Surface to Surface Intersections

N. M. Patrikalakis, T. Maekawa, K. H. Ko, H. Mukundan

May 2004

Massachusetts Institute of Technology

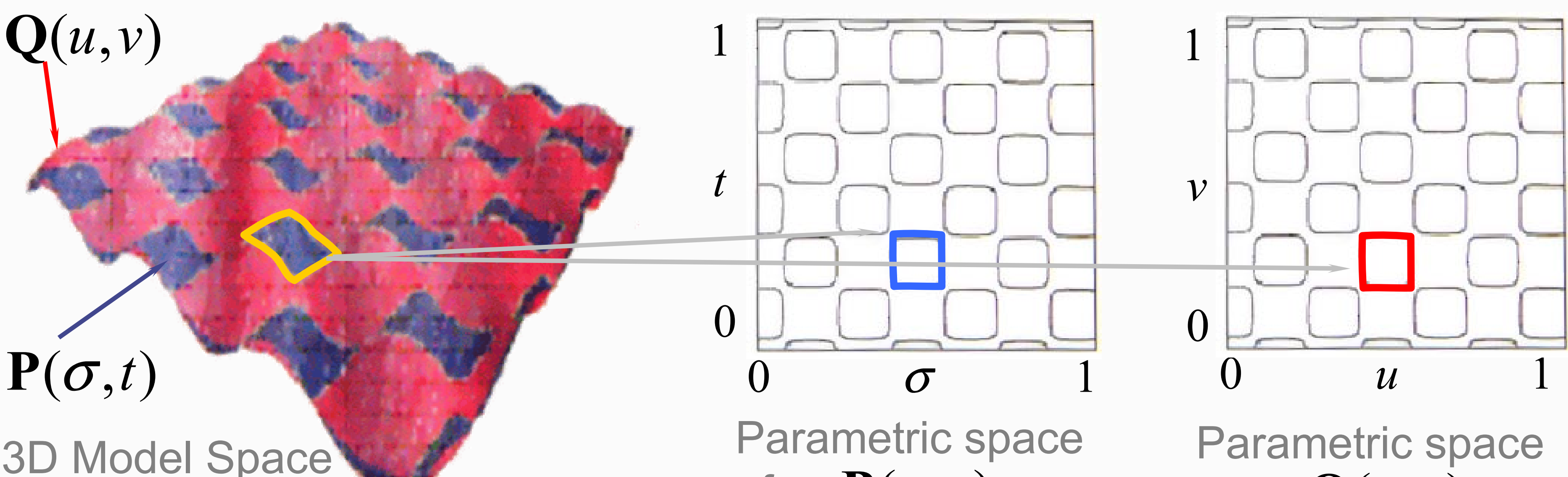
Introduction
Motivation

- Surface to surface intersection (SSI) is needed in:
  - Solid modeling (B-rep)
  - Contouring
  - Numerically controlled machining (Milling)
  - Collision avoidance
  - Feature recognition
  - Manufacturing simulation
  - Computer animation

Introduction

Background

- Intersection of two *parametric surfaces*, $\mathbf{P}(\sigma,t)=\mathbf{Q}(u,v)$ defined in *parametric spaces* $0 \le \sigma,t \le 1$ and $0 \le u,v \le 1$ can have *multiple components* [4].



Parametric space of $\mathbf{P}(\sigma,t)$

Parametric space of $\mathbf{Q}(u,v)$

- An *intersection curve segment* is represented by a continuous trajectory in *parametric space.*

Introduction
Possible Approaches

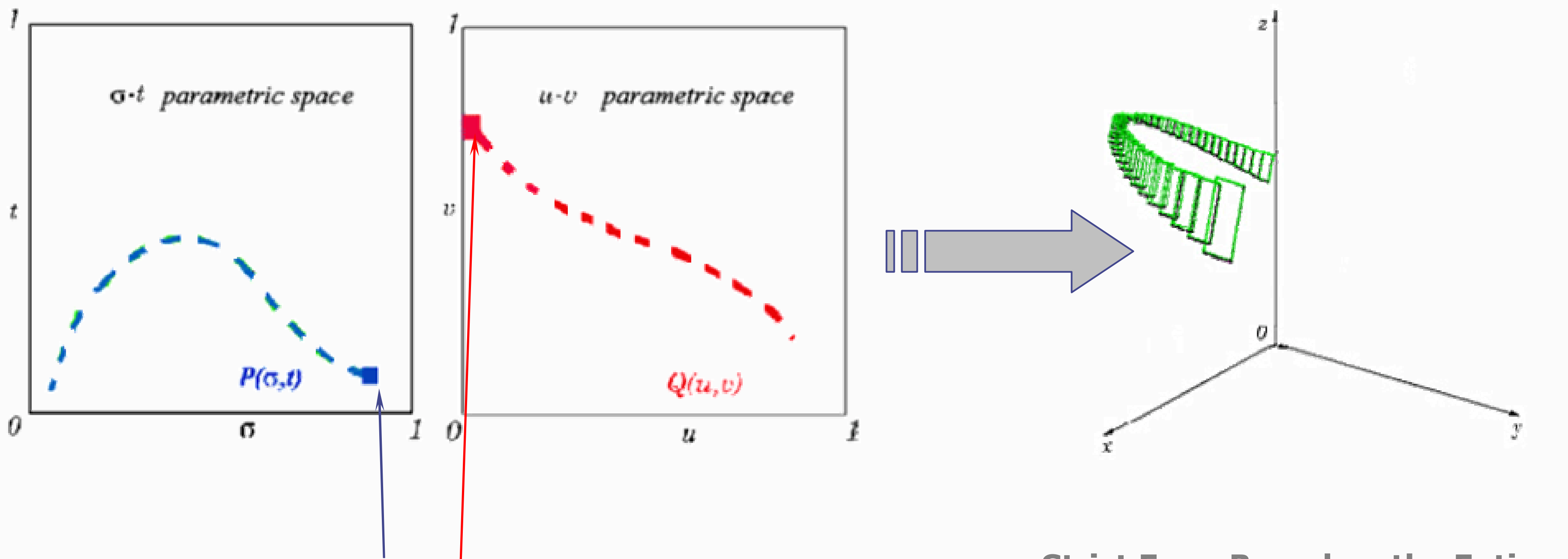
- Three popular methods
  - Lattice methods
    - Issues related to topology, missing roots.
  - Subdivision based methods
    - Issues related to topology, extraneous roots.
  - **Marching scheme (Our Choice)**
    - Intersection curve segment is computed through an IVP.

