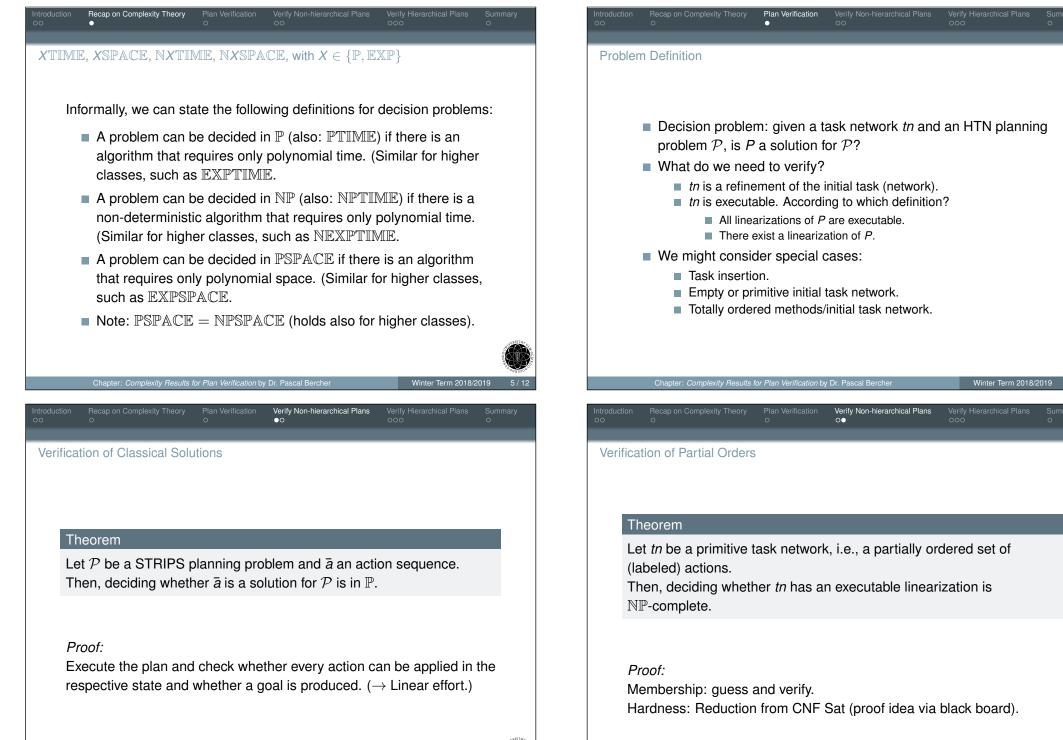
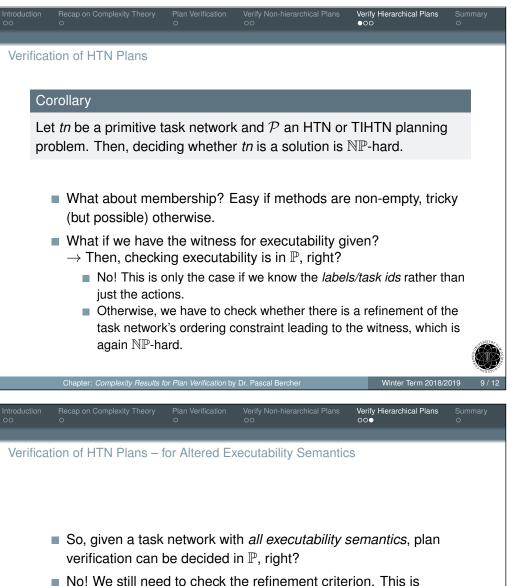


Why are we Interested in Complexity Studies?

Benefits of complexity studies:

- We know how to design algorithms:
 - If a problem is undecidable, any terminating algorithm must be wrong. Similarly: if a problem is Nℙ-complete, it is not a good idea to design a decision procedure that runs in polynomial time.
 - If the complexity of a problem is not known, at which runtime should we aim? P? EXPTIME?
- We can identify special cases to be exploited by algorithms. Example: heuristics! (Most of them exploit special cases that can be decided in P.)
- Insights may also allow for compilation techniques.
- Last, but not-at-all least: they help understanding the problem! (Understanding the problem should always be the first step.)





No! We still need to check the refinement criterion. This is \mathbb{NP} -hard, however. (Reduction from Vertex Cover, maybe later.)

Theorem

Deciding whether a (primitive) task network is a solution is, even for *all* executability semantics, \mathbb{NP} -complete.

ry	Introduction	Recap on Comp O	lexity Theory	Plan Verificat O	ion Verify Non-hier. 00		/erify Hierarchical Plans ⊃●O	Summary O
	Verifica	tion of HTN	Plans –	for Altered	d Executability	Semantics		
	 Let's consider a practically more Useful definition of executability. Let's require a primitive task network to be executable if and only if <i>every linearization</i> is executable. 							
	D	Theorem Deciding whether a (primitive) task network is executable (in the sense given above) is in \mathbb{P} .						
		<i>roof:</i> ack board.						
9 / 12		Chapter: Comp	plexity Results	for Plan Verificat	<i>tion</i> by Dr. Pascal Berch	er	Winter Term 2018	8/2019 10 /
ry	Introduction 00	Recap on Comp O	lexity Theory	Plan Verificat O	ion Verify Non-hier OO		/erify Hierarchical Plans	Summary ●
	Summa	ary						
	So far, we studied the computational complexity of the plan verification problem.							
	It ranges from \mathbb{P} to \mathbb{NP} -complete.							
	 Verifying total-order plans is much easier than verifying partially ordered plans. 							
	The hardness of verifying partially ordered plans depends on whether an executable linearization needs to exist or whether all of them need to be executable.							
			a hiorar	bioal pla	na ia aftan ha	ordor booo	uco wo olco r	hood

- Verifying hierarchical plans is often harder, because we also need to check the refinement criterion.
- Complexity results give raise to specialized algorithms, to heuristics, and to translations to other problem classes.