NTNU

Department of Engineering Cybernetics

Prestudy into the applicability of using a robotic arm in an O&M telepresence system, operating inside a nacelle

Author:
Eivind Berntsen
eivinbe@stud.ntnu.no

Supervisor:
Assoc. Prof. Amund Skavhaug
Amund.Skavhaug@itk.ntnu.no

December 20, 2011

Abstract

This report is concerned with how a robotic arm, mounted on a set of rails, inside a nacelle can be of help, and what challenges comes with this.

The rational for using a robotic arm is mainly that it can help with inspection but it may possibly be of help with maintenance and repair.

The main work for finding answers to the questions that arise when it comes to this system has been a literature review, but some work has also gone into 3D modeling.

One of the topics discussed is telepresence. The main findings with concern to this is that you have to have a good user interface to make such a system work well. The optimal solution should have some elements of augmented virtuality.

Another topic is collision avoidance. A system to handle this is essential for safe operation of the arm. Geometry data from a 3D model of the environment is a good way to solve this problem, and the same data would probably be usable in a augmented virtuality system.

The Kinect from Microsoft is here proposed as a possible source of this 3D data.

Acknowledgments

I'd like to first and foremost thank my supervisor Amund Skavhaug for both his guidance and the freedom I have had in this project. It is good to finally set the agenda for what needs to be done and take things in my one phase.

I must also thank Øyvind Netland the Ph.d. student working on this project, for his input and discussions.

I'd also like to thank Hung Bui and Jeremias Moragues who are working on the same project and whom I share offices with.

I must also direct a huge thanks to my family especially my sister Kjersti and her husband Jan Petter Morten who have helped me with some of the editorial work on the report. And also my nieces and nephews who provide great inspiration and diversions.

And finally thanks Mum!

Contents

Chapter 1

Introduction

As the threats of global warming and depletion of our oil and gas resources are looming, the search for alternative energy resources are becoming more and more important. One alternative is harnessing the powers of the wind with wind turbines. Wind farms require large areas with good wind conditions to be effective. The sea has many large areas where wind conditions are good. Exploring the possibilities this resource has, is something that lately has gained a lot of interest.

There are however many challenges when it comes to building wind turbines at sea, one challenge is operations & maintenance (O&M) which according to estimates will account for around 20-25% of the total income [18]. This studie is aimed at exploring the possibilities remote presence coupled with a robotic arm has at helping with O&M in offshore wind farms, and some of the challenges involved in this solution.

The project has so far been concentrated on reducing the amount of on-site inspection needed. A prototype rail and cart system has been created and a camera has been attached to one of the carts. This system is however limited to inspection, and the camera can only go where there are rails. Rails has to be constructed in such a way that they don't limit the possibility of doing on-site O&M. This may call for a more flexible solution.

1.1 Possible uses of a robotic arm

One can envision many useful things a robotic arm can do inside a wind turbine. The arm can be designed to open up doors to accommodate inspection of the underlying systems, and if the arm gets equipped with a camera it will allow for closeup inspection of parts and to. This gives the opportunity to inspect that part from different angles. And perhaps most important with respect to inspection, it will let you see areas and parts not accessible by the original system. It can further be designed to do small maintenance jobs such as lubrication and possibly changing or realigning PCB cards. Perhaps in some distant future you might have a system which is in large parts self repairing where maintenance consists of resupplying the wind turbine with spare parts and removing the broken parts for analysis and recycling. In Figure 1.1 I have made an illustration to show the basic concept of this system.

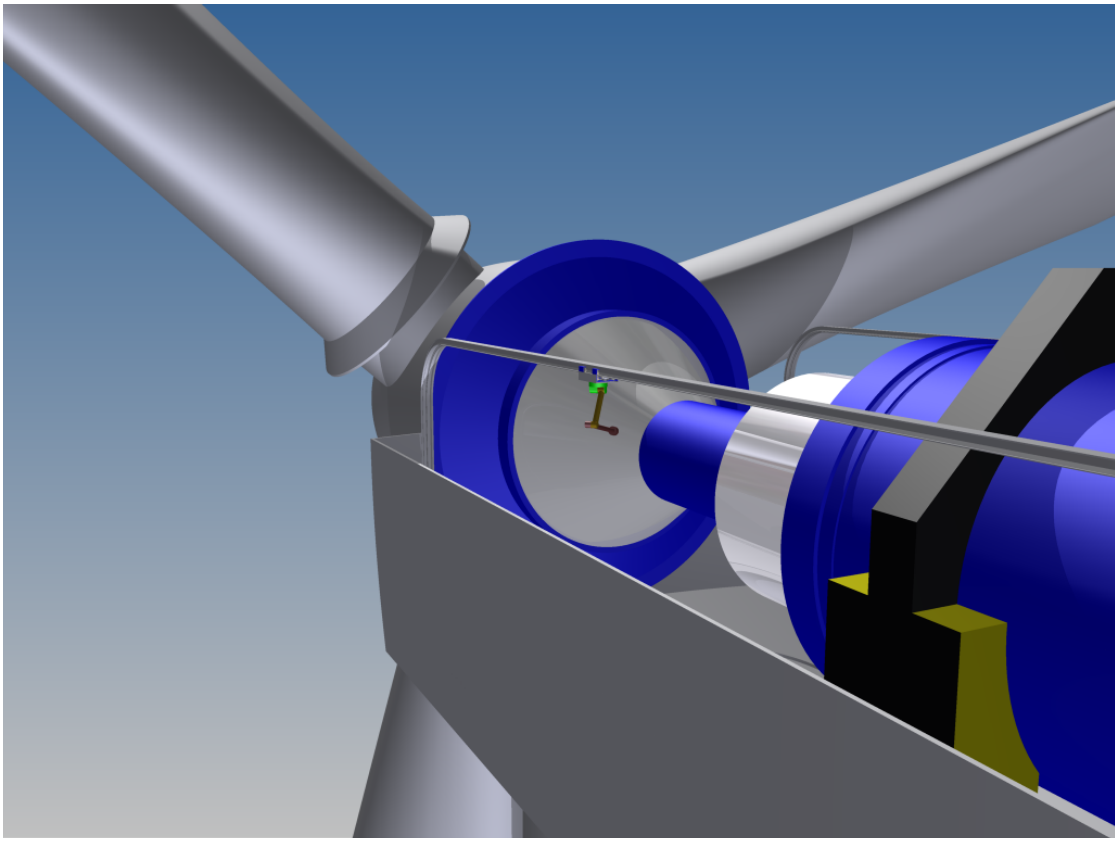


Figure 1.1: Picture illustrating the system. The top of the nacelle has been removed to give a view of the inside. You can see the rail running along the generator, gear and the brake. The cart and robot arm is mounted on the rail and should be visible on the left part of the rail.

1.2 Scope of work

The introduction of a robotic arm inside the nacelle[1] can, depending on design, help reduce the interval between on-site work. However many questions needs to be answered, such as: What tasks are we going to design it to do? How long does it need to be? How many degrees of freedom (DoF) do we need? what end effector is needed? How much weight does it need to support? How can we make certain that the arm don't damage any components of the wind turbine or it self? The goal for me is to find the answer to as many of these questions as possible and also to explore the solution of some of them.

The work has in large parts been a literature study, where the findings of this has been related to this system. This is especially the case for telepresence and collision avoidance.

Some work has also gone into 3D modeling to get a better sense of how the system will work and where the challenges lay.

1.2.1 Outline of chapters

- In chapter 2 the concept of maintenance and inspection are discussed and some solutions for this are discussed. The work done previously in this project is also discussed and how this can be used further along with the arm. The work that gos on in parallel with min is briefly introduced.
- The main topic of chapter 3 is telepresence. The concept is explained and some of the challenges when it comes to telepresence is outlined. preceding this there is a discussion on the solutions one can use to solve the problems in particular augmented virtuality is discussed.
- Chapter 4 deals with collision avoidance that the system can deal with this problem may be essential for it to work properly. A system that does this job properly can be important for both the nacelle and the robot arm.
- In chapter 5: Sensors and tools, the main topic is sensors although tools do get discussed to some extent.
- Chapter 6 deals with some of the concerns of the system when it comes to the mechanical design. some requirements and constraints are discussed. A possible solution is discussed.
- Chapter 7 details some of the work I have don with 3D modeling. And how this may help further along in the project.

[1] The nacelle is the part of the wind turbine that houses the generator and sits on top of the tower

Chapter 2

Background and motivation

2.1 Maintenance

Maintenance is the task of making certain that a system is operational. In some cases you are not allowed to have any down time, this makes for both an expensive system and calls for hard demands on the maintenance routines employed. In our case the constraints are not so strict, as some downtime can be allowed. The driving factors here are economical. From an economical point of view you don't want to do more maintenance than strictly necessarily to keep the wind turbine operational. Obviously when the wind turbine is not working and there are good wind conditions for producing power, the owners are losing money. To make good economical projections you usually want to plan when the system is down for maintenance. To achieve this one usually employs a maintenance strategy called planned maintenance. This can involve cleaning and refurbishing and a general inspection of different parts. If you or the company producing a specific component, have past data for that component, you can also make predictions on the Mean Time To Failure (MTTF) based upon this. Using this data you can make plans for when you want to replace that component. However in some cases it can be more economical and practical to run a component until the end of its lifetime or until it fails. Such a maintenance strategy is called Corrective maintenance. If there are no data to be used for estimating the component's MTTF, and no measurements can be performed, or such a measurement would be to expensive compared to the potential benefits of doing it. In such cases the corrective maintenance strategy can be the one to choose.

A different and more advanced strategy to determine maintenance schedules and the replacement of parts, is to employ a system to record and monitor different data in the system. These sensors give information about the health of the system or specific critical components of the system. This is called Condition monitoring.

This can potentially allow you to predict more precisely when a component will fail and thus you don't have to replace it prematurely or for that matter allow the component to go beyond its point of failure. Such a system can also be used to prevent a failure to happen. This can be achieved by letting a safety system, or possibly an operator, shut down the system before a component has completely degraded. You can then potentially prevent a cascade of failures to components that would otherwise be damaged.

Condition monitoring allows you to go from scheduled or corrective maintenance to predictive maintenance or condition-based maintenance. Condition monitoring however can't catch every kind of failure, not every thing can be effectively measured and the measuring devices can themselves fail. Condition monitoring systems have been employed in some wind-turbines and some experience has been gained with regard to this [2, 18, 27].

In [18] some of the measurements you can use in condition monitoring systems are outlined and discussed:

- Using accelerometers to measure vibration in the gear, shaft and bearings
- Torque measurement to measure the rotor load
- Oil/Debris Analysis to check the health of bearings
- Temperature of bearings
- Acoustic Emission to check the health of bearing and gear
- Stator Current/Power

For a more comprehensive study into these types of measurements take a look at [18].

Wind turbines are quite complex systems and you can't necessarily look at one measurement and determine whether or not that reading is unusual. You have to view the data in correlation with other data to determine whether or not you are in a safe state.

The condition monitoring systems are usually quite good at finding out that a failure has occurred and that there is a fault in the system. Indeed some times it can detect that a failure is about to happen. However it does not necessarily tell you precisely what has happened and specifically what component has failed.

2.1.1 Inspection

Inspection is an important part of maintenance. It can allow you to find parts that are about to fail before they do. Also during an inspection you can do minor repairs. This can help you to keep the wind turbine operational longer.

An important part of this is that you can make corrections on components that otherwise could fail and cause bigger problems in terms of cascading damage from one failure.

When it comes to wind turbines one important part of this is blade inspection. This is usually a visual inspection either performed with personnel doing a rappel down along the blade when the turbine is not operational, or by a camera mounted on a remote controlled mini helicopter. However the German Fraunhofer Institute for Factory Operation and Automation has developed a robot (Figure 2.1) that is capable of doing this inspection automatically. The robot is equipped with infrared thermography sensors, ultra sound sensors and a camera. This means it can provide you with more data on the health of the blades [5].

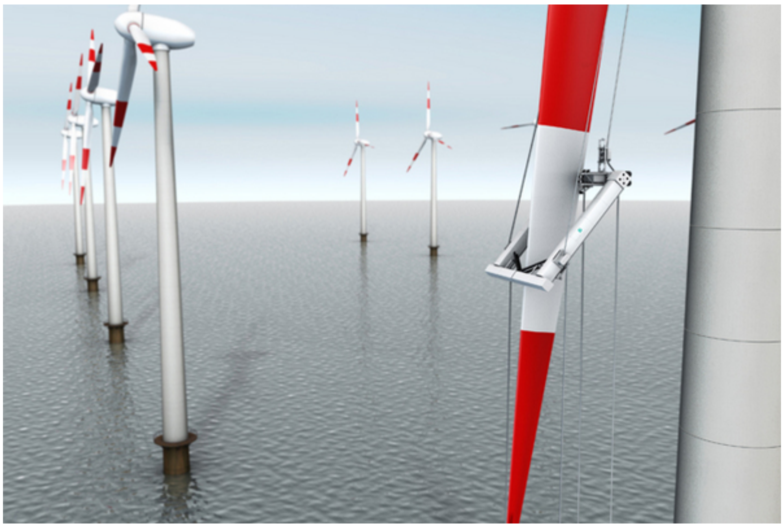


Figure 2.1: The blade inspection robot

There are also rail mounted robots in use for inspection. This is used in pipes and in nuclear reactors.

It should be clear that a lot of money is being spent at keeping these investments running, and so any savings either from heightening reliability and up time or cheaper O&M should be welcome. It is in these areas we believe our system can be of benefit.

2.2 Previous work

To monitor or do inspection inside a nacelle it previous work in this project has argued that a system consisting of a rail and one or more carts is the best option. Indeed if you want to extend the system capabilities to that of maintenance this design decision makes even more sense, as this system should easily allow you to bring the tools you need to where they are needed with out to much fuss.

2.2.1 Rails

Figure 2.2: Rails with carts mounted

A set of rails and carts where developed by Viktor Fidje and the details of the design can be found in [8]. As can be seen in Figure 2.2 the rails consist of two aluminum tubes running in parallel, connected by C shaped crossbeams. The design is relatively inexpensive and the tools used for bending equally so. However, the bending process used has deformed the cross sectional shape of the pipes so that they are no longer circular, but elliptical. This and other factors mean that the wheels of the carts don't have good surface contact. It also seems somewhat difficult to design a good wheel configuration to accommodate this design. In particular with respect to the demand for longevity of the cart and rails it seems hard to make a good suspension system for the rail.

