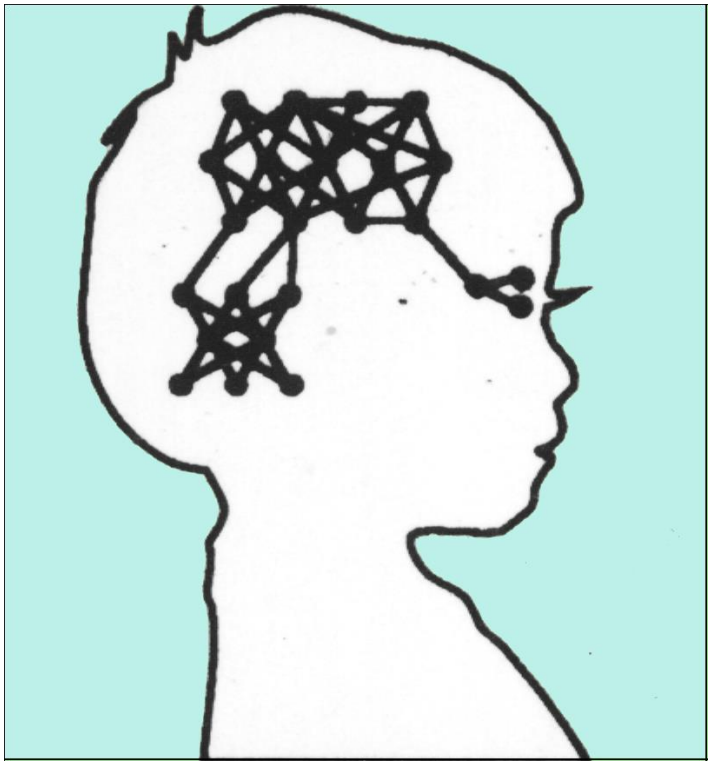
Metody umělé inteligence ve vodním hospodářství



Miloš Starý

VUT FAST Brno, ÚVHK

Umělá inteligence

- **Umělá inteligence (vědní disciplína) – aproximace určitých schopností živých organismů (intelektuálních schopností člověka)**

- **Charakterický rys umělé inteligence – těsné spojení s výpočetní technikou**

- **Výpočetní technika:**

  - umožnila řešit numericky náročné úlohy

  - vedla k hledání kvalitativně nových a efektivnějších postupů často se snahou napodobit intelektuální schopnosti člověka (analýza různých jevů, logické usuzování, rozhodování v podmínkách neurčitosti)

Motivace rozvoje metod UI

- složitost objektivní reality
- neurčitost, která má různý charakter a původ (nahodilost, vágní popis veličin)

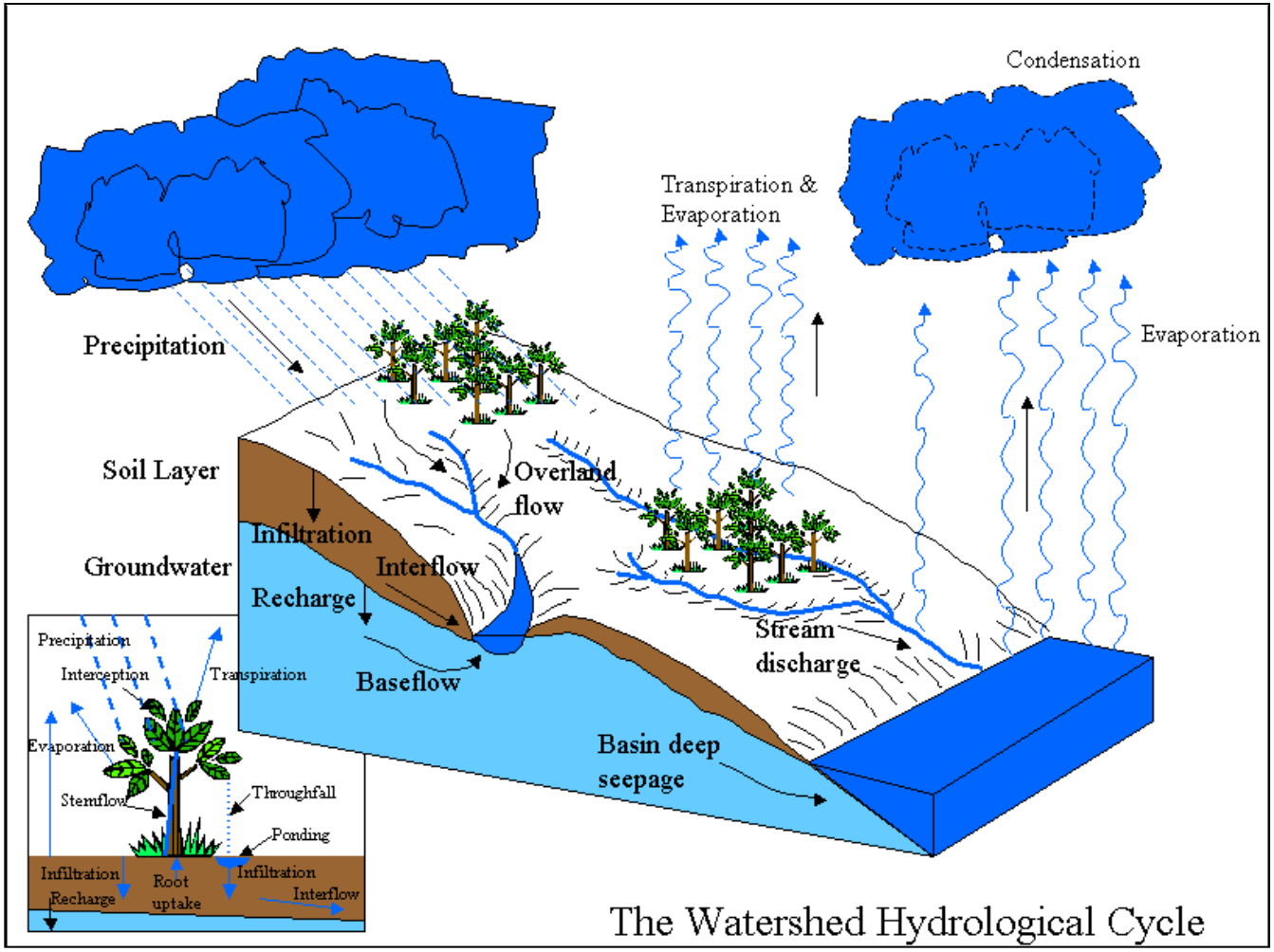
Fenomén neurčitosti

- výrazně ovlivňuje vývoj vědeckého výzkumu, který usiluje o řešení problémů měnícího se reálného světa
- někdy UI definována jako schopnost dosáhnout cíle nebo udržet požadované chování systému v neurčitých podmínkách

Důvěryhodnost matematických modelů – úzce spjata se složitostí a neurčitostí systémů (z důvodu krize dat nebo příliš dlouhé doby výpočtu, např. při operativním řízení různých systémů se používají jednodušší modely zvládnutelné numericky v přijatelném čase, které mají nižší důvěryhodnost

Srážkoodtokový proces

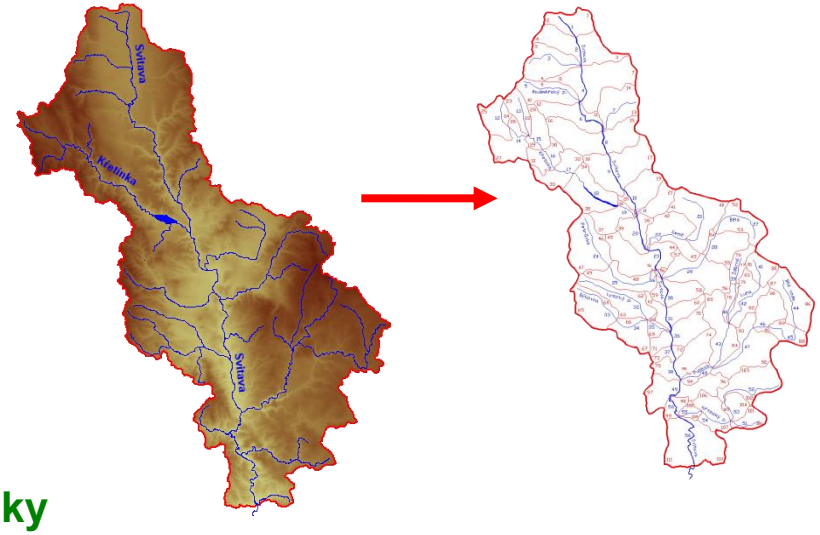
Neurčitost - modelování srážkoodtokového procesu, operativní předpovídání a operativní řízení odtoku vody z povodí za povodňových situací



