Appendix A: Functions and Terminals

Table A.1 shows the name and a reference for functions and terminals (i.e., functions of arity 0) used in this book.

Table A.1 Functions and terminals

Name	Arity	Full name	Section in this book
+	2	Numeric addition	2.2.1
−	2	Numeric subtraction	2.2.1
*	2	Numeric multiplication	2.2.1
%	2	Protected numeric division	2.2.1
>	2	Greater-than comparative function	7.2.1
<	2	Less-than comparative function	7.2.1
2N3904	0	2N3904 transistor model	10.5.1.4
2N3906	0	2N3906 transistor model	10.5.1.4
ABS_SIGNAL	1	Insert an absolute value block into a controller	3.5.1
ADD_3_SIGNAL	3	Insert a 3-input addition block into a controller	3.5.1
ADD_SIGNAL	2	Insert an addition block into a controller	3.5.1
ADF	Variable	Automatically defined function	3.5.4
AH	0	Connect to Åström-Hägglund controller	13.1.2
B_C_E, …, E_C_B	0	Transistor lead permutation terminals	10.5.1.4
BIFURCATE_POSITIVE	0	Bifurcate positive end of wire in inserting transistor	10.5.1.4

(*Continued*)

Table A.1 (*Continued*)

Name	Arity	Full name	Section in this book
BIFURCATE_NEGATIVE	0	Bifurcate negative end of wire in inserting transistor	10.5.1.4
C	2	Capacitor-inserting function	4.2.1.3
C-LAYOUT	2	Geographically-aware capacitor-inserting function	5.2.1.3
C_NEW	1	Create a capacitor	4.7.1.3
C00002	0	Concentration of cofactor ATP	8.5.3
C00116	0	Concentration of glycerol	8.5.3
C00162	0	Concentration of fatty acid	8.5.3
C00165	0	Concentration of diacyl-glycerol	8.5.3
CAP_LEVEL	0	Represents gene expression level in a genetic network	7.3.3
CONSTANT_0	0	Connect to time-domain signal with constant 0 value	3.5.2
CONTROLLER_OUTPUT	0	Connect to a controller's output	3.5.2
CR_1_1	4	1-subtrate, 1-product reactor block	8.4.1.1
CR_1_2	5	1-subtrate, 2-product reactor block	8.4.1.1
CR_2_1	5	2-subtrate, 1-product reactor block	8.4.1.1
CR_2_2	6	2-subtrate, 2-product reactor block	8.4.1.1
DELAY	1	Insert a delay block into a controller	3.5.1
DIFFERENTIAL_ INPUT_INTEGRATOR	2	Insert a differential input integrator block into a controller	3.5.1
DIFFERENTIATOR	1	Insert a differentiator block into a controller	3.5.1
DIV_NUMERIC	2	Protected numeric division for controllers	9.1.1.3
DIV_SIGNAL	2	Insert a division block into a controller	3.5.1
DOWN_OR_RIGHT	0	Parallel-divide down or right	10.1.2
DRAW	2	Drawing turtle function	6.3.1
END	0	Development-ending function	4.2.1.4
F	0	Free variable representing passband boundary	10.3.1.3
F1	0	Free variable representing passband boundary	11.1.1.3

(*Continued*)

Table A.1 (*Continued*)

Name	Arity	Full name	Section in this book
F2	0	Free variable representing stopband boundary	11.1.1.3
FIRST_PRODUCT	1	Selects the first product produced by a chemical reaction	8.4.1.1
FLIP	1	Flip (polarity-reversing) function	4.2.1.3
GAIN	2	Insert a gain block into a controller	3.5.1
GLUCOSE_LEVEL	0	Represents glucose level in a genetic network	7.3.3
HFA3046	0	HFA3046 transistor model	15.3.3
HFA3128	0	HFA3128 transistor model	15.3.3
IF	3	If operator	7.2.1
IF_POSITIVE	3	Insert an if-positive block into a controller	3.5.1
IFGTZ_DEVELOPMENTAL	3	If greater than zero developmental operator	11.1
INPUT_0	2	Connect to a circuit's input	4.7.1.1
INTEGRATOR	1	Insert an integrator block into a controller	3.5.1
INT1, INT2, INT3	0	Intermediate substance terminals	8.5.3
INVERTER	1	Insert an inverter block into a controller	3.5.1
IRF511	0	IRF511 transistor model	15.3.3
IRF9230	0	IRF9230 transistor model	15.3.3
IRFZ44	0	IRFZ44 transistor model	15.3.3
KU	0	Free variable representing plant's ultimate gain	9.2.1.2
L	0	Free variable representing plant's dead time	9.2.1.2
L	2	Inductor-inserting function	10.3.1.4
L1	0	Free variable representing inductance of a specific inductor	10.1.2.4
L_NEW	1	Create an inductor	4.7.1.3
L-LAYOUT	2	Geographically aware inductor-inserting function	5.2.1.3
LACTOSE_LEVEL	0	Represents lactose level in a genetic circuit	7.3.3
LAG	2	Insert a lag block into a controller	3.5.1
LAG2	3	Insert a second order lag block into a controller	3.5.1
LANDMARK	1	Landmark turtle function	6.3.1
LEAD	2	Insert a lead block into a controller	3.5.1
LEFT_1, …, LEFT_4	0	Takeoff point reference terminals	13.1.2

(*Continued*)

Table A.1 (*Continued*)

Name	Arity	Full name	Section in this book
LIMITER	3	Insert a limiter block into a controller	3.5.1
LOG_F	0	Free variable representing logarithm of passband boundary for filter	10.5.1.4
MTP50P03HDL	0	MTP50P03HDL transistor model	15.3.3
MULT_SIGNAL	2	Insert a multiplication block into a controller	3.5.1
NODE	2	Connect distant points in a circuit	10.1.1
NOP	1	No-operation function	4.2.1.3
OUTPUT_0	2	Connect to a circuit's output	4.7.1.1
PAIR_CONNECT_0	3	Connects a pair of distant points in a circuit	4.2.1.3
PAIR_CONNECT_1	3	Connects a pair of distant points in a circuit	4.2.1.3
PARALLEL-LAYOUT-LEFT	4	Geographically aware parallel division function	5.2.1.3
PARALLEL-LAYOUT-RIGHT	4	Geographically aware parallel division function	5.2.1.3
PARALLEL0	4	Parallel-division function, version 0	4.2.1.3
PARALLEL1	4	Parallel-division function, version 1	4.2.1.3
PARALLEL_NEW	4	New version of parallel division function	10.1.2
PLANT_OUTPUT	0	Connect to a controller's plant output	3.5.2
POW	2	Power function	12.2.3
PROGN	Variable	Connective function	6.3.1
Q	6	Transistor-inserting function	10.1.5
Q_DIODE_NPN	1	Insert an NPN diode into a circuit	4.4.1.4
Q_DIODE_PNP	1	Insert a PNP diode into a circuit	4.4.1.4
Q_GND_EMIT_NPN	1	Insert an NPN transistor whose emitter is connected to ground	4.4.1.4
Q_GND_EMIT_PNP	1	Insert a PNP transistor whose emitter is connected to ground	4.4.1.4
Q_POS5V_COLL_NPN	1	Insert an NPN transistor whose collector is connected to a 5 V power supply	4.4.1.4
Q_POS5V_EMIT_PNP	1	Insert a PNP transistor whose emitter is connected to a 5 V power supply	4.4.1.4
Q_THREE_NPN0, …, Q_THREE_NPN11	3	Insert an NPN transistor into a circuit	4.4.1.4

(*Continued*)

Table A.1 (*Continued*)

Name	Arity	Full name	Section in this book
Q_THREE_PNP0, …, Q_THREE_PNP11	3	Insert a PNP transistor into a circuit	4.4.1.4
R	2	Resistor-inserting function	4.2.1.3
R1	0	Free variable representing inductance of a specific resistor	10.1.2.4
$\Re$	0	Random floating-point constants in specified range	3.5.5.1
$\Re_{\text{integer}}$	0	Random integer constants between 0 and 99	6.3.2
$\Re_{\mathrm{p}}$	0	Random perturbable floating-point value in specified range	3.5.5.2
$\Re_{\text{real}}$	0	Random floating-point constants between 0.0 and 1.0	6.3.2
R_NEW	1	Create a resistor	4.7.1.3
REFERENCE_SIGNAL	0	Connect to a controller's reference signal	3.5.2
REPEAT	2	Repeat turtle function	6.3.1
REPRESSOR_LEVEL	0	Represents repressor level in a genetic network	7.3.3
RETAINING_ THREE_GROUND0	3	Connect to ground	4.3.1.4
RETAINING_ THREE_GROUND1	3	Connect to ground	4.3.1.4
RETAINING_ THREE_POS5V_0	3	Connect to 5 V power supply	4.4.1.4
RETAINING_ THREE_POS5V_0	3	Connect to 5 V power supply	4.4.1.4
REXP	1	Protected exponential function	9.2.1.3
RIGHT_1, …, RIGHT_4	0	Takeoff point reference terminals	13.1.2
RLOG	1	Protected natural logarithm function	9.2.1.3
SAFE_CUT	0	Safe-cut developmental function	4.2.1.4
SECOND_PRODUCT	1	Selects the second product produced by a chemical reaction	8.4.1.1
SERIES	3	Series division function	4.2.1.3
SERIES-LAYOUT	3	Geographically aware series division function	5.2.1.3
SIN_THETA	0	Free variable representing sine of a filter's modular angle	10.2.1.4
SUB_SIGNAL	2	Insert a subtraction block into a controller	3.5.1
TAKEOFF	1	New takeoff point function	13.1.1
TARGET	0	Free variable representing identity of target circuit (for squaring/cubing problem and 40/60 dB amplifier problem)	11.3.1.4
TAU	0	Free variable representing plant's time constant	9.1.1.2

(*Continued*)

Table A.1 (*Continued*)

Name	Arity	Full name	Section in this book
THREE_GROUND	3	Three-argument connection to ground	10.2.1.3
TU	0	Free variable representing plant's ultimate period	9.2.1.2
TURN-RIGHT	1	Turn right turtle function	6.3.1
TR	0	Free variable representing plant's rise time	9.2.1.2
TWO_GROUND	2	Two-argument connection to ground	10.2.1.3
TWO_LEAD	3	Two-leaded-component-inserting function	10.1.4
TWO_NEG15V	2	−15 Volt reference voltage source function	11.3.1.3
TWO_POS1V	2	+1 Volt reference voltage source function	11.3.1.3
TWO_POS2V	2	+2 Volt reference voltage source function	11.3.1.3
TWO_POS5V	2	+5 Volt reference voltage source function	11.3.1.3
TWO_POS15V	2	+15 Volt reference voltage source function	11.3.1.3
UP_OR_LEFT	0	Parallel-divide up or left	10.1.2
VIA-TO-GROUND-NEG-LEFT-LAYOUT	3	Geographically aware via to ground function	5.2.1.3
VIA-TO-GROUND-NEG-RIGHT-LAYOUT	3	Geographically aware via to ground function	5.2.1.3
VIA-TO-GROUND-POS-T-LAYOUT	3	Geographically aware via to ground function	5.2.1.3
VIA-TO-GROUND-POS-RIGHT-LAYOUT	3	Geographically aware via to ground function	5.2.1.3
X	0	Represents value on the *X*-axis for symbolic regression problems	2.2.1

Appendix B: Control Parameters

Broadly speaking, we have used substantially the same (almost certainly non-optimal) choices of control parameters from problem to problem over a period of years. Although particular problems in this book could possibly be solved more efficiently by means of a different choice of control parameters, we believe that our policy of substantial consistency in the choice of control parameters helps the reader eliminate superficial concerns that the demonstrated success of genetic programming depends on shrewd or fortuitous choices of the control parameters. That is, the results produced by genetic programming are not the fruit of intricate and astute tailoring of control parameters to a particular problem.

This book continues the policy of *Genetic Programming* (Koza 1992a), *Genetic Programming II* (Koza 1994a), and *Genetic Programming III* (Koza, Bennett, Andre, and Keane 1999a) of using a fixed set of default values for most of the minor control parameters throughout the book. Thus, unless otherwise indicated for a specific problem, the values of all control parameters for all problems in this book are the values specified in *Genetic Programming III* (Koza, Bennett, Andre, and Keane 1999a).

In addition to the values of the control parameters inherited from *Genetic Programming III*, tables B.1 and B.2 present the percentages of genetic operations that are used on or before and after generation 5 of runs of the 10 problems in this book employing the architecture-altering operations. Tables B.3 and B.4 present this information for seven additional problems employing the architecture-altering operations. B.5 presents this information for all the remaining problems. Because the architecture-altering operations were not used on any of the problems in table B.5, one table of percentages is applicable to all generations.

Table B.6 shows the migration strategy (A, B, or C) used for the 41 runs in this book employing a parallel computer system. Strategy A is described in section 17.1.1 and strategies B and C are described in section 17.1.2.

The information in these tables can be summarized as follows: Not noteworthy. Many of the differences simply reflect adjustments necessary to accommodate the presence or absence of the architecture-altering operations. Other differences (such as the migration strategy) arise from some another choice (e.g., the computer system or the population size). Most of the other differences reflect the chronology of our work and the evolution of our thinking on how to maximize the efficiency runs of genetic programming in general (as opposed to any special exigency of a particular problem). At least one of the differences (where the column does not add up to 100%) reflects a typographical error made during programming.

Table B.1 Percentages of operations before generation 5 for 10 problems

	Two-lag plant (section 3.7)	Three-lag plant (section 3.8)	Three-lag plant with five-second delay (section 3.9), non-minimal phase plant (section 3.10), and three-lag plant with a free variable (section 9.1)	Controller for two families of plants (section 9.2)	RC circuit with gain greater than two (section 4.2), Philbrick circuit (section 4.3)	ALU circuit (section 4.5)	Amplifier with layout (section 5.3)
Crossover on internal points	68%	0%	0%	0%	0%	0%	0%
Crossover on terminals	10%	0%	0%	0%	0%	0%	0%
Crossover on non-numeric internal points	0%	45%	45%	46%	50%	50%	50%
Crossover on non-numeric terminals	0%	9%	9%	9%	10%	10%	10%
Crossover on numeric terminals	0%	5%	5%	9%	10%	10%	10%
Reproduction	10%	9%	9%	9%	9%	9%	9%
Subtree mutation	1%	1%	1%	1%	1%	1%	1%
Numeric constant mutation	0%	20%	20%	20%	20%	20%	20%
Subroutine duplication	5%	5%	5%	2%	0%	0%	0%
Argument duplication	0%	0%	0%	0%	0%	0%	0%

Subroutine deletion	1%	1%	1%	2%	0%	0%	0%
Argument deletion	0%	0%	0%	0%	0%	0%	0%
Subroutine creation	5%	5%	5%	2%	0%	0%	0%
Argument creation	0%	0%	0%	0%	0%	0%	0%
Iteration creation	0%	0%	0%	0%	0%	0%	0%
Loop creation	0%	0%	0%	0%	0%	0%	0%
Recursion creation	0%	0%	0%	0%	0%	0%	0%
Storage creation	0%	0%	0%	0%	0%	0%	0%
Max. points for RPB	150	150	150	150	800	300	300
Max. points for ADF	100	100	100	100	NA	NA	NA

Table B.2 Percentages of operations after generation 5 for 10 problems

	Two-lag plant (section 3.7)	Three-lag plant (section 3.8)	Three-lag plant with five-second delay (section 3.9), non-minimal phase plant (section 3.10), and three-lag plant with a free variable (section 9.1)	Controller for two families of plants (section 9.2)	RC circuit with gain greater than two (section 4.2), Philbrick circuit (section 4.3)	ALU circuit (section 4.5)	Amplifier with layout (section 5.3)
Crossover on internal points	76%	0%	0%	0%	0%	0%	0%
Crossover on terminals	10%	0%	0%	0%	0%	0%	0%
Crossover on non-numeric internal points	0%	47%	49%	46%	50%	50%	50%
Crossover on non-numeric terminals	0%	9%	9%	9%	10%	10%	10%
Crossover on numeric terminals	0%	9%	9%	9%	10%	10%	10%
Reproduction	10%	9%	9%	9%	9%	9%	9%
Subtree mutation	1%	1%	1%	1%	1%	1%	1%
Numeric constant mutation	0%	20%	20%	20%	20%	20%	20%
Subroutine duplication	1%	1%	1%	2%	0%	0%	0%
Argument duplication	0%	0%	0%	0%	0%	0%	0%

Subroutine deletion	1%	1%	1%	2%	0%	0%	0%
Argument deletion	0%	0%	0%	0%	0%	0%	0%
Subroutine creation	1%	1%	1%	2%	0%	0%	0%
Argument creation	0%	0%	0%	0%	0%	0%	0%
Iteration creation	0%	0%	0%	0%	0%	0%	0%
Loop creation	0%	0%	0%	0%	0%	0%	0%
Recursion creation	0%	0%	0%	0%	0%	0%	0%
Storage creation	0%	0%	0%	0%	0%	0%	0%
Max. points for RPB	150	150	150	150	800	300	300
Max. points for ADF	100	100	100	100	NA	NA	NA

Table B.3 Percentages of operations before generation 5 for seven problems

	Analog NAND circuit (section 4.4)	Lowpass filter with layout (section 5.2)	Square root computational circuit (section 4.6)	Passive lowpass filter with variable passband boundary (section 10.4)	Lowpass/highpass filter with free variables (section 11.1), Lowpass/highpass filter with variable passband boundary with a free variable (section 11.2)	Yagi-Uda antennas problem (section 6.5)
Crossover on internal points	79%	79%	0%	0%	0%	60%
Crossover on terminals	10%	10%	0%	0%	0%	10%
Crossover on non-numeric internal points	0%	0%	60%	49%	49%	0%
Crossover on non-numeric terminals	0%	0%	10%	10%	10%	0%
Crossover on numeric terminals	0%	0%	0%	0%	0%	0%
Reproduction	10%	10%	9%	9%	9%	9%
Subtree mutation	1%	1%	1%	1%	1%	1%
Numeric constant mutation	0%	0%	20%	20%	20%	20%
Subroutine duplication	0%	0%	0%	5%	5%	0%
Argument duplication	0%	0%	0%	0%	0%	0%

Subroutine deletion	0%	0%	0%	1%	1%	0%
Argument deletion	0%	0%	0%	0%	0%	0%
Subroutine creation	0%	0%	0%	5%	5%	0%
Argument creation	0%	0%	0%	0%	0%	0%
Iteration creation	0%	0%	0%	0%	0%	0%
Loop creation	0%	0%	0%	0%	0%	0%
Recursion creation	0%	0%	0%	0%	0%	0%
Storage creation	0%	0%	0%	0%	0%	0%
Max. points for RPB	300	600	500	300	??	500
Max. points for ADF	NA	NA	NA	100	??	NA

Table B.4 Percentages of operations after generation 5 for seven problems

	Analog NAND circuit (section 4.4)	Lowpass filter with layout (section 5.2)	Square root computational circuit (section 4.6)	Passive lowpass filter with variable passband boundary (section 10.4)	Lowpass/highpass filter with free variables (section 11.1), Lowpass/highpass filter with variable passband boundary with a free variable (section 11.2)	Yagi-Uda antennas problem (section 6.5)
Crossover on internal points	79%	79%	0%	0%	0%	60%
Crossover on terminals	10%	10%	0%	0%	0%	10%
Crossover on non-numeric internal points	0%	0%	60%	57%	57.50%	0%
Crossover on non-numeric terminals	0%	0%	10%	10%	10%	0%
Crossover on numeric terminals	0%	0%	0%	0%	0%	0%
Reproduction	10%	10%	9%	9%	9%	9%
Subtree mutation	1%	1%	1%	1%	1%	1%
Numeric constant mutation	0%	0%	20%	20%	20%	20%
Subroutine duplication	0%	0%	0%	1%	1%	0%
Argument duplication	0%	0%	0%	0%	0%	0%

