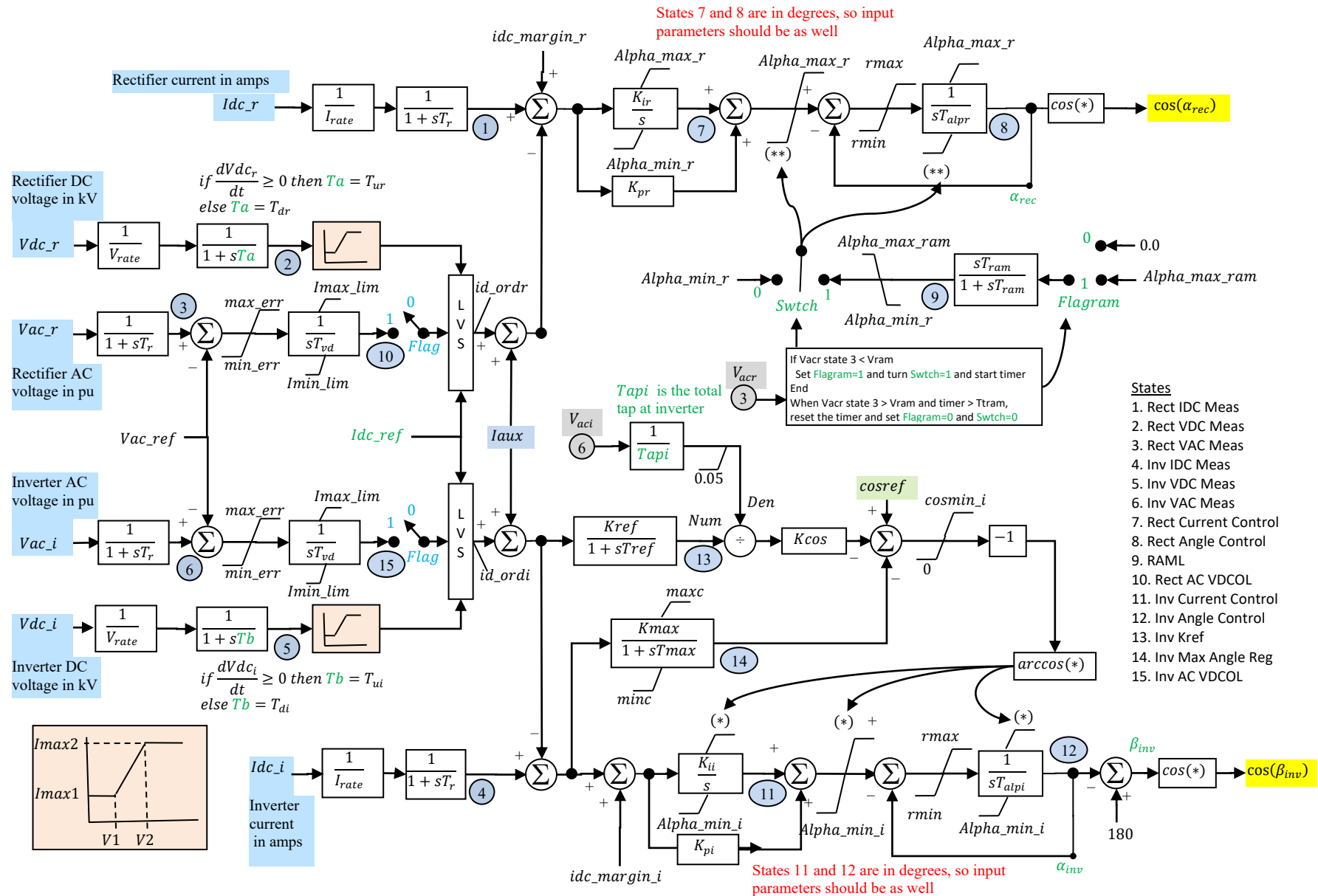


HVDC CHVDC2

Generic High Voltage DC Modelling



Note about writing this model to DYD files due to input Parameters *Vrate* and *Irate*

The DYD record does not include the *Vrate* and *Irate* input parameters that are used in both PowerWorld and also in DYR files. Instead the DYD record always assumes that the *Vrate* and *Irate* are calculated in a hard-coded manner from the DC converter records using

```
VrateDYD = dcc.vdc_base * dcc.nbridge  
IrateDYD = dcc.i_rate
```

To make a model that responds the same, when writing out the DYD record a conversion is done for several input parameters as follows.

