

## 15-424/15-624 Recitation 11

### Game semantics

#### 1. General intuition!

The logic  $\mathbf{dGL}$  can get really confusing, with the current player changing all the time, and winning depending on the modality... so here are some handy-dandy intuitions to keep in mind:

- The starting player of *any* game  $\alpha$  is *always* Angel! In other words, by default, it's Angel making the choices!
- $\langle \alpha \rangle \phi$ : Angel wins if she can ensure  $\phi$ , no matter what Demon does.
- $\langle \alpha \rangle \phi$ : The dual! Demon wins if he can get  $\phi$  no matter what Angel does.

Take the  $\mathbf{dL}$  formulas  $[\alpha]\phi$  and  $\langle \alpha \rangle \phi$ . The first one is “hard” in the sense that we need to check  $\phi$  holds after *every* execution of  $\alpha$ .  $\langle \alpha \rangle \phi$  is easy, since we only need to find one execution in which  $\phi$  holds.

These intuitions carry over to  $\mathbf{dGL}$  quite nicely! First, imagine that there's no dual operator  $\alpha^d$ . In that situation,  $[\alpha]\phi$  states that Angel is making all the choices in a game that Demon needs to win! That's why it's hard! In  $\langle \alpha \rangle \phi$ , on the other hand, Angel is making the choices *and* Angel needs to win. So Angel finds that one execution that satisfies  $\phi$ , like in  $\mathbf{dL}$ .

The amazingness of  $\mathbf{dGL}$  is that choices can pass from one player to the other, unlike in  $\mathbf{dL}$ .

#### 2. Dual operators

In lecture, you saw dual operators for the regular operators. The idea is that every piece of non-determinism is, by default, resolved by Angel. Dual operators allow a single specific choice to be performed by Demon, and sometimes have their own notation!

Angel	Demon	Rewrite
$\alpha \cup \beta$	$\alpha \cap \beta$	$(\alpha^d \cup \beta^d)^d$
$\alpha^*$	$\alpha^\times$	$((\alpha^d)^*)^d$
$\alpha; \beta$	$\alpha; \beta$	$(\alpha^d; \beta^d)^d$
$x := \theta$	$x := \theta$	$(x := \theta)^d$
$x' = \theta$	$(x' = \theta)^d$	$(x' = \theta)^d$
$?H$	$(?H)^d$	$(?H)^d$

For example,  $\alpha \cup \beta$ 's Angel is matched by  $\alpha \cap \beta$  for Demon, and can be rewritten equivalently as  $(\alpha^d \cup \beta^d)^d$ . The dual operators on  $\alpha$  and  $\beta$  guarantee that after  $\cup$  is resolved by Demon, the game control returns to Angel. This is what we mean by  $\alpha \cap \beta$  - only the non-deterministic choice is resolved by Demon, and  $\alpha$  and  $\beta$  play “as usual”.

Neither  $\alpha; \beta$  and  $x := \theta$  have non-determinism, so it doesn't make sense to have a dual operator! We can add as many  $d$  as we want, it won't change!

Both  $\alpha^*$  and  $x' = \theta$  have non-deterministic choices so they get dual operators. Unfortunately, there's no special notation for the ODE... but don't feel sorry for it, because it's demonically evil!

The test  $?H$  is counter-intuitive. It turns out that if you want to win the game, you're not allowed to chicken out and fail the first test you find! Who knew? So, if  $?H$  fails, Angel loses; if  $(?H)^d$  fails, Demon loses; and this is independent of who is trying to achieve  $\phi$ , i.e. independent of the modality used,  $\langle \rangle$  or  $\llbracket \rrbracket$ .

### 3. Winning the game

Speaking of winning the game, let's try to help Angel win the following formula:

$$\langle (x := 0 \cap x := 1)^* \rangle x = 1$$

Demon gets the choice inside the loop, so he can always pick  $x := 0$ , which doesn't help Angel. But Angel gets to decide to repeat! But you can only win a game if it finishes, so Angel is actually not allowed to repeat to infinity. When that happens, Demon has won.

We need to give Angel a chance! Let's try this formula:

$$\llbracket (x := 0 \cap x := 1)^* \rrbracket x = 1$$

Angel should win this one, right? Unfortunately, *NOPE!* Now Demon gets to choose  $x := 1$  every time, and since Angel needs to stop repeating at some point, Demon gets to achieve  $x = 1$ .

How about this one?

$$\langle \langle (x' = -1)^d ; x := 0 \rangle^* \rangle x \geq 0$$

Demon can choose to evolve the ODE for a *really long time* since angels and demons are immortal, he can *really* bore Angel... but eventually he has to stop evolving. And then  $x$  gets assigned to 0, and Angel can decide whether to repeat. She says "Aw *hell*, no!", chuckling at the pun, and then proceeds to promptly win the game.

