Global Knot Insertion Algorithms

Scott Schaefer
Department of Computer Science
Texas A&M University

Ron Goldman
Department of Computer Science
Rice University

Part I:

Introduction to Knot Insertion

Knot Insertion Algorithm

Input

- $T = \{t_1,\ldots,t_{\nu+n}\}$ -- knot sequence
- $P = \{P_0,\ldots,P_\nu\}$ -- control points
- $\Gamma = \{\tau_1,\ldots,\tau_{\mu+n}\}$ -- new knot sequence -- $\Gamma \supset T$

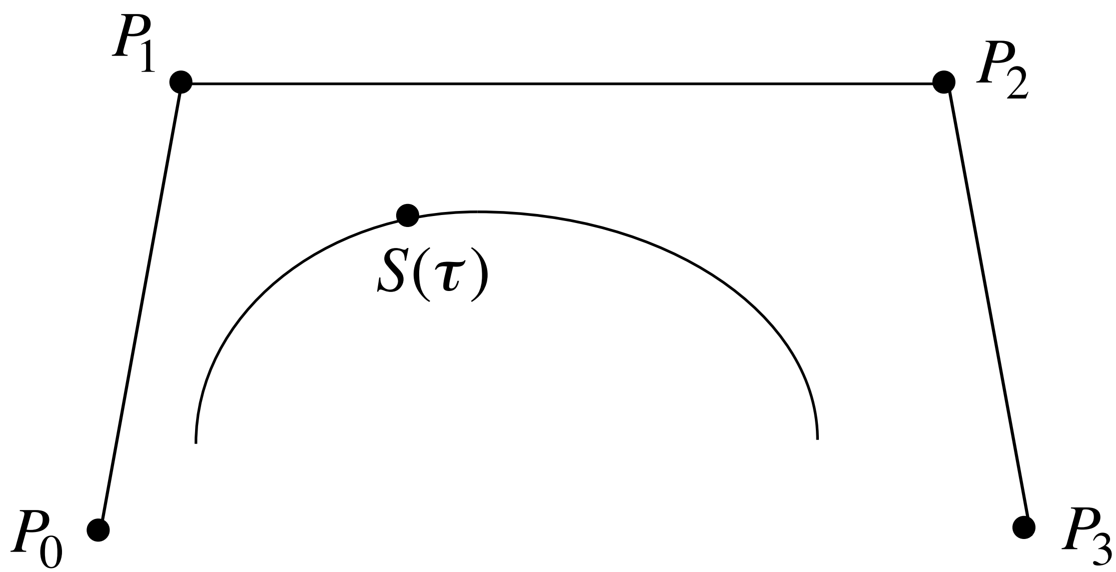
Output

- $Q = \{Q_0,\ldots,Q_\mu\}$ -- new control points

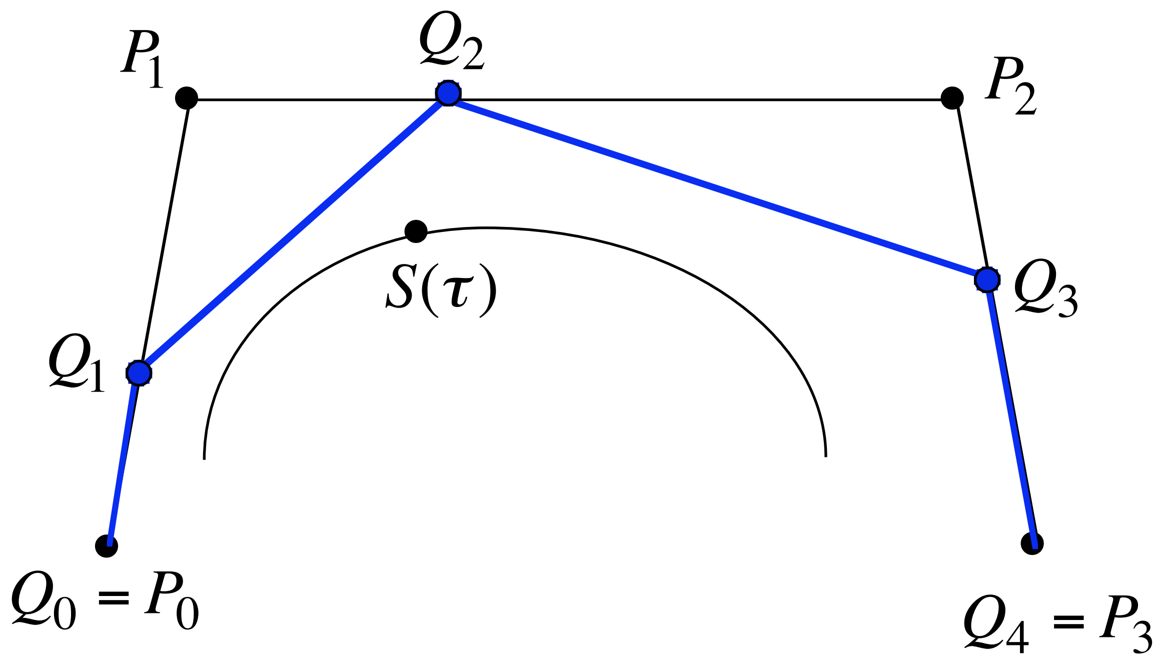
Constraint

- $\sum_{k=0}^{\nu} N_{k,n}(t \mid T) P_k = \sum_{k=0}^{\mu} N_{k,n}(\tau \mid \Gamma) Q_k$

B-spline Curve



Knot Insertion



Theorems

Existence

- All splines are B-splines.

Convergence

- The control polygons generated by knot insertion converge to the B-spline curve for the original control polygon as the knot spacing approaches zero.

Corner Cutting

- Knot insertion is a corner cutting procedure.

Applications of Knot Insertion

- Rendering
- Intersection
- Conversion from B-spline to Piecewise Bezier Form
- Proof of the Variation Diminishing Property

Types of Knot Insertion Algorithms

Local Knot Insertion

- $T = \{t_1, \ldots, t_{v+n}\}$
- $\Gamma = \{t_1, \ldots, t_k, u_{k,1}, \ldots, u_{k,d_k}, t_{k+1}, \ldots, t_{v+n}\}$

Global Knot Insertion

- $T = \{t_1, \ldots, t_{v+n}\}$
- $\Gamma = \{t_1, u_{1,1}, \ldots, u_{1,d_1}, t_2, \ldots, t_{v+n-1}, u_{v+n-1,1}, \ldots, u_{v+n-1,d_{v+n-1}}, t_{v+n}\}$

Examples of Knot Insertion Algorithms

Local Knot Insertion Algorithms

- Boehm’s Algorithm
- Oslo Algorithm
- Sablonniere’s Algorithm
- Factored Knot Insertion

Global Knot Insertion Algorithms

- Chaikin’s Algorithm
- Lane-Riesenfeld Algorithm
- Goldman-Warren Algorithm
- Schaefer’s Algorithm -- NEW

Myths

Global Knot Insertion

- Works Only for Uniform Knot Sequences
  - -- Knots in Arithmetic Progression -- Lane-Riesenfeld (1973)

    $$t_{k+1} = t_k + \alpha$$

  - -- Knots in Geometric or Affine Progression -- Goldman-Warren (1993)

    $$t_{k+1} = \beta t_k + \alpha$$

- Does not Apply to Arbitrary Knot Sequences

Blossoming

- Provides Only New Proofs of Already Known Results
  - -- No New Results
  - -- No New Insights

Part II:

Local Knot Insertion Algorithms

Knot Insertion Algorithms from Blossoming

- Boehm's Algorithm -- Inserts one new knot at a time

- Oslo Algorithm -- Computes one new control point at a time

- Sablonniere's Algorithm -- Local change of basis algorithm

- Factored Knot Insertion -- Forward differencing for knot insertion

Blossoming

Symmetry

- $p(u_1,\ldots,u_n) = p(u_{\sigma(1)},\ldots,u_{\sigma(n)})$ for any permutation σ of $\{1,\ldots,n\}$

Multiaffine

- $p(u_1,\ldots,(1-\alpha)u_k + \alpha w_k,\ldots,u_n) = (1-\alpha)\,p(u_1,\ldots,u_k,\ldots,u_n) + \alpha p(u_1,\ldots,w_k,\ldots,u_n)$

Diagonal

- $p(\underbrace{t,\ldots,t}_{n}) = P(t)$

Dual Functional Property

- $P(t) = \sum_k N_{k,n}(t)P_k \;\Rightarrow\; P_k = p(t_{k+1},\ldots,t_{k+n})$ (B-spline Curves)

