Assignment 1

Problem Statement and Model Description

Chaotic Optical Communication

Nirmal Patel

Course Instructor: Cristina Ruiz Martin

November 1, 2021

1 Problem Statement

Chaos has the deterministic and unpredictable aspects useful for secure communications. Chaos encrypted optical signals provide high degree of security in free space as well as in optical fiber. Though optical fibers are immune to electro-magnetic interference, the exposure of optical networks to eavesdropping via channel access or key access posses a considerable threat about the security of a communication system.

There are several nonlinear optical systems that may be used to produce chaotic optical signals, as for example, electro-optical (EO) modulator with feedback, accousto-optical (AO) modulators with feedback and several others. EO and AO modulator based systems have non-linearity with sinusoidal characteristics. We have used EO modulator based chaotic system for our numerical experiments. Behavior of these systems depends on the physical system as well as the feedback process.

After generating chaos we need to encrypt our message with this chaotic signal and then transmit this message through medium, in this case we have considered optical fiber as medium.

At the receiver end the message encrypted chaotic signal is used to drive the similar non-liner dynamical system which was used at transmitter end. And the output of the non-linear dynamical system is subtracted from the message encrypted chaos signals, hence we will get our original input message at the receiver. [1]

2 Model Details

- **Electro-Optic Modulator (EO):** In electro-optical modulator elctro-optic effect is used to modulate beam of light. The modulation may be imposed on the phase, frequency, amplitude, or polarization of the beam. The electro-optic effect is the change in the refractive index of a material resulting from the application of a DC or low-frequency electric field. This is caused by forces that distort the position, orientation, or shape of the molecules constituting the material.

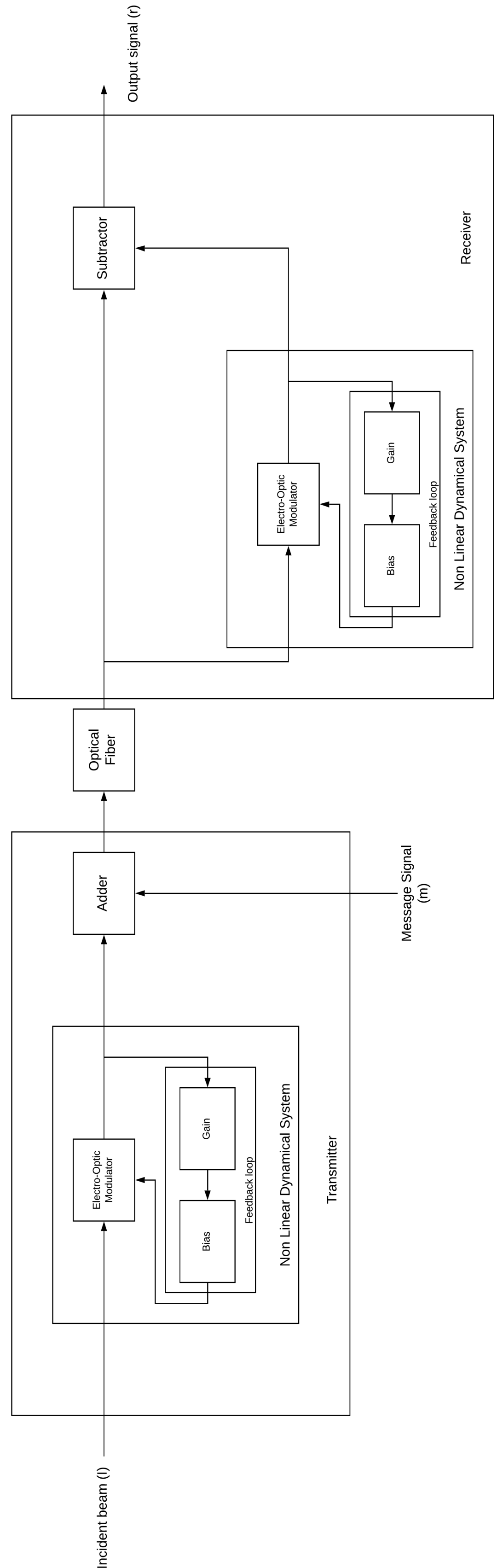
  The intensity of the output beam can be given as [2],

$$Y(n) = I_{in} \cos^2 \theta \tag{1}$$

  Where θ is the electrical input given to the EO modulator and I_{in} is the intensity of optical beam.

