Torstein Nordgård-Hansen

Optical Properties of Palladium Gold Alloy Thin Films and Modeling of Their Use in Hydrogen Sensors

Master's thesis in Electronics Systems Design and Innovation
Supervisor: Dag Roar Hjelme
Co-supervisor: Michael Aaron Fried
June 2022

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Electronic Systems

NTNU
Norwegian University of
Science and Technology

Torstein Nordgård-Hansen

Optical Properties of Palladium Gold Alloy Thin Films and Modeling of Their Use in Hydrogen Sensors

Master's thesis in Electronics Systems Design and Innovation
Supervisor: Dag Roar Hjelme
Co-supervisor: Michael Aaron Fried
June 2022

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Electronic Systems

NTNU
Norwegian University of
Science and Technology

Sammendrag

Denne masteroppgaven omhandler bruk av metalltynnfilmer i optiske overflateplasmonbaserte hydrogensensorer. Metallisk palladium er unikt i at det både kan absorbere og frigjøre hydrogen fra og til atmosfæren under normale forhold. Dette fører til en endring i den optiske brytningsindeksen, og gjør det til et ideelt materiale til bruk i hydrogensensorer. Et antall tynnfilmer av palladium-gullegeringer ble produsert ved hjelp av en simultansprutdeponerings prosess der legeringsforhold og filmtykkelse ble variert. Filmtykkelsene ble målt med et nålprofilometer og et optisk fasescanninterferometer, ruheten ble målt med fasescanninterfero meteret og med et atomkraftmikroskop, legeringsforholdene ble målt med energidispersiv røntgenspektroskopi og brytningsindeksspektrene ble målt med ellipsometri. Dette har resultert i et omfattende datagrunnlag for videre utvikling av overflateplasmonbaserte hydrogensensorer med tynnfilmstrukturer produsert ved hjelp av prosessene benyttet i denne oppgaven.

Det ble dessuten utviklet en programvarepakke som ble benyttet til å simulere hvordan forskjellige designparametere og forhold påvirket fire nøkkelmåltall knyttet til overflateplasmoner i stablede tynnfilmstrukturer. Flere simuleringer ble gjennomført basert på målingene gjort som en del av denne oppgaven, med fokus på effekten av legeringsforhold, filmtykkelse og ruhet på sensitivitet og spesifisitet i overflateplasmonsignaler. Arbeidet resulterte i et fundament for videre arbeid, både teoretisk og praktisk, med det overordnede hydrogensensorprosjektet denne oppgaven er en del av.

Summary

This thesis focuses on the use of metal thin films in optical surface plasmon based hydrogen sensors. Palladium is unique in that it can both absorb and release hydrogen under normal atmospheric conditions, changing its optical refractive index, thereby making it an ideal material for hydrogen sensors. Several thin-films of different palladium gold alloys have been manufactured using co-sputtering, where the alloying ratios and film thicknesses were varied. The thickness of the samples was measured using a stylus profilometer and a 3D optical profilometer using optical phase interferometry scanning, the roughness was measured using optical phase interferometry scanning and atomic force microscopy, the alloying ratios were measured using energy dispersive spectroscopy and the refractive index spectra were measured using ellipsometry. This has provided an extensive dataset for further development of surface plasmon based hydrogen sensors produced using the processing steps investigated here.

In addition, a software package was built and used to simulate the effects of different design parameters and conditions upon four key figures of merit for plasmons in layered thin film structures. Several simulations were run using the material parameters acquired as part of this project, yielding information on the effects of alloying ratio, film thickness and film roughness on sensor sensitivity and precision. This provides a fundament for further work, both theoretical and practical, on the hydrogen sensor project of which this thesis is part.

Acknowledgments

First of all I would like to thank my supervisor, professor Dag Roar Hjelme, for giving me the opportunity to work on this project, as well as contributing invaluable discussions and guidance over the last year. I also want to thank my co-supervisor, Michael Aaron Fried, for many good discussions and guidance, as well as for invaluable help at the lab, and for performing the EDS, AFM and ellipsometry measurements.

The Research Council of Norway is acknowledged for the support of the Norwegian Micro- and Nano-Fabrication Facility, NorFab. I would also like to thank the staff at NTNU NanoLab for accommodating the project and training me in the relevant instruments, as well as Nathan Hale for making the ellipsometer available to this project.

Finally, I want to thank my parents for their support and encouragement, in particular my mother for proofreading and help with the hydrogen absorption equilibrium equations.

Contents

Sammendrag **i**

Summary **ii**

Acknowledgments **iii**

1 Introduction **1**
1.1 Advantages of optical hydrogen sensors 1
1.2 Existing optical hydrogen sensors 2
1.3 Motivation for this thesis 2

2 Theoretical background **4**
2.1 Surface plasmons in thin film structures 4
2.2 Refractive index of palladium gold alloys 7
2.3 Effect of hydrogen on palladium 9
2.4 Effective media approximation 11
2.5 Surface roughness 13
2.6 Ellipsometry 13

3 Thin-film production and characterization **14**
3.1 Overview of produced samples 14
3.2 Cleaning and lithography 15
3.3 Film deposition 16
3.4 Liftoff process 18
3.5 Film characterization 18
3.6 Ellipsometry 19

4 Thin-film results and discussion **20**
4.1 Film thickness measurements 20
4.2 Film roughness measurements 24
4.3 Film composition measurements 29
4.4 Ellipsometry measurements 33

5 Software model **37**
5.1 Intended use of the model 37
5.2 Unified parameter scan system 37

5.2.1 Implemented figures of merit 38
5.3 Physical materials 38
5.4 Toolbox for modeling materials through an effective media approximation . . 38

6 Simulations and discussion 41
6.1 Effects of the gold film 41
6.2 Effects of hydrogen 42
6.3 Optimization of palladium alloy layer thickness 47
6.4 Effects of surface roughness 50

7 Further work 53

8 Conclusion 55

9 Nomenclature 57

10 Sources 59

Appendices 64

A Fabrication schedule 65

B Function by function documentation 67
B.1 Changes to *analyticalmaterials* 67
B.2 The *ema* module 68
B.3 Changes to *materials* 68
B.4 The *targetfigures* module 69
B.4.1 Figure of merit estimators 69
B.4.2 Scan functions 69
B.4.3 System builders 70
B.5 Changes to the project structure 71

C Model source code 72

D Simulation script 102

E Profilometer leveler 111

Chapter 1

Introduction

Hydrogen sensors see broad use, both in traditional industries and as part of the emerging use of hydrogen as an energy carrier. Concerns have been raised over the risk of explosions or leakages with regards to hydrogen transport, storage and tanking, both in the industry and in end-user infrastructure such as hydrogen cars. These sectors are therefore in need of cheap and reliable sensors for monitoring both leakages and storage levels. Sensors are also needed when drilling for hydrogen, an emergent field which may enable future expansion of hydrogen-dependent industries [1]. Similarly, the prospect of underground hydrogen storage will require extensive monitoring of leakages [2]. This will require accurate and low-cost hydrogen sensors able to withstand harsh environments over long times with minimal drift, and without the risk of electric sparks causing an explosive hazard.

1.1 Advantages of optical hydrogen sensors

Hydrogen sensors have a long history stretching back over a hundred years to the filling stations for airships [3]. As described in a review by Hübert et al., there exists many established hydrogen sensor technologies [4]. Though some optical and acoustical sensors have been developed, most sensors have electrical transducers. Among the multitude of available electrical sensors, some exhibit a large range of operation, others a high sensitivity or a quick response time. One of the main drawbacks of many electrical sensors is their poor resistance to electromagnetic (EM) noise which might interfere with analog electrical signals, making them challenging to integrate in some industrial or high voltage systems.

A major advantage of optical fiber sensors is their ability to withstand harsh environments. Fiber Bragg grating based sensors have been reported to work in high temperatures, pressures and radiation levels. By choosing the fiber material and grating production technique, the fibers can be made resistant to hydrogen-induced attenuation, radiation-induced attenuation and annealing effects caused by high temperatures [5]. Unwanted effects caused by high-pressure environments can, for the most part, be mitigated using compensatory pressure sensors and careful calibration. This makes optical sensors a prime candidate for hydrogen sensing in hydrogen production and storage, as well as in the petroleum and aerospace industries. Another benefit of optical sensors is their resistance to EM noise. While electrical sensors may be vulnerable to interference in the analog signals from EM-induced currents, the sensing elements of fiber optical sensors are not based on electricity and are therefore resistant to EM-induced currents.

One major concern of traditional electrical sensors is the danger they might pose in an explosive environment. Most electrical sensors involve transducers with electrical currents or voltages, which might be considered a risk in certain industrial installations since leakage currents might ignite explosive gasses [6, section 7]. The same regulations cover the maximum power emitted by optical equipment, but the limit for emitted power from lasers that is assumed to be able to ignite an explosion, 35 mW, is far above what is typically used by sensor systems [6, section 5.7.1]. The relevant area classifications and guidelines are specified in NEK TS 420C [7].

1.2 Existing optical hydrogen sensors

The field of optical sensors is under rapid development, and has been identified as an enabling technology by both the European Commission and The Research Council of Norway [8], [9]. Recent developments in quick and cheap nanostructure fabrication and the use of alloyed metals for optical devices have allowed for new developments in the field of surface plasmon based optical sensors. This has resulted in several recent papers on the viability of more complex fiber optical hydrogen sensors. The study of surface plasmon based hydrogen sensors is well established, with palladium based prototypes reported on for well over twenty years [10], [11]. The use of optical fiber based sensors together with palladium alloys have also been reported on, and the use of palladium alloys is a field in rapid development [12]–[16].

Several alloys and structures have been proposed. One alloy of interest is gold and palladium, where promising results have been reported [17]. While pure palladium shows a higher sensitivity to hydrogen than its alloys, the gold alloying has been reported to reduces the problematic hysteresis behavior often observed with regards to hydrogen absorption. Recently the use of multi-layered structures has been reported on, with a palladium film deposited on top of a gold film yielding good results [16]. Another recent development in the field is the use of tilted fiber Bragg gratings inscribed in the fiber core [18], [19]. This creates an easily recognizable comb transmission spectrum caused by optical coupling between the fiber core and cladding, which facilitates the detection of changes in the cladding modes caused by changes outside the cladding. Another advantage of this method, is that the sensing films can be deposited on top of the cladding, removing the need to strip away the cladding and exposing the core during production. This greatly reduces the chance of breaking the fiber in the active region.

1.3 Motivation for this thesis

This master thesis is part of a project concerned with developing a good surface plasmon based optical hydrogen sensor, which is again part of the Enersense research area at NTNU [20]. The overarching plan is to produce an optical fiber with a tilted fiber Bragg grating (TFBG) inscribed in the core and a layered nano-film structure deposited outside the cladding as illustrated in figure 1.1. Laser light will enter the fiber core from one end, and the TFBG will introduce a characteristic comb pattern on the transmitted light detected at the other end of the fiber. Some of the light scattered from the core modes to the cladding modes will couple back into the core, thereby modulating the transmission spectrum of the fiber. By changing the surface plasmon conditions in the thin-film structure, the amount of light in the cladding modes that either couples to the surface plasmons, to the core modes or is extinguished, will be changed. This will then change the modulation on the fiber transmission spectrum [16]. Note that this ray-optics based description is simplified, and does not fully describe the coupled mode effects at work. A layered structure consisting of a palladium gold alloy deposited on top of a gold thin-film is believed to yield good results, based both on similar sensors reported on in the literature and on promising simulations reported on in this thesis. The gold film has been shown to improve the overall detectability of the signal and increase the coupling between cladding modes and surface plasmons [16], [21].

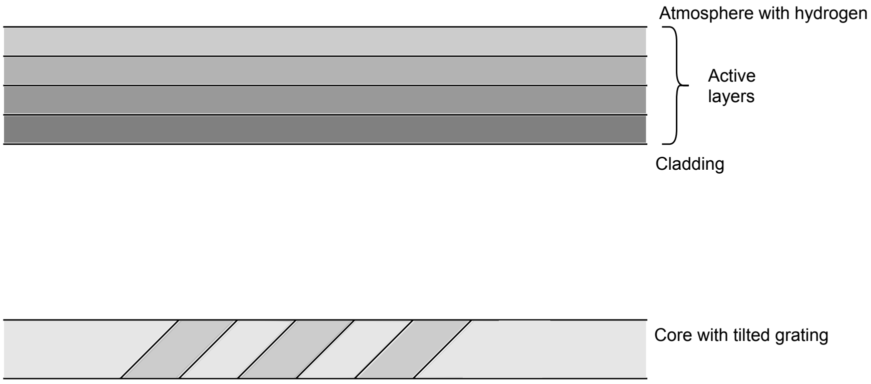


Figure 1.1: Cross section of the proposed hydrogen sensor, the fiber has rotational symmetry around the fiber core. The number of layers are chosen for illustration purposes. The figure is not drawn to scale. Figure and caption from the specialization project [22].

The focus of this thesis has been on the layered thin-film structure. The refractive index of gold is sufficiently characterized in the literature, but the properties of palladium has been shown to vary greatly with small changes in the thin-film production process [23]. Similarly, the refractive index of gold palladium alloys is not sufficiently characterized. As discussed in the specialization project preceding this thesis, accurate characterization of materials is essential for creating useful simulations when developing real-world devices [22].

The proposed thin-film structure has many degrees of freedom in its design parameters, from alloying ratios to layer thicknesses and film roughness. A system for optimizing these parameters based on different figures of merit would be of great benefit for the further development of the project. By providing a fast and reliable simulation framework, the number of prototypes that must be fabricated can be greatly reduced.

The hydrogen sensor project is therefore in need both of reliable data for the gold palladium alloys and a design parameter simulator to help optimize the design based on the acquired data. This project has produced and characterized the optical properties of several gold palladium alloy thin films of different thicknesses using the fabrication processes that is planned to be used for the final hydrogen sensor. In addition, a simulation software package aimed at design parameter scans has been developed based on the thin film simulator created in the specialization project, and simulations have been run based on the refractive index data acquired as part of this thesis [22].

Chapter 2

Theoretical background

2.1 Surface plasmons in thin film structures

A surface plasmon polariton, often called a surface plasmon, is a coupled optical and electron-density wave that can propagate along a metal dielectric interface [24]–[27]. The optical component of the surface plasmon decays exponentially in both directions out from the interface, while the longitudinal electron-density wave propagates at the optical frequency together with the optical wave [26, p. 326]. These surface waves are special in that their wave number β is highly dependent on the refractive index of the plasmon's surroundings, making them an ideal mechanism for sensing small changes in the refractive index of surrounding materials. The dispersion relation of a surface plasmon on the interface between two infinite half planes is given by equation 2.1, where β is the plasmon wave number, k_0 the incident wave number and ϵ_1 and ϵ_2 the electrical permittivities of the half planes [25, eq. 2.14]. The dispersion relation becomes more difficult to express analytically with more complex structures, it is therefore often useful to analyze surface plasmons using numerical simulations.

$$\beta = k_0 \sqrt{\frac{\epsilon_1 \epsilon_2}{\epsilon_1 + \epsilon_2}} \tag{2.1}$$

The existence of and conditions for surface plasmons can be derived from Maxwell's equations for a surface bound wave [24, p. 369]. It can further be shown that surface plasmons will always be transverse magnetic (TM) polarized, it is therefore sufficient so simulate them in this polarization when investigating their properties [25, p. 27].

It is impossible to excite a surface plasmon in a metal dielectric interface directly from the dielectric side without a grating or a similar nanostructure. This is because the wave number of the plasmon, β, will always be larger than the wave number k_0 in the dielectric [25, p. 42]. One solution to excite plasmons is to couple an evanescent wave to the plasmon as shown in figure 2.1 [25], [26], this configuration is called a Kretschmann configuration. A prism is placed on a metal film, and the incident light undergoes total internal reflection on the prism metal interface. The evanescent wave is then coupled to the plasmon on the metal air interface. This configuration is similar to the interaction between cladding modes in a fiber and thin-films deposited on top as described in section 1.3, and has been widely used in surface plasmon based hydrogen sensors [10], [28]. The reflection spectra arising from a Kretschmann coupling is shown in figure 2.2, the characteristic dip in reflection arises from the plasmon coupling.

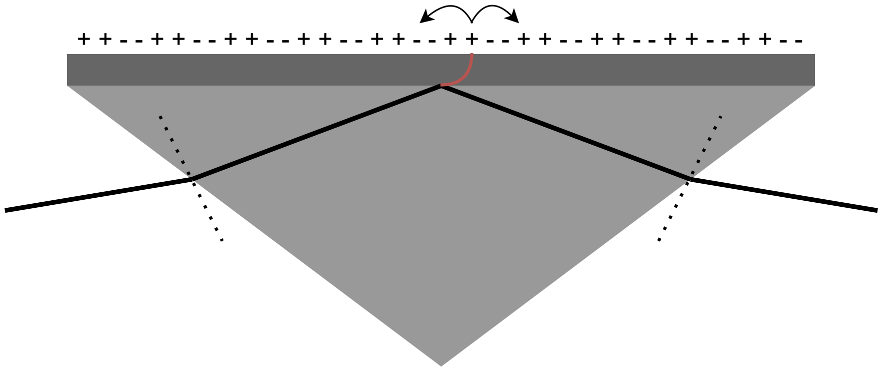
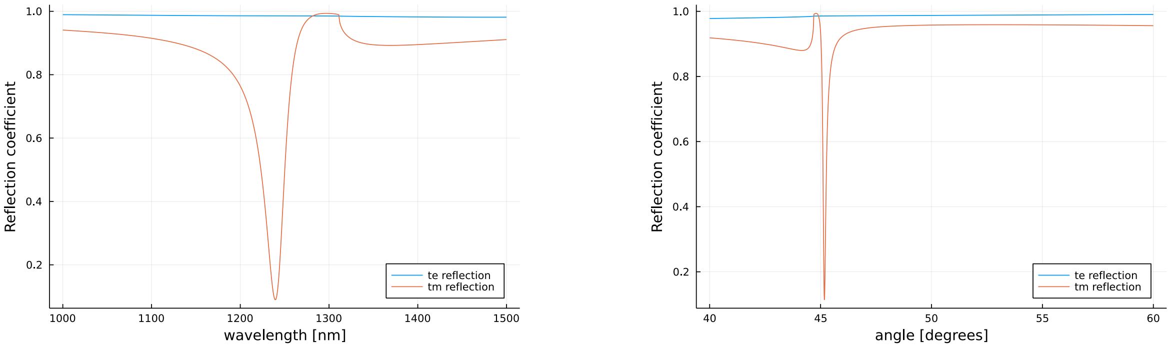


Figure 2.1: Prism coupling to surface plasmon using attenuated total internal reflection in the Kretschmann configuration. The light enters the prism and is fully reflected on the prism metal interface, the evanescent field excites a plasmon. The electron density wave is shown as positive and negative charges moving back and forth in the dielectric and not in the metal for readability. Figure redrawn and caption taken from [25, fig. 3.4].



(a) Wavelength reflection coefficient spectrum.

(b) Angular reflection coefficient spectrum.

Figure 2.2: Wavelength and angular reflection coefficient spectra for a plasmon exited using a Kretschmann configuration with a silica prism and a 30 nm thick gold thin-film. Note that the transverse electric (TE) reflection coefficient (blue) is slightly less than one, and neither the TE nor the TM reflection coefficients are ever larger than one.

Several figures of merit can be defined for the wavelength reflection spectra for thin-film structures supporting surface plasmons. Four primary figures of merit has been used in this thesis, these are illustrated in figure 2.3. They are defined to provide information on how the surface plasmon dip moves and changes in the reflection spectra with different design or environmental parameters. The plasmon dip reflection coefficient minimum and plasmon dip wavelength provides information on how the dip will move in the reflection spectrum, and by extension how sensitive the plasmon dip is to changes. The plasmon dip peak to peak indicates how simple it is to detect the plasmon dip as a distinct feature of the reflection coefficient spectrum, and the plasmon dip full width at half maximum (FWHM) indicates how sharp the peak is, and by extension how precisely the plasmon dip wavelength can be detected. The figures of merit are defined as follows:

- The plasmon dip reflection coefficient minimum is defined as the reflection coefficient at the bottom of the characteristic plasmon dip.
- The plasmon dip wavelength is defined as the wavelength in the reflection spectrum at which the reflection coefficient minimum is observed.
- The plasmon dip peak to peak is defined as the different between the local maximum on the short wavelength side of the plasmon dip in the spectrum and the plasmon dip reflection coefficient minimum.
- The plasmon dip full width at half maximum is defined as the difference in wavelength between the two points on the plasmon dip where the reflection coefficient equals the plasmon dip reflection coefficient minimum plus half the plasmon dip peak to peak.

Note that these figures relate to the thin-film reflection spectrum only. The transmission spectrum of the proposed hydrogen sensor described in section 1.3 will depend on the interaction between the tilted fiber Bragg grating and the thin-film structure deposited on the cladding, where a plasmon dip with a too narrow FWHM has been shown to be less detectable [21].

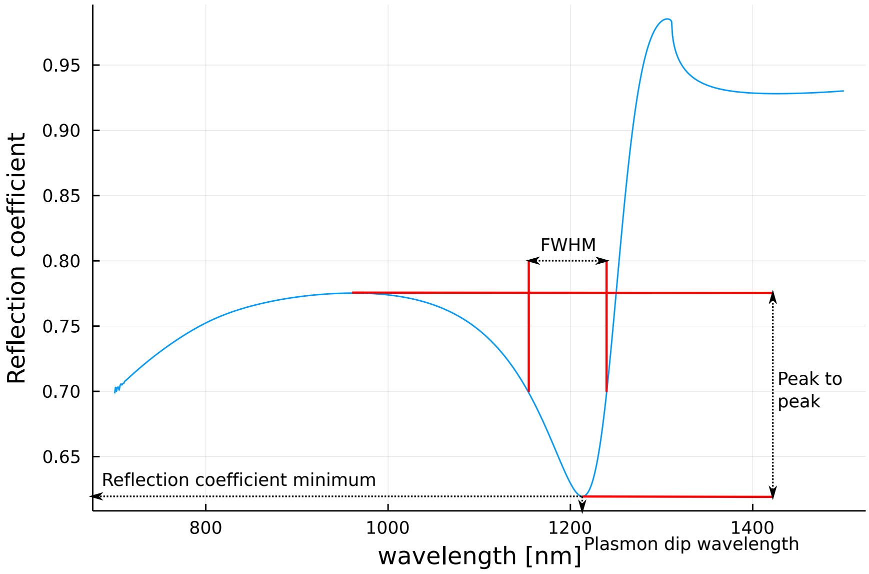


Figure 2.3: Example of a reflection coefficient spectra with several figures of merit indicated. The simulation is of a 30 nm gold film deposited on silica with a 10 nm palladium film on top, the refractive indices of the metals are taken from section 4.4. The plasmon reflection coefficient minimum, plasmon dip peak to peak, plasmon dip wavelength and plasmon dip full width at half maximum (FWHM) are shown.

The interaction between an incident wave and a layered thin-film structure can be modeled using a transfer matrix approach as described in the specialization project and in the literature [22], [26]. The concept is based on the idea of solving the Fresnel equations individually for each interface and bulk material in the structure to create a set of transfer matrices. These are then combined into one scattering matrix for the system, thereby describing the near-infinite number of reflections inside the structure shown in figure 2.4. The process is described in depth in the specialization project, and an implementation was written in Julia [22], [29], [30]. It supports simulations of semi-infinite layered thin-film structures made from media with isotropic and linear optical responses.

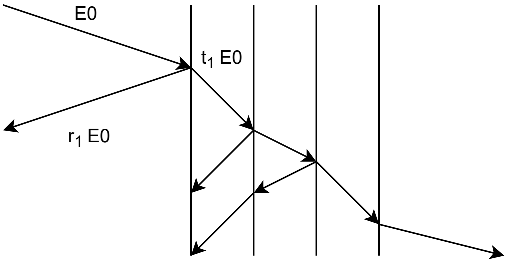


Figure 2.4: A field $E0$ is incident on the left hand side of a layered structure. A field r_1E0 is reflected at the first interface, while another field t_1E0 is transmitted. A near infinite number of similar reflections occur inside the structure. Figure and caption from the specialization project [22].

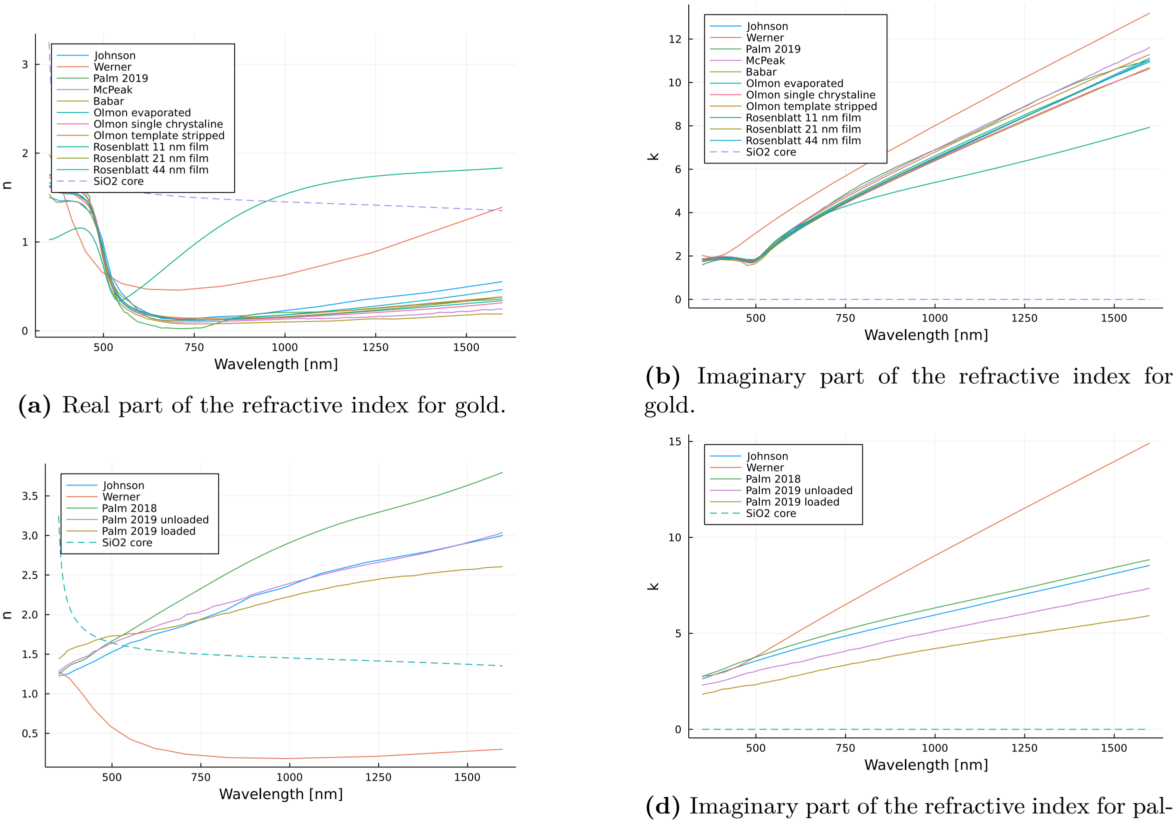
2.2 Refractive index of palladium gold alloys

When modeling the optical response of structures, the complex optical refractive index of the materials is sufficient to describe most linear effects. This index can be defined as in equation 2.2, where n_{real} is the real part of the refractive index and κ the complex part, often called the extinction coefficient in lossy media such as metals [24, eq. 1.6, 1.7]. Note that, for lossless dielectric media, $n = n_{real}$.

$$n = n_{real} - i\kappa \tag{2.2}$$

The refractive index of metals has been extensively studied, and a number of sources for gold and palladium is presented in figure 2.5 along with a theoretical SiO_2 model (dashed) for scale [14], [31]–[39]. With the exception of the measurements by Werner et al. (orange) and the 11 nm thin-film measured by Rosenblatt et al. (green), the different gold measurements shown in figure 2.5a and 2.5b are in agreement [32], [37]. Seeing as the palladium measurements by Werner et al. from the same paper, shown in orange in figure 2.5c and 2.5d, are in disagreement with other data from the literature, it is reasonable to assume that any gold measurements for films at least 21 nm thick should follow the normal curve exemplified by the measurements reported by Johnson et al. [31]. The refractive index of palladium displays a much larger difference between sources than gold. The measurements by Palm et al., labeled “loaded” and “unloaded” for samples with and without hydrogen, show that the difference in palladium’s refractive index caused by exposure to hydrogen is smaller than the difference between different literature sources. This indicates that the properties of palladium differs significantly by production process [33]. It is reasonable to assume the difference is caused by variations in the production process, rather than for instance the measurement process, since the difference between articles is not present in the reported gold measurements originating from the same research groups. This deviation in palladium production has been confirmed by Carter et al. [23].

There is however a clear trend shown in figure 2.5c that, with the exception of Werner et al., n_{real} for palladium is larger than n_{real} for silica [14], [32]. It is reasonable to assume that any palladium film should follow this trend. An effect of this is that total internal reflection of light incident on an infinite half-plane of palladium gold alloy from an infinite half-plane of silica is impossible. Though more complex thin-film structures may circumvent this limitation, it is reasonable to assume that a direct coupling from silica to a palladium alloy thin-film will yield a poor surface plasmon coupling. The improved surface plasmon coupling caused by an intermediary gold layer, which has a significantly smaller real refractive index than silica, has been confirmed in the literature [21].



(a) Real part of the refractive index for gold.

(b) Imaginary part of the refractive index for gold.

(c) Real part of the refractive index for palladium.

(d) Imaginary part of the refractive index for palladium.

Figure 2.5: Real and imaginary parts of the refractive index of gold and palladium as reported in the relevant literature.

The accurate control of the refractive index of materials is one of the most important aspects of designing a successful optical sensor. As described in section 2.1, surface plasmons are sensitive to the refractive index of the materials in which the evanescent tails propagate. Since the sensor design proposed in chapter 1 couples cladding modes to plasmons at a fixed angle, this control becomes even more important. Several theoretical models, such as the Fuchs Sonderheim-Drude Lorentz model, have been proposed to accurately predict the refractive index of metals, they do however require correct parameters [40]. The most promising approach to date is therefore still to directly measure the relevant metals or alloys being used in a proposed device [15]. In the ideal case, the measured samples are fabricated using the same process as the intended final device.

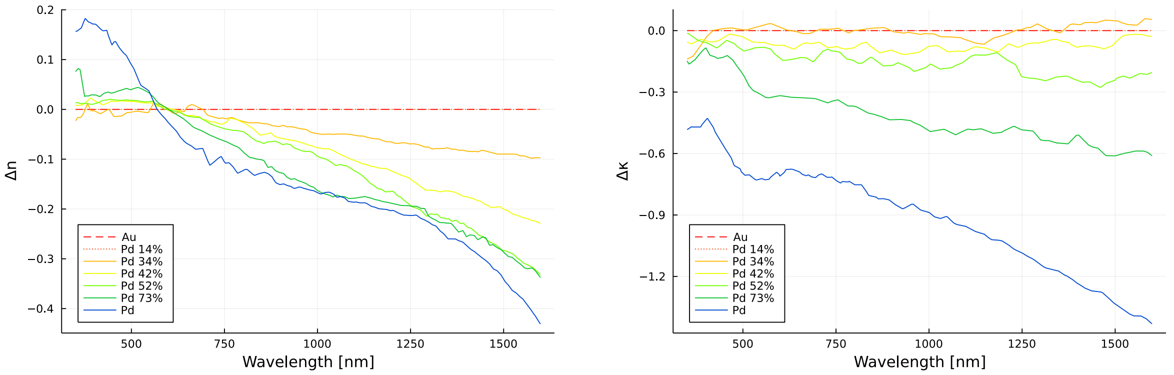
2.3 Effect of hydrogen on palladium

Palladium and its alloys have become one of the main hydrogen sensitive materials used in optical fiber hydrogen sensors due to its ability to both absorb and release hydrogen under normal atmospheric conditions [41]. The study of this interaction is therefore important when developing a surface plasmon based hydrogen sensor. The absorption and release of hydrogen is a normal absorption process, and its equilibrium can therefore be described by the law of mass action [42]. Given an equilibrium $H_2(g) \rightleftharpoons 2H(M)$ between hydrogen in the air $H_2(g)$ and in metal $H(M)$, the law og mass action gives equation 2.3 for the steady state equilibrium, where K is a function of the temperature T, p_{H_2} the partial pressure of hydrogen in the air and $C_{H(M)}$ the atomic concentration of hydrogen absorbed in the metal. It then follows that $C_{H(M)}$ is related to p_{H_2} by equation 2.4 at constant temperature, sometimes known as Sievert's law [10].

$$\frac{C_{H(M)}}{\sqrt{p_{H_2}}} = K \tag{2.3}$$

$$C_{H(M)} \propto \sqrt{p_{H_2}} \tag{2.4}$$

As discussed by Gong et al., alloying metals can enable the creation of tailored functional materials for optical use [15]. This ability also extends to palladium's hydrogen absorption when alloyed with gold as reported by Palm et al. and shown in figure 2.6 [33]. It has been shown that, when exposed to hydrogen, the refractive index of palladium can be expressed as $n(C_{H(M)}) = h(p_{H_2})n_0$, where $n(C_{H(M)})$ is the complex refractive index of palladium with an absorbed atomic concentration of hydrogen $C_{H(M)}$, n_0 is the refractive index of palladium without hydrogen and $h(p_{H_2})$ a non-linear function of the hydrogen partial pressure in the air [11]. It is reasonable to assume that this approximation holds for palladium gold alloys as well, it is therefore possible to estimate h for for any hydrogen concentration and alloying ratio based on the equation 2.4 and experimental data such as what has been reported by Palm et al. [33]. An implementation of such a model is described in appendix C, though it should be noted that an implementations should ideally be made based on data measured on the relevant thin-films at hand and be fitted to more than two measurements per alloy. Though a lower palladium concentration leads to a lower change upon exposure to H_2, and by extension a lower sensitivity in any sensor, alloying with gold has been show to reduce the chance of cracking upon hydrogen exposure. This will lead to an increase in the longevity of any potential sensor [33]. Palladium gold alloys have been successfully demonstrated in previous hydrogen sensor prototypes [43].



