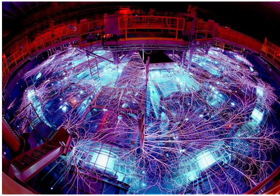
Sandia National Laboratories

SAND2016-5252PE



Derivative Calculations Using Hyper-Dual Numbers

Jeffrey A. Fike

Sandia National Laboratories

June 3, 2016

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2016-XXXX

Derivative Calculations Using Hyper-Dual Numbers

Hyper-Dual Numbers [Fike and Alonso 2011] are an extension of Dual Numbers [Study 1903], one type of Generalized Complex Number.

Ordinary Complex Numbers can be used to compute accurate first derivatives. [Martins, Kroo, and Alonso 2000 and Martins, Sturdza, and Alonso 2003]

- Dual Numbers can be used in a similar manner to produce *exact* first derivatives. [Piponi 2004, Leuck and Nagel 1999]

Hyper-Dual Numbers enable exact calculations of second (or higher) derivatives.

Outline

Derivative Calculations

Mathematical Properties of Hyper-Dual Numbers

Implementation and Use of Hyper-Dual Numbers

Other Details

First-Derivative Finite-Difference Formulas

Forward-difference (FD) Approximation:

$$\frac{\partial f(\mathbf{x})}{\partial x_j} = \frac{f(\mathbf{x} + h\mathbf{e}_j) - f(\mathbf{x})}{h} + \mathcal{O}(h)$$

Central-Difference (CD) approximation:

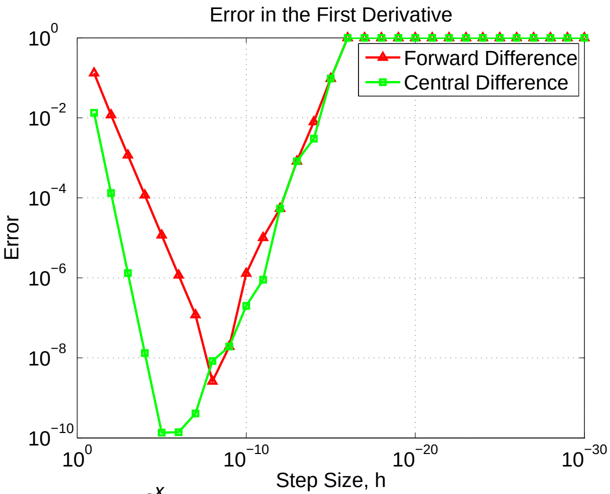
$$\frac{\partial f(\mathbf{x})}{\partial x_j} = \frac{f(\mathbf{x} + h\mathbf{e}_j) - f(\mathbf{x} - h\mathbf{e}_j)}{2h} + \mathcal{O}(h^2)$$

Subject to truncation error and subtractive cancellation error

- Truncation error is associated with the higher order terms that are ignored when forming the approximation.
- Subtractive cancellation error is a result of performing these calculations on a computer with finite precision.

Accuracy of Finite-Difference Calculations

Sandia National Laboratories



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

First-Derivative Complex-Step Approximation

Taylor series with an imaginary step:

$$f(x+hi) = f(x) + hf'(x)i - \frac{1}{2!}h^2 f''(x) - \frac{h^3 f'''(x)}{3!}i + \ldots$$

First-Derivative Complex-Step Approximation

Taylor series with an imaginary step:

$$f(x+hi) = f(x) + hf'(x)i - \frac{1}{2!}h^2 f''(x) - \frac{h^3 f'''(x)}{3!}i + \dots$$

$$f(x+hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2 f''(x) + \dots\right)}_{\text{real}} + \underbrace{h\left(f'(x) - \frac{1}{3!}h^2 f'''(x) + \dots\right)i}_{\text{imaginary}}$$

First-Derivative Complex-Step Approximation

Taylor series with an imaginary step:

$$f(x+hi) = f(x) + hf'(x)i - \frac{1}{2!}h^2 f''(x) - \frac{h^3 f'''(x)}{3!}i + \ldots$$

$$f(x+hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + h\underbrace{\left(f'(x) - \frac{1}{3!}h^2 f'''(x) + \ldots\right)i}_{\text{imaginary}}$$

First-Derivative Complex-Step Approximation: [Martins, Kroo, and Alonso 2000 and Martins, Sturdza, and Alonso 2003]

$$f'(x) = \frac{\text{Im}\left[f(x+hi)\right]}{h} + \mathcal{O}(h^2)$$

- First derivatives are subject to truncation error but are not subject to subtractive cancellation error.

