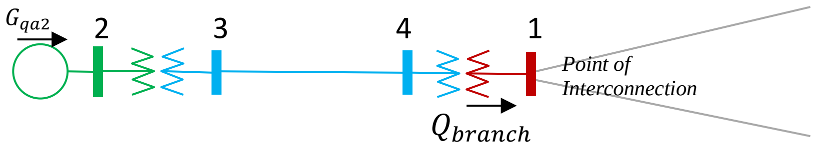
Detailed Documentation of Software Implementation of Voltage Droop With Deadband in Power Flow Calculation

Date : November 2018
Prepared by : James Weber, Ph.D.
Director of Software Development
PowerWorld Corporation
(217) 384-6330 ext. 13
weber@powerworld.com

2001 South First Street
Champaign, IL 61820
(217) 384-6330
www.powerworld.com

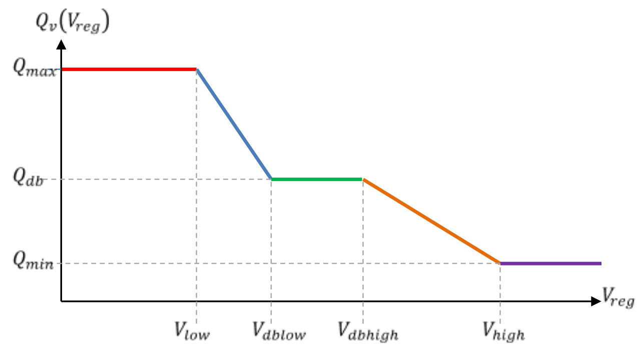
Contents

1. Simple Voltage Droop Control



It is common for wind and solar plants to be modeled as a generator (bus 2 above) connected to an equivalent step-up transformer (bus 2 – 3), in series with an equivalent feeder (bus 3 – 4) and then a substation transformer (bus 4 – 1). One could imagine the generators connected at 0.5 kV, a 13.8 kV feeder, and then a 13.8 to 115 kV transformer at the point of interconnection. In this configuration, the plants are often not configured with a traditional "voltage setpoint", but instead follow a voltage droop control with control centered at the point of interconnection. The voltage droop control is then specified by the user to follow a QV characteristic curve which is a function of the RegBus voltage V_{reg} and the purpose of the curve is to enforce that the total reactive flow going into the RegBus from the respective VoltageDroopControl is equal to this value. The curve will have the following shape and is discussed in more detail later.

To model this in a set of power flow equations, we must introduce a new type of equation. The equations at bus 1, 3, and 4 would remain the same with typical Q equations that take the summation of reactive flows going into the bus and sum them to zero. The equation at bus 2 however is replaced by a new equation that specifies that the flow on the branch arriving bus 1 from bus 4 is equal to the QV-characteristic function.



$$Q_{branch}(V_1, Angle_1, V_4, Angle_4) - Q_v(V_1) = 0$$

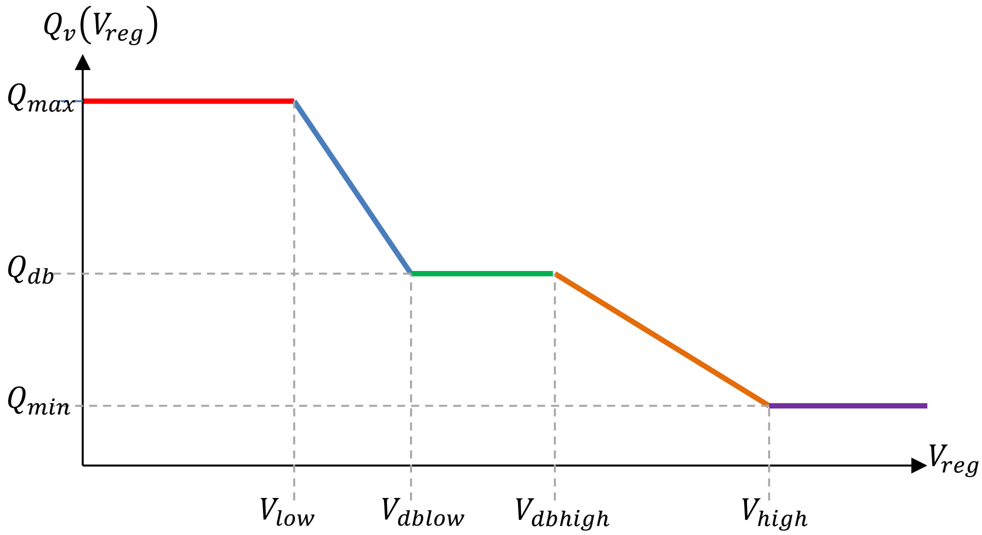
There are several wrinkles to this concept though that complicate the numerical implemention of these equations in the power flow.

1. The function $Q_v(V_1)$ is continuous, but the ***derivative*** of this function is discontinuous. Discontinuous derivatives introduce numerical difficulties that are address by smoothing this function at the corners. This is described in great detail in the next section.
2. Multiple generator buses can act together to perform this VoltageDroopControl. They coordinate their control in the same way that remote voltage regulation sharing is done for voltage control
3. Multiple VoltageDroopControl functions can be applied to the same regulated bus with different sets of generators belonging to each VoltageDroopControl equation
4. Care must be taken when the regulated bus is connected by very low impedance branches. This can create numerical troubles in the solution which must be taken into account.

A completely generalized Voltage Droop Control is described in a subsequent section that handles the wrinkles 2, 3, and 4 above.

2. Voltage Droop Control QV characteristic function

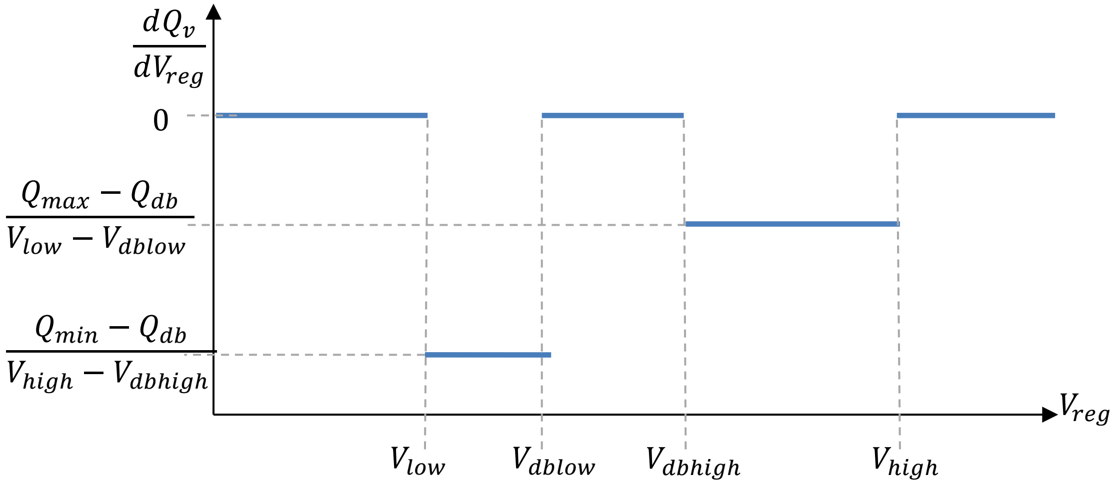
The voltage droop control QV characteristic functions are defined with 7 input variables. There is a deadband region between V_{dblow} and V_{dbhigh} in which the reactive output is Q_{db}. Below this deadband the reactive power linearly increases up to Q_{max} at a voltage of V_{low} and above the deadband the reactive power linearly decreases down to Q_{min} at a voltage of V_{high}.



Within PowerWorld Simulator, we constrain this function with a voltage tolerance (Tol) and also the relationship of reactive values using the following code which ensures that the curve is a non-increasing function of voltage, and also that it is not too steep a derivative!

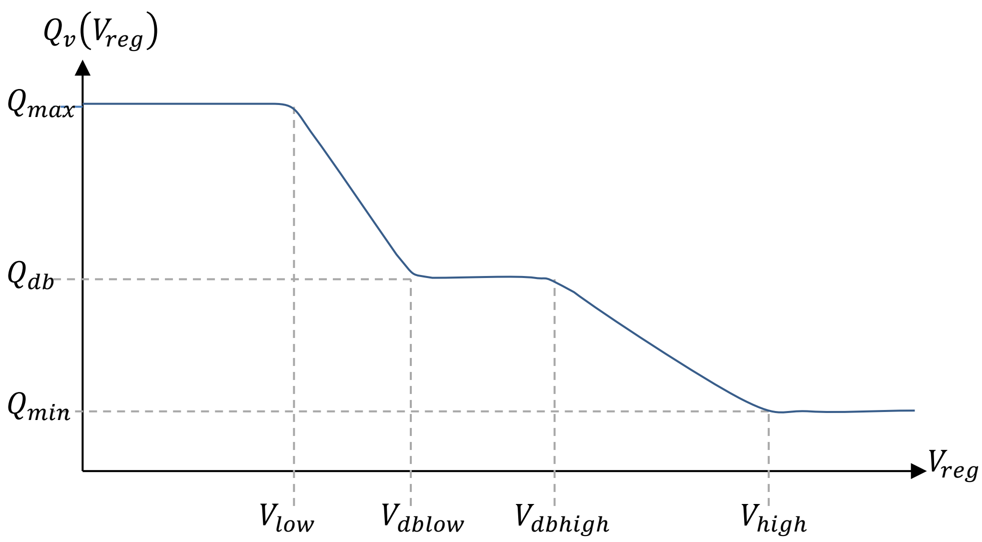
```
// Enforce voltage thresholds that are too close to one another
If (Vdbhigh - Vdblow) < Tol then
  VdbhighUsed = (Vdbhigh+Vdblow)/2
  VdblowUsed  = (Vdbhigh+Vdblow)/2
Else
  VdbhighUsed = Vdbhigh
  VdblowUsed  = Vdblow
EndIf
If (VdblowUsed – Vlow) < Tol then VlowUsed = VdblowUsed – Tol
Else VlowUsed = Vlow
If (Vhigh – VdbhighUsed) < Tol then VhighUsed = VdbhighUsed + Tol
Else VhighUsed = Vhigh
// Enforce decreasing relationship between reactive powers
If Qmax < Qdb then QmaxUsed = Qdb
Else QmaxUsed = Qmax
If Qmin > Qdb then QminUsed = Qdb
Else QminUsed = Qmin
// The next code ensures that the slope isn’t too large
// PowerWorld internally has a tolerance ZBRThres which is 0.0002 by
// default how voltage control works around branches with X below this // This
is similar situation in which we want slope to not be too large
MaxSlope := 1/ZBRThreshold;
dQdV := (QmaxUsed - Qdb)/(VdbLowUsed - VlowUsed);
if dQdV > MaxSlope/Sbase then
  VlowUsed  := VdbLowUsed  – Sbase/MaxSlope*(QmaxUsed - Qdb);
dQdV := (QminUsed - Qdb)/(VdbhighUsed - VhighUsed);
if dQdV > MaxSlope/Sbase then
  VhighUsed := VdbhighUsed - Sbase/MaxSlope*(QminUsed - Qdb);
```

Another problem with this function is that the transition points between sections of the curve introduce a non-continous derivative for $\frac{dQ_v}{dVreg}$ which will create a problem in the iterative solution of the power flow equations.

