

CALCUL MOLÉCULAIRE

OU JETEZ VOS PCs DANS LA POUBELLE

Ordinateur moléculaire

Pourquoi ?

- vitesse
- coûts
- «car c'est là»
- médecin dans la cellule (**Labo de Shapiro**) :
observe les chemins de signalisation, produit des signaux (c-à-d enzymes) lui-même, traitement de maladies au niveau cellulaire

Implantation par biotech

Rappel : notre répertoire d'outils de biotechnologie

- hybridation
- PCR
- enzymes de restriction
- clonage
- electrophoresis
- séquençage

Encodage

Encodage d'information par ADN : c'est presque trivial :)

Exemple : message secret (*microdot*)

A=CGA K=AAG

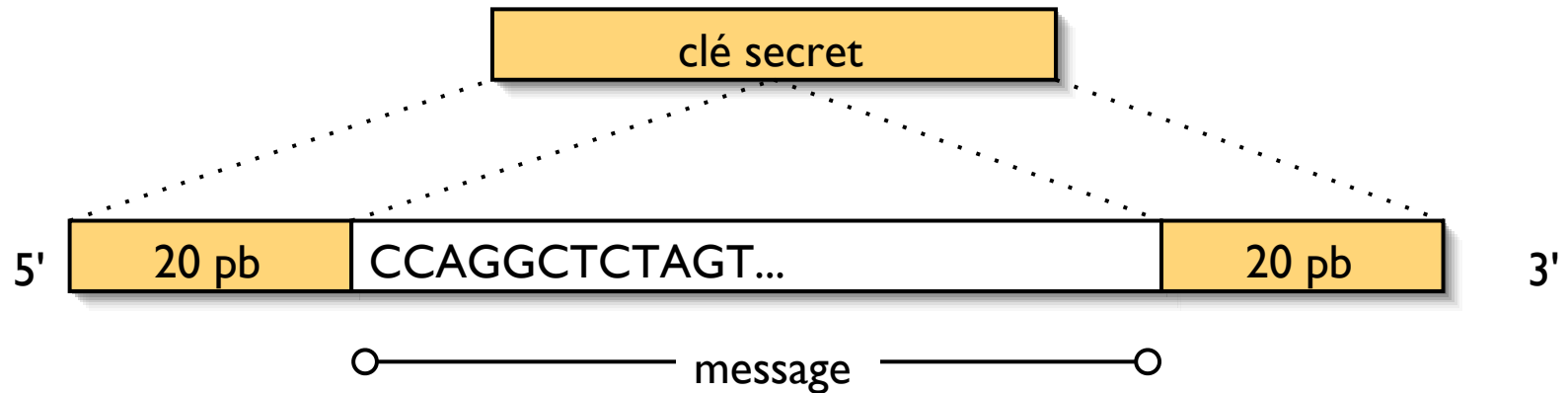
B=CCA L=TGC

C=GTT M=TCC

...

«Bonjour la tristesse» → CCAGGCTCTAGT...

Message secret - 2

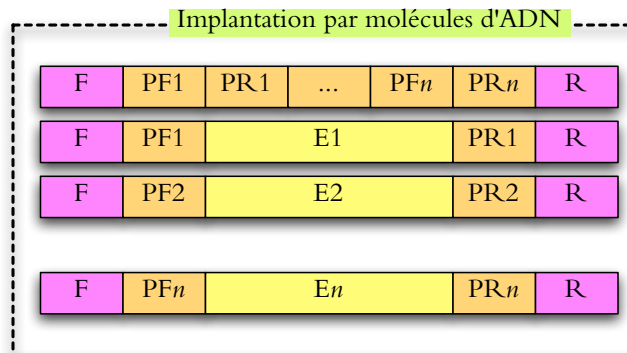
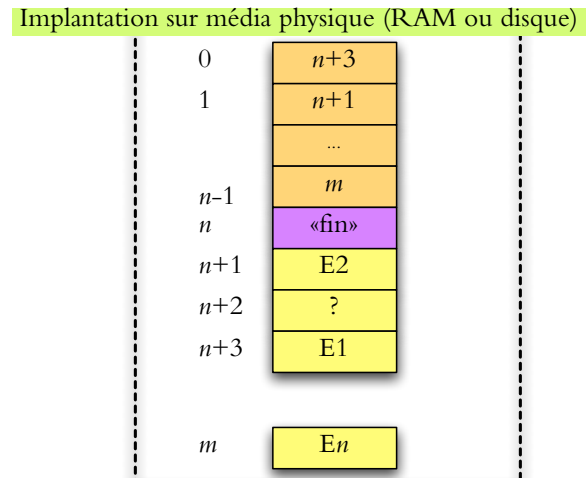
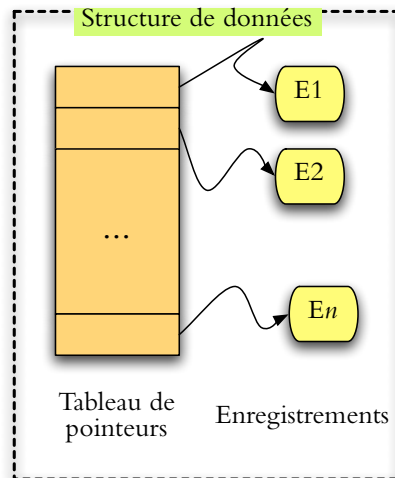


mélanger le message secret avec d'autres molécules ADN

amplification du message par PCR — il faut savoir le clé secret (nombre de possibilités : 4^{40})

Clelland & al. *Nature* 399 :533 (1999)

