Linear algebraic representation of geometric data — Hints for a 3D application standard

Alberto Paoluzzi

(joint work with *Antonio DiCarlo* and *Vadim Shapiro*)

Global 3D Tech Forum — Oct 11, 2012

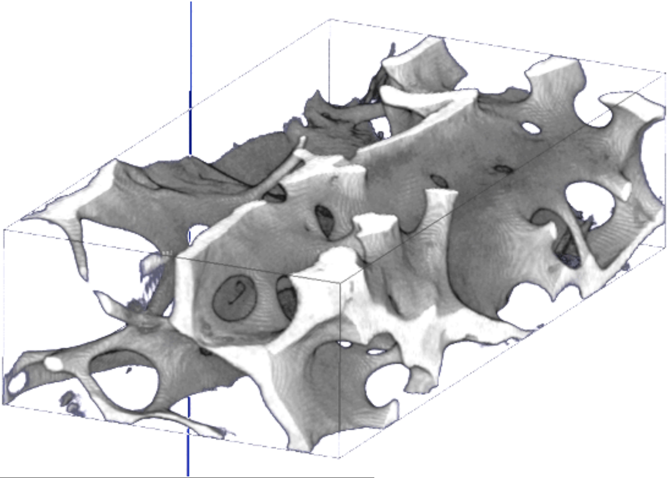
Disclaimer[1]

- partial ongoing work
- results to be proved
- but probably useful ideas
- of course, counterexamples are welcome

[1] Errors and/or misrepresentations are of the presenter, not of the other guys.

The challenge we face up

"it does matter to extract as much topological and metrical information as possible from raw 3D tomographic data"[2] ...

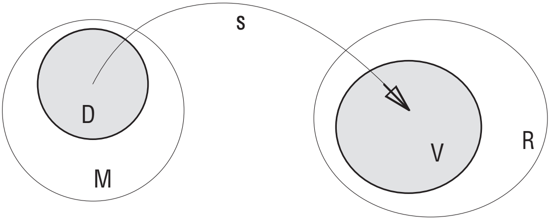


$\times\ 10^4$

[2] C. Bajaj, A. DiCarlo, G. Haiat, P. Laugier, F. Milicchio, S. Naili, A. Paoluzzi and G. Scorzelli, *Extracting trabecular geometry from tomographic images of spongy bone. 5th World Congress of Biomechanics.* Munich, 2006

The concept of *representation scheme*[3,4]

mapping $s : M \rightarrow R$ from a space of math models M to computer representations R



1. The set M contains the mathematical models of the class of solid objects that the scheme aims to represent.
2. The set R contains symbolic representations, i.e. suitable data structures, built according to some appropriate computer grammar.

[3] A. Requicha, *Representations for Rigid Solids: Theory, Methods, and Systems*, ACM Comput. Surv., 1980.

[4] V. Shapiro, *Solid Modeling*, In Handbook of Computer Aided Geometric Design, 2001

Representation schemes (a long list)

Most of such papers introduce or discuss one or more representation schemes ...

1. Requicha, ACM Comput. Surv., 1980 [Req80]
2. Requicha & Voelcker, PEP TM-25, 1977, [RV77]
3. Rossignac & Requicha, Comput. Aided Des., 1991, [RR91]
4. Bowyer, SVLIS, 1994, [Bow95]
5. Baumgart, Stan-CS-320, 1972, [Bau72]
6. Braid, Commun. ACM, 1975, [Bra75]
7. Dobkin & Laszlo, ACM SCG, 1987, [DL87]
8. Guibas & Stolfi, ACM Trans. Graph., 1985, [GS85]
9. Woo, IEEE Comp. Graph. & Appl., 1985, [Woo85]
10. Yamaguchi & Kimura, Comp. Graph. & Appl., 1995, [YK95]
11. Gursoz & Choi & Prinz, Geom.Mod., 1990, [WTP90]
12. S.S.Lee & K.Lee, ACM SMA, 2001, [LL01]
13. Rossignac & O'Connor, IFIP WG 5.2, 1988, [RO90]
14. Weiler, IEEE Comp. Graph. & Appl., 1985, [Wei85]
15. Silva, Rochester, PEP TM-36, 1981, [Sil81]
16. Shapiro, Cornell Ph.D Th., 1991, [Sha91]
17. Paoluzzi et al., ACM Trans. Graph., 1993, [PBCF93]
18. Pratt & Anderson, ICAP, 1994, [PA94]
19. Bowyer, Djinn, 1995, [BS95]
20. Gomes et al., ACM SMA, 1999, [GMR99]
21. Raghothama & Shapiro, ACM Trans. Graph., 1998, [RS98]
22. Shapiro & Vossler, ACM SMA, 1995, [SV95]
23. Hoffmann & Kim, Comput. Aided Des., 2001, [HK01]
24. Raghothama & Shapiro, ACM SMA, 1999, [RS99]
25. DiCarlo et al., IEEE TASE, 2008, [DMPS09]
26. Bajaj et al., CAD&A, 2006, [BPS06]
27. Pascucci et al., ACM SMA, 1995, [PFP95]
28. Paoluzzi et al., ACM Trans. Graph., 1995, [PPV95]
29. Paoluzzi et al., Comput. Aided Des., 1989, [PRS89]
30. Ala, IEEE Comput. Graph. Appl., 1992, [Ala92]

and much more ...

Chain-Complex (LAR) schemes[5]: M $\to$ R

the descriptive power of a scheme is measured from the size of its domain

Domain (Math models)

{ *space of finite CW-complexes* + *class of characteristic maps* }

Range (Computer representations)

{ *sparse matrix repr. of a chain complex of singular p-chains* }, $(0 \leq p \leq d)$

restricted here to:

D = { regular simplicial complexes + piecewise affine maps }

V = { CSR matrices of linear operators between groups of chains/cochains with coefficients either in $\mathbb{Z}_2 = \{0, 1\}$, or in $\{-1, 0, 1\}$, or in $\mathbb{R}$ }

[5] Linear scheme, because the scalars of the chain combinations are taken from a field ($\mathbb{Z}_2$ *is* a field).

Finite CW-complex I

Large class of space decompositions, including simplicial complexes, and much more

An *n-cell* is a space homeomorphic to the open n-disk $\mathrm{int}(D^n)$.
A *cell* is a space which is an n-cell for some $n \geq 0$.

Definition (Cell-decomposition)

A *cell-decomposition* of a space X is a family

$$\mathcal{E} = \{e_\alpha | \alpha \in I\}$$

of subspaces of X such that each e_α is a cell and $X = \bigsqcup_{\alpha \in I} e_\alpha$ (disjoint union of sets).

Definition (n-skeleton)

The *n-skeleton* of X is the subspace

$$X^n = \bigsqcup_{\alpha \in I:\ \dim(e_\alpha) \leq n} e_\alpha .$$

Finite CW-complex II

Large class of space decompositions, including simplicial complexes, and much more

A *finite cell-decomposition* is a cell decomposition consisting of finitely many cells.

Definition (Finite CW-complex)

The pair $(X, \mathcal{E})$, where X is a Hausdorff space and $\mathcal{E}$ is a finite cell-decomposition of X, is called a *finite CW-complex* if for each n-cell $e \in \mathcal{E}$ there is a *characteristic map* $\phi_e : D^n \to X$ restricting to a homeomorphism ${\phi_e}_{|\mathrm{int}(D^n)} : \mathrm{int}(D^n) \to e$ and taking S^{n-1} into X^{n-1}.

Singular p-chains

Let G be any commutative group G. Groups of interest are

- $G = \mathbb{Z}$, the group of integers,
- $G = \mathbb{R}$, the additive group of reals,
- $G = \mathbb{Z}_2 = \mathbb{Z}/2\mathbb{Z}$, the group of integers $\mod 2$.

Definition (Singular complex)

A *generalization of simplicial complex* can be defined on any manifold M or Hausdorff space X. For a singular complex, each singular simplex is a homeomorphism from a (simplicial) simplex in $\mathbb{R}^n$ to a subset of X.

Definition (Singular p-chain)

Singular p-chain on a space X, with coefficients in G, is a *finite formal sum*

$$c_p = g_1\sigma_p^1 + g_2\sigma_p^2 + \cdots + g_r\sigma_p^r$$

of singular simplexes $\sigma_p^s : \Delta_p \to X$, each with coefficient $g_s \in G$, where Δ_p is the standard (Euclidean) p-simplex in $\mathbb{R}^p$.

