Two linear-time algorithms for computing the minimum length polygon of a digital contour

Jacques-Olivier Lachaud and Xavier Provençal

October 1, 2009

LAMA Laboratoire de Mathématiques
LIRMM Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier
Fonds de recherche sur la nature et les technologies Québec

Polyominoes

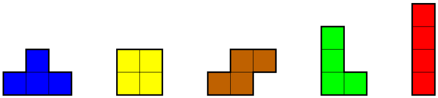
Definition

A *polyomino* is a set of digital squares in the plane such that its topoligical boundary is a Jordan curve.

Polyominoes

Definition

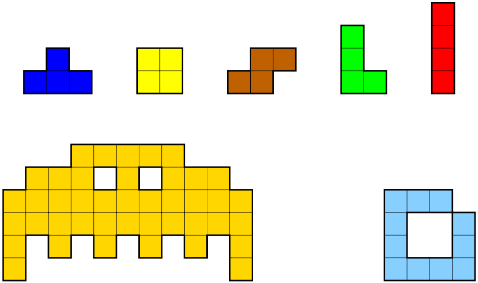
A *polyomino* is a set of digital squares in the plane such that its topoligical boundary is a Jordan curve.



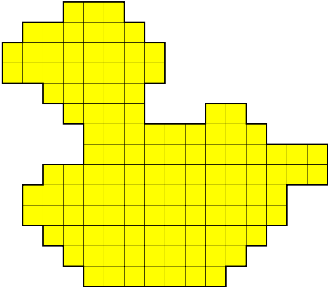
Polyominoes

Definition

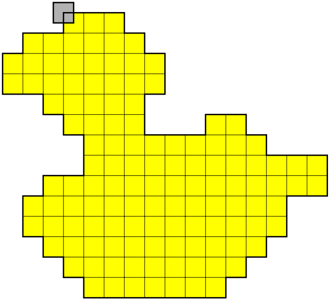
A *polyomino* is a set of digital squares in the plane such that its topoligical boundary is a Jordan curve.



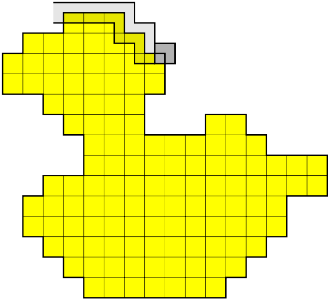
Minimum Length Polygon



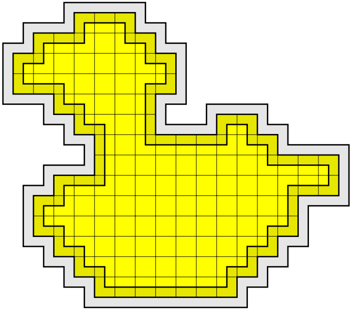
Minimum Length Polygon



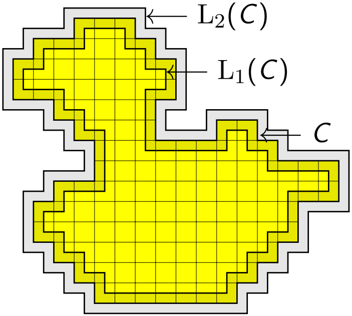
Minimum Length Polygon



Minimum Length Polygon



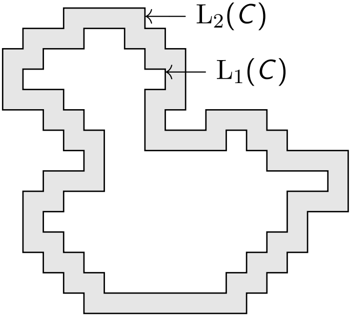
Minimum Length Polygon



Definition

Given a digital contour C, its *inner* (resp. *outer*) *polygon* $\mathrm{L}_1(C)$ (resp. $\mathrm{L}_2(C)$) is the erosion (resp. dilatation) of the body of $I(C)$ by the open unit square centrer on $(0, 0)$.

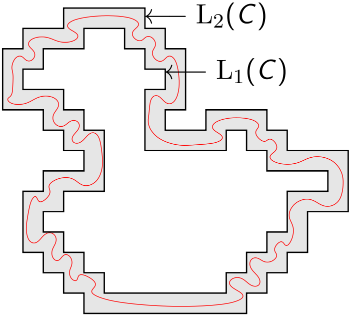
Minimum Length Polygon



Definition

Given a digital contour C, its *inner* (resp. *outer*) *polygon* $\mathrm{L}_1(C)$ (resp. $\mathrm{L}_2(C)$) is the erosion (resp. dilatation) of the body of $I(C)$ by the open unit square centrer on $(0, 0)$.

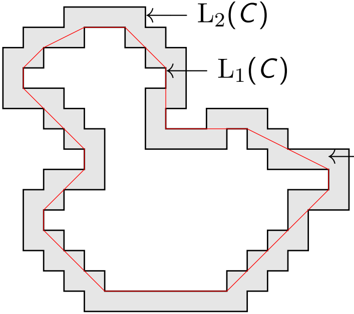
Minimum Length Polygon



Definition

Given a digital contour C, its *inner* (resp. *outer*) *polygon* $L_1(C)$ (resp. $L_2(C)$) is the erosion (resp. dilatation) of the body of $I(C)$ by the open unit square centrer on $(0, 0)$.

Minimum Length Polygon



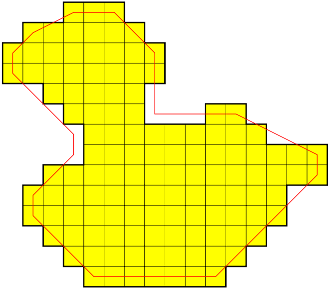
Definition

The *minimum length polygon* of C is a subset $P \in \mathbb{R}^2$ such that,

$$P = \underset{A \in \mathcal{A},\, \mathrm{L}_1(C) \subseteq A,\ \partial A \subset \mathrm{L}_2(C) \setminus \mathrm{L}_1(C)^\circ}{\arg\min} \mathrm{Per}(A)$$

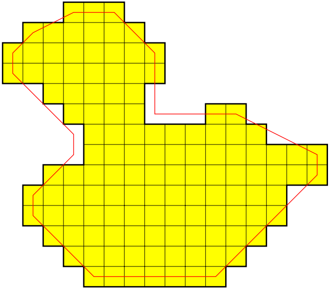
where $\mathcal{A}$ is the family of simply connected compact sets of $\mathbb{R}^2$.

