Computación Bio–inspirada

Tema 8: Complejidad Computacional en Modelos Celulares

David Orellana Martín
Mario de J. Pérez Jiménez

Grupo de Investigación en Computación Natural
Dpto. Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla

dorellana@us.es (http://www.cs.us.es/∼dorellana/)
marper@us.es (http://www.cs.us.es/∼marper/)

Máster Universitario en Lógica, Computación e Inteligencia Artificial
Curso 2025-2026

Índice

- Problemas de decisión y lenguajes.
- Sistemas de membranas reconocedores.
- Resolubilidad eficiente de problemas mediante sistemas de membranas reconocedores.
- Limitaciones de los de sistemas de membranas reconocedores que trabajan a modo de células.
- Limitaciones de los de sistemas de membranas reconocedores que trabajan a modo de tejidos.

Problemas de decisión y lenguajes.

Problema de decision: terna ordenada $(\Sigma_X, I_X, \theta_X)$

- Σ_X es un alfabeto finito.
- I_X es un lenguaje sobre Σ_X (cuyos elementos se denominan *instancias* del problema).
- θ_X es una función total booleana sobre I_X.

Todo problema de *optimización* (un valor ha de ser minimizado o maximizado) se puede transformar "eficientemente" en un problema decisión.

Existe una correspondencia natural entre problemas de decisión y lenguajes.

* Dado un problema de decisión $X = (\Sigma_X, I_X, \theta_X)$, se le asocia un lenguaje L_X de la siguiente manera: $L_X = \{w \in I_X : \ \theta_X(w) = 1\}$.

* Recíprocamente, dado un lenguaje L sobre un alfabeto Γ, se le asocia un problema de decisión como sigue: $X_L = (\Sigma_{X_L}, I_{X_L}, \theta_{X_L})$, en donde $\Sigma_{X_L} = \Gamma, \quad I_{X_L} = \Gamma^*, \quad \theta_{X_L} = \{(w, 1) : \ w \in L\} \cup \{(w, 0) : \ w \notin L\}$.

Resolubilidad de problemas de decisión

Reconocer un lenguaje (por **MTD**s).

* Una máquina de Turing determinista M **reconoce** un lenguaje $L \subseteq \Sigma^*$ **sii** para cada $u \in \Sigma^*$:

  (*a*) Si $u \in L$ entonces $M(u)\downarrow$ y $M(u) =$ **yes**.

  (*b*) Si $u \notin L$ entonces $M(u)\downarrow$ y $M(u) =$ **no**.

Resolver un problema de decisión (por **MTD**s).

* Una máquina de de Turing determinista M **resuelve** un problema de decisión $X = (\Sigma_X, I_X, \theta_X)$ **sii** M reconoce el lenguaje L_X asociado al problema X.

  Es decir, M **resuelve** $X = (\Sigma_X, I_X, \theta_X)$ **sii** para cada instancia $u \in I_X$:

  (*a*) Si $\theta_X(u) = 1$ entonces $M(u)\downarrow$ y $M(u) =$ **yes**.

  (*b*) Si $\theta_X(u) = 0$ entonces $M(u)\downarrow$ y $M(u) =$ **no**.

Resolubilidad de problemas de decisión

Decidir un lenguaje (por **MTND**s).

* Una máquina de Turing no determinista M **decide** un lenguaje $L \subseteq \Sigma^*$ **sii** para cada $u \in \Sigma^*$ se tiene que: $u \in L$ **sii existe al menos** una computación de $M(u)$ que es de parada y devuelve **yes**.

Resolver un problema de decisión (por **MTND**s).

* Una máquina de de Turing no determinista M **resuelve** un problema de decisión $X = (\Sigma_X, I_X, \theta_X)$ **sii** M decide el lenguaje L_X asociado al problema X.

  Es decir, M **resuelve** $X = (\Sigma_X, I_X, \theta_X)$ **sii** para cada instancia $u \in I_X$ se tiene que: $\theta_X(u) = 1$ **sii existe al menos** una computación de $M(u)$ que es de parada y devuelve **yes**.

