

This relation is used in conjunction with matrix $\mathcal{C}^n$ to construct matrix $\mathcal{P}^N$ with elements P_{ij}^N representing the probability of nucleotide i mutating to nucleotide j after N PCR cycles.

$$P_{ij}^N = \frac{1}{2^N}\sum_{n=0}^{N}\binom{N}{n}C_{ij}^n$$

By exploiting the assumption that mutations at different locations along the sequence are independent of each other, the probability $\Pi^N(S^0,S)$ of assembling a sequence S through successive single point mutations on an original sequence S^0 after N PCR cycles is given by

$$\Pi^N(S^0,S) = \frac{1}{2^N}\sum_{n=0}^{N}\binom{N}{n}\prod_{j=1}^{B}\left[C^n\right]_{s_j^0,s_j}$$

where B is the length of the two sequences and s_j^0 and s_j are the nucleotides at position j for sequences S^0 and S respectively. This relation provides the quantitative means to *a priori* estimate the fraction of the sequences obtained after N PCR steps that conform to some target sequence S given the mutation matrix $\mathcal{M}$. Therefore, by adjusting the reaction conditions to control the mutation rate, an experimenter can control the probability of achieving a desired target sequence.

Reference [6] reported a mutation rate matrix $\mathcal{P}^{13}$ after 13 PCR cycles shown in Table 2.1. The proposed model is next employed to recover the per cycle mutation rate matrix $\mathcal{M}$ (see Table 2.1). The average per-cycle mutation rate, assuming an equal concentration of each type of nucleotide, is calculated to be 0.016%. Note that the data presented in ref. [6] correspond to experimental conditions identical to the ones reported by ref. [18]. In the latter PCR study an average per cycle mutation rate of 0.02% is reported, which is very close to the value 0.016% that the proposed model predicts. Figure 2.4 illustrates the effect of GC content on the total number of mutations expected after 12 PCR cycles. Data from error-prone PCR with no Mn^{+2} added [6] is used to derive the per-cycle mutation matrix. As shown in Figure 2.4, a GC rich strand can reduce the number of mutations produced by almost one-half.

In the proposed model, the PCR efficiency is assumed to be 100% meaning that the amount of DNA present doubles from one cycle to the next. In practice, this is not always true since a lack of excess primer or nucleotides may result in incomplete amplification. This assumption affects both the calculation of the amount of DNA present after N cycles and $Z_{N,n}$. For a PCR efficiency ε an amplification of $(1 + \varepsilon)^N$ instead of 2^N is achieved. The calculation of $Z_{N,n}$ also needs to be changed. Furthermore, it is assumed that no mutational “hot spots”, or positions in the sequence with an increased mutation rate, are produced. The lack of “hot spots” is reported by ref. [3] and also by ref. [6]. Finally, nucleotide insertions and/or deletions are not modeled because such events are reported to comprise less than 5% of all mutations [6]. Nevertheless, by augmenting the mutation matrix $\mathcal{M}$ to include deletions and insertions in addition to nucleotide mutations such events can be accommodated at the expense of increased dimensionality.

Section 2.3: Modeling DNA Recombination

The modeling of three different aspects of the DNA recombination process is addressed:

1. *Random Fragmentation Model.* In this model the size distribution of the DNA fragments after treatment of the parental sequences with DNaseI is examined. This provides the necessary quantitative information regarding fragment size distribution necessary for modeling the subsequent DNA shuffling step.
2. *Fragment Assembly Model.* Given the initial fragment size distribution, the objective here is to model the fragment size distribution after each

annealing/extension step. This allows tracking of how effectively the recombination protocol assembles full length genes without regard to sequence or function of the assembled sequences.

3. *Target Sequence Matching.* After all shuffling cycles have been completed, the fraction of fully-assembled genes whose nucleotide sequence matches a given target (*e.g.*, AGGTCC) is quantified.

Section 2.3.1: Random Fragmentation Model

After a gene of length B is treated with DNaseI (random fragmentation), a random distribution of nucleotide fragments is obtained. Random fragmentation implies that each one of the B - 1 nucleotide-nucleotide bonds has an equal probability P_{cut} of being broken. The resulting fragment size probability distribution denoted by Q_L^0 is desired to describe the fraction of fragments of different lengths L present in the reaction mixture.

First the special case $L = B$ is addressed. The only possible way for a fragment of length B to result is if none of the B - 1 bonds are cut. The probability of a single bond remaining intact is (1 - P_{cut}). The random nature of fragmentation implies that bond breaking events are independent therefore,

$$Q_B^0 = (1 - P_{cut})^{B-1}$$

While the generation of a fragment of length B requires that all B - 1 bonds must remain intact, a fragment of length L can be formed after having different numbers of bonds being broken. The total number of broken bonds cannot exceed B - L because in that case at least one of the L - 1 bonds in a fragment of length L must break. Therefore, the calculation of Q_L^0 requires enumerating all possible ways of generating a fragment of length L after breaking $s = 1,\ldots, B - L$ bonds. Mathematically, this implies that Q_L^0 is equal to the sum of the products of the conditional probabilities $P_{L|s}$ of generating a fragment of length L given that s bonds are broken times the probability P_s of breaking s bonds:

$$Q_L^0 = \sum_{s=1}^{B-L} P_s P_{L|s}, \quad L = 1, \dots, B-1$$

There exist

$$\binom{B-1}{s}$$

alternatives for breaking s out of B - 1 bonds. Because bond cutting and bond preservation are independent events, each one of these alternatives has a probability

$$(P_{cut})^s (1 - P_{cut})^{B-1-s}$$

of occurring. By combining these two results we obtain:

$$P_s = \binom{B-1}{s} (P_{cut})^s (1 - P_{cut})^{B-1-s}$$

Random fragmentation implies that the order in which fragments are produced does not affect their respective probabilities of occurrence. For example, two cuts that produce fragments of lengths *a*, *b*, and *c* occur with the same probability as two cuts that produce fragments of lengths *c*, *a*, and *b*. This greatly simplifies the analysis by allowing the placement of the fragment of length *L* at the beginning without any loss of generality. Specifically, given that after breaking *s* bonds a fragment of length *L* is formed, the formation of the fragment of length *L* can be assumed to occur first without any loss of generality. This means that there exists

$$\binom{B-1-L}{s-1}$$

alternatives to form the remaining s - 1 cuts. Each one of these alternatives signifies a way of generating a fragment of length *L*. Because there exist

$$\binom{B-1}{s}$$

ways of creating *s* cuts, the conditional probability $P_{L|s}$ is equal to:

$$P_{L|s} = \binom{B-1-L}{s-1} \Big/ \binom{B-1}{s}$$

By combining the expressions for P_s and $P_{L|s}$ the following result for Q_L^0 is obtained:

$$Q_L^0 = \sum_{s=1}^{B-L} \binom{B-1-L}{s-1} P_{cut}^s \left(1 - P_{cut}\right)^{B-1-s}$$

After rearranging terms and invoking the binomial distribution properties this expression simplifies further to $Q_L{}^0 = P_{cut}\ (1 - P_{cut})^{L-1}$. Therefore, the fragment size probability distribution after random fragmentation is:

$$Q_L^0 = \begin{cases} P_{cut}^s \left(1 - P_{cut}\right)^{L-1}, & \text{for } 1 \le L \le B-1 \\ \left(1 - P_{cut}\right)^{B-1}, & \text{for } L = B \end{cases}$$

It is interesting to note that the resulting expressions for $L \le B - 1$ are independent of the length B of the original gene. Furthermore, it can be shown (see the appendix of ref. [17]) that for small values of P_{cut}, $Q_L{}^0$ approaches the exponential distribution $P_{cut}\ \exp(-P_{cut}L)$ (see also Table 2.2) with a mean of $1\ /\ P_{cut}$. A graph of the expected fragment size distribution after treatment with DNaseI is shown in Figure 2.5. Typically only a range of fragments between L_1 and L_2 are retained (*e.g.*, $L_1 = 50$, $L_2 = 150$) in subsequent DNA shuffling experiments. In this case, $Q_L{}^0$ must be renormalized. Note also that $Q_L{}^0$ is a monotonically decreasing function of L implying that irrespective of the size of B and the fragmentation intensity, quantified by P_{cut}, "small" fragments are always more ubiquitous than "large" ones.

Comparisons of the proposed model predictions with the bands obtained after agarose gel electrophoresis requires converting the fragment size distribution to corresponding signal intensities. The intensity of an agarose gel band, composed of fragments of length L, is proportional to the amount of intercalated ethidium bromide. This is approximately proportional to fragment length since ethidium bromide stains DNA sequences evenly. Therefore, the relative intensity of a band $I_L{}^0$ is proportional to the particular size fragment distribution $Q_L{}^0$ times the number of nucleotides L in the fragment. Thus, the following expression describes the relative intensity distribution.

$$I_L^0 = \begin{cases} L P_{cut}^s \left(1 - P_{cut}\right)^{L-1}, & \text{for } 1 \le L \le B-1 \\ B\left(1 - P_{cut}\right)^{B-1}, & \text{for } L = B \end{cases}$$

Unlike Q_L^0 which is monotonically decreasing, I_L^0 exhibits a sharp maximum in intensity for $L = 1 / P_{cut}$. It is interesting that the location of the peak depends only on the bond breaking probability P_{cut}.

A plot of relative gel intensities I_L^0 after the random fragmentation of a 1 kb gene for $P_{cut} = 0.01$, 0.02, and 0.04 is shown in Figure 2.6. As P_{cut} increases the peak migrates to smaller fragment lengths and the relative intensity distribution broadens. Density plots of the relative intensity shown in Figure 2.7 simulate the appearance of an agarose gel after DNaseI fragmentation of a 2 kb gene. Distributions for $P_{cut} = 0.002$, 0.004, 0.01, 0.04, and 0.1 are shown (top to bottom), which produce intensity peaks at $L = 500$, 250, 100, 25, and 10 bp respectively. The horizontal length scale shown is logarithmic due to the typical rate of DNA migration through a gel. These plots conform to the qualitative features exhibited by agarose gels.

These predictions are next compared with agarose gel data quantifying the fragment size distribution at different points in time. Table 2.3 summarizes the location of the intensity peak at different digestion times observed on an agarose gel for a system examined by [19]. The proposed model predicts that the peak intensity must occur at 1 / P_{cut} (bp). This implies that based on the experimentally observed peak intensities a model-based estimate of P_{cut} can be derived (see Table 2.3).

P_{cut} can alternatively be expressed as the extent of digestion

$$P_{cut} = \frac{C_b^0 - C_b}{C_b^0}$$

where C_b equals the concentration of unbroken nucleotide-nucleotide bonds and C_b^0 equals the initial concentration of bonds. C_b^0 can be represented as $C_{gene}B$, where C_{gene} is the concentration of the gene in solution. Because DNaseI is in excess, a first-order rate expression can be used to fit the rate of digestion:

$$C_b = C_b^0 \exp(-kt)$$

This leads to the following expression for P_{cut}:

$$P_{cut} = 1 - \exp(-kt)$$

After substituting the model predictions for P_{cut} a straight line is obtained after plotting $\ln(1 - P_{cut})$ versus t as shown in Figure 2.8. The slope of this straight line is equal to the rate constant of 0.320 hr^{-1} verifying the model predictions.

Section 2.3.2: Fragment Assembly Model

The goal of this model is to quantitatively describe how the fragment size distribution changes after a shuffling step. The value of this analysis is two-fold: first, it identifies how many shuffling cycles are necessary for reassembling the full length gene. Second, by modeling fragment size distribution, which is experimentally accessible, it provides a unique way of matching experimental with modeling results quantifying important parameters in the model. Such experimental studies based on the GeneScan technology [20, 21] are currently under investigation. In DNA shuffling, fragments are assembled by a PCR-like reaction without added primers. Denatured fragments prime each other during the annealing step creating regions of *overlap*, where annealing has taken place, and *overhangs*, where the fragments do not align (see Figure 2.9). The overhangs then serve as templates for *Taq*-catalyzed extension.

In the proposed model it is assumed that tertiary collisions are not important and that annealing only occurs between pairs of fragments. In compliance with *Taq* polymerase function, fragment assembly only occurs in the direction from 5′ to 3′. Sequences of length no greater than that of the original gene are assembled since the fragments are assumed to anneal only along areas of high homology. This requires that the gene does not have a high amount of repetition. The fraction of fragments that fail to anneal during each annealing step is represented by parameter NA which is assumed to depend on reaction conditions such as concentration and temperature. Fragment annealing is assumed to be governed by second order kinetics so that the probability of a fragment of length X and a fragment of length Y annealing is proportional to the product of their relative concentrations. The proportionality constant, denoted by $A(X,Y,V)$, is assumed to be a function of only overlap (V) and annealed fragment lengths (X,Y). A minimum overlap of V_{min} nucleotides is assumed to be necessary for annealing. V_{min} depends on the degree of homology shared by the parental sequences and reaction conditions and it is usually between 5 to 15 nucleotides [7]. Fragments with an overlap smaller than V_{min} are assumed to denature before extension takes place.

Given the original fragment size distribution Q_L^0 obtained after random fragmentation the next step is to quantify how this distribution will be reshaped after a shuffling step. The fragment probability size distribution after N shuffling cycles is denoted by Q_L^N. During the shuffling step pairs of DNA fragments randomly anneal and subsequently extend giving rise to successively larger DNA fragments from one shuffling cycle to the next. The fragment growth depends on the allowable overlap choices between fragments and their respective chances of annealing and extending. The allowable range of overlap for successful annealing between two fragments of lengths X and Y respectively is illustrated in Figure 2.10. The maximum possible overlap is equal to the length of the smaller of the two fragments, or min(X,Y). Every overlap value from V_{min} up to min(X,Y) - 1 occurs twice, once for each of the two fragment overhang orientations (5′ and 3′). The maximum overlap min(X,Y), however, occurs for $|X - Y| + 1$ internal annealing choices. This means that the multiplicity (degeneracy) d_V for different overlap values V is as follows:

$$d_V = \begin{cases} 2, & \text{for } V_{min} \le V \le \min(X,Y) - 1 \\ |X - Y| + 1, & \text{for } V = \min(X,Y) \end{cases}$$

The probability of observing a particular annealing choice shown in Figure 2.10 depends on the extent of overlap. The following annealing probability model is postulated where high or low overlap values are favored depending on the sign of the exponent α:

$$A(X,Y,V) = d_V V^{\alpha} \Big/ \sum_{V'=V_{\min}}^{min(X,Y)} d_{V'} (V')^{\alpha}$$

For α = -0.5 this annealing probability becomes inversely proportional to the square root of the overlap length as ref. [22] suggested, thus favoring shorter overlap values.

After establishing an annealing probability model the next step is to identify all mechanisms that generate a fragment of a particular length after a single annealing/extension cycle is completed. Six different pathways for producing a fragment of length L are considered which exhaustively enumerate all possibilities (Figure 2.11). A fragment of length L can be produced by (i) the extension of smaller fragments to length L (first two pathways); (ii) a fragment of length L that fails to extend after annealing (next three pathways); or (iii) a fragment of length L that fails to anneal (last pathway). The first five pathways listed above require two fragments to collide and anneal. These collision pathways depend on three probability terms. First, the fragments must anneal, and this occurs with probability (1 - NA) where NA denotes the probability of having a failed annealing. Second, the collision probability between two fragments of lengths X and Y is proportional to the product of their relative concentrations (or size probability distributions):

$$Q_X^{N-1} Q_Y^{N-1}$$

Because many fragment combinations can combine to form a fragment of a particular length L, a summation over all X and Y values that give fragments of length L after extension is necessary. Third, the annealing probability $A(X,Y,V)$ multiplying the product of the fragment size probability distributions is assumed to be a function of the fragment lengths X, Y and the nucleotide overlap V. These three factors govern the collision and annealing of two fragments. Each one of the five possible collision pathways are next examined in detail.

The first pathway (outer extension) describes the 5′ → 3′ successful annealing and extension of two fragments whose lengths X,Y are smaller than L and their overlap $V = X + Y - L$ is such that two single stranded fragments of length L are recovered after denaturing. The length of the first fragment X may vary between L_1 and L while the second fragment Y is bounded between $L - X + V_{min}$ and L. The three probability terms listed above result in the following expression for the size distribution of fragments of length L obtained through the outer extension pathway after the N^{th} shuffling cycle.

$$Q_L^N(outer\ extension) = (1 - NA)\sum_{X=L_1}^{L} Q_X^{N-1} \sum_{Y=L-x+V_{\min}}^{L} Q_Y^{N-1} A(X,Y,X+Y-L)$$

The second pathway (inner extension) considers the case when a smaller fragment anneals completely within a fragment larger than L. Given an appropriate placement the smaller fragment can then be extended to produce a fragment of length L. Similarly, the corresponding size probability distribution term accounting for the inner extension pathway is

$$Q_L^N(inner\ extension) = (1 - NA)\sum_{X=L+1}^{B} Q_X^{N-1} \sum_{Y=L_1}^{L\text{-}1} Q_Y^{N-1} A(X,Y,Y)$$

The third, fourth and fifth pathways describe cases when fragments of length L are retained after annealing but unsuccessful extension. This occurs when a 3′ overhang is created, causing the *Taq*-catalyzed extension to fail. The three failed extension pathways refer to the case where the second fragment is smaller than L (L- failed extension); larger than L (L+ failed extension); or equal to L (L failed extension). The following probability terms quantify the contribution of the third, fourth and fifth pathways to

$$Q_L^N(L^-\ failed\ extension) = (1 - NA)Q_L^{N-1}\sum_{Y=L_1}^{L} Q_Y^{N-1}\left(\sum_{V=V_{\min}}^{Y\text{-}1} A(L,Y,V) + (L-Y)A(L,Y,Y)\right)$$

$$Q_L^N(L^+\ failed\ extension) = (1 - NA)Q_L^{N-1}\sum_{Y=L+1}^{B} Q_Y^{N-1}\sum_{V=V_{\min}}^{L} A(L,Y,V)$$

$$Q_L^N(L\ failed\ extension) = (1 - NA)Q_L^{N-1}Q_L^{N-1}\sum_{V=V_{\min}}^{L\text{-}1} A(L,L,V)$$

Finally, fragments of length L may remain in the reaction mixture after failing to anneal. Failed annealing occurs with a probability of NA, so the following expression represents the portion of fragments of length L that remain unchanged after failed annealing:

$$Q_L^N(failed\ annealing) = (NA)Q_L^{N-1}$$

The sum of the contributions of the six pathways generates a recursive model for $Q_L{}^N$ that tracks the fragment size distribution from one shuffling cycle to the next. An internal consistency check verifies that

$$\sum_L Q_L^N = 1$$

is preserved. The only adjustable parameters in this model are the minimum allowable overlap V_{min}, the probability of failed annealing NA, and the exponent α in the annealing probability expression. Resolving the recursion requires going back shuffling steps, eventually encountering as an input the original fragment size distribution $Q_L{}^0$ obtained after random fragmentation.

Figure 2.12 illustrates the fragment size distribution predicted by the model after 5, 10, and 15 shuffling cycles. The original 1 kb gene is first randomly fragmented and only fragments with sizes between 50 and 150 bp are retained for shuffling. After only 5 shuffling steps the signature of the original fragment pool is still evident in the form of a sharp peak. After 10 cycles this sharp peak is nearly eliminated and a single broad maximum can be found in the fragment size distribution. Finally, after 15 cycles this maximum has migrated to reach the end of the length range and a large portion of the fragments have assembled into full length genes.

Comparisons with experimental data are encouraging. Reference [8] initially studied the assembly of a 1 kb gene. The experiment began with random fragmentation to an approximate mean fragment length of 100 bp verified on an agarose gel implying a value for P_{cut} of 1%. Then fragments sized from 10 to 50 bp were assembled, and aliquots taken after N = 25, 30, 35, 40, and 45 shuffling steps were analyzed on a gel to monitor the progress of the reaction. After 25 cycles, an intensity peak could be seen at approximately L = 250. After 30 cycles, a peak could be seen near L = 450. As the assembly progressed further, the fluorescence broadened, and full length genes were reassembled. The proposed model matches these experimental observations as illustrated in Figure 2.13. Parameter values of P_{cut} = 1%, L_1 = 10 and L_2 = 50 are selected to match the ones employed in Stemmer's work. An α value of -0.5 was chosen [22]. Furthermore, the last two parameters were set at NA = 70% and V_{min} = 5.

Section 2.3.3: Target Sequence Matching

In the Fragment Assembly Model, the process of recovering full length sequences was analyzed without regard to the nucleotide sequence of the assembled genes. In the Target Sequence Matching model, the goal is to relate the nucleotide sequence of the fully assembled genes, obtained after recombination, to the nucleotide sequence and concentration of the parental sequences and experimental conditions. Specifically, given the precise nucleotide sequence of the parental sequences available for recombination, the objective is to find the fraction of the fully assembled sequences whose nucleotide sequence matches a prespecified target (*e.g.*, ATTGG). Reference [23] studied a simplified model assuming that the lengths of the fragments to be reassembled are less than the distances between mutations. Later, refs. [24, 25] considered larger fragment lengths and addressed the case of single [24] and multiple [25] mutations. By building on these contributions, this modeling effort addresses the general case of multiple mutations per strand and arbitrary selections for the fragment lengths.

In our analysis, the nucleotide sequence of only complete DNA products of full length is analyzed. The fraction of the sequences achieving full length can be estimated based on the results presented in the previous section. Also, the parental sequences are assumed to have a high degree of homology so that fragment annealing is possible along the entire gene length. As in the fragment assembly model, a minimum overlap of V_{min} nucleotides is assumed to be necessary for annealing and subsequent assembly, and assembly is assumed to proceed only from 5′ to 3′. Furthermore, it is assumed that the assembly process from a position i until the end B of the sequence is independent of assembly that has occurred before position i. In other words, the annealing of a fragment is independent of all prior fragment annealing that occurred in previous shuffling cycles.

Therefore, if P_i is the probability of reproducing the portion of a target sequence between positions i and its end B then P_i is independent of all P_j where $j < i$.

The correct assembly of a target sequence is achieved if and only if a cascade of four independent events occurs, as shown in Figure 2.14. Each one of these events contributes a probability term to P_i. The first step is to choose a fragment of length L to add to the sequence. Assuming random fragmentation, a fragment of length L is chosen with probability Q_L^0 discussed earlier. The second step in the assembly process is the annealing of the fragment of length L to the rest of the previously assembled sequence. The overlap must be at least V_{min} nucleotides. Thus, the nonoverlapping portion of the fragment adds at most L - V_{min} new nucleotides to the sequence. Therefore, there are L - V_{min} possible ways for a fragment to align itself during annealing with overlaps V ranging from V_{min} to L - 1. The probability of adding L - V new nucleotides with a fragment of length L is denoted as $A_{L-V,L}$ and is defined identically with the annealing probability $A(X,Y,V)$ described in the previous section.

$$A_{L-V,L} = V^{\alpha} \Big/ \sum_{V'=V_{\min}}^{L-1} (V')^{\alpha}$$

After summing up over all possible overlap values this contributes, to P_i, a factor of

$$\sum_{V=V_{min}}^{L-1} A_{L-V,L}$$

The third step is to calculate the probability that the extended sequences will contribute nucleotides that match the ones in the target sequence. Starting from a nucleotide at position i and assuming that a fragment of length L has annealed with an overlap of V nucleotides, the probability of matching the target nucleotide sequence from i to position i + (L - V) - 1 is equal to the fraction of the parental sequences that exactly match the target sequence from position i to position i + (L - V) - 1. Let parameter $\Delta_{a,b}$ denote the number of parental sequences that match the target sequence from positions a to b. Matching between positions i and i + (L - V) - 1 then occurs with a probability equal to $\Delta_{i,i+L-V-1} / K$, where K is the number of parents available for recombination. The fourth and final step is to calculate the probability of reproducing the remainder of the target sequence after adding $L - V$ new nucleotides. Because the annealing of additional fragments is independent of prior additions, simple multiplication by P_{i+L-V} suffices. This establishes a function for P_i that must be evaluated recursively. These four steps result in the expression for P_i shown below, where B is the sequence length in nucleotides, L_1 and L_2 are the smallest and largest recombinatory fragments, V_{min} is the minimum annealing overlap, and K is the number of parental sequences.

$$P_i = \begin{cases} 1, & i > B \\ \Delta_{i,i}/K, & i = B \\ \sum_{L=L_1}^{L_2} Q_L^0 \left[\sum_{V=V_{\min}}^{L-1} A_{L\text{-}V,L} \left(\Delta_{i,i+L-V-1}/K \right) P_{i+L-V} \right], & i < B \end{cases}$$

The above recursive formula calculates the probability P_i of obtaining an assembled sequence that is identical with some target sequence S after nucleotide position i. Therefore, P_1 is equal to the probability of assembling a sequence identical to the target. This target may be either a specific pattern or an entire gene.

The predictions of the Sequence Matching Model are consistent with experimental data. In ref. [7], two markers 75 bp apart were recombined from random fragments of size between 100 to 200 bp and reported that only 11% of the reassembled fragments contained both mutations. Note that independent assembly of the two mutations would have predicted a 25% value. Assuming a required minimum overlap for annealing of $V_{min} = 15$ and $\alpha = -1/2$, this model estimates this probability for the average fragment size of $L = 150$ to be 12.4%, which is very close to the experimentally observed one.

Next the possibility of increasing the probability of containing both mutations in the recombined sequences by appropriately choosing the fragment length is examined. The estimated probability of assembling a two-mutation sequence is plotted as a function of fragment length in Figure 2.15. As shown in Figure 2.15, this probability is a strong function of fragment length exhibiting a sharp maximum at around L = 110 bp of 21.4%. These results clearly demonstrate the importance of being able to predict this "right" fragment length.

Further comparisons with experimental results [26] are shown in Tables 2.4 and 2.5. In Table 2.4, the result of recombining a portion of a 1.3 kb sequence with four mutations and a portion with no mutations is shown. The gene was digested for two minutes with DNaseI, leading to an estimated P_{cut} value of 0.83% (see Table 2.3). Fragments sized less than 50 bp were used for reassembly, so values of $L_1 = 30$, $L_2 = 50$ and $V_{min} = 15$ were used to approximate this. The modeling results confirm the experimentally observed tendency of the mutations at positions 35 and 47 to be "linked". Table 2.5 shows this tendency more clearly by examining only the recombination of the closely spaced mutations.

Section 2.4: Summary and Conclusions

In this chapter, quantitative models for predicting the outcome of DNaseI fragmentation, error prone PCR and DNA shuffling experiments were introduced. Specifically, the Random Fragmentation Model and the Fragment Assembly Models provided the quantitative means of tracking the size probability distribution of fragments in the reacting mixture during DNaseI fragmentation and DNA shuffling respectively. On the other hand, the PCR Model and the Sequence Matching Model establish a formalism for estimating the probability of matching a prespecified nucleotide target.

Section 2.5: References

1. Arnold, F.H. (1996), "Directed Evolution: creating Biocatalysts for the Future," *Chem Eng Sci* **51**: 5091-5102.
2. Leung, D.W., Chen, E. & Goeddel, D.V. (1989), "A method for random mutagenesis of a defined DNA segment using a modified polymerase chain reaction," *Technique* **1**: 11-15.
3. Cadwell, R.C. & Joyce, G.F. (1992), "Randomization of Genes by PCR Mutagenesis," *PCR Methods Appl* **2**: 28-33.
4. Lin-Goerke, J.L., Robbins, D.J. & Burczak, J.D. (1997), "PCR-Based Random Mutagenesis Using Manganese and Reduced dNTP Concentration," *Biotechniques* **23**: 409-412.
5. Eckert, K.A. & Kunkel, T.A. (1991), "DNA Polymerase Fidelity and the Polymerase Chain Reaction," *PCR Methods Appl* **1**: 17-24.
6. Shafikhani, S., Siegel, R.A., Ferrari, E. & Schellenberger, V. (1997), "Generation of Large Libraries of Random Mutants in Bacillus subtilis by PCR-Based Plasmid Multimerization," *Biotechniques* **23**: 304-310.
7. Stemmer, W.P.C. (1994), "DNA shuffling by random fragmentation and reassembly: in vitro recombination for molecular evolution," *Proc Natl Acad Sci USA* **91**: 10747-10751.
8. Stemmer, W.P.C. (1994), "Rapid evolution of a protein in vitro by DNA shuffling," *Nature* **370**: 389-391.
9. Zhao, H., Giver, L., Shao, Z., Affholter, J.A. & Arnold, F.H. (1998), "Molecular evolution by staggered extension process (StEP) in vitro recombination," *Nature Biotech* **16**: 258-261.
10. Shao, Z., Zhao, H., Giver, L. & Arnold, F.H. (1998), "Random-priming in vitro recombination: an effective tool for directed evolution," *Nucl Acids Res* **26**: 681-683.

11. Weiss, G. & Haeseler, A.V. (1995), "Modeling the Polymerase Chain Reaction," *J Comput Biol* **2**: 49-61.
12. Hsu, J.T., Das, S. & Mohapatra, S. (1997), "Polymerase Chain Reaction Engineering," *Biotechnol Bioeng* **55**: 359-366.
13. Rychlik, W., Spencer, W.J. & Rhoads, R.E. (1990), "Optimization of the annealing temperature for DNA amplification in vitro," *Nucl Acids Res* **18**: 6409-6412.
14. Sakuma, Y. & Nishigaki, K. (1994), "Computer Prediction of General PCR Products Based on Dynamical Solution Structures of DNA," *J Biochem* **116**: 736-741.
15. Wu, D.Y., Ugozzoli, L., Pal, B.K., Qian, J. & Wallace, R.B. (1991), "The Effect of Temperature and Oligonucleotide Primer Length on the Specificity and Efficiency of Amplification by the Polymerase Chain Reaction," *DNA Cell Biol* **10**: 233-238.
16. Lin, Z., Thorsen, T. & Arnold, F.H. (1999), "Functional Expression of Horseradish Peroxidase in E. coli by Directed Evolution," *Biotechnol Prog* **15**: 467-471.
17. Moore, G.L. & Maranas, C.D. (2000), "Modeling DNA mutation and recombination for directed evolution experiments," *J Theor Biol* **205**(3): 483-503.
18. Ling, L.L., Keohavong, P., Dais, C. & Thilly, W.G. (1991), "Optimization of the Polymerase Chain Reaction with Regard to Fidelity: modified T7, Taq, and Vent DNA Polymerases," *PCR Methods Appl* **1**: 63-69.
19. Volkov, A.A. & Arnold, F.H. (1999), "Methods for in vitro DNA Recombination and Chimeragenesis," *to appear in Methods Enzymol*.
20. Smith, N.R., Li, A., Aldersley, M., High, A.S., Markham, A.F. & Robinson, P.A. (1997), "Rapid determination of the complexity of cDNA bands extracted from DDRT-PCR polyacrylamide gels," *Nucl Acids Res* **25**: 3552-3554.

21. Fuqua, S.A., Fitzgerald, S.D. & McGuire, W.L. (1990), "A simple polymerase chain reaction method for detection and cloning of low-abundance transcripts," *Biotechniques* **9**: 206-211.

22. Wetnur, J.G. & Davidson, N. (1967), "Kinetics of Renaturization of DNA," *J Mol Biol* **31**: 349-370.

23. Moore, J.C., Jin, H., Kuchner, O. & Arnold, F.H. (1997), "Strategies for the in vitro Evolution of Protein Function: enzyme Evolution by Random Recombination of Improved Sequences," *J Mol Biol* **272**: 336-347.

