#### Setting up a Wave-Optics Model: Modeling a Gaussian Beam to a Focus

#### Justin Mansell, Ph.D.



1 Public Use

### Outline

- The goal of this effort is to outline the procedure for setting up a basic model in WaveTrain.
- We'll use the case of propagating a Gaussian beam to a focus as an example because it is simple enough to have an analytical solution against which we can compare our results.



## **Analytical Solution**

- Setup:
  - 1-mm waistGaussian
  - 1 m focal length lens
  - 1 µm wavelength



For a circular aperture (for comparison):

$$D_{spot} = 2.44 \frac{f \lambda}{D_{beam}} = 2.44 \frac{(1m)(1\mu m)}{(2mm)} = 1.22mm$$

For a Gaussian:

$$w_{spot} = \frac{f\lambda}{\pi w_{beam}} = \frac{(1m)(1\mu)}{\pi (1mm)} = 0.318mm$$
$$D_{spot} \approx 2w_{spot} = 0.637mm$$



## WaveTrain Model Setup

- 1. Determine the propagation mesh and method of propagation.
- 2. Setup the model.
- 3. Analyze the results.



## **Propagation Introduction**

- Wave-optics models light as a 2D grid of complex numbers which represent magnitude and phase of a beam of light.
  - This grid is specified by the sample spacing ( $\delta$ ) and the number of samples in one dimension (N).
- WaveTrain uses a convolution propagator (see J. Goodman's <u>Fourier Optics</u> Ch. 4) which involves a 2D Fourier transform of the field, multiplication by a propagation kernel, and then a 2D inverse Fourier transform.
  - This is sometimes referred to as a 2-step propagator
  - It is done this way to control the mesh spacing in a way that a simple 1-step propagator does not allow



N=4



## Representing a Field on a Mesh

- When choosing mesh parameters for wave-optics modeling, Nyquist sampling theory needs to be applied to not only the amplitude, but the phase as well.
- We have derived equations for determining the mesh parameters based on the propagation setup.
  - The propagation setup requires specifying a diameter of interest in the first and final planes, a wavelength, and a distance between the two planes.
  - We have also included the capability of including the effects of atmospheric aberrations, but the problem we are addressing here does not require this capability, so it will not be substantially addressed here.
- These relationships are codified into an Excel worksheet that we will introduce later.

#### Illustration of Propagation Specification





#### **Illustration of Propagation Parameters**



jmansell@mza.com



#### **SWP vs PWP**

- The 2-step Fourier propagation as described above is done relative to a planar wavefront reference, which enables the mesh spacing to be preserved.
  - This is called plane wave propagation (PWP).
- In some problems, like propagation to a focus, it makes sense to propagate relative to a reference curvature for two reasons:
  - This allows the mesh sample spacing to be changed during the propagation.
  - This reduces the number of mesh points required because the curvature is no longer required to be represented on the mesh (again Nyquist sampling requirements).
- This is called spherical wave propagation (SWP).

#### **Illustration of a Diverging SWP**





#### **Propagation Setup**

- MZA provides for free on its website a worksheet for calculating the mesh required for a given propagation.
- We will be working with a simplified version of this worksheet shown on the right.

#### Wave-Optics Propagation Geometry Tool

Basic Inputs			Color Key
λ	1.03E-06		u ser in put
Δz	2,000		advanced inpu
D <sub>1</sub>	0.15		calculated
D <sub>2</sub>	0.27		
Basic Turbulence	e Inputs		
Cturb	3		
r <sub>01</sub>	0.05		
r <sub>02</sub>	0.05		
Advanced Turbu	lence Inputs	Calculated	
θ <sub>1 blur</sub>	0.0000618	6.18E-05	
θ <sub>2 blur</sub>	0.0000618	6.18E-05	
D <sub>1</sub>	0.3972	0.2736	
D <sub>2</sub> '	0.3936	0.3936	







#### Worksheet Setup

#### Wave-Optics Propagation Geometry Tool







# **Propagation Worksheet Results**

- We use the fast Fourier transform (fft) to implement our propagator in WaveTrain, which requires that the number of points be a power of two.
  - The N<sub>FFT</sub> parameter is the nearest power of two.
- There are three different propagation parameters given in the worksheet:
  - Minimizing N choose these parameters to minimize the number of samples required.
  - Plane Wave choose these parameters to get proper sampling for a planar reference wave propagation.
  - General these parameters provide the user the ability to pick both the mesh sample spacing in the input and output planes, but checks them against the minimum sample spacing in each plane to ensure proper sampling. This is good for trying to have one propagation match the next propagations mesh.