(a) Change in real refractive index upon H_2.

(b) Change in imaginary refractive index upon H_2.

Figure 2.6: Measured complex refractive index change upon exposure to hydrogen of seven different Pd_xAu_{1-x} alloys. The alloys were measured without hydrogen in the lattice and under an atmosphere of 7 bar H_2. Figure redrawn and caption partially taken from Palm et al. [33, fig. 1c].

One concern with regards to palladium and palladium gold alloy thin films is a pronounced hysteresis response when exposed to hydrogen [17]. This effect has been demonstrated to be largest at the first exposure to hydrogen, all sensing films should therefore be exposed to hydrogen before the final measurements are taken [39]. Another concern is the difference in material properties between different device geometries, such as semi-infinite thin-films or nano-particles. It has been demonstrated that the structural differences between these categories cause a fundamental change in hydrogen absorption caused by the differences in internal stress [17]. Any experimental data used for hydrogen absorption in palladium gold alloys should therefore originate from samples with the same geometry as the proposed structure being modeled.

Palladium is known to expand when exposed to hydrogen, and related material stress effects have been observed in palladium gold alloys [33]. This effect will increase the thickness of a palladium gold alloy thin-film when exposed to hydrogen, changing the effective refractive index n_{eff} for any plasmons near the thin-film. This effect on plasmons is separate from the change in refractive index caused by the hydrogen loading. The expansion depends on both the alloying ratio and the geometry of the sample, it is therefore not trivial to model the expansion accurately [17], [33]. Simulations performed as part of this thesis and presented in chapter 6 shows that the effects of the thickness increase and refractive index change is qualitatively the same with respect to full width at half maximum and plasmon dip wavelength. The plasmon dip reflection coefficient minimum and plasmon dip peak to peak are however affected oppositely, making the accumulated effect harder to predict. The combination of the two effects can, given accurate data on both the hydrogen induced refractive index change and the expansion, be modeled using the design parameter simulation system presented in this thesis.

2.4 Effective media approximation

When modeling the refractive index of a composite media, an effective media approximation, EMA, is often used. An EMA is a statistical model that approximates the electrical permittivity of an inhomogeneous material based on its components, these models are therefore heavily dependent on assumptions of the materials microstructure [24, chap. 2.4]. The two most commonly used EMAs are the Maxwell Garnett and Bruggeman models, each with its own separate assumptions [24, chap. 2.5]. These models may be used to give an indication of the optical properties of alloyed materials, though empirical measurements will always be more precise as the EMAs do not account for changes in the nano-structure of metals caused by alloying, where the components do not form distinct domains.

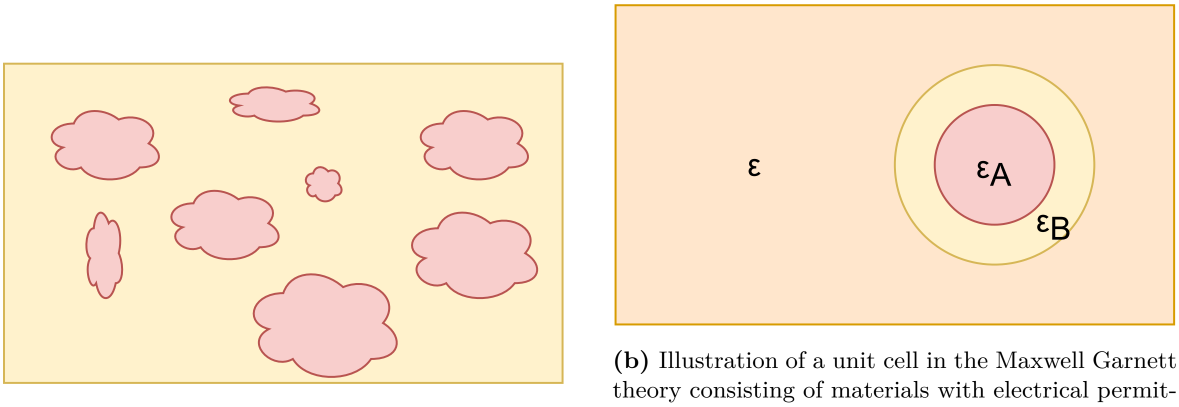
The Maxwell Garnett model assumes that one or more additives is suspended in a host media, and that each domain of the additive can be modeled as a covered sphere as shown in figure 2.7. The model is a generalization of the Lorentz-Lorentz EMA, and can be expressed by equation 2.5, where a total of i materials with permittivity ϵ_i and fill factor f_i are suspended in a material with permittivity ϵ_H, resulting in a total permittivity ϵ [24, eq. 2.22]. In practice it is more useful to solve the equation for ϵ. A general solution for i components are given in equation 2.6. Not that the Mazwell Garnett model only yield good approximations up to a fill factor $f_A \approx 0.4$, and should as a rule of thumb not be used for fill factors above $f_A \approx 0.3$ [24, p. 69].

$$\frac{\epsilon - \epsilon_H}{\epsilon + 2\epsilon_H} = \sum_i f_i \frac{\epsilon_i - \epsilon_H}{\epsilon_i + 2\epsilon_H} \tag{2.5}$$

$$\begin{aligned} \epsilon &= \epsilon_H \frac{1 + 2\Sigma}{1 - \Sigma} \\ \Sigma &= \sum_i f_i \frac{\epsilon_i - \epsilon_H}{\epsilon_i + 2\epsilon_H} \end{aligned} \tag{2.6}$$

The Bruggeman model assumes the components are mixed at a ratio such that any unit cell has a probability equal to the fill factors of being one or the other component as shown in figure 2.8. The model is expressed by equation 2.7 for a composite consisting of i different materials [24, eq. 2.25]. When solved for ϵ, the equation turns into a polynomial of the same order as the number of components, and so no general solution exists for an arbitrary number of components. The solution for a two-composite material is however readily solved as shown in equation 2.8 [24, eq. 2.26]. The Bruggeman EMA is symmetrical with regards to the choice of materials, and is theoretically valid for all values of f_i. Note that, as the number of components increases, so does the number of possible solutions to the polynomial equations. In order to select the correct solution, several physical limits related to electromagnetic fields can be used [24, chap. 2.7]. A fast algorithm for determining the correct solution has been developed by Jansson and Arwin [44].

$$\sum_i f_i \frac{\epsilon_i - \epsilon}{\epsilon_i + 2\epsilon} = 0 \tag{2.7}$$

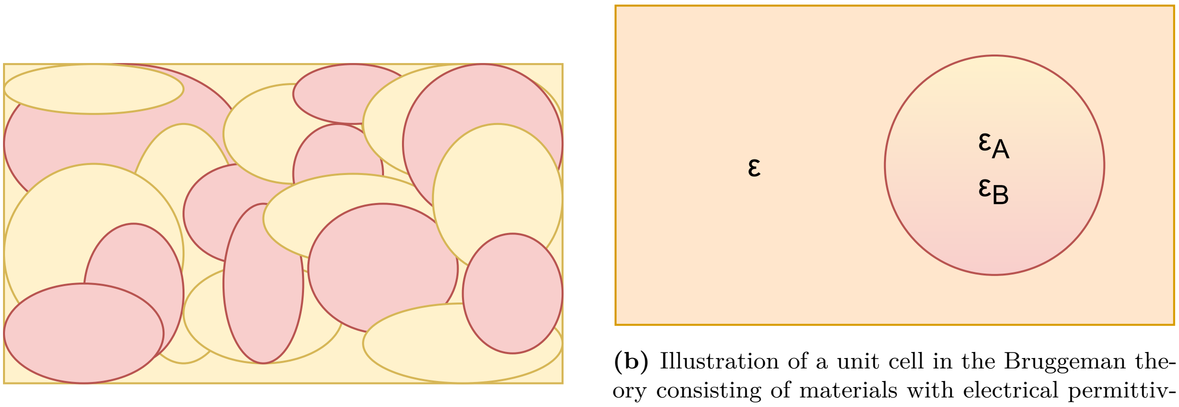


(a) Illustration of microstructure in a material suitable for the Maxwell Garnett theory.

(b) Illustration of a unit cell in the Maxwell Garnett theory consisting of materials with electrical permittivities ϵ_A and ϵ_B in an external atmosphere with permittivity ϵ.

Figure 2.7: Illustration of the microstructure and random unit cell assumed in the Maxwell Garnett theory. Figures redrawn from [24, fig. 2.7].

$$\epsilon = \frac{1}{4}\left[(3f_A - 1)\epsilon_A + (3f_B - 1)\epsilon_B \pm \sqrt{\left[(3f_A - 1)\epsilon_A + (3f_B - 1)\epsilon_B\right]^2 + 8\epsilon_A\epsilon_B}\right] \tag{2.8}$$



(a) Illustration of microstructure in a material suitable for the Bruggeman theory.

(b) Illustration of a unit cell in the Bruggeman theory consisting of materials with electrical permittivities ϵ_A and ϵ_B in an external atmosphere with permittivity ϵ.

Figure 2.8: Illustration of the microstructure and random unit cell assumed in the Bruggeman theory. Figures redrawn from [24, fig. 2.8].

One important use for the Bruggeman model is in modeling the surface roughness of samples when analyzing ellipsometric data. This approximation holds for samples where the diameter of the typical grain size in the roughness is smaller than the optical wavelength used by the ellipsometer. The surface roughness will cause diffraction of incident light unequal to what is expected from a perfectly smooth surface, thereby distorting the results acquired by ellipsometry. It has been shown experimentally that, in most cases, this distortion can be modeled as a layer consisting of the sample and atmosphere with a fill factor of $f_{sample} = 0.5$ using the Bruggeman EMA, where the thickness of the layer is proportional to the root mean square (RMS) roughness of the sample [45]. This approach can also be used when simulating the effect of surface roughness on the optical response of a structure.

2.5 Surface roughness

The roughness of a surface describes how much it deviates from an ideal smooth surface. It consists of both the size distribution of the grains that make up the roughness, as well as the variation in height caused by the grains. A complete description of an arbitrary surface roughness must therefore include a statistical distribution of the grain sizes in all three dimensions, as well as a measure of the grain density. It is however seldom useful to work with more than a couple properties of a surface roughness at a time, various figures can therefore be derived to describe aspects of the roughness.

This thesis focuses on the root mean square (RMS) surface roughness as its primary figure, since it describes the height variation caused by the roughness and can be used to model the optical effects of the roughness as described in section 2.4. The effective media approach used for modeling the roughness does however only hold when the grain size in the roughness is smaller than the optical wavelength. Imaging using instruments such as optical phase interferometers, atomic force microscopes (AFMs) or scanning electron microscopes (SEMs) can be used to verify if the grains are significantly smaller or larger than the optical wavelength by imaging the grains directly. Additionally, the correlation length of a height map is related to the grain size, and indicates the maximum length between two points where the measured heights are related. This figure can be used to investigate trends in the surface roughness grain size in the plain.

2.6 Ellipsometry

Ellipsometry is an optical technique used for thin-film and surface characterization [24, chap. 8]. It was first described in the late 1800s, but took off in the 1970s as a result of fast and cheap computers. The technique is based on measuring the change in polarization of a light beam reflected on a sample, often for several incident angles and optical wavelengths. It is therefore closely related to the field of reflectometry.

The results obtained by ellipsometry are often fitted to a model of the sample in order to obtain the desired properties. When measuring the complex refractive index of a thin-film, the material is often fitted to a parameterized model such as a Drude or B-spline model [24, chap. 8.4]. B-spline modeling is based on fitting polynomials to local parts of a curve together with a B-spline basis function. These models are therefore not physical, and offer only a continuous and smooth curve fit to the observed data. The Drude model on the other hand, is a physical model based on a kinetic understanding of the interactions inside metals. This model may therefore yield better results for some materials, but is unable to contain material effects not explained by the underlying Drude model theory. Though not rooted in a physical model, Kramers-Kronig consistent B-spline models have been shown to give good results in infrared spectroscopy, especially when the exact chemical composition of the sample is not known [46]. When several samples share some optical properties, multi-sample analysis can be used to increase the amount of data available for the model fitting algorithms. It is useful to have some prior knowledge of the thin-film thickness and surface roughness when fitting the models to avoid over-fitting. This is because both of these parameters can heavily influence the reflection measured by the ellipsometer.

Chapter 3

Thin-film production and characterization

The overarching goal of the lab work has been to produce thin-films with varying alloying ratios and thicknesses and measure their complex refractive index. The thickness and surface roughness was measured to improve the ellipsometric results, the atomic composition of the samples was measured to verify the co-sputtering process. The alloying ratios were chosen with a focus on 60% and 70% atomic palladium concentrations, since these alloys have shown promising results in the literature [33]. The film thicknesses were chosen to observe both thin-film and bulk effects in the different alloys [37]. A total of 16 different thin-films were produced with thicknesses of a few tens of nanometers, and four samples were produced with thicknesses over 200 nanometers. The first group was produced to examine thin-films similar to what will be used in the hydrogen sensor, while the second group was made to examine if there are any significant thickness-related effects. One pure gold thin-film and one pure palladium thin-film were produced to be able to compare the process with measurements from the literature described in section 2.2.

3.1 Overview of produced samples

Several samples were produced at NTNU NanoLab for the project [47]. All samples consist of a silicon wafer with a thin film made from either gold, palladium or an alloy of the two. Some of the samples are patterned using a process of negative lithography and liftoff in order to create a step edge on the thin films. The thickness, surface roughness and atomic makeup of all samples were measured before the samples were removed from the cleanroom at NanoLab. The complex refractive index of the samples was later measured using ellipsometry. All thickness and roughness measurements have been made available online, but note that the data is neither sorted nor cataloged [48].

Samples were made according to the schedule given in appendix A, a schematic overview of the production and characterization process is shown in figure 3.1. The main test series consists of samples 5 to 24 A and B, where A and B indicate samples with and without lithographic pattering. Two sub-series of samples were produced. When producing samples 8 to 12 and

13 to 17 A and B, only the thin-film deposition time was varied within each of the ranges. In addition, wafer 2 and samples 1 to 4 were made to develop and tune the lithographic process. These samples were not characterized since their production process was less controlled, and all samples underwent the patterning process. Note that wafer 1 is missing from the naming scheme, as it was damaged partway during the lithography process.

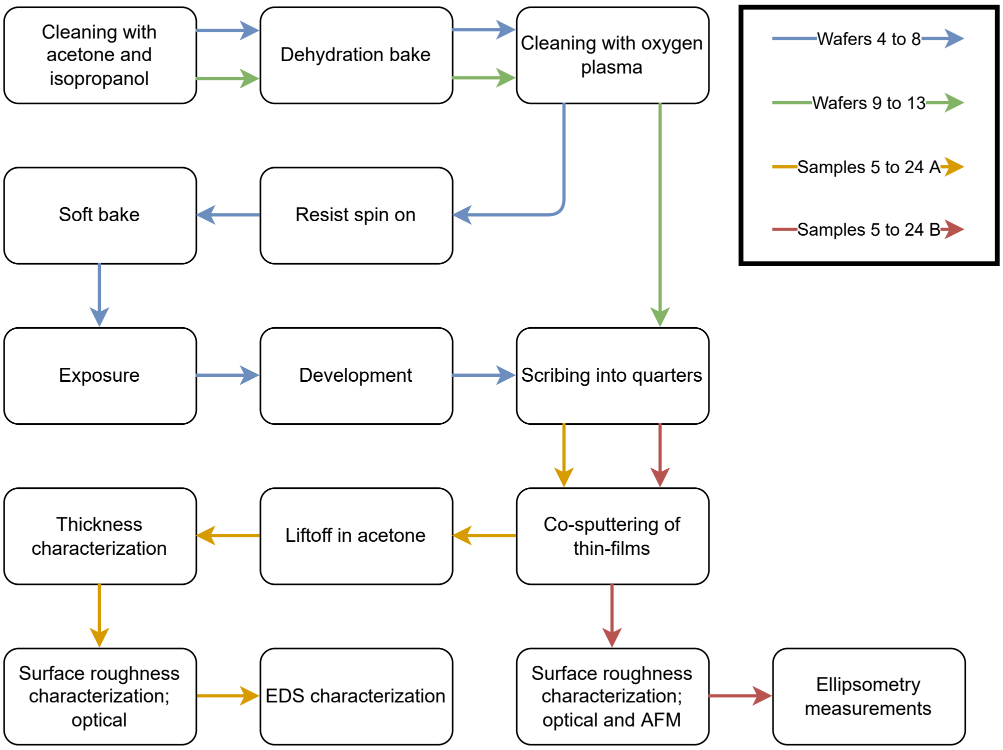

Cleaning with acetone and isopropanol
Dehydration bake
Cleaning with oxygen plasma
Wafers 4 to 8
Wafers 9 to 13
Samples 5 to 24 A
Samples 5 to 24 B
Soft bake
Resist spin on
Exposure
Development
Scribing into quarters
Thickness characterization
Liftoff in acetone
Co-sputtering of thin-films
Surface roughness characterization; optical
EDS characterization
Surface roughness characterization; optical and AFM
Ellipsometry measurements

Figure 3.1: Overview of the production and characterization pipeline of the different samples.

3.2 Cleaning and lithography

All samples were manufactured from clean unused 2” wafers bought from NanoLab. Both the cleaning and lithography were performed on whole wafers, which were later scribed into quarters to produce the samples.

The wafers were cleaned by submersion in acetone between 2 and 5 minutes until all dirt and grime was dissolved, they were then rinsed with isopropanol. The wafers then underwent a dehydration bake at 150 °C for 5 minutes on a hotplate. Afterwards, the wafers were cleaned in a Diener Electronics Femto plasma cleaner for 1 minute at 50% O_2 and 50% power [49].

Wafer 2, the wafers for sample 1 to 4 and 5 to 24 A were patterned using lithography with ma-N 440 negative photoresist and ma-D 332 developer [50], [51]. The resist was spun on at 4000 rpm for 30 seconds with an acceleration of 1000 rpm/s, followed by a soft bake at 95 °C for 5 minutes on a hotplate. The resist was exposed using a Heidelberg MLA 150 maskless aligner with a 450 nm laser and a dosage of 2500 mJ/cm^2 [52]. The exposed pattern is shown in figure 3.2 and is available online [48]. The patterns were developed by submersion in ma-D 332 between 2 and 5 minutes, until 1 minute after the silicon wafer became visible.

The development was stopped by submersion i de-ionized water, the samples were dried with nitrogen. The precursor wafers to the B series of samples did not undergo lithography.

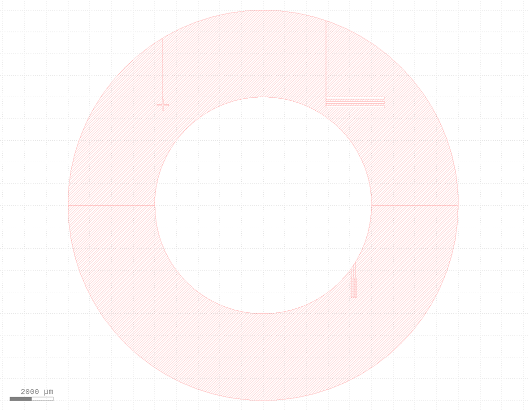


Figure 3.2: The test pattern consists of a doughnut with several details on it. The resulting thin film consists of a circular disk with a well defined step edge, as well as some surrounding patterns used to verify the accuracy of the process.

All wafers except for wafer 2 were scribed into quarters to produce samples 1 to 4 and 5 to 24 A and B, all samples were blown with nitrogen to remove any debris from the scribing process.

3.3 Film deposition

The films were deposited on the substrates using co-sputtering of gold and palladium targets in an AJA custom sputterer [53]. The samples were secured with tape on a carrier wafer in the sample holder, the A and B variant were sputtered together for samples 5 to 24. The design parameters for the film deposition is presented in table 3.1, note that the deposition rates vary for the same power from day to day. The deposition rates were tested each day before any samples were produced, the results are presented in section 4.1. The deposition rate by weight is known to be linear with the applied sputtering bias power, this was verified with the deposition rate tests. Furthermore, the deposition from the different sources in the sputtering machine is known to be independent of each other, the total deposition rate is therefore the sum of the deposition rates measured for each source. This was verified as part of the deposition rate tests. The total deposition rate and alloying concentration can therefore be controlled by changing the sputtering power of each target and deposition time, these parameters can be calculated based on one deposition rate measurement fro each target. Note that the sensor used for the deposition rate measurements is sometimes recalibrated by the NanoLab staff as part of the instrument maintenance scheduled every few days. The deposition rate measurements are therefore only valid for a short time. The deposition was carried out at room temperature and no annealing was performed on the samples. This was done to avoid dewetting and the formation of nano-particles as reported on by Gong et al. [15].

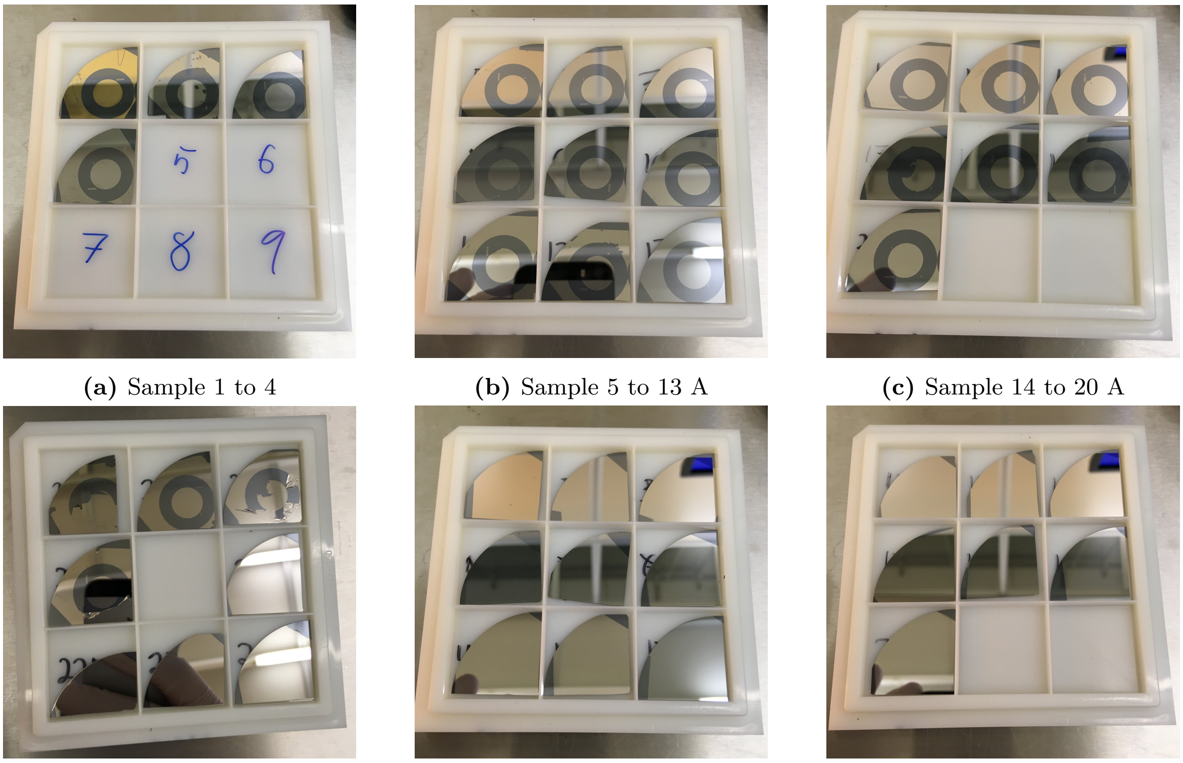
Table 3.1: Sputtering time, sputtering power and gas flow for the produced samples. All films were deposited at a pressure of 3 mTorr and the samples were spun at 30 rpm.

Sample ID	Time [s]	Au power [W]	Pd power [W]	Ar flow [sccm]
Wafer 2	35	75	400	67
Sample 1	217	100	0	Unknown
Sample 2	625	0	200	Unknown
Sample 3	91	50	400	Unknown
Sample 4	55	50	400	Unknown
Sample 5 A and B	120	93	0	67
Sample 6 A and B	120	56	202	67
Sample 7 A and B	120	47	253	67
Sample 8 A and B	18	50	404	67
Sample 9 A and B	54	50	404	67
Sample 10 A and B	90	50	404	67
Sample 11 A and B	126	50	404	67
Sample 12 A and B	162	50	404	67
Sample 13 A and B	30	50	404	67
Sample 14 A and B	60	50	404	67
Sample 15 A and B	120	50	404	67
Sample 16 A and B	180	50	404	67
Sample 17 A and B	240	50	404	67
Sample 18 A and B	90	63	337	67
Sample 19 A and B	60	93	500	67
Sample 20 A and B	120	0	505	67
Sample 21 A and B	600	125	343	67
Sample 22 A and B	600	107	457	67
Sample 23 A and B	600	71	685	67
Sample 24 A and B	649	49	739	67

3.4 Liftoff process

Wafer 2, samples 1 to 4 and 5 to 24 A underwent a liftoff process by submersion in acetone. The samples were submerged in a beaker of acetone placed in an ultrasonic bath. The submersion lasted from 2 to 10 minutes depending on how long it took for the photoresist to dissolve. The variation in liftoff time was caused by the variation in film thickness and step coverage from the sputtering process. All samples were washed with isopropanol and dried with nitrogen immediately after they were removed from the acetone bath.

The finished samples are shown in figure 3.3. The liftoff process resulted in substantial damage to sample 2, 12A, 17A, 21 and 23, as well as lesser damage to sample 11A and 16A.



(a) Sample 1 to 4 **(b)** Sample 5 to 13 A **(c)** Sample 14 to 20 A

(d) Sample 21 to 24 A and B **(e)** Sample 5 to 13 B **(f)** Sample 14 to 20 B

Figure 3.3: Images of the produced samples. Note the circular pattern som samples 1 to 4 and 5 to 24 A.

3.5 Film characterization

The film thickness of samples 5 to 24 A was measured using both a stylus and and optical profilometer [54], [55]. The results are shown in section 4.1. The stylus profilometer works by scanning a needle over the sample, tapping the sample at each point in the scan and measuring the height. It has a needle width of 12.5 µm and a force of 3 mg. The scans were taken over 900 µm with a resolution of 25 nm/sample, and the data was leveled using the Julia script in appendix E. The optical profilometer used green light phase scan interferometry (PSI) on samples 5 to 21 A and white light vertical scan interferometry (VSI) on samples 22 to 24 A. The mode of operation was changed because it proved difficult to obtain good PSI images of the large step height, and the film thickness was large enough that VSI provided sufficient resolution. The interferometer was not able to measure the thickness of samples 5, 8 and 13 A as they were too thin for the instrument. The edge of the films was visible, but the height difference between the film and the substrate was less than the noise floor of the measurements. The optical profilometer data was analyzed using Gwyddion [56].

The surface roughness of the thin films on samples 5 to 24 A and B was measured using the optical 3D profilometer in phase scan interferometry mode in a 95 by 72 µm square. Samples 5 to 24 B were also scanned by Michael Fried in a 500 by 500 nm square using an atomic force microscope (AFM) [57]. The AFM scans a probe over the samples, detecting the atomic forces acting between the probe and the sample, thereby imaging the surface. The results were analyzed using Gwyddion and are presented in section 4.2 [56].

A scanning electron microscope (SEM) was used to verify the existence of pealed off thin-film edges visible in the scanning probe measurements [58]. The SEM images was also intended to verifies the existence of the nanometer scale surface roughness as a physical phenomenon, and not an aberration caused by the AFM. The images were taken by Michael Fried, and are shown in section 4.2.

Samples 5 to 24 A were examined by Michael Fried using the energy dispersive spectroscopy (EDS) sensor in the SEM to quantify the atomic makeup of the samples, the results are shown in section 4.3. The EDS works by detecting the back-scattered x-ray radiation generated by the electron beam hitting the samples, and analyzing the measured energy spectra.

3.6 Ellipsometry

Samples 5B to 24B except for 13B were scanned and fitted by Michael Fried and Nathan Hale using ellipsometry at the physics department at NTNU to determine the refractive index of the alloys as a function of wavelength. A J.A. Woolam RC2+NRI ellipsometer was used in the visible and near-IR spectrum from $\lambda = 210\,\mathrm{nm}$ to $\lambda = 1690\,\mathrm{nm}$, the samples were interrogated at three distinct angles [59]. Samples measured to have similar alloying ratios and a layer thickness between 20 nm and 100 nm were assumed to have optical characteristics similar to bulk metals, and were modeled using multi-sample analysis. The lower limit of 20 nm was chosen based on the thin-film effects observed by Rosenblatt et al., the upper limit of 100 nm was chosen because the thicker samples, sample 21 to 24, yielded significantly different results than the thinner samples [37]. All samples were fitted using either a Drude model or a Kramers-Kronig-consistent B-spline model as explained in section 2.6.

Chapter 4

Thin-film results and discussion

4.1 Film thickness measurements

The results from the thickness measurements described in section 3.5 are shown in table 4.1, the data is available online [48].

Table 4.1: Film thickness as intended based on rate measurements and as measured using stylus and optical interferometry.

Sample ID	Intended [nm]	Stylus [nm]	Optical [nm]
5A	50	15	Unmeasurable
6A	50	20	15
7A	50	20	20
8A	10	15	Unmeasurable
9A	30	25	7
10A	50	20	15
11A	70	20	25
12A	90	40	35
13A	25	20	Unmeasurable
14A	50	25	8
15A	100	30	25
16A	150	40	45
17A	200	60	70
18A	50	35	15
19A	50	40	10
20A	50	30	20
21A	250	320	260
22A	250	300	275
23A	250	250	235
24A	250	225	220

The samples were only measured once with each technique, and no confidence interval is given for the measurements. The stylus profilometer used is known to be accurate to a few nanometers, and the roughness exhibited in the collected stylus data indicate an uncertainty on the order of ±5 nm. The instrument was calibrated according to VLSI standard 4343-26-24 in 2012, and was found to have a precision of 0.26 nm [60]. It is however worth noting that this calibration was performed 10 years prior to the measurements reported in this thesis. The measurements may contain systematic errors caused by the needle scratching the thin-films or unintended dust or acetone being deposited on top of the samples. Similarly, the optical profilometer measurements may contain systematic errors caused by the thin-films letting some light pass through, allowing light to reflect on the thin-film wafer interface. The roughness exhibited in the collected optical data indicate an uncertainty on the order of ±3 nm. The instrument manual claims a vertical resolution of less than 0.1 nm, it does however also claim a max slope on 26.7° on shiny surface and 81° on rough surfaces, both of which are less than the step edge which was fabricated [61]. The different thickness measurements should therefore be understood as a best estimate, suitable for ellipsometric fitting, and not as the exact thickness.