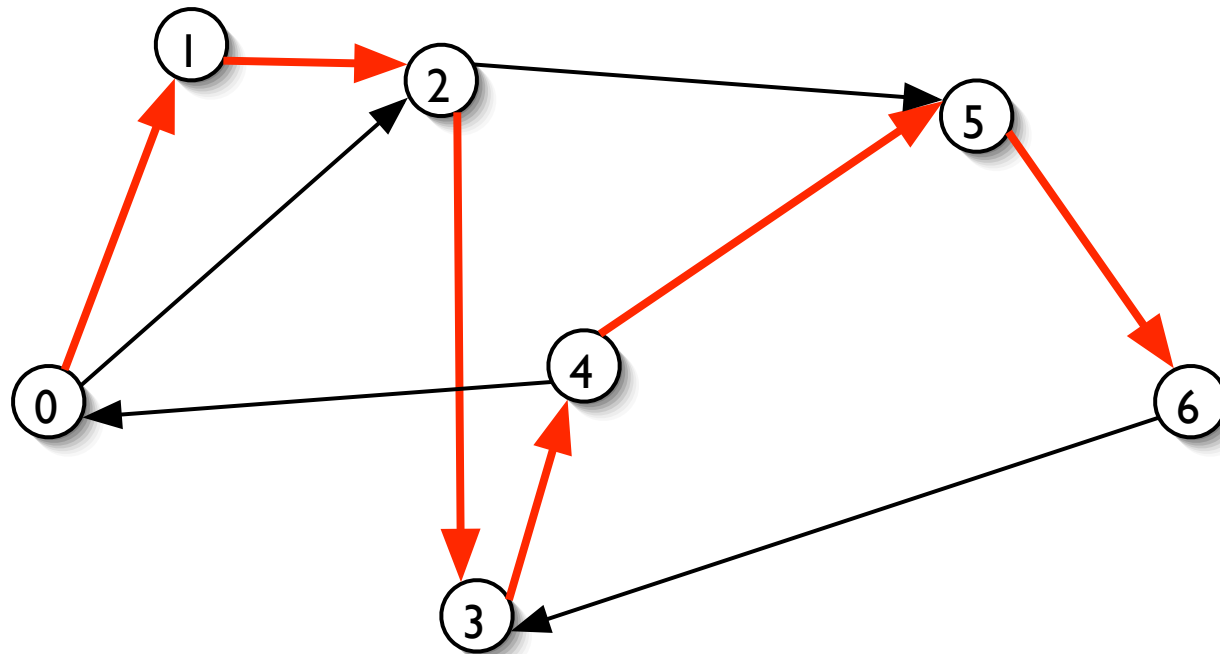
Stockage d'information



Bancroft & al. *Science* 293 :1763 (2001)

Encodage de graphes

Expérience classique de Leonard Adleman (1994) : chemin Hamiltonien

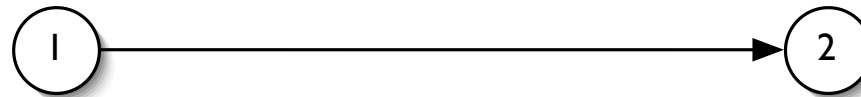


Encodage de graphes - 2

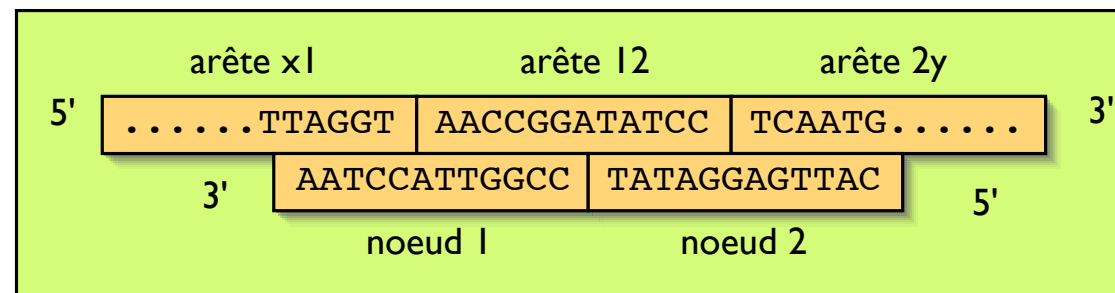
Noeuds encodés par oligonucléotides $O_i = O_i^{(g)} O_i^{(d)}$, arêtes encodés par oligos
 $O_{ij} = O_i^{(d)} O_j^{(g)}$

mélanger : $\overline{O_i}$, pour $i = 1, \dots, 5$ (complément Watson-Crick) ; O_{ij}

Encodage de graphes - 3



0_{12}



chemin encodé

Encodage de graphes - 4

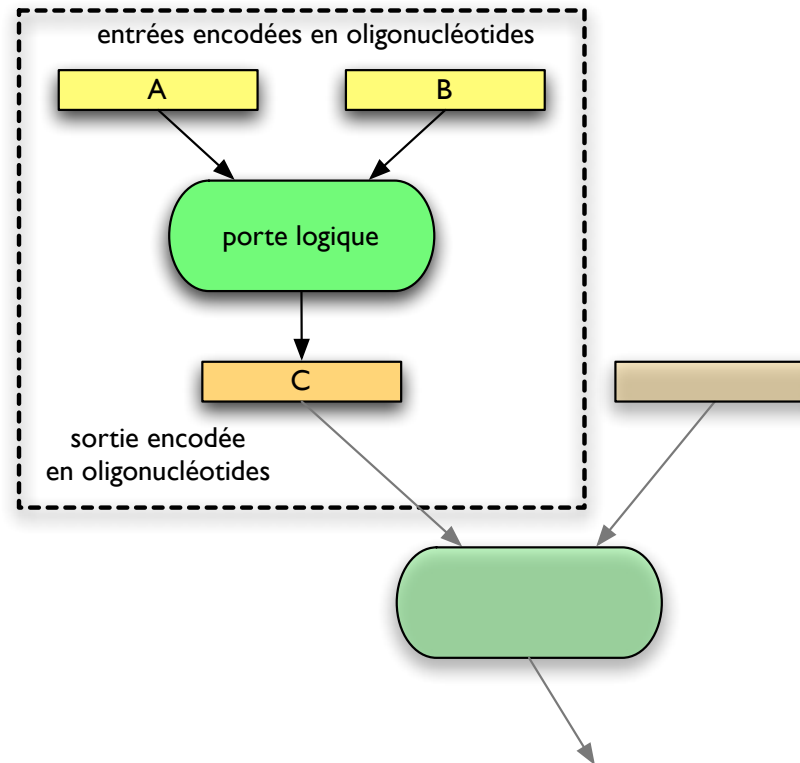
chaque molécule formée correspond à un chemin :

1. amplification par PCR (filtrer les chemins $0 \rightarrow 6$)
2. mesurer la longueur
3. filtrer par chaque O_i pour vérifier sa présence

