

An Optimization Approach to Estimating Stability Regions Using Genetic Algorithms

Benjamin P. Loop, *Student Member, IEEE*, Scott D. Sudhoff, *Senior Member, IEEE*,
Stanislaw H. Żak, and Edwin L. Zivi, *Member, IEEE*,

Abstract—The problem of estimating regions of asymptotic stability for nonlinear dynamic systems is considered as an optimization problem. Genetic algorithms are then proposed to solve the resulting optimization problems. Three test systems are used to evaluate the performance of the proposed genetic algorithms. The test systems are 6th, 8th, and 17th order nonlinear power electronics systems. The performance of the genetic algorithms are also compared with that of the classical Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm and the simplex method of Nelder and Mead. Time domain simulations of the test systems are performed to validate the results of the optimization algorithms. Issues involved with the successful implementation of genetic algorithms to estimate regions of attraction are discussed. It is observed that genetic algorithms outperform the classical optimization algorithms in estimating regions of asymptotic stability.

Index Terms—Genetic algorithms, Stability, Lyapunov methods, Optimization methods, Power electronics systems, quasi-Newton optimization method, Nelder-Mead simplex algorithm.

I. INTRODUCTION

An asymptotically stable equilibrium point has the property that the state of the system will return to the equilibrium point following small disturbances. It is then natural to ask how far the state can be perturbed and still return to the equilibrium point. This question drives the search for regions of asymptotic stability. A region of asymptotic stability (RAS) is a region for which all system trajectories starting within the region asymptotically return to the equilibrium point.

Lyapunov theory gives conditions that ensure that a region bounded by a Lyapunov function contour is an RAS. The problem of finding the largest Lyapunov function contour that bounds an RAS is considered in this paper. In [1], the search for such an RAS is formulated as an optimization problem. Conventional optimization methods are not suitable for such problems because good initial estimates are not readily available, objective function gradients can be difficult or impossible to calculate, and computation times for large-scale systems become prohibitive.

This work was supported in part by the Office of Naval Research under Grant N00014-02-1-0990 “Polytopic Model Based Stability Analysis and Genetic Design of Electric Warship Power Systems” and by the National Science Foundation under Grant 9972752-DGE, “IGERT: Variable Speed Electromechanical Drive Systems.”

B. P. Loop, S. D. Sudhoff, and S. H. Żak are with Purdue University, West Lafayette, IN 47907 USA (phone: 765-494-3487; fax: 765-494-0676; e-mail: loop@ieee.org).

E. L. Zivi is with the US Naval Academy, Annapolis, MD 21402, USA.

In this paper, genetic algorithm (GA) based techniques are used to estimate RASs for 6th, 8th, and 17th order power electronics system models. In addition to this, the effectiveness of alternative optimization approaches to estimating RASs are compared to the GA approach. Because convergence to the global optimum is not guaranteed, the result of the optimization procedure yields only an upper bound on the Lyapunov contour that bounds an RAS. It is, therefore, prudent to validate the RAS estimates. Two methods are proposed to accomplish this. Positive results of the validation procedures increase the system analyst’s confidence in the RAS estimates.

II. BACKGROUND RESULTS FROM STABILITY THEORY

From Lyapunov theory it is known that a region

$$D = \{x : V(x) \leq c\} \quad (1)$$

is an RAS if $\dot{V}(x)$, the time derivative of the Lyapunov function $V(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ along trajectories of the dynamic system in question, is negative in $D \setminus \{0\}$. The application of this theory to constrained system models is discussed next. Then, an optimization approach to finding the largest Lyapunov function contour value that satisfies the condition is set forth.

A. Constrained System Models

Typically, system models are only valid in a certain restricted region of the state space. These types of models are called constrained system models [2]. The feasible region, which is the region of model validity, can often be represented in terms of algebraic constraints of the form

$$g(x) < 0, \quad (2)$$

where $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ and $<$ implies that all components of the vector valued function $g(x)$ are less than zero. The constraint could be due to a change in structure of the system beyond the boundaries or natural restrictions on the operation of a physical system. The case study in Section V is an example of such a system.

