

Aproximace obvodů

Metody založené na modifikaci el. parametrů obvodu

Změna pracovní frekvence, či napájecího napětí...

Funkční aproximace ← zaměření práce

Cílené změny ve specifikaci logické funkce obvodu.

Změny vedou - ke **snížení plochy, zpoždění a příkonu**

- k chybám → aplikace, které **tolerují chyby**

Lze použít ve zpracování obrazu, v dataminingu, v počítačovém vidění, v klasifikaci, ve vyhledávání...

Aproximace je přirozená!

Zkuste vypočítat, zda platí: $17 \cdot 49 > 1$ a $17 \cdot 49 > 660$

Spotřeba energie procesoru je **shodná** u obou výpočtů.

Evoluční návrh

Evoluční design

- Vytváří inovativní řešení.

- Můžeme objevit lépe aproximovaná řešení, než které poskytují konvenční metody.

Problémy, které přináší návrh pomocí evoluce

- škálovatelnost použití

- doba ohodnocení ← zaměření práce

Evoluční návrh je přirozený proces!

Vychází z Darwinovy teorie evoluce. Genetická informace se u nadprůměrně kvalitních jedinců objevuje v dalších generacích častěji než u méně nekvalitních.

Kartézské genetické programování - CGP [1]

Obvod je reprezentován grafem, který je kódován chromozomem.

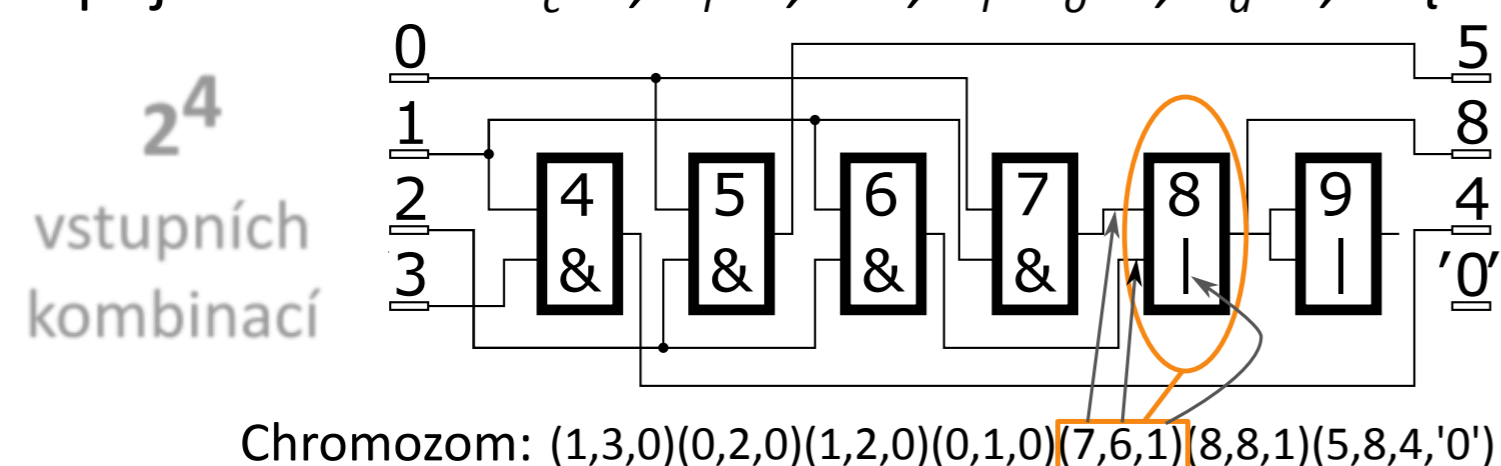
Parametry grafu, uzlů a evoluce

- počet sloupců n_c , řádků n_r , míra propojení sloupců l , počet primárních vstupů n_i a výstupů grafu n_o

- počet vstupů (arita operace) n_a a jejich operace $n_f \in \Gamma$

- počet generací n_g , počet potomků λ , četnost mutací h

Př. zapojení obvodu $n_c=6, n_r=1, l=4, n_i=n_o=4, n_a=2, \Gamma=\{AND^0, OR^1\}$