Subroutine deletion	0%	0%	0%	1%	0.50%	0%
Argument deletion	0%	0%	0%	0%	0%	0%
Subroutine creation	0%	0%	0%	1%	1%	0%
Argument creation	0%	0%	0%	0%	0%	0%
Iteration creation	0%	0%	0%	0%	0%	0%
Loop creation	0%	0%	0%	0%	0%	0%
Recursion creation	0%	0%	0%	0%	0%	0%
Storage creation	0%	0%	0%	0%	0%	0%
Max. points for RPB	300	600	500	300	??	500
Max. points for ADF	NA	NA	NA	100	??	NA

Table B.5 Percentages of operations for the problems not using the architecture-altering operations

	Genetic network for *lac* operon (chapter 7)	Metabolic pathway for phospholipid cycle (section 8.6), Metabolic pathway for ketone bodies (section 8.7)	Lowpass filter without an explicit test fixture (section 4.7), Zobel network problem (two free variables (section 10.2), Third-order elliptic lowpass filter with variable modular angle (section 10.3), Active lowpass filter with variable passband boundary with free variable (section 10.5), Variable Quadratic/cubic computational circuit (section 11.3), Variable 40–60 dB amplifier (section 11.4), Improved PID tuning rules (four freevariables) (section 12.3), all three runs of non-PID parameterized controller (with four free variables), Problem of reducing amplifier distortion by means of negative feedback (section 14.2), all 11 runs of the six problems involving post-2000 patented inventions in chapter 15
Crossover on internal points	63%	58.50%	63%
Crossover on terminals	7%	6.50%	7%
Crossover on non-numeric internal points	0%	0%	0%
Crossover on non-numeric terminals	0%	0%	0%
Crossover on numeric terminals	0%	0%	0%
Reproduction	9%	9%	9%
Subtree mutation	1%	1%	1%
Numeric constant mutation	20%	20%	20%
Subroutine duplication	0%	0%	0%
Argument duplication	0%	0%	0%
Subroutine deletion	0%	0%	0%
Argument deletion	0%	0%	0%
Subroutine creation	0%	0%	0%
Argument creation	0%	0%	0%
Iteration creation	0%	0%	0%
Loop creation	0%	0%	0%
Recursion creation	0%	0%	0%
Storage creation	0%	0%	0%
Max. points for RPB	1,000	500	500
Max. points for ADF	NA	NA	NA

Table B.6 Migration strategy used on the 41 runs

Migration strategy	Problems
A	Two-lag plant (section 3.7), Three-lag plant (section 3.8), Non-minimal phase plant (section 3.10), RC circuit with gain greater than two (section 4.2), Philbrick circuit (section 4.3), NAND circuit (section 4.4), Arithmetic logic unit (ALU) circuit (section 4.5), Layout of lowpass filter (section 5.2)
B	Three-lag plant with five second delay (section 3.9), Lowpass/highpass filter with free variable (section 11.1), Lowpass/highpass filter with variable passband boundary with a free variable (section 11.2)
C	Three-lag plant with free variable (section 9.1), Controller for two families of plants (section 9.2), Square root computational circuit (section 4.6), Lowpass filter without an explicit test fixture (section 4.7), Layout of amplifier (section 5.3), Yagi–Uda antenna (section 6.5), Metabolic pathway for phospholipid cycle (section 8.6), Metabolic pathway for ketone bodies (section 8.7), Zobel network problem (two free variables) (section 10.2), Third-order elliptic lowpass filter with variable modular angle (section 10.3), Passive lowpass filter with variable passband boundary (section 10.4), Active lowpass filter with variable passband boundary with free variable (section 10.5), Variable Quadratic/cubic computational circuit (section 11.3), Variable 40–60 dB amplifier (section 11.4), Improved PID tuning rules (four free variables) (section 12.3), First non-PID parameterized controller (two free variables) (section 13.2.1), Second non-PID parameterized controller (two free variables) (section13.2.2), Third non-PID parameterized controller (two free variables) (section 13.2.3), Negative feedback (section 14.2), Low-voltage balun circuit (section 15.4.1), Mixed analog-digital variable capacitor circuit (section 15.4.2), High-current load circuit (sections 15.4.3.1 and 15.4.3.2), Voltage–current conversion circuit (section 15.4.4), Cubic function generator (sections 15.4.5 and 15.4.5), Tunable integrated active filter (sections 15.4.6.1, 15.4.6.2. 15.4.6.3, and 15.4.6.4)

The current published literature in the fields of genetic algorithms and genetic programming provides a considerable amount of conflicting advice on the topic of choosing control parameters. Moreover, it is rarely clear how one should go about translating a problem's high-level statement in English into particular choices of control parameters. Thus, it is not clear how a user of genetic programming would go about shrewdly or optimally tailoring the control parameters for a particular problem even if that were the intent. Of course, if one has sufficient understanding of the dynamics of the evolutionary process and sufficient insight concerning the fitness landscape of an unseen new problem, there is nothing wrong with doing such tailoring if the primary objective is to solve the particular problem at hand.

Appendix C: Patented or Patentable Inventions Generated by Genetic Programming

Tables C.1 and C.2 provide additional information on the 23 patent-related results (of the 36 human-competitive results produced by genetic programming in table 1.3).

Table C.1 provides additional information on the 21 results that relate to previously patented inventions. Eleven of the 21 results in table C.1 infringe previously issued patents and 10 duplicate the functionality of previously patented inventions in a non-infringing way. The first 10 entries in table C.1 refer to problems that were solved in *Genetic Programming III: Darwinian Invention and Problem Solving* (Koza, Bennett, Andre, and Keane 1999a). The last 11 entries in table C.1 are described in this book. The last six entries in table C.1 relate to patents for analog circuits that were issued after January 1, 2000.

Four of the 21 entries in the body of table C.1 are marked "See text." These entries relate to groups of previously patented inventions (as opposed to single patents) that are described in detail in *Genetic Programming III: Darwinian Invention and Problem Solving* (Koza, Bennett, Andre, and Keane 1999a). Concerning computational circuits (the 7th entry in table C.1), dozens of different computational circuits have been patented, including, for example, square root circuits (Newbold 1962) and logarithmic circuits (Green 1958). Concerning electronic thermometers (the 8th entry in table C.1), at least two dozen temperature-sensing circuits have been patented, including, for example, ones by Haeusler (1976) and Massey (1970). Concerning voltage reference circuits (the 9th entry in table C.1), Robert C. Dobkin and Robert J. Widlar of National Semiconductor Corporation received U.S. patent 3,617,859 for the voltage reference circuit (Dobkin and Widlar 1971). Subsequent to the renowned Dobkin-Widlar circuit, other patents have been issued for voltage reference circuits, including U.S. patent 3,743,923 to Goetz Wolfgang Steudel of RCA Corporation (Steudel 1973). Hundreds of patents have been issued for amplifiers (the 10th entry in table C.1).

Table C.2 shows the two inventions generated by genetic programming for which a patent application has been filed.

Table C.1 Twenty-one previously patented inventions reinvented by genetic programming

	Invention	Date	Inventor	Place	Patent	Reference
1	Darlington emitter-follower section	1953	Sidney Darlington	Bell Telephone Laboratories	2,663,806	Section 42.3 of *Genetic Programming III*
2	Ladder filter	1917	George Campbell	American Telephone and Telegraph	1,227,113	Section 25.15.1 of *Genetic Programming III* and section 5.2 of this book
3	Crossover filter	1925	Otto Julius Zobel	American Telephone and Telegraph	1,538,964	Section 32.3 of *Genetic Programming III*
4	"*M*-derived half section" filter	1925	Otto Julius Zobel	American Telephone and Telegraph	1,538,964	Section 25.15.2 of *Genetic Programming III*
5	Cauer (elliptic) topology for filters	1934 – 1936	Wilhelm Cauer	University of Gottingen	1,958,742, 1,989,545	Section 27.3.7 of *Genetic Programming III*
6	Sorting network	1962	Daniel G. O'Connor and Raymond J. Nelson	General Precision, Inc.	3,029,413	Sections 21.4.4, 23.6, and 57.8.1 of *Genetic Programming III*
7	Computational circuits	See text	See text	See text	See text	Section 47.5.3 of *Genetic Programming III*
8	Electronic thermometer	See text	See text	See text	See text	Section 49.3 of *Genetic Programming III*
9	Voltage reference circuit	See text	See text	See text	See text	Section 50.3 of *Genetic Programming III*
10	60 and 96 dB amplifiers	See text	See text	See text	See text	Section 45.3 of *Genetic Programming III*
11	Second-derivative controller	1942	Harry Jones	Brown Instrument Company	2,282,726	Section 3.7 of this book
12	Philbrick circuit	1956	George Philbrick	George A. Philbrick Researches	2,730,679	Section 4.3 of this book
13	NAND circuit	1971	David H. Chung and Bill H. Terrell	Texas Instruments Incorporated	3,560,760	Section 4.4 of this book
14	PID (proportional, integrative, and derivative) controller	1939	Albert Callender and Allan Stevenson	Imperial Chemical Limited	2,175,985	Section 9.2 of this book

(*Continued*)

Table C.1 (*Continued*)

	Invention	Date	Inventor	Place	Patent	Reference
15	Negative feedback	1937	Harold S. Black	American Telephone and Telegraph	2,102,670, 2,102,671	Chapter 14 of this book
16	Low-voltage balun circuit	2001	Sang Gug Lee	Information and Communications University	6,265,908	Section 15.4.1 of this book
17	Mixed analog-digital variable capacitor circuit	2000	Turgut Sefket Aytur	Lucent Technologies Inc.	6,013,958	Section 15.4.2 of this book
18	High-current load circuit	2001	Timothy Daun-Lindberg and Michael Miller	International Business Machines Corporation	6,211,726	Section 15.4.3 of this book
19	Voltage-current conversion circuit	2000	Akira Ikeuchi and Naoshi Tokuda	Mitsumi Electric Co., Ltd.	6,166,529	Section 15.4.4 of this book
20	Cubic function generator	2000	Stefano Cipriani and Anthony A. Takeshian	Conexant Systems, Inc.	6,160,427	Section 15.4.5 of this book
21	Tunable integrated active filter	2001	Robert Irvine and Bernd Kolb	Infineon Technologies AG	6,225,859	Section 15.4.6 of this book

Table C.2 Two patentable inventions created by genetic programming

	Claimed invention	Date of patent application	Inventors	Reference
1	Improved general-purpose tuning rules for a PID controller	July 12, 2002	Martin A. Keane, John R. Koza, and Matthew J. Streeter	Section 12.3 of this book
2	Improved general-purpose non-PID controllers	July 12, 2002	Martin A. Keane, John R. Koza, and Matthew J. Streeter	Section 13.2 of this book

Bibliography

Aarts, Emile and Korst, Jan. 1989. *Simulated Annealing and Boltzmann Machines*. Chichester: John Wiley and Sons.

Aaserud, O. and Nielsen, I. Ring. 1995. Trends in current analog design: A panel debate. *Analog Integrated Circuits and Signal-Processing*. 7(1)5–9.

Abelson, Harold and diSessa, Andrea. 1980. *Turtle Geometry*. Cambridge, MA: The MIT Press.

Altshuler, Edward E. and Linden, Derek S. 1998. *Process for the Design of Antennas Using Genetic Algorithm*. U.S. patent 5,719,794. Applied for on July 19, 1995. Issued on February 17, 1998.

Altshuler, Edward E. and Linden, Derek S. 1999. Design of wire antennas using genetic algorithms. In Rahmat-Samii, Yahya and Michielssen, Eric (editors). *Electromagnetic Optimization by Genetic Algorithms*. New York, NY: John Wiley and Sons. Chapter 8. Pages 211–248.

Andersson, Bjorn, Svensson, Per, Nordin, Peter, and Nordahl, Mats. 1999. Reactive and memory-based genetic programming for robot control. In Poli, Riccardo, Nordin, Peter, Langdon, William B., and Fogarty, Terence C. 1999. *Genetic Programming: Second European Workshop. EuroGP'99. Proceedings*. Lecture Notes in Computer Science. Volume 1598. Berlin, Germany: Springer-Verlag. Pages 161–172.

Andre, David, Bennett III, Forrest H, and Koza, John R. 1996. Discovery by genetic programming of a cellular automata rule that is better than any known rule for the majority classification problem. In Koza, John R., Goldberg, David E., Fogel, David B., and Riolo, Rick L. (editors). 1996. *Genetic Programming 1996: Proceedings of the First Annual Conference, July 28–31, 1996, Stanford University*. Cambridge, MA: MIT Press. Pages 3–11.

Andre, David and Koza, John R. 1995. Parallel genetic programming on a network of transputers. In Rosca, Justinian (editor). *Proceedings of the Workshop on Genetic Programming: From Theory to Real World Applications*. University of Rochester. National Resource Laboratory for the Study of Brain and Behavior. Technical Report 95-2. June 1995. Pages 111–120.

Andre, David and Koza, John R. 1996a. Parallel genetic programming: A scalable implementation using the transputer architecture. In Angeline, P. J. and Kinnear, K. E. Jr. (editors). 1996. *Advances in Genetic Programming 2*. Cambridge, MA: The MIT Press.

Andre, David and Koza, John R. 1996b. A parallel implementation of genetic programming that achieves super-linear performance. In Arabnia, Hamid R. (editor). *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications*. Athens, GA: CSREA. Volume III. Pages 1163–1174.

Andre, David and Teller, Astro. 1999. Evolving team Darwin United. In Asada, Minoru and Kitano, Hiroaki (editors). *RoboCup-98: Robot Soccer World Cup II*. Lecture Notes in Computer Science. Volume 1604. Berlin: Springer-Verlag. Pages 346–352.

Angeline, Peter J. and Kinnear, Kenneth E. Jr. (editors). 1996. *Advances in Genetic Programming 2*. Cambridge, MA: The MIT Press.

Angeline, Peter J. 1997. An alternative to indexed memory for evolving programs with explicit state representations. In Koza, John R., Deb, Kalyanmoy, Dorigo, Marco, Fogel, David B., Garzon, Max, Iba, Hitoshi, and Riolo, Rick L. (editors). *Genetic Programming 1997: Proceedings of the Second Annual Conference, July 13–16, 1997, Stanford University*. San Francisco, CA: Morgan Kaufmann. Pages 423–430.

Angeline, Peter J. 1998a. Multiple interacting programs: A representation for evolving complex behaviors. *Cybernetics and Systems*. 29(8)779–806.

Angeline, Peter J. 1998b. Evolving predictors for chaotic time series. In Rogers, S., Fogel, D., Bezdek, J., and Bosacchi, B. (editors). *Proceedings of SPIE (Volume 3390): Application and Science of Computational Intelligence*, Bellingham, WA: The International Society for Optical Engineering. Pages 170–180.

Angeline, Peter J. and Fogel, David B. 1997. An evolutionary program for the identification of dynamical systems. In Rogers, S. (editor). *Proceedings of SPIE (Volume 3077): Application and Science of Artificial Neural Networks III*. Bellingham, WA: The International Society for Optical Engineering. Pages 409–417.

Arkin, Adam, Shen, Peidong, and Ross, John. 1997. A test case of correlation metric construction of a reaction pathway from measurements. *Science*. 277. Pages 1275–1279. August 29, 1997.

Armstrong, Edwin Howard. 1914. *Wireless Receiving System*. U.S. patent 1,113,149. Filed October 29, 1913. Issued October 6, 1914.

Åström, Karl J. and Hägglund, Tore. 1995. *PID Controllers: Theory, Design, and Tuning*. Second Edition. Research Triangle Park, NC: Instrument Society of America.

Aytur, Turgut Sefket. 2000. *Integrated Circuit with Variable Capacitor*. U.S. patent 6,013,958. Filed July 23, 1998. Issued January 11, 2000.

Babanezhad, J. N. and Temes, G. C. 1986. Analog MOS computational circuits. *Proceedings of the IEEE Circuits and System International Symposium*. Piscataway, NJ: IEEE Press. Pages 1156–1160.

Babovic, Vladan. 1996. *Emergence, Evolution, Intelligence: Hydroinformatics*. Rotterdam, The Netherlands: Balkema Publishers.

Bagchi, Tapan P. 1999. *Multiobjective Scheduling by Genetic Algorithms*. Boston: Kluwer Academic Publishers.

Balanis, Constantine A. 1982. *Antenna Theory: Analysis and Design*. New York, NY: John Wiley and Sons.

Banzhaf, Wolfgang, Daida, Jason, Eiben, A. E., Garzon, Max H., Honavar, Vasant, Jakiela, Mark, and Smith, Robert E. (editors). 1999. *GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference, July 13–17, 1999, Orlando, Florida USA*. San Francisco, CA: Morgan Kaufmann.

Banzhaf, Wolfgang, Nordin, Peter, Keller, Robert E., and Francone, Frank D. 1998. *Genetic Programming: An Introduction*. San Francisco, CA: Morgan Kaufmann and Heidelberg: dpunkt.

Banzhaf, Wolfgang, Nordin, Peter, Keller, Richard, and Olmer, Markus. 1997. Generating adaptive behavior for a real robot using function regression with genetic programming. In Koza, John R., Deb, Kalyanmoy, Dorigo, Marco, Fogel, David B., Garzon, Max, Iba, Hitoshi, and Riolo, Rick L. (editors). *Genetic Programming 1997: Proceedings of the Second Annual Conference, July 13–16, 1997, Stanford University*. San Francisco, CA: Morgan Kaufmann. Pages 35–43.

Banzhaf, Wolfgang, Poli, Riccardo, Schoenauer, Marc, and Fogarty, Terence C. 1998. *Genetic Programming: First European Workshop. EuroGP'98. Paris, France, April 1998 Proceedings*. Lecture Notes in Computer Science. Volume 1391. Berlin, Germany: Springer-Verlag.

Barnum, H., Bernstein, H.J. and Spector, Lee. 2000. Quantum circuits for OR and AND of ORs. *Journal of Physics A: Mathematical and General*. 33(45)8047–8057. November 17, 2000.

Bennett III, Forrest H. and Koza, John R. 2002. *Method and Apparatus for Automatic Synthesis, Placement and Routing of Complex Structures*. U.S. patent 6,424,959. Filed June 17, 1999. Issued July 23, 2002.

Bennett III, Forrest H, Koza, John R., Andre, David, and Keane, Martin A. 1996. Evolution of a 60 decibel op amp using genetic programming. In Higuchi, Tetsuya, Iwata, Masaya, and Lui, Weixin (editors). *Proceedings of International Conference on Evolvable Systems: From Biology to Hardware (ICES-96)*. Lecture Notes in Computer Science, Volume 1259. Berlin: Springer-Verlag. Pages 455–469.

Bennett III, Forrest H, Koza, John R., Yu, Jessen, and Mydlowec, William. 2000. Automatic synthesis, placement, and routing of an amplifier circuit by means of genetic programming. In Miller, Julian, Thompson, Adrian, Thomson, Peter, and Fogarty, Terence C. (editors). 2000. *Evolvable Systems: From Biology to Hardware. Third International Conference, ICES 2000, Edinburgh, Scotland, UK, April 2000 Proceedings*. Lecture Notes in Computer Science. Volume 1801. Berlin, Germany: Springer-Verlag. Pages 1–10.

