

SC Hybrid Systems: Neural Networks Generation and Tuning

NN Controlled by FL Controllers

- NNs typical update rule:

$$\Delta W_n = -\eta \nabla E(W_n) + \alpha \Delta W_{(n-1)}$$

- NN parameters:

- Learning Rate: η
- Momentum: α
- Steepness: β

- Approaches:

- Jacobs 88: delta-bar-delta rule to change step size (and therefore learning rate) by simple heuristic
- Arabshahi 92: FLC to control learning rate:

$$E, \Delta E \rightarrow \Delta \eta$$

NN Fuzzy Accelerator

$$E, \Delta E \rightarrow \Delta\eta$$

$$E, \Delta E \rightarrow \Delta\alpha$$

- **E = Training Error**
- **ΔE = Change in Training Error**
- **$\Delta\eta$ = Change in Learning Rate**
- **$\Delta\alpha$ = Change in Momentum**

Change in Error	Training Error		
	<i>Small</i>	<i>Medium</i>	<i>Big</i>
<i>Negative</i>	Very Small Increase	Very Small Increase	Small Increase
<i>Zero</i>	No Change	No Change	Small Increase
<i>Positive</i>	Small Decrease	Medium Decrease	Large Decrease

NN Fuzzy Accelerator

$$E, T \rightarrow \beta$$

- **E = Training Error**
- **T = Training Time**
- **β = Steepness of Sigmoid** [$f(x) = 1/(1+e^{-(1/2)\beta x})$]

Training Time	Training Error		
	<i>Small</i>	<i>Medium</i>	<i>Big</i>
<i>Short</i>	Medium	Small	Small
<i>Medium</i>	Large	Medium	Small
<i>Long</i>	Large	Large	Medium

NNs generated by GAs

- GAs: global search, large granule
- NNs (with BP): local search, small granule
- Approaches:
 - **Simultaneous NN topology and weight vector W**
 - Direct encoding (fixed resolution) -> chromosome too long
 - Maniezzo 93: Direct encoding with variable resolution: resolution parameter, encoded in chromosome, determines number of bits for weights
 - **Find weight vector W for given topology:**
 - Mountana & Davis 89: first attempt
 - Kitano 90: used GAs to obtain good W parameter region + BP to do final tuning
 - McInerney 92: BP to find local minimum + GAs to find global optimum guided by location of local minima

NN Generation & Tuning Conclusions

- **Knowledge-driven Search Control**
 - Use local data search to derive models (Structure + Parameters)
 - Translate domain knowledge into an algorithm's controller to improve/manage solution convergence and quality
- **Benefits**
 - Limit size of required training set
 - Shorten training time
 - Improve solution quality
- **Costs**
 - Additional computational overhead in each epoch
 - Possible Fix: Compilation of FLC Knowledge bases
 - Potential downside: poorly written FLC rules could cause instability (unlikely)
 - Unlikely if starting from standard Fuzzy PI structure
 - Could use default tables (see slides) or use an external GA to tune the FLC KB.