POWERFACTORY

PowerFactory 2021

Technical Reference

Fourier Source

ElmFsrc

PF2021

POWER SYSTEM SOLUTIONS

MADE IN GERMANY

Publisher:
DIgSILENT GmbH
Heinrich-Hertz-Straße 9
72810 Gomaringen / Germany
Tel.: +49 (0) 7072-9168-0
Fax: +49 (0) 7072-9168-88
info@digsilent.de

Please visit our homepage at:
https://www.digsilent.de

Copyright © 2020 DIgSILENT GmbH
All rights reserved. No part of this
publication may be reproduced or
distributed in any form without written
permission of DIgSILENT GmbH.

December 1, 2020
PowerFactory 2021
Revision 1

Contents

1 Introduction

The *Fourier Source Element* (ElmFsrc) allows the definition of periodical signals in the frequency domain. It can be connected to any other dynamic *PowerFactory* model, especially to voltage or current source models, thus realizing harmonic voltage or current sources. The element may be used in both the balanced and three phases RMS simulation and in the three phases EMT simulation as well.

Typical applications are:

- Harmonic voltage or current sources for modelling harmonic injections
- Small signal analysis, calculation of transfer functions

2 General Description

Figure 2.1 shows the data and diagram pages of the element dialogue. The user may define minimum frequency and a frequency step size of the harmonic spectrum. For the input of data at every harmonic frequency, additional "cells" need to be appended.

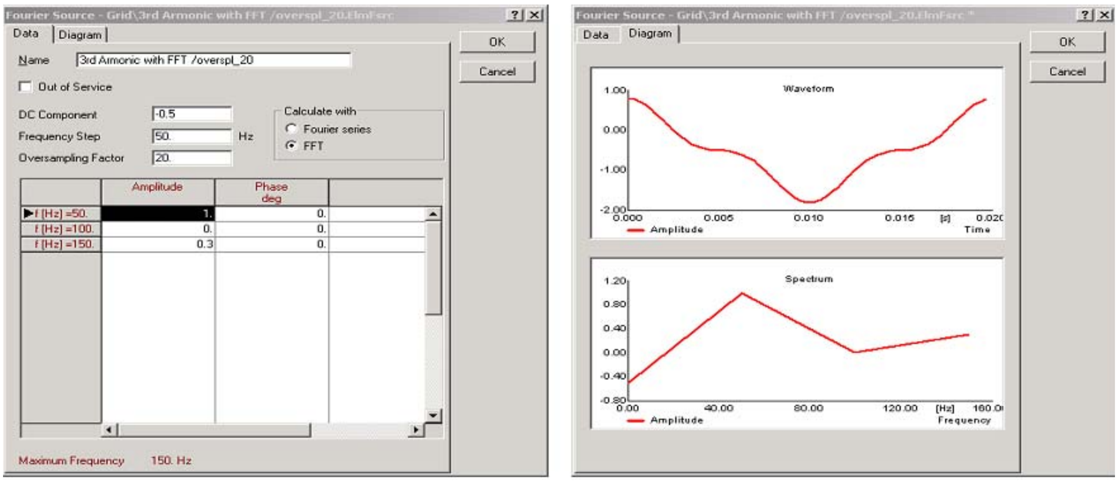

Figure 2.1: The data and diagram pages in the ElmFsrc dialogue

Two methods for generating a time domain signal from the specified spectrum are supported by the Fourier source. These are

- General Fourier series
- Inverse FFT (Fast Fourier Transform)

Both methods are briefly described in the following sections.

The input parameters vary according to the selected calculation approach. While dc-component and frequency step are always to be specified in both cases, a minimum frequency is additionally required for the Fourier Series approach and an over-sampling factor for the FFT one.

For checking purposes, the specified periodic signal is immediately shown in time- and frequency domain on the "Diagram" page of the input dialogue box.