The film thickness measurements show a large discrepancy between the intended and measured thickness as well as a smaller difference between the two measurement methods. This is shown in figure 4.1 where the measured thickness is shown as a function of the intended film-thickness, and in figure 4.2 where the optical and stylus measurements are plotted against each other. The deposition rate monitor in the AJA sputterer used to calculate the intended thickness was, as explained in section 3.3, not calibrated per material. This means that the intended thickness deposition rate is a number proportional to the real weight deposition rate with an unknown proportionality coefficient. This coefficient can only be assumed to be self-consistent while no other lab users have booked the instrument, and is therefore not suitable for film-thickness measurements. This is confirmed by the deposition rate measurements shown in table 4.2, where the deposition rates at the same power vary significantly from day to day. The deposition rate sensor is a quartz balance, and it measures the accumulated weight deposited on the sensor through its lifetime. It can be calibrated to specific materials if the weight to film thickness ratio is known, the sensor must also be recalibrated every few weeks since the buildup of deposited material on the quartz crystal will interfere with the weight measurements. It is therefore reasonable to assume that its thickness measurements were scaled by an unknown coefficient related to the weight to volume ratio of the material. This coefficient may be dependent on both material parameters such as the specific alloy being deposited and the film thickness, as well as how the sensor is calibrated. It follows that the intended thickness reported in table 4.1 calculated based on the deposition rate measurements is the least trustworthy source for the film thickness, and the intended thickness was not achieved.

The optical profilometer was not able to measure the film thickness of samples 5, 8 and 13 A. The measurements show a sharp line at the edge of the film, but no difference is detectable between the film and the substrate. Samples 8 and 13 were the thinnest samples in their respective series, 8 to 12 and 13 to 17. It therefore stands to reason that these two samples were too thin to be measured optically. It is unclear if the lack of optical measurements of sample 5 is caused by the same effect, or because gold is almost transparent to the green light used by the instrument, making the film appear invisible to the instrument.

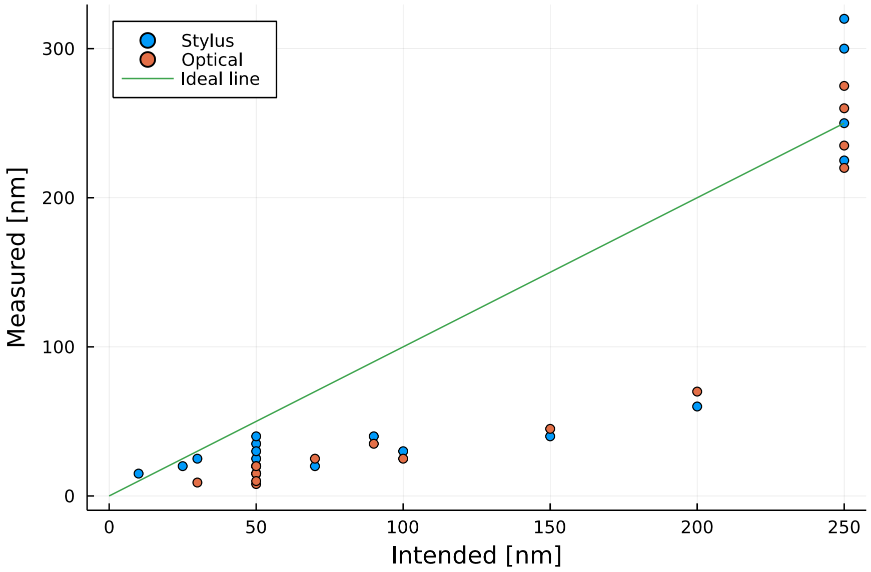


Figure 4.1: Measured thin-film thickness using stylus profilometer and optical interferometer scanning plotted as a function of intended film thickness. The data is plotted together with the ideal line representing perfect control over intended film thickness.

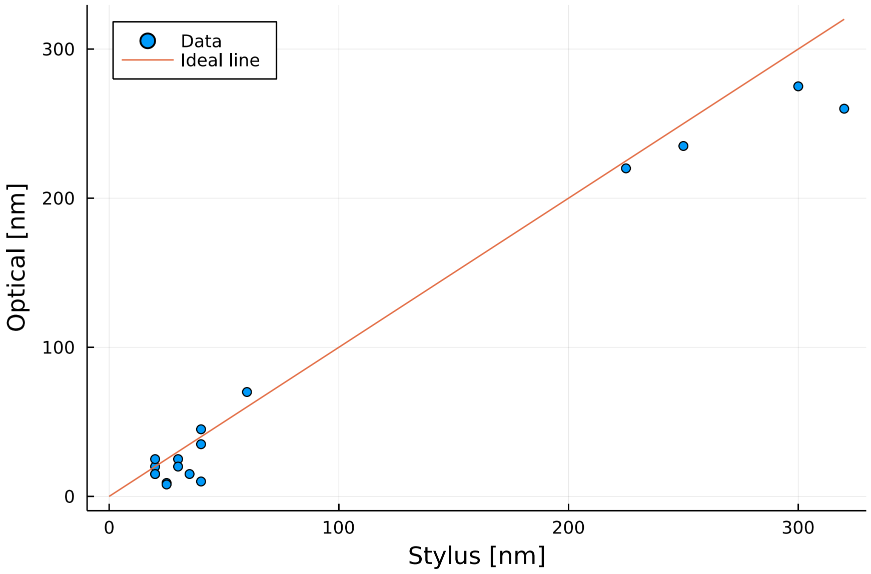


Figure 4.2: Measured thin-film thickness using optical interferometer scanning plotted as a function of measured thin-film thickness using stylus profilometer. The data is plotted together with the ideal line representing identical measurements.

Table 4.2: Deposition rate tests performed on the AJA sputterer during the lab work. The rate is given as measured by the quartz balance sensor in the machine, and is assumed to be proportional to the weight deposition rate.

Date	Deposition	Measured rate [Å/s]
25. March	75 W Au	6.8
	200 W Au	18.9
	200 W Pd	3.5
	400 W Pd	6.8
	75 W Au and 400 W Pd	13.9
30. March	50 W Au	2.3
	100 W Au	4.6
	200 W Au	8.8
	50 W Pd	0.4
	100 W Pd	0.8
	200 W Pd	1.6
3. April	75 W Au	3.5
	150 W Au	6.8
	100 W Pd	0.8
	200 W Pd	1.7
4. April	150 W Au	6.8
	100 W Pd	0.8
5. May	75 W Au	1.8
	150 W Au	3.4
	100 W Pd	0.4
	200 W Pd	0.7

Based on a review article by Lin et al., it can be assumed that the stylus profilometer gives the most correct measurements [62]. The observed difference from the optical profilometer can therefore be assumed to be caused by aberrations resulting from reflections at the sample-substrate interface. The thickness measured by the stylus profilometer was therefore used both for all further discussion and for the ellipsometric fitting described in section 3.6 and reported in section 4.4.

These measurements have shown that, when producing thin-films using co-sputtering, a test film should be produced and its thickness measured shortly before each production batch in order to calibrate the deposition rate monitor measurements to the real deposited thickness. This should be done at most a couple of days before the thin-film production process which needs the thickness accuracy, and it must be assured that no maintenance takes place between the calibration and the production.

4.2 Film roughness measurements

The thin-film surface roughness measurements described in section 3.5 are shown in table 4.3, the data is available online [48].

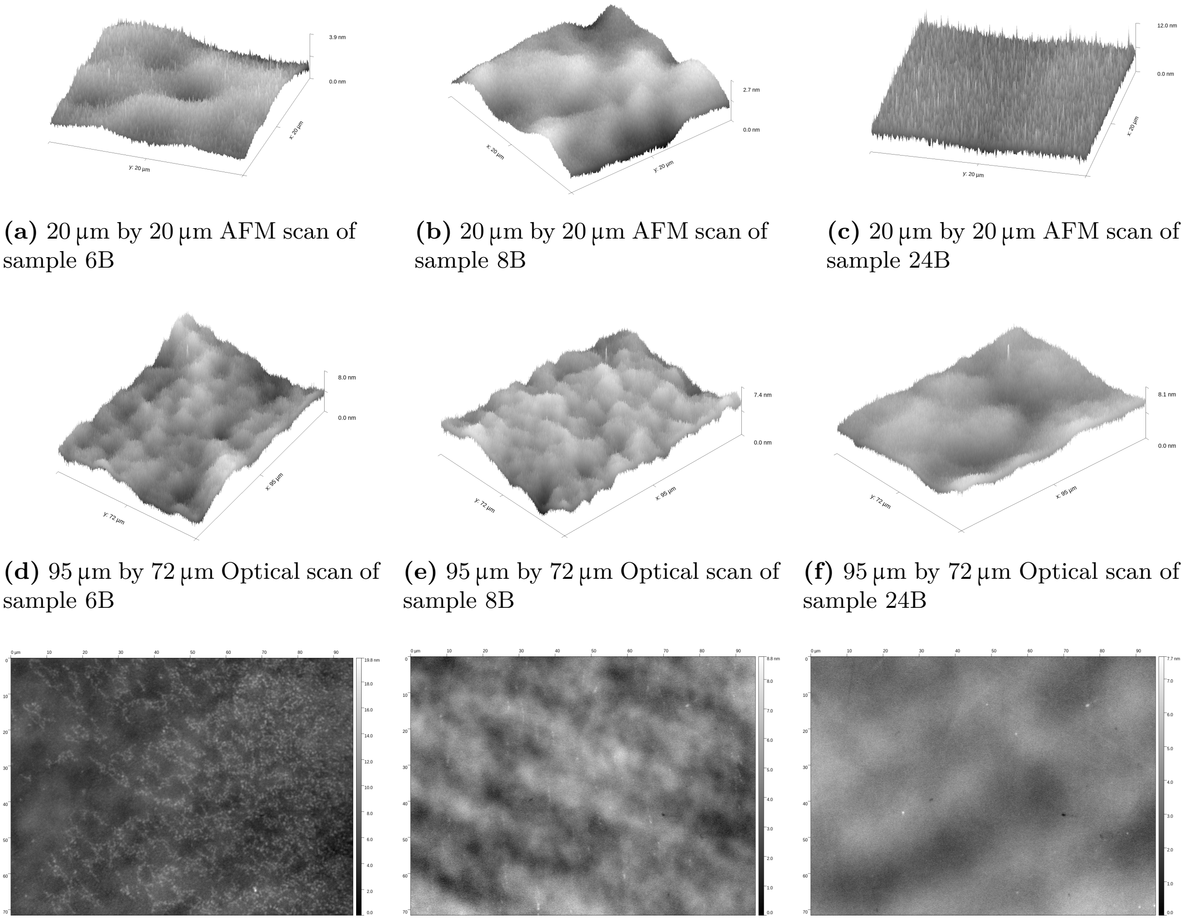
Table 4.3: Film root mean square (RMS) roughness measurements for sample 5 to 24 A and B. Series A and B were measured using the optical profilometer, series B was also measured using an AFM. The correlation length of the B series was extracted from the optical roughness measurements using the extrapolated ACF decay to 1/*e* algorithm in Gwyddion [56].

Sample ID	Optical A [pm]	Optical B [pm]	Correlation B [μm]	AFM B [pm]
5	1057	872.6	9.03	581.6
6	1322	955.5	7.52	584.8
7	1223	917.9	7.70	427.1
8	1000	800.7	4.98	103.6
9	1125	933.1	7.11	129.6
10	1413	955.8	6.41	344.2
11	1067	1028.0	4.73	407.3
12	978	1017.0	10.55	585.2
13	931	897.9	10.00	186.3
14	1172	1114.0	1.29	236.8
15	1095	863.4	13.10	440.8
16	862	940.2	7.60	628.5
17	1184	890.2	9.30	826.0
18	935	1123.0	10.30	297.5
19	935	916.1	8.90	287.2
20	1164	1093.0	14.10	247.7
21	955	933.3	9.00	2126.0
22	770	801.0	11.00	1979.0
23	1414	724.8	12.10	1972.0
24	702	786.9	13.20	2000.0

The samples were only measured once with each technique, and no confidence interval is given for the measurements. The optical profilometer manual claims a vertical resolution of 0.1 nm and a spatial sampling resolution of 60 nm [61]. It can therefore be assumed that it can measure the roughness reported in table 4.3 column 2 and 3, with grain sized as indicated by column 4. The AFM's manual claims the instrument introduces an RMS noise of less than 0.03 nm in the vertical direction, it is therefore reasonable to believe the RMS roughness reported in table 4.3 column 5 [63]. The process of calculating the RMS surface roughness includes a significant amount of post-processing of the measurements, and the uncertainty estimates from the instruments used have therefore not been propagated. As discussed below, each type of roughness was confirmed using multiple independent measuring techniques. This enables the measurements reported to be used as a best estimate, suitable for ellipsometric fitting.

The surface roughness measured using optical phase interferometry and atomic force microscopy show two different sets of results. Figure 4.3 shows a selection of surface scans done for sample 6, 8 and 24 A and B. Note that the AFM scans presented in figure 4.3a, 4.3b and 4.3c were performed on an area of 20 by 20 µm, and are not the same scans used for table 4.3. The optical scans were performed on an area of 95 by 72 µm, the optical scans shown in figure 4.3 are the same as those reported for the relevant samples in table 4.3. The optical scans of samples 6, 8 and 24 A shown in figure 4.3g, 4.3h and 4.3i are shown in 2D to better highlight the damages to the thin-films caused by the liftoff shown as white dots and smears on the scans. This is apparent in table 4.3 column 2 and 3, where the optical RMS roughness measured for the A series is generally larger than that measured for the same thin-films in the B series. Figure 4.3 a to c and d to f shows that both the AFM and the optical profilometer detects a large wavy roughness, but that the AFM also detects a finer roughness that is largely absent from the optical measurements. The correlation length for the optical roughness measurements shown in table 4.3 confirms that the grain sizes in the large-scale roughness is significantly bigger than the roughness shown in the AFM images. Due to misalignment along the fast axis of the AFM measurements, the correlation length can not be calculated in a comparable way for the AFM scans, since aligning the scan lines would include optimizing the alignment for maximal smoothness. It is however clear from the measurements that the grain sizes in the fine-grained roughness is on the nanometer scale. The AFM-scans whose results are presented in table 4.3 were taken over an area of 500 nm by 500 nm, an area smaller than the grain size of the large scale waviness as indicated by table 4.3 column 4 and shown in figure 4.3. It can therefore be assumed that the AFM measurements measure the fine grained roughness, while the optical measurements measure the large wavy roughness.

Both table 4.3 and figures 4.3g, 4.3h and 4.3i shows that the lithography and liftoff process performed on samples 5 to 24 A has a significant chance of increasing both the large scale and the fine grained sub-wavelength surface roughness. As shown in chapter 6, the fine grained roughness should be reduced when constructing a plasmon based sensor to achieve a strong plasmon signal. Any sensor element constructed using the procedure described in chapter 3 should therefore be made in a way that avoids any liftoff or etching steps after the active thin-films are deposited. It is possible that adding a temporary protective layer on top of the active layer may protect against the liftoff induced damages, but that leaves the problem of removing the protective layer.



(a) 20 µm by 20 µm AFM scan of sample 6B

(b) 20 µm by 20 µm AFM scan of sample 8B

(c) 20 µm by 20 µm AFM scan of sample 24B

(d) 95 µm by 72 µm Optical scan of sample 6B

(e) 95 µm by 72 µm Optical scan of sample 8B

(f) 95 µm by 72 µm Optical scan of sample 24B

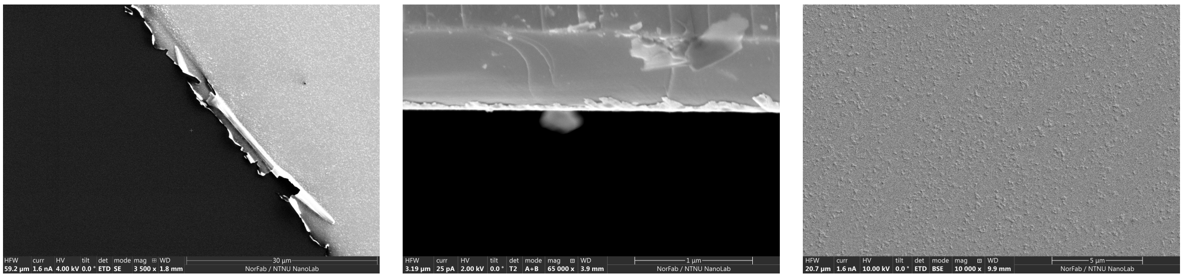
(g) 95 µm by 72 µm Optical scan of sample 6A

(h) 95 µm by 72 µm Optical scan of sample 8A

(i) 95 µm by 72 µm Optical scan of sample 24A

Figure 4.3: Surface roughness scans of samples 6, 8 and 24 A and B using atomic force microscopy and optical phase interferometry. The z axis shows the height above the lowest point in each image.

Several SEM images were taken by Michael Fried as described in section 3.5, a selection of which is presented in figure 4.4. Figure 4.4a and 4.4b verify the peeling of the thin-films caused by the liftoff process, and further confirms that the liftoff damages the thin-films. The existence of the fine grained roughness is confirmed by SEM imaging as shown in figure 4.4c, it can therefore not be considered an artifact of the AFM.


HFW curr HV tilt det mode mag WD
59.2 µm 1.6 nA 4.00 kV 0.0 ° ETD SE 3 500 x 1.8 mm
30 µm
NorFab / NTNU NanoLab
HFW curr HV tilt det mode mag WD
3.19 µm 25 pA 2.00 kV 0.0 ° T2 A+B 65 000 x 3.9 mm
1 µm
NorFab / NTNU NanoLab
HFW curr HV tilt det mode mag WD
20.7 µm 1.6 nA 10.00 kV 0.0 ° ETD BSE 10 000 x 9.9 mm
5 µm
NorFab / NTNU NanoLab

(a) Partially pealed off thin-film edge on wafer 2.

(b) Partially pealed off thin-film edge on sample 5A.

(c) Surface roughness of sample 20 A.

Figure 4.4: SEM images of wafer 2, sample 5A and sample 20A showing thin-film pealing after the liftoff process and surface roughness on a nanometer scale. The gray scale shows detected electrons by the SEM. Images by Michael Fried.

The fine grained surface roughness measured with the AFM shows a clear increasing trend with increasing film thickness. The data is shown in figure 4.5. It is clear from the plot that, though there is a clear positive trend, it is more complex than a simple linear fit. The large scale wavy surface roughness measured with PSI is shown together with measured correlation length of the surface in figure 4.6. The large scale roughness does not show a clear relation to any of the varied parameters of alloying ratio and film thickness. Figure 4.7 shows the AFM and optical RMS roughness measurements plotted against each other. It is clear from the figure that, when samples 5 to 20 and 21 to 24 are considered independently, the two groups show no clear trend in the relationship between the measurement methods. Since the difference in measured roughness between the two groups using the AFM is related to the film thickness as shown in figure 4.5, it is reasonable to assume that the optical and AFM roughness measurements are independent of each other.

As discussed in section 2.4, the effective medium approximation used to model the thin-film air interface assumes that the individual grains are significantly smaller than the optical wavelength. This approach is used both for modeling the sensor as described in chapter 6 and for fitting the ellipsometry results as described in section 3.6. The optical wavelength λ is in the near IR range in both cases, in the order of 0.5 to 2 µm. Based on this, the roughness can be modeled in two disjunct steps. The total roughness can be considered as the superposition of the large wavy roughness and the fine grained roughness. The wavy roughness has, with the possible exception of sample 14, larger grain sizes than the optical wavelengths in the near-IR used by the sensors. It can therefore be assumed that the large-scale roughness will not affect the scattering of light significantly, and can be modeled as an uncertainty in the thin-film thickness. Since the real refractive index of palladium is larger than one at the relevant wavelengths as shown in figure 2.5c, the optical wavelength in palladium will always be shorter than in air and this assumption holds. The fine grained roughness on the other hand have grain sizes in the nanometer range, and will affect the scattering of light

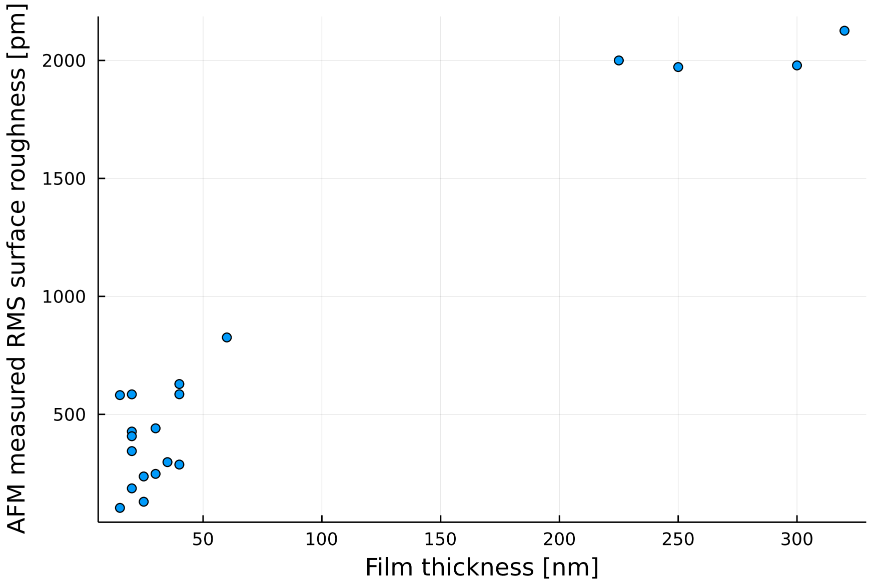


Figure 4.5: Measured fine grained surface roughness plotted as a function of measured film thickness.

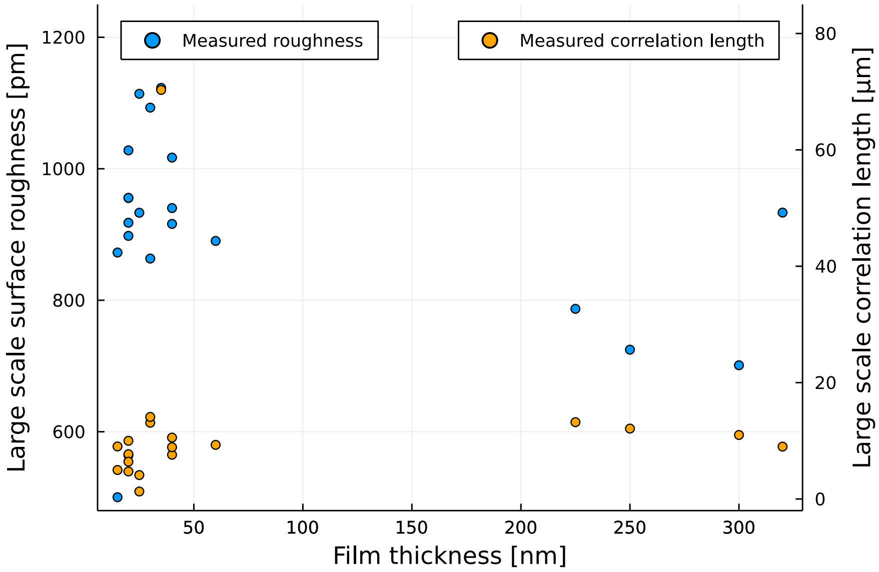


Figure 4.6: Measured large scale surface roughness and correlation length plotted as a function of measured film thickness.

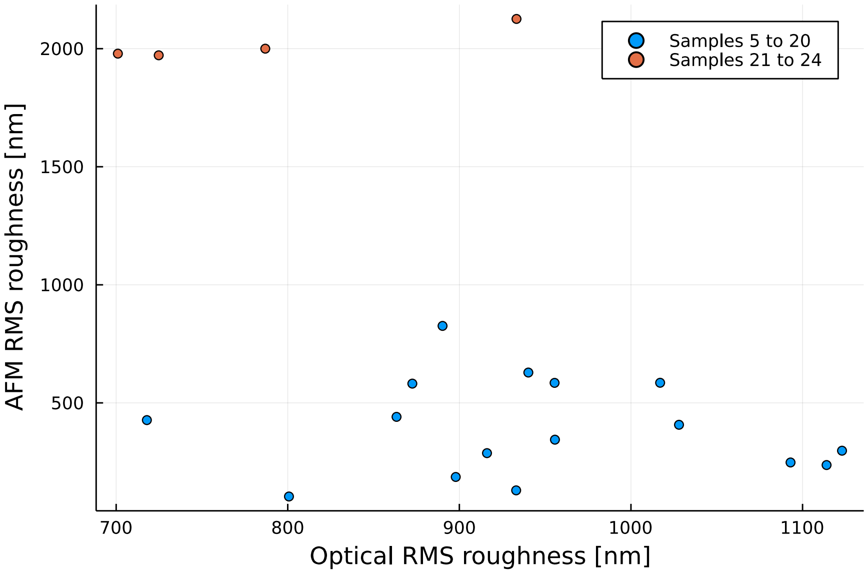


Figure 4.7: Measured roughness with AFM plotted against measured roughness with optical PSI. Note that samples 21 to 24 exhibited an high RMS roughness when measured with the AFM, independently of the optically measured roughness.

differently. As discussed in section 2.4, this roughness can be modeled using an effective media approximation with a mix of the thin-film and air and a thickness proportional to the RMS roughness measured with the AFM.

4.3 Film composition measurements

The results from the EDS measurements described in section 3.5 are shown in table 4.4. The sensor detected substantial amounts of gold, palladium, silicon, vanadium, carbon and oxygen. The silicon was attributed to the wafer on which the thin-films were grown, the carbon, vanadium and oxygen to the sample holder and chamber walls of the SEM which are made from stainless steel. These assumptions were verified by an EDS measurement of a clean wafer. These contributions were therefore removed, and the reported results show the ratios between gold and palladium.

The EDS results verify that the co-sputtering process offers control over the produced alloying ratios. When taking the EDS measurements, the instrument scans the electron beam over the sample, similar to when a SEM image is taken. This results in the spatial distribution of detected atoms being visible when the measurements are performed. It was clear that the samples had an even distribution of gold and palladium down to the nanometer scale. This shows that the grains of the large scale roughness described in section 4.2 consist of alloyed metal, not grains of palladium or gold exclusively. Figure 4.8 shows the relation between intended and measured alloying ratios. It is clear that the thicker films, sample 21 to 24, deviate much more than the other films from the intended alloying ratios. This is further shown in figure 4.9, where both the difference and ratio between intended and measured alloying are shown. It is clear from the figure that there is at tendency for thicker films to

Table 4.4: Percentage of palladium in the gold palladium alloys by weight and atomic count as measured by the sputterer deposition rate and the EDS measurements.

Sample ID	Sputterer weight [%]	EDS weight [%]	EDS atomic [%]
5A	0	0.00	0.00
6A	40	37.76	52.90
7A	50	49.47	64.44
8A	60	60.05	73.57
9A	60	59.26	72.92
10A	60	58.92	72.50
11A	60	58.57	72.35
12A	60	57.97	71.86
13A	60	59.36	73.00
14A	60	58.84	72.57
15A	60	57.14	71.17
16A	60	56.14	70.33
17A	60	55.37	69.67
18A	50	49.05	64.06
19A	50	45.92	61.11
20A	100	100.00	100.00
21A	30	27.66	41.46
22A	40	35.58	50.55
23A	60	45.52	60.73
24A	70	55.01	69.32

have larger deviations in the alloying, this trend is clear both with and without the thickest samples. Figure 4.10 shows the difference and ratio between intended and measured alloying as a function of the intended alloying ratio. The figure shows no clear correlation between the alloying ratio and the deviation. It is not clear from the measurements if the discrepancy is caused by an unknown effect during the thin-film production, by a weakness in the EDS system or by operator inexperience when using either the EDS or the sputtering machine. Until a verifiable correlation between production parameters and the alloying deviation is found, all new co-sputtering configurations should be verified with EDS measurements whenever the exact alloying ratio is critical to the device being produced, and both the sputtering configuration and the EDS measurements should be considered when analyzing experimental data.

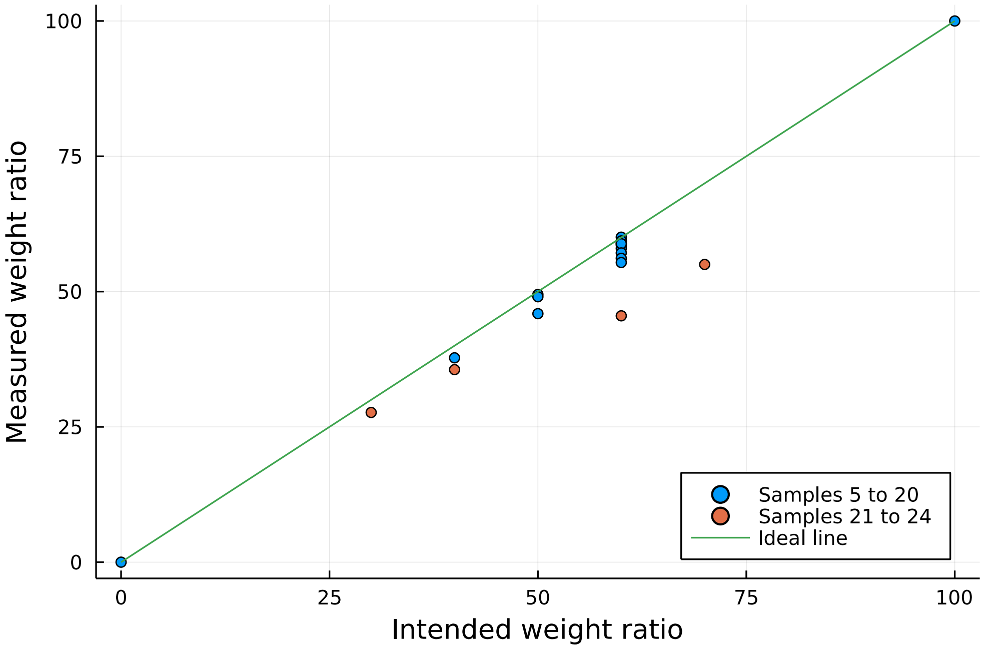


Figure 4.8: Measured alloying ratio using EDS plotted against intended alloying ratio. The ideal line is shown in green.

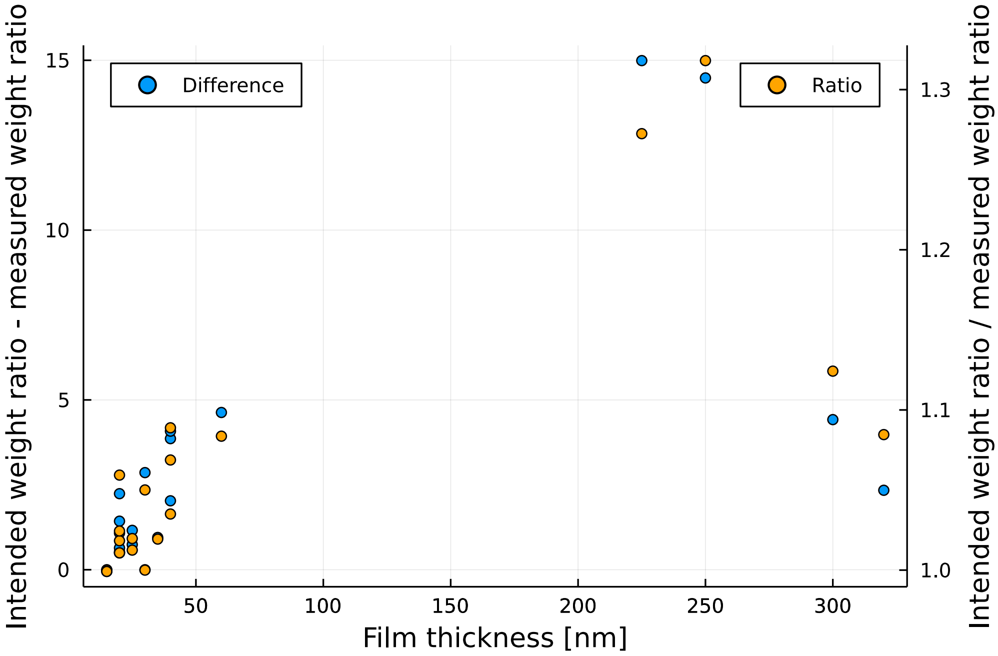


Figure 4.9: Difference and ratio between intended and measured thin film alloying ratio using EDS by palladium weight percentage as a function of measured film thickness.

