Computational Complexity

Lecture 7

Polynomial Hierarchy

Charting (some of) the space between P and PSPACE

(where much of the action happens)

Between P and PSPACE

Between P and PSPACE

- Recall NP

Between P and PSPACE

- Recall NP
  - Class of languages $\{ x \mid \exists u_1 \; F(x,u_1) \}$

Between P and PSPACE

- Recall NP
  - Class of languages $\{ x \mid \exists u_1 \ F(x,u_1) \}$
  - Where F in P: i.e., language $\{ x \mid F(x) \}$ is in P

Between P and PSPACE

- Recall NP
  - Class of languages $\{ x \mid \exists u_1\ F(x,u_1) \}$
  - Where F in P: i.e., language $\{ x \mid F(x) \}$ is in P
  - And $|u_1| < poly(|x|)$ for some polynomial

Between P and PSPACE

- Recall NP
  - Class of languages $\{ x \mid \exists u_1 \ F(x,u_1) \}$
  - Where F in P: i.e., language $\{ x \mid F(x) \}$ is in P
  - And $|u_1| < poly(|x|)$ for some polynomial
- Recall co-NP

Between P and PSPACE

- Recall NP
  - Class of languages $\{ x \mid \exists u_1 \ F(x,u_1) \}$
  - Where F in P: i.e., language $\{ x \mid F(x) \}$ is in P
  - And $|u_1| < poly(|x|)$ for some polynomial
- Recall co-NP
  - Class of languages $\{ x \mid \forall u_1 \ F(x,u_1) \}$

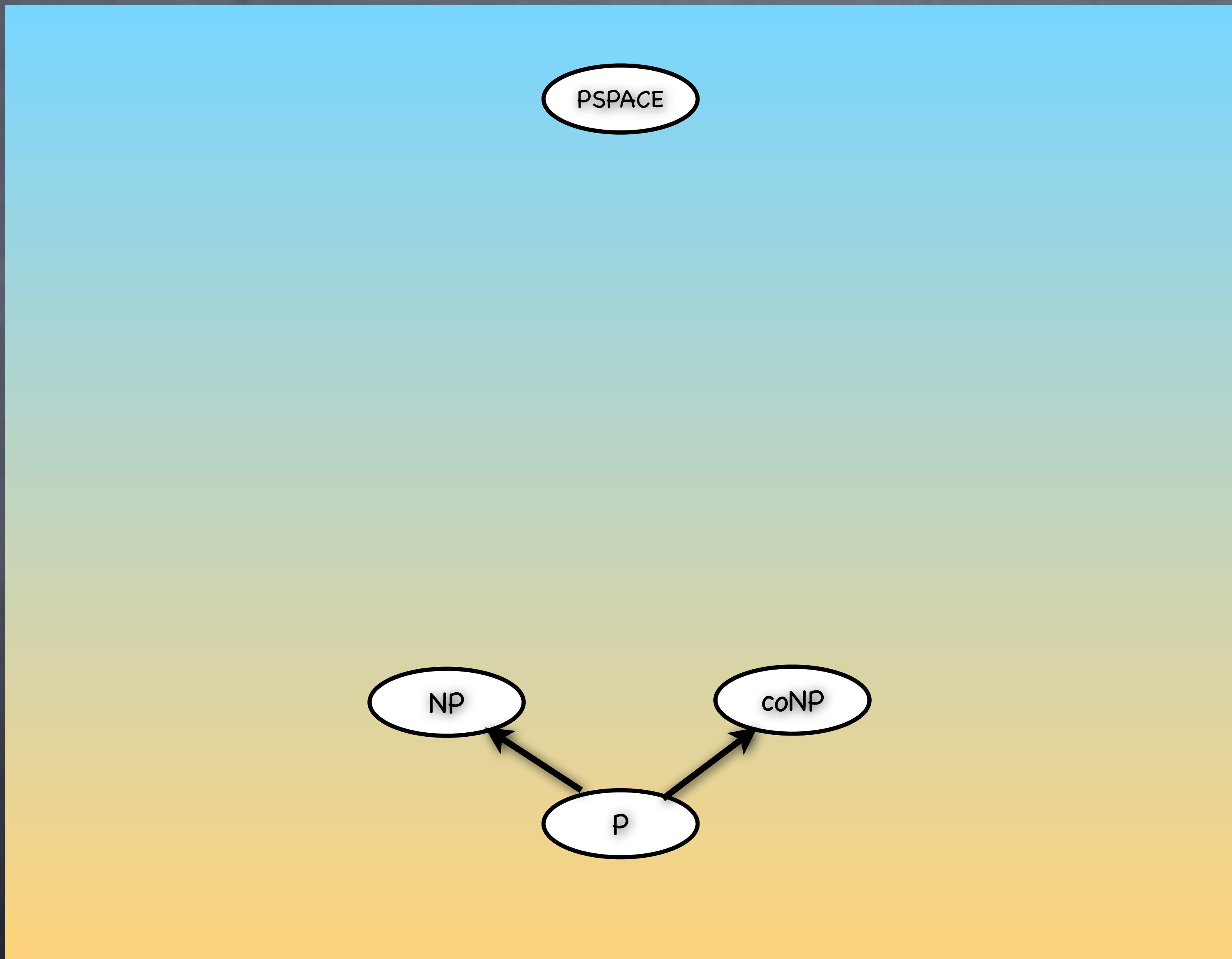
Between P and PSPACE

- Recall NP
  - Class of languages $\{ x \mid \exists u_1\ F(x,u_1) \}$
  - Where F in P: i.e., language $\{ x \mid F(x) \}$ is in P
  - And $|u_1| < poly(|x|)$ for some polynomial
- Recall co-NP
  - Class of languages $\{ x \mid \forall u_1\ F(x,u_1) \}$
- How about languages $\{ x \mid \exists u_1 \forall u_2 \ldots Q u_k\ F(x,u_1,u_2,\ldots,u_k) \}$

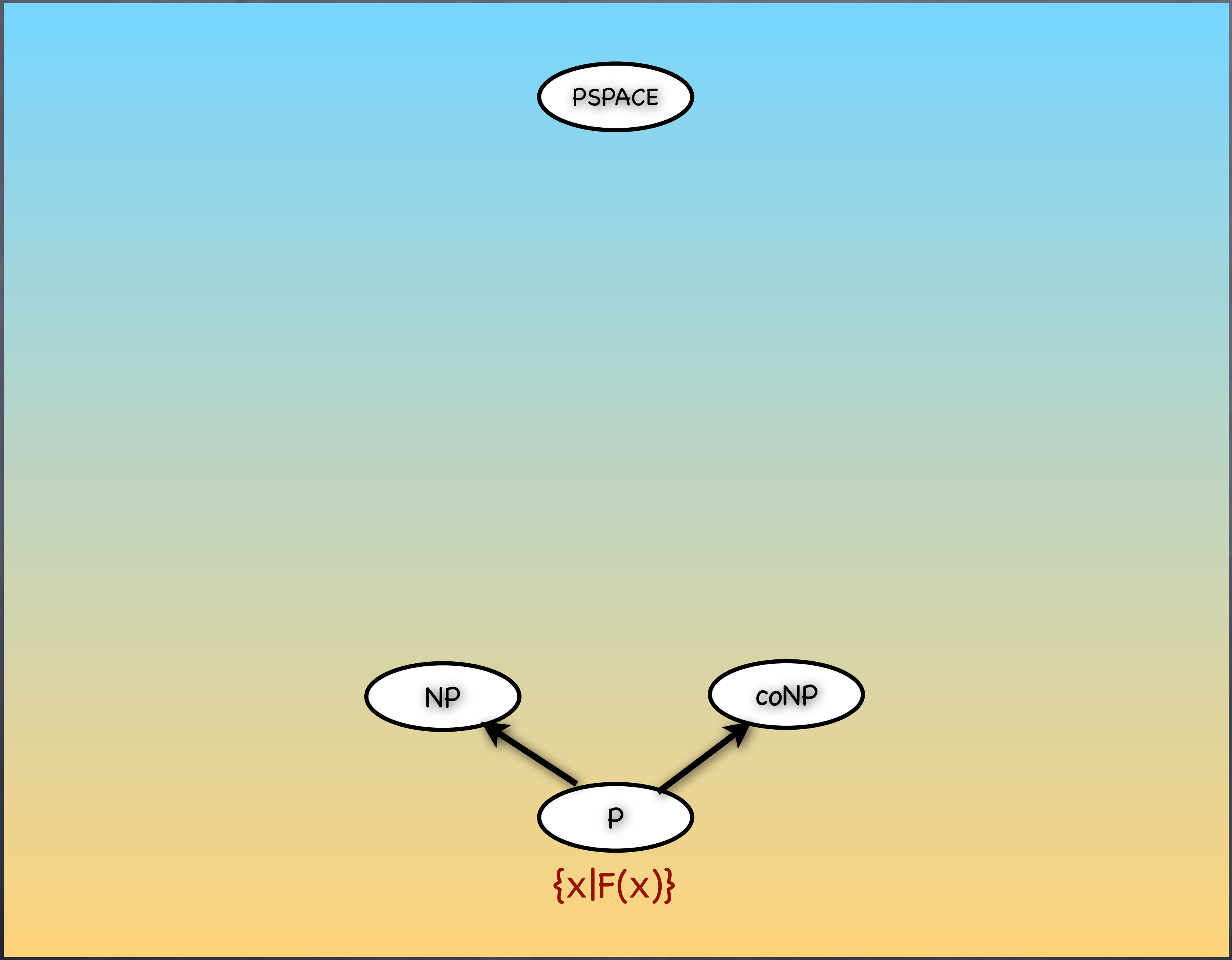
Between P and PSPACE

- Recall NP
  - Class of languages $\{ x \mid \exists u_1 \; F(x,u_1) \}$
  - Where F in P: i.e., language $\{ x \mid F(x) \}$ is in P
  - And $|u_1| < poly(|x|)$ for some polynomial
- Recall co-NP
  - Class of languages $\{ x \mid \forall u_1 \; F(x,u_1) \}$
- How about languages $\{ x \mid \exists u_1 \forall u_2 \ldots Q u_k \; F(x,u_1,u_2,\ldots,u_k) \}$
  - Such languages in PSPACE: same way TQBF is (Recall?)

