Additional CMAC Material BILHR

Algorithm for getting the positions of the L2 neurons of the CMAC:

1) Create the receptive field (imagine it as a 2D grid with neurons at particular coordinates):

e.g. receptive field size $n_a = 5$:
set the neuron positions to:
(0; 3)
(1; 0)
(2; 2)
(3; 4)
(4; 1)

2) Compute the position of the active neurons of L2 depending on the input:

In the following, the receptive field size n_a is termed as *field_size*.

Position of the neurons within the receptive field: *RFpos(r, c)* with *r* as neuron index and *c* as input channel index. You have created the *RFpos(r, c)* in step 1).

float variables (the **input value** for each y1 and y2 has to be **normalized**): *input[]*

integer variables: *resolution*, *field_size*, *input_index*, *shift_amount*, *local_coord*, *coord*

Position of the active neurons of L2: *Position*

Initialize an empty position vector *Position*;
for every neuron *r* of the receptive field, do:
    Initialize an empty coordinate vector *Coord*;
    for every channel *c* of the input channels, do:
        input_index = round(*input[c]* * *resolution*); // use the C++ function *round()*
        shift_amount = field_size – input_index % *field_size*;
        local_coord = (*shift_amount* + *RFpos(r, c)*) % *field_size*;
        coord = input_index + *local_coord*;
        store *coord* into *Coord*;
    end for
    store *Coord* into *Position*;
end for