When Torgeir Welo a professor in mechanical engineering was asked to design a rail profile for this kind of system he came up with a wholly different design using a monorail. This design looks more promising as far as stability goes, and the forces working on the axles of the wheels will be more or less parallel to the axis of the wheels.

The design that has been implemented does however easily beat the monorail design when it comes to ease of construction and the limits posed by the budget. It is also a good start as far as prototyping goes, and it may suffice to test new ideas. However not everything built for this rail can be directly ported to the final system.

2.2.2 Carts

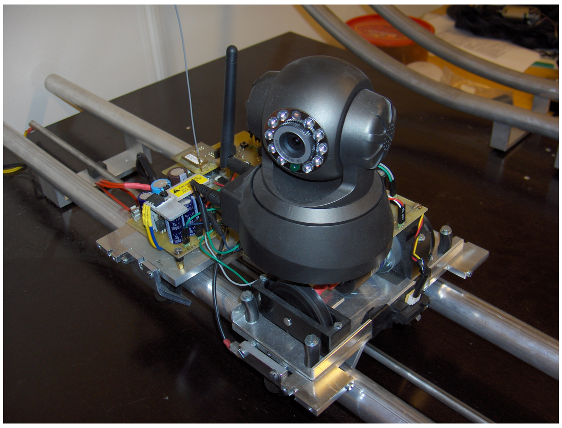


Figure 2.3: First cart

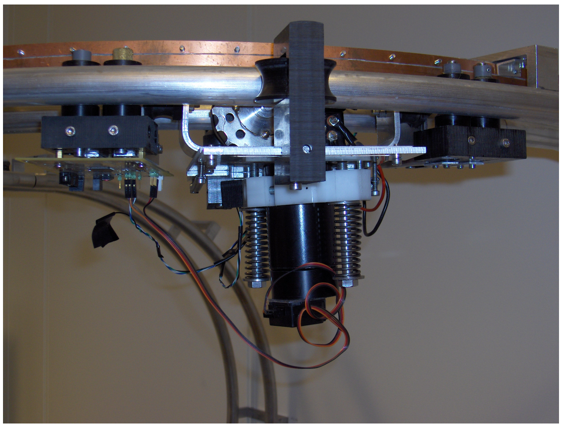


Figure 2.4: Second cart

The first cart created (Figure 2.3) is the one that has the camera installed. It consists of two parts linked together by a joint that only allows for movement in the horizontal plane. This means that it can't ascend or descend along the vertical bends of the rail. To move this cart along the rails the large wheel on the topside has been motorized. It was however found that this did not provide enough friction to move the cart when there was a small inclination of the rails relative to the horizontal plane. This is true even though the wheel has been fitted with a rubber O-ring to offer better traction on the rail. This also has the effect of making it impossible for the cart to handle the bends in the rails, as the link between the two sections is quite stiff and the wheel gets less friction in the bends of the rails. It is thus limited to a straight rail with only small inclinations.

For these reasons a second cart has been developed (Figure 2.4). This cart has a cogwheel that interacts with the bicycle chain to provide locomotion for the cart. The cart consists of three sections, two that provides stability and some room for electronics and potentially camera and sensors, and one in the middle that provides a platform for the motor. The sections are connected with a flexible joint that provides freedom to move in any desired direction along the track. This has however not been tested as the rails are not complete yet.

A big problem with the second design is that it vibrates quite heavily while moving. This is due to the interaction between the cogwheel and the bicycle chain. This makes it unsuited for our purposes as the vibrations are likely to reduce the lifetime of the whole system. The vibrations will probably also propagate to the camera and/or the arm. This can be detrimental to the quality of the visual feed to the operator and control of the arm. These problems has been considered too difficult to tackle when it comes to testing how a remote presence system might work in the an inspection situation.

The cart and rail system developed by the professor can be seen in Figure 2.5. This system has obvious advantages over the previously built systems when it comes to providing a stable platform for a robotic arm. All the wheel axis are parallel to the surface the wheels are in contact with. The drive system proposed will be made from plastic. Which should prevent large vibrations.

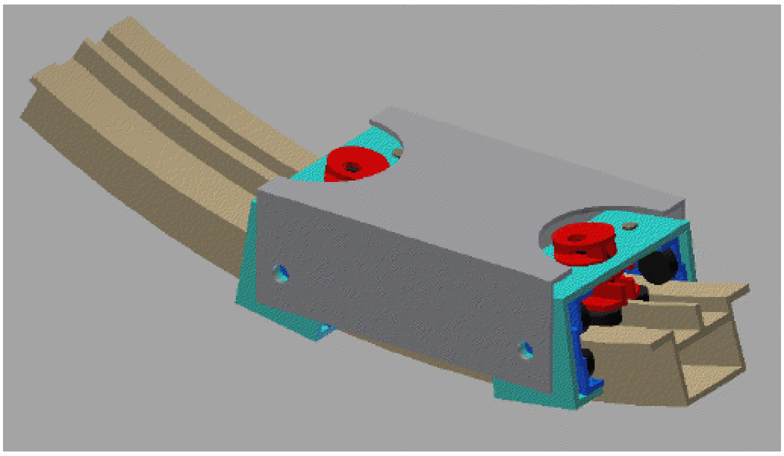


Figure 2.5: Basic Rail and cart design

2.3 Current work on the project

As I am doing my work on the topic of this report Jeremias Moragues and Hung Bui is also working on this project.

Jeremias Moragues is working on his master thesis. This thesis deals with whether or not this system using remote presence is efficient and a feasible solution. A test to see if this is the case is being planned.

He has added a Joystick to use with the interface as a main system controller.

In addition he has studied the anatomy of a real Nacelle and the current maintenance that is done by three different companies.

Hung Bui is like me working the specialization Project. His main focus is on improving the hardware, and has been working with a new Pandaboard and a new camera.

He is also working on finding the main latencies in the system so the delays can be limited.

Chapter 3

Telepresence and control interface

It is important to have a clear understanding of who the users of this system will be and how the system is going to be used. This should help us to determine what features are needed and what might not be strictly needed but nice to have.

The first goal of the project will likely be directed toward getting a system that can give sufficient information during a remote inspection to justify not doing frequent on site inspections inside the nacelle.

To this end we can either use a camera mounted on a cart, or as we will explore further in this prestudy, a robotic arm with various appropriate sensors mounted on it.

As far as who the users are going to be it is likely that these will be trained professionals that at least in the first instance will have first hand experience at the inspection job. But we can not necessarily expect that they have any particular skills when it comes to robot control, especially remotely where you can't directly see what the robot is doing and how it is positioned in its surroundings.

The fact that the users may not have much experience with remote robot control and more specifically this system, with its interface and peculiarities, is important to consider how the system is controlled and what support the interface can give to the operators.

Even though it should be expected that the operators will gain experience and confidence in the system eventually, every new operator will at some point be new to this.

3.1 Technical challenges

The introduction of a robotic arm in an environment such as a nacelle imposes many interesting challenges. These challenges are not limited to the construction of the arm and the capabilities as far as reach and strength goes.

If an operator is allowed to move the arm freely inside the nacelle, the arm can potentially damage components of the nacelle and/or itself. In many ways this defeats the purpose of introducing the arm to the nacelle in the first place. This can make potential buyers of such a system very skeptical to the prospect of installing the system. It is clear that this issue needs to bee further addressed.

There is another challenge that is somewhat linked to the challenge of collision. That is how do we control the arm. The arm may at some instances provide the system with extended capabilities as far as automatic inspection goes. Using perhaps predefined or dynamically calculated paths we can get the arm into different positions, where it can do some type of work. However this control strategy is probably not the best if the operator discovers something he would like to have a closer look at. Then it must be considered how an operator sitting far away from the nacelle can control and interact with the arm in a more direct approach, perhaps by allowing the operator to control the speed of different parts of the arm. The considerations we must make takes us to the realm of telepresence[1].

Before we start to handle the issues of what physical dimensions different designs impose, we are going to discuss the afore mentioned challenges starting with telepresence.

3.2 Telepresence

The concept of telepresence is to give an operator siting far away from the site you want to interact with the feeling of being present at the site. Research into this field has been going on for quite some time although Marvin Minsky first coined the term in an article in the 80s. Before him the term used was "teleoperation" and that can be traced back to Nikola Tesla. Telepresence technologies are used in hazardous environments, pipeline inspection, remote surgery, education and many other areas.

A question one needs to ask before starting to design a system is: What constitutes a quality telepresence system? And what factors can degrade its service? As the field of telepresence is not new there are people who have experience with these systems and many experiments and studies have been conducted [9, 24, 29].

A concept that pops up often in these papers is that of the operator's situational awareness (SA). Different people in this field have defined this in different ways. According to [3], "The most widely accepted definition was developed by Endsley [1988] as, "the perception of elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status

[1]Telepresence refers to a set of technologies which allow a person to feel as if they were present, to give the appearance of being present, or to have an effect, via telerobotics, at a place other than their true location. [25]

in the near future"". However they also go on to develop their own definition suggesting that the definition given by Endsley is not so useful for their purposes.

One way to view SA is as a qualitative measure that says something about how well an operator is able to identify where the robot is located, what speed it has, how it can interact with its surroundings, and so on. This in large parts informs upon what actions need to be performed to reach some goal, and how fast the operator is able to perform the task to be done.

In [28] some of the factors that comes into play in telepresence and SA are outlined:

- A teleoperator will usually not experience any of the acceleration that the robot undergoes "when an observer moves, the vestibular system provides feedback about acceleration that can in principle be used to interpret rate of motion and thus provide a natural scaling of the distances in the environment." It is further explained that this is not only a problem of lacking information of acceleration but also that your visual perception is in conflict with your vestibular systems[2] sense of motion as the operator is standing still.

- Most telepresence systems to date don't operate with stereo-vision. This makes it harder to get a sense of depth and surface shape. You do get some cues from shading, parallax[3], motion, perspective and texture deformation.

- Your perception of motion can also be somewhat skewed by the positioning of cameras. "The relationship between optic flow and rate of motion in the environment depends on our eye height, or camera height for the robotic platform". Basically if the camera is low to the ground and/or the environment you are moving in is close to you, you will perceive the robot moving faster. While if the camera is situated in a higher position in an open environment you will perceive the motion as slower. As you are not experiencing any of the acceleration on your body this Leads to an ambiguity[4].

- The "soda straw" or "keyhole" effect is not only discussed in [28] but also in [20, 26, 30] and many other papers. What the soda straw effect describes is the limited viewing field you get from looking at the world from a single

[2] The vestibular system is located in your inner ear and is used for balance and sens of motion

[3] "Parallax is a displacement or difference in the apparent position of an object viewed along two different lines of sight, and is measured by the angle or semi-angle of inclination between those two lines" [22] A simple example of this is how objects far away seem to move slower than those near to you

[4] If you want to experience the optic flow effect for your self you can try out a car driving game that lets you change the viewing position and compare the effect of driving from the ordinary position and one where the camera is situated right in front of the bumper

camera. You get a good view of what the camera is pointing at but you don't receive any "data" for your peripheral vision. The keyhole effect is the combination of this and the problem of where to look next. A common example to illustrate this can be found in references [26, 28], which describe the scenario of how you are able to effortlessly redirect your gaze when you perceive something of interest, while climbing a flight of stairs, and then contrasting this to the more fixed or slow redirection a robot labors under when it comes to "eye" movement. These things can lead to missing vital events or degrade your SA.

The peripheral vision should not be underestimated when it comes to providing SA in telepresence or indeed your everyday life. In [24] a setup where the operator was given multiple camera angles and monitors to provide peripheral vision while driving a vehicle is described. The operators could decide for themselves where the bandwidth budget was to be spent while driving. That is they could have high resolution in the front and low on the sides or change this however they decided, while also balancing the bandwidth budget. They found that the operators preferred to turn the resolution up on the displays that provided the peripheral vision and adjust the front view down when the speed of the vehicle increased.

- In [20] they found that operators experienced problems with controlling a robot with a pan-tilt camera. The problem occurred after the operator had operated the camera independently of the robot, so the viewing angle was not the same as the direction the robot moved. Often this resulted in the operator crashing the robot.

- One final thing we will consider, is the delay or latency. If you haw relatively large delays between giving a command and seeing a response to that command this quickly becomes frustrating for the controller, and you can no longer expect the operator to perform his tasks with any kind of expedience. Studies into this show that when there are large delays the operator often employ a control technique where for instance they give a command to move forward for some time and then let go to see where the robot ends up. It is not hard to see that this is inefficient.

  This system will likely operate on the Internet and not on a dedicated or private network where you have some control over the traffic. Delays can't be discounted from happening. Indeed the Internet will only give you a best effort guaranty, and the delays may be varying with the amount of traffic in the nodes between the operator station and the wind farm. This comes from the variable amount of time each packet spends waiting in the buffers of the routers to be transmitted. The network will also try to balance the load of the individual routers, and as there are usually many paths the packets can take this can also give rise to a varying delay. In addition there are inherent latencies in the system it self, from capture of video. compression and decompression.

Delays also has consequences for which control strategies you can and should choose.

Also if you have large delays you can't employ Hepatic feedback. If for instance you have a joystick capable of giving resistance when you hit a boundary you are not allowed to pass, delay between hitting the boundary and applying force to simulate resistance in the joystick will cause oscillation in the joystick and this will be feedback to the robot causing isolation there.

3.3 Telepresence and remote camera

Having established some of the problems commonly found in telepresence systems we now ask: how do they relate to our system?

If we look at the system with only a rail mounted camera, many of the problems we have discussed disappear or are not of much interest. The robot can at any time only move in two directions. The rails will have to be placed in such a way that under normal circumstances the robot has a clear path to move along the rails, this should significantly limit the likelihood of collisions.

A large time delay may cause some problems for the operator in that he may drive past something of interest and then have to backup to get a better view. However as he is unlikely to collide with any thing he don't have to employ a very restrictive driving style. If there is a chance of something lying on the rails, or more generally in the path of the cart it's likely that you can circumvent danger of any damage to the system by a simple proximity sensor on both ends of the carts. Such as collision avoidance system has to be created with some care so that it don't react to the rails.

The problems described in [20] of controlling the robot while the camera angle is not aligned with the direction you are driving is also not likely to be a large problem. Again this is due to the limited freedom of the robot and the unlikeliness of crashing.

