What is ACIS ?

- A solid modeler
- Front end in scheme
- Back end in C++

How to use ACIS with scheme?

- Copy these lines to ~/.bashrc on nsl-machines
  - export LD_LIBRARY_PATH=$LD_LIBRARY_PATH :/users/courses/cs336/acis/acis/bin/linux_so:/users/courses/cs336/acis/acis/lib/linux_so
  - export PATH=$PATH:/users/courses/cs336/acis/acis/bin/linux_so
  - alias acis='acis3dt -p /users/courses/cs336/acis/acis/scm/examples:<path_to_your_scm_files_dir>'

Basic Commands

- (view:gl) : opens the graphical window
- (part:clear) : Deletes all entities from the active part
- (part:entities): List all entities in the active part
- (entity:check): Checks the data structure, topology, and geometry on a single or list of entities.
- (entity:set-color): Sets the color of an entity

Topology and Geometry

- Face and surface
  - (surface:from-face) : Returns the underlying surface
  - (pick-face): click and select face
- Edge and curve
  - (curve:from-edge) : Returns the underlying curve
  - (pick-edge): click and select edge

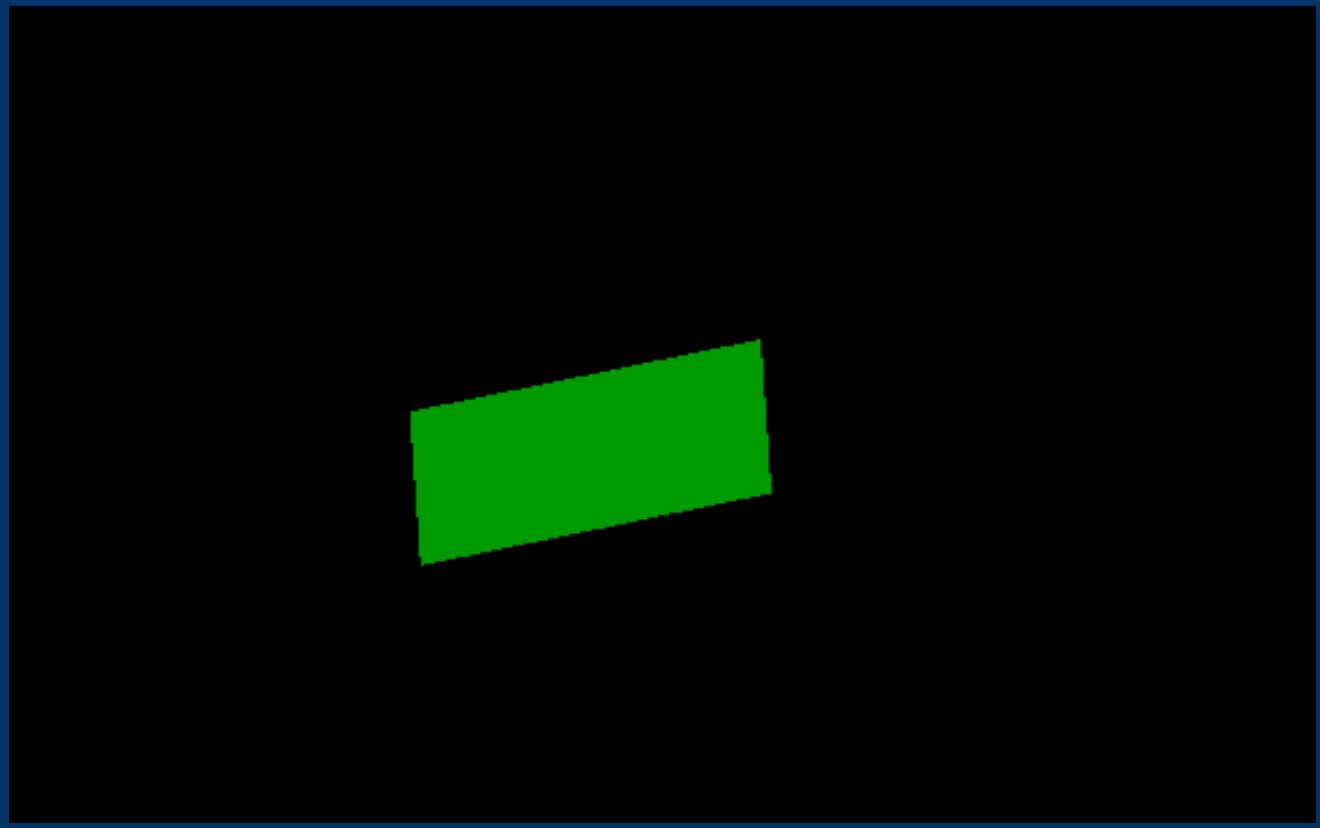
Face

- (face:bs) : Returns a B-spline approximation
- (face:closest-point) : Returns the closest point on the face from a given point.
- (face:intersect) : Computes the curve of intersection of two faces
- (face:spline) : makes spline faces
- (face:plane), (face:sphere), (face:cylinder), (face:cone), (face:torus) etc.

Edge

- (edge:bs) : Returns a B-splines approximation
- (edge:intersect): Intersection between 2 edges
- (edge:end), (edge:start): Start and end positions of edges
- (edge:end-dir), (edge:start-dir): Start and end directions
- (view:edges ON) : display edges in the graphical window
- (edge:helix), (edge:spiral), (edge:linear)