Projevy neurčitosti

- Schematizace povodí
- Automatizace měření a přenosu dat /průtoky, hladiny v nádržích, srážky/
- Předpovědi srážek /ALADIN, německý model, REEDING/
- Volba modelu, kalibrace modelu
- atd. ……

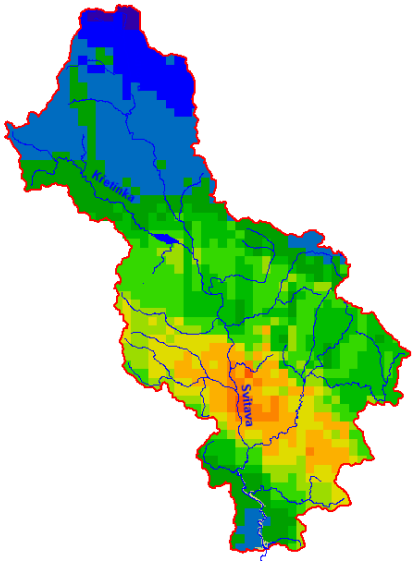
Schematizace povodí



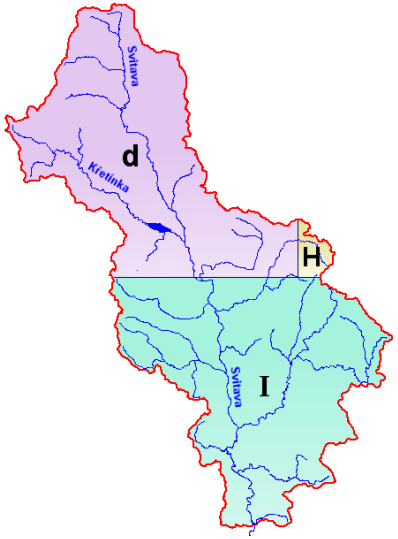
Měřené a předpovězené srážky

Rozložení srážek - Horton

Meteorologický radar



Předpověď srážek - ALADIN



Dělení metod UI

Metody umělé inteligence:

- adaptivita
- učení (umělé neuronové sítě)
- genetické algoritmy
- fuzzy modely
- teorie rozpoznávání
- expertní systémy a znalostní inženýrství
- robotika
- komunikace se strojem v přirozeném jazyce

Adaptivita

Adaptivita

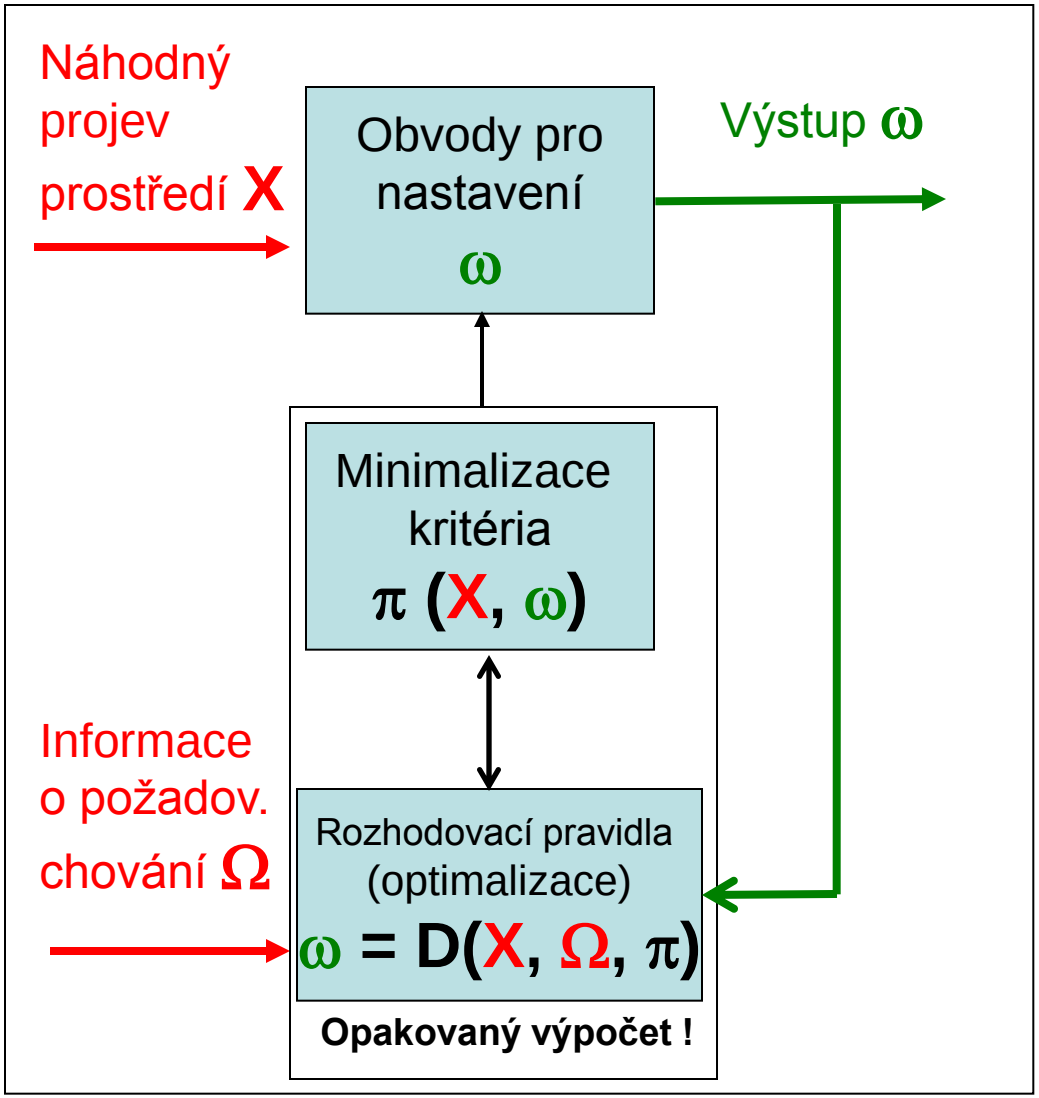
Schopnost živých organismů přizpůsobit se změnám okolního prostředí, a to hlavně když jsou nepříznivé.
Samotný proces přizpůsobování se nazývá adaptivním procesem.

Obecná teorie systémů a kybernetika (Wiener, Neumann)**:**
dosáhnout cíle nebo udržet požadované chování systému v neurčitých podmínkách

Využití ve vodním hospodářství
• konstrukce modelů pro operativní řízení odtoku vody z povodí (zásobní funkce, ochranná funkce)
⇒omezení intuitivního rozhodování při řízení systémů
⇒objektivizace a stabilizace rozhodovacího procesu v podmínkách neurčitosti

Adaptivita – účinný prostředek proti vlivu neurčitosti

Formalizace adaptivního systému



Vstup

- projev prostředí X
- požadované chování Ω

Symbolický zápis pěticí:

AS = [**X, Ω, ω, D, π**]

