

XNLO - XNLO

1.1.0

Generated by Doxygen 1.8.11

Contents

1	Namespace Index	1
1.1	Namespace List	1
2	Class Index	3
2.1	Class List	3
3	Namespace Documentation	5
3.1	XNLO Namespace Reference	5
3.1.1	Detailed Description	5
3.1.2	Function Documentation	5
3.1.2.1	XNLO(ArrayXXcd A_w_active, ArrayXd w_active)	5
4	Class Documentation	7
4.1	XNLO::Config_Settings Class Reference	7
4.1.1	Detailed Description	10
4.1.2	Member Enumeration Documentation	11
4.1.2.1	SN	11
4.1.3	Constructor & Destructor Documentation	11
4.1.3.1	Config_Settings()	11
4.1.4	Member Function Documentation	11
4.1.4.1	alpha()	11
4.1.4.2	alpha_description()	11
4.1.4.3	alpha_description_set(std::string)	11
4.1.4.4	alpha_set(double)	12
4.1.4.5	atoms_per_worker()	12

4.1.4.6	atoms_per_worker_description()	12
4.1.4.7	atoms_per_worker_description_set(std::string)	12
4.1.4.8	atoms_per_worker_set(int)	12
4.1.4.9	CEO()	12
4.1.4.10	CEO_description()	12
4.1.4.11	CEO_description_set(std::string)	12
4.1.4.12	CEO_set(double)	12
4.1.4.13	check_paths(bool print_to_screen=true)	12
4.1.4.14	FWHM()	12
4.1.4.15	FWHM_description()	12
4.1.4.16	FWHM_description_set(std::string)	12
4.1.4.17	FWHM_set(double)	12
4.1.4.18	I_0()	12
4.1.4.19	I_0_description()	12
4.1.4.20	I_0_description_set(std::string)	12
4.1.4.21	I_0_set(double)	12
4.1.4.22	N_t()	12
4.1.4.23	N_t_description()	12
4.1.4.24	N_t_description_set(std::string)	12
4.1.4.25	N_t_set(int)	12
4.1.4.26	output_wavefunction()	12
4.1.4.27	output_wavefunction_description()	13
4.1.4.28	output_wavefunction_description_set(std::string)	13
4.1.4.29	output_wavefunction_set(int)	13
4.1.4.30	P_av()	13
4.1.4.31	P_av_description()	13
4.1.4.32	P_av_description_set(std::string)	13
4.1.4.33	P_av_set(double)	13
4.1.4.34	path_config_file()	13
4.1.4.35	path_config_file_description()	13

4.1.4.36	path_config_file_description_set(std::string)	13
4.1.4.37	path_config_file_set(std::string)	13
4.1.4.38	path_config_log()	13
4.1.4.39	path_config_log_description()	13
4.1.4.40	path_config_log_description_set(std::string)	13
4.1.4.41	path_config_log_set(std::string)	13
4.1.4.42	path_dipole()	13
4.1.4.43	path_dipole_description()	13
4.1.4.44	path_dipole_description_set(std::string)	13
4.1.4.45	path_dipole_set(std::string)	13
4.1.4.46	path_E()	13
4.1.4.47	path_E_description()	13
4.1.4.48	path_E_description_set(std::string)	13
4.1.4.49	path_E_set(std::string)	13
4.1.4.50	path_input_j0()	14
4.1.4.51	path_input_j0_description()	14
4.1.4.52	path_input_j0_description_set(std::string)	14
4.1.4.53	path_input_j0_set(std::string)	14
4.1.4.54	path_laser_A_w_I()	14
4.1.4.55	path_laser_A_w_I_description()	14
4.1.4.56	path_laser_A_w_I_description_set(std::string)	14
4.1.4.57	path_laser_A_w_I_set(std::string)	14
4.1.4.58	path_laser_A_w_R()	14
4.1.4.59	path_laser_A_w_R_description()	14
4.1.4.60	path_laser_A_w_R_description_set(std::string)	14
4.1.4.61	path_laser_A_w_R_set(std::string)	14
4.1.4.62	path_laser_w_active()	14
4.1.4.63	path_laser_w_active_description()	14
4.1.4.64	path_laser_w_active_description_set(std::string)	14
4.1.4.65	path_laser_w_active_set(std::string)	14

4.1.4.66	<code>path_w()</code>	14
4.1.4.67	<code>path_w_description()</code>	14
4.1.4.68	<code>path_w_description_set(std::string)</code>	14
4.1.4.69	<code>path_w_set(std::string)</code>	14
4.1.4.70	<code>pend_path()</code>	14
4.1.4.71	<code>pend_path_description()</code>	14
4.1.4.72	<code>pend_path_description_set(std::string)</code>	14
4.1.4.73	<code>pend_path_set(std::string)</code>	15
4.1.4.74	<code>print()</code>	15
4.1.4.75	<code>print(std::string)</code>	15
4.1.4.76	<code>read_in(std::string, bool print_to_screen=true)</code>	15
4.1.4.77	<code>read_in_laser_pulse()</code>	15
4.1.4.78	<code>read_in_laser_pulse_description()</code>	15
4.1.4.79	<code>read_in_laser_pulse_description_set(std::string)</code>	15
4.1.4.80	<code>read_in_laser_pulse_set(int)</code>	15
4.1.4.81	<code>RR()</code>	15
4.1.4.82	<code>RR_description()</code>	15
4.1.4.83	<code>RR_description_set(std::string)</code>	15
4.1.4.84	<code>RR_set(double)</code>	15
4.1.4.85	<code>set_path(std::string, std::string)</code>	15
4.1.4.86	<code>set_post_path(std::string, std::string)</code>	15
4.1.4.87	<code>set_pre_path(std::string, std::string)</code>	15
4.1.4.88	<code>set_variable(std::string &, std::string &, std::string &)</code>	15
4.1.4.89	<code>spot_radius()</code>	15
4.1.4.90	<code>spot_radius_description()</code>	15
4.1.4.91	<code>spot_radius_description_set(std::string)</code>	15
4.1.4.92	<code>spot_radius_set(double)</code>	15
4.1.4.93	<code>t_max()</code>	15
4.1.4.94	<code>t_max_description()</code>	15
4.1.4.95	<code>t_max_description_set(std::string)</code>	15

4.1.4.96	t_max_set(double)	16
4.1.4.97	t_min()	16
4.1.4.98	t_min_description()	16
4.1.4.99	t_min_description_set(std::string)	16
4.1.4.100	t_min_set(double)	16
4.1.4.101	x_max()	16
4.1.4.102	x_max_description()	16
4.1.4.103	x_max_description_set(std::string)	16
4.1.4.104	x_max_set(double)	16
4.1.4.105	x_min()	16
4.1.4.106	x_min_description()	16
4.1.4.107	x_min_description_set(std::string)	16
4.1.4.108	x_min_set(double)	16
4.1.5	Member Data Documentation	16
4.1.5.1	alpha_	16
4.1.5.2	alpha_description_	16
4.1.5.3	atoms_per_worker_	16
4.1.5.4	atoms_per_worker_description_	16
4.1.5.5	CEO_	16
4.1.5.6	CEO_description_	16
4.1.5.7	FWHM_	16
4.1.5.8	FWHM_description_	16
4.1.5.9	I_0_	17
4.1.5.10	I_0_description_	17
4.1.5.11	N_t_	17
4.1.5.12	N_t_description_	17
4.1.5.13	output_wavefunction_	17
4.1.5.14	output_wavefunction_description_	17
4.1.5.15	P_av_	17
4.1.5.16	P_av_description_	17

4.1.5.17	path_config_file_	17
4.1.5.18	path_config_file_description_	17
4.1.5.19	path_config_log_	17
4.1.5.20	path_config_log_description_	17
4.1.5.21	path_dipole_	17
4.1.5.22	path_dipole_description_	17
4.1.5.23	path_E_	17
4.1.5.24	path_E_description_	17
4.1.5.25	path_input_j0_	17
4.1.5.26	path_input_j0_description_	17
4.1.5.27	path_laser_A_w_I_	17
4.1.5.28	path_laser_A_w_I_description_	17
4.1.5.29	path_laser_A_w_R_	18
4.1.5.30	path_laser_A_w_R_description_	18
4.1.5.31	path_laser_w_active_	18
4.1.5.32	path_laser_w_active_description_	18
4.1.5.33	path_w_	18
4.1.5.34	path_w_description_	18
4.1.5.35	pend_path_	18
4.1.5.36	pend_path_description_	18
4.1.5.37	read_in_laser_pulse_	18
4.1.5.38	read_in_laser_pulse_description_	18
4.1.5.39	RR_	18
4.1.5.40	RR_description_	18
4.1.5.41	setting_name	18
4.1.5.42	spot_radius_	19
4.1.5.43	spot_radius_description_	19
4.1.5.44	t_max_	19
4.1.5.45	t_max_description_	19
4.1.5.46	t_min_	19

4.1.5.47	t_min_description_	19
4.1.5.48	x_max_	19
4.1.5.49	x_max_description_	19
4.1.5.50	x_min_	19
4.1.5.51	x_min_description_	19
4.2	XNLO::DHT Class Reference	19
4.2.1	Detailed Description	19
4.2.2	Constructor & Destructor Documentation	20
4.2.2.1	DHT()	20
4.2.2.2	DHT(int n_r_, maths_textbook &maths_)	20
4.2.3	Member Function Documentation	20
4.2.3.1	backward(Eigen::ArrayXcd f_kr_)	20
4.2.3.2	forward(Eigen::ArrayXcd f_r_)	20
4.2.4	Member Data Documentation	20
4.2.4.1	H	20
4.3	XNLO::grid_rkr Class Reference	20
4.3.1	Detailed Description	21
4.3.2	Constructor & Destructor Documentation	21
4.3.2.1	grid_rkr(int n_r_, double R_, maths_textbook &maths_)	21
4.3.3	Member Data Documentation	21
4.3.3.1	kr	21
4.3.3.2	n_r	21
4.3.3.3	r	21
4.3.3.4	R	21
4.4	XNLO::grid_tw Class Reference	21
4.4.1	Detailed Description	22
4.4.2	Constructor & Destructor Documentation	22
4.4.2.1	grid_tw(int N_t_, double t_min_, double t_max_)	22
4.4.3	Member Data Documentation	22
4.4.3.1	dt	22