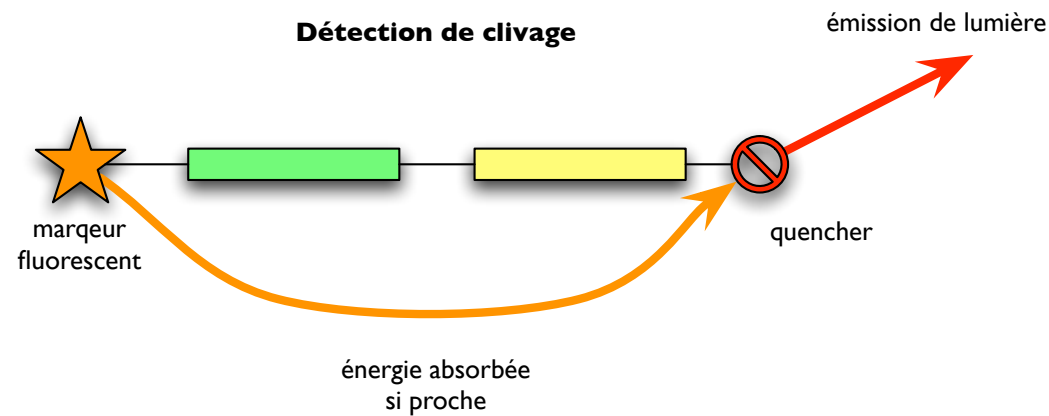
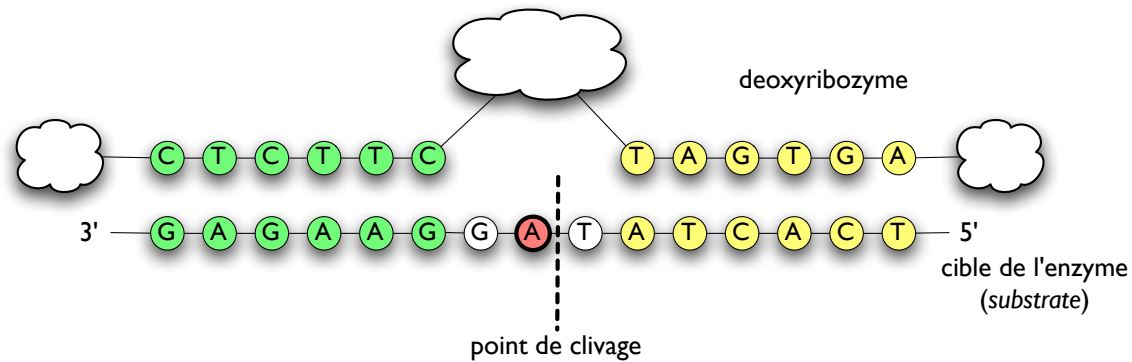
Problèmes : espace exponentiel (nbre exponentiel de chemins), modèle de calcul (formation de molécules en une dimension à partir d'un ensemble de pièces) est décidable $\Rightarrow$ n'est pas aussi puissant que la machine de Turing

Portes logiques

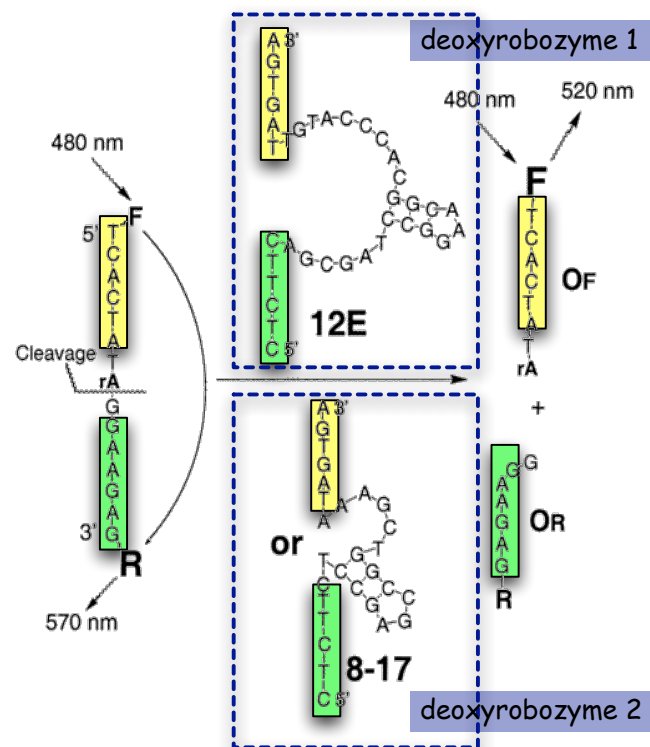
concept de base



Implantation — components de base

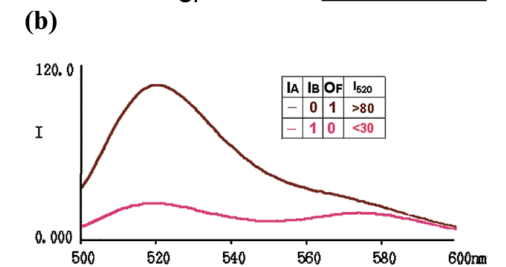
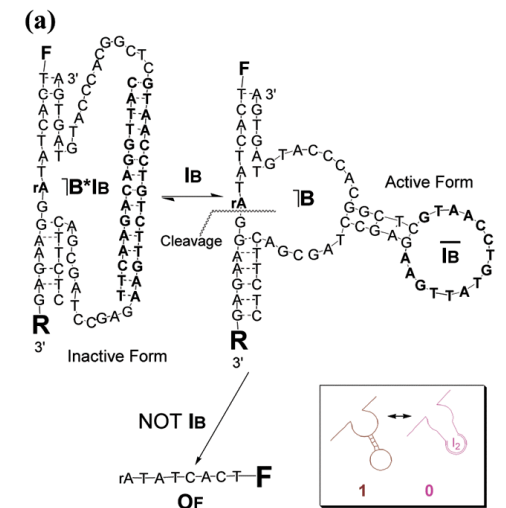
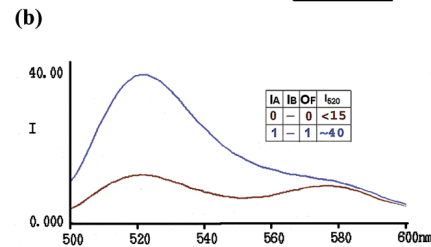
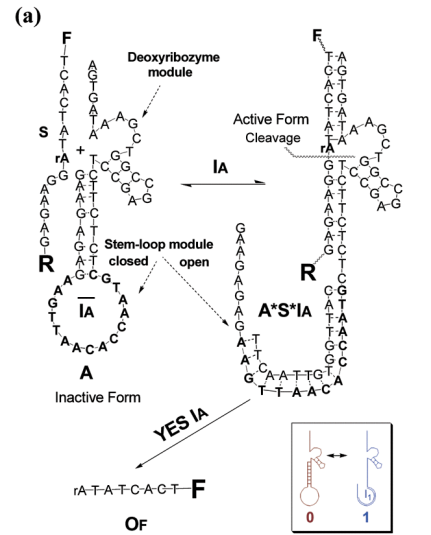


