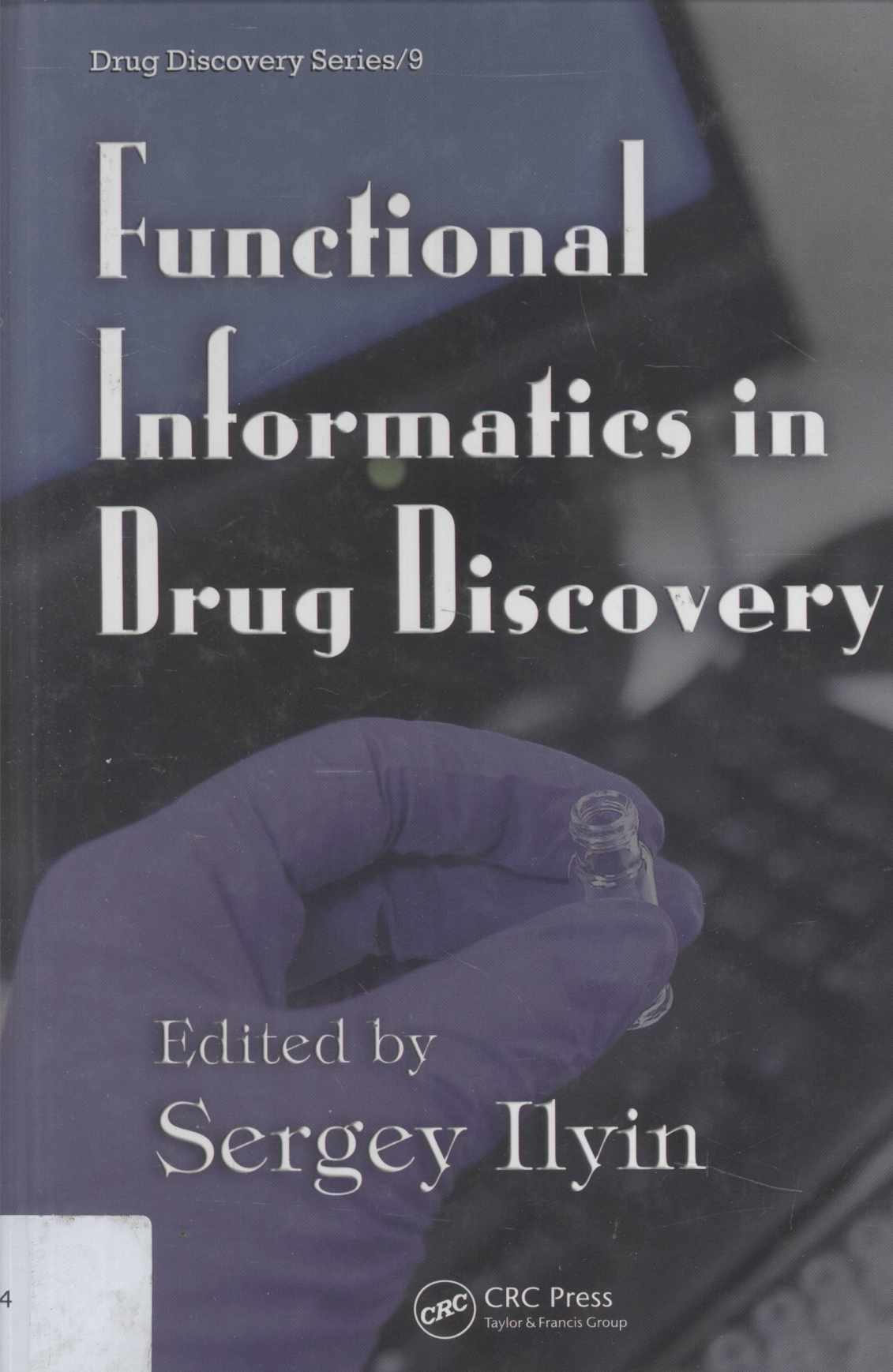
Drug Discovery Series/9

Functional Informatics in Drug Discovery



Edited by

Sergey Ilyin

CRC Press
Taylor & Francis Group

Functional Informatics in Drug Discovery

Edited by

Sergey Ilyin

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business

CRC Press
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Printed in the United States of America on acid-free paper
10 9 8 7 6 5 4 3 2 1

