

Appendix C

Programs

C.1 Contents of the Floppy Disk

The floppy disk that accompanies this book contains:

- Sources of FORTRAN subroutines of the following direct optimization procedures as described in the Chapters 3, 5, and 7 of the book.
 - FIBO Coordinate strategy with Fibonacci division
fiboh (fiboh.f) calls subroutine fibo (fibo.f)
 - GOLD Coordinate strategy with Golden section
goldh (goldh.f) calls subroutine gold (gold.f)
 - LAGR Coordinate strategy with Lagrangian interpolation
lagrh (lagrh.f) calls subroutine lagr (lagr.f)
 - HOJE Strategy of Hooke and Jeeves (pattern search)
hoje (hoje.f) calls subroutine hilf (hilf.f)
 - ROSE Strategy of Rosenbrock (rotating coordinates search)
rose (rose.f) calls subroutine grsmr (grsmr.f)
 - DSCG Strategy of Davies, Swann, and Campey
with Gram-Schmidt orthogonalization
dscg (dscg.f) calls subroutine lineg (lineg.f)
subroutine grsmd (grsmd.f)
 - DSCP Strategy of Davies, Swann, and Campey
with Palmer orthogonalization
dscp (dscp.f) calls subroutine linep (linep.f)
subroutine palm (palm.f)
 - POWE Powell's strategy of conjugate directions
powe (powe.f) calls -
 - DFPS Davidon, Fletcher, Powell strategy (Variable metric)
dfps (dfps.f) calls subroutine seth (seth.f)
subroutine grad (grad.f)
function updot (updot.f) calls dot (dot.f)
function dot (dot.f)

- SIMP	Simplex strategy of Nelder and Mead
simp (simp.f) calls	-
- COMP	Complex strategy of M. J. Box
comp (comp.f) calls	-
- EVOL	Two membered evolution strategy
evol (evol.f) calls	function z (included in evol.f)
- KORR	Multimembered evolution strategy
korr2 (korr2.f) calls	function zulass (included in korr2.f)
	function gaussn (included in korr2.f)
	function bletal (included in korr2.f)
	subroutine pruefg (included in korr2.f)
	subroutine speich (included in korr2.f)
	subroutine mutati (included in korr2.f)
	subroutine umspei (included in korr2.f)
	subroutine minmax (included in korr2.f)
	subroutine gnpool (included in korr2.f)
	subroutine abscha (included in korr2.f)
	subroutine drehng (included in korr2.f)

Additionally, FORTRAN function sources of the 50 test problems are included:

- ZIELFU(N,X) one objective function with a computed GOTO for 50 entries.
- RESTRI(J,N,X) one constraints function with a computed GOTO for 50 entries and J as current number of the single restriction.

No runtime package is provided for this set, however.

- C sources for all strategies mentioned above and C sources for the 50 test problems (GRUP with option REKO is missing since it has become one special case within KORR).
- A set of simple interfaces to run 13 of the above mentioned optimization routines with the above mentioned 50 test problems on a PC or workstation.

C.2 About the Program Disk

The floppy disk contains both FORTRAN and C sources for each of the strategies described in the book. All test problems presented in the catalogue of problems (see appendix A) exist as C code. A set of simple interfaces, easy to understand and to expand, combines the strategies and functions to *OptimA*, a ready for use program package.

The programs are designed to run on a minimally configured PC using a math-coprocessor or having an 80486 CPU and running the DOS or LINUX operating system. To accomplish semantic equivalence with the well tested original FORTRAN codes, all strategies have been translated via `f2c`, a Fortran-to-C converter of AT&T Bell Laboratories. All C codes can be compiled and linked via `gcc` (Gnu C compiler, version 2.4). Of course,

any other ANSI C compiler such as Borland C++ that supports 4-byte-integers should produce correct results as well.

LINUX and `gcc` are freely available under the conditions of the GNU General Public License. Information about ordering the Gnu C compiler in the United States is available through the Free Software Foundation by calling 617 876 3296.

All C programs should compile and run on any UNIX workstation having `gcc` or another ANSI C compiler installed.

C.3 Running the C Programs

The following instructions are appropriate for installing and running the C programs on your PC or workstation. Installation as well as compilation and linking can be carried out automatically.

C.3.1 How to Install OptimA on a PC Using LINUX or on a UNIX Workstation

First, enter the directory where you want *OptimA* to be installed. Then copy the installation file via `mtools` by typing the command:

```
mcopy a:install.sh .
```

If you don't have `mtools`, copy `wb-1p?.tar` from floppy to workspace and untar it. The instruction

```
sh install.sh
```

will copy the whole tree of directories from the disk to your local directory. The following directories and subdirectories will be created:

```
fortran
funct
include
lib
rstruct
strat
util
```

To compile, link, and run *OptimA* go to the `workbench` directory and type

```
make
```

to start a recursive compilation and linking of all C sources.

C.3.2 How to Install OptimA on a PC Under DOS

First, enter the directory where you want *OptimA* to be installed. The instruction

```
a:INSTALL
```

or

```
b:INSTALLB
```

will copy the whole tree of directories from the disk to your local directory. The same directories and subdirectories as mentioned above will be created. To compile, link, and run *OptimA* go to the `workbench` directory and type

```
mkOptimA
```

to start a recursive compilation and linking of all C sources. This will take a while, depending on how fast your machine works.

C.3.3 Running OptimA

After the successful execution of `make` or `mkOptimA`, respectively, the executable file `OptimA` is located in the subdirectory `bin`. Here you can run the program package by issuing the command

```
OptimA
```

First, the program will list the available strategies. After choosing a strategy by typing its number, a list of test problems is displayed. Type a number or continue the listing by hitting the `return` key. Depending on the method and the problem, the program will ask for the parameters to configure the strategy. Please refer to Chapter 6 and Appendix A to choose appropriate values. Of course, you are free to define your own parameter values, but please remember that the behavior of each strategy strongly depends on its parameter settings.

Warnings during the process will inform the user of inappropriate parameter definitions or abnormal program behavior. For example, the message `timeout reached` warns the user that the strategy may find a better result if the user defined maximal time were set to a larger value. The strategies `COMP`, `EVOL`, and `KORR` will try at most five restarts after the first timeout occurred.

If a strategy that can process unrestricted problems only is applied to a restricted problem, a warning will be displayed, too. After the acknowledgement of this message by hitting the `return` key, the user can choose another function.

C.4 Description of the Programs

The following pages briefly describe the programs on which this package is based. A short description of how to incorporate self-defined problem functions to *OptimA* follows.

The directory **FORTTRAN** lists all the original codes described in the book. The reader may write his own interfaces to these programs. For further information please refer to the C sources or to Schwefel (1980, 1981).

All C source codes of the strategies have been translated from FORTRAN to C via **f2c**. Some modifications in the C sources were done to gain higher portability and to achieve a homogeneous program behavior. For example, all strategies are minimizing, use standard output functions, and perform operations on the same data types. All modifications did not change the semantics of any strategy.

To each optimization method a dialogue interface has been added. Here the strategy's specific parameter definition takes place. In the comments within the program listings the meaning and usage of each parameter is briefly described. All names of the dialogue interfaces end with the suffix "**_mod.c**." The strategies together with the interfaces are listed in the directory named **strat**.

The whole catalogue of problems (see Appendix A) has been coded as C functions. They are collected in the subdirectory **funct**.

The problems 2.29 to 2.50 (see Appendix A) are restricted. Therefore, constraints functions to these problems were written and listed in directory **rstruct**. Because in some problems the number of constraints to be applied depends on the dimension of the function to be optimized, this number has to be calculated. This task is performed by the programs with prefix "**rsn_**." The evaluation of the constraint itself is done in the modules with prefix "**rst_**." A restriction holds if its value is negative.

All strategies perform operations on vectors of varying dimensions. Therefore a set of tools to allocate and to define vectors is compiled in the package **vec_util** which is located in the subdirectory **util**. The procedures from this package are used only in the dialogue interfaces. All other programs perform operations on vectors as if they would use arrays of arbitrary but fixed length.

The main program "**OptimA.c**" performs only initialization tasks and runs the dialogue within which the user can choose a strategy and a function number.

The strategies and functions are listed in tables, namely "**func_tab.c**" and "**strt_tab.c**." If the user wants to incorporate new problems to *OptimA* the table "**func_tab.c**" has to be extended. This task is relatively simple for a programmer with little C knowledge if he follows the next instructions carefully.

C.4.1 How to Incorporate New Functions

The following template is typical for every function definition:

```
#include "f2c.h"
#include "math.h"

double real    probl_2_18(int n, double real *x)
```

```

{
    return(0.26*(x[0]*x[0] + x[1]*x[1])-0.48*x[0]*x[1] );
}

```

Please add your own function into the directory `funct`. Here you will find the file “`func_tab.c`.” Include the formal description of your problem into this table. A typical template looks like:

```

{
    5,
    rs_nm_x_x,
    restr_x_x,
    "Problem x_x (restricted problem):\n\t x[1]+x[2]+... ",
    probl_x_x
},

```

with the data type definition:

```

struct functions {
    long int      dim;          /* Problem's dimension */
    long int      (*rs_num)();  /* Calculates the number */
                                /* of constraints */
    doublereal    (*restrictions)(); /* Constraints function */
    char*         name;        /* Mathem. description */
    doublereal    (*function)(); /* Objective function */
};

```

```
typedef struct functions funct_t;
```

- The first item denotes the number of dimensions of the problem. A problem with variable dimension will be denoted by a `-1`. In this case the program should inquire the dimension from the user.
- The second entry denotes the function that calculates the numbers of constraints to be applied to the problem. If no constraints are needed a `NULL` pointer has to be inserted.
- The next line will be displayed to the user during an *OptimA* session. This string provides a short description of the problem, typically in mathematical notation.
- The last item is a function-pointer to the objective function.