In constrained systems some of the system trajectories may leave the feasible region at a certain point in time, then return to the region and asymptotically approach the equilibrium point. Because it is undesirable for trajectories of the system to leave the feasible operating region, these trajectories must be excluded from any RAS estimate. A restricted region of asymptotic stability (RRAS) can, therefore, be defined as a connected set of points in state

space for which a trajectory starting at a point in the set approaches the equilibrium point asymptotically and does not leave the feasible operating range. The region defined in (1) is an RRAS if $\dot{V}(x) < 0$ and $g(x) < 0$ in $D \setminus \{0\}$.

B. Optimization Approach to RAS Estimation

The problem of finding an RAS can be formulated as the optimization problem

$$\begin{aligned} & \text{maximize } c \\ & \text{subject to } \dot{V}(x) < 0 \text{ when } V(x) \leq c. \end{aligned} \quad (3)$$

However, the above optimization problem is not tractable by available techniques because the constraint applies to an entire domain, which in turn implies the existence of an optimization problem within an optimization problem. Indeed, the constraint of (3) involves solving an optimization sub-problem of the form

$$\begin{aligned} & \text{maximize } \dot{V}(x) \\ & \text{subject to } V(x) \leq c. \end{aligned} \quad (4)$$

The optimization problem (3) can be recast in a more suitable form for numerical computation [1]:

$$\begin{aligned} & \text{minimize } V(x) \\ & \text{subject to } \dot{V}(x) \geq 0. \end{aligned} \quad (5)$$

While optimization problem (3) tries to enlarge the RAS by searching ‘good’ regions of state space (where $\dot{V}(x) < 0$), optimization problem (5) seeks the smallest Lyapunov function value in the ‘bad’ regions of state space (where $\dot{V}(x) \geq 0$). The solution to (5) gives an upper bound on the values of $V(x)$ that yield valid RASs. Note that the origin, $x = 0$, is the global optimizer of this problem since $V(x) > 0$ when $x \neq 0$ and $\dot{V}(0) = 0$. This solution corresponds to a trivial RAS, but a non-trivial RAS is guaranteed since it was assumed the equilibrium point was asymptotically stable. It is, therefore, necessary to prohibit the optimization procedure from converging to the origin.

In the case of constrained systems, it is possible that a subset of the RAS predicted above includes points that do not lie in the feasible operating region. To remedy this situation, additional constraints are placed on the optimization problem. The equivalent of (3) for constrained systems is

$$\begin{aligned} & \text{maximize } c \\ & \text{subject to } \dot{V}(x) < 0 \text{ and } g(x) < 0 \text{ when } V(x) \leq c. \end{aligned} \quad (6)$$

As before, this optimization problem enforces a restriction on an entire region. A modification of (5) can alleviate this problem,

$$\begin{aligned} & \text{minimize } V(x) \\ & \text{subject to } \dot{V}(x) \geq 0 \text{ or } g_1(x) \geq 0 \text{ or } \dots \text{ or } g_m(x) \geq 0 \end{aligned} \quad (7)$$

As with (3) and (5), optimization problems (6) and (7) imply searching through ‘good’ and ‘bad’ space, respectively.

III. GENETIC ALGORITHM APPROACH TO STABILITY REGION ESTIMATES

The GA used for this problem is the real-encoded GA described in [3], [4]. This GA has also been used in [5]–[8]. What remains is to specify the fitness function.

GAs are formulated to maximize a unconstrained fitness function. An appropriate fitness function that translates the optimization problems (5) and (7) into unconstrained maximization problems is given by

$$f_{fit}(x) = -V(x) - p(x), \quad (8)$$

where $p(x)$ is an additive penalty function.