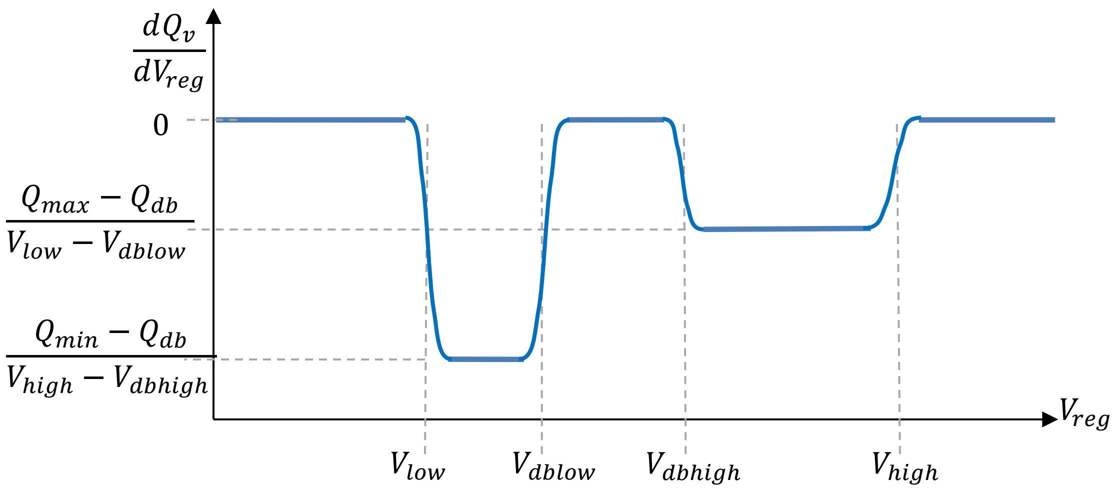


These types of edge points are not a problem in transient stability simulation which include curves like this because at each time point we simply evaluate the functions and use them to feed into integrator blocks which smooth out the system response. In the power flow equations however we are strictly enforcing all these equations simultaneously, so a discontuous derivative like this will create a numerical problem in our solution algorithms.

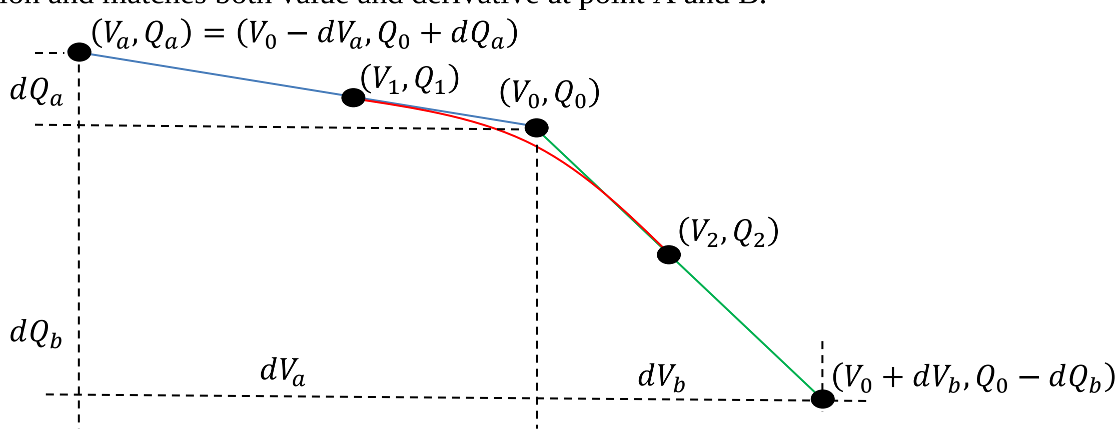
To overcome this problem, we will modify our QV curve slightly by introducing a cubic spline to smooth out these points. Thus the QV characteristic will instead look as below with the corners "rounded" off a bit.



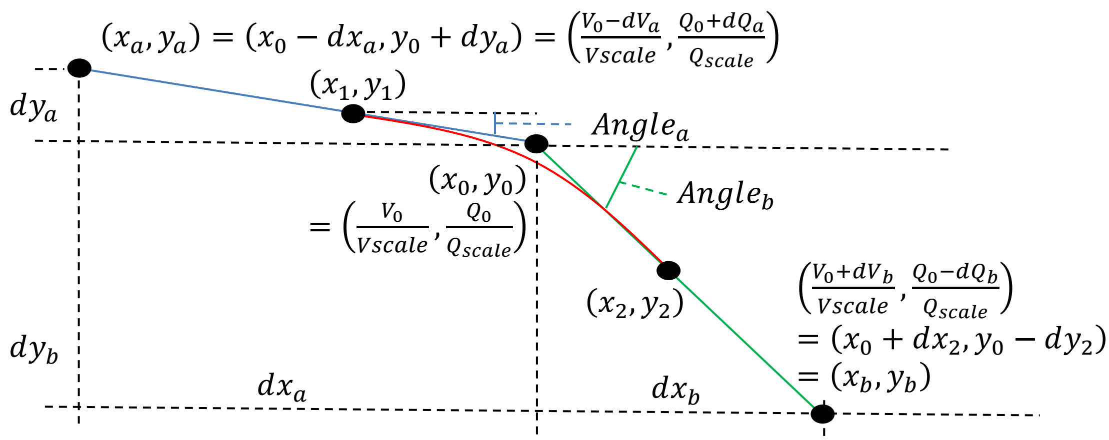
This will result in the $\frac{dQ_v}{dVreg}$ cruve which has a region around each corner where the derivative transitions between the values. This is depicted in the next picture.



The way we will determine this width of this transition region is as follows. For 2 line segments in our QV characteristics we will fine a point on either side of the corner point and then define a spline function that represents the curve between these two points which smooths out the transition and matches both value and derivative at point A and B.



To derive these points and define a function, we first define a scaling for the Voltage and Reactive Power around this corner as described in the next image.



The scaling in the voltage axis is based on 10% of the minimum change in voltage across the two line segments, but no smaller than a minimum tolerance $V_{scalemin} = 0.001$. The scaling on the Q axis is 10% of the total change across both line segments, but no smaller than a minimum tolerance $Q_{scalemin} = MVA\ Convergence\ Tolerance$. In addition, we want points (x_1, y_1) and (x_2, y_2) to be a distance of 1.0 away from the corner point (x_0, y_0) in the scaled coordinates, but not more than 50% of the way down their respective line segments. To make sure they are not 50% of the way, a restriction in the choice of scaling is made such that the length of both line segments in the scaled coordinates is at least 2.0.

Finally, we will use the geometry in the scaled coordinates to choose a function type between points (x_1, y_1) and (x_2, y_2). If the angle difference (calculated in the scaled coordinates) between the two line segments is less than 12 degrees (pi/15 radians), then instead of using a circlular spline we will switch to a cubic spline. This is done because as the angle difference gets smaller, the circular function results in a circle with an extremely large radius. Doing a little geometry you will find that that $R = \frac{1}{\arctan\left(\frac{abs(Angle1-Angle2)}{2}\right)}$. At angle difference 12 degrees this gives $R = 9.58$.

When using a circular spline, the function will following the circle equation of

$$\left(\frac{V_{pu}}{V_{scale}} - x_c\right)^2 + \left(\frac{Q_{pu}}{Q_{scale}} - y_c\right)^2 = R^2$$

Solving this for Q_{pu} as a function of V_{pu} gives

$$Q_{pu} = Q_{scale}\left[y_c \pm \sqrt{R^2 - \left(\frac{V_{pu}}{V_{scale}} - x_c\right)^2}\right]$$

The $\pm$ sign convention will depend on the relationship between the slopes of the lines. This will be discussed in another section. But generally we will write our equation by adding another parameter $Sign$ instead as follows.

$$Q_{pu} = Q_{scale}\left[y_c + Sign\sqrt{R^2 - \left(\frac{V_{pu}}{V_{scale}} - x_c\right)^2}\right]$$

The derivative of this function is then given as

$$\frac{dQ_{pu}}{dV_{pu}} = \frac{Q_{scale}}{V_{scale}}\left[\frac{Sign\left(\frac{V_{pu}}{V_{scale}} - x_c\right)}{\sqrt{R^2 - \left(\frac{V_{pu}}{V_{scale}} - x_c\right)^2}}\right]$$

When using a cubic spline, we will use a function

$$Q_{pu} = Q_{scale}\left[a + b\left(\frac{V_{pu}}{V_{scale}} - x_{shift}\right) + c\left(\frac{V_{pu}}{V_{scale}} - x_{shift}\right)^2 + d\left(\frac{V_{pu}}{V_{scale}} - x_{shift}\right)^3\right]$$

The derivative of this function will be

$$Q_{pu} = \frac{Q_{scale}}{V_{Scale}}\left[b + 2c\left(\frac{V_{pu}}{V_{scale}} - x_{shift}\right) + 3d\left(\frac{V_{pu}}{V_{scale}} - x_{shift}\right)^2\right]$$

2.1. Pseuedo-code to choose Vscale, Qscale and Function Type

```
Input to Psuedo-Code for building corner function
      Va, V0, Vb
      Qa, Q0, Qb
Output of this code will be
      x1, x0, x2, y1, y0, y2 : values in in scaled coordinates
      Vscale, Qscale         : scaling for V and Q axis
      SplineFunction         : either circle or cubic
Function
```

```
dVa = V0-Va
dVb = Vb-V0
dQa = Qa-Q0
dQb = Q0-Qb
if abs(dVa) < abs(dVb) then Vscale = 0.10*abs(dVa)
else Vscale = 0.10*abs(dVb)
if Vscale < 0.001 then Vscale = 0.001
Qscale := 0.10*( abs(dQa) + abs(dQb) )
if Qscale < MVATolerance then Qscale = MVATolerance

dya = dQa/Qscale
dyb = dQb/Qscale
dxa = dVa/Vscale
dxb = dVb/Vscale
da = sqrt( sqr(dxa) + sqr(dya) )
db = sqrt( sqr(dxb) + sqr(dyb) )

if da < db then dMin := da
else dMin := db;
// We need the length in the transformed coordinates to be > 2.0
// otherwise point 1 or 2 will be more than 50% down segment
if dMin < 2 then begin
  Qscale = Qscale * dMin/2
  Vscale = Vscale * dMin/2
  da = da * 2/dMin
  db = db * 2/dMin
  dxa = dxa * 2/dMin
  dya = dya * 2/dMin
  dxb = dxb * 2/dMin
  dyb = dyb * 2/dMin
end;
PercA = 1/da
PercB = 1/db

x0 = Q0/Qscale
y0 = V0/Vscale
x1 = x0 - dxa*PercA
y1 = y0 – dya*PercA
x2 = x0 + dxb*PercB
y2 = y0 + dyb*PercB

AngleA = arctan2(dya, dxa)
AngleB = arctan2(dyb, dxb)
if abs(AngleB – AngleA) > pi/15   // at 12 degrees, switch
then SplineType = Circle Equation // use a circle spline
else SplineType = Cubic Equation  // use a cubic spline
```

2.2. Pseudo-code for Circular Spline Function

```
Input to Psuedo-Code for building corner function
      theQscale, theVscale   : store scaling
      x1, x0, x2, y1, y0, y2 : values in in scaled coordinates
Create an object with the following parameters to represent the Circle
      Qscale, Vscale    : store scaling
      xc, yc, R2, Sign  : values describe the circle in scaled coordinates
```

```
constructor TSplineSegmentCircle.CreateCurve(theVscale, theQscale, x1, y1, x0, y0, x2, y2
: double)
Var m1, m2 : double
begin
  Qscale = theQscale
  Vscale = theVscale
  if    y0 = y1 then begin                 // 1------0
    xc = x1                                //         \
    yc = y2 - (x0-x2)/(y0-y2)*(x1-x2)      //          \
    R2 = sqr(y1-yc)                        //           \
    Sign = +1                              //            2
  end
  else if y0 = y2 then begin               // 1
    xc = x2                                //  \
    yc = y1 - (x0-x1)/(y0-y1)*(x2-x1)      //   \
    R2 = sqr(y2-yc)                        //    \
    Sign = -1                              //     0------2
  end
  else if x0 = x1 then begin               // 1
    yc = y1                                // |
    xc = x2 - (y0-y2)/(x0-x2)*(y1-y2)      // |
    R2 = sqr(x1-xc)                        // 0----
    Sign = -1                              //      \-----2
  end
  else if x0 = x2 then begin               // 1----
    yc = y2                                //      \-----0
    xc = x1 - (y0-y1)/(x0-x1)*(y2-y1)      //            |
    R2 = sqr(x2-xc)                        //            |
    Sign = +1                              //            2
  end
  else begin
    m1 = (y0-y1)/(x0-x1)
    m2 = (y0-y2)/(x0-x2)
    xc = (m1*(m2*y2 + x2) - m2*(m1*y1 + x1))/(m1-m2)
    yc = ((m1*y1 + x1) - (m2*y2 + x2))/(m1-m2)
    R2 = sqr(x1-xc) + sqr(y1-yc)
    if m1 < m2 then Sign = -1
    else Sign = +1
  end
end
```