Please do not add a new formal problem description into the `func_tab` behind the last table entry. The latter denotes the end of the table and should not be displaced.

To inform all problems of the new function, its prototype must be included into the header file `func_names.h`.

As a last step the `Makefile` has to be extended. The lists `FUNCTSRCS` and `FUNCTOBS` denote the files that make up the list of problems. These lists have to be extended by the filename of your program code.

Now step back to the directory `C` and issue the command `make` or `mkOptimA`, respectively, to compile “`OptimA`.”

Restrictions can be incorporated into *OptimA* like functions. Every `C` code from the directory `rstruct` can be taken as template. The name of the constraints function and the name of the function that calculates the number of constraints has to be included in the formal problem description.

C.5 Examples

Here two examples of how *OptimA* works in real life will be presented. The first one describes an application of the multimembered evolution strategy KORR to the corridor model (problem 2.37, function number 32). The second example demonstrates a batch run. The batch mode enables the user to apply a set of methods to a set of functions in one task.

C.5.1 An Application of the Multimembered Evolution Strategy to the Corridor Model

After calling `OptimA` and choosing problem 2.37 by typing 32, a typical dialogue will look like:

```
Multimembered evolution strategy applied to function:
```

```
Problem 2.37 (Corridor model) (restricted problem):
    Sum[-x[i],{i,1,n}]
```

```
Please enter the parameters for the algorithm:
```

```
Dimension of the problem           : 3
Number of restrictions             : 7
```

```
Number of parents                 : 10
Number of descendants              : 100
Plus (p) or the comma (c) strategy : c
Should the ellipsoid be able to rotate (y/n) : y
```

```
You can choose under several recombination types:
```

```

1   No recombination
2   Discrete recombination of pairs of parents
3   Intermediary recombination of pairs of parents
4   Discrete recombination of all parents
5   Intermediary recombination of all parents in pairs
Recombination type for the parameter vector      : 2
Recombination type for the sigma vector         : 3
Recombination type for the alpha vector         : 1
Check for convergence after how many generations (> 2*Dim.) : 10
Maximal computation time in sec.                : 30
Lower bound to step sizes, absolute             : 1e-6
Lower bound to step sizes, relative            : 1e-7
Parameter in convergence test, absolute        : 1e-6
Parameter in convergence test, relative       : 1e-7
Common factor used in step-size changes (e.g. 1) : 1
Standard deviation for the angles
of the mutation ellipsoid (degrees)           : 5.0
Number of distinct step-sizes                 : 3
Initial values of the variables :
0
0
0
Initial step lengths :
1
1
1

Common factor used in step-size changes      : 0.408248
Individual factor used in step-size changes  : 0.537285

```

Starting at : $F(x) = 0$

Time elapsed : 18.099276

Minimum found : -300.000000
at point : 99.999992 100.000000 99.999992
Current best value of population: -300.000000

C.5.2 OptimA Working in Batch Mode

OptimA also supports a batch mode option. This option was introduced to enable a user to test the behavior of any strategy by varying parameter settings automatically. Of

course, any function or method may be changed during a run, as well. The batch file that will be processed should contain the list of input data you would type in manually during a whole session in non-batch mode. *OptimA* in batch mode suppresses the listing of the strategies and functions. That reduces the output a lot and makes it better readable.

A typical batch run looks like:

```
OptimA -b < bat_file > results
```

With a “bat_file” like:

```
8
1
100.100
0.98e-6
0.0e+0
5
5
1
1
0.8e-6
0.8e-6
0.111
0.111
y
```

the file “results” may look like:

```
Method #      : 8
Function #    : 1
```

DFPS strategy (Variable metric) applied to function:

Problem 2.1 (Beale):

$$(1.5-x*(1-y))^2 + (2.25-x*(1-y^2))^2 + (2.625-x*(1-y^3))^2$$

```
Dimension of the problem          : 2
Maximal computation time in sec.  : 100.100000
Accuracy required                 : 9.8e-07
Expected value of the objective function
at the optimum                   : 0
Initial values of the variables :
5
5
Initial step lengths :
1
1
Lower bounds of the step lengths :
```

8e-07

8e-07

Initial step lengths for construction of derivatives :

0.111

0.111

Starting at : $F(x) = 403069$

Time elapsed : 0.033332

Minimum found : 0.000000

at point : 3.000000 0.500000

Both examples have been run on a SUN SPARC S10/40 workstation.

The floppy disk included into this book may not be copied, sold, or redistributed without the permission of John Wiley & Sons, Inc., New York.

Index

- Aarts, E.H.L., 161
Abadie, J., 17, 24
Abe, K., 239
Ablay, P., 163
Absolute minimum, see global minimum
Accuracy of approximation, 26, 27, 29, 32, 38, 41, 70, 76, 78, 81, 91, 92, 94, 116, 146, 167, 168, 173, 175, 206–208, 213, 214, 235
Accuracy of computation, 12, 14, 32, 35, 54, 57, 66, 67, 71, 78, 81, 83, 88, 89, 99, 112–114, 145, 170, 173–175, 206, 209, 236, 329
Ackley, D.H., 152
Adachi, N., 77, 81, 82
Adams, R.J., 96
Adaptation, 5, 6, 9, 100, 102, 105, 142, 147, 152
Adaptive step size random search, 96, 97, 200
AESOP program package, 68
Ahrens, J.H., 116
AID program package, 68
Aizerman, M.A., 90
Akaike, H., 66, 67, 203
Alander, J.T., 152, 246
Aleksandrov, V.M., 95
Algebra, 5, 14, 41, 69, 75, 239
Alland, A., Jr., 244
Allen, P., 102
Allometry, 243
Allowed region, see feasible region
Altman, M., 68
Amann, H., 93
Analogue computers, 12, 15, 65, 68, 89, 99, 236
Analytic optimization, see indirect optimization
Anders, U., 246
Anderson, N., 35
Anderson, R.L., 91
Andrews, H.C., 5
Andreyev, V.O., 94
Animats, 103
Anscombe, F.J., 101
Antonov, G.E., 90
Aoki, M., 23, 93
Apostol, T.M., 17
Appelbaum, J., 48
Applications, 48, 53, 64, 68, 69, 99, 151, 245–246
Approximation problems, 5, 14, see also sum of squares minimization
Archer, D.H., 48
Arrow, K.J., 17, 18, 165
Artificial intelligence, 102, 103
Artificial life, 103
Asai, K., 94
Ashby, W.R., 9, 91, 100, 105
Atmar, J.W., 151
Automata, 6, 9, 44, 48, 94, 99, 102
Avriel, M., 29, 31, 33
Awdejewa, L.I., 18
Axelrod, R., 21
Azencott, R., 161

Bach, H., 23
Bäck, T., 118, 134, 147, 151, 155, 159, 245, 246, 248
Baer, R.M., 67
Balakrishnan, A.V., 11, 18
Balas, E., 18
Balinski, M.L., 19

- Banach, S., 10
 Bandler, J.W., 48, 115
 Banzhaf, W., 103
 Bard, Y., 78, 83, 205
 Barnes, G.H., 233, 239
 Barnes, J.G.P., 84
 Barnes, J.L., 102
 Barr, D.R., 241
 Barrier penalty functions (barrier methods), 16, 107
 Bass, R., 81
 Bauer, F.L., 84
 Bauer, W.F., 93
 Beale, E.M.L., 18, 70, 84, 166, 327, 346
 Beamer, J.H., 26, 29, 39
 Beckman, F.S., 69
 Beckmann, M., 19
 Behnken, D.W., 65
 Beier, W., 105
 Beightler, C.S., 1, 23, 27, 32, 38, 87
 Bekey, G.A., 12, 65, 89, 95, 96, 98, 99
 Belew, R.K., 152
 Bell, D.E., 20
 Bell, M., 44, 178
 Bellman, R.W., 11, 38, 102
 Beltrami, E.J., 87
 Bendin, F., 248
 Berg, R.L., 101
 Berlin, V.G., 90
 Berman, G., 29, 39
 Bernard, J.W., 48
 Bernoulli, Joh., 2
 Bertram, J.E., 20
 Bessel function, 129, 130
 Beveridge, G.S.G., 15, 23, 28, 32, 37, 64, 65
 Beyer, H.-G., 118, 134, 149, 159
 Biasing, 98, 156, 174
 Biggs, M.C., 76
 Binary optimization, 18, 247
 Binomial distribution, 7, 108, 243
 Bionics, 99, 102, 105, 238
 Birkhoff, G., 48
 Bisection method, 33, 34
 Björck, Å., 35
 Blakemore, J.W., 23
 Bledsoe, W.W., 239
 Blind random search, see pure random search
 Blum, J.R., 19, 20
 Boas, A.H., 26
 Bocharov, I.N., 89, 90
 Boltjanski, W.G. (Boltjanskij, V.G.), 18
 Boltzmann, L., 160
 Bolzano method, 33, 34, 38
 Booker, L.B., 152
 Booth, A.D., 67, 329
 Booth, R.S., 27
 Boothroyd, J., 33, 77, 178
 Born, J., 118, 149
 Borowski, N., 98, 240
 Bossert, W.H., 146
 Bourguine, P., 103
 Box, G.E.P., 6, 7, 65, 68, 69, 89, 101, 115, see also EVOP method
 Box, M.J., 17, 23, 28, 54, 56–58, 61, 68, 89, 115, 174, 332, see also complex strategy
 Boxing in the minimum, 28, 29, 32, 36, 41, 56, 209
 Brachistochrone problem, 11
 Bracken, J., 348
 Brajnes, S.N., 102
 Bram, J., 27
 Branch and bound methods, 18
 Brandl, V., 93
 Branin, F.H., Jr., 88
 Braverman, E.M., 90
 Bremermann, H.J., 100, 101, 105, 238
 Brent, R.P., 23, 27, 34, 35, 74, 84, 88, 89, 174
 Broyden, D.H., 95, 98, 99
 Broken rational programming, 20
 Bromberg, N.S., 89
 Brooks, S.H., 58, 87, 89, 91–95, 100, 174
 Brown, K.M., 75, 81, 84
 Brown, R.R., 66
 Broyden, C.G., 14, 77, 81–84, 172, 205

