Generalizations of the Complex-Step Derivative Approximation

Kok-Lam Lai [*] and John L. Crassidis[†]

University at Buffalo, State University of New York, Amherst, NY 14260-4400

This paper presents a general framework for the complex-step derivative approximation to compute numerical derivatives. For first derivatives the complex-step approach does not suffer substraction cancellation errors as in standard numerical finite-difference approaches. Therefore, since an arbitrarily small step-size can be chosen, the complex-step method can achieve near analytical accuracy. However, for second derivatives straight implementation of the complex-step approach does suffer from roundoff errors. Therefore, an arbitrarily small step-size cannot be chosen. In this paper we expand upon the standard complex-step approach to provide a wider range of accuracy for both the first and second derivative approximations. Higher accuracy formulations can be obtained by repetitively applying the Richardson extrapolations. The new extensions can allow the use of one step-size to provide optimal accuracy for both derivative approximations. Simulation results are provided to show the performance of the new complex-step approximations on a second-order Kalman filter.

I. Introduction

Using complex numbers for computational purposes is often intentionally avoided because the nonintuitive nature of this domain. However, this perception should not handicap our ability to seek better solutions to the problems associated with traditional (real-valued) finite-difference approaches. Many physical world phenomena actually have their roots in the complex domain.[1] As an aside we note that some interesting historical notes on the discovery and acceptance of the complex variable can also be found in this reference. A brief introduction of complex variable could be found on Ref. 2, pg. 1108-1109. The complex-step derivative approximation can be used to determine first derivatives in a relatively easy way, while providing near analytic accuracy. Early work on obtaining derivatives via a complex-step approximation in order to improve overall accuracy is shown by Lyness and Moler,[3] as well as Lyness.[1] Various recent papers reintroduce the complex-step approach to the engineering community.[4–8] The advantages of the complex-step approximation approach over a standard finite difference include: 1) the Jacobian approximation is not subject to subtractive cancellations inherent in roundoff errors, 2) it can be used on discontinuous functions, and 3) it is easy to implement in a black-box manner, thereby making it applicable to general nonlinear functions.

The complex-step approximation in the aforementioned papers is derived only for first derivatives. A second-order approximation using the complex-step approach is straightforward to derive; however, this approach is subject to roundoff errors for small step-sizes since difference errors arise, as shown by the classic plot in Figure 1. As the step-size increases the accuracy decreases due to truncation errors associated with not adequately approximating the true slope at the point of interest. Decreasing the step-size increases the accuracy, but only to an "optimum" point. Any further decrease results in a degradation of the accuracy due to roundoff errors. Hence, a tradeoff between truncation errors and roundoff exists. In fact, through numerous simulations, the complex-step second-derivative approximation is markedly *worse* than a standard finite-difference approach. In this paper several extensions of the complex-step approach are derived. These are essentially based on using various complex numbers coupled with Richardson extrapolations[9] to provide

[*]Graduate Student, Department of Mechanical & Aerospace Engineering. Email: klai2@buffalo.edu. Student Member AIAA.

[†]Associate Professor, Department of Mechanical & Aerospace Engineering. Email: johnc@eng.buffalo.edu. Associate Fellow AIAA.

further accuracy, instead of the standard purely imaginary approach of the aforementioned papers. As with the standard complex-step approach, all of the new first-derivative approximations are not subject to roundoff errors. However, they all have a wider range of accuracy for larger step-sizes than the standard imaginary-only approach. The new second-derivative approximations are more accurate than both the imaginary-only as well as traditional higher-order finite-difference approaches. For example, a new 4-point second-derivative approximation is derived whose accuracy is valid up to tenth-order derivative errors. These new expressions allow a designer to choose one step-size in order to provide very accurate approximations, which minimizes the required number of function evaluations.

Reference 10 incorporates up to three function variables as quaternion vector components and evaluates the function using quaternion algebra. Using this method, the Jacobian matrix can be obtained with a single function evaluation, which is far less than the one presented here. On the contrary, the method presented in this paper has no limitation on number of function variables and does not require programming of quaternion algebra, besides standard complex algebra that is often built into existing mathematical libraries of common programming languages. Consequently, the method presented here is more practical for large multi-variable functions. In addition, the Jacobian matrix is simply a byproduct of the determined Hessian matrix. This paper extends the results from Ref. 11 under a generalized framework and thus supersedes it. This work also verifies the optimality of results from that paper.

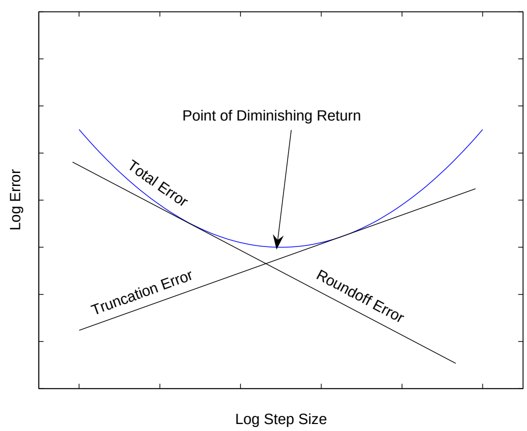


Figure 1. Finite Difference Error Versus Step-Size

The organization of this paper proceeds as follows. First, the complex-step approximation for the first derivative of a scalar function is summarized, followed by the derivation of the second-derivative approximation. Then, the Jacobian and Hessian approximations for multi-variable functions are derived. Next, several extensions of the complex-step approximation are show. A numerical example is then shown that compares the accuracy of the new approximations to standard finite-difference approaches. Then, the second-order Kalman filter is summarized. Finally, simulation results are shown that compare results using the complex-step approximations versus using and standard finite-difference approximations in the filter design.

II. Complex-Step Approximation to the Derivative

In this section the complex-step approximation is shown. First, the derivative approximation of a scalar variable is summarized, followed by an extension to the second derivative. Then, approximations for multi-variable functions are presented for the Jacobian and Hessian matrices.

A. Scalar Case

Numerical finite difference approximations for any order derivative can be obtained by Cauchy's integral formula[12]

$$f^{(n)}(z) = \frac{n!}{2\pi i} \int_{\Gamma} \frac{f(\xi)}{(\xi - z)^{n+1}} d\xi \tag{1}$$

This function can be approximated by

$$f^{(n)}(z) \approx \frac{n!}{mh} \sum_{j=0}^{m-1} \frac{f\left(z + h\, e^{i\frac{2\pi j}{m}}\right)}{e^{i\frac{2\pi j n}{m}}} \tag{2}$$

where h is the step-size and i is the imaginary unit, $\sqrt{-1}$. For example, when $n = 1$, $m = 2$

$$f^{'}(z) = \frac{1}{2\,h}\Big[f(z+h) - f(z-h)\Big] \tag{3}$$

We can see that this formula involves a substraction that would introduce near cancellation errors when the step-size becomes too small.

1. *First Derivative*

The derivation of the complex-step derivative approximation is accomplished by approximating a nonlinear function with a complex variable using a Taylor's series expansion:[7]

$$f(x+ih) = f(x) + ihf^{'}(x) - \frac{h^2}{2!}f^{''}(x) - i\frac{h^3}{3!}f^{(3)}(x) + \frac{h^4}{4!}f^{(4)}(x) + \cdots \tag{4}$$

Taking only the imaginary parts of both sides gives

$$\Im\Big\{f(x+ih)\Big\} = hf^{'}(x) - \frac{h^3}{3!}f^{(3)}(x) + \cdots \tag{5}$$

Dividing by h and rearranging yields

$$f^{'}(x) = \frac{\Im\Big\{f(x+ih)\Big\}}{h} + \cancel{\frac{h^2}{3!}f^{(3)}(x) + \cdots} \; O(h^2) \approx 0 \tag{6}$$

Terms with order h^2 or higher can be ignored since the interval h can be chosen up to machine precision. Thus, to within first-order the complex-step derivative approximation is given by

$$f^{'}(x) = \frac{\Im\Big\{f(x+ih)\Big\}}{h} \quad , \quad E_{\text{trunc}}(h) = \frac{h^2}{6}f^{(3)}(x) \tag{7}$$

where $E_{\text{trunc}}(h)$ denotes the truncation error. Note that this solution is not a function of differences, which ultimately provides better accuracy than a standard finite difference.

2. *Second Derivative*

In order to derive a second derivative approximation, the real components of Eq. (4) are taken, which gives

$$\Re\left\{\frac{h^2}{2!}f^{''}(x)\right\} = f(x) - \Re\Big\{f(x+ih)\Big\} + \frac{h^4}{4!}f^{(4)}(x) + \cdots \tag{8}$$

Solving for $f^{''}(x)$ yields

$$f^{''}(x) = \frac{2!}{h^2}\Big[f(x) - \Re\Big\{f(x+ih)\Big\}\Big] + \frac{2!h^2}{4!}f^{(4)}(x) + \cdots \tag{9}$$

Analogous to the approach shown before, we truncate up to the second-order approximation to obtain

$$f^{''}(x) = \frac{2}{h^2}\Big[f(x) - \Re\Big\{f(x+ih)\Big\}\Big] \quad , \quad E_{\text{trunc}}(h) = \frac{h^2}{12}f^{(4)}(x) \tag{10}$$

As with Cauchy's formula, we can see that this formula involves a substraction that may introduce machine cancellation errors when the step-size is too small.

B. Vector Case

The scalar case is now expanded to include vector functions. This case involves a vector $\mathbf{f}(\mathbf{x})$ of order m function equations and order n variables with $\mathbf{x} = [x_1,\ \ x_2,\ \ \cdots,\ \ x_n]^T$.

1. *First Derivative*

The Jacobian of a vector function is a simple extension of the scalar case. This Jacobian is defined by

$$F_x \triangleq \begin{bmatrix} \frac{\partial f_1(\mathbf{x})}{\partial x_1} & \frac{\partial f_1(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial f_1(\mathbf{x})}{\partial x_p} & \cdots & \frac{\partial f_1(\mathbf{x})}{\partial x_n} \\ \frac{\partial f_2(\mathbf{x})}{\partial x_1} & \frac{\partial f_2(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial f_2(\mathbf{x})}{\partial x_p} & \cdots & \frac{\partial f_2(\mathbf{x})}{\partial x_n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_q(\mathbf{x})}{\partial x_1} & \frac{\partial f_q(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial f_q(\mathbf{x})}{\partial x_p} & \cdots & \frac{\partial f_q(\mathbf{x})}{\partial x_n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_m(\mathbf{x})}{\partial x_1} & \frac{\partial f_m(\mathbf{x})}{\partial x_2} & \cdots & \frac{\partial f_m(\mathbf{x})}{\partial x_p} & \cdots & \frac{\partial f_m(\mathbf{x})}{\partial x_n} \end{bmatrix} \tag{11}$$

The complex approximation is obtained by

$$F_x = \frac{1}{h}\Im \begin{bmatrix} f_1(\mathbf{x}+ih\mathbf{e}_1) & f_1(\mathbf{x}+ih\mathbf{e}_2) & \cdots & f_1(\mathbf{x}+ih\mathbf{e}_p) & \ldots & f_1(\mathbf{x}+ih\mathbf{e}_n) \\ f_2(\mathbf{x}+ih\mathbf{e}_1) & f_2(\mathbf{x}+ih\mathbf{e}_2) & \cdots & f_2(\mathbf{x}+ih\mathbf{e}_p) & \ldots & f_2(\mathbf{x}+ih\mathbf{e}_n) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ f_q(\mathbf{x}+ih\mathbf{e}_1) & f_q(\mathbf{x}+ih\mathbf{e}_2) & \cdots & f_q(\mathbf{x}+ih\mathbf{e}_p) & \ldots & f_q(\mathbf{x}+ih\mathbf{e}_n) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ f_m(\mathbf{x}+ih\mathbf{e}_1) & f_m(\mathbf{x}+ih\mathbf{e}_2) & \cdots & f_m(\mathbf{x}+ih\mathbf{e}_p) & \ldots & f_m(\mathbf{x}+ih\mathbf{e}_n) \end{bmatrix} \tag{12}$$

where $\mathbf{e}_p$ is the p^{th} column of an n^{th}-order identity matrix and f_q is the q^{th} equation of $\mathbf{f}(\mathbf{x})$.

2. *Second Derivative*

The procedure to obtain the Hessian matrix is more involved than the Jacobian case. The Hessian matrix for the q^{th} equation of $\mathbf{f}(\mathbf{x})$ is defined by

$$F_{xx}^q \triangleq \begin{bmatrix} \frac{\partial^2 f_q(\mathbf{x})}{\partial x_1^2} & \frac{\partial^2 f_q(\mathbf{x})}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f_q(\mathbf{x})}{\partial x_1 \partial x_p} & \cdots & \frac{\partial^2 f_q(\mathbf{x})}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f_q(\mathbf{x})}{\partial x_2 \partial x_1} & \frac{\partial^2 f_q(\mathbf{x})}{\partial x_2^2} & \cdots & \frac{\partial^2 f_q(\mathbf{x})}{\partial x_2 \partial x_p} & \cdots & \frac{\partial^2 f_q(\mathbf{x})}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial^2 f_q(\mathbf{x})}{\partial x_n \partial x_1} & \frac{\partial^2 f_q(\mathbf{x})}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f_q(\mathbf{x})}{\partial x_n \partial x_p} & \cdots & \frac{\partial^2 f_q(\mathbf{x})}{\partial x_n^2} \end{bmatrix} \tag{13}$$

The complex approximation is defined by

$$F_{xx}^q \equiv \begin{bmatrix} F_{xx}^q(1,1) & F_{xx}^q(1,2) & \cdots & F_{xx}^q(1,p) & \cdots & F_{xx}^q(1,n) \\ F_{xx}^q(2,1) & F_{xx}^q(2,2) & \cdots & F_{xx}^q(2,p) & \cdots & F_{xx}^q(2,n) \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ F_{xx}^q(n,1) & F_{xx}^q(n,2) & \cdots & F_{xx}^q(n,p) & \cdots & F_{xx}^q(n,n) \end{bmatrix} \tag{14}$$

where $F^q_{xx}(i,j)$ is obtained by using Eq. (10). The easiest way to describe this procedure is by showing pseudocode, given by

$F_{xx} = \mathbf{0}_{n\times n\times m}$

for $\xi = 1$ **to** m

  $\texttt{out1} = \mathbf{f}(\mathbf{x})$

  for $\kappa = 1$ **to** n

    $\mathbf{small} = \mathbf{0}_{n\times 1}$

    $\mathbf{small}(\kappa) = 1$

    $\texttt{out2} = \mathbf{f}(\mathbf{x} + i * h * \mathbf{small})$

    $F_{xx}(\kappa,\kappa,\xi) = \dfrac{2}{h^2}\Big[\texttt{out1}(\xi) - \Re\big\{\texttt{out2}(\xi)\big\}\Big]$

  end

  $\lambda = 1$

  $\kappa = n - 1$

  while $\kappa > 0$

    for $\phi = 1$ **to** κ

      $\mathbf{img\_vec} = \mathbf{0}_{n\times 1}$

      $\mathbf{img\_vec}(\phi \ldots \phi+\lambda, 1) = 1$

      $\texttt{out2} = \mathbf{f}(\mathbf{x} + i * h * \mathbf{img\_vec})$

      $F_{xx}(\phi,\phi+\lambda,\xi) = \left[\dfrac{2}{h^2}\Big[\texttt{out1}(\xi) - \Re\big\{\texttt{out2}(\xi)\big\}\Big] - \displaystyle\sum_{\alpha=\phi}^{\phi+\lambda}\sum_{\beta=\phi}^{\phi+\lambda} F_{xx}(\alpha,\beta,\xi)\right]/2$

      $F_{xx}(\phi+\lambda,\phi,\xi) = F_{xx}(\phi,\phi+\lambda,\xi)$

    end

    $\kappa = \kappa - 1$

    $\lambda = \lambda + 1$

  end

end

where $\Re\{\cdot\}$ denotes the real value operator. The first part of this code computes the diagonal elements and the second part computes the off-diagonal elements. The Hessian matrix is a symmetric matrix, so only the upper or lower triangular elements need to be computed.

