

UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
CAMPUS DE SÃO JOÃO DA BOA VISTA

FRANK GUSTAVSON FILHO

Caracterização de caos em um laser semiconductor sob injeção óptica

São João da Boa Vista

2022

Frank Gustavson Filho

Caracterização de caos em um laser semiconductor sob injeção óptica

Trabalho de Graduação apresentado ao Conselho de Curso de Graduação em Engenharia Eletrônica e de Telecomunicações do Campus de São João da Boa Vista, Universidade Estadual Paulista, como parte dos requisitos para obtenção do diploma de Graduação em Engenharia Eletrônica e de Telecomunicações .

Orientador: Prof^ª Dr^ª. Priscilla Andressa de Sousa Silva

Coorientador: Prof^º Dr. Ivan Aritz Aldaya Garde

São João da Boa Vista

2022

G982c Gustavson Filho, Frank
 Caracterização de caos em um laser semiconductor sob injeção óptica / Frank
 Gustavson Filho. -- São João da Boa Vista, 2022
 75 p. : il., tabs., fotos

 Trabalho de conclusão de curso (Bacharelado - Engenharia de Telecomunicações) -
Universidade Estadual Paulista (Unesp), Faculdade de Engenharia, São João da Boa
Vista

 Orientadora: Priscilla Andressa de Sousa Silva
 Coorientador: Ivan Aritz Aldaya Garde

 1. Comportamento caótico nos sistemas. 2. Sistemas dinâmicos diferenciais. 3.
Comunicações ópticas. 4. Lasers semicondutores. 5. Semiconductor lasers. I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca da Faculdade de Engenharia, São João da Boa Vista. Dados fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

**UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”
FACULDADE DE ENGENHARIA - CÂMPUS DE SÃO JOÃO DA BOA VISTA
GRADUAÇÃO EM ENGENHARIA ELETRÔNICA E DE TELECOMUNICAÇÕES**

TRABALHO DE CONCLUSÃO DE CURSO

**CARACTERIZAÇÃO DE CAOS EM UM LASER SEMICONDUTOR SOB INJEÇÃO
ÓPTICA**

Aluno: Frank Gustavson Filho

Orientadora: Profa. Dra. Priscilla Andressa de Sousa Silva

Banca Examinadora:

- Priscilla Andressa de Sousa Silva (Orientadora)
- Juan Pablo Pantoja Bastidas (Examinador)
- Rafael Abrantes Penchel (Examinador)

A ata da defesa com as respectivas assinaturas dos membros encontra-se no prontuário do aluno (Expediente nº 010/2022)

São João da Boa Vista, 15 de junho de 2022

Dedico este trabalho a toda minha família, por sempre acreditarem em mim e por terem me apoiado muito ao longo de minha vida.

AGRADECIMENTOS

Primeiramente agradeço a Deus, por permitir alcançar meus objetivos durante todos os meus anos de estudos.

Também agradeço a todos meus familiares, que me incentivaram e apoiaram a não desistir dos meus sonhos.

E por fim agradeço também a Prof^a Dr^a. Priscilla Andressa de Sousa Silva por ter aceitado me ajudar e instruir, não só na elaboração deste trabalho, mas de todos os que realizamos ao longo dessa jornada de aprendizado.

*“With God all things are possible“
- Mateus 19:26 -*

RESUMO

Este trabalho objetivou o cálculo do expoente de Lyapunov para caracterização de caos de um sistema de lasers semicondutores bloqueados por injeção óptica (*Optical Injection Locking*, OIL). Inicialmente, foram estudados conceitos fundamentais sobre dinâmica caótica e sobre sistemas de modulação de diodos lasers bloqueados por injeção óptica, a fim de determinar as regiões de bloqueio do laser e também estabelecer condições de estabilidade e instabilidade. Então, o expoente máximo de Lyapunov foi calculado para diversas condições iniciais em diferentes regiões e diferentes condições de estabilidade. Com isso, foram localizados pontos onde existe um comportamento caótico, ou seja, pontos onde o expoente de Lyapunov resultou em um valor positivo. Com esse comportamento caracterizado é possível direcionar futuros estudos da aplicação do sistema em regiões onde o comportamento é regular ou até mesmo investigar a aplicabilidade de métodos de controle de caos.

PALAVRAS-CHAVE: Sistemas caóticos. Caracterização de caos. Laser semicondutor. Sistema de bloqueio por injeção.

ABSTRACT

This work aimed to calculate the Lyapunov exponent for chaos characterization of a system of semiconductor lasers blocked by optical injection (*Optical Injection Locking, OIL*). Initially, fundamental concepts about chaotic dynamics and about diode laser modulation systems blocked by optical injection were studied, in order to determine the laser blocking regions and also to establish stability and instability conditions. Then, the maximum Lyapunov exponent was calculated for different initial conditions in different regions and different stability conditions. With this, points were located where there is a chaotic behavior, that is, points where the Lyapunov exponent resulted in a positive value. With this behavior characterized, it is possible to direct future studies of the application of the system in regions where the behavior is regular or even to investigate the applicability of chaos control methods.

KEYWORDS: Chaotic systems. Chaos characterization. Laser semiconductor. Injection blocking system.

LISTA DE ILUSTRAÇÕES

Figura 1	Esquema do sistema de bloqueio por injeção óptica.	14
Figura 2	Comparação de ganho entre <i>free-running</i> e método OIL.	15
Figura 3	Espectros de ruído de fase para sistemas OPLL, travado por injeção óptica (OIL) e OIPLL (OIL + OPLL).	16
Figura 4	Interpretação geométrica do método de Newton-Raphson.	22
Figura 5	Análise de estabilidade de um ponto fixo.	23
Figura 6	Análise gráfica do método RK4.	25
Figura 7	Mapa de bloqueio e estabilidade.	33
Figura 8	Resultados para diferença de frequência para dessintonização nula e razão de injeção de 0 dB.	36
Figura 9	Resultados para diferença de frequência para dessintonização nula e razão de injeção de -8 dB.	38
Figura 10	Resultados para diferença de frequência para dessintonização nula e razão de injeção de -8.75 dB.	42
Figura 11	Resultados para diferença de frequência para dessintonização nula e razão de injeção de -19.15 dB.	46

LISTA DE TABELAS

Tabela 1 – Identificação dos parâmetros do sistema óptico	27
Tabela 2 – Parâmetros iniciais	31

LISTA DE ABREVIATURAS E SIGLAS

DC	<i>Direct current</i>
DWDM	<i>Dense wavelength division multiplexing</i>
LASER	<i>Light amplification by stimulated emission of radiation</i>
LED	<i>Light-emiting diode</i>
MZM	<i>Mach Zehnder modulator</i>
OSNR	<i>Optical signal-to-noise ratio</i>
OIL	<i>Optical injection locking</i>
OIPLL	<i>Optical injection phase-locked loop</i>
OPLL	<i>Optical phase-locked loop</i>
RK4	Runge-Kutta de quarta ordem

LISTA DE SÍMBOLOS

$A(t)$	Amplitude de campo normalizada
A_0	Magnitude do campo em estado estacionário
A_{inj}	Amplitude do campo da luz injetada
A_{fr}	Magnitude do campo em <i>free-running</i>
g	Coefficiente de ganho do laser
J	Corrente de bombeio
$N(t)$	Número de portadores no laser escravo
N_0	Número de portadores em estado estacionário
N_{th}	Número limite de portadores
R_{int}	Razão de injeção
α	Fator de aprimoramento de largura de linha
τ_{IN}	Tempo de <i>round-trip</i>
κ	Coefficiente de acoplamento
$\phi(t)$	Diferença de fases dos lasers escravo e mestre
ϕ_0	Fase em estado estacionário
r_0	Reflectividade dos espelhos
γ_N	Taxa de recombinação de portadores
γ_P	Taxa de decaimento de fótons
$\Delta\omega_{inj}$	Frequência de dessintonização entre mestre e escravo

SUMÁRIO

1	INTRODUÇÃO	13
2	INTRODUÇÃO A SISTEMAS ÓPTICOS	17
2.1	Transmissão	17
2.2	Modulação	18
2.3	Recepção	18
3	FUNDAMENTAÇÃO TEÓRICA	20
3.1	Sistemas dinâmicos	20
3.1.1	Classificações de sistemas dinâmicos	20
3.1.2	Pontos fixos e órbitas periódicas	21
3.1.3	Dinâmica caótica	23
3.2	Método de Runge-Kutta	24
3.3	Expoente de Lyapunov	26
4	EQUAÇÕES DO SISTEMA	27
5	RESULTADOS	31
5.1	Mapa de bloqueio	31
5.2	Resolução das equações do sistema e cálculo do expoente de Lyapunov	34
6	CONCLUSÃO	51
	REFERÊNCIAS	52
	APÊNDICE A – SCRIPT - MAPA DE BLOQUEIO	54
	APÊNDICE B – SCRIPT - RK4 - SISTEMA OIL	58
	APÊNDICE C – SCRIPT - EXPOENTE LYAPUNOV	66

1 INTRODUÇÃO

Lasers semicondutores são as fontes de luz mais amplamente utilizada em sistemas de comunicações ópticas. Uma vez que são de baixo custo, compactos, de fácil integração e apresentam altas potências de saída, eles têm grande potencial para atender redes ópticas que demandam velocidades cada vez mais altas.

No entanto, esses dispositivos apresentam limitações relevantes em termos de desempenho de modulação. A necessidade de frequências de modulação mais altas que permitam maiores taxas de transmissão de dados tem motivado a busca por formas de melhorar o desempenho da modulação dos lasers semicondutores. Uma das formas encontradas é a injeção óptica, que possibilita aos lasers ultrapassarem seu limite de velocidade de modulação fundamental graças ao aumento da largura de banda e frequência de ressonância dos lasers modulados (LAU, 2006).

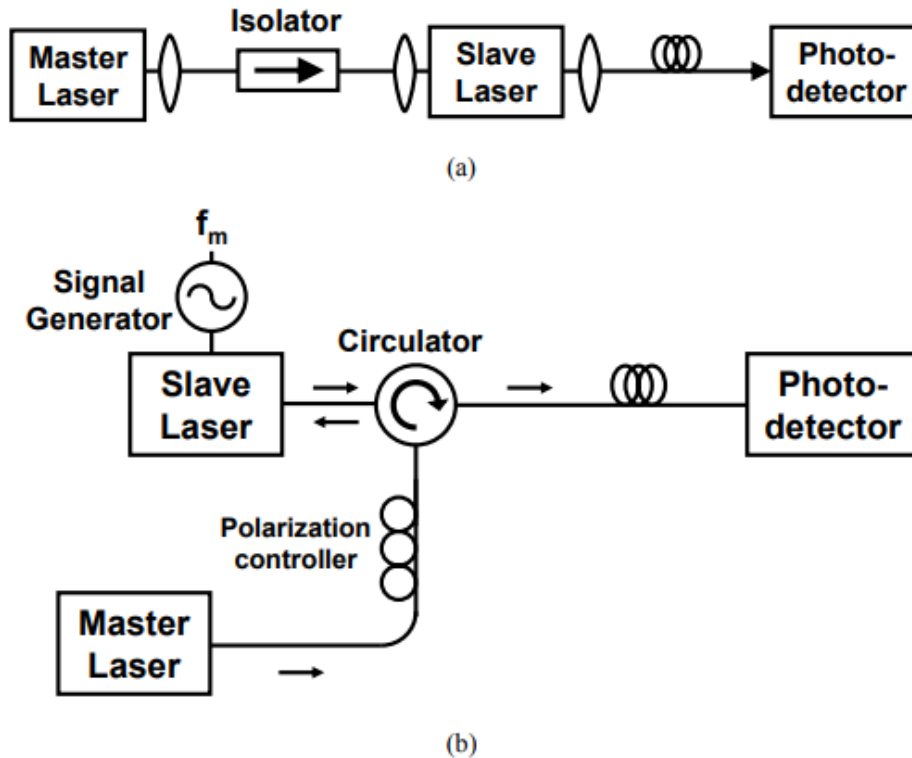
O sistema de injeção óptica surgiu a partir de uma observação feita pelo cientista Christiaan Huygens, em 1666. Em um dos seus experimentos ele observou que dois pêndulos, de relógios distintos, montados na mesma parede e sem nenhuma influência externa, acabavam travando as frequências e oscilando com fases opostas. Com isso ele formulou a hipótese de que os pêndulos deveriam de alguma forma afetar uns aos outros. Por fim, ele concluiu que os pêndulos estavam acoplados, emitindo vibrações que passavam pela parede que os sustentava. Um pêndulo envia vibrações que atravessaram a parede e “injeta” pequenas perturbações no outro pêndulo, eventualmente, bloqueando a frequência e a fase dos dois pêndulos (PANTALEONE, 2002).

Com esse experimento realizado por Huygens, foram introduzidos os primeiros conceitos de bloqueio por injeção, neste caso para sistemas mecânicos. Tempos depois, em 1946, foi publicado, por R. Adler, o primeiro trabalho sobre bloqueio por injeção em um sistema elétrico. Neste sistema, ele promoveu o bloqueio por injeção de um oscilador elétrico utilizando uma fonte de frequência externa.

Alguns anos depois da descoberta de Adler, quando uma fonte de luz coerente foi inventada na forma do diodo laser, finalmente os conceitos de bloqueio por injeção foram aplicados à luz. A partir desse ponto, novos estudos começaram a surgir em cima dos conceitos iniciais, como por exemplo, em 1966 foi feita a aplicação para diodos lasers seguida da demonstração do primeiro laser bloqueado por injeção, utilizando dois lasers vermelhos de HeNe. Por fim, o método ganhou grande destaque em 1980, quando S. Kobayashi e T. Kimura realizaram a primeira demonstração de bloqueio por injeção em lasers semicondutores de GaAs. Nesta demonstração, dois lasers foram bloqueados no mesmo comprimento de onda, o que tornou a ideia atraente para usos de osciladores locais em sistemas de detecção coerentes (LAU, 2006).

O método desenvolvido por Kobayashi e Kimura consiste no uso de dois lasers, um chamado de mestre e o outro escravo. A luz do laser mestre é injetada em um laser escravo e a luz do escravo é a saída do sistema. O sistema completo pode ser visto na Figura 1, onde também podemos ver a adição de um isolador Figura 1(a) ou circulador Figura 1(b) entre o mestre e o escravo para eliminar possíveis reflexões de volta ao mestre. Além disso, é necessário um controlador de polarização para garantir que as polarizações do mestre e do escravo sejam compatíveis.

Figura 1 – Esquema do sistema de bloqueio por injeção óptica.



fonte: (LAU, 2006)

No sistema, a luz do mestre é injetada no escravo, não necessariamente no mesmo comprimento de onda. Após um período de tempo, a dinâmica do laser se estabiliza, o comprimento de onda do escravo é puxado em direção ao comprimento de onda do mestre, até que se iguale ao do mestre, bloqueando sua frequência e fase. Agora, se a frequência do laser mestre for alterada, o escravo rastreará essa frequência até que a diferença entre as frequências do laser mestre e a frequência do laser escravo em *free-running* (sem injeção de uma fonte externa) se torne muito grande. Neste ponto, o escravo se desbloqueia do mestre e dispara o laser em seu comprimento de onda natural.

Como principais características e aplicações do método de bloqueio por injeção em lasers, podemos citar:

Redução de largura de linha. Lasers de largura de linha estreita são cruciais para aplicações que incluem sistemas de comunicação óptica coerente. Sua largura de linha é normalmente reduzida através do prolongamento do tempo de vida do fóton usando uma cavidade externa ou travando o laser em uma referência estável com eletrônica avançada. Em ambos os casos, temos um aumento muito grande nos custos. Nesse ponto entra o método OIL, o qual utiliza lasers de baixo custo com largura de linha relativamente grande que podem ser transformados em lasers de largura de linha estreita de alto desempenho por meio do uso do OIL (LIU; SLAVÍK, 2020).

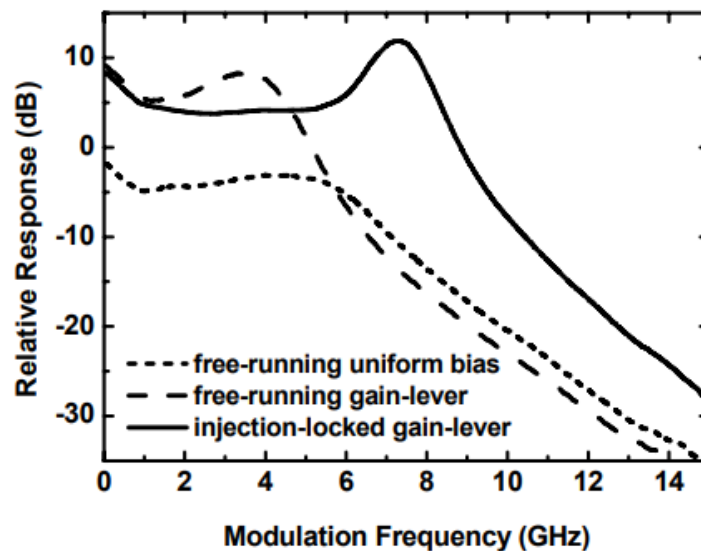
Seleção e amplificação de tons. Em comunicações ópticas DWDM (*Dense Wavelength Division Multiplexing*), muitas vezes é necessário isolar e amplificar um sinal de frequência única. Na maioria dos casos, isso é realizado usando filtros ópticos, que introduzem uma perda de inserções. Por conta das perdas introduzidas, também é necessário utilizar um amplificador para aumentar a

potência do sinal. Contudo, essa combinação de filtro e amplificador reduz a relação sinal-ruído óptico (OSNR). Um método alternativo que fornece uma filtragem de baixo ruído e alto ganho é o sistema OIL. Isso ocorre quando o laser escravo sintoniza no canal de entrada selecionado e realiza a filtragem óptica amplificando o canal selecionado, mas atenua o sinal fora da faixa de bloqueio (LIU; SLAVÍK, 2020).

Redução de *chirp*. O método OIL reduz significativamente o efeito de *chirp* quando o laser escravo opera na região de bloqueio estável. Sob o OIL, a frequência do laser escravo permanece constante, seguindo a frequência do mestre. Embora ainda haja algum nível residual do *chirp* de modulação, o *chirp* de frequência de modulação é fortemente reduzido (LIU; SLAVÍK, 2020).

Maior ganho de enlace. Em um laser de duas seções, cada seção é polarizada em diferentes níveis. A seção de baixa polarização é oblíqua em um ponto mais baixo da curva de ganho, a qual tem um ganho diferencial mais alto e, portanto, maior força de modulação. Infelizmente, a maior inclinação do ganho coincide com o aumento da não linearidade. A alavancagem de ganho é, então, acoplada ao bloqueio de injeção para obter maior ganho diferencial, mas maior linearidade via OIL (LAU, 2006). A Figura 2 mostra a comparação do ganho em *free-running* e utilizando o método OIL. Observando essa imagem podemos verificar que o ganho utilizando o método de injeção óptica é maior em toda a faixa de frequência de modulação, chegando a ser, em média, 10 dB maior que o de *free-running*.

Figura 2 – Comparação de ganho entre *free-running* e método OIL.