24. Sun, F. (1998), "Modeling DNA Shuffling," *Proc 1998 2nd Ann Internatl Conf Comput Mol Biol*: 251.

25. Sun, F. (1999), "Modeling DNA Shuffling," *J Comput Biol* **6**: 77-90.

26. Zhao, H. & Arnold, F.H. (1997), "Optimization of DNA shuffling for high fidelity recombination," *Nucl Acids Res* **25**: 1307-1308.

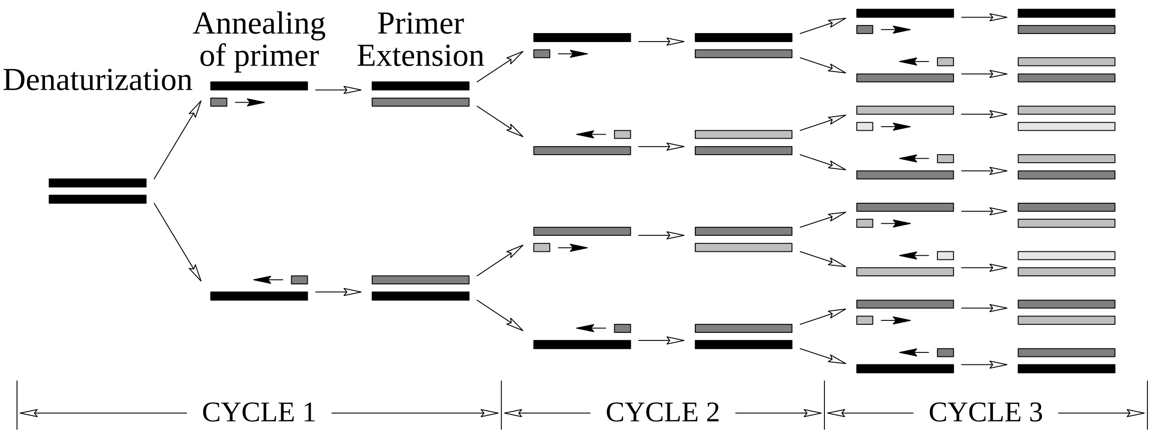


Figure 2.1: Three cycles of PCR produce $2^3 = 8$ total strands after the third cycle, or 16 single strands of nucleotides. Of these 16, 2 are the original DNA double strand, 6 are the result of 1 extension step, 6 are the result of 2 extension steps, and 2 are the result of 3 extension steps. Strands are shown more lightly shaded as they undergo more extension steps.

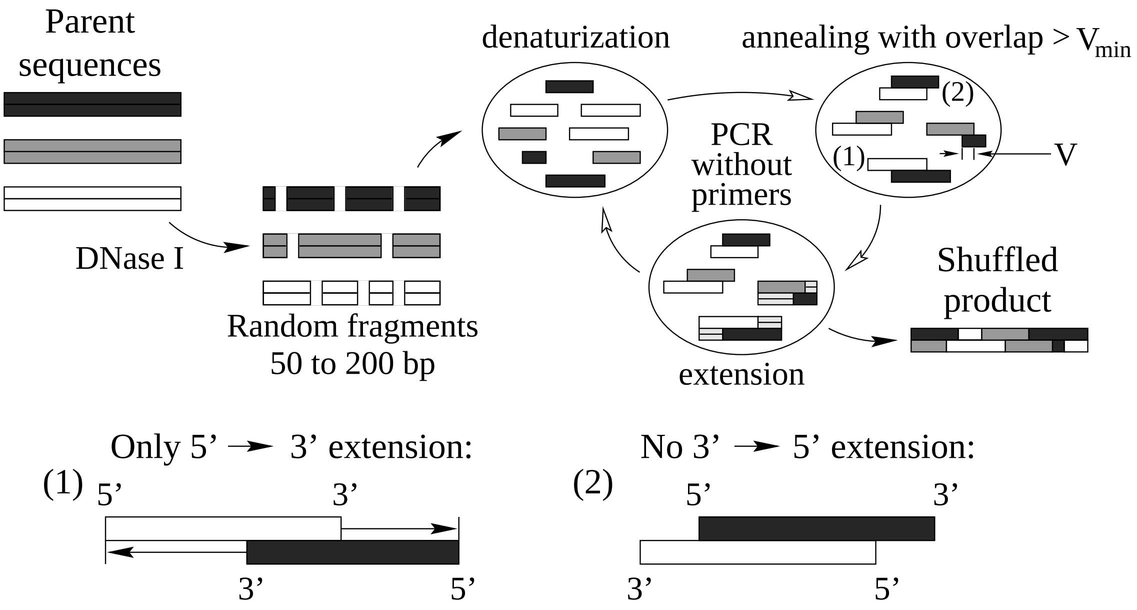


Figure 2.2: DNA shuffling occurs in three steps, the most important of which is a PCR reaction without primers in which reassembly of parental sequences occurs. The product will have a combination of genetic features from all of the parental sequences.

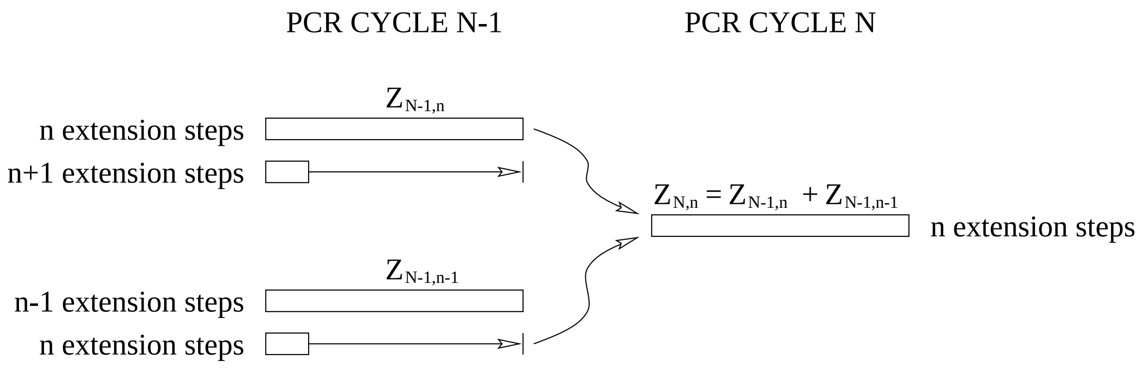


Figure 2.3: After the N^{th} PCR cycle, a strand that is produced after *n* extension steps is either one that was just produced in the N^{th} PCR cycle or one that was already in the reaction mixture before the N^{th} PCR cycle began.

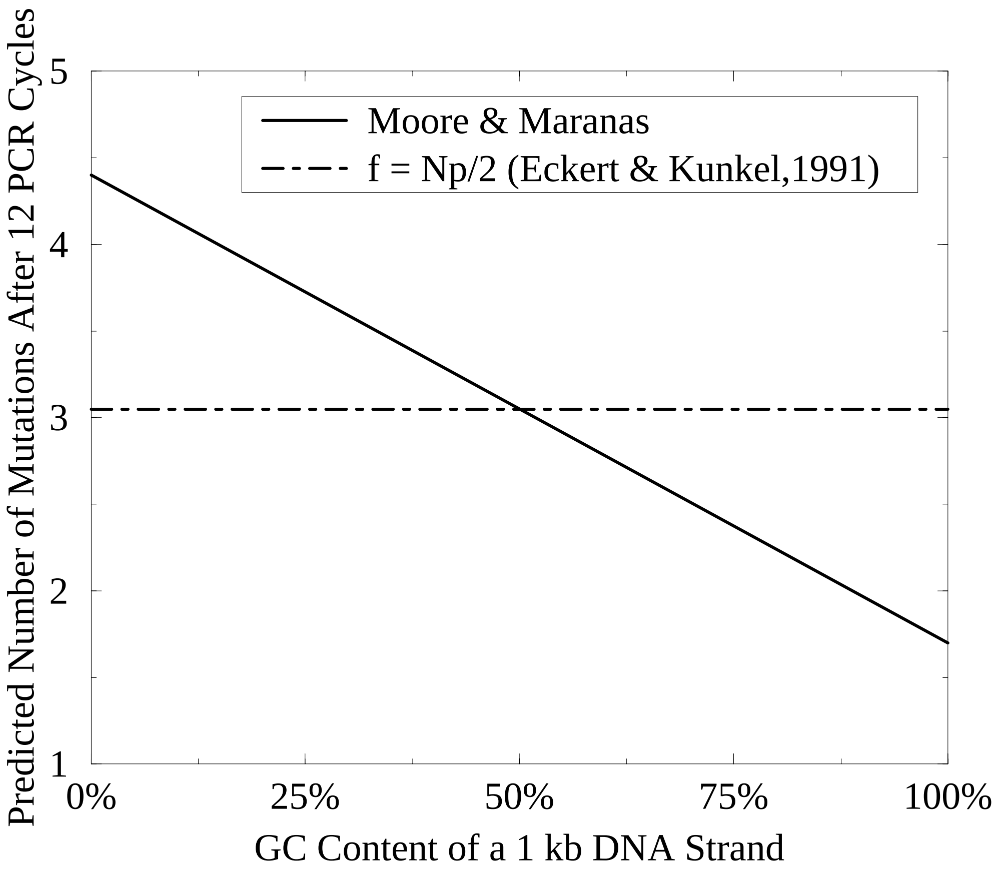


Figure 2.4: The GC content of a DNA strand can significantly alter the number of mutations produced by error-prone PCR. Data shown here is for a 12-cycle PCR with no Mn^{+2} added.

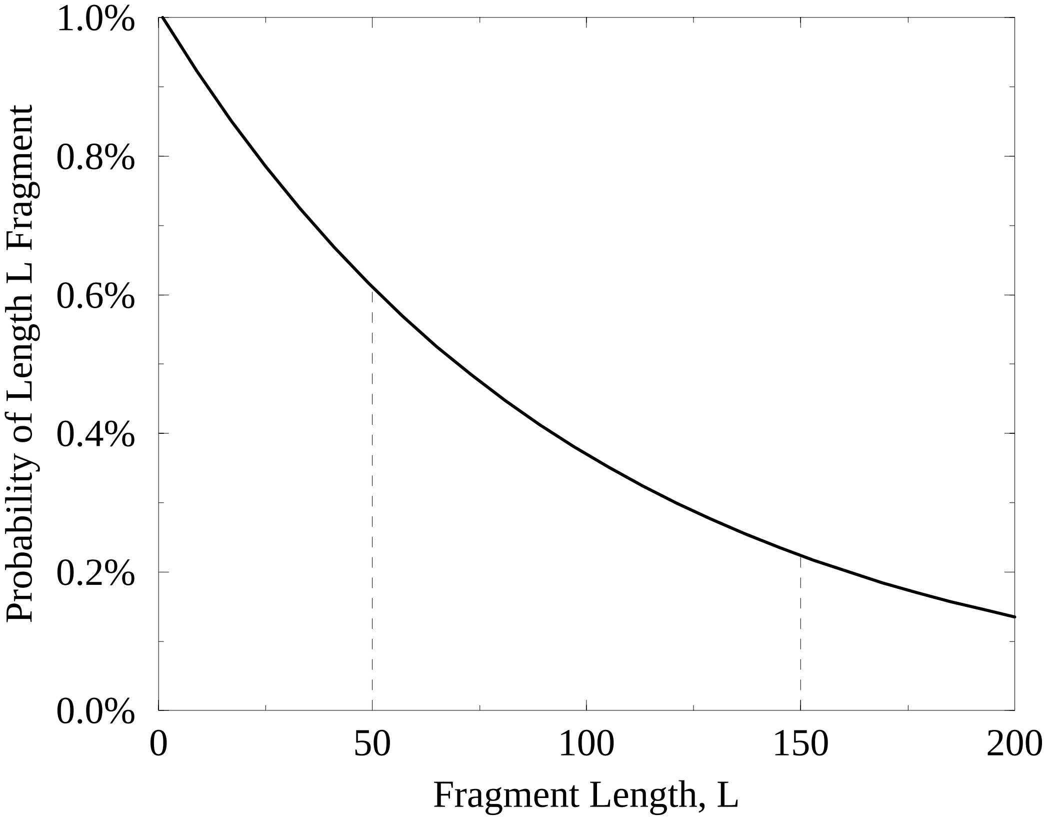


Figure 2.5: Fragment size distribution after a 1000 bp gene is fragmented with DNaseI with $P_{cut} = 0.01$ resulting in a mean fragment length of 100 bp. The dotted lines indicate that only a portion of these fragments are retained for shuffling.

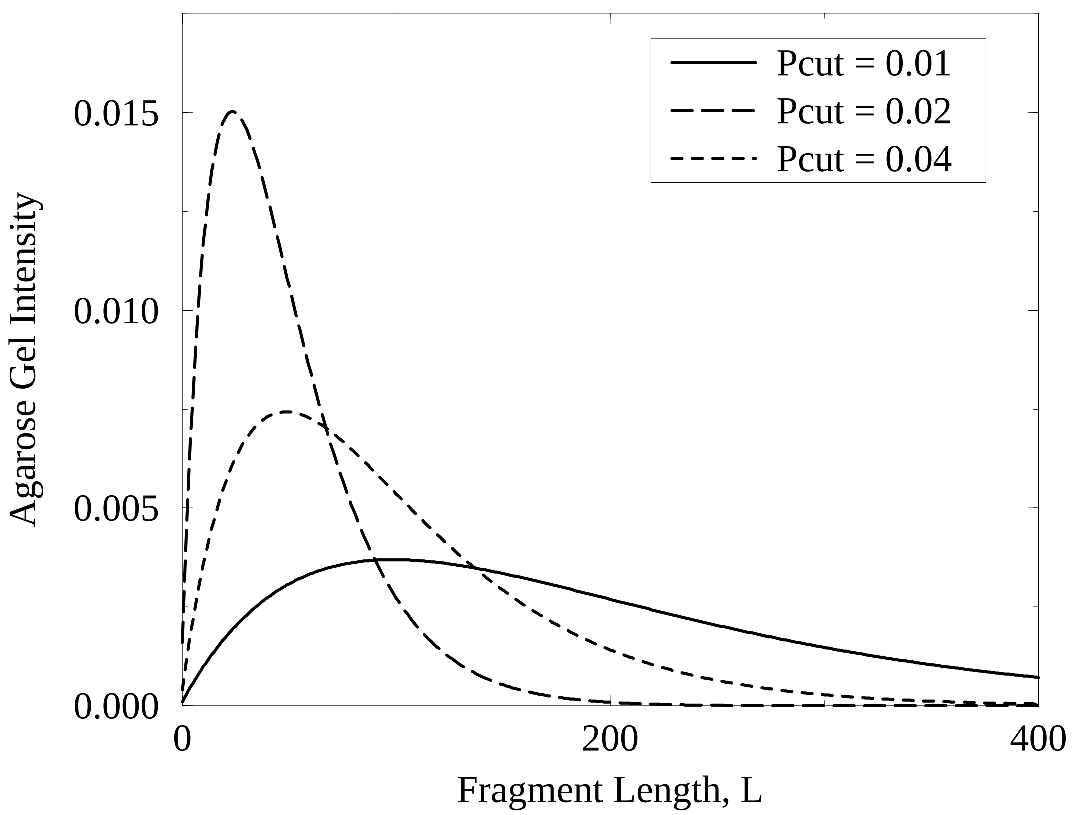


Figure 2.6: Calculated agarose gel intensities for P_{cut} = 0.01, 0.02 and 0.04 for a 1 kb gene.

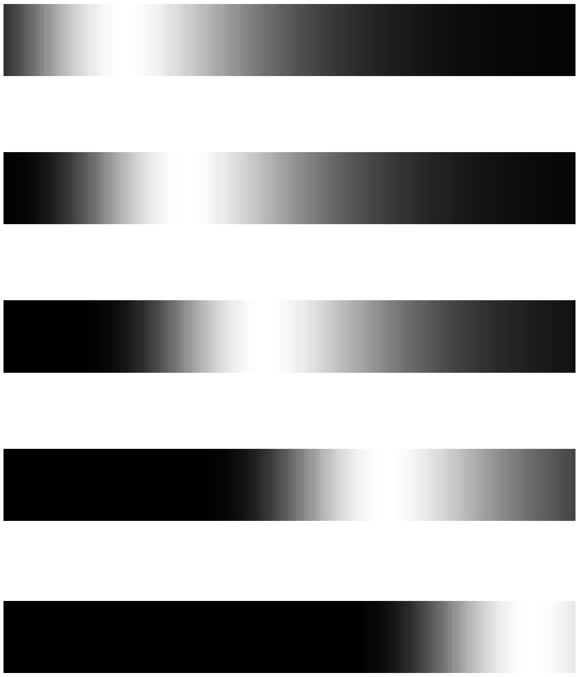


Figure 2.7: Calculated agarose gel intensities for P_{cut} = 0.002, 0.004, 0.01, 0.04 and 0.2 (top to bottom). The gel runs from a maximum of L = 2000 at the left down to L = 1 at the right.

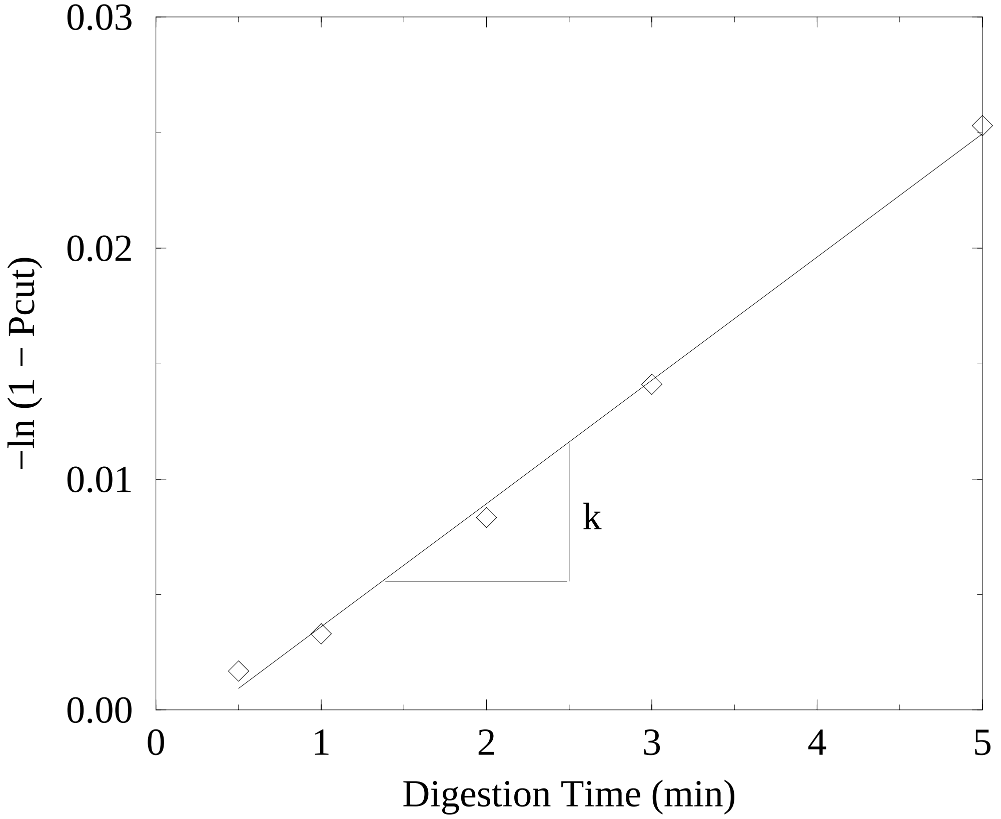


Figure 2.8: First order kinetics of DNaseI digestion.

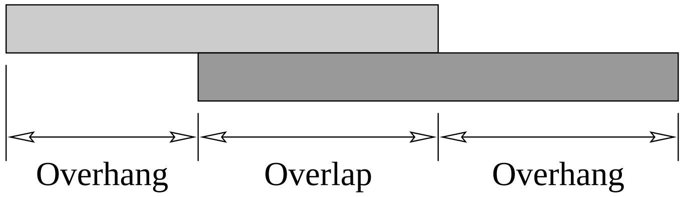


Figure 2.9: The regions of overlap and overhang for a pair of annealed fragments.

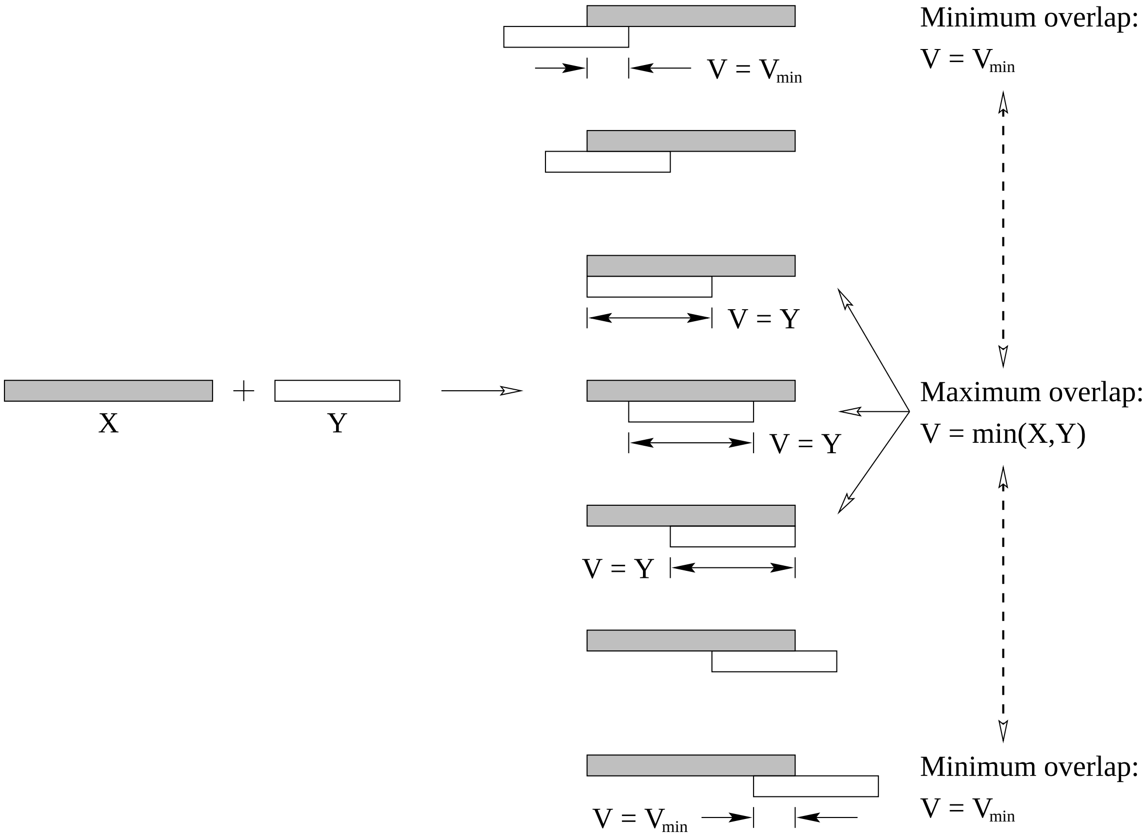


Figure 2.10: Possible overlap alternatives between two annealed sequences.

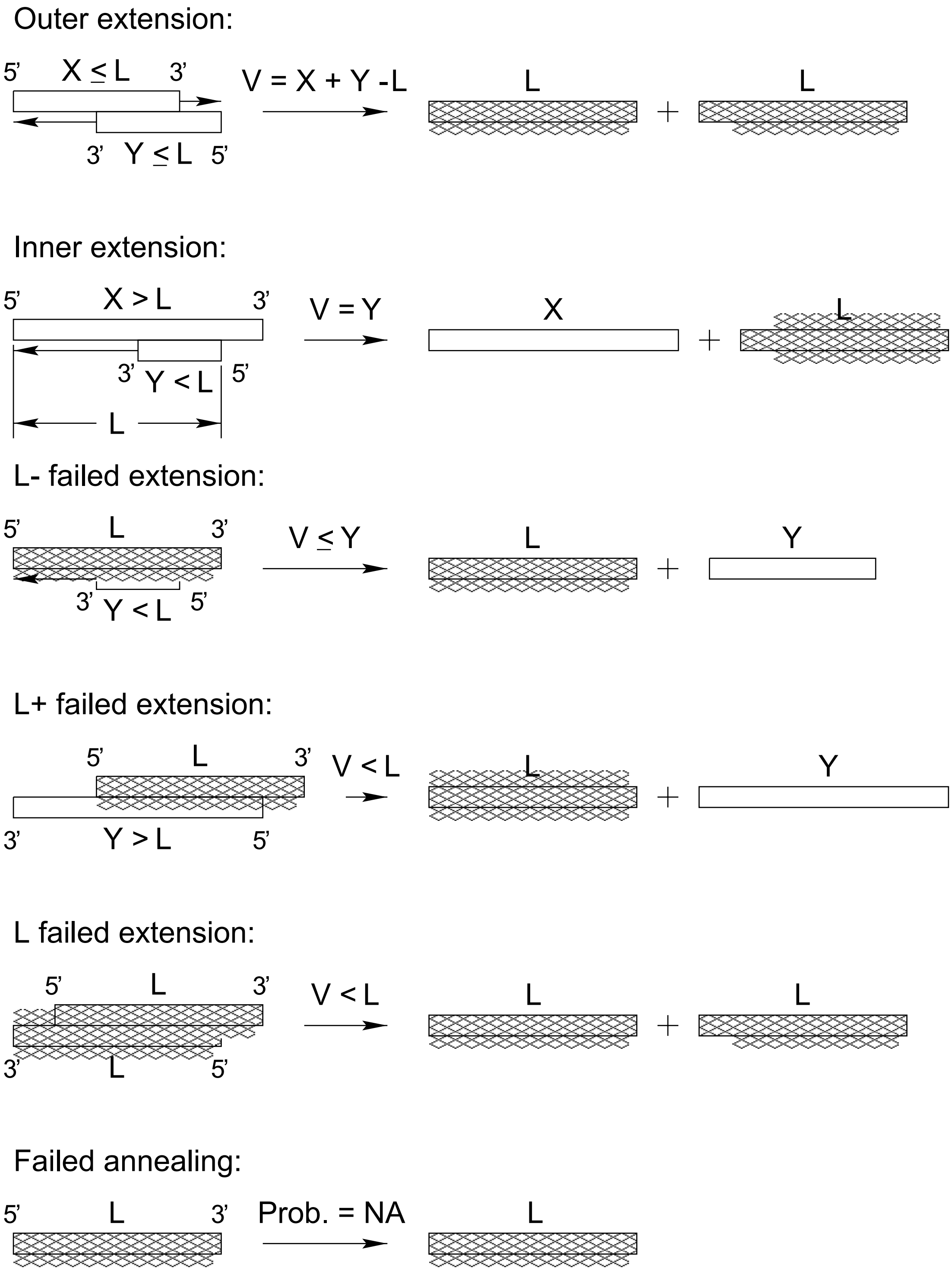


Figure 2.11: The six pathways for producing a fragment of length *L* by extension, failed extension and failed annealing.

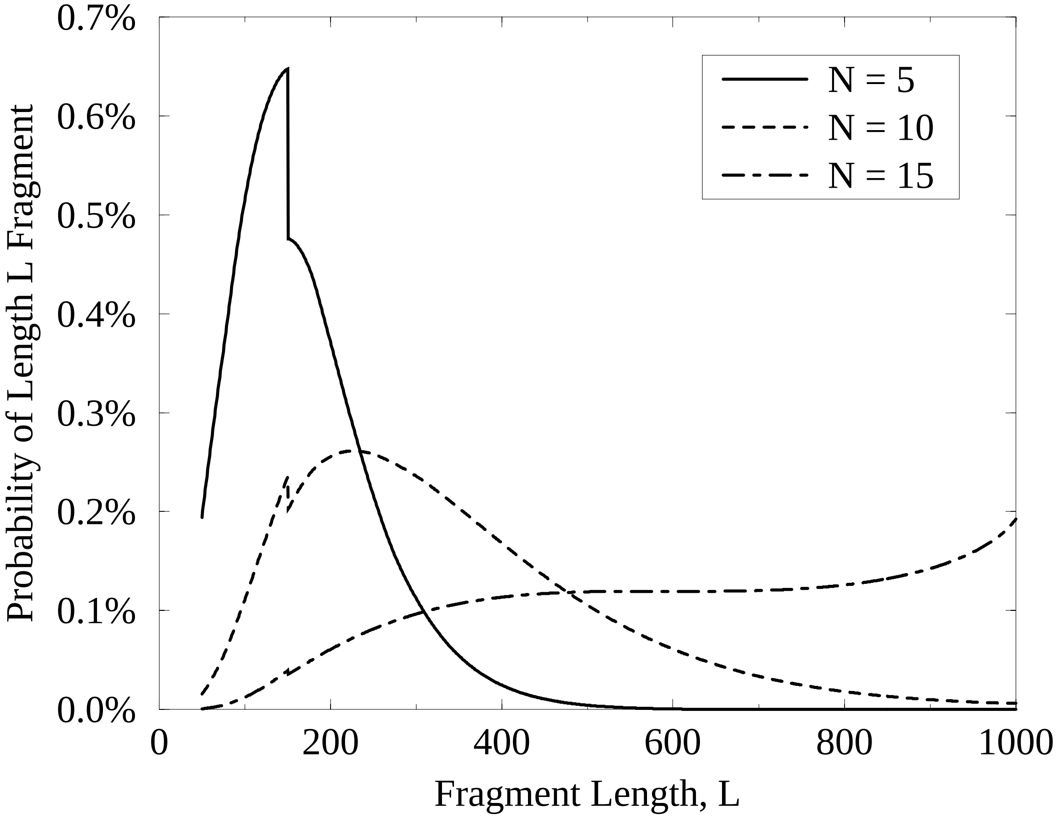


Figure 2.12: Fragment size distributions after N = 5, 10 and 15 shuffling cycles of a random fragment pool (L_1 = 50, L_2 = 100) of a 1000 bp gene (NA = 50%, α = -0.5, V_{min} = 15).

