Preimage in 3-D

David Coeurjolly and Isabelle Sivignon

Laboratoire LIRIS
Université Claude Bernard Lyon 1
43 Bd du 11 Novembre 1918
69622 Villeurbanne CEDEX
France
david.coeurjolly@liris.cnrs.fr
isabelle.sivignon@liris.cnrs.fr

DGW - November 23 - 2005

David Coeurjolly and Isabelle Sivignon

Table of contents

Plan

David Coeurjolly and Isabelle Sivignon

Plan

David Coeurjolly and Isabelle Sivignon

Plan

David Coeurjolly and Isabelle Sivignon

Plan

Table of contents

David Coeurjolly and Isabelle Sivignon

3-D extension of the preimage definition

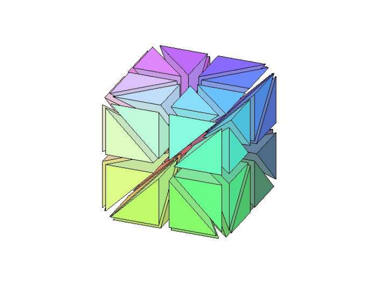
Euclidean plane parametrization

$$\Gamma: \quad ax + by + cz = b$$
$$z = \alpha x + \beta y + \gamma$$

$\Rightarrow (\alpha, \beta, \gamma)$-parameter space

First 48th of space

W.l.o.g. we consider Euclidean planes such that $0 \leq a \leq b \leq c \neq 0$ and thus $(\alpha, \beta, \gamma) \in [0, 1]^2 \times [0, 1[$



David Coeurjolly and Isabelle Sivignon

Digital plan definition

Digitization scheme

$$P(\alpha,\beta,\gamma) = \left\{(x,y,z) \in \mathbb{Z}^3 \mid \lfloor \alpha_0 x + \beta_0 y + \gamma_0 - z \rfloor = 0\right\}$$

DPS Preimage

Let S be a set of voxels, the DPS preimage $\bar{S}$ is given by:

$$\bar{S} = \{(\alpha,\beta,\gamma) \in [0,1]^2 \times [0,1[\mid \forall (x,y,z) \in S,\ 0 \leq \alpha x + \beta y + \gamma - z < 1\}$$

Property 1

$\bar{S}$ is a convex polyhedron in the (α,β,γ)-parameter space.

Digital plan definition

Digitization scheme

$$P(\alpha,\beta,\gamma) = \left\{(x,y,z) \in \mathbb{Z}^3 \mid \lfloor \alpha_0 x + \beta_0 y + \gamma_0 - z \rfloor = 0\right\}$$

DPS Preimage

Let S be a set of voxels, the DPS preimage $\bar{S}$ is given by:

$$\bar{S} = \{(\alpha,\beta,\gamma) \in [0,1]^2 \times [0,1[\mid \forall (x,y,z) \in S,\ 0 \leq \alpha x + \beta y + \gamma - z < 1\}$$

Property 1

$\bar{S}$ is a convex polyhedron in the (α,β,γ)-parameter space.

Digital plan definition

Digitization scheme

$$P(\alpha,\beta,\gamma) = \left\{(x,y,z) \in \mathbb{Z}^3 \mid \lfloor \alpha_0 x + \beta_0 y + \gamma_0 - z \rfloor = 0\right\}$$

DPS Preimage

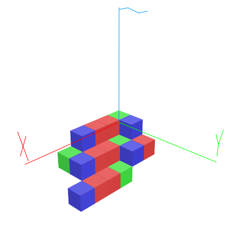
Let S be a set of voxels, the DPS preimage $\bar{S}$ is given by:

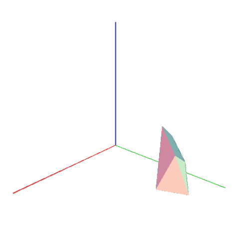
$$\bar{S} = \{(\alpha,\beta,\gamma) \in [0,1]^2 \times [0,1[\; | \forall (x,y,z) \in S,\; 0 \leq \alpha x + \beta y + \gamma - z < 1\}$$

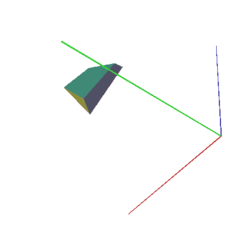
Property 1

$\bar{S}$ is a convex polyhedron in the (α,β,γ)-parameter space.

Examples...







David Coeurjolly and Isabelle Sivignon

Dual transform and strip decomposition

[VC00, CSD+05]

Linear dual transform in 3-D

$$z = \alpha x + \beta y + \gamma \quad \Leftrightarrow \quad (\alpha, \beta, \gamma)$$

Strip associated to a voxel

$$\mathcal{B}(x_0, y_0, z_0) = \begin{cases} 0 \leq \alpha x_0 + \beta y_0 + \gamma - z_0 & (D) \\ \alpha x_0 + \beta y_0 + \gamma - z_0 < 1 & (D') \end{cases}$$

Thus,

$$\bar{S} = \bigcap_{(x_i, y_i, z_i) \in S} \mathcal{B}(x_i, y_i, z_i)$$

Dual transform and strip decomposition

[VC00, CSD+05]

Linear dual transform in 3-D

$$z = \alpha x + \beta y + \gamma \quad \Leftrightarrow \quad (\alpha, \beta, \gamma)$$

Strip associated to a voxel

$$\mathcal{B}(x_0, y_0, z_0) = \begin{cases} 0 \leq \alpha x_0 + \beta y_0 + \gamma - z_0 & (D) \\ \alpha x_0 + \beta y_0 + \gamma - z_0 < 1 & (D') \end{cases}$$

