10/19/06

# James Dolan

# Holodeck Strategies & CCCs

I'll talk about a relationship between:

Cartesian closed categories (CCCs)

products - exponentials algebras

the typed $\lambda$-calculus

programs

combinatorial games

trees

All these are "equivalent" in some sense!

Cartesian closed categories have products

$$A \times B$$

& exponentials

$$B^A$$

both defined by universal properties. To map into $A \times B$ is to map into $A$ and into $B$. To map into $B^A$:

$$X \longrightarrow B^A$$

is the same to map

$$X \times A \longrightarrow B$$

Or:

$$(B^A)^X \cong B^{X \times A}$$

Going from

$$f : X \times A \to B$$

to

$$\tilde{f} : X \to B^A$$

is called "currying". So is $B^{X \times A} \cong (B^A)^X$.

In fact, in the ordinary arithmetic of multiplication & exponentiation we have the same law:

$$B^{X \cdot A} = (B^A)^X$$

We can define a "products-exponentials algebra" to have an elt $\underline{1}$, operations $A \cdot B$ & $A^B$, and the usual laws, like

$$A \cdot B = B \cdot A$$

$$A \cdot (B \cdot C) = (A \cdot B) \cdot C$$

$$(A^B)^C = A^{B \cdot C}$$

$$(A \cdot B)^C = A^C B^C$$

etc. We can study the <u>free</u> products-exponentials algebra on one elt $X$, or (categorifying) the free cartesian closed category on one object $X$.

In fact, while [illegible] elements of the free products-exponentials algebra look like

a mess:

$$(X \cdot X^X)^{(X^X \cdot X \cdot X^{(X^X)})}$$

we can simplify them to a unique normal form using

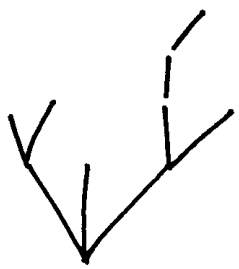
$$(A^B)^C \mapsto A^{BC}$$

$$(A \cdot B)^C \mapsto A^C B^C$$

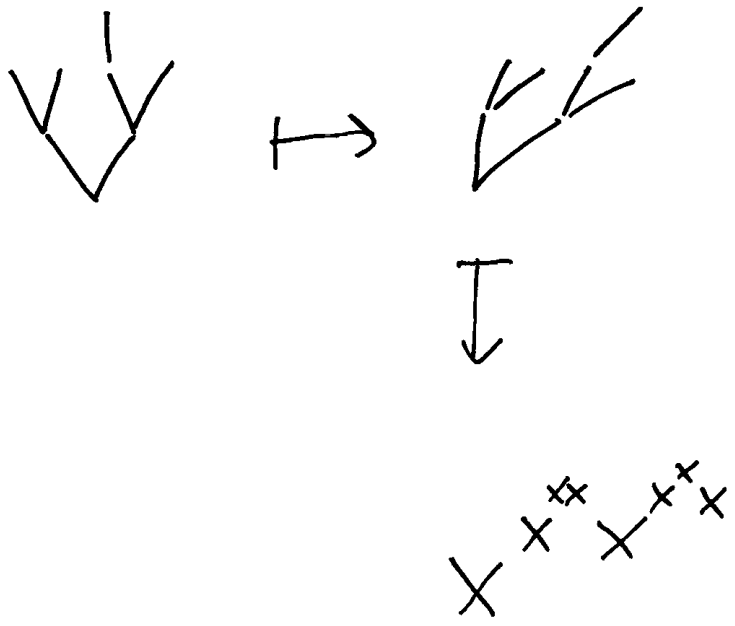
We get a (nonplanar) tree:

$$X^{XX}\ X\ X^{X^{X^X}}\ X$$

corresponds to



Conversely, any tree gives such an expression: just subject the tree to a severe westerly wind!



We can think of these trees as game trees for 2-person games, where players alternate moves & first person who can't move, loses.

Since $0^0 = 1$ ( $\mathrm{Hom}(\phi, \phi)$ has one element), we can evaluate any such expression at $x = 0$. We get 0 if the first player has a winning strategy, & 1 if the second player does!

So, we see a nontrivial relation between products-exponentials algebras & 2-person games, w. game trees corresponding to elts of the free algebra on $X$.

Now let's look at the free

CCC (cartesian closed category)

on one object $X$. Here objects

can be thought of as "data types".

