Guide wire Catheterization: Simulation and Actuation

Lior Kabesa
Guide wire Catheterization: Simulation and Actuation

Research Thesis

In partial fulfilment of the requirements for the degree of
Master of Science in Mechanical Engineering

Lior Kabesa

Submitted to the Senate of
the Technion - Israel Institute of Technology

Cheshvan 5773, Haifa, September 2012
The research thesis was done under the supervision of:

- Prof. Shoham Moshe, in the department of Mechanical Engineering, Technion.

- Dr. Shvalb Nir, in the department of Mechanical Engineering, Ariel University
Table of contents

1  Introduction.................................................................................................................. 3
   1.1  Medical background................................................................................................. 3
   1.2  Guide Wire Models................................................................................................. 4
2  Physical Model and simulation ..................................................................................... 9
   2.1  Basic Model ............................................................................................................ 9
   2.2  Geometrical interpretation ................................................................................... 20
   2.3  Guide wire algorithm I ......................................................................................... 21
   2.4  Local forces model ............................................................................................... 22
       2.4.1  Forces matrix ................................................................................................. 22
       2.4.2  Translation .................................................................................................... 26
       2.4.3  Forces calculation .......................................................................................... 28
       2.4.4  Numerical Relaxation .................................................................................. 30
   2.5  Local Forces Algorithm ......................................................................................... 32
3  Overall System description ........................................................................................ 33
   3.1  Mechanical system – GW manipulator ................................................................. 34
   3.2  Sensor system – GW and space identification ....................................................... 37
   3.3  Process simulation – GW deformation .................................................................... 39
4  Experimentation............................................................................................................. 40
   4.1  Experiment phantom ............................................................................................... 41
   4.2  Potential field algorithm ....................................................................................... 42
   4.3  FBS Algorithm ..................................................................................................... 47
5  Results .......................................................................................................................... 49
   5.1  First Target Navigation ........................................................................................ 50
   5.2  Second Target Navigation ..................................................................................... 54
   5.3  Third Target Navigation ....................................................................................... 58
   5.4  Results Summery .................................................................................................. 62
       5.4.1  Element number ............................................................................................ 62
       5.4.2  Bending ability ............................................................................................... 63
       5.4.3  Accuracy ......................................................................................................... 63
   5.5  Conclusions .......................................................................................................... 64
6  References.................................................................................................................... 67
List of figures

Figure 1: Vascular complications ...........................................................................4
Figure 2: Different GW simulation method..............................................................5
Figure 3: Basic model of the segments after translation .........................................10
Figure 4: Segment’s elasticity model .....................................................................11
Figure 5: $G_i$ wall potential field influence on segment $i$ ........................................13
Figure 6: $\alpha_i$ transformation in 3D representation ...............................................13
Figure 7: Segment geometrical constrains .............................................................18
Figure 8: GW model needs fine adjustment in high radius curvature ....................22
Figure 9: Basic beam deformation formulas ..........................................................23
Figure 10: External forces and internal moments ..................................................24
Figure 11: GW constant translation (left) versus sliding translation (right) ..........26
Figure 12: Possible contact points with lungs .......................................................29
Figure 13: Force modulation vectors in few insertions positions ............................30
Figure 14: Relaxation and sliding moments .........................................................31
Figure 15: The process Flow chart .......................................................................33
Figure 16: The process block diagram ..................................................................34
Figure 17: The manipulator system design ............................................................35
Figure 18: The manipulator system manufactured prototype ..............................36
Figure 19: Stepper Motors control and power board ...........................................36
Figure 20: The Video Motion detection algorithm ...............................................37
Figure 21: Three steps for GW navigation using pattern recognition ....................39
Figure 22: Experiment setup ................................................................................40
Figure 23: Lungs 3D phantom with dimensions .....................................................42
Figure 24: Lungs branches segmentation by levels ...............................................43
Figure 25: Potential possible routes in lungs phantom .........................................44
Figure 26: Overall potential field function in zero height leveling .......................44
Figure 27: Segment’s Vector definitions ...............................................................45
Figure 28: Possible potential vector calculation for three different points ..........46
Figure 29: 2D vector map for potential field .........................................................47
Figure 30: Potential map refinement .....................................................................47
Figure 31: First target simulation .........................................................................51
Figure 32: First simulation Position Errors ...........................................................52
Figure 33: Total segments RMS Error [mm] ..........................................................52
List of figures

