

7-12-2014

Investigation of intracavity phase interferometry applied to nano-metrology

Xuan Luo

Follow this and additional works at: https://digitalrepository.unm.edu/ose_etds

Recommended Citation

Luo, Xuan. "Investigation of intracavity phase interferometry applied to nano-metrology." (2014). https://digitalrepository.unm.edu/ose_etds/22

This Dissertation is brought to you for free and open access by the Engineering ETDs at UNM Digital Repository. It has been accepted for inclusion in Optical Science and Engineering ETDs by an authorized administrator of UNM Digital Repository. For more information, please contact disc@unm.edu.

Candidate

Department

This dissertation is approved, and it is acceptable in quality and form for publication:

Approved by the Dissertation Committee:

_____, Chairperson

Investigation of intracavity phase interferometry applied to nano-metrology

by

Xuan Luo

B.S., Peking University, 2007

M.S., University of New Mexico, 2013

DISSERTATION

Submitted in Partial Fulfillment of the
Requirements for the Degree of

Doctor of Philosophy
Optical Science and Engineering

The University of New Mexico

Albuquerque, New Mexico

May, 2014

©2014, Xuan Luo

Dedication

To my family and friends.

Acknowledgments

I would like to thank all my committee members who were more than generous with their expertise and precious time in general. A special thanks goes to my committee chairman, Professor Jean-Claude Diels, for supporting me during these past six years. Dr. Diels is someone you will instantly find him a friend and never forget once you meet him. He is always energetic and one of the smartest people I know. It was my great pleasure to learn from him and work with him. I am also grateful to Professor Sudhakar Prasad, for his support and mentoring during my early years. He is the best resource for getting my theoretical questions answered and equations solved. And his enthusiasm and love for teaching is contagious. I owe thanks and appreciation to Dr. Ladan Arissian, for her continued support and inspiration with my research. She has been extremely patient to help proofreading and improve my writing. I also have to thank Professor Steven Brueck for kindly agreeing to serve on my committee in the last minute, and taking the effort to read this manuscript.

I will forever be thankful to all my colleagues at the Diels group for their help with my project, such as Dr. Daniel Mirrel, Dr. Andreas Velten, Dr. Alex Braga, Dr. Andreas Schmitt-Sody, Koji Masuda, Xiaozhen Xu, Chengyong Feng, Amin Rasoulof, Shermineh Rostami and Brian Kamer.

I would like to acknowledge and thank the Physics department and Center for High Technology materials for allowing me to conduct my research and providing any assistance requested. Special thanks goes to the staff members for their service and support.

Finally I want to thank my family and friends (too many to list here) for providing support and friendship that I needed throughout the process.

Investigation of intracavity phase interferometry applied to nano-metrology

by

Xuan Luo

B.S., Peking University, 2007

M.S., University of New Mexico, 2013

Ph.D., Optical Science and Engineering, University of New Mexico,
2014

Abstract

Intracavity phase interferometry is sensing technique developed at UNM, in which a physical quantity to be measured is put as integral part of a mode-locked laser. It relies on the fact that any intracavity phase shift of an intracavity pulse will result in a frequency change of the whole pulse train.

The implementations of IPI requires the operation of a mode-locked laser in which two pulses circulate independently, i.e. with no phase coupling between them. IPI has been demonstrated with a variety of laser systems, to detect either non-reciprocal effects (such as rotation, magnetic field), or phase changes that can be made periodic at the repetition rate of the laser cavity. The purpose of this work is to study the feasibility of applying this technique to the measurement of non-periodic (i.e. slow) changes in optical path. The new concept to measure sub-nanometer displacement uses an optoelectronic modulator (EOM) inside the cavity. The operation of the mode-locked laser after insertion of such an element in its cavity is analyzed.

Several laser systems have been tried for the implementation of IPI. Two of them are presented in this thesis. The first one is a Nd:YVO₄ laser, mode-locked by a multiple quantum wells (MQW) saturable absorber. The presence of a solid state saturable absorber introduced a dead band in the beat note response of the system. A new coupling between group and phase velocity was discovered experimentally, and explained through simulation. This coupling affects negatively the operation of the system, since the repetition rate is no longer a reliable fixed quantity. The coupling could be eliminated by replacing the MQW with a dye jet absorber. A first demonstration of a slow optical path change (in the nm range) was made.

The system that appeared at first the most promising is the intracavity optical parametric oscillator (OPO) synchronously pumped by a mode-locked Ti:Sapphire lasers. Bringing the unstable behavior of that laser under control proved considerably more difficult than anticipated, and led to an extensive theoretical analysis of the laser. The instabilities arise from both intensity and phase fluctuations in the OPO pulse train. We simulate the second order nonlinear interactions taking place inside the nonlinear crystal of the OPO, using a new approach in the frequency domain, valid down to a few optical cycles, and taking into account the dispersion of the crystal to all orders. Phase mismatched processes draw our attention as they introduce large effective nonlinear refractive indices (creating self-phase- and cross-phase-modulation) that result in a coupling of intensity and phase instabilities.

A full numerical model of coupled Ti:Sapphire and OPO cavities is established by parameterizing the gain, loss, dispersion and nonlinearities. The pulse evolution of both Ti:Sapphire and OPO is examined at each cavity round trip using the ABCD matrix method in temporal domain invented in this dissertation. The simulation reproduces the observed unstable operation. However, islands of stability are found. That is an operation observed to be stable against perturbation of any of the parameters.

Contents

| | |
|--|----------|
| List of Figures | xiii |
| List of Tables | xxviii |
| Glossary | xxix |
| 1 Introduction | 1 |
| 1.1 The definition of CEO, CEP and beat note | 2 |
| 1.2 IPI applied to sub-nm displacement | 4 |
| 1.2.1 IPI applied to periodic change in optical path length | 4 |
| 1.2.2 IPI applied to slow changes in optical path length | 5 |
| 1.3 Implementations of IPI | 6 |
| 2 Spatial and temporal stability study of pulse propagation in resonators | 8 |
| 2.1 ABCD matrix in spatial domain and beam propagation | 9 |
| 2.2 The concept of cavity oscillation and damping coefficient | 11 |

| | | |
|----------|--|-----------|
| 2.3 | ABCD matrix in temporal domain and pulse propagation | 14 |
| 2.4 | Experimental search for the damping coefficient | 18 |
| 2.4.1 | Attempt at direct measurement | 18 |
| 2.4.2 | Indirect determination through 2-level system analogy | 20 |
| 2.5 | Time versus space matrix | 21 |
| 3 | Optoelectronic modulator inside laser cavity | 22 |
| 3.1 | Mechanisms of phase and amplitude modulator | 23 |
| 3.2 | Phase modulator inside the cavity acting as a Fabry-Perot etalon | 24 |
| 3.3 | Simulation of Fabry-Perot effect of the EOM | 27 |
| 3.4 | Amplitude modulator combined with PBS and two arms | 30 |
| 3.5 | Amplitude and phase of the return pulse | 32 |
| 3.5.1 | Near a complete polarization rotation (φ close to π) | 35 |
| 3.5.2 | Small polarization rotation (φ close to zero) | 37 |
| 3.5.3 | Δ near π | 38 |
| 4 | Nd:YVO₄ laser for IPI | 39 |
| 4.1 | Experimental setup with and displacement measurement | 40 |
| 4.1.1 | The vanadate laser | 40 |
| 4.1.2 | Saturable absorber and generation of a pulse pair | 41 |
| 4.1.3 | Displacement measurement with a dye jet saturable absorber | 42 |

| | | |
|----------|---|-----------|
| 4.2 | Experimental setup with MQW and beat note measurement | 44 |
| 4.2.1 | Beat note measurements | 46 |
| 4.3 | Coupling of phase and group velocity through MQW | 46 |
| 4.3.1 | Experimental setup and results | 46 |
| 4.3.2 | Saturable absorption and group velocity | 48 |
| 4.4 | Conclusion | 52 |
| 5 | Intracavity pumped OPO for IPI | 53 |
| 5.1 | Experimental setups | 54 |
| 5.1.1 | OPO and pump cavities of equal length | 54 |
| 5.1.2 | OPO cavity of twice the length of the pump cavity | 59 |
| 5.2 | Alignment procedure | 63 |
| 5.3 | Instability issues, and attempts to eliminate them | 63 |
| 6 | Simulation of second order nonlinear interactions | 67 |
| 6.1 | Nonlinear equations in frequency domain | 68 |
| 6.1.1 | Summary of the theoretical model | 70 |
| 6.1.2 | Outline of the numerical approach | 70 |
| 6.1.3 | Quasi-Phase Matching | 73 |
| 6.1.4 | Comments on the simulation | 75 |
| 6.2 | Simulation results | 75 |

Contents

| | | |
|----------|---|------------|
| 6.2.1 | SHG only | 76 |
| 6.2.2 | SFG only | 85 |
| 6.2.3 | DFG only | 90 |
| 6.3 | Including all experimental conditions | 94 |
| 6.3.1 | Optimization of input pulse parameters | 95 |
| 6.3.2 | Pulse center shift after one passage | 96 |
| 6.3.3 | Wavelength shift at each passage | 101 |
| 6.3.4 | Near Phase-matched interactions | 104 |
| 6.3.5 | Phase mismatched interactions | 106 |
| 6.3.6 | Gain of OPO at different pump levels | 108 |
| 7 | Simulation of coupled Ti:Sapphire and OPO cavities | 110 |
| 7.1 | Presentation of the problems | 111 |
| 7.2 | Coupled cavities algorithm | 113 |
| 7.2.1 | Energy evolution | 114 |
| 7.2.2 | Chirp evolution | 117 |
| 7.3 | Simulation results | 117 |
| 7.3.1 | Damping effect | 120 |
| 7.3.2 | GVD of Ti:Sapphire and OPO cavities | 121 |
| 7.3.3 | Gain and loss parameters | 124 |
| 7.3.4 | Initial conditions | 126 |

Contents

| | | |
|----------|--|------------|
| 7.3.5 | Cavity detuning and wavelength tracking | 128 |
| 7.3.6 | Cavity length fluctuation study | 129 |
| 7.4 | Conclusion | 134 |
| 8 | Conclusion and future work | 135 |
| 8.1 | Conclusion | 135 |
| 8.2 | Future work | 136 |
| 8.2.1 | Fiber based IPI nanoscope | 136 |
| 8.2.2 | Frequency domain Optical Parametric Amplification | 138 |
| | References | 139 |
| A | Coupled NL wave equations derived in frequency domain | 143 |
| A.1 | Quasi-Phase Matching | 147 |
| B | Matlab code for coupled Ti:Sapphire and OPO cavities | 149 |
| C | Fortran code for second order nonlinear interactions | 157 |

List of Figures

| | | |
|-----|---|----|
| 1.1 | The phase modulator is inserted in a linear cavity in which two pulses circulate, represented here at each end of the cavity. Their crossing point is in the middle of the cavity. D is the beat note detector. D_1 is the detector that extract the laser signal to drive the phase modulator. The purpose of the phase modulator is to add a phase shift on only one of the two pulses. | 5 |
| 1.2 | The Pockels cell is inserted in a linear cavity in which two pulses circulate, represented here at each end of the cavity. The purpose of the Pockels cell is to deflect — totally or partially — the red pulse in the upper arm of the end-of-cavity interferometer. P is a polarizing beam splitter, D the beat note detector. This beat note changes linearly with the displacement ΔL of mirror REF | 6 |
| 2.1 | The beam waist (top) and the inverse of curvature of wavefront (bottom) is plotted as a function of round trips. The model is a typical Ti:Sapphire laser with wavelength at 795nm, beam waist of $211\mu\text{m}$, cavity length $d = 89\text{cm}$ and equivalent radius of curvature $R = 92.5\text{cm}$. The damping coefficient is $\epsilon = 0.01$ | 13 |

List of Figures

| | | |
|-----|---|----|
| 2.2 | Oscillation period in round trips as a function of the cavity parameters $R - d$ | 14 |
| 2.3 | The pulse width (top) and the chirp (bottom) is plotted as a function of round trips. The model is a typical Ti:Sapphire laser with wavelength at 770nm, pulse width about 100fs and pulse energy fixed around 27.5nJ. The Ti:Sapphire crystal is 3mm long with a Kerr coefficient of $10.5 \times 10^{-16} \text{ cm}^2/\text{W}$. The dispersion of the cavity is -800fs^2 . The damping coefficient is $\epsilon = 0.01$ | 17 |
| 2.4 | The sketch of an experimental setup to measure the damping coefficient of a Ti:Sapphire cavity. (a) The laser design with AOM inside the cavity. (b) A magnified look at the end of cavity, with a small pertubation to the system. | 19 |
| 3.1 | Sketch of a experimental setup to measure the phase delay introduced by an EOM inside a laser cavity. The EOM is driven by AC voltage at RF frequency of half the cavity repetition rate. Two pulses are combined through a delay line, the beat note is measured as a function of voltage amplitudes. | 25 |
| 3.2 | Beat frequency plotted as a function of voltage amplitudes. | 25 |
| 3.3 | Sketch of a experimental setup to measure the group delay introduced by the EOM inside a laser cavity. The EOM is driven by an increasing DC voltage (-1kV to 0V). The repetition rate is measured as a function of voltage. | 26 |
| 3.4 | Cavity round trip time plotted as a function of DC voltage. | 27 |
| 3.5 | transmission $ T $ (black), phase ϕ/π (red) and $d\phi/d\delta$ (blue) of a FP are plotted as a function of δ | 28 |

| | | |
|------|--|----|
| 3.6 | Simulation of the group delay versus voltage on the EOM, using $dn(V)/d\Omega = [3 \times 10^{-17} - 0.6 \times 10^{-22} \cdot V]$ s, the later value obtained from a fit to the experimental data of (Fig. 3.4). | 29 |
| 3.7 | Since there are two pulses circulating in the cavity, the detectors D_1 or D_2 (bottom part of the figure) record a signal at 180 MHz, which is divided by 2, phase adjusted, amplified before being sent to the EOM. Unless the two pulses returning to the polarizing beam splitter are exactly dephased by π (relative displacement of $\lambda/4$ of the two mirrors of the end-cavity interefometer) the “red pulse” will be attenuated as sketched in the graph “pulse intensity at modulator (180MHz)”. Because of the reduction of intensity of the “red pulses”, a component at 90 MHz appears in the spectrum analyzer recording of the pulse train on detector D_2 . That 90 MHz component is used in a feedback loop to maintain the bias of the phase modulator for the minimum loss condition, which corresponds to zero signal at 90 MHz. | 31 |
| 3.8 | The signal recorded by detector D_1 is now divided by 4. The “blue” pulse crosses the modulator at the null of the electric field. The “red” pulse crosses at the (+) and (-) peaks of the modulation. | 33 |
| 3.9 | The normalized intensity of the return pulse is plotted as a function of Δ for φ from 0 to π , with $\pi/8$ interval | 36 |
| 3.10 | The phase change of the return pulse is plotted as a function of Δ for $\varphi = \pi$ | 36 |
| 3.11 | The phase change of the return pulse is plotted as a function of Δ for $\varphi = 0.03\pi$ | 38 |

List of Figures

| | | |
|-----|--|----|
| 4.1 | The linear setup of a diode-pumped Nd:YVO ₄ laser. A saturable absorber (S.A.) is located in the middle of the linear cavity, with two identical lenses to focus the beam tightly in the saturable absorber. A pinhole is inserted after one of the lenses to assist passive mode-lock For beat note measurement, the EOM and PBS with two subcavities replace the other end of the cavity after the laser is prealigned. | 41 |
| 4.2 | Linear mode-locked Nd:YVO ₄ laser with dye jet in the middle of the cavity. A thin phase modulator is placed in the reference arm to create a beat note. | 43 |
| 4.3 | The beat note is measured as a function of displacement due to the applied DC voltage on the phase modulator. A total displacement of 250 nm has been measured. | 44 |
| 4.4 | The beat note is measured as a function of the peak to peak voltage on the phase modulator. The data deviated from a straight line that passes through origin indicates the deadband. | 45 |
| 4.5 | Linear mode-locked laser producing two intracavity pulses E_+ and E_- per cavity round-trip. G is the gain medium (Nd:YVO ₄), M the multiple quantum well absorber, and L are two AR-coated lenses for tight focusing of the beam in the MQW absorber at Brewster angle. The detector D is to monitor the repetition rate. Insets c) and d) illustrate two extreme cases where the nodes of the standing wave coincide with either the nodes of MQW, or the anti-nodes of MQW. which corresponds to minimum or maximum saturation respectively. These two conditions correspond to a different saturable absorption, hence a different impact on the group velocity of the pulses. | 47 |

List of Figures

| | | |
|-----|---|----|
| 4.6 | Comparison of the experimental data with simulation. The repetition rate of the laser is plotted as a function of cavity length shortening. Left: the departure in repetition rate from 90 MHz when the MQW is at the center. Right: the departure in repetition rate from 85 MHz, when the MQW is off center. | 49 |
| 4.7 | The repetition rate plotted as a function of the cavity shortening with a non-resonant MQW in the middle of the cavity. | 50 |
| 4.8 | (a) Linear mode-locked Ti:Sapphire laser with MQW in the middle of the cavity and an end mirror mounted on a piezoelectric ceramics (PZT). (b) Repetition rate of the laser recorded as a function of the laser cavity length. A deviation from the straight line is present with a modulation period of half the wavelength. | 51 |
| 5.1 | The sketch of an intracavity OPO pumped by a Ti:Sapphire with equal cavity length. The focal length of the lens for 532 nm is $f = 6$ cm, and the radius of curvature of the two curved mirrors is $R = 7.5$ cm. The beam is focused onto the MQW by a lens with focal length $f = 8$ cm. The radii of curvature of the two curved mirror at the end of Ti:Sapphire cavity are both $R = 10$ cm. The radii of curvature for both of the curved mirrors for OPO cavity are $R = 5$ cm. | 55 |
| 5.2 | The laser outputs taken from a 2% OC are plotted as a function of pump power for different pumping optics alignments. | 56 |
| 5.3 | OPO pulse train monitored on a digital oscilloscope. The capture time on each screen shot is $5 \mu s$. These two screen shots are taken within 2 minutes. (a) two pulse operation, (b) one pulse operation | 59 |

| | | |
|-----|--|----|
| 5.4 | The sketch of an intracavity OPO pumped by a Ti:Sapphire of half cavity length. The focal length of the lens for 532 nm is $f = 4$ cm, and the radii of curvature of the two curved mirrors is $R = 5$ cm. The pair of SF10 prisms is separated by 24 cm. A negative GVD mirror (-150 fs^2) is used as the end mirror. The laser beam is focused into the PPLN crystal by a lens with focal length $f = 3.6\text{cm}$. It is then focused into MQW by another lens with focal length $f = 3.6$ cm. The radii of curvature for both of the curved mirrors of the OPO cavity is $R = 5$ cm. | 60 |
| 5.5 | OPO and Ti:Sapphire pulse trains monitored on a digital oscilloscope. The yellow trace on the top is OPO, and the blue trace on the bottom is Ti:Sapphire. The capture time for the screen shot is $5 \mu\text{s}$. The intensity of the Ti:Sapphire is much higher so the two pulse trains are shown at different amplitude scales. | 61 |
| 5.6 | Figure (a) Shows an oscilloscope recording at 1 ms/division of (top) the RF square wave at 90 MHz , and (bottom) a 0.64kHz beat note recorded on the detector D . (b) Shows a plot of the beat note versus voltage applied to the phase modulator. | 62 |
| 5.7 | Double OPO pulse train monitored on a digital oscilloscope. The capture time for each screen shot is $5 \mu\text{s}$. (a) without SHG from LBO, (b) with SHG from LBO - the two pulse trains can hardly be distinguished. . . . | 64 |
| 5.8 | OPO pulse train and electrical signals monitored on a digital oscilloscope. Yellow trace is the OPO pulse train at 1200nm , green trace is the modulating voltage on the piezo, and blue trace is the signal obtained from the differential amplifier followed by DC amplifier. The piezo is modulated at different frequencies 11 Hz (a) and 22 Hz (b). | 66 |

List of Figures

| | | |
|-----|---|----|
| 6.1 | Diagram illustrated the different sum and difference frequency processes involved in the intracavity pumped parametric oscillator. Only the optical parametric process $\omega_p = \omega_s + \omega_i$ is phase matched. | 69 |
| 6.2 | Phase-matched SHG energy evolution of cw (solid line) and pulse (dashed line), red for pump and blue for SH. The peak intensity of the (fundamental) short pulse (100 fs) is the same as the intensity of the cw case, which is approximately 5×10^{13} W/m ² . The pulse energy is plotted as a function of distance (initial energy normalized to 1). The cw intensity is also plotted, normalized to the initial value of the fundamental. | 76 |
| 6.3 | Phase-mismatched SHG energy evolution of cw (solid line) and pulse (dashed line), red for pump and blue for SH. The two cases have same peak intensity and the energies are normalized. The period varies with different intensities for cw SHG. | 78 |
| 6.4 | Phase-mismatched SHG cw phase modulation. The pump phase (red, right scale) has a linear slope along propagation with some modulation, while the phase variation of the second harmonic (blue, left scale) is much steeper, but nearly perfectly linear. | 79 |
| 6.5 | Phase-mismatched SHG with continuous waves. Each data point is the electrical field of pump (red) or second harmonic (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagate through the crystal. | 81 |

List of Figures

| | | |
|------|--|----|
| 6.6 | Phase-mismatched SHG pulse propagation. The phase of the pump (red curve, right scale) and that of the second harmonic (blue curve, left scale), are plotted as a function of distance. The phase are defined as $\varphi_{avg} = \int \varphi(t)I(t)dt / \int I(t)dt$. The pump pulse (100 fs) has a peak intensity of about $5 \times 10^{13} \text{ W/m}^2$ | 81 |
| 6.7 | Phase-matched SHG with pulses. Each data point is the electrical field of pump (red) or second harmonic (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagrate through the crystal. | 82 |
| 6.8 | Phase-mismatched SHG with pulses Each data point is the electrical field of pump (red) or second harmonic (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagrate through the crystal. | 83 |
| 6.9 | Phase-mismatched evolution of the second harmonic in SHG. | 83 |
| 6.10 | Phase-mismatched evolution of the pump pulse intensity in SHG. | 84 |
| 6.11 | Phase-matched SFG energy evolution for cw (solid line) and pulse (dashed line) cases respectively. Initial intensities are $3 \times 10^{13} \text{ W/m}^2$ (pump) and $2 \times 10^{13} \text{ W/m}^2$ (signal) for both cw and pulse. Red stands for pump frequency, magenta for signal and green for sum. | 85 |
| 6.12 | Phase-matched SFG energy evolution for cw only. Initial intensities are $4.5 \times 10^{13} \text{ W/m}^2$ (pump) and $0.5 \times 10^{13} \text{ W/m}^2$ (signal). Red stands for pump frequency, magenta for signal and green for sum. As the signal wave vanishes the SFG turns into DFG for sum and pump frequencies. When the sum energy decreases to zero it returns to the initial condition of SFG and repeats these processes. | 86 |

| | | |
|------|---|----|
| 6.13 | Phase-mismatched SFG energy evolution for cw (solid line) and pulse (dashed line) cases respectively. Initial intensities are $3 \times 10^{13} \text{W/m}^2$ (pump) and $2 \times 10^{13} \text{W/m}^2$ (signal) for both cw and pulse. Red stands for pump frequency, magenta for signal and green for sum. | 87 |
| 6.14 | Phase-mismatched SFG for cw pump and sum. Each data point is the electrical field of pump (red) or sum (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagate through the crystal. | 87 |
| 6.15 | Phase-mismatched SFG pump and sum pulses. Each data point is the electrical field of pump (red) or sum (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagate through the crystal. | 88 |
| 6.16 | Phase-mismatched evolution of the pump pulse in SFG. | 88 |
| 6.17 | Phase-mismatched evolution of the signal pulse in SFG. | 89 |
| 6.18 | Phase-mismatched evolution of the sum pulse in SFG. | 89 |
| 6.19 | Phase-matched (dashed line) and quasi-phase-matched (solid line) DFG processes. Red stands for pump frequency, blue for signal and yellow for idler. The propagation distance is $100 \mu\text{m}$ and the PPLN grating period is $20.23 \mu\text{m}$ | 90 |
| 6.20 | Quasi-phase-mismatched DFG processes. Red stands for pump frequency, blue for signal and yellow for idler. The propagation distance is 1mm and the PPLN grating period is $22 \mu\text{m}$ | 91 |

List of Figures

| | | |
|------|---|----|
| 6.21 | Quasi-phase-matched DFG processes for pulses. Each data point is the electrical field of pump (red) or signal (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagate through the PPLN crystal which is 1 mm long and whose grating period is $22\mu\text{m}$. . . | 92 |
| 6.22 | Phase-mismatched evolution for pump pulse in DFG. | 93 |
| 6.23 | Phase-mismatched evolution for signal pulse in DFG. | 93 |
| 6.24 | Phase-mismatched evolution for idler pulse in DFG. | 94 |
| 6.25 | Transmission of pump and signal pulse energy after propagation through a 1mm PPLN for different initial pulse duration. Blue stands for pump frequency, red for signal. The pulse duration is varied from 50fs to 300fs, with 50fs interval. | 95 |
| 6.26 | Transmission of pump and signal pulse energy after propagation through a 1mm PPLN for different time delays between them. Red stands for pump frequency, blue for signal. The delay time is varied from -15fs to 270fs, in 15fs interval. | 96 |
| 6.27 | Pulse center shift for pump and signal pulse after propagation in a 1mm PPLN for different delay time. Red stands for pump frequency, blue for signal. The delay time is varied from -15fs to 270fs, with 15fs interval. . . | 97 |
| 6.28 | The group delay in a 1mm PPLN crystal for signal wavelength from $1.06\mu\text{m}$ to $1.26\mu\text{m}$ calculated from Sellmeier equation. | 98 |
| 6.29 | Pulse center shift for pump pulse after propagation in a 1mm PPLN for different signal wavelengths (1110 to 1170nm with 10nm interval) at different delay time (60, 90, 120, 150fs). | 99 |

List of Figures

| | | |
|------|--|-----|
| 6.30 | Pulse center shift for signal pulse after propagation in a 1mm PPLN for different signal wavelengths (1110 to 1170nm with 10nm interval) at different delay time(60, 90, 120, 150fs). | 99 |
| 6.31 | Relative pulse center shift between pump pulse and signal pulse after propagation in a 1mm PPLN for different signal wavelengths(1110 to 1170nm with 10nm interval) at different delay time(60, 90, 120, 150fs). | 100 |
| 6.32 | Wavelength shift for signal pulse after propagation in a 1mm PPLN for different delay time. The delay time is varied from -60 fs to 330 fs, with 30 fs interval. | 102 |
| 6.33 | Wavelength shift for signal pulse after propagation in a 1mm PPLN as a function of signal wavelength for different delay time 90fs (blue), 120fs (black) and 150fs (red). | 103 |
| 6.34 | Wavelength shift for signal pulse after propagation in a 1mm PPLN for different wavelength (1130 to 1150nm), delay (60 to 150fs) and chirp coefficient (0 to 3) | 103 |
| 6.35 | The transmission of pump (blue) and signal (red) pulses after propagation for signal wavelength from 1108nm to 1214nm | 104 |
| 6.36 | The transmission of pump (blue) and signal (red) as a function of signal wavelength for chirp coefficient 0(solid lines), 0.5(dashed lines) and 1(dash-dot lines). | 105 |
| 6.37 | Phase shift of pump (blue) and signal (red) as a function of the intensity of the signal. As is often the case, the cross-phase modulation is larger than the self-modulation. | 105 |
| 6.38 | Effective n_2 of pump (blue) and signal (red) as a function of signal wavelength. | 106 |

List of Figures

| | | |
|------|--|-----|
| 6.39 | Effective n_2 of pump (blue) and signal (red) as a function of signal wavelength for chirp coefficient 0 (solid lines), 0.5 (dashed lines) and 1 (dash-dot lines). | 107 |
| 6.40 | OPO gain as a function of signal pulse energy at three pump levels. . . . | 108 |
| 6.41 | Pump depletion as a function of signal pulse energy at three pump levels. | 109 |
| 7.1 | (a) Sketch of the intracavity pumped OPO. The OPO cavity is twice as long as the Ti:sapphire cavity. (b) top: signal applied to the modulator. Bottom: beat note signal. (c) Beat note recording versus signal on modulator. | 111 |
| 7.2 | Simulation of the intracavity pumped OPO. | 114 |
| 7.3 | The parameters we choose to run the simulation for coupled Ti:Sapphire and OPO cavities are listed in each column. The red indicate the stable range with quasi-equilibrium solutions. | 119 |
| 7.4 | GVD=-1600fs ² (pump), 1000fs ² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.15-0.5× E_s ; Pump depletion=0.91-0.1× E_s ; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm | 120 |
| 7.5 | GVD=-1700fs ² (pump), 1000fs ² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.15-0.5× E_s ; Pump depletion=0.91-0.1× E_s ; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm | 121 |
| 7.6 | GVD=-1600fs ² (pump), 900fs ² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.15-0.5× E_s ; Pump depletion=0.91-0.1× E_s ; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm | 122 |

List of Figures

- 7.7 GVD=780fs² (pump), -500fs² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.15-0.5×E_s; Pump depletion=0.91-0.1×E_s; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm 123
- 7.8 GVD=-1600fs² (pump), 1000fs² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.16-0.5×E_s; Pump depletion=0.91-0.1×E_s; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm 124
- 7.9 GVD=-1600fs² (pump), 1000fs² (signal); Linear loss=10% (pump), 7% (signal); OPO gain=1.15-0.5×E_s; Pump depletion=0.9-0.1×E_s; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm 125
- 7.10 GVD=-1600fs² (pump), 1000fs² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.15-0.5×E_s; Pump depletion=0.91-0.1×E_s; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1136nm 126
- 7.11 GVD=-1600fs² (pump), 1000fs² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.15-0.5×E_s; Pump depletion=0.91-0.1×E_s; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm. Note: Pump energy starts from 80% as before and signal energy starts from 180%. . . 127
- 7.12 GVD=-1600fs² (pump), 1000fs² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.15-0.5×E_s; Pump depletion=0.91-0.1×E_s; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm. Note: A sudden change of -0.5fs is applied to the cavity round trip time. 128
- 7.13 GVD=-1600fs² (pump), 1000fs² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.15-0.5×E_s; Pump depletion=0.91-0.1×E_s; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm. Note: A sudden change of 1.7fs is applied to the cavity round trip time. 129

List of Figures

- 7.14 The blue solid line is a set of normally distributed pseudorandom numbers plotted as a function of round trip time. The noise amplitude (in fs) is proportional to round trip counts, and frequency is 25 MHz. The red dotted line is the square of the noise. 130
- 7.15 GVD=-1600fs² (pump), 1000fs² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.15-0.5× E_s ; Pump depletion=0.91-0.1× E_s ; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm. Note:Cavity round trip time fluctuation (Fig. 7.14) is added to the cavity round trip time. 130
- 7.16 The blue solid line is a set of normally distributed pseudorandom numbers plotted as a function of round trip time. The noise amplitude (in fs) is proportional to round trip counts, and frequency is 2.5 MHz. The red dotted line is the square of the noise. 131
- 7.17 GVD=-1600fs² (pump), 1000fs² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.15-0.5× E_s ; Pump depletion=0.91-0.1× E_s ; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm. Note: Cavity round trip time fluctuation (Fig. 7.16) is added to the cavity round trip time. 131
- 7.18 The blue solid line is a set of normally distributed pseudorandom numbers plotted as a function of round trip time. The noise amplitude (in fs) is proportional to round trip counts, and frequency is 0.25 MHz. The red dotted line is the square of the noise. 132

List of Figures

- 7.19 GVD=-1600fs² (pump), 1000fs² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.15-0.5× E_s ; Pump depletion=0.91-0.1× E_s ; NL loss coeff=1.5e6; Damping coeff=0; Initial λ =1140nm. Note:Cavity round trip time fluctuation (Fig. 7.18) is added to the cavity round trip time. 133
- 8.1 Sketch of the fiber laser sensor. Two pulses are made to oscillate in a cavity made of polarization maintaining (PM) fibers, one propagating on the slow axis, and the other on the fast axis. A X indicates the crossing point of the two pulses, where a carbon nanotube (CNT) saturable absorber is located both for mode-locking and for imposing the pulse crossing point. At the right end of the cavity, a polarizing beam splitter P separates the two pulses, one being sent to a reference arm, the other to the arm with the sensing element. On the left side of the cavity, output couplers and beam combiners make the two output pulse trains interfere on a detector D137

List of Tables

| | | |
|-----|---|-----|
| 7.1 | Gain/loss parameters for the Ti:sapphire laser. | 116 |
|-----|---|-----|

Glossary

| | |
|-------------|------------------------------------|
| <i>CEO</i> | Carrier to envelope offset |
| <i>CEP</i> | Carrier to envelope phase |
| <i>AOM</i> | Acousto-optic modulator |
| <i>EOM</i> | Electro-optic modulator |
| <i>NL</i> | Nonlinear |
| <i>PPLN</i> | Periodically poled lithium niobate |
| <i>SHG</i> | Second harmonic generation |
| <i>SFG</i> | Sum frequency generation |
| <i>DFG</i> | Difference frequency generation |
| <i>PM</i> | Phase matching |
| <i>QPM</i> | Quasi-phase matching |
| τ_{RT} | Cavity round trip time |
| ν_{RT} | Cavity repetition rate |
| ν | Optical frequency |

Glossary

| | |
|------------------|--|
| λ | Wavelength |
| L | Cavity length |
| ΔL | Displacement |
| v_g | Group velocity |
| v_p | Phase velocity |
| $\Delta\varphi$ | Relative phase shift |
| $\Delta\nu$ | Beat note frequency |
| n | Refractive index |
| n_2 | Nonlinear refractive index |
| $\chi^{(2)}$ | Second order nonlinear susceptibility |
| k | Wave vector |
| I | Intensity |
| \tilde{E} | Complex electric field |
| \tilde{P}^{NL} | Nonlinear polarization |
| τ_G | Gaussian pulse width |
| τ_p | FWHM of a pulse |
| ω_0 | Angular center frequency of a pulse |
| Ω | Angular frequency variable in the Fourier domain |

Chapter 1

Introduction

Since the invention of the laser and realization of coherent light, there have been various ways of tailoring laser sources for precision measurements. Intracavity Phase Interferometry (IPI) is realized as a synthesis of “frequency combs”, “Interferometry” and “resonator based laser sensors”. In the case of a continuous laser, utmost stretching of the coherence time leads to a precisely defined single frequency. Comparable precision is achieved through the mode-locking mechanism, creating a train of equally spaced pulses, of which the spectrum is a frequency comb where each tooth is defined with the same precision as that of the cw laser. The group velocity of the intracavity pulse defines the tooth spacing, while the phase velocity determines the absolute position of this comb. The first tooth of the extended comb with respect to the origin is called “carrier to envelope offset (CEO)”. Beating the frequency comb with an unknown frequency is a way of seeking precision measurement in the frequency domain. This work is about beating two entire frequency combs, of the same tooth spacing but different CEO.

In most interferometric measurements, such as performed with a Michelson interferometer, a phase change in the optical paths is observed as an amplitude modulation

in space or time. The sensitivity to a phase perturbation is greater in a resonator. In a Fabry-Perot for instance, a physical quantity that changes the phase of a resonator, is monitored with a change of its transmission. The phase perturbation is repeated by the average number of round-trips in the cavity. The (Q-times) (Q being the quality factor) repeated phase perturbation implies improved sensitivity. The linewidth of a Fabry-Perot can be decreased by inserting gain in the cavity. Interferometry in an active resonator (laser) makes the best of two worlds: exploiting the ultimate Q-factor and converting phase into a frequency rather than an amplitude measurement.

Many physical quantities can be imprinted onto the intracavity light wave. Sub-nanometer displacement, nonlinear index, rotation of earth, magnetic and electric field or anything that changes phase velocity produces a carrier to envelope phase (CEP) shift $\Delta\varphi_{CEP}$, perceived as a relative displacement of the frequency combs or shift in CEO. Two pulse trains of the same repetition rate are generated by a laser in which two intracavity pulses circulate. Any minute relative phase change ($\geq 10^{-10}$ radian) between the two intracavity pulses is converted into a frequency shift between the two pulse trains, measured by beating the two pulse trains emitted by the laser. Interferometry with ultrashort pulses compared to a continuous wave interferometry has the advantage that there is no dead band in the measurement, because the pulses interact only at their crossing point.

1.1 The definition of CEO, CEP and beat note

In a perfect IPI system the physical quantity that changes the optical frequency between the two combs has no effect on the repetition rate. In this ideal case the beat between the two combs is actually the beat between the two CEOs. The CEO

is defined as the change in CEP divided by the pulse period:

$$f_0 = \frac{\Delta\varphi_{CEP}}{2\pi} \cdot \nu_{RT}, \quad (1.1)$$

where f_0 is the CEO and ν_{RT} is the cavity repetition rate.

The latter (CEP) is often defined as the value of the phase of the carrier at the location of the pulse peak. Such a definition however relies on a somewhat arbitrary choice of carrier frequency. A better definition is based on a complex description of the electric field:

$$E(t) = \frac{1}{2}\tilde{E}(t) + c.c. = \frac{1}{2}\mathcal{E}(t)e^{i\varphi(t)} + c.c. \quad (1.2)$$

where $\frac{1}{2}\tilde{E}(t)$ is the Fourier transform of the positive frequency part of the Fourier spectrum of the pulse. The new definition of the CEP is the difference of the phase $\varphi(t_1)$ of the complex field taken at the peak of the amplitude $\mathcal{E}(t_1)$, and the phase $\varphi(t_2)$ taken at the peak of the real field $E(t_2)$ [1, 11]. The time t_1 corresponds to the peak of the envelope, and t_2 to a carrier peak. In a laser cavity, the resonance condition imposed by the cavity length (and the phase velocity) is linked to the time t_1 , while the ratio of the cavity length to the group velocity should set the time t_2 .

IPI measurement is based on the fact that any physical phenomena that changes the CEP in the cavity will modify the optical frequency. There are however many things in the cavity that can affect the optical frequency, such as the gain dynamics and optical elements. The key to this design is to have two pulse trains in the cavity that share almost all the optical elements with only one exception; one pulse train is affected by a “sample”, while the other is not. As a result one pulse train will have a slightly different optical frequency than the other. The relative change in phase $\Delta\varphi$ between two intracavity pulses per round-trip is measured as a beat note

$$\Delta\nu = \Delta\varphi/(2\pi\tau_{RT}), \quad (1.3)$$

where $\tau_{RT} = 1/\nu_{RT}$ is the cavity round trip time. $\Delta\nu$ is measured when the pulse trains corresponding to the two pulses are made to interfere outside the laser cavity.

1.2 IPI applied to sub-nm displacement

IPI has the potential of sensing displacements down to $10^{-8}\lambda$. The challenge addressed in this dissertation is to find a practical, all solid state implementation. Slow length shortening or elongation, rather than periodic motion in repetition rate frequency, is one of the most promising — and challenging — aspect of IPI. We mainly attempt analyze and solve the difficult problems encountered in this task. The distinction between the two types of application of IPI, either to sense periodic or slow change in optical path length, is made through Sections 1.2.1 and 1.2.2.

1.2.1 IPI applied to periodic change in optical path length

The first requirement of IPI is to create two pulse trains inside a laser cavity (for example the pulse crossing is established by a saturable absorber element). The next requirement is to create a phase shift on one of the pulse trains with respect to the other. One way to do this is to insert a phase modulator inside the cavity and drive it at half the laser repetition frequency. A typical configuration is illustrated in Fig. 1.1 This is a classical “colliding pulse mode-locking” configuration which results in two intracavity pulses meeting exactly in the middle of the cavity. The purpose of the phase modulator is to add a phase shift on only one of the two pulses. By overlapping the two pulses through a delay line a beat note $\Delta\nu$ is measured:

$$\frac{\Delta\nu}{\nu_{RT}} = \frac{\Delta\varphi}{2\pi} = \frac{\Delta(nL)}{\lambda}, \quad (1.4)$$

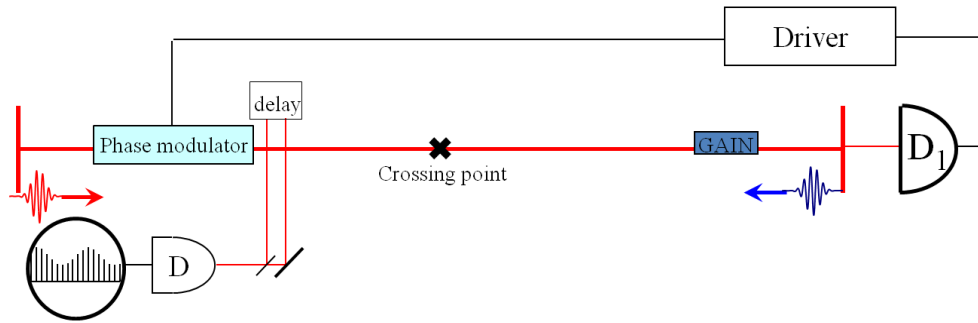


Figure 1.1: The phase modulator is inserted in a linear cavity in which two pulses circulate, represented here at each end of the cavity. Their crossing point is in the middle of the cavity. D is the beat note detector. D_1 is the detector that extract the laser signal to drive the phase modulator. The purpose of the phase modulator is to add a phase shift on only one of the two pulses.

where λ is the laser wavelength, n is the refractive index and $\Delta(nL)$ is the change in optical path length.

1.2.2 IPI applied to slow changes in optical path length

IPI is a potential candidate for a nanoscope combining high spatial and temporal resolutions that are desirable for biomedical imaging. For the precise measurement of length, the two pulses are going through the same cavity and are only separated at the end, where they are diverted into two arms of sample and reference. This design allows us to measure slow changes in the optical path length, which is generated in a sample inserted in one of the arms. In this context the two arms are called sample and reference.

The general laser configuration is sketched in Fig. 1.2. At every round trip, one of these two pulses is deflected to the subcavity terminated by the sample reflector by the combination of electro-optic modulator and polarizing beam splitter (P), while the other pulse circulates in a straight cavity, terminated by the reference mirror

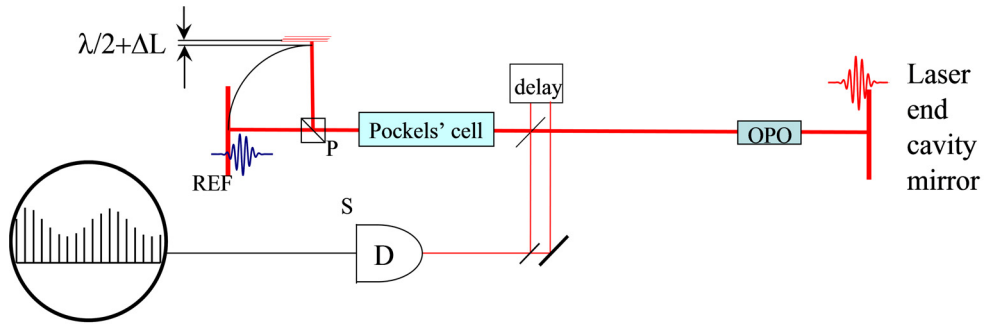


Figure 1.2: The Pockels cell is inserted in a linear cavity in which two pulses circulate, represented here at each end of the cavity. The purpose of the Pockels cell is to deflect — totally or partially — the red pulse in the upper arm of the end-of-cavity interferometer. P is a polarizing beam splitter, D the beat note detector. This beat note changes linearly with the displacement ΔL of mirror REF .

(REF). The signal recorded on the detector D_1 shown in Fig. 1.2 is divided by 2 or by 4 to drive a phase locked loop, and applied after amplification and phase adjustment to the modulator.

1.3 Implementations of IPI

The implementations of IPI requires two pulses circulating inside a laser cavity. It is required that the corresponding two trains extracted from the cavity be independent and that no coupling mechanism lock their frequencies together. Although this may seem an impossible task, it has been realized in various laser systems. A Ti:Sapphire laser with dye jet saturable absorber has been used to measure electro-optical coefficients [6] and magnetic fields [32]. The highest resolution of 0.17Hz was achieved with a synchronously pumped OPO [34].

This dissertation is dedicated to the implementation of IPI for the measurement of nanometer displacements. The simulation of pulse evolution in the laser has been facilitated by a new ABCD temporal matrix which is introduced for the first time

in Chapter 2. The study of Electro Optic Modulators (EOM) as the key mechanism for separating and controlling the two pulses in the cavity is discussed in Chapter 3. A successful demonstration of nanometer resolution through IPI for length measurements is described in Chapter 4 with a Nd:YVO₄ laser. This approach had however to be abandoned because of complex coupling of phase and group velocity analyzed in that Chapter. Chapter 5 is devoted to challenges (performance and instability) of IPI on an intracavity pumped optical parametric oscillator (OPO) platform. Simulation of the second order nonlinear interactions for the Ti:Sapphire laser pumped OPO is in Chapter 6. In Chapter 7 the modeling of the coupled Ti:Sapphire and OPO cavity is presented. The model leads to an understanding of sources of amplitude and phase instability in this system. In the large parameter space of the intracavity pumped OPO, an island of stability has been identified.

Chapter 2

Spatial and temporal stability study of pulse propagation in resonators

In this chapter we will introduce a novel cavity matrix method to investigate pulse propagation in the temporal domain. The pulse evolution of a mode-locked laser is important in many respects, and particularly interests us because we care about small phase shifts that may affect the intracavity pulse at each round trip. In analogy with the ABCD matrix for Gaussian beam propagation, we develop an elegant description for Gaussian pulse propagation in the temporal domain. Simple yet effective, we will later use this approach in Chapter 7 to model a complex coupled cavity system that we built for an IPI application.

We begin with a review of the ABCD matrix in the spatial domain, as it applies to beam propagation in section 2.1. We proceed to the analysis of laser cavities by ABCD matrices in section 2.2. We note that the generally accepted “stability cri-

terium” does not match our general understanding of a stable equilibrium, by which a departure from the equilibrium is followed by an evolution converging towards the stable solution.

If the ABCD matrix is applied multiple times to the beam parameters that correspond to the *exact* cavity solution, the beam waist and radii of curvatures are unchanged at each round-trip. However, given an initial slight departure from the exact cavity solution, the beam parameters will oscillate without any damping with the round-trip index. This does not correspond to the reality of a laser cavity, where we know that the Gaussian beam parameters converge towards a stable solution, rather than oscillate indefinitely. In order to mimic a stable resonator, we introduce a “damping parameter” into the ABCD formalism.

Two experimental attempts to find the value of the damping parameter in a Ti:sapphire laser are presented in section 2.4. We exploit next the space-time analogy to create an ABCD formalism in the time domain, and carefully examine the pulse propagation with chirp in section 2.3. The Chapter concludes with a discussion on the difference in applying of the ABCD formalism in the temporal and spatial domains in section 2.5.

2.1 ABCD matrix in spatial domain and beam propagation

An ABCD matrix [19] is a ray transfer matrix which describes the effect of an optical element on a laser beam. It can be used both in geometrical optics and for propagating Gaussian beams. The paraxial approximation is always required for ABCD

matrix calculations. Tracing of a light path through an optical system can then be performed by multiplying this matrix with a vector representing the light ray:

$$\begin{pmatrix} y_2 \\ \alpha_2 \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \cdot \begin{pmatrix} y_1 \\ \alpha_1 \end{pmatrix} \quad (2.1)$$

where y and α refer to transverse displacement and offset angle from an optical axis respectively. The subscripts ‘1’ and ‘2’ denote the coordinates before and after an optical element. For example, a thin lens with focal length f has the following ABCD matrix:

$$\begin{pmatrix} 1 & 0 \\ -\frac{1}{f} & 1 \end{pmatrix}, \quad (2.2)$$

and propagation through free space over a distance d is associated with the matrix:

$$\begin{pmatrix} 1 & d \\ 0 & 1 \end{pmatrix} \quad (2.3)$$

A convenient quantity, labeled the q parameter, has been defined for Gaussian beams. It concatenates the information on the beam radius w and the radius of curvature R in a single complex quantity defined by:

$$\frac{1}{q} = \frac{1}{R} - i \frac{\lambda}{\pi w^2} \quad (2.4)$$

It has been observed that the modification of the q parameter by an optical element can be expressed in terms of the elements of the $ABCD$ matrix:

$$q_2 = \frac{Aq_1 + B}{Cq_1 + D} \quad (2.5)$$

or the equivalent reciprocal form:

$$\frac{1}{q_2} = \frac{C + D/q_1}{A + B/q_1} \quad (2.6)$$

where q_1 and q_2 represent the value of the q parameter before and after the optical element, respectively. Equation (2.5) is often represented in the form:

$$\begin{pmatrix} q_2 \\ 1 \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} q_1 \\ 1 \end{pmatrix}. \quad (2.7)$$

Please note that, for equivalence with Eq. (2.5), a re-normalization of the ‘ q ’ vector is needed after the matrix multiplication.

2.2 The concept of cavity oscillation and damping coefficient

For a cavity characterized by an ABCD matrix, the evolution of the ‘ q ’ parameter over N round trips is given by:

$$\begin{pmatrix} q_N \\ 1 \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \cdot \begin{pmatrix} q_{N-1} \\ 1 \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}^{N-1} \cdot \begin{pmatrix} q_1 \\ 1 \end{pmatrix} \quad (2.8)$$

It can be shown [19] that, for the beam to be trapped in the cavity, there is a stability condition: $-1 \leq \frac{1}{2}(A + D) \leq 1$.

Equation (2.8) is reminiscent of the equation of rotation of a vector E , characterized by the equation:

$$\frac{\partial E}{\partial \theta} = iE \quad (2.9)$$

which has a solution $E = E_0 e^{i\theta}$. The differential equation (2.9) can also be represented by the successive products:

$$E_N = [1 + i\Delta\theta]E_{N-1} = [1 + i\Delta\theta]^N E_0. \quad (2.10)$$

If \mathcal{M} represents the ABCD matrix, $|U|$ the unit matrix, q_s the steady state solution of the column vector q , and Δq a departure from equilibrium:

$$\mathcal{M}(q_s + \Delta q) = |U|q_s + \mathcal{M}\Delta q = |U|q_s + i|U|\Delta\theta\Delta q \quad (2.11)$$

expressing that the Δq is a periodic solution around the steady state. In the real physical world, given a stable cavity, we know that a perturbation Δq will decay to zero. In order to have a damped solution, one should add an imaginary part to $\Delta\theta \rightarrow \Delta\theta + i\epsilon$. Thus replacing the matrix \mathcal{E} by

$$\mathcal{E} \rightarrow \mathcal{E} - i|U|\epsilon \quad (2.12)$$

should lead to the desired damping. In a real laser, the stability results from the balance of gain and losses. The rays are replenished at a rate determined by the lifetime of a photon in the empty resonator. One can thus assimilate a damping of Δq with the loss factor of the cavity. As a simple model of this effect, we associate a phenomenological damping coefficient ϵ to the ABCD matrix, by replacing \mathcal{M} by $\mathcal{M} - \epsilon$.

Let us consider here the simplest cavity consisting of a flat mirror and a mirror of curvature R at a distance d . The ABCD matrix of the cavity, starting from the flat mirror, is:

$$\begin{pmatrix} 1 - \frac{2d}{R} & 2d - \frac{-2d^2}{R} \\ \frac{-2}{R} & 1 - \frac{2d}{R} \end{pmatrix} = \begin{pmatrix} 1 & d \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ \frac{-2}{R} & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & d \\ 0 & 1 \end{pmatrix} \quad (2.13)$$

Since we start at the flat mirror, that is where the *waist* is located, implying that the radius of curvature is infinite, and the rays inclination zero. The beam waist at the flat mirror is:

$$w_0^2 = \frac{\lambda}{\pi} \sqrt{d(R-d)} \quad (2.14)$$

The modified cavity matrix is:

$$\begin{pmatrix} 1 - \frac{2d}{R} & 2d - \frac{2d^2}{R} \\ \frac{-2}{R} & 1 - \frac{2d}{R} \end{pmatrix} + \begin{pmatrix} -i\epsilon & 0 \\ 0 & -i\epsilon \end{pmatrix}. \quad (2.15)$$

Whenever a parameter — be it w or the curvature R — departs from equilibrium, it returns to equilibrium with a time constant equal to a fixed number of oscillations, for a given ϵ . If $\epsilon = 0$, the departure from equilibrium oscillates forever without any damping. Setting $\epsilon = 0.01$, a damping is clearly added to the oscillations (Fig. 2.1).

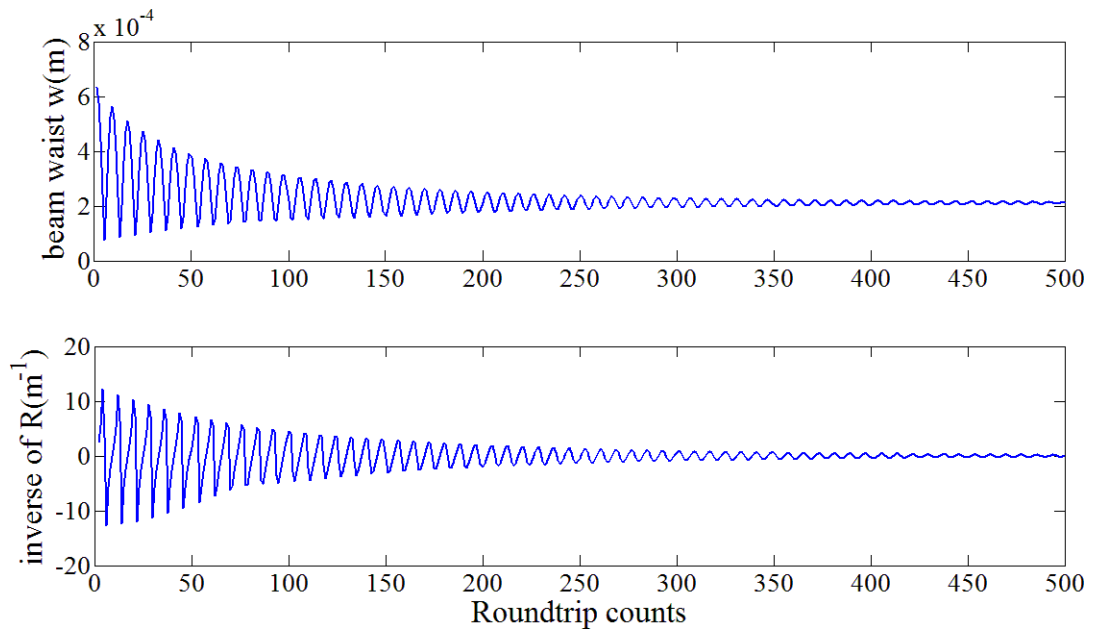


Figure 2.1: The beam waist (top) and the inverse of curvature of wavefront (bottom) is plotted as a function of round trips. The model is a typical Ti:Sapphire laser with wavelength at 795nm, beam waist of $211\mu\text{m}$, cavity length $d = 89\text{cm}$ and equivalent radius of curvature $R = 92.5\text{cm}$. The damping coefficient is $\epsilon = 0.01$.

The frequency of oscillation depends on the type of resonator: it is a low frequency for a cavity close to the flat-flat stability limit. The oscillation period is plotted in Fig. 2.2 as a function of the cavity parameters [in this particular example $(R - d)$].

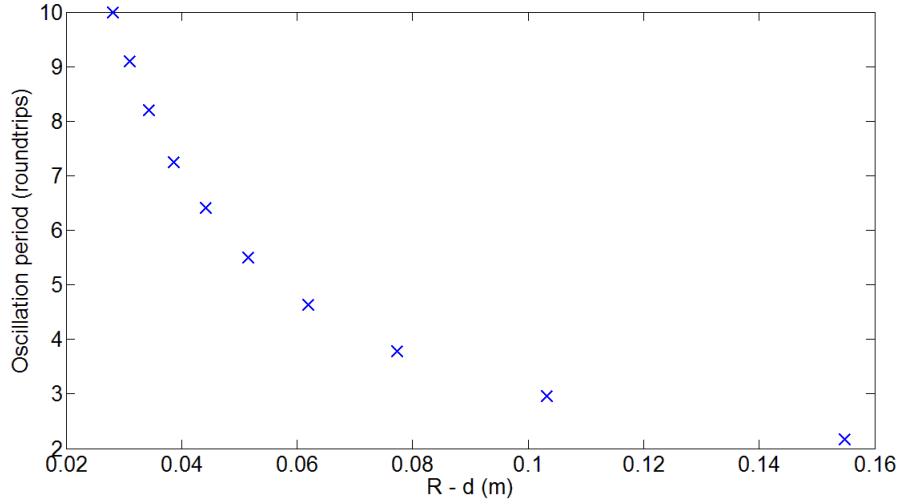


Figure 2.2: Oscillation period in round trips as a function of the cavity parameters $R - d$.

2.3 ABCD matrix in temporal domain and pulse propagation

Space-time duality and the theory of temporal imaging has been developed by [20, 3]. This duality leads naturally to the conclusion that a quadratic phase modulation in time is the analog of a thin lens in space, and dispersion in time the equivalent of propagation (diffraction) in space. Therefore, by a suitable combination of dispersion and quadratic phase modulation (now a “time lens”), we can synthesize the time-domain analog of an imaging system. We establish next the analog of the ABCD matrices in the time domain, which we will apply to the study of mode-locked laser cavities.

A laser cavity in temporal domain can be treated as a simplified combination of Kerr lens and dispersion compensation. If we define the time equivalent of the q

parameter:

$$\frac{1}{p} = \ddot{\varphi} - \frac{2i}{\tau_G^2}, \quad (2.16)$$

where $\ddot{\varphi} = \frac{\partial^2 \varphi}{\partial t^2}$ is the second derivative of the phase in the middle of the pulse, and τ_G [remembering that $\tau_p = \sqrt{2 \ln 2} \tau_G$] is the Gaussian pulse width. With this definition, the matrix for dispersion is:

$$\begin{pmatrix} 1 & k'' \ell \\ 0 & 1 \end{pmatrix}, \quad (2.17)$$

where $k'' = \frac{\partial^2 k}{\partial \Omega^2}$ is the second derivative of wave vector k . If we apply

$$\frac{1}{p} = \frac{C + \frac{D}{p_0}}{A + \frac{B}{p_0}}$$

to Eq. (2.16), using a un-chirped initial $1/p_0 = -2i/\tau_{G0}^2$ we find:

$$\frac{1}{p} = \frac{-i}{\tau_{G0}^2 \left(1 + \frac{4k'' \ell^2}{\tau_{G0}^4}\right)} + \frac{4k'' \ell}{\tau_{G0}^4 \left(1 + \frac{4k'' \ell^2}{\tau_{G0}^4}\right)} \quad (2.18)$$

With the factor 2 in the numerator of the imaginary part of p , the first term obviously shows the correct expression for the pulse broadening:

$$\tau_G = \tau_{G0} \sqrt{1 + \frac{4k'' \ell^2}{\tau_{G0}^4}} \quad (2.19)$$

The second term gives also the correct chirp:

$$\ddot{\varphi}_{disp} = \frac{4k'' \ell}{\tau_{G0}^4 \left(1 + \frac{4k'' \ell^2}{\tau_{G0}^4}\right)} \quad (2.20)$$

For the chirp induced by some modulation, such as the Kerr effect, we have to use the matrix:

$$\begin{pmatrix} 1 & 0 \\ chirp & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \frac{2\pi \ell_{Kerr} n_2 I}{\lambda} & 1 \end{pmatrix} \quad (2.21)$$

where the second part is the Kerr modulation. When applied to the same initial condition as above, we find indeed that the application of the matrix give an unchanged pulse width, and a chirp

$$\ddot{\varphi} = \frac{2\pi\ell_{Kerr}}{\lambda} n_2 \frac{I}{\tau_G^2}.$$

In Eq. (2.16), the time correspondent of the focal length $f \rightarrow -1/\ddot{\varphi}$, and the time equivalent of the Rayleigh range is $\rho_0 \rightarrow \tau_{G0}^2/2$. Applying the correspondence to Eq. (2.14), we find that the minimum pulse duration in that cavity is given by:

$$\frac{\tau_{G0}^2}{2} = \sqrt{k''\ell\left(-\frac{2}{\ddot{\varphi}} - k''\ell\right)}. \quad (2.22)$$

The time equivalent of the “flat mirror – curved mirror” cavity, is one that starts from a bandwidth limited pulse at one end, propagates through the (negative) dispersion of the cavity, goes through a Kerr self phase modulation, then dispersion again to the starting point for a round trip:

$$\begin{pmatrix} 1 + k''\ell\ddot{\varphi} & k''\ell(2 + k''\ell\ddot{\varphi}) \\ \ddot{\varphi} & 1 + k''\ell\ddot{\varphi} \end{pmatrix} \quad (2.23)$$

The result of Eq. (2.14) can be derived from the ABCD matrix, or by simply identifying the wavefront curvature with that of the mirror:

$$\frac{1}{R} = \frac{d}{d^2 + \rho_0^2} \quad (2.24)$$

The time equivalent equation is:

$$-\frac{\ddot{\varphi}}{2} = \frac{k''\ell}{(k''\ell)^2 + \left(\frac{\tau_{G0}^2}{2}\right)^2}. \quad (2.25)$$

where $\ddot{\varphi}$ has to be the applied modulation.