S rostoucím počtem vstupů roste doba ohodnocení obvodu exponenciálně → potřeba různých akceleračních technik

- přeskočení neutrálních mutací, nepoužitých hradel
- **paralelní simulace** kandidátního obvodu v 64 bit. registrech
- předkompilace simulace obvodu do strojového kódu
- heuristická inicializace grafu pomocí aproximovaných obvodů, u kterých je hradlo nahrazeno propojkou [2]

Fitness funkce - Suma absolutních diferencí (SAD) [3]

Počet případů fitness → K Odezva kandidátního obvodu

$$f = \sum_{j=1}^K |v(j) - t(j)|$$

Požadovaná odezva (reference)

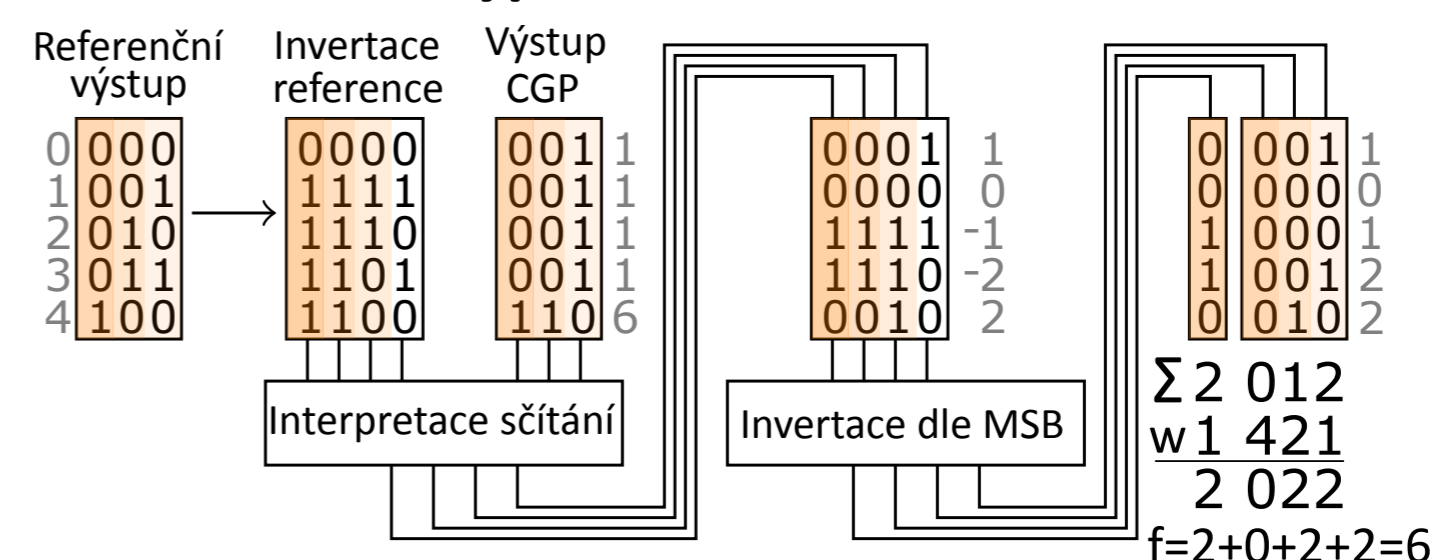
Dosavadní přístup výpočtu fitness

- bitová transpozice vektorů na celočíselné hodnoty (A0)

→ nevhodné při paralelní simulaci obvodu

- celočíselné hodnoty znázorněny šedou barvou

Navržená metoda - výpočet SAD na vektorové úrovni



- interpretace odčítání a získání absolutní hodnoty pomocí bitových operací (A1)

- předkompilace metody A1 do strojového kódu (A2)

- metody porovnány podle vztahu:

$$eps = n_g * \lambda / runtime$$

Násobička	$n_g * 10^6$	A0[eps]	A1[eps]	A2[eps]
4x4	10	54 912	277 923	226 222
5x5	5	12 052	168 688	186 781
6x6	1	2 683	61 673	129 541
7x7	0.1	602	17 644	53 930
8x8	0.01	135	4 782	23 190

170x urychlení

Aproximované 8-bitové násobičky v detekci hran



Aproximace výpočtu normály dle Sobelových hodnot.