Generalized Complex Numbers

Generalized Complex Numbers [Kantor 1989] consist of one real part and one non-real part, $a + bE$

Generalized Complex Numbers

Generalized Complex Numbers [Kantor 1989] consist of one real part and one non-real part, $a + bE$

Addition:

$$(a + bE) + (c + dE) = (a + c) + (b + d)\,E$$

Generalized Complex Numbers

Generalized Complex Numbers [Kantor 1989] consist of one real part and one non-real part, $a + bE$

Addition:

$$(a + bE) + (c + dE) = (a + c) + (b + d)\,E$$

Multiplication:

$$(a + bE)\,(c + dE) = ac + (ad + bc)\,E + bd\,E^2$$

Generalized Complex Numbers

Generalized Complex Numbers [Kantor 1989] consist of one real part and one non-real part, $a + bE$

Addition:

$$(a + bE) + (c + dE) = (a + c) + (b + d)\,E$$

Multiplication:

$$(a + bE)\,(c + dE) = ac + (ad + bc)\,E + bd\,E^2$$

Three types based on choice for the non-real part, E:

- Ordinary Complex Numbers $E^2 = i^2 = -1$
- Double Numbers $E^2 = e^2 = 1$ [Clifford 1873]
- Dual Numbers $E^2 = \epsilon^2 = 0$ [Study 1903]

Generalized Complex Numbers

Ordinary Complex Numbers ($E^2 = i^2 = -1$):

$$f(x+hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + h\underbrace{\left(f'(x) - \frac{1}{3!}h^2 f'''(x) + \ldots\right) i}_{\text{imaginary}}$$

Generalized Complex Numbers

Ordinary Complex Numbers ($E^2 = i^2 = -1$):

$$f(x+hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + h\underbrace{\left(f'(x) - \frac{1}{3!}h^2 f'''(x) + \ldots\right) i}_{\text{imaginary}}$$

Double Numbers ($E^2 = e^2 = 1$):

$$f(x+he) = \underbrace{\left(f(x) + \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + h\underbrace{\left(f'(x) + \frac{1}{3!}h^2 f'''(x) + \ldots\right) e}_{\text{non-real}}$$

Generalized Complex Numbers

Ordinary Complex Numbers ($E^2 = i^2 = -1$):

$$f(x+hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + h \underbrace{\left(f'(x) - \frac{1}{3!}h^2 f'''(x) + \ldots\right) i}_{\text{imaginary}}$$

Double Numbers ($E^2 = e^2 = 1$):

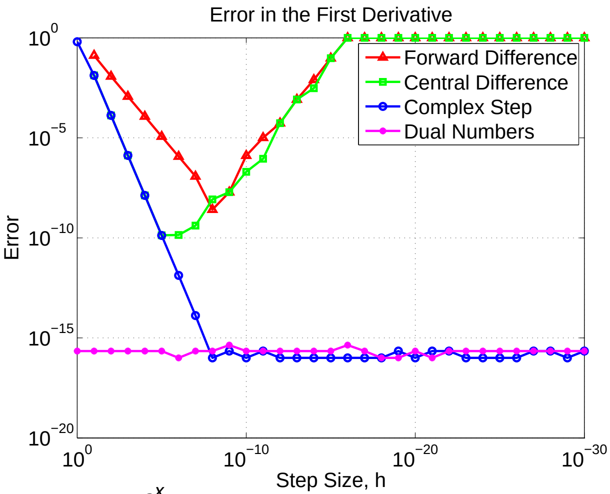
$$f(x+he) = \underbrace{\left(f(x) + \frac{1}{2!}h^2 f''(x) + \ldots\right)}_{\text{real}} + h \underbrace{\left(f'(x) + \frac{1}{3!}h^2 f'''(x) + \ldots\right) e}_{\text{non-real}}$$

Dual Numbers ($E^2 = \epsilon^2 = 0$):

$$f(x + h\epsilon) = \underbrace{f(x)}_{\text{real}} + \underbrace{hf'(x)\epsilon}_{\text{non-real}}$$

Unlimited Release

Accuracy of First-Derivative Calculations



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

Second-Derivative Calculations?

Sandia National Laboratories

Ordinary Complex Numbers ($E^2 = i^2 = -1$):

$$f(x+hi) = \underbrace{\left(f(x) - \frac{1}{2!}h^2 f''(x) + \dots\right)}_{\text{real}} + h\underbrace{\left(f'(x) - \frac{1}{3!}h^2 f'''(x) + \dots\right) i}_{\text{imaginary}}$$

Double Numbers ($E^2 = e^2 = 1$):

$$f(x+he) = \underbrace{\left(f(x) + \frac{1}{2!}h^2 f''(x) + \dots\right)}_{\text{real}} + h\underbrace{\left(f'(x) + \frac{1}{3!}h^2 f'''(x) + \dots\right) e}_{\text{non-real}}$$

Dual Numbers ($E^2 = \epsilon^2 = 0$):

$$f(x + h\epsilon) = \underbrace{f(x)}_{\text{real}} + \underbrace{hf'(x)\epsilon}_{\text{non-real}}$$

Second-Derivative Complex-Step

One Second-Derivative Complex-Step Approximation:

$$f''(x) = \frac{2\left(f(x) - \mathrm{Re}[f(x+ih)]\right)}{h^2} + \mathcal{O}(h^2)$$

