# Engineering Sketch Pad (ESP)

# Training Session 9
# Sensitivities

**John F. Dannenhoffer, III**
jfdannen@syr.edu
Syracuse University

**Bob Haimes**
haimes@mit.edu
Massachusetts Institute of Technology

updated for v1.18

# Overview

- Background / Objective
- Alternative approaches
  - analytic derivatives
  - code differentiation
  - finite differences
- Computed examples
- Application to grid generation
- Conclusions
- Computing sensitivities in `ESP`
- Homework exercises

# Background/ Objective

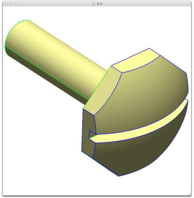
- Background
  - MDAO environments require calculation of sensitivity of objective function(s) w.r.t. the design parameters
  - Many modern CFD systems can produce the objective function sensitivity all the way back to the grid
  - Little work has been done in calculating the sensitivity of the grid w.r.t. the design parameters
- Objective
  - Compute sensitivities directly on parametric, CAD-based geometries

# Possible Approaches

- Analytic derivatives
  - differentiate all operations within the CAD system analytically
  - requires access to CAD system's algorithms
  - produces results that are not susceptible to truncation error
- Code differentiation
  - CAD system source code is automatically differentiated via compiler-like process
  - requires access to CAD system's source code
  - produces results that are not susceptible to truncation error
- Finite differences
  - multiple instances of the configuration are generated and the sensitivities are computed via finite differences
  - requires one to find corresponding points in the configurations
  - picking appropriate step size (or perturbation) requires a trade-off between truncation and round-off errors

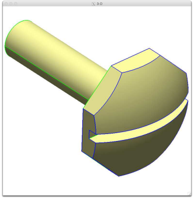
# Review of Construction Process (1)

```
     # bolt example

     # design parameters
 1:  DESPMTR   Thead     1.00   # thickness of head
 2:  DESPMTR   Whead     3.00   # width     of head
 3:  DESPMTR   Fhead     0.50   # fraction  of head that is flat

 4:  DESPMTR   Dslot     0.75   # depth of slot
 5:  DESPMTR   Wslot     0.25   # width of slot

 6:  DESPMTR   Lshaft    4.00   # length   of shaft
 7:  DESPMTR   Dshaft    1.00   # diameter of shaft

 8:  DESPMTR   sfact     0.50   # overall scale factor

     # head
 9:  BOX       0      -Whead/2  -Whead/2  Thead   Whead     Whead
10:  ROTATEX   90  0  0
11:  BOX       0      -Whead/2  -Whead/2  Thead   Whead     Whead
12:  ROTATEX   45  0  0
13:  INTERSECT

...
```

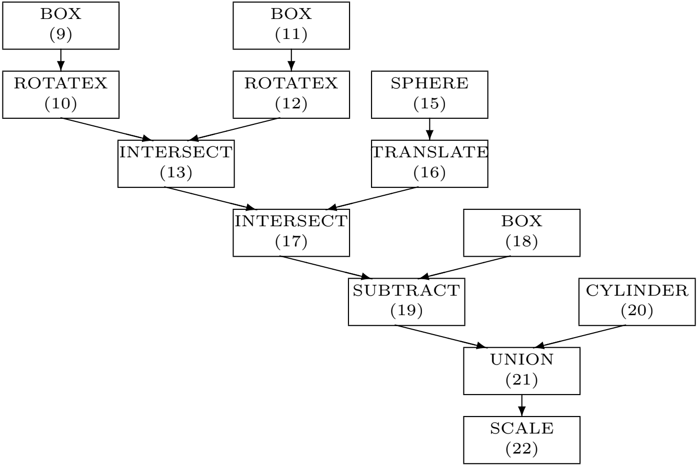


# Review of Construction Process (2)

```
...

SET       Rhead  (Whead^2/4+(1-Fhead)^2*Thead^2)/(2*Thead*(1-Fhead))

SPHERE    0            0   0    Rhead
TRANSLATE Thead-Rhead  0   0
INTERSECT

# slot
BOX       Thead-Dslot  -Wslot/2   -Whead   2*Thead   Wslot    2*Whead
SUBTRACT

# shaft
CYLINDER  -Lshaft   0   0   0   0   0   Dshaft/2
UNION

SCALE     sfact

END
```



