LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY
– LIGO –
CALIFORNIA INSTITUTE OF TECHNOLOGY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Document Type **LIGO-T010160-00-R** **10/15/01**

Simulations of Advanced LIGO: Comparisons between Twiddle and E2E

Richard George, Alan Weinstein

Distribution of this draft:

This is an internal working note
Of the LIGO Project

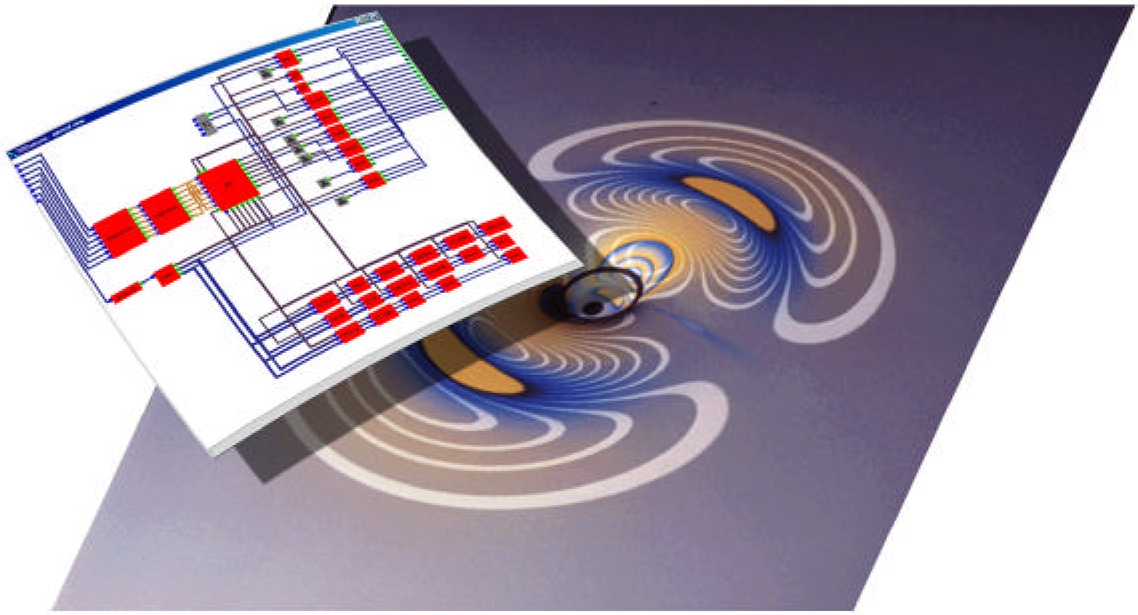
California Institute of Technology
LIGO Project – MS 51-33
Pasadena, CA 91125
Phone (626)395-2129
Fax (626)304-9834
E-mail: info@ligo.caltech.edu

Massachusetts Institute of Technology
LIGO Project – MS 20B-145
Cambridge, MA 01239
Phone (617)253-4824
Fax (617)253-7014
E-mail: info@ligo.mit.edu

WWW: http://www.ligo.caltech.edu

Simulations of Advanced Ligo: Comparisons between Twiddle and E2E

CalTech Surf Programme, Summer 2001
Student: Richard George
Mentor: Alan Weinstein



1. Abstract

The position of mirrors in the Advanced LIGO interferometer must be controlled to sub-nanometer precision in order to achieve full sensitivity to gravity waves. The sensing of the mirror positions via variations on the Pound-Drever[1] Demodulation technique require correct demodulation phases to separate the various signals present in the output light.

I compare two simulations of Advanced LIGO that operate in the frequency and time domains respectively, finding the error signals and demodulation phases that they predict, and examining the consistency of these two different approaches.

2. Introduction

LIGO is an experiment to detect gravitational waves that are predicted to exist by General Relativity. Gravity waves generate strains in the space through which they travel perpendicular to the direction of propagation. LIGO is in essence a sophisticated strain gauge that uses interferometery to detect Gravity waves by measuring the distance between a set of freely suspended mirrors.

Gravity waves that produce strains at the level of 1 part in 10^{-21} are predicted to be generated by coalescing binary stars with a frequency of approximately 3 events a year[2]. Many sources of noise exist that would swamp this motion were it not for careful isolation of the mirrors from their external environment (see figure 2.1). Initial LIGO uses seismic isolation stacks to filter vibrations caused by human and geological activity, and suspends the mirrors as pendulums, freely swinging from a support to provide further isolation.

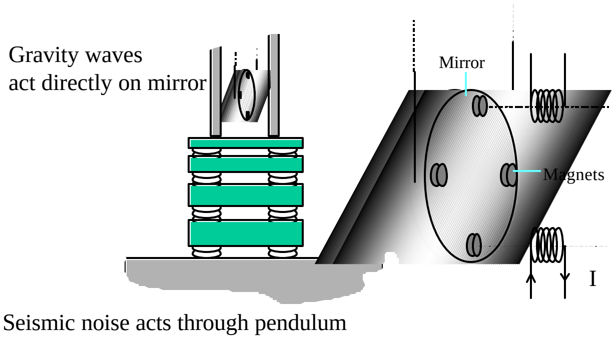


Figure 2.1: Seismic Isolation

This arrangement ensures that any terrestrial source of motion is greatly attenuated; leaving gravity waves as the only influence that can directly affect the mirrors position.

In order to have maximum robustness against photon shot-noise, the intensity of light in the interferometer must be at a maximum, e.g. the light must resonate. In order to control the positions of the mirrors, a feedback system is required.

In LIGO and Advanced LIGO, the sensing of the mirror positions is performed optically. When the carrier is resonant, the intensity of the light is at a maximum, and therefore the intensity of the light is at best a quadratic function of the displacement. To provide a control signal from the small displacements, a signal linearly related to the displacement from resonance must be found.

A solution to this problem is to insert two sidebands onto the main carrier frequency, via phase modulation. There are now three frequencies present in the cavity, the resonant carrier, and the two sidebands that are close to resonance. Comparing the relative amplitudes and phases of the two sidebands to the carrier provides a signal that is linearly dependent on the displacement of the mirror from perfect resonance. Comparing the relative intensities of the carrier and sidebands via demodulation has the added benefit that the derived error signal rejects power fluctuations in the unmodulated laser beam.

A control system is employed in LIGO in order to hold the mirrors locked on the resonant condition. The aim of the E2E[3] and Twiddle[4] simulation packages is to help design these control systems.

The mirrors of LIGO are free to swing on their pendulum supports, but in order to achieve resonance and maintain lock, each mirror must be held stationary to within a fraction of a wavelength. To this end, each test mass is fitted with four magnets, attached to the edges of the circular mirror. Currents fed to coils nearby can adjust the position of the mirror, and control signals are derived from the light escaping from the interferometer, in order to steer the mirrors to their optimal positions, and hold the interferometer in lock.

Advanced LIGO is the proposed upgrade to the current equipment at the Hanford and Livingston sites. The optical configuration and controls for Advanced LIGO will be prototyped at the 40-meter lab here at Caltech.

At low frequencies, seismic and all other environmental noise is dominant, but around 12Hz the isolation causes the noise to plummet, and at high frequencies, where gravity waves are expected to lie, Advanced LIGO aims to be limited only by fundamental noise sources. (Thermal motion of the test masses, and photon shot-noise.)

The ultimate limit on the sensitivity of the interferometer (above approximately 250Hz) is the photon shot noise present in the signals. This comes from the statistical nature of low intensity light beams. When single photons carry an appreciable fraction of the total signal power, (as will occur for small mirror displacements), fluctuations in the photon arrival rate become evident as noise in the observed signal. Advanced LIGO solves this problem with two methods, by having a greater optical power present in the interferometer (600kW versus 5kW of initial LIGO), and through a Signal Recycling Cavity.

The Signal Recycling Cavity allows greater control over the power distribution in the interferometer, and by changing the length of the cavity the shot noise can be tailored, increasing sensitivity at high frequencies. Figures 2.2 and 2.3 show the zero-detuned and tuned interferometer as modeled by Twiddle, one of the tools I used.

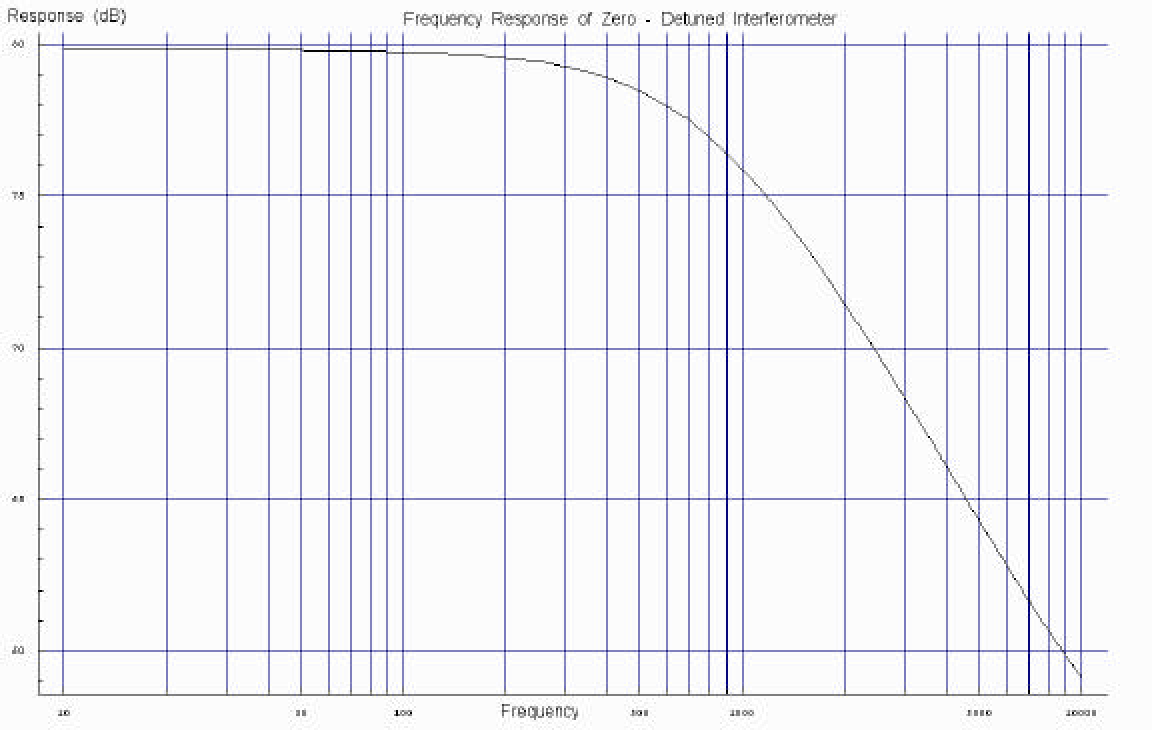


Figure 2.2: Frequency Response of Advanced LIGO without detuning.

