

# Computación biomolecular

Alfonso Rodríguez-Patón

[arpaton@fi.upm.es](mailto:arpaton@fi.upm.es)

Grupo LIA - UPM

[www.lia.upm.es](http://www.lia.upm.es)



[www.lia.upm.es](http://www.lia.upm.es)

Laboratorio de Inteligencia Artificial (LIA)

# Índice

1. Computación biomolecular. ¿Qué es? ¿Para qué?

2. Computación con ADN

- ADN: Estructura y función.
- Ejemplo inicial de computación con ADN:  
Problema del Camino de Hamilton (L. Adleman).
- Autómata de Benenson
- Puertas lógicas de Winfree
- Inferencia con ADN

3. Biología sintética. ¿Qué es?

- Oscilador sintético: "Repressilator".
- Célula/bacteria artificial del JCV Institute.
- Programación evolutiva con bacterias. Ecosistemas bacterianos sintéticos (piedra-papel-tijera).

4. Nanotecnología con ADN: pinzas de ADN, Origami.

5. Problemas y oportunidades. Reflexión final.

# ¿Qué es la "Computación biomolecular"?

Procesamiento de información codificada en biomoléculas utilizando biomoléculas.

Review en: Maojo V, Martín-Sánchez F, Kulikowski C, Rodríguez-Paton A, Fritts M. Nanoinformatics and DNA-Based Computing: Catalyzing Nanomedicine. Pediatric Research. 2010 Jan 28.

- Otras disciplinas relacionadas (pero distintas):
  - Biología computacional
  - Bioinformática

## ¿Por qué computar con ADN?

Porque la biotecnología lo permite

## ¿Para qué no es útil computar con ADN?

Para competir con las computadoras electrónicas actuales

## ¿Para qué computar con ADN?

Para programar/controlar dispositivos biomoleculares

Potenciales aplicaciones en:  
Biomedicina, bioremediación, energía

# Computación con ADN

No se pretende reemplazar a Intel ni la meta es resolver problemas NP-Complejos.

No va a haber ordenadores biológicos en nuestros escritorios sustituyendo a los PCs/Macs.

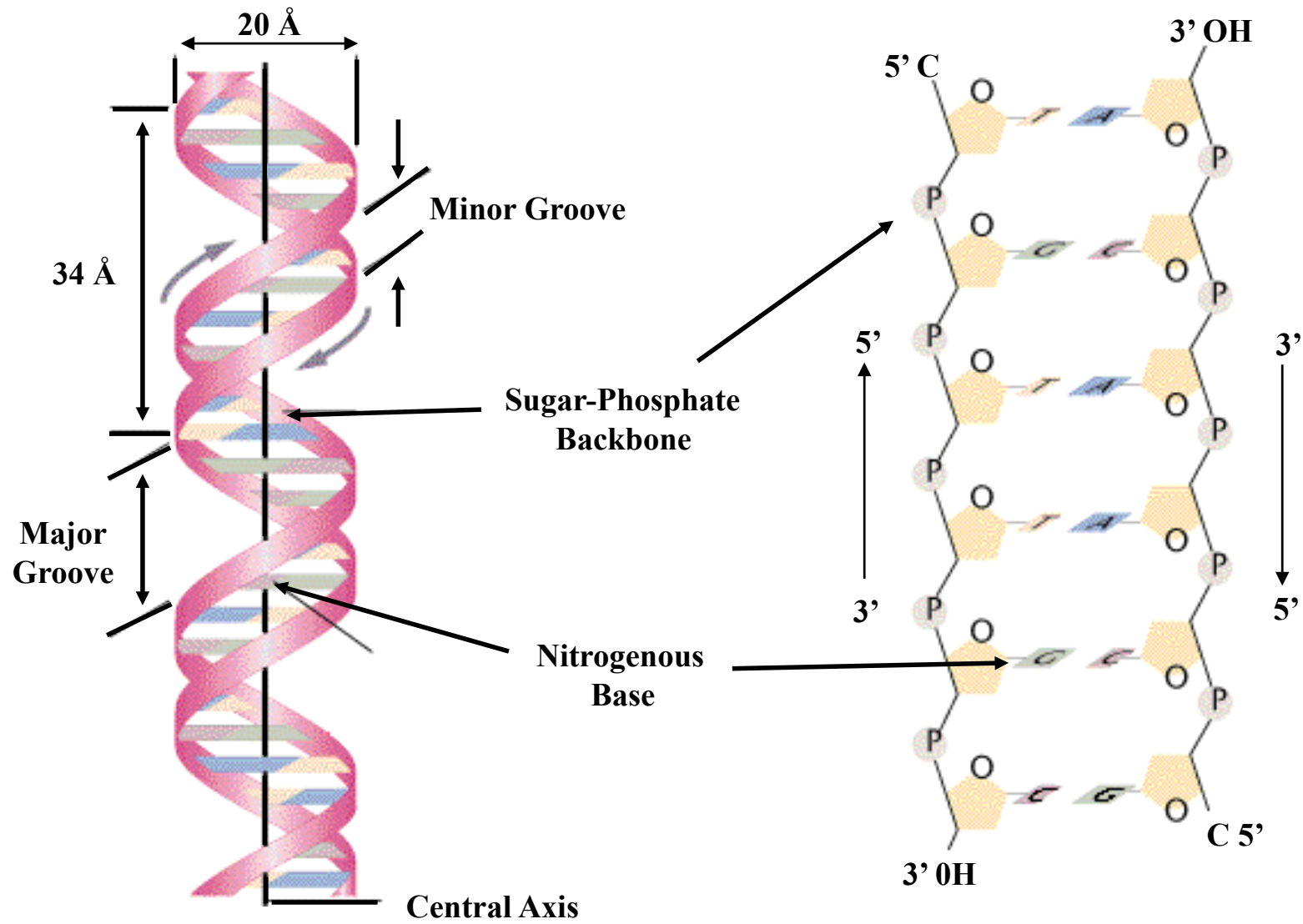
No va a haber una computadora biomolecular ejecutando Office.

Entonces, ¿para qué una computadora biomolecular?

# ¿Para qué puede ser más útil una computadora biomolecular que una electrónica?

- Para procesar información biomolecular (información codificada en señales bioquímicas o biomoléculas)
- Para procesar información biomolecular in vitro o in vivo.
- Para operar dentro de una célula o de un organismo vivo.
  - Ejemplos: Autómata biomolecular de Benenson, puertas lógicas de Winfree, “toggle switch”, Repressilator, sensores lógicos.



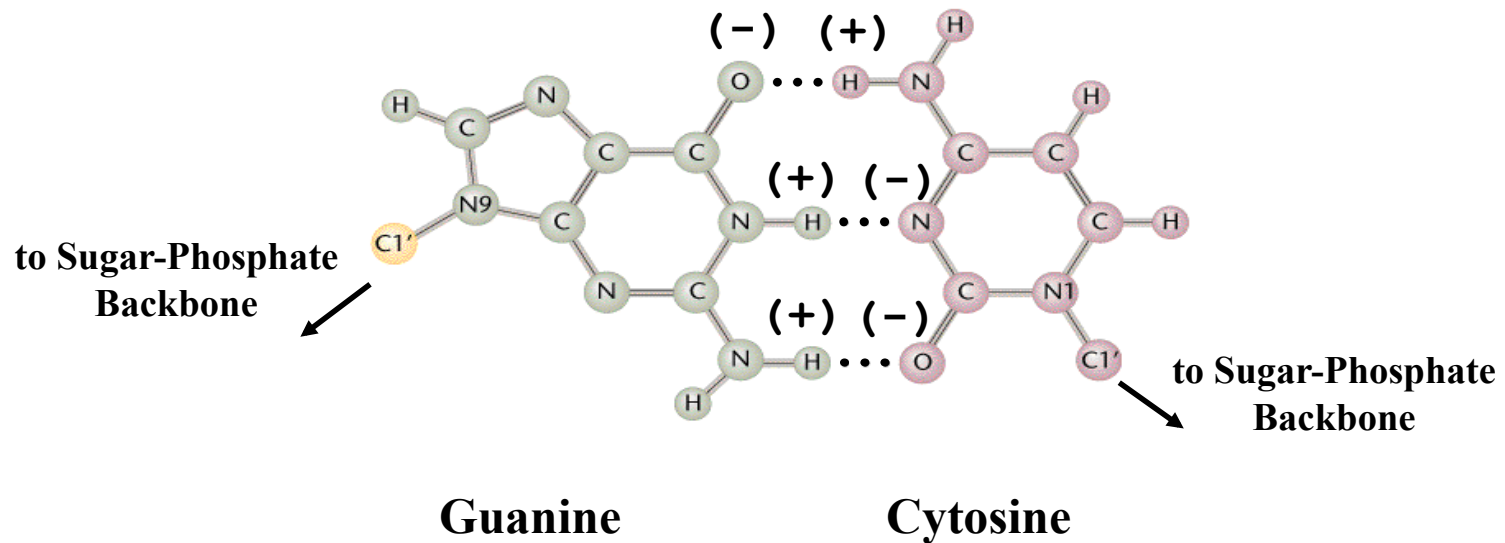
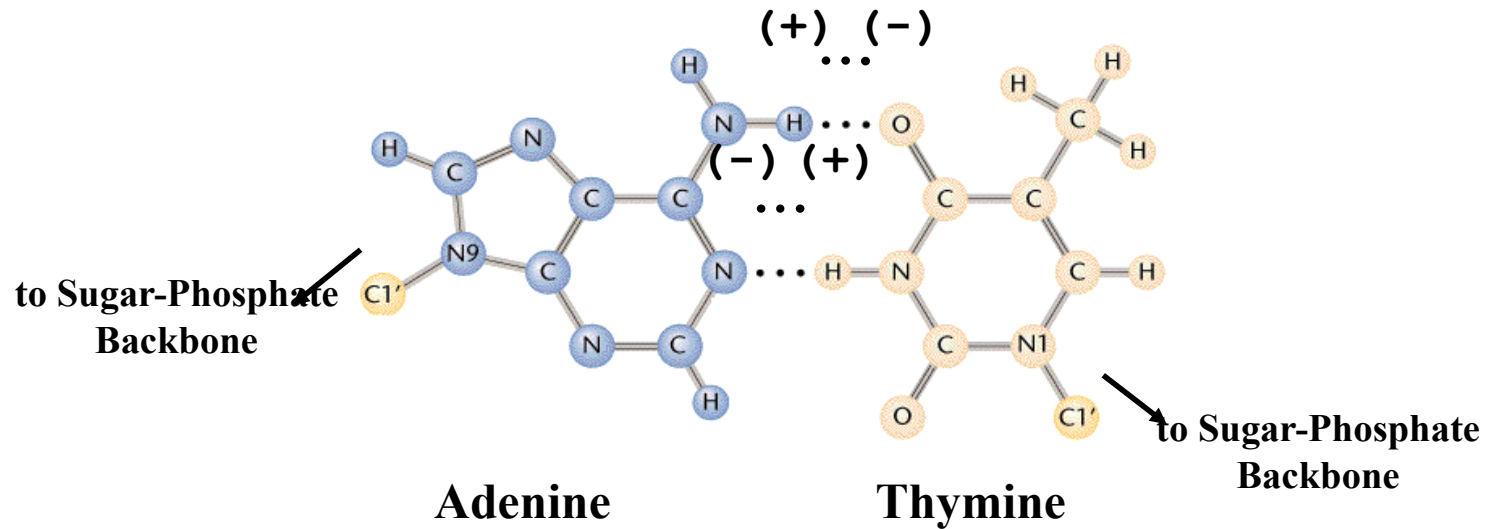




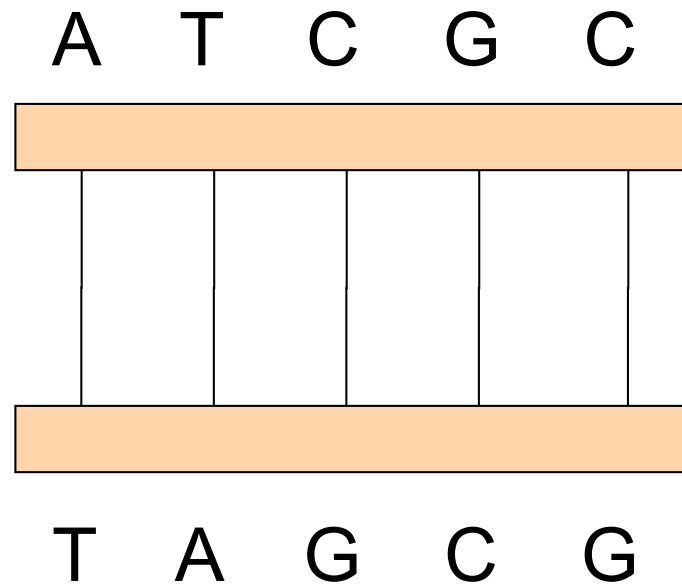
# Complementariedad de Watson-Crick:

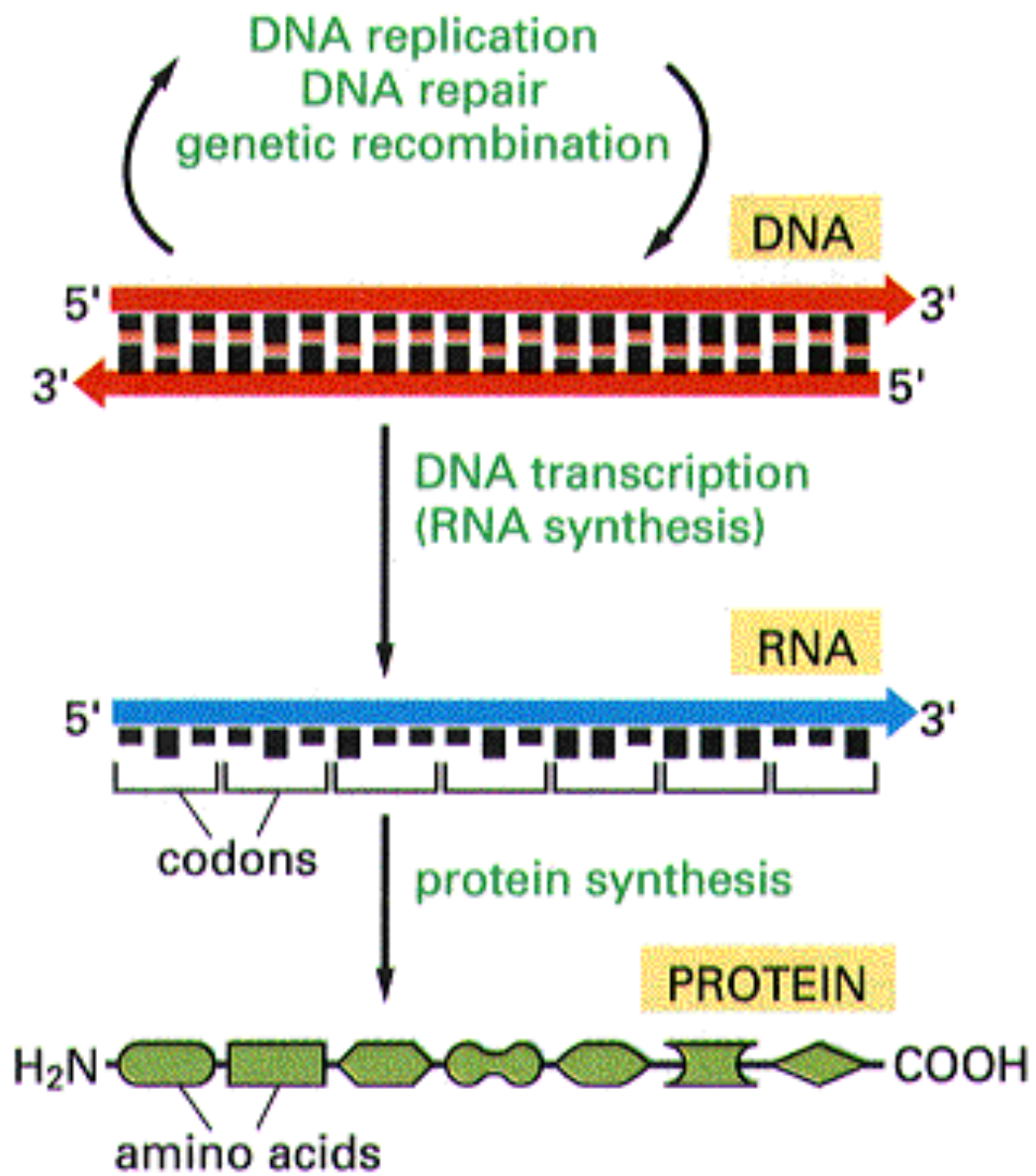
## Adenina – Timina

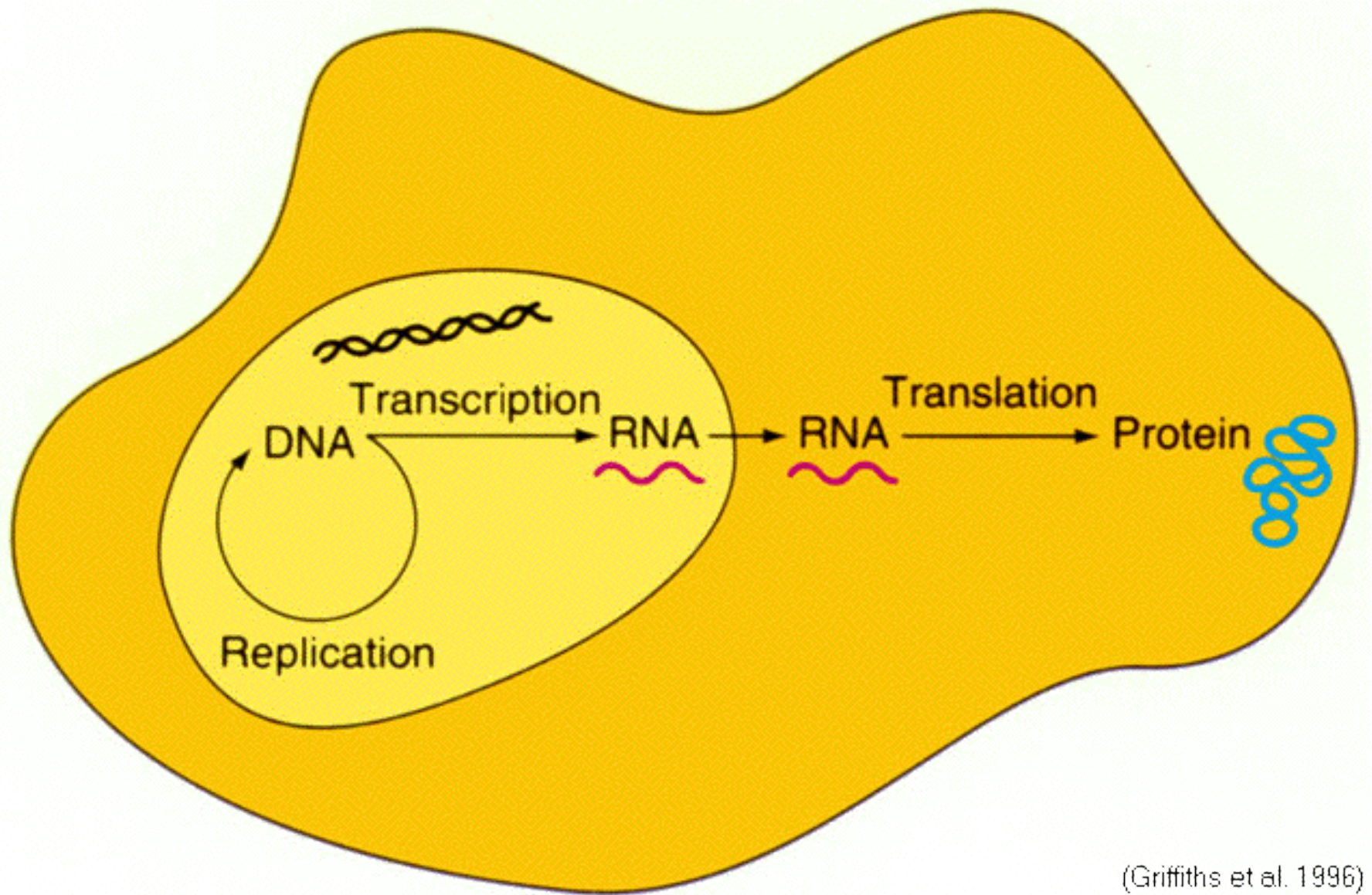
## Citosina - Guanina



# Hibridación y renaturalización







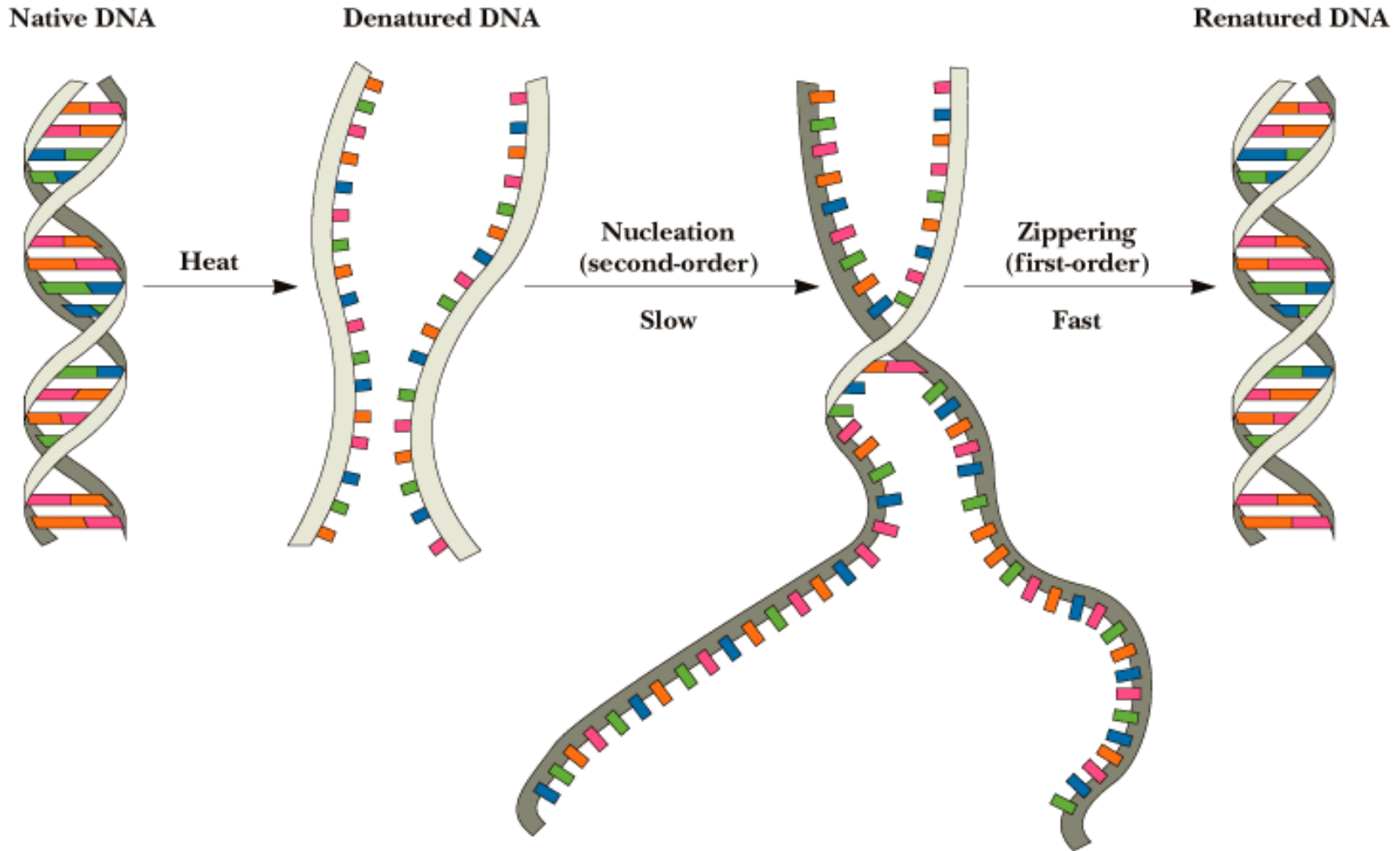
(Griffiths et al. 1996)

# Operaciones con ADN

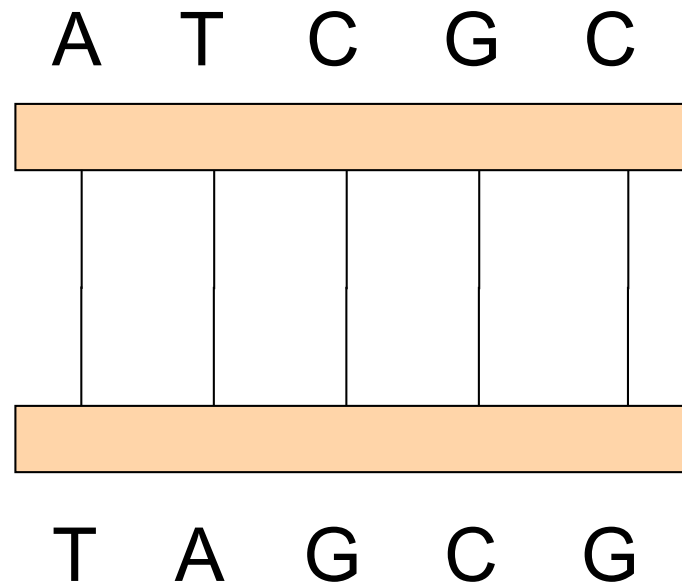
- Sintetizar, secuenciar.
- Hibridación por complementariedad de bases.
- Desnaturalización.
- Corte.
- Separación de cadenas por longitud.  
Electroforesis en gel.
- Extracción.
- Polimerización con ADN Polimerasa.
- Amplificación con PCR. Fotocopiar el ADN.

# Hibridación y renaturalización

Garrett & Grisham: Biochemistry, 2/e  
Figure 12.19



# Hibridación y renaturalización





# Separación, detección, extracción

- **Hibridación con sondas:** método de 'búsqueda' o detección de secuencias concretas de ADN.

**Sonda:** Pequeña hebra monocatenaria complementaria de la hebra buscada.

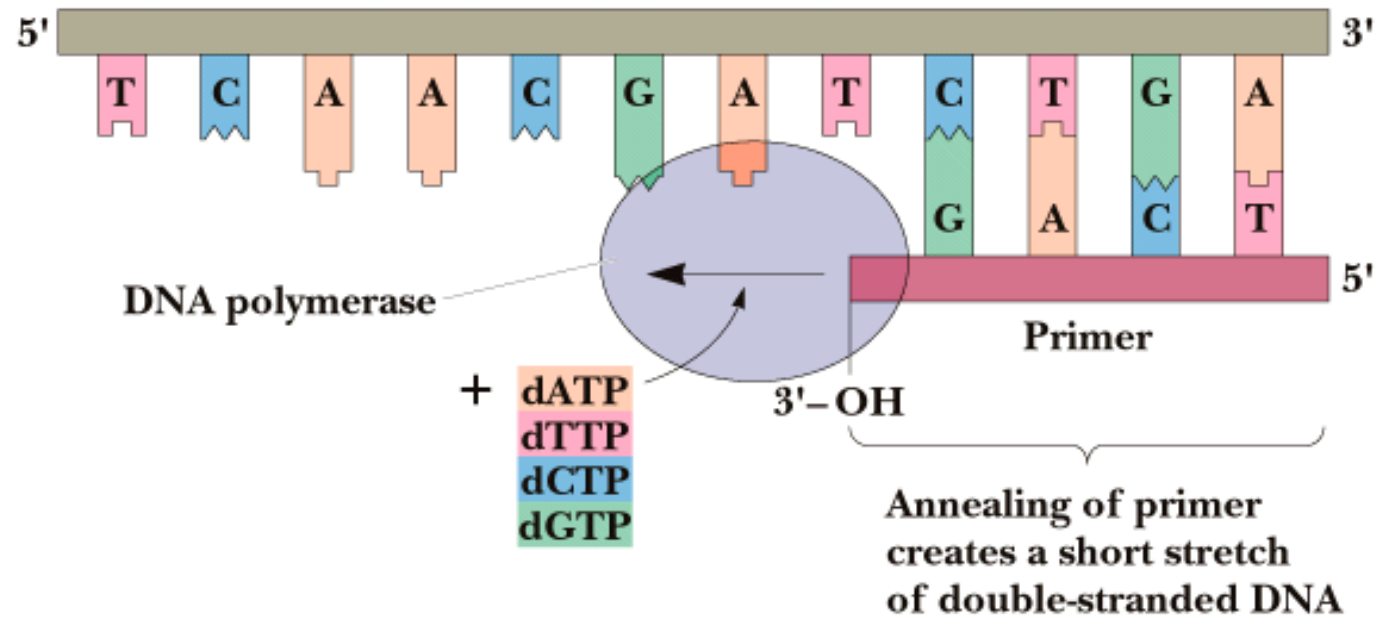
1. Desnaturalización de las hebras objetivo.
2. Añadir una sonda 'marcada' y permitir su hibridación.
3. Examinar si se ha producido la hibridación y extracción del par sonda/hebraobjetivo.



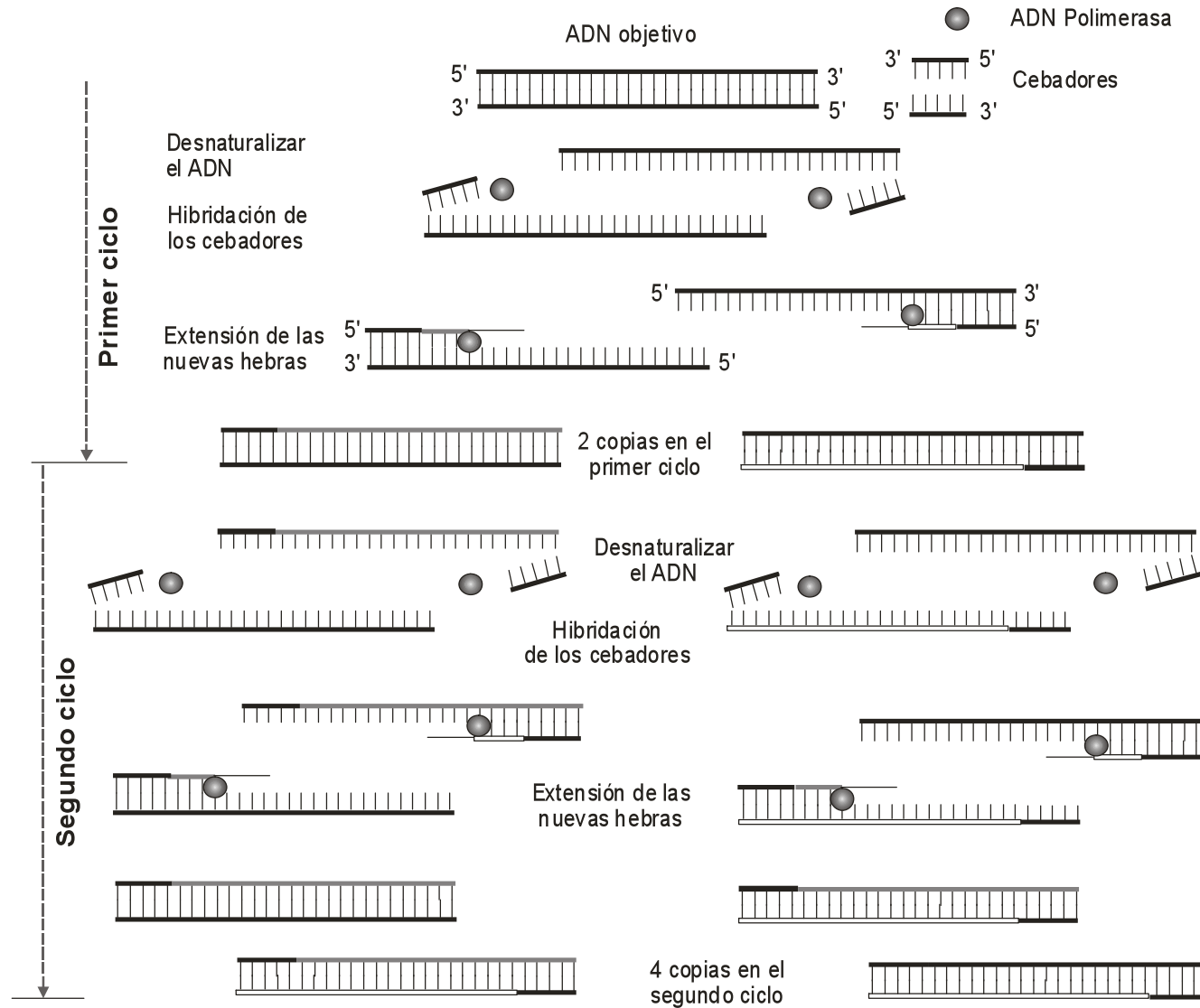
# DNA Polimerasa

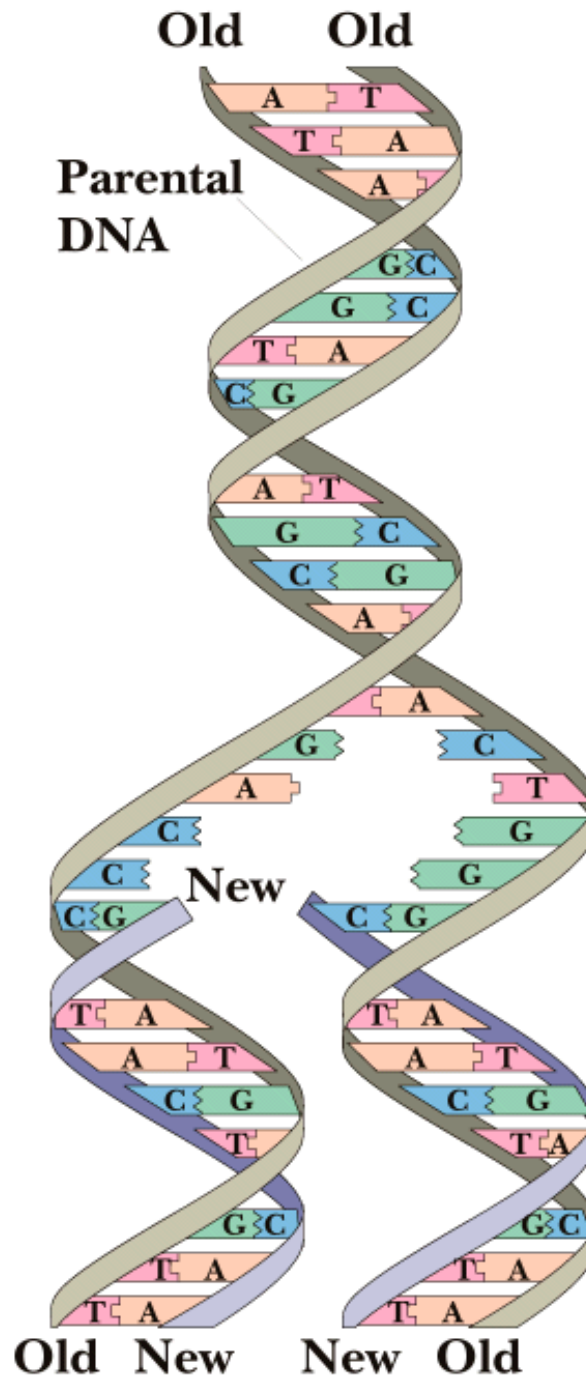
Garrett & Grisham: Biochemistry, 2/e  
Figure 12.2