# Review of Construction Process (3)



# Analytical Sensitivities

- Differentiate expressions for arguments to various operators
- For each Face
  - determine primitive that created the Face
  - differentiate functions used to generate the Face in its original position
  - apply appropriate transformations to sensitivities
- For each Edge
  - compute sensitivities of adjacent Faces
  - find sensitivity that is consistent with them and whose component along the Edge vanishes
- For each Node
  - compute sensitivities of incident Edges
  - find sensitivity that is consistent with them

# Analytical Sensitivity for Faces

- Differentiate function(s) used to create a point on the Face
  - for a box

$$\left(\frac{\partial \vec{x}}{\partial P}\right)_{\text{prim}} = \frac{\partial \overrightarrow{x_0}}{\partial P} + \frac{\partial \vec{S}}{\partial P}\left(\frac{\vec{x}_{\text{prim}} - \overrightarrow{x_0}}{\vec{S}}\right)$$

- Modify the sensitivities based upon transformations traversed in the feature tree
  - for a translation

$$\left(\frac{\partial \vec{x}}{\partial P}\right)_{\text{new}} = \left(\frac{\partial \vec{x}}{\partial P}\right)_{\text{prim}} + \frac{d\overrightarrow{x_0}}{dP}$$

- Take normal component

$$\frac{\partial w}{\partial P} \equiv \frac{\partial \vec{x}}{\partial P} \bullet \vec{n}$$

# Analytical Sensitivity for Edges and Nodes

- Edge sensitivity is consistent with the adjacent Face sensitivities

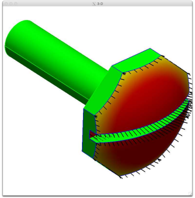
$$\begin{bmatrix} n_{x,\text{left}} & n_{y,\text{left}} & n_{z,\text{left}} \\ n_{x,\text{right}} & n_{y,\text{right}} & n_{z,\text{right}} \\ t_{x,\text{edge}} & t_{y,\text{edge}} & t_{z,\text{edge}} \end{bmatrix} \begin{bmatrix} (\partial x/\partial P)_{\text{edge}} \\ (\partial y/\partial P)_{\text{edge}} \\ (\partial z/\partial P)_{\text{edge}} \end{bmatrix} = \begin{bmatrix} (\partial w/\partial P)_{\text{left}} \\ (\partial w/\partial P)_{\text{right}} \\ 0 \end{bmatrix}$$

- Node sensitivity is consistent with the incident Edge sensitivities

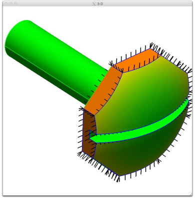
$$\begin{bmatrix} \vec{t}_1 \bullet \vec{t}_1 & -\vec{t}_1 \bullet \vec{t}_2 \\ -\vec{t}_1 \bullet \vec{t}_2 & \vec{t}_2 \bullet \vec{t}_2 \end{bmatrix} \begin{bmatrix} A \\ B \end{bmatrix} = \begin{bmatrix} ((\partial \vec{x}/\partial P)_2 - (\partial \vec{x}/\partial P)_1) \bullet \vec{t}_1 \\ ((\partial \vec{x}/\partial P)_1 - (\partial \vec{x}/\partial P)_2) \bullet \vec{t}_2 \end{bmatrix}$$

$$\left(\frac{\partial \vec{x}}{\partial P}\right)_{\text{node}} = \left(\frac{\partial \vec{x}}{\partial P}\right)_{\text{edge1}} + A\left(\frac{\partial \vec{x}}{\partial t}\right)_{\text{edge1}}$$

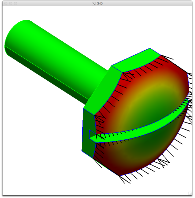
# Analytical Sensitivities Example



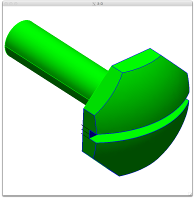
$\partial\vec{x}/\partial(\mathbf{Thead})$



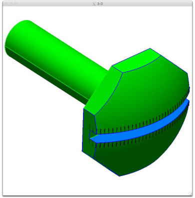
$\partial\vec{x}/\partial(\mathbf{Whead})$



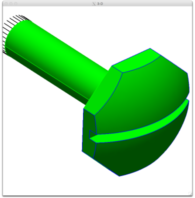
$\partial\vec{x}/\partial(\mathbf{Fhead})$