Introduction
Marching Scheme

- A *marching scheme* involves:
  - Identifying all components
  - Obtaining an accurate starting point in each component
  - **Tracing the given intersection correctly**
- Assumption:
  - The given surfaces are *Rational Polynomial Parametric (RPP).*
  - We are given an intersection curve segment.
  - No singularities exist in the intersection curve segment.

Introduction
Objective

- Given an error bound on the *starting point* in both *parametric spaces,* obtain a bound for the entire *intersection curve segment* in *3D model space.*



Strict Error Bound on Starting Point (Given)

Strict Error Bound on the Entire Intersection Curve Segment (Goal)

Outline

- **Problem Formulation**
- Error Bounds in Parametric Space
- Error Bounds in 3D Model Space
- Results and Examples
- Conclusions

Problem Formulation
Transversal Intersection

- Transversal intersection formulated as a system of *ordinary differential equations* (ODEs) in *parametric space* [4].

From $\mathbf{P}(\sigma,t)=\mathbf{Q}(u,v)$, we obtain :

$$\frac{d\sigma}{ds}=\frac{Det[\mathbf{c},\mathbf{P}_t,\mathbf{N}^P(\sigma,t)]}{\mathbf{N}^P(\sigma,t)\bullet\mathbf{N}^P(\sigma,t)},$$

$$\frac{dt}{ds}=\frac{Det[\mathbf{P}_\sigma,\mathbf{c},\mathbf{N}^P(\sigma,t)]}{\mathbf{N}^P(\sigma,t)\bullet\mathbf{N}^P(\sigma,t)},$$

$$\frac{du}{ds}=\frac{Det[\mathbf{c},\mathbf{Q}_v,\mathbf{N}^Q(u,v)]}{\mathbf{N}^Q(u,v)\bullet\mathbf{N}^Q(u,v)},$$

$$\frac{dv}{ds}=\frac{Det[\mathbf{Q}_u,\mathbf{c},\mathbf{N}^Q(u,v)]}{\mathbf{N}^Q(u,v)\bullet\mathbf{N}^Q(u,v)},$$

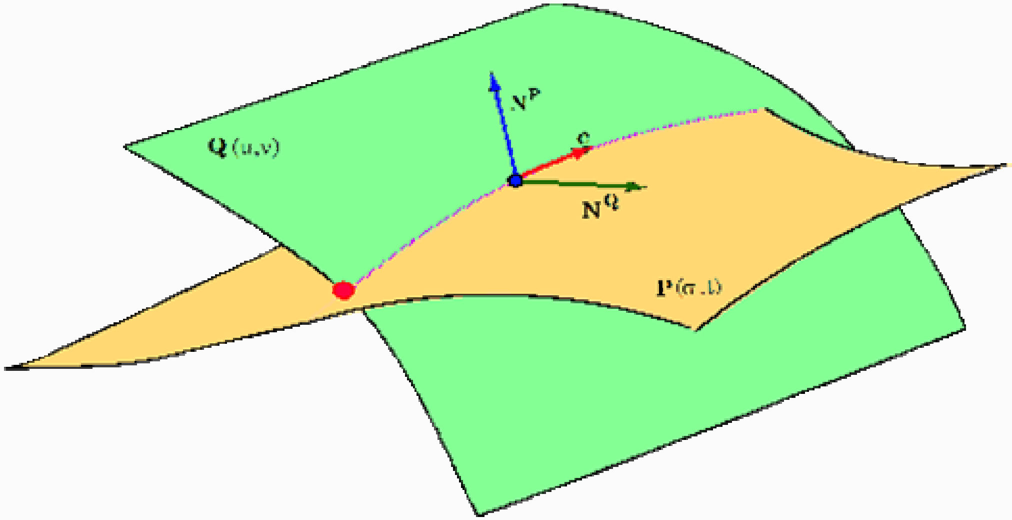
where s is the arc length parameter and

$\mathbf{P}_\sigma,\mathbf{P}_t,\mathbf{Q}_u,\mathbf{Q}_v$ *are partial derivatives*

$\mathbf{N}^P=\mathbf{P}_\sigma\times\mathbf{P}_t$ *Normal to* $\mathbf{P}(\sigma,t)$

$\mathbf{N}^Q=\mathbf{Q}_u\times\mathbf{Q}_v$ *Normal to* $\mathbf{Q}(u,v)$

$$\mathbf{c}=\frac{\mathbf{N}^P\times\mathbf{N}^Q}{\left|\mathbf{N}^P\times\mathbf{N}^Q\right|}$$



Problem Formulation
Tangential Intersection

- ODEs have the same form as in transversal intersection case

$$\sigma' = \frac{Det[\mathbf{c}, \mathbf{P}_t, \mathbf{N}^P(\sigma,t)]}{\mathbf{N}^P(\sigma,t) \bullet \mathbf{N}^P(\sigma,t)}, \quad t' = \frac{Det[\mathbf{P}_\sigma, \mathbf{c}, \mathbf{N}^P(\sigma,t)]}{\mathbf{N}^P(\sigma,t) \bullet \mathbf{N}^P(\sigma,t)}, \quad u' = \frac{Det[\mathbf{c}, \mathbf{Q}_v, \mathbf{N}^Q(u,v)]}{\mathbf{N}^Q(u,v) \bullet \mathbf{N}^Q(u,v)}, \quad v' = \frac{Det[\mathbf{Q}_u, \mathbf{c}, \mathbf{N}^Q(u,v)]}{\mathbf{N}^Q(u,v) \bullet \mathbf{N}^Q(u,v)},$$

$$\mathbf{N}^P = \mathbf{P}_\sigma \times \mathbf{P}_t \quad \textit{Normal to } \mathbf{P}(\sigma,t) \textit{ and } \mathbf{N}^Q = \mathbf{Q}_u \times \mathbf{Q}_v \quad \textit{Normal to } \mathbf{Q}(u,v)$$

$\mathbf{c}$ is determined using the second derivatives of the surfaces.