Minimum Length Polygon



The MLP is:

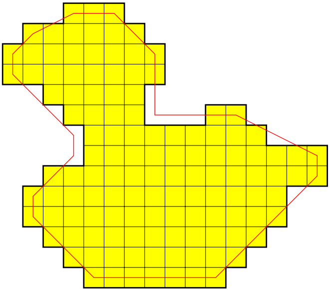
Minimum Length Polygon



The MLP is:

- a good length estimator;

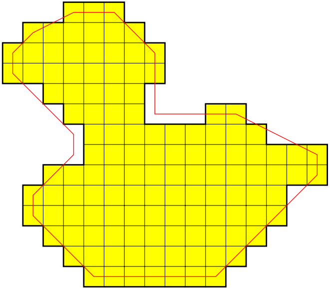
Minimum Length Polygon



The MLP is:

- a good length estimator;
- a good tangent estimator;

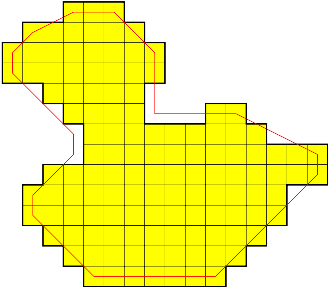
Minimum Length Polygon



The MLP is:

- a good length estimator;
- a good tangent estimator;
- characteristic of the shape's convexity;

Minimum Length Polygon



The MLP is:

- a good length estimator;
- a good tangent estimator;
- characteristic of the shape's convexity;
- reversible*.

Relative Convex Hull

Definition

Let $U \in \mathbb{R}^2$ and $V \subseteq U$. The set V is said to be *U-convex* if for every $x, y \in V$

$$\overline{xy} \subseteq U \implies \overline{xy} \subseteq V.$$

Relative Convex Hull

Definition

Let $U \in \mathbb{R}^2$ and $V \subseteq U$. The set V is said to be *U-convex* if for every $x, y \in V$

$$\overline{xy} \subseteq U \implies \overline{xy} \subseteq V.$$

Definition

Let $V \subseteq U \subseteq \mathbb{R}^2$, the intersection of all U-convex sets containing V is called the *U-convex hull* of V.

Relative Convex Hull

Definition

Let $U \in \mathbb{R}^2$ and $V \subseteq U$. The set V is said to be *U-convex* if for every $x, y \in V$

$$\overline{xy} \subseteq U \implies \overline{xy} \subseteq V.$$

Definition

Let $V \subseteq U \subseteq \mathbb{R}^2$, the intersection of all U-convex sets containing V is called the *U-convex hull* of V.

- Every set $U \subseteq \mathbb{R}^2$ is U-convex.

Relative Convex Hull

Definition

Let $U \in \mathbb{R}^2$ and $V \subseteq U$. The set V is said to be *U-convex* if for every $x, y \in V$

$$\overline{xy} \subseteq U \implies \overline{xy} \subseteq V.$$

Definition

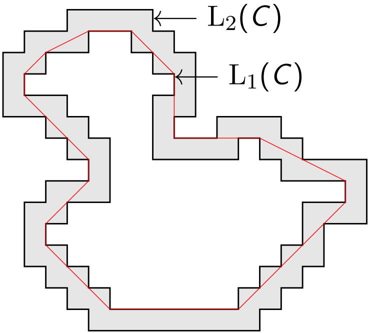
Let $V \subseteq U \subseteq \mathbb{R}^2$, the intersection of all U-convex sets containing V is called the *U-convex hull* of V.

- Every set $U \subseteq \mathbb{R}^2$ is U-convex.
- The usual convexity is the $\mathbb{R}^2$-convexity.

Set-Theoretic MLP

Theorem (Sloboda, Stoer 1994)

The MLP of a digital contour C, is the $\mathrm{L}_2(C)$-convex hull of $\mathrm{L}_1(C)$.



The Arithmetic MLP

Tangential Cover

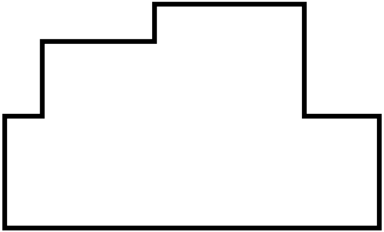
Definition

The *tangential cover* of a discrete contour is the set of its Maximal Digital Straight Segments (MDSS).

Tangential Cover

Definition

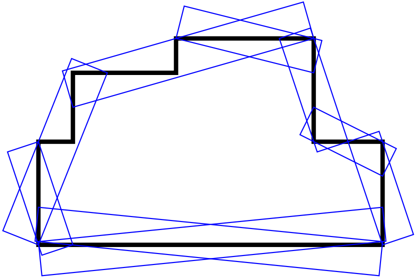
The *tangential cover* of a discrete contour is the set of its Maximal Digital Straight Segments (MDSS).



Tangential Cover

Definition

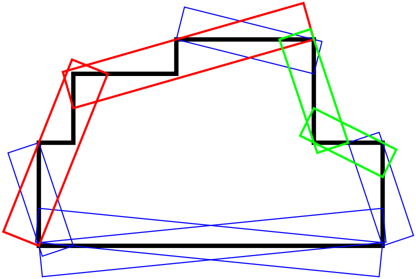
The *tangential cover* of a discrete contour is the set of its Maximal Digital Straight Segments (MDSS).



Tangential Cover

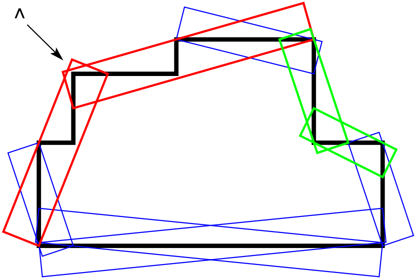
Definition

The *tangential cover* of a discrete contour is the set of its Maximal Digital Straight Segments (MDSS).