- Broyden-Fletcher-Shanno formula, 83
 Brudermann, U., 246
 Brughiera, P., 88
 Bryson, A.E., Jr., 68
 Budne, T.A., 101
 Buehler, R.J., 67, 68
 Bunny-hop search, 48
 Burkard, R.E., 18
 Burt, D.A., 48
- Calculus of observations, see observational
 calculus
- Campbell, D.T., 102
 Campey, I.G., 54, see also DSC strategy
 Campos, I., 248
 Canon, M.D., 18
 Cantrell, J.W., 70
 Carroll, C.W., 16, 57, 115
 Cartesian coordinates, 10
 Casey, J.K., 68, 89
 Casti, J., 239
 Catalogue of problems, 110, 205, 325–366
 Cauchy, A., 66
 Causality, 237
 C ea, J., 23, 47, 68
 Cembrowicz, R.G., 246
  erny, V., 160
 Chambliss, J.P., 81
 Chandler, C.B., 48
 Chandler, W.J., 239
 Chang, S.S.L., 11, 90
 Charalambous, C., 115
 Chatterji, B.N. and Chatterjee, B., 99
 Chazan, D., 239
 Chernoff, H., 75
 Chichinadze, V.K., 88, 91
 χ^2 distribution, 108
 Cholesky, matrix decomposition, 14, 75
 Chromosome mutations, 106, 148
 Circumferential distribution, 95–97, 109
  i ek, F., 106
 Clayton, D.G., 54
 Clegg, J.C., 11
 Cochran, W.G., 7
- Cockrell, L.D., 93, 99
 Cohen, A.I., 70
 Cohn, D.L., 100
 Collatz, L., 5
 Colville, A.R., 68, 174, 175, 339
 Combinatorial optimization, 152
 Complex strategy, 17, 61–65, 89, 115, 177,
 179, 185, 190, 191, 201, 202, 210,
 212, 213, 216, 217, 228–230, 232,
 327, 341, 346, 357, 361–363, 365,
 366
- Computational intelligence, 152
 Computer-aided design (CAD), 5, 6, 23
 Computers, see analogue, digital, hybrid,
 parallel, and process computers
- Concave, see convex
- Conceptual algorithms, 167
- Condition of a matrix, 67, 180, 203, 242,
 326
- Conjugate directions, 54, 69, 74, 82, 88,
 170–172, 202, see also Powell
 strategy
- Conjugate gradients, 38, 68, 69, 77, 81,
 169–172, 204, 235, see also
 Fletcher-Reeves strategy
- Conrad, M., 103
- Constraints, 8, 12, 14–18, 24, 44, 48, 49,
 57, 62, 87, 90–93, 105, 107, 115,
 119, 134, 150, 176, 212–214, 216,
 236
- Constraints, active, 17, 44, 62, 116, 118,
 213, 215
- Constraints satisfaction problem (CSP),
 91
- Contour tangent method, 39
- Control theory, 9, 11, 18, 23, 70, 88, 89,
 99, 112
- Convergence criterion, 113–114, 145–146,
 see also termination of the search
- Converse, A.O., 23
- Convex, 17, 34, 39, 47, 66, 101, 166, 169,
 236, 239
- Cooper, L., 23, 38, 48, 87

- Coordinate strategy, 41–44, 47, 48, 67, 87, 100, 164, 167, 172, 177, 200, 202–204, 207, 209, 228–230, 233, 327, 332, 339, 340, 362, 363, see also Fibonacci division, golden section, and Lagrangian interpolation
- Coordinate transformation, 241
- Cornick, D.E., 70
- Correlation, 118, 240, 241, 243, 246
- Corridor model objective function, 110, 116, 120, 123, 124, 134–142, 215, 231, 232, 351, 352, 361, 364, 365
- Cost of computation, 12, 38, 39, 64, 66, 74, 89, 90, 92, 168, 170, 179, 204, 230, 232, 234, see also rate of convergence
- Cottrell, B.J., 67
- Courant, R., 11, 66
- Covariances, 155, 204, 240, 241
- Cowdrey, D.R., 93
- Cox, D.R., 7
- Cox, G.M., 7
- Cragg, E.E., 70
- Created response surface technique, 16, 57
- Creeping random search, 94, 95, 99, 100, 236, 237
- Crippen, G.M., 89
- Criterion of merit, 2, 7
- Crockett, J.B., 75
- Crossover, 154
- Crowder, H., 70
- Cryer, C.W., 43
- Cubic interpolation, 34, 37, see also Lagrangian and Hermitian interpolation
- Cullum, C.D., Jr., 18
- Cullum, J., 83
- Curry, H.B., 66, 67
- Curse of dimensions, Bellman's, 38
- Curtis, A.R., 66
- Curtiss, J.H., 93
- Curve fitting, 35, 64, 84, 151, 246
- Cybernetics, 9, 101, 102, 322
- Dambrauskas, A.P., 58, 64
- Daniel, J.W., 15, 23, 68, 70
- Dantzig, G.B., 17, 57, 88, 166
- Darwin, C., 106, 109, 244
- Davidon, W.C., 77, 81, 82, 170
- Davidon-Fletcher-Powell strategy, see DFP strategy
- Davidor, Y., 152
- Davies, D., 23, 28, 54, 56, 57, 76, 81, see also Davies-Swann-Campey strategy
- Davies, M., 84
- Davies, O.L., 7, 58, 68
- Davies-Swann-Campey strategy, see DSC strategy
- Davis, L., 152
- Davis, R.H., 70
- Davis, R.S., 66, 89
- Davis, S.H., Jr., 23
- Day, R.G., 97
- Debye series, 130
- Decision theory, 94
- Decision tree methods, 18
- De Graag, D.P., 95, 98
- De Jong, K., 152
- Dekker, T.J., 34
- Demyanov, V.F., 11
- Denn, M.M., 11
- Dennis, J.E., Jr., 75, 81, 84
- Derivative-free methods, 15, 40, 80, 83, 172, 174, see also direct search strategies
- Derivatives, numerical evaluation of, 19, 23, 35, 66, 68, 71, 76, 78, 81, 83, 95, 97, 170–172
- Descendants, number of, 126, 142–144
- Descent, theory of, 100, 109
- Design and analysis of experiments, 6, 58, 65, 89
- D'Esopo, D.A., 41
- DeVogelaere, R., 44, 178
- DFP strategy, 77–78, 83, 97, 170–172, 243

