

Tema III: Modelos de computación molecular basados en ADN

Modelos de computación *orientados a programas*.

- ▶ Estructura de datos (*tubos*)
- ▶ Operaciones básicas (*moleculares*)
- ▶ Función *semántica*

Modelo restringido de Adleman (I)

Es un modelo de computación:

- ▶ Sustrato computacional: **ADN**.
- ▶ Basado en procedimientos de *filtrado*
- ▶ Sin *memoria de acceso aleatorio*

Definición: Un **agregado** sobre un alfabeto Σ es un multiconjunto finito de elementos de Σ .

Definición: Un **tubo** sobre un alfabeto Σ es un multiconjunto finito de agregados sobre Σ .

Modelo restringido de Adleman (II)

Instrucciones moleculares básicas del modelo restringido:

- ▶ **Extraer**(T, s): admite como entrada un tubo, T , y un *símbolo* $s \in \Sigma$, y devuelve como salida dos tubos:

$$+(T, s) = \{\sigma \in T : s \in \sigma\} \quad \text{y} \quad -(T, s) = \{\sigma \in T : s \notin \sigma\}$$

- ▶ **Mezclar**(T_1, T_2): admite como entrada dos tubos T_1 y T_2 , y devuelve como salida un tubo, $T_1 \cup T_2$ (unión de ambos, como multiconjuntos).
- ▶ **Detectar**(T): admite como entrada un tubo T , y devuelve como salida **SI**, en el caso en que T contenga algún agregado, y **NO** en caso contrario.

En el modelo restringido, sólo la operación **extraer** implementa un paralelismo masivo.

Es un modelo de computación universal (*Beaver, 1998*).

Modelo débil de Amos (I)

Es un modelo de computación:

- ▶ Sustrato computacional: **ADN**.
- ▶ Basado en procedimientos de *filtrado*
- ▶ Sin *memoria de acceso aleatorio*

Definición: *Un tubo sobre un alfabeto Σ es un multiconjunto finito de **cadena**s sobre Σ .*

La diferencia fundamental entre el concepto de tubo en el modelo de Amos y el de Adleman radica en la consideración o no del direccionamiento.

Modelo débil de Amos (II)

Instrucciones moleculares básicas del modelo débil:

- ▶ **Quitar**($T, \{\gamma_1, \dots, \gamma_k\}$): admite como entrada un tubo T y unas cadenas $\gamma_1, \dots, \gamma_k \in \Sigma^*$, y devuelve el tubo obtenido de T eliminando aquellas moléculas que contengan alguna aparición de esas cadenas.
- ▶ **Copiar**($T, \{T_1, \dots, T_k\}$): admite como entrada un tubo T y devuelve k tubos, T_1, \dots, T_k , que son copias exactas de T .
- ▶ **Union**($\{T_1, \dots, T_k\}, T$): admite como entrada unos tubos T_1, \dots, T_k , y devuelve un tubo T que es la unión de T_1, \dots, T_k como multiconjuntos.
- ▶ **Selección**(T): admite como entrada un tubo T y selecciona aleatoriamente un elemento de T en el caso en que $T \neq \emptyset$; en caso contrario, devuelve **NO**.

En el modelo débil, sólo la operación **quitar** implementa un paralelismo masivo.

Es un modelo de computación universal (*Beaver, 1998*).

Modelo sticker de Roweis (I)

Es un modelo de computación:

- ▶ Sustrato computacional: **ADN**.
- ▶ Basado en procedimientos de *filtrado*
- ▶ Tiene una *memoria de acceso aleatorio*

Modelo sticker de Roweis (II)

Representación de la información:

- ▶ **Cadena de memoria** del tipo (n, k, m) , con $n \geq k \cdot m$: cadena simple de ADN de longitud n que consta de k subcadenas (*regiones*) de longitud m .

Cadena del tipo (20,4,5)

ATCGG	TCATA	GCACT	AAAAA	
-------	-------	-------	-------	--

- ▶ **Stickers** asociados a una cadena de memoria: cadena simple de longitud m complementaria con una región.

Stickers asociado a la cadena anterior

TAGCC	AGTAT	CGTGA	TTTTT
-------	-------	-------	-------

- ▶ Región *activada*: si está completamente por el sticker asociado.
- ▶ Región *desactivada*: si no está completamente por el sticker asociado.
- ▶ **Complejo de memoria** del tipo (n, k, m) , con $n \geq k$: doble hebra parcial compuesta por una cadena de memoria de dicho tipo, complementada por algunos stickers asociados.

Modelo sticker de Roweis (III)

Los complejos de memoria se identifican con números binarios y/o aplicaciones booleanas.

Conveniencia de disponer de una *frontera* natural entre las regiones de una cadena: evitar situaciones no deseadas.

Comparación de los mecanismos de representación de la información en el paradigma Adleman/Lipton y en el modelo sticker:

- ▶ Ambos están basados en la direccionalidad y en la complementariedad de Watson–Crick.
- ▶ En el paradigma Adleman/Lipton se parte de cadenas simples y cortas que pueden formar dobles hebras, con voladizos pero sin huecos.
- ▶ En el modelo sticker se parte de cadenas largas (complejos) y cortas (stickers) formando dobles hebras con voladizos posibles huecos.
- ▶ Densidad de almacenamiento de información:
 - ▶ En el paradigma Adleman/Lipton: $\frac{1}{20}$
 - ▶ En el modelo sticker: $\frac{1}{m}$

Modelo sticker de Roweis (IV)

En el modelo sticker, las operaciones separar, activar y desactivar implementan un paralelismo masivo.

Es un modelo de computación universal (*Beaver, 1998*).

Datos de entrada del modelo sticker:

- ▶ *Librería* de orden (n, k, l, m) , con $n \geq k \cdot m$ y $1 \leq l \leq k$: conjunto de **todos** los complejos del tipo (n, k, m) con las últimas $k - l$ regiones desactivadas.

Hablaremos, simplemente, de una (k, l) -librería:

- ▶ Consta de 2^l complejos de memoria: números binarios con k dígitos tales que son nulos los últimos $k - l$ dígitos.

Tubo madre: es una (k, l) -librería.