In Fig. 2.3 we plot the chirp and pulse duration oscillations as a function of roundtrips. Both parameters are seen to converge to the steady state solutions Eqs (2.25) and (2.22).

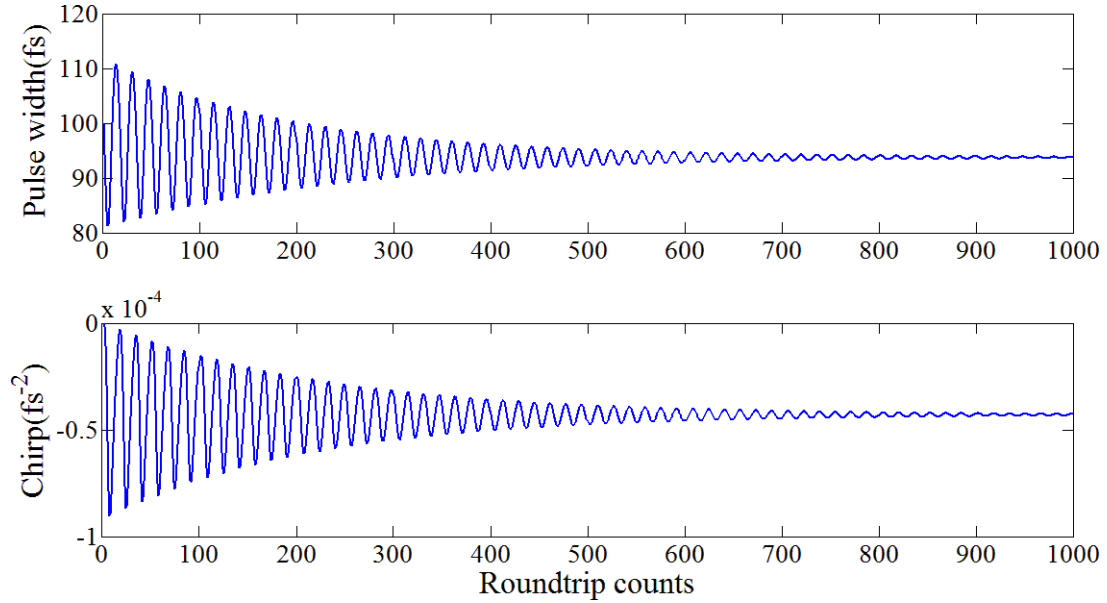


Figure 2.3: The pulse width (top) and the chirp (bottom) is plotted as a function of round trips. The model is a typical Ti:Sapphire laser with wavelength at 770nm, pulse width about 100fs and pulse energy fixed around 27.5nJ. The Ti:Sapphire crystal is 3mm long with a Kerr coefficient of $10.5 \times 10^{-16} \text{ cm}^2/\text{W}$. The dispersion of the cavity is -800fs^2 . The damping coefficient is $\epsilon = 0.01$.

2.4 Experimental search for the damping coefficient

2.4.1 Attempt at direct measurement

Two experimental designs are implemented to measure the damping coefficient. The deviation of beam parameters from equilibrium is introduced by an acousto-optic modulator (AOM), which is much faster than a typical mechanical devices such as a galvanometric mirror. The two approaches are as follows:

1. A cw Ti:Sapphire laser is built to measure the damping coefficient for ray displacement-‘y’ parameter. As shown in Fig. 2.4, an acousto-optic modulator (AOM) is inserted near the end mirror of the cavity. When there is voltage applied on the AOM, the laser beam is deflected. The deflection angle θ is proportional to the driving frequency of the applied voltage. The laser is pre-aligned for the first order of deflection, and then the driving frequency is varied by a square wave to create a small shift from the equilibrium. The laser then lases between these two modes. The measurements are taken by a photomultiplier tube (PMT) a certain distance away from the end mirror so that the two modes are well separated. This experimental setup is sketched in Fig. 2.4
2. A laser beam is sent through a Fabry-Perot (extracavity). The input angle is shifted by an AOM. The output angle is examined as a function of time. It is an outside cavity experiment as opposed to the previous one.

Both approaches were attempted, using a Ti:Sapphire laser. In the first approach of Fig. 2.4, two mode lasing was successfully achieved with the AOM. When the driving frequency of AOM is varied by a square wave, a small shift from the equilibrium is created and we expect to see an intensity oscillation passing through the pinhole.

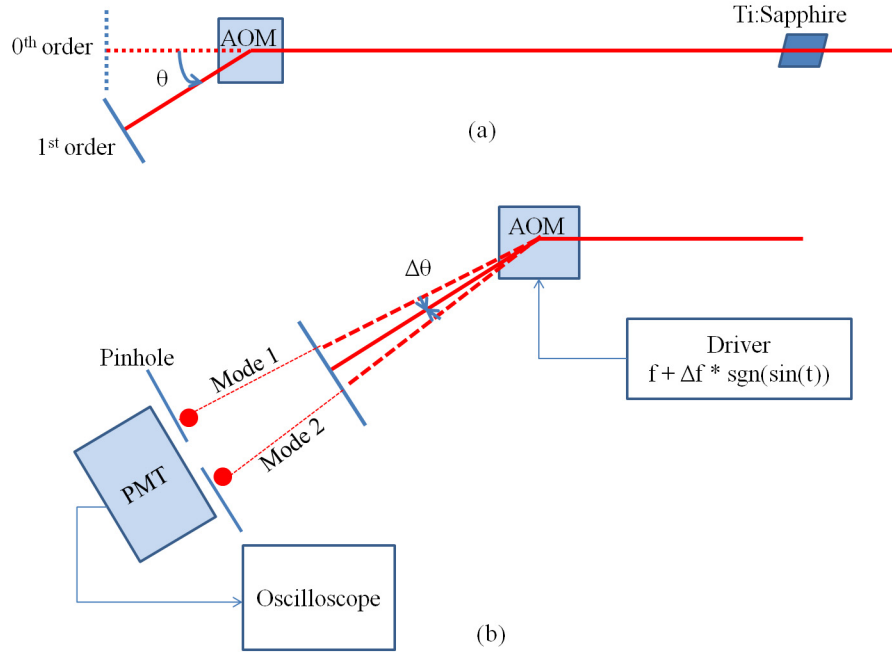


Figure 2.4: The sketch of an experimental setup to measure the damping coefficient of a Ti:Sapphire cavity. (a) The laser design with AOM inside the cavity. (b) A magnified look at the end of cavity, with a small perturbation to the system.

However the intensity signal measured with PMT only showed the gain recovery of the Ti:Sapphire (microseconds). The expected oscillations are on the scale of a few round-trip time, and their frequency should be a function of the cavity configuration, which can be adjusted with the spacing of the two curved mirrors at each side of the Ti:sapphire crystal. One essential condition to observe the transients of the cavity is that the departure from equilibrium be turned on in less than one round-trip. Instead, with the AOM available, the “perturbation” is turned on adiabatically, in a time long compared with the frequency to be observed, and probably even long compared with the damping time that we are seeking to observe.

A successful experiment would require a fast (1 ns) electro-optic deflector, driven by a square wave. While academically interesting, it was decided not to pursue further an experiment that required a too large time and financial investment, and

was distracting from the main goal of the research.

2.4.2 Indirect determination through 2-level system analogy

Whether in space or time, it is difficult at best to pinpoint a physical interpretation to the damping constant ϵ . The answer may reside in an experiment performed previously with a Ti:sapphire laser, which had as an objective to study the analogy between the mode-locked Ti:sapphire laser and a quantum mechanical two-level system [30, 31]. The experimental situation was that of a mode-locked ring laser, in which two pulses can circulate in opposite directions. The intensities $\tilde{\mathcal{E}}_1\tilde{\mathcal{E}}_1^*$ and $\tilde{\mathcal{E}}_2\tilde{\mathcal{E}}_2^*$ in either direction¹ are the equivalent of the diagonal density matrix elements ρ_{11} and ρ_{22} of a two-level system. The off-diagonal element is $\rho_{12} = \tilde{\mathcal{E}}_1\tilde{\mathcal{E}}_2^*$, and the two-level system can be “detuned from resonance” by a frequency Δ equal to the difference in cavity resonance frequency for the two senses of circulation. Rabi oscillations of the populations ρ_{ii} as well as of the off-diagonal matrix elements ρ_{ij} can be produced by letting the laser evolve with only one direction oscillating, as an initial condition. A unidirectional initial condition is created by feeding back one direction into the other with a mirror. At time $t = 0$, the feedback is blocked by a fast Pockel’s cell, and Rabi oscillations manifest themselves as alternating directionality in the ring laser. As in the case of a two-level system, there is an “energy relaxation time” for the diagonal elements, and a “phase relaxation time” for the off-diagonal elements. The measured phase relaxation time of 30 ms may be considered as an upper limit to the damping time constant τ_{rt}/ϵ .

It should be noted that in this particular experiment, the gain and losses remain constant (there is no change in total pulse energy from one round-trip to the next). The situation that will be investigated in Chapter 7 involves dynamics of gain and

¹ $\tilde{\mathcal{E}}_1$ and $\tilde{\mathcal{E}}_2$ are the complex field amplitudes of the pulses circulating counter-clockwise and clockwise, respectively

saturable absorption, hence constant changes in pulse energy and duration characterized by much faster time constants. It will be shown in Chapter 7 that the gain relaxation dynamics plays the same role as a damping coefficient ϵ .

2.5 Time versus space matrix

For the comparison of time and space matrix, one needs to point out that it is possible to have both positive and negative dispersion. It automatically results in an opposite signs of chirp created by the time lens. Therefore the damping coefficient also has to be opposite signs. For instance, a positive Kerr effect requires negative dispersion compensation, the pulse before entering the Kerr medium should have a negative chirp, and ϵ needs to be positive for proper damping.

The temporal ABCD matrix is more complex than the spatial ABCD matrix in the sense that the “temporal lens” is modified at every round trip through energy and pulse duration evolution. Therefore it becomes unpredictable whether the stability condition is always satisfied or not. However this argument is also true for spatial ABCD matrix if a Kerr lens is taken into account for the laser cavity design.

Chapter 3

Optoelectronic modulator inside laser cavity

One of the most challenging aspects in using intracavity phase interferometry to detect slow changes of phase (slow compared to the cavity round-trip), is the splitting of the “sample” and “reference” pulses in two separated arms of the cavity. One possible solution to this challenge is to use an optoelectronic modulator (EOM) to send the two pulses towards a different path at each half round-trip. The facts that it will be used inside a laser cavity, and driven by relatively high voltage at radio frequency add to the complexity. This chapter is devoted to the study of the operation of the EOM inside the laser cavity. The application to length measurement is treated in the next chapter. We first explain the function of the EOM as phase and amplitude modulator outside the cavity in section 3.1. Then we will focus on the effects taking place when it is brought inside a laser cavity. In section 3.2 we discuss the Fabry-Perot effect of a phase modulator inside the cavity, both experimentally and theoretically. Next we study it as an amplitude modulator combined with PBS which splits the laser pulses into two arms, and demonstrate how it is driven electrically in section 3.4. We are interested in the amplitude and phase of the pulse

returned from the two arms, and will describe the corresponding beat note signal that can be generated of this setup in section 3.5.

3.1 Mechanisms of phase and amplitude modulator

An EOM is a device that can be used to control the phase or polarization of an optical beam using the Pockels' effect. The latter is a first order electro-optical effect that introduces a birefringence change. The refractive change along each optical axis is proportional to a constant or varying electrical field V :

$$\Delta n(V) = \frac{2r}{n_0 d} V = \beta V \quad (3.1)$$

where r is the electro-optical coefficient of the material, n_0 is the refractive index when no voltage is applied, and d is the distance between two electrodes where the voltage V is applied. We define β as the coefficient for Pockels' effect that will be measured in section 3.2. The Pockels' effect occurs only in crystals that lack inversion symmetry, such as lithium niobate, lithium tantalate or gallium arsenide and in other noncentrosymmetric media such as electric-field poled polymers or glasses. In this chapter we will use lithium tantalate which has an $r_{33} = 30.4\text{pm/V}$.

An EOM could be used as phase or amplitude modulator, depending on the polarization of the input beam with respect to the crystal axes. The EOM acts as a phase modulator only when the input beam is linearly polarized along one of the crystal axes. If the input polarization has components projected on both crystal axes, the output polarization becomes elliptically polarized. When the elliptical polarization pass through a polarizing beam splitter, it will be separated into two orthogonal polarizations. If we only consider one of the linear polarizations, the EOM

and polarizer combination provides amplitude modulation. The details related to our specific setup will be discussed in section 3.5.

The driver electronics are not trivial for an EOM at radio frequency (RF). Pockels' cells have a non-negligible capacitance. For instance, the capacity of the EOM we use in the lab is in the order of 10pF, while a typical RF amplifier requires 50Ω load. Therefore the impedance of the EOM has to be matched to create a transmission line for the desired RF frequency. We built our own LC circuit based on the design proposed by John Hall in their frequency comb applications [15, 9]. However, in that design, the resonant frequency had only a very limited tuning range than could only be determined by trial and error. Furthermore, we could not achieve a sufficiently narrow resonance, such that a full 90° rotation of the polarization could be achieved with the available RF drive. In order to ensure the best performance, a customized commercial RF amplifier from Amplifier Research is purchased as a dedicated special driver for our EOM.

3.2 Phase modulator inside the cavity acting as a Fabry-Perot etalon

The modulator is first inserted inside a mode-locked laser cavity as a phase modulator. The phase velocity and group velocity of the laser pulse are modified as it propagates through the modulator by applying voltage on it. If the voltage increases over time, the phase and group velocity will be reduced with an increasing refractive index. We define the phase and group velocity delays as:

$$\tau_p(V) = \frac{\phi(V)}{\Omega}, \tag{3.2}$$

$$\tau_g(V) = \frac{d\phi(V)}{d\Omega}, \tag{3.3}$$

where ϕ is the phase of the light wave, $\Omega = c/\lambda$, and “p” and “g” denote “phase” and “group”.

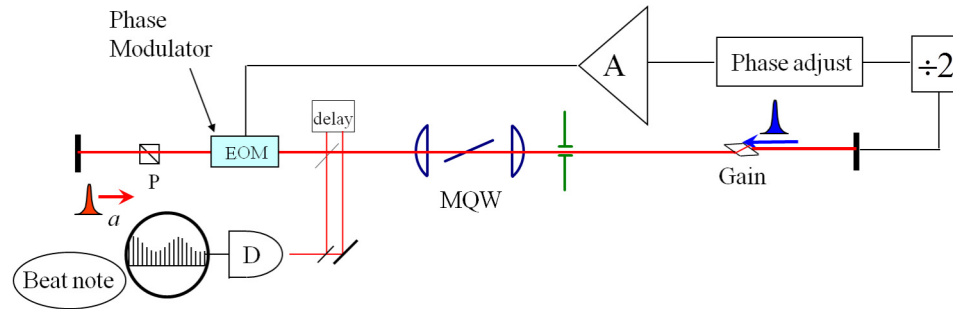


Figure 3.1: Sketch of a experimental setup to measure the phase delay introduced by an EOM inside a laser cavity. The EOM is driven by AC voltage at RF frequency of half the cavity repetition rate. Two pulses are combined through a delay line, the beat note is measured as a function of voltage amplitudes.

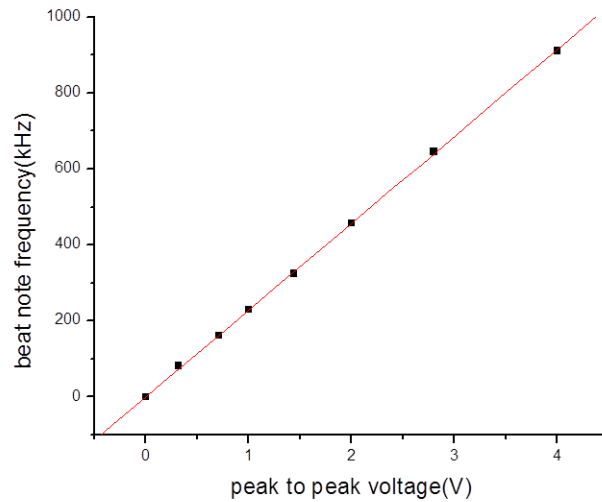


Figure 3.2: Beat frequency plotted as a function of voltage amplitudes.

The phase delay can be measured as the beat note between the two pulses as

shown in Fig. 3.1. The expression for the beat frequency is:

$$\Delta\nu = \frac{\Delta\phi}{2\pi\tau_{RT}} = \frac{(2\pi/\lambda)\Delta n(V)\ell}{2\pi\tau_{RT}} \quad (3.4)$$

where τ_{RT} is the round trip time, ℓ is the length of EOM and $\Delta n(V)$ is defined in Eq. (3.1). Therefore a linear relationship can be found between the beat frequency and the applied voltage. The slope of Fig. 3.2 is used to calculate $\beta \approx 1.65 \times 10^{-8}/V$ defined in Eq. (3.1).

The group delay will be measured by monitoring the repetition rate of the laser illustrated in Fig. 3.3. With the MQW not being in the middle of the cavity, the laser lases with one pulse per cavity. So we extend one end of the cavity from Fig. 3.1 and observe one pulse operation, which allows us to measure the group velocity change only due to the voltage on EOM. One would assume that the group delay also increases linearly with a DC voltage. Counter-intuitively, the group delay is modulated as we increase the DC voltage from -1kV to 0V. As we can see from Fig. 3.4 the cavity round trip time is plotted as a function of DC voltage, with a modulation depth around 80 fs. The explanation of this modulation is that the EOM acts like a Fabry-Perot(FP) inside the cavity. The group velocity associated with the FP structure modifies the group delay of the pulses.

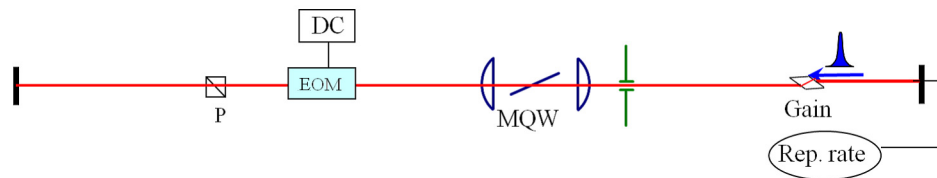


Figure 3.3: Sketch of a experimental setup to measure the group delay introduced by the EOM inside a laser cavity. The EOM is driven by an increasing DC voltage (-1kV to 0V). The repetition rate is measured as a function of voltage.

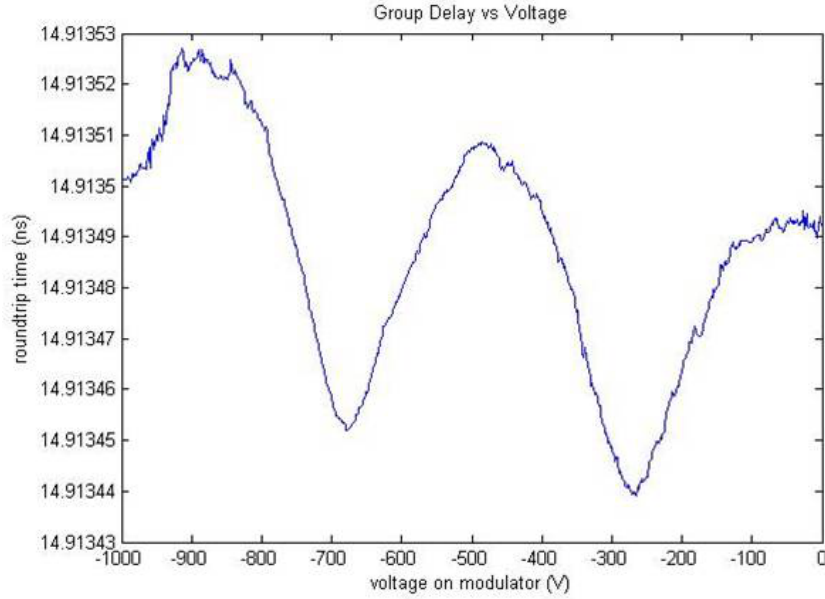


Figure 3.4: Cavity round trip time plotted as a function of DC voltage.

3.3 Simulation of Fabry-Perot effect of the EOM

To explain the periodic structure of the repetition rate, we looked at the phase modification from a FP etalon. The transmission and phase of a FP is written as:

$$T = \frac{(1 - R)e^{-i\delta/2}}{1 - R \cdot e^{-i\delta}} = |T|e^{i\phi}, \quad (3.5)$$

$$\phi = \arctan \left[\frac{1 - R}{1 + R} \tan \left(\frac{\delta}{2} \right) \right], \quad (3.6)$$

where $\delta = \frac{4\pi n\ell}{\lambda} = \frac{\Omega \cdot n(\Omega) \cdot 2\ell}{c}$ when it is normal incidence. The derivative of ϕ is the group delay we are interested in:

$$\frac{d\phi}{d\Omega} = \frac{\partial\phi}{\partial\Omega} + \frac{\partial\phi}{\partial n} \cdot \frac{\partial n}{\partial\Omega} = \frac{d\phi}{d\delta} \left(\frac{n\ell}{c} + \frac{dn}{d\Omega} \cdot \frac{\Omega\ell}{c} \right), \quad (3.7)$$

Fig. 3.5 shows the plot of transmission $|T|$ and phase ϕ (normalized by π) and $d\phi/d\delta$ of a FP as function of δ . The modulation effect is exaggerated as we set

the reflectivity R of both surfaces to be 0.5. In the experiment, with an AR coating $R = 0.025\%$ the modulation becomes much less and manifests itself as the modulation of repetition rate (Fig. 3.4).

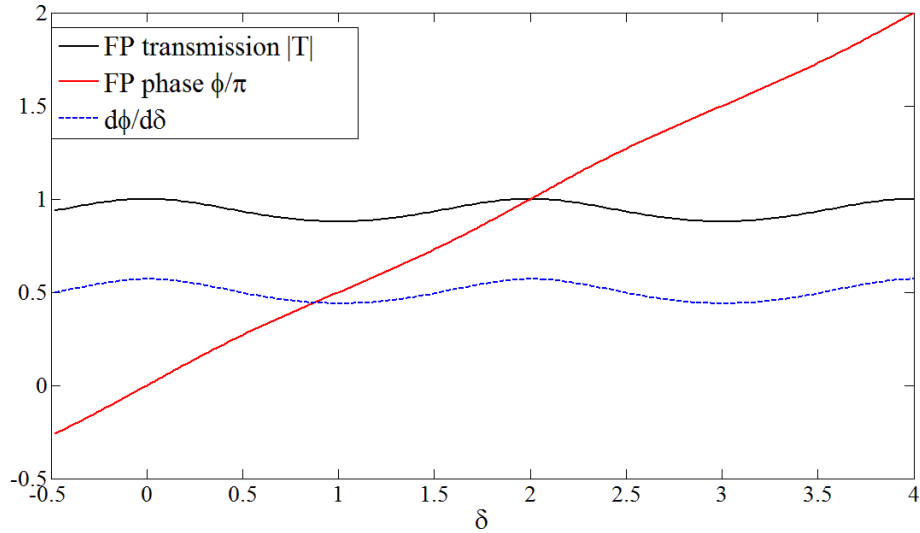


Figure 3.5: transmission $|T|$ (black), phase ϕ/π (red) and $d\phi/d\delta$ (blue) of a FP are plotted as a function of δ .

In Eq. (3.7), $d\phi/d\delta$ is known for a given R , and $n(V)$ can be calculated from $n = n_0 + \beta V$. The only undetermined term is $dn/d\Omega$ in the brackets. We can extract the appropriate value of $dn/d\Omega$ from a fit of the experimental data (Fig. 3.4). Fig. 3.6 is the simulation result of the group delay versus applied voltage on the AOM, using the fitted value $dn(V)/d\Omega = [3 \times 10^{-17} - 0.6 \times 10^{-22} \cdot V]$ s, where V is in volts. The simulation reproduces both the delay about 30 fs and the modulation depth of around 80 fs.

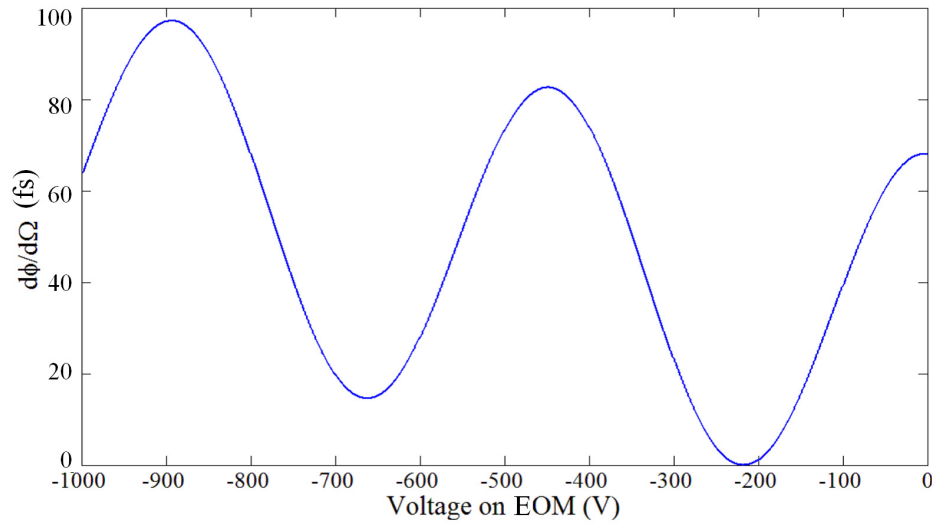


Figure 3.6: Simulation of the group delay versus voltage on the EOM, using $dn(V)/d\Omega = [3 \times 10^{-17} - 0.6 \times 10^{-22} \cdot V]$ s, the later value obtained from a fit to the experimental data of (Fig. 3.4).

3.4 Amplitude modulator combined with PBS and two arms

The role of the EOM and PBS combination is to divert every other pulse (remembering that there are two intracavity pulses circulating in the cavity) towards a specific branch. It is thus placed inside the laser cavity as an amplitude modulator. The laser beam is linearly polarized at an angle 45 degrees to the crystal axis of the modulator, and a polarizing beam splitter is placed behind it to separate the horizontal and vertical polarizations. Let us assume a cavity round-trip frequency of 90 MHz, and the two pulses operation makes pulse repetition rate 180 MHz. The modulator can be either driven at the cavity round-trip frequency of 90 MHz (high modulation frequency, lower peak to peak voltage), or at half that frequency (45 MHz, lower modulation frequency, higher peak to peak voltage). These two options are analyzed in the two sections that follow.

Modulator operating at 90 MHz

The 180 MHz signal from a photodiode is divided by 2, amplified, sent through an adjustable delay, an amplifier, and the modulator, as sketched in Fig. 3.7. The signal from the fast photodetector D_1 is sent through a high pass filter, a TTL converter, and a divider by 2. An accurate delay line is used to ensure that the zero and maximum voltages in the modulator correspond to the arrival time of the optical pulses. A couple of amplifiers are used to boost up the signal to the half-wave voltage. The pulse train is monitored through detector D_2 with an oscilloscope, frequency counter and spectrum analyzer. With the present system, the mode-locking is considered “good” when repetition rate fluctuations on the frequency counter are less than 100 Hz.

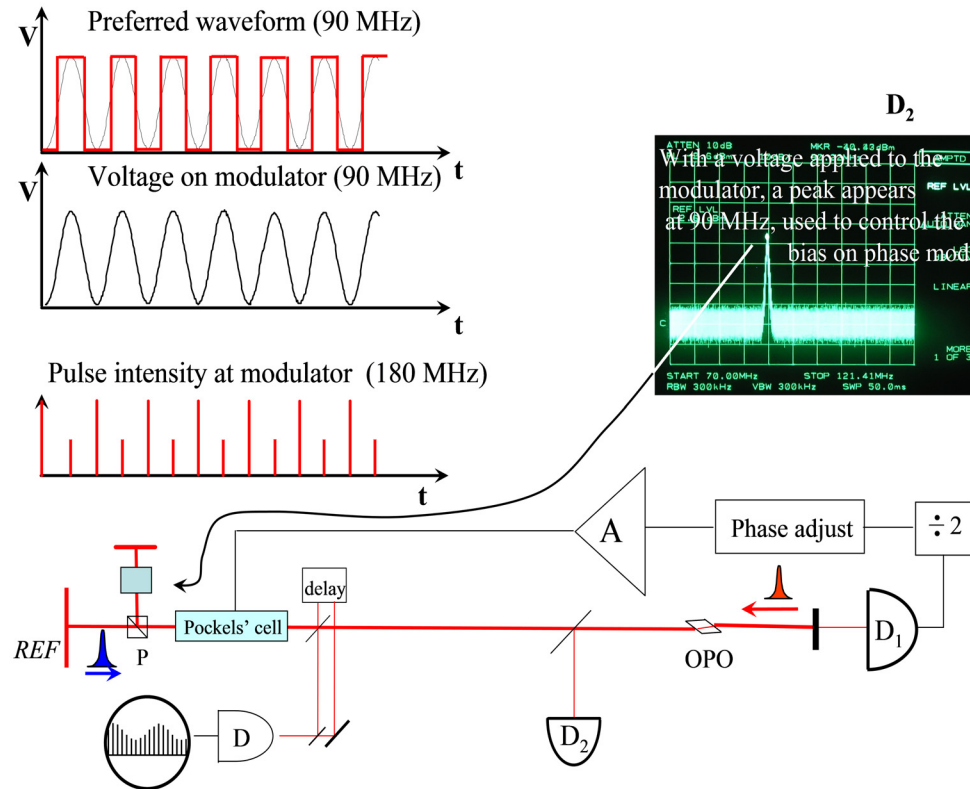


Figure 3.7: Since there are two pulses circulating in the cavity, the detectors D_1 or D_2 (bottom part of the figure) record a signal at 180 MHz, which is divided by 2, phase adjusted, amplified before being sent to the EOM. Unless the two pulses returning to the polarizing beam splitter are exactly dephased by π (relative displacement of $\lambda/4$ of the two mirrors of the end-cavity interferometer) the “red pulse” will be attenuated as sketched in the graph “pulse intensity at modulator (180MHz)”. Because of the reduction of intensity of the “red pulses”, a component at 90 MHz appears in the spectrum analyzer recording of the pulse train on detector D_2 . That 90 MHz component is used in a feedback loop to maintain the bias of the phase modulator for the minimum loss condition, which corresponds to zero signal at 90 MHz.

The electrical drive at 90 MHz should ideally be a square wave, which ensures the pulse seeing the either zero or the maximum voltage when passing through the EOM twice. This allows more freedom on the delay line to adjust the phase of the applied RF signal.

The 90MHz signal to the modulator is from zero to a half wave voltage (90 degree

rotation of the polarization). In the case of the tantalate modulator that we have, the “zero” requires a bias of -50V, to which we have to add a peak to peak voltage of 183 V.

Modulator operating at 45 MHz

An alternative to the modulator drive of the previous section is to operate between 0 V (transmitting the “blue” pulse) and \pm the half wave voltage $\pm V_\pi$. The advantage is a lower frequency. The disadvantage is that twice the peak-to-peak voltage of the previous section is required. The desired waveform should have both flat tops and bottoms, and the symmetry of the waveform must be guaranteed to avoid unwanted bias beat note, since one of the pulses goes through zero voltage that has a steep slope. The corresponding set-up is sketched in Fig. 3.8.

3.5 Amplitude and phase of the return pulse

In this section, we will discuss only the pulse that goes through a non-zero voltage and experiences polarization change, since the other pulse going to the reference arm always sees zero voltage and is not affected by EOM.

Before entering the modulator, the pulse of our interest is vertically polarized, along an axis \hat{y}_1 , orthogonal to an horizontal axis \hat{x}_1 . The initial field is [$E_{x1} = 0$, $E_{y1} = 1 \cdot \exp(i\omega t)$]. We project that field on the principal axes \hat{y}_m and \hat{x}_m of the modulator (oriented at 45°):

$$\begin{aligned} E_{xm} &= E_{y1} \cos \theta_1 = \frac{1}{\sqrt{2}} e^{i\omega t} \\ E_{ym} &= E_{y1} \sin \theta_1 = \frac{1}{\sqrt{2}} e^{i\omega t} \end{aligned} \tag{3.8}$$

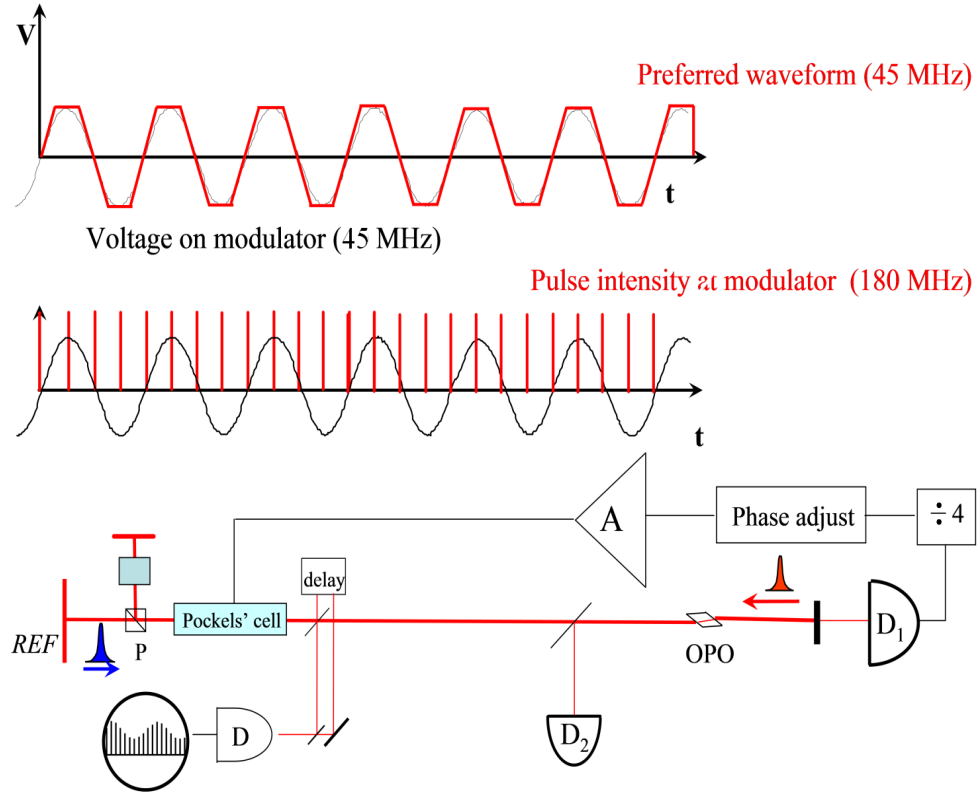


Figure 3.8: The signal recorded by detector D_1 is now divided by 4. The “blue” pulse crosses the modulator at the null of the electric field. The “red” pulse crosses at the (+) and (-) peaks of the modulation.

The modulator introduces a difference in phase (which we will put on the axis y_m) ϕ . At the output of the modulator:

$$E_{ym} = \frac{1}{\sqrt{2}} e^{i(\omega t + \phi)}. \quad (3.9)$$

After the modulator, we project again on vertical \hat{y} and horizontal \hat{x} axis:

$$\begin{aligned} E_x &= \frac{1}{2} (1 - e^{i\phi}) e^{i\omega t} \\ E_y &= \frac{1}{2} (1 + e^{i\phi}) e^{i\omega t} \end{aligned} \quad (3.10)$$

After the beam has been split by the polarizing beam splitter, the returning beams

recombine after having been given a relative phase shift Δ introduced by the difference in two arms:

$$\begin{aligned} E_x &= \frac{1}{2} (1 - e^{i\varphi}) e^{i\Delta} \\ E_y &= \frac{1}{2} (1 + e^{i\varphi}) \end{aligned} \quad (3.11)$$

The beam returns towards the modulator. Projecting on the modulator axis:

$$\begin{aligned} E_{xm} &= \frac{1}{\sqrt{2}} (E_x + E_y) = \frac{1}{2\sqrt{2}} [1 + e^{i\Delta} + e^{i\varphi} (1 - e^{i\Delta})] \\ E_{ym} &= \frac{1}{\sqrt{2}} (E_y - E_x) = \frac{1}{2\sqrt{2}} [1 - e^{i\Delta} + e^{i\varphi} (1 + e^{i\Delta})] \end{aligned} \quad (3.12)$$

The modulator imparts another phase change of φ on the \hat{y}_m component $E_{ym} \rightarrow E_{ym} \times \exp(i\varphi)$. The components along \hat{y}_1 and \hat{x}_1 of the pulse re-entering the main-cavity are thus:

$$\begin{aligned} E_{x1} &= \frac{1}{4} [(1 - e^{i\varphi}) + e^{i\Delta} (1 + e^{i\varphi}) + e^{i\varphi} (1 - e^{i\varphi}) - e^{i\Delta} e^{i\varphi} (1 + e^{i\varphi})] \\ E_{y1} &= \frac{1}{4} [(1 + e^{i\varphi}) + e^{i\Delta} (1 - e^{i\varphi}) + e^{i\varphi} (1 + e^{i\varphi}) - e^{i\Delta} e^{i\varphi} (1 - e^{i\varphi})], \end{aligned} \quad (3.13)$$

which can be re-written:

$$\begin{aligned} E_{x1} &= \frac{1}{4} [(1 - e^{i\varphi}) (1 + e^{i\varphi}) (1 + e^{i\Delta})] \\ E_{y1} &= \frac{1}{4} [(1 + e^{i\varphi})^2 + e^{i\Delta} (1 - e^{i\varphi})^2]. \end{aligned} \quad (3.14)$$

The normalized intensity of the return pulse is plotted as a function of Δ for different φ in Fig. 3.9. The complexity of these equations arises from the fact that the Michelson type interferometer with the polarizer as beam splitter acts as a wave plate, since it introduces a phase shift Δ between the two components. The object of the IPI measurement is precisely to measure the phase shift Δ . The EOM modulator on the other hand is a voltage dependent waveplate, introducing a phase shift $\varphi(V)$.

From this last set of equations, it is obvious that $\Delta = \pi$ returns the correct polarization independently of φ . This is of practical interest, in cases where the sample arm cannot take the full power of the intracavity pulse. Only a small portion (small value of φ) of the beam is sent towards the sample arm. If Δ is close to π (between $2\pi/3$ and $4\pi/3$), over 90% of the pulse intensity $|E_{y1}|^2$ is returned to the main portion of the cavity [see Fig. 3.9]. A beat note of a frequency proportional to the elongation of either reference or sample arm can be recorded, as demonstrated at the end of Chapter 4. The penalty for sending only a weak portion of the pulse to the sample arm is a reduced phase shift upon return. See Section 3.5.3 for a more detailed analysis.

Another interesting limit is when $\varphi = \pi$: the modulator acts as a half wave plate, rotates the polarization of one of the two intracavity pulses by 90° , which sends its full energy to the sample arm. As that pulse returns back, its original vertical polarization is restored at the modulator, and there are no losses. The phase difference shift of the returning signal has the value of Δ , and a beat note of $\Delta\nu = \Delta/(2\pi\tau_{rt})$ can be measured. This case is analyzed in the next Section 3.5.1.

The limit of φ near zero is discussed in Section 3.5.2

3.5.1 Near a complete polarization rotation (φ close to π)

There are two special limits to the phase shift φ of the EOM: near zero and π where the minimum return loss occurs. The ideal value is $\varphi = \pi$, which corresponds to a rotation of the polarization by 90° , consistent with the set of Eqs. (3.10). For $\varphi = \pi$:

$$\begin{aligned} E_{x1} &= 0 \\ E_{y1} &= e^{i\Delta} \end{aligned} \tag{3.15}$$

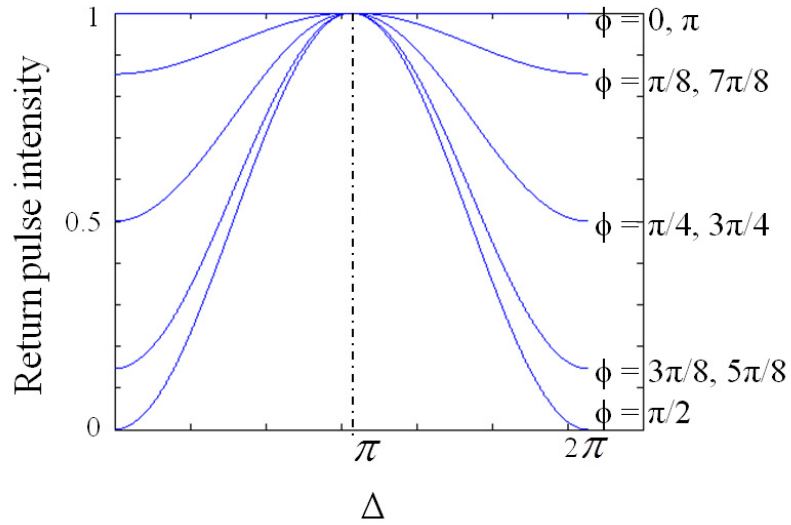


Figure 3.9: The normalized intensity of the return pulse is plotted as a function of Δ for φ from 0 to π , with $\pi/8$ interval

The phase of the return pulse is ideally the relative phase difference Δ between two arms as shown in Fig. 3.10.

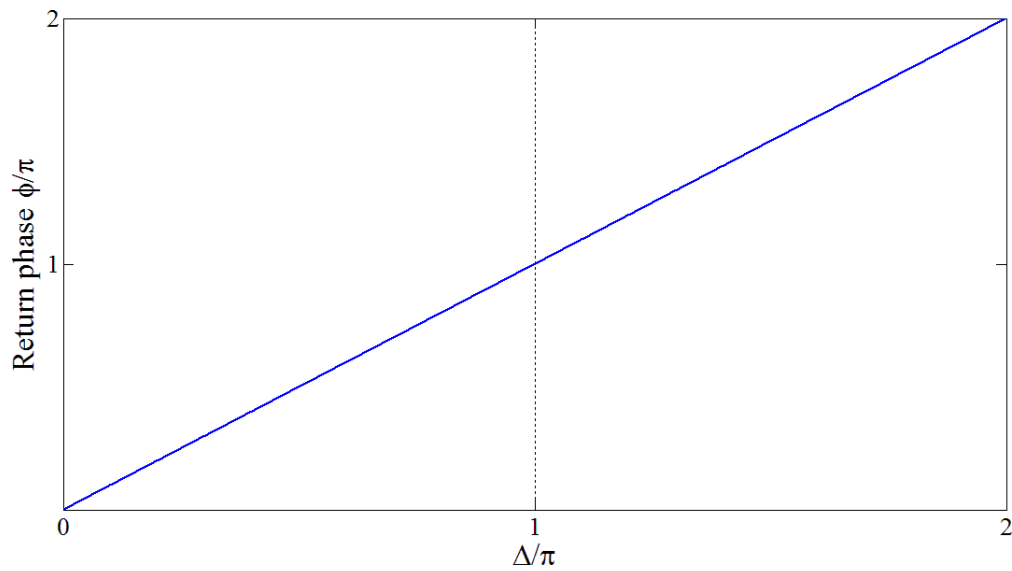


Figure 3.10: The phase change of the return pulse is plotted as a function of Δ for $\varphi = \pi$

Let us now consider a small departure from this ideal condition: $\varphi = \pi - \epsilon$. For small ϵ , the set of Eqs. (3.14) becomes:

$$\begin{aligned} E_{x1} &= \frac{1}{4} \left[i\epsilon (2 - i\epsilon) (1 + e^{i\Delta}) \right] \\ E_{y1} &= \frac{1}{4} \left[(i\epsilon)^2 + e^{i\Delta} (2 - i\epsilon)^2 \right]. \end{aligned} \quad (3.16)$$

The x component of the field represents a loss of the order of ϵ in field, ϵ^2 in intensity. Besides the loss of the y component of the field to the x component, it has a phase shift that remains equal to Δ to first order.

3.5.2 Small polarization rotation (φ close to zero)

If $\varphi = 0$, which means no rotation of the polarization at all, the phase of the return pulse will contain no information about Δ . If φ is small, only a tiny part of the pulse will be sent to the sample arm and the return pulse contains a small phase shift due to Δ , therefore produces a much reduced beat frequency. The small φ approximation of Eqs. (3.14) is:

$$\begin{aligned} E_{x1} &\approx \frac{1}{2} i\varphi (1 + e^{i\Delta}) \\ E_{y1} &\approx \left[1 + i\varphi - \frac{\varphi^2}{4} (1 + e^{i\Delta}) \right]. \end{aligned} \quad (3.17)$$

As shown in Fig. 3.11, the return phase is no longer proportional to Δ as before, instead the return phase presents a sinusoidal behavior. This suggests that the beat note between sample and reference pulses can no longer determine what exactly Δ is.

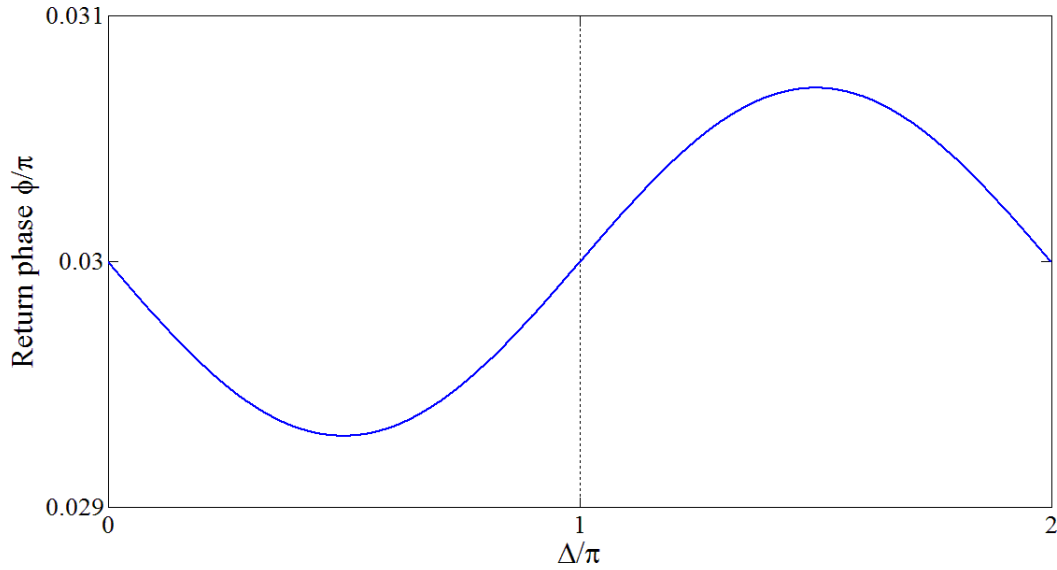


Figure 3.11: The phase change of the return pulse is plotted as a function of Δ for $\varphi = 0.03\pi$

3.5.3 Δ near π

From the previous two sections we find that the relation between return phase and Δ evolves from a straight line to a sinusoidal curve when φ is changed from π to 0.03π . It is safe to conclude that no matter what value of φ is taken (any value between 0 and π), the value of Δ should always be set to a limited region around π . In this case the beat note retrieves an approximate linear dependence on Δ . The price for operating at $\Delta \approx \pi$ is that the bias beat note equals $\phi/(2\pi\tau_{RT})$.

We can introduce a feedback to maintain the polarization along x component zero. This imposes that we set $\Delta = \pi + \eta$, where η is a small value (Δ being maintained near π by a feedback loop, η being a small phase modulation imposed by a phase modulator).

If Δ is set close to π , the return phase can be approximated for small φ by $\varphi + 2 \cdot (\varphi/\pi)^2 \times (\Delta - \pi)$. This last approximation results from a fit of Fig. 3.11.

Chapter 4

Nd:YVO₄ laser for IPI

The first demonstration of Intracavity Phase Interferometry for displacement measurement was performed with Nd:Vanadate laser mode-locked by saturable absorbers in Section 4.1. A total displacement of 250 nm in optical path was measured with the resolution better than 10 nm. While it has successfully demonstrated the capability of sensing nm displacement by sending only fraction of intracavity power to the sample arm, the use of flowing dye jet, involving solvents with harmful vapors and cancerogenic organic dyes is not seen as a viable option for commercial instruments. The following approach is to test an all solid state laser and saturable absorber combination, where the dye jet is replaced by Multiple Quantum Wells (MQW) as the saturable absorber. The laser setup is presented in Section 4.2, with a MQW saturable absorber serving as well as mode-locking element and the generator of a pair of intracavity pulses. Beat note measurements are presented. It was discovered that the MQW introduces a coupling between the two sets of pulse trains. In section 4.3 further experiments and simulations produce actual evidence of the coupling of phase and group velocity through MQW, and, in general, nanostructures inside a cavity.

4.1 Experimental setup with and displacement measurement

4.1.1 The vanadate laser

Nd:YVO₄ exhibits a high pump absorption and gain, with a typical laser emission wavelength at 1064 nm and a fluorescence lifetime around 90 ns [27]. Nd:YVO₄ is well suited for passively mode-locked lasers with very high pulse repetition rate. This feature results mainly from the high laser cross sections and the strong pump absorption.

For beat note measurement, the Electro-Optic Modulator (EOM) and Polarization Beam Splitter (PBS) with two subcavities replace one end of the cavity after the laser is pre-aligned. When the saturable absorber is set to be in the middle of the cavity, this geometry leads to an operation with two pulses in the cavity. It is the configuration of minimal losses because of the mutual saturation in the absorber, where the two counter-propagating waves collide. As shown in Fig. 4.1, two pulse trains can be extracted from this cavity and, given appropriate relative delays, made to interfere on a detector.

The main implementation of the set-up of Fig. 4.1 is to direct one of the two intracavity pulses towards the reference arm, and the other (via a $\lambda/2$ pulse applied to the EOM) to the sample arm. Each of the pulse trains corresponding to these two intracavity pulses will have an average carrier frequency given by $\nu_{ref} = NL/v_p$ and $\nu_s = N(L + \Delta L)/v_p = \nu_{ref} + \Delta\nu$. The interference of the two pulse trains on a detector gives rise to a beat note $\Delta\nu = N\Delta L/v_p = \nu\Delta L/L$, hence a precise measurement of the length difference ΔL between the two arms of the end-of-cavity interferometer.

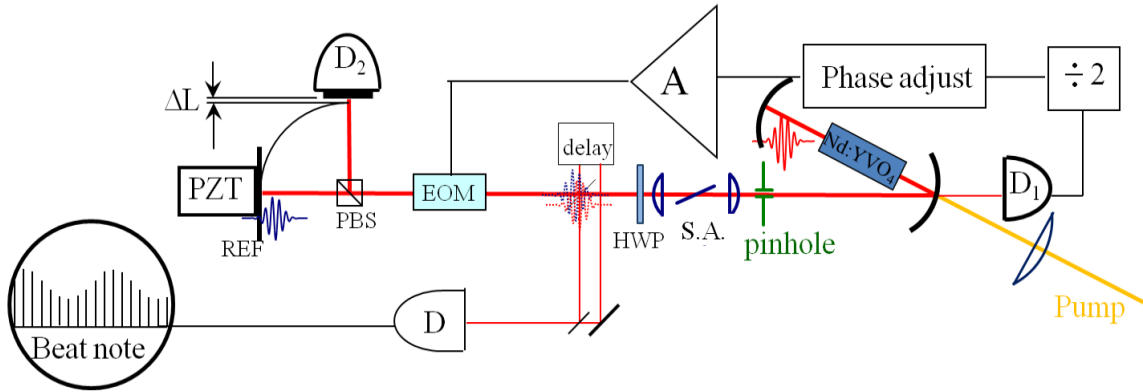


Figure 4.1: The linear setup of a diode-pumped Nd:YVO₄ laser. A saturable absorber (S.A.) is located in the middle of the linear cavity, with two identical lenses to focus the beam tightly in the saturable absorber. A pinhole is inserted after one of the lenses to assist passive mode-lock. For beat note measurement, the EOM and PBS with two subcavities replace the other end of the cavity after the laser is prealigned.

Our Nd:YVO₄ laser is pumped by a diode at 808nm. The diode pump with a fiber output is focused by two lenses into the Nd:YVO₄ crystal. The laser setup is sketched in Fig. 4.1. The Nd:YVO₄ laser cavity is terminated at one end by a curved mirror with a radius of curvature of 10cm. The other curved mirror is transparent to 808nm and highly reflective at 1064nm. A saturable absorber is located in the middle of the linear cavity, with two identical lenses to focus the beam tightly in the saturable absorber. A pinhole is inserted after one of the lenses to assist passive mode-locking. There is no dispersion compensation components in the cavity, resulting in a pulse duration of approximately 30 ps.

4.1.2 Saturable absorber and generation of a pulse pair

Saturable absorption by a broadband organic dye [29] was the first mechanism to initiate mode-locking. It was realized very early that the saturable absorption mechanism, being nonlinear, is enhanced in a standing wave configuration. Locating the

absorber at an end mirror of a linear cavity [29], or in a ring cavity [13], results in a contrast enhancement of the nonlinear process. In a ring laser [14], the larger peak power in the antinodes of the standing wave resulted in enhanced saturation for bidirectional operation, hence a mode of operation with two pulses crossing in the saturable absorber. A similar situation exists in the linear laser: if a saturable absorber is located in the middle of the cavity, mutual saturation favors an operation with two pulses circulating in the cavity and crossing at that location [6]. In presence of counter-propagating pulses of intensity I_1 and I_2 , the attenuation of either beam is given by:

$$\frac{dI_{1,2}}{dz} = -\frac{|\alpha_0|}{1 + \frac{I_{1,2}}{I_s} + \beta \frac{I_{2,1}}{I_s}} I_{1,2}, \quad (4.1)$$

where α_0 is the linear absorption coefficient, I_s the saturation intensity, and β the mutual saturation coefficient. Since $\beta = 2$ for most homogeneously broadened media such as dye solution or semiconductor absorbers, the absorption of counter-propagating pulses of equal intensity will saturate $3 \times$ as easily as one single traveling pulse [12]. Therefore, two pulse operation, where a standing wave is created at the pulse meeting point, is favored.

4.1.3 Displacement measurement with a dye jet saturable absorber

We choose a dye jet saturable absorber (Fig. 4.2) made of Q-Switch V dissolved in EPH solvent. Because of the motion of the particles in the dye jet, the phase of the backscattering is randomized, and there should be no phase coupling between the two pulses. However, the mode locking operation of the laser is noisy, and the mode locking is assisted by the EOM. The DC bias voltage on the EOM is -60 V. With a peak to peak AC voltage of 3.1 V on the EOM, The pulse are here partially

alternating between the reference and sample arm.

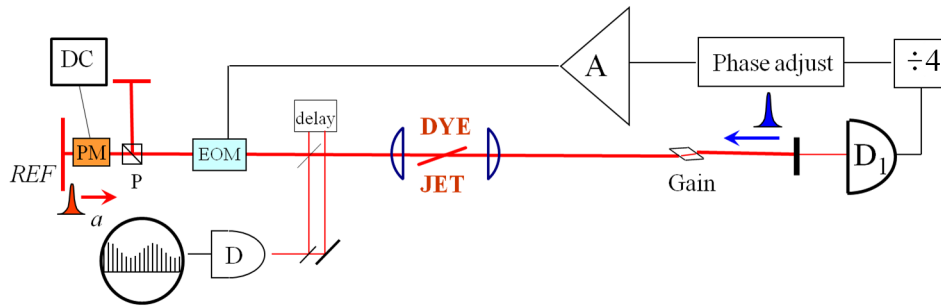


Figure 4.2: Linear mode-locked Nd:YVO₄ laser with dye jet in the middle of the cavity. A thin phase modulator is placed in the reference arm to create a beat note.

Instead of changing the length of the sample arm with a piezoelectric translator, we change the length of the reference arm by applying a DC voltage to a phase modulator inserted in the reference arm. The beat note is measured, and is proportional to the elongation of the optical path in the reference arm achieved through application of a DC voltage on the phase modulator inserted in that path. The voltage applied on the phase modulator is varied from -100 V to -800V, with 50 V interval. The optical path change due to the 700 V change on the phase modulator is calculated to be about 250 nm, given the coefficient for Pockels' effect measured in chapter 3.

In Fig. 4.3 the beat note is plotted as a function of displacement due to the applied DC voltage on the phase modulator. There is still some bias beat note from the system. In total, a displacement of 250 nm has been measured.

Fig. 4.3 shows that we still have a resolution in the nm range, despite the noisy mode locking behavior of a dye jet vanadate laser. For the first time, we demonstrate that we can get a beat note even if only a fraction of the pulse energy is sent towards the sample arm. A beat note change of around 15 kHz is measured as a result of a total displacement of 250 nm in optical path. According to chapter 3, if the pulse is sent completely to the sample arm, a 250 nm change in optical path should result in

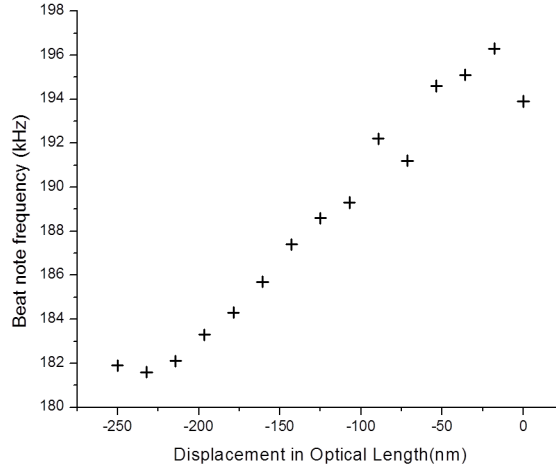


Figure 4.3: The beat note is measured as a function of displacement due to the applied DC voltage on the phase modulator. A total displacement of 250 nm has been measured.

a beat note of 42 MHz instead. The beat note is indeed heavily reduced by the fact that only a small fraction of the pulse energy is sent towards the sample arm. There is a simple estimation of this effect. The AC voltage of 3.1V on the EOM introduces only $\pi/60$ phase shift, therefore the return phase is reduced by a factor of 1/1800, and then generate a beatnote of 23 kHz, which is close to the 15 kHz we measured.

4.2 Experimental setup with MQW and beat note measurement

The facts that the mode locking operation depends on the dye jet temperature, and that the position of jet has to be adjusted with laser intensity and that the dye has a short lifetime, all make the previous setup unsuitable for long time measurements. Therefore the following experiment is to test an all solid state laser and saturable absorber combination, where the dye jet is replaced by Multiple Quantum Wells

(MQW) as the saturable absorber. The MQW substrate is transparent to 1064 nm, and inserted in the cavity at Brewster angle to minimize the loss due to reflection.

There are several sources of noise and error in a displacement measurement, that limit the ability to detect a small beat note $\Delta\nu$. A first experiment is to investigate the beat note that can be produced without using the two arms of the interferometer. To that effect, the EOM is oriented with its axis parallel to the laser polarization, to act simply as a phase modulator. Both pulses propagate then only in the reference sub-cavity. The EOM is driven at the cavity round-trip frequency, with a phase adjustment such that one of the pulse gets a positive phase increment $2\pi\Delta n\ell/\lambda$ and the other a negative increment $-2\pi\Delta n\ell/\lambda$, where ℓ is the path-length in the EOM.

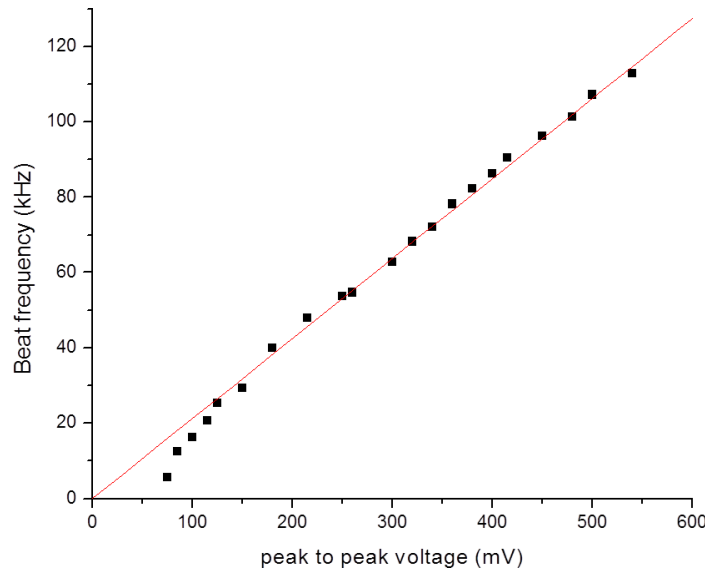


Figure 4.4: The beat note is measured as a function of the peak to peak voltage on the phase modulator. The data deviated from a straight line that passes through origin indicates the deadband.

The laser is operating at 180MHz, with fluctuation of less than 100Hz when EOM and PBS is not inside the cavity. The beat note is plotted as a function of the voltage applied on the EOM in Fig. 4.4. The measured frequency becomes noisier when the voltage approaches zero. This deviation from a straight line that passes through

origin is a clear evidence of a deadband [26].

4.2.1 Beat note measurements

The beat note is measured as a function of the peak to peak voltage on the phase modulator. The data deviated from a straight line that passes through origin indicates the deadband, because of the injection locking between two pulse trains. The observed beat note bandwidth is of the order of 1 kHz, in contrast to the 1Hz bandwidth observed with a similar laser system where the MQW is replaced by a flowing dye jet [32]. In the case of an optical parametric oscillator supporting two intracavity pulses, a beat note as narrow as 0.17Hz is observed [34]. The experiment described in Section 4.3 that follows shed light on this anomaly.