- $P(t) = \sum_k B_k^n(t)P_k \;\Rightarrow\; P_k = p(\underbrace{a,\ldots,a}_{n-k},\underbrace{b,\ldots,b}_{k})$ (Bezier Curves)

Properties of the Blossom

Existence and Uniqueness

- Every Degree n Polynomial $P(t)$ has a Unique Blossom $p(u_1,\ldots,u_n)$

Power Law

- Each Parameter $u_1,\ldots,u_n$ Appears to at Most the First Power
- Equivalent to Multiaffine Axiom

Examples (Existence)

Monomials

$$P(t) = 1 \quad \Rightarrow \quad p(u_1, u_2, u_3) = 1$$

$$P(t) = t \quad \Rightarrow \quad p(u_1, u_2, u_3) = \frac{u_1 + u_2 + u_3}{3}$$

$$P(t) = t^2 \quad \Rightarrow \quad p(u_1, u_2, u_3) = \frac{u_1 u_2 + u_2 u_3 + u_3 u_1}{3}$$

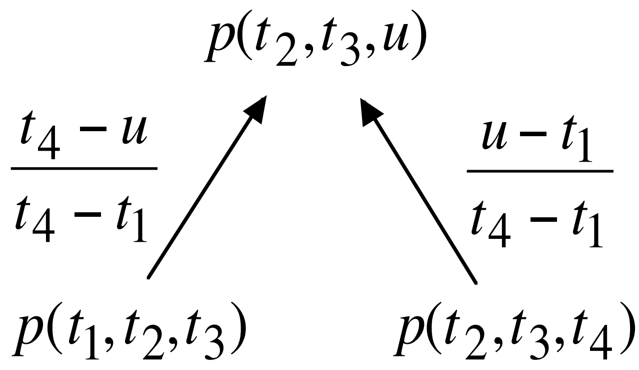
$$P(t) = t^3 \quad \Rightarrow \quad p(u_1, u_2, u_3) = u_1 u_2 u_3$$

Cubics

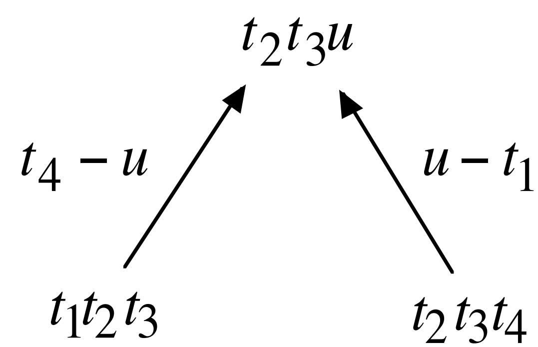
$$P(t) = a_3 t^3 + a_2 t^2 + a_1 t + a_0$$

$$p(u_1, u_2, u_3) = a_3 u_1 u_2 u_3 + a_2 \frac{u_1 u_2 + u_2 u_3 + u_3 u_1}{3} + a_1 \frac{u_1 + u_2 + u_3}{3} + a_0$$

Blossoming Diagrams -- Multiaffine Property



Normalized

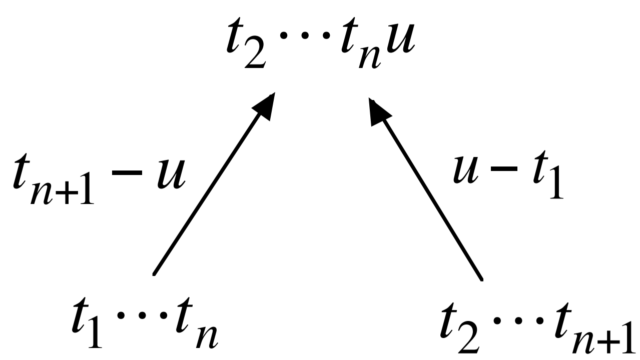


Unnormalized

$$u = \frac{t_4 - u}{t_4 - t_1} t_1 + \frac{u - t_1}{t_4 - t_1} t_4 \quad \Rightarrow \quad p(t_2, t_3, u) = \frac{t_4 - u}{t_4 - t_1} p(t_1, t_2 t_3) + \frac{u - t_1}{t_4 - t_1} p(t_2, t_3, t_4)$$

$$p(t_2, t_3, u) = \frac{t_4 - u}{t_4 - t_1} p(t_1, t_2 t_3) + \frac{u - t_1}{t_4 - t_1} p(t_2, t_3, t_4) \quad \Leftrightarrow \quad t_2 t_3 u = \frac{t_4 - u}{t_4 - t_1} t_1 t_2 t_3 + \frac{u - t_1}{t_4 - t_1} t_2 t_3 t_4$$

Blossoming Diagrams -- Multiaffine Property

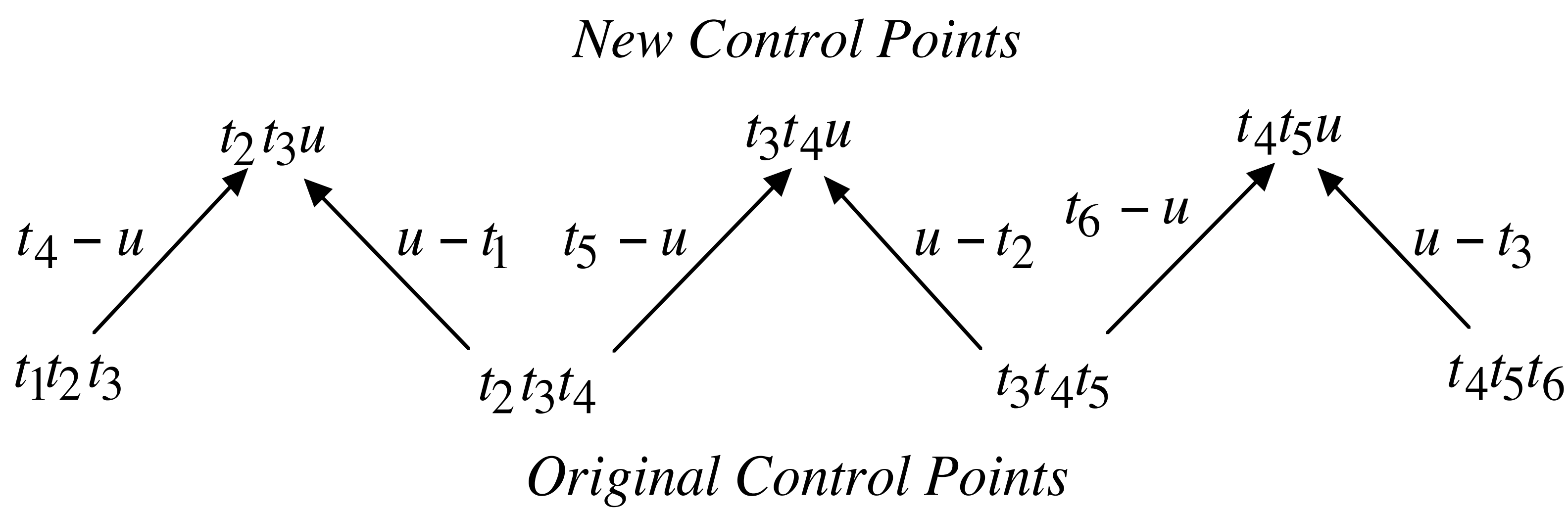


$$u = \frac{t_{n+1} - u}{t_{n+1} - t_1} t_1 + \frac{u - t_1}{t_{n+1} - t_1} t_{n+1}$$

$$p(t_2, \ldots, t_n, u) = \frac{t_{n+1} - u}{t_{n+1} - t_1} p(t_1, \ldots, t_n) + \frac{u - t_1}{t_{n+1} - t_1} p(t_2, \ldots, t_{n+1})$$

$$t_2 \cdots t_n u = \frac{t_{n+1} - u}{t_{n+1} - t_1} t_1 \cdots t_n + \frac{u - t_1}{t_{n+1} - t_1} t_2 \cdots t_{n+1}$$