- Second derivatives are subject to subtractive cancellation error

Second-Derivative Complex-Step

One Second-Derivative Complex-Step Approximation:

$$f''(x) = \frac{2\left(f(x) - \mathrm{Re}[f(x+ih)]\right)}{h^2} + \mathcal{O}(h^2)$$

- Second derivatives are subject to subtractive cancellation error

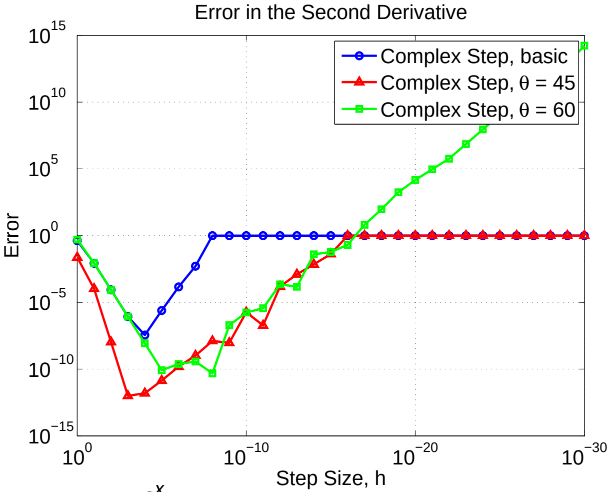
Alternative approximations: [Lai 2008]

$$f''(x) = \frac{\mathrm{Im}\left[f(x+i^{1/2}h) + f(x+i^{5/2}h)\right]}{h^2} + \mathcal{O}(h^4) \;:\; \theta = 45^\circ$$

$$f''(x) = \frac{2\,\mathrm{Im}\left[f(x+i^{2/3}h) + f(x+i^{8/3}h)\right]}{\sqrt{3}h^2} + \mathcal{O}(h^2) \;:\; \theta = 60^\circ$$

- These alternatives may offer improvements, but they are still subject to subtractive cancellation error

Alternative Complex-Step Approximations



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

Multiple Non-Real Parts

To avoid subtractive cancellation error:

- Second-derivative term should be the **leading term** of a non-real part
- First-derivative is already the leading term of a non-real part

Suggests that we need a number with **multiple non-real parts**

- Use higher-dimensional extensions of generalized complex numbers

Quaternions

Quaternions: one real part and three non-real parts

$$\begin{aligned} i^2 = j^2 = k^2 &= -1 \\ ijk &= -1 \end{aligned}$$

Quaternions

Quaternions: one real part and three non-real parts

$$\begin{aligned} i^2 = j^2 = k^2 &= -1 \\ ijk &= -1 \end{aligned}$$

Taylor series for a generic step, *d*:

$$f(x+d) = f(x) + df'(x) + \frac{1}{2!}d^2 f''(x) + \frac{1}{3!}d^3 f'''(x) + \ldots$$

Quaternions

Quaternions: one real part and three non-real parts

$$\begin{aligned} i^2 = j^2 = k^2 &= -1 \\ ijk &= -1 \end{aligned}$$

Taylor series for a generic step, *d*:

$$f(x+d) = f(x) + df'(x) + \frac{1}{2!}d^2 f''(x) + \frac{1}{3!}d^3 f'''(x) + \ldots$$

For a quaternion step:

$$\begin{aligned} d &= h_1 i + h_2 j + 0k \\ d^2 &= -\left(h_1^2 + h_2^2\right) \end{aligned}$$

- d^2 is real, second derivative only appears in the real part

Quaternions

Sandia National Laboratories

Second-Derivative Quaternion-Step Approximation:

$$f''(x) = \frac{2\left(f(x) - \mathrm{Re}[f(x + h_1 i + h_2 j + 0k)]\right)}{h_1^2 + h_2^2} + \mathcal{O}(h_1^2 + h_2^2)$$

- Subject to subtractive-cancellation error

Quaternion multiplication is not commutative, $ij = k$ but $ji = -k$

Quaternions

Second-Derivative Quaternion-Step Approximation:

$$f''(x) = \frac{2\left(f(x) - \mathrm{Re}[f(x + h_1 i + h_2 j + 0k)]\right)}{h_1^2 + h_2^2} + \mathcal{O}(h_1^2 + h_2^2)$$

- Subject to subtractive-cancellation error

Quaternion multiplication is not commutative, $ij = k$ but $ji = -k$

Instead, consider a number with three non-real components E_1, E_2, and $(E_1 E_2)$ where multiplication is commutative, i.e. $E_1 E_2 = E_2 E_1$

Enforce Multiplication to be Commutative

Taylor series:

$$f(x+d) = f(x) + df'(x) + \frac{1}{2!}d^2 f''(x) + \frac{1}{3!}d^3 f'''(x) + \ldots$$