	Fiftering
Minimizing N	-
δ <sub>1</sub> =	8.33333E-05
$\delta_2 =$	8.33333E-05
N≥	144
N <sub>FFT</sub> =	256
Plane Wave	
$\delta_1 = \delta_2 < 0$	8.33333E-05
$\delta_1 = \delta_2 =$	8.30E-05
N≥	144.8686312
N <sub>FFT</sub> =	256
General	
δ <sub>1</sub> =	8.00E-05
$\delta_2 < (dependent)$	8.66667E-05
δ <sub>2</sub> =	8.00E-05
$\delta_1 < (dependent)$	8.66667E-05
N≥	153,125
N <sub>fft</sub> =	256



## **Propagation Decisions**

- Because the beam does not significantly change size during the propagation, we chose to start with a plane wave propagation (PWP).
- Based on the worksheet results, we chose to use the parameters from the minimizing N, which were plane wave ( $\delta_1 = \delta_2$ )

-N = 256

 $-\delta = 83 \mu m$ 

#### Setting up the WaveTrain Model



## **Model Setup Introduction**

- We are presenting here one way of setting up this model.
- Other tutorials are available that describe other ways of doing this same type of model setup



## **Open the System Editor**

Open the tve & click on the system editor icon.



 This should open up the following window:



## Add a Source 1/3

 Right click in the window and choose "Add SubSystem", "System in Library", and then choose your wtlib directory.

🐱 Editing: NewSystem (unspecified	path)		
File Edit Navigation View Status Op	otions Window Help		
🗋 😂 🖬 🖻 🍀 💥 🖻 🖥 🗩 🛛	٨ ጱ 🕾 😢 🗉 ا		
Up one level			
Droperties of NewSystem			
Paste			
Add Subsystem	System in Library 🔸	C:\MZA\wavetrain\v2000.1	1\wtlib
Add Input	System in Directory 🕨	c:\mza\wavetrain\v2000.11	1h\canlib
CONTRACT Add Output	Browse	c:\mza\wavetrain\v2000.11	Th\canlib\
Add Parameter		C:\MZA\Wavetrain\v2000.	11H\wtlib
Add Note			
Help			· · · · · · · · · · ·
Set Connections Color			
	· · · · · · · · · · · · · · · · · · ·		
•			
NewSystem::		-	Hierarchy st



#### Add a Source 2/3

Choose "GaussianCWLaser" from the list of files.



## Add Source 3/3

- Place the source on the left part of the screen.
- Show its
  - inputs,
  - outputs, and
  - parameters
  - by clicking on the images below the grey box.





#### Exposing types, names, and values

- Sometimes the software is left in a state where the data type, name, and/or value has been hidden from view.
- To expose these for inputs, outputs or parameters, click on the IO icons on the toolbar.
- A menu of icons will drop down. The rows correspond to the title bar, the inputs, the outputs, and the parameters respectively. The columns correspond to hiding and showing the list, types (t), the names (n), and values (v).





#### Add the Propagation Controller

• Use the same technique to add a PropagationController.

Editing: NewSystem (unspecified patient of the system)	ith) - modified			
File Edit Navigation View Status Optio	ns Window Helj	p		
) 🚘 🔲 🖻 🛍 💥 53 🖻 🗟 🗩 🗩	- III 💀 💀 🔦	2		
		FF¥		
· · · · · · · · · · · · · · · · · · ·			· · · · · · · · · · · · · · · · · · ·	
· · · · · · · · · · · <b>D D D ?</b> · · · · · · · · ·		••••••	<b>•</b> • • • • • • • • • • • • • • • • • •	
GaussianCwLaser gaussiancwlaser		PropagationContro	ller propagationcontroller	
hool on		WaveTrain in	ucident WaveTrain∩	
Waya Train transmitted		Wave T	rain transmitted	
flast noway	CridCoomotry	towest Crid	grid with origin on machinementy new	dan)
Aret man les et	GildGeometry	-Deference France	gra_wincongin_on_messaprophys,prop	uxy)
	noat	AREIEIEncerocus	0.0	
float apertureKadius	float	yKeterenceFocus	0.0	
float annulusRadius 0.0	. const Filter&	one Time SpatialFilte	er NullFilter()	
float sigma (apertureRadius*(6.0/7.0))	const Filter&	spatialFilter	NullFilter()	
float gridWidth	const Filter&	absorbingBoundary	NullFilter()	
int gridDim	PointSourceMod	el pointSourceModel	DEFAULT_PSM	
	SpeckleModel	speckleModel	DEFAULT_SM	
	float	superApDiameter		
	float	edge Sigma	0.0	
	· · bool	useDispersion	false	
	float	nominalWavelength	0.0	
			· · · · · ·	•
ewSystem::			Hierarchy status: 🥥	– System status: 🤗

#### Commentary on PropagationController

- The PropagationController allows you to specify various properties of the light model including mesh parameters.
- Every source needs a PropagationController between it and a sensor.
  - It is best to put the PropagationController just after the source.



### **PropagationController Parameters**

- The parameters list for each component specifies the type of parameter, the parameter name, and the parameter value.
- Click on the area to the right of the parameter name (the parameter value) to get a dialog box for editing it.
- Start by changing the targetGrid parameter value to "gwoom(nxy,dxy)".
- Upon pushing enter you will see the screen on the next slide pop-up...