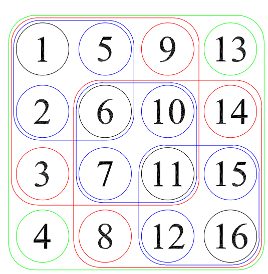


Figure 2. Graphical Illustration for Finding the Hessian matrix

A graphical illustration of the steps in obtaining the Hessian matrix for $n = 4$ is presented in Figure 2. First, we find the second derivative of all diagonal terms, which are circled in black with index $\kappa = 1, \ldots, n = 4$. Second, we find the 2×2 off-diagonal terms circled in blue with index $\lambda = 1, \kappa = 3, \phi = 1, 2, 3$. Similarly, we then proceed to the next higher dimension circled in red the with index $\lambda = 2, \kappa = 2, \phi = 1, 2$, and finally the off-diagonal terms circled in green with index $\lambda = 3, \kappa = 1, \phi = 1$.

III. New Complex-Step Approximations

It can easily be seen from Eq. (4) that deriving second-derivative approximations without some sort of difference is difficult, if not intractable. With any complex number I that has $|I| = 1$, it's impossible for $I^2 \perp 1$ and $I^2 \perp I$. But, it may be possible to obtain better approximations than Eq. (10). Let us again list down the Taylor series expansions with complex step sizes that would assist us in the forthcoming derivation and analysis.

$$i^{\frac{0}{6}} = 1 \qquad \begin{aligned} f(x+h) = {} & f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{3!}f^{(3)}(x) + \frac{h^4}{4!}f^{(4)}(x) + \frac{h^5}{5!}f^{(5)}(x) \\ & + \frac{h^6}{6!}f^{(6)}(x) + \frac{h^7}{7!}f^{(7)}(x) + \frac{h^8}{8!}f^{(8)}(x) + \frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15a}$$

$$i^{\frac{1}{6}} \qquad \begin{aligned} f(x+i^{\frac{1}{6}}h) = {} & f(x) + i^{\frac{1}{6}}hf'(x) + i^{\frac{1}{3}}\frac{h^2}{2}f''(x) + i^{\frac{1}{2}}\frac{h^3}{3!}f^{(3)}(x) + i^{\frac{2}{3}}\frac{h^4}{4!}f^{(4)}(x) + i^{\frac{5}{6}}\frac{h^5}{5!}f^{(5)}(x) \\ & + i\frac{h^6}{6!}f^{(6)}(x) + i^{\frac{7}{6}}\frac{h^7}{7!}f^{(7)}(x) + i^{\frac{4}{3}}\frac{h^8}{8!}f^{(8)}(x) + i^{\frac{3}{2}}\frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15b}$$

$$i^{\frac{2}{6}} = i^{\frac{1}{3}} \qquad \begin{aligned} f(x+i^{\frac{1}{3}}h) = {} & f(x) + i^{\frac{1}{3}}hf'(x) + i^{\frac{2}{3}}\frac{h^2}{2}f''(x) + i\frac{h^3}{3!}f^{(3)}(x) + i^{\frac{4}{3}}\frac{h^4}{4!}f^{(4)}(x) + i^{\frac{5}{3}}\frac{h^5}{5!}f^{(5)}(x) \\ & - \frac{h^6}{6!}f^{(6)}(x) + i^{\frac{7}{3}}\frac{h^7}{7!}f^{(7)}(x) + i^{\frac{8}{3}}\frac{h^8}{8!}f^{(8)}(x) - i\frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15c}$$

$$i^{\frac{3}{6}} = i^{\frac{1}{2}} \qquad \begin{aligned} f(x+i^{\frac{1}{2}}h) = {} & f(x) + i^{\frac{1}{2}}hf'(x) + i\frac{h^2}{2}f''(x) + i^{\frac{3}{2}}\frac{h^3}{3!}f^{(3)}(x) - \frac{h^4}{4!}f^{(4)}(x) + i^{\frac{5}{2}}\frac{h^5}{5!}f^{(5)}(x) \\ & - i\frac{h^6}{6!}f^{(6)}(x) + i^{\frac{7}{2}}\frac{h^7}{7!}f^{(7)}(x) + \frac{h^8}{8!}f^{(8)}(x) + i^{\frac{9}{2}}\frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15d}$$

$$i^{\frac{4}{6}} = i^{\frac{2}{3}} \qquad \begin{aligned} f(x+i^{\frac{2}{3}}h) = {} & f(x) + i^{\frac{2}{3}}hf'(x) + i^{\frac{4}{3}}\frac{h^2}{2}f''(x) - \frac{h^3}{3!}f^{(3)}(x) + i^{\frac{8}{3}}\frac{h^4}{4}f^{(4)}(x) + i^{\frac{10}{3}}\frac{h^5}{5}f^{(5)}(x) \\ & + \frac{h^6}{6}f^{(6)}(x) + i^{\frac{14}{3}}\frac{h^7}{7}f^{(7)}(x) + i^{\frac{16}{3}}\frac{h^8}{8}f^{(8)}(x) - \frac{h^9}{9}f^{(9)}(x) \end{aligned} \tag{15e}$$

$$i^{\frac{5}{6}} \qquad \begin{aligned} f(x+i^{\frac{5}{6}}h) = {} & f(x) + i^{\frac{5}{6}}hf'(x) + i^{\frac{5}{3}}\frac{h^2}{2}f''(x) + i^{\frac{5}{2}}\frac{h^3}{3!}f^{(3)}(x) + i^{\frac{10}{3}}\frac{h^4}{4!}f^{(4)}(x) + i^{\frac{25}{6}}\frac{h^5}{5!}f^{(5)}(x) \\ & + i\frac{h^6}{6!}f^{(6)}(x) + i^{\frac{35}{6}}\frac{h^7}{7!}f^{(7)}(x) + i^{\frac{20}{3}}\frac{h^8}{8!}f^{(8)}(x) + i^{\frac{15}{2}}\frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15f}$$

$$i^{\frac{6}{6}} = i \qquad \begin{aligned} f(x+ih) = {} & f(x) + ihf'(x) - \frac{h^2}{2}f''(x) - i\frac{h^3}{3!}f^{(3)}(x) + \frac{h^4}{4!}f^{(4)}(x) + i\frac{h^5}{5!}f^{(5)}(x) \\ & - \frac{h^6}{6!}f^{(6)}(x) - i\frac{h^7}{7!}f^{(7)}(x) + \frac{h^8}{8!}f^{(8)}(x) + i\frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15g}$$

$$i^{\frac{7}{6}} \qquad \begin{aligned} f(x+i^{\frac{7}{6}}h) = {} & f(x) + i^{\frac{7}{6}}hf'(x) + i^{\frac{7}{3}}\frac{h^2}{2}f''(x) + i^{\frac{7}{2}}\frac{h^3}{3!}f^{(3)}(x) + i^{\frac{14}{3}}\frac{h^4}{4!}f^{(4)}(x) + i^{\frac{35}{6}}\frac{h^5}{5!}f^{(5)}(x) \\ & - i\frac{h^7}{7!}f^{(7)}(x) + i^{\frac{49}{6}}\frac{h^7}{7!}f^{(7)}(x) + i^{\frac{28}{3}}\frac{h^8}{8!}f^{(8)}(x) + i^{\frac{21}{2}}\frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15h}$$

$$i^{\frac{8}{6}} = i^{\frac{4}{3}} \qquad \begin{aligned} f(x + i^{\frac{4}{3}} h) = {} & f(x) + i^{\frac{4}{3}} h f'(x) + i^{\frac{8}{3}} \frac{h^2}{2} f''(x) + \frac{h^3}{3!} f^{(3)}(x) + i^{\frac{16}{3}} \frac{h^4}{4!} f^{(4)}(x) + i^{\frac{20}{3}} \frac{h^5}{5!} f^{(5)}(x) \\ & + \frac{h^6}{6!} f^{(6)}(x) + i^{\frac{28}{3}} \frac{h^7}{7!} f^{(7)}(x) + i^{\frac{32}{3}} \frac{h^8}{8!} f^{(8)}(x) + \frac{h^9}{9!} f^{(9)}(x) \end{aligned} \tag{15i}$$

$$i^{\frac{9}{6}} = i^{\frac{3}{2}} \qquad \begin{aligned} f(x + i^{\frac{3}{2}} h) = {} & f(x) + i^{\frac{3}{2}} h f'(x) - i \frac{h^2}{2} f''(x) + i^{\frac{1}{2}} \frac{h^3}{3!} f^{(3)}(x) - \frac{h^4}{4!} f^{(4)}(x) + i^{\frac{15}{2}} \frac{h^5}{5!} f^{(5)}(x) \\ & + i \frac{h^6}{6!} f^{(6)}(x) + i^{\frac{21}{2}} \frac{h^7}{7!} f^{(7)}(x) + \frac{h^8}{8!} f^{(8)}(x) + i^{\frac{27}{2}} \frac{h^9}{9!} f^{(9)}(x) \end{aligned} \tag{15j}$$

$$i^{\frac{10}{6}} = i^{\frac{5}{3}} \qquad \begin{aligned} f(x + i^{\frac{5}{3}} h) = {} & f(x) + i^{\frac{5}{3}} h f'(x) + i^{\frac{10}{3}} \frac{h^2}{2} f''(x) + i \frac{h^3}{3!} f^{(3)}(x) + i^{\frac{20}{3}} \frac{h^4}{4!} f^{(4)}(x) + i^{\frac{25}{3}} \frac{h^5}{5!} f^{(5)}(x) \\ & - \frac{h^6}{6!} f^{(6)}(x) + i^{\frac{35}{3}} \frac{h^7}{7!} f^{(7)}(x) + i^{\frac{40}{3}} \frac{h^8}{8!} f^{(8)}(x) - i \frac{h^9}{9!} f^{(9)}(x) \end{aligned} \tag{15k}$$

$$i^{\frac{11}{6}} \qquad \begin{aligned} f(x + i^{\frac{11}{6}} h) = {} & f(x) + i^{\frac{11}{6}} h f'(x) + i^{\frac{11}{3}} \frac{h^2}{2} f''(x) + i^{\frac{11}{2}} \frac{h^3}{3!} f^{(3)}(x) + i^{\frac{22}{3}} \frac{h^4}{4!} f^{(4)}(x) + i^{\frac{55}{6}} \frac{h^5}{5!} f^{(5)}(x) \\ & - i \frac{h^6}{6!} f^{(6)}(x) + i^{\frac{77}{6}} \frac{h^7}{7!} f^{(7)}(x) + i^{\frac{44}{3}} \frac{h^8}{8!} f^{(8)}(x) + i^{\frac{33}{2}} \frac{h^9}{9!} f^{(9)}(x) \end{aligned} \tag{15l}$$

$$i^{\frac{12}{6}} = -1 \qquad \begin{aligned} f(x - h) = {} & f(x) - h f'(x) + \frac{h^2}{2} f''(x) - \frac{h^3}{3!} f^{(3)}(x) + \frac{h^4}{4!} f^{(4)}(x) - \frac{h^5}{5!} f^{(5)}(x) \\ & + \frac{h^6}{6!} f^{(6)}(x) - \frac{h^7}{7!} f^{(7)}(x) + \frac{h^8}{8!} f^{(8)}(x) - \frac{h^9}{9!} f^{(9)}(x) \end{aligned} \tag{15m}$$

$$i^{\frac{13}{6}} \qquad \begin{aligned} f(x + i^{\frac{13}{6}} h) = {} & f(x) + i^{\frac{13}{6}} h f'(x) + i^{\frac{13}{3}} \frac{h^2}{2} f''(x) + i^{\frac{13}{2}} \frac{h^3}{3!} f^{(3)}(x) + i^{\frac{26}{3}} \frac{h^4}{4!} f^{(4)}(x) + i^{\frac{65}{6}} \frac{h^5}{5!} f^{(5)}(x) \\ & + i \frac{h^6}{6!} f^{(6)}(x) + i^{\frac{91}{6}} \frac{h^7}{7!} f^{(7)}(x) + i^{\frac{52}{3}} \frac{h^8}{8!} f^{(8)}(x) + i^{\frac{39}{2}} \frac{h^9}{9!} f^{(9)}(x) \end{aligned} \tag{15n}$$

$$i^{\frac{14}{6}} = i^{\frac{7}{3}} \qquad \begin{aligned} f(x + i^{\frac{7}{3}} h) = {} & f(x) + i^{\frac{7}{3}} h f'(x) + i^{\frac{14}{3}} \frac{h^2}{2} f''(x) - i \frac{h^3}{3!} f^{(3)}(x) + i^{\frac{28}{3}} \frac{h^4}{4!} f^{(4)}(x) + i^{\frac{35}{3}} \frac{h^5}{5!} f^{(5)}(x) \\ & - \frac{h^6}{6!} f^{(6)}(x) + i^{\frac{49}{3}} \frac{h^7}{7!} f^{(7)}(x) + i^{\frac{56}{3}} \frac{h^8}{8!} f^{(8)}(x) + i \frac{h^9}{9!} f^{(9)}(x) \end{aligned} \tag{15o}$$