### 4. Semantics

In  $d\mathcal{GL}$ , the meaning of formulas is given by the sets of states in which they are true, denoted  $\llbracket \phi \rrbracket$ . Thus, a formula is valid iff  $\llbracket \phi \rrbracket = \mathcal{S}$ , where  $\mathcal{S}$  is the set of all states.

How to we get the meaning of implication?

$$\llbracket \phi \rightarrow \psi \rrbracket$$

Without immediately resorting to the  $\phi \rightarrow \psi \equiv \neg\phi \vee \psi$  equivalence, we know that an implication is true if its antecedent is false. In region semantics, this is

$$\llbracket \phi \rrbracket^c$$

It can also be true if that's not the case, but the conclusion is true:

$$\llbracket \psi \rrbracket$$

Both these sets of states satisfy the implication, so we add them together:

$$\llbracket \phi \rrbracket^c \cup \llbracket \psi \rrbracket$$

And that's the meaning of implication in set semantics!

For the games themselves, we calculate from which states each player has a winning strategy, i.e. a strategy that gets the the state to satisfy the desired property!

- $\llbracket \langle \alpha \rangle \phi \rrbracket = \varsigma_\alpha(\llbracket \phi \rrbracket)$ , i.e. the set of states that satisfy  $\langle \alpha \rangle \phi$  is the set of states from which Angel has a winning strategy for game  $\alpha$  to get to  $\phi$ .
- $\llbracket [\alpha] \phi \rrbracket = \delta_\alpha(\llbracket \phi \rrbracket)$ , same as above, but Demon that must have the winning strategy.

Let's quickly review some of the semantics for  $\varsigma$ . For sequential composition, we have  $\varsigma_{\alpha;\beta}(S) = \varsigma_\alpha(\varsigma_\beta(S))$ . Something weird is happening here! When we were talking about set-based semantics for hybrid programs at the beginning of the course, we had  $T_{\alpha;\beta}(S) = T_\beta(T_\alpha(S))$ . The difference is that the  $T$  semantics are forward-chaining, whereas the game  $\varsigma/\delta$  semantics are backward-chaining.

Fancy words to say that with the  $T$  semantics, you start with the initial conditions and try to get to the desired end set of states:

$$\llbracket \text{initial conditions} \rrbracket = S_0 \Rightarrow S_1 \Rightarrow \dots \Rightarrow S_n = \llbracket \text{safe} \rrbracket$$

With the game  $\varsigma/\delta$  semantics you start from the end set of states, and go backwards to try to figure out whether you end with the entire set of states:

$$\llbracket \text{initial conditions} \rrbracket = S_0 \Leftarrow S_1 \Leftarrow \dots \Leftarrow S_n = \llbracket \text{safe} \rrbracket$$

Differential equations also have an interesting intuition. The definitions for Angel,  $\varsigma_{x'=\theta}(S)$ , and Demon,  $\delta_{x'=\theta}(S)$ , differ only in a quantifier.

Remember that the game is always being played by Angel, so Angel gets to decide for how long to evolve. As long as there is a time  $t$  at which  $S$  has been reached, Angel wins!

It's much harder for Demon to win. Angel is controlling the ODE, so she can decide to evolve for however long, and stop at any time. This means that the only way to ensure that Demon wins is if Angel is not able to leave  $S$ , i.e., if for all times  $t$ , the ODE keeps the state within  $S$ .

## 5. Angel and Demon pull the ol' switcheroonies

So how is the dual operator handled?

$$\varsigma_{\alpha^d}(S) = (\varsigma_{\alpha}(S^c))^c$$

Well, this isn't helping! But let's examine it by parts. For example, what does  $\varsigma_{\alpha}(S^c)$  mean? Two things changed:

- (a) We changed the property Angel was trying to prove: instead of  $S$ , we now have  $S^c$ . So if Angel was trying to prove  $\phi$ , now she is trying to prove  $\neg\phi$ .
- (b) We changed the game that's being played! Angel was playing the dual game, meaning Demon was making the choices and making it hard for Angel. But now she's playing the original game, so she gets to make those choices!

Wait... So, Angel is now trying to prove the property she didn't want to prove, making the choices she didn't want to make?

### **PLOT TWIST!**

*Angel is actually Demon!*

Essentially,  $\varsigma_{\alpha}(S^c)$  is using Angel's semantics to pretend Angel is Demon, by making what previously were Demon's choices, to try to prove what previously was Demon's goal.

We totally forgot that the original formula had another complement though! So that means...

### **DOUBLE PLOT TWIST!**

*Angel is Angel after all!*

I know... I know. Have a moment, take a deep breath, calm down. Who knew CPS could be so exciting?

The original formula stated  $(\varsigma_{\alpha}(S^c))^c$ . So, Angel has a winning strategy for the dual game only when she can pretend to be Demon *and fails miserably*. If it's impossible for Angel to play Demon and win, then Angel has won!