- **Gain:** An electronic circuit which takes the input signal and increase or decrease the amplitude of the input signal by the gain factor (β). Simply the input signal will be multiplied by the factor of β. Here β is a constant number.

Figure 1: Proposed model for Chaotic optical communication

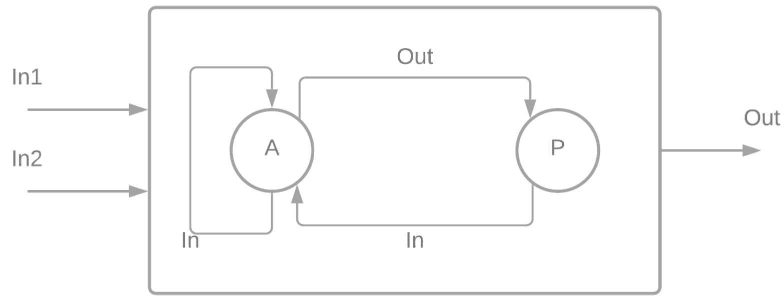


- **Bias:** An electronic set up which adds a constant bias of α to the input signal. This bias can be in form of extranal voltage or current source in the circuit.
- **Non Linear Dynamical System:** A system in which the changes in the output is not proportional to inputs. The system behave in non-linear fashion when provided input. Here we give constant optical beam as input to the non linear system and we get chaotic optical signal in output.
- **Adder:** Takes two optical signals as input and add them together to produce the output. Here, the addition operation is done in discrete manner, $I_1(n) + I_2(n) = O(n)$.
- **Optical Fiber:** We consider optical fiber as out transmission medium. For simplicity and modeling purpose we can consider that the optical fiber only attenuates the signal and all other non-linear effects of the optical fiber is ignored. So basically, the optical fiber reduces the amplitude of the input signal.
- **Subtractor:** Takes two optical signal as input and subtract one from the other and produces the resultant output. Subtractor works same as adder, i.e., $I_1(n) - I_2(n) = O(n)$.
- **Transmitter:** The whole setup which generates the chaos and encrypt the message with it. Then, send it to the optical fiber.
- **Receiver:** The whole setup which receives the signal from the optical fiber and decrypts the message signal from it and re-produces the original message signal.

3 Formal Specifications

As shown in model diagram, the top model takes two inputs, the intensity of incident optical beam and the message data to be sent. And the model produces one output which is retrieved message signal at the receiver end. This experiment frame consist of three high level components, the transmitter, the channel and the receiver. The transmitter can be further broken down in two components the Non-linear dynamic system, which is responsible to generate the chaotic signal and the adder which adds the chaos to the input message signal. Similarly, the receiver is broken into two components one is the identical Non-linear dynamic system and a subtractor which decouples the message signal from the chaotic signal. Furthermore, both the dynamical systems are broken in to two components one being the modulator and the other one being the feedback circuit. And the feedback loop is made up of a gain and a bias circuit. In this section each of this atomic models and coupled models will be defined formally.