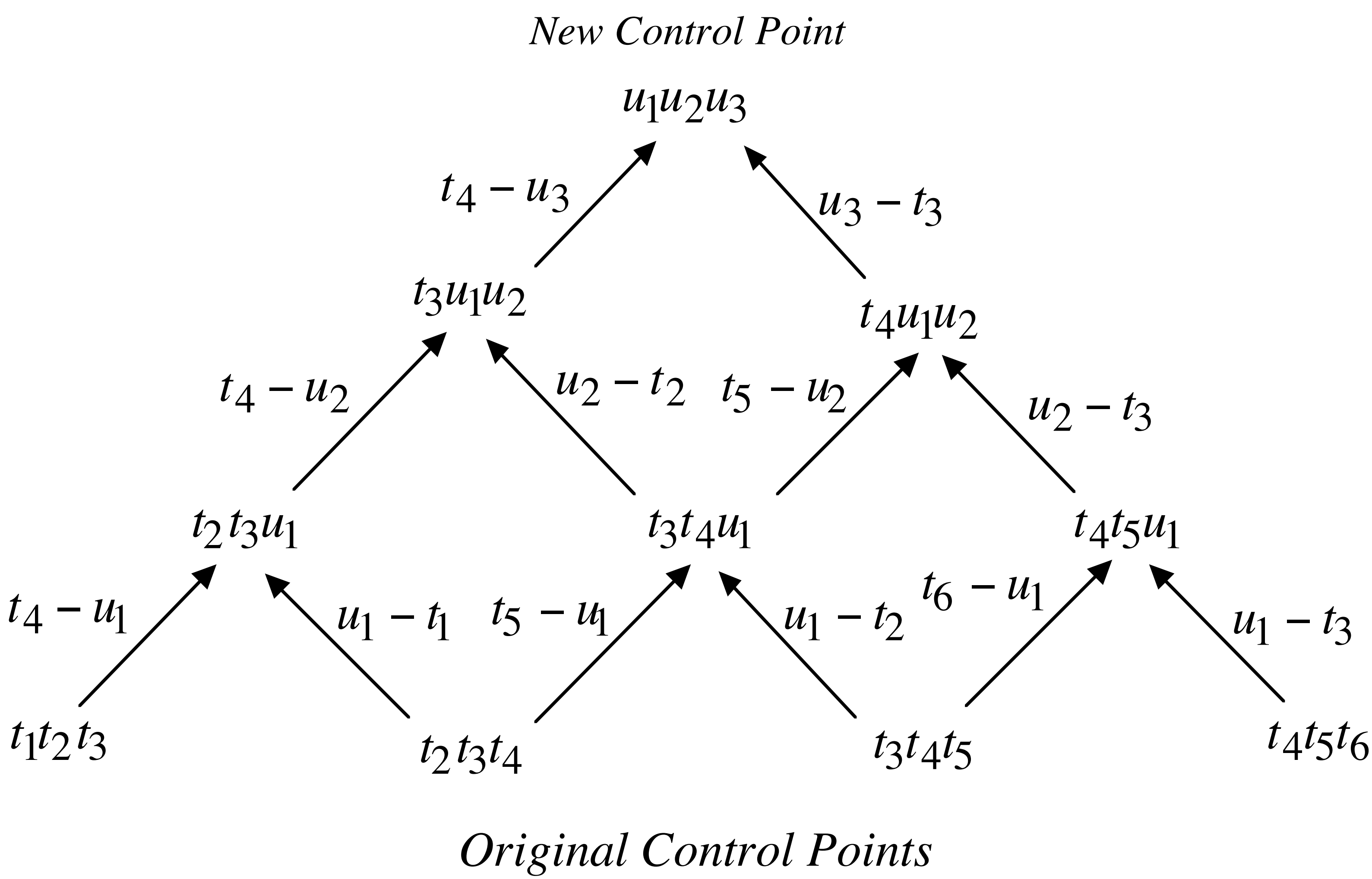
Boehm's Algorithm



Original Knot Sequence: $\ldots, t_1, t_2, t_3, t_4, t_5, t_6, \ldots$

New Knot Sequence: $\ldots, t_1, t_2, t_3, u, t_4, t_5, t_6, \ldots$

Oslo Algorithm



Original Knot Sequence: $\ldots, t_1, t_2, t_3, t_4, t_5, t_6, \ldots$

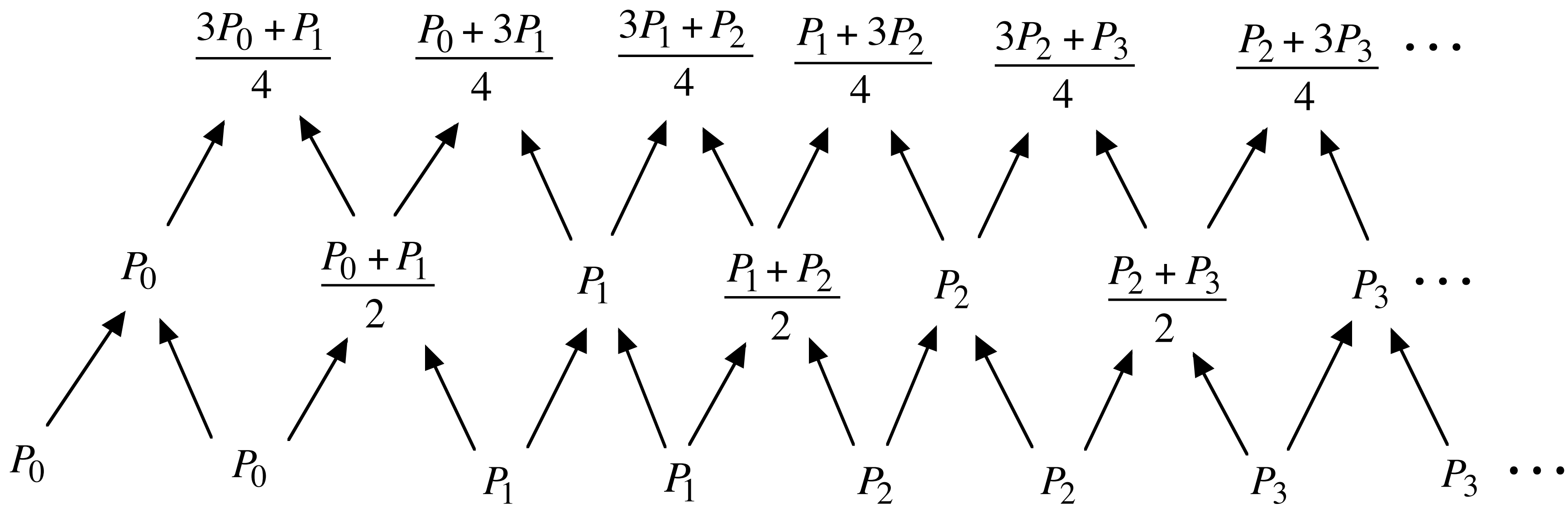
New Knot Sequence: $\ldots, t_1, t_2, t_3, u_1, \ldots, u_d, t_4, t_5, t_6, \ldots$

Part III:

Global Knot Insertion Algorithms

Chaikin's Algorithm:

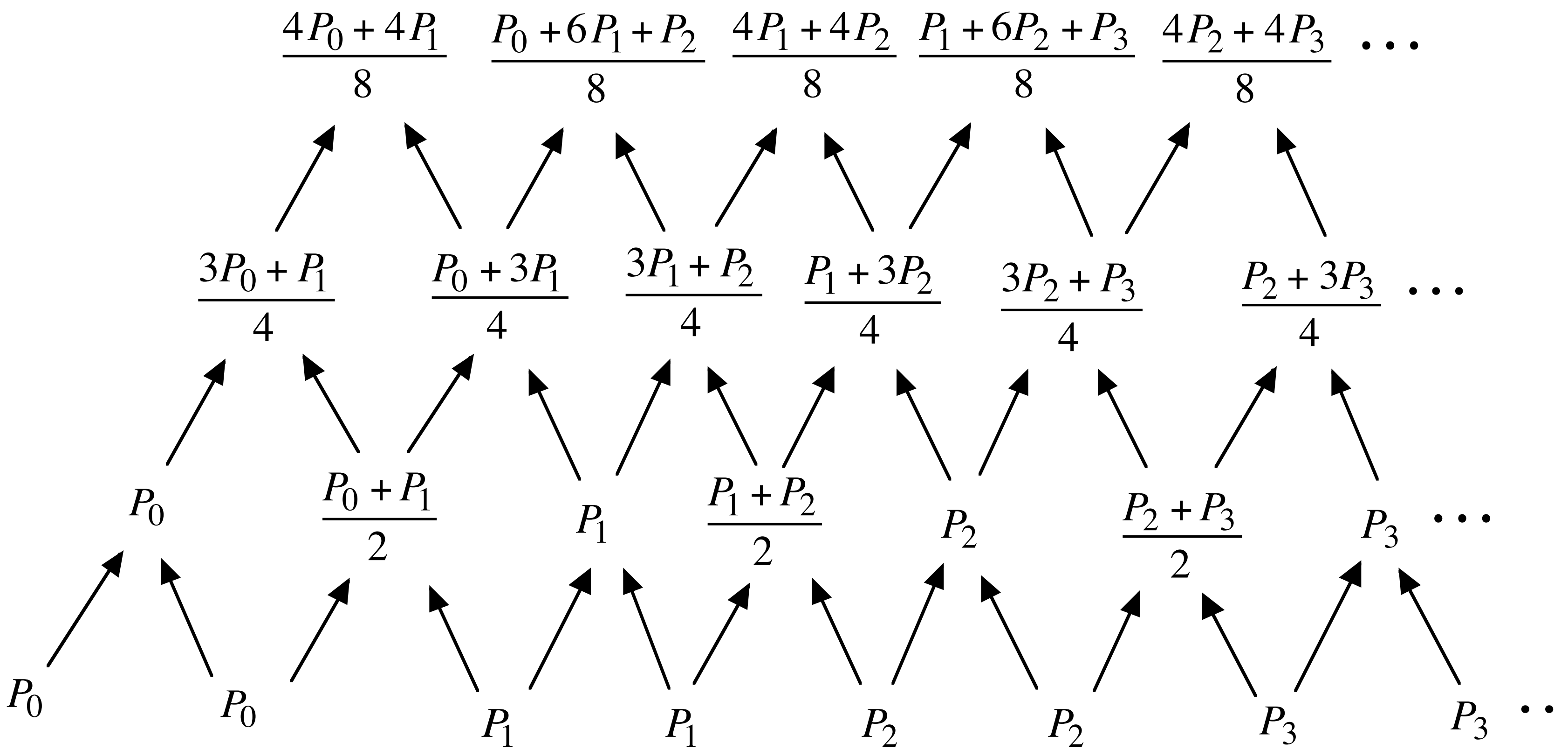
Quadratic B-splines -- Uniform Knots



Split and Average

Lane-Riesenfeld Algorithm:

Cubic B-splines -- Uniform Knots



Split and Average

Proofs of Lane-Riesenfeld Algorithm

1. *Continuous Convolution of B-spline Basis Functions*

J. LANE and R. RIESENFELD (1980): A theoretical development for the computer generation and display of piecewise polynomial surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence 2*, pp. 35–46.

2. *De Boor Recurrence (Induction)*

R. GOLDMAN and J. WARREN (1993): An extension of Chaikin's algorithm to B-spline curves with knots in geometric progression, *CVGIP: Graphical Models and Image Processing*, Vol. 35, pp. 58-62.

$2'$. *De Boor Recurrence (Induction)* $\Rightarrow$ *Oslo Algorithm*

H. PRAUTZSCH (1984): A short proof of the Oslo Algorithm, *Computer Aided Geometric Design*, Vol. 1, pp. 95-96.

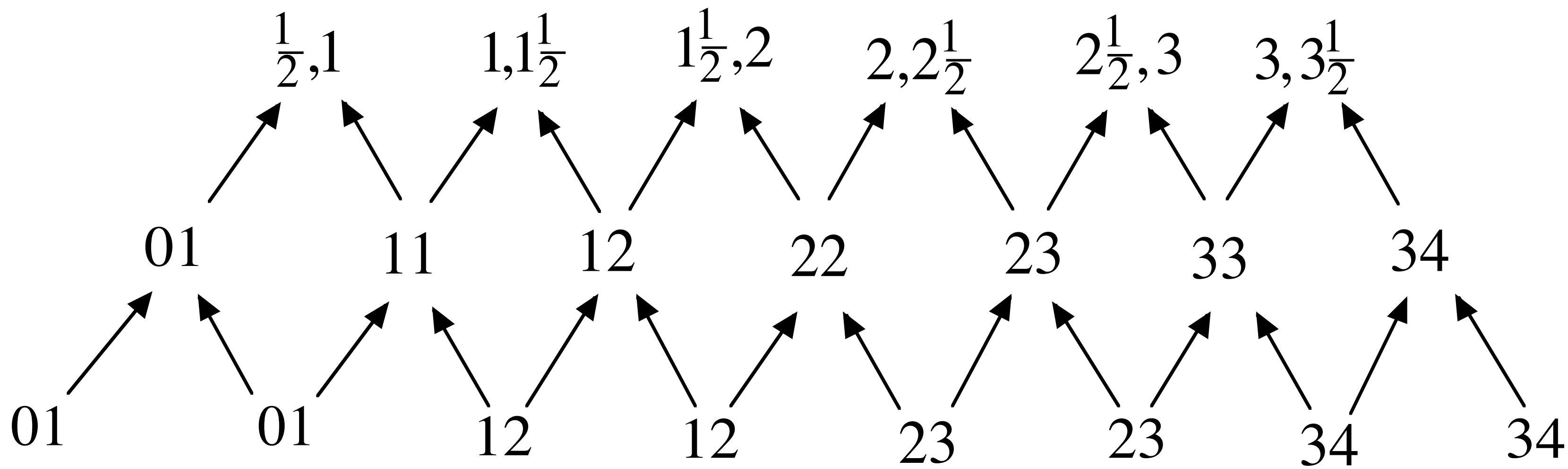
New Proof of Lane-Riesenfeld Algorithm

3. *Blossoming*

E. VOUGA and R. GOLDMAN (2006): Two Blossoming Proofs of the Lane-Riesenfeld Algorithm -- *Dagstuhl Conference on Geometric Modeling 2005*, to appear in *Computing*.