The additive penalty function $p(x)$ is used to enforce the constraints of optimization problems (5) and (7). It takes the value zero when the constraints are met and a large positive number when the constraints are not met. For an unconstrained system,

$$p(x) = \begin{cases} 0 & \text{if } \dot{V}(x) \geq 0 \\ M & \text{otherwise} \end{cases} \quad (9)$$

and for a constrained system,

$$p(x) = \begin{cases} 0 & \text{if } x \neq 0 \text{ or } \dot{V}(x) \geq 0 \text{ or } g_1(x) \geq 0 \text{ or } \dots \\ & \text{or } g_m(x) \geq 0 \\ M & \text{otherwise,} \end{cases} \quad (10)$$

where M is a large positive number, which should be chosen larger than values of $V(x)$ in the search region. Note that since the equilibrium point was assumed stable, $\dot{V}(x)$ is negative definite in some deleted neighborhood of the equilibrium point. Thus, $p(x)$ takes the value M at all points in the deleted neighborhood of the origin, and this fitness function prevents convergence to the origin.

Note that the best solution found by the GA comes from the ‘bad’ region of state space. The value of the fitness function evaluated at this solution is an upper bound of Lyapunov function values for which regions of the form (1) are (R)RASs. To determine a (R)RAS, the value must be scaled by, for example, 99%. Methods for evaluating whether a particular Lyapunov function value corresponds to a (R)RAS are given in the next section. If a negative result occurs with either of the verification procedures, then the value returned by the GA must be scaled down further (for instance, 98% of the original value) until a negative result does not occur. An alternate procedure for scaling the Lyapunov function contour so that it contains a (R)RAS is set forth in [4].

IV. SECONDARY CONFIDENCE CHECKS

As stated above, optimization problems (5) and (7) search through ‘bad’ regions of state space in an attempt to minimize the Lyapunov function. Because the solution found in this manner comes from the ‘bad’ region, it is prudent to scale down the best solution found by the GA and verify that it corresponds to a valid (R)RAS. Two methods are

proposed to check this. The first is a GA that checks the constraints in (3) and (6). The second approach uses time-domain simulations.

A. Genetic Algorithm Confidence Check

For unconstrained systems the GA verification is straightforward. It can be accomplished by solving the optimization problem,

$$\begin{aligned} & \text{maximize} && \dot{V}(x) \\ & \text{subject to} && V(x) = c, \end{aligned} \quad (11)$$

where c is the Lyapunov function value under consideration. The genetic algorithm described above can be used to maximize the fitness function $\dot{V}(x)$. If the fitness function value of the best solution found is non-negative, then the region corresponding to the Lyapunov function value c is not a valid RAS. The constraint $V(x) = c$ can be forced by scaling each individual in the population using

$$x^{(i)} = \sqrt{c/c_i} y^{(i)}, \quad (12)$$

where $y^{(i)}$ is an individual in the GA population, c is the Lyapunov function contour value under consideration, c_i is the Lyapunov function value at $y^{(i)}$, and $x^{(i)}$ is the point on the desired contour where the fitness function is evaluated.

Similar verification GAs can be formulated for constrained systems. In this case, $m + 1$ optimizations must be carried out, where m is the number of constraints bounding the feasible region. In addition to optimization problem (11), optimization problems maximizing the fitness functions $g_1(x), \dots, g_m(x)$ under the constraint $V(x) = c$ must be solved. The predicted region is not an RAS if any of the optimization procedures yields a non-negative fitness function value for the best individual identified by the GA.

B. Confidence Check Using Time Domain Simulation

When it is feasible to run a large number of time-domain simulations of the system under investigation, it is prudent to validate the estimated RAS by this method also. If a trajectory is found that begins inside the proposed RAS and does not approach the equilibrium point, then the region by definition is not a valid RAS. Of course, not finding a trajectory that invalidates the RAS does not validate the RAS. A positive result of this process does, however, increase the confidence of the system analyst in the predicted RAS. For studies herein, initial conditions for the simulations were generated randomly with uniform probability over a predefined area of the state space.