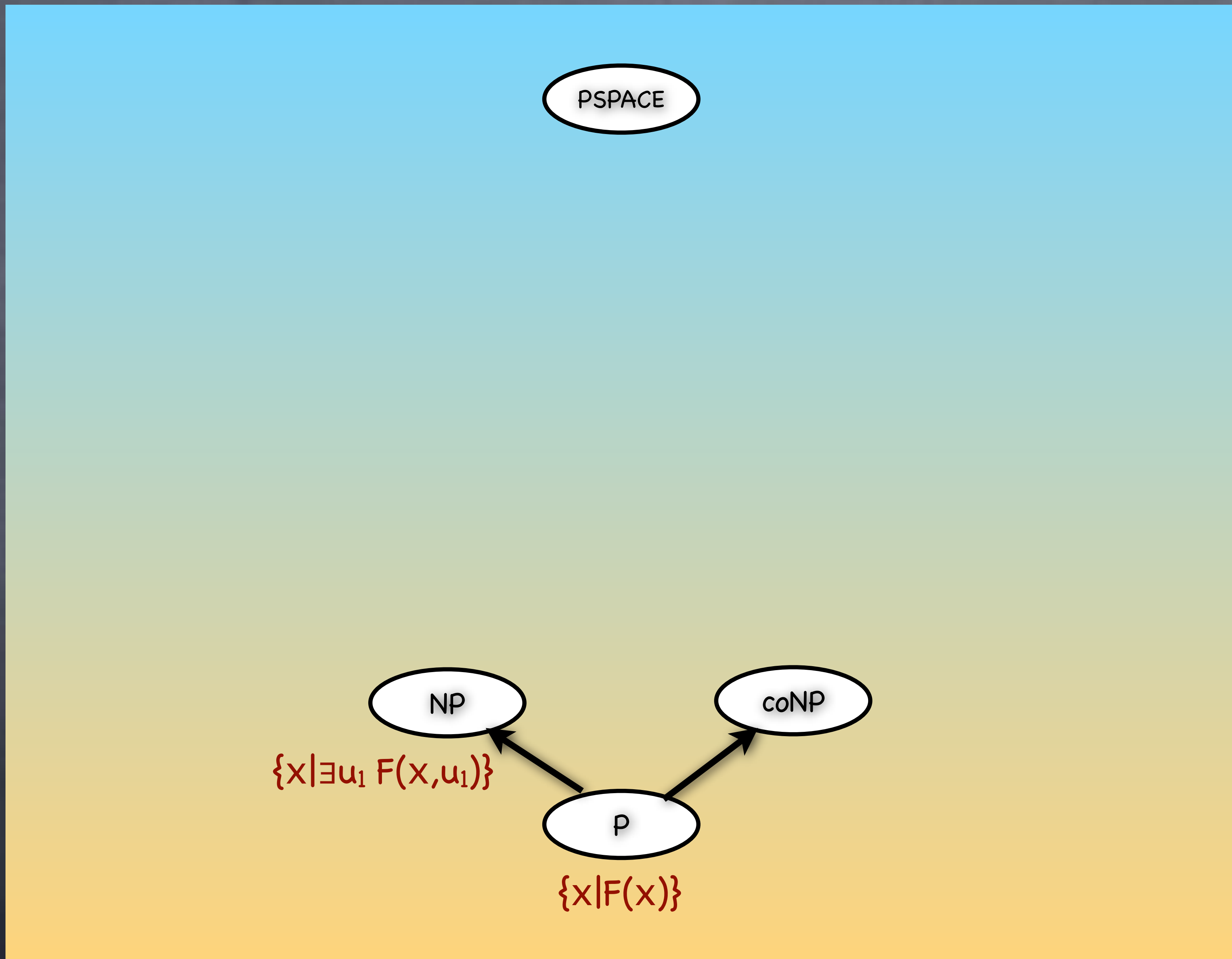
Between P and PSPACE



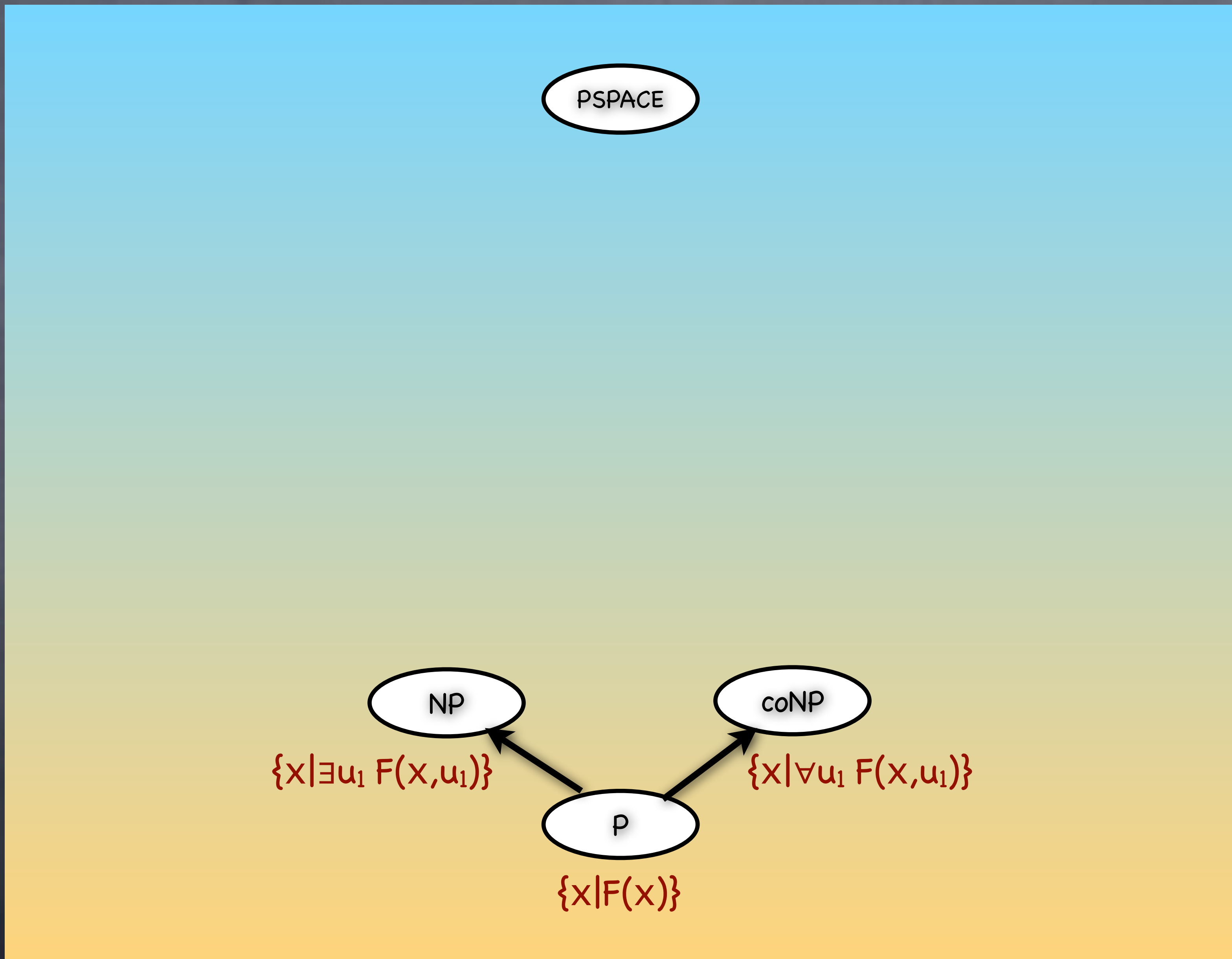
Between P and PSPACE



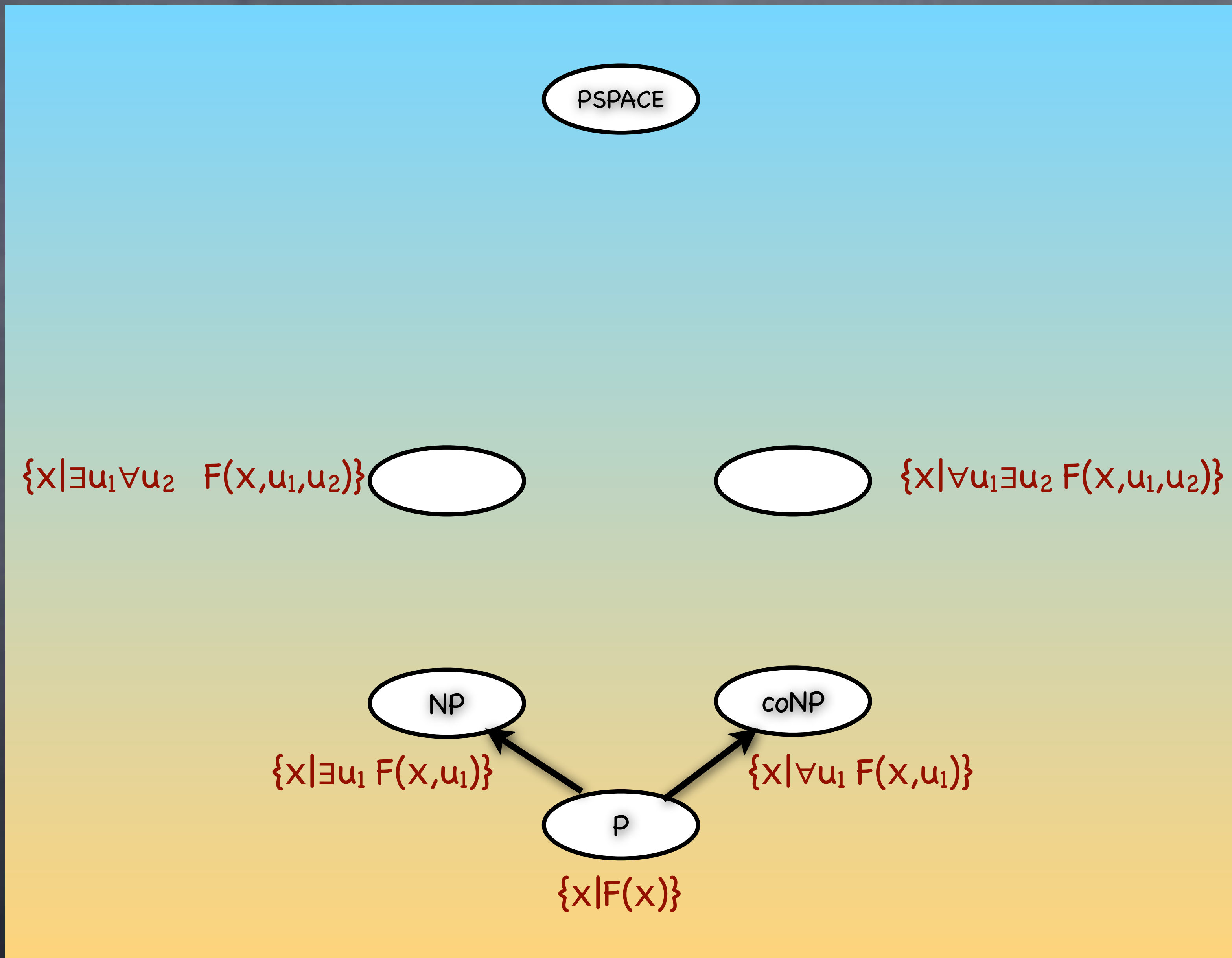
Between P and PSPACE



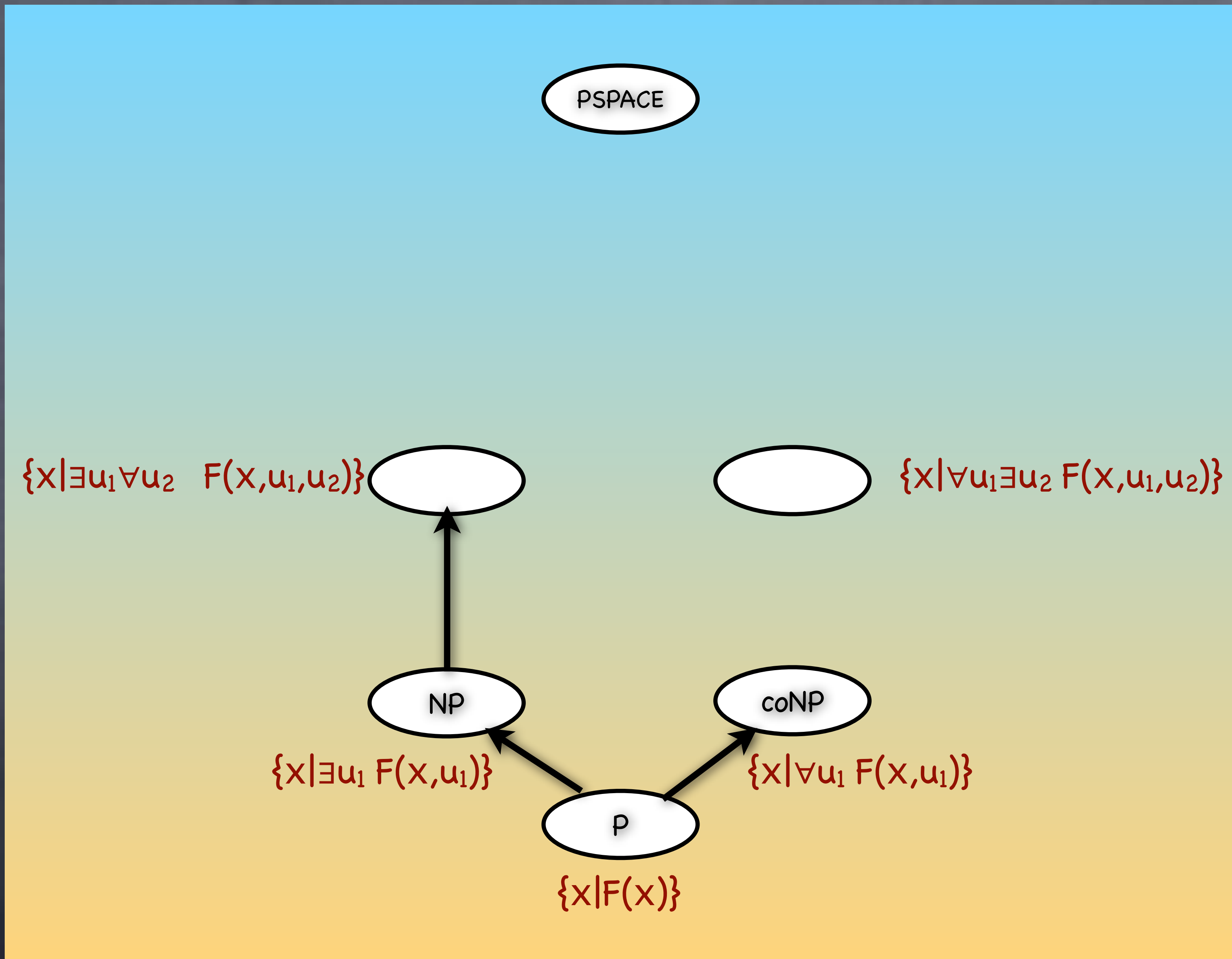
Between P and PSPACE



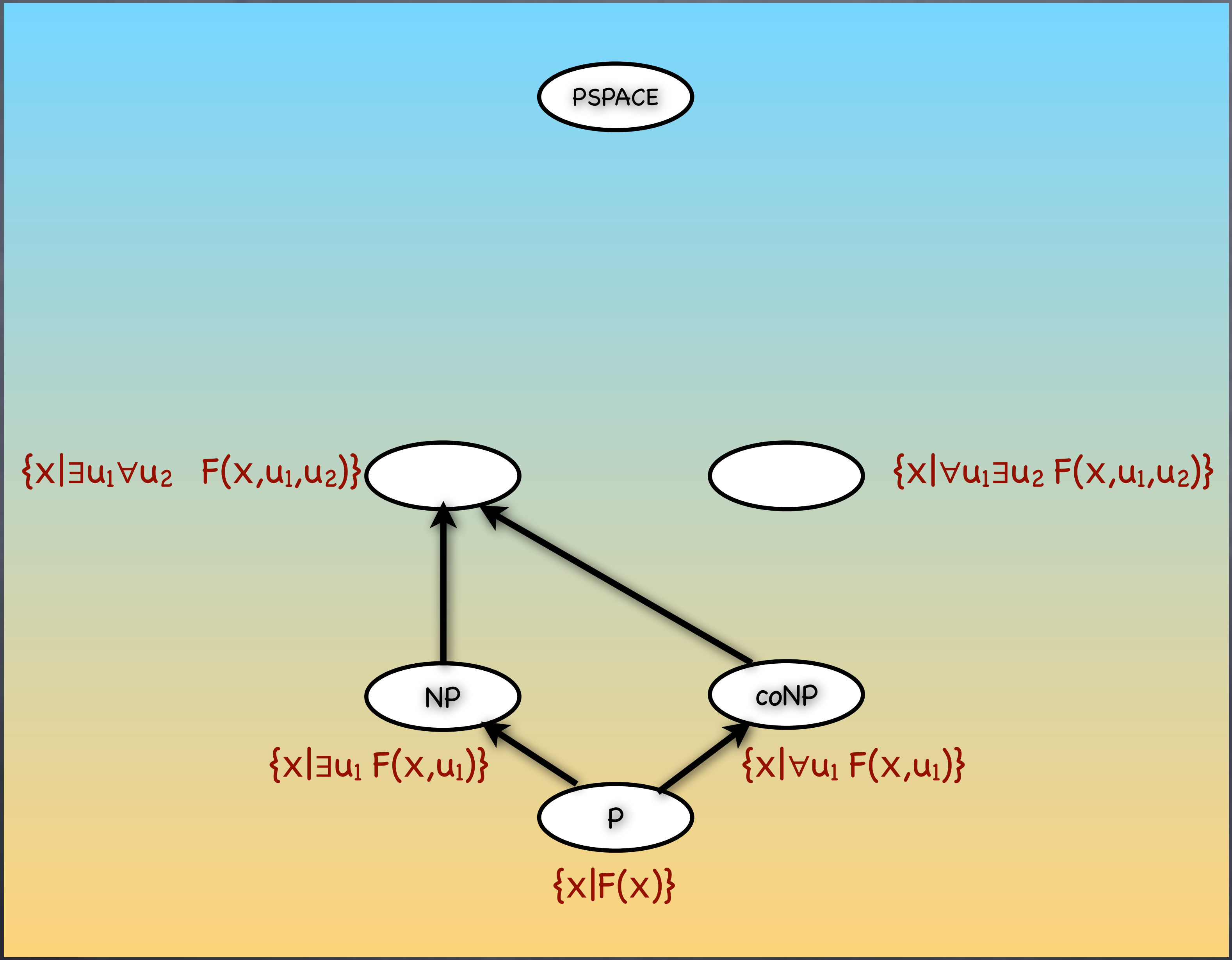
Between P and PSPACE



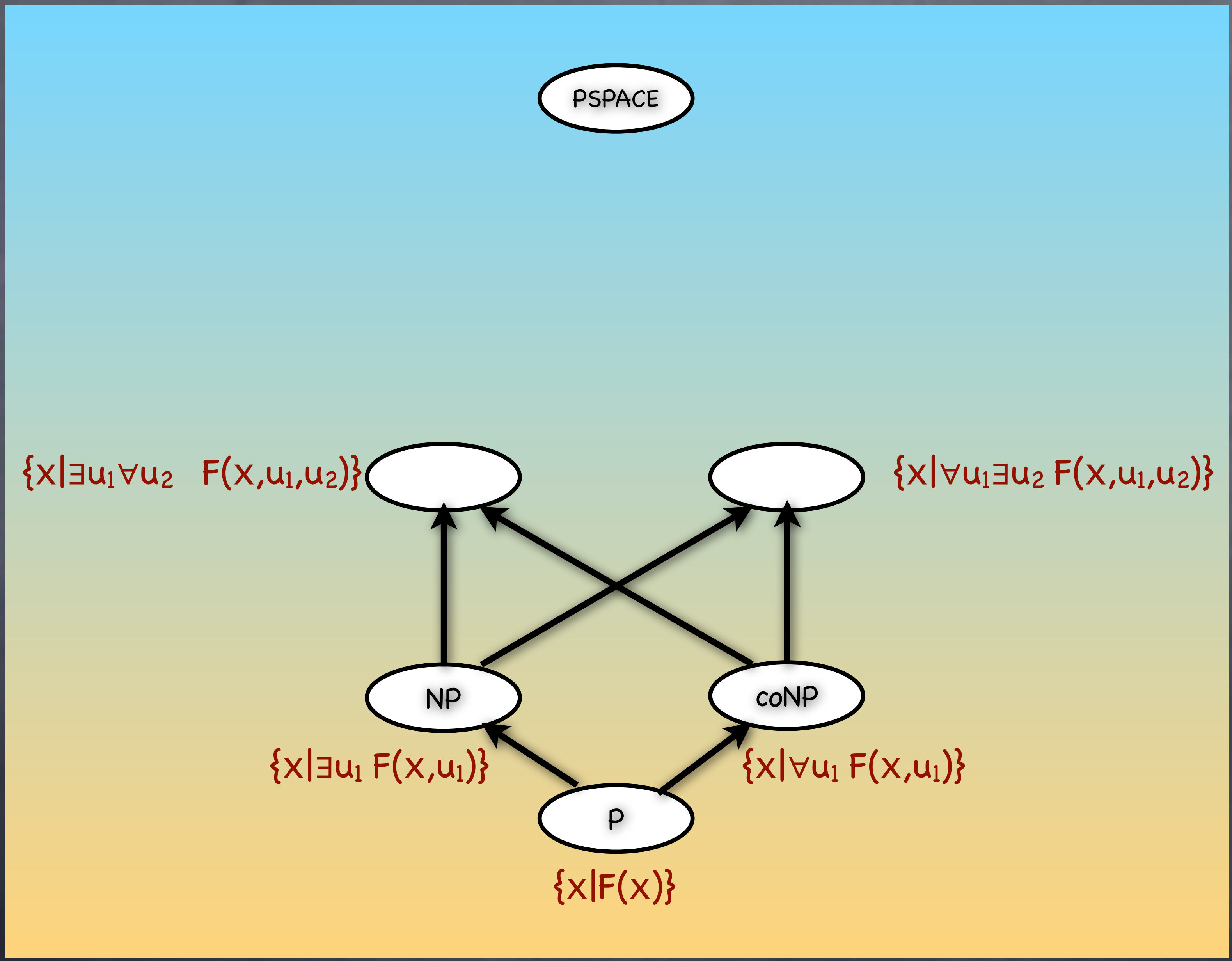
Between P and PSPACE



Between P and PSPACE



Between P and PSPACE



Σ_2^P

Σ_2^P

- Class of languages $\{ x \mid \exists u_1 \forall u_2\ F(x,u_1,u_2) \}$

Σ_2^P

- Class of languages $\{ x \mid \exists u_1 \forall u_2 \; F(x,u_1,u_2) \}$
    - F in P and $|u_i|$ = poly($|x|$)

Σ_2^P

- Class of languages $\{ x \mid \exists u_1 \forall u_2\ F(x,u_1,u_2) \}$
  - F in P and $|u_i| = poly(|x|)$
- e.g.: Two move QBF games, Alice moving first

Σ_2^P

- Class of languages $\{ x \mid \exists u_1 \forall u_2 \; F(x,u_1,u_2) \}$
  - F in P and $|u_i|$ = poly(|x|)
- e.g.: Two move QBF games, Alice moving first
  - Does Alice have a move such that for all moves of the adversary, Alice wins

Σ_2^P

- Class of languages $\{ x \mid \exists u_1 \forall u_2\ F(x,u_1,u_2) \}$
  - F in P and $|u_i| = \text{poly}(|x|)$
- e.g.: Two move QBF games, Alice moving first
  - Does Alice have a move such that for all moves of the adversary, Alice wins
  - Is "$\exists u_1\ \forall u_2\ \varphi(u_1,u_2)$" true?

Σ_2^P

- Class of languages $\{ x \mid \exists u_1 \forall u_2\ F(x,u_1,u_2) \}$
  - F in P and $|u_i| = poly(|x|)$
- e.g.: Two move QBF games, Alice moving first
  - Does Alice have a move such that for all moves of the adversary, Alice wins
  - Is "$\exists u_1\ \forall u_2\ \varphi(u_1,u_2)$" true?

$F(\varphi,u_1,u_2) = \varphi(u_1,u_2)$

Σ_2^P

- Class of languages $\{ x \mid \exists u_1 \forall u_2\ F(x,u_1,u_2) \}$
  - F in P and $|u_i| = \text{poly}(|x|)$
- e.g.: Two move QBF games, Alice moving first