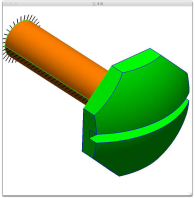
$\partial\vec{x}/\partial(\mathbf{Dslot})$



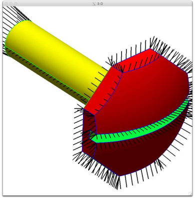
$\partial\vec{x}/\partial(\mathbf{Wslot})$



$\partial\vec{x}/\partial(\mathbf{Lshaft})$



$\partial\vec{x}/\partial(\mathbf{Dshaft})$



$\partial\vec{x}/\partial(\mathbf{sfact})$

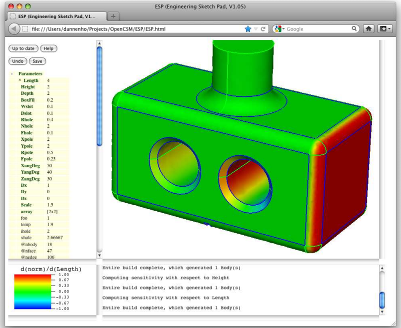
# Finite-difference Sensitivities (1)

- Basic strategy:
  - re-create configuration after perturbing a design parameter
    - requires regeneration
    - step-size must be chosen carefully
  - take finite difference of associated points in the configurations
- Assumptions made in previous approaches:
  - dilitation or contraction is related to Face's bounding parametric coordinates
    - local changes have large effect on whole Face
  - geometry's parametrization can be used to map point movement
    - for NURBs, geometry is based upon knot spacings

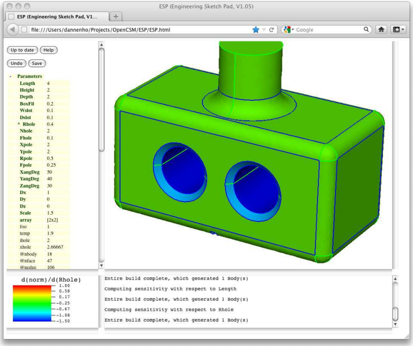
# Finite-difference Sensitivities (2)

- New approach:
  - compute a tessellation in the base configuration
    - discretize the Edges first
    - fill region with triangles only using the Edge points
  - discretize the perturbed Edges
    - use relative arc-lengths
    - find parametric coordinates $\vec{u}$ for adjacent Edges using "Pcurve" evaluations ($\vec{u}(t)$)
    - compute perturbation of space coordinates $\vec{x}$ on the Edges
  - for interior points
    - find barycentric coordinates in base coarse tessellation
    - propagate Edge parametric coordinate perturbations from the Edges to the interior
    - compute perturbation of space coordinates

