

# *PB051: Výpočetní metody v bioinformatice a systémové biologii*

David Šafránek

6.4.2010

Tento projekt je spolufinancován Evropským sociálním fondem a státním rozpočtem České republiky.



# *Obsah*

*Systémové paradigma – síť interakcí*

*Identifikace genových interakcí*

*Analýza genové exprese*

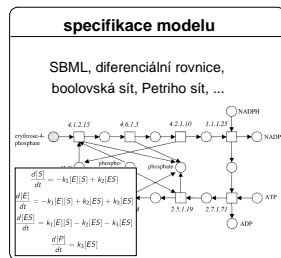
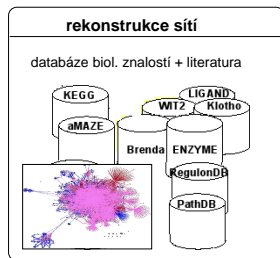
# Obsah

*Systémové paradigma – síť interakcí*

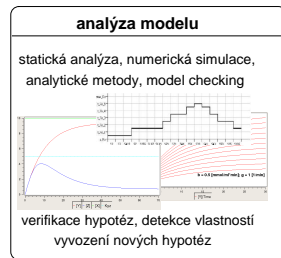
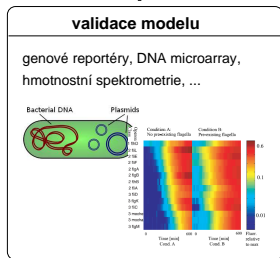
Identifikace genových interakcí