4.3 Coupling of phase and group velocity through MQW

4.3.1 Experimental setup and results

The experimental setup chosen to investigate a potential coupling between phase and group velocities is sketched in Fig. 4.5 (a), with the MQW located in the middle of the linear cavity. The MQW bandgap is designed for peak absorption at the operating wavelength, so that the index of refraction is not affected by saturation modification [2].

The pulse repetition rate is recorded as a function of cavity length with a frequency counter in the setup presented in Fig. 4.5(a). A slow motor (speed 100 micron per minute) was used in order to have adequate resolution in recording the repetition

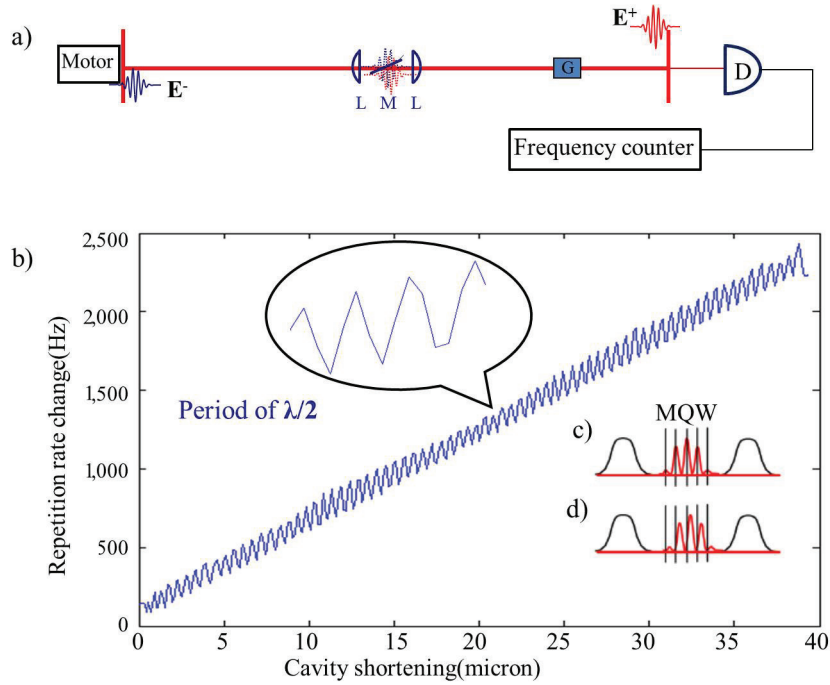


Figure 4.5: Linear mode-locked laser producing two intracavity pulses E_+ and E_- per cavity round-trip. G is the gain medium (Nd:YVO₄), M the multiple quantum well absorber, and L are two AR-coated lenses for tight focusing of the beam in the MQW absorber at Brewster angle. The detector D is to monitor the repetition rate. Insets c) and d) illustrate two extreme cases where the nodes of the standing wave coincide with either the nodes of MQW, or the anti-nodes of MQW, which corresponds to minimum or maximum saturation respectively. These two conditions correspond to a different saturable absorption, hence a different impact on the group velocity of the pulses.

rate. Piezoelectric ceramics were also used to move the mirrors, but the laser cavity being unstabilized, the data were obscured by subwavelength drifts of the cavity. For a linear cavity one expects the repetition rate μ_{RT} to be inversely proportional to the cavity length “ $v_g/2L$ ”. For a small change ΔL in the cavity length, the slope $d\mu_{RT}/dL = -v_g/2L^2$ is approximately a constant. However, we observe a modulation of the repetition rate with a periodicity corresponding to a half wavelength as shown in Fig. 4.5 (b). The MQW used for this experiment has the layout of five 14 nm thick Ga_{0.75}In_{0.25}Al spaced by $\lambda/2$ (134nm) layers of GaAs grown at 350°C.

4.3.2 Saturable absorption and group velocity

Usually the Carrier to Envelope Phase (CEP) value of a pulse in a laser cavity has no effect on its operation. At a fixed location of a crossing point of two circulating pulses in a laser cavity, however, the relative CEP of the two pulses defines the position of the standing wave field. If a nanostructure with high absorption cross section (or gain) is placed at that location, the absorption will reshape the pulse and change the group velocity (repetition rate) of the laser. A standing wave $E_{\text{stand}} = E_+(t) \exp(-ikz_i) + E_-(t) \exp(ikz_i)$ is formed at the location of quantum well z_i where two counterpropagating fields of E_+ and E_- meet. E_+ and E_- are Gaussian electric fields with FWHM of $\sqrt{2 \ln 2} \tau_G$, $E_+(t) = E_0 e^{-[(t-\tau_D)^2/\tau_G^2]}$ and $E_-(t) = E_0 e^{-(t^2/\tau_G^2)}$, where τ_D is the delay time between two pulses at location z_i .

The saturation of absorption is governed by an exponential dependence on the standing wave field [12], for each quantum well

$$a(t)|_{z_i} = a_0 \exp \left[\frac{-\left(\int_0^t E_{\text{stand}} E_{\text{stand}}^* d\tau\right)}{(E_{\text{sat}} E_{\text{sat}}^* T_1)} \right], \quad (4.2)$$

where E_{sat} is the saturation field and T_1 is the fluorescence lifetime. The saturation field increases with the thickness of the quantum well, as the number of electron states increases. A saturable absorber with a lifetime longer than the pulse duration, attenuates the leading edge of the pulse. This results in delaying the center of the pulse and reshaping the envelope. For a nano-structure saturable absorber and two pulses per cavity, the MQW does not trace the envelope but only sees the standing wave at the position of the MQW. As the standing wave moves with respect to the MQW, the saturation of absorption varies periodically. Two extreme cases of maximum and minimum saturation are indicated in the inset of Fig. 4.5.

The coupling is simulated for a 30 ps Gaussian pulse at 1064 nm. Each pulse is modified at the location of the quantum well z_i as it propagates through the medium.

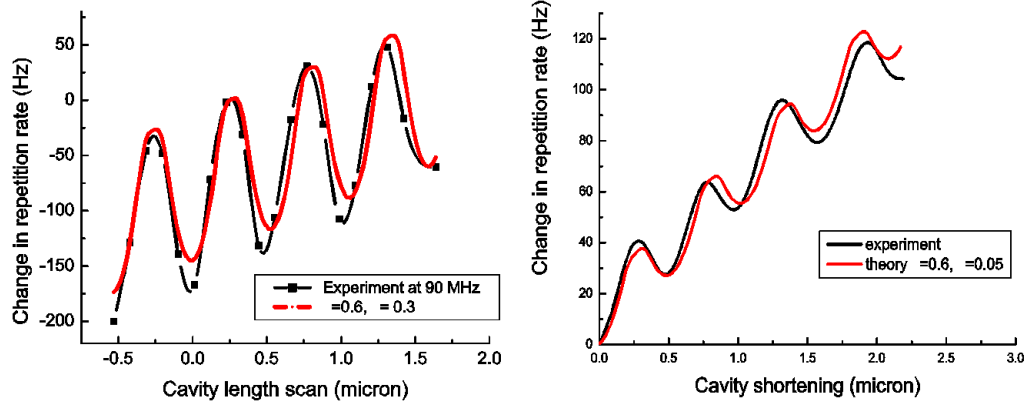


Figure 4.6: Comparison of the experimental data with simulation. The repetition rate of the laser is plotted as a function of cavity length shortening. Left: the departure in repetition rate from 90 MHz when the MQW is at the center. Right: the departure in repetition rate from 85 MHz, when the MQW is off center.

The change in field due to absorption is $\Delta E_{\pm}(t) = a(t)E_{\text{stand}} \exp(\mp ikz_i)$, with $a(t)$ given by Eq. 4.2. As the cavity length scans, the position of the standing wave with respect to the quantum wells moves from z_i to $z_i + \Delta z$ where Δz is the displacement of an end cavity mirror. The pulse overlap is reduced by changing the cavity length to the point where the multiple quantum well is not exactly in the middle. Two situations of fully and partially overlapped pulses are simulated and measured in accordance to the setup in Fig. 4.5 (a). In our modeling $\alpha = [E_{\text{sat}} E_{\text{sat}}^* T_1] / [\sqrt{\pi} E_0^2 \tau_G]$ is a dimensionless parameter that defines the saturation. The values of a_0 , α and τ_D are optimized for best fit of data as shown in Fig. 4.6, where $a_0 = 4 \times 10^{-4}$, $\alpha = 0.6$, $\tau_D = 0$ (full overlap) and $\tau_D = 132\text{ps}$ (partial overlap).

Additional experiments

In the absence of coupling, the repetition rate dependence on the cavity length is a straight line. The data shown in Fig. 4.6 sits beneath this straight line. The

deviation maximizes when the standing wave has its antinodes at the quantum well position. The larger attenuation at the leading edge of the pulse, as opposed to the saturated absorption at the pulse tail, results in a delay of its center of gravity, hence a net reduction of the pulse group velocity averaged over the cavity. To remove the modulation effect on the repetition rate, one straightforward approach is to eliminate the standing wave pattern created when two pulses counter propagate. By moving the MQW far enough from the cavity center, the operation is automatically changed to one pulse per cavity. All other conditions being equal, the repetition rate dependence on cavity length returns a linear relation.

Another approach is to design a non-resonant nano-structure so that there is little coincidence between the standing wave nodes and the nodes or anti-nodes of the MQW. With an aperiodical structure (with spacings of 100nm, 104nm, 108nm and 112nm of GaAs) there is only a small deviation of the repetition rate from the straight line, as the cavity length is modified.(Fig.4.7)

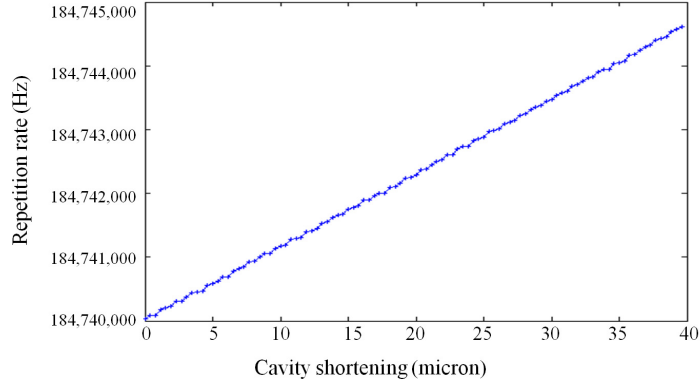


Figure 4.7: The repetition rate plotted as a function of the cavity shortening with a non-resonant MQW in the middle of the cavity.

The modulation is totally eliminated with the use of a saturable absorber dye jet, which does not have any nanostructure. These observations demonstrate that

localized saturable absorption structures (for instance nanostructures) do affect the laser repetition rate by introducing a coupling between phase and group velocities.

To further verify our explanation, an experiment was conducted in a Ti:Sapphire linear laser at 800 nm. Since the substrate is opaque at this wavelength, we have to use the MQW saturable absorber in reflection. The experimental setup is illustrated in Fig. 4.8 (a). Recording the repetition rate of the laser as a function of the laser cavity length for the two pulse per cavity operation, shows the same deviation from the straight line as takes place with the vanadate laser, with a modulation period of 400 nm (Fig. 4.8 (b)).

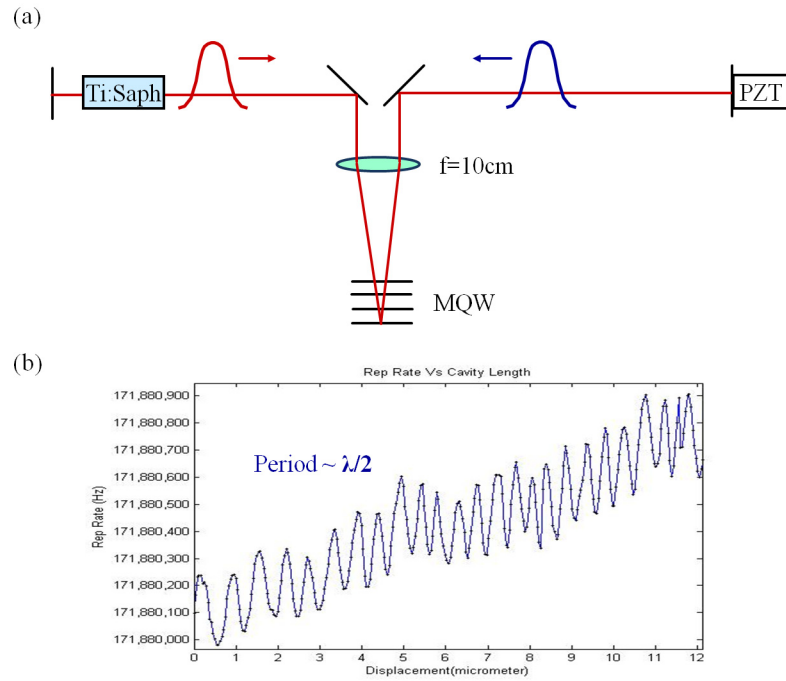


Figure 4.8: (a) Linear mode-locked Ti:Sapphire laser with MQW in the middle of the cavity and an end mirror mounted on a piezoelectric ceramics (PZT). (b) Repetition rate of the laser recorded as a function of the laser cavity length. A deviation from the straight line is present with a modulation period of half the wavelength.

4.4 Conclusion

We have demonstrated IPI response that changes linearly with the displacement ΔL in the reference arm, in the order of nanometers with a solid state laser, with dye jet saturable absorber. By replacing the dye jet with a semiconductor absorber we demonstrate a beat note measurement corresponding to refractive index change in the cavity. It was discovered that the MQW used as saturable absorber introduces a coupling between group and phase velocity, resulting in a beat note bandwidth as large as 10 kHz. Therefore the requirement of IPI that the group velocity be independent of the signal to be measured is not met. A simple theoretical model based on saturation has been shown to match very exactly the shape of the group delay dependence on cavity length [23]. It was therefore concluded that the vanadate laser with saturable absorber was not the ideal system for Intracavity Phase Interferometry.

Chapter 5

Intracavity pumped OPO for IPI

In this chapter we report the implementation of IPI using a mode-locked Ti:Sapphire lasers to synchronously pump an intracavity optical parametric oscillator (OPO). Synchronously pumped OPOs are good candidates for the IPI application because the group delays are externally imposed by a pump laser, without having to insert a physical element that could couple the phases of the intracavity pulses and introduce a dead band. We choose intracavity pumped OPO because the high intracavity power of a laser provides more gain for the OPO, since the nonlinear effect is proportional to the square of pump intensity. Ti:Sapphire laser is chosen as the pump laser source because it provides both high power and short pulse duration. We will demonstrate two types of the experimental setups in section 5.1, and evaluate their performance and results. The systematic alignment procedure of the OPO is outlined in section 5.2. In section 5.3 we discuss the instability issue of the system and attempts to eliminate it by improving the mechanical, thermal conditions as well as inserting nonlinear losses as energy limiter.

5.1 Experimental setups

For an intracavity pumped OPO, the nonlinear crystal is set inside the pump laser as well as two curved mirrors that are AR coated for the pump wavelength and HR coated for the OPO wavelength. The nonlinear crystal we use for the OPO is a periodically poled lithium niobate (PPLN) for its high gain and moderate group velocity mismatch. The 1 mm long PPLN crystal is Brewster cut to avoid the feedback to the pump laser. The PPLN is 1 mm thick and 7 mm high, with grating periods stacked at different height along the mm dimension (every mm): $20.1\mu\text{m}$, $20.4\mu\text{m}$, $20.7\mu\text{m}$, $21\mu\text{m}$, $21.4\mu\text{m}$, $21.6\mu\text{m}$, and $22.9\mu\text{m}$. It gives us the freedom of selecting one of these gratings depending on the pump wavelength and desired signal wavelength.

The cavity length of the OPO needs to be a multiple of the laser cavity for synchronization purposes. The realization of IPI requires two pulses per cavity operation, and there are two approaches to it: one is to simply make the OPO twice as long as the Ti:Sapphire cavity (section 5.1.2); the other is to make them equal and generate two OPO pulses when the pump pulse passes through the PPLN twice (section 5.1.1).

5.1.1 OPO and pump cavities of equal length

The first design is to build a cavity of the same length of the laser cavity. The pump pulse passes through the NL crystal twice to create two pulses. The advantage of this design is that the OPO cavity remains of a reasonable size. The disadvantage however is that the two signal pulses created are usually not of the same intensity, and there is gain competition between them that brings instability to the OPO operation [35, 33].

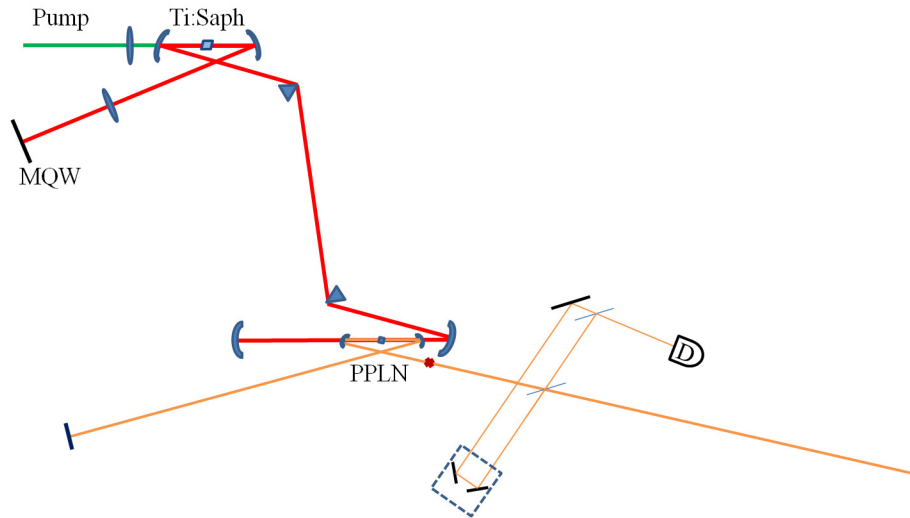


Figure 5.1: The sketch of an intracavity OPO pumped by a Ti:Sapphire with equal cavity length. The focal length of the lens for 532 nm is $f = 6$ cm, and the radius of curvature of the two curved mirrors is $R = 7.5$ cm. The beam is focused onto the MQW by a lens with focal length $f = 8$ cm. The radii of curvature of the two curved mirror at the end of Ti:Sapphire cavity are both $R = 10$ cm. The radii of curvature for both of the curved mirrors for OPO cavity are $R = 5$ cm.

The Ti:Sapphire laser is pumped by a Millennia CW 532 nm laser with maximum output power of 10 W. The full sketch of Ti:Sapphire and OPO cavities is shown in Fig. 5.1. The Ti:Sapphire laser is mode-locked with a MQW mirror in one end of the cavity. A pair of SF10 prisms separated by about 33 cm provide tunable negative group velocity dispersion for compensation of the large positive dispersion of other intracavity components (PPLN, Ti:sapphire). The Brewster cut Ti:Sapphire crystal is 4 mm long and mounted on a water cooled copper mount. The focal length of the lens for 532 nm is $f = 6$ cm, and the radii of curvature of the two curved mirrors is $R = 7.5$ cm. The intracavity beam is focused onto the MQW by a $f = 8$ cm focal lens.

A challenge in the alignment of any Ti:sapphire mode-locked laser is that the maximum output power does not coincide with the lowest threshold configuration.

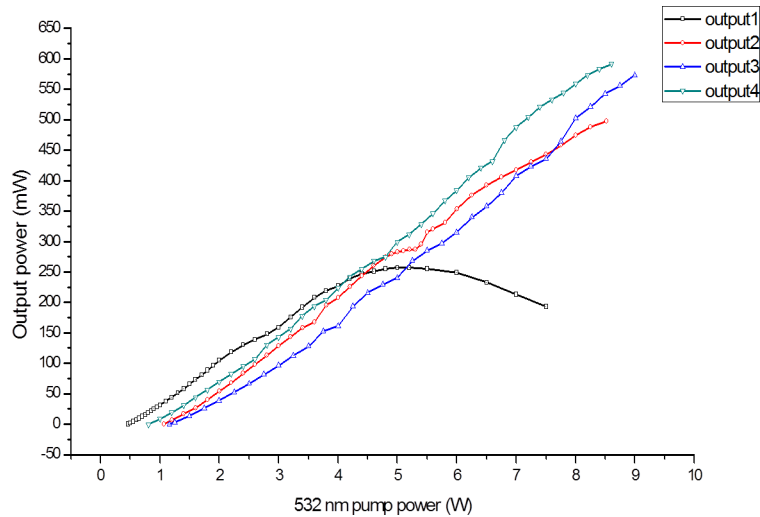


Figure 5.2: The laser outputs taken from a 2% OC are plotted as a function of pump power for different pumping optics alignments.

This is because, even though MQW are used for initiating the mode-locking, Kerr lensing, which may be negligible near threshold, still plays an important role in the laser operation. Unlike situations where the mode-locked laser is just used as an oscillator seed for an amplifier, achieving the highest possible output power is essential for pumping the OPO. To optimize the operation, before inserting the prism pair, the laser output taken from a 2% OC was recorded as a function of pump power as shown in Fig. 5.2. Various parameters (distance between curved mirrors around the gain crystal, position of the Ti:sapphire crystal, pump lens position) were varied in order to optimize the pumping geometry for maximum output *at high pump power*. Up to 600 mW output power has been measured for 8.5 W pump power. The laser is then mode locked after the prism pair was inserted, and the lasing wavelength was found to be tunable from 775nm to 810nm. The shortest pulse duration measured at this stage is around 90 fs at a center wavelength of 790 nm.

In order to pump the OPO inside the laser cavity, we extend the Ti:Sapphire cavity by adding two curved mirrors. The folding mirror that focuses the laser beam

into the PPLN crystal has a radius of curvature of 10 cm, and the end mirror for Ti:Sapphire cavity has the same $R = 10$ cm. The PPLN is located about 5 cm away from the folding mirror and 10 cm away from the end mirror. The distance between these two curved mirrors is measured to be between 13.7 cm and 15 cm in order for the laser to lase. The radii of curvature for both of the curved mirrors for the OPO cavity is $R = 5$ cm. One of them needs to be cut in half for laser beam clearance (the angles of incidence being kept small to minimize astigmatism). These curved mirrors are high reflection coated for OPO wavelength from 1060 nm to 1260 nm, and anti-reflection coated for pump wavelength around 790 nm. The PPLN crystal needs to be heated to avoid photorefractive damage. It is mounted on a copper plate with heating wires around the copper plate and a thermocouple attached to the mount. A temperature controller is used to set the temperature of PPLN to about 10 °C above room temperature to stabilize the temperature.

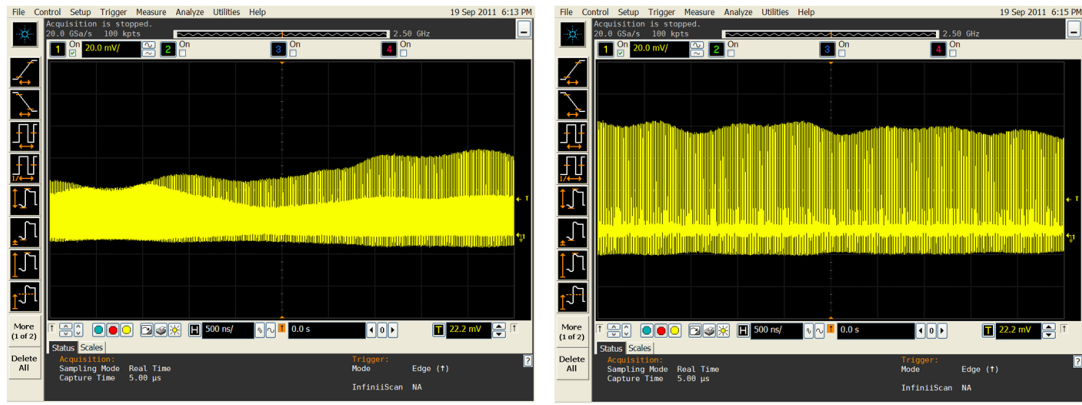
After all the elements are present in the laser cavity we measure the pulse duration by auto-correlation. A minimum of 170 fs pulse duration FWHM could be achieved by adjusting the amount of glass passing through the prisms. This relatively large pulse duration is due to the very large positive dispersion of other elements of the cavity (such as the PPLN) that we are trying to compensate with the pair of very dispersive SF10 prisms. There was no attempt to compensate the third order dispersion. The measured 170 fs are consistent with the measurement of the laser spectrum, which suggest we are close to the time-bandwidth product limit of a Gaussian pulse. The repetition rate of the laser is 108.71 MHz, which corresponds to a cavity length of 1.38 m.

The length of the two arms of the OPO cavity are chosen based on the timing of two passages of the laser pump through the PPLN. Let us define the distance between PPLN and Ti:Sapphire end mirror to be d_1 , the length of left and right arms of the OPO to be d_2 and d_3 . There is a simple relation $d_3 - d_2 = 2 \cdot d_1$ that

equally spaces the two pulse trains in the OPO cavity, i.e. when one pulse is at the end of the OPO cavity, the other pulse is at the other end. Therefore with the same cavity length, the OPO repetition rate is twice that of the Ti:Sapphire's.

Alignment of the OPO cavity is a difficult procedure, mainly because there is no detectable fluorescence signal that could be used as a guide, and because of the stringent length requirement ($\pm 1\mu\text{m}$) for OPO cavity. We managed to align the OPO cavity using the procedure detailed in section 5.2. Strong depletion up to 32.5% of the pump has been observed. The pump wavelength is modified by the OPO pulse. The beam is apertured with razor blades between two prisms to prevent any shift of the pump wavelength when the OPO is working. The wavelength of the OPO signal is tunable from 1060nm to 1180nm only by adjusting the cavity length. The broadest spectral bandwidth measured with the OPO is around 20 nm at 1160 nm, which indicates a very short signal pulse. A beamsplitter is inserted in the OPO cavity to extract the two pulses with a proper delay line the pulses meet at the detector for beatnote measurements. However as shown in Fig. 5.3 (a) the pulse train of the OPO is unstable. The short pulse duration, compounded with the instability problem, exacerbated the difficulty in overlapping two pulses in time. We also tried to overlap the two second harmonics (SH) of the OPO signals which leaks out through the two focusing mirrors, since this SH inherits the phase information of the signal. However further stabilization was still needed before an IPI response could be demonstrated.

It is also difficult in the experiment to realize conditions where the two OPO pulses have equal intensity. Naturally the OPO pulses which are generated from the first laser pass are stronger, however the intensity of the two generated OPO pulses can be equalized or flipped with cavity alignment. The OPO is less stable (i.e. more unstable) when the two pulses are of more or less of equal intensity (Fig. 5.3 (b)). This is because:



(a)

(b)

Figure 5.3: OPO pulse train monitored on a digital oscilloscope. The capture time on each screen shot is $5 \mu\text{s}$. These two screen shots are taken within 2 minutes. (a) two pulse operation, (b) one pulse operation

1. The laser pulse is attenuated after one passage.
2. The laser pulse goes through slightly different passages when it comes and goes.
3. There is gain competition between two OPO pulses.

5.1.2 OPO cavity of twice the length of the pump cavity

A straightforward solution to the previous design is to set the OPO cavity length twice that of the laser. This solution however imposes a 3 m long OPO cavity, which brings more mechanical instabilities. It is therefore desirable to build a laser cavity as compact as possible. The final design is sketched in Fig. 5.4.

The Ti:Sapphire laser is pumped by the Millennium CW 532 nm laser with maximum output power of 10 W. The Brewster cut Ti:Sapphire crystal is replaced by a 3 mm long one with a more stable mount. The focal length of the lens for 532 nm is $f = 4 \text{ cm}$, and the radii of curvature of the two curved mirrors is $R = 5 \text{ cm}$. We also

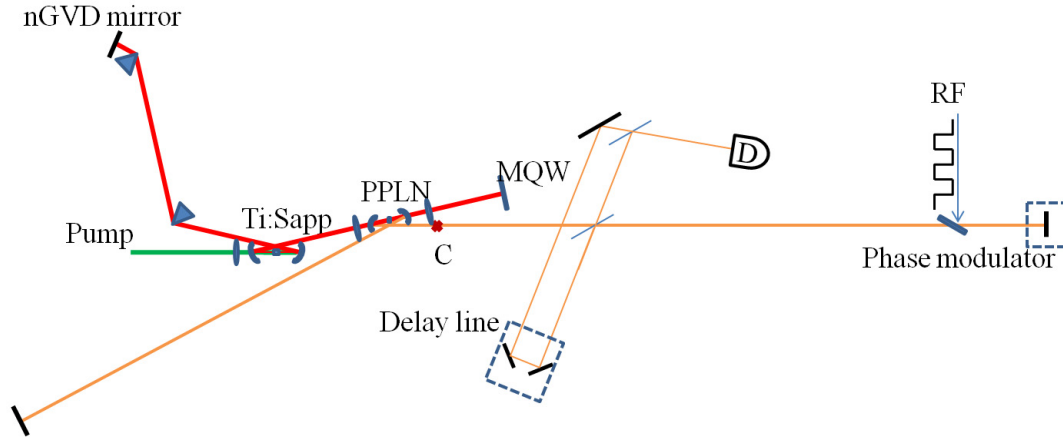


Figure 5.4: The sketch of an intracavity OPO pumped by a Ti:Sapphire of half cavity length. The focal length of the lens for 532 nm is $f = 4$ cm, and the radii of curvature of the two curved mirrors is $R = 5$ cm. The pair of SF10 prisms is separated by 24 cm. A negative GVD mirror (-150 fs^2) is used as the end mirror. The laser beam is focused into the PPLN crystal by a lens with focal length $f = 3.6 \text{ cm}$. It is then focused into MQW by another lens with focal length $f = 3.6$ cm. The radii of curvature for both of the curved mirrors of the OPO cavity is $R = 5$ cm.

replace the MQW by a more recent one. We use the same pair of SF10 prisms, but use a negative GVD mirror (-150 fs^2) as the end mirror to shorten the separation of prism pair to 24 cm. The laser beam is focused into the PPLN crystal by a lens of focal length $f = 3.6 \text{ cm}$. It is then focused into the MQW by another lens of focal length $f = 3.6$ cm. The radii of curvature for both curved mirrors focusing into the OPO crystal is $R = 5$ cm. This setup is the most compact possible cavity that can be built in our lab with current optics and mounts.

The Ti:Sapphire laser is mode-locked at 780 nm, with the shortest pulse duration $\tau_p = 80$ fs measured through auto-correlation. The shorter pulse duration is the result of a better cavity geometry that enabled us to optimize the spacing between the two prisms. The repetition rate of the Ti:Sapphire cavity is 180 MHz, which corresponds to a cavity length of 83 cm. The intracavity power is estimated to be about 6 W. Therefore the energy per pulse is about 30 nJ. This laser configuration

is good for the long OPO (twice the pump cavity) cavity.

The spacing between two curved mirrors of the OPO cavity has a narrow range of 5.15-5.2 cm for the OPO to lase. It is still possible to generate two pulses when the laser pulse passes through the PPLN twice as in section 5.1.1. We can adjust the alignment to favor one of them. There are however always two identical pulses per cavity in the OPO cavity pumped by the Ti:Sapphire pulse train, separated by the Ti:Sapphire cavity round trip time. There is no restriction on the symmetry of the arms of the OPO cavity, only the crossing point of the OPO pulses has to be in air.

The output power of the OPO signal is reduced, as compared to the configuration with equal cavity lengths. The depletion of the pump is up to 14%, measured by the power meter for pump. The pump wavelength is also modified by the OPO pulse. The overall performance of this setup is better than the previous one, although the OPO pulse train is still unstable, and the fluctuation are fed-back to the Ti:Sapphire pulse train as shown in Fig. 5.5. The delay line is adjusted to overlap the two pulses onto a slow IR detector. The proper delay was found when the signal on the detector becomes noisier. However no beat note is observed by translating the delay line yet.

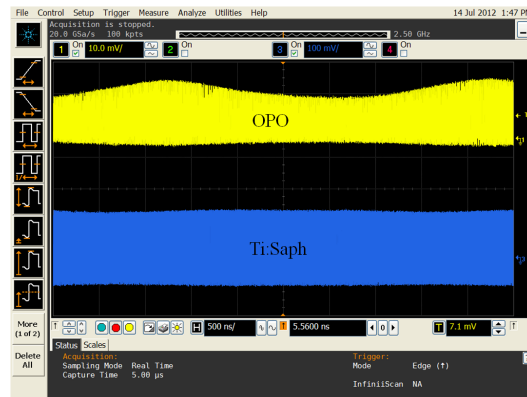


Figure 5.5: OPO and Ti:Sapphire pulse trains monitored on a digital oscilloscope. The yellow trace on the top is OPO, and the blue trace on the bottom is Ti:Sapphire. The capture time for the screen shot is $5 \mu\text{s}$. The intensity of the Ti:Sapphire is much higher so the two pulse trains are shown at different amplitude scales.

Beat note measurement

A phase modulator is inserted in the OPO cavity. It is a 1 mm thick lithium niobate plate oriented near brewster angle to minimize the loss to the OPO. The RF voltage applied on the phase modulator is triggered by the repetition rate signal from the Ti:Sapphire cavity, then divided by two and sent to an electronic delay line. The beat note can be measured when the OPO cavity is tuned off resonance, at the lasing threshold. We vary the amplitude of the voltage and measure the beat frequency correspondingly. The beat frequency is plotted as a function of the amplitude of the voltage in Fig. 5.6. A linear fit to the data calculates the slope of 3.25Hz/mV, and can be converted to 2.1×10^{-7} radian/mV and finally 3.9×10^{-11} refractive index change for this LiNbO₃ phase modulator. From Fig. 5.6 (b) we can tell that there is no measurable deadband for the system, although there is a substantial noise.

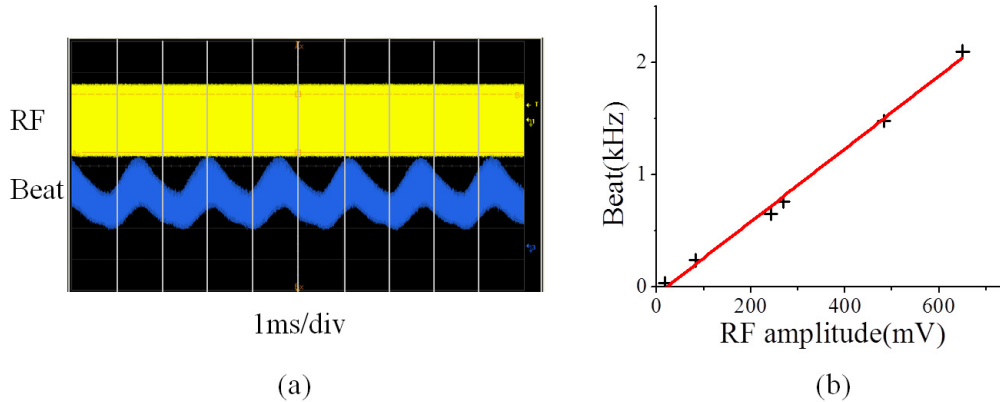


Figure 5.6: Figure (a) Shows an oscilloscope recording at 1 ms/division of (top) the RF square wave at 90 MHz, and (bottom) a 0.64kHz beat note recorded on the detector D . (b) Shows a plot of the beat note versus voltage applied to the phase modulator.

5.2 Alignment procedure

The alignment of the OPO cavity is extremely difficult, as there is no detectable fluorescence, and the cavity length has to be matched within microns. We developed a standard alignment procedure, injecting the beam from a CW diode laser at the designed signal wavelength, which is $1.2 \mu\text{m}$. The alignment laser beam is sent to the laser cavity using a pellicle beam splitter. The polarization of the alignment laser has to be the same as that of the OPO polarization, which is horizontal for the PPLN. The leakage of Ti:Sapphire laser acts as a rough alignment so that the alignment beam at $1.2 \mu\text{m}$ should overlap with it as close as possible. Several iris shutters are placed in the optical path to check the multiple reflection of alignment laser on the OPO end mirrors. By adjusting the angles of the two end mirrors of the OPO, we make the OPO cavity resonant for $1.2 \mu\text{m}$. The resonance of the cavity is monitored by an infrared camera. The interference pattern of the alignment laser serves as indicator for the OPO cavity alignment. Thereafter, the OPO cavity *length* has to be adjusted by a translation stage to find the synchronization length (in the order of 1 micron, depending on the strength of the parametric gain). If everything is appropriate, a blink of the signal pulse will indicate the OPO lasing.

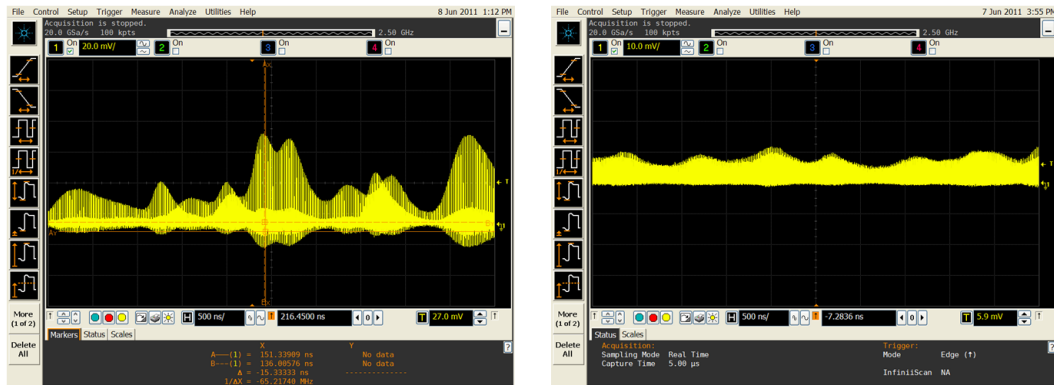
5.3 Instability issues, and attempts to eliminate them

The most important requirement of the OPO is a stable pulse train. However the instability has always been a dominant feature in every design of the intracavity OPO. Several attempts have been made to eliminate the fluctuation:

1. Both laser and OPO cavities are covered by plastic boxes to reduce air turbu-

lence. There is improvement to the stability.

2. The mechanical vibration of the cavity is indeed improved by changing to high quality stable mounts and translation stages in setup 5.1.2.
3. The temperature control of the PPLN crystal is switched from a pulsed mode to continuous mode, which eliminated the unwanted noise of the current cycle.
4. For dispersion compensation purpose we added a pair of prisms in the OPO cavity in setup 5.1.1, however the prism separation is limited and shows no big difference. The GVD contribution from specially coated mirrors (multiple wavelengths) is not clear.
5. We also tried to add more SHG loss to the cavity in setup 5.1.1. We inserted a 5 mm long lithium triborate (LBO) crystal inside the OPO cavity to create additional nonlinear loss for two pulses per cavity operation. We use two lenses of 5 cm focal length to focus the signal pulse into the LBO crystal. When strong SH was generated from LBO, an improvement in stability was observed.



(a)

(b)

Figure 5.7: Double OPO pulse train monitored on a digital oscilloscope. The capture time for each screen shot is $5 \mu\text{s}$. (a) without SHG from LBO, (b) with SHG from LBO - the two pulse trains can hardly be distinguished.

As shown in Fig. 5.7, the comparison between pulse trains with (right) or without (left) LBO are given. Note that the case without LBO shows clearly the gain competition between the two pulses, i.e. the amplitude fluctuations are out of phase, while in the case with LBO the reduced amplitude fluctuations are almost in phase. Further experiments show that even with weak SHG the two amplitude of two OPO pulse operation is locked together, however the beat note is not observed.

6. Active stabilization is realized by attaching one of the OPO end mirrors to a piezoelectric ceramics. The signal wavelength depends on the OPO cavity length. We can feedback this wavelength shift to the piezo to stabilize the cavity length.

First we extracted the frequency doubling of signal and sent it through a pair of prisms to a quadrant detector. This is to convert the wavelength shift into position shift. And then we sent the signal from quadrant detector to a DC amplifier and used the output to drive the piezo. However this feedback loop did not work well because the relatively low sensitivity of the quadrant detector. We then tried to split the same SH beam into two identical avalanche photodiodes, each of which detects opposite slope of the beam. A differential amplifier was built to amplify the difference between these two detectors, which gives the information of beam position. We close the loop with the DC amplifier and a piezo driver.

This setup has been tested when the loop was open and an external modulating voltage is added to the piezo. Fig. 5.8 demonstrates the effect on the OPO pulse train at different modulating frequency. The yellow trace is the OPO pulse train at 1200nm, the green trace is the modulating voltage on the piezo, and blue trace is the signal obtained from the differential amplifier followed by DC amplifier. As we can see in Fig. 5.8 the differential signal (blue) is

180° out of phase with respect to the PZT voltage (green), indicating that the differential signal can be used for stabilization.



Figure 5.8: OPO pulse train and electrical signals monitored on a digital oscilloscope. Yellow trace is the OPO pulse train at 1200nm, green trace is the modulating voltage on the piezo, and blue trace is the signal obtained from the differential amplifier followed by DC amplifier. The piezo is modulated at different frequencies 11 Hz (a) and 22 Hz (b).

The attempts to stabilize the OPO pulse train are effective to some extent, however the fluctuations are never fully eliminated. The fact that a stable operation for beat note measurement can only be achieved at low OPO intracavity power, when the feedback from OPO to pump is minimized, indicates a complex coupling between the Ti:Sapphire and OPO cavities. Therefore a theoretical model containing the nonlinear interactions in the PPLN has been elaborated and presented in Chapters 6 and 7. This model leads to an understanding of the mechanisms that affect the stability of the system, and finds conditions leading to a stable operation.

Chapter 6

Simulation of second order nonlinear interactions

In this chapter we simulate the second order nonlinear (NL) interactions taking place inside the nonlinear crystal of the OPO. Coupled NL wave equations consisting of all the interacting electric fields are solved in the frequency domain for ultrafast pulses in section 6.1. The simulation results of NL interaction propagating through NL crystal for SHG, SFG and DFG are given in section 6.2. In section 6.3 the nonlinear processes have been studied under different experimental conditions to demonstrate various effects including, but not limited to pulse shaping, chirp, wavelength tuning. In particular, phase mismatched processes are emphasized as they introduce large effective nonlinear refractive indices (creating self-phase- and cross-phase-modulation) that result in a coupling of intensity and phase instabilities.

6.1 Nonlinear equations in frequency domain

The nonlinear crystal used in the OPO cavity is a periodically poled lithium niobate (PPLN) crystal, designed for quasi phase matched difference frequency generation (DFG) $\omega_s = \omega_p - \omega_i$ (where ω_p is the pump frequency, resonating in the pump cavity, ω_i the idler, and ω_s the signal that is made to resonate in the OPO cavity). It is well known [8, 25] that, in synchronously pumped lasers, small fluctuations in length of the pump or signal cavity result in large wavelength shift of the signal, hence a departure from the design phase matched condition. In addition there are other phase mismatched processes that take place in the nonlinear crystal. These are second harmonic generation (SHG) of the pump, signal and idler wave, and sum frequency generation (SFG) involving any combination of these waves. These mismatched processes have little impact on the net gain in the crystal, but affect the phase of the various pulses through a large self-phase modulation and cross phase modulation. For instance, the pump intensity is sufficient to create a second harmonic, which, while not representing a significant loss, does create a phase modulation proportional to the intensity (self-phase modulation), hence can be represented by a nonlinear index. The sum of the pump and signal frequency create a cross-phase as well as a self-phase modulation. These effects are very important in the context of the intracavity pumped OPO, as they are responsible for a coupling between phase and amplitude instabilities. More generally, the complete modeling of the interaction between pump and signal inside the nonlinear crystal presented here is warranted in a more general context, in view of the growing importance of optical parametric amplification in the design of high power ultrashort lasers [21].

After establishing the mathematical model, we will take our laser as numerical example, with pump frequency centered at 770 nm, the signal at 1140nm, the idler at $\omega_i = \omega_p - \omega_s$, i.e. 2372nm, the second harmonic of the pump at $\omega_2 = 2\omega_p$, i.e. 385nm, the sum frequency of pump and signal at $\omega_3 = \omega_s + \omega_p$, i.e. 460nm, the second

harmonic of the signal at $\omega_4 = 2\omega_s$, i.e. 570nm, and the sum frequency of pump and idler at $\omega_5 = \omega_i + \omega_p$, i.e. 581nm. These processes are sketched in the diagram of Fig. 6.1. For sake of clarity, and better understanding of the physics involved, we

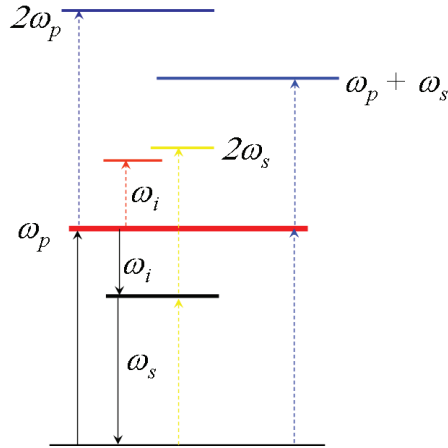


Figure 6.1: Diagram illustrated the different sum and difference frequency processes involved in the intracavity pumped parametric oscillator. Only the optical parametric process $\omega_p = \omega_s + \omega_i$ is phase matched.

will limit the simulation discussed below to the most intense fields. Therefore, we will ignore the last two frequencies ω_4 and ω_5 which involve weaker fields.

To include the frequency dependence of the index of refraction to all orders for short pulses, the second-order nonlinear frequency generations in the PPLN crystal is modeled in the frequency domain. Maxwell's propagation equations for the five fields, written in the frequency domain, can be separated within the reasonable assumption that the Fourier spectra of the pulses at the five frequencies do not overlap. This derivation was first introduced to model phase matched second harmonic generation with birefringent crystals [5, 4]. This model is adapted to the problem of quasi-phase-matched frequency generation, with all beams collinear, and polarized linearly along the optical z axis of the nonlinear crystal (PPLN).

6.1.1 Summary of the theoretical model

To analyze all mixing process, taking into account dispersion to all orders, we solve Maxwell's propagation equations in the frequency domain [12]:

$$\left[\frac{\partial^2}{\partial z^2} + \Omega^2 \mu \epsilon(\Omega) \right] \tilde{\mathbf{E}}(\Omega, z) = -\mu_0 \Omega^2 \tilde{\mathbf{P}}^{NL}(\Omega, z). \quad (6.1)$$

where

$$\epsilon(\Omega) = \epsilon_0 (1 + \chi^{(1)})(\Omega). \quad (6.2)$$

is the relative dielectric constant, and $\tilde{\mathbf{E}}(\Omega, z)$ is the total electric field. The second order nonlinear polarization is made of products of pairs of fields, *in the time domain* $P_i^{NL}(\Omega, z) = \epsilon_0 \left[\sum_j \chi_{jk}^{(2)} E_j E_k + \sum_\ell \chi_{i\ell}^{(2)} E_i E_\ell \right]$ where the first terms are sum frequency (SF) process, and the second terms difference frequency (DF). Substituting for each field, in the frequency domain, $E_i(\Omega, z) = \tilde{\mathcal{E}}(\Omega, z) \exp[-ik_i(\Omega)z]$, in Eq. (6.1), we find that the nonlinear interaction equations can be written in the simple elegant form:

$$\frac{\partial \tilde{\mathcal{E}}_i(\Omega)}{\partial z} = - \sum_{m=1,3,5} \frac{ik_i(\Omega)}{4n_i^2} \chi_{jk}^{(2)} \left\{ \left[\tilde{\mathcal{E}}_j(\Omega, z) e^{-ik_j(\Omega)z} \right] \star \left[\tilde{\mathcal{E}}_k(\Omega, z) e^{-ik_k(\Omega)z} \right] \right\} e^{ik_i z} \frac{\sin(mk_g z)}{m} \quad (6.3)$$

where the symbol “ \star ” represents a convolution for DF, and a correlation for SF, and $k_g = 2\pi/\Lambda$ where Λ is the period of the PPLN grating.

6.1.2 Outline of the numerical approach

In Eq. (6.3), elegance is at the cost of numerical convenience and accuracy. It is unpractical to have a mesh that covers all optical frequencies with MHz steps, as required by Eq. (6.3). It is desirable to shift all spectra to zero frequency, and deal only

with the spectral components that are covered by the pulses. Such a shift removes, in the time domain, the underlying carrier frequencies and leaves only the envelopes of each pulse to be changed along propagation. Since these spectra represent pulses propagating at the group velocity of the respective pulses, in the laboratory frame, the phase factor of each Fourier component will take very large values with increasing distance z , making numerical computation unnecessarily challenging. We chose therefore a retarded frame of reference, propagating at the group velocity of one of the pulses. Since the pump pulse repetition rate is the primary clock of this system, it is natural to chose a frame of reference for the five pulses moving at the pump group velocity.

The total electric field $\tilde{\mathbf{E}}(\Omega, z)$ in Eq. (6.1) includes the pump field centered at ω_p , the signal field centered at ω_s , the idler centered at ω_i , the second harmonic field centered at ω_2 , and the sum field centered at ω_3 . In the nonlinear part of the polarization $\tilde{\mathbf{P}}^{NL}$ we consider only the $\chi^{(2)}$ terms that are associated with the five interacting waves. Each of these terms has to fulfill the wave equation separately. We will assume that, for each nonlinear mixing, the second order susceptibility $\chi^{(2)}$ is frequency independent within the interaction bandwidth. Here we only consider three of the processes shown in Fig. 6.1

$$\begin{aligned}
 \chi_1^{(2)} &: \omega_p = \omega_s + \omega_i \\
 \chi_2^{(2)} &: \omega_2 = \omega_p + \omega_p \\
 \chi_3^{(2)} &: \omega_3 = \omega_p + \omega_i
 \end{aligned} \tag{6.4}$$

The nonlinear polarizations can be decomposed in five contributions, respectively centered at ω_p , ω_s , ω_i , ω_2 and ω_3 :

$$\begin{aligned}
 \tilde{\mathbf{P}}_p^{NL}(t, z) &= \epsilon_0 \left[\chi_1^{(2)} \tilde{E}_s(t, z) \tilde{E}_i(t, z) + \chi_2^{(2)} \tilde{E}_p^*(t, z) \tilde{E}_2(t, z) + \chi_3^{(2)} \tilde{E}_s^*(t, z) \tilde{E}_3(t, z) \right] \\
 \tilde{\mathbf{P}}_s^{NL}(t, z) &= \epsilon_0 \left[\chi_1^{(2)} \tilde{E}_p(t, z) \tilde{E}_i^*(t, z) + \chi_3^{(2)} \tilde{E}_p^*(t, z) \tilde{E}_3(t, z) \right]
 \end{aligned}$$

$$\begin{aligned}
 \tilde{\mathbf{P}}_i^{NL}(t, z) &= \epsilon_0 \chi_1^{(2)} \tilde{E}_p(t, z) \tilde{E}_s^*(t, z) \\
 \tilde{\mathbf{P}}_2^{NL}(t, z) &= \epsilon_0 \chi_2^{(2)} \tilde{E}_p(t, z) \tilde{E}_p(t, z) \\
 \tilde{\mathbf{P}}_3^{NL}(t, z) &= \epsilon_0 \chi_3^{(2)} \tilde{E}_p(t, z) \tilde{E}_s(t, z).
 \end{aligned} \tag{6.5}$$

Most of the equation manipulations and substitutions required for numerical computation are dictated by numerical rather than physical considerations. The details of those can be found in Appendix A.

The final expressions to be solved numerically are using spectral fields limited to a range of frequencies corresponding to the bandwidth at each central frequency, and including the wave-vector with its exact frequency dependence:

$$\begin{aligned}
 \mathcal{P}(\Delta\Omega) &= \tilde{\mathcal{E}}_p(\Delta\Omega) e^{-ik_p(\Delta\Omega)z} \\
 \mathcal{S}(\Delta\Omega) &= \tilde{\mathcal{E}}_s(\Delta\Omega) e^{-ik_s(\Delta\Omega)z} \\
 \mathcal{I}(\Delta\Omega) &= \tilde{\mathcal{E}}_i(\Delta\Omega) e^{-ik_i(\Delta\Omega)z} \\
 \mathcal{N}(\Delta\Omega) &= \tilde{\mathcal{E}}_2(\Delta\Omega) e^{-ik_2(\Delta\Omega)z} \\
 \mathcal{M}(\Delta\Omega) &= \tilde{\mathcal{E}}_3(\Delta\Omega) e^{-ik_3(\Delta\Omega)z}.
 \end{aligned} \tag{6.6}$$

The coupled equations for these newly defined field amplitudes are similar to the general set of equations (6.3):

$$\frac{\partial \tilde{\mathcal{E}}_p(\Delta\Omega)}{\partial z} = A_p \left[\chi_1^{(2)} (\mathcal{S} * \mathcal{I}) + \chi_2^{(2)} (\mathcal{P} * \mathcal{N}) + \chi_3^{(2)} (\mathcal{S} * \mathcal{M}) \right] (\Delta\Omega) e^{ik_p(\Delta\Omega)z} \tag{6.7}$$

$$\frac{\partial \tilde{\mathcal{E}}_s(\Delta\Omega)}{\partial z} = A_s \left[\chi_1^{(2)} (\mathcal{I} * \mathcal{P}) + \chi_3^{(2)} (\mathcal{P} * \mathcal{M}) \right] (\Delta\Omega) e^{ik_s(\Delta\Omega)z} \tag{6.8}$$

$$\frac{\partial \tilde{\mathcal{E}}_i(\Delta\Omega)}{\partial z} = A_i \chi_1^{(2)} (\mathcal{S} * \mathcal{P}) (\Delta\Omega) e^{ik_i(\Delta\Omega)z} \tag{6.9}$$

$$\frac{\partial \tilde{\mathcal{E}}_2(\Delta\Omega)}{\partial z} = A_2 \chi_2^{(2)} (\mathcal{P} * \mathcal{P}) (\Delta\Omega) e^{ik_2(\Delta\Omega)z} \tag{6.10}$$

$$\frac{\partial \tilde{\mathcal{E}}_3(\Delta\Omega)}{\partial z} = A_3 \chi_3^{(2)} (\mathcal{S} * \mathcal{P}) (\Delta\Omega) e^{ik_3(\Delta\Omega)z}. \tag{6.11}$$

The operator $*$ describes a convolution and \star a cross-correlation. The refractive indexes $n(\Omega)$ are required to calculate $k_{p,s,i,2,3}(\Delta\Omega)$. They are obtained from a Sellmeier equation [36]. The coefficients A_i are defined in Appendix A.

6.1.3 Quasi-Phase Matching

A treatment of pulse evolution in phase matching crystals that can be found in one of the most used textbook is unfortunately incorrect. A correct derivation is given in this section with details in Appendix A.

The susceptibility $\chi_n^{(2)}$ of the periodically poled lithium niobate (PPLN) alternates its sign at every poled domain “n”.

$$\chi_n^{(2)} = \chi^{(2)}(-1)^n, \quad (6.12)$$

where n is the index of poled domain. The expression can be decomposed into a Fourier series:

$$\chi^{(2)} = \frac{4\chi^{(2)}}{\pi} \sum_{m=1,3,5\dots}^{\infty} \frac{1}{m} \sin\left(\frac{2\pi mz}{\Lambda}\right) = \frac{2\chi^{(2)}}{i\pi} \sum_{m=1,3,5\dots}^{\infty} \frac{1}{m} \left[e^{imk_g z} - e^{-imk_g z} \right], \quad (6.13)$$

where Λ is the grating period of PPLN crystal, and $k_g = \frac{2\pi}{\Lambda}$. The grating period is chosen such that the poling is reversed before conversion back to the pump occurs, for the wavelength at the center of each pulse:

$$\frac{\delta k^{(0)} \Lambda}{2} = \pi \quad (6.14)$$

The quasi-mismatch of the wave vector is:

$$\delta k(\Delta\Omega, \Delta\Omega') = k_s(\Delta\Omega') + k_i(\Delta\Omega - \Delta\Omega') - k_p(\Delta\Omega) \pm mk_g, \quad (6.15)$$

where $m = 1, 3, 5 \dots \infty$. The set of equations (6.11) for the DFG needs to be rewritten as a sum over $m = 1, 3, 5 \dots \infty$:

$$\frac{\partial \tilde{\mathcal{E}}_p(\Delta\Omega)}{\partial z} = A_p \sum_{m=1,3,5\dots}^{\infty} \left[\chi_1^{(2)} (\mathcal{S} * \mathcal{I}) + \chi_2^{(2)} (\mathcal{P} * \mathcal{N}) + \chi_3^{(2)} (\mathcal{S} * \mathcal{M}) \right] (\Delta\Omega) e^{ik_p(\Delta\Omega)z} \frac{\sin(mk_g z)}{m} \quad (6.16)$$

$$\frac{\partial \tilde{\mathcal{E}}_s(\Delta\Omega)}{\partial z} = A_s \sum_{m=1,3,5\dots}^{\infty} \left[\chi_1^{(2)} (\mathcal{I} * \mathcal{P}) + \chi_3^{(2)} (\mathcal{P} * \mathcal{M}) \right] (\Delta\Omega) e^{ik_s(\Delta\Omega)z} \frac{\sin(mk_g z)}{m} \quad (6.17)$$

$$\frac{\partial \tilde{\mathcal{E}}_i(\Delta\Omega)}{\partial z} = A_i \sum_{m=1,3,5\dots}^{\infty} \chi_1^{(2)} (\mathcal{S} * \mathcal{P}) (\Delta\Omega) e^{ik_i(\Delta\Omega)z} \frac{\sin(mk_g z)}{m} \quad (6.18)$$

$$\frac{\partial \tilde{\mathcal{E}}_2(\Delta\Omega)}{\partial z} = A_2 \sum_{m=1,3,5\dots}^{\infty} \chi_2^{(2)} (\mathcal{P} * \mathcal{P}) (\Delta\Omega) e^{ik_2(\Delta\Omega)z} \frac{\sin(mk_g z)}{m} \quad (6.19)$$

$$\frac{\partial \tilde{\mathcal{E}}_3(\Delta\Omega)}{\partial z} = A_3 \sum_{m=1,3,5\dots}^{\infty} \chi_3^{(2)} (\mathcal{S} * \mathcal{P}) (\Delta\Omega) e^{ik_3(\Delta\Omega)z} \frac{\sin(mk_g z)}{m}. \quad (6.20)$$

A good approximation to a square wave is obtained by limiting the sum up to the value of $m = 5$. Dividing by Λ , we note that Eq. (A.19) is equivalent to the phase matching condition:

$$-k_s^{(0)} - k_i^{(0)} + k_p^{(0)} - k_g = 0 \quad (6.21)$$

The set of equations (6.11) have to be integrated over the thickness of the PPLN crystal, given a set of initial fields at $z = z_0$. As detailed in the Appendix A a final transformation of the numerical solution has to be performed in order to account for the effect of dispersion, before a final inverse Fourier transform is performed to obtain the fields in amplitude and phase in time domain.

The numerical simulation is realized in Fortran. The code is attached in Appendix C.

6.1.4 Comments on the simulation

To the best of our knowledge this is the first model to simulate the evolution of short pulses experiencing second-order nonlinear processes with quasi-phase-matching, taking dispersion into account to all orders. It applies to pulse duration as short as a few optical cycles, the only limitation being that the spectra of the different pulses (signal, pump and idler) may not overlap.

The model uses a plane wave approximation, and therefore does not take into account diffraction. It has been established that, in the case of phase matched nonlinear crystals, the optimum conversion is achieved when the crystal length exceeds by a factor of 3 the Rayleigh range (see reference [7] page 121). This limitation however does not apply to us: the group velocity dispersion of the OPO crystal imposes a length of less than 1 mm, which in all practical cases will be much less than the Rayleigh range.

Introducing a spatial dimension may be important for other OPO applications, such as extremely broadband amplification using an array of crystal in the Fourier plane of a grating arrangement [21]. The large phase modulations that we observe due to non-phase matched conditions will result in spatial chirp and beam deflections in the spatial domain.

6.2 Simulation results

Although the theoretical discussion in the previous section is primarily to simulate the behavior of short pulses around 100 fs, we will start with the simpler case of continuous waves, in order to gain a better physical understanding, and be able to compare the numerical results with analytical solutions. In the present program, the cw solution corresponds to changing the initial pulse duration to 100ns so that the

field amplitude is constant over the range of interest. For all the simulation presented in this section, the nonlinear crystal length is 1mm, and the initial pulse shape is Gaussian with FWHM of 100 fs.

6.2.1 SHG only

Here we start the comparison of numerical and analytical results for SHG because it is the most well studied effect and also the simplest case of all. The phase matching condition is controlled by selecting different values of grating period Λ . For all calculations presented in this section, Λ is $2.2632171\mu\text{m}$ for phase match and $2.25\mu\text{m}$ for phase mismatch.