Definition

The *tangential cover* of a discrete contour is the set of its Maximal Digital Straight Segments (MDSS).

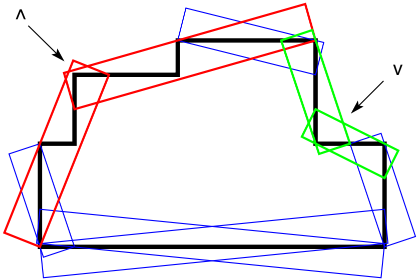


- ∧ stands for a "convex turn",

Tangential Cover

Definition

The *tangential cover* of a discrete contour is the set of its Maximal Digital Straight Segments (MDSS).

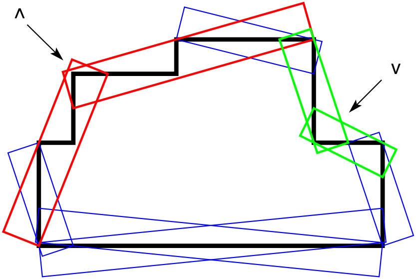


- $\wedge$ stands for a "convex turn",
- $\vee$ stands for a "concave turn".

Tangential Cover

Definition

The *tangential cover* of a discrete contour is the set of its Maximal Digital Straight Segments (MDSS).

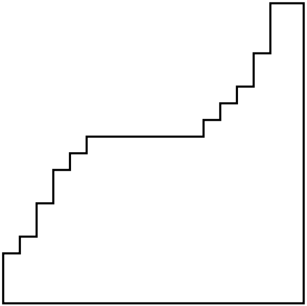


$\dots$	$\wedge \overline{M_i} \wedge \dots \wedge \overline{M_j} \wedge$	$\overline{M_{j+1}}$	$\vee \overline{M_{j+2}} \vee \dots \vee \overline{M_k} \vee$	$\overline{M_{k+1}}$	$\wedge \dots$
$\dots$	convex zone	inflexion	concave zone	inflexion	$\dots$

Tangential Cover and Convexity

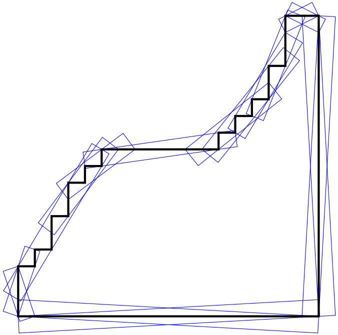
Theorem (Dorksen-Reiter, Debled-Rennesson 2006)

A digital contour is digitally convex iff every couple of consecutive MDSS of its tangential cover is made of $\wedge$-turns.



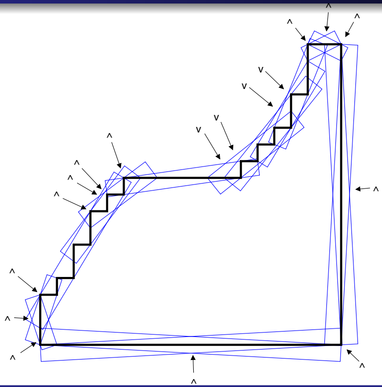
Definition

A digital contour C with tangential cover $M_1, M_2, \ldots, M_l$ is uniquely split into a sequence of closed connected sets with a single point overlap as follows :



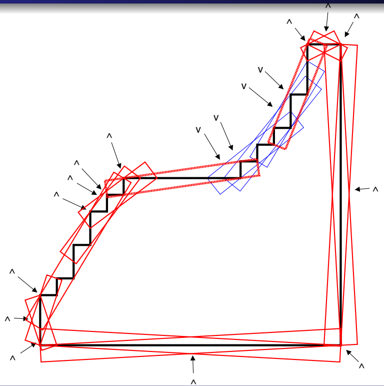
Definition

A digital contour C with tangential cover $M_1, M_2, \ldots, M_l$ is uniquely split into a sequence of closed connected sets with a single point overlap as follows :



Definition

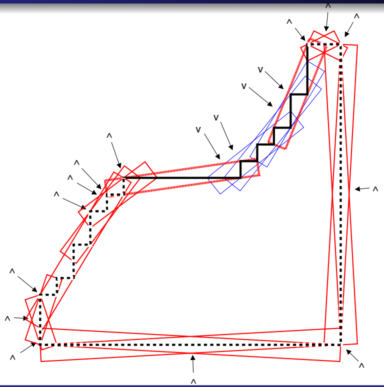
A digital contour C with tangential cover $M_1, M_2, \ldots, M_l$ is uniquely split into a sequence of closed connected sets with a single point overlap as follows :



Definition

A *convex zone or* $(\wedge, \wedge)$*-zone* is an inextensible sequence of consecutive $\wedge$-turns from (M_i, M_{i+1}) to (M_j, M_{j+1}). If $i \neq j$, if starts at $U_l(M_i)$ and ends at $U_f(M_{j+1})$.

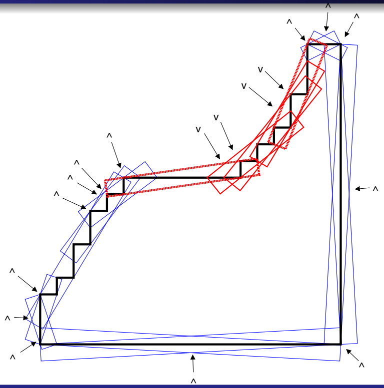
Convex, Concave and Inflexion Zones



Definition

A *convex zone or* $(\wedge, \wedge)$*-zone* is an inextensible sequence of consecutive $\wedge$-turns from (M_i, M_{i+1}) to (M_j, M_{j+1}). If $i \neq j$, if starts at $U_l(M_i)$ and ends at $U_f(M_{j+1})$.