Bennett III, Forrest H, Koza, John R., Keane, Martin A., Yu, Jessen, Mydlowec, William, and Stiffelman, Oscar. 1999. Evolution by means of genetic programming of analog circuits that perform digital functions. In Banzhaf, Wolfgang, Daida, Jason, Eiben, A. E., Garzon, Max H., Honavar, Vasant, Jakiela, Mark, and Smith, Robert E. (editors). 1999. *GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference, July 13–17, 1999, Orlando, Florida USA*. San Francisco, CA: Morgan Kaufmann. Pages 1477–1483.

Bennett III, Forrest H, Koza, John R., Shipman, James, and Stiffelman, Oscar. 1999. Building a parallel computer system for $18,000 that performs a half peta-flop per day. In Banzhaf, Wolfgang, Daida, Jason, Eiben, A. E., Garzon, Max H., Honavar, Vasant, Jakiela, Mark, and Smith, Robert E. (editors). 1999. *GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference, July 13–17, 1999, Orlando, Florida USA*. San Francisco, CA: Morgan Kaufmann. Pages 1484–1490.

Black, Harold S. 1928. *Translating System*. U.S. patent 1,686,792. Filed February 3, 1925. Issued October 9, 1928.

Black, Harold S. 1935. *Wave Translation System*. U.S. patent 2,003,282. Filed August 8, 1928. Issued June 4, 1935.

Black, Harold S. 1937a. *Wave Translation System*. U.S. patent 2,102,670. Filed August 8, 1928. Issued December 21, 1937.

Black, Harold S. 1937b. *Wave Translation System*. U.S. patent 2,102,671. Filed April 22, 1932. Issued December 21, 1937.

Black, Harold S. 1977. Inventing the negative feedback amplifier. *IEEE Spectrum*. December 1977. Pages 55–60.

Blickle, Tobias. 1997. *Theory of Evolutionary Algorithms and Application to System Synthesis*. TIK-Schriftenreihe Nr. 17. Zurich, Switzerland: vdf Hochschul Verlag AG an der ETH Zuerich.

Boutin, Noel. 2002. Use time-domain analysis of Zobel network. *EDN*. July 27, 2002. Page 86.

Bower, James M. and Bolouri, Hamid. 2000. *Computational Modeling of Genetic and Biochemical Networks*. Cambridge, MA: MIT Press.

Boyd, S. P. and Barratt, C. H. 1991. *Linear Controller Design: Limits of Performance*. Englewood Cliffs, NJ: Prentice Hall.

Bryson, Arthur E., and Ho, Yu-Chi. 1975. *Applied Optimal Control*. New York: Hemisphere Publishing.

Burke, Gerald J. 1992. *Numerical Electromagnetics Code—NEC-4: Method of Moments—User*'s *Manual*. Lawrence Livermore National Laboratory report UCRL-MA-109338. Livermore, CA: Lawrence Livermore National Laboratory.

Callender, Albert and Stevenson, Allan Brown. 1939. *Automatic Control of Variable Physical Characteristics*. U.S. patent 2,175,985. Filed February 17, 1936 in the United States. Filed February 13, 1935 in Great Britain. Issued October 10, 1939 in the United States.

Campbell, George A. 1917. *Electric Wave Filter*. Filed July 15, 1915. U.S. patent 1,227,113. Issued May 22, 1917.

Cantu-Paz, Erick. 2000. *Efficient and Accurate Parallel Genetic Algorithms*. Boston: Kluwer Academic Publishers.

Cauer, Wilhelm. 1934. *Artificial Network*. U.S. patent 1,958,742. Filed June 8, 1928 in Germany. Filed December 1, 1930 in the United States. Issued May 15, 1934.

Cauer, Wilhelm. 1935. *Electric Wave Filter*. U.S. patent 1,989,545. Filed June 8, 1928. Filed December 6, 1930 in the United States. Issued January 29, 1935.

Cauer, Wilhelm. 1936. *Unsymmetrical Electric Wave Filter*. U.S. patent 2,048,426. Filed November 10, 1932 in Germany. Filed November 23, 1933 in the United States. Issued July 21, 1936.

Chung, David H. and Terrell, Bill H. 1971. *Logic NAND Gate Circuits*. U.S. patent 3,560,760. Filed February 2, 1970. Issued February 2, 1971.

Cipriani, Stefano and Takeshian, Anthony A. 2000. *Compact Cubic Function Generator*. U.S. patent 6,160,427. Filed September 4, 1998. Issued December 12, 2000.

Coello Coello, Carlos A., Van Veldhuizen, David A., and Lamont, Gary B. 2002. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Boston: Kluwer Academic Publishers.

Cohn, John M., Garrod, David J., Rutenbar, Rob A., and Carley, L. Richard. 1994. *Analog Device-Level Layout Automation*. Boston: Kluwer.

Collado-Vides, Julio and Hofestadt, Ralf. 2002. *Gene Regulation and Metabolism*. Cambridge, MA: The MIT Press.

Comisky, William, Yu, Jessen, and Koza, John. 2000. Automatic synthesis of a wire antenna using genetic programming. *Late Breaking Papers at the 2000 Genetic and Evolutionary Computation Conference, Las Vegas, Nevada*. Pages 179–186.

Crawford, L. S., Cheng, V. H. L., and Menon, P. K. 1999. Synthesis of flight vehicle guidance and control laws using genetic search methods. *Proceedings of 1999 Conference on Guidance, Navigation, and Control*. Reston, VA: American Institute of Aeronautics and Astronautics. Paper AIAA-99-4153.

Darlington, Sidney. 1953. *Semiconductor Signal Translating Device*. U.S. patent 2,663,806. Filed May 9, 1952. Issued December 22, 1953.

Daun-Lindberg, Timothy Charles and Miller, Michael Lee. 2001. *Low Voltage High-Current Electronic Load*. U.S. patent 6,211,726. Filed June 28, 1999. Issued April 3, 2001.

Deb, Kalyanmoy. 2001. *Multi-Objective Optimization using Evolutionary Algorithms*. Boston: Kluwer Academic Publishers.

Dewell, Larry D. and Menon, P. K. 1999. Low-thrust orbit transfer optimization using genetic search. *Proceedings of 1999 Conference on Guidance, Navigation, and Control*. Reston, VA: American Institute of Aeronautics and Astronautics. Paper AIAA-99-4151.

D'haeseleer, Patrik, Wen, Xiling, Fuhrman, Stefanie, and Somogyi, Roland. 1999. Linear modeling of mRNA expression levels during CNS development and injury. In Altman, Russ B. Dunker, A. Keith, Hunter, Lawrence, Klein, Teri E., and Lauderdale, Kevin (editors). *Pacific Symposium on Biocomputing '99*. Singapore: World Scientific. Pages 41–52.

Dobkin, Robert C. and Widlar, Robert J. 1971. *Electrical Regulator Apparatus Including a Zero-Temperature Coefficient Voltage Reference Circuit*. U.S. patent 3,617,859. Filed May 23, 1970. Issued November 2, 1971.

Dorf, Richard C. and Bishop, Robert H. 1998. *Modern Control Systems*. Eighth edition. Menlo Park, CA: Addison-Wesley.

Drechsler, Rolf. 1998. *Evolutionary Algorithms for VLSI CAD*. Boston: Kluwer Academic Publishers.

Foster, James A., Lutton, Evelyne, Miller, Julian, Ryan, Conor, and Tettamanzi, Andrea G. B. (editors). 2002. *Genetic Programming: 5th European Conference, EuroGP 2002, Kinsale, Ireland, April 2002 Proceedings*. Berlin: Springer-Verlag.

Garey, Michael R. and Johnson, David S. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. New York, NY: W. H. Freeman.

Getreu, Ian. 2002. Productivity tools for analog/mixed-signal designs: Ready for prime time? *Electronic Design*. June 10, 2002. Page 40.

Gilbert, Barrie. 1968. A precise four-quadrant multiplier with subnanosecond response. *IEEE Journal of Solid-State Circuits*. Volume SC-3. Number 4. December 1968. Pages 365–373.

Gilbert, Barrie. 1979. *Multiplier Circuit*. U.S. patent 4,156,283. Filed October 3, 1977. Issued May 22, 1979.

Goddard, Robert. 1915. *Method of and Apparatus for Producing Electrical Impulses or Oscillations*. U.S. patent 1,159,209. Filed August 1, 1912. Issued November 2, 1915.

Goldberg, David E. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley.

Goldberg, David E. 1990. A note on Boltzmann tournament selection for genetic algorithms and population-oriented simulated annealing. *Complex Systems*. 4(4)445–460.

Goldberg, David E. 2002. *The Design of Innovation: Lessons from and for Competent Genetic Algorithms*. Boston: Kluwer Academic Publishers.

Green, Milton. 1958. *Logarithmic Converter Circuit*. U.S. patent 2,861,182. Filed June 16, 1953. Issued November 18, 1958.

Grimbleby, J. B. 1995. Automatic analogue network synthesis using genetic algorithms. *Proceedings of the First International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications (GALESIA)*. London: Institution of Electrical Engineers. Pages 53–58.

Gruau, Frederic. 1992a. *Cellular Encoding of Genetic Neural Networks*. Technical report 92–21. Laboratoire de l'Informatique du Parallélisme. Ecole Normale Supérieure de Lyon. May 1992.

Gruau, Frederic. 1992b. Genetic synthesis of Boolean neural networks with a cell rewriting developmental process. In Schaffer, J. D. and Whitley, Darrell (editors). *Proceedings of the Workshop on Combinations of Genetic Algorithms and Neural Networks 1992*. Los Alamitos, CA: The IEEE Computer Society Press.

Haeusler, Jochen, 1976. *Arrangement for Measuring Temperatures*. U.S. patent 3,943,434. Filed February 6, 1974. Issued March 9, 1976.

Haupt, Randy L. 1994. Thinned arrays using genetic algorithms. *IEEE Transactions on Antennas and Propagation*. Volume 42: Pages 993–999.

Higuchi, Tetsuya, Iwata, Masaya, and Lui, Weixin (editors). 1997. *Evolvable Systems: From Biology to Hardware: First International Conference, ICES-96, Tsukuba, Japan, October 1996 Proceedings*. Lecture Notes in Computer Science, Volume 1259. Berlin: Springer-Verlag.

Higuchi, Tetsuya, Niwa, Tatsuya, Tanaka, Toshio, Iba, Hitoshi, de Garis, Hugo, and Furuya, Tatsumi. 1993a. Evolving hardware with genetic learning: A first step towards building a Darwin machine. In Meyer, Jean-Arcady, Roitblat, Herbert L. and Wilson, Stewart W. (editors). *From Animals to Animats 2: Proceedings of the Second International Conference on Simulation of Adaptive Behavior*. Cambridge, MA: The MIT Press. 1993. Pages 417–424.

Higuchi, Tetsuya, Niwa, Tatsuya, Tanaka, Toshio, Iba, Hitoshi, de Garis, Hugo, and Furuya, Tatsumi. 1993b. *Evolvable Hardware–Genetic-Based Generation of Electric Circuitry at Gate and Hardware Description Language (HDL) Levels*. Electrotechnical Laboratory technical report 93-4. Tsukuba, Japan: Electrotechnical Laboratory.

Holland, John H. 1975. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. Ann Arbor, MI: University of Michigan Press. Second edition. Cambridge, MA: The MIT Press 1992.

Hsu, Feng-Hsiung. 2002. *Behind Deep Blue: Building the Computer That Defeated the World Chess Champion*. Princeton, NJ: Princeton University Press.

Iba, Hitoshi. 1996. *Genetic Programming*. Tokyo: Tokyo Denki University Press. In Japanese.

Ikeuchi, Akira and Tokuda, Naoshi. 2000. *Voltage-Current Conversion Circuit*. U.S. patent 6,166,529. Filed February 24, 2000 in the United States. Issued December 26, 2000 in the United States. Filed March 10, 1999 in Japan.

Ince, D. C. (editor). 1992. *Mechanical Intelligence: Collected Works of A. M. Turing*. Amsterdam: North Holland.

Irvine, Robert and Kolb, Bernd. 2001. *Integrated Low-Pass Filter*. U.S. patent 6,225,859. Filed September 14, 1998. Issued May 1, 2001.

Jacob, Christian. 1997. *Principia Evolvica: Simulierte Evolution mit Mathematica*. Heidelberg, Germany: dpunkt.verlag.

Jacob, Christian. 2001. *Illustrating Evolutionary Computation with Mathematica*. San Francisco: Morgan Kaufmann.

Jamshidi, Mo, Coelho, Leandro dos Santos, Krohling, Renato A., and Fleming, Peter J. 2003. *Robust Control Systems with Genetic Algorithms*. Boca Raton, FL: CRC Press.

Johnson, Kenneth S. 1926. *Electric-Wave Transmission*. U.S. patent 1,611,916. Filed March 9, 1923. Issued December 28, 1926.

Johnson, Walter C. 1950. *Transmission Lines and Networks*. New York: NY: McGraw-Hill.

Johnson, J. Michael and Rahmat-Samii, Yahya. 1999. Genetic algorithms and method of moments (GA/MOM) for the design of integrated antennas. *IEEE Transactions on Antennas and Propagation*. 47(10)1606–1614. October 1999.

Jones, Eric A. 1999. *Genetic Design of Antennas and Electronic Circuits*. Ph.D. Thesis. Department of Electrical and Computer Engineering. Duke University.

Jones, Harry S. 1942. *Control Apparatus*. U.S. patent 2,282,726. Filed October 25, 1939. Issued May 12, 1942.

Keane, Martin A., Koza, John R., and Rice, James P. 1993. Finding an impulse response function using genetic programming. *Proceedings of the 1993 American Control Conference*. Evanston, IL: American Automatic Control Council. Volume III. Pages 2345–2350.

Keane, Martin A., Koza, John R., and Streeter, Matthew J. 2002a. *Improved General-Purpose Controllers*. U.S. patent application filed July 12, 2002.

Keane, Martin A., Koza, John R., and Streeter, Matthew J. 2002b. Automatic synthesis using genetic programming of an improved general-purpose controller for industrially representative plants. In Stoica, Adrian, Lohn, Jason, Katz, Rich, Keymeulen, Didier and Zebulum, Ricardo (editors) 2002. *Proceedings of 2002 NASA/DoD Conference on Evolvable Hardware*. Los Alamitos, CA: IEEE Computer Society. Pages 113–122.

Keane, Martin A., Yu, Jessen, and Koza, John R. 2000. Automatic synthesis of both the topology and tuning of a common parameterized controller for two families of plants using genetic programming. In Whitley, Darrell, Goldberg, David, Cantu-Paz, Erick, Spector, Lee, Parmee, Ian, and Beyer, Hans-Georg (editors). *GECCO-2000: Proceedings of the Genetic and Evolutionary Computation Conference, July 10–12, 2000, Las Vegas, Nevada*. San Francisco: Morgan Kaufmann. Pages 496–504.

Keymeulen, Didier, Stoica, Adrian, Lohn, Jason, and Zebulum, Ricardo Salem (editors). 2001. *Proceedings of the Third NASA/DOD Workshop on Evolvable Hardware, Pasadena, California, July 12–14, 2001*. Los Alamitos, CA. IEEE Computer Society.

Kinnear, Kenneth E. Jr. (editor). 1994. *Advances in Genetic Programming*. Cambridge, MA: MIT Press.

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. 1983. Optimization by simulated annealing. *Science* 220. Pages 671–680.

Kitano, Hiroaki. 1990. Designing neural networks using genetic algorithms with graph generation system. *Complex Systems*. 4 (1990) 461–476.

Kitano, Hiroaki. 2001. *Foundations of Systems Biology*. Cambridge, MA: The MIT Press.

Koza, John R. 1988. *Nonlinear Genetic Algorithms for Solving Problems*. U.S. patent application filed May 20, 1988.

Koza, John R. 1989. Hierarchical genetic algorithms operating on populations of computer programs. In *Proceedings of the 11th International Joint Conference on Artificial Intelligence*. San Mateo, CA: Morgan Kaufmann. Volume I. Pages 768–774.

Koza, John R. 1990a. *Genetic Programming: A Paradigm for Genetically Breeding Populations of Computer Programs to Solve Problems*. Stanford University Computer Science Department technical report STAN-CS-TR-90-1314. June 1990.

Koza, John R. 1990b. *Non-Linear Genetic Algorithms for Solving Problems*. U.S. patent 4,935,877. Filed May 20, 1988. Issued June 19, 1990.

Koza, John R. 1992a. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. Cambridge, MA: MIT Press.

Koza, John R. 1992b. *Non-Linear Genetic Algorithms for Solving Problems by Finding a Fit Composition of Functions*. U. S. patent 5,136,686. Filed March 28, 1990. Issued August 4, 1992.

Koza, John R. 1992c. Hierarchical automatic function definition in genetic programming. In Whitley, Darrell (editor). 1993. *Foundations of Genetic Algorithms 2*. San Mateo, CA: Morgan Kaufmann Publishers. Pages 297–318.

Koza, John R. 1992d. A genetic approach to finding a controller to back up a tractor-trailer truck. In *Proceedings of the 1992 American Control Conference*. Evanston, IL: American Automatic Control Council. Volume III. Pages 2307–2311.

Koza, John R. 1993. Discovery of rewrite rules in Lindenmayer systems and state transition rules in cellular automata via genetic programming. *Symposium on Pattern Formation (SPF-93), Claremont, California. February 13, 1993*. A copy of this presented, but otherwise unpublished, paper is available at `http://www.smi.stanford.edu/people/koza`.

Koza, John R. 1994a. *Genetic Programming II: Automatic Discovery of Reusable Programs*. Cambridge, MA: MIT Press.

Koza, John R. 1994b. *Genetic Programming II Videotape: The Next Generation*. Cambridge, MA: MIT Press.

Koza, John R. 1994c. *Architecture-Altering Operations for Evolving the Architecture of a Multi-Part Program in Genetic Programming*. Stanford University Computer Science Department technical report STAN-CS-TR-94-1528. October 21, 1994.

Koza, John R. 1995a. Evolving the architecture of a multi-part program in genetic programming using architecture-altering operations. In McDonnell, John R., Reynolds, Robert G., and Fogel, David B. (editors). 1995. *Evolutionary Programming IV: Proceedings of the Fourth Annual Conference on Evolutionary Programming*. Cambridge, MA: The MIT Press. Pages 695–717.

Koza, John R. 1995b. Gene duplication to enable genetic programming to concurrently evolve both the architecture and work-performing steps of a computer program. *Proceedings of the 14^{th} International Joint Conference on Artificial Intelligence*. San Francisco: Morgan Kaufmann. Pages 734–740.

Koza, John R. 1995c. Two ways of discovering the size and shape of a computer program to solve a problem. In Eshelman, Larry J. (editor). *Proceedings of the Sixth International Conference on Genetic Algorithms*. San Francisco: Morgan Kaufmann. Pages 287–294.

Koza, John R. and Andre, David. 1995. *Parallel Genetic Programming on a Network of Transputers*. Stanford University Computer Science Department technical report STAN-CS-TR-95-1542. January 30, 1995.

Koza, John R., Andre, David, and Tackett, Walter Alden. 1994. *Simultaneous Evolution of the Architecture of a Multi-Part Program to Solve a Problem Using Architecture Altering Operations*. U.S. patent application filed August 4, 1994.

Koza, John R., Andre, David, and Tackett, Walter Alden. 1998. *Simultaneous Evolution of the Architecture of a Multi-Part Program to Solve a Problem Using Architecture Altering Operations*. U. S. patent 5,742,738. Filed August 4, 1994. Issued April 21, 1998.

Koza, John R., Andre, David, and Tackett, Walter Alden. 2000. *Simultaneous Evolution of the Architecture of a Multi-Part Program to Solve a Problem Using Architecture Altering Operations*. U. S. patent 6,058,385. Filed March 7, 1997. Issued May 2, 2000.