Figure 2: Adder and Subtractor atomic model



3.1 Adder Atomic Model

The formal definition for adder atomic model is defined as $< S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_a >$,

S = {active, passive}
X = {in1, in2}
Y = {out}
$\delta_{int}(active) = passive$
$\delta_{ext}(in1, in2, active) = active$
$\delta_{ext}(in1, in2, passive) = active$
$\lambda(active)$

```
{
    out = in1 + in2
}
```

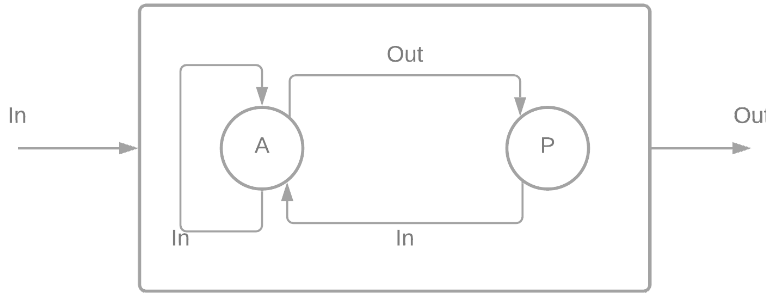
$t_a(passive) = \infty$
$t_a(active) = preparationTime$

3.2 Bias Atomic Model

The formal definition for bias atomic model is defined as $< S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_a >$,

S = {active, passive}
X = {in}
Y = {out}
$\delta_{int}(active) = passive$
$\delta_{ext}(in, active) = active$
$\delta_{ext}(in, passive) = active$
$\lambda(active)$

Figure 3: Bias, Gain, Modulator, Channel and Synchronizer model




```
{
    out = in + bias        //bias is a constant positive number
}
```

$t_a(passive) = \infty$
$t_a(active) = preparationTime$

3.3 Gain Atomic Model

The formal definition for gain atomic model is defined as $< S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_a >$,

S = {active, passive}
X = {in}
Y = {out}
$\delta_{int}(active) = passive$
$\delta_{ext}(in, active) = active$
$\delta_{ext}(in, passive) = active$
$\lambda(active)$

```
{
    out = in * gain        //gain is a constant positive number
}
```

$t_a(passive) = \infty$
$t_a(active) = preparationTime$

3.4 Channel Atomic Model

The formal definition for channel atomic model is defined as $< S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_a >$,

S = {active, passive}

X = {in}
Y = {out}
$\delta_{int}(active) = passive$
$\delta_{ext}(in, active) = active$
$\delta_{ext}(in, passive) = active$
$\lambda(active)$

```
{
    //attenuation is a constant positive number less than one
   out = in * attenuation
}
```

$t_a(passive) = \infty$
$t_a(active) = preparationTime$

3.5 Modulator Atomic Model

The formal definition for channel atomic model is defined as $< S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_a >$,

S = {active, passive}
X = {in}
Y = {out}
$\delta_{int}(active) = passive$
$\delta_{ext}(in, active) = active$
$\delta_{ext}(in, passive) = active$
$\lambda(active)$
{
out = $beam * cos^2 in$ //beam is initial constant intensity
}
$t_a(passive) = \infty$
$t_a(active) = preparationTime$

3.6 Subtractor Atomic Model

The formal definition for subtractor atomic model is defined as $< S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_a >$,

S = {active, passive}
X = {in1, in2}
Y = {out}
$\delta_{int}(active) = passive$

$\delta_{ext}(in1, in2, active) = active$
$\delta_{ext}(in1, in2, passive) = active$
$\lambda(active)$

```
{
    out = in1 - in2
}
```

$t_a(passive) = \infty$
$t_a(active) = preparationTime$

3.7 Synchronizer Atomic Model

The formal definition for Synchronizer atomic model is defined as $< S, X, Y, \delta_{int}, \delta_{ext}, \lambda, t_a >$,

S = {active, passive}
X = {in}
Y = {out}
$\delta_{int}(active) = passive$
$\delta_{ext}(in, active) = active$
$\delta_{ext}(in, passive) = active$
$\lambda(active)$

