Distributed Positioning of Autonomous Mobile Sensors with Application to the Coverage Problem

A Game Theoretic Approach to Multi-Agent Systems

HANS-BERND DÜRR

Master's Degree Project
Stockholm, Sweden 2010

XR-EE-RT 2010:017
DISTRIBUTED POSITIONING OF AUTONOMOUS MOBILE SENSORS WITH APPLICATION TO THE COVERAGE PROBLEM

A Game Theoretic Approach to Multi-Agent Systems

Hans-Bernd Dürr

October 4, 2010

Supervisor KTH Stockholm
M. Stanković

Examiner KTH Stockholm
K. H. Johansson

Examiner Universität Stuttgart
F. Allgöwer
Abstract

In this thesis, general problems are considered where a group of agents should autonomously position themselves in such a way that a global objective function is maximized, whereas each agent uses only the measurement of its own utility function.

Specially constructed extremum seeking schemes for single and multi-agent systems are presented, where the agents have only access to the current value of their individual utility functions and do not know the analytical model of the global or local objectives. By using an approximative system that is calculated using a methodology based on Lie brackets, practical stability of an equilibrium point is proved for the single agent as well as for the multi-agent case. The motion dynamics of the agents are modeled as single integrators, double integrators and unicycles.

A potential game approach is used in order to deduce conditions under which the whole group of agents converges to a region arbitrary close to the maximum of a global objective function, that coincides with the Nash equilibrium of the game.

As an application of the proposed algorithms, the sensor coverage problem is introduced. In this problem, a group of autonomous sensors is meant to position themselves such that a certain region is covered optimally, in the sense that the amount of detected events appearing in this region, is maximized. The problem is interpreted as a potential game where individual utility functions for each sensor are constructed in a way suitable for the direct application of the proposed optimization methodology.
Contents

1 Introduction 1
   1.1 Motivational Example .......................... 1
   1.2 Problem Formulation ............................. 2
   1.3 Related Work .......................... 2
   1.4 Contribution and Outline .......................... 3

2 Mathematical Background 4
   2.1 Noncooperative Game Theory .......................... 4
   2.2 Lie Brackets and Nonlinear Systems .......................... 11
   2.3 Practical Stability .......................... 14
   2.4 Input-To-State Stability .......................... 17

3 Extremum and Nash Equilibrium Seeking 19
   3.1 Quadratic Maps .......................... 20
   3.2 Interpretation as Lie Bracket Motion .......................... 23
   3.3 Game Theoretic Multi-Agent Optimization .......................... 48

4 Sensor Coverage 62
   4.1 Sensor Coverage Problem .......................... 62
   4.2 Related Work .......................... 63
   4.3 Sensor Coverage as a Potential Game .......................... 65

5 Simulation Results 76
   5.1 Prerequisites .......................... 76
   5.2 Choice of Initial Conditions .......................... 79
   5.3 Comparison of Detection Probabilities .......................... 84
   5.4 Distance and Communication Costs .......................... 90
   5.5 Agents with Unicycle Dynamics .......................... 96
   5.6 Discrete Events .......................... 98

6 Summary and Future Work 100
   6.1 Future Work .......................... 101

A One-Dimensional Extremum Seeking 102

Bibliography 105
### List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Example of Autonomous Sensors in an Oceanic Environment</td>
<td>1</td>
</tr>
<tr>
<td>2.1</td>
<td>Composition of two Vector Fields</td>
<td>12</td>
</tr>
<tr>
<td>2.2</td>
<td>Brockett-Integrator with Piecewise Constant Inputs</td>
<td>13</td>
</tr>
<tr>
<td>2.3</td>
<td>Brockett-Integrator with Sinusoids</td>
<td>14</td>
</tr>
<tr>
<td>2.4</td>
<td>$(\sigma \rightarrow \rho)$-stability</td>
<td>14</td>
</tr>
<tr>
<td>3.1</td>
<td>Extremum Seeking Scheme</td>
<td>20</td>
</tr>
<tr>
<td>3.2</td>
<td>Discrete Time Extremum Seeking</td>
<td>22</td>
</tr>
<tr>
<td>3.3</td>
<td>Extremum Seeking for the Single Integrator</td>
<td>26</td>
</tr>
<tr>
<td>3.4</td>
<td>Single Integrator with $J(x_1, x_2) = -x_1^2 - x_2^2$</td>
<td>30</td>
</tr>
<tr>
<td>3.5</td>
<td>Single Integrator with $J(x_1, x_2) = e^{-x_1^2 - 5x_2^2} - 1$</td>
<td>30</td>
</tr>
<tr>
<td>3.6</td>
<td>Double Integrator</td>
<td>31</td>
</tr>
<tr>
<td>3.7</td>
<td>Double Integrator with $J(x_1, x_2) = -x_1^2 - x_2^2$</td>
<td>37</td>
</tr>
<tr>
<td>3.8</td>
<td>Single Integrator with $J(x_1, x_2) = e^{-x_1^2 - 5x_2^2} - 1$</td>
<td>38</td>
</tr>
<tr>
<td>3.9</td>
<td>Extremum Seeking for the Unicycle Model</td>
<td>39</td>
</tr>
<tr>
<td>3.10</td>
<td>Unicycle Model with $J(x_1, x_2) = -x_1^2 - x_2^2$</td>
<td>43</td>
</tr>
<tr>
<td>3.11</td>
<td>Unicycle Model with $J(x_1, x_2) = e^{-x_1^2 - 5x_2^2} - 1$</td>
<td>44</td>
</tr>
<tr>
<td>3.12</td>
<td>Unicycle Model with $J(x_1, x_2) = e^{-x_1^2 - 5x_2^2} - 1$</td>
<td>44</td>
</tr>
<tr>
<td>3.13</td>
<td>Model of Obstacle</td>
<td>45</td>
</tr>
<tr>
<td>3.14</td>
<td>Extremum Seeking with Obstacle Avoidance Feedback</td>
<td>46</td>
</tr>
<tr>
<td>3.15</td>
<td>Obstacle Avoidance</td>
<td>47</td>
</tr>
<tr>
<td>3.16</td>
<td>Extremum Seeking with Individual Utility Function</td>
<td>48</td>
</tr>
<tr>
<td>3.17</td>
<td>Extremum Seeking in a Quadratic Game with the Switching Scheme</td>
<td>52</td>
</tr>
<tr>
<td>3.18</td>
<td>Extremum Seeking in a Quadratic Game with Different Frequencies</td>
<td>57</td>
</tr>
<tr>
<td>3.19</td>
<td>Extremum Seeking with Individual Utility Function</td>
<td>58</td>
</tr>
<tr>
<td>4.1</td>
<td>Communication Tree Example</td>
<td>71</td>
</tr>
<tr>
<td>4.2</td>
<td>Considered Subgraphs for each $G_i$</td>
<td>72</td>
</tr>
<tr>
<td>4.3</td>
<td>Comparison of Event Counts and Continuous Utility function</td>
<td>75</td>
</tr>
<tr>
<td>5.1</td>
<td>$\Omega$ as subspace of $\mathbb{R}^2$</td>
<td>78</td>
</tr>
<tr>
<td>5.2</td>
<td>Evolution of Agent Positions (Switched) - Example 1</td>
<td>81</td>
</tr>
<tr>
<td>5.3</td>
<td>Evolution of Agent Positions (Different Frequencies) - Example 1</td>
<td>82</td>
</tr>
<tr>
<td>5.4</td>
<td>Values of Potential and Individual Utility Function - Example 1</td>
<td>83</td>
</tr>
<tr>
<td>5.5</td>
<td>Agent Coordinates over Time</td>
<td>83</td>
</tr>
<tr>
<td>5.6</td>
<td>Evolution of Agent Positions - Example 2</td>
<td>85</td>
</tr>
<tr>
<td>Figure Number</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>5.7</td>
<td>Potential Function and Individual Utility Functions - Example 2</td>
<td>86</td>
</tr>
<tr>
<td>5.8</td>
<td>Evolution of Agent Positions - Example 3</td>
<td>87</td>
</tr>
<tr>
<td>5.9</td>
<td>Potential Function and Individual Utility Functions - Example 3</td>
<td>88</td>
</tr>
<tr>
<td>5.10</td>
<td>Evolution of Agent Positions - Example 4</td>
<td>89</td>
</tr>
<tr>
<td>5.11</td>
<td>Potential Function and Individual Utility Functions - Example 4</td>
<td>89</td>
</tr>
<tr>
<td>5.12</td>
<td>Communication Tree</td>
<td>90</td>
</tr>
<tr>
<td>5.13</td>
<td>Evolution of Agent Positions - Example 5</td>
<td>91</td>
</tr>
<tr>
<td>5.14</td>
<td>Potential Function and Individual Utility Functions - Example 5</td>
<td>92</td>
</tr>
<tr>
<td>5.15</td>
<td>Evolution of Agent Positions - Example 6</td>
<td>93</td>
</tr>
<tr>
<td>5.16</td>
<td>Potential Function and Individual Utility Functions - Example 6</td>
<td>93</td>
</tr>
<tr>
<td>5.17</td>
<td>Evolution of Agent Positions - Example 7</td>
<td>95</td>
</tr>
<tr>
<td>5.18</td>
<td>Potential Function and Individual Utility Functions - Example 7</td>
<td>95</td>
</tr>
<tr>
<td>5.19</td>
<td>Evolution of Agent Positions - Example 8</td>
<td>97</td>
</tr>
<tr>
<td>5.20</td>
<td>Individual Utility Functions - Example 8</td>
<td>97</td>
</tr>
<tr>
<td>5.21</td>
<td>Evolution of Agent Positions - Example 9</td>
<td>99</td>
</tr>
<tr>
<td>5.22</td>
<td>Individual Utility Functions - Example 9</td>
<td>99</td>
</tr>
</tbody>
</table>
List of Tables

2.1 Prisoner's Dilemma .............................................. 5
2.2 Matching Pennies .............................................. 7
2.3 Potential Function for Prisoner’s Dilemma ................. 8

4.1 Individual Communication Cost Functions .................... 72

5.1 Parameters Exponential Detection Probability ............. 84
5.2 Parameters Continuous Detection Probability .............. 86
5.3 Parameters Constant Detection Probability ................. 87
5.4 Parameters Distance Costs ........................................ 91
5.5 Parameters Communication Costs ............................... 94
5.6 Parameters Discrete Events ....................................... 96
5.7 Parameters Discrete Events ....................................... 98
List of Symbols

$C^\infty$  
Space of smooth functions

$\frac{\partial f(x)}{\partial x}$  
Gradient of $f(x)$ with respect to $x$ if $f(x)$ is a scalar function

$\frac{\partial f(x)}{\partial x}$  
Jacobian of $f(x)$ with respect to $x$ if $f(x)$ is a vector-valued function

$\mathbb{Q}^+$  
The set of positive rational numbers

$\mathbb{R}$  
The set of real-valued numbers

$\mathbb{R}^n$  
The set of real $n$-dimensional vectors

$\nabla_x f(x)$  
Derivative of $f(x)$ with respect to the scalar $x$

$\| x \|$  
Vector norm of $x$

$\| x \|_2$  
Euclidian norm of $x$

$\partial \Omega$  
Boundary of a set $\Omega$

$\phi \circ \psi(x)$  
Concatenation of two functions, $\phi(\psi(x))$

$x_i$  
The $i$’th subcomponent of the vector $x$
Chapter 1

Introduction

1.1 Motivational Example

As motivation for this work, the question is considered of how a group of autonomous robots can position themselves such that they solve a common, globally defined goal by only having local information.

Autonomous vehicles and drones are already used in a lot of applications. Security and efficiency are the main reasons why a lot of tasks are done by drones and unmanned vehicles. Small and cheap robots can be used in order to solve a given problem faster and more accurate than humans can do.

Up to now, these robots can often only be used as single units and tasks where a team of robots is necessary, still require a lot of human interaction. It is therefore in the focus to find ways how these clusters of robots can be programmed in order to solve a global goal without any human leader.

Imagine for example, an underwater oil well, where oil spills out into the ocean. The oil will go up to the surface and distribute in a region near the source. Unfortunately, the exact position of the oil well is unknown but one is interested in the amount of oil that spills out, in order to get an estimate of the damage that is done to nature.

![Figure 1.1: Example of Autonomous Sensors in an Oceanic Environment](image.png)
One is looking for a solution where this task can be accomplished by robots which position themselves in an autonomous way by using only locally measured information. As there is no information about the exact distribution of the oil, there cannot be a global leader who controls the agents.

1.2 Problem Formulation

Given is a global function depending on the positions of all agents whose value should be maximized by finding the optimal position for each agent.

The agents are placed in the plane and can only move based on local information. None of the agents knows the global function and there is also no global leader and no omniscient agent that knows the function or can control the agents from outside. The agents are meant to find at least a local maximum of this unknown global function by moving autonomously.

The goal is to find local individual utility functions for each agent and a method of how these functions can be optimized using only the information available. The local functions may depend on the positions of the agents themselves as well as of the position of all the other agents. Thus, even if one agent decides not to move at all, its utility function can vary when the other agents move. The individual utility functions may also differ from each other and are unknown by other agents. They should be defined in such a way that only local information is used, namely such that only information in a certain region around the position of the agent is considered. The utility functions are not necessarily known by the agents in a closed and analytic form but they are functions that can be measured physically. Each agent is equipped with a sensor that senses its own utility function and returns only a value at the actual position of the agent. Therefore, it is not possible to compute the gradients of the utility functions. This notion coincides very well with the notion of autonomous sensors, as the position where the sensing takes place can directly be translated to the value of an utility function.

To extend the problem even more, one can imagine cases where the measurement of the utility function is disturbed by noise, as this is the usual case in reality.

A special case of this formulation is the sensor coverage problem where a group of sensors should cover a region where a predefined event detection probability is given. The sensors belonging to each agent have a limited range and therefore only local information. The overall covered range of all sensors should be optimal with respect to a global function.

1.3 Related Work

There has been made a lot of work in the field of autonomous sensors. Some of the problem formulations coincide with the one here proposed, but use different approaches, mainly in the optimization technique.

The extremum seeking algorithm has been extensively analyzed in [28] and [29] using the averaging theorem. The authors proposed similar schemes as in this work but with the application to the optimization of quadratic maps by a single agent. In [20] and [21] these ideas were extended to cases with a noisy measurement of a quadratic map.
The authors of [22] proved convergence of the schemes in the multi-agent case with application to quadratic games.

One of the tools used in this document are Lie brackets. They can be found in the recent literature, mainly concerning steering of nonholonomic systems. Different interpretations and applications can for example be found in [5], [6], [13] and [19].

Li and Cassandras formulated the sensor coverage problem in [12] and used a gradient method where each sensor moves into the direction of the steepest ascent of its utility function.

Regarding the sensor coverage from a purely game theoretic view the authors of [14], [30], and [31] propose different round based algorithms for the sensor coverage problem and examined their convergence.

1.4 Contribution and Outline

In this document, a new approach to the presented problem is proposed. The main focus lies on a class of optimization algorithms called extremum seeking. A new proof as well as a novel, more intuitive approach to the extremum seeking feedback will be given. It will be extended to multi-agent systems where a game theoretic formulation opens up a possibility to define a new framework for solving a class of presented problems, based on the proposed extremum seeking algorithms.

Tools such as the Lie bracket motion used in nonlinear systems theory as well as practical stability are used to prove and interpret the formulated theorems. These tools serve as the basis for the analysis of single agent as well as multi-agent systems. In the latter case, potential games will be used together with the Lie bracket approximation which leads to a very suitable and intuitive representation of the dynamical behavior.

Potential games are a special class of games having some nice properties. The idea is to find a potential function for a game theoretic problem and instead of doing extensive analysis of the game, one can only focus on the properties of the potential function. This provides some simple and useful tools and one will see that the notion of potential games fits perfectly into the problem formulation.

The main theoretical results are going to be applied to the sensor coverage problem which is a concrete example of such problems. It will serve as benchmark and example for applications that are close to reality.

The work is structured in the following way. In Chapter 2 the necessary mathematical tools such as game theory and Lie brackets as well as some useful notions like practical stability and input-to-state stability are presented.

In Chapter 3 the main results for the optimization algorithms are presented. A special feedback, called extremum seeking, is introduced and adapted to this special class of problems. Conditions are formulated under which the stability of the proposed single and multi-agent algorithms is proved.

The sensor coverage problem and its small modifications are presented in Chapter 4.

The main theoretical results are illustrated through simulations which are presented and interpreted in Chapter 5.

In Chapter 6 the results are summarized and some comments on the future work will be given.
Chapter 2

Mathematical Background

In this chapter a short introduction to the main mathematical tools that are needed for the rest of the document, are given. The first part consists of an introduction to noncooperative game theory and especially potential games. In the second part the extremum seeking feedback for a quadratic map as well as a proof for the convergence is introduced. The third part rolls out the notion of Lie brackets and where the inspiration to construct a different point of view on the extremum seeking feedback comes from. In the fourth part the practical asymptotical stability that is very useful when working with systems that are driven by a periodic time-varying function, is defined. And in the fifth part the input-to-state stability as well as some useful theorems are introduced.

2.1 Noncooperative Game Theory

Game theory in general, tries to model the behavior of players in a game. The characteristics of a noncooperative game are that every player has its own utility function that he wants to maximize. This utility function is not known to other players, but it depends on the actions of the others. The understanding of a game is meant in a very general context, as well as the notion of a player. Neither of them are hardly linked to board games, card games or other games in the common understanding of games. Game theory rather tries to model the interaction of participants of a system whose actions influence each other.

In the upcoming chapters the terminology agent is used for player instead as this is more convenient in view of the presented problem.

2.1.1 Basic Terminology

In this section some fundamental terminology that is widely used in the recent literature, is introduced. The notations in the following sections will be used in the latter paragraphs and are mostly taken from the recent literature (c.f. [30], [31]).

A strategic game \( \Gamma := \langle V, A, U \rangle \) consists of three components:

1. A set \( V \) enumerating agents \( i \in V := \{1, \ldots, N\} \)

2. An action set \( A := \prod_{i=1}^{N} A_i \) is the space of all actions vectors, where \( s_i \in A_i \) is the action of agent \( i \) and an (multi-player) action \( s \in A \) has components \( s_1, \ldots, s_N \). Note that the action sets can be either continuous or discrete.
3. The collection of utility functions $U$, where the utility function $u_i : A \to \mathbb{R}$ models agent $i$’s preferences over action profiles.

Additionally the following notation is used

$$s_{-i} = (s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_N).$$

It denotes the complementary action set of agent $s_i$. It is also common to write $u_i(s_i, s_{-i})$ instead of $u_i(s)$ to emphasize the difference between agents $i$’s own action and the actions of other agents.

**Example** [Prisoner’s Dilemma in [4]] To get a better picture of this, a common example for a game will be given to illustrate the relevance of each of these definitions. It is taken from [4], and is a very popular example.

The prisoner’s dilemma characterizes a situation in which two criminals, suspected of having committed a serious crime, are detained before a trial. Since there is no direct evidence against them, their conviction depends on whether they confess or not. If the prisoners both confess, then they will be sentenced to 8 years. If neither one confesses, then they will be convicted of a lesser crime and sentenced to 2 years. If only one of them confesses and puts the blame on the other one, then he is set free according to laws of the country and the other one is sentenced to 30 years.

One should mention that the suspects are not allowed to communicate with each other during their decision phase. This kind of constraint renders the game a noncooperative game.

Take a closer look at a more mathematical representation. As there are only four possibilities of decision combinations, it is very comfortable to write this in matrix form:

<table>
<thead>
<tr>
<th>Suspect 1</th>
<th>Suspect 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>(2,2)</td>
<td>(30,0)</td>
</tr>
<tr>
<td>(0,30)</td>
<td>(8,8)</td>
</tr>
</tbody>
</table>

Table 2.1: Prisoner’s Dilemma

This table provides an overview of the different possibilities that can arise in this game. As this is a bimatrix game the notation $(\cdot, \cdot)$ stands for the payoff of each suspect respectively, where the first entry belongs to Suspect 1 and the second to Suspect 2.

As one can expect, the agent set $V$ consists in this case of two agents, namely $V = \{\text{Suspect 1, Suspect 2}\}$, the action set $A = \{(\text{not confess, not confess}), (\text{confess, not confess}), (\text{not confess, confess}), (\text{confess, confess})\}$, whereas the utility functions are $u_i = \{(2,2), (0,30), (30,0), (8,8)\}$, one for each entry in the action set.

The game in this example is not very sophisticated, but nevertheless it is very useful to explain the basic terms in game theory.

### 2.1.2 Nash Equilibria

In the previous section only the terminology that is used in game theory, was introduced. Regarding the Prisoner’s Dilemma, one could ask if any decision of one of the suspects is
foreseeable. The symmetry or anonymity of the suspects, i.e. Suspect 1 and Suspect 2 could exchange their roles without changing the game, is obvious.

Consider what Suspect 1 might think in his situation. If he confesses, and Suspect 2 does not confess, he will go to prison for 30 years and Suspect 2 would be free. If Suspect 2 doesn’t confess, too, they would both get 2 years. Given that they cannot communicate with each other and the second possibility would imply a certain level of trust, it would be the best choice for him to confess.

As mentioned before, the game is symmetric, Suspect 2 will go through the same thinking process. Therefore, it turns out that both will confess at the end, as this is the point where none of them would get a higher reward if he would change his decision.

In game theory, this type of equilibrium is called Nash equilibrium. It is defined as the action set where a change of any of the agents would not end up in a higher value of its individual utility function whereas the change can only be done by one agent at a time.

**Definition 1.** A pure-strategy Nash equilibrium is defined as

\[ u_i(s^*_i, s^*_{-i}) = \max_{s_i \in A_i} u_i(s_i, s^*_{-i}) \].

A more useful definition in the context of the latter introduced Sensor Coverage Game is the notion of a local Nash equilibrium.

Given \( s_i \in A_i \) and \( \epsilon > 0 \), \( B(s_i, \epsilon) \) denotes the ball of radius \( \epsilon \) defined as \( B(s_i, \epsilon) = \{ s'_i \in A_i \mid |s'_i - s_i| < \epsilon \} \).

**Definition 2** (Definition 2 in [2]). A local Nash equilibrium for a real game \( \Gamma \) is a strategy profile \((s_i)_{i=1}^N \in A\) such that, for all \( i \), \( s^*_i \) is a local maximum of \( u_i \), that is, there exists \( \epsilon > 0 \) such that, for all \( s_i \in B(s_i, \epsilon) \), \( u_i(s^*_i, s_{-i}) \geq u_i(s_i, s_{-i}) \).

The authors in [2] used the definition in an economic problem environment to consider cases where the agents do not have full information about the whole action space. In such cases it is necessary to define such a local Nash equilibrium, because none of the agents would move away from such a point, as the information about their individual utility function is not present for them outside this region.

A necessary condition for an action \( s^* \) being a Nash equilibrium for continuously differentiable utility functions and a continuous mission space is

\[ (\nabla_{s_1} u_1(s^*)^T, \ldots, \nabla_{s_N} u_N(s^*)^T)^T = 0 \].

Another type of Nash equilibria are mixed-strategy Nash equilibria. This type of equilibria will not be used in the further chapters, but for the sake of completeness an example will be given that makes clear the difference between a pure-strategy and a mixed-strategy Nash equilibrium. The purpose of this work is to find a special realization and not only a probability distribution as it would turn out if one is looking for mixed-strategy Nash equilibria.

**Example** [Matching Pennies] Imagine now the matching pennies game. It covers the simple game of tossing two pennies. The agents can chose between matching or not matching, meaning that both pennies are either with the same side up or they show different results. The outcome is 1 for the winning and -1 for the losing player. One can see in Table 2.2 that there is no Nash equilibrium because there is no reason for any of the two agents to tend to a specific strategy. But one can rather give a probability for each strategy. The result is that there is an equilibrium in the probability distribution.
A probability distribution $\Delta(A_i)$ over the action space $A_i$ is defined in such a way that $\alpha_i \in \Delta(A_i)$, $\sum_{a_i \in A_i} \alpha_i(a_i) = 1$ and $\alpha_i(a_i) \geq 0$. It assigns each each possible action profile a positive probability.

The support of a probability measure $\mu$ is defined as $x : \mu(x) > 0$. In the context of $\alpha \in \Delta(A_i)$, the support of $\alpha$ is all $a_i$ such that $\alpha(a_i) > 0$. The Nash equilibrium for mixed-strategies is defined in the following way.

**Definition 3.** A mixed-strategy Nash equilibrium is a mixed-strategy profile $\alpha^*$ such that for each agent $i$ and for each $\alpha_i \in \Delta(A_i)$

$$u_i(\alpha^*_i, \alpha^*_{-i}) \geq u_i(\alpha_i, \alpha^*_{-i}).$$

As mentioned before, this type of Nash equilibria is not used here and in the following chapter the focus lies on the pure-strategy Nash equilibria.

### 2.1.3 Potential Games

Games in game theory can be classified in different categories. One of these categories are potential games. They exhibit a couple of properties that are helpful in examining the convergence of algorithms and the existence of equilibria. They were introduced by D. Monderer and L.S. Shapley in [17] in 1996 and are defined in the following way:

**Ordinal Potential Game** (from [17]). Let $\Gamma := \langle V, A, U \rangle$ be a game as defined before, with a set $V$ enumerating the agents, a set $A$ with the space of all action vectors and $U$ as the collection of all utility functions, then a function $P : A \rightarrow \mathbb{R}$ is called an ordinal potential function for $\Gamma$ if for every $i \in N$ and every $s_{-i} \in A_{-i}$

$$u_i(s_i, s_{-i}) - u_i(s_i', s_{-i}) > 0 \iff P(s_i, s_{-i}) - P(s_i', s_{-i}) > 0.$$ 

If a game admits an ordinal potential, then it is called ordinal potential game.

Additionally one can also define an exact potential game as following:

**Exact Potential Game** (from [17]). A function $P : A \rightarrow \mathbb{R}$ is an exact Potential for $\Gamma$ if for every $i \in V$ and every $s_{-i} \in A_{-i}$

$$u_i(s_i, s_{-i}) - u_i(s_i', s_{-i}) = P(s_i, s_{-i}) - P(s_i', s_{-i})$$

If a game admits an exact potential, then it is called exact potential game.

As these definitions are mainly for discrete action sets $A$, there is also a version for continuous action sets $A$ and differentiable utility functions $u_i$. The following lemma is proved according to above definitions, in [17].
Lemma 1 (Lemma 4.4 in [17]). Let $\Gamma$ be a game in which the strategy sets are intervals of real numbers. Suppose the payoff functions $u_i : A_i \rightarrow \mathbb{R}, i \in V$ are continuously differentiable, and let $P : A \rightarrow \mathbb{R}$ be continuously differentiable. Then $P$ is a potential for $\Gamma$ if and only if

$$\frac{\partial u_i}{\partial s_i} = \frac{\partial P}{\partial s_i} \quad \forall i \in N.$$ 

One can see that this characterization implies a certain alignment of all individual utility functions $u_i$ with the potential function $P$. An improvement in the utility function leads to an improvement in the potential function. These definitions allow under some additional assumptions a couple of further conclusions, but before stating them, an example of a potential function is given for the the Prisoner’s Dilemma from above.

Example [Prisoner’s Dilemma continued]

<table>
<thead>
<tr>
<th></th>
<th>$S_1$</th>
<th>$S_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$S_1$</td>
<td>(2,2)</td>
<td>(30,0)</td>
</tr>
<tr>
<td></td>
<td>(0,30)</td>
<td>(8,8)</td>
</tr>
</tbody>
</table>

(a) Utility Functions

<table>
<thead>
<tr>
<th></th>
<th>$S_1$</th>
<th>$S_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>24</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>22</td>
<td>0</td>
</tr>
</tbody>
</table>

(b) Potential Function

Table 2.3: Potential Function for Prisoner’s Dilemma

One can see that the potential function in Table 2.3b fulfills the definition of an Exact Potential.

Take for example the strategy set of $S_2$. A change in its strategy from not-confessing to confessing (given that $S_1$ does not change its strategy) equals the difference in the potential function in the same direction, i.e. $2 - 0 = 24 - 22$. Calculating the equivalent for $S_1$, i.e. $30 - 8 = 22 - 0$. As the prisoners try to minimize their utility functions in this game, the definition of a Nash equilibrium is the minimum of all possible choices. Obviously, the minimum of the potential function coincides with the Nash equilibrium.

This is a good example that shows that the potential function does not always have a physical interpretation, as there is no relation between the values of the potential function and the values of the utility functions.

Theorem 1 (Theorem 4.5 in [17]). Let $\Gamma$ be a game in which the strategy sets are intervals of real numbers. Suppose the payoff functions are twice continuously differentiable. Then $\Gamma$ is a potential game if and only if

$$\frac{\partial^2 u_i}{\partial s_i \partial s_j} = \frac{\partial^2 u_j}{\partial s_i \partial s_j} \quad \text{for every } i,j \in V. \quad (2.1)$$

Moreover, if the payoff functions satisfy (2.1) and $r$ is an arbitrary (but fixed) strategy profile in $A$, then a potential for $\Gamma$ is given by

$$P(s) = \sum_{i \in V} \int_0^1 \frac{\partial u_i}{\partial s_i}(x(t))(x_i)'(t)dt, \quad (2.2)$$

where $x : [0,1] \rightarrow A$ is a piecewise continuously differentiable path in $A$ that connects $r$ to $s$ (i.e., $x(0) = r$ and $x(1) = s$).
Example [Quadratic Games] A special class of games are quadratic games. In [4] they are defined in such a way, that the cost of each player is convex in his action variable and utility functions are given by

\[ u_i = \frac{1}{2} \sum_{j=1}^{N} \sum_{k=1}^{N} s'_j R_{jk}^i s_k + \sum_{j=1}^{N} s'_j r_j^i + c^i. \]  

(2.3)

In order to calculate the Nash equilibrium, that is in this case unique, one could differentiate all \( u_i \)'s with respect to \( s_i \), \( i \in V \) and set it to zero

\[ \frac{\partial u_i}{\partial s_i} = R_{ii}^i s_i + \sum_{j \neq i} R_{ij}^i s_j + r_i^i = 0, \quad i \in V \]

(2.4)

which can be written in compact form as

\[ Rs = -r \]

(2.5)

where

\[ R \triangleq \begin{pmatrix} R_{11}^1 & R_{12}^1 & \cdots & R_{1N}^1 \\ R_{21}^1 & R_{22}^1 & \cdots & R_{2N}^1 \\ \vdots & \vdots & \ddots & \vdots \\ R_{N1}^N & R_{N2}^N & \cdots & R_{NN}^N \end{pmatrix}, \]

(2.6)

\[ s' \triangleq (s_1, s_2, \ldots, s_N), \]

(2.7)

\[ r' \triangleq (r_1, r_2, \ldots, r_N). \]

(2.8)

The quadratic game of that form, admits a unique Nash equilibrium if, and only if, the matrix \( R \) is invertible, this yields to the Nash equilibrium \( s^* = -R^{-1}r \).