- From the condition of *equal normal curvatures* we obtain the equation

$$b_{11}(\sigma')^2 + 2b_{12}(\sigma')(t') + b_{22}(t')^2 = 0$$

where b_{11}, b_{12}, b_{22} are functions of the *first* and *second fundamental form coefficients* of the surfaces.

- For a unique marching direction, $(b_{12}^2 - b_{11}b_{22}) = 0$ and $(b_{12}^2 + b_{11}^2 + b_{22}^2) \neq 0$
  - **Thus if:** $b_{11} \neq 0$, **or if:** $b_{11} = 0,\ b_{22} \neq 0$

$$\mathbf{c} = \frac{\nu \mathbf{P}_\sigma + \mathbf{P}_t}{|\nu \mathbf{P}_\sigma + \mathbf{P}_t|}, \text{ where } \nu = -\frac{b_{12}}{b_{11}}$$

$$\mathbf{c} = \frac{\mathbf{P}_\sigma + \mu \mathbf{P}_t}{|\mathbf{P}_\sigma + \mu \mathbf{P}_t|}, \textit{ where } \mu = -\frac{b_{12}}{b_{22}}$$

Problem Formulation
Vector IVP for ODE

- Given a *starting point (initial condition)* belonging to an *intersection curve segment,* we can integrate the system of ODEs.

- The system of ODEs with the *starting point* represents an initial value problem (IVP).
  - Written in vector notation as:

$$\begin{bmatrix} \frac{d\sigma}{ds} \\ \frac{dt}{ds} \\ \frac{du}{ds} \\ \frac{dv}{ds} \end{bmatrix} = \begin{bmatrix} f_1(\sigma,t,u,v) \\ f_2(\sigma,t,u,v) \\ f_3(\sigma,t,u,v) \\ f_4(\sigma,t,u,v) \end{bmatrix}$$

$$\frac{d\mathbf{y}}{ds} = \mathbf{f}(\mathbf{y}), \quad \mathbf{y}(\mathbf{0}) = \mathbf{y}_0$$

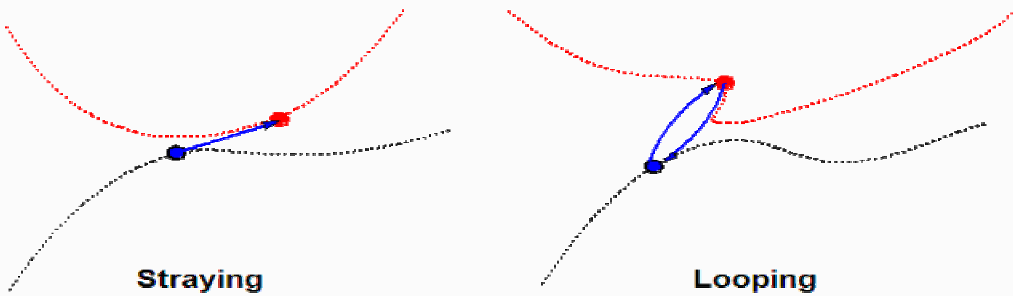
Outline

- Problem Formulation
- **Error Bounds in Parametric Space**
  - **Review of Standard Schemes**
  - **Interval Arithmetic**
  - **Validated Interval Scheme**
- Error Bounds in 3D Model Space
- Results and Examples
- Conclusions

Error Bounds in Parametric Space
Review of Standard Schemes

- Well-known Standard Schemes:
  - Runge-Kutta Method
  - Adams-Bashforth Method
  - Taylor Series Method

- Properties of Standard Schemes:
  - They are approximation schemes and introduce a *truncation error*
  - They do not consider *uncertainty in initial conditions*
  - They are prone to *rounding errors*
  - They suffer from *straying* or *looping* near closely spaced features



Error Bounds in Parametric Space
Interval Arithmetic (Introduction)

- Intervals are defined by [2]:

$$[a] \equiv [\underline{a}, \overline{a}] \equiv \{x | \underline{a} \leq x \leq \overline{a}\}, \ \underline{a}, \overline{a}, x \in \mathbf{R}, \ \underline{a} \leq \overline{a}$$

- Example:

$$\pi = 3.14159265\,3589793238\,46\ldots$$
$$\pi \in [3.141\,,\ 3.142] = [\pi]$$

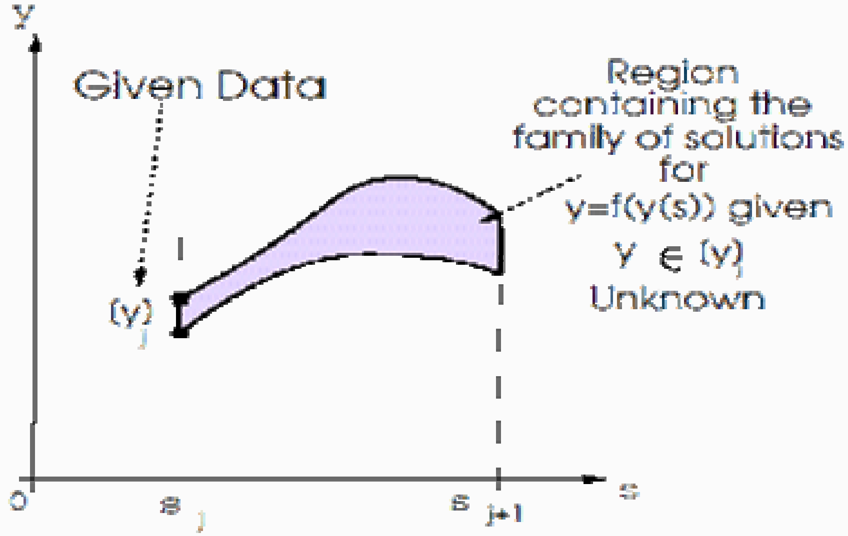
- Basic *interval arithmetic* operations are defined by:

$$\begin{aligned}
[a,b] + [c,d] &= [a+c, b+d] \\
[a,b] - [c,d] &= [a-d, b-c] \\
[a,b] \cdot [c,d] &= [\min(a\cdot c, a\cdot d, b\cdot c, b\cdot d), \max(a\cdot c, a\cdot d, b\cdot c, b\cdot d)] \\
[a,b] \,/\, [c,d] &= [\min(a/c, a/d, b/c, b/d), \max(a/c, a/d, b/c, b/d)], \quad \text{for } 0 \notin [c,d]
\end{aligned}$$

Error Bounds in Parametric Space

Interval Arithmetic (Solution of IVPs)

- For strict bounds for IVPs in *parametric space,* we employ a *validated interval scheme* for ODEs [3].
- The error in *starting point* is bounded by an *initial interval*.
- *Interval solution* represents a family of solutions passing through the *initial interval* satisfying the governing ODEs.



Error Bounds in Parametric Space
Validated Interval Scheme (Introduction)

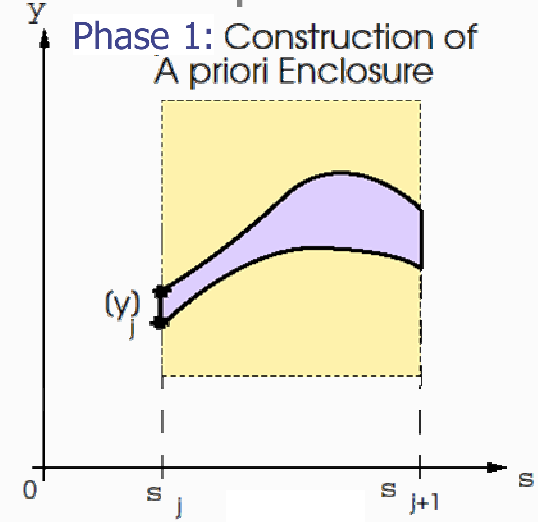
- Every *step* of a *validated interval scheme* involves [3]:
  - Computing an interval valued function $[\mathbf{y}](s)$ such that:
    - $\mathbf{y}(s) \in [\mathbf{y}](s)$, and
    - The width of the $[\mathbf{y}](s)$ is below a given tolerance

    $\forall\ s \in [s_j, s_{j+1}]$
  - Verifying the *existence* and *uniqueness* of the solution in $[s_j, s_{j+1}]$.

Error Bounds in Parametric Space

Validated Interval Scheme (Overview)

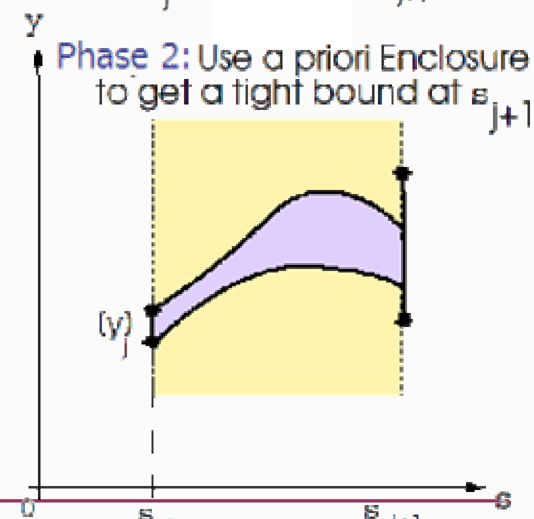
- One *step* of a *validated interval scheme* is done in *two* phases:



- **Phase I Algorithm**
  - A *step size* $h_j = s_{j+1} - s_j$
  - An *a priori enclosure* $[\tilde{y}_j]$ such that:

$$y(s) \in [\tilde{y}_j], \quad \forall \ s \in [s_j, s_{j+1}]$$

- **Phase II Algorithm**
  - Using $[\tilde{y}_j]$ compute a *tighter bound* $[y_{j+1}]$ at s_{j+1} .



Error Bounds in Parametric Space

Validated Interval Scheme (Phase I : Validation)

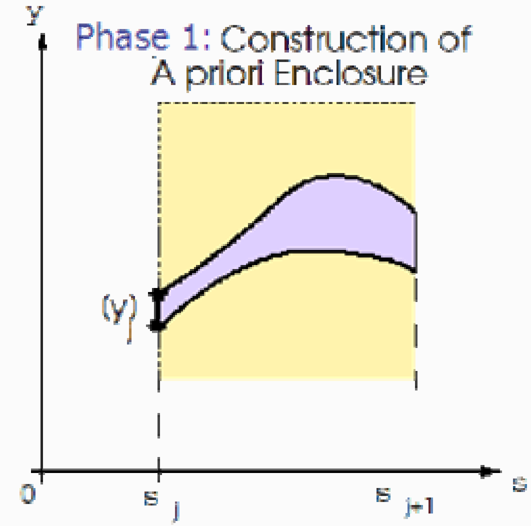
- A pair of $[\tilde{\mathbf{y}}_j]$ and h_j satisfying the relation:

$$[\tilde{\mathbf{y}}_j] \supseteq [\mathbf{y}_j] + \mathbf{f}([\tilde{\mathbf{y}}_j])h_j$$

  - This assures *existence* and *uniqueness* of the solution.
  - This method is called a *constant enclosure method* [3].