Propagatio	nController propagatio	oncontroller
Vave 🗸	Train incident WaveT	rain()
	WaveTrain transmitte	d
GridGeometry	targetGrid	gwoom(nxy,dxy)
float	xReferenceFocus	0.0
float	yReferenceFocus	0.0
const Filter&	one Time Spatial Filte:	r NullFilter()
const Filter&	spatialFilter	NullFilter()
const Filter&	absorbingBoundary	NullFilter()
PointSourceMode	l pointSourceModel	DEFAULT_PSM
SpeckleModel	speckleModel	DEFAULT_SM
float	superApDiameter	0.7*nxy*dxy
float	edge Sigma	0.0
bool	useDispersion	false
float	nominalWavelength	wavelength



#### **Parameter Addition**

- Change the nxy type to int from float.
- Enter a value for nxy of 256.
- Enter a value for dxy of 83e-6.
- Push enter after each enter or the values will not be registered.
- Click on the "Add add Parameter" button twice (once for each of them) to add them as system parameters.

wooni(nxy,uxy)			
Туре	Name	Value	Description
it	nxy	256	
pat	dxy	83e-6	



## **PropagationController Parameters**

- Using the same procedure, change the superApDiameter to 0.7\*nxy\*dxy.
- Change the nominalWavelength to a new parameter called wavelength, and set its initial value to 1.0e-6.

	:: > > - ? ;	<u></u>						
Propagati	onController propagatio	oncontroller						
WaveTrain incident WaveTrain()								
	WaveTrain transmitte	.d						
GridGeometry	targetGrid	gwoom(nxy,dxy)						
float	xReferenceFocus	0.0						
float	yReferenceFocus	0.0						
const Filter&	one Time Spatial Filte	r NullFilter()						
const Filter&	spatialFilter	NullFilter()						
const Filter&	absorbingBoundary	NullFilter()						
PointSourceMod	el pointSourceModel	DEFAULT_PSM						
SpeckleModel	speckleModel	DEFAULT_SM						
float	superApDiameter	0.7*nxy*dxy						
float	edge Sigma	0.0						
bool	useDispersion	false						
float	nominalWavelength	wavelength						



#### **GaussianCWLaser Parameters**

- Set the power to 1.0.
- Right click on the wavelength parameter and choose "Elevate".
  - This will link the wavelength of this component to the same as the value specified for the

PropagationController.

 Set the remaining values to those specified on the right.



## **Connecting the Source and the PC**

 Drag the tip of the "transmitted" output from the GaussianCWLaser to the "incident" input on the PropagationController.

	• • •				÷ +	· •	· •	· ·	· ·				
1.1		<u></u>		1.1	• •	• •	· ·	• •	1.1				
				1.1	• •		· ·						
11				1.1	2.2	11	2.2	2.2	11				
1.1						11		11	11				
	Con	eeion Curl seen a	auccianculaca							· · · · Propagation	Controller propagation	acontroller	
	Jau	ssiditen Laser e		·							controller propagation	CONTOLLCI	
	• • •	· · · · bool on			· •	· •	· •	· ·		Wave T		uno e e e e e e e	
				1.1	· · ·	<u></u>			1.1				
1.1		WaveTrain tr	ansmitted			· · ·	· ·		1.1		Wave Train transmitted	••••••••••••••	
	flee	t nowar	10							CridCoomotry	to prot Crid	ana antara dara	
11	шоа	it power	1.0		2.2	11	2.2	2.2	2.2	GiluGeometry	largeroriu	gwooni(nxy,uxy)	
	floa	t wavelength	wavelength							float	xReferenceFocus	0.0	
			10	1 A .			- ·			a .	<b>D</b> ( <b>D</b>		
	tioa	it apertureKadiu	LS 1.U	1 A A	• •	• •	• •	• •	• •	float	yKeterenceFocus	0.0	
1.1	floa	t annulusRadiu	s 0.0	1.1	· · ·	· · ·	• •	• •		const Filter&	oneTimeSnatialFilter	NullFilter()	
11				11.1	11	11	11	11	11		one rane spunde and		
	floa	it sigma	1.0e-3	1						const Filter&	spatialFilter	NullFilter()	
	flee	+ and Wildth	naar#daar							const Filton®	absorbingDoundor	NullFilter/	
	шоа	ii griuwiuui	плу-шлу	1 A S	÷ +	· •	· •	• •	· •	constructes	ansonningpommary	nummerty	
1.1	int	gridDim	nxy	1.1	• •	• •	• •	• •	1.1	PointSourceModel	pointSourceModel	DEFAULT PSM	
		0	,		1.1	1.1			1.1	a		-	
11	1.1.1			1.1	2.2	11	2.2	2.2	11	SpeckleModel	speckleModel	DEFAULT_SM	
										float	eimeránDiameter	0.7*nov*dov	
										Hoat	superinplanteter	o., my ary	
	• • •			(+,+)	· •	· •	· ·	• •	· •	float	edgeSigma	0.0	
1.1	• • •			1.1	· •	· •	• •	• •	1.1	1 1	n	C 1	
	• • •				• •	• •	• •	• •		0001	useDispersion	Talse	
11				1.1	11	11	11	11	11	float	nominalWavelength	wavelength	
									11		Adminiation aventing bit	narozongut	
					· ·	· •	÷ +	· •	÷ +		· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · ·	