Furthermore, the quadratic game is a potential game, if and only if the matrix \( R \) is symmetric, this follows directly from Theorem [1]. The potential function is

\[ P(s) = s'Rs + s'r + c, \]

(2.9)

where \( c \) is some arbitrary constant. This can easily be checked by differentiating \( P(s) \) with respect to \( s_i \)

\[ \frac{\partial P}{\partial s_i} = R_{ii}^i s_i + \sum_{j \neq i} R_{ij}^i s_j + r_i^i = \frac{\partial u_i}{\partial s_i}. \]

2.1.4 Properties of Potential Games

The work with potential games is often very comfortable because of their properties due to the existence of a potential function.

The following lemmas and theorems are from D. Monderer and L.S. Shapley in [17].

Lemma 2 (Lemma 2.1 in [17]). Let \( P \) be an ordinal potential function for \( \Gamma \). Then the equilibrium set of \( \Gamma \) coincides with the equilibrium set of \( P \). That is, \( s \in A \) is an equilibrium point for \( \Gamma \) if and only if for every \( i \in N \)

\[ P(s) \geq P(q,s_{-i}) \quad \forall q \in A_i, \]

Consequently, If \( P \) admits a maximal value in \( A \), then \( \Gamma \) possesses a (pure strategy) equilibrium.
The previous lemma is very natural and obvious but allows to investigate the upcoming Nash equilibria by regarding only the potential function. Obviously, every local maximum of the potential function is also a local Nash equilibrium. The next corollary is an extension to the previous Lemma and treats the case of finite potentials.

**Corollary 1** (Corollary 2.2 in [17]). *Every finite ordinal potential game possesses a pure-strategy equilibrium.*

Corollary 1 shows how comfortable it is to work with potential games, because it simplifies the proof of the existence of a Nash equilibrium. This is often the first step in order to show convergence to such a point.

Before stating the next lemma the notion of a *path* is introduced. In [17] it is defined as a sequence $\gamma = (s^0, s^1, \ldots)$ such that for every $k \geq 1$ there exists a unique player, say Player $i$, such that $s^k_i = (q, s^k_{\neq i})$ for some $q \neq s^k_i \in A_i$. $s^0$ is called the initial point of $\gamma$, and if $\gamma$ is finite, then its last element is called the terminal point of $\gamma$. $\gamma = (s^0, s^1, \ldots)$ is an improvement path with respect to $\Gamma$ if for all $k \geq 1$ $u_i(s^k) \geq u_i(s^{k-1})$, where $i$ is the unique deviator at step $k$. $\Gamma$ has the finite improvement property if every improvement path is finite.

**Lemma 3** (Lemma 2.3 in [17]). *Every finite ordinal potential game has the finite improvement path.*

The proof is quite simple because once the upper bound of the potential function is reached the path cannot be improved any more.

To go back to the continuous world, one has first to make sure that the individual utility functions $u_i$ are bounded. In this case it is possible to define an $\epsilon$-*equilibrium* of a potential function as

$$P(r) \geq \sup_{q \in A} P(q) - \epsilon,$$

$r$ is called an $\epsilon$-equilibrium point of $P$.

**Lemma 4** (Lemma 4.1 in [17]). *Every bounded potential game possesses an $\epsilon$-equilibrium point for every $\epsilon > 0$.*

Consequently the notion of an improvement path has to be adjusted. In [17] the authors define an $\epsilon$-*improvement path* as follows: a path $\gamma = (s^0, s^1, \ldots)$ is an $\epsilon$-improvement path with respect to $\Gamma$ if for all $k \geq 1$ $u_i(s^k) \geq u_i(s^{k-1}) + \epsilon$, where $i$ is the unique deviator at step $k$. The game $\Gamma$ has the approximate finite improvement property if for every $\epsilon > 0$ every $\epsilon$-improvement path is finite.

**Lemma 5** (Lemma 4.2 in [17]). *Every bounded potential game has the approximate finite improvement property.*

Pure-strategy equilibria are the important type of equilibria one is looking for. Their notion is very close to the notion of equilibria in control engineering. In potential games, one can make a simple statement about the existence of such a pure-strategy Nash equilibrium.

**Lemma 6** (Lemma 4.3 in [17]). *Let $\Gamma$ be a continuous potential game with compact strategy sets. Then $\Gamma$ possesses a pure-strategy Nash equilibrium point.*
Nevertheless, this lemma will not be used here, because the idea behind it, is the same as in constraint optimization. Imagine a space that is compact, then it does not matter if the value function is convex or concave as long as it is bounded. Because of the restriction to a compact set, there will always be a maximum.

It will be shown in the latter document that the applied methods are not possible to use with constraint optimization problems.

There are also notions of robustness of Nash equilibria that are not going to be used in this document. For further information the reader is referred to \[4\) where these concepts are treated in detail.

2.2 Lie Brackets and Nonlinear Systems

For two given vector fields \(f(x)\) and \(g(x)\) one can define a Lie bracket according to the following definition.

**Definition 4.** Lie bracket for vector fields 

\[
[f, g] = \frac{\partial g}{\partial x} f - \frac{\partial f}{\partial x} g.
\]

In nonlinear control theory, especially input-affine systems, Lie brackets are an essential tool to analyze systems and to synthesize inputs for a nonlinear system with nonholonomic constraints in order to steer it from some initial point to some other end point. Lie brackets have several interpretations. In this section, certain notions that are useful for the proofs in the upcoming chapters, are introduced.

First, a geometric understanding that is quite common in the literature and can be found in \[19\] for example, is going to be carried out.

Consider an input-affine system of the form

\[
\dot{x} = f(x)u_1 + g(x)u_2 \quad x(0) = x_0 \quad x \in \mathbb{R}^n
\]

with \(f(x), g(x) \in C^\infty\) and flows \(\phi_t^f\) and \(\phi_t^g\) which belong to the vector fields \(f(x)\) and \(g(x)\) respectively. Imagine piecewise constant inputs \(u_1\) and \(u_2\) that are only active one at a time and switch between 1 and -1 over a certain time \(h\). Input \(u_1\) has the form

\[
u_1 = \begin{cases} 
1 & t \geq 0 \text{ and } t \leq h \\
0 & t \geq h \text{ and } t \leq 2h \\
-1 & t \geq 2h \text{ and } t \leq 3h \\
0 & t \geq 3h \text{ and } t \leq 4h
\end{cases}
\]

whereas input \(u_2\) would be the same as \(u_1\) but shifted about \(h\).

The evolution of the system can be calculated by regarding the concatenation of flows of the vector fields, hence the initial condition will be transported to \(\phi_{-h}^g \circ \phi_{-h}^f \circ \phi_{-h}^g \circ \phi_{-h}^f(x_0)\). Assume now that \(h\) is small, then the Taylor expansion of the flow can be expressed as

\[
\phi_{h}^f(x_0) = x_0 + hf(x_0) + \frac{h^2}{2} \frac{\partial f}{\partial x} f(x_0) + \ldots
\]
and the Taylor expansion of the vector fields
\[ g(x_0 + hf(x_0)) = g(x_0) + h \frac{\partial g}{\partial x} f(x_0) + \ldots. \]

The evolution after \(2h, 3h, 4h\) is respectively:
\[
\phi^g_h \circ \phi^f_h(x_0) = x_0 + hf(x_0) + \frac{h^2}{2} \frac{\partial f}{\partial x} f(x_0) \\
+ hg(x_0) + \frac{h^2}{2} \frac{\partial g}{\partial x} g(x_0) + \frac{h^2}{2} \frac{\partial g}{\partial x} f(x_0) + \ldots
\]
\[
\phi^{-f}_h \circ \phi^g_h \circ \phi^f_h(x_0) = x_0 + h(f(x_0) + g(x_0)) + h^2 \left( \frac{1}{2} \frac{\partial f}{\partial x} f + \frac{1}{2} \frac{\partial g}{\partial x} f + \frac{1}{2} \frac{\partial g}{\partial x} g \right) - \\
- h( f + h \frac{\partial f}{\partial x} (f + g)) + \frac{1}{2} h^2 \frac{\partial f}{\partial x} f + \ldots \\
= x_0 + hg + \frac{h^2}{2} \frac{\partial g}{\partial x} f - \frac{\partial f}{\partial x} g + \frac{1}{2} \frac{\partial g}{\partial x} g + \ldots
\]
\[
\phi^{\neg g}_h \circ \phi^{-f}_h \circ \phi^g_h \circ \phi^f_h(x_0) = x_0 + h^2(\frac{\partial f}{\partial x} f - \frac{\partial f}{\partial x} g) + \ldots \\
= x_0 - h^2[f, g](x_0) + O(h^3).
\]

One can see that linear terms cancel out and only Lie bracket terms \(\frac{\partial g}{\partial x} f - \frac{\partial f}{\partial x} g\) and higher order terms are left over. Regarding Figure 2.1 one can see that there is a gap between the initial value and the final value after applying this input-scheme. This gap is for small \(h\) approximately \(-h^2(\frac{\partial g}{\partial x} f - \frac{\partial f}{\partial x} g)\).

Another important interpretation of the Lie bracket stems from Roger Brockett in [5] and [6] where the author makes use of a special system called the Nonholonomic Integrator in order to show certain properties that arise when dealing with nonlinear systems. This example is also often used to show the notion of Lie brackets together with controllability.

Consider the following input-affine system (Non-holonomic Integrator) with two inputs \(u_1\) and \(u_2\) and three states \(x_i, i = 1, 2, 3 \in \mathbb{R}\).

\[
\begin{align*}
\dot{x}_1 &= u_1 \\
\dot{x}_2 &= u_2 \\
\dot{x}_3 &= x_2u_1 - x_1u_2
\end{align*}
\] (2.11)
Obviously, there arises first the question of controllability because there are three states but only 2 inputs. In order to show that the system is anyhow controllable it is first rewritten as an input-affine system

\[
\dot{x} = \begin{pmatrix} 1 & 0 \\ x_2 \\ f \end{pmatrix} u_1 + \begin{pmatrix} 0 & 1 \\ -x_1 \\ g \end{pmatrix} u_2. \tag{2.12}
\]

Like in the example before, the system consists of two vector fields together with two inputs like in equation (2.10). It is furthermore assumed that the same control law is applied as before. Calculating the Lie bracket, one obtains

\[
[f, g] = \partial g / \partial x f - \partial f / \partial x g = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ x_2 \\ f \end{pmatrix} - \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ x_1 \\ g \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -2 \end{pmatrix}. \tag{2.13}
\]

The resulting direction of the vector field \([f, g]\) points only into direction \(x_3\). Hence, one can conclude that the system is controllable because there exists at least one control law that drives the system into the direction \(x_3\) whereby the directions of \(x_1\) and \(x_2\) are directly controllable. An extended system with a virtual input \(v\) can be defined such that

\[
\dot{x}_e = f(x_e) u_1 + g(x_e) u_2 + [f, g] v. \tag{2.14}
\]

There exists a huge list of literature that give propositions how to construct the inputs \(u_1\) and \(u_2\) for a given \(v\), i.e. [13], [19]. This will not be considered in detail because controllability and steering are not part of this work.

Nevertheless, there are other properties that are interesting and can nicely be shown on this example. Regarding the final value of \(x(4h)\) after one period, one can see that the components \(x_1(4h) = x_1(0)\) and \(x_2(4h) = x_2(0)\) whereas the \(x_3(4h)\) equals the area in the \(x_1 - x_2\)-plane that is enclosed by the trajectory. This can quite easily be shown. The input scheme as well as the resulting trajectory can be found in Figure 2.2.

![Figure 2.2: Brockett-Integrator with Piecewise Constant Inputs](image)

Furthermore, it is not always necessary to restricted the inputs to piecewise constant values, sometimes a similar result can be achieved by applying sinusoids instead. From an heuristic
point of view, sinusoids are only the smooth variant of the piecewise constant inputs. The result can be found in Figure 2.3.

![Figure 2.3: Brockett-Integrator with Sinusoids](image)

**2.3 Practical Stability**

In this section another notion of stability is introduced. It concerns systems that are not approaching the origin exactly when time goes to infinity but stay in a region close to the origin.

The difference between this definition and the notion of limit cycles is that there is a parameter with which one can tune the size of the region around the origin. Furthermore, the cyclic behavior for this class of systems is often due to a periodic time-varying input, whereas a limit cycle can also be a nonlinear effect that is not directly visible in the equations.

In Figure 2.4 one can see a visualization of this idea. The size of the region around the origin can be tuned by some parameter $\epsilon$. Stability of dynamical systems was originally defined by Lyapunov (see for example [10]) as $(\epsilon \to \delta)$ stability, where for each $\epsilon > 0$ there is a $\delta(\epsilon)$ such that the system stays in an $\epsilon$-region around the origin. Asymptotic stability is when the system approaches the origin for $t \to \infty$. 

![Figure 2.4: $(\sigma \to \rho)$-stability](image)
This definition is in some cases too restrictive because not every system can fulfill the necessary conditions of asymptotic stability, meaning that it approaches the origin exact. Nevertheless, one wants to say something about the qualitative behavior of the system, because maybe one can show that the system approaches a region around the origin that can be made arbitrary small by tuning some parameters.

The exact definition for such a stability was given in [25] and is called \((\sigma \to \rho)\)-stability

**Definition 5** \((\sigma \to \rho)\)-stability in [25]). Given \(\sigma > \rho \geq 0\), the origin of the system \(\dot{x} = f(x, t)\) is said to be \((\sigma \to \rho)\)-stable if

1. for each \(\epsilon > \rho\) there exists \(\delta(\epsilon) > 0\) such that for all \(t_o \geq 0\)
   \[|x(t_o)| \leq \delta(\epsilon) \Rightarrow |x(t)| \leq \epsilon \quad \forall t \geq t_o\]

2. for each \(r \in (0, \sigma)\) there exists a finite \(\nu(r) > 0\) such that
   \[|x(t_o)| \leq r \Rightarrow |x(t)| \leq \nu(r) \quad \forall t \geq t_o\]

3. for each \(r \in (0, \sigma)\) and each \(\epsilon > \rho\) there exists a finite \(T(r, \epsilon) > 0\) such that for all \(t_o \geq 0\)
   \[|x(t_o)| \leq r \Rightarrow |x(t)| \leq \epsilon \quad \forall t \geq t_o + T(r, \epsilon)\]

The authors in [25] define a fast time-varying systems of the form \(\dot{x} = f(x, \epsilon)\) as semi-globally practically asymptotically stable if for all numbers \(\sigma\) and \(\rho\) with \(\infty > \sigma > \rho > 0\), there exists \(\epsilon^*\) such that for \(\epsilon \in (0, \epsilon^*)\) the corresponding system \(\dot{x} = f(x, \epsilon)\) is \((\sigma \to \rho)\)-stable.

**Definition 6** (Semi-Global Practical Asymptotic Stability in [25]). The origin of the system \(\dot{x} = f(x, \epsilon)\) with \(\epsilon > 0\) is said to be semi-globally practically asymptotically stable if for all numbers \(\sigma\) and \(\rho\) with \(\infty > \sigma > \rho > 0\), there exists \(\epsilon^*\) such that for \(\epsilon \in (0, \epsilon^*)\), the corresponding system \(\dot{x} = f(x, \epsilon)\) is \((\sigma \to \rho)\)-stable.

For the next theorem, some definitions and assumptions are necessary. Consider the following two systems

\[
\dot{x} = f^\epsilon(t, x) \\
\dot{x} = g(t, x)
\]

(2.15)

(2.16)

where the trajectories of the system \(\dot{x} = f^\epsilon(t, x)\) converge uniformly on compact intervals to the trajectories of the system \(\dot{x} = g(t, x)\) as \(\epsilon \to 0\).

Consider furthermore the following hypotheses, where the first one is a common assumption in Nonlinear Systems Theory to assure the existence and uniqueness of solutions

**Hypothesis 1** (Existence and Uniqueness in [18]). 1. For each \(\epsilon, f^\epsilon : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n\) is continuous and \(f^\epsilon(t, \cdot) : \mathbb{R}^n \to \mathbb{R}^n\) is locally Lipschitz uniformly with respect to \(t\) for \(t\) belonging to compact time intervals.

2. \(g : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n\) is continuous and \(g(t, \cdot) : \mathbb{R}^n \to \mathbb{R}\) is locally Lipschitz uniformly with respect to \(t\) for \(t\) belonging to compact time intervals.

whereas the second hypothesis is very important for the upcoming theorem. It defines a connection between the systems (2.15) and (2.16) in terms of their solutions.
Hypothesis 2 (Convergence of Trajectories in \[18\]). For every $T \in (0, \infty)$ and compact set $K \subset \mathbb{R}^n$ satisfying $\{(t, t_0, x_0) \times \mathbb{R} \times \mathbb{R}^n : t \in [t_0, t_0 + T], x_0 \in K\} \subset \text{Dom } \psi$, for every $d \in (0, \infty)$, there exists $\epsilon^* \in (0, \epsilon_0)$ such that for all $t_0 \in \mathbb{R}$, for all $x_0 \in K$ and for all $\epsilon \in (0, \epsilon^*)$

$$\begin{cases} 
\phi^\epsilon(t, t_0, x_0) \text{ exists} \\
||\phi^\epsilon(t, t_0, x_0) - \phi^\epsilon(t, t_0, x_0)|| < d & \forall t \in [t_0, t_0 + T].
\end{cases}$$

(2.17)

The authors also use their own definition of practical stability that is used for a larger class of systems as in Definition \[6\] that is only valid for systems of the form $\dot{x} = f(x, t)$. If (2.15) was such a system, the next Definition of practical global uniform asymptotical stability would coincide with the Definition of semi-global practical asymptotical stability.

Definition 7 (Practical Global Uniform Asymptotical Stability in \[18\]). Consider the system (2.15). Assume that Hypothesis 1 is satisfied and let $\phi^\epsilon$ denote the flow of the system. The origin of this system is called practically globally uniformly asymptotically stable if the following three conditions are all satisfied:

1. For every $c_2 \in (0, \infty)$, there exist $c_1 \in (0, \infty)$ and $\epsilon \in (0, \epsilon_0]$ such that for all $t_0 \in \mathbb{R}$, for all $x_0 \in \mathbb{R}^n$ with $||x_0|| < c_1$ and for all $\epsilon \in (0, \epsilon)$

$$\begin{cases} 
\phi^\epsilon(t, t_0, x_0) \text{ exists} \\
||\phi^\epsilon(t, t_0, x_0) - \phi^\epsilon(t, t_0, x_0)|| < c_2 & \forall t \in [t_0, \infty).
\end{cases}$$

(2.18)

2. For every $c_1 \in (0, \infty)$, there exist $c_2 \in (0, \infty)$ and $\epsilon \in (0, \epsilon_0]$ such that for all $t_0 \in \mathbb{R}$, for all $x_0 \in \mathbb{R}^n$ with $||x_0|| < c_1$ and for all $\epsilon \in (0, \epsilon)$

$$\begin{cases} 
\phi^\epsilon(t, t_0, x_0) \text{ exists} \\
||\phi^\epsilon(t, t_0, x_0) - \phi^\epsilon(t, t_0, x_0)|| < c_2 & \forall t \in [t_0, \infty).
\end{cases}$$

(2.19)

3. For every $c_1, c_2 \in (0, \infty)$, there exist $T \in (0, \infty)$ and $\epsilon \in (0, \epsilon_0]$ such that for all $t_0 \in \mathbb{R}$, for all $x_0 \in \mathbb{R}^n$ with $||x_0|| < c_1$ and for all $\epsilon \in (0, \epsilon)$

$$\begin{cases} 
\phi^\epsilon(t, t_0, x_0) \text{ exists} \\
||\phi^\epsilon(t, t_0, x_0) - \phi^\epsilon(t, t_0, x_0)|| < c_2 & \forall t \in [t_0, T, \infty).
\end{cases}$$

(2.20)

This definition is essential for the Theorem of the authors. The idea of the Theorem as well as the proof is simple but very useful.

Theorem 2 (Theorem 1 in \[18\]). Given systems (2.15) and (2.16) satisfying Hypothesis 1 and 2. If the origin is a globally uniformly asymptotic stable equilibrium point of (2.16), the origin of (2.15) is practically globally uniformly asymptotically stable.

This Theorem is very helpful when having a system that depends on a parameter like it is in (2.15) and a system that is globally asymptotically stable but it is known for sure that the trajectories are at least for some time close to each other. The authors used an inductive proof to show that if the trajectories are bounded for some time, then they are also bounded for all time, and conclude therefore that the fast time-varying system is practically globally uniformly asymptotically stable.

In \[8\], the authors use a definition of $(\sigma \rightarrow \rho)$-stability in terms of a $\mathcal{K}\mathcal{L}$-function similar to uniform global asymptotic stability in \[10\].
2.4. INPUT-TO-STATE STABILITY

**Definition 8** (Uniformly Asymptotically Stability in [8]). Let $\delta \in \mathbb{R}_{\geq 0}$ and $\Delta \in \mathbb{R}_{\geq 0}$ be given. The ball $B_{\delta}$ is said to be Uniformly Asymptotically Stable on $B_{\Delta}$ for the system $\dot{x} = f(x,t)$ if there exists a class $KL$-function $\beta$ such that the solution of $\dot{x} = f(x,t)$ from any initial state $x_0 \in B_{\Delta}$ and any initial time $t_0 \in \mathbb{R}_{\geq 0}$ satisfies

$$|\phi(t, t_0, x_0)|_{\delta} \leq \beta(|x_0|, t - t_0), \quad \forall t \geq t_0.$$  \hspace{1cm} (2.21)

The norm $|x|_{\delta}$ is defined as $|x|_{\delta} := \inf_{z \in B_{\delta}} |x - z|$.

This definition gives a more compact form of practical stability as it measures the distance of the trajectory at time $t$ to some region around the origin. This definition is not going to be used but it gives a more intuitive view on this type of stability.

**2.4 Input-To-State Stability**

The second type of stability that is going to be used later, concerns systems with inputs and is called *input-to-state stability*. In the field of nonlinear systems science it is helpful in some cases to know how the system behaves qualitatively for a certain input.

As this framework is a useful tool, some important theorems are going to be introduced. Consider the system

$$\dot{x} = f(t, x, u)$$  \hspace{1cm} (2.22)

where $f : [0, \infty) \times \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ is piecewise continuous in $t$ and locally Lipschitz in $x$ and $u$. The input $u(t)$ is a piecewise continuous, bounded function of $t$ for all $t \geq 0$.

**Definition 9** (Definition 4.7 in [10]). The system (2.22) is said to be input-to-state stable if there exist a class $KL$ function $\beta$ and a class $K$ function $\gamma$ such that for any initial state $x(t_0)$ and any bounded input $u(t)$, the solution $x(t)$ exists for all $t \geq t_0$ and satisfies

$$||x(t)|| \leq \beta(||x_0||, t - t_0) + \gamma \left( \sup_{t_0 \leq \tau \leq t} ||u(\tau)|| \right)$$  \hspace{1cm} (2.23)

To show that a system is input-to-state stable, the following theorem is very useful. It is the connection between input-to-state stability and Lyapunov stability and is delivers an easy way to conclude this type by using a convenient Lyapunov function.

**Theorem 3** (Theorem 4.19 in [10]). Let $v : [0, \infty) \times \mathbb{R}^n \rightarrow \mathbb{R}$ be continuously differentiable function such that

$$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(t, x, u) \leq -W(x), \quad \forall ||x|| \geq \rho(||u||) > 0$$  \hspace{1cm} (2.25)

$\forall (t, x, u) \in [0, \infty) \times \mathbb{R}^n \times \mathbb{R}^m$, where $\alpha_1, \alpha_2$ are class $K_{\infty}$ functions, $\rho$ is a class $K$ function, and $W_3(x)$ is a continuos positive definite function on $\mathbb{R}^n$. Then, the system (2.22) is input-to-state stable with $\gamma = \alpha_1^{-1} \circ \alpha_2 \circ \rho$.

The next Lemma considers the connection of a uniformly asymptotic stable system and a system that is input-to-state stable. Regarding the definition of input-to-state stability, one
can expect that such a connection must be globally asymptotically stable, as the input to the input-to-state stable subsystem is converging to the origin.

Consider the cascade system

\[
\begin{align*}
\dot{x}_1 &= f_1(t, x_1, x_2) \\
\dot{x}_2 &= f_2(t, x_1, x_2)
\end{align*}
\] (2.26)

where \( f_1 : [0, \infty) \times \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \to \mathbb{R}^{n_1} \) and \( f_2 : [0, \infty) \times \mathbb{R}^{n_2} \to \mathbb{R}^{n_2} \) are piecewise continuous in \( t \) and locally Lipschitz in \( x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \). Suppose both

\[
\dot{x}_1 = f_1(t, x_1, 0) \] (2.28)

and (2.27) have globally uniformly asymptotically stable equilibrium points at their respective origins.

Lemma 7. Under the stated assumptions, if the system (2.26), with \( x_2 \) as input is input-to-state stable and the origin of (2.27) is globally uniformly asymptotically stable independent of \( x_1 \), then the origin of the cascade system (2.26) and (2.27) is globally uniformly asymptotically stable.

Lemma 7 is very intuitive. Obviously, the feedback connection of a system that is input-to-state stable and an asymptotically stable system is asymptotically stable. This follows directly from the definition of input-to-state stability, as the driving force is decaying to zero the whole system converges to zero.

The proof for this Lemma is the same proof as for Lemma 4.7 in [10]. In that Lemma, the system (2.27) is depending only on \( x_2 \). Thus, if the dependance on the state \( x_1 \) does not change the asymptotic properties of the system, the function \( f(t, x_1, x_2) \) can explicitly depend on \( x_1 \).
Chapter 3

Extremum and Nash Equilibrium Seeking

In this chapter, the main algorithms that are going to be used in the further work, will be introduced. The results are applied to different systems and extended to the multi-agent case.

Consider an unknown, continuous and nonlinear function $f(x, y) : \mathbb{R}^2 \rightarrow \mathbb{R}$ that admits a unique maximum at a point $(x^*, y^*)$. The function $f(x, y)$ is unknown but one wants to search the maximum. Obviously, it is not possible to calculate the gradient in order to apply some gradient ascent algorithm as this implies that the function is known in some way. Nevertheless there exists a possibility to find the maximum of such a function by using a special control loop. In the recent literature such a control loop is called extremum seeking.

The idea is to steer a vehicle to the maximum point of the function $f(x, y)$ where $x$ and $y$ denote the position and $f(x, y)$ some landscape where the vehicle moves in.

To find the extremum it is necessary to excite the system and to estimate the slope of the unknown function at the actual position. It is the same idea like in the field of model identification where one has to apply a known signal and compare the output of the real system with an estimated model.

The extremum seeking is a special class of a time-varying, periodic feedback that steers the system to a neighborhood of the optimum of a function while measuring only the value of the function.

In the first part of this chapter, the work of Krstic et al. in [28] and [29] is presented, as well as an outline of the proof for convergence of the feedback. The main drawback of this analysis is that it needs the assumption of a quadratic map and the results are valid only locally.

In the latter sections a new view of the extremum seeking feedback for a more general nonlinear map of two variables, is presented. It shows global practical stability for general non-quadratic maps and was inspired by a paper of Brockett (cf. [5]), where a periodic time-varying input to a nonlinear system leads to a forward motion.
CHAPTER 3. EXTREMUM AND NASH EQUILIBRIUM SEEKING

3.1 Quadratic Maps

In Figure 3.1 one can see a flowchart of the type of extremum seeking that was proposed. As explained before, there has to be some excitation to the system. In this case there is a time-dependent cosine for the $x$-direction and a time-dependent sine for the $y$-direction added to the system input to make the system circle around some point.

The value of the nonlinear map $f(x, y)$ is evaluated at the position $(x, y)$. The result passes through a high-pass filter in order to filter out the DC-part, since all the information about the gradient is contained in the amplitudes of the sinusoids, and is then multiplied with a sine at the $x$-loop and a cosine at the $y$-loop. To get a better understanding why this set-up is steering the system to $(x^\ast, y^\ast)$ the system equations of the closed loop and an approximated system are examined.

The following equations are taken from [29]. Furthermore the authors assume a quadratic function of the form

$$f(x, y) = f^\ast - q_x (x - x^\ast)^2 - q_y (y - y^\ast)^2.$$  \hspace{1cm} (3.1)

First the authors introduce another timescale $\tau = \omega t$ and perform a coordinate transformation

$$\tilde{x} = x - x^\ast - \alpha \sin(\tau)$$  \hspace{1cm} (3.2)
$$\tilde{y} = y - y^\ast + \alpha \cos(\tau).$$  \hspace{1cm} (3.3)

The equations in the new coordinates $\tilde{x}$ and $\tilde{y}$ are then

$$\Delta = f(\tilde{x}, \tilde{y}) - f^\ast - e = -\left[ q_x (\tilde{x} + \alpha \sin(\tau))^2 + q_y (\tilde{y} - \alpha \cos(\tau))^2 + \epsilon \right]$$  \hspace{1cm} (3.4)

$$\frac{\partial \tilde{x}}{\partial \tau} = + \frac{1}{\omega} c_x \Delta \sin \tau$$
$$\frac{\partial \tilde{y}}{\partial \tau} = - \frac{1}{\omega} c_y \Delta \cos \tau$$
$$\frac{\partial e}{\partial \tau} = + \frac{h}{\omega}$$  \hspace{1cm} (3.5)

where $\tilde{e}$ is the state of the filter. The equations are now in a suitable form in order to use the
averaging theorem (cf. [10])

\[
\begin{align*}
\frac{\partial \tilde{x}_{\text{avg}}}{\partial \tau} &= -\frac{1}{\omega} c_x q_x \tilde{x}_{\text{avg}} \\
\frac{\partial \tilde{y}_{\text{avg}}}{\partial \tau} &= -\frac{1}{\omega} c_y q_y \tilde{y}_{\text{avg}} \\
\frac{\partial e_{\text{avg}}}{\partial \tau} &= -\frac{1}{\omega} h \left[ q_x \tilde{x}_{\text{avg}}^2 + q_y \tilde{y}_{\text{avg}}^2 + e_{\text{avg}} + \frac{\alpha^2}{2} (q_x + q_y) \right].
\end{align*}
\] (3.6)

By calculating the Jacobian of the average system, one can conclude that the origin of the average system is exponentially stable

\[
J_{\text{avg}} = \frac{1}{\omega} \begin{bmatrix}
-\alpha c_x q_x & 0 & 0 \\
0 & -\alpha c_y q_y & 0 \\
0 & 0 & -h
\end{bmatrix}.
\] (3.7)

The averaging theorem can also be used to give an upper bound for the region around the average system, where the original systems can be found.