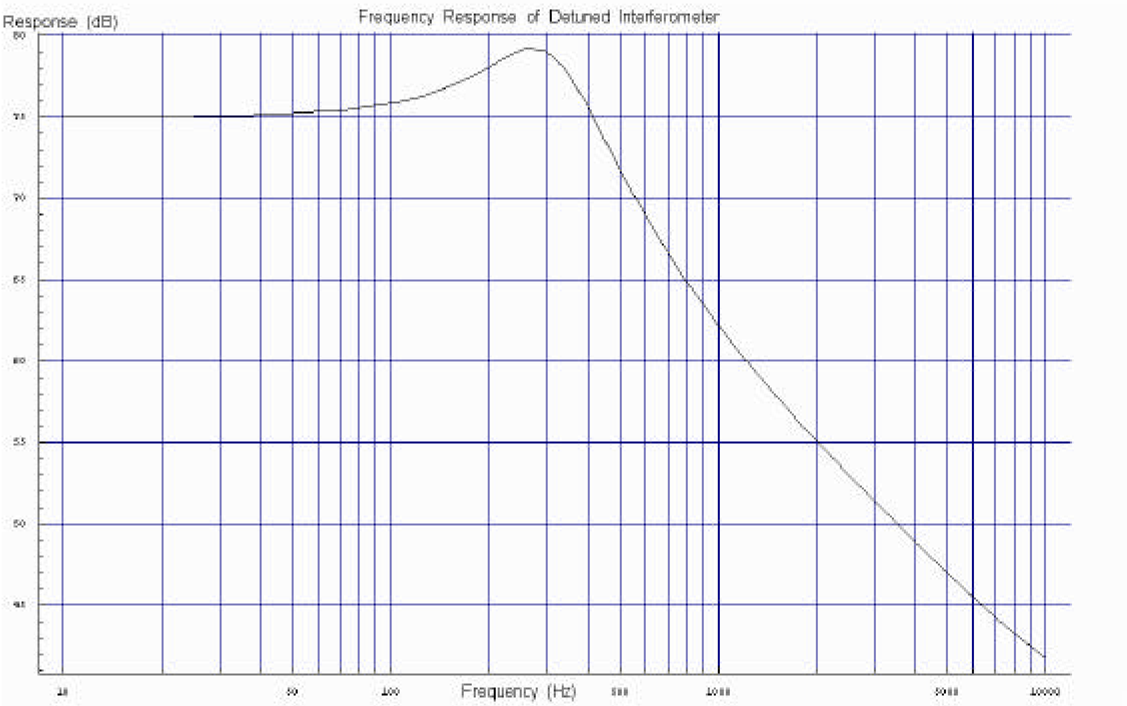


Figure 2.3: Detuned Interferometer with peak response at 100Hz.

The increased optical power in Advanced LIGO will potentially cause unacceptable heating of the mirrors. Heating is more pronounced when the beam has to travel through the substrate of the mirror before being reflected. The signal-recycling cavity can be adjusted to reduce the power traveling through the input test mass substrates while maintaining high optical sensitivity at frequencies above 250Hz, though this aspect was not investigated.

The length sensing and control system of the LIGO interferometer relies on being able to determine the lengths of all optical cavities, and thus the positions of the six test-mass mirrors with respect to the each other. (In my study, I chose to use the beam splitter as the origin of the coordinate system.). The carrier is phase-modulated by a Pockels cell, and this generates five frequencies, each with different responses to mirror motion, to perform length sensing. By choosing the optical path lengths appropriately, different frequencies are made to resonate in different sections of the interferometer.

The carrier is resonant in the arms of the interferometer while neither of the sidebands enter there appreciably, so that the 9MHz and 180MHz sidebands should not be sensitive to the positions of the end test masses. Beats between these "reference" frequencies and the carrier are therefore sensitive to relative movements of the input and end test masses, and allow one to sense their relative positions close to resonance.

Figure 2.3 shows the locations of the beams, in strong color for resonant and faint for non-resonant. There are three points from which light leaves the interferometer. The symmetric port, the asymmetric port, and the pick-off. There are three points in the intererometer at which the signals can be extracted.

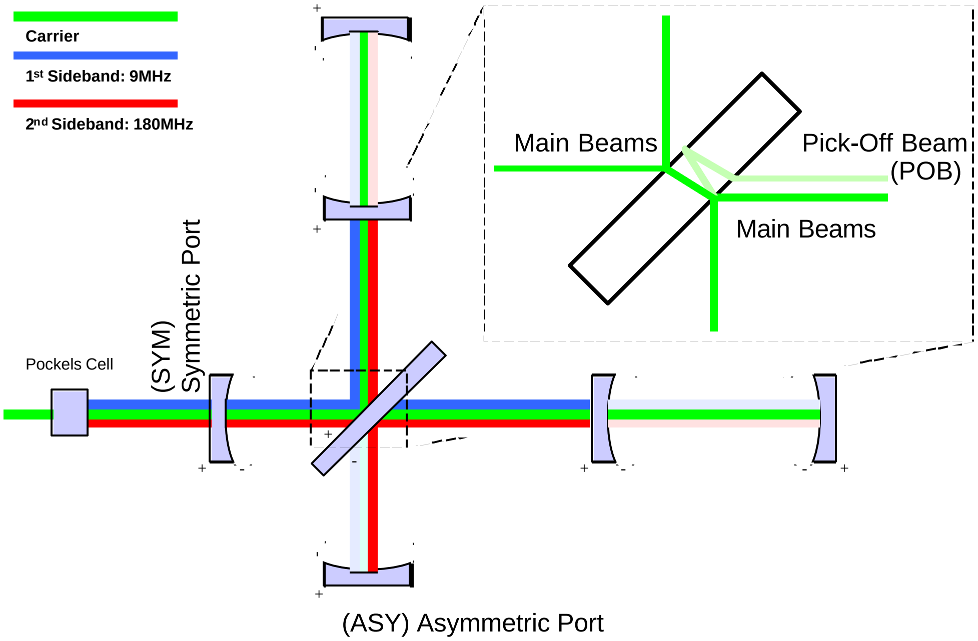
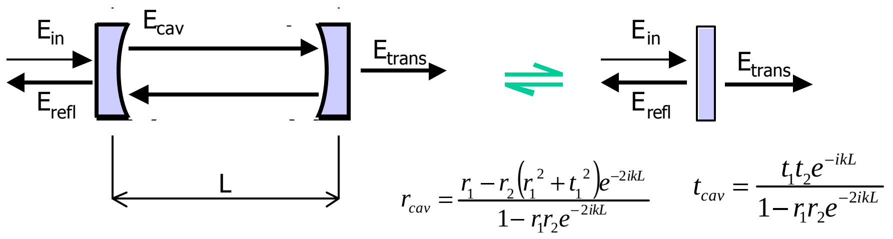


Figure 2.3: Locations of fields in the Dual-Recycling Interferometer.

The ports are termed Symmetric, Anti-symmetric are named after the signals they detect. The light intensity and demodulated signals at the Anti-symmetric port is primarily sensitive to movements of the interferometer mirrors in opposite directions, i.e., the L- degree of freedom, while light from the Symmetric port is sensitive to movement of L+. The pick-off detects the small fraction of light which is reflected out of the main beam path.



$$r_{cav} = \frac{r_1 - r_2\left(r_1^2 + t_1^2\right)e^{-2ikL}}{1 - r_1 r_2 e^{-2ikL}} \qquad t_{cav} = \frac{t_1 t_2 e^{-ikL}}{1 - r_1 r_2 e^{-2ikL}}$$

Figure 2.4: A Fabry-Perot cavity can be treated as a single mirror with complex reflectivity and transmission coefficients.

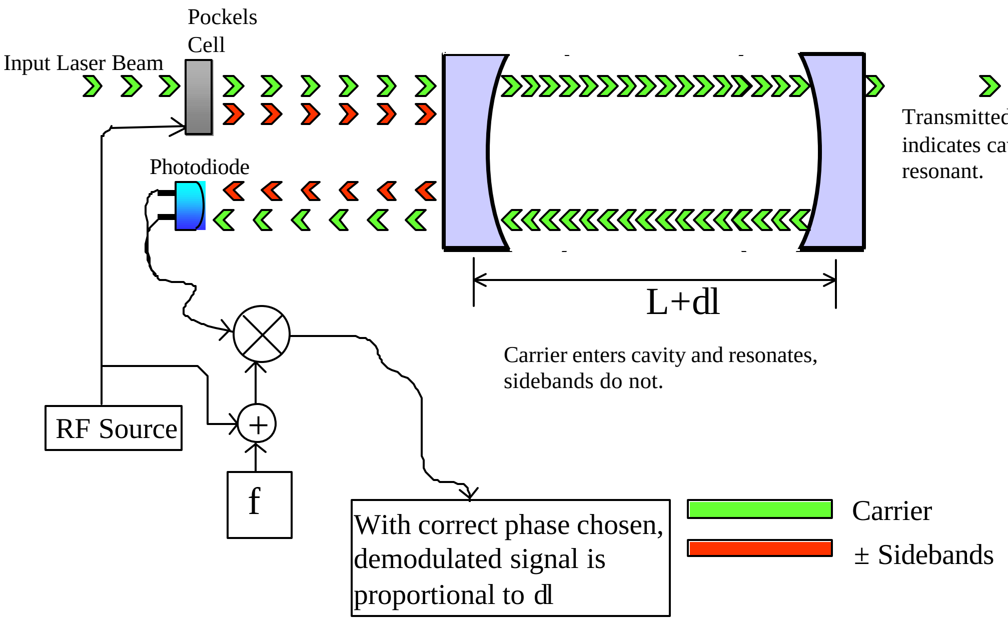

Pockels Cell
Input Laser Beam
Photodiode
Transmitted light indicates cavity is resonant.
L+dl
Carrier enters cavity and resonates, sidebands do not.
RF Source
+
f
With correct phase chosen, demodulated signal is proportional to dl
🟩 Carrier
🟥 ± Sidebands

Figure 2.5: Pound Drever Locking. The relative phases of a carrier and two sidebands are compared via electronic hetrodyne detection.

