MODEL-BASED RESONANCE TRACKING OF LINEAR TIME-VARYING SYSTEMS

A PREPRINT

Thomas Vasileiou
ESA - Advanced Concepts Team
European Space Research and Technology Centre (ESTEC)
Keplerlaan 1
NL-2200AG Noordwijk, Netherlands
Thomas.Vasileiou@esa.int

September 10, 2018

ABSTRACT

This paper develops a recursive algorithm for tracking resonance frequency shifts of linear time-varying systems corrupted by additive white Gaussian noises, in real time. So far, automatic resonance tracking is limited to non-model-based designs that rely solely on the phase difference between a selected input and output of the system. Instead, we propose a transformation of the system into a complex-valued representation, which allows to abstract the resonance shifts as an exogenous disturbance acting on the excitation frequency, perturbing it from the natural frequency of the plant. The resonance tracking problem is then split in two tasks; identifying the frequency disturbance and solving an optimization problem to determine the excitation frequency. The design of the resonance tracker is therefore simplified, due to the application of well-established techniques. We discuss the stability of the proposed scheme, even in cases that seriously challenge the current phase-based approaches, like non-monotonic phase differences and multiple-input multiple-output systems. Numerical simulations further demonstrate the performance of the resonance tracking scheme.

Keywords resonance · frequency tracking · time-varying systems · complex variables · closed-loop identification.

1 Introduction

Precise tracking of the resonance frequency of oscillating systems is of great interest in resonant sensing [1][2] and when driving vibrating loads [3][4]. Resonant sensors, the function of which relies on the resonant characteristic of a vibrating structure, have been proposed for a wide range of measurements like thermometers [5], accelerometers [6], viscometers [7], humidity [8], water cut measurement [9] and gyroscopes [10][11] to mention a few. In term of miniaturization and increased sensitivity, micro-electro-mechanical systems (MEMS) with vibrating cantilevers have emerged as an appealing solution with achievements like atomic force microscopy in space [12] and mass detection into the range of atto- and zepto-grams [2]. Further, resonant electromechanical actuators are widely proposed in power electronics [13][14][15][16], ultrasonic applications [17], thermosonic wire bonding [18] and acoustic particle trapping [19].

In order to increase the sensitivity of the sensors or the power output of vibrating actuators, designers adopt systems with “sharp” resonances (low damping, high quality factor) [19]. On the downside, this leads to diminished performance when the excitation frequency deviates even slightly from the resonance frequency, due to the inherently narrow bandwidth of the system. Even in case of actuators that have been designed to operate at a constant resonance, shifts from the designed operating frequency may occur because of environmental changes, like temperature and humidity [20], aging of the device [21] or changes in the load [18][15]. As a remedy, designers resort to feedback resonance tracking control to compensate for these shifts and achieve maximum efficiency [4][15]. In the case of sensing applications based on
changes of the resonance frequency with the measured quantity, the use of the feedback control is unavoidable and the performance of the control system directly affects the sensitivity, resolution and the bandwidth of the sensor [21, 22].

Regardless the application, the phase locked loop (PLL) [21, 7] and the self-sustained oscillation (SSO) schemes [23, 24] are the norm for resonance tracking. Briefly, both techniques achieve tracking of the resonance frequency by keeping a constant phase difference between the input and the output of the system. Their main difference lies in the fact that the PLL utilizes an external oscillator to generate the signal that excites the system. In the SSO case, the excitation signal is generated by the oscillating structure itself; the system output is amplified and phase-shifted before it is fed back to the system. Design and analysis of PLL and SSO resonant tracking schemes connected with a single-input single-output (SISO) $2^\text{nd}$ order system have been extensively discussed by many authors for various applications [21, 25, 6, 26, 23, 7]. On the downside, these resonance tracking approaches are far from model-based and most importantly the closed-loop robustness and stability cannot be guaranteed. This is the case even for linear SISO systems, if the phase difference between the input and output signal is non-monotonic. This problem has been pointed out for piezoelectric actuators and multi-degree-of-freedom system, where a resonance and an anti-resonance frequency are present [27, 18, 28, 17].

Except for these two main techniques mentioned previously, specialized resonant tracking algorithms have been developed for specific cases. In the absence of phase information, a maximum output signal amplitude detection algorithm has been applied in piezoresponse force microscopy [29]. A maximum power tracking adaptive approach for driving resonant loads has been proposed in [4]: a small sinusoidal perturbation signal is added to driving signal in order to estimate the derivative of the absorbed power and update the excitation frequency. A control algorithm that tunes the system to a specific resonance frequency has been developed to address the issue of on-line modal frequency matching in vibratory gyroscopes [25]. Other specialized schemes employ controller scheduling [30] and fuzzy logic [18]. Nonetheless, all the previous approaches have been developed for operation with a specific system, which makes it hard to generalize, and may need multiple driving signals. Application of these schemes has been limited to SISO systems, making their extension on multiple-input multiple-output (MIMO) rather involved.

Both the PLL and the SSO techniques have proven their capabilities in numerous of applications. By neglecting the system model, these techniques are general enough and therefore applicable in many cases. On the other hand, tuning of the controller parameters may become a tedious task and the theoretical analysis of the closed-loop performance still requires a mathematical description for the system. If a description of the system is available, a model-based controller design approach is expected to yield improved performance. Here, we consider control algorithms able to track the resonance of a linear time-varying model. We introduce a representation of the oscillating plant, which splits the system into a linear time-invariant (LTI) model and an abstract shift of the frequency exciting the system. This model representation makes the application of well-established control and estimation techniques simple and allows to pose the resonance tracking task as a series of optimization problems.

2 Problem statement

We consider the discrete-time linear model:

\begin{align}
  x_{k+1} &= \hat{A}_k x_k + \hat{B}_k u_k + \hat{w}_k \\
  y_k &= \hat{C}_k x_k + \hat{D}_k u_k + \hat{v}_k
\end{align}

(1) (2)

where $x_k \in \mathbb{R}^n$, $u_k \in \mathbb{R}^m$, and $y_k \in \mathbb{R}^p$ denote the state, the input and the output vectors respectively, at the discrete sampling instances $k$; $\hat{w}_k \sim \mathcal{N}(0, Q)$ and $\hat{v}_k \sim \mathcal{N}(0, R)$ are uncorrelated additive white Gaussian noise, modeling the disturbance input and the measurement noise respectively. Let $\hat{\lambda}_1, \hat{\lambda}_2 \cdots \hat{\lambda}_n$ the eigenvalues of $\hat{A}$, where the dependence from $k$ was dropped for the same of simplicity. We assume that for all $k$ the system is stable ($\|\hat{\lambda}_i\| < 1$, $i = 1, 2, \ldots, n$) and that it has at least one pair of conjugated complex eigenvalues which corresponds to the resonance of interest. We denote the eigenvalue of interest with $\hat{\lambda}_i$ and its corresponding resonance $\hat{\omega}_i = \arg(\hat{\lambda}_i) > 0$. The system is subjected to the sinusoidal input $u_k = \beta_k \cos(\theta_k + \psi_k)$, where $\beta_k \in \mathbb{R}_{\geq 0}$ and $\psi_k \in (-\pi, \pi]^m$ denote the instantaneous amplitude and phase of the signal $u_k$, and $\theta_k \in \mathbb{R}$ is the common driving phase. The cosine function is applied to each element of the input vector and the symbol $\odot$ denotes the element-wise multiplication of vectors. In the following, we use the convention that functions with vector arguments are applied element-wise unless stated otherwise. The evolution of $\theta_k$ follows the equation

\begin{equation}
  \theta_{k+1} = \theta_k + \omega T_s = \theta_k + \omega_k
\end{equation}