Something else that may be of little concern is the problem of the operator not knowing where he or more precisely the robot is, inside the nacelle. In many of the papers on telepresence, awareness of relative location crops up as a problem. In these cases they are usually exploring areas unfamiliar to the operator. I think it's fair to assume that the operators working with our system have been inside the nacelle they do remote O&M on, or that they gain the experience they need over time. It's likely that they will become familiar with the areas they are inspecting.

3.4 Telepresence and a robotic arm

Adding a robotic arm to the system will on the other hand complicate matters. The assumption that the robot will have a clear path while moving along the rail no longer holds. The rail placement has to be constructed in such a way that the system can move along it while the arm is in a minimal configuration or that there is some configuration of the arm that allows the robot to pass that part of the rails. However you can't expect that the rails will be placed in such a way that the arm can be extended in any configuration and not collide. Indeed if the arm is constructed to get closer views of components inside the nacelle, or in some way is constructed to interact directly with its environment, placing the rails in such a way that this gets harder is obviously counterproductive.

It is clear that we can't count on the system to be inherently free from the problem of collision. As far as the operation of the arm and the robot is concerned there are some things we can do to help us avoid collisions.

We can restrict the operation of the arm to the extent that you can't move the carts while the arm is extended. The arm can move into an initial position or a configuration in which we more or less know that the arm won't collide into anything while the carts are being moved. This may be a very restrictive way of handling this part of the problem. If the operator is close to getting the arm into the position that is wanted, but not as close as needed, and the only way of getting to that position is moving the carts, he will first have to retract the arm, then reposition the cart and extend the arm. It's obvious that this will be frustrating and if this is something that is done often, the system will not be well received by operators. You could potentially provide an override function to the operator so that he don't have to go trough this procedure. However then you cant guarantee that the arm will not collide while moving along the rails.

One thing the previous solution doesn't handle is the danger of collision while the operator is just operating the arm and not the position of the cart. The factors that contribute to this danger can be reduced by providing the operator with all the information he needs not to crash the arm and make the right decisions. This problem is one that can partly be minimized by good SA. It's hard to know exactly what information will be enough, how it should be presented, how much is to much and also the problem of controlling the robot and avoiding collisions is not the main objective for the operator as his job is O&M.

Technical solutions that could potentially increase the operators SA with respect to the arms position and likelihood of crashing is mostly centered on vision. Unlike a human moving in an environment the operator don't get the same kind of sensory feedback as to the position and movement of limbs in space. As a human moves he can generally speaking identify where his arms and feet are and an approximation of what angle they are relative to each other or the body in general. This is not true for the operators sense of where a potential robotic arm is, as these kind of cues are difficult if at all possible to give to an operator. The option left to us for providing good SA with respect to the arm position in it's environment is then to let him see the arm while it's moving to visually inspect that it's not crashing.

One way would be to put up cameras in different positions that gives a view of the arm. You could for instance place one camera at the base of the arm and possibly also one on a cart at the side or on both sides of the arm. These cameras can give a view of the structures around the arm that the operator has to avoid colliding with. This solution does however have some problems. The cameras might get obstructed from seeing the arm in the areas of the nacelle where the likelihood of colliding is largest, and finding a camera setup that will provide the information needed might prove difficult. Having found one camera setup that works well in one nacelle might not work as well in a different nacelle. And having to develop a different setup for each nacelle is far from ideal.

For the operator it might also be difficult to have to contend with all the different camera angles and if he is forced to control the cameras in addition to the arm this might be difficult. It will be important to carefully consider camera angles, placement of cameras and how often the operators have to interact with this to control the arm.

3.4.1 Augmented Virtuality and robot control

A different way you can lessen the collision problem is to develop a Augmented Virtuality system where you get a third person view of the arm by showing a 3D representation of the arm on the screen and also rendering, from 3D models representations of parts of the environment the arm is operating in that is not in the field of view of the camera.

In Figures 3.1, 3.3 and 3.2 the result of three augmented virtuality approaches are shown.

In the augmented virtuality system described in [17] and shown in Figure 3.1 they have used a LIDAR to get the 3D data on the environment. They then build a 3D model of the environment and use the camera image as a texture map on the 3D model. This research has also been mentioned by Tor Mælum Karlsen in his master thesis.

This does look good for orienting yourself in the environment. As you also can see a 3D representation of the robot this should help improve the operators SA.

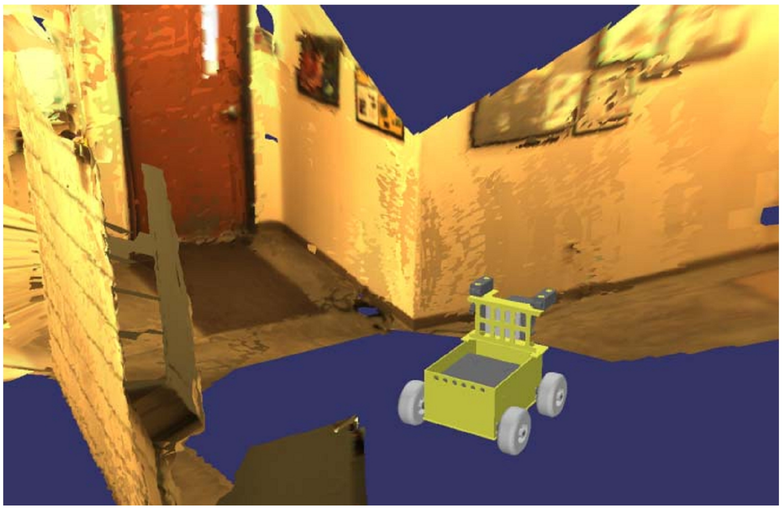


Figure 3.1: Augmented virtuality system with texture wrap and third person view. Picture taken from [23].

If such an approach were to be used inside the nacelle you would have some problems. As can be seen from Figure 3.1 the texture mapping causes the images used in the mapping to become warped and distorted. This can prove problematic when it comes to the inspection task as you quickly get changes in the model from one inspection to the other, simply due to “noise”. This problem increases when the environment has a more complex geometry, such as you would expect to find inside a nacelle.

Also it may be hard to see if the images used as texture maps are newly created or if they are old and possibly outdated. If care is not taken with regard to this you can easily mistake an old image where there are no errors in the system as the correct image and not point the camera in that direction and thus miss important information.

In the augmented virtuality system shown in Figure 3.2 they have used a pre-existing 3D map of the environment and mixed it with the video feed from the cameras, a map and an avatar robot. the camera feed is here simply projected on a flat surface and not warped around the walls. This is probably a better approach than the previous when it comes to inspection.

One thing that may not be so useful in our case is mixing the 3D information and the video feed. As can be seen parts of the image is covered by the blue blocks that represent the walls, and also the limit to where the robot can move. This hides some of the information in the image from the people doing the inspection as important information may be covered up.

The 3D map information is here used to signify the limits of where the robot can

Figure 3.2: Augmented virtuality system with integrated 3D map and third person view. Picture taken from [3].

go. For the robot used in these experiments the only thing that is of significance is what parts of the floor the robot has access to. If we imagine how this would be for a robot that is free to move in all three dimensions this would mean covering even larger parts of the image with 3D information to convey to the operator where he can go. The limits to where the operator can command the robots is probably something that the operator gets equally well from the camera. And if better depth perception is needed in our system it might be better to use stereo vision. Either from using a stereo camera or from extrapolations from the one dimensional camera feed.

Finally we come to Figure 3.3. In this approach to augmented virtuality you simply have a three dimensional rendering of the environment around the camera view. This removes some of what has been called the soda straw effect.

This approach can be combined with a rendering of the robotic arm derived from knowledge about its position in the environment. With a view of the relative position of the cart and joint angles, the operator should be able to effectively see on the screen where he can and can't go. Also there should be nothing obstructing the view of the feed from the camera.

You can also let the operator control what camera angle he wants to use while operating the arm. As the camera showing the virtual environment is it self virtual this can be done relatively fast. If the main view and some different view points

Figure 3.3: Augmented virtuality system to dispel the keyhole effect. Picture taken from [17].

are stored relative to the robot they can easily be accessed again.

This can also be done by using one part of the screen to show the video feed and a different to show a 3D model of the arm in its environment. This is probably the easiest to implement.

3.5 Concluding remarks on telepresence

If we deem it important that the operator should be able to take direct control over the robot and also be able to get inspection information while moving the arm the last method described seems the best as far as augmented virtuality methods are concerned. But you may also be able to solve this problem using several cameras with clever positioning.

Apart from the method where we don't allow the arm to be in any other position than the initial position while moving the carts, non of the technical solutions guarantee that the robotic arm does not crash into anything damaging components. Al they do is decrease the likelihood of that happening while placing the responsibility for that not happening on the operator. Some of them have a potential of being useful in providing the operator with SA when it comes to executing his main task, that of O&M. But if we impose upon the robotic system that no operator error may cause damages or faults in the robot or any components of the nacelle. We need to look at a different way of providing collision avoidance and that is the subject in the next section.

Chapter 4

Collision avoidance

There are mainly two strategies for effective collision avoidance. One approach uses some type of proximity sensor to sense if there is anything that the robot can collide with in the direction of motion. The control system must then take some appropriate action based upon this sensor data. On robotic arms you will usually have to cover a large part of the robot's surface in an array of sensors to get enough coverage, this has commonly been refereed to as a sensor skin or just skin.

The second approach uses a 3D or in some cases 2D model of its environment to restrict the motion of the robot. This model is either static, meaning one assumes a static environment and simply preload the model to be used by the control algorithm, or it can also be dynamic, getting updated by some form of sensor input.

4.1 The sensor skin approach

There are many different ways to detect the proximity of an object, all have different advantages and disadvantages. Some of the principles used for proximity sensing are:

- Capacitive sensors use the difference between dielectric effect from air and the object in the capacitive field of the sensor.

- Active ultrasonic transceivers measure the time it takes from a ultrasonic wave leaves the sensor and is subsequently reflected back from an object.

- Active infrared sensors measure the light intensity from surfaces being radiated by a infrared light source

We will now consider the use of such a system applied on a robotic arm.

As any sensor is only capable of detecting the distance of any object to itself, and not the exact position of that object, you have to rely on more than one sensor to cover the entire stretch of any joint of the arm. This means that you add more weight on the arm also you will have to send more signals down to the control unit and this may mean more wiring. This is something that can be problematic as this can to some extent hinder movement in the joints, and also the wires are subject to more wear and tear when they must be bent back and forth.

In [10, 19] they describe systems employing infrared sensors to measure distance to objects in the operating environment. The systems do work but they are dependent on how well the surface is at reflecting back the infrared light. The sensors also have to emit an amplitude modulated “signal”, so that the reflected light from sensors can be distinguished from ambient light.

The systems also require a large amount of sensors to be completely covered. In the paper describing the most complete system [19] they used “hundreds” of sensors to achieve their goal of collision avoidance. This can in other words be somewhat expensive even though a single sensor might not be extremely expensive.

The robot in [16] uses ultrasonic detectors for collision avoidance. They do however only use two sensors in their experiments. If we where to use this type of sensor in our system it would probably have to have many more sensors to provide complete coverage, while not being to restrictive.

As mentioned above, a change in capacitance when an object enters the electric field of capacitors can be used for determining the distance to objects. A system using this technology is described in [7]. The Sensors they describe seems to be constructed simply by using a three layer circuit board where the sensor is simply etched in. This should make for a relatively inexpensive system that should be easy to manufacture. with an approximate range of 30 cm.

This type of sensor skin can be used and this has been demonstrated to work. One potential problem is if at some point we want the arm to lift up a part or a tool that is not covered by the sensor skin. Then we can’t guarantee that this part can’t crash into something. While this may be a minor thing the system does not guarantee that there will be no collisions in this case.

4.2 The environment model approach

In this approach to collision avoidance you use a model stored in memory to determine where the robot is allowed to go. As mentioned there are two principal ways of doing this either you have a predefined model of the environment or you build one up from sensor data.

4.2.1 Using a predetermined model

One of the arguments used in the papers describing the sensor skin approach of collision avoidance is that you can't always get a good model of the environment. In our case this is not necessarily the case. One should expect that wind turbines today are constructed with CAD drawings and models as a blueprint.

One can then extract the relevant geometry from this data and use this to construct a legal set of space where the arm is allowed to move. It should also be fairly easy to restrict this even further if there are areas that the arm should not move, even though there are no physical objects there. This can be of interest around high voltage areas inside the nacelle.

You can also make a 3D scan of the nacelle before the system is made operational. This can be done with a 3D laser scanner or you could use Microsoft's Kinect.

One thing that must be taken into consideration is how to get the data. If the buyers of the system have all the necessary CAD models used for the construction of the nacelle this will likely not be a problem. This would be the case if the buyer is also the company responsible for construction. However, if the potential customer is the owners of the wind turbine they don't necessarily have this data. In such a case one must try to buy the necessary data from the company responsible for the construction. They may not be wiling to give all the data used for the construction of the wind turbine, or they may not be wiling to do the work of converting the CAD models into the data needed in our system.

It must perhaps be clarified what is meant here by "necessary data". CAD models can in many cases contain geometry data on the inner workings of different components. For instance a CAD model of the gear inside the nacelle can contain details on how the cogs and axles inside should be constructed and aligned inside the gear. Such information is not necessary for the environment model as this is covered up and the robot will not be in danger of colliding with this. Data that are redundant in terms of either collision avoidance or as a means to help with the operators SA, in case it is used for that purpose, should probably be left out so that requirements on the computer power is not so high.

It should be possible to provide collision avoidance when the geometry of the arm changes due to new tools or if an object is picked up. Depending on what type of change occur you might put a close to exact model of the object or toll on the arm in such a way that the geometry follows the movement of the arm. A different option is to let the operator create some simple geometric shape that acts as a boundary. This should guarantee that the system is collision free so long as the shape the operator uses covers the whole real life object. An example to illustrate this, would be if a disk shaped cover plate were to be lifted up. The operator could then create a cylinder boundary and constraint this to the robot arm so that the shape follows the arm as the cover plate would.

One thing that might be somewhat problematic with a fixed predetermined model is that if the environment changes the model has to be updated. Also there can be minor differences between the CAD model and the final solution. If the CAD model is not updated with these changes the arm may crash.

4.2.2 Using a dynamic model

With a "dynamic model" I mean that the model can be updated when it detects changes in its environment. This can be achieved by having a Kinect or a LIDAR unit permanently installed on the cart or the arm.