  - Does Alice have a move such that for all moves of the adversary, Alice wins
  - Is "$\exists u_1\ \forall u_2\ \varphi(u_1,u_2)$" true?
  - Seems inherently more complex than deciding $\exists u_1\ \varphi(u_1)$ or $\forall u_1\ \varphi(u_1)$

$F(\varphi,u_1,u_2) = \varphi(u_1,u_2)$

Σ_2^P

Σ_2^P

- Another example: EXACT-CLIQUE

Σ_2^P

- Another example: EXACT-CLIQUE
  - EXACT-CLIQUE = { (G,k) | largest clique in G is of size k(n) }

Σ_2^P

- Another example: EXACT-CLIQUE
  - EXACT-CLIQUE = { (G,k) | largest clique in G is of size k(n) }
    - G has a clique of size k

Σ_2^P

- Another example: EXACT-CLIQUE
  - EXACT-CLIQUE = { (G,k) | largest clique in G is of size k(n) }
    - G has a clique of size k
    - And all cliques in G have size at most k

Σ_2^P

- Another example: EXACT-CLIQUE
  - EXACT-CLIQUE = { (G,k) | largest clique in G is of size k(n) }
    - G has a clique of size k
    - And all cliques in G have size at most k
  - $\exists G_1 \forall G_2$ G_1 is a clique in G of size k and if G_2 is a clique in G, it is of size at most k

Π_2^P

Π_2^P

- $\Pi_2^P = \text{co-}\Sigma_2^P$

Π_2^P

- $\Pi_2^P = \text{co-}\Sigma_2^P$
  - $\{ x \mid \neg (\exists u_1 \forall u_2 \; F'(x,u_1,u_2)) \}$

Π_2^P

- $\Pi_2^P = \text{co-}\Sigma_2^P$
  - $\{ x \mid \neg (\exists u_1 \forall u_2\ F'(x,u_1,u_2)) \}$
  - $\{ x \mid \forall u_1 \exists u_2\ F(x,u_1,u_2)) \}$

Π_2^P

- $\Pi_2^P = \text{co-}\Sigma_2^P$
  - $\{ x \mid \neg (\exists u_1 \forall u_2 \, F'(x,u_1,u_2)) \}$
  - $\{ x \mid \forall u_1 \exists u_2 \, F(x,u_1,u_2)) \}$
- e.g.: Two-move QBF game, Alice moving second

Π_2^P

- $\Pi_2^P = \text{co-}\Sigma_2^P$
  - $\{ x \mid \neg (\exists u_1 \forall u_2 \ F'(x,u_1,u_2)) \}$
  - $\{ x \mid \forall u_1 \exists u_2 \ F(x,u_1,u_2) \}$
- e.g.: Two-move QBF game, Alice moving second
- EXACT-CLIQUE (again!)