Analýza genové exprese

# Průběh výzkumu v systémové biologii

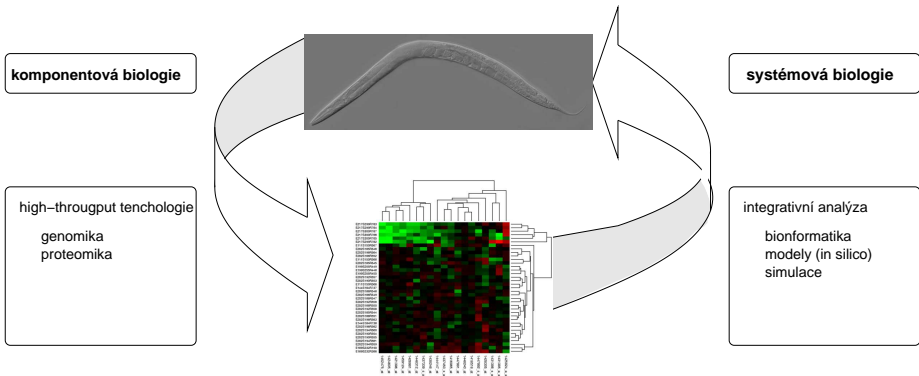


biologická síť  
 →  
 hypotézy  
 →

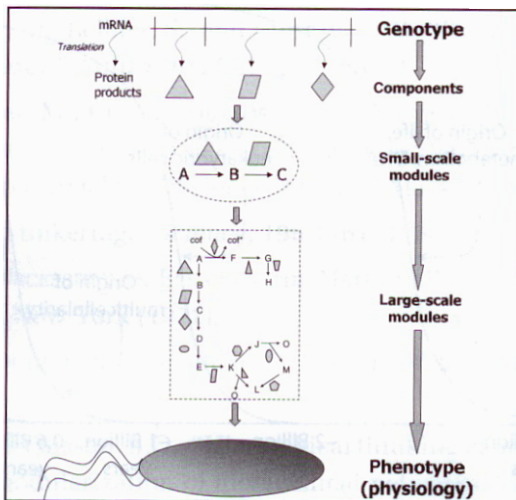


objevené vlastnosti  
 ←  
 dotazy na model  
 →

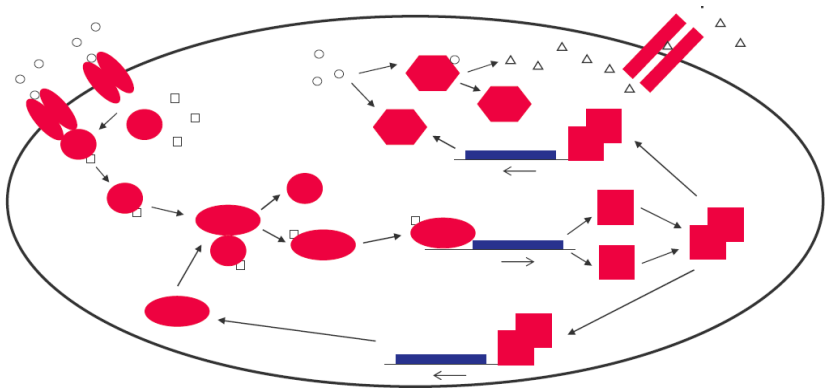
# *Od redukcionismu k integrativnímu přístupu*



# Koncept hierarchie

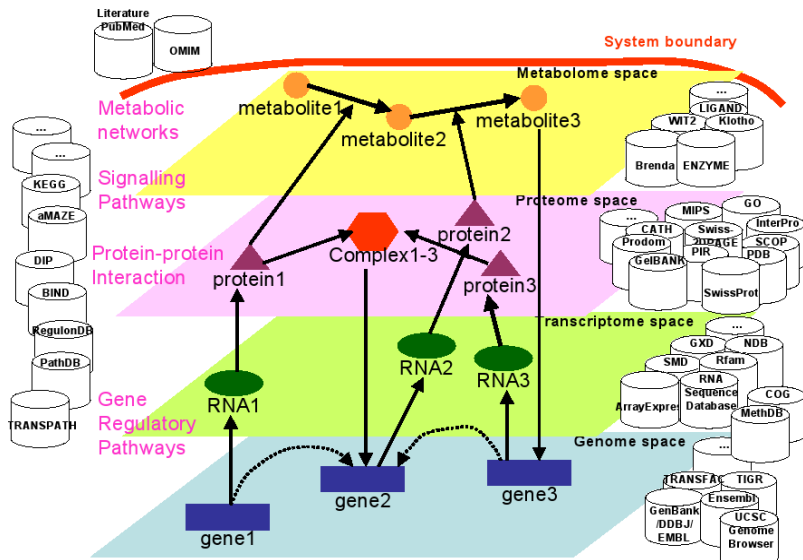


## Biochemické procesy v buňce



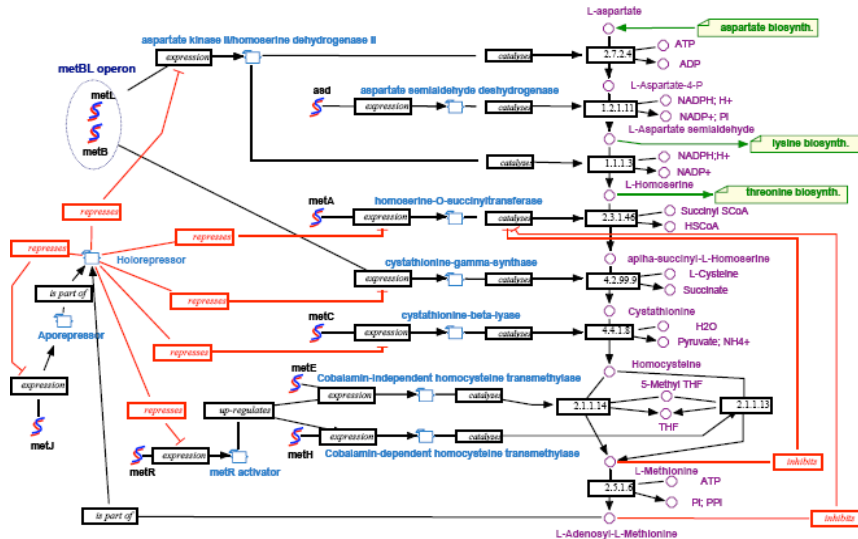
- molekulární komponenty – proteiny, DNA, RNA,...
- interakce na různých úrovních (transkripce, metabolismus,...)
- příjem signálů a živin (nutrientů) na membráně