Sistemas de membranas reconocedores

Se introducen los **sistemas de membranas reconocedores** a fin de:

- ⋆ Proporcionar **soluciones eficientes** de problemas de decisión en el marco de la computación celular con membranas (en particular, de problemas computacionalmente duros: **NP**-completos).

Ahora bien:

- ⋆ Los sistemas de membranas son **dispositivos no deterministas**.
- ⋆ Las **MTND** resuelven problemas **NP**-completos en **tiempo polinomial**.
- ⋆ Las **MTND no son fiables** ya que:
  - No capturan el **verdadero concepto de algoritmo**.
  - El resultado de una ejecución de una **MTND** puede conducir a error.

Sistemas de membranas reconocedores

Establecer "distancias" entre las **MTND**s y los sistemas de membranas.

- ⋆ Las soluciones **eficientes** que proporcionen los sistemas de membranas han de ofrecer ventajas **cualitativas** respecto a las obtenidas por **MTND**s.

- ⋆ Esas soluciones deben capturar el **verdadero concepto de algoritmo**; es decir, para cada dato de entrada del problema:

  - El resultado de **cualquier** computación asociada a esa entrada, debe codificar la respuesta correcta.

Sistemas de membranas reconocedores

Un **sistema de membranas reconocedor**[1] es un sistema de membranas tal que:

- ⋆ El alfabeto de trabajo contiene dos elementos distinguidos: **yes**, **no**.
- ⋆ Existe una membrana/celula de entrada (i_{in}).
- ⋆ La zona de salida del sistema es el entorno (i_{out}: etiqueta del entorno).
- ⋆ **Todas las computaciones** son de **parada**.
- ⋆ Si $\mathcal{C}$ es una computación del sistema, entonces o bien un objeto `yes` o un objeto `no` (pero no ambos) se envían al entorno y, además, **sólo en el último paso de la computación**.

En un sistema de membranas reconocedor, diremos que

- ⋆ Una **computación** es de **aceptación** (resp. de **rechazo**) si el objeto **yes** (resp., **no**) aparece en el entorno asociado a la configuración de parada.

[1] M.J. Pérez-Jiménez, A. Romero, F. Sancho. Complexity classes in models of cellular computing with membranes. *Natural Computing*, 2, 3 (2003), 265-285.

Codificación polinomial

Sea $X = (\Sigma_X, I_X, \theta_X)$ un problema de decisión y $\mathbf{\Pi} = \{\Pi(n) \mid n \in \mathbb{N}\}$ una familia de sistemas de membranas reconocedores.

Codificación polinomial de X en $\mathbf{\Pi}$:

- $\star$ Es un par (cod, s) de funciones sobre I_X computables en tiempo polinomial tales que para cada $u \in I_X$:
  - $s(u)$ es un número natural.
  - $cod(u)$ es un multiconjunto de entrada del sistema $\Pi(s(u))$.
  - $s^{-1}(n)$ es un conjunto finito, para cada $n \in \mathbb{N}$.

Adecuación y completitud

Sea (cod, s) una **codificación polinomial** de X en $\mathbf{\Pi}$.

- ⋆ $\mathbf{\Pi}$ es **adecuada** respecto de (X, cod, s) si para cada $u \in I_X$ se tiene:
  - Si existe una computación de aceptación de $\Pi(s(u)) + cod(u)$, entonces $\theta_X(u) = 1$.

- ⋆ $\mathbf{\Pi}$ es **completa** respecto de (X, cod, s) si para cada $u \in I_X$ se tiene:
  - Si $\theta_X(u) = 1$ entonces toda computación de $\Pi(s(u)) + cod(u)$ es de aceptación.

Resolución de problemas en tiempo polinomial (I)

Sea $\mathcal{R}$ una clase de sistemas de membranas reconocedores.

Sea $X = (\Sigma_X, I_X, \theta_X)$ un problema de decisión.