V. TEST SYSTEMS

The systems used to evaluate the performance of the proposed procedure are subsystems of the Naval Combat Survivability DC Distribution Testbed [9]. The subsystems are composed of three components: a controlled rectifier power supply (depicted in Fig. 1), a dc/dc converter module (depicted in Fig. 2), and an inverter module feeding a resistive load (depicted in Fig. 3). The power supply is designed to supply 500 V dc from a 560 V line-to-line

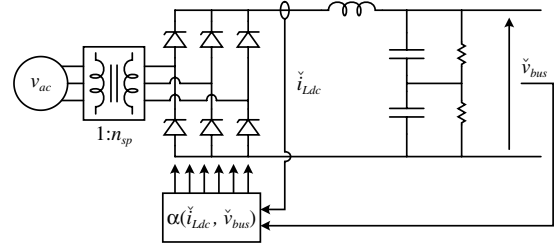


Fig. 1. Rectifier power supply.

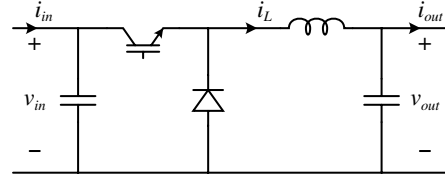


Fig. 2. dc/dc converter module.

rms three-phase ac input. The dc/dc converter module steps a 500 V input down to 420 V and provides isolation to the loads it feeds from the input. The inverter module provides three-phase ac power from its dc input, and is designed to maintain a constant power at its output. For details on the controls of these components, see [9], [10]. In addition these systems are shut down in certain circumstances to safeguard system components. In particular, constraints are placed on bus voltages to prevent them from reaching dangerous levels. This gives rise to a constrained system model. The three systems of interest are:

- 1) Power supply feeding the inverter module load (6 states)
- 2) Power supply feeding the inverter module load with an inductive line impedance represented (8 states)
- 3) Power supply feeding the inverter load through the dc/dc converter module (17 states)

The switching of semiconductor devices in power electronics based systems complicates the stability analysis of these systems, because system quantities are not necessarily constant in the steady state. To enable the use of well-known stability theory of fixed equilibrium points, average value models of these time varying systems are often used [11]–[15].

To illustrate the models of the test systems, the six-order state-space model of the power supply feeding the inverter module load is set forth. First, it should be noted that if controlled appropriately, the inverter module can provide constant power at its output. Therefore, on an average value basis, it behaves as a constant power load. Also, the average value of the voltage out of the rectifier is a function of the firing angle α . Fig. 4 depicts an equivalent circuit for this system.

The first stage of the controller filters measurements of the inductor current, i_L , and bus voltage, v_1 . The filtered

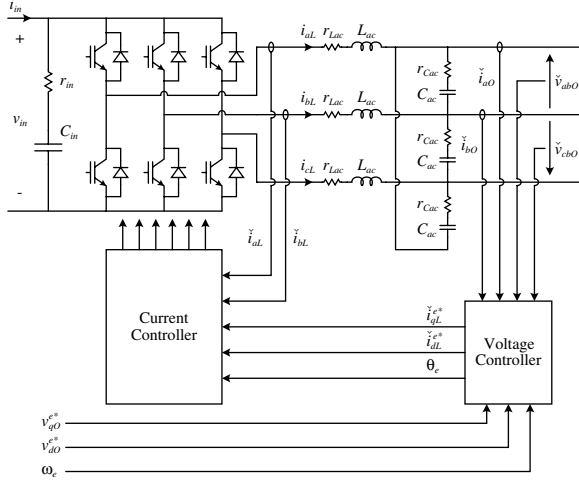


Fig. 3. Inverter module.

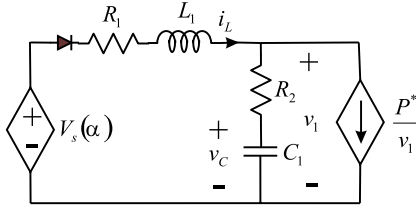


Fig. 4. Average value model of sixth order system.

values are denoted with a '^', and their derivatives are

$$d\hat{i}_L/dt = (i_L - \hat{i}_L)/\tau_{i_L} \quad (13)$$

and

$$\dot{\hat{v}}_1 = (v_1 - \hat{v}_1)/\tau_{v_1} \quad (14)$$

The bus voltage can be calculated in terms of inductor current and capacitor voltage, v_C , as

$$v_1 = (R_2 i_L + v_C + \sqrt{(R_2 i_L + v_C)^2 - 4PR_2})/2,$$

where R_2 is the effective series resistance of the capacitor and P is the constant power out of the inverter module.