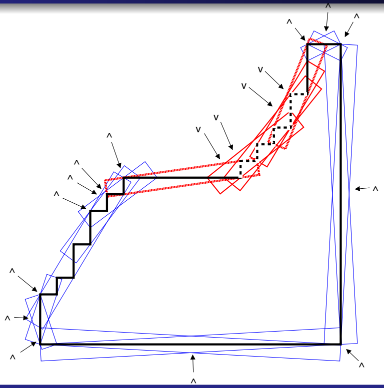
Convex, Concave and Inflexion Zones



Definition

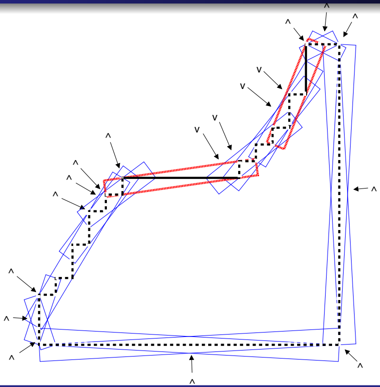
A *concave zone or* $(\vee, \vee)$*-zone* is an inextensible sequence of consecutive $\vee$-turns from (M_i, M_{i+1}) to (M_j, M_{j+1}). If $i \neq j$, if starts at $L_l(M_i)$ and ends at $L_f(M_{j+1})$.

Convex, Concave and Inflexion Zones



Definition

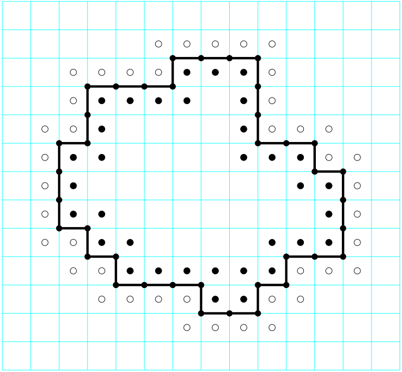
A *concave zone or* $(\vee,\vee)$*-zone* is an inextensible sequence of consecutive $\vee$-turns from (M_i, M_{i+1}) to (M_j, M_{j+1}). If $i \neq j$, if starts at $L_l(M_i)$ and ends at $L_f(M_{j+1})$.



Definition

A *convex inflexion zone or* $(\wedge, \vee)$*-zone* is a $\wedge$-turn followed by a $\vee$-turn around a MDSS M_i. It starts at $U_f(M_i)$ and ends at $L_L(M_i)$. A *concave inflexion zone or* $(\vee, \wedge)$*-zone* is a $\vee$-turn followed by a $\wedge$-turn around a MDSS M_i. It starts at $L_f(M_i)$ and ends at $U_L(M_i)$.

Inside and Outside Pixels



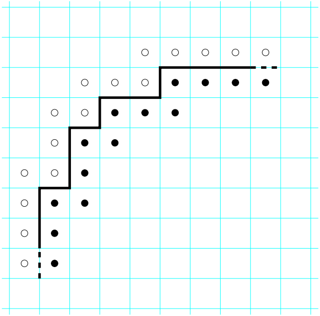
• are called *inside* pixels.
○ are called *outside* pixels.

Definition of AMLP

The AMLP of C is defined on each zone according to its type:

Definition of AMLP

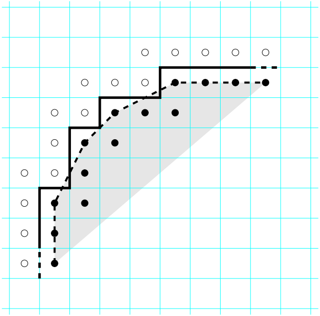
The AMLP of C is defined on each zone according to its type:



Definition of AMLP

The AMLP of C is defined on each zone according to its type:

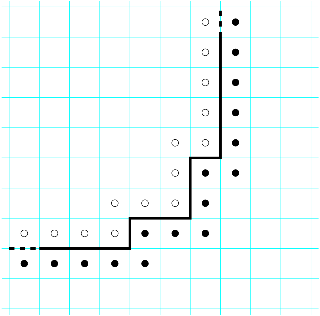
- $(\wedge, \wedge)$: left side of the left enveloppe,



Definition of AMLP

The AMLP of C is defined on each zone according to its type:

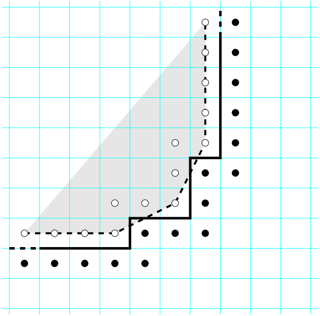
- $(\wedge, \wedge)$: left side of the left enveloppe,
- $(\vee, \vee)$: right side of the right enveloppe,



Definition of AMLP

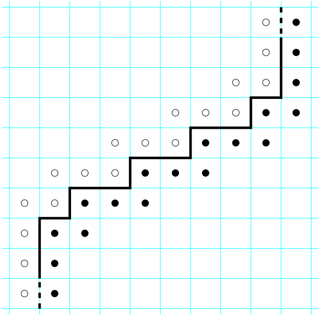
The AMLP of C is defined on each zone according to its type:

- $(\wedge, \wedge)$: left side of the left enveloppe,
- $(\vee, \vee)$: right side of the right enveloppe,



The AMLP of C is defined on each zone according to its type:

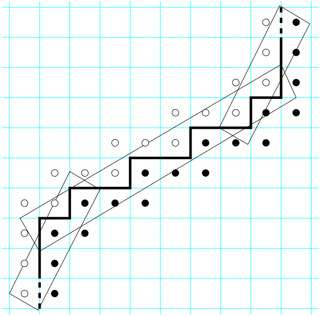
- $(\wedge, \wedge)$: left side of the left enveloppe,
- $(\vee, \vee)$: right side of the right enveloppe,
- $(\wedge, \vee)$:



Definition of AMLP

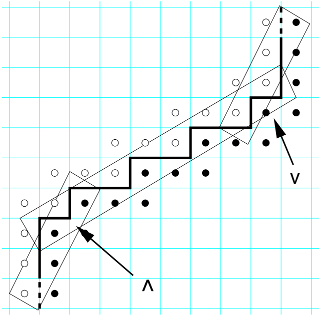
The AMLP of C is defined on each zone according to its type:

- $(\wedge, \wedge)$: left side of the left enveloppe,
- $(\vee, \vee)$: right side of the right enveloppe,
- $(\wedge, \vee)$:



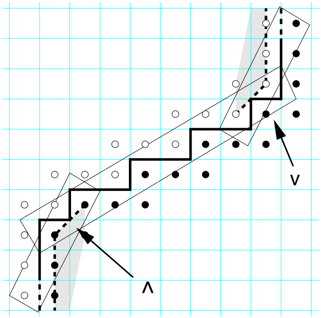
The AMLP of C is defined on each zone according to its type:

- $(\wedge, \wedge)$: left side of the left enveloppe,
- $(\vee, \vee)$: right side of the right enveloppe,
- $(\wedge, \vee)$:



The AMLP of C is defined on each zone according to its type:

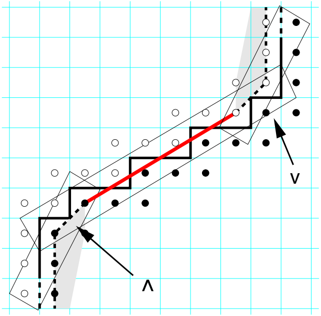
- $(\wedge, \wedge)$: left side of the left enveloppe,
- $(\vee, \vee)$: right side of the right enveloppe,
- $(\wedge, \vee)$:



Definition of AMLP

The AMLP of C is defined on each zone according to its type:

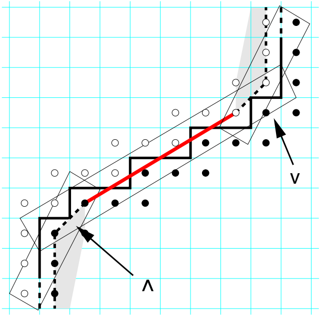
- $(\wedge, \wedge)$: left side of the left enveloppe,
- $(\vee, \vee)$: right side of the right enveloppe,
- $(\wedge, \vee)$: segment joining U_f to L_l,



Definition of AMLP

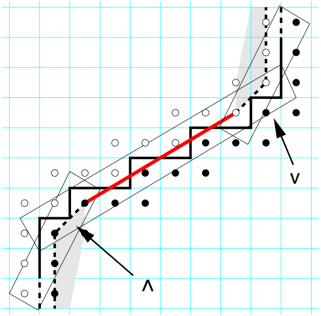
The AMLP of C is defined on each zone according to its type:

- $(\wedge, \wedge)$: left side of the left enveloppe,
- $(\vee, \vee)$: right side of the right enveloppe,
- $(\wedge, \vee)$: segment joining U_f to L_l,
- $(\vee, \wedge)$: segment joining L_f to U_l.



The AMLP of C is defined on each zone according to its type:

- $(\wedge, \wedge)$: left side of the left enveloppe,
- $(\vee, \vee)$: right side of the right enveloppe,
- $(\wedge, \vee)$: segment joining U_f to L_l,
- $(\vee, \wedge)$: segment joining L_f to U_l.



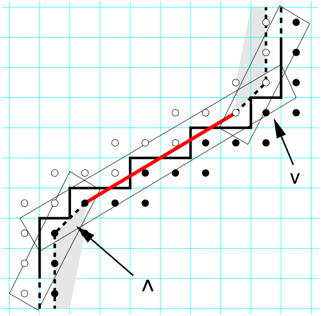
Theorem

Given a digital contour C, the AMLP of C is the MLP of C.

Definition of AMLP

The AMLP of C is defined on each zone according to its type:

- $(\wedge, \wedge)$: left side of the left enveloppe,
- $(\vee, \vee)$: right side of the right enveloppe,
- $(\wedge, \vee)$: segment joining U_f to L_l,
- $(\vee, \wedge)$: segment joining L_f to U_l.



Theorem

Given a digital contour C, the AMLP of C is the MLP of C.

Proof: show that the AMLP of C is the convex hull of $\mathrm{L}_1(C)$ relatively to $\mathrm{L}_2(C)$.

Computation of AMLP

Algorithm

Three steps to compute the AMLP if C:

Computation of AMLP

Algorithm

Three steps to compute the AMLP if C:

(1) Compute $(M_i)_{i=1..N}$ the tangential cover of C;

(1) is performed in $O(n)$ using Lachaud, Vialard and de Vieilleville 2007 (DGCI 2005).

Computation of AMLP

Algorithm

Three steps to compute the AMLP if C:

(1) Compute $(M_i)_{i=1..N}$ the tangential cover of C;

(2) Decompose $(M_i)_{i=1..N}$ is (α, β)-zones. $(\alpha, \beta \in \{\wedge, \vee\})$

(1) is performed in $O(n)$ using Lachaud, Vialard and de Vieilleville 2007 (DGCI 2005).

(2) is performed in $O(N)$ where $N \leq n$.

Computation of AMLP

Algorithm

Three steps to compute the AMLP if C:

(1) Compute $(M_i)_{i=1..N}$ the tangential cover of C;

(2) Decompose $(M_i)_{i=1..N}$ is (α, β)-zones. $(\alpha, \beta \in \{\wedge, \vee\})$

(3) For each (α, β)-zones, compute the associate part of the AMLP.

(1) is performed in $O(n)$ using Lachaud, Vialard and de Vieilleville 2007 (DGCI 2005).

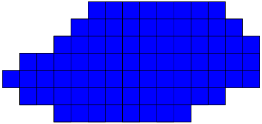
(2) is performed in $O(N)$ where $N \leq n$.

(3) is performed in $O(n)$ using Melkman 1987 on each $(\wedge, \wedge)$ or $(\vee, \vee)$-zones; while each $(\wedge, \vee)$ or $(\vee, \wedge)$-zones is treated in constant time.

The Combinatorial MLP

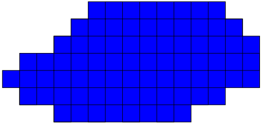
- A *contour word* is the Freeman code of the border of a polyomino.