Thus,

$$\bar{S} = \bigcap_{(x_i, y_i, z_i) \in S} \mathcal{B}(x_i, y_i, z_i)$$

Table of contents

Table of contents

DPS recognition based on the preimage computation

Definition

Given a set of voxels S, S is a piece of digital plane iff the preimage of S is not empty

First computational analysis

- 1 voxel $\longrightarrow$ 2 linear constraints in dimension 3
- n voxels $\longrightarrow$ $2.n$ linear constraints in dimension 3

Intersection of $2.n$ linear constraints

- $O(n)$ to detect if $\bar{S}$ is empty or not [Meg84]
- $O(n \log n)$ to construct the vertices/faces of the preimage [PS85] *(no incremental algorithms exist)*

DPS recognition based on the preimage computation

Definition

Given a set of voxels S, S is a piece of digital plane iff the preimage of S is not empty

First computational analysis

- 1 voxel $\rightarrow$ 2 linear constraints in dimension 3
- n voxels $\rightarrow$ 2.n linear constraints in dimension 3

Intersection of 2.n linear constraints

- $O(n)$ to detect if $\bar{S}$ is empty or not [Meg84]
- $O(n \log n)$ to construct the vertices/faces of the preimage [PS85] *(no incremental algorithms exist)*

David Coeurjolly and Isabelle Sivignon

Is there an arithmetical structure ?

Yes...

As in 2-D, we have:

- rays R(x,y,z)
- intersection of rays are fractions in Farey series
- ...

...but...

- complex ray intersections make the size of the preimage non constant
- Since there is no canonical order of vowels, the arithmetical structure cannot control the intersection between new rays and the current preimage

Is there an arithmetical structure ?

Yes...

As in 2-D, we have:

- rays R(x,y,z)
- intersection of rays are fractions in Farey series
- ...

...but...

- complex ray intersections make the size of the preimage not constant
- Since there is no canonical order of voxels, the arithmetical structure cannot control the intersection between new rays and the current preimage

Is there an arithmetical structure ?

Yes...

As in 2-D, we have:

- rays R(x,y,z)
- intersection of rays are fractions in Farey series
- ...

...but...

- complex ray intersections make the size of the preimage not constant
- Since there is no canonical order of voxels, the arithmetical structure cannot control the intersection between new rays and the current preimage

David Coeurjolly and Isabelle Sivignon

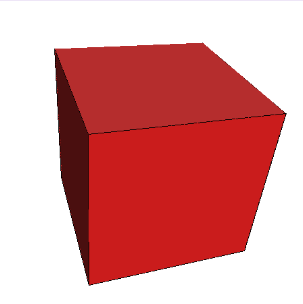
Straightforward incremental DPS recognition

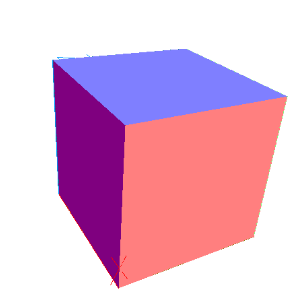
Preimage edge based DPS recognition

Let $\bar{S}$ be the polyhedron defined by the vertices $\{(0,0,0),(0,0,1),(0,1,1),(1,0,0)(1,0,1),(1,1,0),(1,1,1)\}$
for **each** voxel v in S **do**
Let C_1 and C_2 be the two constraints defining the strip $\mathcal{B}(v)$
for each constraint C_i in $\{C_1, C_2\}$ **do**
Let $\bar{S}'$ be an empty list of edges
Let *Coplanar* be an empty list of vertices
for each edge $e_i = [v_i, v_{i+1}]$ of $\bar{S}$ **do**
if C_i cross e_i at v' **then**
Append either the edge $[v_i, v']$ or $[v', v_{i+1}]$ to $\bar{S}'$ according the orientation of C_i
insert v' in the *coplanar* list
else
if both v_i and v_{i+1} belong to C_i **then**
Append the edge e_i to $\bar{S}'$
end if
end if
end for
Append to $\bar{S}'$ edges lying on C_i with vertices in the *Coplanar* list using for example a 2-D convex hull of v' points
if $\bar{S}'$ is empty **then**
"S is not a DSS"
else
$\bar{S} \leftarrow \bar{S}'$
end if
end for
end for
"S is a DSS with preimage $\bar{S}$"

