2011 50th IEEE Conference on Decision and Control and
European Control Conference (CDC-ECC)
Orlando, FL, USA, December 12-15, 2011

Multivariable Newton–Based Extremum Seeking

Azad Ghaffari, Miroslav Krstić, and Dragan Nešić

Abstract— **We present a Newton–based extremum seeking algorithm for the multivariable case. The design extends the recent Newton–based extremum seeking algorithms for the scalar case and introduces a dynamic estimator of the Hessian matrix that removes the difficulty with the possible singularity of this matrix estimate. This estimator has the form of a differential Riccati equation. We prove local stability of the new algorithm for general nonlinear dynamic systems using averaging and singular perturbations. In comparison with the standard gradient–based multivariable extremum seeking, the proposed algorithm removes the dependence of the convergence rate on the unknown Hessian matrix and makes the convergence rate, of both the parameter estimates and of the estimates of the Hessian inverse, user-assignable. In particular, the new algorithm allows all the parameters to converge with the same speed, even with maps that have highly elongated level sets. In the parameter space, the new algorithms produces trajectories straight to the extremum, as opposed to non–direct "steepest descent" trajectories. Simulation results show the advantage of the proposed approach over gradient–based extremum seeking.**

I. Introduction

a) Motivation: Dramatic advances have occurred over the past decade both in the theory [2], [3], [7], [12], [16], [17], [18], [19], [20], [21] and in applications [4], [5], [6], [8], [9], [10], [11], [13], [22], [23] of extremum seeking control. All these references employ gradient–based extremum seeking.

A Newton–based extremum seeking algorithm was introduced in [14] where, for the single-input case, an estimate of the second derivative of the map was employed in a Newton-like continuous-time algorithm. A generalization, employing a different approach than in [14], was presented in [15], where a methodology for generating estimates of higher-order derivatives of the unknown single-input map was introduced, for emulating more general continuous-time optimization algorithms, with a Newton algorithm being a special case.

The key distinction of the Newton algorithm relative to the gradient algorithm is that, while the convergence of the gradient algorithm is dictated by the second derivative (Hessian) of the map, the convergence of the Newton algorithm is independent of the Hessian and can be arbitrarily assigned. This is particularly important in non-model based algorithms, like extremum seeking, where the Hessian is unknown.

The power of the Newton algorithm is particularly evident in multi-input optimization problems. With the Hessian being a matrix in this case, and with it being typically very different from the identity matrix, the gradient algorithm typically results in different elements of the input vector converging at vastly different speeds. This problem is inherent to gradient–based schemes. To rectify it one would need to modify the algorithm using the inverse of the Hessian matrix which is not available as the model of the system is assumed to be unavailable. On the other hand, the Newton algorithm, if equipped with a convergent estimator of the Hessian matrix, achieves convergence of all the elements of the input vector at the same, or at arbitrarily assignable, rates.

b) Results of the paper: In this paper we present a multivariable Newton–based extremum seeking algorithm, which yields arbitrarily assignable convergence rates for each of the elements of the input vector. We generate the estimate of the Hessian matrix by generalizing the idea proposed in [15] for the scalar case.

Generating an estimate of the Hessian matrix in non-model based optimization is not the only challenge. The other challenge is that the Newton algorithm requires an inverse of the Hessian matrix. The estimate of this matrix, as it evolves in continuous time, need not necessarily remain invertible. We tackle this challenge by employing a dynamic system for generating the inverse asymptotically. This dynamic system is a filter in the form of a Riccati differential equation. When fed with a positive/negative–definite estimate of the Hessian matrix over a longer period of time, this filter converges to a positive/negative–definite inverse of the Hessian matrix. Hence, after a transient, our non–model based algorithm behaves (on average) as a model-based Newton algorithm.

While the basic idea of our algorithm is developed for static maps, we provide the analysis of convergence when the algorithm is applied to general nonlinear systems, as in [12]. We apply classical averaging and singular perturbation methods, so our stability result is local—the parameter estimates start not too far from the true parameters and the estimate of the Hessian matrix starts not too far from the true Hessian matrix. We can also prove non–local stability of the proposed scheme in a similar manner as [19] where gradient–based algorithm was investigated.

The continuous–time Newton algorithm that we propose is novel, to our knowledge, even in the case when the cost function being optimized is known. The state–of–the–art continuous–time Newton algorithm in [1] employs a Lyapunov differential equation for estimating the inverse of

Azad Ghaffari is with Joint–Doctoral Programs (Aerospace and Mechanical) between San Diego State University and University of California at San Diego, La Jolla, CA 92093-0411, USA, aghaffar@ucsd.edu.

Miroslav Krstić is with Department of Mechanical and Aerospace Engineering, University of California, San Diego, La Jolla, CA 92093-0411, USA, krstic@ucsd.edu.

Dragan Nešić is with Department of Electrical and Electronic Engineering, The University of Melbourne, VIC 3010, Australia, d.nesic@ee.unimelb.edu.au.

This research was supported by the Australian Research Council under the Discovery Grants scheme.