Once the circular spline object is created functions for Output and Derivative would be written with the following pseudo-code.

```
Function TSplineSegmentCircle.Output(x : double) : double
begin
  x = x/Vscale
  result = R2 - sqr(x - xc);
  if result > 0 then result = yc + sign * sqrt( result )
  else result = yc // function should never be used in this range!
  result := result*Qscale
end;

function TSplineSegmentCircle.Derivative(x : double) : double
begin
  x = x/Vscale
  result = R2 - sqr(x - xc)
  if result > 0 then result = -Sign*(x - xc)/sqrt( result )
  else result = -1E6;
  result = result*Qscale/Vscale
end;
```

2.3. Psuedo-code for Cubic Spline Function

```
Input to Psuedo-Code for building corner function
      theQscale, theVscale    : store scaling
      x1, x0, x2, y1, y0, y2 : values in in scaled coordinates
Create an object with the following parameters to represent Cubic function
      Qscale, Vscale    : store scaling
      Xshift, A, B, C, D : values describe the cubic in scaled coordinates
```

```
constructor TSplineSegmentCubic.CreateCurve(theVscale, theQscale, x1, y1, x0, y0, x2, y2
: double)
local variables
    at, bt, ct, dt : double
    m1, m2, x12 : double
    m1_m_m2, m1_p_m2, x1_m_x2, y1_m_y2 : double
begin
  Vscale := theVscale
  Qscale := theQscale

  Xshift = x1
  x0 = x0 - XShift
  x1 = x1 - XShift
  x2 = x2 - XShift

  // calculation assumes that the x values are increasing x1 < x0 < x2
  m1 = (y0-y1)/(x0-x1)
  m2 = (y0-y2)/(x0-x2)
  m1_p_m2 = m1 + m2
  m1_m_m2 = m1 - m2
  x1_m_x2 = x1 - x2
  y1_m_y2 = y1 - y2

  // Calculate values on shifted coordinates around [(x1+x2)/2, y1]
  at = (y1+y2)/2 - (x1_m_x2)*(m1_m_m2)/8
  bt = 3/2*(y1_m_y2)/(x1_m_x2) - (m1_p_m2)/4
  ct = 1/2*(m1_m_m2)/(x1_m_x2)
  dt = (m1_p_m2)/sqr(x1_m_x2) - 2*(y1_m_y2)/((x1_m_x2)*sqr(x1_m_x2))

  // shift back to the coordinates of centered around original x1, y1
  x12 = (x1+x2)/2
  A = at -   bt*x12 +   ct*sqr(x12) - dt*x12*sqr(x12)
  B = bt - 2*ct*x12 + 3*dt*sqr(x12)
  C = ct - 3*dt*x12
  D = dt
end
```

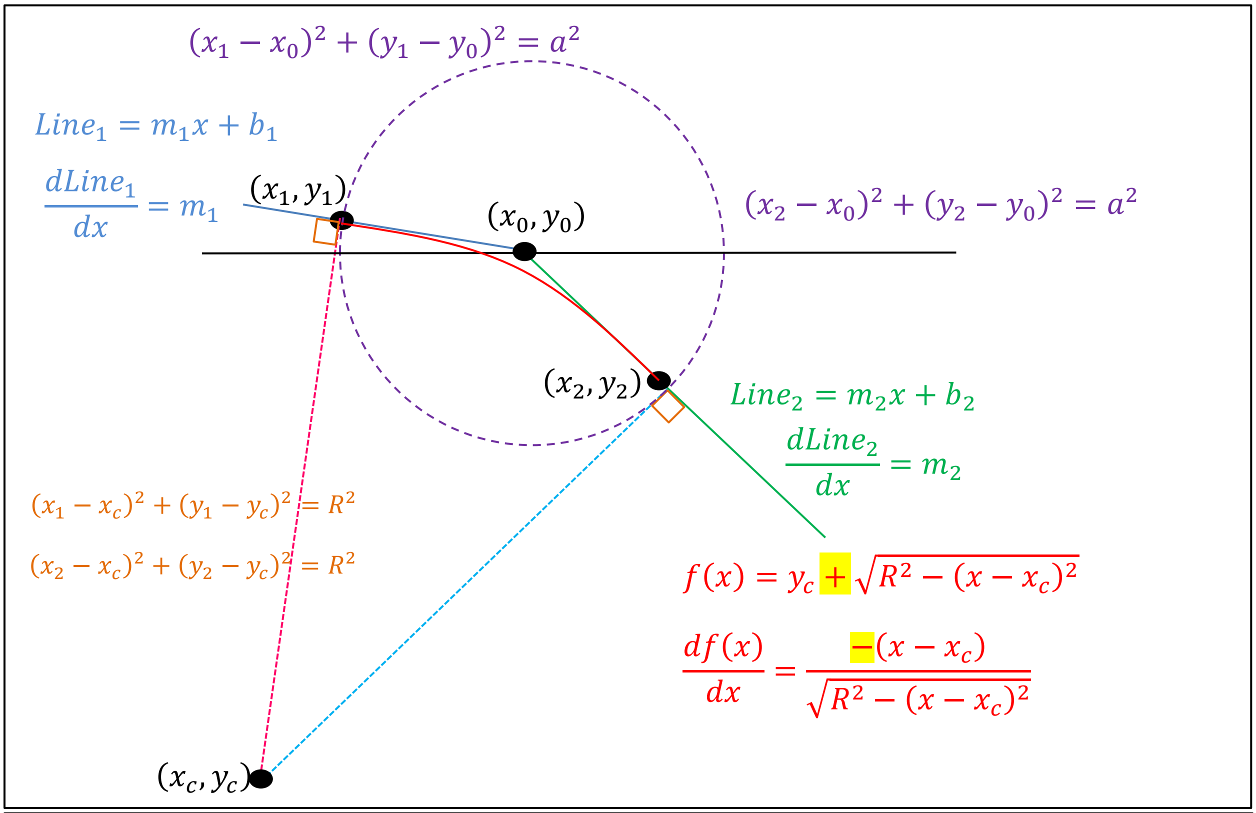
One the cubic spline object is created functions for Output and Derivative would be written with the following pseudo-code.

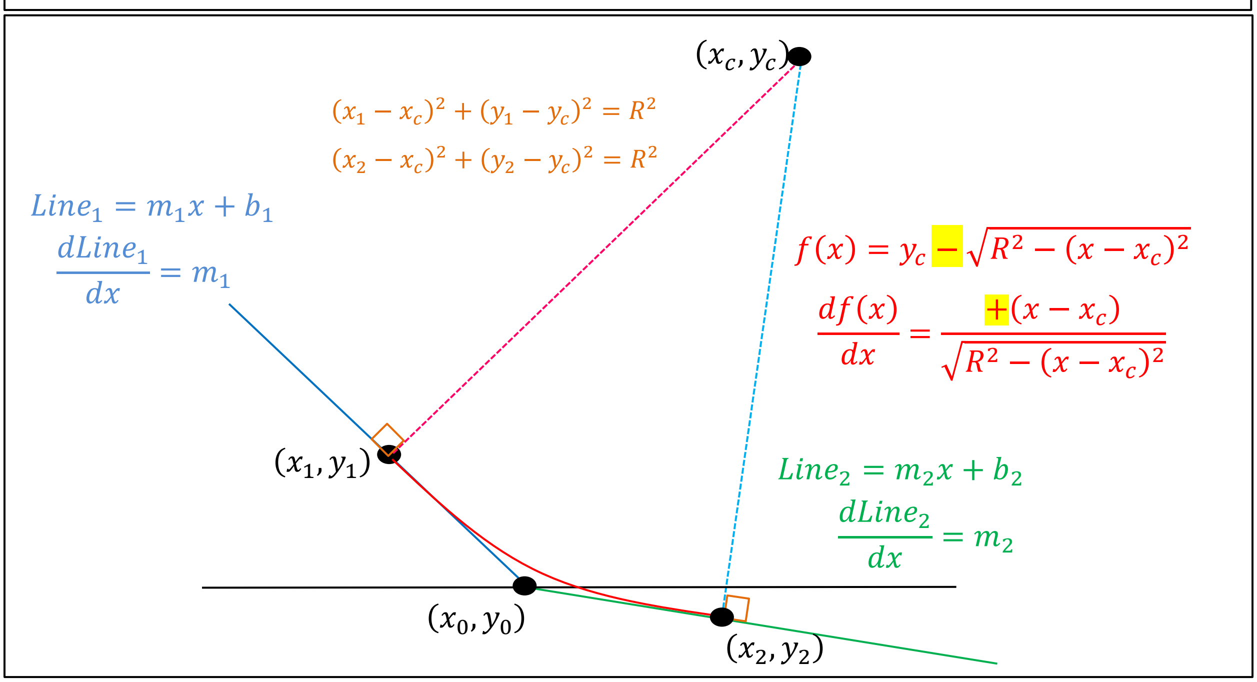
```
function TSplineSegmentCubic.Output(X : double) : double
begin
  X = X/Vscale-XShift
  result = A + B*X + C*sqr(X) + D*X*Sqr(X)
  result = Qscale*result
end

function TSplineSegmentCubic.Derivative(X : double) : double
begin
  X = X/Vscale - XShift
  result = B + 2*C*X + 3*D*Sqr(X)
  result = Qscale/Vscale*result
end
```

2.4. Circular Spline Derivation

We will have two possible circular splines depending on whether the first slope is larger or smaller than the second slope.





Given these constraints that points 1 and 2 are equidistant from point 0, we can calculate the spline as follows.

Define the following slopes: $m_1 = \frac{y_0 - y_1}{x_0 - x_1}$ and $m_2 = \frac{y_0 - y_2}{x_0 - x_2}$.
We must solve for the intersection of the lines going toward (x_c, y_c).
The equation of the line from (x_c, y_c) to (x_1, y_1) is $y = y_1 - \frac{x - x_1}{m_1}$.
The equation of the line from (x_c, y_c) to (x_2, y_2) is $y = y_2 - \frac{x - x_2}{m_2}$.
This means that $y_c = y_1 - \frac{x_c - x_1}{m_1}$ and $y_c = y_2 - \frac{x_c - x_2}{m_2}$. Equate these and get

$y_1 - \frac{x_c - x_1}{m_1} = y_2 - \frac{x_c - x_2}{m_2}$

$+\frac{x_c - x_2}{m_2} - \frac{x_c - x_1}{m_1} = +y_2 - y_1$

$\frac{x_c}{m_2} - \frac{x_c}{m_1} = +y_2 - y_1 + \frac{x_2}{m_2} - \frac{x_1}{m_1}$

$x_c \left(\frac{1}{m_2} - \frac{1}{m_1}\right) = +y_2 - y_1 + \frac{x_2}{m_2} - \frac{x_1}{m_1}$

$x_c = \frac{+y_2 - y_1 + \frac{x_2}{m_2} - \frac{x_1}{m_1}}{\frac{1}{m_2} - \frac{1}{m_1}}$

$x_c = \frac{+m_2 m_1 y_2 - m_2 m_1 y_1 + m_1 x_2 - m_2 x_1}{m_1 - m_2}$

$$x_c = \frac{m_1(x_2 + m_2 y_2) - m_2(x_1 + m_1 y_1)}{m_1 - m_2}$$

With this value of x_c, substitude back into one of y_c equations to solve for y_c.