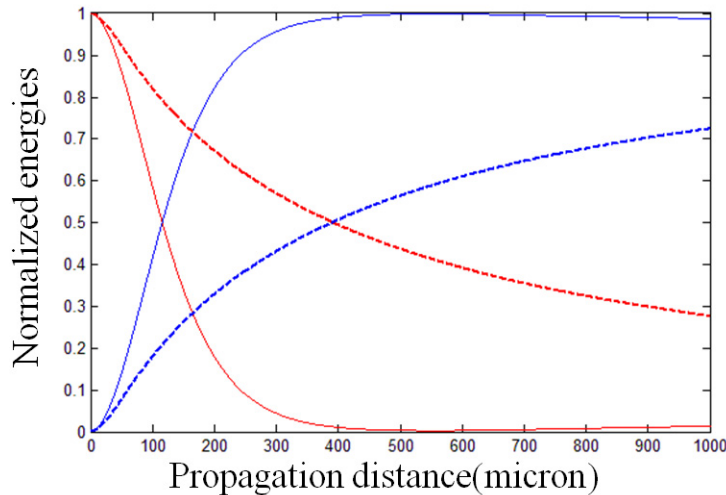


Figure 6.2: Phase-matched SHG energy evolution of cw (solid line) and pulse (dashed line), red for pump and blue for SH. The peak intensity of the (fundamental) short pulse (100 fs) is the same as the intensity of the cw case, which is approximately $5 \times 10^{13} \text{ W/m}^2$. The pulse energy is plotted as a function of distance (initial energy normalized to 1). The cw intensity is also plotted, normalized to the initial value of the fundamental.

The simulation of phase-matched cw SHG follows the hyperbolic tangent and secant as predicted by standard analytical solutions. The accuracy of the numerical

approximation was verified using the analytical solution of the set of differential equations. In the same figure we also plot the phase-matched SHG for 100 fs pulse of the same peak intensity as the cw field. The energy evolution curves for 100 fs pulse clearly have a gentler slope than for cw field. That is because, in the short pulse case, the frequency dependence of the k -vectors has to be taken into account. Only the zero-th order of their Taylor expansions are perfectly phase matched. Because the exact frequency dependence of the k -vectors is introduced via the Sellmeier equations, the lower conversion efficiency in the pulsed case reflects several phenomena usually treated separately through the various orders of a Taylor expansion of the function $k(\Omega)$. These effects are (i) phase mismatch for frequency components away from the central frequency (first order), (ii) temporal walkoff due to the difference group velocities for the fundamental and second harmonic (first order), (iii) pulse reshaping due to second and higher order dispersion. All these effects scale with $(\Delta k)_{max}L$, where $(\Delta k)_{max}$ is the largest k -vector mismatch across the pulse spectrum, and L is the crystal length. Therefore, the cw conversion efficiency can be approached by reducing the size of the crystal L , and increasing the intensity, to the maximum compatible with the damage threshold of the material.

An interesting phenomenon of effective third-order nonlinearity manifests itself if the SHG is phase mismatched as discussed in reference [10] for cw SHG. We indeed reproduce the same effect in our simulation. The energy transfers between pump and SH with a period of $\Delta L = \frac{2\pi}{\Delta k}$, as shown by the solid curves in Fig. 6.3, which represents the evolution of the intensity of the fundamental (red) and second harmonic (blue) for cw radiation, as a function of distance. The curves are plotted for increasing initial intensities, which are $1, 2, 3, 4, 5 \times 10^{13} \text{W/m}^2$. It is seen that the period ΔL decreases with intensity, which suggests either an intensity dependent “equivalent index” at the fundamental $[n_{eff}(\omega)]$ or at the second harmonic $[n_{eff}(2\omega)]$. Further calculations indicate that the nonlinear index applies primarily to the fundamental. The periodicity of second harmonic conversion versus distance is washed

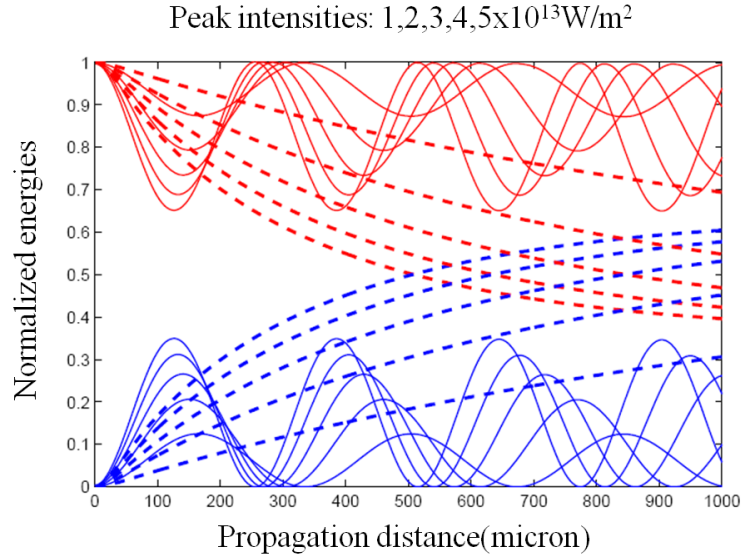


Figure 6.3: Phase-mismatched SHG energy evolution of cw (solid line) and pulse (dashed line), red for pump and blue for SH. The two cases have same peak intensity and the energies are normalized. The period varies with different intensities for cw SHG.

out, as can be seen from the dashed lines in Fig. 6.3. There are two contributions to this effect. One is because of the plurality of frequencies involved in a 100 fs pulse, and the other is due to the large temporal walkoff between fundamental and second harmonic (fig. 6.9).

A considerable simplification in the simulation of interwoven Ti:sapphire-OPO cavities could be made if some simple empirical formulae could be used to approximate the pulse modification as it propagates through the nonlinear crystal. Fig.6.3 suggests some modified hyperbolic tangent and secant can be used, as in the phase matched case of Fig. 6.2. The main difference between phase mismatch and perfectly match for short pulse is that the energy of the phase mismatched SH never reaches maximum (the initial pump energy) no matter how high the pump intensity is. In fact it saturates earlier than the phase matching case.

And we can see in fig. 6.4, for continuous radiation, the pump phase has a lin-

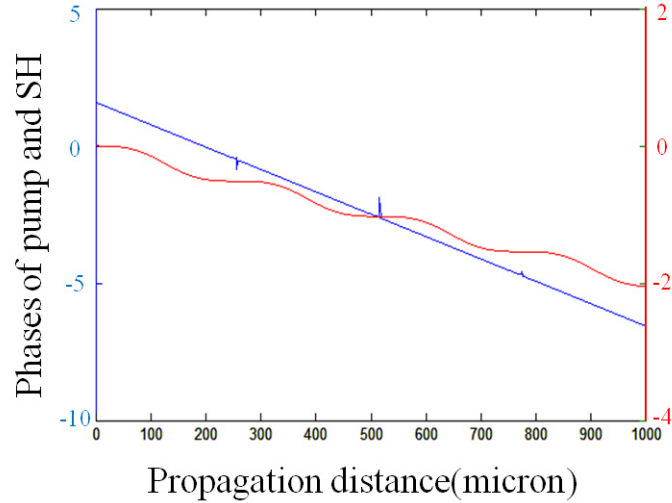


Figure 6.4: Phase-mismatched SHG cw phase modulation. The pump phase (red, right scale) has a linear slope along propagation with some modulation, while the phase variation of the second harmonic (blue, left scale) is much steeper, but nearly perfectly linear.

ear slope along propagation with some modulation, while the phase variation of the second harmonic is much steeper, but nearly perfectly linear. This phase modulation is directly related to the energy transfer through Kramers-Kronig dispersion relations. The phase modulation results from the dispersion associated with a two-photon (quadratic) loss. Kramers Kronig relations have been established for two-photon absorption and nonlinear index n_2 [17]. Whenever there is a two-photon resonance, there is associated, off-resonance, an enhancement of the nonlinear index n_2 (negative below resonance, positive above resonance). The same applies to harmonic generation, since it is also a quadratic loss. The “resonance” is here the phase matching frequency (or in our case, the “quasi” phase matching frequency). For better understanding of this phenomenon we represent the field vectors in a complex plane shown in Fig. 6.5. Each data point is the electrical field of pump (red) or second harmonic (blue) integrated over time and plotted in the complex plane. The purpose of Fig. 6.5 is to indicate the amplitude and phase evolution of the field

vectors as they propagate through the crystal. In the case of phase matching of cw second harmonic generation, the fundamental and second harmonic complex vectors spin counterclockwise at the same rate kz . The second harmonic [curve starting downwards from the origin $(0, 0)$] is generated with a phase difference of $-\pi/2$ with respect to the fundamental, and the end of its complex vector describes an outward spiral. The fundamental complex vector (still cw, phase matched case) shrinks as it spins, tracing an inward spiral. With the phase mismatch of 0.005π in Fig. 6.5, the phase angle of the second harmonic $k_{2\omega}z$ spins faster than the fundamental. The second harmonic returns to zero after having spun an angle of θ_2 , while θ_p is the angle that the fundamental has rotated for that same distance. It is clear from the picture that they satisfy the relation: $\theta_2 + 2\theta_p = \pi$. The intensity dependence of “cascaded n_2 ” is contained in that angle θ_p . As the intensity increases and the rate of shrinking (expansion) of the fundamental (second harmonic) spirals increases, the angle θ_p increases, making it seem like the refractive index for the fundamental is increasing.

The simulation of phase mismatched SHG is also done for short pulses in comparison with the CW case. Energy evolution of pump and SH for phase mismatched SHG is shown in Fig 6.3. Modulated phase along propagation is observed in the pulse regime for phase mismatched SH pulse instead of pump. For pump phase there is only a little deviation from a straight line as shown in Fig. 6.6.

Plots of the field vectors in the complex plane (imaginary part of the fields versus real part) for phase matched SHG (Fig 6.7) and phase mismatched SHG (Fig 6.8), after transmission through 1 mm of PPLN, are shown. As expected, there is little phase modulation in the phase matched case. In the phase mismatched case, the pump shows a linear phase modulation with time. Both results of Fig. 6.6 and Fig 6.8 are consistent with a nonlinear index for the pump beam. The second harmonic presents a periodical behavior similar to the cw case. The trajectory in the complex

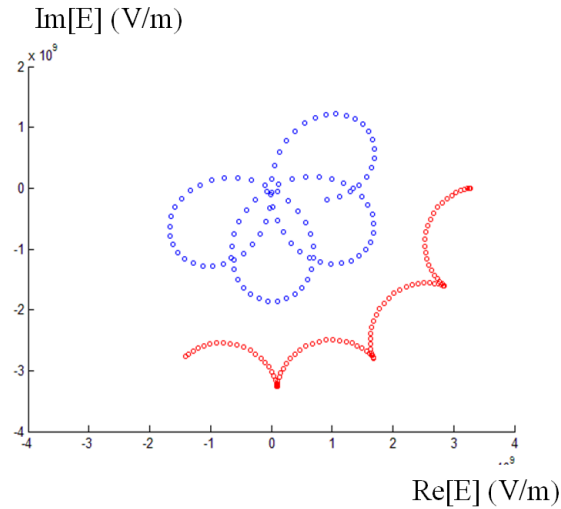


Figure 6.5: Phase-mismatched SHG with continuous waves. Each data point is the electrical field of pump (red) or second harmonic (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagate through the crystal.

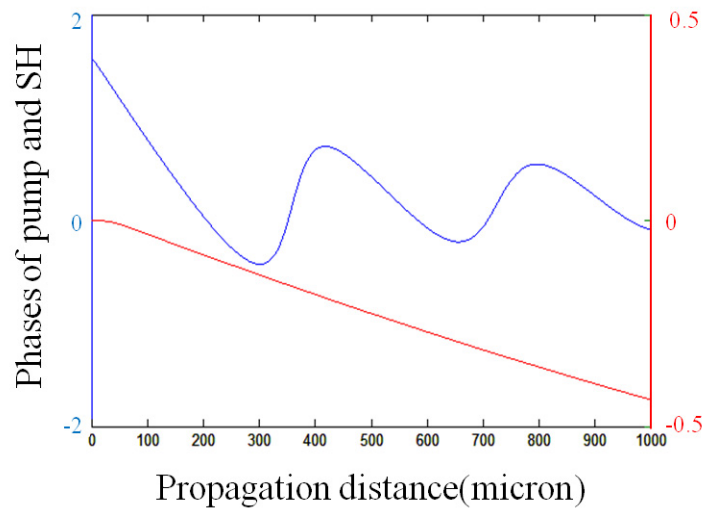


Figure 6.6: Phase-mismatched SHG pulse propagation. The phase of the pump (red curve, right scale) and that of the second harmonic (blue curve, left scale), are plotted as a function of distance. The phase are defined as $\varphi_{avg} = \int \varphi(t)I(t)dt / \int I(t)dt$. The pump pulse (100 fs) has a peak intensity of about $5 \times 10^{13} \text{ W/m}^2$.

plane is an inward spiral, instead of the spinning motion of the cw case.

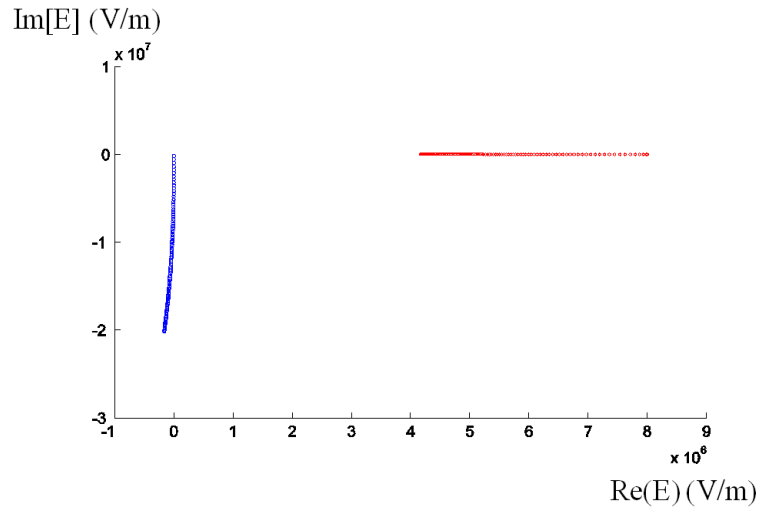


Figure 6.7: Phase-matched SHG with pulses. Each data point is the electrical field of pump (red) or second harmonic (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagate through the crystal.

Group velocity mismatch for short pulses is another well discussed issue in frequency conversions. In the PPLN crystal the SH pulse travels much slower than the pump pulse, and our simulation (Fig.6.9) shows a longer SH pulse generated along propagation. At the entrance of the crystal, the un-depleted fundamental creates the most intense second harmonic. That second harmonic created earlier propagates more slowly, and loses overlap with the fundamental. Further down along the crystal, generation of second harmonic continues with decreasing yield, due to the depletion of the fundamental. As a result, the second harmonic that exits the crystal has a leading edge resembling that of the fundamental, and a long trailing edge characterized by the difference in group velocities, and the rate of depletion of the fundamental. At each passage through the PPLN crystal, the second harmonic is thus broadened and delayed, as shown in Fig.6.9. The reshaping of the pump pulse is less dramatic (Fig. 6.10). Because of larger depletion at the peak of the fundamental

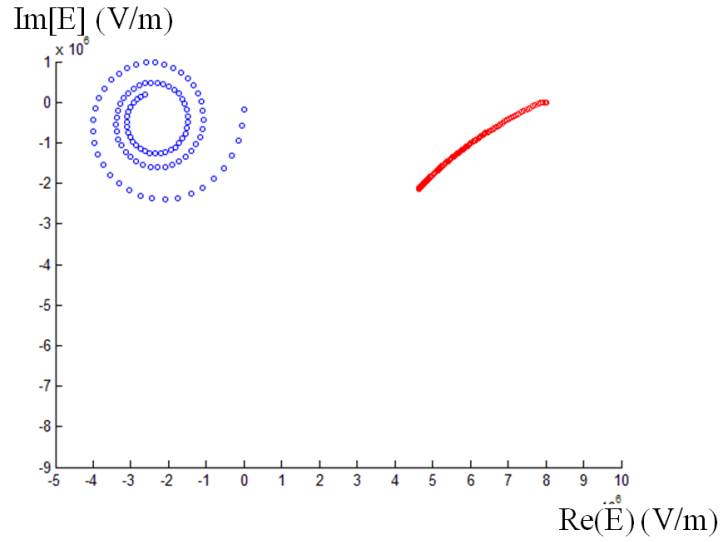


Figure 6.8: Phase-mismatched SHG with pulses Each data point is the electrical field of pump (red) or second harmonic (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagate through the crystal.

pulse, there is some pulse broadening.

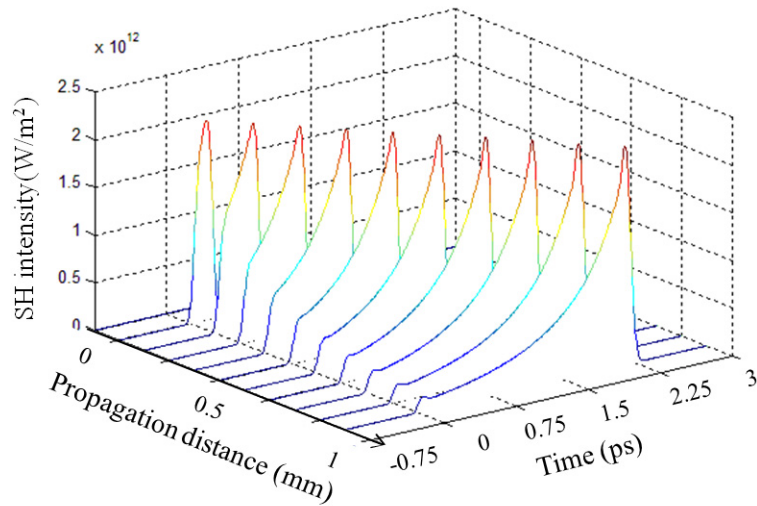


Figure 6.9: Phase-mismatched evolution of the second harmonic in SHG.

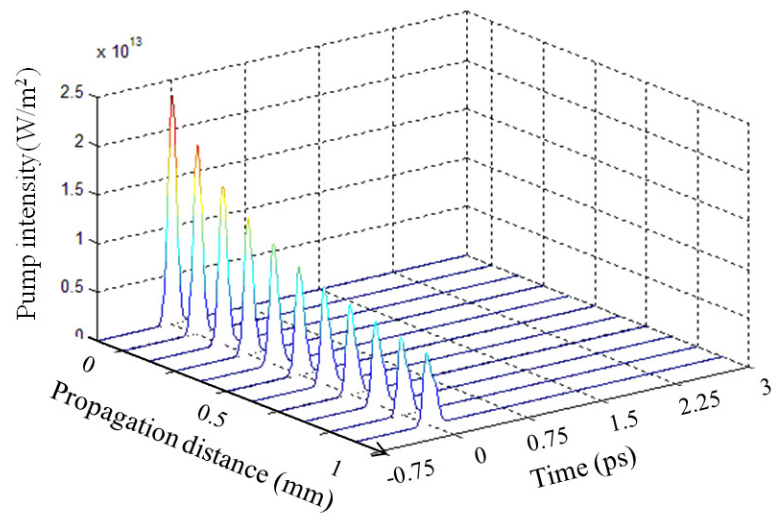


Figure 6.10: Phase-mismatched evolution of the pump pulse intensity in SHG.

6.2.2 SFG only

SHG is considered as a degenerate case of SFG. While SFG shares some similarity with the SHG discussed above, it also presents a few different behaviors. In this section, the grating period of the PPLN crystal is $4.451797 \mu\text{m}$ for phase matching and $4.4 \mu\text{m}$ for phase mismatch. Fig.6.11 shows the phase-matched energy evolution of SFG for cw (solid line) and pulse (dashed line) cases, respectively.

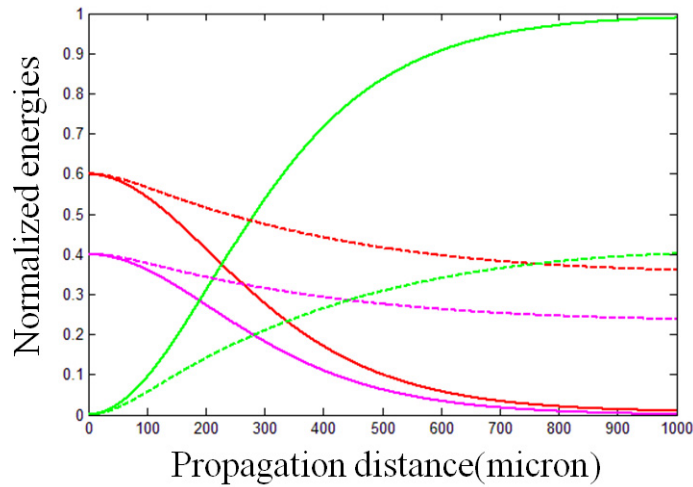


Figure 6.11: Phase-matched SFG energy evolution for cw (solid line) and pulse (dashed line) cases respectively. Initial intensities are $3 \times 10^{13} \text{W/m}^2$ (pump) and $2 \times 10^{13} \text{W/m}^2$ (signal) for both cw and pulse. Red stands for pump frequency, magenta for signal and green for sum.

The main difference comes from the initial energy ratio between two input wavelength, namely 770nm and 1140nm in our simulation. When the interaction is phase matched, unlike SHG, the input waves both disappear only if their energy ratio is the inverse of the wavelength ratio as shown in Fig. 6.11. If their energy ratio does not satisfy this condition, when one input wave loses all its energy, the leftover of the other input wave will start the DFG process with the generated sum wave and gain energy back. This periodic interchange between SFG and DFG is illustrated in

Fig. 6.12.

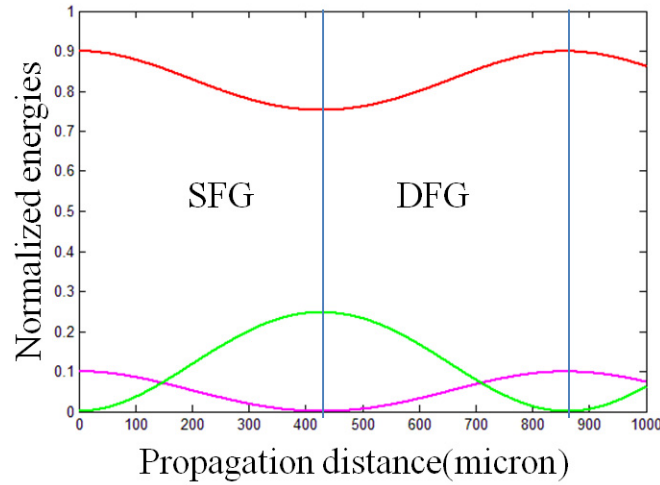


Figure 6.12: Phase-matched SFG energy evolution for cw only. Initial intensities are $4.5 \times 10^{13} \text{W/m}^2$ (pump) and $0.5 \times 10^{13} \text{W/m}^2$ (signal). Red stands for pump frequency, magenta for signal and green for sum. As the signal wave vanishes the SFG turns into DFG for sum and pump frequencies. When the sum energy decreases to zero it returns to the initial condition of SFG and repeats these processes.

If the SFG process is phase mismatched, we observe an energy oscillation in the case of cw radiation. Similarly to the case of SHG, there is an averaging out effect for the pulsed case. Fig. 6.13 shows the phase-mismatched energy evolution of SFG for cw (solid line) and pulse (dashed line) cases, respectively.

The field vectors are presented in the complex plane for pump and sum frequencies in Fig. 6.14 (cw case) and Fig. 6.15 (pulse case), when the SFG is phase mismatched. Each data point is the electrical field of pump (red) or sum (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagate through the crystal. As in the case of SHG, the sum frequency vector starts downwards from the point (0, 0) and spins counterclockwise.

The intensity evolution of pump, signal and sum pulses for phase mismatched

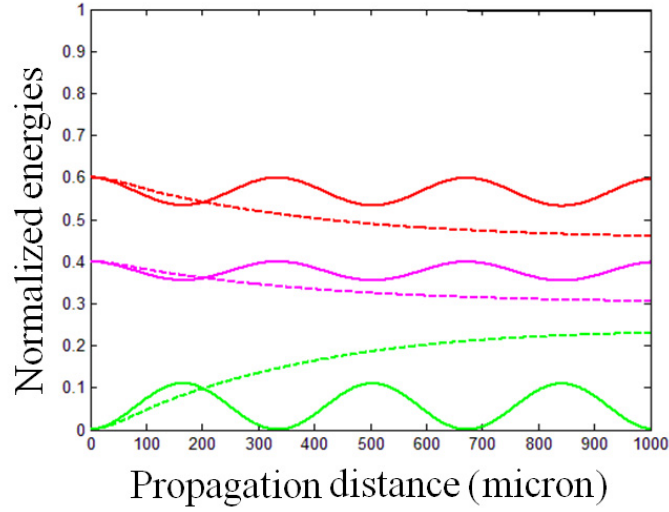


Figure 6.13: Phase-mismatched SFG energy evolution for cw (solid line) and pulse (dashed line) cases respectively. Initial intensities are $3 \times 10^{13} \text{W/m}^2$ (pump) and $2 \times 10^{13} \text{W/m}^2$ (signal) for both cw and pulse. Red stands for pump frequency, magenta for signal and green for sum.

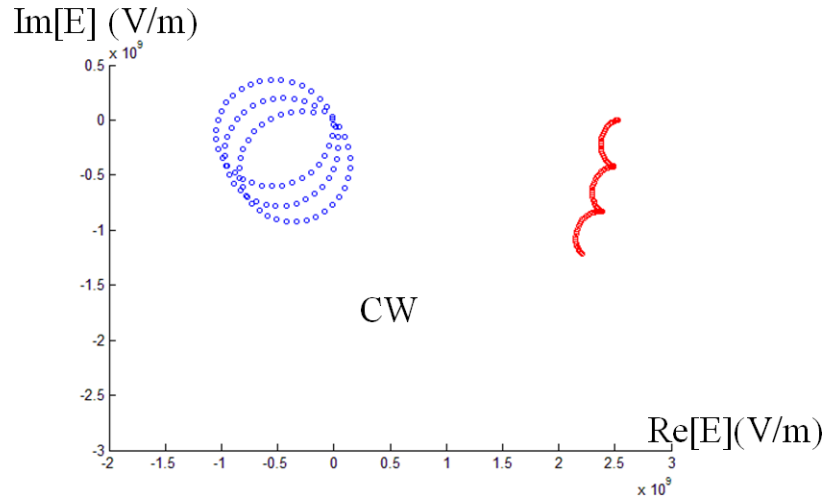


Figure 6.14: Phase-mismatched SFG for cw pump and sum. Each data point is the electrical field of pump (red) or sum (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagate through the crystal.

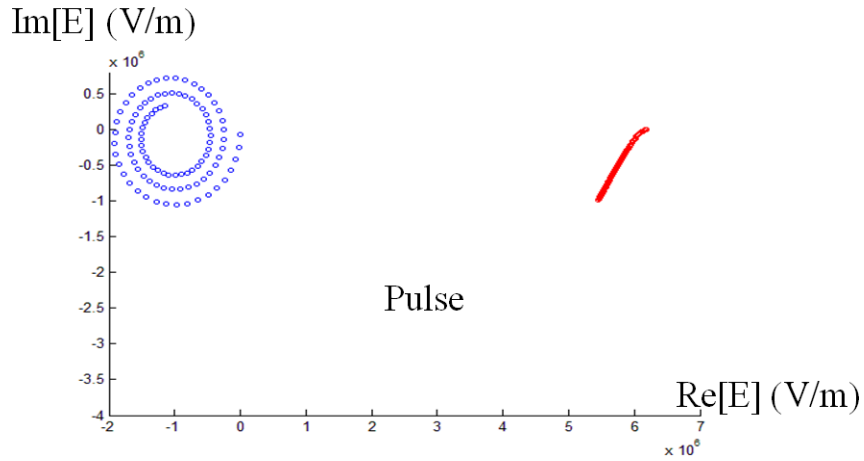


Figure 6.15: Phase-mismatched SFG pump and sum pulses. Each data point is the electrical field of pump (red) or sum (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagate through the crystal.

SFG are plotted in Fig. 6.16, Fig. 6.17 and Fig. 6.18 respectively.

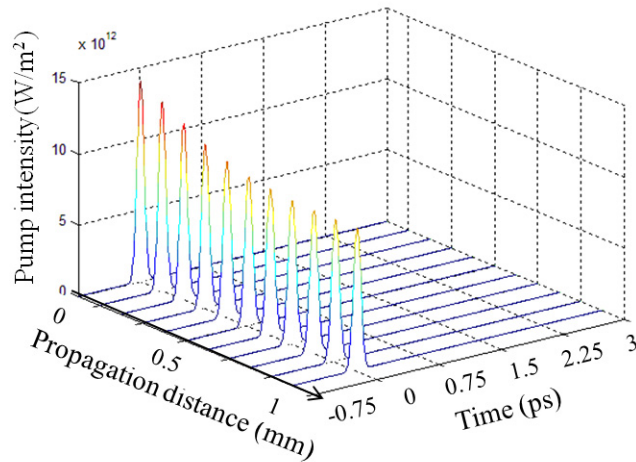


Figure 6.16: Phase-mismatched evolution of the pump pulse in SFG.

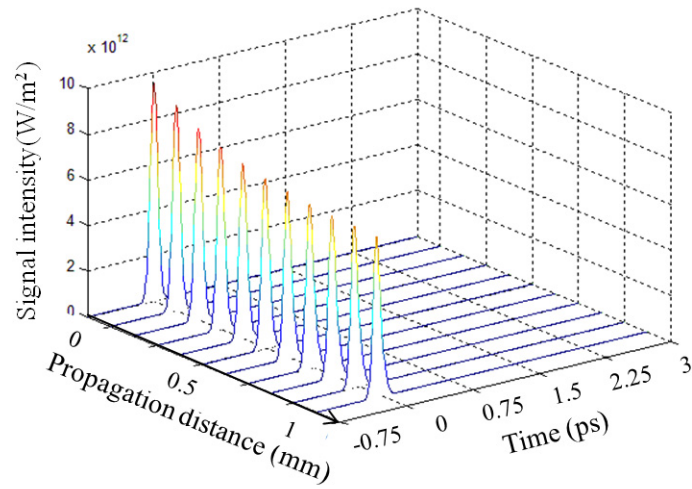


Figure 6.17: Phase-mismatched evolution of the signal pulse in SFG.

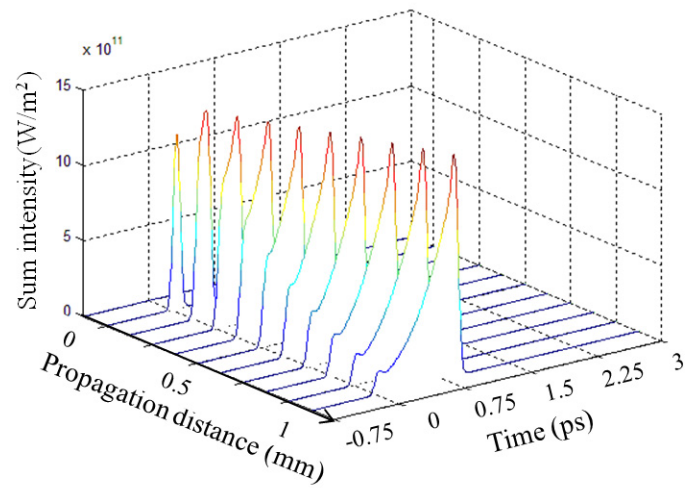


Figure 6.18: Phase-mismatched evolution of the sum pulse in SFG.

6.2.3 DFG only

The DFG process is the dominating all second order nonlinear process in an OPO. In this section we focus on the pulsed case only. The previous discussion of SHG and SFG does not include the quasi phase matching effect. From equation (A.18) we can predict that the interaction strength of quasi-phase-matching is roughly $2/\pi$ times smaller than perfect phase matching, with a periodic modulation corresponding to each poled domain, as shown in Fig. 6.19.

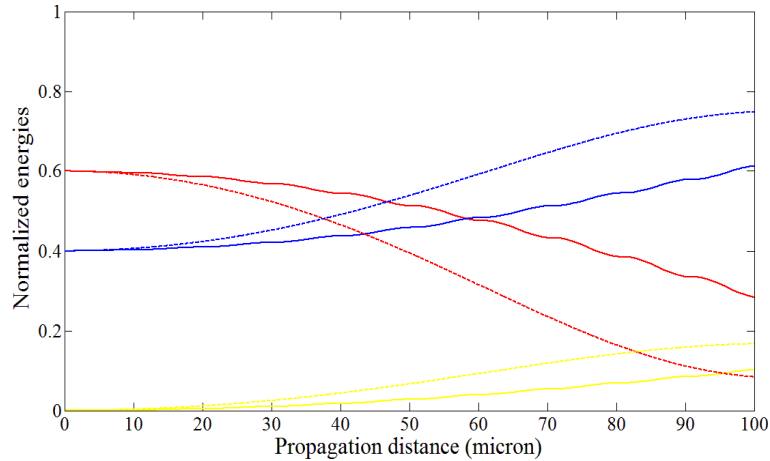


Figure 6.19: Phase-matched (dashed line) and quasi-phase-matched (solid line) DFG processes. Red stands for pump frequency, blue for signal and yellow for idler. The propagation distance is $100\mu\text{m}$ and the PPLN grating period is $20.23\mu\text{m}$.

Fig. 6.20 shows the energy evolution when the PPLN grating period is changed to $22\mu\text{m}$. When the DFG is phase mismatched, energy evolution still preserves the oscillating behavior because the walk-off between pump and signal are not as huge as in the SHG and SFG processes. However if we increase the initial spectrum (decrease pulse duration) the oscillation will be averaged out as shown in SHG and SFG processes.

The field vectors are presented in complex plane for pump and signal pulses in Fig. 6.21. Each data point is the electrical field of pump (red) or signal (blue)

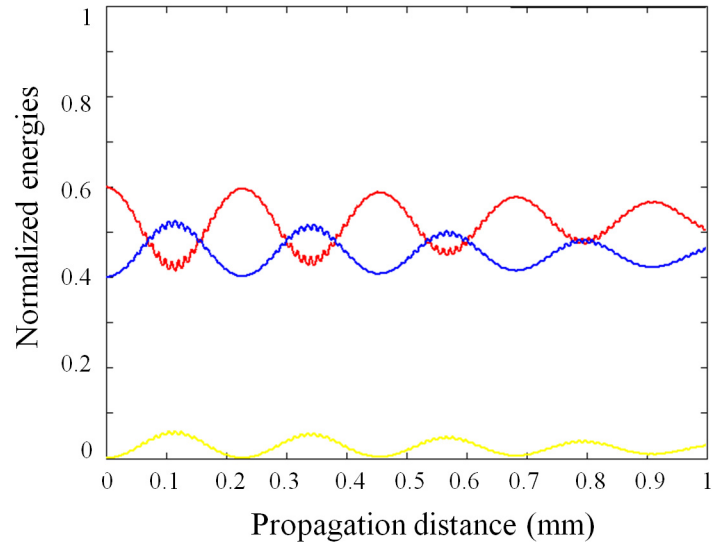


Figure 6.20: Quasi-phase-mismatched DFG processes. Red stands for pump frequency, blue for signal and yellow for idler. The propagation distance is 1mm and the PPLN grating period is $22\mu\text{m}$.

integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagate through the crystal. It also creates an effective n_2 which we will discuss in more details later.

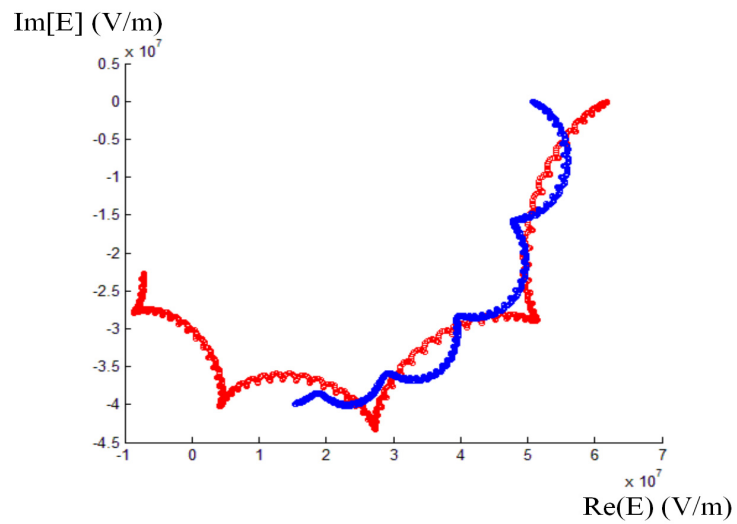


Figure 6.21: Quasi-phase-matched DFG processes for pulses. Each data point is the electrical field of pump (red) or signal (blue) integrated over time and plotted in the complex plane. The purpose is to indicate the amplitude and phase evolution of the field vectors as they propagate through the PPLN crystal which is 1 mm long and whose grating period is $22\mu\text{m}$.

The intensity evolution of pump signal and idler pulses for quasi phase matched DFG are plotted in Fig. 6.22, Fig. 6.23 and Fig. 6.24 respectively.

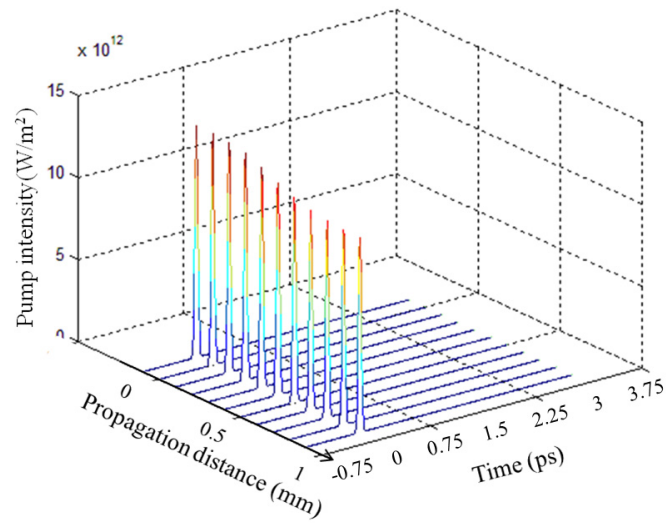


Figure 6.22: Phase-mismatched evolution for pump pulse in DFG.

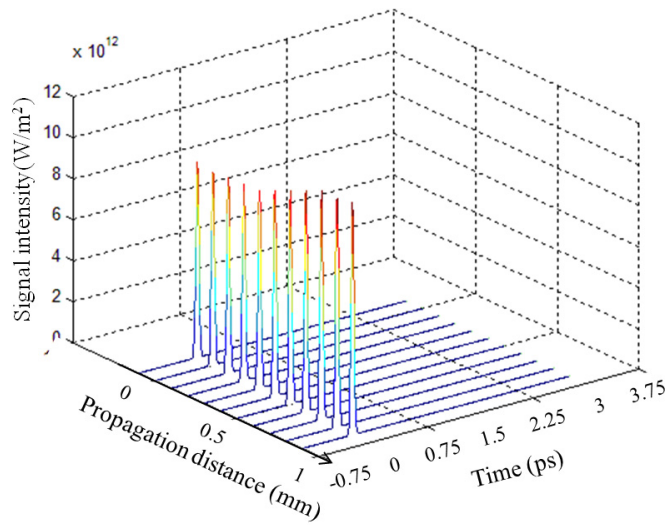


Figure 6.23: Phase-mismatched evolution for signal pulse in DFG.

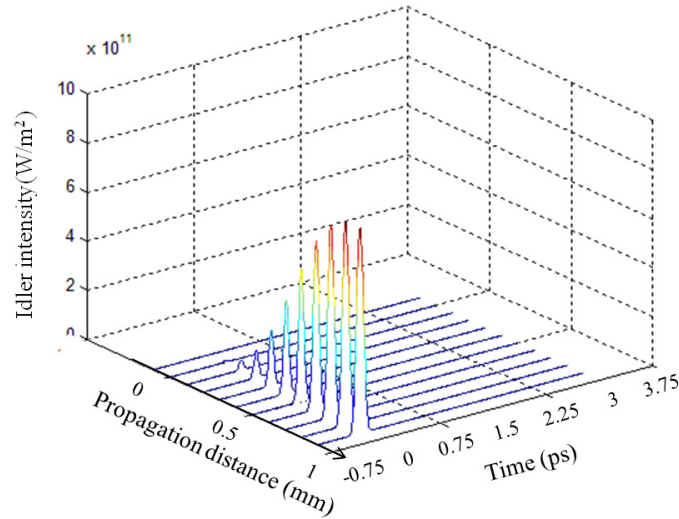


Figure 6.24: Phase-mismatched evolution for idler pulse in DFG.

6.3 Including all experimental conditions

There are many experimental parameters that affect the results of our simulation. In this section we attempt to include all parametric dependence of the second order processes in the PPLN. In our experiment we pump the OPO inside a Ti:Sapphire cavity with a pulse duration on the order of 100fs. The PPLN crystal is 1mm long with a grating period of $20.23\mu\text{m}$. In the complete model containing DFG, SHG and SFG, only the DFG is quasi-phase-matched while the SHG and SFG have a large quasi phase mismatch vector. The simulation has shown that the effect of SHG and SFG can be approximated by a correction to the material nonlinear index. The input of our model is both pump pulse and signal pulse with intensities around $3 \times 10^{13}\text{W}/\text{m}^2$ and $2 \times 10^{13}\text{W}/\text{m}^2$ respectively. The pulses are transform limited if no chirp is specified.

6.3.1 Optimization of input pulse parameters

For 1mm long PPLN crystal, the optimal input pulse duration is around 150 fs given a fixed initial energy (Fig. 6.25), and the optimal delay of signal pulse with respect to pump pulse is around 120 fs (Fig. 6.26). To understand these numbers, we compare it with the group delay mismatch which is calculated to be 233 fs between pump and signal. The optimal delay should indeed be half of the 233 fs mismatch.

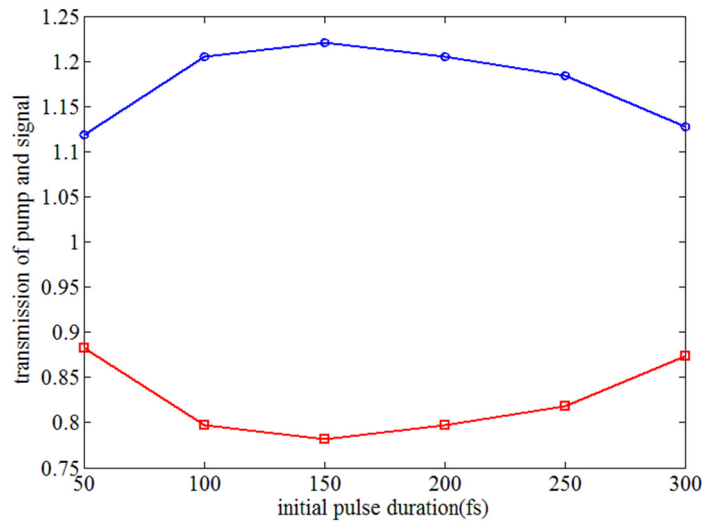


Figure 6.25: Transmission of pump and signal pulse energy after propagation through a 1mm PPLN for different initial pulse duration. Blue stands for pump frequency, red for signal. The pulse duration is varied from 50fs to 300fs, with 50fs interval.

A parabolic fit can be made of the traces of Fig. 6.25,

$$\begin{aligned}
 T_p(\tau) &\approx 0.78 + 6.1 \cdot 10^{-6}(\tau - 150)^2 \\
 T_s(\tau) &\approx 1.23 - 6.2 \cdot 10^{-6}(\tau - 150)^2
 \end{aligned} \tag{6.22}$$

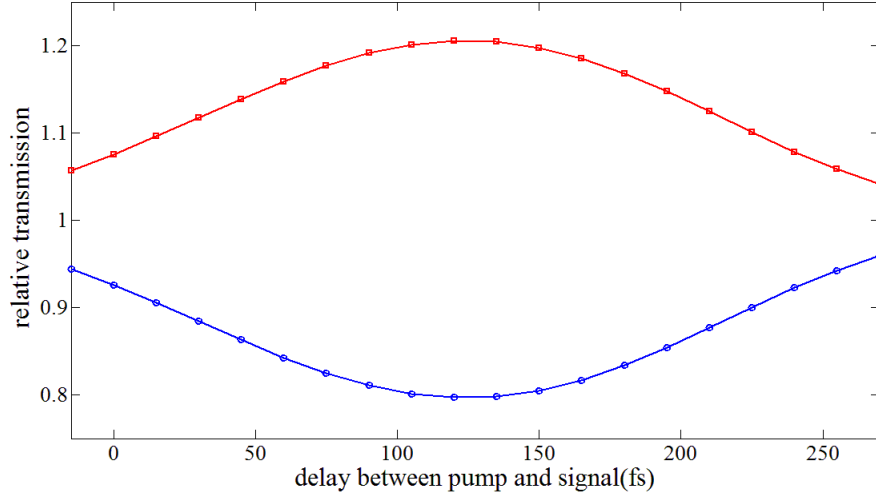


Figure 6.26: Transmission of pump and signal pulse energy after propagation through a 1mm PPLN for different time delays between them. Red stands for pump frequency, blue for signal. The delay time is varied from -15fs to 270fs, in 15fs interval.

and Fig. 6.26.

$$\begin{aligned}
 T_p(\Delta\tau_d) &\approx 0.8 + 7.8 \cdot 10^{-6}(\Delta\tau_d - 120)^2 \\
 T_s(\Delta\tau_d) &\approx 1.2 - 7.9 \cdot 10^{-6}(\Delta\tau_d - 120)^2
 \end{aligned} \tag{6.23}$$

If we combine the two sets of equations, the estimated transmission for pump and signal pulse are:

$$\begin{aligned}
 T_p(\tau, \Delta\tau_d) &\approx 0.79 + 6.1 \cdot 10^{-6}(\tau - 150)^2 + 7.8 \cdot 10^{-6}(\Delta\tau_d - 120)^2 \\
 T_s(\tau, \Delta\tau_d) &\approx 1.215 - 6.2 \cdot 10^{-6}(\tau - 150)^2 + 7.9 \cdot 10^{-6}(\Delta\tau_d - 120)^2
 \end{aligned} \tag{6.24}$$

6.3.2 Pulse center shift after one passage

The DFG interaction between pump and signal will modify their pulse shapes and therefore shift the pulse centers of gravity. The shift depends on the delay between

pump and signal pulse shown in Fig. 6.27. The references are taken at the pulse centers of gravity respectively after pure propagation without nonlinear interaction.

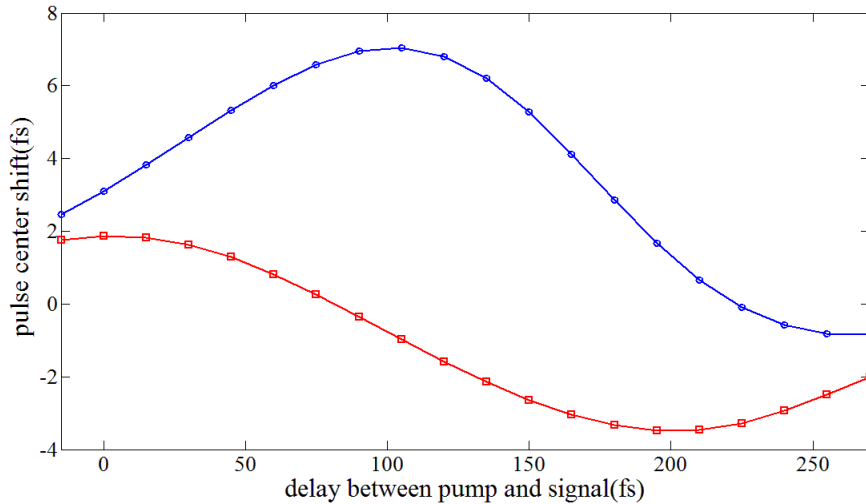


Figure 6.27: Pulse center shift for pump and signal pulse after propagation in a 1mm PPLN for different delay time. Red stands for pump frequency, blue for signal. The delay time is varied from -15fs to 270fs, with 15fs interval.

A parabolic fit for the center part of red trace of Fig. 6.27, and a linear fit for the center part of red trace are made:

$$\begin{aligned}
 Center_p(\Delta\tau_d) &\approx 7.4 - 5.9 \cdot 10^{-4}(\Delta\tau_d - 102)^2 \\
 Center_s(\Delta\tau_d) &\approx 2.7 - 0.034 \cdot \Delta\tau_d
 \end{aligned} \tag{6.25}$$

In the experiment, we observe that the repetition rate of the pump cavity decreases when the OPO starts lasing. For instance, the repetition rate changes from 180.8823 MHz (without OPO lasing) to 180.8821 MHz (with OPO lasing) and indicates a pulse center delay of 6 fs. This phenomenon becomes more obvious if a longer pump pulse is present.

We also consider the case for other signal wavelengths. Fig. 6.29 and Fig. 6.30

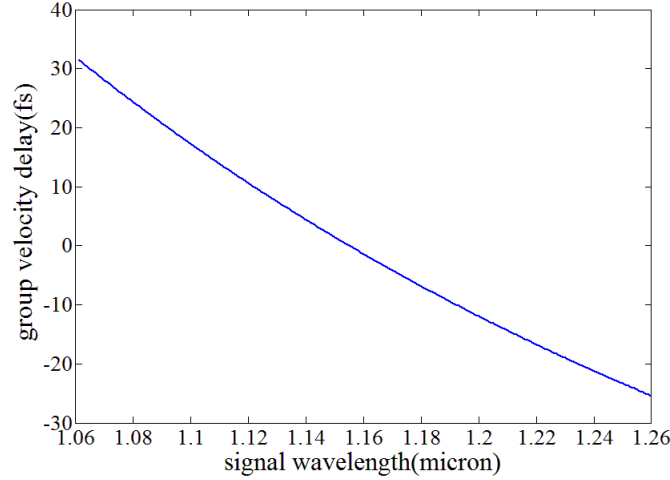


Figure 6.28: The group delay in a 1mm PPLN crystal for signal wavelength from $1.06\mu\text{m}$ to $1.26\mu\text{m}$ calculated from Sellmeier equation.

show the pump and signal pulse center shifts, respectively, as a function of delay for signal wavelengths from 1110 nm to 1170 nm. The relative pulse center shift is made of the difference between pump and signal pulse centers shown in Fig. 6.31. The characteristics in pulse center shift as a function of delay is similar for different wavelengths. The spacing between each curves for different wavelength indicates a wavelength dependence of the group velocity delay of about 1fs for 3nm signal wavelength detuning (Fig. 6.28). Therefore we can rewrite the approximate equations Eq. (6.25) for pulse center shift by adding a proper group velocity delay term corresponding to different wavelength.

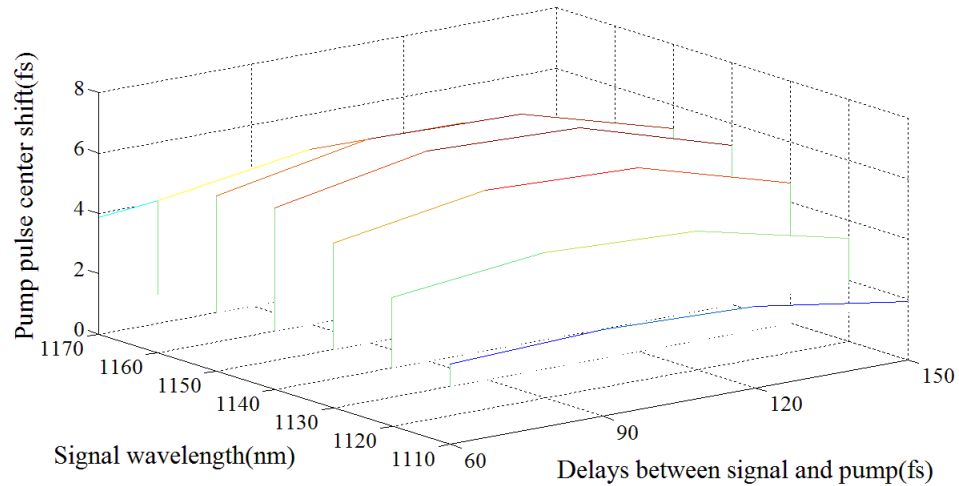


Figure 6.29: Pulse center shift for pump pulse after propagation in a 1mm PPLN for different signal wavelengths (1110 to 1170nm with 10nm interval) at different delay time (60, 90, 120, 150fs).

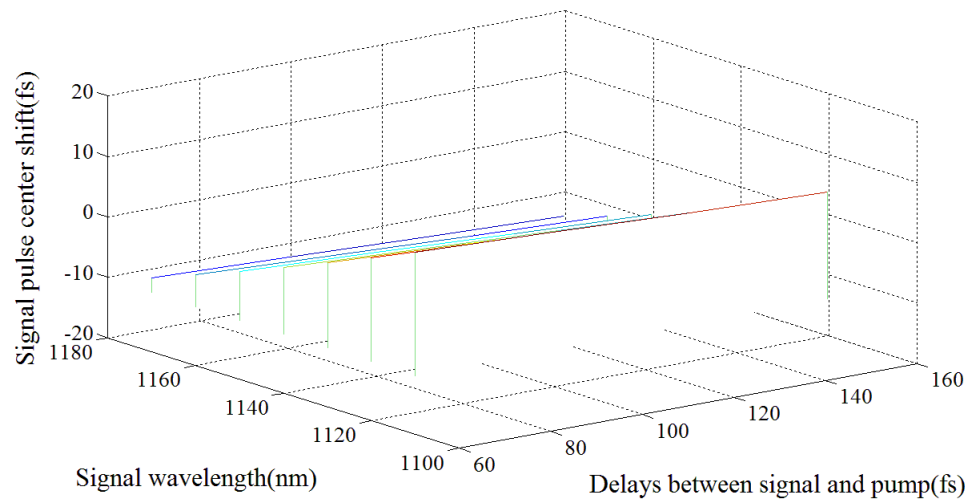


Figure 6.30: Pulse center shift for signal pulse after propagation in a 1mm PPLN for different signal wavelengths (1110 to 1170nm with 10nm interval) at different delay time(60, 90, 120, 150fs).

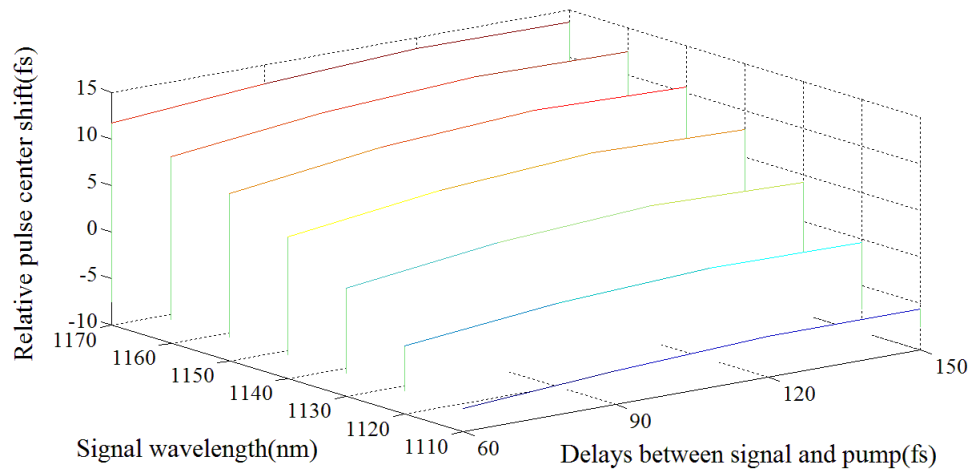


Figure 6.31: Relative pulse center shift between pump pulse and signal pulse after propagation in a 1mm PPLN for different signal wavelengths(1110 to 1170nm with 10nm interval) at different delay time(60, 90, 120, 150fs).

6.3.3 Wavelength shift at each passage

It is an accepted concept that an externally pumped OPO changes wavelength to track the repetition rate of the pump. There are however two different mechanisms that can account for this apparent tracking.

1. If the repetition rate of the pump is modified, the OPO is no longer at synchronism, and the oscillation ceases. The laser starts again from noise, and the wavelength components that match the pump period through their group velocity emerge.
2. There is a real wavelength shift at each passage through the OPO if the repetition rates is not matched.

In the first case, the laser will show maximum instability, since the wave has to die and be resuscitated at each cavity length fluctuation.

The second mechanism, if it exist and can be enhanced, maybe the key to intracavity pumped OPO stabilization.

The spectrum of signal pulse is examined before and after nonlinear interaction, and a tiny wavelength shift is measured through our simulation as a function of the delay between signal and pump. This wavelength shift is less than 0.1% of the signal spectrum, but can be non negligible when the signal pulse propagates many round trips in the OPO cavity.

From delay of 60fs to 180fs the wavelength shift can be approximated as

$$\lambda_s(\Delta\tau_d) \approx 0.0621 - 5.67 \cdot 10^{-4} \Delta\tau_d \quad (6.26)$$

and the following discussion will focus on the linear region from 60fs to 150fs.

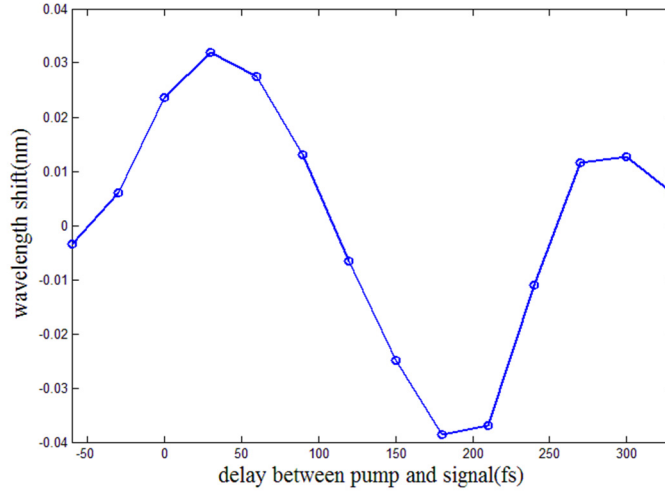


Figure 6.32: Wavelength shift for signal pulse after propagation in a 1mm PPLN for different delay time. The delay time is varied from -60 fs to 330 fs, with 30 fs interval.

We also examined this wavelength shift for different signal wavelength inputs for delays set at 60fs, 90fs, 120fs and 150fs respectively. As shown in Fig. 6.33 we find the output wavelength is shifted towards the phase matched wavelength from a negative slope. This is because of the wavelength dependent gain that modifies the signal wavelength as it propagates through the crystal.

In order to increase this wavelength shift effect, a counter-chirped pump and signal pulse model is proposed. We run the program for wavelength 1130, 1140 and 1150nm, delay time 60, 90, 120 and 150fs, and chirp coefficients 0.5, 1, 1.5, 2 and 3. A linear interpolation for the 3D wavelength shifts data (Fig. 6.34) is performed afterwards. From Fig. 6.34 we can tell that the wavelength shift is indeed magnified with chirp, and the maximum effect is reached with chirp coefficient around 1.5.

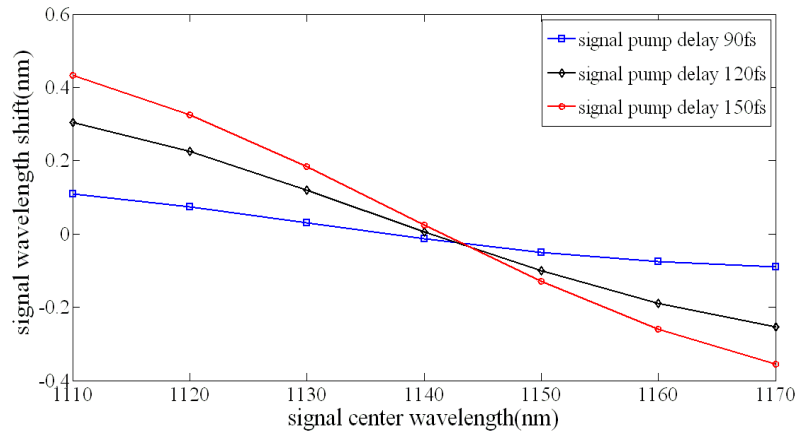


Figure 6.33: Wavelength shift for signal pulse after propagation in a 1mm PPLN as a function of signal wavelength for different delay time 90fs (blue), 120fs (black) and 150fs (red).

