Geometric Paths

by Sven Nilsen, 2018

In this paper I introduce the concept of geometric paths, used to analyze path-connected spaces. I also introduce the concept of anageometric paths, a generalization of geometric paths.

A geometric path \`f\` is a higher order function constructed by an axiom function \`$f_1$\`:

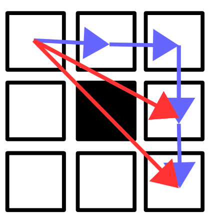
$f_1(a, b)$ <=> continuum(a, b)	Example of an axiom that defines
$f_1(b, c)$ <=> continuum(b, c)	conditional continuum between some points.
$f_1(\_, \_)$ <=> false_1	All other points are disconnected.

$$f := \backslash(a : T, c : T) = \backslash(g : T \rightarrow \mathrm{bool}) = f_1(a, c)(g) \lor \exists\, b : g \{\ f(a, b)(g) \land f(b, c)(g)\ \}$$

$$f : T \times T \rightarrow (T \rightarrow \mathrm{bool}) \rightarrow \mathrm{bool}$$
$$\mathrm{continuum} := \backslash(a : T, b : T) = \backslash(g : T \rightarrow \mathrm{bool}) = g(a) \land g(b)$$

It is only permitted to use \`continuum\` or \`$\mathrm{false}_1$\` for any pair of inputs in \`$f_1$\`.

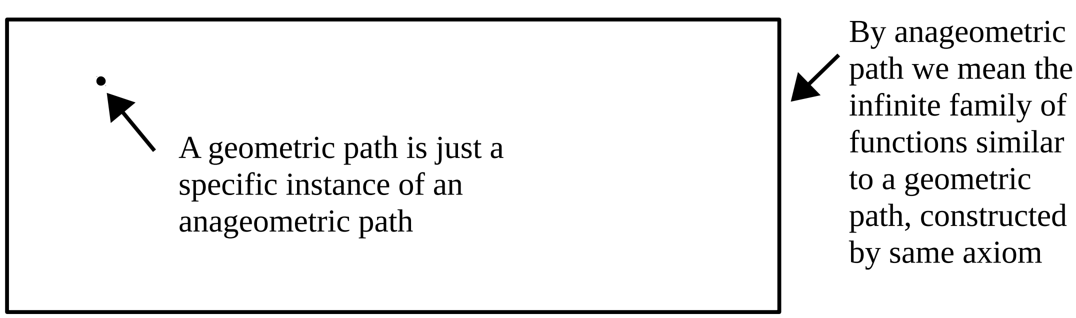
The geometric path can be thought of as connecting a space by composing small arrows:



When there exist a path of blue arrows, then the space is connected by “filling in” with red arrows. The space is undirected, such that all arrows can be flipped around.

A geometric path is a *higher order unlabeled undirected graph* for any function of type \`T → bool\`.

Each geometric path is part of an infinite family of similar functions, called “anageometric path”:



The rest of this paper explains how an anageometric path is constructed.

Assume one has a property `g` that defines some sub-type of 3 variables `a`, `b` and `c`:

$a : [g]\ a'$ $\quad$ $b : [g]\ b'$ $\quad$ $c : [g]\ c'$

$g : T \rightarrow bool$

All possible interpretations of the order "abc" can be given descriptive names:

abc => a'b'c'	
000	external
001	tip
010	bridge
011	tail
100	head
101	wall
110	neg tip
111	internal

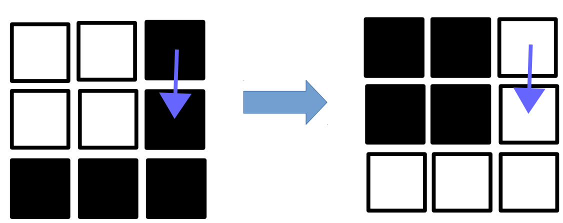
A geometric path is the interpretation "internal", since the property `g` must hold for all points.

It is common to write the geometric path the following way:

f_{111} $\quad$ Using the interpretation code `111` in the anageometric path

By swapping `g` with `¬g`, one gets the interpretation "external" (`¬g <=> not · g`):

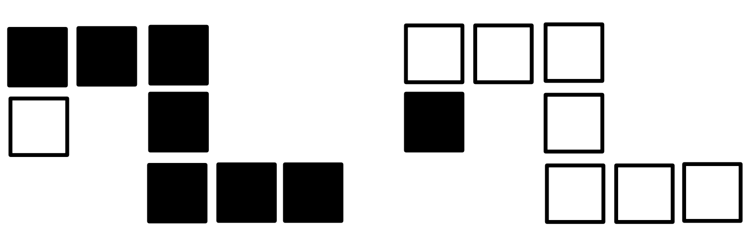
$f_{000} := \backslash(a : T, b : T) = \backslash(g : T \rightarrow bool) = f_{111}(a, b)(\neg g)$



The paths in these two spaces are very different, even the relation between the spaces is simple.

A "head" can be thought of as a path starting inside but then immediately goes outside:

$f_{100} := \backslash(a : T, c : T) = \backslash(g : T \rightarrow bool) = \neg g(a) \wedge f_1(a, b)(true_1) \wedge \exists\, b : g \{ f_{000}(b, c)(g) \}$
$f_{011} := \backslash(a : T, b : T) = \backslash(g : T \rightarrow bool) = f_{100}(a, b)(\neg g)$



The “neg tip” and “tip” are constructed in a similar way:

f_{110} := \(a : T, c : T) = \(g : T → bool) = ¬g(c) ∧ ∃ b : g { f_{000}(a, b)(g) ∧ ¬f_{000}(b, c)($true_1$) }
f_{001} := \(a : T, b : T) = \(g : T → bool) = f_{110}(a, b)(¬g)

The “wall” and “brigde” are constructed the following way:

f_{101} := \(a : T, c : T) = \(g : T → bool) = g(a) ∧ g(c) ∧ ∃ b : ¬g { f_1(a, b)($true_1$) ∧ f_1(b, c)($true_1$) }
f_{010} := \(a : T, b : T) = \(g : T → bool) = f_{101}(a, b)(¬g)

There is another way of constructing a “wall”, without using `$f_1$`:

f_{101} := \(a : T, c : T) = \(g : T → bool) =
∃ b : g { f_{100}(a, b)(g) ∧ f_{110}(a, b)(g) ∧ f_{100}(b, c)(g) ∧ f_{001}(b, c)(g) }

One then generalizes this construction to an interpretation code of arbitrary length:

- Topological equivalent binary numbers up to two bits
- A single bit is interpreted as “wall” or “bridge” respectively

For example:

00000100000 => 00100
00010100111 => 001010011

This topological transform can be described by the following L-system using two rules:

000 => 00 if code is longer than 3
111 => 11 if code is longer than 3

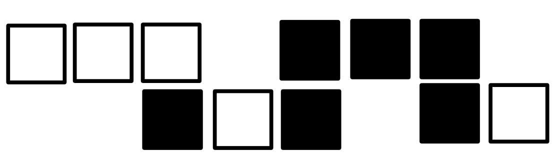
The anageometric path is the **infinite family of functions** from this construction:

f_x where `x` is some interpretation code of the anageometric path

For example, to ask whether some space `f` filled with `g` connects `a` to `b` by the following:

(a, b) : ($f_{1101001}$ g) <=> (a, b) : [$f_{1101001}$ g] true

Here, `a` starts inside the space, then after an internal path goes through a wall, a bridge, an external path and finally inside again to `b`:



With no access to `$f_1$`, it is not possible to construct the anageometric path, because one can not find two points that are only locally connected but not connected through any longer path.