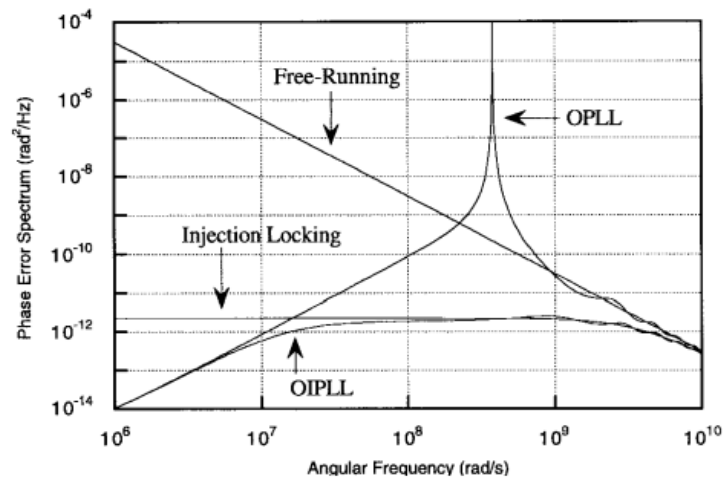


fonte: (LAU, 2006)

Loop óptico de bloqueio de fase. Na comunicação coerente, o sinal transmitido coincide com um sinal óptico DC, chamado oscilador local, na extremidade do receptor. Isso permite um ganho de *link* aumentado e diferentes formatos de modulação. No entanto, o oscilador local deve ser bloqueado em fase para o tom do sinal. Tipicamente, um *loop* óptico de bloqueio de fase (OPLL, do inglês *optical phase-locked loop*) é implementado. No entanto, o ruído na largura de linha do laser é muito rápido para ser reduzido pelo OPLL. Desta forma, para melhorar esse comportamento

pode-se também utilizar o laser bloqueado por injeção como um mecanismo de bloqueio de frequência (LAU, 2006). A Figura 3 mostra uma comparação entre os métodos em relação ao erro no espectro de fase.

Figura 3 – Espectros de ruído de fase para sistemas OPLL, travado por injeção óptica (OIL) e OIPLL (OIL + OPLL).



fonte: (LAU, 2006)

Estudando o método de forma mais aprofundada, verificou-se a existência de regiões que apresentam comportamento caótico. Pensando nisso, este trabalho tem como objetivo implementar o cálculo do expoente máximo de Lyapunov para caracterização de caos no sistema, a fim de guiar os estudos e aplicações do sistema nessas condições.

Para calcular o expoente, foi necessário encontrar as regiões de bloqueio do laser junto com suas respectivas estabilidades, resolver as equações diferenciais que regem o sistema utilizando o integrador numérico Runge-Kutta de quarta ordem e por fim, implementar um algoritmo para calcular o próprio expoente. Com isso, é possível obter um maior entendimento sobre a complexidade dinâmica do sistema, podendo guiar os estudos da aplicação do sistema em condições onde não foi verificado comportamento caótico ou até mesmo motivando a implementação de métodos de controle para sistemas caóticos.

Por fim, este trabalho está organizado da seguinte forma: no Capítulo 2 é apresentada uma breve introdução aos sistemas ópticos. No Capítulo 3 são descritos os conceitos básicos de sistemas dinâmicos e os métodos utilizados para solução do sistema e cálculo do expoente de Lyapunov. No Capítulo 4 as equações que regem o sistema de bloqueio por injeção óptica e por fim, no Capítulo 5, são apresentados os resultados obtidos.

2 INTRODUÇÃO A SISTEMAS ÓPTICOS

Neste capítulo serão introduzidos alguns conceitos básicos de sistemas ópticos, a fim de estabelecer uma base dos elementos e conceitos fundamentais de funcionamento do sistema óptico bloqueado por injeção.

2.1 TRANSMISSÃO

Iniciando pelos transmissores ópticos, temos que sua função é converter o sinal elétrico para o domínio óptico. Para isso podem ser utilizados alguns tipos de fontes ópticas, como por exemplo, um LED (*light-emitting diode*) ou um laser semiconductor.

O LED é um dispositivo semiconductor que consegue alterar a potência óptica conforme uma corrente elétrica passa por ele. O fenômeno que ocorre nesse tipo de dispositivo é chamado de emissão espontânea, ou seja, quando existirem elétrons livres na banda de condução, estes poderão retornar à banda de valência e quando isso ocorre fótons são emitidos em todas as direções e sem relação de fase entre si (CARMO, 2017).

Esses tipos de transmissores normalmente são utilizados com fibras multimodo na janela de transmissão de 850 nm e são mais simples de serem instalados exigindo *drivers* menos complexos. Também possuem um baixo custo, maior tempo de vida e apresentam uma menor variação as condições atmosféricas.

Contudo, os transmissores LED apresentam algumas desvantagens, como por exemplo, potência óptica menor, largura do espectro maior, chaveamento mais lento e pequena taxa de transmissão. Outro ponto extremamente negativo é em relação a eficiência global do LED que é muito baixa, uma vez que apenas 1% da potência elétrica aplicada é convertida em potência luminosa de saída (TELECOMUNICAÇÕES, 2022).

Já os diodos laser (do inglês, *Light Amplification by Stimulated Emission of Radiation*) são os transmissores mais utilizados atualmente para aplicações em comunicações ópticas. O fenômeno predominante que ocorre nesse tipo de dispositivo é chamado de emissão estimulada, ou seja, um fóton perturba um elétron em um estado de energia superior causando a recombinação com a lacuna em um nível inferior (CARMO, 2017).

Para que esse processo ocorra, é necessária a presença de um outro fóton e os fótons emitidos durante este processo possuem a mesma energia, frequência, polarização e direção de propagação da radiação estimulante, ou seja, podemos considerar que a luz emitida por um diodo laser é coerente.

Como vantagens dos lasers podemos citar, maior potência óptica, largura espectral menor, velocidades de modulação maior e menor ruído. Contudo, os lasers também apresentam algumas desvantagens, como por exemplo, maior sensibilidade à temperatura, menor vida útil e custos mais elevados (TELECOMUNICAÇÕES, 2022).

2.2 MODULAÇÃO

Após decidir como é feita a conversão do sinal elétrico para o sinal óptico, é preciso decidir a forma de modulação que será utilizada. A modulação é uma técnica na qual as características de uma onda portadora são modificadas com a finalidade de transmitir informações. As características que podem ser alteradas são amplitude, frequência, fase, etc. A modulação óptica pode ser feita de duas maneiras, a primeira aplicando o sinal elétrico direto na fonte óptica e a segunda aplicando o sinal elétrico em um modulador externo.

Primeiro, para modulação direta temos que a informação é modulada sobre a corrente que alimenta o laser. Isso acaba gerando algumas limitações físicas no uso desse sistema para taxas de transmissão menores de no máximo 2,5 Gb/s ou para sistemas muito curtos de até 10 Gb/s.

Os lasers semicondutores modulados diretamente são transmissores compactos e de baixo custo para sistemas de comunicação fotônica digital e analógica. No entanto, seu uso em sistemas fotônicos analógicos de alto desempenho é limitado por vários problemas de desempenho, como por exemplo, frequência de oscilação de relaxamento, acoplamento não linear de elétron-fóton, ruído de emissão espontânea amplificado e *chirp* (AGRAWAL, 2012).

A segunda forma de modulação é a de eletro-absorção que é a variação do limiar de absorção de um material com base em uma tensão externa. O seu princípio de funcionamento baseia-se em uma alteração do espectro de absorção causado por um campo elétrico aplicado, que normalmente não envolve a excitação de portadores pelo campo elétrico (CARMO, 2017). Por conta da variação da absorção, essa forma de modulação gera um alargamento no espectro, fazendo que apareçam frequências maiores e menores. Esse método também apresenta baixo nível de controle, tornando-se inviável para altas taxas de transmissão.

Como último tipo de modulação, temos a modulação que utiliza o modulador de Mach-Zehnder, nele utiliza-se uma alteração no campo elétrico para alterar os índices de refração do guia de onda. Essa modulação pode ser feita pela amplitude, que seria a alteração da fase dos sinais gerando interferência construtiva ou destrutiva. Também, é possível aproveitar a alteração da fase para adicionar informação a transmissão. Isso é feito utilizando um sinal óptico que será ramificado para vários MZM (*Mach Zehnder Modulator* que alteraram as fases de forma diferentes. No final é necessário amplificar o sinal por conta das ramificações utilizadas e, para esse método, também é necessária uma recepção coerente do sinal (AGRAWAL, 2012).

2.3 RECEPÇÃO

Por fim, devemos escolher qual será a forma de recepção do sinal óptico será utilizada, ou seja, como iremos transformar o sinal óptico novamente para o domínio elétrico. Atualmente são utilizados dois métodos para detecção, os que utilizam receptores coerentes e os que utilizam detecção direta.

O receptor de detecção direta utiliza um método que só detecta a amplitude do sinal, sendo, portanto, insensível à frequência, fase ou polarização da portadora sobre o seu regime de funcionamento. Isso ocorre, basicamente, por conta do processo que capta o número de fótons, e ao detectar o sinal ótico incidente o converte para um sinal elétrico (CARMO, 2017).

Como vantagens desse método temos, principalmente, o baixo custo, uma vez que não exige circuitos integrados específicos nem processadores de sinais digitais. Outro ponto no qual este método leva vantagem é no consumo de energia que é razoavelmente menor, visto que o sistema utiliza menos componentes adicionais.

Como desvantagens temos menor alcance e principalmente a capacidade de tráfego, pois ao descartar informações de frequência e fase, o sistema se torna menos eficiente (HTFUTURE, 2020).

Para receptores com detecção coerente temos que o sinal óptico é misturado com o sinal de um oscilador local. Este sinal óptico, ao passar por um fotodetector, gera uma corrente que possui informações de amplitude, frequência e fase do sinal óptico recebido.

Existem dois tipos de receptores coerentes: os homódinos e os heteródinos. No receptor homódino, a frequência do oscilador local coincide com a frequência da portadora óptica. Já no receptor heteródino, a frequência do oscilador local é diferente da frequência da portadora óptica, correspondendo à diferença entre as frequências do oscilador local e da portadora (CARMO, 2017).

Como vantagens deste método temos o aumento na capacidade, uma vez que este pode extrair mais informações do sinal recebido; maior simplicidade, por utilizar um processador de sinais digitais, o qual compensa grande parte da dispersão cromática e de modo de polarização; e por fim, o alcance que pode chegar a milhares de quilômetros.

Como desvantagens temos os custos e consumo de energia, os quais são maiores quando comparados aos receptores com detecção direta (HTFUTURE, 2020).

3 FUNDAMENTAÇÃO TEÓRICA

Tendo em vista que o sistema de bloqueio por injeção óptica é um sistema com comportamento caótico, neste capítulo são apresentados alguns conceitos fundamentais sobre sistemas dinâmicos, suas formas de classificação e identificação de comportamento caótico. Por fim, também serão introduzidos alguns métodos utilizados ao longo do trabalho para solucionar as equações do sistema.

3.1 SISTEMAS DINÂMICOS

Um sistema é um conjunto de elementos que, de alguma forma, apresentam uma integração entre si, de modo que existem relações de causa e efeito nas respostas exibidas pelos elementos desse conjunto. Em um sistema dinâmico, os elementos interagem entre si ao longo do tempo, seguindo uma regra que determina o estado presente em termos de estados passados ou futuros (ALLIGOOD; SAUER; YORKE, 1996).

Esse conceito foi estabelecido pela necessidade de se construírem modelos gerais para sistemas físicos que evoluem no tempo. Como exemplos de sistemas físicos com comportamento dinâmico temos o balanço de um pêndulo, o movimento de um satélite, o crescimento de uma população de microrganismos, entre outros.

3.1.1 Classificações de sistemas dinâmicos

Os sistemas dinâmicos podem ser classificados de algumas formas. Por exemplo, quanto à natureza da variável temporal, quanto à linearidade do modelo e se são conservativos ou dissipativos.

Quando classificamos sistemas dinâmicos quanto a variável temporal temos duas opções: eles podem ser de tempo discreto ou de tempo contínuo. Um sistema é de tempo discreto se o tempo t é um número inteiro, ou seja, $t \in \mathbb{Z}$, e a evolução do sistema é descrita por equações de diferença. Estas equações relacionam o valor de uma variável em um dado instante t a valores em outros instantes, $t + 1$, $t + 2$, ..., $t + n$. Já um sistema é dito de tempo contínuo quando $t \in \mathbb{R}$, isto é, o tempo assume valores nos números reais. A evolução de sistemas de tempo contínuo são descritas por equações diferenciais escritas em termos de derivadas de variáveis desconhecidas (E-DISCIPLINAS-USP, 2022).

Outra forma de classificação mencionada é quanto à linearidade dos modelos. Nesse caso, temos que um sistema dinâmico pode ser linear ou não linear. Sistemas lineares são conjuntos de equações lineares, ou seja, equações constituídas por relações algébricas entre variáveis de grau um, como mostrado na equação:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m. \end{cases} \quad (1)$$

Graficamente temos que equações lineares podem ser representadas por retas, planos ou hiperplanos (STEINBRUCH; WINTERLE, 1987).

Já para um sistema não linear existe um conjunto de equações não lineares, ou seja, equações constituídas por relações algébricas entre variáveis de grau maior que um ou por combinação de funções algébricas, tais como funções exponenciais, funções logarítmicas, funções trigonométricas, etc (GOUVEIA, 2018). Resolver um sistema não linear significa encontrar uma solução tal que:

$$\begin{cases} f_1(x_1, x_2, \dots, x_n) = 0 \\ f_2(x_1, x_2, \dots, x_n) = 0 \\ \vdots \\ f_n(x_1, x_2, \dots, x_n) = 0. \end{cases} \quad (2)$$

Por fim, podemos classificar sistemas dinâmicos como conservativos ou dissipativos. No primeiro caso, a energia mecânica do sistema é conservada, ou seja, a energia potencial do sistema é integralmente transformada em energia cinética. No caso de sistemas dissipativos, temos que parte da energia mecânica é transformada em outro tipo de energia, tal como energia térmica ou sonora (DOCA; BISCOLOLA; BÔAS, 2012).

Outra maneira de determinar se o sistema é conservativo ou dissipativo é analisando seu espaço de fases. Espaço de fases é um espaço abstrato onde se representa o comportamento de um sistema, de forma que um ponto no espaço fase define um potencial estado do sistema (SOUZA, 2008). Neste caso, para sistemas conservativos temos a preservação de volume no espaço de fase. Já para sistemas dissipativos existe a contração de volume no espaço de fase.

3.1.2 Pontos fixos e órbitas periódicas

Falando agora especificamente de sistemas dissipativos, temos que podem existir regiões limitadas do espaço de fase, chamadas de atratores, para as quais as trajetórias convergem. Um atrator pode ser definido como um conjunto fechado A que obedece às seguintes características:

1. A é um conjunto invariante, ou seja, qualquer trajetória $x(t)$ que começa em A , permanece em A todo o tempo.
2. A atrai um conjunto aberto de condições iniciais, isto é, há um hiper-volume B , que contém A , tal que para qualquer condição inicial pertencente a B , a distância entre a trajetória $x(t)$ e A tende a zero quando t tende a infinito.
3. A é mínimo, ou seja, não há subconjunto de A .

Atratores podem ser caóticos ou periódicos. Para sistemas dinâmicos de tempo contínuo, também chamados de fluxos, essa trajetória periódica é representada por uma curva fechada no espaço de fases. Para sistemas dinâmicos de tempo discreto, chamados de mapas, uma órbita periódica de período k é composta por k pontos diferentes x_1, x_2, \dots, x_k que se repetem periodicamente. Ou seja, de x_1 obtém-se x_2 , de x_2 obtém-se x_3 , e assim por diante, até que, de um determinado ponto x_k , volta-se ao ponto de partida x_1 (MONTEIRO, 2006). A partir daí o ciclo se repete, conforme mostrado na equação:

$$\begin{aligned}
x_2 &= f(x_1) \\
x_3 &= f(x_2) = f(f(x_1)) \\
&\vdots \\
x_k &= f(x_{k-1}) = f(f(x_{k-2})) = f^{(k-1)}(x_1) \\
x_{k+1} &= f(x_k) = f(f(x_{k-1})) = f^{(k)}(x_1) = x_1.
\end{aligned} \tag{3}$$

Um método muito utilizado para determinar numericamente as órbitas periódicas e pontos fixos dos sistemas é o método de Newton-Raphson, que tem como objetivo estimar as raízes da função $F(x) = f^k(x) - x$, onde $f^k(x)$ é a k -ésima aplicação do mapa dinâmico $x_{x+1} = f(x_n)$, $x \in \mathbb{R}^m$, $m \geq 1$. O método é descrito pela equação:

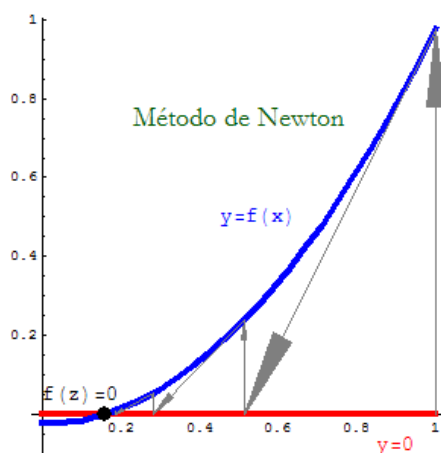
$$x_{n+1} = x_n - J^{-1}F(x_n), \tag{4}$$

onde J é a matriz jacobiana, que se reduz à derivada da função F no caso unidimensional.

Para utilizar o método, escolhe-se uma aproximação inicial e calculam-se correções para a aproximação. Repetindo esse processo, tem-se um método iterativo para encontrar o valor aproximado do zero da função.

Para uma melhor compreensão, podemos estudar a interpretação geométrica do método. Como podemos ver na Figura 4, temos um chute inicial $x^{(1)}$, e para iniciar o método devemos calcular a reta tangente a curva originada em $x^{(1)}$. Depois de calculada a reta tangente, devemos tomar a interseção desta reta com o eixo das abcissas, originando assim $x^{(2)}$. Desta forma, podemos observar que após algumas iterações a interseção da reta tangente e o eixo das abcissas se aproxima da raiz da função até que o ponto de interseção seja o mesmo da raiz. Quando isso ocorre significa que encontramos uma aproximação numérica para raiz da função (PARKER; CHUA, 1989).

Figura 4 – Interpretação geométrica do método de Newton-Raphson.



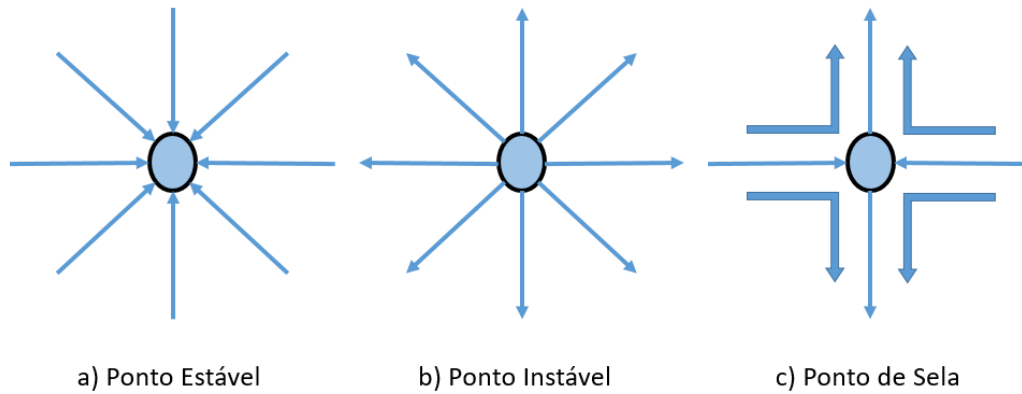
fonte: <<https://www.math.tecnico.ulisboa.pt/~calves/MC/cap2.html>>

Outro ponto muito importante, além da periodicidade de uma solução, é a análise de sua estabilidade. Essa estabilidade pode ser analisada primeiramente observando o comportamento de outras soluções

do sistema com condição inicial próxima a um ponto fixo ou órbita periódica. Ou seja, ao escolhermos um ponto de partida relativamente próximo a um ponto fixo ou órbita periódica e iterarmos as equações do sistema, temos que se a sequência de pontos obtida se aproxima do ponto fixo ou órbita periódica, então essa solução assintótica é estável. Por outro lado, se a sequência de pontos se afasta, o ponto fixo ou órbita periódica é instável (MONTEIRO, 2006).

