Tailoring the shape calculus towards quantitative analysis

Speaker: Federico Buti

federico.buti@unicam.it
School of Science and Technology
Computer Science Division
University of Camerino - Italy

PaCo meeting
University of Camerino
September 15th 2010

Shape calculus - main features

- Non-deterministic timed calculus representing physical entities moving in 3D
- Processes = 3D shapes + dynamic behaviour
- Processes can move, collide and possibly bind
- Behaviours are specified with a timed CCS-like process algebra with channels aka “type of binders”
- $\langle a, X \rangle$, X is a portion of the surface of the process’s able to bind

Shape calculus - main features

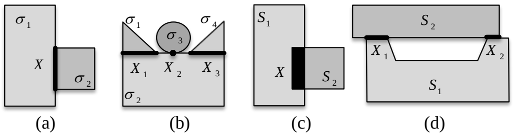
- Bound processes has behaviour equal to the interleaving of the component processes
- Compound processes can split weakly (no reaction) or split strongly (reaction) dividing in as many pieces as the reaction products
- Without communication (i.e. binding) collisions are considered elastic

Shapes in Shape Calculus

Shape Syntax

$S ::= \sigma \mid S \langle X \rangle S$ where $\sigma \in$ Basic and $X \subseteq \mathbb{R}^3$ is a *non-empty* set of points. The set X is intended to be the common surface on which the two shapes are attached.

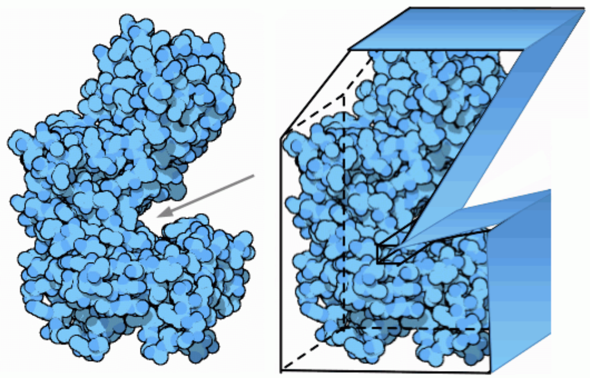
Examples of compound shapes in 2D



Shape example

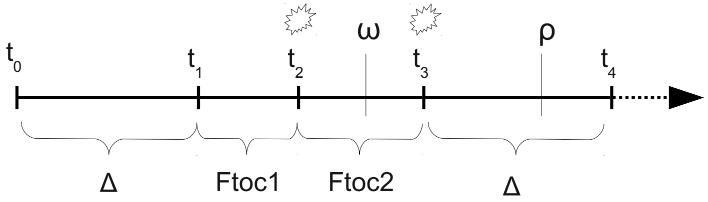
Representation of enzymatic reaction in Shape Calculus

Shape approximation



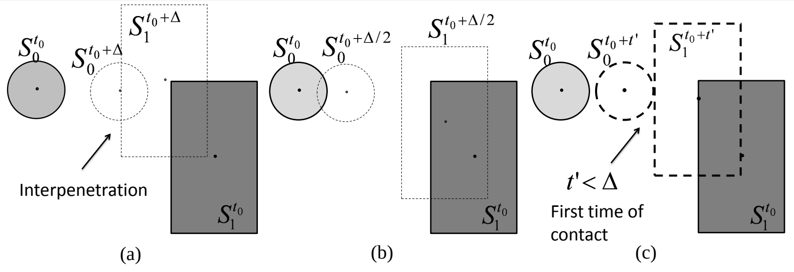
Time evolution and velocity update

- Time domain $\mathbb{T} = \mathbb{R}_0^+$ is then divided into an infinite sequence of movement time steps t_i such that $t_0 = 0$ and $t_i = t_{i-1} + \min(\Delta, Ftc(t_{i-1}), Ftr(t_{i-1}))$
- The updating of the velocities is represented by a function steer: $\mathbb{T} \to \text{Shapes} \hookrightarrow \mathbb{V}$ gives the velocity vector steer t S to assign to shape S at time t



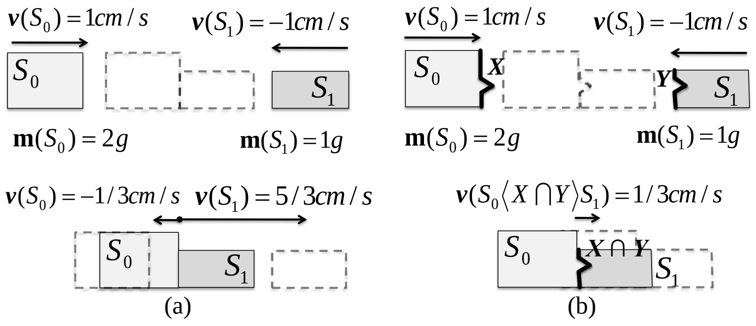
Collision Detection

First time of contact



Collision Response

Elastic and inelastic collision (one dimensional case)



