Array-based compact data structures for triangle meshes

ISAAC, Yokohama, 6-8 December 2011

Luca Castelli Aleardi

Olivier Devillers

Geometriques - INRIA Sophia
Compact Data Structures: motivation and goal

St. Matthew (Stanford’s Digital Michelangelo Project, 2000)

- 6 Giga bytes (for storing on disk)
- 186 millions vertices

David statue (Stanford’s Digital Michelangelo Project, 2000)

- 2 billions polygons
- 32 Giga bytes (without compression)

Geometric object

vertex
triangle

1 reference to a triangle
3 references to vertices
3 references to triangles

Connectivity information

$\log n$ or 32 bits

416\,n \text{ bits}

13\,n \log n

between 30 et 96 bits/vertex

$x$

$y$

$z$
Today we deal with triangulations

- Genus 0 triangle mesh
- (Random) planar triangulation
- 2D Delaunay triangulation
- Triangulations with boundaries
- Higher genus triangulations
Surface triangle mesh endowed with a Schnyder wood

(planar) Triangle meshes

Delaunay triangulation of random points (endowed with a Schnyder wood)

Tutte drawing of a random planar triangulation

Schnyder drawing of a planar triangulation

$g$-Schnyder woods, for genus $g$ triangulations

(Castelli-Fusy-Lewiner, SoCG08)

Schnyder woods for triangulations with multiple boundaries of arbitrary size

(Castelli-Fusy-Lewiner, CCCG2010)
Schnyder woods: applications
(and related properties)

grid drawing

graph counting, random generation
(Poulalhon-Schaeffer, Icalp 03)

Untangling geometric graphs
Bose, Dujmovic, Hurtado, Langerman, Morin, Wood (DCG 2009)

$fix(G) \geq (\frac{n}{3})^{\frac{1}{2}}$

Graph encoding
(Chuang-Garg-He-Kao-Lu Icalp ’98)
(He-Kao-Lu ’99)
(Chiang et al. Soda’01)
(Barbay-Castelli Aleardi-He-Munro Isaac’07)
(Castelli Aleardi-Fusy-Lewiner SoCG08)
(Castelli Aleardi-Fusy-Lewiner CCCG’10)
(Yamanaka-Nakano ’08)

Greedy routing

Every planar triangulation admits a greedy drawing (Dhandapani, Soda08)

Conjectured by Papadimitriou and Ratajczak for 3-connected planar graphs
**Schnyder woods: the definition**

1. Edges are colored and oriented in such a way that each inner node has exactly one outgoing edge of each color.
2. Colors and orientations around each inner node must respect the local Schnyder condition.

**Theorem**

Every planar triangulation admits a Schnyder wood, which can be computed in linear time.

**Theorem**

The three set $T_0$, $T_1$, $T_2$ are spanning trees of (the inner nodes of) $T$: 

![Diagram of Schnyder woods with nodes and edges colored and oriented.](image)
minimal Schnyder woods: the definition (no ccw triangles)

\[ v_0 \ v_1 \ v_2 \text{ outer face} \]

(a) \hspace{1cm} (b) \hspace{1cm} (c)

i) edge are colored and oriented in such a way that each inner node has exactly one outgoing edge of each color

ii) colors and orientations around each inner node must respect the local Schnyder condition
Several kinds of encodings: plane trees (with $n$ nodes)

- Compression schemes
- Succinct representations
- (Explicit) data structures

**Compression schemes**

$$\|B_n\| = \frac{1}{n+1} \binom{2n}{n} \approx 2^{2n} n^{-\frac{3}{2}}$$

$$\log_2 \|B_n\| = 2n + O(\lg n)$$

**Succinct representations**

$2n + o(n)$ bits

$O(1)$ time navigation

**Word-RAM model**

Explicit (pointer based) implementation (array based)

we need $2n$ references, or $\Theta(n \lg n)$ bits
Several kinds of encodings: triangle meshes \( (n \text{ vertices}) \)

Optimal compression scheme

(Explicit) Geometric data structures

(Explicit) Geometric data structures

\( \beta n + O(1) \) references (pointers)

\( 2 \times n \times 6 \times \log n \)

\( \text{or} \)

\( n \times 1 \times \log n \)

Succinct representations

(Explicit) Geometric data structures

\( 13n \) references

\( 13n \log n \) bits

\( \Psi_n = \frac{2(4n+1)!}{(3n+2)!(n+1)!} \approx \frac{16}{27} \sqrt{\frac{3}{2\pi}} n^{-5/2} \left(\frac{256}{27}\right)^n \)