If one finds that it is necessary to bring new equipment into the nacelle, there is no longer a big concern of how this will affect our system's performance or that you will have to do additional work for our system to work properly. On the other hand if you do not have a permanent system, or you use the sensor skin to do this, you will have to manually update the environment model to accommodate these changes.

On thing that has to be taken into account with a system like this is how to update the system when natural changes occur. One example is of this are doors that open and close. This changes the geometry and this has to be updated. One way to handle this might be to direct the Kinect or LIDAR to capture the change in geometry. This change can then be stored so that the next time the door is opened you know how the geometry will be after this change and you just load the stored geometry data into the model.

You can also have the Kinect or LIDAR directed toward the area where the arm is so that you have some additional security. This can be important if for instance the door fails to close when given the command. If You can then detect a discrepancy between your model of the environment and how things actually are.

Using a permanently installed system to generate the environment model does however mean that the equipment used for this generation has to be relatively inexpensive.

Of the two technologies mentioned for this use the Kinect seems to best satisfy the requirements. In the next chapter we will look further into this technology.

4.3 Final remarks on collision avoidance

Using a predetermined model or a dynamic model of the environment seems to be the best solution to collision avoidance. As you don't add any weight to the arm, and it doesn't require extra cables between the joints in the arm.

It is also likely that you can use the same geometric data as part of a augmented virtuality system.

With the dynamic model, the system is independent of CAD drawings that may be hard to get. There are also less concern when it comes to changes inside the nacelle.

Chapter 5

Sensors and tools

When deciding what sensors to put on the arm it's easiest to first start discussing what senses a human uses to familiarize himself with the immediate surroundings, and specifically what can be useful inside a nacelle.

sight is used both for localization and for detecting visible clues on the health of the components of the nacelle. This can be oil leaks, wires that are losing their isolation etc. This is the most obvious sense that we want our system to provide us with. It is also not very difficult to do this using cameras.

It can also be an advantage to have the ability to listen to the mechanics inside the nacelle when the wind turbine is operating. You can then pick up on unnatural sounds such as grinding noises in gears or bearings. Again this is a "sense" we can give our system quite easily trough a microphone.

The two senses we have covered so far are the most obvious. However there are more senses that can be useful. Personnel inside the nacelle will in addition to hearing and seeing also be able to sense heat coming from different components, they can feel vibrations, and they are also able to smell.

There are different ways in which one can give the system a sense of heat. In conjunction with our system the easiest and least expensive being an infrared sensor and/or a thermostat. A more expensive option is to provide the system with thermal imaging cameras. You could also use thermometers however this is not very flexible.

A sense of smell is more difficult to reproduce or "measure". Despite the fact that the human nose is not the best that nature has evolved its still very hard to reproduce this at a low cost. It is still considered more economical and effective to train dogs when it comes to finding suspicious material in airport security and in border crossings. Although machines have been produced that has some capabilities in these areas.

It should be said that sensors mentioned here are not necessarily the best sensors in terms of cost or size. They are mentioned to illustrate what is out there, they are not necessarily recommended.

5.1 Camera

When it comes to cameras for use in this project there are four main concerns: weight, cost, latency in capture to encoding, and image quality. Beyond this there is also the concern of where the camera or cameras should be placed.

As one can see from [11] the placement of cameras in conjunction with control strategies is a nontrivial matter.

It is however obvious that at least one camera should be part of the end effector of the robotic arm. This will greatly extend the number of viewing angles the system can give. As mentioned earlier there is a problem with restricting your viewing angle to only where you can lay the rails. There are inherent limits to how much the rail can be bent, and also it is not likely that there will be designed mechanisms that will allow the cart to be side tracked to places where the tracks can't go due to to little space to take the tracks back again.

The next question is if there should also be other cameras in the system so that they can help the operator to navigate the robot arm. As we have seen in the chapter on telepresence, it is possible to present the operator with a 3D representation of the arm and its environment so this may not be needed. However let's for the sake of argument say that this is not the chosen path for the project. It then becomes important to provide some of what the augmented virtuality system can do to help with navigation.

Øyvind Netland has done some research into this. A possible way to solve the question at hand is to have two cameras where both are mounted on the end effector. One points in the direction you want to inspect, while the other camera is pointed along the direction of the arm. This way you get to look at what is of interest while the other camera acts as a rear view mirror for the arm.

Depending on the shape and geometry of the arm, you may however not necessarily see the whole arm. If the arm is designed with multiple joints that can fold up to conserve space and also give more freedom to the arm, this can obscure large parts of the arm from view of the camera. This is probably not a major issue.

One problem with the multi camera approach is that you solve an issue that can be solved in software. Also when it comes to the augmented virtuality approach you may be able to solve the problem of delay too. Not of course delays in the video stream but, if you run a simulation of the physical arm and use this to animate your 3D model, you get a more or less immediate visual response to your commands. This will however be somewhat in conflict with the video stream so it is hard to say now if this would be helpful. You will also have to find a way to sync up and realign the model with the real robotic arm. In the end it might be better to send joint position information back to the operator and use this to animate the model. This should be more in sync with the video stream.

Øyvind Netland has also suggested that a fiber optic camera can be helpful in inspecting gears. In such a case it is probably a good idea to have this on the end of the arm.

As Hung Bui is working with a camera that may end up being used, this will not be discussed further here.

One thing that might also help when it comes to seeing what is inside the nacelle and getting a better view of the inner workings without being obstructive is mirrors. they can be placed in some key areas with an appropriate angle. The mirrors must probably be angled face down, as otherwise thy might become dirty and no longer be of use, or increase the on site work having to be done. also what material they are constructed of must be considered as yo don't want them to break to easily.

One problem that might occur with windows in conjunction with the Kinect is that it might not be able to see the reflected Ir pattern in the mirrors and thus it wont be able to detect this.

5.2 Microphone

When it comes to the microphone, what is of interest is the frequency range and how it is directed.

The most logical for the microphone is that it is not omni-directional, but points in approximately the same direction as the camera. This way the operator has an intuitive way to find out where the source of some noise comes from.

This can be done in mainly two ways, you can have a direct mono microphone pointing directly from the center of the camera, or you can use a stereo microphone that somewhat mimics the same listening field that humans have.

If you only have one microphone you would have to use use the way the intensity of sound changes as you move the camera. If you notice that the sound gets louder as you pan around, you can deduce that you are moving in the right direction if you want to find the source.

If on the other hand you have two microphones, you can use the relative intensity between the two sources to deduce where the sounds are coming from. This way you don't have to move the camera to get a guess at where the noises are coming from. In this way two microphones would probably more effective at localizing the sound source.

Something that may be challenging inside the nacelle is all the hard surfaces that can reflect sound this can make it somewhat harder to use the microphones to localize the source of some suspicious sound.

You can choose from different microphones with different directionality or polar pattern. The most natural to choose from would be either hyper-cardioid (Figure 5.1), super cardioid (Figure 5.2)or shotgun (Figure 5.3).

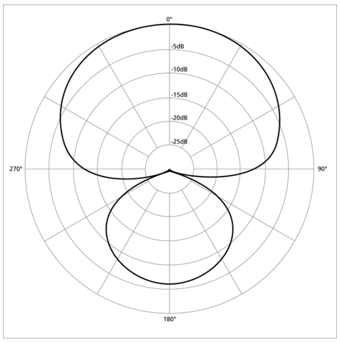


Figure 5.1: Hyper-cardioid pattern

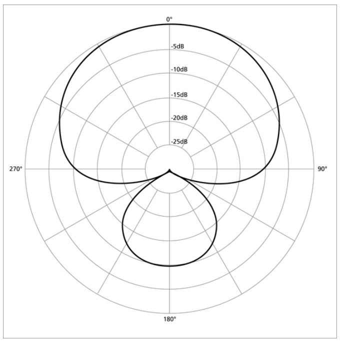


Figure 5.2: super-cardioid pattern

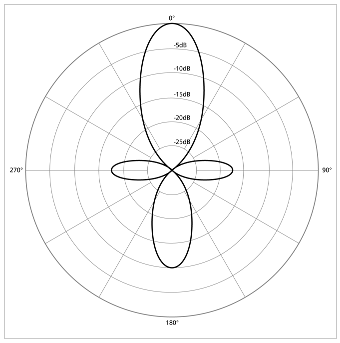


Figure 5.3: Shotgun pattern

The shotgun microphone offers really good qualities when it comes to being able to exactly determine where some sound is coming from. However it might be hard to use it to start localizing where the sound is coming from or indeed if there are any sounds of interest.

The hyper-cardioid or super-cardioid makes it easier to determine that there is something of interest to listen to. One problem that could arise is that if you only have one microphone it might be harder to exactly localize the source of some sound. If, on the other hand, you have two microphones with some overlap this might be easier. If one were to choose between the super-cardioid or the hyper-cardioid microphones it would probably be best to choose the super-cardioid as it would probably receive less interference from behind its primary direction.

5.3 Heat sensors

Two ways to get a temperature reading are going to be discussed in this section Thermal cameras and infrared heat sensors.

5.3.1 Thermal camera

When doing inspections inside a nacelle, it can be nice to have a thermal camera to detect abnormal temperature developments. Øyvind and Tor's had an excursion to Hunnhammarfjellet where they found that they did rely upon this camera during the inspection (A summary of this excursion can be found in [14]).

Such a camera can be used to find faults in electrical equipment such as bad connections, due to corrosion or short circuits in development. Perhaps to some extent it can be used to diagnose gears and bearings to and possibly also parts of wires that are losing their isolation.

When it comes to infrared radiation it is common to distinguish between five different wave bands:

- Near-infrared (NIR): 0.75 - 1.4 μm
- Short-wavelength infrared (SWIR): 1.4 - 3 μm
- Mid-wavelength infrared (MWIR): 3 - 8 μm
- Long-wavelength infrared (LWIR): 8 - 15 μm
- Far infrared (FIR): 15 - 1000 μm

While most objects at room temperature emits IR radiation in the whole spectrum there are peaks in certain regions and this depends on temperature. An object at room temperature mostly emits radiation in the range 8 to 25 μm. For this reason you usually use thermal cameras sensitive in the long-wavelength part of the spectrum.

Cameras in this range are usually constructed using uncooled microbolometers. This is a relatively new invention and previously to these cameras you had to use colling elements to cool the sensor chip and its surroundings to emit noise. The uncooled cameras are cheep relative to the cameras requiring cooling however they are still expensive. The cheapest camera from the FLIR manufacturer in their automation range of cameras cost 4210.00 £. (Figure 5.4)

Figure 5.4: FLIRs A300 IR camera (170x70x70mm 0.7kg) [12]

While sensors in ordinary cameras are sensitive to radiation in the NIR, range they are not as suited as thermal cameras. Mostly these cameras pick up reflections from light sources that emit radiation in the NIR part of the specter.

5.3.2 Infrared heat sensors

Infrared sensors are far cheaper than IR cameras. To some extent they can be used as a spot measuring tool like what the camera does, so this might be a good alternative to using the cameras. The IR sensor depicted in 5.5 has a ratio between the distance to the object and the diameter of measurements of approximately 6:1. The cost of this sensor is 185 USD. You obviously don't get the same level of detail with this as you get with a Thermal camera. It might be interesting to find out if an inspector can get the necessary information from such measurements. It will be harder to spot areas where you don't expect to find heat as a symptom of faults. This is the area where the thermal camera is best suited.

As the measurement area increases with the distance to the object. It makes sense for it to be placed on the arm and not simply on a cart. This also means that the arm needs to get closer to more points in the nacelle to get an accurate temperature reading of local areas.

Figure 5.5: FLIRs A300 IR camera (89mm length 19mm Outer diameter. Weight 0.18kg) [13]

5.4 Vibration

Vibration is easiest to measure with an accelerometer. Accelerometers now come in small chips and are used in many consumer electronics applications.

When choosing an accelerometer, the frequency range it is going to be used in is important to consider. In the condition monitoring system described in [6] they use different accelerometers between a range of 0 to 5 Hz and sensors operating at

1 to 20,000 Hz (these are not the only ranges used). The low frequency accelerometers is used to measure vibrations in the body of the nacelle and in some of the slow moving bearings, while the higher range accelerometers can be used in faster moving parts, such as the components that come after the gear exchange.

As I haven't done a comprehensive study of what frequency range it would be necessary to measure, I wont come with any recommendations as to what accelerometers can or should be used. Perhaps it will be necessary to have more than one accelerometer, or perhaps these things are not so interesting as vibration measurements are often provided trough condition monitoring systems. Instead of using a separate system you can present these data to an inspector perhaps even with historic data.

5.5 Voltage

To search for errors in electrical circuits, it is often desirable to measure voltages. The arm could be equipped with a probe on the end of the arm. If we can assume that the object we are measuring and the robot share a common ground we would only need to be in contact on one spot. If we can't assume this, then such an operation will become more difficult. You could potentially first fasten a grounding probe on whatever part you can consider to be ground relative to the point where you want to measure.

5.6 The Kinect

The Kinect is an exiting new sensor that has attracted a lot of attention in universities for its diverse set of sensors and applications. It is equipped with a RGB camera, a microphone array consisting of four microphones, an infrared camera operating in the NIR spectrum, and a laser that projects a dot pattern on the surroundings (Figures 5.6 and 5.7).

The IR projector and the IR camera are used to provide the Kinect with the capabilities to map its environment. Effectively, the Kinect becomes a 3D scanner [15].

The way this works is by doing image processing of the IR camera's video feed. By interpreting how the dots and patterns are displayed when reflected of different surfaces compared with a reference, the Kinect is capable of extracting geometry data of the environment. With this data you should be capable of constructing a virtual 3D environment.

This should make the Kinect capable of providing data for both the collision avoidance and for a augmented virtuality solution.

