Unified State Feedback Control of a Hybrid Distribution Transformer using Particle Swarm Optimization

1st Dave Figueroa
Institute of Control and Industrial Electronics
Warsaw University of Technology
Warsaw, Poland
email address or ORCID

2nd Alvaro Carreno
Institute of Control and Industrial Electronics
Warsaw University of Technology
Warsaw, Poland
email address or ORCID

3rd Mariusz Malinowski
Institute of Control and Industrial Electronics
Warsaw University of Technology
Warsaw, Poland
email address or ORCID

***Abstract*—Here will be the abstract of the paper.**

***Index Terms*—hybrid distribution transformer, optimal control, particle swarm optimization**

I. Introduction

ń

THE increasing penetration of renewable energy sources= in the electrical grid has led to a significant rise in the use of power electronic converters. These converters are essential for integrating RES into the grid, as they facilitate the conversion of DC power generated by sources like solar panels and wind turbines into AC power compatible with the grid. However, the widespread use of power electronic converters has also introduced challenges related to power quality, such as the injection of harmonics and non-linear loads, which can lead to voltage distortions and other issues in the electrical grid.

One of the advantages of the HDT is its ability to compensate grid voltage disturbances, such as sags and swells, and to compensate the load current harmonics, in case of unbalanced loads. Moreover, since the HDT uses the DT, it provides galvanic isolation between primary and secondary sides and higher short-circuit current capability [1].

II. Model of the Hybrid Distribution Transformer

A. Series Converter

$$
\begin{aligned}
v_s &= R_{fs} i_{fs} + L_{fs} \frac{d\, i_{fs}}{dt} + v_{cs} \\
i_{fs} &= C_{fs} \frac{d\, v_{cs}}{dt} + i_g
\end{aligned}
\tag{1}
$$

Leaving the states on the left side, and converting to $\alpha\beta$ coordinates, the series converter model is given by:

$$
\begin{aligned}
\frac{d\, i_{fs}}{dt} &= -\frac{R_{fs}}{L_{fs}} i_{fs} - \frac{1}{L_{fs}} v_{cs} + \frac{1}{L_{fs}} v_s \\
\frac{d\, v_{cs}}{dt} &= -\frac{1}{C_{fs}} i_{fs} + \frac{1}{C_{fs}} i_g
\end{aligned}
\tag{2}
$$

B. Parallel Converter

$$
\begin{aligned}
v_p &= R_{fp} i_{fp} + L_{fp} \frac{d\, i_{fp}}{dt} + v_{cp} \\
i_{fp} &= C_{fp} \frac{d\, v_{cp}}{dt} - i_Y + i_L
\end{aligned}
\tag{3}
$$

Leaving the states on the left side, and converting to $\alpha\beta$ coordinates, the series converter model is given by:

$$
\begin{aligned}
\frac{d\, i_{fp}}{dt} &= -\frac{R_{fp}}{L_{fp}} i_{fp} - \frac{1}{L_{fp}} v_{cp} + \frac{1}{L_{fp}} v_p \\
\frac{d\, v_{cp}}{dt} &= \frac{1}{C_{fp}} i_{fp} - \frac{1}{C_{fp}} i_Y + \frac{1}{C_{fp}} i_L
\end{aligned}
\tag{4}
$$

C. Distribution Transformer

The transformer is connected in $\Delta - Y$ configuration, with the series converter connected to the delta side, and the parallel converter connected to the Y side. The transformer equations are given by:

$$
\begin{aligned}
v_{Ya} &= N_{LFT}(v_{\Delta a} - v_{\Delta b}) \\
v_{Yb} &= N_{LFT}(v_{\Delta b} - v_{\Delta c}) \\
v_{Yc} &= N_{LFT}(v_{\Delta c} - v_{\Delta a})
\end{aligned}
\tag{5}
$$

This can be expressed in matrix form as:

$$
\begin{aligned}
v_Y &= N_{LFT} \underbrace{\begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ -1 & 0 & 1 \end{bmatrix}}_{K_T'} v_\Delta \\
v_Y &= N_{LFT} K_T' v_\Delta
\end{aligned} \tag{6}
$$

The dynamics of the transformer can be expressed as:

$$
v_Y = R_Y i_Y + L_Y \frac{d\, i_Y}{dt} + v_{cp} \tag{7}
$$

Assuming that there is no zero-sequence current, the transformer model can be expressed in $\alpha\beta$ coordinates as:

$$
\frac{d\, i_Y}{dt} = -\frac{R_Y}{L_Y} i_Y - \frac{1}{L_Y} v_{cp} + \frac{1}{L_Y} N_{LFT} K_T' v_\Delta \tag{8}
$$