Pound Drever Locking works on the basis that close to resonance, the phase of the reflected carrier light is a strong function of the length error δl, while the sidebands are relatively insensitive to the length of the cavity. Comparing the relative phases of the carrier and the sidebands via demodulation gives a signal proportional to the error, that can then be used for control.

3. Methods and Materials

The three tools I used for this investigation were E2E, Twiddle and Matlab. E2E and Twiddle are both packages developed at Caltech to simulate the behavior of interferometers. Twiddle is a Mathematica model that finds the frequency domain transfer functions of the interferometer from first principles, solving analytically the simultaneous equations for the fields at each mirror.

E2E is a time-domain simulation package capable of emulating the entire Advanced LIGO apparatus. E2E is capable of simulating the effects of transients, shot noise, seismic noise, angular misalignments and lock acquisition, which are central to the design of Advanced LIGO.

Twiddle can answer questions about the DC properties, demodulation phases, error signals, transfer function frequency dependence and error signals faster than E2E, but can only usefully simulate the behavior of the interferometer in lock, or for small excursions from a known operating point. My task was to adjust the Dual Recycling Models in Twiddle and E2E to agree with each other, as a way of lending confidence to their predictions.

The first stage of this task was to ensure that Twiddle and E2E predicted the same behavior for the interferometer after the laser is turned on at steady state, with everything perfectly aligned. This required setting up both Twiddle and E2E to have the same model of the interferometer.

LIGO is sensitive to minute displacements of the mirrors; the Fabry-Perot Arms are sensitive to displacements of a fraction of the optical wavelength.

Twiddle's model is represented by a series of equations, each representing an optical element in the interferometer, while E2E's model is constructed by drawing a representation of the interferometer using a GUI. Boxes in E2E represent physical components in the apparatus. Lines between the boxes represent the flow of signals in the interferometer. Figure 3.1 shows the configuration of the optics in E2E.

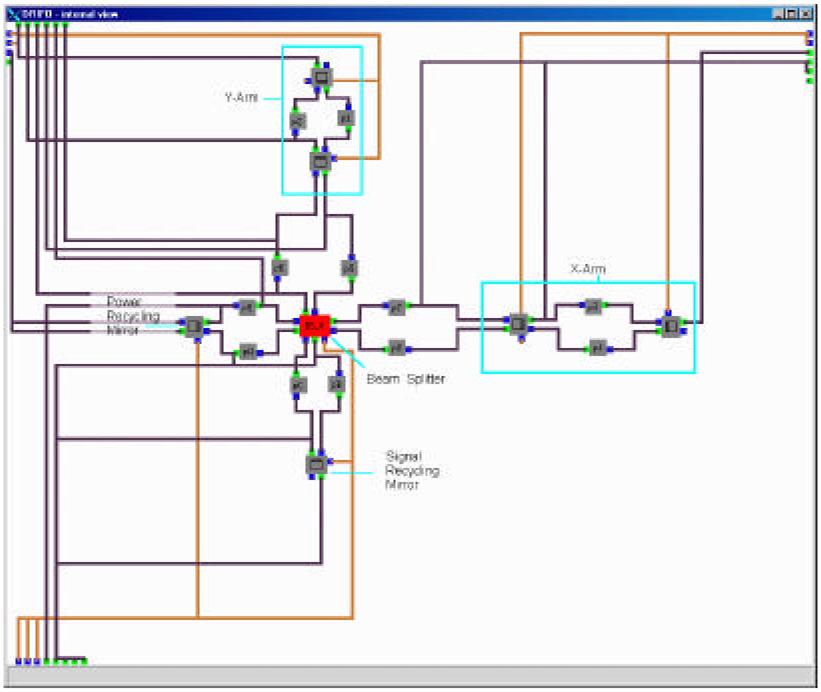
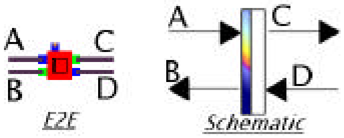


Figure 3.1: The Dual Recycling Interferometer in E2E, showing the Beam splitter, Arms and Recycling Cavities



$$B = -rA + tD$$
$$C = tA + rD$$

Figure 3.2: The Mirrors in E2E, and equations relating fields at the two surfaces

The Twiddle model can currently plot the expected transfer functions for the Dual Recycling LIGO, so was taken as the standard to compare E2E against. I transferred the lengths, reflectivity and losses of each component from Twiddle into E2E, and ran both models.

E2E takes some time to build up the fields in the simulated cavities, starting as it does from zero fields everywhere in the interferometer. In order to accelerate running E2E, I had to turn off the propagation delays that are present in the arms of the interferometer. With the propagation delays turned on, the simulator requires a time step of $6*10^{-10}$ seconds, while with delays turned off, any time step can be selected. Turning off the delays also means that the DC field levels build up rapidly in the simulation.

I began by adapting the Dual Recycling Model. I inserted a monitor box to investigate the optical power in the carrier and the two upper and lower sidebands at each point of relevance in the interferometer, and placed virtual photodiodes at each of these places in order to investigate beats between the different signals. Figures K and L show the location and contents of the monitors.

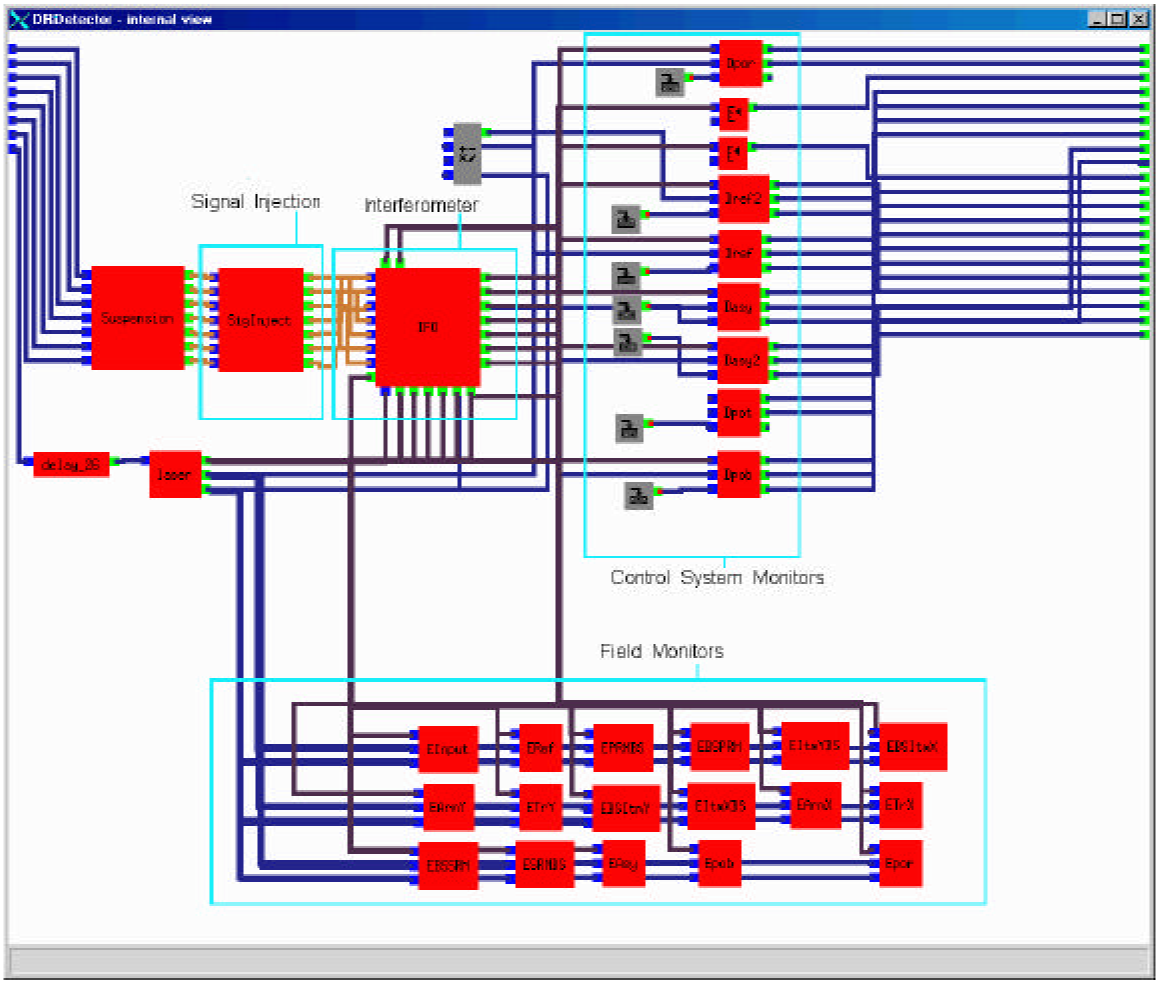
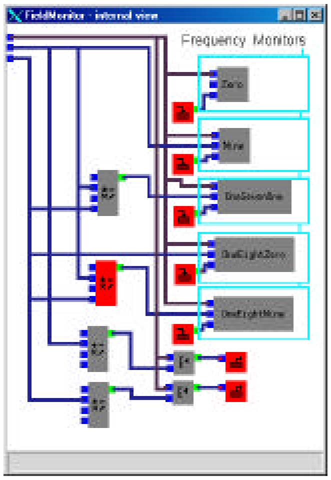
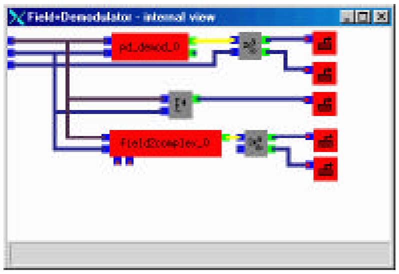


Figure 3.4: The DRDetector, showing the Signal Injecting mechanism, and the Field Monitors.