(3)

where $\omega$ is the angular frequency, $T_s$ is the sampling period (which is kept constant throughout) and $\omega_k$ the normalized angular frequency of excitation.
We note that the conversion of the LTI system to the CSS can be derived by substitution of the analytic signal. We introduce the transformation for the state in the complex notation as

\[ x_{k+1} = Ax_k + Bu_k + \tilde{w}_k \]
\[ y_k = Cx_k + Du_k + \tilde{v}_k \]

which is equivalent to (11) for a fixed value of the state matrices. We denote by \( \lambda_1, \lambda_2, \ldots, \lambda_n \) the eigenvalues of \( A \) and by \( \lambda_x = \arg(\lambda) \) the eigenvalue and the resonance of interest. Inspired by [31, 21], for a LTI excited by the sinusoidal input \( u_k \), we write the state and output variables in the amplitude-phase formulation, \( x_k = a_k \cos(\theta_k + \phi_k) + \tilde{w}_k \) and \( y_k = c_k \cos(\theta_k + \delta_k) \), where \( a_k, c_k, \phi_k \) and \( \delta_k \) are time-varying vectors of appropriate dimensions. Substitution of the phase-amplitude expressions into (4) results in

\[ a_{k+1} \cos(\theta_k + \omega_k + \phi_{k+1}) = A [a_k \cos(\theta_k + \phi_k)] + B [\beta_k \cos(\theta_k + \psi_k)] \]

where in the previous expressions we have neglected the effect of \( \tilde{w}_k \) for now. We use the angle sum trigonometric identities to expand the terms in (6), as

\[ a_{k+1} \cos(\theta_k \cos(\omega_k + \phi_{k+1}) - \sin \theta_k \sin(\omega_k + \phi_{k+1})) = A [a_k \cos(\theta_k \cos \phi_k - \sin \theta_k \sin \phi_k)] + B [\beta_k \cos(\theta_k \cos \psi_k - \sin \theta_k \sin \psi_k)] . \]

We demand that (7) should hold for all \( k \) and be independent of \( \theta_k \). Since the sine and cosine functions are orthogonal, the terms that contain \( \cos \theta_k \) (or \( \sin \theta_k \)) should add up to zero. The resulting system of equations is:

\[ a_{k+1} \cos(\omega_k + \phi_{k+1}) = A (a_k \cos \phi_k) + B (\beta_k \cos \psi_k) \]
\[ a_{k+1} \sin(\omega_k + \phi_{k+1}) = A (a_k \sin \phi_k) + B (\beta_k \sin \psi_k) \]

We introduce the transformation for the state in the complex notation as \( z_k = a_k \cos(\phi_k + j \sin \phi_k) = a_k e^{j \phi_k} \), the output \( q_k = c_k \cos(\phi_k + e^{j \psi_k}) \), and the input formulation \( s_k = \beta_k e^{j \psi_k} \), where \( j = \sqrt{-1} \) is the imaginary unit. This complex representation of signals is similar to the complex envelope representation of bandpass signal in communication channels [32] and to the analytic signal [33]. We write (8) as:

\[ z_{k+1} = (Az_k + Bs_k + w_k) e^{-j \omega_k} \]
\[ q_{k+1} = Cz_k + Ds_k + v_k \]

where the random variables \( w_k \) and \( v_k \) are proper (circular) complex random variables with the complex Gaussian \( CN \) probability density function; that is \( w_k \sim CN(0, Q) \) and \( v_k \sim CN(0, R) \). The complex envelope of white real-valued Gaussian signals has been shown to be complex proper normal, where the properness arises from the stationarity assumption [34, 32]. Equation (11) follows from (5) using the same procedure. We refer to the system (10,11) as the complex state space (CSS) model.

We note that the conversion of the LTI system to the CSS can be derived by substitution of the analytic signal representation directly into (15). The derivation is simpler than the presented one, but it lacks the intuition and the reasoning for the complex representation of the variables. Nonetheless, we apply it for the derivation of a continuous-time CSS, presented in Appendix A.
3.2 Properties of the CSS

The CSS model retains most of the properties of the original LTI. In the following, we treat the CSS model as a linear system with time-varying matrices $A e^{-j\omega_k}$ and $B e^{-j\omega_k}$, with a given sequence of $\omega_i$, $i = 1, 2, \ldots, k$.

**Theorem 1.** Consider the LTI system (4-5) and its associated transform into the CSS model (10-11). The CSS system is globally exponential stable for any given sequence $\omega_i$, $i = 1, 2, \ldots, k$ if and only if the associated LTI model is stable.

**Proof.** Stability of the LTI system (4) implies the existence of a Lyapunov function $V(x_k) = x_k^H P_L x_k$, where $P_L \in \mathbb{R}^{n \times n}$, $P_L > 0$ and symmetric. The notation $P_L > 0$ is used to indicate that the matrix $P_L$ is positive definite. Given the symmetric $Q_L \in \mathbb{R}^{n \times n}$ with $Q_L > 0$, there exists a unique $P_L$ satisfying

$$A^H P_L A - P_L + Q_L = 0$$

where the superscript $H$ denotes the matrix conjugate transpose. We claim that $V(z_k)$ is also a Lyapunov function for the CSS system, for any given $\omega_i$, $i = 1, 2, \ldots, k$. Indeed $V(0) = 0$, $V(z) \rightarrow \infty \Rightarrow \|z\| \rightarrow \infty$ and

$$V(z_{k+1}) - V(z_k) = (A z_k e^{-j\omega_k})^H P_L A z_k e^{-j\omega_k} - z_k^H P_L z_k$$

$$= -z_k^H Q_L z_k \leq -\|Q_L\|\|z_k\|^2 < 0$$

where $\|Q_L\|$ denotes the induced matrix norm. Equation (13) concludes the exponential stability of (10-11). If the LTI system (4) is unstable, then the Lyapunov function will be monotonically increasing over the system trajectory, $V(z_{k+1}) - V(z_k) > 0$, implying that the CSS system is also unstable.

The two system share the same Lyapunov function and thus the reverse statement, that the stability of the LTI system can be inferred by the stability of the CSS model, can be shown the same way.

**Theorem 2.** Consider the LTI system (4-5) and its associated transform into the CSS model (10-11). The CSS model is controllable if and only if the LTI model is controllable and the CSS model is observable if and only if the LTI model is observable. The controllability and the observability of the CSS system is independent of the applied sequence $\omega_i$, $i = 1, 2, \ldots, k$.

**Proof.** Controllability of (4) implies the existence of the controllability gramian $W_c(k, 0) > 0$ in the interval $[0, k]$. Given an initial state $z_0$ and the inputs $s_i$ and $\omega_i$, $i = 0, \ldots, k - 1$, the response $z_k$ of the CSS system at time $k$ is computed by consecutive application of the map (10), resulting in

$$z_k = \Phi(k, 0) z_0 + \sum_{i=0}^{k-1} \Phi(k, i + 1) B e^{-j\omega_i} s_i$$

where $\Phi(k, i) = A^{k-i} \exp(\sum_{m=0}^{k-1} j\omega_m)$ is the state transition matrix. The controllability gramian $W_{cc}(k, 0)$ for the CSS system on the interval $[0, k]$ is

$$W_{cc}(k, 0) = \sum_{i=0}^{k-1} e^{-j\omega_i} \Phi(k, i + 1) B B^H \Phi(k, i + 1) H e^{j\omega_i}$$

$$= \sum_{i=0}^{k-1} A^k B B^T (A^k)^T = W_c(k, 0) > 0$$

which concludes that the CSS system is controllable.

In the same way, if $W_c(k, 0)$ is singular then both systems are uncontrollable. Similarly, since the two systems share the same controllability gramian, the controllability of the LTI model can be inferred for the controllability of the CSS model. Finally, the observability of the CSS system can be shown using the same reasoning for the observability gramian.

The importance of the previous remarks is twofold; first, a stable LTI cannot be destabilized by any sequence of $\omega_k$. Second, in order to be able to excite and detect $\omega_s$, $\lambda_s$ should belong to the controllable and observable subspaces of the system, which remain unaffected by $\omega_k$.

