15-424: Foundations of Cyber-Physical Systems

Lecture Notes on Truth & Proof

André Platzer

Carnegie Mellon University
Lecture 6

1 Introduction[1]

Lecture 5 on Dynamical Systems & Dynamic Axioms investigated dynamic axioms for dynamical systems, i.e. axioms in differential dynamic logic (dL) that characterize operators of the dynamical systems that dL describes by hybrid programs in terms of structurally simpler dL formulas. All it takes to understand the bigger system, thus, is to apply the axiom and investigate the smaller remainders. That lecture did not quite show all important axioms yet, but it still revealed enough to prove a property of a bouncing ball. Yet, there's more to proofs than just axioms. Proofs also have proof rules for combining fragments of arguments into a bigger proof by proof steps. Proofs, thus, are defined by the glue that holds axioms together into a single cohesive argument justifying its conclusion.

Recall that our proof about the (single-hop) bouncing ball from the previous lecture still suffered from at least two issues. While it was a sound proof and an interesting proof, the way we had come up with it was somewhat undisciplined. We just applied axioms seemingly at random at all kinds of places all over the logical formulas. After we see such a proof, that is not a concern, because we can just follow its justifications and appreciate the simplicity and elegance of the steps it took to justify the conclusion.[2] But better structuring would certainly help us find proofs more constructively in the first place. The second issue was that the axioms for the dynamics that Lecture 5 showed us did not actually help in proving the propositional logic and arithmetic

[1] By both sheer coincidence and by higher reason, the title of this lecture turns out to be closely related to the subtitle of a well-known book on mathematical logic [And02], which summarizes the philosophy we pursue here in a way that is impossible to improve upon any further: *To truth through proof*.

[2] Indeed, the proof in Lecture 5 on Dynamical Systems & Dynamic Axioms was creative in that it used axioms quite carefully in an order that minimizes the notational complexity. But it is not easy to come up with such (nonsystematic) shortcut proofs.

parts. So were were left with informal justifications of the resulting arithmetic at the end, which leaves plenty of room for subtle mistakes in correctness arguments.

The lecture today addresses both issues by imposing more structure on proofs and, as part of that, handle the operators of first-order logic that differential dynamic logic inherits (propositional connectives such as $\wedge, \vee, \rightarrow$) and quantifiers ($\forall, \exists$). As part of the structuring, we will make ample and crucial use of the dynamic axioms from Lecture 5. Yet, they will be used in a more structured way than so far. In a way that focuses their use on the top level of the formula and in the direction that actually simplifies the formulas.

These notes are based on [Pla08, Pla10, Chapter 2.5.2], where more information can be found in addition to more information in [Pla10, Appendix A]. Sequent calculus is discussed in more detail also in the handbook of proof theory [Bus98]. More resources and background material on first-order logic is also listed on the course web page.

While the previous Lecture 5 on Dynamical Systems & Dynamic Axioms laid down the most fundamental cornerstones of the Foundations of Cyber-Physical Systems and their rigorous reasoning principles, today's lecture revisits these fundamental principles and shapes them into a systematic proof approach. The most important learning goals of this lecture are:

Modeling and Control: This lecture deepens our understanding from the previous lecture on how discrete and continuous systems relate to one another in the presence of evolution domain constraints, a topic that the previous lecture only touched upon briefly.

Computational Thinking: Based on the core rigorous reasoning principles for CPS developed in the previous lecture, today's lecture is devoted to reasoning rigorously *and systematically* about CPS models. Systematic ways of reasoning rigorously about CPS are, of course, critical to getting more complex CPS right. The difference between the axiomatic way of reasoning rigorously about CPS [Pla12b] as put forth in the previous lecture and the systematic way [Pla08, Pla10] developed in today's lecture is not a big difference conceptually, but more a difference in pragmatics. That does not make it less important, though, and the occasion to revisit gives us a way of deepening our understanding of systematic CPS analysis principles. Today's lecture also explains ways of developing CPS proofs and logic proofs systematically and is an important ingredient for verifying CPS models of appropriate scale. This lecture also adds a fourth leg to the logical trinity of syntax, semantics, and axiomatics considered in Lecture 5. Today's lecture adds pragmatics, by which we mean the question of how to use axiomatics to justify the syntactic renditions of the semantical concepts of interest. That is, how to best go about conducting a proof to justify truth of a CPS conjecture.

CPS Skills: This lecture is mostly devoted to sharpening our analytic skills for CPS. We will also develop a slightly better intuition for the operational effects involved in CPS in that we understand in which order we should worry about operational effects and whether that has an impact on the overall understanding.

2 Truth and Proof

Truth is defined by the semantics of logical formulas. The semantics gives a mathematical meaning to formulas that, in theory, could be used to establish truth of a logical formula. In practice, this is usually less feasible, for one thing, because quantifiers of differential dynamic logic quantify over real numbers (after all their variables may represent real quantities like velocities and positions). Yet, there are (uncountably) infinitely many of those, so determining the truth value of a universally quantified logical formula directly by working with its semantics is challenging since that'd require instantiating it with infinitely many real numbers, which would keep us busy for a while. The same matter is even more difficult for the hybrid system dynamics involved in modalities of differential dynamic logic formulas, because hybrid systems have so many possible behaviors and are highly nondeterministic. Literally following all possible behaviors to check all reachable states hardly sounds like a way that would ever enable us to stop and conclude the system would be safe. Except, of course, if we happen to be lucky and found a bug during just one execution, because that would be enough to falsify the formula.

Yet, we are still interested in establishing whether a logical formula is true. Or, actually, whether the formula is valid, since truth of a logical formula depends on the state (cf. definition of $\nu \models \phi$ in Lecture 4 on Safety & Contracts) whereas validity of a logical formula is independent of the state (cf. definition of $\models \phi$), because validity means truth in all states. And validity of formulas is what we ultimately care about, because we want our safety analysis to hold in all permitted initial states of the CPS, not just one particular initial state ν. In that sense, valid logical formulas are the most valuable ones. We should devote all of our efforts to finding out what is valid, because that will allow us to draw conclusions about all states, including the real world state as well.

While exhaustive enumeration and simulation is hardly an option for systems as challenging as CPS, the validity of logical formulas can be established by other means, namely by producing a proof of that formula. Like the formula itself, but unlike its semantics, a proof is a syntactical object that is amenable, e.g., to representation and manipulation in a computer. The finite syntactical argument represented in a proof witnesses the validity of the logical formula that it concludes. Proofs can be produced in a machine. They can be stored to be recalled as witnesses and evidence for the validity of their conclusion. And they can be checked by humans or machines for correctness. They can also be inspected for analytic insights about the reasons for the validity of a formula, which goes beyond the factual statement of validity. A proof justifies the judgment that a logical formula is valid, which, without such a proof as evidence, is no more than an empty claim. And empty claims would hardly be useful foundations for building any cyber-physical systems on.

Truth and proof should be related intimately, however, because we would only want to accept proofs that actually imply truth, i.e. proofs that imply their consequences to be valid if their premises are. That is, proof systems should be *sound* in order to allow us to draw reliable conclusions from the existence of a proof. And, in fact, this course will exercise great care to identify sound reasoning principles. The converse and equally

intriguing question is that of completeness, i.e. whether all true formulas (again in the sense of valid) can be proved, which turns out to be much more subtle [Pla12a] and won't concern us until much later in this course.

3 Sequents

The proof built from axioms in Lecture 5 on Dynamical Systems & Dynamic Axioms to justify a safety property of a bouncing ball was creative and insightful, but also somewhat spontaneous and disorganized. In fact, it has not even quite become particularly obvious what exactly a proof was, except that it is somehow supposed to glue axioms together into a single cohesive argument.[3] But that is not a definition of a proof.

In order to have a chance to conduct more complex proofs, we need a way of structuring the proofs and keeping track of all questions that come up while working on a proof. But despite all the lamenting about the proof from Lecture 5, it has, secretly, been much more systematic than we were aware of. Even if it went in a non-systematic order as far as the application order of the proof rules is concerned, we still structured the proof quite well (unlike the ad-hoc arguments in Lecture 4 on Safety & Contracts). So part of what this lecture needs to establish is to turn this coincidence into an intentional principle. Rather than just coincidentally structuring the proof well, we want to structure all proofs well and make them all systematic by design.

Throughout this course, we will use *sequents*, which give us a structuring mechanism for conjectures and proofs. Sequent calculus was originally developed by Gerhard Gentzen [Gen35a, Gen35b] for studying properties of natural deduction calculi, but sequent calculi have been used very successfully for numerous other purposes since.

In a nutshell, sequents are essentially a standard form for logical formulas that is convenient for proving purposes, because, intuitively, it neatly aligns all available assumptions on the left and gathers what needs to be shown on the right.

Definition 1 (Sequent). A *sequent* is of the form $\Gamma \vdash \Delta$, where the *antecedent* Γ and *succedent* Δ are finite sets of formulas. The semantics of $\Gamma \vdash \Delta$ is that of the formula $\bigwedge_{\phi\in\Gamma} \phi \rightarrow \bigvee_{\psi\in\Delta} \psi$.

The antecedent Γ can be thought of as the formulas we assume to be true, whereas the succedent Δ can be understood as formulas for which we want to show that at least one of them is true assuming all formulas of Γ are true. So for proving a sequent $\Gamma \vdash \Delta$, we assume all Γ and want to show that one of the Δ is true. For some simple sequents like $\Gamma, \phi \vdash \phi, \Delta$, we directly know that they are valid, because we can certainly show ϕ if we assume ϕ (in fact, we will use this we will use this as a way of finishing a proof). For other sequents, it is more difficult to see whether they are valid (true under all circumstances) and it is the purpose of a proof calculus to provide a means to find out.

[3] It would have been very easy to define, though, by inductively defining formulas to be provable if they are either instances of axioms or follow from provable formulas using modus ponens [Pla12b].

The basic idea in sequent calculus is to successively transform all formulas such that Γ forms a list of all assumptions and Δ the set of formulas that we would like to conclude from Γ (or, to be precise, the set Δ whose disjunction we would like to conclude from the conjunction of all formulas in Γ). So one way of understanding sequent calculus is to interpret $\Gamma \vdash \Delta$ as the task of proving one of the formulas in the succedent Δ from all of the formulas in the antecedent Γ. But since dℒ is a classical logic, not an intuitionistic logic, we need to keep in mind that it is actually enough for proving a sequent $\Gamma \vdash \Delta$ to just prove the disjunction of all formulas in Δ from the conjunction of all formulas in Γ. For the proof rules of real arithmetic, we will later make use of this fact by considering sequent $\Gamma \vdash \Delta$ as an abbreviation for the formula $\bigwedge_{\phi\in\Gamma} \phi \rightarrow \bigvee_{\psi\in\Delta} \psi$, because both have the same semantics in dℒ.

Empty conjunctions $\bigwedge_{\phi\in\emptyset} \phi$ are equivalent to $true$. Empty disjunctions $\bigvee_{\phi\in\emptyset} \phi$ are equivalent to $false$.[4] Hence, the sequent $\vdash A$ means the same as the formula A. The empty sequent $\vdash$ means the same as the formula $false$.

Note 2 (Nonempty trouble with empty sequents)**.** *If you ever reduce a conjecture about your CPS to proving the empty sequent $\vdash$, then you are in trouble, because it is rather hard to prove false, since false isn't ever true. In that case, either you have taken a wrong turn in your proof, e.g., by discarding an assumption that was actually required for the conjecture to be true, or your CPS might take the wrong turn, because its controller can make a move that is actually unsafe.*

4 Structural Proof Rules

Before discussing any particular proof rules of dℒ, let us first understand some common properties of most sequent calculi. The antecedent and succedent of a sequent are considered as sets. So the order of formulas is irrelevant, and we implicitly adopt what is called the *exchange rule* and do not distinguish between the following two sequents

$$\Gamma, A, B \vdash \Delta \qquad \text{and} \qquad \Gamma, B, A \vdash \Delta$$

ultimately since $A \wedge B$ and $B \wedge A$ are equivalent anyhow, nor do we distinguish between

$$\Gamma \vdash C, D, \Delta \qquad \text{and} \qquad \Gamma \vdash D, C, \Delta$$

ultimately since $C \vee D$ and $D \vee C$ are equivalent. Antecedent and succedent are considered as sets, not multisets, so we implicitly adopt what is called the *contraction rule* and do not distinguish between the following two sequents

$$\Gamma, A, A \vdash \Delta \qquad \text{and} \qquad \Gamma, A \vdash \Delta$$

[4] Note that $true$ is the neutral element for the operation $\wedge$ and $false$ the neutral element for the operation $\vee$. That is $A \wedge true$ is equivalent to A for any A and $A \vee false$ is equivalent to A. So $true$ plays the same role that 1 plays for multiplication. And $false$ plays the role that 0 plays for addition. Another aspect of sequents $\Gamma \vdash \Delta$ that is worth mentioning is that other notations such as $\Gamma \Longrightarrow \Delta$ or $\Gamma \longrightarrow \Delta$ are also sometimes used in other contexts.

because $A \wedge A$ and A are equivalent, nor do we distinguish between

$$\Gamma \vdash C, C, \Delta \qquad \text{and} \qquad \Gamma \vdash C, \Delta$$

because $C \vee C$ and C are equivalent.