# Funkční vsrtyvy buňky

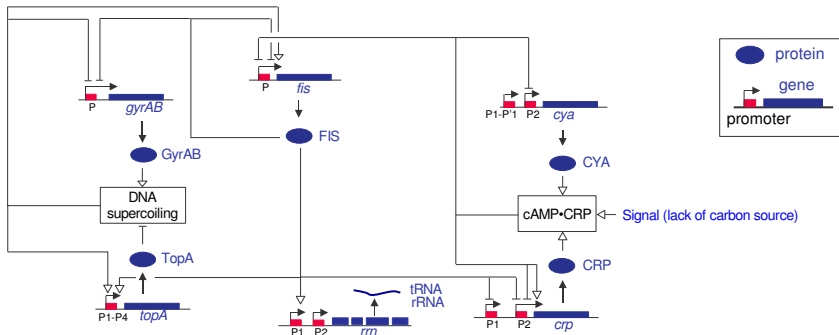




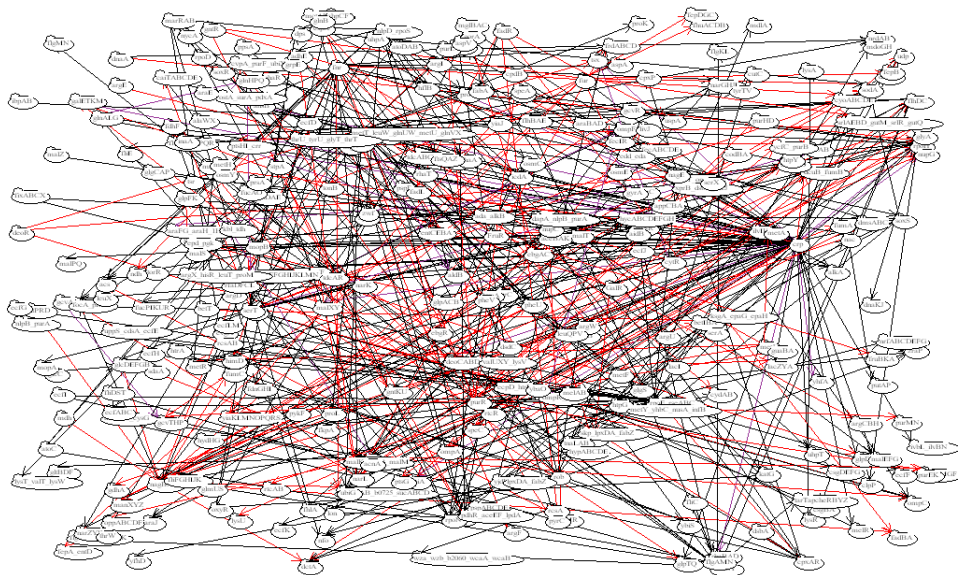
# Příklad geneticky řízené metabolické dráhy



# Příklad modulu genetické regulace v *E. Coli*



# *Kompletní transkripční síť E. Coli*



## Biologická síť jako obecný graf

### Definition

Nechť  $V$  je konečná množina uzlů a  $E \subseteq V \times V$  relace.

*Biologickou sítí* nazveme graf  $G$  reprezentovaný uspořádanou dvojicí  $G \equiv (V, E)$ .

- Pokud  $\forall \langle a, b \rangle \in E. \langle a, b \rangle \in E \rightarrow \langle b, a \rangle \in E$ ,  $G$  nazýváme *neorientovaný*.
- V ostatních případech hovoříme o *orientovaném* grafu.

typ sítě	$V$	$E$	$G$
genové	geny (resp. proteiny)	regulace exprese	or.
proteinové	proteiny	proteinové interakce	neor.
metabolické	metabolity, enzymy	enzymové reakce	or.
signální	molekuly	aktivace/deaktivace	or.