\( \frac{1}{n} \log_2 \Psi_n \approx \log_2 \left(\frac{256}{27}\right) \approx 3.2451 \) bits/vertex

\( 3.2451n + O\left(n \frac{\log \log n}{\log n}\right) = 3.2451n + o(n) \) bits

\( \frac{13n \log n}{\log n} \) bits

\( \frac{\log n}{\log n} \) bits

\( \log \log n \) bits

\( \log n \) bits
Popular (explicit) data structures for surface meshes

<table>
<thead>
<tr>
<th>Data Structure</th>
<th>size</th>
<th>navigation time</th>
<th>vertex access</th>
<th>dynamic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge-based data structures</td>
<td>$18n + n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>yes</td>
</tr>
<tr>
<td>Triangle based DS / Corner Table</td>
<td>$12n + n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>yes</td>
</tr>
<tr>
<td>Directed edge (Campagna et al. ’99)</td>
<td>$12n + n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>yes</td>
</tr>
<tr>
<td>2D Catalogs (Castelli Alcardi et al., ’06)</td>
<td>$7.67n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>yes</td>
</tr>
<tr>
<td>Star vertices (Kallmann et al. ’02)</td>
<td>$7n$</td>
<td>$O(d)$</td>
<td>$O(d)$</td>
<td>no</td>
</tr>
<tr>
<td>TRIPOD (Snoeyink, Speckmann, ’99)</td>
<td>$6n$</td>
<td>$O(1)$</td>
<td>$O(d)$</td>
<td>no</td>
</tr>
<tr>
<td>SOT (Gurung et al. 2010)</td>
<td>$6n$</td>
<td>$O(1)$</td>
<td>$O(d)$</td>
<td>no</td>
</tr>
<tr>
<td>Our Thm 2 (with no vertex permutation)</td>
<td>$5n$</td>
<td>$O(1)$</td>
<td>$O(d)$</td>
<td>no</td>
</tr>
<tr>
<td>Our Thm 3 (with vertex permutation)</td>
<td>$4n$</td>
<td>$O(1)$</td>
<td>$O(d)$</td>
<td>no</td>
</tr>
<tr>
<td>Our Cor 3 (with vertex permutation)</td>
<td>$6n$</td>
<td>$O(1)$</td>
<td>$O(d)$</td>
<td>no</td>
</tr>
</tbody>
</table>

**Half-edge**

- target(e)
- opposite(e)
- next(e)
- prev(e)
- source(e)

**Winged-edge**

- LeftFront(e)
- F_left
- RightFront(e)
- F_right
- Source(e)

**Triangle-based**

- LeftBack(e)
- RightBack(e)
Popular (explicit) data structures for surface meshes

<table>
<thead>
<tr>
<th>Data Structure</th>
<th>size</th>
<th>navigation time</th>
<th>vertex access</th>
<th>dynamic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edge-based data structures</td>
<td>$18n + n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>yes</td>
</tr>
<tr>
<td>Triangle based DS / Corner Table</td>
<td>$12n + n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>yes</td>
</tr>
<tr>
<td>Directed edge (Campagna et al. ’99)</td>
<td>$12n + n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>yes</td>
</tr>
<tr>
<td>2D Catalogs (Castelli Aleardi et al., ’06)</td>
<td>$7.67n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>yes</td>
</tr>
<tr>
<td>Star vertices (Kallmann et al. ’02)</td>
<td>$7n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>no</td>
</tr>
<tr>
<td>TRIPOD (Snoeyink, Speckmann, ’99)</td>
<td>$6n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>no</td>
</tr>
<tr>
<td>SOT (Gurung et al. 2010)</td>
<td>$6n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>no</td>
</tr>
<tr>
<td>SQUAD (Gurung et al. 2011)</td>
<td>$(4 + \varepsilon)n$</td>
<td>$O(1)$</td>
<td>$O(d)$</td>
<td>no</td>
</tr>
<tr>
<td>Our Thm 2 (with no vertex permutation)</td>
<td>$5n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>no</td>
</tr>
<tr>
<td>Our Thm 3 (with vertex permutation)</td>
<td>$4n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>no</td>
</tr>
<tr>
<td>Our Cor 3 (with vertex permutation)</td>
<td>$6n$</td>
<td>$O(1)$</td>
<td>$O(1)$</td>
<td>no</td>
</tr>
<tr>
<td>LR (Gurung et al. 2011)</td>
<td>$(2 + \varepsilon)n$</td>
<td>$O(1)$</td>
<td>$O(d)$</td>
<td>no</td>
</tr>
</tbody>
</table>