Figure 34: First experiment DSC (%) coefficient for two simulations ..................... 53
Figure 35: Second target simulation ...................................................................... 55
Figure 36: Second simulation Position Errors ......................................................... 56
Figure 37: Total segments RMS Error [mm] ............................................................ 56
Figure 38: Second experiment DSC (%) coefficient for two simulations ............... 57
Figure 39: Third target simulation ......................................................................... 59
Figure 40: Third simulation Position Errors ............................................................ 60
Figure 41: Total segments RMS Error [mm] ............................................................ 60
Figure 42: Third experiment DSC (%) coefficient for two simulations ................. 61
List of tables

Table 1: Navigation algorithm ................................................................. 21
Table 2: Navigation algorithm with improvements .................................. 32
Table 3: GW identification algorithms .................................................... 38
Table 4: FBS algorithm for route selection ............................................ 48
Table 5: First Experiment results of simulations VS real GW. .............. 50
Table 6: Second Experiment results of simulations VS real GW. .......... 54
Table 7: Third Experiment results of simulations VS real GW. .............. 58
Table 8: Summary - Naïve algorithm VS Force-loci ............................. 62
ABSTRACT

Minimally invasive techniques such as guide wire insertion for catheterization and stent insertion have become important cardiac procedures. Nevertheless, there are risks of causing harm by forceful use of the guide wire and catheter during these procedures. We introduce a novel catheterization technique using an external manipulator which controls the guide wire using fluoroscopic imaging. In order to enable accurate and continuous navigation of the guide wire, we examine all the relevant factors for the system's model, e.g., boundary conditions and mechanical properties of the wire and the tissue.

This thesis deals with establishing a physical model for the guide wire geometry and with devising an overall system for guide wire navigation. We model the guide wire using concatenated rigid links attached to each other via a torsion spring. The external boundaries determine the deformation of the guide wire inside the artery assuming small quasi-static movements towards the target. For that matter we incorporate external normal forces acting on the guide wire during the collision with the body tissue.

Organization: In chapter 1 we review a short medical background which clarifies the need for this thesis, in addition, we survey some common techniques for computing the guide wire deformation. Chapter 2 is dedicated to the physical model. We first review rigid links energy model [1] [2] which is the basis of this work, having to know the model limitations we introduce our contribution, which allows an enhanced dexterity of the guide-wire. In order to accomplish this we analyze the forces exerted on the tissue wall, this in turn may be used to reduce the vascular complications risks while manipulating the guide-wire. In order to enable a large set of controlled experiments we fabricated a robot that can manipulate the guide-wire from its base point we shall introduce the robot design in Chapter 3. Experiments methods and results are described in Chapters 4 and 5.
List of symbols and abbreviations

GW - Guide Wire. We shall refer to the guide-wire as a catheter as well.

FL – Force-Loci

DOF - Degrees Of Freedom

FEM - Finite Element Method

RT - Real Time

VR - Virtual Reality

3D - Three dimensional

CCD - Charge Couple Device

ODE – Ordinary differential equation

PDE – Partial differential equation

ODE – Ordinary differential equation

PDE – Partial differential equation

RMS – Root mean square

STD – Standard deviation
1 Introduction

1.1 Medical background

Cardiac catheterization is a procedure involving the insertion of a catheter to the coronary vessels or into the heart (atrium/chamber) carried out for both diagnostic and interventional purposes. The procedure involves inserting a guide wire under fluoroscopic guidance through the femoral or radial arteries located in the pelvic area and the wrist respectively. The catheter is then inserted over the guide wire and threaded towards the heart. A group of procedures can be performed using cardiac catheterization such as: angioplasty (a technique of mechanically widening an occluded vessel involving a balloon catheter and/or a stent graft), angiography (a medical imaging technique used to visualize arteries, veins and the heart atrium and chambers), electrophysiology study (a procedure in which an electrode catheter is inserted in order to study the cardiac conduction system) and intracoronary ultrasound.

