```
% Generates the probability of a detected state (Y,X,W,Z) using the theory
% designed by Kieran.  The required state, mode occupancy and relative phase
% between the two branches define this probability.
%% Function
function c = cyxwz(y,w,x,z,lambda,phi,phys,Y_bias,X_bias)
    % phi minus ignored, phi = phim + phip
    T = y + w;
    % T = total number of pairs.
    if phys =='Q'
        % Term outside the sums.
        external = (1-lambda.^2).*(lambda./2).^T*sqrt(factorial(y)*factorial(z)↙
*factorial(w)*factorial(x));
        psi = 0;
        % Sum through k:
        for k = 0:T
            [l,n] = meshgrid(max(k-y,0):min(k,w),max(k-x,0):min(k,z));
            comp_phase = 1i.^(w+z+2.*k-2.*l-2.*n).*exp(-1i.*(phi).*k);
            chooses = bincof(T-k,w-l).*(bincof(T-k,z-n)).*bincof(k,l).*bincof(k,n);
            epsilon = comp_phase.*chooses;
            psi = psi + sum(sum(epsilon))./(factorial(T-k).*factorial(k));
        end
        % Return the norm square of the coefficient c, to give the probability
        % of the state:
        c = abs(external.*psi).^2;
    else
        % Classical probability calculation.
        p = (lambda^(2*T))*(1-lambda^2)^2 * (T+1)*(Y_bias^y)*(1-Y_bias)^(T-y)*(X_bias^x)↙
*(1-X_bias)^(T-x)*bincof(T,y)*bincof(T,x);
        c = p;
    end
end
```