Chain and cochain complex[6]

applies to most domains characterized as cell complexes, without any restrictions on their type, dimension, codimension, orientability, manifoldness, and connectedness

$$\begin{tikzcd}
\cdots & C^3 \arrow[l, "\delta^3"'] & C^2 \arrow[l, "\delta^2"'] & C^1 \arrow[l, "\delta^1"'] & C^0 \arrow[l, "\delta^0"'] \\
\cdots \arrow[r, "\partial_4"] & C_3 \arrow[u, "\cong"'] \arrow[r, "\partial_3"] & C_2 \arrow[u, "\cong"'] \arrow[r, "\partial_2"] & C_1 \arrow[u, "\cong"'] \arrow[r, "\partial_1"] & C_0 \arrow[u, "\cong"']
\end{tikzcd}$$

Definition (Chain complex)

Chain complex is a sequence of Abelian groups $\ldots, C_3, C_2, C_1, C_0$ connected by homomorphisms (*boundary operators*) $\partial_n : C_n \to C_{n-1}$, such that for all n:

$$\partial_{n-1} \circ \partial_n = 0$$

Definition (Cochain complex)

Cochain complex is a sequence of Abelian groups $C^0, C^1, C^2, C^3, \ldots$ connected by homomorphisms (*coboundary operators*) $\delta^n : C^n \to C^{n+1}$, such that for all n:

$$\delta^{n+1} \circ \delta^n = 0$$

[6] A. DiCarlo, F. Milicchio, A. Paoluzzi, and V. Shapiro. *Chain-Based Representations for Solid and Physical Modeling*. 2009. IEEE Transactions on Automation Science and Engineering

Incidence relations[7] vs linear operators

Is a change of paradigm in shape representation

Use *symbols* V, E, F for K_0, K_1, K_2, the bases of linear spaces C_0, C_1, C_2 of chains in 2D.

Incidence/adjacency relations:

$$XY \subset X \times Y$$

	V	E	F
V	VV	VE	VF
E	EV	EE	EF
F	FV	FE	FF

Linear operators:

$$\mathcal{XY} : \mathcal{Y} \to \mathcal{X}$$

$$\begin{tikzcd}
C_0 \arrow[r, Rightarrow, "\mathcal{FV}"] & C_2 \arrow[d, "\mathcal{EF}"] \arrow[dl, "\mathcal{VF}"'] \\
C_0 \arrow[r, "\mathcal{EV}"'] & C_1
\end{tikzcd}$$

$$\mathcal{EF} = \mathcal{EV} \circ \mathcal{VF} = \mathcal{EV} \circ \mathcal{FV}^{\top}$$

[7] See: Woo, *A combinatorial analysis of boundary data structure schemata*, IEEE CG&A, 1985

LAR representation of incidence/adjacency operators[8]

operator composition $\equiv$ matrix product

$$\mathcal{VV} = \mathcal{VE} \circ \mathcal{EV} = \mathcal{EV}^\top \circ \mathcal{EV}$$
$$\mathcal{VE} = \mathcal{EV}^\top$$
$$\mathcal{VF} = \mathcal{FV}^\top$$

$$[\mathcal{VV}] = [\mathcal{EV}]^\top [\mathcal{EV}]$$
$$[\mathcal{VE}] = [\mathcal{EV}]^\top$$
$$[\mathcal{VF}] = [\mathcal{FV}]^\top$$

$$\mathcal{EV}$$
$$\mathcal{EE} = \mathcal{EV} \circ \mathcal{VE} = \mathcal{EV} \circ \mathcal{EV}^\top$$
$$\mathcal{EF} = \mathcal{EV} \circ \mathcal{VF} = \mathcal{EV} \circ \mathcal{FV}^\top$$

$$[\mathcal{EV}]$$
$$[\mathcal{EE}] = [\mathcal{EV}]\,[\mathcal{EV}]^\top$$
$$[\mathcal{EF}] = [\mathcal{EV}]\,[\mathcal{FV}]^\top$$

$$\mathcal{FV}$$
$$\mathcal{FE} = \mathcal{FV} \circ \mathcal{VE} = \mathcal{FV} \circ \mathcal{EV}^\top$$
$$\mathcal{FF} = \mathcal{FV} \circ \mathcal{VF} = \mathcal{FV} \circ \mathcal{FV}^\top$$

$$[\mathcal{FV}]$$
$$[\mathcal{FE}] = [\mathcal{FV}]\,[\mathcal{EV}]^\top$$
$$[\mathcal{FF}] = [\mathcal{FV}]\,[\mathcal{FV}]^\top$$

[8] Only 2-complex structures are given here, involving 3 entities $\mathcal{V}$, $\mathcal{E}$, $\mathcal{F}$ and 2 fundamental operators $\mathcal{FV}$ and $\mathcal{EV}$

Dimension-independent $\mathcal{C}_{p,q}$ generation

The actually needed operators depend on the application

$$\mathcal{C}_{p,q} : C_q \to C_p$$

$\mathcal{C}_{p,q}$ linear operator

C_p linear space of p-chains

C_q linear space of q-chains

$$\begin{array}{|c|ccccccc|}
\hline
[\mathcal{C}_{0,0}] & \Leftarrow & [\mathcal{C}_{1,0}]^\top & \Leftarrow & [\mathcal{C}_{q,0}]^\top & \Leftarrow & [\mathcal{C}_{d,0}]^\top & \\
\hline
\Uparrow & & & & & & & \\
[\mathcal{C}_{1,0}] & & & & \Downarrow & & & \\
\Uparrow & & & & & & & \\
[\mathcal{C}_{p,0}] & & \Longrightarrow & & [\mathcal{C}_{p,q}] & & & \\
\Uparrow & & & & & & & \\
[\mathcal{C}_{d,0}] & & & & & & & \\
\hline
\end{array}$$

Data: $[\mathcal{C}_{d,0}]$, (p, q)
Result: $[\mathcal{C}_{p,q}]$

if $p + q = 0$ **then** $k \leftarrow 1$;
else if $p * q = 0$ **then** $k \leftarrow p + q$;
else $k \leftarrow \min\,(p + q)$;

for $h \in [k, d)$ **do**
| $[\mathcal{C}_{k,0}] \leftarrow \mathit{Facets}([\mathcal{C}_{k+1,0}])$;
end

if $p + q = 0$ **then return** $[\mathcal{C}_{1,0}]^\top[\mathcal{C}_{1,0}]$;
else if $q = 0$ **then return** $[\mathcal{C}_{p,0}]$;
else if $p = 0$ **then return** $[\mathcal{C}_{q,0}]^\top$;
else return $[\mathcal{C}_{p,0}][\mathcal{C}_{q,0}]^\top$

worst case: $n - 1$ sparse matrix computations and 1 s.m. product

Howto compute the boundary operator

For chains over the commutative group $\mathbb{Z}_2 = \{0, 1\}$, i.e. *without cell orientation*

In the remainder we use the symbol $\mathbb{Z}_2$ as a function: $\mathbb{Z}_2 : \mathbb{Z} \to \{0, 1\}; \; k \mapsto (k \bmod 2)$

1. first step forward (call it $\widetilde{\partial}_p$):

$$[\,\widetilde{\partial}_p\,](i, j) = 1$$
$$\text{if} \quad [\mathcal{C}_{p-1,p}](i, j) = \max_j \, [\mathcal{C}_{p-1,p}](i, j)$$
$$\text{else} \quad [\,\widetilde{\partial}_p\,](i, j) = 0$$

2. compose it with the “Boolean transformation” $\mathbb{Z}_2$

Definition (Unoriented boundary operator)

Boundary operator “without orientation” is

$$\partial_p := \mathbb{Z}_2 \circ \widetilde{\partial}_p$$

Oriented boundary operator

For chains over the commutative group $\mathbb{Z}$ of (implicitly) oriented cells

Let every cell σ_p in a p-complex K embedded in $\mathbb{R}^p$ have the *orientation* defined by is canonical representation:

$$\sigma_p^k =< v^{k_0}, v^{k_1}, \dots, v^{k_p} > \qquad \text{with} \qquad k_0 < k_1 < \cdots, < k_p$$

Compute the absolute orientation χ of every p-simplex σ_p^k, by using the matrix V_p^k of *homogeneous coordinates* of its vertices:

$$\chi : K_p \to \{-1, 1\}; \quad \sigma_p^k \mapsto (\text{sign} \circ \det)(V_p^k)$$

Finally, consider the unoriented boundary matrix $[\partial_p] : \mathcal{C}_p \to \mathcal{C}_{p-1}$, and assign to each *unit term in position* h, k the value

$$a_{h,k} = \rho(\sigma_{p-1}^h, \sigma_p^k) := (-1)^\ell \cdot \chi(\sigma_p^k) \in \{-1, 1\}, \qquad \text{with} \qquad \ell = \sum_{m<k} a_{h,m}$$

where $\rho : K_{p-1} \times K_p \to \{-1, 1\}$ provides the relative orientation of σ_{p-1}^h w.r.t. σ_p^k.