Π_2^P

- $\Pi_2^P = \text{co-}\Sigma_2^P$
  - $\{ x \mid \neg (\exists u_1 \forall u_2\ F'(x,u_1,u_2)) \}$
  - $\{ x \mid \forall u_1 \exists u_2\ F(x,u_1,u_2)) \}$
- e.g.: Two-move QBF game, Alice moving second
- EXACT-CLIQUE (again!)
  - ($\exists G_1$ G_1 is a clique in G of size k) and ($\forall G_2$ if G_2 is a clique in G, it is of size at most k)

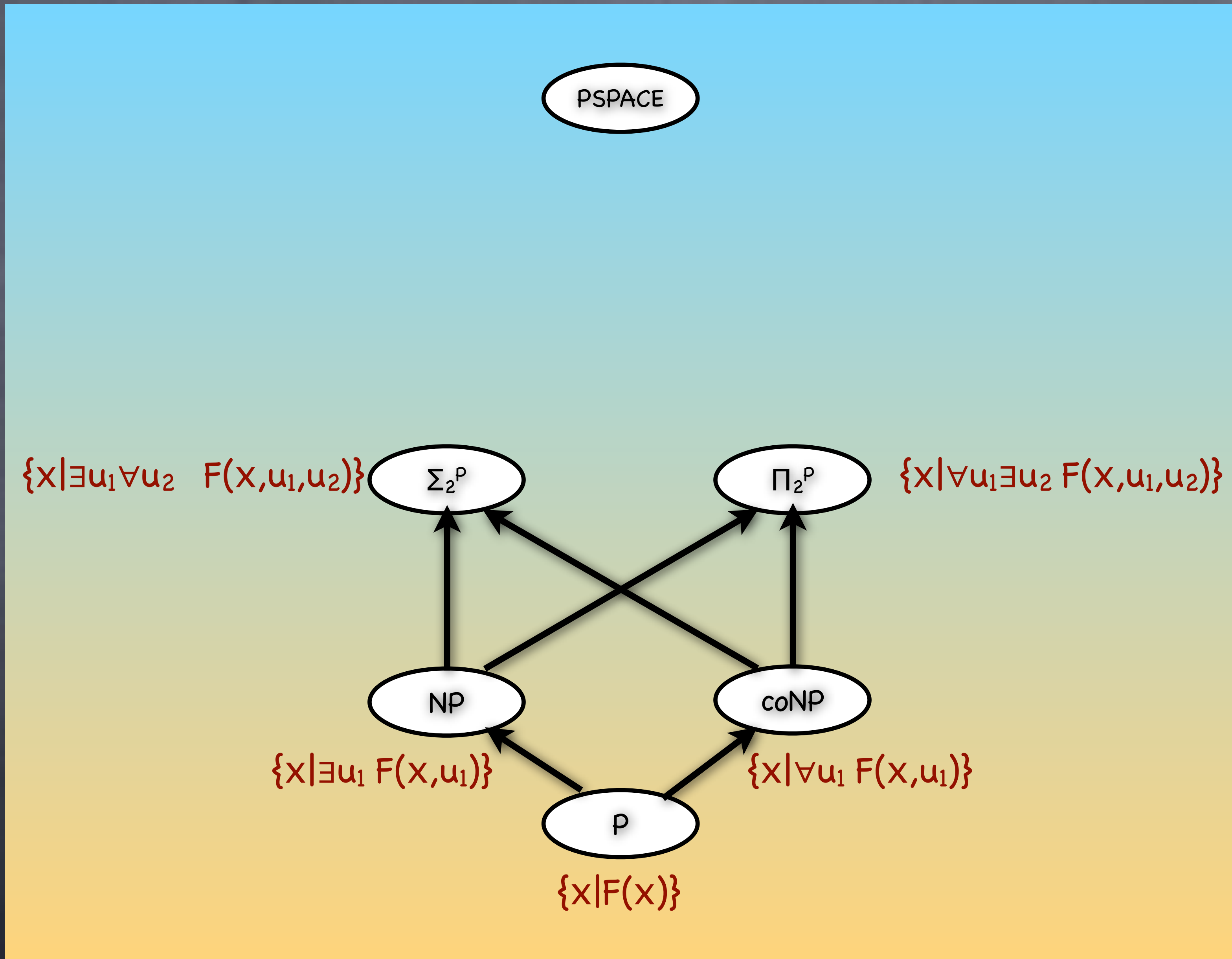
Π_2^P

- $\Pi_2^P = \text{co-}\Sigma_2^P$
  - $\{ x \mid \neg (\exists u_1 \forall u_2 \; F'(x,u_1,u_2)) \}$
  - $\{ x \mid \forall u_1 \exists u_2 \; F(x,u_1,u_2)) \}$
- e.g.: Two-move QBF game, Alice moving second
- EXACT-CLIQUE (again!)
  - ($\exists G_1$ G_1 is a clique in G of size k) and ($\forall G_2$ if G_2 is a clique in G, it is of size at most k)
    - In fact, two prenex form representations with one quantifier each

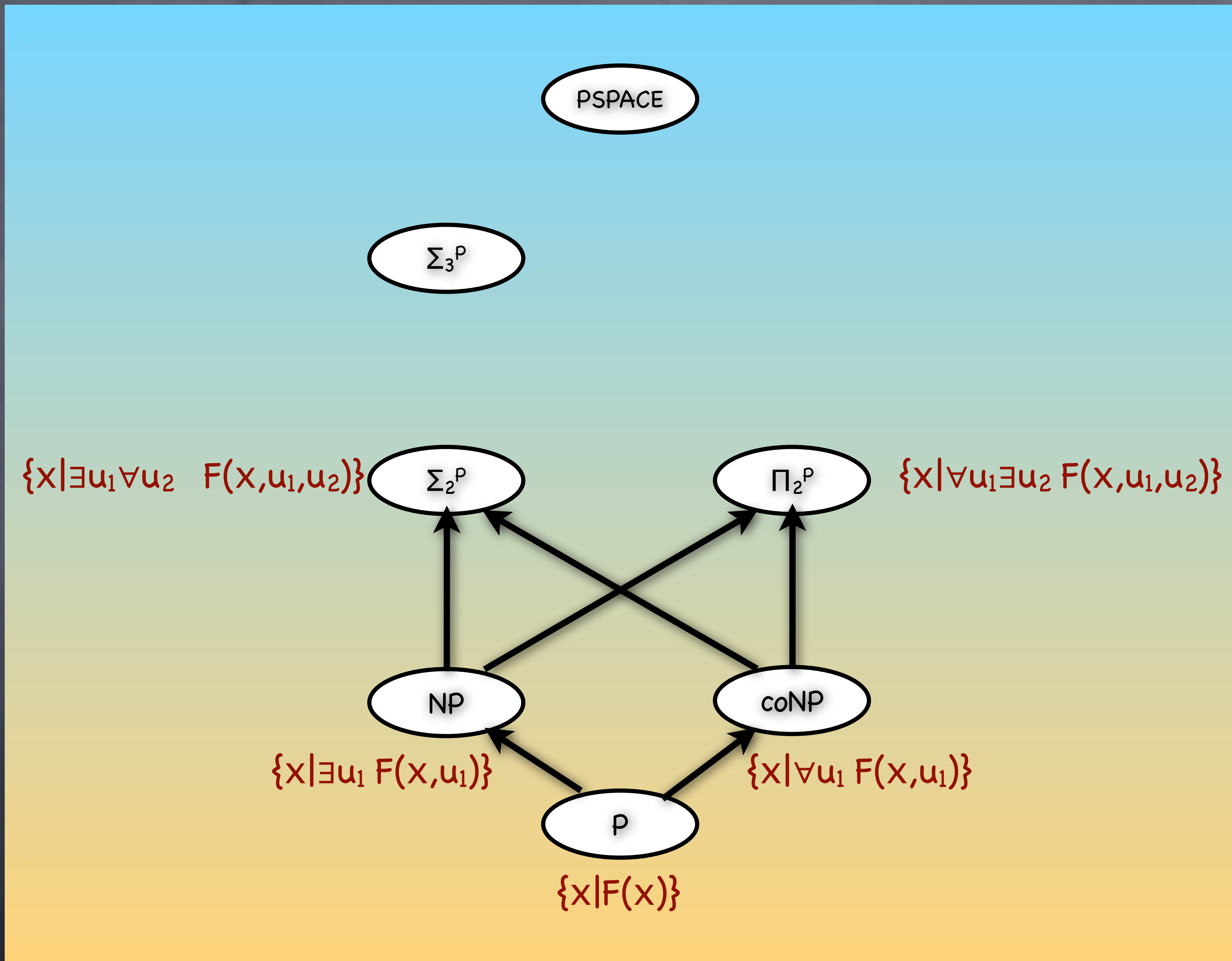
Π_2^P

- Π_2^P = co-Σ_2^P
  - { x | ¬ ($\exists u_1 \forall u_2$ F′(x,u_1,u_2)) }
  - { x | $\forall u_1 \exists u_2$ F(x,u_1,u_2)) }
- e.g.: Two-move QBF game, Alice moving second
- EXACT-CLIQUE (again!)
  - ($\exists G_1$ G_1 is a clique in G of size k) and ($\forall G_2$ if G_2 is a clique in G, it is of size at most k)
    - In fact, two prenex form representations with one quantifier each
- MIN-CKT = { C| $\forall C' \exists x$ C′(x)≠C(x) or C′=C or |C′|>|C| }