Lane-Riesenfeld Algorithm -- Blossoming Interpretation

Quadratic B-splines



Uniform Knots

<u>Lane-Riesenfeld Algorithm</u>

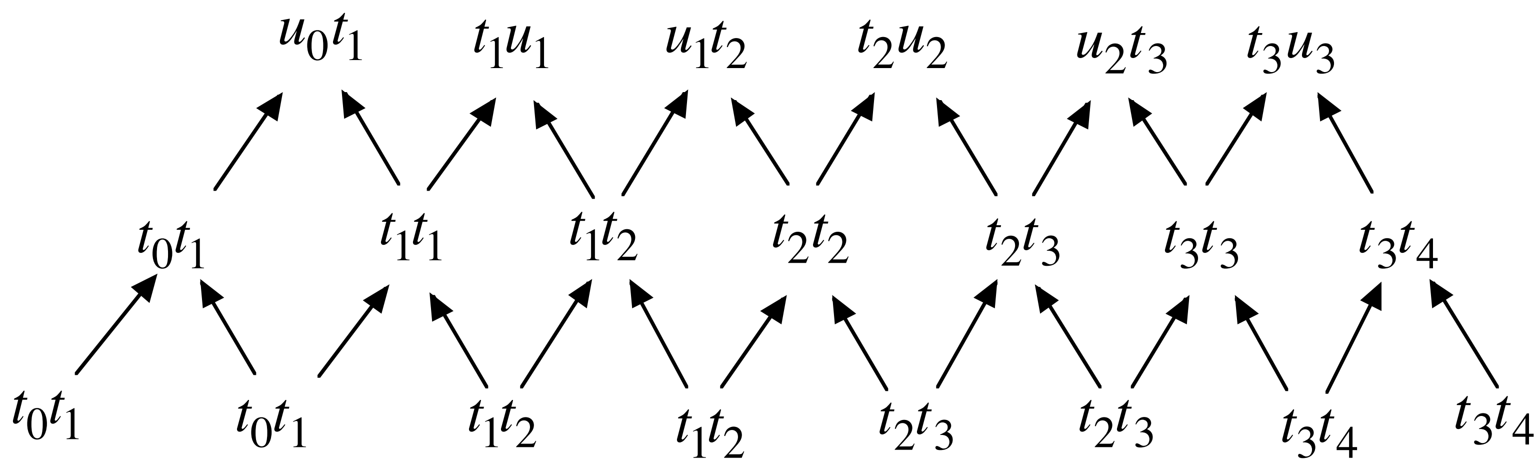
Quadratic B-splines

1. Double the Control Points

2. Convert to Piecewise Bezier Form

3. Insert New Knots Using Boehm's Knot Insertion Algorithm

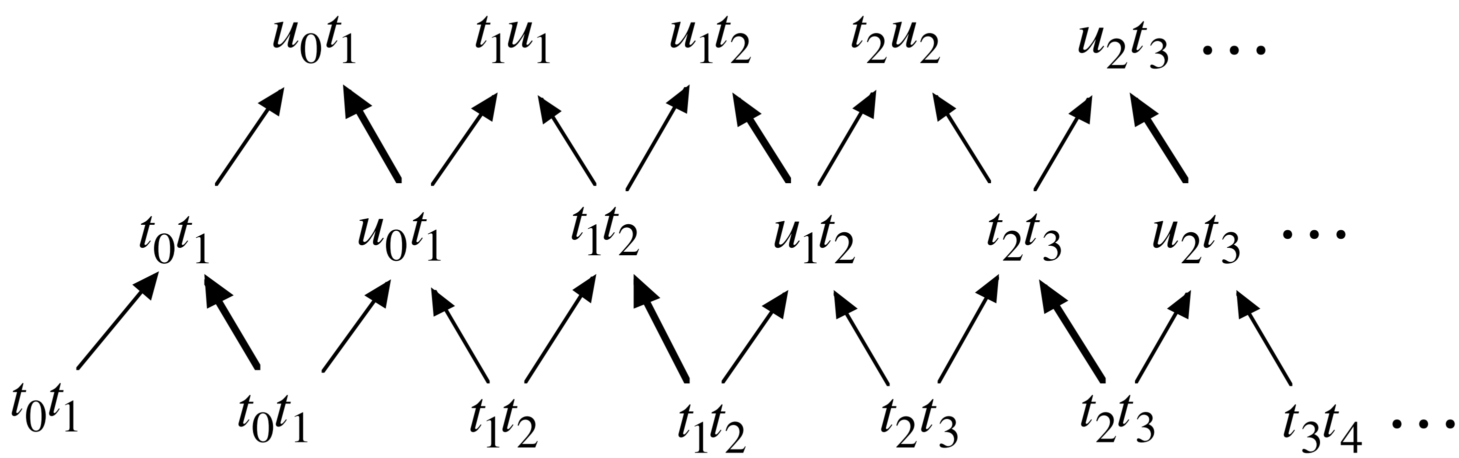
Lane-Riesenfeld Algorithm -- Quadratic B-splines



Arbitrary Knots

$$u = \frac{t_{k+1} - u}{t_{k+1} - t_k} t_k + \frac{u - t_k}{t_{k+1} - t_k} t_{k+1}$$

<u>Schaefer's Algorithm -- Quadratic B-splines</u>



Arbitrary Knots

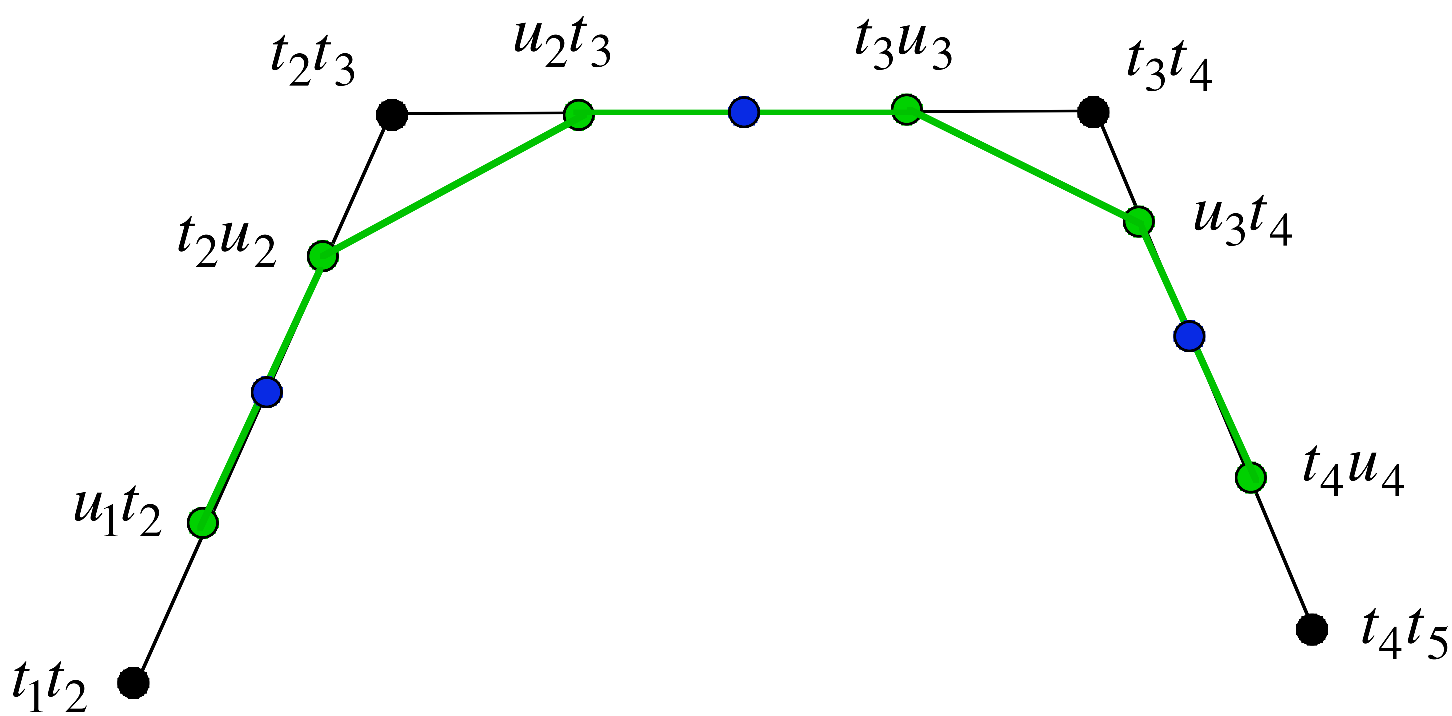
$$u = \frac{t_{k+1} - u}{t_{k+1} - t_k} t_k + \frac{u - t_k}{t_{k+1} - t_k} t_{k+1}$$

Quadratic B-splines

Schaefer's Algorithm

- Inserts New Knots in First Round
- Not Necessary to Convert to Piecewise Bezier Form
- Faster than Lane-Riesenfeld -- Half the Work in the Second Round