$y_c = y_1 - \frac{\left(\frac{+y_2 - y_1 + \frac{x_2}{m_2} - \frac{x_1}{m_1}}{\frac{1}{m_2} - \frac{1}{m_1}}\right) - x_1}{m_1}$

$y_c = y_1 - \left[\frac{+y_2 - y_1 + \frac{x_2}{m_2} - \frac{x_1}{m_1}}{\left(\frac{1}{m_2} - \frac{1}{m_1}\right) m_1} - \frac{x_1}{m_1}\right]$

$y_c = y_1 - \left[\frac{+y_2 - y_1 + \frac{x_2}{m_2} - \frac{x_1}{m_1}}{m_1\left(\frac{1}{m_2} - \frac{1}{m_1}\right)} - \frac{x_1\left(\frac{1}{m_2} - \frac{1}{m_1}\right)}{m_1\left(\frac{1}{m_2} - \frac{1}{m_1}\right)}\right]$

$y_c = y_1 - \left[\frac{+y_2 - y_1 + \frac{x_2 - x_1}{m_2} - \frac{x_1 - x_1}{m_1}}{\left(\frac{m_1 - m_2}{m_2}\right)}\right]$

$y_c = \frac{y_1(m_1 - m_2)}{m_1 - m_2} - \left[\frac{m_2(+y_2 - y_1) + x_2 - x_1}{m_1 - m_2}\right]$

$y_c = \frac{m_1 y_1 - m_2 y_1 - m_2 y_2 + m_2 y_1 - x_2 + x_1}{m_1 - m_2}$

$y_c = \frac{m_1 y_1 - m_2 y_1 - m_2 y_2 + m_2 y_1 - x_2 + x_1}{m_1 - m_2}$

$$y_c = \frac{(+x_1 + m_1 y_1) - (x_2 + m_2 y_2)}{m_1 - m_2}$$

Actually there are some degenerate cases here for a circular spline when the the slope values go to either zero or infinite. Thus the total logic is as follows

If $y_0 = y_1$ then

$x_c = x_1$

$y_c = y_2 - \left(\frac{x_0 - x_2}{y_0 - y_2}\right)(x_1 - x_2)$

$R^2 = (y_1 - y_c)^2$

$Sign = +1$

Else If $y_0 = y_2$ then

$x_c = x_2$

$y_c = y_1 - \left(\frac{x_0 - x_1}{y_0 - y_1}\right)(x_2 - x_1)$

$R^2 = (y_2 - y_c)^2$

$Sign = -1$

Else If $x_0 = x_1$ then

$y_c = y_1$

$x_c = x_2 - \left(\frac{y_0 - y_2}{x_0 - x_2}\right)(y_1 - y_2)$

$R^2 = (x_1 - x_c)^2$

$Sign = -1$

Else If $x_0 = x_2$ then

$y_c = y_2$

$x_c = x_1 - \left(\frac{y_0 - y_1}{x_0 - x_1}\right)(y_2 - y_1)$

$R^2 = (x_2 - x_c)^2$

$Sign = +1$

Else

$m_1 = \frac{y_0 - y_1}{x_0 - x_1}$

$m_2 = \frac{y_0 - y_2}{x_0 - x_2}$

$x_c = \frac{m_1(x_2 + m_2 y_2) - m_2(x_1 + m_1 y_1)}{m_1 - m_2}$

$y_c = \frac{(+x_1 + m_1 y_1) - (x_2 + m_2 y_2)}{m_1 - m_2}$

$R^2 = (x_1 - x_c)^2 + (y_1 - y_c)^2$

If $m_1 < m_2$ Then $Sign = -1$

Else $Sign = +1$

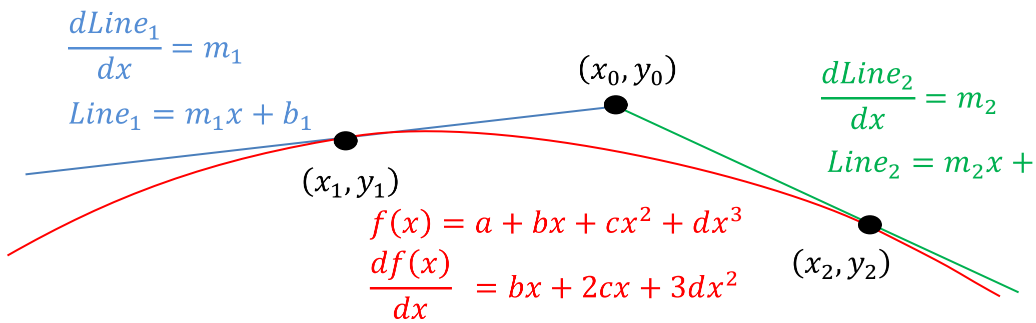
Circular Spline Function and derivative are then

$$f(x) = y_c + Sign\sqrt{R^2 - (x - x_c)^2}$$

$$\frac{df(x)}{dx} = \frac{-Sign(x - x_c)}{\sqrt{R^2 - (x - x_c)^2}}$$

2.5. Cubic Spline Derivation

We are given 3 points (x_1, y_1), (x_0, y_0), and (x_2, y_2). These define two lines that join point 1 to 0 and point 0 to 2. We then want to calculate the cooefficients of the cubic function (a, b, c, and d) such that at point 1 and point 2 the cubic function matches both the value and derivative of the intersecting lines. This is depicted in the following figure.



The coefficients that match this results are as follows.

- $m_1 = \frac{(y_0 - y_1)}{(x_0 - x_1)}$
- $m_2 = \frac{(y_0 - y_2)}{(x_0 - x_2)}$
- $\tilde{a} = \frac{(y_1 + y_2)}{2} - \frac{(x_1 - x_2)(m_1 - m_2)}{8}$
- $\tilde{b} = \frac{3}{2}\frac{(y_1 - y_2)}{(x_1 - x_2)} - \frac{(m_1 + m_2)}{4}$
- $\tilde{c} = \frac{(m_1 - m_2)}{2(x_1 - x_2)}$
- $\tilde{d} = \frac{m_1 + m_2}{(x_1 - x_2)^2} - \frac{2(y_1 - y_2)}{(x_1 - x_2)^3}$
- $a = \tilde{a} - \tilde{b}\left(\frac{x_1 + x_2}{2}\right) + \tilde{c}\left(\frac{x_1 + x_2}{2}\right)^2 - \tilde{d}\left(\frac{x_1 + x_2}{2}\right)^3$
- $b = \tilde{b} - 2\tilde{c}\left(\frac{x_1 + x_2}{2}\right) + 3\tilde{d}\left(\frac{x_1 + x_2}{2}\right)^2$
- $c = \tilde{c} - 3\tilde{d}\left(\frac{x_1 + x_2}{2}\right)$
- $d = \tilde{d}$

The derivation of this results is as follows. We can write 2 equations the require that the values match and points 1 and 2 and two equations that require that the derivatives match at the points.

1. $a + bx_1 + cx_1^2 + dx_1^3 = y_1$
2. $a + bx_2 + cx_2^2 + dx_2^3 = y_2$
3. $b + 2cx_1 + 3dx_1^2 = m_1 = \frac{y_0 - y_1}{x_0 - x_1}$
4. $b + 2cx_2 + 3dx_2^2 = m_2 = \frac{y_0 - y_2}{x_0 - x_2}$

To simplify these equations, transform the coordinates to a modified set such that $\tilde{x}_1 = -\tilde{x}_2$. Do this by shifting the values left by $\frac{(x_1 + x_2)}{2}$.

- $\tilde{x}_1 = x_1 - \frac{(x_1 + x_2)}{2} = +\frac{(x_1 - x_2)}{2}$
- $\tilde{x}_2 = x_2 - \frac{(x_1 + x_2)}{2} = -\frac{(x_1 - x_2)}{2}$

Because we have only shifted left, the m_1, m_2, y_1 and y_2 values have not changed. We can then write equations for our modified function $\tilde{f}$ and $\frac{d\tilde{f}}{d\tilde{x}}$ in our shifted coordinated system as follows.

1. $\tilde{f}(\tilde{x}_1) = \tilde{a} + \tilde{b}\tilde{x}_1 + \tilde{c}\tilde{x}_1^2 + \tilde{d}\tilde{x}_1^3 = y_1$
2. $\tilde{f}(\tilde{x}_2) = \tilde{a} + \tilde{b}\tilde{x}_2 + \tilde{c}\tilde{x}_2^2 + \tilde{d}\tilde{x}_2^3 = y_2$
3. $\frac{d\tilde{f}(\tilde{x})}{d\tilde{x}}(\tilde{x}_1) = \tilde{b} + 2\tilde{c}\tilde{x}_1 + 3\tilde{d}\tilde{x}_1^2 = m_1$
4. $\frac{d\tilde{f}(\tilde{x})}{d\tilde{x}}(\tilde{x}_2) = \tilde{b} + 2\tilde{c}\tilde{x}_2 + 3\tilde{d}\tilde{x}_2^2 = m_2$

To simplify the equations, in our shifted coordinates define $k = +\frac{(x_1 - x_2)}{2}$, which means that $\tilde{x}_1 = k$ and $\tilde{x}_2 = -k$. Thus we can write the constraint equations as

1. $\tilde{a} + \tilde{b}k + \tilde{c}k^2 + \tilde{d}k^3 = y_1$
2. $\tilde{a} - \tilde{b}k + \tilde{c}k^2 - \tilde{d}k^3 = y_2$
3. $\tilde{b} + 2\tilde{c}k + 3\tilde{d}k^2 = m_1$
4. $\tilde{b} - 2\tilde{c}k + 3\tilde{d}k^2 = m_2$

To solve, perform the following algebraic steps.

1. $\tilde{a} + \tilde{c}k^2 = \frac{(y_1 + y_2)}{2}$ (Equation 1 + Equation 2, then divide by 2)
2. $\tilde{b} + \tilde{d}k^2 = \frac{(y_1 - y_2)}{2k}$ (Equation 1 – Equation 2, then divide by $2k$)
3. $\tilde{c} = \frac{(m_1 - m_2)}{4k}$ (Equation 3 – Equation 4, then divide by $4k$)
4. $\tilde{b} + 3\tilde{d}k^2 = \frac{(m_1 + m_2)}{2}$ (Equation 3 + Equation 4, then divide by 2)

Then do the following.