# Finite-difference Sensitivity Example (1)

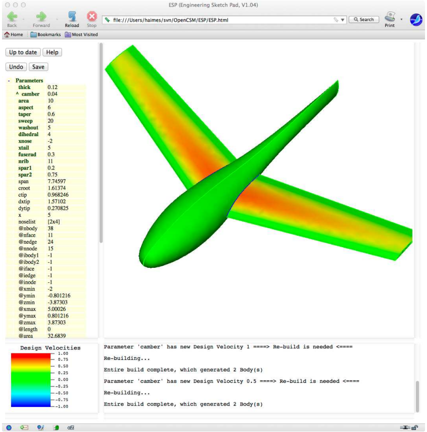



Change in box length

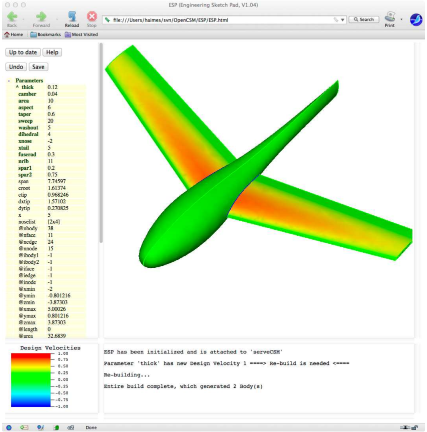



Change in the holes' radii

# Finite-difference Sensitivity Example (2)




Change in camber




Change in thickness

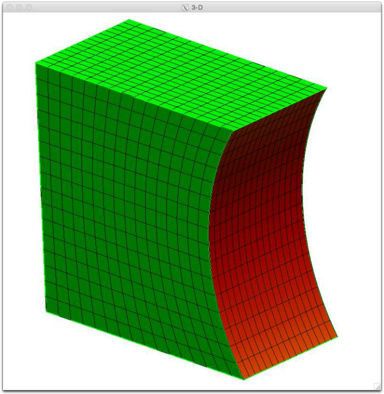
# Application to Grid Sensitivities

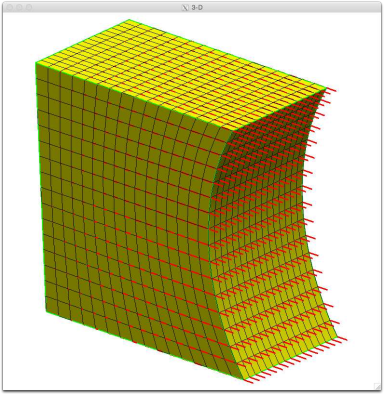
- Use configuration sensitivities to find (normal) change to surface location
- Use derivative of grid generator to find tangential change along surface

$$\left(\frac{d\vec{x}}{dP}\right)_{i,j} = \left(\frac{\partial w}{\partial P}\right)_{i,j} \vec{n}_{i,j} + \left(\frac{\partial \vec{x}}{\partial \vec{u}}\right)_{i,j} \left(\frac{d\vec{u}}{dP}\right)_{i,j}$$

- $d\vec{u}/dP$ in the interior comes from $d\vec{u}/dP$ on the Edges, which come from $d\vec{u}/dP$ at the Nodes
- Process is easily executed by doing Nodes first, then Edges, then Faces

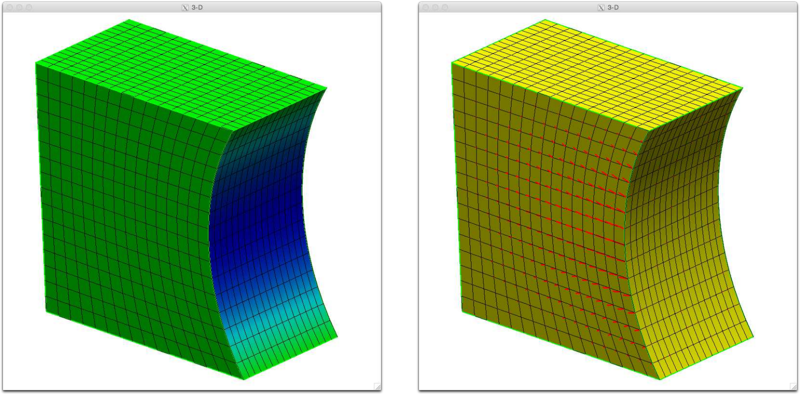
# Grid Sensitivities Example (1)




Sensitivity with respect to the length of the box

# Grid Sensitivities Example (2)




Sensitivity with respect to the depression distance

# Conclusions (1)

- Sensitivities of a parametric, CAD-generated configuration w.r.t. design parameters can be robustly and efficiently found using a combination of techniques
  - Analytic derivatives are used whenever possible
    - efficient — do not require regeneration of configuration
    - accurate — not susceptible to truncation error
    - automatic code differentiation can be used when source code is available and derivatives are too hard to compute by hand
  - Finite differences are used when necessary
    - require regeneration of perturbed configuration
    - the original tessellation is reused to ensure proper point matching between base and perturbed geometries

# Conclusions (2)

- Configuration sensitivity is computed locally based upon a point's initial location in space
  - returns motion normal to Faces and Edges
  - is insensitive to surface parametrization
- Grid sensitivities can be found using just the configuration sensitivities and a knowledge of the grid generation scheme
- Tools are now available to produce the sensitivity of MDAO objective function(s) w.r.t. the engineer's design parameters

# Computing Configuration Sensitivities in ESP (1)

- Build a model with Design Parameters
- For simple sensitivities (that is, with respect to one Design Parameter at a time)
  - select (edit) the Design Parameter
  - press **Compute Sensitivity**
  - configuration will automatically be rebuilt and display will change
    - minimum and maximum sensitivities will be reported in Messages window
    - configuration will be colored in Graphics window
    - Key window will contain the color key, whose limits can be changed by clicking in the Key window