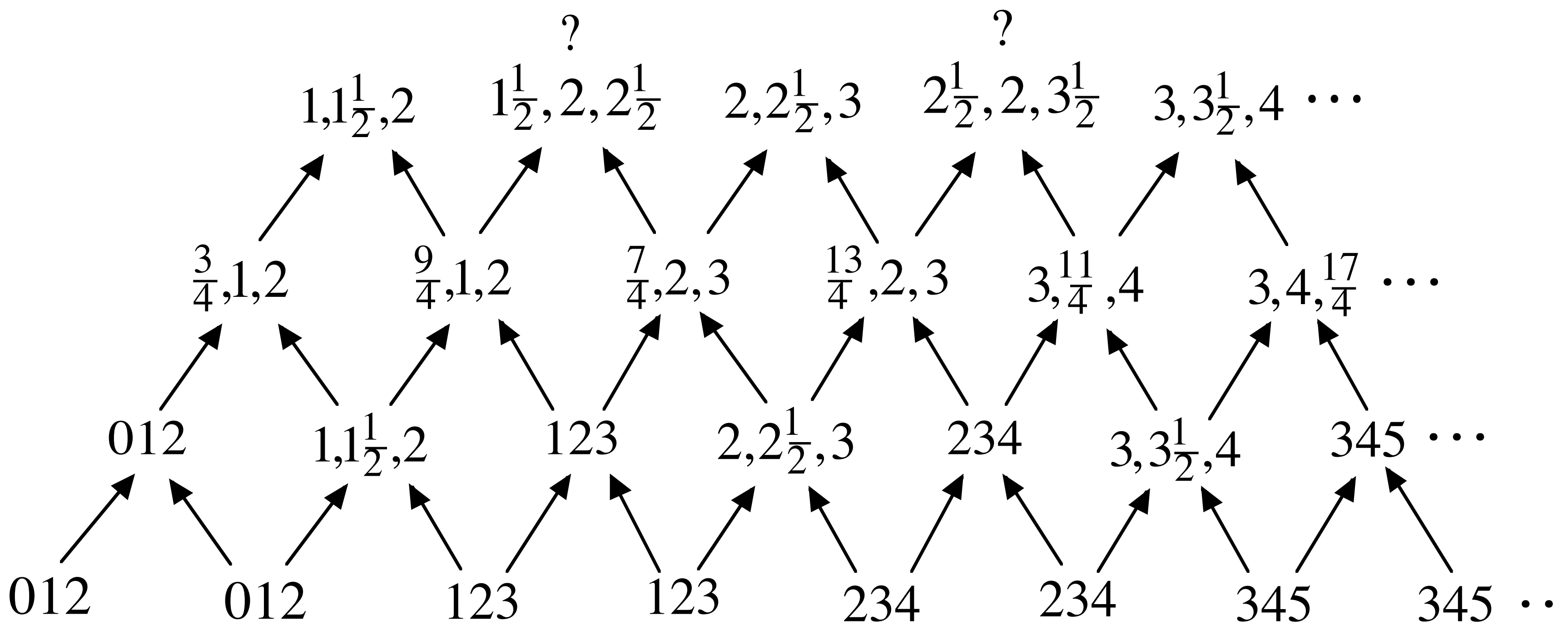
Knot Insertion: Quadratic B-splines



Lane – Riesenfeld: ● $= t_k t_{k+1}$

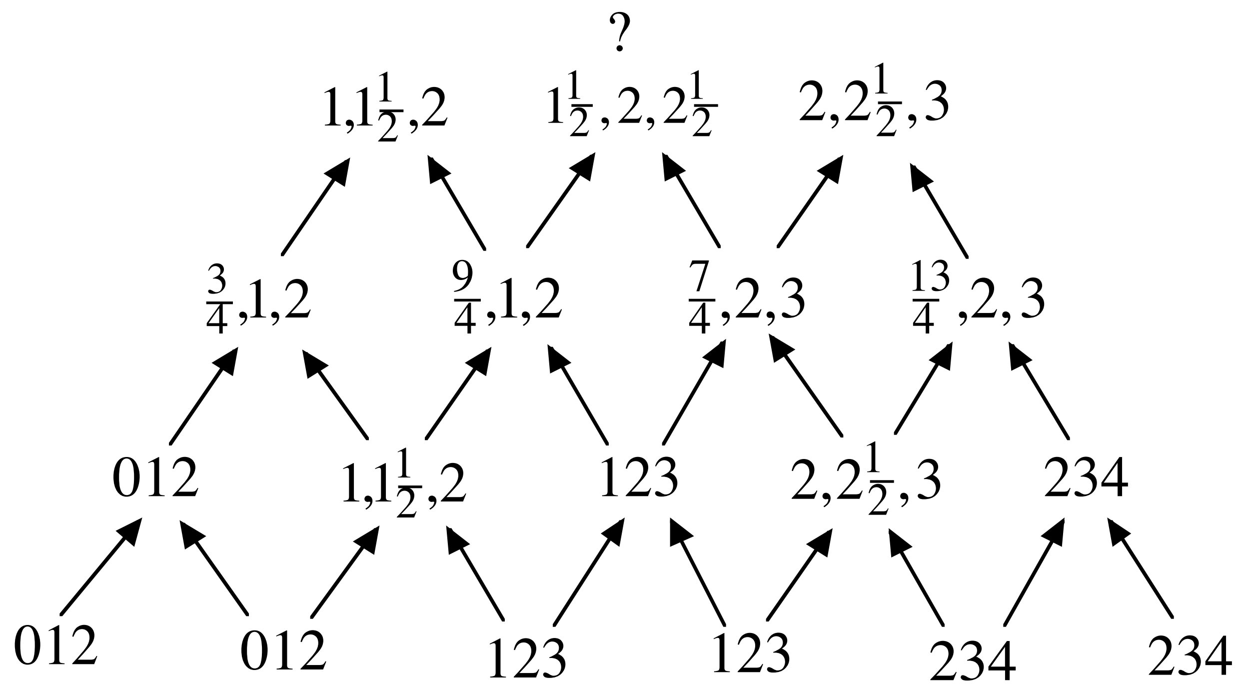
Schaefer: ● $= u_k t_{k+1}$

Lane-Riesenfeld Algorithm -- Cubic B-splines



Uniform Knots

Lane-Riesenfeld Algorithm -- Cubic B-splines



$$\left(\tfrac{9}{4},1,2\right) = \frac{1}{2}\left(1,1\tfrac{1}{2},2\right) + \frac{1}{2}(1,2,3)$$

$$\left(\tfrac{7}{4},2,3\right) = \frac{1}{2}(1,2,3) + \frac{1}{2}\left(2,2\tfrac{1}{2},3\right)$$

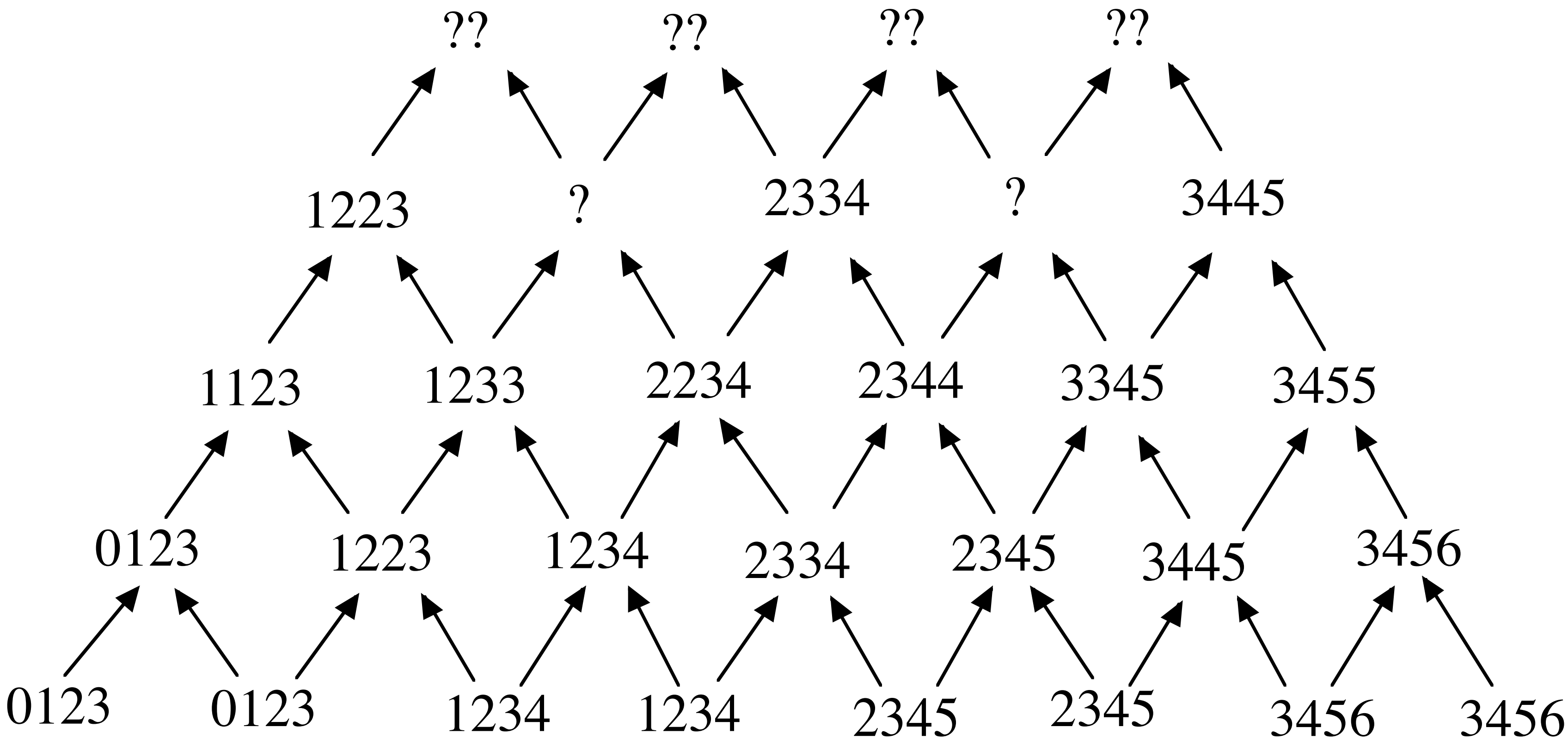
$$\Rightarrow$$

$$? = \frac{1}{4}\left(1,1\tfrac{1}{2},2\right) + \frac{2}{4}(1,2,3) + \frac{1}{4}\left(2,2\tfrac{1}{2},3\right)$$

$$? = \frac{1}{4}\left(1,1\tfrac{1}{2},2\right) + \frac{3}{4}\left(\frac{2}{3}(1,2,3) + \frac{1}{3}\left(2,2\tfrac{1}{2},3\right)\right)$$

$$? = \frac{1}{4}\left(1,1\tfrac{1}{2},2\right) + \frac{3}{4}\left(1\tfrac{1}{2},2,3\right) = \left(1\tfrac{1}{2},2,2\tfrac{1}{2}\right)$$