2.1 Modelling Approaches

2.1.1 Fourier Series Modelling Approach

When the "Fourier Series" approach is selected, the output signal $y_0(t)$ is calculated by means of a Fourier series, as shown in equation (1):

$$y_0(t) = A_0 + \sum_{i=1}^{n} A_i \cdot \cos[2\pi(f_{min} + (i-1) \cdot \Delta f) \cdot t + \varphi_i] \tag{1}$$

where Δf is the frequency step, A_0 the dc component and A_i and φ_i the amplitude and phase of the i_{th}-harmonic.

y_0 defined by (1) is a continuous time domain function based on the specified spectrum. However, many cosine-terms must be evaluated at every time step during a transient simulation, which can lead to slow calculation times in case of many specified frequencies.

2.1.2 Fast Fourier Transform Modelling Approach

In this approach *PowerFactory* calculates the output signal waveform y_0 by means of the inverse Fast Fourier Transform (iFFT) algorithm, which is applied to the discrete spectrum. Unlike the Fourier Series approach, the iFFT must be carried out only once at the beginning of a transient simulation why computational resources are used more efficiently. However, the resulting output signal y_0 is a discrete time function and its time step will generally not match the simulation step size. This means that for each simulation time step the value of y_0 has to be interpolated. This introduces an interpolation error. This interpolation error can be reduced by applying an over-sampling factor, as described below.

The FF Transform pair is defined by equations (2) and (3):

$$Y_0(\omega_k) = Y_0(\frac{2\pi}{N \cdot T_s} \cdot k) = \sum_{n=0}^{N-1} y_0(n \cdot T_s) \cdot e^{-j\omega_k n T_s} \tag{2}$$

$$y_0(t_n) = y_0(n \cdot T_s) = \frac{1}{N} \cdot \sum_{k=0}^{N-1} Y_0(\omega_k) \cdot e^{j\omega_k n T_s} \tag{3}$$

Subscripts n and k represent the discrete time t_n and discrete frequency ω_k respectively and are permitted to range between 0 and $(N-1)$, where N is the number of samples. The following variables correspond to the FFT definition of (2) and (3):

$$T_s = \frac{1}{f_s} = \frac{T_{window}}{N} \tag{4}$$

$$t_n = n \cdot T_s \tag{5}$$

$$f_0 = \frac{1}{T_{window}} = \frac{1}{n \cdot T_s} = \frac{f_s}{n} \quad (6)$$

$$\omega_0 = 2\pi \cdot f_0 = \frac{2\pi}{n \cdot T_s} \quad (7)$$

$$\omega_k = k \cdot \omega_0 = (\frac{2\pi}{n \cdot T_s}) \cdot k \quad (8)$$

As it can be seen from (6), the windowing time T_{window} determines the minimum frequency of the discrete frequency spectrum only, i.e. the frequency resolution or frequency step. Furthermore, T_{window} has no relationship to the maximal frequency.

The maximal frequency is related to the sampling frequency f_s, i.e. to the sampling time T_s as defined in (4). Due to the time sampling, the FFT results in a periodic sequence of uniformly spaced frequencies with a period $f_s = 1/T_s$. Therefore, the sampling frequency must be chosen in such a way, that no superposition of subsequent harmonic spectrums (aliasing) will occur (comply with the sampling theorem). Factors between 5 and 20 seem to be appropriate in most cases. *PowerFactory* enables the user to specify this ratio by means of the over-sampling factor OSF (see Figure 2.2) defined in (9):

$$OSF = \frac{f_s}{2 \cdot f_{max}} \quad (9)$$

where f_{max} is the maximal frequency of the spectrum.

The default value for the over-sampling factor is 10 and the user may increase it as necessary. However, it must be pointed out, that with an increasing over-sampling factor also the number of samples N increases and consequently increasing computational time.

The diagram page in the element dialogue (see Fig. 2.2) helps the user to select a suitable over-sampling factor. By modifying the over-sampling factor in the data page, the curve in the diagram is automatically updated. With the zoom tools the user may analyse whether the approximation is good enough or a higher over-sampling factor is still required.

Figure 2.2 depicts an example of the FFT modelling approach. The curves on the right side are a zoom near the peak value of those on the left side.