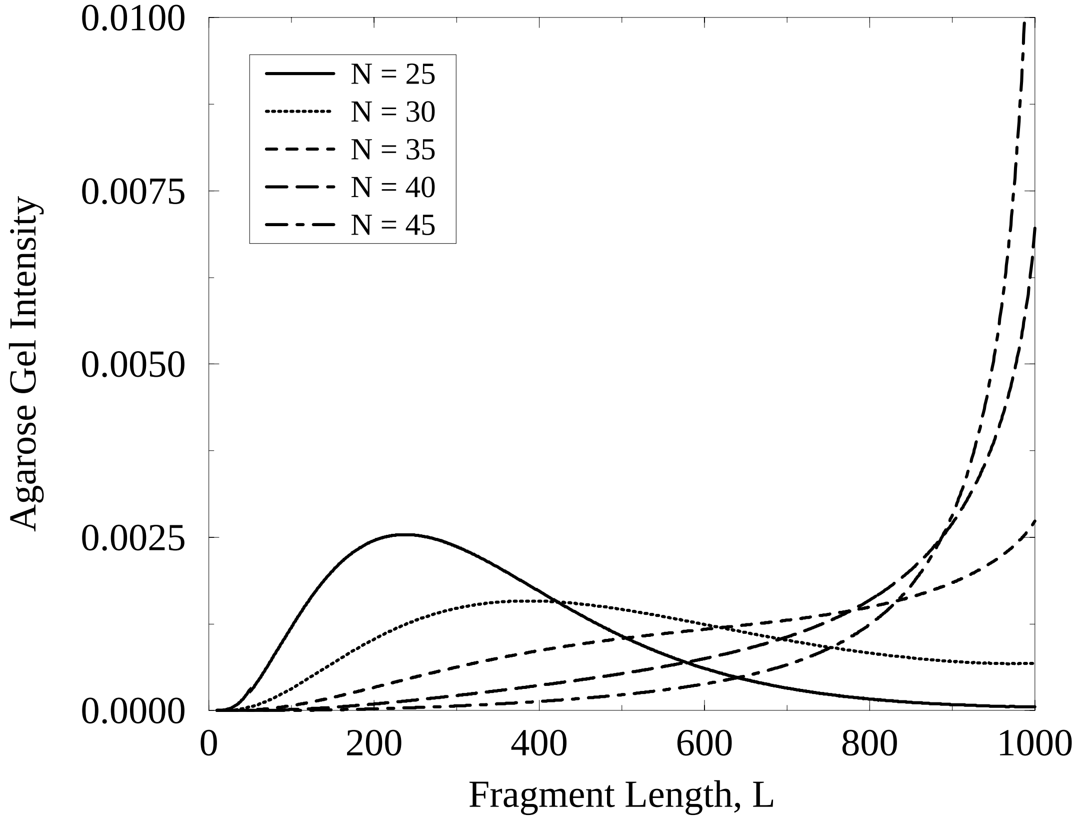
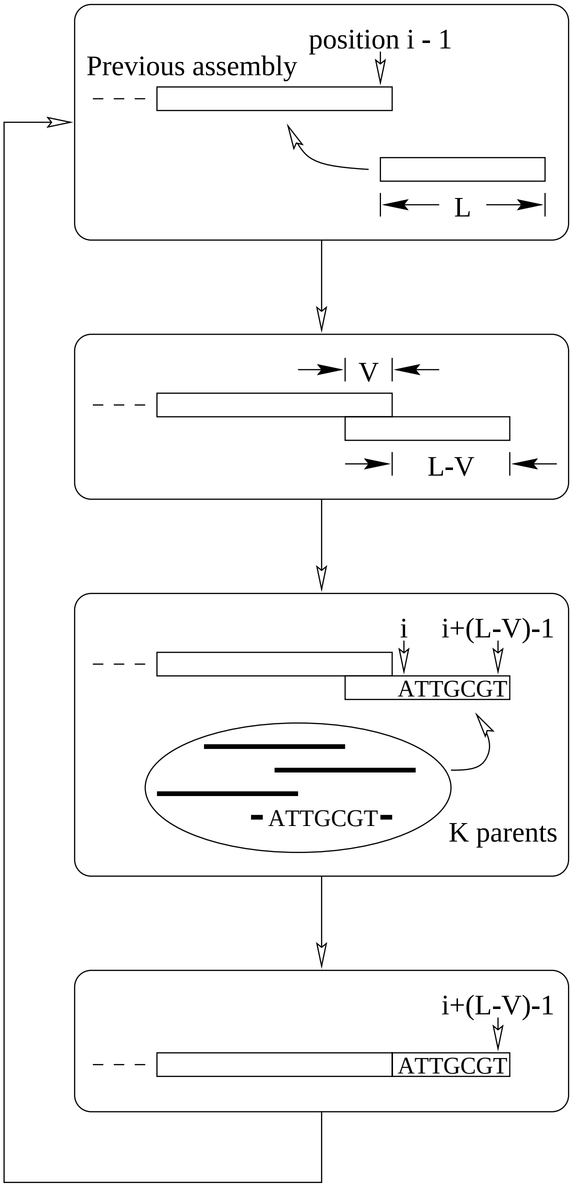


Figure 2.13: Fragment size distributions after *N* = 25, 30, 35, 40 and 45 shuffling cycles of a random fragment pool ($L_1 = 10$, $L_2 = 50$) of a 1000 bp gene ($NA = 70\%$, $\alpha = -0.5$, $V_{min} = 5$).



Step 1: Add a fragment of length L
with probability Q_L^0

Step 2: An overlap V occurs
with probability A(L-V,L)

Step 3: Choose a parent that matches
from positions i to i+(L-V)-1
with probability $\Delta_{i,i+(L-V)-1}$ / K

Step 4: Produce the rest of the sequence
from position i+(L-V) onward
with probability $P_{i+(L-V)}$

Figure 2.14: Four steps of the annealing process as described in the Target Sequence Matching model.

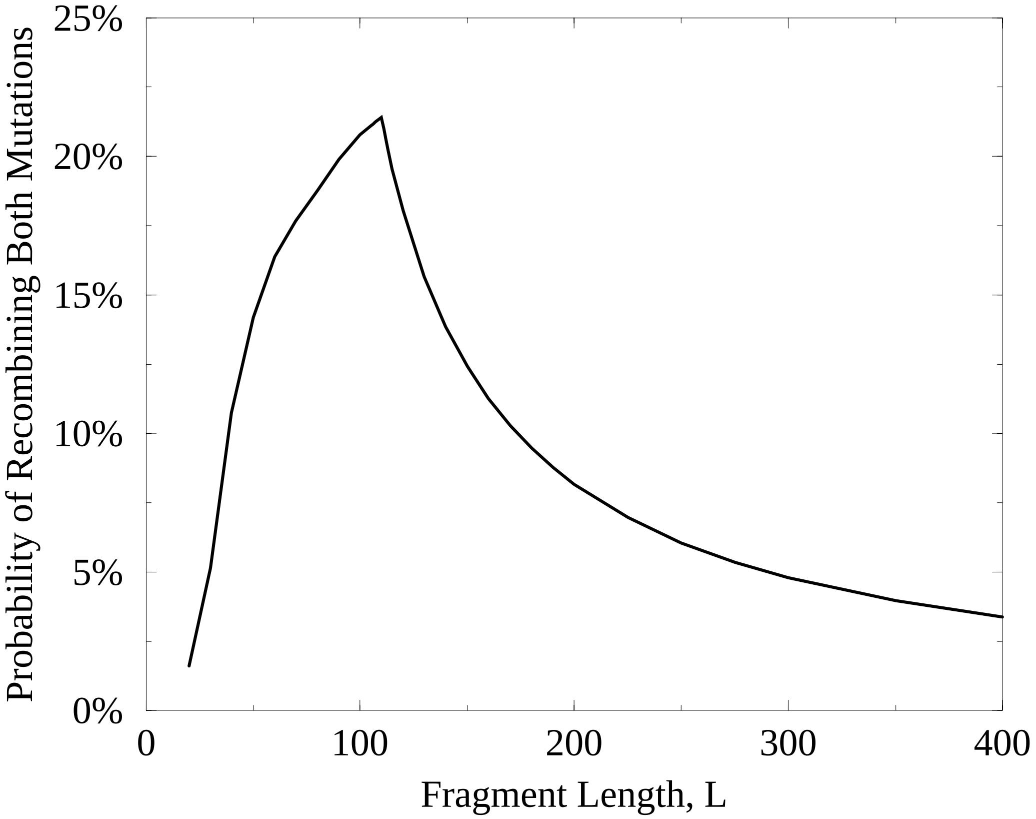
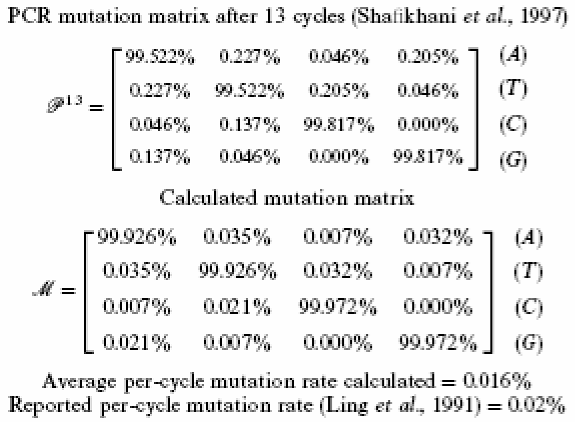


Figure 2.15: Probability of recombining two markers 75 bp apart as a function of the fragment length *L*.

Table 2.1: An example of mutation matrix calculation given reported mutation bias for zero Mn^{+2} concentration.



PCR mutation matrix after 13 cycles (Shafikhani *et al.*, 1997)

$$\mathscr{P}^{13} = \begin{bmatrix} 99.522\% & 0.227\% & 0.046\% & 0.205\% \\ 0.227\% & 99.522\% & 0.205\% & 0.046\% \\ 0.046\% & 0.137\% & 99.817\% & 0.000\% \\ 0.137\% & 0.046\% & 0.000\% & 99.817\% \end{bmatrix} \begin{array}{c} (A) \\ (T) \\ (C) \\ (G) \end{array}$$

Calculated mutation matrix

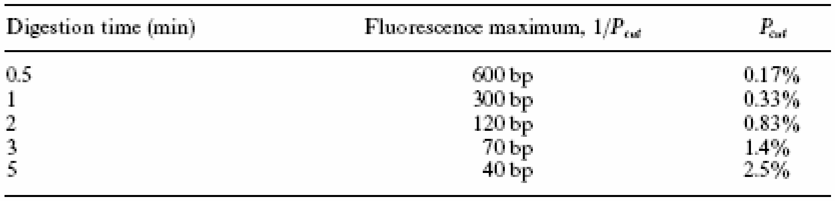
$$\mathscr{M} = \begin{bmatrix} 99.926\% & 0.035\% & 0.007\% & 0.032\% \\ 0.035\% & 99.926\% & 0.032\% & 0.007\% \\ 0.007\% & 0.021\% & 99.972\% & 0.000\% \\ 0.021\% & 0.007\% & 0.000\% & 99.972\% \end{bmatrix} \begin{array}{c} (A) \\ (T) \\ (C) \\ (G) \end{array}$$

Average per-cycle mutation rate calculated = 0.016%
Reported per-cycle mutation rate (Ling *et al.*, 1991) = 0.02%

Table 2.2: Comparison of discrete model vs. exponential approximation for fragment size probability calculation.

P_{cut}	Q^{0}_{100}, discrete model	Q^{0}_{100}, exponential approximation
10^{-4}	0.00990%	0.00990%
10^{-3}	0.0906%	0.0905%
10^{-2}	0.370%	0.368%
10^{-1}	0.000295%	0.000454%

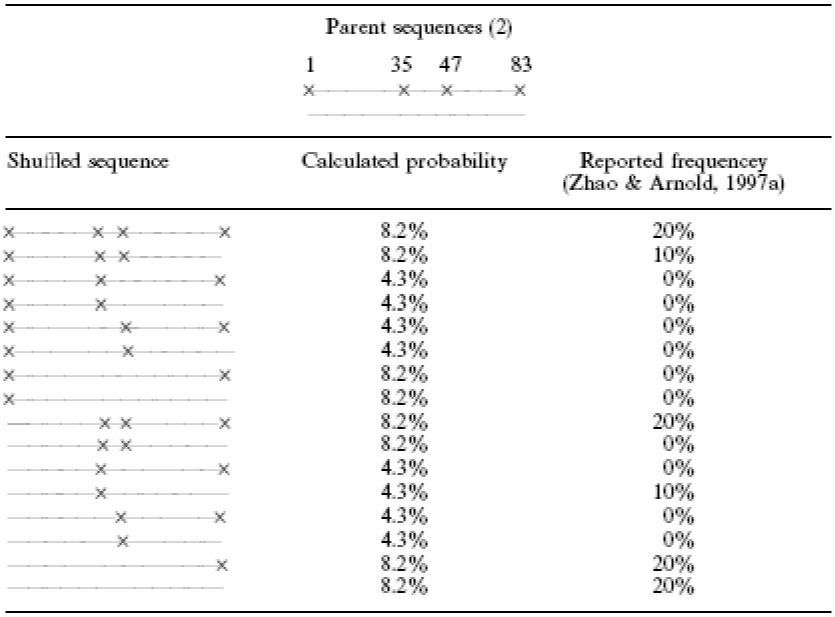
Table 2.3: Random fragmentation reaction progress.



Digestion time (min)	Fluorescence maximum, $1/P_{cut}$	P_{cut}
0.5	600 bp	0.17%
1	300 bp	0.33%
2	120 bp	0.83%
3	70 bp	1.4%
5	40 bp	2.5%

Table 2.4: DNA shuffling calculations for $L_1 = 30$,

$L_2 = 50$, $P_{cut} = 0.83\%$, and $V_{min} = 15$.



Parent sequences (2)

1 35 47 83

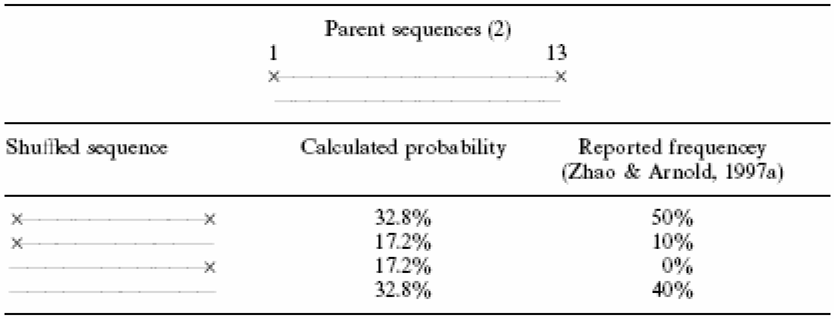
×———×—×———×

————————

Shuffled sequence	Calculated probability	Reported frequencey (Zhao & Arnold, 1997a)
×———×—×———×	8.2%	20%
×———×—×———	8.2%	10%
×———×————×	4.3%	0%
×———×————	4.3%	0%
×————×———×	4.3%	0%
×————×———	4.3%	0%
×—————————×	8.2%	0%
×—————————	8.2%	0%
————×—×———×	8.2%	20%
————×—×———	8.2%	0%
————×————×	4.3%	0%
————×————	4.3%	10%
—————×———×	4.3%	0%
—————×———	4.3%	0%
——————————×	8.2%	20%
——————————	8.2%	20%

Table 2.5: DNA shuffling calculations for $L_1 = 30$,

$L_2 = 50$, $P_{cut} = 0.83\%$, and $V_{min} = 15$.



Parent sequences (2)
1 13
×——————————×
————————————

Shuffled sequence	Calculated probability	Reported frequencey (Zhao & Arnold, 1997a)
×—————————×	32.8%	50%
×——————————	17.2%	10%
——————————×	17.2%	0%
———————————	32.8%	40%

Chapter 3: Modeling Crossover Generation in DNA Shuffling (*e*Shuffle)

Section 3.1: Background

DNA shuffling [1, 2], along with its variants, is one of the earliest and most commonly used DNA recombination protocols. It consists of random fragmentation of parent nucleotide sequences with DNaseI and subsequent fragment reassembly through primerless PCR. Library diversity is generated during reassembly when two fragments originating from different parental sequences anneal and subsequently extend. This gives rise to a *crossover*, the junction point in a reassembled sequence where a template switch takes place from one parental sequence to another. The key advantage of DNA shuffling is that many parental sequences can be recombined simultaneously (*i.e.*, family DNA shuffling [3, 4]) generating multiple crossovers per reassembled sequence. However, crossovers tend to aggregate in regions of high sequence identity due to the annealing-based reassembly. In this chapter, a computational framework named ***e***Shuffle is presented for predicting the number, type, and location of crossovers in full-length sequences generated by DNA shuffling [5, 6].

Section 3.2: Modeling of Annealing Events

During annealing, fragments compete to anneal with a growing template. This competition is quantified by utilizing equilibrium thermodynamics to infer (i) what fraction of these fragments will anneal at a given temperature, (ii) how these annealing events will be distributed between those involving high or low overlap lengths, and (iii) what portion of these annealing events will involve mismatches. An annealing event between fragments originating from the same parental sequence yields a *homoduplex* (assuming in-frame annealing), whereas the annealing of two fragments from different parents gives a *heteroduplex*. Mismatches at exactly the 3′ end will lead to less efficient extension and thus are not counted.

The thermodynamics of duplex formation can be analyzed using nearest-neighbor parameters that describe the enthalpic and entropic contributions of specific nucleotide pairs in the overlapping region [7-12]. The change in free energy ΔG associated with an annealing event can be approximated by summing the free energy gains associated with all 2-nt matches and the free energy penalties associated with the mismatches. Additional corrections are also included for the duplex initiation free energy cost, salt concentration and dangling end stabilization [13]. Enthalpic and entropic parameters at 37°C for the contribution of pairs of matches and mismatches are summarized in a table found in the supplemental material of reference [6].

Given this free energy predictive capability, the extent of duplex formation can be tracked at different temperatures. Specifically, consider the reaction associated with the annealing of a fragment *F* with a template *A* forming a duplex *AF*.

$$A + F \rightleftarrows AF$$

Assuming equilibrium, the equilibrium constant *K(T)* links the mole fractions of the template, fragment and duplex at different temperatures:

$$K(T) = \exp\left(-\frac{\Delta G(T)}{RT} \right) = \frac{x_{AF}}{x_A x_F}.$$

Here x denotes mole fractions and ^{0} denotes initial values of the species in the reaction mixture so that $x_A = x_A^0 - x_{AF}$ and $x_F = x_F^0 - x_{AF}$. Let *a(T)* be the annealing curve defined as the fraction of templates that have annealed at temperature *T*, ($a(T) = x_{AF} / x_A^0 = 1 - x_A / x_A^0$). Upon rearrangement these equations can be solved for x_F, x_A, x_{AF} and *a(T)*. The temperature at which half of the templates have hybridized to form duplexes (*i.e.*, *a(T)* = 1/2) is defined as the *melting temperature* T_m. Comparisons of the predictions obtained with the described free energy modeling framework against those found by an empirical formula commonly used for hybridization experiments [14] are in good agreement (see Table 3.1). Plots of *a(T)* versus *T* reveal that there is a relatively narrow temperature range, centered around T_m, where the majority of annealing events take place (sigmoidal curve). In general, longer overlaps imply higher melting temperatures while shorter overlaps, mismatches and low GC content depress T_m.

During the annealing step of DNA shuffling, not a single, but many different fragments with varying lengths, overlaps and mismatches are competing for a given template.

$$A + F_{mv} \rightleftharpoons AF_{mv}$$

Here *m* refers to a fragment originating from parental sequence *m* and *v* implies an overlap length of *v* nucleotides with the template upon annealing. After adjusting the expression for *a(T)* to reflect the multiplicity of annealing choices and resolving the system of equations the temperature-dependent selectivity

$$s_{mv}(T) = x_{AFmv} / \left(\sum_{m',v'} x_{Fm'v'} \right)$$

for a particular fragment and overlap choice *mv* is estimated. The presence of multiple fragment and overlap choices "spreads" the melting curve over a wider range of temperatures implying that annealing events occur over the entire temperature range (typically 94-55°C). The free energy differences between annealing choices and relative fragment concentrations determine which annealing choice dominates at a given temperature. For instance, at high temperatures fragments with large overlaps that match perfectly with the template dominate all other ones because of the large enthalpic gains that they provide on annealing. As the temperature is lowered, the melting temperatures of fragments with progressively smaller overlaps and even one or two mismatches is reached, resulting in selectivities that are much more uniform.

Because annealing selectivities are temperature dependent, duplex formation must be assessed cumulatively over the entire annealing temperature range. To this end, the annealing step is modeled as a sequence of pseudo-equilibrium states progressively contributing duplexes as the temperature is lowered from 94°C to 55°C. Mathematically, this implies integration of the temperature-dependent selectivities $s_{mv}(T)$ times the annealing rate *da(T)/dT* over the annealing temperature schedule.

$$S_{mv} = \int_{T_{anneal}}^{T_{denature}} s_{mv}(T) \frac{da(T)}{dT} dT$$

Given a pool of fragments competing for a template and an annealing temperature schedule, S_{mv} quantifies the overall annealing selectivities. The effect of the length of overlap and number/severity of mismatches is illustrated in Figure 3.1. The first plot (Figure 3.1a) addresses the case when there are no mismatches. It clearly shows that there is strong preference towards annealing events involving the maximum overlap. However, a non-negligible portion of annealing events involves shorter overlaps. The second plot (Figure 3.1b) considers the effect of the number and type of mismatches on annealing selectivities for a given overlap length. Although the great majority of annealing events involve no mismatches, some mismatch-bearing annealing events also occur that cannot be ignored. Note that, in the present implementation, the type of a mismatch affects its selectivity whereas its distance from the 3′ end does not. Next, the individual annealing statistics are utilized to infer crossover generation in the reassembled sequences.

Section 3.3: Fragment Reassembly

The reassembly process is modeled as a successive sequence of annealing events. Specifically, the selectivity of an annealing event is assumed to depend only on the identity of the fragment added immediately before. For clarity of presentation, only fragments of a unique length L will be used in the reassembly analysis. Nevertheless, fragments with varying lengths can be incorporated in a straightforward manner as described [15, 16].

The key idea of the reassembly procedure is to postulate a set of recursive relations that resolve the question of what is the probability Π^x that a full-length reassembled sequence of B nucleotides has x crossovers. To this end, we define P_{ik}^{x} denoting the probability that reassembly from position i to the end B of the DNA sequence will yield exactly x crossovers, given that the fragment ending at position i - 1 originated from parental sequence k. The selectivities S_{mv}, defined earlier, can then be calculated for different annealing choices. When a fragment from parental sequence m anneals with a fragment from sequence k either a homoduplex ($m = k$) or heteroduplex (m

$\neq k$) is formed. Homoduplex formation implies that no crossover is generated and the recursion must still track x crossovers over the remainder of the reassembly. However, heteroduplex formation implies that only (x - 1) remaining crossovers must be subsequently tracked. The annealing of a fragment of length L with an overlap v implies the addition of L - v nucleotides, extending the template to position (i - 1) + (L - v). This position becomes the new reassembly point completing the recursion. Summation over all parental sequences m and overlap lengths v encompasses all possible reassembly pathways.

$$P_{ik}^{x} = \sum_{v=1}^{L-1} S_{kv} P_{i+L-v,k}^{x} + \sum_{m \neq k} \sum_{v=1}^{L-1} S_{mv} P_{i+L-v,m}^{x-1}, \forall\, x > 0, \forall\, i > L, \text{and } \forall\, k.$$

Resolution of this recursion requires boundary conditions at the start and end of the gene or gene fragment under consideration. At the onset of reassembly, the initial fragment covers the range $i = 1$ to $i = L$ implying that subsequent annealing events add nucleotides starting from position $i = L + 1$. This initial fragment comes from parent m with probability equal to the relative concentration C_m of parent m in the reaction mixture. This implies that the probability Π^x that the reassembled sequences contains x crossovers is the parent relative concentration averaged probability of having x crossovers past position L + 1.

$$\Pi^{x} = \sum_{m} C_m P_{L+1,m}^{x}, \forall\, x = 0, 1, \ldots$$

The boundary conditions for the end position B ensure that no crossovers occur beyond position $i = B$.

$$P_{ik}^{0} = 1, \forall\, i > B \text{ and } \forall\, k$$

$$P_{ik}^{x} = 0, \forall\, x > 0, \forall\, i > B, \text{and } \forall\, k.$$

Because reassembly is a bidirectional process, the reassembly algorithm is also executed in the reverse direction with the complementary DNA sequences and the results are combined. A flowchart outlining the proposed reassembly procedure is shown in Figure 3.2.

Interestingly, the original application of the reassembly algorithm overestimated the total number of crossovers, especially for shuffling sequences that share very high sequence identity. Closer inspection revealed that this was due to the formation of heteroduplexes with fragments involving perfect sequence identity with the growing template. Even though they are indeed crossovers, according to the formal crossover definition, they are completely undetectable experimentally and more importantly they do not contribute any diversity. Therefore, the term *silent crossover* was proposed for them, and the reassembly algorithm was revised to exclude them. Specifically, if the annealing of a fragment from parent m to a growing template ending with a fragment from parent k is equivalent to the continuation of the template with nucleotides from parent k, no crossover is counted.

The proposed reassembly procedure allows the estimation of the fraction of the reassembled sequences containing $x = 0, 1, \ldots$ crossovers. By redefining what constitutes a desirable crossover, different types of crossovers can be assessed separately. For example, in the family DNA shuffling of sequences A, B and C the statistics of all six possible types of crossovers AB, BA, AC, CA, BC and CB can be tracked independently. In addition, one could even track homoduplex extension events such as AA, BB or CC. Next the statistics of the distribution of these crossovers along the reassembled sequences is examined.

Specifically, the question addressed is what is the probability that a given position i in a reassembled sequence is the site of a crossover (*i.e.*, end point of a heteroduplex annealing event). This probability depends on the parent origin of the fragment ending at position $i - 1$. Thus, the probability that a fragment from parent k ends exactly at position $i - 1$ is defined as T_{ik}. A recursion is then established in a similar manner as before. A fragment from parent m ends at position $i - 1$ if and only if it was added to a fragment from parent k ending at position $i - L + v$ with an overlap v. The probability for this particular duplex formation event can be quantified by multiplying the selectivity S_{mv}

times the probability $T_{i-L+v,k}$ that the template is positioned appropriately.

$$T_{im} = \sum_{k} \sum_{v=1}^{L-1} T_{i-L+v,k} S_{mv},\ \forall\, i > L+1 \text{ and } \forall\, m.$$

Boundary conditions ensure that the first nucleotide added to the original fragment comes from a parental sequence k with a probability proportional to its relative concentration. Furthermore, no fragment may end before position $i = L$.

$$T_{L+1,k} = C_k,\ \forall\, k$$

$$T_{ik} = 0,\ \forall\, i \le L \text{ and } \forall\, k.$$

Once the probability T_{ik} that a particular type of template k ends immediately before position i is known, it can be multiplied by the selectivity of a crossover-generating annealing event S_{mv} and summed over all possible annealing choices to infer the probability P_i^{cross} that position i is the site of a crossover.

$$P_i^{cross} = \sum_{k} \sum_{v=1}^{L-1} \sum_{m \ne k} T_{ik} S_{mv}.$$

Again, by tailoring the definition of a crossover, the distribution of different types of crossovers (*i.e.*, AB, BC or AC) along the sequence can be assessed separately. A consistency check reveals that the average number of crossovers calculated based on the probabilities P_i^{cross} quantifying crossover density along the DNA sequence, ($\sum_i P_i^{cross}$), is identical to the one obtained based on the crossover number distribution calculated earlier ($\sum_x x\Pi^x$). Given this versatile algorithmic framework the statistics of any type of crossover can be quantified both in terms of variability among the reassembled sequences and along the length of the gene. Predictions obtained based on the above described analysis are next contrasted against experimental data from DNA shuffling experiments reported in the literature.

Section 3.4: Comparisons with Experimental Results

Although directed evolution studies are being reported in the literature with an accelerating pace, only a few studies report DNA sequencing results for naive (*i.e.*, unselected) DNA libraries. Partial DNA sequencing results allowing for the estimation of the number of crossovers in a small subset of the reassembled sequences are found for the following five studies. Computer simulation of DNA shuffling of these systems provides the basis for the comparisons. Every effort was made to ensure that the fragment length, annealing temperature, and salt and DNA concentrations matched the ones in the experimental study. When no information was provided, default values from the original DNA shuffling protocol [1, 2] were adopted.

The first system considered is two 465-bp IL-1β genes (human and murine) [2] with a sequence identity of only 75%. An extremely low annealing temperature of 25°C was used to boost the generation of crossovers. Nine colonies were sequenced for a total of 17 crossovers implying an average of 1.9 per sequence. Simulation results are in close agreement with experiment, predicting an average of 1.5 crossovers.

The next system involved the family DNA shuffling of four class C cephalosporinase genes, 1.2 kb in length with pairwise sequence identities ranging from 58 to 82% [3]. It was reported that neither of the two active clones sequenced contained any fragments from the *Yersinia enterocolitica* gene (third gene). The question is whether this occurred because fragments originating from this gene have a detrimental effect on activity or simply because pieces from this gene are disproportionately misrepresented in the naive library due to the lack of sufficiently long stretches of near-perfect sequence identity with the other three genes. The average sequence identities of each one of the four genes against the remaining three are 70%, 70%, 65%, and 59% respectively. Simulation results predict that 36% of the naive sequences contain at least one crossover. The fraction of crossover bearing sequences containing at least one piece from each one of the four genes is 85%, 95%, 7%, and 19% respectively. This indicates that *Y. enterocolitica* (third one) is by far the least even though it is not the one with the lowest sequence identity. This suggests a possible explanation for the absence of any piece of *Y. enterocolitica* in the most active clones.

The next system studied involved two genes for glycinamide ribonucleotide transformylase, *Escherichia coli* (*purN*) and human (*hGART*) [17] with a very low sequence identity of 49%. Here the following staggered portions of the two genes were shuffled (*E. coli* positions 1-434) and (human positions 164-611) implying that crossovers could only be formed in the 271-bp shared region (47% sequence identity). This arrangement requires that all reassembled genes of full length start with the *E. coli* gene and end with the human gene yielding an odd numbers of crossovers. In the experimental study only single crossover clones were observed of 10 sequenced clones. This is consistent with the simulation prediction that the ratio of the number of reassembled sequences with three or more crossovers to the number of sequences with a single crossover is less than 10^{-9}. A system with a relatively high sequence identity is analyzed next. It involves the DNA shuffling of two biphenyl oxygenases sharing a sequence identity of 87% [18]. For this system, an average of 3.3 crossovers per sequence is observed experimentally (six sequenced clones), whereas the simulation suggests a slightly smaller average of 2.8.

The last study is the only one where the simulation results deviated from the experimentally observed crossover averages. It involved the DNA shuffling of a 1.3-kb gene for wild-type subtilisin E and that of a clone (1E2A) differing by only 10 point mutations [19]. Slightly larger fragments in the range of 20 to 50 bases were used in place of the default fragment length range of 10 to 50 bases. One would expect that a large average number of crossovers would be generated in this system because only 10 point mutations are present implying a sequence identity of 99.2%. However, this is not observed experimentally as only an average of 1.9 crossovers per sequence is reported [19]. The simulation results on the other hand are consistent with the intuitive expectation, predicting an average of 3.6 crossovers per reassembled sequence. The randomly chosen sequences may not have been representative of the entire DNA library. For instance, recombinations between mutations at positions 520 and 732 in clone 1E2A must be occurring independently because the stretch of perfect identity is much wider than even the maximum fragment size. However, a crossover occurs in only 10% of the reported sequences instead of the 50% frequency expected for independent reassembly. With the exception of this last example, simulation predictions are in good agreement with the published experimental results without adjustable model parameters.

Section 3.5: Subtilase Case Study

Subtilases are serine proteases [20] extensively engineered with directed evolution experiments [4, 21, 22]. A set of 12 subtilases including subtilisins E, BPN′, Carlsberg, 147, ALP I, PB92, and Sendai; serine Proteases C and D; proteinases K and R; and thermitase is next considered to highlight the effect of fragmentation length, annealing temperature, sequence identity and number of shuffled sequences on the number, type and distribution of crossovers. We chose to mirror recent subtilase directed evolution experiments [4] by analyzing the shuffling of only a 500-bp subgenomic region. The average pairwise sequence identity is 58% ranging from 44% to 90%. First a high sequence identity 80% pair (subtilisin E, subtilisin BPN′) is considered.

As shown in Figure 3.3a, for a fragmentation length of $L = 50$ bases, 44% of the reassembled sequences involve no crossovers, 37% one crossover, 15% two crossovers and diminishing percentages for sequences with more than two crossovers. As the fragment length is reduced, a nonlinear increase of crossovers is observed. This nonlinear increase in the average number of crossovers as a function of L is more clearly depicted in Figure 3.3b. Interestingly, the same plot (dashed line) reveals a dramatic increase of silent crossovers for very small fragment lengths (*i.e.*, $L \leq 20$). Figure 3.4 illustrates the distribution of crossovers superimposed against the sequence identity along the sequence. It shows that crossovers are preferentially aggregated in regions of near perfect sequence identity forming a characteristic double peak. The double peak implies that annealing events make full use of the available sequence identity giving rise to two distinct double peaks at the two flanking positions of the sequence identity stretch. Larger fragments afford a wider range of overlaps flattening the two peaks whereas smaller fragments are capable of generating crossovers in relatively narrow regions of high sequence identity. However, in DNA shuffling not a single fragmentation length L is employed but rather a distribution of fragment sizes, typically in the range of 10 to 50 bases, with a size distribution described by an exponentially decaying function [15, 16]. When a range of fragment sizes is employed for the above example, computational results reveal that the crossover statistics are almost identical with the case of utilizing a single “effective” fragment size which for the 10- to 50-bases range is 25 bases.

