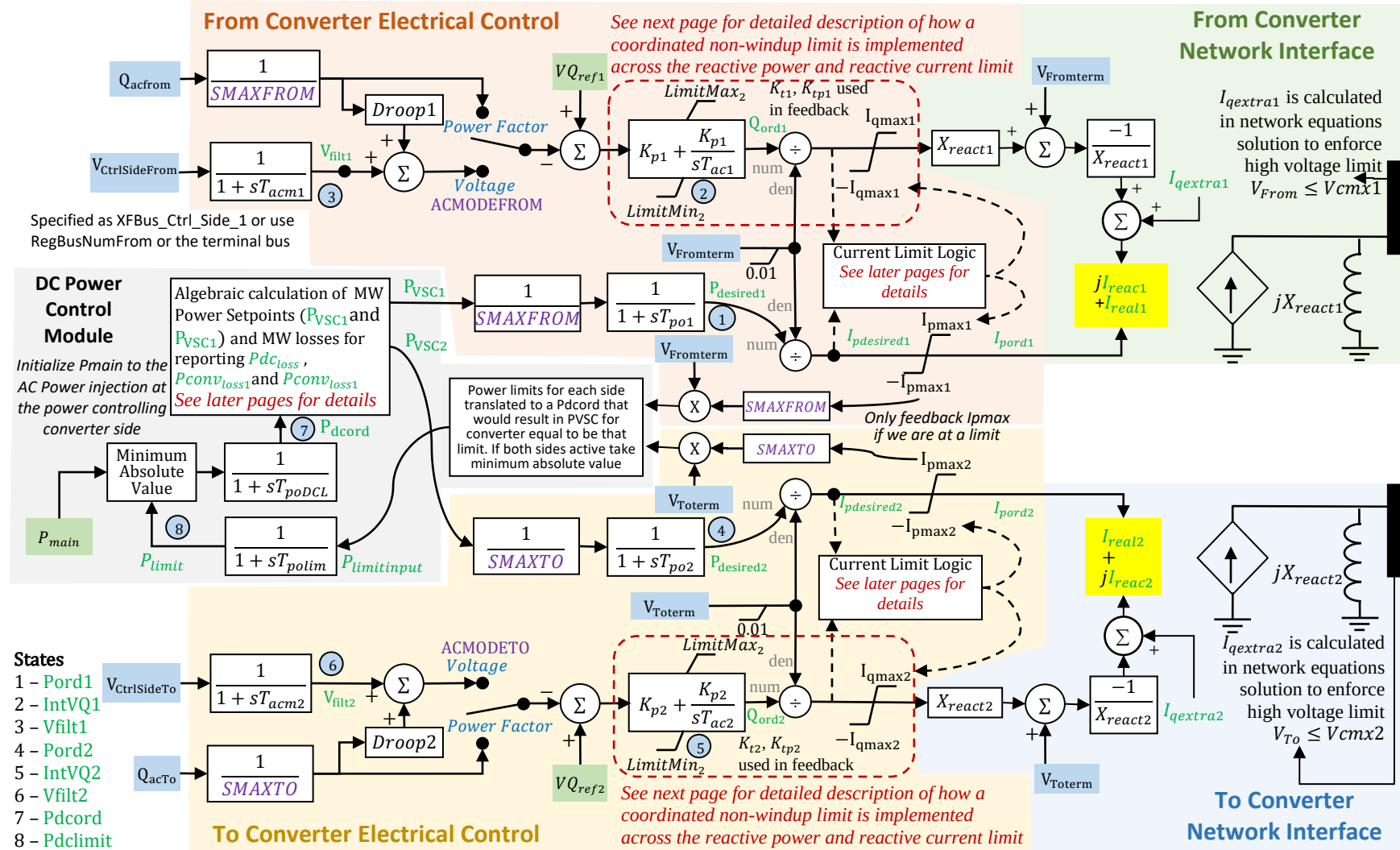


VSC DC Line VSCDCT

VSC DC Line Generic Model VSCDCT

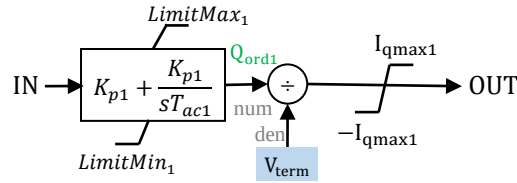


Other values in green text are available for output reporting: P_{dcloss} , $P_{convloss1}$, $P_{convloss2}$, $P_{limitinput}$, P_{VSC1} , Q_{ord1} , $Q_{desired1}$, $I_{qdesired1}$, I_{qord1} , $I_{pdesired1}$, I_{pord1} , $I_{qextra1}$, I_{react1} , I_{real1} , P_{VSC2} , Q_{ord2} , $Q_{desired2}$, $I_{qdesired2}$, I_{qord2} , $I_{pdesired2}$, I_{pord2} , $I_{qextra2}$, I_{react2} , I_{real2}

Following input values are from power flow input data: $SMAXFROM$, $PWFFFROM$, $ACMODEFROM$, $DCMODEFROM$, $SMAXTO$, $PWFTO$, $ACMODETO$, $DCMODETO$

Implementation of Reactive Power and Current non-windup limits

Both the From and To Converter Electrical Control algorithms have reactive power control with Proportional-Integral control with a non-windup limit followed by a reactive current limit.



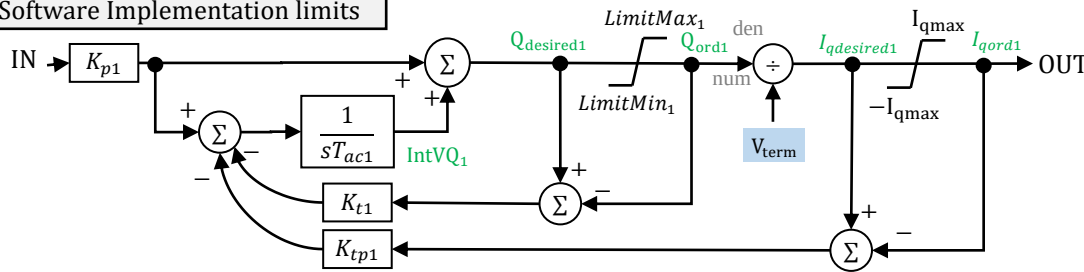
At the From end converter LimitMax1 and LimitMin1 are determined from the input parameters VClm1 and Qmax1 and Qmin1 using the following logic.

If VCLimit1 > 0 then LimitMax1 = VCLimit1 and LimitMin1 = -VCLimit1
 Else LimitMax1 = Qmax1/SMAXFROM and LimitMin1 = Qmin1/SMAXFROM

Where SMAXFROM is from the power flow object.

This PI Controller is implemented in a manner that allows both the reactive power limits and the reactive current limits to act as non-windup limits on the PI control. These utilize the additional feedback gains of K_{t1} and K_{tp1} from the input parameters. The picture shows how this coordinated non-windup limit is implemented in the software tool. The same structure is used for the To end converter.

Software Implementation limits



DC Power Control Module

The DC power control module operates to ensure that the power setpoints remain consistent between the From and To end converters on the DC lines. In the power flow the power setpoint will be maintained at the end marked as DCModeFrom/To = Power and the other end of the VSC DC Line has a power calculated based on the setpoint minus the calculated losses. This model will attempt to maintain the AC power injection at that DC power controlling converter and the value of DC Main will be initialized to be equal to this value. The same algebraic equations used in the power flow solution will be used to calculate setpoints PVSC1 and PVSC2 to pass to the From and To end voltage source converter to represent AC powers at those terminals.

Each VSC control has current limits. When a VSC hits a real current limit it must communicate back to the other side's converter. To achieve this, when at a real current limit that VSC will pass back the presently active limit at that converter in MWs ($MW\ Limit = S_{max} V_{term} I_{pmax}$). If this is at the DC power controlling converter the value is used directly, otherwise we take that value and translate it to a setpoint at the DC power controlling converter that would result in this MW flow at the converter at a new steady state. This conversion is done by solving the equations for VSC setpoints when the power and DC voltage are both specified at the same end of the VSC DC Line using equations shown shortly. In the unlikely situation that both converters are at real current limit simultaneously, then the minimum absolute value of the two is used.

