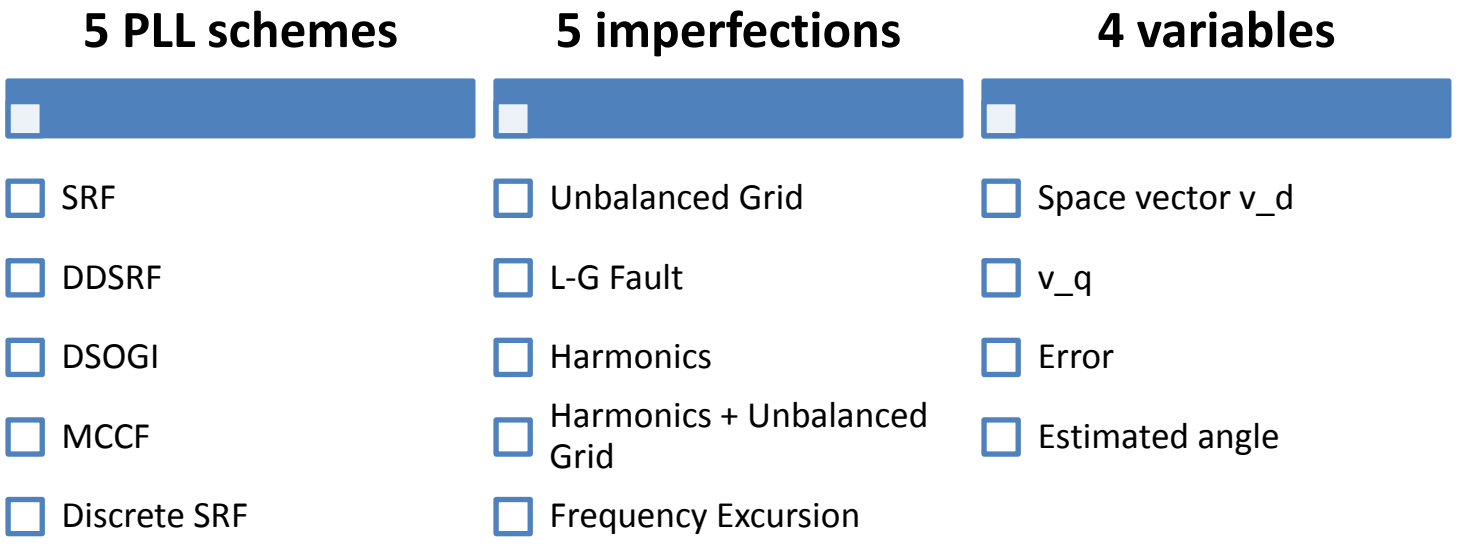
- Recap
  - SRF, DDSRF, DSOGI, MCCF, Discrete SRF
    - **SRF**: conventional PLL – tracks changes in frequency and phase
    - **DDSRF**: attacks the issue of unbalanced grid
    - **DSOGI**, **MCCF**: deal with harmonics and unbalanced grid
  - The above algorithms were studied from literature, mathematically analyzed and implemented using MATLAB Simulink

- Algorithms
  - SRF – Stationary to synchronous reference frame
    - » Loop Filter, Integrator, Closed loop system
  - DDSRF – Decoupling network is the major improvement
    - » Enables tracking in unbalanced situation through decoupling of positive and negative sequence voltages
  - DSOGI – Second order generalized integrator type filter
    - » Filters out the fundamental component in the stationary reference frame
  - MCCF – Multiple Complex Coefficient Filters
    - » Requires two modules for eliminating every harmonic component
  - Discrete SRF – Discretized version of the SRF PLL
    - » Input is sampled, transfer functions in z-domain

- Formulating a **common convention**
  - All existing PLLs are modifications of the conventional SRF PLL
  - Different conventions are followed in literature
  - Single platform between different schemes requires a common convention

- Simulation of each scheme required an in-depth understanding since mathematical models had to be re-designed to suit to our convention
- Each scheme was modeled and tested individually and these models were clubbed into a single self-consistent model for comparison
- Complexity involved – more than **600 comparisons**!!



- Development of an **applet** (learning tool)
  - **Parts**
    - Initialization program
    - Simulink models
    - Intuitive program to plot
  - **Demo**
  - **Importance** of the applet
    - Excellent tool for learning – How?
    - Easy to use
    - To find out which scheme tackles a particular issue better
    - Additions of other schemes (if required) can be easily done

- **Future work**
  - Design optimization aimed at achieving a faster dynamic performance
  - Verification of results through practical setup

Thank you

Back

Synchronous Reference Frame PLL (SRF)

Situations:

1. Frequency Excursion
2. L-G Fault
3. Harmonics
4. Unbalanced Grid
5. Harmonics + Unbalanced Grid

Back

Other PLL scheme comparisons

Situations:

1. Frequency Excursion
2. L-G Fault
3. Harmonics
4. Unbalanced Grid
5. Harmonics + Unbalanced Grid

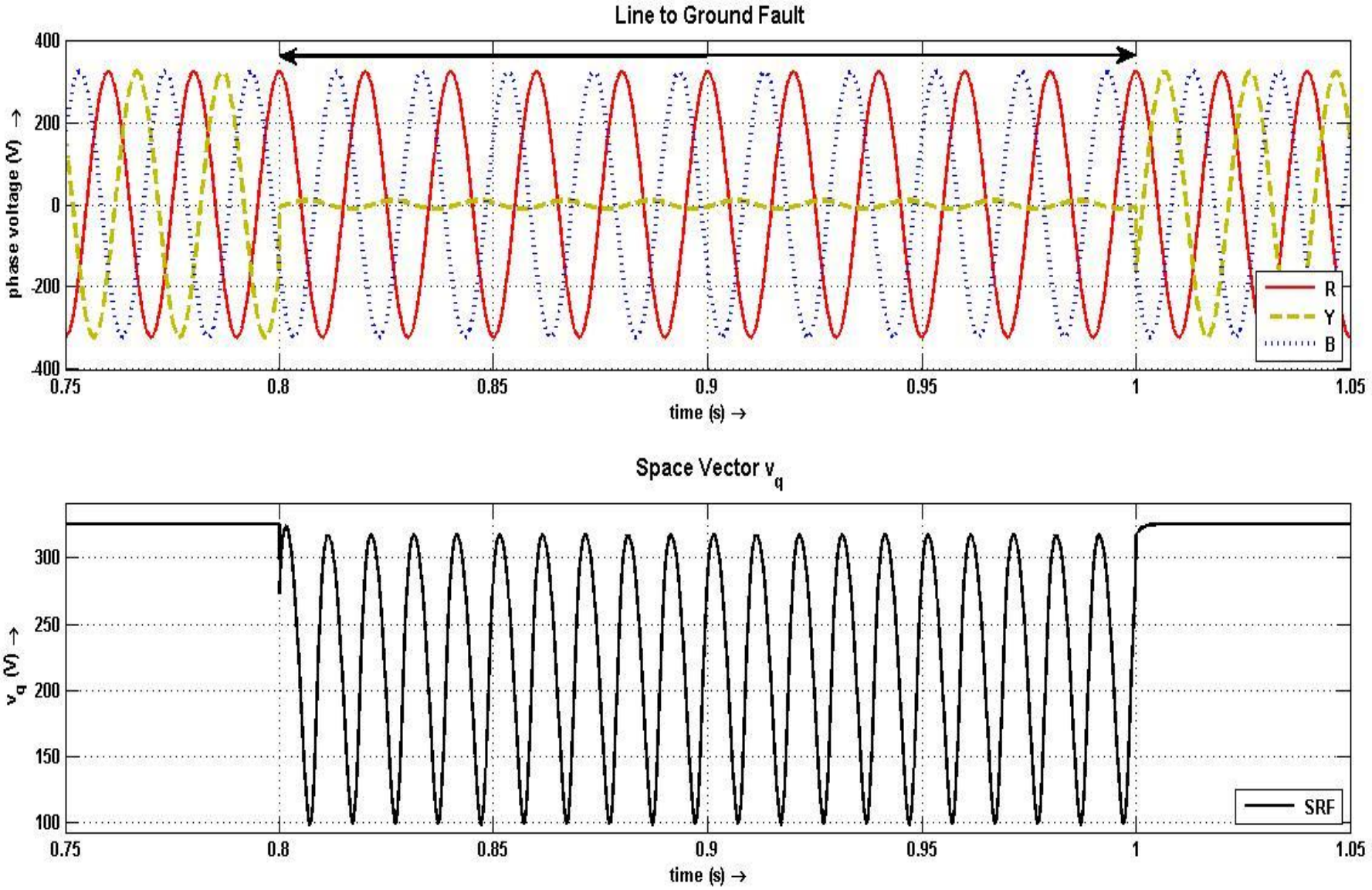
Discrete PLL

Situations:

1. Frequency Excursion
2. L-G Fault
3. Harmonics
4. Unbalanced Grid
5. Harmonics + Unbalanced Grid

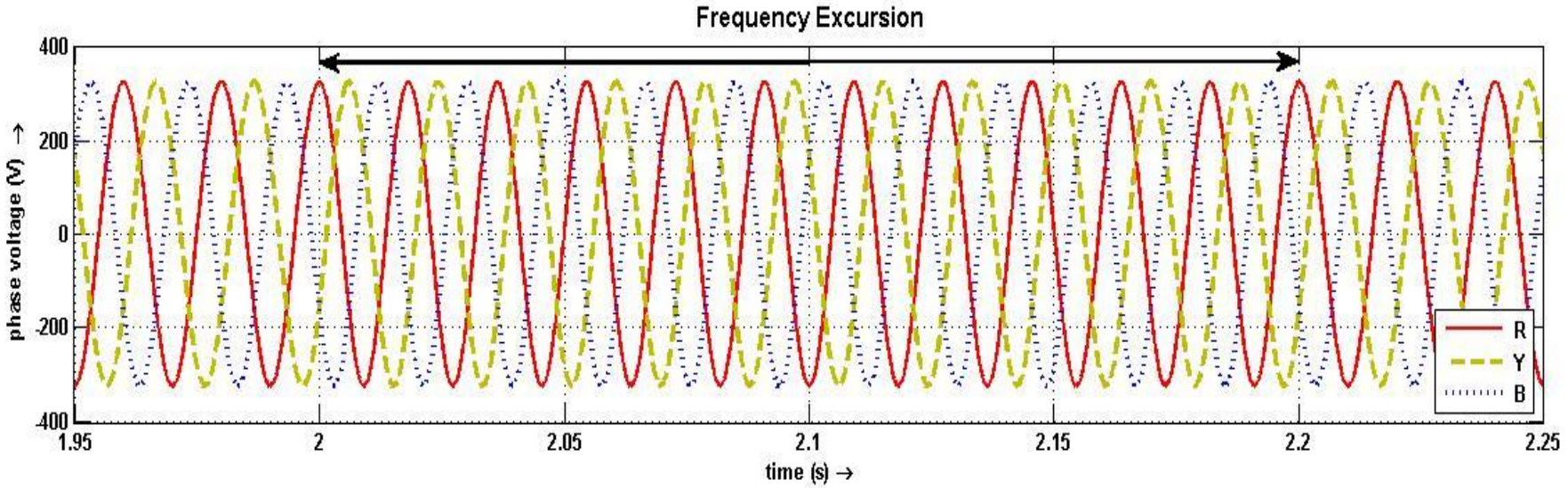
SRF PLL on L-G Fault

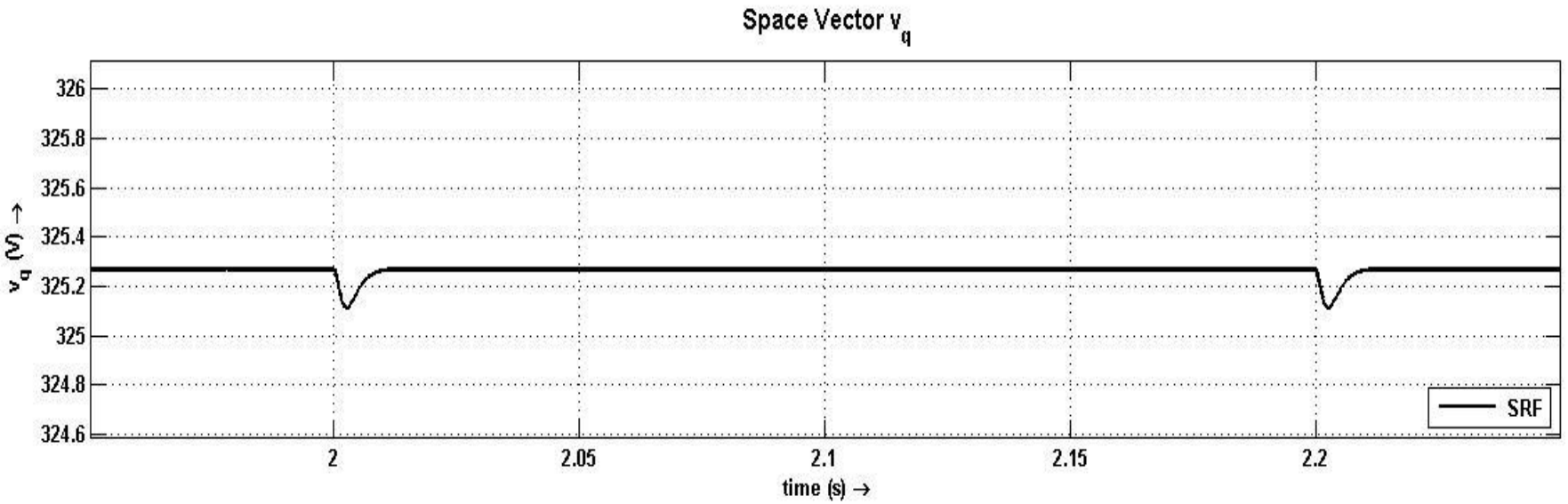
Back



SRF PLL on Frequency Excursion

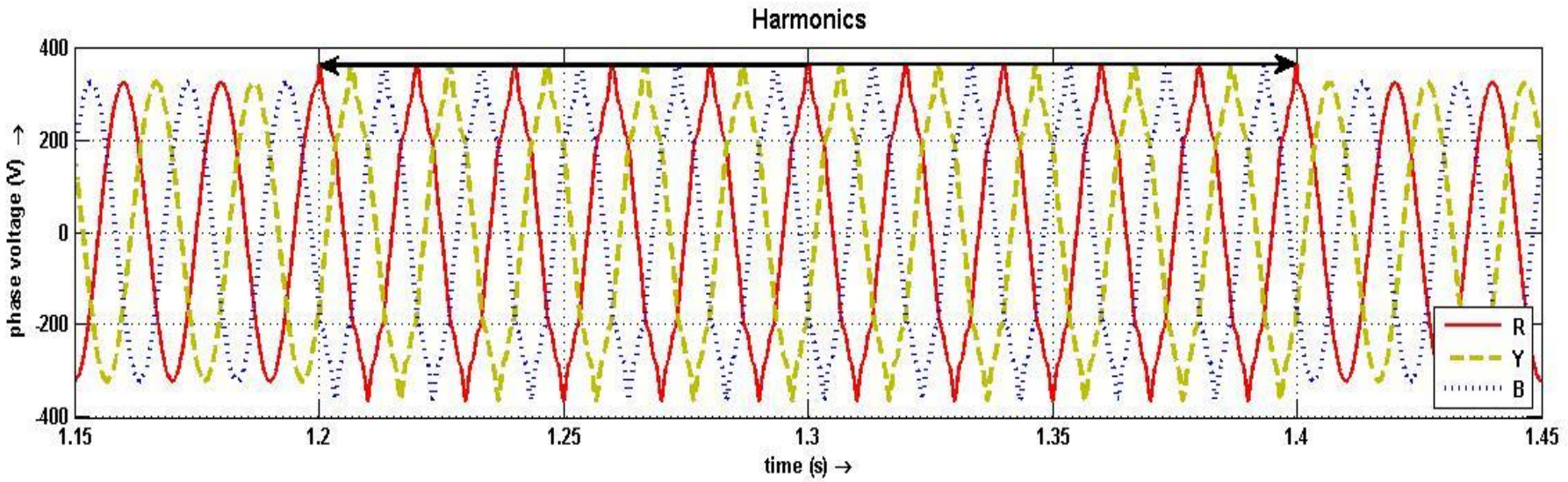
Back

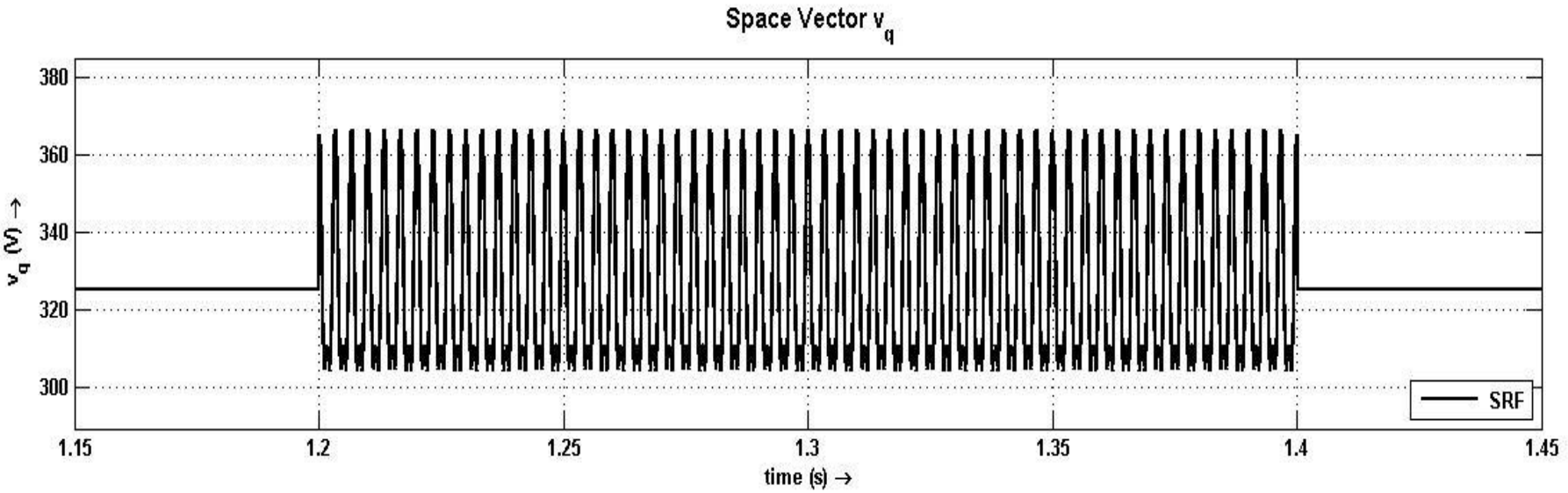




SRF PLL on Harmonics