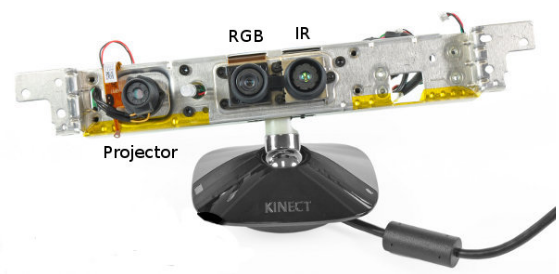


Figure 5.6: Exposed Kinect

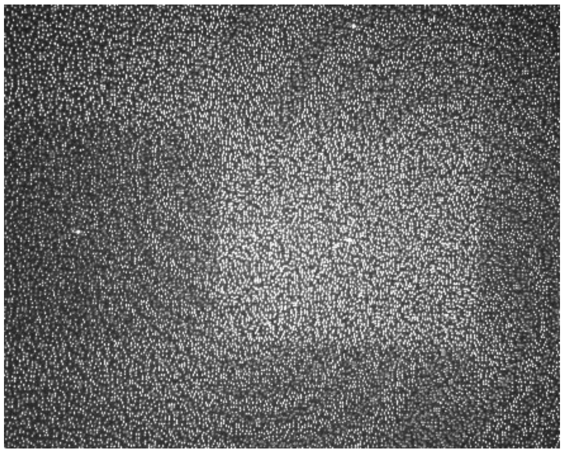


Figure 5.7: Part of the IR pattern from the Kinect

The IR and RGB cameras give a resolution of 640x480 pixels. This is however likely to be scaled up in the next version of the Kinect. Microsoft has announced that they are working on this, however they haven't released any specs on it.

5.7 Tools

When it comes to tools we are moving more into the maintenance and repair area. However, designing a tool and an arm to accommodate this might be very difficult and expensive. This should probably not be done before there is some certainty as to what objects the arm has to interact with and what tasks it should do.

In the summary from the excursion to Hunnhammarfjellet they found that sometimes, the inspectors used a tool for checking and adjusting the torque of bolts and nuts inside the nacelle. This tool was very heavy weighing around 30 to 50 kg. If we design the arm to be able to lift a tool such as this the arm would have to be very big and use strong motors and gears.

The arm might also have to be designed to handle the same amount of torque along its links and its motors as the bolts need to have. However in a nacelle especially designed with this in mind you might be able to have a hole or a pin close to the bolt you want to adjust that the tool can "hold" on to. An illustration of the tool end and a pin and bolt head can be seen in Figure 5.8

Until one know what specific tasks needs to be done and what objects the arm will have to interact with it is not advisable to design an arm with a specific tool. This either calls for a comprehensive study of the inside of a nacelle that is a likely candidate for implementing the system, or a natural evolution once a

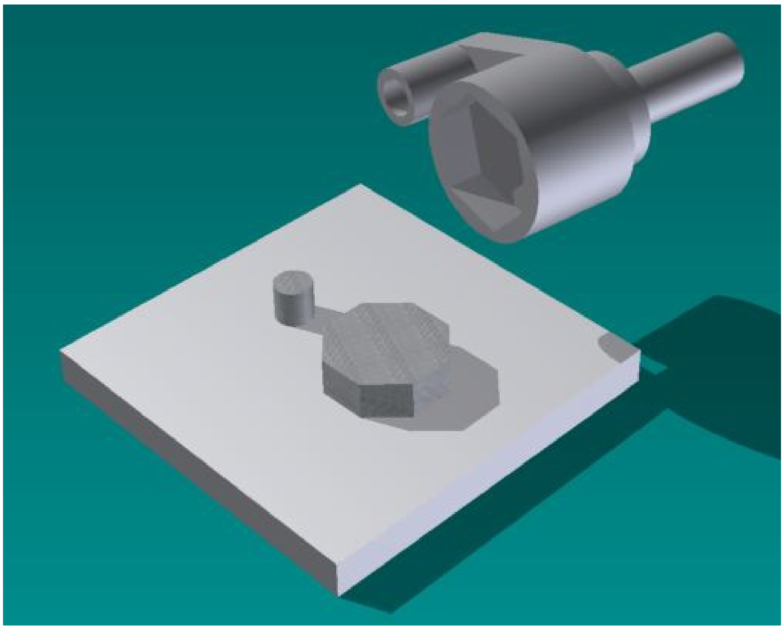


Figure 5.8: Illustration of tool end and bolt and pin.

simpler system is up and running and the necessary experiences with this system and the nacelle has been achieved.

As far as using the system to replace parts there are some rather large challenges to figure out before this is done. And it is one thing to design the system to do this but you also have to consider the parts, specifically the cost of parts that you would need to store on every wind turbine for this to be effective. This is not likely to happen unless the parts that are considered are inexpensive. You also need to consider how often these part break down and if there is money to be saved on this, before you start to think about deigning a system to handle this.

Chapter 6

Mechanical design

How the arm is to be designed is dependent upon what we require from it and where the rails are pleased relative to the points of interest in the nacelle. So first we need to see if we can at least establish some minimum requirements of what is needed from our system.

6.1 What is needed?

The ultimate goal of this project is to make a system that reduces the amount of on site O&M needed. The solution should make sens both from an economical point of view and with respect to health and safety.

At the excursion to Hunnhammarfjellet it was found that the wind farm had a monthly interval for a light inspection where all the tools they brought was simple hand tools such as screwdrivers and pipe-wrenches. They also used a high quality thermal camera. But some of the most important tools they bring are the inspectors eyes, ears and the possibility to feel vibrations.

The monthly inspection is not very elaborate, but every sixth month they have a more detailed inspection where they among other things bring a tool to check and adjust torque. This instrument is relatively heavy and weighing around 30 to 50 kg. A bi-annual full servicing schedule seem to be the standard for land based wind turbines [1]. This work is carried out based upon a check list and usually this is something the manufacturer of the wind turbine has created. The inspection involves checking for oil leaks, oil level, cable inspection, security of fixings for the blade, gearbox jaw bearing attachment, tower base-bolt. They also check break pads and disk, bearings, gears, cable terminations, pitch calibration, oil filters, and more.

But what do we actually need to do this? The system that has been created so far consists of a rail, and some carts capable of moving around on this rail, with a camera mounted on one of the carts. This means that the operator can only get viewing angles from where the rail is. Rails have to be placed in such a way that they don't cause a significant limitation to the personnel doing on site O&M. The rail and cart system is also limited by how large the bends have to be. Will such a system be sufficient? After all personnel doing on site O&M don't have to labor under such constraints as they can move relatively freely inside the nacelle. The only limitation they have is to not get themselves in to a position that poses a danger to themselves or any component of the wind turbine.

A robotic arm with a camera, mounted on one of the carts can extend the viewing angles and positions the operator has to decide whether or not there are anything wrong with the wind turbine or the maintenance system itself. For certain points of interest it might even be necessary to have an arm. This would be the case if you can't place the rail in such a way that you can get to it in any other way.

It can further be equipped with tools to support basic maintenance tasks. Depending on design and the rails you might end up with a system that exceeds what it can do in terms of mobility compared with what humans can do. However what thees tools should be I don't yet know.

In the next sections we will look at what the system should and must do in terms of the constraints we must put on it.

6.2 Rail

As far as the rails go there are many constraints and requirements on it:

- It is constrained to go where it will not be obstructing normal operation of the wind turbine.
- It can't obstruct doors or cover holes that must be opened for inspection.
- It should not significantly obstruct the change of the gearbox or other parts that one would suspect to have to exchange or have to remove.
- It should not obstruct parts in need of on site inspection or refurbishments.
- It should not significantly obstruct people working inside the nacelle.
- It should be close enough to the points of interest for inspection with either the cart and camera or with the arm.
- It must be possible to bend the rails in such a way that it can reach the points of interest.

- The bends in the rail must not be so small that the cart cant travel past them.
- If it is possible for somebody to fall onto it it must withstand the impact without permanent deformation, or if it deforms this must not make it impossible for the cart to pas.
- The rail must not permanently deform when the arm loaded to its maximum limit in its most extreme position. That is when the arm is at its maximum extension.
- The rail support must be able to withstand the weight of the rail and also the forces of the arm and cart.
- The rail must probably deliver power to the cart and arm.

As can be seen there are many things that constraints where the rails can and should go. Finding the optimal path for the rail is not easy.

6.3 Cart

How the cart is made is naturally largely dependent upon the shape of the rail. but also dependent upon the arm.

- The cart must be able to withstand the forces that the arm can exert on it.
- The bearings and wheels must have a long life span.
- The dimensions of the cart must be large enough to handle the bends of the rail.
- The propulsion system should not cause much vibration.

The cart and rail design developed by Torgeir Welo, does look promising as far as providing a sturdy platform for the arm. The work of calculating the correct thickness on the different parts. so that they withstand the forces exerted on it still remains though.

The requirement on the life span comes from the fact that the system as a whole should not ad significantly to the maintenance work or increase the inspection or maintenance schedule.

6.4 Arm

When it comes to the arm there are many questions that It would be nice to have an answer to. The main ones being

- How long does it have to be?
- How many degrees of freedom are needed?
- what sensors or tools do wee need?
- Does it need to be able to lift any thing?
- How much money can we use on it?
- How doe we make certain that it does not crash?
- How much power is needed?

The questions of length an degrees of freedom relates in large parts to where the rails can go and where the points of interest are. It is also dependent upon what tasks we need it to perform, which again should answer the questions of what sensors are needed and If it needs to lift anything else.

The question of what tasks need to be performed is largely an economical question. It is a balance of what one can do and how much you stand to save on doing this without bringing personnel to the windmill. and the cost of the system. There might also be certain tasks that need to be performed to extend the period between on site work. As an example you might build a system that is capable of doing x, y and z of the tasks normally performed during an inspection but if it is seen as necessary to also do a and b and our system can't handle thees task this might mean that on site work is needed with the same interval as before.

As Jeremias is working on finding out what is done during inspections hopefully this will be clearer when he has finished hi master thesis.

The question of economy when it comes to the arm is something that needs to be analyzed. A model consisting of doing O&M for certain tasks with a robotic arm can be compared with the "competing" model where all O&M is done on site. This will however not be threated in this report.

We can of course try to intuit answers to these questions and then prosed to design an arm. It would be interesting in any case to see how much we should expect to spend on the arm.

6.5 Arm construction

In this section I will develop an arm to see what is needed. First we must decide what sensors we want then we must decide how many degrees of freedom we need and what consequences this will have. The length of the arm is also important. From this it is possible to find the correct motors and gears.

6.5.1 Sensors on the end effector

How the end effector is built and what is put on it has large consequences for the construction of the arm. The more weight you put on the end of the arm the larger the motors along the arm has to be.

The camera I will consider using is the one that Hong Bui has used with the Pandaboard. This camera is equipped with microphones, so lets assume that we can use this to.

It might also be prudent to give the arm a light shining in the same direction as the camera. It should be no problem to do this with a white LED light. As the life span of LEDs are quit long and don't easily break LED's seems appropriate choice.

It might be necessary to measure spot temperatures. For inspection there are in many cases good to have a thermal camera as has been discussed previously. However let us assume that we can already know what parts of the wind turbine there might be temperature spikes as an error is about to manifest, so that we can simply measure these areas with the Ir sensor discussed in 5.3.2 and we don't need to locate these spots as you would be able to do with a thermal camera. This could potentially be set up to be performed automatically with warnings popping up if something is abnormal.

The weight of all the sensors comes up to about 350 g where I use the weight of the IR sensor of 180 g, and assumed that when the base of the camera is removed it comes to 150 g. The LED light and electronics comes up at 20 g.

6.5.2 DoF and length

How many degrees of freedom the arm have is important as this has consequence when it comes to the overall weight, and also the cost as you need more motors and encoders.

We will first consider how many DoF we need in the arm and then threat the end effector separately.

If the arm has a base that can rotate and a joint with a allowable rotation of 180 degrees, attached. The end effector will be able to reach any point on a semi sphere where the radius is defined by the length of the joint. This may actually be a solution one could consider at least if the end effector is given three DoF. This will allow the system to get more viewing angles than with just the cart and camera. However, this does greatly limit how useful the system can be and this can also be limiting in terms of further development.

There is also a different concern with this. The arm has to travel along the track and if the arm has a great length this may become problematic in bends and turns. This is due to the arm sticking out beyond the rails.

If we add more joints the arm becomes far more maneuverable and the arm can fold up so that it does not become a problem in bends. Having many joints the arm may also be able to get into even stranger positions and look past obstacles that otherwise would not be assessable with just a single or even two joints.

Having many joints does however become more expensive due to the demand for extra actuators and more powerful actuators due to the added weight. While it may be useful we don't really know that it will be useful in our system. An arm with two joints is what I think we should consider for the arm we are going to make an example of here. With two joints you are able to reach not just the points on the outside of a semi sphere but also inside it and in an arch below it. An example to illustrate this can be found in Figure 6.1.

The number of DoF in the end effector should at the very least be two. The sensors would then be given the opportunity to point in any direction in a semi sphere, on the end of the arm. One motor to rotate the sensors tilt and one to pan the sensors around. However one can also argue that you would want three DoF on the end effector. You would then be able to let the view from the camera maintain a specific angle on the camera view almost no matter what position the arm is put in.

The best argument for using three DoF on the end effector is that it might be easier for the operator to orient himself if the camera is not skewed. Especially if the camera angle puts the operator upside down in the nacelle. This might not be so large a problem though. And can perhaps to some extent be solved by flipping the image.

For this arm we will consider the use of two DoF on the end effector.

If we assume we can use two relative light weight motors with some type of gear exchange to move this the total weight of the end effector, with sensors supporting structure can come some where around 450 g.

6.5.3 Length of the joints and construction of links

A matter it is very hard to decide upon at this moment is the length of the arm. We would like the arm to be able to reach or see all points of interest. But at the moment we don't know exactly where they are or where the rails can go in relation

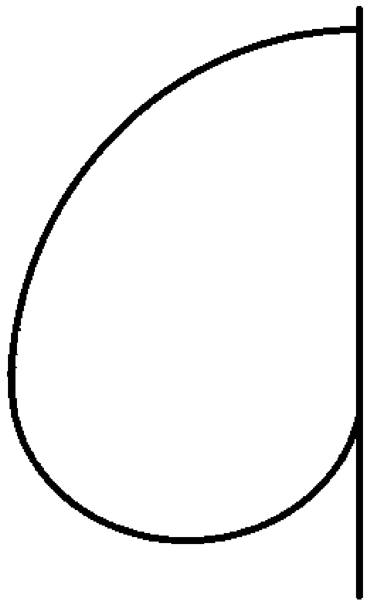


Figure 6.1: Illustration showing the reach of a robot arm with two joints of the same length. If you visualize the form rotated about the axis you get the shape of the points the arm can reach.

to them. It may of course be that at some point we will just have to decide upon some length that we think will give us what we want with out optimizing it.