1. $\tilde{a} = \frac{(y_1 + y_2)}{2} - \frac{k(m_1 - m_2)}{4}$ (Equations 1 $-k^2$* Equation 3)
2. $\tilde{b} = \frac{3(y_1 - y_2)}{4k} - \frac{(m_1 + m_2)}{4}$ (3 * Equation 2 – Equation 4, then divide by 2)
3. $\tilde{c} = \frac{(m_1 - m_2)}{4k}$ (Equation 3)
4. $\tilde{d} = \frac{(m_1 + m_2)}{4k^2} - \frac{(y_1 - y_2)}{4k^3}$ (Equation 4 – Equation 2, Then divide by $2k^2$)

We now have our coefficients in terms of k, so substitute back in $k = \frac{(x_1 - x_2)}{2}$ to get

1. $\tilde{a} = \frac{(y_1 + y_2)}{2} - \frac{(x_1 - x_2)(m_1 - m_2)}{8}$
2. $\tilde{b} = \frac{3}{2}\frac{(y_1 - y_2)}{(x_1 - x_2)} - \frac{(m_1 + m_2)}{4}$
3. $\tilde{c} = \frac{(m_1 - m_2)}{2(x_1 - x_2)}$
4. $\tilde{d} = \frac{(m_1 + m_2)}{(x_1 - x_2)^2} - \frac{2(y_1 - y_2)}{(x_1 - x_2)^3}$

We now have a cubic function on our modified coordinates of the form

$$\tilde{f}(\tilde{x}) = \tilde{a} + \tilde{b}\tilde{x} + \tilde{c}\tilde{x}^2 + \tilde{d}\tilde{x}^3$$

Now write this in terms of x instead by substituting $\tilde{x} = x - \frac{(x_1 - x_2)}{2}$

$$\tilde{f}\left(x - \left(\frac{x_1 + x_2}{2}\right)\right) = \tilde{a} + \tilde{b}\left[x - \left(\frac{x_1 + x_2}{2}\right)\right] + \tilde{c}\left[x - \left(\frac{x_1 + x_2}{2}\right)\right]^2 + \tilde{d}\left[x - \left(\frac{x_1 + x_2}{2}\right)\right]^3$$

Now transform this back to our original coordinates in x as follows.

$$f(x) = \tilde{a} + \tilde{b}x - \tilde{b}\left(\frac{x_1 + x_2}{2}\right) + \tilde{c}x^2 - 2\tilde{c}\left(\frac{x_1 + x_2}{2}\right)x + \tilde{c}\left(\frac{x_1 + x_2}{2}\right)^2 + \tilde{d}x^3 - 3\tilde{d}x^2\left(\frac{x_1 + x_2}{2}\right) + 3\tilde{d}x\left(\frac{x_1 + x_2}{2}\right)^2 - \tilde{d}\left(\frac{x_1 + x_2}{2}\right)^3$$

Group terms

$$\begin{aligned} f(x) = \quad & + \left[\tilde{a} - \tilde{b}\left(\frac{x_1+x_2}{2}\right) + \tilde{c}\left(\frac{x_1+x_2}{2}\right)^2 - \tilde{d}\left(\frac{x_1+x_2}{2}\right)^3\right] \\ & + \left[\tilde{b} - 2\tilde{c}\left(\frac{x_1+x_2}{2}\right) + 3\tilde{d}\left(\frac{x_1+x_2}{2}\right)^2\right]x \\ & + \left[\tilde{c} - 3\tilde{d}\left(\frac{x_1+x_2}{2}\right)\right]x^2 \\ & + [\tilde{d}]x^3 \end{aligned}$$

Thus the coefficients in our original coordinates are as follows.

- $a = \left[\tilde{a} - \tilde{b}\left(\frac{x_1+x_2}{2}\right) + \tilde{c}\left(\frac{x_1+x_2}{2}\right)^2 - \tilde{d}\left(\frac{x_1+x_2}{2}\right)^3\right]$
- $b = \left[\tilde{b} - 2\tilde{c}\left(\frac{x_1+x_2}{2}\right) + 3\tilde{d}\left(\frac{x_1+x_2}{2}\right)^2\right]$
- $c = \left[\tilde{c} - 3\tilde{d}\left(\frac{x_1+x_2}{2}\right)\right]$
- $d = [\tilde{d}]$

What happens if we assume that $x_0 = \frac{x_1+x_2}{2}$. This means that $x_0 - x_1 = -(x_0 - x_2)$
The coefficients that match this results are as follows.

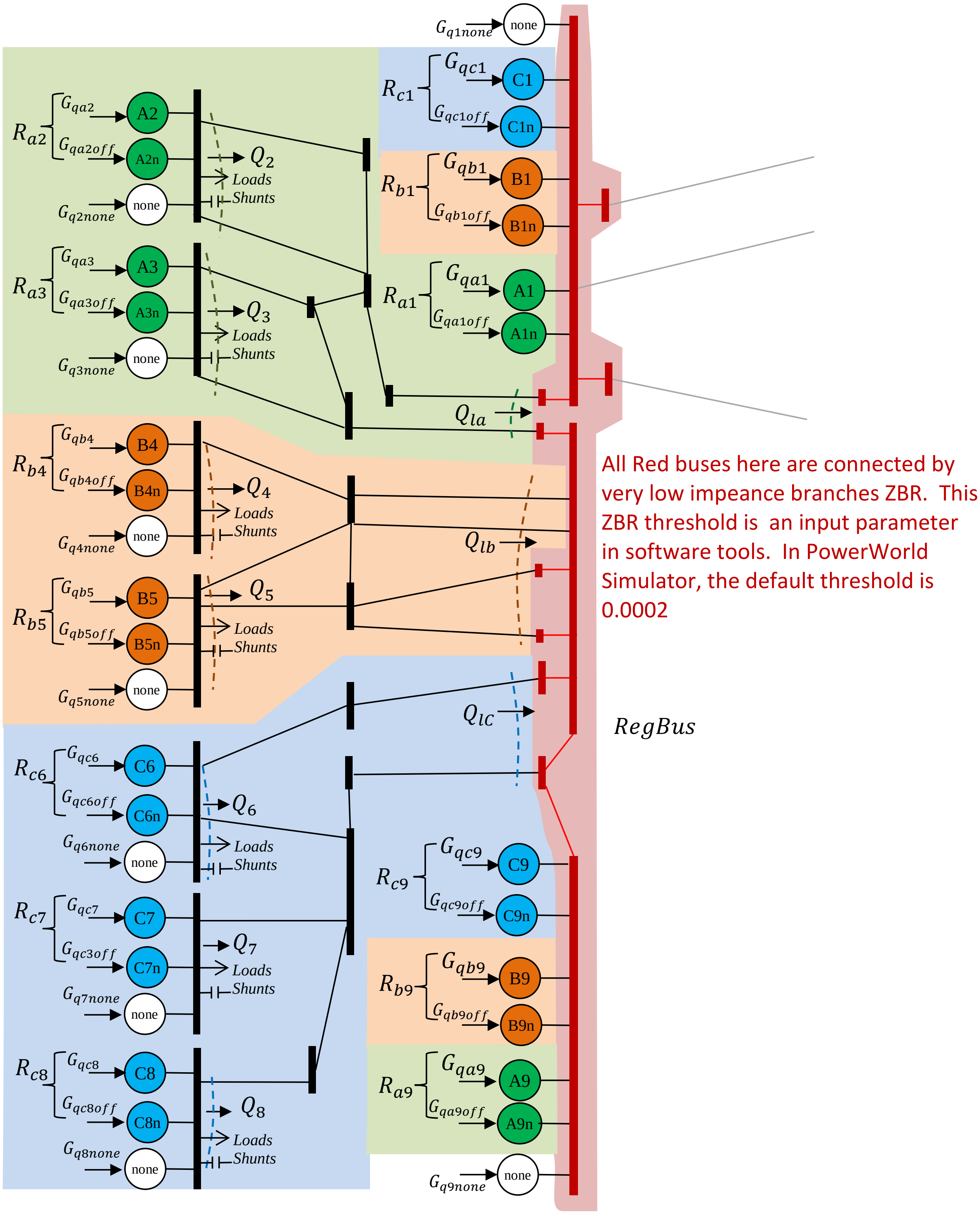
$$m_1 + m_2 = \frac{(y_0-y_1)}{(x_0-x_1)} + \frac{(y_0-y_2)}{(x_0-x_2)} = \frac{(y_0-y_1)}{\left(\frac{x_1+x_2}{2}-x_1\right)} + \frac{(y_0-y_2)}{\left(\frac{x_1+x_2}{2}-x_2\right)} = \frac{(y_0-y_1)}{\left(-\frac{x_1-x_2}{2}\right)} + \frac{(y_0-y_2)}{\left(\frac{x_1-x_2}{2}\right)} = 2\frac{(y_1-y_2)}{(x_1-x_2)}$$

$$m_1 - m_2 = \frac{(y_0-y_1)}{(x_0-x_1)} - \frac{(y_0-y_2)}{(x_0-x_2)} = \frac{(y_0-y_1)}{\left(\frac{x_1+x_2}{2}-x_1\right)} - \frac{(y_0-y_2)}{\left(\frac{x_1+x_2}{2}-x_2\right)} = \frac{(y_0-y_1)}{\left(-\frac{x_1-x_2}{2}\right)} - \frac{(y_0-y_2)}{\left(\frac{x_1-x_2}{2}\right)} = -2\frac{(2y_0-y_1-y_2)}{(x_1-x_2)}$$

This means that the coefficients result in a quadratic equation instead

- $\tilde{a} = \frac{(y_1+y_2)}{2} + \frac{(x_1-x_2)2\frac{(2y_0-y_1-y_2)}{(x_1-x_2)}}{8} = \frac{(y_1+y_2)}{2} + \frac{(2y_0-y_1-y_2)}{4} = \frac{(2y_0+y_1+y_2)}{4}$
- $\tilde{b} = \frac{3}{2}\frac{(y_1-y_2)}{(x_1-x_2)} - \frac{2\frac{(y_1-y_2)}{(x_1-x_2)}}{4} = \frac{(y_1-y_2)}{(x_1-x_2)}$
- $\tilde{c} = \frac{-2\frac{(2y_0-y_1-y_2)}{(x_1-x_2)}}{2(x_1-x_2)} = -\frac{(2y_0-y_1-y_2)}{(x_1-x_2)^2}$
- $\tilde{d} = \frac{2\frac{(y_1-y_2)}{(x_1-x_2)}}{(x_1-x_2)^2} - \frac{2(y_1-y_2)}{(x_1-x_2)^3} = 0$

3. Generalized Voltage Droop Control



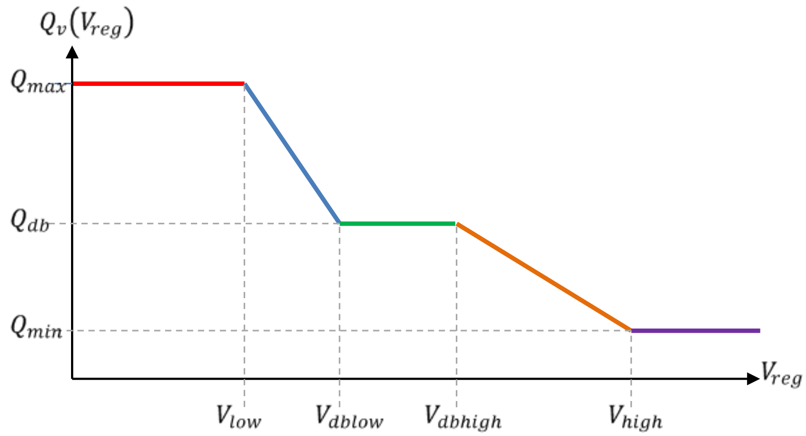
Assume we have three VoltageDroopControl groups which all regulate the same bus (RegBus). Classify the generators at each bus into one of three groups

- Part of VoltageDroopControl, AVR = YES (and presently not flagged as stuck at a Minimum or Maximum Mvar output). G_{qa1}, G_{qa2}, G_{qa3}, G_{qa9} G_{qb1}, G_{qb4}, G_{qb5}, G_{qb9}, G_{qc1}, G_{qc6}, G_{qc7}, G_{qc8}, G_{qc9} represent generators that are part of the respective VoltageDroopControl, and are also presently participating in AVR control.
- Part of VoltageDroopControl, AVR = NO (or presently flagged as stuck at a Minimum or Maximum Mvar output). G_{qa1off}, G_{qa2off}, G_{qa3off}, G_{qa9off}, G_{qb1off}, G_{qb4off}, G_{qb5off}, G_{qb9off}, G_{qc1off}, G_{qc6off}, G_{qc7off}, G_{qc8off}, G_{qc9off} represent generators that are part of the respective VoltageDroopControl, but are NOT presently participating AVR control and thus will act as a fixed Mvar.
- Not part of VoltageDroopControl. G_{q1none}, G_{q2none}, G_{q3none}, G_{q4none}, G_{q5none}, G_{q6none}, G_{q7none}, G_{q8none}, G_{q9none} represent generators that are not on control and also NOT assigned to be part of the VoltageDroopControl regulating the RegBus.