# Computing Configuration Sensitivities in ESP (2)

- The meaning of the various colors is:
  - red (positive sensitivity) are regions where a positive change in the Design Parameter would move the surface in the direction of the local outward-facing surface normal
  - blue (negative sensitivity) are regions where a negative change in the Design Parameter would move the surface in a direction opposite the local outward-facing surface normal
- Example for a cylindrical feature:
  - for a post-like feature, the sensitivity with respect to the diameter would be positive
  - for a hole-like feature, the sensitivity with respect to the diameter would be negative

# Computing Configuration Sensitivities in ESP (3)

- To find the sensitivity with respect to a multi-valued Design Parameter
  - select (edit) the multi-valued Design Parameter
  - press **Clear Design Velocities**
  - press **Set Design Velocity**
    - answer **1** for the entity for which you want the sensitivity
    - answer **0** (the default) for all other entities

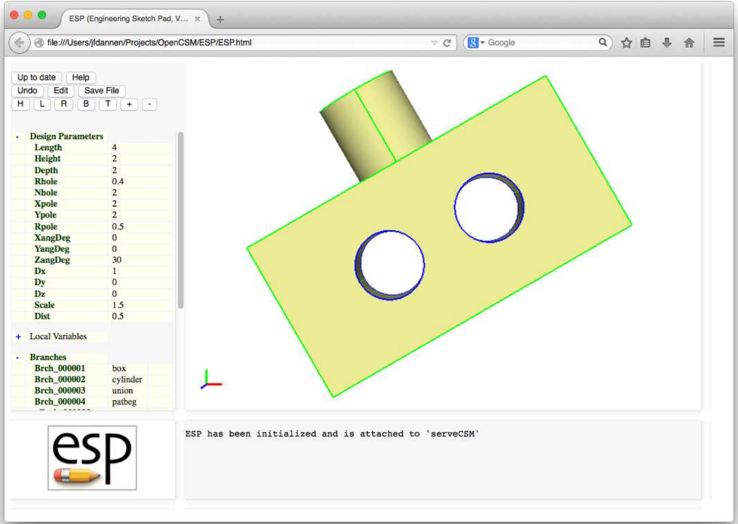
# Computing Configuration Sensitivities in ESP (4)

- To find the sensitivity with respect to a several Design Parameters at the same time (for example, in the direction of the gradient proposed by an optimizer)
  - select any Design Parameter
  - press **Clear Design Velocities**
  - for each Design Parameter whose component to the gradient direction is non-zero
    - press **Set Design Velocity**
    - enter the associated component of the gradient vector
  - press **Press to Re-build**
  - Note: the key window will say `d(norm)/d(***)` to indicate that the sensitivity is with respect to some combination of Design Parameters

# Computing Tessellation Sensitivities in ESP

- Process is same as for Configuration sensitivities, except:
  - `serveCSM` must be started with the `-sensTess` command line option
  - sensitivities are shown both with the color map and with superimposed tufts
  - the lengths of the tufts can be changes by changing the magnitude of the Design Parameter velocities

# Simple Block (1)



# Simple Block (2)

| | | |
|---|---|---|
| Box | | |
| Length | length of box | 4.0 |
| Height | height of box | 2.0 |
| Depth | depth of box | 2.0 |
| | anchored at $X = Z = 0$ | |
| | centered at $Y = 0$ | |
| Holes | | |
| Rhole | radii of the holes | 0.4 |
| Nhole | number of holes | 2 |
| | holes are equally spaced | |
| Pole | | |
| Xpole | $X$-location of top of pole | 2.0 |
| Ypole | $Y$-location of top of pole | 2.0 |
| Rpole | radius of pole | 0.5 |

# Simple Block (3)

| Rotation about origin | | |
|---|---|---|
| XangDeg | $X$ rotation (deg) | 0. |
| YangDeg | $Y$ rotation (deg) | 0. |
| ZangDeg | $Z$ rotation (deg) | 30. |
| Translation | | |
| Dx | | 1.0 |
| Dy | | 0.0 |
| Dz | | 0.0 |
| Scaling | | |
| Scale | overall scaling factor | 1.5 |

# Simple Block (4)

- Starting file is at `$ESP_ROOT/training/ESP/data/session09/simpleBlock.csm`
- What is the configuration sensitivity to each Design Parameter?
- What is the configuration sensitivity if you change two Design Parameters at the same time?
- What is the tessellation sensitivity to each Design Parameter?