Enforce Multiplication to be Commutative

Taylor series:

$$f(x+d) = f(x) + df'(x) + \frac{1}{2!}d^2 f''(x) + \frac{1}{3!}d^3 f'''(x) + \ldots$$

$$\begin{aligned}
d &= h_1 E_1 + h_2 E_2 + 0 E_1 E_2 \\
d^2 &= h_1^2 E_1^2 + h_2^2 E_2^2 + 2 h_1 h_2 E_1 E_2 \\
d^3 &= h_1^3 E_1^3 + 3 h_1 h_2^2 E_1 E_2^2 + 3 h_1^2 h_2 E_1^2 E_2 + h_2^3 E_2^3 \\
d^4 &= h_1^4 E_1^4 + 6 h_1^2 h_2^2 E_1^2 E_2^2 + 4 h_1^3 h_2 E_1^3 E_2 + 4 h_1 h_2^3 E_1 E_2^3 + h_2^4 E_2^4
\end{aligned}$$

Unlimited Release

Enforce Multiplication to be Commutative

Taylor series:

$$f(x+d) = f(x) + df'(x) + \frac{1}{2!}d^2 f''(x) + \frac{1}{3!}d^3 f'''(x) + \ldots$$

$$\begin{aligned}
d &= h_1 E_1 + h_2 E_2 + 0 E_1 E_2 \\
d^2 &= h_1^2 E_1^2 + h_2^2 E_2^2 + 2 h_1 h_2 E_1 E_2 \\
d^3 &= h_1^3 E_1^3 + 3 h_1 h_2^2 E_1 E_2^2 + 3 h_1^2 h_2 E_1^2 E_2 + h_2^3 E_2^3 \\
d^4 &= h_1^4 E_1^4 + 6 h_1^2 h_2^2 E_1^2 E_2^2 + 4 h_1^3 h_2 E_1^3 E_2 + 4 h_1 h_2^3 E_1 E_2^3 + h_2^4 E_2^4
\end{aligned}$$

- d^2 is first term with a non-zero $(E_1 E_2)$ component
- Second derivative is the leading term of the $(E_1 E_2)$ part
- As long as multiplication is commutative, and $E_1 E_2 \neq 0$, second-derivative approximations can be formed that are not subject to subtractive-cancellation error

Several Possible Number Systems

The requirement that $E_1E_2 = E_2E_1$ produces the constraint:

$$(E_1E_2)^2 = E_1E_2E_1E_2 = E_1E_1E_2E_2 = E_1^2E_2^2$$

Several Possible Number Systems

The requirement that $E_1 E_2 = E_2 E_1$ produces the constraint:

$$(E_1 E_2)^2 = E_1 E_2 E_1 E_2 = E_1 E_1 E_2 E_2 = E_1^2 E_2^2$$

This leaves many possibilities for the definitions of E_1 and E_2:

Several Possible Number Systems

The requirement that $E_1 E_2 = E_2 E_1$ produces the constraint:

$$(E_1 E_2)^2 = E_1 E_2 E_1 E_2 = E_1 E_1 E_2 E_2 = E_1^2 E_2^2$$

This leaves many possibilities for the definitions of E_1 and E_2:

- $E_1^2 = E_2^2 = -1$ which results in $(E_1 E_2)^2 = 1$
  - Circular-Fourcomplex Numbers [Olariu 2002]
  - Multicomplex Numbers [Price 1991]

Several Possible Number Systems

The requirement that $E_1 E_2 = E_2 E_1$ produces the constraint:

$$(E_1 E_2)^2 = E_1 E_2 E_1 E_2 = E_1 E_1 E_2 E_2 = E_1^2 E_2^2$$

This leaves many possibilities for the definitions of E_1 and E_2:

- $E_1^2 = E_2^2 = -1$ which results in $(E_1 E_2)^2 = 1$
  - Circular-Fourcomplex Numbers [Olariu 2002]
  - Multicomplex Numbers [Price 1991]
- Constrain $E_1^2 = E_2^2 = (E_1 E_2)^2$
  - $E_1^2 = E_2^2 = (E_1 E_2)^2 = 1$ Hyper-Double Numbers [Fike 2012]
  - $E_1^2 = E_2^2 = (E_1 E_2)^2 = 0$ Hyper-Dual Numbers [Fike 2011]

Several Possible Number Systems

Sandia National Laboratories

The requirement that $E_1E_2 = E_2E_1$ produces the constraint:

$$(E_1E_2)^2 = E_1E_2E_1E_2 = E_1E_1E_2E_2 = E_1^2E_2^2$$

This leaves many possibilities for the definitions of E_1 and E_2:

- $E_1^2 = E_2^2 = -1$ which results in $(E_1E_2)^2 = 1$
  - Circular-Fourcomplex Numbers [Olariu 2002]
  - Multicomplex Numbers [Price 1991]
- Constrain $E_1^2 = E_2^2 = (E_1E_2)^2$
  - $E_1^2 = E_2^2 = (E_1E_2)^2 = 1$ Hyper-Double Numbers [Fike 2012]
  - $E_1^2 = E_2^2 = (E_1E_2)^2 = 0$ Hyper-Dual Numbers [Fike 2011]