David Coeur-jolly and Isabelle Sivignon

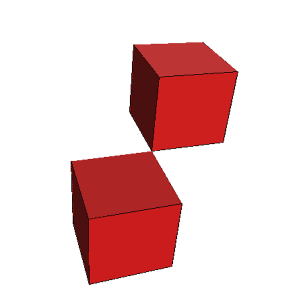
Recognition example

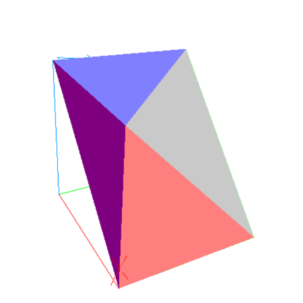




- $(0, 0, 0)$
- $(1, 1, 1)$
- $(2, 1, 1)$
- $(2, 2, 1)$
- $(3, 2, 1)$

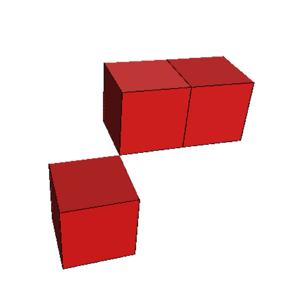
Recognition example

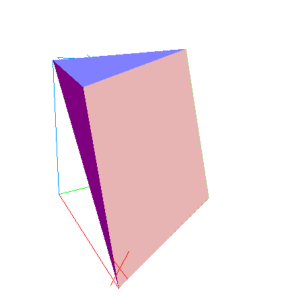




- $(0, 0, 0)$
- $(1, 1, 1)$
- $(2, 1, 1)$
- $(2, 2, 1)$
- $(3, 2, 1)$

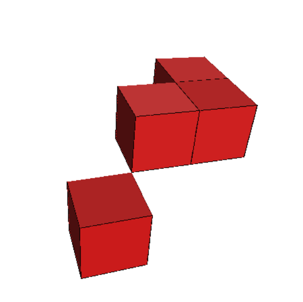
Recognition example

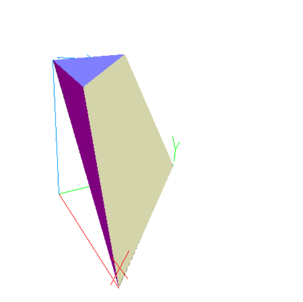




- $(0, 0, 0)$
- $(1, 1, 1)$
- $(2, 1, 1)$
- $(2, 2, 1)$
- $(3, 2, 1)$

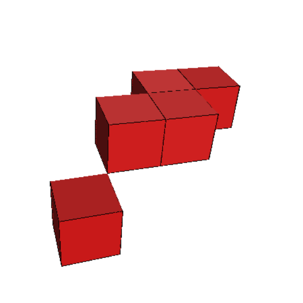
Recognition example

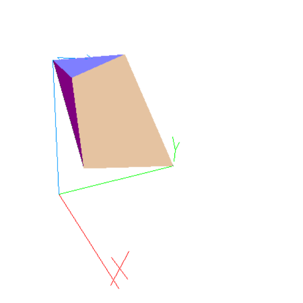




- $(0,0,0)$
- $(1,1,1)$
- $(2,1,1)$
- $(2,2,1)$
- $(3,2,1)$

Recognition example





- $(0,0,0)$
- $(1,1,1)$
- $(2,1,1)$
- $(2,2,1)$
- $(3,2,1)$

Table of contents

What is the computational cost of the recognition algorithm

Sketch of the algorithm

- For each voxel
  - For each edge of the preimage
    - update the preimage according to the two rays associated to the voxel
  - Reconnect the coplanar list of vertices with new edges

$\Rightarrow$ if we have n voxels, the computational cost is $O(n \cdot E)$ where E is the maximum number of edges of the preimage
note: the presented alg. runs in $O(n \cdot (E + E \log E))$

We need a bound on E !

Number of edges of the preimage

Observations

1. E depends on n
2. E depends on the *arithmetical size* of the DP normal vector components

David Coeurjolly and Isabelle Sivignon

Table of contents

Naive plane, leaning points and leaning polygons

[Rev91]

Naive plane

$N(a,b,c,\mu)$ with $a,b,c,\mu, \in \mathbb{Z}$ and $gcd(a,b,c)=1$

$$N = \left\{(x,y,z) \in \mathbb{Z}^3 \mid \mu \leq ax+by+cz < \mu + \max(|a|,|b|,|c|)\right\}$$

$(a,b,c)^T$ is the normal vector and μ is the intercept

Every rational digital plane $P(\alpha,\beta,\gamma)$ is a naive plane $N(a,b,c,\mu)$, and vice versa

W.l.o.g. we suppose $0 \leq a \leq b < c$, hence $\max(|a|,|b|,|c|) = c$

Naive plane, leaning points and leaning polygons

[Rev91]

Naive plane

$N(a,b,c,\mu)$ with $a,b,c,\mu,\in\mathbb{Z}$ and $gcd(a,b,c)=1$

$$N=\left\{(x,y,z)\in\mathbb{Z}^3 \mid \mu\leq ax+by+cz<\mu+\max(|a|,|b|,|c|)\right\}$$

$(a,b,c)^T$ is the normal vector and μ is the intercept

Every rational digital plane $P(\alpha,\beta,\gamma)$ is a naive plane $N(a,b,c,\mu)$, and vice versa

W.l.o.g. we suppose $0\leq a\leq b<c$, hence $\max(|a|,|b|,|c|)=c$

Naive plane, leaning points and leaning polygons

[Rev91]

Naive plane

$N(a,b,c,\mu)$ with $a,b,c,\mu,\in\mathbb{Z}$ and $gcd(a,b,c)=1$

$$N=\left\{(x,y,z)\in\mathbb{Z}^3 \mid \mu\leq ax+by+cz<\mu+\max(|a|,|b|,|c|)\right\}$$

$(a,b,c)^T$ is the normal vector and μ is the intercept

Every rational digital plane $P(\alpha,\beta,\gamma)$ is a naive plane $N(a,b,c,\mu)$, and vice versa

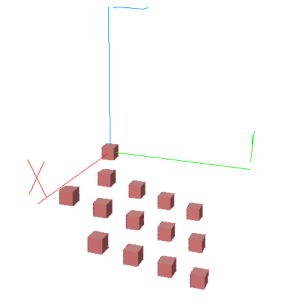
W.l.o.g. we suppose $0\leq a\leq b<c$, hence $\max(|a|,|b|,|c|)=c$

Naive plane, leaning points and leaning polygons

Net decomposition of Naive planes

$$N(a,b,c,\mu) : \bigcup \begin{cases} ax+by+cz=\mu \\ ax+by+cz=\mu+1 \\ \dots \\ ax+by+cz=\mu+c-1 \end{cases}$$

