A NOVEL OPTIMIZATION ALGORITHM AND OTHER TECHNIQUES IN MEDICINAL CHEMISTRY

by

Radleigh G. Santos

A Dissertation Submitted to the Faculty of

The Charles E. Schmidt College of Science

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

Florida Atlantic University

Boca Raton, Florida

May 2012

A NOVEL OPTIMIZATION ALGORITHM AND

OTHER TECHNIQUES IN MEDICINAL CHEMISTRY

by

Radleigh G. Santos

This dissertation was prepared under the direction of the candidate's dissertation advisor, Dr. Dragan Radulovic, Department of Mathematics, and has been approved by the members of his supervisory committee. It was submitted to the faculty of the Charles E. Schmidt College of Science and was accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

SUPERVISORY COMMITTEE:

Dragan Radulovic, Ph.D.
Dissertation Advisor

Richard Houghten, Ph.D.

William Kalies, Ph.D.

Hongwei Long

Hongwei Long, Ph.D.

Rainer Steinwandt, Ph.D.

Lee Klingler

Lee Klingler, Ph.D.
Chair, Department of Mathematics

GW Perry

Gary W. Perry, Ph.D.
Dean, The Charles E. Schmidt College of Science

Barry T. Rosson

Barry T. Rosson, Ph.D.
Dean, Graduate College

March 29, 2012

Date

ACKNOWLEDGEMENTS

Since grade school mathematics has always been a joy and inspiration to me, and I am honored to be among the ranks of the highest level of education in the field. I could not have gotten this far were it not for countless people who have been in my life. These are but a notable few who deserve some credit for my achievement:

I would like to thank all professors, mathematics or otherwise, from whom I have had the privilege to learn during my time at MIT, Georgia Tech, and FAU. Special thanks to scientists I continue to be privileged to work with at TPIMS: Marc Guilianatti, Jose Medina-Franco, Karina Martinez-Mayorga, Clemencia Pinilla, Valeria Judkowski, Dmitriy Minond, Joanna Davies, and Richard Houghten. I am grateful to my committee members William Kalies, Hongwei Long, and Rainer Steinwandt. Thank you also to Lee Klingler for his support during the final months of the process. Most of all I am thankful for being able to work with my advisor Dragan Radulovic, with whom I have never ceased to enjoy our shared enthusiasm over applied mathematics.

Most of all I would like to thank the friends and family who were constant sources of support for me. Thanks to my grandmother Avee. Love, gratitude, and thanks to my wonderful sister Marlisa, who has always been there for me, my brother-in-law Randy, and to my Mother, who always has my back. And finally to my "Mathematical Muse" Denia, my deepest love and thanks, for supporting and believing in me even when I did not.

ABSTRACT

Author:	Radleigh G. Santos
Title:	A Novel Optimization Algorithm and Other Techniques in Medicinal Chemistry
Institution:	Florida Atlantic University
Dissertation Advisor:	Dr. Dragan Radulovic
Degree:	Doctor of Philosophy
Year:	2012

In this dissertation we will present a stochastic optimization algorithm and use it and other mathematical techniques to tackle problems arising in medicinal chemistry. In Chapter 1, we present some background about stochastic optimization and the Accelerated Random Search (ARS) algorithm. We then present a novel improvement of the ARS algorithm, Directed Accelerated Random Search (DARS), motivated by some theoretical results, and demonstrate through numerical results that it improves upon ARS. In Chapter 2, we use DARS and other methods to address issues arising from the use of mixture-based combinatorial libraries in drug discovery. In particular, we look at models associated with the biological activity of these mixtures and use them to answer questions about sensitivity and robustness, and also present a novel method for determining the integrity of the synthesis. Finally, in Chapter 3 we present an in-depth analysis of some statistical and mathematical techniques in combinatorial chemistry, including a novel probabilistic approach to using structural similarity to predict the activity landscape.

A NOVEL OPTIMIZATION ALGORITHM AND

OTHER TECHNIQUES IN MEDICINAL CHEMISTRY

APPENDIXES

LIST OF TABLES

LIST OF FIGURES

INTRODUCTION

Optimization has existed far longer than humans have had the mathematical framework with which to describe it. The search for the input value for which one can expect the “best” possible outcome is endemic into nature itself; indeed, vast and successful branches of mathematical optimization have been built around[1-35], for example, the ability for biological systems to adapt means for survival, or the tendency for thermodynamic systems to find an optimal ground state. And whether through mathematics or not, humans have been struggling to find answers to their optimization problems since the moment they discovered they had them. It is not surprising, then, that the interplay between mathematics in general, and optimization in particular, should be so important in the current study of the natural world.

In Chapter 1, we will present some specific optimization algorithms. Among them is the Accelerated Random Search (ARS) algorithm, a relatively new optimization algorithm that has had a great deal of success both in numerical and theoretical results [1-2]. In particular, ARS answers the three above questions in a manner that is highly rewarding to the user: The algorithm will always find the next best point, it is guaranteed to lead to best overall point, and under a wide range of circumstances will do so quickly. ARS, however, is insufficient in handling a surprisingly common class of optimization problems: those problems which exhibit a high disparity in length scales across different dimensions. This class of functions would at first glance seem esoteric, but in fact this phenomenon is apparent even in attempts to use ARS to find the solutions to simple linear systems[1]. We therefore will present a novel improvement upon ARS, Directed Accelerated Random Search (DARS), and demonstrate that DARS has indeed captured the positive aspects of the ARS algorithm and significantly reduced limitations of ARS.

In Chapters 2 and 3, we will present some novel research in the areas of Medicinal Chemistry and Drug Discovery that use both the DARS algorithm and other applied mathematical and statistical techniques. In Chapter 2, we show that the activities of individual compounds in mixture-based combinatorial libraries can follow the harmonic addition model, and show the positive implications for the use of such mixtures in the drug discovery setting in general. We then present a novel derivation of an extension of this model to a greater class of biological activity systems. Also in Chapter 2, we show that DARS can be used as part of a methodology to determine the integrity of the synthesis process for novel non-peptide mixture-based combinatorial libraries, therefore leading to greater credibility in their use. In Chapter 3, we present a detailed account of mathematical and statistical techniques used in Computational Chemistry to describe many varied aspects of the structural-activity landscapes of a set of compounds. Finally, we present a novel approach combining these Computational Chemistry techniques with joint probability distributions to find classify structural similarity methods and to find previously unknown active compounds.

CHAPTER 1

A DIRECTED ACCELERATED RANDOM SEARCH ALGORITHM

1.1 Definitions and Assumptions

Throughout this chapter, we use the following definitions, assumptions, and notation:

(1.1.1) Let Ω be a subset of a d-dimensional vector space equipped with a norm $\|\cdot\|$ and an orthonormal basis $\{\hat{u}_i\}_{i=1}^{d}$. Further, let (Ω, Σ, μ) be a probability measure space with sigma algebra Σ and measure μ. When $\Omega \subset \mathbb{R}^d$, we assume $\|\cdot\|$ to be the standard Euclidean norm, and μ to be the uniform probability measure on the Borel sigma algebra Σ. We denote elements of Ω as $\vec{x} \in \Omega$. Ω is referred to as the **search domain**.

(1.1.2) Let $F: \Omega \to \mathbb{R}$ be a measurable, real-valued function on Ω. We assume that F has a finite essential supremum F^* attained at a unique $\vec{x}^* \in \Omega$. We refer to F as the **objective function**. We note that any time "maximization" is referenced, we refer to the process by which F^* and $\vec{x}^*$ are determined. When "minimization" of a function f is referred to, we set $F = -f$ and maximize F under the above assumptions. Additional restrictions on F may be imposed depending on context.

(1.1.3) We denote the uniform probability distribution on a set $A \subseteq \Omega$ by $U(A)$.

(1.1.4) We define the **ball** of radius r centered at $\vec{x} \in \Omega$ to be

$B(\vec{x}, r) = \{\vec{y} \in \Omega \mid \|\vec{x} - \vec{y}\| < r\}$.

(1.1.5) We define the **diameter** of a set $A \subseteq \Omega$ as

$Diam(A) = sup_{\vec{x},\vec{y} \in \mathrm{A}}(\|\vec{x} - \vec{y}\|)$.

In this chapter we will present iterative maximization optimization algorithms. We use the following general notation when discussing these algorithms:

(1.1.6) The point $\vec{x}_i \in \Omega$ refers to the output of a given optimization algorithm after *i* iterations. We refer to this as the **current best point** after *i* iterations. After n iterations, an iterative optimization algorithm will create a sequence $\{\vec{x}_i\}_{i=0}^{n}$ in Ω, referred to as the **record sequence** of a given algorithm.

(1.1.7) An optimization algorithm will be formally defined as **consistent** when the sequence $F(\vec{x}_n)$ converging to F^* almost surely, i.e.

$P\left(\left\{\lim_{\mathrm{n}\to\infty}\left(F(\vec{x}_n)\right) = F^*\right\}\right) = 1$.

(1.1.8) A point $\vec{x}_{n+1}$ is referred to as an **improvement point** if $F(\vec{x}_{n+1}) > F(\vec{x}_n)$. In the algorithms presented below, this is equivalent to $\vec{x}_{n+1} \neq \vec{x}_n$.

(1.1.9) We define the **acceptance region** for a current best point $\vec{x}_n$ to be

$AR(\vec{x}_n) = \{\vec{y} \in \Omega \mid F(\vec{y}) > F(\vec{x}_n)\}$.

(1.1.10) Each time a random value is generated during the course of a given algorithm, we assume that the resulting values are independent of one another.

(1.1.11) We refer to the **search region** on the *n*th iteration of an algorithm to be the set that the next best point $\vec{x}_{n+1}$ is generated from.

(1.1.12) All algorithms below are initialized with a point $\vec{x}_0$ generated from $U(\Omega)$.

1.2 Pure Random Search

The Pure Random Search (PRS) algorithm is a standard optimization algorithm known for its consistency but slow convergence[1]. Given a point $\vec{x}_n$, the **PRS** definition for its iterative step is very straightforward:

(1) Generate a point $\vec{x}_{temp}$ from $U(\Omega)$.

(2) If $F\left(\vec{x}_{temp}\right) > F(\vec{x}_n)$, $\vec{x}_{n+1} = \vec{x}_{temp}$. Otherwise, $\vec{x}_{n+1} = \vec{x}_n$. Let $n \to n + 1$ and go to (1).

Thus PRS simply uses the entire search domain as its search region, and picks random points from that region until a more optimal one is found.

The consistency of PRS is a long-established result. It is implied in a broader sense by [25], but it is not clear when or the manner in which it was first established. Here we offer a straightforward proof of it which is relevant to the narrative in light the other stochastic optimization algorithms that are to be discussed:

PROPOSITION 1.2.1 (CONSISTENCY OF PRS)

Let $\left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty}$ *be the record sequence associated with the PRS algorithm.*

Then $P\left(\left\{\lim_{n\to\infty}\left(F\left(\vec{x}_n{}^{(PRS)}\right)\right) = F^*\right\}\right) = 1$

Proof:

Let $\widetilde{\Omega} = \Omega \backslash \{\vec{x} \in \Omega \mid F(\vec{x}) > F^*\}$. By the definition of the essential supremum, $P(\widetilde{\Omega}) = 1$. Clearly, since each $\vec{x}_n$ is generated from $U(\Omega)$, $P(\{\vec{x}_n \in \widetilde{\Omega}\}) = 1$ for all *n* and hence

$P\left(\left\{\left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty} \subset \widetilde{\Omega}\right\}\right) = 1$. Therefore:

$$P\left(\left\{\lim_{n\to\infty}\left(F\left(\vec{x}_n{}^{(PRS)}\right)\right) = F^*\right\}\right)$$

$$= P\left(\left\{\lim_{n\to\infty}\left(F\left(\vec{x}_n{}^{(PRS)}\right)\right) = F^* \mid \left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty} \in \widetilde{\Omega}\ \right\}\right)$$

The visit record for PRS creates a non-decreasing sequence $\left\{F\left(\vec{x}_n{}^{(PRS)}\right)\right\}_{n=1}^{\infty}$ of real numbers which is bounded from above by F^* on $\widetilde{\Omega}$. Clearly $\lim_{n\to\infty}\left(F\left(\vec{x}_n{}^{(PRS)}\right)\right) = F^*$ if and only if $sup_n\left(F\left(\vec{x}_n{}^{(PRS)}\right)\right) = F^*$. Therefore

$$P\left(\left\{\lim_{n\to\infty}\left(F\left(\vec{x}_n{}^{(PRS)}\right)\right) = F^* \mid \left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty} \in \widetilde{\Omega}\ \right\}\right)$$

$$= P\left(\left\{sup_n\left(F\left(\vec{x}_n{}^{(PRS)}\right)\right) = F^* \mid \left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty} \in \widetilde{\Omega}\right\}\right)$$

$$= 1 - P\left(\left\{\exists\ m\ s.t. F\left(\vec{x}_n{}^{(PRS)}\right) \le F^* - \tfrac{1}{m}\ \forall\ n \mid \left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty} \in \widetilde{\Omega}\right\}\right)$$

$$\ge 1 - P\left(\left\{\exists\ m\ s.t. F\left(\vec{x}_n{}^{(PRS)}\right) \le F^* - \tfrac{1}{m}\ i.o. \mid \left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty} \in \widetilde{\Omega}\right\}\right)$$

$$= 1 - P\left(\bigcup_m\left\{F\left(\vec{x}_n{}^{(PRS)}\right) \le F^* - \tfrac{1}{m}\ i.o. \mid \left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty} \in \widetilde{\Omega}\right\}\right)$$

$$\ge 1 - \sum_m P\left(\bigcap_{j=1}^{\infty}\bigcup_{k=j}^{\infty}\left\{F\left(\vec{x}_k{}^{(PRS)}\right) \le F^* - \tfrac{1}{m} \mid \left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty} \in \widetilde{\Omega}\right\}\right)$$

Now

$$P\left(\left\{F\left(\vec{x}_k{}^{(PRS)}\right) < F^* - \tfrac{1}{m} \mid \left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty} \in \widetilde{\Omega}\right\}\right) = \left(1 - P\left(\left\{F^* - \tfrac{1}{m} < F\right\}\right)\right)^k$$

since the former expression occurs only when k points are chosen from outside the set $\left\{F^* - \frac{1}{m} < F\right\}$. But the set $\left\{F^* - \frac{1}{m} < F\right\}$ must be a set of positive measure for any m. Thus there exists $\lambda_m \in (0,1)\ s.t.\left(1 - P\left(\left\{F^* - \frac{1}{m} < F\right\}\right)\right)^k = \lambda_m{}^k$. Therefore

$$1 - \sum_m P\left(\bigcap_{j=1}^{\infty} \bigcup_{k=j}^{\infty} \left\{F\left(\vec{x}_k{}^{(PRS)}\right) \le F^* - \tfrac{1}{m} \middle| \left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty} \in \widetilde{\Omega}\right\}\right)$$

$$= 1 - \sum_m \lim_{j\to\infty} P\left(\bigcup_{k=j}^{\infty} \left\{F\left(\vec{x}_k{}^{(PRS)}\right) \le F^* - \tfrac{1}{m} \middle| \left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty} \in \widetilde{\Omega}\right\}\right)$$

$$\ge 1 - \sum_m \lim_{j\to\infty} \sum_{k=j}^{\infty} \lambda_m{}^k = 1$$

This proves the proposition. □

Although Pure Random Search is consistent, the probability of finding a point in $AR\left(\vec{x}_n{}^{(PRS)}\right)$ can in practice become very small as *n* increases, making successive improvements harder and harder to achieve. Improving the convergence rate of Pure Random Search is an important motivating factor in much research[1-4, 19-25], and was successfully achieved with the Accelerated Random Search algorithm.

1.3 Accelerated Random Search

As stated above, PRS is a consistent algorithm that converges slowly. The importance of consistency in an algorithm cannot be overstated; in general, one does not know what the value of F^* is, and so when one seeks to maximize a function one relies on the optimization algorithm used to deliver a correct answer. At the same time, a slowly converging consistent algorithm offers similar problems, since it is unclear how long to run such an algorithm in order to find the optimal value. Newton and Quasi-Newton algorithms, while lacking consistency, often converge quickly, and in the context of multiple restarts can be very effective on some types of problems [26, 27]. Such algorithms, however, have serious flaws in them: They require the existence of the Gradient and/or Hessian of the objective function, which in general need not exist. Even on occasions when a non-existent Gradient or Hessian can be approximated, these algorithms cannot accommodate flat regions, sometimes becoming numerically unstable in addition to being unable to find the global maximum[25-26]. Furthermore, the numerical complexity of each iterative step is substantial in order to calculate or approximate the Hessian of many functions[26]. Thus a Quasi-Newtonian method requiring fewer function evaluations may nonetheless require substantial computer time to return a solution. This becomes especially true when the maximum is along the boundary of the search domain in constrained optimization[26]; substantial computational effort must be made in order to keep each iterative step inside the desired search domain. Finally there are entire classes of problems for which the Newtonian approach is ill-suited; combinatorial optimization problems, for example, must either be made artificially continuous or substantially constrained in order for a Newtonian algorithm to be applied, with limited success[1, 25].

1.3.1 The Finite Descent ARS Algorithm

The Accelerated Random Search, or ARS, algorithm was first introduced by Appel, Labarre, and Radulovic in [1]. It offers the consistency of PRS at a substantially faster rate of convergence without any of the limitations in applicability or implementation of the Newtonian and Quasi-Newtonian algorithms mentioned above. For real numbers $c > 1$ (which we call the **contraction factor**), $R > 0$ (which we call the **maximal radius**), and $\rho > 0$ (which we call the **precision threshold**), the finite descent ARS heuristic iterative step as given in [1] is

(1) Given $\vec{x}_n \in \Omega$ and $r \in (\rho, R]$, generate $\vec{y}$ from $U(B(\vec{x}_n, r))$.

(2) If $F(\vec{y}) > F(\vec{x}_n)$ then let $\vec{x}_{n+1} = \vec{y}$ and $r = R$.

Otherwise, let $\vec{x}_{n+1} = \vec{x}_n$ and $r = r/c$.

(3) If $r < \rho$, then $r = R$. Let $n \to n + 1$ and go to (1).

In words, ARS shrinks the region around the best point whenever it does not find a better one, and searches the whole search domain again once it does. If $Diam(\Omega) < \infty$, then R is set to $Diam(\Omega)$ so that $U(\Omega)$ is sampled from repeatedly. It may seem counterintuitive to search a larger space once an improvement is found. As a result of this approach, however, we get consistency. The following was first presented in [1], and we include it here with minor alteration because this approach will parallel the consistency proof of a novel algorithm later on:

PROPOSITION 1.3.1 (CONSISTENCY OF ARS)

Let $\left\{\vec{x}_n{}^{(ARS)}\right\}_{n=0}^{\infty}$ *be the record sequence associated with the ARS algorithm with* $Diam(\Omega) < \infty$.

Then $P\left(\left\{\lim_{\mathrm{n}\to\infty}\left(F\left(\vec{x}_n{}^{(ARS)}\right)\right) = F^*\right\}\right) = 1$

Proof:

An infinite subsequence of points generated during the ARS algorithm will be from $U(\Omega)$; call these points $\left\{\vec{x}_{n_k}{}^{(ARS)}\right\}$. These points are precisely a visit record from the PRS algorithm above, so we know immediately that $P\left(\left\{\lim_{\mathrm{k}\to\infty}\left(F\left(\vec{x}_{n_k}{}^{(ARS)}\right)\right) = F^*\right\}\right) = 1$ from Proposition 1.2.1. In particular this means that $P\left(\left\{sup_k\left(F\left(\vec{x}_{n_k}{}^{(ARS)}\right)\right) = F^*\right\}\right) = 1$, which means $P\left(\left\{sup_n\left(F\left(\vec{x}_n{}^{(ARS)}\right)\right) = F^*\right\}\right) = 1$ as well since $sup_k\left(F\left(\vec{x}_{n_k}{}^{(ARS)}\right)\right) \leq sup_n\left(F\left(\vec{x}_n{}^{(ARS)}\right)\right) \leq F^*$. Finally $\left\{\vec{x}_n{}^{(ARS)}\right\}_{n=0}^{\infty}$ forms a non-decreasing sequence, so $P\left(\left\{sup_n\left(F\left(\vec{x}_n{}^{(ARS)}\right)\right) = F^*\right\}\right) = 1$ is equivalent to

$P\left(\left\{\lim_{n\to\infty}\left(F\left(\vec{x}_n{}^{(ARS)}\right)\right) = F^*\right\}\right) = 1.$ □

The counterintuitive nature of moving away from a better point once it is found is therefore justified; in order to ensure consistency, ARS searches the whole search domain infinitely often, but exponentially reduces its search region so that most points are chosen near to the current best point.

In [1] we also have the following result:

THEOREM 1.3.1

Let F be a continuous function on a finite subset of $\Omega = \mathbb{R}^d$ with finitely many global maxima. Let $\left\{\vec{x}_n{}^{(ARS)}\right\}_{n=0}^{\infty}$ be the record sequence associated with the finite descent ARS algorithm, and let $\left\{\vec{x}_n{}^{(PRS)}\right\}_{n=0}^{\infty}$ be the record sequence associated with the PRS algorithm. Given a contraction factor $c > 1$ and a precision threshold $0 < \rho < 1$, then for each $C < \frac{c^{\frac{|ln\,(\rho)|}{ln\,(c)}}}{3^{\frac{|ln\,(\rho)|}{ln\,(c)}}}$ there exists a positive integer N_C such that for all $n > N_C$,

$E\left[F\left(\vec{x}_n{}^{(ARS)}\right)\right] \geq E\left[F\left(\vec{x}_{Cn}{}^{(PRS)}\right)\right].$ □

Essentially Theorem 1.3.1 says that eventually the average output of ARS will exceed the average output of C times as many iterations PRS for any C below the above bound. In typical applications $C > 1000$, meaning that ARS outperforms PRS by orders of magnitudes while maintaining consistency.

An altered version of the ARS algorithm was proven to converge exponentially faster than PRS in [1]. The first notion of an ARS-type meta-heuristic was first introduced by Solis and Wets in [25]. The ARS algorithm has been compared to Simulated Annealing, in the sense that generating points from exponentially decreasing balls generates a similar overall set of test points to the procedure in Simulated Annealing whereby points are discarded with exponential probability. ARS, however, does not require any sort of "cooling function" and is more direct in heuristic and implementation.

1.3.2 The Limitations of Accelerated Random Search

As described in [1], ARS is not without its limitations. It does avoid the guarantee of convergence to a local extremum by periodic sampling of the entire search domain, in contrast to other local search methods such as Gradient Decent or Nelder-Mead [26-27,35]. However, in practice this may take may functions evaluations to achieve, which is why ARS, like other local search algorithms, is best implemented with restarts. More disturbing, however, is the failure of ARS to converge quickly when the acceptance region contains geometries with different length scales. In fact, here we present a novel result demonstrating of how easily ARS can be made to fail when presented with such circumstances:

THEOREM 1.3.2

For the finite descent ARS algorithm with parameters $c > 1$, $R > 0$, and $\rho = \frac{R}{c^M}$, , for any $m > 1$ there exists a function F s.t.

$$E[\|\vec{x}_{n+1} - \vec{x}_n\|] \leq K_m \|\vec{x}_n - \vec{x}^*\|^m$$

for some constant K_m depending only on m and M.

Proof:

Let $F(x, y) = (1 - x) \cdot \mathbb{I}_{\{0 \leq y \leq x^m\} \cup \{x \geq 0\}}$ on $[-1,1]^2$, which has a maximal value of 1 at $(0,0)$. For a current best point (x_n, y_n) it is obvious that $AR(x_n, y_n) = \{0 \leq x \leq x_n, 0 \leq$

$y \leq {x_n}^m\}$. Clearly, the quantity we are bounding above is maximized when $(x_n, y_n) = (\delta, 0)$ for some $\delta \in (0,1)$; for an x-coordinate of δ, points of this form minimize the distance to the origin while maximizing the overlap with $AR(x_n, y_n)$. Therefore we may assume $(x_n, y_n) = (\delta, 0)$ and so $\|\vec{x}_{n+1} - \vec{x}_n\| = \delta$. Let $r_k = \frac{R}{c^k}$, the radius of the search region after k shrinkages. Let L_j be the value of k s.t. $\frac{\delta^j}{c} \leq r_{L_j} < \delta^j$, for $1 \leq j \leq m$, i.e. L_j is the first value of k for which $r_k < \delta^j$. Next, note that, if an improvement occurs from a ball of size r_k, the centroid of the acceptance region is always within $\frac{r_k}{2}$ of $(\delta, 0)$ when $r_k < \delta$, and always within $\frac{\delta}{2}$ of $(\delta, 0)$ when $r_k \geq \delta$. Thus, conditioned on the improvement occurring on the k^{th} iteration,

$$E(\vec{x}_{n+1} - \vec{x}_n) \leq \begin{cases} \frac{\delta}{2}, & r_k \geq \delta \\ \frac{r_k}{2}, & r_k < \delta \end{cases}$$

Finally note that

$$P(Improvement\ on\ k^{th}\ iteration) \leq \frac{\int_{\delta - \min(\delta, r_k)}^{\delta} x^m dx}{\pi {r_k}^2}$$

$$= \begin{cases} \frac{\delta^{m+1}}{(m+1)\pi {r_k}^2}, & r_k \geq \delta \\ \frac{\delta^{m+1} - (\delta - r_k)^{m+1}}{(m+1)\pi {r_k}^2}, & r_k < \delta \end{cases}$$

while for all k, $P(Improvement\ on\ k^{th}\ iteration) \leq \frac{1}{4}$, since whenever $x > x_n$ or $y < 0$ the point lies outside $AR(\delta, 0)$. Therefore

$$E[\|\vec{x}_{n+1} - \vec{x}_n\|] \le \sum_{k=1}^{L_1-1} \frac{\delta}{2}\frac{\delta^{m+1}}{(m+1)\pi {r_k}^2} + \sum_{j=1}^{m}\sum_{k=L_j}^{L_{j+1}-1} \frac{r_k}{2}\frac{\delta^{m+1}-(\delta-r_k)^{m+1}}{(m+1)\pi {r_k}^2} + \sum_{k=L_m}^{M}\frac{r_k}{2}\frac{1}{4}$$

$$\le \sum_{k=1}^{L_1-1} \frac{\delta^{m+2}}{2(m+1)\pi\delta^2} + \sum_{j=1}^{m}\sum_{k=L_j}^{L_{j+1}-1} \frac{\delta^{m+1}-\left(\delta-\delta^j\right)^{m+1}}{2(m+1)\pi\delta^j} + \sum_{k=L_m}^{M}\frac{\delta^m}{8}$$

$$\le \frac{(L_1-1)\delta^m}{2(m+1)\pi} + \sum_{j=1}^{m}\frac{\delta^m}{2(m+1)\pi}\sum_{k=L_j}^{L_{j+1}-1}\frac{1-\left(1-\delta^{j-1}\right)^{m+1}}{\delta^{j-1}} + \frac{(M-L_m)\delta^m}{8}$$

Now note that, for $0 < x < 1$, the function $\frac{1-(1-x)^{m+1}}{x} \le \lim_{x\to 0}\frac{1-(1-x)^{m+1}}{x} = m+1$.

So

$$E[\|\vec{x}_{n+1} - \vec{x}_n\|] \le \frac{(L_1-1)\delta^m}{2(m+1)\pi} + \sum_{j=1}^{m}\frac{\delta^m}{2(m+1)\pi}\sum_{k=L_j}^{L_{j+1}-1}(m+1) + \frac{(M-L_m)\delta^m}{8}$$

$$= \frac{(L_1-1)\delta^m}{2(m+1)\pi} + \sum_{j=1}^{m}\frac{\left(L_{j+1}-L_j\right)(m+1)\delta^m}{2(m+1)\pi} + \frac{(M-L_m)\delta^m}{8}$$

$$\le \left(\frac{1}{2(m+1)\pi} + \frac{m}{2\pi} + \frac{1}{8}\right)M\delta^m$$

□

In other words, when $\left\|\vec{x}_{n+1} - \vec{x}^*\right\| < 1$, ARS can be expected to converge to $\vec{x}^*$ at an arbitrarily slow rate for certain functions. Such “thin” functions as demonstrated in the above Theorem force the ARS search region to shrink many times (and thus go through many function evaluations) before the acceptance region represents a substantial portion of the search region. See Figure 1.1. This results in successive improvements using the ARS algorithm to be very close to one another; the combination of more function evaluations per improvement and little progress on each improvement leads to exceptionally slow convergence to the actual extremum.

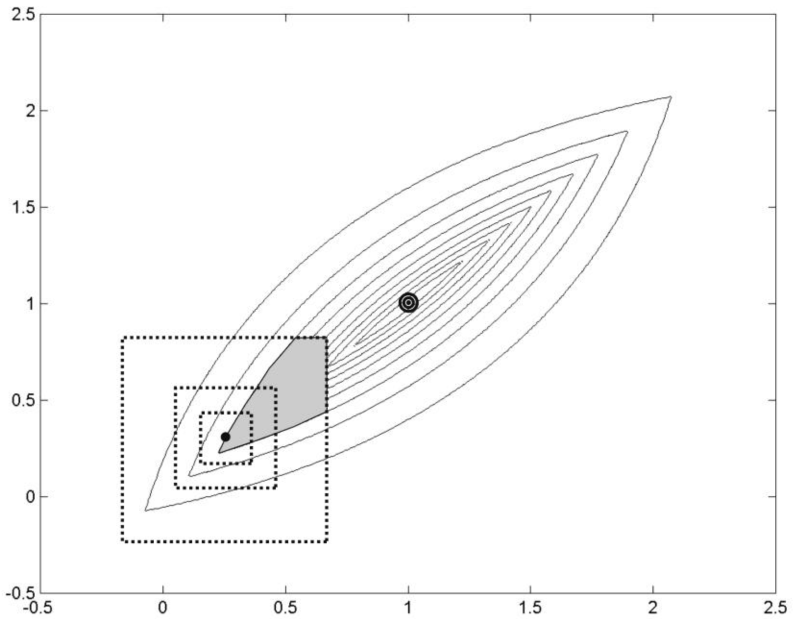


Figure 1.1 ARS optimizing a "thin" function. Shaded area indicates the acceptance region, which is proportionally small for large boxes. This makes successive improvements close to one another.

Alteration of the core ARS algorithm is not uncommon (multiple versions are presented in [1] alone). In practice one often uses hyper-cubes rather than balls when implementing ARS[1], for ease of coding. Modifications to the ARS algorithm, such as using normally distributed variables instead of uniform, have been proposed[38].   ARS was even combined with the Nelder-Mead search heuristic in [51]. However, none of these proposed modifications have improved upon the aforementioned key ARS weakness when applied to search regions with "thin" geometries, nor have they sought out to do so. Nor are such functions as obscure as they may initially sound; as described in [1], in finding the solutions of linear systems of order four or higher, the improvement regions have just such geometries. Below we present a new algorithm that maintains all the strength of the ARS algorithm but avoids this weakness.

1.4 Directed Accelerated Random Search

The most direct way to include directionality is to use the gradient of the objective function in some way; this is not favorable, however, since it limits the resulting algorithm to differentiable functions only. Successful non-gradient directional algorithms, such as Powell's and Rosenbrock's methods, get around this problem, but lack consistency without restarts and have no clear means of incorporating ARS's stochastic heuristic into their own. We therefore develop below the novel approach of incorporating directionality into the ARS algorithm, to offer a new, and, we intend to show, improved optimization algorithm: Directed Accelerated Random Search (DARS).

1.4.1 Directional Information Present in the ARS Visit Record

To motivate the approach used in the DARS algorithm, we first offer a novel result demonstrating that important geometric information is present in the record sequence of ARS which may be exploited.

THEOREM 1.4.1

Let $\{\vec{x}_n\}_{n=0}^{\infty}$ be the record sequence associated with the (finite decent) ARS algorithm. Let the objective function F have the additional property that, for some constant unit vector $\vec{D} \in \Omega$ and for all $\vec{x} \in \Omega$, $AR(\vec{x}) = \{\vec{x} + \vec{v} \,|\vec{v} \cdot \vec{D} > 0\}$. Let $\vec{l}_n$ denote the unit directional vector of a line joining the average of the sequence of improvements points $\sum_{k=1}^{n} \frac{\vec{x}_k}{n}$ with the current best point $\vec{x}_{n+1}$. Then $\vec{l}_n \overset{P}{\rightarrow} \vec{D}$.

Proof:

Since the balls used for the generation of the ARS visit record are spherically symmetric, and since $\vec{D}$ is constant, then without loss of generality we may assume that $\vec{D} = \langle 1,0,0,\dots,0\rangle$, i.e. that the direction of improvement corresponds to one of the basis vectors of Ω. Define $\vec{x}_n = \langle x_n^{(1)}, x_n^{(2)}, \dots, x_n^{(d)}\rangle$. Define $R_i = \frac{R}{c^i}$ to be the radius of the search region of the i^{th} ARS iteration, for $i = 0,1,\dots,I$. For the current best point $\vec{x}_n$ we define ρ_{n+1} to be the radius of the ball from which the improvement to the next best point $\vec{x}_{n+1}$ occurs. Although the search regions are balls centered at $\vec{x}_n$, by definition of the objective function F it is clear that the distribution of ρ_{n+1} does not depend on $\vec{x}_n$. In fact, since a successful improvement will occur from the set $\left\{\vec{x}_{n+1} \in B(\vec{x}_n, R_i) | x_{n+1}^{(1)} > x_n^{(1)}\right\}$, the probability of an improvement occurring on any one ARS iteration is $\frac{1}{2}$ regardless of the value of i or $\vec{x}_n$. Consequently one can show that

$$P(\rho_{n+1} = R_i) = \frac{1}{2^{\mathrm{i}+1}} + \frac{1}{2^{\mathrm{i}+1}}\left(\frac{1}{2^{\mathrm{I}+1}}\right) + \frac{1}{2^{\mathrm{i}+1}}\left(\frac{1}{2^{\mathrm{I}+1}}\right)^2 + \cdots = \frac{2^{\mathrm{I}-\mathrm{i}}}{2^{\mathrm{I}+1} - 1}$$

Moreover, by the definition of the ARS algorithm the $\{\rho_{n+1}\}$ are independent. Next we define $\vec{\Delta}_n \equiv \vec{x}_{n+1} - \vec{x}_n$ and note that $\{\vec{\Delta}_n\}$ are also independent $\forall\, n$, and that if we define $\vec{x}_0 = \vec{0}$ then $\vec{x}_k = \sum_{i=1}^{k} \vec{\Delta}_i$. Furthermore, if we condition on $\rho_{n+1} = R_i$, then the conditional distribution of $\vec{\Delta}_n$ is uniform; i.e. in our notation $\vec{\Delta}_n | R_i \sim U\left(\left\{\vec{v} \in B\left(\vec{0}, R_i\right) | \vec{v}^{(1)} > 0\right\}\right)$. This coupled with the above computation now implies that the $\{\vec{\Delta}_n\}$ are both independent and identically distributed. By this, boundedness, and symmetry, we have $E\left[\Delta_n^{(j)}\right] = 0$ and $Var\left[\Delta_n^{(j)}\right] \equiv C < \infty$ for $j > 1$, while $E\left[\Delta_n^{(1)}\right] \equiv M > 0$ and $Var\left[\Delta_n^{(1)}\right] \equiv E < \infty$.

Now $\vec{l}_n$ is the unit vector in the direction of $\vec{x}_{n+1} - \sum_{k=1}^{n} \frac{\vec{x}_k}{n}$, so it suffices to show that for all $j > 1$, $P\left(\left|\frac{x_{n+1}^{(j)} - \sum_{k=1}^{n} \frac{x_k^{(j)}}{n}}{x_{n+1}^{(1)} - \sum_{k=1}^{n} \frac{x_k^{(1)}}{n}}\right| \geq \varepsilon\right) \to 0$ as $n \to \infty$ for any $\varepsilon > 0$. Rewriting the numerator and denominator in terms of $\vec{\Delta}_n$,

$$P\left(\left|\frac{x_{n+1}^{(j)} - \sum_{k=0}^{n} \frac{x_k^{(j)}}{n}}{x_{n+1}^{(1)} - \sum_{k=0}^{n} \frac{x_k^{(1)}}{n}}\right| \geq \varepsilon\right) = P\left(\left|\frac{\sum_{i=1}^{n} \Delta_i^{(j)} - \frac{1}{n}\sum_{k=1}^{n}\sum_{i=1}^{k} \Delta_i^{(j)}}{\sum_{i=1}^{n} \Delta_i^{(1)} - \frac{1}{n}\sum_{k=1}^{n}\sum_{i=1}^{k} \Delta_i^{(1)}}\right| \geq \varepsilon\right)$$

$$= P\left(\left|\frac{\sum_{i=1}^{n} \Delta_i^{(j)} - \frac{1}{n}\sum_{k=1}^{n}(n-k+1)\Delta_k^{(j)}}{\sum_{i=1}^{n} \Delta_i^{(1)} - \frac{1}{n}\sum_{k=1}^{n}(n-k+1)\Delta_k^{(1)}}\right| \geq \varepsilon\right) = P\left(\left|\frac{\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)\Delta_k^{(j)}}{\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)\Delta_k^{(1)}}\right| \geq \varepsilon\right)$$

$$= P\left(\left|\frac{\frac{1}{n}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)\Delta_k^{(j)}}{\frac{1}{n}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)\Delta_k^{(1)}}\right| \geq \varepsilon\right)$$

Let $A_n = \frac{1}{n}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)\Delta_k^{(j)}$ and $B_n = \frac{1}{n}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)\Delta_k^{(1)}$. We have

$$E[A_n] = E\left[\frac{1}{n}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)\Delta_k^{(j)}\right] = \frac{1}{n}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)E\left[\Delta_k^{(j)}\right] = 0$$

and

$$Var[A_n] = Var\left[\frac{1}{n}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)\Delta_k^{(j)}\right] = \frac{1}{n^2}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)^2 Var\left[\Delta_k^{(j)}\right] = \frac{C}{n^2}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)^2$$

$$= \frac{C(n-1)(2n-1)}{4n^3} \to 0$$

as $n \to \infty$, while

$$E[B_n] = E\left[\frac{1}{n}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)\Delta_k^{(1)}\right] = \frac{1}{n}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)E\left[\Delta_k^{(1)}\right]$$

$$= \frac{M}{n^2}\sum_{k=1}^{n}(k-1) = \frac{M}{2}\left(1+\frac{1}{n}\right) \to \frac{M}{2} > 0$$

as $n \to \infty$ and

$$Var[B_n] = Var\left[\frac{1}{n}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)\Delta_k^{(1)}\right] = \frac{1}{n^2}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)^2 Var\left[\Delta_k^{(1)}\right] \le \frac{E}{n^2}\sum_{k=1}^{n}\left(\frac{k-1}{n}\right)^2$$

$$= \frac{E(n-1)(2n-1)}{4n^3} \to 0$$

as $n \to \infty$. Therefore

$$P(|A_n| \ge \varepsilon) \le \frac{Var[A_n]}{\varepsilon^2} \to 0$$

by Markov's Inequality, so $A_n \xrightarrow{P} 0$, and also

$$P\left(\left|B_n - \frac{M}{2}\right| \ge \varepsilon\right) \le \frac{E\left[\left(B_n - \frac{M}{2}\right)^2\right]}{\varepsilon^2} = \frac{E\left[\left(B_n - \frac{M}{2}\left(1+\frac{1}{n}\right) + \frac{M}{2}\left(1+\frac{1}{n}\right) - \frac{M}{2}\right)^2\right]}{\varepsilon^2}$$

$$\le \frac{2 \cdot Var[B_n] + 2\left(\frac{M}{2}\left(1+\frac{1}{n}\right) - \frac{M}{2}\right)^2}{\varepsilon^2} \to 0$$

so $B_n \xrightarrow{P} \frac{M}{2} > 0$. Therefore for a subsequence $\{n_a\}$ there is a further subsequence $\{n_{a_b}\}$ such that $A_n \to 0\ a.s.$, and given the subsequence$\{n_{a_b}\}$ there exists an even further subsequence $\{n_{a_{b_c}}\}$ such that $B_n \to \frac{M}{2} > 0\ a.s.$. Thus given the subsequence $\{n_a\}$ there exists the subsequence $\{n_{a_{b_c}}\}$ such that $\frac{A_n}{B_n} \to 0\ a.s.$ and therefore $\frac{A_n}{B_n} \xrightarrow{P} 0$.