Next the effect of annealing temperature on crossover generation is studied. What is found is that two underlying mechanisms exist with which annealing temperature affects the crossover statistics (see Figure 3.5). Specifically, for medium to large fragments, lower annealing temperatures imply that the melting temperatures of more annealing choices containing mismatches (*i.e.*, heteroduplexes) are encountered yielding more crossovers upon extension. However, for very small fragments at high temperatures the entropic contribution to the free energy of annealing dominates, blurring the distinction between homoduplexes and heteroduplexes, causing a sharp increase in the total number of crossovers. Clearly, as in the case of fragment length, the annealing temperature cannot be arbitrarily reduced because at some point fragments cease to exhibit strong affinity for annealing in-frame, and out-of-frame additions start to overwhelm the reassembly process.

The limits of DNA shuffling are explored by choosing the low sequence identity pair (serine protease D, proteinase K) which has a 46% sequence identity. As expected, very few crossovers are predicted (see Table 3.2) with only a single narrow region at the end of the sequence coinciding with a short stretch of high sequence identity. Subsequently, the high sequence identity pair (subtilisin E, subtilisin BPN′) is shuffled *in silico* together with the low sequence identity pair (serine protease D, proteinase K) in equal ratios. The key question is whether the low identity pair will simply dilute the fragment pool that can form heteroduplexes depressing crossover generation by a factor of 2, or if synergism in the reassembly will dominate. Even though the average pairwise sequence identity for the four subtilase system is as low as 58%, a comparable number of crossovers with the (subtilisin E, subtilisin BPN′) single pair case is found (see Table 3.2). This implies that synergistic reassembly is taking place alluding to the contribution of "bridging" crossovers by the low sequence identity pairs. The full power of synergistic reassembly is revealed when all 12 subtilases are included providing a computational verification of what is seen experimentally with family DNA shuffling, especially for smaller fragments. Even though the average pairwise sequence identity is only 58% at least as many crossovers are generated (see Table 3.2) as for the high sequence identity 80% pair. More importantly these crossovers span the entire sequence range (see Figure 3.6). Admittedly though, the distribution is still multimodal with peaks tracking the location of high sequence identity, a signature of the annealing-based reassembly characteristic of DNA shuffling. In the next section, we examine the SCRATCHY protocol, which is capable of generating crossovers in nonhomologous regions and reducing the bias seen in Figs. 4 and 6.

Section 3.6: Summary

eShuffle provided for the first time a quantitative framework for the *in silico* exploration of many "what if" scenarios in terms of fragmentation length, annealing temperature and parental choices in the context of DNA shuffling. Comparisons of the ***e***Shuffle predictions against experimental data revealed good agreement, particularly in light of the fact that there are no adjustable parameters in the modeling framework. The only parameters are the free energy contributions used unchanged from literature sources [11]. Therefore, no reparameterization is needed when either the experimental conditions or the sequences to be shuffled change, providing a versatile framework for comparing different protocol choices and setups. In the context of family DNA shuffling [3, 4], the ***e***Shuffle program enabled the estimation of the relative contribution of fragments from different parental sequences to the combinatorial DNA library. Results revealed that the pairwise sequence identities between the parental sequences do not always explain the observed parental crossover frequencies in the libraries. ***e***Shuffle also led to the quantification of synergistic reassembly in family DNA shuffling and the elicitation of the presence of the swapping of identical fragments between high sequence identity parental pairs (silent crossovers).

Section 3.7: References

1. Stemmer, W.P.C. (1994), "Rapid evolution of a protein in vitro by DNA shuffling," *Nature (London)* **370**: 389-391.
2. Stemmer, W.P.C. (1994), "DNA shuffling by random fragmentation and reassembly: in vitro recombination for molecular evolution," *Proc Natl Acad Sci USA* **91**: 10747-10751.
3. Crameri, A., Raillard, S., Bermudez, E. & Stemmer, W.P.C. (1998), "DNA shuffling of a family of genes from diverse species accelerates directed evolution," *Nature (London)* **391**: 288-291.
4. Ness, J.E., Welch, M., Giver, L., Bueno, M., Cherry, J.R., Borchert, T.V., Stemmer, W.P.C. & Minshull, L. (1999), "DNA shuffling of subgenomic sequences of subtilisin," *Nature Biotechnol* **17**: 893-896.
5. Moore, G.L. & Maranas, C.D. (2002), "Predicting out-of-sequence reassembly in DNA shuffling," *J Theor Biol* **219**(1): 9-17.
6. Moore, G.L., Maranas, C.D., Lutz, S. & Benkovic, S.J. (2001), "Predicting crossover generation in DNA shuffling," *Proc Natl Acad Sci USA* **98**(6): 3226-3231.
7. Allawi, H.T. & SantaLucia, J. (1998), "Nearest-Neighbor Thermodynamics of Internal A-C Mismatches in DNA: Sequence Dependence and pH Effects," *Biochem* **37**: 9435-9444.
8. Allawi, H.T. & SantaLucia, J. (1998), "Nearest Neighbor Thermodynamic Parameters for Internal G-A Mismatches in DNA," *Biochem* **37**: 2170-2179.
9. Allawi, H.T. & SantaLucia, J. (1998), "Thermodynamics of internal C-T mismatches in DNA," *Nucleic Acids Res* **26**: 2694-2701.
10. Allawi, H.T. & SantaLucia, J. (1997), "Thermodynamics and NMR of Internal G-T Mismatches in DNA," *Biochem* **36**: 10581-10594.

11. SantaLucia, J. (1998), "A unified view of polymer, dumbbell, and oligonucleotide DNA nearest neighbor thermodynamics," *Proc Natl Acad Sci USA* **95**: 1460-1465.

12. Peyret, N., Seneviratne, P.A., Allawi, H.T. & SantaLucia, J. (1999), "Nearest-Neighbor Thermodynamics and NMR of DNA Sequences with Internal A-A, C-C, G-G, and T-T Mismatches," *Biochem* **38**: 3468-3477.

13. Bommarito, S., Peyret, N. & SantaLucia, J. (2000), "Thermodynamic parameters for DNA sequences with dangling ends," *Nucleic Acids Res* **28**: 1929-1934.

14. Howley, P.M., Israel, M.F., Law, M. & Martin, M.A. (1979), "A rapid method for detecting and mapping homology between heterologous DNAs. Evaluation of polyomavirus genomes," *J Biol Chem* **254**: 4876-4883.

15. Moore, G.L. & Maranas, C.D. (2000), "Modeling DNA Mutation and Recombination for Directed Evolution Experiments," *J Theo Biol* **205**: 483-503.

16. Moore, G.L., Maranas, C.D., Gutshall, K.R. & Brenchley, J.E. (2000), "Modeling and Optimization of DNA Recombination," *Comp Chem Eng* **24**: 693-699.

17. Ostermeier, M., Shim, J.H. & Benkovic, S.J. (1999), "A combinatorial approach to hybrid enzymes independent of DNA homology," *Nature Biotechnol* **17**: 1205-1209.

18. Kumamaru, T., Suenaga, H., Mitsuoka, M., Watanabe, T. & Furukawa, K. (1998), "Enhanced degradation of polychlorinated biphenyls by directed evolution of biphenyl dioxygenase," *Nature Biotechnol* **16**: 663-666.

19. Zhao, H. & Arnold, F.H. (1997), "Functional and nonfunctional mutations distinguished by random recombination of homologous genes," *Proc Natl Acad Sci USA* **94**: 7997-8000.

20. Siezen, R.J. & Leunissen, J.A. (1997), "Subtilases: The superfamily of subtilisin-like serine proteases," *Protein Sci* **6**: 501-523.

21. Chen, K. & Arnold, F.H. (1991), "Enzyme Engineering for Nonaqueous Solvents: Random Mutagenesis to Enhance Activity of Subtilisin E in Polar Organic Media," *Bio/Technology* **9**: 1073-1077.

22. Chen, K., Robinson, A.C., Dam, M.E.V., Martinez, P., Economou, C. & Arnold, F.H. (1991), "Enzyme Engineering for Nonaqueous Solvents. II. Additive Effects of Mutations on the Stability and Activity of Subtilisin E in Polar Organic Media," *Biotechnol Prog* **7**: 125-129.

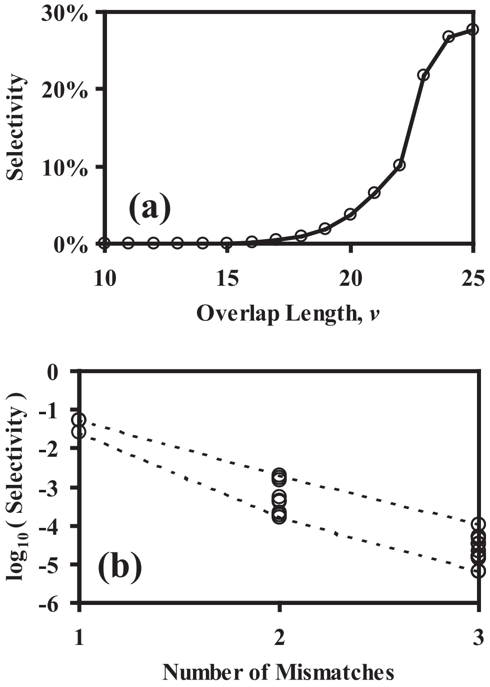


Figure 3.1: Selectivity versus overlap lengths (a) and selectivity for different degrees, types and locations of mismatches (b). Both charts utilize the subtilisin E gene, positions 760-784, and mismatches are evenly distributed in the overlapping region.

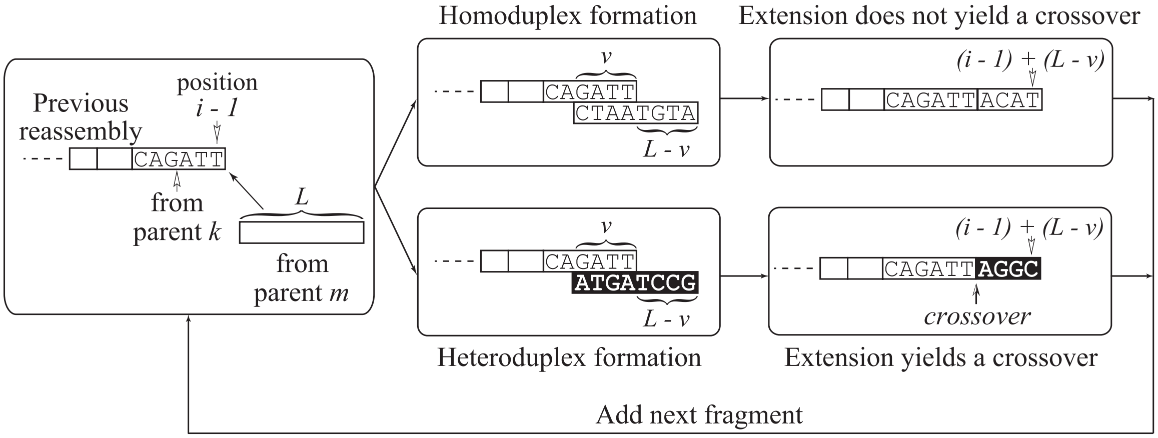


Figure 3.2: A flowchart of the ***e***Shuffle reassembly algorithm.

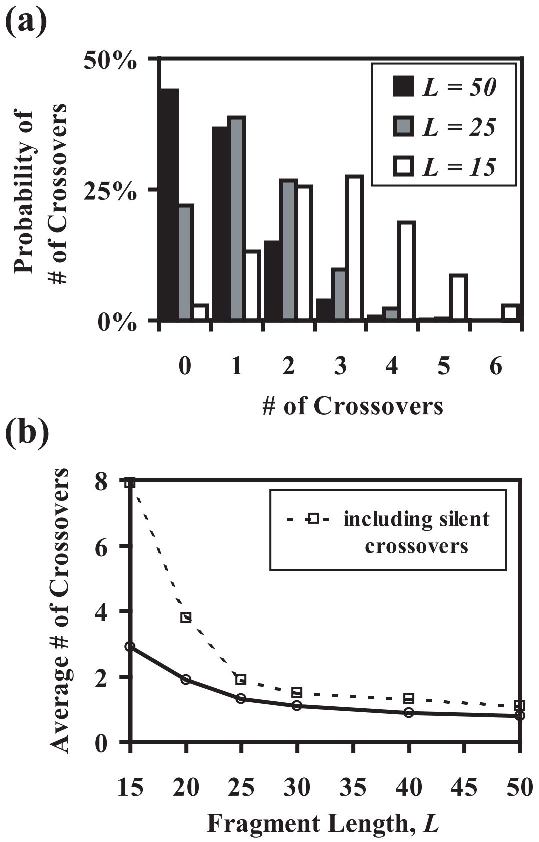


Figure 3.3: (a) Crossover number distribution for DNA shuffling of subtilisin E and subtilisin BPN′ for L = 15, 25 and 50 bases. (b) Average number of crossovers per sequence for the same system plotted versus fragment length in bases. The dotted line includes silent crossovers.

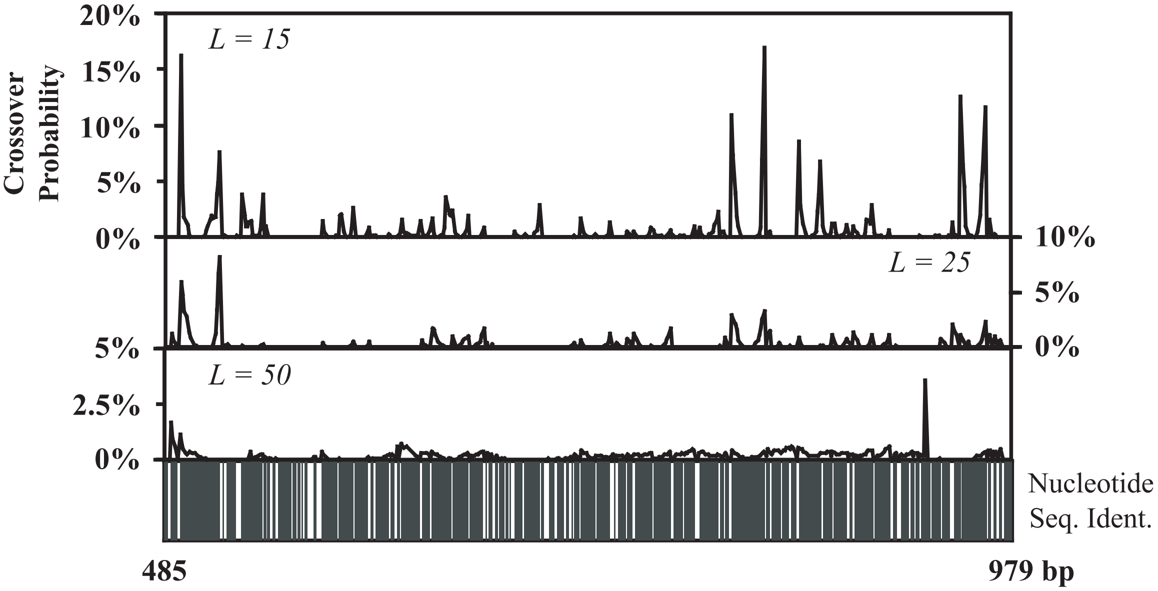


Figure 3.4: Probability of generating a crossover along the length of the sequence for the (subtilisin E, subtilisin BPN′) system for *L* = 15, 25 and 50 bases along the subregion 485-979. Black columns in the bottom strip chart denote identical nucleotides for both sequences, and white lines denote mismatches.

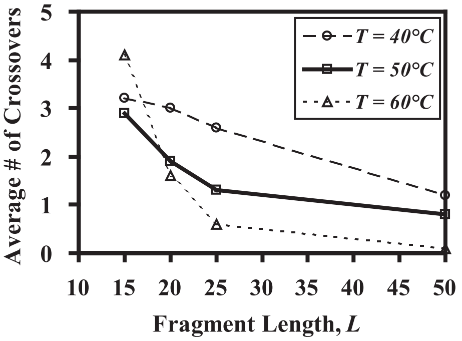


Figure 3.5: Effect of annealing temperature to the number of crossovers produced for the high sequence identity subtilase pair (subtilisin E, subtilisin BPN′).

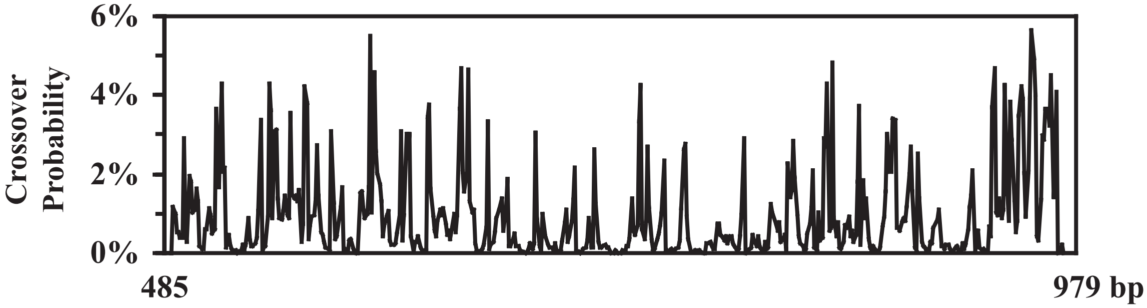


Figure 3.6: Crossover probability distributions for *in silico* family DNA shuffling of all 12 subtilases (L = 15).

Table 3.1: Comparison of melting temperature predictions for different duplexes of fragmented subtilisin E gene between the proposed model and $T_m = 81.5 + 0.41(\%\ GC) - 500/L + 16.6 \log [Na^+]$.

Sequence positions	Overlap length	Percent GC	Melting Temperature (°C)	
			Annealing model	Howley *et al.* [13]
819-828	10	50	26	30
1013-1022	10	30	17	22
529-538	10	60	32	35
804-828	25	52	61	61
779-828	50	50	72	71
729-828	100	55	81	78

Data shown is for $[Na^+] = 0.05$ M and an initial template mole fraction $x_A^0 = 2.7 \times 10^{-8}$ that corresponds to a DNA concentration of 10 mg/liter, typical for DNA shuffling.

Table 3.2: Average numbers of crossovers per sequence calculated for various fragment lengths *L* and parental sets.

L (bases)	High seq. ident. pair	Low seq. ident. pair	Set of 4 subtilases	Set of 12 subtilases
15	2.9	0.5	2.3	4.8
25	1.3	0.1	0.8	1.4
50	0.8	0.0	0.5	0.8

Chapter 4: Codon Optimization for DNA Shuffling (*e*CodonOpt)

Section 4.1: Background

In this chapter, we explore *in silico* the possibility of boosting or even specifically directing the formation of DNA recombination events by exploiting the inherent redundancy in the codon representation while recognizing that host preferences for specific patterns of codon usage may reduce the number of viable codon choices. For example, isoleucine has the following three synonymous codon representations: ATA, ATC and ATT. Therefore, it is possible to optimize the underlying parental DNA sequence codon representation for increasing and/or shaping diversity while at the same time preserving the parental amino acid encodings in the generated combinatorial protein libraries. This strategy is well suited in cases where parental sequences are synthetically generated (*e.g.*, through oligomer ligation). The utility of this approach has been recognized and exploited in an empirical way in the context of industrially developed directed evolution protocols such as oligo shuffling [1, 2] and GeneReassembly [3]. Here, a systematic computational framework named ***e***CodonOpt is proposed for exploring the limits of performance that can be achieved through codon optimization [4]. Specifically, mathematical optimization problems are formulated and solved for identifying the optimal codon representation of a parental protein set in light of different diversity objectives. DNA shuffling [5, 6] is used as the benchmark recombination method to showcase the proposed framework. However, the formulations presented here can be extended in a straightforward manner to other annealing-based recombination protocols such as StEP [7], RACHITT [8] and SCRATCHY [9].

The DNA shuffling protocol is described in detail in refs. [5, 6]. Briefly, it consists of two steps: (i) random fragmentation of a small set of parental nucleotide sequences and (ii) reassembly of the fragments through polymerase chain reaction (PCR) without primers producing a library of full-length nucleotide sequences (see Figure 4.1). During the fragment annealing step, duplexes are formed through in-frame fragment annealing. *Homoduplexes* are formed when the annealed fragments originate from the same parental sequence whereas *heteroduplexes* are formed when the two fragments are derived from different parental sequences (Figure 4.2). Upon extension, heteroduplexes give rise to *crossovers*, the junction points between segments from different parental sequences (see Figure 4.1). Crossovers provide the quantitative means for assessing diversity through recombination in DNA shuffling. Because DNA shuffling utilizes annealing and extension steps during reassembly, crossover positions in turn are biased towards regions where pairs of parental sequences share a high degree of sequence identity. This has been observed experimentally [6] and has been quantitatively modeled [10].

In this chapter, a systematic method, ***e***CodonOpt, is introduced for redirecting crossover positioning by engineering the sequence identity/free energy profile of a sequence set through codon usage optimization. Specifically, model formulations are described for (i) maximizing the average number of crossovers per recombined sequence; (ii) minimizing bias in family DNA shuffling [11] so that each of the parental sequence pair contributes a similar number of crossovers to the library; and (iii) maximizing the relative frequency of crossovers in specific 3-D structures such as loop or scaffold regions. In all cases the ***e***Shuffle software (see Chapter 3 as well as ref. [10]) is used to predict for the number, position, and type of crossovers.

Section 4.2: *e*CodonOpt Modeling Framework

The basic problem addressed in this work can be stated as follows. *Given a set of parental proteins, design the optimal nucleotide sequences encoding those proteins for a given diversity objective.* A constraint-based modeling framework is introduced that only permits nucleotide sequences encoding the underlying parental proteins as solutions. It utilizes 0-1 binary variables as on/off switches to model the presence of a specific codon choice in a given residue position. Next, the index notation, variables, parameters and constraints utilized in the basic ***e***CodonOpt model are listed:

Section 4.2.1: Indices

$$i \in \{1,2,\ldots B\} = \text{set of nucleotide sequence positions}$$

$$k \in \{1,2,\ldots,K_{tot}\} = \text{set of parental sequences}$$

$$n, n_1, n_2 \in \{A,T,C,G\} = \text{set of nucleotides in positions } i, i+1, i+2$$

Section 4.2.2: Variable Set

$$x_{ink} = \begin{cases} 1, & \text{if nucleotide } n \text{ is present at position } i \text{ in parental sequence } k \\ 0, & \text{otherwise} \end{cases}$$

Section 4.2.3: Parameters

$$a_{ink} = \begin{cases} 1, & \text{if nucleotide } n \text{ is permitted at position } i \text{ in parental sequence } k \\ 0, & \text{otherwise} \end{cases}$$

$$b_{inn_1k} = \begin{cases} 1, & \text{if nucleotides } (n,n_1) \text{ are permitted at positions } (i,i+1) \text{ in parental sequence } k \\ 0, & \text{otherwise} \end{cases}$$

$$c_{inn_2k} = \begin{cases} 1, & \text{if nucleotides } (n,n_2) \text{ are permitted at positions } (i,i+2) \text{ in parental sequence } k \\ 0, & \text{otherwise} \end{cases}$$

Specifically, the proposed model utilizes the binary variable x_{ink} to represent the underlying nucleotide representation n = (A, T, C, G) at every sequence position i of the parental protein k. Parameter a_{ink} is equal to one only if there exists at least one codon representation that allows the use of nucleotide n at position i of parental sequence k. Parameter b_{inn1k} is equal to one only if nucleotides (n,n_1) are both permitted at the first two codon positions whereas parameter c_{inn2k} is equal to one if nucleotides (n,n_2) are present at the first and third codon positions. These parameter values are determined by scanning the parental proteins against the codon translation table. See Tables 1-3 in the supplementary material of ref. [4] for a complete list of parameter values for all twenty amino acids.

Section 4.2.4: Codon Constraints

Because only one nucleotide choice n can be present at each position i of sequence k, x_{ink} is allowed a non-zero value for only one of the (A, T, C, G) choices for n for every (i,k) pair (see constraint **1**). In addition, if a particular triplet (i,n,k) is not permitted ($a_{ink} = 0$) then variable x_{ink} is forced to zero (constraint **2**).

$$\sum_{n} x_{ink} = 1, \forall\, i,k \qquad \mathbf{1}$$

$$x_{ink} = 0, \forall\, i,n,k : a_{ink} = 0 \qquad \mathbf{2}$$

Constraints **1**, **2** suffice for residues with a single degenerate position (*e.g.*, alanine). Additional constraints are needed for residues with multiple codon redundancies such as serine, arginine and leucine.

Section 4.2.5: Constraint for Serine Encoding Positions

Specifically for serine with degenerate first and second codon positions, if a consecutive pair (n,n_1) is disallowed ($b_{inn1k} = 0$) then x_{ink} and $x_{i+1,n1,k}$ cannot both be equal to one.

$$x_{ink} + x_{i+1,n_1,k} \le 1, \forall\, i,n,n_1,k : b_{inn_1k} = 0 \qquad \mathbf{3}$$

Section 4.2.6: Constraint for Arginine, Leucine and Serine Encoding Positions

Similarly, for degeneracies in the first and third position for arginine, leucine and serine residues, the following constraint is needed.

$$x_{ink} + x_{i+2,n_2,k} \le 1, \forall\, i,n,n_2,k : c_{inn_2k} = 0 \qquad \mathbf{4}$$

Section 4.2.7: Host Requirements

Substantial evidence exists that specific organisms prefer certain synonymous codons (*i.e.*, for the same amino acid) over others. It has been shown that the frequency of codon usage is directly proportional to the corresponding tRNA population (*e.g.*, *Escherichia coli* [12], *Drosophila melanogaster* [13], and *Caenorhabditis elegans* [14]). Rare codons are generally undesirable because they decrease protein expression levels due to translational stalling [15]. The proposed constraint framework is flexible enough to disallow the presence of rare codons by appropriately redefining parameters a_{ink}, b_{inn1k} and c_{inn2k}. For example, disallowing the rare isoleucine codon ATA simply requires setting $a_{i+2,\mathrm{A}k} = 0$ for all isoleucine positions in the DNA sequence thus eliminating the use of A in the third position. It is worthwhile noting though that the removal of all rare codons can cause protein folding problems [16]. Therefore, instead of completely eliminating rare codons it is possible to construct constraints that maintain codon usage ratios within some upper and lower bounds defined around the average organism-specific codon usage preferences.

A systematic approach for designing an organism-specific codon representation requires the use of a scoring metric to quantify the level of preferred codons present. Here we formulate constraints requiring that the host-specific score for each of the parental sequences is greater than a specified lower bound. The use of two such metrics is investigated: (i) Codon Adaptation Index (*CAI*) [17] and (ii) Major Codon Usage (*MCU*) [12, 18]. In calculating the *CAI*, each codon (n,n_1,n_2) is assigned a weight ω_{nn1n2} that ranges from 0 (low frequency) to 1 (high frequency) based on how often it is utilized in the host organism. For instance, ATC is the most frequently used isoleucine codon, so $\omega_{\mathrm{ATC}} = 1$, while the remaining isoleucine codons are assigned weights less than 1 ($\omega_{\mathrm{ATT}} = 0.185$, $\omega_{\mathrm{ATA}} = 0.008$). A complete table of weights can be found in ref. [17] for *E. coli*. The CAI_k for a particular parental sequence k is found by taking the geometric mean of all the individual codon weights.

$$CAI_k = \prod_{i=1,4,7,\ldots} \left(\sum_{n,n_1,n_2} \omega_{n,n_1,n_2} \left(x_{ink} \cdot x_{i+1,n_1,k} \cdot x_{i+2,n_2,k} \right) \right)^{\frac{1}{B/3}}, \forall\, k \qquad \mathbf{5}$$

Two steps are necessary to express this relation in a linear form: (i) the logarithm is taken on both sides, transforming the geometric mean into an arithmetic one, and (ii) the three-term product is recast at the expense of introducing additional variables. Details of the exact linearization are found in the appendix of ref. [4]. Maintaining CAI_k above a desired lower bound CAI_{min} is attained with the following simple constraint.

$$CIA_k \geq CAI_{min}, \forall\, k \qquad \mathbf{6}$$

An alternative method for scoring a codon representation for a specific host is the calculation of the Major Codon Usage (*MCU*) metric, which quantifies the fraction of codons utilized in a given representation that are “major” for that organism. Major codons are defined as those codons that appear with greater frequency in genes with high levels of codon bias [18]. Whether a codon is a major codon or not is captured by the parameter $\mu_{nn_1n_2}$, which is equal to one if codon (n,n_1,n_2) is a major codon and zero otherwise (*e.g.*, for isoleucine, $\mu_{ATC} = 1$ and $\mu_{ATT} = \mu_{ATA} = 0$). A tabulation of major codons for *E. coli* is found in ref. [12]. The following expression is used to calculate the MCU_k metric for each parental sequence.

$$MCU_k = \frac{1}{B/3}\sum_{i=1,4,7,\ldots}\left(\sum_{n,n_1,n_2}\mu_{n,n_1,n_2}\left(x_{ink}\cdot x_{i+1,n_1,k}\cdot x_{i+2,n_2,k}\right)\right), \forall\, k \qquad \mathbf{7}$$

The three-term product is recast into an equivalent linear form in the same way as constraint **5**, and a lower limit on *MCU* is assigned as follows.

$$MCU_k \geq MCU_{min}, \forall\, k \qquad \mathbf{8}$$

By requiring CAI_k (with constraints **5-6**) or MCU_k (constraints **7-8**) to be greater than a desired lower bound, codon optimization can be performed while maintaining organism-specific usage ratios.

Section 4.2.8: Limiting the Number of Codon Manipulations

Alternatively, one may want to limit the number of codon representation changes (*i.e.*, silent nucleotide mutations) made to the wild-type DNA sequences. Specifically, the total number of silent nucleotide point mutations in the designed sequences could be set to be less than an upper limit *P*. This requires the definition of the following additional parameters:

$$\delta_{nn'} = \begin{cases} 1, & \text{if } n = n' \text{ (nucleotide identity)} \\ 0, & \text{otherwise} \end{cases}$$

w_{ink} = codon representation corresponding to the wild-type (original) sequences

P = maximum number of point mutations permitted from wild-type nucleotide sequences

Constraint **9** establishes an upper bound to the total number of allowable silent point mutations.

$$\sum_k\sum_i\sum_{n,n'}\left(1-\delta_{nn'}\right)x_{ink}\cdot w_{in'k} \leq P \qquad \mathbf{9}$$

This constraint-based modeling framework allows searching the space of possible codon representations (codified in variable x_{ink} and subject to constraints **1-4**) for the one that optimizes a user defined diversity objective. In the next section three such diversity objectives are discussed.

Section 4.3: Diversity Objectives

With the codon constraints in place, a number of different diversity objectives are explored: (i) maximizing the number of crossovers, (ii) minimizing bias in family DNA shuffling, and (iii) maximizing the relative frequency of crossovers in specific structural regions. For objective (i), the effect of *E. coli* preferred codon sets on the number of crossovers is studied by including constraints **5-6** or **7-8** in the optimization model. Optimized sequences for each of the objectives are provided in the supplementary material of ref. [4].

