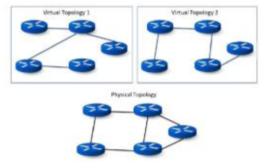
# Virtual Network Topology Control with Oja and APEX Learning

Y. Sinan Hanay, Yuki Koizumi, Shin'ichi Arakawa and Masayuki Murata
Graduate School of Information Sciences and Technology
Osaka University

## Virtual Network Topology (VNT)

- Find a topology optimizing
  - Minimizing the link with heaviest load
  - Minimizing maximum delay between 2 hops
  - Number of hops




## Outline

- Problem Statement
- System Model
- Results
- Conclusion

## VNT Controller

- Protein-gene mechanism [Furusawa08]

$$\frac{dx_i}{dt} = f\left(\sum_{j=1}^{n} w_{ij}x_j\right) V_g + (1 - V_g)x_i + \eta$$

- If the system is in
  - good conditions: deterministic behavior
  - Bad conditions: stochastic behavior
- Embed the topology information into a neural network

## Previous Models

| Previous Work | This Work |
|---|---|
| Builds a traffic matrix | Uses link load information |
| Not adaptive | Adaptive |
| Mostly offline | Online |

## Hebbian Learning

- Simple and the most widely used

$$\Delta w_{i,j} = \rho x_i x_j$$

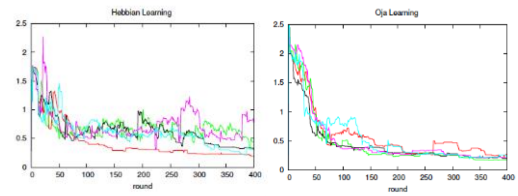
- Low computation times

# Oja Learning

- Weight matrix is similar to Hebbian
- $\Delta w_i = \alpha(x_i y_i - y^2 w_i)$
- Similar computation times as Hebbian

# Oja vs Hebbian Learning

- Oja shows less variance in terms of variance



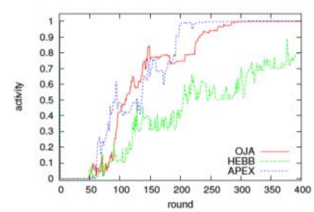

# Adaptive Principal Component Extraction (APEX)

- Additional weight matrix: Lateral weight matrix

$$y = Wx + Py$$

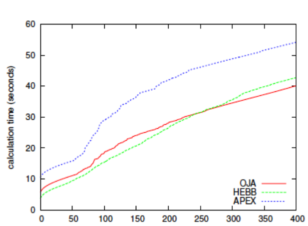
- $\Delta p_{i,j} = \alpha(y_i y_j - p_{i,j} y_i^2)$
- Converges quicker than Oja [Kung90]

# Hebb vs APEX and Oja



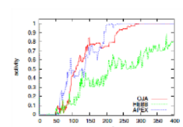

APEX and Oja performs better than Hebbian

# The Calculation Times

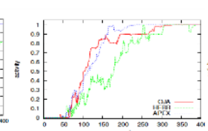



Oja and Hebbian shows similar calculation times

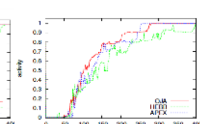
# The Number of Attractors



5 attractors



10 attractors

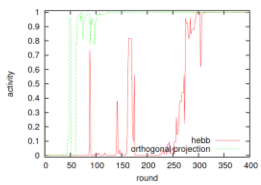


20 attractors

As the number of attractors increases, the performance gain reduces

# Orthogonal Projection

- $X^+$ Pseudo inverse
- $W_o = X^+X.\ X^+$
- 100X slower than the Hebbian




# Conclusion

- $X^+$, APEX and Orthogonal projection performs better than Hebbian
- Orthogonal projection is 100X slower
- Oja is best when the number of attractors are less than 20
- APEX is best for other cases

# References


- [Furusawa08] C. Furusawa and K. Kaneko. A Generic Mechanism for Adaptive Growth Rate Regulation. PLoS Computational Biology, 4(1), 2008.
- [Kung90] S. Kung and K. Diamantaras. A Neural Network Learning Algorithm for Adaptive Principal Component Extraction (apex), in Acoustics, Speech and Signal Processing, 1990.