#### **Status Check**

#### Your system should now look like this:

🟅 Editing: NewSystem (unspecifi	ied path) - modifie	d			
File Edit Navigation View Status	Options Window	Help			
🗋 🚅 🔲 🖻 🛍 💥 53 💼 📓 🗩	> 🚽 🗐 😥 👳	<b>9 9</b>			
		Propagation		ncontroller	
baal an	<b>.</b>	Tiopagano	reonioner propagano Frain incident WaveTr		
WaveTrain transmitted	· · · · · · · · · · · · · · · · · · ·	· · · · · · · · · · · · · ·	WaveTrain transmitte		
float power 1.0	7	GridGeometry	targetGrid	gwoom(nxy,dxy)	
float wavelength wavelength		float	xReferenceFocus	0.0	
float apertureRadius 1.0		float	yReferenceFocus	0.0	
float annulusRadius 0.0		const Filter&	one Time Spatial Filter	NullFilter()	
float sigma 1.0e-3		const Filter&	spatialFilter	NullFilter()	
float gridWidth nxy*dxy		const Filter&	absorbingBoundary	NullFilter()	
int gridDim nxy		PointSourceModel	pointSourceModel	DEFAULT_PSM	
		SpeckleModel	speckleModel	DEFAULT_SM	
		float	superApDiameter	0.7*nxy*dxy	
		float	edge Sigma	0.0	
		0001	useDispersion	Taise	
	• • • • • • • • • • • • • • •		nonman wavelengur	wavelengut	
				• • • • • • • • • • • • • •	<b>_</b>
•					
NewSystem::				Hierarchy status:	System status:



#### Setting an Input to a System Parameter

- Inputs can be set to constant values or as system parameters.
- If an input has no default value, it will appear like this.
- To add a value to this input, click on the area to the right of the name and an edit box will appear.
- Begin typing in this box even though the text will not appear until you press enter.
  - This is a bug in the JAVA interface code.



gaussiancwlase

gaussiancwlas



#### Alternative Method of Setting Inputs, Outputs, and Parameters

- Right click on the system and choose "Properties of gaussiancwlaser".
- A dialog box will then appear that provides a tabular input on the "Interface" tab.

	<u> </u>
	Open GaussianCwLaser
gaussia) bool or	Copy gaussiancwlaser
	Properties of gaussiancwlaser
	Edit Name
	Danlaca euliewetam 🔶

🕻 Properties of GaussianCwLaser gaussiancwlaser 🛛 🗙									
General Interface Description									
Inputs									
Туре	Name	Setting Expr.	Default Value	Description					
bool	on	true		Flag indicati					
   • •									
		Outputs							
Туре	Name	Setting Expr.	Default Value	Description					
Wa∨eTrain	transmitted			The generat					
!  • •									
		Parameters							
Туре	Name	Setting Expr.	Default Value	Description					
float	power	1.0		Power (Wat					
float	wavelength	wavelength		Wavelength					
float	apertureRa	1.0		Circular ape					
float	annulusRadius	0.0	0.0	Circular ann					
float	sigma	1.0e-3	(apertureRa	Radius of 1					
float	gridWidth	nxy*dxy		Width of ou					
int	gridDim	nxy		Number of					
J									
OK Cancel Help									



#### Give yourself more room

- Give yourself more working room by turning off the component types using the icon menu icon shown below and choosing the grey "t".
- Hide the inputs, outputs, and parameters of the components.

🐱 Editing: NewSystem (unspecified path) - modified	<u> </u>
File Edit Navigation View Status Options Window Help	
🗋 😂 🖶 🛍 💥 🔀 🔁 🖫 🤉 🗩 🛓 🗒 🗟 🏖 🤽 🔧	
Add	
gaussiancwlaser	
	1



#### Adding a Focus

- Add a "Focus" from the wtlib library.
- Connect the PropagationController output to the focus "outgoingIncident" input.
- Change the focus distance input value to read "recallableFloat(1.0)".



