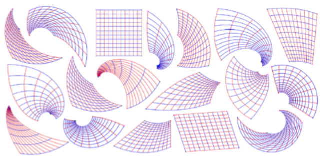
Corner-Operated Tran-Similar (COTS) maps, patterns, and lattices

JAREK ROSSIGNAC

SCHOOL OF INTERACTIVE COMPUTING
GEORGIA INSTITUTE OF TECHNOLOGY

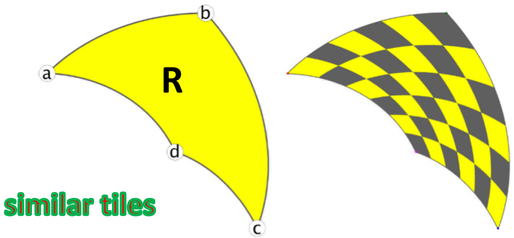


Georgia Tech

Corner-Operated Tran-Similar (COTS) map

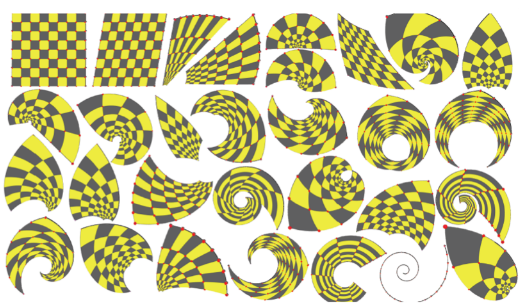
Maps unit square onto a planar & curved quad R

Controlled by four coplanar corner points: a, b, c, d



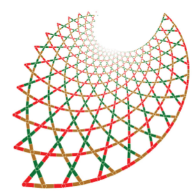
Defines pattern of similar tiles

Supports wide diversity of warps

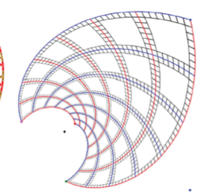


Fully defined by the four corners

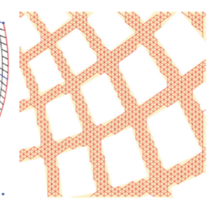
Applications to planar lattices



Kagome

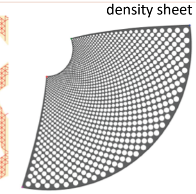


Multi-resolution LOL
(Lattice-of-Lattices)



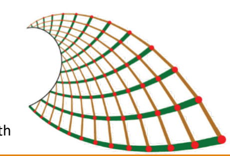
Lattice-in-Lattice (LiL)

Constant aggregate
density sheet



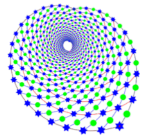
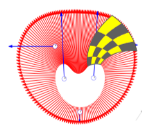
Constant cost queries

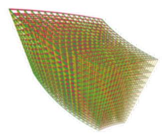
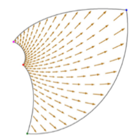
SAL (Stress-Aligned Lattice)
Extension of Michell truss with



Examples of other applications

- Aesthetic texture warps
- Steady patterns
- Steady animation
- Art
- 3D lattices
- Design of steady vector fields

Notation

Integers are in lowercase (i or j). ***Scalars*** are in italics (u or λ). ***Points*** are in bold lowercase ($\mathbf{f}$ or $\mathbf{p_i}$) or in parentheses using Cartesian coordinates ("(x, y)"). ***Vectors*** are in lowercase bold with an overhead arrow ($\vec{\mathbf{v}}$) or in brackets using Cartesian components ("$\langle x, y\rangle$"). Vector from $\mathbf{a}$ to $\mathbf{c}$ is written $\overrightarrow{\mathbf{ac}}$. Sum $\mathbf{p} + \vec{\mathbf{v}}$ denotes $\mathbf{p}$ translated by $\vec{\mathbf{v}}$. The ***angle*** between $\vec{\mathbf{u}}$ and $\vec{\mathbf{v}}$ is written $\vec{\mathbf{u}}^{\vee}\vec{\mathbf{v}}$. The version of $\vec{\mathbf{v}}$ ***rotated*** by angle α is denoted $\vec{\mathbf{v}}^{\circ}\alpha$. Shortcut $\vec{\mathbf{v}}^{\llcorner}$ stands for $\vec{\mathbf{v}}^{\circ}\frac{\pi}{2}$. The ***dot product*** of $\vec{\mathbf{u}}$ and $\vec{\mathbf{v}}$ is denoted $\vec{\mathbf{u}}\cdot\vec{\mathbf{v}}$.

The ***labelled edge*** (closed line segment) from $\mathbf{a}$ to $\mathbf{b}$ is written $\mathbf{ab}$. $|\mathbf{ab}|$ denotes the length of $\mathbf{ab}$, and hence also the distance between $\mathbf{a}$ and $\mathbf{b}$. The ***labelled triangle*** with vertices $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$ is written $\mathbf{abc}$. $\mathbf{abc} \sim \mathbf{def}$ indicates that these triangles are ***similar*** (i.e., related by a similarity). Other ***shapes*** (disk, beam) are in bold caps ($\mathbf{R}$ or $\mathbf{T_{i,j}}$).

Transformations are in curly caps ($\mathcal{S}$ or $\mathcal{M}$). ***Composition*** of $\mathcal{U}$ and $\mathcal{V}$ is written $\mathcal{U}\cdot\mathcal{V}$. Application of $\mathcal{U}$ to point $\mathbf{p}$ (resp. shape X) is written $\mathcal{U}\cdot\mathbf{p}$ (resp. $\mathcal{U}\cdot$X). $\mathcal{U}\cdot\mathcal{V}\cdot\mathcal{W}\cdot\mathbf{p}$ means $\mathcal{U}\cdot(\mathcal{V}\cdot(\mathcal{W}\cdot\mathbf{p}))$. $\mathcal{U}^t$ is the ***power*** of $\mathcal{U}$, $\mathcal{U}^{-1}$ is its inverse, and $\mathcal{U}^0$ is the identity.

Complex numbers are in bold italics and may represent points, vectors, or similarities. Let $\boldsymbol{p}$ and $\boldsymbol{q}$ represent points $\mathbf{p}$ and $\mathbf{q}$, $\boldsymbol{u}$ and $\boldsymbol{w}$ represent vectors $\vec{\mathbf{u}}$ and $\vec{\mathbf{w}}$, and $\boldsymbol{s}$ represent similarity $\mathcal{S}$. Vector $\vec{\mathbf{u}} = \langle\lambda_{\mathrm{u}}\cos\alpha_{\mathrm{u}}, \lambda_{\mathrm{u}}\sin\alpha_{\mathrm{u}}\rangle$, where $\lambda_{\mathrm{u}} = |\vec{\mathbf{u}}|$ and α_{u} is the angle from $\langle 0, 1\rangle$ to $\vec{\mathbf{u}}$, has two ***complex forms***: $\boldsymbol{u} = x_{\mathrm{u}} + \boldsymbol{i}y_{\mathrm{u}}$ and $\boldsymbol{u} = \lambda_{\mathrm{u}}e^{\boldsymbol{i}\alpha_{\mathrm{u}}}$. The complex form of a similarity, $\mathcal{S}$, that preserves the origin and takes vector $\langle 1, 0\rangle$ to vector $\langle\lambda\cos\alpha, \lambda\sin\alpha\rangle$ is $\boldsymbol{s} = \lambda e^{\boldsymbol{i}\alpha}$. The complex form of $\mathcal{S}\cdot\vec{\mathbf{u}}$ is the product $\boldsymbol{su}$. The complex form of a similarity that takes $\vec{\mathbf{w}}$ to $\vec{\mathbf{u}}$ is $\boldsymbol{u}/\boldsymbol{w}$.

