Hybrid SORN Hardware Accelerator for Support Vector Machines

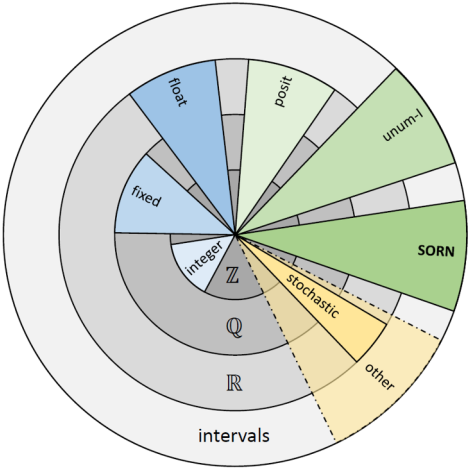
N. Hülsmeier[1] M. Bärthel[1] J. Rust[2] S. Paul[1]

[1]Institute of Electrodynamics and Microelectronics
University of Bremen

[2]DSI Aerospace Technologie GmbH

Conference for Next Generation Arithmetic
March, 2023

Contents



SORN Arithmetic

- Low resolution, interval based, dual number format [1]
- SORNs allow low complex and ultra fast computing with LUTs

Table: LUT for the multiplication of a simple 3 bit SORN data type

×		**0** 100	**(0, 0.5]** 010	**(0.5, 1]** 001
0	100	100	100	100
(0, 0.5]	010	100	010	010
(0.5, 1]	001	100	010	011

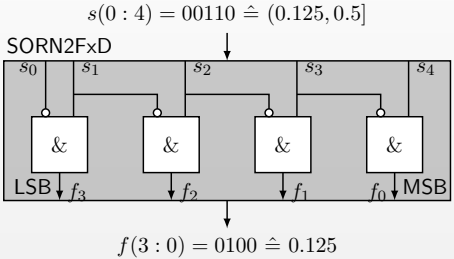
$$\mathcal{D} = \{0(0, 0.5](0.5, 1]\}$$

$$0.3 + 0.2 \hat{=} 010 + 010 = 011 \hat{=} (0, 1]$$
$$0.3 \cdot 0.2 \hat{=} 010 \cdot 010 = 010 \hat{=} (0, 0.5]$$

[1] J. Gustafson et. al, 2016.

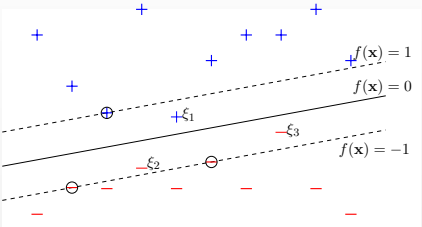
Hybrid SORNs

- Combination of SORN arithmetic and fixed point adders
- SORNs: low complexity, ultra fast
- FxD: higher precision
- SORN to FxD conversion for lower and/or upper interval bounds
- $\mathcal{D} = \{0(0, 0.125], (0.125, 0.25](0.25, 0.5](0.5, 1]\}$



Support Vector Machines

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$$



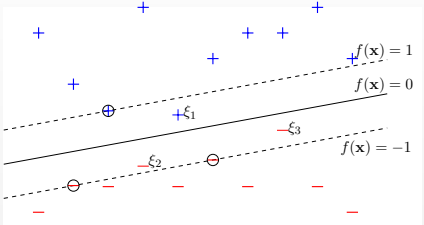
Original optimization problem

$$\min_{w,b,\xi} \frac{\|\mathbf{w}\|^2}{2} + C \sum_{i=1}^{N} \xi_i$$

subject to: $y_i \left(\mathbf{x}_i \cdot \mathbf{w} + b\right) \geq 1 - \xi_i, \ \xi_i \geq 0$

Support Vector Machines

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$$



Original optimization problem

$$\min_{w,b,\xi} \frac{\|\mathbf{w}\|^2}{2} + C \sum_{i=1}^{N} \xi_i$$

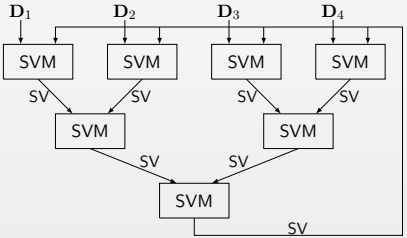
subject to: $y_i \left(\mathbf{x}_i \cdot \mathbf{w} + b\right) \geq 1 - \xi_i, \ \xi_i \geq 0$

Quadratic optimization problem

$$\max_{\mathbf{\Lambda}} -\frac{1}{2}\mathbf{\Lambda}\mathbf{D}\mathbf{\Lambda} + \mathbf{\Lambda}^T\mathbf{1}$$

subject to: $0 \leq \alpha_i \leq C, \ \mathbf{y}^T\mathbf{\Lambda} = 0$

Cascade SVM:



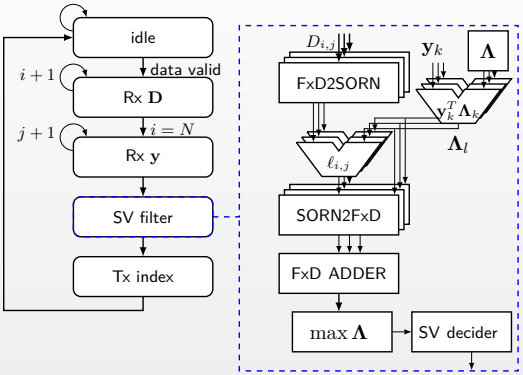
Hybrid SORN SV filter

- Idea: Support vector filtering with SORNs
- Restricted number of combinations in SORN representation allow fast solution of the optimization problem

To solve:

$$\max_{\mathbf{\Lambda}} -\frac{1}{2}\mathbf{\Lambda D \Lambda} + \mathbf{\Lambda}^T\mathbf{1}$$

subject to: $0 \leq \alpha_i \leq C,\ \mathbf{y}^T\mathbf{\Lambda} = 0$



Classification results

- Evaluation on MNIST dataset - 10 classes, 60,000 training images
- Multiclass SVM: one-versus-rest (ovr), one-versus-one (ovo)
- 8×8 subsets
- $|\mathcal{D}| = \{0\ (0, 0.25]\ (0.25, 0.5]\ (0.5, 0.75]\ (0.75, 1]\}$

Implementation	Float		**this work (Hybrid SORN)**	
Interface	CPU		FPGA + CPU	
Method	ovr	ovo	ovr	ovo
Classification error	**0.0208**	0.0326	0.0616	0.0411
Training time [s]	244.68	243.18	177.421	**91.736**

Synthesis results

- FPGA with Zynq-7000 XC7Z100 SoC
- Operating frequency: 200 MHz
- Data transfer via AXI DMA
- Floating point SVM is unable to filter non-SVs from 8×8 subsets

	FPGA Setup	Logic
Frequency	200 MHz	
Latency	$1.585\ \mu$s	$1.32\ \mu$s
Power	2.911 W	
LUT	165869	162047
FF	158224	152833
BRAM	2	0
DSP	0	0

Conclusion

- Hybrid SORN approach for SV filtering
- FPGA architecture at 200 MHz without DSPs
- Acceleration by 2.65 compared to CPU
- Classification error increased by 0.85%
- Acceleration can be expected for other HW accelerators

Future work:

- Evaluation for application specific datasets
- Distributed architecture

Thank you for your attention!