V praxi často intuitivní využívání

Učící se systémy

Učící se systémy

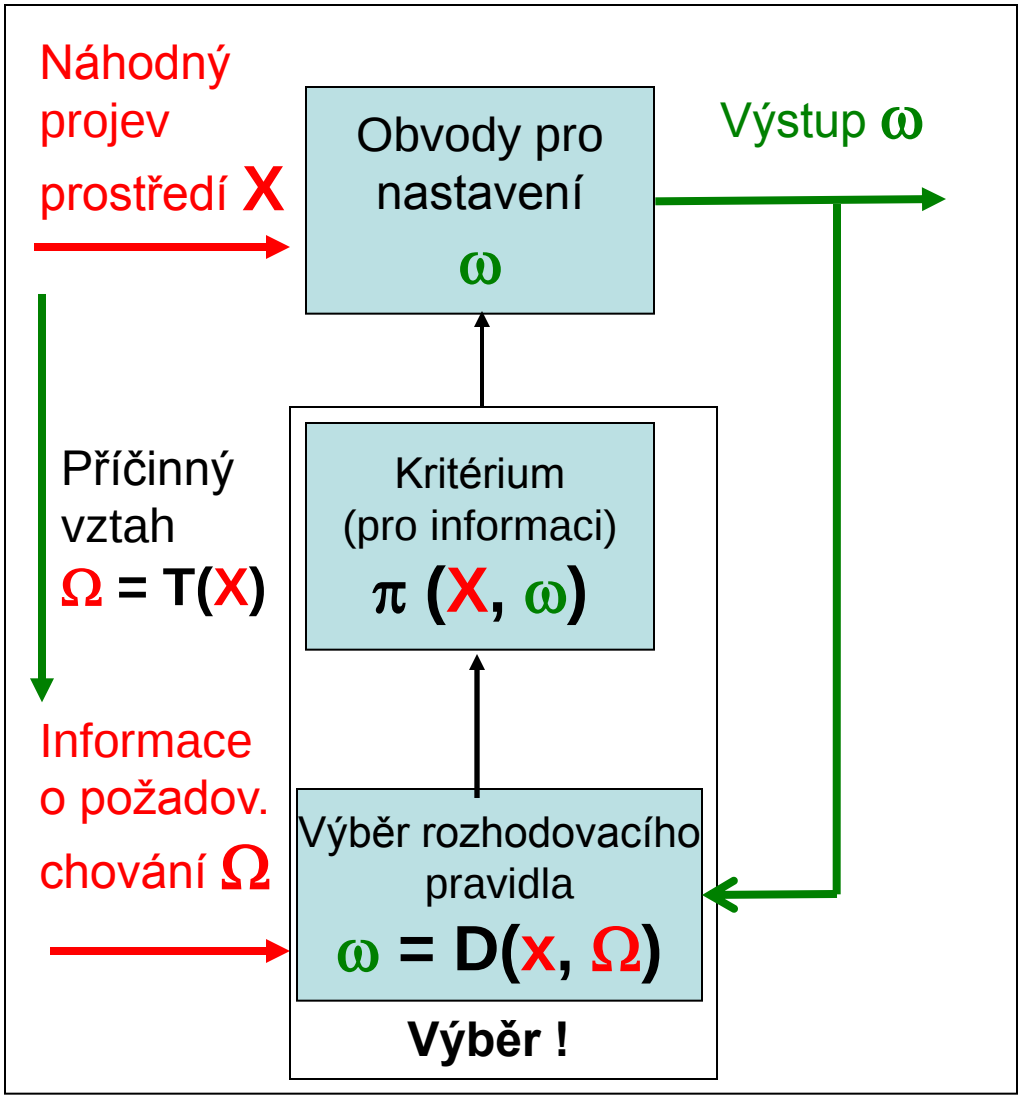
Výzkum živých organismů prokázal jejich další schopnost, a to **schopnost učení**.
Při mnohanásobném opakování adaptace na nějakou změnu prostředí dokáží organismy v průběhu dlouhého vývoje postupně zmenšovat ztráty vynaložené na adaptaci. Opakování je je zdrojem zkušeností, které si organizmus zapamatuje, zhodnocuje a učí se. Tím se snáze přizpůsobuje změněným podmínkám – učí se.

Rozdíl oproti adaptivnímu systému

- prostý adaptivní systém neustále opakuje adaptaci (i při témže projevu prostředí bez ohledu na předchozí rozhodnutí)
- učící se systém vyhledává v minulosti všechny situace uskutečněné adaptací, získané zkušenosti zhodnocuje a volí takový parametr Q_n, který vede k minimalizaci ztráty Z.

Učící se systém – speciální případ adaptivního systému !

Formalizace funkce naučeného systému



Vstup

- projev prostředí **X**
- požadované chování **Ω**

Symbolický zápis šesticí:

AS = [**X**, **Ω**, **ω**, **D**, **T**, **π**]

Vyžaduje

v předstihu zpracovat možná chování systému pomocí využití adaptivity

Genetické algoritmy

Genetické algoritmy (GA)

- Prohledávací metoda založená na principu přírodního výběru a genetiky (Holland, Goldberger)
- V každé buňce téhož jedince stejná soustava chromozonů (řetězce DNA – slouží k ukládání informací o jedinci)
- Chromozon se skládá z genů – části DNA (v každém genu zvláštní rys jedince)

Princip GA jednoduchý - spočívá v kopírování a přesouvání bloků bitů (genů).

Lze nasadit na řešení problémů, u nichž lze nějakým způsobem stanovit kvalitu jedince – fitness hodnota – kritérium π.

Prostor všech možných hodnot řešení – prohledávaný prostor.

Hledaným řešením je takové řešení (poloha bodu v prostoru), pro které fitness hodnota (kritérium) dosáhne maximální, resp. minimální hodnoty

GA

Jedinec (jeden chromozom), **populace**, **generace**

- Reprodukce – z počáteční populace jedinců (rodičů) pomocí křížení a mutace vzniká další generace jedinců. Výběr rodičů – selekce. Geny rodičů určují chromozom nového jedince (poloha bodu v prohledávaném prostoru).

Operátory GA

- ***Binární kódování chromozomů***

  Chromozom A 10110010110011011 10

  Chromozom B 11111100001101001 11

  Kódování pomocí hodnot

Chromozom A	1.54	2.47	5.18	2.36	8.48	3.14	5.67
Chromozom B	1.64	2.41	5.49	2.11	8.01	3.95	5.13