International Standard Book Number-13: 978-1-57444-466-7 (Hardcover)

Library of Congress Cataloging-in-Publication Data

Functional informatics in drug discovery / edited by Sergey Ilyin.
p. ; cm. -- (Drug discovery series ; 9)
Includes bibliographical references and index.
ISBN-13: 978-1-57444-466-7 (hardcover : alk. paper)
ISBN-10: 1-57444-466-2 (hardcover : alk. paper)
1. Bioinformatics. 2. Medical informatics. 3. Computational biology. 4. Pharmaceutical technology--Data processing. I. Ilyin, Sergey. II. Series.
[DNLM: 1. Computational Biology--methods. 2. Medical Informatics. 3. Technology, Pharmaceutical--methods. QU 26.5 F979 2007]

QH324.2.F86 2007
615'.19--dc22 2007013029

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

Functional Informatics in Drug Discovery

Drug Discovery Series

Series Editor

Andrew Carmen

Johnson & Johnson PRD, LLC
San Diego, California, U.S.A.

1. Virtual Screening in Drug Discovery, *edited by Juan Alvarez and Brian Shoichet*
2. Industrialization of Drug Discovery: From Target Selection Through Lead Optimization, *edited by Jeffrey S. Handen, Ph.D.*
3. Phage Display in Biotechnology and Drug Discovery, *edited by Sachdev S. Sidhu*
4. G Protein-Coupled Receptors in Drug Discovery, *edited by Kenneth H. Lundstrom and Mark L. Chiu*
5. Handbook of Assay Development in Drug Discovery, *edited by Lisa K. Minor*
6. *In Silico* Technologies in Drug Target Identification and Validation, *edited by Darryl León and Scott Markel*
7. Biochips as Pathways to Drug Discovery, *edited by Andrew Carmen and Gary Hardiman*
8. Functional Protein Microarrays in Drug Discovery, *edited by Paul F. Predki*
9. Functional Informatics in Drug Discovery, *edited by Sergey Ilyin*

Preface

The biopharmaceutical industry is facing tremendous and fascinating opportunities to improve its overall research and development and business efficiency by integrating various technologies and informational systems and by creating an intelligent interface between different systems and business segments. Currently there is no reference that has (a) looked into all of the different aspects of technology integration and information flow in the biopharmaceutical enterprise and (b) outlined the specifics and commonalities of technologies at different stages of development. An important and challenging aspect of this book is the in-depth analysis of emerging trends and future opportunities in integration and interfacing while maintaining a systematic or programmatic approach. This book comprises well-referenced updated chapters from leaders in the pharmaceutical industry, in academia, and in information technology. Each chapter reviews a particular area.

This book is intended for a heterogeneous audience, essentially anyone who seeks a greater understanding of the concepts and utilization of informatics. At the same time, sufficient detail and updated references are included so that scientists in any discipline, managers, and investors can benefit by better understanding these emerging trends and their applications.

Editor

Sergey Ilyin graduated with distinction from the St. Petersburg State University in Russia with a combined B.Sc./M.Sc. degree in 1993. Following a fellowship at the Institute of Cytology of the Russian Academy of Sciences, he received an award from the Soros Foundation and accepted a full scholarship at the University of Delaware. Ilyin graduated from the University of Delaware with a Ph.D. degree in molecular biology/neuroimmunology under the mentorship of Professor Carlos R. Plata-Salaman, D.Sc., M.D., an expert in neurosciences with a focus in brain cytokine research. In this research, Ilyin described, for the first time, the integrative regulation of the brain interleukin-1 system under various pathophysiological conditions and in the induction and progression of anorexia in various models.