**Theorem 4** (Theorem 2.1 in [29]). There exists \( \bar{\omega} \) such that for all \( \frac{1}{\omega} \in (0, \frac{1}{\omega}) \) the system in Figure 3.3 with nonlinear map of the form (3.1) has a unique exponentially stable periodic solution \( (\tilde{x}_{2\pi/\omega}, \tilde{y}_{2\pi/\omega}, e_{2\pi/\omega}) \) of period \( \frac{2\pi}{\omega} \) and this solution satisfies

\[
\left\| \begin{bmatrix}
\tilde{x}_{2\pi/\omega} \\
\tilde{y}_{2\pi/\omega} \\
\tilde{x}_{2\pi/\omega}^2 + \frac{\alpha^2}{2} (q_x + q_y)
\end{bmatrix} \right\| \leq O\left( \frac{1}{\omega} \right), \quad \forall \tau \geq 0.
\] (3.8)

After transforming the system back to the original coordinates, the solution reaches a region around the optimum defined by

\[
\begin{align*}
limit_{\tau \to \infty} \sup |x - x^*| &\leq O(\alpha + \frac{1}{\omega}) \\
limit_{\tau \to \infty} \sup |y - y^*| &\leq O(\alpha + \frac{1}{\omega}) \\
limit_{\tau \to \infty} \sup |f - f^*| &\leq O(\alpha^2 + \frac{1}{\omega^2}).
\end{align*}
\] (3.9)

One can see that the rate of convergence can be adjusted by pulling up the gains \( c_x \) and \( c_y \).

This is not always recommended, as for example in the case of an inexact measurement of the value function. Imagine that \( f(x, y) \) can only be measured or estimated then the measuring error can be interpreted as some noise \( \nu \) added to the output signal of \( f(x, y) + \nu \). This noise will also be amplified by high gains \( c_x \) and \( c_y \) resulting in bad performance of the algorithm.

Equations (3.9) show the connection between the parameter \( \alpha \) and the region around the optimum where the system converges to. It was desirable that this error tends to zero when the system is close to the optimum.

**Remark** Instead of taking a single-integrator as model for the vehicle, it is also possible to extend the extremum seeking to double-integrators or even to the unicycle model by only making small adjustments like adding a lowpass filter in the double-integrator case, for example.
3.1.1 Extremum Seeking in a Noisy Environment

Stankovic et al. proposed in [20] an approach where the gains $c_x, c_y$ and $\alpha$ are time varying and decaying over time. The advantage of this approach is to converge to the optimum even in the presence of noise.

In contrast to the scheme before, one can see in Figure 3.2 that the gains are time-varying and the utility function $f(x,y)$ is disturbed by white noise, denoted here as $\xi(k)$. The driving inputs are in this case

$$s_x(k) = \alpha(k + 1) \cos((k + 1)\omega) - \alpha(k) \cos(k\omega)$$
$$s_y(k) = \alpha(k + 1) \sin((k + 1)\omega) - \alpha(k) \sin(k\omega)$$
$$t_x(k) = \epsilon(k) \cos(\omega k - \phi)$$
$$t_y(k) = \epsilon(k) \sin(\omega k - \phi).$$

The authors show that the algorithm converges to the optimal solution for a convex quadratic map of the form $f(x,y) = f^* + q_x(x-x^*)^2 + q_y(y-y^*)^2$ and for $k \to \infty$ under the following assumptions

A.1 The sequence $\{\xi(k)\}$ is a martingale difference sequence defined on a probability space $(\Omega, \mathcal{F}, P)$ with a specified sequence of $\sigma$-algebras $\mathcal{F}_k \subseteq \mathcal{F}_{k+1}$, such that the variables $\xi_k$ are measurable with respect to $\mathcal{F}_k$ and they satisfy $E[\xi(k)^2] < \sigma(k)^2 < M < \infty$, $k = 1, 2, \ldots$

A.2 The sequence $\epsilon(k)$ is decreasing, $\epsilon(k) > 0$, $k = 1, 2, \ldots$, $\lim_{k \to \infty} \epsilon(k) = 0$

A.3 The sequence $\alpha(k)$ is decreasing, $\alpha(k) > 0$, $k = 1, 2, \ldots$, $\lim_{k \to \infty} \alpha(k) = 0$

A.4 $\sum_{k=1}^{\infty} \epsilon(k)\alpha(k) = \infty$

A.5 $\sum_{k=1}^{\infty} \epsilon(k)^2 < \infty$

The gain $\alpha(k)$ can be interpreted as the amplitudes of the sinusoids, whereas $\epsilon(k)$ are the gains, that tune the convergence rate and are comparable to the meaning of $c_x$ and $c_y$ in the previous scheme. If $\alpha(k)$ was a constant, there would always be a non-vanishing input that prevents the system from converging. If $\epsilon(k)$ would be a constant, the sensitivity to noise would not decline and one could end up with wobbling around the optimum.
Theorem 5 (Theorem 1 in [20]). Let the assumptions (A.1-5) and \(-\pi/2 < \phi + \psi < \pi/2\) be satisfied. Then, for the scheme from Fig. 3.2, \(x(k)\) converges to \(x^*\) and \(y(k)\) converges to \(y^*\) almost surely under the condition that \(\sup_k(||\hat{Z}(k)||) < B\) (almost surely), \(0 < B < \infty\).

The vector \(\hat{Z}(k)\) is meant to be \([x^* - x(k) - s^*_x(k), y^* - y(k) - s^*_y(k)]\). The requirement of a bounded state is a common assumption in such set-ups. The parameter \(\psi\) denotes the phase shift of the filter \(z^{-1}z + h\).

3.2 Interpretation as Lie Bracket Motion

This section is introducing a completely new view to the extremum seeking feedback. By using a Lie bracket analysis and the notion of practical stability, it can be extended to a larger class of nonlinear functions as well as the analysis of multi-agent systems.

Before stating the main results of this chapter, an important theorem must be introduced. It connects an input-affine system with an approximated Lie bracket system, by giving a bound on the difference of the trajectory of the two systems. It is the basis of the following theorems and allows to conclude practical stability of the extremum seeking feedback.

Consider the following system

\[
\dot{x} = \sum_{i=1}^{m} b_i(x)u_i^\epsilon, \quad x \in \mathbb{R}^n, b_i(x) \in C^\infty : \mathbb{R}^n \to \mathbb{R}^n
\]  

with control input of the form

\[
u_i = \hat{u}_i(t) + \frac{1}{\sqrt{\epsilon}}\tilde{u}_i(t, \theta)
\]

where \(\theta = t/\epsilon\) and the function \(\hat{u}_i\) is periodic with period \(T \in (0, \infty)\), and has zero average

\[
\int_0^T \tilde{u}_i(t, \theta)d\theta = 0.
\]

Theorem 6 (Theorem 2.1 in [13] p. 68). For sufficiently small \(\epsilon\), the trajectory of the system (3.10) is bounded by solution of the following system

\[
\dot{z} = \sum_{i=1}^{m} b_i(z)\bar{u}_i + \frac{1}{T} \sum_{i<j} [b_i, b_j] \nu_{i,j}, \quad z(0) = x(0)
\]

in the sense that

\[
||x - z||_{C[0,T]} \leq \Delta_1 + \Delta_2 \epsilon
\]

where

\[
\nu_{1,2} = \int_0^T \int_0^\theta \tilde{u}_i(t, \tau)\tilde{u}_j(t, \theta)d\tau d\theta
\]

and \((\Delta_1, \Delta_2, \epsilon)\) are parameters that tend to zero as \(\epsilon \to 0\).
The norm $|| \cdot ||_{C[0,T]}$ is defined as

$$||y||_{C[0,T]} = \max_{t \in [0,T]} |y(t)|.$$ 

As the understanding of the Theorem is important for the upcoming results, a short outline of the proof will be given.

**Proof.** The authors of the theorem first calculated the derivative of $x$ with respect to the fast variable $\theta$

$$x' \triangleq \frac{dx}{d\theta} = \epsilon \sum_{i=1}^{m} b_i(x)\bar{u}_i(t) + \sqrt{\epsilon} \sum_{i=1}^{m} b_i(x)\tilde{u}_i(t,\theta)$$

$$t' = \epsilon.$$ (3.12)

A new variable $y \in \mathbb{R}^n$ is introduced and is related to the variable $x$ in the following sense:

$$x = y + \sqrt{\epsilon}\phi_1(y,t,\theta) + \epsilon\phi_2(y,t,\theta) + \ldots$$ (3.13)

with periodic functions $(\phi_1, \phi_2)$ in $\theta$. The variable $y$ satisfies therefore a simpler differential equation

$$y' = \frac{dy}{d\theta} = \epsilon (f_1(y,t) + \sqrt{\epsilon}f_2(y,t) + \ldots)$$ (3.14)

By using the Taylor expansion of $x'$ and $y'$ and integration by parts the authors end up with functions

$$\phi_1(y,t,\theta) = \sum_{i=1}^{k} b_i(y)\int_{0}^{\theta} \tilde{u}_i(t,\tau)d\tau,$$

$$\phi_2(y,t,\theta) = \int_{0}^{\theta} \left( \sum_{i=1}^{k} \frac{\partial b_i(y)}{\partial y} \tilde{u}_i(t,\tau)\phi_1(y,t,\tau) - \sum_{i<j} [b_i, b_j]\nu_{i,j}d\tau \right) + \tilde{\phi}_2(y,t),$$

$$f_1(y,t) = \sum_{i=1}^{m} b_i(y)\bar{u}_i(t) + \frac{1}{T} \sum_{i<j} [b_i, b_j]\nu_{i,j}$$ (3.17)

with $\nu_{i,j}$ defined in the Theorem and finally the function

$$f_2(y,t) = \frac{1}{T} \int_{0}^{T} \left( \sum_{i=1}^{k} \frac{\partial b_i}{\partial y} \phi_2\tilde{u}_i(t,\theta) + \sum_{i=1}^{m} \sum_{k=1}^{n} \tilde{u}_i \left( \frac{\partial^2 b_k}{\partial y^2} \phi_1, \phi_1 e_k \right) \right) d\theta.$$ (3.18)

The average system of $y'$ can now be calculated to

$$\dot{z} = \sum_{i=1}^{m} b_i(z)\bar{u}_i(t) + \frac{1}{T} \sum_{i<j} [b_i, b_j]\nu_{i,j}, \quad z(0) = y(0) = x(0)$$ (3.19)

where this is valid in a region $\Delta_\epsilon \leq \Delta_1,\epsilon + \Delta_2,\epsilon$ with

$$\Delta_1,\epsilon \leq \sqrt{\epsilon}||\phi_1|| + O(\epsilon)$$ (3.20)

$$\Delta_2,\epsilon \leq \sqrt{\epsilon}(e^{TK}/K)||f_2||_{C[0,T]}$$ (3.21)

and $|| \cdot ||$ as the maximum norm in a neighborhood of $z(t)$ and $K$ some constant defined by higher order terms. 

\[\square\]
Remark  It is necessary to mention that this theorem states in its original version, that the distance between the trajectories of the original system and the Lie bracket system is bounded on the time-interval $[0, 2\pi]$. As this theorem is going to be used together with Theorem 2 that requires that the trajectories are bounded for an arbitrary time $T \in (0, \infty)$, one can replace $2\pi$ with an arbitrary time $T$ without changing the result of the theorem. It is always possible to find a sufficiently small $\epsilon$ and inputs $u_i$ such that the distance of the trajectories is bounded for an arbitrary time $T$.

This theorem is essential for the proofs that follow. It simplifies investigation of a class of systems because it allows to examine the Lie bracket system and, based on that, to deduce properties for the original one.

Lie brackets give a better understanding of how sinusoidal excitations can influence the behavior of a system like the one in Equation (3.10).

The idea is to interpret the sinusoids as external inputs in order to be able to use the Lie bracket analysis. In the proofs, the knowledge of the previous chapters is required and will serve as theoretical basis.

In general, Lie brackets are used for nonholonomic systems. The notion of nonholonomicity comes from mechanical systems that underly certain constraints. These systems are characterized by the fact that a periodic change in some parameter does not lead to a periodic change in the state variables.

In the following, this idea is used for the extremum seeking. By applying a special feedback incorporating a sinusoidal input and a nonlinear map, the system is steered along the gradient of the nonlinear map. The direction of the gradient is not explicitly accessible, but it is the response of the system to the sinusoidal inputs.

Furthermore, Theorem 6 was originally formulated for nonholonomic systems, this is not a restriction but a feature of the results. Holonomic systems imply that they can directly be steered from an initial point to an end point without the usage of sinusoids.

The function $J(x) : \mathbb{R}^2 \rightarrow \mathbb{R}$ will denote the objective function that is to be maximized. As global practical asymptotical stability will be proven, there must be made some assumptions:

A.1 $J(x)$ is smooth

A.2 there exists a unique $x^*$ such that $\frac{\partial J(x)}{\partial x}|_{x^*} = 0$

A.3 $J(x) \rightarrow -\infty$ if $\|x\| \rightarrow \infty$.

These assumptions imply that upper and lower bounds on $J(x)$ exist such that $\gamma_1(\|x - x^*\|) \leq -J(x) + J(x^*) \leq \gamma_2(\|x - x^*\|)$ with functions $\gamma_1(\|x\|), \gamma_2(\|x\|) \in K_\infty$. This is necessary as $J(x)$ will be used as Lyapunov function later on. The maximum of $J(x)$ coincides with the point $x^*$. 
### 3.2.1 Single Integrator

The first results are for a simple vehicle model, the single-integrator. The scheme is similar to the scheme of Section 3.1 but with a more general nonlinear map.

\[ \dot{x}_1 = c(J(x) - eh)\sqrt{\omega}\sin(\omega t - \phi) + \alpha\sqrt{\omega}\cos(\omega t) \]
\[ \dot{x}_2 = -c(J(x) - eh)\sqrt{\omega}\cos(\omega t - \phi) + \alpha\sqrt{\omega}\sin(\omega t) \]
\[ \dot{e} = -he + J(x) \]

(3.22)

with \( x = (x_1, x_2)^\top \) the position of the vehicle and \( e \) the state of the filter \( \frac{x}{s+h} \), \( h > 0 \).

**Lemma 8.** Consider the extremum seeking feedback in equation (3.22) and the system

\[
\begin{pmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{e}
\end{pmatrix} = \begin{pmatrix}
\frac{1}{2}(ca\nabla_{x_1} J \cos(\phi) - ca\nabla_{x_2} J \sin(\phi) - c^2\nabla_{x_1} J \cos(\phi)(J - \dot{e}h)) \\
\frac{1}{2}(ca\nabla_{x_2} J \cos(\phi) + ca\nabla_{x_1} J \sin(\phi) + c^2\nabla_{x_2} J \cos(\phi)(J - \dot{e}h)) \\
-\dot{e}h + J
\end{pmatrix},
\]  

(3.23)

\((\bar{x}, \bar{e})^\top |_0 = (x, e)^\top |_0\).

For sufficiently large \( \omega \) the trajectory of the original system (3.22) is bounded by solutions of the System (3.23), such that

\[ ||(x, e)^\top - (\bar{x}, \bar{e})^\top||_{C[0,T]} \leq \Delta_\epsilon \]

with \( \epsilon = \frac{1}{2}\frac{2\pi}{T} \) and \( \Delta_\epsilon \) a parameter with \( \lim_{\epsilon \to 0} \Delta_\epsilon = 0 \).

**Proof.** By using the identities \( \sin(x - y) = \sin(x)\cos(y) - \cos(x)\sin(y) \) and \( \cos(x - y) = \cos(x)\cos(y) + \sin(x)\sin(y) \) this yields to

\[ \dot{x}_1 = c(J - eh)\sqrt{\omega}\sin(\omega t)\cos(\phi) - c(J - eh)\sqrt{\omega}\cos(\omega t)\sin(\phi) + \alpha\sqrt{\omega}\cos(\omega t) \]
\[ \dot{x}_2 = -c(J - eh)\sqrt{\omega}\cos(\omega t)\cos(\phi) - c(J - eh)\sqrt{\omega}\sin(\omega t)\sin(\phi) + \alpha\sqrt{\omega}\sin(\omega t) \]
\[ \dot{e} = -e\dot{h} + J. \]

(3.24)
Writing (3.24) as an input-affine system with inputs $u_1$ and $u_2$:

\[
\begin{align*}
\dot{x}_1 &= \begin{pmatrix} c(J - eh) \cos(\phi) \\ \alpha - c(J - eh) \sin(\phi) \end{pmatrix} \sqrt{\omega(\omega t)} + \begin{pmatrix} \alpha - c(J - eh) \sin(\phi) \\ -c(J - eh) \cos(\phi) \end{pmatrix} \sqrt{\omega(\omega t)} \\
\dot{x}_2 &= \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ -eh + J \end{pmatrix} \frac{1}{u_0}.
\end{align*}
\]

The Lie bracket of the vector fields $f$ and $g$ can be calculated as follows:

\[
[f, g] = \frac{\partial g}{\partial x} f - \frac{\partial f}{\partial x} g
\]

\[
\begin{align*}
&= \begin{pmatrix} c \nabla \bar{x}_1, J \sin(\phi) \\ -c \nabla \bar{x}_1, J \cos(\phi) \end{pmatrix} + \begin{pmatrix} -c \nabla \bar{x}_2, J \sin(\phi) \\ -c \nabla \bar{x}_2, J \cos(\phi) \end{pmatrix} \begin{pmatrix} 0 \\ 0 \end{pmatrix} + \begin{pmatrix} \alpha - c(J - eh) \sin(\phi) \\ -c(J - eh) \cos(\phi) \end{pmatrix} \\
&= \begin{pmatrix} -co \nabla \bar{x}_1, J \cos(\phi) + co \nabla \bar{x}_2, J \sin(\phi) + c^2 \nabla \bar{x}_2, J \cos(\phi)(J - eh)) \\ -co \nabla \bar{x}_2, J \cos(\phi) - co \nabla \bar{x}_1, J \sin(\phi) - c^2 \nabla \bar{x}_1, J \cos(\phi)(J - eh)) \end{pmatrix}.
\end{align*}
\]

Define for some arbitrary $T \in (0, \infty)$, $\tilde{\omega} := \omega \frac{T}{2\pi}$, and $\omega = \omega \frac{2\pi}{T}$. Furthermore, consider the inputs $\tilde{u}_1 = \sqrt{\tilde{\omega}} \sqrt{\frac{2\pi}{T}} \sin(\tilde{\omega} t \frac{2\pi}{T})$ and $\tilde{u}_2 = \sqrt{\tilde{\omega}} \sqrt{\frac{2\pi}{T}} \cos(\tilde{\omega} t \frac{2\pi}{T})$. By using Theorem 6 and $\theta := \frac{t}{T} = \tilde{\omega} t$, one obtains for the approximative system:

\[
\begin{pmatrix} \dot{\tilde{x}}_1 \\ \dot{\tilde{x}}_2 \\ \dot{\tilde{e}} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -\tilde{e}h + J \end{pmatrix} + \frac{1}{T} \nu_{1,2}[f, g]
\]

(3.27)

where $\nu_{1,2}$ is defined as:

\[
\nu_{1,2} = \int_0^T \int_0^{2\pi} \frac{2\pi}{T} \sin(\tau \frac{2\pi}{T}) \cos(\theta \frac{2\pi}{T}) d\tau d\theta = -\frac{T}{2}.
\]

(3.28)

Hence,:

\[
\begin{pmatrix} \dot{\tilde{x}}_1 \\ \dot{\tilde{x}}_2 \\ \dot{\tilde{e}} \end{pmatrix} = \begin{pmatrix} \frac{1}{T}(co \nabla \bar{x}_1, J \cos(\phi) - co \nabla \bar{x}_2, J \sin(\phi) - c^2 \nabla \bar{x}_2, J \cos(\phi)(J - \tilde{e}h)) \\ \frac{1}{T}(co \nabla \bar{x}_2, J \cos(\phi) + co \nabla \bar{x}_1, J \sin(\phi) + c^2 \nabla \bar{x}_1, J \cos(\phi)(J - \tilde{e}h)) \end{pmatrix}.
\]

(3.29)

By results of Theorem 6, the result now follows. \qed

The next theorem states the stability of the introduced extremum seeking scheme. Before stating it, some additional assumptions on the parameters have to be made:

a.1 $\alpha > 0$
Theorem 7. Under the Assumptions A.1-A.3 and a.1-a.3 the point $x^*$ is practically globally uniformly asymptotically stable for the system in Eq. (3.22).

Proof. Consider the Lie bracket system

$$
\begin{pmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{e}
\end{pmatrix} =
\begin{pmatrix}
\frac{1}{2}(c_α \nabla x_1 J \cos(\phi) - c_α \nabla x_2 J \sin(\phi) - c^2 \nabla x_2 J \cos(\phi)(J - \ddot{e}h)) \\
\frac{1}{2}(c_α \nabla x_2 J \cos(\phi) + c_α \nabla x_1 J \sin(\phi) + c^2 \nabla x_1 J \cos(\phi)(J - \ddot{e}h)) \\
-\ddot{e}h + J
\end{pmatrix}
$$

(3.30)

and divide the system into two interconnected subsystems $\dot{x} = (\dot{x}_1, \dot{x}_2)^T = f_1(x, e)$ and $\dot{e} = f_2(x, e)$.

Taking the Lyapunov function candidate $V = -J(\bar{x}) + J(x^*)$ that is under the assumptions a valid Lyapunov function. One obtains for the first subsystem

$$
\dot{V} = -\nabla x_1 J \ddot{x}_1 - \nabla x_2 J \ddot{x}_2
$$

$$
= -c_α(\nabla x_1 J)^2 \cos(\phi) - c_α(\nabla x_2 J)^2 \cos(\phi)
$$

$$
- c^2 J(\bar{x} - \dot{e}h)\nabla x_1 J \nabla x_2 J \cos(\phi) + c^2 J(\bar{x} - \dot{e}h)\nabla x_1 J \nabla x_2 J \cos(\phi)
$$

$$
- c_α \nabla x_1 J \nabla x_2 J \sin(\phi) + c_α \nabla x_2 J \nabla x_1 J \sin(\phi)
$$

(3.31)

$$
\leq 0 \quad \forall \bar{x} \neq 0
$$

one can conclude that the average system is globally uniformly asymptotically stable, independent of $e$. Performing the change of variables $\ddot{e} = \ddot{e} - J(x^*)$ and examining the stability of the second subsystem by taking the Lyapunov function $V = \frac{1}{2} \ddot{e}^2$ one obtains

$$
\dot{V} = \ddot{e} \ddot{e}
$$

$$
= \ddot{e}(-\ddot{e}h + J(\bar{x}) - J(x^*))
$$

$$
= -h \ddot{e}^2 + \ddot{e} J
$$

(3.32)

$$
\leq -h ||\ddot{e}||^2 + ||\ddot{e}|| ||J||
$$

$$
= - (1 - \theta)h ||\ddot{e}||^2 - \theta h ||\ddot{e}||^2 + ||\ddot{e}|| ||J||
$$

for $0 < \theta < 1$

$$
\leq - (1 - \theta)h ||\ddot{e}||^2, \quad \forall ||\ddot{e}|| \geq \frac{|J|}{\theta h}
$$

Therefore, the second $\ddot{e} = f_2(x, e)$ subsystem is by Theorem input-to-state stable with respect to $J$ as input.

One can now use Lemma to conclude that the feedback connection of subsystem 1 and subsystem 2 is globally uniformly asymptotically stable.

By the results of Lemma that assures Hypothesis namely that the trajectories of the Lie bracket system and the original system are bounded, and Theorem one can conclude that the original system (3.22) is practically globally uniformly asymptotically stable. \qed
Remark   Note that because of the state transformation of $\bar{e} \rightarrow \tilde{e}$, the filter state will not go to the origin but to the value $J(x^*)$. This is important to mention, as asymptotic stability implies usually that the state goes to the origin.

Remark   The extremum seeking feedback in Figure 3.3 shows a slightly different version of the extremum seeking in Section 3.1 where the gains of the amplitudes are multiplied with $\sqrt{\omega}$. The simplifications made at the beginning of the chapter are not restrictive. As $\omega$ is a fixed parameter, one could define $\tilde{\alpha} = \alpha \sqrt{\omega}$ and $\tilde{c} = \frac{c}{\sqrt{\omega}}$ that finally leads to the original extremum seeking in Section 3.1.

An important point is the difference of the main ideas for the proofs of the extremum seeking in Section 3.1 where the average theorem (cf. [10]) was used, and in this case, where the main idea is to use Lie brackets in order to get an approximation of the original system.

In the presented approach it is not necessary to assume a quadratic, decoupled form of the nonlinear map. Nevertheless, there are relations between the two approaches. Both of them neglect the higher order terms. This can be seen as some implication concerning the Lie brackets, whereas it has to be made a-priori in the proof in Section 3.1 by using a quadratic map.

The main difference is however, that Lie brackets take coupling between the $x_1$- and the $x_2$-coordinate into account. This gives a better approximation of the qualitative behavior of the trajectories, as it can be seen in the following section where the original system and the Lie bracket system are compared for different nonlinear maps.

3.2.2 Simulation Results for the Single Integrator

The simulations in this section build up on the theory of the previous section and visualize the ideas behind the proofs.

The first case uses a simple quadratic map, that admits an extremum point at $(x^*, y^*) = (0, 0)$, hence it fulfills all the assumptions that were made in Theorem 8. Two different choices of $\omega$ are compared. In Figure 3.4 one can see on the left the simulation results for $\omega = 10$ and on the right for $\omega = 50$, in both examples the values $\alpha = 0.25$, $c = 1$ and $\phi = 0$ were used. The trajectory of the Lie bracket system is drawn as dashed line and the trajectory of the original system as solid line.

This simulation shows very nice that even for a small $\omega$ the Lie bracket system gives a good approximation of the qualitative behavior of the original system. For larger $\omega$, the distance of the trajectories decreases.

In the next example a highly nonlinear map, namely an exponential of a quadratic polynomial will be used. Again, the previously chosen values for $\omega$ are compared. In Figure 3.5 one can see similar results as in the case before. In this example the Lie bracket system approaches the original one, even for lower values of $\omega$. That is due to the fact, that the higher order derivatives of the function $J(x, y)$ are decaying much faster than for the polynomial case.
Figure 3.4: Single Integrator with $J(x_1, x_2) = -x_1^2 - x_2^2$

Figure 3.5: Single Integrator with $J(x_1, x_2) = e^{-x_1^2 - 5x_2^2} - 1$
3.2. INTERPRETATION AS LIE BRACKET MOTION

3.2.3 Double-Integrator

The previous calculations can also be applied to double-integrator dynamics. This is very useful, as there is a large class of systems, that can be transformed to a double-integrator system by feedback-linearization.

Consider the extremum seeking scheme in Figure 3.6 (cf. [29]) with the following state-space equations

\[
\begin{align*}
\dot{x}_1 &= \tilde{v}_{x_1} \\
\dot{v}_{x_1} &= -\alpha \omega \sqrt{\omega} \sin(\omega t) + w_{x_1} + c(J - eh) \sqrt{\omega} \sin(\omega t - \phi) \\
\dot{w}_{x_1} &= p \omega_{x_1} + (p - z)c(J - eh) \sqrt{\omega} \sin(\omega t - \phi) \\
\dot{x}_2 &= \tilde{v}_{x_2} \\
\dot{v}_{x_2} &= \alpha \omega \sqrt{\omega} \cos(\omega t) + w_{x_2} - c(J - eh) \sqrt{\omega} \cos(\omega t - \phi) \\
\dot{w}_{x_2} &= p \omega_{x_2} - (p - z)c(J - eh) \sqrt{\omega} \cos(\omega t - \phi) \\
\dot{e} &= -eh + J.
\end{align*}
\]

with \( x = (x_1, x_2)^\top \) the position of the vehicle and \( e \) the state of the filter \( \frac{x}{s + h}, h > 0 \).