□

The above proof is for functions with a constant direction of improvement; in general this is not the case. Planes, however, do behave in this manner, and differentiable functions can be approximated locally by their tangent plane. One can therefore hypothesize that since ARS spends most iterations searching in a fairly localized area, the last few improvement points of the ARS visit record will behave as if the objective function were a plane, and therefore can still give an approximation of the current direction of improvement. We do not prove this, but rather use it to motivate the improved ARS algorithm presented below and demonstrate this improvement with copious numerical simulations

1.4.2 The DARS Heuristic

The idea behind DARS is that the balls (or hyper-cubes) used by the ARS algorithm weight all directions equally, and do not incorporate information from past iterations in any way. In the case of a function with improvement regions which are of

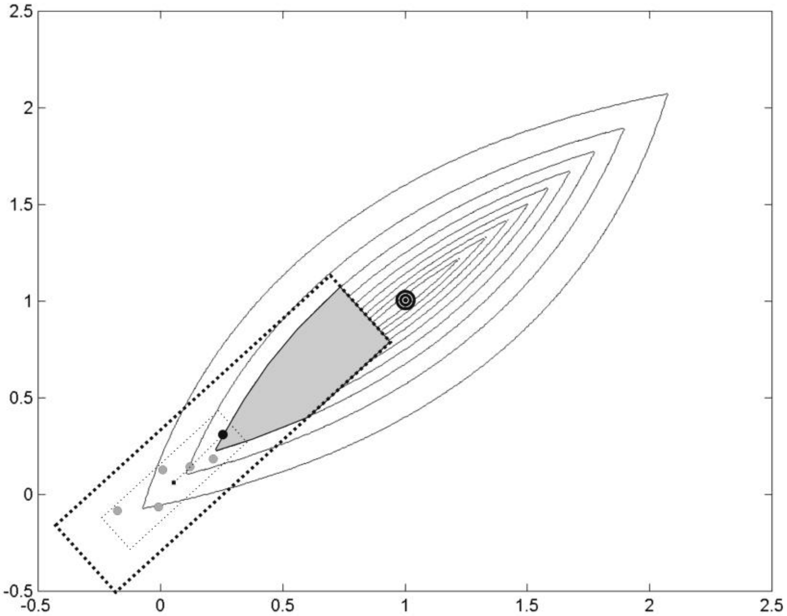


Figure 1.2 DARS optimizing a "thin" function. The ratio and orientation of the box is determined using past values. The shaded area is again the acceptance region, which is now both proportionally larger to the search box and is closer to the optimal point.

similar scale in all directions, this approach works well. However, when this is not the case, successive improvements will differ little in some directions and more in others; indeed, Theorem 1.4.1 implies that ARS improvements will point in the right direction. This information can be used to bias search region in a particular direction, which is exactly what DARS does. See Figure 3.2, and note how not only has the probability of the new acceptance region over doubled that of ARS in Figure 3.1, but that the DARS search region allows for the possibility of points much closer to optimal point than the ARS search region does. We now present the DARS heuristic:

PHASE 1: INITIALIZATION

In addition to the previous parameters used in ARS, we incorporate an additional tunable parameter, K, which will represent the number of past improvement points we will use to determine the aforementioned directionality. In both theory and practice below, $K = 10$ was found to be a reasonable value. The DARS algorithm initializes using ARS:

(1) Set $I = 0$, $n = 1$, and $r = R$. Generate $\vec{x}_1$ from the uniform distribution on Ω.

(2) Given $\vec{x}_n \in \Omega$ and $r \in (\rho, R]$, generate $\vec{y}$ from $U(B(\vec{x}_n, r))$.

(3) If $F(\vec{y}) > F(\vec{x}_n)$ then let $\vec{x}_{n+1} = \vec{y}$, $I = I + 1$, and $r = R$.

(4) If $r < \rho$, then $r = R$. If $I \geq K + 1$, then end Phase 1. Let $n \to n + 1$ and go to (2).

PHASE 2: DIRECTIONALIZATION

We now detail how DARS incorporates directionality. We denote the unit vector in the direction of the vector joining two points X and Y by $\overrightarrow{X, Y} = \frac{\langle x_1 - y_1, x_2 - y_2, \ldots, x_d - y_d \rangle}{\sqrt{\sum (x_i - y_i)^2}}$.

We will be using the fact that the equation of a plane with unit normal vector $\vec{n}$ through a point P is $\overrightarrow{X,P} \cdot \vec{n} = 0$, and that the distance from a point Q to that plane is $D(Q; P, \vec{n}) = \overrightarrow{P,Q} \cdot \vec{n}$.

(1) Calculate $\overrightarrow{A, x_n}$, the unit vector in the direction of the vector joining the average of the last K improvements $\vec{A} = \frac{1}{K}\sum_{k=n-K}^{n-1} \vec{x}_k$ with the current best point $\vec{x}_n$.

(2) Use the Gram-Schmidt process with initial vector $\overrightarrow{A, x_n}$ to develop a new orthonormal basis $\{\overrightarrow{u_i}\}_{i=1}^{d}$, with $\overrightarrow{u_1} = \overrightarrow{A, x_n}$.

(3) For each new basis member, calculate the average distance from the past improvement points to the resulting plane through the average improvement point: $D_j = \frac{1}{K}\sum_{k=n-K}^{n-1} D\left(\vec{x}_k; \vec{A}, \overrightarrow{u_j}\right)$.

(4) Let $Rect\left(\vec{x}_n, \{r_i\}_{i=1}^{d}\right)$ be the hyper-rectangular region centered at $\vec{x}_n$ made up of subsets of planes perpendicular to the vectors in $\{\overrightarrow{u_i}\}_{i=1}^{d}$ with the ith parallel faces a distance r_i apart. For $j \neq i$ let $r_j = \frac{R \cdot max\left(\frac{D_1}{D_j}, \rho\right)}{max\left(min_j\left(\frac{D_1}{D_j}\right), \rho\right)}$. This is done to ensure that no ratio r_j is less than ρ, and that the hyper-rectangle encompasses the full search region to start.

PHASE 3: ARS-LIKE SEARCH REGIONS

Finally, we search in hyper-rectangular regions geometrically similar to $Rect\left(\vec{x}_n, \{r_i\}_{i=1}^{d}\right)$ in an ARS manner:

(1) Given $\vec{x}_n \in \Omega$ generate $\vec{y}$ from the uniform distribution on $Rect(\vec{x}_n, \{r_i\}_{i=1}^{d})$.

(2) If $F(\vec{y}\,) > F(\vec{x}_n)$ then let $\vec{x}_{n+1} = \vec{y}$ and go to Phase 2. Otherwise, let $r_j = r_j / c \ \forall \, j$.

(3) If any $r_j < \rho$, then let that $r_j = \rho$. If all $r_j = \rho$, go to Phase 2, (4). Otherwise, let $n \to n + 1$ and go to (1).

Thus DARS adapts its directionality after each improvement in order to optimize the probability of getting a point closer to the optimal point, but in all other ways remains faithful to the ARS idea of searching the entire space infinitely often while still searching close to the current best point more often than not. As such we have the identical proof of consistency as for Proposition 1.3.1, now applied to the novel algorithm DARS:

PROPOSITION 1.4.1 (CONSISTENCY OF DARS)

Let $\left\{\vec{x}_n^{\ (DARS)}\right\}_{n=0}^{\infty}$ *be the record sequence associated with the DARS algorithm with* $Diam(\Omega) < \infty$. *Then* $P\left(\left\{\lim_{\mathrm{n}\to\infty}\left(F\left(\vec{x}_n^{\ (DARS)}\right)\right) = F^*\right\}\right) = 1$

Proof:

An infinite subsequence of points generated during the DARS algorithm will be from $U(\Omega)$; call these points $\left\{\vec{x}_{n_k}^{\ (DARS)}\right\}$. These points are precisely a visit record from the PRS algorithm above, so we know immediately that $P\left(\left\{\lim_{\mathrm{k}\to\infty}\left(F\left(\vec{x}_{n_k}^{\ (ARS)}\right)\right) = F^*\right\}\right) = 1$ from Proposition 1.2.1. In particular this means that $P\left(\left\{sup_k\left(F\left(\vec{x}_{n_k}^{\ (DARS)}\right)\right) = F^*\right\}\right) = 1$, which means $P\left(\left\{sup_n\left(F\left(\vec{x}_n^{\ (DARS)}\right)\right) = F^*\right\}\right) = 1$ as well since $sup_k\left(F\left(\vec{x}_{n_k}^{\ (DARS)}\right)\right) \leq sup_n\left(F\left(\vec{x}_n^{\ (DARS)}\right)\right) \leq F^*$.

Finally $\left\{\vec{x}_n^{\ (DARS)}\right\}_{n=0}^{\infty}$ forms a non-decreasing sequence, so $P\left(\left\{sup_n\left(F\left(\vec{x}_n^{\ (DARS)}\right)\right) = F* =1\right.\right.$ is equivalent to Plimn$\to\infty Fxn(DARS)=F*=1$ □

In the next section, we demonstrate that, in implementation, DARS indeed outperforms ARS in crucial situations where narrow acceptance regions are present.

1.5 Numerical Results

For each of the following simulations, a version of the C++ code in Appendix A was used. When ARS is referred to, we mean finite descent ARS with lowest level at machine precision 10^{-14}. Similarly, when DARS is referred to, we mean finite descent, bounded ratio DARS with both the lowest level and minimum ration set to machine precision 10^{-14}. In all cases, we use a shrinking constant for both ARS and DARS of 2^{-D} for a function with D-dimensional input values. Also in all cases, the number of past iterations used to determine the orientation of the DARS box was 10; we found this to be a good balance between using past information, which not letting the algorithm get off

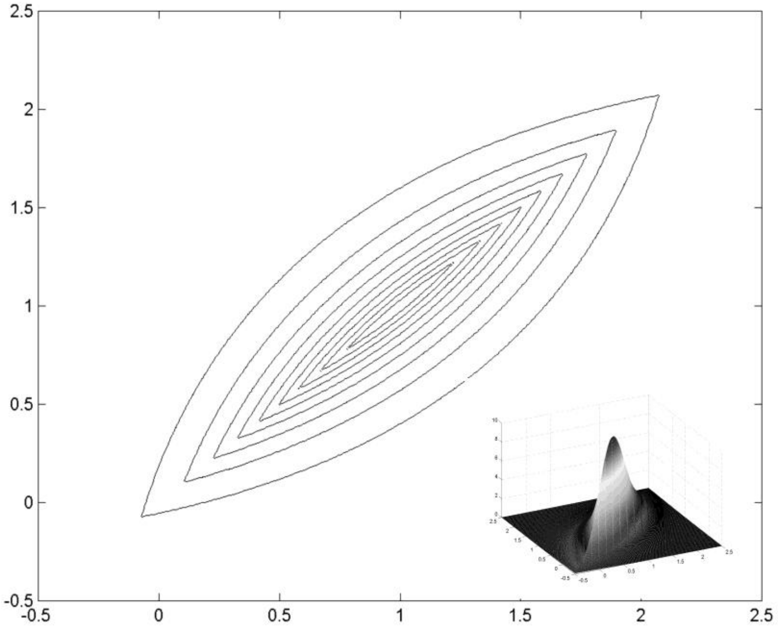


Figure 1.3 A contour plot of an example demonstrative "thin" function, with surface plot inset.

track by past mistakes. In reality, since this number is a tunable parameter, there are no doubt values greater or less than 10 which would yield better results on specific functions. However, there is more value to an algorithm capable of performing well independent of an optimized parameter, and since DARS is such an algorithm, we prefer to present these results with as little manipulation as possible.

1.5.1 Demonstrative Function

We begin with a function designed specifically to demonstrate the relative effectiveness of DARS to ARS on functions with differing length scales. Consider the family of functions of the form

$$f(x,y) = Ae^{-b_1|cos\,\theta(x-1)-sin\,\theta(y-1)|-b_2(cos\,\theta(y-1)+sin\,\theta(x-1))^2}$$

See Figure 1.3 for an example. These functions have a global maximum of A at (1,1), but have differing lengths scales if one approaches from the “side” versus along the “edge.” Note also that the choice of θ changes the orientation of the “edge,” so if DARS is to be successful it will need to properly converge to the direction of improvement. Additionally, these functions are not differentiable at their local maximum, so standard gradient methods cannot work. One thousand functions were generated with uniform randomly chosen $A \in (0,10)$, $b_1 \in (0,100)$, $b_2 \in (0,100)$, and $\theta \in \left(0, \frac{\pi}{2}\right)$.

Both ARS and DARS were then used to find the local maximum of each function. We calculate the relative error

$$\Delta = \frac{f_{max} - f_n}{f_{max}}$$

after n iterations for $n = 100$, 1000, 10000, and 100000. Then, we find the quartile bounds for each value of n for both algorithms. The numerical results in total are

	n=	Min	Q1	Med	Q3	Max
ARS	100	1.37E-04	1.24E-01	4.60E-01	9.40E-01	1.00E+00
	1000	5.42E-09	1.32E-02	8.03E-02	3.54E-01	1.00E+00
	10000	4.01E-09	7.91E-04	5.75E-03	3.10E-02	7.77E-01
	100000	9.01E-11	1.99E-05	1.55E-04	8.34E-04	2.62E-01
DARS	100	1.44E-04	1.31E-01	4.09E-01	8.68E-01	1.00E+00
	1000	2.68E-09	7.14E-06	5.52E-05	5.26E-04	9.93E-01
	10000	1.00E-14	1.67E-11	5.21E-11	1.20E-10	1.87E-02
	100000	1.00E-14	1.00E-14	1.00E-14	1.82E-11	9.25E-11

Table 1.1 Relative error quartiles on 1000 trials of ARS and DARS optimizing the demonstrative, "thin" functions.

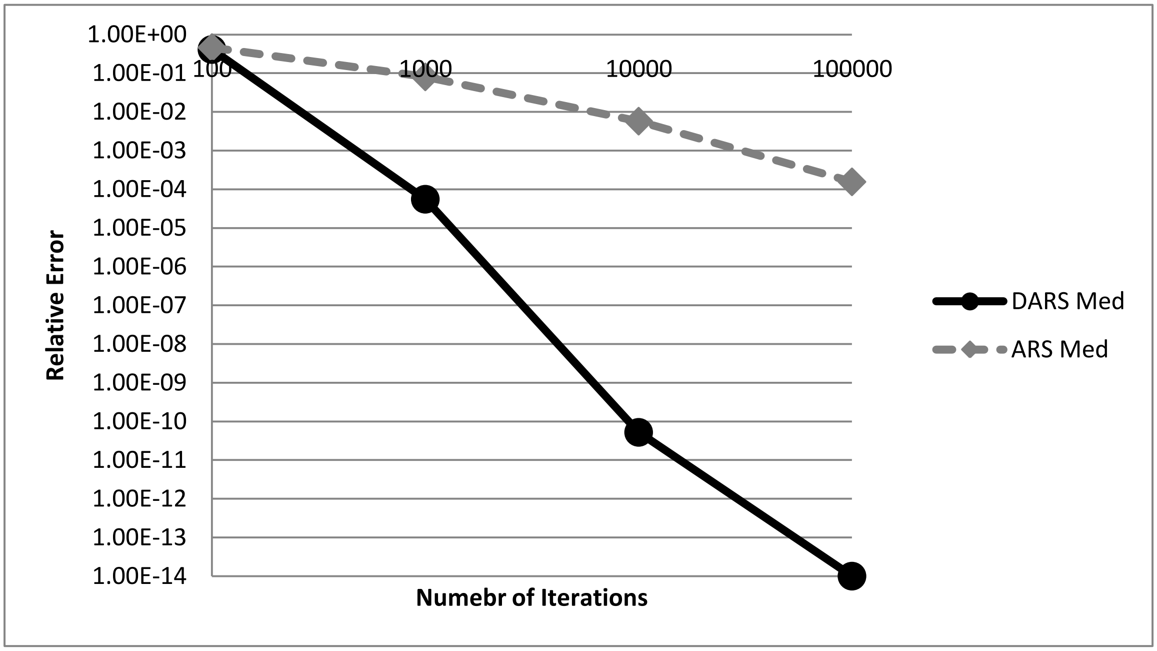


Figure 1.4 Plots of the median values over 1000 trials of the relative error of ARS and DARS on the demonstrative, "thin" functions

available in Table 1.1. Note that while at 100 iterations the algorithms are indistinguishable, by 1000 iterations the worst outcome for DARS is a full power of ten more accurate that the 75th percentile of the ARS outputs. By 100000 iterations, the worst DARS output is as accurate as the best ARS output; the best ARS output is three orders of magnitude worse than the best DARS output. At 100000 iterations, the median output of ARS is a full 10 orders of magnitude worse than that of DARS, with more than 50% of the DARS trials resulting in perfect accuracy to the machine precision used, 10^{-14}. At its worst, ARS does not find the optimal point to even one decimal point of accuracy, while DARS always finds it to at least ten places. The median results are plotted in Figure 1.4.

1.5.2 Benchmark Functions

Almost any introductory paper for a new algorithm demonstrates its effectiveness on a series of canonical functions agreed by the optimization community to be standard benchmarks with which to compare disparate algorithms. Next, we present two such sets of functions that have been used recently in the literature. The functions themselves are available in Appendix B.

THE DIXON AND SZEGÖ FUNCTIONS

These nine functions, first presented in 1978[39], were then reprinted with a summary of the results of numerous optimization algorithms in 1999[40], and then reprinted with an additional algorithm added in 2003[24]. In Table 1.2, we reproduce these results yet again, with a row for ARS and DARS now added. The table indicates the average number of function evaluations n over 25 trials that were necessary to meet the termination criterion

Method	S5	S7	S10	H3	H6	GP	BR	C6	SHU
Bremmerman	LM	LM	LM	LM	LM	LM	250	NA	NA
Mod. Bremmerman	LM	LM	LM	LM	515	300	160	NA	NA
Zilinskas	LM	LM	LM	8641	NA	NA	5129	NA	NA
Gomulka-Branin	5500	5020	4860	NA	NA	NA	NA	NA	NA
Torn	3679	3606	3874	2584	3447	2499	1558	NA	NA
Gomulka-Torn	6654	6084	6144	NA	NA	NA	NA	NA	NA
Gomulka-V.M.	7085	6684	7352	6766	11125	1495	1318	NA	NA
Price	3800	4900	4400	2400	7600	2500	1800	NA	NA
Fagiuoli	2514	2519	2518	513	2916	158	1600	NA	NA
De Biase-Frontini	620	788	1160	732	807	378	587	NA	NA
Mockus	1174	1279	1209	513	1232	362	189	NA	NA
Belisle et. al.	NA	NA	NA	339	302 (70%)	4728	1846	NA	NA
Boender et. al.	567	624	755	235	462	398	235	NA	NA
Snyman-Fatti	845	799	920	365	517	474	NA	178	NA
Kostrowicki-Piela	NF	NF	NF	200	200	120	NA	120	NA
Yao	NA	NA	NA	NA	NA	NA	NA	1132	< 6000
Perttunen	516	371	250	264	NA	82	97	54	197
Perttunen-Stuckman	109	109	109	140	175	113	109	96	LM
Jones et. al.	155	145	145	99	571	191	195	285	2967
Storn-Price	6400	6194	6251	476	7220 (24)	1018	1190	416	1371
MCS (Best Version)	83	129	103	79	111	81	41	42	69
EM	3368	1782	5620	1114	2341	420	315	233	358
ARS	74942 (med)	6247 (med)	106063 (med)	2663	12082 (9)	368	231	191	378
DARS	1307 (med)(17)	10860 (med)(19)	47795 (med)(23)	1003	5990 (7)	368	211	264	295

Table 1.2 Comparison of optimization methods on the Dixon and Szegö functions. The following code is used: LM: *Converged to local minimum;* NA: *No information available;* (med): *Median was used instead of average;* (#): *Number of trials over which the algorithm was successful (if not all 25) Italicized methods are those requiring first or second order differentiability.*

$$\frac{f_{max} - f_n}{f_{max}} \leq 10^{-4}$$

Note that most of the algorithms in Table 3.2 are directional in nature, and seven of them require first or second order differentiability. Also note the values in the table correspond to these methods being used with restarts of various iteration lengths. Neither ARS nor DARS here use restarts here, in order to demonstrate that both of these algorithms will always find the optimal point eventually, even if it takes longer. However, the lack of restarts negatively impact the performance of ARS and DARS on functions with spaced-out sub-optimal minima with values close to optimal. It is for this reason that we see apparently inferior performance for both algorithms on the Shekel functions [S5], [S7], and [S10] and on the Hartman function [H6]. In the case of the Shekel functions we use the median value for ARS and DARS rather than the average, since the average is skewed by a few much larger outliers and does not accurately reflect the performance. For the Shekel functions DARS prematurely converged to a local minimum in 8 trials for [S5], 6 trials for [S7] and 2 trials for [S10], negative performances that would also be avoided through restarts. Both DARS and ARS performed quite well on certain trials of these functions: seven ARS trials and twelve DARS trials for [S5] were below 1500 iterations, reinforcing the fact that use of restarts would cause the median number of iterations to drop precipitously.

For the test functions [GP], [GR], [C6], and [SHU], ARS and DARS perform comparably to both the other various algorithms presented and to each other, performing better than algorithms requiring differentiability even with restarts. When DARS did converge to the proper global minimum, in seven of the nine cases it did so faster than ARS, and in the other two cases the results were comparable. We note in comparing the performance of ARS to DARS on the Shekel functions that the altered search regions result in an increased propensity to converge to local minima for DARS; however, since DARS is consistent, this will only cause a delay in convergence. In those functions where ARS outperforms DARS it does so by under a factor of two, indicating that any additional attraction to local minima in the DARS algorithm no more than doubles the number of function evaluations needed. In summary, DARS performs on par with ARS, and is superior to it more often than not. ARS and DARS were able to converge to global optima without restarts, while most of the other methods in Table 3.2 could not do so. Even without restarts and without using any differentiability, ARS was able to outperform some algorithms on over half of the benchmark functions, and on all but one of those DARS was able to outperform ARS.

THE ARS TEST FUNCTIONS

When the ARS algorithm was published, it was compared to PRS on several test functions used throughout optimizations literature[28, 41-45]. In particular, because the primary purpose of that paper was to compare the convergence rates of ARS to PRS, the experiment performed therein was the following: run PRS for 1,000,000 iterations, and then determine the average number of function iterations required for ARS to overtake it. Since our primary purpose here is to compare DARS with ARS, we therefore apply a similar conceit: ARS was run for 1000 iterations, and its final value recorded. Then, DARS was run until it reached a value better than that of ARS, and the number of iterations was recorded. This procedure was performed 1000 times, and the quartiles of the results for each test function were calculated. The results are in Table 1.3.

	FR	**G1**	**G2**	**Gw**	**Him**	**JS**	**Rast**	**Ros**
Q1	235	227	200	561	868	764	147	58
Med	319	504	489	2082	1071	951	206	407
Q3	442	1601	1667	16298	1289	1168	308	690

Table 1.3 The number of iterations it takes DARS to outperform 1000 iterations of ARS on ARS functions; quartiles over 1000 trials. DARS outperforms ARS almost every time.

We see now that as good as ARS was on these functions in [1], DARS in general performs significantly better. For the Himmelblau and Jenrich-Sampson functions the median was almost exactly 1000, so that half of the time DARS reached the ARS value in fewer iterations, and half the time more. This result is consistent with performance identical to that of ARS. The only function for which DARS performed worse than ARS more than 50% of the time was Griewank's function, for which it took DARS a median value of about twice that of ARS. Griewank's function is one of the most rugged of the tested functions, and it is therefore not surprising that use of directionality caused misdirection on many trials. However, DARS outperformed ARS by half about 25% of the time on Griewank's function even so. On the remaining five functions tested, DARS outperformed ARS by at least half the number of iterations over 50% of the time; in the case of Rastrigin's function by one-third the number of iterations required more than 75% of the time. Of special note are the results on Rosenbrock's "banana" function, which contains a narrow curved valley with the relative minimum at its center. The fact that DARS outperformed ARS so well on this function is an indication of DARS's ability to "turn" its orientation in the proper direction adaptively to follow a changing direction of improvement. In general we see that DARS has superior marks to ARS on most of these test functions, many of which do not have acceptance regions with disparate length scales, and is never substantially worse. The implication of this result is that the incorporation of directional data increases the convergence rate of ARS on many functions, not just those it was specifically designed to improve it upon. And the fact that ARS does not appreciably dominate DARS indicates that the trade-off for the DARS improvement is minimal.

Notable in the publication of the ARS paper[1] was its inability to deal with solving linear systems of equations, because of the long, thin regions that may form when intersecting lines, planes, and hyperplanes. Although there are numerous approaches for solving linear systems without the use of optimization[46-48], it should be noted that computers often have difficulties with using these methods on nearly-singular systems since division by small numbers can occur. Furthermore once the systems become even slightly nonlinear, much of the theory is no longer valid. We therefore show that DARS is capable of addressing this weakness in the ARS algorithm and can thus be more useful in addressing these types of problems.

1.5.3   Systems of Equations

LINEAR SYSTEMS

We begin with standard DxD linear systems constructed with random coefficients designed to ensure that the D-tuple of 1s is the solution. This was done to ensure that the solution was within the 10-unit hypercube search region. Such a system would be the set of equations

	n=	10000		100000	
	D=	**ARS**	**DARS**	**ARS**	**DARS**
(a) Linear	2	974	1000	999	1000
	3	765	958	940	1000
	4	502	700	858	998
	5	224	95	698	908
(b) Power	2	955	987	992	1000
	3	701	900	917	997
	4	412	610	797	985
	5	143	95	598	851
(c) Exp	2	831	930	962	995
	3	444	712	689	946
	4	177	387	388	818
	5	42	71	182	650
(d) Cosine	2	851	913	976	997
	3	394	595	647	934
	4	132	238	312	710
	5	38	44	113	415

Table 1.4 The number of trials for which ARS and DARS were able to output an optimal value less than 10^{-8} *on each type of system of equations (the true optimal value is zero). DARS is often significantly more successful on all types of systems.*

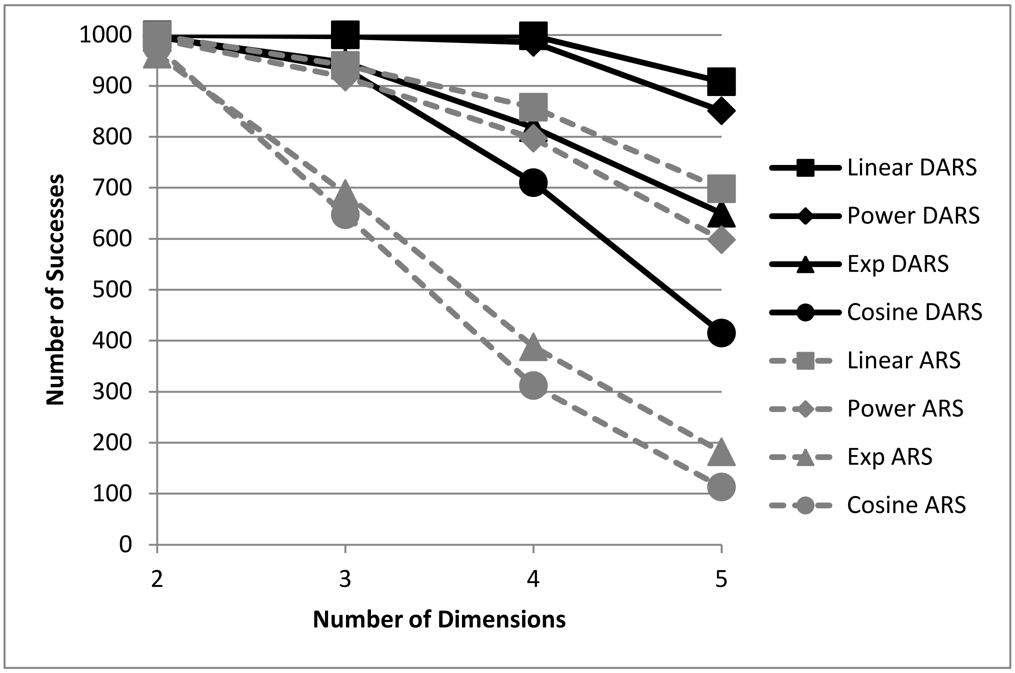


Figure 1.5 Plots of the data in Table 1.4. Once again it is clear that DARS substantially outperforms ARS.

$$\left\{\sum_{j=1}^{D} a_{ij}\left(x_j - 1\right) = 0\right\}_{i=1}^{D}$$

with a_{ij} being randomly chosen from the uniform distribution on [-1,1]. Note that the probability of having two equations generated whose coefficient each differ by at most ε (i.e., a nearly singular system) is $\frac{D(D-1)}{2}(2\varepsilon)^D$, small but by no means zero. We optimize the function

$$f(\vec{x}) = \sum_{i=1}^{D}\left(\sum_{j=1}^{D} a_{ij}\left(x_j - 1\right)\right)^2$$

which clearly has a global minimum value of 0 at the solution to the above system. In this section we define a "success" as being within 10^{-8} of the optimal value, zero. In

Table 1.4a we show the number of successes for the ARS and DARS algorithms at the number of iterations $n = 10000$ and 100000 and $D = 1$, 2, 3, 4, and 5. First note that ARS never has a 100% success rate, even for $D = 2$. The success rates at $n = 100000$ are always in favor of DARS; for $D = 5$, DARS managed to have a 90% success rate, while ARS managed only a 70% success rate.

SYSTEMS WITH POWERS OF X

The next type of system we address is nonlinear:

$$\left\{\sum_{j=1}^{D} a_{ij}\left({x_j}^{\gamma_{ij}} - 1\right) = 0\right\}_{i=1}^{D}$$

with a_{ij} being randomly chosen from the uniform distribution on [-1,1] and γ_{ij} being randomly chosen from the uniform distribution on [1,10]. We optimize

$$f(\vec{x}) = \sum_{i=1}^{D}\left(\sum_{j=1}^{D} a_{ij}\left({x_j}^{\gamma_{ij}} - 1\right)\right)^2$$

In Table 1.4b we show the number of successes for the ARS and DARS algorithms at the number of iterations $n = 10000$ and 100000. We note a slightly decreased overall success rate in both methods corresponding to the increased complexity of the geometry, which may include additional local extrema. However, ARS's success rate decreased more; for example, when $D = 5$ and $n = 100000$, ARS's success rate decreased by 10% to 60%, while DARS's success rate was only reduced by 5% to 85%.

SYSTEMS WITH EXPONENTIALS

The third type of system explored increases the nonlinearity further:

$$\left\{\sum_{j=1}^{D} a_{ij}\left(e^{-b_{ij}|x_j-1|}-1\right)=0\right\}_{i=1}^{D}$$

with a_{ij} being randomly chosen from the uniform distribution on [-1,1] and b_{ij} being randomly chosen from the uniform distribution on [1,10]. We optimize

$$f(\vec{x})=\sum_{i=1}^{D}\left(\sum_{j=1}^{D} a_{ij}\left(e^{-b_{ij}|x_j-1|}-1\right)\right)^2$$

This function is also not differentiable at infinitely many points, including the local minimum value 0. The results are in Table 1.4c. Once again, the results are similar to the linear case; we note a more substantial decrease than in the success rates overall than for the power systems above, but again ARS was more affected than DARS. In particular, at just $D = 4$ we see that DARS has over twice as many successes as ARS at both values of n, and the relationship is even more pronounced at $D = 5$ and $n = 100000$.

COSINE SYSTEMS

Finally, we look at systems with oscillatory functions (in this case cosine) while still preserving the unique solution at D-tuple of 1s:

$$\left\{\sum_{j=1}^{D} a_{ij}\left(\frac{cos\left(b_{ij}(x_j-1)\right)}{1+|x_j-1|}-1\right)=0\right\}_{i=1}^{D}$$

with a_{ij} being randomly chosen from the uniform distribution on [-1,1] and b_{ij} being randomly chosen from the uniform distribution on [0,1]. We optimize

$$f(\vec{x})=\sum_{i=1}^{D}\left(\sum_{j=1}^{D} a_{ij}\left(\frac{cos\left(b_{ij}(x_j-1)\right)}{1+|x_j-1|}-1\right)\right)^2$$

This function is again not differentiable at infinitely many points, including the local minimum value 0. In Table 1.4d, we once again see a reduction in success rates. For $D = 1$, 2, 3, and 4 this reduction is similar to the above exponential systems. For $D = 5$, however, the impact on DARS is more substantial, although it still over doubles ARS's success rate. We believe this is due to a similar phenomenon that occurred with Griewank's function above, namely that the substantial ruggedness caused by the local minima of multiple products of cosines can result in misdirection and therefore slower convergence by DARS. Unlike Griewank's function, however, DARS still substantially outperforms ARS, because the regions of differing length scale impact ARS far worse than ruggedness slows DARS's convergence.

In all cases, by 100000 iterations DARS outperformed ARS by as much as three times more successful runs in 1000 trials. See Figure 1.5. The performance of both algorithms depended only on the complexity of the geometry, and not on the differentiability of the objective function, and DARS demonstrates that it is capable of solving these difficult systems to a much larger degree than ARS. Decreases in success rates for DARS are most likely the result of attractive local minima and not acceptance regions of disparate dimensionality, since ARS began developing failures even for linear 3x3 systems, where few local extrema are possible. As with the benchmark functions, the performance of DARS could increase even further in a multiple-restart framework.

1.5.4 The Domains of Linear and Nonlinear Programming Problems

The final application we address here is to that of linear and nonlinear programming. In particular, we examine the following generalized constraints in two dimensions:

$$\{f_i(x, y) \geq 0\}_{i=1}^{n}$$

Optimizing a function under such constraints using gradient methods is problematic; along the edges of the region, gradient methods must recalculate gradients to stay within the region, a process which grows to be increasingly computationally complex as the constraints grow in difficulty and number, drastically increasing the time it takes for the algorithm to run. Further, in the nonlinear case there is sometimes no clear evidence that the desired constraints produce a region that contains any points at all, and in such cases finding points within the constraint region or determining if they even exist is a problem separate and in addition to optimizing some objective function on this domain. Herein, we offer a novel approach for finding points within the search domain, which could then be used to optimize some objective function on that region.

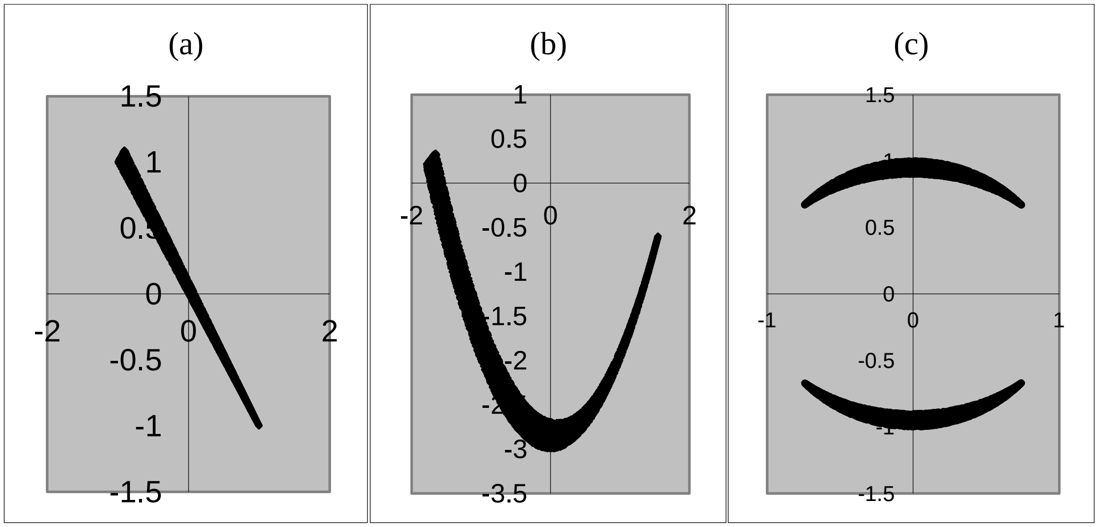


Figure 1.6 The desired domains in some programming problems discussed: (a) Linear, (b) Parabolic, (c) Elliptical

(a) **Linear**		h =	1	0.1	0.01
	ARS	Q1	19	49	127
		Med	31	135	2407
		Q3	47	355	5597
	DARS	Q1	18	49	119
		Med	30	133	314
		Q3	47	202	398
(b) **Parabolic**		h =	0.5	0.1	0.01
	ARS	Q1	10	21	48
		Med	17	35	75
		Q3	29	71	919
	DARS	Q1	10	22	47
		Med	17	36	74
		Q3	29	70	417
(c) **Elliptical**		h =	0.5	0.9	0.99
	ARS	Q1	17	34	74
		Med	26	52	123
		Q3	37	77	638
	DARS	Q1	17	34	74
		Med	26	50	123
		Q3	36	76	821

Table 1.5 Number of iterations required to find the domain region in programming problems discussed. Quartiles over 10000 trials.

THE METHODOLOGY

We wish to construct an optimization problem whose solution will be a point satisfying the above generalized nonlinear programming problem. However, a simple binary objective function

$$F(x,y)=\sum_{i=1}^{n}\mathbb{I}_{\{f_i(x,y)\geq 0\}}(x,y)$$

offers no causality or directionality whatsoever, and no algorithm would perform substantially better than PRS on such a function. Instead, we consider

$$F(x,y)=\sum_{i=1}^{n}\left(\begin{matrix}1, & f_i(x,y)\geq 0\\ \frac{1}{1+\left(f_i(x,y)\right)^2}, & else\end{matrix}\right\}\Bigg)\quad (*)$$

Both functions have a maximum value of n for any point within the desired domain. Note however in (*) how as $f_i(x,y)$ approaches zero, the value of the i^{th} contribution approaches 1, so there is now directedness incorporated into the problem. We consider three test cases and apply ARS and DARS to both of them.