In a real control system, the coordination of power setpoints would be handled by the DC voltage and DC Current controls systems on the VSC. This model assumes those controls act very quickly and thus does not model this detail.

Calculation of the VSC setpoint at the from and to end and MW losses

In the power flow solution there is an option to specified whether the DC MW power setpoint is on the AC side or DC side of the converter. For the VSCDT model we will always use setpoints on the AC side of the converter however because internally this model will be represented by two interfaces at the AC bus directly. The initial condition of the power flow has the injections into the AC power system from the VSC DC line of P_{acfrom} and P_{acto} in MW. For the purposes of our controllers this means that the initial condition values of P_{vsc1} and P_{vsc2} are equal to the values P_{acfrom} and P_{acto} . If DC Power Control is at the From Bus Converter, then the control will initialize $P_{main} = P_{acfrom}$, otherwise $P_{main} = P_{acto}$. Also the initial value of state 8 will be set to P_{main} which ensures that $P_{dcord} = P_{main}$ for the initial condition.

For the remainder of the simulation the value of P_{dcord} will be fed into the block for performing algebraic calculations of P_{VSC1} and P_{VSC2} . The assumption is made in these calculations that the desired DC voltage can always be met and that the P_{dcord} is the new AC power setpoint. Algebraic Equations are solved to

If the power setpoint is at the From side converter and voltage setpoint on the To Side, then the following equations are used.

$$P_{dcord} = P_{acfrom} = -P_{dcfrom} - MWLossFrom$$

$$P_{dcord} = -P_{dcto} - R_{dc}I_{dc}^2 - \frac{ALossFrom}{1000} - BLossFrom|I_{dc}|$$

If $P_{dcord} > 0$ then $I_{dc} < 0$ so we know $|I_{dc}| = -I_{dc}$. Thus

If $P_{dcord} > 0$ then $B_{used} = -BLossFrom$

Else $B_{used} = +BLossFrom$

$$P_{dcord} = -V_{dcto}I_{dc} - R_{dc}I_{dc}^2 - \frac{ALossFrom}{1000} - B_{used}I_{dc}$$

$$P_{dcord} = -V_{SetTo}I_{dc} - R_{dc}I_{dc}^2 - \frac{ALossFrom}{1000} - B_{used}I_{dc}$$

$$0 = P_{dcord} + \frac{ALossFrom}{1000} + V_{SetTo}I_{dc} + B_{used}I_{dc} + R_{dc}I_{dc}^2$$

$$0 = \left(P_{dcord} + \frac{ALossFrom}{1000}\right) + (V_{SetTo} + B_{used})I_{dc} + R_{dc}I_{dc}^2$$

Use the + sign here to get the low current / high voltage solution.

$$I_{dc} = \frac{-(V_{SetTo} + B_{used}) + \sqrt{(V_{SetTo} + B_{used})^2 - 4R_{dc}\left(P_{dcord} + \frac{ALossFrom}{1000}\right)}}{2R_{dc}}$$

Check if the losses in this calculation are too small.

If $\left(\frac{ALossFrom}{1000} - BLossFrom|I_{dc}|\right) < \left(\frac{MinLossFrom}{1000}\right)$ then resolve using

$$0 = \left(P_{dcord} + \frac{MinLossFrom}{1000}\right) + (V_{SetTo})I_{dc} + R_{dc}I_{dc}^2$$

$$I_{dc} = \frac{-(V_{SetTo}) + \sqrt{(V_{SetTo})^2 - 4R_{dc}\left(P_{dcord} + \frac{MinLossFrom}{1000}\right)}}{2R_{dc}}$$

If $R_{dc} = 0$ then adjust previous equations using $P_{dcfrom} = P_{dcto} = V_{SetTo}I_{dc}$ thus we can instead write