2D simplicial complex (non manifold pointset)

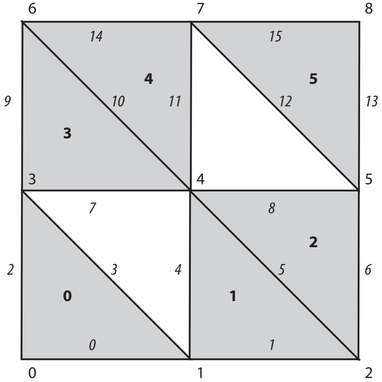
$K = K_0 \cup K_1 \cup K_2$ $\#K_0 =: k_0 = 9;$ $\#K_1 =: k_1 = 16;$ $\#K_2 =: k_2 = 6$

1-cells by 0-cells

```
EV = [[0,1],
      [1,2],
      [0,3],
      [1,3],
      [1,4],
      [2,4],
      [2,5],
      [3,4],
      [4,5],
      [3,6],
      [4,6],
      [4,7],
      [5,7],
      [5,8],
      [6,7],
      [7,8]]
```

2-cells by 0-cells

```
FV = [[0,1,3],
      [1,2,4],
      [2,4,5],
      [3,4,6],
      [4,6,7],
      [5,7,8]]
```



2D Example

Characteristic matrices

```
FV = [[0,1,3],
      [1,2,4],
      [2,4,5],
      [3,4,6],
      [4,6,7],
      [5,7,8]]
```

```
EV = [[0,1],
      [1,2],
      [0,3],
      [1,3],
      [1,4],
      [2,4],
      [2,5],
      [3,4],
      [4,5],
      [3,6],
      [4,6],
      [4,7],
      [5,7],
      [5,8],
      [6,7],
      [7,8]]
```

$$[\mathcal{EV}] = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

$$[\mathcal{FV}] = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

Sparse row-compressed matrices (CSR)[9]

operators $\mathcal{EV} : C_0 \to C_1$ and $\mathcal{FV} : C_0 \to C_2$ between spaces of p-chains ($0 \leq p \leq 2$)

coordinate representation w.r.t. the standard p-chain basis (the single p-cells)

$$[\mathcal{EV}] = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

$$[\mathcal{FV}] = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

sparse matrix representation

[[0,0,1],[0,1,1],[1,1,1],
[1,2,1],[2,0,1],[2,3,1],[3,1,1],
[3,3,1],[4,1,1],[4,4,1],[5,2,1],
[5,4,1],[6,2,1],[6,5,1],[7,3,1],
[7,4,1],[8,4,1],[8,5,1],[9,3,1],
[9,6,1],[10,4,1],[10,6,1],[11,4,1],
[11,7,1],[12,5,1],[12,7,1],[13,5,1],
[13,8,1],[14,6,1],[14,7,1],[15,7,1],
[15,8,1]]

[[0,0,1],[0,1,1],[0,3,1],
[1,1,1],[1,2,1],[1,4,1],[2,2,1],
[2,4,1],[2,5,1],[3,3,1],[3,4,1],
[3,6,1],[4,4,1],[4,6,1],[4,7,1],
[5,5,1],[5,7,1],[5,8,1]]

[9] The actual CSR representation is different.

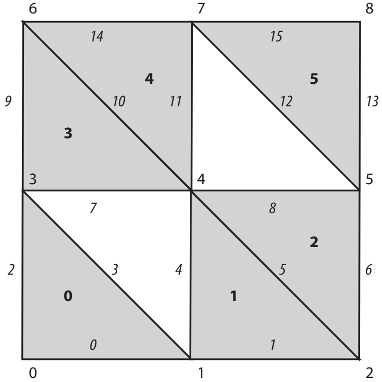
Incidence on vertices

$\mathcal{VV} : C_0 \to C_0$; $\mathcal{EV} : C_0 \to C_1$; $\mathcal{FV} : C_0 \to C_2$

```
VV =
[[0, 1, 3],
 [0, 1, 2, 3, 4],
 [1, 2, 4, 5],
 [0, 1, 3, 4, 6],
 [1, 2, 3, 4, 5, 6, 7],
 [2, 4, 5, 7, 8],
 [3, 4, 6, 7],
 [4, 5, 6, 7, 8],
 [5, 7, 8]]
```

```
EV =
[[0, 2],
 [0, 1, 3, 4],
 [1, 5, 6],
 [2, 3, 7, 9],
 [4, 5, 7, 8, 10, 11],
 [6, 8, 12, 13],
 [9, 10, 14],
 [11, 12, 14, 15],
 [13, 15]]
```

```
FV =
[[0],
 [0, 1],
 [1, 2],
 [0, 3],
 [1, 2, 3, 4],
 [2, 5],
 [3, 4],
 [4, 5],
 [5]]
```



Computation examples:

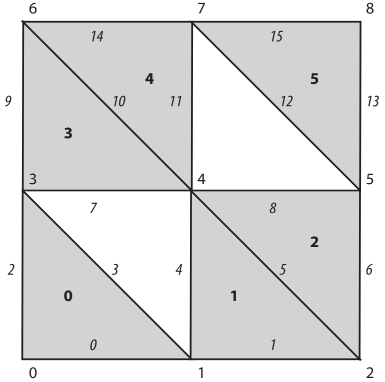
$$[c_0^k] = [0, \dots, 1, \dots, 0]^\top \qquad \text{(0-chain basis element)}$$

$$\mathcal{VV}(c_0^k) \equiv [\mathcal{VV}][c_0^k]; \qquad \mathcal{EV}(c_0^k) \equiv [\mathcal{EV}][c_0^k]; \qquad \mathcal{FV}(c_0^k) \equiv [\mathcal{FV}][c_0^k])$$

Incidence on edges

$\mathcal{VE} : C_1 \to C_0$; $\mathcal{EE} : C_1 \to C_1$; $\mathcal{FE} : C_1 \to C_2$

```
VE =        EE =                                FE =
[[0, 1],    [[0, 1, 2, 3, 4],                   [[0, 1],
 [1, 2],     [0, 1, 3, 4, 5, 6],                 [0, 1, 2],
 [0, 3],     [0, 2, 3, 7, 9],                    [0, 3],
 [1, 3],     [0, 1, 2, 3, 4, 7, 9],              [0, 1, 3],
 [1, 4],     [0, 1, 3, 4, 5, 7, 8, 10, 11],      [0, 1, 2, 3, 4],
 [2, 4],     [1, 4, 5, 6, 7, 8, 10, 11],         [1, 2, 3, 4],
 [2, 5],     [1, 5, 6, 8, 12, 13],               [1, 2, 5],
 [3, 4],     [2, 3, 4, 5, 7, 8, 9, 10, 11],      [0, 1, 2, 3, 4],
 [4, 5],     [4, 5, 6, 7, 8, 10, 11, 12, 13],    [1, 2, 3, 4, 5],
 [3, 6],     [2, 3, 7, 9, 10, 14],               [0, 3, 4],
 [4, 6],     [4, 5, 7, 8, 9, 10, 11, 14],        [1, 2, 3, 4],
 [4, 7],     [4, 5, 7, 8, 10, 11, 12, 14, 15],   [1, 2, 3, 4, 5],
 [5, 7],     [6, 8, 11, 12, 13, 14, 15],         [2, 4, 5],
 [5, 8],     [6, 8, 12, 13, 15],                 [2, 5],
 [6, 7],     [9, 10, 11, 12, 14, 15],            [3, 4, 5],
 [7, 8]]     [11, 12, 13, 14, 15]]               [4, 5]]
```



Computation examples:

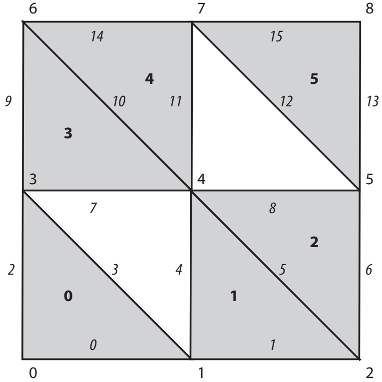
$$[c_1^k] = [0, \ldots, 1, \ldots, 0]^\top \qquad \text{(1-chain basis element)}$$

$$\mathcal{VE}(c_1^k) = [\mathcal{VE}][c_1^k]; \qquad \mathcal{EE}(c_1^k) = [\mathcal{EE}][c_1^k]; \qquad \mathcal{FE}(c_1^k) = [\mathcal{FE}][c_1^k]$$