## *Biologické sítě – výpočetní problémy*

- (re)konstrukce sítí
  - identifikace interakcí
    - ⇒ z experimentálních dat
    - ⇒ integrací dat z databází znalostí
- zpracování sítí
  - vizualizace
  - integrace atributů uzlů, hran
- analýza sítí
  - analýza statistických vlastností
    - ⇒ rozložení konektivity, detekce motivů, ...
  - porovnání sítí

## *Biologické sítě – nástroje*

- Cytoscape

<http://www.cytoscape.org>

- vizualizační layouty
- mapper vizuálních prvků na data
- filtry
- pluginy pro práci s biologickými sítěmi

- VisANT

<http://visant.bu.edu/>

- prakticky tatáž funkčnost jako Cytoscape
- podpora hierarchického zanořování (MetaNodes)
- méně flexibilní prostředí
- více biologicky-relevantních funkcí



# *Obsah*

Systémové paradigma – síť interakcí

*Identifikace genových interakcí*

Analýza genové exprese



## *Identifikace struktury promotorů*

- podobnost vůči konzervované sekvenci “typického” promotoru:  
⇒ promotory většinou zahrnují sekvenci podobnou s:

5'-TATAA-3' (tzv. TATA-box)

- konzervovaná sekvence typicky zapsána ve formátu IUPAC –  
vyjadřuje nejednoznačnosti v sekvenci

T	C	C	C	T
T	C	C	G	A
T	T	G	C	T
T	T	C	C	T
T	T	C	G	T
<hr/>				
T	Y	C	S	T

## *Identifikace struktury promotorů*

- detekce vazebných míst transkripčních faktorů (TFBS)
  - typická délka 5–25 bází
  - poměrně velká tolerance k variabilitě sekvencí
- u prokaryot TFBSs poblíž TSS (transcription starting site)  
⇒ krátké sekvence na pozicích  $-10$  a  $-35$
- u eukaryot je struktura promotoru složitější  
⇒ TFBS mohou být vzdálena i několik kilobází od TSS (ve směru  $3' - 5'$ )
- promotory se strukturně identifikují ve sledu:  
TFBSs → kombinované TFBSs → celý promotor

## *Identifikace struktury promotorů*

- alternativní variantou k popisu vazby TF k DNA je poziční matice (Positional Weight Matrix – PWM)

<i>T</i>	<i>C</i>	<i>C</i>	<i>C</i>	<i>T</i>	
<i>T</i>	<i>C</i>	<i>C</i>	<i>G</i>	<i>A</i>	
<i>T</i>	<i>T</i>	<i>G</i>	<i>C</i>	<i>T</i>	
<i>T</i>	<i>T</i>	<i>C</i>	<i>C</i>	<i>T</i>	
<i>T</i>	<i>T</i>	<i>C</i>	<i>G</i>	<i>T</i>	
0	0	0	0	1	<i>A</i>
0	2	4	3	0	<i>C</i>
0	0	1	2	0	<i>G</i>
5	3	0	0	4	<i>T</i>

- PWM zachycuje frekvenci výskytu nukleotidů na příslušné pozici TFBS afinní pro daný TF

## *Nástroje pro vyhledávání promotorových sekvencí*

- TRANSFAC

<http://www.gene-regulation.com/cgi-bin/pub/databases/transfac/>  
obsahuje data transkripčních faktorů, regulačních sekvencí, včetně PWM

- nástroje pro vyhledávání v TRANSFAC

- MATCH – vyhledává TFBSs v sekvencích, dostupný z TRANSFAC

- PromoterScan

<http://zeon.well.ox.ac.uk/git-bin/proscan>  
vyhledávání globálních promotorových sekvencí v dodané sekvenci

- TRANSCompel

<http://www.gene-regulation.com/pub/databases/transcompel/compelSM.html>  
vyhledávání kombinovaných regulačních sekvencí

- RegulonDB promoter analysis (E. coli)

[http://www.ccg.unam.mx/Computational\\_Genomics/PromoterTools/](http://www.ccg.unam.mx/Computational_Genomics/PromoterTools/)

## *Další přístupy*

- využití DNA mikročipů
  - identifikace genů s podobnými profily exprese a jejich agregace do skupin (tzv. klastrů)
  - prohledávání promotorových sekvencí na přítomnost shodných TFBSs
  - úspěšně aplikováno pro detekci genových interakcí v kvasince
- analýza promotorů ortologických genů (napříč různými druhy)
  - tzv. phylogenetic footprinting
  - viz např. [http://bayesweb.wadsworth.org/binding\\_sites/index.html](http://bayesweb.wadsworth.org/binding_sites/index.html) (E. coli)