$$r = a + I(B).\frac{(b-a)}{2^n}$$

  Stromové kódování **- NS**

- **Křížení**

  Potomek 1 10110010111101001 11

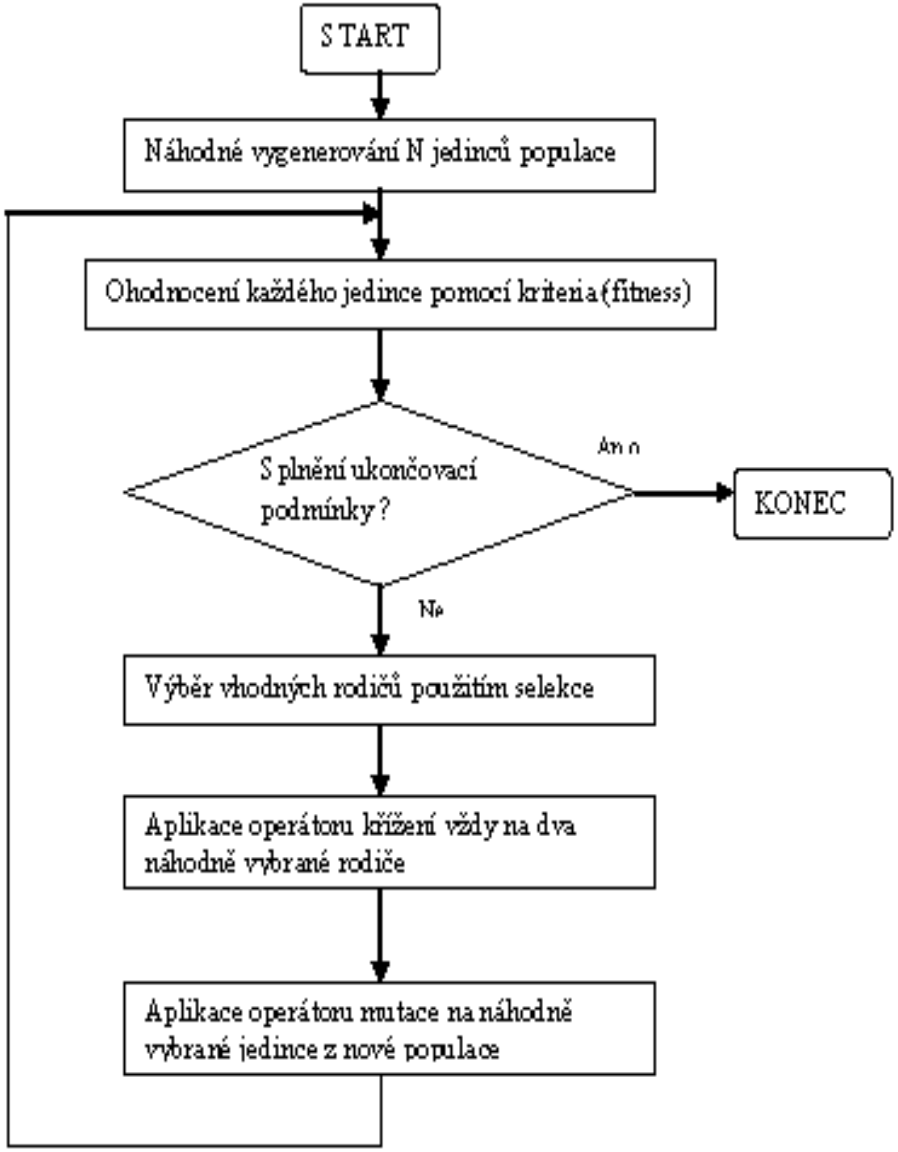
  Potomek 2 11111100000011011 10

- **Mutace** (pouze určité procento jedinců)

  Mut. pot. 1 10110011111101001 11

- **Elitismus**

GA - shrnutí



- Evoluční řešení
- Jedinec (chromozom, kritérium = fitness), populace, generace
- Reprodukce (selekce, křížení, mutace, elitismus)
- Nová populace – naděje, že v ní z hlediska kriteria kvalitnější jedinci

- Prohledávaný prostor = prostor všech možných řešení
- Každý bod prostoru – jedinec ohodnocený kritériem
- hledáním extrému daného kriteria v prostoru

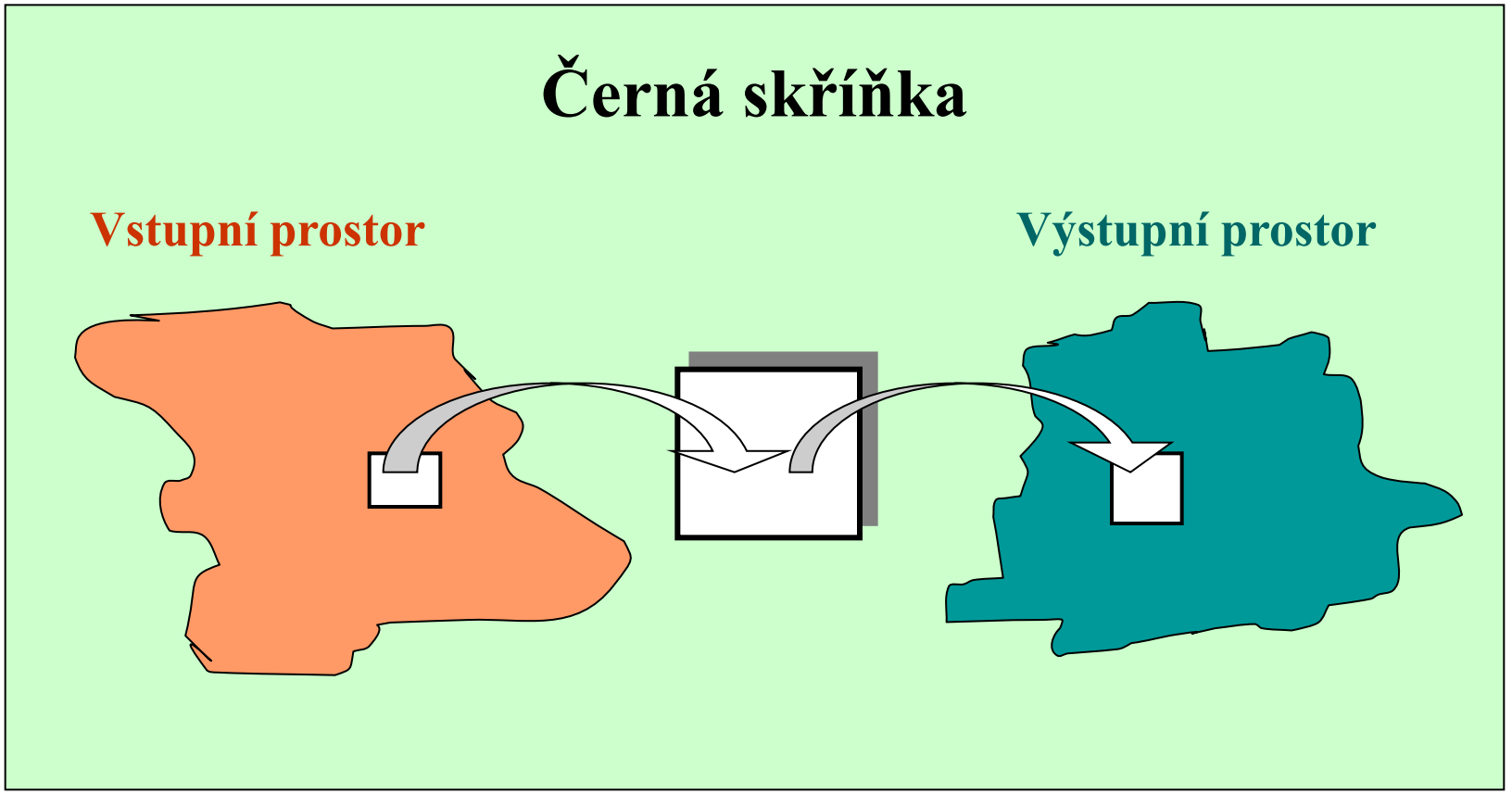
- Základní znaky – náhodnostní algoritmy, paralelismus, méně náchylné k uváznutí v pasti lokálního minima

Využití

- Kalibrace modelů
- Nelineární optimalizace v řídících algoritmech
- Trénování umělých neuronových sítí

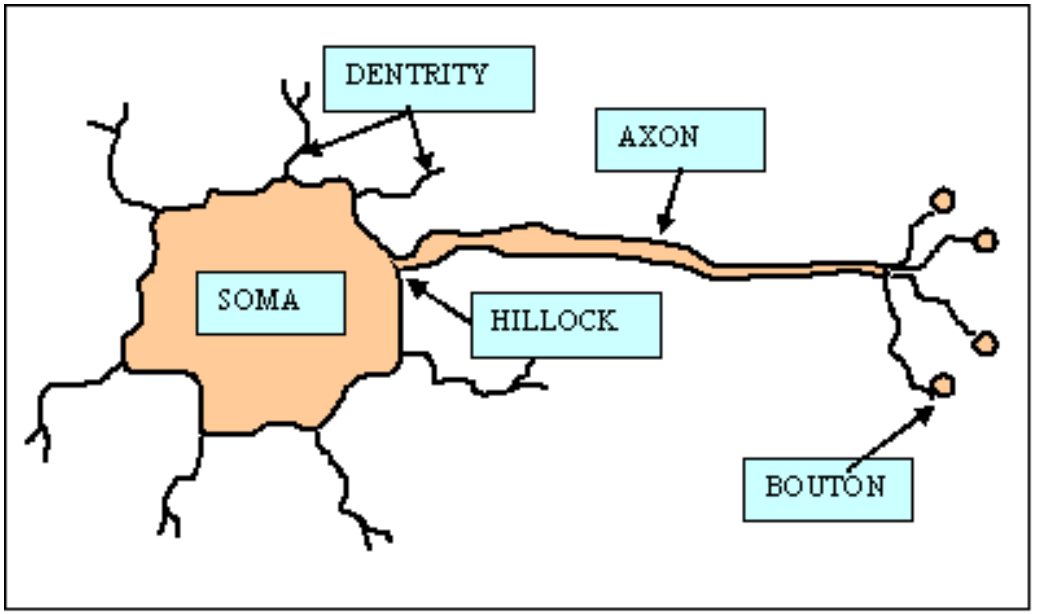
Neuronové sítě

Neuronové sítě (NS)