Outra forma de analisar a estabilidade das soluções de um sistema é linearizando as equações em torno da solução assintótica. Nesse caso, o valor do módulo dos autovalores da matriz Jacobiana, apresentada na Equação (4) determina a estabilidade. Considerando um sistema de tempo discreto com espaço de fases bidimensional e sendo λ_1 e λ_2 os autovalores do sistema, se $|\lambda_1| < 1 < |\lambda_2|$ o ponto fixo é chamado de sela; se $|\lambda_{1,2}| > 1$, o ponto fixo é um nó instável; se $|\lambda_{1,2}| < 1$ o ponto fixo é um nó assintoticamente estável. Já para sistemas de tempo contínuo, temos que um autovalor positivo representa um ponto instável e um autovalor negativo representa um ponto estável. Na Figura 5, temos uma ilustração das definições de estabilidade para um ponto fixo.

Figura 5 – Análise de estabilidade de um ponto fixo.



fonte: Produção do próprio autor.

3.1.3 Dinâmica caótica

Por fim, temos que sistemas dinâmicos complexos podem exibir dinâmica caótica, caracterizada por certa instabilidade, também chamada de sensibilidade às condições iniciais. Essa sensibilidade torna o sistema não previsível na prática, pois soluções que se iniciam em condições iniciais diferentes, porém muito próximas, se separam exponencialmente ao longo do tempo.

Se todas as condições e equações que regem a dinâmica do sistema forem conhecidas com exatidão, o exato resultado no futuro pode ser previsto com exatidão, ou seja, o sistema é dito determinístico. Porém, como as condições atuais exatas muitas vezes não são conhecidas, o comportamento futuro também é difícil de prever. Assim, a alta sensibilidade às condições iniciais dá ao sistema não linear características que muitas vezes são confundidas com as características de sistemas aleatórios. Contudo o comportamento futuro de sistemas caóticos determinísticos pode ser determinados se as condições iniciais e as regras matemáticas que regem a dinâmica forem perfeitamente conhecidas, diferentemente do que ocorre com um sistema aleatório, cujo comportamento futuro não pode ser previsto.

Em sistemas dinâmicos com comportamento caótico, o movimento é organizado em torno de órbitas periódicas instáveis, portanto a determinação de tais órbitas é de fundamental importância para a caracterização de sistemas dinâmicos (BOCCALETTI et al., 2000).

3.2 MÉTODO DE RUNGE-KUTTA

Problemas físicos, matemáticos e de engenharia, assim como o método de bloqueio, podem ser resolvidos de diversas formas, cada uma delas com seu tempo, custo, qualidade e erro associado. Os métodos de resolução podem ser analíticos, experimentais ou numéricos, ou ainda combinações entre eles. Nesta seção serão apresentados os fundamentos e conceitos básicos dos métodos de resolução utilizados neste trabalho.

Em geral, os métodos analíticos são utilizados para problemas com menor complexidade, por serem mais simples e, geralmente, oportunizarem resolução manual ou com auxílio mínimo de calculadoras ou computadores. Por esse motivo, o cálculo analítico nem sempre pode ser aplicado para resolução de problemas de engenharia que envolvam equação não lineares e modelos multidimensionais que exigem resoluções complexas. Contudo, as soluções analíticas podem ser utilizadas como base para o entendimento do sistema, desenvolvimento de métodos numéricos e para a verificação de códigos computacionais.

Os métodos experimentais consistem na criação de protótipos, em escala real ou reduzida, a fim de realizar simulações nas condições de operação do sistema. Como vantagem desse método temos a qualidade dos resultados obtidos, uma vez que foram obtidos de simulações reais. Como desvantagens temos o alto investimento para construção dos protótipos e infraestrutura, a longa duração para construção e testes e por fim a dificuldade para criar as condições reais do sistema de forma precisa.

Por último, temos os métodos numéricos. Neste caso temos que soluções aproximadas são encontradas utilizando algoritmos que dividem um problema complexo em diversas operações mais simples. Com a evolução na capacidade de processamento dos computadores, tornou-se popular o uso desse método para solucionar problemas reais, para os quais resoluções manuais podem ser impossíveis, imprecisas, com alto investimento e longo tempo de execução (BOYCE; DIPRIMA, 2010).

Neste trabalho, em específico, foi utilizado o método numérico de Runge-Kutta para integração numérica das equações que regem a dinâmica do sistema. Criado pelos matemáticos Carl Runge e Martin Wilhelm Kutta por volta de 1900, o método iterativo fornece soluções aproximadas para equações diferenciais ordinárias.

O método de Runge-Kutta utiliza como base a comparação com um polinômio de Taylor para eliminar o cálculo das derivadas. O método pode ser entendido como um aperfeiçoamento do método de Euler, onde a estimativa do valor de y_{n+1} é realizada com o valor de y_n e com a derivada no ponto x_n . Já no método de Runge-Kutta, ao invés de utilizar apenas um ponto no intervalo $[x_n, x_{n+1}]$ são utilizados n pontos, onde n também define qual será a ordem do método.

O método de Runge-Kutta é provavelmente um dos métodos mais populares e o de quarta ordem, conhecido como RK4, é um dos mais utilizados para obter soluções aproximadas de problemas de

valor inicial.

Considerando um problema de valor inicial:

$$y' = f(t, y), \quad y(t_0) = y_0, \quad (5)$$

o método RK4 para este problema é dado por:

$$y_{n+1} = y_n + h/6 \cdot (k_1 + 2 \cdot k_2 + 2 \cdot k_3 + k_4) \quad (6)$$

$$t_{n+1} = t_n + h, \quad (7)$$

onde k_1 , k_2 , k_3 e k_4 são valores de inclinações intermediárias utilizadas para calcular uma média ponderada de inclinações, a qual resultará em uma inclinação final do método. Esses valores de inclinação são dados por:

$$k_1 = f(t_n, y_n) \quad (8)$$

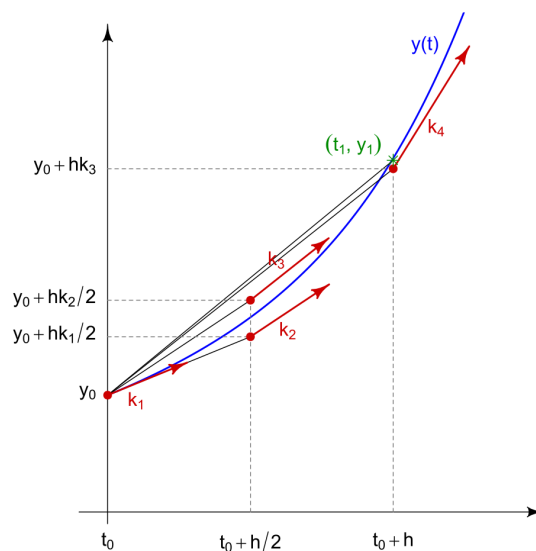
$$k_2 = f(t_n + h/2, y_n + h/2 \cdot k_1) \quad (9)$$

$$k_3 = f(t_n + h/2, y_n + h/2 \cdot k_2) \quad (10)$$

$$k_4 = f(t_n + h, y_n + h \cdot k_3). \quad (11)$$

A representação gráfica do método pode ser vista na Figura 6.

Figura 6 – Análise gráfica do método RK4.



fonte: <https://en.wikipedia.org/wiki/Runge%E2%80%93Kutta_methods>.

3.3 EXPOENTE DE LYAPUNOV

Para caracterizar o comportamento caótico do sistema de bloqueio iremos utilizar o expoente máximo de Lyapunov que foi definido no século XIX, pelo matemático russo Aleksandr Lyapunov, que deu origem ao desenvolvimento de um método para medir o afastamento de duas trajetórias iniciadas em duas condições iniciais próximas (STROGATZ, 1994; PARKER; CHUA, 1989). Para o cálculo do expoente, temos que:

$$\varepsilon(n) = \varepsilon e^{\lambda n}, \quad (12)$$

onde ε é a distância entre dois pontos ao longo das trajetórias e λ é o expoente de Lyapunov. De acordo com essa equação, se $\lambda > 0$ a distância entre as duas trajetórias aumenta, já se $\lambda < 0$ a distância diminui.

Seguindo com a Equação 12, podemos aproximar a distância da n -ésima medida como

$$\varepsilon(n) \approx f^n(x + \varepsilon) - f^n(x), \quad (13)$$

onde $f^n(x)$ é a função do mapa na n -ésima iteração.

Tomando o logaritmo e isolando o expoente de Lyapunov:

$$\lambda \approx \frac{1}{n} \ln \left| \frac{df^n(x)}{dx} \right|. \quad (14)$$

Por fim, aplicando a regra da cadeia na derivada, a propriedade de logaritmo da multiplicação e tomando o limite para n tendendo a infinito, temos:

$$\lambda = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} \ln |f'(x_i)|. \quad (15)$$

Desta forma, um expoente de Lyapunov positivo indica um aumento na distância dos pontos e conseqüentemente uma expansão no espaço de fase, ou seja, para um valor positivo temos uma trajetória caótica.

Se o sistema for um fluxo, ou seja, descrito por equações diferenciais ordinárias, o procedimento é o mesmo, exceto que o expoente resultante é dividido também pelo tamanho do passo da iteração, de modo que tenha unidades de 1/segundos em vez de 1/iterações (SPROTT, 2003).

Outra observação muito importante é que normalmente é preciso milhões de iterações das equações diferenciais para obter um resultado bom com cerca de dois dígitos significativos (SPROTT, 2003).

4 EQUAÇÕES DO SISTEMA

Após apresentados os conceitos básicos de sistemas ópticos e de sistemas dinâmico, vamos apresentar as equações que descrevem a dinâmica de um sistema OIL. Tais equações são, basicamente, as equações de taxa clássicas de um laser com algumas pequenas modificações. Essas modificações são decorrentes dos termos adicionais de intensidade de luz (ou do número de fótons) e deslocamento de fase para descrever a influência do sinal do laser mestre.

As equações de taxa do sistema são:

$$\frac{dA(t)}{dt} = \frac{1}{2} \cdot g \cdot [N(t) - N_{th}] \cdot A(t) + \kappa \cdot A_{inj} \cdot \cos \phi(t) \quad (16)$$

$$\frac{d\phi(t)}{dt} = \frac{\alpha}{2} \cdot \{g \cdot [N(t) - N_{th}]\} - \kappa \cdot \frac{A_{inj}}{A(t)} \cdot \sin \phi(t) - \Delta\omega_{inj} \quad (17)$$

$$\frac{dN(t)}{dt} = J - \gamma_N \cdot N(t) - \{\gamma_P + g \cdot [N(t) - N_{th}]\} \cdot A(t)^2, \quad (18)$$

onde os termos $(\kappa \cdot A_{inj} \cdot \cos \phi(t))$ e $(-\kappa \cdot (A_{inj}/A(t)) \cdot \sin \phi(t) - \Delta\omega_{inj})$ são as modificações das equações clássicas do laser implementadas por S. Kobayashi e T. Kimura (KOBAYASHI; KIMURA, 1981). A identificação de cada uma das variáveis é dada na Tabela 1.

Tabela 1 – Identificação dos parâmetros do sistema óptico

Símbolo	Parâmetro	Unidade
$A(t)$	Amplitude de campo normalizada	[V/m]
A_{inj}	Amplitude do campo da luz injetada	[V/m]
g	Coefficiente de ganho do laser	[1/s]
J	Fluxo de elétrons	[elétrons/s]
$N(t)$	Número de portadores no laser escravo	-
N_{th}	Número limite de portadores	-
α	Fator de aprimoramento de largura de linha	-
κ	Coefficiente de acoplamento	[1/s]
$\phi(t)$	Diferença de fases dos lasers escravo e mestre	[rad]
$\Delta\omega_{inj}$	Frequência de dessintonização entre mestre e escravo	[GHz]
γ_N	Taxa de recombinação de portadores	[1/s]
γ_P	Taxa de decaimento de fótons	[1/s]

fonte: Produção do próprio autor.

O sistema de interesse foi resolvido com base no trabalho de Lau (2006). Para determinar as soluções das Equações (16) a (18) é necessário encontrar as soluções das três equações diferenciais acopladas, o que pode ser feito numericamente. Contudo, uma solução mais fácil e rápida consiste em resolver três equações diferenciais em estado estacionário: uma para a magnitude do campo, fase e número da portadora, definidos aqui como A_0 , ϕ_0 e N_0 , respectivamente.

Para resolver o sistema em estado estacionário, iniciamos calculando a magnitude do campo de *free-running* utilizando a Equação (18). Neste caso, considerando a situação de *free-running*, podemos assumir $\Delta N_0 = N_0 - N_{th} = 0$. Desta forma, obtemos:

$$A_{fr}^2 = \frac{J - \gamma_N \cdot N_{th}}{\gamma_P}. \quad (19)$$

Resolvendo os valores de estado estacionário do laser bloqueado por injeção, obtemos:

$$A_0^2 = \frac{A_{fr}^2 - \frac{\gamma_N}{\gamma_P} \cdot \Delta N_0}{1 + \frac{g \cdot \Delta N_0}{\gamma_P}} \quad (20)$$

$$\phi_0 = \sin^{-1} \left\{ -\frac{\Delta \omega_{inj}}{\kappa \sqrt{1 + \alpha^2}} \cdot \frac{A_0}{A_{inj}} \right\} - tg^{-1} \alpha \quad (21)$$

$$\Delta N_0 = -\frac{2\kappa}{g} \cdot \frac{A_{inj}}{A_0} \cdot \cos \phi_0. \quad (22)$$

Com essas equações, podemos escolher uma razão de injeção (R_{int}):

$$R_{int} = \left(\frac{A_{inj}}{A_{fr}} \right)^2, \quad (23)$$

e uma frequência de dessintonização, $\Delta \omega_{inj}$, e usá-las para resolver as Equações (20) a (22).

Contudo, ainda há uma dependência entre as três equações, ou seja, a magnitude do campo depende do número de portadores, que depende da fase e da magnitude, que por sua vez depende da magnitude do campo. Para simplificar, podemos adotar uma outra técnica de resolução, na qual assumimos um dos valores e calculamos os demais. Uma das opções mais lógicas para esse sistema é assumir um valor de fase (ϕ_0), porque sabemos que os limites da fase ao longo do intervalo de bloqueio são aproximadamente $\cot^{-1} \alpha$ a $-\frac{\pi}{2}$, conforme demonstrado na Equação (33). Como dito anteriormente, também assumimos um valor de razão de injeção, e com isso, substituindo a Equação (22) na Equação (20), obtemos:

$$A_0^3 - \left[\frac{2\kappa}{\gamma_P} \cdot A_{inj} \cdot \cos \phi_0 \right] \cdot A_0^2 - A_{fr}^2 \cdot A_0 - \frac{\gamma_N}{\gamma_P} \cdot \frac{2\kappa}{g} \cdot A_{inj} \cdot \cos \phi_0 = 0. \quad (24)$$

Então, resolvendo a Equação (22) obtemos ΔN_0 e reorganizamos a Equação (21) para encontrar o valor de $\Delta \omega_{inj}$ dado por:

$$\Delta \omega_{inj} = -\kappa \cdot \sqrt{1 + \alpha^2} \cdot \frac{A_{inj}}{A_0} \cdot \sin(\phi_0 + tg^{-1} \alpha). \quad (25)$$

Outra maneira de resolver as equações do sistema OIL é por meio de soluções dinâmicas. As Equações (16) podem ser linearizadas aplicando uma perturbação de pequeno sinal em torno de suas soluções de estado estacionário, tal que:

$$X(t) = X_0 + \Delta X \cdot exp(st), \quad (26)$$

onde X representa as variáveis de estado (A , ϕ , N e J); X_0 as variáveis de estado estacionário (A_0 , ϕ_0 , N_0 e J_0); e ΔX representa as magnitudes de pequenos sinais (ΔA , $\Delta \phi$, ΔN e ΔJ).

Com isso, podemos reescrever as equações em forma matricial:

$$\begin{bmatrix} m_{AA} + s & m_{A\phi} & m_{AN} \\ m_{\phi A} & m_{\phi\phi} + s & m_{\phi N} \\ m_{NA} & 0 & m_{NN} + s \end{bmatrix} \cdot \begin{bmatrix} \Delta A \\ \Delta \phi \\ \Delta N \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \Delta J \end{bmatrix}, \quad (27)$$

onde os termos da matriz são:

$$\begin{aligned} m_{AA} &= z \cdot \cos \phi_0 \\ m_{A\phi} &= z \cdot A_0 \cdot \sin \phi_0 \\ m_{AN} &= -1/2 \cdot g \cdot A_0 \\ m_{\phi A} &= -z \cdot \sin \phi_0 / A_0 \\ m_{\phi\phi} &= z \cdot \cos \pi_0 \\ m_{\phi N} &= -1/2 \cdot \alpha \cdot g \\ m_{NA} &= 2 \cdot A_0 \cdot (\gamma_P - 2 \cdot z \cdot \cos \phi_0) \\ m_{NN} &= \gamma_N + g \cdot A_0^2. \end{aligned} \quad (28)$$

Invertendo a matriz, a magnitude da resposta em frequência é dada por:

$$H(s) = \frac{\Delta A}{\Delta I} = M \cdot \frac{s - Z}{s^3 + A \cdot s^2 + B \cdot s + C}, \quad (29)$$

onde

$$\begin{aligned} A &= m_{AA} + m_{\phi\phi} + m_{NN} \\ B &= m_{AA} \cdot m_{\phi\phi} + m_{AA} \cdot m_{NN} + m_{\phi\phi} \cdot m_{NN} - m_{A\phi} \cdot m_{\phi A} - m_{AN} \cdot m_{NA} \\ C &= m_{AA} \cdot m_{\phi\phi} \cdot m_{NN} + m_{A\phi} \cdot m_{\phi N} \cdot m_{NA} - m_{A\phi} \cdot m_{\phi A} \cdot m_{NN} - m_{AN} \cdot m_{NA} \cdot m_{\phi\phi} \\ Z &= (m_{A\phi} \cdot m_{\phi N} - m_{AN} \cdot m_{\phi\phi}) / m_{AN} \\ M &= -m_{AN} \end{aligned} \quad (30)$$

Após solucionar as equações do sistema, devemos descrever as condições de estabilidade que delimitam a faixa de bloqueio. A primeira condição pode ser extraída da Equação (21), onde só conseguimos uma solução real para fase se o valor em módulo do arco seno não for maior que 1. Com isso, podemos escrever a primeira condição de estabilidade:

$$-\frac{\pi}{2} - \tan^{-1} \alpha \leq \phi \leq \frac{\pi}{2} - \tan^{-1} \alpha. \quad (31)$$

Para a segunda condição, devemos analisar a Equação (22). Nela temos que o número de portadores não deve ultrapassar o limite de *threshold*, ou haverá um ganho instável. Portanto,

$$-\frac{\pi}{2} \leq \phi. \quad (32)$$

Juntando as duas primeiras condições em uma única equação, temos:

$$-\frac{\pi}{2} \leq \phi \leq \cot^{-1} \alpha. \quad (33)$$

Usando essas condições na Equação (25), obtemos também os limites para $\Delta\omega_{inj}$:

$$-\kappa\sqrt{1 + \alpha^2} \cdot \frac{A_{inj}}{A_0} \leq \Delta\omega_{inj} \leq \kappa\sqrt{1 + \alpha^2} \cdot \frac{A_{inj}}{A_0}. \quad (34)$$

Por fim, a última condição de estabilidade, pode ser encontrada com base na solução dinâmica do sistema dada pela Equação (29). Assim, temos que, quando a parte real das raízes do denominador forem positivas, a solução se torna instável (LAU, 2006).

Esses limites podem ser calculados facilmente e terão grande importância ao longo desse trabalho, pois neles serão baseados os estudos para caracterização do caos.