- Per unit voltage input parameters *V1* and *V2* are multiplied $Vrate_PowerWorld/Vrate_DYD$
- Per unit current input parameters *Idc_margin_r*, *Idc_margin_i*, *Imax1*, and *Imax2* are multiplied by $Irate_PowerWorld/Irate_DYD$.
- Current gain parameters *Kir*, *Kpr*, *Kii*, *Kpi*, *Kcos*, *Kref*, and *Kmax* are multiplied by $Irate_DYD/Irate_PowerWorld$.

This converts the input parameters to be consistent with the voltage and current bases used in the DYD file.

Modeling of Commutation Failure

The following approximate method is used to emulate commutation failure, internal to the model. This is a very rough compromise because the actual dynamics and characteristics of commutation failure can only truly be modeled and studied in 3-phase EMT type simulations of the actual equipment.

Three additional parameters are needed in the dynamic model:

- *gamma_cf* – angle below which commutation failure is likely (default value = 10 degrees). Also note that if *gamma_cf* ≤ 0 then we ignore commutation failure emulation
- *Tcf* – minimum time duration that commutation failure is likely to last (default value = 0.034 seconds)
- *Vac_ucf* – voltage above which converter will recover from commutation failure (default value = 0.9)

The logic for emulating commutation failure is as follows.

Set variables internal to this model as follows during model initialization.

```
TimeOfCommutationFailure = 0 // stores the time at which commutation failure occurs  
InverterByPassed = false // this is a boolean indicating if the inverter is presently bypassed  
AlreadyDidCommutationFailure = false // Boolean indicating we just did the commutation failure
```

Then at the beginning of each time step perform the following logic to determine when to bypass and un-bypass the inverter. Note that after bypass has occurred and then been removed, the logic below will not bypass the inverter again until after the firing angle has recovered back above *gamma_cf*.

```
if (gamma_cf > 0) then // ignore this logic if Gamma_cf is zero or negative  
  if (not InverterByPassed) then  
    if (Gamma <= gamma_cf) and (not AlreadyDidCommutationFailure) then  
      // force it bypassed  
      InverterByPassed = true  
      AlreadyDidCommutationFailure = true  
      TimeOfCommutationFailure = PresentTime  
    elseif (Gamma > gamma_cf) and (AlreadyDidCommutationFailure) then  
      // allow commutation failure to occur again after angle recovers  
      AlreadyDidCommutationFailure = false  
    endif  
  elseif InverterByPassed then  
    if ((PresentTime - TimeOfCommutationFailure) >= Tcf) and (VacInverter >= Vac_ucf) then  
      // remove bypass because voltage has recovered  
      InverterByPassed = false  
    endif  
  endif  
endif
```

DC Converter Equations Revisited

Normally the rectifier and inverter equations are written in terms of the *firing angles* α and γ .