Example of the net $x+4y+5z=0$ (bi-periodic structure)



Naive plane, leaning points and leaning polygons

Net decomposition of Naive planes

$$N(a,b,c,\mu) : \bigcup \begin{cases} ax+by+cz = \mu \\ ax+by+cz = \mu+1 \\ \dots \\ ax+by+cz = \mu+c-1 \end{cases}$$

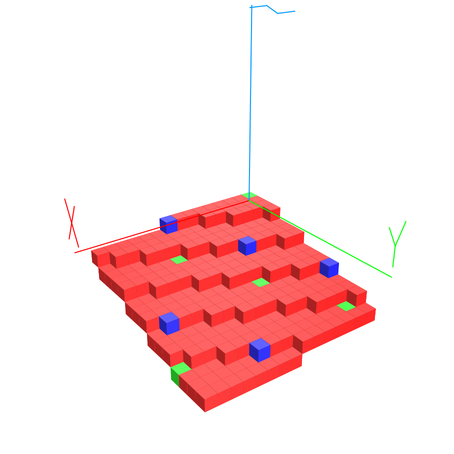
Upper and lower leaning points

Let $p=(x,y,z)$ be a point in $N(a,b,c,\mu)$, p is:

- an upper leaning point if $ax+by+cz=\mu$
- a lower leaning point if $ax+by+cz=\mu+c-1$

Example

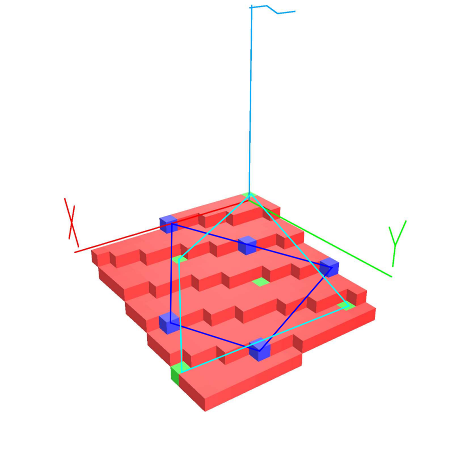
$N(7, 17, 57, 0)$



Uppper/lower leaning polygons for DPS

Definition

Upper (resp. lower) leaning polygon = convex hull of upper (resp. lower) leaning points



Preimage and leaning points

Properties [CSD+05]

Let $S \subset N(a, b, c, \mu)$ be a DPS. Then, the preimage of S, denoted $\bar{S}$, has the following properties:

- The points $v_\mathrm{l} = (\frac{a}{c}, \frac{b}{c}, \frac{\mu}{c})$ and $v_\mathrm{u} = (\frac{a}{c}, \frac{b}{c}, \frac{\mu+1}{c})$ are vertices of $\mathcal{P}(S)$. They correspond to the lower and the upper leaning planes $ax + by + cz = \mu$ and $ax + by + cz = \mu + c - 1$ in the Cartesian space;
- The preimage faces adjacent to v_l (resp. v_u) result from the vertices of the 2D convex hull of lower (resp. upper) leaning points in S.

Preimage and leaning points

Properties [CSD+05]

Let $S \subset N(a,b,c,\mu)$ be a DPS. Then, the preimage of S, denoted $\bar{S}$, has the following properties:

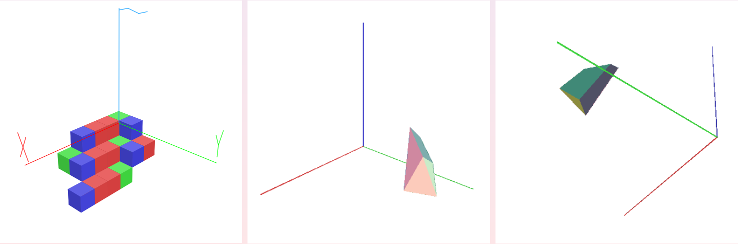
- The points $v_{\mathrm{l}} = (\frac{a}{c}, \frac{b}{c}, \frac{\mu}{c})$ and $v_{\mathrm{u}} = (\frac{a}{c}, \frac{b}{c}, \frac{\mu+1}{c})$ are vertices of $\mathcal{P}(S)$. They correspond to the lower and the upper leaning planes $ax+by+cz=\mu$ and $ax+by+cz=\mu+c-1$ in the Cartesian space;
- The preimage faces adjacent to v_l (resp. v_u) result from the vertices of the 2D convex hull of lower (resp. upper) leaning points in S.

Preimage and leaning points

Properties [CSD+05]

Let $S \subset N(a, b, c, \mu)$ be a DPS. Then, the preimage of S, denoted $\bar{S}$, has the following properties:

- The points $v_{\mathrm{l}} = (\frac{a}{c}, \frac{b}{c}, \frac{\mu}{c})$ and $v_{\mathrm{u}} = (\frac{a}{c}, \frac{b}{c}, \frac{\mu+1}{c})$ are vertices of $\mathcal{P}(S)$. They correspond to the lower and the upper leaning planes $ax + by + cz = \mu$ and $ax + by + cz = \mu + c - 1$ in the Cartesian space;
- The preimage faces adjacent to v_l (resp. v_u) result from the vertices of the 2D convex hull of lower (resp. upper) leaning points in S.



Main results

Theorem 1 [CSD+05]

Let $S \subset N(a,b,c,\mu)$ (with $0 \leq a \leq b < c$) be a piece of naive plane where each point (x_i, y_i, z_i) is such that (x_i, y_i) lies inside the projections onto the plane $z = 0$ of the two leaning polygons. Then S facets are only given by dual transform of leaning polygon points.