5 RESULTADOS

Neste capítulo serão apresentados os resultados obtidos ao longo do trabalho, junto com uma descrição dos procedimentos necessários para obtê-los. Assim, apresentamos os parâmetros do sistema, o passo a passo para a obtenção do mapa de bloqueio do sistema e a resolução das equações do sistema por meio de integração do sistema utilizando o método numérico RK4. Por fim, apresentamos o cálculo do expoente de Lyapunov para certas condições iniciais escolhidas.

Para iniciar a análise das equações do sistema de laser, apresentado na Seção (4), utilizaremos os parâmetros da Tabela (2). Esses valores serão utilizados para obtenção dos resultados que serão mostrados mais à frente.

Tabela 2 – Parâmetros iniciais

Parâmetro/Grandeza	Notação	Valor	Unidade
Coefficiente de ganho do laser	g	5666.67	[1/s]
Número limite de portadores	N_{th}	$2.214 \cdot 10^8$	-
Taxa de decaimento de fótons	γ_P	$3.333 \cdot 10^{11}$	[1/s]
Taxa de recombinação de portadores	γ_N	$1 \cdot 10^9$	[1/s]
Tempo de <i>Round-trip</i>	τ_{IN}	$7 \cdot 10^{-12}$	[s]
Reflectividade dos espelhos	r_0	0.548	-
Fator de largura de linha	α	3	-
Fluxo de elétrons	J	$1.045 \cdot 10^{18}$	[elétrons/s]
Tempo máximo integração	t_{fim}	100	[ns]
Passo de tempo	dt	1/500	[ns]

fonte: (LAU, 2006)

5.1 MAPA DE BLOQUEIO

Com os valores dos parâmetros iniciais, podemos obter o mapa de bloqueio do sistema. Para isso desenvolvemos um passo a passo para obtenção do mapa, conforme mostrado a seguir. Com os valores dos parâmetros iniciais, podemos obter o mapa de bloqueio do sistema. Para isso desenvolvemos um passo a passo para obtenção do mapa, conforme mostrado a seguir. Em complemento foi criado um script em Python, disponível no Apêndice A.

1. Escolher os parâmetros iniciais, conforme Tabela (2);
2. Calcular A_{fr} , utilizando a Equação (19);
3. Iniciar o loop para varredura na razão de injeção (R_{inj}). Neste momento não utilizamos este parâmetro em dB . Faixa utilizada: -2 a 0.5, com passo de 0,00625 com variação logarítmica (400 pontos);
4. Iniciar o loop para varredura em ϕ . Faixa utilizada: $-\pi/2$ a $\arctan(1/\alpha)$, conforme limite encontrado na Equação (33);
5. Escolher um valor inicial de ϕ_0 ;

$$\phi_0 = \pi/2 \cdot \sin(\phi)$$

6. Calcular o valor inicial de κ ;

$$\kappa = \frac{1}{\tau_{IN}} \cdot \frac{1 - r_0^2}{r_0} \cdot R_{inj}$$

7. Determinar as raízes do polinômio da Equação (24);

8. Iniciar o loop para verificar todas as raízes encontradas e se atendem as seguintes características:

[8.1] Se a parte imaginária for menor ou igual a zero;

[8.2] Se for maior ou igual que A_{fr} ;

9. Caso atenda as características do Passo 8, calcular:

[9.1] ΔN_0 , utilizando a Equação (22);

[9.2] $\Delta\omega_{inj}$, utilizando a Equação (25);

[9.3] Valores dos parâmetros definidos em (28);

[9.4] Raízes do polinômio do denominador da Equação (29);

[9.5] Início do loop para verificar todas as raízes encontradas e se atendem as seguintes características:

[9.5.1] Se parte real maior que zero e parte imaginária diferente de zero, o ponto é INSTÁVEL;

[9.5.2] Caso contrário o ponto é ESTÁVEL;

10. Após determinar se o ponto é instável ou estável, podemos plotar o gráfico ou salvar um arquivo de dados com os resultados obtidos. Para isso, utilizamos:

[10.1] Eixo X: razão de injeção (R_{inj}), realizando a conversão para dB ;

$$IR [dB] = 20 \cdot \log_{10}(R_{inj})$$

[10.2] Eixo Y: Frequência de dessintonização $\Delta\omega_{inj}$, realizando a conversão para GHz ;

$$DF [GHz] = \frac{\Delta\omega_{inj}}{2\pi} \cdot (10^{-9})$$

Além do mapa de bloqueio devemos também encontrar os limites analíticos do sistema. Para isso realizamos os seguintes passos:

1. Repetir os Passos de 1 à 6, do script anterior;

2. Se $\phi = -(\pi/2)$ - Limite superior:

[2.1] Eixo X: razão de injeção (R_{inj}), realizando a conversão para dB ;

$$IR_s [dB] = 20 \cdot \log_{10}(R_{inj})$$

[2.2] Eixo Y: Frequência de dessintonização $\Delta\omega_{inj}$, realizando a conversão para GHz ;

$$DF_s [GHz] = \frac{\kappa \cdot \sqrt{1 + \alpha^2}}{2\pi \cdot 10^{-9}}$$

3. Se $\phi = \arctan(1 - \alpha)$ - Limite inferior:

[3.1] Eixo X: razão de injeção (R_{inj}), realizando a conversão para dB ;

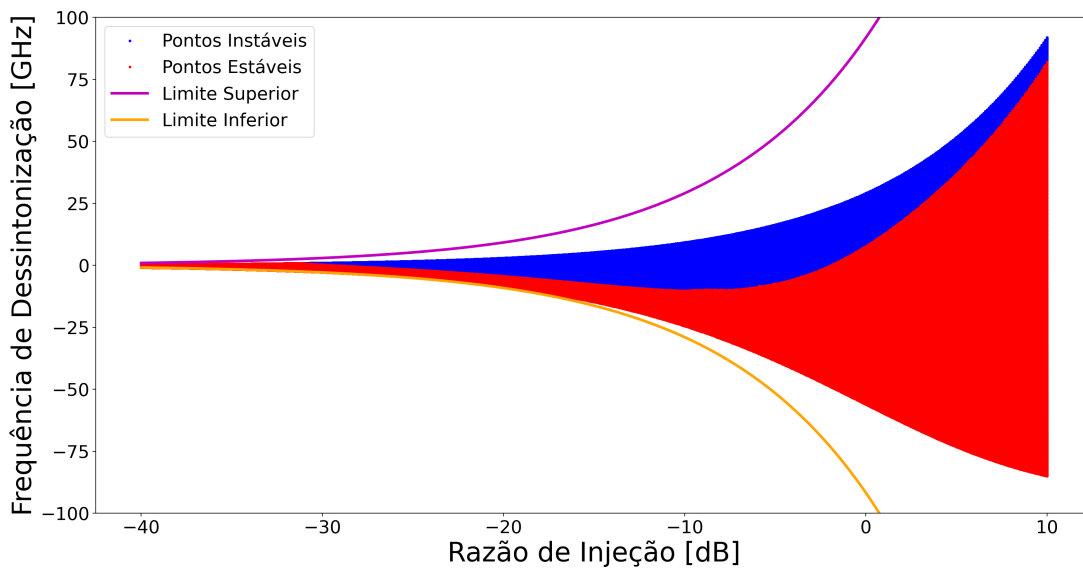
$$IR_i [dB] = 20 \cdot \log_{10}(R_{inj})$$

[3.2] Eixo Y: Frequência de dessintonização $\Delta\omega_{inj}$, realizando a conversão para GHz ;

$$DF_i [GHz] = -\frac{\kappa \cdot \sqrt{1 + \alpha^2}}{2\pi \cdot 10^{-9}}$$

Utilizando o passo a passo como base, calculamos a frequência de dessintonização em função da razão de injeção e aplicamos as condições de estabilidade. Os resultados podem ser vistos na Figura 7, onde podemos observar as linhas roxa e laranja, que foram calculadas analiticamente por meio das Equações (33) e (34). Também podemos observar toda a área acima da linha roxa ou abaixo da laranja que é considerada uma área onde não é possível "travar" a frequência do laser escravo.

Figura 7 – Mapa de bloqueio e estabilidade.



fonte: Produção do próprio autor.

No mapa de bloqueio podemos ver uma região em azul e outra em vermelho. A região vermelha representa a região estável, definida pela última condição de estabilidade, ou seja, as raízes do denominador da Equação (29) têm a parte real negativa. Já na região azul temos a parte instável do sistema, ou seja, apresentam a parte real das raízes positivas.

Ainda neste mapa podemos definir também algumas fronteiras do sistema. A primeira fronteira corresponde a fronteira dinâmica, também conhecida como bifurcação de Hopf, representada no mapa pela intersecção das regiões estáveis e instáveis. Nela o ponto de equilíbrio perde a estabilidade e surge um ciclo limite, ou seja, é um tipo de bifurcação que liga equilíbrio a movimento periódico. A segunda é a fronteira estática, neste caso também chamada de bifurcação sela-nó. Ela é representada no mapa de bloqueio pela intersecção das regiões instáveis e estáveis com a região onde não há bloqueio no

sistema (área branca). Nesse tipo de bifurcação temos que há colisão de um ponto estável com um ponto de sela, ocasionando assim a mudança de estabilidade.

Outra observação que pode ser realizada é sobre o limite inferior da região estável. Este limite é definido pela razão entre a amplitude inicial e a amplitude de *free-running* do laser escravo, que deve ser maior que um.

Por fim, vemos que quanto maior a razão de injeção maior pode ser a frequência de dessintonização e maior é a faixa da região estável, ou seja, quanto maior a relação entre a potência do laser mestre e a potência do laser escravo, maior poderá ser a diferença de frequência entre os lasers sem que haja a dessintonização do sistema. Isso é uma característica de extrema importância para casos onde esse sistema é utilizado para modulação em taxas de transmissão mais altas.

5.2 RESOLUÇÃO DAS EQUAÇÕES DO SISTEMA E CÁLCULO DO EXPOENTE DE LYAPUNOV

O próximo passo para seguir com a análise, consiste em resolver as equações diferenciais que regem o sistema. Para isso devemos utilizar uma condição inicial, escolhida com base no mapa de bloqueio. Após escolher o ponto inicial, resolver-se o problema de valor inicial utilizando o método numérico RK4 a fim de encontrar a solução do sistema.

Neste caso devemos seguir os passos abaixo, que mostram como foi feita a implementação do método de Runge-Kutta para o sistema de lasers. Após escolher o ponto inicial, resolver-se o problema de valor inicial utilizando o método numérico RK4 a fim de encontrar a solução do sistema. O script em Python implementando o procedimento descrito está disponível no Apêndice B.

1. Escolher das variáveis iniciais, conforme Tabela (2);
2. Calcular de A_{fr} , utilizando a Equação (19);
3. Definir as condições iniciais ($\Delta\omega_{inj}$ e R_{inj}), com base no mapa de bloqueio;
4. Definir $A_{inj} = A_{fr}$;
5. Calcular o valor inicial de κ , conforme a equação abaixo;

$$\kappa = \frac{1}{t_{IN}} \cdot \frac{1 - r_0^2}{r_0} \cdot R_{inj}$$

6. Calcular valor de ϕ_0 conforme a Equação (21);
7. Calcular valor de N_0 conforme a Equação (22);
8. Calcular valor de A_0 conforme a Equação (20);
9. Iniciar loop com método RK4, com os seguintes parâmetros:
 - [9.1] Tempo inicial = 0 e Tempo Final = $t_{fim} = 50ns$, com passos $h = dt = 1/500GHz$;
 - [9.2] Valor inicial $y = [A_0, N_0, \phi_0]$;

Por fim, calculamos os expoentes de Lyapunov das soluções do sistema com base no método descrito em (SPROTT, 2003). O script em Python implementando o procedimento descrito está disponível no Apêndice C.

As etapas são:

1. Gerar uma condição inicial;
2. Iterar essa condição inicial por 1000 ns;
3. Selecionar um ponto próximo, separado por uma distância ($d_0 = 10^{-6}$);
4. Iterar os dois pontos por várias iterações. Nessas interações são calculadas a distância entre os pontos (d_1), depois dividir essa distância por d_0 e calcular $\log(d_1/d_0)$, sendo esse o expoente para aquela iteração. Para um melhor resultado do expoente, calculamos a média de todos os expoentes calculados até a iteração atual;
5. Atualizar o valor do ponto próximo a trajetória da condição inicial, sendo este calculado da seguinte forma:

$$\begin{aligned}
 A_2 &= A_1 + (d_0/d_1) \cdot (A_2 - A_1) \\
 \phi_2 &= \phi_1 + (d_0/d_1) \cdot (\phi_2 - \phi_1) \\
 N_2 &= N_1 + (d_0/d_1) \cdot (N_2 - N_1).
 \end{aligned}
 \tag{35}$$

Os procedimentos descritos foram utilizados para realizar várias simulações utilizando diferentes condições iniciais. Para fins de comparação serão apresentados resultados para os quais as condições iniciais foram escolhidas variando a razão de injeção, mas deixando fixo a frequência.

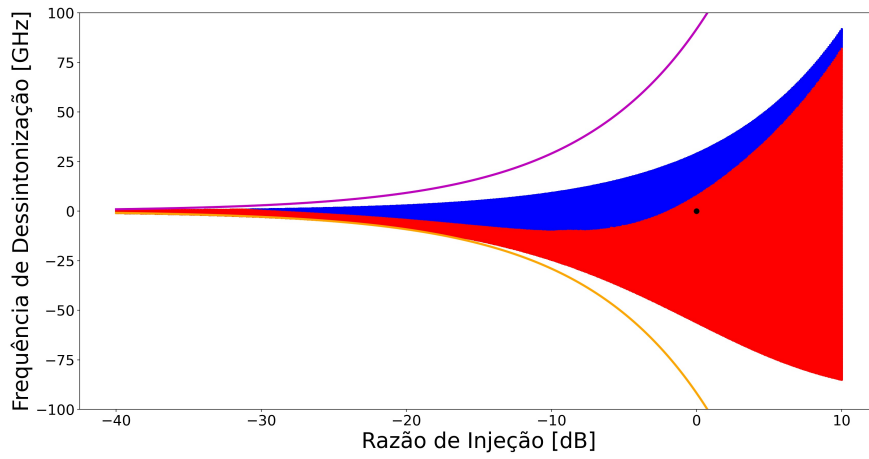
Para a primeira condição inicial escolhida foi utilizada uma diferença de frequência para dessintonização de igual a zero e uma razão de injeção de 0 dB. Essa condição inicial pode ser vista diretamente no mapa de bloqueio conforme mostrado na Figura Figura 8(a). No mapa de bloqueio podemos ver ainda que esse valor de condição inicial está na área vermelha, ou seja, região estável do sistema.

Nas Figuras Figura 8(b), Figura 8(c) e Figura 8(d), vemos os resultados da integração numérica do sistema utilizando o método RK4. Neste caso podemos verificar também que a condição inicial, com o passar do tempo, resulta em uma solução estável e fixa, chamada de ponto fixo em sistemas dinâmicos ou podemos dizer que o laser está em um estado sincronizado estável.

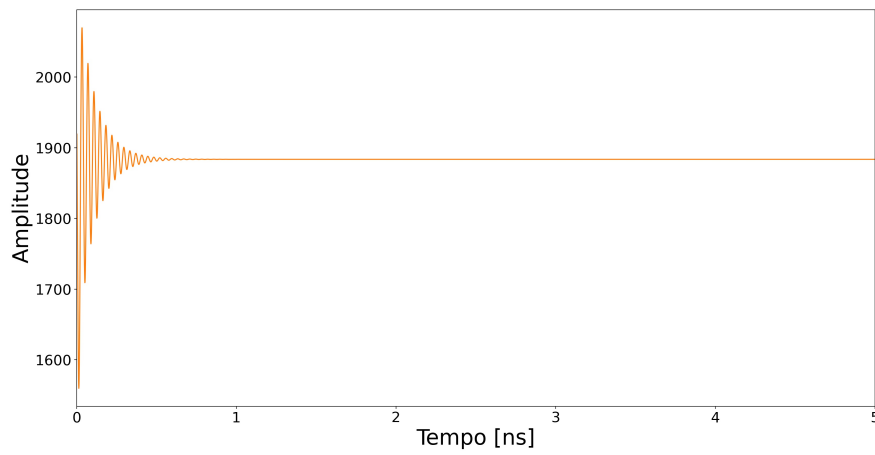
Por fim, na Figura Figura 8(e) que mostra o cálculo do expoente máximo de Lyapunov ao longo da janela de integração. Conforme esperado, para esse caso, foi encontrado um expoente máximo de aproximadamente -9.06. Conforme citado na Seção 3.3, temos que um valor negativo do expoente indica um comportamento estável ou periódico.

Figura 8 – Resultados para diferença de frequência para dessintonização nula e razão de injeção de 0 dB.

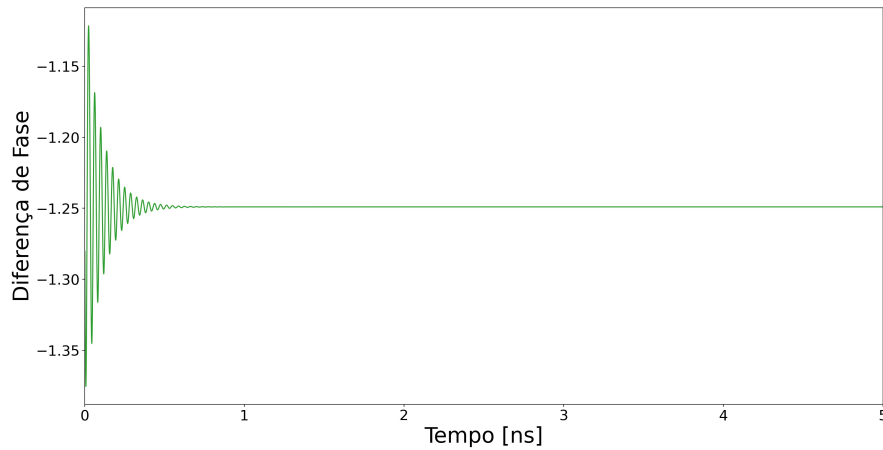
(a) Condição inicial escolhida no mapa de bloqueio.



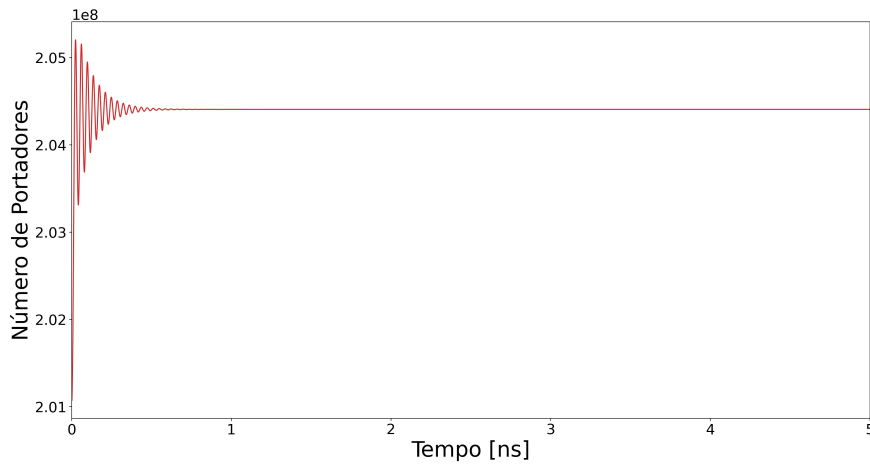
(b) Amplitude do campo para diferença de frequência para dessintonização nula e razão de injeção de 0 dB.