Incidence on faces

$\mathcal{VF} : C_2 \to C_0$; $\mathcal{EF} : C_2 \to C_1$; $\mathcal{FF} : C_2 \to C_2$

```
VF =         EF =                                     FF =
[[0, 1, 3], [[0, 1, 2, 3, 4, 7, 9],                  [[0, 1, 3],
 [1, 2, 4],  [0, 1, 3, 4, 5, 6, 7, 8, 10, 11],         [0, 1, 2, 3, 4],
 [2, 4, 5],  [1, 4, 5, 6, 7, 8, 10, 11, 12, 13],       [1, 2, 3, 4, 5],
 [3, 4, 6],  [2, 3, 4, 5, 7, 8, 9, 10, 11, 14],        [0, 1, 2, 3, 4],
 [4, 6, 7],  [4, 5, 7, 8, 9, 10, 11, 12, 14, 15],      [1, 2, 3, 4, 5],
 [5, 7, 8]]  [6, 8, 11, 12, 13, 14, 15]]               [2, 4, 5]]
```



Computation examples:

$$[c_2^k] = [0, \dots, 1, \dots, 0]^\top \qquad \text{(2-chain basis element)}$$

$$\mathcal{VF}(c_2^k) = [\mathcal{VF}][c_2^k]; \qquad \mathcal{EF}(c_2^k) = [\mathcal{EF}][c_2^k]; \qquad \mathcal{FF}(c_2^k) = [\mathcal{FF}][c_2^k]$$

Boundary operators $\partial_n : C_n \to C_{n-1}$

Linear operators to compute the $(n-1)$-chains that are boundary of a n-chain

$$[\partial_n](i,j) = 1$$

$$\text{if} \quad [\mathcal{C}_{n-1}\mathcal{C}_n](i,j) = \max_j [\mathcal{C}_{n-1}\mathcal{C}_n](i,j)$$

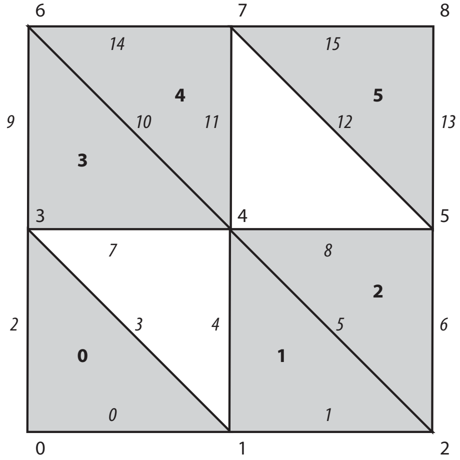
example of $[\partial_2]$

$[\mathcal{E}\mathcal{F}] =$

```
[[2 1 0 0 0 0]
 [1 2 1 0 0 0]
 [2 0 0 1 0 0]
 [2 1 0 1 0 0]
 [1 2 1 1 1 0]
 [0 2 2 1 1 0]
 [0 1 2 0 0 1]
 [1 1 1 2 1 0]
 [0 1 2 1 1 1]
 [1 0 0 2 1 0]
 [0 1 1 2 2 0]
 [0 1 1 1 2 1]
 [0 0 1 0 1 2]
 [0 0 1 0 0 2]
 [0 0 0 1 2 1]
 [0 0 0 0 1 2]]
```

$[\partial_2] =$

```
[[1 0 0 0 0 0]
 [0 1 0 0 0 0]
 [1 0 0 0 0 0]
 [1 0 0 0 0 0]
 [0 1 0 0 0 0]
 [0 1 1 0 0 0]
 [0 0 1 0 0 0]
 [0 0 0 1 0 0]
 [0 0 1 0 0 0]
 [0 0 0 1 0 0]
 [0 0 0 1 1 0]
 [0 0 0 0 1 0]
 [0 0 0 0 0 1]
 [0 0 0 0 0 1]
 [0 0 0 0 1 0]
 [0 0 0 0 0 1]]
```



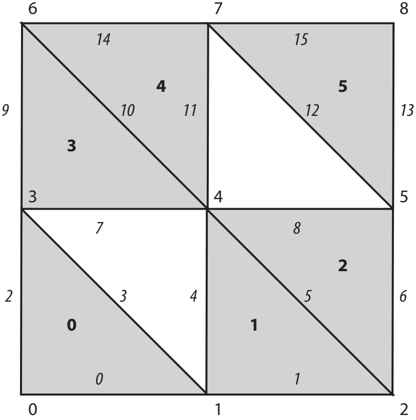
Computation of 1-boundaries

Some examples: take 2-chains $c_2, d_2, e_2 \in C_2$

$$
\begin{aligned}
[c_2] &= [1,1,1,1,1,1]^\top \\
[c_1] &= \mathbb{Z}_2\left([\partial_2][c_2]\right) \in C_1 \\
&= [1,1,1,1,1,0,1,1,1,1,0,1,1,1,1,1]^\top
\end{aligned}
$$

$$
\begin{aligned}
[d_2] &= [1,0,0,0,0,0]^\top \\
[d_1] &= \mathbb{Z}_2\left([\partial_2][d_2]\right) \in C_1 \\
&= [1,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0]^\top
\end{aligned}
$$

$$
\begin{aligned}
[e_2] &= [0,0,0,1,1,1]^\top \\
[e_1] &= \mathbb{Z}_2\left([\partial_2][e_2]\right) \in C_1 \\
&= [0,0,0,0,0,0,0,1,0,1,0,1,1,1,1,1]^\top
\end{aligned}
$$



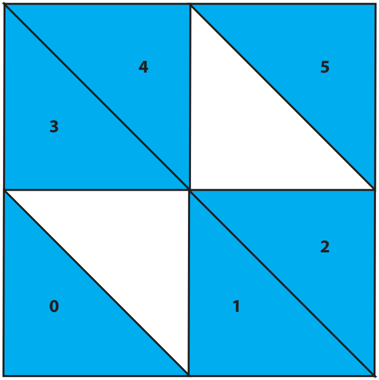
Computation of 1-boundaries

Some examples: take 2-chains $c_2, d_2, e_2 \in C_2$

$$
\begin{aligned}
[c_2] &= [1,1,1,1,1,1]^\top \\
[c_1] &= \mathbb{Z}_2\,([\partial_2][c_2]) \in C_1 \\
&= [1,1,1,1,1,0,1,1,1,1,0,1,1,1,1,1]^\top
\end{aligned}
$$

$$
\begin{aligned}
[d_2] &= [1,0,0,0,0,0]^\top \\
[d_1] &= \mathbb{Z}_2\,([\partial_2][d_2]) \in C_1 \\
&= [1,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0]^\top
\end{aligned}
$$

$$
\begin{aligned}
[e_2] &= [0,0,0,1,1,1]^\top \\
[e_1] &= \mathbb{Z}_2\,([\partial_2][e_2]) \in C_1 \\
&= [0,0,0,0,0,0,0,1,0,1,0,1,1,1,1,1]^\top
\end{aligned}
$$



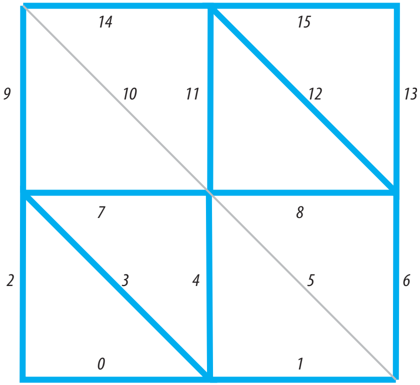
Computation of 1-boundaries

Some examples: take 2-chains $c_2, d_2, e_2 \in C_2$

$$
\begin{aligned}
[c_2] &= [1,1,1,1,1,1]^\top \\
[c_1] &= \mathbb{Z}_2\left([\partial_2][c_2]\right) \in C_1 \\
&= [1,1,1,1,1,0,1,1,1,1,0,1,1,1,1,1]^\top
\end{aligned}
$$

$$
\begin{aligned}
[d_2] &= [1,0,0,0,0,0]^\top \\
[d_1] &= \mathbb{Z}_2\left([\partial_2][d_2]\right) \in C_1 \\
&= [1,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0]^\top
\end{aligned}
$$

$$
\begin{aligned}
[e_2] &= [0,0,0,1,1,1]^\top \\
[e_1] &= \mathbb{Z}_2\left([\partial_2][e_2]\right) \in C_1 \\
&= [0,0,0,0,0,0,0,1,0,1,0,1,1,1,1,1]^\top
\end{aligned}
$$