$$i^{\frac{15}{6}} = i^{\frac{5}{2}} \qquad \begin{aligned} f(x + i^{\frac{5}{2}} h) = {} & f(x) + i^{\frac{5}{2}} h f'(x) + i \frac{h^2}{2} f''(x) + i^{\frac{15}{2}} \frac{h^3}{3!} f^{(3)}(x) - \frac{h^4}{4!} f^{(4)}(x) + i^{\frac{25}{2}} \frac{h^5}{5!} f^{(5)}(x) \\ & - i \frac{h^6}{6!} f^{(6)}(x) + i^{\frac{35}{2}} \frac{h^7}{7!} f^{(7)}(x) + \frac{h^8}{8!} f^{(8)}(x) + i^{\frac{45}{2}} \frac{h^9}{9!} f^{(9)}(x) \end{aligned} \tag{15p}$$

$$i^{\frac{16}{6}} = i^{\frac{8}{3}} \qquad \begin{aligned} f(x + i^{\frac{8}{3}} h) = {} & f(x) + i^{\frac{8}{3}} h f'(x) + i^{\frac{16}{3}} \frac{h^2}{2} f''(x) + \frac{h^3}{3!} f^{(3)}(x) + i^{\frac{32}{3}} \frac{h^4}{4!} f^{(4)}(x) + i^{\frac{40}{3}} \frac{h^5}{5!} f^{(5)}(x) \\ & + \frac{h^6}{6!} f^{(6)}(x) + i^{\frac{56}{3}} \frac{h^7}{7!} f^{(7)}(x) + i^{\frac{64}{3}} \frac{h^8}{8!} f^{(8)}(x) + \frac{h^9}{9!} f^{(9)}(x) \end{aligned} \tag{15q}$$

$$i^{\frac{17}{6}} \qquad \begin{aligned} f(x+i^{\frac{17}{6}}h) = {} & f(x) + i^{\frac{17}{6}}hf'(x) + i^{\frac{17}{3}}\frac{h^2}{2}f''(x) + i^{\frac{34}{3}}\frac{h^4}{4!}f^{(4)}(x) + i^{\frac{85}{6}}\frac{h^5}{5!}f^{(5)}(x) \\ & + i\frac{h^6}{6!}f^{(6)}(x) + i^{\frac{119}{6}}\frac{h^7}{7!}f^{(7)}(x) + i^{\frac{68}{3}}\frac{h^8}{8!}f^{(8)}(x) + i^{\frac{51}{2}}\frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15r}$$

$$i^{\frac{18}{6}} = -i \qquad \begin{aligned} f(x-ih) = {} & f(x) - ihf'(x) - \frac{h^2}{2}f''(x) + i\frac{h^3}{3!}f^{(3)}(x) + \frac{h^4}{4!}f^{(4)}(x) - i\frac{h^5}{5!}f^{(5)}(x) \\ & - \frac{h^6}{6!}f^{(6)}(x) + i\frac{h^7}{7!}f^{(7)}(x) + \frac{h^8}{8!}f^{(8)}(x) - i\frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15s}$$

$$i^{\frac{19}{6}} \qquad \begin{aligned} f(x+i^{\frac{19}{6}}h) = {} & f(x) + i^{\frac{19}{6}}hf'(x) + i^{\frac{19}{3}}\frac{h^2}{2}f''(x) + i^{\frac{19}{2}}\frac{h^3}{3!}f^{(3)}(x) + i^{\frac{38}{3}}\frac{h^4}{4!}f^{(4)}(x) + i^{\frac{95}{6}}\frac{h^5}{5!}f^{(5)}(x) \\ & - i\frac{h^6}{6!}f^{(6)}(x) + i^{\frac{133}{6}}\frac{h^7}{7!}f^{(7)}(x) + i^{\frac{76}{3}}\frac{h^8}{8!}f^{(8)}(x) + i^{\frac{57}{2}}\frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15t}$$

$$i^{\frac{20}{6}} = i^{\frac{10}{3}} \qquad \begin{aligned} f(x+i^{\frac{10}{3}}h) = {} & f(x) + i^{\frac{10}{3}}hf'(x) + i^{\frac{20}{3}}\frac{h^2}{2}f''(x) - \frac{h^3}{3!}f^{(3)}(x) + i^{\frac{40}{3}}\frac{h^4}{4!}f^{(4)}(x) + i^{\frac{50}{3}}\frac{h^5}{5!}f^{(5)}(x) \\ & + \frac{h^6}{6!}f^{(6)}(x) + i^{\frac{70}{3}}\frac{h^7}{7!}f^{(7)}(x) + i^{\frac{80}{3}}\frac{h^8}{8!}f^{(8)}(x) - \frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15u}$$

$$i^{\frac{21}{6}} = i^{\frac{7}{2}} \qquad \begin{aligned} f(x+i^{\frac{7}{2}}h) = {} & f(x) + i^{\frac{7}{2}}hf'(x) - i\frac{h^2}{2}f''(x) + i^{\frac{21}{2}}\frac{h^3}{3!}f^{(3)}(x) - \frac{h^4}{4!}f^{(4)}(x) + i^{\frac{35}{2}}\frac{h^5}{5!}f^{(5)}(x) \\ & + i\frac{h^6}{6!}f^{(6)}(x) + i^{\frac{49}{2}}\frac{h^7}{7!}f^{(7)}(x) + \frac{h^8}{8!}f^{(8)}(x) + i^{\frac{63}{2}}\frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15v}$$

$$i^{\frac{22}{6}} = i^{\frac{11}{3}} \qquad \begin{aligned} f(x+i^{\frac{11}{3}}h) = {} & f(x) + i^{\frac{11}{3}}hf'(x) + i^{\frac{22}{3}}\frac{h^2}{2}f''(x) - i\frac{h^3}{3!}f^{(3)}(x) + i^{\frac{44}{3}}\frac{h^4}{4!}f^{(4)}(x) + i^{\frac{55}{3}}\frac{h^5}{5!}f^{(5)}(x) \\ & - \frac{h^6}{6!}f^{(6)}(x) + i^{\frac{77}{3}}\frac{h^7}{7!}f^{(7)}(x) + i^{\frac{88}{3}}\frac{h^8}{8!}f^{(8)}(x) + i\frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15w}$$

$$i^{\frac{23}{6}} \qquad \begin{aligned} f(x+i^{\frac{23}{6}}h) = {} & f(x) + i^{\frac{23}{6}}hf'(x) + i^{\frac{23}{3}}\frac{h^2}{2}f''(x) + i^{\frac{23}{2}}\frac{h^3}{3!}f^{(3)}(x) + i^{\frac{46}{3}}\frac{h^4}{4!}f^{(4)}(x) + i^{\frac{115}{6}}\frac{h^5}{5!}f^{(5)}(x) \\ & - i\frac{h^6}{6!}f^{(6)}(x) + i^{\frac{161}{6}}\frac{h^7}{7!}f^{(7)}(x) + i^{\frac{92}{3}}\frac{h^8}{8!}f^{(8)}(x) + i^{\frac{69}{2}}\frac{h^9}{9!}f^{(9)}(x) \end{aligned} \tag{15x}$$

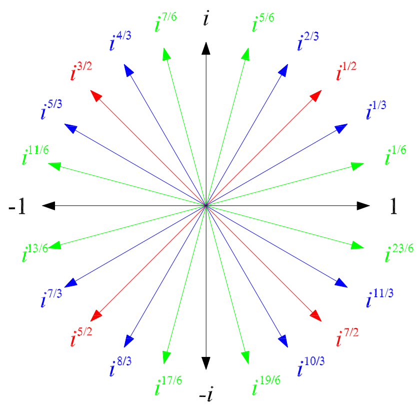
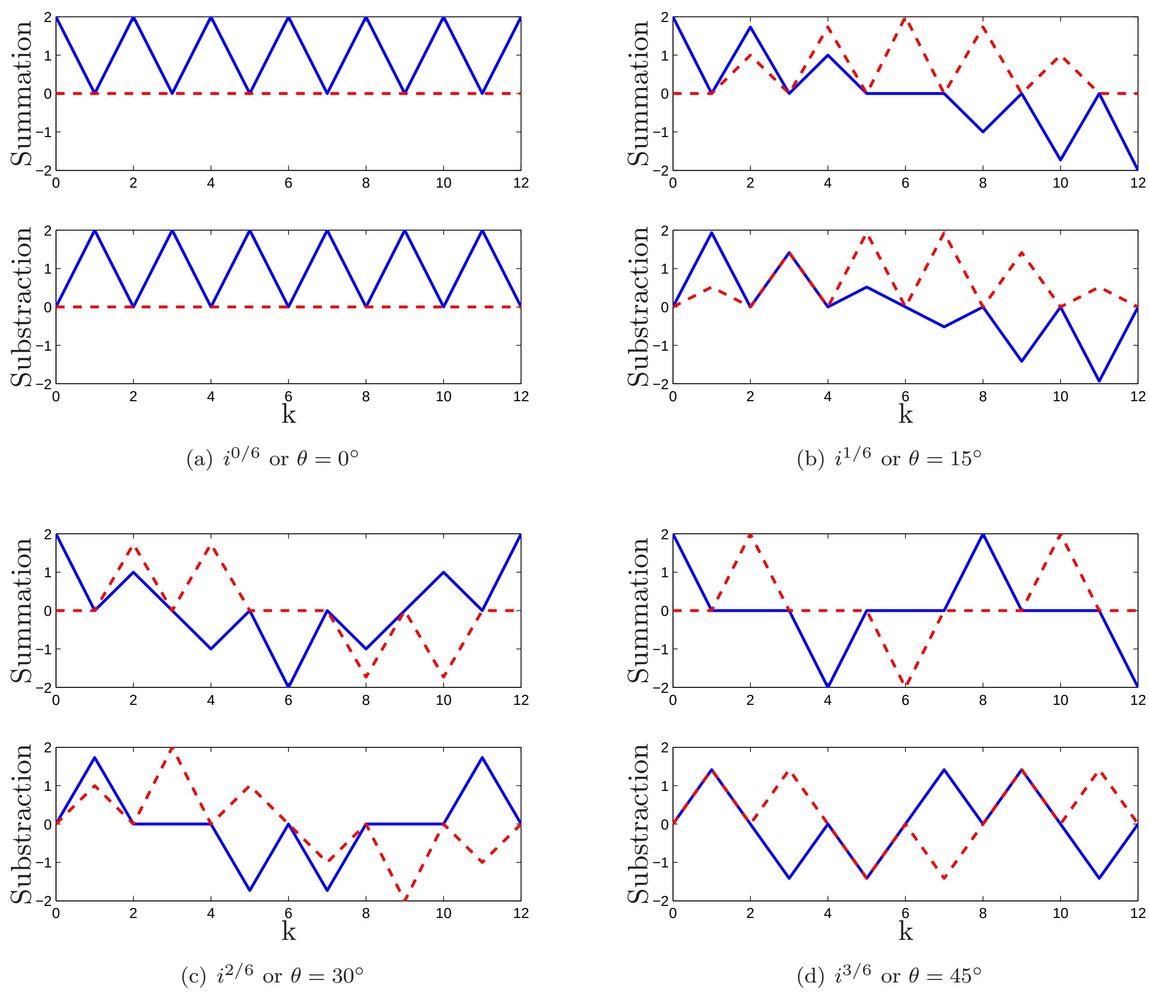


Figure 3. Various Complex Numbers



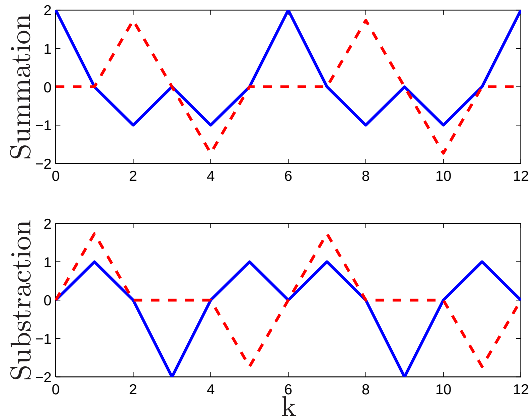
(a) $i^{0/6}$ or $\theta = 0°$

(b) $i^{1/6}$ or $\theta = 15°$

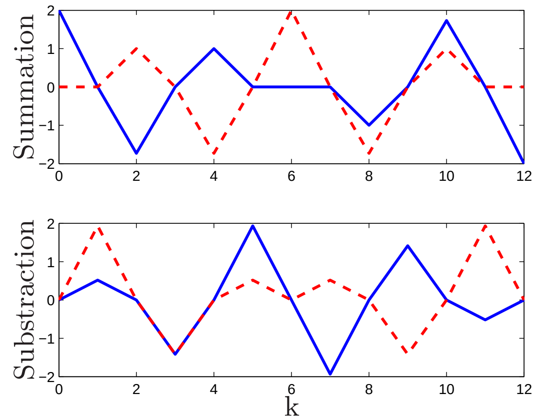
(c) $i^{2/6}$ or $\theta = 30°$

(d) $i^{3/6}$ or $\theta = 45°$

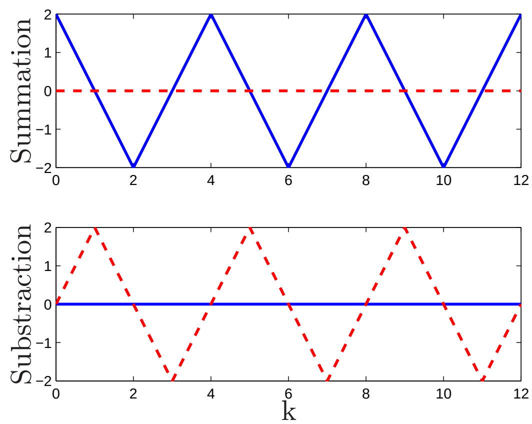
Figure 4. Summation and Subtraction for θ from $0°$ to $45°$ (Solid Lines = Real, Dotted Lines = Imaginary)



