ksim = 1 second

delay\_VBPCurveShift = 10 second delay

delay\_VoltageSampling = 120 second delay

for each ***second*** in analysis:

for each node:

**CALCULATE NET ZIP LOADS USING LOADS FROM KSIM AND**

**INVERTER OUTPUT FROM KSIM-1**

|  |  |
| --- | --- |
| Inputs | Time reference |
| Z loads | ksim |
| I loads | ksim |
| P loads | ksim |
| P inverter (injection/absorption) | ksim-1 |
| Q inverter (injection/absorption) | ksim-1 |
| Outputs | Time reference |
| S (net power demand) | ksim |

end

**RUN FORWARD-BACKWARD SWEEP TO CALCULATE POWER FLOWS**

|  |  |
| --- | --- |
| Inputs | Time reference |
| V0 (bus voltage) | ksim |
| S (net power demand) | ksim |
| feeder mapper | ksim |
| Outputs | Time reference |
| V (line voltage) | ksim |
| I (current) | ksim |
| S (power flow along lines) | ksim |

for each node:

**CALCULATE INVERTER OUTPUT USING INVERTER FUNCTION**

|  |  |
| --- | --- |
| Inputs | Time reference |
| TimeStep – set at 1 |  |
| Filtered voltage (yk) | ksim- delay\_VoltageSampling(knode) |
| solar generation | ksim |
| V (line voltage) | ksim |
| V (line voltage) | ksim-1 |
| VBP | ksim |
| Inverter LPF – set at 1 |  |
| Sbar (max inverter capacity) |  |
| P inverter (injection/absorption) | ksim-1 |
| Q inverter (injection/absorption) | ksim-1 |
| ROC / ROC activate limits |  |
| ksim | ksim |
| delay\_VoltageSampling(knode) |  |
| Outputs | Time reference |
| Filtered voltage (calculated) (yk) | ksim |
| Filtered voltage (used) (yk) | ksim:end |
| P inverter (injection/absorption) | ksim |
| Q inverter (injection/absorption) | ksim |

at every timestep ksim, calculate filtered voltage

if ksim ==1 or mod(ksim, Delay\_VoltageSampling(knode) == 0:

use filtered voltage that was calculated at ksim

else:

use filtered voltage that was calculated at ksim-delay

**CALCULATE FILTERED VOLTAGE USING OBSERVER FUNCTION**

|  |  |
| --- | --- |
| Inputs | Time reference |
| TimeStep – set at 1 |  |
| V (line voltage) | ksim |
| V (line voltage) | ksim-1 |
| HP-filtered voltage | ksim-1 |
| Epsilon-voltage | ksim-1 |
| Filtered voltage (yk) | ksim-1 |
| HP-filter frequency – set at 1 |  |
| LP-filter frequency – set at 0.1 |  |
| Gain energy – set at 1e5 |  |
| Outputs | Time reference |
| HP-filtered voltage | ksim |
| Epsilon-voltage | ksim |
| Filtered voltage | ksim |

**CALCULATE VBP PARAMETERS USING ADAPTIVE CONTROL FUNCTION**

|  |  |
| --- | --- |
| Inputs | Time reference |
| TimeStep – set at delay\_VBPCurveShift |  |
| kp / kq – set at 100 |  |
| HP-filtered voltage | ksim |
| HP-filtered voltage | ksim- delay\_VBPCurveShift(knode) |
| upk / uqk | ksim- delay\_VBPCurveShift(knode) |
| threshold |  |
| Filtered voltage (yk) | ksim |
| Outputs | Time reference |
| upk / uqk | ksim |

if mod(ksim, Delay\_VoltageSampling(knode)) == 0:

run adaptive controller to calculate upk / uqk

adjust VBP accordingly

else:

don’t run adaptive controller & VBP is unchanged

**ADJUST VBP IF NEEDED**

|  |  |
| --- | --- |
| Inputs | Time reference |
| upk / uqk | ksim |
| V1, V2, V3, V4 (parameters) | ksim |
| Outputs | Time reference |
| VBP | ksim:end |

end