Further, the optimal control and optimal estimation problems for the CSS system are directly connected to the one for the LTI system. Assume an observable LTI system (4-5) and consider the state observer design problem with initial state estimate $x_0 = E[x_0]$ and variance $P_{x,0} = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T]$. The trajectory of the optimal state estimate $\hat{x}_{k|k}$
and the covariance matrix $P_{x,k|k}$ are given by the Kalman filter equations. For the CSS model, the optimal state observer with initial conditions $\hat{z}_0 = E[z_0]$ and $P_{z,0} = E((z_0 - \hat{z}_0)H(z_0 - \hat{z}_0)') = P_{x,0}$, is a Kalman filter with $P_{x,k|k} = P_{x,k|k}$ independent of the applied $\omega_i$, $i = 0, \ldots, k$. The previous statement is based on the fact that the properness and normality of complex random variable is retained under affine transformations. Moreover, for the proper random variable $z_k$, the random variable $z_k e^{-j\omega_k}$ has the same first and second order statistical properties. Therefore, the state estimate will be distributed $CN(\hat{z}_{k|k}, P_{k|k})$, for linear observer design. For the sake of completeness, we write the Kalman filter equations for the CSS model as commonly given into a prediction and a correction step

**Prediction step:**

$$\hat{z}_{k|k-1} = (A\hat{z}_{k-1|k-1} + Bs_k) e^{-j\omega_k}$$
$$P_{z,k|k-1} = (A e^{-j\omega_k}) P_{z,k|k-1} (A e^{-j\omega_k})^H + Q$$
$$= AP_{z,k-1|k-1}A^T + Q$$

**Correction Step:**

$$\hat{z}_{k|k} = \hat{z}_{k|k-1} + L_k (q_k - C\hat{z}_{k|k-1} - Ds_k)$$
$$L_k = P_{z,k|k-1}CT (CPC^T + R)^{-1}$$
$$P_{z,k|k} = P_{z,k|k-1} - L_k CP_{z,k|k-1}$$

where $\hat{z}_{k|k-1}$ and $\hat{z}_{k|k}$ are the state estimates given the previous and the current measurement respectively. We use the same convention for $P_{z,k|k-1}$ and $P_{z,k|k}$. Since the variance updates (17) and (20) are identical for the two system, given $P_{z,0} = P_{x,0}$, the trajectories of $P_z$ and $P_{\bar{z}}$ are identical and independent of $\omega_k$. The optimality of the previous filter, in terms of being unbiased and minimum variance, has been shown for linear time-varying complex system with proper noise [36]. Moreover, the fact that $P_{z,k|k} = P_{x,k|k}$ implies that the variance will approach the same steady state value $P_{x\infty}$. The same results can be extended to the linear-quadratic-regulator (LQR) problem, for more details see Appendix B.

For constant $\omega_k$ (thus the CSS is an LTI), the frequency response $G(e^{j\omega})$ of the CSS system is identical to the one of the LTI shifted in the frequency axis by $\omega_k$:

$$G(e^{j\omega}) = C \left( I e^{j\omega} - Ae^{-j\omega_k} \right)^{-1} B e^{-j\omega_k} + D$$
$$= C \left( I e^{j(\omega + \omega_k)} - A \right)^{-1} B + D$$

where $I$ is the $n \times n$ identity matrix. The last term of (21) is the frequency response of (16) at the frequency $\omega + \omega_k$. We note that $G(e^{j\omega})$ is defined with respect to an input with persisting sinusoidal amplitude $\beta_k = e^{j\omega_k}$, namely an amplitude modulated signal with carrier frequency $\omega_k$.

### 3.3 Modeling resonance shifts

The fact that the frequency response of the LTI is shifted by the excitation $\omega_k$, motivates the introduction of a complex-valued frequency disturbance $h_k$ which acts independently to change the resonance frequency of the system. The parameter $h_k$ lump in the time-varying aspect of the system into a single variable. This extension incorporates resonance shifts (and changes in the system damping), in contrast to the CSS where the resonance is determined by the system matrices alone. The system can be now written as

$$z_{k+1} = h_k (Az_k + Bs_k + w_k) e^{-j\omega_k}$$
$$q_{k+1} = Cz_k + Ds_k + v_k.$$ 

We restrict $h_k$ in a compact space such that the resulting model set of all possible plants $M$ is stable, $h_k \in \mathcal{D}$ such that $\mathcal{D} = \{h_k \mid h_k \in \mathbb{C}, \|h_k\lambda_i\| \leq d_m, i = 1, 2, \ldots, n\}$, where $d_m < 1$ is the maximum eigenvalue magnitude.

For constant $h_k$, every model in $M$ is globally exponentially stable, as it can be derived from a stable LTI model with state matrix $\|h_k\|A$ excited at the offset frequency $\omega_k - \arg(h_k)$. Moreover, the controllability and the observability of every model in $M$ is identical to the original LTI, since the eigenbasis of $A$ is not affected by the multiplication by $\|h_k\|$. Regarding the Kalman filter equations, the measurement step remains unchanged, but the variance update of the prediction step (17) has to include the scaling term $\|h_k\|^2$.

We note that multiplication of all the states by a single frequency disturbance parameter may oversimplifies the resonance shift dynamics of the real system. For system with a single resonance, this approach can provide accurate enough modeling, whereas for systems with multiple resonances it may be more sensible to multiply different set of states by separate parameters. In the following, we derive the resonance tracking algorithm using a single $h_k$ for the whole
state vector as in [22, 23]. Extending the algorithm to accommodate multiple frequency disturbances corresponding to different states is straightforward; we provide an example including both \( h_k \) and its conjugate at the numerical simulations section.

## 4 Resonance tracking control

The resonance tracking algorithm is developed in the present section. Following the common practice, the design procedure is split into two parts: the estimation of the state and the resonance shift, and the update of the excitation frequency.

### 4.1 Estimating the frequency shifts

The frequency shift estimation can be cast as an optimization problem; we seek the value of \( h_k \) that best matches the observed output data. We define the estimated value for the frequency disturbance \( \hat{h}_k = \arg\min J_{\text{est}}, \) given as the minimizing variable of the cumulative estimation error

\[
J_{\text{est},k} = \frac{1}{2} \sum_{i=1}^{k} ||q_i - \hat{q}_i||^2
\]

where \( q_k \) is the measured output and \( \hat{q}_k \) is the one predicted by the model. Model identification of dynamical systems based on parameterized estimators has been thoroughly discussed previously [37, 38]; the identification technique is termed prediction error method (PEM). For the CSS system, a suboptimal estimator can be realized as a steady state Kalman filter, formulated as an one-step predictor [39]:

\[
\begin{align*}
\hat{z}_{k+1} &= \hat{h}_k [(A - LC)\hat{z}_k + (B - LD)s_k + Lq_k] e^{-j\omega_k} \\
\hat{q}_k &= C\hat{z}_k + Ds_k
\end{align*}
\]

initialized with the state estimate \( z_0 \). The steady state Kalman gain of the CSS system is computed as

\[
L_\omega = e^{-j\omega_k} A P_\infty C^T (C P_\infty C^T + R)^{-1} = e^{-j\omega_k} L
\]

where \( L \) is the gain of the LTI system. We note that \( \hat{z}_k \) is an unbiased state estimate if the matrix \( \hat{h}_k (A - LC) \) has its eigenvalues strictly inside the unit circle. Therefore, we restrict \( \hat{h}_k \in \hat{D} = \{ \hat{h}_k \mid \hat{h}_k \in \mathbb{C}, ||\hat{h}_k \lambda_i|| \leq d_m, ||\hat{h}_k \lambda_{\text{est},i}|| \leq d_m, i = 1, 2, \ldots, n \} \), where \( \lambda_{\text{est},i} \) are the eigenvalues of \( (A - LC) \).

Given an exponentially stable \( \mathcal{M} \) and an one-step-ahead predictor that is stable and twice differentiable with respect to \( \hat{h}_k \) in \( \hat{D} \), the PEM estimate converges to the local minimum or a boundary point of \( \hat{D} \) as \( k \rightarrow \infty \) [38] theorem 1. In our case, the exponential stability of \( \mathcal{M} \) and the one step predictor has been established. Differentiability of (25) is required for the minimization of \( J_{\text{est},k} \). Although \( J_{\text{est},k} \) is not holomorphic with respect to \( \hat{h}_k \), there exists a second order expansion that can form a Gauss-Newton gradient descent minimization method. As optimization problems that involve real-valued function of complex arguments have become more common, a mathematical framework to address them has been developed under the name of \( \mathbb{C}\mathbb{R} \)-calculus [40, 41]. To apply the theory to our problem, we rewrite (24) as:

\[
J_{\text{est},k} = \frac{1}{2} \sum_{i=1}^{k} ||q_i - Ds_i - C(A\hat{z}_{i-1} + Bs_{i-1}) e^{-j\omega_i} h_k ||^2 = \frac{1}{2} [q_m - g(h_k)]^H [q_m - g(h_k)]
\]