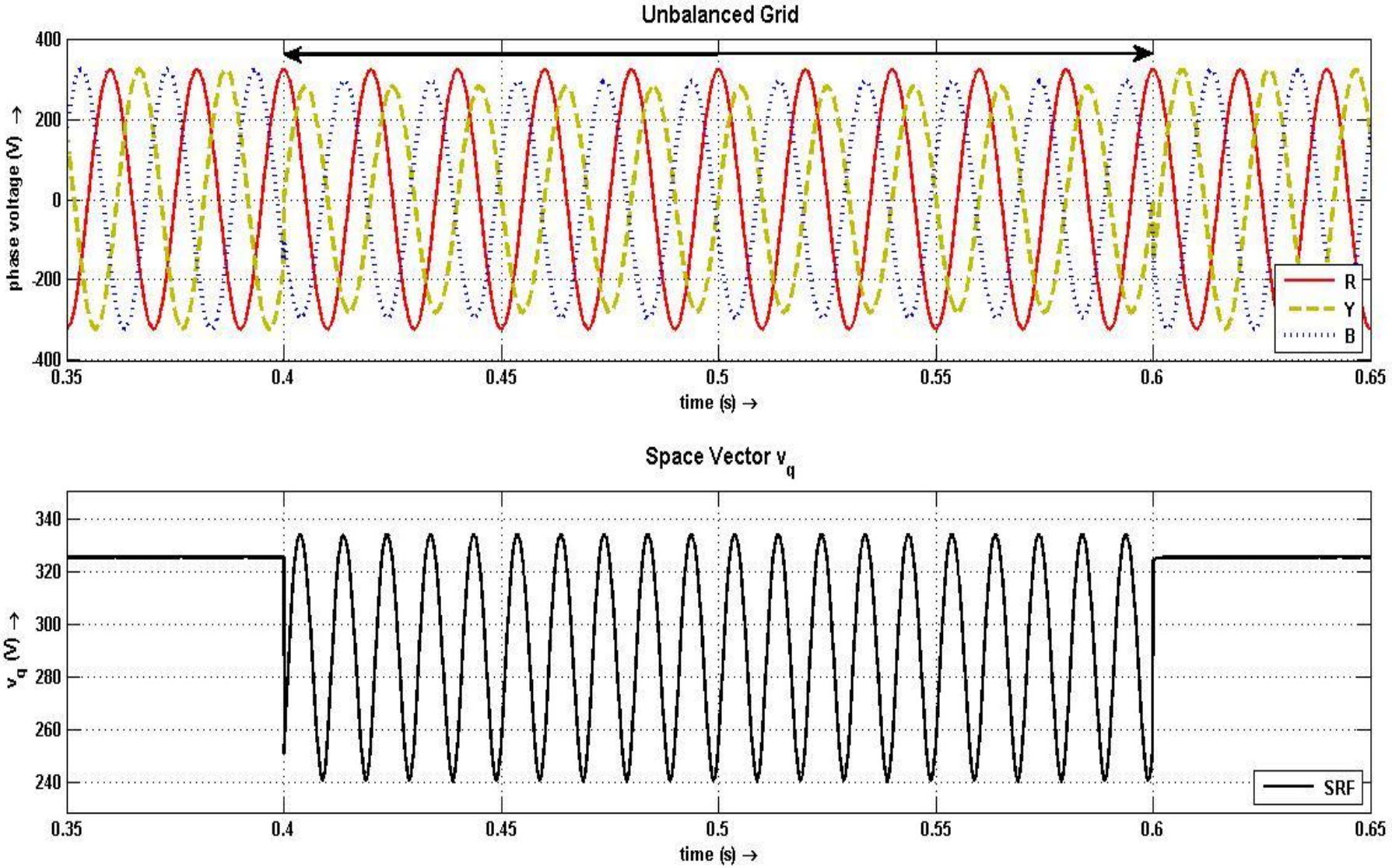
Back





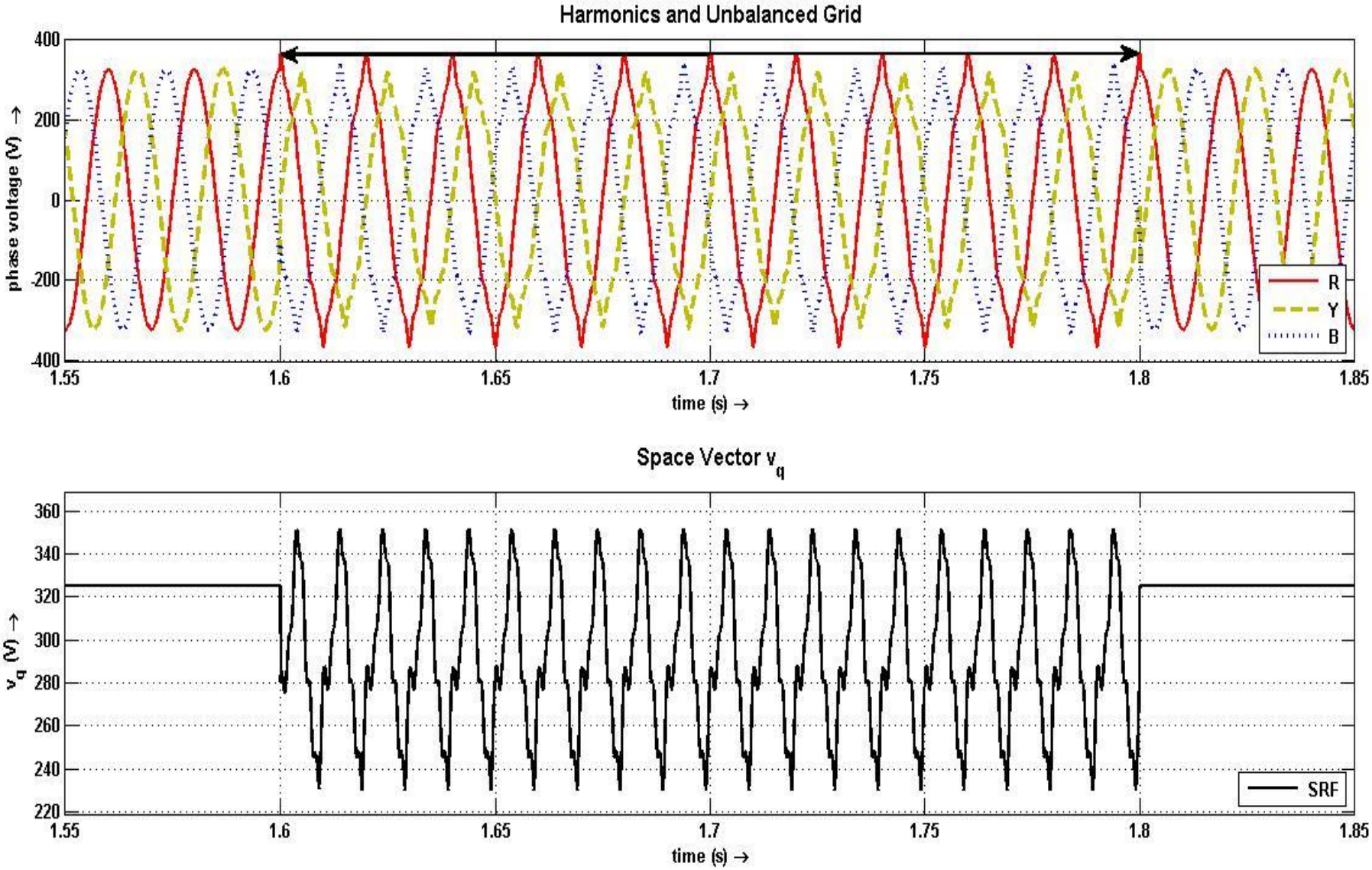
SRF PLL on Unbalanced Grid

Back



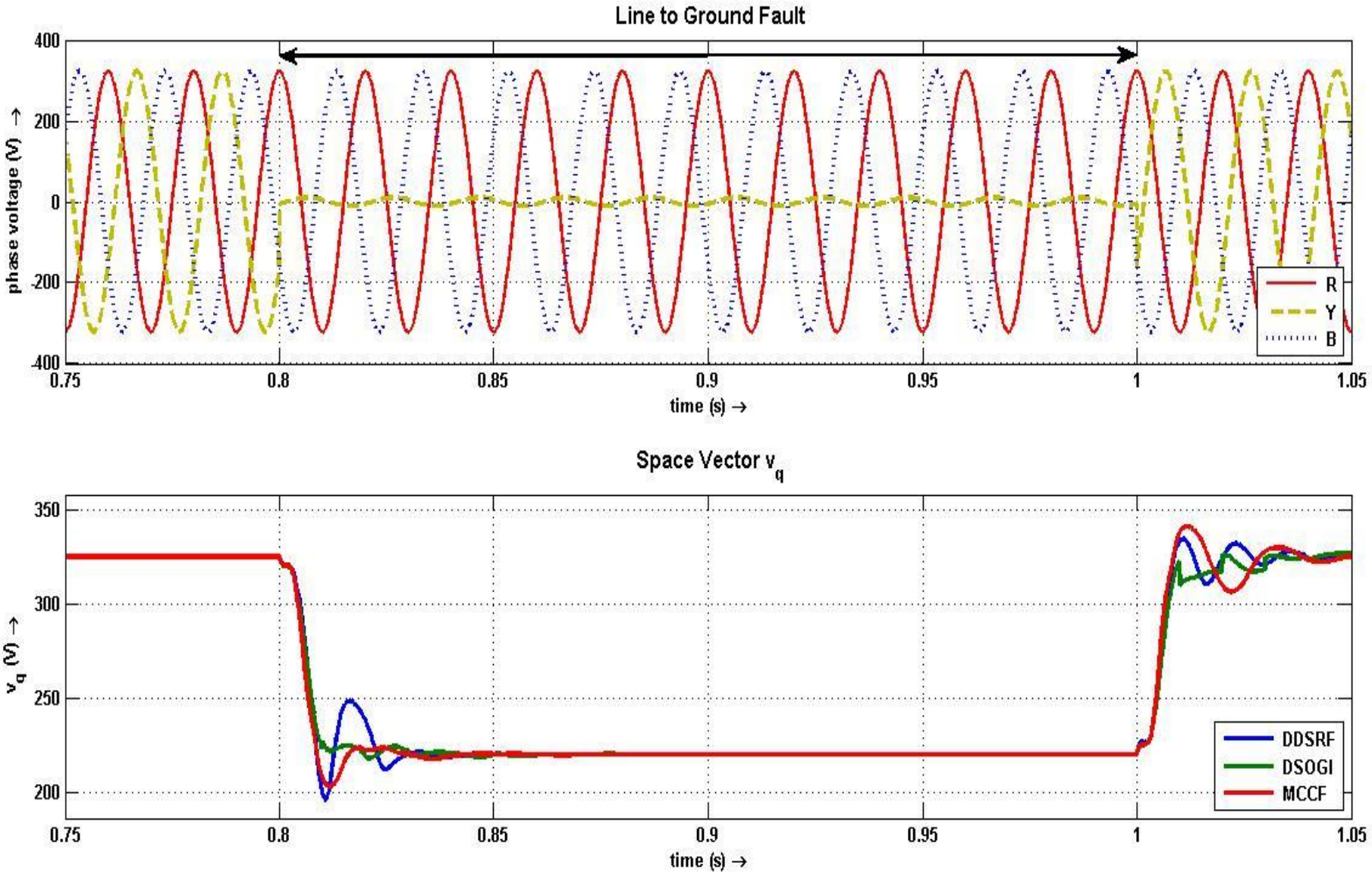
SRF PLL on Harmonics + Unbalanced Grid

Back



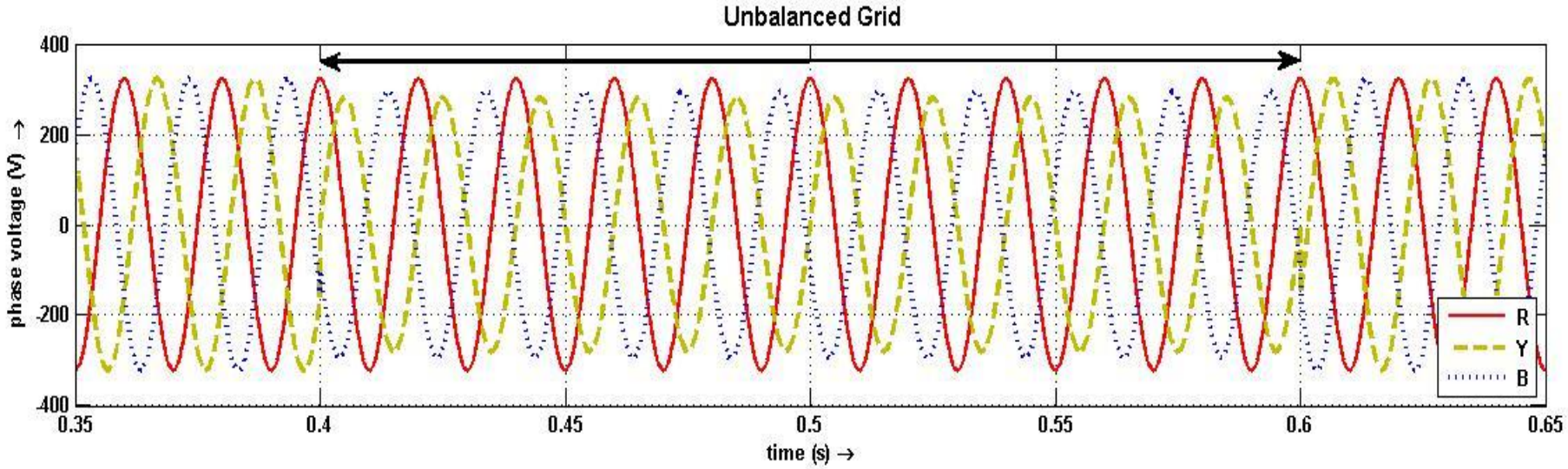
DDSRF, DSOGI, MCCF on L-G Fault

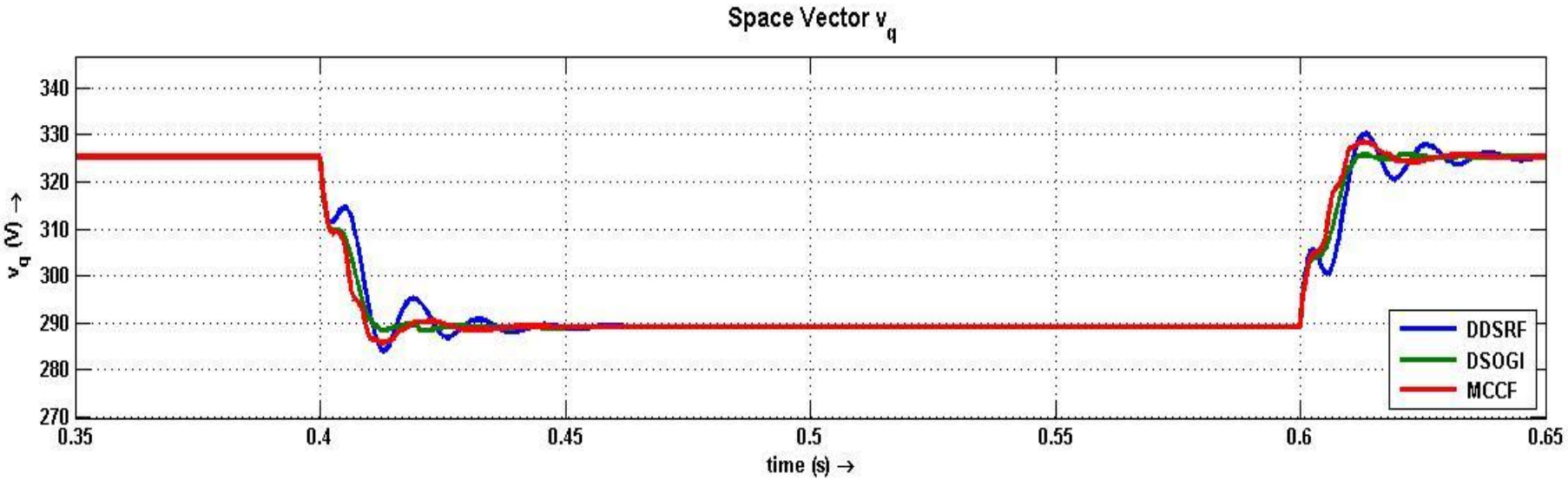
Back



DDSRF, DSOGI, MCCF on Unbalanced Grid