Similarity

Any combination of primitive affinities:

Translation by vector v: $\mathcal{T}_{\mathbf{v}}\cdot\mathbf{p} = \mathbf{p}+\mathbf{v}$

Rotation by angle α around point f: $\mathcal{R}_{\alpha,\mathbf{f}}\cdot\mathbf{p} = \mathbf{f} + \mathbf{fp}^{\circ}\alpha$

Dilation by ratio λ around point f: $\mathcal{D}_{\lambda,\mathbf{f}}\cdot\mathbf{p} = \mathbf{f} + \lambda\mathbf{fp}$

Preserve lines & ratios (affinity) and angles (conformal)

Shapes X and Y are similar (denoted X ∼ Y)

iff there exists a similarity $\mathcal{S}$ such that Y = $\mathcal{S}$ · X.

A similarities may be defined by an edge

Let $\mathcal{S}_{\mathbf{a},\mathbf{c}}$ be a similarity taking $(0, 0)$ to point $\mathbf{a}$ and $(1, 0)$ to point $\mathbf{c}$.

If $\mathbf{p} = (x, y)$, then $\mathcal{S}_{\mathbf{a},\mathbf{c}}\cdot\mathbf{p} = \mathbf{a} + x\overrightarrow{\mathbf{ac}} + y\overrightarrow{\mathbf{ac}}^{\llcorner}$.

Canonical decomposition of true similarity

For any true similarity $\mathcal{S}$ (not pure translation)

There exist ratio λ, angle α, and fixed point f such that

$\mathcal{S} = \mathcal{D}_{\lambda,f} \cdot \mathcal{R}_{\alpha,f} = \mathcal{R}_{\alpha,f} \cdot \mathcal{D}_{\lambda,f}$

$\mathcal{S}$ is represented by its canonical parameters < f , λ , α >

Its effect on a point is evaluated as $\mathcal{S}{\cdot}\mathbf{p} = \mathbf{f} + \lambda \mathbf{fp}°\alpha$

SAS (Steadily-Animated Similarity) *S*(*t*)

Time-parameterized similarity, st: $\exists \mathcal{S}, \forall t, \mathcal{S}(t) = \mathcal{S}^t$

When $\mathcal{S} = <\mathrm{f}, \lambda, \alpha>$, $\mathcal{S}^t = <\mathrm{f}, \lambda^t, t\alpha>$, and therefore

$$\mathcal{S}^t \cdot \mathbf{p} = \mathbf{f} + \lambda^t \overrightarrow{\mathbf{fp}}°(t\alpha).$$

Row and Matrix Patterns

Pattern = Periodic arrangement of instances of a template shape

Row: One-pattern

Matrix: Two pattern (row of rows)

Similarity-Steady Row

In this talk, "steady" will stand for "*similarity*-steady"

Set $\{\mathrm{X_m}\}$ of shapes is a ***steady row*** iff there exist a ***similarity***, $\mathcal{S}$, and a template, X, such that, for each valid m, $\mathrm{X_m} = \mathcal{S}^{\mathrm{m}} \cdot \mathrm{X}$.

A ***translation row*** is a steady row for which $\mathcal{S}$ is a translation.

Steady Matrix

Definition 2.5. The set $\{X_{m,n}\}$ of $\underline{m}\times\underline{n}$ shapes is a ***steady matrix*** iff there exist similarities, $\mathcal{I}$ and $\mathcal{J}$, and a template, X, such that, for each valid (m,n) pair, $X_{m,n} = \mathcal{J}^{n} \cdot \mathcal{I}^{m} \cdot X$.

Note that we underscore ($\underline{m}$ or $\underline{n}$) the matrix "dimensions" (i.e., the total numbers of elements in a row along a direction).

Given a matrix $\{X_{m,n}\}$, let $X_{r,*}$ (resp. $X_{*,r}$) denote the ***n-row*** (resp. ***m-row***) of its elements for which m=r (resp. n=r).

Property of steady matrix

THEOREM 2.6. *In a steady matrix $\{X_{m,n}\}$, for each valid r, row $X_{r,*}$ and row $X_{*,r}$ are each steady. (Proof in Appendix A.1.)*

PROOF. $X_{r,n} = \mathcal{J}^{n} \cdot (\mathcal{I}^{r} \cdot X)$ implies that row $X_{r,*}$ is steady and has $\mathcal{I}^{r} \cdot X$ for template. $X_{m,r} = (\mathcal{J}^{r} \cdot \mathcal{I}^{m} \cdot \mathcal{J}^{-r}) \cdot (\mathcal{J}^{r} \cdot X)$ can be written as $X_{m,r} = \mathcal{S}^{m} \cdot (\mathcal{J}^{r} \cdot X)$ with $\mathcal{S}^{m} = \mathcal{J}^{r} \cdot \mathcal{I}^{m} \cdot \mathcal{J}^{-r}$, and thus $\mathcal{S}^{m} = (\mathcal{J}^{r} \cdot \mathcal{I} \cdot \mathcal{J}^{-r})^{m}$ with $\mathcal{S} = (\mathcal{J}^{r} \cdot \mathcal{I} \cdot \mathcal{J}^{-r})$. Hence, row $X_{*,r}$ is also steady. □

Tran-Similar (TS) matrix

Definition 2.7. A ***Tran-Similar matrix*** (abbreviated ***TS matrix***) is a steady matrix for which $\mathcal{I}$ and $\mathcal{J}$ commute.

Definition 2.8. A ***translation matrix*** is a steady matrix for which $\mathcal{I}$ and $\mathcal{J}$ are translations.

Definition 2.9. A ***regular matrix*** is a translation matrix for which $\mathcal{I}$ translates by $\langle 1, 0\rangle$ and $\mathcal{J}$ translates by $\langle 0, 1\rangle$.

Bilinear map is not TS

Consider, the bilinear map: $\mathcal{B}(x, y) = \mathcal{I}(\mathcal{I}(\mathbf{a}, y, \mathbf{b}), x, \mathcal{I}(\mathbf{d}, y, \mathbf{c}))$, where $\mathcal{I}(\mathbf{a}, y, \mathbf{b})$ is the LERP $\mathbf{a} + y\,\overrightarrow{\mathbf{ab}}$. Its drawbacks are that the tiles may have drastically different distortions (Fig. 5a) and that its range, **R**, may fold (Fig. 5c). COTS of course does not suffer from these defects (Fig. 5b and 5d).

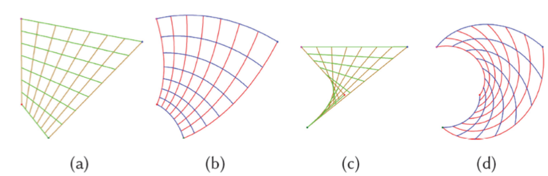


Fig. 5. Bilinear (a) and COTS map (b) for the same quadruplet. The bilinear map may fold (c). The COTS map does not (d).

Möbius map (1)

The ***Möbius map*** is defined (using complex numbers) by $\mathcal{M}(z) = (\boldsymbol{pz} + \boldsymbol{q})/(\boldsymbol{rz} + \boldsymbol{s})$, with $\boldsymbol{ps} \neq \boldsymbol{qr}$. It is popular because of its beautiful mathematical properties: It preserves angle (between directions from the same point) and it preserves clines (curves that are either a circle or a line). But it is not TS. Furthermore, it is not CO and has a DoD of only 2. Indeed, given corners **a**, **b**, and **c**, corner **d** is constrained to a particular point along the circumcircle of **a**, **b**, and **c**. In Fig 6, we compare it to a COTS map with the same corners and illustrate the additional distortion DoFs of the COTS map.