Shapes behaviours

Set $\mathbb{B}$ of *shapes' behaviours* grammar

$$B ::= \mathrm{nil} \mid \langle \alpha, X \rangle.B \mid \omega(\alpha, X).B \mid \rho(L).B \mid \epsilon(t).B \mid B + B \mid K$$

where $\langle \alpha, X \rangle \in \mathcal{C}$, L is a non-empty subset of $\mathcal{C}$ whose channels are pairwise incompatible, $t \in \mathbb{T}$ and K is a process name in $\mathcal{K}$.

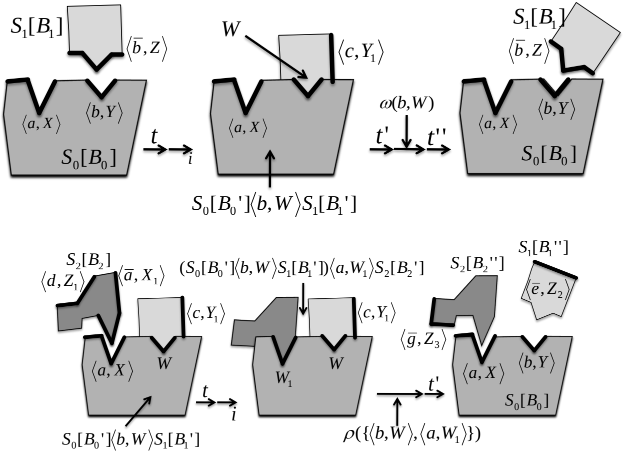
Set 3DP of *3D processes* grammar

$P ::= S[B] \mid P \langle a, X \rangle P$, where $S \in$ Shapes, $B \in \mathbb{B}$, $a \in \Lambda$ and $X \subseteq \mathbb{R}^3$ intersection between the surface active sites that are bound

Modelling Hexokinase in Shape Calculus

$$S_h[\mathrm{HEX}] \text{ where } \mathrm{HEX} = \langle \mathrm{atp}, X_{ha} \rangle.\mathrm{HA} + \langle \mathrm{glc}, Y_{hg} \rangle.\mathrm{HG}$$

Binding and Splitting



Modelling HEX, ATP and Glucose behaviours

$\mathsf{HEX} = \langle \mathsf{atp}, X_{ha} \rangle.\mathsf{HA} + \langle \mathsf{glc}, X_{hg} \rangle.\mathsf{HG},$

$\mathsf{HA} =$
$\omega(\mathsf{atp}, X_{ha}).\mathsf{HEX} + \epsilon(t_h).\langle \mathsf{glc}, X_{hg} \rangle.\rho(\{\langle \mathsf{atp}, X_{ha} \rangle, \langle \mathsf{glc}, Y_{hg} \rangle\}).\mathsf{HEX},$

$\mathsf{HG} =$
$\omega(\mathsf{glc}, X_{hg}).\mathsf{HEX} + \epsilon(t_h).\langle \mathsf{atp}, X_{ha} \rangle.\rho(\{\langle \mathsf{atp}, X_{ha} \rangle, \langle \mathsf{glc}, Y_{hg} \rangle\}).\mathsf{HEX},$
where X_{ha}, Y_{hg} are the surfaces of contact.

$\mathsf{ATP} = \langle \overline{\mathsf{atp}}, X_{ah} \rangle.(\epsilon(t_a).\rho(\{\langle \overline{\mathsf{atp}}, X_{ah} \rangle\}).\mathsf{ADP} + \omega(\overline{\mathsf{atp}}, X_{ah}).\mathsf{ATP})$

Towards quantitative analysis

- Shape Calculus can represent a great variety of scenarios
- Soon it will be integrated with our tool BIOSHAPE for analyse biological phenomena
- Providing formal verification techniques for the Shape Calculus is the next step
- The objective is to find the right abstractions to apply existing quantitative model checking or quantitative equivalence checking techniques

For instance...

- probabilistic timed automata could be useful to describe in more detail the behaviour of the processes and their interactions
- hybrid automata could be used to specify schemes of motion to be associated to certain classes of processes
- Suitable logic languages for specifying the properties must also be identified

Logic(s) for...

- verifying that certain 3D configurations are reached
- verifying that a certain molecule concentration is achieved
- verifying occurrence of certain oscillatory behaviours
- verifying that with a certain probability a reaction can occur
- ...

In the end...

...thanks for your attention! :)