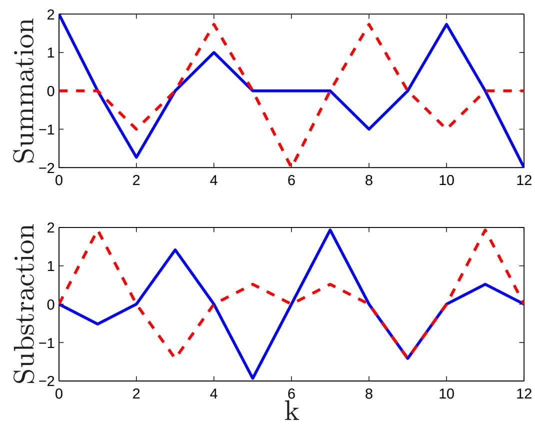
(a) $i^{4/6}$ or $\theta = 60°$



(b) $i^{5/6}$ or $\theta = 75°$



(c) $i^{6/6}$ or $\theta = 90°$



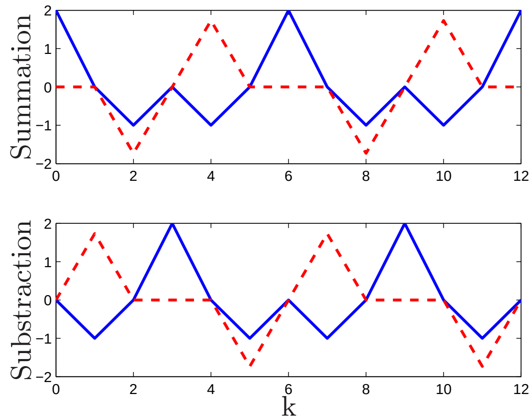
(d) $i^{7/6}$ or $\theta = 105°$

Figure 5. Summation and Subtraction for θ from $60°$ to $105°$ (Solid Lines = Real, Dotted Lines = Imaginary)

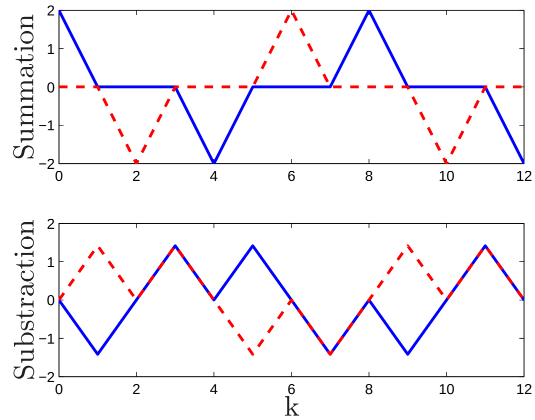
Figure 3 shows the unity magnitude complex number raised to various rational number powers with common denominator of 6, i.e. multiple of 15°. Figures 4, 5 and 6 show the summation and substraction pairs of Taylor series expansion with complex step sizes that are 180° apart. It may be more convenient to represent the complex number in another way. With help from trigonometry identities these can be derived using $i^{p/q} = e^{i\theta}$ with phase angle $\theta = \frac{p}{q}90° = \frac{p}{2q}\pi$ rad. The Taylor series expansion pair with complex step-sizes can then be rewritten as

$$\begin{aligned} f(x + e^{i\theta}h) = f(x) &+ e^{i\theta}hf'(x) + e^{2i\theta}\frac{h^2}{2}f''(x) + e^{3i\theta}\frac{h^3}{3!}f^{(3)}(x) \\ &+ e^{4i\theta}\frac{h^4}{4!}f^{(4)}(x) + e^{5i\theta}\frac{h^5}{5!}f^{(5)}(x) + e^{6i\theta}\frac{h^6}{6!}f^{(6)}(x) \\ &+ e^{7i\theta}\frac{h^7}{7!}f^{(7)}(x) + e^{8i\theta}\frac{h^8}{8!}f^{(8)}(x) + e^{9i\theta}\frac{h^9}{9!}f^{(9)}(x) \\ &+ e^{10i\theta}\frac{h^{10}}{10!}f^{(10)}(x) \end{aligned} \tag{16a}$$

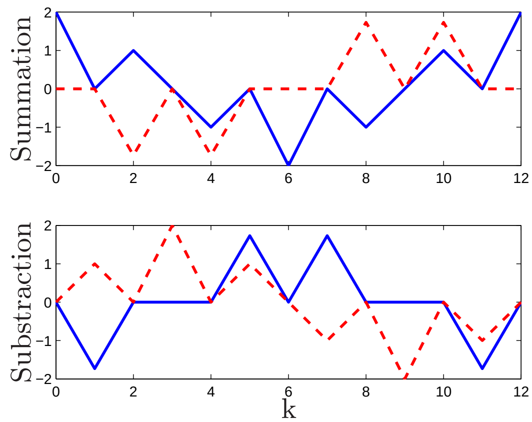
$$\begin{aligned} f(x + e^{i(\theta+\pi)}h) = f(x) &+ e^{i(\theta+\pi)}hf'(x) + e^{2i(\theta+\pi)}\frac{h^2}{2}f''(x) + e^{3i(\theta+\pi)}\frac{h^3}{3!}f^{(3)}(x) \\ &+ e^{4i(\theta+\pi)}\frac{h^4}{4!}f^{(4)}(x) + e^{5i(\theta+\pi)}\frac{h^5}{5!}f^{(5)}(x) + e^{6i(\theta+\pi)}\frac{h^6}{6!}f^{(6)}(x) \end{aligned}$$



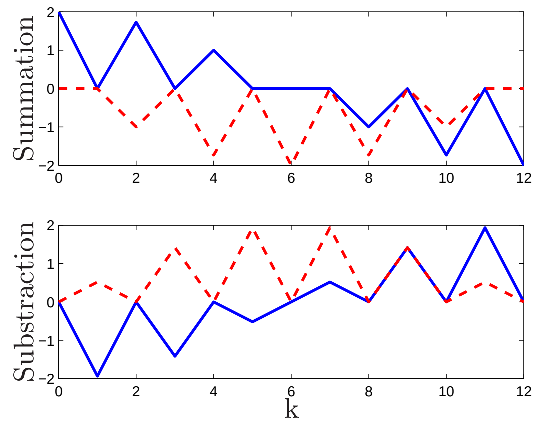
(a) $i^{8/6}$ or $\theta = 120°$



(b) $i^{9/6}$ or $\theta = 135°$



(c) $i^{10/6}$ or $\theta = 150°$



(d) $i^{11/6}$ or $\theta = 165°$

Figure 6. Summation and Subtraction for θ from $120°$ to $165°$ (Solid Lines = Real, Dotted Lines = Imaginary)

$$
\begin{aligned}
&+ e^{7i(\theta+\pi)}\frac{h^7}{7!}f^{(7)}(x) + e^{8i(\theta+\pi)}\frac{h^8}{8!}f^{(8)}(x) + e^{9i(\theta+\pi)}\frac{h^9}{9!}f^{(9)}(x) \\
&+ e^{10i(\theta+\pi)}\frac{h^{10}}{10!}f^{(10)}(x)
\end{aligned}
\tag{16b}
$$

The addition and substraction of the equation pair above gives

$$
\begin{aligned}
f(x+e^{i\theta}h) + f(x+e^{i(\theta+\pi)}h) = 2f(x) &+ 2[\cos 2\theta + i\sin 2\theta]\frac{h^2}{2}f''(x) \\
&+ 2[\cos 4\theta + i\sin 4\theta]\frac{h^4}{4!}f^{(4)}(x) \\
&+ 2[\cos 6\theta + i\sin 6\theta]\frac{h^6}{6!}f^{(6)}(x) \\
&+ 2[\cos 8\theta + i\sin 8\theta]\frac{h^8}{8!}f^{(8)}(x) \\
&+ 2[\cos 10\theta + i\sin 10\theta]\frac{h^{10}}{10!}f^{(10)}(x)
\end{aligned}
\tag{17a}
$$

$$
\begin{aligned}
f(x+e^{i\theta}h) - f(x+e^{i(\theta+\pi)}h) = &\ 2[\cos\theta + i\sin\theta]hf'(x) + 2[\cos 3\theta + i\sin 3\theta]\frac{h^3}{3!}f^{(3)}(x) \\
&+ 2[\cos 5\theta + i\sin 5\theta]\frac{h^5}{5!}f^{(5)}(x)
\end{aligned}
$$

$$+ 2[\cos 7\theta + i\sin 7\theta]\frac{h^7}{7!}f^{(7)}(x)$$
$$+ 2[\cos 9\theta + i\sin 9\theta]\frac{h^9}{9!}f^{(9)}(x) \tag{17b}$$

Solving these two equations for $f''(x)$ and $f'(x)$ yields

$$\begin{aligned} f''(x) = {} & \frac{f(x+e^{i\theta}h) - 2f(x) + f(x+e^{i(\theta+pi)}h)}{[\cos 2\theta + i\sin 2\theta]h^2} - \frac{2[\cos 4\theta + i\sin 4\theta]}{\cos 2\theta + i\sin 2\theta}\frac{h^2}{4!}f^{(4)}(x) \\ & - \frac{2[\cos 6\theta + i\sin 6\theta]}{\cos 2\theta + i\sin 2\theta}\frac{h^4}{6!}f^{(6)}(x) - \frac{2[\cos 8\theta + i\sin 8\theta]}{\cos 2\theta + i\sin 2\theta}\frac{h^6}{8!}f^{(8)}(x) \\ & - \frac{2[\cos 10\theta + i\sin 10\theta]}{\cos 2\theta + i\sin 2\theta}\frac{h^8}{10!}f^{(10)}(x) \end{aligned} \tag{18a}$$

$$\begin{aligned} f'(x) = {} & \frac{f(x+e^{i\theta}h) - f(x+e^{i(\theta+\pi)}h)}{2[\cos\theta + i\sin\theta]h} - \frac{\cos 3\theta + i\sin 3\theta}{\cos\theta + i\sin\theta}\frac{h^2}{3!}f^{(3)}(x) \\ & - \frac{\cos 5\theta + i\sin 5\theta}{\cos\theta + i\sin\theta}\frac{h^4}{5!}f^{(5)}(x) - \frac{\cos 7\theta + i\sin 7\theta}{\cos\theta + i\sin\theta}\frac{h^6}{7!}f^{(7)}(x) \\ & - \frac{\cos 9\theta + i\sin 9\theta}{\cos\theta + i\sin\theta}\frac{h^8}{9!}f^{(9)}(x) \end{aligned} \tag{18b}$$

A. General Form

Now the pattern is obvious from previous derivation and we can generalize the complex Taylor series expansion pair from the previous section as

$$f(x+e^{i\theta}h) = f(x) + \sum_{n=1}^{\infty} e^{ni\theta}\frac{h^n}{n!}f^{(n)}(x) \tag{19a}$$

$$f(x+e^{i(\theta+\pi)}h) = f(x) + \sum_{n=1}^{\infty} e^{ni(\theta+\pi)}\frac{h^n}{n!}f^{(n)}(x) \tag{19b}$$

and their summation and substraction pair

$$f(x+e^{i\theta}h) + f(x+e^{i(\theta+\pi)}h) = 2f(x) + 2\sum_{n=1}^{\infty}\left[\cos 2n\theta + i\sin 2n\theta\right]\frac{h^{2n}}{(2n)!}f^{(2n)}(x) \tag{20a}$$

$$\begin{aligned} f(x+e^{i\theta}h) - f(x+e^{i(\theta+\pi)}h) = 2\sum_{n=1}^{\infty}\Bigg[& \cos[(2n-1)\theta] \\ & + i\sin[(2n-1)\theta]\Bigg]\frac{h^{2n-1}}{(2n-1)!}f^{(2n-1)}(x) \end{aligned} \tag{20b}$$

Finally solving for $f''(x)$ and $f'(x)$ yields

$$\begin{aligned} f''(x) = {} & \frac{f(x+e^{i\theta}h) - 2f(x) + f(x+e^{i(\theta+\pi)}h)}{[\cos 2\theta + i\sin 2\theta]h^2} \\ & - \frac{2}{\cos 2\theta + i\sin 2\theta}\sum_{n=2}^{\infty}\left[\cos 2n\theta + i\sin 2n\theta\right]\frac{h^{2n-2}}{(2n)!}f^{(2n)}(x) \end{aligned} \tag{21a}$$

$$\begin{aligned} f'(x) = {} & \frac{f(x+e^{i\theta}h) - f(x+e^{i(\theta+\pi)}h)}{2[\cos\theta + i\sin\theta]h} \\ & - \frac{1}{\cos\theta + i\sin\theta}\sum_{n=2}^{\infty}\left[\cos[(2n-1)\theta] + i\sin[(2n-1)\theta]\right]\frac{h^{2n-2}}{(2n-1)!}f^{(2n-1)}(x) \end{aligned} \tag{21b}$$

Another way of representing the complex number is raising the imagination number, i, to an appropriate power or by using Euler's identity; however, in the current approach we can clearly identify the real and imagination quantities. This formulation also works for pure real-valued finite differences by simply using $\theta = 0$. The Richardson extrapolation applies here too. The extension of all the aforementioned approximations to multi-variables for the Jacobian and Hessian matrices is straightforward, which follow along similar lines as the previous section.

B. Useful Cases

Observing Figures 4 to 6, there are several *interesting* cases where certain elements (real or imaginary component) of the series annihilate. This phenomenon is desirable and to be taken advantage to increase the convergence rate of the Taylor series approximation towards the original nonlinear function. In fact, this is the main goal of evaluating functions with a complex step size.

Let us take a moment to briefly review a recursive form of Richardson extrapolation.[9] Assuming D as the derivative approximation, let the first column be

$$D_{\alpha,1} = D(h/q^{\alpha-1}) \quad \text{for } \alpha = 1, \ldots, n \tag{22}$$

and other elements as

$$D_{\alpha,\beta} = \frac{q^{k_{\beta-1}} D_{\alpha,\beta-1} - D_{\alpha-1,\beta-1}}{q^{k_{\beta-1}} - 1} \quad \text{for } \beta = 2, \ldots, n \tag{23}$$

Then we can find higher precision approximation by filling up a lower triangle matrix

$$\begin{array}{lllll}
D_{1,1} & & & & \\
D_{2,1} & D_{2,2} = \frac{q^{k_1} D_{2,1} - D_{1,1}}{q^{k_1} - 1} & & & \\
D_{3,1} & D_{3,2} = \frac{q^{k_1} D_{3,1} - D_{2,1}}{q^{k_1} - 1} & D_{3,3} = \frac{q^{k_2} D_{3,2} - D_{2,2}}{q^{k_2} - 1} & & \\
\vdots & \vdots & \vdots & \ddots & \\
D_{n,1} & D_{n,2} = \frac{q^{k_1} D_{n,1} - D_{n-1,1}}{q^{k_1} - 1} & D_{n,3} = \frac{q^{k_2} D_{n,2} - D_{n-1,2}}{q^{k_2} - 1} & \cdots & D_{n,n} = \frac{q^{k_{n-1}} D_{n,n-1} - D_{n-1,n-1}}{q^{k_{n-1}} - 1}
\end{array}$$

where the last element, $D_{n,n}$ is the most accurate approximation with error $O(h^{k_n})$.