Sketch of proof

Based on properties of the linear dual transform and the fact that leaning polygons are convex hull of leaning points.

Corollary

E is bounded by the number of upper and lower leaning polygon points

Main results

Theorem 1 [CSD+05]

Let $S \subset N(a,b,c,\mu)$ (with $0 \leq a \leq b < c$) be a piece of naive plane where each point (x_i, y_i, z_i) is such that (x_i, y_i) lies inside the projections onto the plane $z = 0$ of the two leaning polygons. Then $\underline{S}$ facets are only given by dual transform of leaning polygon points.

Sketch of proof

Based on properties of the linear dual transform and the fact that leaning polygons are convex hull of leaning points.

Corollary

E is bounded by the number of upper and lower leaning polygon points

Main results

Theorem 1 [CSD+05]

Let $S \subset N(a, b, c, \mu)$ (with $0 \leq a \leq b < c$) be a piece of naive plane where each point (x_i, y_i, z_i) is such that (x_i, y_i) lies inside the projections onto the plane $z = 0$ of the two leaning polygons. Then S facets are only given by dual transform of leaning polygon points.

Sketch of proof

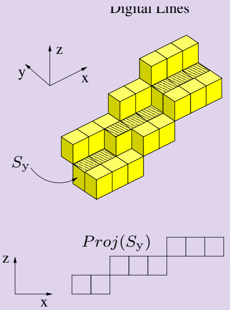
Based on properties of the linear dual transform and the fact that leaning polygons are convex hull of leaning points.

Corollary

E is bounded by the number of upper and lower leaning polygon points

Main results

Decomposition of a DPS into 2-D straight lines



Theorem 2 [CSD+05]

Main results

Decomposition of a DPS into 2-D straight lines

Theorem 2 [CSD+05]

Let $S \subset N(a,b,c,\mu)$ a piece of discrete naive plane such that $S = \bigcup_j S_{\mathrm{j}}$ with $S_{\mathrm{j}} = \{(x,y,z) \in S \mid y = j\}$. We assume that for all j, S_j is connected. If each S_{j} contains at least three leaning points (one lower leaning point, one upper leaning point and any third one), then $\bar{\mathrm{S}}$ facets are only given by dual transform of leaning polygon points.

Main results

Decomposition of a DPS into 2-D straight lines

Theorem 2 [CSD+05]

Sketch of proof

Based on properties of the 2-D straight line preimages and the fact that $\bar{S}$ can be computed using the 2-D preimages of the straight lines.

Conclusion of approach 1

Bound on E [CSD+05]

If S is a rectangular piece of DPS in a $[0,n] \times [0,n]$ window, then

$$E = O(\log(n))$$

Preimage structure

Facets of the preimage corresponds to upper and lower leaning points

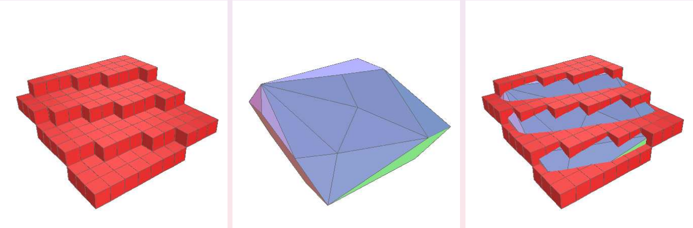
Table of contents

Main statement

Let S be a DPS

E is bounded by the size of the convex hull of S

This bound is not tight !

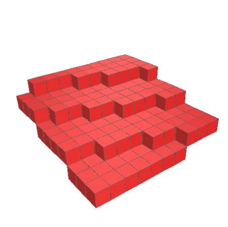
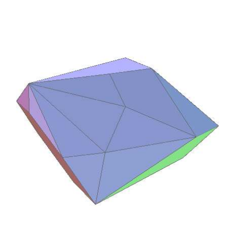
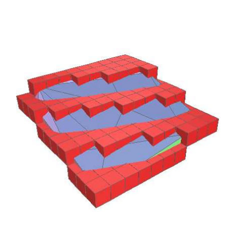


Main statement

Let S be a DPS

E is bounded by the size of the convex hull of S

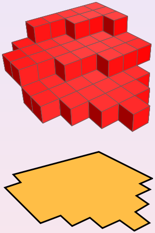
This bound is not tight !

Convex hull of a DPS

What is the maximum size of a DPS convex hull ?

$\Rightarrow$ Number of integer points in rational convex polyhedra



one-to-one and onto mapping of DPS and its basis

DPS as a feasible region of a rational inequality system

Number of linear constraints to define a DPS:

- 2 constraints to define the supporting (leaning) planes
- N constraints to encode the projection of the DPS onto to $xy-$axis plane

Convex hull of a DPS

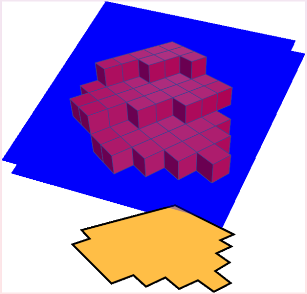
What is the maximum size of a DPS convex hull ?

$\Rightarrow$ Number of integer points in rational convex polyhedra

DPS as a feasible region of a rational inequality system

Number of linear constraints to define a DPS:

- 2 constraints to define the supporting (leaning) planes
- N constraints to encode the projection of the DPS onto to $xy-$axis plane



Convex hull of a DPS

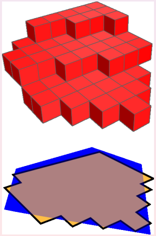
What is the maximum size of a DPS convex hull ?