By using the identities \( \sin(x - y) = \sin(x) \cos(y) - \cos(x) \sin(y) \) and \( \cos(x - y) = \cos(x) \cos(y) + \sin(x) \sin(y) \) and performing the change of variables \( u_{x_1} := \tilde{v}_{x_1} - \alpha \sqrt{\omega} \cos(\omega t) \) and \( v_{x_2} := \tilde{v}_{x_2} - \alpha \sqrt{\omega} \sin(\omega t) \), the equations can be rewritten in the following form

\[
\begin{align*}
\dot{x}_1 &= v_{x_1} + \alpha \sqrt{\omega} \cos(\omega t) \\
v_{x_1} &= w_{x_1} + c(J - eh) \cos(\phi) \sqrt{\omega} \sin(\omega t) - c(J - eh) \sin(\phi) \sqrt{\omega} \cos(\omega t) \\
\dot{w}_{x_1} &= p \omega_{x_1} + (p - z)c(J - eh) \cos(\phi) \sqrt{\omega} \sin(\omega t) - (p - z)c(J - eh) \sin(\phi) \sqrt{\omega} \cos(\omega t) \\
\dot{x}_2 &= v_{x_2} + \alpha \sqrt{\omega} \sin(\omega t) \\
v_{x_2} &= w_{x_2} - c(J - eh) \cos(\phi) \sqrt{\omega} \cos(\omega t) - c(J - eh) \sin(\phi) \sqrt{\omega} \sin(\omega t) \\
\dot{w}_{x_2} &= p \omega_{x_2} - (p - z)c(J - eh) \cos(\phi) \sqrt{\omega} \cos(\omega t) - (p - z)c(J - eh) \sin(\phi) \sqrt{\omega} \sin(\omega t) \\
\dot{e} &= -eh + J.
\end{align*}
\]
Lemma 9. Consider the extremum seeking feedback in equation (3.34) and the system

\[
\begin{pmatrix}
\dot{x}_1 \\
\dot{v}_{x_1} \\
\dot{w}_{x_1} \\
\dot{x}_2 \\
\dot{v}_{x_2} \\
\dot{w}_{x_2} \\
\dot{\epsilon}
\end{pmatrix} =
\begin{pmatrix}
v_{x_1} \\
w_{x_1} \\
p_{x_1} \\
v_{x_2} \\
w_{x_2} \\
p_{x_2} \\
\epsilon
\end{pmatrix} = \begin{pmatrix}
0 \\
\alpha c\nabla_{x_1} J(\phi) + \alpha c\nabla_{x_2} J(\sin(\phi)) \\
\alpha(p - z)c\nabla_{x_1} J(\phi) + \alpha(p - z)c\nabla_{x_2} J(\sin(\phi)) \\
0 \\
\alpha\nabla_{x_2} J(\cos(\phi)) - \alpha\nabla_{x_1} J(\sin(\phi)) \\
\alpha(p - z)c\nabla_{x_2} J(\cos(\phi)) - \alpha(p - z)c\nabla_{x_1} J(\sin(\phi)) \\
0
\end{pmatrix}
\]

(3.35)

\[
(\bar{x}, \bar{e})^\top |_0 = (x, e)^\top |_0.
\]

For sufficiently large \( \omega \), the trajectory of the original system (3.34) is bounded by solutions of the System (3.35), such that

\[
||(x, e)^\top - (\bar{x}, \bar{e})^\top||_{C[0,T]} \leq \Delta_\epsilon
\]

with \( \epsilon = \frac{1}{\omega^2} \) and \( \Delta_\epsilon \) a parameter with \( \lim_{\epsilon \to 0} \Delta_\epsilon = 0. \)

Proof. Writing the system in equation (3.34) as an input-affine system

\[
\begin{pmatrix}
\dot{x}_1 \\
\dot{v}_{x_1} \\
\dot{w}_{x_1} \\
\dot{x}_2 \\
\dot{v}_{x_2} \\
\dot{w}_{x_2} \\
\dot{\epsilon}
\end{pmatrix} =
\begin{pmatrix}
0 \\
c(J - \epsilon h)\cos(\phi) \\
\alpha \\
-c(J - \epsilon h)\sin(\phi) \\
-(p - z)c(J - \epsilon h)\sin(\phi) \\
0
\end{pmatrix}
\]

\[
\begin{pmatrix}
f(x) \\
g(x)
\end{pmatrix} = \begin{pmatrix}
\sqrt{\omega}\sin(\omega t) \\
\sqrt{\omega}\cos(\omega t)
\end{pmatrix} \begin{pmatrix}
\alpha \\
0
\end{pmatrix}
\]

(3.36)
and calculating the Lie-Bracket
\[ [f, g] = \frac{\partial g}{\partial x} f - \frac{\partial f}{\partial x} g \]

\[
\begin{pmatrix}
-\epsilon w_{x_3} J \sin(\phi) \\
-\epsilon w_{x_3} J \sin(\phi) \\
-\epsilon w_{x_3} J \cos(\phi) \\
-\epsilon w_{x_3} J \cos(\phi) \\
\end{pmatrix} 
= 
\begin{pmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 \\
\end{pmatrix} 
\begin{pmatrix}
0 \\
0 \\
0 \\
0 \\
\end{pmatrix} + 
\begin{pmatrix}
0 \\
0 \\
0 \\
0 \\
\end{pmatrix}
\]

\[ (3.37) \]

Define for some arbitrary \( T \in (0, \infty) \), \( \bar{\omega} := \omega \frac{T}{2\pi} \), and \( \omega = \bar{\omega} \frac{2\pi}{T} \). Furthermore, consider the inputs \( \bar{u}_1 = \sqrt{\bar{\omega}} \sqrt{\frac{2\pi}{T}} \sin(\bar{\omega} t \frac{2\pi}{T}) \) and \( \bar{u}_2 = \sqrt{\bar{\omega}} \sqrt{\frac{2\pi}{T}} \cos(\bar{\omega} t \frac{2\pi}{T}) \). By using Theorem 6 and \( \theta := \frac{\tau}{\epsilon} = \bar{\omega} t \), one obtains for the approximative system

\[
\begin{pmatrix}
\dot{\tilde{v}}_{x_1} \\
\dot{\tilde{v}}_{x_2} \\
\dot{\tilde{v}}_{x_2} \\
\dot{\tilde{v}}_{x_2} \\
\end{pmatrix} = 
\begin{pmatrix}
\tilde{v}_x \sin(\bar{\omega} t \frac{2\pi}{T}) \\
\tilde{v}_x \cos(\bar{\omega} t \frac{2\pi}{T}) \\
\tilde{v}_x \sin(\bar{\omega} t \frac{2\pi}{T}) \\
\tilde{v}_x \cos(\bar{\omega} t \frac{2\pi}{T}) \\
\end{pmatrix} + \frac{1}{T} \nu_{1.2}[f, g] 
\]

\[ (3.38) \]

where \( \nu_{1.2} \) is defined as

\[ \nu_{1.2} = \int_0^T \int_0^{\bar{\omega} \frac{T}{2\pi}} \sin(\frac{2\pi}{T}) \cos(\frac{2\pi}{T}) d\tau d\theta = -\frac{T}{2}. \]

\[ (3.39) \]

Hence, the approximative system yields to

\[
\dot{x} = 
\begin{pmatrix}
\dot{v}_{x_1} \\
\dot{v}_{x_2} \\
\dot{v}_{x_2} \\
\dot{v}_{x_2} \\
\end{pmatrix} + \frac{1}{2} 
\begin{pmatrix}
\alpha \epsilon w_{x_3} J \cos(\phi) - \alpha \epsilon w_{x_3} J \sin(\phi) \\
\alpha (p - z) \epsilon w_{x_3} J \cos(\phi) + \alpha (p - z) \epsilon w_{x_3} J \sin(\phi) \\
\alpha \epsilon w_{x_3} J \cos(\phi) - \alpha \epsilon w_{x_3} J \sin(\phi) \\
\alpha (p - z) \epsilon w_{x_3} J \cos(\phi) + \alpha (p - z) \epsilon w_{x_3} J \sin(\phi) \\
\end{pmatrix} 
\]

\[ (3.40) \]

By results of Theorem 6, the result now follows. \( \square \)
As before, practical stability is shown for the system in equation (3.34). Consider first the following assumptions on the parameters:

b.1 $\alpha > 0$

b.2 $c > 0$

b.3 $-\frac{\pi}{2} < \phi < \frac{\pi}{2}$

b.4 $p < 0$

b.5 $z < 0$

b.6 $(p - z) < 0$

b.7 $-\sin(\phi) \left( \int_{x_1}^{x_1} \nabla x_2 J - \int_{x_2}^{x_2} \nabla x_1 J \right) \geq 0$.

These inequalities will lead to a positive definite Lyapunov function.

**Theorem 8.** Under the Assumptions A.1-A.3 and b.1-b.7 the point $x^*$ is practically globally uniformly asymptotically stable for the system in Eq. (3.34).

**Proof.** Consider the approximative system

\[
\dot{x} = \begin{pmatrix}
\bar{v}_{x_1} \\
\bar{w}_{x_1} \\
p \bar{w}_{x_1} \\
\bar{v}_{x_2} \\
\bar{w}_{x_2} \\
p \bar{w}_{x_2} \\
-\bar{e}h + J
\end{pmatrix} + \frac{1}{2} \begin{pmatrix}
0 \\
\alpha c \nabla x_1 J \cos(\phi) + \alpha c \nabla x_2 J \sin(\phi) \\
\alpha (p - z) c \nabla x_1 J \cos(\phi) + \alpha (p - z) c \nabla x_2 J \sin(\phi) \\
0 \\
\alpha c \nabla x_2 J \cos(\phi) - \alpha c \nabla x_1 J \sin(\phi) \\
\alpha (p - z) c \nabla x_2 J \cos(\phi) - \alpha (p - z) c \nabla x_1 J \sin(\phi) \\
0
\end{pmatrix} \tag{3.41}
\]

and choose the following Lyapunov function for the system

\[
V = a(-J(x) + J(x^*)) \cos(\phi) - a \sin(\phi) \left( \int_{x_1}^{x_1} \nabla x_2 J - \int_{x_2}^{x_2} \nabla x_1 J \right) + \frac{b}{2} (\bar{v}^2_{x_1} + \bar{v}^2_{x_2}) + \bar{c} (v_{x_1} w_{x_1} + v_{x_2} w_{x_2}) + \frac{d}{2} (w^2_{x_1} + w^2_{x_2}). \tag{3.42}
\]

Under the assumptions, this is a valid Lyapunov function as it is positive definite, decrescent and radially unbounded. For the derivative of $V$ along the trajectories of the system, one
obtains

\[
\dot{V} = -a \nabla_x J \dot{\phi} \cos \phi - a \nabla_x J \dot{\phi} \sin \phi + b v_x \dot{\phi} + \tilde{c} \dot{\phi} w_x + \tilde{c} v_x \dot{w}_x + d \dot{w}_x w_x + \ldots
\]

\[
= -a \nabla_x J v_x \cos \phi - a \nabla_x J v_x \sin \phi
\]

\[
+ b v_x w_x + \frac{1}{2} b v_x \alpha c \nabla_x J \cos \phi + \frac{1}{2} b v_x \alpha c \nabla_x J \sin \phi
\]

\[
+ \tilde{c} w_x^2 + \frac{1}{2} \tilde{c} w_x \alpha c \nabla_x J \cos \phi + \frac{1}{2} \tilde{c} w_x \alpha c \nabla_x J \sin \phi
\]

\[
+ \tilde{c} p v_x w_x + \frac{1}{2} \tilde{c} v_x \alpha (p - z) c \nabla_x J \cos \phi + \frac{1}{2} \tilde{c} v_x \alpha (p - z) c \nabla_x J \sin \phi
\]

\[
+ d p w_x^2 + \frac{1}{2} w_x \alpha \alpha (p - z) c \nabla_x J \cos \phi + \frac{1}{2} w_x \alpha (p - z) c \nabla_y J \sin \phi
\]

\[
+ \ldots
\]

\[
= (v_x \nabla_x J + v_x \nabla_x J) \cos \phi \left( -a + \frac{1}{2} b \alpha c + \frac{1}{2} \tilde{c} \alpha (p - z) c \right)
\]

\[
+ (v_x \nabla_x J - v_x \nabla_x J) \sin \phi \left( -a + \frac{1}{2} b \alpha c + \frac{1}{2} \tilde{c} \alpha (p - z) c \right)
\]

\[
+ (w_x v_x + v_x w_x) (b + \tilde{c} p)
\]

\[
+ (w_x^2 + w_x^2) (\tilde{c} + dp)
\]

\[
+ (w_x \nabla_x J + w_x \nabla_x J) \cos \phi \left( \frac{1}{2} \tilde{c} \alpha c + \frac{1}{2} \alpha (p - z) c \right)
\]

\[
+ (w_x \nabla_x J + w_x \nabla_x J) \sin \phi \left( \frac{1}{2} \tilde{c} \alpha c + \frac{1}{2} \alpha (p - z) c \right).
\]

Choosing 1. \( b = -\tilde{c} p \)

\[
\dot{V}_x = (v_x \nabla_x J + v_x \nabla_x J) \cos \phi \left( -a - \frac{1}{2} \tilde{c} \alpha z c \right)
\]

\[
+ (v_x \nabla_x J - v_x \nabla_x J) \sin \phi \left( -a + \frac{1}{2} \tilde{c} \alpha z c \right)
\]

\[
+ (w_x^2 + w_x^2) (\tilde{c} + dp)
\]

\[
+ (w_x \nabla_x J + w_x \nabla_x J) \cos \phi \left( \frac{1}{2} \tilde{c} \alpha c + \frac{1}{2} \alpha (p - z) c \right)
\]

\[
+ (w_x \nabla_x J + w_x \nabla_x J) \sin \phi \left( \frac{1}{2} \tilde{c} \alpha c + \frac{1}{2} \alpha (p - z) c \right).
\]

2. \( a = -\frac{1}{2} \tilde{c} \alpha z c \)

\[
\dot{V}_x = (w_x^2 + w_x^2) (\tilde{c} + dp)
\]

\[
+ (w_x \nabla_x J + w_x \nabla_x J) \cos \phi \left( \frac{1}{2} \tilde{c} \alpha c + \frac{1}{2} \alpha (p - z) c \right)
\]

\[
+ (w_x \nabla_x J + w_x \nabla_x J) \sin \phi \left( \frac{1}{2} \tilde{c} \alpha c + \frac{1}{2} \alpha (p - z) c \right).
\]
and with $\tilde{c} = -(p - z)$ this leads to

$$
\dot{V} = (w_{x_1} + w_{x_2}) ((d - 1)p + z) \quad \text{if} \quad (d - 1)p + z < 0.
$$

This term is only negative semi-definite as only the states $w_{x_1}$ and $w_{x_2}$ appear. By using LaSalle’s Invariance Principle, one obtains

$$
\dot{V} = 0 \Rightarrow w_{x_1} = w_{x_2} = 0 \Rightarrow \nabla_x J = 0 \Rightarrow (\bar{x}_1, \bar{x}_2)^\top = (x_1^*, x_2^*)^\top
$$

To make sure that the chosen Lyapunov function is positive definite and its derivative along the trajectories negative semi-definite, the following inequalities must be fulfilled

$$
(d - 1)p + z < 0
$$

Choosing $d = 1$, these inequalities are fulfilled if the parameters are chosen according to assumptions b.1-b.7.

Consider now the filter-state $\tilde{e}$. Performing the change of variables $\tilde{e} = \bar{e} - J(x^*)$ and examining the stability of the second subsystem by taking the Lyapunov function $V = \frac{1}{2}\tilde{e}^2$ one obtains

$$
\dot{V} = \tilde{e} \dot{\tilde{e}}
$$

$$
= \tilde{e}(-\tilde{e} \bar{h} + J(\bar{x}) - J(x^*))
$$

$$
= - \bar{h} \tilde{e}^2 + \tilde{e} \bar{J}
$$

$$
\leq - \bar{h} ||\tilde{e}||^2 + ||\tilde{e}|| ||\bar{J}||
$$

$$
= - (1 - \theta)\bar{h} ||\tilde{e}||^2 - \theta \bar{h} ||\tilde{e}|| ||\bar{J}||
$$

$$
\leq - (1 - \theta)\bar{h} ||\tilde{e}||^2, \quad \forall ||\tilde{e}|| \geq \frac{||\bar{J}||}{\theta \bar{h}}
$$

Therefore, the second $\dot{\tilde{e}} = f_2(x, e)$ subsystem is by Theorem input-to-state stable with respect to $\bar{J}$ as input.

One can now use Lemma to conclude that the feedback connection of subsystem 1 and subsystem 2 is globally uniformly asymptotically stable.

By the results of Lemma that assures Hypothesis namely that the trajectories of the Lie bracket system and the original system are bounded, and Theorem one can conclude that the original system (3.34) is practically globally uniformly asymptotically stable.

**Remark** The parameters of the filter $\frac{s - z}{s - p}$ have to be chosen in such a way, that it can go fast enough to steady-state. As it introduces a phase-shift to the sinusoids, the system first increases its distance to the origin. Depending on the slope of the function $J(x)$, this can lead to numerical problems if the system goes into a region of a smaller slope but with a large value of $J(x)$. 

\[\square\]
Remark Assumption b.7 is necessary as the phase shift of the sinusoids leads to a coupling of the gradient with respect to the $x_1$-direction and the gradient with respect to the $x_2$-direction. For $\phi = 0$ the Assumption b.7 is always fulfilled as this is the trivial case.

Remark Because of the state transformation $v_{x_1} := \tilde{v}_{x_1} - \alpha \sqrt{\omega} \cos \omega t$ and $v_{x_2} := \tilde{v}_{x_2} - \alpha \sqrt{\omega} \sin \omega t$ the initial conditions of the Lie bracket system have to be adjusted, such that $\begin{pmatrix} \bar{x}_1, \bar{v}_{x_1}, \bar{w}_{x_1}, \bar{x}_2, \bar{v}_{x_2}, \bar{w}_{x_2}, e \end{pmatrix}^\top |_0 = (x_1, v_{x_1} - \alpha \sqrt{\omega}, w_{x_1}, x_2, v_{x_2}, w_{x_2}, e)^\top |_0$. This is due to the fact that the system (3.34) is only a transformed version of the system (3.33). This state transformation comes together with some problems. It introduces the parameter $\omega$ in the initial conditions of the Lie bracket system. One cannot use the original system as basis for the analysis as this would imply that the initial conditions of the Lie bracket system would go to infinity as $\omega$ goes to infinity. Therefore, the analysis is done for the system (3.34).

Nevertheless, one can conclude that for a fixed parameter $\omega$ the system (3.33) must also be practically stable, as the state-transformation is valid for all $t$ and always invertible. Therefore the system in (3.33) is equivalent to the system in (3.34).

3.2.4 Simulation Results for the Double Integrator

This section should give some insight to the special behavior of the original double-integrator extremum seeking in equation (3.33) in comparison to the Lie bracket system.

The first case uses a simple quadratic map, that admits an extremum point at $(x^*, y^*) = (0, 0)$, hence it fulfills all the assumptions that were made in Theorem 8. Two different choices of $\omega$ are compared. In Figure 3.4 one can see on the left the simulation results for $\omega = 10$ and on the right for $\omega = 50$, in both examples the values $\alpha = 0.25$, $c = 1$ and $\phi = 0$ as well as $p = -10$ and $z = -2$ were used. This assures that all assumptions especially assumption b.7 is fulfilled, as this can easily be achieved with $\phi = 0$. The trajectory of the Lie bracket system is drawn as dashed line and the trajectory of the original system as solid line.

![Simulation Results for the Double Integrator](image)

(a) $\omega = 10$

(b) $\omega = 50$

Figure 3.7: Double Integrator with $J(x_1, x_2) = -x_1^2 - x_2^2$

This simulation shows very nice that even for a small $\omega$ the Lie bracket system gives a good
approximation of the qualitative behavior of the original system. For larger $\omega$, the distance of the trajectories decreases.

In the next example a highly nonlinear map, namely an exponential of a quadratic polynomial will be used. Again, the previously chosen values for $\omega$ are compared. As for the single integrator the Lie bracket system approaches the original one, even for lower values of $\omega$. That is again due to the fact, that the higher order derivatives of the function $J(x, y)$ are decaying much faster than for the polynomial case.

$$J(x_1, x_2) = e^{-x_1^2 - 5x_2^2} - 1$$

Figure 3.8: Single Integrator with $J(x_1, x_2) = e^{-x_1^2 - 5x_2^2} - 1$
3.2.5 Extremum Seeking for the Unicycle Model

Further investigations showed that even more complicated, nonholonomic systems such as the unicycle model can be analyzed using Lie brackets. One possibility to do the feedback is shown in Figure 3.9 (cf. [28]). The unicycle model is given by the equations

\[
\begin{pmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{\theta}
\end{pmatrix} =
\begin{pmatrix}
u \\
\alpha \sqrt{\omega} \cos(\omega t) - \alpha \sqrt{\omega} \cos(\omega t) \\
\end{pmatrix}
\]

as usual in the literature.

**Lemma 10.** Consider the Unicycle Model in equation (3.50) with extremum seeking feedback

\[
u = (J(x) - e) \sqrt{\omega} \sin(\omega t - \phi) + \alpha \sqrt{\omega} \cos(\omega t),
\]

where \(e\) denotes the state of the filter \(\frac{s}{s+h}\), \(h>0\), and \(v = \Omega = \text{const.}\) a constant input.

Consider furthermore the system

\[
\begin{pmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{e}
\end{pmatrix} =
\begin{pmatrix}
\frac{1}{2}(c \nabla_{x_1} J \cos(\phi) \cos^2(\Omega t) + c \nabla_{x_2} J \cos(\phi) \cos(\Omega t) \sin(\Omega t)) \\
\frac{1}{2}(c \nabla_{x_2} J \cos(\phi) \sin^2(\Omega t) + c \nabla_{x_1} J \cos(\phi) \cos(\Omega t) \sin(\Omega t)) \\
-\dot{e} + J
\end{pmatrix}
\]

\((\bar{x}, \bar{e}) \in \mathbb{R}^\ast\)

For sufficiently large \(\omega \gg \Omega\), the trajectory of the original system (3.50) is bounded by solutions of the reduced Lie bracket System (3.23), such that

\[
|| (x, e)^T - (\bar{x}, \bar{e})^T ||_{C[0,T]} \leq \Delta_e
\]

with \(\epsilon = \frac{1}{\omega} \frac{2\pi}{T} \) and \(\Delta_e\) a parameter with \(\lim_{\epsilon \to 0} \Delta_e = 0\).

**Proof.** The procedure is the same as in the single integrator case. By using the identity \(\sin(x - y) = \sin(x) \cos(y) - \cos(x) \sin(y)\) and \(\theta = \Omega\) one obtains

\[
\begin{align*}
\dot{x}_1 &= (c(J - e) \sqrt{\omega} \sin(\omega t) \cos(\phi) - c(J - e) \sqrt{\omega} \cos(\omega t) \sin(\phi) + \alpha \sqrt{\omega} \cos(\omega t)) \cos(\theta) \\
\dot{x}_2 &= (c(J - e) \sqrt{\omega} \sin(\omega t) \cos(\phi) - c(J - e) \sqrt{\omega} \cos(\omega t) \sin(\phi) + \alpha \sqrt{\omega} \cos(\omega t)) \sin(\theta) \\
\dot{\theta} &= \Omega \\
\dot{e} &= -e + J
\end{align*}
\]
The system equations are written as an input-affine system

\[
\begin{pmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{\theta} \\
\dot{e}
\end{pmatrix} = \begin{pmatrix}
(c(J - \varepsilon h) \cos(\phi) \cos(\theta)) \\
(c(v - \varepsilon h) \cos(\phi) \sin(\theta)) \\
0 \\
0
\end{pmatrix} \begin{pmatrix}
\sqrt{\omega} \sin(\omega t)
\end{pmatrix} + \begin{pmatrix}
-(c(J - \varepsilon h) \sin(\phi) - \alpha) \cos(\theta) \\
-(c(J - \varepsilon h) \sin(\phi) - \alpha) \sin(\theta) \\
0 \\
0
\end{pmatrix} \begin{pmatrix}
\sqrt{\omega} \cos(\omega t)
\end{pmatrix}
\]

\[
+ \begin{pmatrix}
0 \\
0 \\
\Omega \\
-\varepsilon h + J
\end{pmatrix} \nu_1 \Omega
\]

(3.53)

with vector fields \(d(x, e, v)\), \(f(x, e)\) and \(g(x, e)\). The Lie bracket of \([f, g]\) can then be calculated to

\[
[f, g] = \begin{pmatrix}
-c \nabla_{x_1} J \sin(\phi) \cos(\theta) & -c \nabla_{x_2} J \sin(\phi) \cos(\theta) & * & * \\
-c \nabla_{x_1} J \sin(\phi) \sin(\theta) & -c \nabla_{x_2} J \sin(\phi) \sin(\theta) & * & * \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{pmatrix}
\begin{pmatrix}
(c(J - \varepsilon h) \cos(\phi) \cos(\theta)) \\
(c(v - \varepsilon h) \cos(\phi) \sin(\theta)) \\
0 \\
0
\end{pmatrix}
\]

(3.54)

Define for some arbitrary \(T \in (0, \infty), \tilde{\omega} := \omega \frac{T}{2\pi}\), and \(\omega = \tilde{\omega} \frac{2\pi}{T}\). Furthermore, consider the inputs \(\tilde{u}_1 = \sqrt{\omega} \sqrt{\frac{2\pi}{T}} \sin(\tilde{\omega} t \frac{2\pi}{T})\) and \(\tilde{u}_2 = \sqrt{\omega} \sqrt{\frac{2\pi}{T}} \cos(\tilde{\omega} t \frac{2\pi}{T})\). By using Theorem 6 and \(\theta := \frac{t}{\tilde{\omega}} = \tilde{\omega} t\), one obtains for the approximative system

\[
\begin{pmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{\theta} \\
\dot{e}
\end{pmatrix} = \begin{pmatrix}
0 \\
0 \\
v \\
-\varepsilon h + J
\end{pmatrix} + \frac{1}{T} \nu_1 \Omega \begin{pmatrix}
\dot{f} \\
\dot{g}
\end{pmatrix}
\]

(3.55)

where \(\nu_1, 2\) is defined as

\[
\nu_1, 2 = \int_0^T \int_0^\theta 2\pi \sin(\tau \frac{2\pi}{T}) \cos(\theta \frac{2\pi}{T}) d\tau d\theta = -\frac{T}{2}.
\]

(3.56)

Finally,

\[
\begin{pmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{\theta} \\
\dot{e}
\end{pmatrix} = \begin{pmatrix}
0 \\
0 \\
v \\
-\varepsilon h + J
\end{pmatrix} - \frac{1}{2} [f, g]
\]

(3.57)

with \(v = \Omega\) and \(\theta = \Omega t\) the approximative system results in:

\[
\begin{pmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{\theta} \\
\dot{e}
\end{pmatrix} = \begin{pmatrix}
\frac{1}{2} \cos(\phi)(\alpha \nabla_{x_1} J \cos^2(\Omega t) + \alpha \nabla_{x_2} J \sin(\Omega t) \cos(\Omega t)) \\
\frac{1}{2} \cos(\phi)(\alpha \nabla_{x_2} J \sin^2(\Omega t) + \alpha \nabla_{x_1} J \sin(\Omega t) \cos(\Omega t)) \\
0 \\
-\varepsilon h + J
\end{pmatrix}
\]

(3.58)
With Theorem 6, the result now follows.

Taking the same assumptions on the function $J(x)$ and the additional assumptions on the parameters

c.1 $\alpha > 0$
c.2 $c > 0$
c.3 $-\frac{\pi}{2} < \phi < \frac{\pi}{2}$
c.4 $\Omega \neq 0$

it is possible to state the next Theorem, that gives information about the asymptotic behavior of the original extremum seeking.

**Theorem 9.** The point $x^*$ is under the Assumptions A.1-A.3, c.1-c.4 and with inputs $u = (J(x) - \epsilon h)\sqrt{\omega} \sin \omega t + \alpha \sqrt{\omega} \cos \omega t$, $v = \Omega$ practically globally uniformly asymptotically stable for the system in Eq. (3.50).

**Proof.** Applying the same procedure as in the proof of Theorem 8, one first divides the Lie bracket system

$$
\begin{pmatrix}
\dot{\bar{x}}_1 \\
\dot{\bar{x}}_2 \\
\dot{\epsilon}
\end{pmatrix} =
\begin{pmatrix}
\frac{1}{2} \cos(\phi)(\alpha \nabla_{x_1} J C \cos^2(\Omega t) + \alpha \nabla_{x_2} J C \sin(\Omega t) \cos(\Omega t)) \\
\frac{1}{2} \cos(\phi)(\alpha \nabla_{x_2} J C \sin^2(\Omega t) + \alpha \nabla_{x_1} J C \sin(\Omega t) \cos(\Omega t)) \\
-\epsilon h + J
\end{pmatrix}
$$

(3.59)

into two subsystems, namely $\dot{\bar{x}} = f_1(x)$ and $\dot{\epsilon} = f_2(x, \epsilon)$.

Taking the Lyapunov function candidate $V = -J(\bar{x}) + J(x^*)$ that is under the assumptions on $J(\bar{x})$, a valid Lyapunov function. The derivation of $V$ along the trajectories of the first subsystem $\dot{\bar{x}}$ yields to:

$$
\dot{V} = -\nabla_{x_1} J \dot{\bar{x}}_1 - \nabla_{x_2} J \dot{\bar{x}}_2
$$

$$
= -\frac{1}{2} \cos(\phi)\alpha C (\nabla_{x_1} J)^2 \cos^2(\Omega t) - \frac{1}{2} \cos(\phi)\alpha C 2 \nabla_{x_1} J \nabla_{x_2} J \sin(\Omega t) \cos(\Omega t)
$$

$$
- \frac{1}{2} \cos(\phi)\alpha C (\nabla_{x_2} J)^2 \sin^2(\Omega t)
$$

$$
= -\frac{1}{2} \cos(\phi)\alpha C (\nabla_{x_1} J \cos(\Omega t) + \nabla_{x_2} J \sin(\Omega t))^2
$$

(3.60)

$$
\leq 0.
$$

This calculation shows that $\dot{V}$ is only negative semi-definite. This is due to the fact that the system is time-varying, and there are singular points in the state-space, where $\dot{x} = 0$, but which are no steady-states for the system. One possibility would be to find another Lyapunov function. This must be time-varying, because $\dot{V} = \frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} \dot{x}$, and as $\dot{x}$ is zero at some time instances, only a time-varying Lyapunov function can lead to $\dot{V} < 0$ for all time. A simpler method is to use some invariance-like argument.

Injecting $\nabla_{x_1} J \cos(\Omega t) + \nabla_{x_2} J \sin(\Omega t) = 0$ into the differential equation of $\dot{x} = f(x, t)$ yields to

$$
\begin{pmatrix}
\dot{x}_1 \\
\dot{x}_2
\end{pmatrix} =
\begin{pmatrix}
\alpha \nabla_{x_1} J C \cos^2(\Omega t) + \alpha \nabla_{x_1} J C \cos(\Omega t) \cos(\Omega t) \\
\alpha \nabla_{x_2} J C \sin^2(\Omega t) + \alpha \nabla_{x_2} J C \sin(\Omega t) \sin(\Omega t)
\end{pmatrix}
= \begin{pmatrix} 0 \\ 0 \end{pmatrix}.
$$

(3.61)
One can deduce that if \( x_1 = \text{const.} \) and \( x_2 = \text{const.} \) this implies that \( \nabla_{x_1} J(x_{10}, x_{20}) \) and \( \nabla_{x_2} J(x_{10}, x_{20}) \) are also constant. But as there are no constant values such that 
\[
\nabla_{x_1} J(x_{10}, x_{20}) \cos \Omega t + \nabla_{x_2} J(x_{10}, x_{20}) \sin \Omega t = 0, \quad \forall t \in \mathbb{R}_+ 
\]
except 
\[
\nabla_{x_1} J(x_{10}, x_{20}) = \nabla_{x_2} J(x_{10}, x_{20}) = 0,
\]
As \( V(x(t)) \) is monotonically decreasing and bounded from below, it must go to zero for \( t \to \infty \). Therefore the system approaches the maximum with \( t \to \infty \) for all initial conditions.