Détails



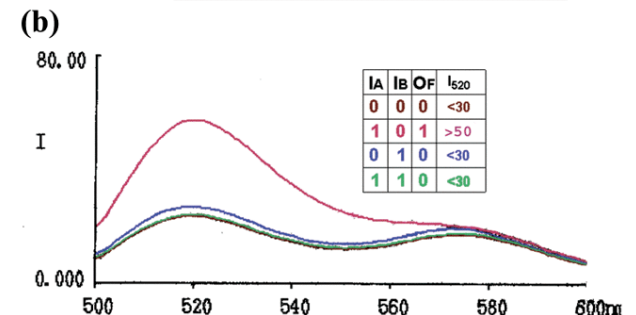
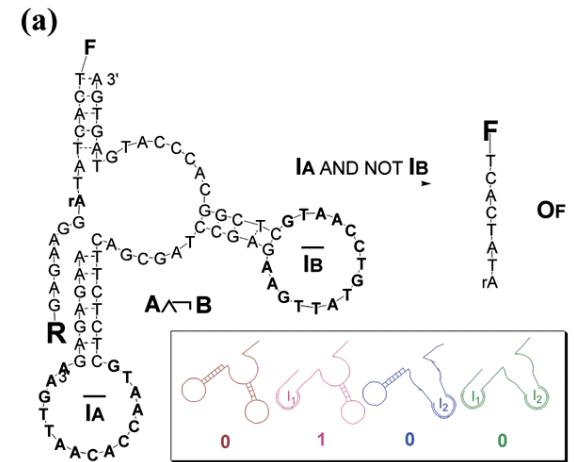
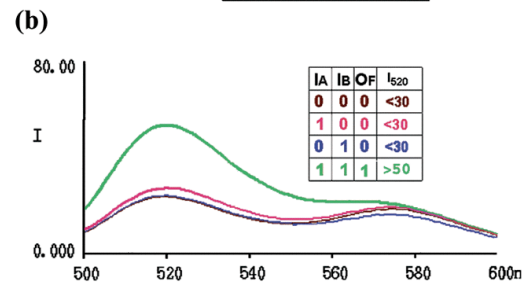
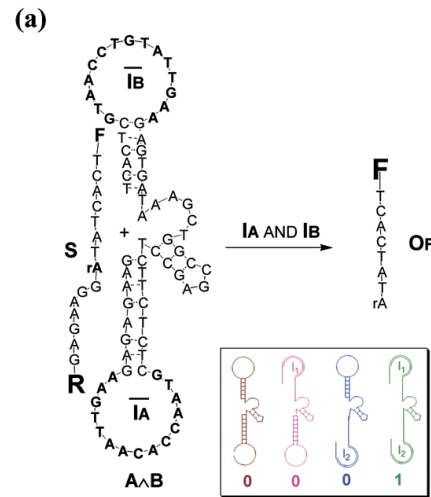
Stojanovic, Mitchell, Stefanovic. *J Am Chem Soc* 124 :3555 (2002)

Portes logiques : OUI et NON



OU logique : mixte de OUI ; NON-ET logique : mixte de NON (de Morgan)