$\Rightarrow$ Number of integer points in rational convex polyhedra

DPS as a feasible region of a rational inequality system

Number of linear constraints to define a DPS:

- 2 constraints to define the supporting (leaning) planes
- N constraints to encode the projection of the DPS onto to $xy-$axis plane



David Coeurjolly and Isabelle Sivignon

Integer Linear Programming

DPS as a feasible region of a rational inequality system

Number of linear constraint to define a DPS:

- 2 constraints to define the supporting (leaning) planes
- N constraints to encode the projection of the DPS onto to $xy-$axis plane

Knapsack polytope [Sch86]

Number of grid points in the convex hull of integer non-negative solutions to the inequality $\sum_{j=1}^{d} a_j x_j \leq a_0$ is $O(\log^d(\sigma))$ where $\sigma = \frac{4 \cdot a_0}{\min\{a_1, \ldots, a_d\}}$

Generalization

Main result

Let S be a DPS on a $[0, n]^3$ grid

$$|Cvx(S)| \leq N \log^2(n)$$

Corollary

If S is a rectangular piece of DPS (*i.e.* $N = 4$), E is bounded by $O(\log^2(n))$

Integer Linear Programming

DPS as a feasible region of a rational inequality system

Number of linear constraint to define a DPS:

- 2 constraints to define the supporting (leaning) planes
- N constraints to encode the projection of the DPS onto to $xy-$axis plane

Knapsack polytope [Sch86]

Number of grid points in the convex hull of integer non-negative solutions to the inequality $\sum_{j=1}^{d} a_j x_j \leq a_0$ is $O(\log^d(\sigma))$ where $\sigma = \frac{4 \cdot a_0}{\min\{a_1,\dots,a_d\}}$

▸ Generalization

Main result

Let S be a DPS on a $[0,n]^3$ grid

$$|Cvx(S)| \leq N \log^2(n)$$

Corollary

If S is a rectangular piece of DPS (*i.e.* $N=4$), E is bounded by $O(\log^2(n))$

[Knapsack theory, Integer linear programming, Number theory,...]

David Coeurjolly and Isabelle Sivignon

Table of contents

Reversible reconstruction in 3-D

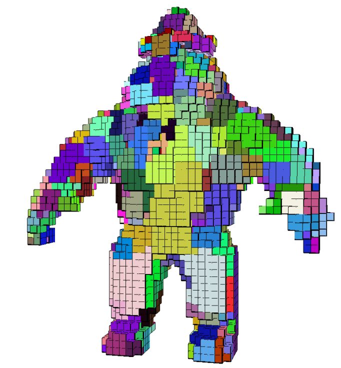
[SDC05]

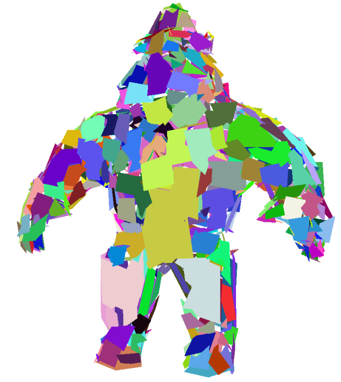
Statement of the problem

Convert a discrete surface into a polyhedron with the reversibility property

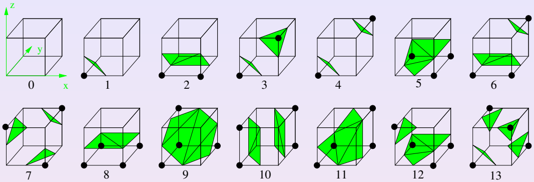
Difficult problem

We can segment the digital surface into maximal pieces of DP and extract facets from the DPS preimages but it is difficult to create the edges and vertices

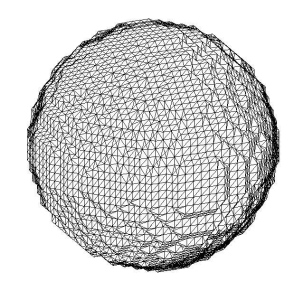




Marching-Cubes [LC87]



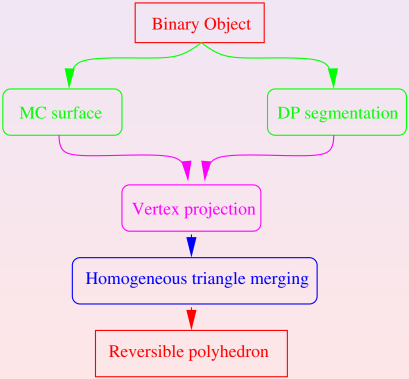
- Reversible polyhedron
- Combinatorial 2-manifold
- huge number of facets



Marching-Cubes Simplification using DP segmentation

Idea [CGS04]

Simplify the MC surface taking into account the DP segmentation and the DPS preimages

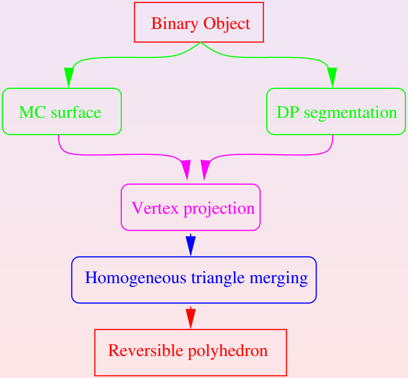


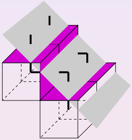
David Coeurjolly and Isabelle Sivignon

Marching-Cubes Simplification using DP segmentation

Idea [CGS04]

