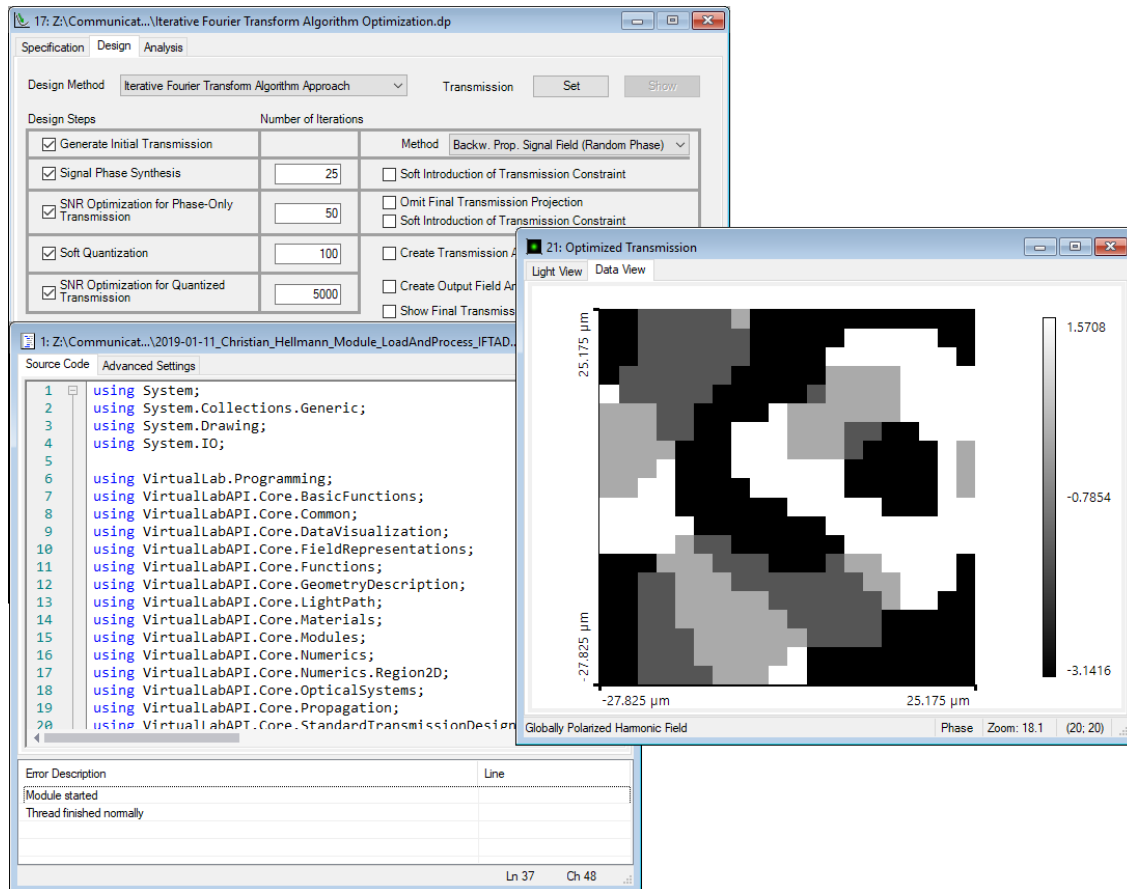


Programming of a Module for Executing an IFTA Design

Abstract



For the design of diffractive optical elements (DOEs), such as beam splitters, usually the iterative Fourier transform algorithm (IFTA) is applied. VirtualLab Fusion offers a step-by-step wizard for the configuration of all the design parameters. However, for some specific design tasks, it can be of interest, to be able to perform the algorithm in an automatized way and without the graphical user interface. Thus, in this document the execution of an IFTA design by using a customized C# module in VirtualLab Fusion is shown.

Task Description & Sample Code

Task:

Design of a diffractive beam splitter (e.g. 5×5) by applying the IFTA, without using the interface of the wizard. After the design process, the performance of the designed element has to be investigated.

Parameters (to be defined by user)

Variable	Description
<code>string</code> pathofIFTAInputData	defines the path of the used files
<code>string</code> filenameIFTA	name of the initial IFTA file
<code>string</code> filenameMeritFunctionValues	Name of the text file for data output

Main Function (first part, continued in the sample file)

```
namespace OwnCode {
    public class VLModule : IVLModule {
        //the path where all the data is located
        string pathofIFTAInputData = @"D:\IFTA Module\Example\";
        //file name of the IFTA document which should be loaded from hard disc
        string filenameIFTA = "Iterative Fourier Transform Algorithm Optimization.dp";
        //define filename for storage of merit function values
        string filenameMeritFunctionValues = "Result.txt";

        public void Run() {
            //load IFTA from hard disc
            DesignAlgorithmHandler design = DesignAlgorithmHandler.Load(Path.Combine(pathofIFTAInputData,
                filenameIFTA));
            //error handling
            if(design == null){
                Globals.DataDisplay.LogError("IFTA could not be loaded!");
                return;
            }

            //error handling
            if(caSignalField == null){
                Globals.DataDisplay.LogError("Signal could not be loaded!");
                return;
            }

            //read sampling parameters from design document
            SamplingParameters sPara = new SamplingParameters();
            sPara = new SamplingParameters(design.ConstraintSpecification.SamplingPoints,
                design.ConstraintSpecification.SamplingDistance);
        }
    }
}
```

Preparation of the IFTA setup

In order to run the module, an initial IFTA document has to be generated:

The image shows a software interface for designing optical systems. The main window is titled "25: Regular Beam Splitter Session Editor". It contains several tabs for configuring different aspects of the beam splitter, such as "Input Beam Parameters", "Optical Setup", "Output Field Shaper", "Morph Function", "Diffraction Order", "Off-Axis Design", and "Output Field Size". A red arrow points from the "Regular Beam Splitter" option in the "Module" menu to this window. Below the main window, a series of smaller, overlapping windows show the configuration steps for the "Regular Beam Splitter". A hand icon with the number "2" is pointing to the "Next >" button in the "Regular Beam Splitter Session Editor" window.

