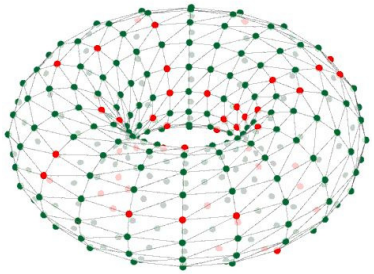
# Adding new models of synaptic plasticity



Jamie Knight

SpiNNaker Workshop
September 2016

European Research Council
Established by the European Commission

Human Brain Project

EPSRC

SpiNNaker

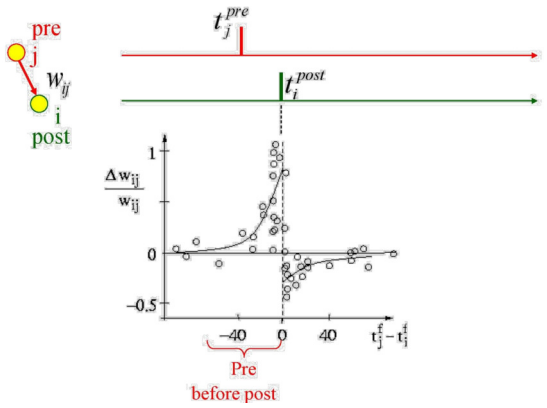
# Outline

- Introduction to spike-timing dependent plasticity
- Simulating STDP
- Limitations of pair-based STDP
- Triplet STDP
- SpiNNaker implementation



# Introduction to spike-timing dependent plasticity

"Cells that fire together, wire together"






http://www.scholarpedia.org/article/Spike-timing_dependent_plasticity

# Simulating STDP - Traces

Pre-synaptic trace

$$\frac{dx_j}{dt} = -\frac{x_j}{\tau_x} + \sum_{t_j^f} \delta(t - t_j^f)$$

Post-synaptic trace

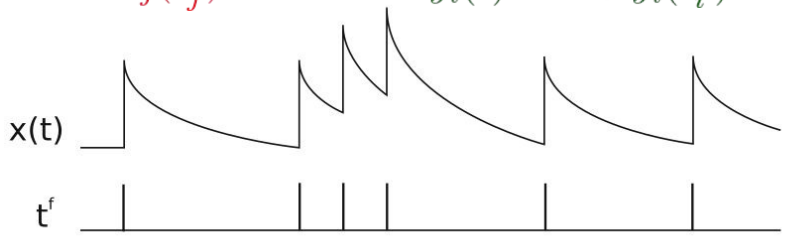
$$\frac{dy_i}{dt} = -\frac{y_i}{\tau_y} + \sum_{t_i^f} \delta(t - t_i^f)$$

At pre-synaptic spike time

$$x_j(t) = 1 + x_j(t_j^f) e^{-\frac{t - t_j^f}{\tau_x}}$$

At post-synaptic spike time

$$y_i(t) = 1 + y_i(t_i^f) e^{-\frac{t - t_i^f}{\tau_y}}$$






Morrison, A., Diesmann, M., & Gerstner, W. (2008). Phenomenological models of synaptic plasticity based on spike timing. Biological Cybernetics, 98, 459–478.

