Embeddings of Stick Models

Ian Barnett

August 6, 2009

1Supported by the NSF grant number DMS-0849637
4 planes $H_1, H_2, H_3, H_4 \in \mathbb{R}^6$. All through the origin.

We know:

- $H_1 \leftarrow \text{intersects} \rightarrow H_2$
- $H_1 \leftarrow \text{intersects} \rightarrow H_3$
- $H_4 \leftarrow \text{intersects} \rightarrow H_2$
- $H_4 \leftarrow \text{intersects} \rightarrow H_3$

GOAL: Define a set of planes which have these properties.

This problem generalizes.
Stick Models

- Stick models represent this same information – we just crunch down the dimension.
- Stick models represent *lines* in *projective space*
Random Vector Generation in a Plane

- Area of diagonal band
  >
- Area of vertical band
For uniformly random $x_i \in [-1, 1], \ i = 1, 2$, we accept when $\sqrt{x_1^2 + x_2^2} \leq 1$, i.e. only when in circle.
For higher dimensions, same procedure.
Think *unit sphere* in *cube*.
For uniformly random \( x_i \in [-1, 1], \ i = 1, 2, \ldots, n, \) we accept when 
\[
\sqrt{x_1^2 + x_2^2 + \ldots + x_n^2} \leq 1.
\]
Let plane 1 be the $x_1 x_2$-plane in $\mathbb{R}^6$
The Algorithm: Embedding Plane 1

- **Completed:** 1

- **No intersections with**
  **Completed:** 4

- **One intersection with**
  **Completed:** 2, 3

- **Two intersections with**
  **Completed:**
The Algorithm: Embedding Plane 2

- Generate a random vector in plane 1 and in $\mathbb{R}^6$. These vectors span plane 2.
The Algorithm: Embedding Plane 2

- Completed: 1, 2

- No intersections with
  Completed:

- One intersection with
  Completed: 3, 4

- Two intersections with
  Completed:
Generate a random vector in plane 2 and in $\mathbb{R}^6$. These vectors span plane 4.
The Algorithm: Embedding Plane 4

- **Completed:** 1, 2, 4

- No intersections with
  **Completed:**

- One intersection with
  **Completed:**

- Two intersections with
  **Completed:** 3
Generate a random vector in plane 1 and in plane 4. These vectors span plane 3.
The Algorithm: Embedding Plane 3

► Completed: 1, 2, 3, 4

► No intersections with
Completed:

► One intersection with
Completed:

► Two intersections with
Completed:
The Algorithm: Pseudocode

- Set **Incomplete** to be the set of all sticks.
- Set **Completed** to be the set of all placed(embedded) sticks (initially empty).
- Set **O** to be the set of all sticks not yet embedded with one intersection with a stick in **Completed**.
- Set **T** to be the set of all sticks not yet embedded with two intersections with two sticks in **Completed**.
- While **Incomplete** is not empty:
  - If **T** is not empty, place the first stick in it.
  - Else if **O** is not empty, place the first stick in it.
  - Else place a stick from **Incomplete**
  - Remove placed stick from **Incomplete**.
  - Add placed stick to **Completed**.
  - Modify **O** and **T** accordingly.