Rectifier Equations	Inverter Equations
$V_{dcr} = \frac{3\sqrt{2}N}{\pi} \left[\frac{D_{base}}{T_{ap}} \right] V_{ac} \cos(\alpha)$ $- N \left[\frac{3X_c}{\pi} + 2R_c \right] I_{dc} - N[V_{diode}]$ $E_{ac} = \left[\frac{D_{base}}{T_{ap}} \right] V_{ac}$ $\mu = \cos^{-1} \left[\cos \alpha - \frac{\sqrt{2}I_{dc}X_c}{E_{ac}} \right] - \alpha$	$V_{dci} = \frac{3\sqrt{2}N}{\pi} \left[\frac{D_{base}}{T_{ap}} \right] V_{ac} \cos(\gamma)$ $+ N \left[-\frac{3X_c}{\pi} + 2R_c \right] I_{dc} + N[V_{diode}]$ $E_{ac} = \left[\frac{D_{base}}{T_{ap}} \right] V_{ac}$ $\mu = \cos^{-1} \left[\cos \gamma - \frac{\sqrt{2}I_{dc}X_c}{E_{ac}} \right] - \gamma$

For transient stability purposes it is more convenient to write the inverter equations in terms of β . β is related to γ by $\beta = \mu + \gamma$. During the simulation we will need to flip back and forth between β and γ , thus express β as a function of γ and vice versa.

$$\beta = \cos^{-1} \left[\cos \gamma - \frac{\sqrt{2}I_{dc}X_c}{E_{ac}} \right] \quad \text{and} \quad \gamma = \cos^{-1} \left[\cos \beta + \frac{\sqrt{2}I_{dc}X_c}{E_{ac}} \right].$$

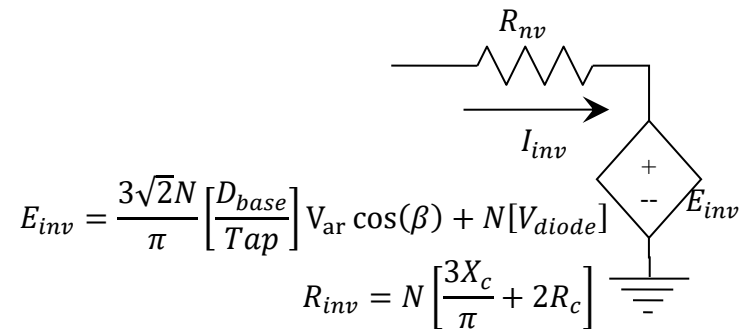
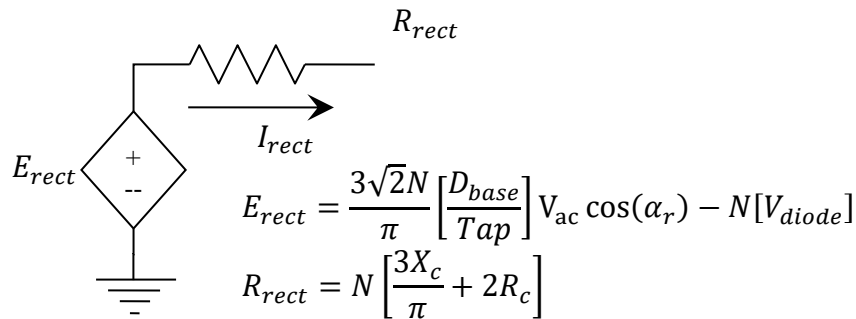
Now using the relationship $\cos \gamma = \cos \beta + \frac{\sqrt{2}I_{dc}X_c}{E_{ac}}$, rewrite the DC voltage equation for the inverter in terms of β .

$$V_{dci} = \frac{3\sqrt{2}N}{\pi} E_{ac} \left[\cos \beta + \frac{\sqrt{2}I_{dc}X_c}{E_{ac}} \right] + N \left[-\frac{3X_c}{\pi} + 2R_c \right] I_{dc} + N[V_{diode}]$$

$$V_{dci} = \frac{3\sqrt{2}N}{\pi} E_{ac} \cos \beta + N \left[\frac{6X_c}{\pi} I_{dc} \right] + N \left[-\frac{3X_c}{\pi} + 2R_c \right] I_{dc} + N[V_{diode}]$$

$$V_{dci} = \frac{3\sqrt{2}N}{\pi} E_{ac} \cos \beta + N \left[\frac{3X_c}{\pi} + 2R_c \right] I_{dc} + N[V_{diode}]$$

Using these equations, model the converter in the DC network equations as follows, with the extra constraint that the currents can NOT be negative.