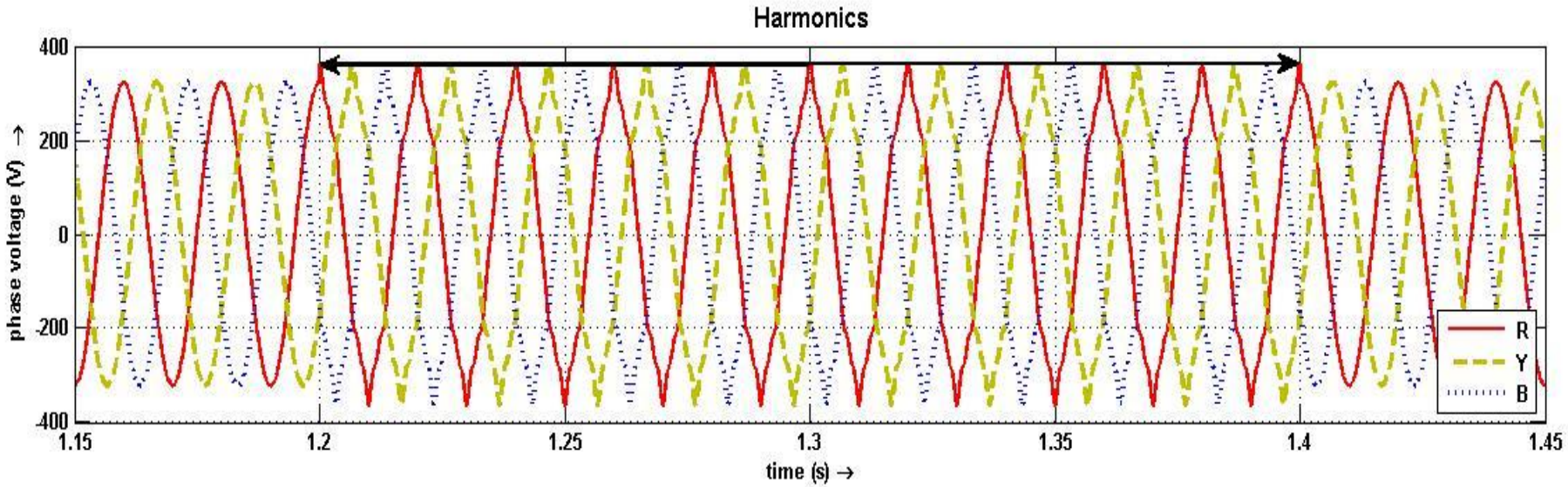
Back

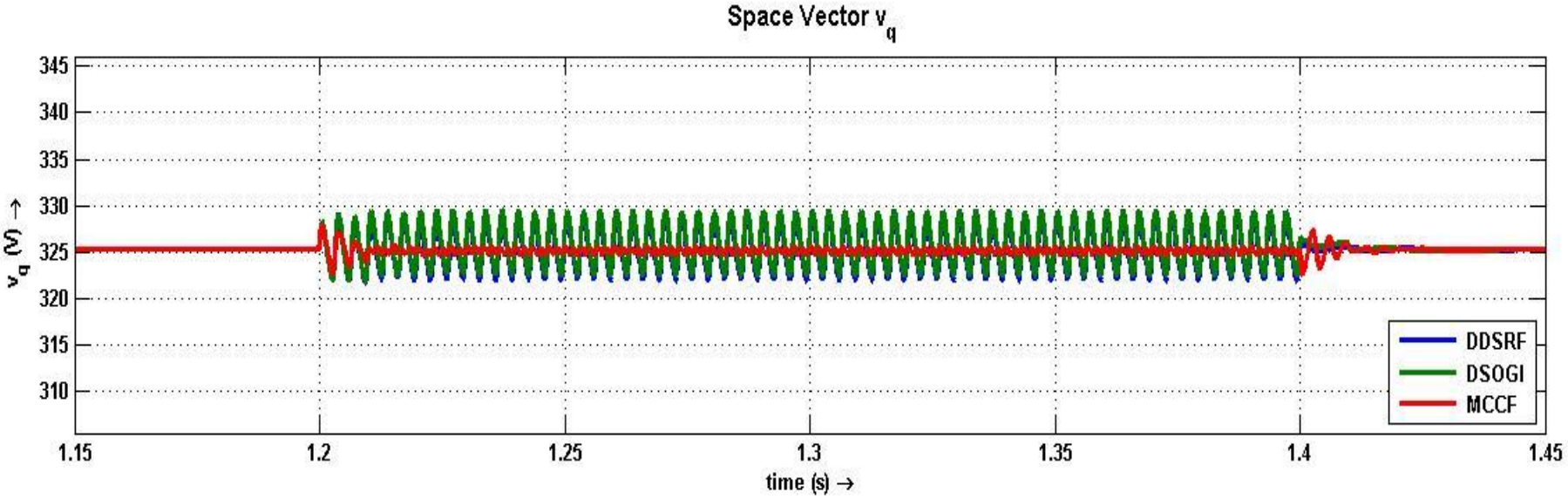




DDSRF, DSOGI, MCCF on Harmonics

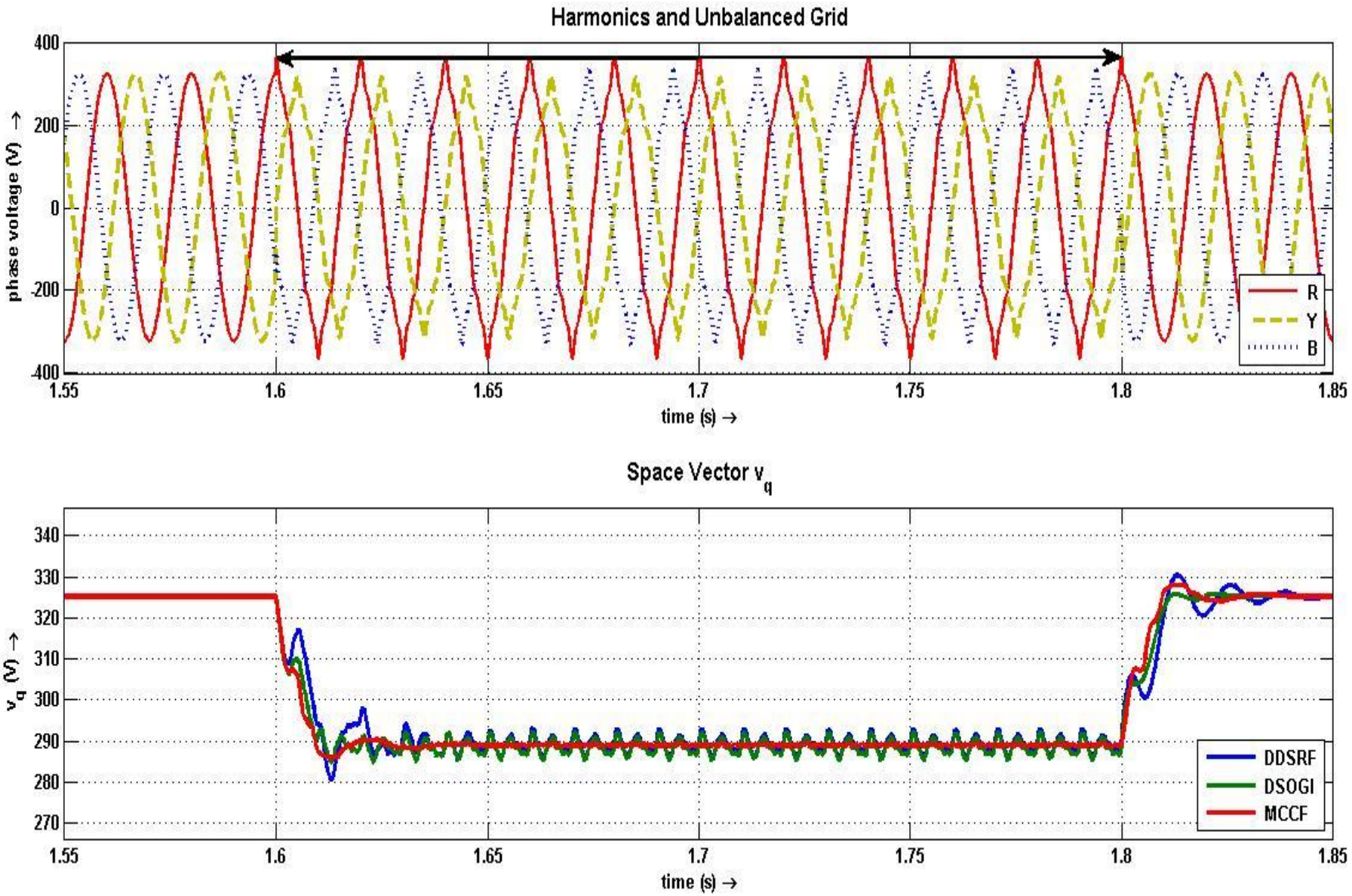
Back





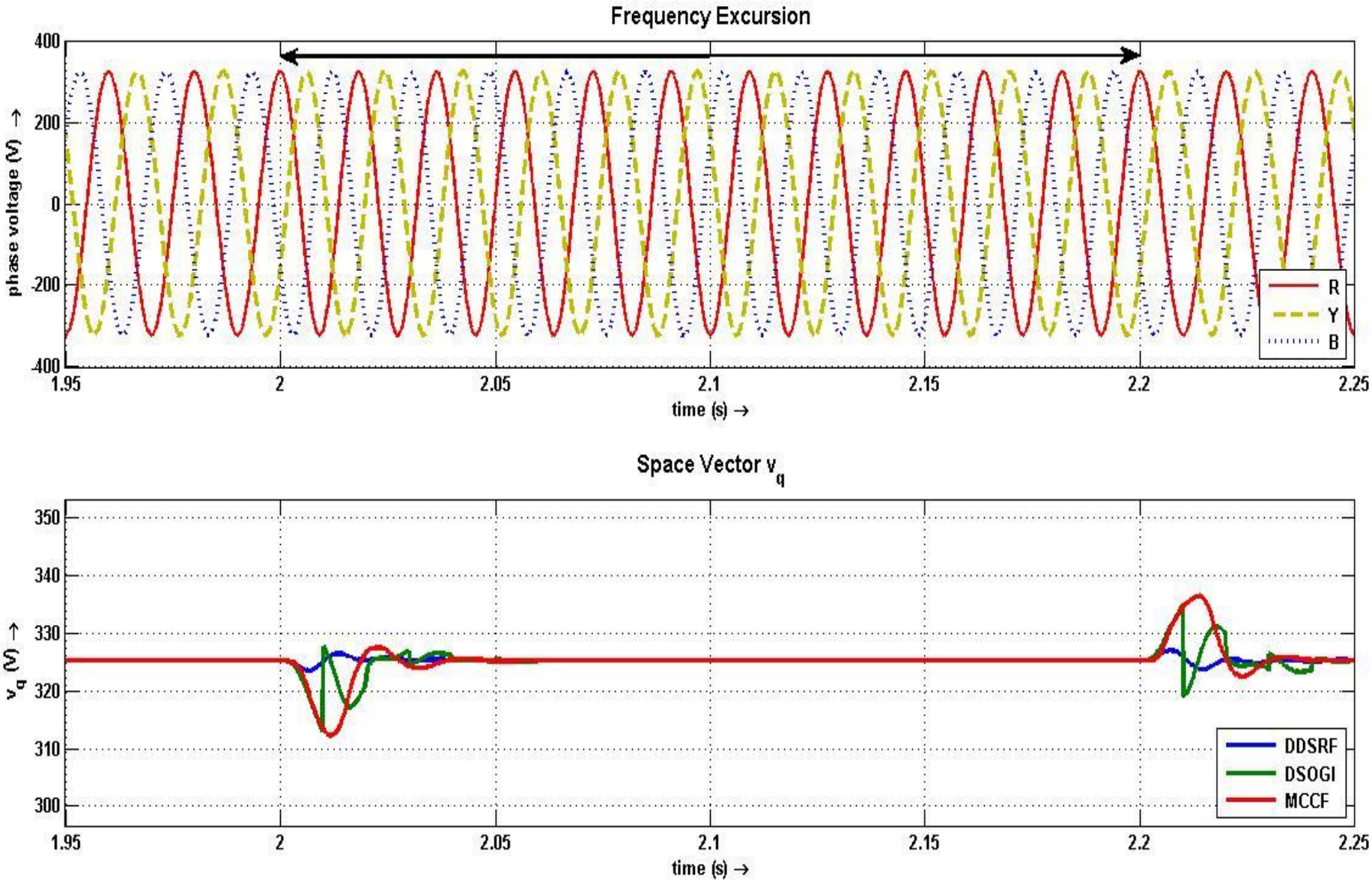
DDSRF, DSOGI, MCCF on Harmonics + Unbalanced Grid

Back



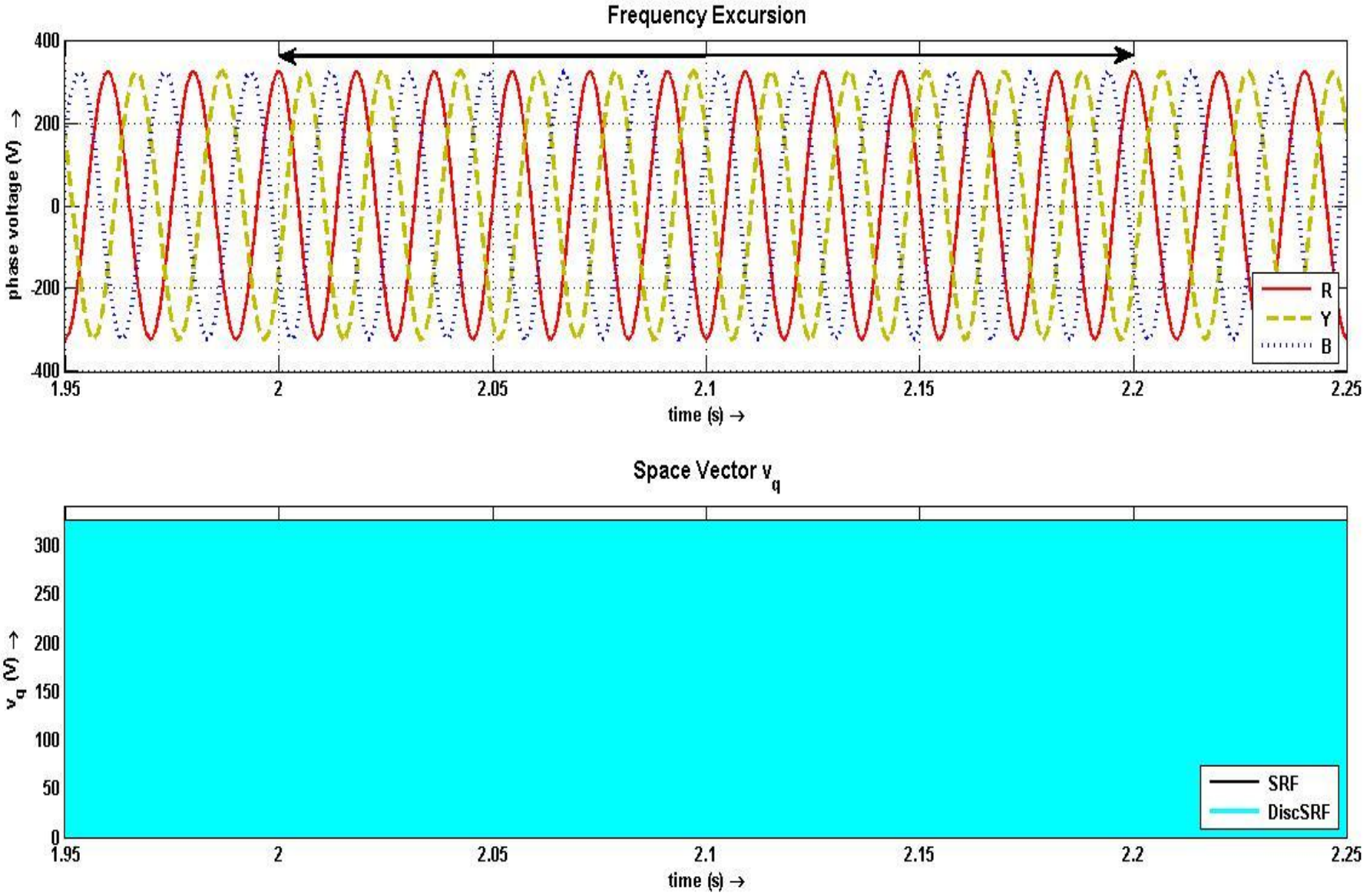
DDSRF, DSOGI, MCCF on Frequency Excursion