Möbius map (2)

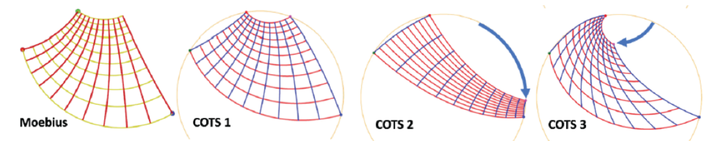


Fig. 6. Möbius is controlled by three corners: $\mathbf{a}_\mathrm{m}$ (red), $\mathbf{b}_\mathrm{m}$ (green), and $\mathbf{d}_\mathrm{m}$ (blue). Its fourth corner, $\mathbf{c}_\mathrm{m}$, lies on their circumcircle, C. Three COTS maps with corners $\mathbf{a} = \mathbf{a}_\mathrm{m}$, $\mathbf{b} = \mathbf{b}_\mathrm{m}$, $\mathbf{c}$, and $\mathbf{d} = \mathbf{d}_\mathrm{m}$: COTS 1 has $\mathbf{c} = \mathbf{c}_\mathrm{m}$ and resembles Möbius, but has log-spiral isocurves. COTS 2 with **c** at a different point on circle C. COTS 3 with corner **c** no longer on circle C.

Restricted FPI: $\mathcal{A} \cdot \mathcal{M}$, of an affinity with a Möbius map.

Four-Point Interpolant (FPI) [Lipman et al. 2012], "FPI has a constant conformal distortion, lower than the maximal conformal distortion of the other maps"

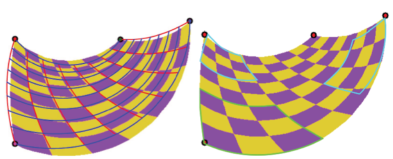


Fig. 7. The isocurves (red and blue) of a COTS map drawn over the tiling of a restricted FPI map (from [Lipman et al. 2012]) with the same control quadruplet (left). To show that FPI is not TS, we draw (right) the border (cyan) of the lower-left 3×3 grid of its tiles and two of its copies transformed by similarities to best align them with other 3×3 grids of tiles.

Decomposition of a COTS map

Theorem 3.2. *A COTS map is the composition $\mathcal{S}_{\mathbf{f},\mathbf{a}} \cdot \mathcal{P} \cdot \mathcal{L}$ of a linear transformation, $\mathcal{L}$, with a log-polar map, $\mathcal{P}$, and with a similarity, $\mathcal{S}_{\mathbf{f},\mathbf{a}}$ (Sec. 2.2). (Proof in Appendix A.5.)*

Proof. $\mathcal{L}$ maps (x, y) onto (λ, α) with $\lambda = x \ln \lambda_u + y \ln \lambda_v$ and $\alpha = x\alpha_u + y\alpha_v$. $\mathcal{P}$ maps (λ, α) onto complex number $z = e^{\lambda + i\alpha}$. Finally, $\mathcal{S}_{\mathbf{f},\mathbf{a}}$ maps $(0, 0)$ to $\mathbf{f}$ and $(1, 0)$ onto $\mathbf{a}$. Hence, their composition, $\mathcal{S}_{\mathbf{f},\mathbf{a}} \cdot \mathcal{P} \cdot \mathcal{L}$, maps (x, y) onto $\mathcal{M}(x, y) = \mathbf{f} + e^{\lambda}\overrightarrow{\mathbf{fa}}^{\circlearrowleft}(\alpha) = \mathbf{f} + \lambda_u^x \lambda_v^y \overrightarrow{\mathbf{fa}}^{\circlearrowleft}(x\alpha_u + y\alpha_v)$ (matching Eq. 12). □

Complex form of a proper similarity

Let $\mathcal{U} = < \mathbf{f}, \lambda, \alpha >$ be a proper similarity that combines a rotation by α and a dilation by λ about the same fixed point $\mathbf{f}$. We represent it in complex form by the tuple $< f, \boldsymbol{u} >$, where $\boldsymbol{u} = \lambda e^{i\alpha}$. The complex form of $\mathcal{U} \cdot \mathbf{p}$ is

$$\mathcal{U} \cdot \mathbf{p} = f + \boldsymbol{u}(p - f). \quad (3)$$

Complex form of a similarity between edges

Let SIM(**a**, **b**, **d**, **c**) be a proper similarity that maps **ab** to **dc**.

THEOREM 4.1. *The complex form of* SIM(**a**, **b**, **d**, **c**) *is*

$$< \frac{\boldsymbol{ac} - \boldsymbol{bd}}{\boldsymbol{a} - \boldsymbol{b} + \boldsymbol{c} - \boldsymbol{d}}, \frac{\boldsymbol{c} - \boldsymbol{d}}{\boldsymbol{b} - \boldsymbol{a}} > .$$

The proof in Appendix A.2.

PROOF. Use Equation 3 to produce the complex forms of the two constraints that define $\mathcal{U}$:
(1) Constraint $\mathcal{U} \cdot \mathbf{a} = \mathbf{d}$ implies

$$d - f = u(a - f). \quad (18)$$

(2) Constraint $\mathcal{U} \cdot \mathbf{b} = \mathbf{c}$ implies

$$c - f = u(b - f). \quad (19)$$

Dividing Eq. 18 by Eq. 19 eliminates u:

$$\frac{d - f}{c - f} = \frac{a - f}{b - f}, \quad (20)$$

which, after simplification, yields

$$f = \frac{ac - bd}{a - b + c - d}, \quad (21)$$

Substituting Eq. 21 into Eq. 18 and simplifying yields

$$u = \frac{c - d}{b - a}. \quad (22)$$

□

Computing true similarity between two edges

Let $[a_i, b_i]$ be the edge with start & end vertices a_i & b_i

If $< f, \lambda, \alpha > = \mathrm{Sim}(a_0, b_0, a_1, b_1)$ maps $[a_0, b_0]$ to $[a_1, b_1]$,
then: $\lambda = |a_1b_1| \,/\, |a_0b_0|$,

$\alpha = a_0b_0 \wedge a_1b_1$, and

$f = a_0 + < w \bullet a_1a_0 , w^{\circ} \bullet a_1a_0 > / w^2$ where

$w = \lambda < \cos\alpha - 1, \ \sin\alpha >$, and $w^{\circ} = w^{\circ}(\pi/2)$

$\mathrm{Sim}(a_0, b_0, a_1, b_1)$ is a true similarity when $a_0b_0 \neq a_1b_1$

CO (Corner-Operated) map: Definition

A planar map, $\mathcal{M}$, takes point $\mathbf{p} = (x, y)$ to a point, which we write as $\mathcal{M}(x, y)$, $\mathcal{M}(\mathbf{p})$, or $\mathcal{M} \cdot \mathbf{p}$, whichever most clearly conveys the concept being presented. Note that, for readability, we denote the input parameters using (x, y), rather than the more traditional (u, v).

Definition 4.2. A map, $\mathcal{M}$, is ***Corner-Operated*** (***CO***) if it satisfies the following four constraints:

$$\mathcal{M}(0,0) = \mathbf{a}, \mathcal{M}(0,1) = \mathbf{b}, \mathcal{M}(1,1) = \mathbf{c}, \mathcal{M}(1,0) = \mathbf{d}, \quad (5)$$

which guarantee that $\mathcal{M}$ is point-interpolating, i.e., that it maps the four corners of the unit-square, parametric domain, **D**, onto given ***quadruplet*** $\{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$ of ***control-corners***.

TS (Tran-Similar) map: Definition

Definition 4.3. A map, $\mathcal{M}$, is ***Tran-Similar*** (***TS***), iff, for every vector $\vec{v}$, there exists a similarity $\mathcal{S}$, such that, for any point $\mathbf{p}$ and for any scalar t:

$$\mathcal{M}(\mathcal{T}_{\vec{v}}^{\,t} \cdot \mathbf{p}) = \mathcal{S}^t \cdot \mathcal{M}(\mathbf{p}). \tag{6}$$

Recall that $\mathcal{T}_{\vec{v}}$ denotes the translation by vector $\vec{v}$.