4.4.3.2	N_t	22
4.4.3.3	t	22
4.4.3.4	t_max	22
4.4.3.5	t_min	22
4.4.3.6	w	22
4.5	XNLO::grid_xkx Class Reference	22
4.5.1	Detailed Description	23
4.5.2	Constructor & Destructor Documentation	23
4.5.2.1	grid_xkx()	23
4.5.2.2	grid_xkx(int N_x_, double x_min_, double x_max_)	23
4.5.3	Member Data Documentation	23
4.5.3.1	dx	23
4.5.3.2	kx	23
4.5.3.3	N_x	23
4.5.3.4	x	23
4.5.3.5	x_max	23
4.5.3.6	x_min	23
4.6	XNLO::IO Class Reference	23
4.6.1	Detailed Description	24
4.6.2	Constructor & Destructor Documentation	24
4.6.2.1	IO()	24
4.6.3	Member Function Documentation	24
4.6.3.1	overwrite(const std::string path)	24
4.6.3.2	read_double(const std::string path, int N_row_, int N_col_, bool print=true) . . .	24
4.6.3.3	read_double(const std::string path, bool skip_header=true, bool print=true) . . .	24
4.6.3.4	read_header(const std::string path, bool print=true)	24
4.6.3.5	write_ascii_double(std::string path, ArrayXXd data)	25
4.6.3.6	write_double(const std::string path, ArrayXXd output, int N_row_, int N_col_) . .	25
4.6.3.7	write_header(const std::string path_, int N_row_, int N_col_)	25
4.6.4	Member Data Documentation	25

4.6.4.1	binary_format	25
4.6.4.2	binary_format_len	26
4.6.4.3	binary_format_subversion	26
4.6.4.4	binary_format_version	26
4.6.4.5	data_size	26
4.6.4.6	double_size	26
4.6.4.7	header_size	26
4.6.4.8	N_col_	26
4.6.4.9	N_row_	26
4.7	XNLO::laser_pulse Class Reference	26
4.7.1	Detailed Description	27
4.7.2	Constructor & Destructor Documentation	27
4.7.2.1	laser_pulse(double P_av_, double RR_, double FWHM_, double l_0_, double C_↵ EO_, double spot_radius_, double ROC_, grid_rkr rkr_, grid_tw tw_, std::string path_A_w_R, std::string path_A_w_l, std::string path_w_active, int read_in_↵ laser_pulse)	27
4.7.2.2	laser_pulse(grid_rkr rkr_, grid_tw tw_, ArrayXXcd A_w_active, ArrayXd w_active)	27
4.7.3	Member Data Documentation	27
4.7.3.1	E	27
4.8	XNLO::maths_textbook Class Reference	27
4.8.1	Detailed Description	28
4.8.2	Constructor & Destructor Documentation	28
4.8.2.1	maths_textbook()	28
4.8.2.2	maths_textbook(std::string path_input_j0_)	28
4.8.3	Member Function Documentation	28
4.8.3.1	interp1D(ArrayXd input_array, int input_length, int output_length, int spline_order)	28
4.8.3.2	trapz(ArrayXd x_, ArrayXd y_)	28
4.8.4	Member Data Documentation	29
4.8.4.1	J0_zeros	29
4.8.4.2	path_input_j0	29
4.8.4.3	pi	29
4.9	XNLO::physics_textbook Class Reference	29

4.9.1	Detailed Description	29
4.9.2	Constructor & Destructor Documentation	29
4.9.2.1	physics_textbook()	29
4.9.3	Member Data Documentation	30
4.9.3.1	c	30
4.9.3.2	E_at	30
4.9.3.3	eps_0	30
4.9.3.4	h_bar	30
4.9.3.5	k_B	30
4.9.3.6	l_at	30
4.9.3.7	m_at	30
4.9.3.8	mu_0	30
4.9.3.9	q_at	30
4.9.3.10	t_at	30
4.9.3.11	w_at	30
4.10	XNLO::Result Struct Reference	30
4.10.1	Member Data Documentation	31
4.10.1.1	acceleration	31
4.10.1.2	E	31
4.10.1.3	w	31
4.10.1.4	wavefunction	31
4.11	XNLO::Schrodinger_atom_1D Class Reference	31
4.11.1	Detailed Description	32
4.11.2	Constructor & Destructor Documentation	32
4.11.2.1	Schrodinger_atom_1D(grid_tw &tw_, double alpha_, int output_wavefunction_)	32
4.11.3	Member Function Documentation	32
4.11.3.1	get_acceleration(int N_it_, double dt_, ArrayXd E_)	32
4.11.3.2	set_GS(int N_it_)	32
4.11.3.3	solve_TDSE_PS(int N_it_, std::complex< double > dt_, ArrayXd E_, int e_)	32
4.11.4	Member Data Documentation	33
4.11.4.1	alpha	33
4.11.4.2	energy	33
4.11.4.3	output_wavefunction	33
4.11.4.4	tw	33
4.11.4.5	V_model	33
4.11.4.6	wfn	33
4.11.4.7	wfn_GS	33
4.11.4.8	wfn_output	33
4.11.4.9	xkx	33

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

XNLO	5
-----------------------	---

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

XNLO::Config_Settings	7
XNLO::DHT	19
XNLO::grid_rkr	20
XNLO::grid_tw	21
XNLO::grid_xkx	22
XNLO::IO	23
XNLO::laser_pulse	26
XNLO::maths_textbook	27
XNLO::physics_textbook	29
XNLO::Result	30
XNLO::Schrodinger_atom_1D	31

Chapter 3

Namespace Documentation

3.1 XNLO Namespace Reference

Classes

- class **Config_Settings**
- class **DHT**
- class **grid_rkr**
- class **grid_tw**
- class **grid_xkx**
- class **IO**
- class **laser_pulse**
- class **maths_textbook**
- class **physics_textbook**
- struct **Result**
- class **Schrodinger_atom_1D**

Functions

- **Result XNLO** (ArrayXXcd A_w_active, ArrayXd w_active)

3.1.1 Detailed Description

XNLO (p. 5) namespace - A container for everything **XNLO** (p. 5), so that all classes etc that are a part of it are self contained and so that it is harder to confuse with other namespaces etc.

3.1.2 Function Documentation

3.1.2.1 Result XNLO::XNLO (ArrayXXcd A_w_active, ArrayXd w_active)

Chapter 4

Class Documentation

4.1 XNLO::Config_Settings Class Reference

```
#include <config_settings.hpp>
```

Public Member Functions

- **Config_Settings** ()
- void **read_in** (std::string, bool print_to_screen=true)
- void **check_paths** (bool print_to_screen=true)
- void **print** ()
- void **print** (std::string)
- int **atoms_per_worker** ()
- void **atoms_per_worker_set** (int)
- std::string **atoms_per_worker_description** ()
- void **atoms_per_worker_description_set** (std::string)
- double **x_min** ()
- void **x_min_set** (double)
- std::string **x_min_description** ()
- void **x_min_description_set** (std::string)
- double **x_max** ()
- void **x_max_set** (double)
- std::string **x_max_description** ()
- void **x_max_description_set** (std::string)
- int **N_t** ()
- void **N_t_set** (int)
- std::string **N_t_description** ()
- void **N_t_description_set** (std::string)
- double **t_min** ()
- void **t_min_set** (double)
- std::string **t_min_description** ()
- void **t_min_description_set** (std::string)
- double **t_max** ()
- void **t_max_set** (double)
- std::string **t_max_description** ()
- void **t_max_description_set** (std::string)
- double **P_av** ()

- void **P_av_set** (double)
- std::string **P_av_description** ()
- void **P_av_description_set** (std::string)
- double **RR** ()
- void **RR_set** (double)
- std::string **RR_description** ()
- void **RR_description_set** (std::string)
- double **FWHM** ()
- void **FWHM_set** (double)
- std::string **FWHM_description** ()
- void **FWHM_description_set** (std::string)
- double **I_0** ()
- void **I_0_set** (double)
- std::string **I_0_description** ()
- void **I_0_description_set** (std::string)
- double **CEO** ()
- void **CEO_set** (double)
- std::string **CEO_description** ()
- void **CEO_description_set** (std::string)
- double **spot_radius** ()
- void **spot_radius_set** (double)
- std::string **spot_radius_description** ()
- void **spot_radius_description_set** (std::string)
- double **alpha** ()
- void **alpha_set** (double)
- std::string **alpha_description** ()
- void **alpha_description_set** (std::string)
- int **output_wavefunction** ()
- void **output_wavefunction_set** (int)
- std::string **output_wavefunction_description** ()
- void **output_wavefunction_description_set** (std::string)
- int **read_in_laser_pulse** ()
- void **read_in_laser_pulse_set** (int)
- std::string **read_in_laser_pulse_description** ()
- void **read_in_laser_pulse_description_set** (std::string)
- std::string **pend_path** ()
- void **pend_path_set** (std::string)
- std::string **pend_path_description** ()
- void **pend_path_description_set** (std::string)
- std::string **path_input_j0** ()
- void **path_input_j0_set** (std::string)
- std::string **path_input_j0_description** ()
- void **path_input_j0_description_set** (std::string)
- std::string **path_laser_A_w_R** ()
- void **path_laser_A_w_R_set** (std::string)
- std::string **path_laser_A_w_R_description** ()
- void **path_laser_A_w_R_description_set** (std::string)
- std::string **path_laser_A_w_I** ()
- void **path_laser_A_w_I_set** (std::string)
- std::string **path_laser_A_w_I_description** ()
- void **path_laser_A_w_I_description_set** (std::string)
- std::string **path_laser_w_active** ()
- void **path_laser_w_active_set** (std::string)
- std::string **path_laser_w_active_description** ()
- void **path_laser_w_active_description_set** (std::string)

- `std::string path_dipole ()`
- `void path_dipole_set (std::string)`
- `std::string path_dipole_description ()`
- `void path_dipole_description_set (std::string)`
- `std::string path_w ()`
- `void path_w_set (std::string)`
- `std::string path_w_description ()`
- `void path_w_description_set (std::string)`
- `std::string path_E ()`
- `void path_E_set (std::string)`
- `std::string path_E_description ()`
- `void path_E_description_set (std::string)`
- `std::string path_config_file ()`
- `void path_config_file_set (std::string)`
- `std::string path_config_file_description ()`
- `void path_config_file_description_set (std::string)`
- `std::string path_config_log ()`
- `void path_config_log_set (std::string)`
- `std::string path_config_log_description ()`
- `void path_config_log_description_set (std::string)`

Private Types

- `enum SN {`
`SN::atoms_per_worker = 0, SN::x_min, SN::x_max, SN::N_t,`
`SN::t_min, SN::t_max, SN::P_av, SN::RR,`
`SN::FWHM, SN::I_0, SN::CEO, SN::spot_radius,`
`SN::alpha, SN::read_in_laser_pulse, SN::output_wavefunction, SN::pend_path,`
`SN::path_input_j0, SN::path_laser_A_w_R, SN::path_laser_A_w_I, SN::path_laser_w_active,`
`SN::path_dipole, SN::path_w, SN::path_E, SN::path_config_file,`
`SN::path_config_log, SN::LAST_SN_ENTRY }`

