

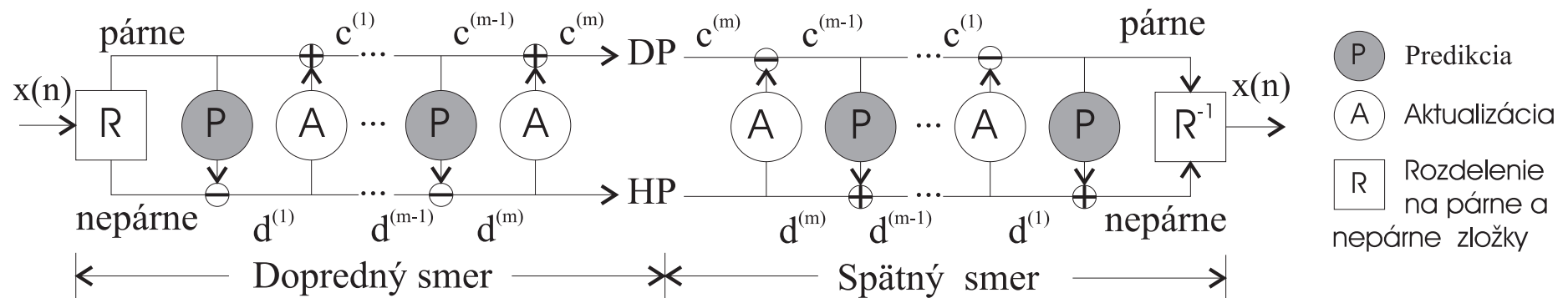
Lifting schéma

- Lifting schéma umožňuje jednoducho zlepšiť špecifické vlastností danej WT – odtiaľ názov „*lifting*“
- Jednoducho opisuje závislosti medzi párami filtrov, ktoré zdieľajú ten istý HP, resp. DP filter. Takto môžeme začať z triviálneho prípadu "lenivého" waveletu a postupne vybudovať pár filtrov s požadovanými vlastnosťami, t.j. postupujeme podľa tzv. *Lifting schémy*
- Umožňuje efektívne realizovať klasické WT s nasledovnými výhodami:
 - urýchlenie implementácie WT (napr. v 1D prípade až dvojnásobne)
 - možnosť vykonať výpočty bez použitia prídavnej pamäte (t.j. "in-place")
 - jednoduchý dizajn vlastných WT (naviac so zaručenou invertovateľnosťou)
- Umožňuje jednoducho rozširovať klasickú WT, zaujímavé sú napr.
 - konštrukcia nelineárnych WT (napr. adaptívnych, celočíselných)
 - použitie WT pre nerovnomerne navzorkované signály
 - konštrukcia WT na intervaloch, krivkách, povrchoch

Konštrukcia lifting schémy je založená na koncepte polyfázového rozkladu bánk filtrov:

- každú FB, môžeme rozložiť (faktorizovať jej filtre) na jednoducho invertovateľnú postupnosť krokov
- Tieto kroky tvoria v algoritme priečkovú štruktúru, ktorú môžeme interpretovať ako postupnosť *predikcií* a *aktualizácií* dvoch množín v obraze až po ich vzájomnú dekoreláciu

Principiálne môžeme waveletovú transformáciu implementovanú lifting schémou prekresliť podľa nasledovného obrázku. Sú vyznačené základné bloky lifting schémy: *rozdelenie*, *predikcia* a *aktualizácia*:



- cieľom predikcie je získanie čo najmenších hodnôt v HP časti po doprednej transformácii.
- aktualizáciou sa snažíme zachovať v DP časti čo najviac vlastností pôvodného obrazu, čo sa stáva dôležitým najmä pri rekurzii v DP časti (aby bola predikcia aj naďalej účinná)

Príklad A: Lenivý wavelet

Aké vlastnosti má naše spektrum?

Vstupný signál je jednoducho podvzorkovaný, takže máme najhorší možný aliasing. Charakter oboch zložiek signálu je rovnaký.

Príklad B: Haarov wavelet

V DP časti chceme dostať priemer susedných koeficientov, v HP časti zase rozdiel. Ak je teda funkcia po častiach konštantná dostávame v HP časti nulové koeficienty.