The next stage of the controller generates the bus voltage command, v_1^* . The derivative of this quantity is

$$\dot{v}_1^* = \text{bnd} \left(V_1^* \text{bnd} \left(\frac{I_{sc} - \hat{i}_L}{I_{sc} - I_{thr}}, 0, 1 \right) - v_1^*, \dot{v}_{1,min}^*, \dot{v}_{1,max}^* \right), \quad (15)$$

where

$$\text{bnd}(x, y, z) = \begin{cases} y & x \leq y \\ x & y < x \leq z \\ z & z < x, \end{cases}$$

V_1^* is the nominal output voltage command, I_{sc} is the maximum allowable current out of the power supply, I_{thr} is the threshold current above which the bus voltage command

will start to be decreased, and $\dot{v}_{1,min}^*$ and $\dot{v}_{1,max}^*$ are the minimum and maximum allowable slew rates for the bus voltage command.

The commanded rectifier output voltage is

$$v_r^* = v_1^* + k_v(v_1^* - \hat{v}_1) + v_{ierror} - k_{ip}\hat{i}_L,$$

where k_v and k_{ip} are control parameters and v_{ierror} is a state variable whose derivative is

$$\dot{v}_{ierror} = k_v(v_1^* - \hat{v}_1)/\tau_v. \quad (16)$$

This component of the controller implements a proportional plus integral voltage control and a proportional current feedback control. The thyristor firing angle is then calculated as

$$\alpha = \arccos(\text{bnd}(v_r^*/V_r, -0.9, 1)),$$

where V_r is the average voltage out of rectifier if there is no phase delay. The average voltage out of the rectifier is

$$v_s = V_r \cos \alpha - 2v_d,$$

where v_d is the voltage drop across a conducting thyristor.

The derivatives of the last two state variables are

$$di_L/dt = (v_s - R_1 i_L - v_1)/L_1 \quad (17)$$

and

$$\dot{v}_C = (v_1 - v_C)/(R_2 C_1), \quad (18)$$

where R_1 is the series resistance of the dc link inductor, L_1 is the dc link inductance, and C_1 is the effective capacitance of the power supply's output capacitance and the inverter module's input capacitance in parallel.

The state-space models of the remaining test systems can be found in [4].

VI. RESULTS OF STABILITY ANALYSIS

The goals of this work are twofold: investigate the use of genetic algorithms in the estimation of regions of asymptotic stability and compare their effectiveness with other methods. To this end the region of asymptotic stability estimation method described above was applied to all three systems. Also, two other optimization methods were investigated. The first is a quasi-Newton method using the BFGS (Broyden, Fletcher, Goldfarb, and Shanno) [16] Hessian matrix approximation update; the second is the Nelder-Mead simplex method [17], [18]. Both of these methods are available in the MATLAB Optimization Toolbox [19]. Finally, the confidence checks proposed above are applied to provide an indication of the accuracy of the approach.

A. Optimization Method Comparison

The optimal population size and number of generations to use in a genetic algorithm is unclear a priori. A number of settings of these parameters was tested for each of the systems. For each setting the GA was run N times, where $N = 200$ for the first two test systems and $N = 100$ for the third test system. The average computation time per

run for each setting was also recorded. The two alternative optimization procedures both require initial conditions; however, good initial conditions are not known. The initial conditions were generated randomly with uniform probability over the same feasible region as was used in the GA. To make the comparison meaningful, it is important to use equal computation time. The two alternative optimization procedures were also run N times corresponding to each population size/generations setting of the GA. Random initial conditions were generated and the procedures run starting at those conditions. Then, this would repeat until the appropriate amount of computation time had elapsed.