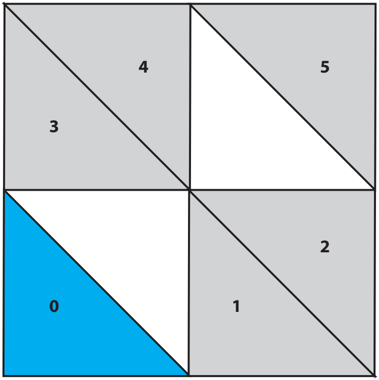
Computation of 1-boundaries

Some examples: take 2-chains $c_2, d_2, e_2 \in C_2$

$$
\begin{aligned}
[c_2] &= [1,1,1,1,1,1]^\top \\
[c_1] &= \mathbb{Z}_2\left([\partial_2][c_2]\right) \in C_1 \\
&= [1,1,1,1,1,0,1,1,1,1,0,1,1,1,1,1]^\top
\end{aligned}
$$

$$
\begin{aligned}
[d_2] &= [1,0,0,0,0,0]^\top \\
[d_1] &= \mathbb{Z}_2\left([\partial_2][d_2]\right) \in C_1 \\
&= [1,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0]^\top
\end{aligned}
$$

$$
\begin{aligned}
[e_2] &= [0,0,0,1,1,1]^\top \\
[e_1] &= \mathbb{Z}_2\left([\partial_2][e_2]\right) \in C_1 \\
&= [0,0,0,0,0,0,0,1,0,1,0,1,1,1,1,1]^\top
\end{aligned}
$$



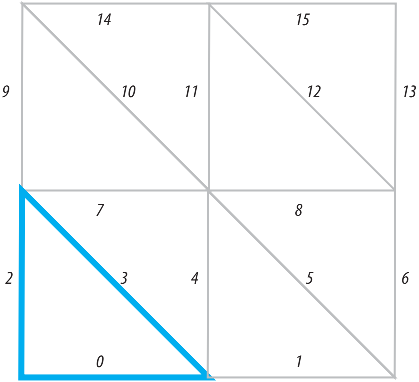
Computation of 1-boundaries

Some examples: take 2-chains $c_2, d_2, e_2 \in C_2$

$$
\begin{aligned}
[c_2] &= [1,1,1,1,1,1]^\top \\
[c_1] &= \mathbb{Z}_2\,([\partial_2][c_2]) \in C_1 \\
&= [1,1,1,1,1,0,1,1,1,1,0,1,1,1,1,1]^\top
\end{aligned}
$$

$$
\begin{aligned}
[d_2] &= [1,0,0,0,0,0]^\top \\
[d_1] &= \mathbb{Z}_2\,([\partial_2][d_2]) \in C_1 \\
&= [1,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0]^\top
\end{aligned}
$$

$$
\begin{aligned}
[e_2] &= [0,0,0,1,1,1]^\top \\
[e_1] &= \mathbb{Z}_2\,([\partial_2][e_2]) \in C_1 \\
&= [0,0,0,0,0,0,0,1,0,1,0,1,1,1,1,1]^\top
\end{aligned}
$$



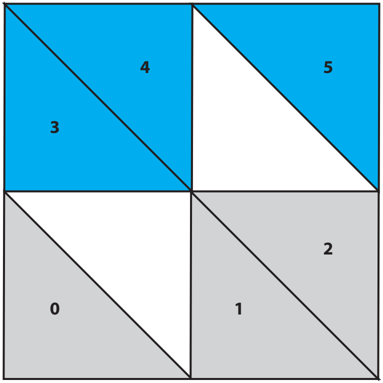
Computation of 1-boundaries

Some examples: take 2-chains $c_2, d_2, e_2 \in C_2$

$$
\begin{aligned}
[c_2] &= [1,1,1,1,1,1]^\top \\
[c_1] &= \mathbb{Z}_2([\partial_2][c_2]) \in C_1 \\
&= [1,1,1,1,1,0,1,1,1,1,0,1,1,1,1,1]^\top
\end{aligned}
$$

$$
\begin{aligned}
[d_2] &= [1,0,0,0,0,0]^\top \\
[d_1] &= \mathbb{Z}_2([\partial_2][d_2]) \in C_1 \\
&= [1,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0]^\top
\end{aligned}
$$

$$
\begin{aligned}
[e_2] &= [0,0,0,1,1,1]^\top \\
[e_1] &= \mathbb{Z}_2([\partial_2][e_2]) \in C_1 \\
&= [0,0,0,0,0,0,0,1,0,1,0,1,1,1,1,1]^\top
\end{aligned}
$$



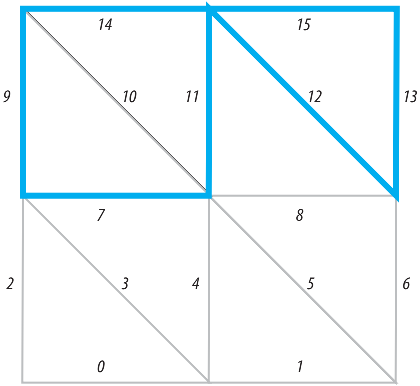
Computation of 1-boundaries

Some examples: take 2-chains $c_2, d_2, e_2 \in C_2$

$$
\begin{aligned}
[c_2] &= [1,1,1,1,1,1]^\top \\
[c_1] &= \mathbb{Z}_2\,([\partial_2][c_2]) \in C_1 \\
&= [1,1,1,1,1,0,1,1,1,1,0,1,1,1,1,1]^\top
\end{aligned}
$$

$$
\begin{aligned}
[d_2] &= [1,0,0,0,0,0]^\top \\
[d_1] &= \mathbb{Z}_2\,([\partial_2][d_2]) \in C_1 \\
&= [1,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0]^\top
\end{aligned}
$$

$$
\begin{aligned}
[e_2] &= [0,0,0,1,1,1]^\top \\
[e_1] &= \mathbb{Z}_2\,([\partial_2][e_2]) \in C_1 \\
&= [0,0,0,0,0,0,0,1,0,1,0,1,1,1,1,1]^\top
\end{aligned}
$$

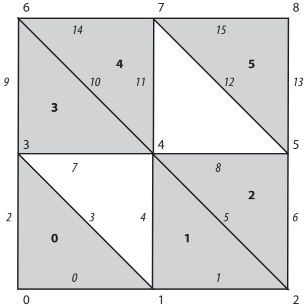


Boundary operators $\partial_n : C_n \to C_{n-1}$

Linear operators to compute the subset of $(n-1)$-cycles [10] that are boundary of a n-chain

$$[\partial_n](i,j) = 1$$

$$\text{if} \quad [\mathcal{C}_{n-1}\mathcal{C}_n](i,j) = \max_j [\mathcal{C}_{n-1}\mathcal{C}_n](i,j)$$



example of $[\partial_1]$

$[\mathcal{V}\mathcal{E}] =$
```
[[1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0]
 [1 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0]
 [0 1 0 0 0 1 1 0 0 0 0 0 0 0 0 0]
 [0 0 1 1 0 0 0 1 0 1 0 0 0 0 0 0]
 [0 0 0 0 1 1 0 1 1 0 1 1 0 0 0 0]
 [0 0 0 0 0 0 1 0 1 0 0 0 1 1 0 0]
 [0 0 0 0 0 0 0 0 0 1 1 0 0 0 1 0]
 [0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1]
 [0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1]]
```

$[\partial_1] =$
```
[[1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0]
 [1 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0]
 [0 1 0 0 0 1 1 0 0 0 0 0 0 0 0 0]
 [0 0 1 1 0 0 0 1 0 1 0 0 0 0 0 0]
 [0 0 0 0 1 1 0 1 1 0 1 1 0 0 0 0]
 [0 0 0 0 0 0 1 0 1 0 0 0 1 1 0 0]
 [0 0 0 0 0 0 0 0 0 1 1 0 0 0 1 0]
 [0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1]
 [0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1]]
```

[10] closed $(n-1)$-chains

$\partial\partial = 0$

Of course, the constraint equations of chain complexes are satisfied ...