1) Signál si rozdelíme na párne a nepárne časti:

$$c^{(0)}(n) = x(2n) \quad d^{(0)}(n) = x(2n+1)$$

2) Aktualizujeme priemer (nenormovaný):

$$c^{(1)}(n) = c^{(0)}(n) + d^{(0)}(n)$$

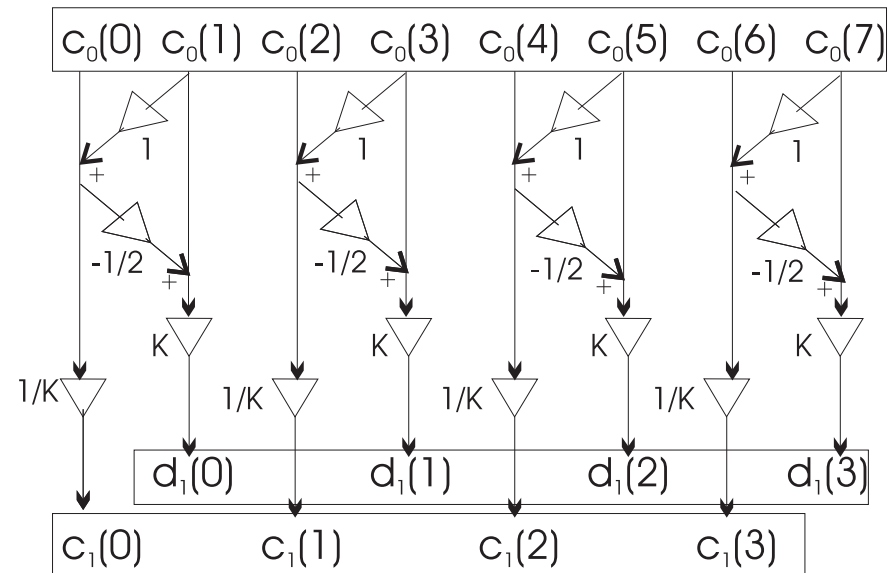
3) Predikujeme HP časť:

$$d^{(1)}(n) = d^{(0)}(n) - 0.5c^{(0)}(n)$$

4) Normalizujeme ($K = \sqrt{2}$):

$$d(n) = Kd^{(1)}(n) = \frac{\sqrt{2}}{2}(x(2n) - x(2n+1))$$

$$c(n) = \frac{1}{K}c^{(1)}(n) = \frac{\sqrt{2}}{2}(x(2n) + x(2n+1))$$



Príklad C: CDF 2,2 wavelet

Čo ak chceme reprodukovať po častiach lineárne funkcie?

1) Najprv si rozdelíme signál na párne a nepárne časti:

$$c^{(0)}(n) = x(2n) \quad d^{(0)}(n) = x(2n+1)$$

2) Waveletové koeficienty nám vtedy určujú mieru, ako sa náš signál líši od lineárneho:

$$d^{(1)}(n) = d^{(0)}(n) - \frac{1}{2} [c^{(0)}(n) + c^{(0)}(n+1)]$$

3) V DP časti chceme zachovať aspoň priemer, t.j. jednosmernú zložku signálu. Hľadáme riešenie v tvare:

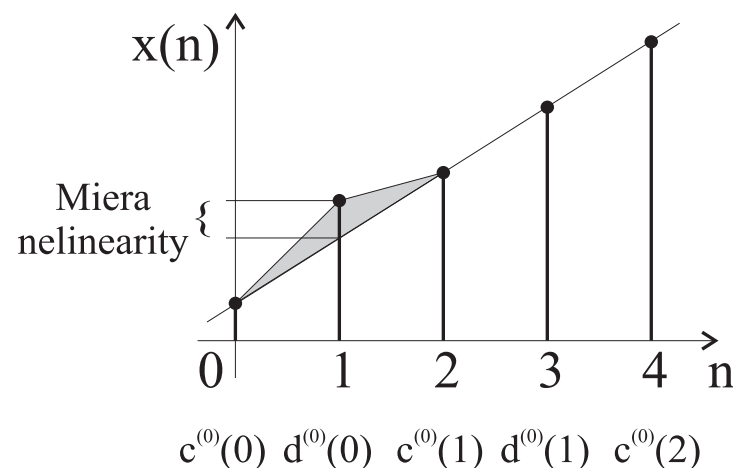
$$c^{(1)}(n) = c^{(0)}(n) + A [d^{(1)}(n) + d^{(1)}(n-1)]$$

Ak uvážime, že koeficientov $c(n)$ je polovičný počet ako $x(n)$, musí platiť:

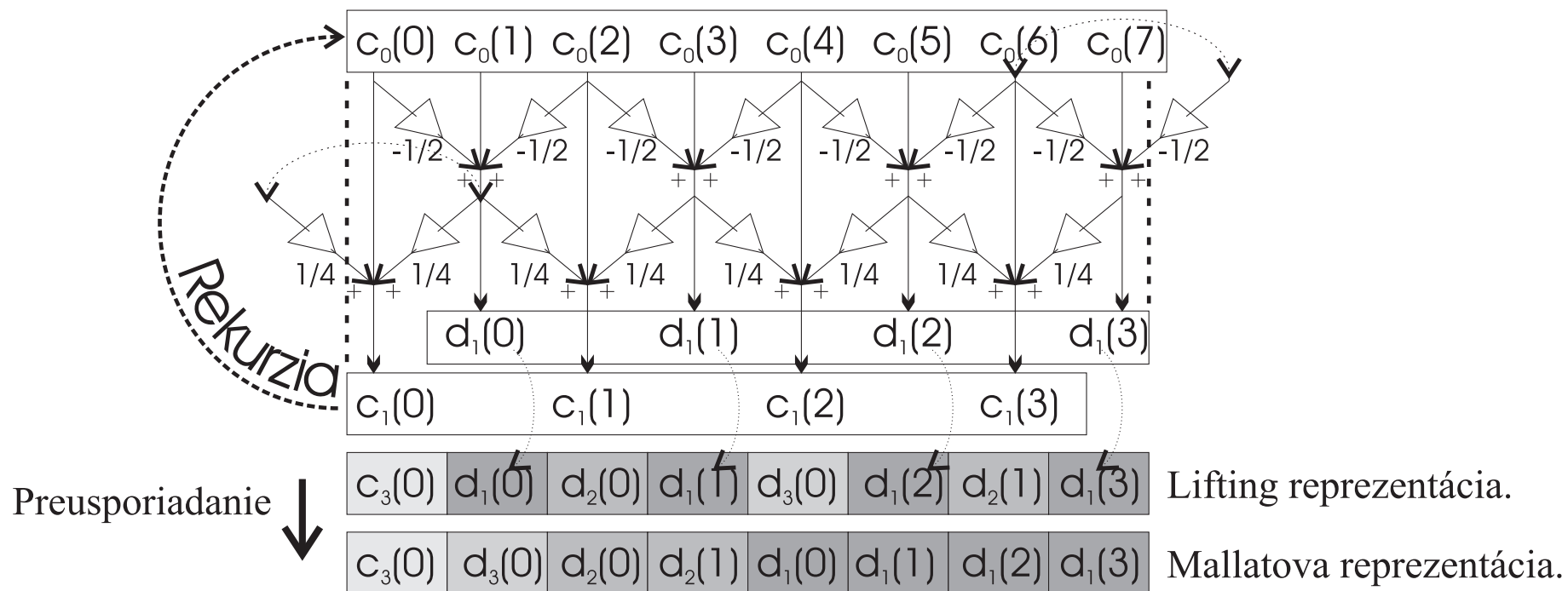
$$\sum_n c^{(1)}(n) = \frac{1}{2} \sum_n x(n)$$

dostaneme hodnotu $A=1/4$.

3) Dostali sme CDF 2,2 filter s $\tilde{h}(n) = \left(-\frac{1}{8}, \frac{1}{4}, \frac{3}{4}, \frac{1}{4}, -\frac{1}{8}\right)$



Reprezentácia signálu po liftingu (príklad CDF 2,2 wavelet)



O prediktoroch

- Vlastnosti výslednej transformácie sú určené vlastnosťami a spôsobom aplikácie prediktorov
- Teoreticky môžeme použiť ľubovoľné prediktory. Priečková štruktúra nám zaručuje biortogonalitu a tým aj úplnú rekonštrukciu. Môžeme použiť nelineárne prediktory, napr. adaptívne alebo celočíselné. Celočíselné prediktory dostaneme napr. jednoduchým zaokrúhlením hodnoty (nelineárna operácia).

Koncept prediktorov predstavuje silný vzťah medzi *transformačným* a *prediktívnym* kódovaním, kde sa snažíme predpovedať (aproximovať) signál a kódovať iba prípadnú diferenciu od originálneho signálu

Lifting schéma takto principiálne umožňuje dvojaký prístup k dekorelácii dát:

- a) použiť transformačný princíp (t.j. WT) a celý postup faktorizovať na kroky liftingu
- b) využiť priamo predikčný princíp a snažiť sa dáta dekorelovať priamo dizajnom sústavy prediktorov použitých v lifting schéme

Lifting – aké je urýchlenie výpočtov?

- Výpočet FFT – komplexnosť rádu $N \cdot \log(N)$
- Výpočet DWT – komplexnosť rádu N
- Výpočet DWT liftingom – ďalšie *zníženie počtu operácií až na polovicu*.

Priemerný počet násobení a sčítaní na výpočet jedného koeficientu pri DWT jednostupňovom rozklade:

Typ waveletu	DWT klasicky	DWT liftingom	Urychlenie
Haar	3	3	0%
Db2	14	9	56%
Db3	22	14	57%
FBI (9,7)	23	14	64%
B-spline(4,2)	17	10	70%
Interpoláčné($N, N_{\tilde{}}$)	$3(N + N_{\tilde{}}) - 2$	$\frac{3}{2}(N + N_{\tilde{}})$	100%

Lifting – ako vykonať celočíselné transformácie?