Section 4.3.1: Objective I: Maximizing the Total Number of Crossovers

Crossover statistics for different parental sequence codon representations can be estimated by the ***e***Shuffle program [10]. However, because the clock time of an ***e***Shuffle run can range from minutes to hours, utilizing ***e***Shuffle in the context of optimization loops is impractical for all but the simplest cases. Instead, two simple surrogate objectives for crossover generation are postulated and subsequently tested: (a) maximization of the pairwise sequence identity between the parental DNA sequences and (b) minimization of the total free energy change upon complete annealing of the two DNA sequences. Both of these surrogates for crossover generation capture the fact that crossover formation in DNA shuffling occurs predominantly within regions of near perfect sequence identity. A flowchart illustrating the sequence of calculations followed for this and other diversity objectives is shown in Figure 4.3.

Section 4.3.1.1: Surrogate (a): Maximizing Pairwise Sequence Identity

This intuitive surrogate for crossover generation implies that the degree of sequence identity between a pair of DNA sequences correlates with the number of crossovers generated. The calculation of the sequence identity is performed by counting the total number of matching nucleotides, $M_{k\widetilde{k}}$, between two aligned parental sequences k and $\widetilde{k}$.

$$M_{k\widetilde{k}} = \sum_{i} \sum_{n,\widetilde{n}} \delta_{n\widetilde{n}} x_{ink} x_{i\widetilde{n}\widetilde{k}}, \ \forall \ k, \widetilde{k} > k \qquad \textbf{10}$$

Note that the nonlinearity introduced by the product of binary variables ($x_{ink} x_{i\widetilde{n}\widetilde{k}}$) is eliminated (see the appendix of ref. [4] for details). Therefore, the first surrogate for maximizing crossover generation upon codon optimization involves *maximizing* $M_{k\widetilde{k}}$ subject to constraints **1-4** and **10**. Constraint sets **5-6** or **7-8** are added if a host-specific codon representation is desired, while constraint **9** is added if a limit on the total number of silent nucleotide mutations is needed. This problem belongs to the class of mixed-integer linear programming (MILP) problems and is solved using CPLEX 7.0 [19] accessed through the GAMS modeling environment [20]. Note without any additional restrictions such as **5-9**, this problem decomposes over codons and can be solved in linear complexity. This decoupling, however, does not hold for the second surrogate.

Section 4.3.1.2: Surrogate (b): Minimizing the Free Energy Change of Annealing

The second surrogate objective implies that crossover generation correlates with the total free energy change upon complete annealing of the recombining pair of DNA sequences. The free energy change is approximated using empirical nearest-neighbor parameters [21] that decompose the free energy calculation into the sum of the contributions of overlapping 2-nucleotide (nt) units (see Figure 4.4). Matching pairs contribute negative free energy terms lowering the total free energy change of annealing, whereas mismatches contribute positive terms increasing the free energy change. Parameter set $\Delta G^{pair}_{nn_1\widetilde{n}\widetilde{n}_1}$ stores the free energy change associated with the annealing of nucleotide pair (n,n_1) with ($\widetilde{n},\widetilde{n}_1$). The total free energy change $\Delta G_{k\widetilde{k}}$ upon complete annealing of two parental sequences ($k,\widetilde{k}$) is calculated by summing over the contribution of all nucleotide pairs at positions (i,i + 1) along the entire sequence length.

$$\Delta G_{k\widetilde{k}} = \sum_{i} \sum_{n,n_1,\widetilde{n},\widetilde{n}_1} \Delta G^{pair}_{nn_1\widetilde{n}\widetilde{n}_1} \left(x_{ink} x_{i+1,n_1k} x_{i\widetilde{n}\widetilde{k}} x_{i+1,\widetilde{n}_1\widetilde{k}} \right), \ \forall \ k, \widetilde{k} > k \qquad \textbf{11}$$

Note that the four-term product in the expression is subsequently expressed in an equivalent linear form. The exact linearization is found in the appendix of ref. [4]. Therefore, the second surrogate for crossover generation in DNA shuffling involves *minimizing* $\Delta G_{k\tilde{k}}$ subject to constraints **1-4**, **11** and optionally **5-6**, **7-8** or **9**.

These two surrogate choices are tested based on the DNA shuffling of two glycinamide ribonucleotide (GAR) transformylases. Specifically, the DNA shuffling of the *E. coli* and human versions of GAR transformylase is studied. The wild-type parental sequences share a very low nucleotide sequence identity of 47% even though the two enzymes share the same function and presumably the same structure. In the absence of any codon optimization, DNA shuffling crossovers are extremely rare for this system as shown previously in ref. [10]; therefore, there is clearly a need to increase the number of crossovers generated.

First, surrogate objective (a), maximizing the sequence identity of the two GAR transformylases, M_{12}, is examined. The maximum sequence identity upon codon optimization is identified for an increasing number of allowed silent nucleotide mutations. These codon-engineered parental sequences are next fed to ***e*****Shuffle** to predict the total number of crossovers expected to be formed upon DNA shuffling. Crossover numbers are plotted in Figure 4.5 from zero (wild-type) to 320 permitted silent mutations. Interestingly, after 90-100 point mutations are accumulated, the total number of crossovers rapidly increases reaching a maximum value of about 1.5 crossovers per sequence. Beyond this point, sequence identity ceases to correlate with crossover generation leading to the plateau effect beyond 140 silent mutations as shown in Figure 4.5. The second surrogate objective, involving the minimization of the free energy change of annealing, ΔG_{12}, provides much better correlation with the extent of crossover formation. Almost twice as many crossovers are formed compared with the previous surrogate (see Figure 4.5). The key difference is that, unlike sequence identity, the free energy change continues to correlate strongly with crossover formation even for very high numbers of silent mutations preventing the onset of the plateau. Interestingly, the extent of crossover formation is only mildly affected by excluding all *E. coli* rare codons from consideration (*i.e.*, ATA, AGA, AGG, TGT, CTA, CCC, CGA and CGG [22]). Even when a lower bound is placed on the *CAI* metric (see Figure 4.6a) or *MCU* criterion (see Figure 4.6b) for the parental sequences, comparable numbers of crossovers are still generated. Even the most stringent requirement ($CAI = MCU = 1$) results in a less than 50% drop in the predicted number of crossovers from the theoretical maximum. These results demonstrate that codon optimization can be effectively performed for organism-specific codon sets leading to higher levels of protein expression in addition to a more diverse combinatorial library.

The strength of correlation of the two surrogate functions with the total number of crossovers generated is shown more clearly in Figure 4.7. It is noteworthy that increasing sequence identity beyond a certain level does not increase crossover generation (Figure 4.7a). In fact, a reversal in the sign of correlation occurs near the end of the plot. On the other hand, free energy change correlates monotonically and almost linearly (Figure 4.7b) with the extent of crossover formation. $\Delta G_{k\tilde{k}}$ outperforms sequence identity as a surrogate for crossover formation because it appropriately weighs the thermodynamic contribution of different matches and mismatches. In addition, by considering the contribution of overlapping nucleotide pairs, it places a higher emphasis on blocks of sequence identity over isolated nucleotide matches. Sequence identity is not as successful as a surrogate because the matching nucleotides do not necessarily group into crossover-generating blocks of sequence identity. The qualitative trends in the result hold for a wide range of example problems studied so far implying that free energy of annealing is universally superior to sequence identity as a predictor of crossover formation. This result has a substantial implication on the way DNA shuffling studies are conducted and parental DNA sequences are engineered.

Section 4.3.2: Objective II: Minimizing Bias in Family DNA Shuffling

Family DNA shuffling [11] extends DNA shuffling to more than two parental sequences allowing the simultaneous mixing of genetic information from many homologous DNA sequences. However, a strong possibility exists for a biased library in which only a small subset of the shuffled family generates crossovers while the remainder of the parental set does not participate in the recombination process. This results in a biased combinatorial library where the majority of crossovers originate from only a few pairs and the majority of parental sequences do not contribute to the genetic diversity of the combinatorial library.

Earlier, it was shown that the free energy change of annealing, $\Delta G_{k\tilde{k}}$, is a good predictor for the number of crossovers generated by a pair of parental sequences. By building on this constraint-based framework the goal here is to ensure that each parental sequence pair contributes an approximately equal number of crossovers to the library while the total number of generated crossovers stays as high as possible. This is ensured mathematically by minimizing the average free energy change over all parental sequence pairs while constraining all of the pairwise free energy changes within a window centered about the mean. The mean free energy change, ΔG_{mean}, is given by

$$\Delta G_{mean} = \frac{\sum_{k,\tilde{k}} \Delta G_{k\tilde{k}}}{K_{tot}(K_{tot}-1)/2} \qquad \textbf{12}$$

The parameter α is used to set the size of the window in which each of the pairwise free energy changes can fall. For example, setting α = 10% ensures that all $\Delta G_{k\tilde{k}}$'s are within 10% from ΔG_{mean}. Two linear constraints are utilized to set the upper and lower bounds on $\Delta G_{k\tilde{k}}$ separately.

$$\begin{aligned} \Delta G_{k\tilde{k}} &\geq (1-\alpha)\Delta G_{mean},\ \forall\ k,\tilde{k} > k \\ \Delta G_{k\tilde{k}} &\leq (1+\alpha)\Delta G_{mean},\ \forall\ k,\tilde{k} > k \end{aligned} \qquad \textbf{13,14}$$

Minimizing ΔG_{mean} subject to constraints **1-4**, **11-14** increases overall crossover frequency while simultaneously reducing bias towards particular parental sequence pairs.

The family DNA shuffling of a family of four cephalosporinases [11] is used here to demonstrate the proposed framework. For the wild-type sequences, ***e***Shuffle predicts that 70% of the crossovers are generated by a single parental pair, *Citrobacter freundii* and *Enterobacter cloacae* (1 and 2), as shown in Figure 4.8. Solving the optimization problem posed above with α = 10% for the four cephalosporinases greatly compacts the range of pairwise free energy changes by a factor of 4.5 (Figure 4.9). This leads to a crossover distribution that is much more even (Figure 4.8). Crossovers between *Citrobacter freundii* and *Enterobacter cloacae* (1 and 2), previously in excess of 70%, are reduced to a contribution of only 17%, while other types of crossovers are boosted. In addition to removing bias from the library, the optimization procedure greatly increases the total number of crossovers per sequence, from 0.87 up to 12.1. The ability to customize the number and type of crossovers for a sequence family can significantly affect the design of family DNA shuffling experiments. Codon optimization can substantially augment the set of feasible parental recombination candidates since homology can be custom-engineered.

Section 4.3.3: Objective III: Directing Crossovers to Specific Structural Regions

Here we examine how crossovers can be directed to specific structural regions through codon optimization. These regions can be secondary structure units such as helices or sheets, specific domains of multi-domain proteins or sites that bind either substrates or cofactors. Currently there is substantial effort in the literature aimed at identifying regions where crossovers are more likely to be tolerated giving rise to functional hybrids. Some approaches are hypothesis driven such as multipool swapping [23] and minimum schema disruption [24], whereas others attempt to identify these regions by employing structural energy calculations [25]. Given these desirable crossover regions a parameter L_i is defined that flags them along the sequence,

$$L_i = \begin{cases} 1, & \text{if position } i \text{ is a desirable crossover position} \\ 0, & \text{otherwise} \end{cases}$$

Preferentially directing crossovers to one region is achieved by minimizing $\Delta G_{annealing}$ in the preferred regions while maximizing $\Delta G_{annealing}$ in the remaining regions. Expressions for the change in free energy for desirable and undesirable crossover regions are given below.

$$\Delta G_{desirable} = \sum_{k,\tilde{k}>k} \sum_{\substack{i:L_i=1 \\ L_{i+1}=1}} \sum_{n,n_1,\tilde{n},\tilde{n}_1} \Delta G^{pair}_{nn_1\tilde{n}\tilde{n}_1} \left(x_{ink} x_{i+1,n_1k} x_{i\tilde{n}\tilde{k}} x_{i+1,\tilde{n}_1\tilde{k}} \right)$$

$$\Delta G_{undesirable} = \sum_{k,\tilde{k}>k} \sum_{\substack{i:L_i=0 \\ L_{i+1}=0}} \sum_{n,n_1,\tilde{n},\tilde{n}_1} \Delta G^{pair}_{nn_1\tilde{n}\tilde{n}_1} \left(x_{ink} x_{i+1,n_1k} x_{i\tilde{n}\tilde{k}} x_{i+1,\tilde{n}_1\tilde{k}} \right)$$

The proposed approach is demonstrated by preferentially allocating crossovers to the loop and scaffold regions of the phosphoribosylanthranilate isomerase (PRAI) domain of a bifunctional enzyme [26]. PRAI is an α/β barrel protein with a scaffold region spanning the inner β-barrel and the eight outer α-helices (shown in purple and gold in Figure 4.10). Loops are defined as the connecting regions between the β-barrel and the α-helices and are shown in white in Figure 4.10. Parameter L_i indicates whether a sequence position is within a loop or not and its values are superimposed on the PRAI 3-D structure. Two design objectives are pursued: (i) directing crossovers to loop regions by minimizing (ΔG_{loop} - $\Delta G_{scaffold}$) and (ii) directing crossovers to the scaffold by minimizing ($\Delta G_{scaffold}$ - ΔG_{loop}). Both of these two optimization problems are solved for the DNA shuffling of *E. coli* and *Salmonella enterica Typhi* versions of the PRAI domain. For the wild-type sequences, ***e***Shuffle predicts that crossovers are predominantly located in the scaffold region (Figure 4.11). Upon loop-optimization, crossovers in loop regions are increased almost twenty-fold outpacing those in the scaffold region by 40%. Alternatively, scaffold-optimization increases the number of crossovers found in the scaffold region by tenfold. The crossover locations after optimization are superimposed against the 3-D structure in Figure 4.12. Codon optimization dramatically reshapes the crossover distribution (see Figure 4.12) by biasing it towards targeted regions. Interestingly, a by-product of the optimization is that for both the loop and scaffold-optimized cases, overall crossover generation is greatly increased. The results obtained for this example demonstrate that codon optimization provides an effective strategy for directing crossovers to desirable protein regions.

Section 4.4: Implementation

Optimization problems were solved using CPLEX 7.0 [19] accessed through the GAMS modeling environment [20] on an IBM RS6000-270 workstation. CPU times were on the order of seconds for Objectives I(a), I(b) and III; and hours for Objective II. ***e***Shuffle runs were performed assuming a standard DNA shuffling setup: annealing temperature 55°C, fragment length 25-nt, DNA concentration 10 ng/µL, $[K^+]$ = 50 mM and $[Mg^{+2}]$ = 2.2 mM. Nucleotide and amino acid sequences utilized in the examples were downloaded from GenBank via the Entrez system [27]. Accession numbers for wild-type proteins were: P08179 and P22102 (*E. coli* and human GAR transformylases); CAA35959, CAC08446, CAA44850 and AAK70221 (*C. freundii*, *E. cloacae*, *Y. enterocolitica* and *K. pneumoniae* cephalosporinases); and AAA57299 and CAD08407 (*E. coli* and *S. enterica Typhi* PRAI domains). The 3-D structure of the PRAI domain (1PII, residues 256-452) was downloaded from the Protein Data Bank [28]. Protein Explorer (http://proteinexplorer.org) was used to render 3-D structures.

Section 4.5: Summary

In this chapter, a systematic computational framework, ***e***CodonOpt, for designing parental DNA sequences for directed evolution experiments through codon usage optimization was introduced. With the proposed MILP formulation, we designed parental sequence sets that met a variety of diversity objectives while observing host-specific codon preferences based on the *CAI* and *MCU* metrics. Initially, the number of crossovers generated by DNA shuffling was boosted substantially by optimizing the annealing free energy profile of two GAR transformylases. Then, crossover bias towards specific parental pairs was reduced for an engineered family of cephalosporinases while simultaneously increasing the total number of crossovers formed by family DNA shuffling. Finally, crossovers were preferentially allocated to specific structural regions in a PRAI domain allowing a customized crossover distribution. Much flexibility is present in the constraint-based framework, permitting the investigation of many other choices for diversity objectives.

We believe that codon engineering is capable of expanding and shaping the sequence space spanned by directed evolution experiments. As our knowledge of how recombination events preserve or disrupt protein structure improves, optimal design of the parental DNA sequence set will allow a more focused probing of sequence space in only those regions that are likely to yield functional hybrids. This, in turn, will lead to a more efficient utilization of experimental resources, saving time and effort by reducing the number of evolutionary cycles that must be performed for a successful protein design effort.

Section 4.6: References

1. Arnold, F.H. & Volkov, A.A. (1999), "Directed evolution of biocatalysts," *Curr Opin Chem Biol* **3**: 54-59.
2. Stemmer, W.P.C. (2000), "US6,132,970: Methods of Shuffling Polynucleotides."
3. Short, J.M. (1999), "US5,965,408: Method of DNA Reassembly by Interrupting Synthesis."
4. Moore, G.L. & Maranas, C.D. (2002), "eCodonOpt: a systematic computational framework for optimizing codon usage in directed evolution experiments," *Nucleic Acids Res* **30**(11): 2407-2416.
5. Stemmer, W.P.C. (1994), "DNA shuffling by random fragmentation and reassembly: in vitro recombination for molecular evolution," *Proc Natl Acad Sci USA* **91**: 10747-10751.
6. Stemmer, W.P.C. (1994), "Rapid evolution of a protein in vitro by DNA shuffling," *Nature* **370**: 389-391.
7. Zhao, H., Giver, L., Shao, Z., Affholter, J.A. & Arnold, F.H. (1998), "Molecular evolution by staggered extension process (StEP) in vitro recombination," *Nature Biotech* **16**: 258-261.
8. Coco, W.M., Levinson, W.E., Crist, M.J., Hektor, H.J., Darzins, A., Pienkos, P.T., Squires, C.H. & Monticello, D.J. (2001), "DNA shuffling method for generating highly recombined genes and evolved enzymes," *Nature Biotech* **19**: 354-359.
9. Lutz, S., Ostermeier, M., Moore, G.L., Maranas, C.D. & Benkovic, S.J. (2001), "Creating multiple-crossover DNA libraries independent of sequence identity," *Proc Natl Acad Sci USA* **98**: 11248-11253.
10. Moore, G.L., Maranas, C.D., Lutz, S. & Benkovic, S.J. (2001), "Predicting crossover generation in DNA shuffling," *Proc Natl Acad Sci USA* **98**: 3226-3231.

11. Crameri, A., Raillard, S., Bermudez, E. & Stemmer, W.P.C. (1998), "DNA shuffling of a family of genes from diverse species accelerates directed evolution," *Nature* **391**: 288-291.
12. Ikemura, T. (1985), "Codon Usage and tRNA Content in Unicellular and Multicellular Organisms," *Mol Biol Evol* **2**: 13-34.
13. Moriyama, E.N. & Powell, J.R. (1997), "Codon Usage Bias and tRNA Abundance in Drosophila," *J Mol Evol* **45**: 514-523.
14. Duret, L. (2000), "tRNA gene number and codon usage in C. elegans genome are co-adapted for the optimal translation of highly expressed genes," *Trends Genet* **16**: 287-289.
15. Baca, A.M. & Hol, W.G.J. (2000), "Overcoming codon bias: A method for high-level overexpression of Plasmodium and other AT-rich parasite genes in Escherichia coli," *Int J Parasitol* **30**: 113-118.
16. Komar, A.A., Lesnik, T. & Reiss, C. (1999), "Synonymous codon substitutions affect ribosome traffic and protein folding during in vitro translation," *FEBS Letters* **462**: 387-391.
17. Sharp, P.M. & Li, W. (1987), "The codon adaptation index - a measure of directional synonymous codon usage bias, and its potential applications," *Nucleic Acids Res* **15**: 1281-1295.
18. Akashi, H. (1995), "Inferring Weak Selection From Patterns of Polymorphism and Divergence at ``Silent" Sites in Drosophila DNA," *Genetics* **139**: 1067-1076.
19. Brooke, A., Kendrick, D., Meeraus, A. & Raman, R., *GAMS: The Solver Manuals*. 1998: GAMS Development Corporation.
20. Brooke, A., Kendrick, D., Meeraus, A. & Raman, R., *GAMS: A User's Guide*. 1998: GAMS Development Corporation.

21. SantaLucia Jr., J. (1998), "A unified view of polymer, dumbbell, and oligonucleotide DNA nearest neighbor thermodynamics," *Proc Natl Acad Sci USA* **95**: 1460-1465.

22. Nakamura, Y., Gojobori, T. & Ikemura, T. (2000), "Codon usage tabulated from international DNA sequence databases: status for the year 2000," *Nucleic Acids Res* **28**: 292.

23. Bogarad, L.D. & Deem, M.W. (1999), "A hierarchical approach to protein molecular evolution," *Proc Natl Acad Sci USA* **96**: 2591-2595.

24. Voigt, C., Wang, Z. & Arnold, F.H. (2000), "A Computational Approach to Directed Evolution," *American Institute of Chemical Engineers Annual Meeting*.

25. Voigt, C.A., Mayo, S.L., Arnold, F.H. & Wang, Z. (2000), "Computational Method to Reduce the Search Space for Directed Protein Evolution," *Proc Natl Acad Sci USA* **98**: 3778-3783.

26. Altamirano, M.M., Blackburn, J.M., Aguayo, C. & Fersht, A.R. (2000), "Directed evolution of new catalytic activity using the alpha/beta-barrel scaffold," *Nature* **98**: 3288-3293.

27. Benson, D.A., Karsch-Mizrachi, I., Lipman, D.J., Ostell, J., Rapp, B.A. & Wheeler, D.L. (2002), "GenBank," *Nucleic Acids Res* **30**: 17-20.

28. Westbrook, J., Feng, Z., Jain, S., Bhat, T.N., Thanki, N., Ravichandran, V., Gilliland, G.L., Bluhm, W., Weissig, H., Greer, D.S., Bourne, P.E. & Berman, H.M. (2002), "The Protein Data Bank: unifying the archive," *Nucleic Acids Res* **30**: 245-248.

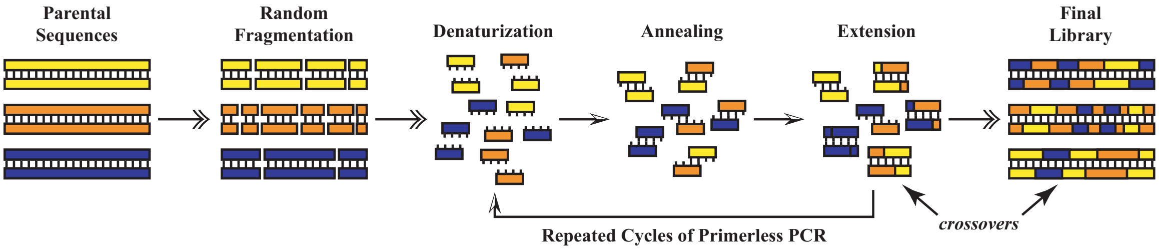


Figure 4.1: Diagram of DNA shuffling. First, the parental sequences are randomly fragmented by the enzyme DNaseI. The fragments are then reassembled by repeated primerless PCR cycles. Each cycle consists of (i) denaturization, when double strands of DNA are separated into single strands, (ii) annealing, when DNA fragments reanneal forming duplexes and (iii) extension, when the addition of new nucleotides is catalyzed by a polymerase enzyme. Crossovers are generated during the extension step when duplexes composed of fragments from different parents have new nucleotides added. After many cycles, full-length sequences are reassembled.

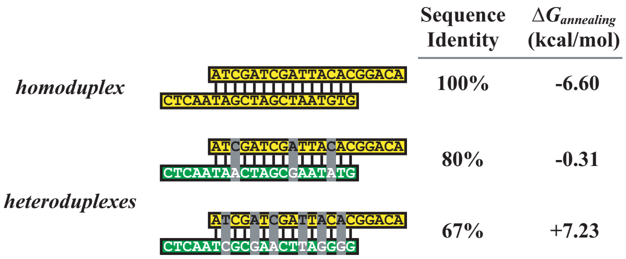


Figure 4.2: One homoduplex and two heteroduplex examples. Gray columns denote mismatches in the heteroduplexes. Calculation of the annealing free energy change is for a DNA concentration of 10 ng/μL, temperature 55°C, $[K^+]$ = 50 mM and $[Mg^{+2}]$ = 2.2 mM.

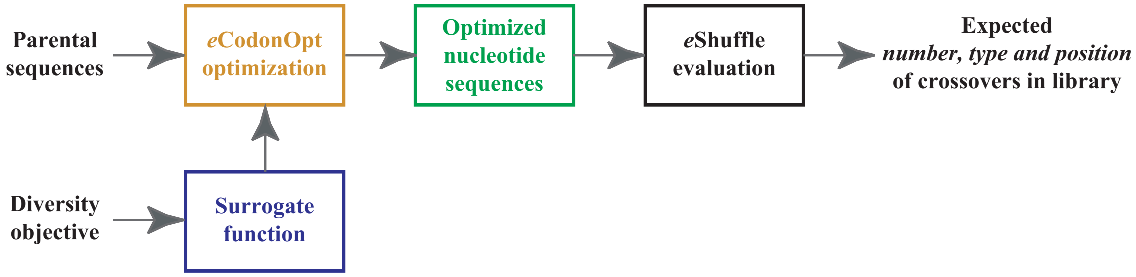
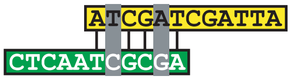


Figure 4.3: Flowchart showing the sequence of calculations followed in the ***e**CodonOpt* optimization procedure.



$$\Delta \boldsymbol{G} = \Delta \boldsymbol{G}_{\mathtt{AT/TC}} + \Delta \boldsymbol{G}_{\mathtt{TC/CG}} + \Delta \boldsymbol{G}_{\mathtt{CG/GC}} + \Delta \boldsymbol{G}_{\mathtt{GA/CG}} + \Delta \boldsymbol{G}_{\mathtt{AT/GA}}$$

Figure 4.4: Calculation of annealing free energy change using overlapping nearest-nucleotide pairs.

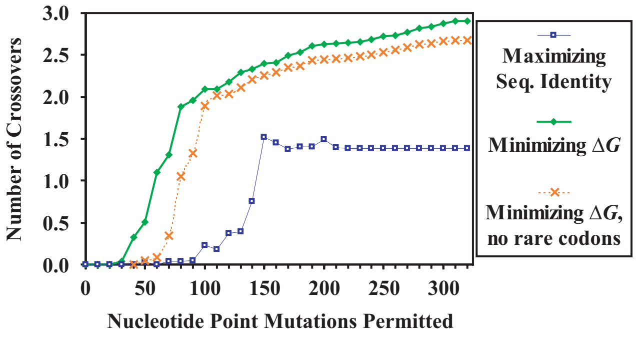


Figure 4.5: The total number of crossovers increases as more point mutations are permitted. Free energy change outperforms sequence identity as a surrogate. When rare *E. coli* codons are excluded, only a slight decrease is seen in the total number of crossovers.

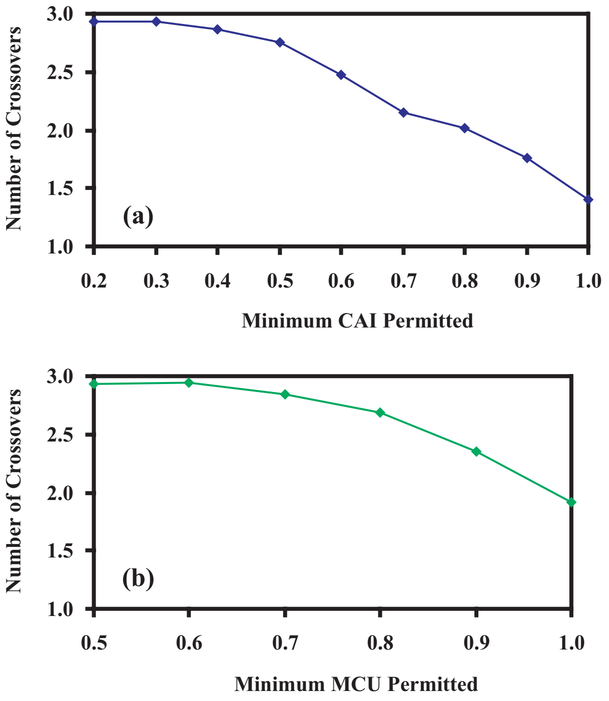


Figure 4.6: (a) As expected, the number of crossovers decreases as the lower bound on the Codon Adaptation Index (CAI_{min}) increases from 0.2 to 1. (b) The number of crossovers decreases as the minimum Major Codon Usage (MCU_{min}) increases from 0.5 to 1.

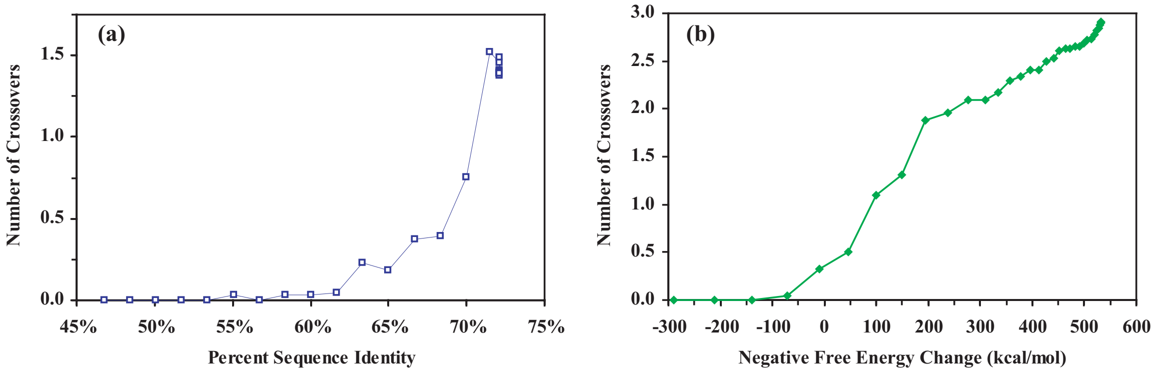
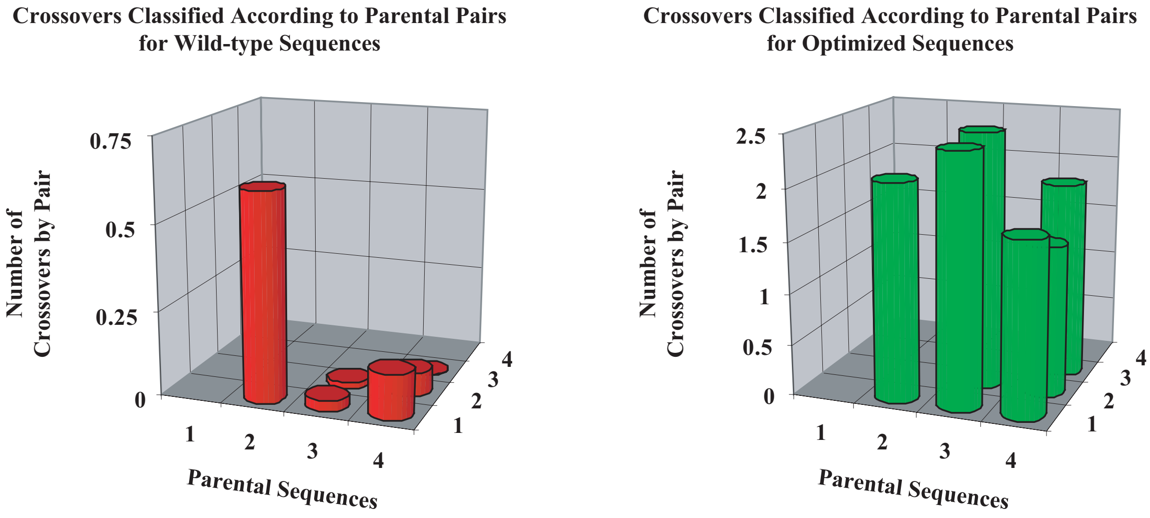


Figure 4.7: (a) Plot of the percent sequence identity of optimized (max M_{12}) sequences versus the total number of crossovers as a function of the number of silent mutations permitted. (b) Plot of the negative of the free energy change for optimized (min ΔG_{12}) sequences versus the number of crossovers as the number of silent mutations permitted is increased.