1. $\underline{\theta = 45^\circ}$

From Eq. (21a) with $\theta = 45^\circ$ and taking only the imaginary components gives

$$f^{''}(x) = \frac{\Im\{f(x + i^{1/2}h) + f(x + i^{5/2}h)\}}{h^2} \quad , \quad E_{\text{trunc}}(h) = \frac{h^4}{360} f^{(6)}(x) \tag{24}$$

Note that when $n = 2$, the imaginary component $\sin 2n\theta = 0$, thus the first non-zero value occurs when $n = 3$ corresponds to $O(h^4)$, which is the main goal of the complex-step derivative approximation. This approximation is still subject to difference errors, but the truncation error associated with this approximation is $h^4 f^{(6)}(x)/360$ whereas the error associated with Eq. (10) is $h^2 f^{(4)}(x)/12$. It will also be shown through simulation that Eq. (24) is less sensitive to roundoff errors than Eq. (10).

Unfortunately, to obtain the first and second derivatives using Eqs. (7) and (24) requires function evaluations of $f(x + ih)$, $f(x + i^{1/2}h)$ and $f(x + i^{5/2}h)$. To obtain a first-derivative expression that involves $f(x+i^{1/2}h)$ and $f(x+i^{5/2}h)$, we substitute $\theta = 45^\circ$ into Eq. (21b) and again take the imaginary components:

$$f^{'}(x) = \frac{f(x + i^{1/2}h) - f(x + i^{5/2}h)}{\sqrt{2}\,(i + 1)\,h} \tag{25}$$

Actually either the imaginary or real parts of Eq. (25) can be taken to determine $f^{'}(x)$; however, it's better to use the imaginary parts since no differences exist (they are actually *additions* of imaginary numbers) since $f(x + i^{1/2}h) - f(x + i^{5/2}h) = f(x + i^{1/2}h) - f(x - i^{1/2}h)$. This yields

$$f^{'}(x) = \frac{\Im\{f(x + i^{1/2}h) - f(x + i^{5/2}h)\}}{h\sqrt{2}} \quad , \quad E_{\text{trunc}}(h) = -\frac{h^2}{6} f^{(3)}(x) \tag{26}$$

The approximation in Eq. (26) has errors equal to Eq. (7). Hence, both forms yield identical answers; however, Eq. (26) uses the same function evaluations as Eq. (24).

Now, a Richardson extrapolation is applied for further refinement. From Eq. (23) with $q = 2$ and $k_1 = 4$,

$$
\begin{aligned}
f^{''}(x) &= \frac{2^4 \frac{\Im\left\{f\left(x+i^{1/2}\frac{h}{2}\right)+f\left(x+i^{5/2}\frac{h}{2}\right)\right\}}{h^2/4} - \frac{\Im\left\{f(x+i^{1/2}h)+f(x+i^{5/2}h)\right\}}{h^2}}{2^4-1} \\
&= \frac{\Im\left\{64\left[f(x+i^{1/2}\frac{h}{2})+f(x+i^{5/2}\frac{h}{2})\right] - \left[f(x+i^{1/2}h)+f(x+i^{5/2}h)\right]\right\}}{15h^2} \quad ,
\end{aligned}
$$

$$
E_{\text{trunc}}(h) = -\frac{h^8}{1,814,400} f^{(10)}(x) \tag{27}
$$

This approach can be continued *ad nauseam* using the next value of k. However, the next highest-order derivative-difference past $O(h^8)$ that has imaginary parts is $O(h^{12})$. This error is given by $\frac{h^{12}}{4.35891456\times10^{10}} f^{(14)}(x)$. Hence, it seems unlikely that the accuracy will improve much by using more terms. The same approach can be applied to the first derivative as well. Applying the Richardson extrapolation with $q = 2$, $k_1 = 2$, to Eq. (26) yields

$$
f^{'}(x) = \Im\left\{8\left[f\left(x+i^{1/2}\frac{h}{2}\right) - f\left(x+i^{5/2}\frac{h}{2}\right)\right] - \left[f\left(x+i^{1/2}h\right) - f\left(x+i^{5/2}h\right)\right]\right\} / (3\sqrt{2}\,h) \quad ,
$$

$$
E_{\text{trunc}}(h) = \frac{h^4}{120} f^{(5)}(x) \tag{28}
$$

Performing the Richardson extrapolation again would cancel fifth-order derivative errors, which leads to the following approximation:

$$
\begin{aligned}
f^{'}(x) = \Im\Bigg\{ & 4096\left[f\left(x+i^{1/2}\frac{h}{4}\right) - f\left(x+i^{5/2}\frac{h}{4}\right)\right] - 640\left[f\left(x+i^{1/2}\frac{h}{2}\right) - f\left(x+i^{5/2}\frac{h}{2}\right)\right] \\
& + 16\left[f(x+i^{1/2}h) - f(x+i^{5/2}h)\right]\Bigg\} / (720\sqrt{2}\,h) \quad , \quad E_{\text{trunc}}(h) = \frac{h^6}{5040} f^{(7)}(x)
\end{aligned} \tag{29}
$$

As with Eq. (26), the approximations in Eq. (28) and Eq. (29) are not subject to substraction cancellation error, so an arbitrarily small value of h can be chosen up to the roundoff error.

2. $\underline{\theta = 60^\circ}$

Using $\theta = 60^\circ$ in Eq. (21) gives

$$
f^{'}(x) = \frac{\Im\left\{f(x+i^{2/3}h) - f(x+i^{8/3}h)\right\}}{\sqrt{3}h} \quad , \quad E_{\text{trunc}}(h) = \frac{h^4}{120} f^{(5)}(x) \tag{30a}
$$

$$
f^{''}(x) = \frac{\Im\left\{f(x+i^{2/3}h) + f(x+i^{8/3}h)\right\}}{\sqrt{3}h} \quad , \quad E_{\text{trunc}}(h) = \frac{h^2}{24} f^{(4)}(x) \tag{30b}
$$

Performing a Richardson extrapolation once on each of these equations yields

$$
f^{'}(x) = \frac{\Im\left\{32\left[f\left(x+i^{2/3}\frac{h}{2}\right) - f\left(x+i^{8/3}\frac{h}{2}\right)\right] - \left[f(x+i^{2/3}h) - f(x+i^{8/3}h)\right]\right\}}{15\sqrt{3}\,h} \quad ,
$$

$$
E_{\text{trunc}}(h) = -\frac{h^6}{5040} f^{(7)}(x) \tag{31a}
$$

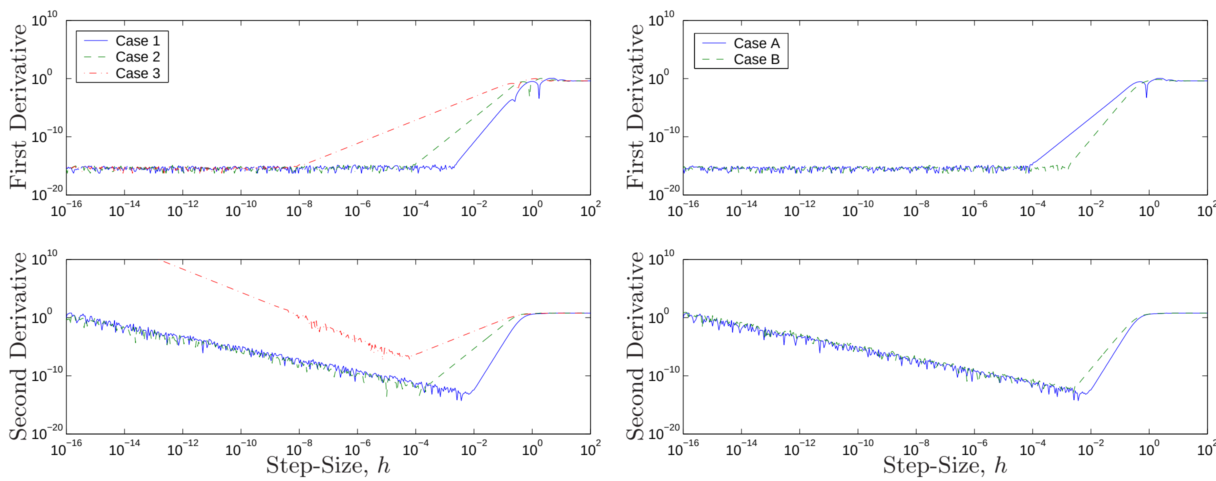
$$
f^{''}(x) = 2\frac{\Im\left\{\left[f(x+i^{2/3}h) + f(x+i^{8/3}h)\right] - 16\left[f(x+i^{2/3}\frac{h}{2}) + f(x+i^{8/3}\frac{h}{2})\right]\right\}}{3\sqrt{3}\,h^2} \quad ,
$$

$$
E_{\text{trunc}}(h) = -\frac{h^6}{40320} f^{(8)}(x) \tag{31b}
$$

These solutions have the same order of accuracy as Eq. (29), but involves less function evaluations. Using $i^{2/3}$ instead of $i^{1/2}$ for the second-derivative approximation yields worse results than Eq. (27) since the approximation has errors on the order of $h^6\, f^{(8)}(x)$ instead of $h^8\, f^{(10)}(x)$. Hence, a tradeoff between the first-derivative and second-derivative accuracy will always exist if using the same function evaluations for both is desired. Higher-order versions of Eq. (31) are given by

$$\begin{aligned} f^{'}(x) = \Im\Big\{&3072\left[f(x+i^{2/3}\frac{h}{4}) - f(x+i^{8/3}\frac{h}{4})\right] - 256\left[f(x+i^{2/3}\frac{h}{2}) - f(x+i^{8/3}\frac{h}{2})\right] \\ &+ 5\left[f(x+i^{2/3}h) - f(x+i^{8/3}h)\right]\Big\}/(645\sqrt{3}\,h) \quad , \\ &E_{\text{trunc}}(h) = \frac{h^{10}}{39916800} f^{(11)}(x) \end{aligned} \tag{32a}$$

$$\begin{aligned} f^{''}(x) = 2\,\Im\Big\{&15\left[f(x+i^{2/3}h) + f(x+i^{8/3}h)\right] + 16\left[f(x+i^{2/3}\frac{h}{2}) + f(x+i^{8/3}\frac{h}{2})\right] \\ &- 4096\left[f(x+i^{2/3}\frac{h}{4}) + f(x+i^{8/3}\frac{h}{4})\right]\Big\}/(237\sqrt{3}\,h^2) \quad , \\ &E_{\text{trunc}}(h) = \frac{h^{8}}{3628800} f^{(10)}(x) \end{aligned} \tag{32b}$$



(a) First and Second Derivative Errors

(b) First and Second Derivative Errors

Figure 7. Comparisons of the Various Complex-Derivative Approaches

C. Simple Examples

Consider the following highly nonlinear function:

$$f(x) = \frac{e^x}{\sqrt{\sin^3(x) + \cos^3(x)}} \tag{33}$$

evaluated at $x = -0.5$. Error results for the first and second derivative approximations are shown in Figure 7(a). Case 1 shows results using Eqs. (29) and (27) for the first and second order derivatives, respectively. Case 2 shows results using Eqs. (28) and (24) for the first and second order derivatives, respectively. Case 3 shows results using Eqs. (7) and (10) for the first and second order derivatives, respectively. We again note that using Eq. (26) produces the same results as using Eq. (7). Using Eqs. (29) and (27) for the approximations allows one to use only one step-size for all function evaluations. For this example, setting $h = 0.024750$ gives a first derivative error on the order of 10^{-16} and a second derivative error on the order of 10^{-15}. Figure 7(b) shows results using Eqs. (28) and (27), Case A, versus results using Eqs. (31a) and

Table 1. Iteration Results of x Using $h = 1 \times 10^{-8}$ for the Complex-Step and Finite-Difference Approaches

Iteration	Complex-Step	Finite-Difference
0	5.0000	5.0000
1	4.5246	4.4628
2	3.8886	5.1509
3	3.4971	2.6087
4	3.0442	3.2539
5	2.4493	2.5059
6	2.0207	3.2198
7	1.6061	5.2075
8	1.0975	1.3786×10^{1}
9	5.9467×10^{-1}	1.3753×10^{1}
10	2.9241×10^{-1}	1.3395×10^{1}
11	6.6074×10^{-2}	1.2549×10^{1}
12	1.2732×10^{-3}	1.2061×10^{1}
13	1.0464×10^{-8}	1.1628×10^{1}
14	-3.6753×10^{-17}	1.1583×10^{1}
15	-3.6753×10^{-17}	1.1016×10^{1}

(31b), Case B, for the first and second derivatives, respectively. For this example using Eqs. (31a) and (31b) provides the best overall accuracy with the least amount of function evaluations for both derivatives.

Another example is given by using Halley's method for root finding. The iteration function is given by

$$x_{n+1} = x_n - \frac{2\, f(x_n)\, f^{'}(x_n)}{2\, [f'(x_n)]^2 - f(x_n)\, f''(x_n)} \tag{34}$$

The following function is tested:

$$f(x) = \frac{(1 - e^x)\, e^{3x}}{\sqrt{\sin^4(x) + \cos^4(x)}} \tag{35}$$

which has a root at $x = 0$. Equation (34) is used to determine the root with a starting value of $x_0 = 5$. Equations (28) and (27) are used for the complex-step approximations. For comparison purposes the derivatives are also determined using a symmetric 4-point approximation for the first derivative and a 5-point approximation for the second derivative:

$$f^{'}(x) = \frac{f(x - 2\,h) - 8\, f(x - h) + 8\, f(x + h) - f(x + 2\,h)}{12h}\,, \qquad E_{\text{trunc}}(h) = \frac{h^4}{30} f^{(5)}(x) \tag{36a}$$

$$f^{''}(x) = \frac{-f(x - 2\,h) + 16\, f(x - h) - 30\, f(x) + 16\, f(x + h) - f(x + 2\,h)}{12\, h^2}\,, \qquad E_{\text{trunc}}(h) = \frac{h^6}{90} f^{(6)}(x) \tag{36b}$$

MATLAB® is used to perform the numerical computations. Various values of h are tested in decreasing magnitude (by one order each time), starting at $h = 0.1$ and going down to $h = 1 \times 10^{-16}$. For values of

$h = 0.1$ to $h = 1 \times 10^{-7}$ both methods converge, but the complex-step approach convergence is faster or (at worst) equal to the standard finite-difference approach. For values less than 1×10^{-7}, e.g. when $h = 1\times10^{-8}$, the finite-difference approach becomes severally degraded. Table 1 shows the iterations for both approaches using 1×10^{-8}. For h values from 1×10^{-8} down to 1×10^{-15}, the complex-step approach always converges in less than 15 iterations. When $h = 1 \times 10^{-16}$ the finite-difference approach produces a zero-valued correction for all iterations, while the complex-step approach converges in about 40 iterations.