Single-stranded DNA



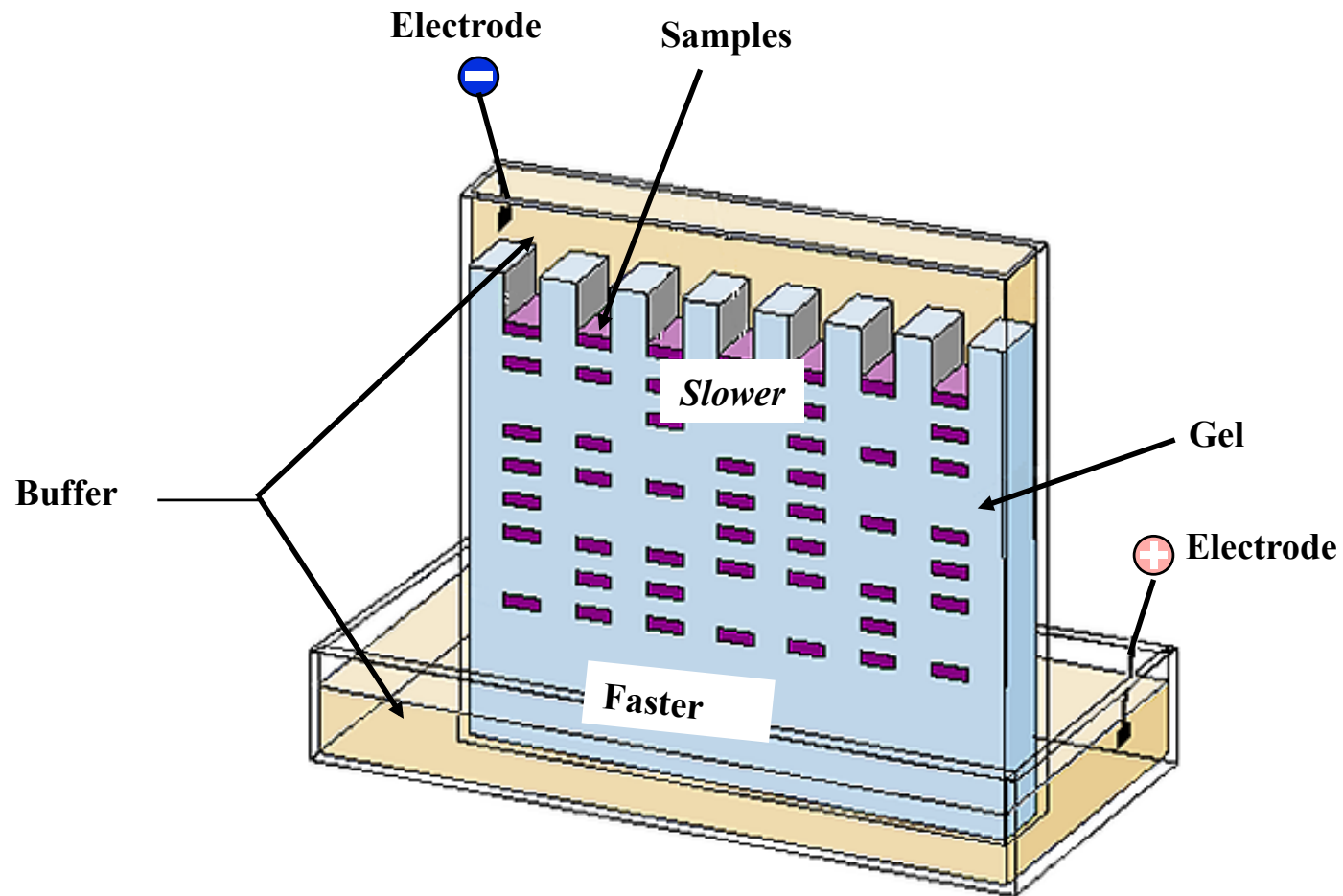
# Reacción en cadena de la Polimerasa (PCR)





# Electroforesis en gel

## Separación de hebras de ADN por longitud



# Primer trabajo: L. Adleman (94)

Molecular Computation of Solutions to Combinatorial Problems. Leonard M. Adleman. Science, Vol. 266, No. 5187. (Nov. 11, 1994), pp. 1021-1024.

## Problema del Camino de Hamilton Dirigido

**Instancia:** Grafo dirigido  $G = (V, A)$  con  $n$  vértices.  
Vértices especiales:  $V_{in}$  ;  $V_{fn}$

**Pregunta :** ¿Existe un camino Hamiltoniano, esto es, una secuencia de arcos  $e_1, e_2, \dots, e_z$  que comienza en  $V_{in}$  y termina en  $V_{fn}$  y accede a cada uno de los restantes vértices una única vez?

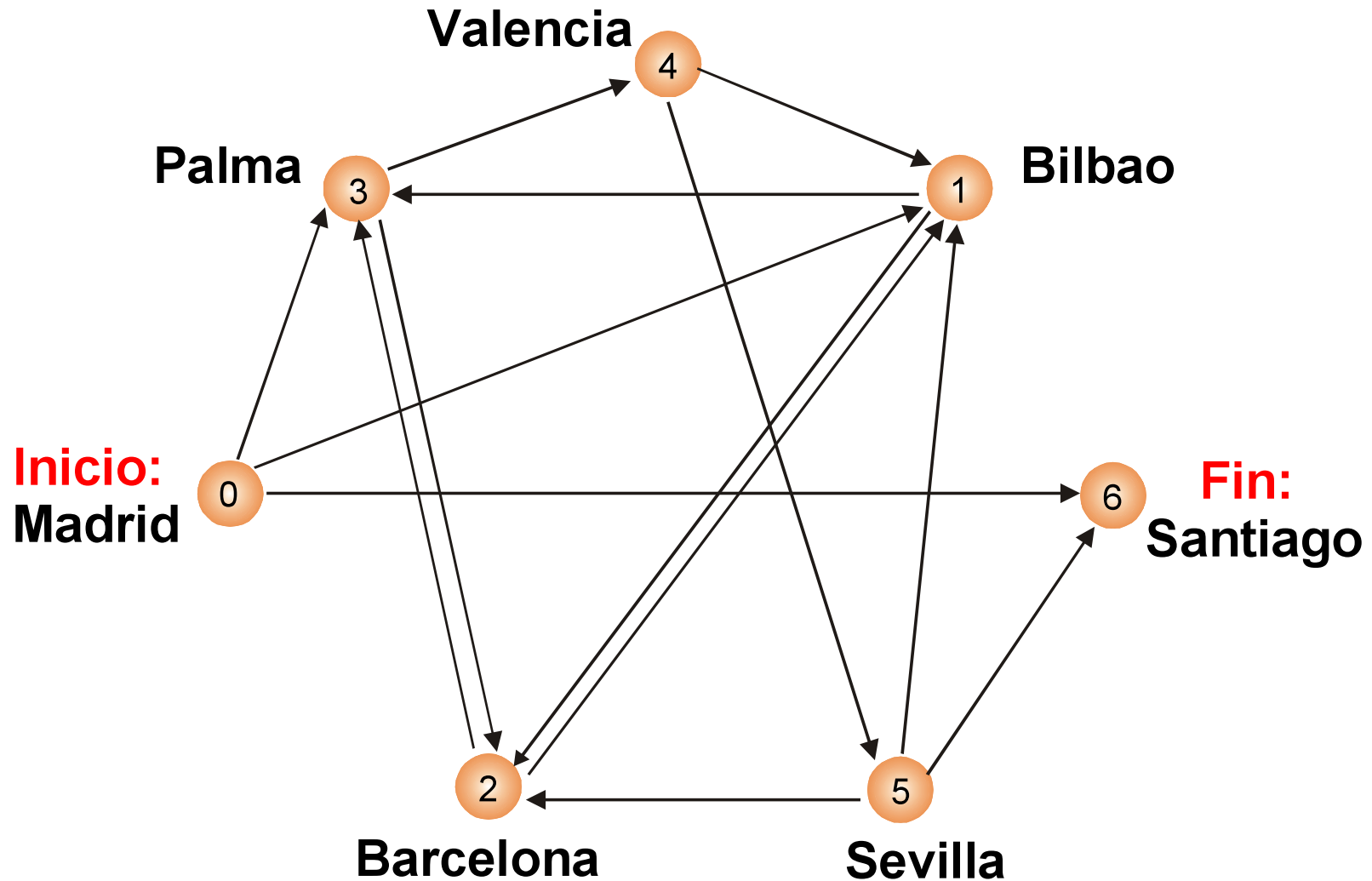
# Problema del Viajante

**Instancia:** Grafo dirigido con  $n$  ciudades conectadas por una serie de vuelos de avión.

Ciudades especiales:  $Ciudad_{inicial}$   
 $Ciudad_{final}$

**Pregunta :** ¿Existe una secuencia de vuelos que comience en  $Ciudad_{inicial}$  y termine en  $Ciudad_{final}$  y acceda a cada una de las restantes ciudades una única vez?

# Grafo de Adleman



# Algoritmo de Adleman

Molecular Computation of Solutions to Combinatorial Problems. Leonard M. Adleman. Science, Vol. 266, No. 5187. (Nov. 11, 1994), pp. 1021-1024.

**Paso 1:** Generar rutas aleatorias a lo largo del grafo.

**Paso 2:** Mantener sólo aquellas rutas que comienzan en  $Ciudad_{inicial}$  y terminan en  $Ciudad_{final}$ .

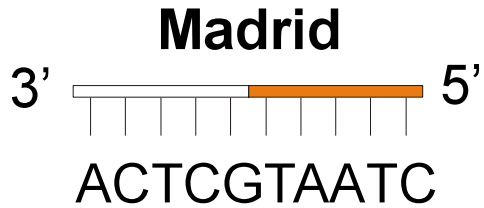
**Paso 3:** Si el grafo posee  $n$  ciudades mantener sólo aquellas rutas que acceden a exactamente  $n$  ciudades.

**Paso 4:** Mantener sólo aquellas rutas que acceden a todas las ciudades del grafo por lo menos una vez.

**Paso 5:** Si todavía queda alguna ruta, responder Sí ; en otro caso, No.

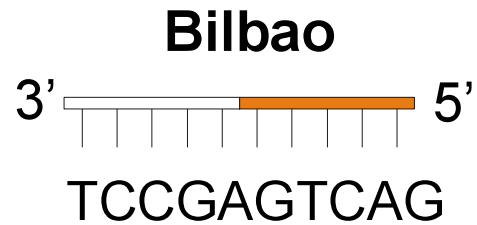


# Codificación de Adleman I



+

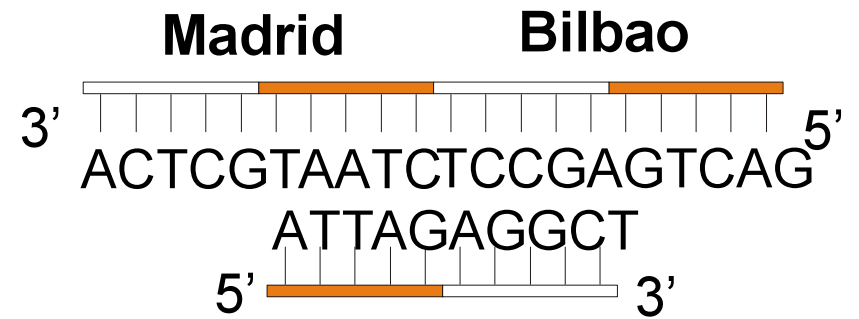
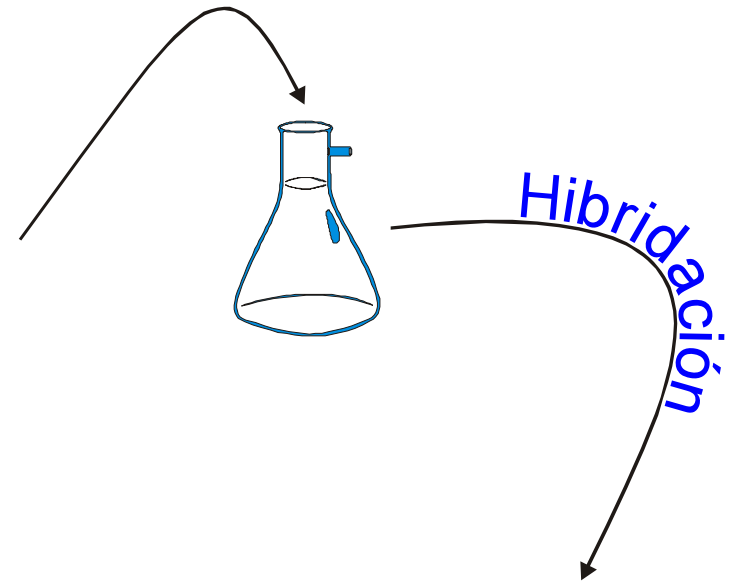
+