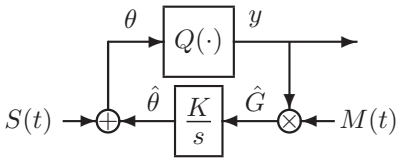

Fig. 1. Gradient–based extremum seeking for a static map.

the Hessian matrix—see (3.2) in [1]. The convergence of this estimator is actually governed by the Hessian matrix itself. This means that the algorithm in [1] removes the difficulty with inverting the estimate of the Hessian, but does not achieve independence of the convergence rate from the Hessian. In contrast, our algorithm's convergence rate is independent from the Hessian and is user–assignable.

c) Organization: We state the problem and review the gradient–based extremum seeking algorithm for a static map in Section II. Section III presents our Newton–based scheme for the static map. In this section we explain how we generate the estimate of the Hessian matrix and the estimate of its inverse. A generalization of the Newton–based scheme to dynamic plants is introduced in Section IV. The main stability result is stated in Section V. Stability analysis of the reduced order model based on the averaging theorem is presented in Section VI. Section VII presents an illustrative example to highlight the difference between the proposed scheme and the standard gradient–based extremum seeking. Proofs of Theorems 1 and 2 are omitted due to space constraints.

II. Review of the Gradient Algorithm for Static Map

Consider a convex static map

$$y = Q(\theta), \quad \theta = \begin{bmatrix} \theta_1 & \theta_2 & \cdots & \theta_n \end{bmatrix}^T, \tag{1}$$

with a local maximum at θ^*. The cost function is not known in (1), but we can measure y and we can manipulate θ. The gradient–based extremum seeking scheme for this multivariable static map is shown in Fig. 1, where K is a positive diagonal matrix, and the perturbation signals are defined as

$$S(t) = \begin{bmatrix} a_1 \sin(\omega_1 t) & \cdots & a_n \sin(\omega_n t) \end{bmatrix}^T, \tag{2}$$

$$M(t) = \begin{bmatrix} \frac{2}{a_1} \sin(\omega_1 t) & \cdots & \frac{2}{a_n} \sin(\omega_n t) \end{bmatrix}^T, \tag{3}$$

where ω_i/ω_j are rational for all i and j, and a_i's are real numbers, with the frequencies chosen such that $\omega_i \neq \omega_j$ and $\omega_i + \omega_j \neq \omega_k$ for distinct $i,\ j$, and k.

Remark 1: A gradient–based extremum seeking for the static map (1) is given by $\dot{\hat{\theta}} = KM(t)y$, $\theta = \hat{\theta} + S(t)$. In the parameter error variable $\tilde{\theta} = \hat{\theta} - \theta^*$, the closed-loop system in Fig. 1 is given by $\dot{\tilde{\theta}} = KM(t)Q\Big(\theta^* + \tilde{\theta} + S(t)\Big)$. The basic idea of the scheme, as well as of the choice of the perturbation signals, is understood by noting that, for the case of a quadratic map, $Q(\theta) = Q^* + \frac{1}{2}(\theta - \theta^*)^T H(\theta - \theta^*)$, the averaged system is given by

$$\dot{\tilde{\theta}} = KH\tilde{\theta}, \tag{4}$$

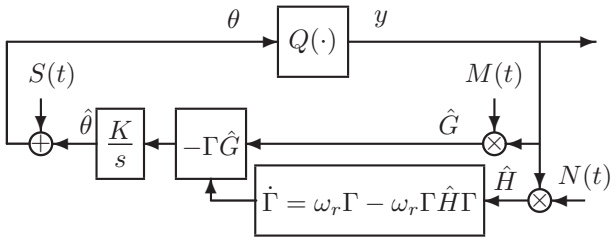

Fig. 2. Newton–based extremum seeking for a static map.

where H is the Hessian of the static map, and it is negative definite. This observation reveals two things: (a) the gradient–based extremum seeking algorithm is locally convergent, and (b) the convergence rate is governed by the unknown Hessian matrix H. One of the features of the Newton algorithm presented in the next section is to eliminate the dependence of the convergence rate on the unknown H.

III. Newton Algorithm for Static Map

The Newton–based extremum seeking algorithm for a static map is shown in Fig. 2, where ω_r is a positive real number. There are two vital parts in the Newton–based algorithm: the perturbation matrix $N(t)$, which generates an estimate of the Hessian, and the Riccati equation, which generates an estimate of the inverse of Hessian matrix, even when the estimate of the Hessian is singular.

The idea for producing the estimate of the Hessian matrix, $H := \partial^2 Q(\theta^*)/\partial\theta^2$, is borrowed from the scalar design in [15]. Referring to the Taylor series expansion of the cost function around the peak, we have

$$\begin{aligned} y =& Q(\theta^* + \tilde{\theta} + S(t)) \\ =& Q(\theta^*) + \frac{1}{2}\Big(\tilde{\theta} + S(t)\Big)^T H\Big(\tilde{\theta} + S(t)\Big) + R(\tilde{\theta} + S(t)), \end{aligned} \tag{5}$$

where $\partial Q(\theta^*)/\partial\theta = 0$ and $R(\tilde{\theta} + S(t))$ stands for higher order terms in $\tilde{\theta} + S(t)$. Product of $N(t)$ and y generates an estimate of the Hessian. We show that by an appropriate selection of matrix $N(t)$, the average value of $\hat{H} = N(t)y$ over the period Π, which is related to ω_i's (see (10)), is an estimate of the Hessian. We start with

$$\frac{1}{\Pi}\int_0^\Pi N(\sigma) y \mathrm{d}\sigma = I + J + \bar{H} + \frac{1}{\Pi}\int_0^\Pi R(\tilde{\theta} + S(\sigma)) N(\sigma) \mathrm{d}\sigma, \tag{6}$$

$$I := \frac{1}{\Pi}\int_0^\Pi \left(Q(\theta^*) + \frac{1}{2}\tilde{\theta}^T H \tilde{\theta}\right) N(\sigma)\mathrm{d}\sigma, \tag{7}$$

$$J := \frac{1}{\Pi}\int_0^\Pi \tilde{\theta}^T H S(\sigma) N(\sigma)\mathrm{d}\sigma, \tag{8}$$

$$\begin{aligned} \bar{H} :=& \frac{1}{\Pi}\int_0^\Pi \frac{1}{2} S(\sigma)^T H S(\sigma) N(\sigma)\mathrm{d}\sigma \\ =& \frac{1}{\Pi}\int_0^\Pi \frac{1}{2}\sum_{i=1}^n\sum_{j=1}^n H_{i,j}\sin(\omega_i\sigma)\sin(\omega_j\sigma)N(\sigma)\mathrm{d}\sigma. \end{aligned} \tag{9}$$