Lane-Riesenfeld Algorithm -- Quartic B-splines

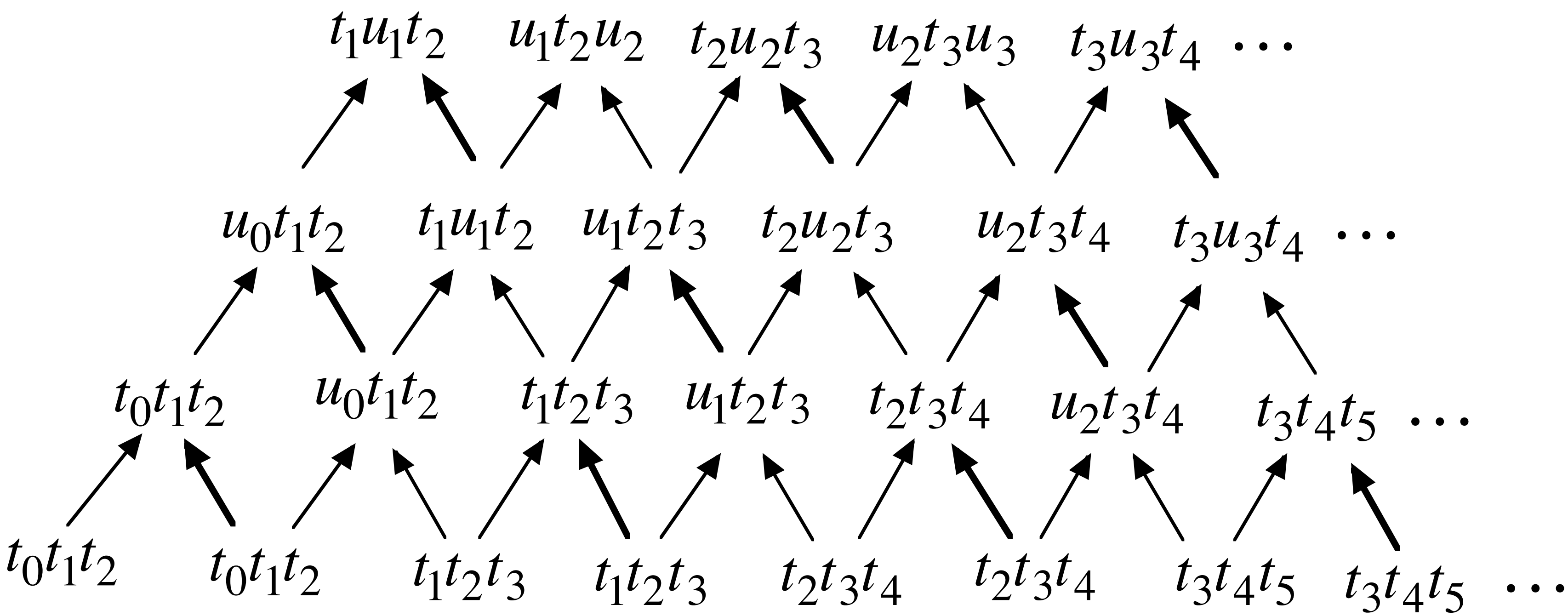


Uniform Knots

Lane-Riesenfeld Algorithm

- Build Algorithm for Next Degree Atop Algorithm for Previous Degree
- Append One Additional Round of Averaging
- Harder and Harder to Prove by Blossoming

Schaefer's Algorithm -- Cubic B-splines



Arbitrary Knots

$$u = \frac{t_k - u}{t_k - t_j} t_j + \frac{u - t_j}{t_k - t_j} t_k$$

Schaefer's Algorithm

Cubic B-splines

- Build Atop Algorithm for Quadratic B-splines

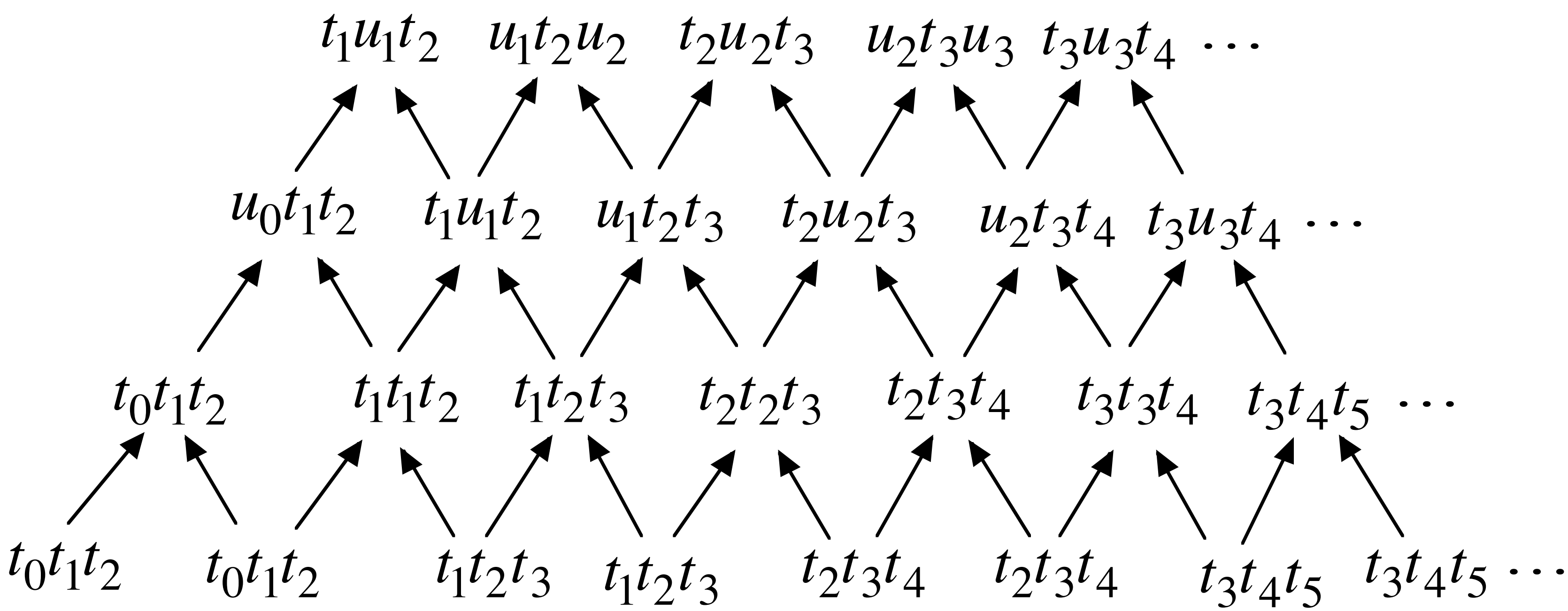
- Append Next Original Knot to Each of the Blossoms on the First Two Stages
  -- Example: $u_0 t_1 \rightarrow u_0 t_1 t_2$