System 1 Results: Fig. 5 depicts the results of the optimization procedures. The data in the legend in Fig. 5(a) is in the form: run time; population size; number of generations. The optimization procedures were run 200 times, that is, $N = 200$, for each setting of population size and number of generations. Then, the results of the 200 runs were sorted in ascending order and plotted. The vertical axis of these plots is the best (smallest) Lyapunov function value found by the optimization procedures. It is desirable for the data series to be lower on the plot and remain low over all of the 200 runs; however, since all of the procedures have a random nature, they will not always converge to the same solution. For this system, all methods found the same minimum; however, the quasi-Newton and Nelder-Mead methods found the minimum more consistently than the GA.

System 2 Results: Fig. 6 depicts the results of the optimization procedures. As with System 1, the optimization procedures were run 200 times ($N = 200$) for each setting of population size and number of generations. In this case, the GA has about the same performance for lower computation times; however, at higher computation times, the GA is able to find the minimum more frequently. As above, all of the methods find roughly the same minimum value.

System 3 Results: Fig. 7 depicts the results of the optimization procedures. The optimization procedures were run 100 times ($N = 100$) for each setting of population size and number of generations. For this system, the quasi-Newton and Nelder-Mead methods could not find the same minimum value as the GA. The minimum values found by the GA, quasi-Newton method and Nelder-Mead method are 845, 1507, and 1143, respectively. For higher computation times, the GA is able to find the minimum value roughly 30% of the time; It is apparent, however, that this problem is getting harder to solve as the system order increases because a significant portion of optimization runs yielded results that were not close to the minimum value.

B. Confidence Check Results

As stated above, the solutions obtained by the optimization procedures are upper bounds. A genetic algorithm was used to solve (11) for all three systems starting at 99% of the minimum Lyapunov function values obtained

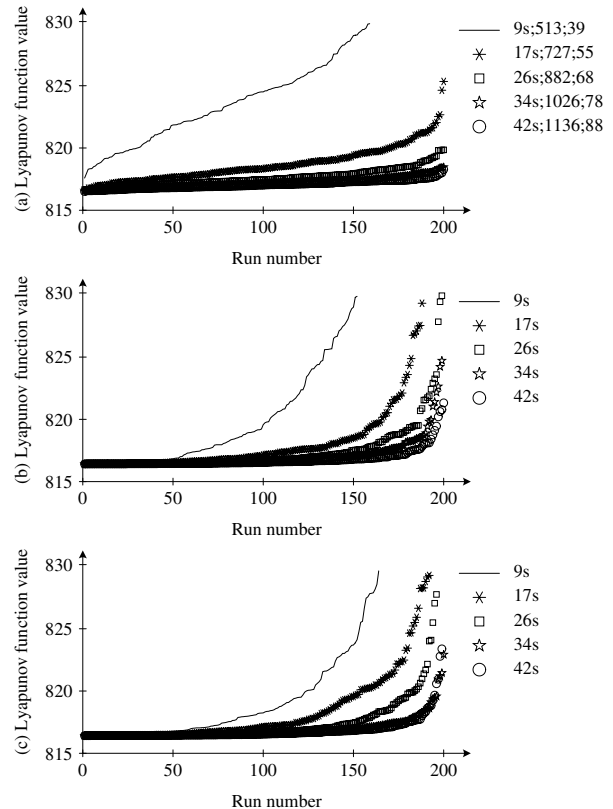


Fig. 5. System 1 optimization technique comparison. (a) Genetic algorithm. (b) Nelder-Mead method. (c) quasi-Newton method

TABLE I
SIMULATION CONFIDENCE CHECK RESULTS

	Simulation RAS contour value	Optimization RAS contour value	Number of simulations
System 1	1050.08	808.24	950,000
System 2	1533.1	1107.9	1,437,414
System 3	856.11	836.33	3,315,341

by the optimization procedures. The confidence check was passed for all three systems at 99% of values obtained. This indicates that the genetic algorithm was able to find the solution to (7) accurately.