In 1999, Ilyin joined the Central Nervous System Research Team at the Johnson & Johnson Pharmaceutical Research Institute as a postdoctoral research fellow. Following this, Ilyin was promoted through positions of increasing scope and responsibility that included establishing novel targets, projects, screening technologies and participation in several key drug-discovery projects. Ilyin is currently a group leader of bioinformatics at the Spring House facility of the Johnson & Johnson Pharmaceutical Research and Development LLC.

He has pioneered bioinformatic approaches in drug discovery; has conducted or directed projects in the areas of CNS, analgesia, vascularity, metabolism, oncology, and enterology; and has developed new approaches for target identification and validation. His activities also include data management and integration as well as research on biological markers for disease, drug efficacy, and safety applications, including identification and analysis of mechanisms of action. His responsibilities were later extended to translational technologies and approaches, including noninvasive imaging.

Ilyin has extensive experience in collaborating with different functions and teams within the company as well as with external networks and academic groups. He has published more than 40 peer-reviewed articles and has editorial responsibilities for several journals and books.

Contributors

G. Mark Anderson
Centocor Research & Development, Inc.
Malvern, Pennsylvania

Greg M. Arndt
Johnson & Johnson Research Pty. Ltd.
Eveleigh, Australia

Douglas E. Brenneman
Johnson & Johnson Pharmaceutical Research and Development LLC
Spring House, Pennyslvania

Katherine R. Calvo
National Institutes of Health
Bethesda, Maryland

Andrew Carmen
Johnson & Johnson Pharmaceutical Research and Development LLC
San Diego, California

Qiming Chen
Centocor Research & Development, Inc.
Malvern, Pennsylvania

James M. Dixon
Johnson & Johnson Pharmaceutical Research and Development LLC
Spring House, Pennsylvania

Mark Flynn
Los Alamos National Laboratory
Los Alamos, New Mexico

Daniel Horowitz
Johnson & Johnson Pharmaceutical Research and Development LLC
Spring House, Pennsylvania

Bingren Hu
University of Miami School of Medicine
Miami, Florida

Sergey E. Ilyin
Johnson & Johnson Pharmaceutical Research and Development LLC
Spring House, Pennsylvania

Fredrik Kamme
Johnson & Johnson Pharmaceutical Research and Development LLC
San Diego, California

Garrett T. Kenyon
Los Alamos National Laboratory
Los Alamos, New Mexico

Lance A. Liotta
National Institutes of Health
Bethesda, Maryland

Changlu Liu
Johnson & Johnson Pharmaceutical Research and Development LLC
San Diego, California

Ewa Malatynska
Johnson & Johnson Pharmaceutical Research and Development LLC
Spring House, Pennsylvania

Bernd Meurers
David Geffen School of Medicine at UCLA
Los Angeles, California

Marian T. Nakada
Centocor Research & Development, Inc.
Malvern, Pennsylvania

Emanuel F. Petricoin
U.S. Food and Drug Administration
Bethesda, Maryland

Albert Pinhasov
Johnson & Johnson Pharmaceutical
Research and Development LLC
Spring House, Pennsylvania

Carlos R. Plata-Salamán
Eli Lilly and Company
Indianapolis, Indiana

Daniel Rosenthal
Johnson & Johnson Pharmaceutical
Research and Development LLC
Spring House, Pennsylvania

Bernard J. Scallon
Centocor Research & Development, Inc.
Malvern, Pennsylvania

Linda A. Snyder
Centocor Research & Development, Inc.
Malvern, Pennsylvania

Da-Thao Tran
Johnson & Johnson Pharmaceutical
Research and Development LLC
San Diego, California

Anil H. Vaidya
Johnson & Johnson Pharmaceutical
Research and Development LLC
Spring House, Pennsylvania

Ole Vesterqvist
Bristol-Myers Squibb Pharmaceutical
Research Institute
Princeton, New Jersey