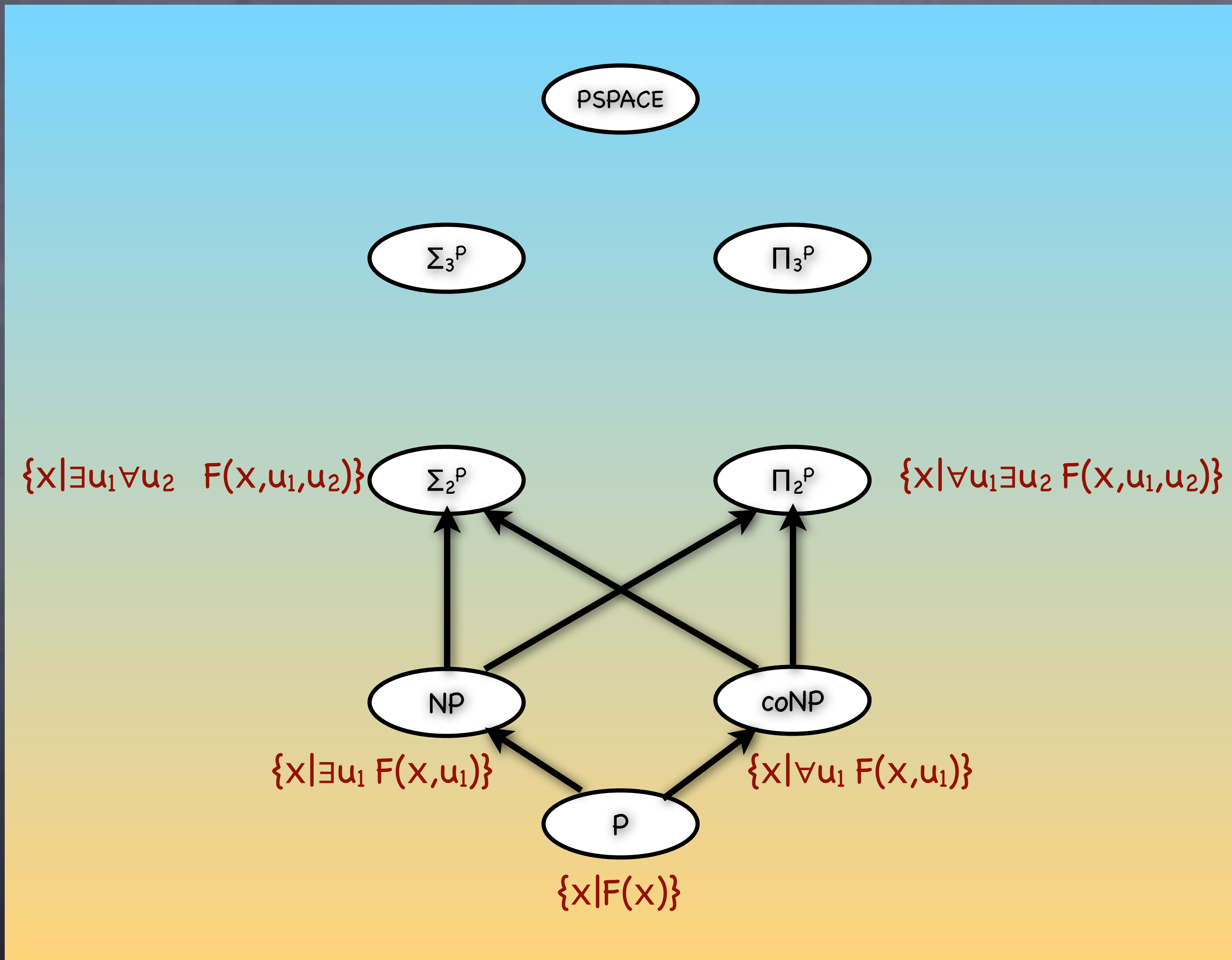
Between P and PSPACE



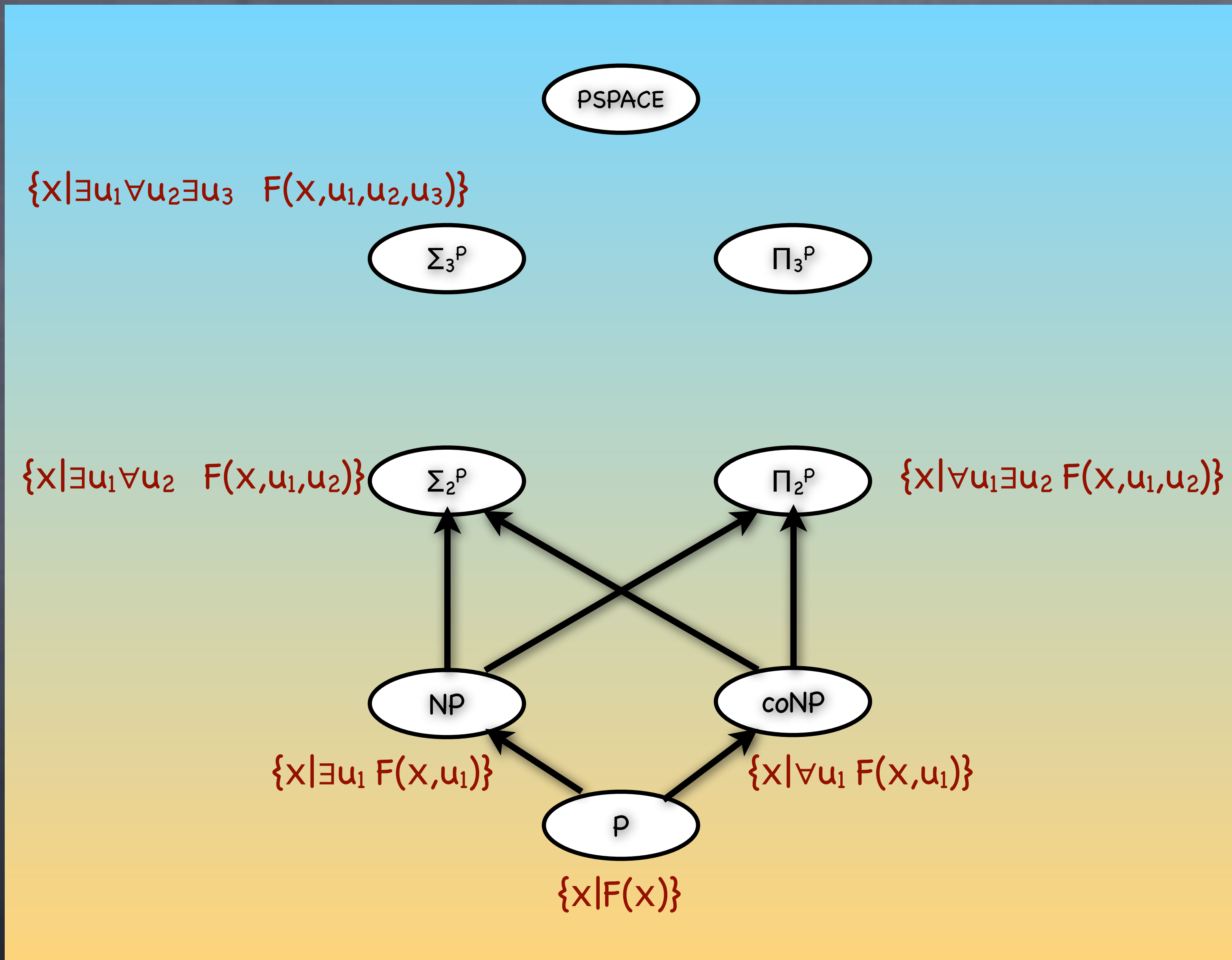
Between P and PSPACE



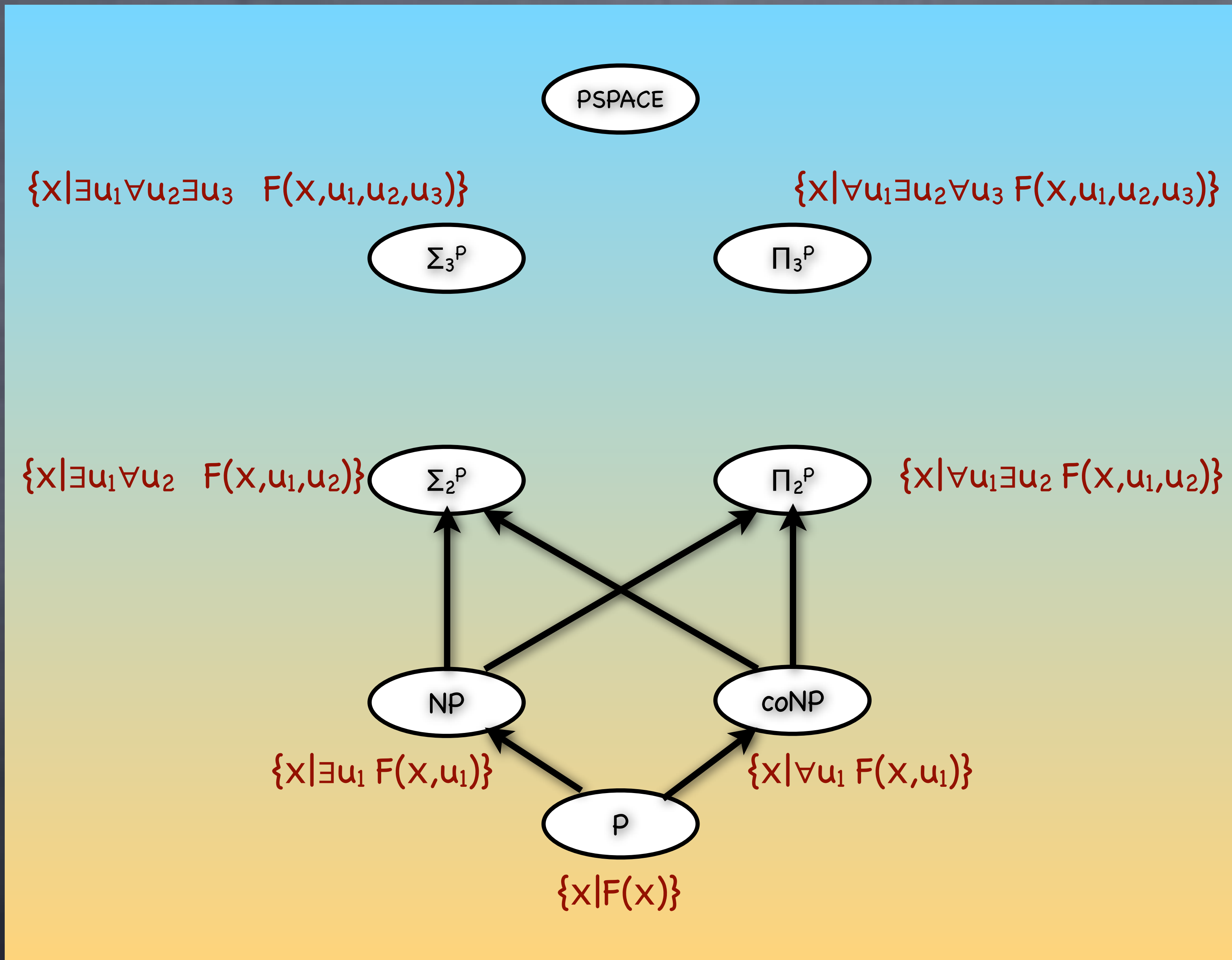
Between P and PSPACE



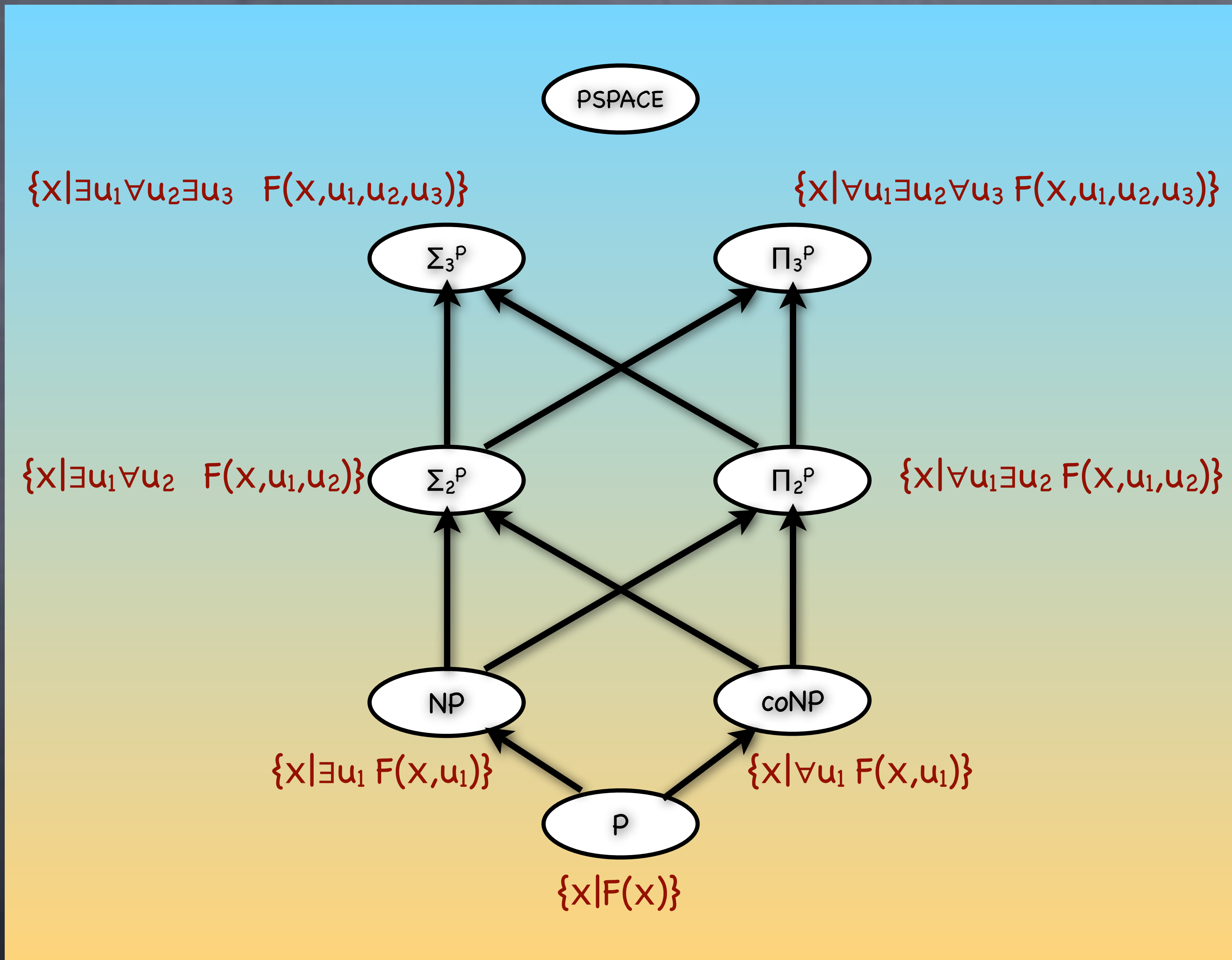
Between P and PSPACE



Between P and PSPACE



Between P and PSPACE



Σ_k^P and Π_k^P

Σ_k^P and Π_k^P

- Σ_k^P : Class of languages $\{ x \mid \exists u_1 \forall u_2 \ldots Q u_k \; F(x,u_1,u_2,\ldots,u_k) \}$

Σ_k^P and Π_k^P

∃ or ∀ based on k odd/even

- Σ_k^P : Class of languages $\{ x \mid \exists u_1 \forall u_2 \ldots Q u_k\ F(x,u_1,u_2,\ldots,u_k) \}$

Σ_k^P and Π_k^P

$\exists$ or $\forall$ based on k odd/even

- Σ_k^P : Class of languages $\{ x \mid \exists u_1 \forall u_2 \ldots Q u_k\ F(x,u_1,u_2,\ldots,u_k) \}$
- Π_k^P : Class of languages $\{ x \mid \forall u_1 \exists u_2 \ldots Q u_k\ F(x,u_1,u_2,\ldots,u_k) \}$

Σ_k^P and Π_k^P

∃ or ∀ based on k odd/even

- Σ_k^P : Class of languages $\{ x \mid \exists u_1 \forall u_2 \ldots Q u_k \; F(x,u_1,u_2,\ldots,u_k) \}$
- Π_k^P : Class of languages $\{ x \mid \forall u_1 \exists u_2 \ldots Q u_k \; F(x,u_1,u_2,\ldots,u_k) \}$
  - F in P and $|u_i| = \text{poly}(|x|)$

Σ_k^P and Π_k^P

$\exists$ or $\forall$ based on k odd/even

- Σ_k^P : Class of languages $\{ x \mid \exists u_1 \forall u_2 \ldots Q u_k \; F(x,u_1,u_2,\ldots,u_k) \}$
- Π_k^P : Class of languages $\{ x \mid \forall u_1 \exists u_2 \ldots Q u_k \; F(x,u_1,u_2,\ldots,u_k) \}$
  - F in P and $|u_i| = \text{poly}(|x|)$
- k alternating quantifiers

Σ_k^P and Π_k^P

∃ or ∀ based on k odd/even

- Σ_k^P : Class of languages $\{ x \mid \exists u_1 \forall u_2 \ldots Q u_k \; F(x,u_1,u_2,\ldots,u_k) \}$
- Π_k^P : Class of languages $\{ x \mid \forall u_1 \exists u_2 \ldots Q u_k \; F(x,u_1,u_2,\ldots,u_k) \}$
  - F in P and $|u_i| = \text{poly}(|x|)$
- k alternating quantifiers
  - Note: can have more than k boolean variables, each u_i being a string of boolean variables

Σ_k^P and Π_k^P

∃ or ∀ based on k odd/even

- Σ_k^P : Class of languages $\{ x \mid \exists u_1 \forall u_2 \ldots Q u_k\ F(x,u_1,u_2,\ldots,u_k) \}$
- Π_k^P : Class of languages $\{ x \mid \forall u_1 \exists u_2 \ldots Q u_k\ F(x,u_1,u_2,\ldots,u_k) \}$
  - F in P and $|u_i|$ = poly($|x|$)
- k alternating quantifiers
  - Note: can have more than k boolean variables, each u_i being a string of boolean variables
  - k constant (independent of input size) for all languages, but number of variables need not be

Σ_k^P and Π_k^P

∃ or ∀ based on k odd/even

- Σ_k^P : Class of languages $\{ x \mid \exists u_1 \forall u_2 \ldots Q u_k\ F(x,u_1,u_2,\ldots,u_k) \}$
- Π_k^P : Class of languages $\{ x \mid \forall u_1 \exists u_2 \ldots Q u_k\ F(x,u_1,u_2,\ldots,u_k) \}$
  - F in P and $|u_i|$ = poly(|x|)
- k alternating quantifiers
  - Note: can have more than k boolean variables, each u_i being a string of boolean variables
  - k constant (independent of input size) for all languages, but number of variables need not be
- P = Σ_0^P = Π_0^P , NP = Σ_1^P and co-NP = Π_1^P