where \( q_m \in \mathbb{C}^k \) is a vector with \( j^\text{th} \) component \( q_{m,i} = q_i - Ds_i \). The function \( g(h_k) : \mathbb{C} \rightarrow \mathbb{C}^k \) returns a vector where each element is \( q_i(h_k) = C\hat{z}_i \) and \( \hat{z}_i \) is produced recursively from (25) for a given value \( h_k \). We remark that \( g(h_k) \) is holomorphic and the derivative with respect to \( h_k \) can be calculated as \( \partial g(h_k)/\partial h_k = C\hat{z}_h,i \), where the \( \hat{z}_h,i \) is given recursively from

\[
\hat{z}_{h,i+1} = [(A - LC)\hat{z}_i + (B - LD)s_i + Lq_i] e^{-j\omega_i} + h_k (A - LC)\hat{z}_{h,i} e^{-j\omega_i}
\]

with initial value \( \hat{z}_{h,0} = 0 \). The step of the Gauss-Newton update at \( \hat{h}_k \) for minimizing \( J_{\text{est},k} \) is written as [40]:

\[
\delta h_k = \mathcal{H}^{-1} \left( \frac{\partial g(h_k)}{\partial h_k} \right)^H (q_m - g(h_k))
\]

\[
\mathcal{H} = \left( \frac{\partial g(h_k)}{\partial h_k} \right)^H \left( \frac{\partial g(h_k)}{\partial h_k} \right)
\]
A second approach is to reformulate the frequency update as an optimization problem, based on the following fact:

Although the convergence criterion for PEM assumes a sequence \( \gamma \) where

\[
H = \text{an approximation of the Hessian matrix.}
\]

Further, we formulate the recursive version of PEM. The update at each time step \( k \) is given as:

\[
\hat{q}_k = C\hat{z}_k + Ds_k
\]

\[
\hat{h}_{k+1} = \hat{h}_k + \gamma_0 \hat{S}_k^{-1}(C\hat{z}_{h,k})^H(q_k - \hat{q}_k)
\]

\[
\hat{S}_{k+1} = \hat{S}_k + \gamma_0 [C(\hat{z}_{h,k})^H(C\hat{z}_{h,k}) - \hat{S}_k + \mu_k]
\]

\[
\hat{z}_{k+1} = \hat{h}_{k+1} [(A - LC)\hat{z}_k + (B - LD)s_k + Lq_k] e^{-j\omega_k}
\]

\[
\hat{z}_{h,k+1} = \frac{\hat{z}_{k+1} - \hat{h}_{k+1}(A - LC)\hat{z}_{h,k}}{\hat{h}_{k+1}} + \hat{h}_{k+1}(A - LC)\hat{z}_{h,k} e^{-j\omega_k}
\]

where \( \hat{S}_k \) approximates \( H \) and \( \mu_k \geq 0 \) introduces damping to the iterative procedure. The gain \( \gamma_0 \) is a sequence of positive scalars tending to zero, which weighs the information contained in the current observation in relation to past observations. The recursive PEM is initialized with state \( \hat{z}_0 = z_0 \), frequency disturbance \( \hat{h}_0 = h_0 \) and \( \hat{S}_0 = S_0 \). The derivative \( \hat{z}_{h,0} \) is initialized to zero. Finally, the projection method of \( \hat{h}_k \) into \( \hat{D} \) is used

\[
\hat{h}_{k+1} = \begin{cases} 
\hat{h}_{k+1}, & \hat{h}_{k+1} \in \hat{D} \\
\hat{h}_k, & \hat{h}_{k+1} \notin \hat{D}.
\end{cases}
\]

The recursive method has the same convergence properties as minimizing (24), namely converge to a local minimum of \( J_{\text{est}} \) or the boundary of \( \hat{D} \) [38] theorem 2, resulting in the best approximation of the true system available in \(\mathcal{M} \). Although the convergence criterion for PEM assumes a sequence \( \gamma_0 \) that tends to zero asymptotically, a constant value can be used if the system changes gradually [42]. For systems with sudden changes in parameters, a variable \( \gamma_0 \) scheme can be applied [43]. The selection of \( \gamma_0 \) and \( L \) affects the settling time and the noise suppression of the algorithm, through the scaling of the gradient direction. Therefore, designing \( L \) based on the actual noise characteristics \( Q \) and \( R \) may not achieve the desired performance.

The same error minimization setup (24) can be used as the basis for a moving horizon estimation (MHE) algorithm [44], which in the case of nonlinear systems has superior performance for the cost of higher computational complexity. In MHE, the minimization is performed on a truncated series of \( J_{\text{est},k} \), compiled from a given number of past measurements. The length of the estimation horizon is denoted by \( N_h \in \mathbb{N}_{>0} \). The MHE update can be summarized as:

\[
\hat{h}_k = \arg\min_i \frac{1}{2} \sum_{i=k-N_h+1}^k ||q_i - \hat{q}_i||^2
\]

\[
\hat{z}_{k-N_h+1} = \hat{h}_k [(A - LC)\hat{z}_{k-N_h} + (B - LD)s_{k-N_h} + Lq_{k-N_h}] e^{-j\omega_{k-N_h}}.
\]

Initialization with some state estimate \( \hat{z}_0 = z_0 \) is necessary for the first step. The minimization problem (38) can be solved numerically using [40]. Although the MHE method are able to include inequality constraints naturally, we found that restriction of the estimate in \( \hat{D} \) is not necessary. As the plant estimator becomes unstable, the prediction error increases disproportionately and thus the constrains are enforced in a similar fashion as in the constrained optimization penalty method [45]. During the numerical experiments, allowing the algorithm to converge slightly outside the boundary of \( \hat{D} \), showed faster convergence.

### 4.2 Update the excitation frequency

Given the output of the estimation step, we discuss schemes that track the resonance frequency of the system. A straightforward approach is to use \( \hat{h}_k \) to correct the excitation frequency for the next step

\[
\omega_{k+1} = \arg(\lambda_s) + \arg(\hat{h}_k) = \omega_s + \arg(\hat{h}_k).
\]

Since the system cannot be destabilized for any sequence of \( \omega_k \), the closed loop will be stable. Therefore convergence of \( \hat{h}_k \) to the correct value is main issue, which is expected for an adequately large number of observations and if the sequence of \( \omega_k \) is informative enough about the system. We refer to this control update as the direct substitution.

A second approach is to reformulate the frequency update as an optimization problem, based on the following fact: consider the left eigenvector \( \tilde{\chi}_s \) of \( \tilde{A} \) corresponding to \( \lambda_s \) and any state estimate \( \hat{z}_k \), then

\[
\tilde{\chi}_s H (\tilde{A} \hat{z}_k - \hat{\lambda}_s \hat{z}_k) = 0.
\]
A closer look at (42) reveals that the current value of \( \hat{\lambda}_k \), where \( \hat{\lambda}_k, \hat{\chi}_k = \text{argmin} J_{c,k} \) and

\[
J_{c,k} = \frac{1}{2} \sum_{i=0}^{k-1} \| \hat{\chi}^H_k (\hat{z}_{A,i} - \hat{\lambda}_k \hat{z}_k) \|^2.
\]

The vector \( \hat{z}_{A,i} = \hat{z}_{i+1} e^{j\omega_i} - \hat{h}_i B s_i \) is an estimate of \( \hat{A} \hat{z}_i \), since the latter is not readily available.