COTS (Corner-Operated Tran-Similar) map: Definition

Definition 4.4. The CO map, $\mathcal{M}$, defined by the ordered quadruplet of control-corners, $\{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$ is a ***COTS map*** if

$$\mathcal{M}(x, y) = \mathcal{V}^y \cdot \mathcal{U}^x \cdot \mathbf{a}, \tag{7}$$

where $\mathcal{U} = \mathrm{SIM}(\mathbf{a}, \mathbf{b}, \mathbf{d}, \mathbf{c})$ and $\mathcal{V} = \mathrm{SIM}(\mathbf{a}, \mathbf{d}, \mathbf{b}, \mathbf{c})$.

Note that $\mathcal{U}$ maps **ab** to **dc** and $\mathcal{V}$ maps **ad** to **bc**, and that COTS moves corner **a** by $\mathcal{U}^x$ and then by $\mathcal{V}^y$.

Remarkable property!

Theorem 4.5. $\mathrm{SIM}(\mathbf{a}, \mathbf{b}, \mathbf{d}, \mathbf{c})$ *and* $\mathrm{SIM}(\mathbf{a}, \mathbf{d}, \mathbf{b}, \mathbf{c})$ *have the same fixed point (Fig. 11a and b).*

Proof. Swapping **b** and **d** in expression 4 does not change **f**. □

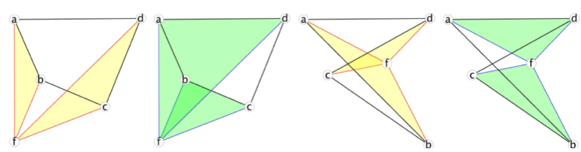


Fig. 11. Given the four corners (**a**, **b**, **c**, **d**) of a non-parallelogram quad. Its Four-Point Similarity-Center, **f**, satisfies **fab** ∼ **fdc** (a) and **fad** ∼ **fbc** (b). This property holds even when the quad is self-crossing (c and d).

Four-Point Similarity-Center (FPSC)

Definition 4.6. The ***Four-Point Similarity-Center*** (***FPSC***) of the quadruplet { **a**, **b**, **c**, **d** } is the common fixed point of $\mathrm{SIM}(\mathbf{a}, \mathbf{b}, \mathbf{d}, \mathbf{c})$ and of $\mathrm{SIM}(\mathbf{a}, \mathbf{d}, \mathbf{b}, \mathbf{c})$.

Let $\mathbf{F}(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d})$ denote the FPSC of this quadruplet. Th. 4.5 holds even when the quad is self-crossing (Fig. 11c and d). Using Expression 4, one can verify that **f** is preserved by a cyclic permutation ($\mathbf{F}(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}) = \mathbf{F}(\mathbf{b}, \mathbf{c}, \mathbf{d}, \mathbf{a})$), and by swapping diagonally-opposite corners ($\mathbf{F}(\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}) = \mathbf{F}(\mathbf{a}, \mathbf{d}, \mathbf{c}, \mathbf{b})$). But, swapping two consecutive corners will, in general, not preserve **f**.

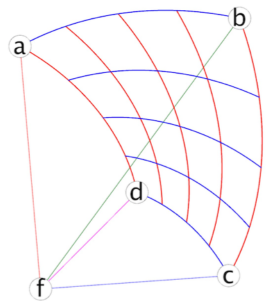
Similarity center f of quad [a,b,c,d]

Let $\mathcal{U}$ = Sim(a,d,b,c) = < f_u, λ_u, α_u >

and $\mathcal{V}$ = Sim(a,b,d,c) = < f_v, λ_v, α_v >

By the "quad similarities" theorem,

$f = f_u = f_v$



Computing SIM(a,b,c,d)

Consider a quadruplet $\{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$ for which $\overrightarrow{\mathbf{ab}} \neq \overrightarrow{\mathbf{dc}}$. Let $\mathcal{U}$ be the similarity SIM($\mathbf{a}, \mathbf{b}, \mathbf{d}, \mathbf{c}$) that takes oriented edge $\mathbf{ab}$ to oriented edge $\mathbf{dc}$. We can compute its canonical representation, $\langle \mathbf{f}, \lambda, \alpha \rangle$ (Sec. 2.2), using Expression 4 or:

$$\lambda = \frac{|\mathbf{dc}|}{|\mathbf{ab}|},\ \alpha = \overrightarrow{\mathbf{ab}}^{\vee}\overrightarrow{\mathbf{dc}},\ \mathbf{f} = \mathbf{a} + \frac{\langle \vec{\mathbf{w}}\cdot\overrightarrow{\mathbf{da}}, \vec{\mathbf{w}}^{\perp}\cdot\overrightarrow{\mathbf{da}} \rangle}{d}, \qquad (8)$$

with $\vec{\mathbf{w}} = \langle \lambda \cos\alpha - 1, \lambda \sin\alpha \rangle$ and $d = \vec{\mathbf{w}}^2$.

The determinant d of this system is $\vec{\mathbf{w}}^2 = (\lambda\cos\alpha - 1)^2 + (\lambda \sin\alpha)^2$. It is null when both $\alpha = 0$ and $\lambda = 1$, hence, when $\overrightarrow{\mathbf{ab}} = \overrightarrow{\mathbf{dc}}$.

A COTS map is TS

It is easy to show that, if $\mathcal{U} = \langle f, \boldsymbol{u} \rangle$ and $\mathcal{V} = \langle f, \boldsymbol{v} \rangle$, then the complex form of $\mathcal{M}$ can be written:

$$\mathcal{M}(x, y) = f + \boldsymbol{u}^x \boldsymbol{v}^y (\boldsymbol{a} - f), \qquad (9)$$

Lemma 4.7. *If $\mathcal{U}$ = SIM($\mathbf{a}, \mathbf{b}, \mathbf{d}, \mathbf{c}$) and $\mathcal{V}$ = SIM($\mathbf{a}, \mathbf{d}, \mathbf{b}, \mathbf{c}$) then $\mathcal{U}$ and $\mathcal{V}$ commute, i.e., $\mathcal{U} \cdot \mathcal{V} = \mathcal{V} \cdot \mathcal{U}$*

Theorem 4.8. *A COTS map, $\mathcal{M}$, is Tran-Similar.*

Proof. Let us show that $\mathcal{M}$ satisfies Eq. 6 in Def. 4.3. Let $\langle x', y' \rangle$ denote the Cartesian components of $\vec{\mathrm{v}}$. From Eq. 9, the complex form of $\mathcal{M}(x + tx', y + ty')$ is

$$f + \boldsymbol{u}^{x+tx'} \boldsymbol{v}^{y+ty'} (\boldsymbol{a} - f), \qquad (23)$$

which yields

$$f + (\boldsymbol{u}^{x'} \boldsymbol{v}^{y'})^t (\boldsymbol{u}^x \boldsymbol{v}^y (\boldsymbol{a} - f)), \qquad (24)$$

and therefore

$$f + (\boldsymbol{u}^{x'} \boldsymbol{v}^{y'})^t ((f + \boldsymbol{u}^x \boldsymbol{v}^y (\boldsymbol{a} - f)) - f), \qquad (25)$$

which is

$$f + (\boldsymbol{u}^{x'} \boldsymbol{v}^{y'})^t (\mathcal{M}(x, y) - f). \qquad (26)$$

□

Evaluating the COTS map

To compute our representation of a COTS map, given quadruplet $\{\mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}\}$, we compute its fixed point $\mathbf{f}$ (Sec. 4.3) and its two pairs of ratios and angles, as follows:

$$\lambda_v = \frac{|\mathbf{dc}|}{|\mathbf{ab}|} \text{ and } \lambda_u = \frac{|\mathbf{bc}|}{|\mathbf{ad}|}, \qquad (10)$$

$$\alpha_v = \overrightarrow{\mathbf{ab}}^{\vee}\overrightarrow{\mathbf{dc}} + 2\pi \mathrm{k_v} \text{ and } \alpha_u = \overrightarrow{\mathbf{ad}}^{\vee}\overrightarrow{\mathbf{bc}} + 2\pi \mathrm{k_u} \qquad (11)$$

where $\mathrm{k_u}$ and $\mathrm{k_v}$ are updated at each frame during interactive editing or animation to select the ***branch*** that minimizes changes in the COTS map in α_u and α_v. A COTS map is represented by points $\mathbf{a}$ and $\mathbf{f}$, and by the four parameters: λ_u, λ_v, α_u, and α_v.