The Linear Case

We consider the following family of domains of linear programming problems, parameterized by h:

$$\begin{gathered}x-y+2\geq 0\\ (-1-h)x-y+h\geq 0\\ x+y\geq 0\end{gathered}$$

See Figure 1.6a. We attempt to optimize (*) with respect to the above system using both ARS and DARS, for $h=1$, 0.1, and 0.01, and record the number of iterations it took to do so over 10000 trials. We present the quartile data in Table 1.5a. We see a substantial reduction in the ability of ARS to find the domain at $h=0.01$, when the search region becomes very thin; it takes a median value of over 2400 iterations to do so. While it does take longer for DARS to find the thinner regions as well, not substantially so, and this increase is expected given the decrease in area of the domain compared to the search space. In the extreme worst case, DARS was able to find the most narrow region $h=$

0.01 in 2164, whereas it took ARS as many as 24686 iterations to do so, a full order of magnitude worse.

The number of iterations required by DARS is, in fact, an improvement over simply picking random points in the search space (i.e., using PRS). To demonstrate this, we take the $h = 1$ case and note that this creates a triangular domain region of area $\frac{4}{3}$. The entire search area used was a square of side length 10, which has an area of 100. Therefore the probability of randomly picking a point in the domain area is $\frac{4/3}{100} \approx 0.0133$. The median number of iterations required by DARS was 30, meaning that 50% of the time DARS took 30 or fewer iterations to find the domain region. The probability of finding a point within the domain in thirty or few tries is only $\sum_{n=0}^{29}(0.0133)(1 - 0.0133)^n \approx 33\%$. The distinction only gets more pronounced as h is decreased.

THE PARABOLIC CASE

Next, we consider the nonlinear parametric family of domains

$$x - y + 2 \geq 0$$
$$(x - h)^2 - y + 3(h - 1) \geq 0$$
$$-x^2 + y + 3 \geq 0$$

this time for $h = 0.5$, 0.1, and 0.01. See Figure 3.7b. The family is of added interest since the domains it generates are not convex. The results for ARS and DARS are in Table 1.5b. Interestingly, we see almost identical behavior from ARS and DARS, and in many cases an improvement over the linear case. This is no doubt due to the overall increase in the size of the domain relative to the search region. However in the extreme case DARS still substantially outperforms ARS: in the narrowest tested region with $h = 0.01$, the third quartile of DARS iterations required was half that of the ARS iterations required. The largest number of iterations it took DARS to find this region was only

5763, but it took ARS as many as 66396 iterations to find it, again an order of magnitude worse than DARS.

The Elliptical Case

Finally we look at the following additional nonlinear parametric family of domains:

$$-x^2 - y^2 + 1 \geq 0$$
$$h^2x^2 + \frac{y^2}{h^2} - 1 \geq 0$$

Note that this family of domains is both not convex and not connected. See Figure 1.6c. The results for optimization when $h = 0.5$, 0.9, and 0.99 by ARS and DARS are in Table 1.5c. In this case ARS and DARS perform quite comparably, with DARS tying or edging ARS out in performance slightly in all but Q3 of the $h = 0.99$ region. In the worst case, ARS takes at most 4899 iterations to find this region, and DARS takes at most 4455, even fewer. Although the domain regions are thin in this case, the function (*) is in fact not, and so it is not surprising that DARS and ARS perform similarly. It should be noted that as in all previous applications when ARS does out-perform DARS, it does not do so to nearly the degree that DARS outperforms ARS, and in general causes convergence to be delayed by less than double. In all cases for all three types of regions, DARS performed well and could be used to generate points in the desired domain. ARS performed well with many regions, but as we have shown before performs poorly when the region is sufficiently narrow in one dimension relative to the other.

1.6 Conclusion

The ARS algorithm offers an incredibly powerful tool for solving optimization problems. While other algorithms may require differentiability in an objective function, ARS does not. While some may not always guarantee convergence to the global optimum, ARS does. When an algorithm can offer guaranteed convergence, one still does not know how long it will take. With ARS, it has been mathematically proven to converge exponentially on a large class of functions, and orders of magnitude better than PRS. In the general optimization context, one often cannot know the specific difficulties an objective function presents. In most cases, ARS will work regardless of the specifics of a function, and does not have behavior which depends drastically on optimizing parameters. If there is an optimization problem at hand, apply ARS to it and, while it will not always as fast as other algorithms, it will converge, and will almost always do so in an acceptable number of iterations.

The class of functions containing acceptance regions with differing length scales, however, does not benefit from the acceleration of ARS. And as stated above, one generally does not know when an objective function may exhibit such behavior. It is desirable, therefore, to have an algorithm with all the strengths of ARS, but removing this singular weakness. In the Chapter 1 above, it has been demonstrated that DARS is this algorithm. We have proven that improvement points from ARS can be used to direct the search region towards better points of improvements, which motivates the DARS heuristic. The numerical simulations comparing DARS with ARS show that one gains substantial speed of convergence on functions with “thin” acceptance regions, and indeed improving convergence speed over ARS in other functions as well. In trade, the DARS loses no consistency and has only exhibited having a small additional tendency for short-term convergence to local extrema. The implementation of a multiple-restart format would completely counteract this issue. That these results were demonstrated in the total absence of multiple restarts means that the performance of DARS presented here represents the true ability of the algorithm to use directional information to find the optimal point.

The importance of having widely applicable, effective optimization algorithms cannot be understated. Almost every science and engineering discipline contains problems which, fundamentally, far into the optimization context, be it to minimize cost or maximize output. The ARS algorithm has already been used in engineering and biotechnology applications[14-15, 38, 49], and the increase in effectiveness the DARS algorithm offers will only increase the application of this technique to future scientific endeavors.

CHAPTER 2

OPTIMIZATION AND NUMERICAL MODELING IN MEDICINAL CHEMISTRY

2.1 Introduction

The breadth and scope of the application of numerical optimization to other sciences is immense. In particular, optimization techniques combined and augmented with numerical modeling methods, offer new insights into the field of medicinal chemistry and drug discovery. Diverse aspects of the fields of medicinal chemistry and drug discovery are researched at the Torrey Pines Institute for Molecular Studies (TPIMS), and marrying this research to the aforementioned mathematical techniques has been fruitful[51-56]. In this chapter and the next, we will present some aspects of this joint research that spans numerous aspects of the fields of chemistry and biology.

2.1.1 Biological Activity

The ultimate goal of research in medicinal chemistry is to find chemical compounds that are **biologically active**. This description varies in meaning, but in general refers to a compound interacting in a desirable way with a biologically relevant molecular system. These interactions are tested in experiments called **assays** and can vary from simply testing the ability of the compound to bind to another molecule (the **target**, in what is known as a **binding assay**) to determining the compound's effect on certain types of cells (a **cell-based** assay) to determining a specific effect of the compound on a more complex living organism (an **in-vivo** assay).

One common way of numerically representing the activity of a compound is the **$\mathbf{IC_{50}}$**. The IC_{50} is the concentration of the compound which elicits a response which is 50% of the maximal response. So, for example, in a binding assay the IC_{50} would represent the concentration at which 50% of the compound is bound to the target. This is a generalization a general **dose-point** representation of activity, ED_X, which is the concentration at which the compound elicits a response that is X% of the maximal response.

Applied mathematical and statistical techniques can be quite useful in answering questions about biological activity. Even the determination of IC_{50} requires numerical determination of the maximal response and 50% point from the experimental data obtained. The proper management of experimental error, and the controlling for systematic errors, is also important. Finally in more complicated assays where multiple outputs are measured, choosing a **lead compound** (or a compound which is chosen for further testing) requires construction of the proper metric for determining which compound is "best."

2.1.2 Mixture-Based Combinatorial Libraries

One of the most important aspects of the research done at TPIMS involves the use of synthetic mixture-based combinatorial libraries[58-59]. A **mixture** (as we will use it in this dissertation) is a mixture of multiple compounds designed to be tested as though it were a single compound. In general, mixtures are useful because they allow multiple compounds to be tested simultaneously for activity. When batches of already-made compounds are mixed together and tested, this is generally referred to as **pooling**. Compounds maybe mixed in equal parts (**equimolar concentration**) or not (such as in **natural products**, or compounds and mixtures derived from nature rather than specifically designed and created). In contrast, **synthetic mixture-based combinatorial libraries** are synthesized directly to be equimolar amounts of thousands of compounds, systematically arranged by structure. Such libraries may be comprised of **peptides**, or chains of amino acids, or **small molecules**, which have different chemical structures.

Regardless, notation for labeling such mixtures is quite straight-forward. In a synthetic mixture-based combinatorial library compounds differ from one another in multiple places in the molecule, known as **positions**. For example, in a peptide, the order of the amino acids (first, second, third, etc.) would indicate an amino acid’s position. If, hypothetically, a library had the amino acids {A, R, S, T} to work with, then RATS would represent the specific peptide with R in the first position, A in the second, T in the third, and S in the fourth. The letter X is generally used to represent the equimolar mixture of all possible values in that position; in the above hypothetical example, RATX would then represent an equimolar mixture of RATA, RATR, RATS, and RATT, while RXXX would represent an equimolar mixture of all 64 compounds with R in the first position. The total number of compounds in a mixture is referred to as its **complexity**.

Since the goal is to find individual biologically active compounds, after mixtures are tested decisions must be made what to test next in a process known as **deconvolution**. For synthetic mixture-based combinatorial libraries this generally occurs in two ways. The first is **iterative**: Referring back to the above hypothetical library, in an iterative deconvolution one would first create AXXX, RXXX, SXXX, and TXXX. Whichever is the most active (say, RXXX) would be chosen, and then RAXX, RRXX, RSXX, and RTXX would be made. Complexity of the mixtures reduces with each stage, until individual compounds are tested and, ideally, an active individual is found. The second approach is **positional scanning**: The four sets of four mixtures {AXXX, RXXX, SXXX, TXXX}, {XAXX, XRXX, XSXX, XTXX}, {XXAX, XXRX, XXSX, XXTX), and {XXXA, XXXR, XXXS, XXXT} are all made and tested. The most active elements of each set are then combined; for example, if RXXX, XAXX, XXTX, and XXXS are the most active mixtures in their respective sets, then the individual compound RATS will be synthesized, tested, and ideally found to be active. Both methods have the benefit of having to perform far fewer biological assays than having to test all 256 compounds in the library individually; this increased efficiency only becomes more pronounced as the number of possible elements at each position, and the number of positions, increases. Either approach has been markedly successful in the finding of lead compounds[58-62, 69-77].

Applied mathematical and statistical techniques can be quite useful in the study of synthetic mixture-based combinatorial libraries and their deconvolution as well. In this chapter, we attempt to address some of the core issues associated with the use of mixtures: What level of activity is necessary to ensure that a given compound is found? Given the activity of a mixture, what percentage of it is active? How can we show the elements of a mixture are equimolar, and to what extent does this need to be the case?

2.2 The Modeling of Chemical Mixtures

The use of mixtures is not a new concept in drug discovery; for example, approximately 1,000 extracts derived from plants and used for the treatment of ailments are written about on tablets dating back to 2600 BC[57]. In modern drug discovery efforts mixtures are still being assessed in order to identify active compounds. The activity of a mixture is, of course driven, by the individual components comprising the mixture. To this extent it is critical to understand how the individual components of a mixture contribute to the overall activity of the mixture sample. The predictive capabilities of averaging models on such mixtures are examined here using 36 case studies from eight different publications.

2.2.1 The Harmonic Mean Model

The use of the harmonic mean as a method of determining the activity of a mixture, by combining the activity of that mixture's constituents, is not new[63-67]. As first described by Finney[63], the use of the harmonic mean as an averaging method is most mathematically suitable to model conditions based on the assumption of simple independent action. In fact, previous studies[66-67] have used the harmonic mean as a metric for determining the extent to which simple independent action is present in a mixture. Its usefulness when applied to modeling the behavior of mixture-based combinatorial libraries associated specifically with drug discovery, however, is worthy of study. In particular, the effect of the mathematical properties of the harmonic mean on the efficacy of the use of mixture-based combinatorial libraries in drug discovery merits explicit exploration. This study therefore begins with a comparison of how the harmonic mean differs from other classical averaging methods, and these methods' relative accuracies when applied to biological data where the assumption of simple independent action is appropriate.

The classical methods of averaging compared herein are the arithmetic mean, the geometric mean, and the harmonic mean. The arithmetic mean is defined by the equation:

$$A = \sum_{i=1}^{N} f_i X_i$$

where f_i is the proportion of the i^{th} mixture constituent with dosing point X_i. N is the total number of mixture constituents; if constituents are present in equal numbers, then $f_i = \frac{1}{N}$ for all i. Similarly, the geometric mean is given by:

$$G = 10^{\sum_{i=1}^{N} f_i log_{10}(X_i)}$$

and the harmonic mean by:

$$H = \frac{N}{\sum_{i=1}^{N} \frac{f_i}{X_i}}$$

Arithmetic meaning has been applied to simulated combinatorial libraries[68]. The geometric mean was calculated as an additional point of comparison, since it represents the use of the arithmetic mean on, for example, $\log_{10}(\text{IC}_{50})$ values. In this study, because of the synthetic methods used to prepare the mixtures[69-70], we assume that mixture constituents are present in equal proportions.

To demonstrate the relative effectiveness of the three above addition models, we use historical data which utilized the iterative process of deconvolution of mixture-based libraries[69-76] reported by TPIMS laboratories and other groups. These data are ideal to compare the performances of each averaging method as a model for the activity of mixtures, because both the IC50 value of each of the mixtures and all of the constituent

Mixture	**IC_{50} (nM)**	
Ac-rfwinx-NH_2	110	
Constituent Compounds	**IC_{50} (nM)**	
Ac-rfwink-NH_2	18	
Ac-rfwinr-NH_2	27	
Ac-rfwina-NH_2	37	
Ac-rfwins-NH_2	130	
Ac-rfwinp-NH_2	130	
Ac-rfwinn-NH_2	130	
Ac-rfwinq-NH_2	140	
Ac-rfwing-NH_2	170	
Ac-rfwinm-NH_2	180	
Ac-rfwinh-NH_2	200	
Ac-rfwint-NH_2	230	
Ac-rfwihy-NH_2	460	
Ac-rfwinl-NH_2	680	
Ac-rfwinf-NH_2	770	
Ac-rfwinw-NH_2	790	
Ac-rfwine-NH_2	960	
Ac-rfwind-NH_2	1,100	
Ac-rfwinv-NH_2	1,300	
Ac-rfwini-NH_2	5,600	
	Predicted Value	Scaled Error
Arithmetic Mean	687	5.24
Geometric Mean	265	1.41
Harmonic Mean	106	0.04

Table 2.1 An example of the type of data used in this study, in this case from [70]

submixtures (or individual compounds) in most cases were determined and have been reported. In total, 36 different mixtures, each consisting of 4 to 19 constituents, were analyzed. The number of compounds in the constituents of the mixtures studied ranged

Reference	**Mixture IC$_{50}$**	**Arithmetic Mean**	**Scaled Error**	**Geometric Mean**	**Scaled Error**	**Harmonic Mean**	**Scaled Error**
Houghten et. al. [69]	250	1,093	3.37	827	2.31	370	0.48
	41	1,322	31.26	1,016	23.79	75	0.83
	4.4	1,322.2	299.51	887.2	200.63	6.8	0.55
	0.38	11.19	28.44	4.90	11.89	0.44	0.17
Dooley et. al. [70	14,000	22,642	0.62	12,414	0.11	5,618	0.60
	1,500	4,663	2.11	2,494	0.66	1,586	0.06
	480	2,049	3.27	1,122	1.34	729	0.52
	110	687	5.24	265	1.41	106	0.04
Dooley et. al. [71]	2,701	14,962	4.54	7,989	1.96	3,954	0.46
	907	1,323	0.46	1,077	0.19	828	0.09
	106	316	1.99	231	1.18	151	0.42
	24	30	0.26	19	0.20	13	0.45
Pinilla et. al. [72]	1,000	738,795	737.80	168,921	167.92	865	0.14
	400	4,267	9.67	3,118	6.79	1,882	3.71
	48	69	0.43	40	0.17	25	0.49
	6.6	9.1	0.38	7.7	0.17	6.9	0.04
Houghten et. al. [73]	20,000	357,916	16.90	136593	5.83	13557	0.32
	860	1,089,794	1,266.2	560,506	650.75	1,705	0.98
	90	33,730	373.78	4,149	45.10	91	0.01
	6	15	1.45	12	1.04	10	0.72
Appel et. al. [74]	200,000	1,253,690	5.27	829,323	3.15	83,762	0.58
	7,329	840,278	113.65	208,754	27.48	5,813	0.21
	376	514	0.37	456	0.21	407	0.08
	257	299	0.16	255	0.01	230	0.11
	12,780	622,127	47.68	292,158	21.86	7,481	0.41
	408	841,453	2,061.4	374,148	916.03	702	0.72
	37	6,120	164.41	788	20.29	27	0.28
Davis et. al. [75]	30	83	1.75	73	1.45	61	1.03
	20	41	1.04	36	0.82	30	0.50
Ecker et. al. [76]	4	7.6	0.91	4.7	0.18	1.7	0.57
	0.5	7.5	14.08	3.5	6.00	0.6	0.15
	0.15	0.62	3.13	0.42	1.82	0.24	0.57
	0.08	0.28	2.53	0.18	1.22	0.11	0.37
	0.05	0.08	0.65	0.07	0.48	0.06	0.26
	0.03	0.04	0.35	0.04	0.27	0.04	0.17
	0.02	0.03	0.38	0.02	0.17	0.02	0.02

Table 2.2 A summary of the results of the three studied prediction models

from 1 to 6,859. In most cases, the measured IC$_{50}$ value was reported and was used for this analysis. For those constituents whose IC$_{50}$ is large and is only published as a lower bound, we use this lower bound as that constituent's IC$_{50}$ if it is greater than the best IC$_{50}$ of the previous iterative step. For example, in Houghten et. al.[69] the mixture Ac-

DVPAXX-NH_2 is reported as having an IC_{50} value ">1,400 μM," and Ac-DVPXXX-NH_2 has a reported IC_{50} value of 41μM , which is less than 1,400 μM, so the mixture Ac-DVPAXX-NH_2 was assigned a value of 1,400 μM . In contrast, in Davis et. al.[75] (egCB)(dG)XXT has a reported IC_{50} of ">10 μM" and (egCB)XXXT has a reported IC_{50} value of 40 μM, which is greater than 10. Data such as this is discarded since no accurate result would be determinable. For those constituents whose IC_{50} was too large to have any published data, we assign to them the IC_{50} of the least active measured compound. For example, in Dooley et. al.[70] the IC_{50} value of Ac-rfgxxx-NH_2 is reported as "ND" since it was inactive at the highest dose tested. It is thus assigned a value of 69,000 nM, that of the least active measured constituent.

For each mixture, the constituents of that mixture were added using each of the three addition models, and the results were compared to actual experimentally obtained mixture value. An example of these data, along with the outputs of each of the three addition models, is provided in Table 2.1. A summary of the results from all sources is presented in Table 2.2. Data points which were altered to have assigned values (as described above) are shown in italics.

Because the data is taken across mixtures with varying complexities and activities, the numerical scale for each data point varies widely. If the error in a model-predicted value were simply the difference between the prediction and the experimental value, they would not be numerically comparable. Therefore, the error was scaled by the experimental value, so that

$$Error_{Scaled} = \frac{\left| IC_{50}^{(Experimental)} - IC_{50}^{(Model)} \right|}{IC_{50}^{(Experimental)}}$$

Thus, the scaled error is as a fraction of the experimental value. The scaled error for each prediction is also included in Table 2.1 and 2.2. The average scaled error for the arithmetic mean model is 144.59, the average scaled error for the geometric mean model is 59.02, and the average error for the harmonic mean model is 0.47. Thus the harmonic mean was the only addition model that consistently was capable of capturing to within an order of magnitude the IC50 value of the resultant mixture given the IC50 value of that mixture's constituents. The maximum harmonic mean scaled error was only 3.71, as compared to maximum scaled errors of 916.03 for the geometric mean and 2061.38 for the arithmetic mean. The harmonic mean has lower scaled errors than both other methods for all but five of the analyzed mixtures; in those five cases, the scaled error is below 0.60 for the harmonic mean, indicating the harmonic mean gave reasonably good approximations in these cases as well.

To compare the ability of each of the three averaging models to predict the experimental mixture IC_{50} value, a least-squares linear regression was performed. Least-squares linear regression allows each model to be evaluated as a whole, rather than looking at individual predictions. Because of the large difference in scale amongst the data points, the regression was performed on the logarithms of the IC50 values. The functional form of the fit curves was therefore

$$log_{10}\left(IC_{50}^{(Experimental)}\right) = a_1 log_{10}\left(IC_{50}^{(Model)}\right) + a_0$$

Clearly, for a perfect model $a_1 = 1$ and $a_0 = 0$, and so a measure of how well the addition model is predicting the mixture IC50 is how close a_1 is to one and a_0 is to zero. Additionally, each fit also includes an R^2 value, representing the percentage of variance in the data that is explained by the model. For the arithmetic mean model, a_1 = 0.7250,

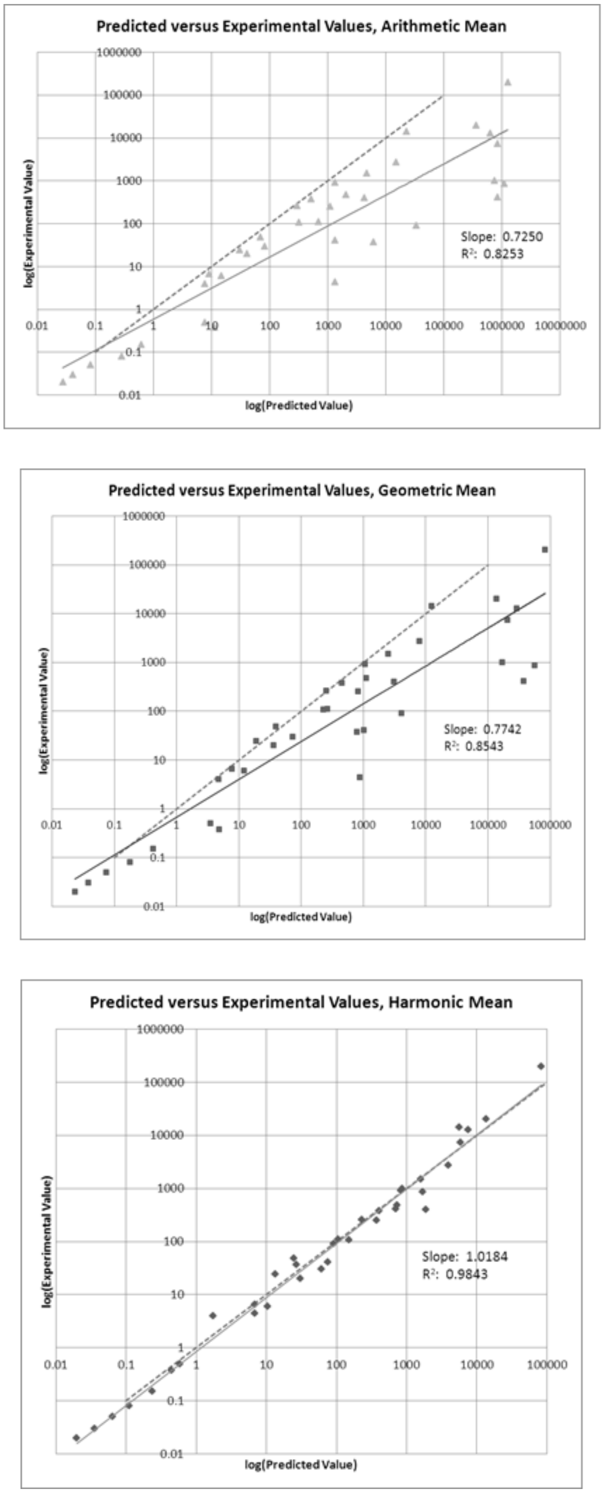


Figure 2.1 Log-log plots of the Model versus Experimental Values for each of the three studied models. The harmonic mean is clearly the best fit.

a_0 = -0.2291, and R^2 = 0.8253. For the geometric mean model, a_1 = 0.7742, a_0 = -0.1685, and R^2 = 0.8543. For the harmonic mean model, a_1 = 1.0184, a_0 = -0.0679, and R^2 = 0.9843. Plots of each of the addition model's predictions against the experimental values, along with the least-squares linear regression best fit line, are in Figure 2.1. The harmonic mean model both provides a better fit to the data given its superior R2 value, and also provides a slope and intercept very close to ideal.

2.2.2 Implications of the Harmonic Mean Model

The above analyses strongly suggest that the theory[63, 65-66] which predicts the harmonic mean as the averaging methodology for mixtures under the assumption of simple independent action is valid in the data analyzed above. It should be noted that in the data surveyed above the complexity of both the parent mixture and its constituents varies drastically from case to case. This variation does not, however, affect the accuracy of the harmonic mean, which yielded significantly more accurate results than the other two standard averaging methodologies studied. In most cases the harmonic mean predicted the experimental results with high precision. This fact strongly suggests that the harmonic mean is an appropriate way of modeling the behavior of many of the mixture-based combinatorial libraries used in basic research and drug discovery[68-77]. The ability to predict the outcomes of combinatorial mixture-based experiments is useful for a variety of practical dosing and experimental design applications, such as deviation from the harmonic mean as a metric for determining the existence of synergy or antagonism[66-67]. In the context of positional scanning, where each position is a different arrangement of the same constituents, the harmonic mean of the IC_{50} values at each position ought to equal one another. For example, Dooley and Houghten[77] presents positional scanning data for six positions. Each position consists of eighteen mixtures, each of which contains 1,889,568 individual compounds. The harmonic means of the IC50s of these mixtures at each position are 340, 271, 168, 248, 263, and 211. Thus it can be seen that the maximal scaled error in this set is only 1.02, and that most pair-wise comparisons have significantly smaller scaled errors. This simultaneously validates the integrity of the chemical synthesis and the accuracy of the biological measurements.

In addition to the above, the mathematical properties of the various averaging methods can help validate the usage of combinatorial mixture libraries in basic research and drug discovery and explain the impressive successes the process has already achieved[58-62, 69-77]. The harmonic mean model differs from the other models primarily in its treatment of extreme numerical ranges within the data; while at low ranges (with a ratio of most to least active less than 100) all three methods perform similarly, the scaled error of the arithmetic and geometric models rises steadily as the range increases, while the harmonic mean maintains a similar level of scaled error throughout. In particular, the harmonic mean is more influenced by active compounds having smaller IC_{50} values (and are therefore more active) than the other addition methods; that the experimental data is well-predicted by the harmonic mean indicates that the experimental behavior of mixture based combinatorial libraries behaves similarly, with active compounds driving the activity rather than inactive compounds diluting it.

To further elucidate this point, we consider the hypothetical situation in which we define an active compound to have a fixed IC_{50} value of α and an inactive compound to

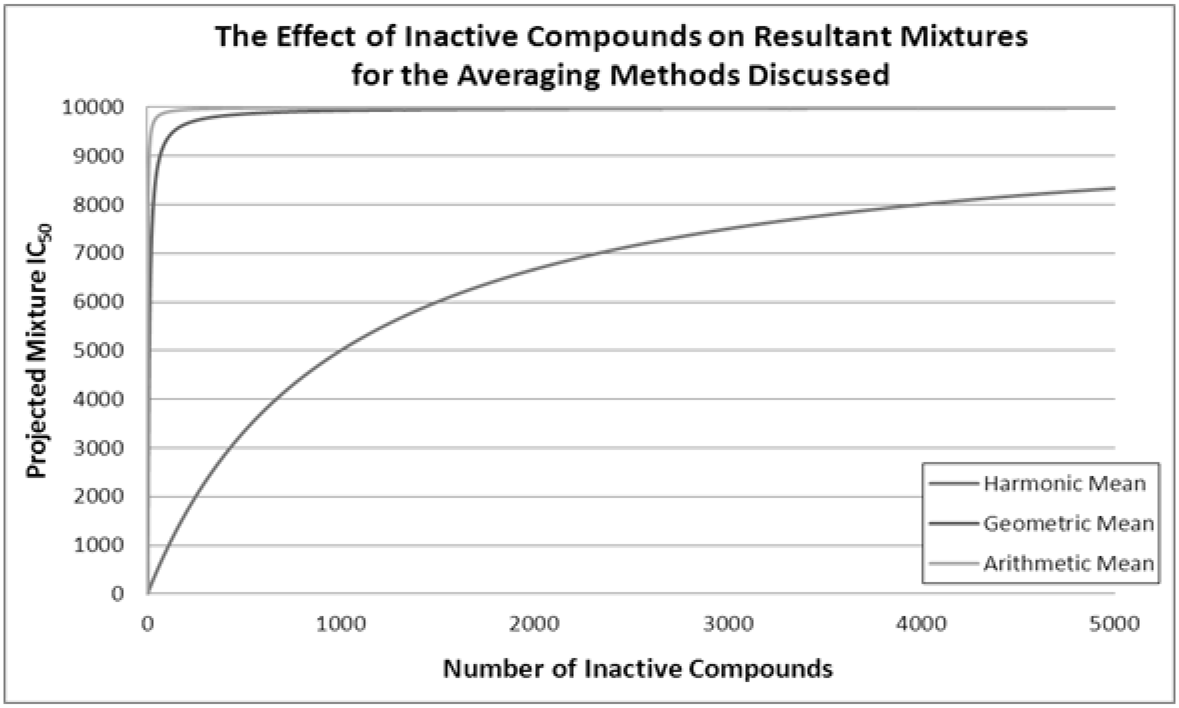


Figure 2.2 The activity of a mixture with a single 10 nM compound using each of the studied averaging models. The harmonic mean model shows that mixture activity is strongly influenced by even one active compound.

have a fixed IC$_{50}$ value of β. Then the IC$_{50}$ of a mixture containing N$_{\alpha}$ active compounds and N$_{\beta}$ inactive compounds is given by

$$A_{\alpha,\beta} = \frac{N_\alpha \alpha \ + \ N_\beta \beta}{N_\alpha + N_\beta}$$

for the arithmetic mean model,

$$G_{\alpha,\beta} = 10^{\frac{N_\alpha log_{10}(\alpha) + N_\beta log_{10}(\beta)}{N_\alpha + N_\beta}}$$

for the geometric mean model, and

$$H_{\alpha,\beta} = \frac{N_\alpha + N_\beta}{\frac{N_\alpha}{\alpha} + \frac{N_\beta}{\beta}}$$

for the harmonic mean model. From these equations, one can evaluate the ability to detect active compounds in a mixture governed by the harmonic mean model, and compare it to the abilities of the arithmetic and geometric mean models.

For example, if a single highly active compound with an IC50 of 10 nM (α=10 and N_α=1) were in a mixture with a number of inactive compounds with IC_{50}s of 10,000 nM (β=10,000), the aforementioned equations would yield a predicted value for the IC_{50} of the resultant mixture. Figure 2.2 is a plot of the number of inactive compounds versus the resulting activity of the mixture, given each of the averaging models. As is clear from the graph, the IC_{50} of the resultant mixture increases much more slowly for the harmonic mean than for the other two methodologies; therefore, mixtures that are modeled by harmonic mean averaging are projected to be significantly more active than if the arithmetic or geometric mean were used. A mixture comprised solely of poorly active compounds (all having an IC_{50} of 10,000nM) would have an IC_{50} of 10,000 nM in this scenario, so in order for the active compound to be detected the resulting mixture must have an IC_{50} value small enough to distinguish itself from the 10,000 nM mixture inclusive of experimental error. For example, if the experimental error is approximately 20%, then in order to ensure the mixture containing the active compound is more active than the 10,000 nM mixture when tested, it must have a true IC_{50} value of 6,600 nM or below. The arithmetic mean model would predict that only three 10,000 nM compounds would be needed to make the mixture with the 10 nM compound have an IC_{50} greater than 6,600 nM. Similarly, the geometric mean model would predict that only seventeen 10,000 nM compounds would be needed. In contrast, the harmonic mean model indicates that 1,939 compounds, each having an activity of 10,000 nM, would be needed. Since

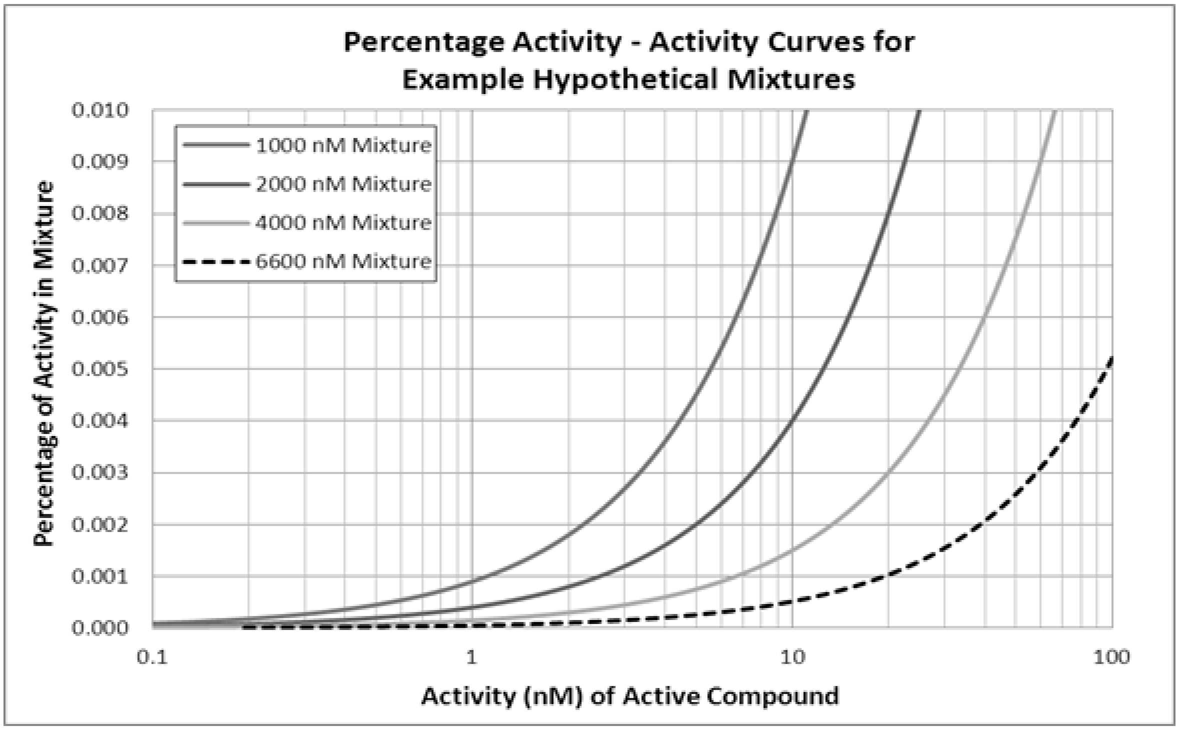


Figure 2.3 Extrapolated percentage of active compounds in a chemical mixture, given the harmonic mean model.

these numbers are independent of scale, these results may be restated: Under the arithmetic mean model, a mixture containing less than 25.0% active compounds would not be detected. Under the geometric mean model, a mixture containing less than 5.5% active compounds would not be detected. But under the harmonic mean model, a mixture containing 0.052% of active compounds would still be detectable. It should be noted that this observation can be applied to mixtures in which the constituents comprising the mixture are not present in the same concentration as in the case of natural product extracts. In other words as long as 0.052% of the total composition of the mixture contains an “active component” that mixture will be distinguishable from a totally inactive mixture sample.

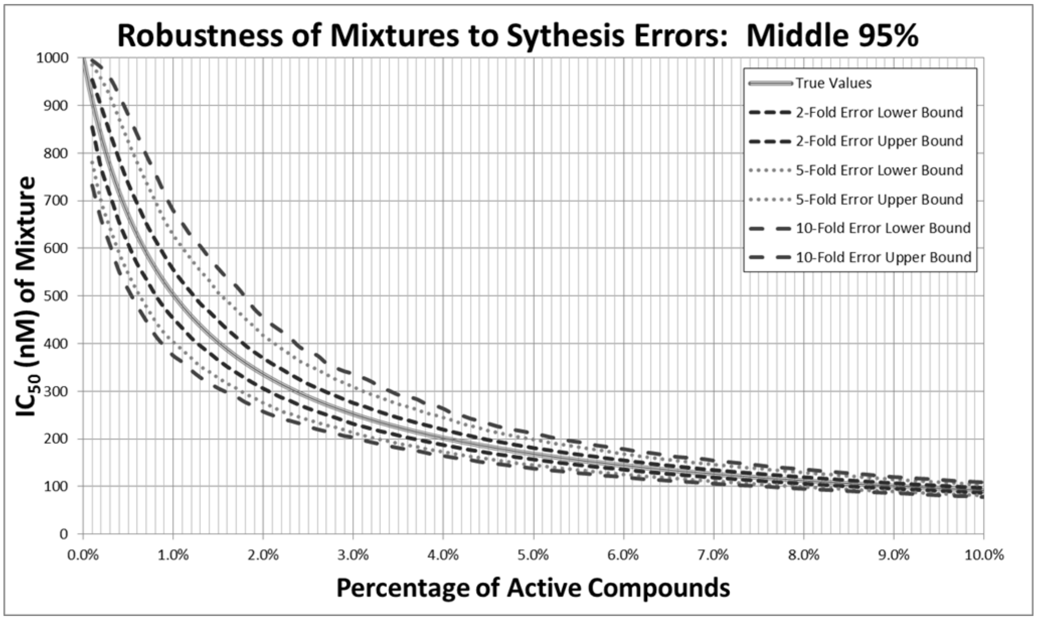


Figure 2.4 The harmonic mean model shows that combinatorial mixtures are highly robust to non-equimolarity in synthesis.

It is also possible to use the above arguments in a reversed fashion. If we continue with the above example in which an inactive compound has an IC$_{50}$ of 10,000 nM, and we assume the validity of the harmonic mean, then given a mixture activity it is possible to mathematically derive a range of activity percentages associated with different activity levels of individual compounds. Such relationships are plotted in Figure 2.3 for differing mixture activity values. A mixture with an IC$_{50}$ of 1000 nM, for example, may contain 1% of individual compounds with IC$_{50}$s of 10 nM, or 0.1% of individual compounds with IC$_{50}$s of 1 nM. The maximal IC$_{50}$ in order to guarantee detectability in this example is again 6600 nM, and so we can see such a mixture may contain 0.5% of individual compounds with IC$_{50}$s of 100 nM, 0.05% of individual compounds with IC$_{50}$s of 10 nM, or 0.005% of individual compounds with IC$_{50}$s of 1 nM. Conversely, a mixture that is indistinguishable from inactive because its IC$_{50}$ exceeds

6600 nM cannot even contain 0.5% of individual compounds with IC_{50} values of 100 nM, justifying the exclusion of such a mixture in further testing.