By taking Π as

$$\Pi = 2\pi \times \mathrm{LCM}\Big\{\frac{1}{\omega_i}\Big\}, \quad i \in \{1,\ 2,\ \cdots,\ n\}, \tag{10}$$

where LCM stands for the least common multiple, we have $I = 0$ if N has zero average over Π. Also, taking N such that $\frac{1}{\Pi}\int_0^\Pi \sin(\omega_i\sigma)N_{j,k}(\sigma)\mathrm{d}\sigma = 0$, holds for all $i,\ j,$ and $k \in \{1, 2, \cdots, n\}$, makes the integral J equal to zero. Furthermore, $\bar{H}$ is equal to H if we choose N such that

$$\frac{1}{\Pi}\int_0^\Pi \sin^2(\omega_i\sigma)N_{i,i}(\sigma)\mathrm{d}\sigma \neq 0 \tag{11}$$

$$\frac{1}{\Pi}\int_0^\Pi \sin(\omega_i\sigma)\sin(\omega_j\sigma)N_{i,j}(\sigma)\mathrm{d}\sigma \neq 0 \tag{12}$$

$$\frac{1}{\Pi}\int_0^\Pi \sin^2(\omega_i\sigma)N_{i,j}(\sigma)\mathrm{d}\sigma = 0 \tag{13}$$

$$\frac{1}{\Pi}\int_0^\Pi \sin(\omega_i\sigma)\sin(\omega_j\sigma)N_{i,i}(\sigma)\mathrm{d}\sigma = 0, \tag{14}$$

for all distinct i and j. Noting that Π is the common period of the probing frequencies we have

$$\frac{1}{\Pi}\int_0^\Pi \sin^2(\omega_i\sigma)\cos(2\omega_i\sigma)\mathrm{d}\sigma = -\frac{1}{4} \tag{15}$$

$$\frac{1}{\Pi}\int_0^\Pi \sin^2(\omega_i\sigma)\sin^2(\omega_j\sigma)\mathrm{d}\sigma = \frac{1}{4} \tag{16}$$

$$\frac{1}{\Pi}\int_0^\Pi \sin^3(\omega_i)\sin(\omega_j)\mathrm{d}\sigma = 0 \tag{17}$$

$$\frac{1}{\Pi}\int_0^\Pi \sin(\omega_i\sigma)\sin(\omega_j\sigma)\cos(2\omega_i\sigma)\mathrm{d}\sigma = 0, \tag{18}$$

for all $i \neq j$. Hence, one possible choice of elements of the $n\times n$ matrix $N(t)$ that satisfy all of the aforementioned constraints is given by

$$N_{i,i} = \frac{16}{a_i^2}\Big(\sin^2(\omega_i t) - \frac{1}{2}\Big) \tag{19}$$

$$N_{i,j} = \frac{4}{a_i a_j}\sin(\omega_i t)\sin(\omega_j t), \qquad i \neq j, \tag{20}$$

where $N^T(t) = N(t)$. Based on this selection, we have

$$\frac{1}{\Pi}\int_0^\Pi N(\sigma)y\mathrm{d}\sigma = H + \frac{1}{\Pi}\int_0^\Pi R(\tilde{\theta}+S(\sigma))N(\sigma)\mathrm{d}\sigma. \tag{21}$$

In Section VI we show that this averaged value is close enough to the actual value of the Hessain, under specific conditions on ω_i and a_i.

Computing the inverse of the Hessian matrix is the next step. Calculating Γ, the estimate of the inverse of the Hessian, in an algebraic fashion creates difficulties when the matrix $\hat{H}$ is close to singularity, or it is indefinite. To deal with this problem, a dynamic estimator is employed to calculate the inverse of $\hat{H}$ using a Riccati equation. Consider the following filter

$$\dot{\mathcal{H}} = -\omega_r\mathcal{H} + \omega_r\hat{H}. \tag{22}$$

Note that the state of this filter converges to $\hat{H}$, an estimate of H. Denote $\Gamma = \mathcal{H}^{-1}$. Since $\dot{\Gamma} = -\Gamma\dot{\mathcal{H}}\Gamma$, then equation (22) is transformed to the differential Riccati equation

$$\dot{\Gamma} = \omega_r\Gamma - \omega_r\Gamma\hat{H}\Gamma. \tag{23}$$

The equilibria of the Riccati equation (23) are $\Gamma^* = 0_{n\times n}$ and $\Gamma^* = \hat{H}^{-1}$, provided $\hat{H}$ settles to a constant. Since $\omega_r > 0$, the equilibrium $\Gamma^* = 0$ unstable, whereas the linearization of (23) around $\Gamma^* = \hat{H}^{-1}$ shows that $\dot{\Gamma} = -\omega_r\Gamma$, the other equilibrium is locally exponentially stable. This shows that, after a transient, the Riccati equation converges to the actual value of the inverse of Hessian matrix if $\hat{H}$ is a good estimate of H.