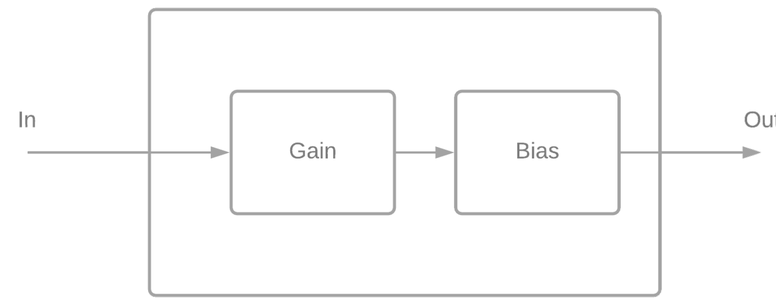
```
{
    out = in * sync        //sync is a constant less than one
}
```

$t_a(passive) = \infty$
$t_a(active) = preparationTime$

3.8 Feedback Coupled Model

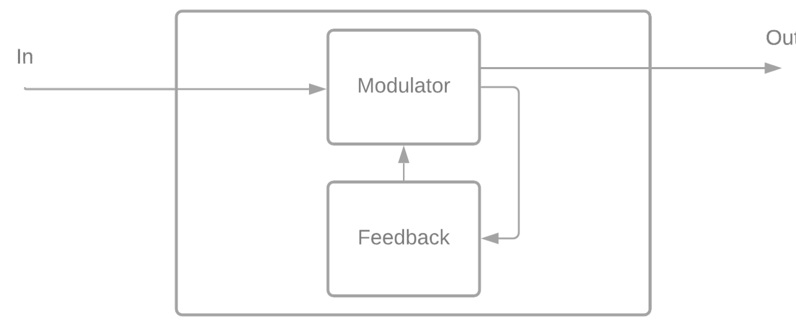
The formal definition for feedback coupled model is defined as
FEEDBACK = $< X, Y, D, EIC, EOC, IC, SELECT >$,
X = {in}
Y = {out}
D = {gain, bias}
EIC = {FEEDBACK.in, gain.in}
EOC = {bias.out, FEEDBACK.out}
IC = {gain.out, bias.in}
SELECT: ({gain, bias}) = gain

Figure 4: Feedback coupled model



3.9 Non-Linear System Coupled Model

Figure 5: Feedback coupled model

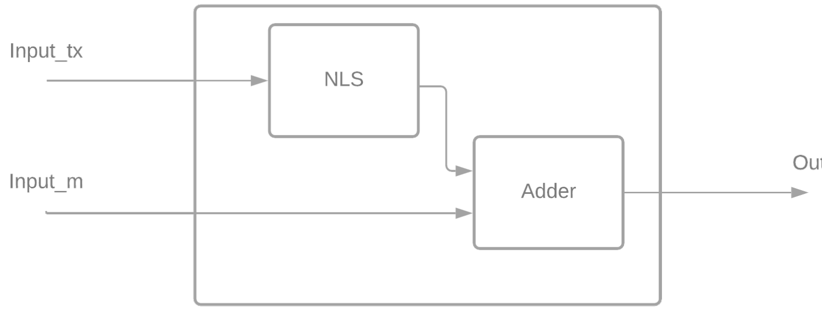


The formal definition for the non-linear system coupled model is defined as,
NLS = $< X, Y, D, EIC, EOC, IC, SELECT >$,
X = {in}
Y = {out}
D = {FEEDBACK, modulator}
EIC = {NLS.in, modulator.in}
EOC = {modulator.out, NLS.out}
IC = { (modulator.out, FEEDBACK.in), (FEEDBACK.out, modulator.out) }
SELECT: ({modulator, FEEDBACK}) = modulator

3.10 Transmitter System Coupled Model

The formal definition for the transmitter system coupled model is defined as,
TRANSMITTER = $< X, Y, D, EIC, EOC, IC, SELECT >$,
X = {input_tx, input_m}
Y = {out}