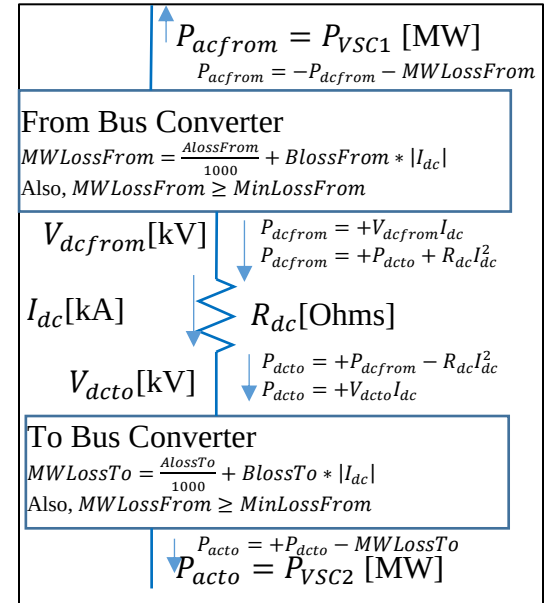
$$P_{dcord} = -V_{SetTo}I_{dc} - \frac{ALossFrom}{1000} - B_{used}I_{dc}$$

$$P_{dcord} + \frac{A}{1000} = -(V_{SetTo} + B_{used})I_{dc}$$

$$I_{dc} = -\left(\frac{P_{dcord} + \frac{ALossFrom}{1000}}{V_{SetTo} + B}\right)$$

Again, If $\left(\frac{ALossFrom}{1000} - BLossFrom|I_{dc}|\right) < MinLossFrom$ then resolve using

$$I_{dc} = -\left(\frac{P_{dcord} + \frac{MinLossFrom}{1000}}{V_{SetTo}}\right)$$



Similar logic is used if the power setpoint is instead at the To side converter and the Voltage setpoint is at the from side.

$$P_{dcord} = P_{acto} = P_{dcto} - MWLossTo$$

$$P_{dcord} = P_{dcfrom} - R_{dc}I_{dc}^2 - \frac{ALossTo}{1000} - BLossTo|I_{dc}|$$

If $P_{dcord} < 0$ then $I_{dc} < 0$ so we know $|I_{dc}| = -I_{dc}$. Thus

If $P_{dcord} < 0$ then $B_{used} = -BLossTo$

Else $B_{used} = +BLossTo$

$$P_{dcord} = +V_{dcfrom}I_{dc} - R_{dc}I_{dc}^2 - \frac{ALossTo}{1000} - B_{used}I_{dc}$$

$$P_{dcord} = +V_{setfrom}I_{dc} - R_{dc}I_{dc}^2 - \frac{ALossTo}{1000} - B_{used}I_{dc}$$

$$0 = P_{dcord} + \frac{ALossTo}{1000} - V_{setfrom}I_{dc} + B_{used}I_{dc} + R_{dc}I_{dc}^2$$

$$0 = \left(P_{dcord} + \frac{ALossTo}{1000}\right) + (-V_{setfrom} + B_{used})I_{dc} + R_{dc}I_{dc}^2$$

Use the – sign to get the low current / high voltage solution!

$$I_{dc} = \frac{(V_{setfrom} - B_{used}) - \sqrt{(-V_{setfrom} + B_{used})^2 - 4R_{dc}\left(P_{dcord} + \frac{ALossTo}{1000}\right)}}{2R_{dc}}$$

Check if the losses in this calculation are too small.

If $\left(\frac{ALossTo}{1000} - BLossTo|I_{dc}|\right) < \left(\frac{MinLossTo}{1000}\right)$ then resolve using

$$0 = \left(P_{dcord} + \frac{MinLossTo}{1000}\right) + (-V_{setfrom})I_{dc} + R_{dc}I_{dc}^2$$

$$I_{dc} = \frac{(V_{setfrom}) - \sqrt{(-V_{setfrom})^2 - 4R_{dc}\left(P_{dcord} + \frac{MinLossTo}{1000}\right)}}{2R_{dc}}$$

If $R_{dc} = 0$ then adjust previous equations using $P_{dcfrom} = P_{dcto} = V_{setfrom}I_{dc}$ thus we can instead write