In the arm under consideration here I will do just that. The length of the first joint will then be 20 cm. While the second joint will be 30 cm. We can assume that the end effector will be going approximately 7 cm beyond this giving a total length of 57 cm from the base.

The links we consider her will primarily consist of carbon fiber tubes. They should be strong enough to withstand the weight of the arm. And here they will act as an exoskeleton. providing both coverings for the parts and the structural strength needed. we shall consider tubing with a dimension of 4 cm diameter and a thickness of 2 mm. The first link will then be approximately 120 g and the second link 150 g. And the center of gravity will approximately be in the middle of the part. a Denavit Hartenberg representation of the arm can be seen in Figure 6.2

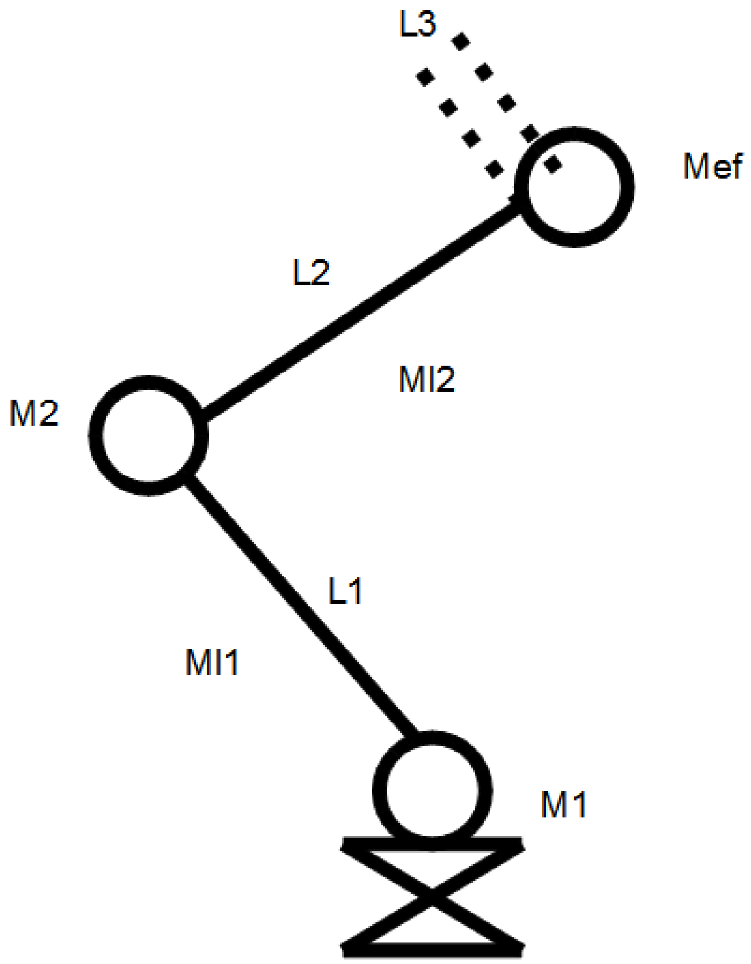


Figure 6.2: Denavit Hartenberg representation of the arm.

6.5.4 Motor's for use in the arm

Now that we know the wight and length of the parts of the arm it's time to consider the motors used for the arm. They must have enough torque to be able to handle the weight of the arm, and also to accelerate the links of the arm. Motor two in 6.2, must together with friction at least be able to hold the weight of the end effector while Motor one is accelerating the arm. The torque needed in motor two with a specific gear ratio δ can be found with:

$$\tau_2 = \frac{(L_3 + L_2)M_{ef} + \frac{L_2}{2}M_{l2} + (M_{l2}(\frac{l_2}{2})^2 + m_{ef}(l_2 + l_3 2)^2)\ddot{\theta}_2}{\delta}$$

For the second motor the equation for the torque becomes:

$$\tau_1 = M_{l1}\frac{L_1}{2} + L_1M_2 + M_{l2}(L_1 + \frac{L_2}{2} + M_{ef}(L_1 + L_2 + L_3) + (M_{l1}(\frac{L_1}{2}^2) +$$

$$M_2L_1^2 + M_{l2}(L_1 + \frac{L_2}{2})^2 + M_{ef}(L_1 + L_2 + L_3)^2)\ddot{\theta}_1$$

If we use the weights and lengths from the previous sections, and a acceleration of joint two set to 10 deg/s^2 we get τ_2 =0.226 Nm If we find a motor and a gear that can deliver this at a weight of 300 g the second motor with acceleration set to 7 deg/s^2 will have to have a torque of 0.471 Nm

I have not considered the added weight and friction from the wiring here. This means the motors might have to be stronger. Also it can be interesting to get faster accelerations,.

The conditions of the arm moving along the rails have neither been considered here. This might also led to a need for more torque in motors one and two, and thus added weight and expense, as servos are not cheap.

The motor torque in the base has neither been calculated. However, this is also something that needs considering. This joint might not have to hold the load of the whole arm outstretched like the other motors instead this might be done with passive elements such as bearings. This is dependent upon how the rails are lain out. If the cart never has to go up and down the motor will never have to carry the load of the arm only counteract the friction.

In the next section we will look at something that may help us figure out how long the arm needs to be.

Chapter 7

3D Models

To help in deciding on the physical dimensions of the arm and also to see some of the challenges, and the benefits involved in using a robotic arm, I wanted to make some 3D models of the arm and its environment. This does also make it easier to communicate what ones ideas.

7.1 3D model and Simulink

The first ting I wanted to do was to couple a real time simulation of the arm with a 3D model in Simulink. However as Simulink don't really have a simulation mode that offers you a direct means to run a simulation at real time in a straight forward manner, I eventually abandoned this idea. However I had already started to constructed a virtual environment and was prepared to couple this with Simulink, so this has been done.

Simulink offers a library for coupling a 3D virtual world, described in a VRML file, and input from a simulation or other Simulink blocks. This library is called "Simulink 3D Animation". The diagram, code and an explanation to how this works can be found in appendix A and B, and also available on the DVD.

When this is simulated it opens a viewer that lets you see the progression of your simulation. This viewer is distributed with MATLAB. It is capable of capturing video and also rendering images (Figures 7.1 and 7.1). Two videos have been made of this model while taking input from Simulink. They can be found on the DVD-ROM.

The Model was created in V-Realm Builder. This program was the best available to create models in the VRML file format. I did try to see if i could use other programs, however this was the best one. VRML is an old format and does not appear to be maintained or had any major changes since the late 90s. The format was intended as a way to show 3D worlds and animations on the web but the X3D

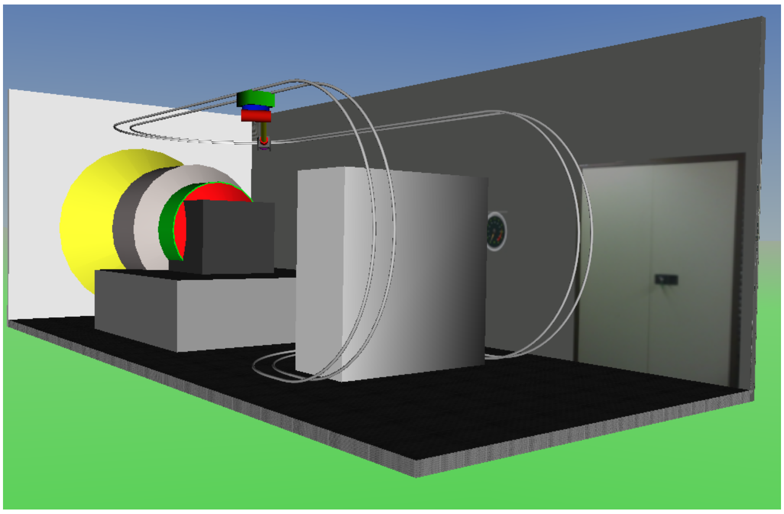


Figure 7.1: Image rendered of the virtual world.

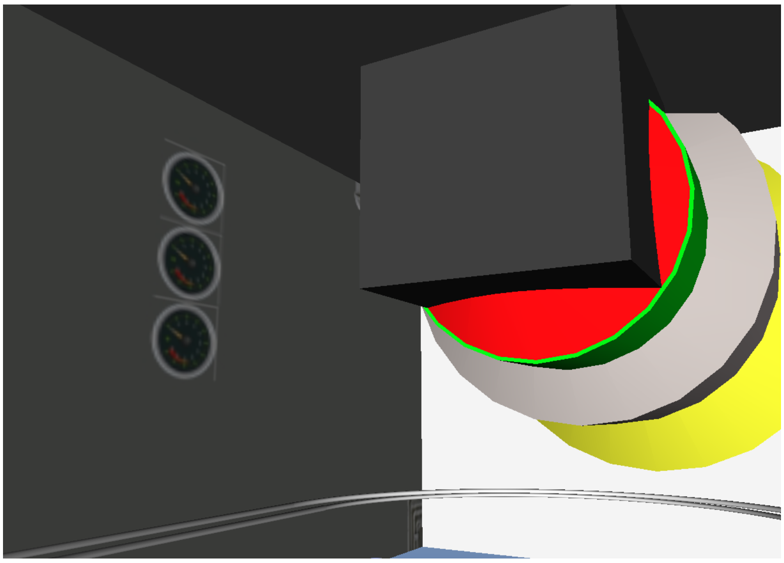


Figure 7.2: Image rendered of the virtual world from the point of view of the camera on the robot.

format has replaced it. VRML is however the only format that the Simulink 3D Animation blocks accept as input.

The V-Realm builder software has like the VRML format not seen much in the form of updates. And while I did manage to create the shapes that I wanted for this project, most importantly the rails. It is somewhat cumbersome to do and making bends that only extends to one quarter of a circle would be close to impossible with the software. The file format does however support doing so. You could go into the file and do this textually.

However if you want to create more advanced geometry you would have to use a more advanced software.

7.2 Inventor

Inventor is a powerful and also very expensive tool. It is meant to be a rapid prototyping mechanical CAD software. The benefits of this software is that you can quickly create parts and see how they fits in with your design. Many companies use these types of tools to cut down on the amount of prototyping iterations. Often you can go straight from your 3D design phase to a finished product in one iteration. That is without making any prototypes. Needless to say, this can cause great savings in R&D.

As mentioned this is an expensive tool costing around 5000 USD however students gets a 14 months free license for this program and 30 other programs form Autodesk. There is also a cooperation between different faculties and institutes on NTNU for license on Inventor and other programs. The institute of Technical cybernetics is one of the institutes that are part of this deal [21].

The software has many tools for creating 3D models. You usually create a sketch and then use one of the tools to get 3D shapes from this sketch. The different parts you create can then be assembled in an assembly file where you put constraints on the parts so they either stay put or move only in the allowed directions.

A design iteration in Inventor can then be done in such a way that you make a general shape for a part with some dimensions you think will be correct. Then you put the part in to the assembly where you want it. you can now see if it fits here. After you are satisfied with this you can use the part you created mold for creating part of the components needed to get the same dimensions as the "mold" part. In this way you can quickly experiment with different designs, will not having to worry to much about how this should be created.

Inventor also allows you to do analysis of the strength of your parts and can in this way help you with deciding upon different structures and also on if the part created will be able to withstand the forces put upon it.

Once you are satisfied with the design, and have done a check that the parts can be created and subsequently assembled in real life, you can easily make drawings to document your design simply by projecting them onto a drawing and then annotating the parts, easily annotating the dimensions of your parts and so on. An example of this can be found in Appendix D. You can also use the models for automatic machining of parts.

Inventor also let you create renderings of the parts you have designed. You can even make videos to help communicate what you have designed. The picture on the front page is a rendering of a wind turbine I created in Inventor.

7.2.1 Work done in Inventor

Figure 7.3: Photo of 5 MW wind turbine (picture taken from Wikipedia)

Figure 7.4: Wind turbine modeled in Inventor

I initially wanted to use Inventor to design the arm and have good documentation for It. But as the work progressed making the arm at this stage made less and less sense. That is not to say that I believe that a robotic arm is an inappropriate thing to have inside the nacelle, only that I think that there is more work to be done concerning major design decisions such as how many links are needed in the arm, how long it should bee and also what sensors and tools are needed.

As part of the process of finding answers to the question of how many links are needed and how long it should be I created a model wind turbine. I did this by looking and making measurements of a photo of a five megawatt wind turbine that I knew some of the dimensions of (Figures 7.3 and 7.4).

Modeling a 5 MW wind turbine seems like a logical choice as the trend seems to be to make big wind turbines at sea, as these are considered to be more economical.

The main purpose of this was to get a estimate for the size of the nacelle. Once the outer dimensions agreed fairly well with what i could see from the photo I started to work on the inside of the nacelle. The last ting i did was to place a monorail with a profile developed by Torgeir Welo, and a cart and robot arm. The result can be seen in Figure 1.1.

Modeling the wind turbine has been helpful when it comes to reasoning about how the system might work and what limitations and constraints the arm will labor under. But the model can also be used in future work. As there are still some questions that remain unanswered when it comes to constructing the arm, it can be useful to use the models to run simulations of possible arm configurations to check how useful different configurations can be.

What I propose is that you can run one computer that simulates the physical signals from the arm and hock this up to the hardware that controls the arm and handles sensor signals such as camera, microphone etc. The sensory data and arm position can then be sent over a wireless network to a WLAN router that sends the data to the computer running the user interface. In this way we can combine Hardware in loop[1] and a simulated virtual 3D World.

In the user interface you can either animate the computer model directly or let the data on joint positions sent from the system animate the arm.

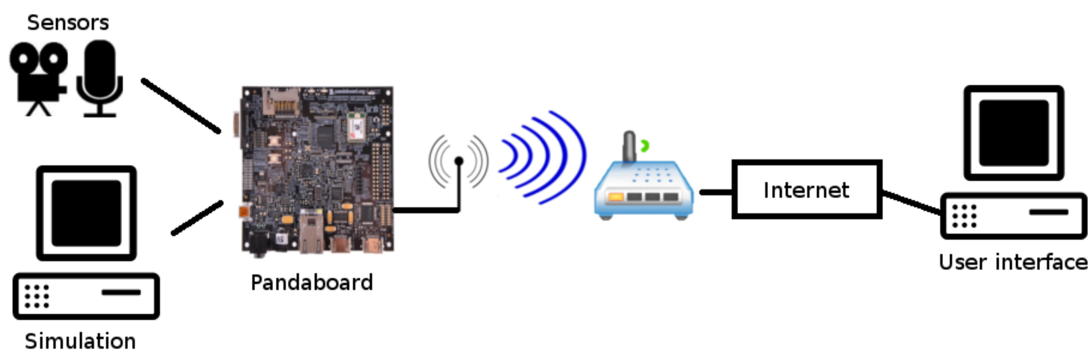


Figure 7.5: System with 3D visuals and Hardware in loop

The advantage with this is that you are likely to get closer to finding the correct arm geometry and you Will be able to design a user interface and experiment

[1]Hardware in loop is a term that is used when you use a computer to generate the signals ordinary hardware would give you. Ex. Using a simulated sensor to generate the signals you want to measure

with different augmented virtuality concepts. You can also make the necessary controllers and you can gradually replace the simulation with physical objects (motors encoders etc.).