Private Member Functions

- `void set_variable (std::string &, std::string &, std::string &)`
- `std::string set_path (std::string, std::string)`
- `std::string set_pre_path (std::string, std::string)`
- `std::string set_post_path (std::string, std::string)`

Private Attributes

- `int atoms_per_worker_ = 2`
- `double x_min_ = 0`
- `double x_max_ = 100e-6`
- `int N_t_ = 262144`
- `double t_min_ = -100e-15`
- `double t_max_ = 100e-15`
- `double P_av_ = 0.18`
- `double RR_ = 1000`
- `double FWHM_ = 15e-15`
- `double I_0_ = 795e-9`
- `double CEO_ = 0`

- double **spot_radius_** = 42e-6
- double **alpha_** = 1.45
- int **output_wavefunction_** = 0
- int **read_in_laser_pulse_** = 0
- std::string **pend_path_** = "prepend"
- std::string **path_input_j0_** = "../input/J0_zeros.bin"
- std::string **path_laser_A_w_R_** = "../UPPE/output/000_1_A_w_R.bin"
- std::string **path_laser_A_w_I_** = "../UPPE/output/000_1_A_w_I.bin"
- std::string **path_laser_w_active_** = "../UPPE/output/000_1_w_active.bin"
- std::string **path_dipole_** = "../output/dipole.bin"
- std::string **path_w_** = "../output/w.bin"
- std::string **path_E_** = "../output/E.bin"
- std::string **path_config_file_** = "./config.txt"
- std::string **path_config_log_** = "../output/config_log.txt"
- std::string **atoms_per_worker_description_** = "(default) (int) The number of atoms per worker"
- std::string **x_min_description_** = "(default) (double) The **x_min** value"
- std::string **x_max_description_** = "(default) (double) the **x_max** value"
- std::string **N_t_description_** = "(default) (int) The **N_t** value"
- std::string **t_min_description_** = "(default) (double) The **t_min** value"
- std::string **t_max_description_** = "(default) (double) The **t_max** value"
- std::string **P_av_description_** = "(default) (double) The **P_av** value"
- std::string **RR_description_** = "(default) (double) The **RR** value"
- std::string **FWHM_description_** = "(default) (double) The **FWHM** value"
- std::string **I_0_description_** = "(default) (double) The **I_0** value"
- std::string **CEO_description_** = "(default) (double) The **CEO** value"
- std::string **spot_radius_description_** = "(default) (double) The **spot_radius** value"
- std::string **alpha_description_** = "(default) (double) The **alpha** value"
- std::string **read_in_laser_pulse_description_** = "(default) (int) Switch to read in laser pulse"
- std::string **output_wavefunction_description_** = "(default) (int) Switch to output wavefunction"
- std::string **pend_path_description_** = "(default) (std::string) Pending switch"
- std::string **path_input_j0_description_** = "(default) (std::string) Path to J0_zeros.bin"
- std::string **path_laser_A_w_R_description_** = "(default) (std::string) Path to A_w_R of laser pulse input"
- std::string **path_laser_A_w_I_description_** = "(default) (std::string) Path to A_w_I of laser pulse input"
- std::string **path_laser_w_active_description_** = "(default) (std::string) Path to w_active of laser pulse input"
- std::string **path_dipole_description_** = "(default) (std::string) Output path of acceleration"
- std::string **path_w_description_** = "(default) (std::string) Output path of w"
- std::string **path_E_description_** = "(default) (std::string) Ouput path of electric field"
- std::string **path_config_file_description_** = "(default) (std::string) config.txt path"
- std::string **path_config_log_description_** = "(default) (std::string) Output path of config_log.txt"

Static Private Attributes

- static const char * **setting_name** []

4.1.1 Detailed Description

Modified by Samuel Senior on 05/02/2017. Reads in input parameters and settings from a config file.

4.1.2 Member Enumeration Documentation

4.1.2.1 enum XNLO::Config_Settings::SN [strong],[private]

Enumerator

atoms_per_worker
x_min
x_max
N_t
t_min
t_max
P_av
RR
FWHM
I_0
CEO
spot_radius
alpha
read_in_laser_pulse
output_wavefunction
pend_path
path_input_j0
path_laser_A_w_R
path_laser_A_w_I
path_laser_w_active
path_dipole
path_w
path_E
path_config_file
path_config_log
LAST_SN_ENTRY

4.1.3 Constructor & Destructor Documentation

4.1.3.1 XNLO::Config_Settings::Config_Settings ()

4.1.4 Member Function Documentation

4.1.4.1 double XNLO::Config_Settings::alpha ()

4.1.4.2 std::string XNLO::Config_Settings::alpha_description ()

4.1.4.3 void XNLO::Config_Settings::alpha_description_set (std::string *description*)

- 4.1.4.4 void XNLO::Config_Settings::alpha_set (double *val*)
- 4.1.4.5 int XNLO::Config_Settings::atoms_per_worker ()
- 4.1.4.6 std::string XNLO::Config_Settings::atoms_per_worker_description ()
- 4.1.4.7 void XNLO::Config_Settings::atoms_per_worker_description_set (std::string *description*)
- 4.1.4.8 void XNLO::Config_Settings::atoms_per_worker_set (int *val*)
- 4.1.4.9 double XNLO::Config_Settings::CEO ()
- 4.1.4.10 std::string XNLO::Config_Settings::CEO_description ()
- 4.1.4.11 void XNLO::Config_Settings::CEO_description_set (std::string *description*)
- 4.1.4.12 void XNLO::Config_Settings::CEO_set (double *val*)
- 4.1.4.13 void XNLO::Config_Settings::check_paths (bool *print_to_screen* = `true`)
- 4.1.4.14 double XNLO::Config_Settings::FWHM ()
- 4.1.4.15 std::string XNLO::Config_Settings::FWHM_description ()
- 4.1.4.16 void XNLO::Config_Settings::FWHM_description_set (std::string *description*)
- 4.1.4.17 void XNLO::Config_Settings::FWHM_set (double *val*)
- 4.1.4.18 double XNLO::Config_Settings::l_0 ()
- 4.1.4.19 std::string XNLO::Config_Settings::l_0_description ()
- 4.1.4.20 void XNLO::Config_Settings::l_0_description_set (std::string *description*)
- 4.1.4.21 void XNLO::Config_Settings::l_0_set (double *val*)
- 4.1.4.22 int XNLO::Config_Settings::N_t ()
- 4.1.4.23 std::string XNLO::Config_Settings::N_t_description ()
- 4.1.4.24 void XNLO::Config_Settings::N_t_description_set (std::string *description*)
- 4.1.4.25 void XNLO::Config_Settings::N_t_set (int *val*)
- 4.1.4.26 int XNLO::Config_Settings::output_wavefunction ()

- 4.1.4.27 `std::string XNLO::Config_Settings::output_wavefunction_description ()`
- 4.1.4.28 `void XNLO::Config_Settings::output_wavefunction_description_set (std::string description)`
- 4.1.4.29 `void XNLO::Config_Settings::output_wavefunction_set (int val)`
- 4.1.4.30 `double XNLO::Config_Settings::P_av ()`
- 4.1.4.31 `std::string XNLO::Config_Settings::P_av_description ()`
- 4.1.4.32 `void XNLO::Config_Settings::P_av_description_set (std::string description)`
- 4.1.4.33 `void XNLO::Config_Settings::P_av_set (double val)`
- 4.1.4.34 `std::string XNLO::Config_Settings::path_config_file ()`
- 4.1.4.35 `std::string XNLO::Config_Settings::path_config_file_description ()`
- 4.1.4.36 `void XNLO::Config_Settings::path_config_file_description_set (std::string description)`
- 4.1.4.37 `void XNLO::Config_Settings::path_config_file_set (std::string val)`
- 4.1.4.38 `std::string XNLO::Config_Settings::path_config_log ()`
- 4.1.4.39 `std::string XNLO::Config_Settings::path_config_log_description ()`
- 4.1.4.40 `void XNLO::Config_Settings::path_config_log_description_set (std::string description)`
- 4.1.4.41 `void XNLO::Config_Settings::path_config_log_set (std::string val)`
- 4.1.4.42 `std::string XNLO::Config_Settings::path_dipole ()`
- 4.1.4.43 `std::string XNLO::Config_Settings::path_dipole_description ()`
- 4.1.4.44 `void XNLO::Config_Settings::path_dipole_description_set (std::string description)`
- 4.1.4.45 `void XNLO::Config_Settings::path_dipole_set (std::string val)`
- 4.1.4.46 `std::string XNLO::Config_Settings::path_E ()`
- 4.1.4.47 `std::string XNLO::Config_Settings::path_E_description ()`
- 4.1.4.48 `void XNLO::Config_Settings::path_E_description_set (std::string description)`
- 4.1.4.49 `void XNLO::Config_Settings::path_E_set (std::string val)`

- 4.1.4.50 `std::string XNLO::Config_Settings::path_input_j0 ()`
- 4.1.4.51 `std::string XNLO::Config_Settings::path_input_j0_description ()`
- 4.1.4.52 `void XNLO::Config_Settings::path_input_j0_description_set (std::string description)`
- 4.1.4.53 `void XNLO::Config_Settings::path_input_j0_set (std::string value)`
- 4.1.4.54 `std::string XNLO::Config_Settings::path_laser_A_w_I ()`
- 4.1.4.55 `std::string XNLO::Config_Settings::path_laser_A_w_I_description ()`
- 4.1.4.56 `void XNLO::Config_Settings::path_laser_A_w_I_description_set (std::string description)`
- 4.1.4.57 `void XNLO::Config_Settings::path_laser_A_w_I_set (std::string value)`
- 4.1.4.58 `std::string XNLO::Config_Settings::path_laser_A_w_R ()`
- 4.1.4.59 `std::string XNLO::Config_Settings::path_laser_A_w_R_description ()`
- 4.1.4.60 `void XNLO::Config_Settings::path_laser_A_w_R_description_set (std::string description)`
- 4.1.4.61 `void XNLO::Config_Settings::path_laser_A_w_R_set (std::string value)`
- 4.1.4.62 `std::string XNLO::Config_Settings::path_laser_w_active ()`
- 4.1.4.63 `std::string XNLO::Config_Settings::path_laser_w_active_description ()`
- 4.1.4.64 `void XNLO::Config_Settings::path_laser_w_active_description_set (std::string description)`
- 4.1.4.65 `void XNLO::Config_Settings::path_laser_w_active_set (std::string value)`
- 4.1.4.66 `std::string XNLO::Config_Settings::path_w ()`
- 4.1.4.67 `std::string XNLO::Config_Settings::path_w_description ()`
- 4.1.4.68 `void XNLO::Config_Settings::path_w_description_set (std::string description)`
- 4.1.4.69 `void XNLO::Config_Settings::path_w_set (std::string val)`
- 4.1.4.70 `std::string XNLO::Config_Settings::pend_path ()`
- 4.1.4.71 `std::string XNLO::Config_Settings::pend_path_description ()`
- 4.1.4.72 `void XNLO::Config_Settings::pend_path_description_set (std::string description_)`