Sweeps

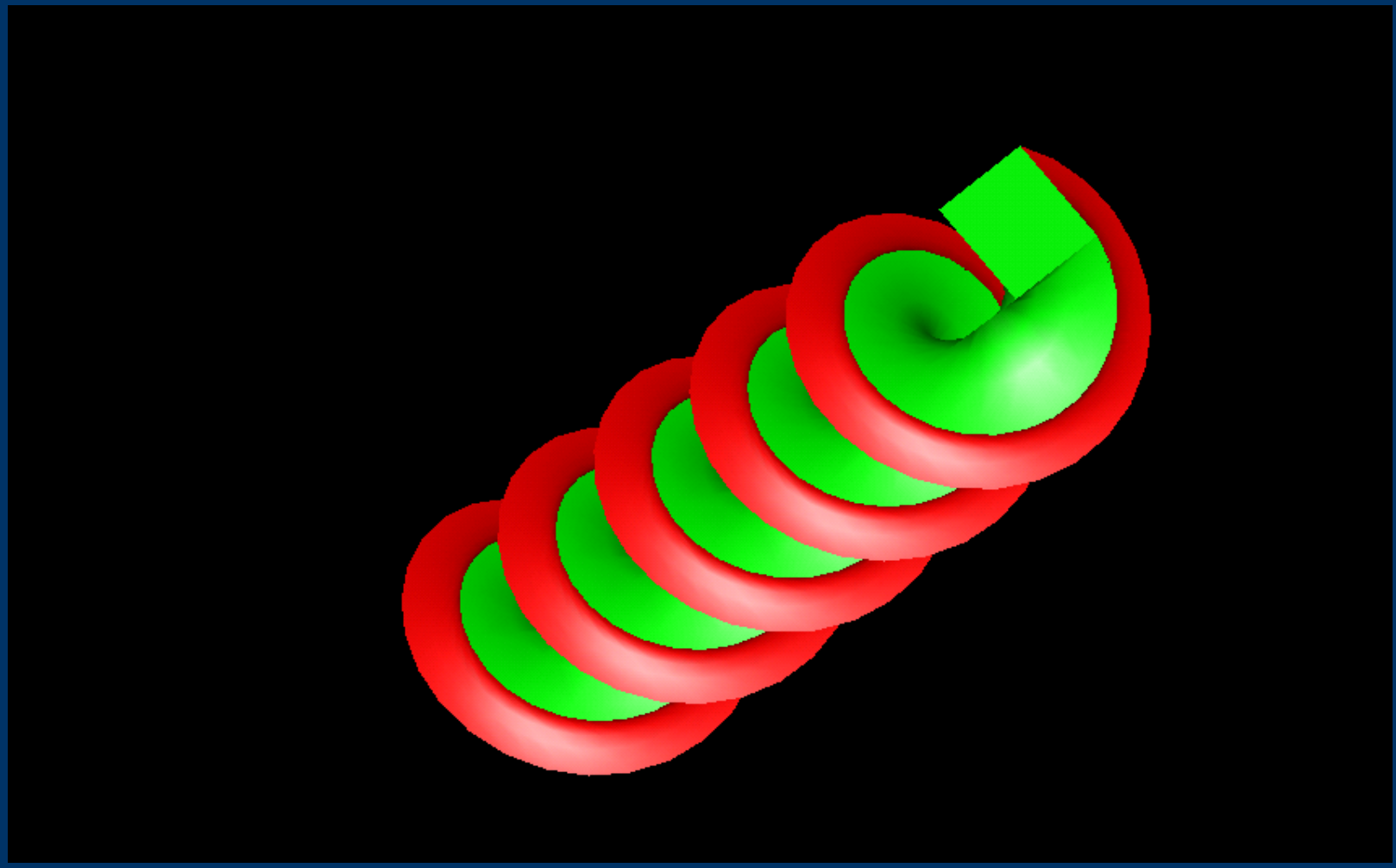


Sweep code

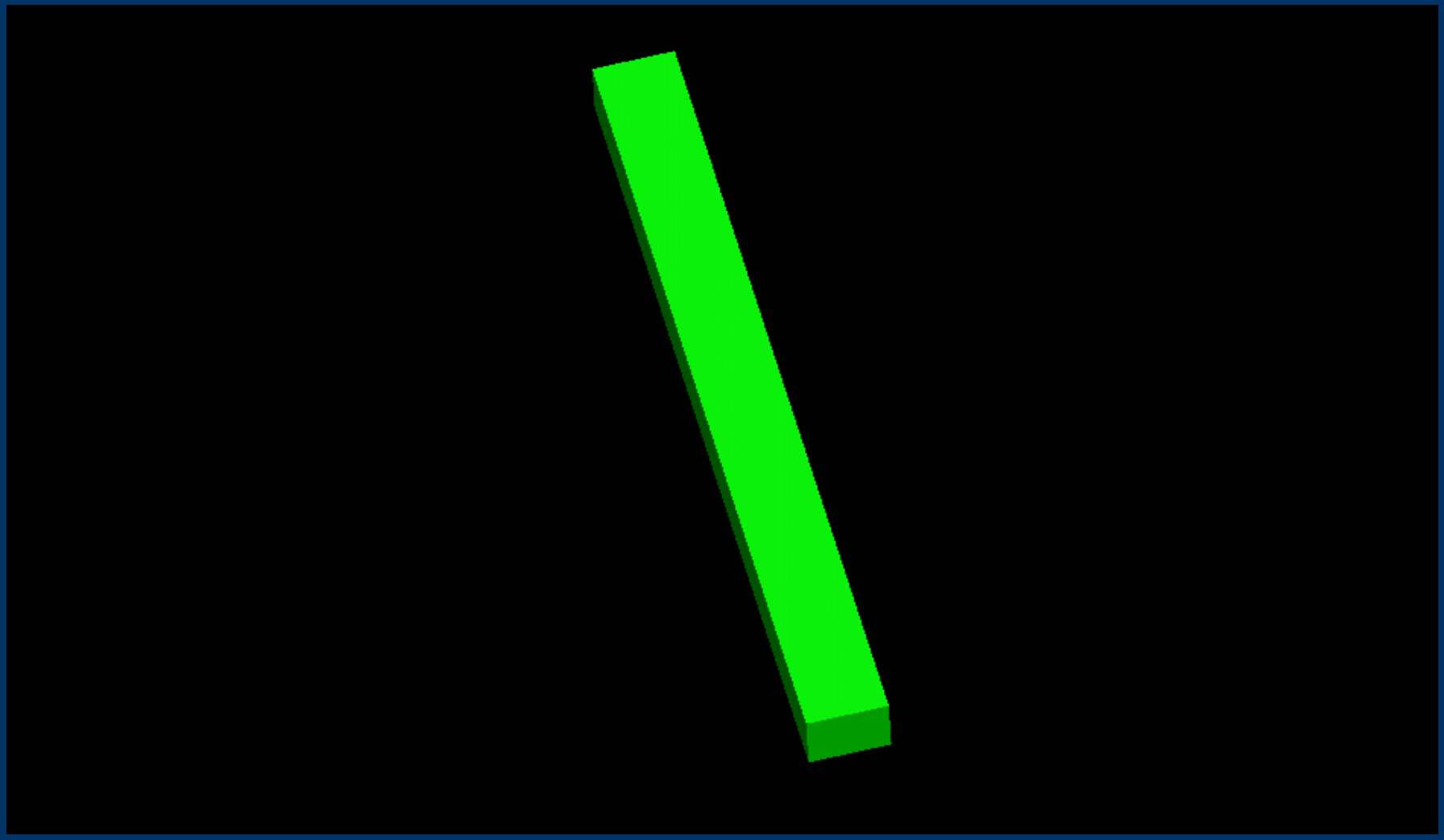
```
(define axis-start1 (position 0 0 0))
(define axis-end1 (position 0 0 10))
(define start-dir1 (gvector 1 0 0))
(define radius1 0.5)
(define thread-distance1 2)
(define helix1
(edge:helix axis-start1 axis-end1 start-dir1
 radius1 thread-distance1))


(define sweep1 (sweep:law spline helix1))
```