Portes logiques : ET et ET-NON

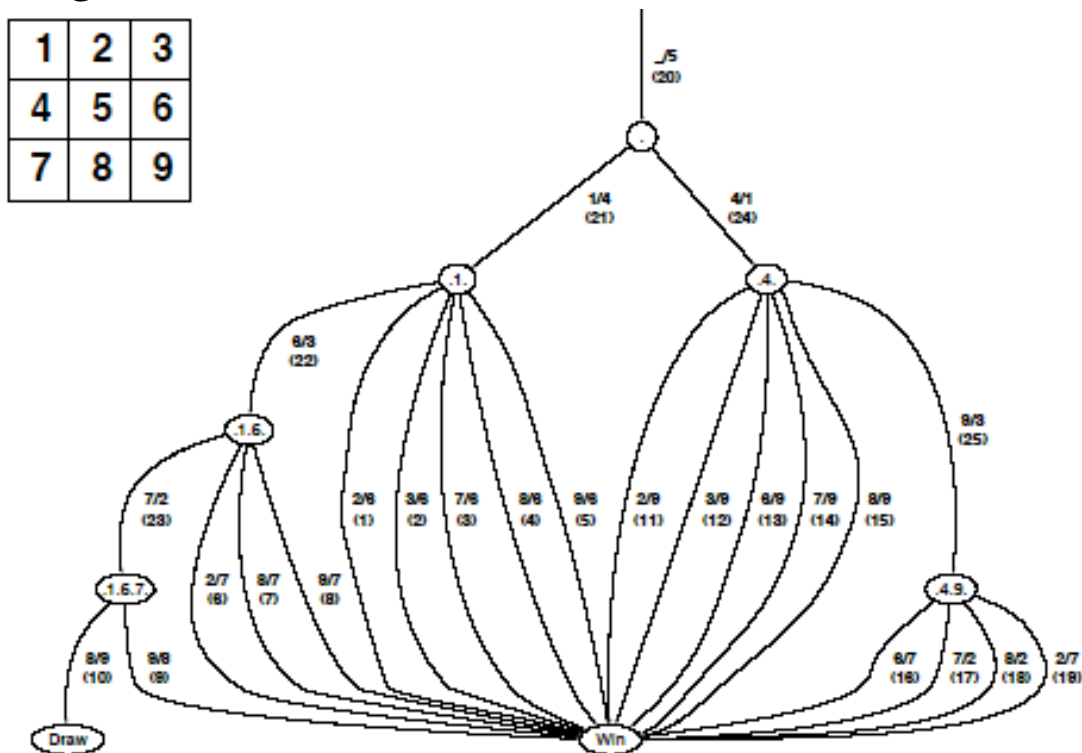


OU exclusive : mixte de ET-NON

Jouer au tic-tac-toe

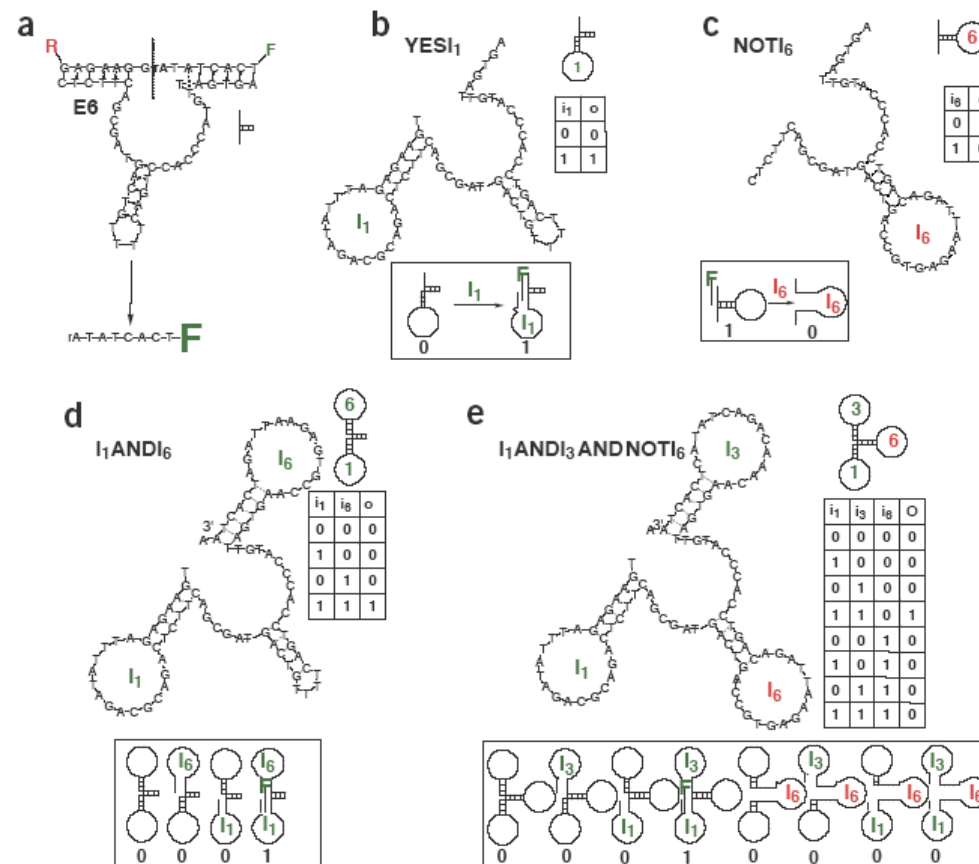
La meilleure stratégie. . .

1	2	3
4	5	6
7	8	9



Stojanovic & Stefanovic. *Nat Biotechnol* 21 :1069 (2003)

Tic-tac-toe : portes



Stojanovic & Stefanovic. *Nat Biotechnol* 21 :1069 (2003)

Tic-tac-toe : expressions logiques

$$o_1 \leftarrow \underbrace{i_4}_{24}$$

$$o_2 \leftarrow \underbrace{(i_1 \wedge i_6 \wedge i_7)}_{23} \parallel \underbrace{(i_4 \wedge i_6 \wedge i_7)}_{17} \parallel \underbrace{(i_4 \wedge i_6 \wedge i_8)}_{18}$$

$$o_3 \leftarrow \underbrace{(i_1 \wedge i_6)}_{22} \parallel \underbrace{(i_4 \wedge i_6)}_{25}$$

$$o_4 \leftarrow \underbrace{i_1}_{21}$$

$$o_5 \leftarrow \underbrace{1}_{20}$$

$$o_6 \leftarrow \underbrace{(i_1 \wedge i_2)}_1 \parallel \underbrace{(i_1 \wedge i_3)}_2 \parallel \underbrace{(i_1 \wedge i_7)}_3 \parallel \underbrace{(i_1 \wedge i_8)}_4 \parallel \underbrace{(i_1 \wedge i_9)}_5$$

$$o_7 \leftarrow \underbrace{(i_1 \wedge i_6 \wedge i_2)}_6 \parallel \underbrace{(i_1 \wedge i_6 \wedge i_8)}_7 \parallel \underbrace{(i_1 \wedge i_6 \wedge i_9)}_8 \parallel \underbrace{(i_4 \wedge i_6 \wedge i_6)}_{10} \parallel \underbrace{(i_4 \wedge i_6 \wedge i_2)}_{19}$$

$$o_8 \leftarrow \underbrace{i_1 \wedge i_6 \wedge i_7 \wedge i_9}_9$$

$$o_9 \leftarrow \underbrace{(i_1 \wedge i_6 \wedge i_7 \wedge i_8)}_{10} \parallel \underbrace{(i_4 \wedge i_2)}_{11} \parallel \underbrace{(i_4 \wedge i_3)}_{12} \parallel \underbrace{(i_4 \wedge i_6)}_{13} \parallel \underbrace{(i_4 \wedge i_7)}_{14} \parallel \underbrace{(i_4 \wedge i_8)}_{15}$$

$$o_1 = \underbrace{i_4}_{24}$$

$$o_2 = \underbrace{(i_6 \wedge i_7 \wedge \neg i_2)}_{23} \vee \underbrace{(i_7 \wedge i_6 \wedge \neg i_1)}_{17} \vee \underbrace{(i_8 \wedge i_6 \wedge \neg i_1)}_{18}$$

$$o_3 = \underbrace{(i_1 \wedge i_6)}_{22} \vee \underbrace{(i_4 \wedge i_6)}_{25}$$

$$o_4 = \underbrace{i_1}_{21}$$

$$o_5 = \underbrace{1}_{20}$$

$$o_6 = \underbrace{(i_1 \wedge i_2 \wedge \neg i_6)}_1 \vee \underbrace{(i_1 \wedge i_3 \wedge \neg i_6)}_2 \vee \underbrace{(i_1 \wedge i_7 \wedge \neg i_6)}_3 \vee \underbrace{(i_1 \wedge i_8 \wedge \neg i_6)}_4 \vee \underbrace{(i_1 \wedge i_9 \wedge \neg i_6)}_5$$