**Vuelo  
Madrid a Bilbao**

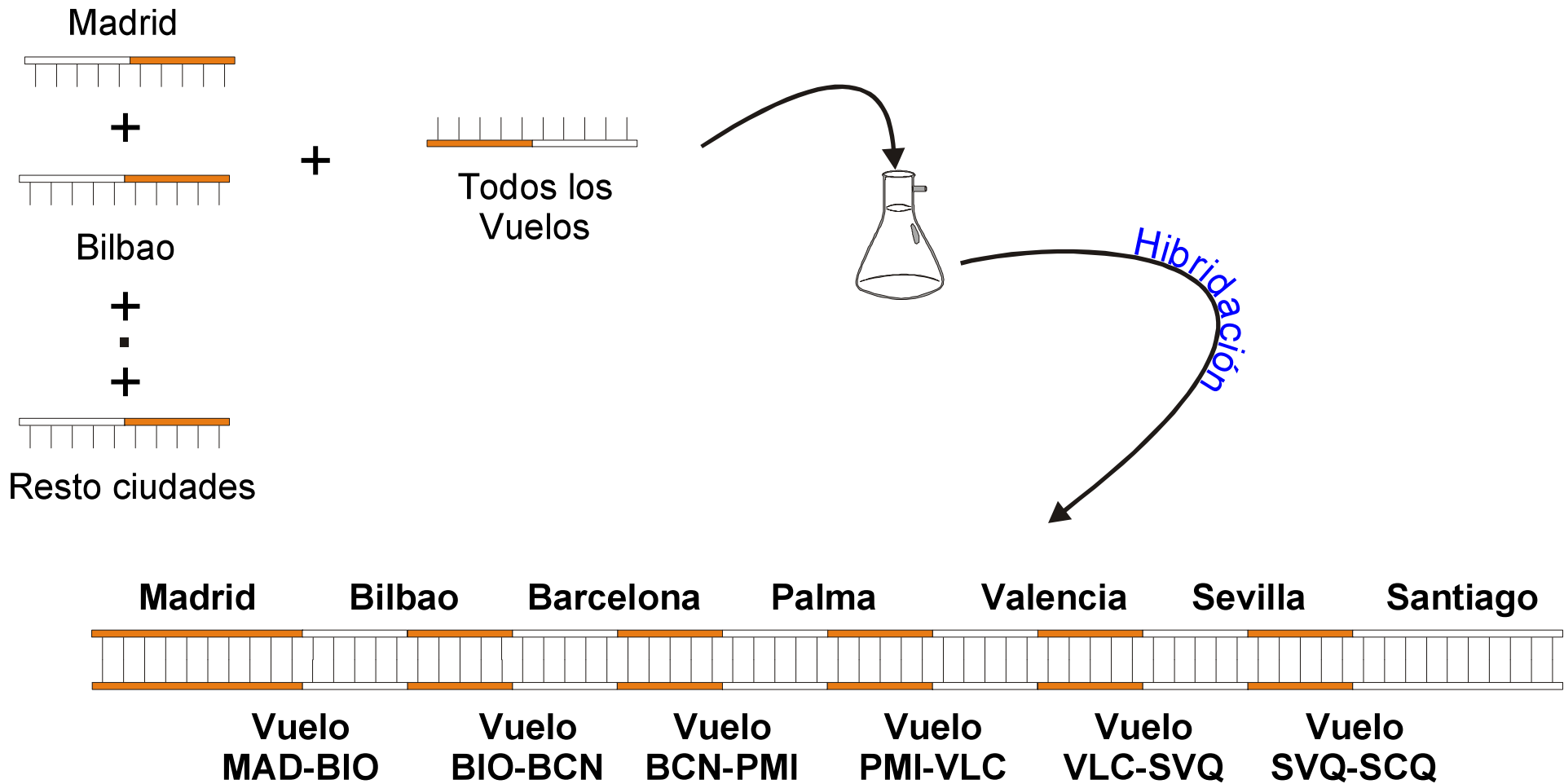


LEGO



**Vuelo  
Madrid a Bilbao**

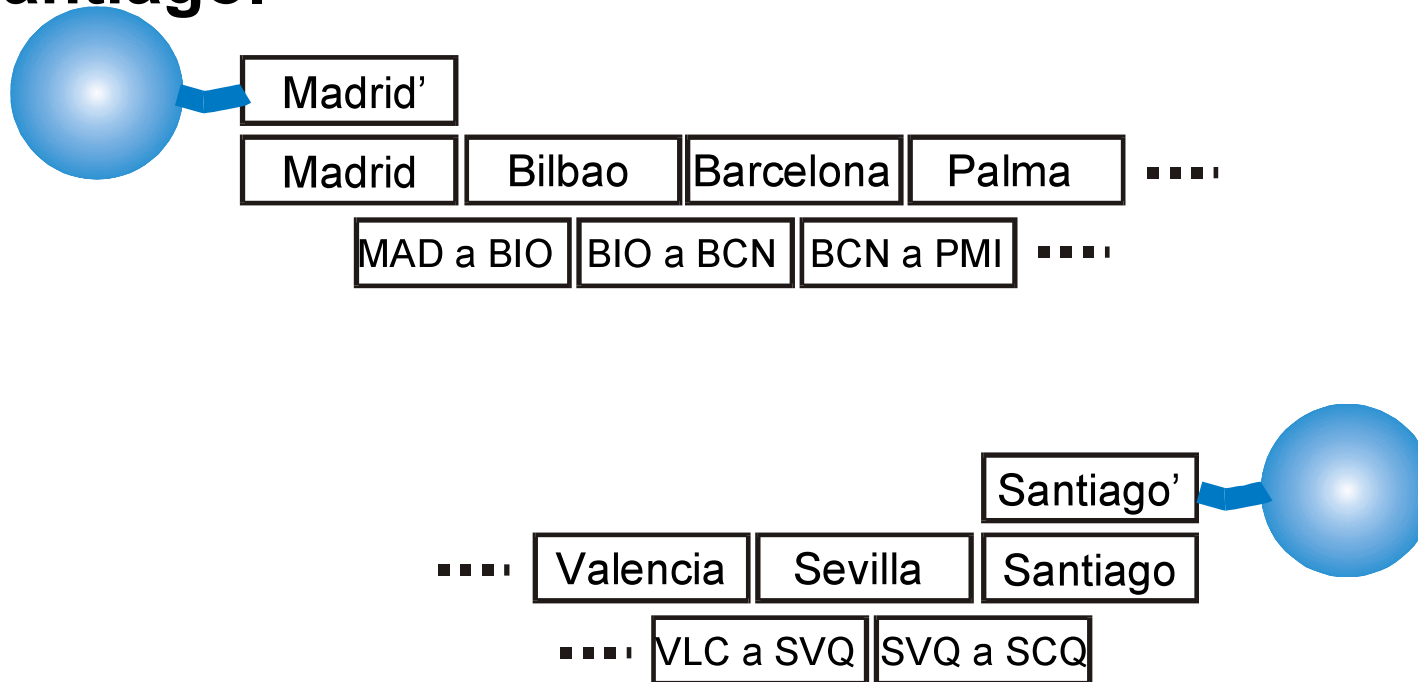
# Codificación de Adleman II



**Camino hamiltoniano:** Madrid-Bilbao-Barcelona-Palma-Valencia-Sevilla-Santiago

# Implementación (experimento)

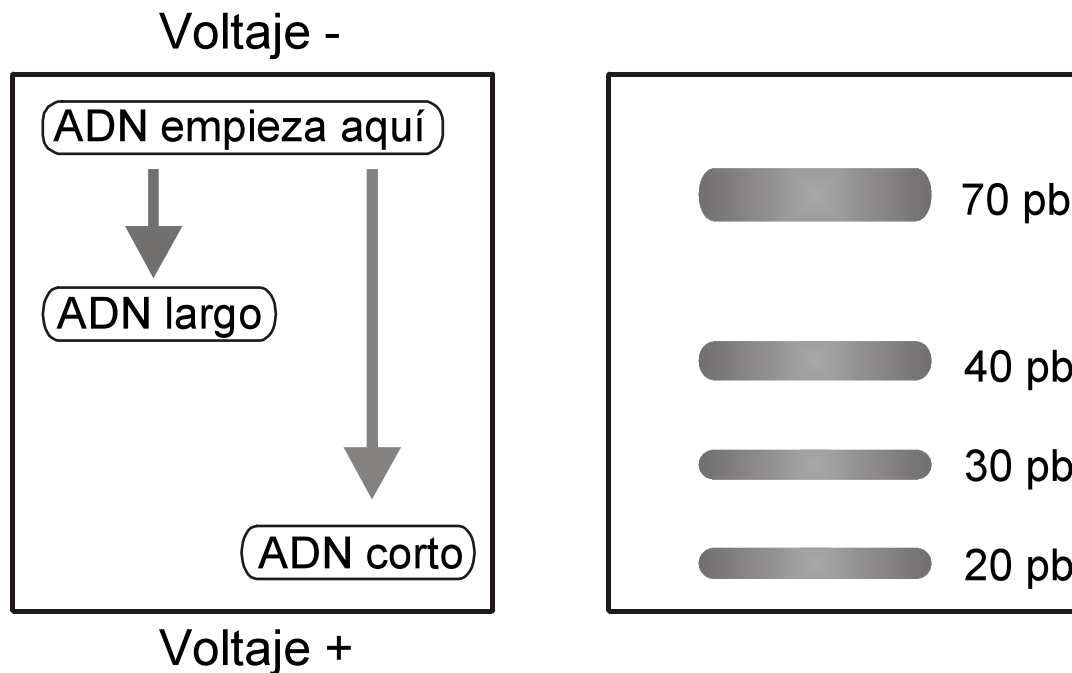
**Paso 2:** Retener (extraer) sólo aquellas hebras que comienzan por Madrid y terminan en Santiago.



# Implementación (experimento)

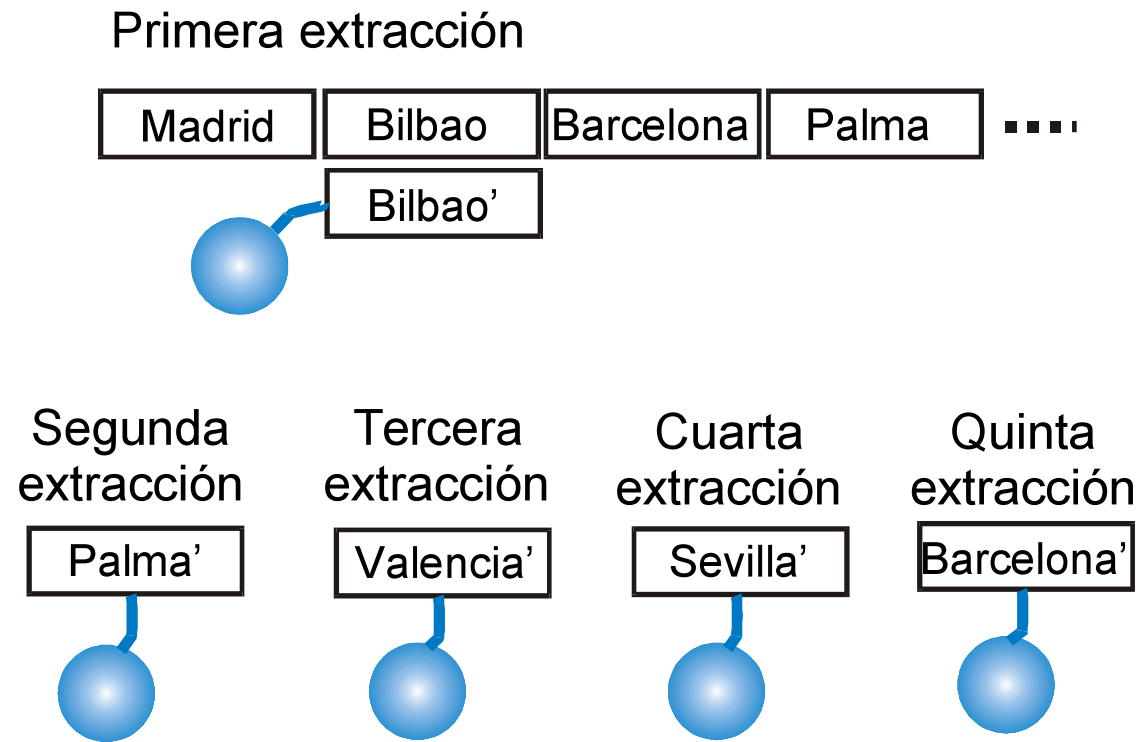
## Paso 3: Electroforesis en gel.

Retener banda de  $7 \times 10 = 70$  bases.



# Implementación (experimento)

**Paso 4:** Seleccionar rutas que contengan todas las ciudades. Extracciones sucesivas con sondas.



# Implementación (experimento)

## **Paso 5: Amplificación. Detección (secuenciación).**

Si queda alguna hebra en el tubo, dicha hebra estará codificando un camino hamiltoniano. En nuestro caso, deberían\* haber sobrevivido las hebras de ADN que codifican la ruta:

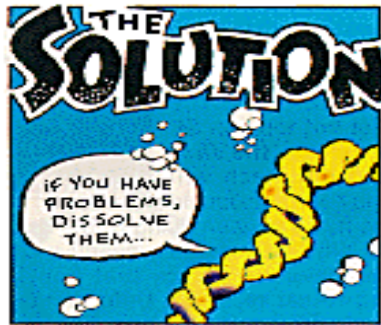
**Madrid - Bilbao – Barcelona – Palma – Valencia – Sevilla – Santiago**

\* Salvo error experimental

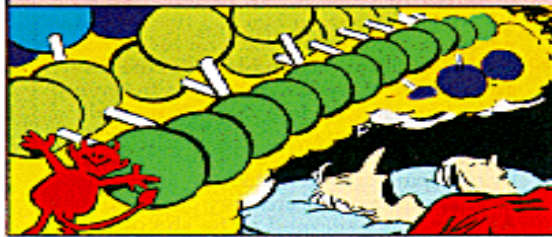


# SCIENCE CLASSICS

BY LARRY CONICK



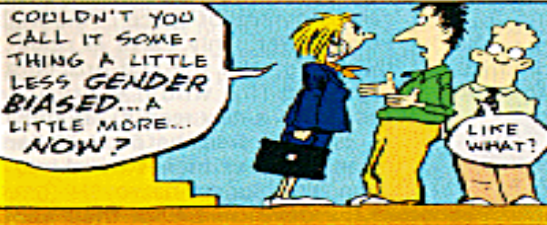
AS COMPUTER COMPONENTS SHRINK YEAR BY YEAR, SCIENTISTS DREAM OF THEIR ULTIMATE GOAL: A CHEMICAL COMPUTER, WHOSE WORKING PARTS WOULD BE INDIVIDUAL MOLECULES.



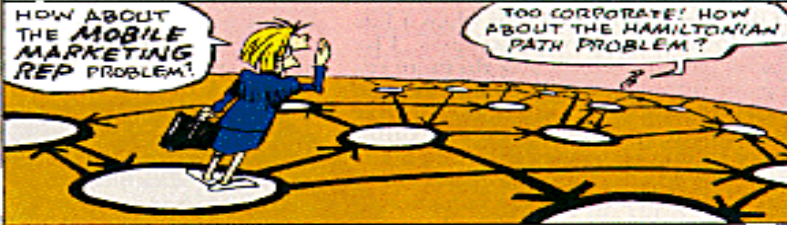
BUT THIS HAS REMAINED ONLY A DREAM—UNTIL NOW. LEONARD ADLEMAN OF THE UNIVERSITY OF SOUTHERN CALIFORNIA HAS JUST SHOWN HOW TO DO COMPUTATION USING DNA.



ADLEMAN, A COMPUTER SCIENTIST, CHOSE A TASK THAT REPRESENTS A WHOLE CLASS OF HARD-TO-SOLVE PROBLEMS. COMPUTER GUYS CALL IT THE TRAVELING SALESMAN PROBLEM.



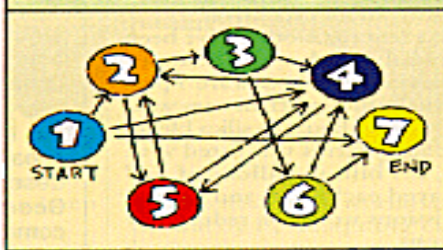
IN THIS VERSION, THE MARKETING REP HAS A MAP OF SEVERAL CITIES WITH ONE-WAY STREETS BETWEEN SOME OF THEM. THE PROBLEM IS TO FIND A ROUTE (IF IT EXISTS) THAT PASSES THROUGH EACH CITY EXACTLY ONCE, WITH A DESIGNATED BEGINNING AND END.



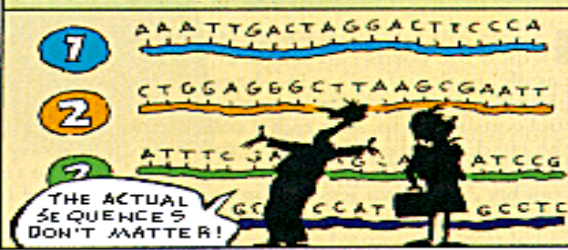
WHEN THE NUMBER OF CITIES IS LARGE—SAY MORE THAN 100—THIS PROBLEM IS TOO MUCH FOR EVEN THE FASTEST COMPUTER.



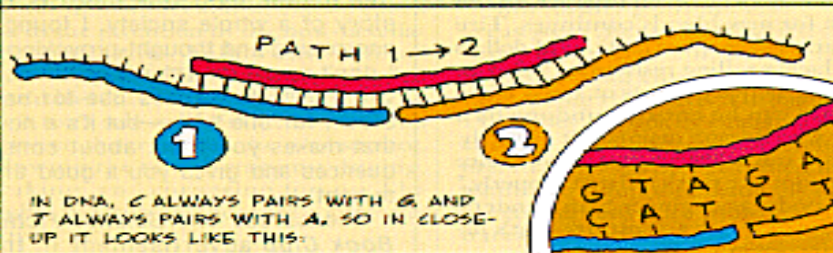
FOR HIS DNA COMPUTATION, ADLEMAN CHOSE THIS SIMPLE ARRANGEMENT OF 7 CITIES AND 13 STREETS.



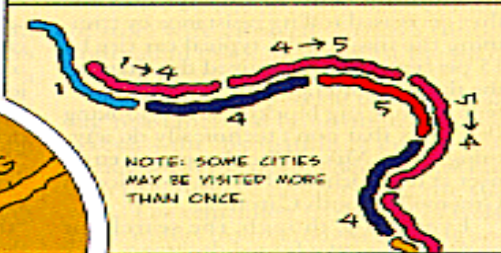
HE REPRESENTED EACH CITY CHEMICALLY BY A SINGLE STRAND OF DNA 20 BASES LONG. ITS SEQUENCE CHOSEN AT RANDOM.



A STREET BETWEEN TWO CITIES IS THE COMPLEMENTARY 20-BASE STRAND THAT OVERLAPS EACH CITY'S STRAND HALFWAY. THIS STREET LITERALLY JOINS THE TWO CITIES.



A MULTICITY TOUR BECOMES A PIECE OF DOUBLE-STRANDED DNA, WITH THE CITIES LINKED IN SOME ORDER BY THE STREETS.



Revista Discover

# Visión de L. Adleman

Leonard Adleman (2000):

“Whether or not DNA computers will ever become stand alone competitors for electronic computers—which is unlikely—is not the point. Every living cell is filled with thousands of incredibly small, amazingly precise instruments in the form of molecules, which comprise Nature's Tool Chest. I believe things like DNA computing, along with the other ways we are learning to use these wonderful tools, will eventually lead the way to a **molecular revolution**, which ultimately will have a very dramatic effect on the world.”



# Modelo de Adleman de computación con ADN

- **Entrada:** Soluciones de un problema codificadas en hebras de ADN en un tubo de ensayo.
- **Memoria:** tubo de ensayo con hebras de ADN.
- **Programa:** Secuencia de operaciones que el ingeniero realiza externamente.
- **Computación:** Secuencia de tubos de ensayo.
- **Solución:** Contenido del tubo final.

Programa no almacenado. Fuerza bruta.

“Generar y examinar”. Filtrado sucesivo.

# Computadoras biológicas autónomas y programa almacenado

- Algoritmo de Adleman: El programa no estaba almacenado.
- El ADN como memoria o sustrato sobre el que codificar información (ya sean datos o programas).
- Autómata biomolecular de Benenson: Las transiciones del autómata (software) y las entradas están codificadas en ADN. El hardware es una proteína (una enzima).
- El ADN como sustrato sobre el que especificar un programa (software) que ejecuta una célula o una bacteria.

# Medicina en 2050: “**Doctor in a Cell**”

**A genetically modified cell that can operate in the human body with an intra-cellular computer ...**

that receives **input** from signal transduction pathways and, based on its program, produces **output** to protein synthesis and secretion pathways effecting any desired molecular medical treatment.

Ehud Shapiro, 2006 (Inst. Weizman, Israel)

# Doctor en una célula

Conocimiento/diagnóstico codificado en un dispositivo biomolecular que se introduce en una célula

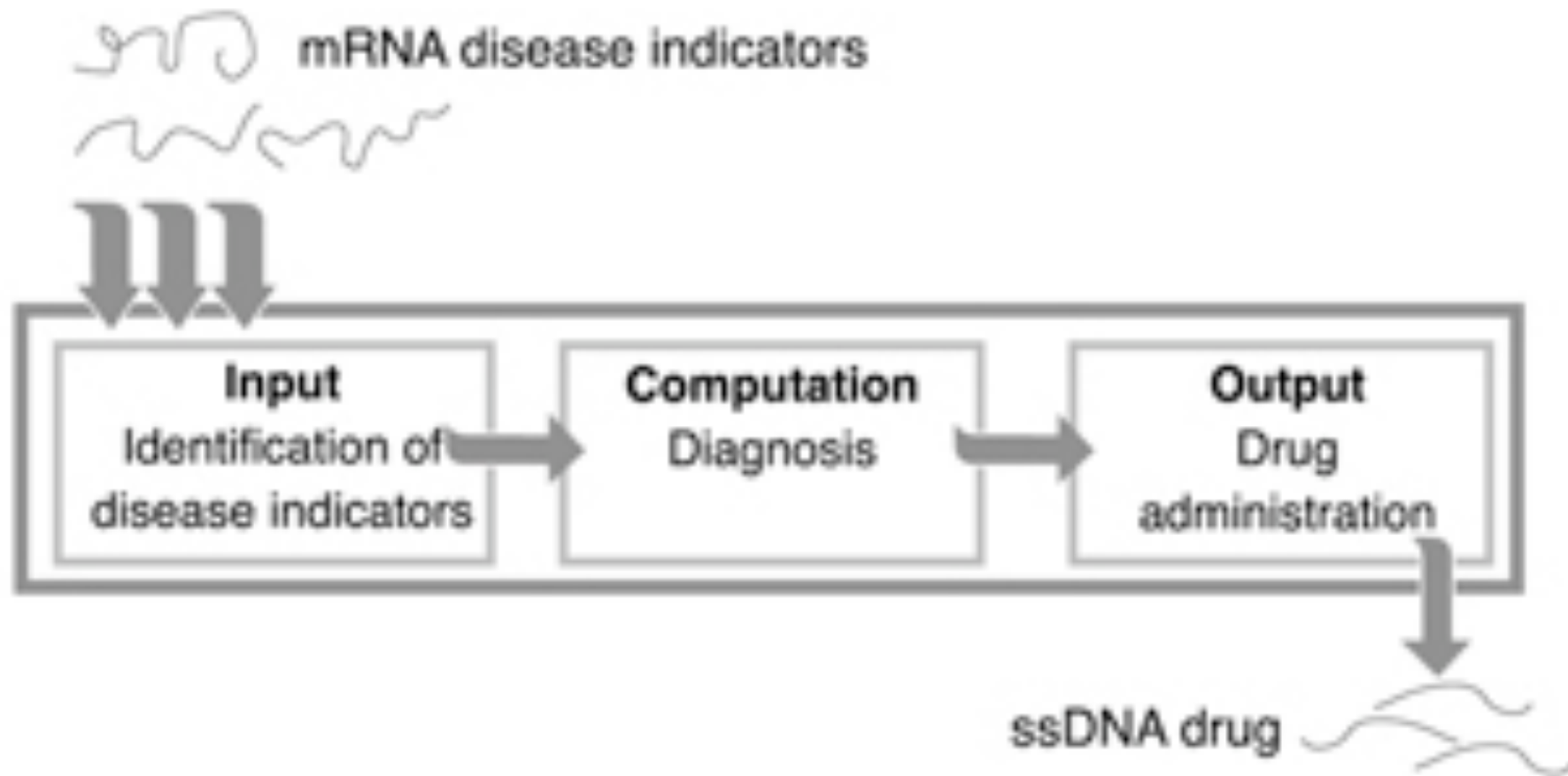
**Fármacos inteligentes:** Diagnóstico + fármaco

¿Cómo?