Násobička při výpočtu nevyužívá všechny kombinace.

Černou barvou

- hodnoty se násobily alespoň jednou

Bílou barvou

- hodnoty se nikdy nenásobily

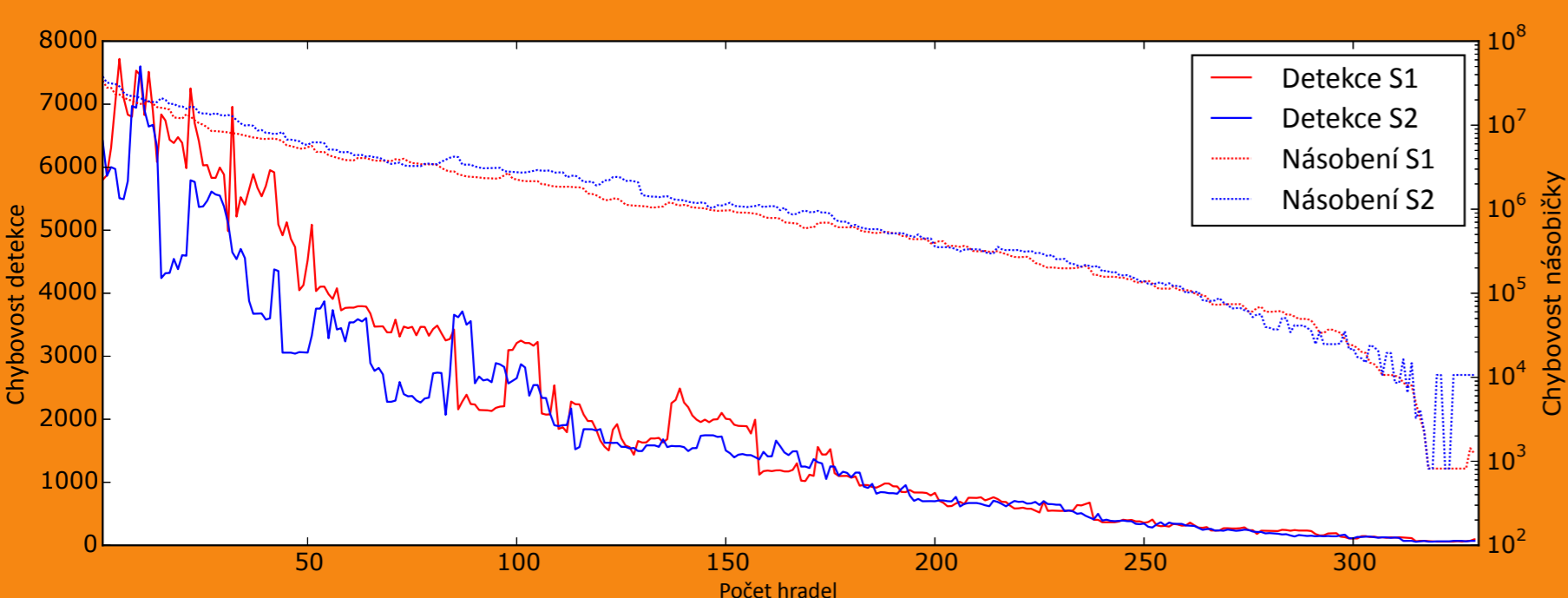
Scénář **S1** ohodnocuje obvod podle **úplné specifikace**.

Scénář **S2** ohodnocuje obvod podle **částečné specifikace**.