# *Obsah*

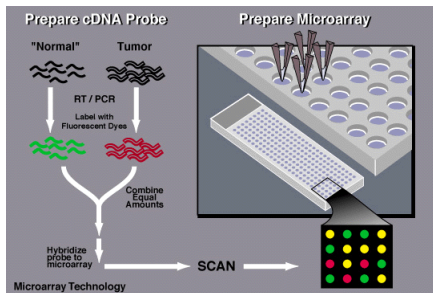
Systémové paradigma – síť interakcí

Identifikace genových interakcí

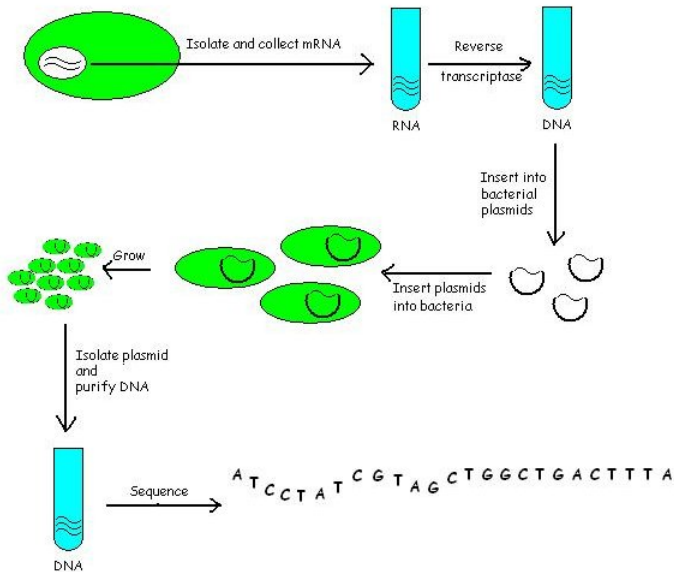
*Analýza genové exprese*

## Měření genové exprese

- nejpoužívanějším nástrojem je technologie DNA microarray
- umožňuje tzv. high-throughput analýzu
  - v daném okamžiku je paralelně nasamplována exprese všech genů v genomu příslušného organismu
  - postaveno na relativním srovnání minimálně dvou různých vzorků
    - exprese v přítomnosti vs. nepřítomnosti  $O_2$
    - exprese při knock-outu určitého genu vs. normální stav
    - ...



# Reverzní transkriptáza a cDNA





## *Reverzní transkriptáza a cDNA*

- enzym EC 2.7.7.49 (druh DNA polymerázy)
- objevena v retrovirech [Temin, Baltimore, 1970]
- přepisuje mRNA na jednořetězcovou (komplementární) DNA (tzv. cDNA)
- umožňuje vytvořit knihovnu DNA

## *Polymerase Chain Reaction (PCR)*

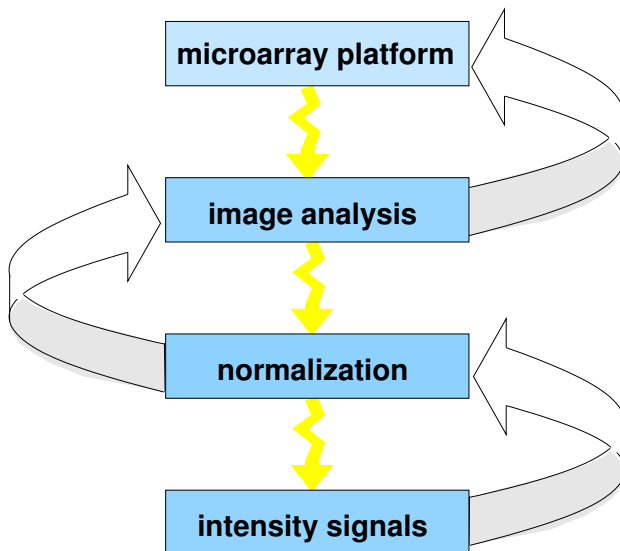
- umožňuje replikaci určité části DNA (tzv. amplifikace)
- DNA je zahřátím rozdělena
- úsek DNA je označen párem oligonukleotidů (15-25 bazí)
  - při snížení teploty hybridizace oligonukleotidů s řetězcem DNA
  - doplnění zbývající sekvence DNA prostřednictvím RNA polymerázy
- <http://www.dnalc.org/resources/animations/pcr.html>
- lze využít i pro mRNA: RT-PCR (reverse transcription PCR)
  - reverzní transkripce mRNA do cDNA
  - amplifikace cDNA (PCR)