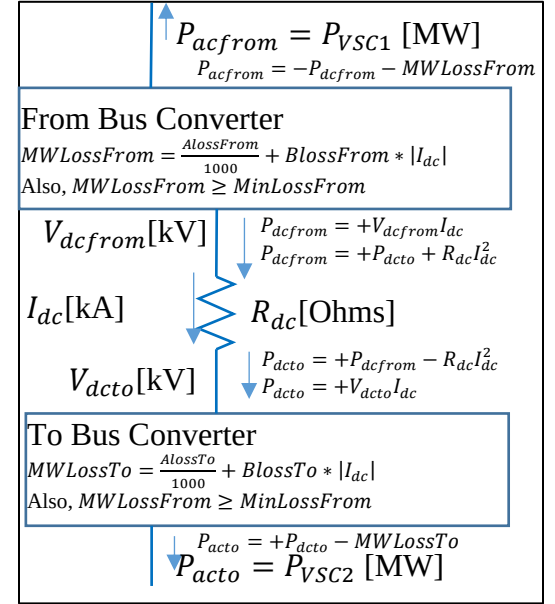
$$P_{dcord} = V_{setfrom}I_{dc} - \frac{ALossTo}{1000} - B_{used}I_{dc}$$

$$P_{dcord} + \frac{ALossTo}{1000} = (V_{setfrom} - B_{used})I_{dc}$$

$$I_{dc} = \left(\frac{P_{dcord} + \frac{ALossTo}{1000}}{V_{setfrom} - B_{used}}\right)$$

If $\left(\frac{ALossTo}{1000} + BLossTo|I_{dc}|\right) < \left(\frac{MinLossTo}{1000}\right)$ then resolve using

$$I_{dc} = \left(\frac{P_{dcord} + \frac{MinLossTo}{1000}}{V_{setfrom}}\right)$$



When the voltage controlling side converter hits a current limit, we may also need equations assuming that the power and voltage are both specified on the same side.

If we have equations with P_{dcord} on the from side and the voltage setpoint also on the from side we get

$$P_{dcord} = P_{acfrom} = -P_{dcfrom} - MWLossFrom$$

If $P_{dcord} > 0$ then $I_{dc} < 0$ so we know $|I_{dc}| = -I_{dc}$. Thus

If $P_{dcord} > 0$ then $B_{used} = -B_{LossFrom}$ Else $B_{used} = +B_{LossFrom}$

$$P_{dcord} = -V_{setFrom}I_{dc} - \frac{ALossFrom}{1000} - B_{used}I_{dc}$$

$$P_{dcord} + \frac{ALossFrom}{1000} = -(V_{setFrom} + B_{used})I_{dc}$$

$$I_{dc} = -\left(\frac{P_{dcord} + \frac{ALossFrom}{1000}}{V_{setFrom} + B_{used}}\right)$$

If $\left(\frac{ALossFrom}{1000} + B_{LossFrom}|I_{dc}|\right) < \left(\frac{MinLossFrom}{1000}\right)$ then resolve using

$$I_{dc} = -\left(\frac{P_{dcord} + \frac{MinLossFrom}{1000}}{V_{setFrom}}\right)$$

If we have equations with P_{dcord} on the to side and the voltage setpoint also on the to side we get

$$P_{dcord} = P_{actto} = +P_{dcto} - MWLossTo$$

If $P_{dcord} < 0$ then $I_{dc} < 0$ so we know $|I_{dc}| = -I_{dc}$. Thus

If $P_{dcord} < 0$ then $B_{used} = -B_{LossTo}$ Else $B_{used} = +B_{LossTo}$

$$P_{dcord} = V_{setTo}I_{dc} - \frac{ALossTo}{1000} - B_{used}I_{dc}$$

$$P_{dcord} + \frac{ALossTo}{1000} = (V_{setTo} - B_{used})I_{dc}$$

$$I_{dc} = \left(\frac{P_{dcord} + \frac{ALossTo}{1000}}{V_{setTo} - B_{used}}\right)$$

If $\left(\frac{ALossTo}{1000} + B_{LossTo}|I_{dc}|\right) < \left(\frac{MinLossTo}{1000}\right)$ then resolve using

$$I_{dc} = \left(\frac{P_{dcord} + \frac{MinLossTo}{1000}}{V_{setTo}}\right)$$

With a value of I_{dc} chosen, variables are passed to the VSC control logic and the loss values are reported.