- A *contour word* is the Freeman code of the border of a polyomino.



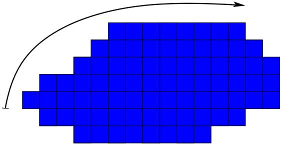
10100101010000000030303332232232222222212212

- A *contour word* is the Freeman code of the border of a polyomino.
- A *quadrant word* is an inextendable factor of a contour word over two letters.



10100101010000000030303332232232222222212212

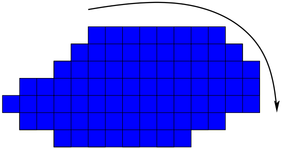
- A *contour word* is the Freeman code of the border of a polyomino.
- A *quadrant word* is an inextendable factor of a contour word over two letters.



101001010100000000303033322322322222222212212

Combinatorial View of Digital Convexity

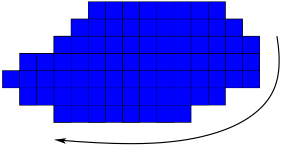
- A *contour word* is the Freeman code of the border of a polyomino.
- A *quadrant word* is an inextendable factor of a contour word over two letters.



10100101010000000030303332232232222222212212

Combinatorial View of Digital Convexity

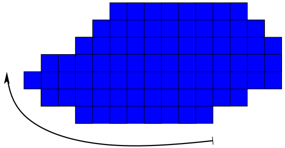
- A *contour word* is the Freeman code of the border of a polyomino.
- A *quadrant word* is an inextendable factor of a contour word over two letters.



101001010100000000303033322322322222222212212

Combinatorial View of Digital Convexity

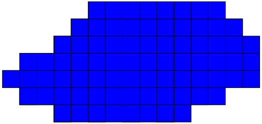
- A *contour word* is the Freeman code of the border of a polyomino.
- A *quadrant word* is an inextendable factor of a contour word over two letters.



1010010101000000003030333223223222222221221 2

Combinatorial View of Digital Convexity

- A *contour word* is the Freeman code of the border of a polyomino.
- A *quadrant word* is an inextendable factor of a contour word over two letters.

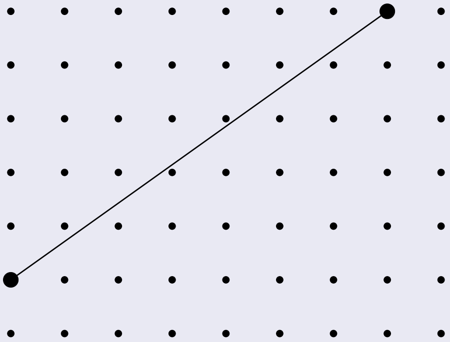


Theorem (Brlek, Lachaud, P., Reutenauer "DGCI 2008")

An hv-convex polyomino P is digitally if and only if the each of its quadrant words q_i is such that its factorization as decreasing Lyndon words $q_i = l_1^{n_1} l_2^{n_2} \cdots l_k^{n_k}$ contains only Christoffel words. In such case, this factorization coincide with its Euclidian convex.

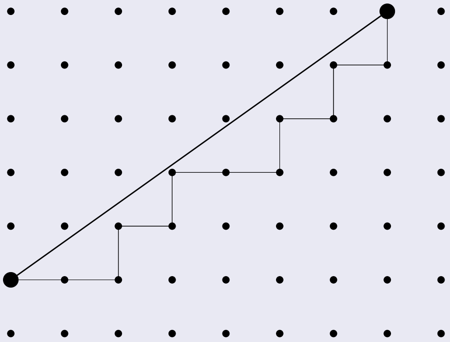
Christoffel Words

Definition (Borel and Laubie 1993)

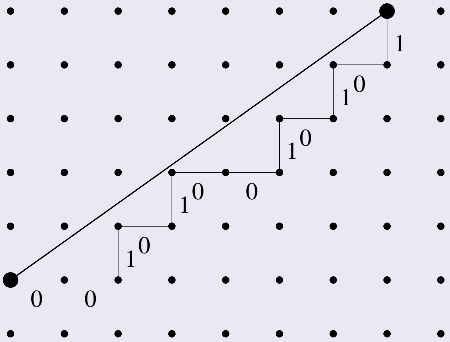


Christoffel Words

Definition (Borel and Laubie 1993)



Definition (Borel and Laubie 1993)



$$\mathcal{C}_{5/7} = 001010010101.$$

Lyndon Words

Definition

A word w over an ordered alphabet is a Lyndon word if for all non-empty words u and v:

$$w = uv \implies w < vu.$$

(where $<$ denotes the lexicographic order.)

Theorem (Lyndon 1950)

Any word w over an ordered alphabet admits a unique factorization as decreasing Lyndon words.

Lyndon Words

Definition

A word w over an ordered alphabet is a Lyndon word if for all non-empty words u and v:

$$w = uv \implies w < vu.$$

(where $<$ denotes the lexicographic order.)

Theorem (Lyndon 1950)

Any word w over an ordered alphabet admits a unique factorization as decreasing Lyndon words.

Theorem (Duval 1983)

Given a word w of length n, its factorization as decreasing Lyndon words $w = l_1^{n_1} l_2^{n_2} \cdots l_k^{n_k}$ is computed in $O(n)$ using Duval's algorithm.

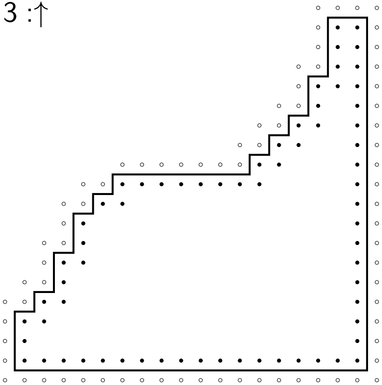
Adapted Version of Duval's Algorithm

Using the repetition properties of Christoffel words, we modify Duval's algorithm in order to stop the computation if the prefix read is not prefix of a Christoffel word.