- The *a priori enclosure* $[\tilde{\mathbf{y}}_j]$ bounds the true solution in the *parametric space* $\forall\ s \in [s_j, s_{j+1}]$.

- Numerical implementation
  - Choosing a $[\tilde{\mathbf{y}}_j]$ and,
  - Iterating to find a corresponding h_j .



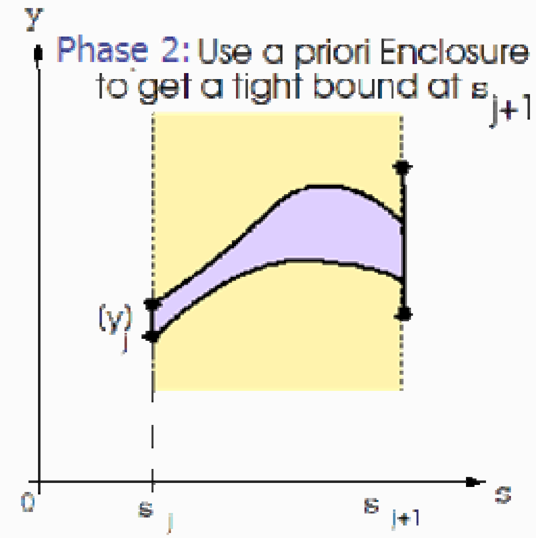
Error Bounds in Parametric Space

Validated Interval Scheme (Phase II : Tighter Bound)

- Using the *a priori enclosure* we
  - find a tighter bound $[\mathbf{y}_{j+1}]$ at s_{j+1} [3].
- This phase helps in the propagation of the solution by providing an *initial interval* for the successive step.
- The key idea is to use:
  - Interval version of *Taylor's formula* [3].

$$[\mathrm{y}_{j+1}] = [\mathrm{y}_j] + \sum_{i=1}^{k-1} h_j^{\,i} \mathbf{f}^{[i]}([\mathrm{y}_j]) + h_j^{\,k} \mathbf{f}^{[k]}([\tilde{\mathrm{y}}_j])$$

where $\mathbf{f}^{[i]}([\mathrm{y}_j])$ represents the i^{th} *Taylor coefficient* obtained using a technique called Automatic Differentiation [3].



Error Bounds in Parametric Space

Validated Interval Scheme (Application to SSI)

- We represent the surfaces as *interval surfaces.*
  - *Interval surfaces* have *interval coefficients* and are written as:

$$[\mathbf{P}](\sigma, t) \text{ and } [\mathbf{Q}](u, v)$$

- We obtain a *vector interval ODE system* :

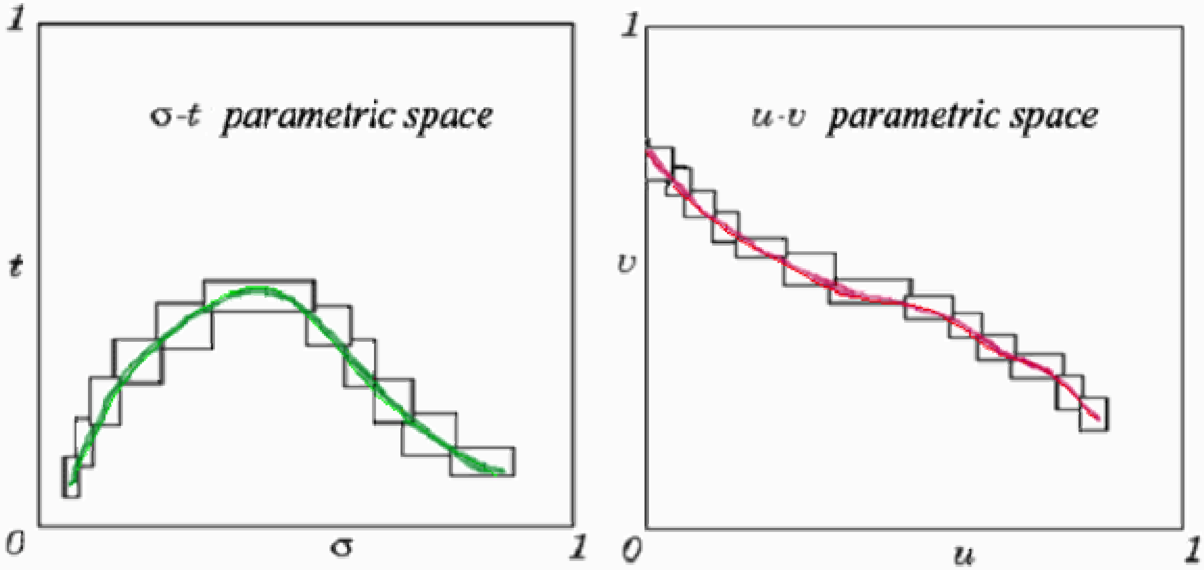
$$\left[\frac{d\sigma}{ds} \ \frac{dt}{ds} \ \frac{du}{ds} \ \frac{dv}{ds} \right]^T = \frac{d\mathbf{y}}{ds} = \mathbf{f}([\mathbf{y}(s)])$$

- With an *interval initial condition* :

$$[\mathbf{y}_0] = [[\sigma_0] \ \ [t_0] \ \ [u_0] \ \ [v_0]]^T$$

Error Bounds in Parametric Space

- Validated ODE solver produces *a priori enclosures* in *parametric space* of each surface, guaranteed to contain the *true intersection curve segment*.



- The union of *a priori enclosures* bounds the *true intersection curve segment* in *parametric space*.