- Promote Every Other Point to the Next Stage with No Additional Computation

- Introduce New Knots Using the Multiaffine Property of the Blossom

- Easy to Prove by Blossoming

Alternative Algorithm -- Cubic B-splines



Arbitrary Knots

$$u = \frac{t_k - u}{t_k - t_j} t_j + \frac{u - t_j}{t_k - t_j} t_k$$

Problem

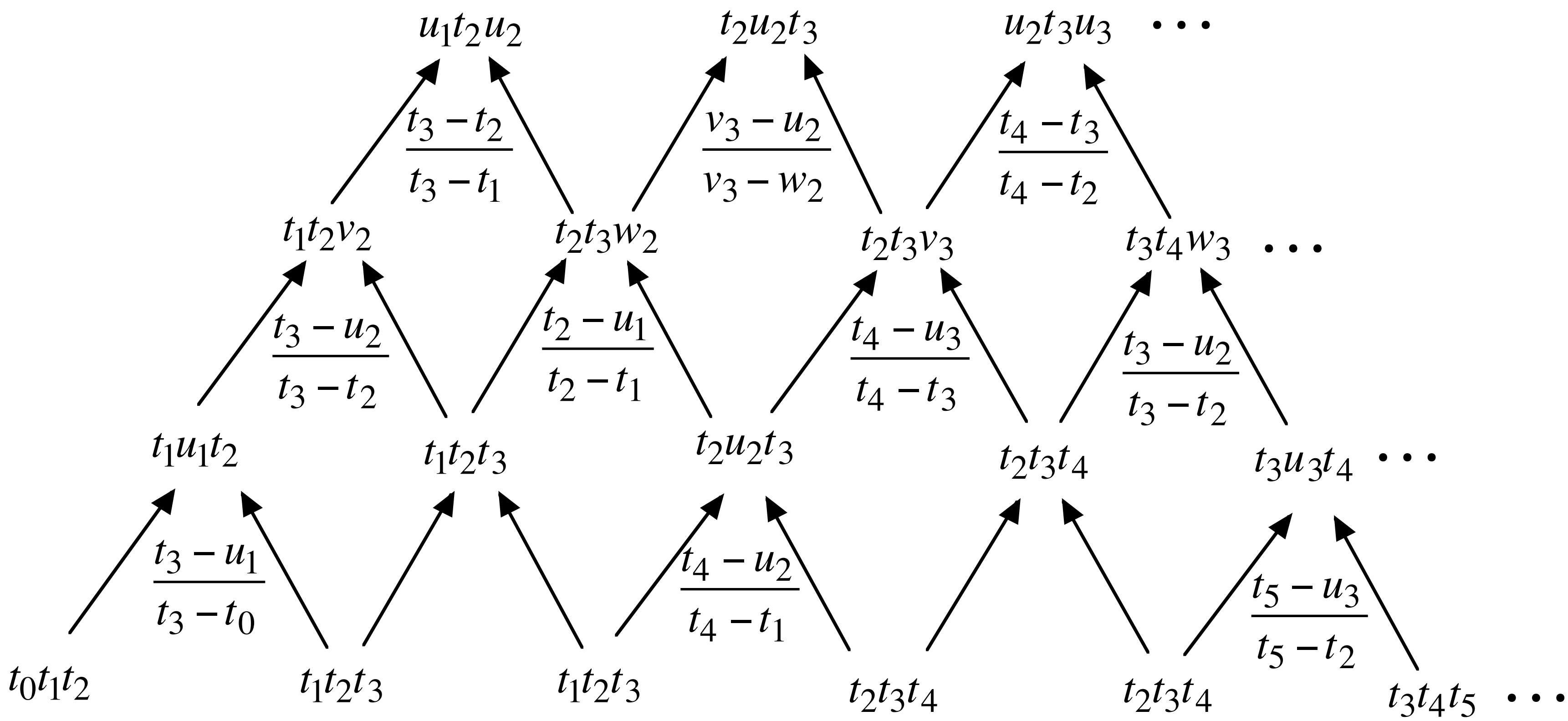
Observation

- Schaefer's Algorithm does not Reduce to the Lane-Riesenfeld Algorithm when the Knots are Uniformly Spaced.

Questions

- Does there Exist a Global Knot Insertion Algorithm that Reduces to the Lane-Riesenfeld Algorithm for Uniform Knots?
- If Such an Algorithm Exists, is it Unique?

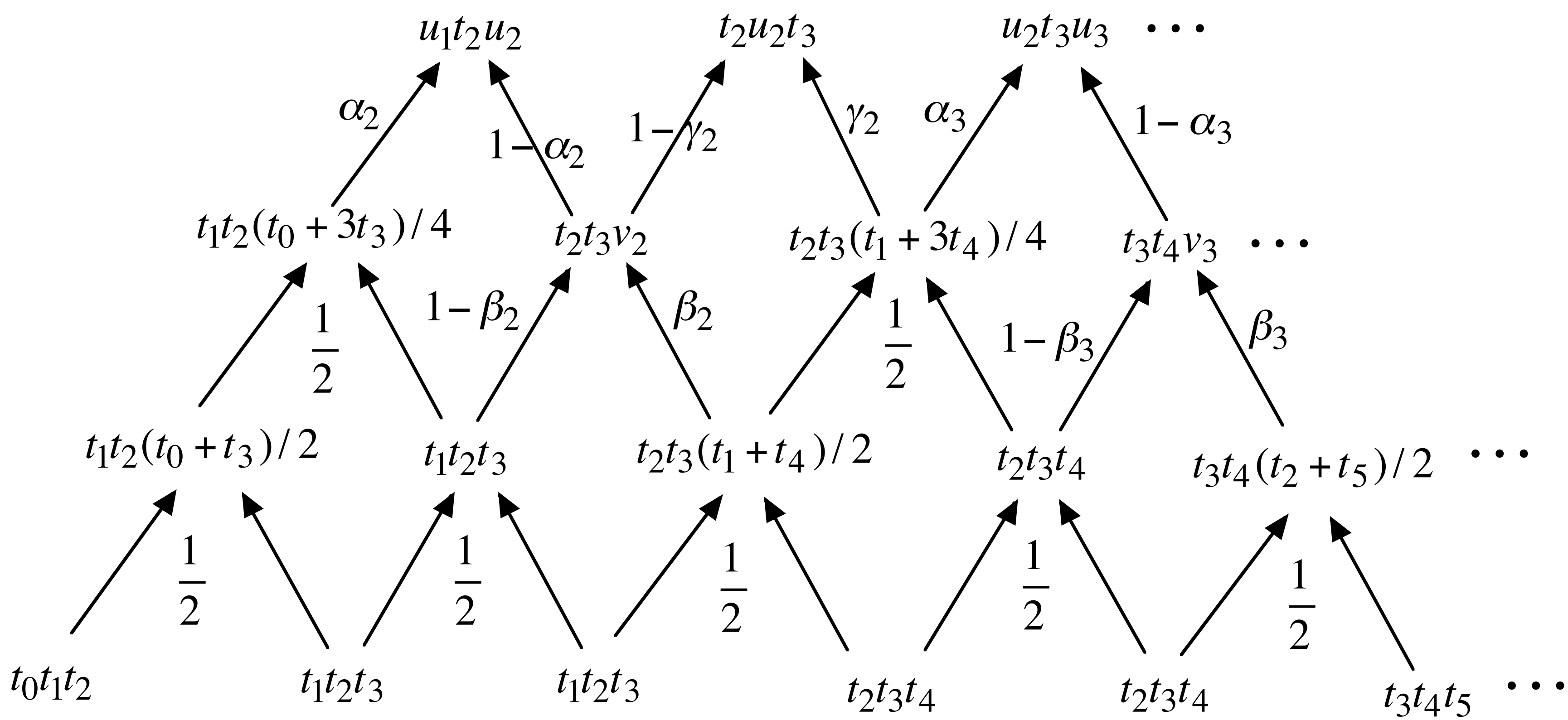
Lane-Riesenfeld Algorithm:
Cubic B-splines -- Arbitrary Knots



$$v_k = t_{k+1} - \frac{(t_{k+1}-u_{k-1})(t_{k+1}-u_k)}{(t_{k+1}-t_k)} \qquad w_k = u_k - \frac{(u_k-t_{k-1})(t_k-u_{k-1})}{(t_k-t_{k-1})}.$$

Lane-Riesenfeld Algorithm:

Cubic B-splines -- Arbitrary Knots



Parameters for Lane-Riesenfeld Algorithm

$$\alpha_k = \frac{4(t_{k+1} - u_{k-1})(t_{k+1} - u_k)}{(t_{k+1} - t_{k-2})(t_{k+1} - t_{k-1})}$$

$$\beta_k = \frac{2(u_k - t_{k-1})(u_{k-1} - t_{k-1})}{(t_{k+2} - t_{k-1})(t_{k+1} - t_{k-1})(1 - \alpha_k)}$$

$$\gamma_k = \frac{4(u_k - v_k)}{t_{k-1} + 3t_{k+2} - 4v_k}$$

$$v_k = t_{k-1} + \frac{\beta_k (t_{k+2} - t_{k-1})}{2}.$$

Summary

Global Knot Insertion Algorithms

- Exist for Arbitrary Knot Sequences
- Are Easily Derived from Blossoming

Open Problems

1. *Find a Global Knot Insertion Algorithm for Arbitrary Degree that Reduces to the Lane-Riesenfeld Algorithm for Uniform Knots.*

2. *Find the Simplest Knot Insertion Algorithm for Arbitrary Degree that Reduces to the Lane-Riesenfeld Algorithm for Uniform Knots.*

    - *How Complicated are the Labels Along the Edges?*