Computation of the CMLP

$0 : \rightarrow,\ 1 : \downarrow,\ 2 : \leftarrow, 3 : \uparrow$

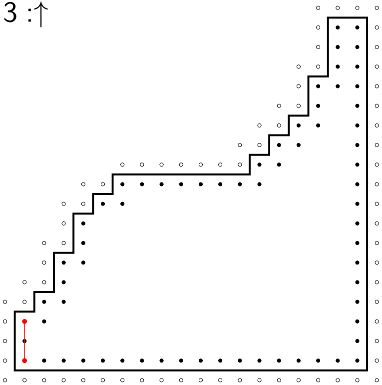
$3 < 0 < 1 < 2$



111010110110101000000010101011011100333333333333333333222222222222222222

$0 : \rightarrow,\ 1 : \downarrow,\ 2 : \leftarrow, 3 : \uparrow$

$3 < 0 < 1 < 2$

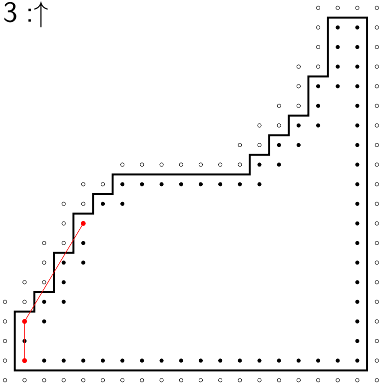


010110110101000000010101011011100333333333333333333222222222222222222

$(1)^3$

$0 : \rightarrow,\ 1 : \downarrow,\ 2 : \leftarrow, 3 : \uparrow$

$3 < 0 < 1 < 2$

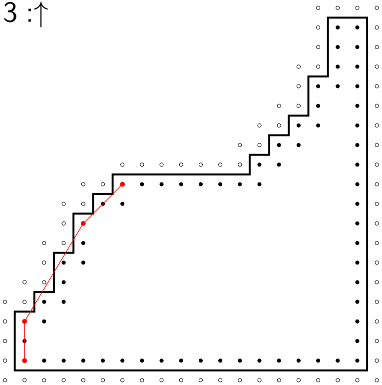


010100000001010101101110033333333333333333322222222222222222222

$(1)^3 \cdot (01011011)$

0 :$\rightarrow$, 1 :$\downarrow$, 2 :$\leftarrow$, 3 :$\uparrow$

$3 < 0 < 1 < 2$



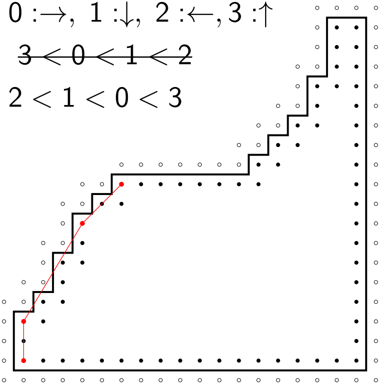
000000010101011011110033333333333333333322222222222222222