Σ_k^P and Π_k^P

Σ_k^P and Π_k^P

- Σ_k^P has languages of the form

Σ_k^P and Π_k^P

- Σ_k^P has languages of the form
  - $L = \{ x \mid \exists u_1 \forall u_2 \ldots Q_k u_k \ (x,u_1,u_2,\ldots,u_k) \in F \}$, where F in P (and $|u_i| = \text{poly}(|x|)$)

Σ_k^P and Π_k^P

- Σ_k^P has languages of the form
  - $L = \{ x \mid \exists u_1 \forall u_2 \ldots Q_k u_k \ (x,u_1,u_2,\ldots,u_k) \in F \}$, where F in P (and $|u_i| = poly(|x|)$)
  - $L = \{ x \mid \exists u_1 \ (x,u_1) \in L' \}$ where L' in Π_{k-1}^P

Σ_k^P and Π_k^P

- Σ_k^P has languages of the form
  - $L = \{ x \mid \exists u_1 \forall u_2 \ldots Q_k u_k\ (x,u_1,u_2,\ldots,u_k) \in F \}$, where F in P (and $|u_i| = \text{poly}(|x|)$)
  - $L = \{ x \mid \exists u_1\ (x,u_1) \in L' \}$ where L' in Π_{k-1}^P
  - Or more generally, in terms of k-i^th^ level

Σ_k^P and Π_k^P

- Σ_k^P has languages of the form
  - $L = \{ x \mid \exists u_1 \forall u_2 \ldots Q_k u_k \ (x,u_1,u_2,\ldots,u_k) \in F \}$, where F in P (and $|u_i| = \text{poly}(|x|)$)
  - $L = \{ x \mid \exists u_1 \ (x,u_1) \in L' \}$ where L' in Π_{k-1}^P
  - Or more generally, in terms of k-i[th] level
    - $L = \{ x \mid \exists u_1 \forall u_2 \ldots Q_i u_i \ (x,u_1,u_2,\ldots,u_i) \in L' \}$ where L' in Π_{k-i}^P (odd i) or Σ_{k-i}^P (even i)

Σ_k^P and Π_k^P

- Σ_k^P has languages of the form
  - $L = \{ x \mid \exists u_1 \forall u_2 \ldots Q_k u_k\ (x,u_1,u_2,\ldots,u_k) \in F \}$, where F in P (and $|u_i| = poly(|x|)$)
  - $L = \{ x \mid \exists u_1\ (x,u_1) \in L' \}$ where L' in Π_{k-1}^P
  - Or more generally, in terms of k-i[th] level
    - $L = \{ x \mid \exists u_1 \forall u_2 \ldots Q_i u_i\ (x,u_1,u_2,\ldots,u_i) \in L' \}$ where L' in Π_{k-i}^P (odd i) or Σ_{k-i}^P (even i)
- Similarly, Π_k^P has languages of the form

Σ_k^P and Π_k^P

- Σ_k^P has languages of the form
  - $L = \{ x \mid \exists u_1 \forall u_2 \ldots Q_k u_k\ (x,u_1,u_2,\ldots,u_k) \in F \}$, where F in P (and $|u_i| = \text{poly}(|x|)$)
  - $L = \{ x \mid \exists u_1\ (x,u_1) \in L' \}$ where L' in Π_{k-1}^P
  - Or more generally, in terms of k-i[th] level
    - $L = \{ x \mid \exists u_1 \forall u_2 \ldots Q_i u_i\ (x,u_1,u_2,\ldots,u_i) \in L' \}$ where L' in Π_{k-i}^P (odd i) or Σ_{k-i}^P (even i)
- Similarly, Π_k^P has languages of the form
  - $L = \{ x \mid \forall u_1 \exists u_2 \ldots Q_i u_i\ (x,u_1,u_2,\ldots,u_i) \in L' \}$ where L' in Σ_{k-i}^P (odd i) or Π_{k-i}^P (even i)

PH

PH

- $PH = \bigcup_{k>0} \Sigma_k^P = \bigcup_{k>0} \Pi_k^P$

PH

- PH = $\bigcup_{k>0} \Sigma_k^P$ = $\bigcup_{k>0} \Pi_k^P$
- PH ⊆ PSPACE

PH

- $PH = \bigcup_{k>0} \Sigma_k^P = \bigcup_{k>0} \Pi_k^P$
- $PH \subseteq PSPACE$
  - Because for each k, $\Sigma_k^P \subseteq PSPACE$

PH

- PH = $\bigcup_{k>0} \Sigma_k^P$ = $\bigcup_{k>0} \Pi_k^P$
- PH $\subseteq$ PSPACE
  - Because for each k, $\Sigma_k^P \subseteq$ PSPACE
  - We don't know if PH $\subsetneq$ PSPACE

PH

- PH = $\bigcup_{k>0} \Sigma_k^P$ = $\bigcup_{k>0} \Pi_k^P$
- PH $\subseteq$ PSPACE
  - Because for each k, $\Sigma_k^P \subseteq$ PSPACE
  - We don't know if PH $\subsetneq$ PSPACE
- We don't know if P $\subsetneq$ PH (or P $\subsetneq$ PSPACE)

PH

- PH = $\bigcup_{k>0} \Sigma_k^P$ = $\bigcup_{k>0} \Pi_k^P$
- PH ⊆ PSPACE
  - Because for each k, $\Sigma_k^P \subseteq$ PSPACE
  - We don't know if PH ⊊ PSPACE
- We don't know if P ⊊ PH (or P ⊊ PSPACE)
  - Believed that $\Sigma_k^P \subsetneq \Sigma_{k+1}^P$ and $\Pi_k^P \subsetneq \Pi_{k+1}^P$ for all k

Complete problems

Complete problems

- For each level of PH (w.r.t Karp reductions)

Complete problems

- For each level of PH (w.r.t Karp reductions)
  - Σ_kSAT: True QBFs with k alternations, starting with ∃

Complete problems

- For each level of PH (w.r.t Karp reductions)
  - Σ_kSAT: True QBFs with k alternations, starting with ∃
    - Complete for Σ_k^P

Complete problems

- For each level of PH (w.r.t Karp reductions)
  - Σ_kSAT: True QBFs with k alternations, starting with $\exists$
    - Complete for Σ_k^P
  - Π_kSAT: True QBFs with k alternations, starting with $\forall$

Complete problems

- For each level of PH (w.r.t Karp reductions)
  - Σ_kSAT: True QBFs with k alternations, starting with $\exists$
    - Complete for Σ_k^P
  - Π_kSAT: True QBFs with k alternations, starting with $\forall$
    - Complete for Π_k^P

Complete problems

- For each level of PH (w.r.t Karp reductions)
  - Σ_kSAT: True QBFs with k alternations, starting with ∃
    - Complete for Σ_k^P
  - Π_kSAT: True QBFs with k alternations, starting with ∀
    - Complete for Π_k^P
  - Why? Consider odd k Σ_k^P and even k Π_k^P (ends with ∃)

Complete problems

- For each level of PH (w.r.t Karp reductions)
  - Σ_kSAT: True QBFs with k alternations, starting with ∃
    - Complete for Σ_k^P
  - Π_kSAT: True QBFs with k alternations, starting with ∀
    - Complete for Π_k^P
  - Why? Consider odd k Σ_k^P and even k Π_k^P (ends with ∃)
    - Recall: $F(X)=1$ iff $CKT_F(X)=1$ iff $\exists w\ \varphi_F(X;w)=1$

Complete problems

- For each level of PH (w.r.t Karp reductions)
  - Σ_kSAT: True QBFs with k alternations, starting with $\exists$
    - Complete for Σ_k^P
  - Π_kSAT: True QBFs with k alternations, starting with $\forall$
    - Complete for Π_k^P
  - Why? Consider odd k Σ_k^P and even k Π_k^P (ends with $\exists$)
    - Recall: $F(X)=1$ iff $CKT_F(X)=1$ iff $\exists w\ \varphi_F(X;w)=1$

Needed a $\exists$ in going from ckt to CNF formula

Complete problems

- For each level of PH (w.r.t Karp reductions)
  - Σ_kSAT: True QBFs with k alternations, starting with $\exists$
    - Complete for Σ_k^P
  - Π_kSAT: True QBFs with k alternations, starting with $\forall$
    - Complete for Π_k^P
  - Why? Consider odd k Σ_k^P and even k Π_k^P (ends with $\exists$)
    - Recall: $F(X)=1$ iff $CKT_F(X)=1$ iff $\exists w\ \varphi_F(X;w)=1$
    - $Qu_1\ldots\exists u_k\ F(\ldots,u_k)$ true iff $Qu_1\ldots\exists u_k,w\ \varphi_F(\ldots,u_k,w)$ true

Needed a $\exists$ in going from ckt to CNF formula

Complete problems

- For each level of PH (w.r.t Karp reductions)
  - Σ_kSAT: True QBFs with k alternations, starting with ∃
    - Complete for Σ_k^P
  - Π_kSAT: True QBFs with k alternations, starting with ∀
    - Complete for Π_k^P
  - Why? Consider odd k Σ_k^P and even k Π_k^P (ends with ∃)
    - Recall: F(X)=1 iff $CKT_F(X)=1$ iff ∃w $\varphi_F(X;w)=1$
    - $Qu_1 \ldots \exists u_k\ F(\ldots,u_k)$ true iff $Qu_1 \ldots \exists u_k,w\ \varphi_F(\ldots,u_k,w)$ true

For the other classes consider co-classes

Needed a ∃ in going from ckt to CNF formula

Complete problems

Complete problems

- Complete problem for PH?

Complete problems

- Complete problem for PH?
  - Then PH collapses to a finite level

Complete problems

- Complete problem for PH?
  - Then PH collapses to a finite level
  - If L is PH-complete, $L \in \Sigma_k^P$ for some k

Complete problems

- Complete problem for PH?
  - Then PH collapses to a finite level
  - If L is PH-complete, $L \in \Sigma_k^P$ for some k
  - But Σ_k^P downward closed under Karp reductions (Exercise)

Complete problems

- Complete problem for PH?
  - Then PH collapses to a finite level
  - If L is PH-complete, $L \in \Sigma_k^P$ for some k
  - But Σ_k^P downward closed under Karp reductions (Exercise)
  - So PH = Σ_k^P

Complete problems

- Complete problem for PH?
  - Then PH collapses to a finite level
  - If L is PH-complete, $L \in \Sigma_k^P$ for some k
  - But Σ_k^P downward closed under Karp reductions (Exercise)
  - So PH = Σ_k^P
- Corollary: If PH = PSPACE, then PH = PSPACE = Σ_k^P for some k

Complete problems

- Complete problem for PH?
  - Then PH collapses to a finite level
  - If L is PH-complete, $L \in \Sigma_k^P$ for some k
  - But Σ_k^P downward closed under Karp reductions (Exercise)
  - So PH = Σ_k^P
- Corollary: If PH = PSPACE, then PH = PSPACE = Σ_k^P for some k
  - Because if PH = PSPACE, TQBF is PH-complete

Ways PH can collapse

Ways PH can collapse

- Popular belief: All classes are distinct

Ways PH can collapse

- Popular belief: All classes are distinct
- What happens if a level collapses on to the one below it (e.g. NP = P or Σ_2^P = NP)

Ways PH can collapse

- Popular belief: All classes are distinct
- What happens if a level collapses on to the one below it (e.g. NP = P or Σ_2^P = NP)
- What happens if at a level co-classes collapse on to each other (e.g. NP = coNP or $\Sigma_2^P = \Pi_2^P$)

Ways PH can collapse

- Popular belief: All classes are distinct
- What happens if a level collapses on to the one below it (e.g. NP = P or Σ_2^P = NP)
- What happens if at a level co-classes collapse on to each other (e.g. NP = coNP or $\Sigma_2^P = \Pi_2^P$)
- Then entire PH collapses! (to that level)

If $\Sigma_k^P = \Pi_k^P$ (k>0)

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- If $\Sigma_k^P = \Pi_k^P$ for some k>0 then $PH = \Sigma_k^P = \Pi_k^P$

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- **If $\Sigma_k^P = \Pi_k^P$ for some k>0 then PH = $\Sigma_k^P = \Pi_k^P$**
  - e.g. If NP = co-NP, then PH = NP

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- **If $\Sigma_k^P = \Pi_k^P$ for some k>0 then PH = $\Sigma_k^P = \Pi_k^P$**
  - e.g. If NP = co-NP, then PH = NP
- Will show that $\Sigma_k^P = \Pi_k^P$ (for k>0) $\Rightarrow \Sigma_{k+1}^P = \Pi_{k+1}^P = \Sigma_k^P = \Pi_k^P$

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- **If $\Sigma_k^P = \Pi_k^P$ for some k>0 then PH = $\Sigma_k^P = \Pi_k^P$**
  - e.g. If NP = co-NP, then PH = NP
- Will show that $\Sigma_k^P = \Pi_k^P$ (for k>0) $\Rightarrow \Sigma_{k+1}^P = \Pi_{k+1}^P = \Sigma_k^P = \Pi_k^P$
  - By induction PH = $\Sigma_k^P = \Pi_k^P$