All are free from subtractive-cancellation error

- Truncation error can be reduced below machine precision
- Effectively exact

Hyper-Dual Numbers

Hyper-dual numbers have one real part and three non-real parts:

$$a = a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2$$

$$\epsilon_1^2 = \epsilon_2^2 = 0$$

$$\epsilon_1 \neq \epsilon_2 \neq 0$$

$$\epsilon_1\epsilon_2 = \epsilon_2\epsilon_1 \neq 0$$

Hyper-Dual Numbers

Hyper-dual numbers have one real part and three non-real parts:

$$a = a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2$$

$$\epsilon_1^2 = \epsilon_2^2 = 0$$

$$\epsilon_1 \neq \epsilon_2 \neq 0$$

$$\epsilon_1\epsilon_2 = \epsilon_2\epsilon_1 \neq 0$$

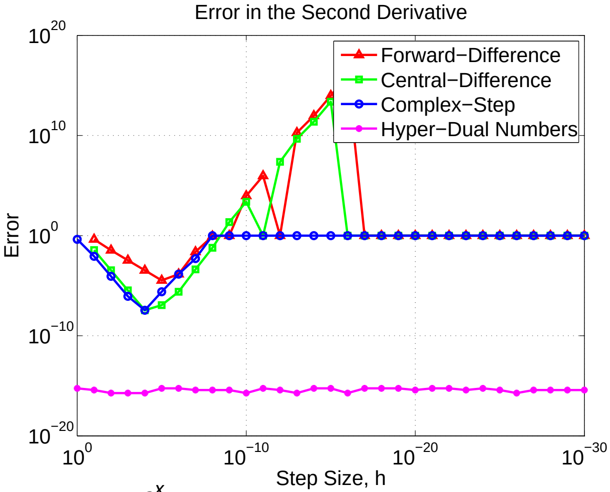
Taylor series truncates exactly at second-derivative term:

$$f(x+h_1\epsilon_1+h_2\epsilon_2+0\epsilon_1\epsilon_2) = f(x)+h_1f'(x)\epsilon_1+h_2f'(x)\epsilon_2+h_1h_2f''(x)\epsilon_1\epsilon_2$$

- No truncation error and no subtractive-cancellation error
- Lack of higher order terms makes implementation easier

[Fike 2011 and Fike 2012]

Accuracy of Second-Derivative Calculations



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

Using Hyper-Dual Numbers

Evaluate a function with a hyper-dual step:

$$f(\mathbf{x} + h_1\epsilon_1\mathbf{e}_i + h_2\epsilon_2\mathbf{e}_j + \mathbf{0}\epsilon_1\epsilon_2)$$

Derivative information can be found by examining the non-real parts:

$$\frac{\partial f(\mathbf{x})}{\partial x_i} = \frac{\epsilon_1 part\left[f(\mathbf{x} + h_1\epsilon_1\mathbf{e}_i + h_2\epsilon_2\mathbf{e}_j + \mathbf{0}\epsilon_1\epsilon_2)\right]}{h_1}$$

$$\frac{\partial f(\mathbf{x})}{\partial x_j} = \frac{\epsilon_2 part\left[f(\mathbf{x} + h_1\epsilon_1\mathbf{e}_i + h_2\epsilon_2\mathbf{e}_j + \mathbf{0}\epsilon_1\epsilon_2)\right]}{h_2}$$

$$\frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} = \frac{\epsilon_1\epsilon_2 part\left[f(\mathbf{x} + h_1\epsilon_1\mathbf{e}_i + h_2\epsilon_2\mathbf{e}_j + \mathbf{0}\epsilon_1\epsilon_2)\right]}{h_1 h_2}$$

Outline

Derivative Calculations

Mathematical Properties of Hyper-Dual Numbers

Implementation and Use of Hyper-Dual Numbers

Other Details

Arithmetic Operations

Consider two Hyper-Dual Numbers:

$$a = a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2 \qquad b = b_0 + b_1\epsilon_1 + b_2\epsilon_2 + b_3\epsilon_1\epsilon_2$$

Arithmetic Operations

Sandia National Laboratories

Consider two Hyper-Dual Numbers:

$$a = a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2 \qquad b = b_0 + b_1\epsilon_1 + b_2\epsilon_2 + b_3\epsilon_1\epsilon_2$$

Addition:

$$a + b = (a_0 + b_0) + (a_1 + b_1)\,\epsilon_1 + (a_2 + b_2)\,\epsilon_2 + (a_3 + b_3)\,\epsilon_1\epsilon_2$$

Arithmetic Operations

Consider two Hyper-Dual Numbers:

$$a = a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2 \qquad b = b_0 + b_1\epsilon_1 + b_2\epsilon_2 + b_3\epsilon_1\epsilon_2$$