- používať celočíselné prediktory (operácia zaokrúhľovania)
- problematická normalizácia pásiem
 - o nerobiť normalizáciu (zvýši nám dynamiku dát)
 - o aproximovať normalizáciu krokmi liftingu, t.j. využiť vlastnosť

$$\begin{pmatrix} K & 0 \\ 0 & 1/K \end{pmatrix} = \begin{pmatrix} 1 & K - K^2 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -1/K & 1 \end{pmatrix} \begin{pmatrix} 1 & K - 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

Intermezzo: Čo sú to interpolačné filtre?

DP Filter $\tilde{H}(z)$ nazývame *interpolačný*, ak je schopný interpolovať koeficienty prestriedané nulami (najradšej ako polynómy R-tého rádu).

Ak je filter *polpásmový* tak je aj interpolačný:

Prečo?

Lebo interpoluje párne koeficienty z nepárnych, resp. naopak (polpásmový filter má v impulzovej charakteristike okrem počiatku párne koeficienty nulové)

Ako vypočítame koeficienty interpolačných filtrov? -> T.z.v Nevillovým algoritmom.

Príklady Interpolačných filtrov: Haarov wavelet – konštantná interpolácia $h(n)=\{1,1\}$, CDF 2,2
– lineárna intarpolácia, $h(n)=\{1/2, 1, 1/2\}$, 4-bodová schéma $h(n)=\{-1/16, 0, 9/16, 1, 9/16, 0, -1/16\}, \dots$

Počítačová grafika a interpolačné delenie intervalov

- Dôležitý cieľ - lokálne opakovane(!) zjemniť danú sieť bodov a ich hodnôt – výsledkom by mala byť hladká funkcia
- Tento process si môžeme predstaviť ako inverznú waveletovú transformáciu bez “detailov”.

Príklad na CDF(2, 2) wavelet

Odpovedajú mu filtre v banke filtrov:

$$\tilde{g}(n) = \left(-\frac{1}{2}, 1, -\frac{1}{2}\right)$$

$$\tilde{h}(n) = \left(-\frac{1}{8}, \frac{1}{4}, \frac{3}{4}, \frac{1}{4}, -\frac{1}{8}\right)$$

Poznámka:

Pozor na umiestnenie nulového prvku (má nad sebou bodku)

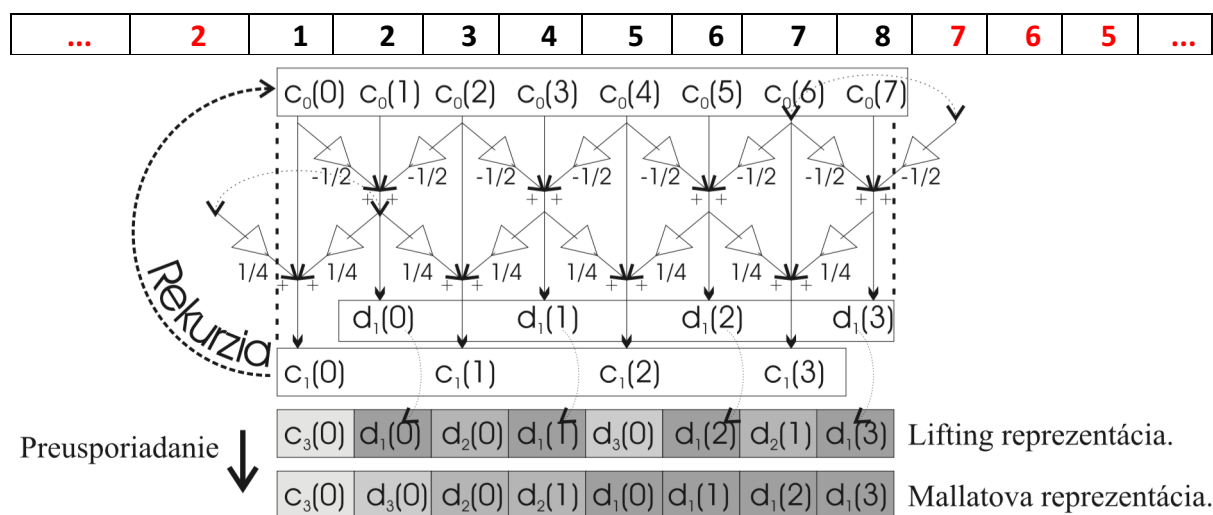
Majme signál $x(n) = (1, 2, 3, 4, 5, 6, 7, 8)$

Úlohy:

- Aká bude jeho DWT pomocou liftingu? (stačí urobiť 2 úrovne rozkladu)
- Následne signál zrekonštruujte!

Riešenie:

Zvolíme správnu symetrizáciu signálu, keďže filtre majú nepárnu dĺžku, zvolíme W symetriu (celobodovú)



n			0		1		2		3				...
$d_1(n)$			0		0		0		1				

n		0		1		2		3					...
$c_1(n)$		1		3		5		7.25					

$$d_1(n) = (0, 0, 0, 1)$$

$$c_1(n) = (1, 3, 5, 7.25)$$

Rekonštruujeme:

- Aktualizujeme $c_1(n)$ pomocou $d_1(n)$, ale namiesto pričítania odčítame!

n		0		1		2		3					...
$c_1(n)$		1		3		5		7					

- Predikujeme $d_1(n)$ pomocou $c_1(n)$, ale namiesto pričítania odčítame!

