


Data Filtering in Prenatal Care Filtrace dat v prenatalní péči

DOLINAY, Viliam¹, LAMPART, Marek² & VAŠEK, Vladimír³

✉ Fakulta aplikované informatiky, UTB ve Zlíně, Nad Stráněmi 4511, Zlín, 760 05

1  vdolinay@fai.utb.cz

2  lampart@fai.utb.cz

3  vasek@fai.utb.cz

 <http://fai.utb.cz>

Abstrakt: Tento příspěvek popisuje metody zpracování signálu získaného při měření pomocí speciálního tělního pásu, který je určen k monitorování žen v průběhu rizikového těhotenství. Tento pás byl navržen a zkonstruován v rámci dřívějšího výzkumu realizovaným externí organizací. Úkolem pásu je monitorovat pohyb břicha těhotné ženy u nichž se předpokládají problémy během těhotenství. Měření je zaměřeno na získání informace o počtu kontrakcí a jejich periodě opakování během dne. Výstupními daty pásu je hodnota napnutí pásu, která je získána z tenzometru spojující konce pásu. Tato hrubá data před další analýzou musí být nejdříve filtrována a tomu se právě věnuje tento příspěvek. Pro tento úkol byly navrženy tři jednoduché filtrační metody. Po testech v reálné situaci vyplynulo, že nejvhodnější průběh získáme jejich vzájemnou kombinací. Zmiňované algoritmy jsou implementovány v testovacím programu, který rovněž zapouzdřuje schopnosti čtení z pásu, vizualizaci a základní analýzu zaměřenou na rozpoznání kontrakcí.

Klíčová slova: filtrace, šum, kontrakce, pás, těhotenství

1 Úkol

Pro monitorování žen s rizikovým těhotenství se využívá speciálního tělního pásu, jehož výstupem jsou hodnoty relativního tlaku měřeného s vysokou periodou. Aby bylo možno tato data vhodně zobrazit a dále analyzovat je žádoucí nejdříve provést odstranění nežádoucích složek, které v tomto případě vznikají převážně prudkými pohyby vyšetřované ženy, dýcháním apod.

2 Řešení

Byly navrženy tři jednoduché metody, které ze skupiny několika předešlých hodnot určují novou - takovou co nejméně odpovídá dosavadnímu průběhu. Všechny tři metody vycházejí s podobného algoritmu.

Jádrem navrženého algoritmu je vyhledávání těžiště trojúhelníka v rovině a porovnávání následujících hodnot s předem získaným těžištěm.

Označme nejprve D množinu vstupních dat. Tuto množinu můžeme chápat jako posloupnost

$$D := \{[t_i, v_i]\}_{i=1}^k,$$

kde t_i jsou časy, ve kterých dochází k naměření hodnot v_i . Celé měření probíhá v konečném časovém úseku, tedy definovaná posloupnost má k prvků. Navíc, měření probíhá v konstantních intervalech, tedy

$$\frac{t_{j+1} - t_j}{2} = c,$$

pro každé $j = 1, 2, 3, \dots, (k-1)$.

Nyní definujme algoritmy (zobrazení) T_1 , T_2 a T_3 , jejichž postupným aplikováním (kompozicí) získáme výsledný algoritmus.

2.1 Algoritmus T_1

Algoritmus T_1 je navržen tak, aby každé po sobě jdoucí trojici bodů z D přiřadil těžiště trojúhelníka tvořeného těmito vrcholy. Výstupem je nová sada dat D_1 , která má třetinový počet bodů vzhledem k původní množině. Tedy, T_1 je zobrazení z D do D_1 definované předpisem:

$$T_1 = T_e \circ T_r,$$

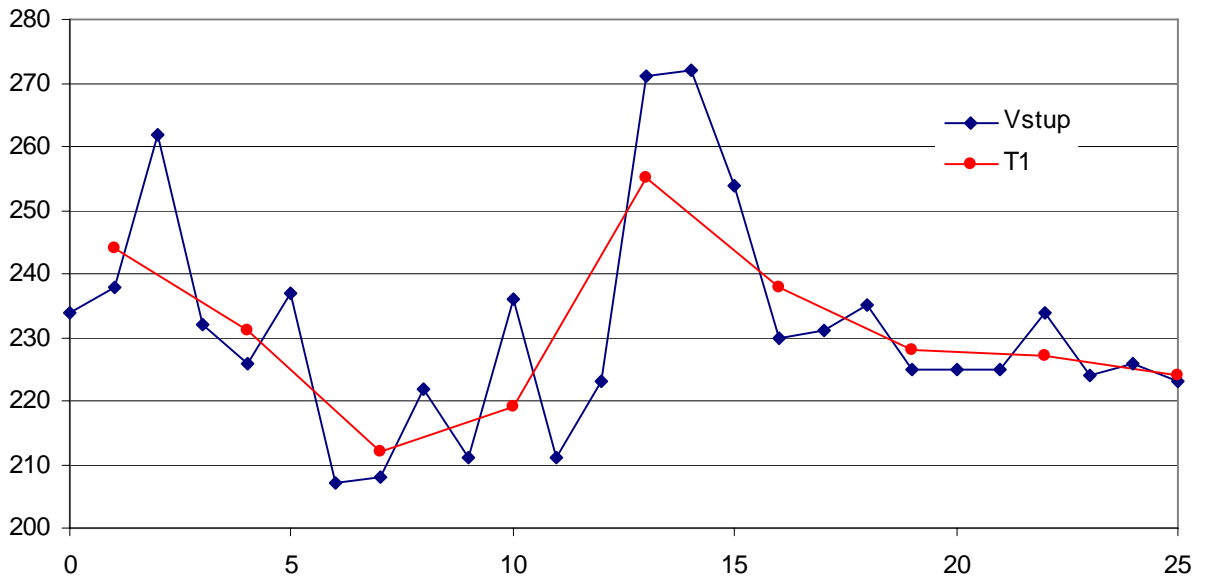
kde $T_r : D \rightarrow D^3$ je definované vztahem

$$T_r\left(\{[t_{i+j}, v_{i+j}]\}_{j=0}^2\right) = \{[t_{i+j}, v_{i+j}]\}_{j=0}^2,$$

kde $i = 1, 2, 3, \dots, \frac{k - \text{mod}_3(k)}{3}$; $\{[t_{i+j}, v_{i+j}]\}_{j=0}^2$ označuje uspořádanou trojici bodů $[t_i, v_i]$,

$[t_{i+1}, v_{i+1}]$, $[t_{i+2}, v_{i+2}]$, analogicky budeme označovat uspořádanou šestici; a $T_e : D^3 \rightarrow D_1$ je definované vztahem:

$$T_e\left(\{[t_{i+j}, v_{i+j}]\}_{j=0}^2\right) = \left[t_{i+1}, \frac{v_i + v_{i+1} + v_{i+2}}{3}\right].$$



Graf 1: Algoritmus T1

2.2 Algoritmus T_2

Algoritmus T_2 je navržen tak, že trojici bodů z D_1 (popř. D) přiřadí těžiště trojúhelníka s těmito vrcholy a poté nahlíží na následující dvě hodnoty. Jsou-li obě tyto hodnoty větší nebo menší než je hodnota předcházejícího těžiště, pak nahrazuje druhou z hodnot hodnotou těžiště. Je-li jedna z hodnot větší a druhá menší než hodnota těžiště, pak nahrazuje následující trojici novým těžištěm. Výstupem je nová sada dat D_2 . Tedy, T_2 je zobrazení z D_1 do D_2 definované předpisem:

$$T_1 = T_{es} \circ T_s$$

kde $T_s : D_1 \rightarrow D^6$ je definované vztahem

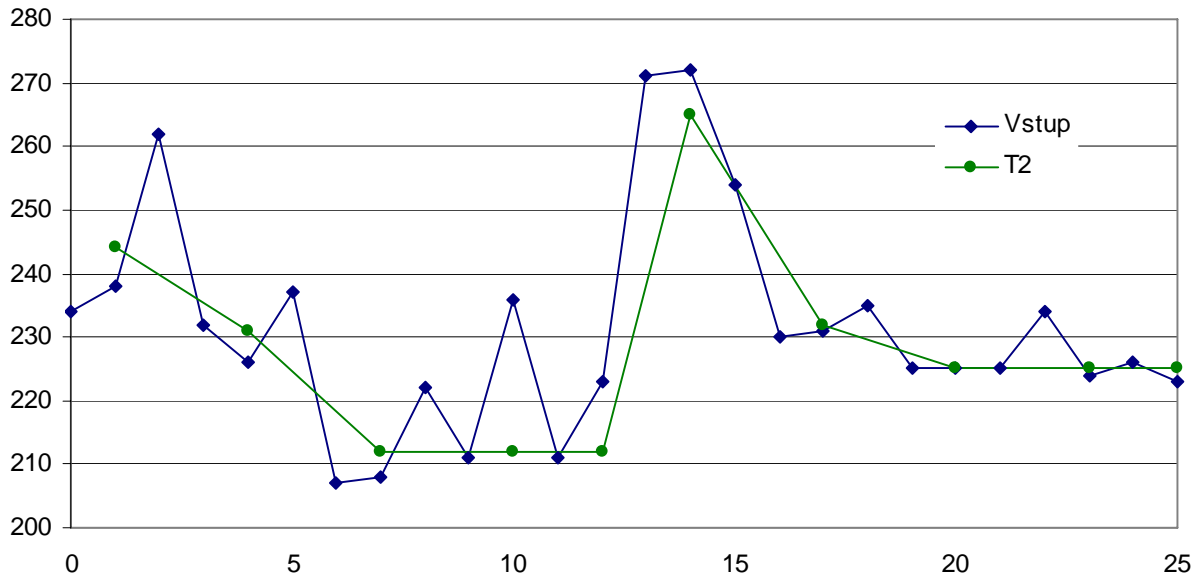
$$T_s \left(\left[\left[t_{i+j}, v_{i+j} \right]_{j=0}^5 \right) = \left[\left[t_{i+j}, v_{i+j} \right]_{j=0}^5 \right),$$