Hong Xin
Johnson & Johnson Pharmaceutical
Research and Development LLC
Spring House, Pennsylvania

Li Yan
Centocor Research & Development, Inc.
Malvern, Pennsylvania

Jingxue Yu
Johnson & Johnson Pharmaceutical
Research and Development LLC
San Diego, California

Jessica Zhu
Johnson & Johnson Pharmaceutical
Research and Development LLC
San Diego, California

Contents

1 Intelligent Automation

James M. Dixon

CONTENTS

INTRODUCTION

Progress in science depends on the elucidation of new concepts as well as the development of better tools for carrying out scientific research. The tools for scientific research have traditionally included instruments for measurement and experimentation. The advent of computers has provided a powerful mechanism for use in scientific research. All applications of computers in science require the development of software. Software applications vary greatly from informatics to database management to intelligent automation. There even exists a multitude of examples that have merged automation and informatics to form what is known as functional informatics [1, 2]. No matter what the application, computers have become a great asset in expanding our scientific knowledge. Of particular note are the great strides in software advances in automation, specifically in laboratory automation. With the use of intelligent software, output can be obtained with little or no user intervention or data analysis. Processes that were previously bottlenecks have evolved to provide a wealth of data and useful information with the adaptation of intelligent automation. In this chapter we will visit reaction optimization and see how the expansion of its adaptation into the biological sciences can expedite data collection and analysis.

LABORATORY AUTOMATION

Over the past several decades there have been major developments in laboratory automation stemming from the emergence of systems capable of processing large numbers of samples in parallel [3–7]. Laboratory automation can range from simple automation, such as automated analysis instruments (i.e., Cell Lab Quanta® by Beckman Coulter, Inc.), to more-complex integrated systems, such as intelligent laboratory workstations (i.e., Biomek® 2000 Laboratory Automation Workstation by Beckman Coulter, Inc.).

The range of laboratory automation can be broken into two levels of user input: open-loop and closed-loop experimentation. In open-loop experimentation, an experiment or analysis is set up and run with no decisions made based on data from previous or ongoing experiments. More often than not, the experiment or analysis was only meant for serial evaluation, and decisions are not necessary. This allows for easy use of the instrument by scientists, but it restricts the utility of the instrument because few, if any, adaptive changes can be made to the experimentation. In contrast, in closed-loop experimentation, ongoing experiments are evaluated, and future experiments are pruned, altered, or spawned based on data from previous or ongoing experiments [8]. Current integrated systems are moving in the direction of closed-loop experimentation, which is flexible and configurable and would allow the automated system to be a walk-away device. In return, this approach offers the prospect of increased productivity while reducing scientist intervention.

One of the more beneficial aspects of integrated systems stems from powerful software that allows scheduling of experimentation. Multiple sets of experiments can be implemented in parallel through the use of a scheduler. A simple scheduler offsets the start time of intact experimental plans and interleaves (in a comblike manner) the individual commands of the respective plans. The resulting schedule consists of a set of experimental plans with offset start times; in this manner the total duration of the set of parallel experimental plans is generally compressed by up to tenfold compared with that for serial implementation [9]. More-complex schedulers exist that can order the experiments in a particular fashion, such as by user ranking or by shortest experiment duration first.

Automated experimentation instruments, created from the combination of computers with robotics, have been assembled for diverse applications ranging from high-throughput screening to library preparation to reaction optimization. Because there is a push toward more-intelligent automation, our efforts will concentrate on reaction optimization, which provides the most desirable benefits of intelligent design.

REACTION OPTIMIZATION

Diverse automation systems, ranging from batch reactors to multireactor workstations, have been constructed with the dominant application of performing reaction optimization. Reaction optimization, an unglamorous but integral component of scientific research, is essential for achieving high-yield products and for developing cost-effective and environmentally benign processes [10]. By definition, optimization

implies the ability to perform experiments, to evaluate data, and to perform modified experiments in an effort to achieve improved results. Box and coworkers [11] were among the pioneers in reaction optimization. Their studies of evolutionary operation in industrial settings were developed over a half a century ago and still hold true today.