Implementation of the numerical solution of the DC Network Dynamics

Implementation of the DC network equations is done in a manner similar to the multi-terminal DC line equations. In the normal network boundary equations and in Simulator's power flow engine, the DC converter control is assumed to be instantaneous. We assume that firing angle move instantaneously to bring DC currents instantaneously to a new operating point. For this model however, we will be removing these assumptions completely and modeling the dynamics of the firing angle control. It is convenient for numerical reasons to make the control output of the DC converter equal to either $\cos(\alpha)$ and $\cos(\beta)$ depending on whether the converter is acting as a rectifier or inverter. Here we will describe how this changes the numerical simulation of the multi-terminal DC line.

Inside the transient stability engine in Simulator, an explicit integration method is used. Thus the general process of solving the equations is to use a numerical integration time step to update all dynamic state variables and then another step to update all the algebraic variables such as the AC system voltage and angle (network boundary equations). The DC network simulation is added into this framework as follows.

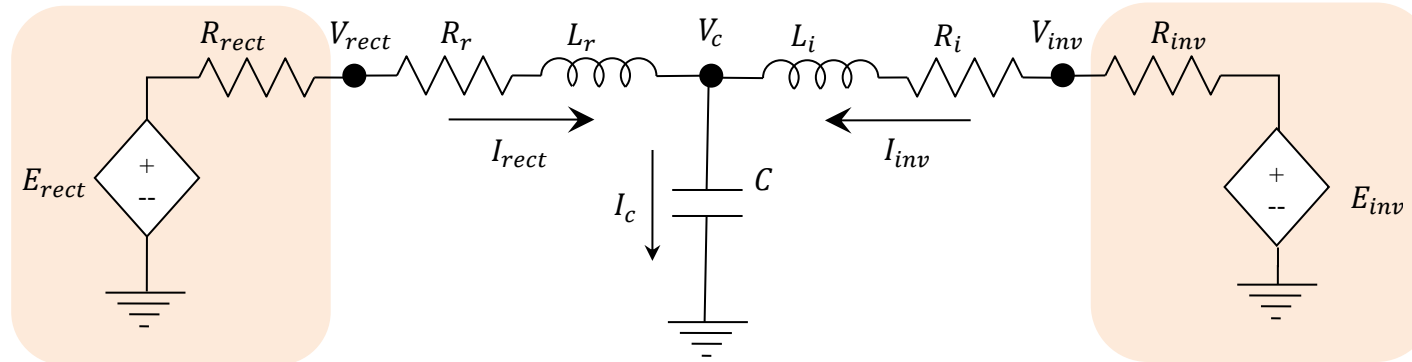
During each time-step of the numerical simulation of the DC Line, the following steps are taken

1. Using the standard routines within Simulator to calculate the new states for the dynamic model CHVDC2. This is done along with all the other thousands of dynamic models. --> Updated Variables are $\cos(\alpha)$ or $\cos(\beta)$ terms for the DC converters
2. Before solving all the other algebraic equations using the AC network boundary equations, take the new $\cos(\alpha)$ and $\cos(\beta)$ terms and use them to model a step change in the DC voltages seen by the DC network equations. Use numerical integration to simulate the change in DC voltages and DC currents on the transmission lines for this time-step. Note that this is the only place in the numerical routines where the DC system currents in the system change.
--> Updated Variables are the DC voltages and DC Currents
3. Finally, when the normal AC network boundary equations are being solved we must modify how the DC network equations are handled. Without the dynamic model for the MTDC line modeled, we assume that the DC currents respond instantaneously to a change in the DC voltages by an instantaneous change in the converter firing angles.