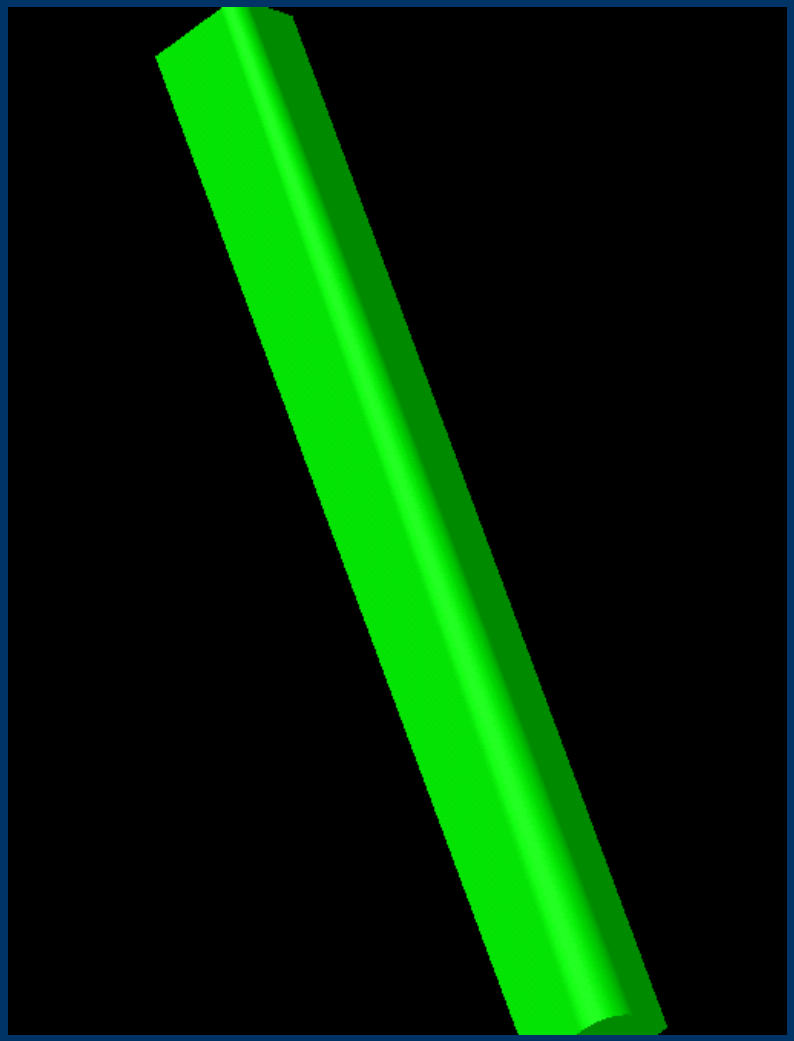
Sweeps - II



Sweeps III



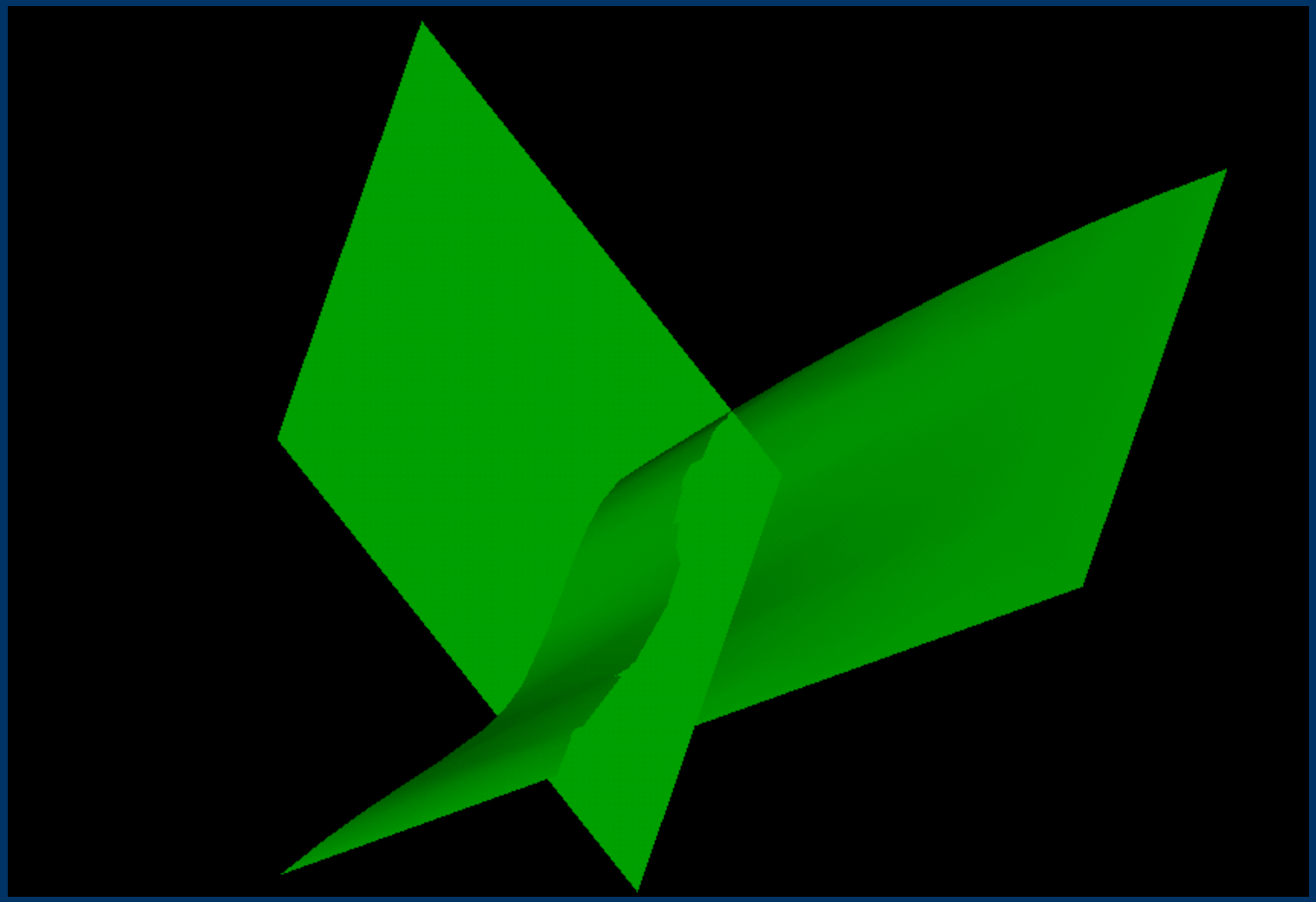
Blends