Performing again the change of variables \( \tilde{e} = \tilde{e} - J(x^*) \) and examining the stability of the second subsystem by taking the Lyapunov function 
\[
V = \frac{1}{2} \tilde{e}^2
\]
One obtains
\[
\dot{V} = \tilde{e} \dot{\tilde{e}} = \tilde{e} (-\tilde{e} + J(x) - J(x^*)) \leq -h ||\tilde{e}||^2 + ||\tilde{e}|| ||J|| \leq - (1 - \theta)h ||\tilde{e}||^2 - \theta h ||\tilde{e}||^2 + ||\tilde{e}|| ||J|| \quad \text{for} \quad 0 < \theta < 1
\]
Therefore, the second \( \dot{e} = f_2(x, e) \) subsystem is by Theorem 3 input-to-state stable with respect \( J(x) \) as input.

One can now use Lemma 4 to conclude that the feedback connection of subsystem 1 and subsystem 2 is globally uniformly asymptotically stable.

Together with Lemma 8 that assures Hypothesis 2 it is possible to use Theorem 2 and conclude that the original system (3.50) with the given feedback is practically globally uniformly asymptotically stable.

Remark An important fact is, that the value of \( \Omega \) needs usually to be smaller than the value of \( \omega \) because otherwise, it is not possible to use the argument of two different time-scales, as it is necessary for the usage of Theorem 6.

Remark In some cases there is no single extremum point. One can argue then, that local minima are, in contrast to local maxima, not stable fix-points. Hence it is always possible to converge to any of the local maxima as the original system has no fix-points because of the driving sinusoids that prevent the system from getting stuck in an unstable minimum.

Remark The parameter \( h \) of the filter determines mainly the time scale of the filter. The assumption that \( \omega \) is sufficiently large, is also meant in relation to the value of \( h \). For a large \( h \), the filter becomes very fast, and introduces an additional phase shift to \( \phi \). This has to be taken into account, as the phase shift of the filter is insignificant only for small values of \( h \) or for large values of \( \omega \).
Remark  The function $J(x)$ has to be smooth, because it is contained in the vector fields of the dynamical equations and Theorem 6 only holds for smooth vector fields.

3.2.6 Simulation Results for the Unicycle Model

The results in this section are based on the equations derived in the previous section. It considers the extremum seeking together with the unicycle model and a constant angular velocity $\Omega$. The same parameters as in the section before are chosen, whereas $\Omega$ is set to 1, the constants $\alpha = 0.01$, $c = 1$ and $\phi = 0$.

The first result is for the quadratic map and for $\omega = 10$ and $\omega = 50$ respectively. As in the section before, Figure 3.10 shows the same behavior as in the single integrator case, where a higher value of $\omega$ results in a better approximation by the Lie bracket system.

Figure 3.12 shows very good that the approximation becomes worse, if the value of $\Omega$ is close to the value of $\omega$, this is due to the fact that there are not two distinct time-scales in the system.

However, the property of practical stability is not lost even if the value of $\Omega$ is close to the value of $\omega$. 

![Simulation Results](image-url)
Figure 3.11: Unicycle Model with $J(x_1, x_2) = e^{-x_1^2 - 5x_2^2} - 1$

(a) $\omega = 10, \Omega = 1$

(b) $\omega = 50, \Omega = 1$

Figure 3.12: Unicycle Model with $J(x_1, x_2) = e^{-x_1^2 - 5x_2^2} - 1$

(a) $\omega = 50, \Omega = 45, J(x_1, x_2) = -x_1^2 - x_2^2$

(b) $\omega = 50, \Omega = 45, J(x_1, x_2) = e^{-x_1^2 - 5x_2^2} - 1$
3.2.7 Obstacle Avoidance

In the following section an extension to the extremum seeking is proposed. Usually the mission space is not convex but contains obstacles. It is desirable to avoid these obstacles without crashing into them. For this purpose one could imagine an additional control applied to the system. This section deals with one possibility of such a control.

It is assumed that the vehicle contains some kind of sensor that can measure the distance to the obstacle and furthermore that the obstacle has a round shape, is situated at \((x_a,y_a)\) and is sufficiently apart from the maximum \((x^*, y^*)\).

Every simple obstacle can be included in a circular region of sufficient size.

A similar approach as the authors in [16] is used. They defined a potential function

\[
V(x, y) = -\min(0, \frac{d_a^2 - R^2}{d_a^2 - r^2})^2
\]

with

\[
d_a = \sqrt{(x - x_a)^2 + (y - y_a)^2}
\]

where \(d_a\) denotes the actual distance from the vehicle to the center of the obstacle. The obstacle is assumed to be circular of radius \(r\), furthermore there is also a circular security region of radius \(R > r\).

![Figure 3.13: Model of Obstacle](image)

The gradient of \(V(x, y)\) is

\[
\frac{\partial V}{\partial x} = \begin{cases} 
0 & \text{if } d_a \geq R \\
-4 \frac{(R^2 - r^2)(d_a^2 - R^2)}{(d_a^2 - r^2)^3} (x - x_a) & \text{if } R > d_a > r \\
0 & \text{if } d_a < r
\end{cases}
\]

\[
\frac{\partial V}{\partial y} = \begin{cases} 
0 & \text{if } d_a \geq R \\
-4 \frac{(R^2 - r^2)(d_a^2 - R^2)}{(d_a^2 - r^2)^3} (y - y_a) & \text{if } R > d_a > r \\
0 & \text{if } d_a < r
\end{cases}
\]  

(3.64)

Obviously the potential as well as its gradient go to infinity by decreasing the distance to the obstacle. By adding the appropriate component of the gradient of \(V(x, y)\) to the input of the integrators like in Figure 3.14 one can prove that the obstacle will not be hit by the vehicle.
Theorem 10. Given a feedback like in Figure 3.14 with the gradient of a potential function defined in (3.64), then obstacle avoidance is assured.

Proof. Take $W = V(x, y) = \min(0, \frac{d_x^2 - R^2}{d_y^2 - r^2})^2$ as a Lyapunov function and rewrite the system equations in (3.5) as

$$
\dot{x}_1 = c_x(J(x) - eh)\sqrt{\omega}\sin(\omega t) + \alpha \sqrt{\omega}\cos(\omega t) + \frac{\partial V}{\partial x_1} = f_{x_1}(x, e) + \frac{\partial V}{\partial x_1}
$$

$$
\dot{x}_2 = -c_y(J(x) - eh)\sqrt{\omega}\cos(\omega t) + \alpha \sqrt{\omega}\sin(\omega t) + \frac{\partial V}{\partial y} = f_{x_2}(x, e) + \frac{\partial V}{\partial x_2}
$$

$$
\dot{e} = -he + J(x) = f_e(x, e)
$$

By calculating $\dot{W}$

$$
\dot{W} = \frac{\partial V}{\partial x_1}f_{x_1}(x, q) + \frac{\partial V}{\partial x_2}f_{x_2}(x, q) + \left(\frac{\partial V}{\partial x_1}\right)^2 + \left(\frac{\partial V}{\partial x_2}\right)^2 + e f_e(x, e)
$$

$$
= \frac{\partial V}{\partial x_1}f_{x_1}(x, y, q) + \frac{\partial V}{\partial x_2}f_{x_2}(x, q) + e f_e(x, e) + \left(\frac{\partial V}{\partial x_1}\right)^2 + \left(\frac{\partial V}{\partial x_2}\right)^2
$$

Obviously part $I$ is bounded because $f(x, y)$ is bounded, but part $II$ is positive and strictly increasing as $d_a \to r$ and hence there exists some $x$ and $y$ where part $II$ dominates $I$ and $\dot{W} > 0$. The point $(x_a, y_a)$ is unstable and there is no trajectory that can go inside $d_a \leq r$.

In order to assure convergence to the optimum in the presence of obstacles, one could imagine to use the Lie bracket system in order to show convergence.

The problem with the Lie bracket system is that the additionally added gradient vector field of the artificial potential function is not smooth, as it is defined only in some region. Therefore, it is not possible anymore to argue the convergence of the original system by using the Lie bracket system.

The simulation results in Chapter 5 show that there are cases where the relation between the original system and the Lie bracket system is lost when obstacles are present.
Furthermore, Theorem 10 only states that the system in not crashing into the obstacle, but it does not state anything about the convergence of the extremum seeking to the optimum.

### 3.2.8 Simulation Results for the Obstacle Avoidance

The results in this section are meant to show the behavior of the extremum seeking together with the obstacle avoidance feedback in comparison to the Lie bracket system. The Lie bracket system was equipped with the same obstacle avoidance control as the extremum seeking. This can actually be done by arguing that the obstacle avoidance feedback is a vector field, that is not time-varying and hence it would not change by applying the average method, as it is needed to be done in Theorem 6.

The equations of the Lie bracket system become therefore

$$
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{e}
\end{bmatrix} = 
\begin{bmatrix}
\alpha \omega c_x \nabla_x J - c_x c_y (J - ch) \nabla_{x_2} J + \frac{\partial V}{\partial x} \\
\alpha \omega c_y \nabla_{x_2} J + c_x c_y (J - ch) \nabla_{x_1} J + \frac{\partial V}{\partial y} \\
-e h + J
\end{bmatrix}
$$

In Figure 3.15a the obstacle is at position \((x_a, y_a) = (-0.83, 0.5)\) while it is at position \((x_a, y_a) = (-0.86, 0.5)\) in Figure 3.15b. In the example, the frequency of the sinusoids are set to \(\omega = 50\) and the other parameters are \(h = 1, \alpha = 0.25, c_x = c_y = 1\). The radius of the obstacle is \(r = 0.1\) with a radius of the security region \(R = 0.3\).

![Figure 3.15: Obstacle Avoidance](image)

Although the value of \(\omega\) is sufficiently large such that the Lie bracket system should give a good approximation as it was showed in the section before, one cannot conclude any quantitative behavior for the original system in the presence of obstacles. The trajectory of the original system is not bounded by the trajectory of the Lie bracket system anymore.

This example shows very good, that the Lie bracket system can not serve as a qualitative approximation for the original system if the obstacle avoidance feedback is a non-smooth function.
3.3 Game Theoretic Multi-Agent Optimization

An interesting property of the Lie bracket interpretation of the extremum seeking allows to extend it to multi-agent systems, where each agent has its own utility function that it tries to maximize. As every utility function usually depends on the actions of the other agents, this problem can be formulated in a game theoretic framework.

Especially potential games suit very nicely, because of their properties introduced in section 2.1.4. By using extremum seeking as local optimization algorithm, it is possible to design a scheme in which each agent optimizes its own utility function resulting in a group of autonomous agents that solve a common goal without knowing the overall utility function.

For this purpose, consider the extremum seeking in Figure 3.16 with $v_i(s)$ as the individual utility function.

![Figure 3.16: Extremum Seeking with Individual Utility Function](image)

Two different approaches are outlined. Both of them incorporate the extremum seeking as optimization algorithm and can be used for the same class of potential games.

The first algorithm uses the finite-improvement path property of potential games. Every agent applies the extremum seeking only for a certain time, whereas this switching scheme results in an improvement of the potential function.

The second algorithm handles the case where all agents move at the same time but with different frequencies. This property can be used to decouple the dynamics of the agents and apply a similar procedure as for the single agent case.

All agents are modeled in the same way but can differ in the frequencies $\omega_i$ of the driving sinusoids and the parameters $\alpha_i$ an $c_i$. This is important especially for the second approach where all agents must have different frequencies $\omega_i$.

Both approaches have the same requirements on the underlying game. It is defined in the following way and all references are made to this definition.
Consider a potential game $\Delta = \langle V, A, U \rangle$ where $V = (1, \ldots, N)$, $A = \mathbb{R}^2 \times \ldots \times \mathbb{R}^2$, and $U = \{v_i(s)\}$ with continuously differentiable and strictly concave potential function $Q(s) : A \to \mathbb{R}$ where $Q(s) \to -\infty$ when $||s|| \to \infty$ and smooth individual functions $v_i(s) : A \to \mathbb{R}$.

This definition implies that the game $\Delta$ admits a unique Nash equilibrium denoted by $s^*$ that coincides with the maximum of $Q(s)$.

### 3.3.1 Multi-Agent Extremum Seeking with Switched Improvement

Before stating the algorithm the following assumptions have to be made:

A.1 Every agent is equipped with the extremum seeking feedback as in Fig. 3.16.

A.2 There is an exact order between the agents, and time instances $t^i_k$ assigned to each agent $i$.

A.3 The time instances are chosen in such a way that $t^i_{k+1} - t^i_k$ is an integer multiple of the individual frequency of the sinusoids of agent $i$.

A.4 $-\frac{\pi}{2} < \phi_i < \frac{\pi}{2}$

A.5 $h > 0$, $\alpha_i > 0$, $c_i > 0$

The algorithm is then

$k := 1$

for all $i$ in $V$

apply extremum seeking for time $t^i_{k+1} - t^i_k$

end for

$k \leftarrow k + 1$

**Theorem 11.** Under the assumptions A.1-A.5 and for large enough $t^i_{k+1} - t^i_k$ and $\omega_i$ the agents converge to a region arbitrary close to the Nash equilibrium of the potential game $\Delta$ that is the maximum of the potential function $Q(s)$.

**Proof.** As the potential function is strictly concave and has a single maximum at $s^*$, the existence of a Nash equilibrium is assured by the fact that the equilibria of $Q(s)$ and the game $\Delta$ coincide.

By Lemma 8 and as $t^i_{k+1} - t^i_k$ is an integer multiple of the individual frequency of each agent, the system equations can be approximated by the Lie bracket system of the form

$$
\begin{pmatrix}
\dot{s}_{ix} \\
\dot{s}_{iy} \\
\dot{e}_i
\end{pmatrix} =
\begin{pmatrix}
\frac{1}{2} (c\alpha \nabla s_{ix} v_i \cos(\phi) - c\alpha \nabla s_{iy} v_i \sin(\phi) - c^2 \nabla s_{iy} v_i \cos(\phi)(v_i - \bar{e}_i h)) \\
\frac{1}{2} (c\alpha \nabla s_{iy} v_i \cos(\phi) + c\alpha \nabla s_{ix} v_i \sin(\phi) + c^2 \nabla s_{ix} v_i \cos(\phi)(v_i - \bar{e}_i h)) \\
-\bar{e}_i h + v_i
\end{pmatrix}
$$

whose trajectories are close to the trajectories of the original system in a region defined by

$$
||(s_i, e)^T - (\bar{s}_i, \bar{e})^T||_{C[0,T]} \leq \Delta_c.
$$

(3.67)

Taking the Lyapunov function candidate $W = -Q_i(\bar{s}) + Q(s^*)$ the position $s_{-i}$ of the other agents are fixed as only one agent moves at a time. $s^*$ denotes the maximum of the potential function $Q(\bar{s})$, such that $W(s^*) = 0$, and $W > 0$ around $s^*$.
Then by standard Lyapunov stability theory, the point $s^*$ is globally uniformly asymptotically stable for the Lie bracket system:

$$
\dot{W} = \frac{\partial W}{\partial \bar{s}_i} \dot{\bar{s}}_i \\
= -\alpha c_x \nabla_{s_{ix}} Q \nabla_{s_{ix}} v_i \cos(\phi_i) + c_x c_y (v_i - e_i h) \nabla_{s_{ix}} Q \nabla_{s_{iy}} v_i \cos(\phi_i) \\
- \alpha c_y \nabla_{s_{iy}} Q \nabla_{s_{iy}} v_i \cos(\phi_i) - c_x c_y (v_i - e_i h) \nabla_{s_{iy}} Q \nabla_{s_{ix}} v_i \cos(\phi_i)
$$

(3.68)

Therefore the Lie bracket system will approach the equilibrium $s^*_i$ and the individual utility function will be increasing in their value at each iteration $v(s_{i}^{k+1}, s_{-i}) \geq v(s_{i}^{k}, s_{-i}) + \epsilon$. As the initial condition of the Lie bracket system and the original one are the same, equation (3.67) assures that the original system will approach $s^*_i$ arbitrary close.

The fact that $\Delta$ is a potential game with $Q(s)$ as the potential function, and that there is only one agent that moves at a time while it improves its utility function, an $\epsilon$-improvement path is constructed. As the potential function admits a maximum, this improvement path is by Lemma 5 an approximate finite improvement path that ends by definition in an $\epsilon$-equilibrium point of $Q(s)$ and because of the strict concavity of $Q(s)$ this equilibrium is the maximum $s^*$.

As the set of equilibria of the potential function $Q(s)$ coincides with the equilibrium set of $\Delta$ by Lemma 2, the maximum of $Q(s)$ is the Nash equilibrium of the Game $\Delta$.

The state $e$ filter is input-to-state stable with respect to the position $s_i$ and therefore irrelevant for the final position of the agents.

**Remark** The idea of the algorithm is the same as the gradient ascent algorithm with unit directions. The definition of potential games, namely the fact that $\frac{\partial Q(s)}{\partial s_i} = \frac{\partial v_i(s)}{\partial s_i}$, is assuring that each agent affects only its own position $s_i$ and as the function $Q(s)$ is bounded, a gradient ascent of its own utility will end in a gradient ascent in the utility function.

This is the reason why it is necessary to use the algorithm only with potential games. With any other game, it can happen, that the agents end up in a limit cycle as the utility functions are depending on the actions of the other agents.

**Remark** Furthermore, Theorem 8 that shows practical stability of the extremum seeking is not necessary for the proof. Nevertheless, the idea of the proof of Theorem 2 and the proof of Theorem 11 are related. They mainly state, that if the trajectory of the original extremum seeking system is close to the trajectory of the Lie bracket system for some time $T$, then it must be close to the trajectory for all time. This is also the main conclusion of Theorem 11 where the evolution of the trajectory of the original system is concluded from the evolution of the trajectory of the Lie bracket system.

**Remark** One could also imagine to change the amplitudes of the sinusoids of the extremum seeking in each iteration, such that the agents reach the Nash equilibrium exactly, when the amplitudes reach zero. The decrement must not go to zero before the Nash equilibrium is reached, otherwise the agents get stuck on their way without the possibility to improve their utility function (cf. [20], [21], [22]).
Remark In the Theorem, there is no estimate how close the agents reach the optimum. Furthermore, it is not possible to relate this type of convergence to practical stability as it was done in the single-agent case, or as it will follow in the upcoming theorems. The main drawback of Theorem 6 is that one cannot assure that there is any guaranteed improvement after one period of the individual frequencies. This is also the reason, why it is necessary to apply the extremum seeking for a sufficiently long time. As soon as the agents reach a region close to $s^*$ where the slope of the individual utility functions is too small to improve sufficiently after the chosen time $t_{i,k+1}^i - t_i^k$, the improvement in the utility functions could be smaller than the term $||(s_i, e)^T - (\bar{s}_i, \bar{e})^T||_{[0,T]} \leq \Delta e$.

Nevertheless, it is obvious that the agents must improve after at least one period of the sinusoids, even if it is not explicitly stated because the Lie bracket motion is sufficiently dominant after one period as shown in Section 2.2. Therefore, one can expect an improvement even if $t_{i,k+1}^i - t_i^k$ is exactly one period.

The authors in [25] gave a proof of this notion for a special class of systems that can be approximated by using the averaging theorem. Their results implies that the original system improves after one period of a periodic input.

In Chapter 5 simulation results will show that this can also be expected here, even for smaller values of $\omega$.

3.3.2 Quadratic Potential Game as Example for Agents with Switched Improvement

To give a better insight in the motion of the agents and the connection to potential games, a simple example is used. As shown in Section 2.1 quadratic games can be potential games under certain requirements.

The quadratic game in this example consists of two agents, $i = 1, 2$ with positions $s_1, s_2 \in \mathbb{R}$ and utility functions

$$u_1(s_1, s_2) = -s_1^2 - s_2^2 - s_1$$
$$u_2(s_1, s_2) = -s_1^2 - s_2^2 + s_2. \quad (3.69)$$
$$\quad (3.70)$$

This game is a potential game with potential function

$$P(s_1, s_2) = -s_1^2 - s_2^2 - s_1 + s_2$$
$$= - \vec{R} \cdot \begin{pmatrix} s_1 \\ s_2 \end{pmatrix} - (s_1 \ s_2) \begin{pmatrix} 1 & -1 \end{pmatrix}. \quad (3.71)$$

As this is a quadratic game, there exists only one Nash equilibrium at $-R^{-1}r = (-1, 1)^T$.

Obviously, this is a very simple case, but only such a simple example can be visualized in three dimensions.

The underlying game is going to be solved with the proposed algorithm in order to give more insight how the algorithm works. It is important to mention, that the extremum seeking that was for used for this example, improves only one variable (cf. Appendix A) instead of two as it was used in the sections before. The algorithm as well as the Theorem can directly be translated to the one-dimensional case.
The resulting trajectories are shown in Figure 3.17. Each agent $i$ can only improve its own position $s_i$, but this improvement results also in an improvement of the potential function. In Figure 3.17 the blue trajectory belongs to Agent 1 and the red to Agent 2. As in the proposed algorithm, only one Agent can move at a time and as they are improving their own utility, they are climbing up the potential function until they reach the Nash Equilibrium.

3.3.3 Multi-Agent Extremum Seeking for Agents with Different Frequencies

The second method does not need to be formulated as an algorithm, as convergence can be assured by using standard stability theory. It uses the same scheme as the algorithm before but with special adaptations for each agent, namely that each agent has its own frequency for the driving sinusoids.

Before stating the Theorem, make the following assumptions on the parameters of each agent.

B.1 $\omega_i = a_i \omega$ and $a_i \neq a_j$, $\forall i \neq j$ and $a_i \in \mathbb{Q}^+$

B.2 $-\frac{\pi}{2} < \phi_i < \frac{\pi}{2}$

B.3 $h > 0$, $\alpha_i > 0$, $c_i > 0$.

To clarify why this is needed, the proof of the Theorem is very enlightening.

**Theorem 12.** Given $N$ agents equipped with the extremum seeking feedback in Figure 3.16. Under the assumptions B.1-B.3 and for sufficiently large $\omega$ the maximum $s^*$ is practically globally uniformly asymptotically stable.
3.3. GAME THEORETIC MULTI-AGENT OPTIMIZATION

Proof. The system equations for each agent $i$ have by definition of the Theorem the following structure

$$\begin{align*}
\dot{s}_{ix} &= c_i(v_i - e_i h)\sqrt{\omega_i} \sin(a_i \omega t - \phi_i) + \alpha_i \sqrt{\omega_i} \cos(a_i \omega t) \\
\dot{s}_{iy} &= -c_i(v_i - e_i h)\sqrt{\omega_i} \cos(a_i \omega t - \phi_i) + \alpha_i \sqrt{\omega_i} \sin(a_i \omega t) \\
\dot{e} &= -e_i h + v_i
\end{align*}$$

(3.72)

By using the identities $\sin(x - y) = \sin(x) \cos(y) - \cos(x) \sin(y)$ and $\cos(x - y) = \cos(x) \cos(y) + \sin(x) \sin(y)$ this yields to

$$\begin{align*}
\dot{s}_{ix} &= c_i(v_i - e_i h)\sqrt{\omega_i} \sin(a_i \omega t) \cos(\phi_i) - c_i(v_i - e_i h)\sqrt{\omega_i} \cos(a_i \omega t) \sin(\phi_i) + \alpha_i \sqrt{\omega_i} \cos(a_i \omega t) \\
\dot{s}_{iy} &= -c_i(v_i - e_i h)\sqrt{\omega_i} \cos(a_i \omega t) \cos(\phi_i) - c_i(v_i - e_i h)\sqrt{\omega_i} \sin(a_i \omega t) \sin(\phi_i) + \alpha_i \sqrt{\omega_i} \sin(a_i \omega t) \\
\dot{e} &= -e_i h + v_i.
\end{align*}$$

(3.73)

The proof idea follows the same steps as the proofs of Lemma 8 and Theorem 8.

The position vectors and the states of the filter of each agent are stacked in a vector $(s_{1x}, s_{1y}, e_1, \ldots, s_{Nx}, s_{Ny}, e_N)^T$.

$$\begin{pmatrix}
\vdots \\
\dot{s}_{ix} \\
\dot{s}_{iy} \\
\dot{e}_i \\
\vdots
\end{pmatrix} = \sum_{i=1}^{N} \begin{pmatrix}
\alpha_i - c_i(v_i - e_i h) \sin(\phi_i) \\
\sqrt{\omega_i} \sin(a_i \omega t) \\
\sqrt{\omega_i} \cos(a_i \omega t) \\
0
\end{pmatrix}_{u_{ia}} + \begin{pmatrix}
\alpha_i - c_i(v_i - e_i h) \sin(\phi_i) \\
0 \\
0 \\
-e_i h + v_i(s)
\end{pmatrix}_{u_{ib}} + \begin{pmatrix}
\vdots \\
0 \\
0 \\
\vdots
\end{pmatrix}_{u_{ib}} \frac{1}{\omega_i}.
$$

(3.74)

As by assumption all $a_i$’s are rational numbers, they can be written as $a_i = p_i/q_i$, and without loss of generality $p_i, q_i \in \mathbb{N}$.

Choose $q = \prod_j q_j$, $\tilde{\omega} := \omega T/2\pi$ and with $\epsilon := q/\tilde{\omega}$ and $u_0 = 1$, all the $u_{ia}$’s and $u_{ib}$’s are periodic in $\theta = t/\epsilon$ with zero average. Define furthermore $\tilde{\alpha}_i := \alpha_i \sqrt{\prod_{j \neq i} q_j}$ and $\tilde{e}_i := c_i \sqrt{\prod_{j \neq i} q_j}$ such that $u_{ia} = \sqrt{\omega/q} \sin(a_i \omega t)$ and $u_{ib} = \sqrt{\omega/q} \cos(a_i \omega t)$. The vector fields $\tilde{b}_a$ and $\tilde{b}_b$ denote the vector fields using $\tilde{\alpha}_i$ and $\tilde{e}_i$. In terms of $\epsilon$ this yields to $\tilde{u}_{ia} = 1/\sqrt{\epsilon} \sqrt{2\pi} \sin(p_i \prod_{j \neq i} q_j \epsilon/2\pi T)$ and $\tilde{u}_{ib} = 1/\sqrt{\epsilon} \sqrt{2\pi} \cos(p_i \prod_{j \neq i} q_j \epsilon/2\pi T)$ whereas by assumption $p_i \prod_{j \neq i} q_j \in \mathbb{N}$.

The drift influences only the state of the filter $e_i$ to which a virtual input $u_{i0} = 1$ is associated that is constant. All $\tilde{u}_{ia}$’s and $\tilde{u}_{ib}$’s are periodic in $T$ and the average is zero.

$$\int_0^T \tilde{u}_{ia}(p_i \prod_{j \neq i} q_j \theta 2\pi T) d\theta = \int_0^T \tilde{u}_{ib}(p_i \prod_{j \neq i} q_j \theta 2\pi T) d\theta = 0.$$ 

(3.75)
Therefore all assumptions of Theorem 6 are fulfilled, and the Theorem can be applied. The approximate system is

\[
\begin{pmatrix}
\dot{s} \\
\dot{\bar{e}}
\end{pmatrix} = -\frac{1}{T} \sum_{i < j} [\tilde{b}_i, \tilde{b}_j] \nu_{i,j} + \sum_i \begin{pmatrix}
\vdots \\
0 \\
0 \\
-\bar{e}_i h + v_i(s) \\
\vdots
\end{pmatrix}, \quad \bar{s}(0) = s(0) \tag{3.76}
\]

with \( \nu_{i,j} = \int_0^T \int_0^\theta u_i(\tau \frac{2\pi}{T}) u_j(\theta \frac{2\pi}{T}) d\tau d\theta \). The summation is done over all \( i \in \{1a, 1b, 2a, 2b, \ldots\} \).