NS

Biologický neuron



Umělý neuron

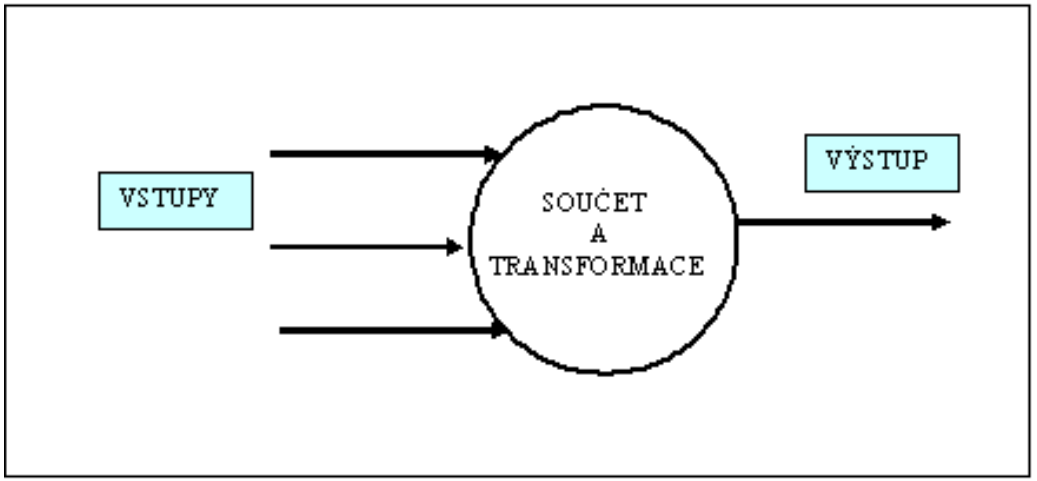
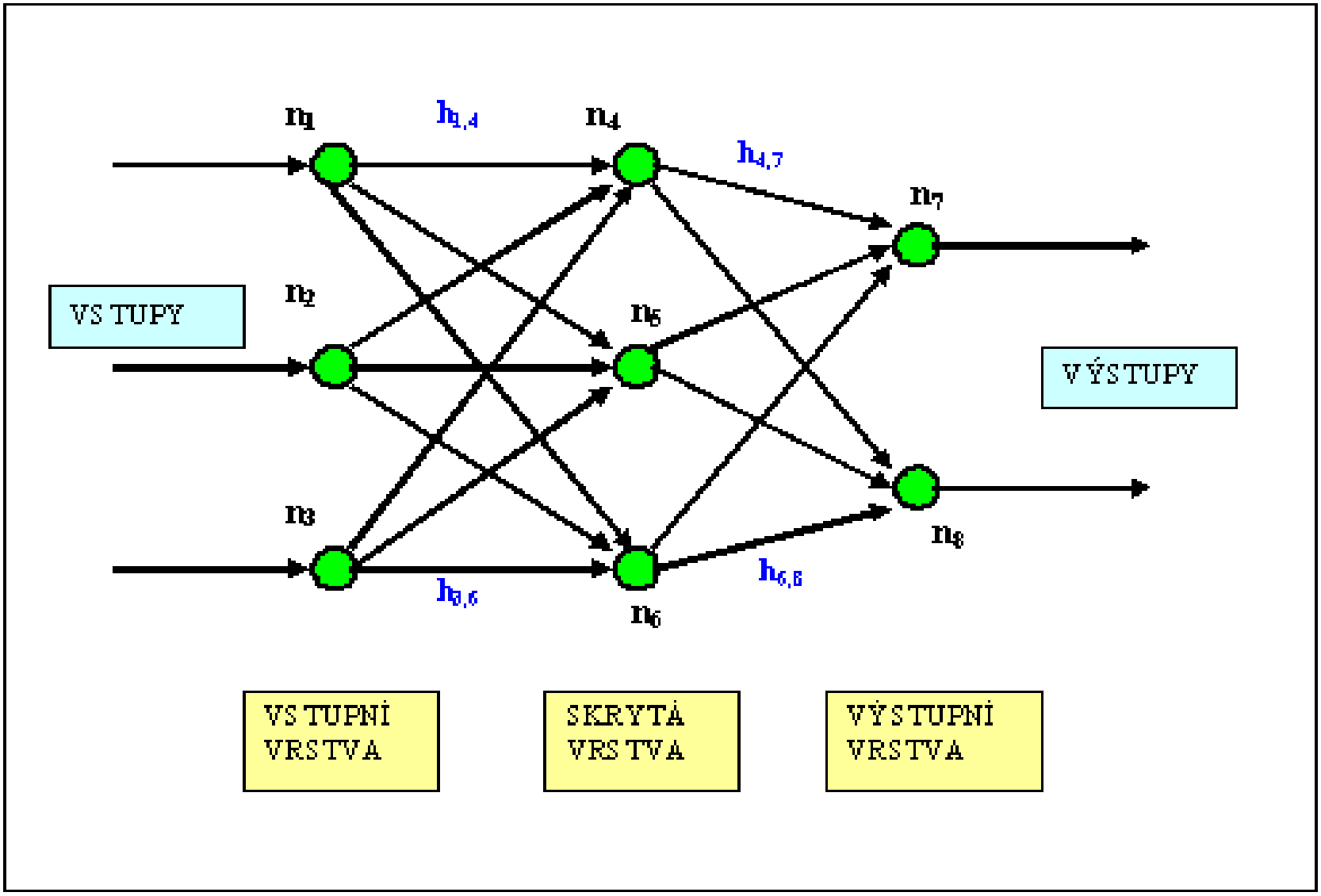
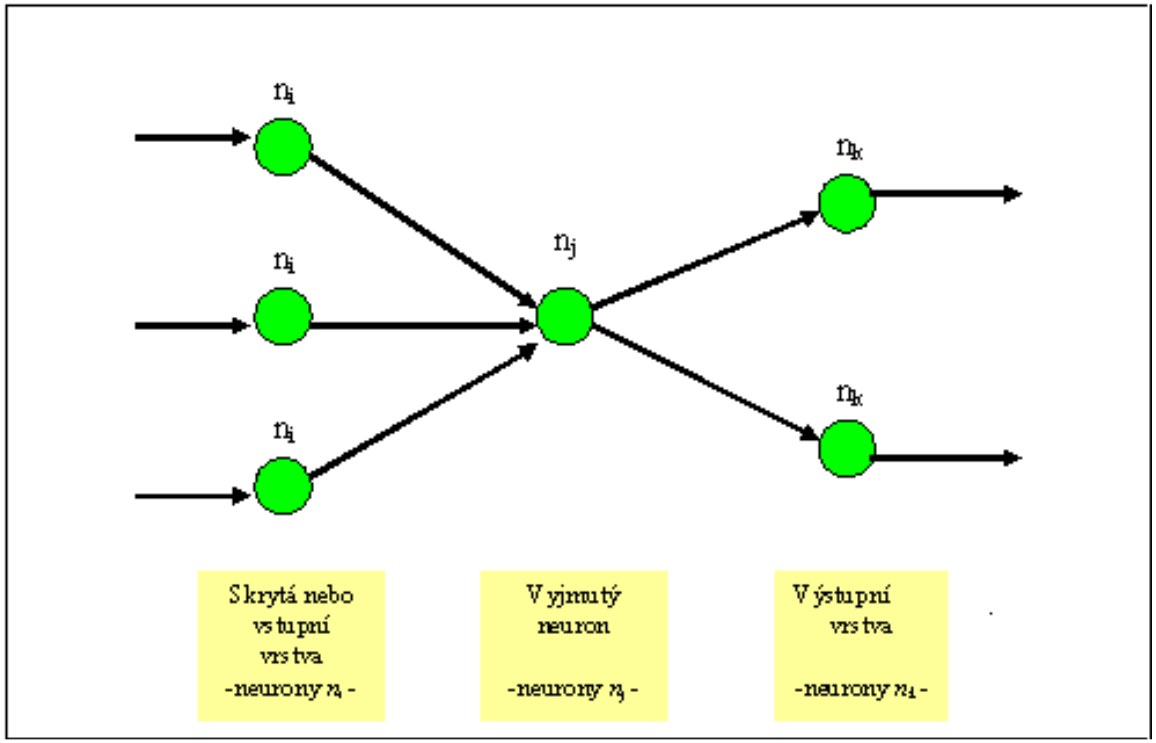