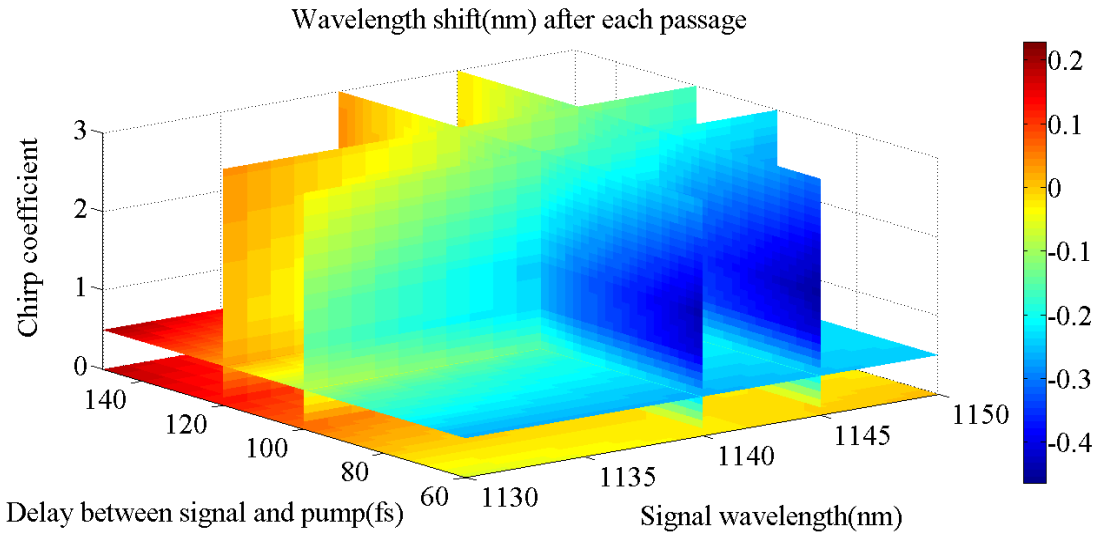


Figure 6.34: Wavelength shift for signal pulse after propagation in a 1mm PPLN for different wavelength (1130 to 1150nm), delay (60 to 150fs) and chirp coefficient (0 to 3)

6.3.4 Near Phase-matched interactions

Limited by mirror coating, λ_s may vary from 1060nm to 1260nm. However for the DFG process only one wavelength λ_s^0 satisfies the quasi-phase-matching condition given a fixed grating period. Therefore the gain of DFG is a function of signal wavelength (Fig. 6.35).

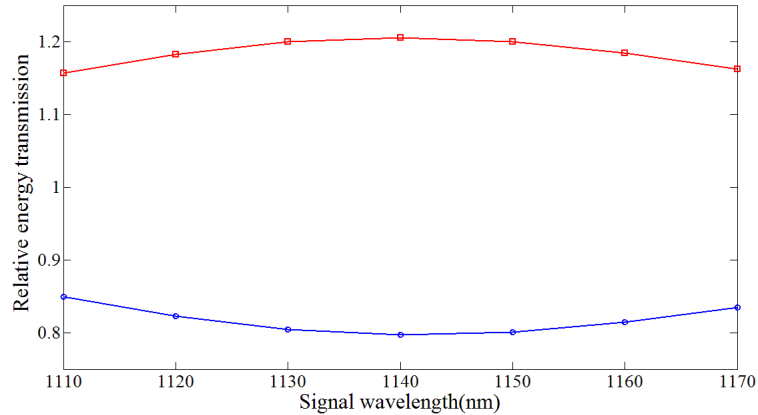


Figure 6.35: The transmission of pump (blue) and signal (red) pulses after propagation for signal wavelength from 1108nm to 1214nm

A parabolic fit can be made of the traces of Fig. 6.35.

$$\begin{aligned}
 T_p(\tau) &\approx 0.8 + 5.5 \cdot 10^{-4}(\lambda - 1140)^2 \\
 T_s(\tau) &\approx 1.2 - 5.5 \cdot 10^{-4}(\lambda - 1140)^2
 \end{aligned} \tag{6.27}$$

The chirp added to the pump and signal pulses will bring down the conversion efficiency, as shown in Fig. 6.36.

The wavelength tunability does not only affect the energy efficiency of DFG, but also the phase of both pump and signal pulses. Fig. 6.37 shows the phase shift of the signal at ω_s (red) and the pump at ω_p (blue) as a function of *signal* intensity, after

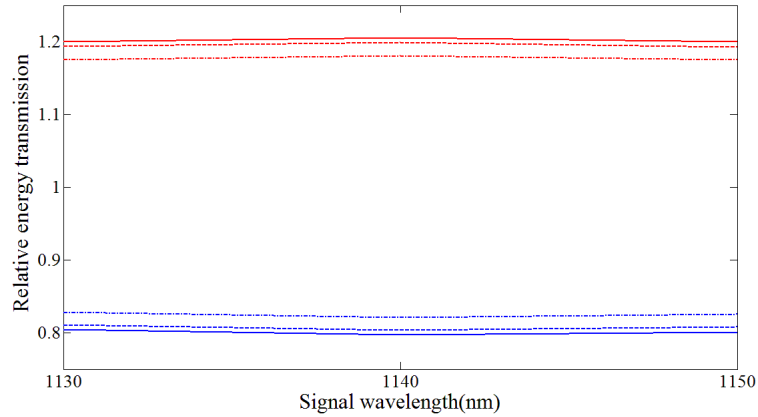


Figure 6.36: The transmission of pump (blue) and signal (red) as a function of signal wavelength for chirp coefficient 0(solid lines), 0.5(dashed lines) and 1(dash-dot lines).

propagation through 1 mm of PPLN. A detuning of 10nm is assumed. The most significant phase shift is that of the pump, corresponding to a large nonlinear index of $n_{2,ps} = -1.1 \cdot 10^{-15} \text{ cm}^2/\text{W}$.

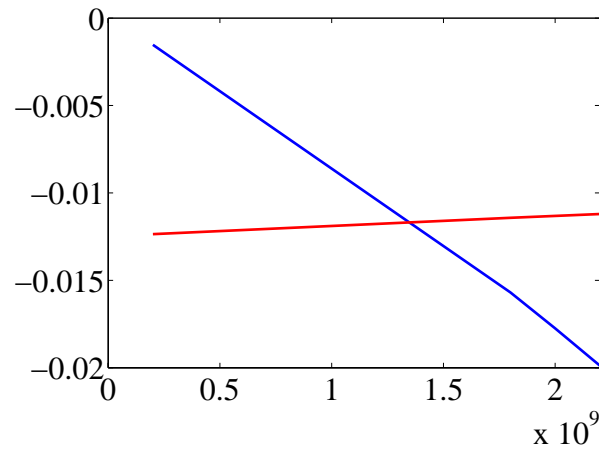


Figure 6.37: Phase shift of pump (blue) and signal (red) as a function of the intensity of the signal. As is often the case, the cross-phase modulation is larger than the self-modulation.

This nonlinear index is wavelength dependent, as shown in the calculation of Fig. 6.38. Good practical approximations for the value of the effective nonlinear

indices for the pump (affected by the signal intensity) $n_{2,ps}$ and the signal (affected by the pump intensity) $n_{2,sp}$ are:

$$n_{2,ps} = \frac{(1140 - \lambda_s)}{39} \cdot 10^{-14} \text{cm}^2/\text{W} \quad (6.28)$$

$$n_{2,sp} = \frac{(1140 - \lambda_s)}{108} \cdot 10^{-14} \text{cm}^2/\text{W} \quad (6.29)$$

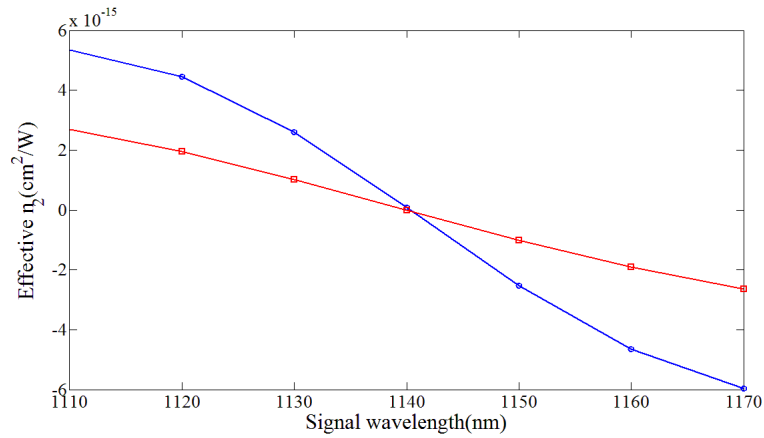


Figure 6.38: Effective n_2 of pump (blue) and signal (red) as a function of signal wavelength.

If the pump and signal pulses are counter-chirped, the effective n_2 is modified, as shown in Fig. 6.39.

6.3.5 Phase mismatched interactions

SHG of pump and SFG of pump and signal are strongly phase mismatched interactions. They produce less significant n_2 effects as compared to the near phase matched DFG process. The simulation of pulsed second harmonic generation shown Fig. 6.6 and Fig 6.8 suggest that the phase mismatched second harmonic of the pump can be assimilated to a nonlinear index. Calculations made as a function of intensity

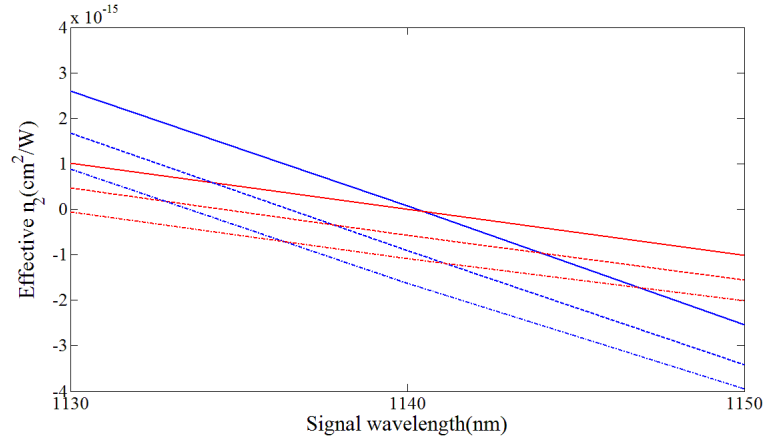


Figure 6.39: Effective n_2 of pump (blue) and signal (red) as a function of signal wavelength for chirp coefficient 0 (solid lines), 0.5 (dashed lines) and 1 (dash-dot lines).

show that, because of the low conversion to second harmonic, the phase modulation is primarily dependent on the pump intensity and can be represented by the index

$$\boxed{n_{2,pp} = -1.1 \cdot 10^{-16} \text{cm}^2/\text{W}} \quad (6.30)$$

The phase modulation is proportional to the pump intensity $\varphi_{pp}(t) = (2\pi/\lambda_p)n_{2,pp}I_pL$. Another nonlinear process affecting the pump is the phase mismatched sum frequency generation $\omega_p + \omega_s$, leading to a phase modulation of the pump $\varphi_{ps}(t) = (2\pi/\lambda_p)n_{2,ps}I_pL$, where

$$\boxed{n_{2,ps} = -1.9 \cdot 10^{-16} \text{cm}^2/\text{W}} \quad (6.31)$$

The same process affects also the signal, imparting a phase modulation on the signal, proportional to the pump intensity: $\varphi_{sp}(t) = (2\pi/\lambda_s)n_{2,sp}I_pL$, where

$$\boxed{n_{2,sp} = -1.8 \cdot 10^{-16} \text{cm}^2/\text{W}} \quad (6.32)$$

Contribution to pump and signal pulse propagation

$$\frac{\partial \tilde{\mathcal{E}}_p}{\partial z} = -\frac{2\pi i(n_{2,pp}I_p + n_{2,ps}I_s)}{\lambda_p} = \frac{2\pi i \times (1.1 \cdot 10^{-16}I_p + 1.9 \cdot 10^{-16}I_s)}{\lambda_p}. \quad (6.33)$$

$$\frac{\partial \tilde{\mathcal{E}}_s}{\partial z} = -\frac{2\pi i n_{2,sp}I_p}{\lambda_s} = \frac{2\pi i \times 1.8 \cdot 10^{-16}I_p}{\lambda_s}. \quad (6.34)$$

6.3.6 Gain of OPO at different pump levels

Last but not least, we study the strength of DFG interaction at different pump and signal levels. The OPO gain and pump depletion are defined as the relative transmission of the signal and pump pulse energy, respectively. We simulate the values for 100 fs pump and signal pulse duration, 50 μm focal spot diameter, and 30nJ, 20nJ, 10nJ of pump pulse input energy. OPO gain (Fig. 6.40) and pump depletion (Fig. 6.41) are plotted as a function of signal pulse input energy from 0.1 nJ to 5 nJ. From the plots we can use linear approximation for the gain/depletion dependence on signal pulse energy.

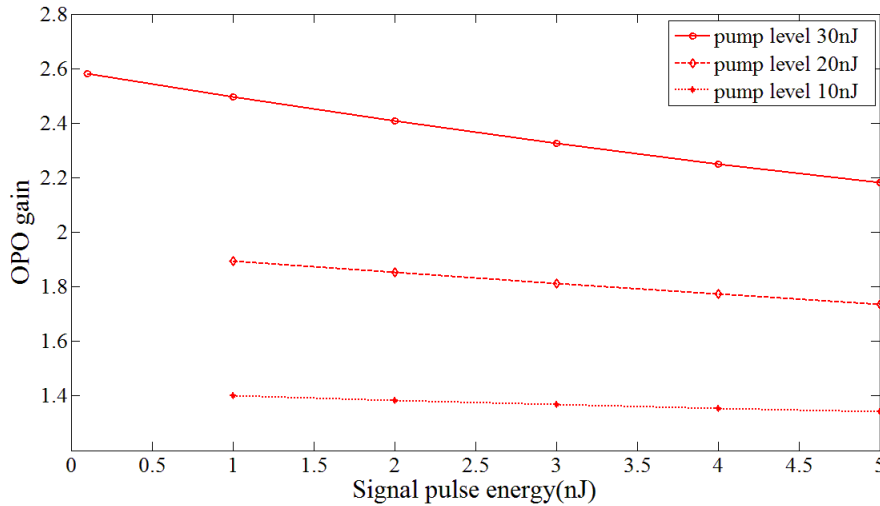


Figure 6.40: OPO gain as a function of signal pulse energy at three pump levels.

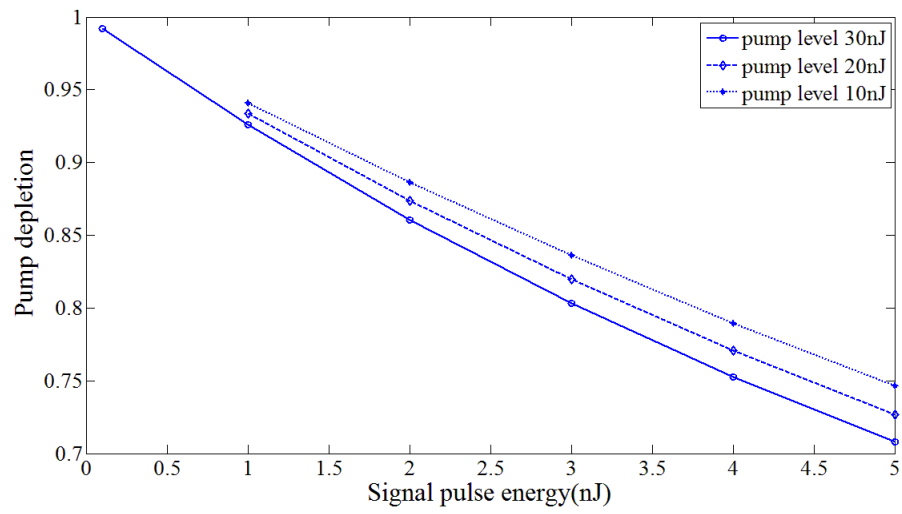


Figure 6.41: Pump depletion as a function of signal pulse energy at three pump levels.

Chapter 7

Simulation of coupled Ti:Sapphire and OPO cavities

A full numerical model of coupled Ti:Sapphire and OPO cavities is established by parameterizing the gain, loss, dispersion and nonlinearities. The pulse evolution of both Ti:Sapphire and OPO is examined at each cavity round trip using the ABCD matrix method in temporal domain introduced in Chapter 2. Conditions to stabilize this complex system are found through the simulation results. The islands of stability are quite narrow, requiring very tight mechanical tolerances and stability. The impact of noise and its dynamics on the stability is also analyzed. This model can also be utilized to analyze and optimize similar systems with gain, loss and chirp.

In section 7.1 we will summarize the problems encountered in the experiment described in Chapter 5. A model of the coupled Ti:Sapphire and OPO cavities is established in section 7.2, including a simplified two-level model for the Ti:sapphire gain, and a matrix approach for the chirp. through energy and chirp evolution .The simulation results are discussed in section 7.3. We finally conclude in section 7.4.

7.1 Presentation of the problems

The intracavity pumped OPO for Intracavity Phase Interferometry (IPI) is sketched in Fig. 7.1 (a).

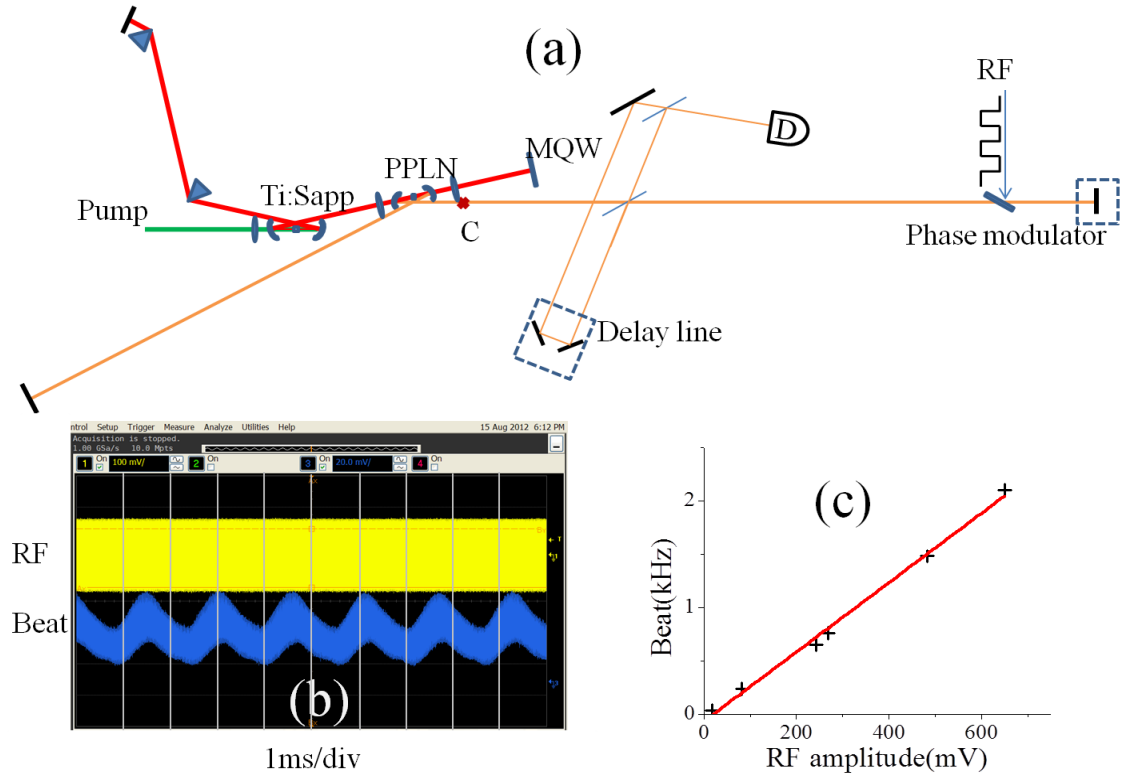


Figure 7.1: (a) Sketch of the intracavity pumped OPO. The OPO cavity is twice as long as the Ti:sapphire cavity. (b) top: signal applied to the modulator. Bottom: beat note signal. (c) Beat note recording versus signal on modulator.

Previous calculations by Andreas Velten of the intracavity pumped OPO [35] showed several types of amplitude instabilities of the pulse train, and suggested negative feedback as a stabilizing mechanism.

1. Q-switching tendency of the Ti:sapphire enhanced by the coupling with the

OPO. Cure:

- a) Use a gain medium with short lifetime (semiconductor, VECSEL)
 - b) Intracavity energy limiter (or simply higher second or higher order non-linear processes inside the signal cavity).
2. Only one pulse in the OPO cavity where 2 are expected. Solutions:
- a) Intracavity energy limiter (or simply higher second or higher order non-linear processes inside the signal cavity)
 - b) OPO cavity length twice that of the pump cavity
3. Cavity detuning: when the cavity lengths are not matched, the OPO signal will drift away from perfect overlap, until it vanishes. Another OPO signal will be generated from noise. In the process: large amplitude instability. Solutions:
- a) Active stabilization

There is a trivial solution to obtain a beat note with (nearly) stable operation: very low intracavity OPO power to avoid nonlinear effects. That led to the beat note shown in Figs. 7.1 (b) and (c). Operating at low power close to threshold is never a desirable situation for any laser. The other recommendation issued from the study of Andreas Velten [35] is to introduce inducible losses. This can be done either by inserting a two-photon absorber in the cavity, or a second harmonic generator, or simply by tightening the focusing into the OPO crystal. Attempting these solutions led to the realization that the problem is considerably more complex than anticipated. The model of Velten addresses only amplitude instabilities. It will be shown in this chapter that phase instabilities are considerably more difficult to eliminate, and have not been taken into account in previous models. Active stabilization as a solution for 3.a). has been unsuccessfully attempted [25, 24]. It seemed logical to use the wavelength of the OPO itself as an error signal, since the excursion in wavelength

is very large for a small cavity length detuning. This unfortunately cannot work, because the laser extinguishes itself first when synchronization is lost, before re-emerging from noise at the synchronized wavelength, and therefore, the instability occurs before the error signal.

The main shortcoming of the previous approach is to have ignored the laser phase fluctuations arising from inexact phase matched processes. In making tighter focusing in the PPLN or adding a nonlinear crystal in the OPO cavity to reduce the amplitude fluctuations mentioned in (1) and (2), we introduce non-phase matched nonlinear processes that couple the fields of the oscillator and pump, in amplitude and phase, and introduce new fields as well. Non-phase-matched processes, that are often neglected, have an important impact on the dynamics of the coupled cavities, contributing for instance to a feedback from the OPO signal to the pump phase and amplitude. The intracavity pumped OPO is a too complex instrument to be developed solely by experimental trial and error. Therefore, a complete theoretical model, containing all the nonlinear interactions, should be elaborated, to understand the mechanisms that affect the stability of the system, and find conditions leading to a stable operation.

7.2 Coupled cavities algorithm

Fig. 7.2 is a block diagram of the algorithm proposed to simulate the coupled OPO-pump cavities. The algorithm is realized in MATLAB, by a main program named “coupledcavities” and two functions named “TiSaph” and “OPO”. An additional function of “jitter” is used when cavity fluctuations are taken into consideration. The code is attached in Appendix B.

It seems reasonable to assume that the pulse shape remains approximately unchanged. The parameters describing the pulse are then limited to the pulse chirp c ,

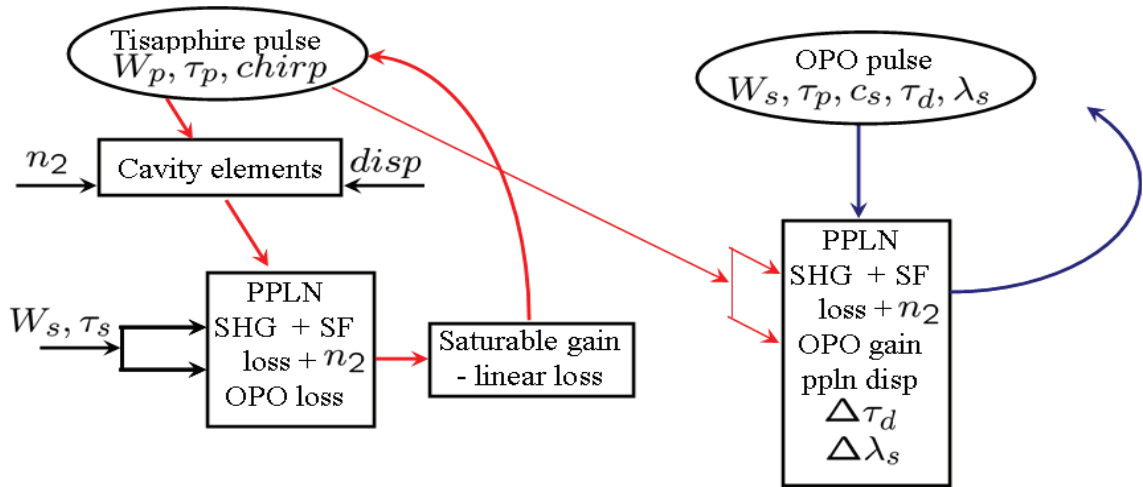


Figure 7.2: Simulation of the intracavity pumped OPO.

the pulse duration τ_p and the pulse energy W_p and the pulse wavelength λ . Some of these are defined by the field envelope:

$$\tilde{\mathcal{E}}(t) = \mathcal{E}_0 e^{-[(\frac{t}{\tau_p})(1+ct)]^2}, \quad (7.1)$$

where \mathcal{E}_0 is a function of W_p . All interactions can be treated in the time domain. It is possible to have a stable (analytical) model of energy evolution, assuming only saturable gain and saturable absorber (see Chapter 5 of [12]).

7.2.1 Energy evolution

Saturated gain and absorption equation for Ti:Sapphire cavity

We chose here to represent the gain and absorber media by a simple two-level system [12], for which analytical expressions define the transmission. It is however an oversimplification in the case of the Ti:sapphire laser. We have therefore adjusted the parameters (gain cross-section, saturation energy) to match the result produced

with a more elaborated 4 level system that used the correct physical parameters for this laser. Defining a gain factor $G = \sigma \Delta N \ell$, where σ is the cross section for stimulated emission, and ℓ is the length of the gain medium traversed per round-trip,

$$\frac{dG}{dt} = -\frac{G - G_e}{T_p} - \frac{G\mathcal{W}}{\mathcal{W}_{gs}T_p}. \quad (7.2)$$

In Eq. (7.2), $\mathcal{W}_{gs} = W_{gs} \times A_g \times \tau_p/T_p$ is an effective saturation energy in the gain medium, where A_g is the cross section of the beam at that location. The steady state ($dG/dt = 0$) solution of Eq. (7.2) is:

$$G(\mathcal{W}) = \frac{G_e}{1 + \frac{\mathcal{W}}{\mathcal{W}_{es}}}. \quad (7.3)$$

The gain equation will be solved *between* pulses (gain recovery). During the pulse, we will integrate separately the gain and absorber equation. We will use both for the gain and the absorber:

$$\frac{dW}{dz} = \alpha W_s (1 - e^{-W/W_s}) \quad (7.4)$$

where W_s is either the saturation energy density, or the energy density at which Kerr lensing turns on. This equation applies for the gain as well, where W_s is then the gain saturation energy density. This equation can be integrated at each passage to yield the pulse energy \mathcal{W}_{g2} after the gain, as a function of the pulse energy \mathcal{W}_{g1} entering the gain medium:

$$\mathcal{W}_{g2} = A_g W_{sg} \ln \left[1 - e^{\alpha_g d_g} \left(1 - e^{\mathcal{W}_{g1}/A_g W_{sg}} \right) \right]. \quad (7.5)$$

The same for the saturable absorber:

$$\mathcal{W}_{g2} = A_a W_{sa} \ln \left[1 - e^{\alpha_a d_a} \left(1 - e^{\mathcal{W}_{a1}/A_a W_{sa}} \right) \right]. \quad (7.6)$$

| physical quantity | symbol | Modifying equation | used for |
|----------------------|---------------|--------------------|------------------------------|
| Pulse Energy | \mathcal{W} | (7.5) | gain/pass |
| Gain factor | G | (7.2) | gain recovery between pulses |
| Pulse energy density | W | (7.6) | saturable abs/pass |

Table 7.1: Gain/loss parameters for the Ti:sapphire laser.

Because of the complex nature of the interactions within the PPLN crystal, it is desirable to use approximation that represent the physical process involved, without going through a detailed calculation of the propagation through the nonlinear crystal at each round-trip. The Ti:sapphire laser is represented by the parameters in table 7.2.1.

OPO gain and pump depletion

With the OPO pumped inside the Ti:Sapphire cavity, we also have to include the energy loss due to the OPO gain in the Ti:Sapphire energy evolution. The gain of the OPO cavity derives solely from the nonlinear interaction with Ti:Sapphire pulse and is studied extensively in Chapter 6 under different conditions.

There are linear losses due to mirror transmission and other elements inside the cavities. The nonlinear loss from non-phase-matched SHG and SFG are is relatively small and can be neglected. This is however not the case for the phase shifts introduced by these processes.

7.2.2 Chirp evolution

In this model we simulate the evolution of chirp and pulse duration using the ABCD matrix method in temporal domain developed in Chapter 2.

The chirp evolution results from dispersion, n_2 of the materials (n_2^{TS} for Ti:Sapphire [12] and n_2^{LN} Lithium Niobate [28]), and from the cascaded nonlinearities ($n_{2,ps}$ for pump and $n_{2,sp}$ for signal).

We make the approximation that the pulse is linearly chirped and Gaussian, and remains so as it cycles through the cavity. The effect of GVD is to create a pulse broadening and a chirp, with the change of the second derivative of the phase versus time equal to:

$$\Delta c_p(disp) = \Delta \left(\frac{\partial^2}{\partial t^2} \varphi(t) \right) = \frac{4k_\ell''}{\tau_p^4} L \quad (7.7)$$

where we have used the characteristics of Gaussian pulse propagation. The chirp induced by self-phase modulation is:

$$\Delta c_p(Kerr) = \Delta \left(\frac{\partial^2}{\partial t^2} \varphi(t) \right) = -\frac{2\pi}{\lambda_\ell} \bar{n}_2 \frac{\partial^2 I}{\partial t^2} L \quad (7.8)$$

For the chirp by cascade nonlinearities, we use the results of numerical calculations. The contributions of SHG and SFG turned out to be small compared to the material Kerr effects. The self and cross phase modulations from DFG are not negligible and more complicated since they involve the intensities of the pump and signal.

7.3 Simulation results

The pulse energy and duration of pump and signal wavelengths are plotted as a function of cavity round trips. Also of interest are the chirp development in the

pump and signal pulses. The delay between signal and pump pulses is another important parameter as it is used in the simulation to determine the synchronization of Ti:Sapphire and OPO cavities. Lastly we examine the signal wavelength evolution as a function of cavity round trips. We are particularly interested in the signal wavelength shifts that track the OPO cavity detuning with respect to the Ti:Sapphire cavity. That will provide the answer to the question whether there is a mechanism for cavity stabilization and what it is.

The pump and signal pulses are chosen to be counter-chirped with an extra chirp in the OPO cavities. This is not a “natural occurrence” in the laser. Without any modification, both the Ti:Sapphire and OPO cavities present positive chirp and negative GVD. As we will see in section 7.3.5, the self-stabilization is only possible when the pump pulse is positively chirped and signal pulse is negatively chirped. The wavelength shift per round trip in this case will compensate for the cavity detuning and return to a stable operation as demonstrated. We can add a negative chirp to the OPO by inserting phase mismatched second harmonic generation in its cavity. A 3 mm long lithium triborate (LBO) is more than enough to create the effective n_2 of $4 \times 10^{-15} \text{ cm}^2/\text{W}$ that is used in the simulation.

We define the equilibrium solution of our model to be stationary pulse energy and pulse duration. If the pulse evolution starts from any non equilibrium value (which is usually the case), the pulse either presents oscillatory behavior until it damps to a steady state, or steps out of the stable zone and never comes back. Fig. 7.3 shows the range of the parameters we choose to generate quasi-equilibrium solutions (oscillatory solutions around equilibrium).

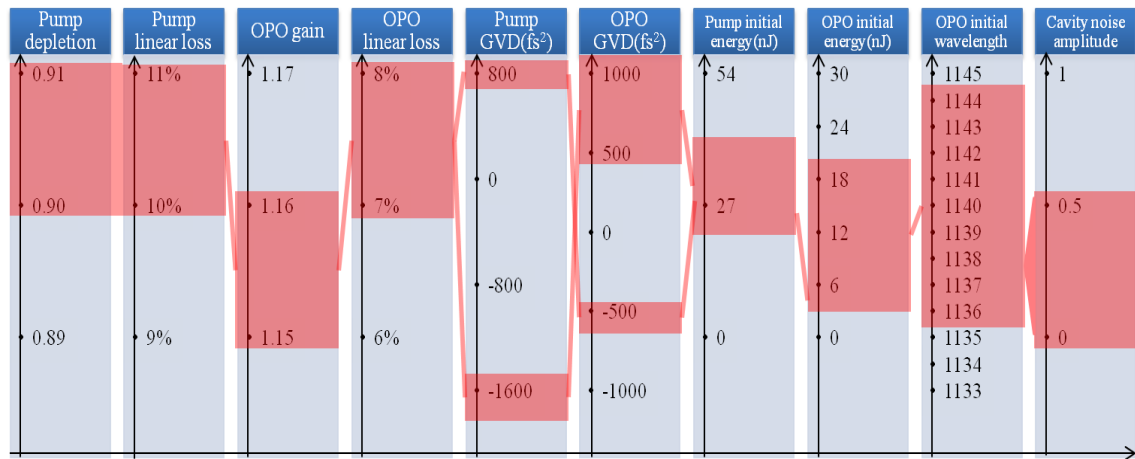


Figure 7.3: The parameters we choose to run the simulation for coupled Ti:Sapphire and OPO cavities are listed in each column. The red indicate the stable range with quasi-equilibrium solutions.

7.3.1 Damping effect

As discussed in Chapter 2, the physical explanation of damping is the cavity lifetime, making the damping time equal to the ring down time of the cavity. This (unproven) assumption is supported by the fact that the oscillation are seen to be damped simply because of the inclusion of gain and loss for both Ti:Sapphire and OPO cavities, as shown in Fig. 7.4, where no damping coefficient ϵ (cf. Chapter 2) was included.

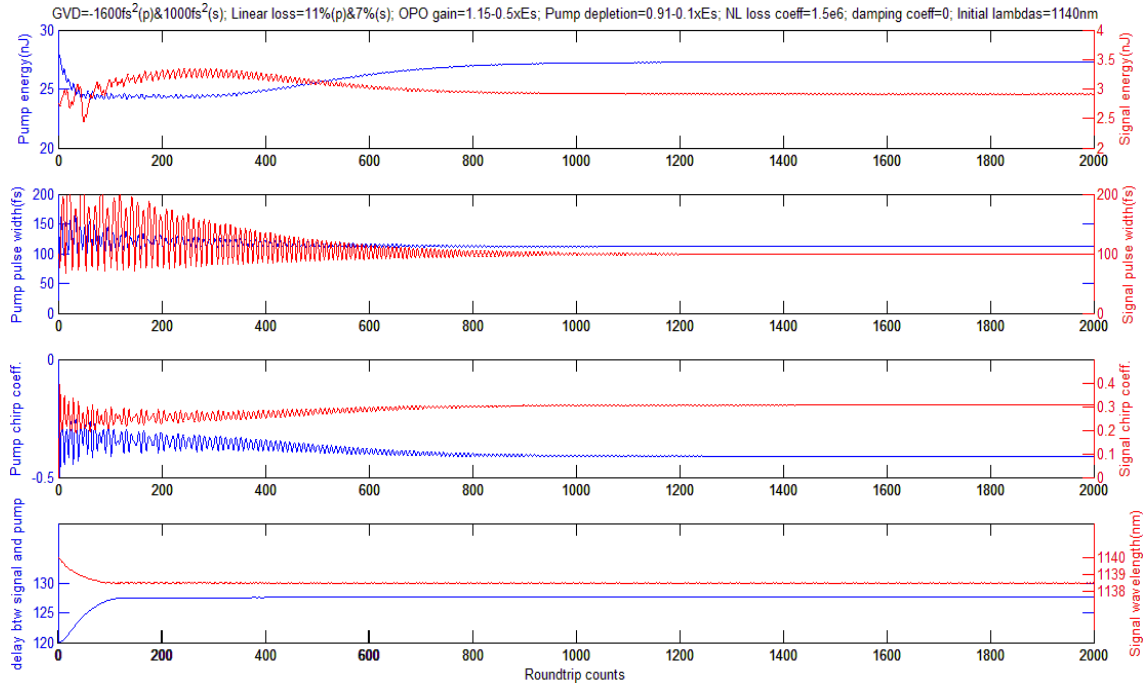


Figure 7.4: $GVD=-1600fs^2$ (pump), $1000fs^2$ (signal); Linear loss=11% (pump), 7% (signal); OPO gain= $1.15-0.5 \times E_s$; Pump depletion= $0.91-0.1 \times E_s$; NL loss coeff= $1.5e6$; Damping coeff=0; Initial lambdas=1140nm

7.3.2 GVD of Ti:Sapphire and OPO cavities

For a pair of counter-chirped pump and signal pulses we need to have opposite signs of the dispersion for Ti:Sapphire and OPO cavities. The GVD of either Ti:Sapphire or OPO cavity will change both pump and signal pulse characteristics. The change of parameters from Fig. 7.4 to Fig. 7.5 is a decrease of the GVD of the Ti:Sapphire cavity from -1600fs^2 to -1700fs^2 . It is seen from the plot that the pump pulse energy decreases while the signal energy increases, and both pump and signal pulses are longer. For the plot of Fig. 7.6, it is the GVD of the OPO cavity that has been decreased from 1000fs^2 to 900fs^2 . The pump pulse energy increases while the signal energy decreases, and both pump and signal pulses become slightly shorter.

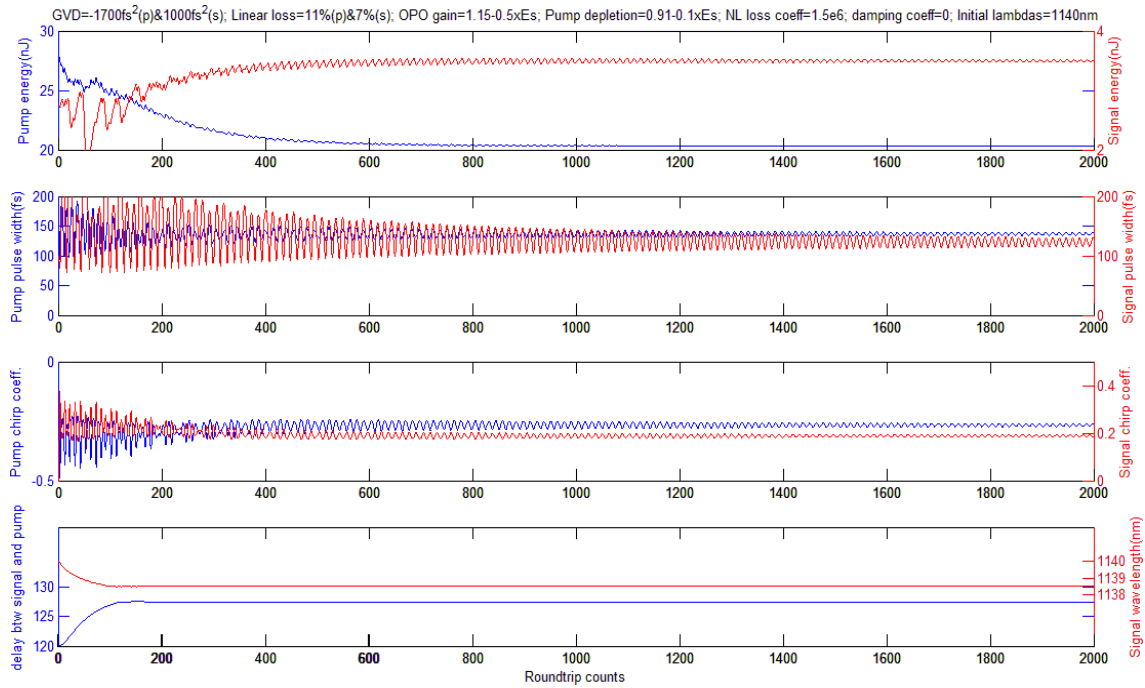


Figure 7.5: $\text{GVD}=-1700\text{fs}^2$ (pump), 1000fs^2 (signal); Linear loss=11% (pump), 7% (signal); OPO gain= $1.15-0.5 \times E_s$; Pump depletion= $0.91-0.1 \times E_s$; NL loss coeff= $1.5e6$; Damping coeff=0; Initial lambdas=1140nm

It is interesting to compare the above results by just flipping the signs in the

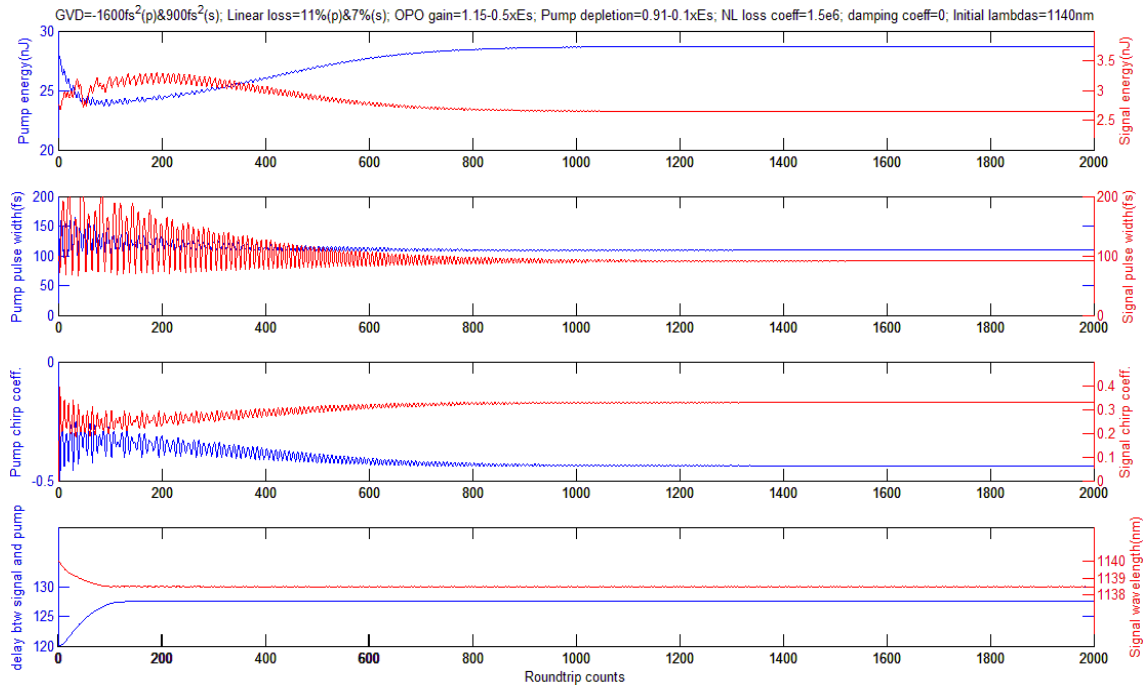


Figure 7.6: $GVD = -1600\text{fs}^2$ (pump), 900fs^2 (signal); Linear loss = 11% (pump), 7% (signal); OPO gain = $1.15 - 0.5 \times E_s$; Pump depletion = $0.91 - 0.1 \times E_s$; NL loss coeff = 1.5×10^{-6} ; Damping coeff = 0; Initial lambda = 1140 nm

two cavities, i.e. the negative chirp being added to the Ti:Sapphire cavity. The new results indicate a much narrower range of stability, along with stronger oscillations (Fig. 7.7).

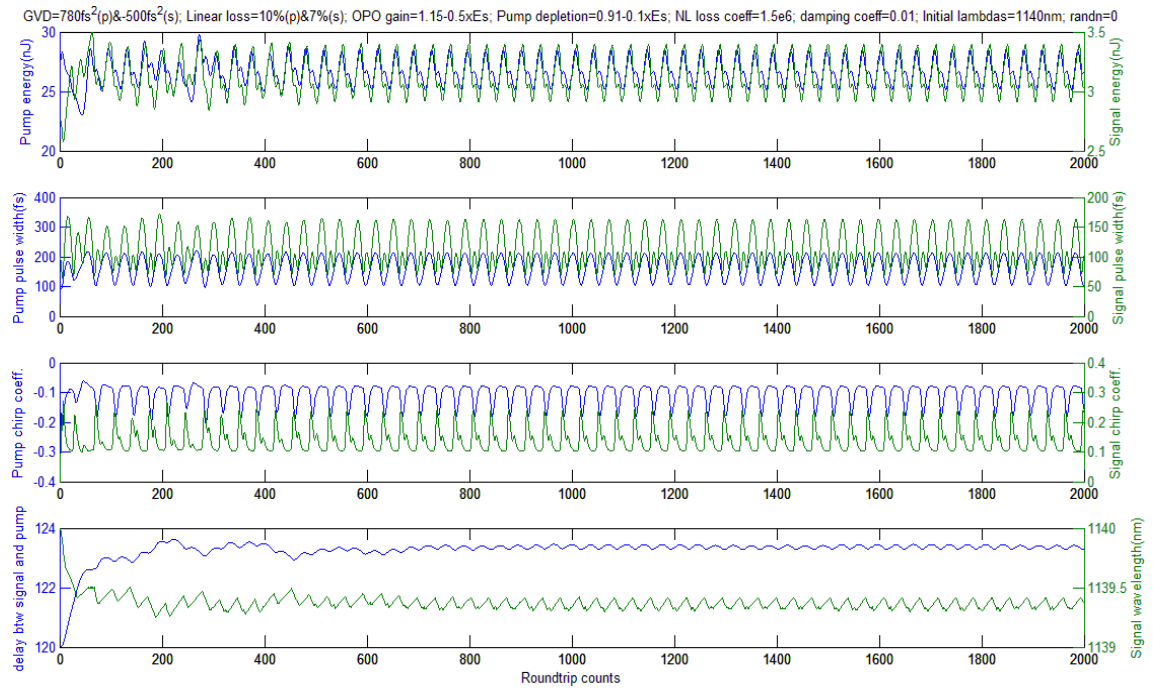


Figure 7.7: $GVD=780fs^2$ (pump), $-500fs^2$ (signal); Linear loss=11% (pump), 7% (signal); OPO gain= $1.15-0.5 \times E_s$; Pump depletion= $0.91-0.1 \times E_s$; NL loss coeff= $1.5e6$; Damping coeff=0; Initial lambdas=1140nm

7.3.3 Gain and loss parameters

Gain and loss parameters are usually adjusted in pairs to obtain equilibrium solutions. Since the gain and saturable absorption of Ti:Sapphire cavity are fixed, we only tune the OPO gain and loss as well as pump depletion due to the OPO interaction. Comparing with Fig. 7.4, Fig. 7.8 is plotted for a 9% increase in OPO gain. The pump pulse energy decreases while the signal energy increases, and both pump and signal pulses are longer. Fig. 7.9 is plotted for an increase in pump depletion and decrease in linear loss of the Ti:Sapphire cavity by 10%. The pump pulse energy increases while the signal energy decreases, and both pump and signal pulses become slightly shorter.

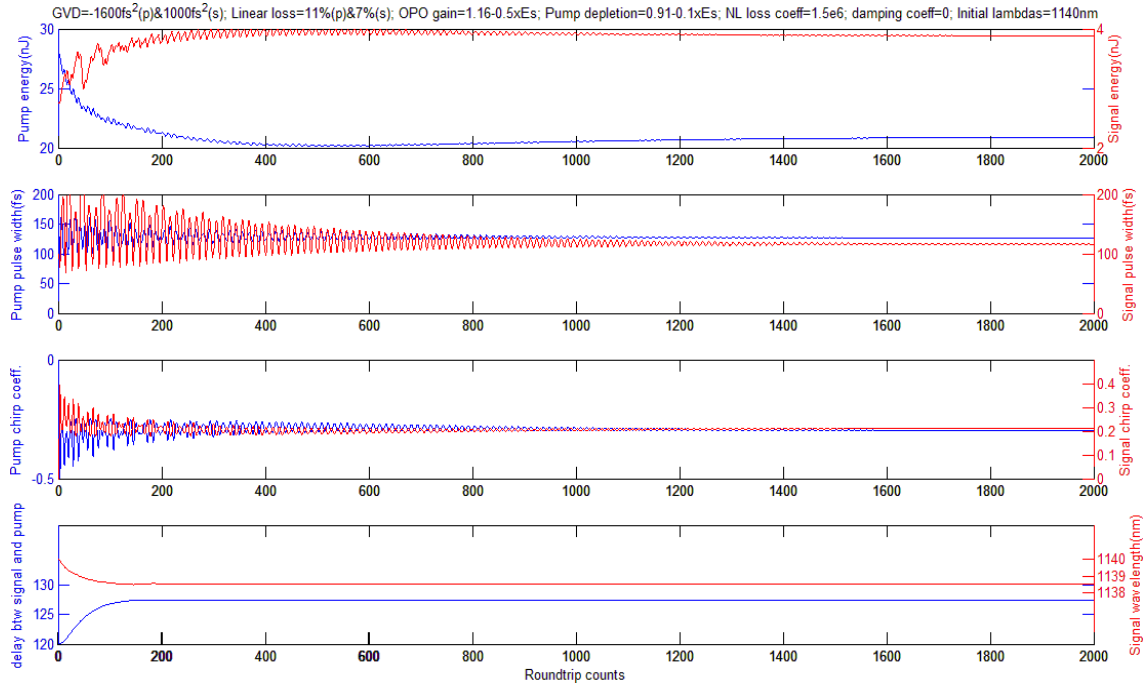


Figure 7.8: $GVD=-1600fs^2$ (pump), $1000fs^2$ (signal); Linear loss=11% (pump), 7% (signal); OPO gain= $1.16-0.5 \times E_s$; Pump depletion= $0.91-0.1 \times E_s$; NL loss coeff= $1.5e6$; Damping coeff=0; Initial lambdas=1140nm

Chapter 7. Simulation of coupled Ti:Sapphire and OPO cavities

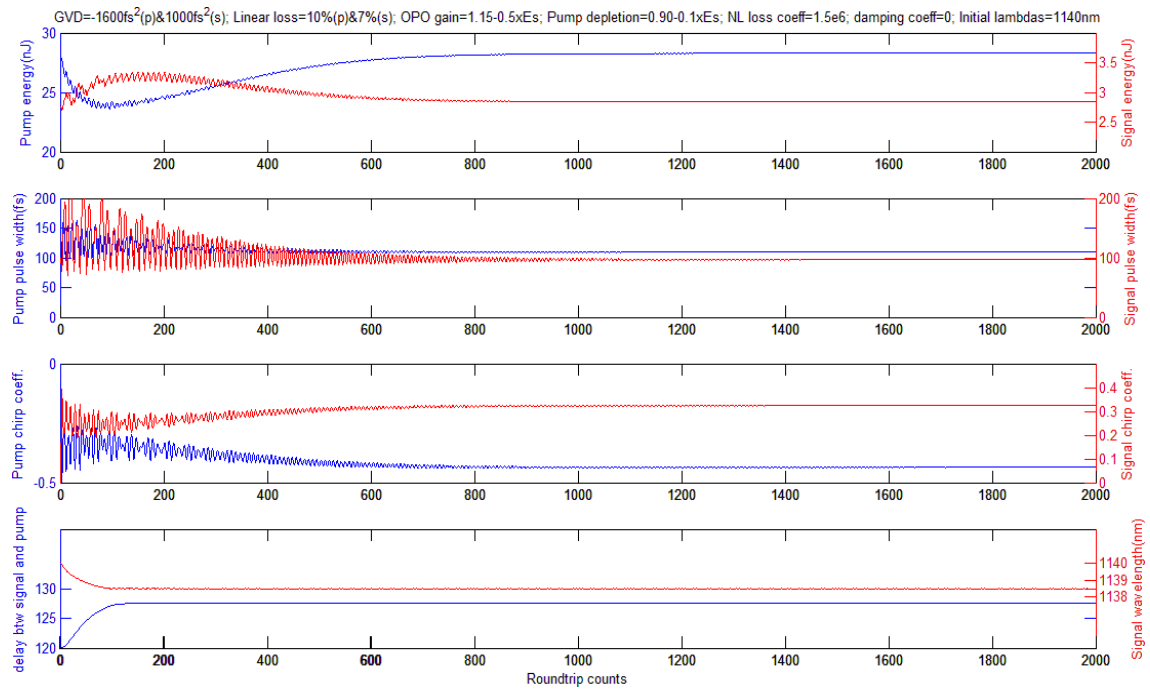


Figure 7.9: GVD=-1600fs² (pump), 1000fs² (signal); Linear loss=10% (pump), 7% (signal); OPO gain=1.15-0.5×E_s; Pump depletion=0.9-0.1×E_s; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm

7.3.4 Initial conditions

The initial signal wavelength, the energy and duration of pump and signal pulses all change the behavior of quasi-equilibrium evolutions, but converge to almost the same final solution. In the following plots we change the initial signal wavelength (Fig. 7.10) and initial pulse energies (Fig. 7.11), respectively.

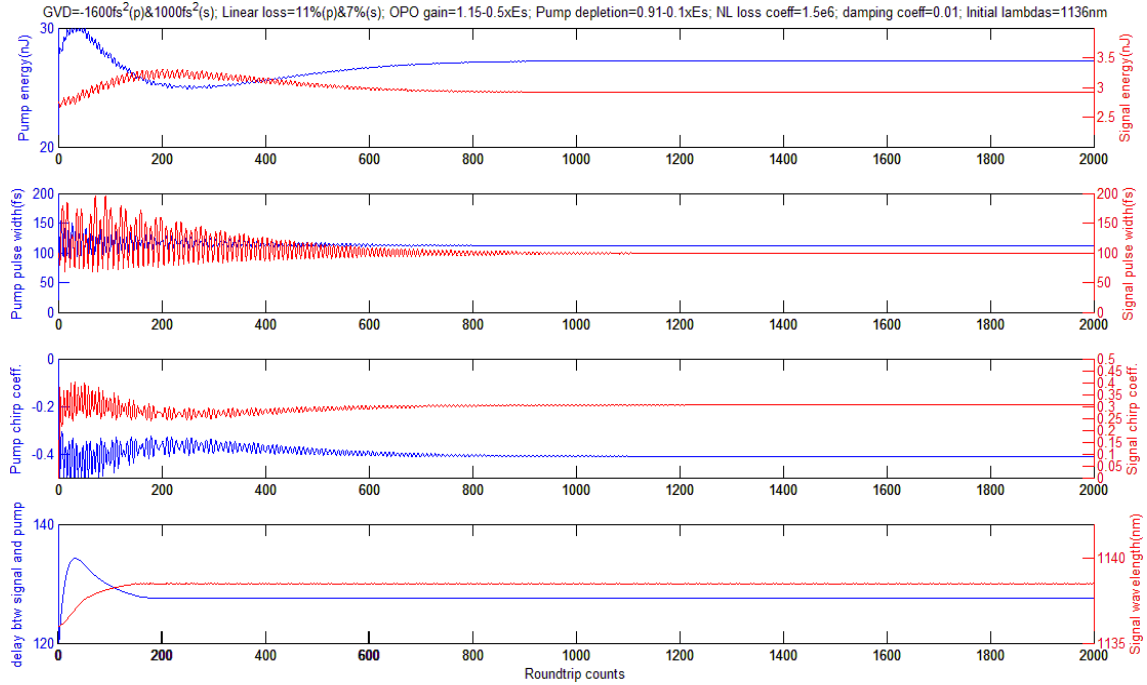


Figure 7.10: $GVD=-1600fs^2$ (pump), $1000fs^2$ (signal); Linear loss=11% (pump), 7% (signal); OPO gain= $1.15-0.5 \times E_s$; Pump depletion= $0.91-0.1 \times E_s$; NL loss coeff= $1.5e6$; Damping coeff=0; Initial lambdas=1136nm

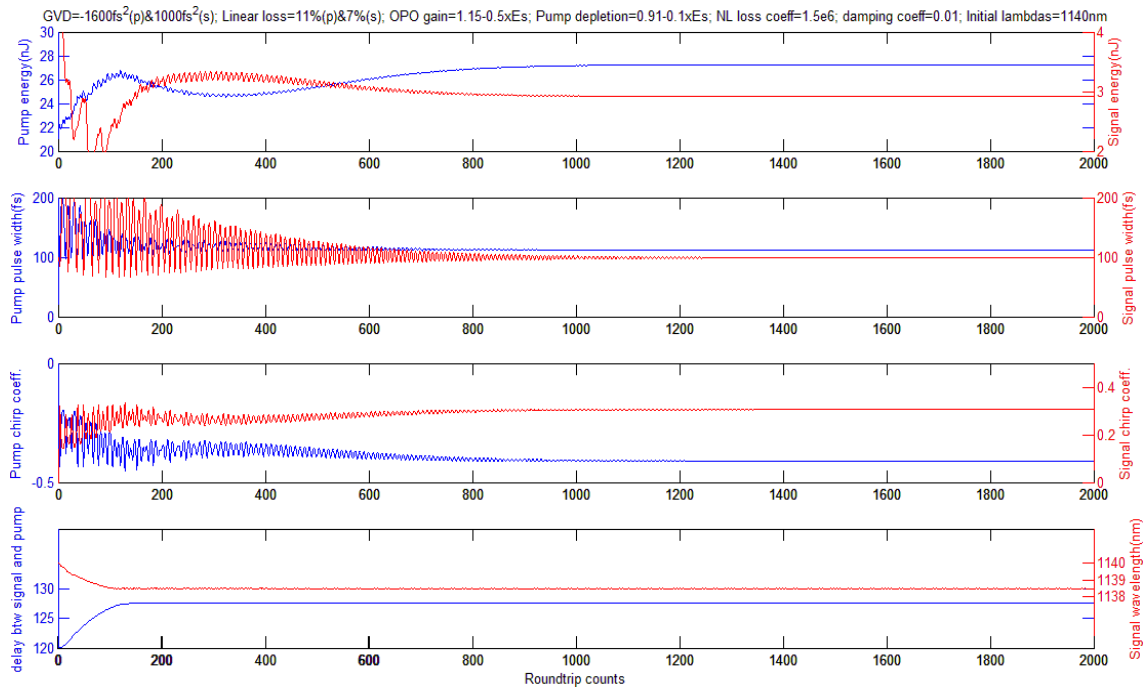


Figure 7.11: $GVD=-1600fs^2$ (pump), $1000fs^2$ (signal); Linear loss=11% (pump), 7% (signal); OPO gain= $1.15-0.5 \times E_s$; Pump depletion= $0.91-0.1 \times E_s$; NL loss coeff= $1.5e6$; Damping coeff=0; Initial lambdas=1140nm. Note: Pump energy starts from 80% as before and signal energy starts from 180%.

7.3.5 Cavity detuning and wavelength tracking

For synchronously pumped OPO, the signal wavelength, within the gain bandwidth, is known to be determined by the cavity length of the OPO. However how the wavelength tracks the repetition rate change remains unclear. In our simulation we find that it is only for a small range of detuning that the wavelength shift per round trip automatically brings it to the demanded wavelength. With the current parameters the OPO cavity can only afford a sudden change from -0.5fs to 1.7fs per round trip, which corresponds to a range of $0.33\mu\text{m}$ change in cavity length. It is thus imperative that the construction of the intracavity pumped OPO be mechanically stable and rigid.

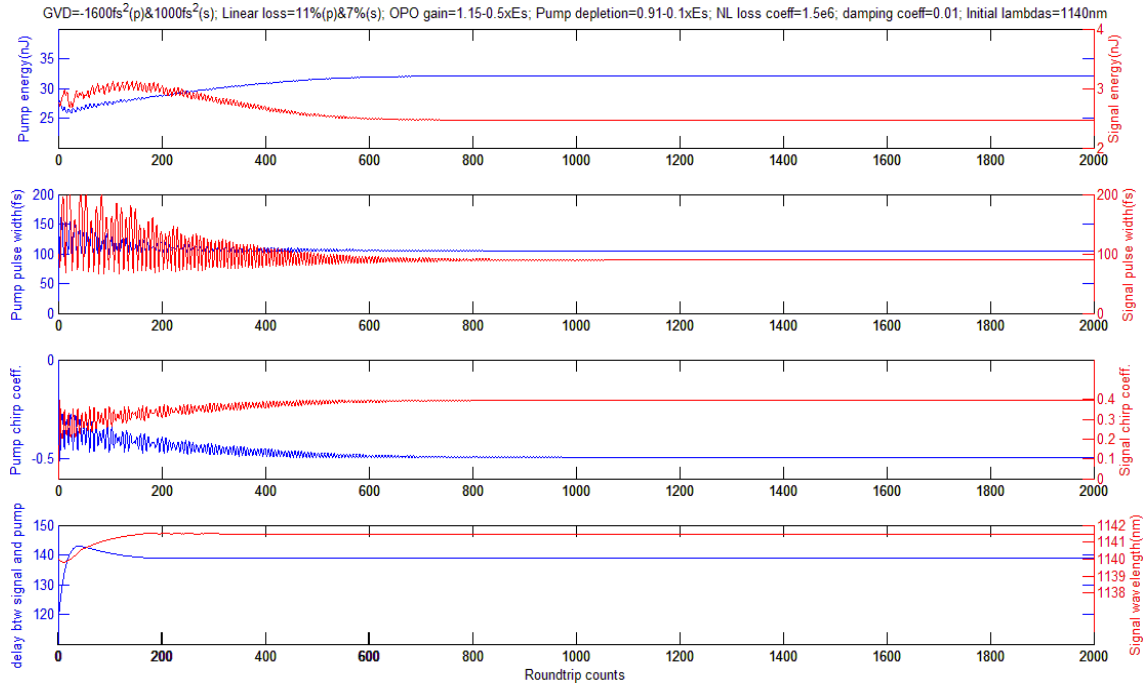


Figure 7.12: $\text{GVD}=-1600\text{fs}^2$ (pump), 1000fs^2 (signal); Linear loss=11% (pump), 7% (signal); OPO gain= $1.15-0.5 \times E_s$; Pump depletion= $0.91-0.1 \times E_s$; NL loss coeff= $1.5e6$; Damping coeff=0; Initial lambda=1140nm. Note: A sudden change of -0.5fs is applied to the cavity round trip time.