Simplify the MC surface taking into account the DP segmentation and the DPS preimages

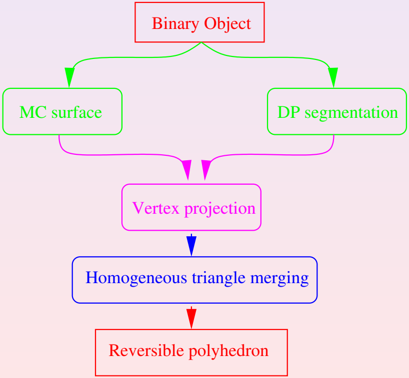


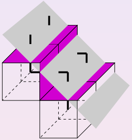


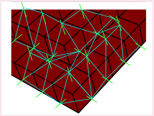
Marching-Cubes Simplification using DP segmentation

Idea [CGS04]

Simplify the MC surface taking into account the DP segmentation and the DPS preimages





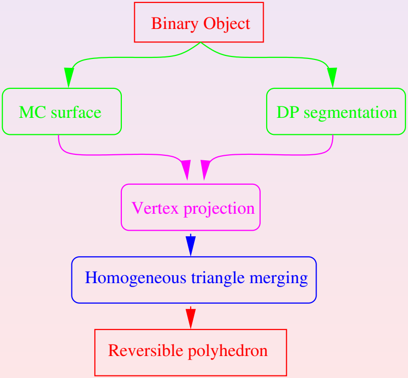


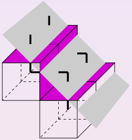
David Coeurjolly and Isabelle Sivignon

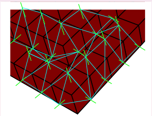
Marching-Cubes Simplification using DP segmentation

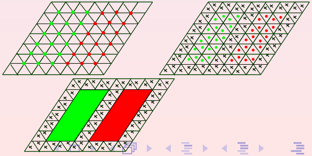
Idea [CGS04]

Simplify the MC surface taking into account the DP segmentation and the DPS preimages

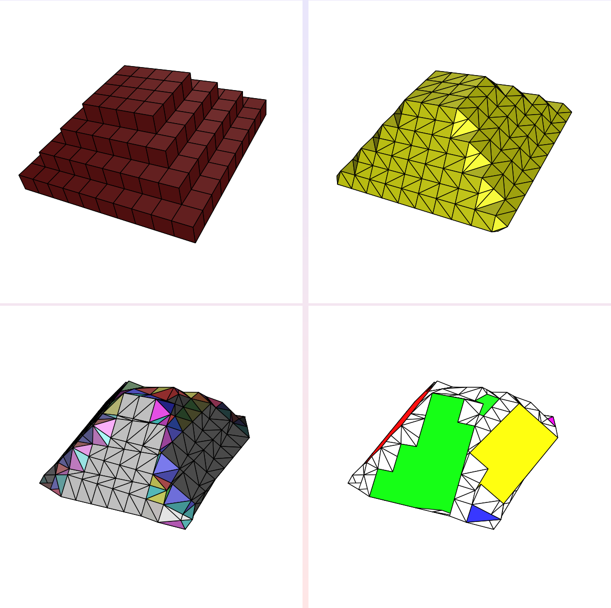




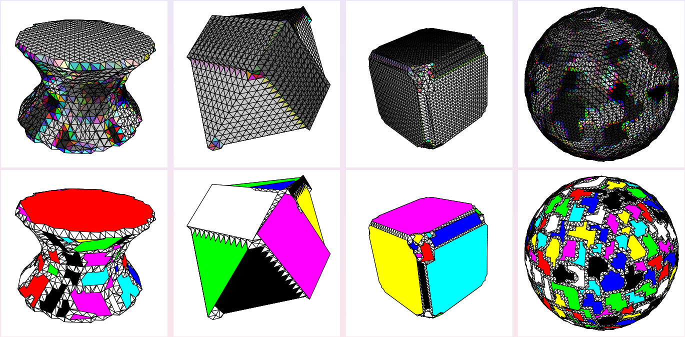




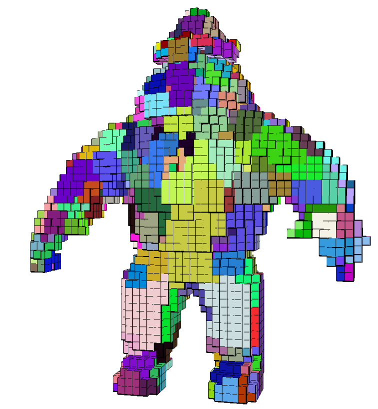
Some results

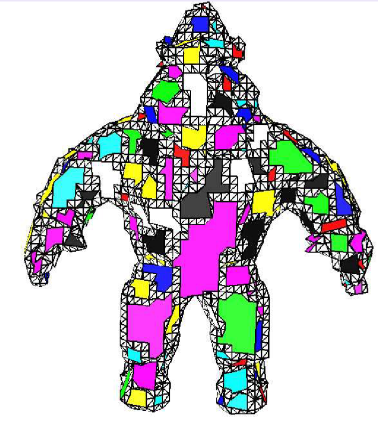


Results



Results





David Coeurjolly and Isabelle Sivignon

Table of contents

Conclusion

Preimage in 3-D

- Convenient output sensitive tool to recognize DPS
- Size of the preimage is *small*

Open problems

- Is it possible to precise the arithmetical structure of $\bar{S}$?
- General tight bound on E in 3-D and in n-D

Number of vertices in Interger Linear Programming problems

[Sch86, BHL92, Zol00, Bar02]