The "28: Iterative Fourier Transform Algorithm Optimization" dialog box is shown on the right. It has tabs for "Specification", "Design", and "Analysis". The "Design" tab is active, showing the "Design Method" as "Iterative Fourier Transform Algorithm Approach" and "Transmission" as "Set". The "Design Steps" table is as follows:

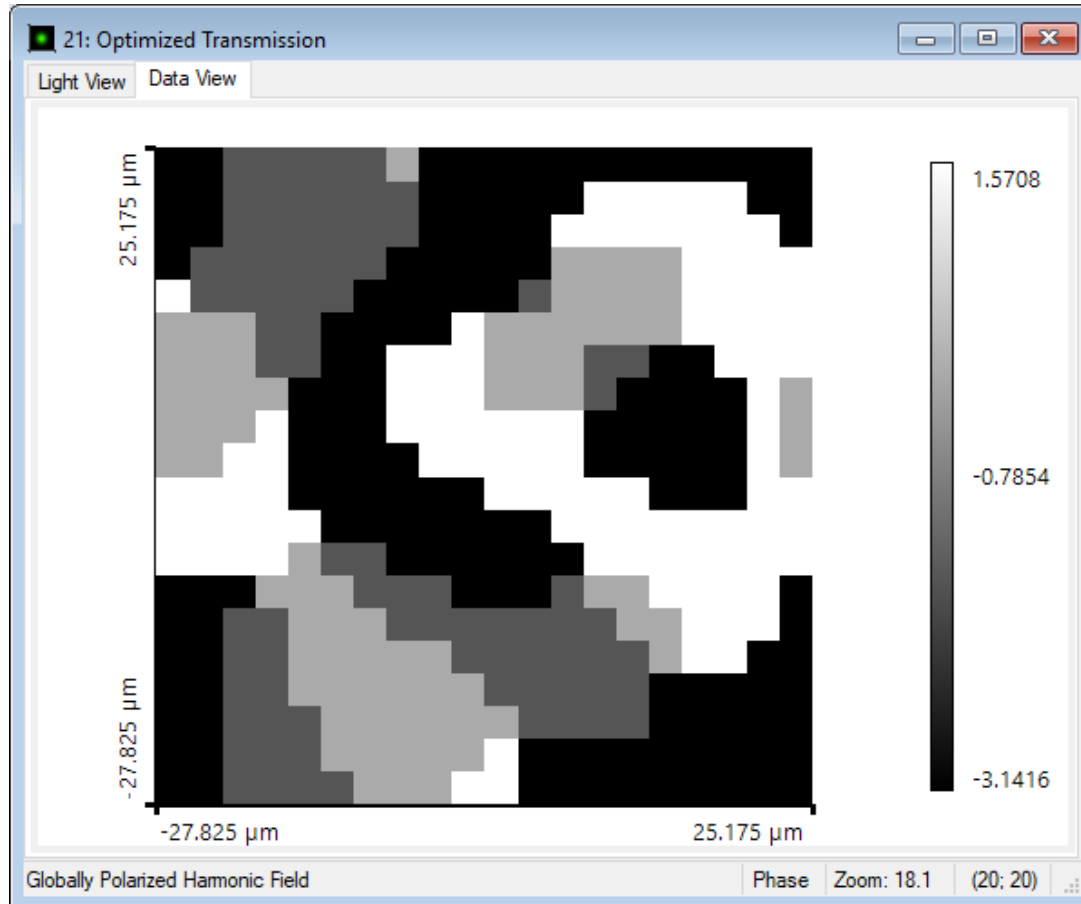
Design Steps	Number of Iterations	Method
<input checked="" type="checkbox"/> Generate Initial Transmission		Backw. Prop. Signal Field (Random Phase)
<input checked="" type="checkbox"/> Signal Phase Synthesis	25	<input type="checkbox"/> Soft Introduction of Transmission Constraint
<input checked="" type="checkbox"/> SNR Optimization for Phase-Only Transmission	50	<input type="checkbox"/> Omit Final Transmission Projection
<input checked="" type="checkbox"/> Soft Quantization	100	<input type="checkbox"/> Soft Introduction of Transmission Constraint
<input checked="" type="checkbox"/> SNR Optimization for Quantized Transmission	5000	<input type="checkbox"/> Create Transmission Animation
		<input type="checkbox"/> Create Output Field Animation
		<input type="checkbox"/> Show Final Transmission and Output Field

The "Logging" section has "Enable Logging" and "Preserve Table" options, both unchecked. The "Start Design" button is at the bottom right.

The different windows enable the configuration of the initial system, whose Parameter can be adapted by the module (in this example, the default settings are used.)

Results

resulting phase function



performance output in text file

Result.txt - Editor

Datei Bearbeiten Format Ansicht ?

Conversion Efficiency = 75.423 %

Uniformity Error = 13.261 %

Document Information

title	Programming of a Module for Executing an IFTA Design
document code	CZT.0110
version	1.0
toolbox(es)	Starter Toolbox, Diffractive Optics Toolbox Silver
VL version used for simulations	7.6.1.18
category	Feature Use Case
further reading	<ul style="list-style-type: none">- Customizable Help for Programmable Elements- How to Work with the Programmable Function & Example (Cylindrical Lens)