A Field Monitor



A Frequency Monitor

Figure 3.5: The boxes that monitor fields in E2E.

I altered Twiddle to write the DC values it predicted into a file, and used Matlab to examine the predictions of both E2E and Twiddle. After a few iterations of changing parameters in E2E and correcting other miscellaneous errors, E2E and Twiddle agreed well for the zero-detuned DC fields.

In the Zero detune case the upper and lower sidebands are identical, so that the signal at carrier+9MHz is the same as the signal at Carrier-9MHz. Detuning is interesting for Advanced LIGO, as it allows for shaping of the shot-noise curve. Twiddle shows how detuning the interferometer shapes the shot-noise curve. Using this it is possible to optimize the IFO shot-noise for maximum sensitivity to binary inspirial in the presence of other sources of displacement noise.

I then repeated the process for the Detuned interferometer, arranged so that more optical power was present in the upper sidebands. Making the detuned models agree with each other was complicated initially by the different units and sign conventions that Twiddle and E2E used for reflection coefficients and microscopic mirror displacement, though experimenting with different permutations revealed the correct choices.

With the DC fields correctly set up, I next tried to measure the response of the interferometer as the mirrors are moved away from their ideal in-tune positions, and monitored the demodulated signals, in order to compare Twiddle and E2E.

The photodiodes in the Advanced LIGO simulation see an intensity that oscillates rapidly with time, at RF frequencies corresponding to the modulation imposed on the input laser beam by the Pockels cells. Different frequencies of oscillation correspond to the 'beats' between different optical frequencies.

The phase of these beats relative to the reference that is injected into the Pockels cell can be adjusted in order to maximize response to a particular degree of freedom, or to provide an error signal that becomes zero when a mirror is aligned perfectly.

The reference phase is set to zero in E2E, and then adjustments are made in Matlab to bring the error signal to zero when there are no mirror displacements. This avoids having to re-run E2E in order to experiment with different demodulation phases, (a time consuming process).

4. Results and Discussion

The DC fields agree well in both the tuned and detuned cases, to within fractions of a percent for all field levels that are appreciably different from zero. This establishes that both E2E and Twiddle fundamentally agree with each other. (See appendix 1) for a complete listing of the fields at each port and demodulation frequency.

E2E and Twiddle do not agree well once the mirrors are swept away from their in-tune positions. Twiddle produces graphs that match the expected behavior of the interferometer, while E2E still has flaws in it's output. The mirrors are swept in the correct fashion, as is established by plotting the degree of freedom inputs and resulting mirror displacements, in appendix 3. E2E and Twiddle use different conventions for displacements and mirror orientation, and in fact the results indicate that E2E needs adapting to compensate for the differences of notation.

5. Conclusions

Further effort will be required in order to obtain agreement between E2E and Twiddle, The DC field agreement shows that the two modeling packages produce consistent results in appropriate conditions, and with a thorough debugging I am confident the demodulation phases will agree too.

6. Acknowledgements

I thank my mentor, Alan Weinstein for help and guidance with this task, and Lauren Stolper and Ken Libbrecht and the SURF office for overseeing the exchange program, and making this an enjoyable summer.

7. References

1 “Notes on the Pound-Drever-Hall Technique”, Eric Black,
April 98 **LIGO-T98045-00-D**

2 Fundamentals of Interferometeric Gravitational Wave Detectors, P.R. Saulson,
1984 World Scientific Publishing,

3 “E2E Primitive Reference Manual”, Biplab, Evans, Rahman, Yamamoto
LIGO-T0000047-01-E

4 “Twiddle (version 3.0)– A Program for analysing Interferometer Frequency Response”
LIGO-T990022-00-R

5 Signal Extraction and Control for an Interferometeric Gravitational Wave Detector
Thesis of Martin Regher, August 1995, **LIGO-P340002-00-I**

LIGO Publications are accessible on the web from
`http://admdbsrv.ligo.caltech.edu/dcc/`

Apendix 1: Model Parameters

Mirror Properties				
Abbreviation	*Name*	*Power Loss*	*Power Transmission*	*Power Reflectance*
BS	Beam Splitter	0.0003	0.49985	0.49985
ItmX / ItmT	Input Test Mass X	0.0000375	0.005	0.9949625
ItmY / ItmR	Input Test Mass Y	0.0000375	0.005	0.9949625
EtmX / EtmT	End Test Mass X	0.0000375	0.000015	0.9999475
EtmY / EtmR	End Test Mass Y	0.0000375	0.000015	0.9999475
PRM	Power Recycling Mirror	0.0000375	0.075	0.9249625
SRM	Signal Recycling Mirror	0.0000375	0.07	0.9299625
Cavity Lengths	*Length (meters)*			
BS-ItmX	4.89452			
BS-ItmY	4.47864			
ItmX-EtmX	3995.2			
ItmY-EtmY	3995.2			
PRM-BS	3.631			
SRM-BS	3.61906			
Wavelength	*(meters)*			
λ	$1.664*10^{-09}$			

Appendix 2: Detuned DC Fields

E2E *Laser Input and Power Recycling Cavity* **Twiddle**

Channel	**Power**	**Gain**	**State**
-2 Sideband	0.310164	1	-
-1 Sideband	0.310164	1	-
Carrier	123.7555	1	-
+1 Sideband	0.310164	1	-
+2 Sideband	0.310164	1	-

Channel	**Power**	**Gain**	**State**	**Error**
-2 Sideband	0.311714	1	-	0.00%
-1 Sideband	0.311714	1	-	0.00%
Carrier	123.7531	1	-	0.00%
+1 Sideband	0.311714	1	-	0.00%
+2 Sideband	0.311714	1	-	0.00%

Ref *From Symmetric Port*

Channel	**Power**	**Gain**	**State**
-2 Sideband	0.264762	0.853619	-
-1 Sideband	0.298007	0.960807	-
Carrier	0.162281	0.001311	-
+1 Sideband	0.29799	0.96075	-
+2 Sideband	0.000762	0.002457	-

Channel	**Power**	**Gain**	**State**	**Error**
-2 Sideband	0.266085	0.853619	-	0.00%
-1 Sideband	0.299497	0.960807	-	0.00%
Carrier	0.163045	0.001318	-	-0.47%
+1 Sideband	0.299479	0.96075	-	0.00%
+2 Sideband	0.000766	0.002457	-	0.00%

PRMBS *From Power Recycling Mirror towards Beam Splitter*

Channel	**Power**	**Gain**	**State**
-2 Sideband	0.59162	1.907442	Resonant
-1 Sideband	15.38931	49.61671	Resonant

Channel	**Power**	**Gain**	**State**	**Error**
-2 Sideband	0.594576	1.907442	Resonant	0.00%
-1 Sideband	15.46621	49.61671	Resonant	0.00%

Carrier	1643.353	13.27903	Resonant
+1 Sideband	15.41328	49.69397	Resonant
+2 Sideband	4.032609	13.00154	Resonant

Carrier	1643.604	13.28131	Resonant	-0.02%
+1 Sideband	15.49029	49.69397	Resonant	0.00%
+2 Sideband	4.05276	13.00154	Resonant	0.00%

BSPRM *From Beam Splitter towards Power Recycling Mirror*

Channel	Power	Gain	State
-2 Sideband	0.636988	2.053714	Resonant
-1 Sideband	15.40088	49.65401	Resonant
Carrier	1766.88	14.27719	Resonant
+1 Sideband	15.42486	49.73132	Resonant
+2 Sideband	4.341848	13.99856	Resonant

Channel	Power	Gain	State	Error
-2 Sideband	0.640171	2.053714	Resonant	0.00%
-1 Sideband	15.47784	49.65401	Resonant	0.00%
Carrier	1767.127	14.27945	Resonant	-0.02%
+1 Sideband	15.50194	49.73132	Resonant	0.00%
+2 Sideband	4.363544	13.99856	Resonant	0.00%

E2E *X Arm* **Twiddle**

BSItmX *From Beam Splitter towards Input Test Mass of X Arm*

Channel	Power	Gain	State
-2 Sideband	0.725454	2.338936	Resonant
-1 Sideband	7.679575	24.75973	Resonant
Carrier	883.1748	7.136452	Resonant
+1 Sideband	7.746293	24.97484	Resonant
+2 Sideband	4.193914	13.52161	Resonant

Channel	Power	Gain	State	Error
Sideband	0.505677	1.622248	Resonant	30.64%
Sideband	7.778319	24.9534	Resonant	-0.78%
Carrier	883.2986	7.137585	Resonant	-0.02%
Sideband	7.735378	24.81565	Resonant	0.64%
Sideband	4.201493	13.47869	Resonant	0.32%

ItmXBS *From Input Test Mass X towards Beam Splitter*

Channel	Power	Gain	State
-2 Sideband	0.503144	1.622187	Resonant
-1 Sideband	7.739351	24.95246	Resonant
Carrier	821.923	6.641509	Resonant
+1 Sideband	7.696625	24.8147	Resonant
+2 Sideband	4.180445	13.47818	Resonant

Channel	Power	Gain	State	Error
-2 Sideband	0.505658	1.622187	Resonant	0.00%
-1 Sideband	7.778023	24.95246	Resonant	0.00%
Carrier	822.0484	6.642646	Resonant	-0.02%
+1 Sideband	7.735084	24.8147	Resonant	0.00%
+2 Sideband	4.201334	13.47818	Resonant	0.00%

ArmX *From Input Test Mass of X Arm towards End Test Mass of X Arm*

Channel	Power	Gain	State
-2 Sideband	0.000885	0.002852	-
-1 Sideband	0.03908	0.125998	-
Carrier	680105.5	5495.559	Resonant
+1 Sideband	0.038864	0.125302	-
+2 Sideband	0.007349	0.023694	-

Channel	Power	Gain	State	Error
-2 Sideband	0.000889	0.002852	-	0.01%
-1 Sideband	0.039273	0.125991	-	0.01%
Carrier	680169.2	5496.177	Resonant	-0.01%
+1 Sideband	0.039056	0.125296	-	0.01%
+2 Sideband	0.007385	0.023693	-	0.01%