## **Commentary on recallableFloat()**

- tempus (the backbone modeling package for WaveTrain) sometimes needs to look back in time to determine a value of a variable or input, so a type of inputs was created called "recallables".
- Most systems do not need this functionality, so a conversion routine was developed called "recallableFloat()" that converts a float to a recallable<float>
- To use this routine, you need to include the file "RecallableFunctions.h" in the "C++ Code" tab of the system properties.
  - From the main menu, select File-Properties
  - Click on the C++ Code tab
  - Enter: #include "tnl/RecallableFunctions.h"

tve: P	roperties c	of Class Test			_Σ
General	Interface	C++ Code	Description	Error/Warning detection	
#include	"tnl/Recalla	bleFunctions	.h*		
an reserve	unyi teedina				



## **Commentary on outgoingIncident**

- "Outgoing" and "incoming" inputs and outputs were designed to make optics bidirectional.
- The present system uses only "outgoing" waves. (Internally, "outgoing" is associated with the Cartesian +z direction).
- For more complex WaveTrain systems that involve light propagating in both directions, it is typical to use both "outgoing" and "incoming" waves processed by the same optical and propagation components.



#### **Commentary on Sources and Sensors**

- Light in WaveTrain always travels from a source to a sensor.
  - Every model needs at least one source and one sensor.
- Sensors initiate the modeling by requesting light from all sources attached to a given sensor.



#### Add a VacuumProp

- Add a vacuumProp and connect it to the outgoingTransmitted output of the focus.
- Change the propagationDistance input to "recallableFloat(1.0)".





#### Add a SimpleFieldSensor

- Add a simpleFieldSensor to the system and connect it to the outgoingTransmitted output of the vacuumProp component.
- Set the parameters as shown below.

•							• •	•	+	• •	• •		÷	÷	• •		•	• •	÷	÷.,	• •		+	•	• •	÷	•	• •	+	• •	•		• •	•	• •		• •	•	•	• •		• •		· ·	• •			•		• •			• •	· ·
ļ	ł	-			H	1	· ·	•	1			÷	•		H	- 1	• •	· ·	-	1				$\left  \right $		•		· ·	1		•	•		-		j			-		ł				: :				÷		: :			: :
t	1	$\sum$	►			1?		1	÷	:	Σ	I				1	2	: :	1	Σ		►				?	:	: :	÷	Σ		•			?	Ľ	: :		1	• •	1	: :	11	11	: :	Σ				?	11	11	11	1.1
ł	g	aus	sia	inc	wł	ase	r	-	P	го	pa	ga	tio	n	on	t	oll	er	ł	:	Fo	cu	s f	òc	us	Ŀ	÷	V	acu	IW	nP	гор	va	cu	un	φı	op		-		-		Si	mple	Fi	eldS	5ens	ог :	siny	ple	field	sen	SOT	
•	•	•	-	•	• •	•	• •	•	•	• •	• •		•	•	• •	•	•	• •	•	•	• •	•	·	•	• •		Wa	we	Гга	iin	i	100	mi	ng]	Гга	ins	mi	tte	Ð	• •		V	Vave	Trai	n i	inci	deni	ł			Wav	eTr	ain(	5
-	-		1	2		1			2			-	÷	2			2		-	2		-	2				Wa	we	Гга	ain	. 01	utg	oir	gT	ra	nsi	mit	ted		Ζ.	Ð	> b	ool			on					true			÷
ļ	1		1	1	: :	1	: :		1	1		1	Ì,	1	: :	1	1	: :	1	2	: :	1	1			l			1			1				1			1		5	> d	oub	le		ехро	osur	eIn	terv	al	0.00	1		÷
Ĵ	1	1	1	1	1	1	1	1	1	1		1	ĵ,	1	: :	1	1	: :	1	1	: :	Ĵ	1		1	1	:	1	1	1		1	: :	:	1	1	1		1	: :	1	> d	oub	e		ехро	sur	eLe	ngt	h I	0.00	000]	1	
ł	ł		1	1	: :	1	: :	1	1	: :		1	ĵ,	ŗ.	: :	1	1	: :	÷	1	: :	ł	1			÷	:	: :	÷	: :		1	: :	:	: :	1	: :		1		1	> <b>d</b>	oub	le		sam	pleI	nte	rval		0.0			
•	•	•	÷	÷	• •	•	• •	•	÷	•		÷	÷	÷	• •	÷	•	• •	÷	•	• •	÷	÷	•	• •	÷	•	• •	÷	• •	•	÷	• •	•	• •		•	•	÷	• •		• •	• •	floa	atı	wav	elen	gth	wa	vel	engi	th	• •	•
ļ	-		4	-		1			1			4	÷	÷		-	-		÷	-		ł	2			-			-			2				1					2			int		nxv		9	nx	v	8.			
Ĵ	1		1	1	: :	1	: :	1	1	1		1	Ĵ,	1	: :	1	1	: :	1	1	: :	1	1			1			1	1		1	: :		: :	1	2.3		1		1	: :	11	flag	a <b>+</b> .	daar			ماد. مادر	, 				11
•		-	1	•	• •	•	• •	•	•	• •	• •	1	1	•	• •	•	÷	• •	1	•	• •		•	•	• •	•	•	• •	•	• •	•	1	• •	•	• •	1	• •	•	•	• •	÷	• •		ща	at	uxy			- avi	y			• •	• •



#### **Commentary on the SimpleFieldSensor**

- The simple field sensor has a variety of inputs corresponding to the timing of the sensor.
- The exposureInterval is the time between exposures.
- The exposureLength is the amount of time that the sensor is exposed to light (shutter open time).
- The sample interval is the time between samples that are averaged on the sensor
  - 0.0 means that one sample is taken at the center of the exposureLength.
- The boolean on input can be driven externally in some situations to delay the on-time in order to stage the exposure of various sensors.