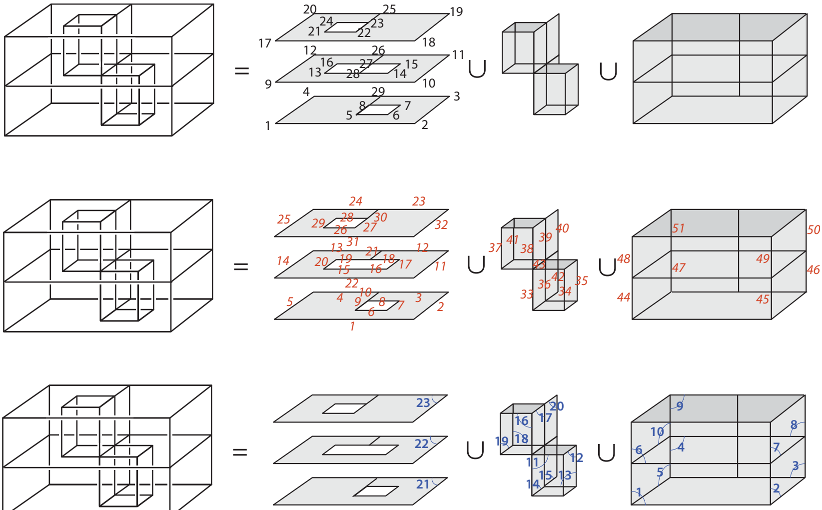
$$[\partial_1]\,[\partial_2]\begin{bmatrix} c & d & e \end{bmatrix} =$$

$$= \begin{bmatrix}
1&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
1&1&0&1&1&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&1&1&0&0&0&0&0&0&0&0&0\\
0&0&1&1&0&0&0&1&0&1&0&0&0&0&0&0\\
0&0&0&0&1&1&0&1&1&0&1&1&0&0&0&0\\
0&0&0&0&0&0&1&0&1&0&0&0&1&1&0&0\\
0&0&0&0&0&0&0&0&0&1&1&0&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&1&1&0&1&1\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&1
\end{bmatrix} \times \begin{bmatrix}
1&0&0&0&0&0\\
0&1&0&0&0&0\\
1&0&0&0&0&0\\
1&0&0&0&0&0\\
0&1&0&0&0&0\\
0&1&1&0&0&0\\
0&0&1&0&0&0\\
0&0&0&1&0&0\\
0&0&1&0&0&0\\
0&0&0&1&0&0\\
0&0&0&1&1&0\\
0&0&0&0&1&0\\
0&0&0&0&0&1\\
0&0&0&0&0&1\\
0&0&0&0&1&0\\
0&0&0&0&0&1
\end{bmatrix} \times \begin{bmatrix}
1&1&0\\
1&1&0\\
1&0&0\\
1&0&1\\
1&0&1\\
1&0&1
\end{bmatrix} = \begin{bmatrix}
2&2&0\\
4&2&0\\
4&0&0\\
4&2&2\\
8&0&4\\
4&0&2\\
4&0&4\\
4&0&4\\
2&0&2
\end{bmatrix}$$

$$\mathbb{Z}_2\left(\begin{bmatrix}
2&2&0\\
4&2&0\\
4&0&0\\
4&2&2\\
8&0&4\\
4&0&2\\
4&0&4\\
4&0&4\\
2&0&2
\end{bmatrix} \right) = \begin{bmatrix}
0&0&0\\
0&0&0\\
0&0&0\\
0&0&0\\
0&0&0\\
0&0&0\\
0&0&0\\
0&0&0\\
0&0&0
\end{bmatrix}$$

The “house-with-two-rooms” example

a contractible 2D CW-complex embedded in 3D.[11]



[11] Allen Hatcher, *Algebraic Topology*, 2002.

The “house-with-two-rooms” example

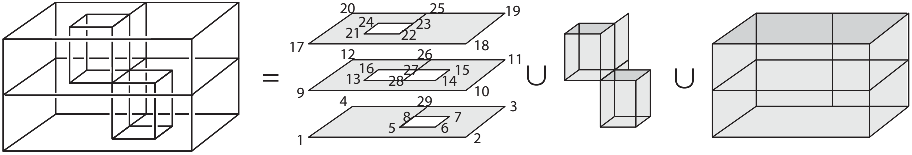
$K = K_0 \cup K_1 \cup K_2$ $\#K_0 =: k_0 = 29;$ $\#K_1 =: k_1 = 51;$ $\#K_2 =: k_2 = 23$

```
EV = [[1,2], [2,3], [3,29], [4,29], [1,4], [5,6], [6,7], [7,8], [5,8], [8,29], [10,11],
[11,26], [12,26], [9,12], [13,28], [28,14], [14,15], [15,27], [16,27], [13,16], [26,27],
[9,10], [19,25], [20,25], [17,20], [21,22], [22,23], [23,24], [21,24], [23,25], [17,18],
[18,19], [5,28], [6,14], [7,15], [8,27], [13,21], [22,28], [23,27], [25,26], [16,24],
[26,29], [27,28], [1,9], [2,10], [3,11], [4,12], [9,17], [10,18], [11,19], [12,20]]

FV = [[1,2,9,10], [2,3,10,11], [13,11,26,29], [4,12,26,29], [1,4,9,12], [9,10,17,18],
[10,11,18,19], [11,19,25,26], [12,20,25,26], [9,12,17,20], [5,6,14,28], [6,7,14,15],
[7,8,15,27], [5,8,27,28], [8,26,27,29], [13,21,22,28], [22,23,27,28], [16,23,24,27],
[13,16,21,24], [23,25,26,27], [1,2,3,4,5,6,7,8,29], [9,10,11,12,13,14,15,16,26,27,28],
[17,18,19,20,21,22,23,24,25]];
```

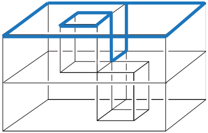
Geometric embedding:

```
V = [[3.0, 0.0, 0.0], [3.0, 4.0, 0.0], [0.0, 4.0, 0.0], [0.0, 0.0, 0.0], [2.0, 2.0, 0.0],
[2.0, 3.0, 0.0], [1.0, 3.0, 0.0], [1.0, 2.0, 0.0], [3.0, 0.0, 1.0], [3.0, 4.0, 1.0], [0.0,
4.0, 1.0], [0.0, 0.0, 1.0], [2.0, 1.0, 1.0], [2.0, 3.0, 1.0], [1.0, 3.0, 1.0], [1.0, 1.0,
1.0], [3.0, 0.0, 2.0], [3.0, 4.0, 2.0], [0.0, 4.0, 2.0], [0.0, 0.0, 2.0], [2.0, 1.0, 2.0],
[2.0, 2.0, 2.0], [1.0, 2.0, 2.0], [1.0, 1.0, 2.0], [0.0, 2.0, 2.0], [0.0, 2.0, 1.0], [1.0,
2.0, 1.0], [2.0, 2.0, 1.0], [0.0, 2.0, 0.0]]
```

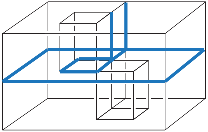


Boundary computation of several chains

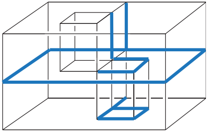
non-manifold and non-solid complex: looks like homotopy retraction ...



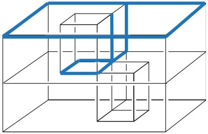
$\partial_2(\sum \sigma_2^{[20,23]})$



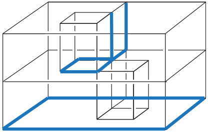
$\partial_2(\sum \sigma_2^{[6-10,16-20,23]})$



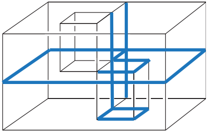
$\partial_2(\sum \sigma_2^{[1-10,16-23]})$



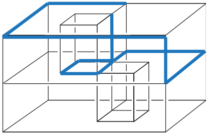
$\partial_2(\sum \sigma_2^{[16-20,23]})$



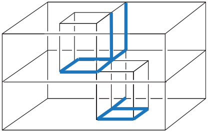
$\partial_2(\sum \sigma_2^{[1-10,16-20,23]})$



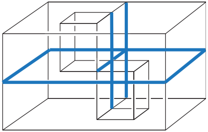
$\partial_2(\sum \sigma_2^{[1-10,15-23]})$



$\partial_2(\sum \sigma_2^{[7,8,16-20,23]})$



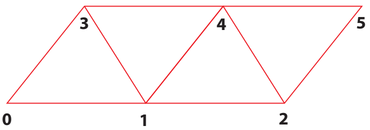
$\partial_2(\sum \sigma_2^{[1-10,16-21,23]})$



$\partial_2(\sum \sigma_2^{[1-23]})$

Facet extraction

$(p-1)$-facets of a p-simplex are given by a combinatorial formula, here in matrix form



each facet σ^h_{p-1} $(0 \leq h \leq p)$ of a simplex σ_p is computed by *dropping a vertex* off σ_p.

In matrix term, *every row of* M_p generates $p+1$ (tentative) rows of M_{p-1}.