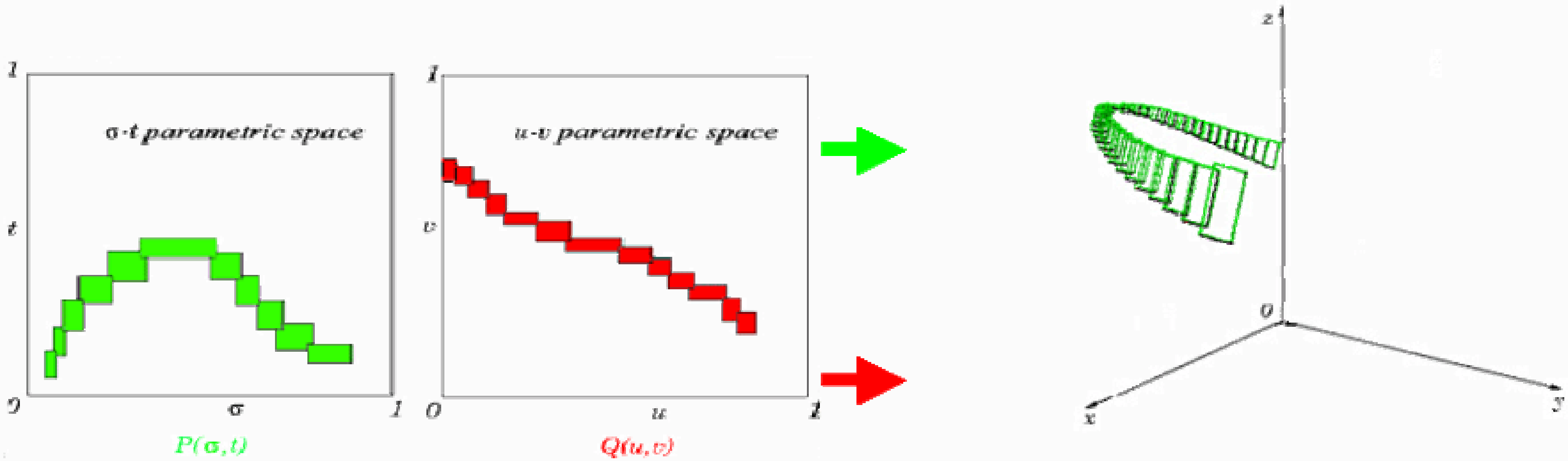
Outline

- Problem Formulation
- Error Bounds in Parametric Space
- **Error Bounds in 3D Model Space**
- Results and Examples
- Conclusions

Error Bounds in 3D Model Space

Mapping into 3D Model Space

- Mapping from *parametric space* to *3D model space*
  - using corresponding surfaces $[\mathbf{P}](\sigma,t)$ or $[\mathbf{Q}](u,v)$
  - coupled with *rounded interval arithmetic* evaluation



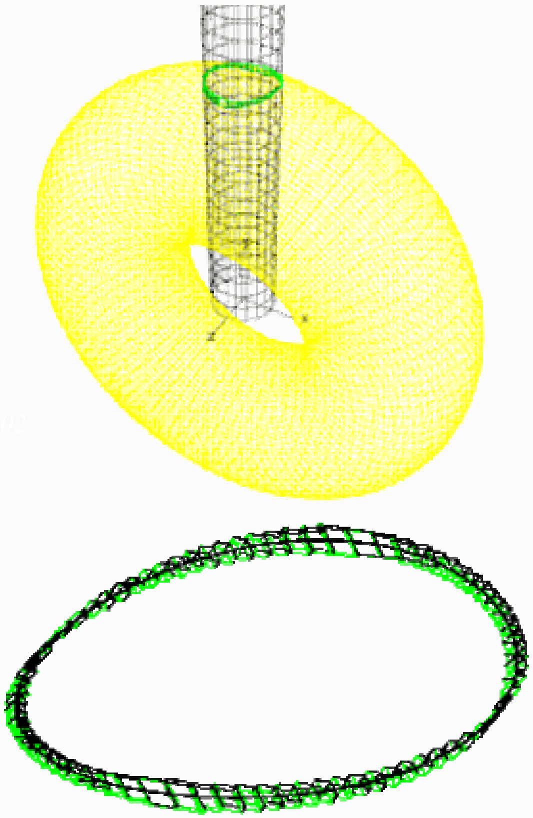
- Ensures continuous error bounds in *3D model space* [1] guaranteed to contain the *true curve of intersection*.

Outline

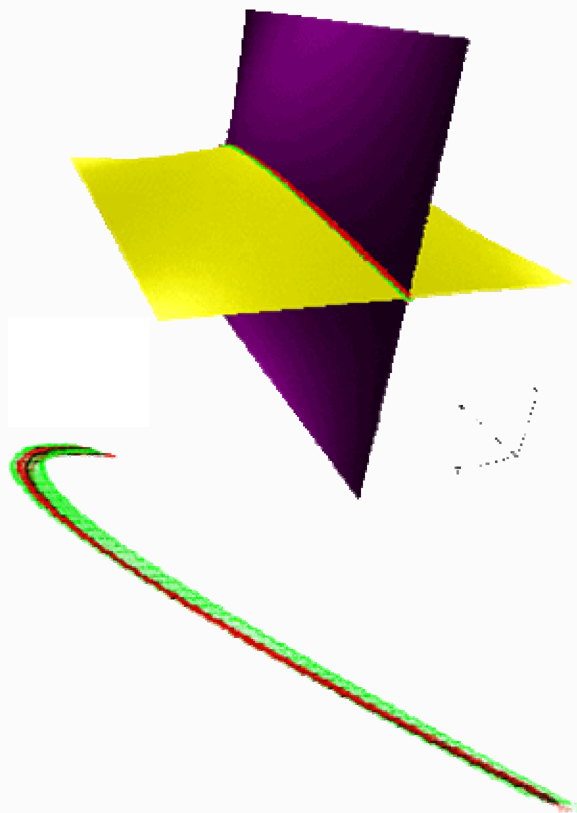
- Problem Formulation
- Error Bounds in Parametric Space
- Error Bounds in 3D Model Space
- **Results and Examples**
- Conclusions