- 4.1.4.73 void XNLO::Config_Settings::pend_path_set (std::string *pend_path_val*)
- 4.1.4.74 void XNLO::Config_Settings::print ()
- 4.1.4.75 void XNLO::Config_Settings::print (std::string *path_*)
- 4.1.4.76 void XNLO::Config_Settings::read_in (std::string *path*, bool *print_to_screen* = true)
- 4.1.4.77 int XNLO::Config_Settings::read_in_laser_pulse ()
- 4.1.4.78 std::string XNLO::Config_Settings::read_in_laser_pulse_description ()
- 4.1.4.79 void XNLO::Config_Settings::read_in_laser_pulse_description_set (std::string *description*)
- 4.1.4.80 void XNLO::Config_Settings::read_in_laser_pulse_set (int *val*)
- 4.1.4.81 double XNLO::Config_Settings::RR ()
- 4.1.4.82 std::string XNLO::Config_Settings::RR_description ()
- 4.1.4.83 void XNLO::Config_Settings::RR_description_set (std::string *description*)
- 4.1.4.84 void XNLO::Config_Settings::RR_set (double *val*)
- 4.1.4.85 std::string XNLO::Config_Settings::set_path (std::string *path*, std::string *pending_string*) [private]
- 4.1.4.86 std::string XNLO::Config_Settings::set_post_path (std::string *path*, std::string *post_path*) [private]
- 4.1.4.87 std::string XNLO::Config_Settings::set_pre_path (std::string *pre_path*, std::string *path*) [private]
- 4.1.4.88 void XNLO::Config_Settings::set_variable (std::string & *variable_name*, std::string & *variable_value_str*, std::string & *input_description_char*) [private]
- 4.1.4.89 double XNLO::Config_Settings::spot_radius ()
- 4.1.4.90 std::string XNLO::Config_Settings::spot_radius_description ()
- 4.1.4.91 void XNLO::Config_Settings::spot_radius_description_set (std::string *description*)
- 4.1.4.92 void XNLO::Config_Settings::spot_radius_set (double *val*)
- 4.1.4.93 double XNLO::Config_Settings::t_max ()
- 4.1.4.94 std::string XNLO::Config_Settings::t_max_description ()
- 4.1.4.95 void XNLO::Config_Settings::t_max_description_set (std::string *description*)

- 4.1.4.96 void XNLO::Config_Settings::t_max_set (double *val*)
- 4.1.4.97 double XNLO::Config_Settings::t_min ()
- 4.1.4.98 std::string XNLO::Config_Settings::t_min_description ()
- 4.1.4.99 void XNLO::Config_Settings::t_min_description_set (std::string *description*)
- 4.1.4.100 void XNLO::Config_Settings::t_min_set (double *val*)
- 4.1.4.101 double XNLO::Config_Settings::x_max ()
- 4.1.4.102 std::string XNLO::Config_Settings::x_max_description ()
- 4.1.4.103 void XNLO::Config_Settings::x_max_description_set (std::string *description*)
- 4.1.4.104 void XNLO::Config_Settings::x_max_set (double *val*)
- 4.1.4.105 double XNLO::Config_Settings::x_min ()
- 4.1.4.106 std::string XNLO::Config_Settings::x_min_description ()
- 4.1.4.107 void XNLO::Config_Settings::x_min_description_set (std::string *description*)
- 4.1.4.108 void XNLO::Config_Settings::x_min_set (double *val*)

4.1.5 Member Data Documentation

- 4.1.5.1 double XNLO::Config_Settings::alpha_ = 1.45 [private]
- 4.1.5.2 std::string XNLO::Config_Settings::alpha_description_ = "(default) (double) The alpha value" [private]
- 4.1.5.3 int XNLO::Config_Settings::atoms_per_worker_ = 2 [private]
- 4.1.5.4 std::string XNLO::Config_Settings::atoms_per_worker_description_ = "(default) (int) The number of atoms per worker" [private]
- 4.1.5.5 double XNLO::Config_Settings::CEO_ = 0 [private]
- 4.1.5.6 std::string XNLO::Config_Settings::CEO_description_ = "(default) (double) The CEO value" [private]
- 4.1.5.7 double XNLO::Config_Settings::FWHM_ = 15e-15 [private]
- 4.1.5.8 std::string XNLO::Config_Settings::FWHM_description_ = "(default) (double) The FWHM value" [private]

- 4.1.5.9 double XNLO::Config_Settings::I_0 = 795e-9 [private]
- 4.1.5.10 std::string XNLO::Config_Settings::I_0_description_ = "(default) (double) The I_0 value" [private]
- 4.1.5.11 int XNLO::Config_Settings::N_t = 262144 [private]
- 4.1.5.12 std::string XNLO::Config_Settings::N_t_description_ = "(default) (int) The N_t value" [private]
- 4.1.5.13 int XNLO::Config_Settings::output_wavefunction_ = 0 [private]
- 4.1.5.14 std::string XNLO::Config_Settings::output_wavefunction_description_ = "(default) (int) Switch to output wavefunction" [private]
- 4.1.5.15 double XNLO::Config_Settings::P_av = 0.18 [private]
- 4.1.5.16 std::string XNLO::Config_Settings::P_av_description_ = "(default) (double) The P_av value" [private]
- 4.1.5.17 std::string XNLO::Config_Settings::path_config_file_ = "./config.txt" [private]
- 4.1.5.18 std::string XNLO::Config_Settings::path_config_file_description_ = "(default) (std::string) config.txt path" [private]
- 4.1.5.19 std::string XNLO::Config_Settings::path_config_log_ = "../output/config_log.txt" [private]
- 4.1.5.20 std::string XNLO::Config_Settings::path_config_log_description_ = "(default) (std::string) Output path of config_log.txt" [private]
- 4.1.5.21 std::string XNLO::Config_Settings::path_dipole_ = "../output/dipole.bin" [private]
- 4.1.5.22 std::string XNLO::Config_Settings::path_dipole_description_ = "(default) (std::string) Output path of acceleration" [private]
- 4.1.5.23 std::string XNLO::Config_Settings::path_E_ = "../output/E.bin" [private]
- 4.1.5.24 std::string XNLO::Config_Settings::path_E_description_ = "(default) (std::string) Output path of electric field" [private]
- 4.1.5.25 std::string XNLO::Config_Settings::path_input_j0_ = "../input/J0_zeros.bin" [private]
- 4.1.5.26 std::string XNLO::Config_Settings::path_input_j0_description_ = "(default) (std::string) Path to J0_zeros.bin" [private]
- 4.1.5.27 std::string XNLO::Config_Settings::path_laser_A_w_l_ = "../UPPE/output/000_1_A_w_l.bin" [private]
- 4.1.5.28 std::string XNLO::Config_Settings::path_laser_A_w_l_description_ = "(default) (std::string) Path to A_w_l of laser pulse input" [private]

- 4.1.5.29 `std::string XNLO::Config_Settings::path_laser_A_w_R = "../UPPE/output/000_1_A_w_R.bin"` [private]
- 4.1.5.30 `std::string XNLO::Config_Settings::path_laser_A_w_R_description = "(default) (std::string) Path to A_w_R of laser pulse input"` [private]
- 4.1.5.31 `std::string XNLO::Config_Settings::path_laser_w_active = "../UPPE/output/000_1_w_active.bin"` [private]
- 4.1.5.32 `std::string XNLO::Config_Settings::path_laser_w_active_description = "(default) (std::string) Path to w_active of laser pulse input"` [private]
- 4.1.5.33 `std::string XNLO::Config_Settings::path_w = "../output/w.bin"` [private]
- 4.1.5.34 `std::string XNLO::Config_Settings::path_w_description = "(default) (std::string) Output path of w"` [private]
- 4.1.5.35 `std::string XNLO::Config_Settings::pend_path = "prepend"` [private]
- 4.1.5.36 `std::string XNLO::Config_Settings::pend_path_description = "(default) (std::string) Pending switch"` [private]
- 4.1.5.37 `int XNLO::Config_Settings::read_in_laser_pulse = 0` [private]
- 4.1.5.38 `std::string XNLO::Config_Settings::read_in_laser_pulse_description = "(default) (int) Switch to read in laser pulse"` [private]
- 4.1.5.39 `double XNLO::Config_Settings::RR = 1000` [private]
- 4.1.5.40 `std::string XNLO::Config_Settings::RR_description = "(default) (double) The RR value"` [private]
- 4.1.5.41 `const char * XNLO::Config_Settings::setting_name` [static], [private]

Initial value:

```
= {
    "atoms_per_worker",
    "x_min", "x_max",
    "N_t", "t_min", "t_max",
    "P_av", "RR", "FWHM", "l_0", "CEO", "spot_radius",
    "alpha",
    "read_in_laser_pulse",
    "output_wavefunction",
    "pend_path",
    "path_input_j0",
    "path_laser_A_w_R",
    "path_laser_A_w_I",
    "path_laser_w_active",
    "path_dipole", "path_w",
    "path_E",
    "path_config_file", "path_config_log",
}
```

- 4.1.5.42 `double XNLO::Config_Settings::spot_radius_ = 42e-6` `[private]`
- 4.1.5.43 `std::string XNLO::Config_Settings::spot_radius_description_ = "(default) (double) The spot_radius value"`
`[private]`
- 4.1.5.44 `double XNLO::Config_Settings::t_max_ = 100e-15` `[private]`
- 4.1.5.45 `std::string XNLO::Config_Settings::t_max_description_ = "(default) (double) The t_max value"` `[private]`
- 4.1.5.46 `double XNLO::Config_Settings::t_min_ = -100e-15` `[private]`
- 4.1.5.47 `std::string XNLO::Config_Settings::t_min_description_ = "(default) (double) The t_min value"` `[private]`
- 4.1.5.48 `double XNLO::Config_Settings::x_max_ = 100e-6` `[private]`
- 4.1.5.49 `std::string XNLO::Config_Settings::x_max_description_ = "(default) (double) the x_max value"` `[private]`
- 4.1.5.50 `double XNLO::Config_Settings::x_min_ = 0` `[private]`
- 4.1.5.51 `std::string XNLO::Config_Settings::x_min_description_ = "(default) (double) The x_min value}"` `[private]`

The documentation for this class was generated from the following files:

- `/home/sam/Project/XNLO/XNLO/src/config_settings.hpp`
- `/home/sam/Project/XNLO/XNLO/src/config_settings.cpp`

4.2 XNLO::DHT Class Reference

```
#include <DHT.hpp>
```

Public Member Functions

- **DHT** ()
- **DHT** (int n_r_, **maths_textbook** &maths_)
- Eigen::ArrayXcd **forward** (Eigen::ArrayXcd f_r_)
- Eigen::ArrayXcd **backward** (Eigen::ArrayXcd f_kr_)

Private Attributes

- MatrixXcd **H**

4.2.1 Detailed Description

Originally created by Patrick Anderson. Modified by Samuel Senior on 10/03/2017. "DHT" evaluates the forward and backward discrete Hankel transform. Based on Fisk, Computer Physics Communications, 43 (1987). Complex datatype used here, should really template/overload.