To provide the system with the capabilities of showing a 3D model I propose the use of Panda 3D which is a Game engine that is free to use even commercially. The engine is made to be used with Python and C++ and boasts of it's short learning curve, and an active community. It accepts geometry data from Blender[2], 3ds Max and Maya. As Maya and 3ds Max are products from Autodesk they are available for free for students for 14 months. 3ds Max and Maya can import geometry data directly from Inventors native files.

It should be possible to use some of the work done with the 3D models to help solve the problems discussed in chapter 3, and 4.

[2]Blender is a Open source 3D drawing program

Chapter 8

Conclusion

A robotic arm for use inside a nacelle can be constructed to do many useful things. However, at the moment the best option seems to do further work to find out what tasks this should be and how long the arm need to be, before one start to construct an arm.

Some arguments for using an arm, is that you will be able to get to more points of interest, either when it comes to getting a view of the object or spot, or to measure vibrations and Ir radiation. This may be coupled with using mirrors inside the nacelle to get a fuller picture of the nacelle.

It should also be useful to do virtual tests of how the arm should be controlled, and how easy this will be to do. In the telepresence chapter we looked at some of the challenges that present themselves when it comes to manual robot control in a remote presence system. An augmented virtuality scheme that either shows 3d representation of the robot in a separate window or where the video-feed is projected on a screen in front of a 3D representation of the robot, seems the best choice as far as help in navigating the robot is concerned.

The augmented virtuality system should be able to help in navigating the robot, but it does not guaranty that the robot don't crash. The possible damage to the robot or the components of the nacelle warrants the use of some sort of collision avoidance scheme.

The best option here seems to be to use a 3D representation of the inside of the nacelle to counter the problem of collisions. If you use some form of 3D scanning devise you don't need to buy any data from manufacturers. This can be a big advantage if the system is deployed commercially.

The Kinect seems an appropriate candidate. It is so cheap that it might even be possible to consider leaving it inside the nacelle. The data from a scan might also be used in the augmented virtuality system.

The models created should be possible to use in the future.

8.1 Future work

The best option for now seems to be to use the models in an virtualization of the system inside a virtual nacelle. Using a game engine to display the arm and nacelle, and possibly developing an augmented virtuality system. This can be combined with modeling one or more robot arms, and running a simulation of this, that provide input to the electronic hardware so that controllers and other hardware, and code can be developed before the arm is made. This should make the quality of the final arm much better. Also I believe this work plays more to my strengths than mechanical engineering does, as parts of my future master thesis. In this way you combine Hardware in loop and a virtual world.

If this is setup using a model that is close to, or exactly like how a nacelle is equipped inside the quality of the work will improve greatly.

This should help determine what the physical dimensions of the arm should be. It should also be possible to get a good user interface from this. Where the models and code used for the simulation, and visualization can be used in the final system.

It will also be easier to communicate how the system is intended to work to interested parties.

This should ensure both quality and fewer prototyping iterations and thus it should also make more sense when it comes to economy.

Bibliography

[1] Jesse Agwandas Andrawus. *MAINTENANCE OPTIMISATION FOR WIND TURBINES*. PhD thesis, Robert Gordon University, 2008.

[2] Peter Caselitz and Jochen Giebhardt. Rotor condition monitoring for improved operational safety of offshore wind energy converters. *Journal of Solar Energy Engineering*, 127(2):253–261, 2005.

[3] Jill L. Drury, Brenden Keyes, and Holly A. Yanco. Lassoing hri: analyzing situation awareness in map-centric and video-centric interfaces. In *Proceedings of the ACM/IEEE international conference on Human-robot interaction*, HRI '07, pages 279–286, New York, NY, USA, 2007. ACM.

[4] Olav Egeland and Jan Tommy Gravdahl. *Modeling and Simulation for Automatic Control*. Marine cybernetics AS, 2002.

[5] N. Elkmann, T. Felsch, and T. Fo andrster. Robot for rotor blade inspection. In *Applied Robotics for the Power Industry (CARPI), 2010 1st International Conference on*, pages 1 –5, oct. 2010.

[6] DG TREN European Commision. advanced maintenance and repari for offshore wind farms using fault prediction and condition monitoring techniques (offshorem&r). Technical report, ISET, 2003 - 2005.

[7] J.T. Feddema and J.L. Novak. Whole arm obstacle avoidance for teleoperated robots. In *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pages 3303 –3309 vol.4, may 1994.

[8] Viktor Fidje. Remote presence on offshore wind turbines. Master's thesis, NTNU, 2010.

[9] Terrence Fong, Charles Thorpe, and Charles Baur. Advanced interfaces for vehicle teleoperation: Collaborative control, sensor fusion displays, and remote driving tools. *Autonomous Robots*, 11:77–85, 2001. 10.1023/A:1011212313630.

[10] D. Gandhi and E. Cervera. Sensor covering of a robot arm for collision avoidance. In *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, volume 5, pages 4951 – 4955 vol.5, oct. 2003.

[11] Stephen Hughes and Michael Lewis. Robotic camera control for remote exploration. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '04, pages 511–517, New York, NY, USA, 2004. ACM.

[12] `http://thermalvisiondirect.com/epages/436721.sf/en_GB/?ObjectPath=/Shops/436721/Products/42701-1001UK`.

[13] `http://www.omega.com/ppt/pptsc.asp?ref=OS136&ttID=OS136&Nav=`.

[14] Tor Mehlum Karlsen. Remote presence for offshore wind turbine. Master's thesis, NTNU, 2011.

[15] `http://idav.ucdavis.edu/~okreylos/ResDev/Kinect/index.html`.

[16] Larry Kutz, Ravindran Rajagopalan, and Kofi Nyamekye. An ultrasonic collision detection system. *The International Journal of Advanced Manufacturing Technology*, 7:93–100, 1992. 10.1007/BF02601575.

[17] D. Labonte, P. Boissy, and F. Michaud. Comparative analysis of 3-d robot teleoperation interfaces with novice users. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 40(5):1331 –1342, oct. 2010.

[18] Bin Lu, Yaoyu Li, Xin Wu, and Zhongzhou Yang. A review of recent advances in wind turbine condition monitoring and fault diagnosis. In *Power Electronics and Machines in Wind Applications, 2009. PEMWA 2009. IEEE*, pages 1 –7, june 2009.

[19] V.J. Lumelsky and E. Cheung. Real-time collision avoidance in teleoperated whole-sensitive robot arm manipulators. *Systems, Man and Cybernetics, IEEE Transactions on*, 23(1):194 –203, jan/feb 1993.

[20] C.W. Nielsen, M.A. Goodrich, and R.J. Rupper. Towards facilitating the use of a pan-tilt camera on a mobile robot. In *Robot and Human Interactive Communication, 2005. ROMAN 2005. IEEE International Workshop on*, pages 568 – 573, aug. 2005.

[21] `http://www.ntnu.no/adm/it/brukerstotte/programvare/autodesk`.

[22] `http://en.wikipedia.org/wiki/Parallax`.

[23] B. Ross, D. LaRose, H. Wilde, and L.D. Baker. High performance teleoperation for industrial work robots. In *Applied Robotics for the Power Industry (CARPI), 2010 1st International Conference on*, pages 1 –5, oct. 2010.

[24] Bill Ross, John Bares, David Stager, Larry Jackel, and Mike Perschbacher. An advanced teleoperation testbed. In Christian Laugier and Roland Siegwart, editors, *Field and Service Robotics*, volume 42 of *Springer Tracts in Advanced Robotics*, pages 297–304. Springer Berlin / Heidelberg, 2008. 10.1007/978-3-540-75404-6_28.

[25] `http://en.wikipedia.org/wiki/Telepresence`.

[26] Martin Voshell, David D. Woods, and Flip Phillips. Overcoming the keyhole in human-robot coordination: Simulation and evaluation. *Human Factors and Ergonomics Society Annual Meeting Proceedings*, 49(3):442–446, 2005.

[27] Edwin Wiggelinkhuizen, Theo Verbruggen, Henk Braam, Luc Rademakers, Jianping Xiang, and Simon Watson. Assessment of condition monitoring techniques for offshore wind farms. *Journal of Solar Energy Engineering*, 130(3):031004, 2008.

[28] D.D. Woods, J. Tittle, M. Feil, and A. Roesler. Envisioning human-robot coordination in future operations. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 34(2):0 –218, may. 2004.

[29] H.A. Yanco and J. Drury. Classifying human-robot interaction: an updated taxonomy. In *Systems, Man and Cybernetics, 2004 IEEE International Conference on*, volume 3, pages 2841 – 2846 vol.3, oct. 2004.

[30] Jie Zhu, Xiangyu Wang, and M. Rosenman. Fusing multiple sensors information into mixed reality-based user interface for robot teleoperation. In *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*, pages 876 –881, oct. 2009.

Appendix A

Simulink diagrams

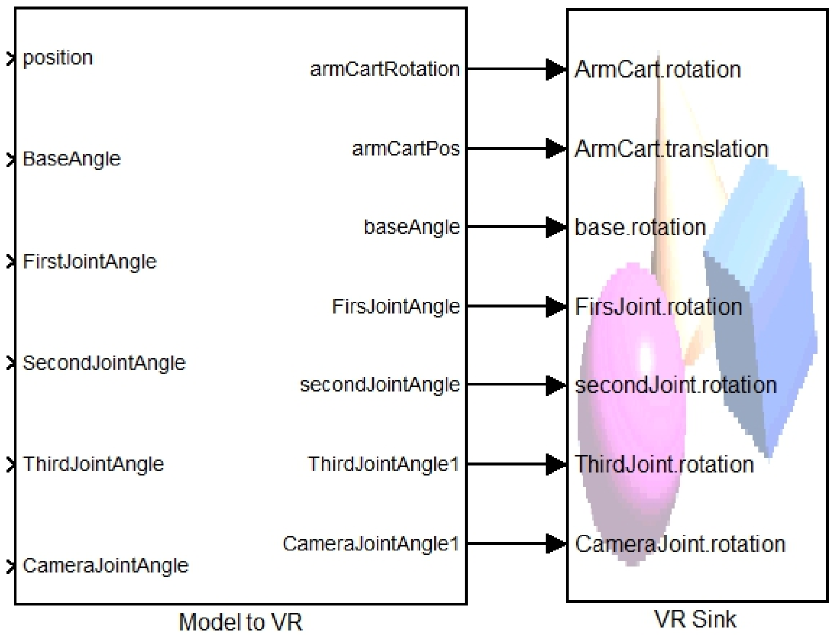


Figure A.1: Top level diagram

The diagram shown in Figure A.1 is the top level diagram. Here you can see the Model to VR diagram that I made. The inside workings of this block can be seen in Figure A.2 and continuing in Figure A.3. The other block in is the VR Sink block. This block is provided by the Simulink 3D animation library. It is configured by first loading the VRML file and then you can choose to animate the parts of that model that are given unique names when the model is created. Her this means translation and rotation of the whole arm and angular control over all the joints of the arm.