We are interested in solving the minimization problem recursively by applying the same principles as in the previous section. As a first step, we calculate the Gauss-Newton approximation of the Hessian as in (31),

\[
\begin{bmatrix}
\hat{z}^H_k \hat{\chi} \hat{\chi}^H_k \hat{z}_k \\
-\hat{\chi}^H_k \hat{z}_k - \hat{\lambda}_k \hat{z}_k \\
(\hat{z}_{A,k} - \hat{\lambda}_k \hat{z}_k)^T
\end{bmatrix}
\]

which in this case is singular. To generate and invertible matrix, and we substitute the projection part of the matrix \( (\hat{z}_{A,k} - \hat{\lambda}_k \hat{z}_k)^T \) by \( I \| \hat{z}_{A,k} - \hat{\lambda}_k \hat{z}_k \|^2 \), since it acts only as a scaling operator for vectors parallel to the direction of projection. The resulting diagonal hessian approximation \( \hat{\mathcal{H}}_c = \text{diag} \left( I \| \hat{z}_{A,k} - \hat{\lambda}_k \hat{z}_k \|^2 \right) \) decouples the update of \( \hat{\lambda}_k \) from \( \hat{\chi}_k \), and introduces a common scaling for the gradient of all the elements of the latter. Therefore, we can form the recursive minimization algorithm as

\[
\hat{\lambda}_{k+1} = \hat{\lambda}_k + \gamma \lambda S_{\hat{\chi} k}^{-1} \hat{z}_k \hat{\chi}_k^H (\hat{z}_{A,k} - \hat{\lambda}_k \hat{z}_k)
\]

\[
\hat{\chi}_{k+1} = \hat{\chi}_k - \gamma \lambda S_{\hat{\chi} k}^{-1} \hat{z}_k \hat{\chi}_k^H (\hat{z}_{A,k} - \hat{\lambda}_k \hat{z}_k)^T
\]

\[
\hat{\chi}_{\lambda,k+1} = \hat{\chi}_{\lambda,k} + \gamma \lambda (\hat{z}_k \hat{\chi}_k^H \hat{z}_k - \hat{\chi}_{\lambda,k} + \mu \lambda)
\]

\[
\hat{\chi}_{\chi,k+1} = \hat{\chi}_{\chi,k} + \gamma \chi \| \hat{z}_{A,k} - \hat{\lambda}_k \hat{z}_k \|^2 - \hat{\chi}_{\chi,k} + \mu \chi.
\]

The parameters \( \gamma \lambda \) and \( \gamma \chi \) scale the step size, and \( \hat{\chi}_{\lambda,k} \) and \( \hat{\chi}_{\chi,k} \) approximate the elements of \( \hat{\mathcal{H}}_c \). The damping factor \( \mu \lambda \) and \( \mu \chi \) are added to improve the convergence of the algorithm. We initialize \( \hat{\lambda}_0 = \lambda_0 \) and \( \hat{\chi}_0 = \chi_0 \) with the eigenvalue and the eigenvector of \( \hat{A} \), \( \hat{\chi}_{\lambda,0} = S_{\chi,0} \) and \( \hat{\chi}_{\chi,0} = S_{\chi,0} \). To ensure numerically stability, we normalize \( \| \hat{\chi}_k \| = 1 \) at the end of each time-step.

A closer look at [42] reveals that the current value of \( \hat{\lambda}_k \) is updated by its difference from a generalized Rayleigh quotient of \( \hat{X}_k \hat{A} \), where \( \hat{X}_k = \hat{\chi}_k \hat{\chi}_k^H \) is the projection matrix on the direction of \( \hat{\chi}_k \). The Rayleigh quotient is known to be a good approximation of the eigenvalue of a matrix [46], as briefly revisited in the following. The projection can be written as \( \hat{z}_k \hat{X}_k = \kappa \hat{\chi}_k^H \), where \( \kappa \) is a constant. Assuming that indeed \( \hat{\chi}_k \) is an eigenvector of \( \hat{A} \), then \( \hat{z}_k \hat{X}_k \hat{A} = \kappa \hat{\chi}_k \hat{\chi}_k^H \). By multiplying both sides by \( \hat{z}_k \) and solving for \( \hat{\lambda}_k \), we recover the last part of (42). Certainly, the assumption that \( \hat{\chi}_k \) is the actual eigenvector is quite strong (although it is the best estimate at the instance \( k \)), but demonstrates the reasoning behind the \( \hat{\lambda}_k \) update. On the other hand, \( \hat{\chi}_k \) is also corrected at each time step, with [43] bearing notable resemblance to the Gram-Schmidt orthogonalization process.

To distinguish this frequency update from the direct substitution, we refer to this approach as the Rayleigh quotient update. The Rayleigh quotient update has an additional interesting application; if the state of the system can be directly measured, it alleviates the need for computing \( \hat{h}_k \). Indeed, for the numerical experiments with the Rayleigh quotient scheme, we assume \( \hat{h}_k = 1 \) even if an estimate of \( \hat{h}_k \) is available, so as to test its effectiveness.

5 Implementation and numerical simulations

5.1 Estimating the analytic representation

The main issue related to the implementation of the proposed frequency tracking schemes is the extraction of the complex envelope \( q_k \) out of the measured output of the plant. This calculation is necessary, because the plant and the one-step predictor [35] are excited at different frequencies, in particular at \( \omega_k \) and at \( \omega_k - \arg(\hat{h}_k) \) respectively. The representation of the plant in the CSS form neatly conceals this issue, which will arise if the estimation algorithm [32, 36] is converted to the time domain.

A possible approach to convert the signal in its analytic representation is the application of the sliding discrete Fourier transform (sDFT) or the sliding Goertzel algorithm [47]. The sDFT is equivalent to the discrete Fourier transform (DFT) applied on a window of length \( N_T \in \mathbb{N}_{>0} \), but it has an output rate equal to the input signal rate and it computed only at a specified center frequency. Still, we need to estimate \( q_k \) at non-integer multiples of \( T_s^{-1} \). In addition, \( \omega_k \) may
change over the computation window, therefore one should consider the non-uniform discrete time Fourier transform [48]. In order to address these issues, we extended the sDFT algorithm to the sliding non-uniform discrete time Fourier transform (sNDTFT) version. Following the derivation of sDFT in [47], the filter formulation is

\[
\hat{Y}_k = \hat{Y}_{k-1} e^{j\omega_k} - y_{k-N_f} e^{j\delta\omega_k} + y_k
\]

(46)

\[
\delta\omega_k = \delta\omega_{k-1} + \omega_k - \omega_{k-N_f}
\]

(47)

where \(\hat{Y}_k\) is the state of the sNDTFT filter and \(\omega_k = 0, y_k = 0\) for \(k < 0\). The complex envelope \(q_k\) is calculated as

\[
q_k = 2 \frac{\hat{Y}_k}{N_f} e^{-j\theta_k}
\]

(48)

where the last term has a dual role: to apply the phase correction introduced in [49], which accounts for the calculation at a non-integer multiple of \(T_s^{-1}\), and to match the phase \(q_k\) with (3). More details are given in Appendix C.

To mitigate the effect of spectral leakage, a Hann window is applied at the frequency domain; we compute (46) for two adjacent frequencies \(\omega_k \pm \frac{2\pi}{N_f}\) and the results are averaged and subtracted from \(\hat{Y}_k\) before calculating (48). This calculation includes the correction factor of 2 needed to recover the correct signal amplitude. Since the calculations are performed at an offset of \(2\pi/N_f\), \(\delta\omega_k\) remains the same for the adjacent frequencies; only for the first \(N_f\) samples will be a mismatch. We point out that the sNDTFT will produce an approximation of the complex envelope, which depends on \(N_f\). Additionally, the responsiveness of the algorithm to changes in \(\omega_k\) is also affected by \(N_f\).

5.2 SISO system with non-monotonic phase

Firstly, we simulate the resonance tracking algorithm on the model of a piezoelectric actuator with a resonance and an anti-resonance. The actuator is modeled as a resistor-inductor-capacitor oscillator (\(R_m, L_m, C_m\)) in parallel with a capacitor \(C_0\), as described in [17]. The transfer function \(G_p(s)\) from the voltage to charge in the Laplace domain is given as

\[
G_p(s) = \frac{C_0 s^2 + s \frac{R_m}{L_m} + \frac{C_m + C_0}{L_m C_m C_0}}{s^2 + s \frac{R_m}{L_m} + \frac{1}{L_m C_m}}
\]

(49)

where \(s\) is the Laplace variable. The nominal values for the parameters (\(C_0 = 2\ \text{nF}, R_m = 50\ \Omega, L_m = 0.103\ \text{H}, C_m = 80\ \text{pF}\)) are taken from [17].

The model exhibits non-monotonic input-output phase, as shown in Fig. 1 which is challenging for resonance tracking techniques based solely on phase information. Since the phase in not unique, multiple equilibria arise, which alternate between stable and unstable modes [27][18][28][17]. The points in Fig. 1 mark the two equilibria for phase difference of \(-\pi/2\) and their stability for a positive gain controller. To demonstrate that the proposed resonance tracking algorithm remains unaffected by the non-monotonic nature of the phase, we assume that the model parameters are not fixed and can vary 10% around their nominal values. This way, the resonance estimates, which initially will be close to the resonance of the nominal plant, may lie in the unstable region of the phase-based tracking schemes.
Figure 2: Tracking of the resonance frequency for a set of 100 plants with 10% parametric uncertainty. The trajectory of $\omega_k/\tilde{\omega}_\lambda$ for the nominal plants is shown with the lines. The region bounding the simulated trajectories for the plant set is indicated by the shaded regions. The star on the frequency axis marks the relative position of the antiresonance. a) Simulation results using the recursive PEM with direct substitution. Simulation results using the LTI plant with the sNDTFT algorithm are shown in solid lines and simulations with the plant transformed to the CSS representation are shown in dashed lines. b) Simulation results using the recursive MHE with direct substitution.