Applying a COTS map

To compute the image $\mathcal{M}(\mathbf{p})$ of point $\mathbf{p} = (x, y)$, we use

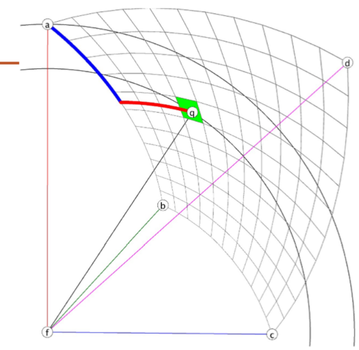
$$\mathcal{M}(x,y) = \mathbf{f} + \lambda_u^x \lambda_v^y \overrightarrow{\mathbf{fa}}^{\circ}(x\alpha_u + y\alpha_v). \tag{12}$$

COTS map (intuition)

Definition: m(*u*,*v*) = $S_{u,v}$ · a

Point map: m(p) = $S_{u,v}$ · a

with **p** = (*u*,*v*)

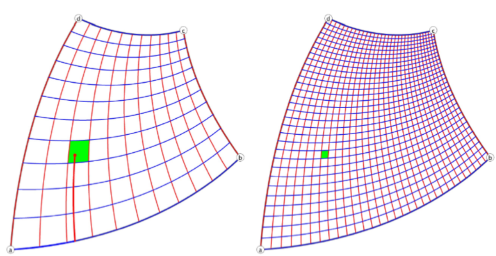


Evaluation: m(*u*,*v*) = $\mathcal{V}^v$ · ($\mathcal{U}^u$ ·a) = f + $\lambda_u^u \lambda_v^v$ fa°($u\alpha_u + v\alpha_v$)

Constant-cost query of tile containing q

Regardless of repetition count n

- (*u*,*v*)=m^{-1}(q)
- i= ⌊*u*n⌋, j= ⌊*v*n⌋



Applying the inverse of a COTS map (1)

THEOREM 4.9. *Given* $\mathbf{q} = \mathcal{M}(x, y)$*, parameters* x *and* y *are the solutions of the following linear system (proof in A.4):*

$$x \ln \lambda_u + y \ln \lambda_v = \ln \lambda \tag{13}$$
$$x\alpha_u + y\alpha_v = \alpha \tag{14}$$

However, in configurations where $k_u \neq 0$ or $k_v \neq 0$ we must consider all branching options and hence, solve the above system with λ_u and λ_v defined by Eq. 10, α_u and α_v defined by Eq. 11,

$$\lambda = \frac{|\mathbf{fq}|}{|\mathbf{fa}|}, \text{ and } \alpha = \overrightarrow{\mathbf{fa}}^{\vee}\overrightarrow{\mathbf{fq}} + 2\pi k, \tag{15}$$

where integer k identifies a ***branching option***. We explain below how we generate and test a sufficient set of these options.

Applying the inverse of a COTS map (2)

The determinant of this system is $d = \alpha_v \log \lambda_u - \alpha_u \log \lambda_v$.

When $d = 0$, there is no solution. This may happen when the quad is a parallelogram, or a point, in which case, the inverse is not defined.

Otherwise, we find the limits $\alpha_{min} = min(0, \alpha_u, \alpha_v, \alpha_u + \alpha_v)$ and $\alpha_{max} = max(0, \alpha_u, \alpha_v, \alpha_u + \alpha_v)$ and, for each candidate value of k for which $\alpha_{min} \leq \alpha \leq \alpha_{max}$, we compute coordinates

$$x = (\alpha_v \log \lambda - \alpha \log \lambda_v)/d \tag{16}$$

$$y = (\alpha - x\alpha_u)/\alpha_v. \tag{17}$$

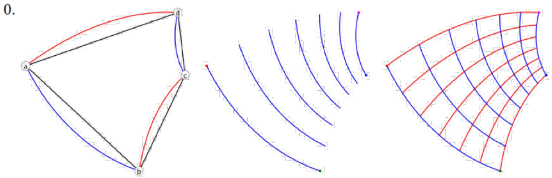
The ***valid candidates*** are those for which x and y are both in [0,1].

Iso-curves of a COTS map

A ***u-curve*** is an isocurve of points $\mathcal{M}(*, y)$, for which x varies, but not y. Similarly, the term ***v-curve*** refers to $\mathcal{M}(x, *)$.

Property 1. *A Tran-Similar map takes lines to log-spirals and constant velocity motions to Steadily-Animated Similarities.*

Consequently, isocurves, and hence borders of **R** are log-spirals (Fig. 12). The borders never cross in simple branching configurations (for which $k_u \neq 0$ and $k_v \neq 0$.



COTS matrix of shapes

Definition 5.1. A ***COTS matrix*** is a Corner-Operated TS matrix, i.e., a Tran-Similar matrix (Def. 2.8) of which the layout is controlled by the four corners of the range **R** of the associated COTS map.

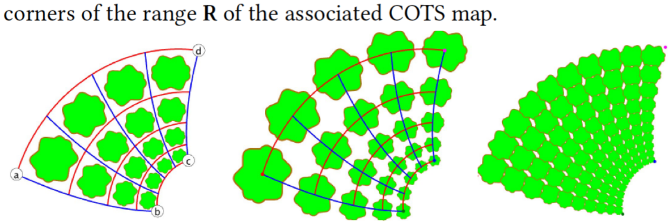
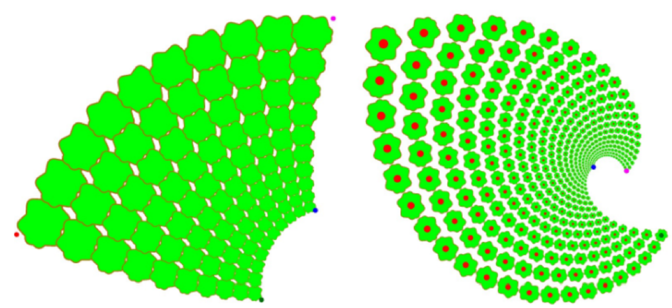


Fig. 13. A 4×4 COTS matrix of tiles can be used to define a 4×4 (a) or a 5×5 (b) COTS matrix of shapes. Adjacent instances may overlap (c).

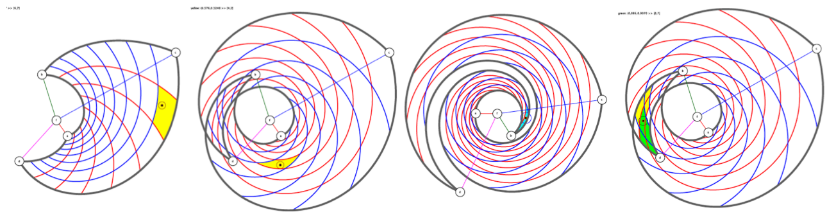
Gaps and overlaps are steady patterns



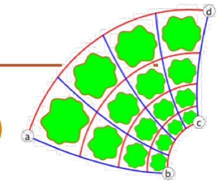
Indices of the tiles that contain a given point

The COTS map has a
constant-cost inversion

Fig. 14. Inversion: We fill the tile that contains the query point **q** indicated by a dot in a red circle. We paint it yellow (a, b, and d), when k=0. We paint it cyan (c), when k=1. We paint it green (d), when k=−1. Note that, when **R** self-overlaps (b and d), point **q** may be contained in more than one tile (d).