TrX *light transmitted through End Test Mass of X Arm*

Channel	Power	Gain	State
-2 Sideband	0	0	-
-1 Sideband	0.000001	0.000002	-
Carrier	10.20158	0.082433	-
+1 Sideband	0.000001	0.000002	-
+2 Sideband	0	0	-

Channel	Power	Gain	State	Error
-2 Sideband	0	0	-	0.00%
-1 Sideband	0.000001	0.000002	-	0.00%
Carrier	10.20307	0.082447	-	-0.02%
+1 Sideband	0.000001	0.000002	-	0.00%
+2 Sideband	0	0	-	0.00%

E2E *Y Arm* **Twiddle**

BSItmY *from Beam Splitter towards Input Test Mass of X Arm*

Channel	Power	Gain	State
-2 Sideband	0.503163	1.622248	Resonant
-1 Sideband	7.739644	24.9534	Resonant
Carrier	883.1748	7.136452	Resonant
+1 Sideband	7.696917	24.81565	Resonant
+2 Sideband	4.180603	13.47869	Resonant

Channel	Power	Gain	State	Error
-2 Sideband	0.729079	2.338936	Resonant	-44.18%
-1 Sideband	7.71795	24.75973	Resonant	0.78%
Carrier	883.2986	7.137585	Resonant	-0.02%
+1 Sideband	7.785	24.97484	Resonant	-0.64%
+2 Sideband	4.214871	13.52161	Resonant	-0.32%

ItmYBS *from Input Test Mass X towards Beam Splitter*

Channel	Power	Gain	State
-2 Sideband	0.725426	2.338848	Resonant
-1 Sideband	7.679284	24.7588	Resonant
Carrier	821.923	6.641509	Resonant
+1 Sideband	7.745999	24.97389	Resonant
+2 Sideband	4.193756	13.5211	Resonant

Channel	Power	Gain	State	Error
-2 Sideband	0.729051	2.338848	Resonant	0.00%
-1 Sideband	7.717657	24.7588	Resonant	0.00%
Carrier	822.0484	6.642646	Resonant	-0.02%
+1 Sideband	7.784705	24.97389	Resonant	0.00%
+2 Sideband	4.214712	13.5211	Resonant	0.00%

ArmY *from Input Test Mass of X Arm towards End Test Mass of X Arm*

Channel	Power	Gain	State
-2 Sideband	0.001275	0.004112	-
-1 Sideband	0.038777	0.12502	-
Carrier	680105.5	5495.559	Resonant
+1 Sideband	0.039114	0.126106	-
+2 Sideband	0.007372	0.02377	-

dle:

Channel	Power	Gain	State	Error
-2 Sideband	0.001282	0.004111	-	0.01%
-1 Sideband	0.038968	0.125013	-	0.01%
Carrier	680169.2	5496.177	Resonant	-0.01%
+1 Sideband	0.039307	0.1261	-	0.01%
+2 Sideband	0.007409	0.023768	-	0.01%

TrY *light transmitted through End Test Mass of X Arm*

Channel	Power	Gain	State
-2 Sideband	0	0	-
-1 Sideband	0.000001	0.000002	-
Carrier	10.20158	0.082433	-
+1 Sideband	0.000001	0.000002	-
+2 Sideband	0	0	-

Channel	Power	Gain	State	Error
-2 Sideband	0	0	-	0.00%
-1 Sideband	0.000001	0.000002	-	0.00%
Carrier	10.20307	0.082447	-	-0.02%
+1 Sideband	0.000001	0.000002	-	0.00%
+2 Sideband	0	0	-	0.00%

E2E *Signal Recycling Cavity* **Twiddle**

BSSRM *from Beam Splitter towards Signal Recycling Mirror*

Channel	Power	Gain	State
-2 Sideband	0.636582	2.052405	Resonant
-1 Sideband	0.024697	0.079624	-
Carrier	0	0	-
+1 Sideband	0.024715	0.079684	-
+2 Sideband	4.33908	13.98964	Resonant

Channel	Power	Gain	State	Error
-2 Sideband	0.639763	2.052405	Resonant	0.00%
-1 Sideband	0.02482	0.079624	-	0.00%
Carrier	0	0	-	0.00%
+1 Sideband	0.024839	0.079684	-	0.00%
+2 Sideband	4.360762	13.98964	Resonant	0.00%

SRMBS *from Beam Splitter towards Signal Recycling Mirror*

Channel	Power	Gain	State
-2 Sideband	0.591997	1.908659	Resonant
-1 Sideband	0.022967	0.074048	-
Carrier	0	0	-
+1 Sideband	0.022984	0.074104	-
+2 Sideband	4.035182	13.00984	Resonant

Channel	Power	Gain	State	Error
-2 Sideband	0.594955	1.908659	Resonant	0.00%
-1 Sideband	0.023082	0.074048	-	0.00%
Carrier	0	0	-	0.00%
+1 Sideband	0.023099	0.074104	-	0.00%
+2 Sideband	4.055345	13.00984	Resonant	0.00%

Asy *transmitted through Signal Recycling Mirror (Asymmetric Port)*

Channel	Power	Gain	State
-2 Sideband	0.044561	0.143668	-
-1 Sideband	0.001729	0.005574	-
Carrier	0	0	-
+1 Sideband	0.00173	0.005578	-
+2 Sideband	0.303736	0.979275	-

Channel	Power	Gain	State	Error
-2 Sideband	0.044783	0.143668	-	0.00%
-1 Sideband	0.001737	0.005574	-	0.00%
Carrier	0	0	-	0.00%
+1 Sideband	0.001739	0.005578	-	0.00%
+2 Sideband	0.305253	0.979275	-	0.00%

Appendix 3:

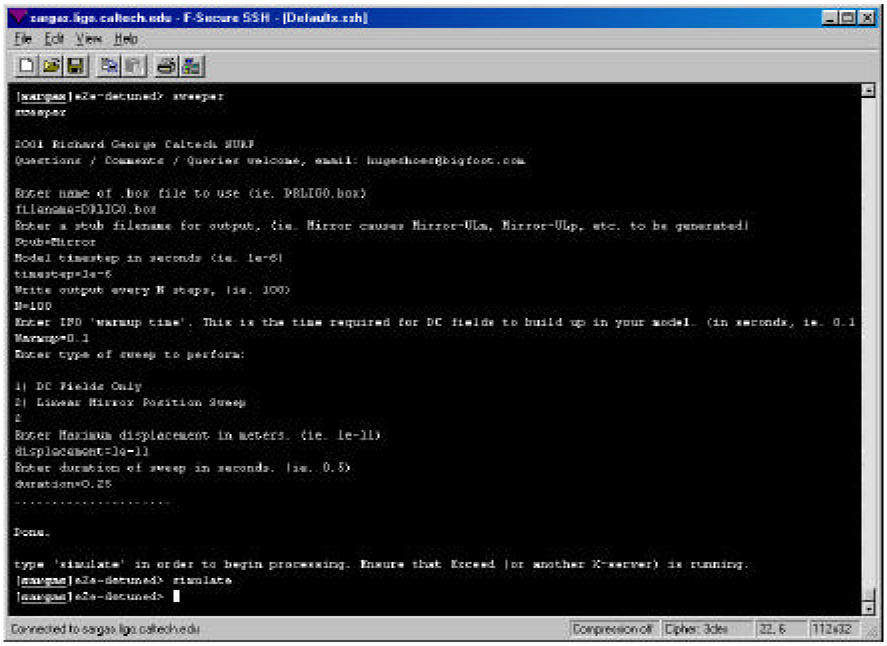
The immediately useful result of this project is the tools I have generated while performing the task. I document them here:

UNIX Commands:

The modeler must be fed with instructions that specify how to conduct the simulation. In order to perform a sweep of the interferometer's response, I found it useful to have a method of generating these configuration files automatically.

`'sweeper'` is a program written to automate performing sweeps with E2E. It takes a set of default settings, common to each run to be performed, and then prompts for the independent variable to be swept in the simulation. Sweeper can inject the same stimulus into each of the degrees of freedom of the interferometer, or sweep one degree of freedom through a range of frequencies. The sweeper program generates the .in and .par files required to perform the simulations, by combining user input and the file 'defaults.par'.

After `'sweeper'` is run, a short script called `'simulate'` is run to dispatch many copies of the modeler program simultaneously.


sargas.ligo.caltech.edu - F-Secure SSH - [Defaults.ssh]
File Edit View Help
Connected to sargas.ligo.caltech.edu
Compression off
Cipher: 3des
22, 6
112x32

```
[sargas]e2e-detuned> sweeper
sweeper

2001 Richard George Caltech SURF
Questions / Comments / Queries welcome, email: hugeshoes@bigfoot.com

Enter name of .box file to use (ie. DRLIGO.box)
filename=DRLIGO.box
Enter a stub filename for output. (ie. Mirror causes Mirror-ULm, Mirror-ULp, etc. to be generated)
Stub=Mirror
Model timestep in seconds (ie. 1e-6)
timestep=1e-6
Write output every N steps, (ie. 100)
N=100
Enter IFO 'warmup time'. This is the time required for DC fields to build up in your model. (in seconds, ie. 0.1
Warmup=0.1
Enter type of sweep to perform:

1) DC Fields Only
2) Linear Mirror Position Sweep
2
Enter Maximum displacement in meters. (ie. 1e-11)
displacement=1e-11
Enter duration of sweep in seconds. (ie. 0.5)
duration=0.25
--------------------

Done.

type 'simulate' in order to begin processing. Ensure that Exceed (or another X-server) is running.
[sargas]e2e-detuned> simulate
[sargas]e2e-detuned> █
```

Figure A3.1: Using `sweeper` to make a sweep of all channels through 10^{-11}m, first pausing for 0.1s to enable DC fields to build up in the interferometer.