$$P_{convloss1} = \frac{ALossFrom}{1000} + B_{LossFrom}|I_{dc}|$$

If $P_{convloss1} < \frac{MinLossFrom}{1000}$ then $P_{convloss1} = \frac{MinLossFrom}{1000}$

$$P_{convloss2} = \frac{ALossFrom}{1000} + B_{LossTo}|I_{dc}|$$

If $P_{convloss2} < \frac{MinLossTo}{1000}$ then $P_{convloss2} = \frac{MinLossTo}{1000}$

$$P_{dcloss} = R_{dc}I_{dc}^2$$

If $DCModeFrom = \text{Power}$ then equations solved are

$$P_{VSC1} = P_{dcord}$$

$$P_{VSC2} = -P_{VSC1} - P_{convloss1} - P_{convloss2} - P_{dcloss}$$

Else ($DCModeTo = \text{Power}$), so we use

$$P_{VSC2} = P_{dcord}$$

$$P_{VSC1} = -P_{VSC2} - P_{convloss1} - P_{convloss2} - P_{dcloss}$$

If $DCModeFrom = \text{Voltage}$ then we also set

$$V_{dcfrom} = \text{Setpoint}$$

$$V_{dcto} = V_{dcfrom} - RDC * I_{dc}$$

Else

$$V_{dcto} = \text{Setpoint}$$

$$V_{dcfrom} = V_{dcto} + RDC * I_{dc}$$

Current Limit Logic

The current limit logic takes as an input the desired real and reactive current ($I_{pdesired}$ and $I_{qdesired}$). The value from the power flow data for the VSC DC Line called PWFFrom and PWFTo is used. This stands for “Power Weighting Factor” and describes whether the VSC is programmed to prefer real power (PWF = 1.0) or whether it prefers reactive power (PWF = 0.0). Values between 0.0 and 1.0 can also be specified to weight. This values is used and changes in the current setpoints are modified so that the following equations are satisfied.

- $(I_{pdesired} + \Delta I_P)^2 + (I_{qdesired} + \Delta I_Q)^2 = I_{acmax}^2$
- $\frac{\Delta I_P}{\Delta I_Q} = \left(\frac{1-PWF}{PWF}\right) \frac{I_{pdesired}}{I_{qdesired}}$

To simplify equations define $I_P = I_{pdesired}$; $I_Q = I_{qdesired}$; and $K = \frac{\Delta I_P}{\Delta I_Q}$

We can then write equations to derive either ΔI_P or ΔI_Q first and we get either of the following

$$(I_{pdesired} + K\Delta I_Q)^2 + (I_{qdesired} + \Delta I_Q)^2 = I_{acmax}^2$$

$$(I_{pdesired} + \Delta I_P)^2 + (I_{qdesired} + \frac{1}{K}\Delta I_P)^2 = I_{acmax}^2$$

When $PWF \rightarrow 0$, then this means the value of $\frac{1}{K} \rightarrow 0$. When $PWF \rightarrow 1$, then this means the value of $K \rightarrow 0$.

Numerically it is better in this equation if the value of K is small, so we'll choose the equation that's numerically better.

Define another more general equation and solve this instead because we can use it regardless of which equations we solve for.

$$(I_P + K_P\Delta I)^2 + (I_Q + K_Q\Delta I)^2 = I_{acmax}^2$$

If $PWF > 0.5$ then $K_P = K$ and $K_Q = 1$ Else $K_P = 1$ and $K_Q = \frac{1}{K}$

Solve this more general equation by expanding and then grouping terms.

$$I_P^2 + 2K_P I_P \Delta I + K_P^2 (\Delta I)^2 + I_Q^2 + 2K_Q I_Q \Delta I + (\Delta I)^2 - I_{acmax}^2 = 0$$

$$(I_P^2 + I_Q^2 - I_{acmax}^2) + 2(K_P I_P + K_Q I_Q) \Delta I + (K_P^2 + K_Q^2) (\Delta I)^2 = 0$$

This is a quadratic equation Solve this using the quadratic equation

$$Argument = [2(K_P I_P + K_Q I_Q)]^2 - 4(K_P^2 + K_Q^2)(I_P^2 + I_Q^2 - I_{acmax}^2)$$