Remark 2: To highlight the contrast between the Newton and gradient algorithms, we refer to Remark 1 where the average behavior of the gradient algorithm is discussed. For the Newton algorithm in Fig. 2, the algorithm is given by

$$\dot{\hat{\theta}} = -K\Gamma M(t)y \tag{24}$$

$$\dot{\Gamma} = \omega_r\Gamma - \omega_r\Gamma N(t)y\Gamma, \tag{25}$$

where $\theta = \hat{\theta} + S(t)$.

In the error variables $\tilde{\theta} = \hat{\theta} - \theta^*, \tilde{\Gamma} = \Gamma - H^{-1}$, when the map is quadratic, $Q(\theta) = Q^* + \frac{1}{2}(\theta-\theta^*)^T H(\theta-\theta^*)$, the averaged closed-loop system is given by

$$\dot{\tilde{\theta}} = -K\tilde{\theta} - K\tilde{\Gamma}H\tilde{\theta} \tag{26}$$

$$\dot{\tilde{\Gamma}} = -\omega_r\tilde{\Gamma} - \omega_r\tilde{\Gamma}H\tilde{\Gamma}, \tag{27}$$

where $K\tilde{\Gamma}H\tilde{\theta}$ is quadratic in $\tilde{\Gamma}$ and $\tilde{\theta}$, and $\omega_r\tilde{\Gamma}H\tilde{\Gamma}$ is quadratic in $\tilde{\Gamma}$. The linearization of this system has all of its eigenvalues at $-K$ and $-\omega_r$. Hence, unlike the gradient algorithm, whose convergence is governed by the unknown Hessian H, the convergence rate of the Newton algorithm can be arbitrarily assigned by the designer with an appropriate choice of K and ω_r.

IV. Newton Algorithm for Dynamic Systems

Consider a general multi-input-single-output (MISO) nonlinear model

$$\dot{x} = f(x, u) \tag{28}$$

$$y = h(x), \tag{29}$$

where $x \in \mathbb{R}^m$ is the state, $u \in \mathbb{R}^n$ is the input, $y \in \mathbb{R}$ is the output, and $f : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^m$ and $h : \mathbb{R}^m \to \mathbb{R}$ are smooth. Suppose that we know a smooth control law $u = \alpha(x, \theta)$ parametrized by a vector parameter $\theta \in \mathbb{R}^n$. The closed–loop system $\dot{x} = f(x, \alpha(x, \theta))$ then has equilibria parametrized by θ. We make the following assumptions about the closed-loop system, as in [12].

Assumption 1: There exists a smooth function $l{:}\mathbb{R}^n \to \mathbb{R}^m$ such that $f(x, \alpha(x, \theta)) = 0$ if and only if $x = l(\theta)$.

Assumption 2: For each $\theta \in \mathbb{R}^n$, the equilibrium $x = l(\theta)$ of the system $\dot{x} = f(x, \alpha(x, \theta))$ is locally exponentially stable uniformly in θ.

Assumption 3: There exists $\theta^* \in \mathbb{R}^n$ such that

$$\frac{\partial}{\partial\theta}(h\circ l)(\theta^*) = 0, \tag{30}$$

$$\frac{\partial^2}{\partial\theta^2}(h\circ l)(\theta^*) = H < 0, \qquad H = H^T. \tag{31}$$

Our objective is to develop a feedback mechanism which maximizes the steady-state value of y but without requiring

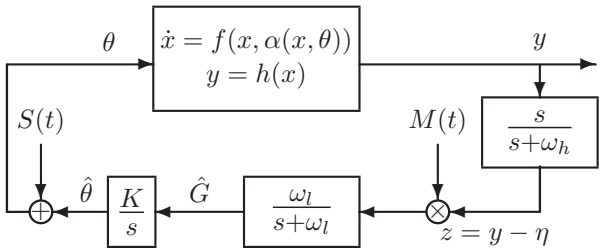

Fig. 3. Gradient–based extremum seeking.

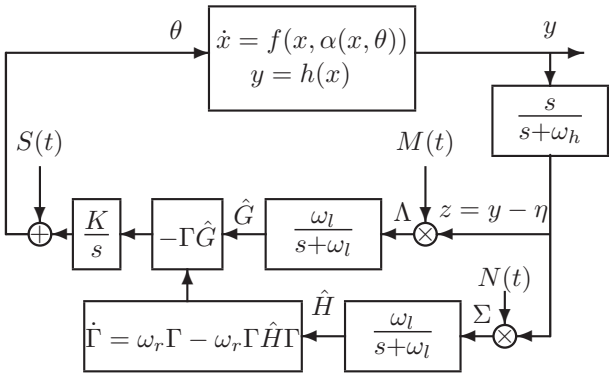

Fig. 4. Newton–based extremum seeking. The initial condition $\Gamma(0)$ should be chosen negative definite and symmetric.

the knowledge of either θ^* or the functions h and l. The gradient–based extremum seeking design that achieves this objective, suitably adapted from [12] to the multivariable case, is shown schematically in Fig. 3. Parallel to this, we present the generalized scheme for multivariable Newton–based extremum seeking as shown in Fig. 4.