Figure 6: Feedback coupled model



D = {NLS, adder}

EIC = {(TRANSMITTER.input_tx, NLS.in), (TRANSMITTER.input_m, adder.in1)}

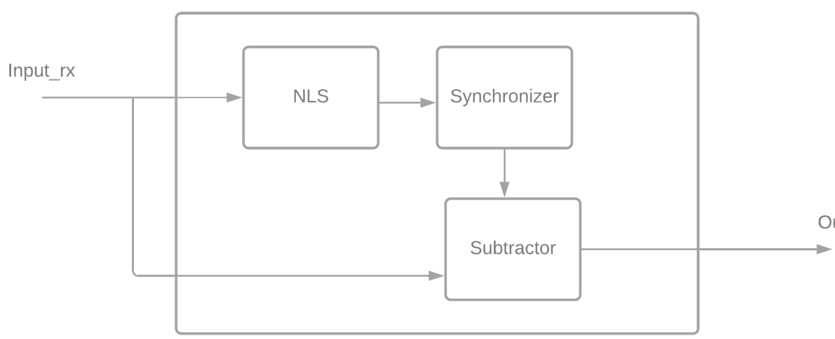
EOC = {adder.out, TRANSMITTER.out}

IC = { NLS.out, adder.in2 }

SELECT: ({NLS, adder}) = NLS

3.11 Receiver System Coupled Model

Figure 7: Feedback coupled model



The formal definition for the receiver system coupled model is defined as,

RECEIVER = $< X, Y, D, EIC, EOC, IC, SELECT >$,

X = {input_rx}

Y = {out}

D = {NLS, subtractor, sync}

EIC = {(RECEIVER.input_rx, NLS.in), (RECEIVER.input_rx, subtractor.in1)}

EOC = {subtractor.out, RECEIVER.out}

IC = { (NLS.out, sync.in), (sync.out, subtractor.in2) }

SELECT: ({NLS, subtractor, sync}) = NLS   ({sync, subtractor}) = sync

3.12 TOP Model

The formal definition for the final model is defined as,

TOP = $< X, Y, D, EIC, EOC, IC, SELECT >$,

X = ϕ

Y = {out}

D = {input_reader, message_reader, TRANSMITTER, channel, RECEIVER}

EIC = ϕ

EOC = {RECEIVER.out, TOP.out}

IC = { (input_reader.out, TRANSMITTER.input_tx), (message_reader.out, TRANSMITTER.input_m), (TRANSMITTER.out,channel.in), (channel.out, RECEIVER.input_rx }

SELECT: ({TRANSMITTER, channel, RECEIVER}) = TRANSMITTER ({channel, RECEIVER}) = channel

4 Revised Model

The transmitter produces output, which drives the subtractor and the non-linear dynamic system at the receiver. Since the non-linear dynamic system is not automic there is mismatch of the timings when the two inputs arrive at the subtractor. Hence, we have to put a synchronizer atomic model in between non-linear dynamic system and the subtractor. The revised model diagram is given on page 11,