The only structural rule of sequent calculus that we will find reason to use explicitly in practice is the *weakening* proof rule (alias *hide* rule) that can be used to remove or hide formulas from the antecedent (Wl) or succedent (Wr), respectively:

$$\text{(Wr)}\ \frac{\Gamma \vdash \Delta}{\Gamma \vdash \phi, \Delta}$$

$$\text{(Wl)}\ \frac{\Gamma \vdash \Delta}{\Gamma, \phi \vdash \Delta}$$

Weakening rules are sound, since it is fine in (structural) logics to prove a sequent with more formulas in the antecedent or succedent by a proof that uses only some of those formulas. This is different in substructural logics such as linear logic. Proof rule Wl proves the conclusion $\Gamma, \phi \vdash \Delta$ from the premise $\Gamma \vdash \Delta$, which dropped the assumption ϕ. Surely, if premise $\Gamma \vdash \Delta$ is valid, then conclusion $\Gamma, \phi \vdash \Delta$ is valid as well, because it even has one more (unused) assumption available (ϕ). Proof rule Wr proves the conclusion $\Gamma \vdash \phi, \Delta$ from the premise $\Gamma \vdash \Delta$, which is fine because $\Gamma \vdash \Delta$ just has one less (disjunctive) option in its succedent. For this, recall that succedents have a disjunctive meaning.

At first sight, weakening may sound like a stupid thing to do in any proof, because rule Wl discards available assumptions (ϕ) and rule Wr discards available options (ϕ) for proving the statement. This seems to make it harder to prove the statement after using a weakening rule. But weakening is actually useful for managing computational and conceptual proof complexity by enabling us to throw away irrelevant assumptions. These assumptions may have been crucial for another part of the proof, but have just become irrelevant for the particular sequent at hand, which can, thus, be simplified to $\Gamma \vdash \Delta$. Weakening, thus, streamlines proofs, which can, e.g., also help speed up arithmetic immensely (Sect. 11).

5 Propositional Proof Rules

The first logical operators encountered during proofs are usually propositional logical connectives, because many dL formulas use forms such as $A \rightarrow [\alpha]B$ to express that all behavior of HP α leads to safe states satisfying B when starting the system in initial states satisfying A. For propositional logic, dL uses the standard propositional rules with the cut rule, which are listed in Fig. 1. Each of these propositional rules decompose the propositional structure of formulas and neatly divides everything up into assumptions (which will ultimately be moved to the antecedent) and what needs to be shown (which will be moved to the succedent). The rules will be developed one at a time in the order that is most conducive to their intuitive understanding.

Proof rule $\wedge$l is for handling conjunctions ($\phi \wedge \psi$) in the antecedent. It expresses that if a conjunction $\phi \wedge \psi$ is among the list of available assumptions in the antecedent, then

$$(\neg\text{r})\ \frac{\Gamma, \phi \vdash \Delta}{\Gamma \vdash \neg\phi, \Delta} \qquad (\vee\text{r})\ \frac{\Gamma \vdash \phi, \psi, \Delta}{\Gamma \vdash \phi \vee \psi, \Delta} \qquad (\wedge\text{r})\ \frac{\Gamma \vdash \phi, \Delta \quad \Gamma \vdash \psi, \Delta}{\Gamma \vdash \phi \wedge \psi, \Delta}$$

$$(\neg\text{l})\ \frac{\Gamma \vdash \phi, \Delta}{\Gamma, \neg\phi \vdash \Delta} \qquad (\vee\text{l})\ \frac{\Gamma, \phi \vdash \Delta \quad \Gamma, \psi \vdash \Delta}{\Gamma, \phi \vee \psi \vdash \Delta} \qquad (\wedge\text{l})\ \frac{\Gamma, \phi, \psi \vdash \Delta}{\Gamma, \phi \wedge \psi \vdash \Delta}$$

$$(\rightarrow\text{r})\ \frac{\Gamma, \phi \vdash \psi, \Delta}{\Gamma \vdash \phi \rightarrow \psi, \Delta} \qquad (ax)\ \frac{}{\Gamma, \phi \vdash \phi, \Delta}$$

$$(\rightarrow\text{l})\ \frac{\Gamma \vdash \phi, \Delta \quad \Gamma, \psi \vdash \Delta}{\Gamma, \phi \rightarrow \psi \vdash \Delta} \qquad (cut)\ \frac{\Gamma \vdash \phi, \Delta \quad \Gamma, \phi \vdash \Delta}{\Gamma \vdash \Delta}$$

Figure 1: Propositional proof rules of sequent calculus

we might just as well assume both conjuncts (ϕ and ψ, respectively) separately. If we set out to prove a sequent of the form in the conclusion ($\Gamma, \phi \wedge \psi \vdash \Delta$), then we can justify this sequent by instead proving the sequent in the premise ($\Gamma, \phi, \psi \vdash \Delta$), where the only difference is that the two assumptions ϕ and ψ are now assumed separately in the premise rather than jointly as a single conjunction as in the conclusion. If we just keep on using proof rule $\wedge$l often enough, then all conjunctions in the antecedent will ultimately have been split into their pieces. Recall that the order of formulas in a sequent $\Gamma \vdash \Delta$ is irrelevant because Γ and Δ are sets, so we can always pretend that the formula that we want to apply the $\wedge$l rule to is last in the antecedent. So $\wedge$l takes care of all conjunctions that appear as top-level operators in antecedents. But there are other logical operators to worry about as well.

Proof rule $\vee$r is similar to $\wedge$l but for handling disjunctions in the succedent. If we set out to prove the sequent $\Gamma \vdash \phi \vee \psi, \Delta$ in the conclusion with a disjunction $\phi \vee \psi$ in the succedent, then we might as well split the disjunction into its two disjuncts and prove the premise $\Gamma \vdash \phi, \psi, \Delta$ instead, since the succedent has a disjunctive meaning anyhow.

Proof rule $\wedge$r works differently, because if we are trying to prove a sequent $\Gamma \vdash \phi \wedge \psi, \Delta$ with a conjunction $\phi \wedge \psi$ in its succedent, it would not be enough at all to just prove $\Gamma \vdash \phi, \psi, \Delta$, because, as in rule $\vee$r, this would only enable us to conclude $\Gamma \vdash \phi \vee \psi, \Delta$. Instead, proving a conjunction in the succedent as in the conclusion of $\wedge$r requires proving both conjuncts, so a proof of $\Gamma \vdash \phi, \Delta$ *and* a proof of $\Gamma \vdash \psi, \Delta$. This is why rule $\wedge$r splits the proof into two branches, one for proving $\Gamma \vdash \phi, \Delta$ and one for proving $\Gamma \vdash \psi, \Delta$. Indeed, if both premises of rule $\wedge$r are valid then so is its conclusion. To see this, it is easier to first consider the case where Δ is empty and then argue by cases, once for the case where the disjunction corresponding to Δ is true and once where it is false.

Similarly, proof rule $\vee$l handles a disjunction in the antecedent. When the assumptions listed in the antecedent of a sequent contain a disjunction $\phi \vee \psi$, then there is no way of knowing which of the two can be assumed only that at least one of them can be assumed to be true. Rule $\vee$l, thus, splits the proof into cases. The left premise considers the case where the assumption $\phi \vee \psi$ held because ϕ was true. The right premise considers the case where assumption $\phi \vee \psi$ held because ψ was true. If both premises are valid

(because we can find a proof for them), then, either way, the conclusion $\Gamma, \phi \vee \psi \vdash \Delta$ will be valid no matter which of the two cases applies.

Proof rule $\rightarrow$r handles implications in the succedent by using the implicational meaning of sequents. The way to understand it is to recall how we would go about proving an implication. In order to prove an implication $\phi \rightarrow \psi$, we would assume the left-hand side ϕ (which $\rightarrow$r pushes into the assumptions listed in the antecedent) and try to prove its right-hand side ψ (which $\rightarrow$r thus leaves in the succedent).

Proof rule $\rightarrow$l is more involved. And one way to understand it is to recall that classical logic obeys the equivalence $(\phi \rightarrow \psi) \equiv (\neg\phi \vee \psi)$. A direct argument explaining $\rightarrow$l uses that when assuming an implication $\phi \rightarrow \psi$, we can only assume its right-hand side ψ after we have shown its respective assumption ϕ on its left-hand side.

Proof rule $\neg$r proves a negation $\neg\phi$ by, instead, assuming ϕ. Again, the easiest way of understanding this is for an empty Δ in which case rule $\neg$r expresses that the way of proving a negation $\neg\phi$ in the succedent of the conclusion is to instead assume ϕ in the antecedent in the premise and then proving a contradiction in the form of the empty succedent, which is false. Alternatively, rule $\neg$r can be understood using the semantics of sequents, since a conjunct ϕ on the left-hand side of an implication is semantically equivalent to a disjunct $\neg\phi$ on the right-hand side.

Proof rule $\neg$l handles a negation $\neg\phi$ among the assumptions in the antecedent of the conclusion by, instead, pushing ϕ into the succedent of the premise. Indeed, for the case of empty Δ, if ϕ were shown to hold assuming Γ, then Γ and $\neg\phi$ imply a contradiction in the form of the empty sequent, which is false. Again, a semantic argument using the semantics of sequents also justifies $\neg$l directly.

All these propositional rules make progress by splitting operators. And that will ultimately lead to atomic formulas, i.e. those formulas without any logical operators. But there is no way to ever properly stop the proof yet. That is what the axiom rule ax is meant for (not to be confused with the axioms from Lecture 5). The axiom rule ax closes a goal (there are no further subgoals, which we sometimes mark by a $*$ explicitly), because assumption ϕ in the antecedent trivially entails ϕ in the succedent (the sequent $\Gamma, \phi \vdash \phi, \Delta$ is a simple syntactic tautology). If, in our proving activities, we ever find a sequent of the form $\Gamma, \phi \vdash \phi, \Delta$, for any formula ϕ, we can immediately use the axiom rule ax to close this part of the prove.

Rule cut is Gentzen's *cut* rule [Gen35a, Gen35b] that can be used for case distinctions: The right subgoal assumes any additional formula ϕ in the antecedent that the left subgoal shows in the succedent. Dually: regardless of whether ϕ is actually true or false, both cases are covered by proof branches. Alternatively, and maybe more intuitively, the cut rule is fundamentally a lemma rule. The left premise proves an auxiliary lemma ϕ in its succedent, which the right premise then assumes in its antecedent (again consider the case of empty Δ first). We only use cuts in an orderly fashion to derive simple rule dualities and to simplify meta-proofs. In practical applications, cuts are not needed in theory. But in practice, complex practical applications make use of cuts for efficiency reasons. Cuts an be used, for example, to simplify arithmetic, or to first prove lemmas and then make ample use of them, in a number of places in the remaining proof.

Even though we write sequent rules as if the principal formula (like $\phi \wedge \psi$ in $\wedge$r,$\wedge$l)

were at the end of the antecedent or at the beginning of the succedent, respectively, the sequent proof rules can be applied to other formulas in the antecedent or succedent, respectively, because we consider their order to be irrelevant (aset).