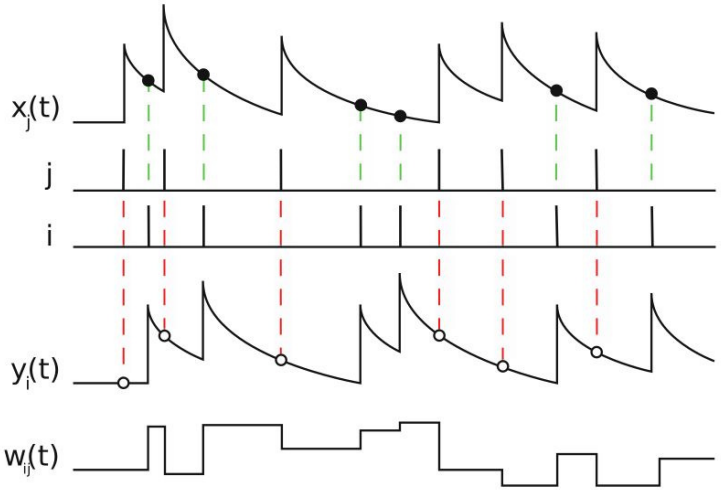
# Simulating STDP - Weight update

Pre-synaptic weight update

$$\Delta w_{ij}^{-} = F_{-}(w_{ij}) y_i(t_j^f)$$

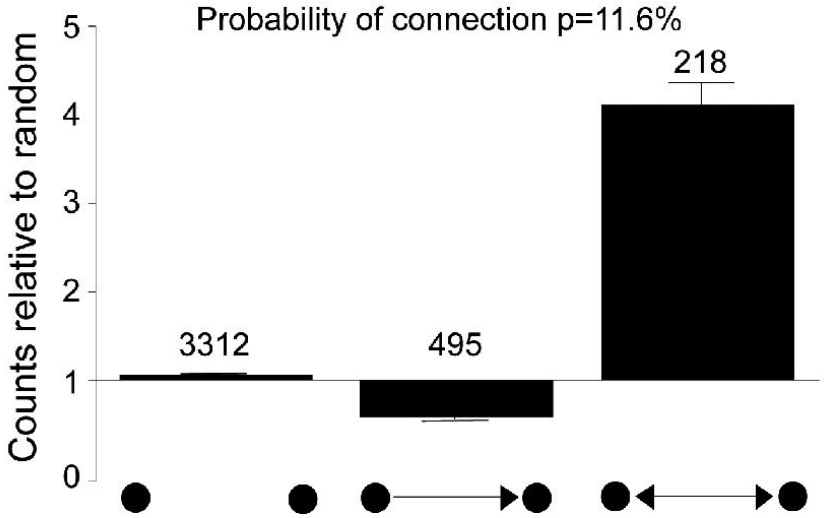
Post-synaptic weight update

$$\Delta w_{ij}^{+} = F_{+}(w_{ij}) x_j(t_i^f)$$





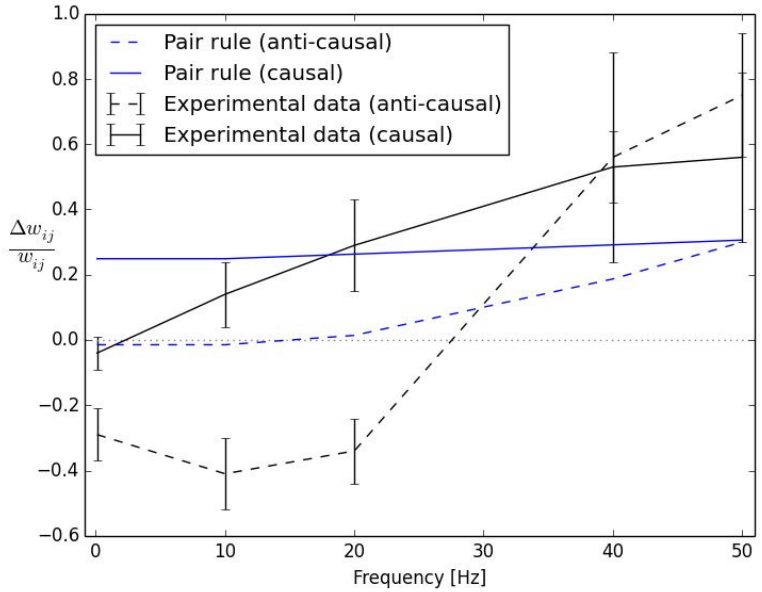

# Limitations of pair-based STDP






Song, S., Sjöström, P. J., Reigl, M., Nelson, S., & Chklovskii, D. B. (2005). Highly nonrandom features of synaptic connectivity in local cortical circuits. PLoS Biology, 3(3), 0507–0519.

# Limitations of pair-based STDP






Sjöström, P. J., Turrigiano, G. G., & Nelson, S. B. (2001). Rate, timing, and cooperativity jointly determine cortical synaptic plasticity. *Neuron*, *32*(6), 1149–64.

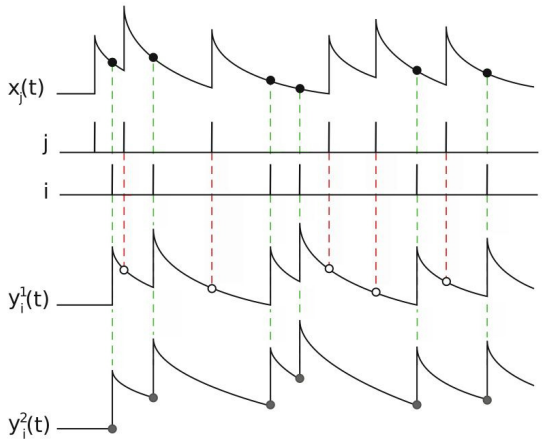
# Triplet STDP

Slow post-synaptic trace

$$y_i^2(t) = \left(1 + y_i^2(t_i^f)\right) e^{-\frac{t - t_i^f}{\tau_y^2}}$$

Post-synaptic weight update

$$\Delta w_{ij}^{+} = F_{+}(w_{ij}) x_j(t_i^f) y_i^2(t_i^{f-})$$






Pfister, J. P., & Gerstner, W. (2006). Triplets of spikes in a model of spike timing-dependent plasticity. The Journal of Neuroscience : The Official Journal of the Society for Neuroscience, 26(38), 9673–82.