A final application of the harmonic mean model specifically impacts the chemical synthesis of combinatorial libraries. In particular, although it is true that such libraries have approximately equimolar concentrations of their constituents[69-70], it is also known that exact equimolarity is impossible to attain. The natural question, then, is exactly to what extent a deviation in equimolarity will affect the testing of a combinatorial mixture. To study this, we consider a hypothetical mixture containing 10 nM IC_{50} active compounds and 1,000 nM IC_{50} inactive compounds associated with a simple competitive binding assay. As above, the IC_{50} of such a mixture as a function of its percentage of active compounds may be calculated directly using the harmonic mean. To determine the effect of m-fold error in the equimolarity of compounds in this setting, for each percentage of active compounds a Monte Carlo simulation was performed , generating a uniform error in relative abundance of at most m-fold for each compound, for m = 2, 5, and 10. The upper and lower bounds of the middle 95% of simulations is plotted in Figure 2.4; as is evident, 95% of the time even a 10-fold maximal error in equimolarity does not result in even a two-fold error in mixture IC_{50}. This magnitude of error is far less than what is believed to actually be present[69-70].

The analyses presented above have significant implications on how mixtures can be used in drug discovery and, in part, explains the previous successes of research efforts where mixtures have been used, such as natural product extracts and systematically arranged mixtures[58-59, 69-79]. The active compounds isolated in various natural product studies[80-83] are often a very small percentage of the original material, and yet are still detectable. A clear distinction in the activity of mixtures containing highly active compounds and those that do not has been observed even in cases where the mixtures contained thousands of separate components. In the past, one of the reasons for this distinction has been posited as an abundance of similarly active compounds in a given mixture[84]. While this can a valid statement in systematically arranged mixtures, it is surely not necessarily true in natural products, and to a large degree the reason for this distinction can rather be attributed to the harmonic meaning of the individual components. Indeed, the reactions occurring in living organisms mirror qualitatively this behavior, with only trace amounts of specific substances playing vital biological roles[85].

2.2.3 Extension of the Harmonic Mean Model

Although simple competitive binding models the chemical behavior of molecules in many assays, these assays are binding assays and therefore far removed from ultimate mechanism of a pharmacological drug. Cell-based assays and in-vivo models both offer a clearer idea of the actual behavior of a molecule as a possible drug, but because of this the behavior to be mathematically modeled becomes necessarily more complex[65]. Some examples of potential complications to the simple competitive binding process include cooperative binding (in which the affinity of a ligand to the active site is altered by the binding of other ligands to other active sites), allosteric regulation (in which one or more ligands binding to a site different from the active site alter the affinity of the active site), and the existence of multiple disparate active targets.

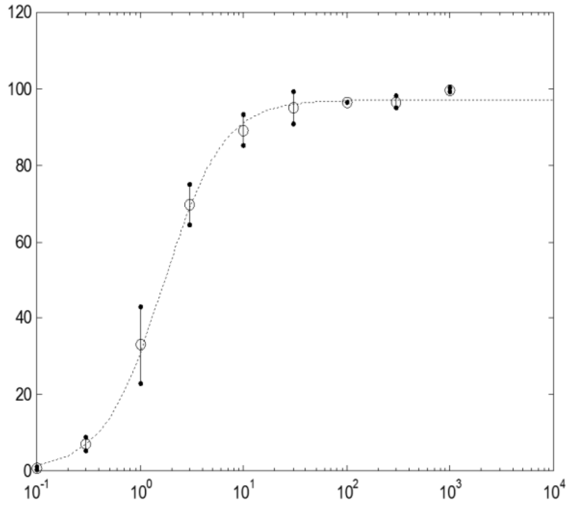


Figure 2.5 Dose response data fit using the DARS algorithm applied to a weighted nonlinear least-squares curve fit. Hundreds of such curve fits have been performed at TPIMS in the early stages of the drug-discovery process.

The dose-response behavior of homogeneous ligands participating in cooperative binding has been long established[86]. Hill's equation is the standard method for modeling cooperative binding scenarios:

$$\%_{binding} = \frac{x^n}{{IC_{50}}^n + x^n}$$

where $\%_{binding}$ is the proportion of binding sites to which the ligand has bound when tested at concentration x. The IC_{50} is, as usual, a constant associated with the ligand representing the concentration at which 50% binding occurs. For $n > 1$, the binging is said to be **positively cooperative**, and represents the scenario in which the binding of a ligand makes it easier for additional ligands to bind. Similarly for $n < 1$, binding is said to be **negatively cooperative** and occurs when the binding of a ligand makes other sites harder to bind to. Finally $n = 1$ represents simple competitive binding. The general procedure for determining the parameters of a dose response curve given experimental data is by weighted nonlinear least squares, an optimization problem to which DARS has been successfully applied numerous times. See Figure 2.5.

To extend the harmonic mean model to scenarios involving an arbitrary hill, we note that in Hill's equation above we have a method of quantifying the proportion of a particular molecule that will bind to its target. Thus we have the probability of that particular molecule binding, and if we assume that in a mixture the presence of other molecules does not affect the affinity of one another, this probability should remain a function of the concentration of that molecule alone, even in the presence of other molecules. Finally we note that in a mixture of *N* compounds, either precisely one compound binds to a particular target, or none of them do; two ligands cannot bind simultaneously to the same specific target. Thus we get the following expression for the probability of the ith compound binding:

$$P(i^{th} compound\ binding)$$

$$= \frac{\%_{binding,i} \cdot \prod_{j=1,j\neq i}^{N} \left(1 - \%_{binding,i}\right)}{\sum_{i=1}^{N} \%_{binding,i} \cdot \prod_{j=1,j\neq i}^{N} \left(1 - \%_{binding,i}\right) + \prod_{j=1}^{N} \left(1 - \%_{binding,i}\right)}$$

In particular, if $\{\beta_i\}_{i=1}^{N}$ represent the relative abundances of the *N* compounds in a mixture (so that $\sum_{i=1}^{N} \beta_i = 1$), then

$$P(i^{th} compound\ binding\ in\ mixture\ of\ concentration\ x\ )$$

$$= \frac{\%_{binding,i}(\beta_i x) \cdot \prod_{j=1,j\neq i}^{N} \left(1 - \%_{binding,j}(\beta_j x)\right)}{\sum_{i=1}^{N} \%_{binding,i}(\beta_i x) \cdot \prod_{j=1,j\neq i}^{N} \left(1 - \%_{binding,j}(\beta_j x)\right) + \prod_{j=1}^{N} \left(1 - \%_{binding,i}(\beta_i x)\right)}$$

and so

$$P(any\ compound\ binding\ in\ mixture\ of\ concentration\ x\ ) = \%_{binding,mixture}(x)$$

$$= \frac{\sum_{i=1}^{N} \%_{binding,i}(\beta_i x) \cdot \prod_{j=1,j\neq i}^{N} \left(1 - \%_{binding,j}(\beta_j x)\right)}{\sum_{i=1}^{N} \%_{binding,i}(\beta_i x) \cdot \prod_{j=1,j\neq i}^{N} \left(1 - \%_{binding,j}(\beta_j x)\right) + \prod_{j=1}^{N} \left(1 - \%_{binding,i}(\beta_i x)\right)}$$

$$= \frac{\sum_{i=1}^{N} \frac{\%_{binding,i}(\beta_i x)}{1 - \%_{binding,i}(\beta_i x)}}{1 + \sum_{i=1}^{N} \frac{\%_{binding,i}(\beta_i x)}{1 - \%_{binding,i}(\beta_i x)}}$$

Thus we have a general equation that predicts the functional form the dose-response curve of a mixture of N compounds given the dose-response curves of the mixture's constituents. For molecules modeled by Hill's equations, the above formulation simplifies to

$$\%_{binding,mixture}(x) = \frac{x^n}{x^n + \frac{1}{\left(\sum_{i=1}^{N} \left(\frac{\beta_i}{IC_{50,i}}\right)^n\right)^{1/n}}}$$

predicting that $IC_{50,mixture} = \frac{1}{\left(\sum_{i=1}^{N} \left(\frac{\beta_i}{IC_{50,i}}\right)^n\right)^{1/n}}$, which of course generalizes to the harmonic mean when $n = 1$. These results are also strikingly similar to the equations derived in [66] but without the cumbersome and arbitrary derivation. On the contrary, we see this result comes directly from first probabilistic principles and only assumes that (1) $\%_{response}(x)$ represents the probability of a ligand causing the measured response and (2) two ligands cannot coexist at the same identical site, or affect each other directly in any way. Therefore it is possible that this formulation may extend into other dose-response situations in which there cannot be interactions between compounds in a synergistic manner.

An additional, and heretofore unobserved, benefit for this formulation is that it allows direct modeling of antagonism – that is, the circumstance in which a ligand binds to, but does not activate, its target. In such a circumstance, the probabilities of binding will not change, but the system will behave as if only certain compounds have bound. Thus if we let $\{\alpha_i\}_{i=1}^{N}$denote whether a compound is an agonist ($\alpha_i = 1$) or antagonist ($\alpha_i = 0$), then the resulting mixture will have a dose response function

$$f_{mixture}(x) = \frac{\sum_{i=1}^{N} \frac{\alpha_i f_i(\beta_i x)}{1 - f_i(\beta_i x)}}{1 + \sum_{i=1}^{N} \frac{f_i(\beta_i x)}{1 - f_i(\beta_i x)}}$$

where $\{f_i\}_{i=1}^{N}$ are the dose response functions of the mixture's constituents and the above assumptions are met.

2.3 Determining the Integrity of Novel Mixture-Based Combinatorial Libraries

Combinatorial chemistry is a valuable tool in the rapid production of compound libraries for biological evaluation [59]. Specifically positional scanning synthetic combinatorial libraries (PS-SCL) enable the systematic screening of thousands to trillions of compounds through the use of only hundreds of samples[77]. A key component of PS-SCL involves the ability to produce near equimolar amounts of each individual compound in the mixture samples. Two methods for accomplishing equimolar mixtures are the use of physical mixtures, in which individually synthesized compounds are mixed together after synthesis, or chemical mixtures, in which compounds are synthesized together simultaneously. While the use of physical mixtures has the advantage of assuring near equimolar incorporation of each individual compound in the mixture samples, it is limited in that as the number of R groups increases around the scaffold the corresponding number of split and mixes required increases exponentially quickly, rendering the technique less valuable. The use of chemical mixtures where “isokinetic mixtures” are applied to produce equimolar mixtures does not have the same limitations of the physical mixture method; however, in order to utilize chemical mixtures in the production of PS-SCL one needs to determine the isokinetic ratios for the reagents used. Here, we introduce a novel method for the comparison of LCMS UV spectra which allows direct relative evaluation of the equimolarity of the compounds contained in that mixture; further, this method is applicable in cases more complex than would be feasible by simply using a reference compound. We apply it to the reaction between the amine termini of a resin bound peptide fragment and a sulfonyl chloride in order to produce a sulfonamide. The isokinetic ratio was determined and the effect of the on resin peptide fragment was evaluated.

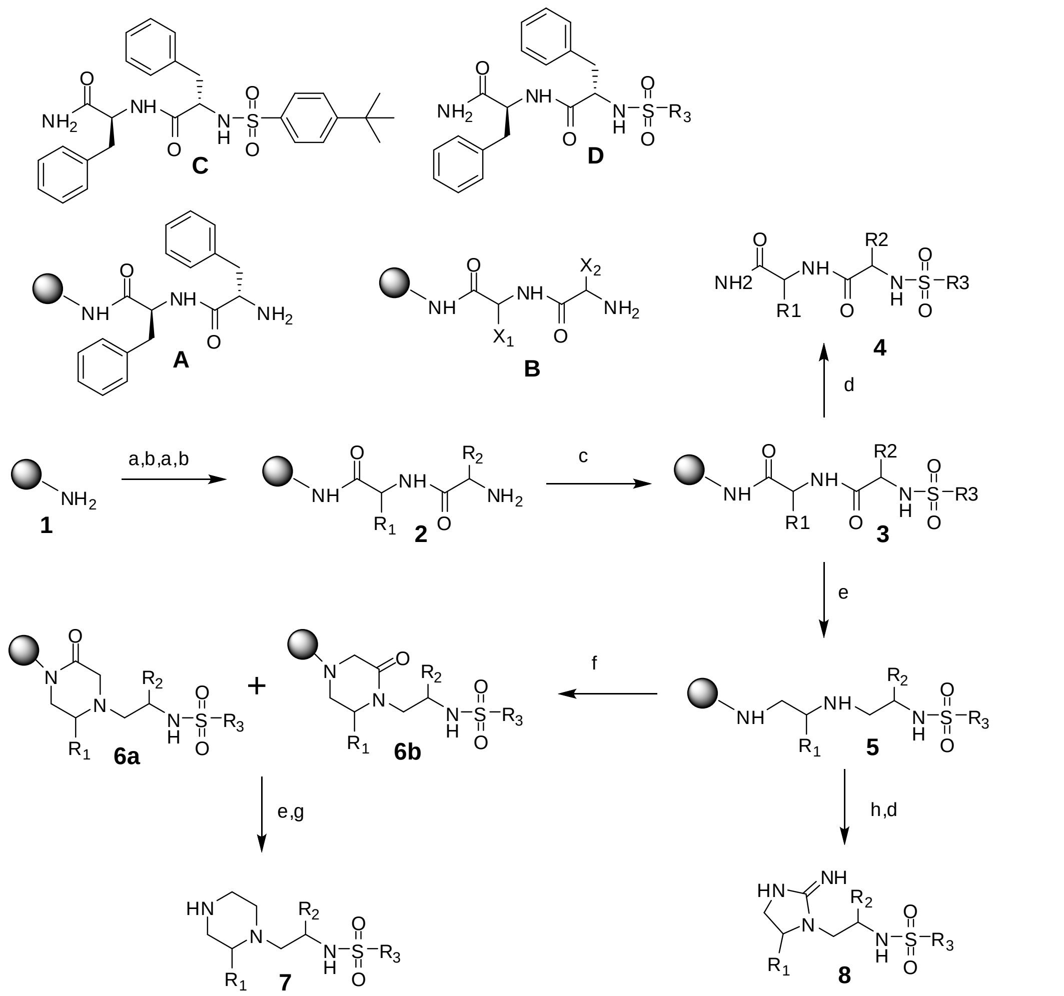
2.3.1 Experimental Synthesis Procedure

The binding ratios obtained for the sulfonyl chlorides have been applied to synthesize two heterocyclic mixture libraries. A linear sulfonamide precursor is synthesized using the “tea bag” method[87]. Two different cyclizations are applied to form either a 2-piperzine sulfonamide library or a 2-guandine sulfonamide library (See Figure 2.6, Structures 7 and 8). In order to determine which optimal amino acids and sulfonyl chlorides will be incorporated into the library, individual compounds with varying functionalities are first synthesized. L-amino acids with different properties such as basic, acidic, aromatic rings (etc.) are analyzed in the R1 and R2 position. 20 Sulfonyl chlorides with varying functionalities are used in the R3 position. From these results 34 L, D and unusual amino acids were selected for the libraries. The sulfonyl chlorides were narrowed down to 15 which yielded acceptable product purity. Each library therefore contains 17,340 compounds (34 x 34 x 15). The 15 sulfonyl chlorides chosen were then used for the ratio binding experiments described below. It may be noted that compounds in the R3 position which contained fluorine substitution yielded low product purity. This may be due to fluorine binding to piperzine during the reduction step (Figure 2.6, Step E2).

Boc-L-Phenylalanine [6eq, 0.1M dimethylformamide (DMF)] was coupled to MBHA resin using DIC(6eq) and HOBT (6eq) for 2 hours at room temperature, followed by washes DMF (2X) and dichloromethane, DCM (2 X) (Scheme 2.1). The Boc was removed using 55% trifluoroacetic acid in DCM for 30 minutes, followed by washes with DCM (2X), IPA (2X), and DCM (3X). Then the resin was neutralized using 5% DIEA in DCM (3X) followed by washes of DCM (3X). This was repeated so that a sequence of L-Phenylalanine, L-Phenylalanine – MBHA resin was obtained.   Sulfonyl chloride (8eq, 0.1M anhydrous DCM) was coupled to the F-F-MBHA resin using DIEA (8eq) overnight. Each coupling reaction was monitored by the use of the ninhydrin test to ensure the coupling was complete. The compounds were cleaved using HF at 0°C for 1.5 hours and extracted with 95% acetic acid and water and then lyophilized.

2.3.2 Determination of Proper Reagent Ratio

In order to synthesize an equimolar mixture of sulfonyl chlorides bound to amino acids, ratios must be determined in order to adjust for the fact that steric and electronic hindrance will vary amongst the different sulfonyl chlorides used in the mixture. The first step in doing so is to determine a correction factor for each compound to account for differences in molar absorbance.



Scheme 1) a) Boc-AA/DIC/HOBt (6ex) in DMF b) 1) 55%TFA/DCM, 2) 5%DIEA/DCM c) Sulfonyl chloride/DIEA (8ex) in DMF d) HF 0°C1.5hr e) 1) Borane-THF 60°C 96hrs 2) piperidine 60°C 24hrs f) Bromoacetic acid/DIC(5ex)/DIEA (2.5ex) g) HF 0°C 7hr h) cyanogen bromide/DMF (5ex). A) mBHA resin-(L-Phenylalanine)-(L-Phenylalanine). B) mBHA resin-(equal molar mixture of amino acids; Ls, Ds, and unusuals)-(equal molar mixture of amino acids; Ls, Ds, and unusuals). C) Compound derived from using L-Phenylalanine (R1), L-Phenylalanine (R2), and 4-tert-butylbenzene-1-sulfonyl chloride (R3). D) A mixture sample defined with L-Phenylalanine (R1), L-Phenylalanine (R2), and a mixture of 15 different sulfonly chlorides at R3.

Figure 2.6 A schematic of the synthesis process.

Individual compounds with the sequence L-Phenylalanine in the R1 and R2 position were synthesized. All the sulfonyl chlorides of interest were coupled to this sequence separately (See Figure 2.6, compound 4). The sequence of L-Phenylalanine in the R1 and R2 position was chosen because of its strong UV absorbance properties at 214nm. All individual compounds were analyzed using reverse phase HPLC-MS. See

Table 1. A compound, obtained from using 4-tert-butylbenzene-1-sulfonyl chloride at the R3 position (Compound C, Figure 2.6), with a sufficiently distinct retention time was selected as the control to which all other compounds were compared. Each compound was then physically mixed with the control to create an equal molar mixture. These samples were analyzed using HPLC-MS. The integration parameters were set to allow the best integration possible for the UV peaks 214nm, and all data sets were analyzed using these same integration parameters. Since the concentration of each compound is known to be equal, any differences in peak area can be attributed to differences in molar absorbance. A correction factor is therefore calculated for each compound as the ratio of the control compound's peak area to the peak area of that compound:

$$CF_n = \frac{Area_{control}}{Area_n}$$

This definition insures that by multiplying the peak area of the n^{th} compound by its corresponding control factor, an area equal to peak area of the control is obtained. This process was performed in duplicate to confirm the correction factors and to minimize experimental error. See Table 2.3.

The rate at which each sulfonyl chloride binds will vary based on steric and electronic hindrance. In order to synthesize an equimolar mixture a ratio of each component of the sulfonyl chloride mixture must be calculated in order to compensate for binding differences. Each compound was synthesized using a 1:1 mixture of that compound with the control, each at a 4-fold excess over the resin (for an 8-fold excess total). The resulting synthetic mixtures were then analyzed by HPLC-MS using the same method and integration parameters as were used to calculate the correction factor.

Sulfonyl Chloride	**Molecular Weight**	**Retention Time**	**UV Absorptivity Correction Factor Relative to Control**	**Binding Ratio Relative to Control**	**Final Reagent Ratio Relative to Control**
thiophene-2-sulfonyl chloride	458.10	17.00	1.30	4.08	3.15
benzenesulfonyl chloride	451.90	17.23	0.89	0.60	0.67
4-methoxybenzene-1-sulfonyl chloride	481.95	17.31	1.06	1.11	1.04
4-methylbenzene-1-sulfonyl chloride	465.95	17.89	0.96	0.70	0.73
2-chlorobenzene-1-sulfonyl chloride	485.80	17.91	0.88	0.77	0.87
3-methylbenzene-1-sulfonyl chloride	466.20	17.96	0.78	0.63	0.81
2-bromobenzene-1-sulfonyl chloride	532.00	18.09	0.87	0.69	0.79
4-chlorobenzene-1-sulfonyl chloride	486.10	18.38	0.97	0.49	0.51
4-bromobenzene-1-sulfonyl chloride	531.90	18.58	1.03	0.51	0.50
naphthalence-2-sulfonyl chloride	502.00	18.66	0.51	0.26	0.52
4-iodobenzene-1-sulfonyl chloride	578.00	18.83	1.10	0.48	0.44
4-(trifluoromethyl)benzene-1-sulfonyl chloride	519.90	18.91	1.02	0.41	0.40
Biphenyl-4-sulfonyl chloride	528.15	19.46	0.84	0.72	0.85
2,4,6-trimethylbenzene-1-sulfonyl chloride	494.05	19.50	0.81	0.37	0.45
4-tert-butylbenzene-1-sulfonyl chloride **(control)**	508.15	20.01	1.00	1.00	1.00

Table 2.3 Calculated values for the various synthesized compounds with L-Phenylalanine in the first two positions and a sulfonyl chloride in the third position

The UV 214nm peak area under the curve is used to calculate the binding ratio:

$$BR_n = \frac{Area_{control}}{CF_n Area_n}$$

The binding ratios obtained are thus the rate at which the n^{th} sulfonyl chloride binds relative to the control compound, with the differences in absorptivity accounted for via multiplication by the control factor. See again Table 2.3.

2.3.3 Differences in Reaction Rate Caused by Multiple Simultaneous Reactions

The above calculations were done as individual comparisons between a given compound and the control. To determine if the presence of other sulfonyl chlorides simultaneously would change the reaction rate, these ratios were then tested in the presence of multiple sulfonyl chlorides including the control. A mixture of all the compounds would not be able to distinguish individual peaks due to some compounds having similar retention times and/or same molecular weights, and therefore individual changes in reaction rates would not be determinable. Because of this, multiple sample sets were set up to maximize the amount of compounds present still distinguishable by

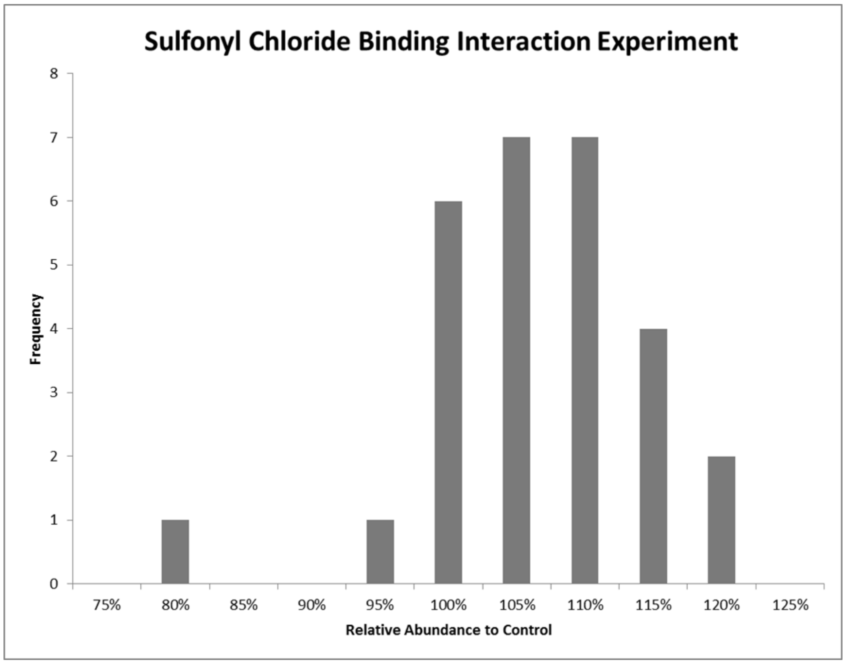


Figure 2.7 Demonstration that the correct binding ratios have been achieved for this experiment; no compound deviates by more than 20% of the control.

HPLC-MS analysis. The control compound was present in every sample set. Each designed mixture of sulfonyl chlorides was coupled to L-Phenylalanine, L-Phenylalanine – MBHA resin (Compound A, Figure 2.6) using an 8-fold excess (meaning the total excess of the sulfonyl chlorides) over the resin. The experiment was done so that every compound was analyzed in duplicate. The HPLC-MS 214nm UV peaks were analyzed using the same integration parameters that were used to obtain the ratios in the previous experiments. The correction factor for each compound was used to compare the amount of that compound to the amount of the control compound. No compound deviates in amount by more than 20% from that of the control (See Figure 2.7). This is well within experimental error and shows that there is no substantial effect on the reaction rates of the individual sulfonyl chlorides from the presence of other sulfonyl chlorides.

2.3.4 Differences in Reaction Rate Caused by Different On-Resin Amino Acids

It has been demonstrated that differences in the on-resin amino acid onto which the reagents bind will cause differences in the reaction rates of the individual reagents [59]. To explore the extent to which this is the case, we have developed a novel technique to quantify the overall difference between two mixtures containing multiple compounds. In general, direct comparison between two different HPLC-MS 214nm UV spectra is complicated by differences in concentration and attenuation from run to run. Further, use of a reference compound cannot quantify differences in peak shape that may result when multiple compounds with similar retention times are present, and can only be used to adjust relative peak heights. However, working under the assumption that two identical samples will have geometrically similar 214nm UV spectra, it is posited that through appropriate shifting and scaling two 214nm UV spectra may be compared directly for similarity. In particular, given two samples, first the HPLC-MS was used to determine the initial retention time of the first peak of interest, t_i, and the final retention time of the final peak of interest, t_f, of both samples. Then, the baseline was assumed to be linear and was removed from both samples via the equation:

$$UV_{without\ baseline}(t) = UV(t) - \frac{UV(t_f) - UV(t_i)}{t_f - t_i}(t - t_i) - UV(t_i)$$

It was assumed that for two identical samples there exist parameters *A*, *b*, *h*, and *k* such that

$$UV_{without\ baseline}^{Sample\ 1}(t) = A \cdot UV_{without\ baseline}^{Sample\ 2}\big(b \cdot (t - h)\big) + k$$

In other words, it was assumed that there exists appropriate vertical and horizontal scaling and shifting parameters so that the UV spectra of identical samples can be made

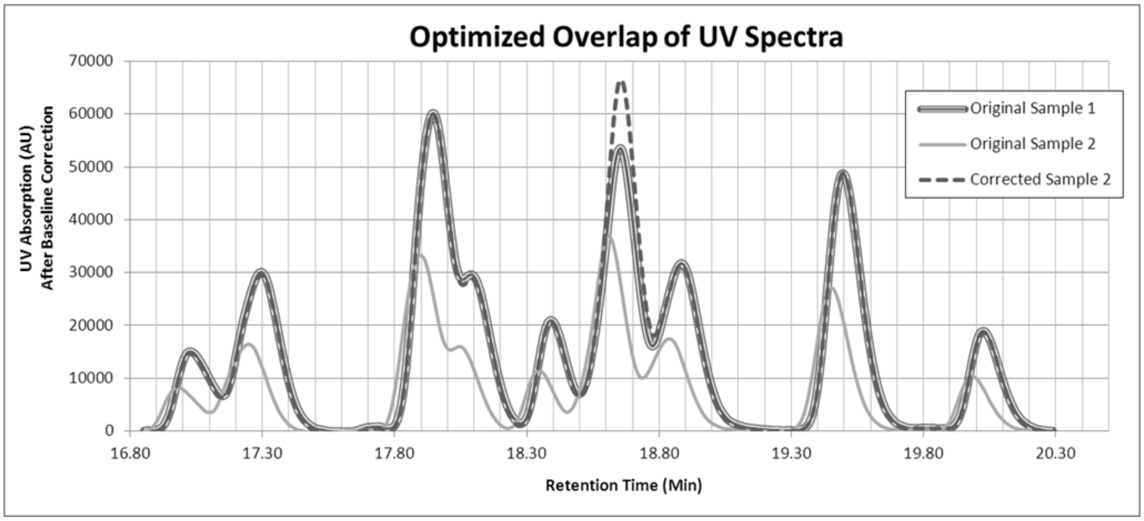


Figure 2.8 An example of two UV spectra which are not directly comparable because of differences in attenuation and concentration, and the result of finding the optimal overlap between them using the DARS algorithm. The difference in the spectra is now apparent and quantifiable.

to "overlap" one another. Furthermore, for two samples that are not identical it may be thought to have the relationship

$$UV_{without\ baseline}^{Sample\ 1}(t) = A \cdot UV_{without\ baseline}^{Sample\ 2}\big(b \cdot (t-h)\big) + k + D(t),$$

where integration of the difference term $D(t)$ corresponds to the area between the two spectra after overlapping has occurred, and therefore quantifies differences in the relative makeups of the two samples. To this end, given two 214nm UV spectra, DARS can be used to find the optimal overlap parameters *A*, *b*, *h*, and *k* which minimize the area between the two spectra; this minimum area serves as a measure of the "difference" between the two samples (See Figure 2.8).

In order to test the resolution of the above method a synthetic mixture of 15 compounds (termed "stock solution" from here forward) was made using L-

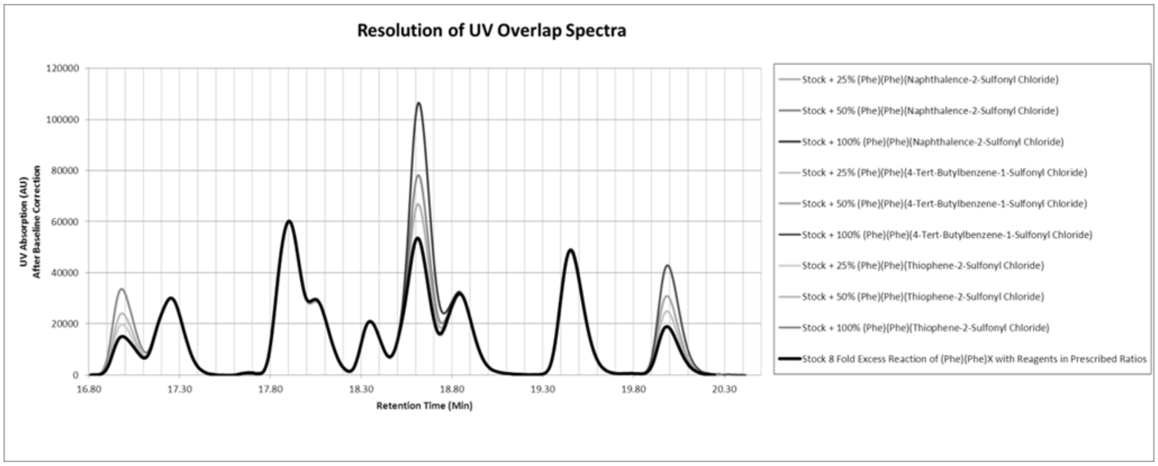


Figure 2.9 214nm UV spectra of stock solution spiked with specific compounds, as compared to the stock solution alone, after optimal overlap correction.

Test Sample	Comparison to Stock Solution
Stock Solution	0.61%
Stock Solution + 25% L-Phenylalanine, L-Phenylalanine, Thiophene-2-sulfonyl chloride	2.22%
Stock Solution + 50% L-Phenylalanine, L-Phenylalanine, Thiophene-2-sulfonyl chloride	3.48%
Stock Solution + 100% L-Phenylalanine, L-Phenylalanine, Thiophene-2-sulfonyl chloride	5.97%
Stock Solution + 25% L-Phenylalanine, L-Phenylalanine, Naphthalence-2-sulfonyl chloride	5.09%
Stock Solution + 50% L-Phenylalanine, L-Phenylalanine, Naphthalence-2-sulfonyl chloride	8.24%
Stock Solution + 100% L-Phenylalanine, L-Phenylalanine, Naphthalence-2-sulfonyl chloride	16.79%
Stock Solution + 25% L-Phenylalanine, L-Phenylalanine, 4-tert-butylbenzene-1-sulfonyl chloride	2.77%
Stock Solution + 50% L-Phenylalanine, L-Phenylalanine, 4-tert-butylbenzene-1-sulfonyl chloride	4.62%
Stock Solution + 100% L-Phenylalanine, L-Phenylalanine, 4-tert-butylbenzene-1-sulfonyl chloride	8.22%

Table 2.4 Percentage error between 214nm UV spectra of stock solution spiked with specific compounds, as compared to the stock solution alone.

Phenylalanine (R1), L-Phenylalanine (R2) and X3 (where X3 represents a mixture of the 15 sulfonyl chlorides of interest using the prescribed ratios calculated as in Table 2.3). Multiple aliquots of this stock solution were made and spiked with one of three different individual compounds at one of three different percentage increases, 25%, 50%, and 100%. The samples were then analyzed using the HPLC-MS 214nm UV spectra, with each sample compared to the un-spiked stock solution. Optimal overlapping was performed as described above, and the area between the resulting curves was calculated (See Figures 2.9 and Table 2.4). The spectra after overlapping clearly show differences

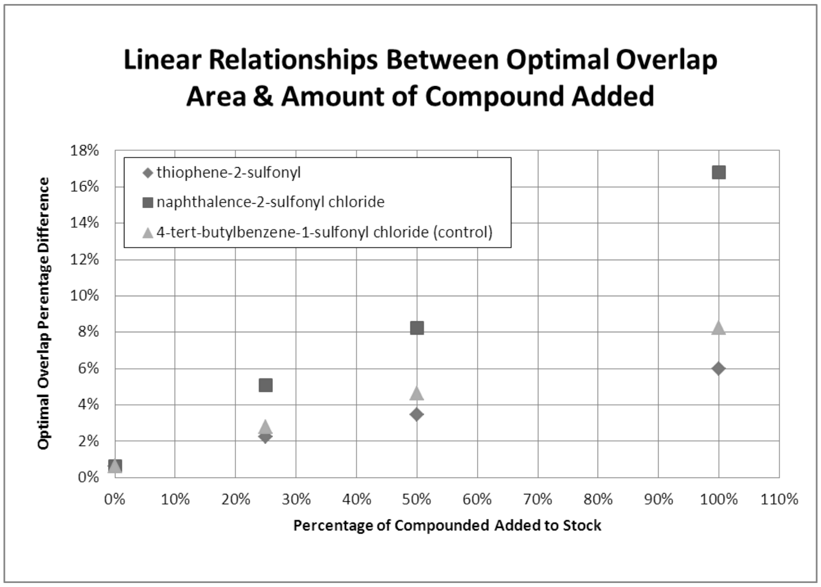


Figure 2.10 Errors are between 214nm UV spectra of stock solution spiked with specific compounds, as compared to the stock solution alone, after optimal overlap correction. The relationship between the percentage errors and amount of compounded added is clearly linear, with all R^2 values exceeding 0.99. Steeper slopes correspond to higher UV absorptivity.

even when only 25% additional compound is added, and the area difference between the curves scales linearly with the amount of compound added (See Figures 2.10). Unsurprisingly, the compound with the lowest correction factor (and therefore highest UV absorptivity) also had the greatest slope; the compound with the highest correction factor (and therefore lowest UV absorptivity) had the lowest slope.

With the ability to compare UV spectra directly, the following experiment was devised to determine the impact of differing on-resin amino acids to the binding rates of the sulfonyl chlorides. Two different on resin starting materials were prepared; one with L-Phenylalanine used in the R1 and R2 positions (Compound A, Figure 2.6) and one with

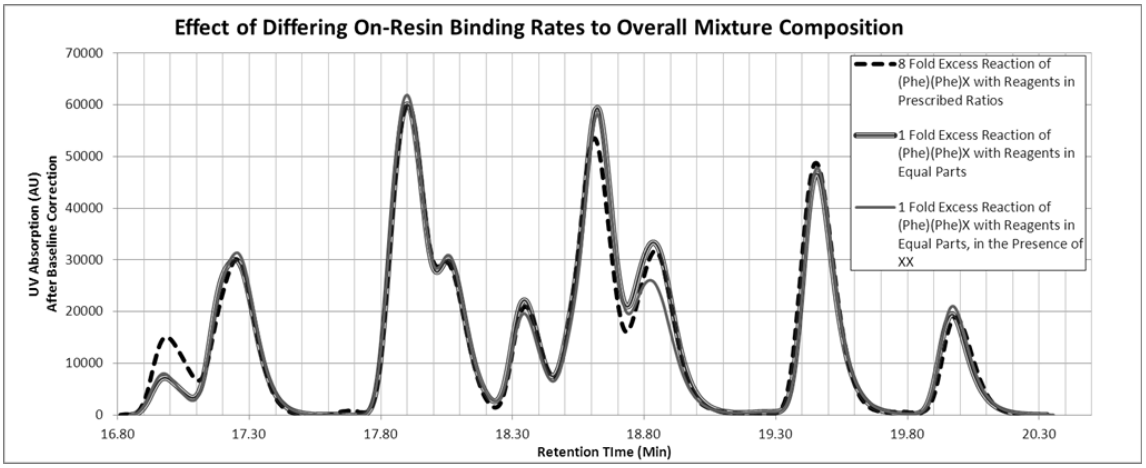


Figure 2.11 214nm UV spectra after optimal overlap correction of 8-fold stock solution and 1-fold reactions. The differences between the samples are not substantial.

an equal molar mixture of 66 amino acids in the R1 and R2 positions (Compound B, Figure 2.6). The two reaction vessels were set up. In the first reaction vessel two tea bags were added, one containing Compound A and the other containing Compound B. In the second reaction vessel two tea bags both containing Compound A were added. To each vessel an equal molar mixture of the 15 sulfonyl chlorides was added. It was added so as to provide only a 1 fold equivalent amount of sulfonyl chloride to available free amine (note this differs from the 8 fold excess used above). The assumption is that in the presence of Compound B, if Compound B has an average binding rate to any of the sulfonyl chlorides that differs drastically from that of Compound A, it will disrupt the equimolarity of the reagents reacting with Compound A and cause an unequal distribution of products. The final mixture products (of the form of Compound D in Figure 2.6) from Compound A reacted in the presence of Compound B and in the absence of Compound B were analyzed using HPLC-MS 214nm UV, and their spectra were compared both to each other and to the aforementioned stock solution (done using the corrected ratios with an eight fold excess). See Figure 2.11.

The largest difference between spectra for the stock solution (using the 8 fold excess) and the spectra from the mixture products derived from Compound A using the 1 fold excess in the presence and absence of Compound B occurs at an approximate retention time of 17.0 minutes. This corresponds to L-Phenylalanine (R1), L-Phenylalanine (R2), Thiophene-2-sulfonyl chloride (R3); this difference is unsurprising since L-Phenylalanine, L-Phenylalanine, Thiophene-2-sulfonyl chloride has such a slow reaction rate (Table 2.3). Most importantly, the spectra for both the one fold excess Compound D samples do not differ substantially from one another, with an overall difference of 7.01%. There are slight differences between the spectra at all retention times, but the largest difference occurs at an approximate retention time of 18.85 minutes. This peak corresponds to L-Phenylalanine (R1), L-Phenylalanine (R2), 4-iodobenzene-1-sulfonyl chloride (R3) and L-Phenylalanine (R1), L-Phenylalanine (R1), 4-(trifluoromethyl)benzene-1-sulfonyl chloride (R3), both of which have similar correction factors to L-Phenylalanine (R1), L-Phenylalanine (R2), 4-tert-butylbenzene-1-sulfonyl chloride (R3) (See again Table 2.3). In the experiment described above, a mixture which contained two-fold the amount of L-Phenylalanine, L-Phenylalanine, 4-tert-butylbenzene-1-sulfonyl chloride corresponded to a difference in spectra of 8.22%, we posit that no single compound molarity in the mixture obtained from reacting Compound A in the presence of Compound B differs more than two-fold from that of the mixture obtained from reacting Compound A in the absence of Compound B. In fact, since the 7.01% error is distributed amongst multiple peaks, and multiple compounds are present in certain peaks, it is likely that the error in equimolarity is far less than two-fold. This means that the average on resin functionality has only a marginal effect on the reaction rates for this specific reaction. Further, as seen in Section 2.2, an error in equimolarity of less than 2 fold should have minimal impact on the viability of this library for biological testing.