Autómatas biomoleculares

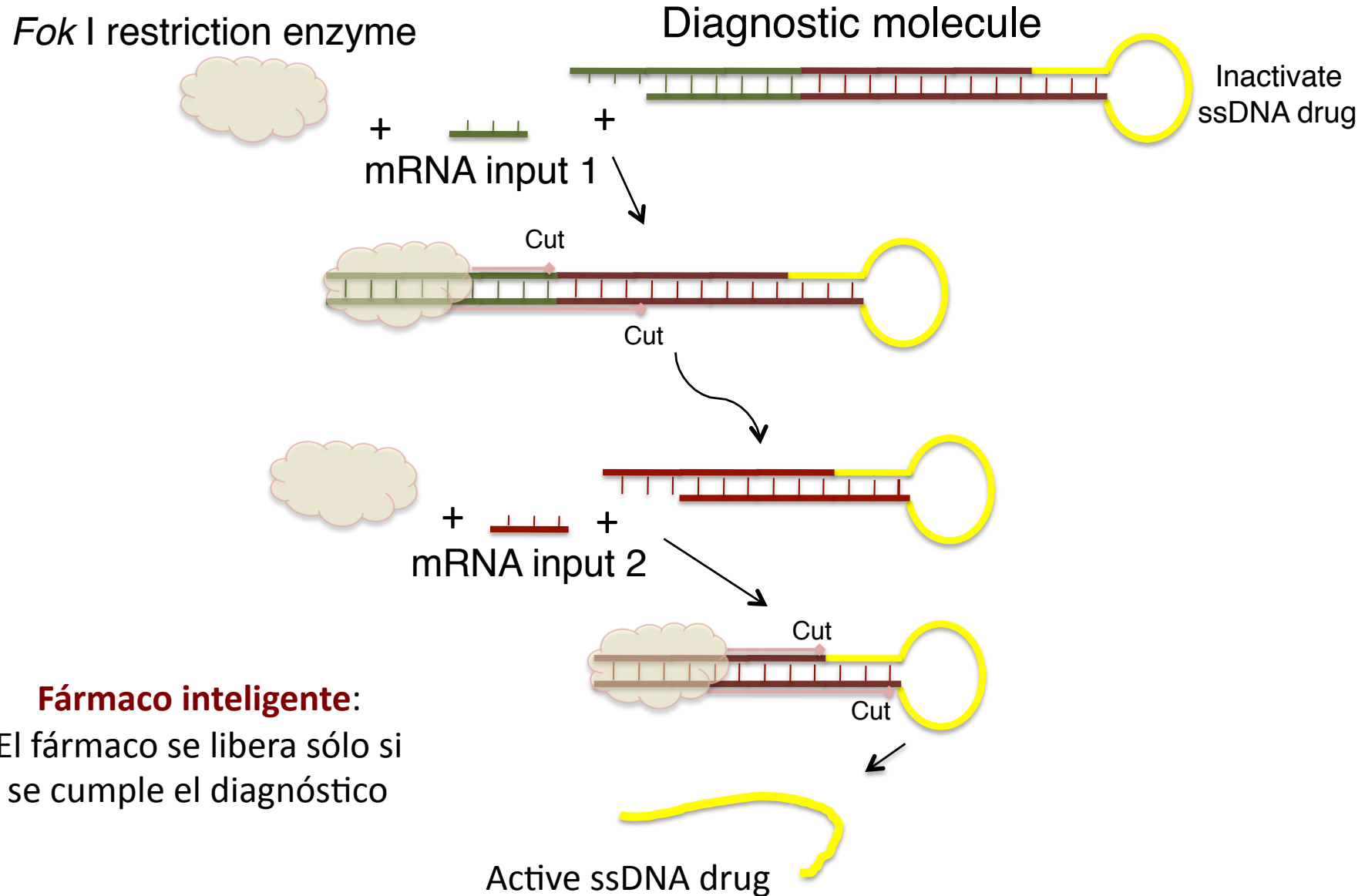
# Autómata biomolecular de Benenson

## a Modular design of the molecular computer



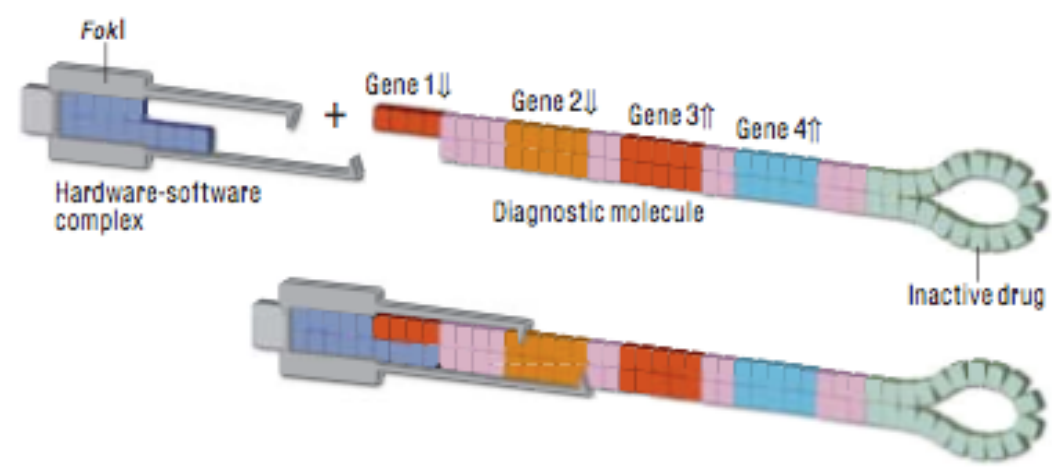
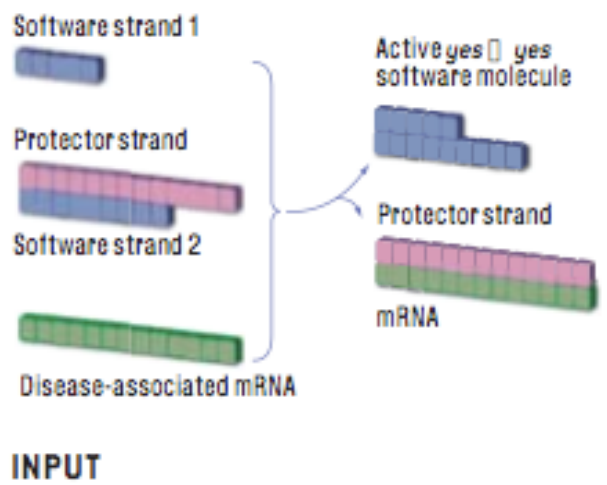
Benenson, Y. Et al. Programmable and autonomous computing machine made of biomolecules. Nature 414, 430-434 (2001).

# Autómata biomolecular que realiza un diagnóstico y libera un fármaco

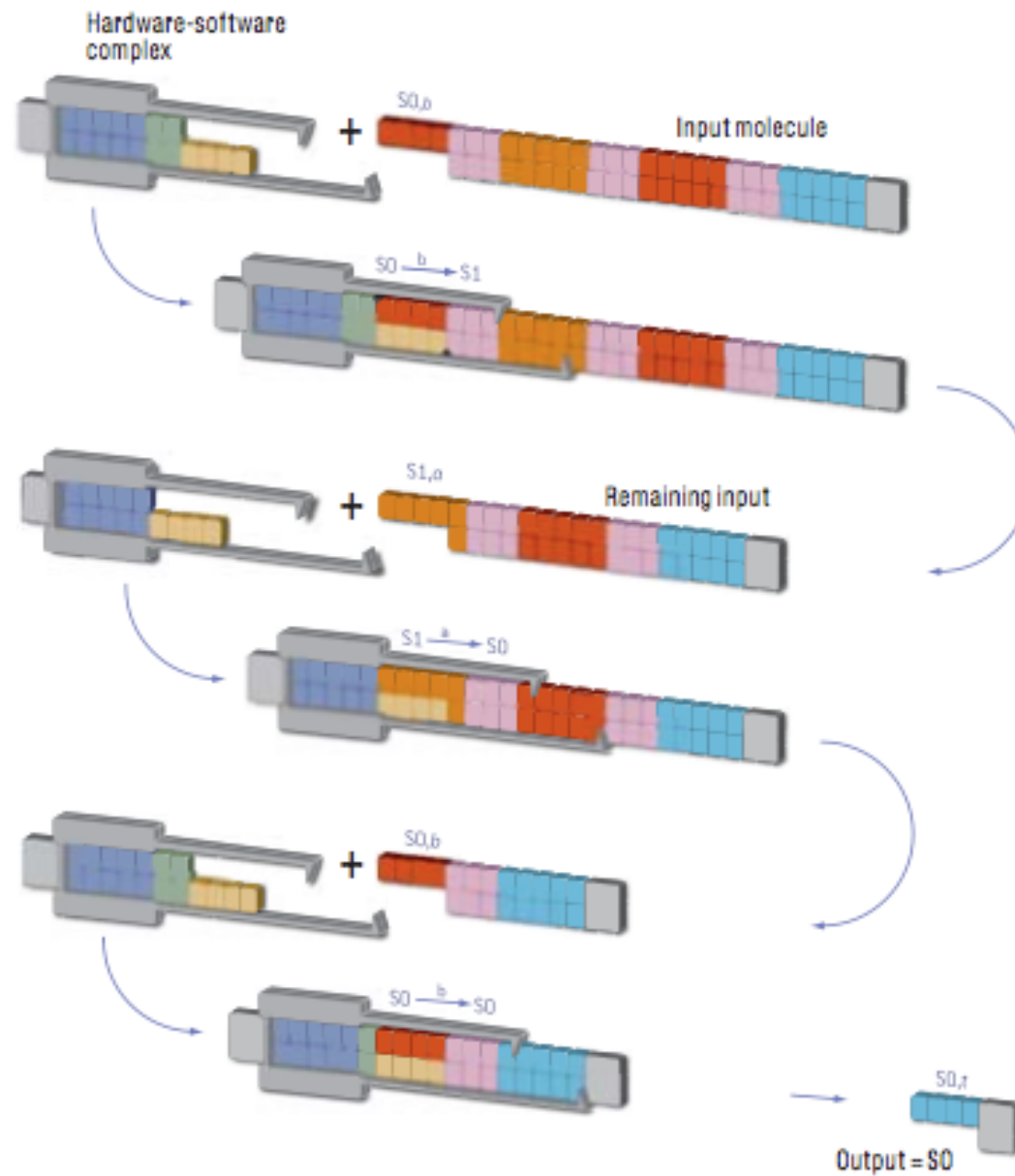


## Fármaco inteligente:

El fármaco se libera sólo si se cumple el diagnóstico



Shapiro, E. & Benenson, Y. Bringing DNA computers to life. *Scientific American* 294, 44-51 (2006).



Shapiro, E. & Benenson, Y. Bringing DNA computers to life. *Scientific American* 294, 44-51 (2006).



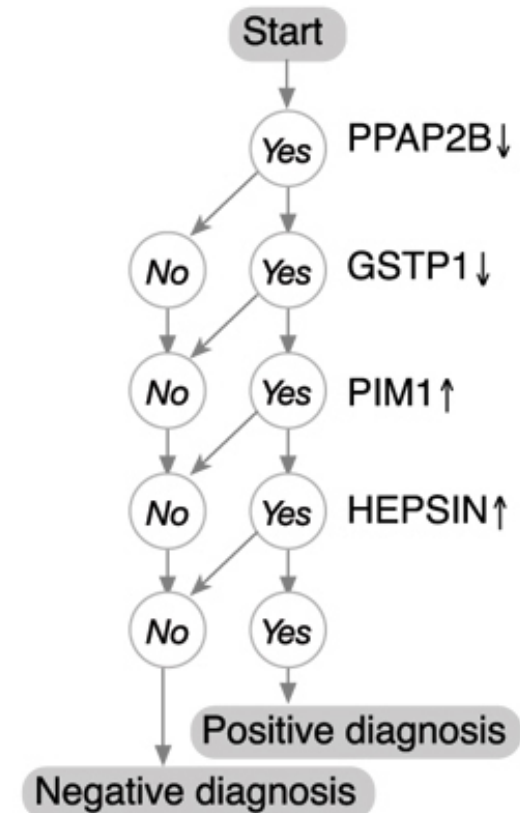
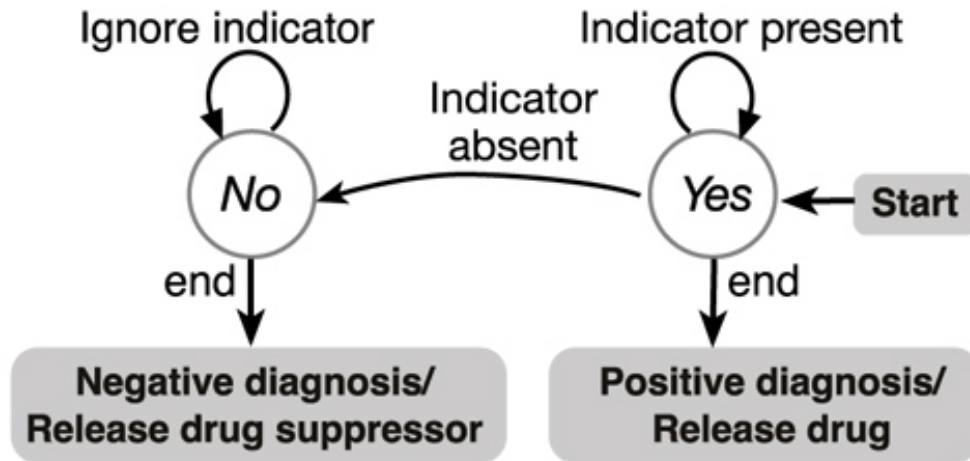
[http://www.wisdom.weizmann.ac.il/~udi/  
PressRoom/new\\_pages/abstract/abstract.swf](http://www.wisdom.weizmann.ac.il/~udi/PressRoom/new_pages/abstract/abstract.swf)

# Autómatas biomoleculares

Regla diagnóstico (sistema experto ☺): “fármaco inteligente”

PPAP2B↓ & GSTP1↓ & PIM1↑ & HEPSIN↑ → Administer GTTGGTATTGCACAT

## Autómata de diagnóstico



# Puertas lógicas de Erik Winfree utilizando ADN e hibridación competitiva

Winfree's group publications:

G. Seelig et al. Enzyme-Free Nucleic Acid Logic Circuits, *Science*, 314:1585-1588, 2006

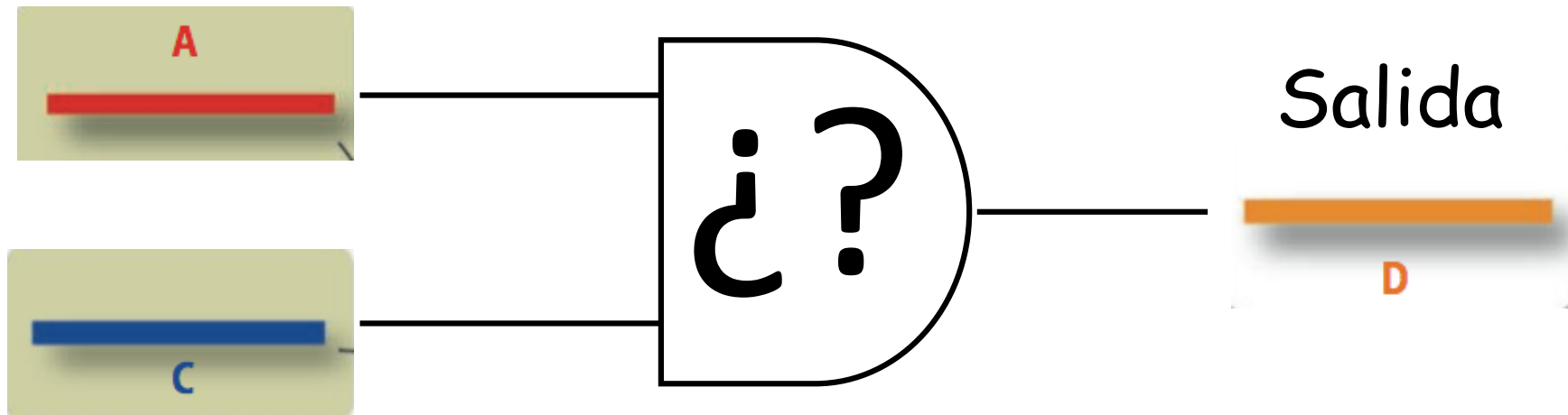
More recent work: Lulu Quian et al. Neural network computation with DNA strand displacement cascades, *Nature*: 475: 368-372, 2011