5 Testing and Analysis of the simulation results

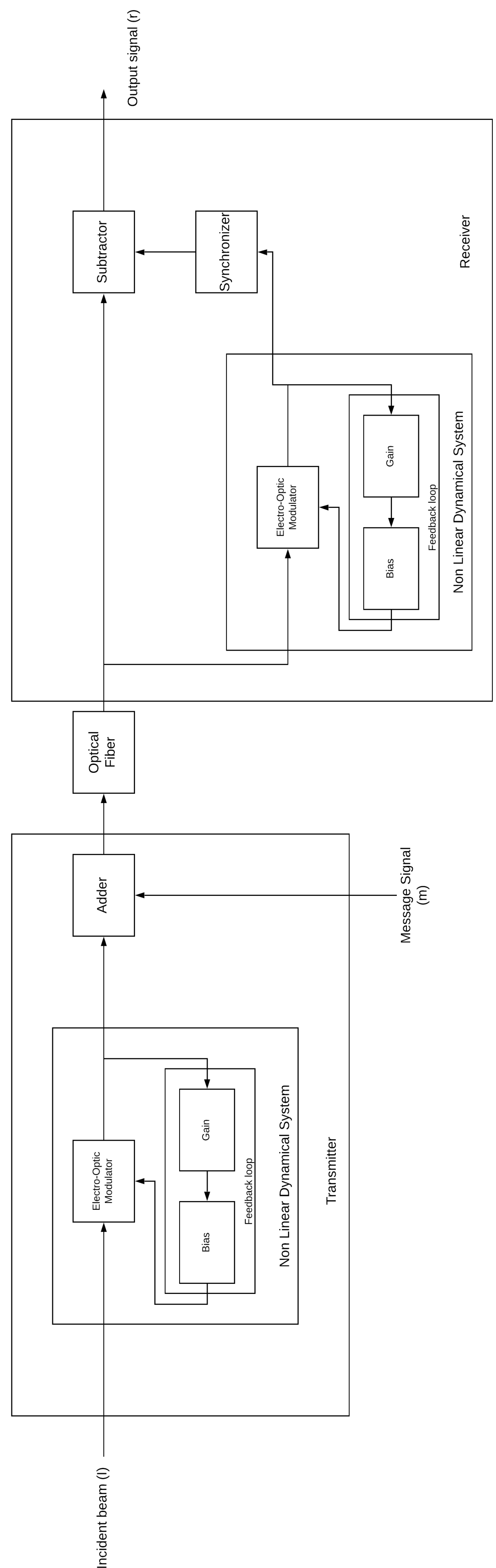
Each model is tested individually using testing .cpp files. The input to each atomic model and coupled model is passed using .txt files. Results and analysis of these tests are given below,

5.1 Testing of Adder atomic model

The adder atomic model takes two inputs and produces the addition of these inputs as the output. There are two test scenario for this model, first one is both the input arrive at the same time instant and second test case is when only one input is available or both inputs arrive at differnt time,

- **Both input ports activate at same time**: Both the ports are active "00:00:15",

Figure 8: Revised model for Chaotic optical communication



```
00:00:15:000
[iestream_input_defs<Signal_t >::out: {1}] generated by model input_reader1
[iestream_input_defs<Signal_t >::out: {8}] generated by model input_reader2
00:00:15:000
[Adder_defs::out: {9}] generated by model adder
```

- **Both inputs at different time**: One input at “00:00;10“ and other one at “00:00:15“,

```
00:00:10:000
[iestream_input_defs<Signal_t >::out: {1}] generated by model input_reader1
00:00:10:000
[Adder_defs::out: {0}] generated by model adder
00:00:15:000
[iestream_input_defs<Signal_t >::out: {8}] generated by model input_reader2
00:00:15:000
[Adder_defs::out: {0}] generated by model adder
```

5.2 Testing for Bias atomic model

Bias atomic model takes one input and adds a constant bias to that input signal and prodcues the output signal. In this case we have define the bias constant as 2, hence the output signal's intensity will be 2 units more than the input signal. Sample result is as below,

```
00:00:10:000
[iestream_input_defs<Signal_t >::out:{5.7}] generated by model input_reader
00:00:10:000
[Bias_defs::out: {7.7}] generated by model bias1
```

5.3 Testing for Gain atomic model

Gain atomic model takes one input and multiplies a constant gain factor and produces the output. In this case we have taken the gain factor as 2, so the output signal's intensity will be twice as the input signal's intensity,

```
00:00:10:000
[iestream_input_defs<Signal_t >::out: {4}] generated by model input_reader
00:00:10:000
[Gain_defs::out: {8}] generated by model gain1
```

5.4 Testing for Channel atomic model

Channel atomic model takes one input and attenuates the signal by a factor of 0.9, so the output signal's intensity will be 0.9 times the input signal's intensity and it also adds delay of 10 seconds so the output signal will be observed after 10 seconds,

```
00:00:00:000
[iestream_input_defs<Signal_t >::out: {10}] generated by model input_reader
00:00:10:000
[Channel_defs::out: {9}] generated by model channel
```