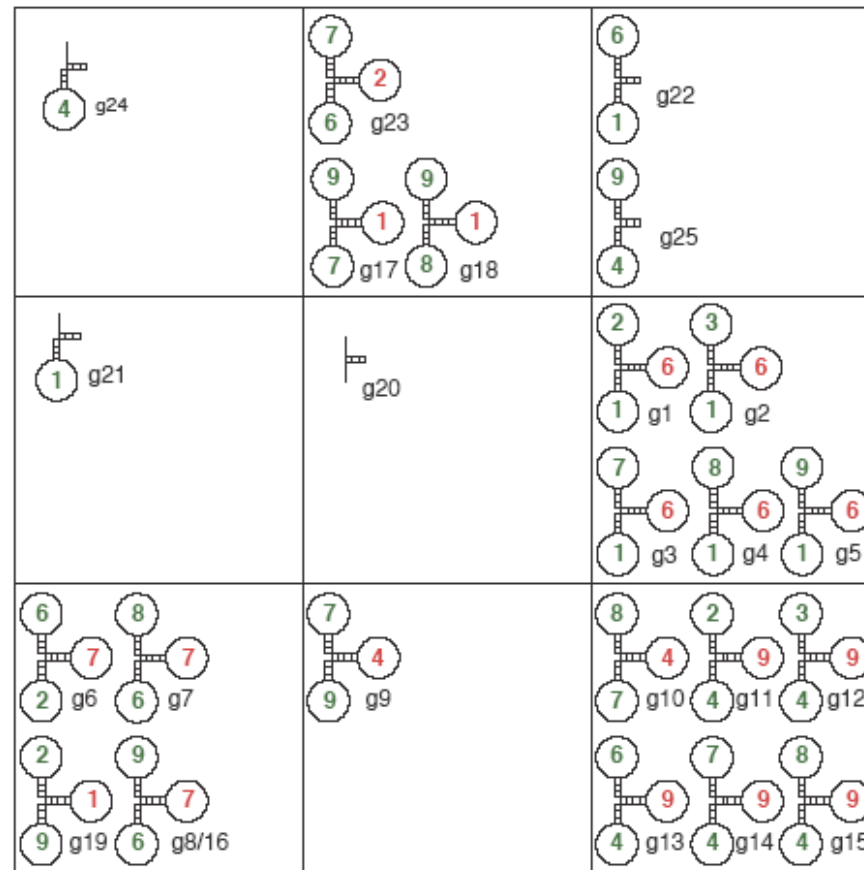
$$o_7 = \underbrace{(i_2 \wedge i_6 \wedge \neg i_7)}_6 \vee \underbrace{(i_8 \wedge i_6 \wedge \neg i_7)}_7 \vee \underbrace{(i_8 \wedge i_6 \wedge \neg i_7)}_{8 \text{ and } 10} \vee \underbrace{(i_9 \wedge i_2 \wedge \neg i_1)}_{19}$$

$$o_8 = \underbrace{i_1 \wedge i_7 \wedge \neg i_4}_9$$

$$o_9 = \underbrace{(i_7 \wedge i_8 \wedge \neg i_4)}_{10} \vee \underbrace{(i_4 \wedge i_2 \wedge \neg i_6)}_{11} \vee \underbrace{(i_4 \wedge i_3 \wedge \neg i_6)}_{12} \vee \underbrace{(i_4 \wedge i_6 \wedge \neg i_9)}_{13} \vee \underbrace{(i_4 \wedge i_7 \wedge \neg i_9)}_{14} \vee \underbrace{(i_4 \wedge i_8 \wedge \neg i_9)}_{15}$$

il faut transformer l'expression initiale [gauche] pour l'implantation

Voilà La machine !



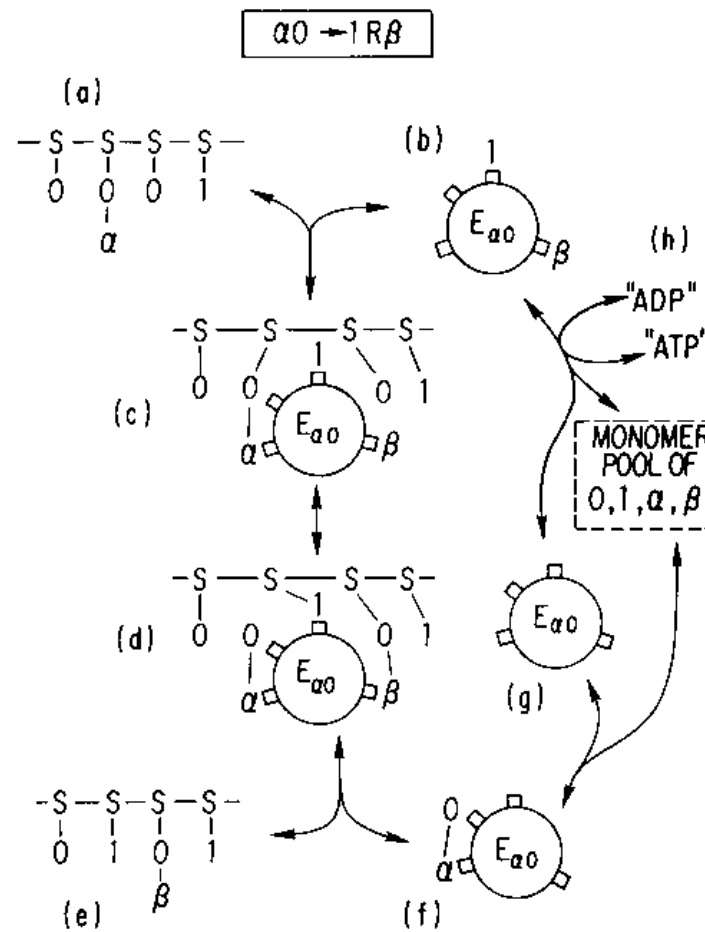
Stojanovic & Stefanovic. *Nat Biotechnol* 21 :1069 (2004)

Ordinateur moléculaire

logiciel, entrée, sortie, hardware

on veut implanter une machine de Turing universelle à l'aide d'ADN, enzymes, etc.