There is a large set of evolutionary approaches to perform reaction optimizations. The approaches can be broken into four algorithm categories: (a) parallel, (b) adaptive, (c) parallel adaptive, and (d) integrated. In the following sections, each type of optimization algorithm is described along with some examples of each type.

Parallel Algorithms

One of the simplest conceptual approaches for reaction optimization, open-loop experimentation, involves the examination of all combinations of factors that affect a given reaction (i.e., a full factorial design). In the case of a search space defined by two factors, all points in a regular two-dimensional grid are examined. The resulting data can then be plotted to give a response surface. Response surfaces for reactions are very valuable to scientists, but generally are not widely available due to the extensive manual labor required to investigate a large number of experiments. One approach to minimize the extent of manual experimentation has been to employ partial factorial designs, which examine only part of the space and then employ statistical approaches to tease out interactions among factors. The methodology of statistically designed experiments is well developed, but it has made limited inroads among scientists.

Automated workstations provide the means to perform a vast number of experiments with minimal scientist intervention, which reduces the laborious task of performing tedious experiments and eliminates the need for statistical treatment. Thus, grid searches (factorial designs) can be a viable option for optimization. The fact that grid searches are performed with no decision-making features means that, upon completion of all experiments, the automation software can only decide whether and where an optimum value exists [12]. As previously mentioned, the duration of the experimentation is reduced dramatically by performing the experiments in parallel. What would have taken an exorbitant amount of time to obtain scientific data can be reduced to a manageable duration.

Adaptive Algorithms

Adaptive algorithms are the foundation of closed-loop experimentation. Algorithms that can make scientist-independent decisions have made great strides toward intelligent automation. One of the most commonly used adaptive algorithms is the simplex algorithm [13], which is a well-known method for hill-climbing optimization. Many modifications to the original simplex algorithm have been developed. Betteridge et al. [14] have devised a robust method, called the composite modified simplex (CMS), that combines the best features of various modified simplex methods [15].

A simplex is an n-dimensional polygon with $(n + 1)$ vertices, where n is the number of control variables for optimization and each vertex has n coordinates. The

simplex is triangular in two-dimensional space, tetrahedral in three-dimensional space, and so forth. The optimization begins with a set of initial experiments whose number is equal to the number of control variables plus 1 ($n + 1$). According to the experimental results, the subsequent vertex is projected in a direction opposite from the worst vertex (Figure 1.1). The new simplex consists of one new point and n points from the previous simplex (i.e., discarding the worst point and replacing it with a new point). Consequently, despite the degree of the dimensional space, only a single

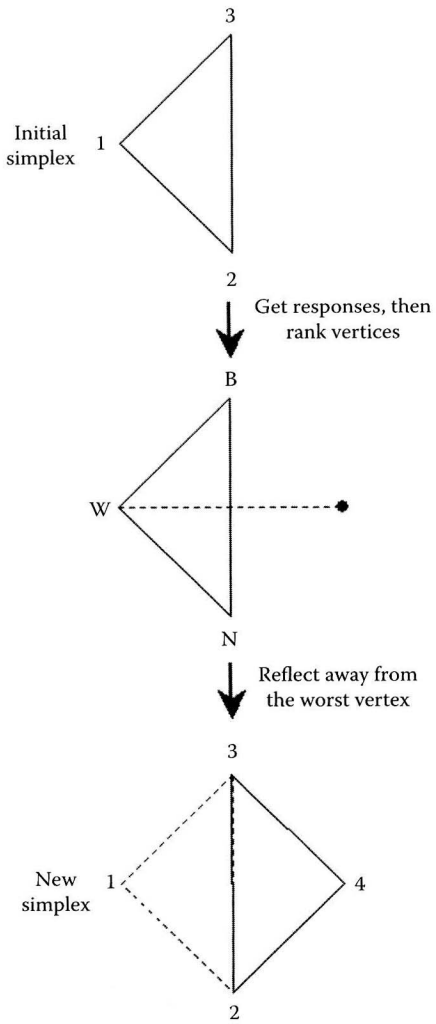


FIGURE 1.1 CMS movement. The basic simplex (i.e., CMS) moves in a two-dimensional search space. In an n-dimensional space, the simplex algorithm discards the one worst point in each simplex, maintains n points, and projects one new vertex with each move. The simplex algorithm requires serial implementation. (B: best response point; N: next best point; W: worst response point.)