- DFP-Stewart strategy, 78–81, 177, 178, 184, 189, 195, 200, 201, 209, 210, 219, 228–231, 337, 341, 343, 363, 364
- Diblock search, 33
- Dichotomous search, 27, 29, 33, 39
- Dickinson, A.W., 93, 98, 174
- Dieter, U., 116
- Differential calculus, 2, 11
- Digital computers, 6, 10–12, 14, 15, 32, 33, 92, 99, 110, 173, 236
- Dijkhuis, B., 37
- Dinkelbach, W., 17
- Diploidy, 106, 148
- Direct optimization, 13–15, 20
- Direct search strategies, 40–65, 68, 90
- Directed random search, 98
- Discontinuity, 13, 23, 25, 42, 88, 91, 116, 176, 211, 214, 231, 236, 341, 349
- Discovery, 2
- Discrete distribution, 110, 243
- Discrete optimization, 11, 18, 32, 39, 44, 64, 88, 91, 108, 152, 160, 243, 247
- Discrete recombination, 148, 153, 156
- Discretization, see parameterization
- Divergence, 35, 76, 169
- Dixon, L.C.W., 15, 23, 29, 34, 35, 58, 71, 76, 78, 81–83
- Dobzhansky, T., 101
- Dominance and recessiveness, 101, 106, 148
- Dowell, M., 35
- Draper, N.R., 7, 65, 69
- Drenick, R.F., 48
- Drepper, F.R., 103, 246
- Drucker, H., 61
- DSC strategy, 54–57, 74, 89, 177, 183, 188, 194, 200–202, 209, 228–230, 362, 363
- Dubovitskii, A.Ya., 11
- Dueck, G., 98, 164
- Duffin, R.J., 14
- Dunham, B., 102
- Dvoretzky, A., 20
- Dynamic optimization, 7, 9, 10, 48, 64, 89–91, 94, 99, 102, 245, 248
- Dynamic programming, 11, 12, 18, 149
- Ebeling, W., 102, 163
- Edelbaum, T.N., 13
- Edelman, G.B., 103
- Effectivity of a method, see robustness
- Efficiency of a method, see rate of convergence
- Eigen, M., 101
- Eigenvalue problems, 5
- Eigenvalues of a matrix, 76, 83, 326
- Eisenberg, M.A., 239
- Eldredge, N., 148
- Elimination methods, see interval division methods
- Elitist strategy, 157
- Elkin, R.M., 44, 66, 67
- Elliott, D.F., 83
- Ellipsoid method, 166
- Emad, F.P., 98
- Emery, F.E., 48, 87
- Engelhardt, M., 20
- Engeli, M., 43
- Enumeration methods, see grid method
- Epigenetic apparatus, 153, 154
- Equation, differential, 15, 65, 68, 93, 246, 345, 346
- Equations, system of, 5, 13, 14, 23, 39, 65, 66, 75, 83, 93, 172, 235, 336
- Equidistant search, see grid method
- Erlicki, M.S., 48
- Ermakov, S., 19
- Ermoliev, Yu., 19, 90
- Errors, computational, 47, 174, 205, 209, 210, 212, 219, 228, 229, 236
- Euclid of Alexandria, 32
- Euclidean norm, 167, 335
- Euclidean space, 10, 24, 49, 97
- Euler, L., 2, 15
- Even block search, 27
- Evolution, cultural, 244
- Evolution, organic, 1, 3, 100, 102, 105, 106, 109, 142, 153, 237, 238

- Evolution strategy, 3, 6, 7, 16, 105–151, 168, 173, 175, 177, 179, 200, 203, 210, 213, 219, 228–230, 232–235, 248, 333, 337, 350, 354, 355, 359, 361, 364, 365, 367, 413, see also two membered and multimembered evolution strategies
- Evolution strategy, asynchronous parallel, 248
- Evolution strategy, parallel, 248
- Evolution strategy, 1/5 success rule, 110, 112, 114, 116, 118, 142, 200, 213–215, 237, 349, 361
- Evolution strategy (1+1), 105–119, 125, 163, 177, 185, 191, 200, 203, 212, 213, 216, 217, 228, 231–233, 328, 349, 363
- Evolution strategy (1+ λ), 123, 134, 145
- Evolution strategy (1, λ), 145
- Evolution strategy (10, 100), 177, 186, 191, 200, 203, 211–215, 217, 228, 231–233
- Evolution strategy (μ +1), 119
- Evolution strategy (μ + λ), 119
- Evolution strategy (μ , λ), 119, 145, 148, 238, 244, 248
- Evolution strategy (μ , κ , λ), 247
- Evolution, synthetic theory, 106
- Evolutionary algorithms, 151, 152, 161
- Evolutionary computation, 152
- Evolutionary operation, see EVOP method
- Evolutionary principles, 3, 100, 106, 118, 146, 244
- Evolutionary programming, 151
- Evolutionism, 244
- EVOP method, 6, 7, 9, 64, 68, 69, 89, 101
- Experimental optimization, 6–9, 36, 44, 68, 89, 91, 92, 95, 110, 113, 245, 247, see also design and analysis of experiments
- Expert system, 248
- Extreme value controller, see optimizer
- Extremum, see minimum
- Faber, M.M., 18
- Fabian, V., 20, 90
- Factorial design, 38, 58, 65, 68, 246
- Faddejew, D.K. and Faddejewa, W.N., 27, 67, 240
- Faggioli, E., 96
- Falkenhausen, K. von, 246
- Favreau, R.F., 95, 96, 98, 100
- Feasible region, 8, 9, 12, 16, 17, 25, 101
- Feasible region, not connected, 217, 239, 360
- Feasible starting point, search for, 62, 91, 115
- Feistel, R., 102, 163
- Feldbaum, A.A., 6, 9, 88–90, 99
- Fend, F.A., 48
- Fiacco, A.V., 16, 76, 81, 115, see also SUMT method
- Fibonacci division, 29–32, 38, 177, 178, 181, 187, 192, 200, 202
- Fielding, K., 83
- Finiteness of a sequence of iterations, 68, 166, 172
- Finkelstein, J.J., 18
- Fisher, R.A., 7
- Fletcher, R., 24, 38, 68–71, 74, 77, 80–84, 97, 170, 171, 204, 205, 335, 349
- Fletcher-Powell strategy, see DFP strategy
- Fletcher-Reeves strategy, 69, 70, 78, 170–172, 204, 233, see also conjugate gradients
- Flood, M.M., 68, 89
- Floudas, C.A., 91
- Fogarty, L.E., 68
- Fogel, D.B., 151
- Fogel, L.J., 102, 105, 151
- Forrest, S., 152
- Forsythe, G.E., 34, 66, 67
- Fox, R.L., 23, 34, 205
- Frankhauser, P., 246
- Frankovič, B., 9
- Franks, R., 95, 96, 98, 100
- Fraser, A.S., 152

- Friedberg, R.M., 102, 152
Friedmann, M., 41
Fu, K.S., 94, 99
Function space, 10
Functional analysis theory, 11
Functional optimization, 10–12, 15, 23, 54, 68, 70, 85, 89, 90, 151, 174
Fürst, H., 98
- Gaede, K.W., 8, 108, 144
Gaidukov, A.L., 98
Gal, S., 31
Galar, R., 102
Game theory, 5, 6, 20
Garfinkel, R.S., 18
Gauss, C.F., 41, 84
Gauss-Newton method, 84
Gauss-Seidel strategy, *see* coordinate strategy
Gaussian approximation, *see* sum of squares minimization
Gaussian distribution, *see* normal distribution
Gaussian elimination, 14, 75, 172
Gaviano, M., 96
Gelatt, C.D., 160
Gelfand, I.M., 89
Gene duplication and deletion, 247
Gene pool, 146, 148
Generalized least squares, 84
Genetic algorithms, 151–160
Genetic code, 153, 154, 243
Genotype, 106, 152, 153, 157
Geoffrion, A.M., 24
Geometric programming, 14
Gérardin, L., 105
Gersht, A.M., 90
Gessner, P., 11
Gibson, J.E., 88, 90
Gilbert, E.G., 68, 89
Gilbert, H.D., 90, 98
Gilbert, P., 239
Gill, P.E., 81
Ginsburg, T., 43, 69
- Girsanov, I.V., 11
Glass, H., 48, 87
Glaß, K., 105
Glatt, C.R., 68
Global convergence, 39, 88, 94, 96, 98, 117, 118, 149, 216, 217, 238, 239
Global minimum, 24–26, 90, 168, 329, 344, 348, 356, 357, 359, 360
Global optimization, 19, 29, 84, 88–91, 236, 244
Global penalty function, 16
Glover, F., 162, 163
Gnedenko, B.W., 137
Goldberg, D.E., 152, 154
Golden section, 32, 33, 177, 178, 181, 187, 192, 200, 202
Goldfarb, D., 81
Goldfeld, S.M., 76
Goldstein, A.A., 66, 67, 76, 81, 88
Goliński, J., 92
Goll, R., 244
Golub, G.H., 57, 84
Gomory, R.E., 18
Gonzalez, R.S., 95
Gorges-Schleuter, M., 159, 247
Gorvits, G.G., 174
GOSPEL program package, 68
Goto, K., 82
Gottfried, B.S., 23
Gould, S.J., 148
Gradient strategies, 6, 15, 19, 37, 40, 65–69, 88–90, 94, 95, 98, 166, 167, 171, 172, 174, 235
Gradient strategies, second order, *see* Newton strategies
Gradstein, I.S., 136
Gram-Schmidt orthogonalization, 48, 53, 54, 57, 69, 177, 178, 183, 188, 194, 201, 202, 209, 229, 230, 362
Gran, R., 88
Graphical methods, 20
Grassé, P.P., 243
Grassmann, P., 100
Grauer, M., 20

- Graves, R.L., 23
 Great deluge algorithm, 164
 Greedy algorithm, 162, 248
 Greenberg, H., 18
 Greenberg, H.-J., 162
 Greenstadt, J., 70, 76, 81, 83, 326
 Grefenstette, J.J., 152
 Grid method, 12, 26, 27, 32, 38, 39, 65, 92, 93, 100, 149, 168, 236
 GROPE program package, 68
 Guilfoyle, G., 38
 Guin, J.A., 64
 Gurin, L.S., 89, 97, 98

 Hadamard, J., 66
 Hadley, G., 12, 17, 166
 Haeckel strategy, 163
 Haefner, K., 103
 Hague, D.S., 68
 Haimes, Y.Y., 10
 Hamilton, P.A., 77
 Hamilton, W.R., 15
 Hammel, U., 245, 248
 Hammer, P.L., 19
 Hammersley, J.M., 93
 Hamming cliffs, 154, 155
 Hancock, H., 14
 Handscomb, D.C., 93
 Hansen, P.B., 239
 Haploidy, 148
 Harkins, A., 89
 Harmonic division, 32
 Hartmann, D., 151, 246
 Haubrich, J.G.A., 68
 Heckler, R., 246, 248
 Heidemann, J.C., 70
 Heinhold, J., 8, 108, 144
 Hemstitching, 16
 Henn, R., 20
 Herdy, M., 164
 Hermitian interpolation, 37, 38, 69, 77, 88
 Herschel, R., 99
 Hertel, H., 105