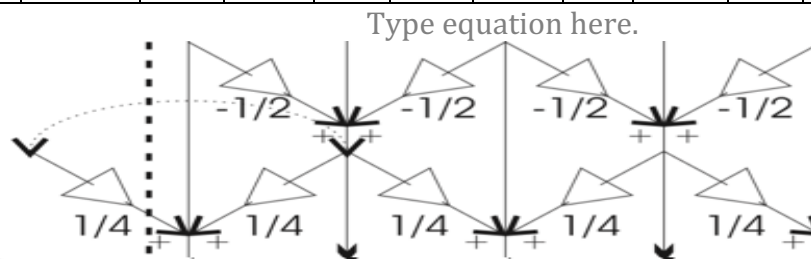
n			0		1		2		3				...
$d_1(n)$			2		4		6		8				

Výsledok:

...		1	2	3	4	5	6	7	8				...
-----	--	---	---	---	---	---	---	---	---	--	--	--	-----

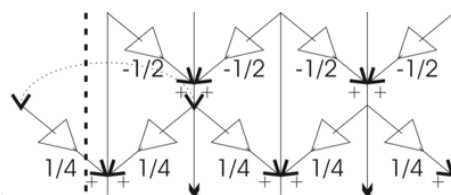
Ak by sme chceli pokračovať rozkladom (na cvičení):

n		0		1		2		3					...
$c_1(n)$		1		2		5		7.25		5			



resp. môžeme si to prekresliť do „nenatiahnutého tvaru“:

n		0	1	2	3		...
$c_1(n)$		1	2	5	7.25	5	

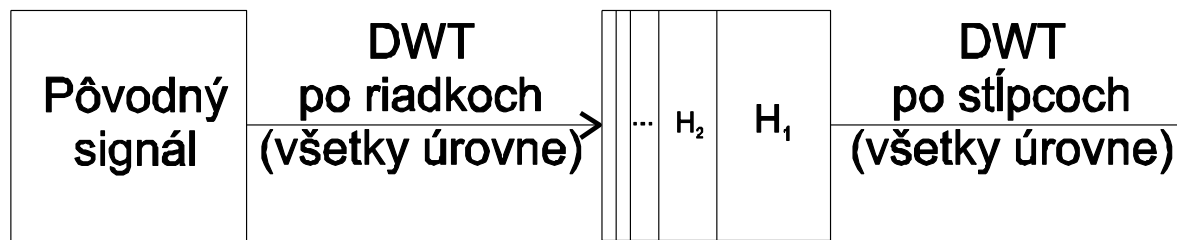


$d_2(n) = \dots$

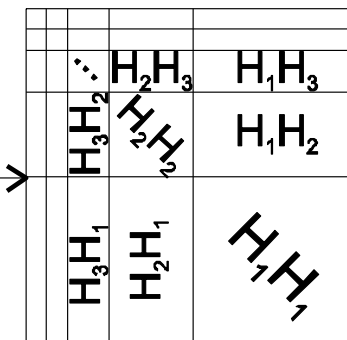
$c_2(n) = \dots$

atď ...