D. Multi-Variable Numerical Example

A multi-variable example is now shown to assess the performance of the complex-step approximations. The infinity norm[‖] is used to access the accuracy of the numerical finite-difference and complex-step approximation solutions. The relationship between the magnitude of the various solutions and step-size is also discussed. The function to be tested is given by two equations with four variables:

$$\mathbf{f} \triangleq \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \begin{bmatrix} x_1^2 x_2 x_3 x_4^2 + x_2^2 x_3^3 x_4 \\ x_1^2 x_2 x_3^2 x_4 + x_1 x_2^3 x_4^2 \end{bmatrix} \tag{37}$$

The Jacobian is given by

$$F_x = \begin{bmatrix} 2x_1x_2x_3x_4^2 & x_1^2x_3x_4^2 + 2x_2x_3^3x_4 & x_1^2x_2x_4^2 + 3x_2^2x_3^2x_4 & 2x_1^2x_2x_3x_4 + x_2^2x_3^3 \\ 2x_1x_2x_3^2x_4 + x_2^3x_4^2 & x_1^2x_3^2x_4 + 3x_1x_2^2x_4^2 & 2x_1^2x_2x_3x_4 & x_1^2x_2x_3^2 + 2x_1x_2^3x_4 \end{bmatrix} \tag{38}$$

The two Hessian matrices are given by

$$F_{xx}^1 = \begin{bmatrix} 2x_2x_3x_4^2 & 2x_1x_3x_4^2 & 2x_1x_2x_4^2 & 4x_1x_2x_3x_4 \\ 2x_1x_3x_4^2 & 2x_3^3x_4 & x_1^2x_4^2 + 6x_2x_3^2x_4 & 2x_1^2x_3x_4 + 2x_2x_3^3 \\ 2x_1x_2x_4^2 & x_1^2x_4^2 + 6x_2x_3^2x_4 & 6x_2^2x_3x_4 & 2x_1^2x_2x_4 + 3x_2^2x_3^2 \\ 4x_1x_2x_3x_4 & 2x_1^2x_3x_4 + 2x_2x_3^3 & 2x_1^2x_2x_4 + 3x_2^2x_3^2 & 2x_1^2x_2x_3 \end{bmatrix} \tag{39a}$$

$$F_{xx}^2 = \begin{bmatrix} 2x_2x_3^2x_4 & 2x_1x_3^2x_4 + 3x_2^2x_4^2 & 4x_1x_2x_3x_4 & 2x_1x_2x_3^2 + 2x_2^3x_4 \\ 2x_1x_3^2x_4 + 3x_2^2x_4^2 & 6x_1x_2x_4^2 & 2x_1^2x_3x_4 & x_1^2x_3^2 + 6x_1x_2^2x_4 \\ 4x_1x_2x_3x_4 & 2x_1^2x_3x_4 & 2x_1^2x_2x_4 & 2x_1^2x_2x_3 \\ 2x_1x_2x_3^2 + 2x_2^3x_4 & x_1^2x_3^2 + 6x_1x_2^2x_4 & 2x_1^2x_2x_3 & 2x_1x_2^3 \end{bmatrix} \tag{39b}$$

Given $\mathbf{x} = [5,\ 3,\ 6,\ 4]^T$ the following analytical solutions are obtained:

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} 14976 \\ 12960 \end{bmatrix} \tag{40a}$$

$$F_x = \begin{bmatrix} 2880 & 7584 & 5088 & 5544 \\ 4752 & 5760 & 3600 & 3780 \end{bmatrix} \tag{40b}$$

$$F_{xx}^1 = \begin{bmatrix} 576 & 960 & 480 & 1440 \\ 960 & 1728 & 2992 & 2496 \\ 480 & 2992 & 1296 & 1572 \\ 1440 & 2496 & 1572 & 900 \end{bmatrix} \tag{40c}$$

$$F_{xx}^2 = \begin{bmatrix} 864 & 1872 & 1440 & 1296 \\ 1872 & 1440 & 1200 & 1980 \\ 1440 & 1200 & 600 & 900 \\ 1296 & 1980 & 900 & 270 \end{bmatrix} \tag{40d}$$

[‖]The largest row sum of a matrix A, $|A|_\infty = \max\{\sum |A^T|\}$.

Numerical Solutions

The step-size for the Jacobian and Hessian calculations (both for complex-step approximation and numerical finite difference) is 1×10^{-4}. The absolute Jacobian error between the true and complex-step solutions, and true and numerical finite-difference solutions, respectively, are

$$|\Delta^c F_x| = \begin{bmatrix} 0.0000 & 0.0000 & 0.3600 & 0.0000 \\ 0.0000 & 0.8000 & 0.0000 & 0.0000 \end{bmatrix} \times 10^{-8} \tag{41a}$$

$$|\Delta^n F_x| = \begin{bmatrix} 0.2414 & 0.3348 & 0.0485 & 0.1074 \\ 0.1051 & 0.4460 & 0.0327 & 0.0298 \end{bmatrix} \times 10^{-7} \tag{41b}$$

$$\tag{41c}$$

The infinity norms of Eq. (41) are 8.0008×10^{-9} and 7.3217×10^{-8}, respectively, which means that the complex-step solution is more accurate than the finite-difference one. The absolute Hessian error between the true solutions and the complex-step and numerical finite-difference solutions, respectively, are

$$|\Delta^c F^1_{xx}| = \begin{bmatrix} 0.0000 & 0.0011 & 0.0040 & 0.0016 \\ 0.0011 & 0.0010 & 0.0011 & 0.0009 \\ 0.0040 & 0.0011 & 0.0019 & 0.0021 \\ 0.0016 & 0.0009 & 0.0021 & 0.0004 \end{bmatrix} \tag{42a}$$

$$|\Delta^n F^1_{xx}| = \begin{bmatrix} 0.0002 & 0.0010 & 0.0041 & 0.0017 \\ 0.0010 & 0.0009 & 0.0011 & 0.0011 \\ 0.0041 & 0.0011 & 0.0021 & 0.0019 \\ 0.0017 & 0.0011 & 0.0019 & 0.0003 \end{bmatrix} \tag{42b}$$

and

$$|\Delta^c F^2_{xx}| = \begin{bmatrix} 0.0018 & 0.0007 & 0.0030 & 0.0064 \\ 0.0007 & 0.0016 & 0.0010 & 0.0018 \\ 0.0030 & 0.0010 & 0.0018 & 0.0004 \\ 0.0064 & 0.0018 & 0.0004 & 0.0029 \end{bmatrix} \tag{43a}$$

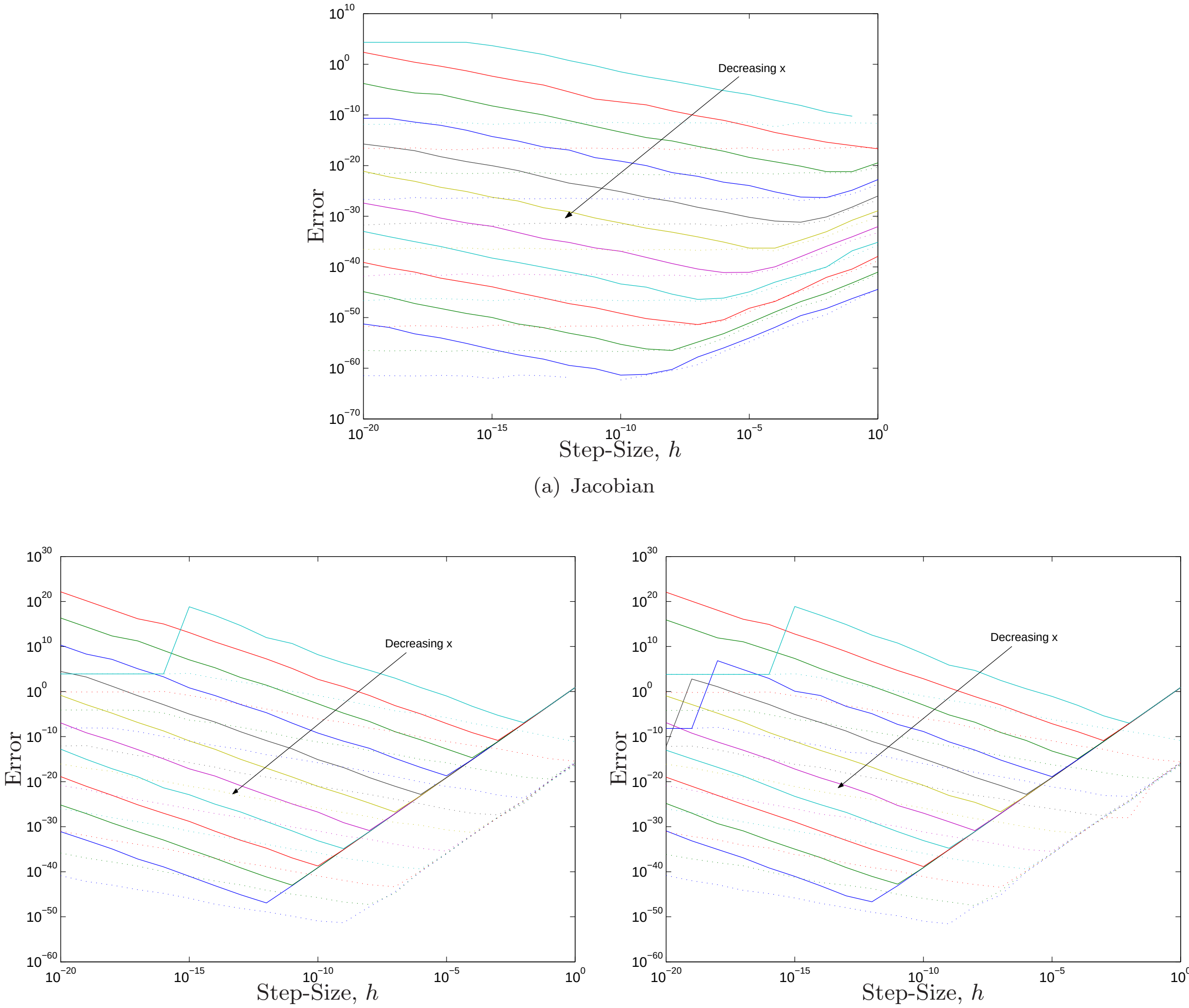
$$|\Delta^n F^2_{xx}| = \begin{bmatrix} 0.0018 & 0.0007 & 0.0031 & 0.0065 \\ 0.0007 & 0.0015 & 0.0008 & 0.0021 \\ 0.0031 & 0.0008 & 0.0018 & 0.0006 \\ 0.0065 & 0.0021 & 0.0006 & 0.0025 \end{bmatrix} \tag{43b}$$

The infinity norms of Eq. (42) are 9.0738×10^{-3} and 9.1858×10^{-3}, respectively, and the infinity norms of Eq. (43) are 1.1865×10^{-3} and 1.2103×10^{-3}, respectively. As with the Jacobian, the complex-step Hessian approximation solutions are more accurate than the finite difference solutions.

Performance Evaluation

The performance of the complex-step approach in comparison to the numerical finite-difference approach is examined further here using the same function. Tables 2 and 3 shows the infinity norm of the error between the true and the approximated solutions. The difference between the finite difference solution and the complex-step solution is also included in the last three rows, where positive values indicate the complex-step solution is more accurate. In most cases, the complex-step approach performs either comparable or better than the finite-difference approach. The complex-step approach provides accurate solutions for h values from 0.1 down to 1×10^{-9}. However, the range of accurate solutions for the finite-difference approach is significantly smaller than that of complex-step approach. Clearly, the complex-step approach is much more robust than the numerical finite-difference approach.

Figure 8 shows plots of the infinity norm of the Jacobian and Hessian errors obtained using a numerical finite-difference and the complex-step approximation. The function is evaluated at different magnitudes by



(a) Jacobian

(b) Hessian 1

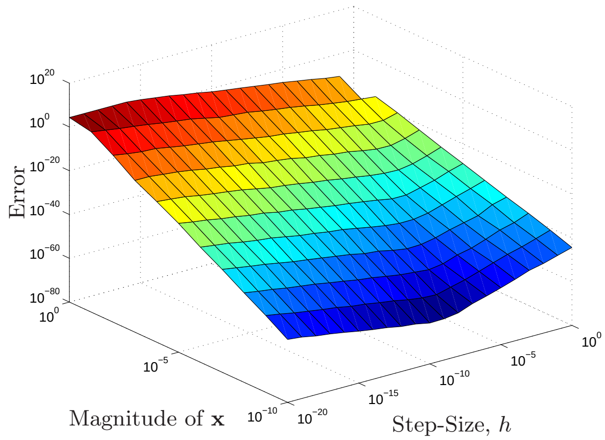
(c) Hessian 2

Figure 8. Infinity Norm of the Error Matrix for Different Magnitudes (Solid Lines = Finite Difference, Dotted Lines = Complex-Step)

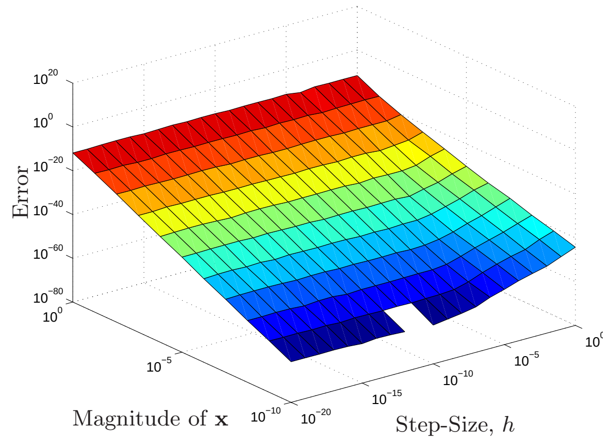
multiplying the nominal values with a scale factor from 1 down to 1×10^{-10}. The direction of the arrow shows the solutions for decreasing $\mathbf{x}$. The solutions for the complex-step and finite-difference approximation using the same $\mathbf{x}$ value are plotted with the same color within a plot.