Point-in-COTS-Pattern test



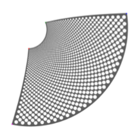
Constant cost to classify query point q

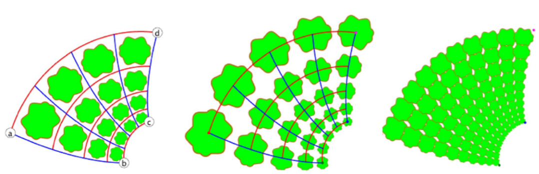
- 9 *types of tiles (4 corner, 4 border, and inner)*
- Pick a reference tile for each type
- For each reference tile, record list of shapes that intersect it
- For each tile (i,j) containing q:
  - Compute the corresponding point, **q'**, in the corresponding reference tile
  - Classify **q'** against the shapes that intersect that reference tile

instances that intersect it. In our representation, we identify each instance by its two index-offsets. So, for example, if the reference tile of a given set is $\{\mathbf{T}_{i,j}\}$, and if it intersects instance $\{\mathbf{X}_{i,j}\}$, we store the index-offset pair (0,0). If it also intersects instance $\{\mathbf{X}_{i+1,j}\}$, we also store index-offset pair (1,0). the representative tile $\mathbf{T}_{i',j'}$ of that set. Then, we compute the point $\mathbf{q}'$ that is to $\mathbf{T}_{i',j'}$ what **q** is to $\mathbf{T}_{i,j}$. (We obtain $\mathbf{q}'$ from **q** by (1) computing the preimage of **q**, (2) translating it by $\langle (i'-i)/\underline{n}, (j'-j)/\underline{n} \rangle$, and (3) applying the COTS map to the result.) Finally, we classify $\mathbf{q}'$ against the list of shape-instances associated with $\mathbf{T}_{i',j'}$.

Total area calculation

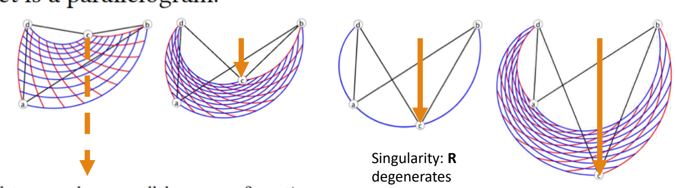
We propose another, constant-cost approach for computing the total area of a COTS matrix of disjoint shapes. Let s denote the area of the template shape, $\mathbf{X}_{0,0}$, t denote the area of the template tile, $\mathbf{T}_{0,0}$, and r denote the total area of **R**. The total area of the COTS matrix is $a = sr/t$. Hence, it may be computed in constant time, independently of the value of $\underline{n}$. This simple closed-form, constant-cost expression





Augmented COTS map

Definition 6.1. The ***Augmented COTS map*** (abbreviated ***ACOTS***) uses a bilinear interpolation (Sec. 3.3) for cases where the control quadruplet is a parallelogram.



When the quadruplet approaches a parallelogram configuration, COTS approaches ACOTS, hence producing a continuous behavior during interactive editing or animation. As demonstrated in Fig. 15,

Corollaries of Tran-Similarity

PROPERTY 2. *If two shapes are related by a translation, their images by a COTS map are related by a similarity.*

PROPERTY 3. *COTS images of rectangular blocks of same repetition-count in each direction in a regular pattern of tiles or of shapes are similar to each other (Fig. 16-left).*

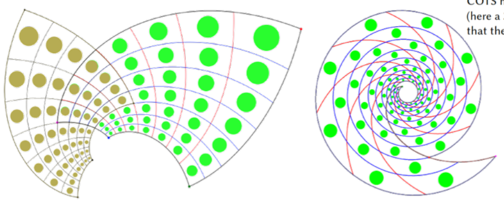
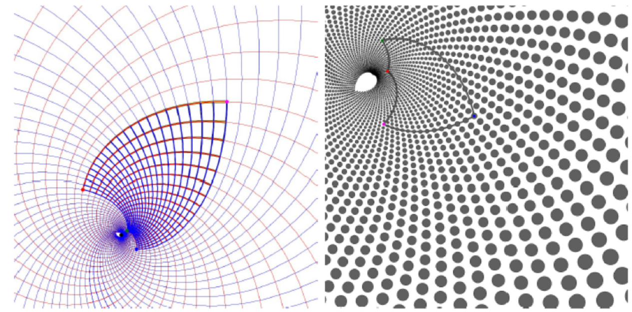


Fig. 16. Tran-Similarity makes it possible to overlap two similar copies of a COTS map so that the tiles and shapes (disks here) in the overlap region (here a 3×4 block) match perfectly (left). It also makes it possible to ensure that the borders of tiles along opposite edges of **R** align perfectly (right).

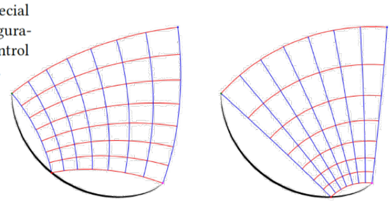
Self-overlapping extended COTS map



Right-Angle COTS map

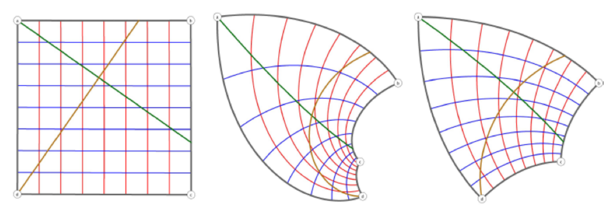
PROPERTY 4. *The opposite angles of each tile of a COTS map are identical.*

By foregoing one degree of freedom, we can constrain the arrangements of control corners so as to ensure that the angles at all four corners of each tile are right angles. This happens when the following constraint is satisfied: $\tan^{-1}(\alpha_u/\ln\lambda_u) - \tan^{-1}(\alpha_v/\ln\lambda_v) = \pi/2$. We use the term ***Right-Angle COTS map*** to identify these special cases. In Fig. 17, we show (as a thick black stroke), for a configuration of the three control corners **b**, **c**, and **d**, the curve where control corner **a** should lie so as to produce a Right-Angle COTS map.



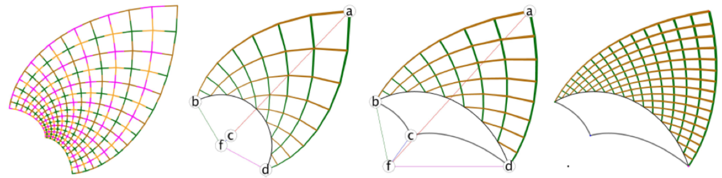
Conformal instances of COTS map

In general, a Right-Angle COTS map is not conformal. For example, in Fig. 18b, we trace the images (green and brown) of two lines orthogonal in parameter space (Fig. 18a). We observe (Fig. 18b) that the two curves do not cross at right angle. They do (Fig. 18c), when the Right-Angle COTS map is symmetric (when **b** and **d** are symmetric with respect to the line through **a** and **c**).



Michell Truss extensions

Fig. 9. The Symmetric Right-Angle COTS lattice (a) appears identical to a pattern of lines of principal action shown in [Hemp 1973] as the analytical solution to the Michell truss. Another Symmetric Right-Angle COTS lattice (b), from which we removed all bars inside the disk of center **f** and radius |**fd**|. A non-symmetric version of it (c) and its refinement (d).