Matlab Commands:

Matlab is the tool I used to analyze the simulation results. Typically, a simulation would involve sending the same signal into each of the input channels of the interferometer, and examining the response of the interferometer. `‘sweeper’` asks for a stub name, and then attaches a short descriptive tag onto the end of the stub name, i.e. if given a stub name of ‘Sweep’ and moving each degree of freedom, it generates data files named:

Sweep-Ulm[1].dat	L_m degree of freedom: The U is for ‘Uppercase L’
Sweep-ULp.dat	L_p degree of freedom
Sweep-lm.dat	l_m degree of freedom
Sweep-lp.dat	l_p degree of freedom
Sweep-ls.dat	l_s degree of freedom
Sweep-lprm.dat	l_{prm} degree of freedom
Sweep-DC.dat	DC response – a baseline with no injected signals

These data files are then loaded into Matlab, by typing:

```
fields=package_fields(‘Sweep’);
```

in the directory which contains the data-files. (NB. Performance of Matlab improves if the files are first copied from the Network disks onto the local hard drive). When this command completes, `fields` contains the injected control signals and the response of the interferometer, recorded against time.

fields.Lm	L_m response
fields.Lp	L_p response
fields.lm	l_m response
fields.lp	l_p response
fields.ls	l_s response
fields.lprm	l_{prm} response (move only Power Recycling mirror)
fields.dc	DC response (nothing moves)

Each field component can also be loaded individually, for example by calling:

```
fields.dc=package_field(‘Mirror-DC’);
```

This is useful when tuning up the DC fields between E2E and Twiddle, since sweeping the mirrors is not useful until DC agreement is established.

The records loaded by `package_field(s)` are a complete listing of injected stimuli and interferometer response, at every recorded time step.

[1] ‘ULm’ and ‘lm’ are used rather than ‘L-‘ and ‘l-‘ in order to keep output filenames distinct.

The simulation time is stored in `fields.Lm.time` and is a column vector that lists the time in seconds.

Signals determined by the user are stored in Inject, so that `fields.Lm.Inject` contains a list of the injected degrees of freedom as the L_m channel is swept:

```
fields.Lm.Inject.Lp
fields.Lm.Inject.Lm
fields.Lm.Inject.lp
fields.Lm.Inject.lm
fields.Lm.Inject.ls
fields.Lm.Inject.lprm
```

likewise, the mirror positions are recorded in

```
fields.Lm.ItmX
fields.Lm.ItmY
fields.Lm.EtmX
fields.Lm.EtmY
fields.Lm.Sig
fields.Lm.Prm
```

A an example, the following typed into Matlab plots the L_m degree of freedom against time, and the response of the End Test Masses.

```
figure
hold on
plot(fields.Lm.time,fields.Lm.Inject.Lm,’b’)
plot(fields.Lm.time,fields.Lm.EtmX,’r:’)
plot(fields.Lm.time,fields.Lm.EtmY,’g:’)
xlabel(‘time (seconds)’)
ylabel(‘displacement (meters)’)
title(‘Injected Signal and Mirror Response’)
legend(‘L_m,’EtmX’,’EtmY’)
```

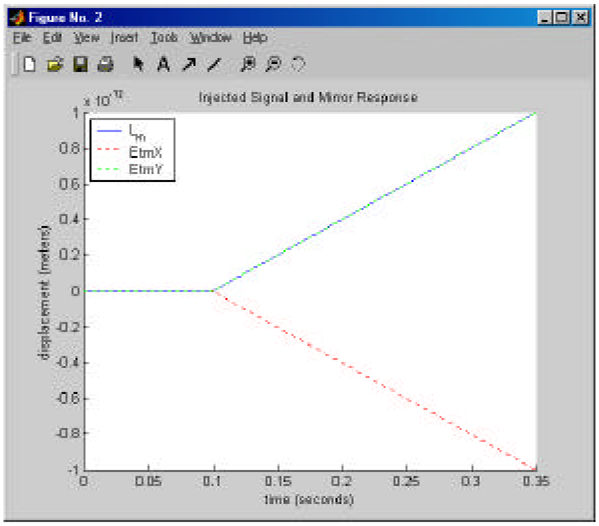


Figure A3.2: Input to Lm channel and response of End Test Mass Mirrors

Each degree of freedom can be inspected in turn in this way, or the commands

`show_all_dofs(fields)` or `show_dofs(fields,'Lm')`

can be used to inspect each degree of freedom. This is useful for debugging an E2E simulation, since it verifies that the inputs are as expected, and will reveal if some control system other than the injected signal is influencing the mirror positions.
Examining the Degrees of Freedom is achieved by:

The rest of the data loaded by `package_fields` and `package_field` is response to the injected signals. The fields are demodulated at each frequency of interest, 0MHz, 9MHz, 171MHz, 180MHz and 189MHz, and optionally the real and imaginary parts of the fields at each frequency are extracted as well.

Field	Position of field on interferometer

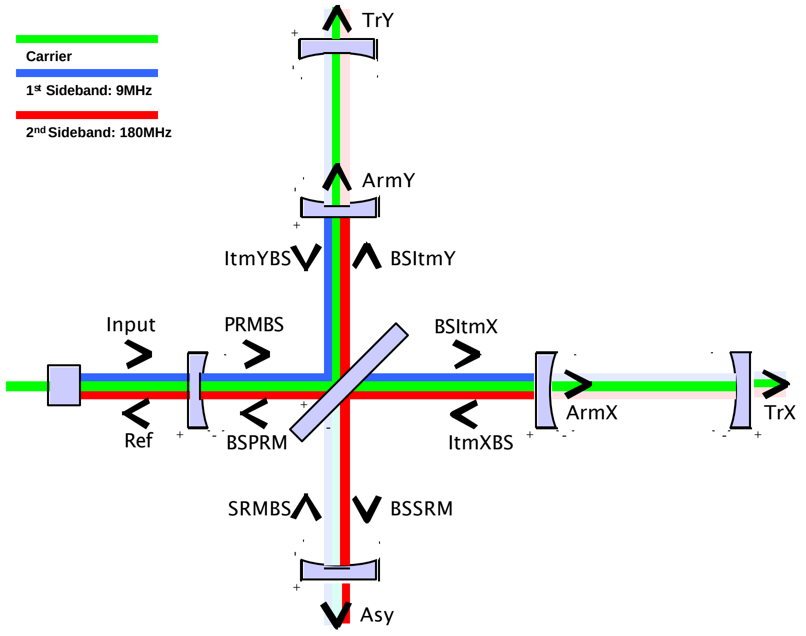
`fields.Lm.Input`
`fields.Lm.Ref`
`fields.Lm.BSPRM`
`fields.Lm.PRMBS`
`fields.Lm.Por`
`fields.Lm.Pob`
`fields.Lm.BSItmX`
`fields.Lm.ItmXBS`
`fields.Lm.ArmX`
`fields.Lm.TrX`
`fields.Lm.BSItmY`
`fields.Lm.ItmYBS`
`fields.Lm.ArmY`
`fields.Lm.TrY`
`fields.Lm.BSSRM`
`fields.Lm.SRMBS`
`fields.Lm.Asy`
	

Figure A3.3: Fields available for inspection with DRLIGO.

Each record in the fields extracted from the interferometer contains data about the fields detected there. There are two properties of interest in this structure,

```
fields.Lm.Asy.demod_freqs
fields.Lm.Asy.d(1..6)
```

`demod_freqs` is an array which lists the frequencies at which demodulation is performed, ie. 0,9,171,180,189 MHz and d(1) is the recorded demodulation at the n'th frequency, so that

```
fields.Lm.Asy.d(2).InPhase
fields.Lm.Asy.d(2).QuadPhase
```

list the In-phase and Quad-phase demodulated signals at demodulation frequency 2 (9MHz) from the Asymmetric port when L_m is excited.

```
fields.Lm.Asy.d(2).Power
```

lists the power present at 9MHz with time.

DC Fields

In order to rapidly compare E2E and Twiddle, it’s useful to load the results of both a Twiddle simulation and an E2E Simulation into Matlab, and analyse both at once. The output of Twiddle can be loaded into Matlab with the command:

```
twiddle=load_twiddle(‘filename’)
```

if filename is omitted, then a dialog box appears, asking you to select a file to load. Since Twiddle only simulates the steady-state conditions in the interferometer and deviations from them, the twiddle data-files are short.

```
compare_dc(fields.dc,twiddle)
```

The DC fields between E2E and Twiddle can be compared using the Matlab command

compare_dc(fields.dc,twiddle)

This generates a table which lists the predictions of E2E against Twiddle, along with a percentage difference between the two, defined as:

$$\frac{Value(E2E)-Value(Twiddle)}{Value(Twiddle)}*100\%$$

Appendix 4: Mirror response to input signals.

E2E is a complicated model, so in order to demonstrate that I understood and could control the mirrors in the system properly, I potted the positions of the mirrors against time.

One area of confusion that arose was the way in which Twiddle and E2E treat displacing the mirrors. Applying a positive displacement to the End Test Masses (ETMs) in Twiddle moves the mirrors away from the Beam Splitter, whilst a positive displacement of the ETMs in E2E moves them toward the Beam Splitter.

Plotting the injected signals along side the mirror displacements shows that the injected signals are decoded properly into mirror displacement.

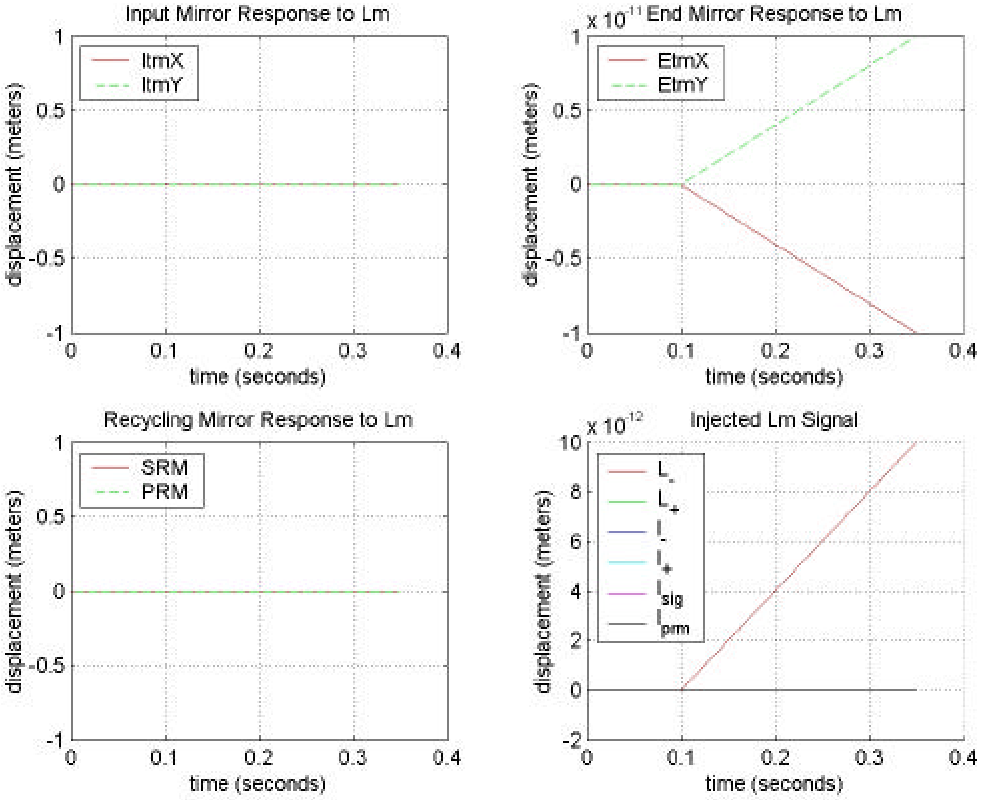


Figure A4.1: Response of Mirror Displacements to the L_- signal.