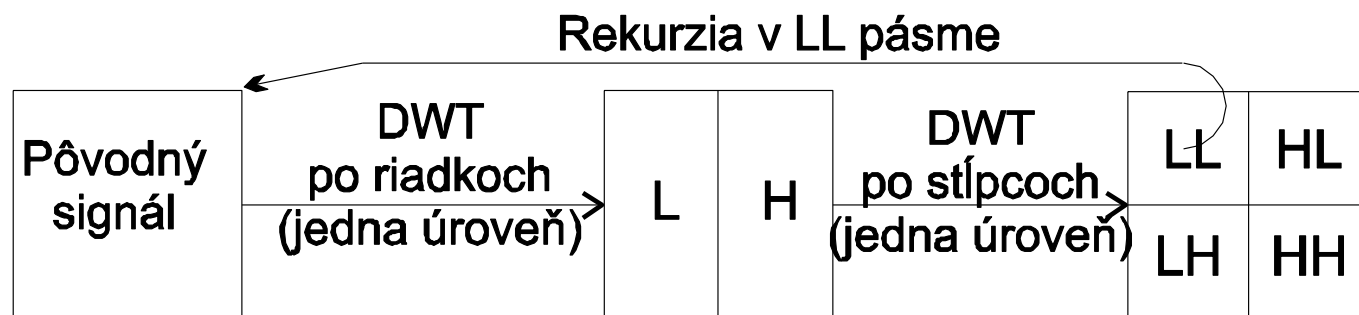
Základné riešenia 2D DWT



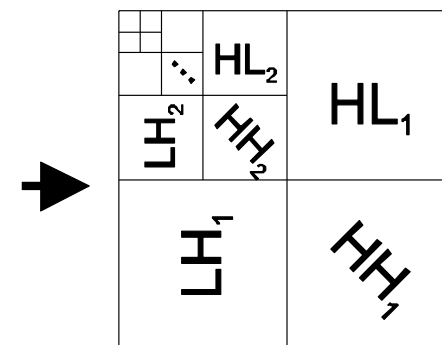
DWT, štandardný rozklad



Rozdelenie spektra do subpásiem

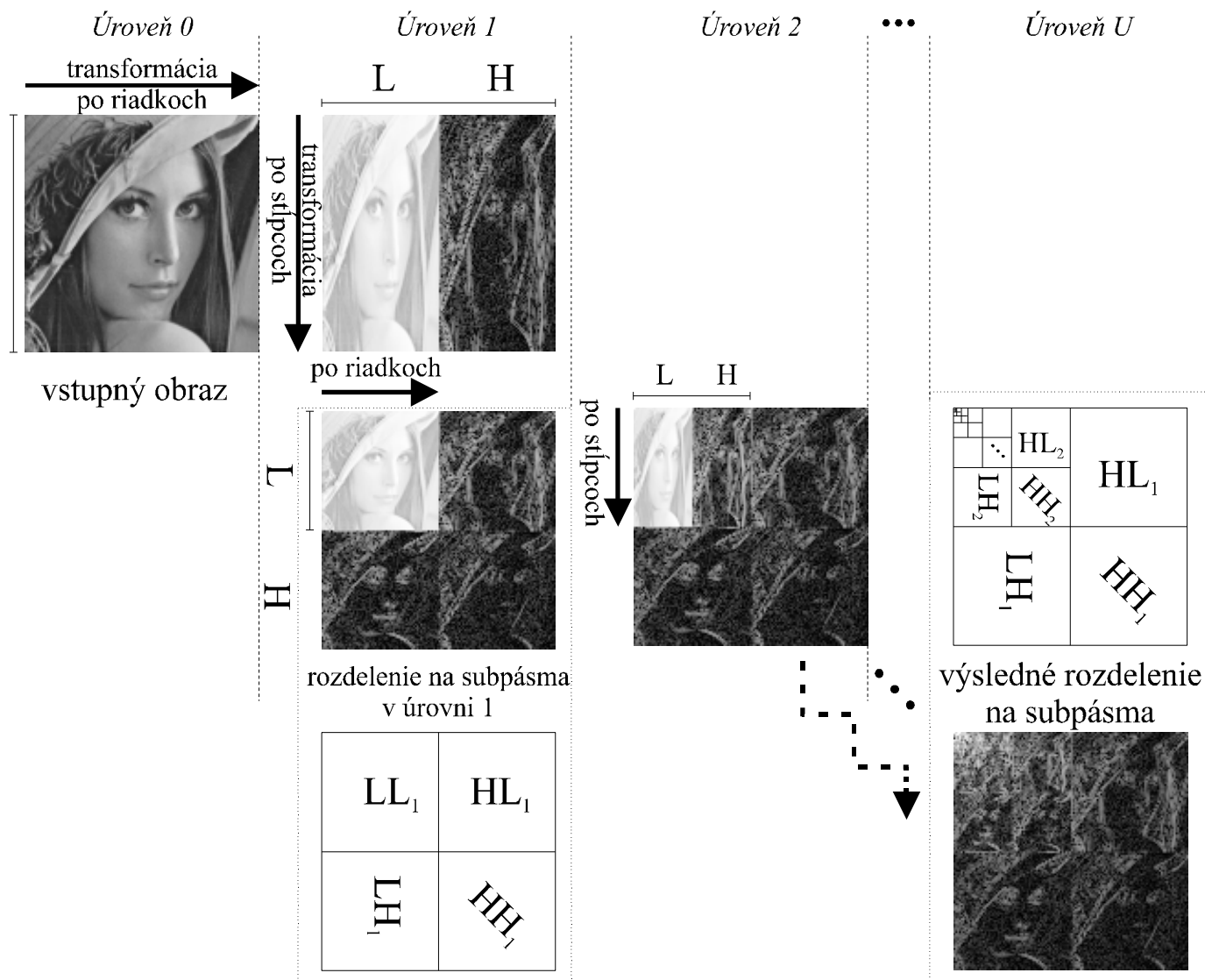


DWT, neštandardný rozklad

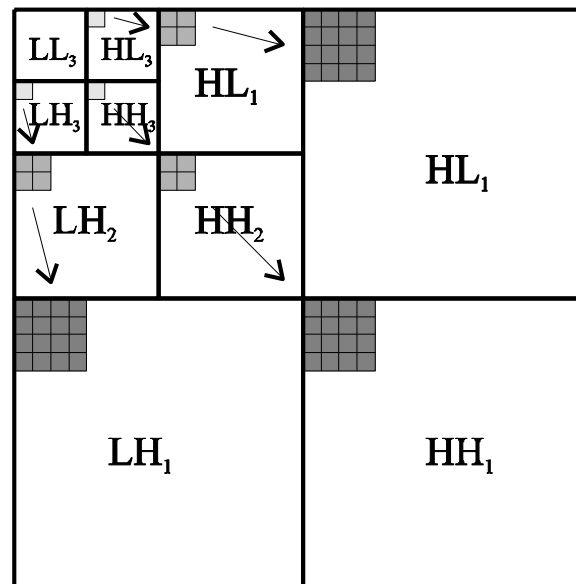


Výhody DWT s neštandardným (NS) rozkladom:

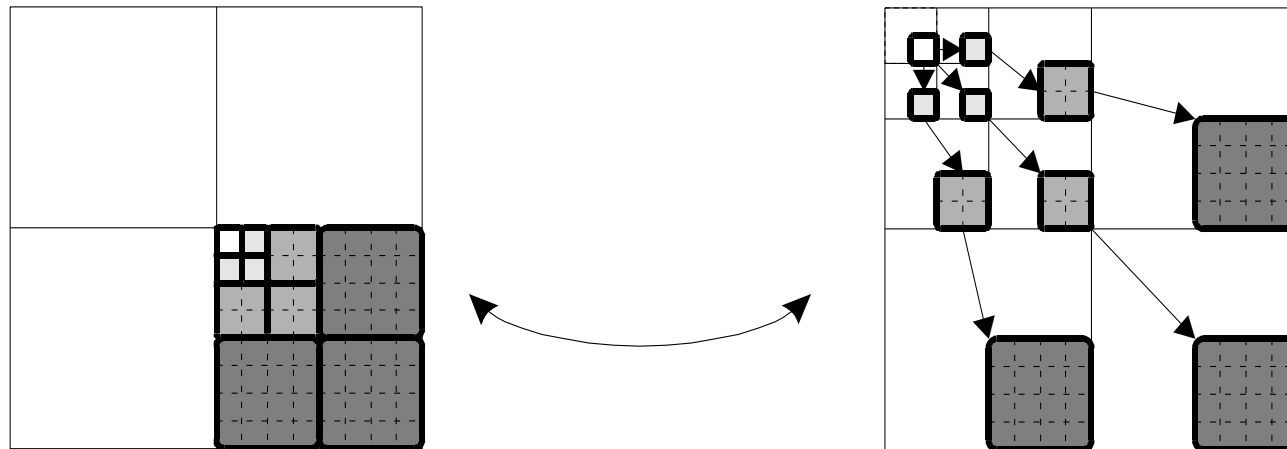
- menší počet potrebných operácií oproti štandardnému rozkladu
- lepšia reprezentácia signálu v priestore.



Spôsob výpočtu 2D DWT rozkladom na subpásma v banke filtrov
 (v každom transformačnom kroku sa signál rozdelí na časť s dolnopriepustným L (Low) a
 hornopriepustným H (high) charakterom, pričom v krokoch sa striedajú smery x a y)