5.5 Testing for Modulator atomic model

Modulator is the non-linear component of this model, it takes the input signal intensity and takes cosine of it and the result of the cosine is squared and multiplied by constant optical signal's intensity (10 in our case). The modulator also adds a delay of 10 seconds to the output,

```
00:00:10:000
[iestream_input_defs<Signal_t >::out: {10}] generated by model input_reader
[Modulator_defs::out: {0}] generated by model modulator
00:00:20:000
[Modulator_defs::out: {7.04041}] generated by model modulator
```

5.6 Testing for Subtractor model

The subtractor model is similar to the adder model, it takes two input signals and then subtracts the second signal's intensity from the first signal. Similar to adder model this model also expects both the signal to arrive at same time,

- **Both input at same time:** Both input at "00:00;15"

```
00:00:15:000
[iestream_input_defs<Signal_t >::out: {1}] generated by model input_reader1
[iestream_input_defs<Signal_t >::out: {8}] generated by model input_reader2
00:00:15:000
[Subtractor_defs::out: {−7}] generated by model subtractor
```

- **Both inputs at different time:** The first input at "00:00:10" and the other at "00:00:15",

```
00:00:10:000
[iestream_input_defs<Signal_t >::out: {1}] generated by model input_reader1
00:00:10:000
[Subtractor_defs::out: {0}] generated by model subtractor
00:00:15:000
[iestream_input_defs<Signal_t >::out: {8}] generated by model input_reader2
00:00:15:000
[Subtractor_defs::out: {0}] generated by model subtractor
```

5.7 Testing for Synchronizer model

The synchronizer model is used to sync the inputs to the subtractor, so it is put between the non-linear system at receiver and the subtractor. It also attenuates signal by factor of 0.9,

```
00:00:10:000
[iestream_input_defs<Signal_t >::out: {10}] generated by model input_reader
[Synchronizer_defs::out: {0}] generated by model sync
00:00:20:000
[Synchronizer_defs::out: {9}] generated by model sync
```

5.8 Testing for Feedback coupled model

The feedback coupled model is made of gain and bias atomic model. Therefore, the input to the feedback coupled model first goes to gain atomic model and output of this gain model passes to bias model. So, the output of the feedback model is twice the input signal's intensity plus two,

```
00:00:10:000
[iestream_input_defs<Signal_t >::out: {4}] generated by model input_reader
00:00:10:000
[Gain_defs::out: {8}] generated by model gain1
00:00:10:000
[Bias_defs::out: {10}] generated by model bias1
```

5.9 Testing for Non-Linear System coupled model

The non-linear coupled model consist of modulator atomic model and feedback coupled model, as this is a non-linear system and it has a feedback loop which makes this non-linear system chaotic. The chaotic signal generated by this coupled model is used for modulating message signal.

```
00:00:10:000  7.12089
00:00:20:000  7.4109
00:00:30:000  1.94676
00:00:40:000  8.55698
00:00:50:000  9.31707
00:01:00:000  0.450155
00:01:10:000  9.42904
00:01:20:000  1.7967
```

From the output it is clear that there is no pattern in the signal's intensity. Furthermore, as this is a circuit with the feedback loop it only requires initial input value, from the second iteration the last iteration's output acts as input. Above given result is for input "00:00:10 10". Since this is a coupled model has feedback loop we cannot run this model until it is active. Therefore, this model is run for 1.5 minute.

5.10 Testing for Transmitter coupled model

Transmitter coupled model consists of non-linear system and adder atomic model. The output from the non-linear system (chaos) is one input to the adder circuit and the other input to the adder is the message signal to transmit. The output of the transmitter is message encrypted chaos. This chaotic message will be then passed to channel for secure communication.