The time-domain simulation confidence check was also run for the three systems. The results of the procedure are summarized in Table I. In this table, the minimum value found by the optimization procedure is compared to the minimum Lyapunov function value corresponding to an unstable simulation trajectory. In all cases, no unstable trajectories were found that would violate the RAS prediction yielded by the optimization procedure.

REFERENCES

- [1] B. Friedland, *Advanced Control System Design*. Englewood Cliffs, NJ: Prentice Hall, 1996.
- [2] K. L. Praprost and K. A. Loparo, "A stability theory for constrained dynamic systems with applications to electric power systems," *IEEE Trans. Automat. Contr.*, vol. 41, no. 11, pp. 1605–1617, Nov. 1996.

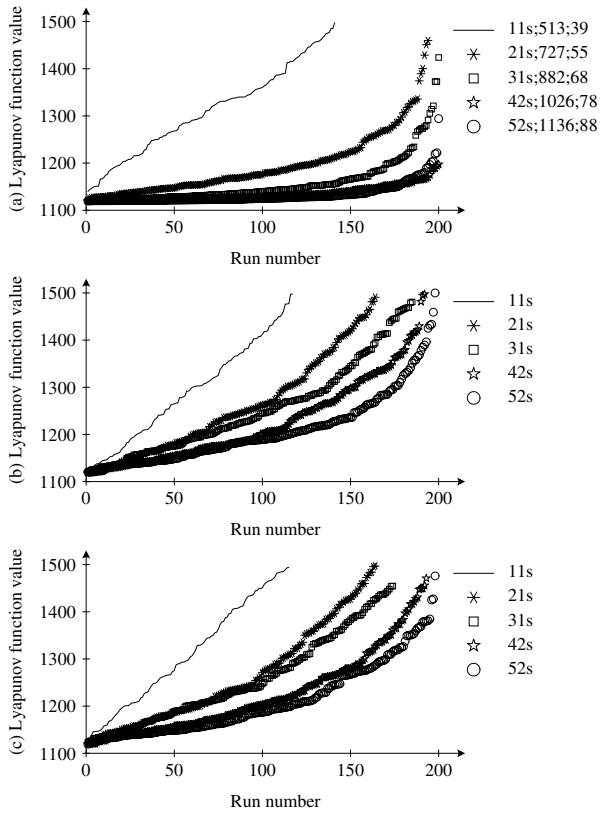


Fig. 6. System 2 optimization technique comparison. (a) Genetic algorithm. (b) Nelder-Mead method. (c) quasi-Newton method

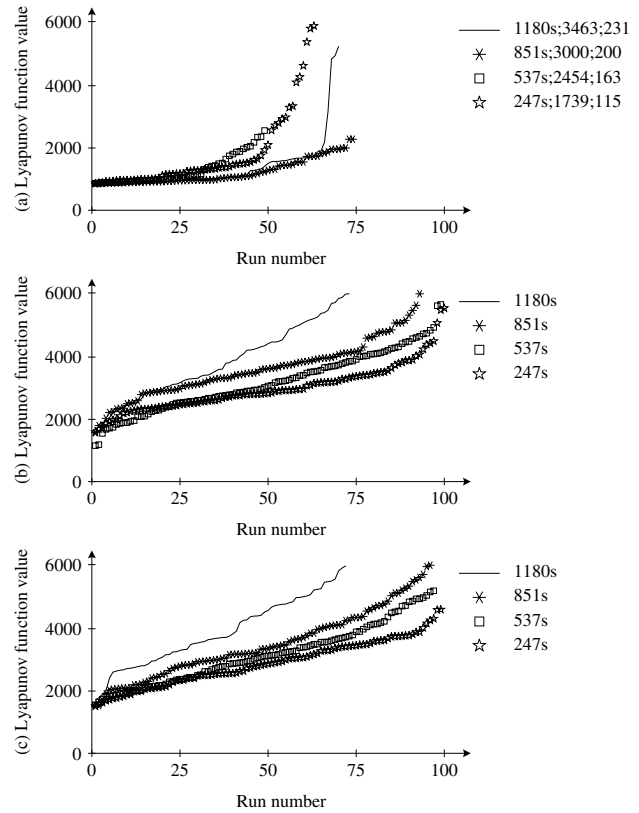


Fig. 7. System 3 optimization technique comparison. (a) Genetic algorithm. (b) Nelder-Mead method. (c) quasi-Newton method