1	1	1	1	1	· •	· · ·	1.1		1	1		1	1
1	1	1	1	1	· •				1	1		1	1
1	1	1	1	1					1	1		1	1
1	1	1							1				1
1													
ĵ,	1	2	2	1					1	2.1		1	
	1								1				
							£ .	•					
÷				•		einnlafialdeane	от	.					
÷						simplementsens	101	•					
Þ	W	/a	ve'	Гга	iin	incident	W	lav	eΤ	ra	in(	)	ŀ
	1		1										
٢.	100	50	_			011		ue					
Σ	do	ou	հհ	e		exposureInterval	1 0.	00	1				ŀ
S	de		հե	_		exposure Length	0	nn	nni	11			
١.													
		<i>.</i>		8		- Providente and and a							
Σ	do	ou	հե	e e		sampleInterval	0.	0					
2	do	ou	bl	e e		sampleInterval Grid <complex></complex>	0. flo	0				:	
>	de	ou	bl	e e		sampleInterval Grid <complex></complex>	0. flo	0				:	
>	do	• u	bl	e fl	oat	sampleInterval Grid <complex> wavelength wav</complex>	0. fla ele:	0 D	: th				
> 	<b>d</b> o		bl	e fl	oat	sampleInterval Grid <complex> wavelength wave</complex>	0. flo ele:	0 1	: th		· · · · ·		
	<b>d</b> o	• u	<b>bl</b>	fl	oat ut	sampleInterval Grid <complex> wavelength wave</complex>	0. file ele:	0 ng	: th			- - - - -	
	<b>d</b>		<b>bl</b>	e fl in fl	oat it oat	sampleInterval Grid <complex> wavelength wave nxy nxy dxy dxy</complex>	0. flo ele:	0 1	: th			- - - - - - - - - - - -	



#### **Inserting Icons**

- Icons can make the system more easy to understand at a glance.
- To add an icon to a component, right click on that component and choose "Edit Icon".
- WaveTrain comes with a variety of icons for making a system more easily readable.





#### **Parameter Viewing**

 To view the system parameters at a glance at the bottom of the screen, go to View-Parameters.





#### Saving the System

- Once the system has been created, select File-Save or Save As to save the system.
- Save the system in a path without spaces in the directory structure or the file name.
- Do not put spaces in the file name!
- It is recommended to save the systems in a "c:\Simulations" directory.
  - For reference, we saved this as "c:\simulations\test\test.tsd"
  - \*.tsd are tempus system description files



#### **Creating a Runset**



43 Public Use

#### **Runset Creation**

- Now you are ready to create a runset.
- From the system editor, go to File-New-Runset for test.
- Enter a name for the runset.
  - we used "test" as the name
- The tempus runset editor (TRE) will pop-up.

ĕ	rre: te	st (for C:\Sin	nulations	s\test\Test)							
Eile	Edit <u>V</u> i	ew <u>B</u> uild <u>O</u> pt	tions Wi	indow <u>H</u> elp				×			
	2 🖬 🛛	X 🖻 🖹 🕇	- + +	🔹 😫 🚦 🖥 (1)   Stop Time:	0.00						
	Run Variables										
		Туре		Name		Value	Desc	ription			
<u></u>											
				System Par-	ameters	1					
	Туре	Name		Value			Description				
1	int	nxy	256								
2	float	dxy	83e-6								
3	float	wavelength	1.0e-6								
P				Modified		Hierarch	ıy status: 🧕	Run status: 🧕			



## **Commentary on Runset Names**

- A tempus results file (trf) is created each time a tempus run is made.
- The code generator will automatically terminate the trf file name with an incrementing number so that the old data is not overwritten.

– Example: TestRunTest1.trf

- Therefore, it is highly recommended that the runset name not end in a number because it can make interpreting the trf file names difficult.
  - Example: "TestRunTest11.trf" could be the first run for a runset named Test1 or the 11<sup>th</sup> run of a runset named Test.



## **Stop Time**

- Upon entering the runset editor, the run status indicator will be red
   – see the bottom right part of the screen
- Click on the indicator to see the list of errors.
- The first error is usually that the stop time is zero.
- Close the error list and select Edit-Edit Stop Time from the main menu of the TRE.
- Enter 0.001 in order to get one field from the simpleFieldSensor (SFS)
  - the SFS's exposureInterval is 1 ms, so it will record a frame every 1ms.