If $Argument \geq 0$ then

$$\Delta I = \frac{[-2(K_P I_P + K_Q I_Q) + \sqrt{Arg}]}{2(K_P^2 + K_Q^2)} \text{ and } \Delta I = \frac{[-2(K_P I_P + K_Q I_Q) - \sqrt{Arg}]}{2(K_P^2 + K_Q^2)}$$

ΔI will be chosen as one with smallest absolute value

$$I_{qord} = I_{qdesired} + K_Q \Delta I$$

$$I_{pord} = I_{pdesired} + K_P \Delta I$$

Else there are no real roots of this equations so set the result to

$$I_{qord} = 0$$

$$I_{pord} = \frac{I_{desired}}{|I_{desired}|} I_{acmax}$$

Code for calculation converter power setpoints and losses

Normally, “PowerSide” is the same as the power flow object’s DC power controlling side. However, when the non-DC power controlling converter is at a real current limit we use this function with a power specified on the same a side as the voltage.

```
Inputs Dynamic Model    : PdcOrd
                        : VoltageSide, PowerSide (either From or To)
Inputs Power Flow Model: FromDCMode, RDC, ToVDCSet, FromVDCSet
                        : FromMinLoss, FromALoss, FromBloss, ToMinLoss, ToALoss, ToBloss,
                        : PowerSide
Local Variables         : _UseBloss, _UseVset
Output Variables        : DCCurrent_kAmp, VdcFrom, VdcTo
                        : PVSC1, PVSC2, Pconv_Loss1, Pconv_Loss2, Pdc_loss

// First Calculate the DC Current in Kiloamps
if VoltageSide = FROM then _UseVset := FromVDCSet Else _UseVset := ToVDCSet

If PowerSide = FROM then begin // From side controls Power
  if PdcOrd > 0 then _UseBloss := -FromBloss else _UseBloss := FromBloss;
  if (abs(RDC) < 0.00001) or (VoltageSide = FROM) then begin
    // Note: _UseVset is done because equations are same for two conditions
    DCCurrent_kAmp := -(PdcOrd + FromALoss/1000)/(_UseVset + _UseBloss);
    if (FromALoss + FromBloss*abs(DCCurrent_kAmp)*1000) < FromMinLoss then
      DCCurrent_kAmp := -(PdcOrd + FromMinLoss/1000)/(_UseVset);
  end
  else begin
    DCCurrent_kAmp := 0.5/RDC*(-(ToVDCSet+_UseBloss)+ sqrt(sqr(ToVDCSet+_UseBloss)-4*RDC*(PdcOrd+FromALoss/1000)));
    if (FromALoss + FromBloss*abs(DCCurrent_kAmp)*1000) < FromMinLoss then
      DCCurrent_kAmp := 0.5/RDC*(-ToVDCSet+sqrt(sqr(ToVDCSet)-4*RDC*(PdcOrd+FromMinLoss/1000)));
  end;
end
else begin // To Side controls Power
  if PdcOrd < 0 then _UseBloss := -ToBloss else _UseBloss := ToBloss;
  if (abs(RDC) < 0.00001) or (VoltageSide = TO) then begin
    // Note: _UseVset is done because equations are same for two conditions
    DCCurrent_kAmp := (PdcOrd + ToALoss/1000)/(_UseVset + _UseBloss);
    if (ToALoss + ToBloss*abs(DCCurrent_kAmp)*1000) < ToMinLoss then
      DCCurrent_kAmp := (PdcOrd + ToMinLoss/1000)/(_UseVset);
  end
  else begin
    DCCurrent_kAmp := 0.5/RDC*((FromVDCSet-_UseBloss)-sqrt(sqr(FromVDCSet-_UseBloss)-4*RDC*(PdcOrd+ToALoss/1000)));
    if (ToALoss + ToBloss*abs(DCCurrent_kAmp)*1000) < (ToMinLoss/1000) then
      DCCurrent_kAmp := 0.5/RDC*(FromVDCSet-sqrt(sqr(FromVDCSet)-4*RDC*(PdcOrd+ToMinLoss/1000)));
  end;
end;
end;

Pconv_loss2 := ToALoss/1000 + ToBloss*abs(DCCurrent_kAmp);
if (Pconv_loss2) < (ToMinLoss/1000) then Pconv_loss2 := (ToMinLoss/1000);
Pconv_loss1 := FromALoss/1000 + FromBloss*abs(DCCurrent_kAmp);
if (Pconv_loss1) < (FromMinLoss/1000) then Pconv_loss1 := (FromMinLoss/1000);
Pdc_loss := Rdc*sqr(DCCurrent_kAmp);
if FromDCMode = "Power" then begin
  PVSC1 := PdcOrd;
  PVSC2 := -PVSC1 - Pconv_loss1 - Pconv_loss2 - Pdc_loss;
end
else begin
  PVSC2 := PdcOrd;
  PVSC1 := -PVSC2 - Pconv_loss1 - Pconv_loss2 - Pdc_loss;
end;

if VoltageSide = FROM then begin
  VdcFrom := FromVDCSet;
  VdcTo := VdcFrom - RDC*DCCurrent_kAmp;
end
else begin
  VdcTo := ToVDCSet;
  VdcFrom := VdcTo + RDC*DCCurrent_kAmp;
end;
end;
```