Crossovers Classified According to Parental Pairs for Wild-type Sequences

Crossovers Classified According to Parental Pairs for Optimized Sequences

Figure 4.8: Crossover statistics before and after optimization for all possible pairs of parental sequences: (1) *Citrobacter freundii*, (2) *Enterobacter cloacae*, (3) *Yersinia enterocolitica*, and (4) *Klebsiella pneumoniae*.

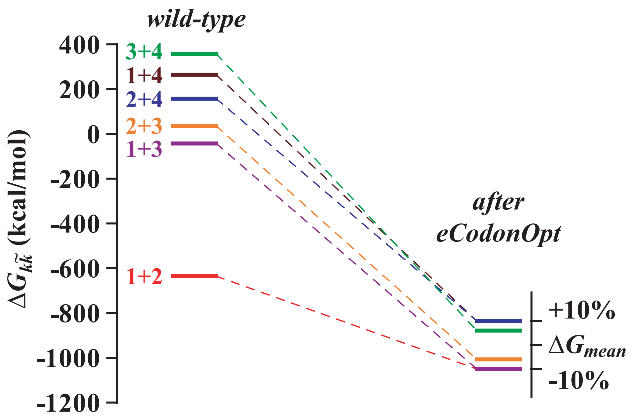


Figure 4.9: Free energy of annealing before and after optimization for all six pairs of parental sequences.

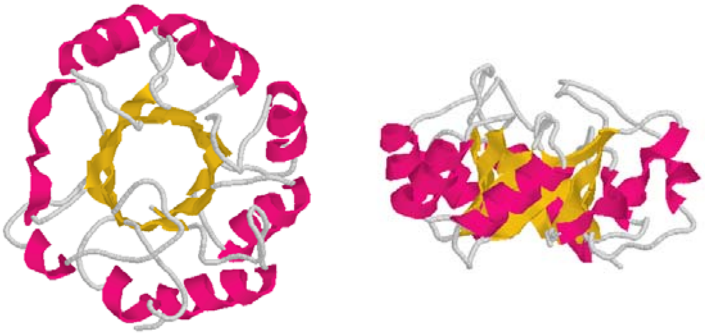


Figure 4.10: Top and side view of the *E. coli* PRAI protein domain. Scaffold regions are colored purple (α-helices) and gold (β-barrel) while connecting loop regions are colored grey.

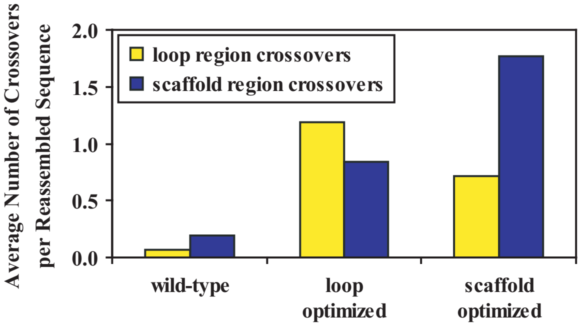


Figure 4.11: Codon optimization results for loop and scaffold regions in the PRAI domain.

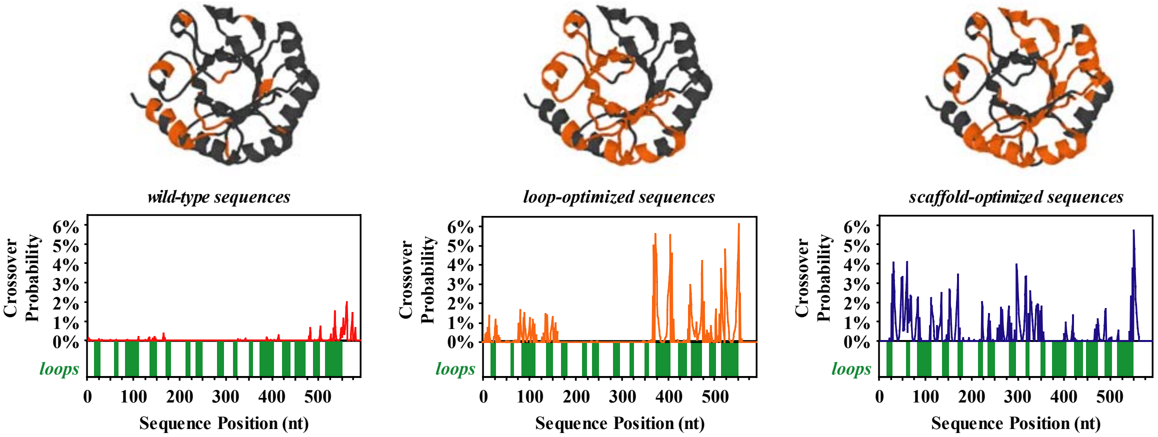


Figure 4.12: Crossover position statistics before and after codon optimization. Loop regions are represented by green bars in the strip chart. Orange residues in the 3-D structures represent positions with crossover probability greater than 0.1%.

Chapter 5: Modeling Crossover Generation in SCRATCHY (*e*SCRATCHY)

Section 5.1: Background

As mentioned in Section 1.1, sequence homology-dependent methods for recombining genes have been successful at evolving improved proteins [1-14]. An inherent limitation of these methods is their dependence on DNA sequence identity for generating diversity. This precludes the creation of crossovers between genes at loci of low homology, biasing crossover positions towards regions of highest homology. In general, a severe bias toward parental recombination is observed when sequences with less than 70% sequence identity are DNA shuffled. Given the fact that protein structure is more frequently conserved than DNA homology, homology-dependent methods for recombining genes may potentially exclude solutions to protein engineering problems.

The need for a recombination protocol capable of freely exchanging genetic diversity without sequence identity limitations has motivated the creation of *i*ncremental *t*runcation for the *c*reation of *h*ybrid enz*y*mes (ITCHY). ITCHY allows one to create comprehensive fusion libraries between fragments of genes without any sequence dependency [15-17]. However, the main drawback of the method, as well as similar techniques [18], is that members of these libraries contain only one crossover per gene. As suggested earlier [19], the DNA shuffling of ITCHY libraries could potentially introduce multiple crossovers between the genes of interest by preserving ITCHY crossovers (prepositioned crossovers) in the starting material and by recombining regions of homology between genes (Figure 5.1). This combination of ITCHY and DNA shuffling has been named SCRATCHY. In this chapter, a modeling framework named ***e***SCRATCHY for quantifying crossover generation in the SCRATCHY protocol is presented.

Section 5.2: *e*SCRATCHY Modeling Framework

An *in silico* modeling framework for crossover statistics prediction named ***e***SCRATCHY was developed in conjunction with experimental work on SCRATCHY.

The modeling framework builds on the ***e***Shuffle program presented above for assessing the generation of crossovers in the context of DNA shuffling [20].

SCRATCHY can be abstracted as the family DNA shuffling of an artificially created superfamily containing all single crossover hybrids between the two genes of interest. The presence of fragments during reassembly that contain prepositioned crossovers extends the sequence space accessed by SCRATCHY compared to the one available to traditional DNA shuffling. Therefore, when fragment-fragment hybridization is considered in the reassembly algorithm of ***e***SCRATCHY, it is necessary to keep track of not only the overlapping region but also if one (or both) fragments contain a prepositioned crossover and whether this crossover is located within or outside the overlapping region (Figure 5.2). These considerations give rise to three hypothetical yet distinct mechanisms for generating crossovers in contrast to the single mechanism (*e.g.*, the extension of a heteroduplex) encountered in ***e***Shuffle – namely, (i) the extension of a heteroduplex as in ***e***Shuffle, (ii) the incorporation of a prepositioned crossover, or (iii) the extension of a hybrid-duplex which occurs when a fragment already containing a prepositioned crossover anneals with another fragment with the crossover positioned in the duplex. Hybrid-duplexes are part stabilizing homoduplex and part crossover-generating heteroduplex presumably enabling the SCRATCHY protocol to generate crossovers within narrower sequence identity stretches than DNA shuffling. It is important to note that these three hypothesized mechanisms reflect, and thus are dependent upon, the abstraction of the proposed reassembly algorithm as a recursive sequence of annealing events. Clearly, the sequence of actual hybridization events occurring in the reacting mixture over multiple cycles defines a process much more complex than the level of detail captured within ***e***SCRATCHY. Specifically, hybrid-duplexes may also occur in DNA shuffling but only after the first reassembly cycle and only between fragments arising from heteroduplex extension in regions of near perfect sequence identity that are largely absent in low sequence identity systems. Annealing choices from all three mechanisms are handled in a straightforward manner within the free energy based scoring system [21]. In addition, the reassembly algorithm is modified to check for each of the three crossover types for every fragment annealing event.

Additional modifications were performed to improve computational performance. The family of single crossover sequences generated in the ITCHY step is much larger than that typically used for molecular breeding, so the original ***e***Shuffle program (which scales as the square of the number of parental sequences) was customized. Specifically, fragments with identical sequences from different ITCHY parents were pooled because they do not change the outcome of fragment-fragment extensions considered by the reassembly algorithm. By aggregating their concentrations instead of considering them separately, CPU times were reduced to scale linearly with the number of parental sequences. In addition, we found that for fragmentation lengths longer than 40-nt, approximating individual duplex melting curves as step functions at the duplex's melting temperature provided a tractable and accurate approximation of the annealing thermodynamics since melting temperatures for larger fragments are significantly above the applied annealing temperature. A 40-nt fragment reassembly confirmed that predictions vary by less than 5% when this approximation is utilized.

eSCRATCHY was next used to address questions concerning the preservation of prepositioned crossovers in reassembled sequences, as well as their contribution towards multiple crossover sequences in comparison with those that also occur in homology-based reassembly. In particular, the effect of fragmentation length and pairwise sequence identity on the number and positioning of crossovers produced and the relative contribution of each of the three postulated crossover mechanisms are examined.

The *purN/hGART* system mentioned above (also see reference [15]) is first examined in detail. In this case study, both in-frame and parental size selection are "idealized" so that the crossovers present in the ITCHY library are not biased in any manner. Predictions from ***e***SCRATCHY indicate that 52% of the reassembled sequences have multiple crossovers for a fragmentation length of 60-nt even though the nucleotide sequence identity is only 49% in the overlapping region. Note that even for fragments as short as 20-nt, predictions by ***e***Shuffle indicate that almost 99.9% of sequences reassembled by DNA shuffling alone will be wild-type. Interestingly, in contrast to DNA shuffling, ***e***SCRATCHY predicts that fragmentation length has little, if any, effect on the average number of crossovers produced per sequence (Figure 5.3). Smaller fragments imply that more annealing choices are available during reassembly and thus more opportunities to generate crossovers, but at the same time, a smaller proportion of fragments contain prepositioned crossovers. These two effects appear to cancel each other for systems with low sequence identity. Thus, relatively large fragments can be utilized in SCRATCHY without reducing the number of crossovers, allowing for easier purification, isolation and reassembly.

In addition, predictions suggest that neglecting hybrid-duplex crossovers in ***e***SCRATCHY would produce drastically different results, as these crossovers contribute 47% of the total number of crossovers. This "emergent" mechanism, not present in ***e***Shuffle, is almost as frequent as the prepositioned crossover mechanism. Heteroduplex crossovers are negligible as expected for a system with 49% sequence identity. The distribution of crossovers along the sequence is shown in Figure 5.4. Prepositioned crossovers are present almost uniformly along the entire sequence, showing that the unbiased nature of the ITCHY library is retained. In contrast, hybrid-duplex based crossovers track regions of high sequence identity and involve a less even distribution. Contrary to homology-based methods, the sum of all types of crossovers fills the entire sequence length with an average frequency of 0.65% per position. The "signature" of DNA shuffling can still be detected in the form of peaks tracking regions of high sequence identity.

Next, we examined the effect of pairwise sequence identity on crossover frequencies for the recombination of the following six sequences with *purN* using

eSCRATCHY and ***e***Shuffle (sequence identity with *purN* in the overlapping region in parentheses): GAR transformylases from human (49%); *Pseudomonas aeruginosa* (54%), *Pasteurella multocida* (60%), *Vibrio cholerae* (64%), *Salmonella typhimurium* (79%); and methionyl-tRNA formyltransferase from *E. coli* (33%). As seen in Figure 5.5, predictions suggest that SCRATCHY is capable of generating crossovers for all sequence pairs, regardless of sequence identity. On the other hand, DNA shuffling requires an approximate "threshold" sequence identity of 60% before any appreciable crossover generation occurs. Even for high sequence identities, we predict that SCRATCHY outperforms DNA shuffling by an average of 1.5 crossovers per sequence. Both prepositioned and hybrid-duplex crossover mechanisms remain prevalent for the entire range of sequence identities and the heteroduplex mechanism begins to contribute at identities greater than 60% (Figure 5.5). Upon utilizing parameters reflecting the specifics of the actual experimental library, ***e***SCRATCHY's predictions of the naive *purN/hGART* SCRATCHY library were reexamined and compared to the experimental data.

Section 5.3: Comparisons with Experimental Results

Section 5.3.1: Experimental SCRATCHY

Two ITCHY libraries encoding either the PurN/hGART (PGX) or the hGART/PurN (GPX) hybrid pairs were constructed (Figure 5.1a), with members distributed over the entire sample space, comparable to data from previous libraries [15, 17]. Functional selection (Figure 5.2b) was used to select for in-frame members of parental size for DNA shuffling. Although the profile of representative sequences in such a library is biased as shown in Figure 5.6, the distribution of the two directional libraries allows for multiple crossovers to occur in the overlapping region.

Equal amounts of both selected libraries (PGX and GPX) were DNA shuffled (Figure 5.1c), and the resulting reassembled sequences were amplified. The primer pair used for amplification anneals to outside portions on either side of the gene, yielding a comprehensive library of possible combinations including wild-type (wt) constructs. From this naive library, the hybrid genes of over 50 individual colonies were analyzed by DNA sequencing, and the results are summarized in Figure 5.7. For further information on the SCRATCHY protocol, see reference [22].

Analysis of the library revealed several interesting characteristics. Most importantly, a significant portion of the sample sequences had multiple crossovers. When considering the location and number of the crossover points in the sequences, an important experimental bias emerges. The majority of sequences (70%) in the library are reassembled duplicates of GPX library members, as if the library was present at a higher concentration than the PGX library during DNA shuffling.

Further examination of the sequencing data reveals a number of additional interesting features. The reassembly of parental wt-sequences in SCRATCHY, in contrast to DNA shuffling of low homology sequences, is not dominant. While few wt-PurN sequences are identified in the naive library, wt-hGART is absent. The deficiency of wt-hGART in the recombination mixture is explained by the paucity of a contiguous bridge of hGART fragments traversing the entire gene length due to the uneven distribution of fusion points in the two ITCHY libraries (Figure 5.6). The same bias, amplified by the higher effective concentration of the GPX library, is also responsible for the preponderance of hGART/PurN/hGART double-crossover sequences over PurN/hGART/PurN hybrids. Reassembly of a PurN/hGART/PurN hybrid requires both a PurN to hGART crossover at the beginning of the overlapping region and a hGART to PurN crossover near the end of the overlapping region. However, both of these crossovers occur infrequently in the starting material explaining their absence. In summary, the data show that the characteristics of the ITCHY libraries are inherited by the SCRATCHY library.

*Section 5.3.2: **e**SCRATCHY Comparisons*

Accurate *in silico* analysis required the integration of two experimental presets:

the crossover distribution of the employed ITCHY libraries and the fragment reassembly-based bias towards hGART/PurN library members. First, the uneven crossover distributions caused by the functional selection of the ITCHY libraries were accounted for in the ***e***SCRATCHY program by fitting the observed crossover data with a smooth function (Figure 5.6), thus customizing the relative concentration of each of the ITCHY library members. Second, as seen in the naive library, hGART/PurN library members dominate the reassembly process. This effect was accounted for by adjusting the concentration ratio of the two libraries to 86% GPX: 14% PGX. This ratio was calculated by examining the 5′ and 3′-termini of the library members. The relative effective concentration of the GPX library was estimated by counting the number of sequences beginning with hGART (47 sequences) and ending with PurN (39 sequences). Similarly, the PGX library estimate totaled 14 (3 + 11), resulting in the 86:14 ratio. Together, these two modifications result in crossover predictions that are in good agreement with the experimental sequence data for the naive library. The distribution matches well with what is found experimentally (Figure 5.8a). The discrepancy between the numbers of multiple crossovers predicted in the “idealized” case (Figure 5.3) and found in the experiment can be attributed to the bias in the starting material. In addition, predictions for crossover position statistics (Figure 5.8b) capture the uneven nature of crossovers found in the reassembled sequences as a result of the same bias, which also leads to an increased 3.6:1 ratio of prepositioned/hybrid-duplex crossovers compared to the “idealized” case.

Another interesting aspect is the contribution of crossovers originating from incremental truncation or homology-based recombination. Experimentally, all fusion points observed in the SCRATCHY libraries have counterparts at locations corresponding to prepositioned crossovers, originating from the ITCHY libraries. However, the origin of the crossovers in the homologous region between amino acids 100 to 110 can not conclusively been attributed to ITCHY or DNA shuffling. In the ***e***SCRATCHY model, heteroduplex crossovers are rare across the entire sequence.

Section 5.4: Summary

The ***e***SCRATCHY framework (i) led to a newly hypothesized mechanism for the generation of crossovers based on the extension of hybrid-duplexes, (ii) revealed that fragmentation length has little effect on crossover statistics, and (iii) verified complete coverage of gene length with potential crossover sites. An *in silico* case study of six pairs of parental sequences ranging in sequence identity from 33% to 79% revealed that SCRATCHY outperforms DNA shuffling by approximately 1.5 crossovers per sequence for all six sequence pairs. Comparisons of ***e***SCRATCHY statistics with experimental naive library sequence data were in good agreement after adjusting the concentration ratio of the incremental truncation libraries. Both ***e***SCRATCHY and experimental results confirmed that the crossover distributions of the incremental truncation libraries are inherited by the SCRATCHY library.

Section 5.5: References

1. Brakmann, S. (2001), "Discovery of superior enzymes by directed molecular evolution," *ChemBioChem* **2**(12): 865-871.
2. Petrounia, I.P. & Arnold, F.H. (2000), "Directed evolution of enzymatic properties," *Curr Opin Biotechnol* **11**: 325-330.
3. Schmidt-Dannert, C. (2001), "Directed evolution of single proteins, metabolic pathways, and viruses," *Biochemistry* **40**(44): 13125-13136.
4. Miyazaki, K., Wintrode, P.L., Grayling, R.A., Rubingh, D.N. & Arnold, F.H. (2000), "Directed evolution study of temperature adaptation in a psychrophilic enzyme," *J Mol Biol* **297**(4): 1015-1026.
5. Altamirano, M.M., Blackburn, J.M., Aguayo, C. & Fersht, A.R. (2000), "Directed evolution of new catalytic activity using the alpha/beta-barrel scaffold," *Nature (London)* **403**(6770): 617-622.
6. Crameri, A., Dawes, G., Rodriguez, E., Silver, S. & Stemmer, W.P.C. (1997), "Molecular evolution of an arsenate detoxification pathway by DNA shuffling," *Nature Biotechnol* **15**: 436-438.
7. Furukawa, K. (2000), "Engineering dioxygenases for efficient degradation of environmental pollutants," *Curr Opin Biotechnol* **11**: 244-249.
8. Bruhlmann, F. & Chen, W. (1999), "Tuning biphenyl dioxygenase for extended substrate specificity," *Biotechnol Bioeng* **63**(5): 544-551.
9. Wackett, L.P. (1998), "Directed Evolution of New Enzymes and Pathways for Environmental Biocatalysis," *Ann NY Acad Sci* **864**: 142-152.
10. Patten, P.A., Howard, R.J. & Stemmer, W.P.C. (1997), "Applications of DNA shuffling to pharmaceuticals and vaccines," *Curr Opin Biotechnol* **8**: 724-733.
11. Whalen, R.G., Kaiwar, R., Soong, N.W. & Punnonen, J. (2001), "DNA shuffling and vaccines," *Curr Opin Mol Ther* **3**: 31-36.

12. Marzio, G., Verhoef, K., Vink, M. & Berkhout, B. (2001), "In vitro evolution of a highly replicating, doxycycline-dependent HIV for applications in vaccine studies," *Proc Natl Acad Sci USA* **98**(11): 6342-6347.
13. Soong, N.W., Nomura, L., Pekrun, K., Reed, M., Sheppard, L., Dawes, G. & Stemmer, W.P.C. (2000), "Molecular breeding of viruses," *Nature Genet* **25**: 436-439.
14. Powell, S.K., Kaloss, M.A., Pinskstaff, A., McKee, R., Burimski, I., Pensiero, M., Otto, E., Stemmer, W.P. & Soong, N.W. (2000), "Breeding of retroviruses by DNA shuffling for improved stability and processing yield," *Nature Biotechnol* **18**: 1279-1282.
15. Ostermeier, M., Shim, J.H. & Benkovic, S.J. (1999), "A combinatorial approach to hybrid enzymes independent of DNA homology," *Nature Biotechnol* **17**: 1205-1209.
16. Ostermeier, M. & Benkovic, S.J. (2001), "Construction of Hybrid Gene Libraries Involving the Circular Permutation of DNA," *Biotech Lett* **23**: 303-310.
17. Lutz, S., Ostermeier, M. & Benkovic, S.J. (2001), "Rapid generation of incremental truncation libraries for protein engineering using alpha-phosphothioate nucleotides," *Nucleic Acids Res* **29**: E16.
18. Sieber, V., Martinez, C.A. & Arnold, F.H. (2001), "Libraries of hybrid proteins from distantly related sequences," *Nature Biotechnol* **19**(5): 456-460.
19. Ostermeier, M., Nixon, A.E. & Benkovic, S.J. (1999), "Incremental Truncation as a Strategy in the Engineering of Novel Biocatalysts," *Bioorg Med Chem* **7**: 2139-2144.
20. Moore, G.L., Maranas, C.D., Lutz, S. & Benkovic, S.J. (2001), "Predicting crossover generation in DNA shuffling," *Proc Natl Acad Sci USA* **98**(6): 3226-3231.

21. SantaLucia, J. (1998), "A unified view of polymer, dumbbell, and oligonucleotide DNA nearest neighbor thermodynamics," *Proc Natl Acad Sci USA* **95**: 1460-1465.

22. Lutz, S., Ostermeier, M., Moore, G.L., Maranas, C.D. & Benkovic, S.J. (2001), "Creating multiple crossover libraries independent of sequence identity," *Proc Natl Acad Sci USA* **98**: 11248-11253.

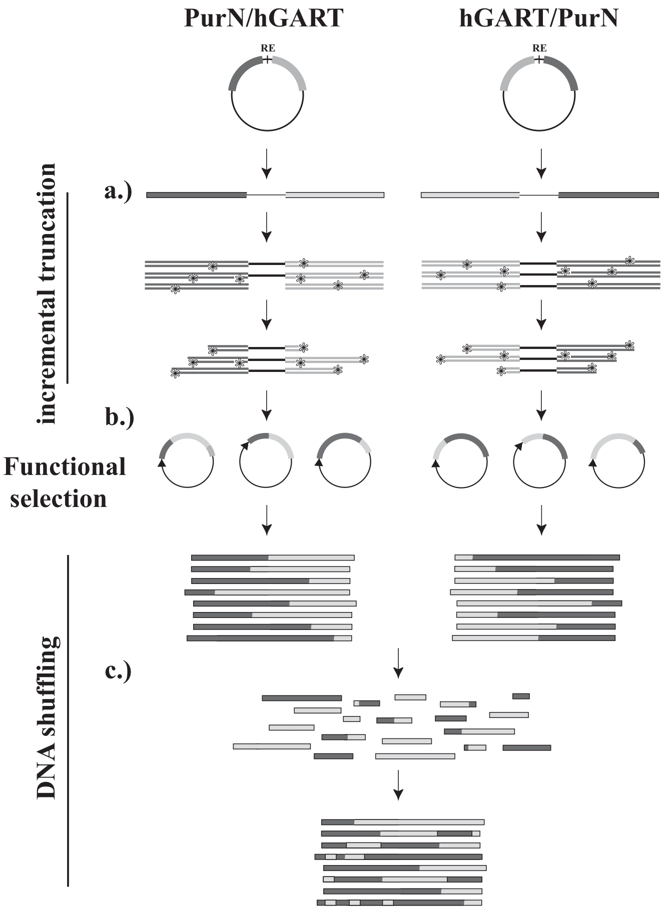


Figure 5.1: Schematic overview of SCRATCHY. Initially, individual incremental truncation (ITCHY) libraries of the two complementary constructs were created (a). Following functional selection (b) to recover in-frame hybrids of parental size, the libraries were mixed and submitted to DNA shuffling (c).

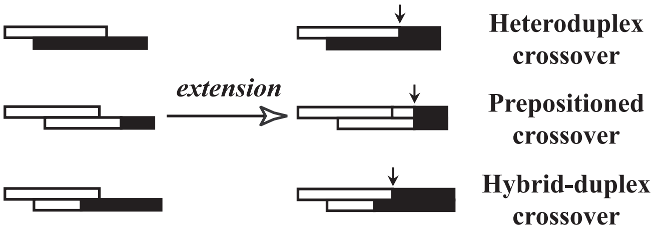


Figure 5.2: The three mechanisms for generating crossovers that are tracked *in silico*.

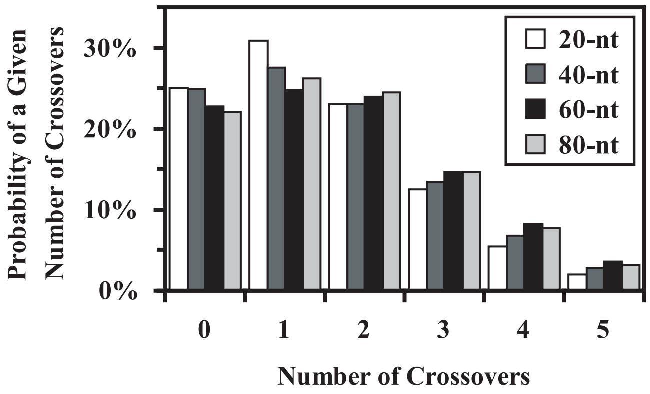


Figure 5.3: Probability that a hybrid sequence contains a given number of crossovers after the "idealized" SCRATCHY of PurN and hGART for fragmentation sizes of 20, 40, 60 and 80 nucleotides (54°C annealing temperature). Note that the distributions are similar for each of the sizes.

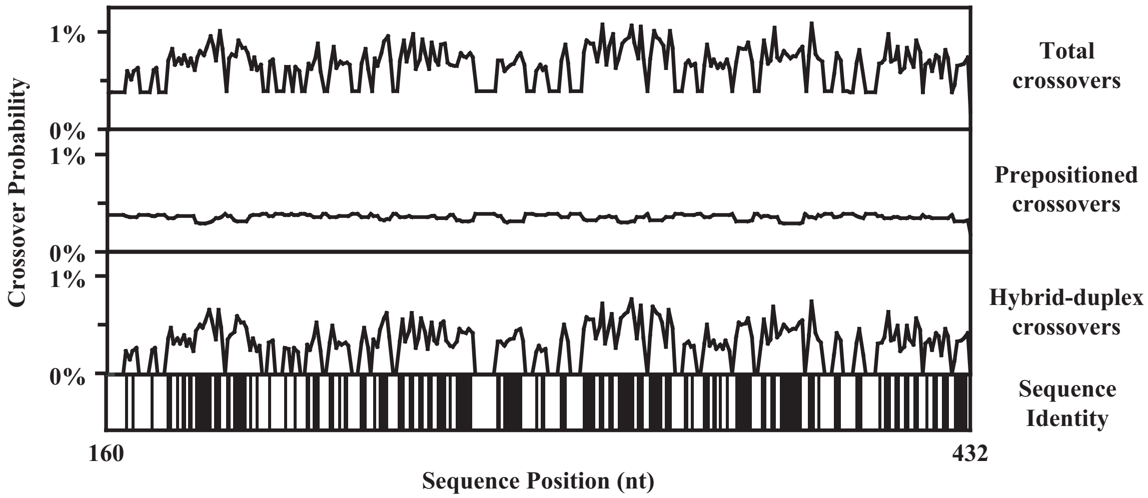


Figure 5.4: Distribution of the different types of crossovers along the sequence after the "idealized" SCRATCHY of PurN and hGART (20-nt fragments, 54°C annealing temperature). Note that no gaps appear along the entire crossover range when the crossover types are totaled. Heteroduplex crossovers are negligible and are not pictured.

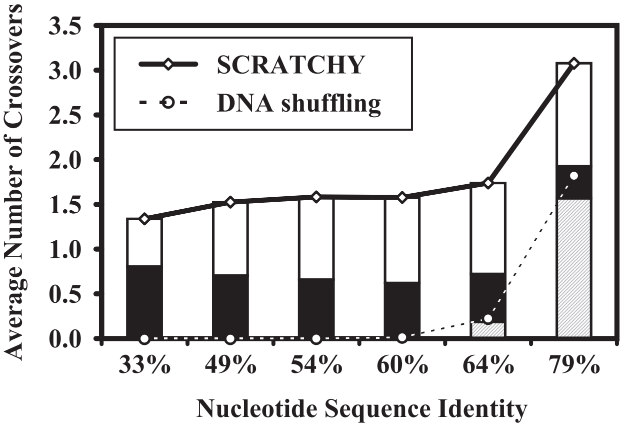


Figure 5.5: A comparison of the numbers of crossovers predicted for “idealized” SCRATCHY and DNA shuffling for sequence pairs of various sequence identities (20-nt fragments, 54°C annealing temperature). White bars represent contributions to SCRATCHY from prepositioned crossovers; black bars, hybrid-duplex crossovers; and crosshatched bars, heteroduplex crossovers.

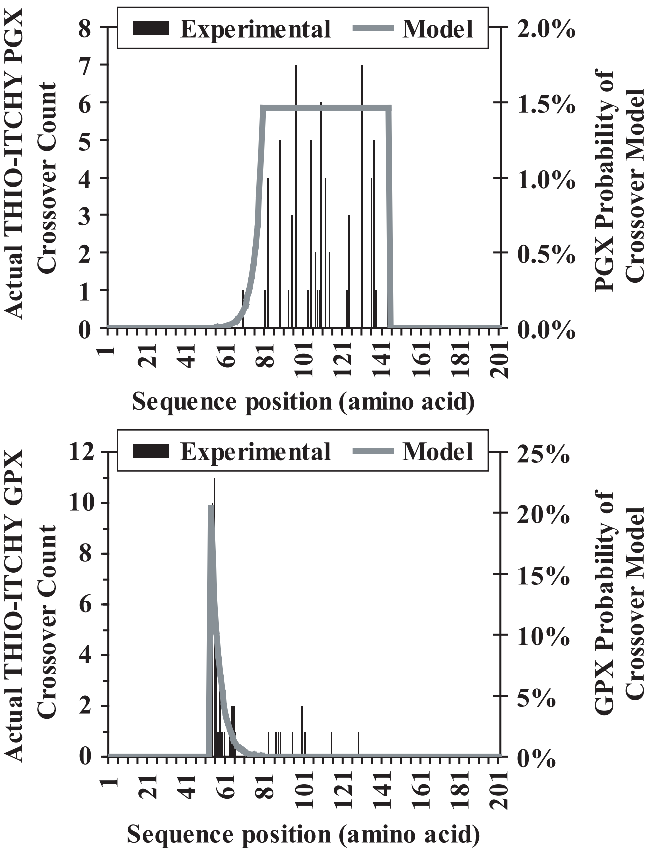


Figure 5.6: Profiles of crossover positions for the PGX and GPX libraries, including experimental counts (bars) and smooth fitted functions of crossover probability (lines).