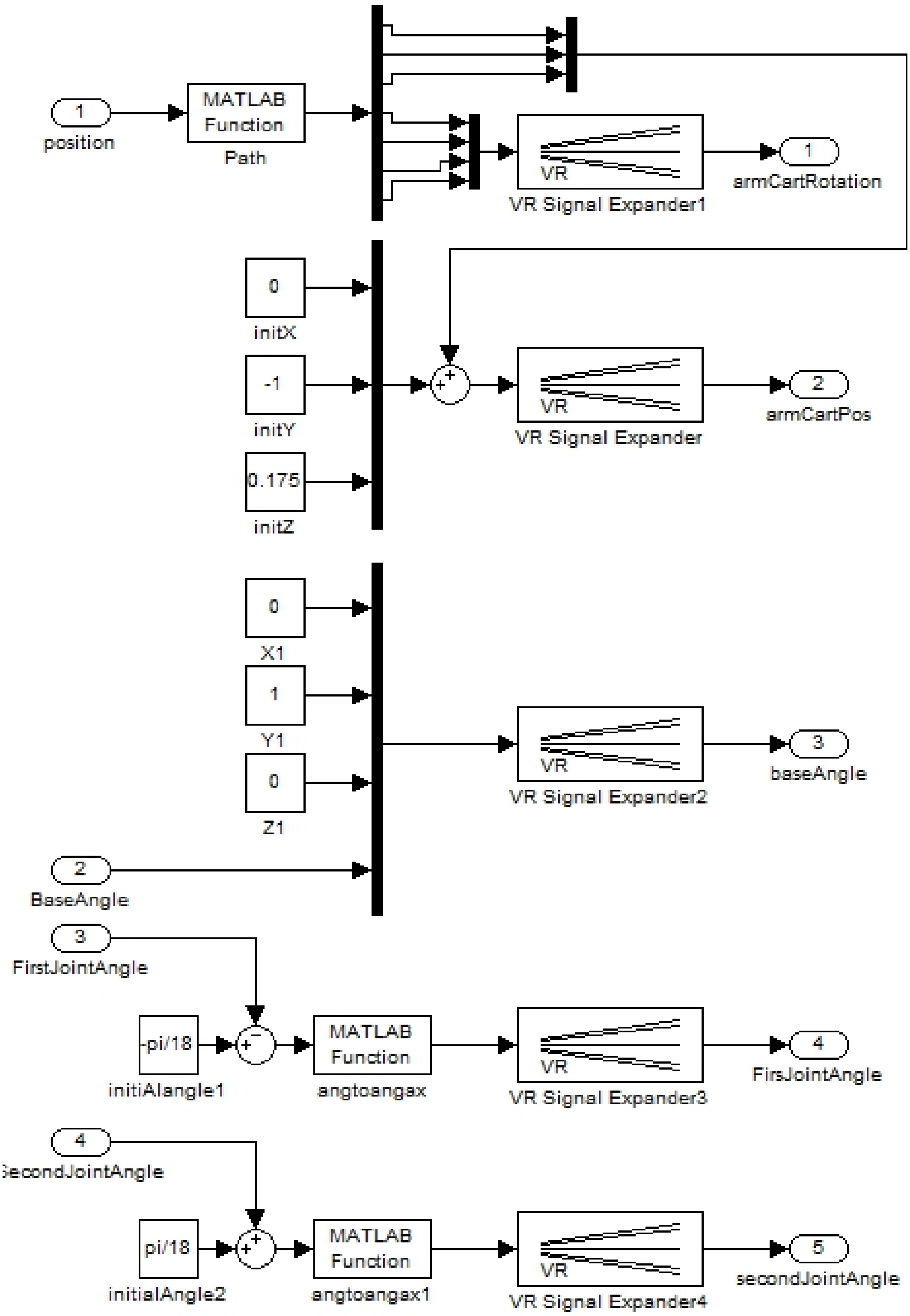


Figure A.2: Model to VR diagram part 1

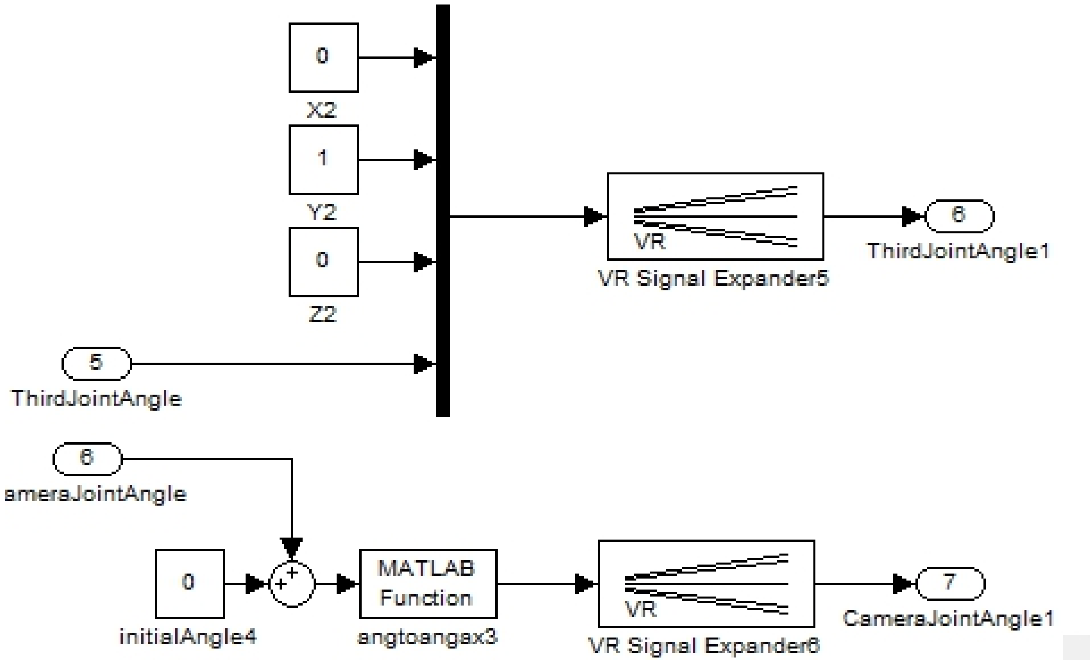


Figure A.3: Model to VR diagram part 2

The model to VR block shown in A.2 and A.3, Is both responsible for placing the robot in the right position along the path and controlling the position of the arm. The VR Signal expander block makes certain that the data vector entering into the VR Sink is in a correct format. The Path and angtoangaxis blocs are MATLAB scripts that will be explained in the code appendix.

Appendix B

Code

B.1 code for path generation

```
function [pos] = path(u)
    p=mod(u, 18.195131716320700);
    if p < pi && p >= 0
        pos = [sin(p) 1-cos(p) 0 0 0 1 p];
    elseif p >= pi && p < 6.141592653589793
        b = p-pi;
        pos = [-b 2 0 0 0 1 pi];
    elseif p < 9.047565858160352 && p >= 6.141592653589793
        b = p-6.141592653589793;
        pos = [-3+0.925*sin(-b/0.925) 2 0.925-0.925*cos(b
           /0.925) atoaa(0, -b/0.925, pi)];
    elseif p >= 9.047565858160352 && p < 12.047565858160352
        b= p-9.047565858160352;
        pos = [b-3 2 1.85 -1 0 0 pi];
    elseif p < 15.189158511750145 && p >=
       12.047565858160352
        b = p-12.047565858160352;
        pos = [sin(b) 1+cos(b) 1.85 atoaa(pi, 0, -b)];
    elseif p < 15.339158511750146 && p >=
       15.189158511750145
        b = p-15.189158511750145;
        pos = [-b 0 1.85 0 1 0 pi];
    elseif 15.189158511750145 <= p && p <
       18.095131716320703
        b = p-15.189158511750145;
```

```
            pos = [-0.15-0.925*sin(b/0.925) 0 0.925+0.925*cos(b
                /0.925) 0 1 0 pi-b/0.925];
        else
            b = p-18.095131716320703;
            pos = [-0.15+b 0 0 0 0 0 0];
        end
end
```

This function takes in the position, in the form of a scalar describing the length traveled from the initial position. The function tests the argument for which part of the track it should draw the cart and then calculates position or translation in x, y, z. As the VRML format uses angle axis representation to describe rotation the atoaa function is used to translate from the three angle representation of roll, pitch and yaw to the angle axis representation.

B.1.1 From roll, pitch, yaw to angle axis

```
function [u]= atoaa(x, y, z)
     R=rotzr(z)*rotyr(y)*rotxr(x);
     [n, e]=shepperd(R);
     [fi, u]=euParToAngAx(n, e);
     u(4)=fi;
end
```

The function first calculates the rotation matrix using th functions in B.1.2. It then presides to calculate the Euler parameters with Shepperd's algorithm B.1.3. The Euler parameters can easily be transformed to the angle axis representation and this is done by B.1.4.

B.1.2 Rotation matrix calculation

```
function [Rmat]=rotzr(angle)
     Rmat=[cos(angle) -sin(angle) 0
           sin(angle) cos(angle) 0;
           0   0   1];
end

function [Rmat]=rotyr(angle)
     Rmat=[cos(angle) 0 sin(angle);
           0   1  0;
           -sin(angle) 0 cos(angle)];
```

```
end

function [Rmat]=rotxr(angle)
    Rmat=[1 0 0;
          0 cos(angle) -sin(angle);
          0 sin(angle) cos(angle)];
end
```

These functions return a rotation matrix based upon the angle given. The angle must correspond to a rotation around the correct axis depending upon what function you use.

B.1.3 Shepperd's algorithm

```
function [n, e] = shepperd(R)
    T=trace(R);
    index=0;
    m=T;
    for j=1 : 3
        if m < R(j,j)
            index=j;
            m=R(j,j);
        end
    end
    if index == 0
        n=sqrt(1+T)/2;
        e(1)=(R(3,2)-R(2,3))/(n*4);
        e(2)=(R(1,3)-R(3,1))/(n*4);
        e(3)=(R(2,1)-R(1,2))/(n*4);
    elseif index == 1
        e(1)=sqrt(1+2*R(1,1)-T)/2;
        n=(R(3,2)-R(2,3))/(e(1)*4);
        if n < 0
            e(1)=e(1)*-1;
            n=n*-1;
        end
        e(2)=(R(1,3)-R(3,1))/(n*4);
        e(3)=(R(2,1)-R(1,2))/(n*4);
    elseif index == 2
        e(2)=sqrt(1+2*R(2, 2)-T)/2;
        n=(R(1,3)-R(3,1))/(e(2)*4);
```

```
        if n < 0
            e(2)=e(2)*-1;
            n=n*-1;
        end
        e(1)=(R(3,2)-R(2,3))/(n*4);
        e(3)=(R(2,1)-R(1,2))/(n*4);
    else
        e(3)=sqrt(1+2*R(3, 3)-T)/2;
        n=(R(2,1)-R(1,2))/(e(3)*4);
        if n < 0
            e(3)=e(3)*-1;
            n=n*-1;
        end
        e(1)=(R(3,2)-R(2,3))/(n*4);
        e(2)=(R(1,3)-R(3,1))/(n*4);
    end
end
```

This is an implementation of Shepperd's algorithm as it is described in pseudo code in [4]. It takes as an argument a rotation matrix and gives out the Euler parameters. What makes this a good algorithm for the job i s that it is guarantied not to have any division by zero.

Shepperd's algorithm is not the fastest algorithm to find the Euler parameters. But as speed is not of great importance in this instance, this is not a problem.

B.1.4 Euler parameters to angle axis

```
function [fi, k] = euParToAngAx(n, e)
    fi=acos(n)*2;
    si=sin(fi/2);
    if si == 0
        k = [0 0 0];
    else
        k=e/si;
    end
end
```

B.1.5 Function used in the Model to VR block

The angletoangleaxis function used in the Model to VR block is basically the same function as the atoaa function described above but differs in that it takes the angle in the form of degrees not as above represented by radians.

Appendix C

The DVD

Included on the DVD is the videos and photos rendered from both the VRML file and Inventor. I have also included the VRML file along with the Simulink diagrams and the MATLAB code this is runnable if this is desirable. The Inventor files are also included. This can be viewed with a viewer downloadable from `http://usa.autodesk.com/adsk/servlet/pc/index?id=10535296&siteID=123112` if you don't have Inventor available. This is included as additional material that can be viewed if the reader finds this interesting. The most interesting file is WindTurbine.iam as this contains most of the models made. The images should however show the most interesting parts.

Appendix D

Drawing example

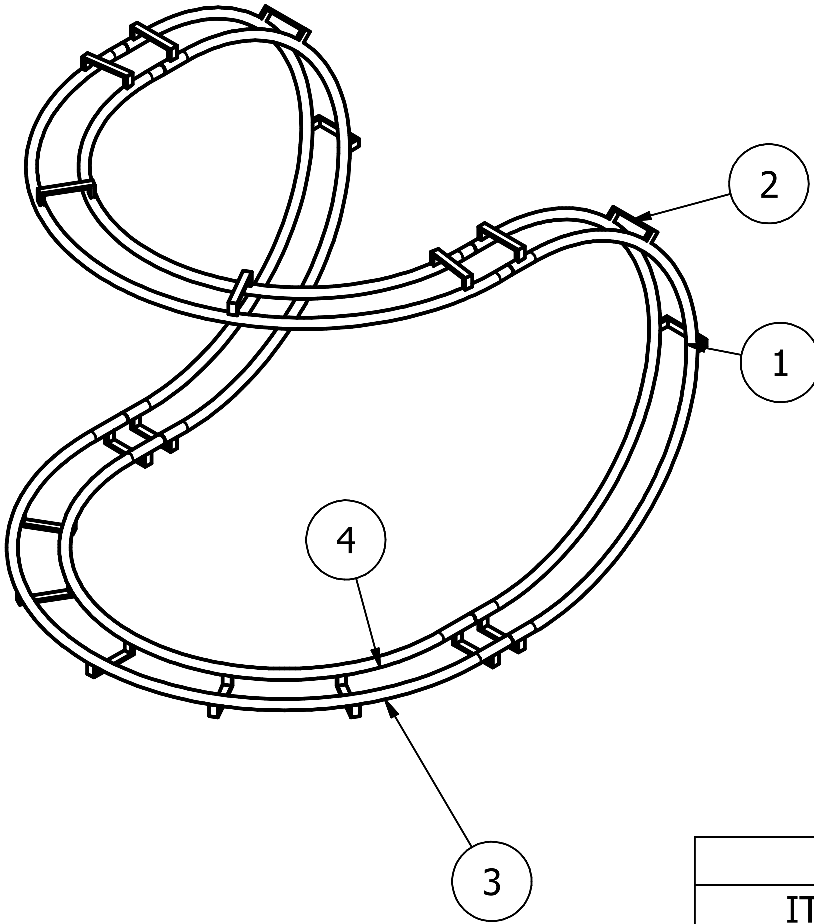
This is not meant as a complete documentation of the rails as thy have been built. It is only to show some of the capabilities of Inventor.

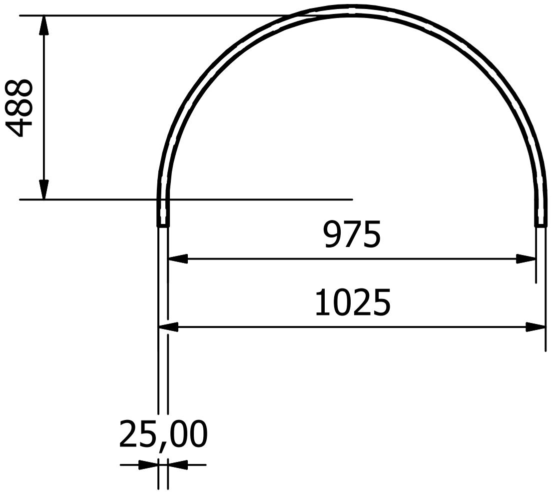
I made a 3D model of the rails as a preparation to design a cart for it that would accommodate an arm. However this option did not seem to be appropriate at the moment, so I abandoned this.

Here it is used to show how easily documentation can be made. To generate this drawings the model is simply projected with the appropriate view onto the paper. Then Inventor automatically generate a BOM list for it from the parts you have created. You can then annotate such tings as dimensions, weldments and threading for screws and so forth.

Some examples of this are shown in the drawing. If I wanted to completely document this I would have to show the dimensions of every part and also show where screw holes should be placed and the threading if these.

When plotted on a proper printer or plater this drawing will retain the proper dimensions so that you can use a ruler to measure different parts. You can then calculate the real dimensions from the scaling factor used in the drawing.





PARTS LIST

ITEM	QTY	PART NUMBER	DESCRIPTION
1	4	vertical rail	
2	19	cros beam	
3	2	outer rail	
4	2	iner rail	

Designed by	Checked by	Approved by	Date	Date
Eivind				13.12.2011

OldyRail	Edition	Sheet
		1 / 1

PRODUCED BY AN AUTODESK EDUCATIONAL PRODUCT