Current Limit Logic Pseudocode

Inputs Dynamic Model: **Ipdesired**, **Iqdesired**, **Iacmax**,

Inputs Power Flow : **PWF**

Outputs : **Ipord** and **Iqord**

Local Variable : **_K**, **_KP**, **_KQ**, **_A**, **_B**, **_C**, **_Argument**, **_DeltaA**, **_DeltaB**

```
if (sqrt(sqr(Ipdesired) + sqr(Iqdesired)) <= Iacmax) then begin
    Ipord := Ipdesired;
    Iqord := Iqdesired;
end
else begin
    if (PWF = 0) or (abs(Iqdesired) < 0.000001) then begin
        // Reduce P first, or Q is already almost zero!
        if abs(Iqdesired) > Iacmax then begin
            Ipord := 0;
            Iqord := Iacmax;
            if Iqdesired < 0 then Iqord := -Iqord;
        end
        else begin
            Iqord := Iqdesired;
            Ipord := sqrt(sqr(Iacmax) - sqr(Iqord));
            if Ipdesired < 0 then Ipord := -Ipord;
        end
    end
    else if (PWF = 1) or (abs(Ipdesired) < 0.000001) then begin
        // Reduce Q first, or P is already almost zero!
        if abs(Ipdesired) > Iacmax then begin
            Iqord := 0;
            Ipord := Iacmax;
            if Ipdesired < 0 then Ipord := -Ipord;
        end
        else begin
            Ipord := Ipdesired;
            Iqord := sqrt(sqr(Iacmax) - sqr(Ipord));
            if Iqdesired < 0 then Iqord := -Iqord;
        end
    end
    else begin
        _K := (Ipdesired/Iqdesired) * ((1-PWF)/PWF);
        if (abs(PWF) > 0.5) then begin // --> means that K is approaching a small number
            _KP := _K;
            _KQ := 1;
        end
        else begin // --> means that 1/K is approaching a small number
            _KQ := 1/_K;
            _KP := 1;
        end
        _A := sqrt(_KP) + sqrt(_KQ);
        _B := 2*Ipdesired*_KP + 2*Iqdesired*_KQ;
        _C := sqrt(Ipdesired) + sqrt(Iqdesired) - sqrt(Iacmax);
        _Argument := sqrt(_b) - 4*_a*_c;
        if _Argument < 0 then begin
            // No solution exists. Reduce Q to zero and keep the MW output
            Iqord := 0;
            Ipord := Iacmax;
            if Ipdesired > 0 then Ipord := -Ipord;
        end
        else begin
            _Argument := sqrt(_Argument);
            _DeltaA := (-_b + _Argument)/(2*_a);
            _DeltaB := (-_b - _Argument)/(2*_a);
            if abs(_DeltaA) < abs(_DeltaB) then begin
                Iqord := Iqdesired + _DeltaA*_KQ;
                Ipord := Ipdesired + _DeltaA*_KP;
            end
            else begin
                Iqord := Iqdesired + _DeltaB*_KQ;
                Ipord := Ipdesired + _DeltaB*_KP;
            end
        end
    end
end;
```