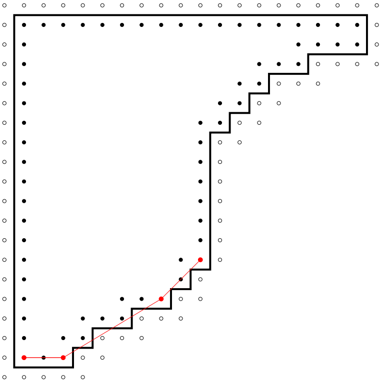
$(1)^3 \cdot (01011011) \cdot (0101)$

$0 :\rightarrow,\ 1 :\downarrow,\ 2 :\leftarrow, 3 :\uparrow$

~~$3 < 0 < 1 < 2$~~

$2 < 1 < 0 < 3$





000000010101011011100333333333333333333322222222222222222

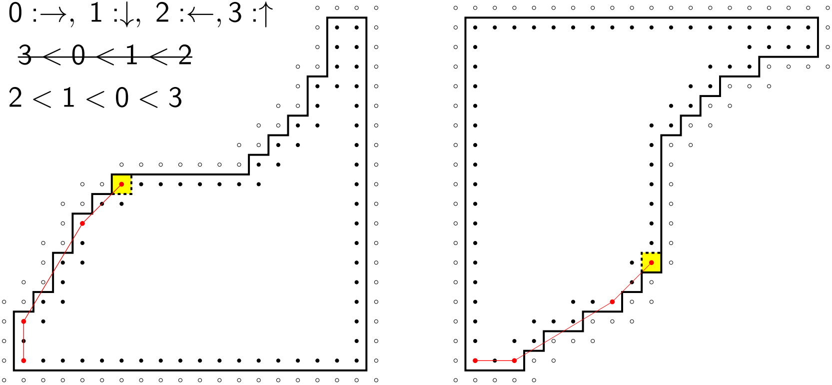
$(1)^3 \cdot (01011011) \cdot (0101)$

Computation of the CMLP

$0 :\rightarrow,\ 1 :\downarrow,\ 2 :\leftarrow, 3 :\uparrow$

~~$3 < 0 < 1 < 2$~~

$2 < 1 < 0 < 3$



1000000101010110111003333333333333333332222222222222222222

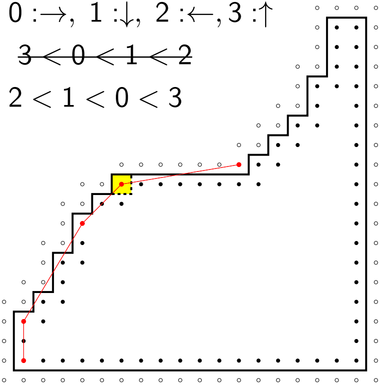
$(1)^3 \cdot (01011011) \cdot (0101)$

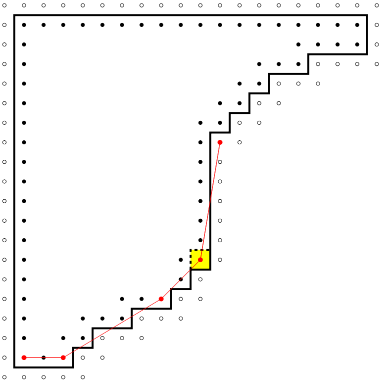
Computation of the CMLP

$0 : \rightarrow,\ 1 : \downarrow,\ 2 : \leftarrow, 3 : \uparrow$

~~$3 < 0 < 1 < 2$~~

$2 < 1 < 0 < 3$





10101011011100333333333333333333222222222222222222

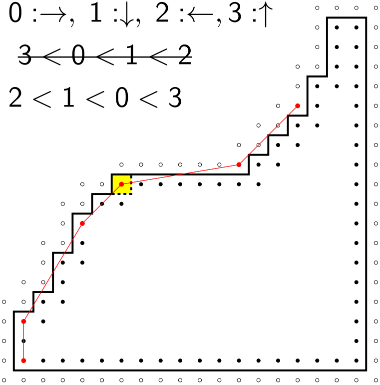
$(1)^3 \cdot (01011011) \cdot (0101) \cdot (1000000)$

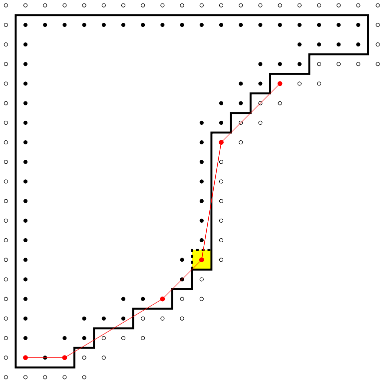
Computation of the CMLP

$0 :\rightarrow,\ 1 :\downarrow,\ 2 :\leftarrow, 3 :\uparrow$

~~$3 < 0 < 1 < 2$~~

$2 < 1 < 0 < 3$





11011100333333333333333333222222222222222222

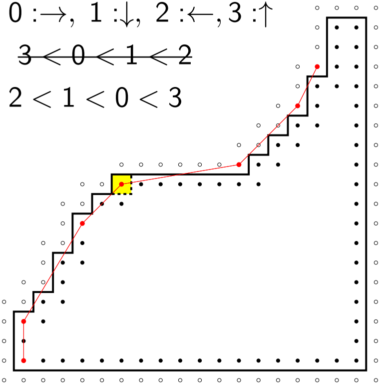
$(1)^3 \cdot (01011011) \cdot (0101) \cdot (1000000) \cdot (101010)$

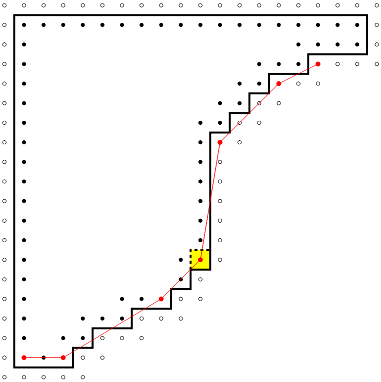
Computation of the CMLP

0 :→, 1 :↓, 2 :←, 3 :↑

~~$3 < 0 < 1 < 2$~~

$2 < 1 < 0 < 3$





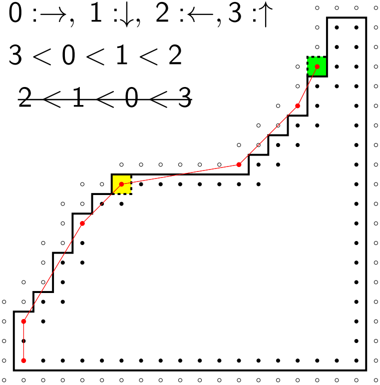
111003333333333333333332222222222222222222

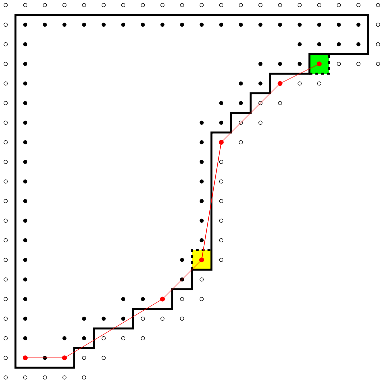
(1)^3·(01011011)·(0101)·(1000000)·(101010)·(110)

$0 :\rightarrow,\ 1 :\downarrow,\ 2 :\leftarrow, 3 :\uparrow$

$3 < 0 < 1 < 2$

~~$2 < 1 < 0 < 3$~~





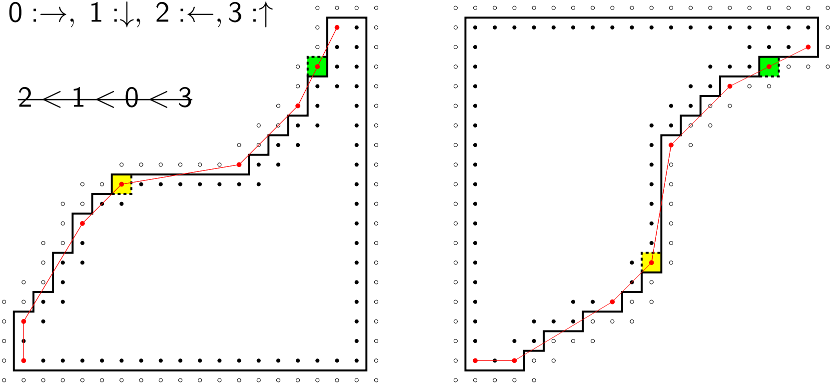
0110033333333333333333322222222222222222

$(1)^3 \cdot (01011011) \cdot (0101) \cdot (1000000) \cdot (101010) \cdot (110)$

Computation of the CMLP

$0 :\rightarrow,\ 1 :\downarrow,\ 2 :\leftarrow, 3 :\uparrow$

~~$2 < 1 < 0 < 3$~~



00333333333333333333222222222222222222

$(1)^3 \cdot (01011011) \cdot (0101) \cdot (1000000) \cdot (101010) \cdot (110) \cdot (011)$

Both algorithms AMLP and CMLP have been implemented and are include in the ImaGene project.
http://gforge.liris.cnrs.fr/projects/imagene

Thank you!