The outputs of the generators are expressed as G_{qa1}, G_{qc1}, G_{qc1}, etc.. These are not part of the power flow equations when the generators are operating between their minimum and maximum Mvar outputs, but they will be used during the derivation of the equations below. We write them as the variable G to distinguish them from the Q variables in the remainder of the section. All the Q quantities below represent quantities that are a function of voltage and angle (flows on AC branches, Load, Shunts values) and thus these Q variables are never solution variables, but instead functions that require calculation and will effect the Jacobian matrix calculations.

Also, within the software, buses that are connected to one another by very small impedance branches can cause numerical difficulty. If there is such a group of buses at the regulated bus these will be grouped and equations slightly modified within this group of buses.

Each of the three VoltageDroopControls has a QV characteristic curve which is a function of the RegBus voltage V_{reg} and the purpose of the curve is to enforce that the total reactive flow going into the RegBus from the respective VoltageDroopControl is equal to this value. The curve will have the following shape and is discussed in more detail shortly.



In this example, we have three such VoltageDroopcontrols and they will enforce the following equations.

$$Q_{va}(V_{reg}) = Q_{la} + G_{qa1} + G_{qa1off} + G_{qa9} + G_{qa9off}$$
$$Q_{vb}(V_{reg}) = Q_{lb} + G_{qb1} + G_{qb1off} + G_{qb9} + G_{qb9off}$$
$$Q_{vc}(V_{reg}) = Q_{lc} + G_{qc1} + G_{qc1off} + G_{qc9} + G_{qc9off}$$

Q_{la}, Q_{lb}, and Q_{lc} represent the summation of reactive power flow arriving at any bus within the group of buses connected to the RegBus by very low impedance lines on all AC branches that connect to the generators in the respective VoltageDroopControl.

G_{qa1}, G_{qa1off}, G_{qa9}, G_{qa9off}, G_{qb1}, G_{qb1off}, G_{qb9}, G_{qb9off}, G_{qc1}, G_{qc1off}, G_{qc9}, G_{qc9off} represent the Mvar output at all generators assigned to a VoltageDroopControl at any bus inside the group of buses connected by very low impedance lines. We include the contributions of both generator with AVR=YES and NO, so the generator must only be assigned to the Voltage Droop Control.

Shortly we will also use the values Q_1, Q_2, Q_3, Q_4, Q_5, Q_6, Q_7, Q_8, and Q_9 which represent the summation of branch flows leaving the respective bus summed with loads, shunts, DC lines, and other non generator devices.

$$Q_2 = Q_{2load} + Q_{2shunt} + \sum_{\substack{k=dc\ lines \\ adjacent \\ to\ 2}} [Q_{dc\ device\ 2-k}] + \sum_{\substack{k=Nodes \\ adjacent \\ to\ 2}} [Q_{branch\ 2-k}]$$

The R_{a1}, R_{a2}, R_{a3}, R_{a9}, R_{b1}, R_{b4}, R_{b5}, R_{b9}, R_{c1}, R_{c6}, R_{c7}, R_{c8} and R_{c9} represent the remote regulation factors used to represent how Mvars are shared between buses in each of the three VoltageDroopControl objects. Normally these factors are input parameters from the user, however they can also be specified to be equal to a generator Maximum Mvar output subtracted from the Minimum Mvar output (when using respecive options). To make notation easier, we will define the normalized factors in each group as follows.

$$K_{a1} = \frac{R_{a1}}{R_{a1}+R_{a2}+R_{a3}+R_{a9}} \qquad K_{a2} = \frac{R_{a2}}{R_{a1}+R_{a2}+R_{a3}+R_{a9}}$$
$$K_{a3} = \frac{R_{a3}}{R_{a1}+R_{a2}+R_{a3}+R_{a9}} \qquad K_{a9} = \frac{R_{a9}}{R_{a1}+R_{a2}+R_{a3}+R_{a9}}$$
$$K_{b1} = \frac{R_{b1}}{R_{b1}+R_{b4}+R_{b5}+R_{b9}} \qquad K_{b4} = \frac{R_{b4}}{R_{b1}+R_{b4}+R_{b5}+R_{b9}}$$
$$K_{b5} = \frac{R_{b5}}{R_{b1}+R_{b4}+R_{b5}+R_{b9}} \qquad K_{b9} = \frac{R_{b9}}{R_{b1}+R_{b4}+R_{b5}+R_{b9}}$$
$$K_{c1} = \frac{R_{c1}}{R_{c1}+R_{c6}+R_{c7}+R_{c8}+R_{c9}} \qquad K_{c6} = \frac{R_{c6}}{R_{c1}+R_{c6}+R_{c7}+R_{c8}+R_{c9}}$$
$$K_{c7} = \frac{R_{c7}}{R_{c1}+R_{c6}+R_{c7}+R_{c8}+R_{c9}} \qquad K_{c8} = \frac{R_{c8}}{R_{c1}+R_{c6}+R_{c7}+R_{c8}+R_{c9}} \qquad K_{c9} = \frac{R_{c9}}{R_{c1}+R_{c6}+R_{c7}+R_{c8}+R_{c9}}$$

Also, these factors are calculated so that as generators at a bus hit their Mvar limits, then those regulation factors are no longer included in the calculation of the K values.

To integrate these into the power flow equations, we will continue to use the same *real* power equations at each node. The reactive/voltage equation at the RegBus and at each bus that has a generator in any of the three VoltageDroopControl will be replaced as follows.

3.1. Regulated Bus Equation

At the RegBus, the equation is simply that the summation of the QV characteristics. This is written as

$$+Q_{va} + Q_{vb} + Q_{vc} = \begin{bmatrix} +Q_{la} + G_{qa1} + G_{qa1off} + G_{qa9} + G_{qa9off} \\ +Q_{lb} + G_{qb1} + G_{qb1off} + G_{qb9} + G_{qb9off} \\ +Q_{lc} + G_{qc1} + G_{qc1off} + G_{qc9} + G_{qc9off} \end{bmatrix}$$

Now, group terms slightly

$$+Q_{va} + Q_{vb} + Q_{vc} = \begin{bmatrix} +Q_{la} + Q_{lb} + Q_{lc} \\ +G_{qa1} + G_{qa1off} + G_{qb1} + G_{qb1off} + G_{qc1} + G_{qc1off} \\ +G_{qa9} + G_{qa9off} + G_{qb9} + G_{qb9off} + G_{qc9} + G_{qc9off} \end{bmatrix}$$

The individual generator Mvar outputs are not part of our solution variables. We can write those as follows to get them in terms of either fixed generator Mvar outputs (G_{q1none}, G_{q9none}) and the summation variables discussed above (Q_1, Q_9).

$$+G_{qa1} + G_{qa1off} + G_{qb1} + G_{qb1off} + G_{qc1} + G_{qc1off} = +Q_1 - G_{q1none}$$
$$+G_{qa9} + G_{qa9off} + G_{qb9} + G_{qb9off} + G_{qc9} + G_{qc9off} = +Q_9 - G_{q9none}$$

Making these substitutions you get

$$+Q_{va} + Q_{vb} + Q_{vc} = +Q_{la} + Q_{lb} + Q_{lc} + Q_1 - G_{q1none} + Q_9 - G_{q9none}$$

This gives us our mismatch equation of

$$+(-Q_1 + G_{q1none}) + (-Q_9 + G_{q9none}) + (Q_{va} - Q_{la}) + (Q_{vb} - Q_{lb}) + (Q_{vc} - Q_{lc}) = 0$$

We also need to be careful when a situation occurs that ALL generators within the VoltageDroopControl are hitting a Mvar limit. In that situation the value $(+Q_{va} - Q_{la})$ must be replaced with the fixed Mvar output of the generators at the regulated bus which are in the VoltageDroopControl "A". Thus if the generators in VoltageDroopControl A are all at Mvar limits, then the equation becomes the following.

$$(-Q_1 + G_{q1none}) + (-Q_9 + G_{q9none}) + (G_{qa1off} + G_{qa9off}) + (Q_{vb} - Q_{lb}) + (Q_{vc} - Q_{lc}) = 0$$

Actually, this concept extends more generally to a situation where a particular VoltageDroopControl does not have ANY generators connects to the RegBus (or any of it's low-impedance connected neighbors). For example, if we assume that VoltageDroopControl B and C did not have ANY generation at either bus 1 or 9 in this example, then we can simply the regulated bus equation to.

$$(-Q_1 + G_{q1none}) + (-Q_9 + G_{q9none}) + (Q_{va} - Q_{la}) = 0$$

If we assume that no generators in the this example are connected to either bus 1 or bus 9, then the regulated bus equation defaults all the way back to its original equation of simply.

$$(-Q_1 + G_{q1none}) = 0$$

3.2. Remotely Regulating Bus Equation

At each remotely regulating generator bus, we have an equation to enforce the ratios of reactive flows

$$-G_{qa2} + K_{a2}\left(G_{qa2} + G_{qa3} + G_{qa1} + G_{qa9}\right) = 0$$

As mentioned earlier though, the generator Mvar outputs are not part of the power flow solution variables, so we must write these in terms of the AC branch flow variables using the following relationships. The relationship at the remotely regulating buses is simply the summations of all devices at that bus except the generators on control.

$$G_{qa2} = Q_2 - G_{qa2off} - G_{q2none}$$
$$G_{qa3} = Q_3 - G_{qa3off} - G_{q3none}$$

The relationship for any generators at the regulated bus (or those connected by very low impedance branches) can be expressed using the QV characteristic with the summation of branch flows coming in from the VoltageDroopControl subtracted.

$$G_{qa1} + Q_{ga9} = Q_{va} - Q_{la} - Q_{a1off} - Q_{a9off}$$

Thus we can substitute in the Q to get

$$-G_{qa2} + K_{a2}\begin{pmatrix} +G_{qa2} \\ +G_{qa3} \\ +G_{qa1} + G_{qa9} \end{pmatrix} = -Q_2 + G_{q2none} + G_{qa2off} + K_{a2}\begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} \\ +Q_3 - G_{qa3off} - G_{q3none} \\ +Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} \end{pmatrix} = 0$$

Again, it means that there is a some nice structure because the value Q_2 is something that's already being calculated at all normal PQ buses anyway.