D. Overall HDT Model

The overall HDT model can be expressed in state-space form as:

$$
\begin{aligned}
\frac{d}{dt}\begin{bmatrix} x_s \\ x_p \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_s & \mathbf{P}_{ig}\mathbf{M}_p \\ \mathbf{P}_{vc}\mathbf{M}_s & \mathbf{A}_p \end{bmatrix} \begin{bmatrix} x_s \\ x_p \end{bmatrix} + \begin{bmatrix} \mathbf{B}_s & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_p \end{bmatrix} \begin{bmatrix} u_s \\ u_p \end{bmatrix} \\
&\quad + \begin{bmatrix} \mathbf{0} \\ \mathbf{P}_{vg} \end{bmatrix} v_g + \begin{bmatrix} \mathbf{0} \\ \mathbf{P}_{iL} \end{bmatrix} i_L
\end{aligned} \tag{9}
$$

where the matrices $\mathbf{M}_p = \begin{bmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \end{bmatrix}$ and $\mathbf{M}_s = \begin{bmatrix} \mathbf{0} & \mathbf{I} \end{bmatrix}$ are used to select the appropriate states from the parallel and series converter state vectors respectively.

III. Control Strategy

Since we have access to all of the states of the system, we can implement a state feedback control strategy. The control law is given by:

$$
u(t) = -\mathbf{K}_x x(t) + \mathbf{K}_r r(t) + \mathbf{K}_{err} \int (r(t) - y(t)) dt \tag{10}
$$

where $\mathbf{K}_x$ is the state feedback gain matrix, $\mathbf{K}_r$ is the resonance gain matrix, $\mathbf{K}_{err}$ is the integral gain matrix, r is the reference input vector, and y is the output vector. The matrices $\mathbf{K}_x$, $\mathbf{K}_r$, and $\mathbf{K}_{err}$ are designed using LQR, which solves the Riccati equation to minimize the cost function:

$$
J = \int_0^\infty (x^T \mathbf{Q} x + u^T \mathbf{R} u) dt \tag{11}
$$

where $\mathbf{Q}$ is the state weighting matrix and $\mathbf{R}$ is the control weighting matrix. The selection of $\mathbf{Q}$ and $\mathbf{R}$ affects the performance of the controller, and they can be tuned using PSO to achieve the desired transient response and steady-state error. The PSO algorithm optimizes the elements of $\mathbf{Q}$ and $\mathbf{R}$ by minimizing a cost function that considers overshoot, settling time, and steady-state error.

A. Particle Swarm Optimization

The PSO algorithm is a population-based optimization technique inspired by the social behavior of birds and fish. It consists of a swarm of particles, where each particle represents a potential solution to the optimization problem. The particles move through the search space, updating their positions based on their own experience and the experience of their neighbors. The velocity and position of each particle are updated using the following equations:

$$
\begin{aligned}
v_i(t+1) &= w v_i(t) + c_1 r_1 (pbest_i - x_i(t)) \\
&\quad + c_2 r_2 (gbest - x_i(t)) \\
x_i(t+1) &= x_i(t) + v_i(t+1)
\end{aligned} \tag{12}
$$

where $v_i(t)$ is the velocity of particle i at time t, $x_i(t)$ is the position of particle i at time t, $pbest_i$ is the best position found by particle i, $gbest$ is the best position found by the entire swarm, w is the inertia weight, c_1 and c_2 are cognitive and social coefficients, and r_1 and r_2 are random numbers between 0 and 1. The PSO algorithm iteratively updates the positions and velocities of the particles until a stopping criterion is met, such as a maximum number of iterations or a satisfactory solution. The best solution found by the swarm is then used to design the state feedback controller for the HDT.

IV. Simulation Results

A. Grid Voltage Sag/Swell Compensation

B. Grid Voltage Harmonics Compensation

C. Load Unbalance Compensation

D. Non-linear Load Compensation

E. Load Harmonics Compensation

V. Conclusions

References

[1] A. Carreno, M. Malinowski, and M. A. Perez, “Circulating Active Power Flow and DC-Link Voltage Ripple in Hybrid Transformers,” in *IECON 2023- 49th Annual Conference of the IEEE Industrial Electronics Society*. Singapore, Singapore: IEEE, Oct. 2023, pp. 1–6.