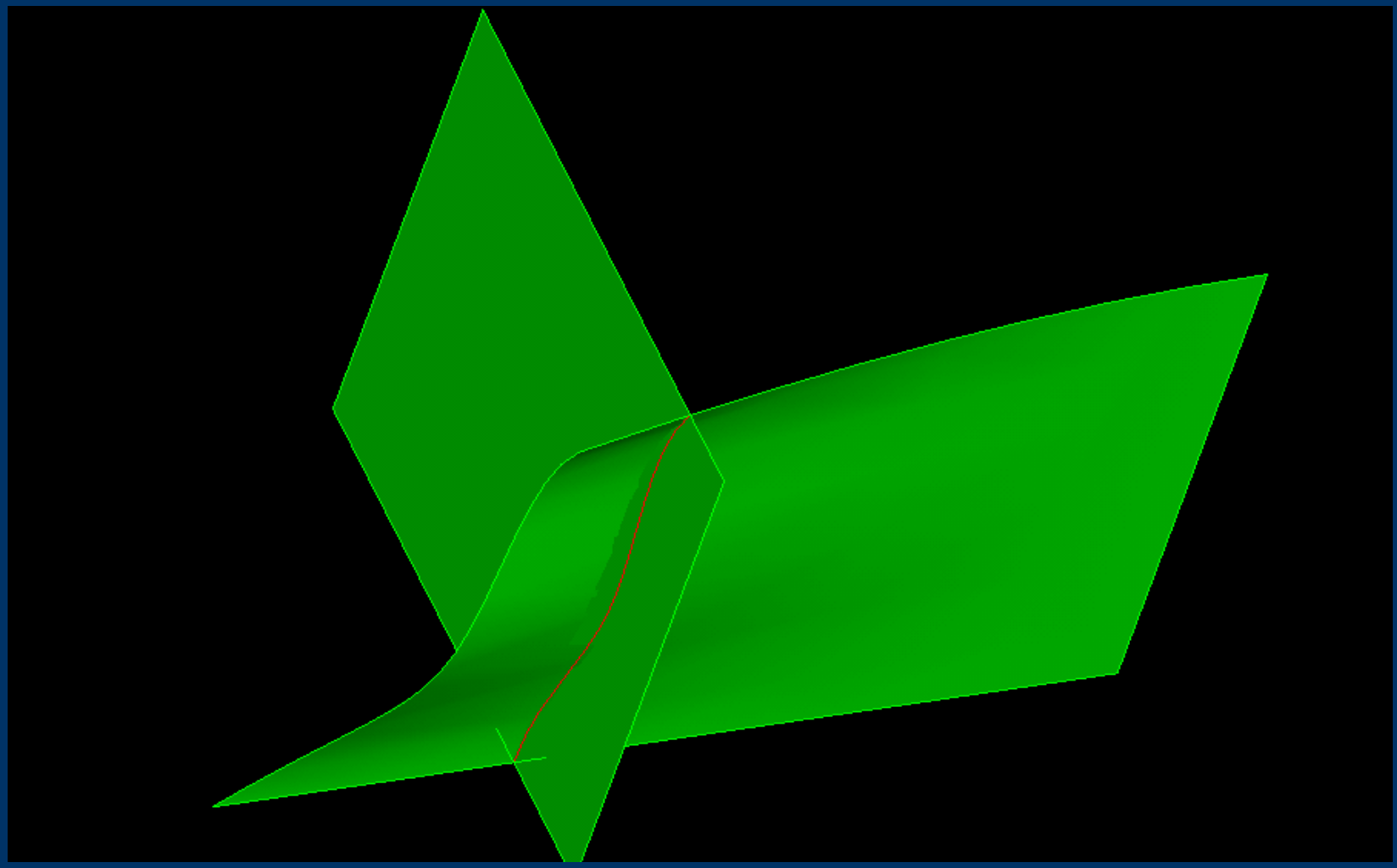
- (define pos-list
  (list (edge:end e1)
  (edge:start e1))

  (solid:blend-edges-pos-rads e1 pos-list (list 0.2 0.4))

Intersection



Intersection II



Other information

Acis help on NSL machines :
file:///users/courses/cs336/docs/ACIS.htm