experiment is proposed aside from the $(n + 1)$ points in the initial cycle. Repetitive measurements of the response and the reflection of simplices form the basis for the most elementary simplex algorithm. The CMS optimization is attractive because the calculations are straightforward, the search space is expandable to encompass multiple dimensions, and the method is robust in the presence of experimental noise [9].

Parallel Adaptive Algorithms

A fusion of parallel and adaptive algorithms is the culmination of efforts to create closed-loop experimentation. Parallel adaptive algorithms allow for wide searching of search spaces with convergent properties. A straightforward example of a parallel adaptive algorithm is a grid search that has evolutive properties to converge on an optimum response. A regular grid of points is examined in one cycle; then a second, more focused, grid is situated around the region of optimal response. A third even more tightly focused grid is then examined around the region of optimal response observed in the preceding cycle (Figure 1.2). In this manner, the entire search space is examined, the region of optimal response is identified, and an increasingly fine-grained search is implemented in the region of optimal response. This algorithm for experimentation affords both a coarse-grained response surface for the entire search space and a fine-grained evaluation in the region of greatest interest without requiring the entire search space to be examined with the same fine graining [16].

Another useful parallel adaptive algorithm is the multidirectional search (MDS) method. The MDS algorithm was created by Torczon to overcome the serial nature

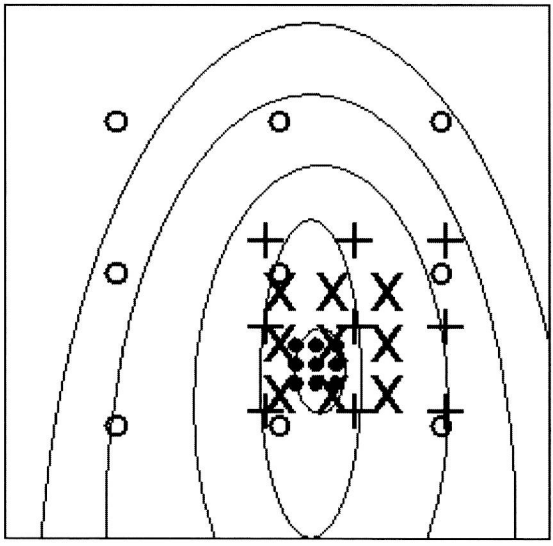


FIGURE 1.2 Successively focused grid searches. The first grid is cast broadly (○). The second grid (+) is centered around the point from the first grid that gave the highest point. A third grid (□) and fourth grid (•) are projected in a similar fashion. In this manner, the entire surface is examined in a coarse-grained fashion, and the optimal region is examined in a fine-grained fashion.

of the simplex method [17–20]. Torczon's work stemmed from considering how best to take advantage of multiple parallel processors in the computational evaluation of unconstrained optimization problems. The resulting MDS algorithm is also simplex-based, but differs in a fundamental way from the traditional simplex method. As previously stated, in a simplex algorithm, each move occurs by reflection away from the one worst point, creating a new simplex that contains one new point, regardless of search-space dimension. In each move, the one worst point is discarded, and the one new point of the new simplex is evaluated. In contrast, in the MDS method, each move occurs by reflection away from all but the one best point (Figure 1.3). The new simplex in n-dimensional space is composed of n new points and only the single best point from the previous simplex.