DC Network Equations

$$R_{rect} = N_{rect} \left[\frac{3X_{cr}}{\pi} + 2R_{cr} \right]$$

$$R_{inv} = N_{inv} \left[\frac{3X_{ci}}{\pi} + 2R_{ci} \right]$$



$$E_{rect} = \frac{3\sqrt{2}N_{rect}}{\pi} \left[\frac{D_{baser}}{T_{apr}} \right] V_{acr} \cos(\alpha) - N_{rect}[V_{diode}]$$

$$E_{inv} = \frac{3\sqrt{2}N_{inv}}{\pi} \left[\frac{D_{basei}}{T_{api}} \right] V_{aci} \cos(\beta) + N_{inv}[V_{diodei}]$$

$$R_r = \frac{\text{DC Line Resistance}}{2}$$

$$L_r = \frac{\text{DC Line Inductance}}{2} + L_{smoothr} + \frac{1.75N_{rect}X_{cr}}{2\pi f}$$

$$L_i = \frac{\text{DC Line Inductance}}{2} + L_{smoothi} + \frac{1.75N_{inv}X_{ci}}{2\pi f}$$

Following are AC per unit voltages from the AC network equations.

$$V_{acr}, V_{aci}$$

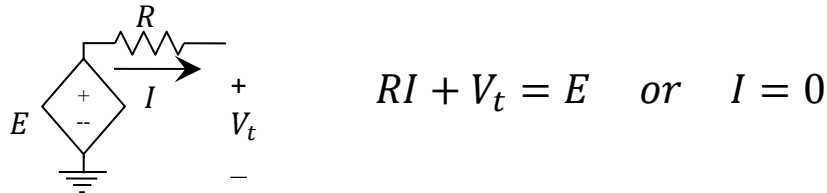
Following are outputs of the various dynamic DC converter models.

$$\cos(\beta), \cos(\alpha)$$

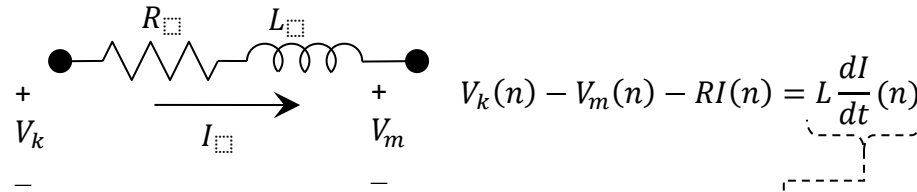
- The 1.75 factor is an approximation of the time-average value of commutating inductance due to overlap. See CIGRE-92 "DC Side Harmonics and Filtering in HVDC Transmission Systems" or M. P. Bahrman, K. J. Peterson and R. H. Lasseter, "DC system resonance analysis", IEEE Trans. PWRD, January 1987.
- Division by $2\pi f$ because X_c terms are given in Ohms assuming nominal frequency. Divide to convert Ohms to Henries.

Numerical Integration to model the change in DC currents and DC voltages after a change in the $\cos(\alpha)$ and $\cos(\beta)$ terms

DC Converter equations are as follows, where R and E were described earlier. The unknown variable associated with this equation is the DC current. Note that it's possible for this to result in a current that is impossible (negative current injected into DC network for a rectifier or positive injection for an inverter). In that case we use a different equation which sets the current to zero.



DC transmission line equations are as follows. The unknown variable associate with this equation is the DC line current.