For the simulations, we discretized the system with $T_s = 1 \mu s$ and converted to the minimal and balance state space realization. We performed a Monte-Carlo sampling of the model parameters and we created a set of one hundred random plants. We set $s_k = 1$ V and the noise was selected to have power of around 10% of the signal power, specifically $Q_s = 0.01 V^2$ (for estimator design $Q = B^H Q_s B$) and $R = 64 \mu C^2$. We tested both the recursive PEM and the MHE estimation with the direct substitution update. For the recursive PEM, we set $\gamma = 0.0025, \hat{S}_0 = 0.01 \mu C^2, \mu_e = 0.0008 \mu C^2$ and the sNDTFT window $N_f = 24$. For the MHE, the simulation were performed with $N_h = 350$ and $N_f = 32$. To assess the effect of sNDTFT, we also repeated the recursive PEM simulation with the plant model transformed to CSS, where the output $q_k$ is readily available. The trajectories of $\omega_k/\tilde{\omega}_\lambda$ are shown in Fig. 2. The star on the y-axis indicates the mean relative position of the anti-resonance, which also separates the stable from the unstable region for the phase-based resonance tracking techniques. The direct substitution update is able to converge to the resonance of the system, for both estimation algorithms. The sNDTFT algorithm can be satisfactory combined with the tracking algorithms, although it impacts the the behavior closed loop system. The MHE estimation has faster convergence, almost half the one of the RPEM, for cost of higher computation burden.

The performance of the Rayleigh quotient control scheme is shown in Fig. 3. The parameters of the estimators were set as before and $\hat{S}_{\lambda,0} = 2.5, \hat{S}_{\chi,0} = 0.25, \gamma_\lambda = \gamma_\chi = 0.15$ and $\mu_\lambda = \mu_\chi = 0.0025$. The performance of the Rayleigh quotient update scheme is comparable to the one of the direct substitution.
Figure 3: Tracking of the resonance frequency for a set of 100 plants with 10% parametric uncertainty. The trajectory of $\omega_k/\tilde{\omega}_\lambda$ for the nominal plants is shown with the lines. The region bounding the simulated trajectories for the plant set is indicated by the shaded regions. The star on the frequency axis marks the relative position of the antiresonance.

(a) Simulation results using the recursive PEM with the Rayleigh quotient update. Simulation results using the LTI plant with the with sNDTFT algorithm are shown in solid lines and simulations with the plant transformed to the CSS representation are shown in dashed lines. b) Simulation results using the recursive MHE with the Rayleigh quotient update.

5.3 MIMO system of a gyroscope

Next, we looked into the tracking of the resonance frequency a MIMO system, in particular a vibrating structure gyroscope [10, 11]. Briefly, the gyroscope contains two proof masses vibrating in a plane. When the structure is rotated perpendicular to that plane, energy is transferred between the proof masses due to the Coriolis effect. Vibratory gyroscope can be modeled as two 2nd order spring-mass-damper oscillators and coupled by cross damping and spring terms. The cross terms include the Coriolis effect but also parasitic mechanical and electrical coupling. The usual technique for acquiring the rotational speed is to excite one of the oscillators (which is referred as the primary mode) at a constant amplitude while keeping the other (secondary mode) still. As a result, the Coriolis effect acts on the secondary with a force that is proportional to the oscillating amplitude of the primary [10]. To measure the rotational speed accurately though, the parasitic coupling either has to be eliminated or identified and corrected. Here, we propose an alternative principle for acquiring the rotational speed acting on the gyroscope. Both oscillators are excited at the same frequency, which coincides with the resonance of the primary. The Coriolis effect alters the resonance frequency of the system, which in turns allows to determine the rotational speed.

The input to our model is the control force $u_g = [u_{pm}, u_{sm}]^T$ that can be exerted on the primary and the secondary. The subscripts $pm$ and $sm$ indicate variables of the primary and the secondary mode, respectively. Given the displacement of the oscillators $x_g = [x_{pm}, x_{sm}]^T$, which are also the system output, the system dynamics are described by

$$\ddot{x}_g + D_g \dot{x}_g + K_g x_g = u_g - 2\Omega \dot{x}_g$$  \hspace{1cm} (50)
where \( D_g \) and \( K_g \) are the damping and stiffness matrices of the system respectively. The Coriolis effect is captured by

\[
\Omega = \begin{bmatrix} 0 & -\omega_z \\ \omega_z & 0 \end{bmatrix}
\]

(51)

where \( \omega_z \) is the rotation speed to be measured. For our simulations, the model parameters were set to

\[
K_f = \begin{bmatrix} 355.3 & 70.99 \\ 70.99 & 532.9 \end{bmatrix}, \quad D_f = \begin{bmatrix} 0.01 & 0.002 \\ 0.002 & 0.01 \end{bmatrix}
\]

as proposed in [11]. The model is normalized and all the units are dropped in the following. To increase the sensitivity of the sensor and improve its dynamics response, we used state feedback based on the LQR design (see Appendix B). We selected the state and input weights as diagonal matrices \( Q_c = \text{diag}(\{54, 84, 0.0036, 189\}) \) and \( R_c = \text{diag}(\{0.4, 0.49\}) \). As a result, the sensitivity was increased roughly from 0.15 to 0.5 Hz s rad\(^{-1}\). The input to the CSS system is then synthesized as \( s_k = K \hat{z}_k + s_r \), where \( K \) is the state feedback matrix and \( s_r \) is a constant offset.

The system has two resonance frequencies, each one associated with the respective oscillator. Under the effect of \( \omega_z \) they shift towards different directions, one to higher frequencies while the other to lower. To better capture this effect, we do not apply \( h_k \) to all of the system states; we correct the states of the primary mode by \( \tilde{h}_k \) and the states of the secondary by the conjugate \( \bar{h}_k \). Thus, some modifications of the basic algorithm are necessary to accommodate the change. For the calculation of the Gauss-Newton direction in (30), \( h_k \) and \( \tilde{h}_k \) are treated as separate variables [40]. Therefore, we calculate two separate recursion for the state derivatives, \( \tilde{z}_{h, k} \) and \( \tilde{z}_{\bar{h}, k} \), with respect to \( h_k \) and \( \tilde{h}_k \), respectively. The \( \tilde{h}_k \) update becomes

\[
\tilde{h}_{k+1} = \tilde{h}_k + \gamma_k (S_k^{-1} + \tilde{S}_{c,k}^{-1}) \left[ (C\tilde{z}_{h,k})^H(q_k - \tilde{q}_k) + (C\tilde{z}_{\bar{h},k})^H(q_k - \bar{q}_k) \right]
\]

(52)
where $\hat{S}_{c,k}$ is approximation of the Gauss-Newton direction for $\overline{h_k}$.

For the simulations we used $s_r = [10 \quad 100]^T$, to achieve comparable signal magnitudes for both oscillators since the primary is in resonance. The noise levels were set to $Q = \text{diag}\left(\begin{bmatrix} 0.04 & 0.04 \end{bmatrix}\right)$ and $R = \text{diag}\left(\begin{bmatrix} 1.6 & 1.6 \times 10^{-5} \end{bmatrix}\right)$. The Kalman gain was designed with different noise characteristics than the ones used in the simulation, in particular with $Q = 1500I$ and $R = 0.1I$. The higher relative scaling of $Q$ partially accounts for the effect of $\omega_z$ as a disturbance input. The estimation parameters were set to $\gamma = 0.013$, $S_0 = \hat{S}_{c,0} = 1000$ and $\mu_e = 0$ for the recursive PEM and $N_h = 185$ for the MHE simulations. In all cases, we set $N_f = 32$. For the Rayleigh quotient update, we set $\hat{S}_{\lambda,0} = \hat{S}_{\chi,0} = 1000$, $\gamma_\lambda = \gamma_\chi = 0.02$ and $\mu_\lambda = \mu_\chi = 0$.