## *Postup při DNA microarray experimentu*

1. konstrukce čipu z cDNA knihovny (amplifikace a rozmístění)
2. odběr celkové mRNA z experimentálních vzorků (typicky 2)
3. reverzní transkripce do cDNA asociované s fluorescenčním barvivem
4. hybridizace odebrané cDNA s cDNA na čipu
5. omytí čipu a oskenování výsledku
6. analýza dat
7. komerční čipy používají místo cDNA knihovny skupinu oligonukleotidů pro každý gen  
⇒ pouze jeden vzorek mRNA je analyzován na jednom čipu (porovnání více identicky připravených čipů)

<http://www.bio.davidson.edu/courses/genomics/chip/chip.html>

# *Workflow produkce výsledků DNA microarray experimentu*



## *Validace a zpracování výsledků*

- validace dat separátním měřením koncentrací mRNA nepřímo (pomocí RT-PCR)
  - RT-PCR spuštěna pro oba vzorky (shodný počet kroků PCR)
  - porovnání koncentrací příslušných cDNA
- klastrování dat
  - zjišťování podobnosti mezi datovými vektory
  - agregace do specifických skupin (klastřů)

## *Databáze microarray dat*

- Stanford Microarray Database – různé pohledy na data, filtrace  
<http://smd.stanford.edu/cgi-bin/cluster/drpGetData.pl>
- ArrayExpress – statisticky zpracovaná data  
<http://www.ebi.ac.uk/gxa/>
- Gene Expression Omnibus (GEO)  
<http://www.ncbi.nih.gov/geo/>
- MUSC DNA Microarray Database  
<http://proteogenomics.musc.edu/ma/>
- GenExpDB (E. Coli specifická data)  
<http://genexpdb.ou.edu/>

## *Klastrování microarray dat*

- předpokládejme matici se sondami pro  $n$  genů
- uvažujme sadu  $p$  experimentů
- pro každý gen  $i$  dostáváme vektor  $\mathbf{x}_i = (x_{i1}, \dots, x_{ip})$  zachycující posloupnost výsledků (tzv. *expresní profil*)
- definujeme míru vzdálenosti  $d : \mathbb{R}^p \times \mathbb{R}^p \rightarrow \mathbb{R}$ :

$$d(\mathbf{x}_n, \mathbf{x}_m) = \left( \sum_{i=1}^p |x_{ni} - x_{mi}|^q \right)^{\frac{1}{q}}$$

- pro  $q = 2$  dostáváme Euklidovskou vzdálenost

## *Klastrování microarray dat*

- existují dva hlavní přístupy ke klastrování
  - partitioning – cílem je najít jedno nejvhodnější rozdělení do klastrů (parametrem je počet požadovaných klastrů)
    - ⇒ Self-organizing maps, K-means
  - hierarchické metody – vytvořen celý strom hierarchie
    - ⇒ kořen – klastr obsahující všechny experimenty, v listech
    - ⇒ listy – jednoprvkový klastr pro každý experiment
- klastry mohou být identifikovány i pro vektory tvaru  $\mathbf{x}_j' = (x_{1j}, \dots, x_{nj})$



## *Algoritmus pro hierarchické klastrování*

- nejpoužívanější metoda je tzv. aglomerativní (zdola-nahoru)
- parametrem je míra podobnosti hodnot  $d(x_i, x_j)$
- postup ( $t$  značí aktuální úroveň):
  1.  $t = n \Rightarrow$  inicializuj pro každý gen  $i \leq n$ :  $C_i^n = \{x_i\}$
  2. spoj dva klustery  $C_k^t$  a  $C_l^t$  s minimální vzdáleností  $D(C_k^t, C_l^t)$
  3. update  $D$  dle nového rozdělení
  4.  $t := t - 1$
  5. iteruj (2-4) dokud  $t > 1$