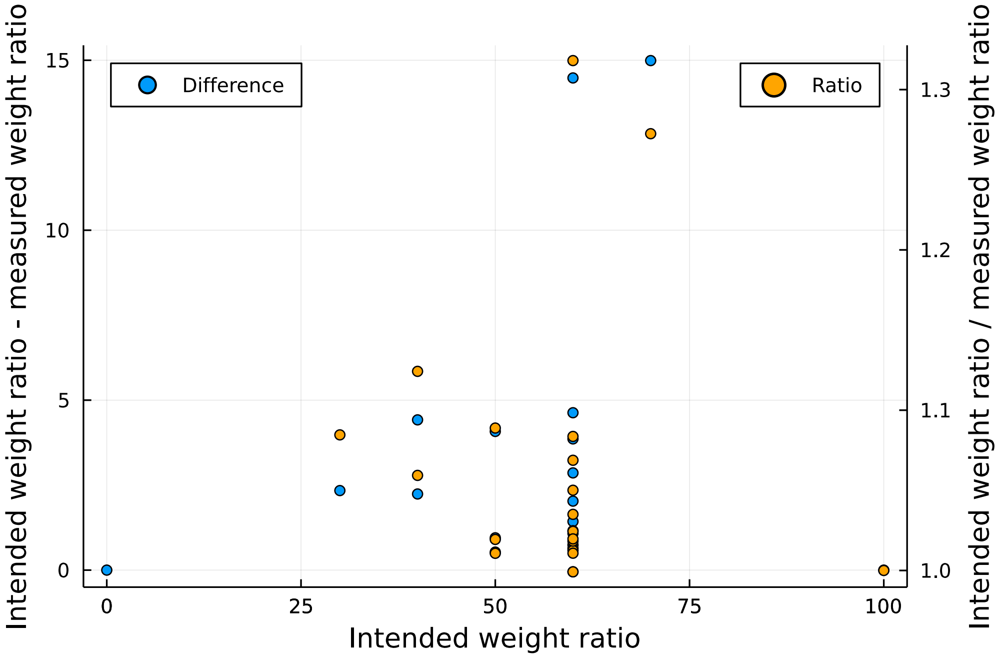


Figure 4.10: Difference and ratio between intended and measured thin film alloying ratio using EDS by palladium weight percentage as a function of intended alloying ratio.

4.4 Ellipsometry measurements

The ellipsometric data was gathered and fitted as described in section 3.6. The results are presented in figure 4.11, the data is available online as part of the simulator package described in chapter 5 [30]. The results were fitted using the stylus measurements for thickness and the AFM measurements for roughness with the uncertainties described in sections 4.1 and 4.2. The only samples fitted using a Drude model were samples 8 and 19. For sample 8, this was done because a B-spline model yielded a poor fit the the experimental data and for sample 19 it was done to show both that a Drude and a B-spline fitting can yield similar results based on the same ellipsometric measurements. The other measured samples were fitted using a B-spline model with either single or multi-sample fitting. An overview of the samples and fitting models is shown in table 4.5. The refractive index data reported originates from different fittings, and are not dependent on other samples than those fitted together using multi-sample fitting. Note that sample 13 was not fitted, since the ellipsometry measurements could not be fitted properly. Note that the ellipsometric measurements from sample 13 was not possible to fit to a refractive index model, and is therefore not reported. This, together with the fact that the film thickness could not be measured optically as discussed in section 4.1, indicates that the sample is either too thin or too transparent to be properly measured using optical techniques such as phase scan interferometry or ellipsometry. The fitting algorithms used by the ellipsometer are closed and proprietary, it is therefore not possible to propagate the uncertainties originating in the ellipsometer or the thickness and roughness ranges passed to the ellipsometry software.

Table 4.5: Overview of samples and fitting of ellipsometric data. The multi-sample 1 and 2 fitting groups are both multi-sample fitting to a B-spline model, but were fitted independently of each other.

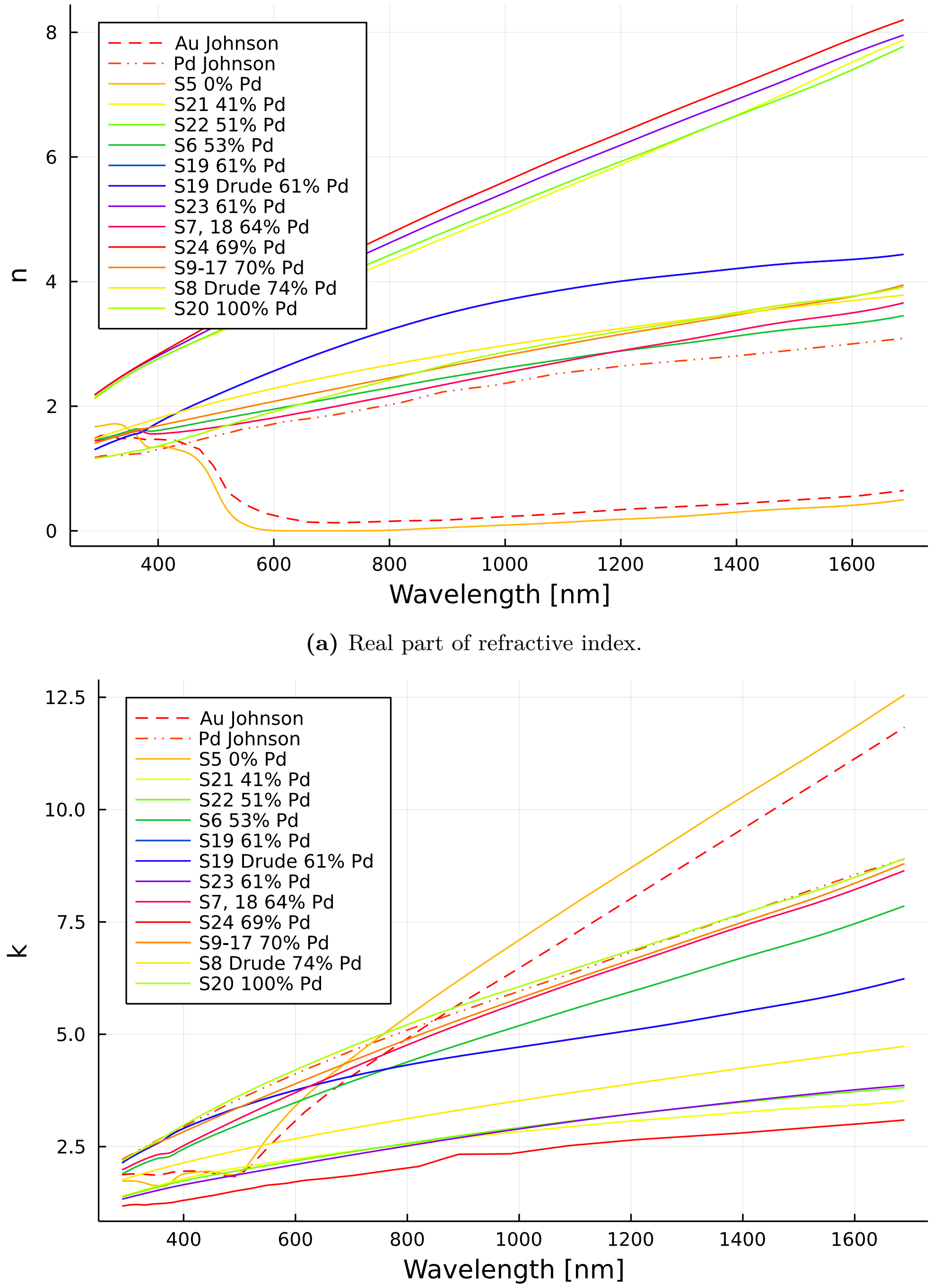
Sample Fitting	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
Multi-sample 1					X	X	X	X		X	X	X	X							
Multi-sample 2			X											X						
Single B-spline	X	X													X	X	X	X	X	X
Single Drude				X											X					

The ellipsometry data shows a large difference between samples 6 to 20 and 21 to 24, where the real part of the refractive index shown in figure 4.11a is up to twice as large for the thicker samples (21 to 24). The difference between the real refractive index of the two groups is significantly larger than the difference between different alloys within the same group. This implies that the difference between the two groups can not be explained by uncertainties in the ellipsometric measurements, and is instead caused by a physical difference between the samples The difference between the two groups is not as significant for the extinction coefficient shown in figure 4.11b, where the differences between individual samples are larger.

Selected ellipsometry results are plotted together with measurements of similar alloyed thin-films reported on by Palm et al. in figure 4.12 [33]. The results show a good correlation between the pure gold film on sample 5 and the experimental data from Johnson et al. [31]. This is as expected based on the theory presented in section 2.2, and indicates that the ellipsometry results are valid. The measurement for the pure palladium film on sample 20 is similar to palladium measurements in the literature, the measurements are within the expected range based on the data gathered in figure 2.5c and 2.5d.

With the exception of samples 21 to 22, all plotted samples show a real refractive index in the same range as those reported in the literature with similar atomic alloying ratios. The imaginary part of the refractive index vary significantly, but the difference is not unexpected when considering the difference between sources on the complex refractive index of palladium shown in figure 2.5d. It is reasonable to assume that the difference between sample 21 at 41.46% and the reported index of 42% palladium alloy, as well as between sample 22 at 50.55% and the reported index of 52% palladium alloy, is caused by film thickness-dependent effects, and not the alloying ratios. This is due to the differences in roughness and alloying ratio deviation between samples 6 to 20 and 21 to 24 discussed in sections 4.2 and 4.3.

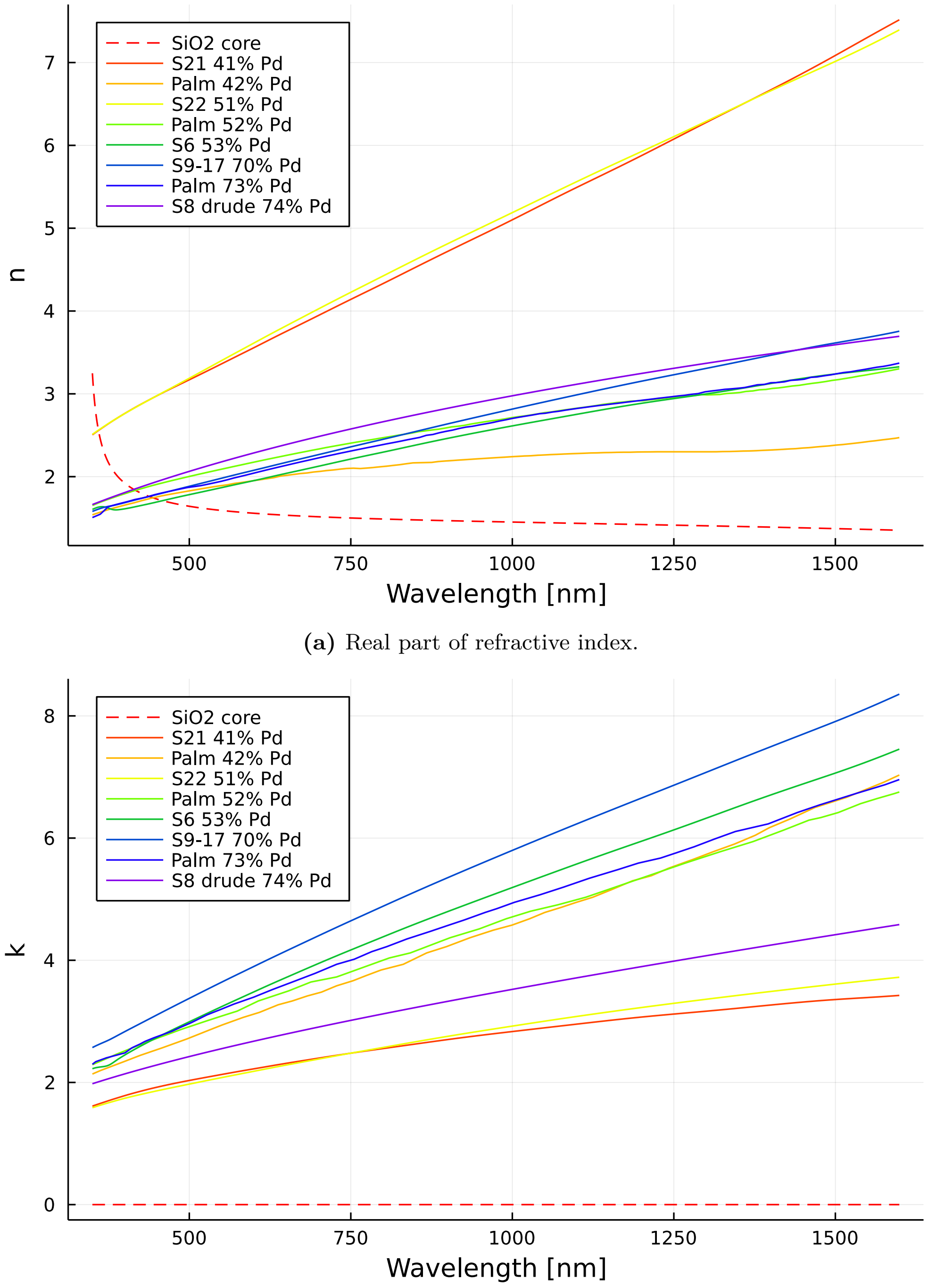
Lastly, note that all measured samples have a real refractive index higher than silica for all wavelengths in the near IR. This makes total internal reflection of light coming from an infinite half-plane of silica impossible on an infinite half-plane of any of the palladium gold alloys measured. It is therefore challenging to couple light directly from silica to a surface plasmon in a gold palladium alloy thin-film, and an intermediary layer with a lower refractive index might aid the plasmon coupling. This has been investigated in section 6.1.



(a) Real part of refractive index.

(b) Imaginary part of refractive index.

Figure 4.11: Complex refractive index spectra for the produced samples as measured with ellipsometry, S refers to the sample number in the legends. The data is plotted together with the refractive index spectra for gold and palladium as measured by Johnson et al. [31], [38].



(a) Real part of refractive index.

(b) Imaginary part of refractive index.

Figure 4.12: Complex refractive index spectra for the produced samples as measured with ellipsometry compared with palladium gold alloy data measured by Palm et al., S refers to sample number in the legends [33]. The data is plotted together with an analytical model for silica [14].

Chapter 5

Software model

The optical matrix simulator for layered thin films first presented in the specialization project preceding this thesis has been expanded upon [22]. The three main areas of improvement have been the creation of a unified parameter scan system, an expansion of the included material library and a thorough toolbox for modeling materials through an effective media approximation. The complete source code for the simulator is available both on GitHub and in appendix C, it is made available under the MIT license [30]. All changes and additions to the simulator have been documented in appendix B.

5.1 Intended use of the model

The software model is intended to be used with simple one-off simulation scripts. These scripts should, in most cases, be written in Julia, but other languages can be used with a proper foreign function interface [29]. It is recommended to only import the sub-modules of the *simulator* module that will be used by the simulator, thereby saving precompilation time.

An example of a simulation script is provided in appendix D listing D.1. The script includes several relevant parts of the *simulator* module, before defining control parameters for the simulation. In the main part of the script, several test systems are set up based on the control parameters, and the results are plotted. Note that no system or scan functions are defined in the script, all computation-heavy functions are instead imported from the *targetfigures* module. This is done in order to utilize the Julia precompiler to its fullest, and results in a drastic performance increase in the simulator. If other system or scan functions are needed, it is recommended to either expand the core modules or write new libraries instead of writing one-off functions in the simulation scripts.

5.2 Unified parameter scan system

The unified parameter scan system is shown in appendix C listing C.8. It provides a framework for implementing efficient scans over one or two design parameters for a layered structure. The parameter scan system generates an abstract description of the system in a functional programming paradigm, which is then evaluated for each element in a parameter list [64]. The specific system is then analyzed by one or more figure of merit functions, the return values of which make up the parameter scan results. Care was taken to write the parameter scan system with single pass algorithms, this ensures a run time of $\Theta(p)$ for single parameter scans and $\Theta(p^2)$ for dual parameter scans, where p is the length of the design parameter list and $\Theta(f(p))$ denotes that the runtime of an algorithm will have an upper and lower bound proportional to a function $f(p)$ for sufficiently large values of p [65].

5.2.1 Implemented figures of merit

The different figures of merit extract data from a reflection spectrum similar to what is shown in section 2.1 figure 2.3.

The **plasmon_minima** function implements an algorithm aimed at finding the minimum reflection coefficient that characterizes a plasmon in the reflection spectra. It does this by performing a single pass scan of the reflection spectra for a system, searching for the minimum. This is done instead of a numerical search in order to work with arbitrary spectra, also discontinuous ones. It features a *predipps* parameter, indicating the number of candidates for global minimum the algorithm will discard when searching for the plasmon. This feature is intended to help searching for plasmon dips in reflection coefficient spectra where the reflection coefficient of the structure for sufficiently short wavelengths is lower than the plasmon dip minimum.

The **plasmon_halfwidth** function implements an algorithm to find the plasmon dip FWHM, as well as the plasmon dip peak to peak. It is a single pass algorithm, and features a *predipps* parameter similar to the **plasmon_minima** function. The algorithm defines the dip as the region from the local maximum on the short wavelength side of the plasmon dip to the point on the long wavelength side with the same reflection coefficient as the short wavelength side maximum.

5.3 Physical materials

Several materials from the literature are included in the simulator. They can be found in the *specifics* module in the larger *materials* module, and provides a baseline for simulations using gold, palladium and silica. All included materials are described in tables 5.1 and 5.2, they are functions on the form $f : \lambda \rightarrow n$. Both silica models as well as the silver and lithium fluoride materials were unchanged from the specialization project proceeding this thesis, the **Au_Johnson** and **Pd_Werner** materials were renamed from **Au** and **Pd**. All other material functions were collected or produced as part of this project.

5.4 Toolbox for modeling materials through an effective media approximation

A toolbox for working with the effective media models described in section 2.4 was developed. The system is intended for easy use when modeling the surface roughness of thin-films, but the software is valid for other purposes. The toolbox extends the Maxwell Garnet function written as part of the preceding specialization project to be able to solve for media consisting of an arbitrary number of components. It also includes a function to calculate the Bruggeman EMA for two and three components, as well as an implementation of the algorithm by Jansson and Arwin mentioned in section 2.4 to select the correct result.

Table 5.1: Materials from the literature included in the simulator

Name	Material	Source
Au_Johnson	Gold	Johnson and Christy [31]
Au_Werner	Gold	Werner et al. [32]
Au_11nm_Rosenblatt	Gold	Rosenblatt et al. [37]
Au_21nm_Rosenblatt	Gold	Rosenblatt et al. [37]
Au_44nm_Rosenblatt	Gold	Rosenblatt et al. [37]
Au_Babar	Gold	Babar and Weaver [35]
Au_McPeak	Gold	McPeak et al. [34]
Au_OlmonEvaporated	Gold	Olmon et al. [36]
Au_OlmonSingleChrystaline	Gold	Olmon et al. [36]
Au_OlmonTemplateStripped	Gold	Olmon et al. [36]
Pd_Johnson	Palladium	Johnson and Christy [38]
Pd_Werner	Palladium	Werner et al. [32]
Pd_Palm_2018	Palladium	Kevin J. Palm et al. [39]
Ag	Silver	Johnson and Christy [31]
LiF	Lithium fluoride	H. H. Li [66]
SiO2_core_Sellmeier	Silica	Downes and Taylor [14]
SiO2_thinfilm_Ciprian	Silica	Downes and Taylor [14]
Air	Air	N.A.
Au_unloaded	Gold	Palm et al. [33]
Pd014_unloaded	Palladium gold alloy	
Pd034_unloaded	Palladium gold alloy	
Pd034_loaded	Palladium gold alloy	
Pd042_unloaded	Palladium gold alloy	
Pd042_loaded	Palladium gold alloy	
Pd052_unloaded	Palladium gold alloy	
Pd052_loaded	Palladium gold alloy	
Pd073_unloaded	Palladium gold alloy	
Pd073_loaded	Palladium gold alloy	
Pd_unloaded	Palladium	
Pd_loaded	Palladium	

Table 5.2: Materials produced and characterized as part of this thesis included in the simulator. The EDS Pd% refer to the measured palladium concentration reported in section 4.3.

Name	Material	EDS Pd% [%]	Sample ID
Pd000S5	Gold	0	5
Pd100S20	Palladium	100	20
Pd070S9to17	Palladium gold alloy	70	9 to 12 and 14 to 17
Pd041S21	Palladium gold alloy	41	21
Pd051S22	Palladium gold alloy	51	22
Pd061S23	Palladium gold alloy	61	23
Pd069S24	Palladium gold alloy	69	24
Pd064S7and18	Palladium gold alloy	64	7 and 18
Pd053S6	Palladium gold alloy	53	6
Pd074S8drude	Palladium gold alloy	74	8
Pd061S19	Palladium gold alloy	61	19
Pd061S19drude	Palladium gold alloy	61	19

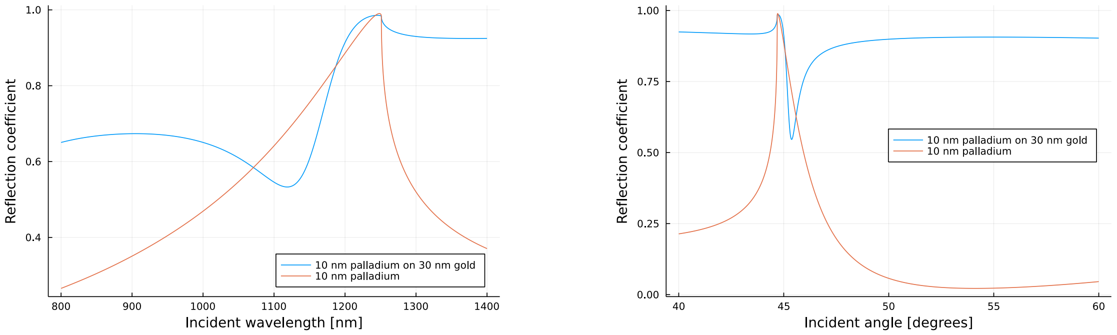
Chapter 6

Simulations and discussion

Several simulations were run for a layered structure consisting of a palladium gold alloy thin-film on top of a thicker gold thin-film deposited on top of silica. This was done to investigate the different effects of design parameters and hydrogen absorption upon the optical response of the thin-film structures used in the proposed hydrogen sensor described in section 1.3. The refractive index of the cladding of optical fibers vary based on fiber type and manufacturer. A generic model for silica was therefore chosen for the simulations, since the specific fiber for the final hydrogen sensor has not yet been determined. The simulation script is included in appendix D listing D.1 and is based on the work described in chapter 5. The experimental data reported in section 4.4 was used for simulating gold, palladium and palladium gold alloy thin-films, with the exception of the hydrogen simulations presented in section 6.2. For section 6.2, the experimental data reported by Palm et al. was used for the palladium gold alloys since this data includes both refractive index and hydrogen response for the same films, thus remaining consistent with the origins of the hydrogen model described in sections 2.3 and B.1 [33]. The Palm alloy thin films were fabricated using co-sputtering similarly to the samples reported on in this thesis, it is therefore expected that the trends found by the simulations are valid for the hydrogen sensor project. The simulations were run with a 45° angle of interface on the silica metal interface when not stated otherwise. All simulations not relating to surface roughness assume ideally smooth films. This was done to greatly speed up the simulations, allowing for higher resolution in wavelength, incident angle and film-thickness. The simulations relating to the effects of surface roughness use the Bruggeman effective media approximation explained in section 2.4 and implemented as described in section 5.4. The roughness is modeled as a 50% blend of air and the 53% palladium gold alloy reported in section 4.4.

6.1 Effects of the gold film

The simulator was used to investigate different potential designs, among those the effects of depositing a gold film between the silica and the palladium gold thin-film. A comparison was made between a structure of 10 nm 53% palladium gold alloy deposited on silica and the same structure with a 30 nm gold layer between the metal layer and the silica. The layer thicknesses were chosen to be in the same range as the thin-films on a similar sensor reported in the literature [21]. The alloying ratio was chosen to be representative for the measurements presented in section 4.4. The results are shown in figure 6.1, where the wavelength and angular spectra of a structure with and without the gold thin-film were investigated. Figure 6.1a shows that a broad plasmon dip is present in the structure with the gold film at $\lambda \approx 1100\,\text{nm}$, while the alloy thin-film deposited directly on the silica only shows the increase in reflection coefficient after the dip. This is confirmed on the angular scan shown in figure 6.1b, where only the structure with the gold film shows both the peak and the following dip of the plasmon signal. These results are in agreement with section 2.2 concerning the enhanced coupling resulting from the intermediary gold layer.



(a) Wavelength reflection spectra with and without gold layer.

(b) Angular reflection spectra with and without gold layer.

Figure 6.1: Wavelength and angular reflection spectra for different thin-film structures. The simulation was run for 10 nm 53% palladium gold alloy deposited on silica with and without a 30 nm gold layer in between.

6.2 Effects of hydrogen

The parameter scan system described in chapter 5 was used to simulate the effects of hydrogen on the gold and palladium gold alloy thin-film structure described in section 6.1, with 10 nm palladium gold alloy on 30 nm gold. The layer thicknesses were chosen to be the same as the structures described in section 6.1. All hydrogen simulations were performed using the gold refractive index from sample 5 reported in section 4.4 and gold palladium alloy refractive indices measured by Palm et al. [33]. The simulation results include the change to the plasmon dip minimum reflection coefficient, the plasmon dip wavelength, the plasmon dip peak to peak and the plasmon dip full width at half maximum (FWHM) as the hydrogen concentration in the atmosphere is increased. The results are shown in figure 6.2. The hydrogen is assumed to affect the refractive index of palladium alloys according to the model described in section 2.3, the expansions of the thin-films were not simulated due to the geometrical dependencies explained in section 2.3. The proposed sensor is intended to sense hydrogen concentrations of less than 5%, the simulations were therefore run for hydrogen concentrations between 0% and 10% to capture all relevant effects which might affect the final hydrogen sensor.

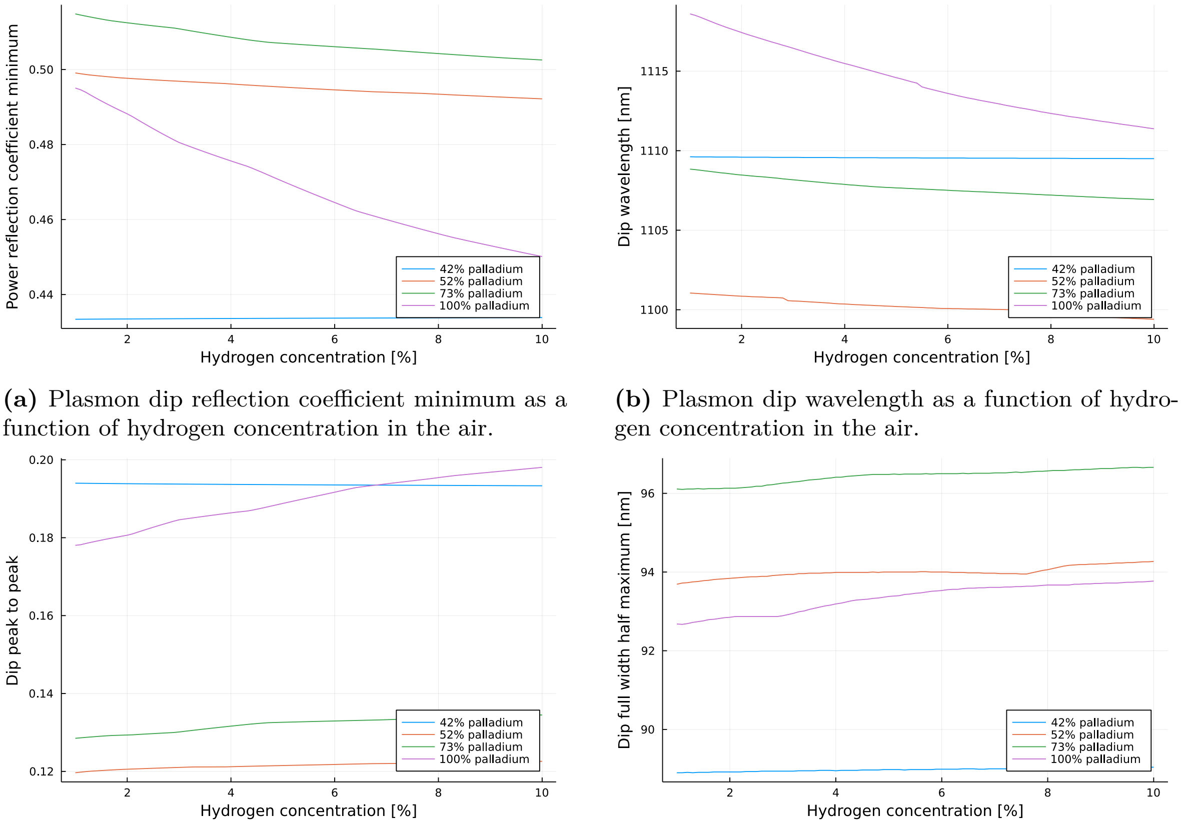
Figures 6.2a and 6.2b both show that the unalloyed palladium film yields the highest sensitivity to hydrogen, and the change to the plasmon dip shrinks significantly as the palladium alloy concentration decreases. Looking at figure 6.2c, it is clear that the plasmon dip in of itself stays almost unchanged with the exception of the pure palladium thin-film, but instead moves both in reflection coefficient minima and wavelength. Figure 6.2d shows that, as the hydrogen concentration increases, the signal grows wider and therefore less specific. It is however negligible in most cases, seeing as the largest observed change in FWHM is under 2% relative change for a hydrogen change of 0% to 10% concentration.

A significant finding is the variation in signal strength observed between the different alloys. Figure 6.2c shows that the plasmon signal is strongest for pure palladium and the 42% palladium alloy depending on hydrogen concentration, while the 73% and 52% thin-films yield significantly weaker signals. This detectability must be weighted against concerns over precision. Figure 6.2d show that 42% palladium has the most narrow plasmon dip signal, but the differences between the signals are small, and may vary with small changes to the other materials in the structure. The difference in plasmon dip FWHM may not be significant in all sensor configurations, as the difference between the best and worst is approximately a 10% relative change. These results should therefore be considered for each specific device configuration.

The same simulations were run with an incident angle of 45.96°, as opposed to the 45.00° used in the other simulations. The results are shown in figure 6.3, note that the small ripple shown in figure 6.3d is caused by the wavelength resolution of the simulation, and is not physical. Many of the same hydrogen induced trends shown in figure 6.2 are present in these simulations. They do however show plasmon dip wavelengths between 1300 nm and 1310 nm for all alloys and hydrogen concentrations, confirming the possibility of the surface plasmons reported by Zhang et al. in a similar structure [21]. The change in plasmon dip wavelength is as expected, since increased incident angle changes the component of the incident wave number k_0 that is parallel to the surface plasmon with a fixed wave number β. Note the increased plasmon dip peak to peak shown in figure 6.3c compared to the simulations at 45°. This is to be expected since the palladium gold alloy refractive index spectra used for the simulations show a higher real refractive index for longer wavelengths as shown for the alloys in figure 4.12a and for pure palladium in figure 2.5c.Another change caused by the increased incident angle is the lower FWHM shown in figure 6.3d. This indicates that the incident angle can be used as a parameter to tune the plasmon dip FWHM as needed for the final hydrogen sensor. The absolute values of the hydrogen induced changes is however similar to what was simulated for a 45° incident angle. This indicates that the effects of the increased hydrogen sensitivity and the increased dampening at higher wavelengths reported by Palm et al. are of approximately the same magnitude, since the dampening is expected to weaken the plasmon signal observed in the reflection spectra [33]. The effects of the incident angle do therefore require further studies and simulations if the optimal incident angle is to be found.

The hydrogen effects on the structure were also simulated for much tinner palladium gold alloy layers of 1 nm with the gold film retaining a thickness of 30 nm, the results are shown in figure 6.4. The thickness of 1 nm was chosen to be close to the ideal layer thickness for plasmon coupling indicated by the simulations presented in section 6.3. Note that the small ripple effect shown in figure 6.4d is caused by the wavelength resolution of the simulation, and is not physical. The results shown in figures 6.4a and 6.4b show that the the hydrogen induced change in the plasmon dip reflection coefficient and wavelength are significantly smaller than what was simulated for the thicker films. It is therefore clear that a certain thickness of the films is required for the hydrogen-induced effects to affect the plasmon signal to a significant degree, this must therefore be taken into account when using the thin-films in devices.

All simulations related to hydrogen concentration have shown that the changes to all figures of merit is strongest for the 100% palladium , and the effect is decreasing with the palladium concentration in the alloys. This is as expected from the theory presented in section 2.3, since the refractive index change caused by the hydrogen is stronger for alloys with higher palladium concentrations. This does not necessarily mean that a higher palladium concentration is always ideal when designing sensors, since alloying can bot be used to tune the dip peak to peak and FWHM as needed, as well as help to avoid cracking and hysteresis in the sensing films.