If

$$U = \{1, \ldots, 3\}$$

$$T = \{1, \ldots, 5\}$$

then

$$T^U$$

is an "array" data type. An

"array" $A \in T^U$ is something where

$A(u) \in T$ for any $u \in U$.

We call

$$T \times U$$

a "record" data type (in PASCAL)

or a "struct" data type (in C++).

We can build up fancy data types using

"array" & "record" methods, & these are

just objects in a CCC. The morphisms

in this CCC are "programs" in a

programming language.

In the free CCC on some

objects, the objects are "abstract"

data types & morphisms are "abstract" programs. A product-preserving functor from this free CCC to some other one realizes these "abstract" data types & programs as actual, specific data types & programs.

What are the morphisms in the free CCC on one object in terms of games? They are strategies of a certain sort.

In other words, a morphism

$$1 \longrightarrow X^{\times x^{\times^{\times}}}$$

is some sort of strategy for the game with that tree:  .

What are products & exponentials like, in terms of games? For example, what's

$$\text{chess} \cdot \text{go}$$

or

$$\text{go}^{\text{chess}}$$

(after making these games into ideal

games of our sort, which always terminate & don't allow "draws"?

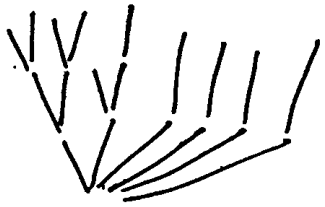
Products are easy: if

chess = 

&

go = 

then

chess · go = 

First player picks either chess or go & makes first move in that game; then they play that game!

Exponentials are subtler:

$$\left(\vee\right)^{\vee} = (x\,x^x)^{xx}$$

$$= x^{xx}\,(x^x)^{xx}$$

$$= x^{xx}\;x^{xxx}$$

$$= \;\vee\!\!\vee$$

Graft a copy of the exponent tree to each bottom-level edge of the base tree. In terms of games, second player of $G^H$ can, on his first move, either play $G$ as usual or switch to $H$ & become first player.

Given a game $G$ let $\mathrm{Strat}(G)$ be the set of (winning) 2nd-player strategies. We can describe this recursively:

$$\mathrm{Strat}(G) = \prod_{\substack{\text{opening moves} \\ m \text{ of } G}} \left( \sum_{\substack{\text{opening moves} \\ n \text{ of } G_m}} \mathrm{Strat}(G_{m,n}) \right)$$

where $G_m$ is the game $G$ after the opening move $m$ has been made, & $G_{m,n}$ is the game $G$ after the opening move $m$ & second move $n$ have been made.

For any object in the free CCC on $X$, say $G$, morphisms

$$f: 1 \to G$$

are in 1-1 correspondence with (winning) holodeck strategies for the 2nd player.

In a holodeck strategy, the 2nd player can take back moves - "rewinding the tape" to go back to a previous position. Since he can do this ad infinitum, we only say he has a winning holodeck strategy if he can force a win after finitely many moves.

So, if you think about it, the set HolStrat(G) of (winning) holodeck strategies for the second player satisfies:

$$\mathrm{HolStrat}(G) = \prod_{\substack{\text{opening moves} \\ m \text{ of } G}} \left( \sum_{\substack{\text{opening moves} \\ n \text{ of } G_m}} \mathrm{HolStrat}\left(G_{m,n}^{G_m}\right) \right)$$

This is like the formula for Strat(G), but with $G_{m,n}^{G_m}$ instead of $G_{m,n}$. The second player of $G_{m,n}^{G_m}$ can, on his first move, either play $G_{m,n}$ as usual or switch to $G_m$ & become the first player — by our result on $G^H$. This amounts to letting him take back the move $n$!

Holodeck strategies for $G$ are the same as ordinary winning strategies for another game $\boxed{G}$, the holodeck version of $G$. (This is an infinite game, so a winning strategy is one that ensures a win after finitely many move.)

In fact:

$$\boxed{G} = \prod_{\substack{\text{opening moves} \\ m \text{ of } G}} \chi \left( \prod_{\substack{\text{opening moves} \\ n \text{ of } G_m}} \chi^{\boxed{G_{m,n}^{G_n}}} \right)$$

So: winning strategies for $\boxed{G}$ correspond to elements of the object $G$, i.e. morphisms $f: 1 \to G$.