Handling of Time

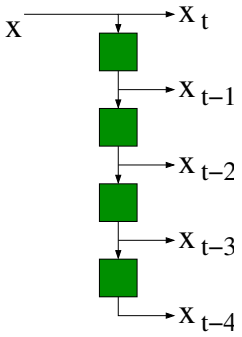
Central Pattern Generator

Neuronal circuit which produces rhythmic output

- Locomotion (walking, swimming, flying)
- Breathing
- Scratching

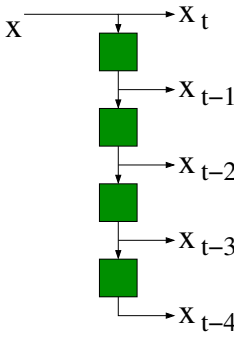
- Mutual inhibition
- Pacemaker cell properties

- Temporal signals are tricky
- Transform temporal signals to spatial
- Tapped Delay Line

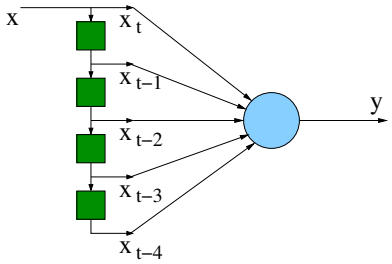


Design choices

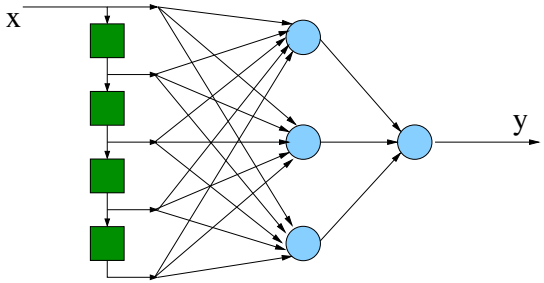
- Memory depth
  Number of time-steps saved
- Resolution
  Size of time-steps



Simplest form: "Neuronal filter"

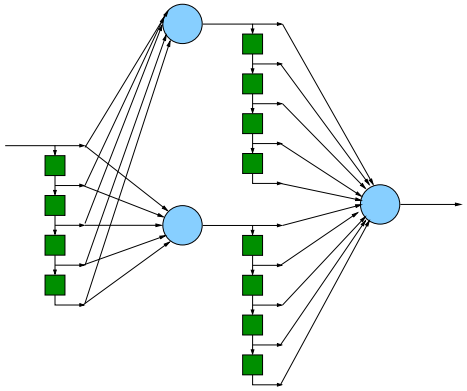


Multi-Layer Network



Delay of all signals

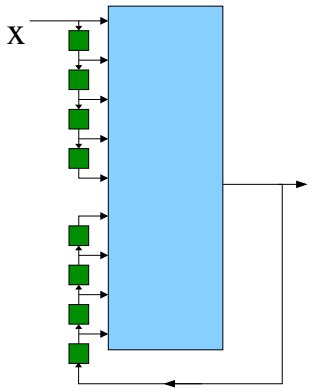
- More general
- Harder to train



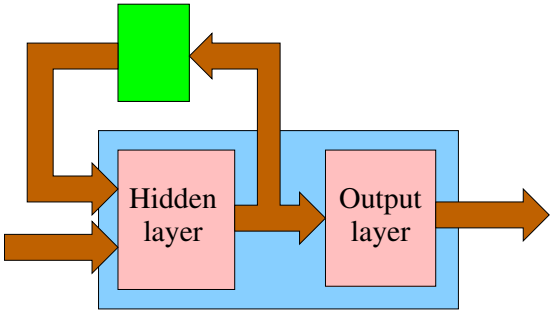
How can we train a network with delays?

- Ordinary Back-Prop?
- Works when only the input is delayed
- Generalizations are needed when general delays are included
- The error signal must be matched against old activity values

Feedback of the output

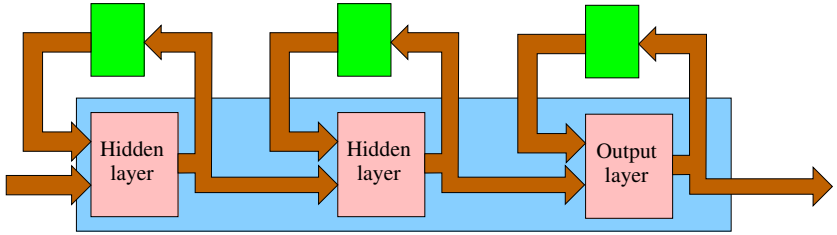


Feedback from the hidden layer in next time-step



Context Units

Multi-Layered Recurrent Network



Second Order Networks

- Product of input signal and feedback
- Every pair: <in-signal $\times$ feedback> has its own weight
- Many (specific) weights

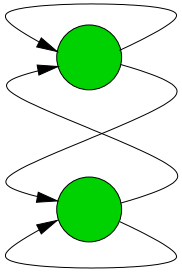
What can a recurrent network do?

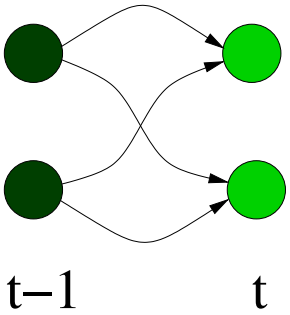
- Finite-State Machine
- State corresponds to activity in the context nodes
- Combination of state and input produces next state

Is it possible to train a recurrent network to reproduce given sequences?

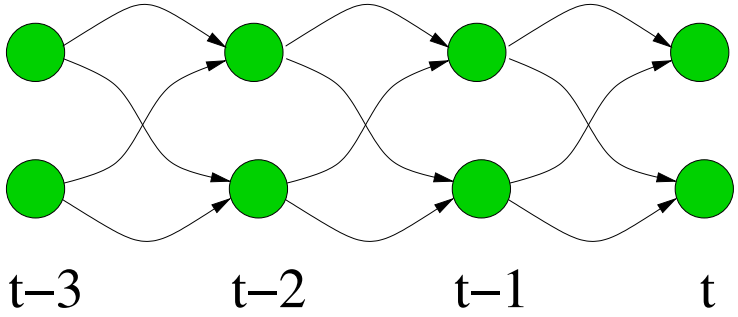
- Back-Propagation Through Time

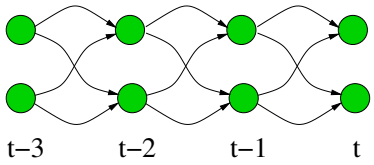
- Recurrent network with delayed connections
- Current state depends (only) on the previous state



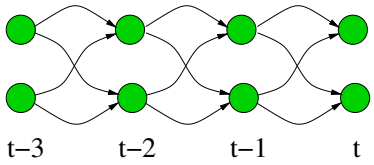


Unfolding of the History





- Back-Propagation Through Time
- Target values are spread out over different layers
- Add contributions from all goals
- The same weight occurs in multiple places!
- Add contribution from all places



- All old activity values must be stored
- Problematic when learning long sequences
- Truncation of the history