- **input message signal:**

  ```
  00:00:10  1
  00:00:20  1
  00:00:30  1
  00:00:40  1
  00:00:50  0
  00:01:00  0
  00:01:10  0
  00:01:20  0
  ```

- **Chaos Encrypted message (output of transmitter)**

  ```
  00:00:10:000  8.12089
  00:00:20:000  8.4109
  00:00:30:000  2.94676
  ```

```
00:00:40:000  9.55698
00:00:50:000  9.31707
00:01:00:000  0.450155
00:01:10:000  9.42904
00:01:20:000  1.7967
```

From the output signal of transmitter, we can see that the output from the 5.9 is increased by 1 when the message signal is 1 and it is unchanged when the message signal is 0.

5.11 Testing for Receiver coupled model

Receiver coupled model is similar to the transmitter coupled model, only changes are it has a subtractor instead of adder and the input to receiver comes from channel unlike the transmitter. As we know that the output of channel will be chaotic signal for testing puroose we have passed a signal with random values. Hence, the output of the receiver will not give much information about the input signal for now.

```
00:00:10:000  −1.96182e−44        //Garbage value
00:00:20:000  −1.75287
00:00:30:000  0.0457162
00:00:40:000  −3.1354
00:00:50:000  −9.08659
00:01:00:000  5.58997
00:01:10:000  −9.39667
00:01:20:000  0.450905
```

5.12 Simulation Result for Top model

The whole system is made up of transmitter, channel and the receiver. The transmitter takes two inputs from the user, one is the initial input to the non-linear system and the other input is the message to be transmitted. Now since the receiver and the non-linear system is driven by the output signal from the transmitter there is a delay in the output at the receiver. So the input sent at time 't' will be available at receiver at time 't+10'. Moreover, the first output from the subtractor will be a garbage value. And since this model has couple of feedback loops the model will run until the time specified, so if the message signal stops before that then the last bit received by receiver will be sent until time ends.(or infinitely).

- **input message signal:**

```
00:00:10 1
00:00:20 1
00:00:30 0
00:00:40 1
00:00:50 0
00:01:00 0
00:01:10 0
00:01:20 1
```

- **Output at Receiver**

```
00:00:10:000 −1.96182e−44    //Garbage value
00:00:20:000 0.900001        //First bit of input message received
00:00:30:000 0.9
00:00:40:000 0
00:00:50:000 0.9
00:01:00:000 0
00:01:10:000 0
00:01:20:000 0
00:01:30:000 0.9             //Last bit of input message received
00:01:40:000 −2.83833        //Garbage values ignore from now
.
.
.
00:04:50:000 −6.48244        //End of simulation time
```

The input value of '1' is obtained as approximately '0.9' at the receiver due to the effect of channel, 10% of input signal intensity is reduced.

Note: Currently, the simulator is set to run for 5 minutes. If message data is more than that then first make change to main.cpp file and then run the simulation. The message signal frequency should be 10 seconds apart.

6 Conclusion and Future work

The input message bit sent at the transmitter (0 or 1) is encrypted using chaos generated at the non-linear system. This encrypted message is almost impossible to decrypt unless we cam built identical non-linear system at the receiver. There is condition to generate chaos at the non-linear system it depends on the system parameter like intensity of incident optical beam, the initial input to modulator, the bias constant and the gain constant. The combination of this four parameters determine the chaotic nature of the system. So unless we know this four parameters it is impossible to retrieve message from the encrypted signal.

There are more secure encryption method than the one used in this report (chaos masking), like chaos modulation and chaos shift keying. The behavoir of the channel i.e., the optical fiber is also a complex model. We can further go in detail of the non-linear behavior of the optical fiber channel and then model the system to obtain more accurate results.

References

[1] Anjan K. Ghosh, Nirmal Patel *Changes in the characteristics of chaotic optical signals owing to propagation in optical fibers*

[2] Aditi Datta, Anjan K. Ghosh, Anjan Mukherjee *Characterization of Chaotic Electro Optic Modulator with the help of Lyapunov Exponent and Bifurcation Analysis Technique.*