The perturbation signals are defined by equations (2), (3), (19) and (20). The probing frequencies ω_i's, the filter coefficients ω_h, ω_l, and ω_r and gain K are selected as

$$\omega_i = \omega\omega_i' = O(\omega), \qquad i \in \{1, 2, \cdots, n\} \tag{32}$$

$$\omega_h = \omega\omega_H = \omega\delta\omega_H' = O(\omega\delta) \tag{33}$$

$$\omega_l = \omega\omega_L = \omega\delta\omega_L' = O(\omega\delta) \tag{34}$$

$$\omega_r = \omega\omega_R = \omega\delta\omega_R' = O(\omega\delta) \tag{35}$$

$$K = \omega K' = \omega\delta K'' = O(\omega\delta), \tag{36}$$

where ω and δ are small positive constants, ω_i' is a rational number, ω_H', ω_L', and ω_R' are $O(1)$ positive constants, and K'' is a $n \times n$ diagonal matrix with $O(1)$ positive elements.

The analysis of [2], [12], [16] shows that, in the gradient–based scheme, for "sufficiently small" ω and $|a|$, where $a = [a_1\ a_2\ \cdots\ a_n]^T$, and sufficiently small δ, which imply small filter cut–off frequencies, the states $(x, \hat{\theta})$ of the closed-loop system exponentially converge to an $O(\omega + \delta + |a|)$–neighborhood of $(l(\theta^*), \theta^*)$, and the output y converges to an $O(\omega + \delta + |a|)$–neighborhood of the optimum output $y^* = (h \circ l)(\theta^*)$.

In Section VI we show that the average value of $\Sigma(t)$ over the period Π is close enough to the actual value of the Hessian, under specific conditions on ω, δ and a. Since we are integrating over a finite time period, and we set the phase delays of the periodic perturbation signals equal to zero, it is possible to exclude condition $\omega_i \neq \omega_j + \omega_k$. The probing frequencies need to satisfy

$$\omega_i' \notin \left\{\omega_j', \frac{1}{2}(\omega_j' + \omega_k'), \omega_j' + 2\omega_k', \omega_j' + \omega_k' \pm \omega_l'\right\}, \tag{37}$$

for all distinct i, j, k, and l. As we see in section VI, ignoring these conditions is shifting the estimate of the parameter away from its true value, and leading to inaccurate estimates of the gradient vector and Hessian matrix.

V. Stability of the Closed–Loop System with the Newton–Based Extremum Seeking Algorithm

We summarize the system in Fig. 4 as

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} x \\ \tilde{\theta} \\ \hat{G} \\ \tilde{\Gamma} \\ \tilde{H} \\ \tilde{\eta} \end{bmatrix} = \begin{bmatrix} f(x, \alpha(x, \theta^* + \tilde{\theta} + S(t))) \\ -K(\tilde{\Gamma} + H^{-1})\hat{G} \\ -\omega_l\hat{G} + \omega_l\Big(y - h\circ l(\theta^*) - \tilde{\eta}\Big)M(t) \\ \omega_r(\tilde{\Gamma} + H^{-1})\Big(I - (\tilde{H} + H)(\tilde{\Gamma} + H^{-1})\Big) \\ -\omega_l\tilde{H} - \omega_l H + \omega_l\Big(y - h\circ l(\theta^*) - \tilde{\eta}\Big)N(t) \\ -\omega_h\tilde{\eta} + \omega_h\Big(y - h\circ l(\theta^*)\Big) \end{bmatrix}. \tag{38}$$

To conduct a stability analysis we introduce error variables $\tilde{\theta} = \hat{\theta} - \theta^*$, $\tilde{\eta} = \eta - h\circ l(\theta^*)$, $\tilde{\Gamma} = \Gamma - H^{-1}$, and $\tilde{H} = \hat{H} - H$, where $\theta = \hat{\theta} + S(t)$. We perform a slight abuse of notation by stacking matrix quantities $\tilde{\Gamma}$ and $\tilde{H}$ along with vector quantities, as alternative notational choices would be more cumbersome. Our main stability result is stated in the following theorem.

Theorem 1: Consider the feedback system (38) under Assumptions 1, 2 and 3. There exist $\bar{\delta}, \bar{a} > 0$ and for any $|a| \in (0, \bar{a})$ and $\delta \in (0, \bar{\delta})$ there exists $\bar{\omega} > 0$ such that for any given a and δ and any $\omega \in (0, \bar{\omega})$ there exists a neighborhood of the point $(x, \hat{\theta}, \hat{G}, \Gamma, \hat{H}, \eta) = (l(\theta^*), \theta^*, 0, H^{-1}, H, h\circ l(\theta^*))$ such that any solution of systems (38) from the neighborhood exponentially converges to an $O(\omega + \delta + |a|)$–neighborhood of that point. Furthermore, $y(t)$ converges to an $O(\omega+\delta+|a|)$–neighborhood of $h\circ l(\theta^*)$.

We summarize the system (38) in the time scale $\tau = \omega t$ as

$$\omega\frac{\mathrm{d}x}{\mathrm{d}\tau} = f(x, \alpha(x, \theta^* + \tilde{\theta} + \bar{S}(\tau))) \tag{39}$$

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\begin{bmatrix}\tilde{\theta}^T & \hat{G}^T & \tilde{\Gamma}^T & \tilde{H}^T & \tilde{\eta}\end{bmatrix}^T = \delta\begin{bmatrix} -K''(\tilde{\Gamma} + H^{-1})\hat{G} \\ -\omega_L'\hat{G} + \omega_L'\Big(y - h\circ l(\theta^*) - \tilde{\eta}\Big)\bar{M}(\tau) \\ \omega_R'(\tilde{\Gamma} + H^{-1})\Big(I - (\tilde{H} + H)(\tilde{\Gamma} + H^{-1})\Big) \\ -\omega_L'(\tilde{H} + H) + \omega_L'\Big(y - h\circ l(\theta^*) - \tilde{\eta}\Big)\bar{N}(\tau) \\ -\omega_H'\tilde{\eta} + \omega_H'\Big(y - h\circ l(\theta^*)\Big) \end{bmatrix}, \tag{40}$$

where $\bar{S}(\tau) = S(t/\omega)$, $\bar{M}(\tau) = M(t/\omega)$ and $\bar{N}(\tau) = N(t/\omega)$.