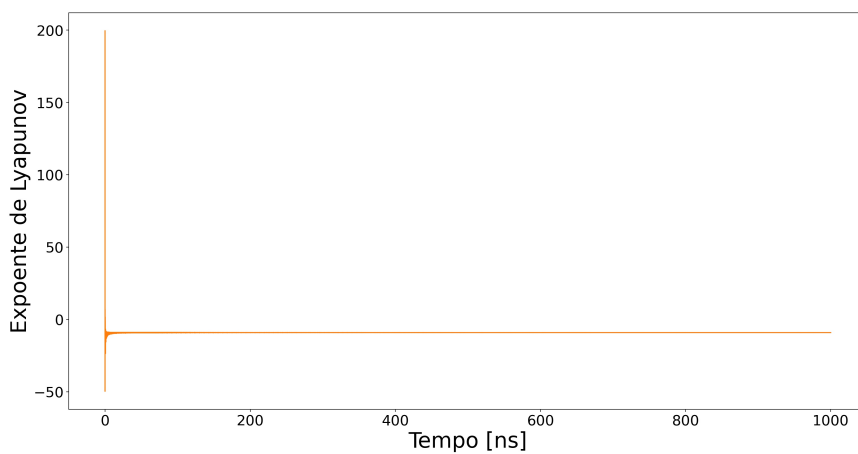
(c) Diferença de fases para diferença de frequência para dessintonização nula e razão de injeção de 0 dB.



(d) Número de portadores para diferença de frequência para dessintonização nula e razão de injeção de 0 dB.



(e) Expoente máximo de Lyapunov para diferença de frequência para dessintonização nula e razão de injeção de 0 dB.



fonte: Produção do próprio autor.

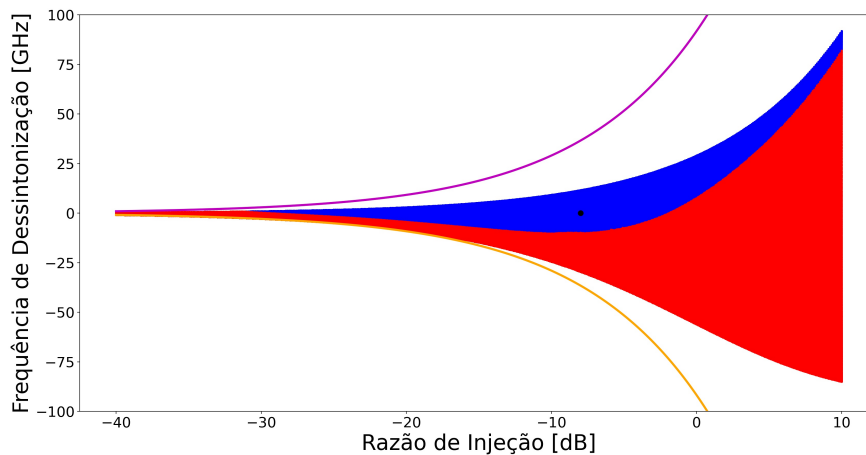
Para segunda condição inicial utilizada, temos novamente uma diferença de frequência para dessintonização nula, mas uma razão de injeção de -8 dB, essa condição inicial pode ser vista diretamente no mapa de bloqueio mostrado na Figura 9(a). Ao verificar essa figura temos que a condição inicial já não está mais na região estável do sistema.

Nas Figuras 9(b), 9(c) e 9(d), vemos os resultados da integração numérica do sistema utilizando o método RK4. Neste caso, a solução converge para o que a princípio parece ser uma órbita periódica de período dois, contudo ao ampliarmos os gráficos podemos verificar que não se trata de um período dois e sim de um período oito.

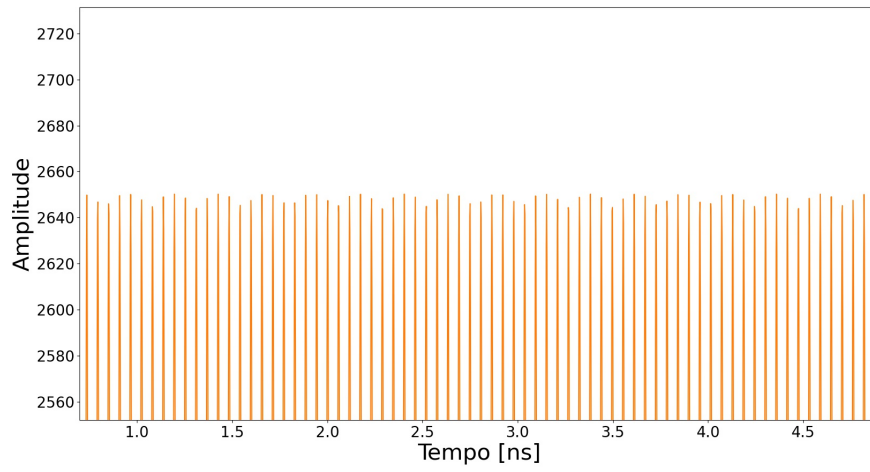
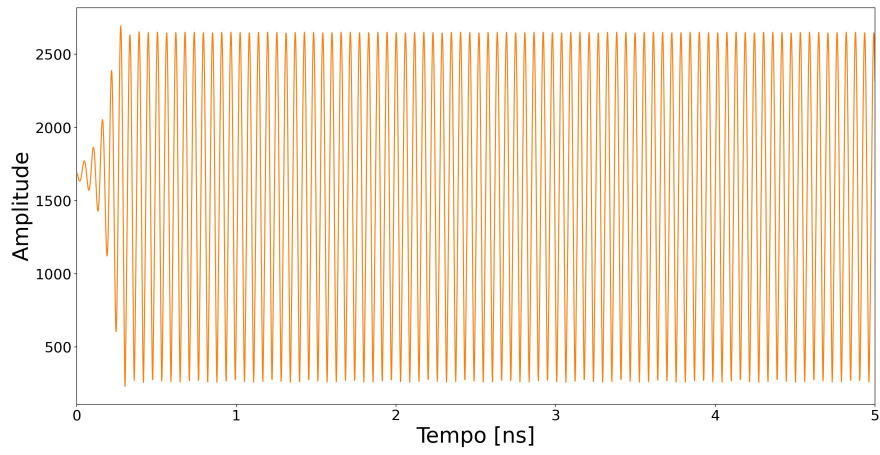
Por fim, na Figura 9(e) observamos um valor de expoente de -7.90 . Com isso podemos validar que na região onde a condição inicial se encontra, o sistema não apresenta ainda comportamento caótico, apesar de estar em uma região considerada instável pelas condições de estabilidade vistas na Seção 4.

Figura 9 – Resultados para diferença de frequência para dessintonização nula e razão de injeção de -8 dB.

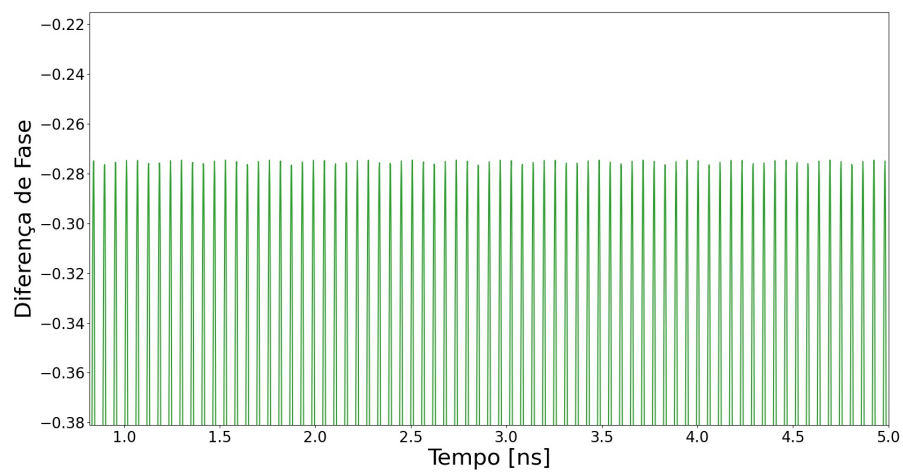
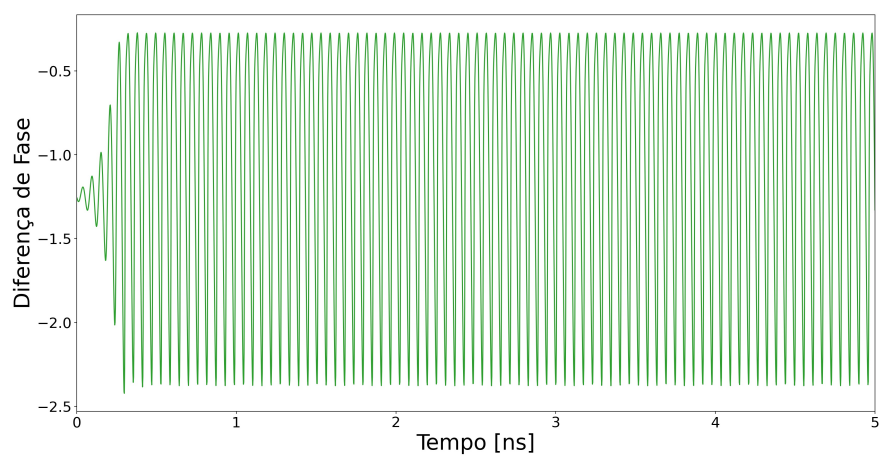
(a) Condição inicial escolhida no mapa de bloqueio.



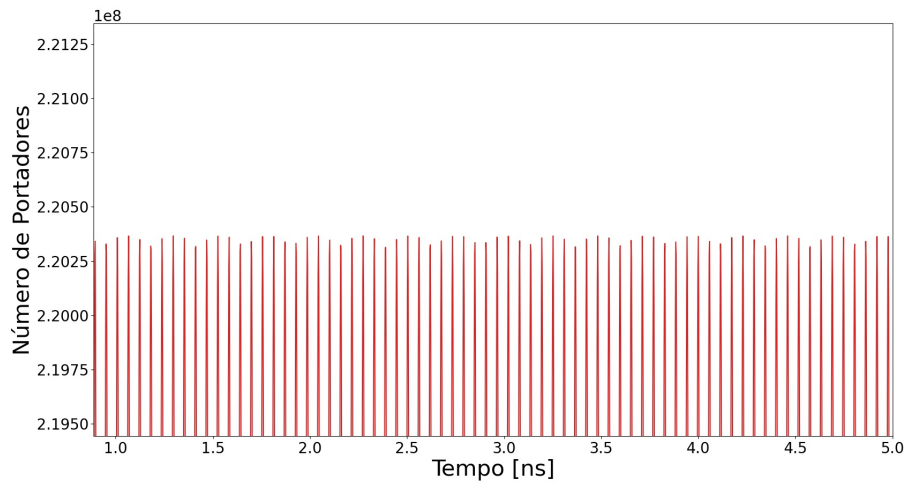
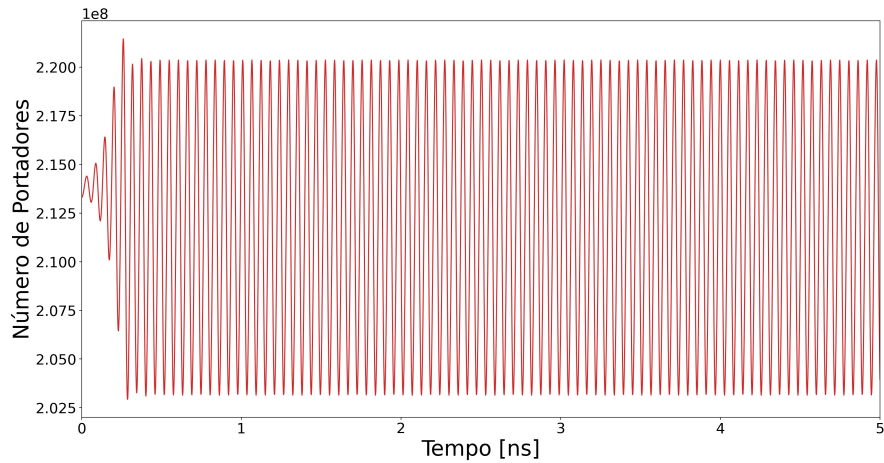
(b) Amplitude do campo para diferença de frequência para dessintonização nula e razão de injeção de -8 dB.



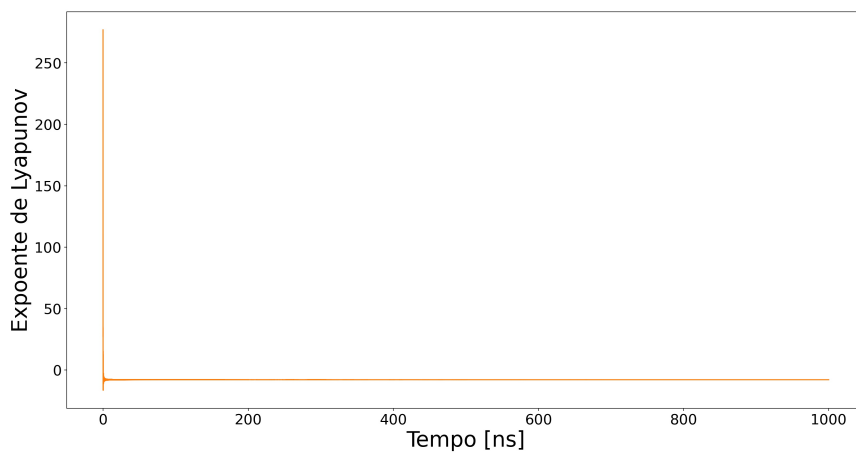
(c) Diferença de fases para diferença de frequência para dessintonização nula e razão de injeção de -8 dB.



(d) Número de portadores para diferença de frequência para dessintonização nula e razão de injeção de -8 dB.



(e) Expoente máximo de Lyapunov para diferença de frequência para dessintonização nula e razão de injeção de -8 dB.



fonte: Produção do próprio autor.

Para terceira condição inicial, utilizamos novamente uma diferença de frequência para dessintoni-

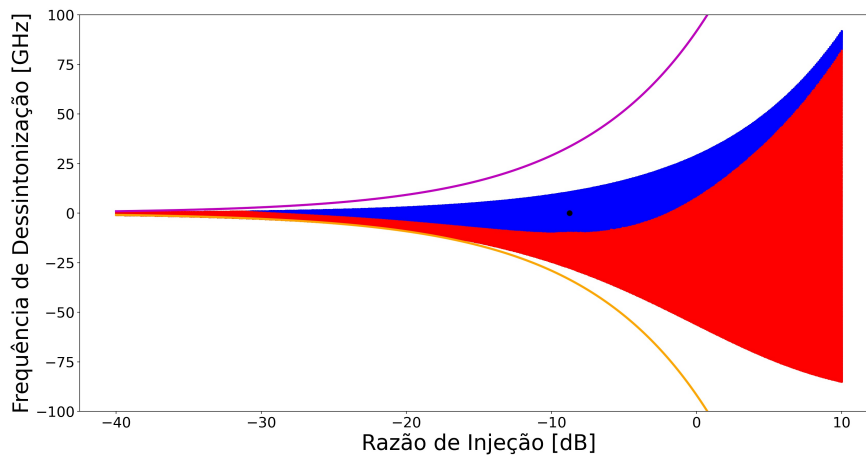
zação nula e razão de injeção de $-8,75$ dB, essa condição inicial também pode ser vista diretamente no mapa de bloqueio mostrado na Figura 10(a). Assim como a condição inicial anterior temos que a condição se encontra na região instável do sistema.

Nas Figuras 10(b), 10(c) e 10(d), vemos os resultados da integração. Neste caso, obtemos uma resposta muito semelhante a da condição inicial anterior. A princípio o sistema parece convergir para uma órbita periódica de período dois, contudo ao ampliarmos os gráficos podemos verificar que não se trata de um período dois e sim uma órbita de período muito superior.

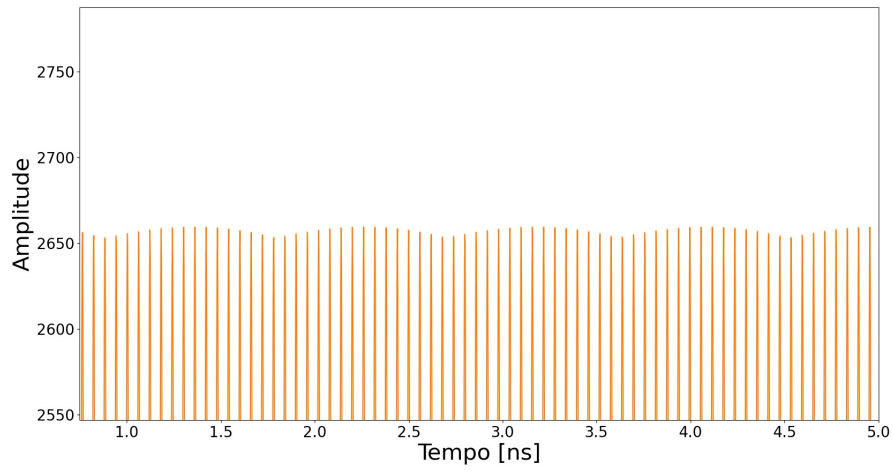
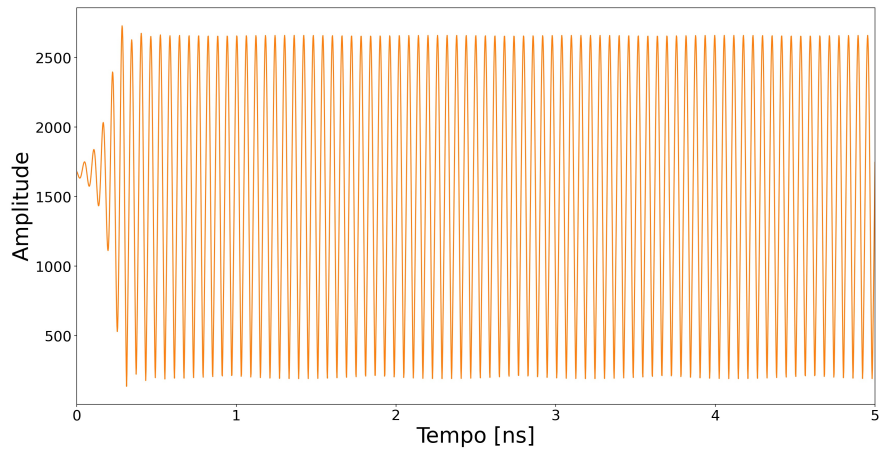
Por fim, na Figura 10(e) obtemos um valor de expoente de -7.26 , ou seja, expoente negativo o sistema ainda não apresenta um comportamento caótico, apesar de estar em uma região considerada instável no mapa de bloqueio.

Figura 10 – Resultados para diferença de frequência para dessintonização nula e razão de injeção de -8.75 dB.

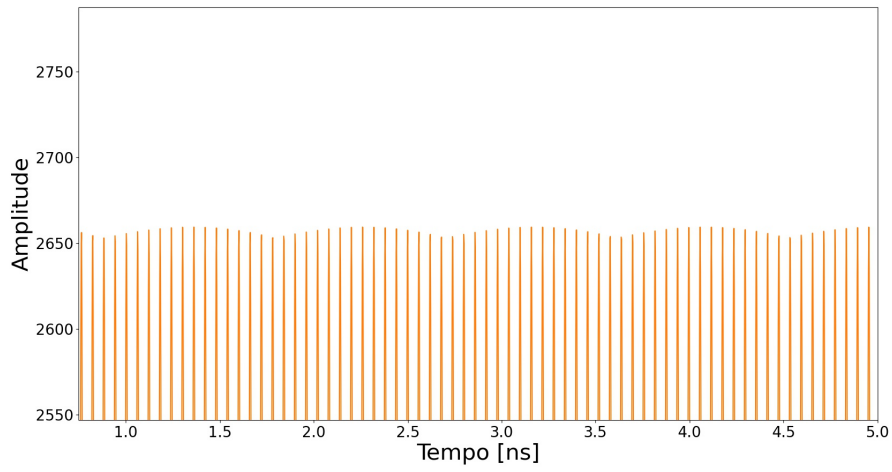
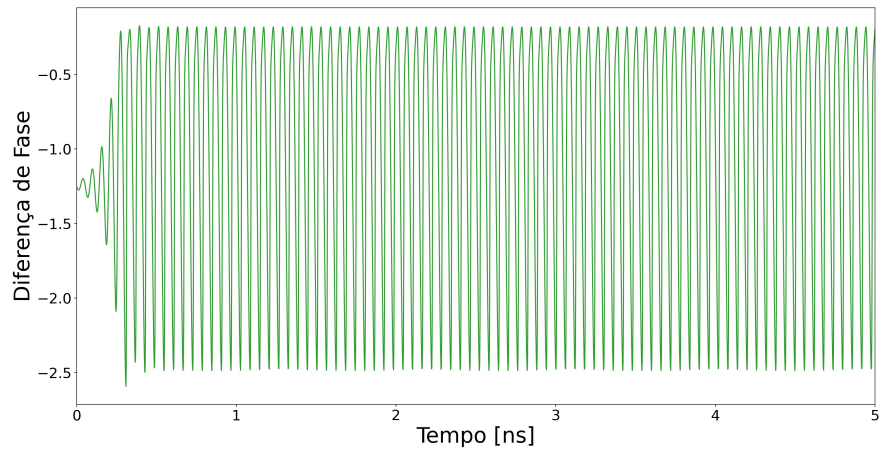
(a) Condição inicial escolhida no mapa de bloqueio.