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- **If $\Sigma_k^P = \Pi_k^P$ for some k>0 then PH = $\Sigma_k^P = \Pi_k^P$**
  - e.g. If NP = co-NP, then PH = NP
- Will show that $\Sigma_k^P = \Pi_k^P$ (for k>0) $\Rightarrow \Sigma_{k+1}^P = \Pi_{k+1}^P = \Sigma_k^P = \Pi_k^P$
  - By induction PH = $\Sigma_k^P = \Pi_k^P$
  - Enough to show $\Sigma_k^P = \Pi_k^P$ (for k>0) $\Rightarrow \Sigma_{k+1}^P \subseteq \Sigma_k^P$

If $\Sigma_k^P = \Pi_k^P$ (k>0)

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- Consider L in Σ_{k+1}^P

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- Consider L in Σ_{k+1}^P
  - $L = \{x \mid \exists u_1 \forall u_2 \ldots Q_{k+1} u_{k+1}\ F(x,u_1,u_2,\ldots,u_{k+1})\}$

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- Consider L in Σ_{k+1}^P
  - $L = \{x \mid \exists u_1 \forall u_2 \ldots Q_{k+1} u_{k+1}\ F(x,u_1,u_2,\ldots,u_{k+1})\}$
- Define $L' = \{(x,u_1) \mid \forall u_2 \ldots Q_{k+1} u_{k+1}\ F(x,u_1,u_2,\ldots,u_{k+1})\}$

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- Consider L in Σ_{k+1}^P
  - $L = \{x \mid \exists u_1 \forall u_2 \ldots Q_{k+1} u_{k+1}\ F(x,u_1,u_2,\ldots,u_{k+1})\}$
- Define $L' = \{(x,u_1) \mid \forall u_2 \ldots Q_{k+1} u_{k+1}\ F(x,u_1,u_2,\ldots,u_{k+1})\}$
  - $L = \{x \mid \exists u_1\ (x,u_1) \in L'\}$ and L' in Π_k^P

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- Consider L in Σ_{k+1}^P
  - $L = \{x \mid \exists u_1 \forall u_2 \ldots Q_{k+1} u_{k+1}\ F(x,u_1,u_2,\ldots,u_{k+1})\}$
- Define $L' = \{(x,u_1) \mid \forall u_2 \ldots Q_{k+1} u_{k+1}\ F(x,u_1,u_2,\ldots,u_{k+1})\}$
  - $L = \{x \mid \exists u_1\ (x,u_1) \in L'\}$ and L' in Π_k^P
- $\Pi_k^P = \Sigma_k^P \Rightarrow L'$ in Σ_k^P

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- Consider L in Σ_{k+1}^P
  - $L = \{x \mid \exists u_1 \forall u_2 \ldots Q_{k+1} u_{k+1}\ F(x,u_1,u_2,\ldots,u_{k+1})\}$
- Define $L' = \{(x,u_1) \mid \forall u_2 \ldots Q_{k+1} u_{k+1}\ F(x,u_1,u_2,\ldots,u_{k+1})\}$
  - $L = \{x \mid \exists u_1\ (x,u_1) \in L'\}$ and L' in Π_k^P
- $\Pi_k^P = \Sigma_k^P \Rightarrow L'$ in Σ_k^P
  - $\Rightarrow L' = \{(x,u_1) \mid \exists v_2 .. Q'_{k+1} v_{k+1}\ F'(x,u_1,v_2,\ldots,v_{k+1})\}$

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- Consider L in Σ_{k+1}^P
  - $L = \{x \mid \exists u_1 \forall u_2 ... Q_{k+1} u_{k+1}\ F(x,u_1,u_2,...,u_{k+1})\}$
- Define $L' = \{(x,u_1) \mid \forall u_2 ... Q_{k+1} u_{k+1}\ F(x,u_1,u_2,...,u_{k+1})\}$
  - $L = \{x \mid \exists u_1\ (x,u_1) \in L'\}$ and L' in Π_k^P
- $\Pi_k^P = \Sigma_k^P \Rightarrow L'$ in Σ_k^P
  - $\Rightarrow L' = \{(x,u_1) \mid \exists v_2 .. Q'_{k+1} v_{k+1}\ F'(x,u_1,v_2,...,v_{k+1})\}$
  - $\Rightarrow L = \{x \mid \exists u_1 \exists v_2 .. Q'_{k+1} v_{k+1}\ F'(x,u_1,v_2,...,v_{k+1})\}$ in Σ_k^P

If $\Sigma_k^P = \Pi_k^P$ (k>0)

- Consider L in Σ_{k+1}^P
  - $L = \{x \mid \exists u_1 \forall u_2 \ldots Q_{k+1} u_{k+1}\ F(x,u_1,u_2,\ldots,u_{k+1})\}$
- Define $L' = \{(x,u_1) \mid \forall u_2 \ldots Q_{k+1} u_{k+1}\ F(x,u_1,u_2,\ldots,u_{k+1})\}$
  - $L = \{x \mid \exists u_1\ (x,u_1) \in L'\}$ and L' in Π_k^P
- $\Pi_k^P = \Sigma_k^P \Rightarrow L'$ in Σ_k^P
  - $\Rightarrow L' = \{(x,u_1) \mid \exists v_2 .. Q'_{k+1} v_{k+1}\ F'(x,u_1,v_2,\ldots,v_{k+1})\}$
  - $\Rightarrow L = \{x \mid \exists u_1 \exists v_2 .. Q'_{k+1} v_{k+1}\ F'(x,u_1,v_2,\ldots,v_{k+1})\}$ in Σ_k^P
- $\Rightarrow \Sigma_{k+1}^P \subseteq \Sigma_k^P$

If $\Sigma_{k+1}^{P} = \Sigma_{k}^{P}$

If $\Sigma_{k+1}^P = \Sigma_k^P$

- **If $\Sigma_{k+1}^P = \Sigma_k^P$** (equivalently $\Pi_{k+1}^P = \Pi_k^P$) **then PH = Σ_k^P**

If $\Sigma_{k+1}^P = \Sigma_k^P$

- **If $\Sigma_{k+1}^P = \Sigma_k^P$** (equivalently $\Pi_{k+1}^P = \Pi_k^P$) **then PH = Σ_k^P**
- Because then $\Pi_k^P \subseteq \Sigma_{k+1}^P = \Sigma_k^P$ and hence $\Pi_k^P = \Sigma_k^P$

If $\Sigma_{k+1}^P = \Sigma_k^P$

- **If $\Sigma_{k+1}^P = \Sigma_k^P$** (equivalently $\Pi_{k+1}^P = \Pi_k^P$) **then PH = Σ_k^P**
- Because then $\Pi_k^P \subseteq \Sigma_{k+1}^P = \Sigma_k^P$ and hence $\Pi_k^P = \Sigma_k^P$
  - So for k>0, implies PH = Σ_k^P

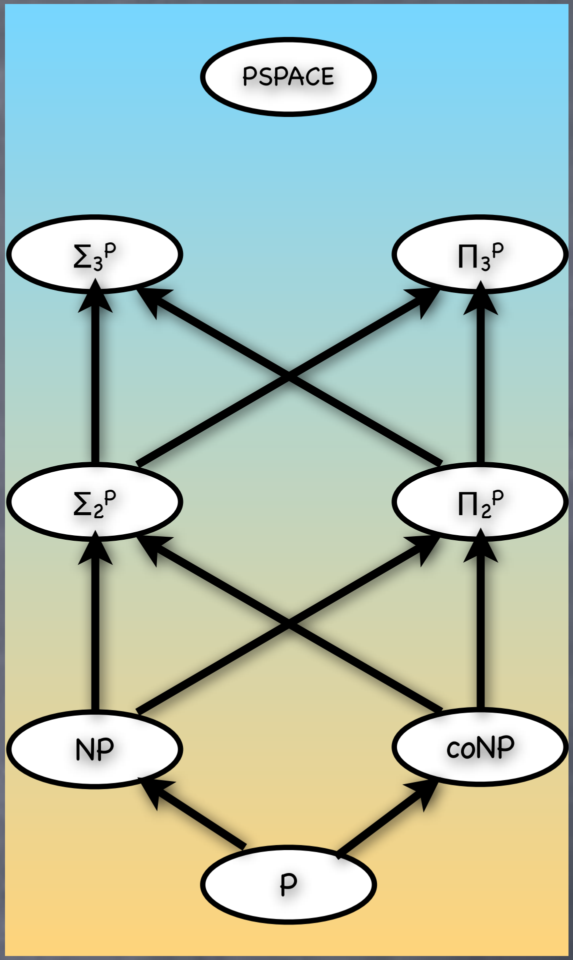
If $\Sigma_{k+1}^P = \Sigma_k^P$

- **If $\Sigma_{k+1}^P = \Sigma_k^P$** (equivalently $\Pi_{k+1}^P = \Pi_k^P$) **then PH = Σ_k^P**
- Because then $\Pi_k^P \subseteq \Sigma_{k+1}^P = \Sigma_k^P$ and hence $\Pi_k^P = \Sigma_k^P$
  - So for k>0, implies PH = Σ_k^P
- Holds for k=0 too: i.e., NP = P $\Rightarrow$ PH = P

If $\Sigma_{k+1}^P = \Sigma_k^P$

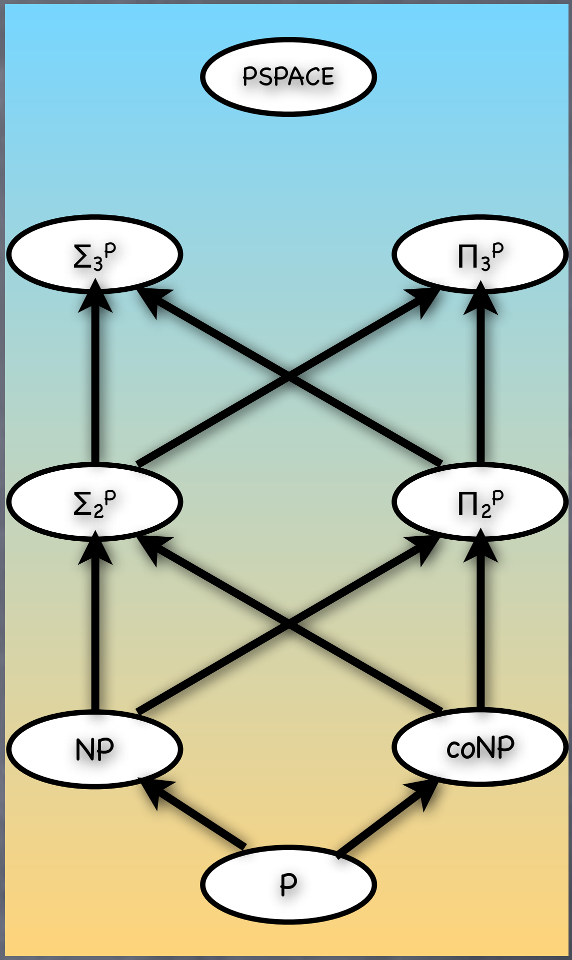
- **If $\Sigma_{k+1}^P = \Sigma_k^P$** (equivalently $\Pi_{k+1}^P = \Pi_k^P$) **then PH = Σ_k^P**
- Because then $\Pi_k^P \subseteq \Sigma_{k+1}^P = \Sigma_k^P$ and hence $\Pi_k^P = \Sigma_k^P$
  - So for k>0, implies PH = Σ_k^P
- Holds for k=0 too: i.e., NP = P $\Rightarrow$ PH = P
  - NP = P $\Rightarrow$ NP = co-NP $\Rightarrow$ PH = NP (= P)