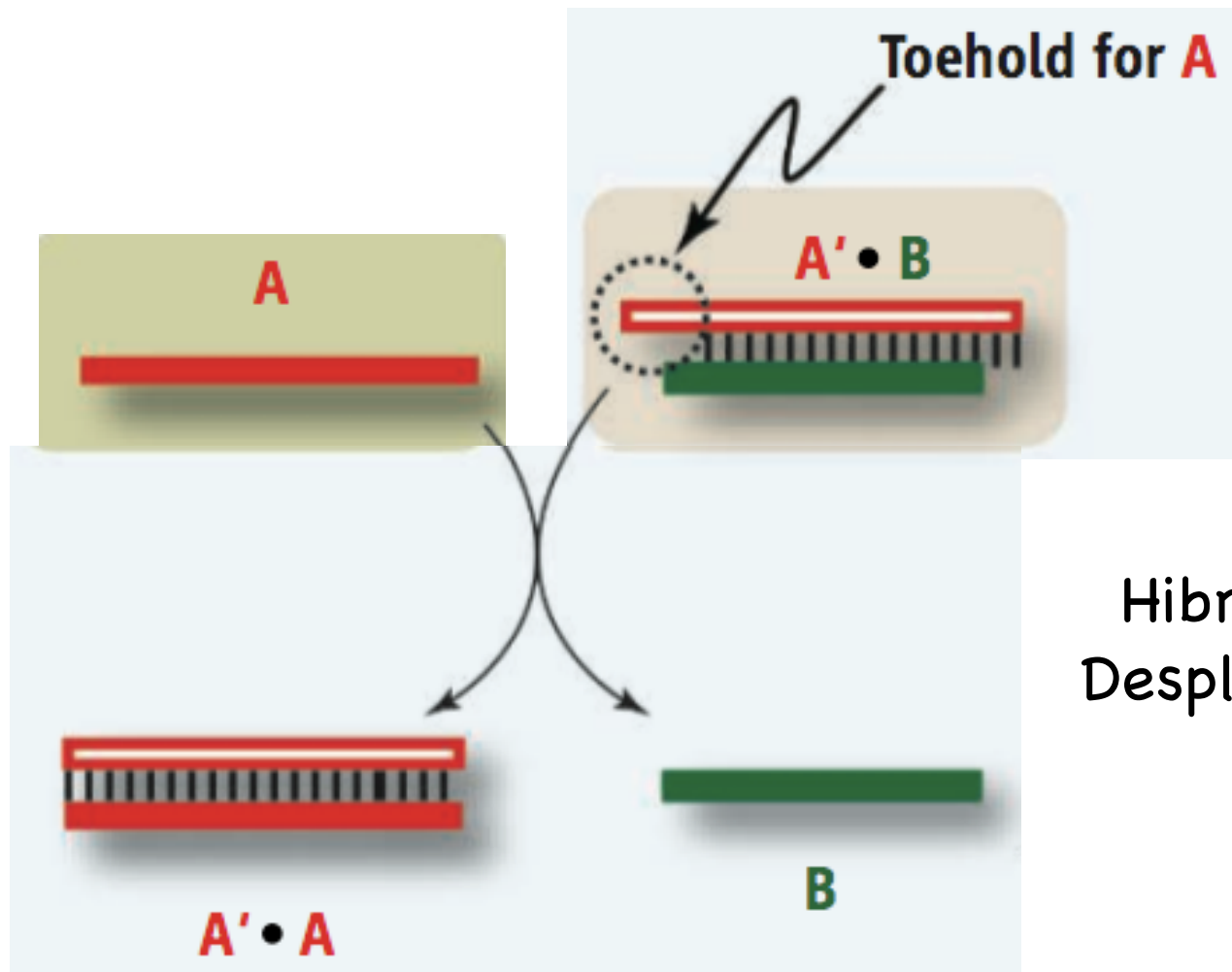
# Diseño de una puerta lógica AND que procesa hebras de ADN

$$\text{AND}(A,C) = D$$

Entradas A y C

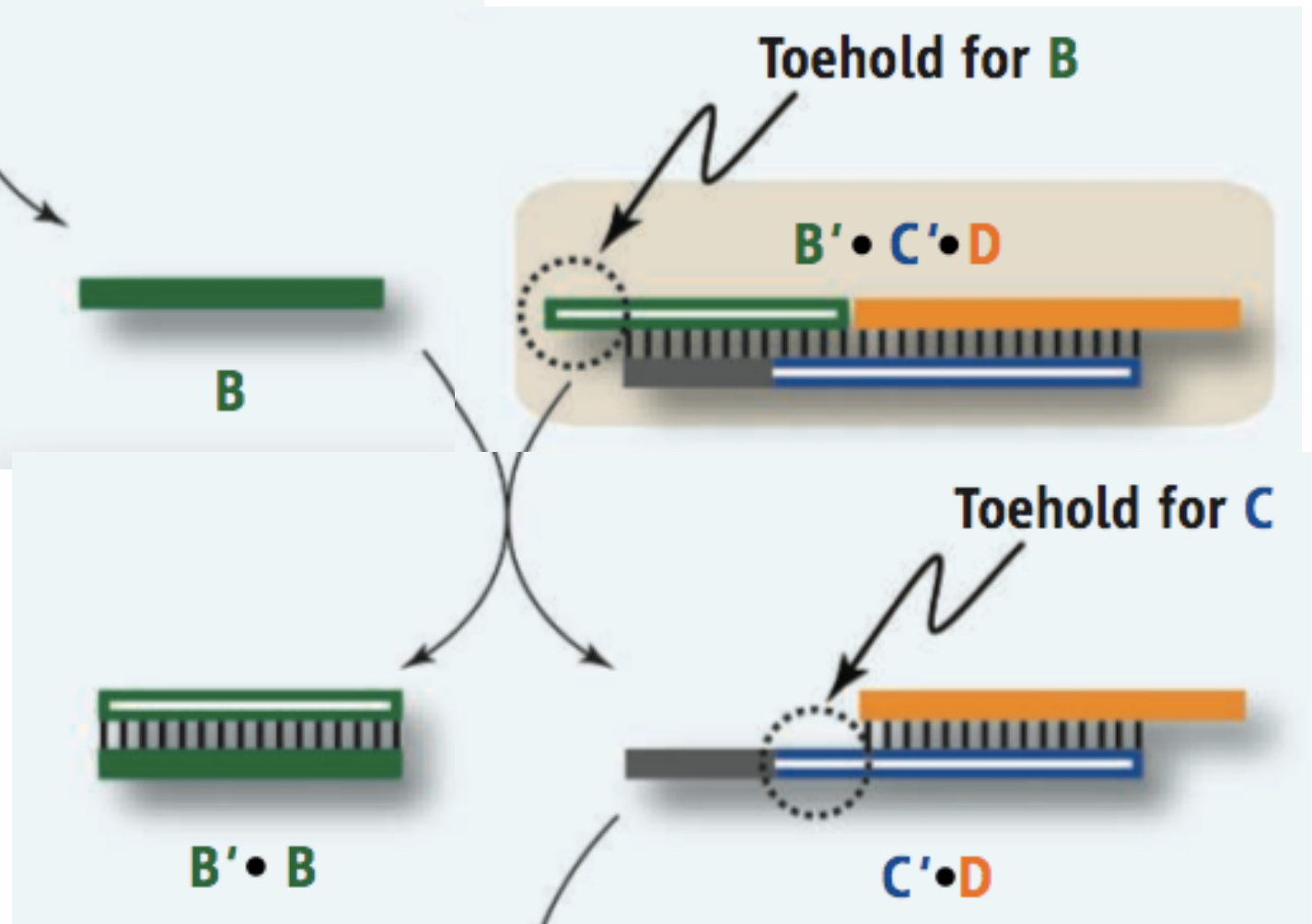
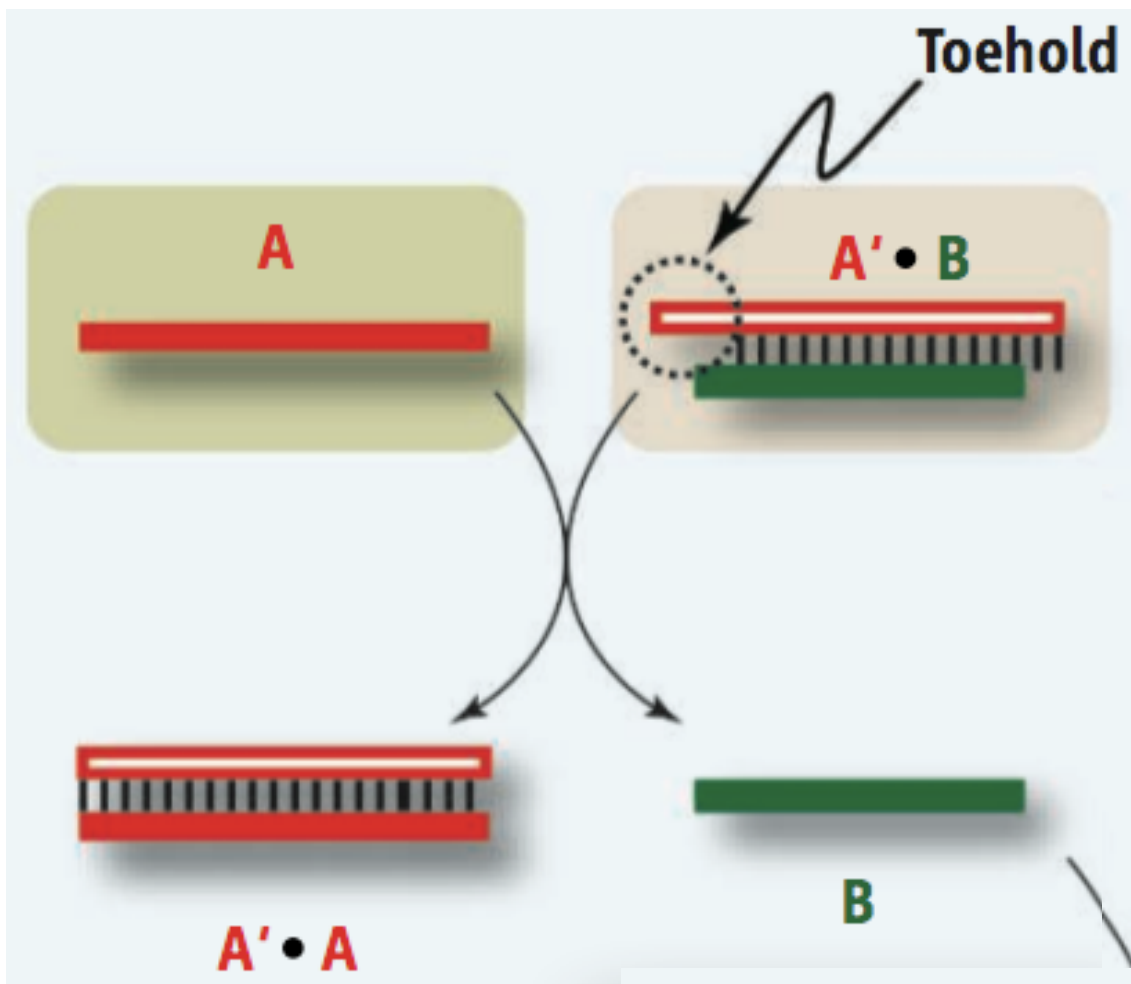
Puerta AND

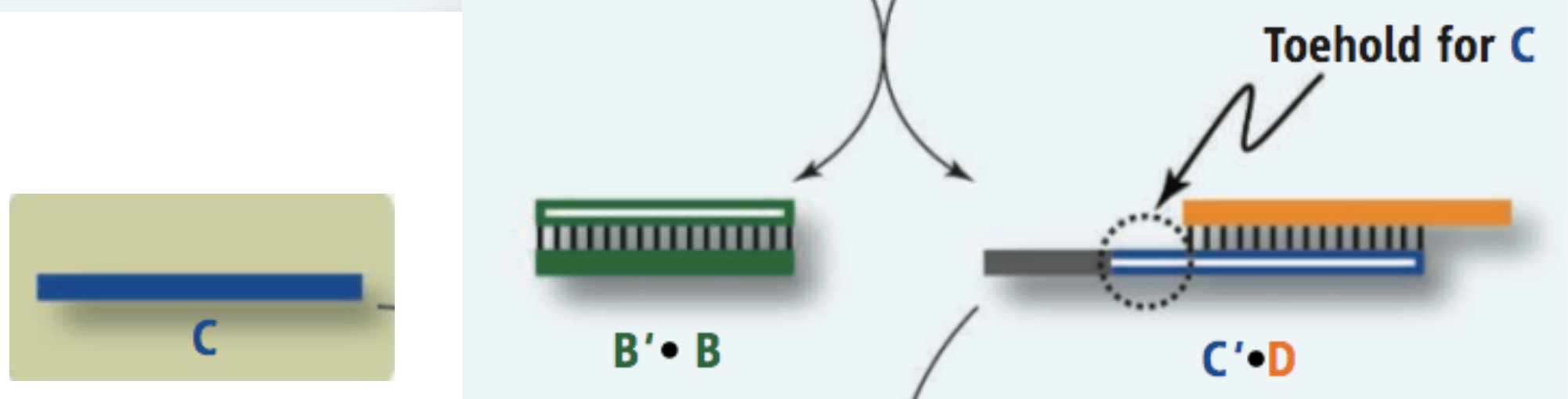
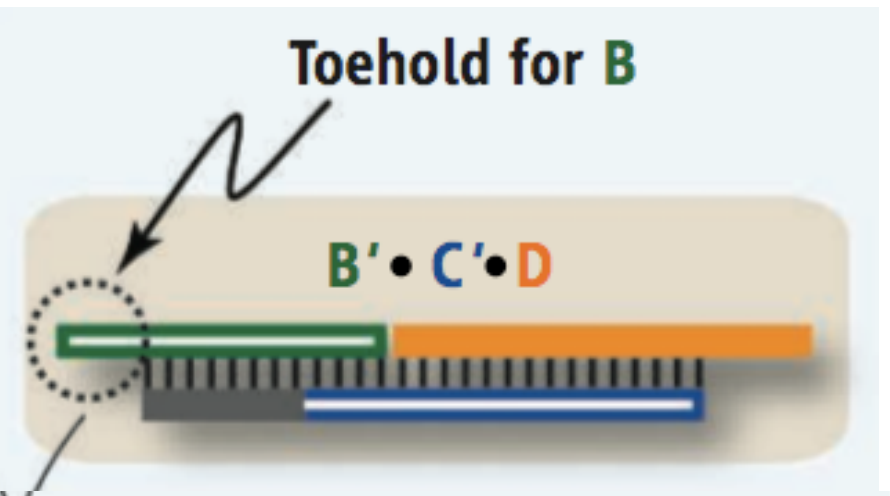
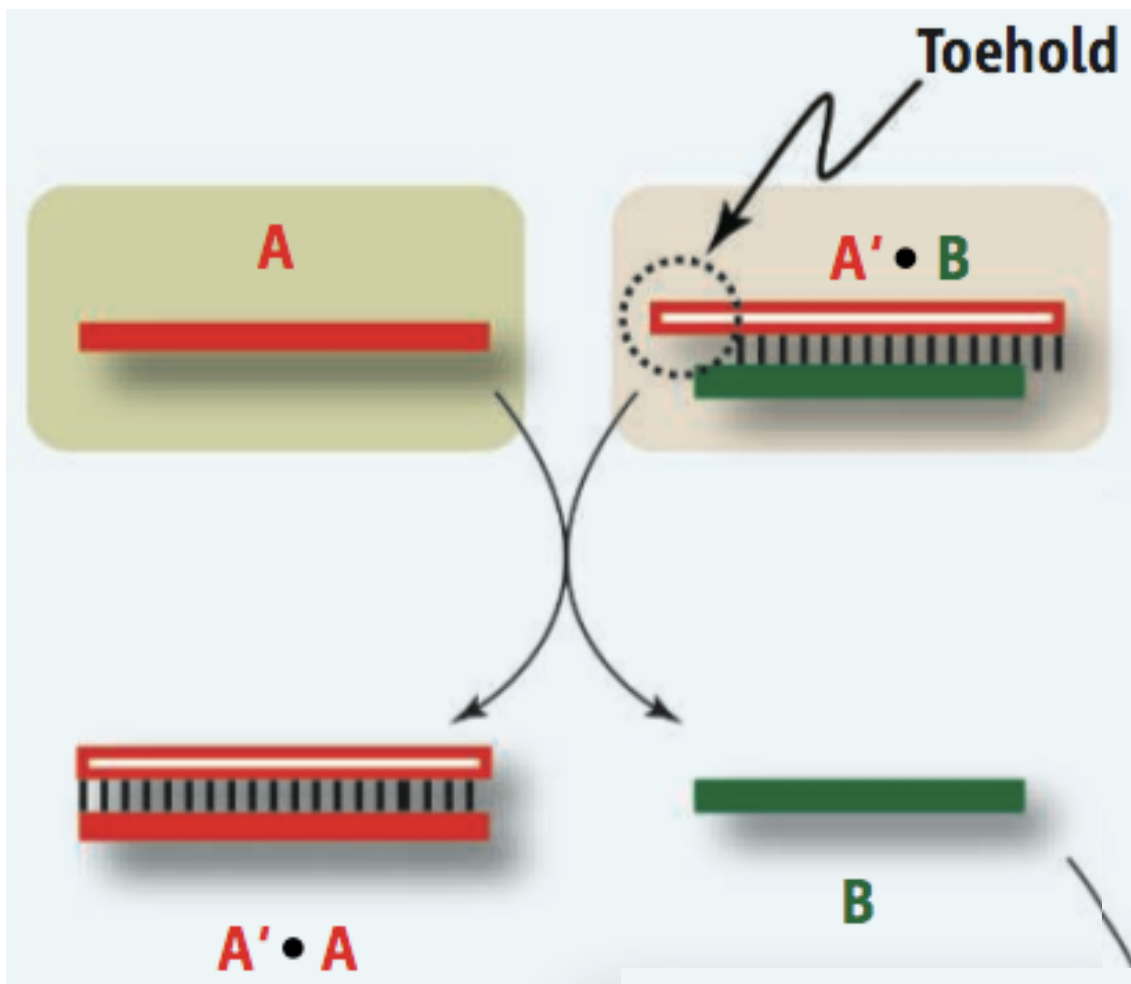


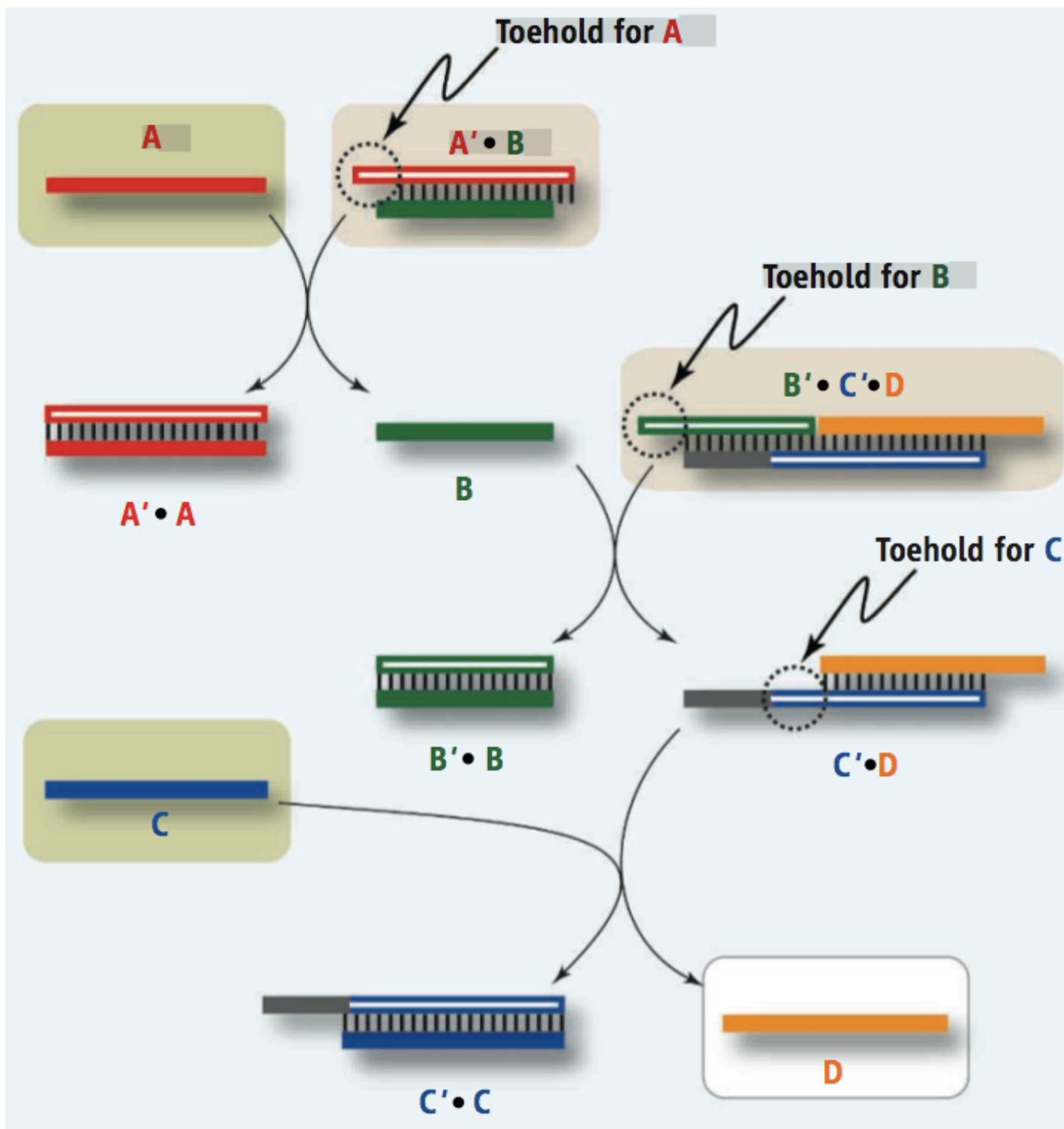


Hibridación competitiva  
Desplazamiento de hebras

Figure from: G. Seelig et al. Enzyme-Free Nucleic Acid Logic Circuits, *Science*, 314:1585-1588, 2006