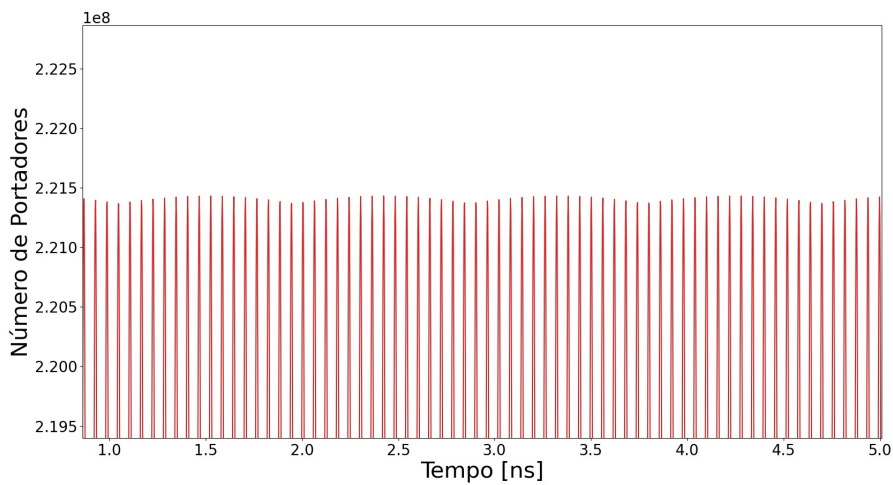
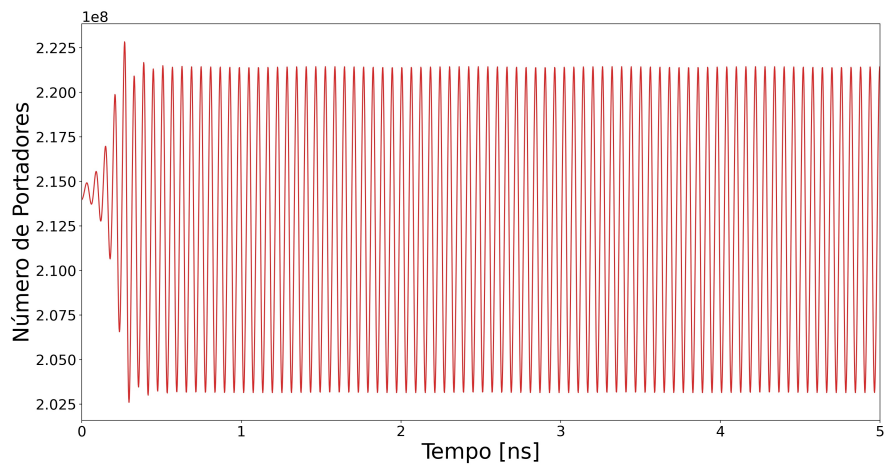
(b) Amplitude do campo para diferença de frequência para dessintonização nula e razão de injeção de -8.75 dB.



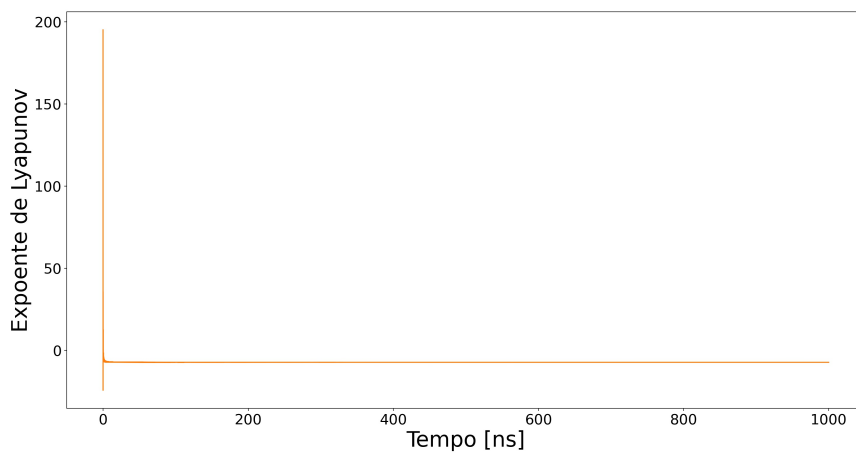
(c) Diferença de fases para diferença de frequência para dessintonização nula e razão de injeção de -8.75 dB.



(d) Número de portadores para diferença de frequência para dessintonização nula e razão de injeção de -8.75 dB.



(e) Expoente máximo de Lyapunov para diferença de frequência para dessintonização nula e razão de injeção de -8.75 dB.



fonte: Produção do próprio autor.

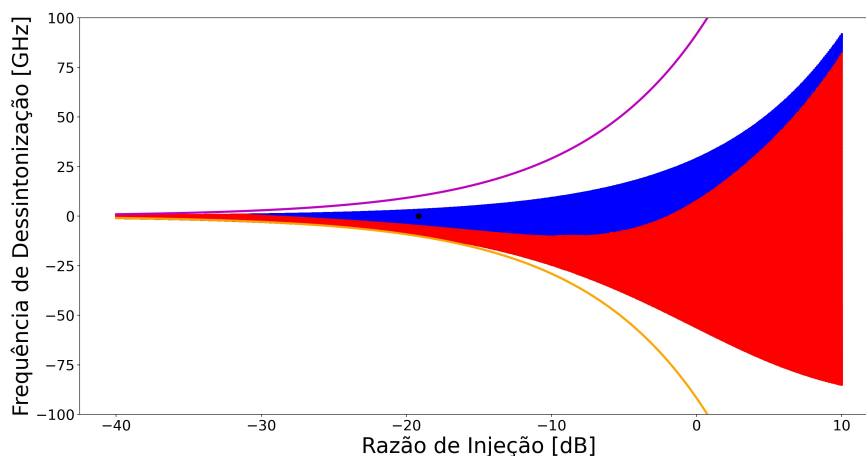
Como última condição inicial exemplificada neste trabalho, temos uma diferença de frequência

para dessintonização nula e uma razão de injeção de -19.15 dB, podendo ser vista diretamente no mapa de bloqueio mostrado na Figura 11(a), a qual se encontra em uma região instável do sistema.

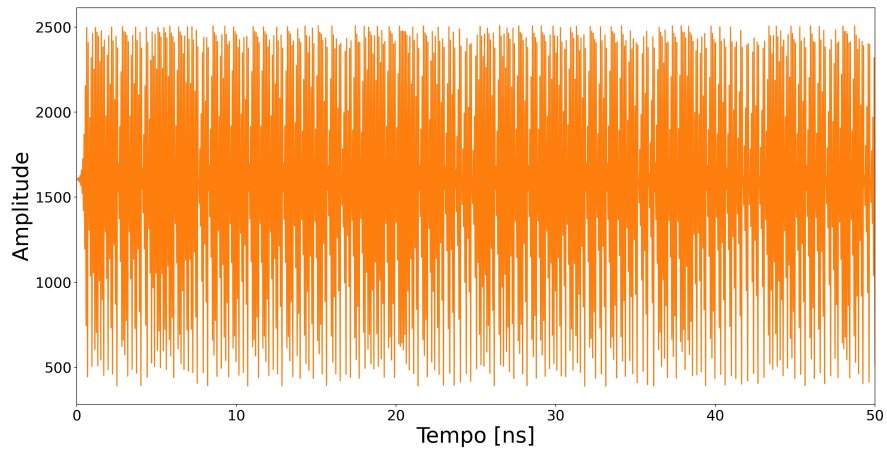
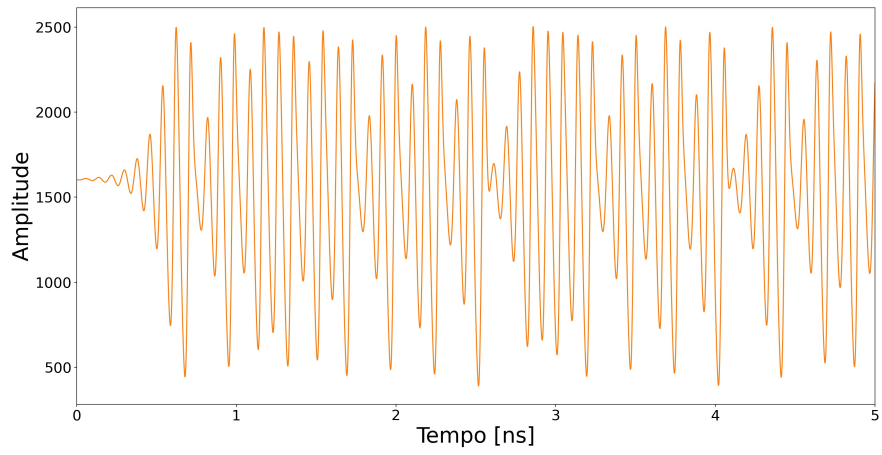
Nas Figuras 11(b), 11(c) e 11(d), vemos os resultados da integração utilizando o método RK4. Neste caso, obtemos uma resposta bem diferente das demais, que a princípio não tem uma periodicidade bem definida, podendo indicar uma região que tenha comportamento caótico. Para confirmar essa hipótese calculamos o expoente máximo de Lyapunov que convergiu para uma solução de aproximadamente 6.97, ou seja, expoente positivo indica um aumento na distância entre os pontos, que por sua vez leva para uma trajetória caótica. O resultado do cálculo do expoente máximo pode ser visto na Figura 11(e).

Figura 11 – Resultados para diferença de frequência para dessintonização nula e razão de injeção de -19.15 dB.

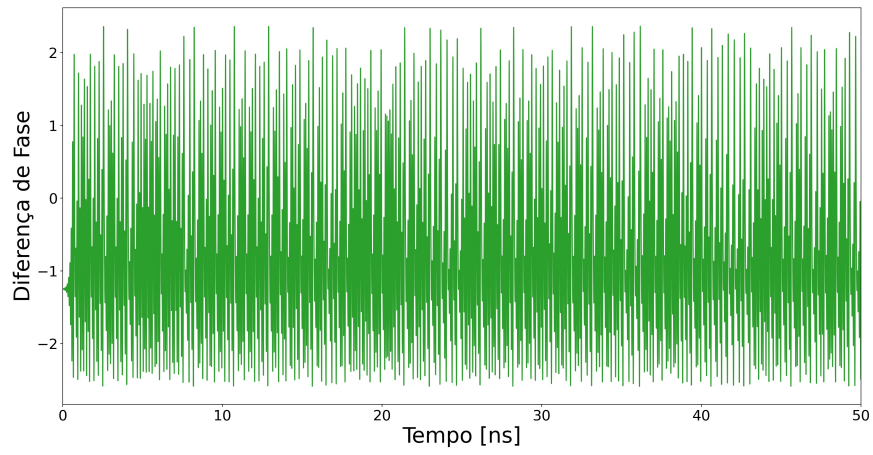
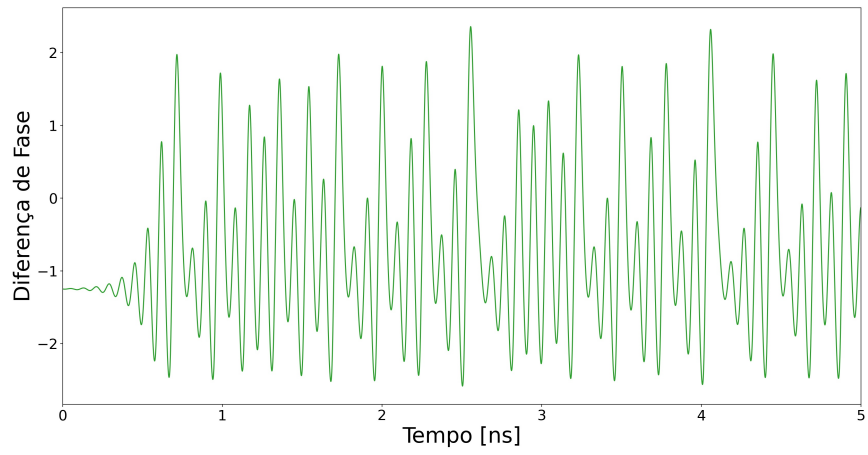
(a) Condição inicial escolhida no mapa de bloqueio.



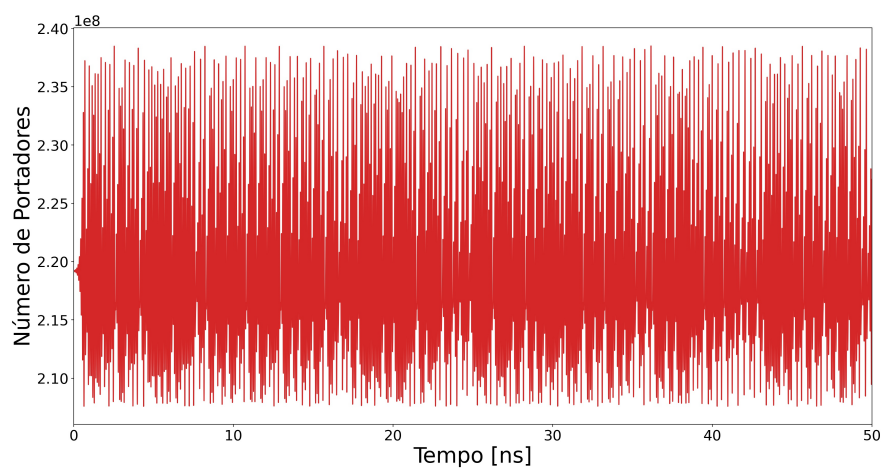
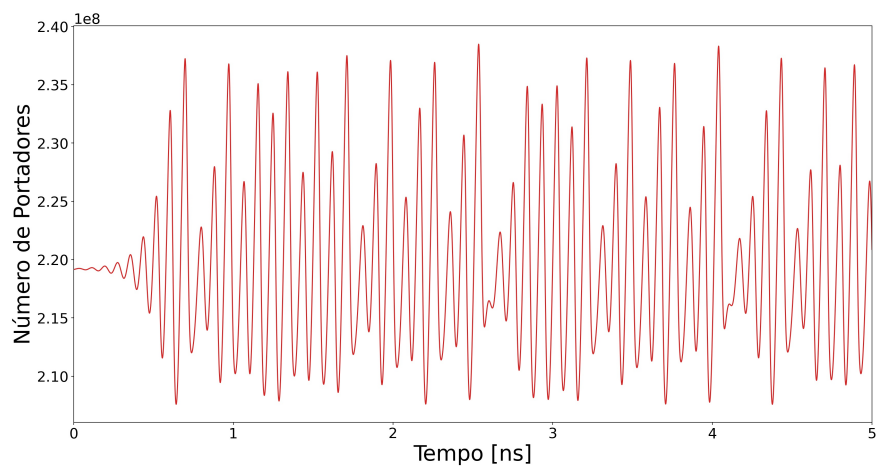
(b) Amplitude do campo para diferença de frequência para dessintonização nula e razão de injeção de -19.15 dB.



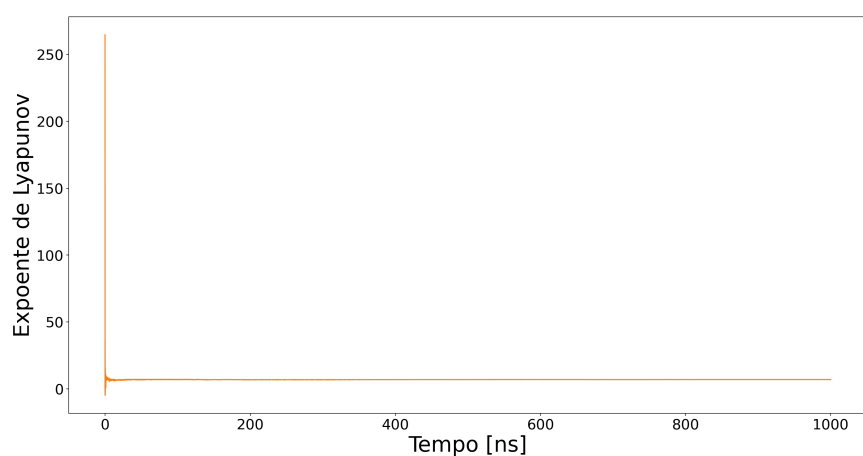
(c) Diferença de fases para diferença de frequência para dessintonização nula e razão de injeção de -19.15 dB.



(d) Número de portadores para diferença de frequência para dessintonização nula e razão de injeção de -19.15 dB.



(e) Expoente máximo de Lyapunov para diferença de frequência para dessintonização nula e razão de injeção de -19.15 dB.



fonte: Produção do próprio autor.

As quatro condições iniciais exibidas nesse trabalho foram escolhidas de forma a mostrar uma

gama maior de possibilidades e resultados encontrados. Outras condições iniciais foram testadas, a fim de validar o cálculo do expoente e em todos os casos alcançamos resultados semelhantes.

6 CONCLUSÃO

Este trabalho consistiu na caracterização de caos para um sistema de lasers semicondutores bloqueados por injeção óptica (OIL) por meio do cálculo do expoente máximo de Lyapunov, iniciando com o estudo das bases teóricas sobre sistemas dinâmicos e sobre o sistema óptico. Para o sistema óptico, foram descritas suas principais aplicações e vantagens, como por exemplo, redução de largura de linha, seleção e amplificação de tons, redução de *chirp*, maior ganho de enlace e possibilidade para modulação em taxas de transmissão maiores. Adicionalmente, o trabalho apresenta uma descrição sobre os métodos numéricos para integração das equações do sistema e o cálculo do expoente máximo de Lyapunov para indicar se há comportamento caótico.

O mapa de bloqueio e estabilidade do sistema óptico foi gerado, identificando os limites superior e inferior de operação dos sistemas, ou seja, a região onde é possível bloquear a frequência do laser escravo. Também foram identificadas as regiões de estabilidade e instabilidade do sistema. Desta forma, com o mapa de bloqueio calculado foi possível decidir quais condições iniciais seriam utilizadas para realizar o cálculo do expoente máximo de Lyapunov.

Uma vez escolhidas as condições iniciais, as equações diferenciais que regem o sistema foram integradas utilizando um integrador Runge-Kutta de quarta ordem. O método utilizado teve uma resposta muito boa, resolvendo as equações diferenciais com precisão e rapidez. Contudo, vale ressaltar que o método utilizado foi de passo fixo, de forma que deve-se utilizar um passo de tempo (dt) apropriado para o sistema, o que implica em fazer diversos testes iniciais para estabelecer esse valor adequadamente, observando a consistência dos resultados. Uma alternativa possível a esta etapa é o uso de um integrador numérico de passo variável com controle de erro.