## **Selection Outputs to Record**

- The runset has no outputs selected for recording upon creation.
- To select the outputs for recording, click on the output recording button or go to Edit-Edit Output Recording from the main menu.
- To select an output for recording, check the box to the left of the output.
- Click OK when all the desired outputs are selected.

😈 Output Recording	>
Please select out	outs to be recorded
Test ⊡-simplefieldsensor 	
simplefieldsensor.fld (When char	nged)
Reg. Epr.: jsimplefieldsensor.fld	
<ul> <li>When changed</li> <li>Exactly</li> </ul>	y every isec i 🔿 Do Not Record
□ Record dt = 0 changes	🗖 At least every 📃 sec
🗆 Compress data (if possible)	🗆 At most every 📃 🛛 sec
Flags	🗖 Begin time 🛛 🗖 sec
🗖 Variable size	🗆 End time 📃 📃 sec
ОК Са	ncel Help



#### **Compile and Link**

- Select the compile and link button from the toolbar or select Build-Make from the main menu.
- A console window will pop-up and should compile your .cpp code to an .obj file and then link the .obj's into an .exe file.

🐼 C:\WINDOWS\system32\cmd.exe - C:\MZA\tempus\v2000.11H\bin\bldrun.bat C:\MZA\tempus\v20
cl.exe /nologo /MD /₩ /Ox /Og /Ob1 /G7 /GX /Zi /MD /I "C:\MZA\Wavetrain\
v2000.11H\include" /I "C:\MZA\Wavetrain\v2000.11H\canlib" /I "C:\MZA\Wavetrain\v
2000.11H\canlib\support\include" /I "C:\MZA\Tempus\v2000.11H\tempus\include\temp
us" /I "C:\MZA\Tempus\v2000.11H\tNL\include" /I "C:\MZA\Tempus\v2000.11H\tempus\
include\recorder" /I "C:\MZA\Tempus\v2000.11H\tempus\include\matrixio" /I "C:\MZ
A\Tempus\v2000.11H\tempus\include\fftengine" /I "C:\MZA\Tempus\v2000.11H\tempus\
include\fftw1-3" /I "C:\MZA\Tempus\v2000.11H\tempus\include\mli" /I "C:\MZA\Temp
us\v2000.11H\tempus\include\portinc"/I "C:\M2A\Tempus\v2000.11H\tempus\include\
pisuite" /1 "C:\M2H\Tempus\02000.TH\trftoois\include" /1 "C:\M2H\Tempus\02000.T
TH\trm\include /I U:\MHILHB6p5pI\extern\include\cpp /I U:\MHILHB6p5pI\extern \include" /I "C \MZ0\Tempue\u2000 11U\tetl\include" /D "WIN22" /D "NDEDUC" /D "
<pre>(Include /I C:\M2H\Tempus\02000.TTH\LStI\Include /D winsz /D WDEB0G /D _ CONSOLE" /D " MPCS" /D "MSUC" /D "MSUTND" /En"C.\Simulations\teet)TeetBunteet ne </pre>
<pre>b" /YY /Fo"C.\Simulations\tost\\" /Ed"C.\Simulations\tost\\" /ED /c "C.\Simulati</pre>
one/test/TestPuntest onn"
cl · Command line warning D4029 · optimization is not available in the standard
edition compiler
TestRuntest.cpp
link.exe @C:\DOCUME~1\jmansel1\LOCALS~1\Temp\nm111.tmp
C:\Simulations\test>if "msvc7" == "msvc8" nmake /f "C:\MZA\tempus\v2000.11H\bin\
trapp8.mak" CFG="trapp - Win32 Release"
Created executable TestRuntest.exe with optimization (Release version).
Make completed successfully
Press any key to continue 💌



## Linking / Compiling Problems

- It is not uncommon that the runsets do not properly compile and link.
- The console window will list the problems with the compilation or linking and give you some indication as to where the problem is and sometimes how to fix it.
- With persistent issues in this regard, call MZA's technical support.



## **Executing the Simulation**

- To execute the simulation press the execute button (red exclamation point) or go to Build-Execute on the main menu.
- The model will compile, link, then execute.
- When execution begins, the tempus runset monitor will pop-up to show the progress.
  - For simple systems the runset monitor may not have time to execute before the system completes, so the monitor will report an error.

👅 tempı	us Runset Monit	or 📘	. 🗆 🗙
-Summary	/		
Runset	<del>_</del> ,	- · · ·	
Name:	Test	Runtest3	
trf File:	Tes	stRuntest3.trf	
Dir.:			
Disk Spa	ce Used:		3.7 MB
Est. Disk	Space Required:	Cor	npleted
Total Rur	ns: 1	Complete:	100.0 %
Elapsed <sup>-</sup>	Time:		2.5 sec.
Estimated	d Total Time:	Cor	npleted
Status:	С	ompleted	
Current R	un		
Run Num	ber: 1	Complete:	100.0 %
Elapsed	Time		2.5 sec.
Estimated	d Total Time:	Cor	npleted
Virtual Sto	op Time:	0.010000	100 sec.
Current Vi	irtual Time:	Cor	npleted



#### **Results - .trf files**

- As soon as the model execution starts, a .trf file is created.
- As the model runs this file is filled up with model data.
- You can click on the TrfView button in the runset editor to view the most recently created .trf file for the runset.
- You can also visualize the .trf file results with Matlab.