Schéma NS – graf G(N,h)



Neuron n_j vyjmutý ze skryté vrstvy



Aktivační funkce neuronu n_j

$$A_j = \sum_{i=u}^{v} W_{i,j}.O_{i,j} - \theta_j$$

Vzplanutí neuronu n_j

$$\sum_{i=u}^{v} W_{i,j}.O_{i,j} > \theta_j$$

Výstup z neuronu n_j

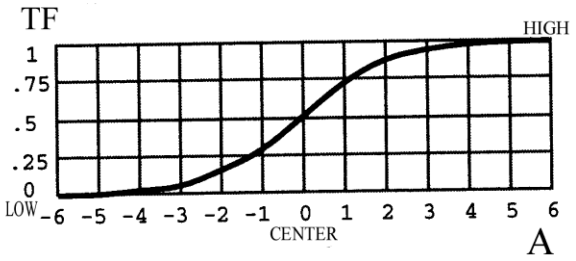
$$O_j = TF_j.(\sum_{i=u}^{v} W_{i,j}.O_{i,j} - \theta_j)$$

Lineární neuronová síť

$$O_j = A_j = \sum_{i=u}^{v} W_{i,j}.O_{i,j} - \theta_j$$

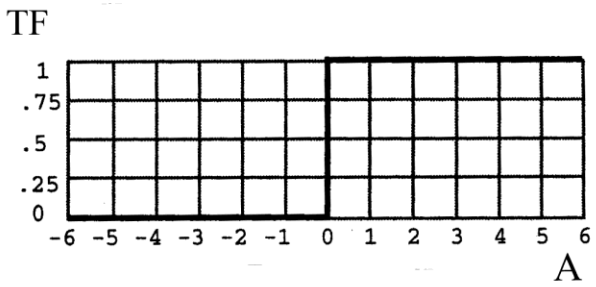
Přenosové funkce TF(.)

- **Sigmoida**

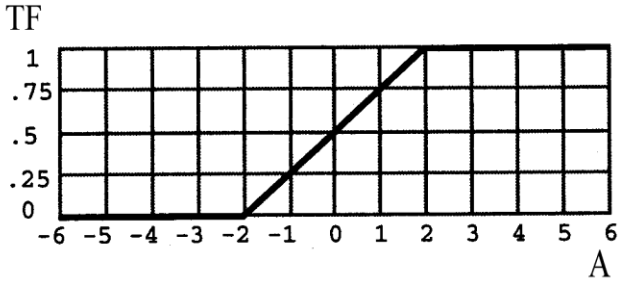


$$O_j = \frac{1}{1+\exp[-\beta.(\sum_i W_{i,j}.O_{i,j} - \theta_j)]}.$$

- **Skoková**



- **Lineární prahová**



- **Gaussova**

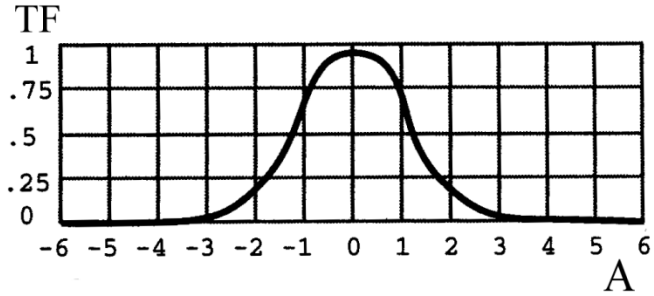
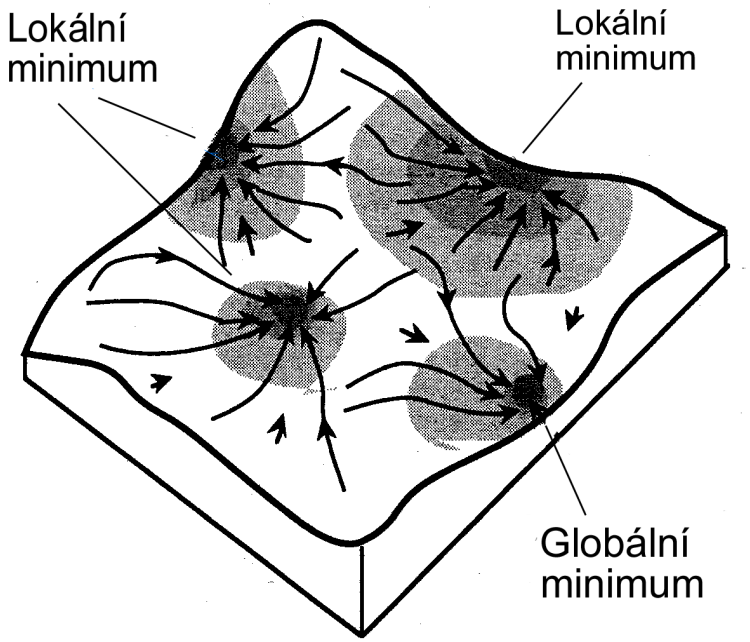


Schéma tréninkové matice

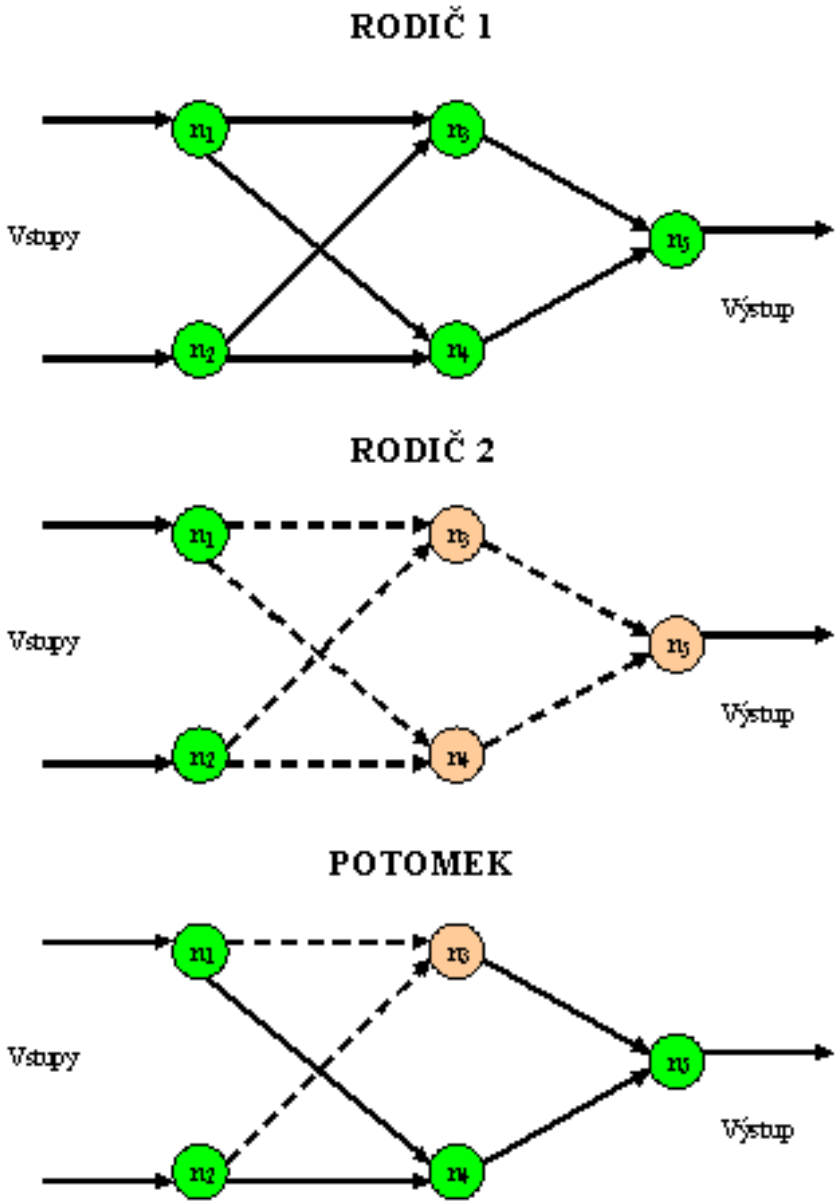
	↓	↓	↓	↓	↑	↑
i	**X1**	**X2**	**X3**	**X4**	**Y1**	**Y2**
1	0.4	12.7	5.1	26.5	6.8	0.3
2	0.8	15.6	3.8	21.7	3.3	0.2
3	0.6	11.1	4.3	29.6	4.4	0.5
.	.	.	.	.	.	.
.	.	.	.	.	.	.
n	0.3	8.5	3.7	18.9	5.1	0.4