 Hesse, R., 88
 Hessian matrix (Hesse, L.O.), 13, 69, 75, 169, 170
 Hestenes, M.R., 11, 14, 69, 70, 81, 172
 Heuristic methods, 7, 18, 40, 88, 91, 98, 102, 162, 173
 Heusener, G., 245
 Hext, G.R., 57, 58, 64, 68, 89
 Heydt, G.T., 93, 98, 99
 Heynert, H., 105
 Hilbert, D., 10, 11
 Hildebrand, F.B., 66
 Hill climbing strategies, 23ff, 85, 87
 Hill, I.D., 33, 178
 Hill, J.C., 88, 90
 Hill, J.D., 94
 Himmelblau, D.M., 23, 48, 81, 87, 174, 176, 229, 339
 Himsworth, F.R., 57, 58, 64, 68, 89
 History vector method, 98
 Hit-or-miss method, 93
 Ho, Y.C., 68
 Hock, W., 174
 Hodáňová, D., 106
 Hoffmann, U., 23, 74
 Hoffmeister, F., 151, 234, 246, 248
 Höfler, A., 151, 246
 Hofmann, H., 23, 74
 Holland, J.H., 105, 152, 154
 Hollstien, R.B., 152
 Holst, W.R., 67
 Homeostat, 9, 91, 100
 Hoo, S.K., 88
 Hooke, R., 44, 87, 90, 92
 Hooke-Jeeves strategy, 44–48, 87, 90, 177, 178, 182, 188, 193, 200, 202, 210, 228, 230, 233, 332, 339
 Hopper, M.J., 178
 Horner, computational scheme of, 14
 Horst, R., 91
 Hoshino, S., 57, 81
 Hotelling, H., 36
 House, F.R., 77
 Householder, A.S., 27, 75

- Householder method, 57
Houston, B.F., 48
Howe, R.M., 68
Hu, T.C., 18
Huang, H.Y., 70, 78, 81, 82
Huberman, B.A., 103
Huelsman, L.P., 68
Huffman, R.A., 48
Hull, T.E., 93
Human brain, 6, 102
Humphrey, W.E., 67
Hunter, J.S., 65
Hupfer, P., 92, 94, 98
Hurwicz, L., 17, 18, 165
Hutchinson, D., 61
Hwang, C.L., 20
Hybrid computers, 12, 15, 68, 89, 99, 236
Hybrid methods, 38, 162–164, 169
Hyperplane annealing, 162
Hyslop, J., 206
- Idelsohn, J.M., 93, 94
Illiac IV, 239
Imamura, H., 89
Indirect optimization, 13–15, 27, 35, 75, 170, 235
Indusi, J.P., 87
Infimum, 9
Information theory, 5
Integer optimization, 18, 247
Interior point method, 166
Intermediary recombination, 148, 153, 156
Interpolation methods, 14, 27, 33–38
Interval division methods, 27, 29–33, 41
Invention, 2
Inverse Hessian matrix, 77, 78
Inversion of a matrix, 76, 170, 175
Isolation, 106, 244
Iterative methods, 11, 13
Ivakhnenko, A.G., 102
- Jacobi, C.G.J., 15
Jacobi method, 65, 326
Jacobian matrix, 16, 84
Jacobson, D.H., 12
Jacoby, S.L.S., 23, 67, 174
James, F.D., 33, 178
Janáč, K., 90
Jarratt, P., 34, 35, 84
Jarvis, R.A., 91, 93, 94, 99
Jeeves, T.A., 44, 84, 87, 90, 92, see also Hooke-Jeeves strategy
Johannsen, G., 99
John, F., 166
John, P.W.M., 7
Jöhnk, M.D., 115
Johnson, I., 38
Johnson, M.P., 81
Johnson, S.M., 31, 32
Jones, A., 84
Jones, D.S., 81
Jordan, P., 109
- Kamiya, A., 100
Kammerer, W.J., 70
Kantorovich, L.V., 66, 67
Kaplan, J.L., 64
Kaplinskii, A.I., 90
Kappler, H., 18, 166
Karmarkar, N., 166, 167
Karnopp, D.C., 93, 94, 96
Karp, R.M., 239
Karplus, W.I., 12, 89
Karr, C.L., 160
Karreman, H.F., 11
Karumidze, G.V., 94
Katkovnik, V.Ya., 88, 90
Kaupe, A.F., Jr., 39, 44, 178
Kavanaugh, W.P., 95, 98, 99
Kawamura, K., 70
Keeney, R.E., 20
Kelley, H.J., 15, 68, 70, 81
Kempthorne, O., 7, 67, 68
Kenworthy, I.C., 69
Kesten, H., 20
Kettler, P.C., 82, 83
Khachiyan, L.G., 166, 167
Khovanov, N.V., 96, 102

- Khurgin, Ya.I., 89
 Kiefer, J., 19, 29, 31, 32, 178
 Kimura, M., 239
 King, R.F., 35
 Kirkpatrick, S., 160
 Kitajima, S., 94
 Kivelidi, V.Kh., 89
 Kiwiel, K.C., 19
 Kjellström, G., 98
 Klerer, M., 24
 Klessig, R., 15, 70
 Klimenko, E.S., 94
 Klingman, W.R., 48, 87
 Klockgether, J., 7, 245
 Klötzler, R., 11
 Kobelt, D., 246
 Koch, H.W., 244
 Kopp, R.E., 18
 Korbut, A.A., 18
 Korn, G.A., 12, 24, 89, 93, 99
 Korn, T.M., 12, 89
 Korst, J., 161
 Kosako, H., 99
 Kovács, Z., 80, 179
 Kowalik, J.S., 23, 42, 67–69, 84, 174, 334, 335, 345
 Koza, J., 152
 Krallmann, H., 246
 Krasnushkin, E.V., 99
 Krasovskii, A.A., 89
 Krasulina, T.P., 20
 Krauter, G.E., 246
 Kregting, J., 93
 Krelle, W., 17, 18, 166
 Krolak, P.D., 38
 Kuester, J.L., 18, 58, 179
 Kuhn, H.W., 17, 166
 Kuhn-Tucker theorem, 17, 166
 Kulchitskii, O.Yu., 90
 Kumar, K.K., 160
 Künzi, H.P., 17, 18, 20, 166
 Kursawe, F., 102, 148, 245, 248
 Kushner, H.J., 20, 90
 Kussul, E., 101
 Kwakernaak, H., 89
 Kwasnicka, H. and Kwasnicki, W., 102
 Kwatny, H.G., 90
 Laarhoven, P.J.M. van, 161
 Lagrange multipliers, 15, 17
 Lagrange, J.L., 2, 15
 Lagrangian interpolation, 27, 35–37, 41, 56, 64, 73, 80, 89, 101, 177, 182, 187, 193, 200, 202
 Lam, L.S.-B., 100
 Lance, G.M., 54
 Land, A.H., 18
 Lange-Nielsen, T., 54
 Langguth, V., 89
 Langton, C.G., 103
 Lapidus, L., 68
 Larichev, O.I., 174
 Larson, R.E., 239
 Lasdon, L.S., 70
 Lattice search, see grid method
 Laußermair, T., 162
 Lavi, A., 23, 48, 93
 Lawler, E.L., 160
 Lawrence, J.P., 87, 98
 Learning (and forgetting), 9, 54, 70, 78, 98, 101, 103, 162, 236
 Least squares method, see sum of squares minimization
 LeCam, L.M., 102
 Lee, R.C.K., 11
 Lehner, K., 248
 Leibniz, G.W., 1
 Leitmann, G., 11, 18
 Lemarechal, C., 19
 Leon, A., 68, 89, 174, 337, 356
 Leonardo of Pisa, 29
 Lerner, A.Ja., 11
 Leśniak, Z.K., 92
 Lethal mutation, 115, 136, 137, 158
 Levenberg, K., 66, 84
 Levenberg-Marquardt method, 84
 Levine, L., 65
 Levine, M.D., 10