#### **Visualizing the Results**



## trf Routines in Matlab

- MZA provides a whole suite of routines in Matlab for reading in data from .trf files, including:
  - trfopen
  - trfload
  - tmxparams
- The most commonly used tool is a GUI package called "trfview".
  - Run it by typing "trfview" at the command line and pressing enter.



#### **Matlab Path**

- If none of the trf tools work in Matlab, make sure your Matlab path has the directories listed below in it.
  - Go to File-Set Path for the following dialog box:

📣 Set Path	
All changes take effect imm	ediately.
	MATLAB search path:
Add Folder	🖻 C:\MZA\tempus\v2000.11H\tempus\mfiles\tmx.tbx 💦 🔼
Add with Subfolders Remove	C:\MZA\tempus\v2000.11H\trftools\mfiles\trfview.tbx
Save Close F	Revert Default Help





#### trfview

- Open your trf file using File-Open from the main menu.
- A list of variables and parameters should appear.

<b>//</b> 1	RF View	er			<u> </u>
Eile	<u>V</u> ariable	Parameter			3
File C:V	<b>ename</b> Simulatior	ns\test\TestRuntest4.trf			
Ru	Ins: 1	Select 1	🗹 View All Runs		
		Variable Name	Data Type	Count	
tes	t.simplefi	eldsensor.fld	complex	65536	

Parameter Name	Data Type	Value
dxy	float	8.3e-005
nxy	int	256
stopTime	double	0.001
wavelength	float	1e-006



## **Viewing Output Data**

- To view output data, double click on the named variable and a variable display box should appear.
- To plot that data, click on the plot button in the bottom right of the screen.

📣 Variable Disp	lay			X							
Variable Name	test.simplefieldsensr	or.fld									
Description	Integrated field										
Туре	complex	Byte Count	1048576								
Overall Statist Average 5.099e Std. Dev. 4.696e Max -9.666e Min 0	ics Calc e-007 e-005 e-010 at 3.29e+0 ) at 1.651e+	1 1									
Data Blocks			Pletting	]							
Item Count Block Count Save .mat	65536 X Dim 1 Y Dim	. 256 ×	Plot	Block 1							



#### False-Color Plot

 This window should appear showing the field magnitude.





## **Estimation of Data Correctness**

- If we zoom in on this plot, we can approximately count the number of pixels across the Gaussian beam is at the focus.
- In this example, there are about 10 pixels and the mesh spacing is 83 µm, so the beam diameter is about 830 µm.
- This corresponds reasonably well to the expected 640-µm diameter calculated from theory, but it is not very scientific.





#### **Extracting the Data to Matlab**

- The data can be extracted to Matlab by right clicking on the variable name in trfview and then selecting "Copy to Workspace".
- The data is then available as the variable name (with the periods).

📣 TRF Viewer				_ 🗆 🗙
File Variable Para	ameter			ъ Ч
Filename C:\Simulations\tes	t\TestRuntest4.trf			
Runs: 1	Select 1	View All Runs		
Va	riable Name	Data Type	Count	
test.simplefieldse	nso Show Selected Va	riable X	65536	
	Copy to Workspa	ice		
Doro	motor Nomo	Data Turca	Valua	
dxy		float	8.3e-005	
nxý		int	256	
stopTime		double	0.001	
wavelength		float	1e-006	

#### **Data Interpretation**

- The data in the workspace is in vector form with each time sample as a new column.
- A single time-step can be converted into a 2D field with the following command:
  - E=reshape(test.simplefieldsensor.fld(:,1),256, 256);
- The field can be converted to an intensity profile with:

-I = E .\* conj(E);

#### Data Analysis

- Matlab can be used to calculate the second moments of the intensity profile as follows:
  - n=256; dx=83e-6; x = (-n/2:1:n/2-1)\*dx;
  - [xx,yy]=meshgrid(x,x);
  - $Ix2 = sum(sum(xx.^2 .* I));$
  - $Iy2 = sum(sum(yy.^2 .* I));$
  - Isum = sum(sum(I));
  - -x2 = sqrt(Ixx/Isum); y2 = sqrt(Ixx/Isum);
- 4 times the Intensity second moment is the beam diameter.
  - For our data the beam diameter is 637 µm, which corresponds exactly with our initial theoretical prediction.

#### Conclusions

- In this example, we showed how to start modeling with WaveTrain.
- We used the example of propagating a Gaussian beam to a focus.
- Our WaveTrain results corresponded exactly with our theoretical prediction.