6 Proofs

The dℒ calculus has further proof rules beyond the structural and propositional rules. But before investigating those additional rules, let us first understand what exactly a proof is, what it means to prove a logical formula, and how we know whether a proof rule is sound. The same notions of proof, provability and soundness work for propositional logic as for differential dynamic logic, except that the latter has more proof rules.[5] The soundness notion that will be sufficient for our purposes in this course is the following.

Definition 2 (Global Soundness)**.** A sequent calculus proof rule of the form

$$\frac{\Gamma_1 \vdash \Delta_1 \quad \ldots \quad \Gamma_n \vdash \Delta_n}{\Gamma \vdash \Delta}$$

is *sound* iff the validity of all premises implies the validity of the conclusion, i.e.

$$\vDash (\Gamma_1 \vdash \Delta_1) \ldots \text{ and } \vDash (\Gamma_n \vdash \Delta_n) \text{ implies } \vDash (\Gamma \vdash \Delta)$$

Recall from Def. 1 that the meaning of a sequent $\Gamma \vdash \Delta$ is $\bigwedge_{\phi\in\Gamma} \phi \rightarrow \bigvee_{\psi\in\Delta} \psi$, so that $\vDash (\Gamma \vdash \Delta)$ stands for $\vDash \left(\bigwedge_{\phi\in\Gamma} \phi \rightarrow \bigvee_{\psi\in\Delta} \psi\right)$.

A formula ϕ is provable or derivable (in the dℒ calculus) if we can find a dℒ proof for it that starts with axioms (rule ax) at all its leaves and ends with a sequent $\vdash \phi$ at the bottom and that has only used dℒ proof rules in between to go from their premises to their conclusion. The shape of a dℒ proof, thus, is a tree with the axioms at the top leaves and the formula that the proof proves at the bottom root. While constructing proofs, however, we would start with the desired goal $\vdash \phi$ a the bottom that we want as the eventual conclusion of the proof and we work our way backwards to the subgoals until they can be proven to be valid as axioms (ax). Once all subgoals have been proven to be valid axioms, they entail their respective conclusion, which, recursively, entail the original goal $\vdash \phi$. This property of preserving truth or preserving entailment is called soundness. Thus, while constructing proofs, we work bottom-up from the goal and apply all proof rules from the desired conclusion to the required premises. When we have found a proof, we justify formulas conversely from the axioms top-down to the original goal, because validity transfers from the premises to the conclusion when using sound proof rules.

[5] There is one subtlety with the interaction of $\forall$ and $\exists$ for automation purposes and how that leads to a more general notion of soundness [Pla08]. But this generalization is not needed for the purposes of this course.

construct proofs upwards $\uparrow$ $(\wedge\text{r})\ \dfrac{\Gamma \vdash \phi, \Delta \quad \Gamma \vdash \psi, \Delta}{\Gamma \vdash \phi \wedge \psi, \Delta}$ $\downarrow$ validity transfers downwards

We write $\vdash_{\mathsf{d}\mathcal{L}} \phi$ iff $\mathsf{d}\mathcal{L}$ formula ϕ can be *proved* with $\mathsf{d}\mathcal{L}$ rules from $\mathsf{d}\mathcal{L}$ axioms. That is, a $\mathsf{d}\mathcal{L}$ formula is inductively defined to be *provable* in the $\mathsf{d}\mathcal{L}$ sequent calculus if it is the conclusion (below the rule bar) of an instance of one of the $\mathsf{d}\mathcal{L}$ sequent proof rules, whose premises (above the rule bar) are all provable. A formula ψ is *provable* from a set Φ of formulas, denoted by $\Phi \vdash_{\mathsf{d}\mathcal{L}} \psi$, iff there is a finite subset $\Phi_0 \subseteq \Phi$ of formulas for which the sequent $\Phi_0 \vdash \psi$ is provable.

Example 3. A very simple (in fact propositional) proof of the formula

$$v^2 \leq 10 \wedge b > 0 \rightarrow b > 0 \wedge (\neg(v \geq 0) \vee v^2 \leq 10) \tag{1}$$

is shown in Fig. 2. The proof starts with the desired proof goal as a sequent at the bottom:

$$\vdash v^2 \leq 10 \wedge b > 0 \rightarrow b > 0 \wedge (\neg(v \geq 0) \vee v^2 \leq 10).$$

and proceeds by applying suitable sequent proof rules upwards.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{v^2 \leq 10, b > 0 \vdash b > 0}\,*}{v^2 \leq 10 \wedge b > 0 \vdash b > 0}\,ax}{v^2 \leq 10 \wedge b > 0 \vdash b > 0}\,\wedge\text{l} \quad \dfrac{\dfrac{\dfrac{}{v^2 \leq 10, b > 0 \vdash \neg(v \geq 0), v^2 \leq 10}\,*}{v^2 \leq 10 \wedge b > 0 \vdash \neg(v \geq 0), v^2 \leq 10}\,ax}{v^2 \leq 10 \wedge b > 0 \vdash \neg(v \geq 0) \vee v^2 \leq 10}\,\wedge\text{l}}{v^2 \leq 10 \wedge b > 0 \vdash b > 0 \wedge (\neg(v \geq 0) \vee v^2 \leq 10)}\,\wedge\text{r}$$
$$\vdash v^2 \leq 10 \wedge b > 0 \rightarrow b > 0 \wedge (\neg(v \geq 0) \vee v^2 \leq 10)\,\rightarrow\text{r}$$

Figure 2: A simple propositional example proof in sequent calculus

The first (i.e., bottom most) proof step applies proof rule $\rightarrow$r to turn the implication ($\rightarrow$) to the sequent level by moving its left-hand side into the assumptions tracked in the antecedent. The next proof step applies rule $\wedge$r to split the proof into the left branch for showing that conjunct $b > 0$ follows from the assumptions in the antecedent and into the right branch for showing that conjunct $\neg(v \geq 0) \vee v^2 \leq 10$ follows from the antecedent also. On the left branch, the proof closes with an axiom ax after splitting the conjunction $\wedge$ in the antecedent into its conjuncts with rule $\wedge$l. We mark closed proof goals with $*$, to indicate that we did not just stopped writing but that a subgoal is actually proved successfully. It makes sense that the left branch closes by the axiom rule ax, because its assumption $b > 0$ in the antecedent trivially implies the formula $b > 0$ in the succedent, because both formulas are identical. The right branch closes with an axiom ax after splitting the disjunction ($\vee$) in the succedent with rule $\vee$r and then splitting the conjunction ($\wedge$) in the antecedent with rule $\wedge$l. On the right branch,

the first assumption formula $v^2 \leq 10$ in the antecedent trivially implies the last formula in the succedent $v^2 \leq 10$, because both are identical, so the axiom rule ax applies. Now that all branches of the proof have closed (with ax and marked by $*$), we know that all leaves at the top are valid, and, hence, since the premises are valid, each application of a proof rule ensures that their respective conclusions are valid also, by soundness. By recursively following this proof from the leaves at the top to the original root at the bottom, we conclude that the original goal at the bottom is valid and formula (1) is, indeed, true under all circumstances (valid). And that is what we set out to prove, that formula (1) is valid, which the proof in Fig. 2 justifies.

While this proof does not prove any particularly exciting formula, it still shows how a proof can be built systematically in the $\mathsf{d}\mathcal{L}$ calculus and gives an intuition as to how validity is inherited from the premises to the conclusions. Note that the proof has been entirely systematic. All we did to come up with it was successively inspect the top-level operator in one of the logical formulas in the sequent and apply its corresponding propositional proof rule to find the resulting subgoals. All the while we were doing this, we carefully watched to see if the same formula shows up in the antecedent and succedent, for then the axiom rule ax closes that subgoal. There would be no point in proceeding with any other proof rule if the ax rule closes a subgoal.

Most interesting formulas will not be provable with the sequent proof rules we have seen so far, because those were only propositional and structural rules. Next, we, thus, set out to find sequent proof rules for the other operators of $\mathsf{d}\mathcal{L}$.

First, though, notice that the sequent proof rules are sound. We consider only one of the proof rules to show how soundness works. Soundness is crucial, however, so you are invited to prove soundness for the other rules (Exercise 3).

Proof. The proof rule $\wedge\mathsf{r}$ is sound. For this, consider any instance for which both premises $\Gamma \vdash \phi, \Delta$ and $\Gamma \vdash \psi, \Delta$ are valid and show that the conclusion $\Gamma \vdash \phi \wedge \psi, \Delta$ is valid. To show the latter, consider any state ν. If there is a formula $F \in \Gamma$ in the antecedent that is not true in ν (i.e. $\nu \not\models F$) there is nothing to show, because $\nu \models (\Gamma \vdash \phi \wedge \psi, \Delta)$ then holds trivially, because not all assumptions in Γ are satisfied in ν. Likewise, if there is a formula $G \in \Delta$ in the succedent that is true in ν (i.e. $\nu \models G$) there is nothing to show, because $\nu \models (\Gamma \vdash \phi \wedge \psi, \Delta)$ then holds trivially, because one of the formulas in the succedent is already satisfied in ν. Hence, the only interesting case to consider is the case where all formulas in $F \in \Gamma$ are true in ν and all formulas $G \in \Delta$ are false. In that case, since both premises were assumed to be valid, and Γ is true in ν but Δ false in ν, the left premise implies that $\nu \models \phi$ and the right premise implies that $\nu \models \psi$. Consequently, $\nu \models \phi \wedge \psi$ by the semantics of $\wedge$. Thus, $\nu \models (\Gamma \vdash \phi \wedge \psi, \Delta)$. As the state ν was arbitrary, this implies $\vDash (\Gamma \vdash \phi \wedge \psi, \Delta)$, i.e. the conclusion of the considered instance of $\wedge\mathsf{r}$ is valid. □

7 Dynamic Proof Rules

When making the proof for the bouncing ball from Lecture 5 on Dynamical Systems & Dynamic Axioms systematic by turning it into a sequent calculus proof, the first propositional step succeeds to turn $\rightarrow$ into a sequent, but the rest if the proof involves modalities referring to the behavior of hybrid programs. The next set of sequent rules for $\mathsf{d}\mathcal{L}$ handles the hybrid programs in the modalities. Lecture 5 on Dynamical Systems & Dynamic Axioms has already shown and (at least informally) justified axioms for dynamical systems that correspond to each of the operators of hybrid programs in $[\cdot]$ modalities of differential dynamic logic [Pla12b]. These were equivalence axioms which represent schemata of valid formulas such as

$$([\cup])\ [\alpha \cup \beta]\phi \leftrightarrow [\alpha]\phi \wedge [\beta]\phi$$

How can such valid equivalences be used in the context of a sequent calculus? There is more than one productive way to do that.

The $\mathsf{d}\mathcal{L}$ axioms such as axiom $[\cup]$ are primarily meant to be used for replacing the left-hand side $[\alpha \cup \beta]\phi$ by the structurally simpler right-hand side $[\alpha]\phi \wedge [\beta]\phi$, because that direction of use assigns meaning to $[\alpha \cup \beta]\phi$ in logically simpler terms, i.e. as a structurally simpler logical formula. Thus, whenever there is an occurrence of a formula of the form $[\alpha \cup \beta]\phi$, the equivalence axiom $[\cup]$ ought to be used from left to right to get rid of $[\alpha \cup \beta]\phi$ and replace it by the structurally simpler right-hand side of the axiom. The following two sequent proof rules allow replacements in that direction for formulas in the antecedent ($[\cup]l$) and succedent ($[\cup]r$), respectively.

$$([\cup]r)\ \frac{\Gamma \vdash [\alpha]\phi \wedge [\beta]\phi, \Delta}{\Gamma \vdash [\alpha \cup \beta]\phi, \Delta}$$

$$([\cup]l)\ \frac{\Gamma, [\alpha]\phi \wedge [\beta]\phi \vdash \Delta}{\Gamma, [\alpha \cup \beta]\phi \vdash \Delta}$$

The sequent proof rules $[\cup]r$,$[\cup]l$ are more systematic in that they orient the use of the axiom $[\cup]$ in the direction that makes formulas structurally simpler. Without such direction, proofs could apply axiom $[\cup]$ from left to right and then from right to left and from left to right again forever without making any actual progress. That does not happen with $[\cup]r$,$[\cup]l$, because they cannot simply go back.[6] Furthermore, the sequent rules $[\cup]r$,$[\cup]l$ focus the application of axiom $[\cup]$ to the top level of sequents. That is, $[\cup]r$,$[\cup]l$ can only be used for formulas of the succedent or antecedent, respectively, that are of the form $[\alpha \cup \beta]\phi$, not to any subformulas within those formulas that happen to be of this form. Abiding both of those restrictions imposes more structure on the proof, compared to the proof we produced in Lecture 5. In particular, there is exactly one sequent proof rule that can be applied to a formula of the form $[\alpha \cup \beta]\phi$ in a sequent.[7]

[6] Albeit, going back is still possible indirectly when using a reasonably creative *cut*. But that requires an intentional extra effort to do so, hence, does not happen accidentally during proof search.