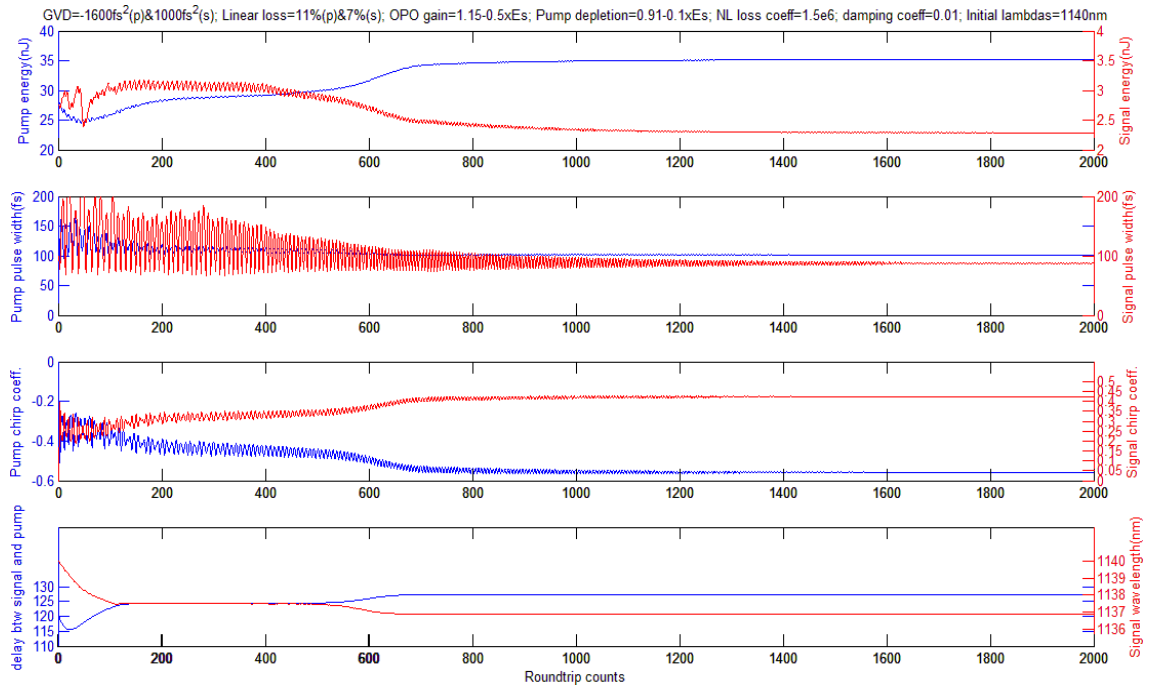


Figure 7.13: GVD=-1600fs² (pump), 1000fs² (signal); Linear loss=11% (pump), 7% (signal); OPO gain=1.15-0.5×E_s; Pump depletion=0.91-0.1×E_s; NL loss coeff=1.5e6; Damping coeff=0; Initial lambdas=1140nm. Note: A sudden change of 1.7fs is applied to the cavity round trip time.

7.3.6 Cavity length fluctuation study

Cavity length fluctuations play an important role in the instability of our system. In this simulation we add to the cavity round trip time a set of normally distributed pseudo-random numbers whose amplitudes are proportional to round trip counts. As we expect, the equilibrium solution becomes noisier than previous results, depending on the magnitude of added noises as well as the noise frequency. The noise frequencies are calculated by Fourier transform and are around 25 MHz (Fig. 7.14), 2.5 MHz (Fig. 7.16) or 0.25 MHz (Fig. 7.18) respectively (the cavity repetition rate is 100 MHz).

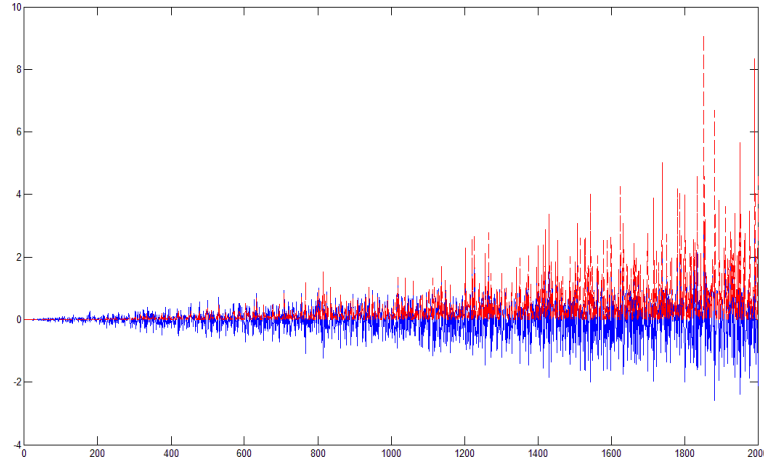


Figure 7.14: The blue solid line is a set of normally distributed pseudorandom numbers plotted as a function of round trip time. The noise amplitude (in fs) is proportional to round trip counts, and frequency is 25 MHz. The red dotted line is the square of the noise.

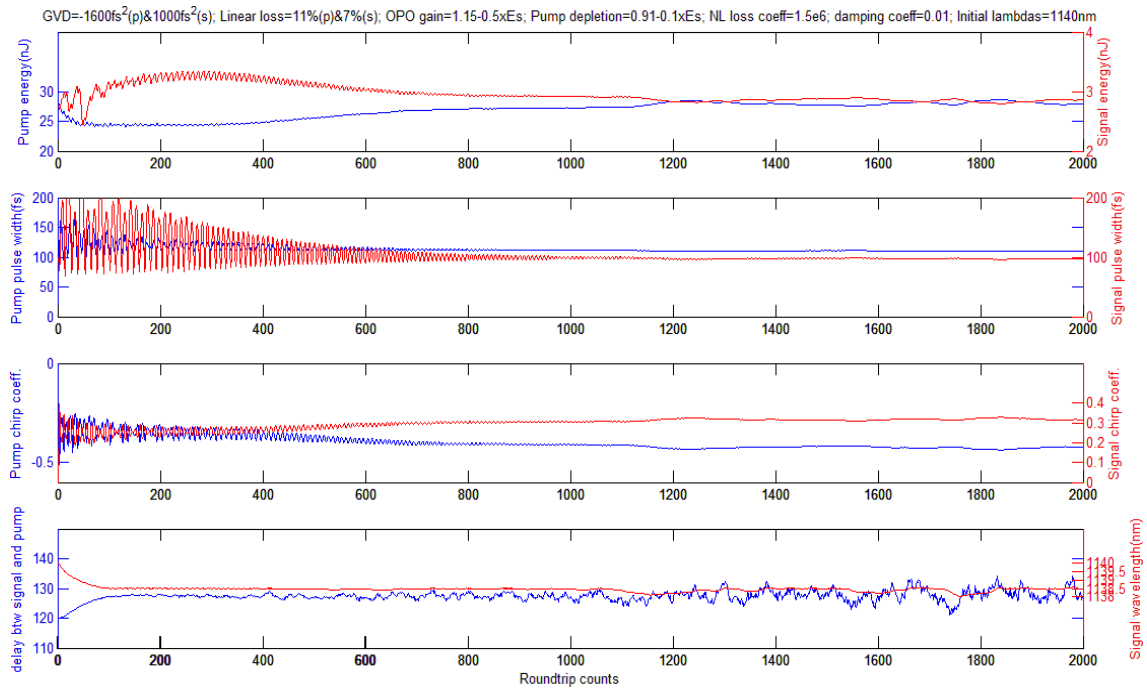


Figure 7.15: $GVD=-1600fs^2$ (pump), $1000fs^2$ (signal); Linear loss=11% (pump), 7% (signal); OPO gain= $1.15-0.5 \times E_s$; Pump depletion= $0.91-0.1 \times E_s$; NL loss coeff= $1.5e6$; Damping coeff=0; Initial lambdas=1140nm. Note:Cavity round trip time fluctuation (Fig. 7.14) is added to the cavity round trip time.

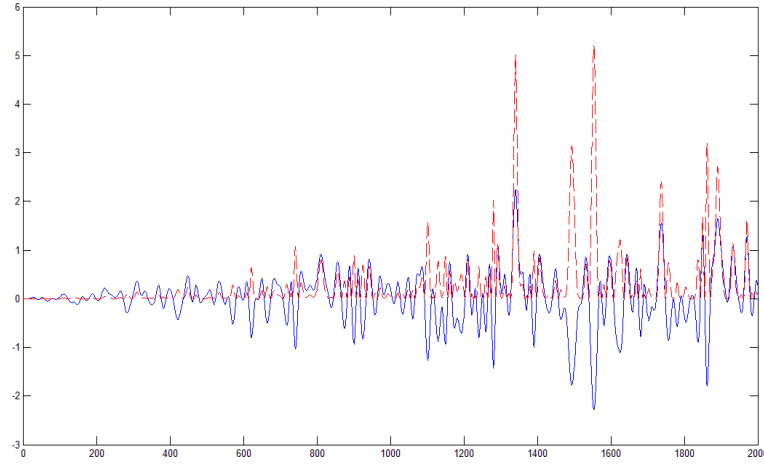


Figure 7.16: The blue solid line is a set of normally distributed pseudorandom numbers plotted as a function of round trip time. The noise amplitude (in fs) is proportional to round trip counts, and frequency is 2.5 MHz. The red dotted line is the square of the noise.

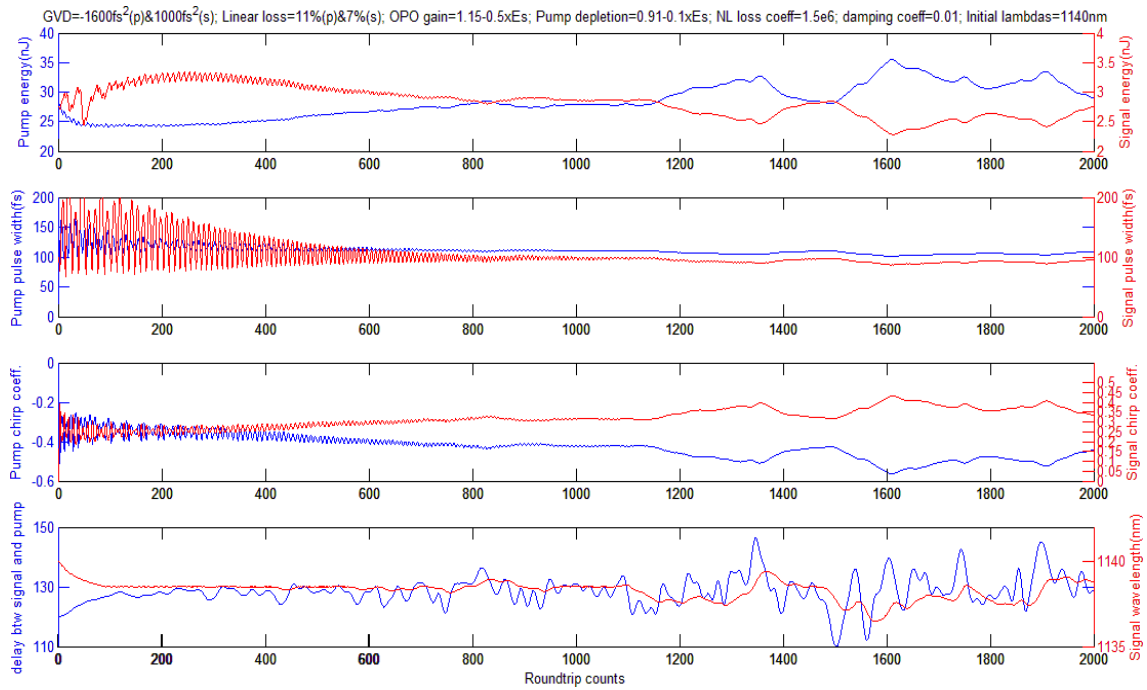


Figure 7.17: $GVD=-1600fs^2$ (pump), $1000fs^2$ (signal); Linear loss=11% (pump), 7% (signal); OPO gain= $1.15-0.5 \times E_s$; Pump depletion= $0.91-0.1 \times E_s$; NL loss coeff= $1.5e6$; Damping coeff=0; Initial lambdas=1140nm. Note: Cavity round trip time fluctuation (Fig. 7.16) is added to the cavity round trip time.

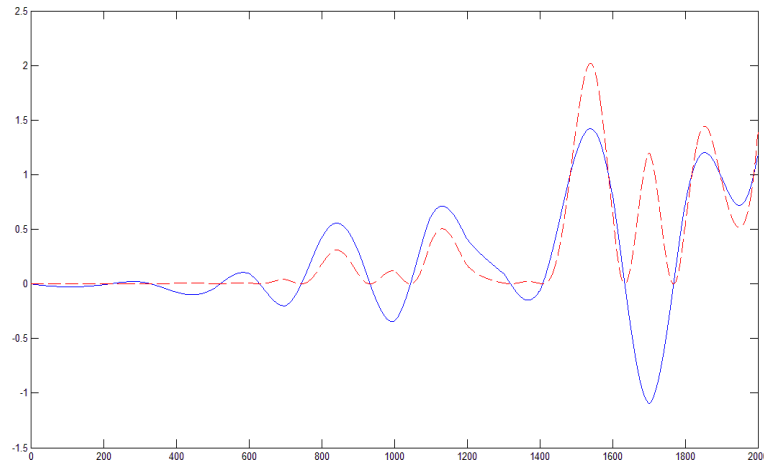


Figure 7.18: The blue solid line is a set of normally distributed pseudorandom numbers plotted as a function of round trip time. The noise amplitude (in fs) is proportional to round trip counts, and frequency is 0.25 MHz. The red dotted line is the square of the noise.

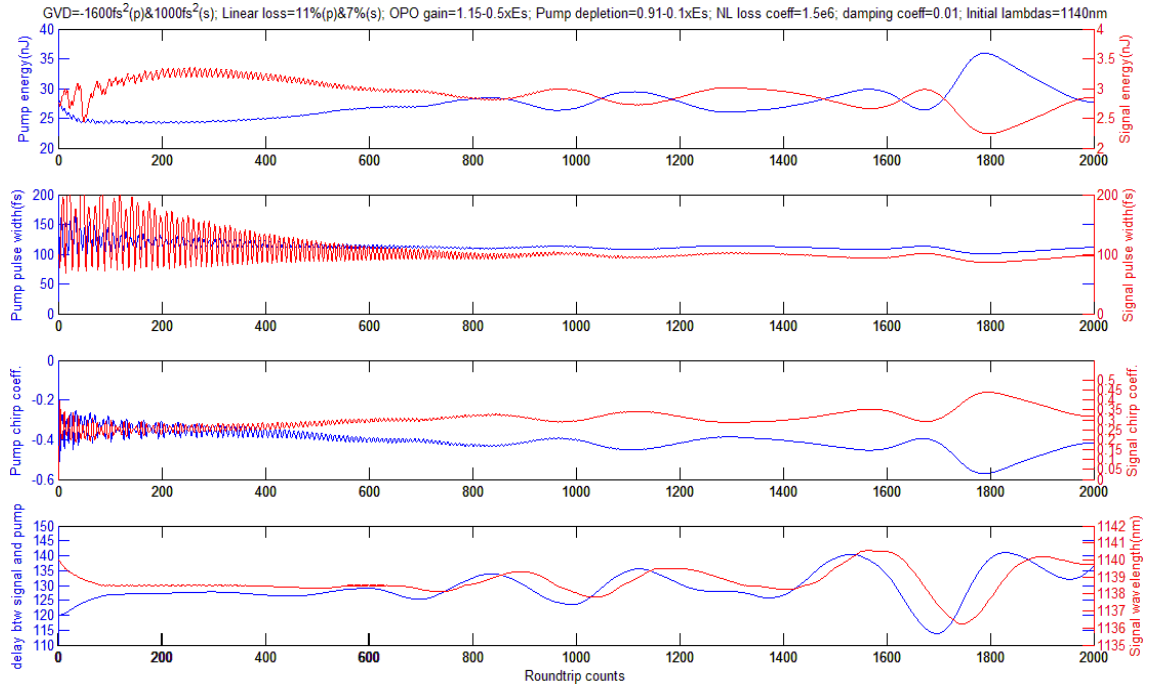


Figure 7.19: $GVD=-1600\text{fs}^2$ (pump), 1000fs^2 (signal); Linear loss=11% (pump), 7% (signal); OPO gain= $1.15-0.5 \times E_s$; Pump depletion= $0.91-0.1 \times E_s$; NL loss coeff= $1.5e6$; Damping coeff=0; Initial lambdas=1140nm. Note:Cavity round trip time fluctuation (Fig. 7.18) is added to the cavity round trip time.

As we can see from these three figures, our system of coupled cavities is more tolerant to faster fluctuations, given the same noise amplitude. The noise amplitude range for which the system can return to stability is only several femtoseconds.

7.4 Conclusion

Intracavity pumped OPO was proposed as an ideal candidate to implement IPI measurements. However the experimental development in Chapter 5 reveals it has never reached its full potential because of the stability problems. In order to study the coupling between pump and signal pulses, the knowledge of nonlinear interaction taken place in PPLN is prepared in Chapter 6. In this chapter we successfully establish a full numerical model of coupled Ti:Sapphire and OPO cavities by parameterizing the gain, loss, dispersion and nonlinearities. Condition to stabilize this complex system is a situation where any change of relative timing of the pump and signal in the OPO leads to a wavelength shift of the signal that will modify its group velocity towards better synchronization. The simulation results show that this condition can be fulfilled if the pump pulse has a positive chirp, and the signal a negative chirp, provided the signal pulse enters the PPLN with some delay with respect to the pump. For a stable steady state to take place in the coupled cavities in experiment, the average nonlinear index should be negative in the OPO cavity. This can be achieved by inserting a non-phase matched second harmonic generating crystal in the OPO cavity.

Chapter 8

Conclusion and future work

8.1 Conclusion

We have demonstrated an IPI response that changes linearly with the displacement ΔL of the reference arm, in the order of a few nanometers with a solid state laser, with a dye jet as saturable absorber. A total displacement of 250 nm in optical path was measured with the resolution better than 10 nm, even with a small fraction of the intracavity power sent to the sample. By replacing the dye jet with a semiconductor absorber we demonstrate a beat note measurement corresponding to a refractive index change in the cavity. It was discovered that the MQW used as saturable absorber introduces a coupling between group and phase velocity, resulting in a beat note bandwidth as large as 10 kHz. Therefore the requirement of IPI that the group velocity be independent of the signal to be measured was not met. A simple theoretical model based on saturation has been shown to match very exactly the shape of the group delay dependence on cavity length [23]. It was therefore concluded that the vanadate laser with saturable absorber was not the ideal system for Intracavity Phase Interferometry.

Intracavity pumped OPO was proposed as an ideal candidate to implement IPI measurements. However the experimental developments reveal it has never reached its full potential because of stability problems. The fact that a stable operation for beat note measurement can only be achieved at low OPO intracavity power, when the feedback from OPO to pump is minimized, indicates a complex coupling between the Ti:Saph and OPO cavities.

A comprehensive theoretical model including all nonlinear interactions (phase matched and mismatched), as well as dispersion to all orders, in the nonlinear crystal of the OPO was elaborated. We found the often neglected phase-mismatched processes to have an important impact on the dynamics of the coupled cavities through cross phase modulation [22]. A full numerical model of coupled pump and OPO cavities is established by parameterizing the gain, loss, dispersion and nonlinearities. The pulse evolution of both pump laser and OPO is calculated at each cavity round trip using the ABCD matrix method in temporal domain. The simulation reproduces the observed instabilities. However, it is found that the same cascade nonlinearities that cause unstable phase and amplitude coupling can be exploited to stabilize the coupled OPO-pump in amplitude, phase and wavelength. The stable cavity uses counter-chirp in the pump and OPO cavities. A negative chirp is produced in the OPO by inserting phase mismatched second harmonic generation in its cavity

8.2 Future work

8.2.1 Fiber based IPI nanoscope

The idea of the nano-metrology we investigated in this thesis can be transferred from free space laser to fiber based systems. The fiber laser is mode locked by a carbon nanotube saturable absorber, which is distributed over a tapered section of

the fiber [18]. The use of polarization maintaining fibers and differentiating the two pulses by two orthogonal polarizations is a promising solution to either the coupling between two pulses or the coupling between two cavities. The setup is illustrated in Fig. 8.1.

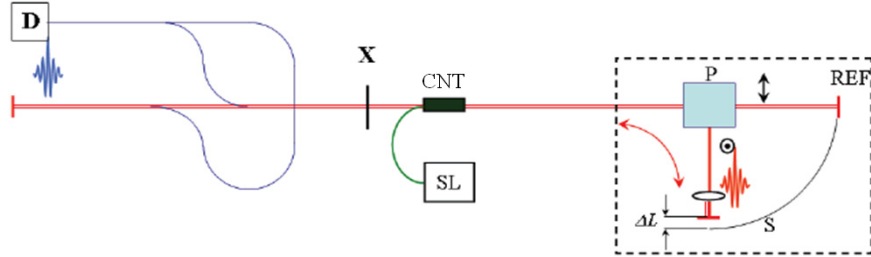


Figure 8.1: Sketch of the fiber laser sensor. Two pulses are made to oscillate in a cavity made of polarization maintaining (PM) fibers, one propagating on the slow axis, and the other on the fast axis. A X indicates the crossing point of the two pulses, where a carbon nanotube (CNT) saturable absorber is located both for mode-locking and for imposing the pulse crossing point. At the right end of the cavity, a polarizing beam splitter P separates the two pulses, one being sent to a reference arm, the other to the arm with the sensing element. On the left side of the cavity, output couplers and beam combiners make the two output pulse trains interfere on a detector D

One can measure the difference in optical path lengths when there is a transparent element in the sample arm. It is possible to reconstruct a three dimensional image using tomographic approach, given the sample sitting on a translation and rotation stage. The portability and feasibility of fiber will make it suitable nanoscopy in biomedical applications.

This system can be further improved by an optical parametric oscillator, synchronously pumped by the mode-locked fiber laser. Controlling the pulse timing by pulsing the gain is conceptually the ideal configuration of the IPI. The approach seems promising, Gowda et al. having recently demonstrated a bidirectional OPO fiber laser [16].

8.2.2 Frequency domain Optical Parametric Amplification

Our study of second order nonlinear interactions can be applied to many research areas. For instance, amplification of few-cycle laser pulses with their corresponding octave spanning spectra still remains a formidable challenge since the universal dilemma of gain narrowing sets limits for both real level pumped amplifiers as well as parametric amplifiers. Bruno Schmidt et al. demonstrate that employing parametric amplification in the frequency domain rather than in time domain opens up new design opportunities for ultrafast laser science, with the potential to generate single cycle multi-terawatt pulses [21]. Our theory of parametric interactions is valid down to a few optical cycles, the only limitation being that the spectra of the different pulses (signal, pump and idler) may not overlap. It will be a suitable tool to simulate various effects and predict performance, given the experimental parameters.

We can also further extend our approach to other nonlinear interactions. For instance, Raman scattering and four-wave mixing have wide applications, but their simulations are still under development, especially for ultrafast pulses. Our simulation deals with complex electric field and provides the information both in amplitude and phase.

References

- [1] Ladan Arissian and Jean-Claude Diels. Investigation of carrier to envelope phase and repetition rate — fingerprints of mode-locked laser cavities. *Journal of Physics B: At. Mol. Opt. Phys*, 42:183001, 2009.
- [2] Ladan Arissian, Jean-Claude Diels, Andreas Stintz, and Vaclav Kubecek. Multiple quantum wells for ring and linear lasers with long lifetime gain. In Shori Hoffman, editor, *Solid State lasers IX*, volume 5707-52, pages 295–301, San Jose, CA, 2005. SPIE.
- [3] C. V. Bennett and B. H. Kolner. Principles of parametric temporal imaging, part i: system configurations. *IEEE Journal of Quantum Electronics*, 36:430–437, 2000.
- [4] J. Biegert and J. C. Diels. Compression of pulses of a few optical cycles through harmonic conversion. *Journal of Optical Society B*, 18:1218–1226, 2001.
- [5] Jens Biegert, Vaclav Kubecek, and Jean-Claude Diels. Pulse compression: Type II second harmonic pulse compression. In John G. Webster, editor, *Encyclopedia of Electrical and Electronics Engineering (#17)*, volume 17, pages 446–454. IEEE, 1998.
- [6] Matthew J. Bohn, Jean-Claude Diels, and R. K. Jain. Measuring intracavity phase changes using double pulses in a linear cavity. *Optics Lett.*, 22:642–644, 1997.
- [7] R. Boyd. *Nonlinear Optics*. Academic Press, Boston, 1977.
- [8] Kent C. Burr, C. L. Tang, Mark A. Arbore, and Martin M. Fejer. High-repetition-rate femtosecond optical parametric oscillator based on periodically poled lithium niobate. *Appl. Phys. Lett.*, 70:3341–3343, 1997.

References

- [9] K Y Chen, W Liu, Y H Feng, and W X Li. Optimum design and performance of a 1.4 ghz loop-gap resonator used in an esr spectrometer. *J. Phys. E: Sci. Instrum.*, 21:660–662, 1987.
- [10] Richard DeSalvo, David J. Hagan, G. Stegeman, and Eric W. Van Stryland. Self-focusing and self-defocusing by cascaded second-order effects in KTP. *Optics Letters*, 17:28–30, 1993.
- [11] J.-C. Diels, Xuan Luo, Xiaozhen Xu, Koji Masuda, and Ladan Arissian. Definitions and control of the CEP and CEO. *Laser Physics*, 20:1038–1043, 2010.
- [12] J.-C. Diels and Wolfgang Rudolph. *Ultrashort laser pulse phenomena*. Elsevier, ISBN 0-12-215492-4; second edition, Boston, 2006.
- [13] W. Dietel, J. J. Fontaine, and J.-C. Diels. Intracavity pulse compression with glass: a new method of generating pulses shorter than 60 femtoseconds. *Optics Letters*, 8:4–6, 1983.
- [14] J. J. Fontaine, W. Dietel, and J.-C. Diels. Chirp in a mode-locked ring dye laser. *IEEE J. of Quantum Electron.*, QE-19:1467, 1983.
- [15] W. Froncisz and J. S. Hyde. The loop-gap resonator: a new microwave lumped circuit esr sample structure. *J. Magn. Res.*, 47:515–521, 1982.
- [16] R. Gowda, K. Kieu, J-C. Diels, R. Norwood, and N. Peyghambarian. All-fiber bidirectional optical parametric oscillator. *manuscript in preparation*, 2013.
- [17] D. C. Hutchings, M. Sheik-Bahae, D. J. Hagan, and E. W. Van Stryland. Kramers-kronig relations in nonlinear optics. *Opt. and Quant. Electr.*, 24:1–30, 1992.
- [18] K. Kieu and M. Mansuripur. All-fiber bidirectional passively mode-locked ring laser. *Opt. Lett.*, 33:64–66, 2008.
- [19] H. W. Kogelnik and T. Li. Laser beams and resonators. *Appl. Opt.*, 5:1550–1567, 1966.
- [20] B. H. Kolner and M. Nazarathy. Temporal imaging with a time lens. *Optics Lett.*, 14:630–632, 1989.
- [21] F. Legare. High harmonic generation with intense ultrashort infrared laser pulses. Hotel La Sapinière, Val David, Quebec, 2013.
- [22] Xuan. Luo, L. Arissian, and J.-C. Diels. Stability analysis of an intracavity pumped opo. In *CLEO, 2013*, page JW2a.27 (147), San Jose, CA, 2013. Optical Society of America.

References

- [23] Xuan Luo, Ladan Arissian, Koji Masuda, and Jean-Claude Diels. Group and phase velocity coupling of colliding pulses in a nanostructure. *Opt. Lett.*, 36:3885–3887, 2011.
- [24] Xianmei Meng. *Ultra-short Pulse Optical Parametric Oscillator Sensor*. PhD thesis, University of New Mexico, Albuquerque, New Mexico, 2003.
- [25] Xianmei Meng, Raphael Quintero, and Jean-Claude Diels. Intracavity pumped optical parametric bidirectional ring laser as a differential interferometer. *Opt. Comm*, 233:167–172, 2004.
- [26] M. Navarro, O. Chalus, and Jean-Claude Diels. Mode-locked ring lasers for backscattering measurement of mirror. *Optics Letters*, 31:2864–2866, 2006.
- [27] J. R. O’Connor. Unusual crystal field energy levels and efficient laser properties of $\text{YVO}_4:\text{Nd}$. *Applied Physics Letters*, 9:407, 1966.
- [28] L. Palfalvi, J. Hebling, J. Kuhl, A. Peter, and K. Polgar. Temperature dependence of the absorption and refraction of mg-doped congruent and stoichiometric LiNbO_3 in the thz range. *Journal of Applied Physics*, 97(12):123505–123505–6, 2005.
- [29] I. S. Ruddock and D. J. Bradley. Bandwidth-limited subpicosecond pulse generation in mode-locked cw dye lasers. *Appl. Phys. Lett.*, 29:296, 1976.
- [30] Andreas Schmitt-Sody, Ladan Arissian, Andreas Velten, Jean-Claude Diels, and Dave Smith. Rabi cycling of two pulses in a mode-locked ring laser cavity with electro-optical control. *Physical Review A*, 78:063802, 2008.
- [31] Andreas Schmitt-Sody, Jean-Claude Diels, and Ladan Arissian. Ultra-slow dynamics of an ultra-fast laser. In *QELS, 2007*, page 130 (QWC4), Baltimore, MA, 2007. Optical Society of America.
- [32] Andreas Schmitt-Sody, Andreas Velten, Koji Masuda, and Jean-Claude Diels. Intra-cavity mode locked laser magnetometer. *Optics Communications*, 283:3339–3341, 2010.
- [33] Andreas Velten and Jean-Claude Diels. *Optical parametric oscillators for Intracavity Phase Interferometry: Performing ultra-sensitive measurements inside a mode-locked cavity*. PhD thesis, The University of New Mexico, Albuquerque, New Mexico, 2010.
- [34] Andreas Velten, Andreas Schmitt-Sody, and Jean-Claude Diels. Precise intracavity phase measurement in an optical parametric oscillator with two pulses per cavity round-trip. *Optics Letters*, 35:1181–1183, 2010.

References

- [35] Andreas Velten, Alena Zavadilova, Vaclav Kubecek, and Jean-Claude Diels. Instabilities in intracavity pumped optical parametric oscillators and methods of stabilization. *Applied Physics B*, 98:13–25, 2010.
- [36] D. E. Zelmon, D. L. Small, and D. Jundt. Infrared corrected sellmeier coefficients for congruently grown lithium niobate and 5 mol. % magnesium oxide doped lithium niobate. *J. Opt. Soc. Am. B*, 14:3319, 1997.

Appendix A

Coupled NL wave equations derived in frequency domain

The various polarization have been defined in Eq. (6.5) of Chapter 6. Most calculations being performed in the frequency domain, the Fourier transform of these equations will be used:

$$\begin{aligned}
 \tilde{\mathbf{P}}_p^{NL}(\Omega, z) &= \epsilon_0 \left[\chi_1^{(2)} \int \tilde{E}_s(\Omega', z) \tilde{E}_i(\Omega - \Omega', z) d\Omega' + \chi_2^{(2)} \int \tilde{E}_p^*(\Omega', z) \tilde{E}_2(\Omega + \Omega', z) d\Omega' \right. \\
 &\quad \left. + \chi_3^{(2)} \int \tilde{E}_s^*(\Omega', z) \tilde{E}_3(\Omega + \Omega', z) d\Omega' \right] \\
 \tilde{\mathbf{P}}_s^{NL}(\Omega, z) &= \epsilon_0 \left[\chi_1^{(2)} \int \tilde{E}_i^*(\Omega', z) \tilde{E}_p(\Omega + \Omega', z) d\Omega' + \chi_3^{(2)} \int \tilde{E}_p^*(\Omega', z) \tilde{E}_3(\Omega + \Omega', z) d\Omega' \right] \\
 \tilde{\mathbf{P}}_i^{NL}(\Omega, z) &= \epsilon_0 \chi_1^{(2)} \int \tilde{E}_s^*(\Omega', z) \tilde{E}_p(\Omega + \Omega', z) d\Omega' \\
 \tilde{\mathbf{P}}_2^{NL}(\Omega, z) &= \epsilon_0 \chi_2^{(2)} \int \tilde{E}_p(\Omega', z) \tilde{E}_p(\Omega - \Omega', z) d\Omega' \\
 \tilde{\mathbf{P}}_3^{NL}(\Omega, z) &= \epsilon_0 \chi_3^{(2)} \int \tilde{E}_p(\Omega', z) \tilde{E}_s(\Omega - \Omega', z) d\Omega' \tag{A.1}
 \end{aligned}$$

Inserting these five polarizations into the wave Eq. (6.1), and grouping terms of the same central frequency, leads to the set of five equations:

$$\begin{aligned}
 \left(\frac{\partial^2}{\partial z^2} + \Omega^2 \mu \epsilon(\Omega) \right) \tilde{E}_p(\Omega, z) &= -\epsilon_0 \mu_0 \Omega^2 \left[\chi_1^{(2)} \int \tilde{E}_s(\Omega', z) \tilde{E}_i(\Omega - \Omega', z) d\Omega' \right. \\
 &\quad + \chi_2^{(2)} \int \tilde{E}_p^*(\Omega', z) \tilde{E}_2(\Omega + \Omega', z) d\Omega' \\
 &\quad \left. + \chi_3^{(2)} \int \tilde{E}_s^*(\Omega', z) \tilde{E}_3(\Omega + \Omega', z) d\Omega' \right] \\
 \left(\frac{\partial^2}{\partial z^2} + \Omega^2 \mu \epsilon(\Omega) \right) \tilde{E}_s(\Omega, z) &= -\epsilon_0 \mu_0 \Omega^2 \left[\chi_1^{(2)} \int \tilde{E}_i^*(\Omega', z) \tilde{E}_p(\Omega + \Omega', z) d\Omega' \right.
 \end{aligned}$$

Appendix A. Coupled NL wave equations derived in frequency domain

$$\begin{aligned}
& +\chi_3^{(2)} \int \tilde{E}_p^*(\Omega', z) \tilde{E}_3(\Omega + \Omega', z) d\Omega' \Big] \\
\left(\frac{\partial^2}{\partial z^2} + \Omega^2 \mu \epsilon(\Omega) \right) \tilde{E}_i(\Omega, z) &= -\epsilon_0 \mu_0 \Omega^2 \chi_1^{(2)} \int \tilde{E}_s^*(\Omega', z) \tilde{E}_p(\Omega + \Omega', z) d\Omega' \\
\left(\frac{\partial^2}{\partial z^2} + \Omega^2 \mu \epsilon(\Omega) \right) \tilde{E}_2(\Omega, z) &= -\epsilon_0 \mu_0 \Omega^2 \chi_2^{(2)} \int \tilde{E}_p(\Omega', z) \tilde{E}_p(\Omega - \Omega', z) d\Omega' \\
\left(\frac{\partial^2}{\partial z^2} + \Omega^2 \mu \epsilon(\Omega) \right) \tilde{E}_3(\Omega, z) &= -\epsilon_0 \mu_0 \Omega^2 \chi_3^{(2)} \int \tilde{E}_p(\Omega', z) \tilde{E}_s(\Omega - \Omega', z) d\Omega'. \quad (\text{A.2})
\end{aligned}$$

A description in complex spectral amplitude $\tilde{a}_{p,s,i,2,3}(\Omega, z)$ and fast varying spectral phase $-ik_{p,s,i,2,3}(\Omega)z$ is next chosen for each pulse:

$$\tilde{E}_{p,s,i,2,3}(\Omega, z) = \frac{1}{2} \tilde{a}_{p,s,i,2,3}(\Omega, z) e^{-ik_{p,s,i,2,3}(\Omega)z} \quad (\text{A.3})$$

where Maxwell's wave equation imposes the condition

$$k^2(\Omega) = \Omega^2 \mu \epsilon(\Omega). \quad (\text{A.4})$$

Substituting in the five equations (A.2) leads to a set of five differential equations for the evolution with distance of the five complex spectral amplitude functions:

$$\begin{aligned}
\frac{\partial \tilde{a}_p(\Omega, z)}{\partial z} &= -i \frac{\Omega^2}{4c^2 k_p(\Omega)} \left\{ \chi_1^{(2)} \int \tilde{a}_s(\Omega', z) \tilde{a}_i(\Omega - \Omega', z) e^{i[-k_s(\Omega') - k_i(\Omega - \Omega') + k_p(\Omega)]z} d\Omega' \right. \\
&+ \chi_3^{(2)} \int \tilde{a}_s^*(\Omega', z) \tilde{a}_3(\Omega + \Omega', z) e^{i[k_s(\Omega') - k_3(\Omega + \Omega') + k_p(\Omega)]z} d\Omega' \\
&+ \left. \chi_2^{(2)} \int \tilde{a}_p^*(\Omega', z) \tilde{a}_2(\Omega + \Omega', z) e^{i[k_p(\Omega') - k_2(\Omega + \Omega') + k_p(\Omega)]z} d\Omega' \right\} \\
&- \frac{i}{2k_p(\Omega)} \frac{\partial^2}{\partial z^2} \tilde{a}_p(\Omega, z) \\
\frac{\partial \tilde{a}_s(\Omega, z)}{\partial z} &= -i \frac{\Omega^2}{4c^2 k_s(\Omega)} \left\{ \chi_1^{(2)} \int \tilde{a}_i^*(\Omega', z) \tilde{a}_p(\Omega + \Omega', z) e^{i[k_i(\Omega') - k_p(\Omega + \Omega') + k_s(\Omega)]z} d\Omega' \right. \\
&+ \left. \chi_3^{(2)} \int \tilde{a}_p^*(\Omega', z) \tilde{a}_3(\Omega + \Omega', z) e^{i[k_p(\Omega') - k_3(\Omega + \Omega') + k_s(\Omega)]z} d\Omega' \right\} \\
&- \frac{i}{2k_s(\Omega)} \frac{\partial^2}{\partial z^2} \tilde{a}_s(\Omega, z) \\
\frac{\partial \tilde{a}_i(\Omega, z)}{\partial z} &= -i \frac{\Omega^2 \chi_1^{(2)}}{4c^2 k_i(\Omega)} \int \tilde{a}_s^*(\Omega', z) \tilde{a}_p(\Omega + \Omega', z) e^{i[k_s(\Omega') - k_p(\Omega + \Omega') + k_i(\Omega)]z} d\Omega' \\
&- \frac{i}{2k_i(\Omega)} \frac{\partial^2}{\partial z^2} \tilde{a}_i(\Omega, z)
\end{aligned}$$

Appendix A. Coupled NL wave equations derived in frequency domain

$$\begin{aligned}
\frac{\partial \tilde{a}_2(\Omega, z)}{\partial z} &= -i \frac{\Omega^2 \chi_2^{(2)}}{4c^2 k_2(\Omega)} \int \tilde{a}_p(\Omega', z) \tilde{a}_p(\Omega - \Omega', z) e^{i[-k_p(\Omega') - k_p(\Omega - \Omega') + k_2(\Omega)]z} d\Omega' \\
&\quad - \frac{i}{2k_2(\Omega)} \frac{\partial^2}{\partial z^2} \tilde{a}_2(\Omega, z) \\
\frac{\partial \tilde{a}_3(\Omega, z)}{\partial z} &= -i \frac{\Omega^2 \chi_3^{(2)}}{4c^2 k_3(\Omega)} \int \tilde{a}_s(\Omega', z) \tilde{a}_p(\Omega - \Omega', z) e^{i[-k_s(\Omega') - k_p(\Omega - \Omega') + k_3(\Omega)]z} d\Omega' \\
&\quad - \frac{i}{2k_3(\Omega)} \frac{\partial^2}{\partial z^2} \tilde{a}_3(\Omega, z)
\end{aligned} \tag{A.5}$$

It has been shown [5, 4] that the second derivative is generally negligible, consistent with the slowly varying envelope approximation, even down to a few optical cycles. As stated earlier, computational considerations impose that the various spectral envelopes should be centered at the origin of the frequency axis, which is achieved by defining the shifted functions

$$\begin{aligned}
\tilde{\mathcal{E}}_{p,s,i,2,3}(\Delta\Omega, z) &= \tilde{a}_{p,s,i,2,3}(\Omega, z) \\
k_{p,s,i,2,3}(\Delta\Omega) &= k_{p,s,i,2,3}(\Omega) \\
\Delta\Omega_{p,s,i,2,3} &= \Omega - \omega_{p,s,i,2,3}.
\end{aligned} \tag{A.6}$$

In the set of equations (A.5), the envelopes $\tilde{a}_{p,s,i,2,3}$ are continuous functions of the frequency Ω , that covers the spectrum from infrared to ultraviolet. In numerical computation, these functions will be represented by discrete arrays, at discrete frequencies limited to a range of a few inverse pulse durations. The size and steps of the arrays are not arbitrary, but matched to the energy conservation.

Since we are not dealing here with pulses of a few optical cycle duration, we can make the approximation that the variation of the scaling factor in front of the integral is negligible in this frequency range, and we can assume it to be constant for each wave equation.

$$\begin{aligned}
\frac{\partial \tilde{\mathcal{E}}_p(\Delta\Omega)}{\partial z} &= -i \frac{\omega_p^2}{4c^2 k_p^{(0)}} \\
&\quad \left\{ \chi_1^{(2)} \int \tilde{\mathcal{E}}_s(\Delta\Omega', z) \tilde{\mathcal{E}}_i(\Delta\Omega - \Delta\Omega', z) e^{i[k_p(\Delta\Omega) - k_s(\Delta\Omega') - k_i(\Delta\Omega - \Delta\Omega')]z} d\Delta\Omega' \right. \\
&\quad + \chi_3^{(2)} \int \tilde{\mathcal{E}}_s^*(\Delta\Omega', z) \tilde{\mathcal{E}}_3(\Delta\Omega + \Delta\Omega', z) e^{i[k_s(\Delta\Omega') - k_3(\Delta\Omega + \Delta\Omega') + k_p(\Delta\Omega)]z} d\Delta\Omega' \\
&\quad \left. + \chi_2^{(2)} \int \tilde{\mathcal{E}}_p^*(\Delta\Omega', z) \tilde{\mathcal{E}}_2(\Delta\Omega + \Delta\Omega', z) e^{i[k_p(\Delta\Omega') - k_2(\Delta\Omega + \Delta\Omega') + k_p(\Delta\Omega)]z} d\Delta\Omega' \right\} \\
\frac{\partial \tilde{\mathcal{E}}_s(\Delta\Omega)}{\partial z} &= -i \frac{\omega_s^2}{4c^2 k_s^{(0)}} \\
&\quad \left\{ \chi_1^{(2)} \int \tilde{\mathcal{E}}_i^*(\Delta\Omega', z) \tilde{\mathcal{E}}_p(\Delta\Omega + \Delta\Omega', z) e^{i[k_i(\Delta\Omega') - k_p(\Delta\Omega + \Delta\Omega') + k_s(\Delta\Omega)]z} d\Delta\Omega' \right.
\end{aligned}$$

Appendix A. Coupled NL wave equations derived in frequency domain

$$\begin{aligned}
& + \chi_3^{(2)} \int \tilde{\mathcal{E}}_p^*(\Delta\Omega', z) \tilde{\mathcal{E}}_3(\Delta\Omega + \Delta\Omega', z) e^{i[k_p(\Delta\Omega') - k_3(\Delta\Omega + \Delta\Omega') + k_s(\Delta\Omega)]z} d\Delta\Omega' \Big\} \\
& \frac{\partial \tilde{\mathcal{E}}_i(\Delta\Omega)}{\partial z} = -i \frac{\omega_i^2 \chi_1^{(2)}}{4c^2 k_i^{(0)}} \\
& \int \tilde{\mathcal{E}}_s^*(\Delta\Omega', z) \tilde{\mathcal{E}}_p(\Delta\Omega + \Delta\Omega', z) e^{i[k_s(\Delta\Omega') - k_p(\Delta\Omega + \Delta\Omega') + k_i(\Delta\Omega)]z} d\Delta\Omega' \\
& \frac{\partial \tilde{\mathcal{E}}_2(\Delta\Omega)}{\partial z} = -i \frac{\omega_2^2 \chi_2^{(2)}}{4c^2 k_2^{(0)}} \\
& \int \tilde{\mathcal{E}}_p(\Delta\Omega', z) \tilde{\mathcal{E}}_p(\Delta\Omega - \Delta\Omega', z) e^{i[-k_p(\Delta\Omega') - k_p(\Delta\Omega - \Delta\Omega') + k_2(\Delta\Omega)]z} d\Delta\Omega' \\
& \frac{\partial \tilde{\mathcal{E}}_3(\Delta\Omega)}{\partial z} = -i \frac{\omega_3^2 \chi_3^{(2)}}{4c^2 k_3^{(0)}} \\
& \int \tilde{\mathcal{E}}_s(\Delta\Omega', z) \tilde{\mathcal{E}}_p(\Delta\Omega - \Delta\Omega', z) e^{i[-k_s(\Delta\Omega') - k_p(\Delta\Omega - \Delta\Omega') + k_3(\Delta\Omega)]z} d\Delta\Omega'
\end{aligned} \tag{A.7}$$

The Taylor expansion of a pump k-vector can be written as:

$$k_p(\Delta\Omega) = k_p^{(0)} + k_p^{(1)} \cdot (\Delta\Omega) + \frac{1}{2} k_p^{(2)} \cdot (\Delta\Omega)^2 + \dots \tag{A.8}$$

where $k_p^1 = \frac{1}{v_p}$ is responsible for group delay. The phase matching condition is chosen to apply at the central pulse frequencies $\Delta\Omega_{p,i,s} = 0$:

$$\delta k^{(0)} = k_s^{(0)} + k_i^{(0)} - k_p^{(0)} = 0. \tag{A.9}$$

In order to choose a retarded frame of reference moving with the group velocity of the pump pulse, it is essential in the simulation to subtract the same $k_p^1 \cdot (\Delta\Omega)$ term from all five k-vectors. To condense the notation and also for simulation consideration, the following quantities are introduced:

$$\begin{aligned}
A_p &= \frac{-i\omega_p^2}{4c^2 k_p^{(0)}} = \frac{-i\pi}{2n_p \lambda_p} \\
A_s &= \frac{-i\omega_s^2}{4c^2 k_s^{(0)}} = \frac{-i\pi}{2n_s \lambda_s} \\
A_i &= \frac{-i\omega_i^2}{4c^2 k_i^{(0)}} = \frac{-i\pi}{2n_i \lambda_i} \\
A_2 &= \frac{-i\omega_2^2}{4c^2 k_2^{(0)}} = \frac{-i\pi}{2n_2 \lambda_2}
\end{aligned}$$

$$A_3 = \frac{-i\omega_3^2}{4c^2k_3^{(0)}} = \frac{-i\pi}{2n_3\lambda_3} \quad (\text{A.10})$$

$$\begin{aligned} \mathcal{P}(\Delta\Omega) &= \tilde{\mathcal{E}}_p(\Delta\Omega)e^{-ik_p(\Delta\Omega)z} \\ \mathcal{S}(\Delta\Omega) &= \tilde{\mathcal{E}}_s(\Delta\Omega)e^{-ik_s(\Delta\Omega)z} \\ \mathcal{I}(\Delta\Omega) &= \tilde{\mathcal{E}}_i(\Delta\Omega)e^{-ik_i(\Delta\Omega)z} \\ \mathcal{N}(\Delta\Omega) &= \tilde{\mathcal{E}}_2(\Delta\Omega)e^{-ik_2(\Delta\Omega)z} \\ \mathcal{M}(\Delta\Omega) &= \tilde{\mathcal{E}}_3(\Delta\Omega)e^{-ik_3(\Delta\Omega)z} \end{aligned} \quad (\text{A.11})$$

to obtain the coupled equations for the field amplitudes:

$$\frac{\partial \tilde{\mathcal{E}}_p(\Delta\Omega)}{\partial z} = A_p \left[\chi_1^{(2)} (\mathcal{S} * \mathcal{I}) + \chi_2^{(2)} (\mathcal{P} * \mathcal{N}) + \chi_3^{(2)} (\mathcal{S} * \mathcal{M}) \right] (\Delta\Omega) e^{ik_p(\Delta\Omega)z} \quad (\text{A.12})$$

$$\frac{\partial \tilde{\mathcal{E}}_s(\Delta\Omega)}{\partial z} = A_s \left[\chi_1^{(2)} (\mathcal{I} * \mathcal{P}) + \chi_3^{(2)} (\mathcal{P} * \mathcal{M}) \right] (\Delta\Omega) e^{ik_s(\Delta\Omega)z} \quad (\text{A.13})$$

$$\frac{\partial \tilde{\mathcal{E}}_i(\Delta\Omega)}{\partial z} = A_i \chi_1^{(2)} (\mathcal{S} * \mathcal{P}) (\Delta\Omega) e^{ik_i(\Delta\Omega)z} \quad (\text{A.14})$$

$$\frac{\partial \tilde{\mathcal{E}}_2(\Delta\Omega)}{\partial z} = A_2 \chi_2^{(2)} (\mathcal{P} * \mathcal{P}) (\Delta\Omega) e^{ik_2(\Delta\Omega)z} \quad (\text{A.15})$$

$$\frac{\partial \tilde{\mathcal{E}}_3(\Delta\Omega)}{\partial z} = A_3 \chi_3^{(2)} (\mathcal{S} * \mathcal{P}) (\Delta\Omega) e^{ik_3(\Delta\Omega)z}. \quad (\text{A.16})$$

Here the operator $*$ describes a convolution and \star a cross-correlation. The refractive indexes $n(\Omega)$ are required to calculate $k_{p,s,i,2,3}(\Delta\Omega)$. They are obtained from a Sellmeier equation [36].

A.1 Quasi-Phase Matching

The susceptibility $\chi_n^{(2)}$ of the periodically poled lithium niobate (PPLN) alternates its sign at every poled domain "n".

$$\chi_n^{(2)} = \chi^{(2)}(-1)^n, \quad (\text{A.17})$$

where n is the index of poled domain. The expression can be decomposed into a Fourier series:

$$\chi^{(2)} = \frac{4\chi^{(2)}}{\pi} \sum_{m=1,3,5,\dots}^{\infty} \frac{1}{m} \sin\left(\frac{2\pi mz}{\Lambda}\right) = \frac{2\chi^{(2)}}{i\pi} \sum_{m=1,3,5,\dots}^{\infty} \frac{1}{m} \left[e^{imk_g z} - e^{-imk_g z} \right], \quad (\text{A.18})$$

Appendix A. Coupled NL wave equations derived in frequency domain

where Λ is the grating period of PPLN crystal, and $k_g = \frac{2\pi}{\Lambda}$. The grating period is chosen such that the poling is reversed before conversion back to the pump occurs, for the wavelength at the center of each pulse:

$$\frac{\delta k^{(0)} \Lambda}{2} = \pi \quad (\text{A.19})$$

Equation (A.7) for the DFG needs to be rewritten as a sum over $m = 1, 3, 5 \dots \infty$:

$$\begin{aligned} \frac{\partial \tilde{\mathcal{E}}_i(\Delta\Omega)}{\partial z} = & -\frac{\pi\omega_i^2\chi_1^{(2)}}{2c^2k_i^{(0)}} \sum_{m=1,3,5\dots}^{\infty} \frac{1}{m} \int \left\{ \tilde{\mathcal{E}}_s^*(\Delta\Omega', z) \tilde{\mathcal{E}}_p(\Delta\Omega + \Delta\Omega', z) \right. \\ & \left. \left\{ e^{i[k_s(\Delta\Omega') - k_p(\Delta\Omega + \Delta\Omega') + k_i(\Delta\Omega) + mk_g]z} \right. \right. \\ & \left. \left. - e^{i[k_s(\Delta\Omega') - k_p(\Delta\Omega + \Delta\Omega') + k_i(\Delta\Omega) - mk_g]z} \right\} d\Delta\Omega' \right\}. \end{aligned} \quad (\text{A.20})$$

A good approximation to a square wave is obtained by limiting the sum up to the value of $m = 5$. Dividing by Λ , we note that Eq. (A.19) is equivalent to the phase matching condition:

$$-k_s^{(0)} - k_i^{(0)} + k_p^{(0)} - k_g = 0 \quad (\text{A.21})$$

The numerical solution can be achieved by integrating the set of equations (6.7) to (6.11) over the thickness of the PPLN crystal, given a set of initial fields at $z = z_0$. However this solution does not include the effect of the linear dispersion of the crystal on the phase of each individual pulse, because the transformation (A.3) has removed the effect of dispersion. This transformation has to be reversed in order to get the complete electric field. For propagation from z_0 to z_1 the reverse transformation is a multiplication by $\exp[ik_j(\Delta\Omega)(z_1 - z_0)]$ where j takes the value $p, s, i, 2$ or 3 .

$$\tilde{E}_p(\Delta\Omega, z_1) = \tilde{\mathcal{E}}_p(\Delta\Omega, z_0) e^{ik_p(\Delta\Omega)(z_1 - z_0)} \quad (\text{A.22})$$

$$\tilde{E}_s(\Delta\Omega, z_1) = \tilde{\mathcal{E}}_s(\Delta\Omega, z_0) e^{ik_s(\Delta\Omega)(z_1 - z_0)} \quad (\text{A.23})$$

$$\tilde{E}_i(\Delta\Omega, z_1) = \tilde{\mathcal{E}}_i(\Delta\Omega, z_0) e^{ik_i(\Delta\Omega)(z_1 - z_0)} \quad (\text{A.24})$$

$$\tilde{E}_2(\Delta\Omega, z_1) = \tilde{\mathcal{E}}_2(\Delta\Omega, z_0) e^{ik_2(\Delta\Omega)(z_1 - z_0)} \quad (\text{A.25})$$

$$\tilde{E}_3(\Delta\Omega, z_1) = \tilde{\mathcal{E}}_3(\Delta\Omega, z_0) e^{ik_3(\Delta\Omega)(z_1 - z_0)}. \quad (\text{A.26})$$

By performing the inverse Fourier transform on equations (A.22) through (A.26) we obtain the expression of final fields in time domain.