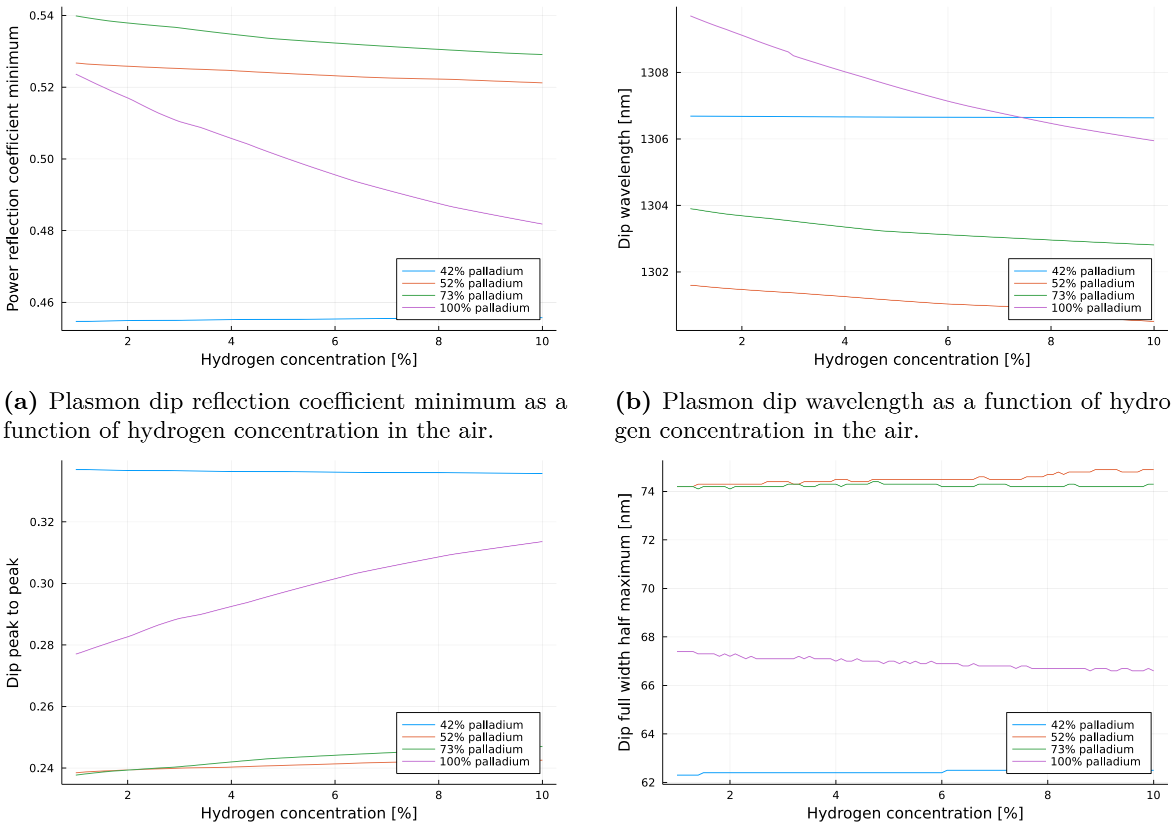
(a) Plasmon dip reflection coefficient minimum as a function of hydrogen concentration in the air.

(b) Plasmon dip wavelength as a function of hydrogen concentration in the air.

(c) Plasmon dip peak to peak as a function of hydrogen concentration in the air.

(d) Plasmon dip full width half maximum as a function of hydrogen concentration in the air.

Figure 6.2: Figures of merit for the characteristic plasmon dip for varying hydrogen concentration in the atmosphere. The structure consists of a 10 nm thin-film of a varying gold palladium alloy deposited on silica with a 30 nm gold thin-film between. It is assumed that the hydrogen only affects the refractive index of the palladium gold alloy.



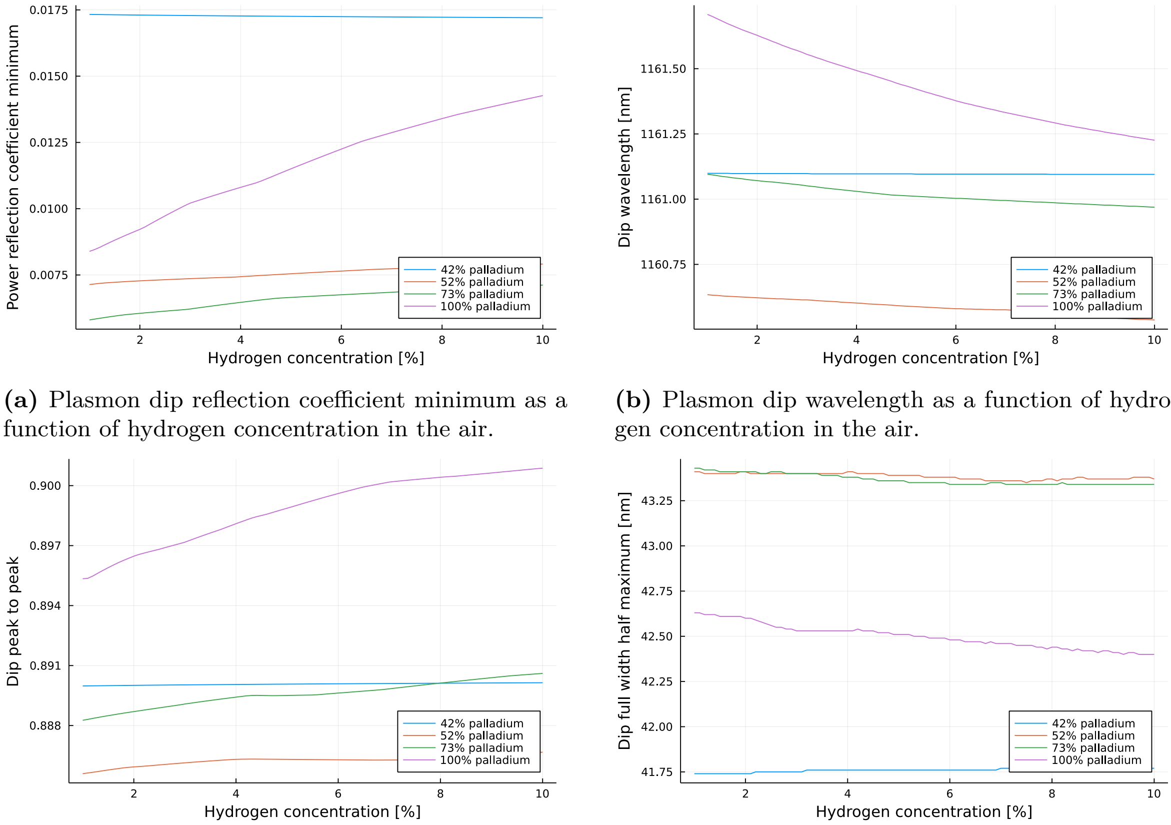
(a) Plasmon dip reflection coefficient minimum as a function of hydrogen concentration in the air.

(b) Plasmon dip wavelength as a function of hydrogen concentration in the air.

(c) Plasmon dip peak to peak as a function of hydrogen concentration in the air.

(d) Plasmon dip full width half maximum as a function of hydrogen concentration in the air.

Figure 6.3: Figures of merit for the characteristic plasmon dip for varying hydrogen concentration in the atmosphere. The structure and assumptions are the same as in figure 6.2, but the optical incident angle is set to 45.96° instead of 45.00°. The small ripple is caused by the wavelength resolution in the simulations.



(a) Plasmon dip reflection coefficient minimum as a function of hydrogen concentration in the air.

(b) Plasmon dip wavelength as a function of hydrogen concentration in the air.

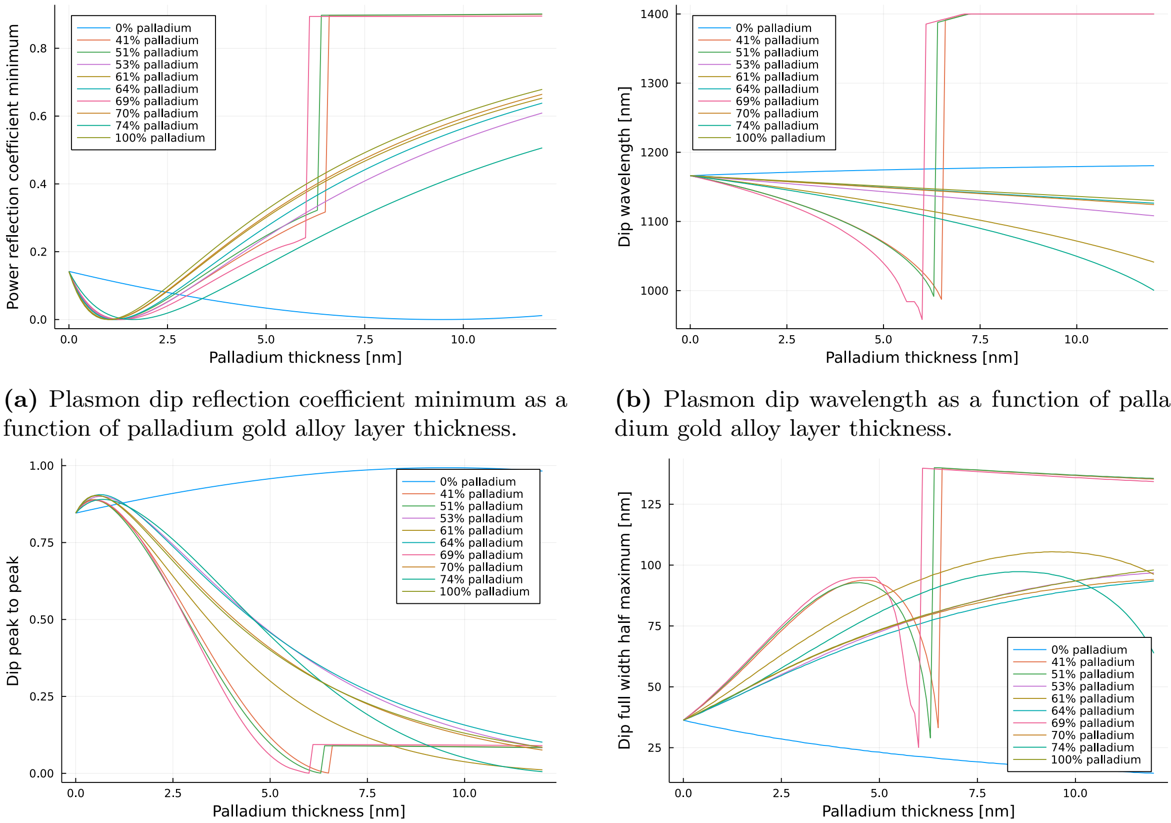
(c) Plasmon dip peak to peak as a function of hydrogen concentration in the air.

(d) Plasmon dip full width half maximum as a function of hydrogen concentration in the air.

Figure 6.4: Figures of merit for the characteristic plasmon dip for varying hydrogen concentration in the atmosphere. The structure consists of a 1 nm thin-film of a varying gold palladium alloy deposited on silica with a 30 nm gold thin-film between. It is assumed that the hydrogen only affects the refractive index of the palladium gold alloy. The small ripple is caused by the wavelength resolution in the simulations.

6.3 Optimization of palladium alloy layer thickness

The effect of the palladium gold alloy layer thickness was simulated for the same figures of merits as those used in section 6.2. The simulated structure is the same gold and palladium gold alloy structure described in section 6.1, and the experimental data presented in section 4.4 is used. The simulation results are presented in figure 6.5.



(a) Plasmon dip reflection coefficient minimum as a function of palladium gold alloy layer thickness.

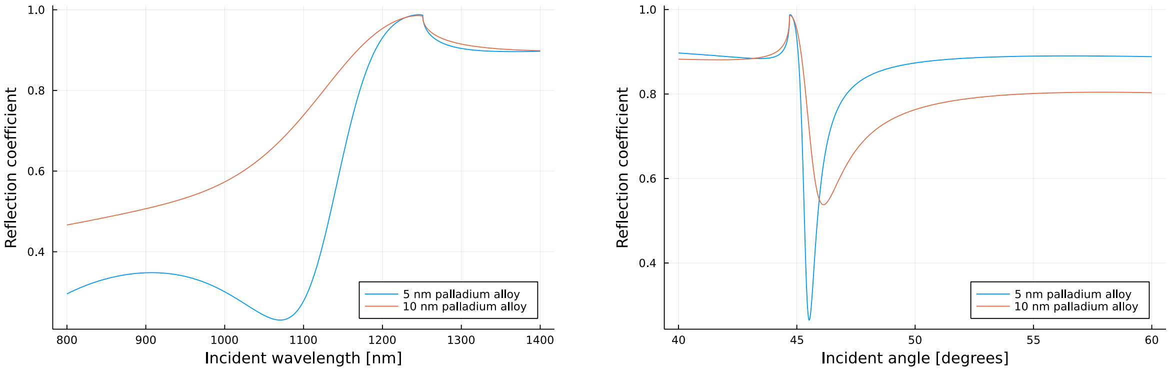
(b) Plasmon dip wavelength as a function of palladium gold alloy layer thickness.

(c) Plasmon dip peak to peak as a function of palladium gold alloy layer thickness.

(d) Plasmon dip full width half maximum as a function of palladium gold alloy layer thickness.

Figure 6.5: Figures of merit for the characteristic plasmon dip for varying palladium gold alloy layer thickness. The structure consists of a 0 nm to 12 nm thin-film of a varying gold palladium alloy deposited on silica with a 30 nm gold thin-film between. Note that the data is invalid for 41%, 51% and 69% palladium for layer thicknesses over 6 nm.

Note that, for the 41%, 51% and 69% palladium gold alloys, the plasmon dip disappears in the reflection coefficient spectra at a layer thickness of approximately 6 nm. This effect causes the figure of merit simulations to fail, creating artifacts in the resulting figures as shown in all parts of figure 6.5. Though the plasmon does not disappear, and is visible in the angular reflection spectrum, it is not present in the wavelength spectrum. This effect is demonstrated in figure 6.6, where the wavelength and angular spectra are simulated for the 41% alloy with an alloy layer thickness of 5 nm and 10 nm. The data from the alloys in question originates from sample 21, 22 and 24, which were found to be significantly different than the thinner samples as discussed in chapter 4. Furthermore, the samples in question are several times thicker than the optimal thickness indicated by the simulations. It is therefore reasonable to focus the analysis on the thinner samples. This means ignoring the simulation results from samples 21 to 24, including the only sample produced with a 41% atomic palladium concentration. This does not affect the trustworthiness of the other results, since they are not dependent on the samples in question.



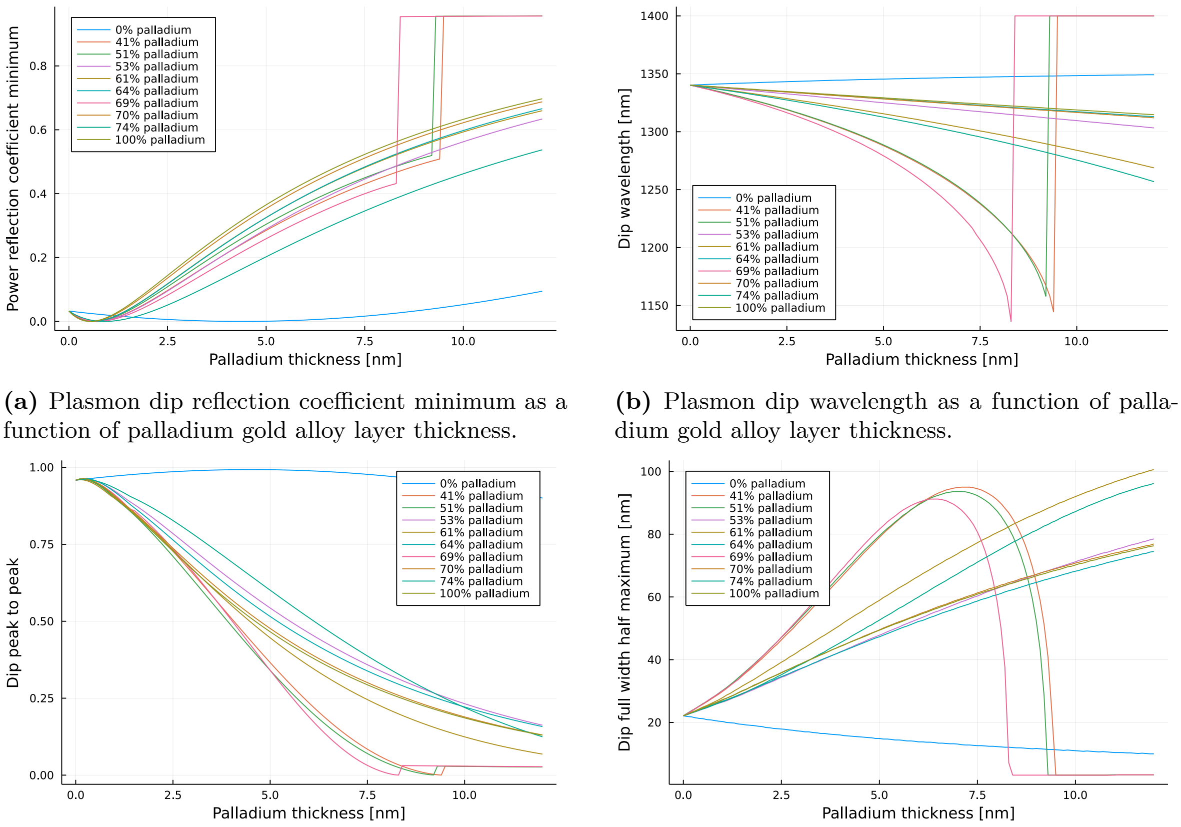
(a) Wavelength reflection coefficient spectrum.

(b) Angular reflection coefficient spectrum.

Figure 6.6: Wavelength and angular reflection coefficient spectra of a 5 nm and 10 nm 41% palladium gold alloy thin-film on a 30 nm gold thin-film on top of silica.

The simulations show that, for all palladium gold alloys, the power reflection coefficient minimum is the smallest and the plasmon dip peak to peak the largest for a layer thickness of a few nanometers. This indicates that, the thinner the palladium gold alloy layer, the larger the plasmon signal will be. Figure 6.5b shows that, as the layer thickness increases, the wavelength of the plasmon dip minima is changed. Finally, figure 6.5d shows how, for films of a few nanometers, an increase in film-thickness leads to an increase in plasmon dip FWHM. The simulations does however show that, past a critical thickness, the FWHM decreases, making the signal more precisely detectable. This is caused by the decrease in plasmon dip peak to peak shown in figure 6.5c, and is an effect of the plasmon dip being too weak to be observed. It is therefore not feasible to increase the alloy layer thickness in order to increase precision of the plasmon signal. It should be noted that absorption of hydrogen will increase the palladium gold alloy thickness in addition to the hydrogen induced refractive index change. Both the two effects will decrease the plasmon dip wavelength and increase the FWHM, but the refractive index dip reflection coefficient and dip peak to peak will be changed differently. It is therefore necessary to simulate the hydrogen induced thin-film expansion to accurately predict the total effect on these two figures of merit.

The film thickness simulations were also run with an incident angle of 45.96°, similar to the hydrogen simulations presented in figure 6.3. The results are shown in figure 6.7. Note that the maximum plasmon dip peak to peak and minimum plasmon dip power reflection coefficient appear at thinner palladium gold alloy layer thicknesses than for the simulations at 45.00° shown in figure 6.5. Since the change in geometrical path length in the thin-films caused by the slight change in incident angle is negligible, it is reasonable to assume the shift is caused by the large increase in extinction coefficient κ for the higher wavelengths reported in section 4.4. This indicates that the incident angle can be used together with the palladium gold alloy layer thickness to tune the plasmon signal which will then in turn be modulated by the hydrogen induced effects discussed in section 6.2.



(a) Plasmon dip reflection coefficient minimum as a function of palladium gold alloy layer thickness.

(b) Plasmon dip wavelength as a function of palladium gold alloy layer thickness.

(c) Plasmon dip peak to peak as a function of palladium gold alloy layer thickness.

(d) Plasmon dip full width half maximum as a function of palladium gold alloy layer thickness.

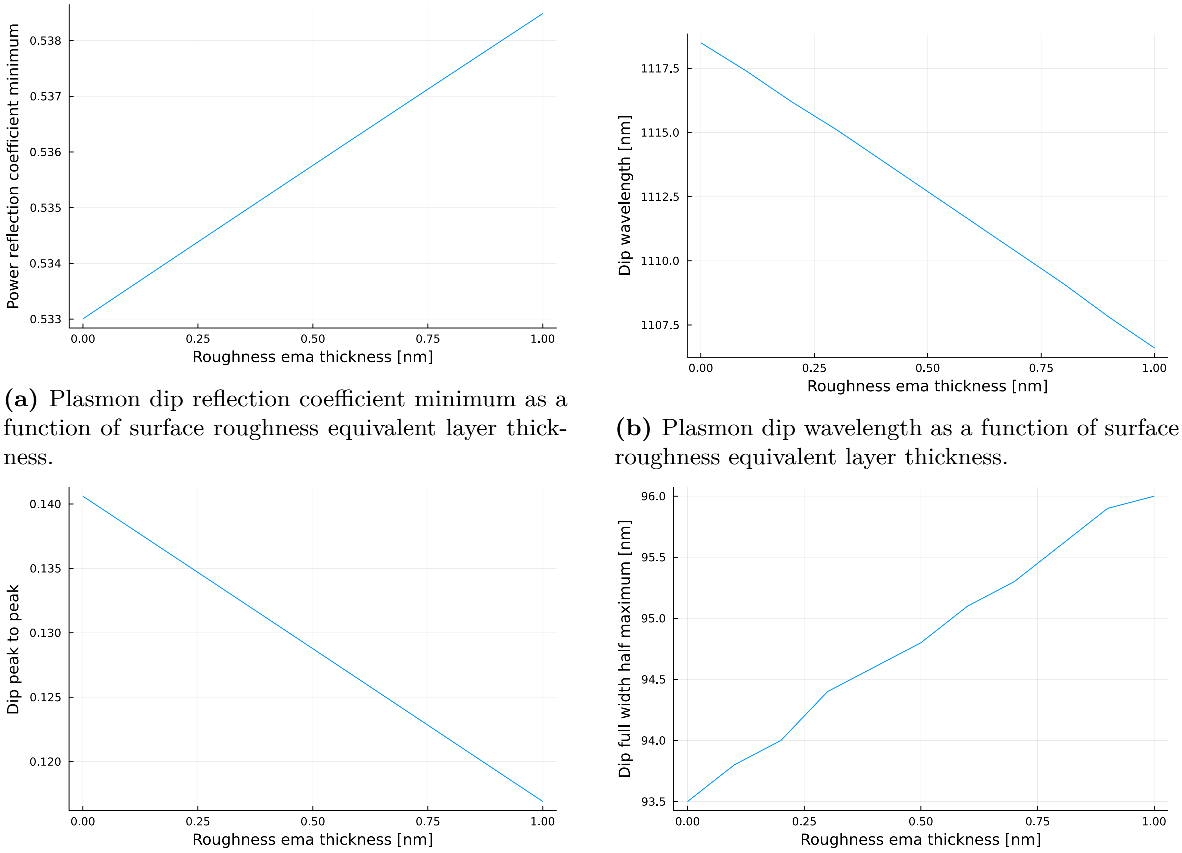
Figure 6.7: Figures of merit for the characteristic plasmon dip for varying palladium gold alloy layer thickness. The structure consists of a 0 nm to 12 nm thin-film of a varying gold palladium alloy deposited on silica with a 30 nm gold thin-film between. The incident angle was set to 45.96°. Note that the data is invalid for 41%, 51% and 69% palladium for layer thicknesses over 8 nm.

The simulations have shown that, in order to increase the detectability and precision of the plasmon signal when only trace amounts of hydrogen is present in the air, the palladium gold alloy thin film should be between 0 and 2.5 nm in thick when the incident angle is 45° as shown in figure 6.5a and 6.5c. It has also been shown that the incident angle can be used to tune the optical wavelength at which the plasmon is coupled into the structure, thereby affecting the ideal palladium gold alloy layer thickness. This must however be weighted against other concerns related to the optical wavelength, such as the hydrogen sensitivity described in section 2.3 or the chromatic dispersion in optical fibers. A compromise must be made between the results presented here and in section 6.2, because if the thin-film is made too thin, its hydrogen-induced refractive index change will not affect the plasmon significantly. This would result in a sensor with a clear plasmon signal which would barely be affected by hydrogen, leading to a decreased sensitivity.

6.4 Effects of surface roughness

Finally, the effects of surface roughness on the palladium gold thin-films were simulated. It was assumed that the effects caused by the large scale surface roughness can be ignored since the grains were found to be significantly larger than the optical wavelengths at which the plasmon dips occur. It was further assumed that the fine grained surface roughness can be modeled using the Bruggeman effective medium approximation described in section 2.4, similar to how its often done in ellipsometry [45]. The simulations were run on the 53% palladium gold alloy on gold structure described in section 6.1, but with an additional layer of 50% alloy, 50% air, Bruggeman EMA layer on top. The thickness of the EMA layer was simulated from 0 nm to 1 nm in order to simulate roughnesses on the same scale as the fine grained roughness reported in section 4.2, the results are shown in figure 6.8.

All the results shown in figure 6.8 indicate that increased surface roughness decreases the quality of the structure for use as a sensing element. Increased roughness results in a lower plasmon dip peak to peak and a higher plasmon FWHM, making the signal both harder to detect and to measure accurately. The simulations show that, even with an EMA layer of 1 nm, the plasmon signal is still measurable, with a 20% decrease in plasmon dip peak to peak and an approximate 5% increase in FWHM. In the ideal case, the sensing film should therefore be perfectly smooth. A small amount of surface roughness can however be tolerated if needed, since thin-film production processes with less control over roughness are generally simpler.



(a) Plasmon dip reflection coefficient minimum as a function of surface roughness equivalent layer thickness.

(b) Plasmon dip wavelength as a function of surface roughness equivalent layer thickness.

(c) Plasmon dip peak to peak as a function of surface roughness equivalent layer thickness.

(d) Plasmon dip full width half maximum as a function of surface roughness equivalent layer thickness.

Figure 6.8: Figures of merit for the characteristic plasmon dip for varying surface roughness equivalent layer thickness. The structure consists of a 10 nm thin-film of a 53% palladium gold alloy deposited on silica with a 30 nm gold thin-film between and with a capping layer of varying thickness modeled as a 50% palladium gold, 50% air Bruggeman effective media.

Chapter 7

Further work

A good next step for the hydrogen sensor project after this thesis is to gather reliable measurements for the hydrogen-induced refractive index change and thin-film expansion. The current hydrogen simulations depend on experimental data from other labs and processing facilities, and can therefore only give limited insight into the quantitative effects of the hydrogen. In addition, the project currently has no model for the increase in film-thickness caused by the hydrogen. With this data, the software model can be adapted to simulate the full effect of hydrogen upon the proposed thin-film structure. This will allow for more accurate information on the trade off between the layer thickness and hydrogen sensitivity concerns discussed in this thesis.

Another useful continuation of the project would be to create a software model for simulating tilted fiber Bragg gratings, and the coupling between core and cladding modes. If this model is developed, it should be accompanied by simulations to optimize the incident angle for the thin-film structure. As discussed in chapter 6, the incident angle can be used to tune the surface plasmon dip wavelength, thereby changing which parts of the refractive index spectra of the materials which are relevant for the sensor response. This can in turn be used to tune figures of merit such as FWHM. Such a model could, if written either in Julia or a language with a compatible foreign function interface, be combined with the simulator and the experimental data gathered in this thesis to simulate the optical response of the hydrogen sensor described in section 1.3. A simulation of the transmission spectrum for the complete sensor will allow for optimization of the design parameters such as layer thickness and material choices, since it is not certain that a highly specific plasmon dip in the thin-film structure reflection spectrum will result in a detectable modulation on the hydrogen sensor's comb-shaped transmission spectrum. Such a model would therefore greatly reduce the needed number of physical prototypes when developing the final sensor, reducing development time and improving the final result.

As mentioned in the specialization project report preceding this thesis, it should be remembered that the model assumes a semi-infinite planar structure, and can therefore only approximate the behavior of the slightly curved fiber surface [22]. Any further work reliant on the simulator should therefore consider simulating the final optimized structure with a tool such as COMSOL that supports curvatures to verify the results [67].

It would also be useful to produce and measure a significant number of gold palladium alloy thin films and characterize their optical properties in a similar way to what has been done as part of this thesis. If a significantly large number of samples are produced for several alloying concentrations and film-thicknesses, statistical analysis can be applied to yield better control over the production of future devices. Such a work should focus on palladium gold alloy thin-films in the range of 1 nm to 30 nm, since the simulations presented in this thesis indicate the optimal thickness with regards to the trade off between plasmon coupling and hydrogen sensitivity lies within this interval. This will greatly improve the accuracy of the simulations, and allow the refractive index data used to evolve from simple experimental measurements to a more complete understanding of the alloys.

Finally, the next step after this thesis should include selecting a specific optical fiber for the sensor, so that the plasmon coupling from the cladding to the layered thin-film structure can be simulated using the correct refractive index for the cladding.

Chapter 8

Conclusion

Several gold palladium alloy thin-films have been produced as part of this project using co-sputtering with different alloying ratios and film thicknesses. These have been characterized with respect to thickness, roughness, alloying and refractive index spectra. This has resulted in a comprehensive experimental foundation for further development of optical sensors using these materials. It has been shown that both the fine grained surface roughness and the deviation from intended alloying ratios increase with the thickness of the thin-films. The existence of a large grained surface roughness has been confirmed, it has been shown that this roughness is unaffected by thin-film thickness and is uncorrelated with the fine grained roughness. The grains of the large grained surface roughness were confirmed to be larger than the optical wavelengths in the near IR range, the optical effects of the two types of roughness are therefore qualitatively different in this range. It has also been shown that the measured thin-films under 100 nm have refractive index spectra similar to what has previously been reported in the literature.

An important application of these new experimental data is to investigate the effects of design parameters and hydrogen concentration on proposed surface plasmon based hydrogen sensors. For this purpose, an open source simulator for investigating the effects of sensor parameters such as layer thickness, surface roughness and hydrogen concentration on figures of merit related to surface plasmons has been developed. The simulator is written in Julia and is based on a previous thin-film simulator written for the specialization project preceding this thesis [22]. This simulator was used together with the experimental data gathered and data from the literature to investigate the trends in sensitivity and precision that can be expected from the proposed hydrogen sensor as a function of palladium gold alloy layer thickness, surface roughness and hydrogen in the atmosphere. Four key figures of merit were simulated, resulting in a comprehensive understanding of how the characteristic plasmon dip moves in the reflection coefficient spectra of different thin-film structures. It was shown that an increased surface roughness will lead to a poorer surface plasmon signal in the thin-film structures reflection spectra, but the effect is minimal for roughnesses on the scale reported in this thesis.

The simulator was also used to investigate the qualitative effects of the incident angle on surface plasmons in layered structures. It was shown that changing the incident angle can be used as a design parameter to tune the plasmon dip wavelength. This will in turn change which region of the refractive index spectra of the media in the structure is relevant for the sensor response, it has been shown that this can in turn change different figures of merit such as FWHM.

Finally, the simulator was used to demonstrate the improved plasmon coupling resulting from an intermediary layer between the palladium gold alloy thin-film and the optical fiber cladding with a lower refractive index than the cladding, such as a gold layer. The simulations have shown that a structure consisting of a thin palladium gold alloy layer deposited on a thicker gold layer on top of the fiber cladding yields a highly tunable structure with good surface plasmon coupling and hydrogen sensitivity. This can provide a good foundation for optimizing surface plasmon based hydrogen sensors with such a thin-film structure as part of the sensing element. The experimentally gathered data has been included in the software package, and can provide a basis for further work with surface plasmon based hydrogen sensors.

Chapter 9

Nomenclature

List of symbols

Symbol	Explanation
$\mathbb{C}$	Set of all complex numbers
$C_{H(M)}$	The atomic concentration of hydrogen absorbed in a metal
d	A geometrical path-length
f	An arbitrary function
f_i	Fill factor of a specific material i in a composite by volume
h	A function relating p_{H_2} to the refractive index change in a palladium alloy
K	The activity coefficient in the law of mass action
k_0	Wave number of an optical wave
n	Refractive index of a medium
n_{eff}	Effective refractive index for a wave in a specific environment
p_{H_2}	The partial pressure of hydrogen gas in the air
T	A temperature
β	Wave number of an optical wave or plasmon along an interface
ϵ	Electrical permittivity
ϵ_i	Electrical permittivity of a specific material i
θ	Incident angle of an optical wave on a border
κ	Extinction coefficient of a material
λ	Optical wavelength

List of abbreviations

Abbreviation	Explanation
AFM	Atomic force microscope
EDS	Energy dispersive spectroscopy
EM	Electromagnetic
EMA	Effective media approach
FWHM	Full width at half maximum
NTNU	Norges teknisk-naturvitenskapelige universitet
PSI	Phase scan interferometry
RMS	Root mean square
SEM	Scanning electron microscope
TFBG	Tilted fiber Bragg grating
TE	Polarization where the electric component is normal to the border
TM	Polarization where the magnetic component is normal to the border
VSI	Vertical scan interferometry

Chapter 10

Sources

[1] A. Prinzhofer, C. Sidy, T. Ciss and A. B. Diallo, ‘Discovery of a large accumulation of natural hydrogen in bourakebougou (mali)’, 2018. DOI: 10.1016/j.ijhydene.2018.08.193. [Online]. Available: https://doi.org/10.1016/j.ijhydene.2018.08.193.