Back



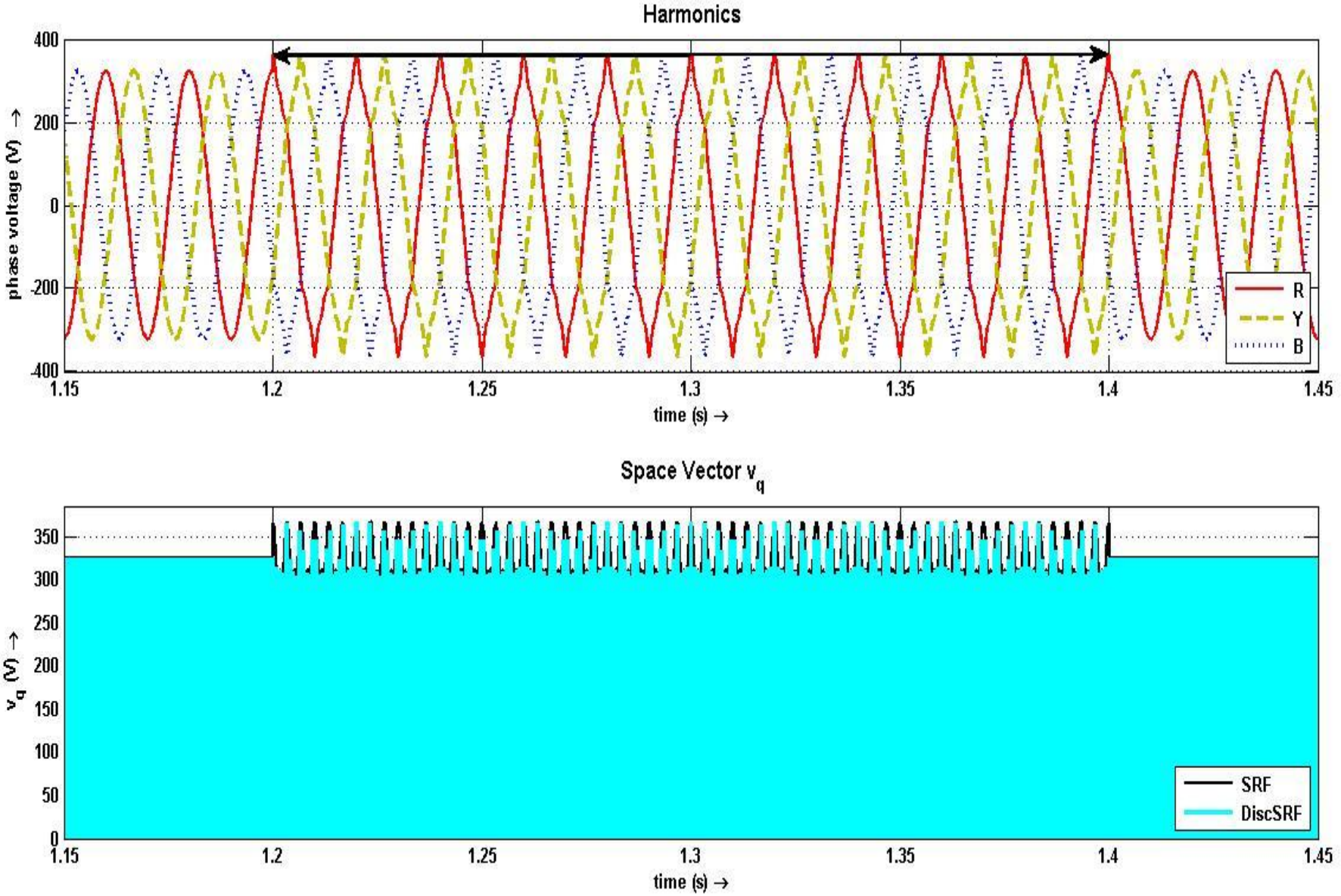
Discrete PLL on Frequency Excursion

Back



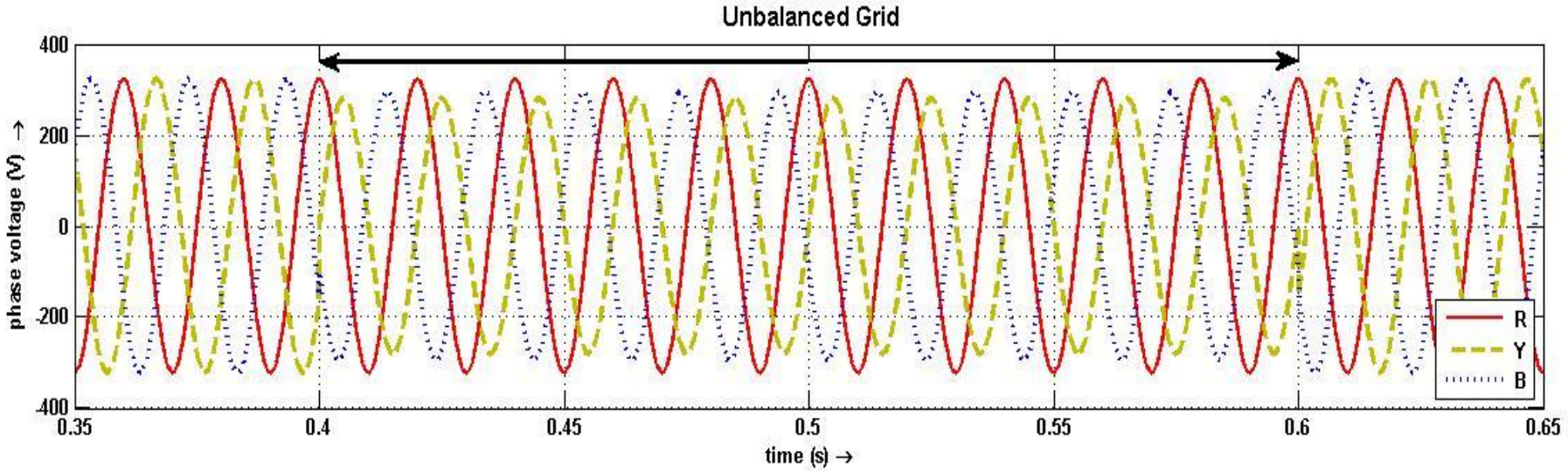
Discrete PLL on Harmonics

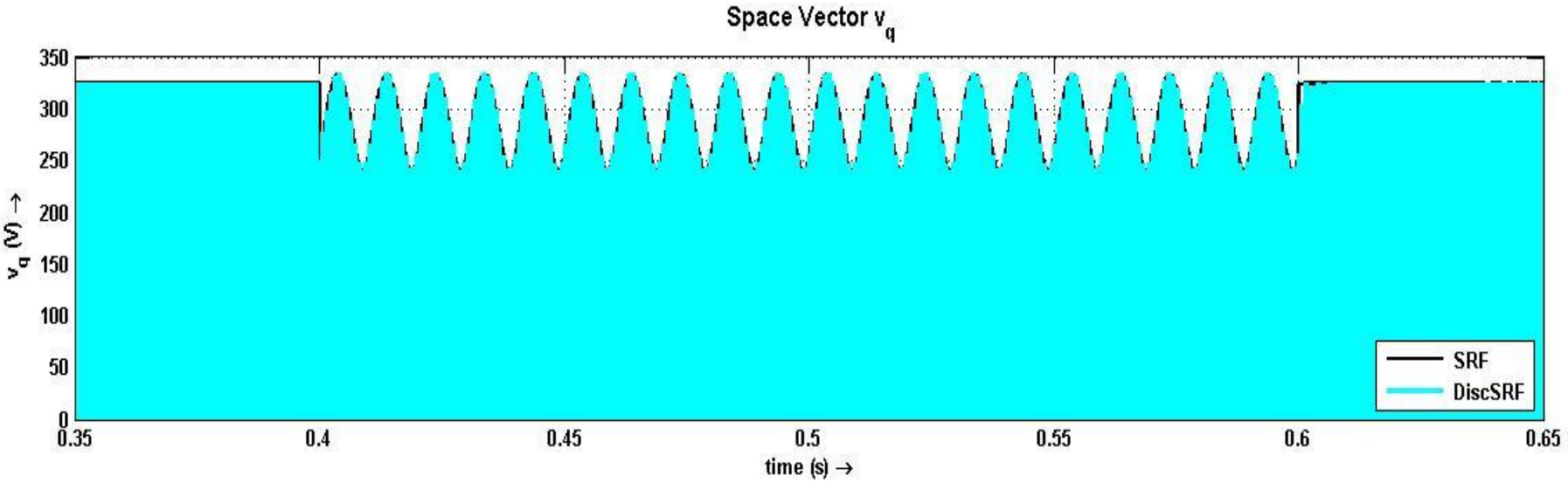
Back



Discrete PLL on Unbalanced Grid

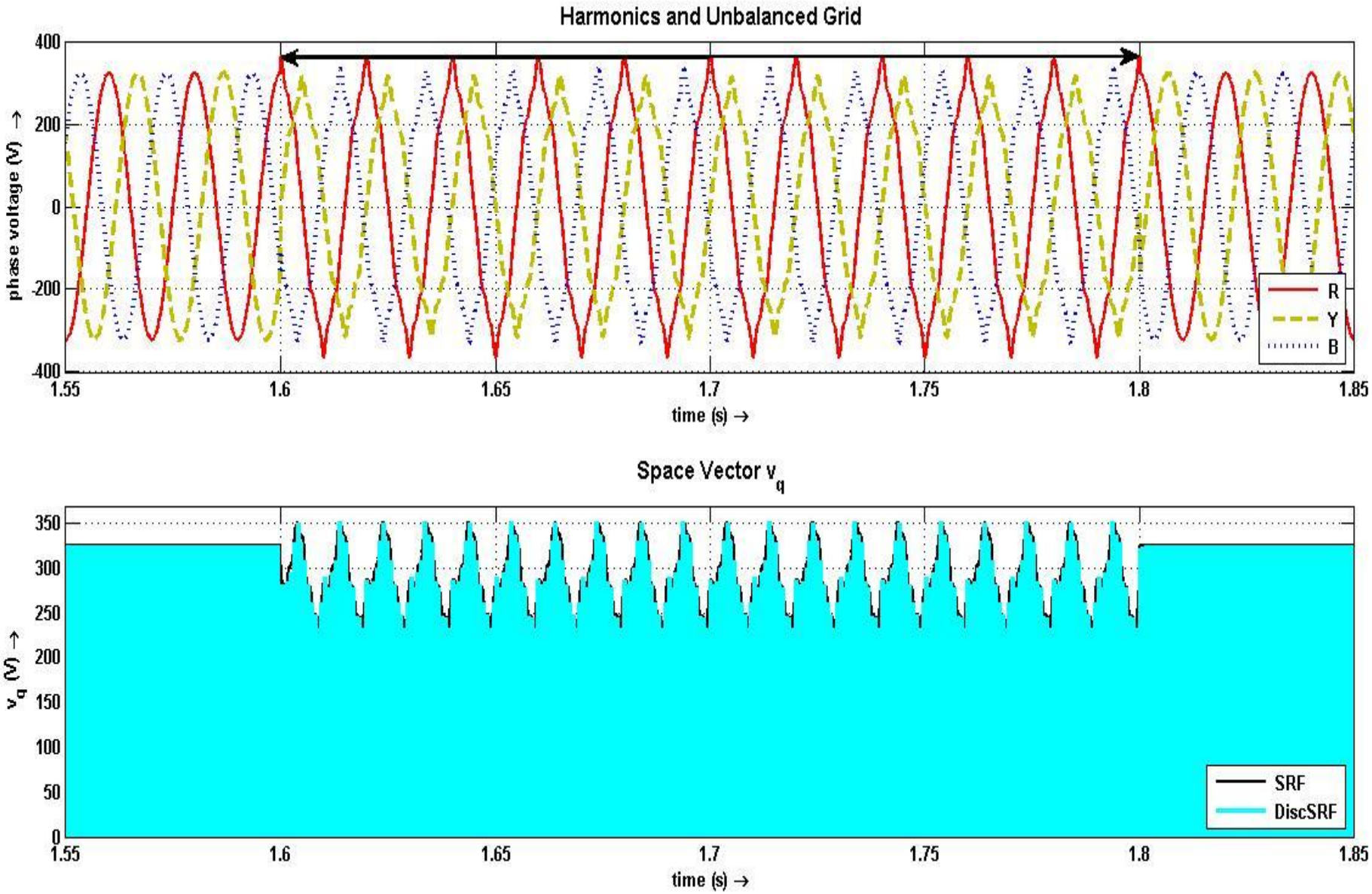
Back





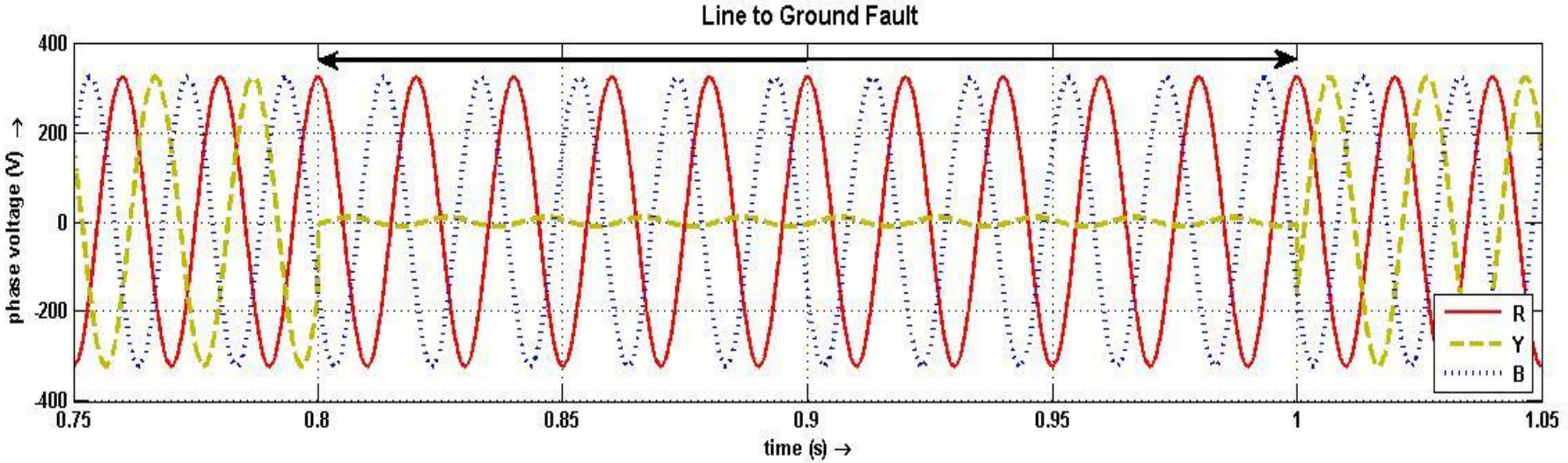
Discrete PLL on Harmonics + Unbalanced Grid