Zobrazenie priestorových závislostí v NS DWT spektre 2D signálu



a) V blokoch oddelené spektrá
pri blokovej transformácii

b) Hierarchické rozdelenie
subpásom pri 2D DWT

Porovnanie tvaru spektier pri Blokovej transformácii a DWT.



a) Haar , MSE: 454.3



b) Daubechies 4, MSE: 391.24



c) CDF 5/3, MSE: 368.5



d) CDF 5/3 s vymenenými
filtrami pre analýzu a
syntézu, MSE: 546.5



e) Daubechies 20, MSE: 322.4



f) FBI 9/7, MSE: 324.3

**Aproximácia obrazu vybranými waveleovými transformáciami pri
vynulovaní 98.5 % waveletových koeficientov**

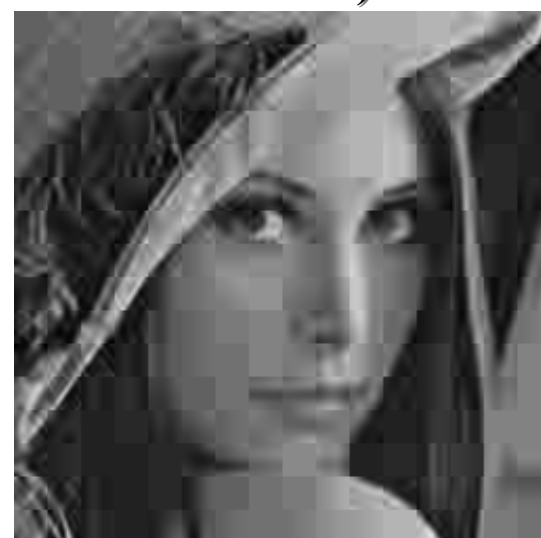
Porovnanie kompresných výsledkov štandardu JPEG a progresívneho kódovania algoritmom SPIHT použitím DWT a DCT (výrez obrazu Lena, veľkosť 256x256)