[2] R. Tarkowski, B. Uliasz-Misiak and P. Tarkowski, ‘Storage of hydrogen, natural gas, and carbon dioxide e geological and legal conditions’, 2021. DOI: 10.1016/j.ijhydene.2021.03.131. [Online]. Available: www.sciencedirect.com.

[3] MAN Company, ‘Vorrichtung zur fortlaufenden bestimmung reaktionsfaehiger gasbestandteile, insbesondere von spurenweise vorhandenem sauerstoff’, DRP 165 349, 1904.

[4] T. Hübert, L. Boon-Brett, G. Black and U. Banach, ‘Hydrogen sensors-a review’, *Sensors and Actuators B*, vol. 157, pp. 329–352, 2011. DOI: 10.1016/j.snb.2011.04.070.

[5] S. J. Mihailov, ‘Fiber bragg grating sensors for harsh environments’, *Sensors*, vol. 12, no. 2, pp. 1898–1918, 2012, ISSN: 1424-8220. DOI: 10.3390/s120201898. [Online]. Available: https://www.mdpi.com/1424-8220/12/2/1898.

[6] Norsk elektroteknisk komite, *Elektriske installasjoner i eksposjonsfarlige områder* (NEK (trykt utg.)), 5th ed. Oslo, 2016, vol. 420A:2016.

[7] Norsk elektroteknisk komite, *Eksplosive stoff og varer* (NEK (trykt utg.)), 1st ed. Oslo, 2014, vol. 420C:2014.

[8] European Commission. ‘Key enabling technologies’. (2020), [Online]. Available: https://ec.europa.eu/info/research-and-innovation/research-area/industrial-research-and-innovation/key-enabling-technologies_en (visited on 10/12/2021).

[9] The Research Council of Norway. ‘Porteføljeplan for muliggjørende teknologier’. (11th Sep. 2020), [Online]. Available: https://www.forskningsradet.no/om-forskningsradet/portefoljer/muliggjorende-teknologier/portefoljeplanen-for-muliggjorende-teknologier/overordnede-mal-og-prioriteringer/faglige-og-teknologiske-prioriteringer/ (visited on 08/12/2021).

[10] B. Chadwick, J. Tann, M. Brungs and M. Gal, ‘A hydrogen sensor based on the optical generation of surface plasmons in a palladium alloy’, *Sensors and Actuators:*

B. Chemical, vol. 17, pp. 215–220, 3 1994, ISSN: 09254005. DOI: 10.1016/0925-4005(93)00875-Y.

[11] P. Tobiška, O. Hugon, A. Trouillet and H. Gagnaire, 'An integrated optic hydrogen sensor based on spr on palladium', *Sensors and Actuators B: Chemical*, vol. 74, no. 1, pp. 168–172, 2001, Proceedings of the 5th European Conference on Optical Chemical Sensors and Biosensors, ISSN: 0925-4005. DOI: https://doi.org/10.1016/S0925-4005(00)00728-0. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925400500007280.

[12] X. Bévenot, A. Trouillet, C. Veillas, H. Gagnaire and M. Clément, 'Surface plasmon resonance hydrogen sensor using an optical fibre*', *Meas. Sci. Technol*, vol. 13, pp. 118–124, 2002. [Online]. Available: http://www.univ-st-etienne.fr/tsi.

[13] Y. Liu, Y.-P. Chen and H. Song, 'Modeling analysis and experimental study on the optical fiber hydrogen sensor based on pd-y alloy thin film', *Rev. Sci. Instrum*, vol. 83, p. 75 001, 2012. DOI: 10.1063/1.4731725. [Online]. Available: https://doi.org/10.1063/1.4731725.

[14] F. Downes and C. Taylor, 'Theoretical investigation into the optimisation of an optical fibre surface plasmon resonance hydrogen sensor based on a pdy alloy', *Measurement Science and Technology*, vol. 28, no. 1, p. 015 104, 2016.

[15] T. Gong, P. Lyu, K. J. Palm, S. Memarzadeh, J. N. Munday and M. S. Leite, 'Emergent opportunities with metallic alloys: From material design to optical devices', *Advanced Optical Materials*, vol. 8, pp. 1–21, 23 2020, ISSN: 21951071. DOI: 10.1002/adom.202001082.

[16] C. Zhang, C. Shen, X. Liu *et al.*, 'Pd/au nanofilms based tilted fiber bragg grating hydrogen sensor', *Optics Communications*, vol. 502, p. 127 424, August 2021 2022, ISSN: 00304018. DOI: 10.1016/j.optcom.2021.127424. [Online]. Available: https://doi.org/10.1016/j.optcom.2021.127424.

[17] L. J. Bannenberg, F. A. A. Nugroho, H. Schreuders *et al.*, 'Direct comparison of pdau alloy thin films and nanoparticles upon hydrogen exposure', *ACS Applied Materials and Interfaces*, vol. 11, pp. 15 489–15 497, 17 2019, ISSN: 19448252. DOI: 10.1021/acsami.8b22455.

[18] J. Albert, L. Y. Shao and C. Caucheteur, 'Tilted fiber bragg grating sensors', *Laser and Photonics Reviews*, vol. 7, pp. 83–108, 1 2013, ISSN: 18638880. DOI: 10.1002/lpor.201100039.

[19] J. Yu, Z. Wu, X. Yang, X. Han and M. Zhao, 'Tilted fiber bragg grating sensor using chemical plating of a palladium membrane for the detection of hydrogen leakage', *Sensors (Switzerland)*, vol. 18, 12 2018, ISSN: 14248220. DOI: 10.3390/s18124478.

[20] NTNU. 'Enersense - energy and sensor systems'. (), [Online]. Available: https://www.ntnu.edu/enersense#/view/about (visited on 10/06/2022).

[21] X. Zhang, S. Cai, F. Liu *et al.*, 'In situ determination of the complex permittivity of ultrathin h2-infused palladium coatings for plasmonic fiber optic sensors in the near infrared', *Journal of Materials Chemistry C*, vol. 6, pp. 5161–5170, 19 2018, ISSN: 20507526. DOI: 10.1039/c8tc01278d.

[22] T. Nordgård-Hansen, 'Modeling of surface plasmons in layered thin films for hydrogen sensors', NTNU, NTNU, Institutt for elektroniske systemer, 7491 Trondheim, Tech. Rep., Dec. 2021, Specialization Project.

[23] R. M. Carter, P. Morrall, R. R. J. Maier, B. J. S. Jones, S. McCulloch and J. S. Barton, 'Optical characterisation of rf sputter coated palladium thin films for

hydrogen sensing', *21st International Conference on Optical Fiber Sensors*, vol. 7753, 77535T, 2011, ISSN: 0277786X. DOI: 10.1117/12.886057.

[24] H. Arwin, *Thin film optics and polarized light*, 7nd ed. Hasn Arwin, 2021.

[25] S. A. Maier, *Plasmonics: Fundamentals and Applications: Fundamentals and Applications.* Springer, 2007.

[26] B. E. Saleh and M. C. Teich, *Fundamentals of photonics* (Wiley series in pure and applied optics), 3nd ed. Hoboken, N.J: John Wiley & Sons, Inc, 2019, ISBN: 9781118770092.

[27] H. Raether, *Surface plasmons on smooth and rough surfaces and on gratings.* Springer-Verlag, 1988, vol. 111, ISBN: 9783540474418.

[28] S. Cai, Á. González-Vila, X. Zhang, T. Guo and C. Caucheteur, 'Palladium-coated plasmonic optical fiber gratings for hydrogen detection', *Optics Letters*, vol. 44, p. 4483, 18 2019, ISSN: 0146-9592. DOI: 10.1364/ol.44.004483.

[29] JuliaLang. 'The julia programming language'. (2012), [Online]. Available: https://julialang.org/ (visited on 29/10/2021).

[30] Torstein Nordgård-Hansen. 'Optical matrixsimulator'. (19th Dec. 2021), [Online]. Available: https://github.com/torsteinnh/optical_matrixsimulator (visited on 29/10/2021).

[31] P. B. Johnson and R. W. Christy, 'Optical constants of the noble metals', *Phys. Rev. B*, vol. 6, pp. 4370–4379, 12 Dec. 1972. DOI: 10.1103/PhysRevB.6.4370. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevB.6.4370.

[32] W. S. M. Werner, K. Glantschnig and C. Ambrosch-Draxl, 'Optical constans and inelastic electron-scattering data for 17 elemental metals', *J. Phys. Chem. Ref. Data*, vol. 38, p. 1013, 2009. DOI: 10.1063/1.3243762. [Online]. Available: https://doi.org/10.1063/1.3243762.

[33] K. J. Palm, J. B. Murray, J. P. Mcclure, M. S. Leite and J. N. Munday, 'In situ optical and stress characterization of alloyed pd x au 1−x hydrides', *ACS Appl. Mater. Interfaces*, vol. 11, p. 27, 2019. DOI: 10.1021/acsami.9b14244. [Online]. Available: www.acsami.org.

[34] Babar and Weaver. 'Refractiveindex.info'. (2015), [Online]. Available: https://refractiveindex.info/?shelf=main&book=Au&page=McPeak (visited on 08/05/2022).

[35] Babar and Weaver. 'Refractiveindex.info'. (2015), [Online]. Available: https://refractiveindex.info/?shelf=main&book=Au&page=Babar (visited on 08/05/2022).

[36] R. L. Olmon, B. Slovick, T. W. Johnson *et al.*, 'Optical dielectric function of gold', *Phys. Rev. B*, vol. 86, p. 235 147, 23 Dec. 2012. DOI: 10.1103/PhysRevB.86.235147. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevB.86.235147.

[37] G. Rosenblatt, B. Simkhovich, G. Bartal and M. Orenstein, 'Nonmodal plasmonics: Controlling the forced optical response of nanostructures', 2020. DOI: 10.1103/PhysRevX.10.011071. [Online]. Available: https://doi.org/10.1103/PhysRevX.10.011071.

[38] P. B. Johnson and R. W. Christy, 'Optical constants of transition metals: Ti, v, cr, mn, fe, co, ni, and pd', *Phys. Rev. B*, vol. 9, pp. 5056–5070, 12 Jun. 1974. DOI: 10.1103/PhysRevB.9.5056. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevB.9.5056.

[39] K. J. Palm, J. B. Murray, T. C. Narayan and J. N. Munday, 'Dynamic optical properties of metal hydrides', *ACS Photonics*, vol. 5, pp. 4677–4686, 2018. DOI: 10.1021/acsphotonics.8b01243. [Online]. Available: https://pubs.acs.org/sharingguidelines.

[40] N. Sharma, A. Joy, A. Mishra and R. Verma, 'Fuchs sondheimer-drude lorentz model and drude model in the study of spr based optical sensors: A theoretical study', *Optics Communications*, vol. 357, Nov. 2015. DOI: 10.1016/j.optcom.2015.08.092.

[41] G. Wang, J. Dai and M. Yang, 'Fiber-optic hydrogen sensors: A review', *IEEE Sensors Journal*, c 2020, ISSN: 15581748. DOI: 10.1109/JSEN.2020.3029519.

[42] C. M. Guldberg and P. Waage, *Studier over Affiniteten*, Norwegian. 1864.

[43] S. Cai, F. Liu, R. Wang *et al.*, 'Narrow bandwidth fiber-optic spectral combs for renewable hydrogen detection', *Science China Information Sciences*, vol. 63, pp. 1–9, 12 2020, ISSN: 18691919. DOI: 10.1007/s11432-020-3058-2.

[44] R. Jansson and H. Arwin, 'Selection of the physically correct solution in the n-media bruggeman effective medium approximation', *Optics Communications*, vol. 106, pp. 133–138, 1994.

[45] A. Yanguas-Gil and H. Wormeester, 'Relationship between surface morphology and effective medium roughness', in *Ellipsometry at the Nanoscale*, M. Losurdo and K. Hingerl, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 179–202, ISBN: 978-3-642-33956-1. DOI: 10.1007/978-3-642-33956-1_4. [Online]. Available: https://doi.org/10.1007/978-3-642-33956-1_4.

[46] J. Mohrmann, T. E. Tiwald, J. S. Hale, J. N. Hilfiker and A. C. Martin, 'Application of a b-spline model dielectric function to infrared spectroscopic ellipsometry data analysis', *Journal of Vacuum Science & Technology B*, vol. 38, p. 014 001, 1 Jan. 2020, ISSN: 2166-2746. DOI: 10.1116/1.5126110.

[47] NTNU. 'Ntnu nanolab'. (), [Online]. Available: https://www.ntnu.edu/nano/nanolab (visited on 14/04/2022).

[48] Torstein Nordgård-Hansen and Michael Aaron Fried. 'Rawdata from the lab'. (14th Jun. 2022), [Online]. Available: https://pvv.ntnu.no/~torsteno/masterdata/ (visited on 09/05/2022).

[49] V. Håkonsen. 'Plasma cleaner'. (1st Jan. 2008), [Online]. Available: http://ntnu.norfab.no/WebForms/Equipment/EquipmentView.aspx?toolId=38 (visited on 18/04/2022).

[50] Micro resist technologies. 'Ma-n 400 series'. (18th Jun. 2020), [Online]. Available: https://www.microresist.de/en/produkt/ma-n-400-series/ (visited on 19/04/2022).

[51] Micro resist technologies. 'Ma-d 332'. (), [Online]. Available: https://www.microresist.de/en/produkt/ma-d-332/ (visited on 19/04/2022).

[52] M. Chiappa. 'Mla 150'. (17th Feb. 2017), [Online]. Available: http://ntnu.norfab.no/WebForms/Equipment/EquipmentView.aspx?toolId=100 (visited on 18/04/2022).

[53] M. de Roosz. 'E beam evaporator & sputter aja'. (28th Sep. 2012), [Online]. Available: http://ntnu.norfab.no/WebForms/Equipment/EquipmentView.aspx?toolId=73 (visited on 19/04/2022).

[54] J. Vinje. 'Veeco dektak 150 profilometer'. (), [Online]. Available: http://ntnu.norfab.no/WebForms/Equipment/EquipmentView.aspx?toolId=9 (visited on 21/04/2022).

[55] J. Vinje. ‘Bruker contour gt -k 3d optical profiler’. (24th Jan. 2018), [Online]. Available: `http://ntnu.norfab.no/WebForms/Equipment/EquipmentView.aspx?toolId=105` (visited on 21/04/2022).

[56] D. Necas and P. Klapetek, *Gwyddion*, version 2.61, Department of Nanometrology, Czech Metrology Institute, 2nd May 2022. [Online]. Available: `http://gwyddion.net/` (visited on 11/05/2022).

[57] J. Vinje. ‘Afm veeco dimultimode v’. (15th Feb. 2006), [Online]. Available: `http://ntnu.norfab.no/WebForms/Equipment/EquipmentView.aspx?toolId=2` (visited on 10/05/2022).

[58] V. Håkonsen. ‘Sem apreo’. (25th Jan. 2017), [Online]. Available: `http://ntnu.norfab.no/WebForms/Equipment/EquipmentView.aspx?toolId=96` (visited on 22/04/2022).

[59] J.A. Woollam Ellipsometry Solutions. ‘Rc2 ellipsometer’. (), [Online]. Available: `https://www.jawoollam.com/products/rc2-ellipsometer` (visited on 26/05/2022).

[60] D. Ovtcharov, ‘Certificate of calibration: Step height standard’, Standards *ncorporated*, NTNU NanoLab, Certificate of calibration, 27th Apr. 2022.

[61] Bruker, ‘Contourgt objectives chart, Turret mountable, standard objective series (parfocal with each other)’, NTNU NanoLab.

[62] L. Lin, C. Feng, R. Lopez and O. Coronell, ‘Identifying facile and accurate methods to measure the thickness of the active layers of thin-film composite membranes - a comparison of seven characterization techniques’, *Journal of Membrane Science*, vol. 498, pp. 167–179, Jan. 2016, ISSN: 18733123. DOI: `10.1016/J.MEMSCI.2015.09.059`. [Online]. Available: `http://dx.doi.org/10.1016/j.memsci.2015.09.059`.

[63] Veeco, *Dimultimode v, The world’s highest-resolution spm*, 2006. [Online]. Available: `http://www.eicsac.com/multimodev.pdf` (visited on 11/06/2022).

[64] K. D. Lee, *Foundations of programming languages*, 2nd ed. 2017, ISBN: 9783319707891.

[65] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, *Introduction to Algorithms*, 3rd ed. 2009, ISBN: 9780262033848.

[66] H. H. Li. ‘Refractiveindex.info’. (1976), [Online]. Available: `https://refractiveindex.info/?shelf=main&book=LiF&page=Li` (visited on 02/11/2021).

[67] Comsol. ‘Comsol wave optics module’. (2021), [Online]. Available: `https://www.comsol.com/wave-optics-module` (visited on 08/12/2021).

[68] Ankit Rohatgi. ‘Webplotdigitizer 4.5’. (15th Aug. 2021), [Online]. Available: `https://automeris.io/WebPlotDigitizer/` (visited on 02/05/2022).

Appendices

Appendix A

Fabrication schedule

Table A.1: List of samples fabricated for the project, the ratio of palladium in the palladium gold alloys are given by weight part 1/2.

Sample ID	Sputtered Pd% [%]	Cleaning	Lithography	Sputtering	Liftoff
Wafer 2	51	23.3.2022	23.3.2022	25.3.2022	25.3.2022
Sample 1	0	29.3.2022	29.3.2022	30.3.2022	30.3.2022
Sample 2	100	29.3.2022	29.3.2022	30.3.2022	30.3.2022
Sample 3	56	29.3.2022	29.3.2022	30.3.2022	30.3.2022
Sample 4	56	29.3.2022	29.3.2022	30.3.2022	30.3.2022
Sample 5A	0	2.4.2022	2.4.2022	3.4.2022	5.4.2022
Sample 5B	0	2.4.2022		3.4.2022	
Sample 6A	40	2.4.2022	2.4.2022	3.4.2022	5.4.2022
Sample 6B	40	2.4.2022		3.4.2022	
Sample 7A	50	2.4.2022	2.4.2022	3.4.2022	5.4.2022
Sample 7B	50	2.4.2022		3.4.2022	
Sample 8A	60	2.4.2022	2.4.2022	4.4.2022	5.4.2022
Sample 8B	60	2.4.2022		4.4.2022	
Sample 9A	60	2.4.2022	2.4.2022	4.4.2022	5.4.2022
Sample 9B	60	2.4.2022		4.4.2022	
Sample 10A	60	2.4.2022	2.4.2022	4.4.2022	5.4.2022
Sample 10B	60	2.4.2022		4.4.2022	
Sample 11A	60	2.4.2022	2.4.2022	4.4.2022	5.4.2022
Sample 11B	60	2.4.2022		4.4.2022	
Sample 12A	60	2.4.2022	2.4.2022	4.4.2022	5.4.2022
Sample 12B	60	2.4.2022		4.4.2022	
Sample 13A	60	2.4.2022	2.4.2022	3.4.2022	5.4.2022
Sample 13B	60	2.4.2022		3.4.2022	
Sample 14A	60	2.4.2022	2.4.2022	3.4.2022	5.4.2022
Sample 14B	60	2.4.2022		3.4.2022	

Table A.2: List of samples fabricated for the project, the ratio of palladium in the palladium gold alloys are given by weight part 2/2.

Sample ID	Sputtered Pd% [%]	Cleaning	Lithography	Sputtering	Liftoff
Sample 15A	60	2.4.2022	2.4.2022	3.4.2022	5.4.2022
Sample 15B	60	2.4.2022		3.4.2022	
Sample 16A	60	2.4.2022	2.4.2022	3.4.2022	5.4.2022
Sample 16B	60	2.4.2022		3.4.2022	
Sample 17A	60	2.4.2022	2.4.2022	3.4.2022	5.4.2022
Sample 17B	60	2.4.2022		3.4.2022	
Sample 18A	50	2.4.2022	2.4.2022	4.4.2022	5.4.2022
Sample 18B	50	2.4.2022		4.4.2022	
Sample 19A	50	2.4.2022	2.4.2022	4.4.2022	5.4.2022
Sample 19B	50	2.4.2022		4.4.2022	
Sample 20A	100	2.4.2022	2.4.2022	3.4.2022	5.4.2022
Sample 20B	100	2.4.2022		3.4.2022	
Sample 21A	30	4.5.2022	4.5.2022	5.5.2022	5.5.2022
Sample 21B	30	4.5.2022		5.5.2022	
Sample 22A	40	4.5.2022	4.5.2022	5.5.2022	5.5.2022
Sample 22B	40	4.5.2022		5.5.2022	
Sample 23A	60	4.5.2022	4.5.2022	5.5.2022	5.5.2022
Sample 23B	60	4.5.2022		5.5.2022	
Sample 24A	70	4.5.2022	4.5.2022	5.5.2022	5.5.2022
Sample 24B	70	4.5.2022		5.5.2022	

Appendix B

Function by function documentation

The core of the matrix simulator is documented in the specialization project report preceding this thesis, this appendix provides documentation for the expansion to the model [22, chapter 3]. Julia modules are written in *italics*, function names in **bold**.

B.1 Changes to *analyticalmaterials*

The **n_MaxwellGarnett** function was moved to the *ema* module as part of the work on effective media simulations. This function is otherwise unchanged both in interface and implementation.

The *hs_Palm* sub-module was added as part of the *analyticalmaterials* module, providing an approximation of the hydrogen dependent change in the refractive index of palladium as described in section 2.3. The module exports the function **h** such that $n_{Pd_c} = \sqrt{h} * n_{Pd_0}$, where n_{Pd_c} is the refractive index of palladium in an atmosphere with a hydrogen partial pressure $p_{H_2} = c$, and n_{Pd_0} is the refractive index of palladium in the absence of hydrogen.

The module is based on data scraped from Palm et al. [33]. This data is loaded once at compile time as *rawdatafile034*, *rawdatafile042*, *rawdatafile052*, *rawdatafile073* and *rawdatafile100*. The scraped data is available with the source code on GitHub [30].

The **loadLine** function is an internal tool for reading the aforementioned data and creating an anonymous function mapping hydrogen concentration to a h-value for a specific wavelength λ and gold palladium alloy ratio.

The **c_to_HM** function is an internal tool for estimating the relation between the atomic hydrogen concentration in air and in a gold palladium alloy. It assumes that $C_{H(M)} \propto \sqrt{p_{H_2}}$ as discussed in section 2.3, where $C_{H(M)}$ is the concentration of hydrogen in the alloy and p_{H_2} the partial pressure of hydrogen in the air.

The **h** function is the only exported function. It takes a wavelength λ, a hydrogen concentration in an atmosphere with a pressure of 1 atm *H_c* and an atomic palladium alloy concentration *Pd_c* as its arguments and returns a complex number *h*.

B.2 The *ema* module

The **n_MaxwellGarnett** functions implement the Maxwell Garnett effective media approximation [24, eq. 2.23]. The simplest form approximates the refractive index of a two component media, it takes the electrical permittivity of the host material, the electrical permittivity of the embedded material and the fill factor of the embedded material as its arguments. The second form works for a medium with an arbitrary number of components. Instead of numbers, it takes a vector of electrical permittivities and fill factors as its second and third arguments. Both functions return a single refractive index. Note that the simple form was first written for the specialization project [22].

The **n_Bruggeman** functions implements the Bruggeman effective media approximation [24, eq. 2.26]. It is implemented for both two and three component media. The approximation supports media with more components, but this has not been implemented in this project. The two component variant takes the electrical permittivity of the first material, the electrical permittivity of the second material and the fill factor of the first as its arguments. The three component variant takes the electrical permittivity of the first, second and third material, followed by the fill factor of the first, second and third as its arguments. Though all three fill factors sum up to one, all three are taken for ease of use. Both functions return a single refractive index.

The **select_Wiener_circle** and **select_Wiener_polygon** functions implement an algorithm by Jannson and Arwin to select the correct solution to the Bruggeman effective media approximation [44]. The **select_Wiener_circle** takes a potential approximated electrical permittivity as well as the permittivity of two materials as its arguments, and returns true if the approximation is a physically viable blend of the two others. The **select_Wiener_polygon** takes an approximated permittivity and a vector of permittivities as its arguments, and returns true if the approximation lies within the complex polygon spanned by the other materials. These functions are only used by the **n_Bruggeman** functions and should not be used independently without consulting the explanation in the paper [44].

B.3 Changes to *materials*

The **LoadMaterial** function from the specialization project was expanded with another variant that takes a path for real data, a path for complex data and a number indicating which column to read as its arguments. The function loads refractive index data on the export form of WebPlotDigitizer when digitizing refractive index data split between multiple figures [68]. The function returns a function mapping optical wavelength to complex refractive index on the same format as the other variant.

The *materials* module was extended with several example materials. All are functions on the form $f : \lambda \to n$. They are documented in section 5.3.

B.4 The *targetfigures* module

The *targetfigures* module contains three kinds of functions: figure of merit estimators, parameter scan functions and system builders. All parts of the system is implemented in a semi-functional paradigm, and systems are represented as higher order functions. It is therefore recommended to read both the documentation and the source code before using this module.

B.4.1 Figure of merit estimators

The **plasmon_minima** function is a tool for finding the minima of the plasmon dip in the reflection spectrum of a system. It takes a system function, a vector of wavelengths and an optional predipp parameter as described in section 5.2.1 as its arguments, and returns the minimum reflection coefficient as well as the corresponding wavelength.

The **plasmon_halfwidth** function is a tool for finding the full width at half maximum (FWHM) of the plasmon dip in the reflection spectrum. It takes a system function, a vector of wavelengths, an optional predipp parameter and an optinal widthratio parameter as its arguments. The widthratio can be used to change the threshold for finding the maximum, it defaults to half maximum. The function returns the width, the pulse peak to peak, the leftmost and rightmost edge of the dip and the wavelength at which the reflection coefficient minimum occurs.

The **scan_minima** and **scan_maxima** functions are general utilities for finding the global minimum and maximum of a curve. They both take two vectors of numbers as their arguments, each representing one axis of the curve. The function return the coordinates for the global minimum or maximum in the given curve using a single-pass scan implemented in the code. The implementation uses manual bounds checking to omit valid index checking at each point and speed up the compiled code.

B.4.2 Scan functions

The **scan_singleparameter** function provides a standardized utility for scanning a system as a function of a parameter range. It applies a customizable post-processing function at every point along the scan, making the function highly reusable. The arguments are a system-function on the form $f : \mathbb{C} \to T$, a vector of numbers and an optional post-processing function on the form $g : T \to G$, where T and G are generic types. The function returns the scan range and a vector of type G containing the results from the scan. The default post-processing function finds the power reflection coefficient from a scattering matrix, and is intended to be used for a wavelength scan on a layered system.

The **scan_plasmon_singleparameter** function is a utility for scanning a structure for a single design parameter, for example layer thickness, and computing several figure of merits related to plasmon sensing at each point. Its arguments are a system function, a vector of parameters to scan, a vector of wavelengths for the plasmon detection and a predip parameter as described in section 5.2.1. The system function maps numbers to functions that again maps wavelength to scattering matrices, these kinds of system functions can easily be generated by the system builders present in this module. The **scan_plasmon_singleparameter** returns the parameter scan range, a vector of the minimum power reflection coefficient, a vector of the minimum power reflection coefficient wavelength, a vector of plasmon dip line-width and a vector of plasmon dip peak-to-peaks.

The **scan_plasmon_dualparameter** function is a utility for scanning a structure for two design parameters. It takes a system function, two vectors of design parameters, a vector of wavelengths, a predip parameter and a width-ratio parameter. The function scans the system for both parameters, finding the parameter from the second vector that minimizes the power reflection coefficient for each of the points in the first parameter vector. This is done using the **scan_minima** function on the plot of reflection coefficient minimum and second parameter vector for each element in the first parameter vector. The width-ratio defaults to 0.5, and it can be used to change the width of the full width at half maxima scan. It returns the first parameter vector, the corresponding parameters yielding the minimum reflection coefficient and the corresponding parameters yielding the minimum plasmon dip wavelength. Note that the last return value is a placeholder meant to preserve API-stability in the event that a suitable target function other than global minimum is found for the plasmon dip wavelengths.

The **scan_plasmon_dualthicknesses** function is closely related to the **scan_plasmon_dualparameter** function. It takes a system function, minimum, maximum and step lengths for two distance intervals, a predip parameter and a layer tolerance parameter as arguments. The function scans the system for both thicknesses. For each thickness in the first range, it finds the thickness in the second range that yields the minimum plasmon dip reflection coefficient, the plasmon dip wavelength, the minimum plasmon dip width and the maximum plasmon dip peak to peak. The function returns the first range of thicknesses, the corresponding second thicknesses yielding the minimum reflection coefficient, the corresponding second thicknesses yielding the minimum wavelength, the corresponding second thickness yielding the minimum full width at half maximum and the corresponding wavelength yielding the maximum plasmon dip peak to peak. Note that the minimum plasmon dip wavelength is included as a placeholder to maintain API-stability in the event that a suitable target figure is found.

B.4.3 System builders

The **make_layered_tm_system** and **make_layered_θ_tm_system** functions take a vector of refractive index functions on the form $f : \lambda \to n$, a vector of layer thicknesses in meters and either a wavelength or an incident angle for the incoming light. It returns a function on the form $f : \lambda \to S$ or $f : \theta \to S$, where S is the scattering matrix corresponding to the layered system described by the input vectors at either a certain optical wavelength λ or an incident angle θ in radians.

The **make_d3_system** and **make_d4_system** functions takes five refractive index functions, four layer thicknesses and an incident angle as its arguments. They then return a system function that takes the thickness of the missing third or fourth layer as its argument and returns a wavelength-dependent system function for the resulting system. The wavelength-dependent function is of the same type as those returned by the **make_layered_tm_system**function.

The **make_d2_dualparameter_system** function takes five refractive index functions, three layer thicknesses and an incident angle as its arguments. It returns a function that takes the thickness of the missing second layer as its arguments, returning a system function on the form generated by the **make_d3_system** function, which again takes the thickness of the missing third layer as its argument and returns a function on the form generated by the **make_layered_tm_system** function.

The **make_hc_system** function takes five refractive index functions, five layer thicknesses, and incident angle and a palladium atomic alloying percentage as its arguments. It returns a function on the form $f : p_{H_2} \rightarrow S$, where p_{H_2} is the partial pressure of hydrogen in the environment measured in atmospheres. The function assumes that only the third layer contains palladium, and that no other layers interacts with the hydrogen.

B.5 Changes to the project structure

The simulation has been changed from version 0.1.0 to 0.2.0 as shown in listing C.1. It also shows the removal of *Ipopt* and *JuMP* as dependencies, as well as the additions of *GeometricalPredicates*, *Match* and *Polynomials*. Listing C.2 shows the addition of the *ema* and *targetfigures* modules to the project.

Appendix C

Model source code

The full source code is available on GitHub under the MIT license [30]. The core of the program was written as part of the specialization project preceding this thesis, the additions to the original work is presented in chapter 5 [22]. The following source files are included in this appendix:

- Listing C.1 contains the manifest for initializing the simulator as a Julia pkg package.
- Listing C.2 contains the top module of the simulator, and should be loaded when using the simulator.
- Listing C.3 contains the matrix calculations at the core of the simulator.
- Listing C.4 contains an implementation of the Fresnel equations to be used in conjunction with listing C.3.
- Listing C.5 contains a toolbox for working analytically with materials relevant for plasmon based hydrogen sensors. Note that the *hs_Palm* module relies heavily on experimental data from the literature [33].
- Listing C.6 contains an extensive library of refractive index data for gold, palladium and silica.
- Listing C.7 contains a toolbox for modeling composites based on an effective medium approximation.
- Listing C.8 contains a framework for performing parameter scans for layered structures.