Integer Linear Programming problem

Maximize the linear function $\sum_{j=1}^{n} c_j x_j$ subject to $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$ $(i = 1, \ldots, m)$ with:

- $x_j \in \mathbb{Z}$ $(j = 1, \ldots, n)$
- a_{ij}, c_j, b_i are integers

- m is the number of inequalities
- n is the dimension of the inequalities
- $A = (a_{ij}) \in \mathbb{Z}^{m \times n}$, $b = (b_i) \in \mathbb{Z}^m$, $c = (c_j) \in \mathbb{Z}^n$
- $\alpha = \max\{|a_{ij}|, i = 1, \ldots, m, j = 1, \ldots, n\}$

Main result

The number of vertices of the convex hull $N(A, b)$ of the feasible region (supposed non-empty) $Ax \leq b$ is given by:

$$c'_n m^{\lfloor n/2 \rfloor} \log^{n-1}(\alpha) \leq |N(A, b)| \leq c_n m^{\lfloor n/2 \rfloor} \log^{n-1}(1 + \alpha)$$

Return

Number of vertices in Interger Linear Programming problems

[Sch86, BHL92, Zol00, Bar02]

Integer Linear Programming problem

Maximize the linear function $\sum_{j=1}^{n} c_j x_j$ subject to $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$ $(i = 1, \ldots, m)$ with:

- $x_j \in \mathbb{Z}$ $(j = 1, \ldots, n)$
- a_{ij}, c_j, b_i are integers

- m is the number of inequalities
- n is the dimension of the inequalities
- $A = (a_{ij}) \in \mathbb{Z}^{m \times n}$, $b = (b_i) \in \mathbb{Z}^m$, $c = (c_j) \in \mathbb{Z}^n$
- $\alpha = \max\{|a_{ij}|, i = 1, \ldots, m, j = 1, \ldots, n\}$

Main result

The number of vertices of the convex hull $N(A, b)$ of the feasible region (supposed non-empty) $Ax \leq b$ is given by:

$$c'_n m^{\lfloor n/2 \rfloor} \log^{n-1}(\alpha) \leq |N(A, b)| \leq c_n m^{\lfloor n/2 \rfloor} \log^{n-1}(1 + \alpha)$$

Return

Number of vertices in Interger Linear Programming problems

[Sch86, BHL92, Zol00, Bar02]

Integer Linear Programming problem

Maximize the linear function $\sum_{j=1}^{n} c_j x_j$ subject to $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$ $(i = 1, \ldots, m)$ with:

- $x_j \in \mathbb{Z}$ $(j = 1, \ldots, n)$
- a_{ij}, c_j, b_i are integers

- m is the number of inequalities
- n is the dimension of the inequalities
- $A = (a_{ij}) \in \mathbb{Z}^{m \times n}$, $b = (b_i) \in \mathbb{Z}^m$, $c = (c_j) \in \mathbb{Z}^n$
- $\alpha = \max\{|a_{ij}|, i = 1, \ldots, m, j = 1, \ldots, n\}$

Main result

The number of vertices of the convex hull $N(A, b)$ of the feasible region (supposed non-empty) $Ax \leq b$ is given by:

$$c'_n m^{\lfloor n/2 \rfloor} \log^{n-1}(\alpha) \leq |N(A, b)| \leq c_n m^{\lfloor n/2 \rfloor} \log^{n-1}(1 + \alpha)$$

Return

Barany.
Random points, convex bodies, lattices.
In *ICM: Proceedings of the International Congress of Mathematicians*, 2002.

Barany, Howe, and Lovasz.
On integer points in polyhedra: A lower bound.
Combinatorica, 12, 1992.

D. Coeurjolly, Alexis Guillaume, and I. Sivignon.
Reversible discrete volume polyhedrization using marching cubes simplification.
In *SPIE Vision Geometry XII*, volume 5300, pages 1–11, San Jose, USA, 2004.

D. Coeurjolly, I. Sivignon, F. Dupont, F. Feschet, and J.-M. Chassery.
On digital plane preimage structure.
Discrete Applied Mathematics, 151(1–3):78–92, 2005.

W. E. Lorensen and H. E. Cline.
Marching cubes: a high resolution 3D surface construction algorithm.
In M. C. Stone, editor, *SIGGRAPH '87 Conference Proceedings (Anaheim, CA, July 27–31, 1987)*, pages 163–170. Computer Graphics, Volume 21, Number 4, July 1987.

N. Megiddo.
Linear programming in linear time when the dimension is fixed.
Journal of the ACM, 31(1):114–127, January 1984.

F. P. Preparata and M. I. Shamos.
Computational Geometry : An Introduction.
Springer-Verlag, 1985.

J.-P. Reveillès.
Géométrie discrète, calcul en nombres entiers et algorithmique.
PhD thesis, Université Louis Pasteur - Strasbourg, 1991.

A. Schrijver.
Theory of Linear and Integer Programming.
Wiley and Sons, 1986.

Isabelle Sivignon, Florent Dupont, and Jean-Marc Chassery.
Reversible polygonalization of a 3D planar discrete curve: Application on discrete surfaces.

In *DGCI*, pages 347–358, 2005.

J. Vittone and J.-M. Chassery.
Recognition of digital naive planes and polyhedization.
In G. Borgefors, I. Nyström, and G. Sanniti di Baja, editors, *9th Discrete Geometry for Computer Imagery*, volume 1953 of *LNCS*, pages 296–307. Springer, 2000.

N. Y. Zolotykh.
On the number of vertices in integer linear programming problems.
Technical report, University of Nizhni Novgorod, 2000.