2.4 Conclusion

In this chapter, we have demonstrated how mathematical tools of varying complexity, including the DARS algorithm developed in Chapter 1, can be used in the fields of medicinal chemistry and drug discovery. In particular, the study of the synthesis process and the biological assay results associated with the mixture-based combinatorial libraries the TPIMS specializes in is a fruitful source of constructive collaborative efforts. The harmonic mean model and its subsequent extension offer important insights into the strengths and limitations of a mixture-based approach; such modeling efforts have only begun to study the ever-increasing hierarchy of complex biological assays. The method of overlapping LCMS-UV peaks to determine errors in relative abundance will, at the same time, aide in the development of new mixture-based combinatorial libraries that will broaden the expanse of the search for new lead compounds. By thus enhancing both the breadth and scope of the usage of mixture-based combinatorial libraries at TPIMS, applied mathematical methods have helped progress medicinal pharmacology further towards the drugs of tomorrow.

CHAPTER 3

STATISTICAL AND PROBABILISTIC METHODS IN COMPUTATIONAL CHEMISTRY

3.1 Introduction

Computational chemistry, broadly defined, is the study of the quantitative information associated with chemical compounds. This includes analyses related to their structural and chemical properties and their interactions with other molecular targets. The purpose of these studies are both to develop a larger understanding of the ways in which successful active compounds behave, and also to use this information to further investigate for novel lead compounds.

3.1.1 The Structural-Activity Relationship

The **structural-activity relationship** (SAR) of a set of compounds is the information associated with how structural differences between two compounds effects the relative activities of those compounds. In order to explore this area, it is useful to have a method for determining how similar two compounds are; in fact, many such methods have been developed[117-118]. We will discuss some methods in more detail herein, specifically the methods of defining similarity based on molecular properties and activity, but other universally established methods will be used without further detail.

The **chemical space** of a set of compounds is set of all property and structural information about those compounds. The SAR of a data set can be conceptualized as an

activity landscape in which biological activity is added a dependent variable to the chemical space[104]. The activity landscape is referred to as a **continuous** SAR where (similar to the mathematical definition) small changes in molecular structure are associated with small changes in activity[102]. A **discontinuous** SAR or **rugged activity landscape**, however, is populated with molecules with small changes in structure but large changes in activity (such discontinuities are referred to as **activity cliffs**)[102]. Such landscapes are common in lead optimization. It can also happen that structurally diverse compounds have similar activity, which is called **scaffold hopping**[105] and may suggest different binding modes or sites, or may reveal the effect of additional mechanisms such as the interaction with membranes that are not typically considered in several modeling approaches[103]. Understanding the activity landscape of a data set is, however, a difficult task because the overall landscape may be highly complex, involving a combination of smooth and rugged regions[101]. To further complicate matters, the dependence of chemical space with molecular representation and similarity method used[106-107] can cause the same set of compounds to have drastically different activity landscapes, similar to the way a function can be discontinuous under some metrics but not others. Mathematical and statistical analyses of the enormous amount of information that is available for even a small set of compounds is vital both for describing the activity landscape in the most useful and compelling way, and also for finding the important information in the SAR for further utility.

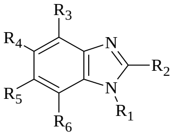
Quantitative characterization of the structural-activity relationships of small molecules plays a key role in lead optimization[51-52, 54, 113]. To this end, a number of methods can be employed (including quantitative structure-activity relationships

(QSAR), rule-based methods, neural networks or pharmacophore modeling, to name few examples[95-98]). However, several methodologies like conventional QSAR make assumptions that are not necessarily valid and, thus, may present misleading results including non-predictive models[99]. For example, one common assumption is that a lead series of compounds has a common binding mode or mechanism of action[100-101], which may not be the case. For this reason, understanding the activity landscape and early detection of activity cliffs[102] can be crucial to the success of computational models[103].

Different approaches are emerging to characterize systematically the activity landscape and are reviewed in Bajorath et al[104]. These include the Structure-Activity Relationship Index (SARI)[108], Structure-Landscape Index (SALI)[103,109], Structure-Activity Similarity (SAS) maps[110], and network-like similarity graphs (NSG)[111] SARI and NSG have been used to detect molecules with small changes in structure but large changes in selectivity (**selectivity cliffs**)[112]. Our group proposed using multiple structural representations to derive a consensus model for the activity landscape and identify consensus activity cliffs[113]. Recently, 3D representations of the activity landscape were proposed, confirming the existence of consensus activity cliffs and representation-dependent cliffs[114].

3.1.2 The Biological Target for Study

Parasitic diseases are still a major health problem in developing countries. Mucosal infections by protozoa infect more than a billon of people every year[88]. Among the most common protozoa infections are giardiosis, caused by *G. intestinalis*,



	R_1	R_2	R_3	R_4	R_5	R_6	pIC_{50} *T. vaginalis*	pIC_{50} *G. intestinalis*	ΔpIC_{50}	Ref.
1	H	$\mathbf{CF_3}$	H	H	H	H	5.50	6.97	1.47	1,2
2	CH_3	$\mathbf{CF_3}$	H	CF_3	H	H	5.39	5.94	0.55	1
3	CH_3	$\mathbf{CF_3}$	H	H	CF_3	H	5.27	5.05	-0.22	1
4	CH_3	$\mathbf{CF_3}$	H	Propylthio	H	H	6.70	4.98	-1.72	3
5	CH_3	$\mathbf{CF_3}$	H	H	Propylthio	H	5.59	5.85	0.26	3
6	CH_3	$\mathbf{CF_3}$	H	Benzoyl	H	H	4.53	5.96	1.43	3
7	CH_3	$\mathbf{CF_3}$	H	H	Benzoyl	H	4.97	5.89	0.92	3
8	H	CF_3	H	Br	Br	H	6.66	6.92	0.26	4
9	H	CF_3	Br	Br	Br	Br	8.70	7.25	-1.45	4
10	H	C_2F_5	H	Cl	Cl	H	6.52	6.25	-0.27	4
11	H	CF_3	H	NO_2	NO_2	H	6.24	6.62	0.38	4
12	H	C_2F_5	Br	Br	Br	Br	5.00	7.64	2.64	4
13	CH_3	$CONH_2$	H	H	Cl	H	6.96	7.12	0.16	5
14	CH_3	$CONHCH_3$	H	H	Cl	H	6.98	7.15	0.17	5
15	CH_3	$CON(CH_3)_2$	H	H	Cl	H	6.63	7.40	0.77	5
16	CH_3	$COOCH_2CH_3$	H	H	Cl	H	7.72	7.32	-0.4	5
17	CH_3	$CONH_2$	H	Cl	H	H	6.73	6.63	-0.1	5
18	CH_3	$CONHCH_3$	H	Cl	H	H	6.45	6.45	0	5
19	CH_3	$CON(CH_3)_2$	H	Cl	H	H	6.68	6.61	-0.07	5
20	CH_3	$COOCH_2CH_3$	H	Cl	H	H	7.57	7.40	-0.17	5
21	CH_3	$CONH_2$	H	Cl	Cl	H	6.87	6.34	-0.53	5
22	CH_3	$CONHCH_3$	H	Cl	Cl	H	6.65	6.82	0.17	5
23	CH_3	$CON(CH_3)_2$	H	Cl	Cl	H	7.12	7.13	0.01	5
24	CH_3	$COOCH_2CH_3$	H	Cl	Cl	H	7.53	7.56	0.03	5
25	CH_3	$CONH_2$	H	H	H	H	6.78	7.03	0.25	5
26	CH_3	$CONHCH_3$	H	H	H	H	6.98	7.22	0.24	5
27	CH_3	$CON(CH_3)_2$	H	H	H	H	6.37	6.29	-0.08	5
28	CH_3	$COOCH_2CH_3$	H	H	H	H	7.07	7.16	0.09	5
29	CH_3	$COCH_3$	H	H	H	H	6.68	7.06	0.38	5
30	CH_3	$COCH_3$	H	Cl	H	H	6.88	7.30	0.42	5
31	CH_3	$COCH_3$	H	H	Cl	H	6.64	7.17	0.53	5
32	CH_3	$COCH_3$	H	Cl	Cl	H	7.20	7.46	0.26	5

Table 3.1 Chemical structures of benzimidazoles and biological activity against T. Vaginalis and G. intestinalis

and trichomonosis, a genitourinary infection caused by *T. vaginalis*[89-90]. As part of on-going efforts to develop compounds as giardicidal and trichomonicidal agents several benzimidazole derivatives have been synthesized and tested finding compounds in the low nanomolar range[91-94]. However, systematic and quantitative studies of the structure-activity relationships (SAR) of benzimidazoles as antiprotozoal agents are still limited. Herein we conducted a comprehensive analysis of the activity landscape of 32 benzimidazoles mostly synthesized and tested in our group against *T. vaginalis* and *G. intestinalis* (See Table 3.1). Compounds are non 2-methylcarbamates and their mechanism of action remains unknown[115-116]. The analysis was based on pairwise comparisons of the activity similarity and molecular similarity. For each parasite, pairwise SAR was visually depicted using SAS maps. Quantitative analyses of the SAS maps were the basis to identify consensus activity cliffs and develop consensus models of the activity landscape. We also compared the SAR of the benzimidazoles between the two parasites.

3.2 The Standard Methodology

3.2.1 Description of Data Set

The chemical structure of 32 previously reported benzimidazoles[91-94] is presented in Table 1 along with the biological activity against *T. vaginalis* and *G. intestinalis*. Table 3.1 lists the 50% inhibitory concentration (IC_{50}) in in vitro susceptibility assays for each parasite as pIC50 = –$\log_{10}(IC_{50})$. For *T. vaginalis* the activity ranges from 2 nM (pIC50 = 8.7) to 29,512 nM (pIC50 = 4.53); median pIC50 = 6.7. For *G. intestinalis* the activity ranges from 22.9 nM (pIC50 = 7.64) to 10,471 nM (pIC50 = 4.98); median pIC50 = 7.0. The activity of the 32 compounds was obtained by the same group under similar conditions.

3.2.2 2D and 3D Structural Similarity

For each pair of molecules m_i and m_j, pairwise **structural similarities** (SS_{ij}) were computed using the Tanimoto coefficient[117-118] with the following 2D molecular representations as implemented in Molecular Operating Environment (MOE):32 MACCS keys (166 bits), pharmacophore graph triangle (i.e., graph-based three point pharmacophores) (GpiDAPH3), typed graph distance (TGD), and typed graph triangle (TGT). We also used the following 2D (32-bit) fingerprints as implemented in Canvas:33 radial (also known as extended connectivity fingerprints), dendritic, pairwise and MOLPRINT 2D. To compute 3D similarities, a single low-energy conformation was considered for each molecule obtained with geometry optimization using the PM3 semiempirical method. 3D similarity values were calculated with the MOE pharmacophore atom triangle (piDAPH3) and pharmacophore atom quadruplet (piDAPH4) fingerprints, and Canvas 3 and 4 points pharmacophores. Despite the inherent conformational issues, the use of 3D structural representations were valuable to characterize the activity landscapes.

3.2.3 Property Similarities

The following properties were computed with Canvas: molecular weight (MW), number of rotatable bonds (RB), hydrogen bond acceptors (HBA), hydrogen bond donors (HBD), polar surface area (PSA), and the octanol/water partition coefficient (AlogP). Properties were first auto-scaled with mean centering using the equation:

$$p_{ki} = \frac{P_{ki} - \overline{P_k}}{\sigma_{P_k}}$$

where p_{ki} denotes the scaled version of the k^{th} property for the i^{th} molecule, P_{ki} denoted the unscaled value, and $\overline{P_k}$ and σ_{P_k} denote, respectively, the mean and standard deviation of the *k*th property over all molecules in the study.

The standard Euclidean distance between a pair of molecules' sets of scaled structures was then computed with the expression[118]:

$$d_{ij} = \left[\sum_{k=1}^{K} (p_{ki} - p_{kj})^2 \right]^{1/2}$$

where d_{ij} denotes the Euclidean distance between the i^{th} and j^{th} molecules, and p_{ki} , and p_{kj} denote the value of the scaled property k of the i^{th} and j^{th} molecules, respectively. In this study, $K = 6$.

Euclidean distances were scaled from 0 to 1 as follows:

$$sd_{ij} = \frac{d_{ij} - \min d_{ij}}{\max d_{ij} - \min d_{ij}}$$

where sd_{ij} is the scaled distance. Pairwise property similarities were measured with the expression:

$$PS_{ij} = 1 - sd_{ij}$$

where PS_{ij} is the **property similarity** of the i^{th} and j^{th} molecules, and sd_{ij} is the scaled distance.

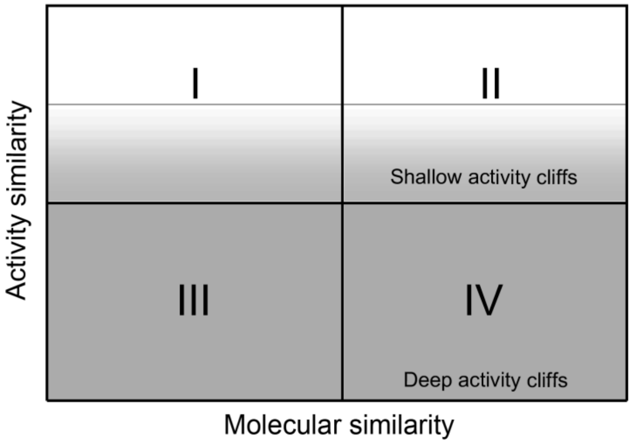


Figure 3.1 General form of the structure-activity similarity (SAS) map showing four major regions. Regions I and II contain data pairs for scaffold hopping and smooth SAR, respectively. Region IV indicates discontinuous SAR and activity cliffs. Regions of deep and shallow activity cliffs are also shown.

3.2.4 Activity Similarity

For each pair of molecules the activity similarity for *T. vaginalis* and *G. intestinalis* was measured as follows:

$$AS_{i,j} = 1 - \frac{|A_i - A_j|}{\max - \min}$$

where A_i and A_j are the activities of the i^{th} and j^{th} molecules (pIC50 values) and max-min indicate the range of activities in the data set.

3.2.5 Activity Landscape with SAS Maps

For each target parasite, SAS maps[110] were generated by plotting the activity similarity against the structural similarity for each pair of compounds. A general form of the SAS map is presented in Figure 3.1. In this map the activity similarity is represented in the Y-axis and molecular similarity is plotted in the X-axis. A variant of the SAS maps is representing potency difference in the Y-axis[113-114].

SAS maps provide a visual and quantitative characterization of the activity landscape[104]. Four zones can be distinguished in Figure 3.1, labeled as regions I-IV. Data points that fall into region I correspond to pairs of molecules with high activity similarity and low molecular similarity and therefore are associated with regions of scaffold hopping. If the compounds in the data set share the same core scaffold and the difference are the attachment points, then region I is associated with **side chain hopping**. Points plotted in region II denote pairs of molecules with high molecular similarity and high activity similarity. Thus, compounds in this region are in a smooth or continuous SAR landscape. Data points in region III denote pairs of molecules with low molecular similarity and low activity similarity. Region IV identifies pairs of molecules that have high molecular similarity and low activity similarity and therefore correspond to activity cliffs or discontinuous SAR. Data points that are consistently put in the same region by a number of molecular representations contribute to defining a **consensus model** of the activity landscape.

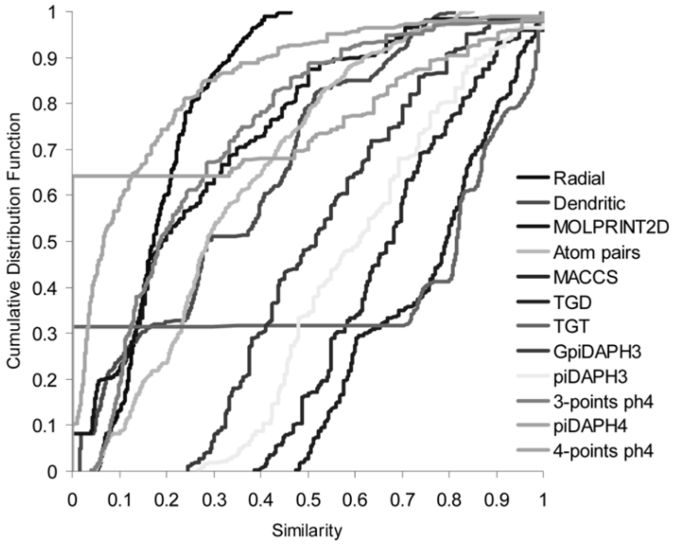
In order to characterize the SAS maps obtained with different similarity measures, each map was partitioned by imposing activity and molecular similarity thresholds along the Y- and X-axis, respectively, and then counting the number of data pairs in the resultant regions I-IV. A similar strategy was recently employed to successfully characterize potency difference vs. structure similarity plots[113]. In this study, two activity similarity thresholds were investigated namely 0.5 and 0.75 corresponding approximately to 1 and 2 log units in potency difference for *T. vaginalis*, and 0.7 and 1.4

log units for *G. intestinalis*. For similarity, the median similarity of the most active compounds in the data set was considered. For *T. vaginalis* seven compounds with pIC50 ≥ 7.00 (IC50 ≤ 100 nM) were regarded as active (9, 16, 20, 23, 24, 28 and 32). For *G. intestinalis* also seven compounds with pIC50 ≥ 7.30 (IC50 ≤ 50 nM) were regarded as active (12, 15, 16, 20, 24, 30 and 32). The reason to use slightly different thresholds of pIC50 was twofold; in order to select the same number of actives for each parasite and because the median of the pIC50 values for *G. intestinalis* is greater than the median for *T. vaginalis*. Note, however, that other thresholds for activity could be applied. Since different molecular representations leads to different ranges of similarity values for the same set of compounds, the threshold for the structural similarity depends on the representation used. Therefore normalizing to the median similarity of a descriptor allowed us the ability to compare between different descriptor sets[113].

To further compare the SAS maps obtained from different structural similarity methods we used the **Degree of Consensus** (DoC) introduced earlier[113]. DoC measures the number of data points consistently put into the same region and was computed with the following expression:

$$DoC_{m,n}^{R} = \frac{Cp_{m,n}}{p_m + p_n - Cp_{m,n}}$$

where $Cp_{m,n}$ is the number of **Consensus Pairs** in region *R* (R = I – IV) between methods *m* and *n*; p_m is the number of pairs of molecules assigned by method *m* in region *R* and p_n is the number of pairs of molecules assigned by method *n* in the same region. To note, DoC depends on the thresholds used to define each region. Results were summarized as DoC matrices.



	Max	Q3	Median	Q1	Min	Mean	STD
Radial	0.46	0.24	0.17	0.13	0.05	0.19	0.09
Dendritic	0.81	0.48	0.29	0.11	0.02	0.33	0.23
MOLPRINT 2D	1.00	0.42	0.20	0.12	0.00	0.27	0.22
Atom pairs	0.85	0.49	0.29	0.21	0.05	0.34	0.20
MACCS	1.00	0.80	0.68	0.55	0.39	0.68	0.16
TGD	1.00	0.88	0.80	0.59	0.47	0.76	0.16
TGT	1.00	0.90	0.82	0.00	0.00	0.60	0.41
GpiDAPH3	1.00	0.69	0.52	0.38	0.25	0.54	0.18
piDAPH3	1.00	0.76	0.61	0.46	0.27	0.62	0.18
3-points ph4	1.00	0.37	0.19	0.11	0.04	0.27	0.21
piDAPH4	1.00	0.58	0.00	0.00	0.00	0.24	0.34
4-points ph4	1.00	0.20	0.07	0.03	0.00	0.15	0.20

Figure 3.2 Cumulative distribution functions of 496 pairwise structural similarities using different 2D and 3D fingerprint representations. The table summarizes the information of the distributions. Q3 and Q1 indicate the third and first quartile, respectively.

3.2.6 Consensus SAS maps

To develop consensus models of the activity landscapes, we combined structural similarities obtained with uncorrelated representations into a single similarity measure. There are a number of ways to combine similarity values[121]. In this work we computed the mean similarity of four orthogonal fingerprints for this data set (radial, MACCS keys, TGD and piDAPH3) but other measures can be explored. Property similarity was not considered to obtain the mean. Consensus SAS maps for each parasite were generated by plotting the activity similarity against the mean fingerprint similarity.

3.3 Results and Discussion of Standard Methods

3.3.1 Distribution of Fingerprint Similarity Measures

The 496 pairwise similarities of the 32 benzimidazoles calculated with the 12 fingerprint-based structural representations are summarized in Figure 3.2 as cumulative distribution functions (CDF). The table at the bottom of the figure summarizes the statistics of each curve indicating the maximum, third and first quartile, median, mean, and standard deviation.

2D and 3D fingerprints showed a wide variation of distributions. 2D fingerprints with the highest similarity values were TGD, MACCS keys and GpiDAPH3 which had median values of 0.80, 0.68 and 0.52, respectively, and comparable standard deviation (0.16-0.18). 2D fingerprints with the lowest similarity values were dendritic, atom pairs, MOLPRINT 2D and radial. Concerning the 3D fingerprints, the spatial three-point pharmacophore piDAPH3 had a nearly normal distribution. In contrast, similarity values obtained with piDAPH4, 3 and 4 points pharmacophores showed non-normal distributions (as can be deduced from the non-sigmoidal shape of the corresponding CDF). Despite the 32 molecules share the benzimidazole scaffold, it was noteworthy that several fingerprints were able to differentiate the compounds detecting activity cliffs.

3.3.2 Correlation Between Molecular Similarities

The correlation between 2D and 3D fingerprint representations for the 496 pairwise similarities is shown in Table 3.2. The correlation matrix shows the

	Radial	Dendritic	MOLPRINT 2D	Atom pairs	MACCS	TGD	TGT	GpiDAPH3	piDAPH3	3-points php	piDAPH4	4-points php	Properties	AS *T. vaginalis*	AS *G. instestinalis*
Radial	1.00														
Dendritic	0.94	1.00													
MOLPRINT 2D	0.88	0.86	1.00												
Atom pairs	0.92	0.92	0.93	1.00											
MACCS	0.78	0.81	0.84	0.89	1.00										
TGD	0.71	0.82	0.69	0.76	0.77	1.00									
TGT	0.69	0.82	0.63	0.71	0.73	0.93	1.00								
GpiDAPH3	0.79	0.76	0.77	0.83	0.76	0.69	0.56	1.00							
piDAPH3	0.76	0.73	0.74	0.79	0.73	0.69	0.57	0.94	1.00						
3-points ph4[a]	0.78	0.75	0.81	0.83	0.77	0.68	0.55	0.82	0.81	1.00					
piDAPH4	0.40	0.36	0.43	0.48	0.53	0.32	0.19	0.61	0.62	0.49	1.00				
4-points ph4[a]	0.73	0.69	0.78	0.79	0.72	0.63	0.49	0.78	0.77	0.97	0.50	1.00			
Properties	0.69	0.69	0.74	0.78	0.78	0.65	0.59	0.73	0.74	0.75	0.44	0.69	1.00		
AS[b] *T. vaginalis*	0.43	0.40	0.44	0.45	0.28	0.26	0.19	0.48	0.47	0.42	0.25	0.39	0.37	1.00	
AS[b] G. *intestinalis*	0.28	0.20	0.28	0.25	0.15	-0.01	-0.05	0.29	0.30	0.27	0.17	0.26	0.20	0.31	1.00

[a] ph4: pharmacophore

[b] *AS*: activity similarity

Table 3.2 Correlation matrix for the pairwise activity, property and structure similarities

relationships between the 12 fingerprints. Additionally, the matrix shows the relationship between the fingerprint, property similarity, and activity similarity. Very high correlations between 2D methods occur for radial and dendritic; MOLPRINT 2D and atom pairs; radial and atom pairs; dendritic and atom pairs; TGD and TGT (correlation ≥ 0.92). Other high correlations between 2D methods are atom pairs and MACCS (0.89); MOLPRINT 2D and MACCS (0.84); atom pairs and GpiDAPH3 (0.83). High correlations between 3D fingerprints occur for Canvas 3 and 4 points pharmacophores (0.97), and between piDAPH3 and 3 points pharmacophore (0.82). Comparing the correlation between 2D and 3D fingerprints the highest correlation was for GpiDAPH3 and piDAPH3 (0.94). The correlation between property similarity and fingerprint similarity ranges between 0.44 (piDAPH4) and 0.78 (atom pairs and MACCS). The matrix also showed a low correlation between any of the molecular representations with activity similarity for *T. vaginalis* (correlation ≤ 0.48) or *G. intestinalis* (≤ 0.31). The correlation between activity similarities for *T. vaginalis* and *G. intestinalis* was low (0.31) indicating the presence of pairs of compounds with different effects against the two parasites.

Despite the high correlations between several 2D and 3D fingerprints we selected as many orthogonal fingerprint representations as possible to characterizing the landscapes. Thus, we selected radial, MACCS, TGD (2D) and piDAPH3 (3D). The maximum correlation between any of these five selected fingerprints was 0.78 (radial and MACCS) and the minimum correlation was 0.69 (TGD and piDAPH3). In addition, we employed property similarities as described below.

3.3.3 The Activity Landscape

SAS MAPS

Figures 3.3 and 3.4 depict the SAS maps for *T. vaginalis* and *G. intestinalis*, respectively. The maps show the relationship between activity similarity and molecular similarity obtained with four selected fingerprints and property similarity. Each plot contains 496 data points that represent a pairwise comparison. Data points were further distinguished by the activity of the most active compound in the pair in a continuous scale[113].

As expected from the CDF in Figure 3.2, similarity values obtained with radial fingerprints are shifted to the low similarity value range along the X-axis (< 0.46) while the similarity values for MACCS, TGD and piDAPH3 are more spread out. Interestingly

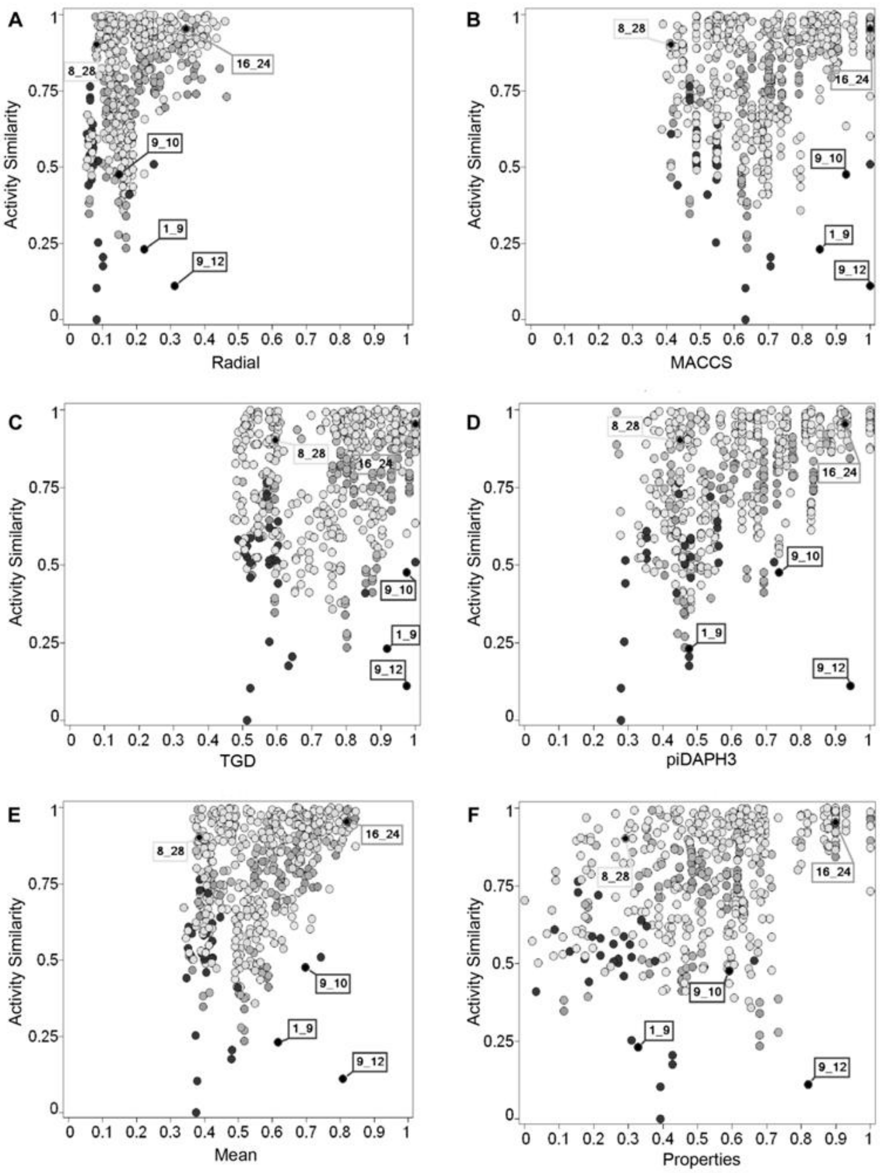


Figure 3.3 SAS maps for T. vaginalis with different structural representations. Each data point indicates a pairwise comparison of 32 benzimidazoles (496 data points total). Each panel corresponds to a different structural representation: (A) Radial; (B) MACCS keys; (C) TGD; (D) piDAPH3; (E) mean fingerprint similarity and (F) properties. Selected pairs are marked in black and labeled with the compound numbers.

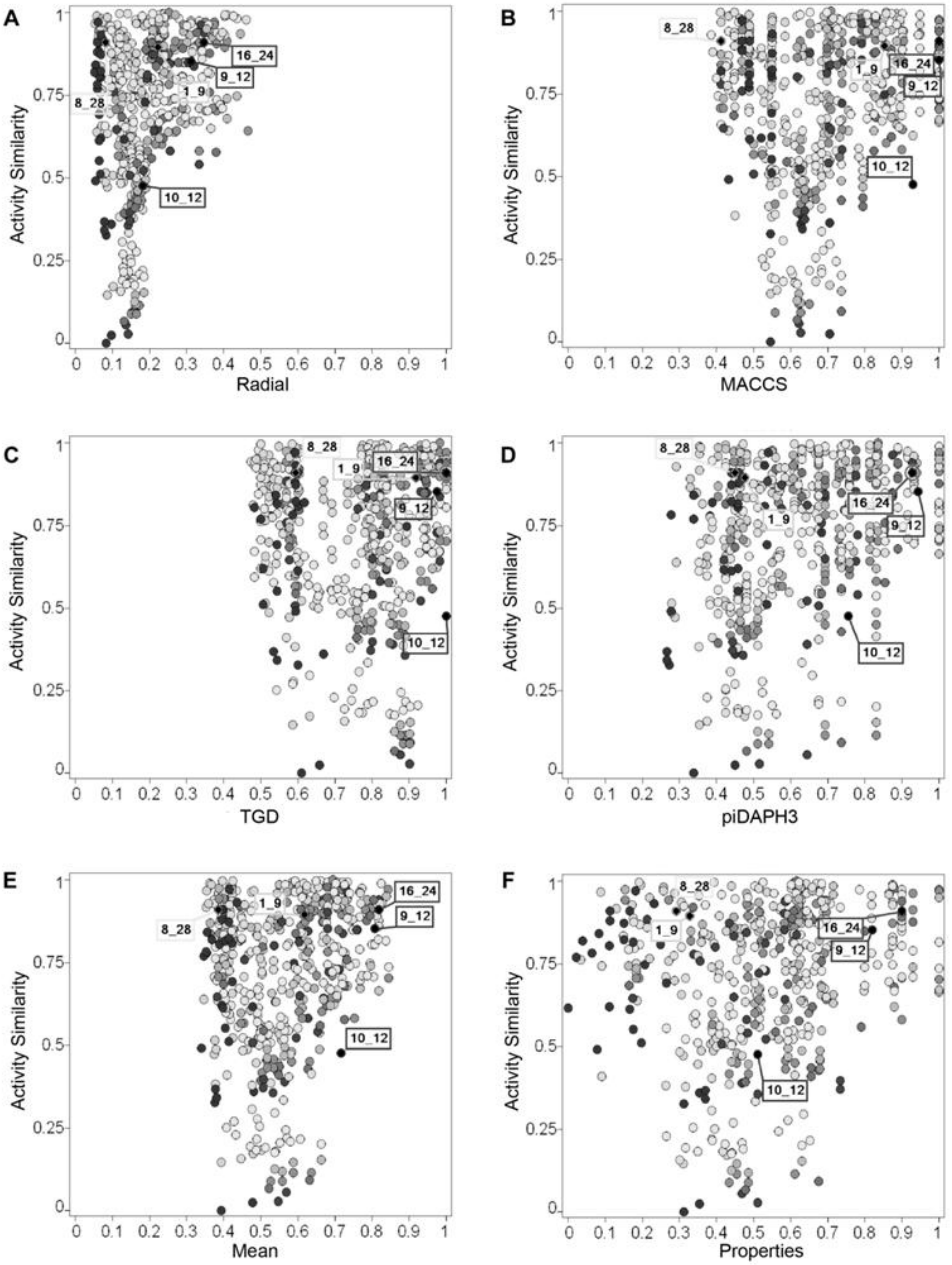


Figure 3.4 SAS maps for G. intestinalis with different structural representations. Each data point indicates a pairwise comparison of 32 benzimidazoles (496 data points total). Each panel corresponds to a different structural representation: (A) Radial; (B) MACCS keys; (C) TGD; (D) piDAPH3; (E) mean fingerprint similarity and (F) properties. Selected pairs are marked and labeled with the compound numbers.

there are several pairs in Figures 3.3 and 3.4 with similarities of 1.0 (21 pairs for MACCS keys, 15 for TGD, 18 for piDAPH3 and 8 for property similarity). A number of these points correspond to positional isomers or compounds with different halogen substitution pattern . Notably, radial fingerprints can distinguish all 496 pairs due to its high resolution. This is an expected behavior since radial fingerprints were designed for structure activity studies in contrast with topological fingerprints that were developed for substructure and similarity searching[120].

As described in the Methods section, the four regions of the SAS maps (I-IV in Figure 3.1) can also be identified in Figures 3.3 and 3.4. Data pairs in regions I and II are located in a continuous SAR while pairs of molecules in region IV represent activity cliffs. As discussed above, the boundary between regions I/II and III/IV depends on the molecular representation used. However, it is possible to detect pairs of compounds that are located in the same *relative* region of each map, i.e., **consensus pairs**. Figures 3.3 and 3.4 show examples of consensus pairs in regions I (compounds 8 and 28) and II (16 and 24) for *T. vaginalis* and *G. intestinalis*. Figure 3.3 also show a consensus pair in region IV (9 and 12) for *T. vaginalis*.

QUANTITATIVE CHARACTERIZATION OF THE SAS MAPS

In order to conduct a systematic and quantitative analysis of the data obtained in this study, the SAS maps were divided into four quadrants (regions I-IV in Figure 3.1) by defining thresholds for activity similarity and molecular similarity (see Methods section). Table 3.3 indicates the median similarity of the active compounds and the number of data pairs that can be found in regions I-IV for different molecular representations. Table 3.3 also summarizes the number of pairs with at least one active molecule or *active pairs*.

Representation	Median similarity	I		II		III		IV	
(*AS* threshold) [a]	of actives[b]	Total[c]	Active pairs[d]	Total	Active pairs	Total	Active pairs	Total	Active pairs
T. vaginalis (0.5)									
Radial	0.22	286	94	160	66	47	34	3	2
MACCS	0.71	271	102	175	58	42	31	8	5
TGD	0.95	401	138	45	22	48	34	2	2
piDAPH3	0.78	346	129	100	31	49	35	1	1
Properties	0.62	285	120	161	40	38	25	12	11
T. vaginalis (0.75)									
Radial	0.22	145	50	149	59	188	78	14	9
MACCS	0.71	142	59	152	50	171	74	31	13
TGD	0.95	254	89	40	20	195	83	7	4
piDAPH3	0.78	204	80	90	29	191	84	11	3
Properties	0.62	160	74	134	35	163	71	39	16
G. intestinalis (0.5)									
Radial	0.24	301	83	114	50	81	38	0	0
MACCS	0.79	286	91	129	42	76	35	5	3
TGD	0.93	327	90	88	43	80	37	1	1
piDAPH3	0.84	356	113	59	20	81	38	0	0
Properties	0.61	241	87	174	46	63	26	18	12
G. intestinalis (0.75)									
Radial	0.24	184	54	76	35	198	67	38	15
MACCS	0.79	171	58	89	31	191	68	45	14
TGD	0.93	199	57	61	32	208	70	28	12
piDAPH3	0.84	216	72	44	17	221	79	15	3
Properties	0.61	138	54	122	35	166	59	70	23

[a] Regions I-IV are defined by the median similarity of active compounds and a threshold of activity similarity. Two thresholds of activity similarity (*AS*) were investigated 0.5 and 0.75 (see also Figure 1 and text for details).

[b] Median similarity of compounds **9**, **16**, **20**, **23**, **24**, **28**, **32** for *T. vaginalis*, and **12**, **15**, **16**, **20**, **24**, **30**, **32** for *G. intestinalins*.

[c] Total number of data points (pair of compounds) in the region.

[d] Number of data points with at least one active compound in the pair.

Table 3.3 Distribution of data points in different regions of the SAS maps

Interestingly, the median fingerprint similarity of the actives for *T. vaginalis* and *G. intestinalis* (Table 3.3) is slightly higher than the median similarity of the 32 compounds (Figure 3.2). Considering an activity similarity threshold of 0.5 for *T. vagnalis* and *G. intestinalis*, most pairs of compounds were in region I and the frequency decreased in the order I > II > III > IV (Table 3.3). This occurred for most fingerprint representations and property similarity. A similar result was obtained for the number of active pairs.