[7] With the exception of differential equations and, to a lesser extent, loops, the whole $\mathsf{d}\mathcal{L}$ sequent calculus singles out exactly one proof rule to apply, just depending on the top-level logical operators in the formula. Differential equations are slightly more complicated, because there will eventually be more options for proving differential equations.

Reconsidering the contract-type rules from Lecture 4 on Safety & Contracts, we could have turned axiom [∪] from Lecture 5 on Dynamical Systems & Dynamic Axioms into the following two sequent proof rules instead of into the two sequent rules [∪]r,[∪]l:

$$\text{(R14)}\ \frac{\Gamma \vdash [\alpha]\phi, \Delta \quad \Gamma \vdash [\beta]\phi, \Delta}{\Gamma \vdash [\alpha \cup \beta]\phi, \Delta}$$

$$\text{(R15)}\ \frac{\Gamma, [\alpha]\phi, [\beta]\phi \vdash \Delta}{\Gamma, [\alpha \cup \beta]\phi \vdash \Delta}$$

These rules R14,R15 already split into separate subgoals (R14) or separate formulas (R15), respectively. It would be fine to use sequent rules R14,R15 instead of [∪]r,[∪]l, and, in fact, earlier versions of KeYmaera did. The disadvantage of rules R14,R15 compared to [∪]r,[∪]l is that rules R14,R15 have a less obvious relation to axiom [∪] and that they are asymmetric (they both look surprisingly different). This nuisance is overcome in the rules [∪]r,[∪]l, from which rules R14,R15 follow immediately with just one more application of rules ∧r or ∧l, respectively. Thus, [∪]r,[∪]l are more elementary and more atomic in that they isolate the proof-theoretical meaning of $[\alpha \cup \beta]\phi$, as opposed to already incorporating parts of the meaning of ∧ as well, which, after all, is what the propositional rules ∧r,∧l are supposed to capture. Consequently, while the result of alternative rules R14,R15 is what will ultimately happen after applying sequent rules [∪]r,[∪]l regardless, we decide against the alternatives R14,R15, because they blur the essence of the meaning of $[\alpha \cup \beta]\phi$ unnecessarily and make the symmetry of the antecedent and succedent use more apparent.

The other dℒ axioms from Lecture 5 on Dynamical Systems & Dynamic Axioms translate into sequent calculus proof rules in exactly the same way. The equivalences that the dℒ axioms identify are lifted to the sequent level by introducing a pair of sequent rules, one for the antecedent and one for the succedent, and orienting the equivalence such that formulas always get structurally simpler when applying the sequent rules. Thus, all dynamic modality rules of the dℒ sequent calculus transform a hybrid program into structurally simpler logical formulas by symbolic decomposition or symbolic execution.

In order to simplify notation, we adopt a convention that exhibits the symmetry of antecedent and succedent rules. Instead of rules [∪]r,[∪]l, Fig. 3 shows a single *symmetric rule* [∪] that does not mention the sequent sign ⊢ :

$$\frac{[\alpha]\phi \wedge [\beta]\phi}{[\alpha \cup \beta]\phi}$$

This is abbreviated notation to say that the same rule from a conclusion with a formula $[\alpha \cup \beta]$ in either antecedent or succedent can be proved from a premise with formula $[\alpha]\phi \wedge [\beta]\phi$ in the antecedent or succedent, respectively. That is, we consider the symmetric rule [∪] as an abbreviation for the two rules [∪]r,[∪]l. Fig. 3 lists a single symmetric rule [∪] but we pretend it had both rules [∪]r,[∪]l. The same applies to the other symmetric rules in Fig. 3, which each have a version of the rule for the antecedent and a version of the rule for the succedent. The antecedent version of [;] is called [;]l, its succedent version is called [;]r. The antecedent version of [′] is called [′]l, its succedent

version is called $[']r$ and so on (a full list is in Fig. 8 for reference). Furthermore, Fig. 3 lists rules both for formulas of the form $[\alpha]\phi$ and for formulas of the form $\langle\alpha\rangle\phi$.

$$(\langle;\rangle)\ \frac{\langle\alpha\rangle\langle\beta\rangle\phi}{\langle\alpha;\beta\rangle\phi} \qquad (\langle^{*n}\rangle)\ \frac{\phi\lor\langle\alpha\rangle\langle\alpha^*\rangle\phi}{\langle\alpha^*\rangle\phi} \qquad (\langle:=\rangle)\ \frac{\phi_x^\theta}{\langle x:=\theta\rangle\phi}$$

$$([;])\ \frac{[\alpha][\beta]\phi}{[\alpha;\beta]\phi} \qquad ([^{*n}])\ \frac{\phi\land[\alpha][\alpha^*]\phi}{[\alpha^*]\phi} \qquad ([:=])\ \frac{\phi_x^\theta}{[x:=\theta]\phi}$$

$$(\langle\cup\rangle)\ \frac{\langle\alpha\rangle\phi\lor\langle\beta\rangle\phi}{\langle\alpha\cup\beta\rangle\phi} \qquad (\langle?\rangle)\ \frac{H\land\psi}{\langle?H\rangle\psi} \qquad (\langle'\rangle)\ \frac{\exists t{\geq}0\,\big((\forall 0{\leq}\tilde{t}{\leq}t\,\langle x:=y(\tilde{t})\rangle H)\land\langle x:=y(t)\rangle\phi\big)}{\langle x'=\theta\,\&\,H\rangle\phi}\ ^1$$

$$([\cup])\ \frac{[\alpha]\phi\land[\beta]\phi}{[\alpha\cup\beta]\phi} \qquad ([?])\ \frac{H\to\psi}{[?H]\psi} \qquad ([']) \ \frac{\forall t{\geq}0\,\big((\forall 0{\leq}\tilde{t}{\leq}t\,[x:=y(\tilde{t})]H)\to[x:=y(t)]\phi\big)}{[x'=\theta\,\&\,H]\phi}\ ^1$$

[1] t and $\tilde{t}$ are fresh logical variables and $\langle x:=y(t)\rangle$ is the discrete assignment belonging to the solution y of the differential equation with constant symbol x as symbolic initial value.

Figure 3: Dynamic proof rules of $\mathsf{d}\mathcal{L}$ sequent calculus

Nondeterministic choices split into their alternatives ($\langle\cup\rangle$,$[\cup]$). For rule $[\cup]$: If all α transitions lead to states satisfying ϕ (i.e., $[\alpha]\phi$ holds) and all β transitions lead to states satisfying ϕ (i.e., $[\beta]\phi$ holds), then, all transitions of program $\alpha\cup\beta$ that choose between following α and following β also lead to states satisfying ϕ (i.e., $[\alpha\cup\beta]\phi$ holds). Dually for rule $\langle\cup\rangle$, if there is an α transition to a ϕ state ($\langle\alpha\rangle\phi$) or a β-transition to a ϕ state ($\langle\beta\rangle\phi$), then, in either case, there is a transition of $\alpha\cup\beta$ to ϕ ($\langle\alpha\cup\beta\rangle\phi$ holds), because $\alpha\cup\beta$ can choose which of those transitions to follow. A general principle behind the $\mathsf{d}\mathcal{L}$ proof rules that is most noticeable in $\langle\cup\rangle$,$[\cup]$ is that these proof rules symbolically decompose the reasoning into two separate parts and analyse the fragments α and β separately, which is good for scalability. For these symbolic structural decompositions, it is very helpful that $\mathsf{d}\mathcal{L}$ is a full logic that is closed under all logical operators, including disjunction and conjunction, for then the premises in $[\cup]$,$\langle\cup\rangle$ are $\mathsf{d}\mathcal{L}$ formulas again (unlike in Hoare logic [Hoa69]).

Sequential compositions are proven using nested modalities ($\langle;\rangle$,$[;]$). For rule $[;]$: If after all α-transitions, all β-transitions lead to states satisfying ϕ (i.e., $[\alpha][\beta]\phi$ holds), then also all transitions of the sequential composition $\alpha;\beta$ lead to states satisfying ϕ (i.e., $[\alpha;\beta]\phi$ holds). The dual rule $\langle;\rangle$ uses the fact that if there is an α-transition, after which there is a β-transition leading to ϕ (i.e., $\langle\alpha\rangle\langle\beta\rangle\phi$), then there is a transition of $\alpha;\beta$ leading to ϕ (that is, $\langle\alpha;\beta\rangle\phi$), because the transitions of $\alpha;\beta$ are just those that first do any α-transition, followed by any β-transition.

Rules $\langle^{*n}\rangle$,$[^{*n}]$ are the usual iteration rules, which partially unwind loops. Rule $\langle^{*n}\rangle$ uses the fact that ϕ holds after repeating α (i.e., $\langle\alpha^*\rangle\phi$), if ϕ holds at the beginning (for ϕ holds after zero repetitions then), or if, after one execution of α, ϕ holds after any number of repetitions of α, including zero repetitions (i.e., $\langle\alpha\rangle\langle\alpha^*\rangle\phi$). So rule $\langle^{*n}\rangle$ expresses that for $\langle\alpha^*\rangle\phi$ to hold, ϕ must hold either immediately or after one or more repetitions of α. Rule $[^{*n}]$ is the dual rule expressing that ϕ must hold after all of those

combinations for $[\alpha^*]\phi$ to hold.

Tests are proven by showing (with a conjunction in rule $\langle ? \rangle$) or assuming (with an implication in rule $[?]$) that the test succeeds, because test $?H$ can only make a transition when condition H actually holds true. Thus, for dL formula $\langle ?H \rangle \phi$, rule $\langle ? \rangle$ is used to prove that H holds true (otherwise there is no transition and thus the reachability property is false) and that ϕ holds after the resulting no-op. Rule $[?]$ for dL formula $[?H]\phi$, in contrast, assumes that H holds true (otherwise there is no transition and thus nothing to show) and shows that ϕ holds after the resulting no-op.