Use the trapezoidal rule to write the $dI/dt(n)$ term

$$V_k(n) - V_m(n) - RI(n) = \frac{2L}{h} I(n) - \left[\frac{2L}{h} I(n-1) + L \frac{dI}{dt}(n-1) \right]$$

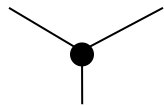
Use the standard relationship above to write the $LdI/dt(n-1)$ term

$$V_k(n) - V_m(n) - RI(n) = \frac{2L}{h} I(n) - \left[\frac{2L}{h} I(n-1) \right] - [V_k(n-1) - V_m(n-1) - RI(n-1)]$$

Finally, group the terms that are a function of values at time (n) on the left and time (n-1) on the right

$$V_k(n) - V_m(n) + \left(-R - \frac{2L}{h} \right) I(n) = \left(+R - \frac{2L}{h} \right) I(n-1) - V_k(n-1) + V_m(n-1)$$

DC Bus equation is just Kirchhoff's Current Law (KCL). The unknown variable associated with the equation is the DC bus voltage.



$$I_a + I_b + I_c + \dots = 0$$

Remember the trapezoidal integration rule:

$$\frac{\frac{dx}{dt}(n) + \frac{dx}{dt}(n-1)}{2} = \frac{x(n) - x(n-1)}{h}$$

which gives

$$\frac{dx}{dt}(n) = \frac{2}{h} x(n) - \left[\frac{2}{h} x(n-1) + \frac{dx}{dt}(n-1) \right]$$

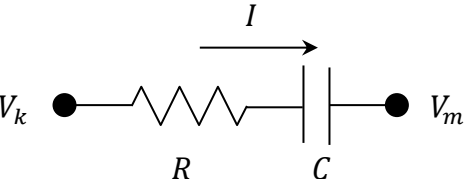
Where n = the present integer time step

h = the duration of the time step

$x(n)$ = value at present time

$x(n-1)$ = value at previous time step

Capacitor Equations.



$$I = C \frac{d(V_k - V_m - IR)}{dt}$$

$$I(n) = C \left[\frac{2}{h} [V_k(n) - V_m(n) - I(n)R] - \frac{2}{h} [V_k(n-1) - V_m(n-1) - I(n-1)R] - \frac{\frac{I(n-1)}{C}}{dt} \right]$$

$$I(n) + \frac{2RC}{h} I(n) - \frac{2C}{h} V_k(n) + \frac{2C}{h} V_m(n) = -\frac{2C}{h} V_k(n-1) + \frac{2C}{h} V_m(n-1) + \frac{2RC}{h} I(n-1) - I(n-1)$$

$$\left[1 + \frac{2RC}{h} \right] I(n) - \frac{2C}{h} V_k(n) + \frac{2C}{h} V_m(n) = \left[-1 + \frac{2RC}{h} \right] I(n-1) - \frac{2C}{h} V_k(n-1) + \frac{2C}{h} V_m(n-1)$$

Now Simplify this if $R = 0$ and $V_m = 0$ (ground)

$$I(n) - \frac{2C}{h} V_k(n) = -I(n-1) - \frac{2C}{h} V_k(n-1)$$

The following matrix equations are created for a numerical simulation of the DC network.

	Irect	Iinv	Icap	Vrect	Vinv	Vcap	X	B
LineRect	$(-R_r - 2L_r/h)$					-1	Irect _(n)	$(+R_r - 2L_r/h) \cdot \text{Irect}_{(n-1)} - \text{Vrect}_{(n-1)} + \text{Vcap}_{(n-1)}$
LineInv		$(-R_i - 2L_i/h)$				-1	Iinv _(n)	$(+R_i - 2L_i/h) \cdot \text{Iinv}_{(n-1)} - \text{Vinv}_{(n-1)} + \text{Vcap}_{(n-1)}$
Cap			+1			$-2C/h$	Icap _(n)	$-\text{Icap}_{(n-1)} - 2C/h \cdot \text{Vcap}_{(n-1)}$
Rect	Rrect			+1			Vrect _(n)	Erect
Inv		Rinv			+1		Vinv _(n)	Erect
KCLCap	+1	+1	-1			0	Vcap _(n)	0