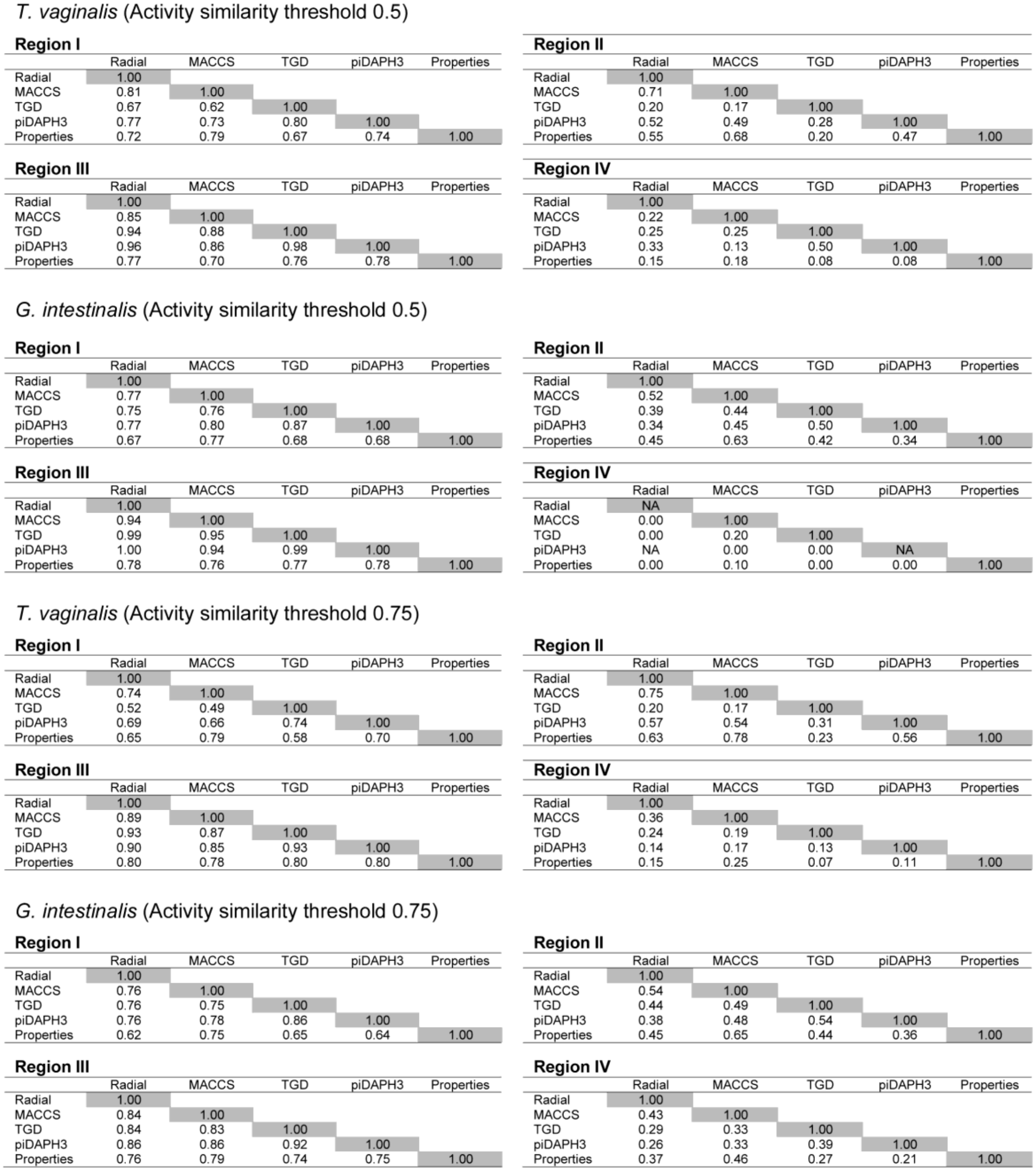
Not surprisingly, considering an activity similarity threshold of 0.75 the total number of pairs and active pairs in regions I and II decreased, and the number of pairs in regions III and IV increased (as compared to the populations at threshold of 0.5). This was observed for all representations and for both parasites (Table 3.3). However, for *G. intestinalis* most of the pairs were in region III for all molecular representations (and most representations for *T. vaginalis*). Similar results were obtained for the number of active pairs. In general, at the activity similarity threshold of 0.75 the frequency of total pairs and active pairs decreased in the order III > I > II > IV (Table 3.3).

DEEP AND SHALLOW ACTIVITY CLIFFS

The number of activity cliffs in the data set depends on the criteria to consider a pair of compounds as similar. It follows that activity cliffs can be classified further using, for example, different thresholds of activity similarity. In this work we define a **deep activity cliff** if the pair of similar compounds have “a large” difference in activity (activity similarity ≤ 0.5), and define a **shallow activity cliff** if the difference in activity is smaller (0.5 < activity similarity ≤ 0.75)[114]. This is schematically illustrated in Figure 3.1. Different activity similarity thresholds can be used to define deep and shallow cliffs. Moreover, the degree of molecular similarity can be used also to define deep or shallow activity cliffs. According to these definitions there were between one and eight deep activity cliffs in the landscape of *T. vaginalis*, depending on the fingerprint representation. The number of deep activity cliffs for *G. intestinalis* was lower, between zero and five (i.e., the number of deep activity cliffs equals the number of pairs in region IV at activity threshold of 0.5, Table 3.3). Considering molecular properties as molecular representation the number of deep cliffs for *T. vaginalis* and *G. intestinalis* was 12 and 18, respectively. Noteworthy, the numbers of deep and shallow cliffs, as defined in this work, are relative to the data set. This is because deep and shallow cliffs are defined based on activity similarity that depends on the activity range of the data set.

The number of shallow activity cliffs can be calculated from Table 3.3 taking the difference between the number of total pairs in region IV at activity similarity thresholds of 0.75 and 0.5. For example, for *T. vaginalis* there are 14 – 3 = 11 shallow cliffs considering radial fingerprints and 31 – 8 = 23 shallow cliffs considering MACCS keys. Noteworthy, for all molecular representations the number of shallow cliffs for *G. intestinalis* was higher than the number of shallow cliffs for *T. vaginalis*. Examples of deep and shallow activity cliffs are discussed below.

Results above and previous studies, support the importance of considering several representations[113-114] and lead to the following question: Are there *consensus pair*s?[113] and is it possible to derive a consensus model of the activity landscape for a given data set?



T. vaginalis (Activity similarity threshold 0.5)

Region I

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.81	1.00			
TGD	0.67	0.62	1.00		
piDAPH3	0.77	0.73	0.80	1.00	
Properties	0.72	0.79	0.67	0.74	1.00

Region II

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.71	1.00			
TGD	0.20	0.17	1.00		
piDAPH3	0.52	0.49	0.28	1.00	
Properties	0.55	0.68	0.20	0.47	1.00

Region III

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.85	1.00			
TGD	0.94	0.88	1.00		
piDAPH3	0.96	0.86	0.98	1.00	
Properties	0.77	0.70	0.76	0.78	1.00

Region IV

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.22	1.00			
TGD	0.25	0.25	1.00		
piDAPH3	0.33	0.13	0.50	1.00	
Properties	0.15	0.18	0.08	0.08	1.00

G. intestinalis (Activity similarity threshold 0.5)

Region I

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.77	1.00			
TGD	0.75	0.76	1.00		
piDAPH3	0.77	0.80	0.87	1.00	
Properties	0.67	0.77	0.68	0.68	1.00

Region II

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.52	1.00			
TGD	0.39	0.44	1.00		
piDAPH3	0.34	0.45	0.50	1.00	
Properties	0.45	0.63	0.42	0.34	1.00

Region III

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.94	1.00			
TGD	0.99	0.95	1.00		
piDAPH3	1.00	0.94	0.99	1.00	
Properties	0.78	0.76	0.77	0.78	1.00

Region IV

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	NA				
MACCS	0.00	1.00			
TGD	0.00	0.20	1.00		
piDAPH3	NA	0.00	0.00	NA	
Properties	0.00	0.10	0.00	0.00	1.00

T. vaginalis (Activity similarity threshold 0.75)

Region I

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.74	1.00			
TGD	0.52	0.49	1.00		
piDAPH3	0.69	0.66	0.74	1.00	
Properties	0.65	0.79	0.58	0.70	1.00

Region II

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.75	1.00			
TGD	0.20	0.17	1.00		
piDAPH3	0.57	0.54	0.31	1.00	
Properties	0.63	0.78	0.23	0.56	1.00

Region III

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.89	1.00			
TGD	0.93	0.87	1.00		
piDAPH3	0.90	0.85	0.93	1.00	
Properties	0.80	0.78	0.80	0.80	1.00

Region IV

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.36	1.00			
TGD	0.24	0.19	1.00		
piDAPH3	0.14	0.17	0.13	1.00	
Properties	0.15	0.25	0.07	0.11	1.00

G. intestinalis (Activity similarity threshold 0.75)

Region I

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.76	1.00			
TGD	0.76	0.75	1.00		
piDAPH3	0.76	0.78	0.86	1.00	
Properties	0.62	0.75	0.65	0.64	1.00

Region II

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.54	1.00			
TGD	0.44	0.49	1.00		
piDAPH3	0.38	0.48	0.54	1.00	
Properties	0.45	0.65	0.44	0.36	1.00

Region III

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.84	1.00			
TGD	0.84	0.83	1.00		
piDAPH3	0.86	0.86	0.92	1.00	
Properties	0.76	0.79	0.74	0.75	1.00

Region IV

	Radial	MACCS	TGD	piDAPH3	Properties
Radial	1.00				
MACCS	0.43	1.00			
TGD	0.29	0.33	1.00		
piDAPH3	0.26	0.33	0.39	1.00	
Properties	0.37	0.46	0.27	0.21	1.00

Figure 3.5 Degree of consensus (DoC) matrices for each region. Each entry in the corresponding matrix represents the agreement between two methods to place a pair of compounds into the same region.

Pair	Activity similarity		Fingerprint similarity					Property similarity
	T. vaginalis	*G. intestinalis*	Radial	MACCS	TGD	piDAPH3	Mean (STD)	
4_28[a]	0.911	0.180	0.138	0.612	0.848	0.561	0.540 (0.295)	0.443
8_28[a]	0.902	0.910	0.082	0.413	0.594	0.450	0.385 (0.216)	0.291
10_28[a]	0.868	0.658	0.080	0.413	0.618	0.487	0.400 (0.229)	0.348
21_28[a]	0.952	0.692	0.188	0.617	0.805	0.620	0.557 (0.262)	0.503
28_32[a,b]	0.969	0.887	0.190	0.698	0.945	0.693	0.631 (0.317)	0.603
7_12[b]	0.993	0.342	0.079	0.632	0.544	0.266	0.380 (0.254)	0.369
16_20[a,b]	0.964	0.970	0.375	1.000	1.000	1.000	0.844 (0.313)	1.000
16_24[a,b]	0.954	0.910	0.346	1.000	1.000	0.928	0.818 (0.317)	0.900
16_28[a,b]	0.844	0.940	0.351	0.905	0.949	0.941	0.786 (0.291)	0.900
15_16[a,b]	0.739	0.970	0.351	0.708	0.983	0.836	0.719 (0.270)	0.616
16_19[a,b]	0.751	0.733	0.253	0.708	0.983	0.836	0.695 (0.315)	0.616
16_30[a,b]	0.799	0.992	0.269	0.791	0.931	0.836	0.707 (0.297)	0.615
16_31[a,b]	0.741	0.944	0.375	0.791	0.931	0.836	0.733 (0.246)	0.615
8_9[a]	0.511	0.876	0.250	1.000	1.000	0.723	0.743 (0.354)	0.664
1_9[a]	0.233	0.895	0.222	0.852	0.918	0.476	0.617 (0.327)	0.327
9_10[a]	0.477	0.624	0.148	0.929	0.975	0.737	0.697 (0.380)	0.591
9_12[a]	0.113	0.853	0.313	1.000	0.975	0.944	0.808 (0.331)	0.819
10_12[b]	0.635	0.477	0.182	0.929	1.000	0.756	0.717 (0.371)	0.511

[a] Active pair for *T. vaginalis*

[b] Active pair for *G. intestinalis*

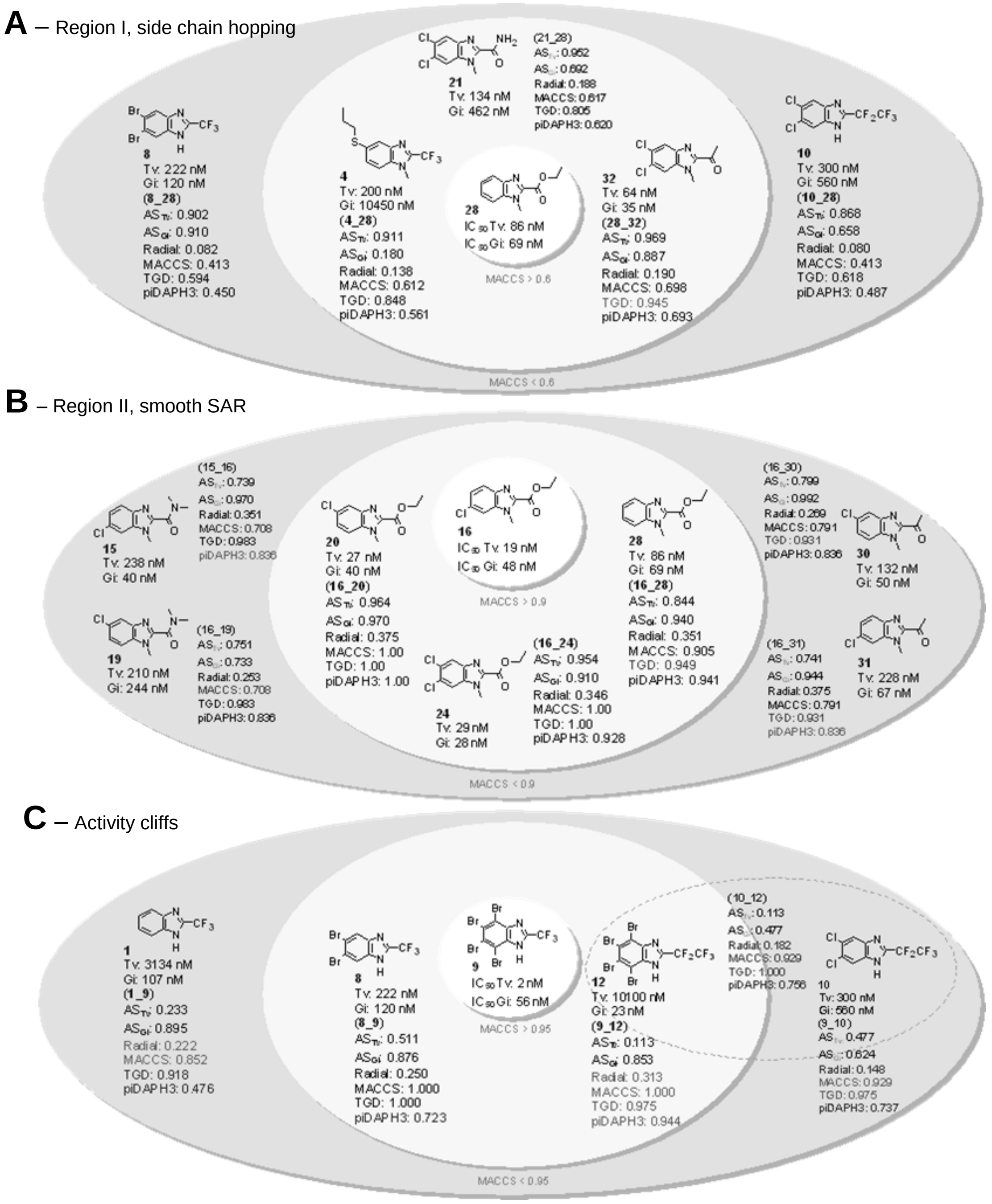
Table 3.4 Examples of consensus pairs of compounds in the SAS maps

DEGREE OF CONSENSUS

Despite the mid-to-low correlations between the molecular representations (Table 3.2) and the different distributions of pairwise similarity values (Figure 3.2), it was possible to find a number of consensus pairs in different regions of the landscape. The number of pairs of compounds that two methods put into the same region i.e., number of consensus pairs, were recorded for *T. vaginalis* and *G. intestinalis* at the two activity similarity thresholds. Examples of consensus pairs are discussed later in this section. The degree of consensus (DoC) between two methods is presented in Figure 3.5. For both parasites, at the two activity similarity thresholds, DoC has high values in region III (0.83 - 1.00) followed by region I (0.49 – 0.98). In contrast, DoC has low values in region IV in particular for *G. intestinalis* at activity similarity threshold of 0.5 (0.0 – 0.2). This means that there was a better agreement between the methods to assign molecules to region III than in any other region. In contrast, it was more difficult to identify consensus activity cliffs than to identify consensus pairs in continuous regions of the SAR. Note that DoC is dependent on the criteria used to define the four regions.

Table 3.4 lists several examples of consensus pairs in the three most informative regions (I-II and IV) of the SAS map of *T. vaginalis* and *G. intestinalis*. Table 3.4 also lists the molecular similarity for selected fingerprint representations, property similarity and activity similarity.

For *T. vaginalis,* several examples of consensus pairs with low structural similarity and high activity similarity (region I) involve the active compound 28 (IC50 = 86 nM) such as 4_28, 8_28, 10_28, 21_28 and 32_28 (Table 4). Figure 3.6A shows a comparison of the chemical structures for selected pairs in region I along with the activity



A – Region I, side chain hopping

B – Region II, smooth SAR

C – Activity cliffs

Figure 3.6 Representative consensus pairs in the activity landscapes of T. vaginalis and G. intestinalis arranged in concentric similarity ovals compounds in the inner ring are more structurally similar to the reference (center) than compounds in the outer ring: (A) region I, side chain hopping; (B) region II, smooth SAR and (C) activity cliffs.

and molecular similarity measures. In this figure, concentric ovals indicate different degrees of structural similarity to 28. For example, compounds 8 and 10 are less similar to 28 as compared to 4, 21 and 32. Interestingly, 8 and 10 are also less active than 4, 21 and 32. Since all compounds in the set have the same scaffold, pairs in region I can be considered as examples of side chain hopping (see above). Noteworthy, several whole-molecule fingerprints used in this study were able to detect low similarity due to the side chain substitutions. For example, the similarity for the above mentioned pairs with the known "low resolution" 166-bits MACCS keys,37 ranges between 0.698 (32_28) and 0.413 (8_28 and 10_28). In contrast, for the same pairs the radial similarity ranges between 0.190 (32_28) and 0.080 (10_28).

Some of the pairs in region I of the landscape of *T. vaginalis* were also in region I of the landscape of *G. intestinalis*. Examples were 8_28 and 28_32 with activity similarly values of 0.910 and 0.887, respectively (Table 3.4 and Figure 3.6A). These results suggest a similar SAR for both parasites. Notable exceptions were 4_28 and 7_12 which have low activity similarity for *G. intestinalis* (0.18 and 0.342, respectively) indicating that in some instances the same change in the structure of the benzimidazoles produces different effects in the activity of *T. vaginalis* and *G. intestinalis*.

We also identified several consensus pairs in region II of the landscape of *T. vaginalis*. To note, a number of these pairs, with high structure similarity and high activity similarity (> 0.84), included the active compound 16 (IC50 = 19 nM). Examples are pairs 16_20, 16_24 and 16_28 (Figure 3.6B). Interestingly, several fingerprint representations including MACCS keys, TGD and piDAPH3 were unable to distinguish the positional isomer 16_20 (similarity of 1.0). However, the radial fingerprint did differentiate this pair demonstrating the high resolution of this type of fingerprints[122]. All compounds in these pairs have an ethyl ester at R2 and are in a smooth region of landscape; changes in the substitution pattern with chlorine at positions 5 and 6 of the benzimidazole scaffold produces small changes in the activity (activity similarity between 0.844 – 0.964).

The pairs 16_15, 16_19, 16_30 and 16_31 are also located in region II of the landscape of *T. vaginalis* (Figure 3.6B). The structural similarity of 15, 19, 30 and 31 with respect to 16 decreases (as captured by several molecular representations) and the activity similarity also decreases (down to 0.739 – 0.799). These results schematically illustrated in Figure 3.6B, are in agreement with the 'similarity principle'[123] and further illustrate the smooth SAR associated with region II. To note, none of these compounds have an ethyl ester at R2 emphasizing the importance of this substituent in the activity against *T. vaginalis*.

Several pairs with high structure similarity and high activity similarity for *T. vaginalis* were also located in region II of the landscape of *G. intestinalis* as illustrated by the pairs containing compound 16 (IC50 = 48 nM, Table 3.4). These results indicated that, in general, substitution with ethyl ester at R2 increases the activity against both parasites. Similarly, substitution with one or two chlorine atoms at positions 5 and 6 of the benzimidazole scaffold does not affect significantly the activity with both parasites.

CONSENSUS DEEP AND SHALLOW ACTIVITY CLIFFS AND APPARENT CLIFFS

The importance of activity cliffs in activity landscapes has been discussed in literature[102,104,106]. Activity cliffs are valuable for detecting specific structural changes important for activity. Furthermore, consensus activity cliffs have been conceptualized as those cliffs that occur across different molecular representations[113]. Figure 3.6C depicts examples of consensus cliffs (region IV) for different molecular representations.

For *T. vaginalis* we found only one **consensus deep cliff**, pair 9_12 (cliff for all molecular representations with at least two log units in potency difference, Table 3.4). The structural difference between 9 and 12 is a CF2 group at R2 (CF3 versus CF2-CF3). This change in structure produces a dramatic decrease in activity against *T. vaginalis* from IC50 = 2 nM (9) to 10000 nM (12). It is anticipated that any of these two compounds or both would be apparent outliers in a QSAR study[102]. Interestingly, 9_12 was not an activity cliff for *G. intestinalis* (IC50 = 56 nM versus 23 nM, respectively). A second consensus deep cliff in the landscape of *T. vaginalis* was the pair 1_9, detected by radial and MACCS keys only. This is an example of an **apparent cliff** conceptualized as cliff identified just for some molecular representations[113]. One more example of an apparent cliff is the pair 9_10 identified by MACCS keys and TGD. A border line case between deep and shallow cliff for *T. vaginalis* is 8_9 (activity similarity value of 0.511) identified as cliff by radial, MACCS and TGD (but not by piDAPH3). Note, however, that MACCS and TGD could not distinguish this pair (similarity = 1).

Few consensus activity cliffs were identified in the landscape of *G. intestinalis*. For example, the pair 10_12 (Table 3.4 and Figures 3.4 and 3.6C) was identified as deep cliff by MACCS and TGD only (e.g., apparent cliff). As discussed above, when the activity similarity threshold is set to less restrictive changes in potency difference the number of activity cliffs increases. An example of a shallow (and also apparent) cliff was the pair 9_10 identified by MACCS and TGD. This pair was also identified as an apparent cliff in the landscape of *T. vaginalis* (Table 3.4).

The presence of pairs of compounds in continuous and discontinuous regions of the landscape for *T. vaginalis* and *G. intestinalis* revealed the heterogeneous SAR for both parasites. Heterogeneous SAR have been reported for other activity classes[101,113-114]. The landscape of *T. vaginalis* is characterized by the presence of two consensus deep activity cliffs and several data points in continuous regions of the SAR. In contrast, the landscape of *G. intestinalis* did not show consensus deep activity cliffs but a larger number of shallow cliffs as compared to *T. vaginalis*.

3.3.4 Consensus Models of the Activity Landscape: Consensus SAS Maps

The activity landscape depends on the molecular representation[113-114]. However, the consensus pairs found in several regions of the landscape suggests the possibility to derive, at least approximately, consensus models of the activity landscape. To this end, we employed in this work the principles of data fusion[121] producing Consensus SAS maps. For each pair of compounds we calculated the mean and standard deviation of radial, MACCS, TGD and piDAPH3 similarity values (four selected orthogonal fingerprints) as detailed above. Figures 3.3E and 3.4E show the consensus SAS maps for *T. vaginalis* and *G. intestinalis*, respectively. Figures 3E and 4E shows the position of representative consensus pairs previously identified by radial, MACCS, TGD and piDAPH3 fingerprints (Figures 3.3A-D and 3.4A-D). The mean fingerprint similarity values and standard deviation is provided in Table 3.4. In general, the pair of compounds in the consensus SAS map in Figure 3.3E occupies a relative similar position (regions I-IV) as in the SAS maps obtained separately with the radial, MACCS, TGD and piDAPH3 fingerprints (Figures 3.3A-D). Similar results were obtained comparing the consensus SAS map of *G. intestinalis* (Figure 3.4E) with the SAS maps obtained independently with the four fingerprints (Figures 3.4A-D). Therefore, the consensus SAS maps captured well the information obtained with different fingerprint representations and provided a good approximation of the overall activity landscape of the benzimidazoles tested against *T. vaginalis* and *G. intestinalis*. These results suggested that consensus SAS maps could provide valuable information for other data sets with other biological endpoints.

3.3.5 Dual-Parasite SAR and Consensus Selectivity Cliffs

We compared the activity of the 32 benzimidazoles against the two parasites. The difference of pIC50 values is indicated in Table 3.1. Several compounds showed a similar activity (low ΔpIC50) with *T. vaginalis* and *G. intestinalis*. For example, compounds 13, 14, 17-19, 20, 22-24, 27 and 28 have a $|\Delta pIC50| < 0.20$. These results suggest that these non 2-methylcarbamates have a common mechanism of action against the two protozoan. In addition, these results encourage the simultaneous lead optimization of compounds active against both *T. vaginalis* and *G. intestinalis*.

We also identified molecules with large potency difference against the two parasites. Compounds 1, 6 and 12 are selective for *G. intestinalis* whereas 4 and 9 are selective for *T. vaginalis* with more than one log unit in potency difference, respectively (Table 3.1). Interestingly, the pair of compounds 9_12 has a high structural similarity; the only difference is a CF2 group (this difference was captured by all similarity methods, Table 3.4). However, the selectivity is quite different. This is an example of a selectivity cliff where a small change in the structure has a major impact in the selectivity[112]. A second example of a selectivity cliff was the pair 1_9 indicating that introducing bromine atoms in 2-(trifluoromethyl)benzimidazole has an opposite effect in the selectivity profile (Table 3.1). The pair 8_9 also has high structural similarity (Table 3.4; the difference is two bromine atoms at positions 4 and 7) but produces an opposite change in the activity against the two parasites. However, the effect for the pair 8_9 is less dramatic than for the pairs 9_12 and 1_9. To note, 1_9 and 9_12 are deep cliffs while 8_9 is a shallow cliff in the landscape of *T. vaginalis*. However, the same pairs are in a smooth region of the landscape of *G. intestinalis* (Figures 3.3 and 3.4, respectively).

3.3.6 Conclusions and Perspectives

We have reported a systematic characterization of the SAR of 32 (non 2-methylcarbamate) benzimidazoles with activity against *T. vaginalis* and *G. intestinalis*. The analysis was based on pairwise comparisons of the activity similarity and molecular similarity using different molecular representations. The chemical structures were represented using a set of six general drug-like properties and 12 2D and 3D structural fingerprints. The correlation between the structural fingerprints was evaluated. Four fingerprints namely radial, MACCS keys, TGD and piDAPH3 showed low correlation (< 0.80) and similar (approximately normal) distributions. The four uncorrelated fingerprints, along with molecular properties, were used to characterize the activity landscape for each parasite. To note, the purpose of using multiple molecular representations was not to identify the “best” representation but to identify the set of fingerprint representations that helps to find consensus data pairs. The landscape was portrayed using Structure-Activity Similarity (SAS) maps. The SAS maps were categorized into four regions for further quantitative comparisons using the degree of consensus. The overall good consensus between structural representations allowed developing consensus models for the activity landscape for *T. vaginalis* and *G. intestinalis*. The consensus models were represented using consensus SAS maps. For both parasites, several consensus pairs of compounds were identified in the smooth region of the landscape. Also a number of pairs were identified in the side chain hopping region. For *T. vaginalis* we identified two deep consensus activity cliffs (1_9 and 9_12). It is anticipated that these compounds will be apparent "outliers" in traditional computational models such as QSAR. In contrast, for *G. intestinalis* we identified apparent cliffs and shallow cliffs. In summary, a heterogeneous SAR was found for both parasites. We also compared the compounds selectivity for each parasite. Several compounds were active for both parasites and showed similar SAR for *T. vaginalis* and *G. intestinalis*. These results suggested that the molecules may have similar mechanism of action in both parasites and encourage simultaneous lead optimization efforts against both organisms. However we also detected molecules with opposite selectivity profile, in particular poly-bromated molecules. In addition we found consensus selectivity cliffs.

Despite the data set of molecules studied in this work share the same core scaffold, whole-molecule fingerprint-based similarity methods were able to study the activity landscape, derive a consensus model of the landscape and, in particular, detect activity cliffs. Radial fingerprints were able to distinguish the molecules in great detail and differentiate positional isomers.

Expansion of this systematic study of the SAR to larger data sets including compounds currently synthesized and tested is ongoing[52]. The systematic approach presented here to develop consensus models of the activity landscape of benzimidazole analogues against *T. vaginalis* and *G. intestinalis* is general. The approach can be applied to other larger data sets with other biological end-points[53].

3.4 Predictivity Analysis of Structural and Property Similarities

The above analysis of the SAR is entirely descriptive in nature; in particular, it makes no claims that the overall behavior of the SAR will be useful in when assessing future groups of compounds with the intent of finding lead compounds. Herein, we present a novel approach using conditional probability distributions to both relatively compare how portable the SAR information is for the varying similarity methods, and also to assess how accurate these methods may be in determining similarity to potential leads.

3.4.1 Threshold Determination Heuristic

For a training set *T* of molecules with known activity on a specific target, the pairwise structural similarity scores, property similarity scores, and activity similarity scores can all be calculated, and analyses such as in Section 3.3 can be performed. Given another disjoint set of molecules *S* for which the activities on the same target are unknown, the goal is to be able to predict structural-activity relationships amongst the pairs of molecules from set *S*, and also to determine if some molecules from set *S* have similar activities to the known activities of molecules in set *T*. We propose using the joint probability distribution of activity similarity and structural similarity on the set $\boldsymbol{T} \times \boldsymbol{T}$ of comparisons of molecules from set *T* to determine the range of structural similarity scores that give meaningful activity similarity information. In particular, for a fixed activity similarity threshold $\boldsymbol{A}^{*}$ and a given confidence $\boldsymbol{C}^{*}$ we calculate the minimum *s*, which we will call $\boldsymbol{s}^{*}$, for which $\boldsymbol{P}(\boldsymbol{AS} \geq \boldsymbol{A}^{*} | \boldsymbol{SS} \geq \boldsymbol{s}) \geq \boldsymbol{C}^{*}$ on the set of activity scores *AS* and structural scores *SS* on $\boldsymbol{T} \times \boldsymbol{T}$. Essentially, we are searching for the minimal structural similarity for which the likelihood of a large activity discrepancy (i.e., an activity cliff) is small. We then posit that this $\boldsymbol{s}^*$ may be used as a threshold for structural similarity scores on the set $\boldsymbol{S} \times \boldsymbol{S}$ of pairs of molecules from set *S* to specify those pairs of molecules most likely to have similar activities. Also, we hypothesize that this $\boldsymbol{s}^*$ may be applied as a threshold to structural similarity scores on the set $\boldsymbol{T} \times \boldsymbol{S}$ of comparisons of molecules between the two sets, for the purpose of determining what molecules in set *S* may have similar activities to the known activities of the molecules in set *T*.

3.4.2 Testing Methodology

To test is the above process is indeed an accurate way of predicting activity similarity, we use an extension of the above dataset of 78 compounds as described in [51-52]. As above, we use the structural fingerprints Radial, Pharm4Pts, MACCS, piDAPH3, and TGD, as well as the Property Similarity Scores described above. We again use activity similarity information for both *T. vaginalis* and *G. intestinalis*. The activity similarity threshold was set to one order of magnitude, which for *T. vaginalis* was 0.760 and for *G. intestinalis* was 0.624. The confidence was set to 0.95. For a single testing trial, the dataset was randomly subdivided into two equal sets of 39 compounds; 2-fold cross-validation was used so that each of the two sets was both the training and test set, and both were used to analyze the set of comparisons between the two sets. Thus for a single testing trial each compounds was used in both the testing and training sets. For each target, a total of 1000 testing trials were performed using different randomly generated subdivisions of the set of compounds, so that a total of 2000 predictions were performed on each of the sets of comparisons within the test set (which we will call the **disjoint** set), and comparisons between the training and testing set.

For analysis of the disjoint set, we are interested in whether the $\boldsymbol{s}^*$ described above can also predict the absence of activity cliffs in the comparisons between novel compounds; in other words, is the relative absence of activity cliffs in the specified region a true representation of the activity landscape under the specified similarity method, or is it only a subject of the particular compounds involved. Therefore, for each trial and for both the cross-validated sets we calculate the total number of compounds in the test set for which the similarity is greater than $\boldsymbol{s}^*$, $\boldsymbol{N}_{\boldsymbol{s}\geq\boldsymbol{s}^*}$, and also the number of compounds which meet to previous criterion and are also close in activity, $\boldsymbol{N}_{\boldsymbol{AS}\geq\boldsymbol{A}^*,\boldsymbol{SS}\geq\boldsymbol{s}^*}$. We then define the **success rate** as $\frac{\boldsymbol{N}_{\boldsymbol{AS}\geq\boldsymbol{A}^*,\boldsymbol{SS}\geq\boldsymbol{s}^*}}{\boldsymbol{N}_{\boldsymbol{s}\geq\boldsymbol{s}^*}}$. For comparison, we also perform the above process on similarity scores that offer no information (i.e., all values are fixed to 1.0) and for "perfect" similarity scores which equal the activity similarity.

For analysis of comparisons between the training and testing set, we are primarily interested in determining if comparisons with active compounds in the training set which have similarity greater than $\boldsymbol{s}^*$ are indicative of active compounds in the test set, and to what extent this procedure can classify compounds in the test set at all. An active compound is defined as having a $pIC_{50} > 7.00$ for *T. vaginalis* and a $pIC_{50} > 7.30$ for *G. intestinalis* as previously discussed in [51-51]. For each trial and for both the cross-validated sets we calculate the total number of active compounds in the test set, $\boldsymbol{N}_{\boldsymbol{Active\ in\ Test\ Set}}$. For each compound in the test set, the number of times it is involved in a comparison with an active compound for which the similarity exceeds $\boldsymbol{s}^*$ is calculated

			Success	Success Rate	Success	% Trials
Score	Target	Test Set	Rate Q1	Median	Rate Q3	> 90%
Property Similarity	T. vaginalis	Disjoint	90.0%	100.0%	100.0%	73.10%
Radial	T. vaginalis	Disjoint	88.2%	100.0%	100.0%	69.10%
Pharm4Pts	T. vaginalis	Disjoint	89.5%	100.0%	100.0%	71.75%
MACCS	T. vaginalis	Disjoint	75.0%	83.3%	91.7%	30.00%
piDAPH3	T. vaginalis	Disjoint	91.5%	96.4%	100.0%	81.15%
TGD	T. vaginalis	Disjoint	75.0%	83.7%	91.2%	28.90%
No Score	T. vaginalis	Disjoint	55.2%	58.3%	61.8%	0.00%
Perfect	T. vaginalis	Disjoint	95.7%	96.6%	97.3%	98.25%
Property Similarity	G. intestinalis	Disjoint	92.9%	100.0%	100.0%	86.40%
Radial	G. intestinalis	Disjoint	87.9%	94.7%	100.0%	66.05%
Pharm4Pts	G. intestinalis	Disjoint	87.5%	96.4%	100.0%	66.00%
MACCS	G. intestinalis	Disjoint	92.9%	100.0%	100.0%	79.20%
piDAPH3	G. intestinalis	Disjoint	90.9%	95.0%	100.0%	76.85%
TGD	G. intestinalis	Disjoint	81.0%	86.0%	90.9%	29.10%
No Score	G. intestinalis	Disjoint	67.2%	69.0%	70.5%	0.00%
Perfect	G. intestinalis	Disjoint	95.1%	96.0%	96.5%	99.95%

Table 3.5 Distributions of success rates when testing the proposed structural predictivity methodology on the described data set.

as $\boldsymbol{N_{Compared\ to\ Active}}$. Similarly, the number of times it is involved in a comparison with an inactive compound for which the similarity exceeds $\boldsymbol{s}^*$ is calculated as $\boldsymbol{N_{Compared\ to\ Inactive}}$, and the overall percentage of times it is compared favorably to an active compound is calculated to be $\frac{\boldsymbol{N_{Compared\ to\ Active}}}{\boldsymbol{N_{Compared\ to\ Active}}+\boldsymbol{N_{Compared\ to\ Inactive}}}$. If this ratio exceeds 0.5 (i.e., if $\boldsymbol{N_{Compared\ to\ Active}} > \boldsymbol{N_{Compared\ to\ Inactive}}$) then the compound is classified as active; otherwise, it is classified as inactive. For each trial the total percentage of compounds in the test set classified as either active or inactive, $\%_{\boldsymbol{Classified}}$, is recorded. Also, the total number of compounds classified as active, $\boldsymbol{N_{Classified\ Active}}$, and the number of compounds both classified as active and actually active,

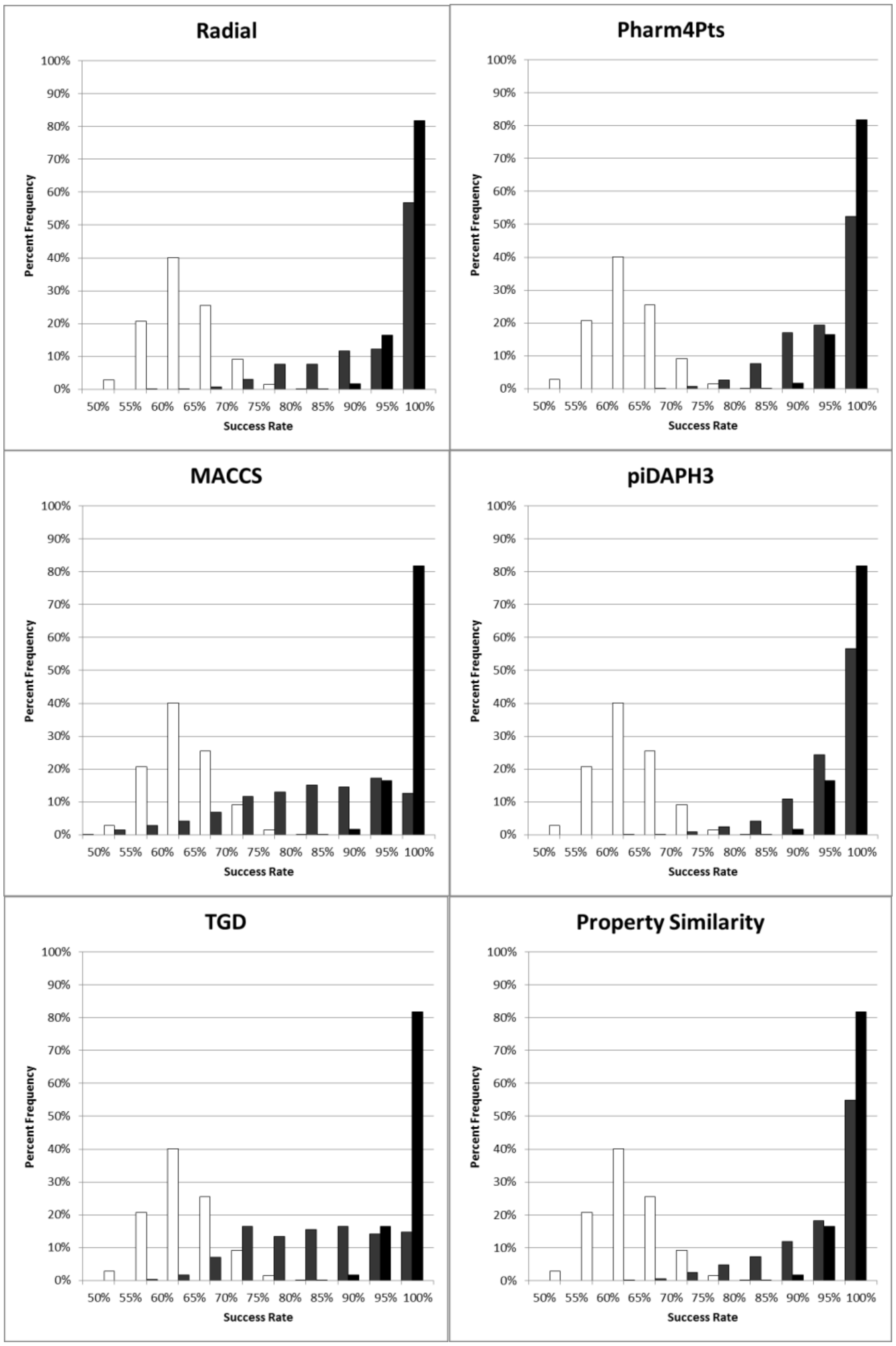


Figure 3.7 Distributions of success rates when testing the proposed structural predictivity methodology for T. vaginalis against the disjoint set of comparisons. Structural similarity method (grey) is shown compared to no structural information (white) and perfect structural information (black).