Results & Examples

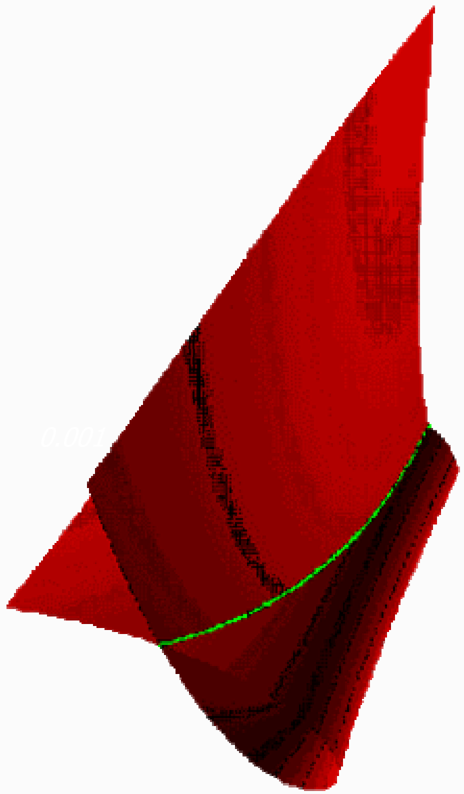
Error Bounds in 3D Model Space (Transversal)