(a) (b) (c) (d)

Morphing between two vectors

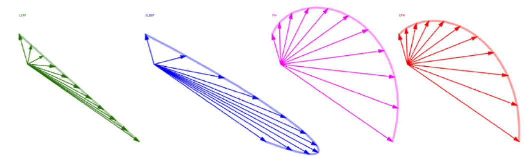


Fig. 19. Interpolations between two vectors: LERP (green), SLERP (blue), PM (magenta), and LPM (red), which is steady and traces a log-spiral,

PM variant of COTS

More voluptuous, but not TS

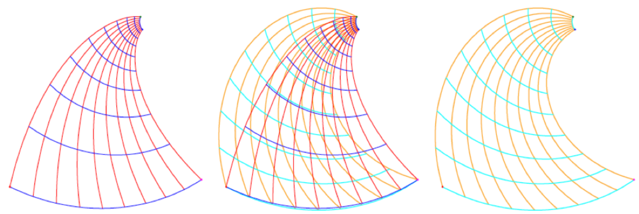


Fig. 20. A COTS map (left), which is based on LPM, and its PM variant (right), which is based on PM and is not TS. Their overlay (center).

Coons variant of a COTS map

We use a Coons patch [Coons 1967] to produce a map that interpolates the four log-spiral border-edges of the range **R** of a COTS map. The Coons construction adds the linear interpolations between pairs of opposite borders and subtract the bilinear interpolation of the corners. Although this variant resembles the COTS map, it is not TS (Fig. 21).

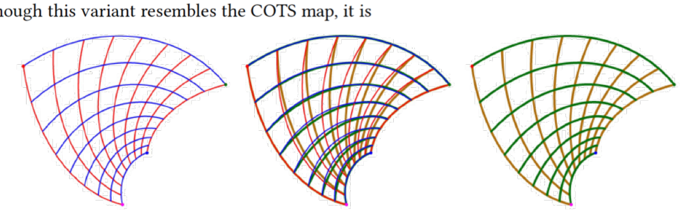


Fig. 21. COTS map (left), Coons patch (right), their overlay (center).

Neville SAS interpolation of 4 edges

$\mathcal{U}$ = SIM(**a**, **b**, **d**, **c**) defines a SAS (Def. 2.2) that morphs **ab** to **dc**.
$\mathcal{V}$ = SIM(**a**, **d**, **b**, **c**) defines a SAS that morphs **ad** to **bc** (Fig. 22). A

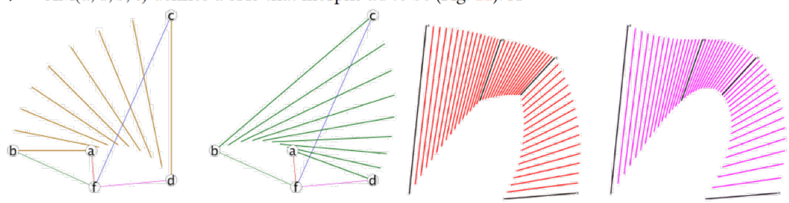
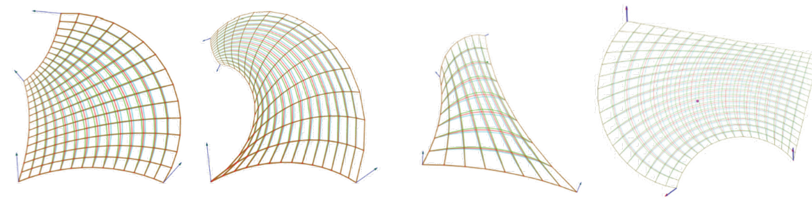


Fig. 22. Two steady rows of ($\underline{n}$+1) edges created using the SAS (Steadily-Animated Similarities) defined by quadruplet {**a**, **b**, **c**, **d**}: Edges $\mathcal{U}^{i/\underline{n}} \cdot \mathbf{ab}$ (a) and Edges $\mathcal{V}^{j/\underline{n}} \cdot \mathbf{ad}$ (b). A strip of SAS (c) that interpolates 4 control edges (black). An interpolating strip that smoothly blends these (d).

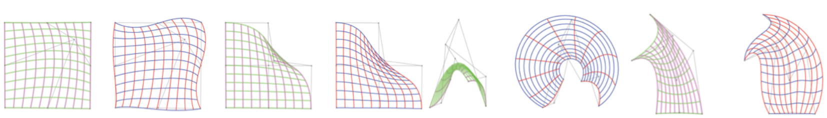
Symmetric Log-Spiral (LSL) map

The ***Symmetric Log-Spiral*** (***SLS***) map is controlled by 8 points, {**a**, **a′**, **b**, **b′**, **c**, **c′**, **d**, **d′**}, which define a quadruplet of labeled edges, {**aa′**, **bb′**, **cc′**, **dd′**} (Fig. 23). To define it, we start with the bilinear interpolation (Sec. 3.3), $\mathcal{I}(\mathcal{I}(\mathbf{a}, x, \mathbf{b}), y, \mathcal{I}(\mathbf{d}, x, \mathbf{c}))$, but replace each LERP by the SAS (Sec. 2.3) between the corresponding edges. For example, we replace $\mathcal{I}(\mathbf{a}, x, \mathbf{b}) = (1 - x)\mathbf{a} + x\mathbf{b}$ by SAS $\mathcal{U}^x \cdot \mathbf{aa'}$ where $\mathcal{U}$ = SIM(**a**, **a′**, **b**, **b′**). To make the solution symmetric, we perform the evaluation twice (swapping the u-first and v-first options) and average the results, using the mid-course edge of a SAS.



Bézier Patch and Curve of COTS

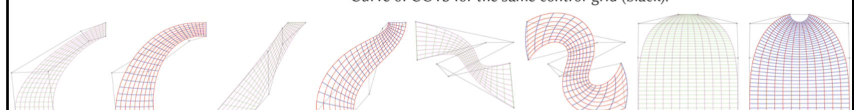
Fig. 24. Quadratic Bézier patch: We show (a,c,e,g) the iso-curves of standard bi-quadratic Bézier patch formulated as a composition of bilinear interpolations and (b,d,g,h), for each of these, the iso-curves of the bi-quadratic Bézier patch of COTS for the same 3 × 3 control-grid (black), which we compute by simply replacing the bilinear interpolation by a COTS map. Notice that the COTS version is more expressive (b,f) and less prompt to fold (d), but may exhibit undesired undulations (h).



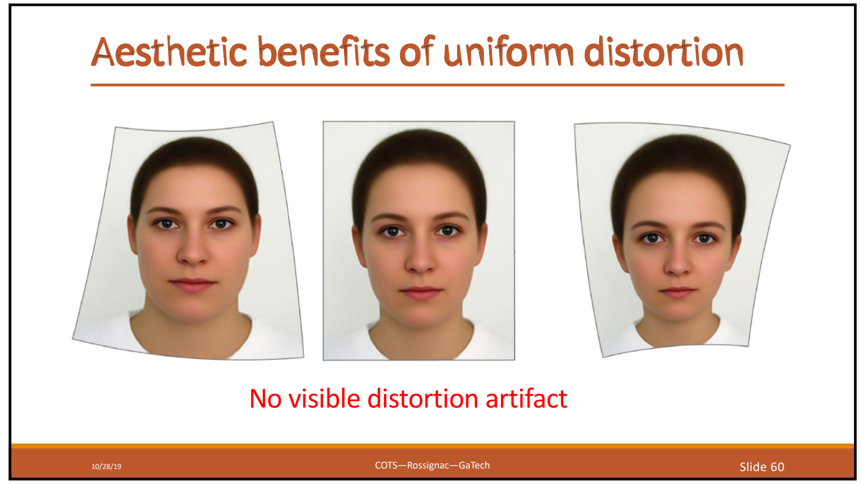
Cubic Bezier Curve of COTS (CBCC)

In Fig. 25, we compare a mixed-degree Bézier patch with a 4 × 2 control-grid to what we call a ***Cubic Bézier Curve of COTS*** (***CBCC***), which we evaluate using a cascade of 3, then 2, then 1 COTS maps. Again, the COTS version affords more expressive power and does not exhibit unexpected undulations. It is easier to control than the bi-quadratic Bézier patch of COTS. It may offer a valuable alternative to the variant proposed in Sec. 7.3.