Koza, John R., Banzhaf, Wolfgang, Chellapilla, Kumar, Deb, Kalyanmoy, Dorigo, Marco, Fogel, David B., Garzon, Max H., Goldberg, David E., Iba, Hitoshi, and Riolo, Rick. (editors). 1998. *Genetic Programming 1998: Proceedings of the Third Annual Conference*. San Francisco, CA: Morgan Kaufmann.

Koza, John R., and Bennett III, Forrest H. 1999. Automatic synthesis, placement, and routing of electrical circuits by means of genetic programming. In Spector, Lee, Langdon, William B., O'Reilly, Una-May, and Angeline, Peter (editors). *Advances in Genetic Programming 3*. Cambridge, MA: MIT Press. Chapter 6. Pages 105–134.

Koza, John R., Bennett III, Forrest H, Andre, David, and Keane, Martin A. 1996a. Automated design of both the topology and sizing of analog electrical circuits using genetic programming. In Gero, John S. and Sudweeks, Fay (editors). *Artificial Intelligence in Design '96*. Dordrecht: Kluwer Academic Publishers. Pages 151–170.

Koza, John R., Bennett III, Forrest H, Andre, David, and Keane, Martin A. 1996b. Automated WYWIWYG design of both the topology and component values of analog electrical circuits using genetic programming. In Koza, John R., Goldberg, David E., Fogel, David B., and Riolo, Rick L. (editors). *Genetic Programming 1996: Proceedings of the First Annual Conference, July 28–31, 1996, Stanford University*. Cambridge, MA: The MIT Press. Pages 123–131.

Koza, John R., Bennett III, Forrest H, Andre, David, and Keane, Martin A. 1996c. Reuse, parameterized reuse, and hierarchical reuse of substructures in evolving electrical circuits using genetic programming. In Higuchi, Tetsuya, Iwata, Masaya, and Liu, Weixin (editors). 1997. *Proceedings of International Conference on Evolvable Systems: From Biology to Hardware (ICES-96)*. Lecture Notes in Computer Science, Volume 1259. Berlin: Springer-Verlag. Berlin: Springer-Verlag. Pages 312–326.

Koza, John R., Bennett III, Forrest H, Andre, David, and Keane, Martin A. 1996d. Toward evolution of electronic animals using genetic programming. In Langton, Christopher G. and Shimohara, Katsunori (editors). 1997. *Artificial Life V: Proceedings of the Fifth International Workshop on the Synthesis and Simulation of Living Systems*. Cambridge, MA: The MIT Press. Pages 327–334.

Koza, John R., Bennett III, Forrest H, Andre, David, and Keane, Martin A. 1996e. Four problems for which a computer program evolved by genetic programming is competitive with human performance. *Proceedings of the 1996 IEEE International Conference on Evolutionary Computation*. IEEE Press. Pages 1–10.

Koza, John R., Bennett III, Forrest H, Andre, David, and Keane, Martin A. 1996f. *Method and Apparatus for Automated Design of Electrical Circuits Using Genetic Programming*. U.S. patent application filed February 20, 1996.

Koza, John R., Bennett III, Forrest H, Andre, David, and Keane, Martin A. 1999a. *Genetic Programming III: Darwinian Invention and Problem Solving*. San Francisco, CA: Morgan Kaufmann.

Koza, John R., Bennett III, Forrest H, Andre, David, and Keane, Martin A. 1999b. *Method and Apparatus for Automated Designs of Complex Structures using Genetic Programming*. Filed February 20. 1996. U. S. patent 5,867,397. Issued February 2, 1999.

Koza, John R., Bennett III, Forrest H, Andre, David, and Keane, Martin A. 1999c. *Genetic Programming Problem Solver with Automatically Defined Stores, Loops, and Recursions*. U.S. patent application filed April 12, 1999.

Koza, John R., Bennett III, Forrest H, Andre, David, and Keane, Martin A. 2002. *Method and Apparatus for Automated Design of Complex Structures using Genetic Programming*. U.S. patent 6,360,191. Filed February 20, 1996 and January 5, 1999. Issued March 19, 2002.

Koza, John R., Bennett III, Forrest H, Andre, David, and Keane, Martin A. 2003. *Genetic Programming Problem Solver with Automatically Defined Stores, Loops, and Recursions*. U.S. patent 6,532,453. Filed April 12, 1999. Issued March 11, 2003.

Koza, John R., Bennett III, Forrest H, Andre, David, Keane, Martin A., and Brave, Scott. 1999. *Genetic Programming III Videotape: Human-Competitive Machine Intelligence*. San Francisco, CA: Morgan Kaufmann.

Koza, John R., Bennett III, Forrest H, Andre, David, Keane, Martin A., and Dunlap, Frank. 1997. Automated synthesis of analog electrical circuits by means of genetic programming. *IEEE Transactions on Evolutionary Computation*. 1(2). Pages 109–128.

Koza, John R., Bennett, Forrest H, III, Hutchings, Jeffrey L., Bade, Stephen L., Keane, Martin A., and Andre, David. 1997. Evolving sorting networks using genetic programming and the rapidly reconfigurable Xilinx 6216 field-programmable gate array. *Proceedings of the 31st Asilomar Conference on Signals, Systems, and Computers*. Piscataway, NJ: IEEE Press. Pages 404–410.

Koza, John R., Bennett, Forrest H, III, Hutchings, Jeffrey L., Bade, Stephen L., Keane, Martin A., and Andre, David. 1998. Evolving computer programs using rapidly reconfigurable field-programmable gate arrays and genetic programming. *Proceedings of the ACM Sixth International Symposium on Field Programmable Gate Arrays*. New York: ACM Press. Pages 209–219.

Koza, John R., Bennett III, Forrest H, Keane, Martin A., and Andre, David. 1997. Automatic programming of a time-optimal robot controller and an analog electrical circuit to implement the robot controller by means of genetic programming. In *Proceedings of 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation*. Los Alamitos, CA: Computer Society Press. Pages 340–346.

Koza, John R., Bennett III, Forrest H, Keane, Martin A., Yu, Jessen, Mydlowec, William, and Stiffelman, Oscar. 1999. Searching for the impossible using genetic programming. In Banzhaf, Wolfgang, Daida, Jason, Eiben, A. E., Garzon, Max H., Honavar, Vasant, Jakiela, Mark, and Smith, Robert E. (editors). 1999. *GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference, July 13–17, 1999, Orlando, Florida USA*. San Francisco, CA: Morgan Kaufmann. Pages 1083–1091.

Koza, John R., Bennett III, Forrest H, and Stiffelman, Oscar. 1999a. Genetic programming as a Darwinian invention machine. In Poli, Riccardo, Nordin, Peter, Langdon, William B., and Fogarty, Terence C. 1999. *Genetic Programming: Second European Workshop. EuroGP'99. Proceedings*. Lecture Notes in Computer Science. Volume 1598. Berlin, Germany: Springer-Verlag. Pages 93–108.

Koza, John R., Bennett III, Forrest H, and Stiffelman, Oscar. 1999b. *An Invention Machine that Automatically Creates Novel Designs*. U.S. patent application filed April 13, 2000.

Koza, John R., Deb, Kalyanmoy, Dorigo, Marco, Fogel, David B., Garzon, Max, Iba, Hitoshi, and Riolo, Rick L. (editors). 1997. *Genetic Programming 1997: Proceedings of the Second Annual Conference, July 13–16, 1997, Stanford University*. San Francisco, CA: Morgan Kaufmann.

Koza, John R., Goldberg, David E., Fogel, David B., and Riolo, Rick L. (editors). 1996. *Genetic Programming 1996: Proceedings of the First Annual Conference, July 28–31, 1996, Stanford University*. Cambridge, MA: MIT Press.

Koza, John R., and Keane, Martin A. 1990a. Cart centering and broom balancing by genetically breeding populations of control strategy programs. In *Proceedings of International Joint Conference on Neural Networks, Washington, January 15–19, 1990*. Hillsdale, NJ: Lawrence Erlbaum. Volume I, Pages 198–201.

Koza, John R., and Keane, Martin A. 1990b. Genetic breeding of nonlinear optimal control strategies for broom balancing. In *Proceedings of the Ninth International Conference on Analysis and Optimization of Systems. Antibes, France, June, 1990*. Berlin: Springer-Verlag. Pages 47–56.

Koza, John R., Keane, Martin A., Bennett III, Forrest H, Yu, Jessen, Mydlowec, William, and Stiffelman, Oscar. 1999. Automatic creation of both the topology and parameters for a robust controller by means of genetic programming. *Proceedings of the 1999 IEEE International Symposium on Intelligent Control, Intelligent Systems, and Semiotics*. Piscataway, NJ: IEEE. Pages 344–352.

Koza, John R., Keane, Martin A., and Streeter, Matthew J. 2003. Evolving inventions. *Scientific American*. February 2003. 288(2) 52–59.

Koza, John R., Keane, Martin A., Yu, Jessen, Bennett III, Forrest H, and Mydlowec, William. 2000. Automatic creation of human-competitive programs and controllers by means of genetic programming. *Genetic Programming and Evolvable Machines*. Volume 1. Number 1/2. Pages 121–164.

Koza, John R., Keane, Martin A., Yu, Jessen, Bennett III, Forrest H, Mydlowec, William, and Stiffelman, Oscar. 1999. Automatic synthesis of both the topology and parameters for a robust controller for a non-minimal phase plant and a three-lag plant by means of genetic programming. *Proceedings of 1999 IEEE Conference on Decision and Control*. Pages 5292–5300.

Koza, John R., Keane, Martin A., Yu, Jessen, Bennett III, Forrest H, and Mydlowec, William. 2003. *Method and Apparatus for Automatic Synthesis of Controllers*. U.S. patent application filed September 10, 1999. Application number 09/393,863. Allowed October 30, 2002.

Koza, John R., Keane, Martin A., Yu, Jessen, and Mydlowec, William. 2000. Automatic synthesis of electrical circuits containing a free variable using genetic programming. In Whitley, Darrell, Goldberg, David, Cantu-Paz, Erick, Spector, Lee, Parmee, Ian, and Beyer, Hans-Georg (editors). *GECCO-2000: Proceedings of the Genetic and Evolutionary Computation Conference, July 10–12, 2000, Las Vegas, Nevada*. San Francisco: Morgan Kaufmann. Pages 551–557.

Koza, John R., Keane, Martin A., Yu, Jessen, Mydlowec, William, and Bennett, Forrest H III. 2000a. Automatic synthesis of both the topology and parameters for a controller for a three-lag plant with a five-second delay using genetic programming. In Cagnoni, Stafano et al. (editors). *Real World Applications of Evolutionary Computing. EvoWorkshops 2000. EvoIASP, Evo SCONDI, EvoTel, EvoSTIM, EvoRob, and EvoFlight, Edinburgh, Scotland, UK, April 2000, Proceedings*. Lecture Notes in Computer Science. Volume 1803. Berlin, Germany: Springer-Verlag. Pages 168–177.

Koza, John R., Keane, Martin A., Yu, Jessen, Mydlowec, William, and Bennett, Forrest H III. 2000b. Automatic synthesis of both the control law and parameters for a controller for a three-lag plant with five-second delay using genetic programming and simulation techniques. In *Proceedings of the 2000 American Control Conference, Chicago, Illinois, June 28–30, 2000*. Evanston, IL: American Automatic Control Council. Pages 453–459.

Koza, John R., Mydlowec, William, Lanza, Guido, Yu, Jessen, and Keane, Martin A. 2000a. *Reverse Engineering and Automatic Synthesis of Metabolic Pathways from Observed Data Using Genetic Programming*. Stanford Medical Informatics Technical Report SMI-2000-0851. November 7, 2000. `http://smi-web.stanford.edu/pubs/SMI_Abstracts/SMI-2000-0851.html`

Koza, John R., Mydlowec, William, Lanza, Guido, Yu, Jessen, and Keane, Martin A. 2000b. Reverse engineering of metabolic pathways from observed data using genetic programming. In Altman, Russ B. Dunker, A. Keith, Hunter, Lawrence, Lauderdale, Kevin, and Klein, Teri (editors). *Pacific Symposium on Biocomputing'99*. Singapore: World Scientific. Pages 434–445.

Koza, John R., Mydlowec, William, Lanza, Guido, Yu, Jessen, and Keane, Martin A. 2001a. Automated reverse engineering of metabolic pathways by means of genetic programming. In

Kitano, Hiroaki. 2001. *Foundations of Systems Biology*. Cambridge, MA: The MIT Press. Pages 95–121.

Koza, John R., Mydlowec, William, Lanza, Guido, Yu, Jessen, and Keane, Martin A. 2001b. Automatic synthesis of both the topology and sizing of metabolic pathways using genetic programming. In Spector, Lee, Goodman, E., Wu, A., Langdon, William B., Voigt, H.-M., Gen, M., Sen, S., Dorigo, Marco, Pezeshk, S., Garzon, Max, and Burke, E. (editors). 2001. *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-2001*. San Francisco, CA: Morgan Kaufmann. Pages 57–65.

Koza, John R., and Rice, James P. 1991. Genetic generation of both the weights and architecture for a neural network. In *Proceedings of International Joint Conference on Neural Networks, Seattle, July 1991*. Los Alamitos, CA: IEEE Press. Volume II. Pages 397–404.

Koza, John R., and Rice, James P. 1992a. *Genetic Programming: The Movie*. Cambridge, MA: MIT Press.

Koza, John R., and Rice, James P. 1992b. *A Non-Linear Genetic Process for Data Encoding and for Solving Problems Using Automatically Defined Functions*. U.S. patent application filed May 11, 1992.

Koza, John R., and Rice, James P. 1992c. *A Non-Linear Genetic Process for Use with Plural Co-Evolving Populations*. U. S. patent No. 5,148,513. Filed September 18, 1990. Issued September 15, 1992.

Koza, John R., and Rice, James P. 1994a. *A Non-Linear Genetic Process for Data Encoding and for Solving Problems Using Automatically Defined Functions*. U. S. patent application filed May 11, 1992. U. S. patent No. 5,343,554. Issued August 30, 1994.

Koza, John R., and Rice, James P. 1994b. *A Non-Linear Genetic Process for Data Encoding and for Solving Problems Using Automatically Defined Functions*. U.S. patent 5,343,554. Filed May 11, 1992. Issued August 30, 1994.

Koza, John R., and Rice, James P. 1995. *Process for Problem Solving Using Spontaneously Emergent Self-Replicating and Self-Improving Entities*. U. S. patent application filed June 16, 1992. U. S. patent No. 5,390,282. Issued February 14, 1995.

Koza, John R., Rice, James P., and Roughgarden, Jonathan. 1992. Evolution of food foraging strategies for the Caribbean *Anolis* lizard using genetic programming. *Adaptive Behavior*. Volume 1, number 2, pages 47–74.

Koza, John R., Yu, Jessen, Keane, Martin A., and Mydlowec, William. 2000a. Evolution of a controller with a free variable using genetic programming. In Poli, Riccardo, Banzhaf, Wolfgang, Langdon, William B., Miller, Julian, Nordin, Peter, and Fogarty, Terence C. 2000. *Genetic Programming: European Conference, EuroGP 2000, Edinburgh, Scotland, UK, April 2000, Proceedings*. Lecture Notes in Computer Science. Volume 1802. Berlin, Germany: Springer-Verlag. Pages 91–105.

Koza, John R., Yu, Jessen, Keane, Martin A., and Mydlowec, William. 2000b. Use of conditional developmental operators and free variables in automatically synthesizing generalized circuits using genetic programming. *Proceedings of the Second NASA/DoD Workshop on Evolvable Hardware, July 13–15 2000, Palo Alto, California*. Los Alamitos, CA: IEEE Computer Society Press. Pages 5–15.

Kruiskamp, Marinum Wilhelmus. 1996. *Analog Design Automation using Genetic Algorithms and Polytopes*. Eindhoven, The Netherlands: Data Library Technische Universiteit Eindhoven.

Kruiskamp, Marinum Wilhelmus and Leenaerts, Domine. 1995. DARWIN: CMOS opamp synthesis by means of a genetic algorithm. *Proceedings of the 32nd Design Automation Conference*. New York, NY: Association for Computing Machinery. Pages 433–438.

Laing, Shoudan, Fuhrman, Stefanie, and Somogyi, Roland. 1998. REVEAL: A general reverse engineering algorithm for inference of genetic network architecture. In Altman, Russ B.

Dunker, A. Keith, Hunter, Lawrence, and Klein, Teri E. (editors). *Pacific Symposium on Biocomputing '98*. Singapore: World Scientific. Pages 18–29.

Langdon, William B. 1998. *Genetic Programming and Data Structures: Genetic Programming + Data Structures = Automatic Programming!* Amsterdam: Kluwer.

Langdon, W. B., Cantu-Paz, E., Mathias, K., Roy, R., Davis, D., Poli, R., Balakrishnan, K., Honavar, V., Rudolph, G., Wegener, J., Bull, L., Potter, M. A., Schultz, A. C., Miller, J. F., Burke, E., and Jonoska, N. (editors). 2002. *Proceedings of the 2002 Genetic and Evolutionary Computation Conference*. San Francisco, CA: Morgan Kaufmann.

Langdon, William B. and Poli, Riccardo. 2002. *Foundations of Genetic Programming*. Berlin: Springer-Verlag.

Lanza, Guido, Mydlowec, William, and Koza, John R. 2000. Automatic creation of a genetic network for the *lac* operon from observed data by means of genetic programming. Poster paper accepted at First International Conference on Systems Biology in Tokyo on November 14–16, 2000.

Lee, Sang Gug. 2001. *Low Voltage Balun Circuit*. U.S. patent 6,265,908. Filed December 15, 1999. Issued July 24, 2001.

Lee, Thomas H. 1998. *The Design of CMOS Radio-Frequency Integrated Circuits*. Cambridge: Cambridge University Press.

Linden, Derek S. 1997. *Automated Design and Optimization of Wire Antennas Using Genetic Algorithms*. Ph.D. Thesis. Department of Electrical Engineering and Computer Science. Massachusetts Institute of Technology.

Lindenmayer, Aristid. 1968. Mathematical models for cellular interactions in development, I & II. *Journal of Theoretical Biology*. Volume 18. Pages 280–315.

Liu, Yong, Tanaka, Kiyoshi, Iwata, Masaya, Higuchi, Tetsuya, and Yasunaga, Moritoshi (editors). 2001. *Evolvable Systems: From Biology to Hardware, 4th International Conference, ICES 2001, Tokyo, Japan, October 2001 Proceedings*. Lecture Notes in Computer Science, Volume 2210. Berlin: Springer-Verlag.

Lohn, Jason, Stoica, Adrian, Keymeulen, Didier, and Colombano, Silvano (editors). 2000. *Proceedings of the Second NASA/DoD Workshop on Evolvable Hardware, July 13–15 2000, Palo Alto, California*. Los Alamitos, CA: IEEE Computer Society Press.

Loomis, William F. and Sternberg, Paul W. 1995. Genetic networks. *Science*. Pages 269–649. August 4, 1995.

Luke, Sean. 1998. Genetic programming produced competitive soccer softbot teams for RoboCup97. In Koza, John R., Banzhaf, Wolfgang, Chellapilla, Kumar, Deb, Kalyanmoy, Dorigo, Marco, Fogel, David B., Garzon, Max H., Goldberg, David E., Iba, Hitoshi, and Riolo, Rick. (editors). *Genetic Programming 1998: Proceedings of the Third Annual Conference, July 22–25, 1998, University of Wisconsin, Madison, Wisconsin*. San Francisco, CA: Morgan Kaufmann. Pages 214–222.