Addition:

$$a + b = (a_0 + b_0) + (a_1 + b_1)\,\epsilon_1 + (a_2 + b_2)\,\epsilon_2 + (a_3 + b_3)\,\epsilon_1\epsilon_2$$

Multiplication:

$$\begin{aligned} a * b \quad = \quad & (a_0 * b_0) + (a_0 * b_1 + a_1 * b_0)\,\epsilon_1 + (a_0 * b_2 + a_2 * b_0)\,\epsilon_2 \\ & + (a_0 * b_3 + a_1 * b_2 + a_2 * b_1 + a_3 * b_0)\,\epsilon_1\epsilon_2 \end{aligned}$$

Arithmetic Operations

Consider two Hyper-Dual Numbers:

$$a = a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2 \qquad b = b_0 + b_1\epsilon_1 + b_2\epsilon_2 + b_3\epsilon_1\epsilon_2$$

Addition:

$$a + b = (a_0 + b_0) + (a_1 + b_1)\,\epsilon_1 + (a_2 + b_2)\,\epsilon_2 + (a_3 + b_3)\,\epsilon_1\epsilon_2$$

Multiplication:

$$\begin{aligned} a * b \;&=\; (a_0 * b_0) + (a_0 * b_1 + a_1 * b_0)\,\epsilon_1 + (a_0 * b_2 + a_2 * b_0)\,\epsilon_2 \\ &\quad + (a_0 * b_3 + a_1 * b_2 + a_2 * b_1 + a_3 * b_0)\,\epsilon_1\epsilon_2 \end{aligned}$$

- Hyper-Dual addition: 4 real additions
- Hyper-Dual multiplication: 9 real multiplications and 5 additions

Other Operations

The inverse:

$$\frac{1}{a} = \frac{1}{a_0} - \frac{a_1}{a_0^2}\epsilon_1 - \frac{a_2}{a_0^2}\epsilon_2 - \left(\frac{2a_1a_2}{a_0^3} - \frac{a_3}{a_0^2}\right)\epsilon_1\epsilon_2$$

- Only exists for $a_0 \neq 0$

This suggests a definition for the norm:

$$norm\left(a\right) = \sqrt{a_0^2}$$

This in turn implies that comparisons should only be made based on the real part.

- i.e. $a > b$ is equivalent to $a_0 > b_0$
- This allows the code to follow the same execution path as the real-valued code.

Mathematical Properties of Hyper-Dual Numbers

Sandia National Laboratories

- Additive associativity, i.e. $(a+b)+c=a+(b+c)$,
- Additive commutativity, i.e. $a+b=b+a$,
- Additive identity, there exists a zero element, $z=0+0\epsilon_1+0\epsilon_2+0\epsilon_1\epsilon_2$, such that $a+z=z+a=a$,
- Additive inverse, i.e. $a+(-a)=(-a)+a=0$,
- Multiplicative associativity, i.e. $(a*b)*c=a*(b*c)$,
- Multiplicative commutativity, i.e. $a*b=b*a$,
- Multiplicative identity, there exists a unitary element, $1+0\epsilon_1+0\epsilon_2+0\epsilon_1\epsilon_2$, such that $a*1=1*a=a$,
- Left and right distributivity, i.e. $a*(b+c)=(a*b)+(a*c)$ and $(b+c)*a=(b*a)+(c*a)$.

These properties make hyper-dual numbers a commutative unital associative algebra.

Mathematical Properties of Hyper-Dual Numbers

Hyper-Dual Numbers are a commutative unital associative algebra.

Hyper-Dual Numbers are not a field (a commutative division algebra)

A division algebra requires the properties on the previous slide, plus a multiplicative inverse

- i.e. there exists an inverse, a^{-1}, such that $a * a^{-1} = a^{-1} * a = 1$ for every $a \neq 0 + 0\epsilon_1 + 0\epsilon_2 + 0\epsilon_1\epsilon_2$

Hyper-Dual Numbers have an inverse for every a with $norm(a) \neq 0$ (i.e. $a_0 \neq 0$)

Hyper-Dual Functions

Sandia National Laboratories

Differentiable functions can be defined using the Taylor series for a generic hyper-dual number:

$$f(a) = f(a_0) + a_1 f'(a_0)\epsilon_1 + a_2 f'(a_0)\epsilon_2 + \left(a_3 f'(a_0) + a_1 a_2 f''(a_0)\right)\epsilon_1\epsilon_2$$

For instance:

$$a^3 = a_0^3 + 3a_1a_0^2\epsilon_1 + 3a_2a_0^2\epsilon_2 + \left(3a_3a_0^2 + 6a_1a_2a_0\right)\epsilon_1\epsilon_2$$

$$\begin{aligned}\sin a &= \sin a_0 + a_1\cos a_0\epsilon_1 + a_2\cos a_0\epsilon_2\\ &\quad + \left(a_3\cos a_0 - a_1a_2\sin a_0\right)\epsilon_1\epsilon_2\end{aligned}$$