We simulated the response of the tracking algorithms assuming step and ramp changes in $\omega_z$. The results are shown in Fig. 4. The update with Rayleigh quotient was able to follow the theoretical values of $\tilde{\omega}_s$ closely, whereas the direct substitution showed a slight offset. In general, the performance of the algorithm with RPEM and the Rayleigh quotient update is satisfactory, even for systems with fast changing parameters, confirming the effectiveness of the proposed schemes.

6 Conclusion

In this work, we describe the model-based resonance frequency tracking algorithm for linear time-varying systems. We introduced a state transformation of linear systems into a complex-value representation and we lumped the time-varying aspect of the system into a single variable. This transformation allows to solve the resonance tracking problem by recasting it as a parameter identification and an optimization task. As a result, the proposed algorithm demonstrated sufficient performance even in cases where commonly used techniques may fail, like in non-monotonic phase systems, or are not readily applicable, like in MIMO systems. We confirmed our claims using numerical simulations and we discussed implementation issues.

Although we discussed the tracking of a single frequency of the system, modifying our algorithm to track multiple resonances is straightforward. Moreover, the circularity of the complex noise and disturbance variables can be relaxed, extending the application of the tracking scheme to systems with widely linear complex random variables.

Acknowledgements

The author would like to thank Dario Izzo (ESA, Advanced Concepts Team) for his valuable input and comments.

References

[1] P Hauptmann. Resonant sensors and applications. Sensors and Actuators A: Physical, 26(1-3):371–377, 1991.
[2] Anja Boisen, Søren Dohn, Stephan Sylvest Keller, Silvan Schmid, and Maria Tenje. Cantilever-like micromechanical sensors. Reports on Progress in Physics, 74(3):036101, 2011.
[3] Frank Claeyssen, Nicolas Lhermet, and T Maillard. Magnetostriuctive actuators compared to piezoelectric actuators. In European Workshop on Smart Structures in Engineering and Technology, volume 4763, pages 194–201. International Society for Optics and Photonics, 2003.
[4] C Gökcek. Resonance seeking control. In Advanced Intelligent Mechatronics. Proceedings, 2005 IEEE/ASME International Conference on, pages 1348–1353. IEEE, 2005.
[5] Tom Larsen, Silvan Schmid, L Grönberg, AO Niskanen, J Hassel, Søren Dohn, and Anja Boisen. Ultrasensitive string-based temperature sensors. Applied Physics Letters, 98(12):121901, 2011.
[6] M Aikele, K Bauer, W Ficker, F Neubauer, U Prechtl, J Schalk, and H Seidel. Resonant accelerometer with self-test. Sensors and Actuators A: Physical, 92(1):161–167, 2001.
[7] Tobias Brack, Dominik Kern, Mengdi Chen, and Jürg Dual. Dynamics and stability of phase controlled oscillators. Journal of Dynamic Systems, Measurement, and Control, 138(7):071007, 2016.
[8] Lei Sheng, Chen Dajing, and Chen Yuquan. A surface acoustic wave humidity sensor with high sensitivity based on electrospun MWCNT/Nafion nanofiber films. Nanotechnology, 22(26):265504, 2011.
[9] Heron EL Ávila, Fernando R Sousa, and Daniel J Pagano. Resonant cavity water cut meter with automatic resonance tracking system. In Instrumentation and Measurement Technology Conference (I2MTC), 2017 IEEE International, pages 1–6. IEEE, 2017.
[10] Johan Raman, Edmon Cretu, Pieter Rombouts, and Ludo Weyten. A closed-loop digitally controlled MEMS gyroscope with unconstrained sigma-delta force-feedback. *IEEE Sensors journal*, 9(3):297–305, 2009.

[11] Juntao Fei, Weifeng Yan, and Yuzheng Yang. Adaptive nonsingular terminal sliding mode control of MEMS gyroscope based on backstepping design. *International Journal of Adaptive Control and Signal Processing*, 29(9):1099–1115, 2015.

[12] Mark Stephen Bentley, Herman Arends, Bart Butler, Jose Gavira, Harald Jeszenszky, Thurid Mannel, Jens Romstedt, Roland Schmied, and Klaus Torkar. MIDAS: lessons learned from the first spaceborne atomic force microscope. *Acta Astronautica*, 125:11–21, 2016.

[13] Hong Li and Zhiyuan Jiang. On automatic resonant frequency tracking in LLC series resonant converter based on zero-current duration time of secondary diode. *IEEE Transactions on Power Electronics*, 31(7):4956–4962, 2016.

[14] Amit K Singh, Pritam Das, and SK Panda. High voltage high frequency resonant DC-DC converter for electric propulsion for micro and nanosatellites. In *Telecommunications Energy Conference (INTELEC), 2014 IEEE 36th International*, pages 1–5. IEEE, 2014.

[15] Roman Bosshard, Johann W Kolar, and Bernhard Wunsch. Control method for inductive power transfer with high partial-load efficiency and resonance tracking. In *Power Electronics Conference (IPEC-Hiroshima 2014-ECCE-ASIA), 2014 International*, pages 2167–2174. IEEE, 2014.

[16] Sungjin Choi, Myoungghan Ryu, Sangmin Lee, and Bo H Cho. Adaptive frequency control strategy for piezoelectric transformer in AC/DC adapter applications using phase detector. In *Applied Power Electronics Conference and Exposition, 2006. APEC ’06. Twenty-First Annual IEEE*, pages 4–pp. IEEE, 2006.

[17] Xiaotian Sun, Roberto Horowitz, and K Komvopoulos. Stability and resolution analysis of a phase-locked loop natural frequency tracking system for MEMS fatigue testing. *Journal of Dynamic Systems, Measurement, and Control*, 124(4):599–605, 2002.

[18] SH Albrecht, Pr Grütter, D Horne, and D Rugar. Frequency modulation detection using high-Q cantilevers for enhanced force microscope sensitivity. *Journal of Applied Physics*, 69(2):668–673, 1991.

[19] Sevil Zeynep Lulec, Ulas Adiyani, Goksen G Yaralioglu, Yusuf Leblebici, and Hakan Urey. MEMS cantilever sensor array oscillators: Theory and experiments. *Sensors and Actuators A: Physical*, 237:147–154, 2016.

[20] Sergi Gorreta, Joan Pons-Nin, Gema López, Eduard Figueras, Roger Jové-Casulleras, Carles Araguz, Pol Via, Adriano Camps, and Manuel Domínguez-Pumar. A CubeSAT payload for in-situ monitoring of pentacene degradation due to atomic oxygen etching in LEO. *Acta Astronautica*, 126:456–462, 2016.

[21] Bjoern Hammarstrom, Mikael Evander, Jacob Wahlstrom, and Johan Nilsson. Frequency tracking in acoustic trapping for improved performance stability and system surveillance. *Lab on a Chip*, 14(5):1005–1013, 2014.

[22] TR Albrecht, Pr Grütter, D Horne, and D Rugar. Frequency modulation detection using high-Q cantilevers for enhanced force microscope sensitivity. *Journal of Applied Physics*, 69(2):668–673, 1991.

[23] Sevil Zeynep Lulec, Ulas Adiyani, Goksen G Yaralioglu, Yusuf Leblebici, and Hakan Urey. MEMS cantilever sensor array oscillators: Theory and experiments. *Sensors and Actuators A: Physical*, 237:147–154, 2016.

[24] Sergi Gorreta, Joan Pons-Nin, Gema López, Eduard Figueras, Roger Jové-Casulleras, Carles Araguz, Pol Via, Adriano Camps, and Manuel Domínguez-Pumar. A CubeSAT payload for in-situ monitoring of pentacene degradation due to atomic oxygen etching in LEO. *Acta Astronautica*, 126:456–462, 2016.

[25] Sung Park, Chin-Woo Tan, Haedong Kim, and Sung Kyung Hong. Oscillation control algorithms for resonant sensors with applications to vibratory gyroscopes. *Sensors*, 9(8):5952–5967, 2009.

[26] Hossein Sohanian-Haghighi and Amir Hossein Davaiya-Markazi. Resonance tracking of nonlinear MEMS resonators. *IEEE/ASME transactions on Mechatronics*, 17(4):617–621, 2012.