An important fact now is, that \( \nu_{i,j} = 0 \) for \( u_i \)'s with different periods of sinusoids because for all \( n, m \in \mathbb{N}, n \neq m \)

\[
\int_0^{2\pi} \int_0^\theta \sin(m\tau) \cos(n\theta) d\tau d\theta = 0 \\
\int_0^{2\pi} \int_0^\theta \sin(n\tau) \cos(m\theta) d\tau d\theta = 0 \\
\int_0^{2\pi} \int_0^\theta \sin(n\tau) \sin(n\theta) d\tau d\theta = \frac{\pi}{n}.
\]

and therefore

<table>
<thead>
<tr>
<th>( \nu_{i,j} )</th>
<th>( u_{1a} )</th>
<th>( u_{1b} )</th>
<th>( \ldots )</th>
<th>( u_{Na} )</th>
<th>( u_{Nb} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( u_{1a} )</td>
<td>0</td>
<td>(-\frac{T}{2})</td>
<td>( \ldots )</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( u_{1b} )</td>
<td>*</td>
<td>0</td>
<td>( \ldots )</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( \vdots )</td>
<td>*</td>
<td>*</td>
<td>( \ldots )</td>
<td>( \vdots )</td>
<td>( \vdots )</td>
</tr>
<tr>
<td>( u_{Na} )</td>
<td>*</td>
<td>*</td>
<td>( \ldots )</td>
<td>0</td>
<td>(-\frac{T}{2P_N} \sum_{j \neq N} q_j )</td>
</tr>
<tr>
<td>( u_{Nb} )</td>
<td>*</td>
<td>*</td>
<td>( \ldots )</td>
<td>*</td>
<td>0</td>
</tr>
</tbody>
</table>

\tag{3.78}

the approximated system yields with equation (3.78), resubstitution of \( \tilde{\alpha}_i, \tilde{c}_i \) and Theorem 6 to

\[
\begin{pmatrix}
\dot{s} \\
\dot{\bar{e}}
\end{pmatrix} = -\frac{1}{2} \sum_i [b_{ia}, b_{ib}] + \sum_i \begin{pmatrix}
\vdots \\
0 \\
0 \\
-\bar{e}_i h + v_i(s) \\
\vdots
\end{pmatrix}, \quad (\bar{s}(0), \bar{e}(0))^\top = (s(0), e(0))^\top \tag{3.79}
\]

One obtains for the equations of agent \( i \)

\[
\dot{s}_ix = \frac{1}{2} (c_i \alpha_i \nabla_{s_i \alpha} v_i(s) \cos(\phi_i) - c_i \alpha_i \nabla_{s_i \phi} v_i(s) \sin(\phi_i) - c_i^2 \nabla_{s_i v} v_i(s) \cos(\phi_i) (v_i(s) - e_i h))) \\
\dot{s}_iy = \frac{1}{2} (c_i \alpha_i \nabla_{s_i \alpha} v_i(s) \cos(\phi_i) + c_i \alpha_i \nabla_{s_i \phi} v_i(s) \sin(\phi_i) + c_i^2 \nabla_{s_i z} v_i(s) \cos(\phi_i) (v_i(s) - e_i h)) \\
\dot{\bar{e}}_i = -\bar{e}_i h + v_i(s).
\]

\tag{3.80}

The position vectors of each agent can now be treated separately from the filter states. Consider for this purpose the reduced system \( s = (s_{1x}, s_{1y}, \ldots, s_{Nx}, s_{Ny})^\top \).
Using the potential function $W = -Q(\bar{s}) + Q(s^*)$ as a Lyapunov function and performing the change of variables $\bar{s} := \bar{s} - s^*$ that is always possible under the assumptions, one obtains for the derivative along the trajectories of the Lie bracket system

$$\dot{W} = -\nabla_{s_{1x}} Q(s) \dot{s}_{1x} - \nabla_{s_{1y}} Q(s) \dot{s}_{1y} - \ldots - \nabla_{s_{Nz}} Q(s) \dot{s}_{Nz} - \nabla_{s_{Ny}} Q(s) \dot{s}_{Ny}. \quad (3.81)$$

As this is a potential game, where the identity $\nabla_{s_i} Q(s) = \nabla_{s_i} v_i$ can be used, this yields to

$$\dot{W} = -\frac{c_1\alpha_1}{2}(\nabla_{s_{1x}} Q(s))^2 \cos(\phi_1) - \frac{c_1\alpha_1}{2}(\nabla_{s_{1y}} Q(s))^2 \cos(\phi_1) - \ldots - \frac{c_N\alpha_N}{2}(\nabla_{s_{Nz}} Q(s))^2 \cos(\phi_N) - \frac{c_N\alpha_N}{2}(\nabla_{s_{Ny}} Q(s))^2 \cos(\phi_N) < 0 \quad \forall \bar{s} \neq 0. \quad (3.82)$$

Therefore the maximum $s^*$ is globally uniformly asymptotically stable. Consider now the state space equations of the filters

$$\dot{\bar{e}} = \begin{pmatrix} \bar{e}_1 \\ \vdots \\ \bar{e}_N \end{pmatrix} = \begin{pmatrix} -\bar{e}_1 h + v_1(\bar{s}) \\ \vdots \\ -\bar{e}_N h + v_N(\bar{s}) \end{pmatrix} \quad (3.83)$$

and perform the change of variables $\tilde{e}_i = \bar{e}_i(\bar{s}) - v_i(s^*)$. This yields to

$$\dot{\tilde{e}} = \begin{pmatrix} \tilde{e}_1 \\ \vdots \\ \tilde{e}_N \end{pmatrix} = \begin{pmatrix} -\tilde{e}_1 h + v_1(\bar{s}) - v_1(s^*) \\ v_1(\bar{s}) \\ \vdots \\ -\tilde{e}_N h + v_N(\bar{s}) - v_N(s^*) \end{pmatrix}. \quad (3.84)$$

Obviously, all $\tilde{e}_i$’s are decoupled and with bounded and smooth $v_i(\bar{s})$’s all $\tilde{e}_i$’s are input-to-state stable with respect to $\tilde{v}_i(\bar{s})$ respectively. Concluding from the calculation before, with $s \to s^*$ for $t \to \infty$, together with Lemma 7 this yields to $\tilde{e}_i \to 0$. One can conclude with Theorem 2 that the agents converge arbitrary close to the maximum $s^*$ in the sense that $s^*$ is practically globally uniformly asymptotically stable.

**Example** [Calculation for two Agents] To show the calculations in detail, an example with two agents is going to be performed. For the sake of simplicity, the states of the filter are omitted.
Consider 2 agents. The stacked position vector of the system yields to

\[
\begin{pmatrix}
\dot{s}_{1x} \\
\dot{s}_{1y} \\
\dot{s}_{2x} \\
\dot{s}_{2y}
\end{pmatrix} =
\begin{pmatrix}
u_{1a} \\
\alpha \\
0 \\
0
\end{pmatrix}
\begin{pmatrix}
\sin \omega t \\
\cos \omega t
\end{pmatrix}
+ \begin{pmatrix}
u_{1b} \\
0 \\
\alpha \\
\alpha
\end{pmatrix}
\begin{pmatrix}
\dot{s}_{2x} \\
\dot{s}_{2y}
\end{pmatrix} = \begin{pmatrix}
u_{1a} \\
\nu_{1b} \\
\nu_{2a} \\
\nu_{2b}
\end{pmatrix} + \begin{pmatrix}
u_{1b} \\
\alpha \\
-\nu_{2b} \\
\alpha
\end{pmatrix}
\begin{pmatrix}
\dot{s}_{2x} \\
\dot{s}_{2y}
\end{pmatrix}.
\] (3.85)

Calculate first the \( v_{i,j} \)

\[
\begin{align*}

\begin{array}{c|cccc}
   & u_{1a} & u_{1b} & u_{2a} & u_{2b} \\
\hline
u_{1a} & 0 & -\pi & 0 & 0 \\
u_{1b} & * & 0 & 0 & 0 \\
u_{2a} & * & * & 0 & -\frac{\pi}{2} \cdot \sqrt{2} \\
u_{2b} & * & * & * & 0
\end{array}
\end{align*}
\]

(3.86)

This matrix is skew-symmetric and contains a lot of zeros. This yields to the Lie bracket system of the form

\[
\dot{s} = \frac{1}{2\pi} \nu_{1a,1b}[b_{1a}, b_{1b}] + \frac{1}{2\pi} \nu_{2a,2b}[b_{2a}, b_{2b}]
\] (3.87)

Therefore only the Lie brackets \([b_{1a}, b_{1b}]\) and \([b_{2a}, b_{2b}]\) are necessary, as all the other Lie brackets do not influence the system. With

\[
[b_{1a}, b_{1b}] = \frac{\partial b_{1a}}{\partial s} b_{1a} - \frac{\partial b_{1b}}{\partial s} b_{1b}
\]

\[
= \begin{pmatrix}
\nabla_{s_{1x}} u_{1a} \\
\nabla_{s_{1y}} u_{1a} \\
0 \\
0
\end{pmatrix}
\begin{pmatrix}
\alpha \\
0 \\
0 \\
0
\end{pmatrix}
= \begin{pmatrix}
\nabla_{s_{1x}} u_{1a} \\
\nabla_{s_{1y}} u_{1a} \\
\nabla_{s_{2x}} u_{1a} \\
\nabla_{s_{2y}} u_{1a}
\end{pmatrix}
\begin{pmatrix}
\alpha \\
-\nu_{1a} \\
\nu_{2a} \\
\nu_{2b}
\end{pmatrix}
\]

(3.88)

and

\[
[b_{2a}, b_{2b}] = \begin{pmatrix}
0 \\
0 \\
-\alpha \nabla_{s_{2x}} u_{2a} + u_{2a} \nabla_{s_{2y}} u_{2a} \\
-\alpha \nabla_{s_{2y}} u_{2a} + u_{2a} \nabla_{s_{2x}} u_{2a}
\end{pmatrix}
\]

(3.89)
the Lie bracket system finally is

\[
\begin{pmatrix}
\dot{s}_{1x} \\
\dot{s}_{1y} \\
\dot{s}_{2x} \\
\dot{s}_{2y}
\end{pmatrix}
= \frac{1}{2}
\begin{pmatrix}
\alpha \nabla_{s_1x} u_1(s) - u_1(s) \nabla_{s_1y} u_1(s) \\
\alpha \nabla_{s_1x} u_1(s) + u_1(s) \nabla_{s_1y} u_1(s) \\
\alpha \nabla_{s_2x} u_2(s) - u_2(s) \nabla_{s_2y} u_2(s) \\
\alpha \nabla_{s_2x} u_2(s) + u_2(s) \nabla_{s_2y} u_2(s)
\end{pmatrix}
\]  

(3.90)

with \( V = -Q(s) + Q(s^*) \) as Lyapunov function this yields for the derivative of \( V \) to

\[
\dot{V} = -\nabla_{s_1x} Q(s) \dot{s}_{1x} - \nabla_{s_1y} Q(s) \dot{s}_{1y} - \nabla_{s_2x} Q(s) \dot{s}_{2x} - \nabla_{s_2y} Q(s) \dot{s}_{2y}
\]

\[
= -\alpha (\nabla_{s_1x} Q(s))^2 - \alpha (\nabla_{s_1y} Q(s))^2 - \alpha (\nabla_{s_2x} Q(s))^2 - \alpha (\nabla_{s_2y} Q(s))^2
\]

\[
+ \nabla_{s_1x} Q(s) \nabla_{s_1y} Q(s) u_1(s) - \nabla_{s_1x} Q(s) \nabla_{s_1y} Q(s) u_1(s)
\]

\[
+ \nabla_{s_2x} Q(s) \nabla_{s_2y} Q(s) u_2(s) - \nabla_{s_2x} Q(s) \nabla_{s_2y} Q(s) u_2(s)
\]

\[
= -\alpha (\nabla_{s_1x} Q(s))^2 - \alpha (\nabla_{s_1y} Q(s))^2 - \alpha (\nabla_{s_2x} Q(s))^2 - \alpha (\nabla_{s_2y} Q(s))^2
\]

\[
< 0 \quad \forall \bar{s} \neq s^*
\]


doe the identities \( \nabla_{s_1x} Q(s) = \nabla_{s_1x} u_1(s) \), \( \nabla_{s_1y} Q(s) = \nabla_{s_1y} u_1(s) \), \( \nabla_{s_2x} Q(s) = \nabla_{s_2x} u_2(s) \) and

\( \nabla_{s_2y} Q(s) = \nabla_{s_2y} u_2(s) \) were used.

Quadratic Potential Game as Example for Agents with Different Frequencies The second Example is meant to show the difference of the solution trajectories of the agents of the quadratic game that was also used in the example for the first algorithm. As the parameter \( \omega \) is chosen to be equal to 10, \( \omega_1 = 10 \) and \( \omega_2 = 20 \). The parameter \( c \) is for both agents equal to 0.1, whereas \( \alpha_1 = 0.1 \).

\[\text{Figure 3.18: Extremum Seeking in a Quadratic Game with Different Frequencies}\]

The resulting trajectory of the agents is shown in Figure 3.18.

Comparing this to the previous example, there is now a common trajectory of the whole system consisting of Agent 1 and Agent 2. The trajectory is aligned with the gradient of...
the potential function, whereas the trajectories of the Algorithm before, consisted of the unit directions.

### 3.3.4 Multi-Agent Extremum Seeking for the Unicycle Model and Different Frequencies

The previous method can also be extended to the Unicycle model as the following Theorems shows. Consider the more sophisticated unicycle model for each agent given by the equations

\[
\dot{s}_x = u \cos(\theta) \\
\dot{s}_y = u \cos(\theta) \\
\dot{\theta}_i = \Omega_i \\
\dot{e}_i = -e_i h + v_i.
\]  

(3.92)

The extremum seeking feedback controls only the velocity of the vehicle, whereas the the angular velocity is constant. Choose the extremum seeking feedback \( u = (c_i(v_i - e_i h)\sqrt{\omega_i \sin(\omega_i t - \phi_i) + \alpha_i \sqrt{\omega_i \cos(\omega_i t)})} \) and \( v = \Omega_i \) and make the following assumptions on the parameters of the feedback

C.1 \( \omega_i = a_i \omega \) and \( a_i \neq a_j \), \( \forall i \neq j \) and \( a_i \in \mathbb{Q}^+ \)

C.2 \( -\frac{\pi}{2} < \phi_i < \frac{\pi}{2} \)

C.3 \( h > 0, \alpha_i > 0, c_i > 0 \)

C.4 \( \Omega_i \neq 0, i \in V \).

The Theorem is similar to the single integrator case.

**Theorem 13.** Given \( N \) agents equipped with the extremum seeking feedback in Figure 3.19. Under the assumptions C.1-C.4 and for sufficiently large \( \omega \) the maximum \( s^* \) is practically globally uniformly asymptotically stable.

**Proof.** First, the given inputs are plugged into the system equations of the unicycle model. This yields to the equations

\[
\dot{s}_x = (c_i(v_i - e_i h)\sqrt{\omega_i \sin(\omega_i t - \phi_i) + \alpha_i \sqrt{\omega_i \cos(\omega_i t)})} \cos(\theta) \\
\dot{s}_y = (c_i(v_i - e_i h)\sqrt{\omega_i \sin(\omega_i t - \phi_i) + \alpha_i \sqrt{\omega_i \cos(\omega_i t)})} \sin(\theta) \\
\dot{\theta}_i = \Omega_i \\
\dot{e}_i = -e_i h + v_i.
\]  

(3.93)
By using the identity \( \sin(x - y) = \sin(x)\cos(y) - \cos(x)\sin(y) \) and \( \theta = \Omega \) one obtains
\[
\begin{align*}
\dot{s}_{ix} &= (c_i(v_i - e_i h) \sqrt{\bar{\omega}} \sin(a_i \omega t) \cos(\phi_i) - c_i(v_i - e_i h) \sqrt{\bar{\omega}} \cos(a_i \omega t) \sin(\phi_i)) + \alpha_i \sqrt{\bar{\omega}} \cos(a_i \omega t) \cos(\theta), \\
\dot{s}_{iy} &= (c_i(v_i - e_i h) \sqrt{\bar{\omega}} \sin(a_i \omega t) \cos(\phi_i) - c_i(v_i - e_i h) \sqrt{\bar{\omega}} \cos(a_i \omega t) \sin(\phi_i)) + \alpha_i \sqrt{\bar{\omega}} \cos(a_i \omega t) \sin(\theta), \\
\dot{\theta}_i &= \Omega_i, \\
\dot{e}_i &= -e_i h + v_i.
\end{align*}
\]

The position vectors and the states of the filter of each agent are stacked in a vector \( (s_{1x}, s_{1y}, \theta_1, e_1, \ldots, s_{Ny}, s_{Ny}, \theta_N, e_N)^\top \)
\[
\begin{pmatrix}
\vdots \\
\dot{s}_{ix} \\
\dot{s}_{iy} \\
\dot{\theta}_i \\
\dot{e}_i \\
\vdots
\end{pmatrix} = \sum_{i=1}^{N} \begin{pmatrix}
\vdots \\
c_i(v_i - e_i h) \cos(\phi_i) \cos(\theta) \\
c_i(v_i - e_i h) \cos(\phi_i) \sin(\theta) \\
0 \\
0 \\
\vdots
\end{pmatrix} \frac{\sqrt{\bar{\omega}} \sin(a_i \omega t)}{u_{ia}} + \begin{pmatrix}
\vdots \\
-(c_i(v_i - e_i h) \sin(\phi_i) - \alpha_i) \cos(\theta) \\
-(c_i(v_i - e_i h) \sin(\phi_i) - \alpha_i) \sin(\theta) \\
0 \\
0 \\
\vdots
\end{pmatrix} \frac{\sqrt{\bar{\omega}} \cos(a_i \omega t)}{u_{ib}} + \begin{pmatrix}
0 \\
0 \\
\Omega_i \\
-e_i h + v_i
\end{pmatrix} \frac{1}{u_{ic}}.
\]

As by assumption all \( a_i \)'s are rational numbers, they can be written as \( a_i = p_i/q_i \), and without loss of generality \( p_i, q_i \in \mathbb{N} \).

Choose \( q = \prod_j q_j \), \( \tilde{\omega} := \omega T \) and with \( \epsilon := q/\tilde{\omega} \) and \( u_0 = 1 \), all the \( u_{ia} \)'s and \( u_{ib} \)'s are periodic in \( \theta = t/\epsilon \) with zero average. Define furthermore \( \tilde{\alpha}_i := \alpha_i \sqrt{\bar{\omega}}/q \sin(a_i \omega t) \) and \( \tilde{\epsilon}_i := c_i \sqrt{\bar{\omega}}/q \cos(a_i \omega t) \) such that \( u_{ia} = \sqrt{\bar{\omega}}/q \sin(a_i \omega t) \) and \( u_{ib} = \sqrt{\bar{\omega}}/q \cos(a_i \omega t) \). The vector fields \( \tilde{b}_{ia} \) and \( \tilde{b}_{ib} \) denote the vector fields using \( \tilde{\alpha}_i \) and \( \tilde{\epsilon}_i \). In terms of \( \epsilon \) this yields to \( \tilde{u}_{ia} = 1/\sqrt{\tilde{\epsilon}} \sqrt{2\pi} \sin(p_i \prod_{j \neq i} q_j t/\epsilon 2\pi) \) and \( \tilde{u}_{ib} = 1/\sqrt{\tilde{\epsilon}} \sqrt{2\pi} \cos(p_i \prod_{j \neq i} q_j t/\epsilon 2\pi) \) whereas by assumption \( p_i \prod_{j \neq i} q_j \in \mathbb{N} \).

The drift influences only the state of the filter \( e_i \) to which a virtual input \( u_{ia} = 1 \) is constant and equal to 1, is associated. As furthermore all \( \Omega_i \)'s are constant, one can directly use the analytic solution of the differential equations \( \tilde{\theta}_i = \Omega_i \) that is \( \tilde{\theta}(t) = \Omega_i t \). Therefore, with the results of Lemma [3] and the same arguments as in the proof before, the approximative system
for agent \( i \) is obtained

\[
\dot{\bar{s}}_{ix} = \frac{1}{2} \left( c_i \alpha_i \nabla_{\bar{s}_{ix}} v_i \cos(\phi_i) \cos^2(\Omega_i t) + \alpha_i \nabla_{\bar{s}_{iy}} v_i \cos(\phi_i) \cos(\Omega_i t) \sin(\Omega_i t) \right) \\
\dot{\bar{s}}_{iy} = \frac{1}{2} \left( c_i \alpha_i \nabla_{\bar{s}_{iy}} v_i \cos(\phi_i) \sin^2(\Omega_i t) + \alpha_i \nabla_{\bar{s}_{ix}} v_i \cos(\phi_i) \cos(\Omega_i t) \sin(\Omega_i t) \right)
\]

(3.96)

whereas one can conclude that the trajectories of the original system are bounded by the trajectories in the sense that \(||(s, e)^T - (\bar{s}, \bar{e})^T||_{C[0,T]} < \Delta_x||\).

The position vectors of each agent can now be treated separately from the filter states. Consider the reduced system consisting only of the position vectors \( s = [s_{1x}, s_{1y}, \ldots, s_{N_x}, s_{N_y}] \).

Using the potential function \( W = -Q(\bar{s}) + Q(s) \) as a Lyapunov function and performing the change of variables \( \bar{s} := s - s^* \), one obtains for the derivative along the trajectories of the approximative system

\[
\dot{W} = -\nabla_{s_{1x}} Q(s) \dot{\bar{s}}_{1x} - \nabla_{s_{1y}} Q(s) \dot{\bar{s}}_{1y} - \cdots - \nabla_{s_{N_x}} Q(s) \dot{\bar{s}}_{N_x} - \nabla_{s_{N_y}} Q(s) \dot{\bar{s}}_{N_y}.
\]

(3.97)

As this is a potential game, the individual utility functions fulfill the identity \( \nabla_{s_i} Q(s) = \nabla_{s_i} v_i \). This yields to

\[
\dot{W} = -\frac{c_i \alpha_i}{2} \left( \nabla_{\bar{s}_{ix}} Q(\bar{s}) \cos(\Omega_i t) + \nabla_{\bar{s}_{iy}} Q(\bar{s}) \sin(\Omega_i t) \right)^2 \cos(\phi_i) - \cdots - \frac{c_i \alpha_i}{2} \left( \nabla_{\bar{s}_{Ny}} Q(\bar{s}) \cos(\Omega_N t) + \nabla_{\bar{s}_{Ny}} Q(\bar{s}) \sin(\Omega_N t) \right)^2 \cos(\phi_N)
\]

(3.98)

\[
< 0 \quad \forall \bar{s} \neq 0.
\]

This calculation shows that \( \dot{W} \) is only negative semi-definite, this is due to the fact that the system is time-varying, and there are singular points in the state-space, where \( \dot{s} = 0 \), but which are no steady-states for the system. The components of \( \dot{W} \) can only be equal to zero if \( \nabla_{\bar{s}_{ix}} Q(\bar{s}) \cos(\Omega_i t) + \nabla_{\bar{s}_{iy}} Q(\bar{s}) \sin(\Omega_i t) = 0 \). Injecting this into the differential equations of the position vectors one obtains that \( \dot{s}_{ix} = \dot{\bar{s}}_{ix} = 0 \) and therefore \( \bar{s}_{ix} = \text{const.} \) and \( \bar{s}_{ix} = \text{const.} \). One can deduce that if \( \bar{s}_{ix} = \text{const.} \) and \( \bar{s}_{iy} = \text{const.} \). This implies that \( \nabla_{\bar{s}_{ix}} Q(\bar{s}) \) and \( \nabla_{\bar{s}_{iy}} Q(\bar{s}) \) are also constant. But as there are no constant values such that \( \nabla_{\bar{s}_{ix}} Q(\bar{s}) \cos(\Omega_i t) + \nabla_{\bar{s}_{iy}} Q(\bar{s}) \sin(\Omega_i t) = 0 \), \( \forall \Omega_i t \in \mathbb{R} \) except \( \nabla_{\bar{s}_{ix}} Q(\bar{s}) = \nabla_{\bar{s}_{iy}} Q(\bar{s}) = 0 \) as one can see by injecting \( t = 0 \) that implies \( \nabla_{\bar{s}_{ix}} Q(s) = 0 \) and \( t = \frac{\pi}{\Omega_i} \) that leads to \( \nabla_{\bar{s}_{ix}} Q(\bar{s}) = 0 \).

As \( V(x(t)) \) is monotonically decreasing and bounded from below, it will go to zero for \( t \to \infty \). Therefore the system approaches the maximum with \( t \to \infty \) for all initial conditions. Therefore the maximum \( s^* \) is globally uniformly asymptotically stable.

Consider now the state space equations of the filters

\[
\dot{\bar{e}} = \begin{pmatrix}
\bar{e}_1 \\
\vdots \\
\bar{e}_N
\end{pmatrix} = \begin{pmatrix}
-\bar{e}_1 h + v_1 \\
\vdots \\
-\bar{e}_N h + v_N
\end{pmatrix}
\]

(3.99)
and perform the change of variables $\tilde{e}_i = e_i - v_i(s^*)$ and $\hat{e}_i = e_i - v_i(s^*)$. This yields to

$$
\dot{\tilde{e}} = \begin{pmatrix}
\tilde{e}_1 \\
\vdots \\
\tilde{e}_N
\end{pmatrix} = \begin{pmatrix}
-\tilde{e}_1 h + v_1 - v_1(s^*) \\
\dot{v}_1 \\
\vdots \\
-\tilde{e}_N h + v_N - v_N(s^*) \\
\dot{v}_N
\end{pmatrix}.
$$

(3.100)

Obviously, all $\tilde{e}_i$'s are decoupled and with bounded and smooth $v_i(s)$'s all $\tilde{e}_i$'s are input-to-state stable with respect to $\tilde{v}_i(s)$ respectively. Concluding from the calculation before, with $s \to s^*$ for $t \to \infty$, together with this yields to $\tilde{e}_i \to 0$.

By the results of Theorem 2 one can conclude that the agents converge arbitrary close to $s^*$ in the sense that $s^*$ is practically globally uniformly asymptotically stable.

\textbf{Remark} For the proof it is necessary that $p_i \prod_{j \neq i} q_j \neq p_k \prod_{j \neq k} q_j$. But this is fulfilled if $a_i \neq a_k$, as one can easily verify. Cancelling out all $q_j$'s that appear in both products, one ends up with $p_i q_k \neq p_k q_i$, this is equal to $\frac{p_i}{q_i} \neq \frac{p_k}{q_k} \Rightarrow a_i \neq a_j$.

\textbf{Remark} The strict concavity of the potential function is for the algorithm with different frequencies not necessarily needed. As in the single agent case, it would be sufficient if $Q(s)$ is continuously differentiable, has a unique point at $s^*$ such that $\frac{\partial Q(s)}{\partial s}|_{s^*} = 0$ and if $Q(s) \to -\infty$ with $||s|| \to \infty$.

The smoothness assumption has only to be made for the individual utility functions, as they are part of the vector fields in the dynamics of the agents, and the potential function is only used as a Lyapunov function.
Chapter 4

Sensor Coverage

In this chapter the sensor coverage problem will be introduced under different aspects. It serves as a specific application for the proposed multi-agent approaches of the previous chapters.

The motivation for the sensor coverage problem is to find a way to distribute a group of agents in such a way that they are able to cover a region in an optimal way in order to sense events that take place in this region according to some probabilistic distribution.

One can assume that this distribution has certain properties but is unknown. Furthermore the agents should make decisions based only on local measurements and information received from neighboring agents, according to some maximal communication range between the agents. There can be situations where the agents are unable to communicate to each other because their relative distances are too large.

The problem was posed by W. Li and C.G. Cassandras in [12]. They also proposed a solution method that is a simple gradient method and requires the calculation of the gradient numerically. Unfortunately, this comes together with the need of a lot of calculation power and is quite sensitive to numerical errors and implies an analytic form of the distribution.

In this work, the problem is approached from a game theoretic viewpoint so that it fits in the framework of the previous chapters.

A short review of the solutions that were proposed by other authors in the recent literature, are given.

4.1 Sensor Coverage Problem

The position of each agent $i$ is denoted by $s_i$ and a detection probability function $p_i$ depends on the position and the detection region around the agent. One can imagine that a decreasing function with respect to the distance of each agent and an event position, may give a good representation of such a detection probability. Furthermore it is assumed that this is a radially symmetric function. The probability of detecting an event at position $x$ is a function like

$$ p_i(||x - s_i||) = \begin{cases} \in [0, 1] & \forall \|x - s_i\| \leq r_{max} \\ \approx 0 & \text{otherwise.} \end{cases} \quad (4.1) $$

Additionally one can assume that

$$ \frac{\partial p_i(||x - s_i||)}{\partial x} \approx 0 \quad \forall \|x - s_i\| \geq r_{max}. \quad (4.2) $$
The maximal detection radius is denoted by $r_{\text{max}}$. As the agents are supposed to communicate with their neighbors, the communication radius is assumed to be twice the detection radius.

The mission space $\Omega$ will be subregion of $\mathbb{R}^2$ and is assumed to be convex. Furthermore there is an unknown event probability distribution $R(x) : \Omega \to \mathbb{R}_0^+$ assigning each point a positive probability. The overall detection probability is therefore

$$F(s) = \int_{\Omega} R(x) P(x, s) dx$$

$P(x, s) = 1 - \prod_{i=1}^N (1 - p_i(||x - s_i||))$. \hfill (4.3)

The equation for $F(s)$ defines an event detection frequency in terms of the position of all the agents. The more events detected by the agents, the higher the value. Note that doubly detected events do count less, as it is important that the events are detected by only one agent, and not two agents at the same time. Otherwise, it would imply that the agents concentrate at the same position, and overlap maximal in their detecting region.

The reason for this is the function $P(x, s)$. As $p_i$ is the probability of detecting an event, the term $\prod_{i=1}^N (1 - p_i(||x - s_i||))$ is the probability that an event is not detected, and therefore $1 - \prod_{i=1}^N (1 - p_i(||x - s_i||))$ is the probability that an event at the position $x$ is detected by at least one agent.

As this is a more general formulation of the sensor coverage where not only discrete events were taken into account, but also event distributions such as oil slick for example, the summation over $x$ that denotes the position of events, is represented as an integral over a region.

4.2 Related Work

In this section different solutions to the previously defined Sensor Coverage Problem are presented. Before stating the developed results, two solution methods by authors in the recent literature whose approach is similar to the one in this document, are compared.

The first one considers a gradient method. The second one is a game theoretic approach with a discretized mission space.

4.2.1 Gradient Ascent Approach

The authors in [12] who formulated the sensor coverage problem, also proposed an algorithm to solve the problem. Their idea is to solve the problem by using a gradient ascent algorithm. In order to do that, they first calculated the derivative of (4.3) with respect to the position $s_i$ of each agent:

$$\frac{\partial F}{\partial s_i} = \int_{\Omega} R(x) \prod_{k \in B_i} [1 - p_k(x)] p_i'(||x - s_i||) \frac{s_i - x}{||x - s_i||}$$ \hfill (4.4)
where \( p'_i(|x - s_i|) \) denotes the derivative with respect to its argument. They compute the integral with a discretization

\[
\frac{\partial F}{\partial s_i x} \approx \Delta^2 \sum_{u=-V}^{V} \sum_{v=-V}^{V} \frac{\tilde{R}_i(u, v) \tilde{B}_i(u, v) \tilde{p}_i(u, v) u}{\sqrt{u^2 + v^2}}
\]

\[
\frac{\partial F}{\partial s_i y} \approx \Delta^2 \sum_{u=-V}^{V} \sum_{v=-V}^{V} \frac{\tilde{R}_i(u, v) \tilde{B}_i(u, v) \tilde{p}_i(u, v) v}{\sqrt{u^2 + v^2}}
\]

(4.5)

where \( \tilde{R}_i(u, v), \tilde{B}_i(u, v), \tilde{p}_i(u, v) \) as the discrete values of \( R(x) \), \( \prod_{k \in B} [1 - p_k(x)] \) and \( p'(x) \) respectively, \( V \) denotes a discretization parameter.

Each agent can locally compute its gradient, and update its local position according to

\[
s_i^{k+1} = s_i^k + \alpha^k \frac{\partial F}{\partial s_i^k}.
\]

(4.6)

With this approach the agents have a high computational load and are supposed to know exactly the values of \( R(x) \) and all \( p_i(x, s) \) in order to get an accurate estimate of the gradient. There are only a few application that allow such strong restrictions, as a known function \( R(x) \) implies that the agents can also be positioned by a global leader.