For instance, in the figure above, the L- degree of freedom is excited, by displacing the end test masses in opposite directions, whilst holding the Input Test masses and the Signal and Power Recycling mirrors fixed.

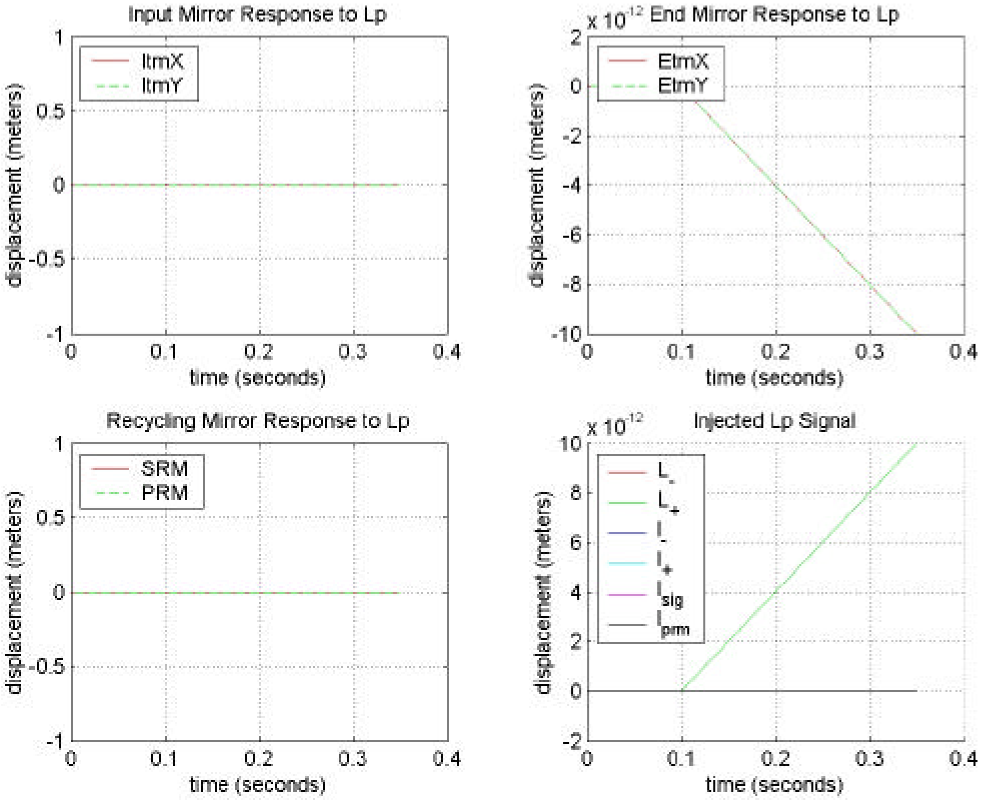


Figure A4.2: Response of Mirror Displacements to L_+ signal.

When the L+ degree of freedom is excited, the end test masses should move physically away from the Beam Splitter. In E2E a positive displacement advances the coated side of a mirror. The End Test Masses, and the Power and Signal Recycling Mirrors have their coated sides facing the Beam-Splitter, so to lengthen the cavities, it's necessary to send negative displacement signals to the End mirrors, as shown in the top right hand figure.

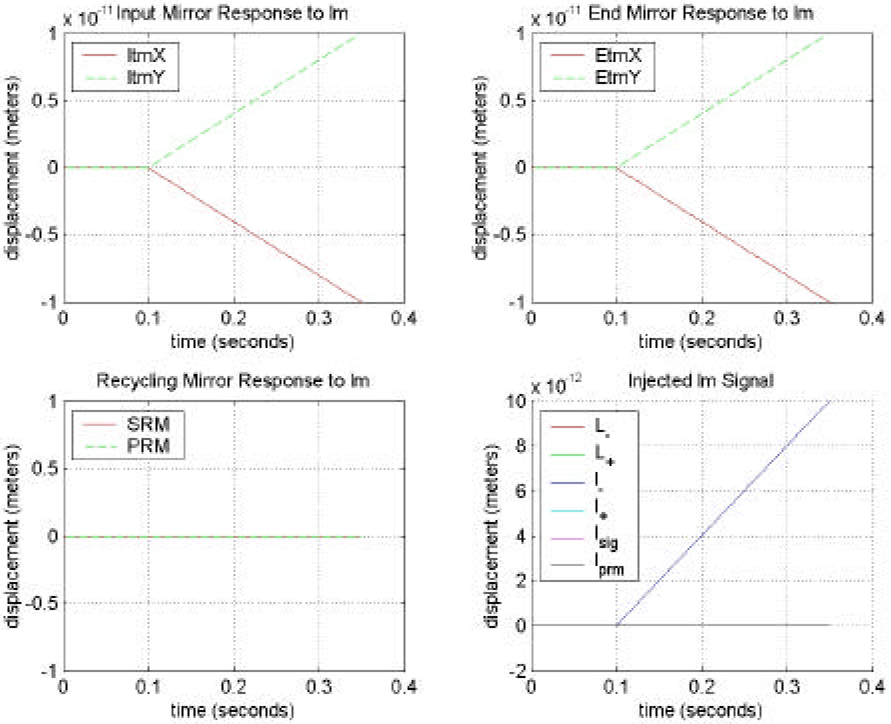


Figure A4.3: Response of Mirror Displacements to the l signal.

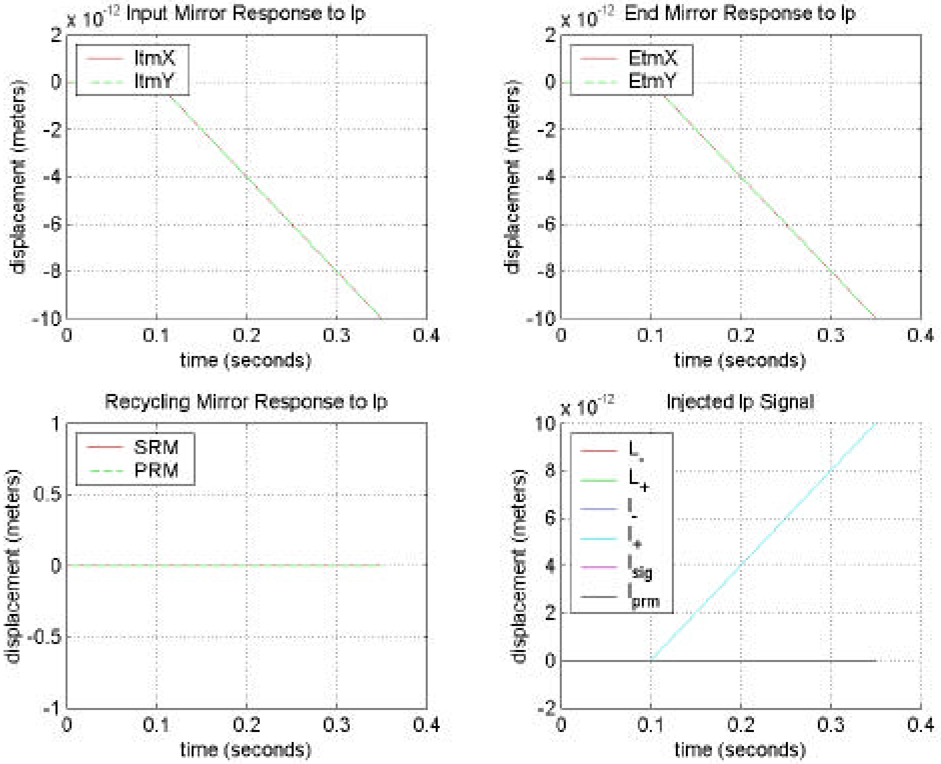


Figure A4.4: Response of the Mirror Displacements to the l_+ signal.

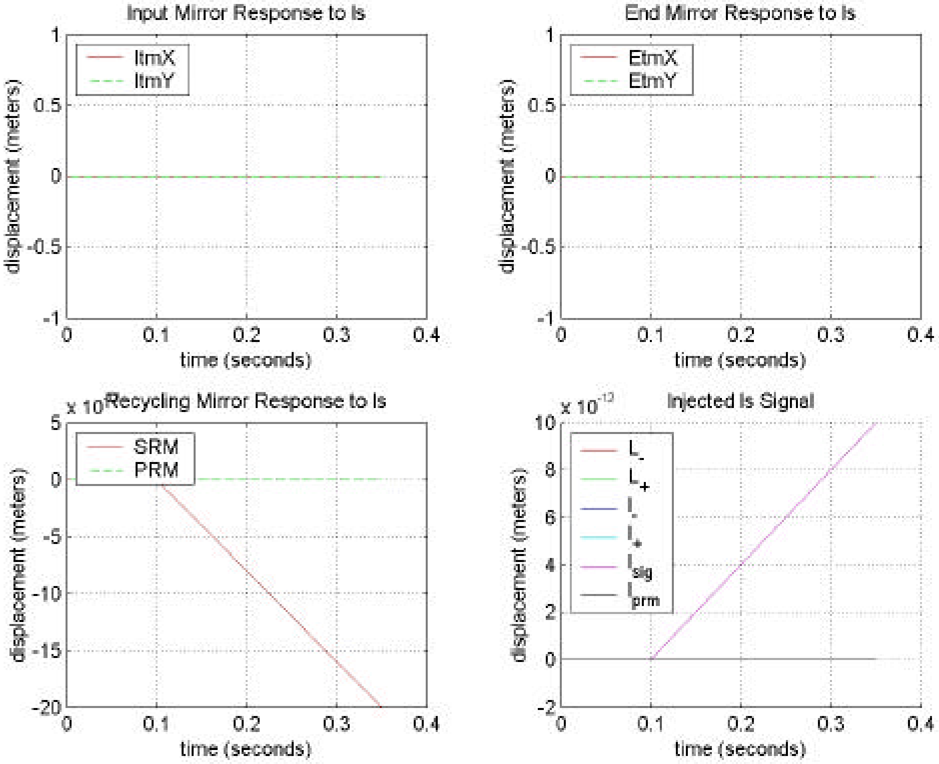


Figure A4.5: Response of the Mirror Displacements to the l_{sig} signal.