4.2.2 Constructor & Destructor Documentation

4.2.2.1 XNLO::DHT::DHT ()

Default constructor

4.2.2.2 XNLO::DHT::DHT (int *n_r*, *maths_textbook* & *maths*)

Parameterized constructor

4.2.3 Member Function Documentation

4.2.3.1 Eigen::ArrayXcd XNLO::DHT::backward (Eigen::ArrayXcd *f_kr*)

Backward transform

4.2.3.2 Eigen::ArrayXcd XNLO::DHT::forward (Eigen::ArrayXcd *f_r*)

Forward transform

4.2.4 Member Data Documentation

4.2.4.1 MatrixXcd XNLO::DHT::H [private]

The documentation for this class was generated from the following files:

- /home/sam/Project/XNLO/XNLO/src/DHT.hpp
- /home/sam/Project/XNLO/XNLO/src/DHT.cpp

4.3 XNLO::grid_rkr Class Reference

```
#include <grid_rkr.hpp>
```

Public Member Functions

- **grid_rkr** (int *n_r*, double *R*, *maths_textbook* &*maths*)

Public Attributes

- ArrayXd **r**
- ArrayXd **kr**
- int **n_r**
- double **R**

4.3.1 Detailed Description

Originally created by Patrick Anderson. Modified by Samuel Senior on 10/03/2017. "grid_rkr" is a non-uniform radial grid. The spectral counterpart of this grid is evaluated and accessible.

4.3.2 Constructor & Destructor Documentation

4.3.2.1 XNLO::grid_rkr::grid_rkr (int *n_r_*, double *R_*, maths_textbook & *maths_*)

Default constructor

Parameterized constructor

4.3.3 Member Data Documentation

4.3.3.1 ArrayXd XNLO::grid_rkr::kr

4.3.3.2 int XNLO::grid_rkr::n_r

4.3.3.3 ArrayXd XNLO::grid_rkr::r

4.3.3.4 double XNLO::grid_rkr::R

The documentation for this class was generated from the following files:

- /home/sam/Project/XNLO/XNLO/src/grid_rkr.hpp
- /home/sam/Project/XNLO/XNLO/src/grid_rkr.cpp

4.4 XNLO::grid_tw Class Reference

```
#include <grid_tw.hpp>
```

Public Member Functions

- **grid_tw** (int *N_t_*, double *t_min_*, double *t_max_*)

Public Attributes

- ArrayXd *t*
- ArrayXd *w*
- int *N_t*
- double *t_min*
- double *t_max*
- double *dt*

4.4.1 Detailed Description

Modified by Patrick Anderson on 07/05/2015. "grid_tw" is a linear temporal grid. The spectral counterpart of this grid is evaluated and made accessible.

4.4.2 Constructor & Destructor Documentation

4.4.2.1 XNLO::grid_tw::grid_tw (int *N_t*, double *t_min*, double *t_max*)

Constructor

4.4.3 Member Data Documentation

4.4.3.1 double XNLO::grid_tw::dt

4.4.3.2 int XNLO::grid_tw::N_t

4.4.3.3 ArrayXd XNLO::grid_tw::t

4.4.3.4 double XNLO::grid_tw::t_max

4.4.3.5 double XNLO::grid_tw::t_min

4.4.3.6 ArrayXd XNLO::grid_tw::w

The documentation for this class was generated from the following files:

- /home/sam/Project/XNLO/XNLO/src/grid_tw.hpp
- /home/sam/Project/XNLO/XNLO/src/grid_tw.cpp

4.5 XNLO::grid_xkx Class Reference

```
#include <grid_xkx.hpp>
```

Public Member Functions

- **grid_xkx** ()
- **grid_xkx** (int *N_x*, double *x_min*, double *x_max*)

Public Attributes

- ArrayXd **x**
- ArrayXd **kx**
- int **N_x**
- double **x_min**
- double **x_max**
- double **dx**

4.5.1 Detailed Description

Modified by Patrick Anderson on 07/05/2015. "grid_xkx" is a linear 1D spatial grid. The spectral counterpart of this grid is evaluated and accessible.

4.5.2 Constructor & Destructor Documentation

4.5.2.1 XNLO::grid_xkx::grid_xkx ()

Default

4.5.2.2 XNLO::grid_xkx::grid_xkx (int *N_x*, double *x_min*, double *x_max*)

Parameterised

4.5.3 Member Data Documentation

4.5.3.1 double XNLO::grid_xkx::dx

4.5.3.2 ArrayXd XNLO::grid_xkx::kx

4.5.3.3 int XNLO::grid_xkx::N_x

4.5.3.4 ArrayXd XNLO::grid_xkx::x

4.5.3.5 double XNLO::grid_xkx::x_max

4.5.3.6 double XNLO::grid_xkx::x_min

The documentation for this class was generated from the following files:

- /home/sam/Project/XNLO/XNLO/src/grid_xkx.hpp
- /home/sam/Project/XNLO/XNLO/src/grid_xkx.cpp

4.6 XNLO::IO Class Reference

```
#include <IO.hpp>
```

Public Member Functions

- **IO** ()
- void **read_header** (const std::string path, bool print=true)
- ArrayXXd **read_double** (const std::string path, int **N_row_**, int **N_col_**, bool print=true)
- ArrayXXd **read_double** (const std::string path, bool skip_header=true, bool print=true)
- void **write_ascii_double** (std::string path, ArrayXXd data)
- void **write_double** (const std::string path, ArrayXXd output, int **N_row_**, int **N_col_**)
- void **write_header** (const std::string path_, int **N_row_**, int **N_col_**)
- void **overwrite** (const std::string path)

Public Attributes

- std::string **binary_format**
- int **binary_format_version**
- int **binary_format_subversion**
- int **binary_format_len**
- int **data_size**
- int **double_size**
- int **N_row_**
- int **N_col_**
- int **header_size**

4.6.1 Detailed Description

Modified by Patrick Anderson on 09/05/2015.

4.6.2 Constructor & Destructor Documentation

4.6.2.1 XNLO::IO::IO ()

Class constructor. Sets unrealistic values for header variables so that if one isn't succesfully read/written then it will be caught quickly.

4.6.3 Member Function Documentation

4.6.3.1 void XNLO::IO::overwrite (const std::string path)

Overwrites the given binary file.

4.6.3.2 ArrayXXd XNLO::IO::read_double (const std::string path, int N_row_, int N_col_, bool print = true)

Read a two-dimensional array of doubles into an to Eigen array from a binary file. The number of rows and columns of the array are given by N_row_ and N_col_, as passed in from the function arguments.

4.6.3.3 ArrayXXd XNLO::IO::read_double (const std::string path, bool skip_header = true, bool print = true)

Read a two-dimensional array of doubles into an to Eigen array from a binary file. The number of rows and columns of the array are taken as the class variables N_row_ and N_col_.

4.6.3.4 void XNLO::IO::read_header (const std::string path, bool print = true)

Read the XNLO/UPPE binary header of a given binary file.

The header takes the form:

Offset	Size (Bytes)	Type/Contents	Description
0	4	'XNLO'/'UPPE'	Binary format name
4	4	int	Version number
8	4	int	Subversion number
12	4	int	Size of header
16	4	int	N_row
20	4	int	N_col
24	4	int	Total size of data
28	4	int	Size of each double in the data

That is to say, the first four bytes of an XNLO or UPPE binary header are the either the four characters 'XNLO' or 'UPPE', used to specify which file type it is. The next four bytes give an integer, which is the version number. The next four give the subversion number as integer, and so on.

4.6.3.5 void XNLO::IO::write_ascii_double (std::string path, ArrayXXd data)

Write to an ascii text file from an Eigen two odimensional array of doubles.

4.6.3.6 void XNLO::IO::write_double (const std::string path, ArrayXXd output, int N_row_, int N_col_)

Write to the binary file from a N_col_ by N_row_ Eigen array of doubles.

4.6.3.7 void XNLO::IO::write_header (const std::string path_, int N_row_, int N_col_)

Writes the XNLO binary header to a given binary file.

The **XNLO** (p. 5) binary header takes the form:

Offset	Size (Bytes)	Type/Contents	Description
0	4	'XNLO'	Binary format name
4	4	int	Version number
8	4	int	Subversion number
12	4	int	Size of header
16	4	int	N_row
20	4	int	N_col
24	4	int	Total size of data
28	4	int	Size of each double in the data

That is to say, the first four bytes of an XNLO binary header are the four characters 'XNLO', used to specify which file type it is. The next four bytes give an integer, which is the version number. The next four give the subversion number as integer, and so on.

4.6.4 Member Data Documentation

4.6.4.1 std::string XNLO::IO::binary_format

The binary format name.

4.6.4.2 `int XNLO::IO::binary_format_len`

The length of the binary format name.

4.6.4.3 `int XNLO::IO::binary_format_subversion`

The binary format subversion number.

4.6.4.4 `int XNLO::IO::binary_format_version`

The binary format version number.

4.6.4.5 `int XNLO::IO::data_size`

The total number of elements of the data in the two dimensional array.

4.6.4.6 `int XNLO::IO::double_size`

The size of a double in bytes.

4.6.4.7 `int XNLO::IO::header_size`

The size of the header in bytes.

4.6.4.8 `int XNLO::IO::N_col_`

The number of columns to the two dimensional Eigen array.

4.6.4.9 `int XNLO::IO::N_row_`

The number of rows to the two dimensional Eigen array.

The documentation for this class was generated from the following files:

- `/home/sam/Project/XNLO/XNLO/src/IO.hpp`
- `/home/sam/Project/XNLO/XNLO/src/IO.cpp`

4.7 `XNLO::laser_pulse` Class Reference

```
#include <laser_pulse.hpp>
```

Public Member Functions

- **laser_pulse** (double *P_av_*, double *RR_*, double *FWHM_*, double *I_0_*, double *CEO_*, double *spot_radius_*, double *ROC_*, **grid_rkr** *rkr_*, **grid_tw** *tw_*, std::string *path_A_w_R*, std::string *path_A_w_I*, std::string *path_w_active*, int *read_in_laser_pulse*)
- **laser_pulse** (**grid_rkr** *rkr_*, **grid_tw** *tw_*, ArrayXXcd *A_w_active*, ArrayXd *w_active*)

Public Attributes

- ArrayXXd **E**

4.7.1 Detailed Description

Modified by Patrick Anderson on 07/05/2015. "laser_pulse" contains a time varying electric field. The initial conditions are passed to the constructor and field can be updated as it propagates.