- Levy, A.V., 70, 78, 81
Lew, A.Y., 96
Lew, H.S., 100
Lewallen, J.M., 174
Lewandowski, A., 20
Leyßner, U., 151, 246
Lilienthal, O., 238
Lill, S.A., 80, 178, 179
Lindenmayer, A., 103
Line search, 25–38, 42, 54, 66, 70, 71, 77, 89, 101, 167, 170, 171, 173, 180, 214, 228, see also interval division and interpolation methods
Linear convergence, 34, 168, 169, 172, 173, 236, 365
Linear model objective function, 96, 124–127
Linear programming, 17, 57, 88, 100, 101, 151, 166, 212, 235, 353
Little, W.D., 93, 244
Lobac, V.P., 89
Local minimum, 13, 23–26, 88, 90, 329
Locker, A., 102
Log-normal distribution, 143, 144, 150
Loginov, N.V., 90
Lohmann, R., 164
Long step methods, 66
Longest step procedure, 66
Lootsma, F.A., 24, 81, 174
Lowe, C.W., 69, 101
Lucas, E., 32
Luce, A.D., 21
Luenberger, D.G., 18
Luk, A., 101
Lyvers, H.I., 16
MacDonald, J.R., 84
MacDonald, P.A., 48
Machura, M., 54, 179
MacLane, S., 48
MacLaurin, C., 13
Madsen, K., 35
Mamen, R., 81
Manderick, B., 152
Mandischer, M., 160
Mangasarian, O.L., 18, 24
Männer, R., 152
Marfeld, A.F., 6
Markwich, P., 246
Marquardt, D.W., 84
Marti, K., 118
Masters, C.O., 61
Masud, A.S.M., 20
Mathematical biosciences, 102
Mathematical optimization, 6–9
Mathematical programming, 15–17, 23, 85, see also linear, quadratic, and non-linear programming
Mathematization, 102
Matthews, A., 76, 81
Matyas, J., 97–99, 240, 338
Maximum likelihood method, 8
Maximum, see minimum
Maybach, R.L., 97
Mayne, D.Q., 12, 81
Maze method, 44
McArthur, D.S., 92, 94, 98
McCormick, G.P., 16, 67, 70, 76, 78, 81, 82, 88, 115, 348, see also SUMT method
McGhee, R.B., 65, 68, 89, 93
McGlade, J.M., 102
McGrew, D.R., 10
McGuire, M.R., 239
McMillan, C., Jr., 18
McMurtry, G.J., 94, 99
Mead, R., 58, 84, 97, see also simplex strategy
Medvedev, G.A., 89, 99
Meerkov, S.M., 94
Meissinger, H.F., 99
Meliorization, 1
Memory gradient method, 70
Meredith, D.L., 160
Merzenich, W., 101
Metropolis, N., 160
Meyer, J.-A., 103
Michalewicz, Z., 152, 159
Michel, A.N., 70

- Michie, D., 102
 Mickey, M.R., 58, 89, 95
 Midpoint method, 33
 Miele, A., 68, 70
 Mifflin, R., 19
 Migration, 106, 248
 Miller, R.E., 239
 Millstein, R.E., 239
 Milyutin, A.A., 11
 Minima and maxima, theory of, see optimality conditions
 Minimax concept, 26, 27, 31, 34, 92
 Minimum, 8, 13, 16, 24, 36
 Minimum χ^2 method, 8
 Minot, O.N., 102
 Minsky, M., 102
 Miranker, W.L., 233, 239
 Missing links, 1
 Mitchell, B.A., Jr., 99
 Mitchell, R.A., 64
 Mixed integer optimization, 18, 164, 243
 Mize, J.H., 18
 Mlynski, D., 69, 89
 Mockus, J.B., see Motskus, I.B.
 Model, internal (of a strategy), 9, 10, 28, 38, 41, 90, 169, 204, 231, 235–237
 Model, mathematical (of a system), 7, 8, 65, 68, 160, 235
 Modified Newton methods, 76
 Moler, C., 87
 Moment rosetta search, 48
 Monro, S., 19
 Monte-Carlo methods, 92–94, 109, 149, 160, 168
 Moran, P.A.P., 101
 Moré, J.J., 81, 179
 Morgenstern, O., 6
 Morrison, D.D., 84
 Morrison, J.F., 334
 Motskus, I.B., 88, 94
 Motzkin, T.S., 67
 Movshovich, S.M., 96
 Mufti, I.H., 18
 Mugele, R.A., 44
 Mühlenbein, H., 163
 Mulawa, A., 54, 179
 Muller, M.E., 115
 Müller, P.H., 98
 Müller-Merbach, H., 17, 166
 Multicellular individuals, 247
 Multidimensional optimization, 2, 38ff, 85
 Multimembered evolution strategy, 101, 103, 118–151, 153, 158, 235–248, 329, 333, 335, 344, 347, 355–357, 359, 360, 362, 363, 365, 366, 375, 413, see also evolution strategy (μ, λ) and $(\mu + \lambda)$
 Multimodality, 12, 24, 85, 88, 157, 159, 239, 245, 248
 Multiple criteria decision making (MCDM), 2, 20, 148, 245
 Munson, J.K., 95
 Murata, T., 44
 Murray, W., 24, 76, 81, 82
 Murtagh, B.A., 78, 82
 Mutation, 3, 100–102, 106–108, 154, 155, 237
 Mutation rate, 100, 101, 154, 237
 Mutator genes, 142, 238
 Mutseniyeks, V.A., 99
 Myers, G.E., 70, 78, 81
 Nabla operator, 13
 Nachtigall, W., 105
 Nag, A., 58
 Nike, F., 49
 Narendra, K.S., 94
 Nashed, M.Z., 70
 Neave, H.R., 116
 Neighborhood model, 247
 Nelder, J.A., 58, 84, 97
 Nelder-Mead strategy, see simplex strategy
 Nemhauser, G.L., 18
 Nenonen, L.K., 70
 Network planning, 20
 Neumann, J. von, 6
 Neustadt, L.W., 11, 18
 Newman, D.J., 39

- Newton, I., 2, 14
 Newton direction, 70, 75–77, 84
 Newtonian interpolation, 27, 35
 Newton-Raphson method, 35, 75, 76, 97, 167, 169–171
 Newton strategies, 40, 71, 74–85, 89, 171, 235
 Neyman, J., 102
 Niching, 100, 106, 238, 248
 Nicholls, R.L., 23
 Nickel, K., 168
 Niederreiter, H., 115
 Niemann, H., 5
 Nikolić, Ž.J., 94
 Nissen, V., 103
 Nollau, V., 98
 Non-linear programming, 17, 18, 166
 Non-smooth or non-differentiable optimization, 19
 Nonstationary optimum, 248
 Norkin, K.B., 88
 Normal distribution, 7, 90, 94, 95, 97, 101, 108, 116, 120, 128, 153, 236, 240, 243
 North, J.H., 102
 North, M., 246
 Numerical mathematics, 5, 27, 239
 Numerical optimization, see direct optimization
 Nurminski, E.A., 19
 Objective function, 2, 8
 Observational calculus, 5, 7
 Odd block search, 27
 Odell, P.L., 20
 Oettli, W., 18, 167
 O'Hagan, M., 48, 87
 Oi, K., 82
 Oldenburger, R., 9
 Oliver, L.T., 31
 One dimensional optimization, 25–38, see also line search
 One step methods, see relaxation methods
 O'Neill, R., 58, 179
 Ontogenetic learning, 163
 Opačić, J., 88
 Operations research, 5, 17, 20
 Optimal control, see control theory
 Optimality conditions, 2, 13–15, 23, 167ff, 235
 Optimality of organic systems, 99, 100, 105
 Optimization, prerequisites for, 1
 Optimization problem, 2, 5–8, 14, 20, 24
 Optimizer, 9, 10, 48, 99, 248
 Optimum, see minimum
 Optimum, maintaining (and hunting), see dynamic optimization
 Optimum gradient method, 66
 Optimum principle of Bellman, 11, 12
 Oren, S.S., 82
 Ortega, J.M., 5, 27, 41, 42, 82, 84
 Orthogonalization, see Gram-Schmidt and Palmer orthogonalization
 Osborne, M.R., 23, 42, 68, 69, 84, 174, 335, 345
 Osche, G., 106, 119
 Ostermeier, A., 118
 Ostrowski, A.M., 34, 66
 Overadaptation, 148
 Overholt, K.J., 31–33, 178
 Overrelaxation and underrelaxation, 43, 67
 Owens, A.J., 102, 105, 151
 Page, S.E., 155
 Pagurek, B., 70
 Palmer, J.R., 57, 178
 Palmer orthogonalization, 57, 177, 178, 183, 188, 194, 202, 209, 230
 Papageorgiou, M., 23
 Papentin, F., 102
 Parallel computers, 161, 163, 234, 239, 243, 245, 247, 248
 Parameter optimization, 6, 8, 10–13, 15, 16, 20, 23, 105
 Parameterization, 15, 151, 346
 Pardalos, P.M., 91
 Pareto-optimal, 20, 245