Torus and *cylinder*



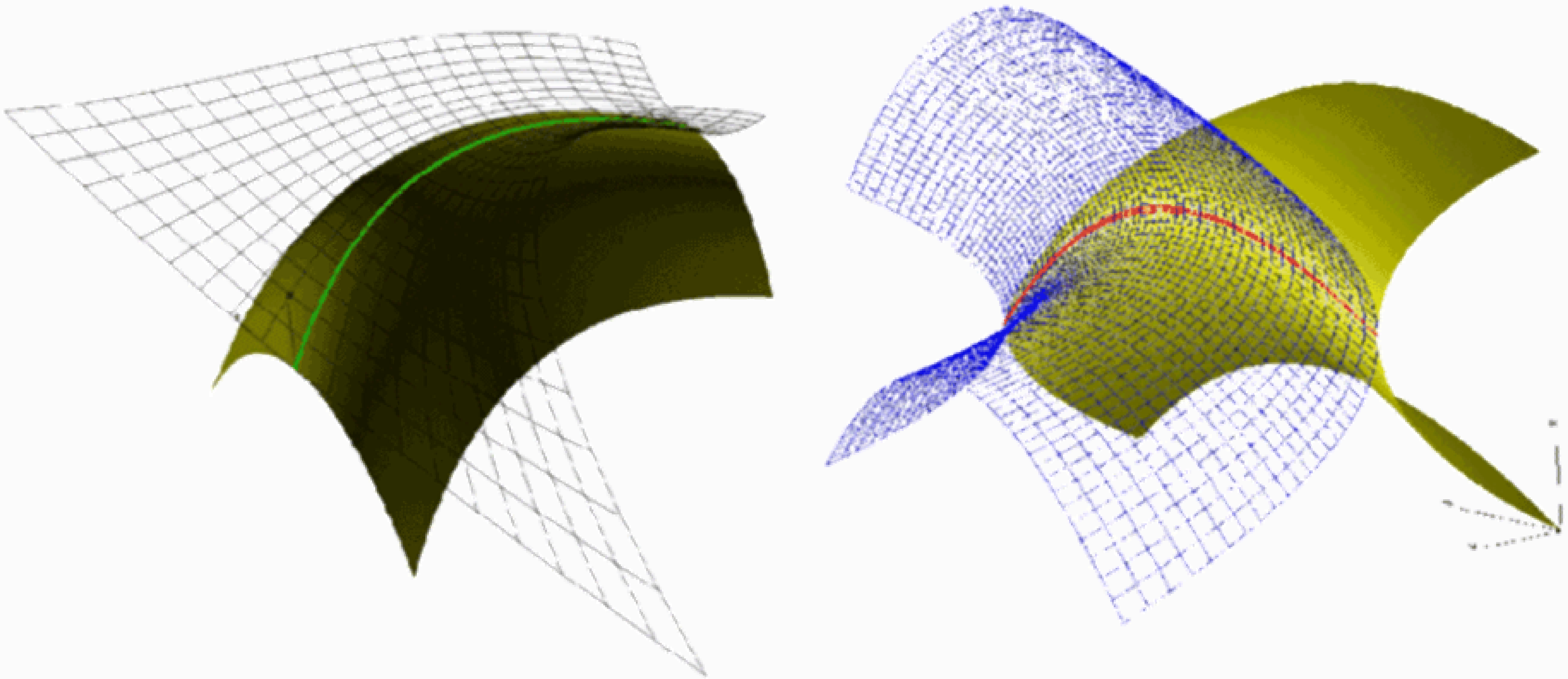
Two *bi-cubic* surfaces



Self intersection of
a *bi-cubic* surface

Results & Examples

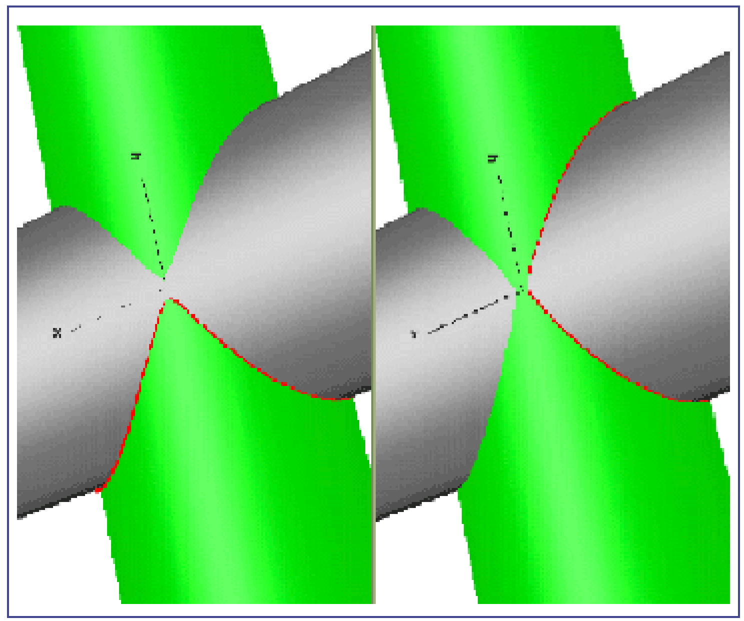
Error Bounds in 3D Model Space (Tangential)

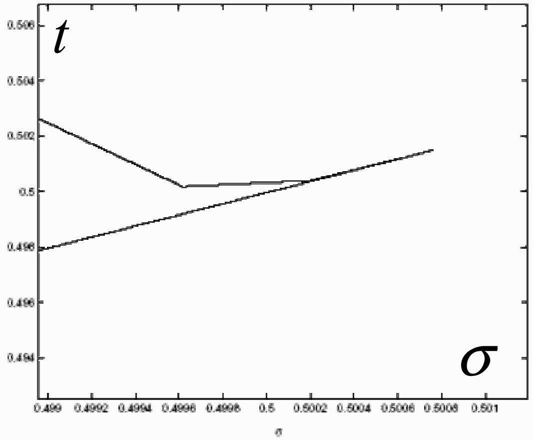


Tangential intersections of parametric surfaces

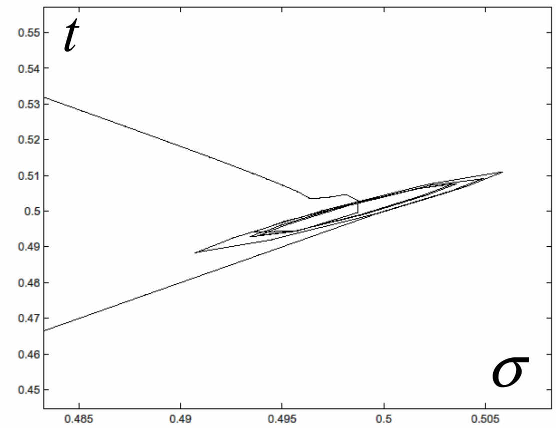
Results & Examples

Preventing Straying and Looping



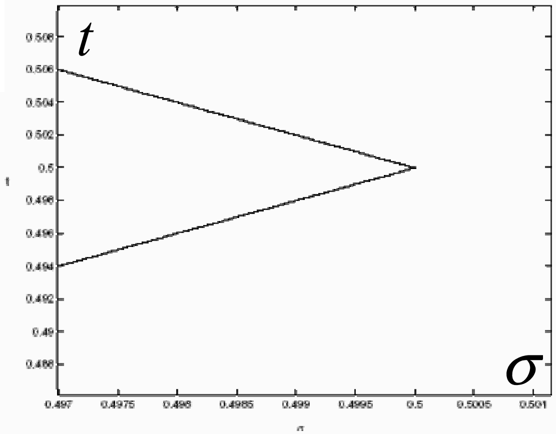


Adams-Bashforth



Runge-Kutta

Perturbation	Steps Required by the Method
+0.000003	1139
0.0	Singularity Reported
-0.000003	1303



Result from a validated interval scheme

Validated ODE solver can correctly trace the *intersection curve segment* even through closely spaced features, where standard methods fail.

Outline

- Problem Formulation
- Error Bounds in Parametric Space
- Error Bounds in 3D Model Space
- Results and Examples
- **Conclusions**

Conclusions

Merits

- We realize *validated error bounds* in *3D model space* which enclose the *true curve of intersection.*
- The scheme can prevent the phenomenon of *straying or looping*.
- The scheme can accommodate the errors in:
  - initial condition
  - rounding during digital computation
- *Validated error bounds* for surface intersection is essential in *interval boundary representation* for consistent *solid models* [5].

Conclusions
Limitations and Future Work

- Limitations
  - We assume that we have
    - Identified each intersection curve segment
    - *Strict error bound* on the starting point
  - Increasing width of the interval solutions due to
    - Rounding
    - Phenomenon of *wrapping*
- Scope for future work
  - Identification of all components
  - Accurate evaluation of *starting points* in each of the component

Acknowledgements

- National Science Foundation
- Prof. T. Sakkalis
- Prof. N. Nedialkov

References

1. Tracing surface intersections with a validated ODE system solver, Mukundan, H., Ko, K. H., Maekawa, T. Sakkalis, T., and Patrikalakis, N. M., Proceedings of the Ninth EG/ACM Symposium on Solid Modeling and Applications, G. Elber and G. Taubin, editors. Genova, Italy, June 2004. Eurographics Press.

2. Moore R. E.. Interval Analysis. Prentice-Hall, Englewood Cliffs, 1966.

3. Nedialkov N. S.. Computing the Rigorous Bounds on the Solution of an Initial Value Problem for an Ordinary Differential Equation. PhD thesis, University of Toronto, Toronto, Canada, 1999.

4. Patrikalakis N. M. and Maekawa T.. Shape Interrogation for Computer Aided Design and Manufacturing. Springer-Verlag, Heidelberg, 2002.

5. Sakkalis T., Shen G. and Patrikalakis N.M., Topological and Geometric Properties of Interval Solid Models, Graphical Models, 2001.

6. Grandine T. A., Klein F. W.: A new approach to the surface intersection problem. *Computer Aided Geometric Design 1*4, 2 (1997), 111–134. 650.