The MDS algorithm has a further distinction from that of a simplex movement. In addition to those points that are required to iterate simplex projections through the space (mandatory points), exploratory points can be evaluated in each cycle of MDS to the extent that resources are available to do so. The exploratory points are identified by the look-ahead projection of possible future simplexes. Examples of the range of exploratory points that can be examined for the initial cycle are shown in Figure 1.4. The number of experiments implemented per cycle depends on the dimension of the search space and the batch capacity of the workstation (which limits the number of exploratory experiments). Only one search space is investigated in a given course of experiments. The MDS method provides a means of evolutionary optimization via parallel experimentation. Thus, a simplex search projects only one new point per cycle, whereas MDS projects at least n new points per cycle [21].

Another approach to increase parallelism with simplex methods is to project multiple simplexes on a single search space. The simplex searches are independent in the direction of their movement but march in lockstep. A set of experiments of number equal to $m \times (n + 1)$ is generated as the initial trial, and m experiments are proposed for the subsequent cycle, where m is the number of independent simplex searches and n is the dimension of the search space. In a given search, the number of simplexes can be chosen by the user; the number generally can be determined by a set pattern to achieve an even distribution of initial simplexes, and the number increases with search-space dimension (Figure 1.5) [22, 23]. One of the greatest advantages of this method is that it minimizes an inherent flaw of simplex searches. With one simplex, convergence is possible on a local maximum/minimum. However, multiple simplexes have a much higher probability of converging on the global maximum/minimum because of the sheer number of converging simplexes.

Integrated Algorithms

Integrated algorithms combine the screening capabilities of parallel algorithms with the convergence properties of adaptive algorithms, but they do so in a unique way compared with parallel adaptive algorithms. There are many methods for integrating the different optimization algorithms. One particular example of interest is a technique that uses a two-tiered approach. The first tier employs a broad search to mark promising areas (breadth-first search). The second tier employs in-depth searches according to the results of the first-tier survey [24]. Figure 1.6 illustrates this

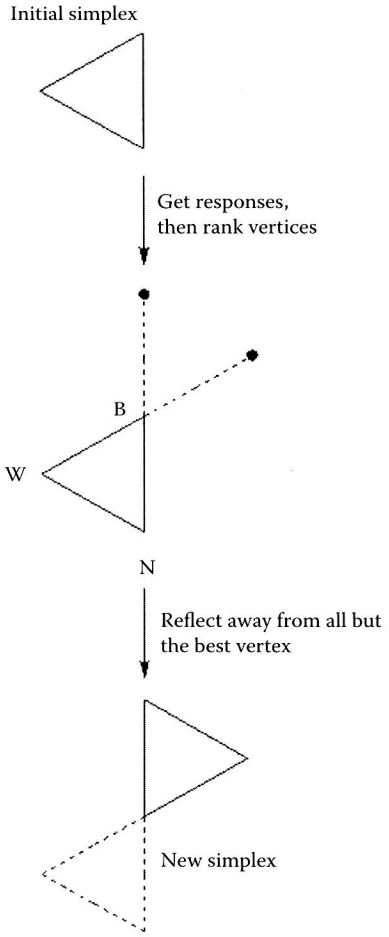


FIGURE 1.3 Basic MDS movement in a two-dimensional search space. In an n-dimensional space, the MDS algorithm discards n points in each simplex (all but the best), maintains the one best point, and projects n new vertices with each move. The MDS algorithm is amenable to parallel implementation. (B, N, and W are as defined in Figure 1.1.)

approach. This technique has several great advantages over normal methods of optimization: (1) less experimentation is performed because areas with a lower likelihood of having the optimum response are ignored, (2) focusing on the optimum response in the first tier allows the convergence of adaptive algorithms to remain fine grained, and (3) chemical and specimen samples are conserved because fewer resources are needed to perform an experiment.

Parallel, adaptive and integrated approaches take the best of both parallel and adaptive features, creating an amalgam of convergent and exploratory experimentation. This, in turn, provides a greater throughput and the possibility of walk-away experimentation compared with either parallel or adaptive experimentation individually.