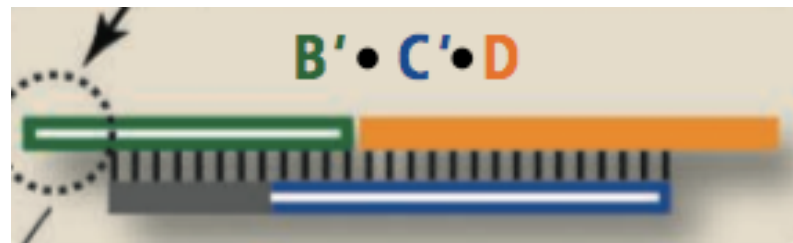
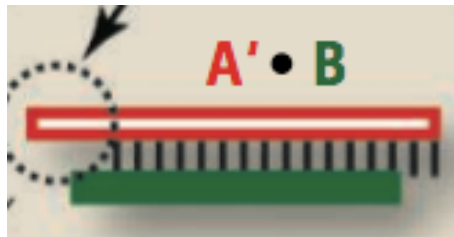


$$\text{AND}(A,C) = D$$

Inputs



AND gate



Output



Figures from: G. Seelig et al. Enzyme-Free Nucleic Acid Logic Circuits, Science, 314:1585-1588, 2006

# Inferencia con ADN

Procesamiento lógico o diagnóstico y análisis inteligente de ácidos nucleicos.

Modus ponens y Modus Tollens

Alfonso Rodríguez-Patón et al. Autonomous Resolution Based on DNA Strand Displacement. LNCS, 2011, vol. 6937, 190-203

Alfonso Rodríguez-Patón, Iñaki Sain de Murieta y JM Larrea. Patente española P201000694: “Dispositivos de ácidos nucleicos para la realización de inferencia lógica”.

Sainz de Murieta, I; Miró-Bueno, JM; Rodríguez-Patón, A.  
Biomolecular computers. Current Bioinformatics, 6, 173-184 (2011)

# Inferencia con ADN

## MODUS PONENS

Si P entonces Q

P

-----

Por lo tanto Q

## MODUS TOLLENS

Si P entonces Q

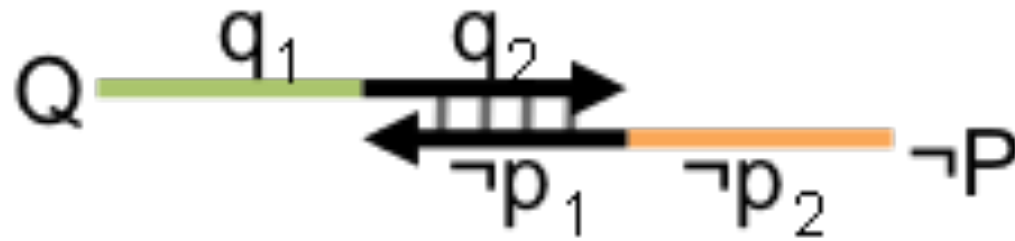
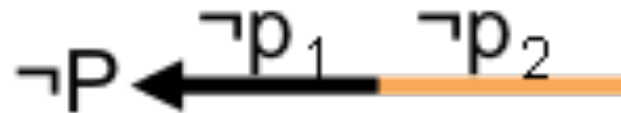
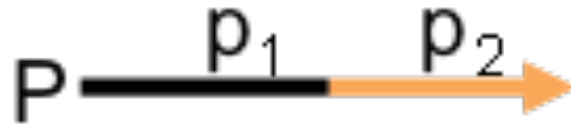
No Q

-----

Por lo tanto no P

# Inferencia con ADN

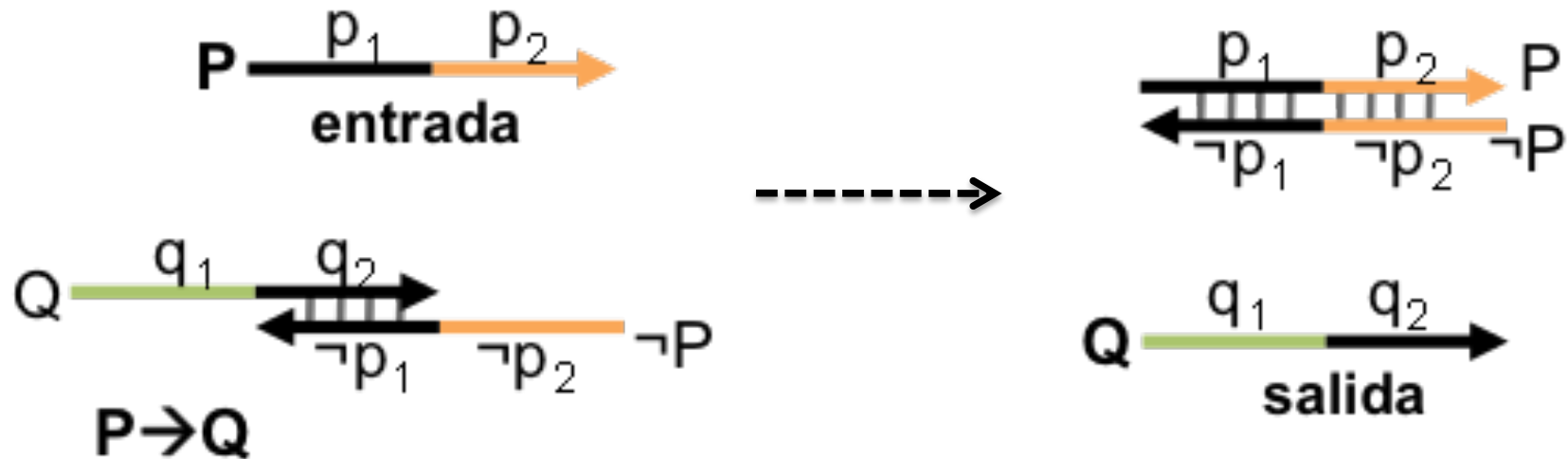
Codificación de variables y de reglas en ADN



$P \rightarrow Q$

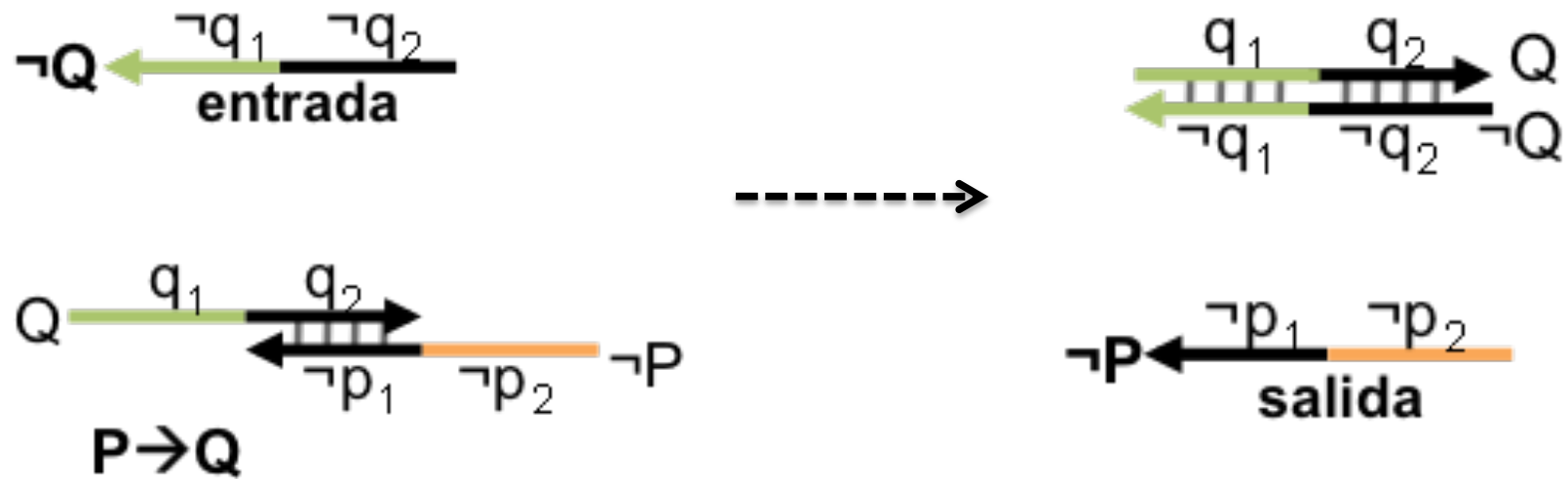
# Inferencia con ADN

**MODUS PONENS:** Si P entonces Q  
P  
Por lo tanto Q

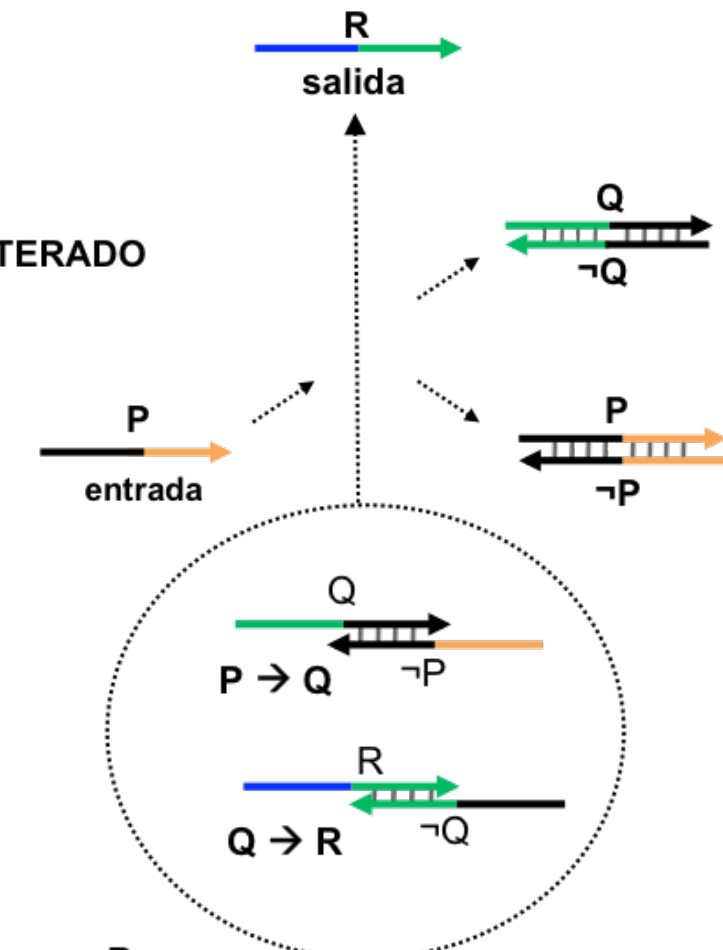


# Inferencia con ADN

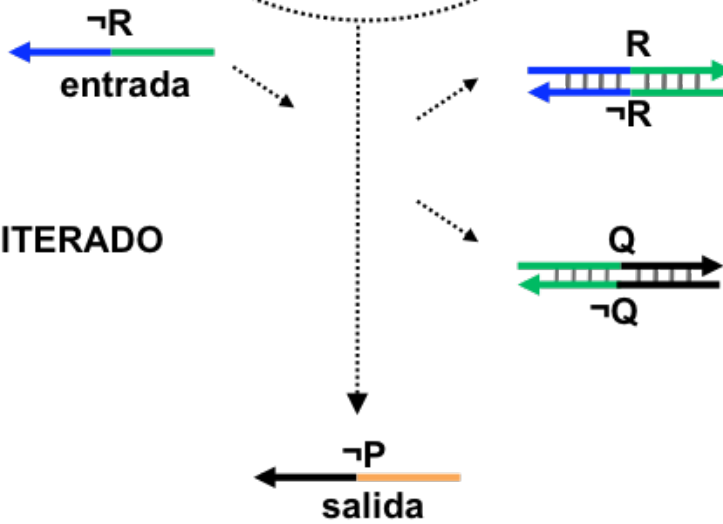
**MODUS TOLLENS:** Si P entonces Q  
No Q  
Por lo tanto no P



MODUS PONENS ITERADO



MODUS TOLLENS ITERADO



# Nanotecnología con ADN

Robots moleculares: DNA spiders,  
walkers, tweezers, gliders

Auto-ensamblamiento de ADN:  
Origamis con ADN

Biology is the Nanotechnology that works” Tom Knight, MIT.



Visionario: Richard Feynman  
Pionero de la DNA Nanotech: Nadrian  
Seeman



# (Bio)-Nanotecnología

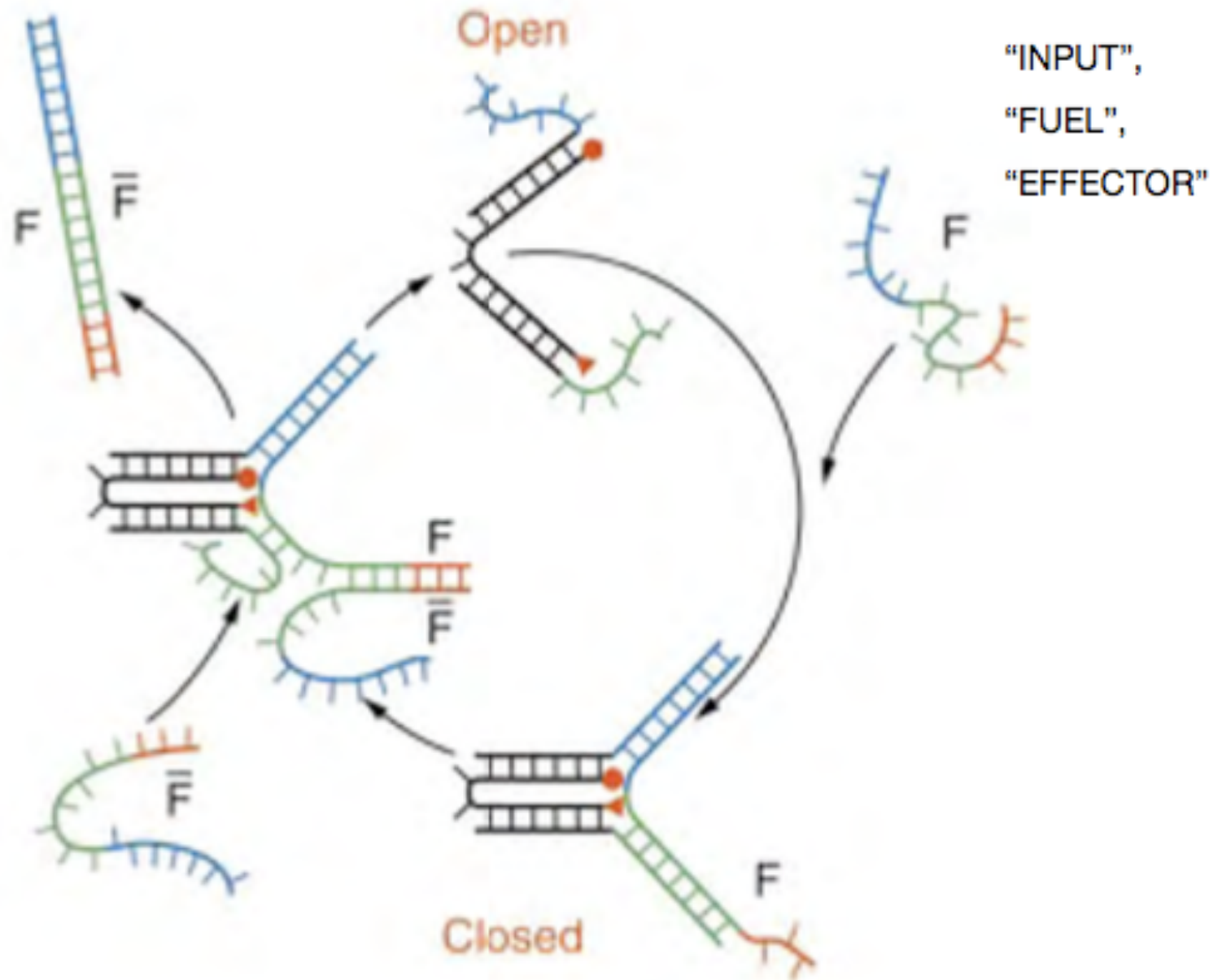
1 metro =  $10^3$  mm =  $10^6$   $\mu$ m =  $10^9$  nm

E. Coli: 2x1  $\mu$ m.

Hélice de ADN: 2.5 nm ancho. Largo: Hasta 3m.

Proteína: 1 a 100 nm.