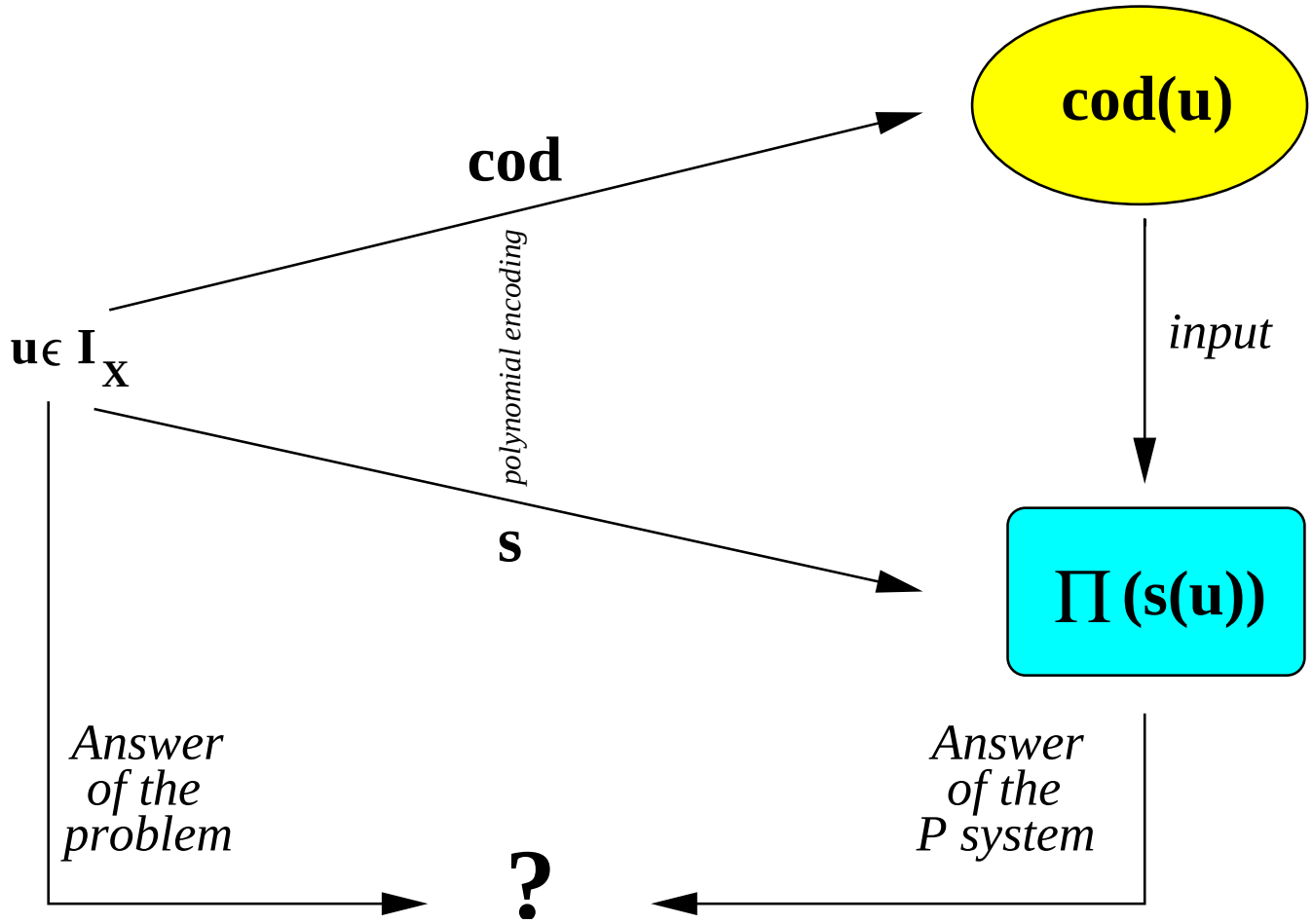
Sea $\mathbf{\Pi} = \{\Pi(n) : \ n \in \mathbb{N}\}$ una familia de sistemas de membranas de la clase $\mathcal{R}$.