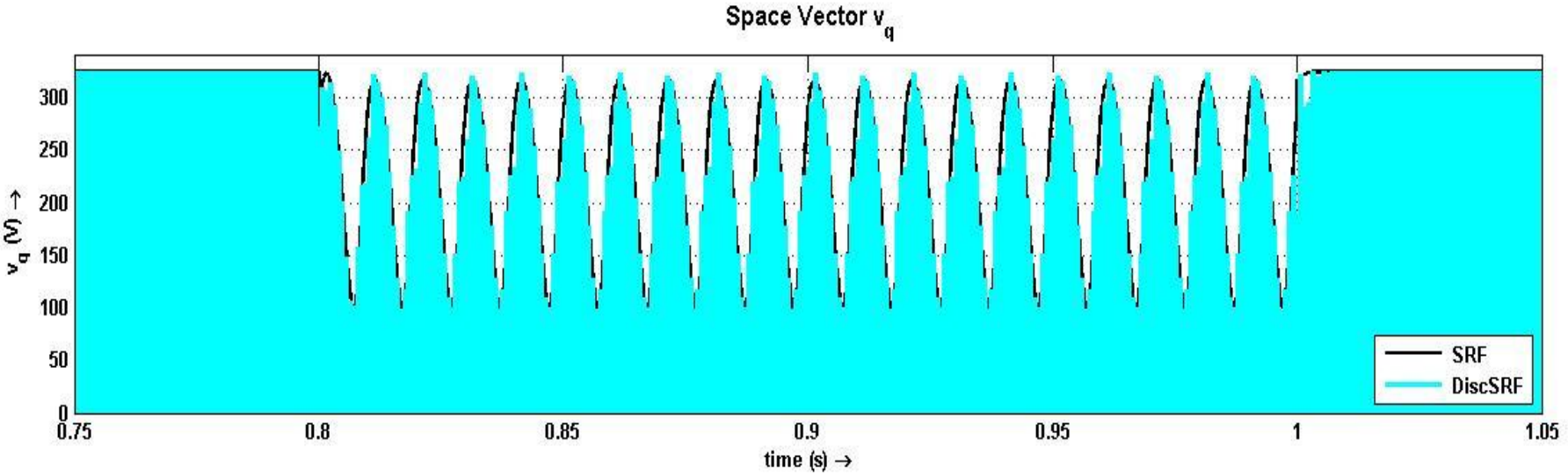
Back



Discrete PLL on L-G Fault

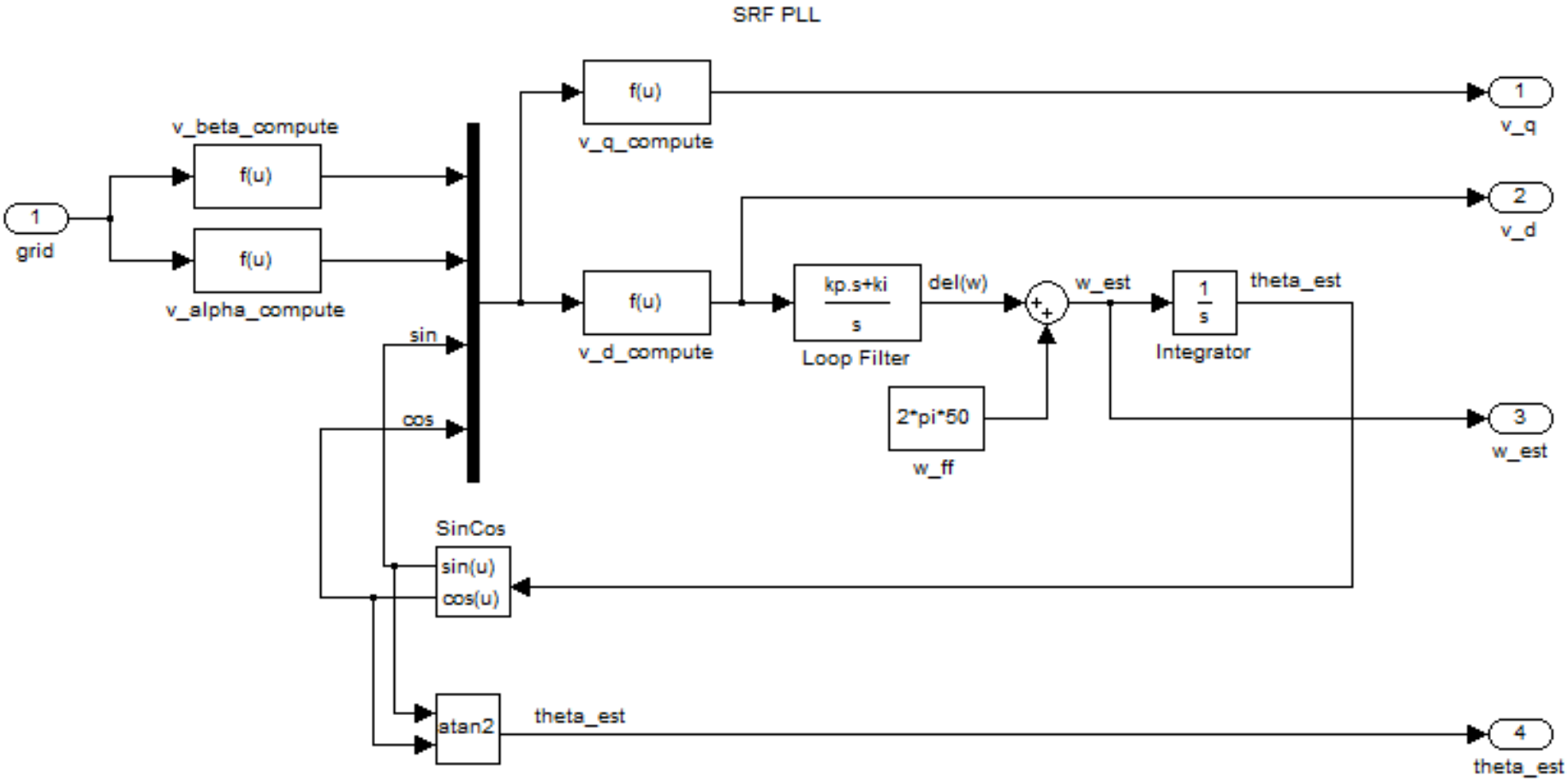
Back





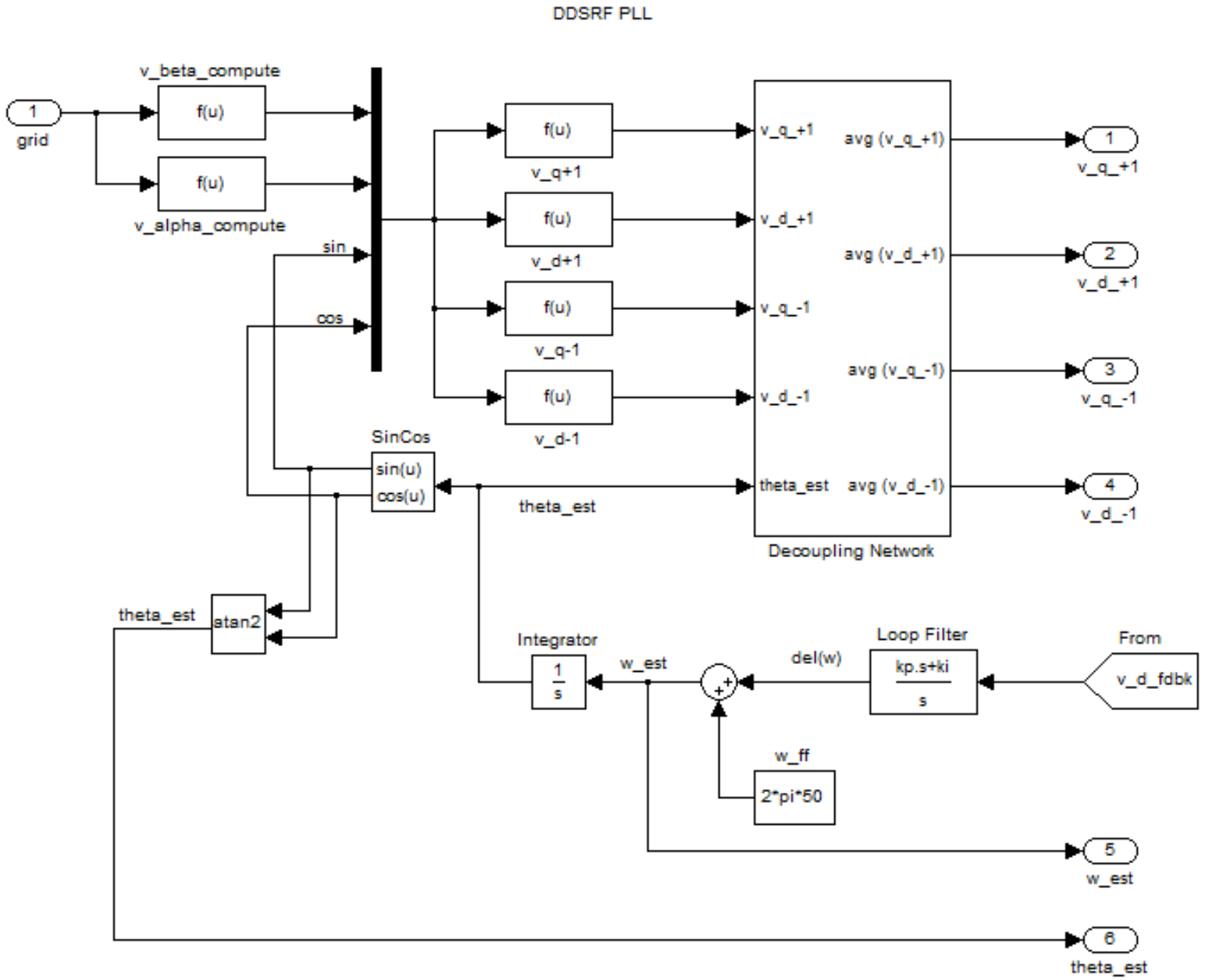
Conventional SRF PLL

Back



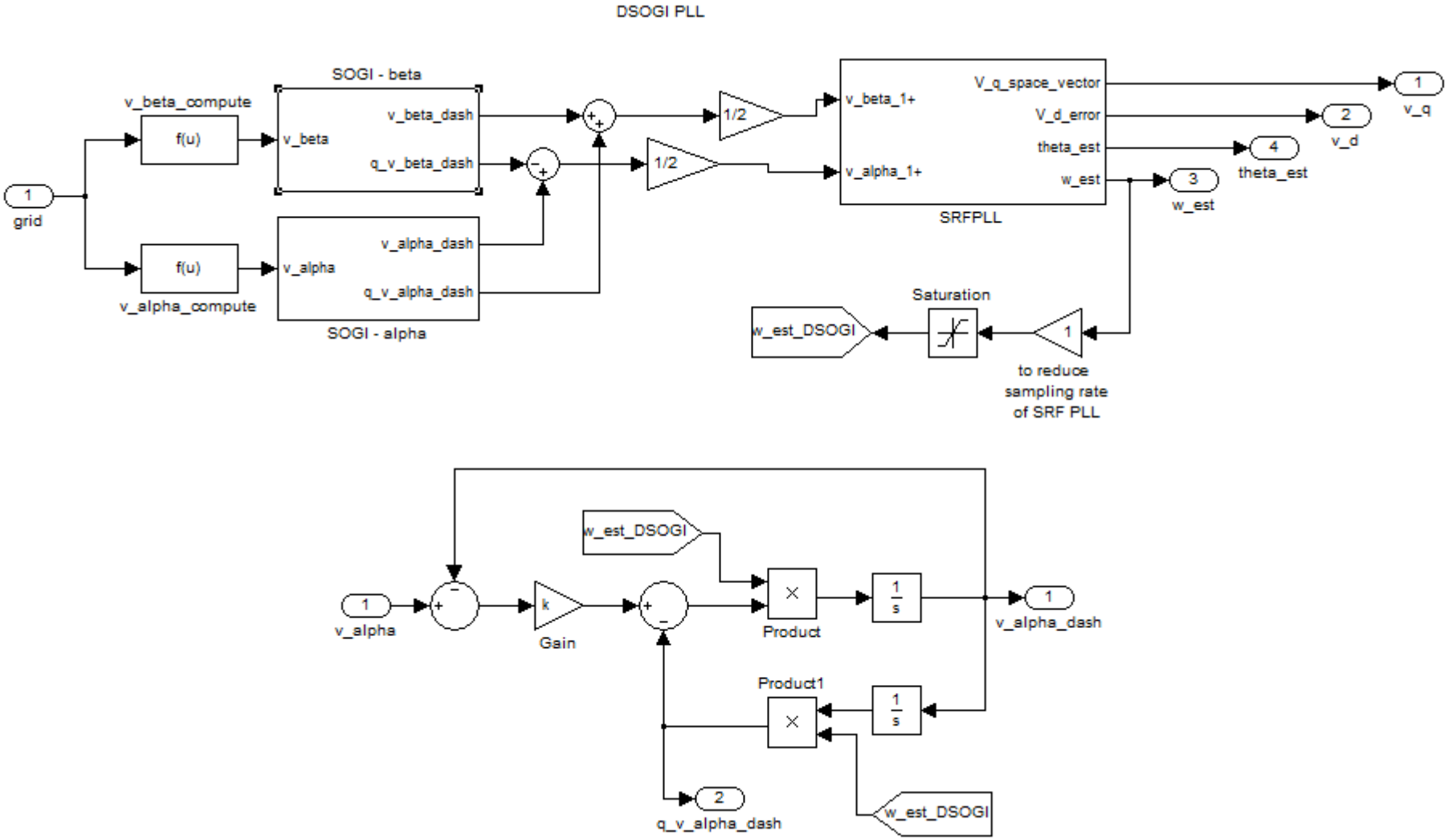
DDSRF PLL

Back

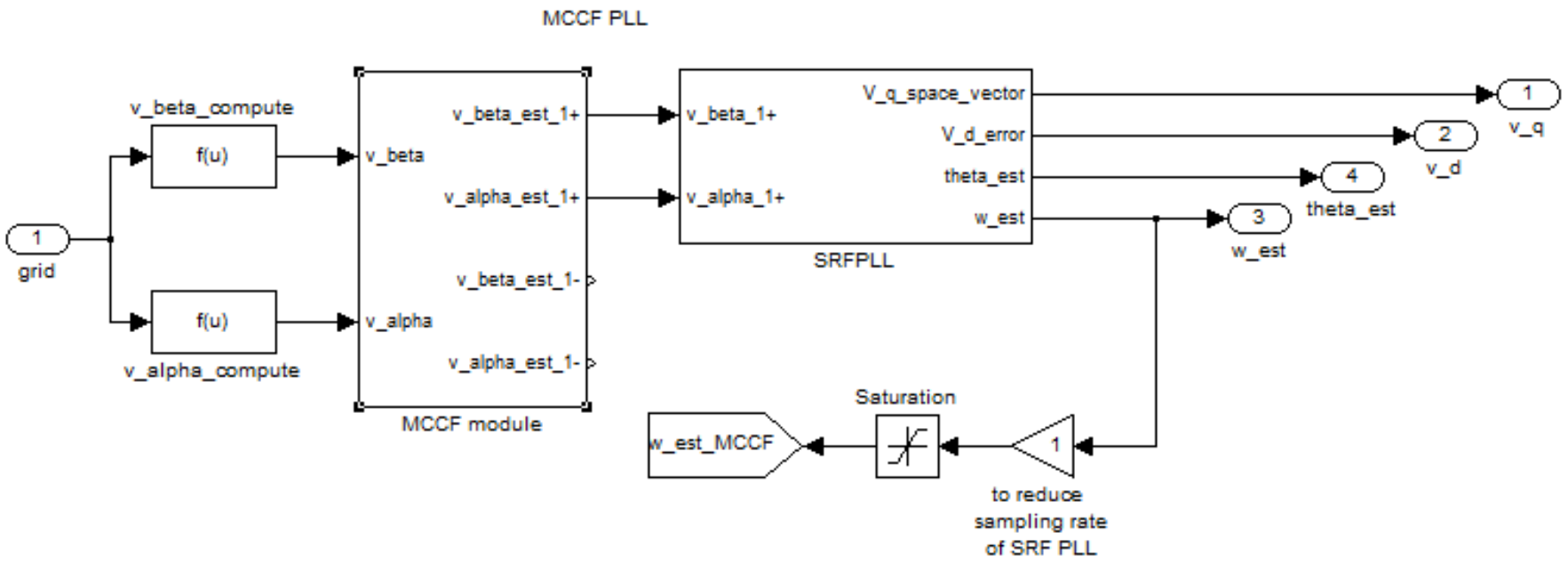


DSOGI PLL

Back

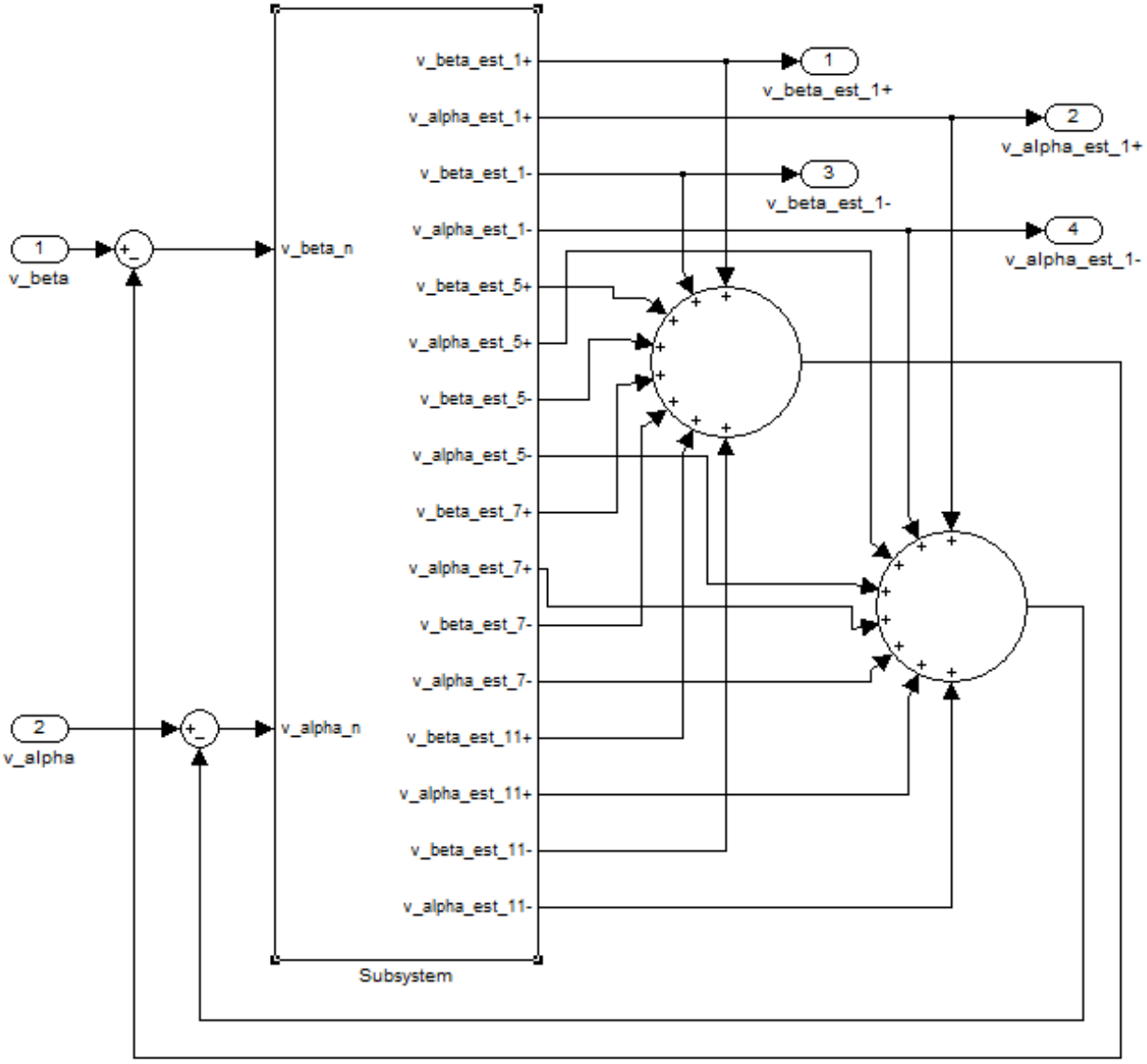


MCCF PLL