- [3] Y. Lee and S. D. Sudhoff, "Energy Systems Analysis Consortium (ESAC) Genetic Optimization and Systems Engineering Tool, v. 1.05 Manual," School Electrical and Computer Engr., Purdue Univ., West Lafayette, IN, 47907, 2004.
- [4] B. P. Loop, "Estimating regions of asymptotic stability of nonlinear systems with applications to power electronics systems," Ph.D. dissertation, Purdue Univ., West Lafayette, IN, Dec. 2004, in preparation.
- [5] C. Kwon and S. D. Sudhoff, "A genetic algorithm based induction machine characterization procedure with application to maximum torque per amp control," *submitted to IEEE Trans. Energy Conversion*, 2004.
- [6] —, "A genetic algorithm based induction machine characterization procedure," *accepted to International Electric Machines and Drives Conference*, May 2005.
- [7] —, "An adaptive maximum torque per amp control strategy," *accepted to International Electric Machines and Drives Conference*, May 2005.
- [8] B. Cassimere, S. D. Sudhoff, B. Cassimere, and M. Swinney, "Time-domain design of motor drive current regulators using genetic algorithms," *accepted to International Electric Machines and Drives Conference*, May 2005.
- [9] S. D. Sudhoff, S. Pekarek, B. Kuhn, S. Glover, J. Sauer, and D. Delisle, "Naval combat survivability testbeds for investigation of issues in shipboard power electronics based power and propulsion systems," *Power Engineering Society Summer Meeting*, vol. 1, pp. 347–350, July 2002.
- [10] S. F. Glover, "Modeling and stability analysis of power electronics based systems," Ph.D. dissertation, Purdue Univ., West Lafayette, IN, May 2003.
- [11] S. D. Sudhoff and S. F. Glover, "Modeling techniques, stability analysis, and design criteria for dc power systems with experimental validation," *1998 Society of Automotive Engineers Transactions, Journal of Aerospace*, pp. 52–67.
- [12] S. Sanders and G. Verghese, "Synthesis of averaged circuit models for switched power converters," *IEEE International Symposium on Circuits and Systems*, vol. 1, pp. 679–683, May 1990.
- [13] P. T. Krein, J. Bentsman, R. M. Bass, and B. L. Lesieutre, "On the use of averaging for the analysis of power electronic systems," *IEEE Trans. Energy Conversion*, vol. 5, pp. 182–190, Apr. 1990.
- [14] B. T. Kuhn, S. D. Sudhoff, and C. A. Whitcomb, "Performance characteristics and average-value modeling of auxiliary resonant commutated pole converter based induction machine drives," *IEEE Trans. Power Electron.*, vol. 14, pp. 493–499, Sept. 1999.
- [15] J. Sun and H. Grotstollen, "Symbolic analysis methods for averaged modeling of switched power converters," *IEEE Trans. Power Electron.*, vol. 12, pp. 537–546, May 1997.
- [16] E. K. P. Chong and S. H. Żak, *An Introduction to Optimization*, 2nd ed. New York, NY: John Wiley and Sons, Inc., 2001.
- [17] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, "Convergence properties of the Nelder-Mead simplex method in low dimensions," *SIAM J. Optim.*, vol. 9, no. 1, pp. 112–147, Dec. 1998.
- [18] J. A. Nelder and R. Mead, "A simplex method for function minimization," *Computer Journal*, vol. 7, pp. 308–313, 1965.
- [19] T. Coleman, M. A. Branch, and A. Grace, *Optimization Toolbox for Use with Matlab*. Natick, MA: Mathworks, Inc., 1999.