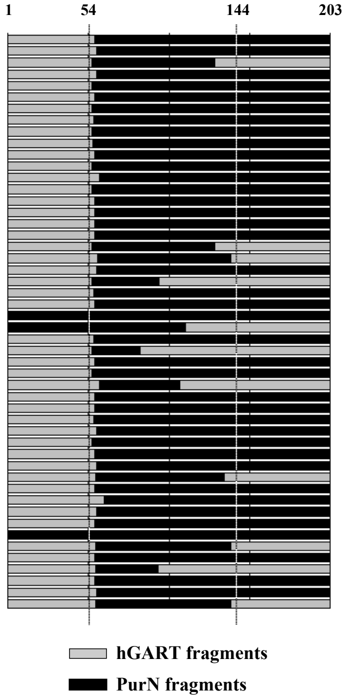


Figure 5.7: Sequence data for the naive SCRATCHY library. The dotted lines indicate the borders of the overlapping region between amino acid positions 54 and 144.

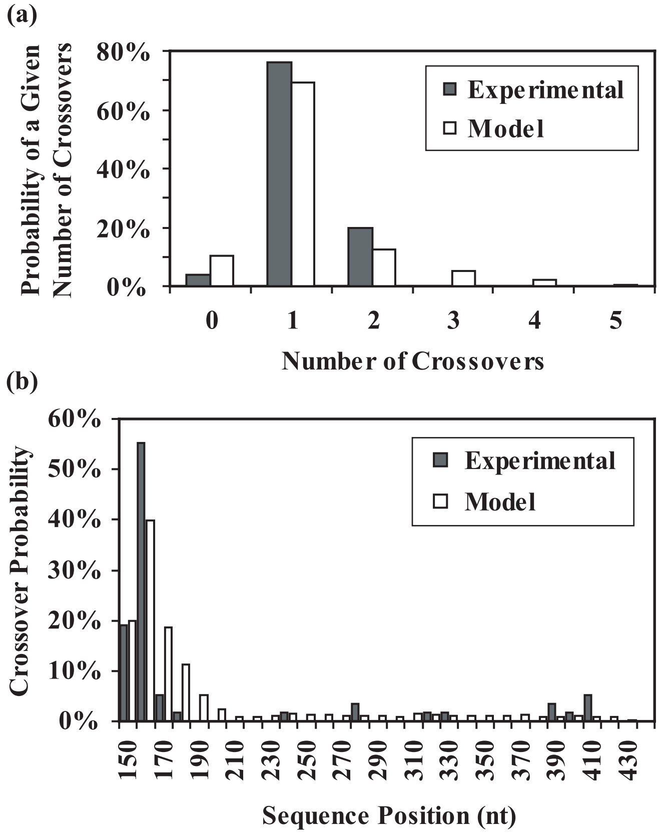


Figure 5.8: Comparing *e*SCRATCHY predictions (fragmentation length 70-nt, annealing temperature 54°C) for (a) the number of crossovers per naive library member and (b) naive library crossover positions against experimental data. In (b), data is grouped in histogram form with each bar representing a range of 10-nt.

Chapter 6: Identifying Residue-Residue Clashes in Protein Hybrids (SIRCH)

Section 6.1: Background

The use of DNA mutagenesis and/or recombination in the context of directed evolution experiments has emerged as a leading strategy in protein engineering [1-3]. However, the majority of generated protein hybrids have either substantially reduced or even completely lost functionalities. Therefore, the *a priori* classification of protein hybrids with respect to their potential to be functional is widely being recognized as an overarching challenge for many combinatorial protein engineering efforts. The majority of past successful combinatorial efforts involved the recombination of parental sequences sharing relatively high sequence identity (*i.e.*, greater than 70% at the DNA level). With the advent of a number of experimental protocols capable of recombining parental sequences with low sequence identity (*e.g.*, ITCHY/SCRATCHY [4, 5], SHIPREC [6], GeneReassembly [7]), it has been observed that the fraction of functional hybrids in the combinatorial library decreases dramatically as the level of sequence identity shared in the parental set is reduced [5, 6]. Given that most members of a protein family share pairwise sequence identities of less than 70%, this implies that a large portion of protein diversity may be left unexplored due to the scarcity of functional hybrids. This leads to the following dilemma: how can diversity generated by the recombination of low sequence identity parental sequences be effectively explored without severely curtailing the chances of success? To effectively resolve this dilemma, it is necessary to *a priori* elucidate what crossovers or crossover combinations are likely to lead to hybrids with preserved/improved functionality.

A number of hypotheses have been advanced to explain how crossovers affect the integrity of proteins. Monte Carlo simulations by Deem and co-workers [8] suggested that the swapping of low-energy structures was least disruptive to protein structure, but delineating these structures has so far not been straightforward. The SCHEMA algorithm [9] postulated structural disruption when a contacting residue pair in a hybrid does not match at least one of the parental proteins, and it was used to explain the crossover distributions found in a number of experiments. Though promising, this approach cannot differentiate between hybrids with different directionality (*i.e.*, an A-B versus a B-A crossover), which have been shown to often have very different functional crossover profiles [5].

Earlier, programs for estimating the frequency and location of crossovers in combinational DNA libraries were presented; however, in this chapter, the SIRCH (**S**econd-order mean-field **I**dentification of **R**esidue-residue **C**lashes in protein **H**ybrids) procedure for evaluating protein hybrids is described. Residue-residue clashes may arise due to a different directionality in the parental sequences with regards to a charged pair, residue sizes or hydrogen bond (see Figure 6.1), among other reasons. SIRCH consists of three steps: (i) calculation of possible rotamer-backbone, rotamer-intrinsic and rotamer-rotamer conformational energies (including van der Waals, electrostatic and solvation contributions) using atomistic representations of both the native and denatured states; (ii) use of an extended, *second-order* mean-field description to elucidate the probabilities of all possible residue-residue combinations in a minimum Helmholtz free energy ensemble; and (iii) systematic detection of clashes in potential hybrids through the evaluation of pairwise substitution patterns uncovered by the second-order mean-field description. A complete characterization of the entire collection of all possible residue-residue combinations with respect to the protein family is generated. The SIRCH procedure is used to analyze pairwise substitution patterns in the dihydrofolate reductase (DHFR) enzyme and to assess the result of the recombination of *E. coli* and human glycinamide ribonucleotide (GAR) transformylases [5, 10, 11]. Results demonstrate that experimentally determined functional crossover positions for the GAR transformylases are consistent with the predicted residue-residue clashes, capturing the effect of crossover directionality (*i.e.*, an A-B versus a B-A crossover) observed in experimental crossover distributions.

Section 6.2: Method

Section 6.2.1: Conformational Energy Calculation

Conformational energy has been widely used [12-17] as a scoring function to query whether a particular hybrid protein will likely retain functionality or whether unfavorable energetic interactions and geometric clashes brought about by recombination will prevent the hybrid from even conforming to the backbone structure. Rotamer combinations (the term *rotamer* is used here to include side-chain conformers of all residue types) are used to describe hybrid protein conformations and designs. The protein family (and fold) of interest is represented by the backbone coordinates of a single representative structure. The coordinates of the backbone atoms along with any wild-type proline residues are locked throughout the calculation (neither Pro → X nor X → Pro mutations are permitted; also, cis/trans isomerization is not allowed).

The conformational energy of a rotamer combination in the native state is expressed as the sum of: (i) rotamer/backbone energies, $e_i^{bb}(r)$, (ii) rotamer-intrinsic energies, $e_i^{int}(r)$ and (iii) rotamer/rotamer energies, $e_{ij}(rs)$. Here *i* and *j* refer to sequence positions and *r* and *s* refer to rotamer choices at positions *i* and *j* respectively. The total energy *E* of a specific combination of rotamers in the native state can be written as follows.

$$E_{combination} = \sum_{i=1}^{N} e_i(r) + \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} e_{ij}(rs) \qquad \mathbf{1}$$

where *N* represents the total number of residues in the protein and $e_i(r) = e_i^{bb}(r) + e_i^{int}(r)$. The first two terms describe rotamer/backbone and rotamer-intrinsic interaction energies, while the third term describes rotamer/rotamer interaction energies. For every position, excluding the termini (1 and *N*), a backbone-dependent (*i.e.*, on φ and ψ dihedral angles) set of rotamers is considered, in accordance with the library of Dunbrack & Cohen [13]. For the termini, a backbone-independent rotamer library [13] is used. For each sequence position, the rotamer library (excluding proline rotamers) encompasses 320 different rotamer/residue combinations. Prior to the calculation of the rotamer-backbone and rotamer-intrinsic energies, rotamers are subjected to 50 steps of conjugate gradient minimization [16] using CHARMM [18].

The CHARMM program is used along with version (22) of the protein parameters [19] to estimate conformational energies. The backbone and rotamers are represented atomistically with explicit hydrogen atoms. Three contributions to conformational energy are considered: (i) van der Waals, (ii) electrostatics (including hydrogen bonds) and (iii) solvation. Van der Waals energies are calculated on an atom-by-atom basis with a 6-12 Lennard-Jones potential. For both van der Waals and electrostatics, a cutoff distance of 14 Angstroms is used without any scaling of the 1-4 interactions. A Coulombic potential is used with a constant dielectric constant ($\varepsilon = 8$), as suggested in ref. [16]. Solvation energies are described as the sum of the solvation energies for the individual atoms in the rotamer. The solvation energy of each atom is assumed to be proportional to its accessible surface area as determined analytically by a 1.4 Angstroms probe using CHARMM. The proportionality constants of Wesson & Eisenberg [20], developed specifically for use in CHARMM, are used to estimate solvation energies based on accessible surface areas. Rotamer-rotamer solvation energies are estimated using the method of Street & Mayo [21], in which the difference in solvation energy due to the overlap of two isolated side-chains is scaled down by 50% to prevent overcounting.

The three contributions to conformational energy are employed without any empirical balancing. However, comparison of rotamers of different types can be misleading without the use of a reference energy [16]. For instance, without consideration of a reference energy, arginine residues are highly favored over other types due to their high solubility and large size. Therefore, the establishment of a reference state for each of the different residue types is necessary for providing a consistent basis of comparison. We use the “expanded” state of Elcock [22] to represent the denatured state ensemble, allowing the calculation of standardized rotamer energy differences $\delta e_i(r)$ and standardized rotamer-rotamer energy differences $\delta e_{ij}(rs)$. This representation of the denatured state has two advantages over dipeptide/tripeptide systems. First, the number and type of atoms remain constant, and second, the topology of the protein fold is retained, so that atoms that are in close proximity in the native state remain relatively close to each other in the denatured state. This procedure is described in detail in the supporting information of ref. [23]. The standardized conformational energy ΔE for a specific rotamer combination can then be written as

$$\Delta E_{combination} = \sum_{i=1}^{N} \delta e_i(r) + \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \delta e_{ij}(rs) \qquad \mathbf{2}$$

Prior to the calculation of rotamer-rotamer conformational energies, rotamers are screened out of the library if $\delta e_i(r)$ is greater than 50 kcal/mol or they are not among the ten lowest energy choices for a particular residue type [17]. Typically, about 100-120 rotamers are retained for each sequence position, encompassing all residue choices considered.

Section 6.2.2: Ensemble of Rotamer/Residue States

The objective of this study is to determine whether a residue-residue pair brought about by recombination and/or mutation is structurally favorable or unfavorable. This necessitates the establishment of the proper trade-off between structural fitness (energy) and sequence/conformational variation (entropy) characteristic of protein families. To this end, a statistical mechanics description of the residue/rotamer space of states (ensemble) is adopted. An ensemble of states is defined as the collection of all possible rotamer and residue combinations. The membership probabilities $\mathcal{P}$ of each state are found by equilibrating the ensemble. The expressions for the total energy and entropy of the ensemble, containing not only different rotamer choices but also different residue choices for each sequence position, are, as shown next, functions of the respective state probabilities $\mathcal{P}$.

$$U_{ensemble} = \sum_{\substack{\text{all rotamer}\\ \text{combinations}}} \mathcal{P}_{combination} \Delta E_{combination} \qquad \mathbf{3}$$

$$S_{ensemble} = -R \sum_{\substack{\text{all rotamer}\\ \text{combinations}}} \mathcal{P}_{combination} \ln \mathcal{P}_{combination} \qquad \mathbf{4}$$

Assuming a canonical ensemble (a closed system with constant temperature T), the state probabilities are determined at equilibrium by minimizing the Helmholtz free energy $A_{ensemble} = U_{ensemble} - TS_{ensemble}$. The use of the Helmholtz free energy allows the systematic exploration of trade-offs between conformational energy and entropy. However, the direct solution of this problem is intractable because the number of possible rotamer/residue choices is prohibitively large. For example, a 200-residue protein with 120 rotamer choices for each position gives rise to $120^{200} \approx 10^{416}$ possible rotamer combinations. Mean-field approximations are used to restore tractability to the ensemble equilibration problem.

Section 6.2.3: First-Order Mean-Field Approximation

Earlier mean-field approximations to the Helmholtz free energy [12, 24, 25], referred to herein as *first-order*, were based on the assumption that the probability $\mathcal{P}$ of a specific rotamer combination can be approximated as the product of individual rotamer *site* probabilities $p_i(r)$ of each sequence position i. This implies that the site probabilities at each position are assumed to vary independently from one another.

$$\mathcal{P}^{(1)}_{combination} = \prod_{i=1}^{N} p_i(r) \qquad \mathbf{5}$$

This simplification substantially reduces the number of state probabilities required to describe the ensemble (*e.g.*, from 10^{416} to $200 \cdot 120 = 24{,}000$ for a 200-residue protein). Substituting the first-order approximation, Equation **5**, into the expressions for the energy and entropy of a rotamer sequence (Equations **3** and **4**) leads to the following expressions for first-order mean-field energy $U^{(1)}$ and entropy $S^{(1)}$.

$$U^{(1)} = \sum_{i=1}^{N} \sum_{r \in \mathcal{R}_i} p_i(r) \delta\varepsilon_i(r) + \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \sum_{r \in \mathcal{R}_i} \sum_{s \in \mathcal{R}_j} p_i(r) p_j(s) \delta\varepsilon_{ij}(rs) \qquad \mathbf{6}$$

$$S^{(1)} = -R\sum_{i=1}^{N}\sum_{r\in\mathcal{R}_i} p_i(r)\log p_i(r) \qquad \mathbf{7}$$

where $\mathcal{R}_i$ and $\mathcal{R}_j$ represent the set of rotamer choices available at positions i and j respectively. Minimization of the first-order mean-field free energy $A^{(1)} = U^{(1)} - TS^{(1)}$, subject to the condition that the site probabilities sum to one ($\sum_r p_i(r) = 1$), yields

$$p_i(r) = \frac{\exp(-mfe_i(r)/RT)}{\sum\limits_{r'\in\mathcal{R}_i}\exp(-mfe_i(r')/RT)}, \forall\, i, r \in \mathcal{R}_i \qquad \mathbf{8}$$

$$mfe_i(r) = \delta e_i(r) + \sum_{\substack{j=1\\ j\neq i}}^{N}\sum_{s\in\mathcal{R}_j} p_j(s)\delta e_{ij}(rs), \forall\, i, r \in \mathcal{R}_i \qquad \mathbf{9}$$

The mean-field energy $mfe_i(r)$ can be thought of as the energy of rotamer r placed at sequence position i plus the average interaction energy that it experiences from other rotamer choices s at other positions j in the ensemble. As shown in Equation **8**, the site probabilities are Boltzmann-distributed with respect to their mean-field energies. Typical solution procedures involve uniform initialization of the rotamer probabilities and iterative calculation of the mean-field energies (Equation **9**) and site probabilities (Equation **8**) until self-consistency is achieved [12, 24-27]. Koehl & Delarue [24] and Lee [25] used a first-order mean-field approach for estimating the conformational entropy of side-chains and positioning them. Voigt *et al.* [12] and Saven and co-workers [17, 28] extended the ensemble to include both residue and rotamer choices to investigate the fitness of single residue substitutions in mutagenesis experiments.

A key limitation of the first-order mean-field approximation is that it cannot capture whether and/or how the substitution patterns at two sequence positions i and j are related. Therefore, no information can be gleaned as to how a site probability distribution at one position is influenced by placing a specific rotamer at another position (*i.e.*, conditional probability). However, this is exactly the type of information needed to evaluate the impact of bringing together two new sets of residues in hybrids generated by recombination. To overcome these limitations, a second-order mean-field approximation to the Helmholtz free energy is developed that allows for the explicit consideration of rotamer-rotamer *joint* probabilities.

Section 6.2.4: Second-Order Mean-Field Approximation

A second-order approximation is proposed that can explicitly track joint probabilities, represented by $P_{ij}(rs)$. The Bethe approximation [29] is used to estimate the ensemble probability $\mathcal{P}$ as the product of all joint probabilities, appropriately scaled to avoid double-counting.

$$\mathcal{P}^{(2)}_{combination} = \prod_{i=1}^{N-1}\prod_{j=i+1}^{N} P_{ij}(rs) \Big/ \prod_{i=1}^{N} p_i(r)^{N-2} \qquad \textbf{10}$$

The Bethe approximation was originally developed to assess the entropy within metallic superlattices [29, 30], but in recent years it has been applied in the field of computer vision [31] and has been shown to be analogous to the use of belief propagation methods [32] in resolving Bayesian causal networks [33].

Substituting the second-order mean-field approximation (Equation **10**) into the equations for ensemble energy (Equation **3**) and entropy (Equation **4**) leads to the following expressions.

$$U^{(2)} = \sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\sum_{r\in\mathcal{R}_i}\sum_{s\in\mathcal{R}_j} P_{ij}(rs)\big(\delta e_{ij}(rs) + \delta e_i(r) + \delta e_j(s)\big) - (N-2)\sum_{i=1}^{N}\sum_{r\in\mathcal{R}_i} p_i(r)\,\delta e_i(r) \qquad \textbf{11}$$

$$S^{(2)} = -R\left[\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\sum_{r\in\mathcal{R}_i}\sum_{s\in\mathcal{R}_j} P_{ij}(rs)\ln P_{ij}(rs) - (N-2)\sum_{i=1}^{N}\sum_{r\in\mathcal{R}_i} p_i(r)\ln p_i(r)\right] \qquad \textbf{12}$$

As described earlier, the minimization of the ensemble free energy for the first-order mean-field approximation can readily be converted into a recursive relation resolved through direct substitution. Such a conversion for a second-order mean-field approximation is much more elusive. To accomplish this, a set of variable transformations is needed. First, the energy expression can be written in a form analogous to that of the entropy by substituting $\phi_i(r) = \exp(-\delta e_i(r)/RT)$ and $\psi_{ij}(rs) = \exp(-\delta e_{ij}(rs)/RT)$ into the expressions for the second-order energy and entropy (Equations **11** and **12**). By combining the resulting expressions via $A^{(2)} = U^{(2)} - TS^{(2)}$, the following expression for the Bethe free energy (scaled by RT) is derived.

$$\begin{aligned}\frac{A^{(2)}}{RT} &= \sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\sum_{r\in\mathcal{R}_i}\sum_{s\in\mathcal{R}_j} P_{ij}(rs)\left(\ln P_{ij}(rs) - \ln(\psi_{ij}(rs)\phi_i(r)\phi_j(s)\right) \\ &\quad -(N-2)\sum_{i=1}^{N}\sum_{r\in\mathcal{R}_i} p_i(r)\left(\ln p_i(r) - \ln\phi_i(r)\right)\end{aligned} \qquad \mathbf{13}$$

The joint probabilities $P_{ij}(rs)$ are then equilibrated in the ensemble by minimizing $A^{(2)}/RT$, subject to:

$$\sum_{r\in\mathcal{R}_i} p_i(r) = 1, \forall\, i \qquad \mathbf{14}$$

$$\sum_{r\in\mathcal{R}_i}\sum_{s\in\mathcal{R}_j} P_{ij}(rs), \forall\, i, j > i \qquad \mathbf{15}$$

$$\sum_{s\in\mathcal{R}_j} P_{ij}(rs) = p_i(r), \forall\, i, j \neq i, r \in \mathcal{R}_i \qquad \mathbf{16}$$

Equations **14** and **15** ensure that both the site and joint probability choices sum to one for a given position or pair of positions respectively, while Equation **16** ensures consistency between joint probabilities and respective site probabilities. The dimensionality of the resulting nonlinear optimization problem is too high to allow for direct numerical solution. For example, for a 200-residue protein, more than 10^8 probability variables are present. To remedy this, we employ the method of Lagrangean multipliers for converting a constrained nonlinear optimization problem into a system of nonlinear algebraic equations. The Lagrangean function $\mathcal{L}$ is formed by augmenting the original function $A^{(2)}/RT$ by adding all three constraints to the objective function with multipliers γ_i, Γ_{ij}, and $\lambda_{ji}(r)$, respectively.

$$\begin{aligned}\mathcal{L} = &\ \frac{A^{(2)}}{RT} + \sum_{i=1}^{N}\gamma_i\left(1 - \sum_{s\in\mathcal{R}_j} p_i(r)\right) \\ &+ \sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\Gamma_{ij}\left(1 - \sum_{r\in\mathcal{R}_i}\sum_{s\in\mathcal{R}_j} P_{ij}(rs)\right) \\ &+ \sum_{i=1}^{N}\sum_{\substack{j=1\\ j\neq i}}^{N}\sum_{r\in\mathcal{R}_i}\lambda_{ji}(r)\left(p_i(r) - \sum_{s\in\mathcal{R}_j} P_{ij}(rs)\right)\end{aligned} \qquad \mathbf{17}$$

Minima of $\mathcal{L}$ are located at points where derivatives with respect to each of the variables (*i.e.*, rotamer probabilities and multipliers) are equal to zero. Setting $\partial\mathcal{L}/\partial p_i(r) = 0$ yields

$$p_i(r) = z_i \phi_i(r) \exp\left(\sum_{\substack{j=1 \\ j \neq i}}^{N} \frac{\lambda_{ji}(r)}{N-2} \right), \forall\, i, r \in \mathcal{R}_i \qquad \mathbf{18}$$

where z_i is chosen so as to normalize $p_i(r)$ (Equation **14**). Similarly, $\partial\mathcal{L}/\partial p_{ij}(rs) = 0$ provides

$$P_{ij}(rs) = Z_{ij} \psi_{ij}(rs) \phi_i(r) \phi_j(s) \exp\left(\lambda_{ji}(r) + \lambda_{ij}(s)\right), \forall\, i, j > i, r \in \mathcal{R}_i, s \in \mathcal{R}_j \qquad \mathbf{19}$$

where Z_{ij} enforces the normalization of $P_{ij}(rs)$ (Equation **15**).

Note that when the derivatives of $\mathcal{L}$ with respect to the multipliers are set to zero, the original three constraints (Equations **14-16**) are recovered. The set of five nonlinear equations (Equations **14-16, 18,** and **19**) is recast further by substituting message variables $m_{ij}(s)$ for multipliers $\lambda_{ij}(s)$.

$$\lambda_{ij}(s) = \ln \prod_{\substack{k=1 \\ k \neq i,j}}^{N} m_{kj}(s), \forall\, i, j \neq i, s \in \mathcal{R}_j \qquad \mathbf{20}$$

This variable substitution is motivated by methods used to resolve Bayesian networks by belief propagation [32]. The message variables $m_{ij}(s)$ describe how the set of rotamer choices at position i interacts with the choice of rotamer s at position j, providing the following expression for $p_i(r)$:

$$p_i(r) = z_i \phi_i(r) \prod_{\substack{j=1 \\ j \neq i}}^{N} m_{ji}(r), \forall\, i, r \in \mathcal{R}_i \qquad \mathbf{21}$$

An expression for $P_{ij}(rs)$ is derived in a similar fashion.

$$P_{ij}(rs) = Z_{ij} \psi_{ij}(rs) \phi_i(r) \phi_j(s) \left(\prod_{\substack{k=1 \\ k \neq i,j}}^{N} m_{ki}(r) m_{kj}(s) \right), \forall\, i, j > i, r \in \mathcal{R}_i, s \in \mathcal{R}_j \qquad \mathbf{22}$$

Equations **21** and **22** are then combined via Equation **16** to derive a recursion, also known as belief propagation, containing only the message variables.

$$m_{ij}(s) = \sum_{r \in \mathcal{R}_i} \phi_i(r) \psi_{ij}(rs) \left(\prod_{\substack{k=1 \\ k \neq i,j}}^{N} m_{ki}(r) \right), \forall\, i, j \neq i, s \in \mathcal{R}_j \qquad \textbf{23}$$

Three factors are considered in the belief propagation recursion: (i) how rotamer *r* at position *i* fits with rotamer *s* at position *j* $\psi_{ij}(rs)$, (ii) how rotamers *r* at position *i* fit the backbone $\sum_r \phi_i(r)$, and (iii) how other positions *k* interact with rotamer *r* at position *i* $\Pi_k\, m_{ki}(r)$. Self-consistent resolution of this recursion yields values for the message variables, which are then substituted into Equations **21** and **22** to calculate the site and joint probabilities. Site and joint probabilities for specific residues *a* and residue pairs *a,b* are examined by aggregating the corresponding rotamer probabilities (where $\mathcal{R}_i^a$ represents the set of rotamers of residue type *a* available at position *i*).

$$p_i(a) = \sum_{r \in R_i^a} p_i(r); \qquad P_{ij}(ab) = \sum_{r \in R_i^a} \sum_{s \in R_j^b} P_{ij}(rs) \qquad \textbf{24}$$

A flowchart summarizing the steps of the complete computational procedure is shown in Figure 6.2. With the second-order mean-field approximation in place, the correct temperature of the ensemble is estimated by matching the entropy of the natural Pfam [34] protein family to the entropy of the ensemble (see supporting information of ref. [23] for details).

Section 6.2.5: Substitution Dependency D_{ij}

The identified site and joint ensemble probabilities are used to determine the tolerance of the protein structure, or lack thereof, for different residue combinations. Residue pairs that are favorable or unfavorable can be identified by examining the probability ratio $\alpha_{ij}(ab)$ that quantifies the departure of the joint probabilities from the independent substitution assumption. Specifically,

$$\alpha_{ij}(ab) = \frac{P_{ij}(ab)}{p_i(a) p_j(b)} \begin{cases} > 1, \text{ } a \text{ and } b \text{ are favored at } i,j \\ < 1, \text{ } a \text{ and } b \text{ are unfavored at } i,j \\ = 1, \text{ no preference} \end{cases} \qquad \textbf{25}$$

The standard deviation of $\alpha_{ij}(ab)$ over all residue combinations provides a quantitative metric for the *substitution dependency D_{ij}*:

$$D_{ij} = \left[\sum_{a=1}^{20} \sum_{b=1}^{20} P_{ij}(ab) \left(\log_2 \alpha_{ij}(ab) - \mu_{ij} \right)^2 \right]^{1/2} \qquad \textbf{26}$$

$$\text{where } \mu_{ij} = \sum_{a=1}^{20} \sum_{b=1}^{20} P_{ij}(ab) \log_2 \alpha_{ij}(ab)$$

A zero value for the substitution dependency D_{ij} implies that residue positions *i* and *j* have independent substitution patterns. Non-zero (positive) values for D_{ij} signify correlation in the substitution patterns. The larger the value of D_{ij}, the stronger the correlation is between positions *i* and *j*. The substitution dependency metric D_{ij} along with the probability ratios $\alpha_{ij}(ab)$ can be used not only for elucidating substitution correlation between two residue positions but also for querying whether residue pairs in a protein hybrid comply or clash with the family protein structure in comparison to the parental sequences.

Section 6.3: Correlation in the Substitution Patterns of the DHFR Protein Family

The well-studied dihydrofolate reductase (DHFR) protein family is first addressed to examine whether well known correlated substitution patterns can be revealed by SIRCH. The substitution dependencies D_{ij} based on four different DHFR crystals (*i.e.*, *E. coli:* 1rx2, M20 closed [35], 1rx5, M20 occluded [35], 1ra9, M20 open [35]; and *Lactobacillus casei:* 3dfr, M20 closed [36]) downloaded from the Protein Data Bank [37] are calculated. The first three crystals are snapshots of important steps in the *E. coli* DHFR catalytic cycle [35], while the fourth is a non-*E. coli* DHFR. Figure 6.3 depicts the substitution dependency plots for the four structures. The plots are almost identical, demonstrating that the choice of crystal does not substantially alter the results. The only significant difference is between the results for the *open* M20 structure (1ra9) and the two *closed* structures (1rx2, 3dfr). Specifically, for the closed structures, residues 25-50 exhibit a more pronounced substitution dependency. This is consistent with the fact that in the closed conformation residues 25-50 are approached by the M20 loop and other connecting residues.

In the residue-residue substitution dependency plot for 1rx2 (Figure 6.4a), blue implies no correlation whereas green, yellow, orange and red depict residue pairs with increased levels of correlation in substitution patterns. Interestingly, strong correlation between the contacting M20 and FG loops (*i.e.*, residues 7-24 and 116-132 respectively) as well as between the end of the M20 loop (residues 20-25) and the GH loop (residues 142-150) is correctly predicted. Quite remarkably, strong correlation between the M20/Hinge region (20-38) with both the region from residues 45-50 and the region from residues 93-97 is also elucidated even though these domains are not contacting (distance greater that 8 Angstroms), alluding to the fact that correlation information appears to be propagated through a network of interacting residues. The ability of the method to capture distal correlations in substitution patterns is shown more clearly in Figures 6.4b and c, in which the substitution dependency density plot is contrasted against the set of contacting residues. It appears that important correlation information between residue pairs is encoded within D_{ij} that does not necessarily require them to be contacting. Another important observation involves a comparison of the residue pairs that exhibit correlated motion (in the same direction), based on the molecular dynamics study of Radkiewicz & Brooks [38], and the substitution dependency plot (see Figure 6.5). The strong similarity between the two alludes that residues that “move” in the same direction must also be substituted in a coordinated manner.

Next, the *a priori* classification of crossovers with respect to their functionality through SIRCH is addressed. This is accomplished by contrasting the experimental results for the *E. coli* and human GAR transformylase system with the model predictions.

Section 6.4: *In silico* GAR Transformylase Hybrid Prescreening

By using the structure of *E. coli* glycinamide ribonucleotide (GAR) transformylase (Protein Data Bank code 1gar [39]) as a reference, SIRCH is used to characterize all single crossover hybrids between *E. coli* and human versions of GAR transformylase (protein sequence identity of 45%). The locations of all functional crossovers in bidirectional hybrids generated through incremental truncation [5, 10, 11] are depicted as vertical bars in Figure 6.6. The incremental truncation window is between residues 50 to 150. Clearly, functional crossovers are distributed quite differently depending on the directionality of the incremental truncation library (compare Figure 6.6a and b).