VI. Averaging Analysis

The main step in our analysis is to study the system in Fig. 4. We "freeze" x in (39) at its equilibrium value

$x = l(\theta^* + \tilde{\theta} + \bar{S}(\tau))$, and substitute it into (40), getting the reduced system

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\begin{bmatrix}\tilde{\theta}_r^T & \hat{G}_r^T & \tilde{\Gamma}_r^T & \tilde{H}_r^T & \tilde{\eta}_r\end{bmatrix}^T = \delta\begin{bmatrix} -K''(\tilde{\Gamma}_r + H^{-1})\hat{G}_r \\ -\omega_L'\hat{G}_r + \omega_L'\Big(\nu(\tilde{\theta}_r + \bar{S}(\tau)) - \tilde{\eta}_r\Big)\bar{M}(\tau) \\ \omega_R'(\tilde{\Gamma}_r + H^{-1})\Big(I + (\tilde{H}_r + H)(\tilde{\Gamma}_r + H^{-1})\Big) \\ -\omega_L'\tilde{H}_r - \omega_L'H + \omega_L'\Big(\nu(\tilde{\theta}_r + \bar{S}(\tau)) - \tilde{\eta}_r\Big)\bar{N}(\tau) \\ -\omega_H'\tilde{\eta}_r + \omega_H'\nu(\tilde{\theta}_r + \bar{S}(\tau)) \end{bmatrix}, \quad (41)$$

where $\nu(z) = h \circ l(\theta^* + z) - h \circ l(\theta^*)$. In view of Assumption 3, $\nu(0) = 0$, $\partial\nu(0)/\partial z = (0)$, $\partial^2\nu(0)/\partial z^2 = H < 0$. To prove the overall stability of (38), first we show that the reduced system (41) has a unique exponentially stable periodic solution around its equilibrium.

Theorem 2: Consider system (41) under Assumption 3. There exist $\bar{\delta}, \bar{a} > 0$ such that for all $\delta \in (0, \bar{\delta})$ and $|a| \in (0, \bar{a})$ system (41) has a unique exponentially stable periodic solution $\Big(\tilde{\theta}_r^{\Pi}(\tau), \hat{G}_r^{\Pi}(\tau), \tilde{\Gamma}_r^{\Pi}(\tau), \tilde{H}_r^{\Pi}(\tau), \tilde{\eta}_r^{\Pi}(\tau)\Big)$ of period Π and this solution satisfies

$$\Big|\tilde{\theta}_{r,i}^{\Pi}(\tau) - \sum_{j=1}^{n} c_{j,j}^i a_j^2\Big| \le O(\delta + |a|^3) \quad (42)$$

$$\Big|\hat{G}_r^{\Pi}(\tau)\Big| \le O(\delta) \quad (43)$$

$$\Big|\tilde{\Gamma}_r^{\Pi}(\tau) + \sum_{i=1}^{n}\sum_{j=1}^{n} H^{-1}W^iH^{-1}c_{j,j}^i a_j^2\Big| \le O(\delta + |a|^3) \quad (44)$$

$$\Big|\tilde{H}_r^{\Pi}(\tau) - \sum_{i=1}^{n}\sum_{j=1}^{n} W^i c_{j,j}^i a_j^2\Big| \le O(\delta + |a|^3) \quad (45)$$

$$\Big|\tilde{\eta}_r^{\Pi}(\tau) - \frac{1}{4}\sum_{i=1}^{n} H_{i,i}a_i^2\Big| \le O(\delta + |a|^4) \quad (46)$$

for all $\tau \ge 0$, where

$$\Big(W^i\Big)_{j,k} = \frac{\partial^3\nu(0)}{\partial z_i \partial z_j \partial z_k}, \quad \forall i, j, \text{and } k \in \{1, 2, \cdots, n\} \quad (47)$$

$$\begin{bmatrix} c_{j,j}^1 \\ \vdots \\ c_{j,j}^{i-1} \\ c_{j,j}^i \\ c_{j,j}^{i+1} \\ \vdots \\ c_{j,j}^n \end{bmatrix} = -\frac{1}{12}H^{-1}\begin{bmatrix} \frac{\partial^3\nu}{\partial z_j \partial z_1^2}(0) \\ \vdots \\ \frac{\partial^3\nu}{\partial z_j \partial z_{j-1}^2}(0) \\ \frac{3}{2}\frac{\partial^3\nu}{\partial z_j^3}(0) \\ \frac{\partial^3\nu}{\partial z_j \partial z_{j+1}^2}(0) \\ \vdots \\ \frac{\partial^3\nu}{\partial z_j \partial z_n^2}(0) \end{bmatrix}, \forall i, j \in \{1, 2, \cdots, n\}. \quad (48)$$

VII. Simulation Results

To illustrate the results and highlight the difference between the gradient–based and Newton–based extremum seeking methods, the following static quadratic input-output map is considered $y = Q(\theta) = Q^* + \frac{1}{2}(\theta - \theta^*)^T H(\theta - \theta^*)$.