Duplicates are eliminated by sorting the rows and reducing the double rows.

$$M_p = [\mathcal{FV}] = \begin{pmatrix} 0 & 1 & 3 \\ 1 & 2 & 4 \\ 1 & 3 & 4 \\ 2 & 4 & 5 \end{pmatrix} \to \begin{pmatrix} 1 & 3 \\ 0 & 3 \\ 0 & 1 \\ 2 & 4 \\ 1 & 4 \\ 1 & 2 \\ 3 & 4 \\ 1 & 4 \\ 1 & 3 \\ 4 & 5 \\ 2 & 5 \\ 2 & 4 \end{pmatrix} \to M_{p-1} = [\mathcal{EV}] = \begin{pmatrix} 0 & 1 \\ 0 & 3 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \\ 2 & 4 \\ 2 & 5 \\ 3 & 4 \\ 4 & 5 \end{pmatrix}$$

Chain extrusion from dimension $p \to p+1$

$\xi : C_p \to C_{p+1}$ *input*: CSR matrix M_p; *output*: CSR matrix M_{p+1}

add a column of ones to the M_p matrix ($p = 2$)

$$M_2 = \begin{pmatrix} 0 & 1 & 3 \\ 1 & 3 & 4 \\ 1 & 2 & 4 \\ 2 & 4 & 5 \end{pmatrix} \to \begin{pmatrix} 1 & 0 & 1 & 3 \\ 1 & 1 & 3 & 4 \\ 1 & 1 & 2 & 4 \\ 1 & 2 & 4 & 5 \end{pmatrix} = \widetilde{M}_2$$

duplicate the vertex set V (for a single extrusion step), by union with a translated set $\mathbf{T}(\mathbf{t})\, V$

$$\widehat{V} = V \cup \mathbf{T}(\mathbf{t})\, V,$$

a matrix $\widehat{M}_p := \widetilde{M}_p\, E_{r\times 2(p+1)}$ is generated where *rows contain* the indices of the $p+1$ vertices of a *p-cell and of its translated instance*

$$\widehat{M}_2 = \widetilde{M}_2\, E = \widetilde{M}_2 \begin{pmatrix} 0 & 0 & 0 & 6 & 6 & 6 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

each row $\widehat{M}_p^h$ of $\widehat{M}_p$, with $2(p+1)$ elements, *is mapped into* $p+1$ *rows* with $p+2$ elements, by product by a permutation Π^h and a projection P

$$M_3 = \xi(M_2) := \bigoplus_h \widetilde{M}_2^h\, \Pi^h\, P$$

where $\Pi^h = \Pi^{h-1}\,\Pi$ and $P : \mathbb{R}^{2(p+1)} \to \mathbb{R}^{p+2}$

Extrusion examples — structured mesh

Let call $\xi : C_p \to C_{p+1}$ the extrusion operator from p-chains to $(p+1)$-chains

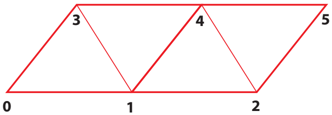
if we start from:

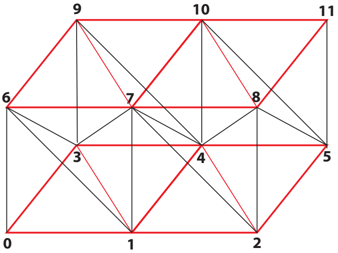
$$M_1 = \begin{pmatrix} 0 & 1 \\ 1 & 2 \end{pmatrix} =: [\mathcal{EV}]$$

we get

$$M_2 = \xi(M_1) = \begin{pmatrix} 0 & 1 & 3 \\ 1 & 2 & 4 \\ 1 & 3 & 4 \\ 2 & 4 & 5 \end{pmatrix} =: [\mathcal{FV}]$$

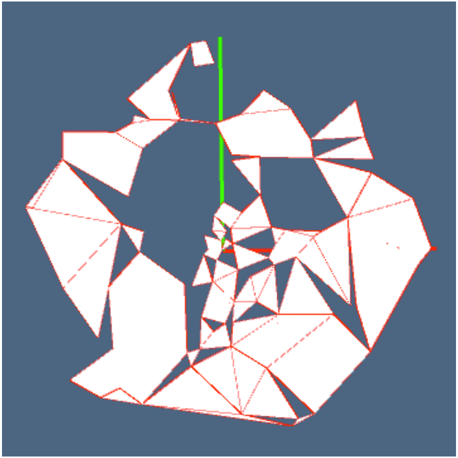
$$M_3 = \xi(M_2) = \begin{pmatrix} 0 & 1 & 3 & 6 \\ 1 & 2 & 4 & 7 \\ 1 & 3 & 4 & 7 \\ 1 & 3 & 6 & 7 \\ 2 & 4 & 5 & 8 \\ 2 & 4 & 7 & 8 \\ 3 & 4 & 7 & 9 \\ 3 & 6 & 7 & 9 \\ 4 & 5 & 8 & 10 \\ 4 & 7 & 8 & 10 \\ 4 & 7 & 9 & 10 \\ 5 & 8 & 10 & 11 \end{pmatrix} =: [\mathcal{CV}]$$

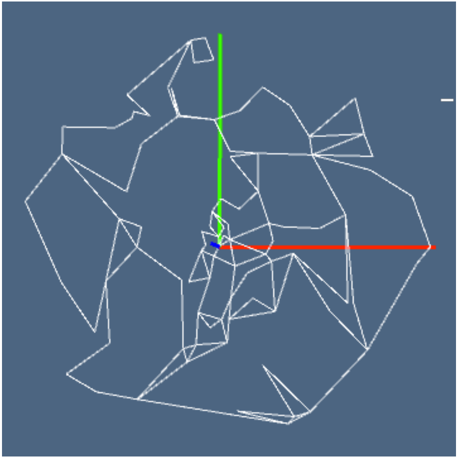




A triangulated polygon

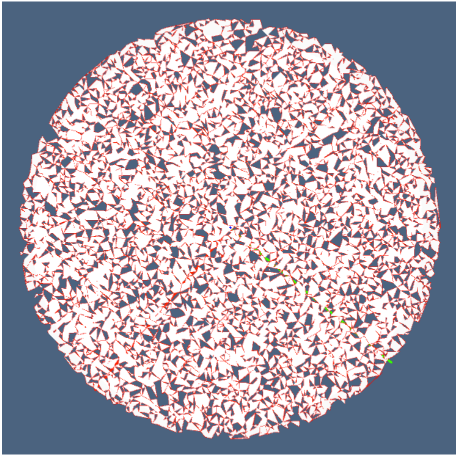
and its boundary



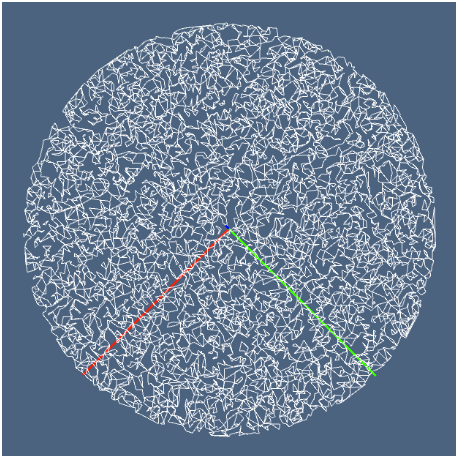


A random polygon and its boundary

Euler characteristic $\chi = k_0 - k_1 + k_2$ is pretty $\neq 1$ ...