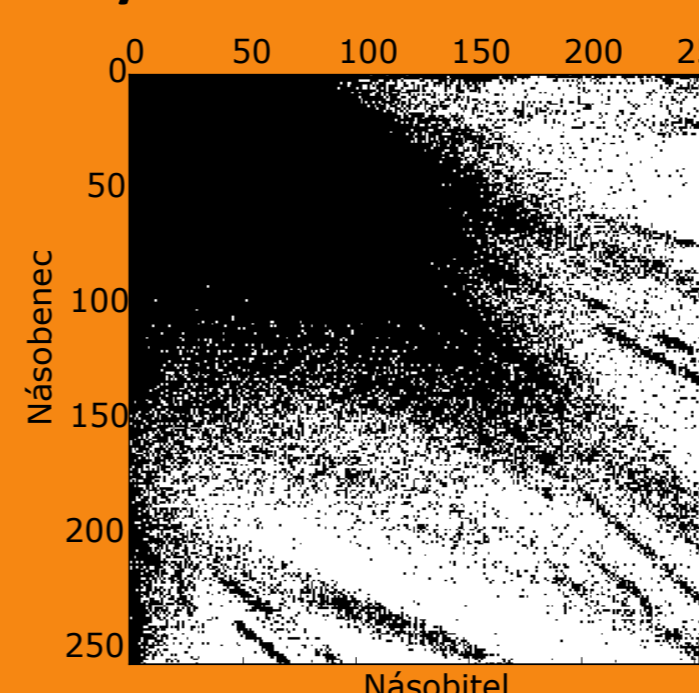
Metoda **S2 urychlila** ohodnocení obvodu **dvojnásobně**.

S2 má **hůře** aproximované násobičky než S1.

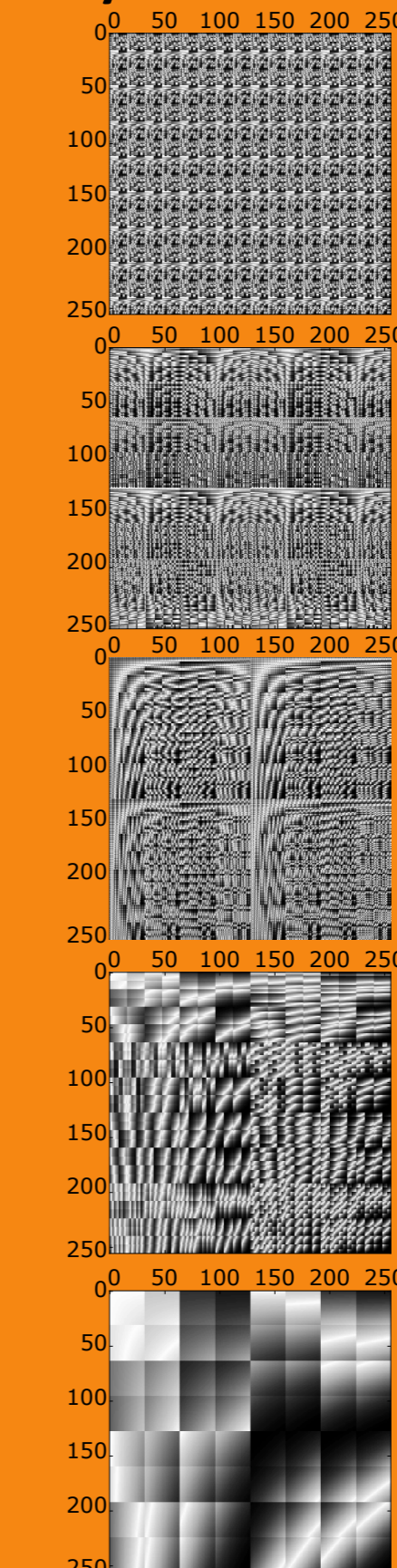
Detekce podle S2 je podobná a někdy i lepší než S1.



Využití násobení v detekci



Chybovost násobení



Detekce



Vlastnosti aproximovaných násobiček

Počet hradel	Relativní plocha	Relativní zpoždění
1	40	1.76
66	3 472	36.11
132	6 760	39.39
198	10 096	48.64
264	13 824	52.56
330	16 752	56.66

[1] Julian F. Miller, *Cartesian Genetic Programming*, Natural Computing Series, Springer Berlin, Heidelberg, 2011

[2] Zdeněk Vašíček, Lukáš Sekanina, *Evolutionary approach to approximate digital circuits design*. IEEE Transactions on Evolutionary Computation, 2015

[3] Lukáš Sekanina, Zdeněk Vašíček, *Approximate circuits by means of evolvable hardware*. In 2013 IEEE International Conference on Evolvable Systems (ICES), 2013