Appendix 5: Interferometer layout in E2E and Twiddle

Naming Conventions in E2E and Twiddle:

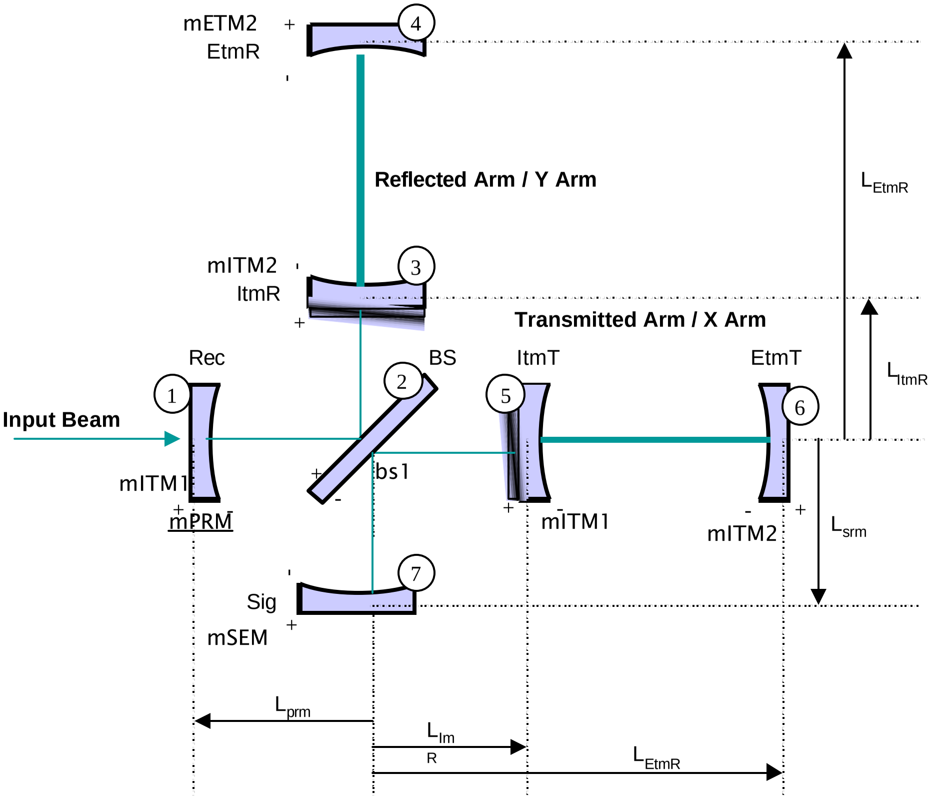
Twiddle in italics

E2E in plain script

Important Points:

- In E2E, mirrors move their coated side forwards when given a positive displacement, while in twiddle all mirrors move away from the laser source when given a positive displacement. This means that giving the mirrors a positive displacement in E2E moves the end test masses and Signal Recycling Mirror towards the beam-splitter, while in Twiddle, the same positive displacement moves mirrors away from the beam-splitter.

- Phase advances in E2E and Twiddle have opposite signs for the 'back-to-front' end mirrors for the same reason.



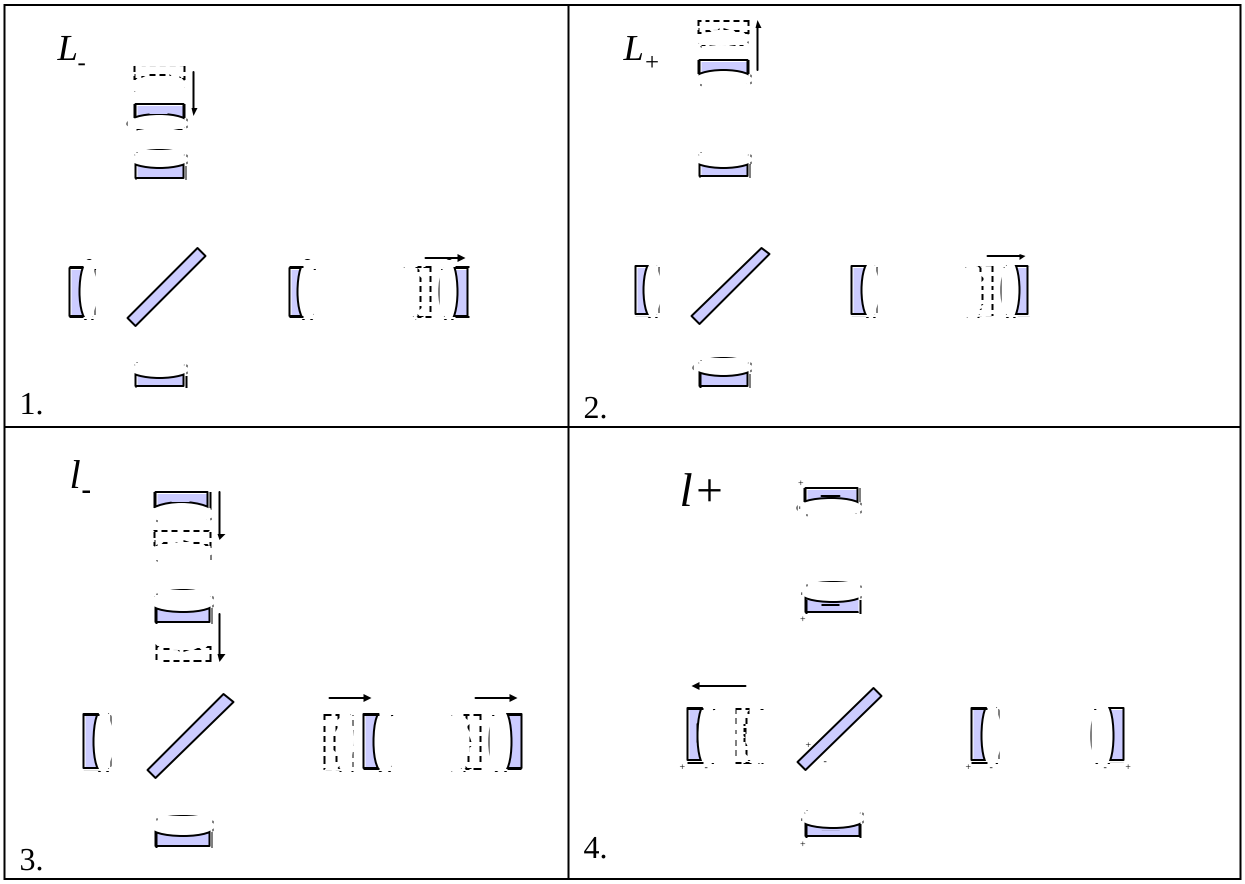
Appendix 5: Degrees of Freedom

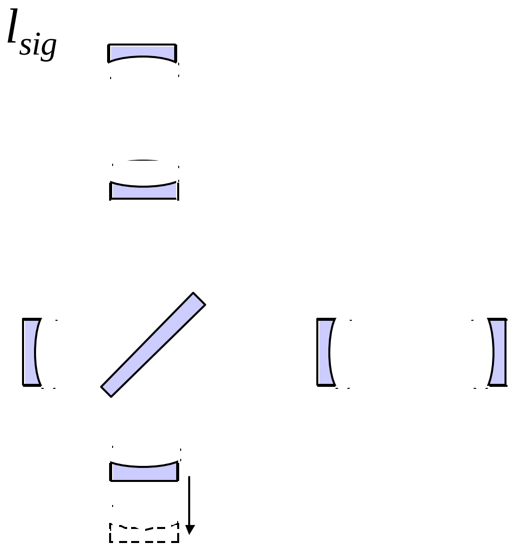
The degrees of freedom in the interferometer are chosen to correspond to signals that can be easily extracted by demodulation of the signals from the output ports.

NB. There is an error in the DRLIGO software regarding the channel labels.

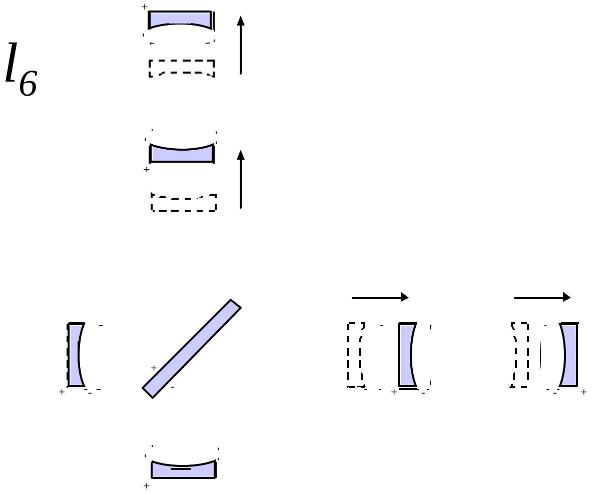
The “l+” channel is consistently referred to in the DRLIGO model as “lprm”. Wherever l+ is required for comparison with Twiddle, the lprm channel should be substituted.

The label “lp” in DRLIGO refers to sub-figure 6 below.





5.



6.