Appendix B

Matlab code for coupled Ti:Sapphire and OPO cavities

```
%main program of coupled cavities
clear all
clear global
global Lts n2ts Lln n2lnp n2lns Ws alphagd alphaad Ag Aa Aln Energy0

%pump wavelength
lambdap = 770e-7; %cm
%T:S crystal
Lts = 0.3*2; %cm
n2ts = 10.5e-16; %cm^2/W
%ppln crystal
Lln = 0.1*2; %cm
n2lnp = 4e-15; %cm^2/W at 770nm
n2lns = 1.6e-15; %cm^2/W at 1140nm

%engery evolution parameters
sigma = 5e-20; %cm^2 5e-20 in the book
hbar = 1.054571726e-19; %J*fs
Ws = hbar*(2*pi*3e-5/lambdap)/(2*sigma)*1e-3;
%J/cm^2 %reduced due to approx. to 3 level system
alphagd = 0.3;%0.22;%0.20539; %gain parameter
alphaad = -0.1; %was -0.02 %absorber parameter
Ag = 2e-5; %beam area[cm^2] in the TS crystal
Aa = Ag/10; %beam area[cm^2] in SA ,was /1.5
Aln = 2e-5;%~pi*2.5e-5/4; %beam area[cm^2] in the ppln crystal
```

```

%simulation results interpolated
delta = zeros(4,3,6); %[delay(60-150), lambda(1130-1150),
chirp(0,0.5,1,1.5,2,3)]
delta(:,1,1)= [0.8289;-0.4330;-1.7416;-2.6466];
%delay from 60 to 150, lambda 1130, chirp 0
delta(:,1,2) = [3.8124;2.1204;-0.3232;-2.8856];
%delay from 60 to 150, lambda 1130, chirp 0.5
delta(:,1,3)= [5.3478;3.1087;-0.0024;-3.2765];
%delay from 60 to 150, lambda 1130, chirp 1
delta(:,1,4)=[5.1623;2.8420;-0.0742;-3.0690];
%delay from 60 to 150, lambda 1130, chirp 1.5
delta(:,1,5)= [4.2305;2.1574;-0.2504;-2.6746];
%delay from 60 to 150, lambda 1130, chirp 2
delta(:,1,6)= [2.4211;1.0050;-0.5060;-1.9973];
%delay from 60 to 150, lambda 1130, chirp 3

delta(:,2,1)= [0.3909;0.1842;-0.0938;-0.3522];
%delay from 60 to 150, lambda 1140, chirp 0
delta(:,2,2)=[3.7455;2.8535;1.1716;-0.9082];
%delay from 60 to 150, lambda 1140, chirp 0.5
delta(:,2,3)= [5.8223;4.3461;1.8387;-1.1730];
%delay from 60 to 150, lambda 1140, chirp 1
delta(:,2,4)=[6.1340;4.3894;1.8590;-1.0325];
%delay from 60 to 150, lambda 1140, chirp 1.5
delta(:,2,5)= [5.4666;3.7759;1.5916;-0.8085];
%delay from 60 to 150, lambda 1140, chirp 2
delta(:,2,6)= [3.7168;2.4568;1.0161;-0.4986];
%delay from 60 to 150, lambda 1140, chirp 3

delta(:,3,1)= [-0.1326;0.6888;1.4049;1.7935];
%delay from 60 to 150, lambda 1150, chirp 0
delta(:,3,2)=[3.3239;3.2115;2.3826;0.9655];
%delay from 60 to 150, lambda 1150, chirp 0.5
delta(:,3,3)= [5.7174;5.0559;3.3396;0.8672];
%delay from 60 to 150, lambda 1150, chirp 1
delta(:,3,4)=[6.4745;5.4179;3.4854;0.9674];
%delay from 60 to 150, lambda 1150, chirp 1.5
delta(:,3,5)= [6.1169;4.9314;3.1748;1.0396];
%delay from 60 to 150, lambda 1150, chirp 2
delta(:,3,6)= [4.6133;3.6195;2.3817;0.9873];
%delay from 60 to 150, lambda 1150, chirp 3
%in Hz
dmu = delta/7.5e-15/8191;

```

Appendix B. Matlab code for coupled Ti:Sapphire and OPO cavities

```
lambda = [1130,1140,1150];
for i = 1:6
    mlambda(:, :, i) = [lambda;lambda;lambda;lambda];
end
dlambda = -dmu.*(mlambda*1e-9).^2/0.3;
[X,Y,Z] = meshgrid(lambda,60:30:150,[0,0.5,1,1.5,2,3]);
[Xq,Yq,Zq]=meshgrid(1130:1150,60:150,0:0.1:3);
dlambdaq = interp3(X,Y,Z,dlambda,Xq,Yq,Zq);

%preparation of arrays for main loop
N = 2000; %# of round trips
Energy0 = zeros(1,N);
Energys = zeros(1,N);
phi2p = zeros(1,N);
phi2s = zeros(1,N);
taup = zeros(1,N);
taus = zeros(1,N);
invp = zeros(1,N);
dtp = zeros(1,N);
dts = zeros(1,N);
lambdas = zeros(1,N);
abarp = zeros(1,N);
abars = zeros(1,N);

%initial values
Energy0=2.75173410813534e-08; %steady state pump energy without OPO
Energy0(1)=Energy0*1;
Energys(1)=Energy0*0.1;
taup(1) = 85; %Gaussian pulse width in fs
taus(1) = 85; %Gaussian pulse width in fs
phi2p(1)= 0;
phi2s(1)= 0;
dtp(1)= 0;
dts(1)= 120;
lambdas(1) = 1140e-7;
chirp(1) = 2*pi/lambdap*n2ts*Energy0(1)/Ag/taup(1)*1e15/taup(1)^2*Lts;
abarp(1) = chirp(1)*taup(1)^2/4;

%noise
% m = 100;
% gamma = 0.5;%.*(1:N)*0.5e-3
% [noise, fn, vafn] = jitter(m,N,gamma); %noise with increasing amplitude
noise = zeros(1,N); %noise with constant amplitude
```

Appendix B. Matlab code for coupled Ti:Sapphire and OPO cavities

```

%main loop
for j = 2:N
    [Energyp(j),taup(j),phi2p(j),dtp(j),abarp(j)] = TiSaph4(Energyp(j-1)
        ,Energys(j-1),taup(j-1),taus(j-1),phi2p(j-1),abarp(j-1),dtp(j-1),
        dts(j-1),lambdap,lambdas(j-1));
    [Energys(j),taus(j),phi2s(j),dts(j),abars(j),lambdas(j)] = OP04(
        Energyp(j-1),Energys(j-1),taup(j-1),taus(j-1),phi2s(j-1),abars(j-1),
        dtp(j-1),dts(j-1),lambdap,lambdas(j-1),Lln,n2lns,Aln,dlambdaq,
        noise(j-1));
    if lambdas>1150e-7
        break
    elseif lambdas<1130e-7
        break
    end
end

%write the file
dlmwrite('GVDp-1600s1000LLp11%s7%OP0gain1_15-0_5xEsPumpdepletion0_91-0_1xEs
NLlosscoeff1_5e6Damping0Initiallambdas1140.txt',...
[Energyp*1e9;Energys*1e9;taup;taus;abarp;abars;lambdas*1e7;noise'],'
delimiter','\t','newline','pc','precision','%10.5f')

%make the plot
figure
subplot(4,1,1);
[AX1,H1,H2] = plotyy(1:N,Energyp*1e9,1:N,Energys*1e9,'plot');
set(get(AX1(1),'Ylabel'),'String','Pump energy(nJ)')
set(get(AX1(2),'Ylabel'),'String','Signal energy(nJ)')
ylim(AX1(1),[20 40])
set(AX1(2),'YColor','r')
set(H2,'Color','r')
ylim(AX1(2),[2 4])
title('GVD=-1600fs^2(p)&1000fs^2(s); Linear loss=11%(p)&7%(s);
OPO gain=1.15-0.5xEs; Pump depletion=0.91-0.1xEs;
NL loss coeff=1.5e6; damping coeff=0; Initial lambdas=1140nm')
subplot(4,1,2);
[AX2,H3,H4] = plotyy(1:N,taup*1.17,1:N,taus*1.17,'plot');
set(get(AX2(1),'Ylabel'),'String','Pump pulse width(fs)')
set(get(AX2(2),'Ylabel'),'String','Signal pulse width(fs)')
ylim(AX2(1),[0 200])
set(AX2(2),'YColor','r')
set(H4,'Color','r')

```

Appendix B. Matlab code for coupled Ti:Sapphire and OPO cavities

```

ylim(AX2(2),[0 200])
subplot(4,1,3);
[AX3,H5,H6] = plotyy(1:N,abarp,1:N,abars,'plot');
set(get(AX3(1),'Ylabel'),'String','Pump chirp coeff.')
```

```

set(get(AX3(2),'Ylabel'),'String','Signal chirp coeff.')
```

```

ylim(AX3(1),[-0.6 0])
set(AX3(2),'YColor','r')
set(H6,'Color','r')
ylim(AX3(2),[0 0.6])
subplot(4,1,4);
[AX4,H7,H8] = plotyy(1:N,dts-dtp,1:N,lambda_s*1e7,'plot');
set(get(AX4(1),'Ylabel'),'String','delay btw signal and pump')
```

```

set(get(AX4(2),'Ylabel'),'String','Signal wavelength(nm)')
```

```

ylim(AX4(1),[110 150])
set(AX4(2),'YColor','r')
set(H8,'Color','r')
ylim(AX4(2),[1135 1142])
xlabel('Roundtrip counts')
```

```

%simulate the pulse evolution of Ti:Sapphire cavity
function [Energyyp,taup,phi2p,dtp,abarp] = TiSaph4(
Energyyp,Energys,taup,taus,phi2p,abarp,dtp,dts,lambda_p,lambda_s)

global Lts n2ts Lln n2lnp Ws alphagd alphaad Ag Aa Aln Energy0

%Ti:Sapphire gain and loss
Energyyp = Ag*Ws*log(1-exp(alphagd)*(1-exp(Energyyp/(Ag*Ws)))); %gain
Energyyp = Aa*Ws*log(1-exp(alphaad)*(1-exp(Energyyp/(Aa*Ws)))); %loss
Energyyp = 0.89*Energyyp; %linear loss

%Dispersive process, energy is conserved
%chirp coeff. for the TS crystal
Ip = Energyyp/(Ag*taup*1e-15); %pump intensity
Is = Energys/(Aln*taus*1e-15); %signal intensity
chirp1 = (2*pi/lambda_p)*n2ts*Ip/taup^2*Lts; %= -4*abar/tau1^2; 1/fs^2
%chirp coeff. for the PPLN crystal
chirp2 = (2*pi/lambda_s)*n2lnp*(Ip*Ag/Aln+2*Is)/taup^2*Lln;
%material 1/fs^2
n2ps = ((1140+abarp*4)*1e-7-lambda_s)/39*1e-7+0.8*abarp*1e-15;
%effective n2ps
chirp2 = chirp2 + (2*pi/lambda_p)*n2ps*Is/taup^2*Lln;%cross PM
chirp = chirp1 + chirp2;
```


Appendix B. Matlab code for coupled Ti:Sapphire and OPO cavities

```

%Dispersion
GVD1 = -1600; %fs^2   -1700<=GVD1<=-1600 (800fs;0.91-0.1;1.15-0.5,0.93)
GVD = 2*GVD1;
%Matrices
Kerr = [1,0; chirp,1];
Disp1 = [1,GVD1; 0,1];
Disp = [1,GVD; 0,1];
epsil = 0;%0.01i; %damping coeff.
M = Disp*Kerr+epsil*eye(2); %cavity matrix
stbl = 0.5*(M(1,1)+M(2,2));
%disp(stbl)
invp = phi2p-2i/taup^2; %value from last roundtrip
abarp = -chirp*taup^2/4;
invp = (M(2,1)+M(2,2)*invp)/(M(1,1)+M(1,2)*invp); %current value
phi2p = real(invp); %output chirp
taup = sqrt(-2/imag(invp)); %output pulse duration

%OPO interaction
g = sqrt(2*log(2));%factor in front of Gaussian pulse duration
Energyp = (0.91-0.1*Energys/Energy0+6.1e-6*(taup*g-150)^2+7.8e-6*
(dts-dtp-120)^2+5.5e10*(lambdas-1140e-7)^2)*Energyp;

dtp = dtp+7.4 + 5.9e-4*(dts-dtp-102)^2-7.59116; %output pulse center

-----
%simulate the pulse evolution of Ti:Sapphire cavity
function [Energys,taus,phi2s,dts,abars,lambdas] = OP04(Energyp,Energys,
taup,taus,phi2s,abars,dtp,dts,lambdap,lambdas,
Lln,n2lns,Aln,dlambdaq,noise)
Energy0=2.75173410813534e-08; %steady state pump energy
%chirp coeff. for the PPLN crystal
Ip = Energyp/(Aln*taup*1e-15); %pump intensity
Is = Energys/(Aln*taus*1e-15); %signal intensity
chirp1 = (2*pi/lambdap)*(n2lns-4e-15)*(Is+2*Ip)/taus^2*Lln;
%material 1/fs^2
n2sp = ((1140-abars*4)*1e-7-lambdas)/108*1e-7-0.8*abars*1e-15;
%effective n2sp
chirp2 = (2*pi/lambdas)*n2sp*Ip/taus^2*Lln; %cross PM
chirp = chirp1 + chirp2;
%Dispersion
GVD1 = 1000; %fs^2   500<=GVD1<=1100(pump GVD 1600)
GVD = 2*GVD1;

```

Appendix B. Matlab code for coupled Ti:Sapphire and OPO cavities

```

%Matrices
Kerr = [1,0; chirp,1];
Disp1 = [1,GVD1; 0,1];
Disp = [1,GVD; 0,1];
epsil = 0;%-0.01i; %damping coeff.
M = Disp*Kerr+epsil*eye(2); %OPO cavity matrix

%Dispersive process, energy is conserved
invp = phi2s-2i/taus^2; %value from last roundtrip
abars = -chirp*taus^2/4;
invp = (M(2,1)+M(2,2)*invp)/(M(1,1)+M(1,2)*invp); %current value
phi2s = real(invp); %output chirp
taus = sqrt(-2/imag(invp)); %output pulse duration

%OPO interaction
g = sqrt(2*log(2));%factor in front of Gaussian pulse duration
beta = 1.5e6; % ~0.5% SHG loss
Energys = (1.15-0.5*Energys/Energy0-beta*Energys-6.2e-6*(taus*g-150)^2
-7.9e-6*(dts-dtp-120)^2-5.5e10*(lambdas-1140e-7)^2)*Energys;
Energys = 0.93*Energys; %linear loss, output energy

dts = dts+2.7 - 0.034*(dts-dtp)+1.38-(lambdas*1e7-1140)*0.3+noise;
%output pulse center(-0.5<=dx<=1.7)
delayn = round(dts - dtp)-59;
lambdan = round(lambdas*1e7)-1129;
chirpn = round(abars*10)+1;
dlambdas = dlambdaq(delayn,lambdan,chirpn);
lambdas = lambdas+dlambdas*1e-7; %output signal wavelength

-----
%The generation and analysis of random noise on round trip time
function [noise2,f2,vaf2] = jitter(m,N,gamma)
dt = 10e-9; %10 ns round trip time
df = 1/dt; %1e8
T = ([1:N]-N/2)*dt; %ns
noise = randn(1,N)*gamma;
%noise with random # generated every m round trips
noise2 = interp(noise,m);
%crop the interpolated noise to the same scale N
noise2 = noise2(N*m/2-N/2+1:N*m/2+N/2);
figure
plot(noise2)

```

Appendix B. Matlab code for coupled Ti:Sapphire and OPO cavities

```
hold on
plot(noise2.^2,'r--')
NFFT = 2^nextpow2(N); % Next power of 2 from N
%fourier transform of noise
fn = fft(noise,NFFT)/N;
fx = df/2*linspace(0,1,NFFT/2+1);%x axis of the fourier transform
fn2 = fft(noise2,NFFT)/N;
% figure
% plot(fx,2*abs(fn(1:NFFT/2+1)))
% hold on
% plot(fx,2*abs(fn2(1:NFFT/2+1)),'r')
%calculate the frequency and variance of noise.
f = sum(fx.*abs(fn(1:NFFT/2+1)).^2)/sum(abs(fn(1:NFFT/2+1)).^2);
vaf = sum(fx.^2.*abs(fn(1:NFFT/2+1)).^2)/sum(abs(fn(1:NFFT/2+1)).^2)-f^2;
f2 = sum(fx.*abs(fn2(1:NFFT/2+1)).^2)/sum(abs(fn2(1:NFFT/2+1)).^2);
vaf2 = sum(fx.^2.*abs(fn2(1:NFFT/2+1)).^2)/sum(abs(fn2(1:NFFT/2+1)).^2)
-f2^2;
```

Appendix C

Fortran code for second order nonlinear interactions

```
program simppln
! for DFG only
! no phase front tilt and collinear phasematching

implicit NONE
integer N

parameter(N=8191)          !has to be (2^n)-1

!-- declare common blocks -----

common/art1m/tpre1m,tpim1m,tsre1m,tsim1m,tire1m,tiim1m,t2re1m,t2im1m,
t3re1m,t3im1m
common/art0/tpre0,tpim0,tsre0,tsim0,tire0,tiim0,t2re0,t2im0,t3re0,t3im0
common/arhp/tprehp,tpimhp,tsrehp,tsimhp,tirehp,tiimhp,t2rehp,t2imhp,
t3rehp,t3imhp
common/art1p/tpre1p,tpim1p,tsre1p,tsim1p,tire1p,tiim1p,t2re1p,t2im1p,
t3re1p,t3im1p
common/arD1m/Dpre1m,Dpim1m,Dsre1m,Dsim1m,Dire1m,Diim1m,D2re1m,D2im1m,
D3re1m,D3im1m,Dsire1m,Dsiim1m,Dp2re1m,Dp2im1m,Ds3re1m,Ds3im1m,Dpire1m,
Dpiim1m,Dp3re1m,Dp3im1m
common/arD0/Dpre0,Dpim0,Dsre0,Dsim0,Dire0,Diim0,D2re0,D2im0,D3re0,D3im0,
Dsire0,Dsiim0,Dp2re0,Dp2im0,Ds3re0,Ds3im0,Dpire0,Dpiim0,Dp3re0,Dp3im0
common/arDhp/Dprehp,Dpimhp,Dsrehp,Dsimhp,Direhp,Diimhp,D2rehp,D2imhp,
D3rehp,D3imhp,Dsirehp,Dsiimhp,Dp2rehp,Dp2imhp,Ds3rehp,Ds3imhp,Dpirehp,
Dpiimhp,Dp3rehp,Dp3imhp
```

Appendix C. Fortran code for second order nonlinear interactions

```

common/arD1p/Dpre1p,Dpim1p,Dsre1p,Dsim1p,Dire1p,Diim1p,D2re1p,D2im1p,
      D3re1p,D3im1p,Dsire1p,Dsiim1p,Dp2re1p,Dp2im1p,Ds3re1p,Ds3im1p,Dpire1p,
      Dpiim1p,Dp3re1p,Dp3im1p

common/arkv/kvp,kvs,kvi,kv2,kv3,kvg      !kvg is grating period of ppln
common/vdbl/det,dom,facp,facs,faci,fac2,fac3
common/vinp/taup,taus,dp,ds,ampp,amps,np,ns,ni,n2,n3
      !delay and amplitude for p and s
common/vcon/minval,tpa,tsa,tia,t2a,t3a
common/vrea/dist
common/angles/woff,tilt,intang
common/kvecs/k0p,k1p,k0s,k1s,k0i,k1i,k02,k12,k03,k13

!-- declare variables -----

character*8      fnamep,fnames,fname2,fnamec,fnamei,fname3
character*4      conname
character*1      chue,czee,ceie,czed,ceid
integer          dhue,dzee,deie,dzed,deid,tlen,tlde
integer          i,j,k,l,sign,mult
integer          loopnum,loopmax,oloop,iloop,omax,imax
double precision st,dist,zmax,deltaz
double precision taup,taus,dp,ds,ampp,amps,np,ns,ni,n2,n3,kvg
double precision det,dom,facp,facs,faci,fac2,fac3

double precision tpre1m(N),tpim1m(N),Dpre1m(N),Dpim1m(N),Dsire1m(N),
      Dsiim1m(N),Dp2re1m(N),Dp2im1m(N),Ds3re1m(N),Ds3im1m(N)
double precision tsre1m(N),tsim1m(N),Dsre1m(N),Dsim1m(N),Dpire1m(N),
      Dpiim1m(N),Dp3re1m(N),Dp3im1m(N)
double precision tire1m(N),tiim1m(N),Dire1m(N),Diim1m(N)
double precision t2re1m(N),t2im1m(N),D2re1m(N),D2im1m(N)
double precision t3re1m(N),t3im1m(N),D3re1m(N),D3im1m(N)
double precision tpre0(N),tpim0(N),Dpre0(N),Dpim0(N),Dsire0(N),
      Dsiim0(N),Dp2re0(N),Dp2im0(N),Ds3re0(N),Ds3im0(N)
double precision tsre0(N),tsim0(N),Dsre0(N),Dsim0(N),Dpire0(N),
      Dpiim0(N),Dp3re0(N),Dp3im0(N)
double precision tire0(N),tiim0(N),Dire0(N),Diim0(N)
double precision t2re0(N),t2im0(N),D2re0(N),D2im0(N)
double precision t3re0(N),t3im0(N),D3re0(N),D3im0(N)
double precision tprehp(N),tpimhp(N),Dprehp(N),Dpimhp(N),Dsirehp(N),
      Dsiimhp(N),Dp2rehp(N),Dp2imhp(N),Ds3rehp(N),Ds3imhp(N)
double precision tsrehp(N),tsimhp(N),Dsrehp(N),Dsimhp(N),Dpirehp(N),
      Dpiimhp(N),Dp3rehp(N),Dp3imhp(N)

```

Appendix C. Fortran code for second order nonlinear interactions

```

double precision tirehp(N),tiimhp(N),Direhp(N),Diimhp(N)
double precision t2rehp(N),t2imhp(N),D2rehp(N),D2imhp(N)
double precision t3rehp(N),t3imhp(N),D3rehp(N),D3imhp(N)
double precision tpre1p(N),tpim1p(N),Dpre1p(N),Dpim1p(N),Dsire1p(N),
      Dsiim1p(N),Dp2re1p(N),Dp2im1p(N),Ds3re1p(N),Ds3im1p(N)
double precision tsre1p(N),tsim1p(N),Dsre1p(N),Dsim1p(N),Dpire1p(N),
      Dpiim1p(N),Dp3re1p(N),Dp3im1p(N)
double precision tire1p(N),tiim1p(N),Dire1p(N),Diim1p(N)
double precision t2re1p(N),t2im1p(N),D2re1p(N),D2im1p(N)
double precision t3re1p(N),t3im1p(N),D3re1p(N),D3im1p(N)

double precision minval,tpa,tsa,tia,t2a,t3a,width,fsqr,oldwidth
double precision arge,argt,tmpre,tmpim,newwidth
double precision kvp(N),kvs(N),kvi(N),kv2(N),kv3(N)
double precision agp,ags,agi,ag2,ag3
complex*16      cwp(N),cws(N),cwi(N),cw2(N),cw3(N)
complex*16      hlpp(N),hlps(N),hlpi(N),hlp2(N),hlp3(N)
double precision pi,tpi,ftor,pf,bdiam
double precision error,delta,deltaold,tmp
double precision chi,lambda,waist,energp,energs,energi,energ2,energ3
double precision fldp,flds,fldi,fld2,fld3,totfld,woff,tilt,intang
double precision k0p,k1p,k0s,k1s,k0i,k1i,k02,k12,k03,k13

double precision p1m,p0,c1m,c0,Dp1m,Dp0,Dphp,Dhp1m
double precision Dhp0,Dc1m,Dc0,Dchp,Dc1p

!-----
! define input parameters
!-----

!-- define constants -----
pi   = 3.14159265359d+0    ! equals PI
tpi  = 6.28318530718d+0    ! equals 2 PI
ftor = 0.707106781188d+0    ! equals 1/sqrt(2)
pf   = .00132720936472d+0
      ! equals epsilon*c/2 units are A^2 * s^3 / ( kg * m^2 )

!-- define pulse parameters -----
taup  = 111.8d+0    ! FWHM of e gaussian [fs]
taus  = 111.8d+0
lambda = 770d+0    ! wavelength [nm]

```

Appendix C. Fortran code for second order nonlinear interactions

```

bdiam = .05d+0    ! in [cm]
tilt   = dble( 0 )    ! pulsefront tilt in degrees !was 40
intang = dble( 0 )    ! int. half angle betw. funds for noncoll.

!-- define crystal parameters -----
!(SNLO v52)
np     = 2.170464d+0 ! refractive index along e-axis for pump 770nm
ns     = 2.144053d+0 ! for signal 1140nm
ni     = 2.108331d+0 ! for idler 2372.432432nm
n2     = 2.340316d+0 ! for SH of pump 385nm
n3     = 2.262916d+0 ! for SF of pump and signal 459.5811518nm
! kvg = tpi/22d+0 ! k vector of grating period 22um [1/um]
kvg    = tpi/20.23473d+0 ! phase-matching period 20.23473um [1/um]
chi    = 2.0d+0*15.3d+0 ! chi^(2) in 10^-12 [m/V] chi = 2d

zmax   = 1d+3      ! total crystal length [um]
woff   = dble( 0 ) ! walkoff angle

!-- numerics parameters -----

det     = 7.5d+0 ! spacing betw. two succ. pts in fs, was 0.75
st      = 1d-1   ! 1 calc. is performed every stepsize um
deltaz  = 1d+2   ! 5 output is written every deltaz um

error   = 1d-3 ! allowed diff. between pred. and corr.
minval  = 1d-10 ! variable < minval --> val=0
loopmax = 5     ! corrector gives up after loopmax iter.

k0p     = 177109.448241068d+0 ! k0 in 1/cm
k1p     = 75.7d+0             ! k'=1/v_g in ps/cm
k0s     = 118170.897433459d+0 ! k0 in 1/cm
k1s     = 73.37d+0           ! k'=1/v_g in ps/cm
k0i     = 55838.1413128332d+0 ! k0 in 1/cm
k1i     = 72.5218377575062d+0 ! k'=1/v_g in ps/cm
k02     = 381938.678061229d+0 ! k0 in 1/cm
k12     = 97.4584223863297d+0 ! k'=1/v_g in ps/cm
k03     = 309362.936522663d+0 ! k0 in 1/cm
k13     = 86.3660819646103d+0 ! k'=1/v_g in ps/cm
kvg     = kvg*1d+4           ! 1/um -> 1/cm

!-- calculate parameters -----

zmax    = zmax*1d-4          ! um -> cm

```

Appendix C. Fortran code for second order nonlinear interactions

```

st      = st*1d-4           ! um -> cm
deltaz= deltaz*1d-4       ! um -> cm
det     = det*1d-3        ! fs -> ps

taup   = taup*1d-3       ! fs -> ps
taus   = taus*1d-3

dom     = tpi/(N*det)      ! delta omega
waist  = pi*(bdiam**2)/4  ! beam waist [cm^2]

p1m    = 23d+0/5d+0
p0     = 28d+0/5d+0
c1m    = 1d+0/31d+0
c0     = 32d+0/31d+0
Dp1m   = 26d+0/15d+0*st*1d-2      ! *.01 bec cm -> m (st in cm)
Dp0    = 4d+0*st*1d-2
Dphp   = 32d+0/15d+0*st*1d-1
Dhp1m  = 3d+0/8d+0*st*1d-2
Dhp0   = 9d+0/8d+0*st*1d-2
Dc1m   = 1d+0/93d+0*st*1d-2
Dc0    = 12d+0/93d+0*st*1d-2
Dchp   = 64d+0/93d+0*st*1d-2
Dc1p   = 15d+0/93d+0*st*1d-2

omax   = int(zmax/deltaz)        ! parameter for outer loop
imax   = int(deltaz/st)         ! parameter for inner loop

! prefactors in front of integrals
facp= (chi)/(1d+3*lambda*np)
facs= (chi)/(1d+3*lambda/770d+0*1140d+0*ns)
faci= (chi)/(1d+3*lambda/770d+0*2372.432432d+0*ni)
fac2= (chi)/(0.5d+3*lambda*n2)   ! prefactor for convolution
fac3= (chi)/(1d+3*lambda/770d+0*459.5811518d+0*n3)

!-- determine output files -----

energp=0.6d-3      ! energp=3e3 mJ for cw and 3e-3 for pulse PM,
                  !10x larger for unPM
energs=0.4d-3     ! energs=3e1 mJ for cw and 3e-5 for pulse PM,
                  !10x larger for unPM
print *,'energp=',energp,' mJ','energs=',energs,' mJ'

dhue=int(tlen/100)

```


Appendix C. Fortran code for second order nonlinear interactions

```
dzee=int((tlen-(dhue*100))/10)
deie=int(tlen-(dhue*100)-(dzee*10))
chue=char(dhue+48)
cze=char(dzee+48)
ceie=char(deie+48)

open(unit=8,file="inputdata.dat",status='new')
write(8,664) ns, chi/(fac*1d+3*ns)

664 format(2x,' ns =',f8.4,' ',3x,' lambdas =',f8.2,' nm')

write(8,665) energp,energs

665 format(2x,' energp =',f8.4,' mJ',3x,' energs =',f8.4,' mJ')

write(8,666) taup,taus

666 format(2x,' taup =',f8.4,' ps',3x,' taus =',f8.4,' ps')

!pulse centers
      dp=0d+0 !-dble(tlde)/2000.    !ps
      ds=120d-3 !+dble(tlde)/2000.  !ps
      dzed=int(tlde/10)
      deid=int(tlde-(dzed*10))
      czed=char(dzed+48)
      ceid=char(deid+48)

      write(8,667) dp,ds
667 format(2x,' delay dp',f8.4,' ps',2x,' delay ds',f8.4,' ps')

      ampp = 86802109.8203d+0*dsqrt(energp/(np*waist*taup)) !~1.3e7
      amps  =86802109.8203d+0*dsqrt(energs/(ns*waist*taus)) !~1.3e5
      print *, 'writing into ', 'fnamep,', 'fnames,', 'fname2', ampp, amps
      ! units for ampp/e are [V/m] if energp/e is in [mJ], waist in
      ! [cm^2] and tauo/e in [ps]
      open(unit=1,file="pump.dat",status='new')
      open(unit=2,file="signal.dat",status='new')
      open(unit=3,file="idler.dat",status='new')
      open(unit=4,file="2nd.dat",status='new')
      open(unit=5,file="sum.dat",status='new')
      open(unit=6,file="kvs.dat",status='new')
      open(unit=9,file="energy.dat",status='new')
```

Appendix C. Fortran code for second order nonlinear interactions

```
!-----  
! start the simulation  
!-----  
  
!-- initialize -----  
  
        call selma          ! get k-vectors from the Sellmeier equation  
  
!-- clear arrays -----  
  
        do 10 i=1,N,1  
        Dpre1m(i)=dble(0)    ! D means derivative with respect to z  
        Dpre0(i) =dble(0)    ! pre means pump pulse real part  
        Dprehp(i)=dble(0)    ! 1m means at step (n-1)  
        Dpre1p(i)=dble(0)    ! 0 : (n), hp : (n+0.5), 1p : (n+1)  
  
        Dpim1m(i)=dble(0)  
        Dpim0(i) =dble(0)  
        Dpimhp(i)=dble(0)  
        Dpim1p(i)=dble(0)  
  
        Dsre1m(i)=dble(0)    ! sre means signal pulse real part  
        Dsre0(i) =dble(0)  
        Dsrehp(i)=dble(0)  
        Dsre1p(i)=dble(0)  
  
        Dsim1m(i)=dble(0)  
        Dsim0(i) =dble(0)  
        Dsimhp(i)=dble(0)  
        Dsim1p(i)=dble(0)  
  
        Dire1m(i)=dble(0)    ! ire means idler pulse real part  
        Dire0(i) =dble(0)  
        Direhp(i)=dble(0)  
        Dire1p(i)=dble(0)  
  
        Diim1m(i)=dble(0)  
        Diim0(i) =dble(0)  
        Diimhp(i)=dble(0)  
        Diim1p(i)=dble(0)  
  
        D2re1m(i)=dble(0)    ! 2re means SHG of pump real part  
        D2re0(i) =dble(0)
```

Appendix C. Fortran code for second order nonlinear interactions

```
D2rehp(i)=dble(0)
D2re1p(i)=dble(0)

D2im1m(i)=dble(0)
D2im0(i) =dble(0)
D2imhp(i)=dble(0)
D2im1p(i)=dble(0)

D3re1m(i)=dble(0)      ! 3re means SFG of pump and signal real part
D3re0(i) =dble(0)
D3rehp(i)=dble(0)
D3re1p(i)=dble(0)

D3im1m(i)=dble(0)
D3im0(i) =dble(0)
D3imhp(i)=dble(0)
D3im1p(i)=dble(0)

Dsire1m(i)=dble(0)     ! sire means signal+idler pulse real part
Dsire0(i) =dble(0)
Dsirehp(i)=dble(0)
Dsire1p(i)=dble(0)

Dsiim1m(i)=dble(0)
Dsiim0(i) =dble(0)
Dsiimhp(i)=dble(0)
Dsiim1p(i)=dble(0)

Dpire1m(i)=dble(0)     !  pire means pump-idler pulse real part
Dpire0(i) =dble(0)
Dpirehp(i)=dble(0)
Dpire1p(i)=dble(0)

Dpiim1m(i)=dble(0)
Dpiim0(i) =dble(0)
Dpiimhp(i)=dble(0)
Dpiim1p(i)=dble(0)

Dp2re1m(i)=dble(0)     ! p2re means SHG-pump pulse real part
Dp2re0(i) =dble(0)
Dp2rehp(i)=dble(0)
Dp2re1p(i)=dble(0)
```

Appendix C. Fortran code for second order nonlinear interactions

```
Dp2im1m(i)=dble(0)
Dp2im0(i) =dble(0)
Dp2imhp(i)=dble(0)
Dp2im1p(i)=dble(0)

Dp3re1m(i)=dble(0)      ! p3re means sum-pump pulse real part
Dp3re0(i) =dble(0)
Dp3rehp(i)=dble(0)
Dp3re1p(i)=dble(0)

Dp3im1m(i)=dble(0)
Dp3im0(i) =dble(0)
Dp3imhp(i)=dble(0)
Dp3im1p(i)=dble(0)

Ds3re1m(i)=dble(0)      ! s3re means sum-signal pulse real part
Ds3re0(i) =dble(0)
Ds3rehp(i)=dble(0)
Ds3re1p(i)=dble(0)

Ds3im1m(i)=dble(0)
Ds3im0(i) =dble(0)
Ds3imhp(i)=dble(0)
Ds3im1p(i)=dble(0)

10      continue

!-- initial conditions -----

      call initial !get the temporal profile of the two input functions

      do 11 i=1,N,1
      hlpp(i)=cplx(tpre0(i), tpim0(i))
        hlps(i)=cplx(tsre0(i), tsim0(i))
        hlpi(i)=cplx(tire0(i), tiim0(i))
        hlp2(i)=cplx(t2re0(i), t2im0(i))
        hlp3(i)=cplx(t3re0(i), t3im0(i))

      call fft(hlpp,N,-1)
      call fft(hlps,N,-1)
      call fft(hlpi,N,-1)
      call fft(hlp2,N,-1)
      call fft(hlp3,N,-1)
```

Appendix C. Fortran code for second order nonlinear interactions

```

do 12 i=1,N,1
  k=((1-N)/2)-1+i
      !intensities of p,s,i,2,3
write(1,*)0.,pf*dreal(hlpp(i)), pf*dimag(hlpp(i))/dble(1e13)
  write(2,*)0.,pf*dreal(hlps(i)), pf*dimag(hlps(i))/dble(1e13)
  write(3,*)0.,pf*dreal(hlpi(i)), pf*dimag(hlpi(i))/dble(1e13)
  write(4,*)0.,pf*dreal(hlp2(i)), pf*dimag(hlp2(i))/dble(1e13)
  write(5,*)0.,pf*dreal(hlp3(i)), pf*dimag(hlp3(i))/dble(1e13)

12  continue
  write(1,*)
  write(1,*)
  write(2,*)
  write(2,*)
  write(3,*)
  write(3,*)
  write(4,*)
  write(4,*)
  write(5,*)
  write(5,*)

  call cor0ps          !p - s returns Dpsre0() and Dpsim0()
  call cor0pi          ! returns Dpire0() and Dpiim0()
  call cor0si          ! returns Dsire0() and Dsiim0()

  do 14 i=1,N,1
  Dpre0(i) = Dsire0(i)+Dp2re0(i)+Ds3re0(i)
  Dpim0(i) = Dsiim0(i)+Dp2im0(i)+Ds3im0(i)
  Dsre0(i) = Dpire0(i)+Dp3re0(i)
  Dsim0(i) = Dpiim0(i)+Dp3im0(i)
14  continue

!- calculating fcts @ n-1 -----
! assume all fcts are known @ n. Fcts @ n-1 are calculated using a
! linear approximation

  do 13 i=1,N,1
  tpre1m(i) = tpre0(i) - Dpre0(i)*st
  tpim1m(i) = tpim0(i) - Dpim0(i)*st

  tsre1m(i) = tsre0(i) - Dsre0(i)*st

```

Appendix C. Fortran code for second order nonlinear interactions

```
tsim1m(i) = tsim0(i) - Dsim0(i)*st

tire1m(i) = tire0(i) - Dire0(i)*st
tiim1m(i) = tiim0(i) - Diim0(i)*st

t2re1m(i) = t2re0(i) - D2re0(i)*st
t2im1m(i) = t3im0(i) - D2im0(i)*st

t3re1m(i) = t3re0(i) - D3re0(i)*st
t3im1m(i) = t3im0(i) - D3im0(i)*st
13 continue

call cor1mps          ! returns Dpre1m() and Dpim1m()
call cor1mpi
call cor1msi

do 15 i=1,N,1
  Dpre1m(i) = Dsire1m(i)+Dp2re1m(i)+Ds3re1m(i)
  Dpim1m(i) = Dsiim1m(i)+Dp2im1m(i)+Ds3im1m(i)
  Dsre1m(i) = Dpire1m(i)+Dp3re1m(i)
  Dsim1m(i) = Dpiim1m(i)+Dp3im1m(i)
15 continue

!-- calculation -----

dist = 0d+0          ! set z to 0 cm

do 20 oloop=0,(omax-1),1 ! outer loop for calculation
  do 30 iloop=0,(imax-1),1 ! inner loop for output into files

    dist = dist+st      ! increment dist by st
    write(*,995)"z[cm]= ",dist

  call cor0ps          ! returns Dpre0() and Dpim0()
  call cor0pi
  call cor0si

do 16 i=1,N,1
  Dpre0(i) = Dsire0(i)+Dp2re0(i)+Ds3re0(i)
  Dpim0(i) = Dsiim0(i)+Dp2im0(i)+Ds3im0(i)
  Dsre0(i) = Dpire0(i)+Dp3re0(i)
  Dsim0(i) = Dpiim0(i)+Dp3im0(i)
16 continue
```

Appendix C. Fortran code for second order nonlinear interactions

```

!- half-point calculation -----
! calculate the fcts and derivs @ the half-point

      do 31 i=1,N,1
      tprehp(i) = tpre1m(i) + Dhp0*Dpre0(i) + Dhp1m*Dpre1m(i)
      tpimhp(i) = tpim1m(i) + Dhp0*Dpim0(i) + Dhp1m*Dpim1m(i)

      tsrehp(i) = tsre1m(i) + Dhp0*Dsre0(i) + Dhp1m*Dsre1m(i)
      tsimhp(i) = tsim1m(i) + Dhp0*Dsim0(i) + Dhp1m*Dsim1m(i)

      tirehp(i) = tire1m(i) + Dhp0*Dire0(i) + Dhp1m*Dire1m(i)
      tiimhp(i) = tiim1m(i) + Dhp0*Diim0(i) + Dhp1m*Diim1m(i)

      t2rehp(i) = t2re1m(i) + Dhp0*D2re0(i) + Dhp1m*D2re1m(i)
      t2imhp(i) = t2im1m(i) + Dhp0*D2im0(i) + Dhp1m*D2im1m(i)

      t3rehp(i) = t3re1m(i) + Dhp0*D3re0(i) + Dhp1m*D3re1m(i)
      t3imhp(i) = t3im1m(i) + Dhp0*D3im0(i) + Dhp1m*D3im1m(i)

31      continue

      call corhpps          ! returns Dprehp() and Dpimhp()
      call corhppi
      call corhpsi

      do 17 i=1,N,1
      Dprehp(i) = Dsirehp(i)+Dp2rehp(i)+Ds3rehp(i)
      Dpimhp(i) = Dsiimhp(i)+Dp2imhp(i)+Ds3imhp(i)
      Dsrehp(i) = Dpirehp(i)+Dp3rehp(i)
      Dsimhp(i) = Dpiimhp(i)+Dp3imhp(i)
17      continue

!-- predictor -----

      delta = dble(0)          ! set criterium for loop to 0
      tmpre = dble(0)         ! reset aux variable
      tmpim = dble(0)

      do 150 i=1,N,1
      tpre1p(i) = p0*tpre0(i) - p1m*tpre1m(i) + Dphp*Dprehp(i) -
      Dp0*Dpre0(i) - Dp1m*Dpre1m(i)

```

Appendix C. Fortran code for second order nonlinear interactions

```

tpim1p(i) = p0*tpim0(i) - p1m*tpim1m(i) + Dphp*Dpimhp(i) -
Dp0*Dpim0(i) - Dp1m*Dpim1m(i)
tsrelp(i) = p0*tsre0(i) - p1m*tsre1m(i) + Dphp*Dsrehp(i) -
Dp0*Dsre0(i) - Dp1m*Dsre1m(i)
tsim1p(i) = p0*tsim0(i) - p1m*tsim1m(i) + Dphp*Dsimhp(i) -
Dp0*Dsim0(i) - Dp1m*Dsim1m(i)
tire1p(i) = p0*tire0(i) - p1m*tire1m(i) + Dphp*Direhp(i) -
Dp0*Dire0(i) - Dp1m*Dire1m(i)
tiim1p(i) = p0*tiim0(i) - p1m*tiim1m(i) + Dphp*Diimhp(i) -
Dp0*Diim0(i) - Dp1m*Diim1m(i)
t2relp(i) = p0*t2re0(i) - p1m*t2re1m(i) + Dphp*D2rehp(i) -
Dp0*D2re0(i) - Dp1m*D2re1m(i)
t2im1p(i) = p0*t2im0(i) - p1m*t2im1m(i) + Dphp*D2imhp(i) -
Dp0*D2im0(i) - Dp1m*D2im1m(i)
t3relp(i) = p0*t3re0(i) - p1m*t3re1m(i) + Dphp*D3rehp(i) -
Dp0*D3re0(i) - Dp1m*D3re1m(i)
t3im1p(i) = p0*t3im0(i) - p1m*t3im1m(i) + Dphp*D3imhp(i) -
Dp0*D3im0(i) - Dp1m*D3im1m(i)

tmpre=tmpre1p(i)
tmpim=tmpim1p(i)
delta=delta+dsqrt(tmpre**2 + tmpim**2)
150      continue

!-- corrector -----

200      loopnum = 1           ! reset loop counter
        deltaold = delta
        delta    = dble(0)    ! reset delta
        tmpre   = dble(0)    ! reset aux variable
        tmpim   = dble(0)

        call cor1pps          ! returns D0re1p() and Dpim1p()
        call cor1ppi
        call cor1psi

        do 18 i=1,N,1
        Dpre1p(i) = Dsire1p(i)+Dp2re1p(i)+Ds3re1p(i)
        Dpim1p(i) = Dsiim1p(i)+Dp2im1p(i)+Ds3im1p(i)
        Dsre1p(i) = Dpire1p(i)+Dp3re1p(i)
        Dsim1p(i) = Dpiim1p(i)+Dp3im1p(i)
18      continue

```


Appendix C. Fortran code for second order nonlinear interactions

```

do 250 i=1,N,1
  tpre1p(i) = c0*tpre0(i) - c1m*tpre1m(i) + Dchp*Dprehp(i) +
             Dc1p*Dpre1p(i) + Dc0*Dpre0(i) - Dc1m*Dpre1m(i)
  tpim1p(i) = c0*tpim0(i) - c1m*tpim1m(i) + Dchp*Dpimhp(i) +
             Dc1p*Dpim1p(i) + Dc0*Dpim0(i) - Dc1m*Dpim1m(i)
  tsre1p(i) = c0*tsre0(i) - c1m*tsre1m(i) + Dchp*Dsrehp(i) +
             Dc1p*Dsre1p(i) + Dc0*Dsre0(i) - Dc1m*Dsre1m(i)
  tsim1p(i) = c0*tsim0(i) - c1m*tsim1m(i) + Dchp*Dsimhp(i) +
             Dc1p*Dsim1p(i) + Dc0*Dsim0(i) - Dc1m*Dsim1m(i)
  tire1p(i) = c0*tire0(i) - c1m*tire1m(i) + Dchp*Direhp(i) +
             Dc1p*Dire1p(i) + Dc0*Dire0(i) - Dc1m*Dire1m(i)
  tiim1p(i) = c0*tiim0(i) - c1m*tiim1m(i) + Dchp*Diimhp(i) +
             Dc1p*Diim1p(i) + Dc0*Diim0(i) - Dc1m*Diim1m(i)
  t2re1p(i) = c0*t2re0(i) - c1m*t2re1m(i) + Dchp*D2rehp(i) +
             Dc1p*D2re1p(i) + Dc0*D2re0(i) - Dc1m*D2re1m(i)
  t2im1p(i) = c0*t2im0(i) - c1m*t2im1m(i) + Dchp*D2imhp(i) +
             Dc1p*D2im1p(i) + Dc0*D2im0(i) - Dc1m*D2im1m(i)
  t3re1p(i) = c0*t3re0(i) - c1m*t3re1m(i) + Dchp*D3rehp(i) +
             Dc1p*D3re1p(i) + Dc0*D3re0(i) - Dc1m*D3re1m(i)
  t3im1p(i) = c0*t3im0(i) - c1m*t3im1m(i) + Dchp*D3imhp(i) +
             Dc1p*D3im1p(i) + Dc0*D3im0(i) - Dc1m*D3im1m(i)

  tmpre=tpre1p(i)
  tmpim=tpim1p(i)

  delta=delta+dsqrt(tmpre**2 + tmpim**2)
250  continue

  tmp=dabs((delta-deltaold)/(delta+minval))

  print *, 'delta = ',tmp, ' loop # ',loopnum

  if (loopnum.ge.loopmax) goto 300
  loopnum=loopnum+1
  if (tmp.gt.error) goto 200

300  continue

!-- shifting arrays -----
  tpa = dble(0)
  tsa = dble(0)
  tia = dble(0)

```

Appendix C. Fortran code for second order nonlinear interactions

```
t2a = dble(0)
t3a = dble(0)

do 400 i=1,N,1          ! backwards
Dpre1m(i) = Dpre0(i)   ! (n-1) <-- (n)
  Dpre0(i) = Dpre1p(i) ! (n) <-- (n+1)
  tpre1m(i) = tpre0(i) ! (n-1) <-- (n)
  tpre0(i) = tpre1p(i) ! (n) <-- (n+1)

  Dpim1m(i) = Dpim0(i)
  Dpim0(i) = Dpim1p(i)
  tpim1m(i) = tpim0(i)
  tpim0(i) = tpim1p(i)

  Dsre1m(i) = Dsre0(i)
  Dsre0(i) = Dsre1p(i)
  tsre1m(i) = tsre0(i)
  tsre0(i) = tsre1p(i)

  Dsim1m(i) = Dsim0(i)
  Dsim0(i) = Dsim1p(i)
  tsim1m(i) = tsim0(i)
  tsim0(i) = tsim1p(i)

Dire1m(i) = Dire0(i)
  Dire0(i) = Dire1p(i)
  tire1m(i) = tire0(i)
  tire0(i) = tire1p(i)

  Diim1m(i) = Diim0(i)
  Diim0(i) = Diim1p(i)
  tiim1m(i) = tiim0(i)
  tiim0(i) = tiim1p(i)

  D2re1m(i) = D2re0(i)
  D2re0(i) = D2re1p(i)
  t2re1m(i) = t2re0(i)
  t2re0(i) = t2re1p(i)

  D2im1m(i) = D2im0(i)
  D2im0(i) = D2im1p(i)
  t2im1m(i) = t2im0(i)
  t2im0(i) = t2im1p(i)
```

Appendix C. Fortran code for second order nonlinear interactions

```
D3re1m(i) = D3re0(i)
      D3re0(i) = D3re1p(i)
      t3re1m(i) = t3re0(i)
      t3re0(i) = t3re1p(i)

      D3im1m(i) = D3im0(i)
      D3im0(i) = D3im1p(i)
      t3im1m(i) = t3im0(i)
      t3im0(i) = t3im1p(i)

tpa = tpa+(tpre0(i)**2 + tpim0(i)**2)
      tsa = tsa+(tsre0(i)**2 + tsim0(i)**2)
      tia = tia+(tire0(i)**2 + tiim0(i)**2)
      t2a = t2a+(t2re0(i)**2 + t2im0(i)**2)
t3a = t3a+(t3re0(i)**2 + t3im0(i)**2)

400      continue

      tpa = tpa*dom
      tsa = tsa*dom
           tia = tia*dom
      t2a = t2a*dom
           t3a = t3a*dom

      write(9,998)dist, dble(np*tpa+ns*tsa+ni*tia+n2*t2a+n3*t3a),
      dble(np*tpa),dble(ns*tsa),dble(ni*tia),dble(n2*t2a),dble(n3*t3a),
      tpre0(4096),tpim0(4096),tsre0(4096),tsim0(4096)

!-----

30      continue           ! end of inside loop

      do 40 i=1,N,1
      agp=dble((kvp(i))*dist)
      ags=dble((kvs(i))*dist)
           agi=dble((kvi(i))*dist)
      ag2=dble((kv2(i))*dist)
           ag3=dble((kv3(i))*dist)

      agp=dmod(agp,tpi)
      ags=dmod(ags,tpi)
           agi=dmod(agi,tpi)
```

Appendix C. Fortran code for second order nonlinear interactions

```

ag2=dmod(ag2, tpi)
      ag3=dmod(ag3, tpi)

cwp(i)=cplx( (tpre1m(i)*dcos(agp))+(tpim1m(i)*dsin(agp)),
      -(tpre1m(i)*dsin(agp))+(tpim1m(i)*dcos(agp)) )
cws(i)=cplx( (tsre1m(i)*dcos(ags))+(tsim1m(i)*dsin(ags)),
      -(tsre1m(i)*dsin(ags))+(tsim1m(i)*dcos(ags)) )
cwi(i)=cplx( (tire1m(i)*dcos(agi))+(tiim1m(i)*dsin(agi)),
      -(tire1m(i)*dsin(agi))+(tiim1m(i)*dcos(agi)) )
cw2(i)=cplx( (t2re1m(i)*dcos(ag2))+(t2im1m(i)*dsin(ag2)),
      -(t2re1m(i)*dsin(ag2))+(t2im1m(i)*dcos(ag2)) )
cw3(i)=cplx( (t3re1m(i)*dcos(ag3))+(t3im1m(i)*dsin(ag3)),
      -(t3re1m(i)*dsin(ag3))+(t3im1m(i)*dcos(ag3)) )

40      continue

      call fft(cwp,N,-1)
      call fft(cws,N,-1)
      call fft(cwi,N,-1)

call fft(cw2,N,-1)
      call fft(cw3,N,-1)

print*, '-----> files'

do 70 i=1,N,1
      k=((1-N)/2)-1+i
      write(1,*)dist, pf*dreal(cwp(i)), pf*dimag(cwp(i))/dble(1e13)
      write(2,*)dist, pf*dreal(cws(i)), pf*dimag(cws(i))/dble(1e13)
      write(3,*)dist, pf*dreal(cwi(i)), pf*dimag(cwi(i))/dble(1e13)
      write(4,*)dist, pf*dreal(cw2(i)), pf*dimag(cw2(i))/dble(1e13)
      write(5,*)dist, pf*dreal(cw3(i)), pf*dimag(cw3(i))/dble(1e13)

70      continue

      write(1,*)
      write(1,*)
      write(2,*)
      write(2,*)
      write(3,*)
      write(3,*)
      write(4,*)
      write(4,*)

```

Appendix C. Fortran code for second order nonlinear interactions

```
        write(5,*)
        write(5,*)

20      continue          ! end of outside loop

!-- end -----

994    format(a)
995    format(a,f6.4)      ! for z-output on screen
!996   format(13x,a,f7.5,1x,a,i1) ! for delta/loop#-output on screen
996    format(f7.5,i1)    ! for delta/loop#-output on screen
998    format(f4.2,10(2x,e10.4)) ! for energy.dat
999    format(f4.2,1x,d8.4,1x,d18.4) ! for o.dat, e.dat, 2.dat

        close(1)
        close(2)
        close(3)
          close(4)
            close(5)
        close(9)

2      continue
1      continue

        print *, ' I am done'
        pause
        stop
        end

!-----
! subroutines
!-----

!- initial.f -----

        subroutine initial

        parameter(N=8191)          !has to be (2^n)-1

        common/art0/tpre0,tpim0,tsre0,tsim0,tire0,tiim0,t2re0,t2im0,
          t3re0,t3im0
```

Appendix C. Fortran code for second order nonlinear interactions

```

common/vdbl/det,dom,facp,facs,faci,fac2,fac3
common/vinp/taup,taus,dp,ds,ampp,amps,np,ns,ni,n2,n3
common/vcon/minval,tpa,tsa,tia,t2a,t3a
common/vrea/dist

integer i,k
double precision taup,taus,dp,ds,ampp,amps,np,ns,ni,n2,n3,minval,
      tpa,tsa,tia,t2a,t3a
double precision det,dom,tmp,mulo,mule,facp,facs,faci,fac2,fac3
double precision tpre0(N),tpim0(N),tsre0(N),tsim0(N),tire0(N),tiim0(N),
      t2re0(N),t2im0(N),t3re0(N),t3im0(N)

double precision tpaim,tpare,tsare,tsaim,ftor,tpi, abar
complex*16      cwp(N),cws(N),cwi(N),cw2(N),cw3(N)

ftor=0.707106781188d+0      ! equals 1/sqrt(2)
tpi =6.28318530718d+0      ! equals 2 PI

88      format(6(e9.2,2x))

! ----- generates gaussian pulses -----
abar = -0.5d+0
write (8,668) -abar,abar
668 format(18H chirp coefficient,f8.3,9H for pump,4x,f8.3,11H for signal)

do 10 i=1,N,1
      k=((1-N)/2)-1+i
      cwp(i)=cmplx(ampp*dexp(-(((dble(k)*det)-dp)*dble(1.17741)/taup)**2)
      *dcos(-abar*(((dble(k)*det)-dp)*dble(1.17741)/taup)**2),
      ampp*dexp(-(((dble(k)*det)-dp)*dble(1.17741)/taup)**2)
      *dsin(-abar*(((dble(k)*det)-dp)*dble(1.17741)/taup)**2))

      cws(i)=cmplx(amps*dexp(-(((dble(k)*det)-ds)*dble(1.17741)/taus)**2)
      *dcos(abar*(((dble(k)*det)-ds)*dble(1.17741)/taus)**2),
      amps*dexp(-(((dble(k)*det)-ds)*dble(1.17741)/taus)**2)
      *dsin(abar*(((dble(k)*det)-ds)*dble(1.17741)/taus)**2))

      tire0(i)=dble(0)
      tiim0(i)=dble(0)
      t2re0(i)=dble(0)
      t2im0(i)=dble(0)
      t3re0(i)=dble(0)
      t3im0(i)=dble(0)

```

Appendix C. Fortran code for second order nonlinear interactions

```
10      continue

      call fft(cwp,N,1)
         call fft(cws,N,1)

      do 20 i=1,N,1
         tpre0(i) = dreal(cwp(i))
            tpim0(i) = dimag(cwp(i))
         tsre0(i) = dreal(cws(i))
            tsim0(i) = dimag(cws(i))
20      continue
      print *, 'initial - survived'

      return
      end

!- selma.f -----

      subroutine selma

      parameter(N=8191)           !has to be (2^n)-1

      common/arkv/kvp,kvs,kvi,kv2,kv3,kvg
      common/vdbl/det,dom,facp,facs,faci,fac2,fac3
      common/angles/woff,tilt,intang
      common/kvecs/k0p,k1p,k0s,k1s,k0i,k1i,k02,k12,k03,k13

      integer i,j,k,l
      double precision kvp(N),kvs(N),kvi(N),kv2(N),kv3(N),kvg
      double precision det,dom,facp,facs,faci,fac2,fac3
      double precision ae,be,ce,de,Ame,Bme,Cme,Dme,Eme,Fme,selm,selmir
      double precision tsubp,tsubs,tsubi,tsub2,tsub3
      double precision tpne,tsne,tine,t2ne,t3ne
      double precision lambda,tstep,angle,lsqf,lsq2,omega,comega,omegap,
         omegas,omegai,omega2,omega3
         double precision woff, tilt, coeff,intang
         double precision k0p,k1p,k0s,k1s,k0i,k1i,k02,k12,k03,k13

!---- define constants -----
!      for PPLN
      ae = 4.582d+0           ! constants for the e polarization in VIS region
      be = 0.099169d+0
      ce = 0.044432d+0
```

Appendix C. Fortran code for second order nonlinear interactions

```
de = 0.02195d+0

! constants for e polarization in the IR region up to 5um
! data taken from JOSA B, Vol. 14, Issue 12, pp. 3319-3322 (1997)
Ame = 2.2454d+0
Bme = 0.01242d+0
Cme = 1.3005d+0
Dme = 0.05313d+0
Eme = 6.8972d+0
Fme = 331.33d+0

lambda=0.770d+0           !in um
comega=1883.65156731d+0/lambda !center frequency in 1/ps
coeff =3.14159265359d+0/180d+0

!----- calculating k-vectors -----

do 100 i=1,N,1
  l=i-((N+1)/2)
  omega=comega+dbble(l)*dom
  omegap = omega
  omegas = comega*770d+0/1140d+0 +dbble(l)*dom
  omegai = comega*770d+0/2372.432432d+0 +dbble(l)*dom
  omega2 = comega*2d+0 +dbble(l)*dom
  omega3 = comega*770d+0/459.5811518d+0 +dbble(l)*dom

!-----
! calculate the n_e for pump

      tpne = dsqrt(selmir(Ame,Bme,Cme,Dme,Eme,Fme,omegap))

!-----
! calculate the n_e for signal, idler, 2nd harmonic and sum frequencies

      tsne = dsqrt(selmir(Ame,Bme,Cme,Dme,Eme,Fme,omegas))
      tine = dsqrt(selmir(Ame,Bme,Cme,Dme,Eme,Fme,omegai))
      t2ne = dsqrt(selmir(Ame,Bme,Cme,Dme,Eme,Fme,omega2))
      t3ne = dsqrt(selmir(Ame,Bme,Cme,Dme,Eme,Fme,omega3))

      tsubp = (k1p*dbble(l)*dom)

      kvp(i)=(33.4448160535d+0*tpne*omegap)-tsubp !33.444... is 1/c in ps/cm
      kvs(i)=(33.4448160535d+0*tsne*omegas)-tsubp
```


Appendix C. Fortran code for second order nonlinear interactions

```

        kvi(i)=(33.4448160535d+0*tine*omegai)-tsubp
        kv2(i)=(33.4448160535d+0*t2ne*omega2)-tsubp
        kv3(i)=(33.4448160535d+0*t3ne*omega3)-tsubp

        write(6,*) omega,kvp(i),kvs(i)
100      continue
          close(6)
print *,'done with k-vectors'
        return
        end

#####
!---- n^2 calculation for visible -----

        function selm(ta,tb,tc,td,omega)

! calculates n^2 according to the sellmaier formula
!
!
!
!           tb           2
! n^2 = ta + ----- - td * (K/omega)
!           2
!         (K/omega) - tc

        double precision ta,tb,tc,td,omega,tfac,selm

        tfac = 1883.65156731d+0      !2pi*c in um/ps
        selm = ta + (tb/(((tfac/omega)**2) - tc)) - (td * (tfac/omega)**2)
        return
        end

!---- n^2 calculation for infrared-----■

        function selmir(tA,tB,tC,tD,tE,tF,omega)

! calculates n^2 according to the sellmaier formula
!
!
!           2           2           2
!         tA *(K/omega)   tC *(K/omega)   tE *(K/omega)
! n^2 = 1 + ----- + ----- + -----
!           2           2           2
!         (K/omega) - tB   (K/omega) - tD   (K/omega) - tF

```

Appendix C. Fortran code for second order nonlinear interactions

```

double precision tA,tB,tC,tD,tE,tF,omega,tfac,selmir

    tfac = 1883.65156731d+0          !2pi*c in um/ps
    selmir = 1d+0+(tA*(tfac/omega)**2/((tfac/omega)**2-tB))+tC*(tfac/omega)**2
    /((tfac/omega)**2-tD))+tE*(tfac/omega)**2/((tfac/omega)**2-tF))
        return
    end

!- fft.f -----

subroutine fft(tmparray,N,sign)

!-----
! performs the Symmetric Complex Fast-Fourier Transform
!
! tmparray : double precision complex array of length N
!           the input function is given in this array and the
!           output is written into the same array
! N         : integer, to be (2^n)-1,
!           defines the arraylength
! sign      : integer, to be +1 or -1
!           +1 : FFT, -1 : inverse FFT
!
! det       : given as common in double precision
!           defines delta t
! dom       : given as common in double precision
!           defines delta omega
!           dom := (2*PI)/(N*det)
!
! F(w) := 1/sqrt(2*Pi) Int{ f(t) e^( i w*t) } dt ; sign: +1
! f(t) := 1/sqrt(2*Pi) Int{ F(w) e^(-i w*t) } dw ; sign: -1
!-----

!-- declare variables -----

    parameter(m=8191)          !has to be (2^n)-1

    integer i,j,k,N,sign
    common/vdbl/det,dom,facp,facs,faci,fac2,fac3
    double precision array(2*(m+1)),factor
    double precision det,dom,facp,facs,faci,fac2,fac3

```

Appendix C. Fortran code for second order nonlinear interactions

```

complex*16 tmparray(m)

!-- into workarrays -----

      j=0
      do 1 i=((N+1)/2),N,1
        j=j+1
          array(j) = real(tmparray(i))
        j=j+1
          array(j) = dimag(tmparray(i))
1    continue
      j=j+1
      array(j) = dble(0.)
      j=j+1
      array(j) = dble(0.)
      do 2 i=1,((N-1)/2),1
        j=j+1
          array(j) = real(tmparray(i))
        j=j+1
          array(j) = dimag(tmparray(i))
2    continue

!-- call fft -----

      call four1(array,(N+1),sign)

!-- sort -----

      if(sign.gt.0) then
        factor=det/2.50662827463d+0
      else
        factor=2.50662827463d+0/(dble(N+1)*det)
      endif
      j=0
      do 100 i=1,N,2
        k=N+1-i
        j=j+1
        tmparray(j)=cmplx(array(k),array(k+1))*factor
100   continue
      do 110 i=(N+4),((2*N)+1),2
        k=(3*N)+5-i
        j=j+1
        tmparray(j)=cmplx(array(k),array(k+1))*factor

```

Appendix C. Fortran code for second order nonlinear interactions

```
110    continue
      return
      end

!-- subroutine from Numerical Recipes -----

      subroutine four1(DATA,NN,ISIGN)

      INTEGER NN,ISIGN,I,N,J,M,MMAX
      DOUBLE PRECISION WR,WI,WPR,WPI,WTEMP,THETA
      DOUBLE PRECISION DATA(*)

      N=2*NN
      J=1
      DO 11 I=1,N,2
        IF(J.GT.I)THEN
          TEMPR=DATA(J)
          TEMPI=DATA(J+1)
          DATA(J)=DATA(I)
          DATA(J+1)=DATA(I+1)
          DATA(I)=TEMPR
          DATA(I+1)=TEMPI
        ENDIF
      M=N/2
1      IF ((M.GE.2).AND.(J.GT.M)) THEN
        J=J-M
        M=M/2
        GO TO 1
      ENDIF
      J=J+M
11     CONTINUE
      MMAX=2
2      IF (N.GT.MMAX) THEN
        ISTEP=2*MMAX
        THETA=DBLE(6.28318530717959D0)/DBLE(ISIGN*MMAX)
        WPR=-2.DO*DSIN(0.5D0*THETA)**2
        WPI=DSIN(THETA)
        WR=1.DO
        WI=0.DO
        DO 13 M=1,MMAX,2
          DO 12 I=M,N,ISTEP
            J=I+MMAX
            TEMPR=SNGL(WR)*DATA(J)-SNGL(WI)*DATA(J+1)
```

Appendix C. Fortran code for second order nonlinear interactions

```

        TEMPI=SNGL(WR)*DATA(J+1)+SNGL(WI)*DATA(J)
        DATA(J)=DATA(I)-TEMPR
        DATA(J+1)=DATA(I+1)-TEMPI
        DATA(I)=DATA(I)+TEMPR
        DATA(I+1)=DATA(I+1)+TEMPI
12      CONTINUE
        WTEMP=WR
        WR=WR*WPR-WI*WPI+WR
        WI=WI*WPR+WTEMP*WPI+WI
13      CONTINUE
        MMAX=ISTEP
        GO TO 2
        ENDIF
        RETURN
      END

!- pfield.f -----

      subroutine pfield(array,N,fsqr)

!-----
! calculates the energy of the largest pulse in the array
! neglecting satellites (largest in amplitude)
!
! array : complex array, double precision, N points
! N      : number of points in the array
! fsqr  : field squared in double precision; has to be
!         multiplied by n and det or dom
!
! written by Jens Biegert, 07/97
!-----

      integer i,j,N,tmpnum,upnum,lonum
      double precision tmpnew,tmpold,tmpminold,tmpminnew,fsqr
      complex*16 array(N)

! -- get max amplitude -----

      tmpold = 0.
      tmpnum = 0
      do 10 i=1,N,1
        tmpnew=dsqrt(real(array(i))**2 + dimag(array(i))**2)
        if(tmpnew.ge.tmpold)then
```

Appendix C. Fortran code for second order nonlinear interactions

```
        tmpold = tmpnew          ! value of max amp
        tmpnum = i              ! position of max
    else
        goto 15
    endif
10  continue
15  continue

! -- find min for points to the right of the max -----

    tmpminold = tmpold          ! put max amp in here
    upnum = 0
    do 20 i=tmpnum,N,1
        tmpminnew=sqrt(real(array(i))**2 + (array(i))**2)
        if(tmpminnew.le.tmpminold)then
            tmpminold = tmpminnew
            upnum = i
        else
            goto 25
        endif
20  continue
25  continue

! -- find min for points to the left of the max -----

    tmpminold = tmpold          ! put max amp in here
    lonum = 0
    do 30 i=1,tmpnum-1,1
        j = tmpnum-i
        tmpminnew=dsqrt(real(array(j))**2 + dimag(array(j))**2)
        if(tmpminnew.le.tmpminold)then
            tmpminold = tmpminnew
            lonum = j
        else
            goto 35
        endif
30  continue
35  continue

! -- calculate the field squared -----

    fsqr = dble(0.)
    do 40 i=lonum,upnum,1
```