[27] Shigeo Hayashi. On the tracking of resonance and antiresonance of a piezoelectric resonator. ii. accurate models of the phase locked loop. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 39(6):787–790, 1992.

[28] Tobias Brack, Robin Vujanic, and Jurg Dual. Simultaneous phase control of multiple frequencies of multi-degree-of-freedom systems. *Journal of Vibration and Control*, page 1077546316644366, 2017.

[29] Brian J Rodriguez, Clint Callahan, Sergei V Kalinin, and Roger Proksch. Dual-frequency resonance-tracking atomic force microscopy. *Nanotechnology*, 18(47):475504, 2007.

[30] Jin Woo Song, Hyeong-Min Song, Young Jae Lee, Chan Gook Park, and Sangkyung Sung. Design of oscillation control loop with coarse-precision mode transition for solid-state resonant gyroscope. *IEEE Sensors Journal*, 16(6):1730–1742, 2016.
Appendices

Appendix A  Continuous time CSS transformation

Consider the continuous-time LTI, in the state-space representation

\[
\dot{x}(t) = Ax(t) + Bu(t) + \tilde{w}(t) \tag{53}
\]
\[
y(t) = Cx(t) + Du(t) + \tilde{v}(t) \tag{54}
\]
where \( \dot{x}(t) \) is the time derivative of \( x(t) \) with respect to the time \( t \). We assume that \( u(t) = \beta(t) \odot \exp(j \omega(t)t + j \psi(t)) \) and \( x(t) = \alpha(t) \odot \exp(j \omega(t)t + j \phi(t)) \). In the following, we drop the notation \( x(t) \) in favor of \( x \) for convenience.
introduce the complex envelope variables \( z = \alpha \exp(j\phi) \) and \( s = \beta \exp(j\psi) \). We compute the time derivative of the state variables

\[
\dot{z} = \dot{\alpha}e^{j\phi} + j\alpha \dot{\phi}e^{j\phi} \quad (55)
\]

\[
\dot{s} = \dot{\alpha}e^{j(\omega t + \phi)} + j\alpha (\omega + \dot{\omega} t + \dot{\phi})e^{j(\omega t + \phi)} = \dot{z}e^{j\omega t} + j(\omega + \dot{\omega} t)ze^{j\omega t}. \quad (56)
\]

Substitution into (53-54) and elimination of the \( e^{j\omega t} \) terms produces the continuous-time CSS representation

\[
\dot{z} = (A - j(\omega + \dot{\omega})I)z + Bs + w \quad (57)
\]

\[
q = Cz + Ds + v \quad (58)
\]

where \( q \) is the complex envelope of \( y \). The noise is also transformed in its complex representation, similarly to the discrete-time case.

The similarities between the LTI and the CSS model that were discussed in the discrete-time case also apply for the continuous-time models. We note though that the zero-order hold discretization of (57-58) results in the discrete-time CSS (10-11). For constant \( \omega \) on the time interval of length \( T \), the matrix exponential of \( A - j\omega I \) is

\[
e^{(A-j\omega I)T} = e^{-j\omega T}e^{AT} \quad (59)
\]

since the matrices \( A \) and \( j\omega I \) commute. The discretization of the complex matrix term \( j\omega I \) results in multiplication by \( e^{-j\omega T} \) seen in the discrete-time CSS.

**Appendix B  Optimal control for CSS**

Consider the optimal control problem, with state update (10) and the quadratic cost function

\[
J_c = E \left[ z_N^H Q_{c,N_c} z_{N_c} + \sum_{i=0}^{N_c-1} z_i^H Q_{c} z_i + s_i^H R_c s_i \right] \quad (60)
\]

where \( Q_{c,N_c}, Q_c \) and \( R_c \) are real positive definite matrices of appropriate dimensions, which penalize the terminal cost, the state and the control input respectively. The optimal cost trajectory can be computed by applying the dynamic programming algorithm [50], starting for the final cost

\[
J_{c,N_c}^* (z_{N_c}) = z_{N_c}^H Q_{c,N_c} z_{N_c} \quad (61)
\]

where \( J_{c,k}^* (z_k) \) denotes the optimal cost at time \( k \) from \( z_k \). Similarly at time \( N_c - 1 \), the optimal cost is given as

\[
J_{c,N_c-1}^* (z_{N_c-1}) = \min_{s_{N_c-1}} E \left[ z_{N_c-1}^H Q_{c} z_{N_c-1} + s_{N_c-1}^H R_c s_{N_c-1} + J_{c,N_c}^* ((Az_{N_c-1} + Bs_{N_c-1} + w_{N_c-1})e^{-j\omega k}) \right] \quad (62)
\]

By differentiating with respect to \( s_{N_c-1} \) and setting the derivative to zero, we recover the optimal input

\[
s_{N_c-1}^* = -(R_c + B^H Q_{c,N_c} B)^{-1} B^H Q_{c,N_c} Az_{N_c-1}. \quad (63)
\]

Substitution of the optimal input into (62) results in

\[
J_{c,N_c-1}^* (z_{N_c-1}) = z_{N_c-1}^H V_{N_c-1} z_{N_c-1} + E \left[ w_{N_c-1}^H Q_{c,N_c} w_{N_c-1} \right] \quad (64)
\]

where the optimal cost is quadratic with respect to the current state \( z_{N_c-1} \). The symmetric matrix \( V_{N_c-1} \) is equal to

\[
V_{N_c-1} = A^H Q_{c,N_c} B(R_c + B^H Q_{c,N_c} B)^{-1} B^H Q_{c,N_c} A + A^H Q_{c,N_c} A + Q_c. \quad (65)
\]

Recursive application of the dynamic programming algorithm results in a quadratic representation of the optimal cost. The weight matrix of the cost is given by the recursion

\[
V_{k-1} = A^H V_k B(R_c + B^H V_k B)^{-1} B^H V_k A + A^H V_k A + Q_c \quad (66)
\]

with terminal value \( V_{N_c} = Q_{c,N_c} \). Most importantly, the weighting matrix for the LTI model follows the same recursion; for equal terminal costs the trajectories for the LTI and the CSS are identical. As a result, the steady state cost matrices for both systems are equal.
Appendix C  Sliding non-uniform discrete time Fourier transform

The conversion of measured real-valued signal into the complex envelope representation can be accomplished by applying the non-uniform discrete time Fourier transform (NDTFT). The non-uniformity comes from the fact that the instantaneous frequency of the signal may not be constant (although known). Thus, the complex envelope \( q_k \) can be approximated by the \( N_f \) length NDTFT \( Y_k \) of the measured signal \( y_k \), as

\[
Y_k = y_{k-N_f+1} + \sum_{i=2}^{N_f} y_{k-N_f+i} \exp(-j \sum_{l=2}^{i} \omega_{k-N_f+l}) \\
= \sum_{i=0}^{N_f-1} y_{k-i} \exp(-j \delta \theta_k(i,N_f-1))
\]  

(67)

where \( \delta \theta_k(i,l) = \theta_k(i+1) - \theta_k(i-1) \) is the phase difference between the samples. Following [49, 47], we derive a recursive way to compute (67). First, we multiply both sides of (67) by \( e^{-j \delta \theta_k(0,N_f-1)} \),

\[
Y_k = e^{-j \delta \theta_k(0,N_f-1)} \sum_{i=0}^{N_f-1} y_{k-i} e^{j \delta \theta_k(i,0)} = e^{-j \delta \theta_k(0,N_f-1)} \tilde{Y}_k.
\]  

(68)

The second term can be computed recursively as

\[
\tilde{Y}_k = \tilde{Y}_{k-1} e^{j \omega_k} - y_{k-N_f} e^{j \delta \theta_k(0,N_f)} + y_k.
\]  

(69)

The value of \( \delta \theta_k(0,N_f) \) can be update at each time-step as in (47). A phase and magnitude correction has to be applied to \( Y_k \) so as to recover \( q_k \). The previous calculation of the NDTFT assumes zero phase at the start of the computation window, so we have to offset the calculation by \( \theta_k-N_f+1 \) to be consistent when comparing phase shifts to \( \theta_k \),

\[
q_k = \frac{2}{N_f} Y_k e^{j \theta_k-N_f+1} = \frac{2}{N_f} \tilde{Y}_k e^{j \theta_k}.
\]  

(70)