a) JPEG (quality 4, optimize, baseline)
0.139 bit/bod, MSE: 212.2



b) DCTII, bloky 8x8, SPIHT,
0.139 bit/bod, MSE: 203.38



c) DCTII, bloky 16x16, SPIHT
0.139 bit/bod, MSE: 134.64



d) DWT Daubechies 20, SPIHT
0.139 bit/bod, MSE: 137.92



e) DWT CDF 9/7, SPIHT
0.139 bit/bod, MSE: 107.3



f) DWT CDF 9/7, SPIHT
0.053 bit/bod, MSE: 241.3

Aplikácie waveletov

Spojité/diskrétné signály všeobecne

- detekcia diskontinuít (aj v deriváciách)
- zistenie charakteru signálu (škálogramy)
- odstraňovanie šumu - kompresia signálov

Audio aplikácie:

- kompresia zvuku waveletovými paketmi
- audio broadcasting
- odstraňovanie šumu
- watermarking

Spracovanie obrazu & videa

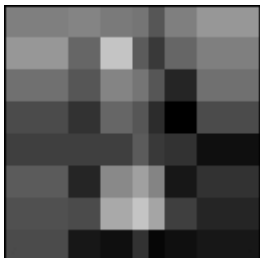
- kompresia statického obrazu a sekvencií obrazov -> JPEG 2000, Motion Jpeg 2000
- odstraňovanie šumu
- počítačová grafika
- watermarking

Prenosové systémy

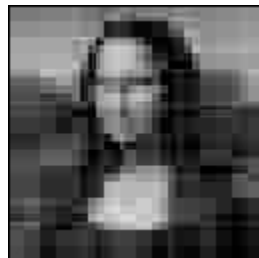
- multiplexery a demultiplexery na báze bánk filtrov

Aplikácie waveletov v počítačovej Grafike

Obrazy



20 koeficientov

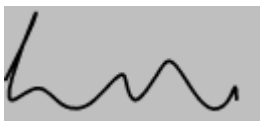


200 koeficientov



16,000 koeficientov

Krivky



úroveň 3.1



úroveň 5.4



úroveň 8.0

Povrchy



229 trojuholníkov



2,000 trojuholníkov



10,000 trojuholníkov

Simulácie



bez detailov



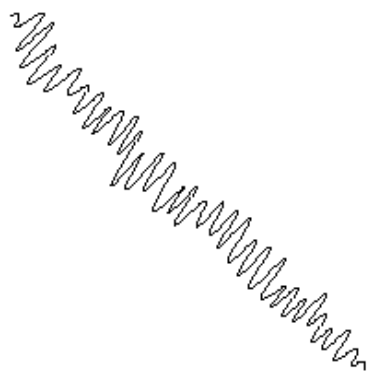
6 x detaily



finálna podoba

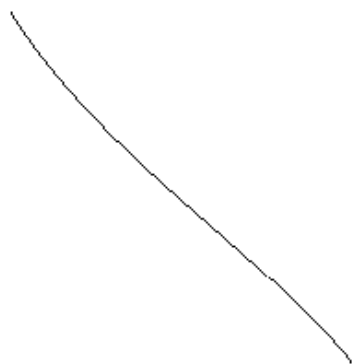
Manipulácie s krivkami

I) zmena chrakteru bez zmien detailu



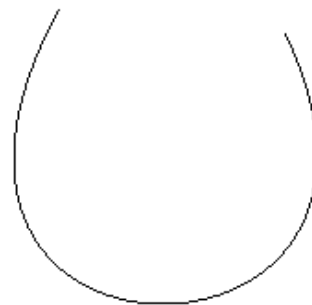
(a)

Originál



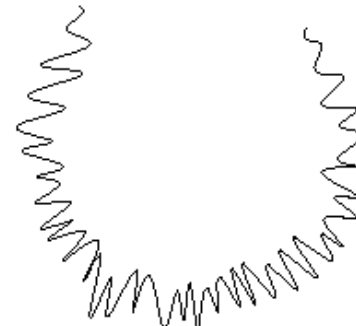
(b)

Extrahovaný
charakter



(c)

Zmena
charakteru



(d)

Spätné pridanie
detailov

II) zmiana detailu bez zmian charakteru



Využitie waveletov v počítačovej grafike:

Príklad modelovania objektov pomocou waveletovej transformácie

(modifikovaných 47 spektrálnych koeficientov)