4.7.2 Constructor & Destructor Documentation

4.7.2.1 XNLO::laser_pulse::laser_pulse (double *P_av_*, double *RR_*, double *FWHM_*, double *I_0_*, double *CEO_*, double *spot_radius_*, double *ROC_*, **grid_rkr** *rkr_*, **grid_tw** *tw_*, std::string *path_A_w_R*, std::string *path_A_w_I*, std::string *path_w_active*, int *read_in_laser_pulse*)

Constructor

4.7.2.2 XNLO::laser_pulse::laser_pulse (**grid_rkr** *rkr_*, **grid_tw** *tw_*, ArrayXXcd *A_w_active*, ArrayXd *w_active*)

Constructor

4.7.3 Member Data Documentation

4.7.3.1 ArrayXXd XNLO::laser_pulse::E

The documentation for this class was generated from the following files:

- /home/sam/Project/XNLO/XNLO/src/laser_pulse.hpp
- /home/sam/Project/XNLO/XNLO/src/laser_pulse.cpp

4.8 XNLO::maths_textbook Class Reference

```
#include <maths_textbook.hpp>
```

Public Member Functions

- **maths_textbook** ()
- **maths_textbook** (std::string path_input_j0_)
- double **trapz** (ArrayXd x_, ArrayXd y_)
- ArrayXd **interp1D** (ArrayXd input_array, int input_length, int output_length, int spline_order)

Public Attributes

- double **pi**
- ArrayXd **J0_zeros**

Private Attributes

- std::string **path_input_j0**

4.8.1 Detailed Description

Modified by Patrick Anderson on 07/05/2015. "maths_textbook" is a container for mathematical constants and common functions.

4.8.2 Constructor & Destructor Documentation

4.8.2.1 XNLO::maths_textbook::maths_textbook ()

Constructor

4.8.2.2 XNLO::maths_textbook::maths_textbook (std::string path_input_j0_)

Constructor

4.8.3 Member Function Documentation

4.8.3.1 ArrayXd XNLO::maths_textbook::interp1D (ArrayXd input_array, int input_length, int output_length, int spline_order)

4.8.3.2 double XNLO::maths_textbook::trapz (ArrayXd x_, ArrayXd y_)

Trapezoidal integration, vectorized

4.8.4 Member Data Documentation

4.8.4.1 `ArrayXd XNLO::maths_textbook::J0_zeros`

4.8.4.2 `std::string XNLO::maths_textbook::path_input_j0` `[private]`

4.8.4.3 `double XNLO::maths_textbook::pi`

The documentation for this class was generated from the following files:

- `/home/sam/Project/XNLO/XNLO/src/maths_textbook.hpp`
- `/home/sam/Project/XNLO/XNLO/src/maths_textbook.cpp`

4.9 XNLO::physics_textbook Class Reference

```
#include <physics_textbook.hpp>
```

Public Member Functions

- `physics_textbook ()`

Public Attributes

- `double E_at`
- `double I_at`
- `double m_at`
- `double q_at`
- `double t_at`
- `double w_at`
- `double c`
- `double eps_0`
- `double mu_0`
- `double h_bar`
- `double k_B`

4.9.1 Detailed Description

Modified by Patrick Anderson on 07/05/2015. "physics_textbook" is a container for physical constants.

4.9.2 Constructor & Destructor Documentation

4.9.2.1 `XNLO::physics_textbook::physics_textbook ()`

Constructor

4.9.3 Member Data Documentation

4.9.3.1 `double XNLO::physics_textbook::c`

4.9.3.2 `double XNLO::physics_textbook::E_at`

4.9.3.3 `double XNLO::physics_textbook::eps_0`

4.9.3.4 `double XNLO::physics_textbook::h_bar`

4.9.3.5 `double XNLO::physics_textbook::k_B`

4.9.3.6 `double XNLO::physics_textbook::l_at`

4.9.3.7 `double XNLO::physics_textbook::m_at`

4.9.3.8 `double XNLO::physics_textbook::mu_0`

4.9.3.9 `double XNLO::physics_textbook::q_at`

4.9.3.10 `double XNLO::physics_textbook::t_at`

4.9.3.11 `double XNLO::physics_textbook::w_at`

The documentation for this class was generated from the following files:

- `/home/sam/Project/XNLO/XNLO/src/physics_textbook.hpp`
- `/home/sam/Project/XNLO/XNLO/src/physics_textbook.cpp`

4.10 XNLO::Result Struct Reference

```
#include <XNLO.hpp>
```

Public Attributes

- `ArrayXXd acceleration`
- `ArrayXXd w`
- `ArrayXXd E`
- `ArrayXXcd wavefunction`

4.10.1 Member Data Documentation

4.10.1.1 ArrayXXd XNLO::Result::acceleration

4.10.1.2 ArrayXXd XNLO::Result::E

4.10.1.3 ArrayXXd XNLO::Result::w

4.10.1.4 ArrayXXcd XNLO::Result::wavefunction

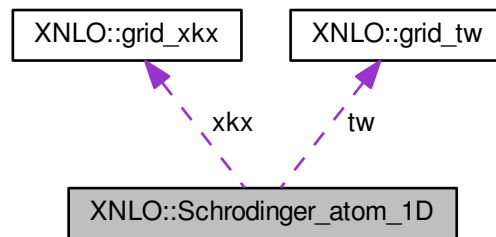
The documentation for this struct was generated from the following file:

- /home/sam/Project/XNLO/XNLO/src/XNLO.hpp

4.11 XNLO::Schrodinger_atom_1D Class Reference

```
#include <Schrodinger_atom_1D.hpp>
```

Collaboration diagram for XNLO::Schrodinger_atom_1D:



Public Member Functions

- **Schrodinger_atom_1D** (**grid_tw** &tw_, double alpha_, int output_wavefunction_)
- void **set_GS** (int N_it_)
- ArrayXd **get_acceleration** (int N_it_, double dt_, ArrayXd E_)
- ArrayXd **solve_TDSE_PS** (int N_it_, std::complex< double > dt_, ArrayXd E_, int e_)

Public Attributes

- **grid_tw** tw
- **grid_xkx** xkx
- double **alpha**
- ArrayXd **V_model**
- ArrayXcd **wfn_GS**
- ArrayXcd **wfn**
- double **energy**
- int **output_wavefunction**
- ArrayXXcd **wfn_output**

4.11.1 Detailed Description

Modified by Patrick Anderson on 07/05/2015. "Schrodinger_atom_1D" encapsulates the interaction of an isolated atom with a strong laser field. The interaction is restricted to a single active electron and spatial dimension.

4.11.2 Constructor & Destructor Documentation

4.11.2.1 XNLO::Schrodinger_atom_1D::Schrodinger_atom_1D (grid_tw & tw_, double alpha_, int output_wavefunction_)

Class constructor.

4.11.3 Member Function Documentation

4.11.3.1 ArrayXd XNLO::Schrodinger_atom_1D::get_acceleration (int N_it_, double dt_, ArrayXd E_)

Find the electron acceleration generated by an arbitrary field. This is achieved by solving the TDSE using the pseudo-spectral method and then using the wavefunction in conjunction with the Ehrenfest theorem to obtain the electron acceleration by

$$a(t) = \langle [\hat{H}, [\hat{H}, \hat{x}]] \rangle.$$

M. B. Gaarde and K. J. Schafer. Theory of attosecond pulse generation. Springer series in Optical Sciences, 177:11-31, 2013.

4.11.3.2 void XNLO::Schrodinger_atom_1D::set_GS (int N_it_)

Find the GS wavefunction by imaginary time propagation, also display the energy. The time-dependent Schrodinger equation (TDSE) in atomic units is given as

$$i \frac{\partial \psi}{\partial t} = \hat{H} \psi.$$

Transforming to imaginary time causes the TDSE to become a diffusion equation,

$$t = -i\tau.$$

As time increases, the wavefunction then converges to the ground state,

$$\psi(x, t) \rightarrow e^{-\tau E_0} c_0 \phi_0.$$

P. Bader et al. Solving the Schrodinger eigenvalue problem by imaginary time propagation techniques using splitting methods with complex coefficients, J. Chem. Phys., 139, 2013

The energy expectation value is given by

$$\langle E \rangle = \int \psi^* \hat{K} \psi dx + \int \psi^* \hat{V} \psi dx.$$

4.11.3.3 ArrayXd XNLO::Schrodinger_atom_1D::solve_TDSE_PS (int N_it_, std::complex< double > dt_, ArrayXd E_, int e_)

TDSE solver (pseudo-spectral method). The time propagator of the pseudo-spectral method is given as

$$\psi(x, t + \Delta t) = e^{\frac{-i\hat{K}\Delta t}{2}} e^{-i\hat{V}\Delta t} e^{\frac{-i\hat{K}\Delta t}{2}} \psi(x, t).$$

P. L. DeVries and J. Hasbun. A first course in computational physics. Jones and Bartlett, 2nd edition, 2010.

4.11.4 Member Data Documentation

4.11.4.1 double XNLO::Schrodinger_atom_1D::alpha

The parameter α is used to scale the Coulomb potential to match the ground state energy of the atom.

4.11.4.2 double XNLO::Schrodinger_atom_1D::energy

The energy expectation value of the elctron.

4.11.4.3 int XNLO::Schrodinger_atom_1D::output_wavefunction

A switch to toggle outputting and saving the electron wavefunction, for every position and time step.

4.11.4.4 grid_tw XNLO::Schrodinger_atom_1D::tw

The linear temporal grid.

4.11.4.5 ArrayXd XNLO::Schrodinger_atom_1D::V_model

The soft Coulomb potential,

$$V_{Coulomb} = \frac{-1}{\sqrt{\alpha + x^2}}.$$

4.11.4.6 ArrayXcd XNLO::Schrodinger_atom_1D::wfn

The wavefunction of the electron at a time t and as a function of position.

4.11.4.7 ArrayXcd XNLO::Schrodinger_atom_1D::wfn_GS

The ground state wavefunction of the electron at the initial time and as a function of position.

4.11.4.8 ArrayXXcd XNLO::Schrodinger_atom_1D::wfn_output

The electron wavefunction at every position and time step.

4.11.4.9 grid_xkx XNLO::Schrodinger_atom_1D::xkx

The linear spatial grid.