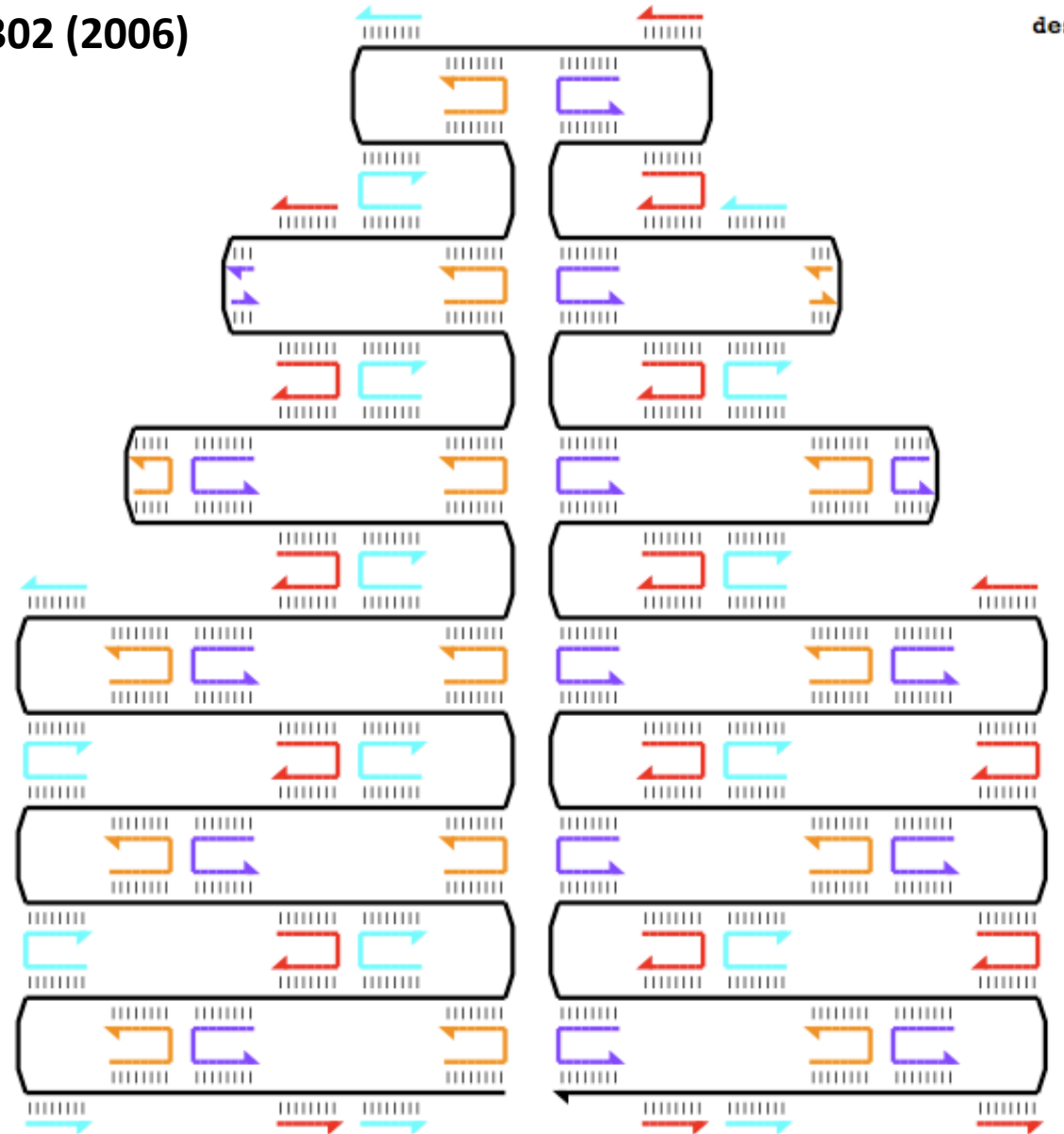
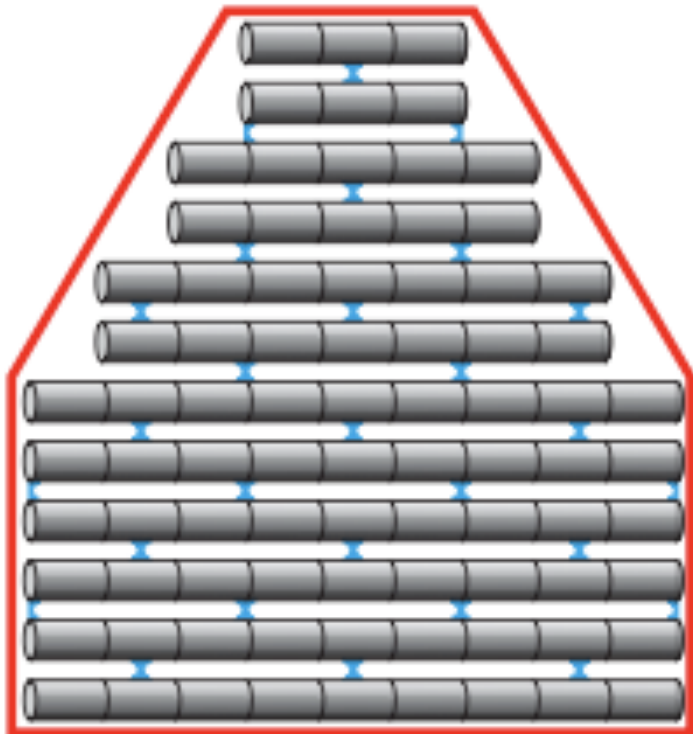
# Pinzas de ADN (Yurke, Simmel. Nature, 2000)



# Folding DNA to create nanoscale shapes and patterns

Paul W. K. Rothemund<sup>1</sup>

Nature **440**, 297-302 (2006)



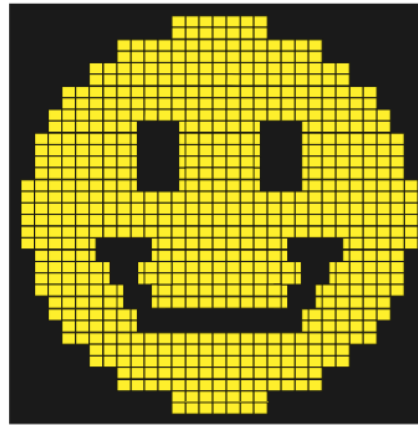
des

# Proceso de diseño de un origami con ADN

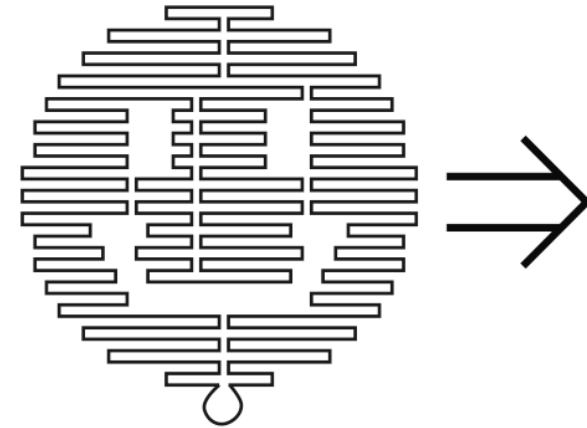
target shape



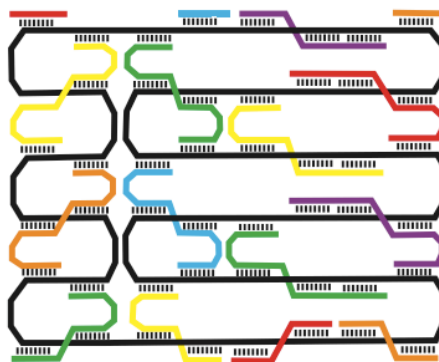
discrete approximation



folding path

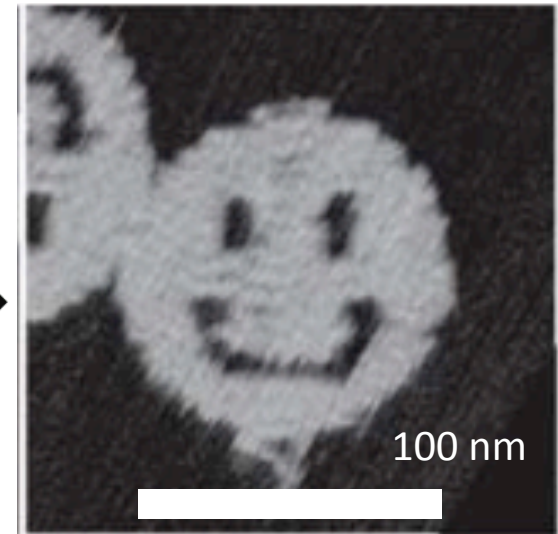
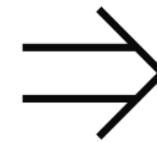


staple strand geometry



DNA sequences

```
s0m25k, A1, AAACACCAGAACGAGTAGTAAATTGGGCTTGA
s10m15g, B1, ATATATTTTAG
s10m17g, C1, TCCTTGAAAACAACCTTTTTCAA
s10m19g, D1, ATTCATTTCAATTCCCTTAGAA
s1m0j, E1, ACGCAAAGACACCACGGAATAAGTTTATTTTGG
s1m10e, F1, CGTACTCAACCCTCAGAACCGCCAAGCAAGCC
s1m12e, G1, CAATAGGAAGTACAAACTACAACGAGCGTAAC
s1m14e, H1, GATCTAAAATTTTCTGTATGGGATGTGAGAAT
s1m16e, A2, AGAAAGGATGCTTTCGAGGTGAATGACAATGA
s1m18e, B2, CAACAACCATCGGAACGAGGGTAGTTTTCAT
s1m20e, C2, GAGGAAGTAAAACACTCATCTTTGGTACAACG
s1m22e, D2, GAGATTTGGACCAACTTTGAAAGAAGGCTGGC
s1m24e, E2, TGACCTTCGGCTTGCCCTGACGAG
s1m28i, F2, GCAACAGTAGGCGGTGAGTATTAAGGGAGAGG
s1m2i, G2, ATGTTAGCGATTAAGACTCCTTATACCGATTG
s1m30i, H2, GTAATAAAGCAGATTCACCAGTCAATAGGGTT
s1m32i, A3, AGTAGAAGATTAGTAATAACATCAATCGGAAC
...
```

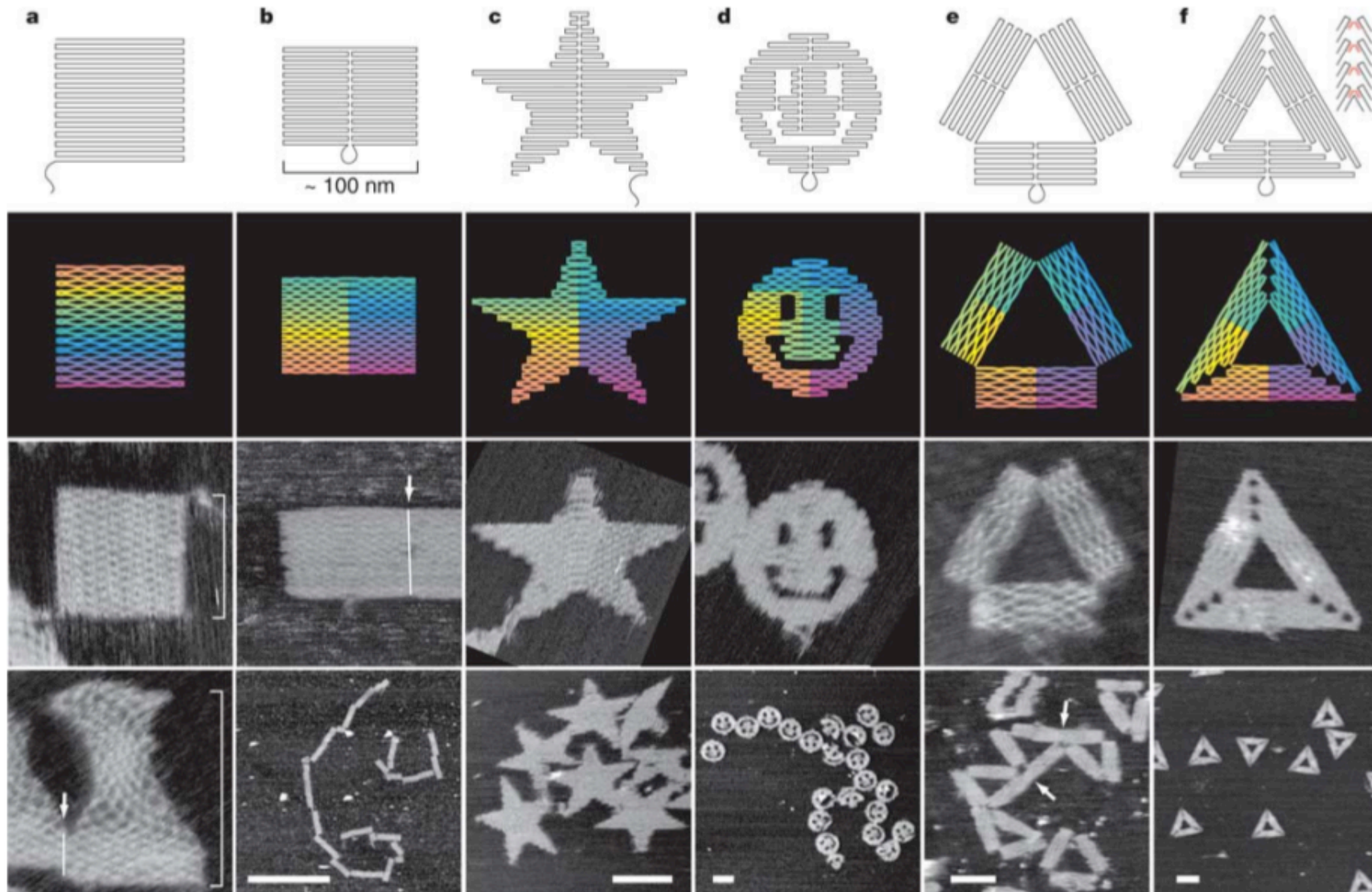




# Folding DNA to create nanoscale shapes and patterns

Paul W. K. Rothemund<sup>1</sup>

Nature **440**, 297-302 (2006)



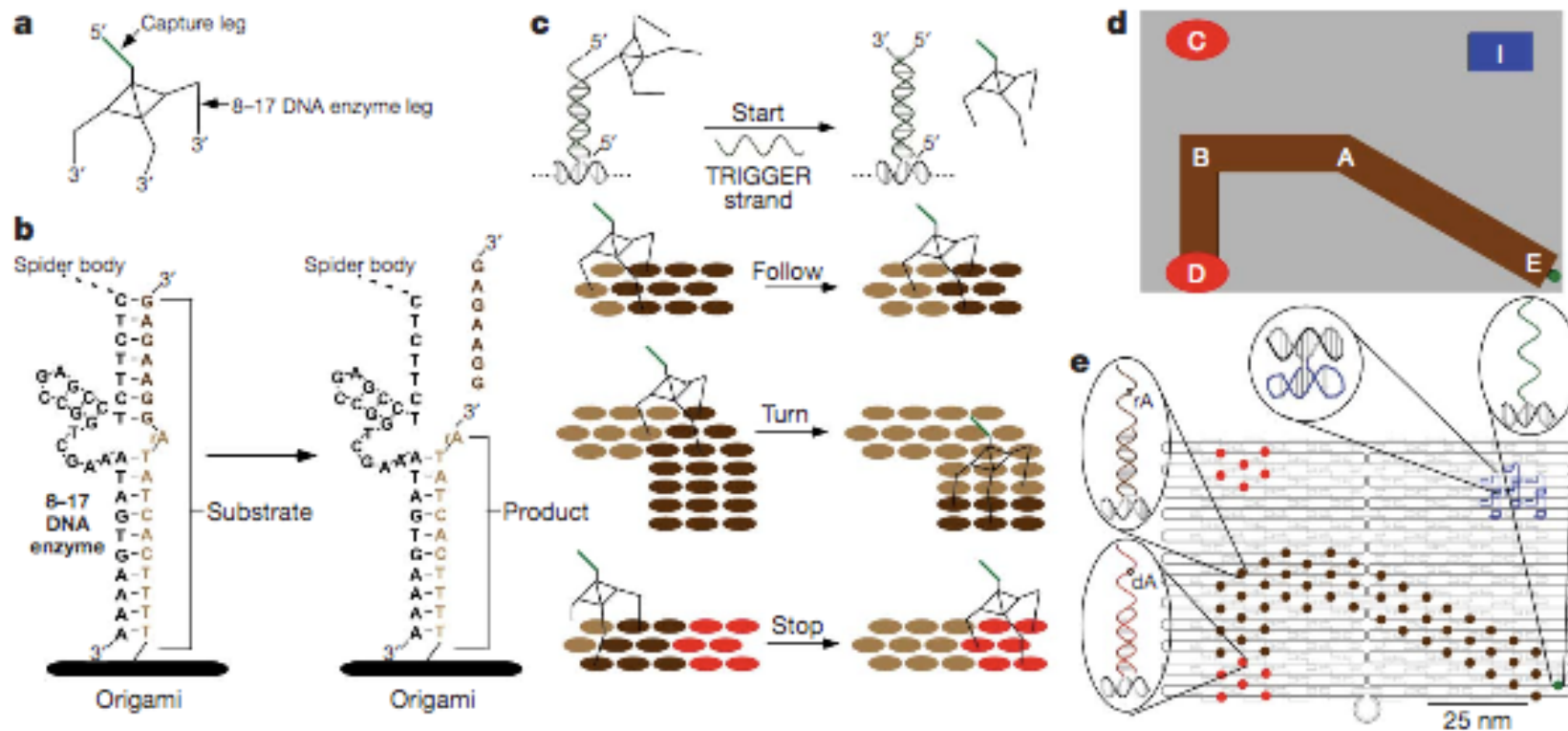
# DNA Origami

- <http://wyss.harvard.edu/viewmedia/32/dna-origamibased-selfassembly;jsessionid=C5B9006BAB1C5D3E8121DBDF8E1EB5FC.wyss2>

# Molecular robots guided by prescriptive landscapes

Kyle Lund<sup>1,2</sup>, Anthony J. Manzo<sup>3</sup>, Nadine Dabby<sup>4</sup>, Nicole Michelotti<sup>3,5</sup>, Alexander Johnson-Buck<sup>3</sup>, Jeanette Nangreave<sup>1,2</sup>, Steven Taylor<sup>6</sup>, Renjun Pei<sup>6</sup>, Milan N. Stojanovic<sup>6,7</sup>, Nils G. Walter<sup>3</sup>, Erik Winfree<sup>4,8,9</sup> & Hao Yan<sup>1,2</sup>

Nature 465, 205 (2010)





Biology is the Nanotechnology that works” Tom Knight, MIT.