Machine de Turing moléculaire



Bennett *Int J Theor Phys* 21 :905 (1982)

Calcul par automate fini

Automate fini à deux états : $S0$ et $S1$, alphabet : $\{a, b\}$

Symbol	a	b	terminator (t)
encodings & <state, symbol> sticky ends	$\langle S1, a \rangle$  $\langle S0, a \rangle$	$\langle S1, b \rangle$  $\langle S0, b \rangle$	$\langle S1, t \rangle$  $\langle S0, t \rangle$

Benenson & al. *PNAS* 100 :2191 (2003)

AF : transitions



T1: $SO \xrightarrow{a} SO$



T5: $S1 \xrightarrow{a} SO$



T2: $SO \xrightarrow{a} S1$



T6: $S1 \xrightarrow{a} S1$



T3: $SO \xrightarrow{b} SO$



T7: $S1 \xrightarrow{b} SO$

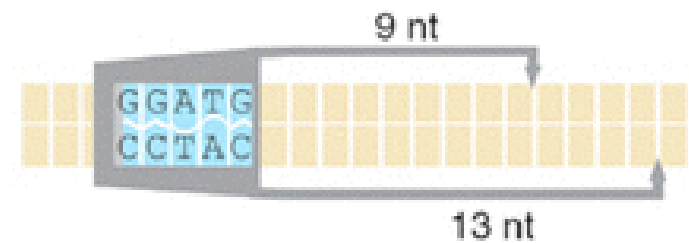
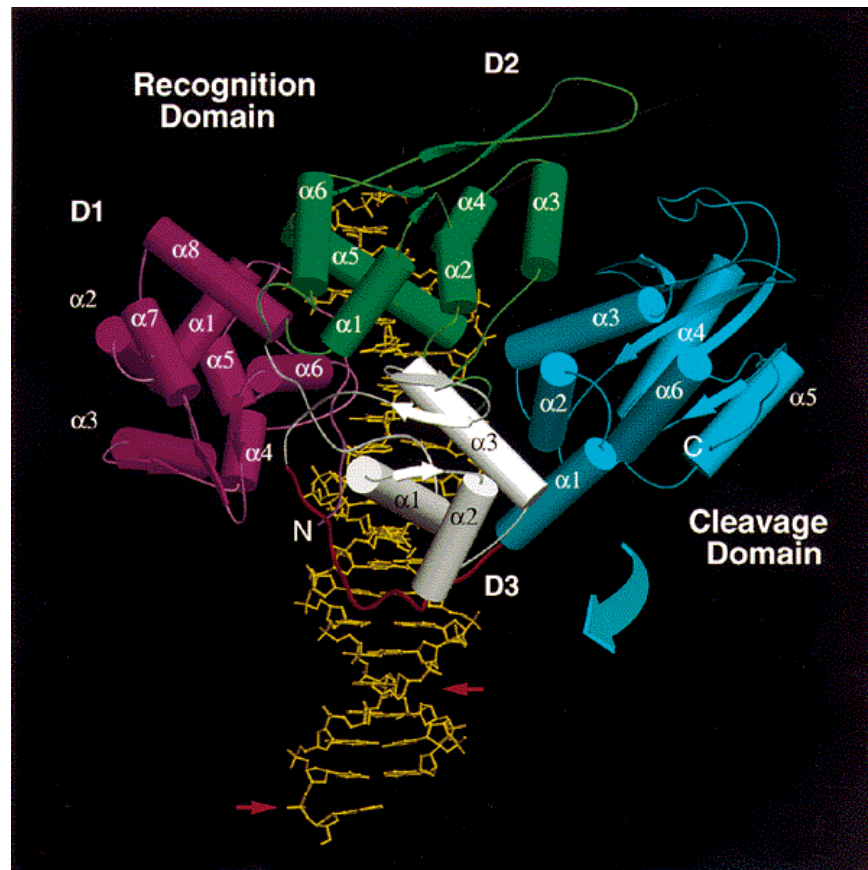


T4: $SO \xrightarrow{b} S1$

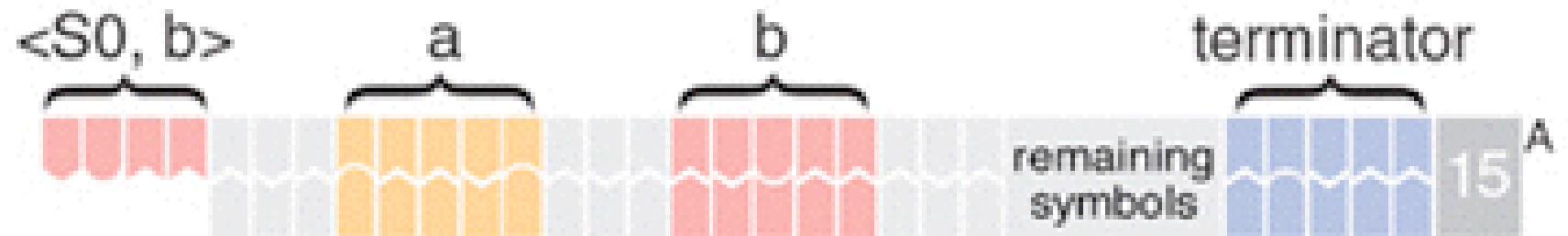


T8: $S1 \xrightarrow{b} S1$

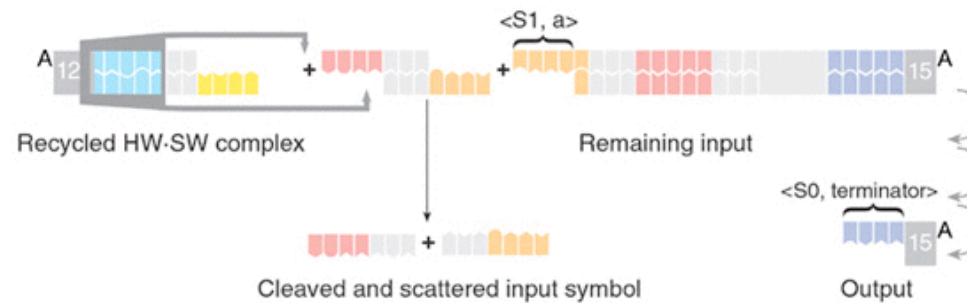
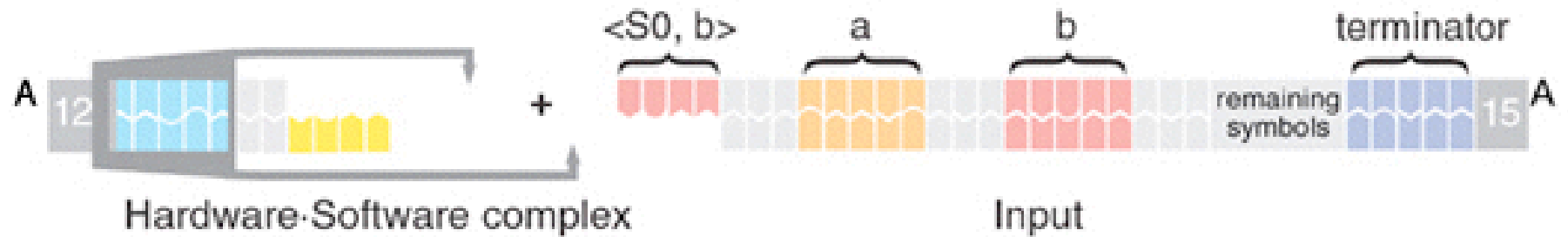
AF : hardware — *FokI*