Given first-order definable flows for their differential equations, proof rules $\langle ' \rangle$,$[']$ handle continuous evolutions. These flows are combined in the discrete jump set $x := y(t)$. Given a solution $x := y(t)$ for the differential equation system with symbolic initial values $x_1, \ldots, x_n$, continuous evolution along differential equations can be replaced by a discrete jump $\langle x := y(t) \rangle$ with an additional quantifier for the evolution time t. The effect of the constraint on H is to restrict the continuous evolution such that its solution $x := y(\tilde{t})$ remains in the evolution domain H at all intermediate times $\tilde{t} \leq t$. This constraint simplifies to $true$ if the evolution domain restriction H is $true$, which makes sense, because there are no special constraints on the evolution (other than the differential equations) if the evolution domain region is described by $true$, hence the full space $\mathbb{R}^n$. A notable special case of rules $[']$ and $\langle ' \rangle$ is when the evolution domain H is $true$:

$$\frac{\forall t \geq 0\, [x := y(t)]\phi}{[x' = \theta]\phi} \qquad \frac{\exists t \geq 0\, \langle x := y(t) \rangle \phi}{\langle x' = \theta \rangle \phi} \tag{2}$$

Finally note that rules $[']$ and $\langle ' \rangle$ apply in similar ways to the case of differential equation systems [Pla08, Pla12b] (Exercise 5):

$$x_1' = \theta_1, \ldots, x_n' = \theta_n \,\&\, H$$

For a very simple example of a proof, see Fig. 4. This proof is still not very interesting.

$$\dfrac{\dfrac{\dfrac{\vdash v^2 \leq 10 \land -(-b) > 0 \to b > 0 \land (\neg(v \geq 0) \lor v^2 \leq 10)}{\vdash [c := 10]\big(v^2 \leq 10 \land -(-b) > 0 \to b > 0 \land (\neg(v \geq 0) \lor v^2 \leq c)\big)}\,[:=]r}{\vdash [a := -b][c := 10]\big(v^2 \leq 10 \land -a > 0 \to b > 0 \land (\neg(v \geq 0) \lor v^2 \leq c)\big)}\,[:=]r}{\vdash [a := -b; c := 10]\big(v^2 \leq 10 \land -a > 0 \to b > 0 \land (\neg(v \geq 0) \lor v^2 \leq c)\big)}\,[;]r$$

Figure 4: A simple dynamic example proof in sequent calculus

Incidentally, the proof in Fig. 4 ends with a premise at the top that is identical to the (provable) conclusion at the bottom of Fig. 2. So gluing both proofs together leads to a proof of the conclusion at the bottom of Fig. 4:

$$[a := -b; c := 10]\big(v^2 \leq 10 \land -a > 0 \to b > 0 \land (\neg(v \geq 0) \lor v^2 \leq c)\big)$$

Notice how substitutions are applied in the proof shown in Fig. 4 when using proof rule $[:=]r$, which requires quite some care. Observe another important subtlety. The proof

in Fig. 4 ends in a formula mentioning $-(-b) > 0$ while the proof in Fig. 2 starts with a formula mentioning $b > 0$ in the same place. Both formulas are, of course, equivalent, but, in order to glue both proofs, we still need to add a proof rule for this arithmetic transformation. We could add the following proof rule for such a purpose (Exercise 1), but will ultimately decide on adding a more powerful proof rule instead:

$$\frac{\Gamma, \theta > 0 \vdash \Delta}{\Gamma, -(-\theta) > 0 \vdash \Delta}$$

8 Quantifier Proof Rules

When trying to make the proof for the bouncing ball from Lecture 5 on Dynamical Systems & Dynamic Axioms systematic by turning it into a sequent calculus proof, the first propositional step succeeds, then a couple of steps succeed for splitting the hybrid program, but, ultimately, proof rule $[']r$ produces a quantifier that needs to be handled. And, of course, a mere inspection of the syntax of dL shows that there are logical operators that have no proof rules yet.

The proof rules for quantifiers come in two sets. The first set is standard in first-order logic. The second set is more unique to first-order logic of real arithmetic (but would also work for other decidable theories). Rules ∃r,∀l,∀r,∃l are standard proof rules for first-order logic, listed in Fig. 5. For explaining these quantifier proof rules, let us first assume for a moment there are no (existential) free variables $X_1, \ldots, X_n$ (i.e. $n = 0$, so with $\phi(s)$ instead of $\phi(s(X_1, \ldots, X_n))$) and use what is known as the ground calculus for dL.

$$(\exists\text{r})\ \frac{\Gamma \vdash \phi(\theta), \exists x\, \phi(x), \Delta}{\Gamma \vdash \exists x\, \phi(x), \Delta}\,{}^{1} \qquad (\forall\text{r})\ \frac{\Gamma \vdash \phi(s(X_1, .., X_n)), \Delta}{\Gamma \vdash \forall x\, \phi(x), \Delta}\,{}^{2}$$

$$(\forall\text{l})\ \frac{\Gamma, \phi(\theta), \forall x\, \phi(x) \vdash \Delta}{\Gamma, \forall x\, \phi(x) \vdash \Delta}\,{}^{1} \qquad (\exists\text{l})\ \frac{\Gamma, \phi(s(X_1, .., X_n)) \vdash \Delta}{\Gamma, \exists x\, \phi(x) \vdash \Delta}\,{}^{2}$$

[1] θ is an arbitrary term, often a new (existential) logical variable X.
[2] s is a new (Skolem) function and $X_1, .., X_n$ are all (existential) free logical variables of $\forall x\, \phi(x)$.

Figure 5: Proof rules for first-order quantifiers

The quantifier proof rules work much as in mathematics. Consider the proof rule ∀r, where we want to show a universally quantified property. When a mathematician wants to show a universally quantified property $\forall x\, \phi(x)$ to hold, he could choose a fresh symbol s (known as a *Skolem function symbol* or Herbrand function symbol in logic) and set out to prove that $\phi(s)$ holds (for s). Once he found a proof for $\phi(s)$, the mathematician would remember that s was arbitrary and his proof did not assume anything special about the value of s. So he would conclude that $\phi(s)$ must indeed hold for all s, and that, hence, $\forall x\, \phi(x)$ holds true. For example, to show that the square of all

numbers is nonnegative, a mathematician could start out by saying "let s be an arbitrary number", prove $s^2 \geq 0$ for s, and then conclude $\forall x\,(x^2 \geq 0)$, since s was arbitrary. Proof rule ∀r essentially makes this reasoning formal. It chooses a *new* (function) symbol s and replaces the universally quantified formula in the succedent by a formula for s (with all free logical variables $X_1, \ldots, X_n$ added as arguments, as we explain below, for now, think of $n = 0$ so no arguments). Notice, of course, that it is important to choose a new symbol s that has not been used (in the sequent) before. Otherwise, we would assume special properties about s in Γ, Δ that would not be justified to assume.

Consider proof rule ∃r, where we want to show an existentially quantified property. When a mathematician proves $\exists x\,\phi(x)$, he could directly produce any witness θ for this existential property and prove that, indeed, $\phi(\theta)$, for then he would have shown $\exists x\,\phi(x)$ with this witness. For example, to show that there is a number whose cube is less than its square, a mathematician could start by saying "let me choose 0.5 and show the property for 0.5". Then he could prove $0.5^3 < 0.5^2$, because $0.125 < 0.25$, and conclude that there, thus, is such a number, i.e., $\exists x\,(x^3 < x^2)$, because 0.5 was a perfectly good witness for that. Proof rule ∃r does that. It allows the choice of *any* term θ for x and accepts a proof of $\phi(\theta)$ as a proof of $\exists x\,\phi(x)$. However note that the claim "θ is a witness" may turn out to be wrong, for example, the choice 2 for x would have been a pretty bad start for attempting to show $\exists x\,(x^3 < x^2)$. Consequently, proof rule ∃r keeps both options $\phi(\theta)$ and $\exists x\,\phi(x)$ in the succedent.[8] If the proof with θ is successful, the sequent is valid and the part of the proof can be closed successfully. If the proof with θ later turns out to be unsuccessful, another attempt can be used to prove $\exists x\,\phi(x)$, e.g., by applying rule ∃r again to the same formula $\exists x\,\phi(x)$ that is still in the succedent, just with another attempt for a different witness θ_2.

This approach already hints at a practical problem. If we are very smart about our choice of the witness θ, rule ∃r leads to very short and elegant proofs. If not, we may end up going in circles without much progress in the proof. That is why KeYmaera allows you to specify a witness if you can find one (and you should if you can, because that gives much faster proofs) but also allows you to keep going without a witness, as detailed in Sect. 13.

Rules ∀l,∃l are dual to ∃r,∀l. Consider proof rule ∀l, where we have a universally quantified formula in the assumptions (antecedent) that we can use, and not in the succedent, which we want to show. In mathematics, when we know a universal fact, we can use this knowledge for any particular instance. If we know that all positive numbers have a square root, then we can also use the fact that 5 has a square root, because 5 is a positive number. Hence from assumption $\forall x\,(x > 0 \rightarrow \mathit{hasSqrt}(x))$ in the antecedent, we can also assume the particular instance $5 > 0 \rightarrow \mathit{hasSqrt}(5))$ that uses 5 for x. Rule ∀l can produce an instance $\phi(\theta)$ of the assumption $\forall x\,\phi(x)$ for an arbitrary term θ. Since we may need the universal fact $\forall x\,\phi(x)$ for multiple instantiations with

[8] KeYmaera does not actually keep $\exists x\,\phi(x)$ around in the succedent for rule ∃r and, for a fundamental reason [Pla08], does not have to. The same holds for rule ∀l, where KeYmaera does not keep $\forall x\,\phi(x)$ around in the antecedent, because it does not have to. That means, however, that if you conjecture θ to produce the right instance, and your conjecture turns out wrong during the proof, then you have to go back in the proof and undo your instantiation with θ.

$\theta_1, \theta_2, \theta_3$ during the proof, rule ∀l keeps the assumption $\forall x\, \phi(x)$ in the antecedent so that it can be used repeatedly to obtain different instances.

Consider proof rule ∃l in which we can use an existentially quantified formula from the antecedent. In mathematics, if we know an existential fact, then we can give a name to the object that we then know does exist. If we know that there is a smallest integer less than 10 that is a square, we can call it s, but we cannot denote it by a different term like 5, because 5 may be (and in fact is) the wrong answer. Rule ∃l gives a fresh name s (with all logical variables $X_1, \ldots, X_n$ as arguments) to the object that exists. Since it does not make sense to give a different name for the same existing object later, $\exists x\, \phi(x)$ is removed from the antecedent when adding $\phi(s(X_1, \ldots, X_n))$.

9 A Sequent Proof for a Non-Bouncing Ball

Recall the bouncing ball abbreviations from Lecture 5:

$$A_{x,v} \overset{\text{def}}{\equiv} 0 \leq x \wedge x = H \wedge v = 0 \wedge g > 0 \wedge 1 \geq c \geq 0$$
$$B_{x,v} \overset{\text{def}}{\equiv} 0 \leq x \wedge x \leq H$$
$$(x'' = -g) \overset{\text{def}}{\equiv} (x' = v, v' = -g)$$

And the single-hop bouncing ball formula from Lecture 5:

$$A_{x,v} \rightarrow [x'' = -g; (?x = 0; v := -cv \cup ?x \geq 0)]B_{x,v}$$

This time, we include the evolution domain but leave out the discrete part:

$$A_{x,v} \rightarrow [x'' = -g \,\&\, x \geq 0]B_{x,v} \tag{3}$$

Let there be proof, this time a proper sequent proof:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{*}{A_{x,v}, r{\geq}0 \vdash 0{\leq}r{\leq}r} \quad [:=]r\dfrac{A_{x,v}, r{\geq}0, H - \frac{g}{2}s^2 \geq 0 \vdash B_{H-\frac{g}{2}r^2,-gt}}{A_{x,v}, r{\geq}0, [x := H - \frac{g}{2}s^2]x \geq 0 \vdash [x := H - \frac{g}{2}r^2]B_{x,v}}}{A_{x,v}, r{\geq}0, 0{\leq}r{\leq}r \rightarrow [x := H - \frac{g}{2}s^2]x \geq 0 \vdash [x := H - \frac{g}{2}r^2]B_{x,v}}\rightarrow\text{l}}{A_{x,v}, r{\geq}0, \forall 0{\leq}s{\leq}r\, [x := H - \frac{g}{2}s^2]x \geq 0 \vdash [x := H - \frac{g}{2}r^2]B_{x,v}}\forall\text{l}}{A_{x,v}, r{\geq}0 \vdash \forall 0{\leq}s{\leq}r\, [x := H - \frac{g}{2}s^2]x \geq 0 \rightarrow [x := H - \frac{g}{2}r^2]B_{x,v}}\rightarrow\text{r}}{A_{x,v} \vdash r{\geq}0 \rightarrow (\forall 0{\leq}s{\leq}r\, [x := H - \frac{g}{2}s^2]x \geq 0 \rightarrow [x := H - \frac{g}{2}r^2]B_{x,v})}\rightarrow\text{r}}{A_{x,v} \vdash \forall t{\geq}0\, (\forall 0{\leq}s{\leq}t\, [x := H - \frac{g}{2}s^2]x \geq 0 \rightarrow [x := H - \frac{g}{2}t^2]B_{x,v})}\forall\text{r}}{A_{x,v} \vdash [x'' = -g \,\&\, x \geq 0]B_{x,v}}[']r}{\vdash A_{x,v} \rightarrow [x'' = -g \,\&\, x \geq 0]B_{x,v}}\rightarrow\text{r}$$

This proof boldly stated that the left premise closes, except that

$$A_{x,v}, r{\geq}0 \vdash 0{\leq}r{\leq}r$$

is not exactly an instance of the ax rule. So even here we need simple arithmetic to conclude that $0 \leq r \leq r$ is equivalent to $r \geq 0$ by reflexivity and flipping sides, at which point the left premise turns into a formula that can be closed by the ax rule:

$$ax\frac{*}{A_{x,v}, r{\geq}0 \vdash r{\geq}0}$$

A full formal proof and a KeYmaera proof, thus, need an extra proof step of arithmetic in the left premise. In paper proofs, we will frequently accept such minor steps as abbreviations but always take care to write down the reason. In the above example, we might, for example remark the arithmetic reason "by reflexivity of $\leq$ and by flipping $0 \leq r$ to $r \geq 0$".