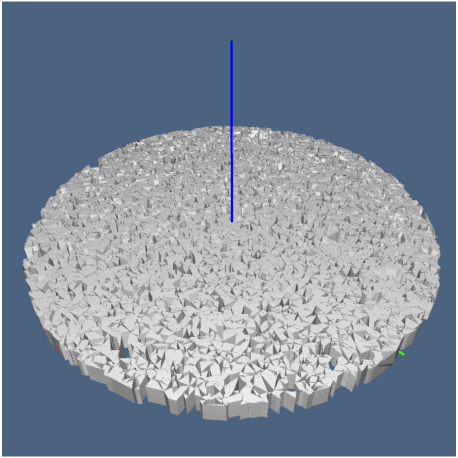
chain c_2



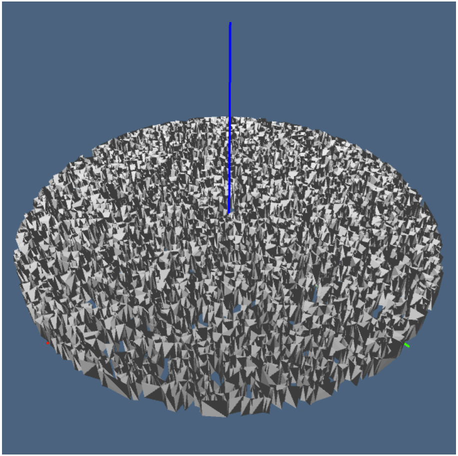
chain $c_1 = \partial_2(c_2)$

An extruded polygon and its extruded boundary

The *extrusion* operator $\xi : C_p \to C_{p+1}$ is dimension-independent



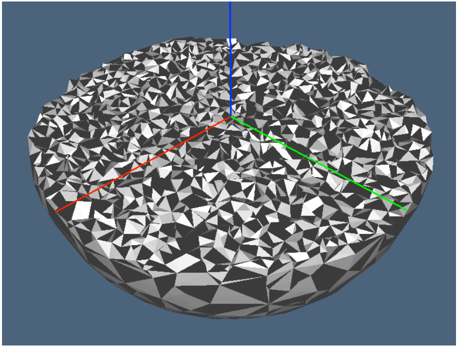
chain $c_3 = \xi(c_2)$



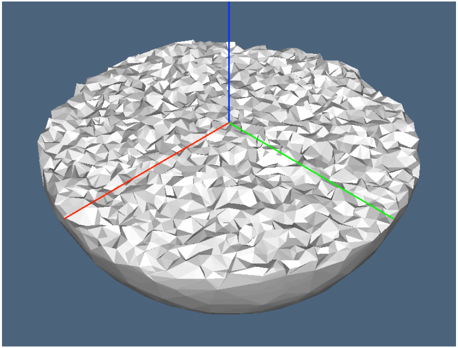
chain $c_2' = (\xi \circ \partial_2)(c_2)$

A random polyhedron and its *oriented* boundary

To recover the *orientation* of 2-cells (unit chains) from that of their coboundary is easy



chain c_3



chain $c_2 = \partial_3(c_3)$

Conclusion[12]

Properties of the LAR (Chain-Complex) representation scheme

- not only B-reps support, but **cellular decompositions**;
- general kind of — piecewise-linear — cells, **extensible to curved** ones;
- easy **dimension independence**;
- **no need of traversing** or searching over linked data structures;
- easy **parallelisation** (OpenGL, HTML5);
- **hardware support** (OpenCL, WebCL);

- closed math form, allowing for **further advances?** (we hope ...)

[12] We restricted here the domain of our *LAR* examples to simplicial complexes, but much wider domains can be represented with sparse binary matrices, and appropriate characteristic maps.

Thank you for attention !

Questions?

- on-going prototype implementation https://github.com/cvdlab/lar
- demo web application: https://github.com/cvdlab/lar-demo

References I

- Seshagiri Rao Ala, *Performance anomalies in boundary data structures*, IEEE Comput. Graph. Appl. **12** (1992), no. 2, 49–58.
- Bruce G. Baumgart, *Winged edge polyhedron representation.*, Tech. Report Stan-CS-320, Stanford, CA, USA, 1972.
- A. Bowyer, *Svlis set-theoretic kernel modeller: Introduction and user manual*, Information Geometers, 1995.
- C. Bajaj, A. Paoluzzi, and G. Scorzelli, *Progressive conversion from b-rep to bsp for streaming geometric modeling*, Computer-Aided Design and Applications **3** (2006), no. 5-6.
- I. C. Braid, *The synthesis of solids bounded by many faces*, Commun. ACM **18** (1975), no. 4, 209–216.
- A. Bowyer and Geometric Modelling Society, *Introducing Djinn: A geometric interface for solid modelling*, Information Geometers [for] the Geometric Modelling Society, 1995.

References II

- D. P. Dobkin and M. J. Laszlo, *Primitives for the manipulation of three-dimensional subdivisions*, Proceedings of the third annual symposium on Computational geometry (New York, NY, USA), SCG '87, ACM, 1987, pp. 86–99.
- A. DiCarlo, F. Milicchio, A. Paoluzzi, and V. Shapiro, *Chain-based representations for solid and physical modeling*, Automation Science and Engineering, IEEE Transactions on **6** (2009), no. 3, 454 –467.
- Abel Gomes, Alan Middleditch, and Chris Reade, *A mathematical model for boundary representations of n-dimensional geometric objects*, Proceedings of the fifth ACM symposium on Solid modeling and applications (New York, NY, USA), SMA '99, ACM, 1999, pp. 270–277.
- Leonidas Guibas and Jorge Stolfi, *Primitives for the manipulation of general subdivisions and the computation of voronoi*, ACM Trans. Graph. **4** (1985), no. 2, 74–123.
- Christoph M. Hoffmann and Ku-Jin Kim, *Towards valid parametric cad models*, Computer-Aided Design **33** (2001), no. 1, 81–90.

References III

- Sang Hun Lee and Kunwoo Lee, *Partial entity structure: a compact non-manifold boundary representation based on partial topological entities*, Proceedings of the sixth ACM symposium on Solid modeling and applications (New York, NY, USA), SMA '01, ACM, 2001, pp. 159–170.
- Michael J. Pratt and Bill D. Anderson, *A shape modelling api for the STEP standard*, in Fourteenth International Conference on Atomic Physics, 1994, pp. 1–7.
- A. Paoluzzi, F. Bernardini, C. Cattani, and V. Ferrucci, *Dimension-independent modeling with simplicial complexes*, ACM Trans. Graph. **12** (1993), no. 1, 56–102.
- Valerio Pascucci, Vincenzo Ferrucci, and Alberto Paoluzzi, *Dimension-independent convex-cell based HPC: representation scheme and implementation issues*, Proceedings of the third ACM Symposium on Solid Modeling and Applications (New York, NY, USA), SMA '95, ACM, 1995, pp. 163–174.
- Alberto Paoluzzi, Valerio Pascucci, and Michele Vicentino, *Geometric programming: a programming approach to geometric design*, ACM Trans. Graph. **14** (1995), no. 3, 266–306.

References IV

A. Paoluzzi, M. Ramella, and A Santarelli, *Boolean algebra over linear polyhedra*, Comput. Aided Des. **21** (1989), no. 10, 474–484.

Aristides G. Requicha, *Representations for rigid solids: Theory, methods, and systems*, ACM Comput. Surv. **12** (1980), no. 4, 437–464.

J. R. Rossignac and M. A. O'Connor, *SGC: a dimension-independent model for pointsets with internal structures and incomplete boundaries*, Geometric modeling for product engineering: selected and expanded papers from the IFIP WG 5.2/NSF Working Conference on Geometric Modeling, Rensselaerville, U.S.A., 18-22 September, 1988, North-Holland, 1990.

Jarek R. Rossignac and Aristides A. G. Requicha, *Constructive non-regularized geometry*, Comput. Aided Des. **23** (1991), no. 1, 21–32.

Srinivas Raghothama and Vadim Shapiro, *Boundary representation deformation in parametric solid modeling*, ACM Trans. Graph. **17** (1998), no. 4, 259–286.

References V

- ______, *Consistent updates in dual representation systems*, Proceedings of the fifth ACM symposium on Solid modeling and applications (New York, NY, USA), SMA '99, ACM, 1999, pp. 65–75.
- A. A. G. Requicha and H. B. Voelcker, *Constructive solid geometry*, Tech. Report TM-25, Production Automation Project, Univ. of Rochester, 1977.
- Vadim Shapiro, *Representations of semi-algebraic sets in finite algebras generated by space decompositions*, Ph.D. thesis, Ithaca, NY, USA, 1991, UMI Order No. GAX91-31407.
- C. E. Silva, *Alternative definitions of faces in boundary representations of solid objects*, Tech. Report TM-36, Production Automation Project, Univ. of Rochester, 1981.
- Vadim Shapiro and Donald L. Vossler, *What is a parametric family of solids?*, Proceedings of the third ACM symposium on Solid modeling and applications (New York, NY, USA), SMA '95, Acm, 1995, pp. 43–54.

References VI

- K. Weiler, *Edge-based data structures for solid modeling in curved-surface environments*, Computer Graphics and Applications, IEEE **5** (1985), no. 1, 21 –40.
- T. Woo, *A combinatorial analysis of boundary data structure schemata*, Computer Graphics & Applications, IEEE **5** (1985), no. 3, 19–27.
- M.J. Wozny, J.U. Turner, and K. Preiss, *Geometric modeling for product engineering: selected and expanded papers from the ifip wg 5.2/nsf working conference on geometric modeling, rensselaerville, u.s.a., 18-22 september, 1988*, North-Holland, 1990.
- Y. Yamaguchi and F. Kimura, *Nonmanifold topology based on coupling entities*, Computer Graphics and Applications, IEEE **15** (1995), no. 1, 42–50.