## *Algoritmus pro hierarchické klustrování*

Při aglomeraci se používá míra podobnosti dvou klastrů na téže úrovni  $t$ :

- $D(C_k^t, C_l^t) = \min_{x_i \in C_k^t, x_j \in C_l^t} d(x_i, x_j)$  (single linkage)
- $D(C_k^t, C_l^t) = \max_{x_i \in C_k^t, x_j \in C_l^t} d(x_i, x_j)$  (complete linkage)
- $D(C_k^t, C_l^t) = \frac{1}{|C_k^t| |C_l^t|} \sum_{x_i \in C_k^t, x_j \in C_l^t} d(x_i, x_j)$  (average linkage)

## Algoritmus pro hierarchické klastrování

- update míry vzdálenosti (krok (3)):

$$D(C_m^{t-1}, C_k^t \cup C_l^t) = \alpha_k D(C_m^t, C_k^t) + \alpha_l D(C_m^t, C_l^t) + \gamma |D(C_m^t, C_l^t) - D(C_m^t, C_k^t)|$$

- single linkage:  $\alpha_k = \alpha_l = 0.5$ ,  $\gamma = -0.5$
- complete linkage:  $\alpha_k = \alpha_l = 0.5$ ,  $\gamma = 0.5$
- average linkage:  $\alpha_i = \frac{|C_i^t|}{|C_k^t| + |C_l^t|}$ ,  $i \in \{k, l\}$ ,  $\gamma = 0$

## *Metoda K-means*

- založeno na optimalizaci odchylky mezi expresními profily vzhledem ke středu (průměrnému profilu) klastru
- nejčastěji je tato optimalizace reprezentována minimalizací
- pevně dán počet požadovaných klastrů
- náhodně se inicializují střední profily
  - metoda je přesnější při větším počtu pokusů
- klastry jsou průběžně modifikovány při minimalizaci odchylek (Euklidovské vzdálenosti) od středových profilů

## *Metoda K-means*

- algoritmus K-means má dvě základní fáze
  - výpočet vzdáleností jednotlivých vektorů od vektoru středových hodnot
  - update vzhledem k optimalizační funkci
- nejpoužívanější metrikou je Euklidovská vzdálenost
- vektor středových hodnot je vypočítán jako aritmetický průměr vektorů aktuálně přiřazených danému klasteru

## Algoritmus *K*-means

- vstup: počet iterací (inicializací), počet klastrů  $K$ , práh přesnosti  $\epsilon$
- náhodně inicializuj rozdělení do klastrů  $C_1^1, \dots, C_K^1$  se středy  $c_1^1, \dots, c_K^1$  a vypočítej hodnotu optimalizační funkce  $W^1$
- v  $i$ -tém kroku proved':
  - výpočet  $C_1^{i+1}, \dots, C_K^{i+1}$  – přiřaď každý datový vektor  $x$  ke klastru s nejmenší vzdáleností středového vektoru od  $x$
  - přepočítej středové vektory  $c_1^{i+1}, \dots, c_K^{i+1}$  a minimalizuj  $W^{i+1}$
- dokud  $\exists k, |c_k^i - c_k^{i+1}| \geq \epsilon$ , iteruj

## *Nástroje pro cluster-based analýzu*

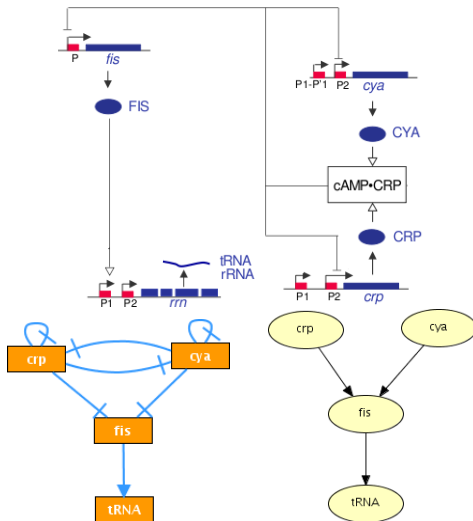
- klastrování lze využít pro detekci skupin shodně regulovaných genů
- kombinace klastrování dle genů a experimentů
  - odhady regulátorů jednotlivých klastrů
  - odhady programů regulace
- nástroje
  - Genomica (dříve GeneXPress)  
<http://genomica.weizmann.ac.il/>
  - FunCluster (balík pro R)  
<http://cran.r-project.org/web/packages/FunCluster/index.html>
  - STEM  
<http://www.cs.cmu.edu/~jernst/stem/>
  - EisenLab tools  
<http://rana.lbl.gov/EisenSoftware.htm>

