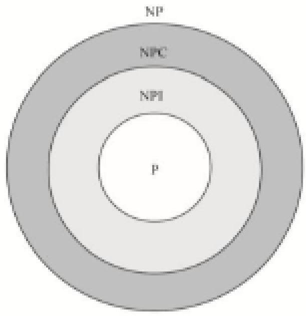
TEMA 5. Otras clases de complejidad temporal.

1- La clase NPI.

Asumiendo que $P \neq NP$, se tiene que NP incluye una subclase NPI, problemas de dificultad intermedia comparados con los de las subclases P y NPC.



Teorema de Ladner: Establece que si $B \notin P$ es un lenguaje recursivo$\rightarrow \exists$ un lenguaje $D \in P$ con:

- $A = D \cap B$ no pertenece a P
- A se reduce polinomialmente a B
- B no se reduce polinomialmente a A

Este teorema nos permite probar la existencia de la subclase NPI.

Usando la misma idea, partiendo de un lenguaje $A1 \in NPI$, se puede obtener un subconjunto $A2 \in NPI$ más fácil que A1 ($A2 \; \alpha_P \; A1$ y no $A1 \; \alpha_P \; A2$). Del mismo modo obtenemos un A3 a partir de A2 y así sucesivamente. Luego en NPI se puede definir una jerarquía infinita de problemas de dificultad intermedia a partir de cada A1.

Del mismo teorema se obtiene que en NPI existen pares de lenguajes incomparables, no se pueden reducir polinomialmente el uno al otro.

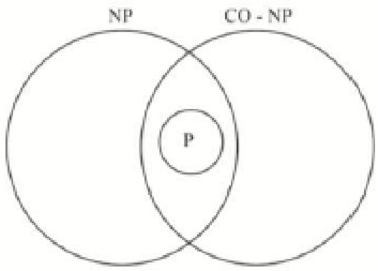
Hay problemas en NP que no disponen de resoluciones eficientes pero tampoco se ha podido probar que pertenezcan a NPC, por lo que son candidatos a estar en NPI. (p.e. el isomorfismo de grafos, la factorización).

Vimos que los lenguajes NPC no pueden ser dispersos. Sin embargo, en NPI sí pueden serlo.

2- La clase Co-NP.

CO-NP agrupa, dentro de EXP, a los lenguajes complemento de los NP.

La suposición NP ≠ CO-NP es más fuerte que la asunción P ≠ NP: si P= NP, como P es cerrada con respecto al complemento → también lo es NP, por lo que NP = CO-NP.



Definición alternativa: L∈CoNP si existe un polinomio p(n) y una MTD M en tiempo polinómico, tal que para cada x∈**Σ***, x∈L si y solo si (M(x,u) = 1 para toda u∈**Σ*** de longitud p(|x|).

Un problema de CoNP es CoNP-completo si todos los problemas de CoNP se reducen polinomialmente a él.

Si un problema de NPC →su complemento es CoNP-Completo.

BVAL={problema de la validez de las fórmulas booleanas, consiste en determinar si una fórmula booleana es satisfacible por toda asignación de valores de verdad} es CoNP-Completo.

Asumiendo NP ≠ CO-NP, se tiene que NPC y NP ∩ CoNP son disjuntos.

Se tiene más información en los problemas de NP ∩ CO-NP que en los problemas de NP – (NP ∩ CO-NP). Y al ser
NPC la subclase de los problemas más difíciles de NP, entonces los problemas de NP∩CO-NP deberían ser de dificultad baja (los de P) o intermedia (los de NPI).

Si L es NPC y además Co-NP → NP=Co-NP

3- Tiempo exponencial.

Consideramos las clases de problemas (p(n) es un polinomio):

PEXP = DTIME($2^{p(n)}$)
NPEXP = NTIME($2^{p(n)}$)

2-PEXT= DTIME($2^{2\wedge p(n)}$)
2-NEXT= NTIME($2^{2\wedge p(n)}$)
3-PEXT= DTIME($2^{2\wedge 2\wedge p(n)}$)
3-NEXT= NTIME($2^{2\wedge 2\wedge p(n)}$)
….
Ejemplos:

- El problema de las expresiones regulares con exponenciación: consiste en determinar si una expresión regular con exponenciación denota todas las cadenas de un alfabeto.
- El problema de decisión en la teoría de los números reales con adición (sin multiplicación): consiste en determinar la verdad o falsedad de una fórmula de dicha teoría.
- Problemas sobre grafos representados sucintamente. Para la representación sucinta se emplean técnicas de codificación. La representación sucinta hace que distintos problemas sobre grafos varíen ahora dentro del rango de tiempo exponencial, determinístico y no determinístico.