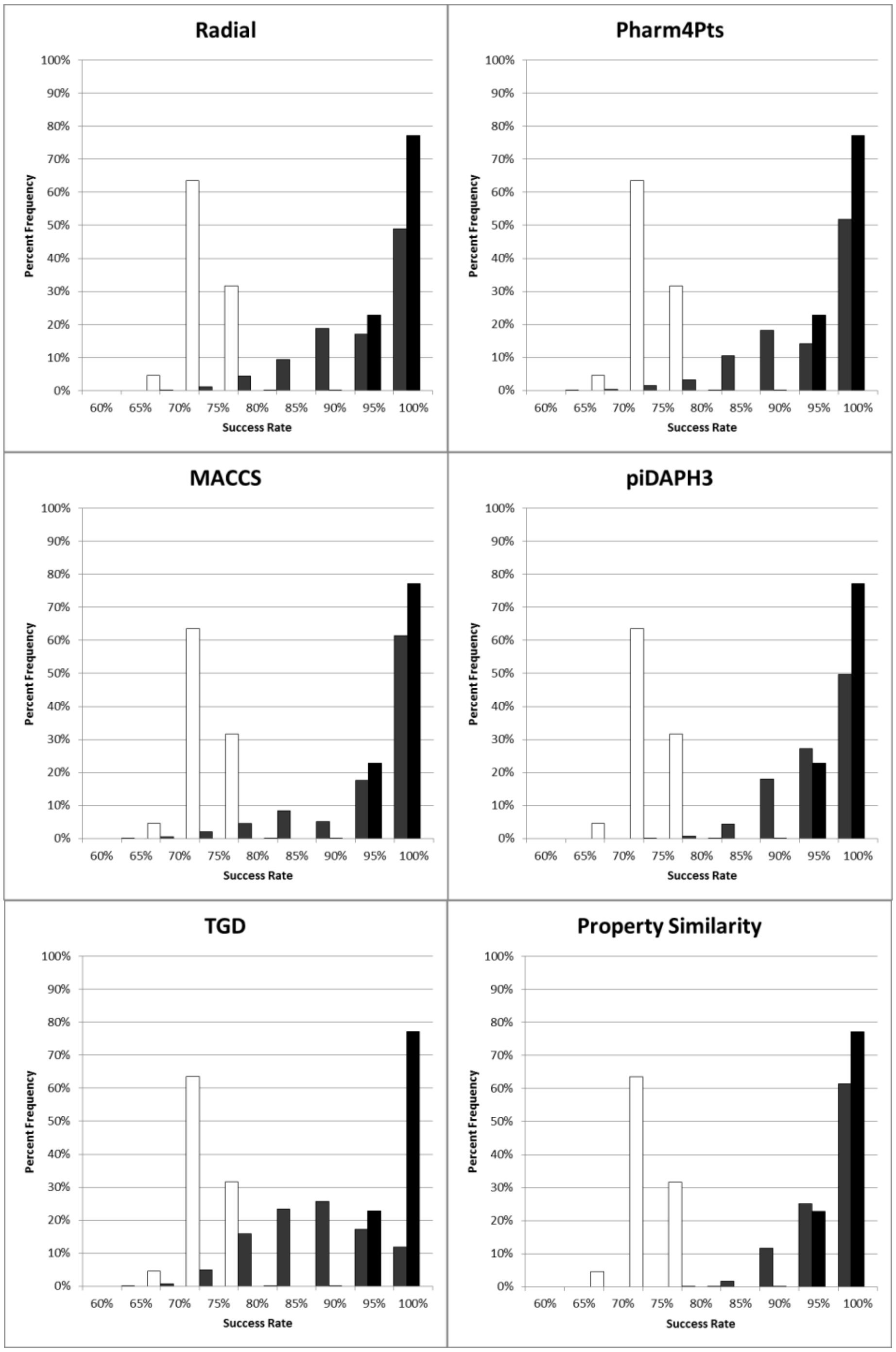


Figure 3.8 Distributions of success rates when testing the proposed structural predictivity methodology for G. intestinalis against the disjoint set of comparisons. Structural similarity method (grey) is shown compared to no structural information (white) and perfect structural information (black).

Similarity Method	Target	Median Coverage Rate	Median True Positive Rate	% of Trials with > 50% Coverage	% of Trials with > 75% True Positive	% of Trials w/ both > 50% Coverage & > 75% True Positive
Property Similarity	T. vaginalis	46.67%	90.00%	38.45%	85.35%	32.40%
Radial	T. vaginalis	63.16%	90.91%	77.35%	82.30%	62.50%
Pharm4Pts	T. vaginalis	33.33%	100.00%	18.35%	79.70%	12.70%
MACCS	T. vaginalis	46.67%	85.71%	37.05%	69.30%	27.35%
piDAPH3	T. vaginalis	63.16%	81.82%	65.60%	67.20%	41.15%
TGD	T. vaginalis	75.00%	85.71%	90.65%	82.45%	76.30%
Property Similarity	G. intestinalis	41.67%	77.78%	26.65%	51.60%	11.30%
Radial	G. intestinalis	41.67%	77.78%	26.85%	50.65%	7.30%
Pharm4Pts	G. intestinalis	18.18%	50.00%	6.75%	8.80%	0.00%
MACCS	G. intestinalis	57.14%	75.00%	60.25%	44.65%	26.30%
piDAPH3	G. intestinalis	23.08%	42.86%	10.45%	4.10%	0.05%
TGD	G. intestinalis	60.00%	61.54%	68.25%	10.10%	5.65%

Table 3.6 Distributions of coverage and true positive rates when testing the proposed structural predictivity methodology on the set of comparisons between the training and test sets.

$\boldsymbol{N_{Classified\ Active\ \&\ Active}}$, are recorded. Finally, the **coverage rate** is calculated as $\frac{\boldsymbol{N_{Classified\ Active\ \&\ Active}}}{\boldsymbol{N_{Active\ in\ Test\ Set}}}$ and the **true positive rate** as $\frac{\boldsymbol{N_{Classified\ Active\ \&\ Active}}}{\boldsymbol{N_{Classified\ Active}}}$.

3.4.3 Results and Discussion

PREDICTIONS APPLIED TO THE DISJOINT SET

Table 3.5 and Figures 3.7-3.8 show the distribution of success rates for use of the training set threshold $\boldsymbol{s}^*$ in the prediction of finding pairs of similar compounds in the disjoint test set. This may be seen as a measure of how much a structural similarity methodology accurately captures core important information about a specific target, since this information is being applied to an unknown set of compounds on the same target. The data shows that although all methods provided a distribution of success rates significantly better than the absence of any structural information, they did not perform equally well. In particular, on *T. vaginalis* both MACCS and TGD had much lower median success rates and had much fewer highly accurate (success rate > 90%) trials. For *G. intestinalis* MACCS performs much better, while TGD does not. piDAPH3 was the top performing structural fingerprint (based on the number of trials with a > 90% success rate), and property similarity scores performed comparably to piDAPH3 on both targets.

PREDICTIONS APPLIED TO THE COMPARISON BETWEEN THE SETS

Table 3.6 and Figures 3.9-3.12 show the distributions of coverage rates and true positive rates for use of the training set threshold $\boldsymbol{s}^*$ in the classification of active compounds in the test set. In an ideal scenario, the prescribed method would be able to identify a large percentage of the active compounds in the test set, which not falsely identifying too many inactive compounds. As before, different similarity methods performed markedly different, but interestingly performance was not consistent with accuracy in the analysis of the disjoint set. For *T. vaginalis,* Radial and TGD (one of the worst performers when analyzing the disjoint set) found more than half the active compounds in the test set at an accuracy of over 75% in more than 60% of the trials. All methods except Pharm4Pts were able to perform at such a high level of success at least a quarter of the time. Lower overall numbers are attributable to either an inability to identify compounds (a low coverage rate) or to be accurate (a low true positive rate) . For *G. intestinalis*, both coverage and true positive rates reduced markedly overall; this may be partially attributable to the differing thresholds used for *G. intestinalis* versus *T. vaginalis* (on average, there were 8% more active compounds in the test sets of the *T.*

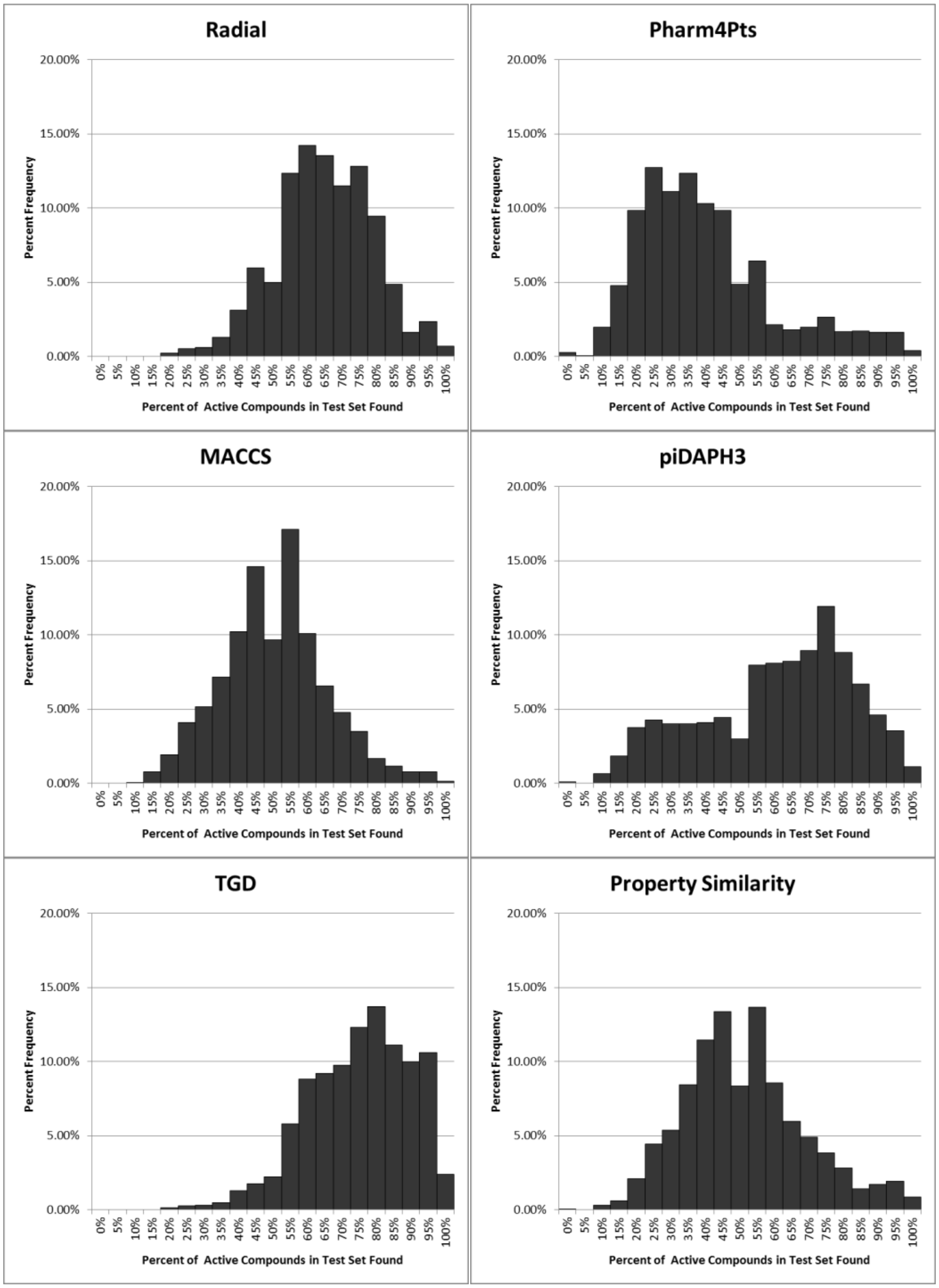


Figure 3.9 Distributions of coverage rates (the percentages of active compounds in the test set which were positively identified) when testing the proposed structural predictivity methodology for T. vaginalis against the set of comparisons between the training and test sets

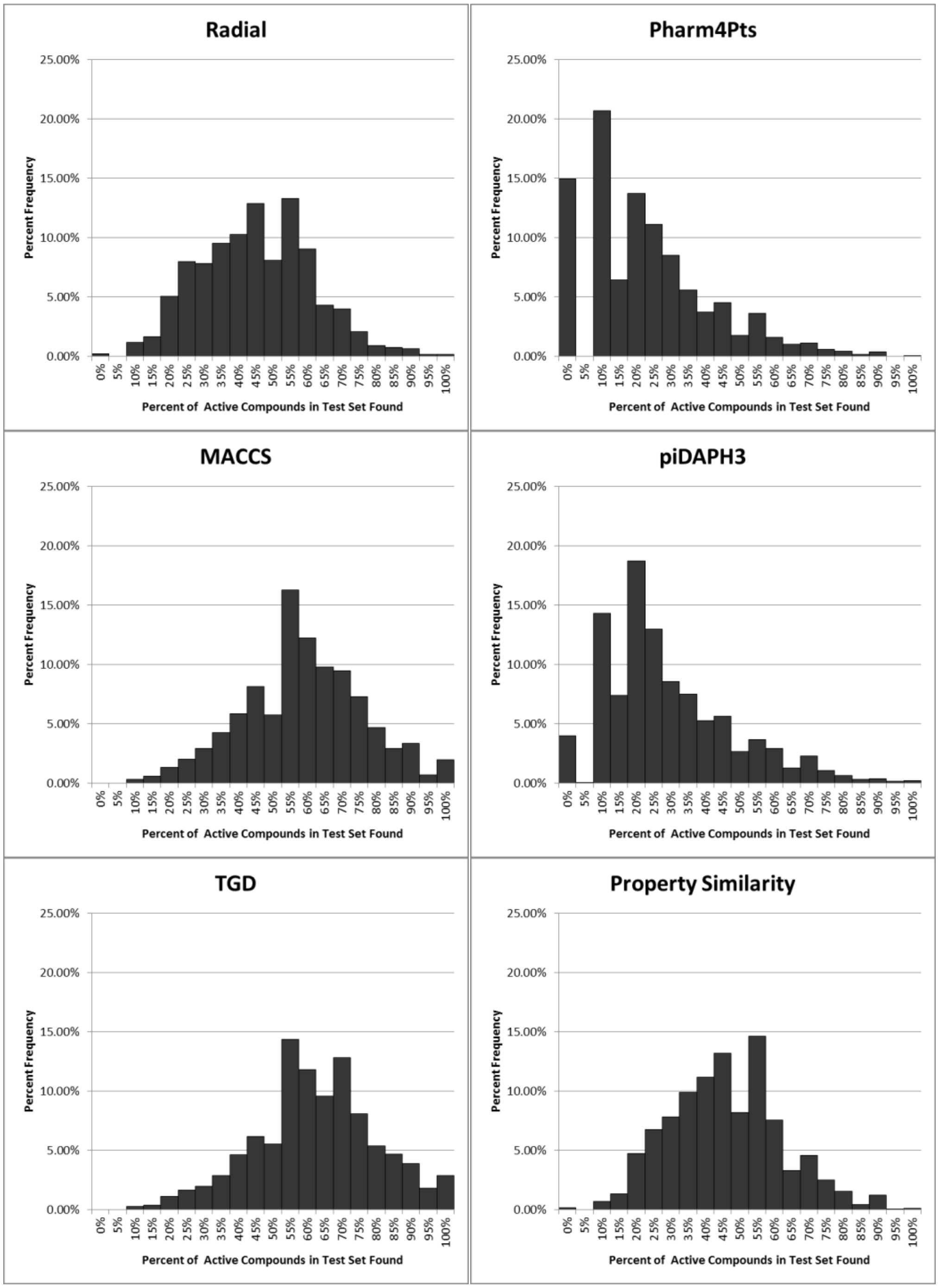


Figure 3.10 Distributions of coverage rates (the percentages of active compounds in the test set which were positively identified) when testing the proposed structural predictivity methodology for G. intestinalis against the set of comparisons between the training and test sets

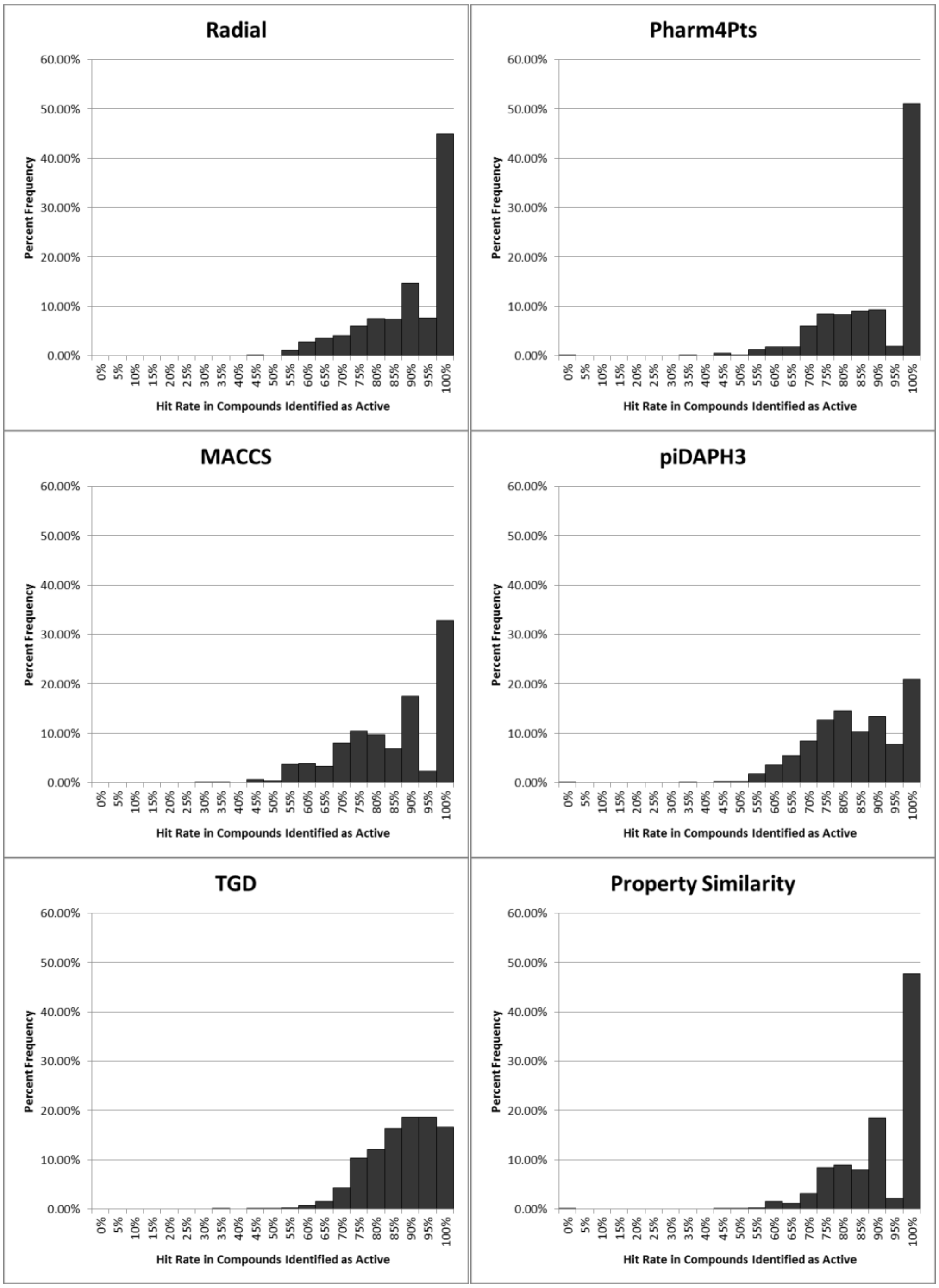


Figure 3.11 Distributions of true positive rates (the percentage of compounds classified as active which were actually active) when testing the proposed structural predictivity methodology for T. vaginalis against the set of comparisons between the training and test sets

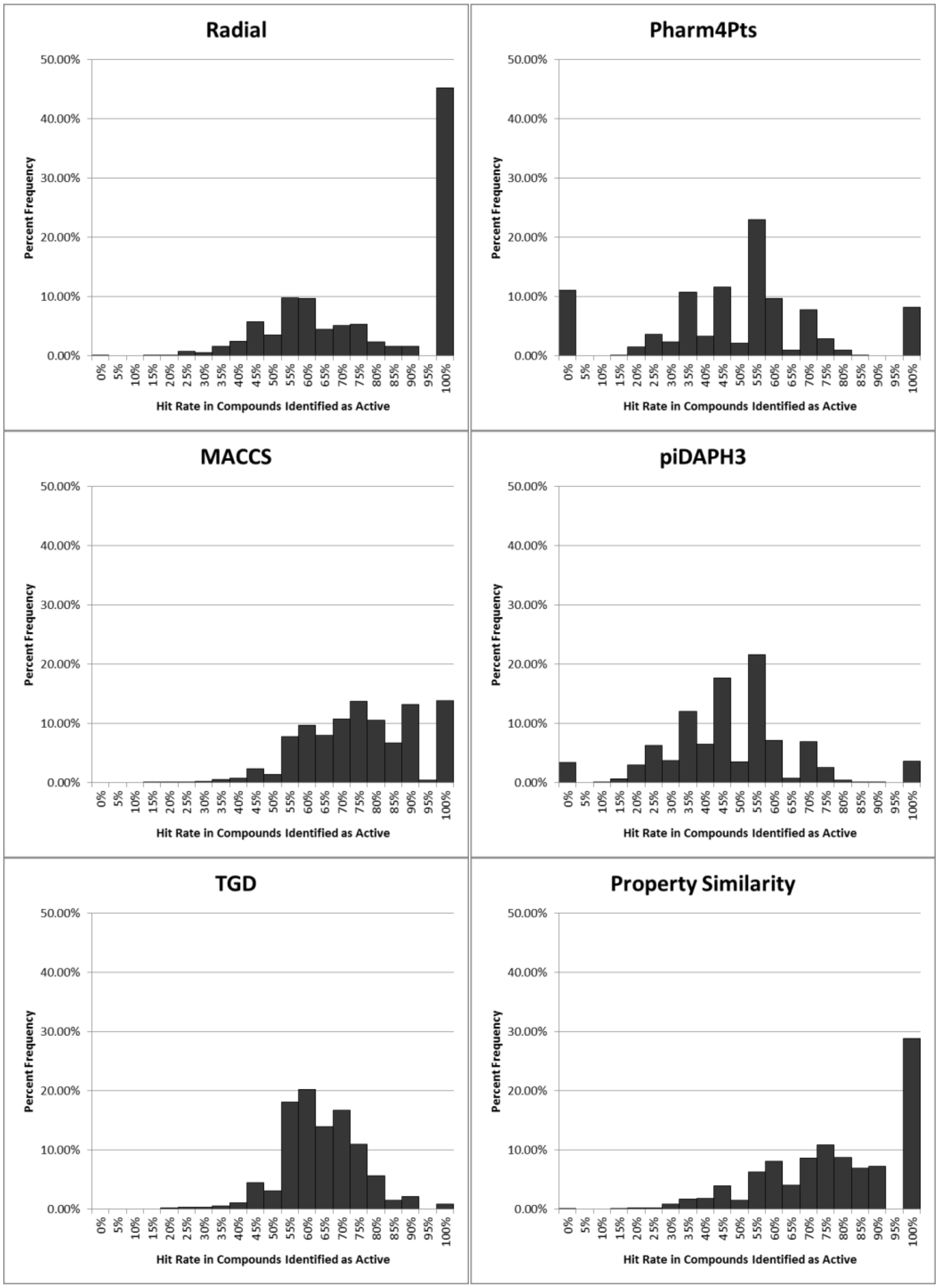


Figure 3.12 Distributions of true positive rates (the percentage of compounds classified as active which were actually active) when testing the proposed structural predictivity methodology for G. intestinalis against the set of comparisons between the training and test sets

vaginalis trials than there were in the test sets of the *G. intestinalis* trials). Only MACCS was able to have a coverage rate greater than 50% and a true positive rate greater than 75% in over a quarter of the trials. For both targets, all methods except Pharm4Pts and piDAPH3 (on *G. intestinalis*) were consistently able to find at least one active compound while having the majority of compounds identified confirmed active.

3.5 Conclusion

In this chapter we have presented several qualitative and quantitative ways of analyzing the SAR of a set of compounds. The mathematical tools involved are straightforward but are capable of creating a robust picture of the chemical space associated with the *T. vaginalis* and *G. intestinalis* targets. Furthermore, by exploring the joint probability distributions of the structural and activity similarities we have developed a method of measuring the ability for these structural representations to describe other compounds within the chemical space and, in particular, to pinpoint which compounds of unknown activities are likely to have activities close to compounds of known activities. This expands the usefulness of these structural similarity methods beyond finding sets of structural diversity and into the realm of pinpointing potential leads. We believe that exploring the SAR representations of the chemical space from the probabilistic standpoint outlined here will yet result in more novel explorations of SAR and the chemical space, and that this will be the subject of much fruitful reseach for years to come.

APPENDIX A

COMMENTED C++ CODE

Below we quote the annotated C++ code for the DARS algorithm, both for the reader's own use and to confirm that the simulations in the numerical results were done in accordance with the text:

```
#include<iostream>
#include<stdlib.h>
#include<math.h>
#include<cmath>
#include<time.h>

#define PI 3.14159265359

//Insert Functional Dimension Here

#define DIM 2

//Insert Function Name Here; Change Input Parameters Twice Below

#define FUNCTIONNAME LinearSystem

//Insert Maximum Function Value Here

#define FUNCTIONMAXVALUE 0

#define IM1 2147483563
#define IM2 2147483399
#define AM (1.0/IM1)
#define IMM1 (IM1-1)
#define IA1 40014
#define IA2 40692
#define IQ1 53668
#define IQ2 52774
#define IR1 12211
#define IR2 3791
```

```
#define NTAB 32
#define NDIV (1+IMM1/NTAB)
#define EPS 1.2e-7
#define RNMX (1.0-EPS)

using namespace std;

//Random Number Generator, courtesy of Numerical
//Recipes in C[50].  Renamed from ran2 to ran0 for
//enumerative convenience within this project.

long double ran0(long *idum)
{
	int j;
	long k;
	static long idum2=123456789;
	static long iy=0;
	static long iv[NTAB];
	long double temp;
	if (*idum <= 0) {
	if (-(*idum) < 1) *idum=1;
	else *idum = -(*idum);
	idum2=(*idum);
	for (j=NTAB+7;j>=0;j--) {
	k=(*idum)/IQ1;
	*idum=IA1*(*idum-k*IQ1)-k*IR1;
	if (*idum < 0) *idum += IM1;
	if (j < NTAB) iv[j] = *idum;
	}
	iy=iv[0];
	}
	k=(*idum)/IQ1;
	*idum=IA1*(*idum-k*IQ1)-k*IR1;
	if (*idum < 0) *idum += IM1;
	k=idum2/IQ2;
	idum2=IA2*(idum2-k*IQ2)-k*IR2;
	if (idum2 < 0) idum2 += IM2;
	j=iy/NDIV;
	iy=iv[j]-idum2;
	iv[j] = *idum;
	if (iy < 1) iy += IMM1;
	if ((temp=AM*iy) > RNMX) return RNMX;
	else return temp;
}
```

```
//These functions are used to ensure that the search area
//does not extend beyond the total search domain

long double centerfind(long double newC, long double newW, long double oldC, long
double oldW)
{
        if ((newC - 0.5*newW) < (oldC - 0.5*oldW))
        {
                return (oldC - 0.5*oldW + 0.5*newW);
        }
        else
        {
                if ((newC + 0.5*newW) > (oldC + 0.5*oldW))
                {
                        return (oldC + 0.5*oldW - 0.5*newW);
                }

                else
                {
                        return newC;
                }
        }
}

long double centeroffset(long double newC, long double newW, long double oldC, long
double oldW)
{
        if ((newC - 0.5*newW) < (oldC - 0.5*oldW))
        {
                return (oldC - 0.5*oldW - newC + 0.5*newW);
        }

        else
        {
                if ((newC + 0.5*newW) > (oldC + 0.5*oldW))
                {
                        return (oldC + 0.5*oldW - newC - 0.5*newW);
                }

                else
                {
                        return 0;
                }
        }
}
```

```
//Standard Min and Max functions for a vector of length DIM

long double MINIMUM(long double *VV)
{
	int k;

	long double m=VV[0];

	for (k=1; k<DIM; k++)
	{
		if (m>VV[k])
		{
			m=VV[k];
		}
	}

	return m;
}

long double MAXIMUM(long double *VV)
{
	int k;

	long double M=VV[0];

	for (k=1; k<DIM; k++)
	{
		if (M<VV[k])
		{
			M=VV[k];
		}
	}

	return M;
}

//Dot Product for two vectors of length DIM

long double DOT(long double *V, long double *W)

{
	int k;
	long double sum=0;
```

```
        for (k=0; k<DIM; k++)
        {
                sum += V[k]*W[k];
        }

        return sum;
}

//The sum of a vector of length L, specified in input

long double SUM(long double *VVV, int L)

{
        int k;

        long double s=0;

        for (k=0; k<L; k++)
        {
                s += VVV[k];
        }

        return s;
}

//The linear combination of vectors UU and VV.  Output RR is by reference

void LINEAR(long double *RR, long double AA, long double *UU, long double BB,
long double *WW)
{
        int k;

        for (k=0; k<DIM; k++)
        {
                RR[k] = AA*UU[k]+BB*WW[k];
        }
}

//The distance from a point Q to line defined by point C and direction vector V

long double LINEDIST(long double *Q, long double *C, long double *V)

{
        long double W[DIM], TEMP[DIM];
        LINEAR(W, 1.0, Q, -1.0, C);

        LINEAR(TEMP, 1.0, W, -1.0*DOT(W, V)/DOT(V, V), V);

        return sqrt(DOT(TEMP, TEMP));
}

//The distance from point Q to plane defined by point P and normal vector V

long double PLANEDIST(long double *Q, long double *C, long double *V)

{
        long double W[DIM], dot, mag;

        LINEAR(W, 1.0, Q, -1.0, C);

        dot = fabs(DOT(V, W));

        mag = sqrt(DOT(V, V));

        return (dot/mag);
}

//Begin main DARS Algorithm

void main()
{
        cout << "\n  ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~  \n";
        cout <<   "  ~ ~ ~ DARS Function Optimizer ~ ~ ~  \n";
        cout <<   "  ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~  \n\n";

        //Definition and Overall Initialization of Variables

        int i, ii, j, jj, k, kk, I, K, fail, small, improve, shrink, maxi, arsi, test;

        //Random Number Generator seeded by time

        long seed=time(NULL);

        long double rx[DIM], x[DIM], mean[DIM], BASIS[DIM][DIM], vector[DIM];
        long double direction[DIM], F, sigma, LAST[DIM][10], planardists[DIM][9],
meandist[DIM];

        long double W[DIM], XC[DIM], INITWIDTH[DIM], INITCENTER[DIM];
```

```
	long double maxx[DIM], maxF;

	FILE *ofp;

	const long double ERRTOL = 1e-15;

	//MAXITTER is the maximum number of function evaluations allowed
	//per trial before termination.

	const int MAXITTER = 10000000;
	const int MAXSMALL = 10000000;
	const int Ntrial = 25;
	long double WIDTHDEC = pow(2,(-(1.0/DIM)));

	const int INITIALIZATIONNUMBER=15;


	long double ACTUALMAX = FUNCTIONMAXVALUE;

	//SIGFIG is the number of significant figures of accuracy
	//away from the actual function value in order to terminate.

	const int SIGFIG=4;

	for (j=0; j<DIM; j++)
	{
		INITWIDTH[j]=20;
		INITCENTER[j]=0;
	}

	//Filename for Output File

	ofp = fopen("DARSFinalResults_FUNCTIONNAME.txt", "w");

//Main For Loop for total number of Ntrial trials

for (I=0; I<Ntrial; I++)
{
	//Initialization for each trial

	i=0;

	for (j=0; j<DIM; j++)
	{
		W[j] = INITWIDTH[j];
```

```
        XC[j] = INITCENTER[j];
        maxx[j] = -99999999;
    }

    maxF=-99999999;

    improve=0;
    shrink=0;
    small=0;

    K=1;

    //The outer While Loop restarts the entire search process
    //if the algorithm shrinks MAXSMALL times without finding
    //a better point.  This helps against conversion to local
    //extrema.

    while ((i<=MAXITTER) && (fabs(1-maxF/ACTUALMAX) > pow(0.1,
SIGFIG)))
    {
        if(small>MAXSMALL)
        {
            improve=0;

            for (j=0; j<DIM; j++)
            {
                W[j] = INITWIDTH[j];

                XC[j] = INITCENTER[j];

                maxx[j] = -99999999;
            }

            maxF=-99999999;

            small=0;
            shrink=0;

            K=1;
        }

        //The alogirthm starts with ARS until INITIALIZATIONNUMBER of
        //improvements have been made
        while ((i<=MAXITTER) && (improve<INITIALIZATIONNUMBER)
&& (fabs(1-maxF/ACTUALMAX) > pow(0.1, SIGFIG)) && (small<=MAXSMALL))

        {
            //When ARS gets too small a search area, it returns to initial size
(finite descent)

            if (MINIMUM(W) < ERRTOL)
            {
                for (j=0; j<DIM; j++)
                {
                    W[j] = INITWIDTH[j];

                    XC[j] = INITCENTER[j];
                }

                shrink=0;
                K=1;

                small++;
            }

            //Generation of Random Point within search region

            for (j=0; j<DIM; j++)
            {
                rx[j] = ran0(&seed);

                x[j] = XC[j] - 0.5*W[j] + W[j]*rx[j];
            }

            //Function evaluation.  Note function inputs must be altered
            //to accommodate individual functtions, hear
            //and in DARS step below

            F = FUNCTIONNAME(x[0],x[1]);


            if (F>maxF)
            {
                //If a better point is found, ARS stores it as its best point
                //and returns to initial search area

                improve++;
                for (j=0; j<DIM; j++)
                {
                    LAST[j][improve % 10]=x[j];
                }

                maxF = F;
                maxi=i;

                for (j=0; j<DIM; j++)
                {
                    maxx[j]=x[j];
                    W[j] = INITWIDTH[j];

                    XC[j] = INITCENTER[j];
                }

                K=1;
                small=0;

                shrink=0;
            }
            else
            {
                //Otherwise, search region shrinks around current best point

                W[0] = W[0]*WIDTHDEC;

                XC[0] = centerfind(maxx[0], W[0], INITCENTER[0],
INITWIDTH[0]);
                for (j=1; j<DIM; j++)
                {
                    W[j] = W[j]*WIDTHDEC;

                    XC[j] = centerfind(maxx[j], W[j], INITCENTER[j],
INITWIDTH[j]);
                }

                shrink++;
            }

            i++;
            arsi=i;
        }
        //True DARS Shrinking can occur once 10 improvements through ARS
happen

        if((i<=MAXITTER) && (fabs(1-maxF/ACTUALMAX) > pow(0.1,
SIGFIG)) && (small<=MAXSMALL))
        {
            //This section occurs only the first time after ARS Initialization

            if (arsi == i)
            {
                //Basis is created using Grahm-Schmitt Process with the
                //vector from the average of the last 10 improvements to the
                //current best point

                for (j=0; j<DIM; j++)
                {
                    mean[j] = 0.1*SUM(LAST[j], 10);
                }

                LINEAR(BASIS[0], 1.0, maxx, -1.0, mean);

                LINEAR(BASIS[0],
1.0/(sqrt(DOT(BASIS[0],BASIS[0]))), BASIS[0], 0, BASIS[0]);

                for (j=1; j<DIM; j++)
                {
                    BASIS[j][0] = -1.0*BASIS[0][j]/BASIS[0][0];

                    for (k=1; k<DIM; k++)
                    {
                        if (k==j)
                        {
                            BASIS[j][k] = 1.0;
                        }
                        else
                        {
                            BASIS[j][k] = 0.0;
                        }
                    }
                }

                for (j=1; j<DIM; j++)
                {
                    for (k=0; k<=j-1; k++)
                    {
                        LINEAR(BASIS[j], 1.0, BASIS[j], -
1.0*DOT(BASIS[j],BASIS[k]), BASIS[k]);
                    }

                    LINEAR(BASIS[j],
1.0/(sqrt(DOT(BASIS[j],BASIS[j]))), BASIS[j], 0, BASIS[0]);
                }

                //Then, distances are calculated from the 9 previous best
                //points to the planes defined by the basis elements. This
                //will be used to determine shrinking ratios

                for (j=0; j<10; j++)
                {
                    for (jj=0; jj<DIM; jj++)
                    {
                        direction[jj] = LAST[jj][j];
                    }

                    if (j < improve%10)
                    {
                        for (kk=0; kk<DIM; kk++)
                        {
                            planardists[kk][j] =
PLANEDIST(direction, maxx, BASIS[kk]);
                        }
                    }
                    else
                    {
                        if (j > improve%10)
                        {
                            for (kk=0; kk<DIM; kk++)
                            {
                                planardists[kk][j-1] =
PLANEDIST(direction, maxx, BASIS[kk]);
                            }
                        }
                    }
                }

                for (jj=0; jj<DIM; jj++)
                {
                    meandist[jj] = (1.0/9.0)*SUM(planardists[jj], 9);
                }
```

```
        //Finally, widths are again initialized

        for (j=0; j<DIM; j++)
        {
                W[j] = INITWIDTH[j];
        }

        K=1;

        shrink=0;
        small=0;
}

//Below occurs each iteration

//As with ARS, if the search region is too small DARS
//returns to full search area

if (MINIMUM(W) < ERRTOL)
{
        for (j=0; j<DIM; j++)
        {
                W[j] = INITWIDTH[j];
        }

        K=1;

        shrink=0;
        small++;
}

//The generation of a random point within the search region
//as defined by the new, "tilted" basis

if (shrink==0)
{
        for (j=0; j<DIM; j++)
        {
                rx[j] = ran0(&seed);

                x[j] = INITCENTER[j] - 0.5*W[j] + W[j]*rx[j];
        }
}
else
{
```

```
                for (j=0; j<DIM; j++)
                {
                    rx[j] = (W[j]*(ran0(&seed)-0.5)-
centeroffset(maxx[j], W[j], INITCENTER[j], INITWIDTH[j]));

                    x[j] = maxx[j];
                }

                for (j=0; j<DIM; j++)
                {
                    LINEAR(x, 1.0, x, rx[j], BASIS[j]);
                }
            }

            for (j=0; j<DIM; j++)
            {
                if (x[j] > INITCENTER[j] + 0.5*INITWIDTH[j])
                {
                    x[j] = INITCENTER[j] + 0.5*INITWIDTH[j];
                }

                if (x[j] < INITCENTER[j] - 0.5*INITWIDTH[j])
                {
                    x[j] = INITCENTER[j] - 0.5*INITWIDTH[j];
                }
            }

            //Function evaluation.  Again, alter input parameters to
            //suit actual function

            F = FUNCTIONNAME(x[0],x[1]);

            if (F>maxF)
            {
                //If a better point is found, DARS stores it as its best point
                //and returns to initial search area

                improve++;

                for (j=0; j<DIM; j++)
                {
                    LAST[j][improve % 10]=x[j];
                }

                maxF = F;
                maxi=i;

                for (j=0; j<DIM; j++)
                {
                    if ( x[j] > INITCENTER[j] + 0.5*INITWIDTH[j])
                    {
                        maxx[j]=x[j] - 10*ERRTOL;
                    }
                    else
                    {
                        if (x[j] < INITCENTER[j] -
0.5*INITWIDTH[j])
                        {
                            maxx[j]=x[j] + 10*ERRTOL;
                        }
                        else
                        {
                            maxx[j]=x[j];
                        }
                    }
                }

                //Then, the basis must be re-created to account for the new
point...

                for (j=0; j<DIM; j++)
                {
                    mean[j] = 0.1*SUM(LAST[j], 10);
                }

                LINEAR(BASIS[0], 1.0, x, -1.0, mean);

                LINEAR(BASIS[0],
1.0/(sqrt(DOT(BASIS[0],BASIS[0]))), BASIS[0], 0, BASIS[0]);

                for (j=1; j<DIM; j++)
                {
                    BASIS[j][0] = -1.0*BASIS[0][j]/BASIS[0][0];

                    for (k=1; k<DIM; k++)
                    {
                        if (k==j)
                        {
                            BASIS[j][k] = 1.0;
                        }
                        else
                        {
                            BASIS[j][k] = 0.0;
                        }
                    }
                }

                for (j=1; j<DIM; j++)
                {
                    for (k=0; k<=j-1; k++)
                    {
                        LINEAR(BASIS[j], 1.0, BASIS[j], -
1.0*DOT(BASIS[j],BASIS[k]), BASIS[k]);
                    }

                    LINEAR(BASIS[j],
1.0/(sqrt(DOT(BASIS[j],BASIS[j]))), BASIS[j], 0, BASIS[0]);
                }

                //... and distances re-calculated

                for (j=0; j<10; j++)
                {
                    for (jj=0; jj<DIM; jj++)
                    {
                        direction[jj] = LAST[jj][j];
                    }

                    if (j < improve%10)
                    {
                        for (kk=0; kk<DIM; kk++)
                        {
                            planardists[kk][j] =
PLANEDIST(direction, maxx, BASIS[kk]);
                        }
                    }
                    else
                    {
                        if (j > improve%10)
                        {
                            for (kk=0; kk<DIM; kk++)
                            {
                                planardists[kk][j-1] =
PLANEDIST(direction, maxx, BASIS[kk]);
                            }
                        }
                    }
                }

                for (jj=0; jj<DIM; jj++)
                {
                    meandist[jj] = (1.0/9.0)*SUM(planardists[jj], 9);
                }

                for (j=0; j<DIM; j++)
                {
                    W[j] = INITWIDTH[j];
                }

                K=1;
                small=0;
                shrink=0;
            }
            else
            {
                //Otherwise, shirinking of the search region occurs.

                //To create the new ratio, dimensions that are "too big"
                //are shrunk down first until the proper shape is created

                test = 0;

                for (j=1; j<DIM; j++)
                {
                    if (W[j]*WIDTHDEC >
(meandist[j]/meandist[0])*MAXIMUM(INITWIDTH))
                    {
                        W[j] = W[j]*WIDTHDEC;

                        test++;
                    }
                }

                //And then all dimensions are shrunk equally

                if (test == 0)
                {
                    for (j=0; j<DIM; j++)
                    {
                        W[j] = W[j]*WIDTHDEC;
                    }
                }

                shrink++;
            }

            i++;
        }
    }

    //Write to file the results of each trial

    fprintf(ofp, "%d\t%d\t", I+1, maxi);

    for (j=0; j<DIM; j++)
    {
        fprintf(ofp, "%.22Lf\t", maxx[j]);
    }
    fprintf(ofp, "%.22Lf\n", maxF);

    cout << "Round:  " << I+1 << "\n";

}

    fclose(ofp);

    cout << "\nPress any key to exit\n\n";

    getchar();
}

//End MAIN DARS ALGORITHM
```

APPENDIX B

BENCHMARK FUNCTIONS

Here we list the benchmark functions used in Chapter 1.