AF : configurations



AF : calcul

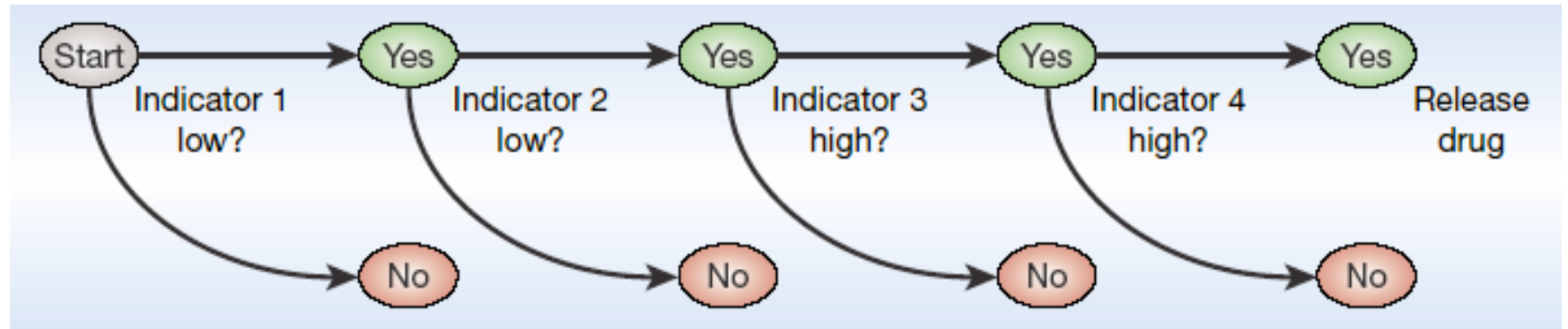


Vitesse et énergie

vitesse du calcul : 20–40s par transition (mais $6.6 \cdot 10^{10} s^{-1}$ par μl ... 66 GHz)

énergie : $34kT$ où $k = 1.38 \cdot 10^{-23} JK^{-1}$ (min : $0.5kT$, neurones : $10^{11}kT$)

«Médicament intelligent»

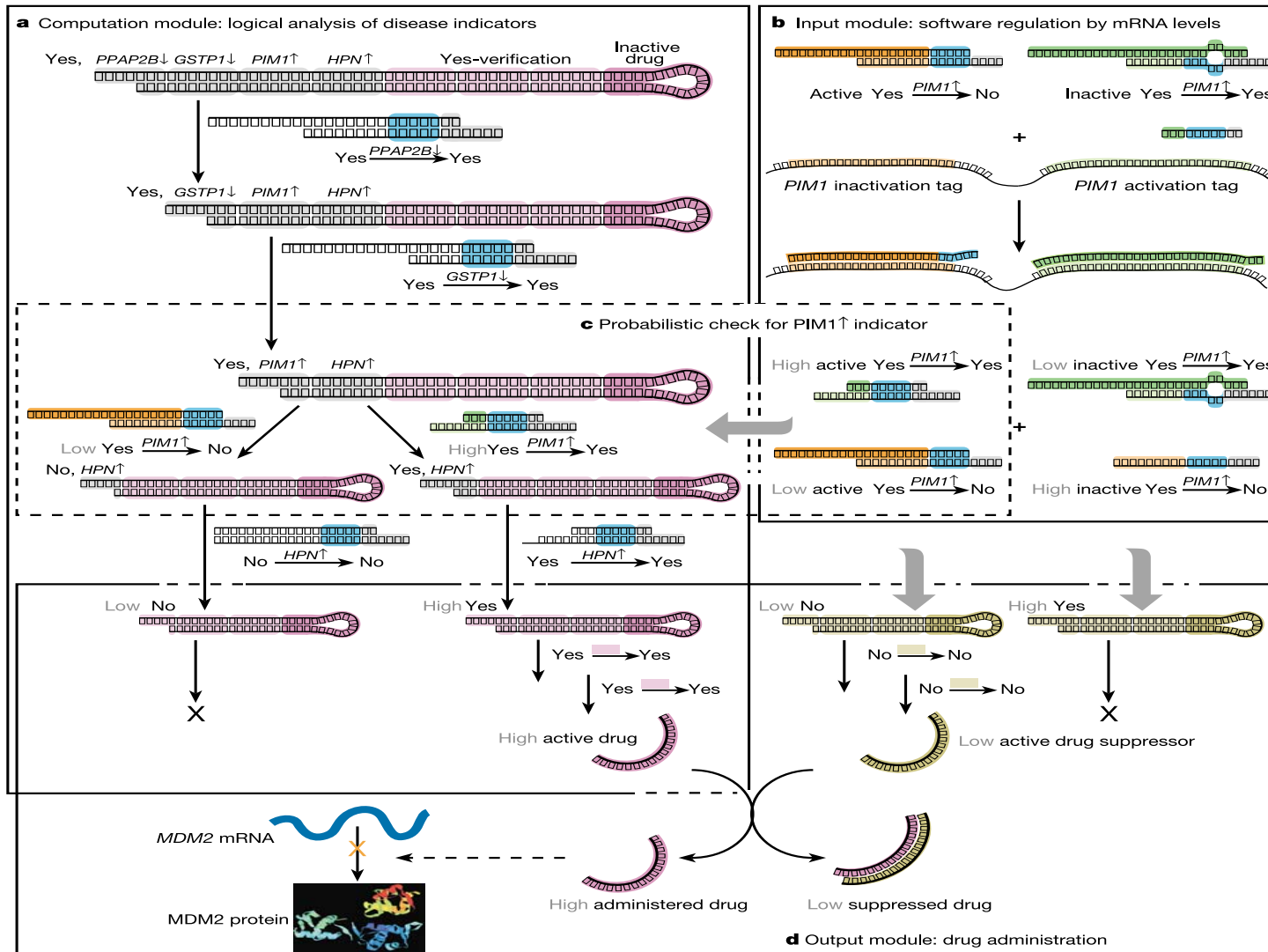


configurations encodées de même façon qu'avant ; transitions exécutées par FokI

challenge : lier le niveau d'expression d'un gène au symbole qui l'encode

Condon. *Nature* 429 :351 (2004)

10¹² ordinateurs dans une goutte d'eau



Benenson et al. *Nature* 429 :423 (2004) ; Ehud Shapiro