ε between 0.09 and 0.3

ε about 0.8 and 0.3

Half-edge

Winged-edge

Triangle-based

![Graph data structure diagram](image)
Popular mesh data structures: space requirements

**Half-edge**

```
class Point{
    float x;
    float y;
    float z;
}
```

geometry

```
class Halfedge{
    Halfedge next, opposite;
    Vertex source;
}
```

connectivity

```
class Vertex{
    Halfedge e;
    Point p;
}
```

size (number of references)

\[3 \times 2e + n = 18n + n\]

Winged-edge

Size (number of references)

\[(4 + 2) \times e + n = 18n + n\]
Popular mesh data structures: space requirements

Triangle-based

```java
class Triangle{
    Triangle t1, t2, t3;
    Vertex v1, v2, v3;
}
class Point{
    float x;
    float y;
    float z;
}
class Vertex{
    Triangle root;
    Point p;
}
```

Size (number of references)

\[(3 + 3) \times f + n = 6 \times 2n + n = 13n\]
Non compact vs. compact mesh data structures

perform face reordering
perform face reordering

SOT data structure (Gurung et al. 2010)

TRIPOD data structure (Snoeyink and Speckmann, 1999)
use Schnyder woods (store 3 edges per vertex)
Experimental comparison

Tested on 3D models and random planar triangulations

Winged edge vs. Our Compact DS

1.2 - 1.9 times slower

1.19 - 1.35 times slower

(timings are expressed in nanoseconds/vertex)

vertex degree (only topological navigation)

vertex normals (navigation + geometric computations)
First simple Compact DS (size $6n$) $e := (u, v)$

- $0 \leq v \leq n - 1$
- $0 \leq e \leq 3n$

array based implementation of

a variation of TRIPOD (J. Snoeyink and B. Speckmann, 1999)

+ reordering of edges according to the (original) vertex numbering

$n$ lines
3 columns
2 references per edge
First simple Compact DS (size 6n)

\[
e := (u, v) \quad 0 \leq v \leq n - 1
\]
\[
0 \leq e \leq 3n
\]

\[
\begin{align*}
\text{retrieve } (w, u) &:= \begin{cases} (e + 1) \mod 3 & \text{case 1} \\
(T[e] + 2) \mod 3 & \text{case 2} \\
T[T[e]] & \text{case 3}
\end{cases} \\

\end{align*}
\]

\[
\begin{align*}
u &:= \text{source}(e) = e/3 \\
(w, v) &:= T[2e] \\
(z, v) &:= T[2e + 1] \\
\text{color}(e) &:= e \mod 3
\end{align*}
\]

retrieving \((z, u)\) is similar
More compact DS (size $5n$): use minimal Schnyder woods
(less redundant and "slightly more difficult to implement")

only 1 reference for black edges

forbidden configuration
More compact DS (size 5\(n\)): use minimal Schnyder woods (less redundant and "slightly more difficult to implement")

**implementation:** many cases to distinguish

\[
\begin{align*}
\text{retrieve } & (w, u) := \begin{cases} 
T[T[5u] - 3] & \text{case 1a} \\
T[T[T[5u]]] & \text{case 1b} \\
\ldots & \ldots 
\end{cases}
\end{align*}
\]

case analysis for black edges

(similar case analysis for the other two colors)
Most compact DS (4n references)
(still more compact using vertex reordering)
use the DFUDS (Depth First Unary Degree Sequence) order on $T_0$

case analysis for red edges

only 1 reference for black and red edges
Compact DS for higher genus meshes (5n references)
combine previous ideas with the use of genus $g$ Schnyder woods
(Castelli Aleardi, Fusy, Lewiner SoCG’08)

toroidal triangulation endowed with a $g$-Schnyder wood
almost all vertices have outgoing degree 3

more cases to consider

$G$
$\mathcal{E}^s = \{(u, w), (v, w)\}$

$G^s$
$n + 4g$
vertices
$2n + 2g$
faces
all vertices have outgoing degree at most 3

$1a$

$1b$

$2a$

$2b$
Conclusion

- possible extension to more general surface meshes (planar quadrangulations and 3-connected planar maps)

- is it possible to further reduce the space requirements: from $4n$ to $3n$ or less?