### 4.2.2 Discretized Mission Space

The following problem solution stems from [14] and is based on game theory. The authors propose a learning algorithm based on probability distributions.

First of all, they model the mission space \( \Omega \) in a discrete way. There are only discrete sectors in the mission space, where the agents can be situated.

Furthermore there is also a discrete time or a round time, where only one player can move at a time. The global utility function that also plays the role of a potential function was chosen to be

\[
\phi(s) = \sum_{s \in S} V(s)[1 - \prod_{P_i \in P} [1 - p_i(s, a_i)]]
\]

(4.7)

where \( P \) is the set of all players \( P_i \), the set \( S \) contains all sectors in the mission space, and \( a_i \in A_i \) is the action set of each player. In this case where no obstacles are existent, \( A_i = S \), so each player can chose its position no matter where in the mission space. The definition of \( p_i(s, a_i) \) is the same as in equation (4.1). The individual utility functions are calculated by using the Wonderful Life Utility that is going to be explained later, and yields to

\[
U_i(a) = \phi(a_i, a_{-i}) - \phi(a_i^0, a_{-i}) = \sum_{s \in S} V(s) \prod_{P_i \in P} [1 - p_i(s, a_i)].
\]

(4.8)

The author’s algorithm is known as the spatial adaptive play (SAP), where one player at a time chooses its next action randomly out of its strategy set \( A_i \) according to its strategy \( p_i(t) \in \Delta(A_i) \). \( \Delta(A_i) \) denotes the set of probability distributions over the action set \( A_i \). The probability \( p_i(t) \) is chosen accordingly to

\[
p_i^{a_i}(t) = \frac{\exp(\beta U_i(a_i, a_{-i}(t-1)))}{\sum_{a_i \in A_i} \exp(\beta U_i(a_i, a_{-i}(t-1)))}.
\]

(4.9)
By adjusting the parameter $\beta$ one can tune how probable it is for each agent, to chose the best possibility of all choices. The authors proved that the stationary distribution will be

$$\mu(a) = \frac{\exp(\beta \phi(a))}{\sum_{\tilde{a} \in A_i} \exp(\beta \phi(a))}. \quad (4.10)$$

Thus, the parameter $\beta$ can influence how close the agents approach a maximum of the potential function.

### 4.3 Sensor Coverage as a Potential Game

In this section a new approach to the Sensor Coverage Game is going to be presented. The intention is to have an algorithm that allows a more realistic realization than the previous ones.

The approach in Section 4.2.1 assumes a known function $R(x)$ in order to calculate a good estimate of the gradient. Furthermore it is necessary to have adequate computational power to evaluate the discrete sum in equation (4.5). These assumptions exclude a huge variety of applications.

In the second approach in Section 4.2.2 the authors require a previously defined discretization of the mission space and assume that each agent only moves from one sector to another. As in the approach before, the agents somehow have to know the value of their utility function in the neighboring sectors without having moved there. In some publications that deal with game theory, the authors use an oracle to find the best response in each stage of the game. This makes it clear how difficult it is to use such algorithms in a realistic environment.

Obviously these approaches are more academic examples and make it hard to apply the proposed algorithms to real world problems.

In the following section a more applicable solution is presented. It combines all advantages of the previous algorithms but omits their disadvantages such as the requirement of high computational power or global knowledge of the function $R(x)$.

A game theoretic approach together with the extremum seeking feedback as optimization algorithm is combined.

#### 4.3.1 Sensor Coverage as a Game with Continuous Mission Space

Recall the event detection frequency that can be seen as some global goal that is to be maximized

$$F(s) = \int_{\Omega} R(x) \left[ 1 - \prod_{i=1}^{N} \left( 1 - p_i(||x - s_i||) \right) \right] dx, \quad (4.11)$$

the question arises how to define the individual utility functions of each agent such that this common goal is maximized without knowing the function. In [27] the authors propose the so-called Wonderful Life Utility (WLU) that was also used by [14]. It serves as a tool to construct individual utility functions out of a global utility function. A further advantage is that the so constructed game becomes a potential game and has therefore a couple of useful properties.
The WLU measures the marginal contribution of each agent with respect to the global goal. In this case it is used in the following way:

Given a global utility function \( \phi(s) \) the individual utility functions are

\[
 u_i(s_i, s_{-i}) = \phi(s_i, s_{-i}) - \phi(s_0, s_{-i}) \tag{4.12}
\]

where \( s_0 \) denotes the null-actions of agent \( i \). The null-action is defined as the action of agent \( i \) as if it was not present in the game and therefore all sensing capacities were turned off.

By definition the individual utility functions \( u_i(s_i, s_{-i}) \) constructed in this way measure the change in the global utility function by agent \( i \). Unfortunately there is not always a physical interpretation of the \( u_i \)'s that are constructed by using the WLU, and therefore it is not always possible to use them. An advantage of the WLU is that one obtains a potential game as

\[
\frac{\partial u_i(s_i, s_{-i})}{\partial s_i} = \frac{\partial \phi(s_i, s_{-i})}{\partial s_i} - \frac{\partial \phi(s_0, s_{-i})}{s_i} = \frac{\partial \phi(s_i, s_{-i})}{\partial s_i} \tag{4.13}
\]

that is actually equal to the definition of an exact potential game.

In the Sensor Coverage Game one can use \( F(s) \) as defined in equation \((4.3)\) as a global potential function. The individual utility functions are obtained by the Wonderful Life Utility as

\[
u_i(s_i, s_{-i}) = F(s_i, s_{-i}) - F(s_0, s_{-i})
\]

\[
= \int \Omega R(x) \left[ 1 - \prod_{j=1}^{N} (1 - p_j(x, s_j)) \right] dx - \int \Omega R(x) \left[ 1 - \prod_{j=1, j \neq i}^{N} (1 - p_j(x, s_j)) \right] dx
\]

\[
= - \int \Omega R(x)(1 - p_i(x, s_i)) \prod_{j=1, j \neq i}^{N} (1 - p_j(x, s_j))dx + \int \Omega R(x) \prod_{j=1, j \neq i}^{N} (1 - p_j(x, s_j))dx
\]

\[
= \int \Omega R(x)p_i(x, s_i) \left[ \prod_{j=1, j \neq i}^{N} (1 - p_j(x, s_j)) \right] dx
\]

\[
= \int \Omega R(x)p_i(x, s_i) \left[ \prod_{j=1, j \neq i}^{N} (1 - p_j(x, s_j)) \right] dx. \tag{4.14}
\]

The integral in equation \((4.14)\) can be evaluated only over the local region \( \Omega_i \) around the agents position because of the assumption \( p_i(x, s_i) \approx 0, \forall x \geq r_{max} \). Hence the individual utility function is

\[
 u_i(s_i, s_{-i}) = \int \Omega_i R(x)p_i(x, s_i) \left[ \prod_{j=1, j \neq i}^{N} (1 - p_j(x, s_j)) \right] dx. \tag{4.15}
\]

In this special case, the individual utility functions have a nice physical interpretation. They give a notion about the detected amount only by agent \( i \). That means that the value of the
detection frequency decreases if the overlapping region of two agent increases. That is quite obvious, because one wants to position the agents in such a way that the covered region is maximized.

**Theorem 14** (Sensor Coverage Game). The Game $\Gamma_1 = \langle V, A, U \rangle$ with $V := \{1, \ldots, n\}$ as the player set, $A = \Omega$, the mission space, $U = \{u_i(s_i, s_{-i}) \mid \forall i \in V\}$ and

$$u_i(s_i, s_{-i}) = \int_{\Omega} R(x)p_i(x, s_i) \left[ \prod_{j=1 \atop j \neq i}^{N} (1 - p_j(x, s_j)) \right] dx$$

(4.16)

with continuously differentiable functions $R(x)$ and $p_i(x, s_i)$ on the domain $\Omega$, $\Gamma_1$ is a potential game with potential function

$$F(s) = \int_{\Omega} R(x) \left[ 1 - \prod_{j=1}^{N} (1 - p_j(x, s_j)) \right] dx.$$  

(4.17)

To assure that this is a potential game one has to make sure that the potential function as well as the individual utility functions are continuously differentiable, as this is necessary to verify the condition

$$\frac{\partial P(s)}{\partial s_i} = \frac{\partial u_i(s)}{\partial s_i}.$$  

(4.18)

As the differentiability of an integral expression is hard to check, one could use the Leibniz rule to exchange the integration and the differentiation. What follows is

$$\frac{\partial u_i(s_i, s_{-i})}{\partial s_i} = \frac{\partial}{\partial s_i} \int_{\Omega} R(x)p_i(||x - s_i||) \left[ \prod_{j=1 \atop j \neq i}^{N} (1 - p_j(||x - s_j||)) \right] dx$$

$$= \int_{\Omega} R(x) \frac{\partial p_i(||x - s_i||)}{\partial s_i} \left[ \prod_{j=1 \atop j \neq i}^{N} (1 - p_j(||x - s_j||)) \right] dx.$$  

This is only valid for continuously differentiable functions $p_i(||x - s_i||)$ and $R(x)$. Thus it is included as requirement in the Proposition.

**Remark** It is known that the Sensor Coverage Game in its discrete version is a submodular game. Submodular games are - like potential games - a special class of games where for example resource allocation problems are often belonging to. The authors in [13] state that finding the global optimum in such games is NP-complete. Furthermore, there are algorithms like the Greedy-Update Rule where each agent chooses not only a better-response but the best-response. In this case it is possible to calculate the so-called Price Of Anarchy that measures the ratio between the worst-case Nash equilibrium and the best-case Nash equilibrium. Using such an algorithm, the Price Of Anarchy is $\frac{1}{2}$. 
4.3.2 Sensor Coverage Game with Distance Costs

In some applications it is necessary to take into account the cost of communication between the agents, as this implies that the energy consumption for the communication decreases if the distance from one agent to another decreases.

The connection graph is in this case undirected and fixed over time. It is also convenient that this graph is modeled as a tree because usually the information is transmitted from one agent to another till it reaches the base station. It would not be very useful to have a cyclic graph in such a setup. Nevertheless, this model also allows to have this type of graphs.

To take the distance costs into account, a simple model was used. It is derived from an approach in [14] where the authors modeled a consensus problem as a potential game. One can show that if the euclidian distances between neighboring nodes are used as utility functions, the agents reach consensus.

The idea is to define a positive weight \( w \) for the consensus part. By increasing this weight, the agents will get closer to each other and by decreasing the weight, they will position themselves as good as possible in order to perform a good sensing.

The proposed potential function for the consensus problem in [14] is defined as

\[
H(s) = \sum_{P_i \in \mathcal{P}} \sum_{P_j \in N_i} \frac{|s_i - s_j|^2}{2},
\]  

(4.19)

and is the sum of least squares over the positions of all agents and their neighbors \( N_i \). Based in this, the potential function is defined in the following form

\[
K(s) = F(s) - wH(s) = \int_{\Omega} R(x)P(x,s)dx - w \sum_{P_i \in \mathcal{P}} \sum_{P_j \in N_i} \frac{|s_i - s_j|^2}{2}.
\]

(4.20)

The extended individual utility functions \( w_i \) can be constructed with the help of the Wonderful life utility

\[
v_i(s_i, s_{-i}) = K(s_i, s_{-i}) - K(s_0, s_{-i})
= u_i(s_i, s_{-i}) - w \sum_{P_j \in N_i} ||s_i - s_j||^2
\]

\[= \int_{\Omega} R(x)p_i(x, s_i) \left[ \prod_{j=1}^{N} (1 - p_j(x, s_j)) \right] dx - w \sum_{P_j \in N_i} ||s_i - s_j||^2. \]  

(4.21)

Obviously this can still be evaluated locally under the assumptions that the positions of the neighbors are known. The resulting utility functions are still a potential game with potential function \( K(s) \).

**Theorem 15** (Sensor Coverage Game with Distance Costs). The Game \( \Gamma_2 = \langle V, A, U \rangle \) with \( V := \{1, \ldots, n\} \) as the player set, \( A = \Omega^N \) the mission space, \( U = \{v_i(s_i, s_{-i}) \mid \forall i \in V \} \) and

\[
v_i(s_i, s_{-i}) = \int_{\Omega} R(x)p_i(x, s_i) \left[ \prod_{j=1}^{N} (1 - p_j(x, s_j)) \right] dx - w \sum_{P_j \in N_i} ||s_i - s_j||^2.
\]

(4.22)
with continuously differentiable functions $R(x)$ and $p_i(x, s_i)$ on the domain $\Omega$, $\Gamma_2$ is a potential game with potential function

$$K(s) = \int_\Omega R(x) \left[ 1 - \prod_{j=1}^N (1 - p_j(x, s_j)) \right] dx - w \sum_{P_i \in P} \sum_{P_j \in N_i} \frac{||s_i - s_j||^2}{2}.$$  

(4.23)

Obviously the function $H(s)$ is continuously differentiable with respect to all $s_i$, therefore there does not need to be any assumption on that function.

### 4.3.3 Sensor Coverage Game with Communication Costs

The authors of the sensor coverage problem [12] also extended the problem by taking communication costs directly into account.

Usually the autonomous agents are dependent on a finite energy resource. Hence, one might wish to optimize the relative positions of the agents weighted by the amount of detected data. The authors defined a global utility function that is increasing in the covered area denoted by $F(s)$ and decreasing in a function of the communication range weighted with all events that were detected, denoted by $G(s)$. The further a agent is distinct from its neighbors but the more events it detects, the lower is the value of the utility functions. The weight of the communication part can also be tuned by a parameter $w > 0$.

$$\max_s \int_\Omega R(x) P(x, s) dx - w \sum_{i=1}^N r_i(s_i) c_i(s)$$

(4.24)

The overall utility can therefore be written as

$$J(s) = F(s) - wG(s).$$

The part of $G(s)$ is proportional to the frequency that events are detected by agent $i$

$$r_i = \int_{\Omega_i} R(x) p_i(||x - s_i||) dx$$

and some cost $c_i$ that increases proportionally to the distance of two agents

$$c_i = \sum_{(j,k) \in l_i} e_{jk}$$

$$e_{ij} = \epsilon(||s_i - s_j||).$$

$L_i$ denotes here the set of shortest paths between agents. By considering that the graph of shortest distances is always represented by a tree it is possible to write down the next-hop nodes of each agent as a vector $H$ with $h_i$ as the next-hop node of agent $i$.

$$H = (h_1, \ldots, h_N)$$

with

$$h_i \in \{0, 1, \ldots, N\}$$
The data that is transmitted over this network consists of a cumulative vector \( z_i \) that contains all the data from the outer nodes until node \( i \)

\[
u_j^i = \mathbf{1}[h_j = i], \quad z_i = r_i + \sum_{j=1}^{N} u_j^i z_j.
\]

The cost function \( G(s) \) can hence be written as

\[
G(s) = \sum_{i=1}^{N} r_i(s_i)c_i(s)
= \sum_{(j,k) \in \mathcal{E}} e_{jk} \left\{ \sum_{i=1}^{N} 1[(j,k) \in \mathcal{L}_i] r_i \right\}
= \sum_{i=1}^{N} e_i h_i z_i.
\]  

As introduced before, one also wants to find a solution of the coverage game where the communications costs are taken into account. For this purpose the individual utility functions have to be adjusted. Again, the Wonderful Life Utility is used to construct the individual utility functions. The potential function for the communication costs is

\[
G(s) = \sum_{(j,k) \in \mathcal{E}} e_{jk} \left\{ \sum_{n=1}^{N} 1[(j,k) \in \mathcal{L}_n] r_n \right\}.
\]

Applying the WLU the resulting individual utility is

\[
g_i(s) = G(s) - G(s_0, s_{-i})
= \sum_{(j,k) \in \mathcal{E}} e_{jk} \left\{ \sum_{n=1}^{N} 1[(j,k) \in \mathcal{L}_n] r_n \right\} - \sum_{(j,k) \in \mathcal{E}} e_{jk} \left\{ \sum_{n=1}^{N} 1[(j,k) \in \mathcal{L}_n] r_n \right\}
= \sum_{(j,k) \in \mathcal{E}} e_{jk} \left\{ \sum_{n=1}^{N} 1[(j,k) \in \mathcal{L}_n] r_n \right\}.
\]

The individual utility functions are a combination of the sensor coverage part and the communication cost part and therefore

\[
w_i = u_i + w g_i
= \int_{\Omega} R(x) p_i(x, s_i) \left[ \prod_{j=1}^{N} (1 - p_j(x, s_j)) \right] \, dx
+ w \sum_{(j,k) \in \mathcal{E}} e_{jk} \left\{ \sum_{n=1}^{N} 1[(j,k) \in \mathcal{L}_n] r_n \right\}.
\]  

(4.26)
The utility functions $v_i$ still do have a certain interpretation. They make an agent to consider all the communication that is produced by agents that have a lower hierarchical status in the communication tree and additionally all the communication starting from itself and that is directed to the root node.

**Theorem 16 (Sensor Coverage Game with Communication Costs).** The game $\Gamma_2 = (V, A, U)$ with $V := \{1, \ldots, n\}$ as the player set, $A = \Omega^N$ the mission space, $U = \{w_i(s_i, s_{-i}) \mid \forall i \in V\}$ and

$$w_i(s_i, s_{-i}) = \int_{\Omega} R(x)p_i(x, s_i) \left[ \prod_{j=1, j \neq i}^{N} (1 - p_j(x, s_j)) \right] dx$$

$$+ w \sum_{j=i \lor k=i \lor r_n=r_i} e_{jk} \left\{ \sum_{n=1}^{N} 1([j, k] \in l_n]r_n \right\}$$

(4.27)

with continuously differentiable functions $R(x)$ and $p_i(x, s_i)$ on the domain $\Omega$, $\Gamma_2$ is a potential game with potential function

$$J(s) = \int_{\Omega} R(x)P(x, s)dx - w \sum_{i=1}^{N} r_i(s_i)c_i(s).$$

(4.28)

**Example** To get a better understanding of the definitions from the last section a simple example is presented. Consider the graph in Figure 4.1, where each node represents an agent. Obviously the graph is a tree where each node sends its collected information to its next-hop node until the root node - in this case agent 1 - receives it.

![Figure 4.1: Example Graph](image)

The next-hop vector looks in this case like

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_i$</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

where for example the node 2 is the next-hop node of nodes 4 and 5. The vector of cumulative data $z_i$ looks therefore

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$z_i$</td>
<td>$r_1 + r_2 + r_3 + r_4 + r_5$</td>
<td>$r_2 + r_4 + r_5$</td>
<td>$r_3$</td>
<td>$r_4$</td>
<td>$r_5$</td>
</tr>
</tbody>
</table>
Finally the utility function \( G(s) \) that contains the communication costs has the form

\[
G(s) = \sum_{i=1}^{N} e_{ih_i} z_i = e_{21}(r_2 + r_4 + r_5) + e_{31}(r_3) + e_{42}(r_4) + e_{52}(r_5)
\]

with \( e_{jk} \) proportional to the distance between node \( j \) and \( k \).

In Figure 4.2 one can see the subgraphs that are considered by each node. Obviously node 1 has to consider the whole graph because it is the root node. In Table 4.1 one can see the

\[
\begin{array}{c|c}
\text{Node} & g_i \\
1 & e_{21}(r_2 + r_4 + r_5) + e_{31}(r_3) + e_{42}(r_4) + e_{52}(r_5) \\
2 & e_{21}(r_2 + r_4 + r_5) + e_{42}(r_4) + e_{52}(r_5) \\
3 & e_{31}(r_3) \\
4 & e_{21}(r_4) + e_{42}(r_4) \\
5 & e_{21}(r_5) + e_{52}(r_5)
\end{array}
\]

Table 4.1: Individual Communication Cost Functions

resulting \( G_i \)'s for each agent. Unfortunately this extension to the coverage game needs a lot more communication and global knowledge.
4.3.4 Discrete Events

As in the previous sections, the utility function of the sensor coverage is a continuous function defined as an integral over a certain region.

As mentioned before, it is impossible to evaluate this integral in reality, because the event distribution is unknown. Nevertheless, one can reinterpret the individual utility functions under the assumptions that there are realizations of discrete events with a certain probability associated with the event density function \( R(x) \).

The value of the utility functions is therefore the sum of events taking place in the visible region of each agent where only the events detected by agent \( i \) and by none of the other agents are taking into account.

If for example there are discrete events distributed over the mission space \( \Omega \) the individual utility function is proportional to the number of events that take place in the region \( \Omega_i \) that only agent \( i \) detects

\[
u_i(s_i, s_{-i}) \equiv N(x \in \Omega_i | \text{detected only by agent } i).
\]

One can imagine a communication protocol where the agents exchange information about the detected events such that every agent only counts the events detected by himself. This necessitates some communication but can easily be realized if one associates a number with each event. Every agent has to communicate the numbers of events that were detected.

In the following, different interpretations and models for the events and how they can arise will be given. It will turn out, that it is not necessary to have a correct measurement of the individual utility functions, but that the estimate can be interpreted as a noisy version of them.

The integral in equation (4.14) can be interpreted as the probability of detecting an event during a certain time-instance \( t_k \). Normalizing the integral, one can state the probability of an event detected by agent \( i \) per time-unit as

\[
p = \frac{\int_{\Omega} R(x)p_i(x, s_i) \left[ \prod_{j=1}^{N} (1 - p_j(x, s_j)) \right] dx}{\int_{\Omega} R(x)dx}.
\]

**Events as Poisson Process** Assuming that each agents wait at position \( s_i \) for \( k \) time-units it will sense a whole set of events. As this is nothing else than counting events that pop up with a certain probability, this can be modeled as a Poisson process

\[
u_i(s_i, s_{-i}) = \frac{1}{T} \sum_{k=1}^{T} n_k = \frac{1}{T} \bar{N}(T)
\]

where \( \bar{N}(T) \) is a Poisson distributed variable with parameter \( \lambda = pT \), mean \( \mu = \lambda \) and \( \sigma^2 = \lambda \).

For large times \( T \), the Poisson distribution can be approximated by a Gaussian distribution. A new variable \( v \) is introduced that is meant to be white noise with \( \sigma^2 = \lambda \), hence equation (4.30) can be written as

\[
u_i(s_i, s_{-i}) = \frac{1}{T} \left[ \lambda + v \right]
\]

\[= p + \frac{v}{T}
\]

\[= p + w \]
where \( w \) is white noise \( \sigma^2 = \frac{1}{T^2} = \frac{p}{T} \). This result is quite intuitive and shows that the measurement of discrete events can be interpreted as the measurement of the exact representation of the individual utility function with additive white noise with \( \sigma^2 = \frac{p}{T} \). The longer the time \( T \) is chosen, the better the approximation becomes.

**Events as Bernoulli Process** If one assumes that the events happen one after the other in equidistant time intervals, they are Bernoulli distributed with the same parameter \( p \). Taking the same estimator for \( p \) that is in this case also the mean, one obtains

\[
 u_i(s_i, s_{-i}) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\text{Event detected}}
 = p + v
\]

with \( v \) as white noise with variance \( \frac{p(1-p)}{n} \) and \( \mathbf{1} \) as the indicator function if an event was detected only by agent \( i \).

**Events as Binomial Process** Imagine another scenario where \( n \) events are created at each time unit over the whole space. Detecting an event at position \( s_i \) is binomial distributed with \( p \) and \( n \). As \( p \) and \( n \) are constants the product \( \lambda = pn \) is also constant, and for large \( n \) this tends to a Poisson distribution with parameter \( \lambda \). The individual utility function is in this case the temporal mean over a time \( T_k \):

\[
 u_i(s_i, s_{-i}) = \frac{1}{T_k} \sum_{i=1}^{T_k} n_i
 = \lambda + v = pn + v
\]

with \( v \) as white noise with \( \sigma^2_v = \frac{pn}{T_k} \).

Important is that all these models have in common that the measurement of the individual utility functions can be performed by using counts of discrete events instead of calculating the integral expression and the error can be modeled as white noise. Every estimator has a *Mean-Square Error* that gives a notion about the error that is made with respect to the estimated variable. For mean estimators this mean-square error is equal to the variance of the estimated variable.

The idea is to model the error as white noise. This approach can be very restrictive as the standard deviation usually is not always white noise but can also be colored noise. Nevertheless, white noise is a standard assumption in such problems and can be seen as the worst-case scenario.

The extremum seeking of Section 3.2.1 is only recommendable for small noise variances of the noisy measurement of the utility function. For this purpose, the extremum seeking of the authors in [22] is going to be used. A short review of the algorithm was given in Section 3.1.1.

As the stochastic version of the extremum seeking requires a different approach, there will be no proof for the convergence of the Algorithm in the Sensor Coverage Game, but simulations in Chapter 5 show that an heuristic approach is very satisfactory in this case.
Example For the events that are realized as a Bernoulli process, an example where the continuous representation of the utility function are compared to the estimated version, is given here. In figure 4.3a a path for Agent 1 was chosen. The isolines $R(x) = c$ are drawn as circles. Furthermore, another agent, denoted as Agent 2, is fixed at its position.

The spatial distribution $R(x)$ of the agents is a Gaussian distribution with a standard deviation of $\Sigma = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and mean $\mu = (0, 0)$.

There are $n = 800$ events created in the mission space at each time instant where it waits for $T_k = 20$ to estimate the term $\lambda = pn$.

In Figure 4.3b one can see different realizations of the utility function in blue as they were measured if only discrete events are available and compared to that in red, the continuous version with the direct computation of the integral. The dashed lines are the standard deviation $\sigma_x = \sqrt{\frac{pn}{T_k}}$.

The local minimum in the utility function at $(x, y) = (0, 0)$ is due to Agent 2 that is fixed at its position. When both agents overlap on a large region, the utility functions of both agents decrease, as this is less optimal. One can conclude that the discrete model for the utility function with additive noise is a good approximation for the continuous version.
Chapter 5

Simulation Results

In this chapter the results of the previously introduced theories are illustrated with simulations. The simulations are meant to underline the results and visualize the ideas.

The first part consists of theoretical aspects that are necessary to be looked at in more detail in order to use the proposed methods.

The latter sections concern different aspects of the sensor coverage game together with the extremum seeking where the theoretical results can not directly be applied but simulations deliver still good results and are used for a heuristic argumentation.

The sensor coverage game was introduced independently of the number of dimensions, but the extremum seeking for multi-agent systems only for two dimensions, so that all results are only for the two-dimensional case.

Numerical issues of the simulations are going to be omitted, as this is not the main view point of this work.

5.1 Prerequisites

In this section, different results of applying the extremum seeking feedback to the sensor coverage game are presented.

To make sure that the requirements of Theorem 11 and Theorem 12 are fulfilled one has to make some assumptions. For both theorems it is necessary to make sure that the potential function is continuously differentiable, strictly concave and admits a single maximum. For the Sensor Coverage Game it is not possible to find a set-up where these assumptions are fulfilled globally. Nevertheless, one can restrict the concavity assumption to a subregion, namely some region of attraction of a local maximum of the potential function. As mentioned before, such a local maximum is also a local Nash equilibrium (see Definition 2) for the Sensor Coverage Game.

Lemma 4 assures the existence of an \( \epsilon \)-equilibrium if the potential function is bounded. Check first the boundedness assumption of the potential function and the utility functions. As the potential function is required to be defined on the whole space \( \mathbb{R}^{2N} \) with \( N \) as the number of agents this means for the special case of the Sensor Coverage Game that

\[
F(s) = \int_{\mathbb{R}^2} R(x) \left( 1 - \prod_{i=1}^{N} (1 - p_i(||x - s_i||)) \right) dx < \infty. \tag{5.1}
\]
As the functions \( p_i(\cdot) \) are defined on in the interval \([0, 1]\), the boundedness mainly depends on the function \( R(x) \).

There is still a large class of functions with the property

\[
\int_{\mathbb{R}^2} R(x) < \infty. \tag{5.2}
\]

Such a function could be for example a Gaussian function \( R(x) = \frac{1}{2\pi} e^{-\frac{1}{2}(x_1^2+x_2^2)} \) as this yields to

\[
\int_{\mathbb{R}^2} \frac{1}{2\pi} e^{-\frac{1}{2}(x_1^2+x_2^2)} dx = 1. \tag{5.3}
\]

Secondly, it must be assured that the \( \epsilon \)-equilibrium is not at \( s_i \to \infty, i = 1, \ldots, N \), but this is, in the case of a Gaussian function obviously fulfilled, as the value of the potential function decreases when the agents move away from the center and \( R(x) \to 0 \) as \( x \to \infty \).

For the upcoming simulations this function will be used for all examples. It has other nice properties, for example a single maximum, that assures that the agents agglomerate close to this maximum. This property however, does not imply that the potential function is strictly concave and admits only one Nash equilibrium.

One could also imagine other functions \( R(x) \) with more than one maximum, but this just adds local maxima to the potential function and therefore a function with a single maximum is the simplest case.

The same reasoning can be done for the individual utility functions, as they admit the same form as the potential function.

Lemma 6 assures the existence of a pure-strategy Nash equilibrium if the strategy space is compact. This is not applicable here, as the extremum seeking does not allow to restrict the action space. It is necessary to define the mission space \( \Omega = \mathbb{R}^2 \) to make sure that the agents do not collide with a boundary. One could imagine a function \( R(x) \) that goes to infinity when \( x \) goes to infinity and just restrict the mission space \( \Omega \) to a compact of \( \mathbb{R}^2 \). This would imply that the potential function is still bounded, as the integration is only done over a finite space.

In this case it might happen that one maximum is at the boundary \( \partial \Omega \) of the mission space. The problem by using the extremum seeking however is, that it needs a smooth space, where the sinusoids can be performed without obstacles as the used versions of the extremum seeking do not have any obstacle avoidance.

Nevertheless, if the function \( R(x) \) admits a maximum that is known approximately, it is always possible to define a mission space \( \Omega \) such that the agents converge to some interior point without colliding with the mission space boundary.

Imagine for example a quadratic function \( R(x) \). Obviously, the integration of \( \int_{\mathbb{R}^2} R(x) dx \) would be infinity. But if one restricts the mission space \( \Omega \) to a subspace of \( \mathbb{R}^2 \), the integral would be finite. In this case, the mission space has to be chosen such that the agents stay for sure in the interior of \( \Omega \). This can be done for a quadratic function whose maximum is known, as the agents will surely converge close to that maximum.