# SpiNNaker - Pair traces

timing_pair_impl.h
line 7

```
typedef int16_t post_trace_t;
```

timing_pair_impl.h
lines 46-49

```
static inline post_trace_t timing_get_initial_post_trace()
{
  return 0;
}
```



# SpiNNaker - Triplet traces

timing_triplet_impl.h
line 7

```
typedef struct post_trace_t
{
  int16_t y1;
  int16_t y2;
} post_trace_t;
```

timing_triplet_impl.h
lines 46-49

```
static inline post_trace_t timing_get_initial_post_trace()
{
  return (post_trace_t){.y1 = 0, .y2 = 0};
}
```



# SpiNNaker - Pair trace update

timing_pair_impl.h
lines 54-66

$$y_i(t) = 1 + y_i(t_i^f)e^{-\frac{t-t_i^f}{\tau_y}}$$

```
// Get time since last spike
uint32_t delta_time = time - last_time;

// Decay previous trace (y)
int32_t new_y = STDP_FIXED_MUL_16X16(last_trace,
  DECAY_TAU_Y(delta_time));

// Add energy caused by new spike to trace
new_y += STDP_FIXED_POINT_ONE;

log_debug("\tdelta_time=%d, y=%d\n", delta_time, new_y);

// Return new trace_value
return (post_trace_t)new_y;
```



# SpiNNaker - Triplet trace update

timing_triplet_impl.h
lines 77-87

$$y_i^2(t) = \left(1 + y_i^2(t_i^f)\right)e^{-\frac{t-t_i^f}{\tau_y^2}}$$

```
// Y2 is sampled in timing_apply_post_spike BEFORE the spike
// Therefore, if this is the first spike, y2 must be zero
int32_t new_y2;
if(last_time == 0)
{
  new_y2 = 0;
}
// Otherwise, add energy of spike to last value and decay
else
{
  new_y2 = STDP_FIXED_MUL_16X16(
    last_trace.y2 + STDP_FIXED_POINT_ONE,
    DECAY_TAU_Y2(delta_time));
}
```

# SpiNNaker - Pair weight update

timing_pair_impl.h
lines 136-150

$$\Delta w_{ij}^{+} = F_{+}(w_{ij}) x_j(t_i^f)$$

```
uint32_t delta_t = time - last_pre_time;

// If spikes are not co-incident
if (delta_t > 0)
{
  // Calculate x(time) = x(last_pre_time) * e^(-delta_t/tau_x)
  int32_t x = STDP_FIXED_MUL_16X16(last_pre_trace,
    DECAY_TAU_X(delta_t));

  log_debug("\t\t\tdelta_t=%u, x=%d\n",
    delta_t, x);

  // Apply potentiation to synapse state
  return weight_one_term_apply_potentiation(previous_state, x);
}
```



# SpiNNaker - Triplet weight update

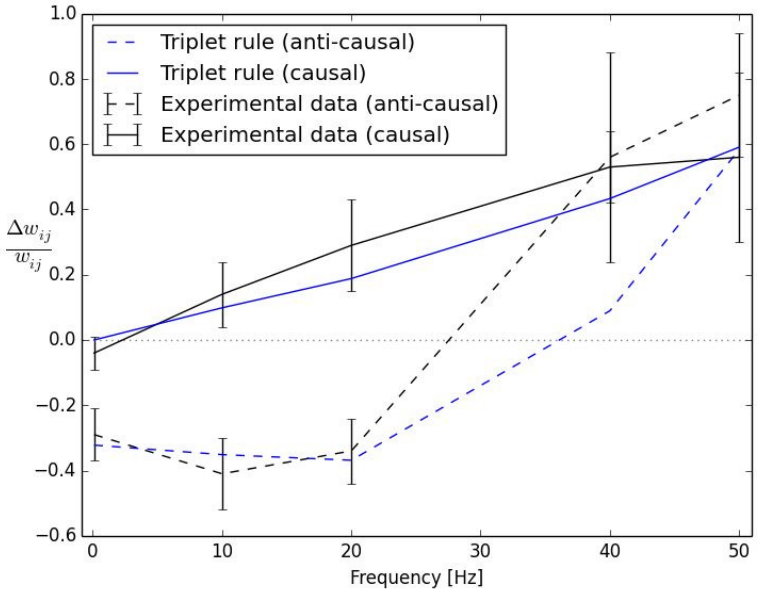
timing_triplet_impl.h
lines 165-179

$$\Delta w_{ij}^{+} = F_{+}(w_{ij}) x_j(t_i^f) y_i^2(t_i^{f-})$$

```
if (delta_t > 0)
{
  // Calculate x(time) = x(last_pre_time) * e^(-delta_t/tau_x)
  int32_t x = STDP_FIXED_MUL_16X16(last_pre_trace,
    DECAY_TAU_X(delta_t));

  // Multiply this by y2(time) to get triplet term
  int32_t x_y2 = STDP_FIXED_MUL_16X16(x, trace.y2);

  log_debug("\t\t\tdelta_t=%u, x=%d, y2=%d, x_y2=%d\n",
    delta_t, x, trace.y2, x_y2);

  // Apply potentiation to synapse state
  return weight_one_term_apply_potentiation(previous_state, x_y2);
}
```



# Triplet model






# Thank you!

## Any questions?

james.knight@manchester.ac.uk