## *Předpověď (reinženýring) regulačních sítí*

- regulační síť lze předpovídat z microarray dat
- předpověď struktury sítě
  - detekce podmíněných závislostí proměnných
  - charakter korelace proměnných
- předpověď dynamiky proměnných
  - fitování naměřených dat na (spojitý) model
  - pravděpodobnostní rozložení diskrétních hodnot

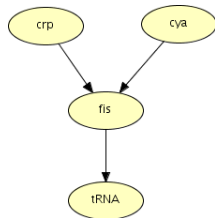
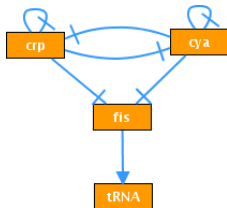


# Předpověď (reinženring) regulačních sítí



# Předpověď (reinženýring) regulačních sítí

Boolovské vs. Bayesovské sítě



$$crp(t+1) = \neg crp(t) \wedge \neg cya(t)$$

$$cya(t+1) = \neg cya(t) \wedge \neg crp(t)$$

$$fis(t+1) = \neg crp(t) \wedge \neg cya(t)$$

$$tRNA(t+1) = fis(t)$$

$$P(X_{crp})$$

$$P(X_{cya})$$

$$P(X_{fis} | X_{crp}, X_{cya})$$

$$P(X_{tRNA} | X_{fis})$$

# *Předpověď (reinženýring) regulačních sítí*

## *Bayesovské sítě*

$$P(V|W) = \frac{P(V, W)}{P(W)}$$

$$P(W|V) = \frac{P(W, V)}{P(V)}$$

$$P(V, W) = P(W, V) = P(V|W) \cdot P(W) = P(W|V) \cdot P(V)$$

Bayesův vzorec:

$$P(V|W) = \frac{P(W|V) \cdot P(V)}{P(W)}$$

Obecně pro pravděpodobnost současných jevů platí řetězové pravidlo:

$$\begin{aligned} P(V, W, Y) &= P(V|W, Y) \cdot P(W, Y) \\ &= P(V|W, Y) \cdot P(W|Y) \cdot P(Y) \end{aligned}$$

# *Předpověď (reinženýring) regulačních sítí*

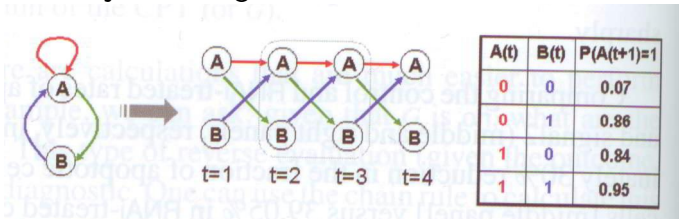
## *Algoritmy pro bayesovské sítě*

- strojové učení z experimentálních dat:
  - algoritmy učení struktury
  - algoritmy učení pravděpodobnostního rozložení  
např. Expectation Maximization (EM) – iterativní metoda maximalizující  $P(\text{data}|\text{model})$
  - kombinované algoritmy
- pro úspěšný výsledek vyžadována rozsáhlá sada dat
- nástroje:
  - Hugin (<http://www.hugin.com/>)
  - Genomica (<http://genomica.weizmann.ac.il/>)

# Předpověď (reinženýring) regulačních sítí

## Algoritmy pro bayesovské sítě

- problémem jsou zpětné vazby (cykly v síti)
- řešením je unfolding v diskrétním čase:



- původní síť s  $n$  uzly je nahrazena sítí s  $2n$  uzly
- tabulka podmíněných pravděpodobností charakterizuje pravděpodobnost přechodů mezi jednotlivými konfiguracemi