- Parkinson, J.M., 61
 Partan (parallel tangents) method, 67–69
 Pask, G., 101
 Path-oriented strategies, 98, 160, 236, 248
 Patrick, M.L., 239
 Pattern recognition, 5
 Pattern search, see Hooke-Jeeves strategy
 Paviani, D.A., 87
 Pearson, J.D., 38, 70, 76, 78, 81, 82, 205
 Peckham, G., 84
 Penalty function, 15, 16, 48, 49, 57, 207
 Perceptron, 102
 Peschel, M., 20
 Peters, E., 163, 248
 Peterson, E.L., 14
 Phenotype, 106, 153–155, 157, 158
 Pierre, D.A., 23, 48, 68, 95
 Pierson, B.L., 82
 Pike, M.C., 33, 44, 178
 Pincus, M., 93
 Pinkham, R.S., 93
 Pinsker, I.Sh., 44
 Pixner, J., 33, 178
 Pizzo, J.T., 23, 67, 174
 Plane, D.R., 18
 Plaschko, P., 151, 246
 Pleiotropy, 243
 Pluznikov, L.N., 94
 Polak, E., 15, 18, 70, 76, 77, 167, 169
 Policy, 11
 Polyak, B.T., 70
 Polygeny, 243
 Polyhedron strategies, see simplex and complex strategies
 Ponstein, J., 17
 Pontrjagin, L.S., 18
 Poor man's optimizer, 44
 Population principle, 101, 119, 238
 Posynomes, 14
 Powell, D.R., 84
 Powell, M.J.D., 57, 70, 71, 74, 77, 82, 84, 88, 97, 170, 202, 205, 335, 337, 349, see also DFP, DFP-Stewart, and Powell strategies
 Powell, S., 18
 Powell strategy, 69–74, 88, 163, 170–172, 177, 178, 183, 189, 195, 200, 202, 204, 209, 210, 219, 228–230, 327, 332, 339, 341, 343, 364
 Poznyak, A.S., 90
 Practical algorithms, 167
 Predator-prey model, 247
 Press, W.H., 115
 Price, J.F., 76, 81, 88
 Probabilistic automaton, 94
 Problem catalogue, see catalogue of problems
 Process computers, 10
 Projected gradient method, 57, 70
 Proofs of convergence, 42, 47, 66, 77, 97, 167, 168
 Propoi, A.I., 90
 Prusinkiewicz, P., 103
 Pseudo-random numbers, see random number generation
 Pugachev, V.N., 95
 Pugh, E.L., 89
 Pun, L., 23
 Punctuated equilibrium, 148
 Pure random search, 91, 92, 100, 237
 Q-properties, 169, 170, 172, 179, 243
 Quadratic convergence, 68, 69, 74, 76, 78, 81–83, 168, 169, 200, 202, 236
 Quadratic interpolation, see Lagrangian and Hermitian interpolation
 Quadratic programming, 166, 233, 235
 Quandt, R.E., 76
 Quasi-Newton method, 37, 70, 76, 83, 89, 170, 172, 205, 233, 235, see also DFP and DFP-Stewart strategies
 Rabinowitz, P., 84
 Raiffa, H., 20, 21
 Rajtora, S.G., 82
 Ralston, A., 27
 Random direction, 20, 88, 90, 98, 101, 202
 Random evolutionary operation, see REVOP method

- Random exchange step, 88, 166
 Random number generation, 115, 150, 210, 212, 217, 237
 Random sequence, 87, 93
 Random step length, 95, 96, 108
 Random strategies, 3, 12, 19, 87–103, 105, 240
 Random walk, 247
 Randomness, 87, 91, 93, 237
 Rank one methods, 82, 83, 172
 Raphson, J., see Newton-Raphson method
 Rappl, G., 118
 Raster method, see grid method
 Rastrigin, L.A., 93, 95, 96, 98, 99
 Rate of convergence, 7, 38, 39, 64, 66, 67, 69, 90, 94–98, 101, 110, 118, 120–141, 167–169, 179–204, 217–232, 234, 236, 239, 240, 242, see also linear and quadratic convergence
 Rauch, S.W., 82
 Rawlins, G.J.E., 152
 Rayleigh-Ritz method, 15
 Razor search, 48
 Rechenberg, I., 6, 7, 97, 100, 105, 107, 118–120, 130, 142, 149, 164, 168, 172, 179, 231, 238, 245, 352
 Recognition processes, 102
 Recombination, 3, 101, 106, 146–148, 153–159, 186, 191, 200, 203, 204, 211–213, 215–217, 228, 231, 232, 240, 335, 355, 357, 363, 365, 366, see also discrete and intermediary recombination
 Reeves, C.M., 38, 69, 93, 170, 204, see also Fletcher-Reeves strategy
 References, 249–323
 Regression, 8, 19, 84, 235, 246
 Regression, non-linear, 84
 Regula falsi (falsorum), 27, 34, 35, 39
 Reid, J.K., 66
 Rein, H., 100
 Reinsch, C., 14
 Relative minimum, 38, 42, 43, 66, 209
 Relaxation methods, 14, 20, 41, 172, see also coordinate strategy
 Reliability, see robustness
 Repair enzymes, 142, 238
 Replicator algorithm, 163
 Restart of a search, 61, 67, 70, 71, 88, 89, 169, 201, 202, 205, 210, 219, 228–230, 362, 364
 REVOP method, 101
 Reynolds, O., 238
 Rhead, D.G., 74
 Rheinboldt, W.C., 5, 27, 41, 42, 82, 84
 Ribière, G., 70, 82
 Rice, J.R., 57
 Richardson, D.W., 155
 Richardson, J.A., 58, 179
 Richardson, M., 239
 Riding the constraints, 16
 Riedl, R., 102, 153
 Ritter, K., 24, 70, 82, 88, 168
 Rivlin, L., 48
 Robbins, H., 19
 Roberts, P.D., 70
 Roberts, S.M., 16
 Robots, 6, 9, 103
 Robustness, 3, 13, 34, 37–39, 53, 61, 64, 70, 90, 94, 118, 178, 204–217, 236, 238
 Rockoff, M.L., 41
 Rodloff, R.K., 246
 Rogson, M., 100
 Roitblat, H., 103
 Rosen, J.B., 18, 24, 57, 91, 352
 Rosen, R., 100
 Rosenblatt, F., 102
 Rosenbrock, H.H., 23, 29, 48, 50, 54, 343, 349
 Rosenbrock strategy, 16, 48–54, 64, 177, 179, 184, 190, 196, 201, 202, 207, 209, 212, 213, 216, 228, 230–232, 357, 363, 365, 366
 Rosenman, E.A., 11
 Ross, G.J.S., 84

- Rotating coordinates method, see Rosenbrock and DSC strategies
- Rothe, R., 25
- Roughgarden, J.W., 102
- Rounding error, see accuracy of computation
- Rozonoer, L.I., 90
- Rozvany, G., 247
- Ruban, A.I., 99
- Rubin, A.I., 95
- Rubinov, A.M., 11
- Rudd, D.F., 98, 356
- Rudelson, L.Ye., 102
- Rudolph, G., 91, 118, 134, 151, 154, 161, 162, 241, 243, 248
- Rustay, R.C., 68, 89
- Rutishauser, H., 5, 41, 43, 48, 65, 75, 172, 326
- Rybashov, M.V., 68
- Ryshik, I.M., 136
- Saaty, T.L., 20, 27, 166
- Sacks, J., 20
- Saddle point, 13, 14, 17, 23, 25, 35, 36, 39, 66, 76, 88, 168, 176, 209, 211, 345
- Salaff, S., 100
- Sameh, A.H., 239
- Samuel, A.L., 102
- Sargent, R.W.H., 78, 82
- Saridis, G.N., 90, 98
- Satterthwaite, F.E., 98, 101
- Saunders, M.A., 57
- Savage, J.M., 119
- Savage, L.J., 41
- Sawaragi, Y., 48
- Sayama, H., 82
- Scaling of the variables, 7, 44, 54, 58, 74, 146–148, 232, 239
- Schaffer, J.D., 152
- Schechter, R.S., 15, 23, 28, 32, 37, 41–43, 64, 65
- Scheeffer, L., 14
- Scheel, A., 118
- Schema theorem, 154
- Scheraga, H.A., 89
- Scheuer, E.M., 241
- Scheuer, T., 98, 164
- Schinzinger, R., 67
- Schittkowski, K., 174
- Schley, C.H., Jr., 70
- Schlierkamp-Voosen, D., 163
- Schmalhausen, I.I., 101
- Schmetterer, L., 90
- Schmidt, E., see Gram-Schmidt orthogonalization
- Schmidt, J.W., 35, 39
- Schmitt, E., 20, 90
- Schmutz, M., 103
- Schneider, G., 246
- Schneider, M., 100
- Schrack, G., 98, 240
- Schumer, M.A., 89, 93, 96–99, 101, 200, 240
- Schuster, P., 101
- Schwarz, H.R., 5, 41, 65, 75, 172, 326
- Schwefel, D., 246
- Schwefel, H.-P., 7, 102, 103, 118, 134, 148, 151, 152, 155, 163, 204, 234, 239, 242, 245–248
- Schwetlick, H., 39
- Scott, E.L., 102
- Sebald, A.V., 151
- Sebastian, D.J., 82
- Sebastian, H.-J., 24
- Secant method, 34, 39, 84
- Second order gradient strategies, see Newton strategies
- Sectioning algorithms, 14
- Seidel, P.L., 41, see also coordinate strategy
- Selection, 3, 100–102, 106, 142, 153, 157
- Sensitivity analysis, 17
- Separable objective function, 12, 42
- Sequential methods, 27ff, 38ff, 88, 237
- Sequential unconstrained minimization technique, see SUMT method
- Sergiyevskiy, G.M., 69
- Sexual propagation, 3, 101, 106, 146, 147