Today



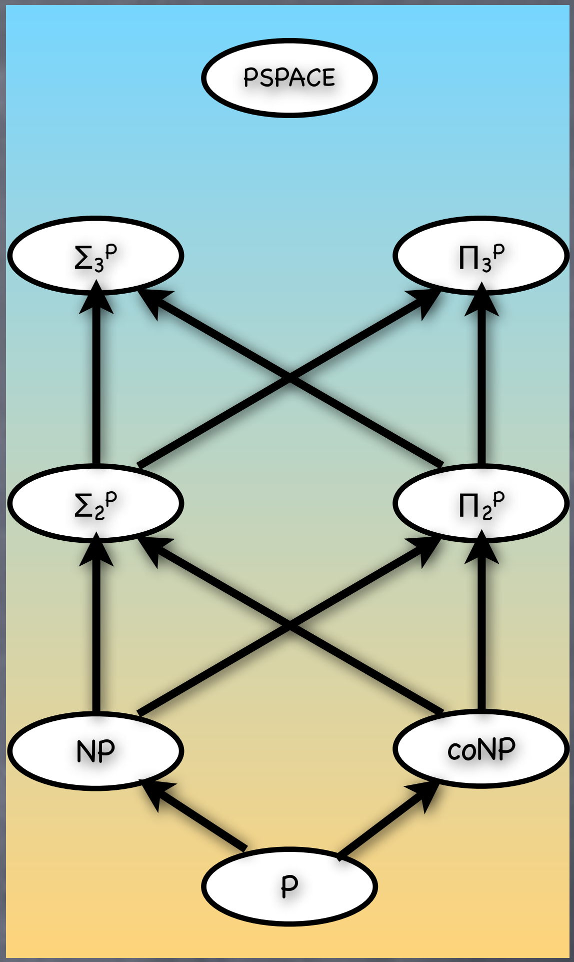
Today

- Polynomial Hierarchy



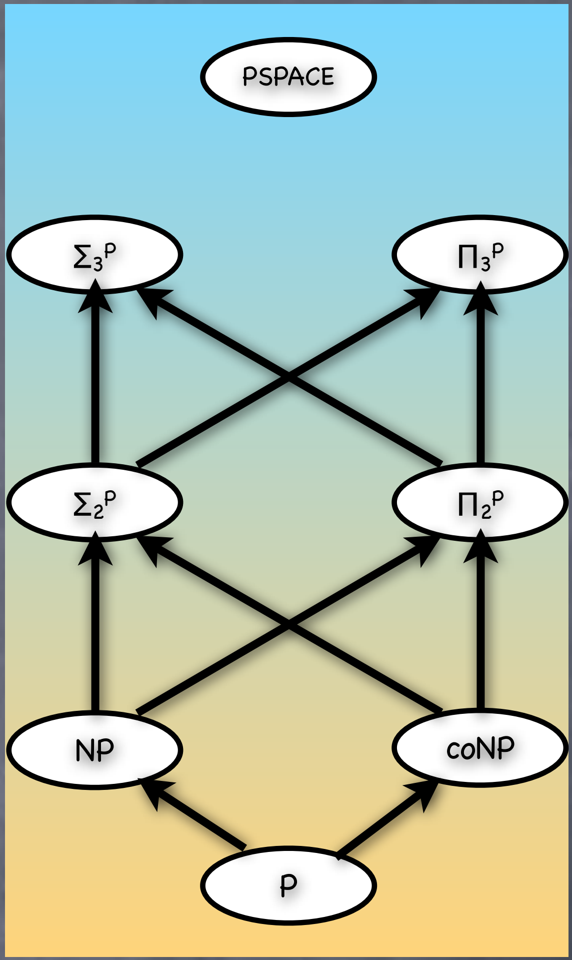
Today

- Polynomial Hierarchy
  - Σ_k^P, Π_k^P, PH



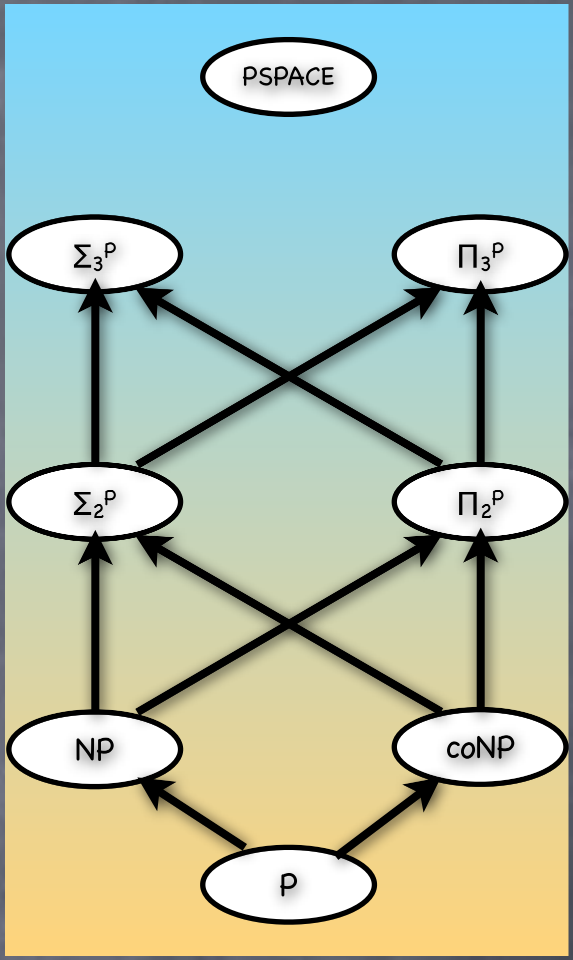
Today

- Polynomial Hierarchy
  - Σ_k^P, Π_k^P, PH
- Collapse of Polynomial Hierarchy



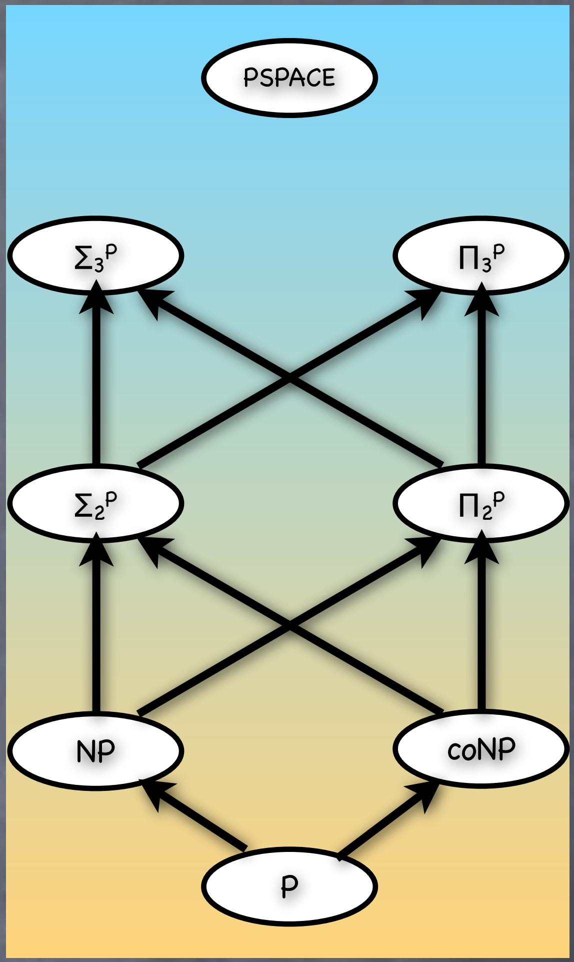
Today

- Polynomial Hierarchy
  - Σ_k^P, Π_k^P, PH
- Collapse of Polynomial Hierarchy
  - Believed not to collapse



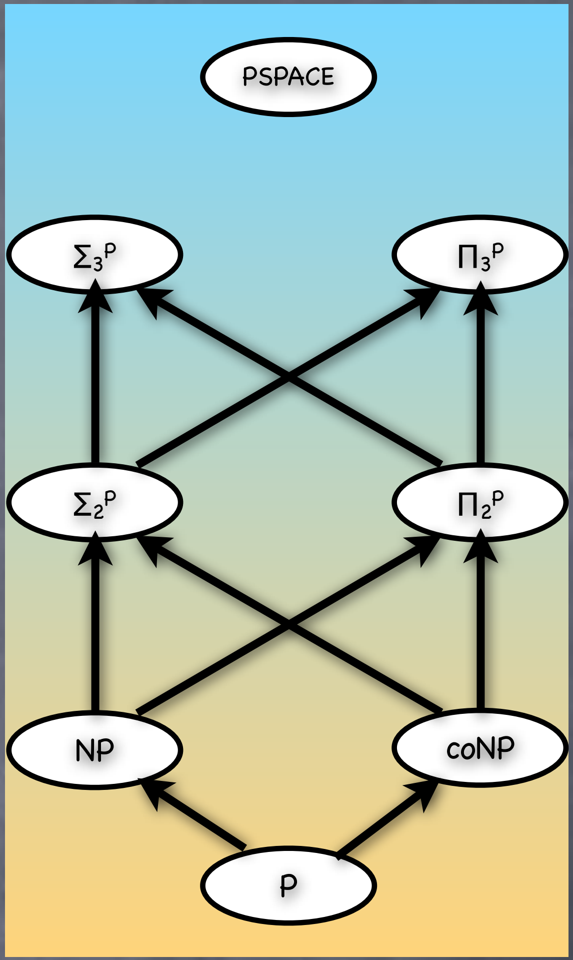
Today

- Polynomial Hierarchy
  - Σ_k^P, Π_k^P, PH
- Collapse of Polynomial Hierarchy
  - Believed not to collapse
    - at least not at the lower levels



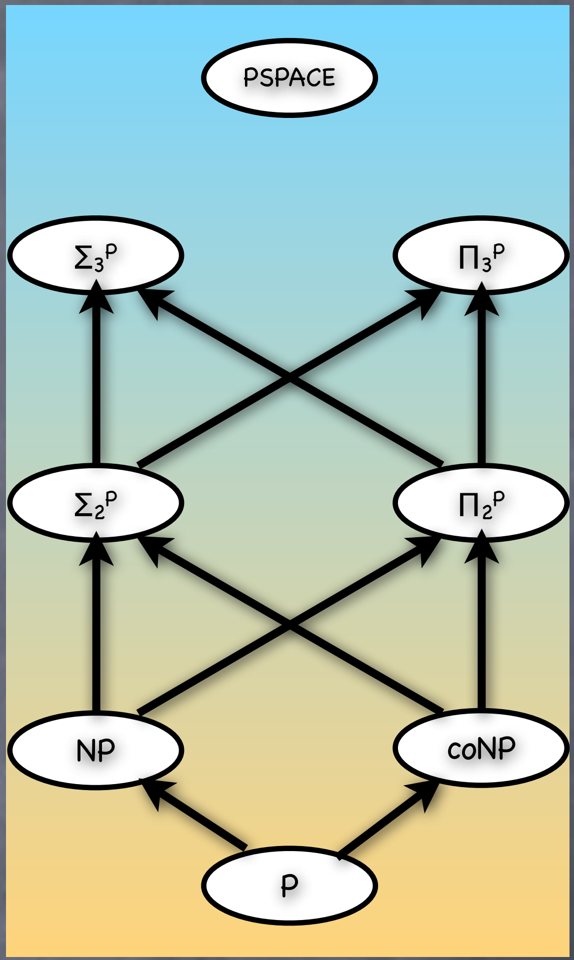
Today

- Polynomial Hierarchy
  - Σ_k^P, Π_k^P, PH
- Collapse of Polynomial Hierarchy
  - Believed not to collapse
    - at least not at the lower levels
  - If $\Sigma_k^P = \Pi_k^P$ for some k>0 then PH = $\Sigma_k^P = \Pi_k^P$

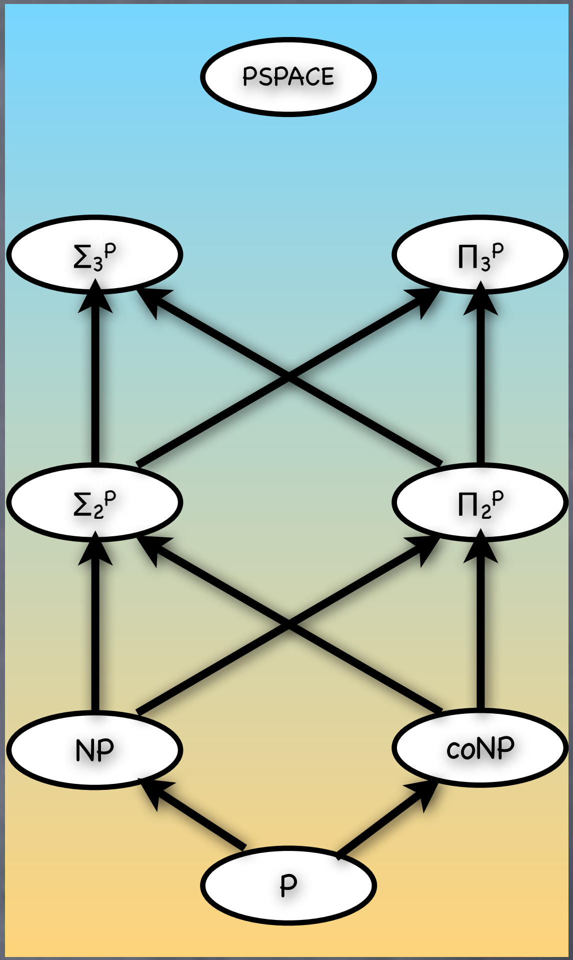


Today

- Polynomial Hierarchy
  - Σ_k^P, Π_k^P, PH
- Collapse of Polynomial Hierarchy
  - Believed not to collapse
    - at least not at the lower levels
  - If $\Sigma_k^P = \Pi_k^P$ for some k>0 then PH = $\Sigma_k^P = \Pi_k^P$
  - If $\Sigma_{k+1}^P = \Sigma_k^P$ (i.e., $\Pi_{k+1}^P = \Pi_k^P$) then PH = Σ_k^P



Today



- Polynomial Hierarchy
  - Σ_k^P, Π_k^P, PH
- Collapse of Polynomial Hierarchy
  - Believed not to collapse
    - at least not at the lower levels
  - If $\Sigma_k^P = \Pi_k^P$ for some $k>0$ then $PH = \Sigma_k^P = \Pi_k^P$
  - If $\Sigma_{k+1}^P = \Sigma_k^P$ (i.e., $\Pi_{k+1}^P = \Pi_k^P$) then $PH = \Sigma_k^P$
- Coming up: More ways to look at the polynomial hierarchy