Macbeth, Ian. 2002. FPAAs: Synthesis by construction. *EDN*. November 28, 2002. Pages 44.

Mahfoud, S. W. and Goldberg, David E. 1995. Parallel recombinative simulated annealing: A genetic algorithm. *Parallel Computing*. Amsterdam: Elsevier Science. Volume 21. Pages 1–28.

Man, K. F., Tang, K. S., Kwong, S., and Halang, W. A. 1997. *Genetic Algorithms for Control and Signal-Processing*. London: Springer-Verlag.

Man, K. F., Tang, K. S., Kwong, S., and Halang, W. A. 1999. *Genetic Algorithms: Concepts and Designs*. London: Springer-Verlag.

Marcano, Diogenes and Duran, Filinto. 1999. Synthesis of linear and planar arrays using genetic algorithms. In Rahmat-Samii, Yahya and Michielssen, Eric (editors). *Electromagnetic Optimization by Genetic Algorithms*. New York, NY: John Wiley and Sons. Chapter 6. Pages 157–179.

Marenbach, Peter, Bettenhausen, Kurt D., and Freyer, Stephan. 1996. Signal path oriented approach for generation of dynamic process models. In Koza, John R., Goldberg, David E., Fogel, David B., and Riolo, Rick L. (editors). *Genetic Programming 1996: Proceedings of the First Annual Conference, July 28–31, 1996, Stanford University*. Cambridge, MA: MIT Press. Pages 327–332.

Massey, John. 1970. *Compensated Resistance Bridge-Type Electrical Thermometer*. U.S. patent 3,541,857. Filed November 27, 1968. Issued November 24, 1970.

Maziasz, Robert L. and Hayes, John P. 1992. *Layout Minimization of CMOS Cells*. Boston: Kluwer.

Mazumder, Pinaki and Rudnick, Elizabeth M. (editors). 1999. *Genetic Algorithms for VLSI Design, Layout and Test Automation*. Upper Saddle River, NJ: Prentice Hall.

McAdams, Harley H. and Shapiro, Lucy. 1995. Circuit simulation of genetic networks. *Science*. Volume 269. Pages 650–656. August 4, 1995.

Mendes, Pedro and Kell, Douglas B. 1998. Nonlinear optimization of biochemical pathways: Applications to metabolic engineering and parameter estimation. *Bioinformatics*. 14(10)869–883.

Menon, P. K., Yousefpor, M., Lam, T., and Steinberg, M. L. 1995. Nonlinear flight control system synthesis using genetic programming. *Proceedings of 1995 Conference on Guidance, Navigation, and Control*. Reston, VA: American Institute of Aeronautics and Astronautics. Pages 461–470.

Messina, Paul, Sterling, Thomas, and Smith, Paul H. (editors). 1999. *Petaflops II: Second Conference on Enabling Technologies for Peta(fl)ops Computing, February 15–19, 1999, Santa Barbara*.

Miller, Julian, Thompson, Adrian, Thomson, Peter, and Fogarty, Terence C. (editors). 2000. *Evolvable Systems: From Biology to Hardware. Third International Conference, ICES 2000, Edinburgh, Scotland, UK, April 2000 Proceedings*. Lecture Notes in Computer Science. Volume 1801. Berlin, Germany: Springer-Verlag.

Miller, Julian, Tomassini, Marco, Lanzi, Pier Luca, Ryan, Conor, Tettamanzi, Andrea G. B., and Langdon, William B. (editors). 2001. *Genetic Programming: 4th European Conference, EuroGP 2001, Lake Como, Italy, April 2001 Proceedings*. Berlin: Springer.

Mittenthal, Jay E., Ao Yuan, Bertrand Clarke, and Scheeline, Alexander. 1998. Designing metabolism: Alternative connectivities for the pentose phosphate pathway. *Bulletin of Mathematical Biology*. Volume 60. Pages 815–856.

Moore, Gordon E. 1996. Can Moore's law continue indefinitely? *Computerworld Leadership Series*. 2(6)2–7. July 15, 1996.

Moretti, Gabe. 2002. The next wave: Synthesis tools help with mixed-signal designs. *EDN*. November 28, 2002. Pages 43–50.

Mydlowec, William and Koza, John. 2000. Use of time-domain simulations in automatic synthesis of computational circuits using genetic programming. *Late Breaking Papers at the 2000 Genetic and Evolutionary Computation Conference, Las Vegas, Nevada*. Pages 187–197.

Newbold, William F. 1962. *Square Root Extracting Integrator*. U.S. patent 3,016,197. Filed September 15, 1958. Issued January 9, 1962.

Newborn, Monty. 2002. *Deep Blue: An Artificial Intelligence Milestone*. New York: Springer.

Nordin, Peter. 1997. *Evolutionary Program Induction of Binary Machine Code and Its Application*. Munster, Germany: Krehl Verlag.

O'Connor, Daniel G. and Nelson, Raymond J. 1962. *Sorting System with N-Line Sorting Switch*. U.S. patent 3,029,413. Issued April 10, 1962.

Ogata, Katsuhiko. 1997. *Modern Control Engineering*. Third edition. Upper Saddle River, NJ: Prentice Hall.

Ohr, Stephan. 2002. Anadigm fields reconfigurable analog device. *EE Times*. September 2, 2002.

Olsson, Jan Roland. 1994a. Inductive functional programming using incremental program transformation. *Artificial Intelligence*. Volume 74. Pages 55–81.

Olsson, Jan Roland. 1994b. *Inductive Functional Programming Using Incremental Program Transformation*. Dr. Scient. thesis. University of Oslo.

O'Neill, Michael and Ryan, Conor. 2003. *Grammatical Evolution: Evolutionary Automatic Programming in an Arbitrary Language*. Boston: Kluwer Academic Publishers.

Osyczka, A. 1984. *Multicriterion Optimization in Engineering with FORTRAN programs*. Ellis Horwood Limited.

Pease, Robert. 1992. What's all this Muntzing stuff, anyhow? *Electronic Design*. July 23, 1992.

Pease, Robert. 1996. What's all this R-C filter stuff, anyhow? *Electronic Design*. March 18, 1996.

Pease, Robert. 1999. What's all this logarithmic stuff, anyhow? *Electronic Design*. August 19, 1999.

Philbrick, George A. 1956. *Delayed Recovery Electric Filter Network*. Filed May 18, 1951. U.S. patent 2,730,679. Issued January 10, 1956.

Poli, Riccardo, Nordin, Peter, Langdon, William B., and Fogarty, Terence C. 1999. *Genetic Programming: Second European Workshop, EuroGP'99. Proceedings*. Lecture Notes in Computer Science. Volume 1598. Berlin, Germany: Springer-Verlag.

Poli, Riccardo, Banzhaf, Wolfgang, Langdon, William B., Miller, Julian, Nordin, Peter, and Fogarty, Terence C. 2000. *Genetic Programming: European Conference, EuroGP 2000, Edinburgh, Scotland, UK, April 2000, Proceedings*. Lecture Notes in Computer Science. Volume 1802. Berlin, Germany: Springer-Verlag.

Prusinkiewicz, Przemyslaw and Hanan, James. 1980. *Lindenmayer Systems, Fractals, and Plants*. New York: Springer-Verlag.

Prusinkiewicz, Przemyslaw, and Lindenmayer, Aristid. 1990. *The Algorithmic Beauty of Plants*. New York: Springer-Verlag.

Ptashne, Mark. 1992. *A Genetic Switch: Phage λ and Higher Organisms*. Second Edition. Cambridge, MA: Cell Press and Blackwell Scientific Publications.

Quarles, Thomas, Newton, A. R., Pederson, D. O., and Sangiovanni-Vincentelli, A. 1994. *SPICE 3 Version 3F5 User*'s *Manual*. Department of Electrical Engineering and Computer Science, University of California. Berkeley, CA. March 1994.

Rahmat-Samii, Yahya and Michielssen, Eric (editors). 1999. *Electromagnetic Optimization by Genetic Algorithms*. New York, NY: John Wiley and Sons.

Rechenberg, Ingo. 1965. *Cybernetic solution path of an experimental problem*. Royal Aircraft Establishment, Ministry of Aviation, Library Translation 1112. Farnborough.

Rechenberg, Ingo. 1973. *Evolutionsstrategie: Optimierung Technischer Systeme nach Prinzipien der Biolgischen Evolution*. Stuttgart-Bad Cannstatt: Verlag Frommann-Holzboog.

Riolo, Rich and Worzel, William. 2003. *Genetic Programming: Theory and Practice*. Boston: Kluwer Academic Publishers.

Robertson, George. 1987. Parallel implementation of genetic algorithms in a classifier system. In Davis, Lawrence (editor). *Genetic Algorithms and Simulated Annealing* London: Pittman.

Ryan, Conor. 1999. *Automatic Re-engineering of Software Using Genetic Programming*. Amsterdam: Kluwer Academic Publishers.

Salamon, Peter, Sibani, Paolo, and Frost, Richard. 2002. *Facts, Conjectures, and Improvements for Simulated Annealing*. Philadelphia: Society for Industrial and Applied Mathematics.

Samuel, Arthur L. 1983. AI: Where it has been and where it is going. *Proceedings of the Eighth International Joint Conference on Artificial Intelligence*. Los Altos, CA: Morgan Kaufmann. Pages 1152–1157.

Sanchez, Eduardo and Tomassini, Marco (editors). 1996. *Towards Evolvable Hardware*. Lecture Notes in Computer Science, Volume 1062. Berlin: Springer-Verlag.

Schaeffer, Jonathan. 1997. *One Jump Ahead: Challenging Human Supremacy in Checkers*. New York: Springer.

Sechen, Carl. 1988. *VLSI Placement and Global Routing using Simulated Annealing*. Boston, MA: Kluwer.

Sheingold, Daniel H. (editor). 1976. *Nonlinear Circuits Handbook*. Norwood, MA: Analog Devices, Inc.

Sipper, Moshe, Mange, Daniel, and Perez-Uribe, Andres (editors). 1998. *Evolvable Systems: From Biology to Hardware. Second International Conference, ICES 98, Lausanne, Switzerland, September 1998 Proceedings*. Lecture Notes in Computer Science 1478. Berlin: Springer-Verlag.

Smith, Steven F. 1980. *A Learning System Based on Genetic Adaptive Algorithms*. Ph.D. dissertation. Pittsburgh, PA: University of Pittsburgh.

Song, Bang-Sup and Harjani, Ramesh. 1995. In Chen, Wai-Kai (editor). *The Circuits and Filters Handbook*. Boca Raton, FL: CRC Press. Pages 2072–2127.

Spector, Lee, Barnum, Howard, and Bernstein, Herbert J. 1998. Genetic programming for quantum computers. In Koza, John R., Banzhaf, Wolfgang, Chellapilla, Kumar, Deb, Kalyanmoy, Dorigo, Marco, Fogel, David B., Garzon, Max H., Goldberg, David E., Iba, Hitoshi, and Riolo, Rick. (editors). *Genetic Programming 1998: Proceedings of the Third Annual Conference*. San Francisco, CA: Morgan Kaufmann. Pages 365–373.

Spector, Lee, Barnum, Howard, and Bernstein, Herbert J. 1999. Quantum computing applications of genetic programming. In Spector, Lee, Langdon, William B., O'Reilly, Una-May, and Angeline, Peter (editors). *Advances in Genetic Programming 3*. Cambridge, MA: The MIT Press. Pages 135–160.

Spector, Lee, Barnum, Howard, Bernstein, Herbert J., and Swamy, N. 1999. Finding a better-than-classical quantum AND/OR algorithm using genetic programming. In IEEE. *Proceedings of 1999 Congress on Evolutionary Computation*. Piscataway, NJ: IEEE Press. Pages 2239–2246.

Spector, Lee, and Bernstein, Herbert J. 2003. Communication capacities of some quantum gates, discovered in part through genetic programming. In Shapiro, Jeffrey H. and Hirota, Osamu (editors). *Proceedings of the Sixth International Conference on Quantum Communication, Measurement, and Computing*. Paramus, NJ: Rinton Press.

Spector, Lee, Goodman, E., Wu, A., Langdon, William B., Voigt, H. -M., Gen, M., Sen, S., Dorigo, Marco, Pezeshk, S., Garzon, Max, and Burke, E. (editors). 2001. *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-2001*. San Francisco, CA: Morgan Kaufmann.

Spector, Lee, Langdon, William B., O'Reilly, Una-May, and Angeline, Peter (editors). 1999. *Advances in Genetic Programming 3*. Cambridge, MA: The MIT Press.

Spector, Lee and Stoffel, Kilian. 1996a. Ontogenetic programming. In Koza, John R., Goldberg, David E., Fogel, David B., and Riolo, Rick L. (editors). 1996. *Genetic Programming 1996: Proceedings of the First Annual Conference, July 28–31, 1996, Stanford University*. Cambridge, MA: MIT Press. Pages 394–399.

Spector, Lee and Stoffel, Kilian. 1996b. Automatic generation of adaptive programs. In Maes, Pattie, Mataric, Maja J., Meyer, Jean-Arcady, Pollack, Jordan, and Wilson, Stewart W. (editors). 1996. *From Animals to Animats 4: Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*. Cambridge, MA: The MIT Press. Pages 476–483.

Sripramong, Thanwa and Toumazou, Christofer. 2002. The invention of CMOS amplifiers using genetic programming and current-flow analysis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. 21(11). November 2002. Pages 1237–1252.

Stender, Joachim (editor). 1993. *Parallel Genetic Algorithms*. Amsterdam: IOS Publishing.

Stephens, C. R. and Waelbroeck, H. 1997. Effective degrees of freedom in genetic algorithms and the block hypothesis. In Back, Thomas (editor). 1997. *Genetic Algorithms: Proceedings of the Seventh International Conference*. San Francisco, CA: Morgan Kaufmann. Pages 34–40.

Sterling, Thomas L. 1998a. Beowulf-class clustered computing: Harnessing the power of parallelism in a pile of PCs. In Koza, John R., Banzhaf, Wolfgang, Chellapilla, Kumar, Deb, Kalyanmoy, Dorigo, Marco, Fogel, David B., Garzon, Max H., Goldberg, David E., Iba, Hitoshi, and Riolo, Rick L. (editors). *Genetic Programming 1998: Proceedings of the Third Annual Conference, July 22–25, 1998, University of Wisconsin, Madison, Wisconsin*. San Francisco, CA: Morgan Kaufmann. Pages 883–887.

Sterling, Thomas L. 1998b. *Proceedings of Petaflops-Systems Operations Working Review, Bodega Bay, California, June 1–5, 1998*.

Sterling, Thomas L. and Foster, Ian. 1996a. *Proceedings of Petaflops Architecture Workshop (PAWS '96), April 22–25, 1996*.

Sterling, Thomas L. and Foster, Ian. 1996b. *Proceedings of Petaflops System Software Summer Study (Peta Soft '96), June 17–21, 1996*.

Sterling, Thomas L., Salmon, John, and Becker, Donald J., and Savarese, Daniel F. 1999. *How to Build a Beowulf: A Guide to Implementation and Application of PC Clusters*. Cambridge, MA: MIT Press.

Steudel, Goetz Wolfgang. 1973. *Reference Voltage Generator and Regulator*. U.S. patent 3,743,923. Filed December 2, 1971. Issued July 3, 1973.

Stoica, Adrian, Keymeulen, Didier, and Lohn, Jason (editors). 1999. *Proceedings of the First NASA/DOD Workshop on Evolvable Hardware, Pasadena, California, July 19–21, 1999*. Los Alamitos, CA. IEEE Computer Society.

Stoica, Adrian, Lohn, Jason, Katz, Rich, Keymeulen, Didier and Zebulum, Ricardo Salem (editors). 2002. *Proceedings of 2002 NASA/DoD Conference on Evolvable Hardware*. Los Alamitos, CA: IEEE Computer Society.

Stoica, Adrian, Zebulum, Ricardo, and Keymeulen, Didier. 2001. Polymorphic electronics. In Liu, Yong, Tanaka, Kiyoshi, Iwata, Masaya, Higuchi, Tetsuya, and Yasunaga, Moritoshi (editors). *Evolvable Systems: From Biology to Hardware, 4th International Conference, ICES 2001, Tokyo, Japan, October 2001 Proceedings*. Lecture Notes in Computer Science, Volume 2210. Berlin: Springer-Verlag. Pages 291–302.

Streeter, Matthew J., Keane, Martin A., and Koza, John R. 2002a. Iterative refinement of computational circuits using genetic programming. In Langdon, W. B., Cantu-Paz, E., Mathias, K., Roy, R., Davis, D., Poli, R., Balakrishnan, K., Honavar, V., Rudolph, G., Wegener, J., Bull, L., Potter, M. A., Schultz, A. C., Miller, J. F., Burke, E., and Jonoska, N. (editors). 2002. *Proceedings of the 2002 Genetic and Evolutionary Computation Conference*. San Francisco, CA: Morgan Kaufmann. Pages 877–884.

Streeter, Matthew J., Keane, Martin A., and Koza, John R. 2002b. Routine duplication of post-2000 patented inventions by means of genetic programming. In Foster, James A., Lutton, Evelyne, Miller, Julian, Ryan, Conor, and Tettamanzi, Andrea G. B. (editors). 2002. *Genetic Programming: 5th European Conference, EuroGP 2002, Kinsale, Ireland, April 2002 Proceedings*. Berlin: Springer. Pages 26–36.

Streeter, Matthew J., Keane, Martin A., and Koza, John R. 2003. Automatic synthesis using genetic programming of both the topology and sizing for five post-2000 patented analog and mixed analog-digital circuits. *Proceedings of 2003 Southwest Symposium on Mixed-Signal Design, February 23–25, 2003, Las Vegas, Nevada, U.S.A*. Pages 5–10.

Stutzman, Warren. L. and Thiele, Gary A. 1998. *Antenna Theory and Design*. Second edition. New York, NY: John Wiley and Sons.

Sweriduk, G. D., Menon, P. K., and Steinberg, M. L. 1998. Robust command augmentation system design using genetic search methods. *Proceedings of 1998 Conference on Guidance, Navigation, and Control*. Reston, VA: American Institute of Aeronautics and Astronautics. Pages 286–294.

Sweriduk, G. D., Menon, P. K., and Steinberg, M. L. 1999. Design of a pilot-activated recovery system using genetic search methods. *Proceedings of 1998 Conference on Guidance, Navigation, and Control*. Reston, VA: American Institute of Aeronautics and Astronautics.

Tanese, Reiko. 1989. *Distributed Genetic Algorithm for Function Optimization*. Ph.D. dissertation. Department of Electrical Engineering and Computer Science. University of Michigan.

Teller, Astro. 1996a. *Evolving Programmers: SMART Mutation*. Technical Report CMU-CS-96. Computer Science Department, Carnegie Mellon University.

Teller, Astro. 1996b. Evolving programmers: The co-evolution of intelligent recombination operators. In Angeline, Peter J. and Kinnear, Kenneth E. Jr. (editors). *Advances in Genetic Programming 2*. Cambridge, MA: The MIT Press.

Teller, Astro. 1998. *Algorithm Evolution with Internal Reinforcement for Signal Understanding*. Ph.D. Thesis. Computer Science Department. Carnegie Mellon University. Pittsburgh, Pennsylvania.

Teller, Astro. 1999. The internal reinforcement of evolving algorithms. In Spector, Lee, Langdon, William B., O'Reilly, Una-May, and Angeline, Peter (editors). 1999. *Advances in Genetic Programming 3*. Cambridge, MA: The MIT Press.

Teller, Astro, and Veloso, Manuela. 1995a. *Learning Tree Structured Algorithms for Orchestration into an Object Recognition System*. Technical Report CMU-CS-95-101. Computer Science Department, Carnegie Mellon University.