MCCF PLL
v_beta_compute
f(u)
v_alpha_compute
f(u)
1
grid
v_beta
v_alpha
v_beta_est_1+
v_alpha_est_1+
v_beta_est_1-
v_alpha_est_1-
MCCF module
v_beta_1+
v_alpha_1+
V_q_space_vector
V_d_error
theta_est
w_est
SRFPLL
1
v_q
2
v_d
4
theta_est
3
w_est
Saturation
w_est_MCCF
1
to reduce
sampling rate
of SRF PLL

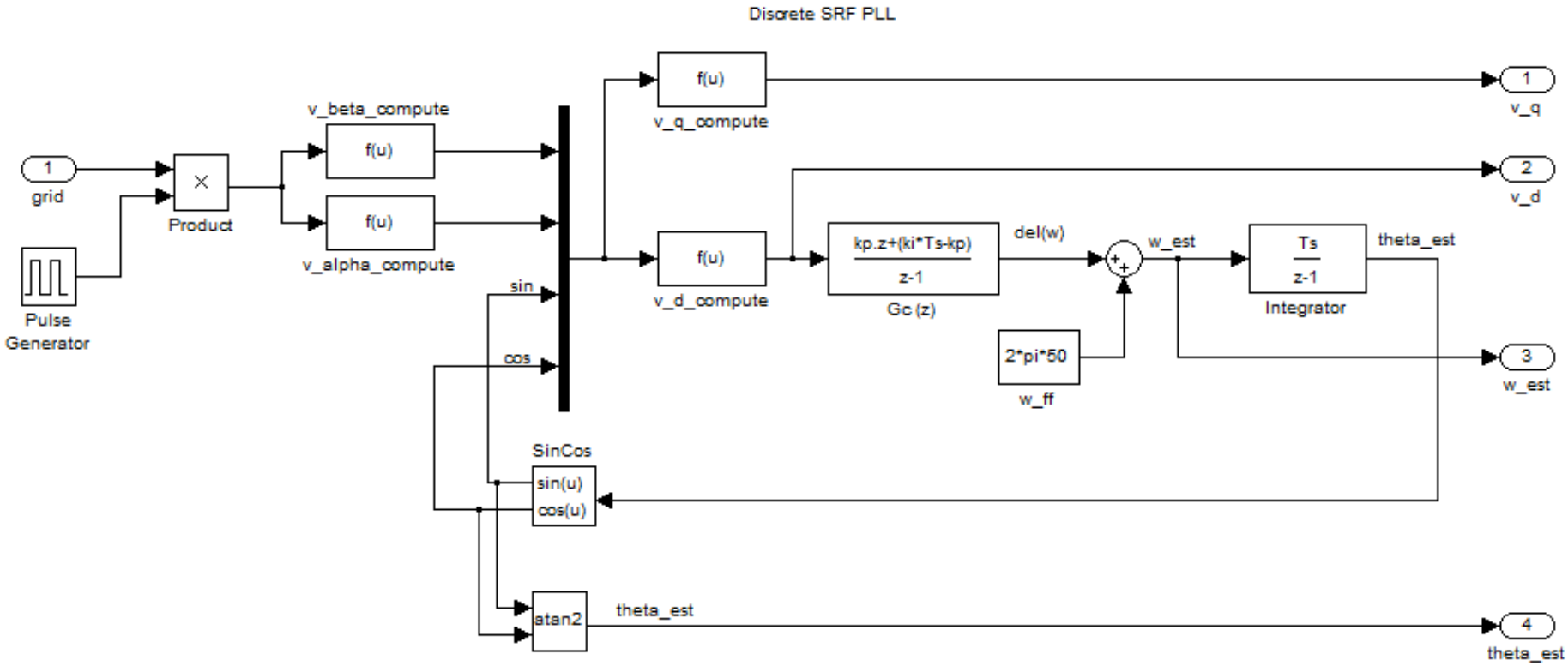
MCCF module (continued)