Example Evaluation

A simple example hyper-dual function evaluation:

$$f(x) = \sin^3 x$$

This function can be evaluated as:

$$\begin{aligned} t_0 &= x \\ t_1 &= \sin t_0 \\ t_2 &= t_1^3 \end{aligned}$$

Example Evaluation

A simple example hyper-dual function evaluation:

$$f(x) = \sin^3 x$$

This function can be evaluated as:

$$\begin{aligned}
t_0 &= x + h_1\epsilon_1 + h_2\epsilon_2 + 0\epsilon_1\epsilon_2 \\
t_1 &= \sin t_0 \\
&= \sin x + h_1 \cos x\epsilon_1 + h_2 \cos x\epsilon_2 - h_1 h_2 \sin x\epsilon_1\epsilon_2 \\
t_2 &= t_1^3 \\
&= \sin^3 x + 3h_1 \cos x \sin^2 x\epsilon_1 + 3h_2 \cos x \sin^2 x\epsilon_2 \\
&\quad -\frac{3}{4} h_1 h_2 \left(\sin x - 3 \sin 3x\right) \epsilon_1\epsilon_2
\end{aligned}$$

Outline

Derivative Calculations

Mathematical Properties of Hyper-Dual Numbers

Implementation and Use of Hyper-Dual Numbers

Other Details

Hyper-Dual Number Implementation

To use hyper-dual numbers, every operation in an analysis code must be modified to operate on hyper-dual numbers instead of real numbers

- Basic Arithmetic Operations: Addition, Multiplication, etc.
- Logical Comparison Operators: $\geq$, $\neq$, etc.
- Mathematical Functions: exponential, logarithm, sine, absolute value, etc.
- Input/Output Functions to write and display hyper-dual numbers

Hyper-dual numbers are implemented as a class using operator overloading in C++, CUDA, MATLAB and Fortran

- Change variable types, but body and structure of code is unaltered
- MPI datatype and reduction operations also implemented
- Implementations publicly available: `http://adl.stanford.edu/hyperdual`
- Implementations by others for Python and Julia

Variations of Hyper-Dual Numbers

Dual numbers produce exact first derivatives

Hyper-dual numbers, as described so far, produce exact second-derivatives

Third (or higher) derivatives can be computed by including additional non-real parts

- Third derivatives require an ϵ_3 term and its combinations

  $d = h_1\epsilon_1 + h_2\epsilon_2 + h_3\epsilon_3 + 0\epsilon_1\epsilon_2 + 0\epsilon_1\epsilon_3 + 0\epsilon_2\epsilon_3 + 0\epsilon_1\epsilon_2\epsilon_3$

Derivatives of complex-valued functions can be computed by defining hyper-dual numbers with complex-valued components

Vector-mode version propagates entire gradient and Hessian

- Eliminates redundant calculations, but increased memory requirements [Fike 2012]

Analysis Codes Using Hyper-Dual Numbers

Hyper-Dual Numbers can be applied to codes of arbitrary complexity in order to compute exact derivatives of output quantities of interest with respect to input parameters.

- Computational Fluid Dynamics
  - JOE, a parallel unstructured, 3-D, unsteady Reynolds-averaged Navier-Stokes code developed at Stanford University as part of PSAAP (the Department of Energy's Predictive Science Academic Alliance Program)
- Structural Dynamics
  - Sierra/SD (aka Salinas), a massively parallel, high-fidelity, structural dynamics finite element analysis code developed by Sandia National Laboratories

Converting Codes to Use Hyper-Dual Numbers

At a high level, converting a code to use Hyper-Dual Numbers requires little more than changing the variables types from real numbers to hyper-dual numbers.

- In some cases, there can be more effort required
- Requires modifying the source code
- Some codes make use of external libraries for which the source code is unavailable
  - Linear Solvers
  - Eigenvalue Solvers

Converting Codes to Use Hyper-Dual Numbers

At a high level, converting a code to use Hyper-Dual Numbers requires little more than changing the variables types from real numbers to hyper-dual numbers.

- In some cases, there can be more effort required
- Requires modifying the source code
- Some codes make use of external libraries for which the source code is unavailable
  - Linear Solvers
  - Eigenvalue Solvers

Hyper-Dual numbers can still be used to compute derivatives even if not all parts of a code can be modified

- Requires replicating the effect of a hyper-dual calculation, i.e. returning hyper-dual valued output containing the required derivative information

Differentiating the Solution of a Linear System

Solving the system:

$$\mathbf{A}(\mathbf{x})\mathbf{y}(\mathbf{x}) = \mathbf{b}(\mathbf{x})$$

Differentiating both sides with respect to the i^{th} component of **x** gives

$$\frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i}\mathbf{y}(\mathbf{x}) + \mathbf{A}(\mathbf{x})\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_i} = \frac{\partial \mathbf{b}(\mathbf{x})}{\partial x_i}$$