- Shah, B.V., 67, 68
Shanno, D.F., 76, 82–84
Shapiro, I.J., 94
Shedler, G.S., 239
Shemeneva, V.V., 95
Shimelevich, L.I., 88
Shimizu, T., 92
Shindo, A., 64
Short step methods, 66
Shrinkage random search, 94
Shubert, B.O., 29
Sigmund, K., 21
Silverman, G., 84
Simplex method, *see* linear programming
Simplex strategy, 57–61, 64, 84, 89, 97, 177, 179, 184, 190, 196, 201, 202, 208, 210, 228–231, 341, 361–364
Simplex, 17, 58, 353
Simulated annealing, 160–162
Simulation, 13, 93, 102, 103, 152, 245, 246
Simultaneous methods, 26–27, 92, 168, 237
Singer, E., 44
Single step methods, *see* relaxation methods
Singularity, 70, 74, 78, 82, 205, 209
Sirisena, H.R., 15
Slagle, J.R., 102
Slezak, N.L., 241
Smith, C.S., 54, 71, 74
Smith, D.E., 174
Smith, F.B., Jr., 84
Smith, J. Maynard, 21, 102
Smith, L.B., 44, 178
Smith, N.H., 98, 356
Soeder, C.-J., 103
Somatic mutations, 247
Sondak, N.E., 66, 89
Sonderquist, F.J., 48
Sorenson, H.W., 68, 70
Southwell, R.V., 20, 41, 43, 65
Spang, H.A., 93, 174
Späth, H., 84
Spears, W., 152
Spedicato, E., 82
Spendley, W., 57, 58, 61, 64, 68, 84, 89
Speyer, J.L., 70
Sphere model objective function, 110, 117, 120, 123, 124, 127–134, 142, 173, 179, 203, 215, 325, 338
Spider method, 48
Sprave, J., 247, 248
Spremann, K., 11
Stagnation, 47, 58, 61, 64, 67, 87, 88, 100, 157, 201, 205, 238, 341
Standard deviation, *see* variance
Stanton, E.L., 34
Stark, R.M., 23
Static optimization, 9, 10
Stebbins, G.L., 106
Steepest descent/ascent, 66–68, 166, 169, 235
Steiglitz, K., 87, 96, 98, 99, 101, 200, 240
Stein, M.L., 14, 67
Steinberg, D., 23
Steinbuch, K., 6
Stender, J., 152
Step length control, 110–113, 142–145, 168, 172, 237, *see also* evolution strategy, 1/5 success rule
Steuer, R.E., 20
Stewart, E.C., 95, 98, 99
Stewart, G.W., 78, 84, *see also* DFP-Stewart strategy
Stiefel, E., 5, 41, 43, 65, 67, 69, 75, 172, 326
Stochastic approximation, 19, 20, 64, 83, 90, 94, 99, 236
Stochastic optimization, 18
Stochastic perturbations, 9, 20, 36, 58, 68, 69, 89, 91, 92, 94, 95, 97, 99, 236, 245
Stoer, J., 18
Stoller, D.S., 241
Stolz, O., 14
Stone, H.S., 239
Storage requirement, 47, 53, 57, 180, 232–234, 236

- Storey, C., 23, 50, 54
 Strategy, 2, 6, 100
 Strategy comparison, 57, 64, 68, 71, 78, 80, 83, 84, 92, 97, 165–234
 Strategy parameter, 144, 204, 238, 240–242
 Stratonovich, R.L., 90
 Strong minimum, 24, 328, 333
 Strongin, R.G., 94
 Structural optimization, 247
 Struggle for existence, 100, 106
 Suboptimum, 15
 Subpopulations, 248
 Success/failure routine, 29
 Suchowitzki, S.I., 18
 Sugie, N., 38
 Sum of squares minimization, 5, 83, 331, 335, 346
 SUMT method, 16
 Supremum, 9
 Sutti, C., 88
 Suzuki, S., 352
 Svechinskii, V.B. (Svečinskij, V.B.), 90, 102
 Swann, W.H., 23, 28, 54, 56, 57, see also DSC strategy
 Sweschnikow, A.A., 137
 Sworder, D.D., 83
 Sydow, A., 68
 Sylvester, criterion of, 240
 Synge, J.L., 44
 Sysoyev, V.V., 95
 Szegö, G.P., 24, 70, 88

 Tabak, D., 18, 78, 82
 Tabu search, 162–164
 Tabulation method, see grid method
 Takamatsu, T., 82
 Talkin, A.I., 68
 Tammer, K., 24
 Tan, S.T., 17
 Tapley, B.D., 174
 Taran, V.A., 94
 Taylor, G., 84
 Taylor series (Taylor, B.), 75, 84
 Tazaki, E., 64
 Tchebycheff approximation (Tschebyschow, P.L.), 5, 331, 370
 Termination of the search, 35, 38, 49, 54, 59, 64, 67, 71, 96, 113, 114, 117, 145, 146, 150, 167, 168, 175, 176, 180, 212, 238
 Ter-Saakov, A.P., 69
 Theodicee, 1
 Theory of maxima and minima, 11
 Thom, R., 102
 Thomas, M.E., 20
 Three point scheme, 29
 Threshold strategy, 98, 164
 Tietze, J.L., 70
 Timofejew-Ressowski, N.W., 101
 Todd, J., 326
 Togawa, T., 100
 Tokumaru, H., 82
 Tolle, H., 18, 68
 Tomlin, F.K., 44, 178
 Törn, A., 91
 Total step procedure, 65
 Tovstucha, T.I., 89
 Trabattoni, L., 88
 Trajectory optimization, see functional optimization
 Traub, J.F., 14, 27
 Travelling salesperson problem (TSP), 159, 161
 Treccani, G., 70, 88
 Trial and error, 13, 41
 Trial polynomial, 27, 33–35, 37, 68, 235
 Trinkaus, H.F., 35
 Trotter, H.F., 76
 Tschebyschow, P.L., see Tchebycheff approximation
 Tse, E., 239
 Tseitlin, B.M., 44
 Tsetlin, M.L., 89
 Tsypkin, Ya.Z., 6, 9, 89, 90
 Tucker, A.W., 17, 166
 Tui, H., 88

- Turning point, see saddle point
- Two membered evolution strategy, 97, 101, 105–118, 172, 238, 329, 352, 357, 359, 363, 366, 367, 374, see also evolution strategy (1+1)
- Tzschach, H.G., 18
- Ueing, U., 88, 358, 359
- Umeda, T., 64
- Unbehauen, H., 18
- Uncertainty, interval of, 26–28, 32, 39, 92, 180
- Uniform distribution, 91, 92, 95, 115
- Unimodality, 24, 27, 28, 39, 168, 236
- Uzawa, H., 18
- Vagin, V.N., 102
- Vajda, S., 7, 18
- Vanderplaats, G.N., 23
- VanNice, R.I., 44
- VanNorton, R., 41
- Varah, J.M., 68
- Varela, F.J., 103
- Varga, J., 18
- Varga, R.S., 43
- Variable metric, 70, 77, 83, 169–172, 178, 233, 242, 243, 246, see also DFP and DFP-Stewart strategies
- Variables, 2, 8, 11
- Variance analysis, 8
- Variance ellipse, 109
- Variance methods, 82
- Variational calculus, 2, 11, 15, 66
- Vaysbord, E.M., 90, 94, 99
- Vecchi, M.P., 160
- Venter, J.H., 20
- Vetters, K., 39
- Vilis, T., 10
- Viswanathan, R., 94
- Vitale, P., 84
- Vogelsang, R., 5
- Vogl, T.P., 24, 48, 93
- Voigt, H.-M., 163
- Voltaire, F.M., 1
- Volume-oriented strategies, 98, 160, 236, 248
- Volz, R.A., 12, 70
- Wacker, H., 12
- Waddington, C.H., 102
- Wagner, K., 151, 246
- Wald, A., 7, 89
- Walford, R.B., 93
- Wallack, P., 68
- Walsh, J., 27
- Walsh, M.J., 102, 105, 151
- Ward, L., 58
- Wasan, M.T., 19
- Wasscher, E.J., 68, 76
- Weak minimum, 24, 25, 113, 328, 332, 333
- Weber, H.H., 17
- Wegge, L., 76
- Weierstrass, K., theorem of, 25
- Weinberg, F., 18, 91
- Weisman, J., 23, 47, 89
- Weiss, E.A., 48
- Wells, M., 77
- Werner, J., 82
- Wets, R.J.-B., 19
- Wetterling, W., 5
- Wheatley, P., 38
- Wheeling, R.F., 95, 98
- White, L.J., 97
- White, R.C., Jr., 93, 95
- Whitley, L.D., 152, 155
- Whitley, V.W., 76
- Whitting, I.J., 84
- Whittle, P., 18
- Wiedemann, J., 151
- Wiener, N., 6
- Wierzbicki, A.P., 20
- Wilde, D.J., 1, 20, 23, 26, 27, 29, 31–33, 38, 39, 87
- Wilf, H.S., 27
- Wilkinson, J.H., 14, 75
- Wilson, E.O., 146
- Wilson, K.B., 6, 65, 68, 89

- Wilson, S.W., 103
Witt, U., 103
Witte, B.F.W., 67
Witten, I.H., 94
Witzgall, C., 18
Wolfe, P., 19, 23, 39, 66, 70, 82, 84, 166,
360
Wolff, W., 103
Wolfowitz, J., 19
Wood, C.F., 47, 48, 89
Woodside, C.M., 70
Wright, S.J., 179
- Yates, F., 7
Youden, W.J., 101
Yudin, D.B., 90, 94, 99
Yvon, J.P., 96
- Zach, F., 9
Zadeh, N., 41
Zahradnik, R.L., 23
Zakharov, V.V., 94
Zangwill, W.I., 18, 41, 66, 71, 74, 170,
202
Zehnder, C.A., 18, 91
Zelevnik, F.J., 84
Zellnik, H.E., 66, 89
Zener, C., 14
Zerbst, E.W., 105
Zero-one optimization, see binary opti-
mization
Zettl, G., 344, 348
Zhigljavsky, A.A., 91
Zigangirov, K.S., 89
Žilinskas, A., 91
Zoutendijk, G., 18, 70
Zurmühl, R., 27, 35, 172
Zwart, P.B., 88
Zypkin, Ja.S., see Tsypkin, Ya.Z.