Back



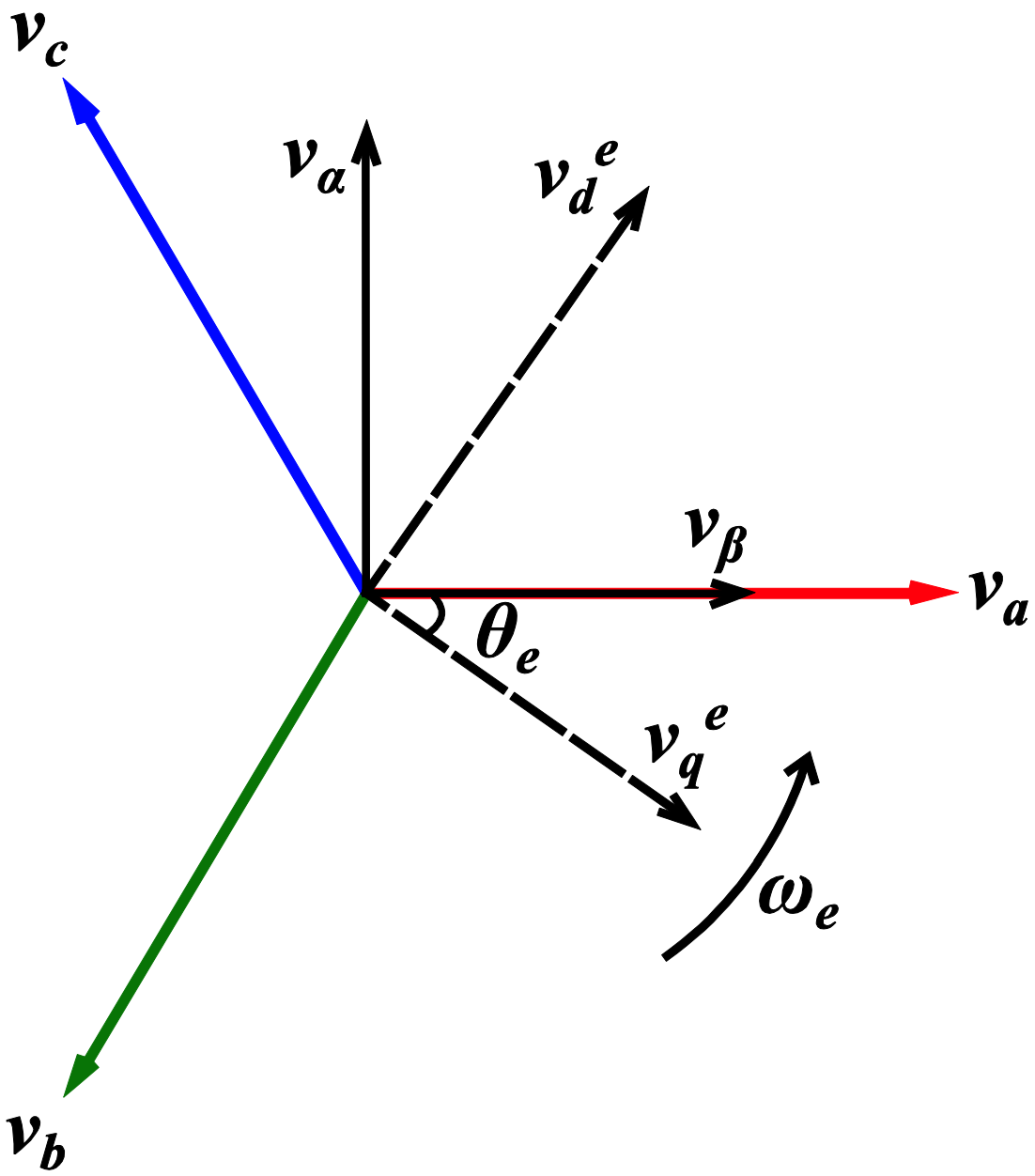
Discrete SRF PLL

Back

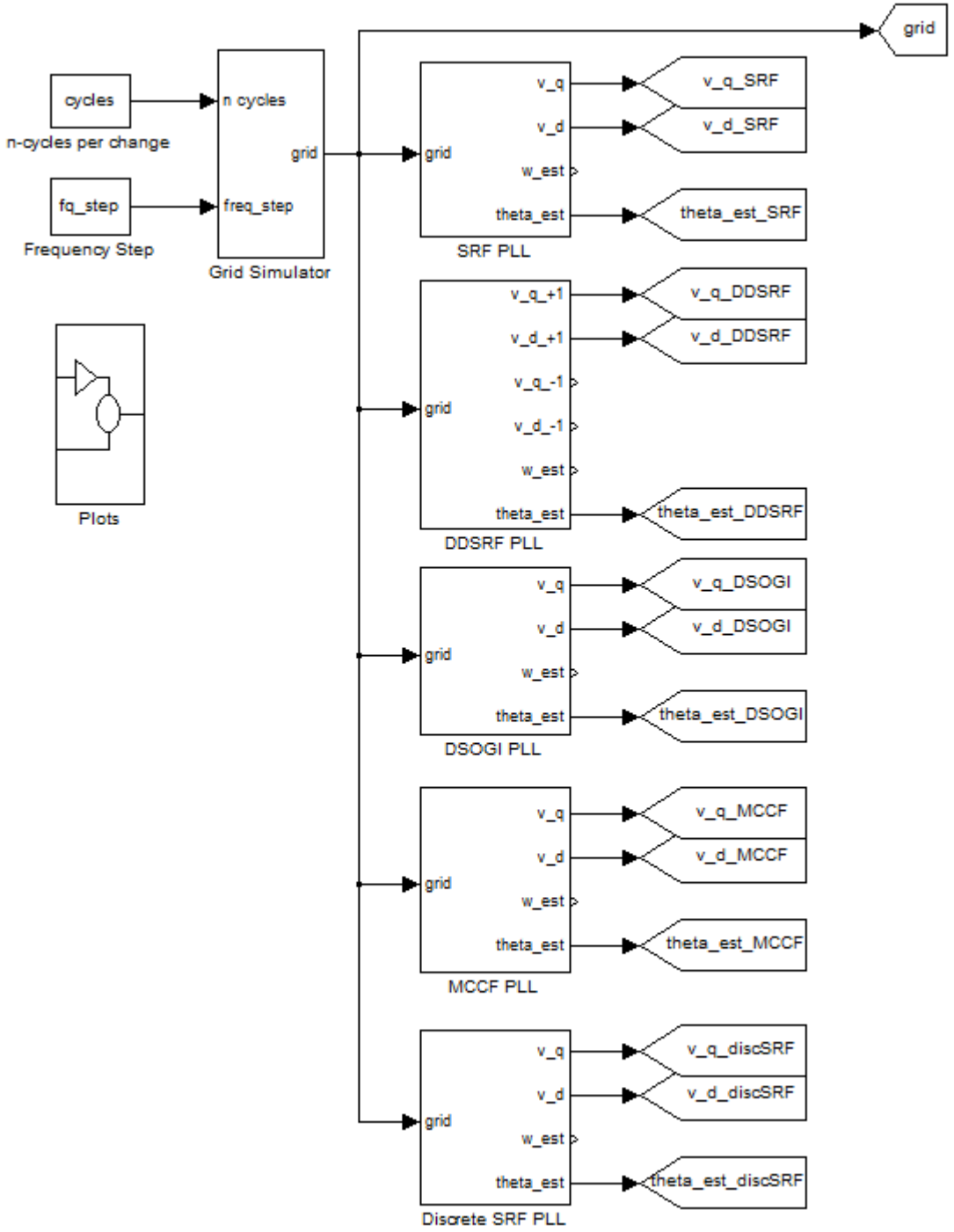


Convention followed

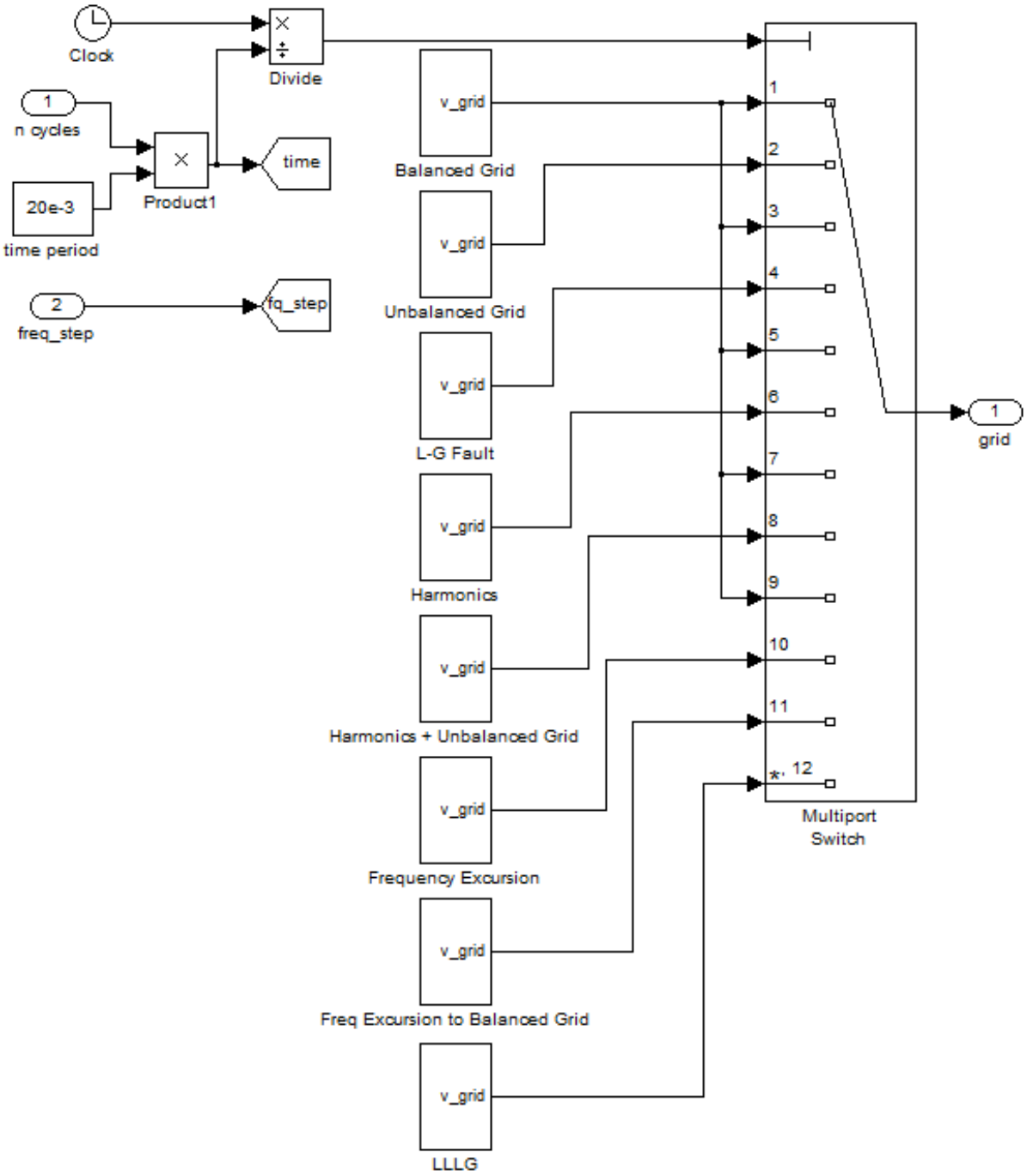
Back



Self-consistent model



Grid simulator



[Back]