Residue-residue clashes predicted for single-crossover hybrids are shown pictorially as arcs of different color linking the corresponding residues (see Figure 6.6). These clashes are only present in hybrids with a crossover positioned between the two residues (*i.e.*, cutting the arc). The severity of the clash is quantified by contrasting the hybrid residue pair probability ratio against the probability ratios corresponding to the two parental (wild-type) sequences (*i.e.*, *E. coli* and human). By using the parental residue pairs as a baseline, the comparison only reveals clashes generated in the hybrid that are absent in the parental sequences. Blue arcs signify a relatively small difference in probability ratio between the hybrid and the parental sequences whereas orange and red arcs denote clashes of increasing intensity based on the hybrid/parental sequence probability ratio difference. For the human/*E. coli* library (Figure 6.6a), a large cluster of functional crossovers is present at the beginning of the recombination range, followed by an abrupt end at position 66. Remarkably, position 66 is the location of the first residue for the first clash in the recombination window. Past the first clashing pair, a few functional crossovers are present that again disappear after encountering a pair of nested clashes. Unlike the human/*E. coli* library, no functional crossovers are present at the beginning of the recombination range for the *E. coli*/human library (Figure 6.6b), which is consistent with the numerous clashes found within the range 54-77. A large number of functional crossovers (81-115) violate only a mild clash, whereas the group between positions 125-150 is inconsistent with a severe clash between residues 119 and 162. Molecular modeling for these two positions reveals a steric hindrance between histidine and valine that cannot be relieved without substantial backbone movement. In this case, it appears that this movement did not affect catalytic activity or binding affinity, pointing at some of the limitations of mean-field based approximation techniques. Overall, SIRCH appears to be quite successful, though not perfect, at classifying crossovers in terms of their potential to yield functional hybrids. More importantly, by identifying a relatively small set of clashing residue combinations, SIRCH provides valuable information for designing strategies based on site-directed mutagenesis for relieving these clashes.

Section 6.5: Summary

In this paper, a second-order mean-field approach was described for the complete description of the entire residue substitution space of a protein family. The procedure was implemented in the SIRCH program for identifying and quantifying the severity of residue-residue clashes in protein hybrids. This information can then be used to suggest site-directed mutagenesis strategies for (i) the parental sequences and/or (ii) hybrids with residual functionalities that will lead to the reduction or elimination of clashes in the protein combinatorial library. Note that the obtained results appear to be largely insensitive to the starting protein crystal and that a strong correlation between residue substitution dependency patterns and residue motions in the crystal was observed.

Computational results uncovered correlated substitution patterns for the DHFR family, not only between contacting but also between widely separated domains, alluding to the propagation of residue substitution correlation information through a network of interacting residues [40]. In addition, the distribution of functional crossovers for the incremental truncation libraries [5, 10, 11] of *E. coli*/human GAR and human/*E. coli* GAR transformylases was in very good agreement with the residue-residue clashes revealed by SIRCH. These results are currently being used to identify site-directed mutagenesis strategies for ratcheting up the functionality of barely active hybrids. So far, the only information gleaned from the sequence data of protein families [37] involved setting the entropy of the computationally equilibrated ensemble. Nevertheless, additional restrictions can be imported into the ensemble by appending appropriate equality or even inequality constraints. These constraints may, for example, fix the consensus active site residues, restrict the fraction of charged residues present in the library or establish hydrophobic/polar patterning requirements.

Section 6.6: References

1. Petrounia, I.P. & Arnold, F.H. (2000), "Designed evolution of enzymatic properties," *Curr Opin Biotechnol* **11**: 325-330.
2. Brakmann, S. (2001), "Discovery of superior enzymes by directed molecular evolution," *ChemBioChem* **2**: 865-871.
3. Schmidt-Dannert, C. (2001), "Directed evolution of single proteins, metabolic pathways, and viruses," *Biochemistry* **40**: 13125-13136.
4. Ostermeier, M., Shim, J.H. & Benkovic, S.J. (1999), "A combinatorial approach to hybrid enzymes independent of DNA homology," *Nature Biotech* **17**: 1205-1209.
5. Lutz, S., Ostermeier, M., Moore, G.L., Maranas, C.D. & Benkovic, S.J. (2001), "Creating multiple-crossover DNA libraries independent of sequence identity," *Proc Natl Acad Sci USA* **98**: 11248-11253.
6. Sieber, V., Martinez, C.A. & Arnold, F.A. (2001), "Libraries of hybrid proteins from distantly related sequences," *Nature Biotech* **19**: 456-460.
7. Short, J.M. (1999), "US5,965,408: Method of DNA Reassembly by Interrupting Synthesis."
8. Bogarad, L.D. & Deem, M.W. (1999), "A hierarchical approach to protein molecular evolution," *Proc Natl Acad Sci USA* **96**: 2591-2595.
9. Voigt, C.A., Martinez, C., Wang, Z.-G., Mayo, S.L. & Arnold, F.H. (2002), "Protein building blocks preserved by recombination," *Nature Struct Biol* **9**: 553-558.
10. Lutz, S., Ostermeier, M. & Benkovic, S.J. (2001), "Rapid generation of incremental truncation libraries for protein engineering using alpha-phosphothioate nucleotides," *Nucleic Acids Res* **29**: 16.

11. Ostermeier, M., Nixon, A.E. & Benkovic, S.J. (1999), "Incremental truncation as a strategy in the engineering of novel biocatalysts," *Bioorg Med Chem* **7**: 2139-2144.

12. Voigt, C.A., Mayo, S.L., Arnold, F.H. & Wang, Z.-G. (2001), "Computational method to reduce the search space for directed protein evolution," *Proc Natl Acad Sci USA* **98**: 3778-3783.

13. Dunbrack Jr., R.L. & Cohen, F.E. (1997), "Bayesian statistical analysis of protein sidechain rotamer preferences," *Prot Sci* **6**: 1661-1681.

14. Koehl, P. & Levitt, M. (1999), "De Novo Protein Design. I. In Search of Stability and Specificity," *J Mol Biol* **293**: 1161-1181.

15. Raha, K., Wollacott, A.M., Italia, M.J. & Desjarlais, J.R. (2000), "Prediction of amino acid sequence from structure," *Prot Sci* **9**: 1106-1119.

16. Wernisch, L., Hery, S. & Wodak, S.J. (2000), "Automatic Protein Design with All Atom Force-fields by Exact and Heuristic Optimization," *J Mol Biol* **301**: 713-736.

17. Kono, H. & Saven, J.G. (2001), "Statistical Theory for Protein Combinatorial Libraries. Packing Interactions, Backbone Flexibility, and the Sequence Variability of a Main-chain Structure," *J Mol Biol* **306**: 607-628.

18. Brooks, B., Bruccoleri, R., Olafson, B., States, D., Swaminathan, S. & Karplus, M. (1983), "CHARMM: a program for macromolecular energy, minimisation, and dynamics calculations," *J Comput Chem* **4**: 187-217.

19. MacKerell Jr., A.D., Bashford, D., Bellott, M., Dunbrack Jr., R.L., Evanseck, J.D., Field, M.J., Fischer, S., Gao, J., Guo, H., Ha, S., Joseph-McCarthy, D., Kuchnir, L., Kuczera, K., Lau, F.T.K., Mattos, C., Michnick, S., Ngo, T., Nguyen, D.T., Prodhom, B., Reiher III, W.E., Roux, B., Schlenkrich, M., Smith, J.C., Stote, R., Straub, J., Watanabe, M., Wiorkiewicz-Kuczera, J., Yin, D. & Karplus,

M. (1998), "All-atom Empirical Potential for Molecular Modeling and Dynamics Studies of Proteins," *J Phys Chem B* **102**: 3586-3616.

20. Wesson, L. & Eisenberg, D. (1992), "Atomic solvation parameters applied to molecular dynamics of proteins in solution," *Prot Sci* **1**: 227-235.
21. Street, A.G. & Mayo, S.L. (1998), "Pairwise calculation of protein solvent-accessible surface areas," *Fold Des* **3**: 253-258.
22. Elcock, A.H. (1999), "Realistic Modeling of the Denatured States of Proteins Allows Accurate Calculations of the pH Dependence of Protein Stability," *J Mol Biol* **294**: 1051-1062.
23. Moore, G.L. & Maranas, C.D. (2003), "Identifying residue-residue clashes in protein hybrids by using a second-order mean-field approach," *Proc Natl Acad Sci U S A* **100**(9): 5091-5096.
24. Koehl, P. & Delarue, M. (1994), "Application of a self-consistent mean field theory to predict protein side-chains conformation and estimate their conformational entropy," *J Mol Biol* **239**: 249-275.
25. Lee, C. (1994), "Predicting Protein Mutant Energetics by Self-consistent Ensemble Optimization," *J Mol Biol* **236**: 918-939.
26. Koehl, P. & Delarue, M. (1995), "A self consistent mean field approach to simultaneous gap closure and side-chain positioning in homology modeling," *Nat Struct Biol* **2**: 163-170.
27. Vasquez, M. (1995), "An Evaluation of Discrete and Continuum Search Techniques for Conformational Analysis of Side Chains in Proteins," *Biopolymers* **36**: 53-70.
28. Zou, J. & Saven, J.G. (2000), "Statistical Theory of Combinatorial Libraries of Folding Proteins: Energetic Discrimination of a Target Structure," *J Mol Biol* **296**: 281-294.

29. Bethe, H.A. (1935), "Statistical Theory of Superlattices," *Proc Royal Soc of London A* **150**: 552-575.

30. Pathria, R.K., *Statistical Mechanics*. 1996: Butterworth-Heinemann.

31. Freeman, W.T., Pasztor, E.C. & Carmichael, O.T. (2000), "Learning Low-Level Vision," *Int J Comp Vis* **40**: 25-47.

32. Pearl, J., *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. 1988: Morgan Kaufmann Publishers, Inc.

33. Yedidia, J.S., *Advanced Mean Field Methods: Theory and Practice*. 2001: The MIT Press.

34. Bateman, A., Birney, E., Cerruti, L., Durbin, R., Etwiller, L., Eddy, S.R., Griffiths-Jones, S., Howe, K.L., Marshall, M. & Sonnhammer, E.L. (2002), "The Pfam Protein Families Database," *Nucleic Acids Res* **20**: 276-280.

35. Sawaya, M.R. & Kraut, J. (1997), "Loop and Subdomain Movements in the Mechanism of Escherichia coli Dihydrofolate Reductase: Crystallographic Evidence," *Biochemistry* **36**: 586-603.

36. Bolin, J.T., Filman, D.J., Matthews, D.A., Hamlin, R.C. & Kraut, J. (1982), "Crystal structures of Escherichia coli and Lactobacillus casei dihydrofolate reductase refined at 1.7 &#197; resolution," *J Biol Chem* **257**: 13650-13662.

37. Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N. & Bourne, P.E. (2000), "The Protein Data Bank," *Nucleic Acids Res* **28**: 235-242.

38. Radkiewicz, J.L. & III, C.L.B. (2000), "Protein Dynamics in Enzymatic Catalysis: Exploration of Dihydrofolate Reductase," *J Am Chem Soc* **122**: 225-231.

39. Klein, C., Chen, P., Arevalo, J.H., Stura, E.A., Marolewski, A., Warren, M.S., Benkovic, S.J. & Wilson, I.A. (1995), "Towards structure-based drug design: crystal structure of a multisubstrate adduct complex of glycinamide

ribonucleotide transformylase at 1.96 &#197; resolution," *J Mol Biol* **249**: 153-175.

40. Agarwal, P.K., Billeter, S.R., Rajagopalan, P.T., Benkovic, S.J. & Hammes-Schiffer, S. (2002), "Network of coupled promoting motions in enzyme catalysis," *Proc Natl Acad Sci USA* **99**: 2794-2799.

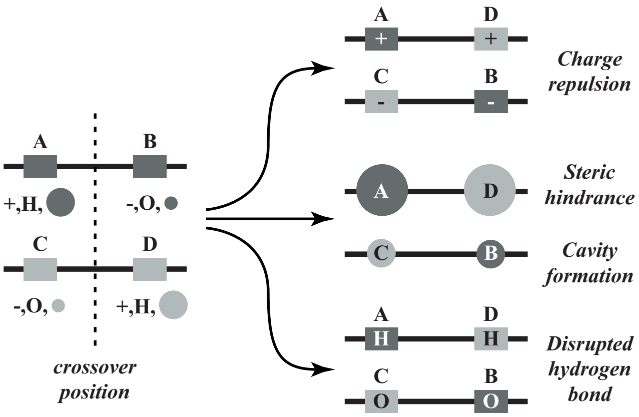


Figure 6.1: Residue-residue clashes may arise in protein hybrids due to a different directionality in the parental sequences of a charged pair, residue sizes or hydrogen bond (H represents proton donor; O, proton acceptor). Upon recombination this leads to a charge-charge repulsion, steric hindrance, cavity formation or a disrupted hydrogen bond.

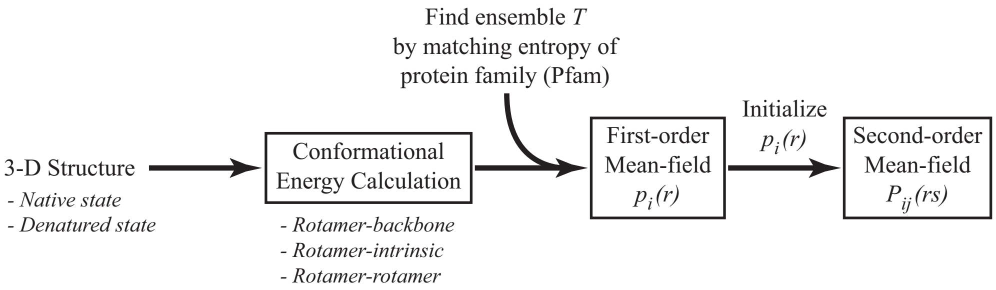


Figure 6.2: First, the backbone coordinates of a crystal belonging to the protein family of interest are downloaded. Next, the complete table of rotamer-backbone, rotamer-intrinsic and rotamer-rotamer conformational energies are calculated. Based on these energies, a first-order mean-field calculation is used to initialize the site probabilities $p_i(r)$ for the final second-order mean-field calculation that identifies the joint probabilities $P_{ij}(rs)$. Specifically, the second-order mean-field calculation requires the following steps:

Step 1. Initialize the message variables with the first-order mean-field rotamer probabilities $p_i^{(1)}(r)$:

$$\{m_{ij}(s)\}^0 \to \left(p_j^{(1)}(s)/\phi_j(s)\right)^{1/(N-1)}$$

Step 2. Evaluate the belief propagation recursion (Equation **24**) to calculate updated messages:

$$\{m_{ij}(s)\}^{k+1} \to \left\{\sum_{r\in\mathcal{R}_i}\psi_{ij}(rs)\phi_i(r)\left(\prod_{\substack{k=1\\k\neq i,j}}^{N} m_{ki}(r)\right)\right\}^{k}$$

Step 3. Check for convergence:

If $\max\limits_{i,j,s}\left(\left|\{m_{ij}(s)\}^{k+1}-\{m_{ij}(s)\}^{k}\right| \Big/ \{m_{ij}(s)\}^{k}\right) < 10^{-3}$, terminate. Otherwise return to Step 2.

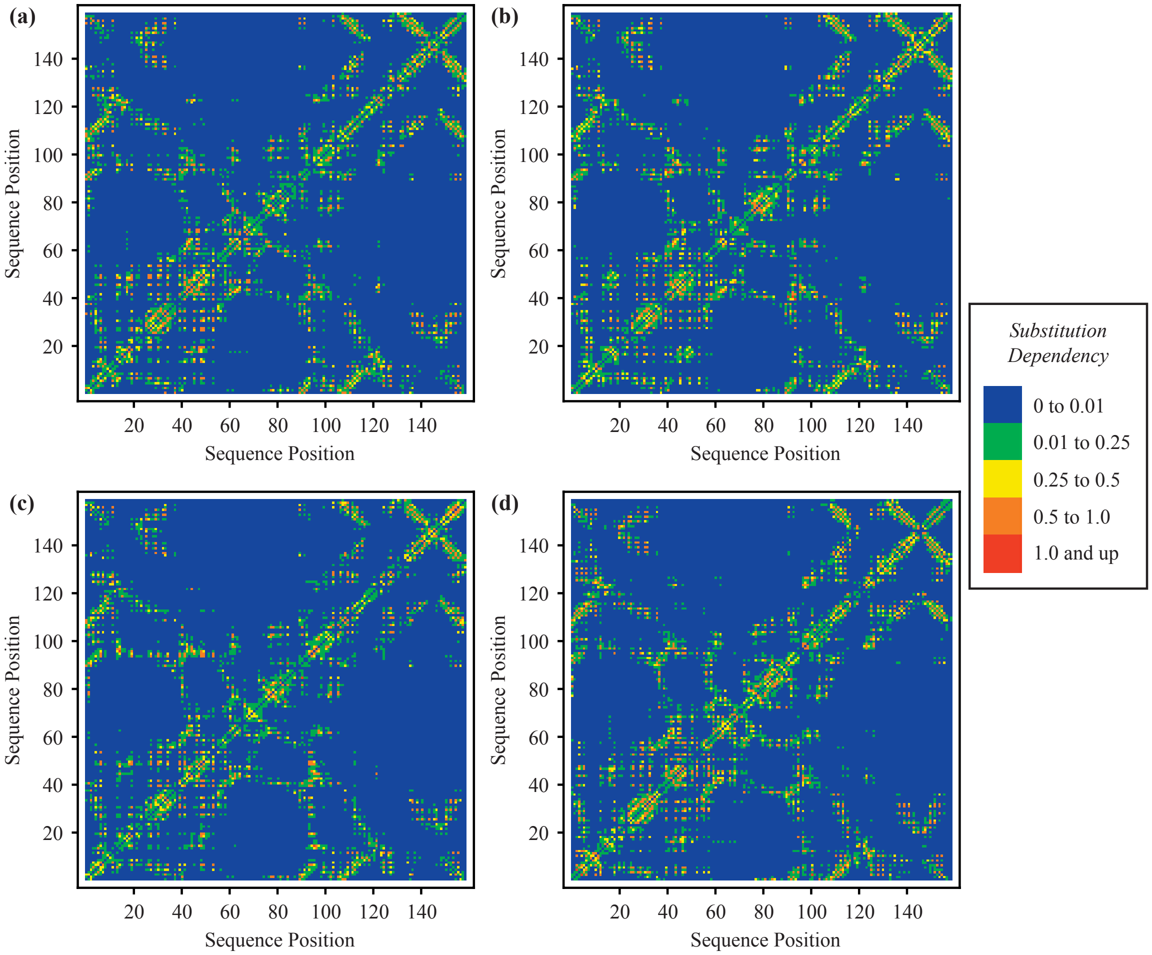


Figure 6.3: Map of substitution dependency for (a) *E. coli* DHFR, closed M20 (1rx2), (b) *E. coli* DHFR, occluded M20 (1rx2), (c) *E. coli* DHFR, open M20 (1ra9), (d) *L. casei* DHFR, closed M20 (3dfr).

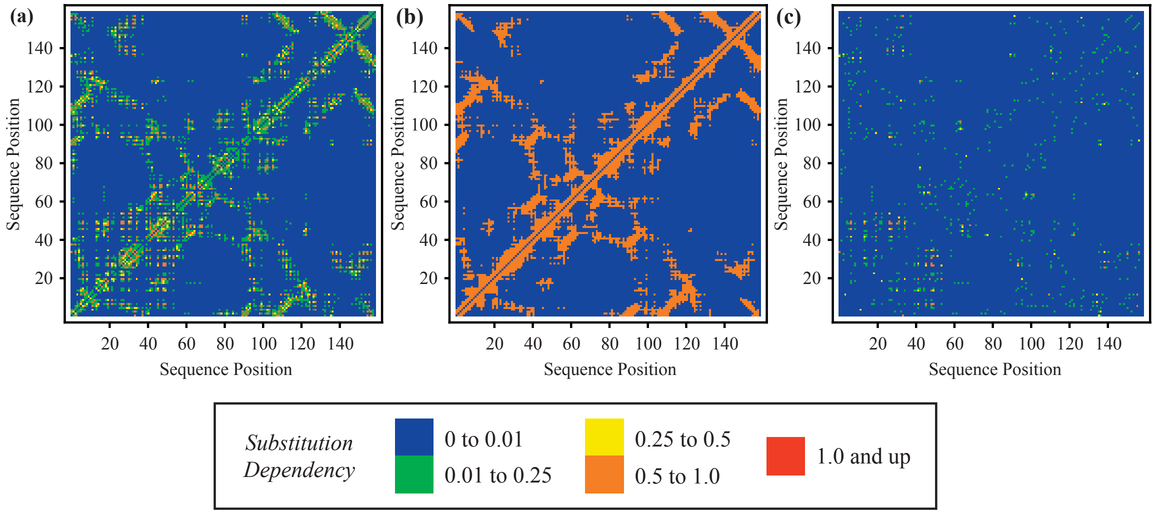


Figure 6.4: (a) Map of substitution dependency for E. coli DHFR, closed M20 (1rx2). Blue indicates no correlation whereas green, yellow, orange and red depict residue pairs with increasing levels of substitution dependency. (b) Contact map (< 8 Angstroms) for 1rx2. Orange denotes contacting residue pairs. (c) Map of substitution dependency after removing contacting residue pairs depicted in (a) for 1rx2.

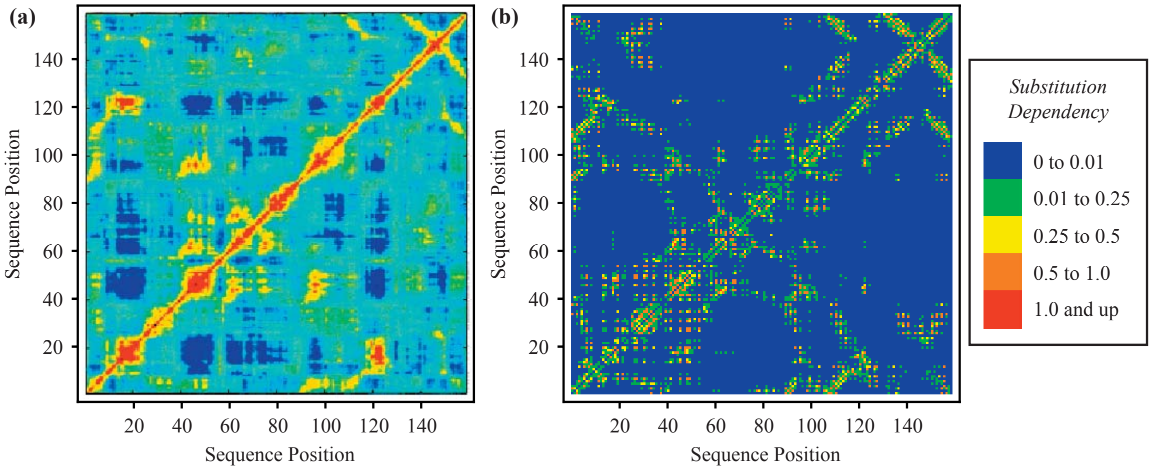


Figure 6.5: (a) Pairwise correlated motions found in ref. [38]. Red and yellow indicate residue pairs that move in the same direction, dark blue indicates pairs that move in opposite directions. Reprinted with permission from ref. [38]. Copyright 2000 American Chemical Society. (b) Map of substitution dependency for *E. coli* closed (1rx2).

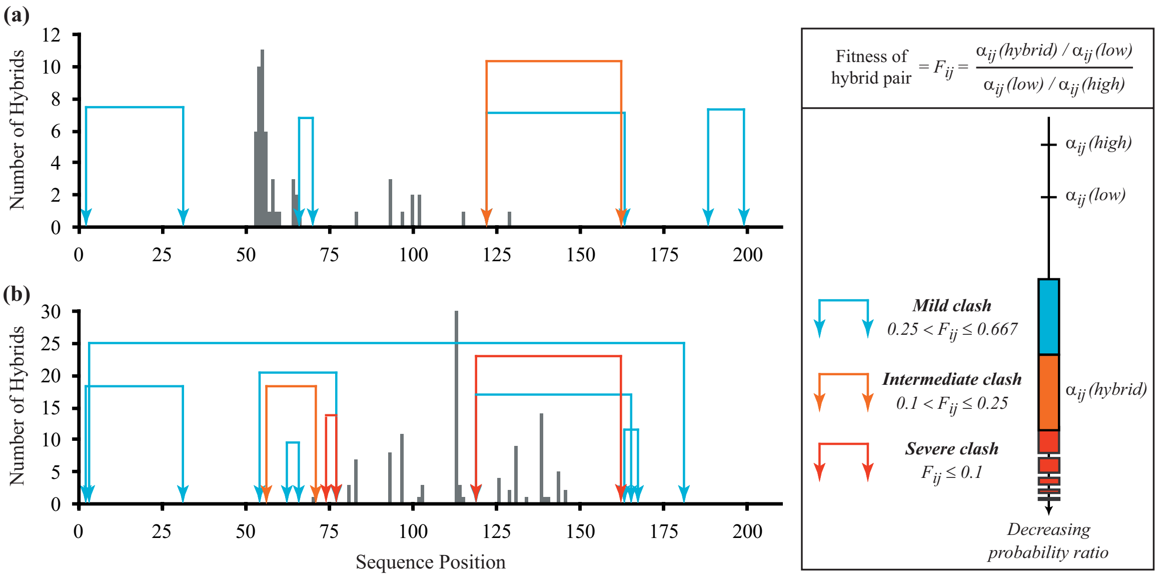


Figure 6.6: Clashing residue pairs in (a) human/*E. coli* hybrids and (b) *E. coli*/human hybrids. Clashes are classified as mild, intermediate or severe based on the fitness metric F_{ij}, which is calculated by comparing the probability ratio of the hybrid residue pair α_{ij}(*hybrid*) to the probability ratios of the parental sequences α_{ij}(*low*), α_{ij}(*high*), where *low* refers to the parental sequence with the lower α_{ij} and *high* refers to the higher-valued one. Vertical bars indicate positions where functional crossovers have been found in incremental truncation experiments [5, 12, 13].

Chapter 7: Conclusions

Section 7.1: Future Perspectives

As we enter the post-genomic era, we have in our hands an abundance of protein designs, experimental techniques and computational approaches. By creatively applying the ever-growing palette of molecular biology techniques, a variety of protocols are currently available for constructing combinatorial libraries with customized statistics of mutations and/or parental fragments. Future protocol developments are likely to be driven by the need to navigate around the increasingly complicated intellectual property landscape. To this end, the use of synthetic oligomers, taking advantage of substantial reductions in price, is likely to dominate, thus providing the means for exquisite control of combinatorial library diversity.

These enabling technology developments, along with the emerging trend of recombining more distant homologues, will further stress the need to computationally assess protein hybrids for stability and even functionality. The key dilemma of computational developments lies at establishing the proper trade-off between modeling accuracy and evaluation speed. Force fields are increasingly becoming more elaborate and customized to the task of protein engineering. However, there is almost unanimous agreement that their accuracy is still limited. For instance, an adequate and computationally tractable description of electrostatics remains elusive. Notable contributions in this direction include the recent work of Hellinga's group [1]. In response to the inherent difficulty of designing potentials with a firm grounding on biophysics fundamentals, a number of researchers are increasingly developing and successfully making use of scoring functions heavily parameterized to predict existing folds [2]. A recent impressive contribution along these lines is the *in silico* design and verification of a novel fold by Baker's group [3].

Even though ample experimental evidence shows that proteins have not evolved to maximize their stability, most computational approaches have aimed to design proteins with this as an objective. This is primarily a manifestation of our inability to *a priori* predict functionality rather than an affirmation that stability and functionality are always correlated. Clearly, there is a need to move beyond stability as a monolithic surrogate for functionality. To this end, sequence information gleaned from protein family databases (*e.g.*, Pfam [4]) can indirectly provide some answers. In the same way that protein structures in the Protein Data Bank [5] have been used to design potential energy functions for protein design, protein family sequence data, spanning all of nature's known solutions, can be used to constrain the solutions for various protein engineering problems. In fact, Lockless and Ranganathan [6] have found that statistical sequence database-derived coupling energies correlate with thermodynamic coupling free energies (*i.e.*, $\Delta\Delta G$ from double mutant cycle analysis) in a small protein domain.

Furthermore, it is important to stress that current protein design methods rely on a static picture for proteins. However, it is increasingly being accepted that proteins require the coordinated motion of an extensive network of interacting residues for correct catalytic function (see ref. [7] for review). Hybrid quantum-classical molecular dynamics (MD) simulations of wild-type and mutant dihydrofolate reductases uncovered a network of coupled promoting motions that occur as the wild-type hydride transfer reaction progresses [8]. The network was found to be disrupted in the mutant, reflecting its reduced reaction rate. In addition, recent MD simulations have revealed a link between thermostability and the fluctuations of surface loops away from the native state [9]. Incorporating dynamic information into protein design frameworks is likely to be challenging but may prove necessary to design proteins with novel functions.

The ever-accelerating rate of searching sequence space, driven by increased computational speed and clever algorithm design, is likely to continue. Particularly promising will be methods that can effectively combine the ability of stochastic methods (*e.g.*, genetic algorithms and simulated annealing) to scan vast amounts of sequence space with deterministic algorithms (*e.g.*, dead-end elimination) that can produce provably optimal solutions. Motivated by the need to design protein-based therapeutics and proteins with novel functionalities, exciting developments are likely to be forthcoming fueled by the inventiveness and constrained only by the imagination of experimentalists and theoreticians.

Section 7.2: References

1. Wisz, M.S. & Hellinga, H.W. (2003), "An empirical model for electrostatic interactions in proteins incorporating multiple geometry-dependent dielectric constants," *Proteins* **51**(3): 360-377.
2. Kuhlman, B. & Baker, D. (2000), "Native protein sequences are close to optimal for their structures," *Proc Natl Acad Sci USA* **97**(19): 10383-10388.
3. Kuhlman, B., Dantas, G., Ireton, G.C., Varani, G., Stoddard, B.L. & Baker, D. (2003), "Design of a novel globular protein fold with atomic-level accuracy," *Science* **302**(5649): 1364-1368.
4. Bateman, A., Birney, E., Cerruti, L., Durbin, R., Etwiller, L., Eddy, S.R., Griffiths-Jones, S., Howe, K.L., Marshall, M. & Sonnhammer, E.L. (2002), "The Pfam protein families database," *Nucleic Acids Res* **30**(1): 276-280.
5. Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N. & Bourne, P.E. (2000), "The Protein Data Bank," *Nucleic Acids Res* **28**(1): 235-242.
6. Lockless, S.W. & Ranganathan, R. (1999), "Evolutionarily conserved pathways of energetic connectivity in protein families," *Science* **286**(5438): 295-299.
7. Benkovic, S.J. & Hammes-Schiffer, S. (2003), "A perspective on enzyme catalysis," *Science* **301**(5637): 1196-1202.
8. Agarwal, P.K., Billeter, S.R., Rajagopalan, P.T., Benkovic, S.J. & Hammes-Schiffer, S. (2002), "Network of coupled promoting motions in enzyme catalysis," *Proc Natl Acad Sci USA* **99**(5): 2794-2799.
9. Wintrode, P.L., Zhang, D., Vaidehi, N., Arnold, F.H. & Goddard III, W.A. (2003), "Protein dynamics in a family of laboratory evolved thermophilic enzymes," *J Mol Biol* **327**(3): 745-757.

VITA

Gregory L. Moore

Ph.D. in Chemical Engineering, Penn State University, expected May 2005.

Ph.D. Thesis Title: Modeling and Optimization in Directed Evolution Protocols and Protein Engineering

Advisor: Professor Costas D. Maranas

GPA = 3.98 out of 4.00 (52 credits)

B.S. in Chemical Engineering, Penn State University, May 1998.

GPA = 3.87 out of 4.00 (141 credits)

University Schreyer Scholar

Minor in Chemistry, Penn State University, May 1998.