Com a integração do sistema e variando as condições iniciais, foram encontradas diversas soluções, incluindo, pontos fixos, órbitas periódicas, de períodos menores e maiores e também soluções onde não se observa periodicidade. Na sequência, foi implementado o cálculo do expoente máximo de Lyapunov, o qual foi consistente com os resultados esperados quanto à indicação de comportamento caótico. Contudo vale lembrar também que, em geral, para sistemas de equações diferenciais, são necessários milhões de passos do integrador para obter um bom resultado do expoente máximo, com cerca de dois dígitos significativos, o que implica diminuir o passo do integrador numérico e aumentar o tempo fim da janela de integração.

Por fim, ao analisar os resultados obtidos, podemos concluir que foi possível identificar regiões estáveis do sistema com comportamento periódicos, assim como esperado. Contudo a principal contribuição do trabalho foi a análise das condições iniciais que estavam na região instável do mapa de bloqueio.

Com isso, estudos futuros sobre as regiões instáveis podem ser realizados, podendo ampliar as áreas de aplicação do sistema ou mesmo aumentando o desempenho do sistema para aplicações já existentes.

REFERÊNCIAS

- AGRAWAL, G. P. **Fiber-optic communication systems**. Rochester, New York: John Wiley & Sons, 2012. v. 222.
- ALLIGOOD, K. T.; SAUER, T.; YORKE, J. **Chaos: An Introduction to Dynamical Systems**. New York: Springer, 1996.
- BOCCALETTI, S. et al. The control of chaos: theory and applications. **Physics Reports**, v. 329, n. 3, p. 103 – 197, 2000. ISSN 0370-1573.
- BOYCE, W.; DIPRIMA, R. **Equações diferenciais elementares e problemas de valores de contorno**. Rio de Janeiro: Grupo Gen - LTC, 2010. ISBN 9788521617563.
- CARMO, R. **Sistemas de comunicações óticas em espaço livre explorando formatos de modulação avançados**. Tese (Doutorado) — IST/AFA, 2017.
- DOCA, R. H.; BISCUOLA, G. J.; BÔAS, N. V. **Tópicos de Física 1 - Mecânica**. São Paulo: Editora Saraiva, 2012. ISBN 98850218106.
- E-DISCIPLINAS-USP. **Sistemas Contínuos e Discretos**. 2022. Disponível em: <https://edisciplinas.usp.br/pluginfile.php/1881397/mod_resource/content/0/Aula1_Sistemas.pdf#:~:text=Sistemas%20Cont%C3%ADnuos%20e%20Discretos,X%20%C3%A9%20um%20conjunto%20discreto.> Acesso em: 3 de jun. 2022.
- GOUVEIA, P. B. O. **Resolução numérica de sistemas de equações não-lineares**. Tese (Doutorado) — UNIVERSIDADE DA BEIRA INTERIOR, 2018.
- HTFUTURE. **Detecção coerente vs. direta para 100G Metro Data Center Inter-connectivity (DCI)**. 2020. Disponível em: <<http://pt.oadm-cwdm-dwdm.com/info/coherent-vs-direct-detection-for-100g-metro-d-52186916.html>>. Acesso em: 3 de jun. 2022.
- KOBAYASHI, S.; KIMURA, T. Injection locking in algaas semiconductor laser. **IEEE Journal of Quantum Electronics**, v. 17, n. 5, p. 681–689, 1981.
- LAU, E. K. **High-Speed Modulation of Optical Injection-Locked Semiconductor Lasers**. Tese (Doutorado) — EECS Department, University of California, Berkeley, Dec 2006. Disponível em: <<http://www2.eecs.berkeley.edu/Pubs/TechRpts/2006/EECS-2006-188.html>>. Acesso em: 3 de jun. 2022.
- LIU, Z.; SLAVÍK, R. Optical injection locking: From principle to applications. **Journal of Lightwave Technology**, IEEE, v. 38, n. 1, p. 43–59, 2020.
- MONTEIRO, L. **Sistemas dinâmicos**. São Paulo: Editora Livraria da Física, 2006. ISBN 9788588325081.
- PANTALEONE, J. Synchronization of metronomes. **American Journal of Physics**, American Association of Physics Teachers, v. 70, n. 10, p. 992–1000, 2002.
- PARKER, T. S.; CHUA, L. O. **Practical Numerical Algorithms for Chaotic Systems**. New York: Springer, 1989. ISBN 9780387966892,0387966897.

SOUZA, E. G. Caracterização de sistemas dinâmicos através de gráficos de recorrência. **Universidade Federal do Paraná. Tese de doutorado**, 2008.

SPROTT, J. **Chaos and Time-series Analysis**. New York: Oxford University Press, 2003. ISBN 9780198508403.

STEINBRUCH, A.; WINTERLE, P. **Álgebra linear**. São Paulo: Pearson Makron Books, 1987. ISBN 9780074504123.

STROGATZ, S. **Nonlinear Dynamics And Chaos**. Reading: Perseus Books, 1994.

TELECOMUNICAÇÕES, T. I. E. **FSO I: Técnicas de Transmissão**. 2022. Disponível em: <https://www.teleco.com.br/tutoriais/tutorialfsoeab1/pagina_4.asp>. Acesso em: 3 de jun. 2022.

APÊNDICE A – SCRIPT - MAPA DE BLOQUEIO

```

1 ''' ***** '''
2 ''' @author: Frank Gustavson Filho '''
3 ''' @date: 05 de Junho de 2022 '''
4 ''' @title: Locking Map and Stability '''
5 ''' ***** '''
6
7 #*****#
8 # IMPORTAR BIBLIOTECAS
9 #*****#
10 import numpy as np
11 from matplotlib import pyplot as plt
12
13 #*****#
14 # VARIÁVEIS INICIAIS
15 #*****#
16 # Definir valores dos gráficos
17 fig=plt.figure(figsize=(1,1))
18 parameters = {'axes.labelsize': 30, 'axes.titlesize': 30, 'xtick.labelsize': 20,
19              'ytick.labelsize': 20, 'legend.fontsize': 20}
20 plt.rcParams.update(parameters)
21
22 # Variáveis iniciais - Physical constants
23 q = 1.6022*10**(-19)
24 c = 2.9979*10**(10)
25 mu0 = np.pi*4*10**(-9)
26 e0 = 1/(c**2)/mu0
27 eta0 = (mu0/e0)**(0.5)
28
29 # Variáveis iniciais - Theoretical quantities
30 ns = 1*10**(-9)
31 ps = 1*10**(-12)
32 pJ = 1*10**(-12)
33 kHz = 1*10**(3)
34 MHz = 1*10**(6)
35 GHz = 1*10**(9)
36 um = 1*10**(-4)
37 nm = 1*10**(-7)
38 Ang = 1*10**(-8)
39 mA = 1*10**(-3)
40 kA = 1*10**(3)
41 mW = 1*10**(-3)
42
43 g = (2.5*(10**(-16)))*(8.5*(10**(9)))/(1.5*(10**(-10)))*0.4
44 N_th = 2.214*(10**(8))

```



```

45 y_p = 1/(3*(10**(-12)))
46 y_n = 1*(10**(9))
47 tIn = 7*(10**(-12))
48 r0 = 0.548
49 alpha = 3
50 Ith = 33.5*(10**(-3))
51 J_th = Ith/q
52 J = 5*J_th
53
54 A_inj = 0
55 A_fr = ((J-(y_n*N_th))/y_p)**(0.5)
56 A_inj = A_fr
57
58 delta_w = 0
59 delta_N0 = 0
60
61 # Criar vetores
62 pi_x = []
63 pi_y = []
64 pe_x = []
65 pe_y = []
66
67 ls_x = []
68 ls_y = []
69 li_x = []
70 li_y = []
71
72 line_size = 3
73 point_size = 2
74
75 #*****#
76 #                               VARIAVEIS INICIAIS
77 #*****#
78 # Loop - R_inj e phi
79 for R_inj in np.logspace(-2.0, 0.5, num=501):
80     for phi in np.linspace(-np.pi/2, np.arctan(1/alpha), num=501):
81
82         # Calculo de phi_0 e K_inj
83         phi_0 = (np.pi/2)*np.sin(phi)
84         K_inj = ((1/tIn)*((1-(r0**2))/r0))*(R_inj)
85
86         # Calcular raizes: P(A0)=1.0*(A0**3)+cf2*(A0**2)+cf1*A0+cf0
87         cf3 = 1
88         cf2 = -2.0*K_inj*A_inj*np.cos(phi_0)/y_p
89         cf1 = -A_fr**2
90         cf0 = -y_n*2.0*K_inj*A_inj*np.cos(phi_0)/(y_p*g)
91
92         coeff_1 = [cf3, cf2, cf1, cf0]
93         A0_root = np.roots(coeff_1)

```

```

94
95     # Loop - Raizes
96     for A0 in A0_root:
97         if A0.imag <= 0:
98             if A0 >= A_fr:
99                 A0 = A0.real
100
101                 # Calculo de delta_N0 e delta_w
102                 delta_N0 = -(2*K_inj*A_inj*np.cos(phi_0))/(g*A0)
103                 delta_w = -(K_inj*((1+(alpha**2))**(0.5))*(A_inj/A0)*np.sin(
phi_0+np.arctan(alpha)))
104
105                 # Calculo da matriz
106                 z = K_inj*A_inj/A0
107                 m11 = z*np.cos(phi_0)
108                 m12 = z*A0*np.sin(phi_0)
109                 m13 = -0.5*g*A0
110                 m21 = -(z/A0)*np.sin(phi_0)
111                 m22 = z*np.cos(phi_0)
112                 m23 = -0.5*alpha*g
113                 m31 = 2*A0*(y_p-2*z*np.cos(phi_0))
114                 m33 = y_n+(g*(A0**2))
115
116                 # Calcular raizes: D(s)=cf3*s^3+cf2*s^2+cf1*s+cf0 (eq. 2.21
Lau Thesis)
117                 cf3 = 1
118                 cf2 = m11 + m22 + m33
119                 cf1 = m11*m22 + m11*m33 + m22*m33 - m12*m21 - m13*m31
120                 cf0 = m11*m22*m33 + m12*m23*m31 - m12*m21*m33 - m13*m31*m22
121
122                 coeff_2 = [cf3, cf2, cf1, cf0]
123                 eq_21 = np.roots(coeff_2)
124
125                 # Loop - Definir pontos estaveis e instaveis
126                 for root in eq_21:
127                     # SE RAIZ POSITIVA E COM PARTE IMAGINARIA = INSTAVEL
128                     if root.real > 0:
129                         if root.imag != 0:
130                             IR_db = 20*np.log10(R_inj)
131                             DF_ghz = (delta_w/(2*np.pi))*(10**(-9))
132
133                             # Adicionar pontos no vetor
134                             pi_x.append(IR_db)
135                             pi_y.append(DF_ghz)
136                             break
137
138                     # CASO CONTRARIO RAIZ = ESTAVEL
139                     else:
140                         IR_db = 20*np.log10(R_inj)

```

```

141         DF_ghz = (delta_w/(2*np.pi))*(10**(-9))
142
143         # Adicionar pontos no vetor
144         pe_x.append(IR_db)
145         pe_y.append(DF_ghz)
146         break
147
148     # Limite superior
149     if phi == -(np.pi/2):
150         ls_x.append(20*np.log10(R_inj))
151         ls_y.append(((K_inj*((1+(alpha**2))**(0.5)))/(2*np.pi))*(10**(-9)))
152
153     # Limite inferior
154     if phi == np.arctan(1/alpha):
155         li_x.append(20*np.log10(R_inj))
156         li_y.append((-K_inj*((1+(alpha**2))**(0.5)))/(2*np.pi))*(10**(-9)))
157
158 # Plotar Dados
159 plt.plot(pi_x,pi_y, 'o', color='blue', markersize=point_size, label='Pontos
    Instáveis')
160 plt.plot(pe_x,pe_y, 'o', color='red', markersize=point_size, label='Pontos
    Estáveis')
161 plt.plot(ls_x, ls_y, color='m', linewidth=line_size, label='Limite Superior')
162 plt.plot(li_x, li_y, color='orange', linewidth=line_size, label='Limite Inferior
    ')
163
164 #plt.title('IR_db x DF_ghz', weight='bold')
165 plt.xlabel('Taxa de Injeção [dB]')
166 plt.ylabel('Frequência de Dessintonização [GHz]')
167 plt.ylim(-100,100)
168 plt.legend()
169
170 # Salvar imagem
171 fig.set_size_inches(20, 10)
172 plt.savefig('1_Locking Map and Stability', dpi=200)
173
174 # Plotar figura
175 #plt.show()

```

APÊNDICE B – SCRIPT - RK4 - SISTEMA OIL

```

1  ''' ***** '''
2  ''' @author: Frank Gustavson Filho          '''
3  ''' @date:   05 de Junho de 2022          '''
4  ''' @title:  Integracao do Sistema OIL     '''
5  ''' ***** '''
6
7  #*****#
8  #                                     IMPORTAR BIBLIOTECAS
9  #*****#
10 import numpy as np
11 from matplotlib import pyplot as plt
12
13 #*****#
14 #                                     VARIAVEIS INICIAIS
15 #*****#
16 # Definir valores dos graficos
17 fig=plt.figure(figsize=(1,1))
18 parameters = {'axes.labelsize': 30, 'axes.titlesize': 30, 'xtick.labelsize': 20, 'ytick.labelsize': 20, 'legend.fontsize': 20}
19 plt.rcParams.update(parameters)
20 line_size = 3
21 point_size = 2
22
23 # Variaveis iniciais - Physical constants
24 q = 1.6022*10**(-19)
25 c = 2.9979*10**(10)
26 mu0 = np.pi*4*10**(-9)
27 e0 = 1/(c**2)/mu0
28 eta0 = (mu0/e0)**(0.5)
29
30 # Variaveis iniciais - Theoretical quantities
31 ns = 1*10**(-9)
32 ps = 1*10**(-12)
33 pJ = 1*10**(-12)
34 kHz = 1*10**(3)
35 MHz = 1*10**(6)
36 GHz = 1*10**(9)
37 um = 1*10**(-4)
38 nm = 1*10**(-7)
39 Ang = 1*10**(-8)
40 mA = 1*10**(-3)
41 kA = 1*10**(3)
42 mW = 1*10**(-3)
43
44 g = (2.5*(10**(-16)))*(8.5*(10**(9)))/(1.5*(10**(-10)))*0.4

```

```

45 N_th = 2.214*(10**(8))
46 y_p = 1/(3*(10**(-12)))
47 y_n = 1*(10**(9))
48 tIn = 7*(10**(-12))
49 r0 = 0.548
50 alpha = 3
51 Ith = 33.5*(10**(-3))
52 J_th = Ith/q
53 J = 5*J_th
54
55 delta_w = 0
56 delta_N0 = 0
57
58 # Criar vetores
59 pi_x = []
60 pi_y = []
61 pe_x = []
62 pe_y = []
63
64 ls_x = []
65 ls_y = []
66 li_x = []
67 li_y = []
68
69 #*****#
70 #                               EQUACOES SISTEMA OIL
71 #*****#
72 def deriv(z):
73     global g, N_th, K_inj, A_inj, alpha, K_inj, dW_inj, y_n, y_p
74
75     A = z[0]
76     phi = z[1]
77     N = z[2]
78
79     dA = 0.5*g*(N-N_th)*A + K_inj*A_inj*np.cos(phi)
80     dphi = 0.5*alpha*g*(N-N_th) - K_inj*(A_inj/A)*np.sin(phi)-dW_inj
81     dN = J - y_n*N - (y_p + g*(N-N_th))*(A**2)
82
83     zprime = [dA, dphi, dN]
84
85     return zprime
86
87 #*****#
88 #                               METODO RUNGE-KUTTA
89 #*****#
90 def rk4(y, h):
91     nn = 3
92     dydx = np.zeros(nn, dtype=np.float64)
93     dym = np.zeros(nn, dtype=np.float64)

```

```

94     dyt = np.zeros(nn, dtype=np.float64)
95     yt = np.zeros(nn, dtype=np.float64)
96
97     dydx = deriv(y)
98     i = 0
99     while i < nn:
100         yt[i] = y[i] + 0.5*h*dydx[i]
101         i = i + 1
102
103     dyt = deriv(yt)
104     i = 0
105     while i < nn:
106         yt[i] = y[i] + 0.5*h*dyt[i]
107         i = i + 1
108
109     dym = deriv(yt)
110     i = 0
111     while i < nn:
112         yt[i] = y[i] + h*dym[i]
113         i = i + 1
114
115     i = 0
116     while i < nn:
117         dym[i] = dyt[i] + dym[i]
118         i = i + 1
119
120     dyt = deriv(yt)
121     i = 0
122     while i < nn:
123         y[i]=y[i]+(h/6)*(dydx[i]+dyt[i]+(2.0*dym[i]))
124         i = i + 1
125
126     return y
127
128 #*****#
129 #                                     LOCKING MAP AND STABILITY
130 #*****#
131 # Variaveis iniciais
132 A_inj = 0
133 A_fr = ((J-(y_n*N_th))/y_p)**(0.5)
134 A_inj = A_fr
135
136 delta_w = 0
137 delta_N0 = 0
138
139 # Loop - R_inj e phi
140 for R_inj in np.logspace(-2.0, 0.5, num=501):
141     for phi in np.linspace(-np.pi/2, np.arctan(1/alpha), num=501):
142

```

```

143     # Calculo de phi_0 e K_inj
144     phi_0 = (np.pi/2)*np.sin(phi)
145     K_inj = ((1/tIn)*(1-(r0**2))/r0)*(R_inj)
146
147     # Calcular raizes: P(A0)=1.0*(A0**3)+cf2*(A0**2)+cf1*A0+cf0
148     cf3 = 1
149     cf2 = -2.0*K_inj*A_inj*np.cos(phi_0)/y_p
150     cf1 = -A_fr**2
151     cf0 = -y_n*2.0*K_inj*A_inj*np.cos(phi_0)/(y_p*g)
152
153     coeff_1 = [cf3, cf2, cf1, cf0]
154     A0_root = np.roots(coeff_1)
155
156     # Loop - Raizes
157     for A0 in A0_root:
158         if A0.imag <= 0:
159             if A0 >= A_fr:
160                 A0 = A0.real
161
162                 # Calculo de delta_N0 e delta_w
163                 delta_N0 = -(2*K_inj*A_inj*np.cos(phi_0))/(g*A0)
164                 delta_w = -(K_inj*((1+(alpha**2)**(0.5))*(A_inj/A0)*np.sin(
phi_0+np.arctan(alpha)))
165
166                 # Calculo da matriz
167                 z = K_inj*A_inj/A0
168                 m11 = z*np.cos(phi_0)
169                 m12 = z*A0*np.sin(phi_0)
170                 m13 = -0.5*g*A0
171                 m21 = -(z/A0)*np.sin(phi_0)
172                 m22 = z*np.cos(phi_0)
173                 m23 = -0.5*alpha*g
174                 m31 = 2*A0*(y_p-2*z*np.cos(phi_0))
175                 m33 = y_n+(g*(A0**2))
176
177                 # Calcular raizes: D(s)=cf3*s^3+cf2*s^2+cf1*s+cf0
178                 cf3 = 1
179                 cf2 = m11 + m22 + m33
180                 cf1 = m11*m22 + m11*m33 + m22*m33 - m12*m21 - m13*m31
181                 cf0 = m11*m22*m33 + m12*m23*m31 - m12*m21*m33 - m13*m31*m22
182
183                 coeff_2 = [cf3, cf2, cf1, cf0]
184                 eq_21 = np.roots(coeff_2)
185
186                 # Loop - Definir pontos estaveis e instaveis
187                 for root in eq_21:
188                     # SE RAIZ POSITIVA E COM PARTE IMAGINARIA = INSTAVEL
189                     if root.real > 0:
190                         if root.imag != 0:

```

```

191         IR_db = 20*np.log10(R_inj)
192         DF_ghz = (delta_w/(2*np.pi))*(10**(-9))
193
194         # Adicionar pontos no vetor
195         pi_x.append(IR_db)
196         pi_y.append(DF_ghz)
197         break
198
199         # CASO CONTRARIO RAIZ = ESTAVEL
200     else:
201         IR_db = 20*np.log10(R_inj)
202         DF_ghz = (delta_w/(2*np.pi))*(10**(-9))
203
204         # Adicionar pontos no vetor
205         pe_x.append(IR_db)
206         pe_y.append(DF_ghz)
207         break
208
209     # Limite superior
210     if phi == -(np.pi/2):
211         ls_x.append(20*np.log10(R_inj))
212         ls_y.append(((K_inj*((1+(alpha**2))**(0.5)))/(2*np.pi))*(10**(-9)))
213
214     # Limite inferior
215     if phi == np.arctan(1/alpha):
216         li_x.append(20*np.log10(R_inj))
217         li_y.append((-K_inj*((1+(alpha**2))**(0.5)))/(2*np.pi))*(10**(-9)))
218
219
220 #*****#
221 #                                     INTEGRACAO DO SISTEMA
222 #*****#
223 # Variaveis iniciais
224 f_det = -10*GHz           # Frequência de dessintonizacao
225 IR = -15                  # Taxa de injecao em dB
226 R_inj2 = 10**(IR/20)     # Converter taxa de injecao
227
228 A_inj = 0
229 K_inj = 0
230 dW_inj = 0
231 A_fr = ((J-(y_n*N_th))/y_p)**(0.5)
232
233 t_win = 100*ns
234 dt = 1/(1000*GHz)
235 tPTS = int(t_win/dt)
236
237 Res_t = []
238 Res_a = []
239 Res_p = []

```



```

240 Res_n = []
241
242 # Loop de integracao
243 while True:
244     dW_inj = 2*np.pi*f_det
245     A_inj = A_fr
246     R_inj = R_inj2
247     K_inj = ((1/tIn)*(1-(r0**2))/r0)*(R_inj)
248
249     phi_i = np.arcsin((-dW_inj/K_inj)/(1+(alpha**2)**(0.5)) - np.arctan(alpha)
250     N_i = N_th - 2*K_inj*np.cos(phi_i)/g
251     dN_i = N_i - N_th
252
253     # Condicao de parada
254     tmp = ((A_inj**2) - (y_n/y_p)/dN_i)/(1 + g*dN_i/y_p)
255     if tmp <= 0:
256         print ('Parar - Valor Negativo')
257         break
258
259     A_i = tmp**(0.5)
260
261     x = np.array([A_i, phi_i, N_i], dtype=np.float64)
262
263     t = 0.0
264     while t <= t_win:
265         x = rk4(x,dt)
266         t = t+dt
267
268         Res_t.append(t*(10**9))
269         Res_a.append(x[0])
270         Res_p.append(x[1])
271         Res_n.append(x[2])
272
273     break
274
275 #*****#
276 #                                     PLOTAR GRAFICOS
277 #*****#
278 plot1 = plt.figure(1)
279 plt.plot(pi_x,pi_y, 'o', color='blue', markersize=point_size, label='Pontos
    Instãaveis')
280 plt.plot(pe_x,pe_y, 'o', color='red', markersize=point_size, label='Pontos
    Estãaveis')
281 plt.plot(ls_x, ls_y, color='m', linewidth=line_size, label='Limite Superior')
282 plt.plot(li_x, li_y, color='orange', linewidth=line_size, label='Limite Inferior
    ')
283 #ax1.title('IR_db x DF_ghz', weight='bold')
284 plt.xlabel('Taxa de Injeããço [dB]')
285 plt.ylabel('Frequãncia de Dessintonizaããço [GHz]')

```

```

286 plt.ylim([-100,100])
287
288 IR_db = 20*np.log10(R_inj)
289 DF_ghz = (dW_inj/(2*np.pi))*(10**(-9))
290 plt.plot(IR_db,DF_ghz, 'o', color='black', markersize=point_size+5)
291
292 plot1.set_size_inches(20, 10)
293 IR_db = float("%.2f" % IR_db)
294 R_inj = float("%.2f" % R_inj)
295 plt.savefig('2_Locking Map - IR_db = ' + str(IR_db) + ', Fdet = ' + str(f_det/
    GHz) + ' GHz.jpg', dpi=200)
296
297
298 point_size = 7
299
300 plot2 = plt.figure(2)
301 plt.plot(Res_t, Res_a, 'tab:orange')
302 #plt.scatter(Res_t, Res_a, s=point_size)
303 #plt.set_title('A x Time', weight='bold')
304 plt.xlabel('Tempo [ns]')
305 plt.ylabel('Amplitude')
306 plt.xlim([0, 2])
307
308 plot2.set_size_inches(20, 10)
309 plt.savefig('3_Amplitude - IR_db = ' + str(IR_db) + ', Fdet = ' + str(f_det/GHz)
    + ' GHz.jpg', dpi=200)
310
311
312 plot3 = plt.figure(3)
313 plt.plot(Res_t, Res_p, 'tab:green')
314 #plt.scatter(Res_t, Res_p, s=point_size)
315 #plt.set_title('Phi x Time', weight='bold')
316 plt.xlabel('Tempo [ns]')
317 plt.ylabel('Diferena de Fase')
318 plt.xlim([0, 2])
319
320 plot3.set_size_inches(20, 10)
321 plt.savefig('4_Fase - IR_db = ' + str(IR_db) + ', Fdet = ' + str(f_det/GHz) + '
    GHz.jpg', dpi=200)
322
323
324 plot4 = plt.figure(4)
325 plt.plot(Res_t, Res_n, 'tab:red')
326 #plt.scatter(Res_t, Res_n, s=point_size)
327 #plt.set_title('N x Time', weight='bold')
328 plt.xlabel('Tempo [ns]')
329 plt.ylabel('Nmero de Portadores')
330 plt.xlim([0, 2])
331

```

```
332 plot4.set_size_inches(20, 10)
333 plt.savefig('5_Portadores - IR_db = ' + str(IR_db) + ', Fdet = ' + str(f_det/GHz
    ) + ' GHz.jpg', dpi=200)
334 #plt.show()
```

APÊNDICE C – SCRIPT - EXPOENTE LYAPUNOV

```

1 ''' ***** '''
2 ''' @author: Frank Gustavson Filho '''
3 ''' @date: 05 de Junho de 2022 '''
4 ''' @title: Expoente de Lyapunov '''
5 ''' ***** '''
6
7 #*****#
8 # Importar bibliotecas
9 #*****#
10 import numpy as np # Importar biblioteca numpy
11 import math
12 from matplotlib import pyplot as plt
13
14 #*****#
15 # VARIAVEIS INICIAIS
16 #*****#
17 # Definir valores dos gráficos
18 fig = plt.figure()
19 parameters = {'axes.labelsize': 30, 'axes.titlesize': 30, 'xtick.labelsize': 20,
20 'ytick.labelsize': 20, 'legend.fontsize': 20}
21 plt.rcParams.update(parameters)
22 line_size = 3
23 point_size = 2
24
25 # Variaveis iniciais - Physical constants
26 q = 1.6022*10**(-19)
27 c = 2.9979*10**(10)
28 mu0 = np.pi*4*10**(-9)
29 e0 = 1/(c**2)/mu0
30 eta0 = (mu0/e0)**(0.5)
31
32 # Variaveis iniciais - Theoretical quantities
33 ns = 1*10**(-9)
34 ps = 1*10**(-12)
35 pJ = 1*10**(-12)
36 kHz = 1*10**(3)
37 MHz = 1*10**(6)
38 GHz = 1*10**(9)
39 um = 1*10**(-4)
40 nm = 1*10**(-7)
41 Ang = 1*10**(-8)
42 mA = 1*10**(-3)
43 kA = 1*10**(3)
44 mW = 1*10**(-3)

```

```

45 g = (2.5*(10**(-16)))*(8.5*(10**(9)))/(1.5*(10**(-10)))*0.4
46 N_th = 2.214*(10**(8))
47 y_p = 1/(3*(10**(-12)))
48 y_n = 1*(10**(9))
49 tIn = 7*(10**(-12))
50 r0 = 0.548
51 alpha = 3
52 Ith = 33.5*(10**(-3))
53 J_th = Ith/q
54 J = 5*J_th
55
56 delta_w = 0
57 delta_N0 = 0
58
59 # Criar vetores
60 pi_x = []
61 pi_y = []
62 pe_x = []
63 pe_y = []
64
65 ls_x = []
66 ls_y = []
67 li_x = []
68 li_y = []
69
70 #*****#
71 #                               EQUACOES SISTEMA OIL
72 #*****#
73 def deriv(z):
74     global g, N_th, K_inj, A_inj, alpha, K_inj, dW_inj, y_n, y_p
75
76     A = z[0]
77     phi = z[1]
78     N = z[2]
79
80     dA = 0.5*g*(N-N_th)*A + K_inj*A_inj*np.cos(phi)
81     dphi = 0.5*alpha*g*(N-N_th) - K_inj*(A_inj/A)*np.sin(phi)-dW_inj
82     dN = J - y_n*N - (y_p + g*(N-N_th))*(A**2)
83
84     zprime = [dA, dphi, dN]
85
86     return zprime
87
88 #*****#
89 #                               METODO RUNGE-KUTTA
90 #*****#
91 def rk4(y, h):
92
93     nn = 3

```

```

94     dydx = np.zeros(nn, dtype=np.float64)
95     dym = np.zeros(nn, dtype=np.float64)
96     dyt = np.zeros(nn, dtype=np.float64)
97     yt = np.zeros(nn, dtype=np.float64)
98
99     dydx = deriv(y)
100    i = 0
101    while i < nn:
102        yt[i] = y[i] + 0.5*h*dydx[i]
103        i = i + 1
104
105    dyt = deriv(yt)
106    i = 0
107    while i < nn:
108        yt[i] = y[i] + 0.5*h*dym[i]
109        i = i + 1
110
111    dym = deriv(yt)
112    i = 0
113    while i < nn:
114        yt[i] = y[i] + h*dym[i]
115        i = i + 1
116
117    i = 0
118    while i < nn:
119        dym[i] = dyt[i] + dym[i]
120        i = i + 1
121
122    dyt = deriv(yt)
123    i = 0
124    while i < nn:
125        y[i]=y[i]+(h/6)*(dydx[i]+dyt[i]+(2.0*dym[i]))
126        i = i + 1
127
128    return y
129
130 #*****#
131 #                               LOCKING MAP AND STABILITY                               #
132 #*****#
133 # Variaveis iniciais
134 A_inj = 0
135 A_fr = ((J-(y_n*N_th))/y_p)**(0.5)
136 A_inj = A_fr
137
138 delta_w = 0
139 delta_N0 = 0
140
141 # Loop - R_inj e phi
142 for R_inj in np.logspace(-2.0, 0.5, num=501):

```

```

143     for phi in np.linspace(-np.pi/2, np.arctan(1/alpha), num=501):
144
145         # Calculo de phi_0 e K_inj
146         phi_0 = (np.pi/2)*np.sin(phi)
147         K_inj = ((1/tIn)*((1-(r0**2))/r0))*(R_inj)
148
149         # Calcular raizes: P(A0)=1.0*(A0**3)+cf2*(A0**2)+cf1*A0+cf0
150         cf3 = 1
151         cf2 = -2.0*K_inj*A_inj*np.cos(phi_0)/y_p
152         cf1 = -A_fr**2
153         cf0 = -y_n*2.0*K_inj*A_inj*np.cos(phi_0)/(y_p*g)
154
155         coeff_1 = [cf3, cf2, cf1, cf0]
156         A0_root = np.roots(coeff_1)
157
158         # Loop - Raizes
159         for A0 in A0_root:
160             if A0.imag <= 0:
161                 if A0 >= A_fr:
162                     A0 = A0.real
163
164                     # Calculo de delta_N0 e delta_w
165                     delta_N0 = -(2*K_inj*A_inj*np.cos(phi_0))/(g*A0)
166                     delta_w = -(K_inj*((1+(alpha**2))**(0.5))*(A_inj/A0)*np.sin(
167 phi_0+np.arctan(alpha)))
168
169                     # Calculo da matriz
170                     z = K_inj*A_inj/A0
171                     m11 = z*np.cos(phi_0)
172                     m12 = z*A0*np.sin(phi_0)
173                     m13 = -0.5*g*A0
174                     m21 = -(z/A0)*np.sin(phi_0)
175                     m22 = z*np.cos(phi_0)
176                     m23 = -0.5*alpha*g
177                     m31 = 2*A0*(y_p-2*z*np.cos(phi_0))
178                     m33 = y_n+(g*(A0**2))
179
180                     # Calcular raizes: D(s)=cf3*s^3+cf2*s^2+cf1*s+cf0 (eq. 2.21
181 Lau Thesis)
182                     cf3 = 1
183                     cf2 = m11 + m22 + m33
184                     cf1 = m11*m22 + m11*m33 + m22*m33 - m12*m21 - m13*m31
185                     cf0 = m11*m22*m33 + m12*m23*m31 - m12*m21*m33 - m13*m31*m22
186
187                     coeff_2 = [cf3, cf2, cf1, cf0]
188                     eq_21 = np.roots(coeff_2)
189
190                     # Loop - Definir pontos estaveis e intaveis
191                     for root in eq_21:

```

```

190         # SE RAIZ POSITIVA E COM PARTE IMAGINARIA = INSTAVEL
191         if root.real > 0:
192             if root.imag != 0:
193                 IR_db = 20*np.log10(R_inj)
194                 DF_ghz = (delta_w/(2*np.pi))*(10**(-9))
195
196                 # Adicionar pontos no vetor
197                 pi_x.append(IR_db)
198                 pi_y.append(DF_ghz)
199                 break
200
201         # CASO CONTRARIO RAIZ = ESTAVEL
202         else:
203             IR_db = 20*np.log10(R_inj)
204             DF_ghz = (delta_w/(2*np.pi))*(10**(-9))
205
206             # Adicionar pontos no vetor
207             pe_x.append(IR_db)
208             pe_y.append(DF_ghz)
209             break
210
211     # Limite superior
212     if phi == -(np.pi/2):
213         ls_x.append(20*np.log10(R_inj))
214         ls_y.append(((K_inj*((1+(alpha**2))**(0.5)))/(2*np.pi))*(10**(-9)))
215
216     # Limite inferior
217     if phi == np.arctan(1/alpha):
218         li_x.append(20*np.log10(R_inj))
219         li_y.append((-K_inj*((1+(alpha**2))**(0.5)))/(2*np.pi))*(10**(-9)))
220
221
222     #*****#
223     #                                     INTEGRACAO DO SISTEMA
224     #*****#
225     # Variaveis iniciais
226     f_det = 0*GHz           # Frequência de dessintonizacao
227     IR = 0                 # Taxa de injecao em dB
228     R = 10**(IR/20)       # Converter taxa de injecao
229
230     A_inj = 0
231     K_inj = 0
232     dW_inj = 0
233     A_fr = ((J-(y_n*N_th))/y_p)**(0.5)
234
235     t_win = 1000*ns
236     dt = 1/(200*GHz)
237     tPTS = int(t_win/dt)
238

```



```

239 Res_ly = []
240 Res_t = []
241
242 # Loop de integracao + expoente
243 while True:
244     dW_inj = 2*np.pi*f_det
245     A_inj = A_fr
246     R_inj = R
247     K_inj = ((1/tIn)*((1-(r0**2))/r0))*(R_inj)
248
249     phi_i = np.arcsin((-dW_inj/K_inj)/(1+(alpha**2)**(0.5)) - np.arctan(alpha)
250     while phi_i > 2*math.pi:
251         phi_i = phi_i % 2*math.pi
252     N_i = N_th - 2*K_inj*np.cos(phi_i)/g
253     dN_i = N_i - N_th
254
255     # Condicao de parada
256     tmp = ((A_inj**2) - (y_n/y_p)/dN_i)/(1 + g*dN_i/y_p)
257     if tmp <= 0:
258         print ('Parar - Valor Negativo')
259         break
260
261     A_i = tmp**(0.5)
262
263     x = np.array([A_i, phi_i, N_i], dtype=np.float64)
264
265     d_0 = 10**(-6)
266
267     t = 0.0
268     while t < t_win:
269         x = rk4(x, dt)
270         t = t+dt
271
272     x_0 = x
273     x_1 = x = np.array([x[0]+d_0, x[1]+d_0, x[2]], dtype=np.float64)
274
275     soma = 0
276     cont = 0
277     i = 0
278
279     t = 0.0
280     t_transiente = 0*ns
281     while t <= t_win:
282         x_0 = rk4(x_0, dt)
283         x_1 = rk4(x_1, dt)
284
285         d_1 = math.sqrt((x_1[0] - x_0[0])**2 + (x_1[1] - x_0[1])**2 + (x_1[2] -
286         x_0[2])**2)

```

```

287     if d_1 != 0:
288         x_1[0] = x_0[0] + (d_0/d_1)*(x_1[0] - x_0[0])
289         x_1[1] = x_0[1] + (d_0/d_1)*(x_1[1] - x_0[1])
290         x_1[2] = x_0[2] + (d_0/d_1)*(x_1[2] - x_0[2])
291
292     if t > t_transiente:
293         ly = d_1/d_0
294
295         if ly != 0:
296             log_ly = math.log(ly)
297         else:
298             log_ly = 0
299
300         soma = soma + log_ly
301         cont = cont + 1
302
303         exp_ly = (soma/(dt*10**(9)))/cont)
304
305         Res_t.append(t*(10**9))
306         Res_ly.append(exp_ly)
307
308     t = t+dt
309     break
310
311 print(IR)
312 print(exp_ly)
313 #*****#
314 #                                     PLOTAR GRAFICOS
315 #*****#
316 plot1 = plt.figure(1)
317 plt.plot(pi_x,pi_y, 'o', color='blue', markersize=point_size, label='Pontos
    InstÃaveis')
318 plt.plot(pe_x,pe_y, 'o', color='red', markersize=point_size, label='Pontos
    EstÃaveis')
319 plt.plot(ls_x, ls_y, color='m', linewidth=line_size, label='Limite Superior')
320 plt.plot(li_x, li_y, color='orange', linewidth=line_size, label='Limite Inferior
    ')
321 #ax1.title('IR_db x DF_ghz', weight='bold')
322 plt.xlabel('Taxa de InjeÃÃo [dB]')
323 plt.ylabel('FrequÃncia de DessintonizaÃÃo [GHz]')
324 plt.ylim([-100,100])
325
326 IR_db = 20*np.log10(R_inj)
327 DF_ghz = (dW_inj/(2*np.pi))*(10**(-9))
328 plt.plot(IR_db,DF_ghz, 'o', color='black', markersize=point_size+5)
329
330 plot1.set_size_inches(20, 10)
331 IR_db = float("%.2f" % IR_db)
332 R_inj = float("%.2f" % R_inj)

```

```
333 plt.savefig('2_Locking Map - IR_db = ' + str(IR_db) + ', Fdet = ' + str(f_det/
    GHz) + ' GHz.jpg', dpi=200)
334
335 point_size = 7
336
337 plot2 = plt.figure(2)
338 plt.plot(Res_t, Res_ly, 'tab:orange')
339 #plt.scatter(Res_t, Res_ly, s=point_size)
340 #plt.title('Exp_ly x Time', weight='bold')
341 plt.xlabel('Tempo [ns]')
342 plt.ylabel('Expoente de Lyapunov')
343 #v.xlim([0.23*(10**-7), 0.235*(10**-7)])
344
345 plot2.set_size_inches(20, 10)
346 IR_db = float("%.2f" % IR_db)
347 R_inj = float("%.2f" % R_inj)
348 plt.savefig('6_Expoente - IR_db = ' + str(IR_db) + ', Fdet = ' + str(f_det/GHz)
    + ' GHz.jpg', dpi=200)
349
350 #plt.show()
```