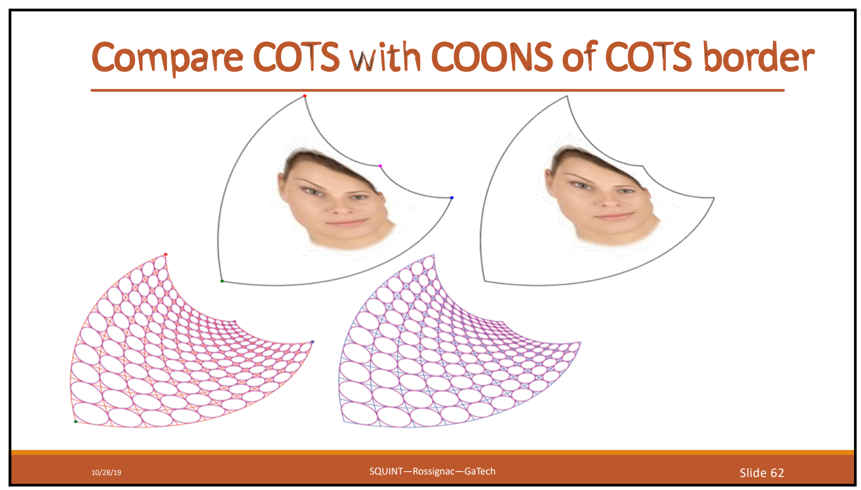
Fig. 25. We show (a,c,e,g) the iso-curves of mixed-degree (4 × 2) Bézier patch and (b,d,g,h), for each of these, the iso-curves of the Cubic Bézier Curve of COTS for the same control-grid (black).




Texture warping
10/28/19
COTS—Rossignac—GaTech
Slide 59


Aesthetic benefits of uniform distortion
No visible distortion artifact
10/28/19
COTS—Rossignac—GaTech
Slide 60


Artistic opportunities?
10/28/19
SQUINT—Rossignac—GaTech
Slide 61


Compare COTS with COONS of COTS border
10/28/19
SQUINT—Rossignac—GaTech
Slide 62

Copy-Warp-And-Paste (CWAP) map

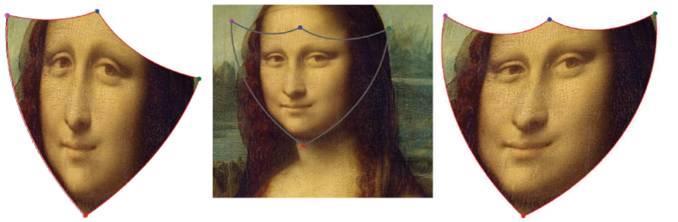
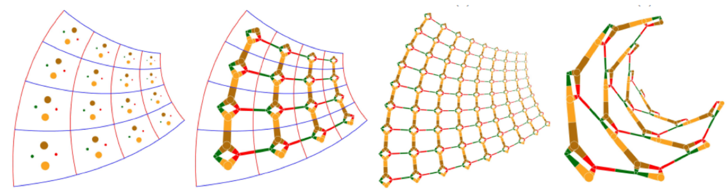


Fig. 26. In a CWAP application, the range $\mathbf{R}_1$ of COTS map $\mathcal{M}_1$ is used (center) to select a portion, $\mathbf{T}$, of an image. COTS map $\mathcal{M}_2$ controls where $\mathbf{T}$ is pasted and how it is distorted (left and right).

COTS lattices

Fig. 27. Clean COTS lattice with a 4×4 matrix of disk-groups (a), 6 matrices of bars, 4 matrices of hubs (b). A 9×9 version (c) and a distorted version (d).



Five-point-operated Planar Steady Lattice (PSL)

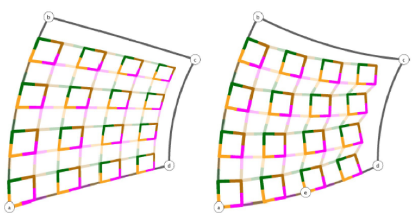
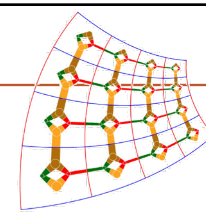


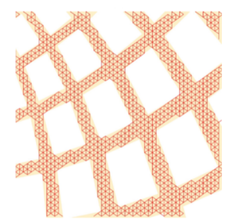
Fig. 28. COTS lattice (left) with $\mathcal{U}$ = SIM(a, b, d, c) and $\mathcal{V}$ = SIM(a, d, b, c), for which all intra-group quads (made of a cycle of bars in solid-colors) and all inter-group quads (made of a cycle of semi-transparent bars) are similar to each other. A PSL (right) with $\mathcal{U}$ = SIM(a, e, e, d)^2 and $\mathcal{V}$ = SIM(a, d, b, c), for which all intra-group quads are similar to each other, but the inter-group quads are not and the m-rows are not steady.

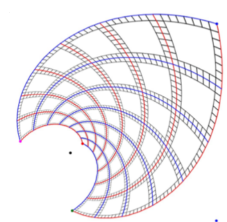
Exceptions



Borders (9 lists)

Filtered lattices





Lattice-in-Lattice (LiL)

3D lattices

Steady row in 3D of planar COTS slices

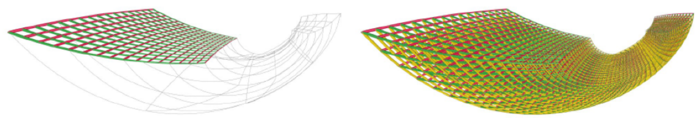
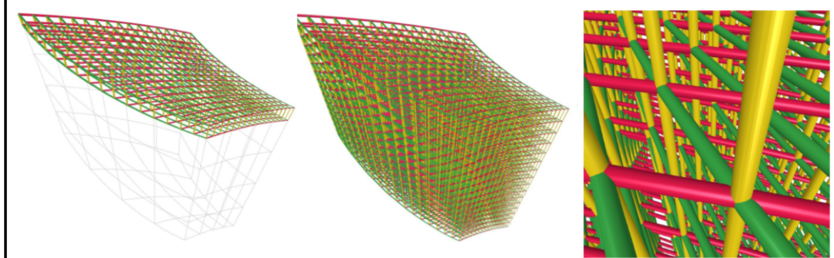


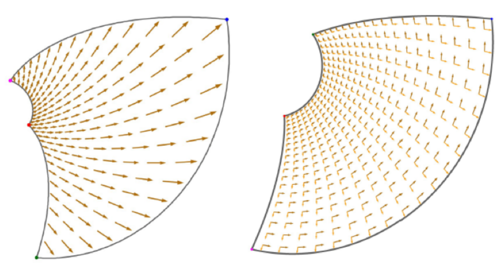
Fig. 10. A COTS Lattice-Slab (left) is used to make a Lattice-Brick (right).

Use steadiness to accelerate PMC & mass?

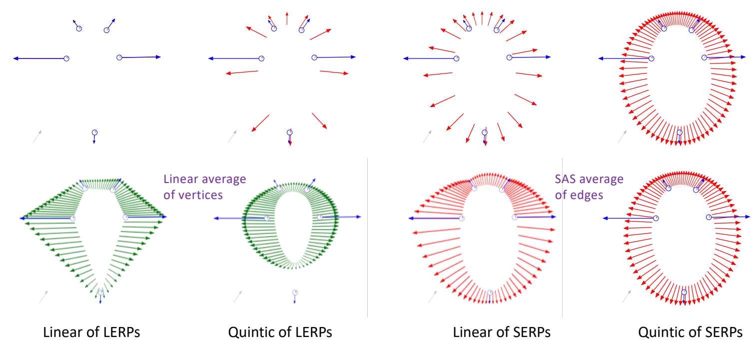


Vector Fields

Steady vector-field defined by a COTS map and a template vector



Subdividing a Ring-of-Arrows (RoA)



Animating a COTS map along a RoA