The documentation for this class was generated from the following files:

- /home/sam/Project/XNLO/XNLO/src/Schrodinger_atom_1D.hpp
- /home/sam/Project/XNLO/XNLO/src/Schrodinger_atom_1D.cpp

Index

- acceleration
 - XNLO::Result, 31
- alpha
 - XNLO::Config_Settings, 11
 - XNLO::Schrodinger_atom_1D, 33
- alpha_
 - XNLO::Config_Settings, 16
- alpha_description
 - XNLO::Config_Settings, 11
- alpha_description_
 - XNLO::Config_Settings, 16
- alpha_description_set
 - XNLO::Config_Settings, 11
- alpha_set
 - XNLO::Config_Settings, 11
- atoms_per_worker
 - XNLO::Config_Settings, 11, 12
- atoms_per_worker_
 - XNLO::Config_Settings, 16
- atoms_per_worker_description
 - XNLO::Config_Settings, 12
- atoms_per_worker_description_
 - XNLO::Config_Settings, 16
- atoms_per_worker_description_set
 - XNLO::Config_Settings, 12
- atoms_per_worker_set
 - XNLO::Config_Settings, 12
- backward
 - XNLO::DHT, 20
- binary_format
 - XNLO::IO, 25
- binary_format_len
 - XNLO::IO, 25
- binary_format_subversion
 - XNLO::IO, 26
- binary_format_version
 - XNLO::IO, 26
- c
 - XNLO::physics_textbook, 30
- CEO_
 - XNLO::Config_Settings, 16
- CEO_description
 - XNLO::Config_Settings, 12
- CEO_description_
 - XNLO::Config_Settings, 16
- CEO_description_set
 - XNLO::Config_Settings, 12
- CEO_set
 - XNLO::Config_Settings, 12
- CEO
 - XNLO::Config_Settings, 12
- check_paths
 - XNLO::Config_Settings, 12
- Config_Settings
 - XNLO::Config_Settings, 11
- DHT
 - XNLO::DHT, 20
- data_size
 - XNLO::IO, 26
- double_size
 - XNLO::IO, 26
- dt
 - XNLO::grid_tw, 22
- dx
 - XNLO::grid_xkx, 23
- E
 - XNLO::Result, 31
 - XNLO::laser_pulse, 27
- E_at
 - XNLO::physics_textbook, 30
- energy
 - XNLO::Schrodinger_atom_1D, 33
- eps_0
 - XNLO::physics_textbook, 30
- FWHM_
 - XNLO::Config_Settings, 16
- FWHM_description
 - XNLO::Config_Settings, 12
- FWHM_description_
 - XNLO::Config_Settings, 16
- FWHM_description_set
 - XNLO::Config_Settings, 12
- FWHM_set
 - XNLO::Config_Settings, 12
- FWHM
 - XNLO::Config_Settings, 11, 12
- forward
 - XNLO::DHT, 20
- get_acceleration
 - XNLO::Schrodinger_atom_1D, 32
- grid_rkr
 - XNLO::grid_rkr, 21
- grid_tw
 - XNLO::grid_tw, 22

- grid_xkx
 - XNLO::grid_xkx, 23
- H
 - XNLO::DHT, 20
- h_bar
 - XNLO::physics_textbook, 30
- header_size
 - XNLO::IO, 26
- interp1D
 - XNLO::maths_textbook, 28
- IO
 - XNLO::IO, 24
- J0_zeros
 - XNLO::maths_textbook, 29
- k_B
 - XNLO::physics_textbook, 30
- kr
 - XNLO::grid_rkr, 21
- kx
 - XNLO::grid_xkx, 23
- l_0
 - XNLO::Config_Settings, 11, 12
- l_0_
 - XNLO::Config_Settings, 16
- l_0_description
 - XNLO::Config_Settings, 12
- l_0_description_
 - XNLO::Config_Settings, 17
- l_0_description_set
 - XNLO::Config_Settings, 12
- l_0_set
 - XNLO::Config_Settings, 12
- l_at
 - XNLO::physics_textbook, 30
- LAST_SN_ENTRY
 - XNLO::Config_Settings, 11
- laser_pulse
 - XNLO::laser_pulse, 27
- m_at
 - XNLO::physics_textbook, 30
- maths_textbook
 - XNLO::maths_textbook, 28
- mu_0
 - XNLO::physics_textbook, 30
- N_col_
 - XNLO::IO, 26
- n_r
 - XNLO::grid_rkr, 21
- N_row_
 - XNLO::IO, 26
- N_t
 - XNLO::Config_Settings, 11, 12
 - XNLO::grid_tw, 22
- N_t_
 - XNLO::Config_Settings, 17
- N_t_description
 - XNLO::Config_Settings, 12
- N_t_description_
 - XNLO::Config_Settings, 17
- N_t_description_set
 - XNLO::Config_Settings, 12
- N_t_set
 - XNLO::Config_Settings, 12
- N_x
 - XNLO::grid_xkx, 23
- output_wavefunction
 - XNLO::Config_Settings, 11, 12
 - XNLO::Schrodinger_atom_1D, 33
- output_wavefunction_
 - XNLO::Config_Settings, 17
- output_wavefunction_description
 - XNLO::Config_Settings, 12
- output_wavefunction_description_
 - XNLO::Config_Settings, 17
- output_wavefunction_description_set
 - XNLO::Config_Settings, 13
- output_wavefunction_set
 - XNLO::Config_Settings, 13
- overwrite
 - XNLO::IO, 24
- P_av
 - XNLO::Config_Settings, 11, 13
- P_av_
 - XNLO::Config_Settings, 17
- P_av_description
 - XNLO::Config_Settings, 13
- P_av_description_
 - XNLO::Config_Settings, 17
- P_av_description_set
 - XNLO::Config_Settings, 13
- P_av_set
 - XNLO::Config_Settings, 13
- path_E_
 - XNLO::Config_Settings, 17
- path_E_description
 - XNLO::Config_Settings, 13
- path_E_description_
 - XNLO::Config_Settings, 17
- path_E_description_set
 - XNLO::Config_Settings, 13
- path_E_set
 - XNLO::Config_Settings, 13
- path_config_file
 - XNLO::Config_Settings, 11, 13
- path_config_file_
 - XNLO::Config_Settings, 17
- path_config_file_description
 - XNLO::Config_Settings, 13
- path_config_file_description_
 - XNLO::Config_Settings, 17

path_config_file_description_set
 XNLO::Config_Settings, 13
 path_config_file_set
 XNLO::Config_Settings, 13
 path_config_log
 XNLO::Config_Settings, 11, 13
 path_config_log_
 XNLO::Config_Settings, 17
 path_config_log_description
 XNLO::Config_Settings, 13
 path_config_log_description_
 XNLO::Config_Settings, 17
 path_config_log_description_set
 XNLO::Config_Settings, 13
 path_config_log_set
 XNLO::Config_Settings, 13
 path_dipole
 XNLO::Config_Settings, 11, 13
 path_dipole_
 XNLO::Config_Settings, 17
 path_dipole_description
 XNLO::Config_Settings, 13
 path_dipole_description_
 XNLO::Config_Settings, 17
 path_dipole_description_set
 XNLO::Config_Settings, 13
 path_dipole_set
 XNLO::Config_Settings, 13
 path_E
 XNLO::Config_Settings, 11, 13
 path_input_j0
 XNLO::Config_Settings, 11, 13
 XNLO::maths_textbook, 29
 path_input_j0_
 XNLO::Config_Settings, 17
 path_input_j0_description
 XNLO::Config_Settings, 14
 path_input_j0_description_
 XNLO::Config_Settings, 17
 path_input_j0_description_set
 XNLO::Config_Settings, 14
 path_input_j0_set
 XNLO::Config_Settings, 14
 path_laser_A_w_I_
 XNLO::Config_Settings, 17
 path_laser_A_w_I_description
 XNLO::Config_Settings, 14
 path_laser_A_w_I_description_
 XNLO::Config_Settings, 17
 path_laser_A_w_I_description_set
 XNLO::Config_Settings, 14
 path_laser_A_w_I_set
 XNLO::Config_Settings, 14
 path_laser_A_w_R_
 XNLO::Config_Settings, 17
 path_laser_A_w_R_description
 XNLO::Config_Settings, 14
 path_laser_A_w_R_description_
 XNLO::Config_Settings, 14
 XNLO::Config_Settings, 14
 path_laser_A_w_R_set
 XNLO::Config_Settings, 14
 path_laser_A_w_R_
 XNLO::Config_Settings, 11, 14
 path_laser_A_w_R_
 XNLO::Config_Settings, 11, 14
 path_laser_w_active
 XNLO::Config_Settings, 11, 14
 path_laser_w_active_
 XNLO::Config_Settings, 18
 path_laser_w_active_description
 XNLO::Config_Settings, 14
 path_laser_w_active_description_
 XNLO::Config_Settings, 18
 path_laser_w_active_description_set
 XNLO::Config_Settings, 14
 path_laser_w_active_set
 XNLO::Config_Settings, 14
 path_w
 XNLO::Config_Settings, 11, 14
 path_w_
 XNLO::Config_Settings, 18
 path_w_description
 XNLO::Config_Settings, 14
 path_w_description_
 XNLO::Config_Settings, 18
 path_w_description_set
 XNLO::Config_Settings, 14
 path_w_set
 XNLO::Config_Settings, 14
 pend_path
 XNLO::Config_Settings, 11, 14
 pend_path_
 XNLO::Config_Settings, 18
 pend_path_description
 XNLO::Config_Settings, 14
 pend_path_description_
 XNLO::Config_Settings, 18
 pend_path_description_set
 XNLO::Config_Settings, 14
 pend_path_set
 XNLO::Config_Settings, 14
 physics_textbook
 XNLO::physics_textbook, 29
 pi
 XNLO::maths_textbook, 29
 print
 XNLO::Config_Settings, 15
 q_at
 XNLO::physics_textbook, 30
 R
 XNLO::grid_rkr, 21
 r
 XNLO::grid_rkr, 21

RR_
 XNLO::Config_Settings, 18
 RR_description
 XNLO::Config_Settings, 15
 RR_description_
 XNLO::Config_Settings, 18
 RR_description_set
 XNLO::Config_Settings, 15
 RR_set
 XNLO::Config_Settings, 15
 read_double
 XNLO::IO, 24
 read_header
 XNLO::IO, 24
 read_in
 XNLO::Config_Settings, 15
 read_in_laser_pulse
 XNLO::Config_Settings, 11, 15
 read_in_laser_pulse_
 XNLO::Config_Settings, 18
 read_in_laser_pulse_description
 XNLO::Config_Settings, 15
 read_in_laser_pulse_description_
 XNLO::Config_Settings, 18
 read_in_laser_pulse_description_set
 XNLO::Config_Settings, 15
 read_in_laser_pulse_set
 XNLO::Config_Settings, 15
 RR
 XNLO::Config_Settings, 11, 15