Teller, Astro, and Veloso, Manuela. 1995b. Program evolution for data mining. In Louis, Sushil (editor). Special Issue on Genetic Algorithms and Knowledge Bases. *The International Journal of Expert Systems*. JAI Press. (3)216–236.

Teller, Astro, and Veloso, Manuela. 1995c. A controlled experiment: evolution for learning difficult problems. *Proceedings of Seventh Portuguese Conference on Artificial Intelligence*. Springer-Verlag. Pages 165–76.

Teller, Astro, and Veloso, Manuela. 1995d. Algorithm Evolution for Face Recognition: What Makes a Picture Difficult? *Proceedings of the IEEE International Conference on Evolutionary Computation*. IEEE Press.

Teller, Astro, and Veloso, Manuela. 1995e. Language Representation Progression in PADO. *Proceedings of AAAI Fall Symposium on Artificial Intelligence*. Menlo Park, CA: AAAI Press.

Teller, Astro and Veloso, Manuela. 1996. PADO: A new learning architecture for object recognition. In Ikeuchi, Katsushi and Veloso, Manuela (editors). *Symbolic Visual Learning*. Oxford University Press. Pages 81–116.

Teller, Astro, and Veloso, Manuela. 1997. Neural programming and an internal reinforcement policy. In Yao, Xin, Kim, Jong-Hwan, and Furuhashi, T. (editors). *Simulated Evolution and Learning*. Lecture Notes in Artificial Intelligence. Volume 1285. Heidelberg, Germany: Springer-Verlag. Pages 279–286.

Thompson, Adrian. 1996. Silicon evolution. In Koza, John R., Goldberg, David E., Fogel, David B., and Riolo, Rick L. (editors). 1996. *Genetic Programming 1996: Proceedings of the First Annual Conference, July 28–31, 1996, Stanford University*. Cambridge, MA: MIT Press. Pages 444–452.

Thompson, Adrian. 1998. *Hardware Evolution: Automatic Design of Electronic Circuits in Reconfigurable Hardware by Artificial Evolution*. Conference of Professors and Heads of Computing / British Computer Society Distinguished Dissertation series. Berlin: Springer-Verlag.

Tomita, Masaru, Hashimoto, Kenta, Takahashi, Kouichi, Shimizu, Thomas Simon, Matsuzaki, Yuri, Miyoshi, Fumihiko, Saito, Kanako, Tanida, Sakura, Yugi, Katsuyuki, Venter, J. Craig, Hutchison, Clyde A. III. 1999. E-CELL: Software environment for whole cell simulation. *Bioinformatics*. Volume 15 (1)72–84.

Turing, Alan M. 1948. Intelligent machinery. Reprinted in Ince, D. C. (editor). 1992. *Mechanical Intelligence: Collected Works of A. M. Turing*. Amsterdam: North Holland. Pages 107–127. Also reprinted in Meltzer, B. and Michie, D. (editors). 1969. *Machine Intelligence 5*. Edinburgh: Edinburgh University Press.

Turing, Alan M. 1950. Computing machinery and intelligence. *Mind*. 59(236)433–460. Reprinted in Ince, D. C. (editor). 1992. *Mechanical Intelligence: Collected Works of A. M. Turing*. Amsterdam: North Holland. Pages 133–160.

Uda, S. 1926. Wireless beam of short electric waves. *Journal of the IEE (Japan)*. March 1926. Pages 273–282.

Uda, S. 1927. Wireless beam of short electric waves. *Journal of the IEE (Japan)*. March 1927. Pages 1209–1219.

Ullman, Jeffrey D. 1984. *Computational Aspects of VLSI*. Rockville, MD: Computer Science Press.

Van Valkenburg, M. E. 1982. *Analog Filter Design*. Fort Worth, TX: Harcourt Brace Jovanovich.

Villagran, Victor and Sbarbaro, Daniel. 1998. A new approach for turning PID controller based on iterative learning. *Proceedings of the 1998 IEEE International Conference on Control Applications*. Volume I. Pages 139–143.

Vladimirescu, Andrei. 1994. *The SPICE Book*. New York, NY: John Wiley and Sons.

Voit, Eberhard O. 2000. *Computational Analysis of Biochemical Systems*. Cambridge: Cambridge University Press.

Wakerly, John F. 1990. *Digital Design Principles and Practices*. Englewood Cliffs, NJ: Prentice Hall.

Webb, Edwin C. 1992. *Enzyme Nomenclature 1992: Recommendations of the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology*. San Diego, CA: Academic Press.

Whitley, Darrell, Goldberg, David, Cantu-Paz, Erick, Spector, Lee, Parmee, Ian, and Beyer, Hans-Georg (editors). 2000. *GECCO-2000: Proceedings of the Genetic and Evolutionary Computation Conference, July 10–12, 2000, Las Vegas, Nevada*. San Francisco: Morgan Kaufmann.

Whitley, Darrell, Gruau, Frederic, and Preatt, Larry. 1995. Cellular encoding applied to neurocontrol. In Eshelman, Larry J. (editor). *Proceedings of the Sixth International Conference on Genetic Algorithms*. San Francisco, CA: Morgan Kaufmann. Pages 460–467.

Whitley, Darrell. 2000. Functions as permutations: Regarding no free lunch, Walsh analysis and summary statistics. In Schoenauer, Marc, Deb, Kalyanmoy, Gunter, Rudolph, Yao, Xin, Lutton, Evelyne, Merelo, Juan Julian, and Schwefel, Hans-Paul (editors). 2000. *Parallel Problem Solving from Nature: 6th International Conference, Paris, France, September 2000 Proceedings*. Berlin: Springer. Page 169–178.

Williams, Arthur B. and Taylor, Fred J. 1995. *Electronic Filter Design Handbook*. Third Edition. New York, NY: McGraw-Hill.

Witczak, Marcin. 2003. *Identification and Fault Detection of Non-Linear Dynamic Systems*. University of Zielona Gora Press.

Wolpert, D. H. and Macready, W. G. 1997. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*. 1(1) 67– 82. April 1997.

Wong, D. F., Leong, H. W., and Liu. C. L. 1988. *Simulated Annealing for VLSI Design*. Boston, MA: Kluwer.

Wong, Man Leung and Leung, Kwong Sak. 2000. *Data Mining Using Grammar Based Genetic Programming and Applications*. Amsterdam: Kluwer Academic Publishers.

Yagi, H. 1928. Beam transmission of ultra short waves. *Proceedings of the IRE*. Volume 26: Pages 714–741. June 1928.

Yu, Jessen, Keane, Martin A., and Koza, John R. 2000. Automatic design of both topology and tuning of a common parameterized controller for two families of plants using genetic programming. In *Proceedings of Eleventh IEEE International Symposium on Computer-Aided Control System Design (CACSD) Conference and Ninth IEEE International Conference on Control Applications (CCA) Conference, Anchorage, Alaska, September 25–27, 2000*.

Yuh, Chiou-Hwa, Bolouri, Hamid, and Davidson, Eric H. 1998. Genomic cis-regulatory logic: Experimental and computational analysis of a sea urchin gene. *Science*. 279. Pages 1896–1902.

Zebulum, Ricardo Salem, Pacheco, Marco Aurelio C., and Vellasco, Marley Maria B. R. 2002. *Evolutionary Electronics: Automatic Design of Electronic Circuits and Systems by Genetic Algorithms*. Boca Raton, FL: CRC Press.

Ziegler, J. G. and Nichols, N. B. 1942. Optimum settings for automatic controllers. *Transactions of ASME*. (64) 759–768.

Zitzler, Eckart, Deb, Kalyanmoy, Thiele, Lothar, Coello Coello, Carlos A., and Corne, David (editors). 2001. *Evolutionary Multi-Criterion Optimization, First International Conference, EMO 2001, Zurich, Switzerland, March 2001, Proceedings*. Lecture Notes in Computer Science. Volume 1993. Berlin, Germany: Springer-Verlag.

Zobel, Otto Julius. 1926. *Electrical Network and Method of Transmitting Electric Currents*. Filed August 9, 1922. U.S. patent 1,603,305. Issued October 19, 1926.

Zobel, Otto Julius. 1925. *Wave Filter*. Filed January 15, 1921. U.S. patent 1,538,964. Issued May 26, 1925.

Zobel, Otto Julius. 1928. Distortion correction in electrical circuits with constant resistance networks. *Bell Systems Technical Journal*. July 1928. Page 438.

Acknowledgment Concerning Figures

Figures 2.1, 4.8, 4.12, 10.4, 16.5, 16.10, 16.11, 16.12, 16.13, and 16.14 of this book were reprinted from *Genetic Programming III: Darwinian Invention and Problem Solving* by John R. Koza, Forrest H Bennett III, David Andre and Martin A. Keane, pages 38, 582, 835, 403, 589, 666, 667, 667, 637, and 641, respectively, Copyright 1999, with permission from Elsevier.

Index

Aarts, E. 61, 506
Aaserud, O. 130
Abelson, H. 209
absolute value (ABSV) 54, 84
AC small signal analysis (AC sweep) 187, 295, 315, 327, 335, 343–4, 349
active filter 434
  integrated, dynamically tunable 332
active lowpass filter 301
  with free variable for passband boundary 332
    AI ratio for 339
    preparatory steps for 333–5
    results for 335–8
    routineness for 339
adder 53, 55, 261
AI ratio 4–5, 13, 112, 119, 120, 123, 124, 127, 145, 153, 159, 162, 168, 174, 197, 203, 227, 275, 278, 291, 300, 312, 324, 332, 339, 348, 357, 385, 412, 419, 482
  for genetically evolved results 529
Alpha 524
  microprocessor 516
  parallel computer system 516–17, 523, 525, 527
Altshuler, E.E. 206–9, 215, 218, 484–5
amplification 414
  factor 54
amplifiers 176, 415
  40/60 dB amplifier
    preparatory steps for 364–6
    results for 366
  60 dB amplifier problem with layout
    AI ratio for 203
    preparatory steps for 197–9
    results for 199–202
    routineness for 202–3
    problem of reducing distortion 413
analog
  computational circuits 324
    design of 425
  electrical circuits 129–30
    topology and sizing of 130
    of naturally occurring genetic operations 29, 31
analog-digital circuit 421
Andersson, B. 62
Andre, D. 4, 40–1, 45–7, 62–3, 70, 91, 100, 104, 130, 134–5, 140, 142, 156, 162–3, 169, 174, 177, 196, 263, 280, 302, 304, 311, 342, 415, 421, 450, 480, 486–7, 494, 496, 511, 513, 515, 523, 525, 539, 551
Angeline, P.J. 47, 62
antenna-creating program tree 212
antenna design techniques, conventional 207
antennas, automatic synthesis of 205
  AI ratio for 220
  endfire 208
  geometry 206–7
  illustrative problem of 207–9
  preparatory steps for 212–16
  repertoire of functions and terminals 209–11
  results for 216
  simulation 215
  wire segments in 69
Ao Yuan, B.C. 279
architecture-altering operations 32, 34–5, 72, 314, 539
arithmetic functions 34–6
arithmetic logic unit (ALU) circuit 130
  two-instruction, automatic synthesis of 159
  problem
    AI ratio for 162
    preparatory steps for 160–61
    results for 161
    routineness for 161
arithmetic-performing subtree 33, 69–71, 103, 140, 178, 187, 283, 292–3, 308, 326, 343, 434
Arkin, A. 229–30
Armstrong, C.M. 15
Armstrong, E.H. 17, 20
  positive type of feedback 415
arrays
  linear and planar 206
  thinned, design of 206

artificial intelligence 3
  and machine learning, toy problems from various fields 43
"artificial-to-intelligence" ratio *see* AI ratio
Åström K.J. 50, 57–8, 113–16, 119–20, 122–3, 283, 287–9, 292, 295, 367, 375–7, 382
Åström and Hägglund
  block 393
  four families of plants 291, 367, 370–2, 389, 411
  PID controller 114, 116–18, 122–4, 287–9, 295–7, 299, 380, 390, 393–4, 398, 400, 407
    disturbance rejection ITAE of 387, 394, 401, 497
    maximum sensitivity of 394, 401, 407
    minimum attenuation of 377, 381, 394
    performance of 373–4
    reciprocal of minimum attenuation of 401, 407
    setpoint ITAE of 387, 394, 401, 407
    transfer function for 370
  PID tuning rules 22, 27, 368, 370, 380, 384–5, 388, 398, 400, 404, 407, 526
    disturbance rejection ITAE of 377, 381,
    maximum sensitivity of 377, 381
    reciprocal of minimum attenuation of 377, 381
    setpoint ITAE of 377, 381
  sub-controller 391, 404
  surfaces 374
  techniques in 115, 283
  tuning rules 9, 371–2, 369, 374, 391, 412
ATP 230, 273
  cofactor 231, 257
AT&T 15, 17, 21
attenuator 415
audio power amplifier 306
automated problem-solving
  method 101
  orientations 96
automatic controllers 53
  synthesis, possible approach using genetic programming 62–3
automatically defined copy (ADC) 486
automatically defined function (ADF) 39–40, 66, 68–9, 106–7, 282, 351, 486
automatically defined iterations (ADI) 40, 494
automatically defined loops (ADL) 40, 495
automatically defined recursions (ADR) 40, 495
automatically defined stores (ADS) 40, 495
Aytur, T.S. 422–3, 437, 452, 553

Babenezhad, J.N. 425
Babovic, V. 47
Bade, S.L. 142
Bagchi, T.P. 93, 480
Balakrishnan, K. 47
Balanis, C.A. 205, 219
balun circuits 427, 450
Banzhaf, W. 47, 62
Barratt, C.H. 50, 58
Becker, D.J. 515
beneficial "combination[s] of genes" (schemata) 509
Bennett III, F. H. 4, 18, 40–1, 45–7, 62–3, 70, 80, 91, 100, 104, 110, 118, 123, 127, 130, 134–5, 140, 142–3, 151, 156, 157, 161–3, 169, 174, 177, 196, 263, 280, 302, 304, 311, 342, 415, 421, 450, 486–7, 494–6, 511, 513, 515, 523, 525, 539, 551
Beowulf-style parallel cluster computer system 95, 390, 516–17
Beowulf systems, home-built 515
Berkeley SPICE, SPICE3 version of 78
Bettenhausen, K.D. 62
Beyer, H.-G. 47
b-galactosidase 222
bifurcation point 261, 274
Bishop, R.H. 50, 87–8, 104–6, 119
Black, H.S. 17–20, 413–14, 417, 419, 509, 553
  negative feedback 413, 415
  patents 415
Blickle, T. 47
block diagram 58, 64
  for controllers 74
  lines in 55
  of plant and PID controller 54
blocks 53
  tuning of 54
Bode plots 58, 127
Bolouri, H. 221, 229
Boltzmann equation 61, 506
books on genetic programming
  main points of 42–6
Boolean function 154
Boutin, N. 306–7, 311
Bower, J.M. 229
Boyd, S.P. 50, 58
Brave, S. 45, 47, 130
breath-first order of evaluation 304
British Patent Office 21, 300
Bryson, A.E. 50
Bull, L. 47
Burke, E. 47
Burke, G.J. 205

C 42
C++ 42
Callado-Vides, J. 230

Callender A. 56–7, 111, 300, 367, 509, 552
Callender and Stevenson patent 1939 112, 299
Campbell, G.A. 195, 323, 552
  filter circuit 196
  ladder topology 8
  patented ladder filter 324
Cantu-Paz, E. 47, 480–1, 515
capacitor 84
  parameterized 497
  sizing of 318
CAP gene 221–2
Carley, L.R. 176
cascodes 480, 486
catalysts 234
cathode-ray oscilloscopes 136
Cauer, W. 552
  elliptic filter 8, 324
Checkers 5
Chellapilla, K. 47
chemical reaction
  cycles, preparatory steps 263–7
  functions 69, 250
  networks, representation of 263
  rates of 236
  types of 234–49
Cheng, V.H.L. 61
chess 4
Chinook checker-playing computer program 5
chromosome 503–4, 508
  for circuit consisting for two identical T-sections 491
  of offspring 502
  strings 209, 507
Chung, D.H. 157, 552
Church-Turing thesis 43
Cipriani, S. 424–5, 427, 442, 553
circuit-constructing
  functions 178
  program tree 132, 139–40, 177, 492, 501
    developing into single T-section 493
    evaluation of each individual 187
    functions in 133
    preservation of syntactic validity and executability of 501
  terminals in 133
circuits
  automatic synthesis of 129
    commercial practicality of genetic programming for 478–81
    of topology and sizing of 131–5
  computational 162
    cubic 425
  conversion 434
  defined by chromosomes 501
  designing, process of 175
  diagram 131
  laid-out 175
  sizing 129, 319, 329, 336
  topology 129, 329, 336
    non-symmetric 451
    and sizing, placement, and routing, automatic synthesis of 175–86, 202
closed-loop
  feedback system 51
  system bandwidth 115
Coelho, L.de S. 61
Coello Coello, C.A. 93, 480
Cohn, J.M. 176
Comisky, W. 219
command signal 50, 67
commercial circuits 167
Comombano, S. 135
component-creating functions 178, 179, 305
  for *npn* transistors, *pnp* transistors 198
computational effort 44, 174
computational power, exponential growth in 515
computer
  code for implementing genetic programming 42
  evolution of 159–62
  power and genetic programming
    effect of order-of-magnitude on results 526–7
    increasing, results in synchrony 25
  programs, automatically creating 45–6
  systems, five,
    qualitative nature of results produced by 524
    used in 15-year period 523–4
  time 518
conditional operators 54, 341
connection list 64
connective functions 210
"constant *K*"
  highpass filter, design formulas for 352
  ladder
    classical 351
    sections 347
  lowpass ladder filter 354
constrained syntactic structure 33, 38–9, 69, 71, 140, 210, 292–3, 307, 309, 314, 358, 434
  for program trees 73
construction-continuing subtree 140, 198, 343
control
  loops 57
  parameters 30, 95, 115, 539–49
  variable 51, 67
CONTROLLER_OUTPUT terminal 67

controllers 50–2
additional representations of 73–86
automatic synthesis of 49
of topology and tuning of 64–73
design
considerations for 52–3
possible techniques for 58–63
genetically evolved 87
parameterized, for two families of plants 291
AI ratio for 300
human-competitiveness of result for 299
preparatory steps for 292–6
results for 296–300
routineness for 300
parsimony 402
and plant, representation
as connection list 75–7
as SPICE netlist 78
representation
by block diagram 53–8
as LISP symbolic(s) expression 74
in Mathematica 75
as program tree 74
of transfer function 73
signal-processing blocks 62–3, 281
synthesis by genetic and evolutionary computation 62
topology and tuning for 63
for two-lag plant 87
copy body branch (CBB) 486
copy control branch (CCB) 486
Corne, D. 93
Crawford, L.S. 61
crossover 29, 31, 34, 501, 503
fragment 511
and mutation operations, comparison of 511
operation 35, 38, 305, 492, 501–2
disruptive 302
two-offspring version of 32, 38
(woofer-tweeter) filter, genetically evolved 497
cubic function generator 9, 424–7
fitness measure for 442
results for 461–6
test fixture for 432
cubing circuit 361
cultural search 7
current
mirrors 480, 486
sink 432
cycles 59
cyclic graph 134
labeled 131–2
Daida, J. 47
Darlington emitter-follower 8, 486, 509
section 480, 512
Darlington, S. 552
Darwinian selection 33, 509
probabilistic 506
Daun-Lindberg, T.C. 422, 424,427, 440, 553
and Miller, parallel FET transistor structure 454
Davidson, E.H. 221, 230
Davis, D. 47
DC
operating point analysis 422
sweeps 162
dead time 292, 391
Deb, K. 47, 93, 480
Deep Blue 4
de Garis, H. 135
delay 54
delayed-recovery 147
`DELAY` function 121
deletion 72
Dellinger, J.H. 219
depth-first evaluation 304–5
derivative (D-type) control 56
design of complex structure 49
Deutsch-Jozsa “early promise” problem 8
development
approach 63
process 186
analogs of 29
in problem solving 500
developmental genetic programming 41–2
development-controlling function 140, 178, 186
`NOOP` 198
development-controlling terminals 146
Dewell, L.D. 61
D’haeseller, P. 230
diacyl-glycerol 230–1, 262, 273
`DIFFERENTIATOR` block 56
digital circuits 129
digital-to-analog (DAC) converter 425
circuit 165
directors 208
discrete-time problems 62
DiSessa, A. 209
distortion 414
disturbance rejection *see* integral of time-weighted absolute error
divider 54
division (`DIVV`) 84
Dobkin, R.C. 551
Dobkin-Widlar circuit 551
domain knowledge 96, 99–101
human-supplied 101

dominant pole design technique 369
donor parents 474–7
Dorf and Bishop controller 87–8, 104, 109
Dorf, R.C. 50, 87–8, 104–6, 119
Dorigo, M. 47
double-bandpass filter 496
  circuit, genetically evolved 495–6
Drechsler, R. 135
driven elements 207
DSP-driven algorithm 425
D-type control 103
dummy variable (formal parameter) 499
Dunlap, F. 63, 130, 135, 304
duplication 72
Duran, F. 206