The right premise is

$$A_{x,v}, r{\geq}0, H - \frac{g}{2}s^2 \geq 0 \vdash B_{H-\frac{g}{2}r^2,-gt}$$

which, when resolving abbreviations turns into

$$0 \leq x \land x = H \land v = 0 \land g > 0 \land 1 \geq c \geq 0, r{\geq}0, H - \frac{g}{2}s^2 \geq 0 \vdash 0 \leq H - \frac{g}{2}r^2 \land H - \frac{g}{2}r^2 \leq H$$

This sequent proves using $\land$r plus simple arithmetic for the left branch

$$0 \leq x \land x = H \land v = 0 \land g > 0 \land 1 \geq c \geq 0, r{\geq}0, H - \frac{g}{2}s^2 \geq 0 \vdash 0 \leq H - \frac{g}{2}r^2$$

resulting from $\land$r. We should again remark the arithmetic reason as "by flipping $0 \leq H - \frac{g}{2}r^2$ to $H - \frac{g}{2}r^2 \geq 0$". Some more arithmetic is needed on the right branch resulting from $\land$r:

$$0 \leq x \land x = H \land v = 0 \land g > 0 \land 1 \geq c \geq 0, r{\geq}0, H - \frac{g}{2}s^2 \geq 0 \vdash H - \frac{g}{2}r^2 \leq H$$

where we should remark the arithmetic reason "$g > 0$ and $r^2 \geq 0$". Finishing the above sequent proof up as discussed for the right premise, thus, shows that dℒ formula (3) at the conclusion of the proof is provable.

Throughout this course, you are strongly advised to write down such arithmetic reasons in your paper proofs to justify that the arithmetic is valid. KeYmaera provides a number of ways for proving arithmetic that will be discussed next.

10 Instantiating Real Arithmetic

Real arithmetic can be very challenging. That does not come as a surprise, because cyber-physical systems and the behavior of dynamical systems themselves is challenging. It is amazing that differential dynamic logic reduces challenging questions about CPS to just plain real arithmetic. Of course, that means that you may be left with challenging arithmetic, of quite noticeable computational complexity. This is one part

where you can use your creativity to master challenging verification questions by helping KeYmaera figure them out. While there will soon be more tricks in your toolbox to overcome the challenges of arithmetic, we discuss some of them in this lecture.

Providing instantiations for quantifier rules ∃r,∀l can significantly speed up real arithmetic decision procedures. The proof in Sect. 9 instantiated the universal quantifier $\forall s$ for an evolution domain constraint by the end point r of the time interval using quantifier proof rule ∀l. This is a very common simplification that usually speeds up arithmetic significantly (Note 4). It does not always work, because the instance one guesses may not always be the right one. Even worse, there may not always be a single instance that is sufficient for the proof, but that is a phenomenon that later lectures will examine.

Note 4 (Extreme instantiation)**.** *The proof rule ∀l for universal quantifiers in the antecedent as well as the rule ∃r for existential quantifiers in the succedent allow instantiation of the quantified variable x with any term θ.*

$$(\forall l)\ \frac{\phi(\theta), \forall x\, \phi(x) \vdash}{\forall x\, \phi(x) \vdash}\ ^{a}$$

The way this rule is used in KeYmaera is with a direct use of weakening rule Wl to hide the quantified formula:

$$(\forall l)\ \frac{\phi(\theta) \vdash}{\forall x\, \phi(x) \vdash}\ ^{b}$$

This instantiation is very helpful if only a single instance θ is important for the argument. Often, an extremal value for x is all it takes for the proof.

This happens often for quantifiers coming from the handling of evolution domains in proof rule [′]r. The proof steps that often help then is instantiation of intermediate time s by the end time t:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{*}{\Gamma, t{\geq}0 \vdash 0{\leq}t{\leq}t, [x := y(t)]\phi} \quad \dfrac{\dots}{\Gamma, t{\geq}0, [x := y(t)]H \vdash [x := y(t)]\phi}}{\Gamma, t{\geq}0, 0{\leq}t{\leq}t \to [x := y(t)]H \vdash [x := y(t)]\phi}\,{\to}l}{\Gamma, t{\geq}0, \forall 0{\leq}s{\leq}t\, [x := y(s)]H \vdash [x := y(t)]\phi}\,\forall l}{\Gamma, t{\geq}0 \vdash (\forall 0{\leq}s{\leq}t\, [x := y(s)]H) \to [x := y(t)]\phi}\,{\to}r}{\Gamma \vdash t{\geq}0 \to \big((\forall 0{\leq}s{\leq}t\, [x := y(s)]H) \to [x := y(t)]\phi\big)}\,{\to}r$$
$$\dfrac{}{\Gamma \vdash \forall t{\geq}0\, \big((\forall 0{\leq}s{\leq}t\, [x := y(s)]H) \to [x := y(t)]\phi\big)}\,\forall r$$

Similar instantiations can simplify arithmetic in other cases as well.

[a] θ is an arbitrary term, often a new (existential) logical variable X.
[b] θ is an arbitrary term, often a new (existential) logical variable X.

11 Weakening Real Arithmetic

Weakening rules Wl,Wr can be useful to hide irrelevant parts of a sequent to make sure they do not be a distraction for real arithmetic decision procedures.

In the proof in Sect. 9, the left premise was

$$A_{x,v}, r{\geq}0 \vdash 0{\leq}r{\leq}r$$

The proof of this sequent did not make use of $A_{x,v}$ at all. Here, the proof worked easily. But if $A_{x,v}$ were a very complicated formula, then proving the same sequent might have been very difficult, because our proving attempts could have been distracted by the presence of $A_{x,v}$ and all the lovely assumptions it provides. We might have applied lots of proof rules to $A_{x,v}$ before finally realizing that the sequent proves because of $r{\geq}0 \vdash 0{\leq}r{\leq}r$ alone.

The same kind of distraction can happen in decision procedures for real arithmetic, sometimes shockingly so [Pla10, Chapter 5]. Consequently, it often saves a lot of proof effort to simplify irrelevant assumptions away as soon as they have become unnecessary. Fortunately, there already is a proof rule for that purpose called weakening, which we can use on our example from the left premise in the proof of Sect. 9:

$$\text{WI}\frac{r{\geq}0 \vdash 0{\leq}r{\leq}r}{A_{x,v}, r{\geq}0 \vdash 0{\leq}r{\leq}r}$$

You are generally advised to get rid of assumptions that you no longer need. This will help you manage the relevant facts about your CPS and will also help the arithmetic in KeYmaera to succeed much quicker. Just be careful not to hide an assumption that you still need. But if you accidentally do, that can also be a valuable insight, because you found out what the safety of your system critically depends on.

12 Real Arithmetic

What, in general, can be done to prove real arithmetic? We managed to convince ourselves with ad-hoc arithmetic reasons that the simple arithmetic in the above proofs was fine. But that is neither a proper proof rule nor should we expect to get away with such simple arithmetic arguments for the full complexity of CPS.

Later lectures will discuss the handling of real arithmetic in much more detail. For now, the focus is on the most crucial elements for proving CPS. Differential dynamic logic and KeYmaera make use of a fascinating miracle: the fact that first-order logic of real arithmetic, however challenging it might sound, is perfectly decidable [Tar51]. In a nutshell, the notation $\text{QE}(\phi)$ denotes the use of real arithmetic reasoning on formula ϕ. For a formula ϕ of first-order real arithmetic, $\text{QE}(\phi)$ is a logical formula that is equivalent to ϕ but simpler, because $\text{QE}(\phi)$ is quantifier-free.

Definition 4 (Quantifier elimination)**.** A first-order theory admits *quantifier elimination* if, with each formula ϕ, a quantifier-free formula $\text{QE}(\phi)$ can be associated effectively that is equivalent, i.e. $\phi \leftrightarrow \text{QE}(\phi)$ is valid (in that theory).

Theorem 5 (Tarski [Tar51]). *The first-order logic of real arithmetic admits quantifier elimination and is, thus, decidable.*

The operation QE is further assumed to evaluate ground formulas (i.e., without variables), yielding a decision procedure for closed formulas of this theory (i.e., formulas without free variables). For a closed formula ϕ, all it takes is to compute its quantifier-free equivalent $\mathrm{QE}(\phi)$ by quantifier elimination. The closed formula ϕ is closed, so has no free variables or other free symbols, and neither will $\mathrm{QE}(\phi)$. Hence, ϕ as well as its equivalent $\mathrm{QE}(\phi)$ are either equivalent to *true* or to *false*. Yet, $\mathrm{QE}(\phi)$ is quantifier-free, so which one it is can be found out simply by evaluating the (variable-free) concrete arithmetic in $\mathrm{QE}(\phi)$.

Example 6. Quantifier elimination yields, e.g., the following equivalence by real arithmetic:

$$\mathrm{QE}(\exists x\,(ax+b=0)) \;\equiv\; (a\neq 0 \vee b=0).$$

Both sides are easily seen to be equivalent, i.e.

$$\vDash \exists x\,(ax+b=0) \leftrightarrow (a\neq 0 \vee b=0)$$

because a linear equation with nonzero inhomogeneous part has a solution iff its linear part is nonzero as well. The left-hand side of the equivalence may be hard to evaluate, because it conjectures the existence of an x and it is not clear how we might get such an x. The right-hand side, instead, is trivial to evaluate, because it is quantifier-free and directly says to compare the values of a and b to zero and that an x such that $ax+b=0$ will exist if and only if $a\neq 0$ or $b=0$. This is easy to check at least if a,b are either concrete numbers or fixed parameters for your CPS. Then all you need to do is make sure they satisfy these constraints.

Now, if we have quantifiers, QE can remove them for us. But we first need quantifiers. Rules ∀r,∃r,∀l,∃r went through a lot of trouble to get rid of the quantifiers in the first place. Oh my! That makes it kind of hard to eliminate them equivalently later on. Certainly the proof rules in Fig. 5 have not been particularly careful about eliminating quantifiers equivalently. Just think of what might happen if we did try to use ∃r with the wrong witness and then weaken the $\exists x\,\phi(x)$ away. That is cheaper than quantifier elimination, but hardly as precise and useful.

But if we misplaced a quantifier using the rules from Fig. 5, then all we need to do is to dream it up again and we are in business for eliminating quantifiers by QE. The key to understanding how that works is to recall that the Skolem function symbols were originally universal (and existential logical variables were originally existential).

With the rule i∀, we can reintroduce a universal quantifier for a Skolem term $s(X_1,\ldots,X_n)$, which corresponds to a previously universally quantified variable in the succedent or a previously existentially quantified variable in the antecedent. The point of reintroducing the quantifier is that this makes sense when the remaining formulas are first-order in the quantified variable so that they can be handled equivalently by quantifier elimination in real-closed fields. When we have proven the subgoal (with for all X) then

this entails the goal for the particular $s(X_1,\ldots,X_n)$. In particular, when we remove a quantifier with ∀r,∃l to obtain a Skolem term, we can continue with other proof rules to handle the dynamic modalities and then reintroduce the quantifier for the Skolem term with i∀ once quantifier elimination for real arithmetic becomes applicable.