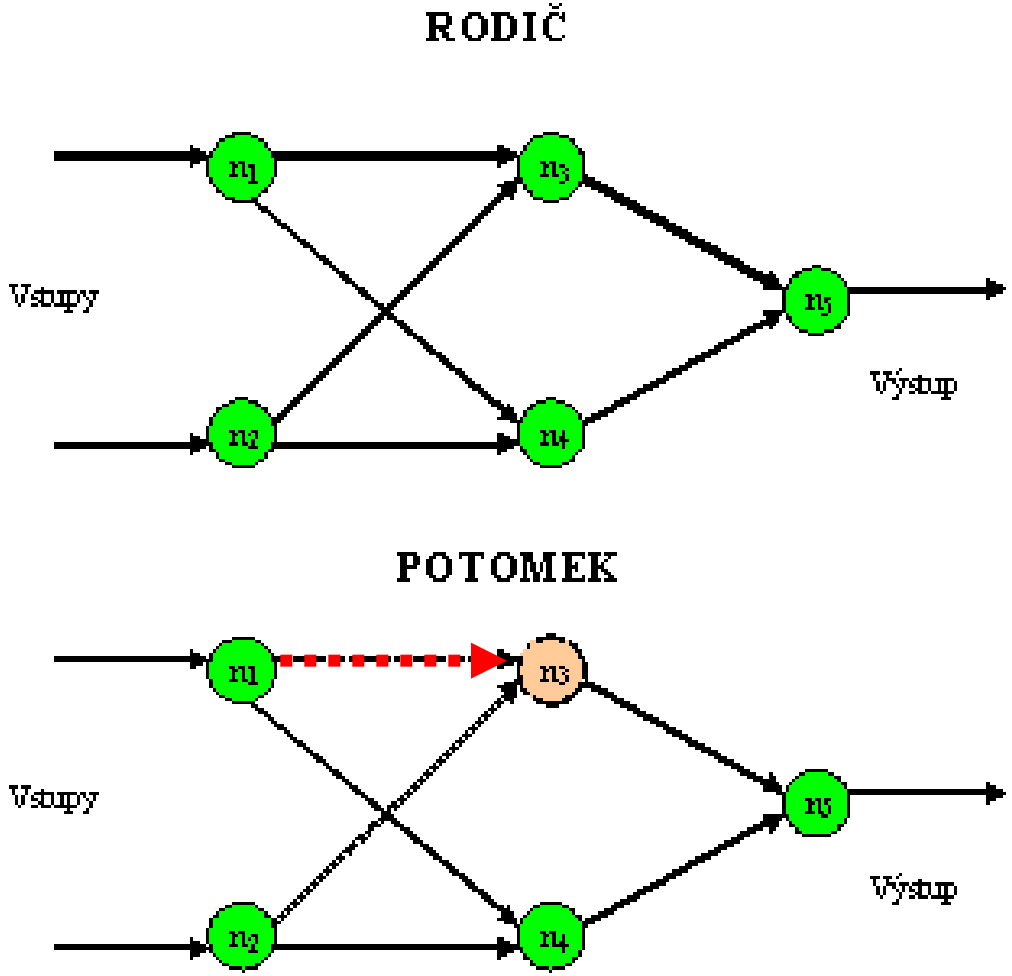
Metody trénování NS

- **Metoda zpětného šíření**
- **Genetické algoritmy**
- **Mřížková metoda**
- **Gradientní metody**

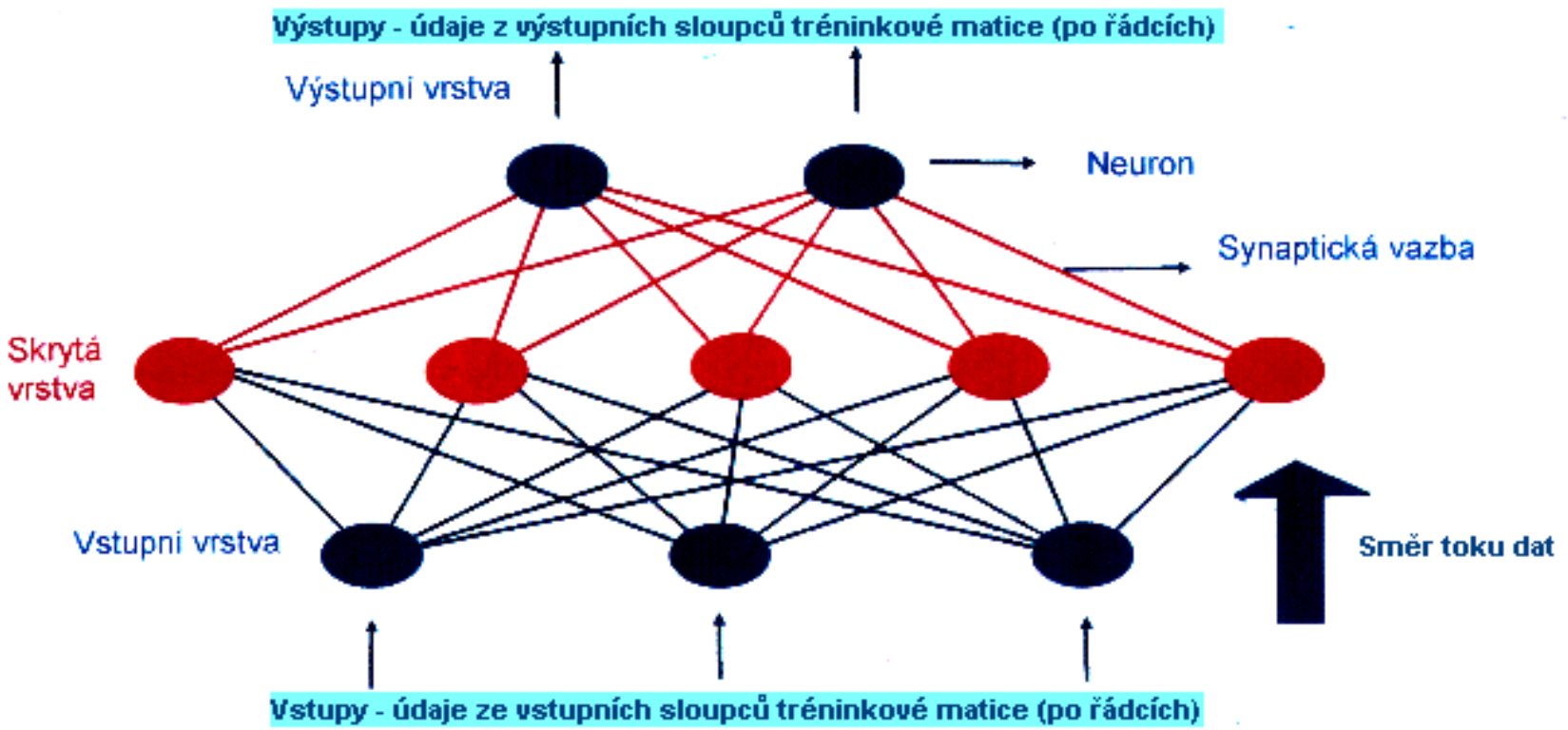
BrainMaker + GTO - křížení



BrainMaker + GTO - mutace



Neuronové sítě - shrnutí



Funkce neuronu ve skryté vrstvě

Výstup vstupuje do každého neuronu ve výstupní vrstvě

- vynásobte každý vstup váhou spojnice
- sečtěte všechny tyto výsledky
- dosaďte součet do přechodové funkce F ("signoida")
- zašlete výstup z každé F do každého neuronu v horní vrstvě