THE DIXON AND SZEGÖ FUNCTIONS[39]

Shekel [S5]:

For $A = \begin{pmatrix} 4 & 1 & 8 & 6 & 3 \\ 4 & 1 & 8 & 6 & 7 \\ 4 & 1 & 8 & 6 & 3 \\ 4 & 1 & 8 & 6 & 7 \end{pmatrix}$ and $C = \begin{pmatrix} 0.1 \\ 0.2 \\ 0.2 \\ 0.4 \\ 0.4 \end{pmatrix}$ then

$$f(x_1, x_2, x_3, x_4) = -\sum_{j=1}^{5} \frac{1}{c_j + \sum_{i=1}^{4} \left(x_i - a_{ij}\right)^2} \ ; \ \ x_i \in [-5,5]$$

Global Minimum Value: -10.1532

Shekel [S7]:

For $A = \begin{pmatrix} 4 & 1 & 8 & 6 & 3 & 2 & 5 \\ 4 & 1 & 8 & 6 & 7 & 9 & 5 \\ 4 & 1 & 8 & 6 & 3 & 2 & 5 \\ 4 & 1 & 8 & 6 & 7 & 9 & 5 \end{pmatrix}$ and $C = \begin{pmatrix} 0.1 \\ 0.2 \\ 0.2 \\ 0.4 \\ 0.4 \\ 0.6 \\ 0.3 \end{pmatrix}$ then

$$f(x_1, x_2, x_3, x_4) = -\sum_{j=1}^{7} \frac{1}{c_j + \sum_{i=1}^{4} \left(x_i - a_{ij}\right)^2} ; \ \ x_i \in [-5,5]$$

Global Minimum Value: -10.4029

Shekel [S10]:

For $A = \begin{pmatrix} 4 & 1 & 8 & 6 & 3 & 2 & 5 & 8 & 6 & 7 \\ 4 & 1 & 8 & 6 & 7 & 9 & 5 & 1 & 2 & 3.6 \\ 4 & 1 & 8 & 6 & 3 & 2 & 5 & 8 & 6 & 7 \\ 4 & 1 & 8 & 6 & 7 & 9 & 5 & 1 & 2 & 3.6 \end{pmatrix}$ and $C = \begin{pmatrix} 0.1 \\ 0.2 \\ 0.2 \\ 0.4 \\ 0.4 \\ 0.6 \\ 0.3 \\ 0.7 \\ 0.5 \\ 0.5 \end{pmatrix}$ then

$$f(x_1, x_2, x_3, x_4) = -\sum_{j=1}^{10} \frac{1}{c_j + \sum_{i=1}^{4} \left(x_i - a_{ij}\right)^2}; \;\; x_i \in [-5,5]$$

Global Minimum Value: -10.5367

Hartman [H3]:

For $= \begin{pmatrix} 3 & 0.1 & 3 & 0.1 \\ 10 & 10 & 10 & 10 \\ 30 & 35 & 30 & 35 \end{pmatrix}$, $P = \begin{pmatrix} 0.36890 & 0.46990 & 0.10910 & 0.03815 \\ 0.11700 & 0.43870 & 0.87320 & 0.57430 \\ 0.26730 & 0.74700 & 0.55470 & 0.88280 \end{pmatrix}$, and $C = \begin{pmatrix} 1 \\ 1.2 \\ 3 \\ 3.2 \end{pmatrix}$ then

$$f(x_1, x_2, x_3) = -\sum_{j=1}^{4} c_j e^{-\sum_{i=1}^{3} a_{ij}(x_i - p_{ij})^2}; \;\; x_i \in [-5,5]$$

Global Minimum Value: -3.8628

Hartman [H6]:

For $= \begin{pmatrix} 10 & 0.05 & 3 & 17 \\ 3 & 10 & 3.5 & 8 \\ 17 & 17 & 1.7 & 0.05 \\ 3.5 & 0.1 & 10 & 10 \\ 1.7 & 8 & 17 & 0.1 \\ 8 & 14 & 8 & 14 \end{pmatrix}$, $P = \begin{pmatrix} 0.1312 & 0.2329 & 0.2348 & 0.4047 \\ 0.1696 & 0.4135 & 0.1451 & 0.8828 \\ 0.5569 & 0.8307 & 0.3522 & 0.8732 \\ 0.0124 & 0.3736 & 0.2883 & 0.5743 \\ 0.8283 & 0.1004 & 0.3047 & 0.1091 \\ 0.5886 & 0.9991 & 0.6650 & 0.0381 \end{pmatrix}$, and $C = \begin{pmatrix} 1 \\ 1.2 \\ 3 \\ 3.2 \end{pmatrix}$ then

$$f(x_1, x_2, x_3, x_4, x_5, x_6) = -\sum_{j=1}^{4} c_j e^{-\sum_{i=1}^{6} a_{ij}(x_i - p_{ij})^2}\ ;\ \ x_i \in [-5{,}5]$$

Global Minimum Value: -3.3224

Goldstein-Price [GP]:

$$f(x,y) = \left(1 + (x+y+1)^2(19 - 14x + 3x^2 - 14y + 6xy + 3y^2)\right)$$

$$\cdot \left(30 + (2x - 3y)^2(18 - 32x + 12x^2 + 48y - 36xy + 27y^2)\right);\ \ x, y \in [-5{,}5]$$

Global Minimum Value: 3.0000

Branin [BR]:

$$f(x,y) = \left(y - \tfrac{5.1}{4\pi^2}x^2 + \tfrac{5}{\pi}x - 6\right)^2 + 10\left(1 - \tfrac{1}{8\pi}\right) cos\, x + 10;\ \ x, y \in [-5{,}5]$$

Global Minimum Value: 0.3979

Six Hump Camel [C6]:

$$f(x,y) = 4x^2 - 2.1x^4 + \tfrac{1}{3}x^6 + xy - 4y^2 + 4y^4;\ \ x, y \in [-5{,}5]$$

Global Minimum Value: -1.0316

Shubert [SHU]:

$$f(x,y) = \left(\sum_{n=1}^{5} n\, cos\big((n+1)x + n\big)\right)\left(\sum_{n=1}^{5} n\, cos\big((n+1)y + n\big)\right);\ \ x, y \in [-5{,}5]$$

Global Minimum Value: -186.7309

THE ARS TEST FUNCTIONS[1]

Freudenstein-Roth [FR]:

$$f(x,y) = \Big((x - 13) + \big((5 - y)y - 2y\big)\Big)^2\ ;\ \ x, y \in [-10{,}10]$$

Global Minimum Value: 0.000

Gaussian Function 1 [G1]:

For $h = \binom{5}{1}$, $s = \binom{10}{0.5}$, and $m = \begin{pmatrix} 0 & 0 \\ 0 & 5 \\ 0 & -5 \\ 5 & 0 \\ -5 & 0 \end{pmatrix}$, let

$g(x, y; h, s, m_i) = (h - (x - m_{i1})^2 - (y - m_{i2})^2)e^{-s(x-m_{i1})^2-s(y-m_{i2})^2}$. Then

$$f(x, y) = g(x, y; h_1, s_1, m_1) + g(x, y; h_2, s_2, m_2) + g(x, y; h_2, s_2, m_3) + g(x, y; h_2, s_2, m_4) + g(x, y; h_2, s_2, m_5)\ ;\ x, y \in [-10,10]$$

Global Maximum Value: 5.000

Gaussian Function 2 [G2]:

For $h = \binom{5}{1}$, $s = \binom{100}{0.5}$, and $m = \begin{pmatrix} 0 & 0 \\ 0 & 5 \\ 0 & -5 \\ 5 & 0 \\ -5 & 0 \end{pmatrix}$, let

$g(x, y; h, s, m_i) = (h - (x - m_{i1})^2 - (y - m_{i2})^2)\mathrm{e}^{-\mathrm{s}(\mathrm{x}-\mathrm{m}_{\mathrm{i}1})^2-\mathrm{s}(\mathrm{y}-\mathrm{m}_{\mathrm{i}2})^2}$. Then

$$\mathrm{f}(\mathrm{x}, \mathrm{y}) = \mathrm{g}(\mathrm{x}, \mathrm{y}; \mathrm{h}_1, \mathrm{s}_1, \mathrm{m}_1) + \mathrm{g}(\mathrm{x}, \mathrm{y}; \mathrm{h}_2, \mathrm{s}_2, \mathrm{m}_2) + \mathrm{g}(\mathrm{x}, \mathrm{y}; \mathrm{h}_2, \mathrm{s}_2, \mathrm{m}_3) + \mathrm{g}(\mathrm{x}, \mathrm{y}; \mathrm{h}_2, \mathrm{s}_2, \mathrm{m}_4) + \mathrm{g}(\mathrm{x}, \mathrm{y}; \mathrm{h}_2, \mathrm{s}_2, \mathrm{m}_5)\ ;\ \mathrm{x}, \mathrm{y} \in [-10,10]$$

Global Maximum Value: 5.000

Griewank’s Function [Gw]:

$$\mathrm{f}(\mathrm{x}, \mathrm{y}) = 1 + \frac{\mathrm{x}^2}{10} + \frac{\mathrm{y}^2}{10} - \cos(\mathrm{x})\cos\left(\frac{\mathrm{y}}{\sqrt{2}}\right);\ \ \mathrm{x}, \mathrm{y} \in (-100,100)$$

Global Minimum Value: 0.000

Himmelblau’s Function [Him]:

$$\mathrm{f}(\mathrm{x}, \mathrm{y}) = (\mathrm{x}^2 + \mathrm{y} - 11)^2 + (\mathrm{x} + \mathrm{y}^2 - 7)^2;\ \ \mathrm{x}, \mathrm{y} \in [-5,5]$$

Global Minimum Value: 0.000

Jennrich-Sampson [JS]:

$$\mathrm{f}(\mathrm{x},\mathrm{y}) = \sum_{\mathrm{k}=1}^{10}\left(2+2\mathrm{k}-\mathrm{e}^{\mathrm{kx}}-\mathrm{e}^{\mathrm{ky}}\right);\ \ \mathrm{x},\mathrm{y}\in(-1,1)$$

Global Minimum Value: 124.362

Rastrigin's Function [Rast]:

$$\mathrm{f}(\mathrm{x},\mathrm{y}) = \mathrm{x}^2+\mathrm{y}^2-\cos(18\mathrm{x})-\cos(18\mathrm{y});\ \ \mathrm{x},\mathrm{y}\in[-1,1]$$

Global Minimum Value: -2.000

Rosenbrock's "Banana" Function [Ros]:

$$\mathrm{f}(\mathrm{x},\mathrm{y}) = 100(\mathrm{y}-\mathrm{x}^2)^2+(1-\mathrm{x})^2;\ \ \mathrm{x},\mathrm{y}\in[-5,5]$$

Global Minimum Value: 0.000

REFERENCES

[1] Appel, M. J.; LaBarre, R.; Radulovic, D. (2004) On Accelerated Random Search *SIAM J. Optim.*, 14, 708- 731.

[2] Radulović, D. (2010) Pure Random Search with exponential rate of convergency. *Optim.*, 59, 289-303.

[3] Patel, N. R.; Smith, R.; Zabinsky, Z. B. (1988) Pure adaptive search in Monte Carlo optimization. *Math. Prog.*, 43, 317-328.

[4] Esquivel, M. L. (2006) A conditional Gaussian martingale algorithm for global optimization. *Lect. Notes Comp. Sci.*, 3982, 813-823.

[5] Closas, P; Pernandez-Prades, C; Fernandez-Rubio, J. A. (2009) Cramer-Rao bound analysis of position approaches in GNSS receivers. *IEEE Trans. Sig. Proc.*, 57, 3775-3786.

[6] Ouyang, Y.; Ye, F.; Liang, Y. (2009) A modified electronegativity equalization method for fast and accurate calculation of atomic charges in large biological molecules. *Phys. Chem. Chem. Phys.*, 11, 6082-6089.

[7] Rubinstein, R. (1999) The cross-entropy method for combinatorial and continuous optimization. *Meth. Comp. Appl. Prob.*, 1, 127-190.

[8] Fukushima, M. (1992) Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems. *Math. Prog.*, 53, 99-110.

[9] Cuthrell, J. E.; Biegler, L. T. (1987) On the optimization of differential-algebraic process systems. *Amer. Inst. Chem. Eng. J.*, 33, 1257-1270.

[10] Muller, P.; Parmigiani, G. (1995) Optimal design via curve fitting of Monte Carlo experiments. *J. Amer. Stat. Assoc.*, 90, 1322-1331.

[11] Ong, C. J.; Wong, Y. S.; Loh, H. T.; Hong, X. G. (1996) An optimization approach for biarc curve-fitting of B-spline curves. *Comp.-Aid. Des.*, 28 951-959.

[12] Narendra, K. S.; Parthasarathy, K. (1991) Gradient methods for the optimization of dynamical systems containing neural networks. *IEEE Trans. Neur. Net.*, 2, 252-262.

[13] Yao, X. (1999) Evolving artificial neural networks. *Proc. IEEE*, 87, 1423-1447.

[14] Reader, A. J. (2008) The promise of new PET image reconstruction. *Phys. Med.*, 24, 49-56.

[15] Beaulieu, J.; Goldberg, M. (1989) Hierarchy in picture segmentation: A stepwise optimization approach. *IEEE Trans. Pat. Anal. Mach. Int.*, 11, 150-163.

[16] Wolpert, D. H.; Macready, W. G. (1997) No free lunch theorems for optimization. *IEEE Trans. Evol. Comp.*, 1, 67-82.

[17] Streeter, M. (2003) Two broad classes of functions for which a no free lunch result does not hold. *Gen. Evol Comp. GECCO*, 2003, 1418–1430.

[18] Lloyd, S. (2002) Computational capacity of the universe. *Phys. Rev. Lett.*, 88, 237901-237904.

[19] Metropolis, N.; Rosenbluth, A; Rosenbluth, M.; Teller, A; Teller, E. (1953) Equations of state calculations by fast computing machines. *J. Chem. Phys.*, 21, 1087-1091.

[20] Baba, N.; Shoman, T.; Sawaragi, Y. (1977) A modified convergence theorem for a random optimization algorithm. *Info. Sci.*, 13, 159-166.

[21] Gaviano, M. Some general results on the convergence of random search algorithms. In *Toward global optimization*. Dixon, L.; Szego, G. Eds.; North-Holland: Amsterdam, 1975.

[22] Brunelli, R.; Tecchiolli, G. P. (1995) Stochastic minimization with adaptive memory. *J. Comp. Appl. Math.*, 57, 329-343.

[23] Tang, Z. B. (1998) Optimal sequential sampling policy of partitioned random search and its applications. *J. Optim. Theor. Appl.*, 2, 431-448.

[24] Birbil, I.; Fang, S. (2003) An electromagnetism-like mechanism for global optimization. *J. Glob. Optim.*, 25, 263-282.

[25] Solis, F. J.; Wets, R. (1981) Minimization by random search techniques. *Math. Oper. Res.*, 6, 19-30.

[26] Weise, T. *Global Optimization Algorithms: Theory and Applications*. 2009 http://www.it-weise.de/

[27] Dennis, J. E., Jr.; More, J. J. (1977) Quasi-Newton methods, motivation, and theory. *SIAM Rev.*, 19, 46-89.

[28] Rosenbrock, H. H. (1960) An automatic method for finding the greatest or least value of a function. *Comp. J.*, 3, 175-184.

[29] Bazaraa, M. S.; Sherali, H. D.; Shetty, C. M. *Nonlinear programming: theory and algorithms*. Wiley: New York, 1993.

[30] Powell, M. (1978) A fast algorithm for nonlinearly constrained optimization calculations. *Num. Anal.*, 144-157.

[31] Dorigo, C. M.; Maniezzo, V. Distributed optimization by ant colonies. In *Actes de la première conférence européenne sur la vie artificielle*. Elsevier: Paris, 1991, 134-142.

[32] *Kennedy,* J.; Eberhart, R. (1995) Particle swarm o*ptimization. Proc. IEEE Int. Conf. Neur. Net. IV, 1942–1948.*

[33] *S*hi, Y.; Eberhart, R.C. (1998) *A mo*dified particle swarm optimizer. *Proc. IEEE Int. Conf. Evol. Comp., 69–73.*

[34] Poli, R. *An analysis of public*ations on particle swarm optimiz*ation applications.* In *Technical Report CSM-469*. Univ. *Essex*: Essex, 2007.

[35] Nelder, J. A.; Mead, R. (1965) A simplex method for function minimization. Comp. J., 7, 308–313.

[36] MATLAB^{TM} Student Version 5.3.0, Copyright 1984-1999 The MathWorks, Inc.

[37] Back, T. *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. Oxford Univ. Press.: Oxford, 1996.

[38] Marino, I. P.; Miguez, J. (2007) Monte Carlo method for multiparameter estimation in coupled chaotic systems. *Phys. Rev. E,* 76, 057203.

[39] Dixon, L. C. W.; Szego, G. P. The global optimization problem: An introduction. In *Towards global optimization 2*. North-Holland: Amsterdam, 1978, 1-15.

[40] Huyer, W.; Neumaier, A. (1999) Global optimization by multilevel coordinate search. *J. Glob. Optim.*, 14, 331-355.

[41] Reklaitis, G. V.; Ravindrin, A.; Ragsdell, K. M. *Engineering OptimizationMethods*. Wiley: New York, 1983.

[42] Freudenstein, F.; Roth, B. (1963) Numerical solutions of systems of nonlinear equations. *J. ACM*, 10, 550-556.

[43] Jennrich, R. I.; Sampson, P. F. (1968) Application of stepwise regression to nonlinear estimation. *Technometrics*, 10, 63-72.

[44] Simoes, A.; Costa, E. Using genetic algorithms with sexual and asexual transposition: A comparative study. In *Proceedings of the IEEE congress on evolutionary computation*. IEEE Press: San Diego, 2000.

[45] Rastrigin, L. A. *Systems of extremal control*. Nauka: Moscow, 1974.

[46] Greenbaum, A. *Iterative methods for solving linear systems*. SIAM: Philadelphia, 1997.

[47] Hestenes, M. R.; Stiefel, E. (1952) Methods of conjugate gradients for solving linear systems. *J. Res. Nat'l. Bur. Stand.*, 49, 409-436.

[48] Saad, Y.; Schultz, M. H. (1986) GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comp.*, 7, 856-869.

[49] Foss, E. J.; Radulovic, D.; Schaffer, S. A.; Ruderfer, D. M.; Bedalov, A; Goodlett, D. R.; Kruglyak, L. (2007) Genetic basis of proteome variation in yeast. *Nature Gen.*, 39, 1369-1375.

[50] Press, W. H.; Teukolsky, S. A.; Vetterling, W. T.; Flannery, B. P. *Numerical recipes in C: The art of scientific computing*. Cambridge Univ. Press: Cambridge, 2002.

[51] Pérez-Villanueva, J.; Santos, R.; Hernández-Campos, A.; Giulianotti, M. A.; Castillo, R.; Medina-Franco, J. L. (2010) Towards a systematic characterization

of the antiprotozoal activity landscape of benzimidazole derivatives. *Bioorg. Med. Chem.*, 18, 7380-7391.

[52] Pérez-Villanueva, J.; Santos, R.; Hernández-Campos, A.; Giulianotti, M. A.; Castillo, R.; Medina-Franco, J. L. (2011) Structure–activity relationships of benzimidazole derivatives as antiparasitic agents: Dual activity-difference (DAD) maps. *Med. Chem. Comm.*, 2, 44-49.

[53] Santos, R. G.; Giulianotti, M. A.; Dooley, C. T.; Pinilla, C.; Appel, J. R.; Houghten, R. A. (2011) Use and Implications of the Harmonic Mean Model on Mixtures for Basic Research and Drug Discovery *ACS Comb. Sci.*, 13 , 337–344.

[54] Yongye, A. B.; Byler, K.; Santos, R.; Martínez-Mayorga, K.; Maggiora, G. M.; Medina-Franco, J. L. (2011) Consensus Models of Activity Landscapes with Multiple Chemical, Conformer, and Property Representations. *J. Chem. Inf. Model.*, 51, 1259–1270.

[55] Martínez-Mayorga, K.; Peppard, T. L.; Yongye, A. B.; Santos, R.; Giulianotti, M.; Medina-Franco, J. L. (2011) Characterization of a comprehensive flavor database. *J. Chemometrics*, 25, 550–560.

[56] Judkowski, V.; Bunying, A.; Ge, F.; Appel, J. R.; Law, K.; Sharma, A.; Raja-Gabaglia, C.; Norori, P.; Santos, R. G.; Giulianotti, M. A.; Slifka, M. K.; Douek, D. C.; Graham, B. S.; Pinilla, C. (2011) GM-CSF Production Allows the Identification of Immunoprevalent Antigens Recognized by Human CD4+ T Cells Following Smallpox Vaccination. *PLoS ONE*, 6, e24091. doi:10.1371/journal.pone.0024091

[57] Newman, D. J.; Cragg, G. M.; Snader, K. M. The influence of natural products upon drug discovery. (2000) *Nat. Prod. Rep.*, 17, 215-234.

[58] Furka, A. (2002) Combinatorial chemistry: 20 years on… *Drug Discovery Today*, 7, 1-4.

[59] Nagasaki, H. (2009) The pharmacological properties of novel MCH1 receptor antagonist isolated from combintatorial libraries. *Eur. J. Pharmacol*. 602, 194-202;

[60] Houghten, R.A.; Pinilla C.; Giulianotti M. A.; Appel J. R.; Dooley C. T.; Nefzi A.; Ostresh J. M.; Yu Y.; Maggiora G. M.; Medina-Franco J. L.;, Brunner D. (2008) Schneider J. Strategies for the use of mixture-based synthetic combinatorial libraries: Scaffold ranking, direct testing, in vivo, and enhanced deconvolution by computational methods. *J. Comb. Chem.*, 10, 3-19

[61] Kainkaryam, R. M.; Woolf, P. J. (2009) Pooling in high-throughput drug screening. *Cur. Opin. Drug Discov. Devel.*, 12, 339-350.

[62] Wolf, K. K.; Vora S.; Webster L. O.; Generaux G. T.; Polli J. W.; Brouwer K. L. (2010) Use of cassette dosing in sandwich-cultured rat and human hepatocytes to identify drugs that inhibit bile acid transport. *Toxicol. in Vitro*, 24, 297-309;

[63] Smith, N. F.; Raynaud, F. I.; Workman, P. (2007) The application of cassette dosing for pharmacokinetic screening in small-molecule cancer drug discovery. *Mol. Cancer Ther.*, 6, 428-440.

[64] Finney, D. J. *Probit Analysis, 3rd Edition;* Cambridge University Press: Cambridge, 1971.

[65] Smyth, H. F., Jr.; Weil, C. S.; Carpenter, C. P. (1969) An exploration of joint toxic action: twenty-seven industrial chemicals intubated in rats in all possible pairs. *Toxicol. Appl. Pharmacol.*, 14, 340-347.

[66] Hoel, D. G. Statistical Aspects of Chemical Mixtures. In *Methods for Assessing the Effects of Mixtures of Chemicals*; Vouk, V. B.; Butler, G. C.; Upton, A. C.; Parke, D. V.; Asher, S. C. Eds.; Wiley: New York, 1987; pp 369-377.

[67] Talalay, P.; Chou, T. Quantitative analysis of dose-effect relationships: The combined effects of multiple drugs or enzyme inhibitors. In *Advances in Enzyme Regulation*; Webber, G. Ed.; Pergamon: New York, 1984; Vol. 22, pp 27-55.

[68] Feng, B. Y.; Shiochet, B. K. (2006) Synergy and antagonism of promiscuous inhibition in multiple-compound mixtures. *J. Med. Chem.*, 49, 2151-2154.

[69] Frier, S. M.; Konings, D. A. M.; Wyatt, J. R.; Ecker, D. J. (1995) Deconvolution of combinatorial libraries for drug discovery: a model system. *J. Med. Chem.*, 38, 344-352.

[70] Houghten, R. A.; Pinilla, C.; Blondelle, S. E.; Appel, J. R.; Dooley, C. T.; Cuervo, J. H. (1991) Generation and use of synthetic peptide combinatorial libraries for basic research and drug discovery. *Nature*, 354, 84-86.

[71] Dooley, C. T.; Chung, N. N.; Wilkes, B. C.; Schiller, P. W.; Bidlack, J. M.; Pasternak, G. W.; Houghten, R. A. (1994) An all D-amino acid opioid peptide with central anagesic activity from a combinatorial library. *Science*, 266, 2019-2022.

[72] Dooley, C. T.; Chung, N. N.; Schiller, P. W.; Houghten, R. A. (1993) Acetalins: Opioid receptor antagonists determined through the use of synthetic peptide combinatorial libraries. *Proc. Natl. Acad. Sci U.S.A.*, 90, 10811-10815.

[73] Pinilla, C.; Appel, J. R.; Houghten, R. A. (1993) Synthetic peptide combinatorial libraries (SPCLs): identification of the antigenic determinant of β-endophin recognized by monoclonal antibody 3E7. *Gene*, 128, 71-76.

[74] Houghten, R. A.; Appel, J. R.; Blondelle, S. E.; Cuervo, C. T.; Dooley, C. T.; Pinilla, C. (1992) The use of synthetic peptide combinatorial libraries for the identification of bioactive peptides. *Biotechniques*, 13, 412-421.

[75] Appel, J. R.; Pinilla, C.; Houghten, R. A. (1992) Identification of related peptides recognized by a monoclonal antibody using a synthetic peptide combinatorial library. *Immunomethods*, 1, 17-23.

[76] Davis, P. W.; Vickers, T. A.; Wilson-Lingardo, L.; Wyatt, J. R.; Guinosso, C. J.; Sanghvi, Y. S.; DeBaets, E. A.; Acevedo, O. L.; Cook, P. D.; Ecker, D. J. (1995) Drug leads from combinatorial phosphodiester libraries. *J. Med. Chem.*, 38, 4363-4366.

[77] Ecker, D. J.; Vickers, T. A.; Hanecak, R.; Driver, V.; Anderson, K. (1993) Rational screening of oligonucleotide combinatorial libraries for drug discovery. *Nucleic Acids Res.*, 21, 1853-1856.

[78] Li, J. W. H.; Vederas, J. C. (2009) Drug Discovery and Natural Products: End of an Era or an Endless Frontier? *Science*, 325, 161-165.

[79] Butler, M.S. (2004) The role of natural product chemistry in drug discovery. *J. Nat. Prod.*, 67, 2141-2153.

[80] Cantrell, C. L.; Richheimer, S. L.; Nicholas, G. M.; Schmidt, B. K.; Bailey, D. T. (2005) seco-Hinokiol, a new abietane diterpenoid from Rosmarines officinalis. *J. Nat. Prod.,* 68, 98-100.

[81] Nicholas, G. M.; Molinski, T. F. (2000) Enantiodivergent biosynthesis of the dimeric sphingolipid oceanapiside from the marine sponge Oceanapia phillipensis. Determination of remote stereochemistry. *J. Am. Chem. Soc.*, 122, 4011-4019.

[82] Marquez, B.; Verdier-Pinard, P.; Hamel, E.; Gerwick, W. H. (1998) Curacin D: an antimitotic agent from the marine cyanobacterium Lyngbya majuscule. *Phytochemistry*, 49, 2387-2389.

[83] Gerard, J.; Lloyd, R.; Barsby, T.; Haden, P.; Kelly, M. T.; Andersen, R. J. (1997) Massetolides A-H, antimycrobacterial cyclic depsipeptides produced by two pseudomonads isolated from marine habitats. *J. Nat. Prod.*, 60, 223-229.

[84] Terrett, N. K.; Gardner, M.; Gordon, D. W.; Kobylecki, R. J.; Steele, J. (1995) Combinatorial synthesis – The design of compound libraries and their application to drug discovery. *Tetrahedron*, 51, 8135-8173.

[85] Uppsala University Hospital ("Laborationslista"). Artnr 40284 Sj74a. Issued on April 22, 2008

[86] Hill, A. V. (1910). The possible effects of the aggregation of the molecules of hæmoglobin on its dissociation curves. *J. Physiol.*, 40, iv-vii.

[87] Pinilla, C.; Appel, J. R.; Houghten, R. A. Tea bag synthesis of positional scanning synthetic combinatorial libraries and their use for mapping antigenic determinants. In: Methods In Molecular Biology: Epitope Mapping Protocols (Morris, G. E., Eds.), Humana Press Inc., Totowa, NJ, pp. 171-179, 1996.

[88] World Health Organization. Global prevalence and incidence of selected curable sexually transmitted infections: overview and estimates, Geneva, 2001.

[89] Upcroft, P.; Upcroft, J. A. Clin. Microbiol. Rev. 2001, 14, 150.

[90] Berkman, D. S.; Lescano, A. G.; Gilman, R. H.; Lopez, S.; Black, M. M. Lancet 2002, 359, 564.

[91] Navarrete-Vazquez, G.; Rojano-Vilchis, M. D.; Yepez-Mulia, L.; Melendez, V.; Gerena, L.; Hernandez-Campos, A.; Castillo, R.; Hernandez-Luis, F. European Journal of Medicinal Chemistry 2006, 41, 135.

[92] Navarrete-Vázquez, G.; Yépez, L.; Hernández-Campos, A.; Tapia, A.; Hernández-Luis, F.; Cedillo, R.; González, J.; Martínez-Fernández, A.; Martínez-Grueiro, M.; Castillo, R. Bioorganic & Medicinal Chemistry 2003, 11, 4615.

[93] Andrzejewska, M.; Yépez-Mulia, L.; Cedillo-Rivera, R.; Tapia, A.; Vilpo, L.; Vilpo, J.; Kazimierczuk, Z. European Journal of Medicinal Chemistry 2002, 37, 973.

[94] Valdez-Padilla, D.; Rodríguez-Morales, S.; Hernández-Campos, A.; Hernández-Luis, F.; Yépez-Mulia, L.; Tapia-Contreras, A.; Castillo, R. Bioorganic & Medicinal Chemistry 2009, 17, 1724.

[95] Ooms, F. Curr. Med. Chem. 2000, 7, 141.

[96] Medina-Franco, J. L.; Lopez-Vallejo, F.; Castillo, R. Educación Química 2006, 17, 114.

[97] Kubinyi, H. Drug Discovery Today 1997, 2, 457.

[98] Kubinyi, H. Drug Discovery Today 1997, 2, 538.

[99] Scior, T.; Medina-Franco, J. L.; Do, Q. T.; Martinez-Mayorga, K.; Yunes Rojas, J. A.; Bernard, P. Current Medicinal Chemistry 2009, 16, 4297.

[100] Klebe, G. Journal of Molecular Medicine 2000, 78, 269.

[101] Peltason, L.; Bajorath, J. Chem. Biol. 2007, 14, 489.

[102] Maggiora, G. M. J. Chem. Inf. Model. 2006, 46, 1535.

[103] Guha, R.; VanDrie, J. H. J. Chem. Inf. Model. 2008, 48, 646.

[104] Bajorath, J.; Peltason, L.; Wawer, M.; Guha, R.; Lajiness, M. S.; Van Drie, J. H. Drug Discovery Today 2009, 14, 698.

[105] Schneider, G.; Neidhart, W.; Giller, T.; Schmid, G. Angew. Chem., Int. Ed. 1999, 38, 2894.

[106] Eckert, H.; Bajorath, J. Drug Discov. Today 2007, 12, 225.

[107] Medina-Franco, J. L.; Martinez-Mayorga, K.; Giulianotti, M. A.; Houghten, R. A.; Pinilla, C. Curr. Comput.-Aided Drug Des. 2008, 4, 322.

[108] Peltason, L.; Bajorath, J. J. Med. Chem. 2007, 50, 5571.

[109] Guha, R.; Van Drie, J. H. J. Chem. Inf. Model. 2008, 48, 1716.

[110] Shanmugasundaram, V.; Maggiora, G. M. In 222nd ACS National Meeting, Chicago, IL, United States; American Chemical Society, Washington, D. C: Chicago, IL, United States, 2001.

[111] Wawer, M.; Peltason, L.; Weskamp, N.; Teckentrup, A.; Bajorath, J. J. Med. Chem. 2008, 51, 6075.

[112] Peltason, L.; Hu, Y.; Bajorath, J. ChemMedChem 2009, 4, 1864.

[113] Medina-Franco, J. L.; Martínez-Mayorga, K.; Bender, A.; Marín, R. M.; Giulianotti, M. A.; Pinilla, C.; Houghten, R. A. J. Chem. Inf. Model. 2009, 49, 477.

[114] Peltason, L.; Iyer, P.; Bajorath, J. J. Chem Inf. Model. 2010, 50, 1021.

[115] Navarrete-Vázquez, G.; Cedillo, R.; Hernández-Campos, A.; Yépez, L.; Hernández-Luis, F.; Valdez, J.; Morales, R.; Cortés, R.; Hernández, M.; Castillo, R. Bioorg. Med. Chem. Lett. 2001, 11, 187.

[116] Valdez, J.; Cedillo, R.; Hernandez-Campos, A.; Yepez, L.; Hernandez-Luis, F.; Navarrete-Vazquez, G.; Tapia, A.; Cortes, R.; Hernandez, M.; Castillo, R. Bioorg. Med. Chem. Lett. 2002, 12, 2221.

[117] Jaccard, P. Bull. Soc. Vaudoise Sci. Nat. 1901, 37, 547.

[118] Willett, P.; Barnard, J. M.; Downs, G. M. J. Chem. Inf. Comput. Sci. 1998, 38, 983.

[119] MOE, p Molecular Operating Environment (MOE).

[120] Sastry, M.; Lowrie, J. F.; Dixon, S. L.; Sherman, W. J. Chem Inf. Model. 2010, 50, 771.

[121] Willett, P. Drug Discovery Today 2006, 11, 1046.

[122] Rogers, D.; Hahn, M. J. Chem Inf. Model. 2010, 50, 742.

[123] Johnson, M. A.; Maggiora, G. M. Concepts and Applications of Molecular Similarity; Wiley: New York, 1990.