Appendix C. Fortran code for second order nonlinear interactions

```

        fsqr = fsqr+(real(array(i))**2 + dimag(array(i))**2)
40 continue

! isqr has to be multiplied by det or dom and the
! refractive index

        return
        end

!- cor0ps.f -----

        subroutine cor0ps

        common/art0/tpre0,tpim0,tsre0,tsim0,tire0,tiim0,t2re0,t2im0,
                t3re0,t3im0
        common/arD0/Dpre0,Dpim0,Dsre0,Dsim0,Dire0,Diim0,D2re0,D2im0,
                D3re0,D3im0,Dsire0,Dsiim0,Dp2re0,Dp2im0,Ds3re0,Ds3im0,Dpire0,
                Dpiim0,Dp3re0,Dp3im0
        common/arkv/kvp,kvs,kvi,kv2,kv3,kvg
        common/vdbl/det,dom,facp,facs,faci,fac2,fac3
        common/vrea/dist

        parameter(N=8191)          !has to be (2^n)-1

        integer          i
        double precision dist
        double precision tpre0(N),tpim0(N),tsre0(N),tsim0(N),tire0(N),tiim0(N),
        t2re0(N),t2im0(N),t3re0(N),t3im0(N)
        double precision Dpre0(N),Dpim0(N),Dsre0(N),Dsim0(N),Dire0(N),Diim0(N),
        D2re0(N),D2im0(N),D3re0(N),D3im0(N),Dsire0(N),Dsiim0(N),Dp2re0(N),
        Dp2im0(N),Ds3re0(N),Ds3im0(N),Dpire0(N),Dpiim0(N),Dp3re0(N),Dp3im0(N)
        double precision kvp(N),kvs(N),kvi(N),kv2(N),kv3(N),kvg
        double precision arg1,arg2,arg3,tpi
        double precision det,dom,facp,facs,faci,fac2,fac3
        complex*16          cwp(N),cws(N),cwi(N)

        tpi = 6.29318530718d+0

        do 20 i=1,N,1          ! 1D arrays --> 2D arrays
                arg1=-kvp(i)*dist          ! always define as e^(-i phi)
                arg2=-kvs(i)*dist
                arg1=dmod(arg1,tpi)

```

Appendix C. Fortran code for second order nonlinear interactions

```

    arg2=dmod(arg2, tpi)
    cwp(i)=cmplx( (tpre0(i)*dcos(arg1))-(tpim0(i)*dsin(arg1)),
                 (tpre0(i)*dsin(arg1))+(tpim0(i)*dcos(arg1)) )
    cws(i)=cmplx( (tsre0(i)*dcos(arg2))-(tsim0(i)*dsin(arg2)),
                 (tsre0(i)*dsin(arg2))+(tsim0(i)*dcos(arg2)) )
20    continue

    call fft(cwp,N,-1)
call fft(cws,N,-1)

    do 30 i=1,N,1
    cwi(i)=cmplx( real(cws(i))*real(cwp(i))+dimag(cwp(i))*dimag(cws(i)),
                 dimag(cwp(i))*real(cws(i)) - real(cwp(i))*dimag(cws(i)) )
30    continue

    call fft(cwi,N,1)

    do 40 i=1,N,1          ! 2D arrays --> 1D arrays
    arg3=+kvi(i)*dist+kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
    Dire0(i)=faci*( (dimag(cwi(i))*dcos(arg3))
                   +(real(cwi(i))*dsin(arg3)) )
    Diim0(i)=faci*( (dimag(cwi(i))*dsin(arg3))
                   -(real(cwi(i))*dcos(arg3)) )
    arg3=+kvi(i)*dist-kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
    Dire0(i)=Dire0(i)-faci*( (dimag(cwi(i))*dcos(arg3))
                            +(real(cwi(i))*dsin(arg3)) )
    Diim0(i)=Diim0(i)-faci*( (dimag(cwi(i))*dsin(arg3))
                            -(real(cwi(i))*dcos(arg3)) )
    arg3=+kvi(i)*dist+3d+0*kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
    Dire0(i)=Dire0(i)+1/3d+0*faci*( (dimag(cwi(i))*dcos(arg3))
                                    +(real(cwi(i))*dsin(arg3)) )
    Diim0(i)=Diim0(i)+1/3d+0*faci*( (dimag(cwi(i))*dsin(arg3))
                                    -(real(cwi(i))*dcos(arg3)) )
    arg3=+kvi(i)*dist-3d+0*kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
    Dire0(i)=Dire0(i)-1/3d+0*faci*( (dimag(cwi(i))*dcos(arg3))
                                    +(real(cwi(i))*dsin(arg3)) )
    Diim0(i)=Diim0(i)-1/3d+0*faci*( (dimag(cwi(i))*dsin(arg3))
                                    -(real(cwi(i))*dcos(arg3)) )
    arg3=+kvi(i)*dist+5d+0*kvg*dist-tpi/4d+0

```


Appendix C. Fortran code for second order nonlinear interactions

```

arg3=dmod(arg3, tpi)
  Dire0(i)=Dire0(i)+1/5d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diim0(i)=Diim0(i)+1/5d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
  arg3=+kvi(i)*dist-5d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3, tpi)
  Dire0(i)=Dire0(i)-1/5d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diim0(i)=Diim0(i)-1/5d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
40  continue

  return
  end

!- cor1mps.f -----

subroutine cor1mps

common/art1m/tpre1m, tpim1m, tsre1m, tsim1m, tire1m, tiim1m, t2re1m, t2im1m,
  t3re1m, t3im1m
common/arD1m/Dpre1m, Dpim1m, Dsre1m, Dsim1m, Dire1m, Diim1m, D2re1m, D2im1m,
  D3re1m, D3im1m, Dsire1m, Dsiim1m, Dp2re1m, Dp2im1m, Ds3re1m, Ds3im1m,
Dpire1m, Dpiim1m, Dp3re1m, Dp3im1m
common/arkv/kvp, kvs, kvi, kv2, kv3, kvg
common/vdbl/det, dom, facp, facs, faci, fac2, fac3
common/vrea/dist

parameter(N=8191)          !has to be (2^n)-1

integer          i
double precision dist
double precision tpre1m(N), tpim1m(N), tsre1m(N), tsim1m(N), tire1m(N), tiim1m(N),
t2re1m(N), t2im1m(N), t3re1m(N), t3im1m(N)
double precision Dpre1m(N), Dpim1m(N), Dsre1m(N), Dsim1m(N), Dire1m(N), Diim1m(N),
D2re1m(N), D2im1m(N), D3re1m(N), D3im1m(N), Dsire1m(N), Dsiim1m(N), Dp2re1m(N),
Dp2im1m(N), Ds3re1m(N), Ds3im1m(N), Dpire1m(N), Dpiim1m(N), Dp3re1m(N), Dp3im1m(N)
double precision kvp(N), kvs(N), kvi(N), kv2(N), kv3(N), kvg
double precision arg1, arg2, arg3, tpi
double precision det, dom, facp, facs, faci, fac2, fac3
complex*16      cwp(N), cws(N), cwi(N)

```

Appendix C. Fortran code for second order nonlinear interactions

```

tpi = 6.29318530718d+0

do 20 i=1,N,1                ! 1D arrays --> 2D arrays
    arg1=-kvp(i)*dist        ! always define as e-i phi
    arg2=-kvs(i)*dist
    arg1=dmod(arg1, tpi)
    arg2=dmod(arg2, tpi)
    cwp(i)=cmplx( (tpre1m(i)*dcos(arg1))-(tpim1m(i)*dsin(arg1)),
                  (tpre1m(i)*dsin(arg1))+(tpim1m(i)*dcos(arg1)) )
    cws(i)=cmplx( (tsre1m(i)*dcos(arg2))-(tsim1m(i)*dsin(arg2)),
                  (tsre1m(i)*dsin(arg2))+(tsim1m(i)*dcos(arg2)) )
20  continue

    call fft(cwp,N,-1)
    call fft(cws,N,-1)

do 30 i=1,N,1
    cwi(i)=cmplx( real(cws(i))*real(cwp(i)) + dimag(cwp(i))*dimag(cws(i)),
                  dimag(cwp(i))*real(cws(i)) - real(cwp(i))*dimag(cws(i)) )
30  continue

    call fft(cwi,N,1)

do 40 i=1,N,1                ! 2D arrays --> 1D arrays
    arg3=+kvi(i)*dist+kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
    Dire1m(i)=faci*( (dimag(cwi(i))*dcos(arg3))
                     +(real(cwi(i))*dsin(arg3)) )
    Diim1m(i)=faci*( (dimag(cwi(i))*dsin(arg3))
                     -(real(cwi(i))*dcos(arg3)) )
    arg3=+kvi(i)*dist-kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
    Dire1m(i)=Dire1m(i)-faci*( (dimag(cwi(i))*dcos(arg3))
                               +(real(cwi(i))*dsin(arg3)) )
    Diim1m(i)=Diim1m(i)-faci*( (dimag(cwi(i))*dsin(arg3))
                               -(real(cwi(i))*dcos(arg3)) )
    arg3=+kvi(i)*dist+3d+0*kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
    Dire1m(i)=Dire1m(i)+1/3d+0*faci*( (dimag(cwi(i))*dcos(arg3))
                                       +(real(cwi(i))*dsin(arg3)) )
    Diim1m(i)=Diim1m(i)+1/3d+0*faci*( (dimag(cwi(i))*dsin(arg3))
                                       -(real(cwi(i))*dcos(arg3)) )
    arg3=+kvi(i)*dist-3d+0*kvg*dist-tpi/4d+0

```

Appendix C. Fortran code for second order nonlinear interactions

```

arg3=dmod(arg3, tpi)
  Dire1m(i)=Dire1m(i)-1/3d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diim1m(i)=Diim1m(i)-1/3d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
  arg3+=kvi(i)*dist+5d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3, tpi)
  Dire1m(i)=Dire1m(i)+1/5d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diim1m(i)=Diim1m(i)+1/5d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
  arg3+=kvi(i)*dist-5d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3, tpi)
  Dire1m(i)=Dire1m(i)-1/5d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diim1m(i)=Diim1m(i)-1/5d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
40   continue

  return
  end

!- corhpps.f -----

  subroutine corhpps

  common/arthp/tprehp, tpimhp, tsrehp, tsimhp, tirehp, tiimhp, t2rehp, t2imhp,
    t3rehp, t3imhp
  common/arDhp/Dprehp, Dpimhp, Dsrehp, Dsimhp, Direhp, Diimhp, D2rehp, D2imhp,
    D3rehp, D3imhp, Dsirehp, Dsiimhp, Dp2rehp, Dp2imhp, Ds3rehp, Ds3imhp,
Dpirehp, Dpiimhp, Dp3rehp, Dp3imhp
  common/arkv/kvp, kvs, kvi, kv2, kv3, kvg
  common/vdbl/det, dom, facp, facs, faci, fac2, fac3
  common/vrea/dist

  parameter(N=8191)          !has to be (2^n)-1

  integer          i
  double precision dist
  double precision tprehp(N), tpimhp(N), tsrehp(N), tsimhp(N), tirehp(N), tiimhp(N),
t2rehp(N), t2imhp(N), t3rehp(N), t3imhp(N)
  double precision Dprehp(N), Dpimhp(N), Dsrehp(N), Dsimhp(N), Direhp(N), Diimhp(N),
D2rehp(N), D2imhp(N), D3rehp(N), D3imhp(N), Dsirehp(N), Dsiimhp(N), Dp2rehp(N),

```

Appendix C. Fortran code for second order nonlinear interactions

```

Dp2imhp(N),Ds3rehp(N),Ds3imhp(N),Dpirehp(N),Dpiimhp(N),Dp3rehp(N),Dp3imhp(N)
double precision kvp(N),kvs(N),kvi(N),kv2(N),kv3(N),kvg
double precision arg1,arg2,arg3,tpi
double precision det,dom,facp,facs,faci,fac2,fac3
      complex*16      cwp(N),cws(N),cwi(N)

tpi = 6.29318530718d+0

      do 20 i=1,N,1          ! 1D arrays --> 2D arrays
          arg1=-kvp(i)*dist    ! always define as e^(-i phi)
          arg2=-kvs(i)*dist
          arg1=dmod(arg1,tpi)
          arg2=dmod(arg2,tpi)
          cwp(i)=cmplx( (tprehp(i)*dcos(arg1))-(tpimhp(i)*dsin(arg1)),
                        (tprehp(i)*dsin(arg1))+(tpimhp(i)*dcos(arg1)) )
          cws(i)=cmplx( (tsrehp(i)*dcos(arg2))-(tsimhp(i)*dsin(arg2)),
                        (tsrehp(i)*dsin(arg2))+(tsimhp(i)*dcos(arg2)) )
20      continue

      call fft(cwp,N,-1)
      call fft(cws,N,-1)

      do 30 i=1,N,1
          cwi(i)=cmplx( real(cws(i))*real(cwp(i)) + dimag(cwp(i))*dimag(cws(i)),
                        dimag(cwp(i))*real(cws(i)) - real(cwp(i))*dimag(cws(i)) )
30      continue

      call fft(cwi,N,1)

      do 40 i=1,N,1          ! 2D arrays --> 1D arrays
          arg3=+kvi(i)*dist+kvg*dist-tpi/4d+0
          arg3=dmod(arg3,tpi)
          Direhp(i)=faci*( (dimag(cwi(i))*dcos(arg3))
                           +(real(cwi(i))*dsin(arg3)) )
          Diimhp(i)=faci*( (dimag(cwi(i))*dsin(arg3))
                           -(real(cwi(i))*dcos(arg3)) )
          arg3=+kvi(i)*dist-kvg*dist-tpi/4d+0
          arg3=dmod(arg3,tpi)
          Direhp(i)=Direhp(i)-faci*( (dimag(cwi(i))*dcos(arg3))
                                       +(real(cwi(i))*dsin(arg3)) )
          Diimhp(i)=Diimhp(i)-faci*( (dimag(cwi(i))*dsin(arg3))
                                       -(real(cwi(i))*dcos(arg3)) )
          arg3=+kvi(i)*dist+3d+0*kvg*dist-tpi/4d+0

```

Appendix C. Fortran code for second order nonlinear interactions

```

arg3=dmod(arg3, tpi)
  Direhp(i)=Direhp(i)+1/3d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diimhp(i)=Diimhp(i)+1/3d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
  arg3=+kvi(i)*dist-3d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3, tpi)
  Direhp(i)=Direhp(i)-1/3d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diimhp(i)=Diimhp(i)-1/3d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
  arg3=+kvi(i)*dist+5d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3, tpi)
  Direhp(i)=Direhp(i)+1/5d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diimhp(i)=Diimhp(i)+1/5d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
  arg3=+kvi(i)*dist-5d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3, tpi)
  Direhp(i)=Direhp(i)-1/5d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diimhp(i)=Diimhp(i)-1/5d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
40    continue

    return
    end

!- cor1pps.f -----

    subroutine cor1pps

    common/art1p/tpre1p, tpim1p, tsre1p, tsim1p, tire1p, tiim1p, t2re1p, t2im1p,
      t3re1p, t3im1p
    common/arD1p/Dpre1p, Dpim1p, Dsre1p, Dsim1p, Dire1p, Diim1p, D2re1p, D2im1p,
      D3re1p, D3im1p, Dsire1p, Dsiim1p, Dp2re1p, Dp2im1p, Ds3re1p, Ds3im1p,
Dpire1p, Dpiim1p, Dp3re1p, Dp3im1p
    common/arkv/kvp, kvs, kvi, kv2, kv3, kvg
    common/vdbl/det, dom, facp, facs, faci, fac2, fac3
    common/vrea/dist

    parameter(N=8191)          !has to be (2^n)-1

```

Appendix C. Fortran code for second order nonlinear interactions

```

integer          i
double precision dist
double precision tpre1p(N),tpim1p(N),tsre1p(N),tsim1p(N),tire1p(N),tiim1p(N),
t2re1p(N),t2im1p(N),t3re1p(N),t3im1p(N)
double precision Dpre1p(N),Dpim1p(N),Dsre1p(N),Dsim1p(N),Dire1p(N),Diim1p(N),
D2re1p(N),D2im1p(N),D3re1p(N),D3im1p(N),Dsire1p(N),Dsiim1p(N),Dp2re1p(N),
Dp2im1p(N),Ds3re1p(N),Ds3im1p(N),Dpire1p(N),Dpiim1p(N),Dp3re1p(N),Dp3im1p(N)
double precision kvp(N),kvs(N),kvi(N),kv2(N),kv3(N),kvg
double precision arg1,arg2,arg3,tpi
double precision det,dom,facp,facs,faci,fac2,fac3
complex*16      cwp(N),cws(N),cwi(N)

tpi = 6.29318530718d+0

do 20 i=1,N,1          ! 1D arrays --> 2D arrays
  arg1=-kvp(i)*dist    ! always define as e-i phi
  arg2=-kvs(i)*dist
  arg1=dmod(arg1,tpi)
  arg2=dmod(arg2,tpi)
  cwp(i)=cmplx( (tpre1p(i)*dcos(arg1))-(tpim1p(i)*dsin(arg1)),
                (tpre1p(i)*dsin(arg1))+(tpim1p(i)*dcos(arg1)) )
  cws(i)=cmplx( (tsre1p(i)*dcos(arg2))-(tsim1p(i)*dsin(arg2)),
                (tsre1p(i)*dsin(arg2))+(tsim1p(i)*dcos(arg2)) )
20  continue

call fft(cwp,N,-1)
call fft(cws,N,-1)

do 30 i=1,N,1
  cwi(i)=cmplx( real(cws(i))*real(cwp(i)) + dimag(cwp(i))*dimag(cws(i)),
                dimag(cwp(i))*real(cws(i)) - real(cwp(i))*dimag(cws(i)) )
30  continue

call fft(cwi,N,1)

do 40 i=1,N,1          ! 2D arrays --> 1D arrays
  arg3=+kvi(i)*dist+kvg*dist-tpi/4d+0
  arg3=dmod(arg3,tpi)
  Dire1p(i)=faci*( (dimag(cwi(i))*dcos(arg3))
                  +(real(cwi(i))*dsin(arg3)) )
  Diim1p(i)=faci*( (dimag(cwi(i))*dsin(arg3))
                  -(real(cwi(i))*dcos(arg3)) )
  arg3=+kvi(i)*dist-kvg*dist-tpi/4d+0

```

Appendix C. Fortran code for second order nonlinear interactions

```

arg3=dmod(arg3, tpi)
  Dire1p(i)=Dire1p(i)-faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diim1p(i)=Diim1p(i)-faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
  arg3=+kvi(i)*dist+3d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3, tpi)
  Dire1p(i)=Dire1p(i)+1/3d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diim1p(i)=Diim1p(i)+1/3d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
  arg3=+kvi(i)*dist-3d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3, tpi)
  Dire1p(i)=Dire1p(i)-1/3d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diim1p(i)=Diim1p(i)-1/3d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
  arg3=+kvi(i)*dist+5d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3, tpi)
  Dire1p(i)=Dire1p(i)+1/5d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diim1p(i)=Diim1p(i)+1/5d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
  arg3=+kvi(i)*dist-5d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3, tpi)
  Dire1p(i)=Dire1p(i)-1/5d+0*faci*( (dimag(cwi(i))*dcos(arg3))
    +(real(cwi(i))*dsin(arg3)) )
  Diim1p(i)=Diim1p(i)-1/5d+0*faci*( (dimag(cwi(i))*dsin(arg3))
    -(real(cwi(i))*dcos(arg3)) )
40   continue

      return
      end

!- cor0pi.f -----
      subroutine cor0pi

      common/art0/tpre0, tpim0, tsre0, tsim0, tire0, tiim0, t2re0, t2im0,
        t3re0, t3im0
      common/arD0/Dpre0, Dpim0, Dsre0, Dsim0, Dire0, Diim0, D2re0, D2im0,
        D3re0, D3im0, Dsire0, Dsiim0, Dp2re0, Dp2im0, Ds3re0, Ds3im0, Dpire0,

```

Appendix C. Fortran code for second order nonlinear interactions

```

        Dpiim0,Dp3re0,Dp3im0
        common/arkv/kvp,kvs,kvi,kv2,kv3,kvg
        common/vdbl/det,dom,facp,facs,faci,fac2,fac3
        common/vrea/dist

        parameter(N=8191)          !has to be (2^n)-1

        integer          i
        double precision dist
        double precision tpre0(N),tpim0(N),tsre0(N),tsim0(N),tire0(N),tiim0(N),
        t2re0(N), t2im0(N),t3re0(N),t3im0(N)
        double precision Dpre0(N),Dpim0(N),Dsre0(N),Dsim0(N),Dire0(N),Diim0(N),
        D2re0(N), D2im0(N),D3re0(N),D3im0(N),Dsire0(N),Dsiim0(N),Dp2re0(N),
        Dp2im0(N),Ds3re0(N), Ds3im0(N),Dpire0(N),Dpiim0(N),Dp3re0(N),Dp3im0(N)
        double precision kvp(N),kvs(N),kvi(N),kv2(N),kv3(N),kvg
        double precision arg1,arg2,arg3,tpi
        double precision det,dom,facp,facs,faci,fac2,fac3
        complex*16      cwp(N),cws(N),cwi(N)

        tpi = 6.29318530718d+0

        do 20 i=1,N,1          ! 1D arrays --> 2D arrays
            arg1=-kvp(i)*dist  ! always define as e^(-i phi)
            arg2=-kvi(i)*dist
            arg1=dmod(arg1,tpi)
            arg2=dmod(arg2,tpi)
            cwp(i)=cmplx( (tpre0(i)*dcos(arg1))-(tpim0(i)*dsin(arg1)),
                (tpre0(i)*dsin(arg1))+(tpim0(i)*dcos(arg1)) )
            cwi(i)=cmplx( (tire0(i)*dcos(arg2))-(tiim0(i)*dsin(arg2)),
                (tire0(i)*dsin(arg2))+(tiim0(i)*dcos(arg2)) )
20        continue

        call fft(cwp,N,-1)
        call fft(cwi,N,-1)

        do 30 i=1,N,1
            cws(i)=cmplx( real(cwi(i))*real(cwp(i)) + dimag(cwp(i))*dimag(cwi(i)),
                dimag(cwp(i))*real(cwi(i)) - real(cwp(i))*dimag(cwi(i)) )
30        continue

        call fft(cws,N,1)

        do 40 i=1,N,1          ! 2D arrays --> 1D arrays

```


Appendix C. Fortran code for second order nonlinear interactions

```

      arg3=+kvs(i)*dist+kvg*dist-tpi/4d+0
      arg3=dmod(arg3, tpi)
      Dpire0(i)=facs*( (dimag(cws(i))*dcos(arg3))
                      +(real(cws(i))*dsin(arg3)) )
      Dpiim0(i)=facs*( (dimag(cws(i))*dsin(arg3))
                      -(real(cws(i))*dcos(arg3)) )
      arg3=+kvs(i)*dist-kvg*dist-tpi/4d+0
      arg3=dmod(arg3, tpi)
      Dpire0(i)=Dpire0(i)-facs*( (dimag(cws(i))*dcos(arg3))
                                +(real(cws(i))*dsin(arg3)) )
      Dpiim0(i)=Dpiim0(i)-facs*( (dimag(cws(i))*dsin(arg3))
                                -(real(cws(i))*dcos(arg3)) )
      arg3=+kvs(i)*dist+3d+0*kvg*dist-tpi/4d+0
      arg3=dmod(arg3, tpi)
      Dpire0(i)=Dpire0(i)+1/3d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                        +(real(cws(i))*dsin(arg3)) )
      Dpiim0(i)=Dpiim0(i)+1/3d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                        -(real(cws(i))*dcos(arg3)) )
      arg3=+kvs(i)*dist-3d+0*kvg*dist-tpi/4d+0
      arg3=dmod(arg3, tpi)
      Dpire0(i)=Dpire0(i)-1/3d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                        +(real(cws(i))*dsin(arg3)) )
      Dpiim0(i)=Dpiim0(i)-1/3d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                        -(real(cws(i))*dcos(arg3)) )
      arg3=+kvs(i)*dist+5d+0*kvg*dist-tpi/4d+0
      arg3=dmod(arg3, tpi)
      Dpire0(i)=Dpire0(i)+1/5d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                        +(real(cws(i))*dsin(arg3)) )
      Dpiim0(i)=Dpiim0(i)+1/5d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                        -(real(cws(i))*dcos(arg3)) )
      arg3=+kvs(i)*dist-5d+0*kvg*dist-tpi/4d+0
      arg3=dmod(arg3, tpi)
      Dpire0(i)=Dpire0(i)-1/5d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                        +(real(cws(i))*dsin(arg3)) )
      Dpiim0(i)=Dpiim0(i)-1/5d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                        -(real(cws(i))*dcos(arg3)) )
40      continue

      return
      end

!- cor1mpi.f -----

```

Appendix C. Fortran code for second order nonlinear interactions

```

subroutine cor1mpi

common/art1m/tpre1m,tpim1m,tsre1m,tsim1m,tire1m,tiim1m,t2re1m,t2im1m,
      t3re1m,t3im1m
common/arD1m/Dpre1m,Dpim1m,Dsre1m,Dsim1m,Dire1m,Diim1m,D2re1m,D2im1m,
      D3re1m,D3im1m,Dsire1m,Dsiim1m,Dp2re1m,Dp2im1m,Ds3re1m,Ds3im1m,
Dpire1m,Dpiim1m,Dp3re1m,Dp3im1m
common/arkv/kvp,kvs,kvi,kv2,kv3,kvg
common/vdbl/det,dom,facp,facs,faci,fac2,fac3
common/vrea/dist

parameter(N=8191)          !has to be (2^n)-1

integer          i
double precision dist
double precision tpre1m(N),tpim1m(N),tsre1m(N),tsim1m(N),tire1m(N),tiim1m(N),
t2re1m(N),t2im1m(N),t3re1m(N),t3im1m(N)
double precision Dpre1m(N),Dpim1m(N),Dsre1m(N),Dsim1m(N),Dire1m(N),Diim1m(N),
D2re1m(N),D2im1m(N),D3re1m(N),D3im1m(N),Dsire1m(N),Dsiim1m(N),Dp2re1m(N),
Dp2im1m(N),Ds3re1m(N),Ds3im1m(N),Dpire1m(N),Dpiim1m(N),Dp3re1m(N),Dp3im1m(N)
double precision kvp(N),kvs(N),kvi(N),kv2(N),kv3(N),kvg
double precision arg1,arg2,arg3,tpi
double precision det,dom,facp,facs,faci,fac2,fac3
      complex*16      cwp(N),cws(N),cwi(N)

tpi = 6.29318530718d+0

do 20 i=1,N,1          ! 1D arrays --> 2D arrays
      arg1=-kvp(i)*dist      ! always define as e^(-i phi)
      arg2=-kvi(i)*dist
      arg1=dmod(arg1,tpi)
      arg2=dmod(arg2,tpi)
      cwp(i)=cmplx( (tpre1m(i)*dcos(arg1))-(tpim1m(i)*dsin(arg1)),
                    (tpre1m(i)*dsin(arg1))+(tpim1m(i)*dcos(arg1)) )
      cwi(i)=cmplx( (tire1m(i)*dcos(arg2))-(tiim1m(i)*dsin(arg2)),
                    (tire1m(i)*dsin(arg2))+(tiim1m(i)*dcos(arg2)) )
20  continue

call fft(cwp,N,-1)
call fft(cwi,N,-1)

do 30 i=1,N,1
      cws(i)=cmplx( real(cwi(i))*real(cwp(i)) + dimag(cwp(i))*dimag(cwi(i)),

```

```

                                dimag(cwp(i))*real(cwi(i)) - real(cwp(i))*dimag(cwi(i)) )
30      continue

      call fft(cws,N,1)

      do 40 i=1,N,1                ! 2D arrays --> 1D arrays
        arg3=+kvs(i)*dist+kvg*dist-tpi/4d+0
      arg3=dmod(arg3,tpi)
        Dpire1m(i)=facs*( (dimag(cws(i))*dcos(arg3))
                          +(real(cws(i))*dsin(arg3)) )
        Dpiim1m(i)=facs*( (dimag(cws(i))*dsin(arg3))
                          -(real(cws(i))*dcos(arg3)) )
        arg3=+kvs(i)*dist-kvg*dist-tpi/4d+0
      arg3=dmod(arg3,tpi)
        Dpire1m(i)=Dpire1m(i)-facs*( (dimag(cws(i))*dcos(arg3))
                                      +(real(cws(i))*dsin(arg3)) )
        Dpiim1m(i)=Dpiim1m(i)-facs*( (dimag(cws(i))*dsin(arg3))
                                      -(real(cws(i))*dcos(arg3)) )
        arg3=+kvs(i)*dist+3d+0*kvg*dist-tpi/4d+0
      arg3=dmod(arg3,tpi)
        Dpire1m(i)=Dpire1m(i)+1/3d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                              +(real(cws(i))*dsin(arg3)) )
        Dpiim1m(i)=Dpiim1m(i)+1/3d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                              -(real(cws(i))*dcos(arg3)) )
        arg3=+kvs(i)*dist-3d+0*kvg*dist-tpi/4d+0
      arg3=dmod(arg3,tpi)
        Dpire1m(i)=Dpire1m(i)-1/3d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                              +(real(cws(i))*dsin(arg3)) )
        Dpiim1m(i)=Dpiim1m(i)-1/3d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                              -(real(cws(i))*dcos(arg3)) )
        arg3=+kvs(i)*dist+5d+0*kvg*dist-tpi/4d+0
      arg3=dmod(arg3,tpi)
        Dpire1m(i)=Dpire1m(i)+1/5d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                              +(real(cws(i))*dsin(arg3)) )
        Dpiim1m(i)=Dpiim1m(i)+1/5d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                              -(real(cws(i))*dcos(arg3)) )
        arg3=+kvs(i)*dist-5d+0*kvg*dist-tpi/4d+0
      arg3=dmod(arg3,tpi)
        Dpire1m(i)=Dpire1m(i)-1/5d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                              +(real(cws(i))*dsin(arg3)) )
        Dpiim1m(i)=Dpiim1m(i)-1/5d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                              -(real(cws(i))*dcos(arg3)) )
40      continue

```

Appendix C. Fortran code for second order nonlinear interactions

```

return
end

!- corhppi.f -----

subroutine corhppi

common/arhph/tprehp,tpimhp,tsrehp,tsimhp,tirehp,tiimhp,t2rehp,t2imhp,
      t3rehp,t3imhp
common/arDhp/Dprehp,Dpimhp,Dsrehp,Dsimhp,Direhp,Diimhp,D2rehp,D2imhp,
      D3rehp,D3imhp,Dsirehp,Dsiimhp,Dp2rehp,Dp2imhp,Ds3rehp,Ds3imhp,
Dpirehp,Dpiimhp,Dp3rehp,Dp3imhp
common/arkv/kvp,kvs,kvi,kv2,kv3,kvg
common/vdbl/det,dom,facp,facs,faci,fac2,fac3
common/vrea/dist

parameter(N=8191)          !has to be (2^n)-1

integer          i
double precision dist
double precision tprehp(N),tpimhp(N),tsrehp(N),tsimhp(N),tirehp(N),tiimhp(N),
t2rehp(N),t2imhp(N),t3rehp(N),t3imhp(N)
double precision Dprehp(N),Dpimhp(N),Dsrehp(N),Dsimhp(N),Direhp(N),Diimhp(N),
D2rehp(N),D2imhp(N),D3rehp(N),D3imhp(N),Dsirehp(N),Dsiimhp(N),Dp2rehp(N),
Dp2imhp(N),Ds3rehp(N),Ds3imhp(N),Dpirehp(N),Dpiimhp(N),Dp3rehp(N),Dp3imhp(N)
double precision kvp(N),kvs(N),kvi(N),kv2(N),kv3(N),kvg
double precision arg1,arg2,arg3,tpi
double precision det,dom,facp,facs,faci,fac2,fac3
      complex*16      cwp(N),cws(N),cwi(N)

tpi = 6.29318530718d+0

do 20 i=1,N,1          ! 1D arrays --> 2D arrays
      arg1=-kvp(i)*dist      ! always define as e^(-i phi)
      arg2=-kvi(i)*dist
      arg1=dmod(arg1,tpi)
      arg2=dmod(arg2,tpi)
      cwp(i)=cmplx( (tprehp(i)*dcos(arg1))-(tpimhp(i)*dsin(arg1)),
                    (tprehp(i)*dsin(arg1))+(tpimhp(i)*dcos(arg1)) )
      cwi(i)=cmplx( (tirehp(i)*dcos(arg2))-(tiimhp(i)*dsin(arg2)),
                    (tirehp(i)*dsin(arg2))+(tiimhp(i)*dcos(arg2)) )
20  continue

```

Appendix C. Fortran code for second order nonlinear interactions

```

call fft(cwp,N,-1)
call fft(cwi,N,-1)

do 30 i=1,N,1
  cws(i)=cmplx( real(cwi(i))*real(cwp(i)) + dimag(cwp(i))*dimag(cwi(i)),
              dimag(cwp(i))*real(cwi(i)) - real(cwp(i))*dimag(cwi(i)) )
30 continue

call fft(cws,N,1)

do 40 i=1,N,1          ! 2D arrays --> 1D arrays
  arg3=+kvs(i)*dist+kvg*dist-tpi/4d+0
arg3=dmod(arg3,tpi)
  Dpirehp(i)=facs*( (dimag(cws(i))*dcos(arg3))
                    +(real(cws(i))*dsin(arg3)) )
  Dpiimhp(i)=facs*( (dimag(cws(i))*dsin(arg3))
                    -(real(cws(i))*dcos(arg3)) )
  arg3=+kvs(i)*dist-kvg*dist-tpi/4d+0
arg3=dmod(arg3,tpi)
  Dpirehp(i)=Dpirehp(i)-facs*( (dimag(cws(i))*dcos(arg3))
                               +(real(cws(i))*dsin(arg3)) )
  Dpiimhp(i)=Dpiimhp(i)-facs*( (dimag(cws(i))*dsin(arg3))
                               -(real(cws(i))*dcos(arg3)) )
  arg3=+kvs(i)*dist+3d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3,tpi)
  Dpirehp(i)=Dpirehp(i)+1/3d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                       +(real(cws(i))*dsin(arg3)) )
  Dpiimhp(i)=Dpiimhp(i)+1/3d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                       -(real(cws(i))*dcos(arg3)) )
  arg3=+kvs(i)*dist-3d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3,tpi)
  Dpirehp(i)=Dpirehp(i)-1/3d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                       +(real(cws(i))*dsin(arg3)) )
  Dpiimhp(i)=Dpiimhp(i)-1/3d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                       -(real(cws(i))*dcos(arg3)) )
  arg3=+kvs(i)*dist+5d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3,tpi)
  Dpirehp(i)=Dpirehp(i)+1/5d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                       +(real(cws(i))*dsin(arg3)) )
  Dpiimhp(i)=Dpiimhp(i)+1/5d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                       -(real(cws(i))*dcos(arg3)) )
  arg3=+kvs(i)*dist-5d+0*kvg*dist-tpi/4d+0

```

Appendix C. Fortran code for second order nonlinear interactions

```

    arg3=dmod(arg3, tpi)
    Dpirehp(i)=Dpirehp(i)-1/5d+0*facs*( (dimag(cws(i))*dcos(arg3))
        +(real(cws(i))*dsin(arg3)) )
    Dpiimhp(i)=Dpiimhp(i)-1/5d+0*facs*( (dimag(cws(i))*dsin(arg3))
        -(real(cws(i))*dcos(arg3)) )
40    continue

    return
    end

!- cor1ppi.f -----
    subroutine cor1ppi

    common/art1p/tpre1p, tpim1p, tsre1p, tsim1p, tire1p, tiim1p, t2re1p, t2im1p,
        t3re1p, t3im1p
    common/arD1p/Dpre1p, Dpim1p, Dsre1p, Dsim1p, Dire1p, Diim1p, D2re1p, D2im1p,
        D3re1p, D3im1p, Dsire1p, Dsiim1p, Dp2re1p, Dp2im1p, Ds3re1p, Ds3im1p,
Dpire1p, Dpiim1p, Dp3re1p, Dp3im1p
    common/arkv/kvp, kvs, kvi, kv2, kv3, kvg
    common/vdbl/det, dom, facp, facs, faci, fac2, fac3
    common/vrea/dist

    parameter(N=8191)          !has to be (2^n)-1

    integer                    i
    double precision dist
    double precision tpre1p(N), tpim1p(N), tsre1p(N), tsim1p(N), tire1p(N), tiim1p(N),
t2re1p(N), t2im1p(N), t3re1p(N), t3im1p(N)
    double precision Dpre1p(N), Dpim1p(N), Dsre1p(N), Dsim1p(N), Dire1p(N), Diim1p(N),
D2re1p(N), D2im1p(N), D3re1p(N), D3im1p(N), Dsire1p(N), Dsiim1p(N), Dp2re1p(N),
Dp2im1p(N), Ds3re1p(N), Ds3im1p(N), Dpire1p(N), Dpiim1p(N), Dp3re1p(N), Dp3im1p(N)
    double precision kvp(N), kvs(N), kvi(N), kv2(N), kv3(N), kvg
    double precision arg1, arg2, arg3, tpi
    double precision det, dom, facp, facs, faci, fac2, fac3
        complex*16          cwp(N), cws(N), cwi(N)

    tpi = 6.29318530718d+0

    do 20 i=1,N,1              ! 1D arrays --> 2D arrays
        arg1=-kvp(i)*dist      ! always define as e^(-i phi)
        arg2=-kvi(i)*dist
        arg1=dmod(arg1, tpi)

```

Appendix C. Fortran code for second order nonlinear interactions

```

arg2=dmod(arg2, tpi)
  cwp(i)=cmplx( (tpre1p(i)*dcos(arg1))-(tpim1p(i)*dsin(arg1)),
               (tpre1p(i)*dsin(arg1))+(tpim1p(i)*dcos(arg1)) )
  cwi(i)=cmplx( (tire1p(i)*dcos(arg2))-(tiim1p(i)*dsin(arg2)),
               (tire1p(i)*dsin(arg2))+(tiim1p(i)*dcos(arg2)) )
20  continue

  call fft(cwp,N,-1)
  call fft(cwi,N,-1)

  do 30 i=1,N,1
    cws(i)=cmplx( real(cwi(i))*real(cwp(i)) + dimag(cwp(i))*dimag(cwi(i)),
                 dimag(cwp(i))*real(cwi(i)) - real(cwp(i))*dimag(cwi(i)) )
30  continue

  call fft(cws,N,1)

  do 40 i=1,N,1          ! 2D arrays --> 1D arrays
    arg3=+kvs(i)*dist+kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
    Dpire1p(i)=facs*( (dimag(cws(i))*dcos(arg3))
                     +(real(cws(i))*dsin(arg3)) )
    Dpiim1p(i)=facs*( (dimag(cws(i))*dsin(arg3))
                     -(real(cws(i))*dcos(arg3)) )
    arg3=+kvs(i)*dist-kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
    Dpire1p(i)=Dpire1p(i)-facs*( (dimag(cws(i))*dcos(arg3))
                                 +(real(cws(i))*dsin(arg3)) )
    Dpiim1p(i)=Dpiim1p(i)-facs*( (dimag(cws(i))*dsin(arg3))
                                 -(real(cws(i))*dcos(arg3)) )
    arg3=+kvs(i)*dist+3d+0*kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
    Dpire1p(i)=Dpire1p(i)+1/3d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                         +(real(cws(i))*dsin(arg3)) )
    Dpiim1p(i)=Dpiim1p(i)+1/3d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                         -(real(cws(i))*dcos(arg3)) )
    arg3=+kvs(i)*dist-3d+0*kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
    Dpire1p(i)=Dpire1p(i)-1/3d+0*facs*( (dimag(cws(i))*dcos(arg3))
                                         +(real(cws(i))*dsin(arg3)) )
    Dpiim1p(i)=Dpiim1p(i)-1/3d+0*facs*( (dimag(cws(i))*dsin(arg3))
                                         -(real(cws(i))*dcos(arg3)) )
    arg3=+kvs(i)*dist+5d+0*kvg*dist-tpi/4d+0

```

Appendix C. Fortran code for second order nonlinear interactions

```

arg3=dmod(arg3, tpi)
  Dpire1p(i)=Dpire1p(i)+1/5d+0*facs*( (dimag(cws(i))*dcos(arg3))
    +(real(cws(i))*dsin(arg3)) )
  Dpiim1p(i)=Dpiim1p(i)+1/5d+0*facs*( (dimag(cws(i))*dsin(arg3))
    -(real(cws(i))*dcos(arg3)) )
  arg3=+kvs(i)*dist-5d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3, tpi)
  Dpire1p(i)=Dpire1p(i)-1/5d+0*facs*( (dimag(cws(i))*dcos(arg3))
    +(real(cws(i))*dsin(arg3)) )
  Dpiim1p(i)=Dpiim1p(i)-1/5d+0*facs*( (dimag(cws(i))*dsin(arg3))
    -(real(cws(i))*dcos(arg3)) )
40  continue

  return
  end

!- cor0si.f -----

subroutine cor0si

common/art0/tpre0, tpim0, tsre0, tsim0, tire0, tiim0, t2re0, t2im0,
  t3re0, t3im0
common/arD0/Dpre0, Dpim0, Dsre0, Dsim0, Dire0, Diim0, D2re0, D2im0,
  D3re0, D3im0, Dsire0, Dsiim0, Dp2re0, Dp2im0, Ds3re0, Ds3im0, Dpire0,
  Dpiim0, Dp3re0, Dp3im0
common/arkv/kvp, kvs, kvi, kv2, kv3, kvg
common/vdbl/det, dom, facp, facs, faci, fac2, fac3
common/vrea/dist

parameter(N=8191)          !has to be (2^n)-1

  integer          i
  double precision dist
double precision tpre0(N), tpim0(N), tsre0(N), tsim0(N), tire0(N), tiim0(N),
t2re0(N), t2im0(N), t3re0(N), t3im0(N)
double precision Dpre0(N), Dpim0(N), Dsre0(N), Dsim0(N), Dire0(N), Diim0(N),
D2re0(N), D2im0(N), D3re0(N), D3im0(N), Dsire0(N), Dsiim0(N), Dp2re0(N),
Dp2im0(N), Ds3re0(N), Ds3im0(N), Dpire0(N), Dpiim0(N), Dp3re0(N), Dp3im0(N)
double precision kvp(N), kvs(N), kvi(N), kv2(N), kv3(N), kvg
double precision arg1, arg2, arg3, tpi
double precision det, dom, facp, facs, faci, fac2, fac3
  complex*16      cwp(N), cws(N), cwi(N)

```


Appendix C. Fortran code for second order nonlinear interactions

```

tpi = 6.29318530718d+0

!   print *, 'cor0si'

      do 20 i=1,N,1           ! 1D arrays --> 2D arrays
         arg1=-kvs(i)*dist    ! always define as e-i phi
         arg2=-kvi(i)*dist
      arg1=dmod(arg1, tpi)
      arg2=dmod(arg2, tpi)
         cws(i)=cmplx( (tsre0(i)*dcos(arg1))-(tsim0(i)*dsin(arg1)),
                      (tsre0(i)*dsin(arg1))+(tsim0(i)*dcos(arg1)) )
         cwi(i)=cmplx( (tire0(i)*dcos(arg2))-(tiim0(i)*dsin(arg2)),
                      (tire0(i)*dsin(arg2))+(tiim0(i)*dcos(arg2)) )
20    continue

      call fft(cws,N,-1)
      call fft(cwi,N,-1)

      do 30 i=1,N,1
         cwp(i)=cmplx( real(cws(i))*real(cwi(i))- dimag(cws(i))*dimag(cwi(i)),
                      real(cws(i))*dimag(cwi(i)) + dimag(cws(i))*real(cwi(i)) )
30    continue

      call fft(cwp,N,1)

      do 40 i=1,N,1           ! 2D arrays --> 1D arrays
         arg3=+kvp(i)*dist+kvg*dist-tpi/4d+0
      arg3=dmod(arg3, tpi)
         Dsire0(i)=facp*( (dimag(cwp(i))*dcos(arg3))
                          +(real(cwp(i))*dsin(arg3)) )
         Dsiim0(i)=facp*( (dimag(cwp(i))*dsin(arg3))
                          -(real(cwp(i))*dcos(arg3)) )
         arg3=+kvp(i)*dist-kvg*dist-tpi/4d+0
      arg3=dmod(arg3, tpi)
         Dsire0(i)=Dsire0(i)-facp*( (dimag(cwp(i))*dcos(arg3))
                                     +(real(cwp(i))*dsin(arg3)) )
         Dsiim0(i)=Dsiim0(i)-facp*( (dimag(cwp(i))*dsin(arg3))
                                     -(real(cwp(i))*dcos(arg3)) )
         arg3=+kvp(i)*dist+3d+0*kvg*dist-tpi/4d+0
      arg3=dmod(arg3, tpi)
         Dsire0(i)=Dsire0(i)+1/3d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                                             +(real(cwp(i))*dsin(arg3)) )
         Dsiim0(i)=Dsiim0(i)+1/3d+0*facp*( (dimag(cwp(i))*dsin(arg3))

```

Appendix C. Fortran code for second order nonlinear interactions

```

                -(real(cwp(i))*dcos(arg3)) )
        arg3=+kvp(i)*dist-3d+0*kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
        Dsire0(i)=Dsire0(i)-1/3d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                +(real(cwp(i))*dsin(arg3)) )
        Dsiim0(i)=Dsiim0(i)-1/3d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                -(real(cwp(i))*dcos(arg3)) )
        arg3=+kvp(i)*dist+5d+0*kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
        Dsire0(i)=Dsire0(i)+1/5d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                +(real(cwp(i))*dsin(arg3)) )
        Dsiim0(i)=Dsiim0(i)+1/5d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                -(real(cwp(i))*dcos(arg3)) )
        arg3=+kvp(i)*dist-5d+0*kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
        Dsire0(i)=Dsire0(i)-1/5d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                +(real(cwp(i))*dsin(arg3)) )
        Dsiim0(i)=Dsiim0(i)-1/5d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                -(real(cwp(i))*dcos(arg3)) )
40      continue

      return
      end

!- cor1msi.f -----

      subroutine cor1msi

      common/art1m/tpre1m, tpim1m, tsre1m, tsim1m, tire1m, tiim1m, t2re1m, t2im1m,
                t3re1m, t3im1m
      common/arD1m/Dpre1m, Dpim1m, Dsre1m, Dsim1m, Dire1m, Diim1m, D2re1m, D2im1m,
                D3re1m, D3im1m, Dsire1m, Dsiim1m, Dp2re1m, Dp2im1m, Ds3re1m, Ds3im1m,
Dpire1m, Dpiim1m, Dp3re1m, Dp3im1m
      common/arkv/kvp, kvs, kvi, kv2, kv3, kvg
      common/vdbl/det, dom, facp, facs, faci, fac2, fac3
      common/vrea/dist

      parameter(N=8191)          !has to be (2^n)-1

      integer          i
      double precision dist
      double precision tpre1m(N), tpim1m(N), tsre1m(N), tsim1m(N), tire1m(N), tiim1m(N),
t2re1m(N), t2im1m(N), t3re1m(N), t3im1m(N)

```

Appendix C. Fortran code for second order nonlinear interactions

```

double precision Dpre1m(N),Dpim1m(N),Dsre1m(N),Dsim1m(N),Dire1m(N),Diim1m(N),
D2re1m(N),D2im1m(N),D3re1m(N),D3im1m(N),Dsire1m(N),Dsiim1m(N),Dp2re1m(N),
Dp2im1m(N),Ds3re1m(N),Ds3im1m(N),Dpire1m(N),Dpiim1m(N),Dp3re1m(N),Dp3im1m(N)
double precision kvp(N),kvs(N),kvi(N),kv2(N),kv3(N),kvg
double precision arg1,arg2,arg3,tpi
double precision det,dom,facp,facs,faci,fac2,fac3
      complex*16      cwp(N),cws(N),cwi(N)

tpi = 6.29318530718d+0

      do 20 i=1,N,1          ! 1D arrays --> 2D arrays
      arg1=-kvs(i)*dist      ! always define as e^(-i phi)
      arg2=-kvi(i)*dist
      arg1=dmod(arg1,tpi)
      arg2=dmod(arg2,tpi)
      cws(i)=cplx( (tsre1m(i)*dcos(arg1))-(tsim1m(i)*dsin(arg1)),
                  (tsre1m(i)*dsin(arg1))+(tsim1m(i)*dcos(arg1)) )
      cwi(i)=cplx( (tire1m(i)*dcos(arg2))-(tiim1m(i)*dsin(arg2)),
                  (tire1m(i)*dsin(arg2))+(tiim1m(i)*dcos(arg2)) )
20      continue

      call fft(cws,N,-1)
      call fft(cwi,N,-1)

      do 30 i=1,N,1
      cwp(i)=cplx( real(cws(i))*real(cwi(i))- dimag(cws(i))*dimag(cwi(i)),
                  real(cws(i))*dimag(cwi(i)) + dimag(cws(i))*real(cwi(i)) )
30      continue

      call fft(cwp,N,1)

      do 40 i=1,N,1          ! 2D arrays --> 1D arrays
      arg3=+kvp(i)*dist+kvg*dist-tpi/4d+0
      arg3=dmod(arg3,tpi)
      Dsire1m(i)=facp*( (dimag(cwp(i))*dcos(arg3))
                      +(real(cwp(i))*dsin(arg3)) )
      Dsiim1m(i)=facp*( (dimag(cwp(i))*dsin(arg3))
                      -(real(cwp(i))*dcos(arg3)) )
      arg3=+kvp(i)*dist-kvg*dist-tpi/4d+0
      arg3=dmod(arg3,tpi)
      Dsire1m(i)=Dsire1m(i)-facp*( (dimag(cwp(i))*dcos(arg3))
                      +(real(cwp(i))*dsin(arg3)) )
      Dsiim1m(i)=Dsiim1m(i)-facp*( (dimag(cwp(i))*dsin(arg3))

```

Appendix C. Fortran code for second order nonlinear interactions

```

                -(real(cwp(i))*dcos(arg3)) )
        arg3=+kvp(i)*dist+3d+0*kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
        Dsire1m(i)=Dsire1m(i)+1/3d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                +(real(cwp(i))*dsin(arg3)) )
        Dsiim1m(i)=Dsiim1m(i)+1/3d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                -(real(cwp(i))*dcos(arg3)) )
        arg3=+kvp(i)*dist-3d+0*kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
        Dsire1m(i)=Dsire1m(i)-1/3d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                +(real(cwp(i))*dsin(arg3)) )
        Dsiim1m(i)=Dsiim1m(i)-1/3d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                -(real(cwp(i))*dcos(arg3)) )
        arg3=+kvp(i)*dist+5d+0*kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
        Dsire1m(i)=Dsire1m(i)+1/5d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                +(real(cwp(i))*dsin(arg3)) )
        Dsiim1m(i)=Dsiim1m(i)+1/5d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                -(real(cwp(i))*dcos(arg3)) )
        arg3=+kvp(i)*dist-5d+0*kvg*dist-tpi/4d+0
    arg3=dmod(arg3, tpi)
        Dsire1m(i)=Dsire1m(i)-1/5d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                +(real(cwp(i))*dsin(arg3)) )
        Dsiim1m(i)=Dsiim1m(i)-1/5d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                -(real(cwp(i))*dcos(arg3)) )
40      continue

      return
      end

!- corhpsi.f -----

      subroutine corhpsi

      common/arthp/tprehp, tpimhp, tsrehp, tsimhp, tirehp, tiimhp, t2rehp, t2imhp,
            t3rehp, t3imhp
      common/arDhp/Dprehp, Dpimhp, Dsrehp, Dsimhp, Direhp, Diimhp, D2rehp, D2imhp,
            D3rehp, D3imhp, Dsirehp, Dsiimhp, Dp2rehp, Dp2imhp, Ds3rehp, Ds3imhp,
Dpirehp, Dpiimhp, Dp3rehp, Dp3imhp
      common/arkv/kvp, kvs, kvi, kv2, kv3, kvg
      common/vdbl/det, dom, facp, facs, faci, fac2, fac3
      common/vrea/dist

```

Appendix C. Fortran code for second order nonlinear interactions

```

parameter(N=8191)          !has to be (2^n)-1

integer                    i
double precision dist
double precision tprehp(N),tpimhp(N),tsrehp(N),tsimhp(N),tirehp(N),tiimhp(N),
t2rehp(N),t2imhp(N),t3rehp(N),t3imhp(N)
double precision Dprehp(N),Dpimhp(N),Dsrehp(N),Dsimhp(N),Direhp(N),Diimhp(N),
D2rehp(N),D2imhp(N),D3rehp(N),D3imhp(N),Dsirehp(N),Dsiimhp(N),Dp2rehp(N),
Dp2imhp(N),Ds3rehp(N),Ds3imhp(N),Dpirehp(N),Dpiimhp(N),Dp3rehp(N),Dp3imhp(N)
double precision kvp(N),kvs(N),kvi(N),kv2(N),kv3(N),kvg
double precision arg1,arg2,arg3,tpi
double precision det,dom,facp,facs,faci,fac2,fac3
complex*16                cwp(N),cws(N),cwi(N)

tpi = 6.29318530718d+0

do 20 i=1,N,1              ! 1D arrays --> 2D arrays
  arg1=-kvs(i)*dist        ! always define as e^(-i phi)
  arg2=-kvi(i)*dist
  arg1=dmod(arg1,tpi)
  arg2=dmod(arg2,tpi)
  cws(i)=cmplx( (tsrehp(i)*dcos(arg1))-(tsimhp(i)*dsin(arg1)),
               (tsrehp(i)*dsin(arg1))+(tsimhp(i)*dcos(arg1)) )
  cwi(i)=cmplx( (tirehp(i)*dcos(arg2))-(tiimhp(i)*dsin(arg2)),
               (tirehp(i)*dsin(arg2))+(tiimhp(i)*dcos(arg2)) )
20 continue

call fft(cws,N,-1)
call fft(cwi,N,-1)

do 30 i=1,N,1
  cwp(i)=cmplx( real(cws(i))*real(cwi(i))- dimag(cws(i))*dimag(cwi(i)),
               real(cws(i))*dimag(cwi(i)) + dimag(cws(i))*real(cwi(i)) )
30 continue

call fft(cwp,N,1)

do 40 i=1,N,1              ! 2D arrays --> 1D arrays
  arg3=+kvp(i)*dist+kvg*dist-tpi/4d+0
  arg3=dmod(arg3,tpi)
  Dsirehp(i)=facp*( (dimag(cwp(i))*dcos(arg3))
                   +(real(cwp(i))*dsin(arg3)) )
  Dsiimhp(i)=facp*( (dimag(cwp(i))*dsin(arg3))

```

Appendix C. Fortran code for second order nonlinear interactions

```

                -(real(cwp(i))*dcos(arg3)) )
        arg3=+kvp(i)*dist-kvg*dist-tpi/4d+0
        arg3=dmod(arg3, tpi)
        Dsirehp(i)=Dsirehp(i)-facp*( (dimag(cwp(i))*dcos(arg3))
                +(real(cwp(i))*dsin(arg3)) )
        Dsiimhp(i)=Dsiimhp(i)-facp*( (dimag(cwp(i))*dsin(arg3))
                -(real(cwp(i))*dcos(arg3)) )
        arg3=+kvp(i)*dist+3d+0*kvg*dist-tpi/4d+0
        arg3=dmod(arg3, tpi)
        Dsirehp(i)=Dsirehp(i)+1/3d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                +(real(cwp(i))*dsin(arg3)) )
        Dsiimhp(i)=Dsiimhp(i)+1/3d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                -(real(cwp(i))*dcos(arg3)) )
        arg3=+kvp(i)*dist-3d+0*kvg*dist-tpi/4d+0
        arg3=dmod(arg3, tpi)
        Dsirehp(i)=Dsirehp(i)-1/3d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                +(real(cwp(i))*dsin(arg3)) )
        Dsiimhp(i)=Dsiimhp(i)-1/3d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                -(real(cwp(i))*dcos(arg3)) )
        arg3=+kvp(i)*dist+5d+0*kvg*dist-tpi/4d+0
        arg3=dmod(arg3, tpi)
        Dsirehp(i)=Dsirehp(i)+1/5d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                +(real(cwp(i))*dsin(arg3)) )
        Dsiimhp(i)=Dsiimhp(i)+1/5d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                -(real(cwp(i))*dcos(arg3)) )
        arg3=+kvp(i)*dist-5d+0*kvg*dist-tpi/4d+0
        arg3=dmod(arg3, tpi)
        Dsirehp(i)=Dsirehp(i)-1/5d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                +(real(cwp(i))*dsin(arg3)) )
        Dsiimhp(i)=Dsiimhp(i)-1/5d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                -(real(cwp(i))*dcos(arg3)) )
40      continue

      return
      end

!- cor1psi.f -----

      subroutine cor1psi

      common/art1p/tpre1p, tpim1p, tsre1p, tsim1p, tire1p, tiim1p, t2re1p, t2im1p,
                t3re1p, t3im1p
      common/arD1p/Dpre1p, Dpim1p, Dsre1p, Dsim1p, Dire1p, Diim1p, D2re1p, D2im1p,

```

Appendix C. Fortran code for second order nonlinear interactions

```

          D3re1p,D3im1p,Dsire1p,Dsiim1p,Dp2re1p,Dp2im1p,Ds3re1p,Ds3im1p,
Dpire1p,Dpiim1p,Dp3re1p,Dp3im1p
          common/arkv/kvp,kvs,kvi,kv2,kv3,kvg
          common/vdbl/det,dom,facp,facs,faci,fac2,fac3
          common/vrea/dist

          parameter(N=8191)           !has to be (2^n)-1

          integer          i
          double precision dist
          double precision tpre1p(N),tpim1p(N),tsre1p(N),tsim1p(N),tire1p(N),tiim1p(N),
t2re1p(N),t2im1p(N),t3re1p(N),t3im1p(N)
          double precision Dpre1p(N),Dpim1p(N),Dsre1p(N),Dsim1p(N),Dire1p(N),Diim1p(N),
D2re1p(N),D2im1p(N),D3re1p(N),D3im1p(N),Dsire1p(N),Dsiim1p(N),Dp2re1p(N),
Dp2im1p(N),Ds3re1p(N),Ds3im1p(N),Dpire1p(N),Dpiim1p(N),Dp3re1p(N),Dp3im1p(N)
          double precision kvp(N),kvs(N),kvi(N),kv2(N),kv3(N),kvg
          double precision arg1,arg2,arg3,tpi
          double precision det,dom,facp,facs,faci,fac2,fac3
          complex*16      cwp(N),cws(N),cwi(N)

          tpi = 6.29318530718d+0

          do 20 i=1,N,1           ! 1D arrays --> 2D arrays
              arg1=-kvs(i)*dist    ! always define as e^(-i phi)
              arg2=-kvi(i)*dist
              arg1=dmod(arg1,tpi)
              arg2=dmod(arg2,tpi)
              cws(i)=cmplx( (tsre1p(i)*dcos(arg1))-(tsim1p(i)*dsin(arg1)),
                           (tsre1p(i)*dsin(arg1))+(tsim1p(i)*dcos(arg1)) )
              cwi(i)=cmplx( (tire1p(i)*dcos(arg2))-(tiim1p(i)*dsin(arg2)),
                           (tire1p(i)*dsin(arg2))+(tiim1p(i)*dcos(arg2)) )
20          continue

          call fft(cws,N,-1)
          call fft(cwi,N,-1)

          do 30 i=1,N,1
              cwp(i)=cmplx( real(cws(i))*real(cwi(i))- dimag(cws(i))*dimag(cwi(i)),
                           real(cws(i))*dimag(cwi(i)) + dimag(cws(i))*real(cwi(i)) )
30          continue

          call fft(cwp,N,1)

```

Appendix C. Fortran code for second order nonlinear interactions

```

do 40 i=1,N,1          ! 2D arrays --> 1D arrays
  arg3=+kvp(i)*dist+kvg*dist-tpi/4d+0
arg3=dmod(arg3,tpi)
  Dsire1p(i)=facp*( (dimag(cwp(i))*dcos(arg3))
                    +(real(cwp(i))*dsin(arg3)) )
  Dsiim1p(i)=facp*( (dimag(cwp(i))*dsin(arg3))
                    -(real(cwp(i))*dcos(arg3)) )
  arg3=+kvp(i)*dist-kvg*dist-tpi/4d+0
arg3=dmod(arg3,tpi)
  Dsire1p(i)=Dsire1p(i)-facp*( (dimag(cwp(i))*dcos(arg3))
                               +(real(cwp(i))*dsin(arg3)) )
  Dsiim1p(i)=Dsiim1p(i)-facp*( (dimag(cwp(i))*dsin(arg3))
                               -(real(cwp(i))*dcos(arg3)) )
  arg3=+kvp(i)*dist+3d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3,tpi)
  Dsire1p(i)=Dsire1p(i)+1/3d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                                       +(real(cwp(i))*dsin(arg3)) )
  Dsiim1p(i)=Dsiim1p(i)+1/3d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                                       -(real(cwp(i))*dcos(arg3)) )
  arg3=+kvp(i)*dist-3d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3,tpi)
  Dsire1p(i)=Dsire1p(i)-1/3d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                                       +(real(cwp(i))*dsin(arg3)) )
  Dsiim1p(i)=Dsiim1p(i)-1/3d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                                       -(real(cwp(i))*dcos(arg3)) )
  arg3=+kvp(i)*dist+5d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3,tpi)
  Dsire1p(i)=Dsire1p(i)+1/5d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                                       +(real(cwp(i))*dsin(arg3)) )
  Dsiim1p(i)=Dsiim1p(i)+1/5d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                                       -(real(cwp(i))*dcos(arg3)) )
  arg3=+kvp(i)*dist-5d+0*kvg*dist-tpi/4d+0
arg3=dmod(arg3,tpi)
  Dsire1p(i)=Dsire1p(i)-1/5d+0*facp*( (dimag(cwp(i))*dcos(arg3))
                                       +(real(cwp(i))*dsin(arg3)) )
  Dsiim1p(i)=Dsiim1p(i)-1/5d+0*facp*( (dimag(cwp(i))*dsin(arg3))
                                       -(real(cwp(i))*dcos(arg3)) )
40  continue

return
end

```