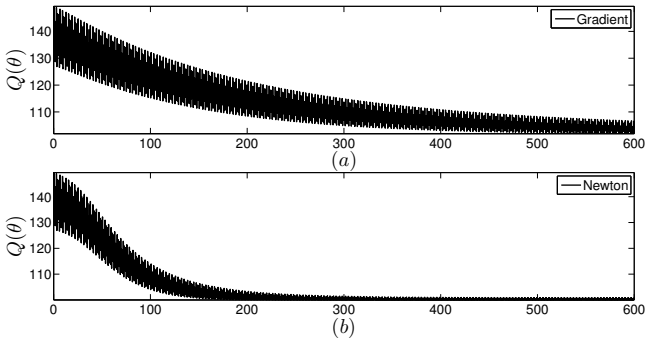

Fig. 5. The estimate of the minimum versus time.

To make a fair comparison between the two methods, all parameters are chosen the same except the gain matrix. Before selecting matrix K we investigate the performance of the gradient–based scheme versus the Newton–based scheme.

Recall (26) and (27). The initial convergence rate for the Newton–based scheme is governed by the time-varying matrix $-K_n\Gamma(t)H$. Equation (4) shows that in the gradient–based scheme the convergence depends on the eigenvalues of K_gH. This means that, to have a fair comparison between the two methods, we should select K_g and K_n such that $K_g = -K_n\Gamma(0)$.

We perform our tests with the following parameters, $\delta = 0.1$, $\omega = 0.1\text{rad/s}$, $\omega_1 = 70\omega$, $\omega_2 = 50\omega$, $\omega_L' = 10$, $\omega_H' = 8$, $\omega_R' = 10$, $a = [0.1 \;\; 0.1]^T$, $K_g'' = 10^{-4}\text{diag}([-25 \;\; -25])$, $K_n'' = \text{diag}([1 \;\; 1])$, $\Gamma_0^{-1} = 400\text{diag}([1 \;\; 1])$, $\hat{\theta}_0 = [2.5 \;\; 5]^T$, $Q^* = 100$, $\theta^* = [2 \;\; 4]^T$, $H_{11} = 100$, $H_{12} = H_{21} = 30$, and $H_{22} = 20$.

Fig. 5 illustrates the estimate of the minimum. Evolution of the parameters is depicted in Fig. 6. Since the initial estimate of the Hessian is not true, each parameter starts to update with a different rate. As seen in Fig. 7, after 40 seconds the estimate of the Hessian is close enough to its actual value. Hence, the convergence rates of both parameters are the same after 40 seconds. Furthermore, Fig. 6(c) shows that the Newton–based extremum seeking moves the parameters to the peak along a shorter trajectory than the gradient–based method. The Hessian matrix converges to its actual value as depicted in Fig. 7. Also it is worth noting that the Hessian converges faster to its actual value than $\hat{G}$ and $\hat{\theta}$. As illustrated in Fig. 8 the estimate of the gradient vector converges to zero after Hessian matrix finds its true value.

VIII. Conclusions

Using the gradient–based extremum seeking in the multivariable case without having a good information about the curvature of the cost function, namely, the Hessian matrix, may result inappropriate performance. With a growing number of the parameters, it is almost impossible to tune the convergence rate of all parameters in a desirable fashion. The Newton–based extremum seeking, which relies on the estimation of the gradient and Hessian matrix of the

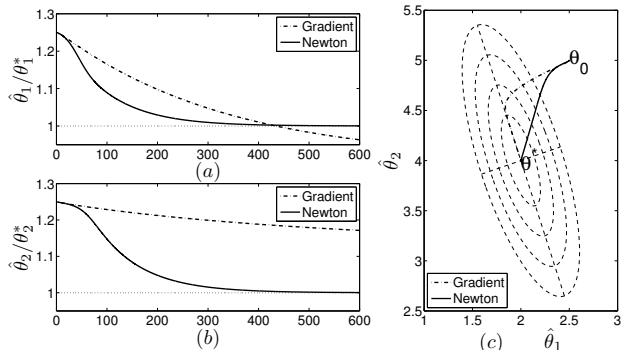

Fig. 6. Parameter estimates. (a and b) time responses. (c) phase portrait. The Newton trajectory is straight to the extremum, whereas the gradient trajectory follows the curved, steepest-descent path.

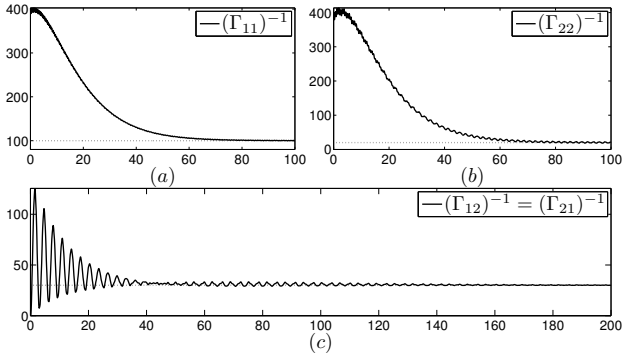

Fig. 7. Time evolution of the Hessian matrix estimator Γ^{-1}. The true value of H is reached in 40 seconds. Note in Fig. 6 that the Newton and gradient trajectories coincide for the first 40 seconds, after which, Newton takes a straight path.