Code Listing C.1: Project.toml

```
name = "simulator"
uuid = "ff2c548e-ca0f-4ef0-904d-a9760906fddf"
authors = ["torsteinnh <torsteinnh@gmail.com>"]
version = "0.2.0"

[deps]
```

```
DelimitedFiles = "8bb1440f-4735-579b-a4ab-409b98df4dab"
GeometricalPredicates = "fd0ad045-b25c-564e-8f9c-8ef5c5f21267"
Interpolations = "a98d9a8b-a2ab-59e6-89dd-64a1c18fca59"
LinearAlgebra = "37e2e46d-f89d-539d-b4ee-838fcccc9c8e"
Match = "7eb4fadd-790c-5f42-8a69-bfa0b872bfbf"
Plots = "91a5bcdd-55d7-5caf-9e0b-520d859cae80"
Polynomials = "f27b6e38-b328-58d1-80ce-0feddd5e7a45"
Test = "8dfed614-e22c-5e08-85e1-65c5234f0b40"
```

Code Listing C.2: src/simulator.jl

```
module simulator

export matrixcore, fresnelltools, analyticalmaterials, cmtgratings,
    utilities, materials

include("utilities.jl")
include("analyticalmaterials.jl")
include("ema.jl")
include("materials.jl")
include("matrixcore.jl")
include("fresnelltools.jl")
include("targetfigures.jl")

end # simulator
```

Code Listing C.3: src/matrixcore.jl

```
module matrixcore

export StoM, MtoS, CascadeScattering, Repeating

using LinearAlgebra


function StoM(S::Matrix{T})::Matrix{<:Number} where T <: Number
    # Converts a 2x2 scattering matrix to a transfer matrix
    # See Saleh & Teich 3.ed eq.7.1-6
    # Verified manually
    @assert(size(S) == (2, 2))
    @inbounds begin
        t12 = S[1, 1]
        r21 = S[1, 2]
        r12 = S[2, 1]
        t21 = S[2, 2]
    end
    M = [
        t12*t21 - r12*r21 r21;
        -r12              1
    ]
    (1/t21) .* M
end

function MtoS(M::Matrix{T})::Matrix{<:Number} where T <: Number
    # Converts a 2x2 transfer matrix to a scattering matrix
    # See Saleh & Teich 3.ed eq.7.1-5
    # Verified by tests as inverfse of StoM
    StoM(M)
end


function CascadeScattering(layers::Vector)::Matrix{Number}
    # Generates the total S matrix for a layered system
    # Matrixes are given in the order the elements appear, the array is inversed inside the function
    accumulated = I

    for S in view(layers, length(layers):-1:1)
        M = StoM(S)
        accumulated *= M
    end

    MtoS(accumulated)
end


function Repeating(cell::Matrix{T}, repetition::Number)::Matrix{T} where T <: Number
    # Simple helpertool to make sure gratings are handleded correctly
    # This helper helps achieve the goal that transfer matrices are only handled in this module
    M = StoM(cell)
    MtoS(M ^ repetition)
end


end # matrixcore
```

Code Listing C.4: src/fresnelltools.jl

```
module fresnelltools

using LinearAlgebra

using ..matrixcore
```

```julia

export FresnellBoundrary, FresnellSlab, Grating, ContinousBorder, ThreeLayerSystem, Interrogator, BorderInterrogator


function FresnellBoundrary(n1::Number, n2::Number)::Matrix{Number}
    # Equation from Saleh & Teich 3.ed. eq.6.2-8 & eq. 6.2-9
    # Returns the scattering matrix for the relevant border

    FresnellBoundrary(n1, n2, 0)[1]
end

function FresnellBoundrary(n1::Number, n2::Number,  1::Number)::Tuple{Matrix{Number}, Matrix{Number}, Number}
    # Equation from Saleh & Teich 3.ed. eq.6.2-8 & eq. 6.2-9
    # It is similar to the other FresnellBoundrary function, but takes incident angle into account
    # Returns the TE scattering matrix, the TM scattering matrix and the outgoing transmitted angle

    # The imaginary coefficient 1e-14im has no physical reason to exist, and is only added for stability in the program.
    # It is assumed to be needed because of type instability in the asin function i Julia.
    # See https://github.com/JuliaLang/julia/issues/24296
    if ((typeof(n1) <: Complex) & (typeof(n2) <: Real))
        complex_modifier = 1e-14im
    else
        complex_modifier = 0im
    end
    2 = asin(complex_modifier + sin (1) * n1 / n2)

    r_te_12 = (n1 * cos (1) - n2 * cos (2)) / (n1 * cos (1) + n2 * cos (2))
    r_te_21 = (n2 * cos (2) - n1 * cos (1)) / (n2 * cos (2) + n1 * cos (1))

    r_tm_12 = (n1 * sec (1) - n2 * sec (2)) / (n1 * sec (1) + n2 * sec (2))
    r_tm_21 = (n2 * sec (2) - n1 * sec (1)) / (n2 * sec (2) + n1 * sec (1))

    t_te_12 = 1 + r_te_12
    t_te_21 = 1 + r_te_21

    t_tm_12 = (1 + r_tm_12) * cos (1) / cos (2)
    t_tm_21 = (1 + r_tm_21) * cos (2) / cos (1)


    Ste = [
        t_te_12 r_te_21;
        r_te_12 t_te_21
    ]
    Stm = [
        t_tm_12 r_tm_21;
        r_tm_12 t_tm_21
    ]

    return Ste, Stm, 2
end

function FresnellSlab(n::Number, k_0::Number, d::Number, ::Number)::Matrix{Number}
    # Equation from Saleh & Teich 3.ed. eq.7.1-4
    delay = ^(-1im * n * k_0 * d * cos())
    S = [
        delay 0;
        0     delay
    ]

    if abs(S[1, 1]) < 1e-5
        S /= abs(S[1, 1])
        S *= 1e-5
    end

    S
end

function Grating(n_spechial::Number, n_normal::Number, d_spechial::Real, d_normal::Real, layers::Int, k_0::Number)::Matrix{Number}
    # A utility for creating the scattering matrix of a periodic grating

    border_normal_spechial = FresnellBoundrary(n_normal, n_spechial)
    bulk_spechial = FresnellSlab(n_spechial, k_0, d_spechial, 0)
    border_spechial_normal = FresnellBoundrary(n_spechial, n_normal)
    bulk_normal = FresnellSlab(n_normal, k_0, d_normal, 0)

    cell = CascadeScattering([border_normal_spechial, bulk_spechial, border_spechial_normal, bulk_normal])
    Repeating(cell, layers)
end
```

```julia

function Grating(n_spechial::Number, n_normal::Number, d_spechial::Real, d_normal::Real, layers::Int, k_0::Number, 1::Real)::Tuple{Matrix{Number}, Matrix{Number}}
    # A utility similar to Grating, but it takes an angle and assumes an infinitely wide grating
    # Returns the TE and TM scattering matrices for the grating
    # Unlike the angleded FresnellBoundrary tool, no angle is returned, this is because the grating assumes normal material on both sides

    n_s_te, n_s_tm, 2 = FresnellBoundrary(n_normal, n_spechial, 1)
    s_n_te, s_n_tm, _  = FresnellBoundrary(n_spechial, n_normal, 2)

    n_b = FresnellSlab(n_normal, k_0, d_normal, 1)
    s_b = FresnellSlab(n_spechial, k_0, d_spechial, 2)

    cell_te = CascadeScattering([n_s_te, s_b, s_n_te, n_b])
    cell_tm = CascadeScattering([n_s_tm, s_b, s_n_tm, n_b])

    Repeating(cell_te, layers), Repeating(cell_tm, layers)
end

function ContinousBorder(n_from::Number, n_to::Number, d::Real, k_0::Number, stepps::Int)::Matrix{Number}
    # A utility for creating a semi-continous change in refractive index
    # The gradient is approximated as linear

    components = Array{Matrix{Number}}(undef, 2 * stepps)

    n_stepp = (n_to - n_from) / stepps
    d_stepp = d / stepps
    for i in 1:stepps
        n1 = n_from + (i - 1) * n_stepp
        n2 = n1 + n_stepp

        components[2 * i - 1] = FresnellBoundrary(n1, n2)
        components[2 * i] = FresnellSlab(n2, k_0, d_stepp, 0)
    end

    CascadeScattering(components)
end

function ThreeLayerSystem(n_1::Number, n_bulk, n_3::Number, , , d)
    interface1_te, interface1_tm, 2 = FresnellBoundrary(n_1,
        n_bulk (),  )
    bulk = FresnellSlab(n_bulk (),  2* /  , d,  2)
    interface2_te, interface2_tm, _ = FresnellBoundrary(n_bulk (),
        n_3,  2)
    te_system = CascadeScattering([interface1_te, bulk,
        interface2_te])
    tm_system = CascadeScattering([interface1_tm, bulk,
        interface2_tm])

    te_system, tm_system
end

function Interrogator(layers::Vector{Function}, distances::Vector{
    Float64}, step::Float64)::Tuple{Vector{Float64}, Vector{Number}}
    # A small utility providing a default lambda for the
        Interrogator tool
    Power(Up, Um) = abs(Up + Um)^2
    Interrogator(layers, distances, step, Power)
end

function Interrogator(layers::Vector{Function}, distances::Vector{
    Float64}, step::Float64, expression::Function)::Tuple{Vector{
    Float64}, Vector{Number}}
    # A tool for inspecting the field inside a multilayer system

    total_system = Array{Matrix{Number}}(undef, length(layers))
    for i in 1:1:length(layers)
        total_system[i] = layers[i](distances[i])
    end
    U0_plus = 1
    U0_minus = (CascadeScattering(total_system) * [U0_plus, 0])[2]

    maxlen = ceil(Int, length(distances) * 2 + sum(distances) /
        step)
    d_values = Vector{Number}(undef, maxlen)
    u_values = Vector{Number}(undef, maxlen)

    d_values[1] = 0.0
    u_values[1] = expression(U0_plus, U0_minus)

    m_accumulated = I
    m_local = I
    m_partial = I
    d_accumulated = 0.0
    i = 1
    for (layer, distance) in zip(layers, distances)
        m_local = m_accumulated

        m_partial = StoM(layer(step))
        j = 0
        for _ in 0:step:distance
            i += 1
            j += 1
            d_accumulated += distance > step ? step : distance
            if (j % 1000) == 0
                m_local = StoM(layer(step * j)) * m_accumulated
            else
                m_local = m_partial * m_local
            end

            U_plus, U_minus = m_local * [U0_plus, U0_minus]
            measure = expression(U_plus, U_minus)
            d_values[i] = d_accumulated
            u_values[i] = measure
        end

        d_rest = distance % step
        if (d_rest > (step / 1000)) & (step > distance)
            m_partial = StoM(layer(d_rest))
            i += 1
            d_accumulated += d_rest
            m_local = m_partial * m_local

            (U_plus, U_minus) = m_local * [U0_plus, U0_minus]

            measure = expression(U_plus, U_minus)
            d_values[i] = d_accumulated
            u_values[i] = measure
        end

        m_accumulated = StoM(layer(distance)) * m_accumulated
    end

    u_values[1:i], d_values[1:i]
end

function BorderInterrogator(layers::Vector{Matrix{Number}}, distances::Vector{Float64}, expression::Function)::Tuple{Vector{Number}, Vector{Number}, Vector{Real}}
    # A tool for calculating the field at the borders in a structure independent of the Interrogator function
    # Returns the forward propagating fields, the backward propagating fields and the distances

    U0p = 1
    U0n = (CascadeScattering(layers) * [U0p, 0])[2]

    Upv = Vector{Number}()
    Unv = Vector{Number}()
    Dv = Vector{Float64}()

    push!(Upv, U0p)
    push!(Unv, U0n)
    push!(Dv, 0.0)

    accumulated_m = I
    accumulated_d = 0.0
    for (layer, distance) in zip(layers, distances)
        accumulated_m = StoM(layer) * accumulated_m
        accumulated_d += distance

        Up, Un = accumulated_m * [U0p, U0n]

        push!(Upv, Up)
        push!(Unv, Un)
        push!(Dv, accumulated_d)
    end

    return expression.(Upv), expression.(Unv), Dv
end


end # fresnelltools
```

Code Listing C.5: src/analyticalmaterials.jl

```
module analyticalmaterials

using ..utilities

export  _plasmone, n_drude,  _grating_coupling,  , n, hs_Palm


function  (n::Number)::Number
    real(n)^2 - imag(n)^2 + 1im * 2 * real(n) * imag(n)
end

function n (::Number)::Number√

end

function n_drude (_0::Real,  ::Real,  ::Real)::Number
```

```
     = 2 *   * c_0 / √
    _c_drude (_0,  , )
end


function _c_drude(_0::Real, ::Real, ::Real)::Number
    # See Saleh & Teich 3.ed eq.8.2-11

    _p = √(_0 / (_0 * ))
    _0 * (1 + _p^2 / (-(^2) + 1im *  / ))
end


function _plasmone(n1::Number, n2::Number, k_0::Number)::Number
    # See Maier eq. 2.14 and eq. 1.11a and b

    _1 = (n1)
    _2 = (n2)
    k_0 * √(_1 * _2 / (_1 + _2))
end

function _plasmone(ni::Number, n1::Number, n2::Number, k_0::Number)::Number
     = real(_plasmone(n1, n2, k_0))
    ki = real(ni) * k_0

    if (( / ki) > 1)
        return NaN
    end
    asin( / ki)
end


function _grating_coupling(, neff_core, n_cladding, Λ)
    asin((( * 2 / Λ) - neff_core) / n_cladding)
end

function _grating_coupling(, n_core)
    # A default for the grating in question
    _grating_coupling(, 1.4676, n_core, 1073.81e-9)
end


module hs_Palm
# An approximation of the nonlinear function h(c) such that (c) = h(c) (0) for palladium and gold-palladium alloys.
# Estimate based on linear approximation from Palm2019 and theory
```

```
form Tobiska2001.
# This module should only be used as a first approximation, not for spesific optimization.

using Interpolations
using DelimitedFiles
using Match

import ..


export h


rawdatafile034 = readdlm("materials/palm_PdAu-H/HM_Pd034.csv", ',', skipstart=2)
rawdatafile042 = readdlm("materials/palm_PdAu-H/HM_Pd042.csv", ',', skipstart=2)
rawdatafile052 = readdlm("materials/palm_PdAu-H/HM_Pd052.csv", ',', skipstart=2)
rawdatafile073 = readdlm("materials/palm_PdAu-H/HM_Pd073.csv", ',', skipstart=2)
rawdatafile100 = readdlm("materials/palm_PdAu-H/HM_Pd100.csv", ',', skipstart=2)


function loadLine(filerawdata,  ::Float64, Pd_c::Float64)::Function
    point = @match  begin
        300e-9 => 1
        600e-9 => 2
        900e-9 => 3
        1200e-9 => 4
        1500e-9 => 5
        _ => 0
    end
    offsett = 10

    raw_n = filerawdata[:, (point*2 - 1):(point*2)]
    raw_k = filerawdata[:, (point*2 - 1 + offsett):(point*2 + offsett)]

    raw_n = raw_n[(raw_n[:, 1] .!= ""), :]
    raw_k = raw_k[(raw_k[:, 1] .!= ""), :]

    sorted_n = raw_n[sortperm(raw_n[:, 1]), :]
    sorted_k = raw_k[sortperm(raw_k[:, 1]), :]

    n_estimator = LinearInterpolation(sorted_n[:, 1], sorted_n[:, 2])
    k_estimator = LinearInterpolation(sorted_k[:, 1], sorted_k[:, 2])

    h_total(c) = (n_estimator(c_to_HM(c, Pd_c)) + k_estimator(c_to_HM(c, Pd_c)) * 1im) / (sorted_n[1, 2] + sorted_k[1, 2] * 1im)

    h_total
end


function c_to_HM(H_c::Float64, Pd_c::Float64)::Float64
    # Based on Palm2019 fig. S7
    # Assumes concentration under 1 atmosphere pressure
    # Assumes the effect of a low pressure of pure hydrogen is the same as the equivalent partial pressure hydrogen in an otherwize inert atmosphere totalling 1 atmosphere
    # Assumes the relation between HM and H_c is proportional to the square root of H_c, as shown in separate note
    # H_c is the partial pressure of hydrogen in the atmosphere given in atmospheres, the approximation is only valid for H_c   0.25
    # Pd_c is the atomic alloying ratio between Pd and Au, and should be between 0.4 and 1
    # The following data was extracted from the figure:
    # 41.50974025974026 0.0011235955056179137
    # 99.9512987012987  0.6089887640449437

    HM_Pd_c025 = 1.04012484394507 * Pd_c - 0.43062952559301

    sqrt(H_c * 1.01325) * HM_Pd_c025 / sqrt(0.25)
end

function h(::Float64, H_c::Float64, Pd_c::Float64)::Number
    # H_c and Pd_c is the hydrogen concentration and the Pd atomic concentration in the Pd-Au alloy, both between 0 and 1

    filerawdata = @match Pd_c begin
        0.34 => rawdatafile034
        0.42 => rawdatafile042
        0.52 => rawdatafile052
        0.73 => rawdatafile073
        1.0 => rawdatafile100
        _ => "Illegal Pd_c"
    end

    _lower = 300e-9
    _upper = 1500e-9
    if 300e-9 <=   <= 600e-9
        _lower = 300e-9
        _upper = 600e-9
    elseif 600e-9 <=   <= 900e-9
        _lower = 600e-9
        _upper = 900e-9
    elseif 900e-9 <=   <= 1200e-9
        _lower = 900e-9
        _upper = 1200e-9
    elseif 1200e-9 <=   <= 1500e-9
        _lower = 1200e-9
        _upper = 1500e-9
    end

    h_lower = loadLine(filerawdata,  _lower, Pd_c)(H_c)
    h_upper = loadLine(filerawdata,  _upper, Pd_c)(H_c)

    _partial = ( -  _lower) / (_upper -  _lower)
    h_real = (h_upper - h_lower) *  _partial + h_lower

    h_real
end

end


end # analyticalmaterials
```

Code Listing C.6: src/materials.jl

```
module materials

using Interpolations
using DelimitedFiles


export LoadMaterial, spesifics


function LoadMaterial(path) # Loads material where complex refractive index is contained in one file
    material = readdlm(path, ',', header=true)[1][:, 1:3]

    n_data = [material[:, 1] .* 1e-6, material[:, 2]]
    k_data = [material[:, 1] .* 1e-6, material[:, 3]]
    n_real = LinearInterpolation(n_data[1], n_data[2])
    n_complex = LinearInterpolation(k_data[1], k_data[2])

    n_total () = n_real () - abs(n_complex ()) * 1im
    n_total
end

function LoadMaterial(path, number::Number) # Loads material where several complex refractive inicies are contained in one file
    material = readdlm(path, ',', skipstart=2)

    n_data = [material[:, 1] .* 1e-9, material[:, (number*2)]]
    k_data = [material[:, 1] .* 1e-9, material[:, (number*2 + 1)]]
    n_real = LinearInterpolation(n_data[1], n_data[2])
    n_complex = LinearInterpolation(k_data[1], k_data[2])

    n_total () = n_real () - abs(n_complex ()) * 1im
    n_total
end

function LoadMaterial(path_n, path_k, number) # Loads material where n and k are split and dumped by webplotdigitizer
    material_n = readdlm(path_n, ',', skipstart=2)[:, (number*2 - 1):(number*2)]
    material_k = readdlm(path_k, ',', skipstart=2)[:, (number*2 - 1):(number*2)]

    material_n = material_n[(material_n[:, 1] .!= ""), :]
    material_k = material_k[(material_k[:, 1] .!= ""), :]

    material_n = material_n[sortperm(material_n[:, 1]), :]
    material_k = material_k[sortperm(material_k[:, 1]), :]

    n_estimator = LinearInterpolation(material_n[:, 1] .* 1e-9, material_n[:, 2])
    k_estimator = LinearInterpolation(material_k[:, 1] .* 1e-9, material_k[:, 2])

    n_total () = n_estimator () - abs(k_estimator ()) * 1im
    n_total
end


module spesifics

import ..LoadMaterial
using ...analyticalmaterials
```

```


export Air, Au_Johnson, Au_Werner, Au_11nm_Rosenblatt, Au_21nm_Rosenblatt, Au_44nm_Rosenblatt, Au_Babar, Au_McPeak, Au_OlmonEvaporated, Au_OlmonSingleChrystaline, Au_OlmonTemplateStripped, Pd_Johnson, Pd_Werner, Pd_Palm_2018, Ag, LiF, SiO2_core_Sellmeier, SiO2_thinfilm_Ciprian, Au_unloaded, Pd014_unloaded, Pd034_unloaded, Pd034_loaded, Pd042_unloaded, Pd042_loaded, Pd052_unloaded, Pd052_loaded, Pd073_unloaded, Pd073_loaded, Pd_unloaded, Pd_loaded, Pd000S5, Pd100S20, Pd070S9to17, Pd041S21, Pd051S22, Pd061S23, Pd069S24, Pd064S7and18, Pd053S6, Pd074S8drude, Pd061S19, Pd061S19drude


Air () = 1


Au_11nm_Rosenblatt_estimator = LoadMaterial("materials/refractive_index/Au_11nm_Rosenblatt.csv")
Au_11nm_Rosenblatt () = Au_11nm_Rosenblatt_estimator ()

Au_21nm_Rosenblatt_estimator = LoadMaterial("materials/refractive_index/Au_21nm_Rosenblatt.csv")
Au_21nm_Rosenblatt () = Au_21nm_Rosenblatt_estimator ()

Au_44nm_Rosenblatt_estimator = LoadMaterial("materials/refractive_index/Au_44nm_Rosenblatt.csv")
Au_44nm_Rosenblatt () = Au_44nm_Rosenblatt_estimator ()

Au_Babar_estimator = LoadMaterial("materials/refractive_index/Au_Babar.csv")
Au_Babar () = Au_Babar_estimator ()

Au_McPeak_estimator = LoadMaterial("materials/refractive_index/Au_McPeak.csv")
Au_McPeak () = Au_McPeak_estimator ()

Au_OlmonEvaporated_estimator = LoadMaterial("materials/refractive_index/Au_OlmonEvaporated.csv")
Au_OlmonEvaporated () = Au_OlmonEvaporated_estimator ()

Au_OlmonSingleChrystaline_estimator = LoadMaterial("materials/refractive_index/Au_OlmonSingleChrystaline.csv")
Au_OlmonSingleChrystaline () = Au_OlmonSingleChrystaline_estimator ()

Au_OlmonTemplateStripped_estimator = LoadMaterial("materials/
refractive_index/Au_OlmonTemplateStripped.csv")
Au_OlmonTemplateStripped () = Au_OlmonTemplateStripped_estimator ()


Au_Johnson_estimator = LoadMaterial("materials/refractive_index/Au_Johnson.csv")
Au_Johnson () = Au_Johnson_estimator ()

Pd_Johnson_estimator = LoadMaterial("materials/refractive_index/Pd_Johnson.csv")
Pd_Johnson () = Pd_Johnson_estimator ()


Au_Werner_estimator = LoadMaterial("materials/refractive_index/Au_Werner.csv")
Au_Werner () = Au_Werner_estimator ()

Pd_Werner_estimator = LoadMaterial("materials/refractive_index/Pd_Werner.csv")
Pd_Werner () = Pd_Werner_estimator ()


Ag_estimator = LoadMaterial("materials/refractive_index/Ag.csv")
Ag () = Ag_estimator ()

LiF_estimator = LoadMaterial("materials/refractive_index/LiF.csv")
LiF () = LiF_estimator ()

Pd_Palm_2018_estimator = LoadMaterial("materials/refractive_index/Pd_Palm.csv")
Pd_Palm_2018 () = Pd_Palm_2018_estimator ()


# F. Downes eq.1
# Note that small wavelengths give imaginary refractive index, this gives unphysical  results
SiO2_core_Sellmeier () = √( 1
    +  0.6961663*(*1e6) ^2/((*1e6)^2-0.0684043)
    +  0.4079426*(*1e6) ^2/((*1e6)^2-0.1162414)
    +  0.8974794*(*1e6) ^2/((*1e6)^2-9.896161) )

# F. Downes eq. 2
# Note that small wavelengths give imaginary refractive index, this gives unphysical  results
SiO2_thinfilm_Ciprian () = √( 1 +  1.1336*(*1e6) ^2/((*1e6)^2-9.261e-2) )

```

```

Au_unloaded_estimator = LoadMaterial("materials/palm_PdAu-alloys/unloaded_n.csv", "materials/palm_PdAu-alloys/unloaded_k.csv", 1)
Au_unloaded () = Au_unloaded_estimator ()

Pd014_unloaded_estimator = LoadMaterial("materials/palm_PdAu-alloys/unloaded_n.csv", "materials/palm_PdAu-alloys/unloaded_k.csv", 2)
Pd014_unloaded () = Pd014_unloaded_estimator ()

Pd034_unloaded_estimator = LoadMaterial("materials/palm_PdAu-alloys/unloaded_n.csv", "materials/palm_PdAu-alloys/unloaded_k.csv", 3)
Pd034_unloaded () = Pd034_unloaded_estimator ()

Pd042_unloaded_estimator = LoadMaterial("materials/palm_PdAu-alloys/unloaded_n.csv", "materials/palm_PdAu-alloys/unloaded_k.csv", 4)
Pd042_unloaded () = Pd042_unloaded_estimator ()

Pd052_unloaded_estimator = LoadMaterial("materials/palm_PdAu-alloys/unloaded_n.csv", "materials/palm_PdAu-alloys/unloaded_k.csv", 5)
Pd052_unloaded () = Pd052_unloaded_estimator ()

Pd073_unloaded_estimator = LoadMaterial("materials/palm_PdAu-alloys/unloaded_n.csv", "materials/palm_PdAu-alloys/unloaded_k.csv", 6)
Pd073_unloaded () = Pd073_unloaded_estimator ()

Pd_unloaded_estimator = LoadMaterial("materials/palm_PdAu-alloys/unloaded_n.csv", "materials/palm_PdAu-alloys/unloaded_k.csv", 7)
Pd_unloaded () = Pd_unloaded_estimator ()


Pd034_loaded_estimator = LoadMaterial("materials/palm_PdAu-alloys/loaded_n.csv", "materials/palm_PdAu-alloys/loaded_k.csv", 3)
Pd034_loaded () = Pd034_loaded_estimator ()

Pd042_loaded_estimator = LoadMaterial("materials/palm_PdAu-alloys/loaded_n.csv", "materials/palm_PdAu-alloys/loaded_k.csv", 4)
Pd042_loaded () = Pd042_loaded_estimator ()

Pd052_loaded_estimator = LoadMaterial("materials/palm_PdAu-alloys/loaded_n.csv", "materials/palm_PdAu-alloys/loaded_k.csv", 5)
Pd052_loaded () = Pd052_loaded_estimator ()

```

```
Pd073_loaded_estimator = LoadMaterial("materials/palm_PdAu-alloys/loaded_n.csv", "materials/palm_PdAu-alloys/loaded_k.csv", 6)
Pd073_loaded () = Pd073_loaded_estimator ()

Pd_loaded_estimator = LoadMaterial("materials/palm_PdAu-alloys/loaded_n.csv", "materials/palm_PdAu-alloys/loaded_k.csv", 7)
Pd_loaded () = Pd_loaded_estimator ()



Pd000S5_estimator = LoadMaterial("materials/thesisdata/Au_Pd_Multi_sample_9_17.txt", 1)
Pd000S5 () = Pd000S5_estimator ()

Pd100S20_estimator = LoadMaterial("materials/thesisdata/Au_Pd_Multi_sample_9_17.txt", 2)
Pd100S20 () = Pd100S20_estimator ()

Pd070S9to17_estimator = LoadMaterial("materials/thesisdata/Au_Pd_Multi_sample_9_17.txt", 3)
Pd070S9to17 () = Pd070S9to17_estimator ()


Pd041S21_estimator = LoadMaterial("materials/thesisdata/Samples_21_24_B_spline_N_k.txt", 1)
Pd041S21 () = Pd041S21_estimator ()

Pd051S22_estimator = LoadMaterial("materials/thesisdata/Samples_21_24_B_spline_N_k.txt", 2)
Pd051S22 () = Pd051S22_estimator ()

Pd061S23_estimator = LoadMaterial("materials/thesisdata/Samples_21_24_B_spline_N_k.txt", 3)
Pd061S23 () = Pd061S23_estimator ()

Pd069S24_estimator = LoadMaterial("materials/thesisdata/Samples_21_24_B_spline_N_k.txt", 4)
Pd069S24 () = Pd069S24_estimator ()


Pd064S7and18_estimator = LoadMaterial("materials/thesisdata/Remaining_samples.txt", 1)
Pd064S7and18 () = Pd064S7and18_estimator ()

Pd053S6_estimator = LoadMaterial("materials/thesisdata/Remaining_samples.txt", 2)
Pd053S6 () = Pd053S6_estimator ()
```

```

Pd074S8drude_estimator = LoadMaterial("materials/thesisdata/Remaining_samples.txt", 3)
Pd074S8drude () = Pd074S8drude_estimator ()

Pd061S19_estimator = LoadMaterial("materials/thesisdata/Remaining_samples.txt", 4)
Pd061S19 () = Pd061S19_estimator ()

Pd061S19drude_estimator = LoadMaterial("materials/thesisdata/Remaining_samples.txt", 5)
Pd061S19drude () = Pd061S19_estimator ()

end # spesifics



end # materials
```

Code Listing C.7: src/ema.jl

```
module ema

using Polynomials
using GeometricalPredicates

using ..analyticalmaterials

export n_MaxwellGarnett, n_Bruggeman


function n_MaxwellGarnett (_a::Number, _b::Number, f_A::Real)::Number
    # Hans Arwin, eq. 2.23
    # Coated spheres model
    # f_A is the fill factor

    _total = _b * (_a + 2*_b + 2*f_A *(_a - _b)) / (_a + 2*_b - f_A *(_a - _b))
    n (_total)
end

function n_MaxwellGarnett (_H::Number, _is::Vector{<:Number}, f_is::Vector{<:Real})::NumberΣ
    _i = sum(f_i * (_i - _H) / (_i + 2*_H) for (_i, f_i) in zip(_is, f_is))

    _total = _H * Σ(2*_i + 1) / (1 - Σ_i)
    n (_total)
end


function n_Bruggeman (_a::Number, _b::Number, f_A::Real)::Number
    # Hans Arwin, eq. 2.26
    #   determination from Jansson & Arwin

    f_B::Real = 1 - f_A
    u::Number = (3*f_A - 1)*_a + (3*f_B - 1)*_b
    _total_1 = (u + sqrt(u^2 + 8*_a*_b)) / 4
    _total_2 = (u - sqrt(u^2 + 8*_a*_b)) / 4


    if select_Wiener_circle (_total_1, _a, _b)
        return n (_total_1)
    elseif select_Wiener_circle (_total_2, _a, _b)
        return n (_total_2)
    else
        @error "No valid solution from Wiener limits, returning best match."
        return imag (_total_1) > imag (_total_2) ? _total_1 : _total_2
    end
end

function n_Bruggeman (_a::Number, _b::Number, _c::Number, f_a::Real, f_b::Real, f_c::Real)::Number
    # Solved by hand in journal page 37 from Hans Arwin

    k_0 = _a * _b * _c * (f_a + f_b + f_c)
    k_1 = 0
    k_2 = 0
    k_3 = 0

    for (f_i, _i, _2, _3) in ((f_a, _a, _b, _c), (f_b, _b, _a, _c), (f_c, _c, _b, _b))
        k_1 += f_i * _i * _2 * 2 + f_i * _i * _3 * 2 - f_i * _2 * _3
        k_2 += f_i * _i * 4 - f_i * _2 * 2 - f_i * _3 * 2
        k_3 -= 4 * f_i
    end

    _total_1, _total_2, _total_3 = roots(Polynomial([k_0, k_1, k_2, k_3]))

    for _potential in (_total_1, _total_2, _total_3)
```

```
        if select_Wiener_polygon(_potential, [_a, _b, _c])
            return _potential
        elseif (select_Wiener_circle(_potential, _a, _b) | select_Wiener_circle(_potential, _a, _c) | select_Wiener_circle(_potential, _b, _c))
            return _potential
        else
            @error "No valid solution from Wiener limits, returning best matach."
            return 0
        end
    end
end


function select_Wiener_circle(_s::Number, _1::Number, _2::Number)::Bool
    # Check if on line with origo, Jansson & Arwin algorithm

    if (real(_1) * imag(_2) - real(_2) * imag(_1)) < 1e-8 * abs(_1)
        function w(z::Number)::Number
            z * conj(_2 - _1) / abs(_2 - _1)
        end
        w_s, w_1, w_2 = w(_s), w(_1), w(_2)

        if real(w_2) < real(w_1)
            w_s, w_1, w_2 = w_s * -1, w_1 * -1, w_2 * -1
        end

        return (imag(w_s) + w_1   w_1) & (real(w_1) <= real(w_s) + 1e-8) & (real(w_s) <= real(w_2) + 1e-8)
    end

    z_0 = _1 * _2 * (conj(_1) - conj(_2)) / (conj(_1) * _2 - _1 * conj(_2))

    function  (z::Number)::Number
        (z - z_0) * conj(_2 - _1) / (abs(z_0) * abs(_2 - _1))
    end
    _s, _1, _2 =  (_s),  (_1),  (_2)

    if imag((_1 + _2)/2) < 0
        _s, _1, _2 = _s * -1, _1 * -1, _2 * -1
    end

    return (abs(_s) <= 1 + 1e-8) & (imag(_s) >= imag((_1 + _2)/2)
        + 1e-8)
end

function select_Wiener_polygon(_s::Number, _v::Vector{<:Number})::Bool
    # Geometrical limits from Jansson & Arwin, solution with modern tools

    polygonpoints = Vector{Point2D}()
    for   in _v
        push!(polygonpoints, Point(real (), imag ()))
    end
    poly = Polygon(polygonpoints...)

    inpolygon(poly, Point(real (_s), imag (_s)))
end

end # ema
```