For the case of the finite-difference Jacobian, shown in Figure 8(a), at some certain point of decreasing step-size, as mentioned before, the substraction cancellation error becomes dominant which decreases the accuracy. The complex-step solution does not exhibit this phenomenon and the accuracy continues to increase with decreasing step-size up to machine precision. As a higher-order complex-step approximation is used, Eq. (31a) instead of Eq. (7), the truncation errors for the complex-step Jacobian at larger step-sizes are also greatly reduced to the extent that the truncation errors are almost unnoticeable, even at large $\mathbf{x}$ values. The complex-step approximation for the Hessian case also benefits from the higher-order approximation, as shown in Figures 8(b) and 8(c). The complex-step Hessian approximation used to generate these results is given by Eq. (31b). One observation is that there is always only one (global) optimum of specific step-size with respect to the error.

Figures 9 and 10 represent the same information in more intuitive looking three-dimensional plots. The "depth" of the error in log scale is represented as a color scale with dark red being the highest and dark blue being the lowest. A groove is clearly seen in most of the plots (except the complex-step Jacobian), which corresponds to the optimum step-size. The "empty surface" in Figure 9 corresponds to when the difference

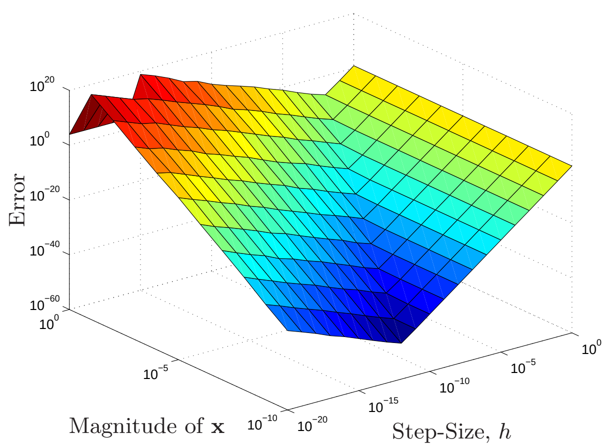


(a) Jacobian - Finite-Difference

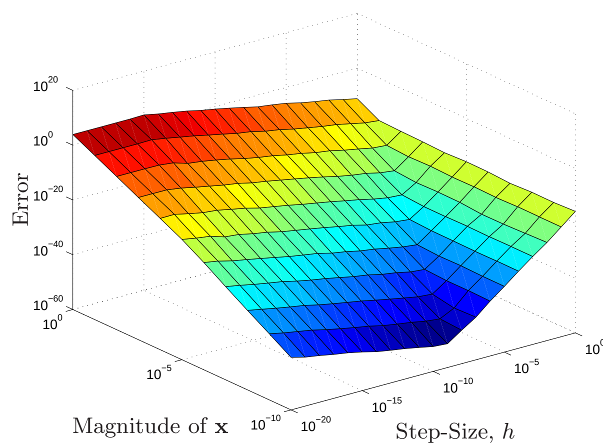


(b) Jacobian - Complex-Step

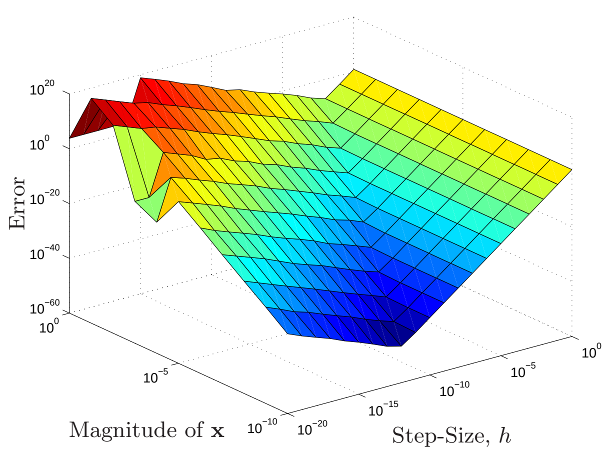
Figure 9. Infinity Norm of the Jacobian Error Matrix for Different Magnitudes and Step-Sizes



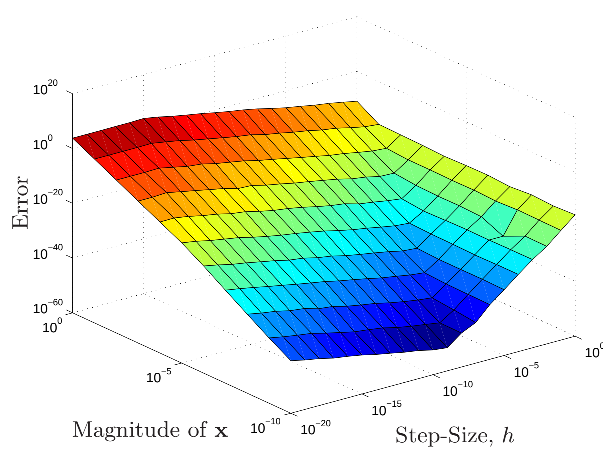
(a) Hessian 1 - Finite-Difference



(b) Hessian 1 - Complex-Step



(c) Hessian 2 - Finite-Difference



(d) Hessian 2 - Complex-Step

Figure 10. Infinity Norm of the Hessian Error Matrix for Different Magnitudes and Step-Sizes

Table 2. Infinity Norm of the Difference from Truth for Larger Step-Sizes, h

h	1×10^{0}	1×10^{-1}	1×10^{-2}
$\lvert\Delta^n F_x\rvert$	8.0004×10^{-9}	8.0554×10^{-9}	8.2664×10^{-9}
$\lvert\Delta^c F_x\rvert$	8.0026×10^{-9}	8.0004×10^{-9}	8.0013×10^{-9}
$\lvert\Delta^n F^1_{xx}\rvert$	8.0000	9.1000×10^{-3}	9.1000×10^{-3}
$\lvert\Delta^c F^1_{xx}\rvert$	9.1000×10^{-3}	9.1000×10^{-3}	9.1000×10^{-3}
$\lvert\Delta^n F^2_{xx}\rvert$	7.9990	1.1100×10^{-2}	1.1900×10^{-2}
$\lvert\Delta^c F^2_{xx}\rvert$	1.1900×10^{-2}	1.1900×10^{-2}	1.1900×10^{-2}
$\lvert\Delta^n F_x\rvert - \lvert\Delta^c F_x\rvert$	-2.2737×10^{-12}	5.5024×10^{-11}	2.6512×10^{-10}
$\lvert\Delta^n F^1_{xx}\rvert - \lvert\Delta^c F^1_{xx}\rvert$	7.9909	-5.0477×10^{-11}	3.5698×10^{-10}
$\lvert\Delta^n F^2_{xx}\rvert - \lvert\Delta^c F^2_{xx}\rvert$	7.9871	-8.0000×10^{-4}	-6.5184×10^{-8}

h	1×10^{-3}	1×10^{-4}
$\lvert\Delta^n F_x\rvert$	9.6984×10^{-9}	7.3218×10^{-8}
$\lvert\Delta^c F_x\rvert$	8.0026×10^{-9}	8.0008×10^{-9}
$\lvert\Delta^n F^1_{xx}\rvert$	9.1000×10^{-3}	9.2000×10^{-3}
$\lvert\Delta^c F^1_{xx}\rvert$	9.1000×10^{-3}	9.1000×10^{-3}
$\lvert\Delta^n F^2_{xx}\rvert$	1.1900×10^{-2}	1.2100×10^{-2}
$\lvert\Delta^c F^2_{xx}\rvert$	1.1900×10^{-2}	1.1900×10^{-2}
$\lvert\Delta^n F_x\rvert - \lvert\Delta^c F_x\rvert$	1.6958×10^{-9}	6.5217×10^{-8}
$\lvert\Delta^n F^1_{xx}\rvert - \lvert\Delta^c F^1_{xx}\rvert$	1.5272×10^{-6}	1.1200×10^{-4}
$\lvert\Delta^n F^2_{xx}\rvert - \lvert\Delta^c F^2_{xx}\rvert$	-5.3940×10^{-8}	2.3823×10^{-4}

Table 3. Infinity Norm of the Difference from Truth for Smaller Step-Sizes, h

h	1×10^{-5}	1×10^{-6}	1×10^{-7}
$\lvert\Delta^n F_x\rvert$	1.0133×10^{-6}	6.4648×10^{-6}	5.8634×10^{-5}
$\lvert\Delta^c F_x\rvert$	8.0026×10^{-9}	8.0004×10^{-9}	8.0026×10^{-9}
$\lvert\Delta^n F^1_{xx}\rvert$	1.0160×10^{-1}	7.6989	9.5627×10^{2}
$\lvert\Delta^c F^1_{xx}\rvert$	9.1000×10^{-3}	9.1000×10^{-3}	9.1000×10^{-3}
$\lvert\Delta^n F^2_{xx}\rvert$	7.3500×10^{-2}	4.2094	3.1084×10^{2}
$\lvert\Delta^c F^2_{xx}\rvert$	1.1900×10^{-2}	1.1900×10^{-2}	1.1700×10^{-2}
$\lvert\Delta^n F_x\rvert - \lvert\Delta^c F_x\rvert$	1.0053×10^{-6}	6.4568×10^{-6}	5.8626×10^{-5}
$\lvert\Delta^n F^1_{xx}\rvert - \lvert\Delta^c F^1_{xx}\rvert$	9.2500×10^{-2}	7.6898	9.5626×10^{2}
$\lvert\Delta^n F^2_{xx}\rvert - \lvert\Delta^c F^2_{xx}\rvert$	6.1600×10^{-2}	4.1976	3.1082×10^{2}

h	1×10^{-8}	1×10^{-9}	1×10^{-10}
$\lvert\Delta^n F_x\rvert$	5.0732×10^{-4}	3.5000×10^{-3}	3.1200×-2
$\lvert\Delta^c F_x\rvert$	8.0013×10^{-9}	7.9995×10^{-9}	7.9999×-9
$\lvert\Delta^n F^1_{xx}\rvert$	5.2882×10^{4}	2.2007×10^{6}	1.5916×8
$\lvert\Delta^c F^1_{xx}\rvert$	9.1000×10^{-3}	1.4800×10^{-2}	1.2730×-1
$\lvert\Delta^n F^2_{xx}\rvert$	4.9658×10^{4}	7.6182×10^{5}	2.4253×8
$\lvert\Delta^c F^2_{xx}\rvert$	1.3500×10^{-2}	8.8000×10^{-3}	9.9700×-2
$\lvert\Delta^n F_x\rvert - \lvert\Delta^c F_x\rvert$	5.0731×10^{-4}	3.5000×10^{-3}	3.1200×-2
$\lvert\Delta^n F^1_{xx}\rvert - \lvert\Delta^c F^1_{xx}\rvert$	5.2882×10^{4}	2.2007×10^{6}	1.5916×8
$\lvert\Delta^n F^2_{xx}\rvert - \lvert\Delta^c F^2_{xx}\rvert$	4.9658×10^{4}	7.6182×10^{5}	2.4253×8

Table 4. Discrete Second-Order Kalman Filter

Model	$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) + \mathbf{w}_k, \quad \mathbf{w}_k \sim N(\mathbf{0}, Q_k)$ $\tilde{\mathbf{y}}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{v}_k, \quad \mathbf{v}_k \sim N(\mathbf{0}, R_k)$
Initialize	$\hat{\mathbf{x}}(t_0) = \hat{\mathbf{x}}_0$ $P_0 = E\{\tilde{\mathbf{x}}_0 \tilde{\mathbf{x}}_0^T\}$
Propagation	$\hat{\mathbf{x}}_{k+1}^- = \mathbf{f}_k(\hat{\mathbf{x}}_k^+) + \frac{1}{2}\sum_{i=1}^n \mathbf{e}_i \mathrm{Tr}\left\{F_{xx,k}^i P_k^+\right\}$ $P_{k+1}^- = F_{x,k} P_k^+ F_{x,k}^T + \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n \mathbf{e}_i \mathbf{e}_j^T \mathrm{Tr}\left\{F_{xx,k}^i P_k^+ F_{xx,k}^j P_k^+\right\} + Q_k$ $\hat{\mathbf{y}}_{k+1} = \mathbf{h}_{k+1}(\hat{\mathbf{x}}_{k+1}^-) + \frac{1}{2}\sum_{i=1}^m \mathbf{e}_i \mathrm{Tr}\left\{H_{xx,k+1}^i P_{k+1}^-\right\}$
Gain	$P_{k+1}^{xy} = H_{x,k+1} P_{k+1}^- H_{x,k+1}^T + \frac{1}{2}\sum_{i=1}^m \sum_{j=1}^m \mathbf{e}_i \mathbf{e}_j^T \mathrm{Tr}\left\{H_{xx,k+1}^i P_{k+1}^- H_{xx,k+1}^j P_{k+1}^-\right\} + R_{k+1}$ $K_{k+1} = P_{k+1}^- H_{x,k+1}^T [P_{k+1}^{xy}]^{-1}$
Update	$\hat{\mathbf{x}}_{k+1}^+ = \hat{\mathbf{x}}_{k+1}^- + K_{k+1}[\tilde{\mathbf{y}}_{k+1} - \hat{\mathbf{y}}_{k+1}]$ $P_{k+1}^+ = P_{k+1}^- - K_{k+1} P_{k+1}^{xy} K_{k+1}^T$

between the complex-step solution and the truth is below machine precision. This is shown as "missing line" in Figure 8(a). Clearly, the complex-step approximation solutions are comparable or more accurate than the finite-difference solutions.

IV. Second-Order Kalman Filter

The extended Kalman filter (EKF) is undeniably the most widely used algorithm for nonlinear state estimation. A brief history on the EKF can be found in Ref. 13. The EKF is actual a "pseudo-linear" filter since it retains the linear update of the linear Kalman filter, as well as the linear covariance propagation. It only uses the original nonlinear function for the state propagation and definition of the output vector to form the residual.[14,15] The heart of the EKF lies in a first-order Taylor series expansion of the state and output models. Two approaches can be used for this expansion. The first expands a nonlinear function about a nominal (prescribed) trajectory, while the second expands about the current estimate. The advantage of the first approach is the filter gain can be computed offline. However, since the nominal trajectory is usually not as close to the truth as the current estimate in most applications, and with the advent of fast processors in modern-day computers, the second approach is mostly used in practice over the first. Even though the EKF is an approximate approach (at best), its use has found many applications, e.g. in inertial navigation,[16] and it does remarkably well.