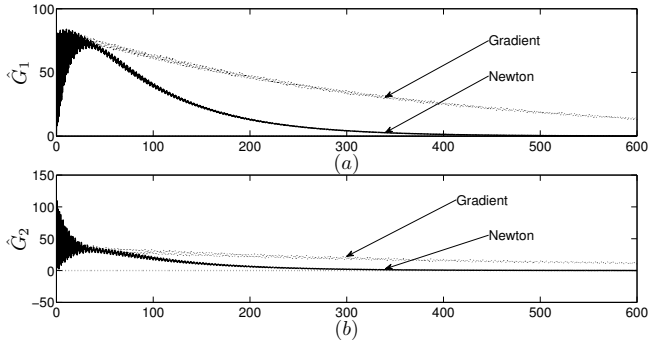

Fig. 8. The estimate of the gradient vector versus time.

cost function at the same time, removes the trial and error process to update all parameters uniformly. Furthermore, the proposed Newton scheme eliminates the concern about the inversion of the Hessian estimate matrix by performing the inversion dynamically using a Riccati equation filter. The convergence rates of both the parameter and of the estimator of the Hessian inverse are independent of the unknown Hessian and can be assigned arbitrary by the user.

REFERENCES

[1] R. Airapetyan, "Continuous newton method and its modification," *Applicable Analysis*, vol. 73, pp. 463–484, 1999.

[2] K. B. Ariyur and M. Krstić, "Analysis and design of multivariable extremum seeking," in *Proc. of the American Control Conference*, 2002.

[3] ——, *Real-Time Optimization by Extremum Seeking Feedback*. Wiley-Interscience, 2003.

[4] A. Banaszuk, K. B. Ariyur, M. Krstić, and C. A. Jacobson, "An adaptive algorithm for control of combustion instability," *Automatica*, vol. 40, pp. 1965–1972, 2004.

[5] R. Becker, R. King, R. Petz, and W. Nitsche, "Adaptive closed-loop separation control on a high-lift configuration using extremum seeking," *AIAA Journal*, vol. 45, pp. 1382–1392, 2007.

[6] D. Carnevale, A. Astolfi, C. Centioli, S. Podda, V. Vitale, and L. Zaccarian, "A new extremum seeking technique and its application to maximize RF heating on FTU," *Fusion Engineering and Design*, vol. 84, pp. 554–558, 2009.

[7] J.-Y. Choi, M. Krstić, K. B. Ariyur, and J. S. Lee, "Extremum seeking control for discrete-time systems," *IEEE Trans. Automat. Contr.*, vol. 47, pp. 318–323, 2002.

[8] J. Cochran, E. Kanso, S. D. Kelly, H. Xiong, and M. Krstić, "Source seeking for two nonholonomic models of fish locomotion," *IEEE Trans. Robot. Automat.*, vol. 25, pp. 1166–1176, 2009.

[9] J. Cochran and M. Krstić, "Nonholonomic source seeking with tuning of angular velocity," *IEEE Trans. Automat. Contr.*, vol. 54, pp. 717–731, 2009.

[10] M. Guay, M. Perrier, and D. Dochain, "Adaptive extremum seeking control of nonisothermal continuous stirred reactors," *Chemical Engineering Science*, vol. 60, pp. 3671–3681, 2005.

[11] N. J. Killingsworth, S. M. Aceves, D. L. Flowers, F. Espinosa-Loza, and M. Krstić, "HCCI engine combustion-timing control: Optimizing gains and fuel consumption via extremum seeking," *IEEE Trans. Contr. Syst. Technol.*, vol. 17, pp. 1350–1361, 2009.

[12] M. Krstić and H.-H. Wang, "Stability of extremum seeking feedback for general nonlinear dynamic systems," *Automatica*, vol. 36, pp. 595–601, 2000.

[13] L. Luo and E. Schuster, "Mixing enhancement in 2d magnetohydrodynamic channel flow by extremum seeking boundary control," in *Proc. of the American Control Conference*, 2009.

[14] W. H. Moase, C. Manzie, and M. J. Brear, "Newton-like extremum-seeking for the control of thermoacoustic instability," *IEEE Trans. Automat. Contr.*, vol. 55, pp. 2094–2105, 2010.

[15] D. Nešić, Y. Tan, W. H. Moase, and C. Manzie, "A unifying approach to extremum seeking: adaptive schemes based on estimation on the estimation of derivatives," in *Proc. of IEEE Conf. on Decision and Control*, 2010.

[16] M. A. Rotea, "Analysis of multivariable extremum seeking algorithms," in *Proc. of the American Control Conference*, 2000.

[17] M. S. Stankovic, K. H. Johansson, and D. M. Stipanovic, "Distributed seeking of nash equilibria in mobile sensor networks," in *Proc. of IEEE Conf. on Decision and Control*, 2010.

[18] M. S. Stankovic and D. M. Stipanovic, "Extremum seeking under stochastic noise and applications to mobile sensors," *Automatica*, vol. 46, pp. 1243–1251, 2010.

[19] Y. Tan, D. Nešić, and I. Mareels, "On non-local stability properties of extremum seeking control," *Automatica*, vol. 42, pp. 889–903, 2006.

[20] A. R. Tee1 and D. Popovic, "Solving smooth and nonsmoooth multivariable extremum seeking problems by the methods of nonlinear programming," in *Proc. of the American Control Conference*, 2001.

[21] H.-H. Wang and M. Krstić, "Extremum seeking for limit cycle minimization," *IEEE Trans. Automat. Contr.*, vol. 45, pp. 2432–2437, 2000.

[22] H.-H. Wang, S. Yeung, and M. Krstić, "Experimental application of extremum seeking on an axial-flow compressor," *IEEE Trans. Contr. Syst. Technol.*, vol. 8, pp. 300–309, 2000.

[23] C. Zhang, D. Arnold, N. Ghods, A. Siranosian, and M. Krstić, "Source seeking with nonholonomic unicycle without position measurement and with tuning of forward velocity," *Systems & Control Letters*, vol. 56, pp. 245–252, 2007.