Diremos que el problema X es **resoluble en tiempo polinomial** por la familia $\mathbf{\Pi}$ (y notaremos $X \in \mathbf{PMC}_{\mathcal{R}}$) **sii** :

- ⋆ $\mathbf{\Pi}$ es **polinomialmente uniforme** por MT: **Existe una MTD** que trabaja en tiempo polinomial y construye $\Pi(n)$ a partir de $n \in \mathbb{N}$.
- ⋆ Existe una **codificación polinomial** (cod, s) de X en $\mathbf{\Pi}$ tal que:
  - • $\mathbf{\Pi}$ es **polinomialmente acotada**: Existe un polinomio $p(n)$ tal que para cada $u \in I_X$, todas las computaciones de $\Pi(s(u)) + cod(u)$ paran en, a lo sumo, $p(|u|)$ pasos.
  - • $\mathbf{\Pi}$ es **adecuada** y **completa** respecto a (X, cod, s).

En este contexto, la instancia $u \in I_X$ será procesada por el sistema $\Pi(s(u))$ con multiconjunto de entrada $cod(u)$.

Resolución de problemas en tiempo polinomial (II)



Estabilidad de las clases de complejidad polinomial

Sea $\mathcal{R}$ una clase de sistemas de membranas reconocedores.

La clase de complejidad $\mathbf{PMC}_{\mathcal{R}}$:

- ⋆ Es **estable bajo reducibilidad** en tiempo polinomial.
  - Si $X_1 \leq^p X_2$ y $X_2 \in \mathbf{PMC}_{\mathcal{R}}$, entonces $X_1 \in \mathbf{PMC}_{\mathcal{R}}$.
- ⋆ Es **cerrada bajo complementario**.
  - Si $X \in \mathbf{PMC}_{\mathcal{R}}$, entonces $\overline{X} \in \mathbf{PMC}_{\mathcal{R}}$.

De las propiedades anteriores se deduce lo siguiente:

- ⋆ Si un problema **NP**-completo pertenece a la clase $\mathbf{PMC}_{\mathcal{R}}$, entonces se verifica que $\mathbf{NP} \cup \mathbf{co\text{-}NP} \subseteq \mathbf{PMC}_{\mathcal{R}}$.

Sistemas P que trabajan a modo de células: Limitaciones

$\mathcal{T}$: clase de los sistemas P básicos de transición reconocedores.

- $\star$ $\mathbf{PMC}_{\mathcal{T}} = \mathbf{P}$ (resultado establecido en [2])

$\mathcal{AM}$: clase de los sistemas P con membranas activas reconocedores.

$\mathcal{NAM}$: clase de los sistemas P reconocedores con membranas activas que **no** usan las reglas de división de membranas.

- $\star$ $\mathbf{PMC}_{\mathcal{NAM}} = \mathbf{P}$ (resultado establecido en [3])

En el tema siguiente veremos que familias de sistemas de $\mathcal{AM}$ pueden resolver problemas **NP**-completos.

---

[2] M.A. Gutiérrez, M.J. Pérez-Jiménez, A. Riscos, F.J. Romero, A. Romero. Characterizing tractability by cell-like membrane systems. En K.G. Subramanian, K. Rangarajan, M. Mukund (eds.) *Formal models, languages and applications*, World Scientific, Series in Machine Perception and Artificial Intelligence - Vol. 66, 2006, chapter 9, pp. 137-154.

[3] A. E. Porreca. Computational complexity classes for membrane systems. *Master Degree Thesis*, Università di Milano-Bicocca, Italy, 2008.

Sistemas P que trabajan a modo de tejidos: Limitaciones

$\mathcal{TC}$: clase de los sistemas P de tejidos con reglas symport/antiport reconocedores.

$\mathcal{TDC}$: clase de los sistemas P de tejidos con división celular reconocedores.

$\mathcal{TDC}(k)$: clase de los sistemas P de tejidos con división celular reconocedores y cuyas reglas de comunicación tienen **longitud, a lo sumo,** $k \geq 1$.

- $\star$ $\mathbf{P} = \mathbf{PMC}_{\mathcal{TC}} = \mathbf{PMC}_{\mathcal{TDC}(1)}$ (resultado establecido en [4])

En el tema siguiente veremos que familias de sistemas de $\mathcal{TDC}(4)$ pueden resolver problemas **NP**-completos.

[4] R. Gutiérrez-Escudero, M.J. Pérez-Jiménez, M. Rius-Font. Characterizing tractability by tissue-like P systems. *Lecture Notes in Computer Science*, 5957 (2010), 289-300.