Differentiating this result with respect to the j^{th} component of **x** gives

$$\frac{\partial^2 \mathbf{A}(\mathbf{x})}{\partial x_j \partial x_i}\mathbf{y}(\mathbf{x}) + \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i}\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_j} + \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_j}\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_i} + \mathbf{A}(\mathbf{x})\frac{\partial^2 \mathbf{y}(\mathbf{x})}{\partial x_j \partial x_i} = \frac{\partial^2 \mathbf{b}(\mathbf{x})}{\partial x_j \partial x_i}$$

Differentiating the Solution of a Linear System

This can be solved as:

$$\begin{bmatrix} \mathbf{A}(\mathbf{x}) & 0 & 0 & 0 \\ \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i} & \mathbf{A}(\mathbf{x}) & 0 & 0 \\ \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_j} & 0 & \mathbf{A}(\mathbf{x}) & 0 \\ \frac{\partial^2 \mathbf{A}(\mathbf{x})}{\partial x_j \partial x_i} & \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_j} & \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i} & \mathbf{A}(\mathbf{x}) \end{bmatrix} \begin{Bmatrix} \mathbf{y}(\mathbf{x}) \\ \frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_i} \\ \frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_j} \\ \frac{\partial^2 \mathbf{y}(\mathbf{x})}{\partial x_j \partial x_i} \end{Bmatrix} = \begin{Bmatrix} \mathbf{b}(\mathbf{x}) \\ \frac{\partial \mathbf{b}(\mathbf{x})}{\partial x_i} \\ \frac{\partial \mathbf{b}(\mathbf{x})}{\partial x_j} \\ \frac{\partial^2 \mathbf{b}(\mathbf{x})}{\partial x_j \partial x_i} \end{Bmatrix}$$

Or

$$\mathbf{A}(\mathbf{x})\mathbf{y}(\mathbf{x}) = \mathbf{b}(\mathbf{x})$$

$$\mathbf{A}(\mathbf{x})\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_i} = \frac{\partial \mathbf{b}(\mathbf{x})}{\partial x_i} - \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i}\mathbf{y}(\mathbf{x})$$

$$\mathbf{A}(\mathbf{x})\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_j} = \frac{\partial \mathbf{b}(\mathbf{x})}{\partial x_j} - \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_j}\mathbf{y}(\mathbf{x})$$

$$\mathbf{A}(\mathbf{x})\frac{\partial^2 \mathbf{y}(\mathbf{x})}{\partial x_j \partial x_i} = \frac{\partial^2 \mathbf{b}(\mathbf{x})}{\partial x_j \partial x_i} - \frac{\partial^2 \mathbf{A}(\mathbf{x})}{\partial x_j \partial x_i}\mathbf{y}(\mathbf{x}) - \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_i}\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_j} - \frac{\partial \mathbf{A}(\mathbf{x})}{\partial x_j}\frac{\partial \mathbf{y}(\mathbf{x})}{\partial x_i}$$

Derivatives of Eigenvalues and Eigenvectors

Eigenvalues and eigenvectors are solutions of the equation

$$(K - \lambda_\ell M)\,\phi_\ell = F_\ell \phi_\ell = 0$$

The first derivative of an eigenvalue is

$$\frac{\partial \lambda_\ell}{\partial x_i} = \phi_\ell^T \left(\frac{\partial K}{\partial x_i} - \lambda_\ell \frac{\partial M}{\partial x_i} \right) \phi_\ell$$

The first derivative of the eigenvector is

$$\frac{\partial \phi_\ell}{\partial x_i} = z_i + c_i \phi_\ell$$

where

$$F_\ell z_i = -\frac{\partial F_\ell}{\partial x_i} \phi_\ell$$

and

$$c_i = -\frac{1}{2} \phi_\ell^T \frac{\partial M}{\partial x_i} \phi_\ell - \phi_\ell^T M z_i$$

Outline

Derivative Calculations

Mathematical Properties of Hyper-Dual Numbers

Implementation and Use of Hyper-Dual Numbers

Other Details

Matrix Representation of Generalized Complex Numbers

Ordinary Complex Numbers:

$$a + bi = \begin{bmatrix} a & -b \\ b & a \end{bmatrix} = a \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + b \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

Double Numbers:

$$a + be = \begin{bmatrix} a & b \\ b & a \end{bmatrix} = a \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + b \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Dual Numbers:

$$a + b\epsilon = \begin{bmatrix} a & 0 \\ b & a \end{bmatrix} = a \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + b \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$$

Matrix Representation of Hyper-Dual Numbers

Ordinary Complex Numbers:

$$a_0 + a_1\epsilon_1 + a_2\epsilon_2 + a_3\epsilon_1\epsilon_2 = \begin{bmatrix} a_0 & 0 & 0 & 0 \\ a_1 & a_0 & 0 & 0 \\ a_2 & 0 & a_0 & 0 \\ a_3 & a_2 & a_1 & a_0 \end{bmatrix}$$

Questions?