The dual rule i∃ can reintroduce an existential quantifier for a free logical variable that was previously existentially quantified in the succedent or previously universally quantified in the antecedent. Again, this makes sense when the resulting formula in the premise is first-order in the quantified variable X so that quantifier elimination can eliminate the quantifier equivalently. When we remove a quantifier with ∃r,∀l to obtain a free logical variable, we can continue using other proof rules to handle the dynamic modalities and then reintroduce the quantifier for the free logical variable with i∃ once quantifier elimination is applicable.

$$\text{(i}\forall\text{)}\ \frac{\vdash \mathrm{QE}(\forall X\,(\Phi(X)\vdash\Psi(X)))}{\Phi(s(X_1,..,X_n))\vdash\Psi(s(X_1,..,X_n))}\ {}^{1} \qquad \text{(i}\exists\text{)}\ \frac{\vdash \mathrm{QE}(\exists X\ \bigwedge_i(\Phi_i\vdash\Psi_i))}{\Phi_1\vdash\Psi_1\ \ \ldots\ \ \Phi_n\vdash\Psi_n}\ {}^{2}$$

[1] X is a new logical variable. Further, QE needs to be defined for the formula in the premise.

[2] Among all open branches, free logical variable X only occurs in the branches $\Phi_i\vdash\Psi_i$. Further, QE needs to be defined for the formula in the premise, especially, no Skolem dependencies on X can occur.

In the grand scheme of things, it may not be so particularly apparent why it was a good idea to get rid of the quantifiers in the first place. But if there is a way of getting the quantifiers back and then eliminating them, once and for all, equivalently by QE, then that is pretty reasonable. Can you speculate on the reason why we would first want to mumble quantifiers away with the slightly impoverished rules in Fig. 5 before ultimately coming back with the big steamroller in the form of QE?

Before you read on, see if you can find the answer for yourself.

It is useful to get quantifiers out of the way first using the rules ∀r,∃r,∀l,∃l, because further sequent rules only work in the top-level, so quantifiers need to get out of the way before any other proof rules could be applied. And if the formula underneath the quantifier contains modalities with hybrid programs, then that is a bit much to ask from quantifier elimination to solve them for us as well. So the key is to first get rid of quantifiers by using extra Skolem functions or existential variables, work out the proof arguments for the remaining hybrid program modalities and then reintroduce quantifiers by i∀,i∃ to ask quantifier elimination for the answer to the arighmetic.

Real arithmetic equivalences can be used in differential dynamic logic to eliminate quantifiers as indicated in the proof rules i∀,i∃. But real arithmetic can also be used to otherwise simplify arithmetic (formally because other symbols can simply be considered as if they were Skolem constants).

13 Quantifier Proof Rules with Free Variables (Expert Reading)

There are two ways of using the proof rules in Fig. 5. One way is as we have explained in Sect. 8 by avoiding free variables X_i altogether and only choose ground terms without variables for instantiations θ in ∃r,∀l. In that case, all Skolem functions used in ∀r,∃l will have $n = 0$ free logical variables $X_1, \ldots, X_n$ as arguments (which is why they are also called Skolem constants in that case). This case is called a *ground calculus*, because free variables are never used and all term instantiations are ground (no free variables).

The other way is to leverage free variables and always use some fresh (existential) logical variable X for instantiation of θ every time ∃r,∀l are used. This is a free-variable calculus [HS94, Fit96, FM99] where ∃r,∀l are also called γ-rules and ∀r,∃l are called δ^+-rules [HS94], which is an improvement of what is known as the δ-rule [Fit96, FM99]. This case is called a *free-variable calculus*, because instantiations are with free variables. Later in the proof, these free variables can be requantified [Pla08]. The free variables $X_1, \ldots, X_n$ in the Skolem terms keep track of the dependencies of symbols and prevent instantiations where we instantiate X_1 by a term such as $s(X_1, \ldots, X_n)$ which already depends on X_1. The ground calculus and free-variable calculus uses of Fig. 5 can also be mixed, which can be a good idea in practice when you can guess values for some witnesses but are unsure about others. These dependency conditions, for example prevent the wrong of of reintroducing quantifiers in Fig. 13 and only allow the right way Fig. 6.

14 Creatively Cutting Real Arithmetic

Weakening is not the only propositional proof rule that can help speed your arithmetic. The *cut* rule is not just a curiosity, but can be very helpful in practice. It can speed up real arithmetic a lot when using a cut to replace a difficult arithmetic formula by a simpler one that is sufficient for the proof.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\text{i}\forall \text{ is not applicable}}{\vdash \mathrm{QE}(\exists X\ (2X+1 < s(X)))}\,\text{i}\exists}{\vdash 2X+1 < s(X)}\,\langle:=\rangle}{\vdash \langle x := 2X+1\rangle(x < s(X))}\,\forall\text{r}}{\vdash \forall y\, \langle x := 2X+1\rangle(x < y)}\,\exists\text{r}}{\vdash \exists x \forall y\, \langle x := 2x+1\rangle(x < y)}$$

Figure 6: **a** Wrong rearrangement attempt for quantifiers

$$\dfrac{\dfrac{\dfrac{\dfrac{\overbrace{\vdash \mathrm{QE}\ (\exists X\ \mathrm{QE}(\forall s\ (2X+1 < s)))}^{\text{false}}}{\vdash \mathrm{QE}(\forall s\ (2X+1 < s))}\,\text{i}\exists}{\vdash 2X+1 < s(X)}\,\text{i}\forall}{\vdash \langle x := 2X+1\rangle(x < s(X))}\,\langle:=\rangle}{\vdash \forall y\, \langle x := 2X+1\rangle(x < y)}\,\forall\text{r}}{\vdash \exists x \forall y\, \langle x := 2x+1\rangle(x < y)}\,\exists\text{r}$$

6**b**: Correct reintroduction order

For example, suppose $\psi(x)$ is a big and very complicated formula of first-order real arithmetic. Then proving the following formula

$$(x-y)^2 \leq 0 \land \psi(x) \to \psi(y)$$

by just real arithmetic will turn out to be surprisingly difficult and can take ages (even if it ultimately terminates). Yet, thinking about it, $(x-y)^2 \leq 0$ implies that $y = x$, which should make the rest of the proof easy since, $\psi(x)$ should easily imply $\psi(y)$ if $y = x$. How do we exhibit a proof based on these thoughts?

The critical idea to make such a proof work is to use *cut* for a creative cut with the suitable arithmetic. So we choose $y = x$ as the cut formula ϕ in *cut* and proceed as follows:

$$\dfrac{\dfrac{\dfrac{(x-y)^2 \leq 0 \vdash y = x}{(x-y)^2 \leq 0 \vdash y = x, \psi(y)}\,\text{Wr}}{(x-y)^2 \leq 0, \psi(x) \vdash y = x, \psi(y)}\,\text{Wl} \qquad \dfrac{\dfrac{}{\psi(x), y = x \vdash \psi(x)}\,ax^{*}}{\psi(x), y = x \vdash \psi(y)}\,=\text{r}}{(x-y)^2 \leq 0, \psi(x), y = x \vdash \psi(y)}\,\text{Wl}}{(x-y)^2 \leq 0, \psi(x) \vdash \psi(y)}\,cut$$
$$\dfrac{(x-y)^2 \leq 0, \psi(x) \vdash \psi(y)}{(x-y)^2 \leq 0 \land \psi(x) \vdash \psi(y)}\,\land\text{l}$$
$$\dfrac{(x-y)^2 \leq 0 \land \psi(x) \vdash \psi(y)}{\vdash (x-y)^2 \leq 0 \land \psi(x) \to \psi(y)}\,\to\text{r}$$

Indeed, the left premise proves easily using real arithmetic. The right premise proves comparably easily as well. This proof uses proof rule =r that is discussed next. Observe that proofs like this one benefit a lot from weakening to get rid of superfluous assumptions to simplify the resulting arithmetic.

15 Applying Equations by Substitution

The above cut proof uses the following proof rule for applying an equation to a formula ϕ by substituting the left-hand side x of an equation by its right-hand side θ. This substitution is sound, because x is assumed to be equal to θ in the antecedent. The same rule works applies to formulas ϕ that are in the antecedent (=l) as well as in the

succedent (=r). Obviously, the assumed equality $x = \theta$ has to be in the antecedent for the rule to be sound.

$$\text{(=r)}\ \frac{\Gamma, x = \theta \vdash \phi(\theta), \Delta}{\Gamma, x = \theta \vdash \phi(x), \Delta} \qquad \text{(=l)}\ \frac{\Gamma, x = \theta, \phi(\theta) \vdash \Delta}{\Gamma, x = \theta, \phi(x) \vdash \Delta}$$

It would be okay to use the equation in the other direction for replacing all occurrences of θ by x, because the equation $\theta = x$ is equivalent to $x = \theta$. Both proof rules, =r and =l apply an equation $x = \theta$ from the antecedent to an occurrence of x in the antecedent or succedent to substitute θ for x. By using the proof rule sufficiently often, multiple occurrences of x in Γ and Δ can be substituted.

Again, quantifier elimination would have been able to prove the same fact, but with significantly more time and effort. So you are advised to exploit these proof shortcuts whenever you find them.

16 Summary

The differential dynamic logic sequent proof rules that we have seen in this lecture are summarized in Fig. 7. They are sound [Pla08]. There are further proof rules of differential dynamic logic that later lectures will examine [Pla08, Pla12b].

Exercises

Exercise 1. Prove soundness of the following special purpose proof rule and use it to continue the proof in Fig. 4 similar to the proof in Fig. 2:

$$\text{(R19)}\ \frac{\Gamma, \theta > 0 \vdash \Delta}{\Gamma, -(-\theta) > 0 \vdash \Delta}$$

Exercise 2 (*). Since we are not adding proof rule R19 to the dℒ proof calculus, show how you can derive the same proof step using a creative combination of rule i∀ and the other proof rules.

Exercise 3. Prove soundness for the structural and propositional sequent proof rules considered in this lecture.

Exercise 4. Prove soundness for the dynamic sequent proof rules considered in this lecture. You can use a general argument how soundness of the dynamic sequent proof rules follows from soundness of the dℒ axioms considered in Lecture 5 on Dynamical Systems & Dynamic Axioms, but first need to prove soundness of those dℒ axioms.

Exercise 5 (*). Generalize the proof rules $[']$ and $\langle' \rangle$ to the case of differential equation systems:

$$x_1' = \theta_1, \ldots, x_n' = \theta_n \,\&\, H$$

First consider the easier case where $H \equiv true$.

Note 7.

$$(\neg\text{r})\ \frac{\Gamma, \phi \vdash \Delta}{\Gamma \vdash \neg\phi, \Delta} \qquad (\vee\text{r})\ \frac{\Gamma \vdash \phi, \psi, \Delta}{\Gamma \vdash \phi \vee \psi, \Delta} \qquad (\wedge\text{r})\ \frac{\Gamma \vdash \phi, \Delta \quad \Gamma \vdash \psi, \Delta}{\Gamma \vdash \phi \wedge \psi, \Delta}$$

$$(\neg\text{l})\ \frac{\Gamma \vdash \phi, \Delta}{\Gamma, \neg\phi \vdash \Delta} \qquad (\vee\text{l})\ \frac{\Gamma, \phi \vdash \Delta \quad \Gamma, \psi \vdash \Delta}{\Gamma, \phi \vee \psi \vdash \Delta} \qquad (\wedge\text{l})\ \frac{\Gamma, \phi, \psi \vdash \Delta}{\Gamma, \phi \wedge \psi \vdash \Delta}$$

$$(\to\text{r})\ \frac{\Gamma, \phi \vdash \psi, \Delta}{\Gamma \vdash \phi \to \psi, \Delta} \qquad (ax)\ \frac{}{\Gamma, \phi \vdash \phi, \Delta} \qquad (\text{Wr})\ \frac{\Gamma \vdash \Delta}{\Gamma \vdash \phi, \Delta}$$

$$(\to\text{l})\ \frac{\Gamma \vdash \phi, \Delta \quad \Gamma, \psi \vdash \Delta}{\Gamma, \phi \to \psi \vdash \Delta} \qquad (cut)\ \frac{\Gamma \vdash \phi, \Delta \ \ \Gamma, \phi \vdash \Delta}{\Gamma \vdash \Delta} \qquad (\text{Wl})\ \frac{\Gamma \vdash \Delta}{\Gamma, \phi \vdash \Delta}$$

$$(\langle;\rangle)\ \frac{\langle\alpha\rangle\langle\beta\rangle\phi}{\langle\alpha;\beta\rangle\phi} \qquad (\langle^{*n}\rangle)\ \frac{\phi \vee \langle\alpha\rangle\langle\alpha^*\rangle\phi}{\langle\alpha^*\rangle\phi} \qquad (\langle:=\rangle)\ \frac{\phi_x^\theta}{\langle x := \theta\rangle\phi}$$