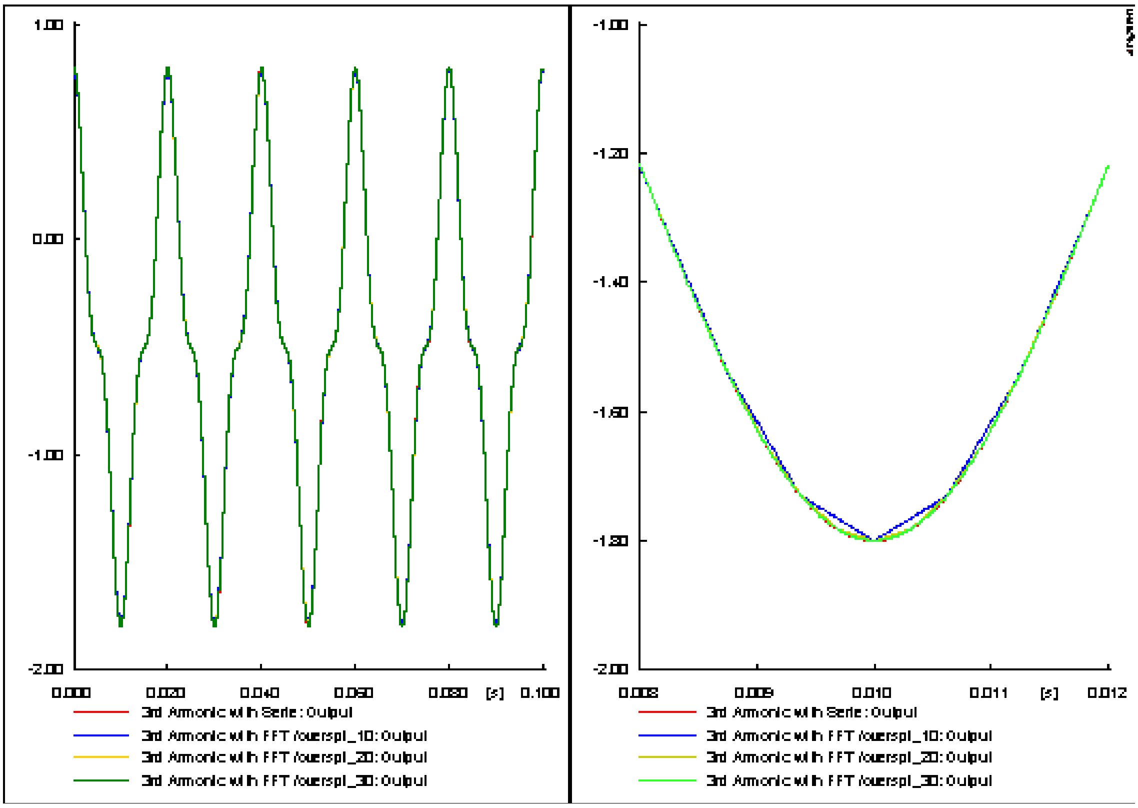

Figure 2.2: Influence of the over-sampling factor

Regardless of whether the time discrete output signal y_0 is a finite-length sequence or a periodic sequence, the FFT treats the N samples of y_0 as though they characterise one period of a periodic sequence. This period corresponds to the inverse of the frequency step defined by the user.

For each transient simulation time step, the output value of y_0 is linearly interpolated between those values resulting from the iFFT. This interpolation lets introduce an additional selection criterion for the over-sampling factor. It seems reasonably, that the sampling time T_s in Equation (6) not be in any case smaller than the simulation step size T_{step} and therefore, the over-sampling factor may be determined as following:

$$OSF = \frac{1}{2 \cdot f_{max} \cdot T_{step}} \tag{10}$$

A Model Parameters

Table A.1: Parameters of AC Current Source

Parameter	Description	Unit
loc_name	Name	
outserv	Out of Service	
dc_com	DC - Component	
delta_f	Frequency Step	Hz
overspl	Over-sampling Factor (only for the FFT option)	
f_min	Minimum Frequency (only for Fourier Series option)	Hz
f_max	Maximum Frequency (only for Fourier Series option)	Hz
ampl_	Amplitude	
phase_	Phase	Grads

B Input/Output Signals

Table B.1: Output variables

Name	Description	Unit	Type	Model
yo	Output signal		OUT	EMT

List of Figures

List of Tables