E-CELL cell simulation model 231, 234, 266
Eiben, A.E. 47
electric filter network 147
electrical circuit
  diagram representing 238
  design process for 129
    of electrical circuit of resistors and capacitors with gain greater than one 135
electrical terminology for modeling plants and controllers 52
embryo 132, 177, 313–14, 342, 428
  modifiable wires in 139
embryonic circuit 63
embryonic neural network 63
embryonic pattern 63
embryonic structure, applying functions in program tree o 63
`END` terminal 210
energy level 59
enzymes 234–5, 279
Escherichia coli 221
established beliefs 413
  overcoming 20–1
Euro-GP conferences 47
evolutionary methods 59
evolutionary process 103
executional steps of genetic programming 10
explicit test fixture, automatic circuit synthesis without 168–74

fatty acid 230–1, 257
feedback
  from plant to controller 50
  transfer function 118
field-effect transistors (FETs) 415, 424, 440
  model `IRFZ44` 415
field-programmable analog array (FPAA) 142, 147
field-programmable gate array (FPGA) 142
field-programmable transistor array (FPTA) 142, 147
filter, attenuation of 169
First International Conference on Evolutionary Multi-Criterion Optimization 93
`FIRST-PRODUCT` function 252, 265
fitness 33, 59
  cases 33, 139
fitness measure 12, 30, 33, 35, 59, 131, 145, 230
  three-element 414
  two-element 280
fitness-proportionate selection 37
fitness-selected parent 510
fixed hard-wired external entity 139
fixed-length strings 302, 485
Fleming, P.J. 61
floating embryo 169, 307, 333, 428, 479
floating-point
  MIPS 426
  number 210, 484, 504
Fogel, D.B. 47, 62
Forgarty, T.C. 47, 135
Foster, I. 515
four-argument `DIV_SIGNAL` function 66
Fourier analysis 422, 436–7
four-rung ladder 194
Francone, F.D. 47
free variables 282, 293, 358, 363, 370, 391
frequency of beneficial genes 509
frequency-domain
  analysis 422
  behavior 501
  parameters 292
  simulations 525
frequency-sweep analyses 436
frequently-used combinations of components 480
Freyer, S. 62
Frost, R. 61, 506
Fuhrman, S. 230
function-defining branches 73
functions
  repertoire of 65–6, 209
  set 30
    identification 11
    and terminal sets 30, 144
  and terminals 533–8
Furuya, T. 135

Gacs-Kurdyumov-Levin (GKL) rule 9
gain 53
  block 55–6, 400
  function 85, 103
Garey, M.R. 176
Garrod, D.J. 176

Garzon, M. 47
Gaussian circuits 162
Gaussian mutation
  operation 71
  on perturbable numerical values 73
Gaussian perturbation 71
Gelatt, C.D. 61, 506
Gen, M. 47
genes
  combination of 507–8, 513
  deletion 29
  duplication 29
genetic algorithm 59
  and genetic programming, search by 61
  operating on fixed-length character strings 207–8, 494–5, 503–4
  operating on variable-length strings 495
genetic diversity 20
Genetic and Evolutionary Computation Conference (GECCO) 47
genetic and evolutionary computation, techniques of 515
genetic network problem
  AI ratio for 227
  preparatory steps, 224–5
  results 225
genetic networks
  automatic synthesis of 221
  functions, repertoire of 223
  representation by computer programs 223–4
genetic operations 34
genetic programming 59
  advanced features of 38–42
  as automated invention machine 15–23, 530
  automatic creation of parameterized topologies 530
  executional steps of 31–8
  flowchart of genetic programming 32
  with increasing computer power 531
  paradigm 43
  preparatory steps of 29–31
  sources of additional information 46–8
  system 96
Genetic Programming Conference 47
Genetic Programming Problem Solver (GPPS) 41, 499
  architecture-altering operations in 500
genetic search method 514
genetic switches 69
geometry table 205, 207
Getreu, I. 129
Gilbert, B. 425
`GLUCOSE` 221–2
Glycerol 230–1, 253–5, 258
Glycerol-3-Phosphate 262
Goddard, R. 17
Goldberg, D.E. 47, 208, 480, 506, 514–15
Goodman, E. 47
gradient methods 59
  search by 60, 505–6, 511–12, 514
graph isomorphism algorithm 450
greedy search methods 505
Green, M. 551
Grimbelby, J.B. 135
Grover's database search problem 8
Gruau, F. 42, 62–3, 132

Haeusler, J. 551
Hägglund, T. 50, 57–8, 113–16, 119–20, 122–3, 283, 287–9, 292, 295, 367, 375, 377, 382
Halang, W.A. 61
Hanan, J. 209–10
hardware, evolvable 134–5
Hashimoto, K. 229, 231–2, 266
Hayes, J.P. 176
high-current load circuit 434
  fitness measure for 440
  test fixture for 430
  results for 454–8
high-frequency transistors 464, 466
high level circuit requirements 175
highpass filter 169
  network 147
high-return
  human-competitive machine intelligence 529
  meaning of 4
  of results produced by genetic programming 10
Higuchi, T. 135
hill climbing 59, 505–6, 511–12, 514
  search by 59–60
hits, number of 327
Ho, Y.-C. 50
Hofestadt, R. 230
Holland, J.H. 7, 507–8, 513, 515
  genetic algorithm 513
    fundamental theorem of 508
Honavar, V. 47
Hsu, F.H. 4
human-competitiveness 220, 4, 8, 111, 151, 158, 195, 299, 382, 411, 418, 481
  criteria for automatically created result 4
  meaning of 3–4
  results in genetic programming 7–10, 26, 46
hunting and instability, problems of 56–7
Hutchings, J.L. 142
Hutchison, C.A.III 229, 231–2, 226

Iba, H. 47, 135
ICONTROL 442–3, 472

IFGTZ_DEVELOPMENTAL operator 345, 359–60
IF_POSITIVE function 92
Ikeuchi, A. 422–3, 441, 460, 553
Ikeuchi-Tokuda circuit 432
improved general purpose controllers 398
  AI ratio for 412
  human-competitiveness of result for 411
  preparatory steps for 387–90
  results for 390–410
  routineness for 412
improved PID tuning rules
  AI ratio for 385
  automatic synthesis of 367
  human-competitiveness of 382–4
  preparatory steps for 374–6
  results for 376–82
Ince, D.C. 7, 513
inductor-capacitor rung 509, 512
inductor, sizing of 319
information
  contained in preparatory steps 101
  stored in population 511
initial circuit 177
initial random population of run of genetic programming 102
in-sample or training cases 330, 372
Institute of Radio Engineers (IRE) 219
integral of time-weighted absolute error (ITAE) 52, 78, 86–7, 92, 116–17, 292, 297
  calculation of 86
  component of fitness 295
  for disturbance rejection 292
  optimum transfer function 87
integrative (I-type) control 56
INTEGRATOR function 53, 56, 77
International Conference on Evolvable Systems 135
International Society for Genetic and Evolutionary Computation (ISGEC) 47
intertwined spiral problem 139–42
invention
  and evolution, illogical nature of 19–20
  process, automating 21
inverter 54, 84
Irvine, R. 332, 335, 422, 426, 469, 553
"island" version of parallel genetic programming 516
isotropic radiator 215
I-type control 103
Iwata, M. 135

Jacob, C. 47
Jakiela, M. 47
Jamshidi, M. 61
Java 42
Johnson, D.S. 176
Johnson, J.M. 206
Johnson, K.S. 324
  patented "bridged T" filter 324
Johnson, W.C. 487
Jones, H. 111, 206, 286, 290, 552
Jonoska, N. 47

Kasparov, G. 4
Katz, R. 135
  controllers 408
  PID tuning rules 377
Keane, M. A. 4, 22–3, 27, 40, 45–7, 62–3, 70, 91, 100, 104, 110, 118, 123, 127, 130, 134–5, 140, 142–3, 151, 156, 157, 161–3, 169, 174, 177, 196, 263, 274, 280, 190, 199, 302, 304, 311, 331, 342, 357, 375, 384, 395, 411, 415, 421, 450, 466, 480, 486–7, 494–6, 511, 513, 515, 523, 525–7, 539, 551
Kell, D.B. 229–30, 279
Keller, R. 47, 62
ketone bodies 230, 233
  see also metabolic pathway problem
Keymeulen, D. 135, 142, 147
Kinnear, K.E. Jr. 47
Kirkpatrick, S. 61, 61
Kitano, H. 42, 63, 132, 230
KKS (Keane-Koza-Streeter)
knock-out time series 266
knowledge-based methods 101
Kolb, B. 332, 335, 422, 426, 442, 469, 553
Korst, J. 61, 506
Koza, J.R. 4, 22–3, 25, 27, 29, 37–8, 40–3, 45–7, 62–3, 70, 91, 98, 100, 104, 110, 118, 123, 127, 130, 132, 134–5, 139–40, 142–3, 151, 156, 157, 161–3, 169, 174, 177, 196, 219, 221, 263, 274, 280, 282, 290, 299, 302, 304, 311, 331, 342, 357, 375, 384, 395, 411, 415, 421, 450, 466, 480, 486–7, 494–6, 511, 513, 515, 523–7, 531, 539, 551
Krohling, R.A. 61
Kruiskamp, M.W. 135
Kwong, S. 61

Lackawanna Ferry 16, 19, 413–14, 419
lac operon, schematic representation of genetic network, 221–2
LACTOSE 221–2
lag 54
  second-order 54, 66
Laing, S. 230
Lamont, G.B. 93, 480
Lam, T. 62
Langdon, W.B. 47–8, 481, 508

Lanza, G. 221, 274
Lanzi. P.L. 47
Laplace transform variable 74
Lee, S.G. 17, 413, 422–3, 427–8, 446, 448, 553
  circuit 449
  low voltage balun circuit 447
  patent 448, 450
  patented circuit 445–6
Leenaerts, D. 135
Leong, H.W. 176
Leung, K.S. 47
`LIMITER` function 54, 66, 75–6
Linden, D.S. 205–9, 215, 218, 484–5
Lindenmayer, A. 209–10
  rewrite rules 63
  rules for creating structures 42, 132
  systems 209–10
Linux operating system 516–17
LISP machine 209, 523–4, 527
  code 42
  programming language 35, 64, 251, 255
  symbolic expressions (S-expressions) 75, 103, 107, 131, 206, 230, 264, 342, 360, 362
    codes 363
    for crossover (woofer-tweeter) filter 498
Liu, C.L. 176
Liu, Y. 135
logic-driven search 6
Logo programming language 209
Lohn, J. 135
Loomis, W.F. 221, 229
low voltage balun circuit
  fitness measure for 436–7
  test fixture for 428
lowpass/highpass filter
  results for 344–8
  with a variable passband boundary, results for 350–7
lowpass filter 24, 169, 176, 202
  parameterized 355
lowpass filter problem
  with layout 186
    AI ratio for 197
    human-competitiveness of result for 195
    preparatory steps for 186–8
    results for 188–95
  without explicit test fixture
    AI ratio for 174
    preparatory steps for 169–73
    result for 173–4
    routineness for 174
lowpass pre-filter 87, 113
lowpass third-order elliptic filter with free variable for modular angle
  AI ratio for 324
  results for 318–23
  routineness for 324
lowpass/highpass filter circuit with free variable 342
  AI ratio for 348
  preparatory steps for 342–4
lowpass/highpass filter with variable passband boundary with a free variable 348
  AI ratio for 357
  preparatory steps for 349–50
  routineness for 357
low-voltage balun circuit 422–3, 434
  results for 444–51
low-voltage cubic 434
  function generator 422
low-voltage high-current circuit
  for testing voltage source 422
  transistor 424
Lui, W. 135

Macbeth, I. 142, 147
machine
  intelligence 1, 15
    high-return human-competitive 1, 3–15
    meaning of 6–7
  learning 3
Macready, W.G. 514
Mahfoud, S.W., 506
Mange, D. 135
Man, K.F. 62
Marcano, D. 206
Marenbach, P. 62
Massey, J. 551
mathematica 64
mathematical functions and electrical components, mixing of 83
Mathias, K. 47
Mats, N. 62
Matsuzaki, Y. 229, 231–2, 266
Maxwell's equations 205
Maziasz, R.L. 176
Mazumder, P. 135
McAdams, H.H. 221, 229
M-derived half-section filter 324
measure of merit 59
Meltzer, B. 7, 508
Mendes, P. 229–30, 279
Menon, P.K. 61
Messina, P. 515
metabolic pathway problem
  for ketone bodies, synthesis and degradation of 275–8
    AI ratio for 278
    routineness for 278
  for phospholipid cycle

AI ratio for 275
results for 267–75
metabolic pathways
alternative chemical reaction functions 279
automatic synthesis of 229
designing alternative metabolisms 279–80
future work on 278–80
minimum amount of data needed 278
null enzyme 278
metabolism of lactose 222
Metropolis algorithm 61, 506
Michaelis constant 236
Michaelis-Menten
equations 229, 236, 238
one-substrate 241
two-substrate 246–7
law 237
rate law 245
for one-substrate chemical reaction 236
Michie, D. 7, 508
Michielssen, E. 206
migration strategy 549
Miller, J.F. 47, 135
Miller, M.L. 422, 424, 427, 440, 553
parallel FET transistor structure *see* Daun-Lindberg
Mittenthal, J.E. 279
mixed analog-digital circuits 129, 434
mixed analog-digital variable capacitor circuit 422–3
fitness measure for 437–40
results for 451–4
test fixture for 429–30
Miyoshi, F. 229, 231–2, 266
monoacyl-glycerol 260, 262
Moore, G.E. 26, 479, 515, 518, 524
Moore's law 26, 28, 479, 515, 518, 521, 532
historical perspective on 523
Mooreware 28, 532
Moretti, G. 129
MOSFET transistor 427, 435
mRNA 221, 223
multi-input circuit 421
multiobjective optimization, options 93
multi-output circuit 421
multiple interacting programs 62
multiplication (MULV) 84
multiplier 54
multipolar single-stage filters 426
multi-rung ladder filter 509, 512
Muntz, E. 469
Muntzing 469
mutation 29, 32, 34, 72, 492, 503
and crossover, viewing in common framework 510
and reproduction operations 35
operation 37, 511–12, 514
in genetic programming and genetic algorithm 507
mutual coupling 207
Mydlowec, W. 110, 18, 123, 127, 143, 151, 157, 161, 165, 221, 274, 290, 331, 357

NAND function, circuit for 9, 130, 153–9, 163
problem
AI ratio for 149
human-competitiveness of result for 158
preparatory steps for 154–6
results for 157–8
routineness for 159
textbook TTL 157, 158
NASA/DoD Conference on Evolvable Hardware 135
NEC *see* Numerical Electromagnetics Code
negative feedback 51, 98, 297, 509
amplifier 17
invention of 414–15
reinvention of 413
Nelson, R.J. 552
netlist 90, 141, 238
network of chemical reactions
analog electrical circuit 231
constrained syntactic structure 251
diagram representing 238
problem of synthesis of 274
corresponding to metabolic pathway 251, 259
program tree 231
repertoire of functions in 250
representation as 250–5
reaction network 231
repertoire of terminals 251
representation
by computer programs 250–63
as analog electrical circuit 259–62
as symbolic expression 231, 255
as system of nonlinear differential equations 256
sizing of 229
SPICE netlist 231
system, nonlinear differential equations 231
topology of 229
neural networks 139
Newbold, W.F. 551
Newborn, M. 4
Newton, A.R. 65, 78, 238, 480
nicheware 531
Nichols, N.B. 367, 382
Nielsen, I.R. 130
Niwa, T. 135

n-lag plants, family of 372
Noah run 402
NODE function for connecting distant points 302–3
No Free Lunch Theorems 514
non-greedy nature of genetic methods 505–6
nonlinear mapping (NLM) 69, 335, 405, 407
non-minimal phase plant problem 125
  AI ratio for 127
  preparatory steps for 125–7
  results for 125
  routineness for 127
non-PID controller 9, 370, 518, 526
  genetically evolved 408, 509
non-trivial analog electrical circuits 486
Nordin, P. 47
npn transistor 201
number and complexity of simulations 526
Numerical Electromagnetics Code (NEC) 205
  antenna 213–14
  code 215
  simulator 205, 213–14, 220
numerical parameter values, establishing three approaches for 69–73
Nyquist criterion 136

objective function 59
O'Connor, D.G. 552
offspring
  circuit 503
  programs population of 33
Ogata, K. 50
Ohr, S. 142, 147
Olmer, M. 62
Olsson, J.R. 43
  ADATE system 43
one-argument
  ABS_SIGNAL function 66
  DELAY function 66
  DIFFERENTIATOR function 65
  FIRST-PRODUCT function 250
  GAIN function 77
  INTEGRATOR function 65
  INVERTER function 65
  LANDMARK function 210
  NOOP ("no operation") function 186
  SECOND-PRODUCT function 250
  TAKEOFF function 388
  transistor-creating NPN-TRANSISTOR-LAYOUT function 180
  TURN-RIGHT function 209
one-input, one-output initial circuit 178
  with one floating modifiable wire 333, 358
  with two modifiable wires 314
one-offspring crossover 72
one-substrate
  one-product reaction 234–43
  two-product chemical reaction 243–4
    function 253
    illustrative circuit for 244
ontogenetic programming 341
open loop systems 51
optimal or near-optimal entity 59
order-of-magnitude increase in computer power 26–8
O'Reilly, U.-M. 47
Osyczka, A. 93, 480
overfitting 327
overshoot 52–3, 99