Even with its wide acceptance, we still must remember that the EKF is merely a linearized approach. Many state estimation designers, including the present authors, have fallen into the fallacy that it can work well for any application encountered. But even early-day examples have shown that the first-order expansion approach in the EKF may not produce adequate state estimation results.[17] One obvious extension of the EKF involves a second-order expansion,[18] which provides improved performance at the expense of an increased computational burden due to the calculation of second derivatives. Other approaches are shown in Ref. 18 as well, such as the iterated EKF and a statistically linearized filter. Yet another approach that is rapidly gaining attention is based on a filter developed by Julier, Uhlmann and Durrant-Whyte.[19] This filter approach, which they call the *Unscented Filter*[20] (UF), has several advantages over the EKF, including: 1) the expected error is lower than the EKF, 2) the new filter can be applied to non-differentiable functions, 3) the new filter avoids the derivation of Jacobian matrices, and 4) the new filter is valid to higher-order expansions than the standard EKF. The UF works on the premise that with a fixed number of parameters it should be easier to approximate a Gaussian distribution than to approximate an arbitrary nonlinear function. Also, the UF uses the standard Kalman form in the post-update, but uses a different propagation of the covariance and pre measurement update with no local iterations.

The UF performance is generally equal to the performance of the second-order Kalman filter (SOKF) since its accuracy is good up to fourth-order moments.[21] The main advantage of the UF over a SOKF is that partials need not be computed. For simple problems this poses no difficulties, however for large scale problems, such as determining the position of a vehicle from magnetometer measurements,[22] these partials are generally analytically intractable. One approach to compute these partials is to use a simple numerical derivative. This approach only works well when these numerical derivatives are nearly as accurate as the analytical derivatives. The new complex-step derivative approximations are used in the SOKF in order to numerically compute these derivatives.

The second-order Kalman filter used here is called the *modified Gaussian second-order filter*. The algorithm is summarized in Table 4 for the discrete models, where $\mathbf{e}_i$ represents the i^{th} basis vector from the identity matrix of appropriate dimension, F_x and H_x are the Jacobian matrices of $\mathbf{f}(\mathbf{x})$ and $\mathbf{h}(\mathbf{x})$, respectively, and F_{xx}^i and H_{xx}^i are the i^{th} Hessian matrices of $\mathbf{f}(\mathbf{x})$ and $\mathbf{h}(\mathbf{x})$, respectively. All Jacobian and Hessian matrices are evaluated at the current state estimates. Notice the extra terms associated with these equations over the standard EKF. These are correction terms to compensate for "biases" that emerge from the nonlinearity in the models. If these biases are insignificant and negligible, the filter reduces to the standard EKF. The SOKF is especially attractive when the process and measurement noise are small compared to the bias correction terms. The only setback of this filter is the requirement of Hessian information, which is often challenging to analytically calculate, if not impossible, for many of today's complicated systems. These calculations are replaced with the complex-step Jacobian and hybrid Hessian approximations.

A. Application of Complex-Step Derivative Approximation

The example presented in this section was first proposed in Ref. 17 and has since become a standard performance evaluation example for various other nonlinear estimators.[23] In Ref. 23 a simpler implementation of this problem is proposed by using a coordinate transformation to reduce the computation load and implementation complexity. However, this coordinate transformation is problem specific and may not apply well to other nonlinear systems. In this section the original formulation is used and applied on a SOKF with the Jacobian and Hessian matrices obtained via both the numerical finite-difference and complex-step approaches. The performance is compared with the EKF, which uses the analytical Jacobian. The equations of motion of the system are given by

$$\dot{x}_1(t) = -x_2(t) \tag{44a}$$

$$\dot{x}_2(t) = -e^{-\alpha x_1(t)} x_2^2(t) x_3(t) \tag{44b}$$

$$\dot{x}_3(t) = 0 \tag{44c}$$

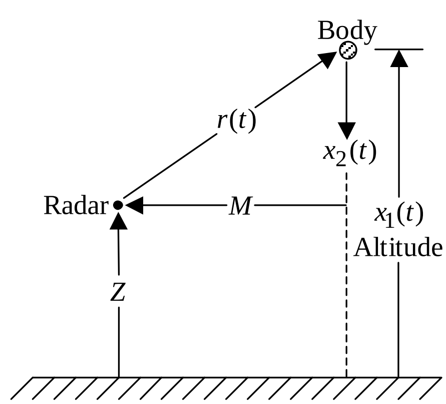


Figure 11. Vertically Falling Body Example

where $x_1(t)$ is the altitude, $x_2(t)$ is the downward velocity, $x_3(t)$ is the constant ballistic coefficient and $\alpha = 5 \times 10^{-5}$ is a constant that's relates air density with altitude. The range observation model is given by

$$\tilde{y}_k = \sqrt{M^2 + (x_{1,k} - Z)^2} + \nu_k \tag{45}$$

where ν_k is the observation noise, and M and Z are constants. These parameters are given by $M = 1 \times 10^5$ and $Z = 1 \times 10^5$. The variance of ν_k is given by 1×10^4.

The true state and initial estimates are given by

$$x_1(0) = 3 \times 10^5, \quad \hat{x}_1(0) = 3 \times 10^5 \tag{46}$$

$$x_2(0) = 2 \times 10^4, \quad \hat{x}_2(0) = 2 \times 10^4 \tag{47}$$

$$x_3(0) = 1 \times 10^{-3}, \quad \hat{x}_3(0) = 3 \times 10^{-5} \tag{48}$$

Clearly, an error is present in the ballistic coefficient value. Physically this corresponds to assuming that the body is "heavy" whereas in reality the body is "light." The initial covariance for all filters is given by $P(0) = \text{diag}[1 \times 10^6 \;\; 4 \times 10^6 \;\; 1 \times 10^{-4}]$. Measurements are sampled at 1-second intervals. Figure 12 shows the average position error, using a Monte-Carlo simulation of 10 runs, of the EKF with analytical derivatives, the SOKF with complex-step derivatives, and SOKF with finite-difference derivatives. The step-size, h, for all finite difference and complex-step operations is set to 1×10^{-4}. For all Monte-Carlo simulations the SOKF with finite-difference derivatives diverges using this step-size. For other step-sizes the SOKF with finite-difference derivatives does converge, but only within a narrow region of step-sizes (in this case only between 1.0 and 1×10^{-3}). The performance is never better than using the complex-step approach. From Figure 12 there is little difference among the EKF and SOKF with complex-step derivatives approaches before the first 12 seconds when the altitude is high. When the drag becomes significant at about 9 seconds, then these two filters exhibit large errors in position estimation. This coincides with the time when the falling body is on the same level as the radar, so the system becomes nearly unobservable. Eventually, the two filters demonstrate convergence with the EKF being the slowest. This is due to the deficiency of the EKF to capture the high nonlinearities present in the system. The SOKF with complex-step derivatives performs clearly better than the EKF. It should be noted that the SOKF with the complex-step derivative approximation performs equally well as using the analytical Jacobian and Hessian matrices for all values of h discussed here.

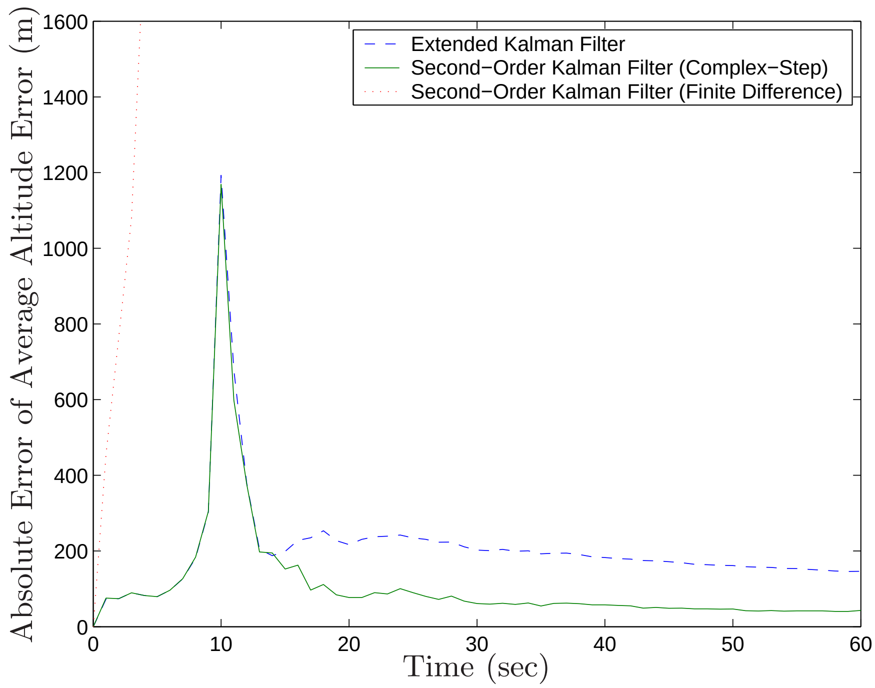


Figure 12. Absolute Mean Position Error

V. Conclusion

This paper demonstrated the ability of numerically obtaining derivative information via complex-step approximations. For the Jacobian case, unlike standard derivative approaches, more control in the accuracy of the standard complex-step approximation is provided since it does not succumb to roundoff errors for small step sizes. For the Hessian case, however, an arbitrarily small step-size cannot be chosen due to roundoff errors. Also, using the standard complex-step approach to approximate second derivatives was found to be less accurate than the numerical finite-difference obtained one. The accuracy was improved by deriving a number of new complex-step approximations for both first and second derivatives. These new approximations allow for high accuracy results in both the Jacobian and Hessian approximations by using the same function evaluations and step-sizes for both. The main advantage of the approach presented in this paper is that a "black box" can be employed to obtain the Jacobian or Hessian matrices for any vector function. The complex-step derivative approximations were used in a second-order Kalman filter formulation. Simulation results showed that this approach performed better than the extended Kalman filter and offers a wider range of accuracy than using a numerical finite-difference approximation.

References

[1] Lyness, J. N., "Numerical Algorithms Based on the Theory of Complex Variable," Proceedings - A.C.M. National Meeting, 1967, pp. 125–133.

[2] Greenberg, M. D., *Advanced Engineering Mathematics*, Prentice Hall, Upper Saddle River, New Jersey, 2nd ed., 1998.

[3] Lyness, J. N. and Moler, C. B., "Numerical Differentiation of Analytic Functions," *SIAM Journal for Numerical Analysis*, Vol. 4, No. 2, June 1967, pp. 202–210.

[4] Martins, J. R. R. A., Sturdza, P., and Alonso, J. J., "The Connection Between the Complex-Step Derivative Approximation and Algorithmic Differentiation," *AIAA Paper* 2001-0921, Jan. 2001.

[5] Kim, J., Bates, D. G., and Postlethwaite, I., "Complex-Step Gradient Approximation for Robustness Analysis of Nonlinear Systems," .

[6] Cerviño, L. I. and Bewley, T. R., "On the extension of the complex-step derivative technique to pseudospectral algorithms," *Journal of Computational Physics*, Vol. 187, No. 2, 2003, pp. 544–549.

[7] Squire, W. and Trapp, G., "Using Complex Variables to Estimate Derivatives of Read Functions," *SIAM Review*, Vol. 40, No. 1, Mar. 1998, pp. 110–112.

[8] Martins, J. R. R. A., Sturdza, P., and Alonso, J. J., "The Complex-Step Derivative Approximation," *ACM Transactions on Mathematical Software*, Vol. 29, No. 3, Sept. 2003, pp. 245–262.

[9] Pozrikidis, C., *Numerical Computation in Science and Engineering*, chap. 1, Oxford University Press., New York, NY, 1998, pp. 45–47.

[10] Turner, J. D., "Quaternion-Based Partial Derivative and State Transition Matrix Calculations for Design Optimization," *AIAA Paper* 2002-0447, 2002.

[11] Lai, K.-L., Crassidis, J. L., Cheng, Y., and Kim, J., "New Complex-Step Derivative Approximations with Application to Second-Order Kalman Filtering," *AIAA Paper* 2005-5944, 2005.

[12] Martins, J. R. R. A., Kroo, I. M., and Alonso, J. J., "An Automated Method for Sensitivity Analysis Using Complex Variables," *American Institute of Aeronautics and Astronautics*, 2000, AIAA-2000-0689.

[13] Jazwinski, A. H., *Stochastic Processes and Filtering theory*, Vol. 64 of *Mathematics in Science and Engineering*, Academic Press, New York, 1970.

[14] Stengel, R. F., *Optimal Control and Estimation*, Dover Publications, New York, NY, 1994.

[15] Crassidis, J. L. and Junkins, J. L., *Optimal Estimation of Dynamic System*, Chapman & Hall/CRC, Boca Raton, FL, 2004.

[16] Chatfield, A. B., *Fundamentals of High Accuracy Inertial Navigation*, chap. 10, American Institute of Aeronautics and Astronautics, Inc., Reston, VA, 1997.

[17] Athans, M., Wishner, R. P., and Bertolini, A., "Suboptimal State Estimation for Continuous-Time Nonlinear Systems from Discrete Noisy Measurements," *IEEE Transactions on Automatic Control*, Vol. 13, No. 5, Oct. 1968, pp. 504–514.

[18] Gelb, A., editor, *Applied Optimal Estimation*, The MIT Press, Cambridge, MA, 1974.

[19] Julier, S. J., Uhlmann, J. K., and Durrant-Whyte, H. F., "A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators," *IEEE Transactions on Automatic Control*, Vol. AC-45, No. 3, March 2000, pp. 477–482.

[20] Julier, S. J., Uhlmann, J. K., and Durrant-Whyte, H. F., "A New Approach for Filtering Nonlinear Systems," *American Control Conference*, Seattle, WA, June 1995, pp. 1628–1632.

[21] Wan, E. and van der Merwe, R., "The Unscented Kalman Filter," *Kalman Filtering and Neural Networks*, edited by S. Haykin, chap. 7, John Wiley & Sons, New York, NY, 2001.

[22] Psiaki, M. L., "Autonomous Orbit and Magnetic Field Determination Using Magnetometer and Star Sensing Data," *Journal of Guidance, Control, and Dynamics*, Vol. 18, No. 3, May-June 1995, pp. 584–592.

[23] Nam, K. and Tahk, M.-J., "A Second-Order Stochastic Filter Involving Coordonate Transformation," *IEEE Transactions on Automatic Control*, Vol. 44, No. 3, Mar. 1999, pp. 603–608.