Vstupy z dolní vrstvy neuronů

- Každá spojnice má svou vlastní váhu
- Váhy se postupně mění během trénování množiny

Simulátory NS

- **BrainMaker Professional +GTO** (California Scientific Software)
- **ARTINT** (Mentar +EGU)
- **MATLAB + Neural Networks Toolbox**
- **STATISTICA**
- **WinNN**
- **Neural Connection**
- **Neurex**

NS - topologie

- Které veličiny vstupní a které výstupní (počet vstupních a výstupních neuronů) – dáno charakterem řešené úlohy
- Počet vrstev skrytých neuronů a počet neuronů v nich (Kosko – 1 skrytá vrstva a v ní 2m+1 neuronů = univerzální aproximátor)
- Počet parametrů (vah) větší než počet vzorů (Kůrková), přetrénování (overfiting) – zapamatování vzorů, ztráta generalizační schopnosti (převážená síť)
- Čím méně vzorů v porovnání s počtem neuronů máme, tím větší pravděpodobnost, že síť natrénujeme, ale ne správně. Narůstá pravděpodobnost libovolného vztahu mezi vstupy a výstupy

- Lawrencová $$TV > \frac{SN}{PT}.VN$$

  Počet tréninkových vzorů (TV) ve vztahu k počtu skrytých neuronů (SN), povolené toleranci (PT) a počtu vstupních neuronů (VN)
- Praxe: *TV > SN . (1/PT)*

NS – topologie – pokr.1

- Jaký je nejlepší počet skrytých neuronů? Existují jen doporučení!!
- Kolik vrstev skrytých neuronů? Existují jen doporučení!! Více vrstev prodlužuje výrazně dobu trénování. Pokud nejde vícevrstvá síť natrénovat => odebrat vrstvu a znovu opakovat trénování
- Past lokálního minima – přidání náhodného šumu po natrénování a pokračování v tréninku (rozkmitání)
- Přidání náhodného šumu zvláště když malý počet vzorů
- Přidání náhodného šumu zvláště když mnoho nul ve vzorech
- Doplňování nových vzorů k již natrénované síti a dotrénování – jde pouze několikrát, pak trénink kompletně znovu
- Zpočátku jde síť dobře trénovat a pak se trénink zpomalí a skoro zastaví – tréninková matice obsahuje špatné vzory (lze připustit určité procento špatných vzorů)
- Někdy vyřeší problém přidání dalších vzorů ty podpoří „zdánlivě špatné vzory“

NS – topologie – pokr.2

- Špatné vzory umožňují vyloučit analyzátory tréninkové matice
- Dynamické neuronové sítě (Dynamic Node Creation) – začnou trénink s malou sítí a pokud nejde natrénovat, postupně přidávají do skrytých vrstev neurony a skryté vrstvy. Vždy začínají trénink znovu.
- Funguje i opačný postup – velká síť má malé generalizační schopnosti
- Jde dělat manuálně – po natrénování sledovat výstupy z neuronů (příliš velké nebo malé, dva stejné) => neuron lze vyjmout
- Skeletonizace (Mozer, Smolensky) algoritmus pro posuzování vlivu neuronu na funkci dalších neuronů
- Rychlost učení určuje velikost oprav neznámých veličin v procesu učení (váhy, prahy, strmosti, atd.)
- Vyhlazovací faktor určuje míru vlivu, se kterým jsou uvažovány minulé korekce opravovaných veličin, a míru vlivu, se kterým jsou uvažovány korekce nově vypočtené (konjugované gradienty)

NS – topologie – pokr.3

- Rekurentní neuronové sítě se snaží zdokonalit proces učení zavedením spojů mezi neurony jednotlivých neuronových vrstev. Systém má zabránit tomu, toto doplnění má zabránit tomu, aby dva neurony z téže vrstvy reagovaly na vstupní data totožným způsobem
- Rekurentní neurony vznikly jako pokus obohatit neuronové sítě o časový rozměr. Neuronové sítě s rekurentními neurony se hodí pro zpracovávání dat, která tvoří časové řady. Každý neuron ve skryté vrstvě má svůj vlastní rekurentní neuron, který zpracovává svůj předešlý výstup
- Optimalizace topologie a postupu trénování neuronové sítě pomocí mřížkové (grid) metody
- Kompetiční model s Kohonenovou mapou tedy provádí *shlukovou analýzu tréninkové množiny*, tj. určení počtu shluků vstupů tréninkové množiny a jejich rozmístění ve vstupním prostoru. Model je možno použít jako *reduktoru mohutnosti tréninkové matice*

NS – standardizace dat

- Standardizací vstupních dat rozumíme transformaci, která tato data převede na požadované intervaly
- Standardizaci vstupních nebo výstupních dat. V zásadě může být tato transformace *nelineární* nebo *lineární* (výstup – TA(.) sigmoida)
- Transformace vstupů do menších hodnot (větší citlivost sigmoidy) a nejlépe symetricky kolem nuly. Velikost intervalu, na který transformujeme, určuje význam příslušného vstupu. Lze cíleně využít.
- Nejčastěji transformujeme vstupní veličinu x z $N(\mu_x, \sigma_x)$ na x^t z $N(0,1)$

$$x^t = \frac{x - \mu_x}{\sigma_x}$$

- Lze využít i lineární transformaci výstupu z O_k na O_k^t

$$O_k^t = \frac{O_k - O_{\min}}{O_{\max} - O_{\min}},$$

NS a regresní rovnice

Částečná analogie mezi regresními rovnicemi a neuronovými sítěmi

- **Vícerozměrná lineární regrese**

$$Y = a_1.X_1 + a_2.X_2 + a_3.X_3 + ... + a_p.X_p,$$

- **Jednosměrná lineární neuronová síť** s jedinou vrstvou vstupních neuronů, jediným výstupním neuronem a s nulovým prahem u všech neuronů

$$O_k = W_{1k}.O_{1k} + W_{2k}.O_{2k} + W_{3k}.O_{3k} + ... + W_{pk}.O_{pk},$$

- **Sarle uvádí:** *"Výzkumníci z oboru neuronových sítí znovu vynalezli metody známé ze statistické nebo matematické literatury desetiletí či století, ale často nechápou, jak tyto metody fungují"*

NS a RR – pokr.

Výhody NS

- Neuronové sítě nevyžadují volbu regresní závislosti (tvar regresní rovnice)
- Učení (trénink) neuronové sítě je analogickým termínem pro kalibraci regresního modelu
- Neuronové sítě mohou obsahovat více výstupů

Závěr

- Pro jednoduché úlohy se lépe hodí použít klasické regresní rovnice. Technika kalibrace těchto modelů je značně propracovaná
- Pro složitější úlohy z oblasti vícerozměrné nelineární regrese se však jeví jako vhodnější použít neuronové sítě, zejména pokud požadujeme souběžně více výstupů

NS - využití

- Neuronové sítě jsou vynikajícím prostředkem pro rozpoznávání závislostí mezi vstupními a výstupními údaji (konstrukci varovných systémů před povodňovými průtoky v ohrožených lokalitách)
- Neuronové sítě nevynikají přesností
- Pokud však bude rychlost výpočtu důležitější než přesnost, neuronové sítě jsou tím pravým prostředkem pro řešení. Svoje uplatnění mohou pro tuto vlastnost nalézt zejména při operativním řízení vodohospodářských objektů a soustav, které probíhá za značných podmínek neurčitosti
- Pokud není možno popsat vztah mezi soubory dat exaktním vztahem, ale pokud je známo, které veličiny jsou vstupní a mají vliv na veličiny výstupní, neuronové sítě mohou nalézt požadovaný vztah. Ať už je absence exaktního vztahu dána jeho značnou složitostí nebo prostě tím, že není znám
- Výhodou neuronových sítí rovněž je, že pokud se proces, který řídíme, resp. analyzujeme a vyhodnocujeme poněkud změní, je postačující doplnit nové tréninkové vzory do tréninkové matice a provést přetrénování neuronové sítě (snadnější než nový vzorec)