$$([;])\ \frac{[\alpha][\beta]\phi}{[\alpha;\beta]\phi} \qquad ([^{*n}])\ \frac{\phi \wedge [\alpha][\alpha^*]\phi}{[\alpha^*]\phi} \qquad ([:=])\ \frac{\phi_x^\theta}{[x := \theta]\phi}$$

$$(\langle\cup\rangle)\ \frac{\langle\alpha\rangle\phi \vee \langle\beta\rangle\phi}{\langle\alpha \cup \beta\rangle\phi} \qquad (\langle?\rangle)\ \frac{H \wedge \psi}{\langle?H\rangle\psi} \qquad (\langle'\rangle)\ \frac{\exists t{\geq}0\,\big((\forall 0{\leq}\tilde{t}{\leq}t\,\langle x := y(\tilde{t})\rangle H) \wedge \langle x := y(t)\rangle\phi\big)}{\langle x' = \theta\,\&\,H\rangle\phi}\ [1]$$

$$([\cup])\ \frac{[\alpha]\phi \wedge [\beta]\phi}{[\alpha \cup \beta]\phi} \qquad ([?])\ \frac{H \to \psi}{[?H]\psi} \qquad ([']) \ \frac{\forall t{\geq}0\,\big((\forall 0{\leq}\tilde{t}{\leq}t\,[x := y(\tilde{t})]H) \to [x := y(t)]\phi\big)}{[x' = \theta\,\&\,H]\phi}\ [1]$$

$$(\exists\text{r})\ \frac{\Gamma \vdash \phi(\theta), \exists x\,\phi(x), \Delta}{\Gamma \vdash \exists x\,\phi(x), \Delta}\ [2] \qquad (\forall\text{r})\ \frac{\Gamma \vdash \phi(s(X_1, .., X_n)), \Delta}{\Gamma \vdash \forall x\,\phi(x), \Delta}\ [3]$$

$$(\forall\text{l})\ \frac{\Gamma, \phi(\theta), \forall x\,\phi(x) \vdash \Delta}{\Gamma, \forall x\,\phi(x) \vdash \Delta}\ [2] \qquad (\exists\text{l})\ \frac{\Gamma, \phi(s(X_1, .., X_n)) \vdash \Delta}{\Gamma, \exists x\,\phi(x) \vdash \Delta}\ [3]$$

$$(\text{i}\forall)\ \frac{\Gamma \vdash \text{QE}(\forall X\,(\Phi(X) \vdash \Psi(X))), \Delta}{\Gamma, \Phi(s(X_1, .., X_n)) \vdash \Psi(s(X_1, .., X_n)), \Delta}\ [4] \qquad (\text{i}\exists)\ \frac{\Gamma \vdash \text{QE}(\exists X\ \bigwedge_i(\Phi_i \vdash \Psi_i)), \Delta}{\Gamma, \Phi_1 \vdash \Psi_1, \Delta \ \ \ldots \ \ \Gamma, \Phi_n \vdash \Psi_n, \Delta}\ [5]$$

[1] t and $\tilde{t}$ are fresh logical variables and $\langle x := y(t)\rangle$ is the discrete assignment belonging to the solution y of the differential equation with constant symbol x as symbolic initial value.
[2] θ is an arbitrary term, often a new (existential) logical variable X.
[3] s is a new (Skolem) function and $X_1, .., X_n$ are all (existential) free logical variables of $\forall x\,\phi(x)$.
[4] X is a new logical variable. Further, QE needs to be defined for the formula in the premise.
[5] Among all open branches, free logical variable X only occurs in the branches $\Gamma, \Phi_i \vdash \Psi_i, \Delta$. Further, QE needs to be defined for the formula in the premise, especially, no Skolem dependencies on X can occur.

Figure 7: Most proof rules of the $\mathsf{d}\mathcal{L}$ sequent calculus

$$(\langle;\rangle r)\ \frac{\Gamma \vdash \langle\alpha\rangle\langle\beta\rangle\phi, \Delta}{\Gamma \vdash \langle\alpha;\beta\rangle\phi, \Delta} \qquad (\langle^{*n}\rangle r)\ \frac{\Gamma \vdash \phi \vee \langle\alpha\rangle\langle\alpha^*\rangle\phi, \Delta}{\Gamma \vdash \langle\alpha^*\rangle\phi, \Delta} \qquad (\langle:=\rangle r)\ \frac{\Gamma \vdash \phi_x^\theta, \Delta}{\Gamma \vdash \langle x:=\theta\rangle\phi, \Delta}$$

$$([;]r)\ \frac{\Gamma \vdash [\alpha][\beta]\phi, \Delta}{\Gamma \vdash [\alpha;\beta]\phi, \Delta} \qquad ([^{*n}]r)\ \frac{\Gamma \vdash \phi \wedge [\alpha][\alpha^*]\phi, \Delta}{\Gamma \vdash [\alpha^*]\phi, \Delta} \qquad ([:=]r)\ \frac{\Gamma \vdash \phi_x^\theta, \Delta}{\Gamma \vdash [x:=\theta]\phi, \Delta}$$

$$(\langle\cup\rangle r)\ \frac{\Gamma \vdash \langle\alpha\rangle\phi \vee \langle\beta\rangle\phi, \Delta}{\Gamma \vdash \langle\alpha\cup\beta\rangle\phi, \Delta} \qquad (\langle?\rangle r)\ \frac{\Gamma \vdash H \wedge \psi, \Delta}{\Gamma \vdash \langle?H\rangle\psi, \Delta}$$

$$([\cup]r)\ \frac{\Gamma \vdash [\alpha]\phi \wedge [\beta]\phi, \Delta}{\Gamma \vdash [\alpha\cup\beta]\phi, \Delta} \qquad ([?]r)\ \frac{\Gamma \vdash H \rightarrow \psi, \Delta}{\Gamma \vdash [?H]\psi, \Delta}$$

$$(\langle'\rangle r)\ \frac{\Gamma \vdash \exists t{\geq}0\ \big((\forall 0{\leq}\tilde{t}{\leq}t\ \langle x:=y(\tilde{t})\rangle H) \wedge \langle x:=y(t)\rangle\phi\big), \Delta}{\Gamma \vdash \langle x'=\theta\,\&\,H\rangle\phi, \Delta}\ {}^{1}$$

$$([']r)\ \frac{\Gamma \vdash \forall t{\geq}0\ \big((\forall 0{\leq}\tilde{t}{\leq}t\ [x:=y(\tilde{t})]H) \rightarrow [x:=y(t)]\phi\big), \Delta}{\Gamma \vdash [x'=\theta\,\&\,H]\phi, \Delta}\ {}^{1}$$

$$(\langle;\rangle l)\ \frac{\Gamma, \langle\alpha\rangle\langle\beta\rangle\phi \vdash \Delta}{\Gamma, \langle\alpha;\beta\rangle\phi \vdash \Delta} \qquad (\langle^{*n}\rangle l)\ \frac{\Gamma, \phi \vee \langle\alpha\rangle\langle\alpha^*\rangle\phi \vdash \Delta}{\Gamma, \langle\alpha^*\rangle\phi \vdash \Delta} \qquad (\langle:=\rangle l)\ \frac{\Gamma, \phi_x^\theta \vdash \Delta}{\Gamma, \langle x:=\theta\rangle\phi \vdash \Delta}$$

$$([;]l)\ \frac{\Gamma, [\alpha][\beta]\phi \vdash \Delta}{\Gamma, [\alpha;\beta]\phi \vdash \Delta} \qquad ([^{*n}]l)\ \frac{\Gamma, \phi \wedge [\alpha][\alpha^*]\phi \vdash \Delta}{\Gamma, [\alpha^*]\phi \vdash \Delta} \qquad ([:=]l)\ \frac{\Gamma, \phi_x^\theta \vdash \Delta}{\Gamma, [x:=\theta]\phi \vdash \Delta}$$

$$(\langle\cup\rangle l)\ \frac{\Gamma, \langle\alpha\rangle\phi \vee \langle\beta\rangle\phi \vdash \Delta}{\Gamma, \langle\alpha\cup\beta\rangle\phi \vdash \Delta} \qquad (\langle?\rangle l)\ \frac{\Gamma, H \wedge \psi \vdash \Delta}{\Gamma, \langle?H\rangle\psi \vdash \Delta}$$

$$([\cup]l)\ \frac{\Gamma, [\alpha]\phi \wedge [\beta]\phi \vdash \Delta}{\Gamma, [\alpha\cup\beta]\phi \vdash \Delta} \qquad ([?]l)\ \frac{\Gamma, H \rightarrow \psi \vdash \Delta}{\Gamma, [?H]\psi \vdash \Delta}$$

$$(\langle'\rangle l)\ \frac{\Gamma, \exists t{\geq}0\ \big((\forall 0{\leq}\tilde{t}{\leq}t\ \langle x:=y(\tilde{t})\rangle H) \wedge \langle x:=y(t)\rangle\phi\big) \vdash \Delta}{\Gamma, \langle x'=\theta\,\&\,H\rangle\phi \vdash \Delta}\ {}^{1}$$

$$([']l)\ \frac{\Gamma, \forall t{\geq}0\ \big((\forall 0{\leq}\tilde{t}{\leq}t\ [x:=y(\tilde{t})]H) \rightarrow [x:=y(t)]\phi\big) \vdash \Delta}{\Gamma, [x'=\theta\,\&\,H]\phi \vdash \Delta}\ {}^{1}$$

[1] t and $\tilde{t}$ are fresh logical variables and $\langle x:=y(t)\rangle$ is the discrete assignment belonging to the solution y of the differential equation with constant symbol x as symbolic initial value.

Figure 8: Dynamic proof rules of $\mathsf{d}\mathcal{L}$ sequent calculus (left and right rules corresponding to the symmetric rules in Fig. 3)

References

[And02] Peter B. Andrews. *An Introduction to Mathematical Logic and Type Theory: To Truth Through Proof*. Kluwer, 2nd edition, 2002.

[Bus98] Samuel R. Buss. An introduction to proof theory. In Samuel R. Buss, editor, *Handbook of Proof Theory*, chapter 1, pages 1–78. Elsevier, 1998.

[Fit96] Melvin Fitting. *First-Order Logic and Automated Theorem Proving*. Springer, New York, 2nd edition, 1996.

[FM99] Melvin Fitting and Richard L. Mendelsohn. *First-Order Modal Logic*. Kluwer, Norwell, MA, USA, 1999.

[Gen35a] Gerhard Gentzen. Untersuchungen über das logische Schließen. I. *Math. Zeit.*, 39(2):176–210, 1935.

[Gen35b] Gerhard Gentzen. Untersuchungen über das logische Schließen. II. *Math. Zeit.*, 39(3):405–431, 1935.

[Hoa69] Charles Antony Richard Hoare. An axiomatic basis for computer programming. *Commun. ACM*, 12(10):576–580, 1969.

[HS94] Reiner Hähnle and Peter H. Schmitt. The liberalized δ-rule in free variable semantic tableaux. *J. Autom. Reasoning*, 13(2):211–221, 1994.

[LIC12] *Proceedings of the 27th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2012, Dubrovnik, Croatia, June 25–28, 2012*. IEEE, 2012.

[Pla08] André Platzer. Differential dynamic logic for hybrid systems. *J. Autom. Reas.*, 41(2):143–189, 2008. doi:10.1007/s10817-008-9103-8.

[Pla10] André Platzer. *Logical Analysis of Hybrid Systems: Proving Theorems for Complex Dynamics*. Springer, Heidelberg, 2010. doi:10.1007/978-3-642-14509-4.

[Pla12a] André Platzer. The complete proof theory of hybrid systems. In LICS [LIC12], pages 541–550. doi:10.1109/LICS.2012.64.

[Pla12b] André Platzer. Logics of dynamical systems. In LICS [LIC12], pages 13–24. doi:10.1109/LICS.2012.13.

[Tar51] Alfred Tarski. *A Decision Method for Elementary Algebra and Geometry*. University of California Press, Berkeley, 2nd edition, 1951.