3.3. Remotely Regulated Bus connected to RegBus by low impedance branches

There also need to be some special treatment of buses the are not the regulated bus, but are within the group of buses connected to the RegBus by very low impedance lines (This is bus 9 in this example). At this bus we must write a summation of generator balancing across the VoltageDroopControls.

$$\begin{bmatrix} -G_{qa9} + K_{a9}\begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} \\ +Q_3 - G_{qa3off} - G_{q3none} \\ +Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} \end{pmatrix} \\ -G_{qb9} + K_{b9}\begin{pmatrix} +Q_4 - G_{qb4off} - G_{q4none} \\ +Q_5 - G_{qb5off} - G_{q5none} \\ +Q_{vb} - Q_{lb} - G_{qb1off} - G_{qb9off} \end{pmatrix} \\ -G_{qc9} + K_{c9}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} \\ +Q_7 - G_{qc7off} - G_{q7none} \\ +Q_8 - G_{qc8off} - G_{q8none} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} \end{pmatrix} \end{bmatrix} = 0$$

This then has a summation of $\left(-G_{qa9} - G_{qb9} - G_{qc9}\right)$ which needs to be replaced with.

$$\left(-G_{qa9} - G_{qb9} - G_{qc9}\right) = -Q_9 + G_{q9none} + G_{ga9off} + G_{gb9off} + G_{gc9off}$$

Thus the total equation at bus 9 can be written as.

$$-Q_9 + G_{q9none} + \begin{bmatrix} +G_{ga9off} + K_{a9}\begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} \\ +Q_3 - G_{qa3off} - G_{q3none} \\ +Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} \end{pmatrix} \\ +G_{gb9off} + K_{b9}\begin{pmatrix} +Q_4 - G_{qb4off} - G_{q4none} \\ +Q_5 - G_{qb5off} - G_{q5none} \\ +Q_{vb} - Q_{lb} - G_{qb1off} - G_{qb9off} \end{pmatrix} \\ +G_{gc9off} + K_{c9}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} \\ +Q_7 - G_{qc7off} - G_{q7none} \\ +Q_8 - G_{qc8off} - G_{q8none} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} \end{pmatrix} \end{bmatrix} = 0$$

3.4. Summary of Q Equations

Thus in our example, we have the following Q equations at the respective buses

[1] $\left(-Q_1 + G_{q1none}\right) + \left(-Q_9 + G_{q9none}\right) + (Q_{va} - Q_{la}) + (Q_{vb} - Q_{lb}) + (Q_{vc} - Q_{lc}) = 0$

[2] $-Q_2 + G_{q2none} + G_{qa2off} + K_{a2}\begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} \\ +Q_3 - G_{qa3off} - G_{q3none} \\ +Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} \end{pmatrix} = 0$

[3] $-Q_3 + G_{q3none} + G_{qa3off} + K_{a3}\begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} \\ +Q_3 - G_{qa3off} - G_{q3none} \\ +Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} \end{pmatrix} = 0$

[4] $-Q_4 + G_{q4none} + G_{qb4off} + K_{b4}\begin{pmatrix} +Q_4 - G_{qb4off} - G_{q4none} \\ +Q_5 - G_{qb5off} - G_{q5none} \\ +Q_{vb} - Q_{lb} - G_{qb1off} - G_{qb9off} \end{pmatrix} = 0$

[5] $-Q_5 + G_{q5none} + G_{qb5off} + K_{b5}\begin{pmatrix} +Q_4 - G_{qb4off} - G_{q4none} \\ +Q_5 - G_{qb5off} - G_{q5none} \\ +Q_{vb} - Q_{lb} - G_{qb1off} - G_{qb9off} \end{pmatrix} = 0$

[6] $-Q_6 + G_{q6none} + G_{qc6off} + K_{c6}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} \\ +Q_7 - G_{qc7off} - G_{q7none} \\ +Q_8 - G_{qc8off} - G_{q8none} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} \end{pmatrix} = 0$

[7] $-Q_7 + G_{q7none} + G_{qc7off} + K_{c7}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} \\ +Q_7 - G_{qc7off} - G_{q7none} \\ +Q_8 - G_{qc8off} - G_{q8none} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} \end{pmatrix} = 0$

[8] $-Q_8 + G_{q8none} + G_{qc8off} + K_{c8}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} \\ +Q_7 - G_{qc7off} - G_{q7none} \\ +Q_8 - G_{qc8off} - G_{q8none} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} \end{pmatrix} = 0$

[9] $-Q_9 + G_{q9none} + \begin{bmatrix} +G_{ga9off} + K_{a9}\begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} \\ +Q_3 - G_{qa3off} - G_{q3none} \\ +Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} \end{pmatrix} \\ +G_{gb9off} + K_{b9}\begin{pmatrix} +Q_4 - G_{qb4off} - G_{q4none} \\ +Q_5 - G_{qb5off} - G_{q5none} \\ +Q_{vb} - Q_{lb} - G_{qb1off} - G_{qb9off} \end{pmatrix} \\ +G_{gc9off} + K_{c9}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} \\ +Q_7 - G_{qc7off} - G_{q7none} \\ +Q_8 - G_{qc8off} - G_{q8none} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} \end{pmatrix} \end{bmatrix} = 0$

The red portions of these equations are the same as the normal mismatch equations for a PQ bus (ignoring generators), so we don't need to do anything in the code but allow those to be calculated. Then we simply add in the remaining portion in to the code and take appropriate derivatives.

Also, if we're using the option to keep all generators at the same point in their Mvar range, then these equations should be modified to shift var output equations to start at their minimum output and go up from there using modfied K factors which are proportional to the total (MvarMax – MvarMin) of each generator instead.

[1] $(-Q_1 + G_{q1none}) + (-Q_9 + G_{q9none}) + (Q_{va} - Q_{la}) + (Q_{vb} - Q_{lb}) + (Q_{vc} - Q_{lc}) = 0$

[2] $$-Q_2 + G_{q2none} + G_{qa2off} + G_{qa2min} + K_{a2}\begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} - G_{qa2min} \\ +Q_3 - G_{qa3off} - G_{q3none} - G_{qa3min} \\ +Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} - G_{qa1min} - G_{qa9min} \end{pmatrix} = 0$$

[3] $$-Q_3 + G_{q3none} + G_{qa3off} + G_{qa3min} + K_{a3}\begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} - G_{qa2min} \\ +Q_3 - G_{qa3off} - G_{q3none} - G_{qa3min} \\ +Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} - G_{qa1min} - G_{qa9min} \end{pmatrix} = 0$$

[4] $$-Q_4 + G_{q4none} + G_{qb4off} + G_{qb4min} + K_{b4}\begin{pmatrix} +Q_4 - G_{qb4off} - G_{q4none} - G_{qb4min} \\ +Q_5 - G_{qb5off} - G_{q5none} - G_{qb5min} \\ +Q_{vb} - Q_{lb} - G_{qb1off} - G_{qb9off} - G_{qb1min} - G_{qb9min} \end{pmatrix} = 0$$

[5] $$-Q_5 + G_{q5none} + G_{qb5off} + G_{qb5min} + K_{b5}\begin{pmatrix} +Q_4 - G_{qb4off} - G_{q4none} - G_{qb4min} \\ +Q_5 - G_{qb5off} - G_{q5none} - G_{qb5min} \\ +Q_{vb} - Q_{lb} - G_{qb1off} - G_{qb9off} - G_{qb1min} - G_{qb9min} \end{pmatrix} = 0$$

[6] $$-Q_6 + G_{q6none} + G_{qc6off} + G_{qc6min} + K_{c6}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} - G_{qc6min} \\ +Q_7 - G_{qc7off} - G_{q7none} - G_{qc7min} \\ +Q_8 - G_{qc8off} - G_{q8none} - G_{qc8min} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} - G_{qc1min} - G_{qc9min} \end{pmatrix} = 0$$

[7] $$-Q_7 + G_{q7none} + G_{qc7off} + G_{qc7min} + K_{c7}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} - G_{qc6min} \\ +Q_7 - G_{qc7off} - G_{q7none} - G_{qc7min} \\ +Q_8 - G_{qc8off} - G_{q8none} - G_{qc8min} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} - G_{qc1min} - G_{qc9min} \end{pmatrix} = 0$$

[8] $$-Q_8 + G_{q8none} + G_{qc8off} + G_{qc7min} + K_{c8}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} - G_{qc6min} \\ +Q_7 - G_{qc7off} - G_{q7none} - G_{qc7min} \\ +Q_8 - G_{qc8off} - G_{q8none} - G_{qc8min} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} - G_{qc1min} - G_{qc9min} \end{pmatrix} = 0$$

[9] $$-Q_9 + G_{q9none} + \begin{bmatrix} +G_{ga9off} + G_{qa9min} + K_{a9}\begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} - G_{qa2min} \\ +Q_3 - G_{qa3off} - G_{q3none} - G_{qa3min} \\ +Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} - G_{qa1min} - G_{qa9min} \end{pmatrix} \\ +G_{gb9off} + G_{qb9min} + K_{b9}\begin{pmatrix} +Q_4 - G_{qb4off} - G_{q4none} - G_{qb4min} \\ +Q_5 - G_{qb5off} - G_{q5none} - G_{qb5min} \\ +Q_{vb} - Q_{lb} - G_{qb1off} - G_{qb9off} - G_{qb1min} - G_{qb9min} \end{pmatrix} \\ +G_{gc9off} + G_{qc9min} + K_{c9}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} - G_{qc6min} \\ +Q_7 - G_{qc7off} - G_{q7none} - G_{qc7min} \\ +Q_8 - G_{qc8off} - G_{q8none} - G_{qc8min} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} - G_{qc1min} - G_{qc9min} \end{pmatrix} \end{bmatrix} = 0$$

3.5. Derivative Terms for Jacobian Matrix

The derivate of the Q equations is a straightforward, though adding many additional terms. It makes each bus a function of the neighbors of all other buses in the same voltage droop control as well as both terminals of any “arriving branch” in the voltage droop control.

[1] $\left(-\frac{dQ_1}{dx}\right)+\left(-\frac{dQ_9}{dx}\right)+\left(\frac{dQ_{va}}{dx}-\frac{dQ_{la}}{dx}\right)+\left(\frac{dQ_{vb}}{dx}-\frac{dQ_{lb}}{dx}\right)+\left(\frac{dQ_{vc}}{dx}-\frac{dQ_{lc}}{dx}\right)$

[2] $-\frac{dQ_2}{dx}+K_{a2}\left(+\frac{dQ_2}{dx}+\frac{dQ_3}{dx}+\frac{dQ_{va}}{dx}-\frac{dQ_{la}}{dx}\right)$

[3] $-\frac{dQ_3}{dx}+K_{a3}\left(+\frac{dQ_2}{dx}+\frac{dQ_3}{dx}+\frac{dQ_{va}}{dx}-\frac{dQ_{la}}{dx}\right)$

[4] $-\frac{dQ_4}{dx}+K_{b4}\left(+\frac{dQ_4}{dx}+\frac{dQ_5}{dx}+\frac{dQ_{vb}}{dx}-\frac{dQ_{lb}}{dx}\right)$

[5] $-\frac{dQ_5}{dx}+K_{b5}\left(+\frac{dQ_4}{dx}+\frac{dQ_5}{dx}+\frac{dQ_{vb}}{dx}-\frac{dQ_{lb}}{dx}\right)$

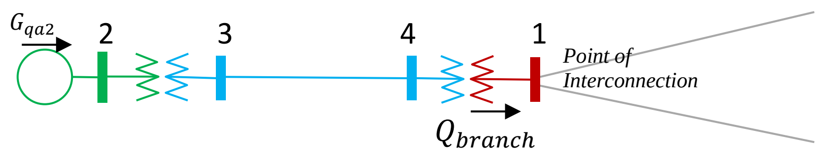
[6] $-\frac{dQ_6}{dx}+K_{c6}\left(+\frac{dQ_6}{dx}+\frac{dQ_7}{dx}+\frac{dQ_8}{dx}+\frac{dQ_{vc}}{dx}-\frac{dQ_{lc}}{dx}\right)$

[7] $-\frac{dQ_7}{dx}+K_{c7}\left(+\frac{dQ_6}{dx}+\frac{dQ_7}{dx}+\frac{dQ_8}{dx}+\frac{dQ_{vc}}{dx}-\frac{dQ_{lc}}{dx}\right)$

[8] $-\frac{dQ_8}{dx}+K_{c8}\left(+\frac{dQ_6}{dx}+\frac{dQ_7}{dx}+\frac{dQ_8}{dx}+\frac{dQ_{vc}}{dx}-\frac{dQ_{lc}}{dx}\right)$

[9] $-\frac{dQ_9}{dx}+\begin{bmatrix} +K_{a9}\left(+\frac{dQ_2}{dx}+\frac{dQ_3}{dx}+\frac{dQ_{va}}{dx}-\frac{dQ_{la}}{dx}\right) \\ +K_{b9}\left(+\frac{dQ_4}{dx}+\frac{dQ_5}{dx}+\frac{dQ_{vb}}{dx}-\frac{dQ_{lb}}{dx}\right) \\ +K_{c9}\left(+\frac{dQ_6}{dx}+\frac{dQ_7}{dx}+\frac{dQ_8}{dx}+\frac{dQ_{vc}}{dx}-\frac{dQ_{lc}}{dx}\right) \end{bmatrix}$

Also, you must be careful of the simple case where a single generator is remotely regulating a bus such as in the following figure.