 Schrodinger_atom_1D
 XNLO::Schrodinger_atom_1D, 32
 set_GS
 XNLO::Schrodinger_atom_1D, 32
 set_path
 XNLO::Config_Settings, 15
 set_post_path
 XNLO::Config_Settings, 15
 set_pre_path
 XNLO::Config_Settings, 15
 set_variable
 XNLO::Config_Settings, 15
 setting_name
 XNLO::Config_Settings, 18
 SN
 XNLO::Config_Settings, 11
 solve_TDSE_PS
 XNLO::Schrodinger_atom_1D, 32
 spot_radius
 XNLO::Config_Settings, 11, 15
 spot_radius_
 XNLO::Config_Settings, 18
 spot_radius_description
 XNLO::Config_Settings, 15
 spot_radius_description_
 XNLO::Config_Settings, 19
 spot_radius_description_set
 XNLO::Config_Settings, 15

 spot_radius_set
 XNLO::Config_Settings, 15

 t
 XNLO::grid_tw, 22
 t_at
 XNLO::physics_textbook, 30
 t_max
 XNLO::Config_Settings, 11, 15
 XNLO::grid_tw, 22
 t_max_
 XNLO::Config_Settings, 19
 t_max_description
 XNLO::Config_Settings, 15
 t_max_description_
 XNLO::Config_Settings, 19
 t_max_description_set
 XNLO::Config_Settings, 15
 t_max_set
 XNLO::Config_Settings, 15
 t_min
 XNLO::Config_Settings, 11, 16
 XNLO::grid_tw, 22
 t_min_
 XNLO::Config_Settings, 19
 t_min_description
 XNLO::Config_Settings, 16
 t_min_description_
 XNLO::Config_Settings, 19
 t_min_description_set
 XNLO::Config_Settings, 16
 t_min_set
 XNLO::Config_Settings, 16
 trapz
 XNLO::maths_textbook, 28
 tw
 XNLO::Schrodinger_atom_1D, 33

 V_model
 XNLO::Schrodinger_atom_1D, 33

 w
 XNLO::Result, 31
 XNLO::grid_tw, 22
 w_at
 XNLO::physics_textbook, 30
 wavefunction
 XNLO::Result, 31
 wfn
 XNLO::Schrodinger_atom_1D, 33
 wfn_GS
 XNLO::Schrodinger_atom_1D, 33
 wfn_output
 XNLO::Schrodinger_atom_1D, 33
 write_ascii_double
 XNLO::IO, 25
 write_double
 XNLO::IO, 25
 write_header

- XNLO::IO, 25
- x
 - XNLO::grid_xkx, 23
- x_max
 - XNLO::Config_Settings, 11, 16
 - XNLO::grid_xkx, 23
- x_max_
 - XNLO::Config_Settings, 19
- x_max_description
 - XNLO::Config_Settings, 16
- x_max_description_
 - XNLO::Config_Settings, 19
- x_max_description_set
 - XNLO::Config_Settings, 16
- x_max_set
 - XNLO::Config_Settings, 16
- x_min
 - XNLO::Config_Settings, 11, 16
 - XNLO::grid_xkx, 23
- x_min_
 - XNLO::Config_Settings, 19
- x_min_description
 - XNLO::Config_Settings, 16
- x_min_description_
 - XNLO::Config_Settings, 19
- x_min_description_set
 - XNLO::Config_Settings, 16
- x_min_set
 - XNLO::Config_Settings, 16
- XNLO::Config_Settings, 7
 - alpha, 11
 - alpha_, 16
 - alpha_description, 11
 - alpha_description_, 16
 - alpha_description_set, 11
 - alpha_set, 11
 - atoms_per_worker, 11, 12
 - atoms_per_worker_, 16
 - atoms_per_worker_description, 12
 - atoms_per_worker_description_, 16
 - atoms_per_worker_description_set, 12
 - atoms_per_worker_set, 12
 - CEO_, 16
 - CEO_description, 12
 - CEO_description_, 16
 - CEO_description_set, 12
 - CEO_set, 12
 - CEO, 11, 12
 - check_paths, 12
 - Config_Settings, 11
 - FWHM_, 16
 - FWHM_description, 12
 - FWHM_description_, 16
 - FWHM_description_set, 12
 - FWHM_set, 12
 - FWHM, 11, 12
 - I_0, 11, 12
 - I_0_, 16
 - I_0_description, 12
 - I_0_description_, 17
 - I_0_description_set, 12
 - I_0_set, 12
 - LAST_SN_ENTRY, 11
 - N_t, 11, 12
 - N_t_, 17
 - N_t_description, 12
 - N_t_description_, 17
 - N_t_description_set, 12
 - N_t_set, 12
 - output_wavefunction, 11, 12
 - output_wavefunction_, 17
 - output_wavefunction_description, 12
 - output_wavefunction_description_, 17
 - output_wavefunction_description_set, 13
 - output_wavefunction_set, 13
 - P_av, 11, 13
 - P_av_, 17
 - P_av_description, 13
 - P_av_description_, 17
 - P_av_description_set, 13
 - P_av_set, 13
 - path_E_, 17
 - path_E_description, 13
 - path_E_description_, 17
 - path_E_description_set, 13
 - path_E_set, 13
 - path_config_file, 11, 13
 - path_config_file_, 17
 - path_config_file_description, 13
 - path_config_file_description_, 17
 - path_config_file_description_set, 13
 - path_config_file_set, 13
 - path_config_log, 11, 13
 - path_config_log_, 17
 - path_config_log_description, 13
 - path_config_log_description_, 17
 - path_config_log_description_set, 13
 - path_config_log_set, 13
 - path_dipole, 11, 13
 - path_dipole_, 17
 - path_dipole_description, 13
 - path_dipole_description_, 17
 - path_dipole_description_set, 13
 - path_dipole_set, 13
 - path_E, 11, 13
 - path_input_j0, 11, 13
 - path_input_j0_, 17
 - path_input_j0_description, 14
 - path_input_j0_description_, 17
 - path_input_j0_description_set, 14
 - path_input_j0_set, 14
 - path_laser_A_w_l_, 17
 - path_laser_A_w_l_description, 14
 - path_laser_A_w_l_description_, 17
 - path_laser_A_w_l_description_set, 14
 - path_laser_A_w_l_set, 14

- path_laser_A_w_R_, 17
- path_laser_A_w_R_description, 14
- path_laser_A_w_R_description_, 18
- path_laser_A_w_R_description_set, 14
- path_laser_A_w_R_set, 14
- path_laser_A_w_I, 11, 14
- path_laser_A_w_R, 11, 14
- path_laser_w_active, 11, 14
- path_laser_w_active_, 18
- path_laser_w_active_description, 14
- path_laser_w_active_description_, 18
- path_laser_w_active_description_set, 14
- path_laser_w_active_set, 14
- path_w, 11, 14
- path_w_, 18
- path_w_description, 14
- path_w_description_, 18
- path_w_description_set, 14
- path_w_set, 14
- pend_path, 11, 14
- pend_path_, 18
- pend_path_description, 14
- pend_path_description_, 18
- pend_path_description_set, 14
- pend_path_set, 14
- print, 15
- RR_, 18
- RR_description, 15
- RR_description_, 18
- RR_description_set, 15
- RR_set, 15
- read_in, 15
- read_in_laser_pulse, 11, 15
- read_in_laser_pulse_, 18
- read_in_laser_pulse_description, 15
- read_in_laser_pulse_description_, 18
- read_in_laser_pulse_description_set, 15
- read_in_laser_pulse_set, 15
- RR, 11, 15
- set_path, 15
- set_post_path, 15
- set_pre_path, 15
- set_variable, 15
- setting_name, 18
- SN, 11
- spot_radius, 11, 15
- spot_radius_, 18
- spot_radius_description, 15
- spot_radius_description_, 19
- spot_radius_description_set, 15
- spot_radius_set, 15
- t_max, 11, 15
- t_max_, 19
- t_max_description, 15
- t_max_description_, 19
- t_max_description_set, 15
- t_max_set, 15
- t_min, 11, 16
- t_min_, 19
- t_min_description, 16
- t_min_description_, 19
- t_min_description_set, 16
- t_min_set, 16
- x_max, 11, 16
- x_max_, 19
- x_max_description, 16
- x_max_description_, 19
- x_max_description_set, 16
- x_max_set, 16
- x_min, 11, 16
- x_min_, 19
- x_min_description, 16
- x_min_description_, 19
- x_min_description_set, 16
- x_min_set, 16
- XNLO::DHT, 19
 - backward, 20
 - DHT, 20
 - forward, 20
 - H, 20
- XNLO::IO, 23
 - binary_format, 25
 - binary_format_len, 25
 - binary_format_subversion, 26
 - binary_format_version, 26
 - data_size, 26
 - double_size, 26
 - header_size, 26
 - IO, 24
 - N_col_, 26
 - N_row_, 26
 - overwrite, 24
 - read_double, 24
 - read_header, 24
 - write_ascii_double, 25
 - write_double, 25
 - write_header, 25
- XNLO::Result, 30
 - acceleration, 31
 - E, 31
 - w, 31
 - wavefunction, 31
- XNLO::Schrodinger_atom_1D, 31
 - alpha, 33
 - energy, 33
 - get_acceleration, 32
 - output_wavefunction, 33
 - Schrodinger_atom_1D, 32
 - set_GS, 32
 - solve_TDSE_PS, 32
 - tw, 33
 - V_model, 33
 - wfn, 33
 - wfn_GS, 33
 - wfn_output, 33
 - xkx, 33

XNLO::grid_rkr, 20
 grid_rkr, 21
 kr, 21
 n_r, 21
 R, 21
 r, 21
XNLO::grid_tw, 21
 dt, 22
 grid_tw, 22
 N_t, 22
 t, 22
 t_max, 22
 t_min, 22
 w, 22
XNLO::grid_xkx, 22
 dx, 23
 grid_xkx, 23
 kx, 23
 N_x, 23
 x, 23
 x_max, 23
 x_min, 23
XNLO::laser_pulse, 26
 E, 27
 laser_pulse, 27
XNLO::maths_textbook, 27
 interp1D, 28
 J0_zeros, 29
 maths_textbook, 28
 path_input_j0, 29
 pi, 29
 trapz, 28
XNLO::physics_textbook, 29
 c, 30
 E_at, 30
 eps_0, 30
 h_bar, 30
 k_B, 30
 l_at, 30
 m_at, 30
 mu_0, 30
 physics_textbook, 29
 q_at, 30
 t_at, 30
 w_at, 30
XNLO, 5
 XNLO, 5
xkx
 XNLO::Schrodinger_atom_1D, 33