In Figure 5.1a the agents are expected to stay inside the region \( \Omega \) whereas they tend to reach the boundary of \( \Omega \) in Figure 5.1b. In a real example this restriction is not important, as there is always a bounded mission space but there is usually also a certain approximate knowledge of the event probability function \( R(x) \), so that its maximum is in the interior.
Furthermore the individual utility functions should be smooth, too. To check this, one can take the definition of the individual utility functions

\[
u_i(s_i, s_{-i}) = \int_{\Omega} R(x)p_i(||x - s_i||) \left[ \prod_{j=1, j \neq i}^{N} (1 - p_j(||x - s_j||)) \right] dx
\]  

and requesting this to be smooth, it must be possible to exchange the differentiation and the integration

\[
\frac{\partial^n u_i(s_i, s_{-i})}{\partial s_i^n} = \frac{\partial^n}{\partial s_i^n} \int_{\Omega} R(x)p_i(||x - s_i||) \left[ \prod_{j=1, j \neq i}^{N} (1 - p_j(||x - s_j||)) \right] dx
\]

\[
= \int_{\Omega} R(x) \frac{\partial^n p_i(||x - s_i||)}{\partial s_i^n} \left[ \prod_{j=1, j \neq i}^{N} (1 - p_j(||x - s_j||)) \right] dx.
\]

It follows that the terms \(R(x), p_i(||x - s_i||)\) and \(p_j(||x - s_j||)\) have to be smooth with respect to \(x\) and \(s_i\), otherwise it was not allowed to exchange the integration and differentiation. Assuming that \(R(x)\) is smooth, and all the \(p_i(||x - s_i||), i \in V\) are all equally chosen, meaning that all agents have a similar detection probability, one is left over with demanding that \(p_i(r)\) is smooth with respect to \(r\), as \(||x - s_i||\) is smooth almost everywhere except at \(x = s_i\). Therefore, if \(p_i(r)\) is smooth, the individual utility function \(u_i(s_i, s_{-i})\) is smooth.

There are smooth functions that are decaying very fast with \(||x - s_i|| \to \infty\) such that one could approximate the integral by integrating only over a subregion.

As these functions are only approximations of a real sensor detection probability, it is no problem to find suitable functions. The authors in [12] who introduced the Sensor Coverage
Problem, proposed the function
\[ p_i(||x - s_i||) = p_{0i}e^{-\lambda_i||x - s_i||^2} \]
where \( \lambda_i \) determines the decay-rate of the detection probability from the center of the sensor to the far field, and \( p_{0i} \) as the probability of detecting that happens at the same position as of the sensor.

But there are also smooth functions with a compact support. Such a function could be for example
\[
p_i(||x - s_i||) = \begin{cases} 
p_{0i}e^{-\frac{1}{r_{\text{max}} - ||x - s_i||^2 + \frac{1}{r_{\text{max}}}}} & ||x - s_i|| \leq r_{\text{max}} \\
0 & \text{else} \end{cases}.
\]
In the coming sections different realizations for \( p_i(||x - s_i||) \) are compared. It will be shown that the restriction of smooth functions can be released to continuously differentiable functions.

As now all necessary conditions for the two theorems are checked one can apply the methods to solve the Sensor Coverage Game. The upcoming sections will show different aspects of the requirements and how they can be released or interpreted.

As the function \( R(x) \) is chosen to be the same for all the following examples, the isolines where \( R(x) = \text{const.} \) are drawn as circles on every figure, such that one can see how the Nash equilibrium is related to the maximum of \( R(x) \).

5.2 Choice of Initial Conditions

The first question, after the choice of the parameters, concerns the initial position for the agents. Obviously, this depends on the application but also on the problem formulation and the problem setup.

Assume that for the Sensor Coverage Game with a Gaussian distribution \( R(x) \) as well as smooth detection probabilities are given. In order to apply Theorem 11 it is necessary to find initial conditions that are sufficiently close to a local maximum as local minima or saddle-points would be steady states for the Lie bracket system.

This example is meant to show that this assumption is only necessary for the theoretical framework of the proof. As all local minima as well as saddle points are unstable steady states for the Lie bracket system, they have in fact to be excluded, otherwise the Lie bracket system would not move.

Fortunately, the Lie bracket system is only an approximation for the original extremum seeking. As the original extremum seeking has no steady states, because it is a time-driven system, local minima and saddle points can be neglected.

To give a better understanding of that, a concrete example is going to be shown in this section.

Assume a perfectly symmetric function \( R(x) \), as the previously Gaussian distribution. Assume furthermore only two agents with the same detection probabilities that are also perfectly symmetric to their positions \( s_i \). Such a function could be the function
\[
p_i(||x - s_i||) = p_{0i}e^{-\lambda_i||x - s_i||^2}.
\]
The parameter \( \lambda_i \) and \( p_{0i} \) are assumed to be the same for both agents and are denoted by \( \lambda \) and \( p_0 \).

The individual utility function of Agent 1 is then

\[
u_1(s_1, s_2) = \int_{\Omega} R(x)p_0e^{-\lambda||x-s_1||^2}(1 - p_0e^{-\lambda||x-s_2||^2})dx
\]

(5.7)

with its derivative

\[
\frac{\partial u_1(s_1, s_2)}{\partial s_1} = -2\lambda \int_{\Omega} (x - s_1)R(x)p_0e^{-\lambda||x-s_1||^2}(1 - p_0e^{-\lambda||x-s_2||^2})dx.
\]

(5.8)

For \( s_1 = s_2 = (0, 0) \) this yields to

\[
\frac{\partial u_1(0, 0)}{\partial s_1} = -2\lambda \int_{\Omega} xR(x)p_0e^{-\lambda||x||^2}(1 - p_0e^{-\lambda||x||^2})dx.
\]

(5.9)

Performing a variable transformation to polar-coordinates \( x_1 = r \cos \theta \) and \( x_2 = r \sin(\theta) \) and \( dx = rdrd\theta \), the derivative is

\[
\frac{\partial u_1(0, 0)}{\partial s_1} = \begin{pmatrix}
-2\lambda \int_{\theta=0}^{2\pi} \int_{r=0}^{r_{\text{max}}} r \cos(\theta) R(r)p_0e^{-\lambda r^2}(1 - p_0e^{-\lambda r^2})rdrd\theta \\
-2\lambda \int_{\theta=0}^{2\pi} \int_{r=0}^{r_{\text{max}}} r \sin(\theta) R(r)p_0e^{-\lambda r^2}(1 - p_0e^{-\lambda r^2})rdrd\theta
\end{pmatrix}.
\]

(5.10)

The function \( R(x) \) is a symmetric function with respect to the origin, therefore it is only a function of \( r \) and is not depending in \( \theta \).

Using Fubini's rule, the integrals can easily be calculated to

\[
\begin{pmatrix}
-2\lambda \int_{\theta=0}^{2\pi} \cos(\theta) \int_{r=0}^{r_{\text{max}}} R(r)p_0e^{-\lambda r^2}(1 - p_0e^{-\lambda r^2})rdrd\theta \\
-2\lambda \int_{\theta=0}^{2\pi} \sin(\theta) \int_{r=0}^{r_{\text{max}}} R(r)p_0e^{-\lambda r^2}(1 - p_0e^{-\lambda r^2})rdrd\theta
\end{pmatrix} = \begin{pmatrix}
0 \\
0
\end{pmatrix}
\]

(5.11)

as the integral over a whole period of a sinusoid is equal to zero and therefore independent of the integral in \( r \).

As the utility function for Agent 2 is equal to the one of Agent 2, and its derivative also equal to zero, one can conclude that the point \( s_1 = s_2 = (0, 0) \) is an equilibrium.

It is also possible to say, that this cannot be a local maximum, because if both agents are at the same position they influence each other in a negative way as they both would detect the same events.

The idea now is to apply the proposed algorithm to this problem setup, and using the position \( s_1 = s_2 = (0, 0) \) as initial condition for the agents.

As it was shown that this is a steady state for the Lie bracket system, one could expect that the agents don’t move from this point.

The simulation results in Figure 5.2, where the positions of the agents at different time instances are drawn, show definitely that the agents move away from the origin and don’t get stuck at this point, although it is a steady state for the Lie bracket system.
5.2. CHOICE OF INITIAL CONDITIONS

One can conclude that the requirement that the start point of the agents must be close to a local maximum is not very restrictive as singular points like local minima do not disturb the convergence of the algorithm.

Comparing the result of the switched improvement approach to the results of Theorem 12, the results are similar. As one can see in Figure 5.4, the individual utility functions as well as the potential function reach the same values.

Obviously, the value of the initial condition must be a local minimum, as the potential function as well as the individual utility functions of both agents increase with distance to the origin as one can see in Figure 5.4.

Another interesting fact that is worth mentioning, is the nature of the local maximum of the potential function. Obviously, the final relative positions of the agents is the same in Figure 5.2 and Figure 5.3 but the global position is slightly different, although the value of the potential function reached the same value.

This is due to the fact that the maximum is not one point \( s^* \) but rather a manifold \( f(s_1^*, s_2^*) = 0 \) defined in such a way that the optimal positions of the agents are depending on each other. In this case, this manifold is obviously a circle with a certain positive radius and with its center at the maximum of the function \( R(x) \). All positions where the agents are
opposing each other around the maximum have equal values.

This example also shows that the Nash equilibria are related to the maxima of $R(x)$ only in a wider sense.

Figure 5.5 also shows the difference between the switched scheme (5.5a), and the scheme with different frequencies (5.5b) for each agent.
5.2. CHOICE OF INITIAL CONDITIONS

Figure 5.4: Potential Function and Individual Utility Functions over Time

Figure 5.5: Agent Coordinates over Time
5.3 Comparison of Detection Probabilities

This section is meant to show different models for the event detection probabilities and to draw conclusions about the conditions needed for the proposed algorithm with the switching scheme of Theorem 11 to converge. The results can directly be translated to the scheme with different frequencies; therefore, only results for the switched scheme will be treated in detail.

**Smooth Event Detection Probability**

In the first example, a slightly different event detection probability as it was originally proposed in [12], is chosen. It takes the form

\[ p_i(||x - s_i||) = p_0 e^{-\lambda_i ||x - s_i||^2}, \]  

and the difference to the function in [12] is that the square of the euclidian norm was chosen, instead of the euclidian norm itself.

This function is a smooth function on the whole space \( \mathbb{R}^2 \) and with an a-priori event distribution as it was proposed before, this Sensor Coverage Game fulfills all necessary assumptions of Theorem 11 locally and can therefore be solved by the proposed algorithm. The values for the different parameters can be found in Table 5.1.

| \( \omega_i \) | 10 |
| \( \alpha_i \) | \( 0.01 \sqrt{10} \) |
| \( c_i \) | \( \frac{50}{\sqrt{10}} \) |
| \( \phi_i \) | 0 |
| \( \lambda_i \) | 15 |
| \( p_{0_i} \) | 0.7 |

Table 5.1: Parameter Values

agent moves is 120 periods of its own sinusoid until the next agents is allowed to move.

Figure 5.6 shows the trajectories of the agents, and the final positions.

The radii were drawn accordingly to the \( \lambda_i \)'s and in this case they denote a radius of \( ||x - s_i||^2 = 0.3 \), that corresponds to a value of \( 0.7 e^{-15 \cdot 0.3^2} \approx 0.18 \) of the event detection probabilities.

In Figure 5.7 one can see the evolution of the potential function. Although the individual utility functions of the agents are influencing each other in a negative way, meaning that during the movement of an agent, the utility functions of the other agents can decrease, the potential function is monotonically increasing over time until it reaches a constant level. This behavior coincides with the definition of the approximate finite improvement path in Lemma 5.

**Continuously Differentiable Event Detection Probability**

This result will now be compared to a different model for the detection probability of the agents.
85 5.3. COMPARISON OF DETECTION PROBABILITIES

Again, all agents are assumed to be equipped with the same detection probability function, that is in this case not a smooth but only a continuously differentiable function

\[ p_i(r_i) = \begin{cases} 
\frac{p_0}{r_i^{4}} (r_i^2 - r_{max_i}^2)^2 & r_i \leq r_{max_i} \\
0 & r_i > r_{max_i} \end{cases} \] (5.13)

Obviously this function allows to perform the integration only over the region \( ||x - s_i|| \leq r_{max_i} \), as it is zero outside. The derivative is also continuous and the resulting game with individual utility functions is a potential game. As this function leads to non-smooth utility functions and non-smooth potential functions, it does not fulfill all the assumptions of Theorem 11. Nevertheless, the following results show that it is still possible to use the proposed algorithm.

The parameters are in this case similar to the case before and can be found in Table 5.2. At the place of \( \lambda_i \) there is now the maximal visible radius \( r_{max_i} \), and has a similar meaning, namely to give an approximate model of the sensing radius of the sensors. It is also possible to have agents with different detection probability functions and different sensors. The game would still be a potential game.

The results of the simulations with the chosen detection probability function are shown in Figure 5.8. The trajectories as well as the final positions are similar to the case before, and
the agents converge close to expected positions. In Figure 5.9 one can see the evolution of the individual utility functions and of the potential function over time. Again, the potential function is monotonically increasing over time, while the individual utility functions can be influencing each other in a negative way.

One can conclude that this choice for the detection probability functions leads to good results, although it does not fulfill the assumptions of Theorem 11. The restriction to smooth utility functions can therefore be released to continuously differentiable functions, at least for some cases. Evidently, this is not to understand as a proof, but as a heuristic result.

### Piecewise Constant Event Detection Probability

Another question that might arise, is if the assumption to continuously differentiable function can be released even more. An obvious choice for a simple function, that is even not continuous

---

**Figure 5.7: Potential Function and Individual Utility Functions over Time**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_i$</td>
<td>10</td>
</tr>
<tr>
<td>$\alpha_i$</td>
<td>0.01\sqrt{10}</td>
</tr>
<tr>
<td>$c_i$</td>
<td>$\frac{50}{\sqrt{10}}$</td>
</tr>
<tr>
<td>$\phi_i$</td>
<td>0</td>
</tr>
<tr>
<td>$r_i$</td>
<td>0.50</td>
</tr>
<tr>
<td>$r_{max,i}$</td>
<td>0.25</td>
</tr>
<tr>
<td>$p_{0i}$</td>
<td>0.7</td>
</tr>
</tbody>
</table>

**Table 5.2: Parameter Values**
but could be a simple model for a detection probability would be a function like

\[ p_i(r_i) = \begin{cases} p_{0i} & r_i \leq r_{\text{max},i} \\ 0 & r_i > r_{\text{max},i}. \end{cases} \]  

(5.14)

One could think about a sensor that detects all the events in a previously determined region with a constant probability and for sure nothing, if it happens outside that region.

The parameters are chosen similarly to the previous cases and can be found in Table 5.3. The resulting positions for different time-instances are shown in Figure 5.8. The trajectories were omitted because they are distracting as the agents jump around in the mission space.
without converging. This can also be seen in Figure 5.11 where the individual utility functions as well as the potential functions are drawn over time. As the individual utility functions are obviously discontinuous, there are also jumps in the utility functions and the agents are not able to climb up the gradient.

It is also important to mention that for this class of detection probabilities it is impossible to check if this set-up leads to a potential game, as it is not allowed to exchange the differentiation and the integration by differentiating the potential function and the individual utility functions

\[ \frac{\partial P}{\partial s_i} = \frac{\partial u_i}{\partial s} \neq \int_{\mathbb{R}^2} \frac{\partial}{\partial s} R(x)p_i(||x - s_i||) \prod_{j=1, j \neq i}^{N} (1 - p_j(||x - s_j||)) dx. \]  

(5.15)
5.3. COMPARISON OF DETECTION PROBABILITIES

Figure 5.10: Positions of Agents at different Time Instances

Figure 5.11: Potential Function and Individual Utility Functions over Time
5.4 Distance and Communication Costs

In the following simulations, different possibilities of taking communication costs into account according to Sections 4.3.2 and 4.3.3 are compared. In all of the following cases the communication graph is considered to be a chain with agent 1 as the base station. This agent is fixed and cannot move. The graph is shown in Figure 5.12.

Figure 5.12: Communication Tree

The event probability $R(x)$ is again chosen to be a Gaussian distribution $R(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(x^2+y^2)}$ and the event detection probability functions of the agents are chosen to be $p_i(||x-s_i||) = p_0 e^{-\lambda ||x-s_i||^2}$. This assures furthermore that the utility functions and the potential function are smooth with respect to $s_i$.

5.4.1 Distance Cost

As the Sensor Coverage part of the potential function fulfills locally all the assumptions that are necessary for the Theorems 11 and 12, one has only to check the additional term $H(s)$ caused by the distance costs, as this is only an additive function

$$K(s) = \int_{\Omega} R(x) \left[ 1 - \prod_{j=1}^{N} (1 - p_j(x, s_j)) \right] - w \sum_{P_i \in \mathcal{P}} \sum_{P_j \in N_i} \|s_i - s_j\|^2_2 \right].$$  \hspace{1cm} (5.16)

$H(s)$ is always bounded for bounded $s_i$ and is quadratic; therefore it is a smooth function. These individual utility functions as well as the potential function fulfill all necessary assumptions locally.

The first simulation in this section shows the simplest case where only distance costs are considered as they were introduced in Section 4.3.2. It has the form of a chain where every node has only one neighbor. As shown before, this type of game is again a potential game, hence the improvement strategy can be applied in order to end up in a local Nash equilibrium. The chosen constants as well as the weight between Sensor Coverage and Distance Costs are shown in Table 5.4.
### 5.4. Distance and Communication Costs

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_i$</td>
<td>10</td>
</tr>
<tr>
<td>$\alpha_i$</td>
<td>$0.002\sqrt{10}$</td>
</tr>
<tr>
<td>$c_i$</td>
<td>$\frac{10}{\sqrt{10}}$</td>
</tr>
<tr>
<td>$\phi_i$</td>
<td>0</td>
</tr>
<tr>
<td>$p_{0i}$</td>
<td>1</td>
</tr>
<tr>
<td>$\lambda_i$</td>
<td>15</td>
</tr>
<tr>
<td>$w$</td>
<td>0.07</td>
</tr>
</tbody>
</table>

Table 5.4: Parameter Values

The first result in this subsection is shown in Figure 5.13. The communication graph is drawn as black lines between the agents. The evolution of the individual utility functions as well as the potential function as shown in Figure 5.14. Obviously, the behavior is the same as expected and similar to the examples before. The value of the potential function is monotonically increasing over time until it reaches a final value and the agents stop.

In contrast to this example, another slightly different example is shown. In Figure 5.15, one can see the simulation result for the same case with different initial conditions.

![Figure 5.13: Positions of Agents at different Time Instances](image)

(a) $t = 0$
(b) $t = 3.14 \cdot 10^3$
(c) $t = 6.29 \cdot 10^3$
(d) $t = 9.42 \cdot 10^3$
(e) $t = 1.26 \cdot 10^4$
(f) $t = 1.57 \cdot 10^4$

Figure 5.13: Positions of Agents at different Time Instances
This example is meant to show that the final positions of the agents are not always as one would expect. The problem that the algorithm only converges to local Nash equilibria is leading to unexpected results sometimes as one can see in Figure 5.15.

Nevertheless, the potential function reaches the same value as with in the previous case as one can see in Figure 5.16.

This is only a special case as there exist more complex examples where local equilibria do not lead to the same value of the potential function. This is always to be bared in mind when using the algorithm. Unfortunately, this is due to the fact, that the problem was posed in such a way that only local information can be used, therefore there is no possibility to find a global Nash equilibrium by only having local information.
5.4. DISTANCE AND COMMUNICATION COSTS

Figure 5.15: Positions of Agents at different Time Instances

Figure 5.16: Potential Function and Individual Utility Functions over Time
5.4.2 Communication Cost

In this example, the requirements for the potential function have to be checked again. The additional term in the potential function that defines the communication costs

\[ J(s) = \int_{\Omega} R(x)P(x,s)dx - w \sum_{i=1}^{N} r_i(s_i)c_i(s) . \]  

is defined as the weighted version of the distance costs before. Therefore, the functions are also bounded and smooth for the previously defined functions \( R(x) \) and \( p_i(||x - s_i||) \).

This example shows the results that are obtained with the potential game with communication costs as they were defined in Section 4.3.3. The communication graph is the same as in the example before. The chosen constants are shown in Table 5.5. This example is meant to show that, although the communication load of the agents is much higher than in the case before, the resulting positions in Figure 5.17 of the agents are similar and therefore one can conclude that a simpler solution can lead to a comparable result.

One can see that the individual utility functions have a higher value the more an agents is distinct from the base station in terms of its hierarchical state in the communication tree. This is due to the fact how the distance costs were defined. Every node has to take all the information into account that all the following nodes are sending.

The individual utility function of each agent also depends on its place in the communication tree. The fact that Agent 2 has a lower utility function comes from the fact that it is further away from the maximum of \( R(x) \) but also from the fact that the amount of sensed information of the other agents influence its utility function in a negative way.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \omega_i )</td>
<td>10</td>
</tr>
<tr>
<td>( \alpha_i )</td>
<td>0.002(\sqrt{10} )</td>
</tr>
<tr>
<td>( c_i )</td>
<td>( \frac{10}{\sqrt{10}} )</td>
</tr>
<tr>
<td>( \phi_i )</td>
<td>0</td>
</tr>
<tr>
<td>( p_{0i} )</td>
<td>1</td>
</tr>
<tr>
<td>( \lambda_i )</td>
<td>15</td>
</tr>
<tr>
<td>( w )</td>
<td>0.3</td>
</tr>
</tbody>
</table>

Table 5.5: Parameter Values
5.4. DISTANCE AND COMMUNICATION COSTS

Figure 5.17: Positions of Agents at different Time Instances

Figure 5.18: Potential Function and Individual Utility Functions over Time
5.5 Agents with Unicycle Dynamics

This section is meant to give more insight to the algorithm where the agents underly unicycle dynamics. Furthermore, the $c_i$’s are chosen differently, such that the speed of convergence of each agent is different. The event detection probability of the agents is chosen to be

$$p_i(||x - s_i||) = \begin{cases} p_{0i}e^{-\frac{1}{r_{max}^2 ||x - s_i||^2 + \frac{3}{2}r_{max}^2}} & ||x - s_i|| \leq r_{max} \\ 0 & \text{else} \end{cases}$$  \hspace{1cm} (5.18)

This is, as introduced before, a smooth function with a compact support. The parameters were chosen accordingly to Table 5.6.

<table>
<thead>
<tr>
<th>Agent</th>
<th>$\Omega_i$</th>
<th>$\omega_i$</th>
<th>$c_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha_i$</td>
<td>0.1</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>$\phi_i$</td>
<td>0</td>
<td>11</td>
<td>20</td>
</tr>
<tr>
<td>$r_{max,i}$</td>
<td>0.3</td>
<td>12</td>
<td>30</td>
</tr>
<tr>
<td>$p_{0i}$</td>
<td>1</td>
<td>13</td>
<td>40</td>
</tr>
<tr>
<td>(a)</td>
<td>5</td>
<td>14</td>
<td>50</td>
</tr>
<tr>
<td>(b)</td>
<td>6</td>
<td>15</td>
<td>60</td>
</tr>
</tbody>
</table>

Table 5.6: Parameter Values

The resulting trajectories are shown in Figure 5.19. Because of the choice of different $c_i$, the agents converge with different velocities.

One can also see that only the $\omega_i$’s have to be different, whereas the $\Omega_i$’s can be the same without destroying the convergence.

The potential function as well as the values of the utility functions are shown in Figure 5.20. As expected, the potential function is steadily increasing until it reaches a constant level, whereas the individual utility functions are not monotonically increasing.
5.5. AGENTS WITH UNICYCLE DYNAMICS

Figure 5.19: Positions of Agents at different Time Instances

Figure 5.20: Individual Utility Functions over Time
5.6 Discrete Events

In this section the simulation results that were done with the algorithm presented in section 3.1.1 and the measurements of discrete events, are presented. As one can see in Table 5.7 the gains are chosen in such a way, that all the assumptions of Theorem 5 are fulfilled. The author who introduced this algorithm also proposed to chose different frequencies for the different agents, such that the agents do no influence each other.

\[
\begin{align*}
\alpha_i(k) & \quad 0.5 \\
\epsilon_i(k) & \quad 0.5 \\
T_i & \quad 1 \\
h & \quad 0.07 \\
p_{0i} & \quad 1 \\
\omega_i & \quad 0.6\pi \\
p_{0i} & \quad 2 \\
\omega_i & \quad 0.7\pi \\
p_{0i} & \quad 3 \\
\omega_i & \quad 0.8\pi
\end{align*}
\]

(a) (b)

Table 5.7: Parameter Values

The model for the event appearance is the same as in the example in section 4.3.4, where a Bernoulli Process was underlying. It is assumed that at every time instance there are \( n = 400 \) events created, and each agent waits for \( T_k = 15 \) time instances to calculate an average.

The events are Gaussian distributed over the mission space with \( \Sigma = \begin{pmatrix} 0.5 & 0 \\ 0 & 0.5 \end{pmatrix} \) and mean \( \mu = (0, 0) \). The model for the detection probability is chosen as before to be an exponential function of the form \( p_i(||x - s_i||) = p_{0i} \exp \frac{||x - s_i||^2}{2} \). On Figure 5.21 the evolution of the positions of each agent are shown for different time-instances.

The evolution of the utility functions of the agents is shown in Figure 5.22, where one can see the noisy nature of the utility function.

The disadvantage of this method is, that the initial conditions must be chosen close enough to the maximum of the Gaussian distribution, because otherwise it could happen that the agents don’t detect any events. This is a quite obvious requirement.

Remarks on the Implementation In order to simulate the detection of stochastically appearing events, a suitable implementation of the event detection model has to be found. In this case, there were 400 events created at each time step of the agents. For each event a uniformly distributed number was created and compared to the value of the function \( p_i(||x - s_i||) \) with \( x \) as the position of the event. If the random number was less than the value of the function, the event was considered as detected, and if not it was considered as not detected. Afterwards, all events detected by two agents were sorted out again. Therefore, only events detected by a single agent were considered.
Figure 5.21: Positions of Agents at different Time Instances

Figure 5.22: Individual Utility Functions over Time
Chapter 6

Summary and Future Work

This work shows how the extremum seeking can be used to solve distributed optimization problems in dynamic, multi-agent environments.

In the single agent case, the extremum seeking was shown for different dynamical models such as the single integrator, double integrator and the unicycle model to be practically globally uniformly asymptotically stable. In the focus of the analysis is the approximation of the original extremum seeking by a Lie bracket system.

The analysis with the help of Lie brackets is a new method of how the extremum seeking can be interpreted. Even more complex systems such as the unicycle model can be examined in that way. In connection with the notion of practical stability this gives a very general view to this kind of algorithms, where a sinusoidal input leads to a gradient motion.

In the multi-agent case, two different schemes were proposed and proved, building up on nonlinear system techniques and game theory. That some of the requirements can be released was shown by simulations and concrete examples. The results also showed the limits of the algorithms such as the convergence to a local maximum. The fact that the algorithms were proven for strictly concave problems with a single maximum but applied to problems with multiple maxima and local minima had to be treated in detail. That local minima and saddle-points do not have to be excluded as starting points, as well as the individual utility functions can be chosen as continuously differentiable functions instead of smooth functions, was shown by simulations.

The interpretation of the utility functions in the sensor coverage problem as the measurement of discrete events brings the algorithm closer to real-world applications. Even for a high level of noise, the extremum seeking converged in the sensor coverage game. This shows a certain level of robustness of the algorithm.

The idea of using only local measurements and still maximizing a global goal by the usage of a special feedback and individually constructed utility functions is forward-looking and still natural. Natural in this sense means that this modeling is similar to the models used in society where each individual is equipped with its own utility function at yet it seems that a common goal is solved.
In natural sciences the problem is often inverted. The individual utility functions of participants in a group are known, but the global goal is unknown. Take for example the global economy and the stock market, it is almost a philosophical question where this evolution will lead, having in mind that every company has its own goal. This might be one reason why game theory is a major tool in economy.

The use of such an approach, where every agent has its own utility function that it tries to optimize, also brings additional robustness to the system. It is not necessary to know how many agents are participating the group, neither is it necessary to tune any parameters. The optimization will always end up in a local maximum of the potential function even if agents are joining the group during the optimization process.

Obviously, this idea brings a lot of advantages to the field of multi-agent systems and can be extended to other applications than the sensor coverage problem.

6.1 Future Work

There are some open questions concerning the extremum seeking connected to potential games and especially the sensor coverage game. It is interesting to investigate how to extend the idea of Lie brackets to the case where obstacles are present. One idea is to find a smooth feedback that allows to avoid obstacles and to use the Lie brackets as approximative system.

A very interesting feature of Theorem 6 is that the inputs can have an additional time-varying term. Therefore, the inputs of the extremum seeking can be extended to functions with decaying gains, such that the maximum is reached exactly. This idea is similar to the extremum seeking for a noisy measurement. Further investigations are necessary in order to find a proof for the convergence of the extremum seeking with a noisy version of the individual utility function.

The theories that were introduced here, are using an unconventional approach, as they construct a connection between nonlinear systems science and game theory - but this is also the reason why they are closer to real-world problems as other approaches. It could also be nice to implement the algorithm into some real-world systems like robots in order to see how it can be applied in reality.
Appendix A

One-Dimensional Extremum Seeking

As some examples refer to the one-dimensional version of the extremum seeking, this section should give more insight of how one can relate the one-dimensional to the two-dimensional extremum seeking.

The system equations for the one-dimensional extremum seeking yield to

\[
\dot{s}_i = c(v_i(s) - e_i h) \sqrt{\omega} \sin(\omega t) + \alpha g(s_i, e_i) \sqrt{\omega} \cos(\omega t)
\]

\[
\dot{e}_i = -e_i h + v_i(s).
\]

The Lie bracket of the vector fields \( f \) and \( g \) can be calculated to

\[
[f, g] = -\alpha c \nabla_{s_i} v_i(s)
\]

and the Lie bracket system finally yields with Theorem 6 to

\[
\dot{s}_i = \frac{1}{2} \alpha \nabla_{s_i} v_i(\bar{s})
\]

\[
\dot{e}_i = -e_i h + v_i(\bar{s}).
\]

This is a simplified version of the two-dimensional case, and describes the continuous version of a gradient ascent algorithm. Therefore, all results of the two-dimensional case can be translated to the one-dimensional case.
Bibliography