This means we have only one remotely regulating bus and the value of $K_{a2}=1.0$, which causes the terms related to $\frac{dQ_2}{dx}$ to cancel out, so we have only an equation that is a function of voltage and Angle at bus 1 and 4.

[2] $\frac{dQ_v(V_1,Angle_1)}{dx}-\frac{dQ_{41}(V_1,Angle_1,V_4,Angle_4)}{dx}$

The problem with this is the Jacobian Matrix diagonal entries for the Q equation at bus 2 will be zero which would require the implementation of row pivoting in the numerical matrix calculations. In order to avoid this, in PowerWorld Simulator’s source code we slightly modify that Jacobian matrix to include extra terms in this example that set the sensitivity as 100 times smaller than the sensitivity of the Qbranch flow with respect to the regulated voltage.

$$\frac{dQEquation_{Bus2}}{dV_2}=0.01\frac{dQ_{41}(V_1,Angle_1,V_4,Angle_4)}{dV_1}$$
$$\frac{dQEquation_{Bus2}}{dAngle_2}=0.01\frac{dQ_{41}(V_1,Angle_1,V_4,Angle_4)}{dAngle_1}$$

In the software it will be the summation of all Q_{la} sensitivities with respect to the bus voltage/angle at bus in the group of buses in the ZBR bus.

This is 100 times smaller than the real entries needed there, so we would not expect there to be any troubles

3.6. How is Reactive Power Flow Equation being enforced at Regulated Bus

It may seem like the Mvar balance equation at the RegBus itself is not being enforced by these equations, but that is not correct. If you take the summation of all the Q-equations at the non-RegBus buses and subtract the Q equation at the RegBus you get the following. We have broken the parts of the equation at the RegBus (1) and the Bus 9 and moved them into other parts to result in the total summation below which must sum to zero.

$$(Q_1 - G_{q1none})$$

$$-Q_2 + G_{q2none} + G_{qa2off} + G_{qa2min} + K_{a2}\begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} - G_{qa2min} \\ +Q_3 - G_{qa3off} - G_{q3none} - G_{qa3min} \\ +Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} - G_{qa1min} - G_{qa9min} \end{pmatrix}$$

$$-Q_3 + G_{q3none} + G_{qa3off} + G_{qa3min} + K_{a3}\begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} - G_{qa2min} \\ +Q_3 - G_{qa3off} - G_{q3none} - G_{qa3min} \\ +Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} - G_{qa1min} - G_{qa9min} \end{pmatrix}$$

$$-Q_{va} + Q_{la} + G_{ga9off} + G_{qa9min} + K_{a9}\begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} - G_{qa2min} \\ +Q_3 - G_{qa3off} - G_{q3none} - G_{qa3min} \\ +Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} - G_{qa1min} - G_{qa9min} \end{pmatrix}$$

$$-Q_4 + G_{q4none} + G_{qb4off} + G_{qb4min} + K_{b4}\begin{pmatrix} +Q_4 - G_{qb4off} - G_{q4none} - G_{qb4min} \\ +Q_5 - G_{qb5off} - G_{q5none} - G_{qb5min} \\ +Q_{vb} - Q_{lb} - G_{qb1off} - G_{qb9off} - G_{qb1min} - G_{qb9min} \end{pmatrix}$$

$$-Q_5 + G_{q5none} + G_{qb5off} + G_{qb5min} + K_{b5}\begin{pmatrix} +Q_4 - G_{qb4off} - G_{q4none} - G_{qb4min} \\ +Q_5 - G_{qb5off} - G_{q5none} - G_{qb5min} \\ +Q_{vb} - Q_{lb} - G_{qb1off} - G_{qb9off} - G_{qb1min} - G_{qb9min} \end{pmatrix}$$

$$-Q_{vb} + Q_{lb} + G_{gb9off} + G_{qb9min} + K_{b9}\begin{pmatrix} +Q_4 - G_{qb4off} - G_{q4none} - G_{qb4min} \\ +Q_5 - G_{qb5off} - G_{q5none} - G_{qb5min} \\ +Q_{vb} - Q_{lb} - G_{qb1off} - G_{qb9off} - G_{qb1min} - G_{qb9min} \end{pmatrix}$$

$$-Q_6 + G_{q6none} + G_{qc6off} + G_{qc6min} + K_{c6}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} - G_{qc6min} \\ +Q_7 - G_{qc7off} - G_{q7none} - G_{qc7min} \\ +Q_8 - G_{qc8off} - G_{q8none} - G_{qc8min} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} - G_{qc1min} - G_{qc9min} \end{pmatrix}$$

$$-Q_7 + G_{q7none} + G_{qc7off} + G_{qc7min} + K_{c7}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} - G_{qc6min} \\ +Q_7 - G_{qc7off} - G_{q7none} - G_{qc7min} \\ +Q_8 - G_{qc8off} - G_{q8none} - G_{qc8min} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} - G_{qc1min} - G_{qc9min} \end{pmatrix}$$

$$-Q_8 + G_{q8none} + G_{qc8off} + G_{qc7min} + K_{c8}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} - G_{qc6min} \\ +Q_7 - G_{qc7off} - G_{q7none} - G_{qc7min} \\ +Q_8 - G_{qc8off} - G_{q8none} - G_{qc8min} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} - G_{qc1min} - G_{qc9min} \end{pmatrix}$$

$$-Q_9 + G_{q9none} - Q_{vc} + Q_{lc} + G_{gc9off} + G_{qc9min} + K_{c9}\begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} - G_{qc6min} \\ +Q_7 - G_{qc7off} - G_{q7none} - G_{qc7min} \\ +Q_8 - G_{qc8off} - G_{q8none} - G_{qc8min} \\ +Q_{vc} - Q_{lc} - G_{qc1off} - G_{qc9off} - G_{qc1min} - G_{qc9min} \end{pmatrix}$$

$$+Q_9 - G_{q9none} = 0$$

Now make the following definitions.

$$Q_A = \begin{pmatrix} +Q_2 - G_{qa2off} - G_{q2none} - G_{qa2min} \\ +Q_3 - G_{qa3off} - G_{q3none} - G_{qa3min} \\ +Q_{va} - Q_{la} - G_{qa9off} - G_{qa9min} \end{pmatrix}$$

$$Q_B = \begin{pmatrix} +Q_4 - G_{qb4off} - G_{q4none} - G_{qb4min} \\ +Q_5 - G_{qb5off} - G_{q5none} - G_{qb5min} \\ +Q_{vb} - Q_{lb} - G_{qb9off} - G_{qb9min} \end{pmatrix}$$

$$Q_C = \begin{pmatrix} +Q_6 - G_{qc6off} - G_{q6none} - G_{qc6min} \\ +Q_7 - G_{qc7off} - G_{q7none} - G_{qc7min} \\ +Q_8 - G_{qc8off} - G_{q8none} - G_{qc8min} \\ +Q_{vc} - Q_{lc} - G_{qc9off} - G_{qc9min} \end{pmatrix}$$

We can then rewrite the entire summation as

$$\begin{aligned} &(Q_1 - G_{q1none}) \\ &-Q_A + (K_{a2} + K_{a3} + K_{a9})(+Q_A - G_{qa1off} - G_{qa1min}) \\ &-Q_B + (K_{b4} + K_{b5} + K_{b9})(+Q_B - G_{qb1off} - G_{qb1min}) \\ -Q_9 + G_{q9none} + Q_9 - G_{q9none} &- Q_C + (K_{c6} + K_{c7} + K_{c8} + K_{c9})(+Q_C - G_{qc1off} - G_{qc1min}) \\ &= 0 \end{aligned}$$

The bus 9 values cancel out, and we can then flip the sign on the remaining equations and add extra terms to get.

$$\begin{aligned} &(-Q_1 + G_{q1none}) \\ +Q_A - (+Q_A - G_{qa1off} - G_{qa1min}) + (+Q_A - G_{qa1off} - G_{qa1min}) &- (K_{a2} + K_{a3} + K_{a9})(+Q_A - G_{qa1off} - G_{qa1min}) \\ +Q_B - (+Q_B - G_{qb1off} - G_{qb1min}) + (+Q_B - G_{qb1off} - G_{qb1min}) &- (K_{b4} + K_{b5} + K_{b9})(+Q_B - G_{qb1off} - G_{qb1min}) \\ +Q_C - (+Q_C - G_{qc1off} - G_{qc1min}) + (+Q_C - G_{qc1off} - G_{qc1min}) &- (K_{c6} + K_{c7} + K_{c8} + K_{c9})(+Q_C - G_{qc1off} - G_{qc1min}) \\ &= 0 \end{aligned}$$

This then can be written as.

$$\begin{aligned} &(-Q_1 + G_{q1none}) \\ +(+G_{qa1off} + G_{qa1min}) &+ (1 - (K_{a2} + K_{a3} + K_{a9}))(+Q_A - G_{qa1off} - G_{qa1min}) \\ +(+G_{qb1off} + G_{qb1min}) &+ (1 - (K_{b4} + K_{b5} + K_{b9}))(+Q_B - G_{qb1off} - G_{qb1min}) \\ +(+G_{qc1off} + G_{qc1min}) &+ (1 - (K_{c6} + K_{c7} + K_{c8} + K_{c9}))(+Q_C - G_{qc1off} - G_{qc1min}) \\ &= 0 \end{aligned}$$

At this point, remember that the summation of the K-factors for each VoltageDroopControl sums to 1.0 by definition, thus we can write this as

$$\begin{aligned} &(-Q_1 + G_{q1none}) \\ &+G_{qa1off} + G_{qa1min} + K_{a1}(+Q_A - G_{qa1off} - G_{qa1min}) \\ &+G_{qb1off} + G_{qb1min} + K_{b1}(+Q_B - G_{qb1off} - G_{qb1min}) \\ &+G_{qc1off} + G_{qc1min} + K_{c1}(+Q_C - G_{qc1off} - G_{qc1min}) \\ &= 0 \end{aligned}$$

Then realize that the Mvar sharing equations mean that

- $G_{qa1} = G_{qa1min} + K_{a1}(+Q_A - G_{qa1off} - G_{qa1min})$
- $G_{qb1} = G_{qb1min} + K_{b1}(+Q_B - G_{qb1off} - G_{qb1min})$
- $G_{qc1} = G_{qc1min} + K_{c1}(+Q_C - G_{qc1off} - G_{qc1min})$

This means that we can rewrite our equations summation as

$$-Q_1 + G_{q1none} + G_{qa1off} + G_{qa1} + G_{qb1off} + G_{qb1} + G_{qc1off} + G_{qc1} = 0$$

That summation is the Mvar mismatch equation at the RegBus. Thus it is being enforced because all the summation of all the other equations result it in being enforced.

3.7. Calculate of Generator Mvar after solution

The previous equations will work for solving for the voltages and angles in the case, but we then have to think about how to calculate the generator Mvar outputs (G_{qa1}) after the solution is obtained. For generators at the remotely regulating buses this is simply done by summing up the branch flows and applying the equation.

$$G_{qa2} = Q_2 - G_{qa2off} - G_{qa2none}$$

For any generators located at the regulated bus however, we must calculate them using the following equation.

$$G_{qa1} + G_{qa9} = Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off}$$

Then allocate in the $G_{qa1} + G_{qa9}$ according to their K factors across all other generators that are connected to a bus within the very low impedance group as follows.

$$G_{qa1} = G_{qa1min} + \frac{K_{a1}}{K_{a1} + K_{a9}}\left(Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} - G_{qa1min}\right)$$

$$G_{qa9} = G_{qa9min} + \frac{K_{a9}}{K_{a1} + K_{a9}}\left(Q_{va} - Q_{la} - G_{qa1off} - G_{qa9off} - G_{qa1min}\right)$$

Finally, all these equations give the total Mvar for generators at a bus with AVR=YES and those NOT stuck at a limit. The Mvars must then be allocated in the proportion specified by their K factors across all other generators at the bus.