Code Listing C.8: src/targetfigures.jl

```
module targetfigures

using ..matrixcore
using ..fresnelltools
using ..analyticalmaterials.hs_Palm

export scan_singleparameter, scan_plasmon_singleparameter, scan_plasmon_dualparameter, scan_plasmon_dualthicknesses, make_layered_tm_system, make_layered__tm_system, make_d3_system, make_d4_system, make_d2_dualparameter_system, make_hc_system


function scan_singleparameter(system::Function, scanrange::Vector{<:Number}, postprocessing::Function=v -> abs(v[2, 1])^2)::Tuple{Vector{<:Number}, Vector}
    # This function provides a unified interface for scaning structures for different design parameters.
    # It akes a parameter "system", a function returning (for instance) a total scattering matrix, a scans it for all elements in the scanrange and a postprocessing expression.
    # Returns the scanrange and the output data.
    # The default postprocessing stage takes a scattering matrix and returns the power reflection coefficient.

    output = Vector(undef, length(scanrange))

    for i in 1:length(scanrange)
```

```julia
        instance = system(scanrange[i])
        output[i] = postprocessing(instance)
    end

    scanrange, output
end


function plasmon_minima(system::Function, s::Vector{Float64}, predipps::Int64=1)::Tuple{Float64, Float64}
    # This function finds the plasmon minima, as well as the relevant frequency at which it appears.
    # It takes as its arguments a function returning a scattering matrix as a function of a wavelength, a range of wavelengths and a number of predipps.
    # The predipps number is the number of local maxima the function can encounter before searching for a global minima.
    # This is useful for situations in which the plasmon dip is weaker than the high-frequency (low wavelength) absorption of the material.
    # Note however that, if set too high, the algorithm might measure a local minima pas the plasmon dip instead.
    # The function returns the plasmon minima power reflection and the wavelength at which it was found.

    min = s[1]
    minr = abs(system( min)[2, 1])^2
    initialrise = predipps
    thisr = 1

    for  in s
        lastr = thisr
        thisr = abs(system ()[2, 1])^2

        if (initialrise >= 1) && (thisr < lastr)
            initialrise -= 1
            minr = thisr
            min =
        end

        if thisr < minr
            minr = thisr
            min =
        end
    end

    minr, min
```

```
end

function plasmon_halfwidth(system::Function, s::Vector{Float64}, predipps::Int64=1, widthratio::Float64=0.5)::Tuple{Float64, Float64, Float64, Float64, Float64}
    # This function finds the plasmon dip full width at half maximum (half can be changed by widthratio parameter).
    # It opperates similarly to the plasmon_minima function.

    leftmax = 0.0

    left = 0.0
    right = 0.0
    min = 0.0

    minr = 1.0
    maxr = 0.0

    initialrise = predipps
    thisr = abs(system (s[1])[2, 1])^2

    for  in s
        lastr = thisr
        thisr = abs(system ()[2, 1])^2

        if initialrise >= 1
            if thisr < lastr
                initialrise -= 1
                minr = thisr
                min = 
            end

            if thisr > maxr
                maxr = thisr
               leftmax = 
            end
        end

        if thisr < minr
            minr = thisr
            min = 
        end

    end

    rthreshold = minr + ((maxr - minr) * widthratio)

    for   in  s
        if   >=  leftmax
            thisr = abs(system ()[2, 1])^2

            if (thisr > rthreshold) &&  (  <=  min)
                left =
            end

            if (thisr < rthreshold) &&  (  >=  min)
                right =
            end

            if (thisr > rthreshold) &&  (  >=  min)
                break
            end

        end
    end

    width =  right -  left

    width, maxr - minr, left, right, min
end


function scan_minima(xs::Vector{<:Number}, ys::Vector{<:Number})::
    Tuple{Number, Number}
    # An optimized function for finding the minimum of a curve.

    @assert(length(xs) == length(ys))
    @assert(length(xs) > 0)

    @inbounds begin
        xmin = xs[1]
        ymin = ys[1]

        for i in 1:length(xs)
            if ys[i] < ymin
                xmin = xs[i]
                ymin = ys[i]
            end
        end
    end

    xmin, ymin
end

```

```julia
function scan_maxima(xs::Vector{<:Number}, ys::Vector{<:Number})::Tuple{Number, Number}
    # An optimized function for finding the maximum of a curve.

    @assert(length(xs) == length(ys))
    @assert(length(xs) > 0)

    @inbounds begin
        xmax = xs[1]
        ymax = ys[1]

        for i in 1:length(xs)
            if ys[i] > ymax
                xmax = xs[i]
                ymax = ys[i]
            end
        end
    end

    xmax, ymax
end


function scan_plasmon_singleparameter(system::Function, scanrange::Vector{<:Number}, s::Vector{Float64}, predipps::Int64=1)::Tuple{Vector{<:Number}, Vector{Float64}, Vector{Float64}, Vector{Float64}, Vector{Float64}}
    # A tool for running a single parameter plasmon scan.
    # Takes the following arguments:
    #   system: A function of the scanparameter returning a function on the form  -> S.
    #   scanrange: A range of scan parameters.
    #    s: A range of wavelengths.
    #   predipps: The number of local maxima in the reflection spectrum to scan through before finding the plasmon minima.
    # Returns the scanrange, the minimum reflection coefficients and the relevant wavelengths.
    # Functions similarly to the single thickness scan one-off written in demos/thickness.jl.

    plasmonfinder(subsystem) = plasmon_minima(subsystem, s, predipps)
    plasmonwidthfinder(subsystem) = plasmon_halfwidth(subsystem, s, predipps)

    _, plasmon_minima_output = scan_singleparameter(system, scanrange, plasmonfinder)
    _, plasmon_width_output = scan_singleparameter(system, scanrange, plasmonwidthfinder)

    scanrange, [x[1] for x in plasmon_minima_output], [x[2] for x in plasmon_minima_output], [z[1] for z in plasmon_width_output], [z[2] for z in plasmon_width_output]
end


function scan_plasmon_dualparameter(system::Function, parameter1s::Vector{<:Number}, parameter2s::Vector{<:Number},  s::Vector{Float64}, predipps::Int64=1)::Tuple{Vector{<:Number}, Vector, Vector}
    # A tool for scaning for plasmon peaks across two parameters.
    # The function runs the single parameter scan on parameter 2 for each of parameter 1, it then plots the optimal parameter 2 as a function of parameter 1.

    minRP2s = Vector(undef, length(parameter1s))
    minP2s = Vector(undef, length(parameter1s))

    for i in 1:length(parameter1s)
        p1 = parameter1s[i]

        _, reflections, wavelengths, _, _ = scan_plasmon_singleparameter(system(p1), parameter2s,  s, predipps)
        minRP2, _ = scan_minima(parameter2s, reflections)
        minP2, _ = scan_minima(parameter2s, wavelengths) # TODO this might be a suboptimal target function, try to find something better based on the plateaus observed.

        minRP2s[i] = minRP2
        minP2s[i] =  minP2
    end

    parameter1s, minRP2s,  minP2s
end

function scan_plasmon_dualthicknesses(system::Function, d1min, d1max, d1step, d2min, d2max, d2step,  s::Vector{Float64}, predipps::Int64=1, layertolerance::Float64=0.5)::Tuple{Vector{<:Number}, Vector, Vector, Vector, Vector}
    # A tool for scaning for plasmon peaks across two parameters.
    # The function runs the single parameter scan on parameter 2 for each of parameter 1, it then plots the optimal parameter 2 as a function of parameter 1.

    parameter1s = [x for x in d1min:d1step:d1max]

    minRP2s = Vector(undef, length(parameter1s))
    minP2s = Vector(undef, length(parameter1s))
    minFWHMP2s = Vector(undef, length(parameter1s))Δ
    maxRP2s = Vector(undef, length(parameter1s))

    for i in 1:length(parameter1s)
        p1 = parameter1s[i]

        parameter2s = [x for x in d2min:d2step:min(d2max, p1 * layertolerance)]

        _, reflections, wavelengths, peakwidths, peakdeltas = scan_plasmon_singleparameter(system(p1), parameter2s, s, predipps)
        minRP2, _ = scan_minima(parameter2s, reflections)
        minP2, _ = scan_minima(parameter2s, wavelengths) # TODO this might be a suboptimal target function, try to find something better based on the plataus observed.
        minFWHMP2, _ = scan_minima(parameter2s, peakwidths)Δ
        maxRP2, _ = scan_maxima(parameter2s, peakdeltas)

        minRP2s[i] = minRP2
        minP2s[i] =  minP2
        minFWHMP2s[i] = minFWHMP2Δ
        maxRP2s[i] = ΔmaxRP2
    end

    parameter1s, minRP2s,  minP2s, minFWHMP2s, ΔmaxRP2s
end



function make_layered_tm_system(ns::Vector{Function}, ds::Vector{Float64}, i::Float64)::Function
    # A tool for generating  -> S functions for arbitrary layered systems.

    @assert(length(ns) == length(ds))

    function S ()
        w = i

        elements = Vector(undef, 2 * length(ns) - 2)

        for i in 1:(length(ns) - 1)
            bulk = FresnellSlab(ns[i ](), 2*/, ds[i], w)
            _, border, w = FresnellBoundrary(ns[i ](), ns[i + 1](), w)

            elements[2*i - 1] = bulk
            elements[2*i] = border
        end

        CascadeScattering(elements)
    end

    S
end


function make_layered__tm_system(ns::Vector{Function}, ds::Vector{Float64}, ::Float64)::Function
    # A tool for generating  -> S functions for arbitrary layered systems.

    @assert(length(ns) == length(ds))

    function S (i)
        w = i

        elements = Vector(undef, 2 * length(ns) - 2)

        for i in 1:(length(ns) - 1)
            bulk = FresnellSlab(ns[i ](), 2*/, ds[i], w)
            _, border, w = FresnellBoundrary(ns[i ](), ns[i + 1](), w)

            elements[2*i - 1] = bulk
            elements[2*i] = border
        end

        CascadeScattering(elements)
    end

    S
end


function make_d3_system(n1::Function, n2::Function, n3::Function, n4::Function, n5::Function, d1::Float64, d2::Float64, d4::Float64, d5::Float64, i::Float64)::Function
```

```julia
    function tm_singleparameter_system(d3::Float64)::Function
        make_layered_tm_system([n1, n2, n3, n4, n5], [d1, d2, d3, d4, d5], i)
    end

    tm_singleparameter_system
end
function make_d4_system(n1::Function, n2::Function, n3::Function, n4::Function, n5::Function, d1::Float64, d2::Float64, d3::Float64, d5::Float64, i::Float64)::Function
    function tm_singleparameter_system(d4::Float64)::Function
        make_layered_tm_system([n1, n2, n3, n4, n5], [d1, d2, d3, d4, d5], i)
    end

    tm_singleparameter_system
end
function make_d2_dualparameter_system(n1::Function, n2::Function, n3::Function, n4::Function, n5::Function, d1::Float64, d4::Float64, d5::Float64, i::Float64)::Function
    function tm_dualparameter_system(d2::Float64)::Function
        make_d3_system(n1, n2, n3, n4, n5, d1, d2, d4, d5, i)
    end

    tm_dualparameter_system
end
function make_hc_system(n1::Function, n2::Function, n3::Function, n4::Function, n5::Function, d1::Float64, d2::Float64, d3::Float64, d4::Float64, d5::Float64, i::Float64, Pd_c::Float64)::Function
    function tm_singleparameter_system(H_c::Float64)::Function

        function modified_n3 (::Float64)::Number√
            (h (, H_c, Pd_c) * n3 ()^2)
        end

        make_layered_tm_system([n1, n2, modified_n3, n4, n5], [d1, d2, d3, d4, d5], i)
    end

    tm_singleparameter_system
end


end # targetfigures
```

Appendix D

Simulation script

Listing D.1 provides an example of how the design-parameter scan is intended to be used. This script runs the structure simulations presented in this thesis.

Code Listing D.1: demos/thesissimulations.jl

```
using Plots

using simulator.materials.spesifics
using simulator.targetfigures
using simulator.analyticalmaterials
using simulator.ema


# Simulation parameters start

save = true

n1 = SiO2_core_Sellmeier
n2 = Pd000S5
n3 = Pd053S6
n4 = Air
n4_ema () = n_Bruggeman ((n3 ()), (Air ()), 0.5)
n5 = Air

d1 = 10e-9
d2 = 30e-9
d3 = 10e-9
d4 = 0e-9
d5 = 10e-9

d2smax = 40e-9
d3smax = 40e-9
d4smax = 5e-9
```

```
d2ss = 1e-9
d3ss = 1e-10
d4ss = 1e-10
d3smrd2sm = 0.4
d4smrd3sm = 0.1

 = 1200e-9
s = [x for x in 800e-9:1e-10:1400e-9]
predipps = 2

Hcs = [x for x in 1e-2:1e-3:10e-2]
Pd_c = 0.52

n3sd = [Pd000S5, Pd041S21, Pd051S22, Pd053S6, Pd061S19,
   Pd064S7and18, Pd069S24, Pd070S9to17, Pd074S8drude, Pd100S20]
n3sh = [Pd034_unloaded, Pd042_unloaded, Pd052_unloaded,
   Pd073_unloaded, Pd_unloaded]
Pd_cs_d = [0.00, 0.41, 0.51, 0.53, 0.61, 0.64, 0.69, 0.70, 0.74,
   1.00]
Pd_cs_h = [0.34, 0.42, 0.52, 0.73, 1.00]
Pd_d3s = [d3, d3, d3, d3, d3]

d = 45
ds = [x for x in 40:0.001:60]

fig_path = "../rapport/figures/thesisfigures_staged/"
batch = ""

# Simulation parameters end

if save
    gr()
else
    plotly()
end
1 = d *   / 180
s = ds .* ( / 180)



# Run thickness simulations
println("\nScan d3:")
@time begin
    system_d3 = make_d3_system(n1, n2, n3, n4, n5, d1, d2, d4, d5,
        1)
    d3_xs, d3_ysr, d3_ys, d3_ysw, Δd3_ys =
       scan_plasmon_singleparameter(system_d3, [x for x in 0:d3ss:
        min(d3smax, d2*d3smrd2sm)], s, predipps)
end
d3_xsnm = d3_xs *= 1e9
fig_d3r = plot(d3_xsnm, d3_ysr, xaxis="Palladium thickness [nm]",
    yaxis="Power reflection coefficient minimum", legend=false, tiks
    =:native)
fig_d3 = plot(d3_xsnm, d3_ys .* 1e9, xaxis="Palladium thickness [
    nm]", yaxis="Dip wavelength [nm]", legend=false, tiks=:native)
fig_d3w = plot(d3_xsnm, d3_ysw .* 1e9, xaxis="Palladium thickness [
    nm]", yaxis="Dip full width half maximum [nm]", legend=false,
    tiks=:native)Δ
fig_d3 = plot(d3_xsnm, Δd3_ys, xaxis="Palladium thickness [nm]",
    yaxis="Dip peak to peak", legend=false, tiks=:native)
if save
    savefig(fig_d3r, fig_path * "d3r" * batch * ".pdf")
    savefig( fig_d3, fig_path * "d3l" * batch * ".pdf")
    savefig(fig_d3w, fig_path * "d3w" * batch * ".pdf")
    savefig(Δfig_d3, fig_path * "d3d" * batch * ".pdf")
else
    display(fig_d3r)
    display( fig_d3)
    display(fig_d3w)
    display(Δfig_d3)
end


println("\nScan d2 and d3:")
@time begin
    system_d23 = make_d2_dualparameter_system(n1, n2, n3, n4, n5,
        d1, d4, d5, 1)
    d23_xs, d23_ysr, d23_ys, d23_ysw, Δd23_ys =
        scan_plasmon_dualthicknesses(system_d23, 0, d2smax, d2ss, 0,
         d3smax, d2ss, s, predipps, d3smrd2sm)
end
d23_xsnm = d23_xs .* 1e9
fig_d23r = plot(d23_xsnm, d23_ysr .* 1e9, xaxis="Gold thickness [nm
    ]", yaxis="Palladium thickness [nm]", legend=false, tiks=:native
    )
fig_d23w = plot(d23_xsnm, d23_ysw .* 1e9, xaxis="Gold thickness [nm
    ]", yaxis="Palladium thickness [nm]", legend=false, tiks=:native
    )Δ
fig_d23 = plot(d23_xsnm, Δd23_ys .* 1e9, xaxis="Gold thickness [nm]
    ", yaxis="Palladium thickness [nm]", legend=false, tiks=:native)
if save
    savefig(fig_d23r, fig_path * "d23r" * batch * ".pdf")
    savefig(fig_d23w, fig_path * "d23w" * batch * ".pdf")
    savefig(Δfig_d23, fig_path * "d23d" * batch * ".pdf")
```

```julia
else
    display(fig_d23r)
    display(fig_d23w)
    display(Δfig_d23)
end



# Run demo of structure with and without gold
println("\nSingle plasmon with without gold")
@time begin
    system_with = make_layered_tm_system([n1, n2, n3, n4, n5], [d1, 30e-9, 10e-9, d4, d5], 1)
    system_without = make_layered_tm_system([n1, n2, n3, n4, n5], [d1, 0e-9, 10e-9, d4, d5], 1)
    with_xs, with_ys = scan_singleparameter(system_with, s)
    without_xs, without_ys = scan_singleparameter(system_without, s)

    system_with_ = make_layered__tm_system([n1, n2, n3, n4, n5], [d1, 30e-9, 10e-9, d4, d5], )
    system_without_ = make_layered__tm_system([n1, n2, n3, n4, n5], [d1, 0e-9, 10e-9, d4, d5], )
    with_xs, with_ys = scan_singleparameter( system_with_, s)
    without_xs, without_ys = scan_singleparameter( system_without_, s)
end
fig_with = plot(with_xs .* 1e9, with_ys, title="", xaxis="Incident wavelength [nm]", yaxis="Reflection coefficient", label="10 nm palladium on 30 nm gold", legend=:bottomright, tiks=:native)
plot!(fig_with, without_xs .* 1e9, without_ys, label="10 nm palladium")
fig_with_ = plot (ds, with_ys, title="", xaxis="Incident angle [degrees]", yaxis="Reflection coefficient", label="10 nm palladium on 30 nm gold", legend=:right, tiks=:native)
plot!( fig_with_, ds, without_ys, label="10 nm palladium")
if save
    savefig(fig_with, fig_path * "withwithout" * batch * ".pdf")
    savefig( fig_with_, fig_path * "withwithouttheta" * batch * ".pdf")
else
    display(fig_with)
    display( fig_with_)
end



```

```
# Run hydrogen simulation
println("\nScan hydrogen:")
@time begin
    system_Hc = make_hc_system(n1, n2, n3, n4, n5, d1, d2, d3, d4, d5, 1, Pd_c)
    Hc_xs, Hc_ysr, Hc_ys, Hc_ysw, ΔHc_ys = scan_plasmon_singleparameter(system_Hc, Hcs, s, predipps)
end
Hc_xsp = Hc_xs .* 1e2
fig_Hcr = plot(Hc_xsp, Hc_ysr, xaxis="Hydrogen concentration [%]", yaxis="Power reflection coefficient minimum", legend=false, tiks=:native)
fig_Hc = plot(Hc_xsp, Hc_ys .* 1e9, xaxis="Hydrogen concentration [%]", yaxis="Dip wavelength [nm]", legend=false, tiks=:native)
fig_Hcw = plot(Hc_xsp, Hc_ysw .* 1e9, xaxis="Hydrogen concentration [%]", yaxis="Dip full width half maximum [nm]", legend=false, tiks=:native)Δ
fig_Hc = plot(Hc_xsp, ΔHc_ys, xaxis="Hydrogen concentration [%]", yaxis="Dip peak to peak", legend=false, tiks=:native)
if save
    savefig(fig_Hcr, fig_path * "Hcr" * batch * ".pdf")
    savefig( fig_Hc, fig_path * "Hcl" * batch * ".pdf")
    savefig(fig_Hcw, fig_path * "Hcw" * batch * ".pdf")
    savefig(Δfig_Hc, fig_path * "Hcd" * batch * ".pdf")
else
    display(fig_Hcr)
    display( fig_Hc)
    display(fig_Hcw)
    display(Δfig_Hc)
end



# Run alloy comparisons
println("\nScan alloys d3:")
fig_alloys_d3r = plot(xaxis="Palladium thickness [nm]", yaxis="Power reflection coefficient minimum", legend=:topleft, tiks=:native)
fig_alloys_d3 = plot(xaxis="Palladium thickness [nm]", yaxis="Dip wavelength [nm]", legend=:topleft, tiks=:native)
fig_alloys_d3w = plot(xaxis="Palladium thickness [nm]", yaxis="Dip full width half maximum [nm]", legend=:bottomright, tiks=:native)Δ
fig_alloys_d3 = plot(xaxis="Palladium thickness [nm]", yaxis="Dip peak to peak", legend=:topright, tiks=:native)
@time begin
    for (n3_local, Pd_c_local) in zip(n3sd, Pd_cs_d)
```

```
        @show Pd_c_local
        system_alloysd3 = make_d3_system(n1, n2, n3_local, n4, n5, d1, d2, d4, d5, 1)
        alloysd3_xs, alloysd3_ysr, alloysd3_ys, alloysd3_ysw, Δalloysd3_ys = scan_plasmon_singleparameter(system_alloysd3, [x for x in 0:d3ss:min(d3smax, d2*d3smrd2sm)], s, predipps)

        alloysd3_xsnm = alloysd3_xs .* 1e9
        plot!(fig_alloys_d3r, alloysd3_xsnm, alloysd3_ysr, label=(string(Int(Pd_c_local * 100)) * "% palladium"))
        plot!( fig_alloys_d3, alloysd3_xsnm, alloysd3_ys .* 1e9 , label=(string(Int(Pd_c_local * 100)) * "% palladium"))
        plot!(fig_alloys_d3w, alloysd3_xsnm, alloysd3_ysw .* 1e9 , label=(string(Int(Pd_c_local * 100)) * "% palladium"))
        plot!(Δfig_alloys_d3, alloysd3_xsnm, Δalloysd3_ys, label=(string(Int(Pd_c_local * 100)) * "% palladium"))
    end
end
if save
    savefig(fig_alloys_d3r, fig_path * "alloys_d3r" * batch * ".pdf")
    savefig( fig_alloys_d3, fig_path * "alloys_d3l" * batch * ".pdf")
    savefig(fig_alloys_d3w, fig_path * "alloys_d3w" * batch * ".pdf")
    savefig(Δfig_alloys_d3, fig_path * "alloys_d3d" * batch * ".pdf")
else
    display(fig_alloys_d3r)
    display( fig_alloys_d3)
    display(fig_alloys_d3w)
    display(Δfig_alloys_d3)
end


println("\nSingle plasmon in d3 scan")
@time begin
    system_d3_dip = make_layered_tm_system([n1, n2, Pd041S21, n4, n5], [d1, d2, 5e-9, d4, d5], 1)
    system_d3_nodip = make_layered_tm_system([n1, n2, Pd041S21, n4, n5], [d1, d2, 10e-9, d4, d5], 1)
    d3dip_xs, d3dip_ys = scan_singleparameter(system_d3_dip, s)
    d3nodip_xs, d3nodip_ys = scan_singleparameter(system_d3_nodip, s)

    system_d3_dip_ = make_layered__tm_system([n1, n2, Pd041S21, n4, n5], [d1, d2, 5e-9, d4, d5], )
    system_d3_nodip_ = make_layered__tm_system([n1, n2, Pd041S21, n4, n5], [d1, d2, 10e-9, d4, d5], )
    d3dip_xs, d3dip_ys = scan_singleparameter(system_d3_dip_, s)
    d3nodip_xs, d3nodip_ys = scan_singleparameter(system_d3_nodip_, s)
end
fig_d3dip = plot(d3dip_xs .* 1e9, d3dip_ys, xaxis="Incident wavelength [nm]", yaxis="Reflection coefficient", label="5 nm palladium alloy", legend=:bottomright, tiks=:native)
plot!(fig_d3dip, d3nodip_xs .* 1e9, d3nodip_ys, label="10 nm palladium alloy")
fig_d3dip_ = plot (ds, d3dip_ys, xaxis="Incident angle [degrees]", yaxis="Reflection coefficient", label="5 nm palladium alloy", legend=:bottomright, tiks=:native)
plot!(fig_d3dip_, ds, d3nodip_ys, label="10 nm palladium alloy")
if save
    savefig(fig_d3dip, fig_path * "d3dip" * batch * ".pdf")
    savefig(fig_d3dip_, fig_path * "d3diptheta" * batch * ".pdf")
else
    display(fig_d3dip)
    display(fig_d3dip_)
end


println("\nScan alloys hydrogen:")
fig_alloys_Hcr = plot(xaxis="Hydrogen concentration [%]", yaxis="Power reflection coefficient minimum", legend=:bottomright, tiks=:native)
fig_alloys_Hc = plot(xaxis="Hydrogen concentration [%]", yaxis="Dip wavelength [nm]", legend=:bottomright, tiks=:native)
fig_alloys_Hcw = plot(xaxis="Hydrogen concentration [%]", yaxis="Dip full width half maximum [nm]", legend=:bottomright, tiks=:native)Δ
fig_alloys_Hc = plot(xaxis="Hydrogen concentration [%]", yaxis="Dip peak to peak", legend=:bottomright, tiks=:native)
@time begin
    for (n3_local, Pd_c_local, d3_local) in collect(zip(n3sh, Pd_cs_h, Pd_d3s))[2:length(n3sh)]
        @show Pd_c_local
        system_alloys_Hc = make_hc_system(n1, n2, n3_local, n4, n5, d1, d2, d3_local, d4, d5, 1, Pd_c_local)
        alloysHc_xs, alloysHc_ysr, alloysHc_ys, alloysHc_ysw, ΔalloysHc_ys = scan_plasmon_singleparameter(system_alloys_Hc, Hcs, s, predipps)

        alloysHc_xsp = alloysHc_xs .* 1e2
        plot!(fig_alloys_Hcr, alloysHc_xsp, alloysHc_ysr, label=(string(Int(Pd_c_local * 100)) * "% palladium"))
        plot!( fig_alloys_Hc, alloysHc_xsp, alloysHc_ys .* 1e9 , label=(string(Int(Pd_c_local * 100)) * "% palladium"))
        plot!(fig_alloys_Hcw, alloysHc_xsp, alloysHc_ysw .* 1e9 , label=(string(Int(Pd_c_local * 100)) * "% palladium"))
        plot!(Δfig_alloys_Hc, alloysHc_xsp, ΔalloysHc_ys, label=(string(Int(Pd_c_local * 100)) * "% palladium"))
    end
end
if save
    savefig(fig_alloys_Hcr, fig_path * "alloys_Hcr" * batch * ".pdf")
    savefig( fig_alloys_Hc, fig_path * "alloys_Hcl" * batch * ".pdf")
    savefig(fig_alloys_Hcw, fig_path * "alloys_Hcw" * batch * ".pdf")
    savefig(Δfig_alloys_Hc, fig_path * "alloys_Hcd" * batch * ".pdf")
else
    display(fig_alloys_Hcr)
    display( fig_alloys_Hc)
    display(fig_alloys_Hcw)
    display(Δfig_alloys_Hc)
end



# Run roughness simulations
println("\nScan d4 roughness:")
@time begin
    system_d4_ema = make_d4_system(n1, n2, n3, n4_ema, n5, d1, d2, d3, d5,  1)
    d4_ema_xs, d4_ema_ysr, d4_ema_ys, d4_ema_ysw, Δd4_ema_ys = scan_plasmon_singleparameter(system_d4_ema, [x for x in 0:d4ss:min(d4smax, d3*d4smrd3sm)], s, predipps)
end
d4_ema_xsnm = d4_ema_xs *= 1e9
fig_d4_emar = plot(d4_ema_xsnm, d4_ema_ysr, xaxis="Roughness ema thickness [nm]", yaxis="Power reflection coefficient minimum", legend=false, tiks=:native)
fig_d4_ema = plot(d4_ema_xsnm, d4_ema_ys .* 1e9, xaxis="Roughness ema thickness [nm]", yaxis="Dip wavelength [nm]", legend=false, tiks=:native)
fig_d4_emaw = plot(d4_ema_xsnm, d4_ema_ysw .* 1e9, xaxis="Roughness ema thickness [nm]", yaxis="Dip full width half maximum [nm]",
   legend=false, tiks=:native)Δ
fig_d4_ema = plot(d4_ema_xsnm, Δd4_ema_ys, xaxis="Roughness ema
   thickness [nm]", yaxis="Dip peak to peak", legend=false, tiks=:
   native)
if save
    savefig(fig_d4_emar, fig_path * "d4_emar" * batch * ".pdf")
    savefig( fig_d4_ema, fig_path * "d4_emal" * batch * ".pdf")
    savefig(fig_d4_emaw, fig_path * "d4_emaw" * batch * ".pdf")
    savefig(Δfig_d4_ema, fig_path * "d4_emad" * batch * ".pdf")
else
    display(fig_d4_emar)
    display( fig_d4_ema)
    display(fig_d4_emaw)
    display(Δfig_d4_ema)
end
```

Appendix E

Profilometer leveler

Listing E.1 shows a simple stylus scan leveling script made to analyze the data from section 3.5.

Code Listing E.1: other_demos/profilometer_plotter.jl

```julia
using Plots
using DelimitedFiles

save = false


if save
    gr()
else
    plotly()
end


function LoadStylus(path, flattening)
    data = readdlm(path, ',', skipstart=34)[:, 1:2]

    correction = data[flattening, 2]
    for i in 1:length(data[:, 2])
        data[i, 2] = data[i, 2] - correction * i / flattening
    end

    data[:, 1], data[:, 2]
end


stylusfig = plot(title="Stylus measurements", xlabel="distance [m]", ylabel="height [nm]", ticks=:native)
for sample in ["5", "6", "7", "8", "9", "10", "11", "12", "13", "14", "15", "16", "17", "18", "19", "20"]
```

```
    xs, ys = LoadStylus("../NanoLab_process/20220405/S" * sample * ".csv", 16000)

    plot!(stylusfig, xs, ys, label=sample)
end
display(stylusfig)
if save savefig(stylusfig, "../../rapport/figures/profiles/stylus.pdf") end


stylusfig_contested = plot(title="Stylus measurements", xlabel="distance [m]", ylabel="height [nm]", ticks=:native)
for sample in [
        "20220405/S5", "20220424/5A",
        "20220405/S8", "20220424/8A",
        "20220405/S9", "20220424/9A",
        "20220405/S10", "20220424/10A",
        "20220405/S12", "20220424/12A",
        "20220405/S13", "20220424/13A",
        "20220405/S14", "20220424/14A",
        "20220405/S18", "20220424/18A",
        "20220405/S19", "20220424/19A",
        "20220405/S20", "20220424/20A",
        ]
    xs, ys = LoadStylus("../NanoLab_process/" * sample * ".csv", 15500)

    plot!(stylusfig_contested, xs, ys, label=sample)
end
display(stylusfig_contested)
if save savefig(stylusfig_contested, "../../rapport/figures/profiles/stylus.pdf") end


stylusfig_extra = plot(title="Stylus measurements", xlabel="distance [m]", ylabel="height [nm]", ticks=:native)
for sample in ["21", "22", "23", "24"]
    xs, ys = LoadStylus("../NanoLab_process/20220505/" * sample * "A.csv", 8000)

    plot!(stylusfig_extra, xs, ys, label=sample)
end
display(stylusfig_extra)
if save savefig(stylusfig_extra, "../../rapport/figures/profiles/stylus.pdf") end
```

Torstein Nordgård-Hansen

Optical Properties of Palladium Gold Alloy Thin Films and Modeling of Their Use in Hydrogen Sensors

NTNU
Norwegian University of
Science and Technology