$$\begin{bmatrix} \text{Irect}_{(n)} \\ \text{Iinv}_{(n)} \\ \text{Icap}_{(n)} \\ \text{Vrect}_{(n)} \\ \text{Vinv}_{(n)} \\ \text{Vcap}_{(n)} \end{bmatrix} = \begin{bmatrix} (+R_r - 2L_r/h) \cdot \text{Irect}_{(n-1)} - \text{Vrect}_{(n-1)} + \text{Vcap}_{(n-1)} \\ (+R_i - 2L_i/h) \cdot \text{Iinv}_{(n-1)} - \text{Vinv}_{(n-1)} + \text{Vcap}_{(n-1)} \\ -\text{Icap}_{(n-1)} - 2C/h \cdot \text{Vcap}_{(n-1)} \\ \text{Erect} \\ \text{Erect} \\ 0 \end{bmatrix}$$

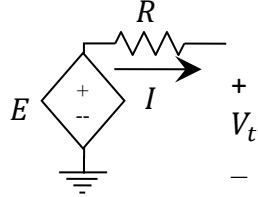
Simulator solves this set of equations by splitting the actual integration time-step used in the standard numerical integration and dividing it by 10. We set the h variable above to the time step divided by 10 then iterate this set of equations 10 times. When solving these equations, we initially assume that none of the currents end up the wrong sign. Then after each sub time step we check if the converter currents end up as the wrong sign. If converter currents have the wrong sign, we automatically change the equation for the offending converter to force that converter current to zero and redo this sub time step. We assume the converter current remains zero during the remainder of this time step and only allow it to back-off this limit during the following time step. I believe the current would never bounce around during a time step anyway, because this is fundamentally only a set of RL circuits so you will only get a first-order RL circuit exponential decay response toward the new steady state.

As an example, if the current at Celilo1 ended up the wrong sign, then its matrix equation would be rewritten as

LineRect	1					0	Irect _(n)	0
----------	---	--	--	--	--	---	----------------------	---

Solution of the algebraic change in DC voltages during the AC network boundary equation solution

DC Converter equations are as follows, where R and E were described earlier. Also note that the DC current is assumed constant so it is moved to the right side as a known quantity. The unknown variable associated with this equation is the voltage at the terminal.



$$V_{rect} = E_{rect} - R_{rect}I$$

Because the DC transmission line has a capacitor in the middle of the line, this means that V_c cannot change instantaneously. Thus for our DC line model we don't need to update V_{cap} . However, if the Capacitance $C = 0$, then the voltage in the middle of the line can change instantly.

$$V_c = V_{rect} - I_{rect}R_r - L_r \frac{dI_{rect}}{dt} = V_{inv} - I_{inv}R_i - L_i \frac{dI_{inv}}{dt}$$

However, because $C=0$, then $I_{inv} = -I_{rect}$, so let's just call that I .

$$V_{rect} - IR_r - \frac{L_r dI}{dt} = V_{inv} + IR_i + \frac{L_i dI}{dt}$$

$$V_{rect} - V_{inv} - IR_r - IR_i = (L_r + L_i) \frac{dI}{dt}$$

This can then be simplified to

$$\frac{dI}{dt} = \frac{V_{rect} - V_{inv} - IR_r - IR_i}{L_r + L_i}$$

We can now solve for V

$$V_{cap} = (V_{rect} - IR_r) - L_r \left(\frac{V_{rect} - V_{inv} - IR_r - IR_i}{L_r + L_i} \right)$$

$$V_{cap} = \left(\frac{V_{rect}L_r + V_{rect}L_i - IR_rL_r - IR_iL_i}{L_r + L_i} \right) - \left(\frac{V_{rect}L_r - V_{inv}L_r - IR_rL_r - IR_iL_r}{L_r + L_i} \right)$$

$$V_{cap} = \left(\frac{V_{rect}L_i - IR_rL_i}{L_r + L_i} \right) - \left(\frac{-V_{inv}L_r - IR_iL_r}{L_r + L_i} \right)$$

$$V_{cap} = \left(\frac{(V_{rect} - IR_r)L_i + (V_{inv} + IR_i)L_r}{L_r + L_i} \right)$$