kde $i = 1, 2, 3, \dots, \frac{k - \text{mod}_6(k)}{3}$ a $T_{es} : D^6 \rightarrow D_2$ je definované vztahem

$$T_{es} \left(\left[\left[t_{i+j}, v_{i+j} \right]_{j=0}^5 \right) = \begin{cases} \left[\left[t_{i+1}, te_i \right], \left[t_{i+4}, te_i \right] \right] \text{ pokud je } v_{i+3}, v_{i+4} < te_i, \\ \text{nebo } v_{i+3}, v_{i+4} > te_i, \\ \left[\left[t_{i+1}, te_i \right], \left[t_{i+4}, te_{i+3} \right] \right] \text{ jinak,} \end{cases}$$

kde

$$te_i = \frac{v_i + v_{i+1} + v_{i+2}}{3}.$$



Graf 2: Algoritmus T2

2.3 Algoritmus T_3

Algoritmus T_3 je navržen tak, že trojici bodů z D_2 (popř. z D_1 či D) přiřadí těžiště trojúhelníka s těmito vrcholy a poté nahlíží na následující tři hodnoty. Jsou-li všechny tři tyto hodnoty větší nebo menší než hodnota předcházejícího těžiště, pak nahrazuje třetí z hodnot hodnotou těžiště. Pokud je jedna z hodnot větší (resp. menší) a zbylé menší (resp. větší) než hodnota těžiště, pak nahrazuje tuto trojici novým těžištěm. Výstupem je nová sada dat D_3 . Tedy, T_3 je zobrazení z D_2 do D_3 definované předpisem:

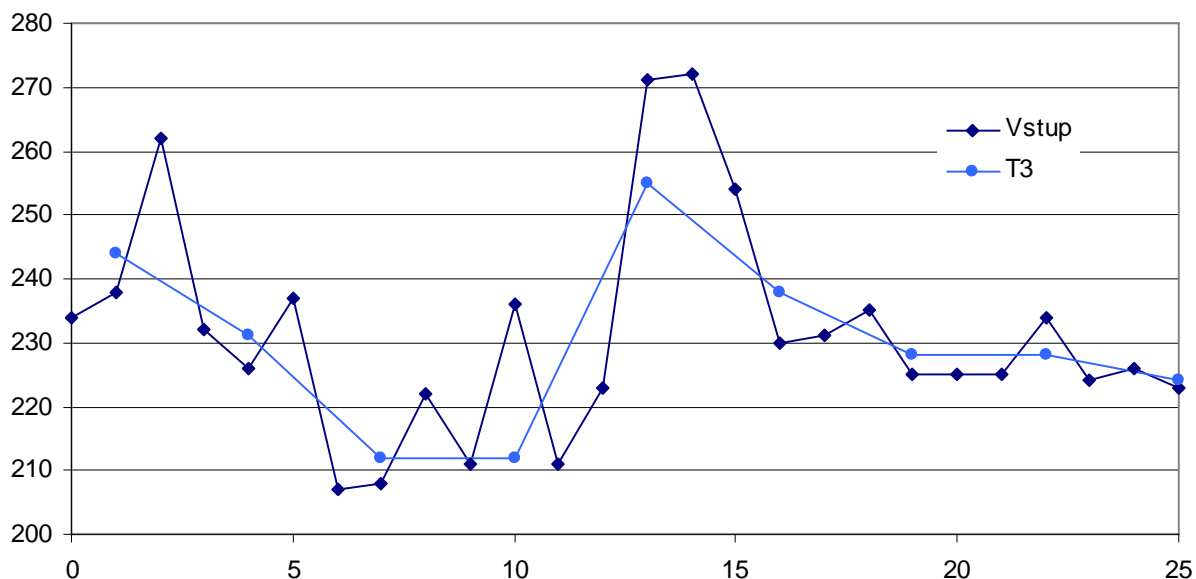
$$T_3 = T_{ee} \circ T_s,$$

kde $T_{ee} : D^6 \rightarrow D_3$ je definované vztahem

$$T_{ee} \left(\left[\left[t_{i+j}, v_{i+j} \right]_{j=0}^5 \right) = \begin{cases} [t_{i+1}, te_i], [t_{i+5}, te_i] & \text{pokud je } v_{i+3}, v_{i+4}, v_{i+5} < te_i, \\ & \text{nebo } v_{i+3}, v_{i+4}, v_{i+5} > te_i, \\ [t_{i+1}, te_i], [t_{i+4}, te_{i+3}] & \text{jinak,} \end{cases}$$

kde

$$te_i = \frac{v_i + v_{i+1} + v_{i+2}}{3}.$$



Graf 3: Algoritmus T3

3 Výsledný algoritmus

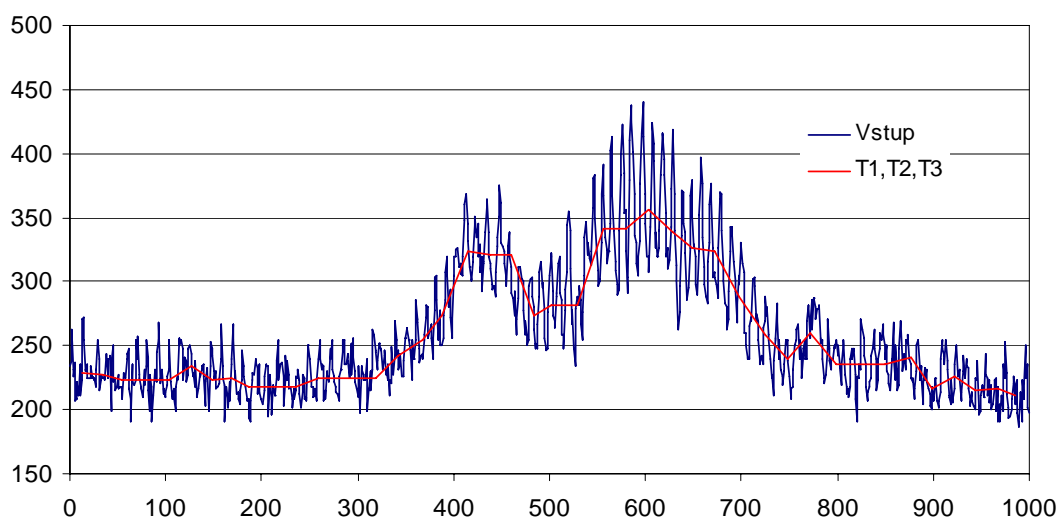
Výsledný algoritmus používaný k filtraci předem naměřených dat je kombinací předcházejících algoritmů T_1 , T_2 a T_3 , přesně v tomto pořadí. Označme ho A .

Tedy, A je definován jako zobrazení

$$A: D \rightarrow D_3,$$

vztahem

$$A: T_3 \circ T_2 \circ T_1.$$



Graf 4: Algoritmus A

Výstupního zobrazování se dosahuje primitivním (tj. lineárním) napojováním sousedních bodů z výsledné množiny dat.

4 Závěr

Navržený algoritmus dává překvapivě kvalitní výsledky. Nespornou výhodou tohoto algoritmu je jeho jednoduchost a nenáročnost jak po stránce programovatelnosti, tak po stránce náročnosti na hardwarové vybavení. Celý algoritmus, nebo i jeho části, se dají používat i při filtraci jiných dat z odlišných odvětví.

V poslední řadě je nutné vyzdvihnout, že při navrhování tohoto algoritmu byl kladen důraz na jeho možnost použití v reálném čase. Toho bylo dosaženo a je tedy možné využít daný algoritmus jako základ pro další vývoj mobilního hardwaru.

Výsledný graf je dosažen jako po částech lineární funkce, kterou by šlo vyhlazovat klasickými i nestandardními metodami, což je opět námětem pro další výzkum.

5 Použitá literatura

REKTORYS, Karel: *Přehled užití matematiky 1*. 7. vyd. - Praha : Prometheus, 2000. ISBN 80-7196-180-9

ADAMS, R. A., *Single Variable Calculus (paperback)* (Edition: 5) Pearson Education Canada, 2003. ISBN: 0-201-79805-0