P (proportional)
  block 369
  element of controller 74
Pacheco, M.A.C. 135
parachutists 506
parallel cluster computer systems 515
parallel implementation and computer time 515
PARALLEL-LAYOUT family of four-argument functions 183
PARALLEL_NEW function 308–9, 314, 434
  five-argument 304
parallel processing 26
parallel state-space search, massive 4–5
parameter values (tuning) for a PID-type controller 110
parent
  fitness-selected 511
  randomly selected 511
  receiving 476–7
Parmee, I. 47
parsimony 323, 389, 409, 414, 443, 469, 472–3
  element of fitness measure 398, 404, 470
  equational 310, 323
  penalty 390, 437, 443
  progressive improvement in 405
  types of 310
Parsytec PowerPC® parallel machine 523–5
  64-node 515
passband 168
  boundary 24, 312, 321, 476
  deviation 323
    of filter, maximum 321
  ripple 169
  voltage, minimum acceptable 316
passing parameters to substructures 497–9
passive lowpass filter 301
  with free variable for passband boundary 324
    AI ratio for 331
    preparatory steps for 325–7
    results for 328–31
    routineness for 331
patch antennas 206

patentability 2
patentable new inventions 2
  produced by genetic programming 22–3
patented inventions
patent law, doctrine of equivalencies of 454
Patent Office
  criteria for patent-worthiness 152, 196, 419
  requirement 449
  20th-century and 21st-century, duplicating 2
  generated by genetic programming 551–3
payoff 59
Pease, R. 135–6, 147, 425, 469
  challenge 135–6, 145
Pederson, D.O. 65, 78, 238, 480
Pentium II parallel machine 527
  1,000-node 26
Pentium parallel systems 517–18, 523–4
Perez-Uribe, A. 135
performance penalty 437, 443
permease 222
perturbable numerical values 69–70, 115, 375
petacycles, number of 518
Petaflop computing 515
Pezeshk, S. 47
Philbrick circuit 9, 130, 136, 151–3, 159
  defining characteristics of 149
  one-input, one-output initial circuit with one modifiable wire 148
  reinvention of 147–53
Philbrick circuit problem
  AI ratio for 153
  human-competitiveness of the result for 151–2
  preparatory steps for 148–50
  results for 150
  routineness for 152–3
Philbrick, G. 136, 142, 147, 552
  patent 149
phospholipid cycle 231–4, 262
  see also metabolic pathway problem
phosphotidylglycerol 242
"PI" (proportional-integrative) controller 57
PID (proportional, integrative, and derivative) block 404
PID compensator 87, 109, 113
PID controller 56–8, 77–8, 85, 111, 117, 400, 509
  rediscovery by genetic programming 299
  with setpoint weighting 289
  tuning of 483–4
  tuning rules 50
    genetically evolved 377, 384–5
    patentability 384
    quality of 367
PID-D2 controller 111, 286, 291
PID sub-controller 509
PID technology, conventional 87
plant 50
  representing 67–8
  response 50, 52, 54
  test bed of 371–4
  two-lag 87–113
  ultimate gain 375, 391
`PLANT_OUTPUT` terminal 67, 106
pnp transistor 201
`PNP-TRANSISTOR-LAYOUT` function 180
point-labeled tree 74
poison pill 509, 512
pole
  cancellation 92
  elimination 92
Poli, R. 47, 481, 508
population
  of randomly generated computer programs 31
  size 390
population-based methods employing recombination 514
post-2000 patent circuits, six, 422
  AI ratio for 482
  automated re-invention of 421
  computer time consumed 478
  human-competitiveness of results for 481
  preparatory steps for 428–44
  problems
    inputs, outputs, and power sources for 434
    uniformity of treatment of six problems 426–8
  routineness for 481
Potter, M.A. 47
`POW` function 388
Preatt, L. 62
preparatory steps
  five major, for basic version of genetic programming 10–11
  knowledge incorporated into 96
printed circuit board 177
prior art balun circuit 446, 450
problem
  characteristics
    suggesting use of genetic algorithm 483–4
    suggesting use of genetic programming 484–514
  computer time consumed 519–21
involving low pass filter 203
  suitability for genetic programming 483
problem-independent executional steps of run of genetic programming 10
problem-solving method 4
problems
  40/60 dB amplifier 364–6
  active lowpass filter with free variable for passband boundary 332–9

problems *contd.*
  arithmetic logic unit (ALU) circuit 159–62
  controller for two families of plants 291–300
  cubic function generator 461–6
  high-current load circuit 454–8
  improved PID tuning rules 374–85
  layout of amplifier 197–203
  lowpass filter
    layout of 186–97
    without explicit test fixture 169–74
  lowpass/highpass filter
    circuit with free variable 342–8
    with variable passband boundary with a free variable 348–57
  low-voltage balun circuit 422–3, 434
  metabolic pathway
    for ketone bodies 275–8
    for phospholipid cycle 267–75
  mixed analog-digital variable capacitor circuit 451–4
  `NAND` circuit 153–9
  parameterized controllers (with two free variables) 387–412
  passive lowpass filter with free variable for passband boundary 324–32
  Philbrick circuit problem 148–53
  quadratic/cubic computational circuit with free variable 358–64
  quadratic polynomial 34–8
  RC circuit with gain greater than two 142–7
  square root circuit 162–8
  third-order elliptic lowpass filter with a free variable for the modular angle 312–24
  three-lag plant 113–20
    with a five-second delay 120–4
    with free variable, parameterized controller for 282–92
  tunable integrated active filter 466–78
  two-lag plant 88–113
  voltage-current conversion circuit 431–58
  Yagi-Uda antenna 216–19
programmable microprocessors 381
program architecture and architecture-altering operations 40
program trees 35–6, 62–4, 131, 206–7, 230, 263
  converted to block diagram 63
  in initial random generation 68
  representation
    improved 278
    of PID controller 75
  types of functions in 89
promiseware 531
Prusinkiewicz, P. 209–10
Ptashne, M. 221
P-type control 56, 103

Q transistor-creating function, new 305–6
  six-argument 306
quadratic equation 504
quadratic polynomial 34–8
quadratic/cubic computational circuit 358
  preparatory steps for 358–60
  results for 360–3
Quarles, T. 65, 78, 238, 480

Rahmat-Samii, Y. 206
rate laws 236
RC circuit 159
  with gain greater than two
    AI ratio for 145–7
    results for 142–3
reaction function 244
reactor 264
recombination
  (crossover) operation 505
  with population in genetic methods 506–14
reference
  controlled plant 289
  controller 289
  plant 289
  signal 50, 54
`REFERENCE_SIGNAL` terminal 67, 106
reflectors 208
regulatory genes 221
regulatory proteins 221
`REPRESSOR` gene 221–2
reproduction operation 29, 32, 34, 37, 72, 492
Riccati equations, algebraic 58
Rice, J.P. 40, 43, 46, 62, 282
Riolo, R.L. 47
rise time 52
RLC circuit 332
  model 476–7
Robertson, G. 515
root loci 58
Ross, J. 229–30
Roughgarden, J. 62
"routine", meaning of 5–6
routineness of transition in problems
  circuit synthesis without layout to circuit synthesis with layout 196
  controller synthesis to circuit synthesis 143
  non-parameterized circuit to parameterized circuit 311
  non-parameterized controller to parameterized controller 290–1
  parameterized topologies for controllers to PID tuning rules 385
  parameterized topology without conditional developmental operators to problem with conditional developmental operators 34

synthesizing controllers, circuits, circuits with layout, antennas, and genetic networks to synthesis of network of chemical reactions 274
synthesizing controllers, circuits, and circuit layout to synthesizing an antenna 219–20
synthesizing controllers, circuits, circuits with layout, and antennas to genetic network synthesis 226–8
Roy, R. 47
RTL (Resistor-Transistor Logic) 157
Rudnick, E.M. 135
Rudolph, G.47
Rutenbar, R.A. 176
Ryan, C. 47

Saito, K. 229, 231–2, 266
Salamon, P. 61, 506
Salmon, J. 515
Samuel A. 3, 6, 411
criterion for artificial intelligence and machine learning 112, 159, 196, 300, 384, 412, 419, 481
Sanchez, E. 135
Sangiovanni-Vincentelli, A. 65, 78, 480
Savarese, D.F. 515
sawtooth time series 265
Sbarbaro, D. 125–6
scalable automatic programming 43
Schaeffer, J. 5
Scheeline, A. 279
schemata, changing mix of 507
Schema Theorem 513
Schoenauer, M. 47
Schultz, A.C. 47
search techniques 59
Sechen, C. 176
`SECOND-PRODUCT` function 252, 265
selector function 251
semantics-preserving architecture-altering operations 521
semi-isolated subpopulations (demes) of individuals 517
Sen, S. 47
sensor noise 53, 292
attenuation 373
serial computer 25
series composition 495
`SERIES-LAYOUT` function 182
setpoint 50, 67
weighting 287, 369–70
settling time 52
S-expressions *see* LISP
Shapiro, L. 221, 229
Sheingold, D.H. 425
Shen, P. 229–30
Shimizu, T.S. 229, 231–2, 266
Shipman, J. 515
Sibani, P. 61, 506
signal-processing
blocks 74–5, 69, 77, 389
functions 53, 66, 98
silicon wafer 177
simulated annealing 59–60, 511–12, 514
and genetic methods 506
search by 61
single gene value 507–8
single numerical value 99
of fitness 132
single parental program tree 510
single perturbable numerical value 198
Sipper, M. 135
sizing *229*
Smith, P.H. 515
Smith, R.E. 47
Smith, S.F. 485, 495
genetic algorithm operating on variable-length strings 486
soccer-playing program 8
soloware 531
solution, prespecified 485
Somogyi, R. 230
source resistor 190
RSRC 191
specificity order 507
Spector, L. 42, 47, 63, 132
SPICE (Simulation Program with Integrated Circuit Emphasis) 65, 78, 160, 165, 167, 239, 247, 309, 315, 335, 343, 431, 443
commands 91
DC sweep 199
input file for PID controller, a two-lag plant, and an in-line calculation of ITAE 79–81
mathematical functions in 83
netlist 64–5, 131, 265
simulation 86, 94, 290, 299, 376
simulator 65, 78, 91, 93, 99, 100, 107, 141, 146, 156, 167, 170, 187, 205, 213, 220, 238, 265–6, 284–5, 294, 295, 373, 375–6, 389, 391
subcircuit, definition in 83
SPICE3 simulator 480
square root circuit
AI ratio for 168
preparatory steps for 162
results for 163
routineness for 168
square root computational circuit 130
squaring circuit 360–1

Sripramong, T. 480
standard block diagram manipulations 109, 117
state-space search, massive parallel 4–5
Steinberg, M.L. 62
Steinmetz, C. P. 18
Stender, J. 515
Stephens, C.R. 508
Sterling, T.L. 515
Sternberg, P.W. 221, 229
Steudel, G.W. 551
Stevenson, A. 56–7, 111, 300, 367, 509, 552
Stiffelman, O. 110, 118, 127, 143, 151, 161, 165, 280, 515
Stoffel, K. 42, 63, 132
Stoica, A. 135, 142, 147
stopband 169, 325
  attenuation 321
  boundary 312, 316
  ripple 169, 325, 344
Streeter, M.J. 22–3, 27, 375, 384, 395, 411, 466, 526–7
strong typing 39
Stutzman, W.L. 205
subcircuit definition for derivative (`DIFFB`) 85
sub-string producing T-section 495
substructures
  hierarchical references among 496
  number of 495–6
    arguments possessed by 500
  reuse of 486–95
  type of 499–500
subtractor 54
subtree 37
  implementing negative feedback 512
sum-integrator 238, 240, 246, 249
Svensson, P. 62
Sweriduk, G.D. 62
symbolic expression 34, 64, 131
symbolic regression 34–5
symmetric antenna 213
symmetry-breaking procedure using geometric coordinates 303–4
system
  bandwidth 106
  identification 34–5

T.J. Watson Research Center 4
Tackett, W.A. 41
Takahashi, K. 229, 231–2, 266
`TAKEOFF` functions 389
takeoff point reference terminal 388
Takeshian, A.A. 424–5, 427, 442, 553
Tanaka, K. 135
Tanaka, T. 135
Tanese, R. 515
Tang, K.S. 61
Tanida, S. 229, 231–2, 266
Taylor, F.J. 312–13, 320, 487
Teller, A. 62
Temes, G.C. 425
terminal `HALF-MM-WIRE` 210
terminal set 71
  identification 11
terminals, repertoire of 67
termination criterion 31, 35
  and results designation 95
Terrell, W.H. 157, 552
test fixtures 132, 139, 177, 342, 428
Tettamanzi, A.G.B. 47
Texas Instruments LISP machine *see* LISP
Thiele, G.A. 205
Thiele, L. 93, 480
third-order elliptic filter 319, 323
third-order elliptic lowpass filter 301
  with free variable for modular angle 312–24
Thompson, A. 135, 142
Thomson, P. 135
three-argument
  `ADDITION` function 77
  addition subcircuit 84–5
  `IFGTZ_DEVELOPMENTAL` operator 342
  IF operators 224
  `IF_POSITIVE` function 66
  `LAG2` 66
  `LIMITER` function 66, 77
  `LIMIT` subcircuit 86
  `THREE_GROUND` function 171, 492–3
three-bit digital register 429
three-lag blocks 391
three-lag plant 113–20, 281, 372
three-lag plant problem
  AI ratio for 119–20
  preparatory steps for the 114–15
  results for 115–18
  routineness for 119
three-lag plant with five-second delay 120
  AI ratio for 123–4
  preparatory steps for 120–2
  results for 122–3
  routineness for 123
three-lag plant with free variable, parameterized controller for 282–92
  AI ratio 291–2
  preparatory steps 283–6
  results for 286–90
three-ported automatically defined function 497–8
three-ported subcircuits 480
  produced by automatically defined function 496

three-ported voltage divider subcircuits 480
three-way decomposition 5
time constant 391
time-derivative of deviation 57
time-domain
  analysis 421
  functions 65
  signals 55
  simulations 525
  two parameters 368
time-integral of deviation 57
time-steps 59
time-weighted absolute error *see* integral of time-weighted absolute error (ITAE)
Tinsley, M. 5
Tokuda, N. 422–3, 441, 460, 553
Tomassini, M. 47, 135
Tomita, M. 229, 231–2, 266
topologies, parameterized 2, 67
  automatic creation by genetic programming 23–5
  for circuits
    automatic synthesis of 301
    five new techniques 301–6
  with conditional developmental operators for circuits, automatic synthesis of 341
  containing free variables 504
  for controllers
    automatic synthesis of 281
    for multiple plants 291
    improved, with free variable, automatic synthesis of 387
topology
  of controller 55
  of PID controller with nonzero setpoint weighting of reference signal 369
  for PID-D2 9
  and sizing of networks of chemical reactions, automatic synthesis of 230–1
topology-modifying `FLIP` function 140
topology-modifying functions 178, 181–6, 305
  for series 198
topology-modifying `SERIES` function 140
total harmonic distortion (THD) 414–15, 436–7, 445
Toumazou, C. 480
tournament selection 37
"toy" (proof of principle) problems 43
training cases 139
transfer function 64, 109
  of two-lag plant 87
transfer of information 507
transient phenomena 324
transistor-creating functions 303
transistor models, (*nmos*) and (*pmos*) 435
Transtech® transputer machine 525
  64-node 523
trees, functions and terminals in 73
T-sections 487, 489, 491, 501
  cascade of identical 487
  circuit
    consisting of single, 487, 491
    consisting of cascade of six identical 489
    consisting of cascade of two identical 488
T-shaped subcircuit 487
TTL (Transistor-Transistor Logic) 157
tunable integrated 434
  active filter 422, 426
    fitness measure for 442–4
    patented circuit for 467
    results 466–78
    test fixture for 432
tuning 58
  rules for parameterized PID controller, automatic synthesis of 281
Turing, A. 1, 6–7, 508, 513
turtle-controlling functions and terminals 211
two-argument
  addition 84
    subtraction, multiplication, and division functions 66
  address 391
  `ADD_SIGNAL`, `SUB_SIGNAL`, and `MULT_SIGNAL` functions 65
  capacitor-creating `LAYOUT-C` function 179
  `DIFFERENTIAL_INPUT_INTEGRATOR` function 65
  `DRAW` function 209
  `GAIN` function 65
  inductor-creating `L-LAYOUT` function 188
  `INPUT_0` function 170
  `LAG` function 66, 77
  `LEAD` function 66
  `NODE` function 303
  `OUTPUT_0` function 170
  `POW` function 375
  `REPEAT` function 209
  `SERIES-LAYOUT` function 181
  `SUBTRACT` function 77
  `TRANSLATE-RIGHT` function 210
  `TWO_GROUND` function 171
two-dimensional 6×11 toroidal mesh 95
two-input, one-output initial circuit 155
two-lag plant problem 96
  AI ratio for 112–13
  choice of control parameters for 100
  fitness measure for 90
  genetically evolved solution to 109
  high-level requirements for 90
  human-competitiveness of the result for 111–12

two-lag plant problem *contd.*
preparatory steps for 88
results for 102–10
terminal set in 97
transition to three-lag plant problem 119
TWO_LEAD function 171, 433
for inserting two-leaded components, new 305
two-offspring crossover operation 36
two-substrate
one-product chemical reaction function 252
one-product reaction 244–8
two-product reaction 248–50

U.S. Patent Office 21, 300
U.S. patents 417, 426, 442, 446–7, 451–2, 456–8, 460–2, 464, 469–70, 474, 478
Uda, S. 219
ULIMIT function 388
Ullman, J.D. 176
ultimate gain 292, 368
ultimate period 292, 368, 391
unbalanced coaxial video source 429
United States Code 21–2, 111, 152, 384, 449
unity-gain follower 136

vacuum tube transistor 414
vagueware 531
value-setting subtrees 70–2, 172, 198
Van Veldhuizen, D.A. 93, 480
variable capacitor circuit 434
variable-length chromosomes 485
Vecchi, M.P. 61, 506
Vellasco, M.M.B.R. 135
Veloso, M. 62
Venter, J. C. 229, 231–2, 266
VIA and PAIR_CONNECT functions 302–3
vias 176, 184
VIA-TO-GROUND-LAYOUT family of three-argument functions 184
Villagran and Sbarbaro controller 125
Villagran, V. 125–6
Vladimirescu, A. 414, 436
Voigt, H.-M. 47
Voit, E.O. 221, 229
voltage
adders 238, 244, 246, 249
divider subcircuits 486
gain stages 480, 486
source 238, 246, 249
voltage-controlled current source (VCCS) 84
voltage-current 434
conversion circuit 422–3
fitness measure for 441–2
results for 458
test fixture for 431
V-shaped element 216
VSWR 214–15

Waelbroeck, H. 508
Wegener, J. 47
weighted reference signal 369
Wen, X. 230
Westinghouse's "nearly universal idiom" 17
Whitley, D. 47, 62, 514
Widlar, R. J. 551
Williams, A.B. 312–13, 320, 487
wire antennas 206
design of 206, 484
method-of-moments (MoM) simulator for 205
Wolpert, D.H. 514
Wong, D.F. 176
Wong, M.L. 47
woofer speaker of hi-fi system 23
Workshop of Genetic Programming Theory and Practice 47
Wu, A. 47

Yagi, H. 219
Yagi-Uda antenna 207–8, 220, 485
results 216–19
topology for 219
Yasunaga, M. 135
yield, return measured in terms of A-to-I ratio 5
Yu, J. 110, 118, 123, 127, 143, 151, 157, 161, 165, 219, 274, 290, 299, 331, 357
Yugi, K. 229, 231–2, 266
Yuh, C.-H. 221, 229
Yousefpor, M. 62

Zebulum, R.S. 135, 142, 147
zero-argument
automatically defined functions 67
END function 141, 186
SAFE_CUT function 141
Ziegler, J.G. 367, 382
Ziegler-Nichols
PID tuning rules 367
tuning rules 9, 22, 27, 368–9, 382, 384–5, 411, 526
Zitzler, E. 93, 480
Zobel, O.J. 307, 323, 552
"*M*-derived half section" and "constant *K*" filter sections 8
Zobel network problem involving two free variables 301–2, 306–7, 311, 323
AI ratio 312
preparatory steps for 307–10
results 310–11
simple 518

Disclaimer