Studies concluded that the most common complications are vascular. One of the studies [3], which included 59,792 patients, showed 0.43% risk of vascular damage. Another study of 13,878 patients showed 5.37% of risks. The main risks include retroperitoneal bleeding, hematoma formation and arteries wall damage (See Figure 1). Other risks such as pseudoaneurysm, arteriovenous fistula formation, infection could occur but in much lower percentages. These conditions are partially due to manually pushing the catheter to the desired location causing arterial wall damage. This damage is a function of the friction coefficient between the GW and the body tissue and also related to the normal forces from the interaction between the vessel and the GW. In addition, there are difficult operating circumstances (such as advancing the catheter or the guide wire through tortuous or stenotic vessels and the crossing of the stenotic or the calcified aortic valve) which require alternative techniques in order not to cause damage. Some approaches include using various guide wires and catheters that are easier to track more bendable and with new low friction materials while other approaches target modifying the surgical technique itself or providing a training environment for the surgeon. In some countries, for
special catheterization process such as brain catheterizations it is required to pass training in VR environment which simulate the process prior to the treatment.

![Image of vascular complications]

**Figure 1: Vascular complications**

In this thesis we introduce a first step for an automated catheterization technique using an external manipulator. This manipulator controls the guide wire navigation (by translation and rotation) and will use fluoroscopic imaging for sensing and updating the simulation of the GW in RT. By examining the forces (see chapter 2 and 3) which are required for a continuous and smooth navigation of the guide wire, one may be able to minimize arterial wall damage and reduce the release of emboli as a result of that damage. Knowing the result of the GW navigation and estimating the pressure which is applied on the tissue together with an automated system could reduce the vascular complications.

Obviously, the arteries diameter varies with type and location along their axial direction therefore estimating the expected normal forces (given the geometry) is essential for the automated catheterization technique and is the main part of this research. We believe that simulating the GW deformation prior to insertion will enable navigation into narrower areas and improve the GW positioning inside the body.

1.2 **Guide Wire Models**

There are many techniques which may be used to model a guide-wire deformation We shall now shortly review the relevant factors of the main methods and pay attention to accuracy and calculation time.

Literature lists numerous techniques for representing the spatial wire. These can be categorized into three main special representations for the wire (Figure 2): (1) separated mass point with an additional spline or springs traversing them; (2) a
discrete coordinate frames along the wire; (3) concatenated rigid links. The wire elasticity on the other hand may be formulated for each of the above using a metric tensor, Cauchy tensor (see [13-14]) or simple torsion springs [16]. Alternatively one may use minimum energy concepts (least action method for the static case).

![Figure 2: Different GW simulation method. A – Rigid links, B – Mass Spring, C – Dynamic Spline, D – FEM, E – Cosserat.](image)

**Multi-Rigid body link** (Lubos et al. [4]) (Figure 2 A) connected by spherical joints was one of the first models which deal with GW deformation. In this method, which came originally from the robotics field (usually more relevant model for endoscope simulation), external forces are applied on the links at collision areas and additional forces and torques created from the user manipulation at insertion point. Inner forces from the wire elasticity are considered as well (calculated as springs). However, this method is missing a précised representation of internal forces between adjacent links. In this approach the spring coefficients were determined by comparing the virtual catheter simulations with real catheters. The virtual catheter was inserted and manipulated as the real catheter was, and then images from the real and the virtual insertion were overlaid. When the images of the catheters differed, the spring coefficients were adjusted so that the virtual catheter matched the real one as best as possible. For simulation optimization two measurements were used: the
maximum distance between the centrelines of the two catheters and the Dice Similarity Coefficient (DSC). The result is an unrealistic behaviour of the GW in this simulation since its complex elasticity which might not be linear in high curvature radii. However, this method could be used for endoscope simulation because its links structure and the elasticity properties. Later on Mass spring model was introduced by Wang et al. [12] (Figure 2 b), improving the method above. Wang modelled the GW mass attached with group of (mass less) linear and angular springs attached to each link produce stretching, compressing, bending and twisting between and on the nodes. This way, the dynamic behaviour of the GW is modelled but still it lacks an accurate static physical behaviour of elastic thin rods since it is using only linear springs. A thin rod such as GW has a variable resistance to bending (non-linear) therefore for our usage of static GW deformation, this model is not preferred. In our research we pay attention mostly to the accurate shape which need high deformation ability that is essential for a realistic modelling of the wire which will give the most précised normal forces. We may assume that this simulation is made during a quasi-static threading process, and neglect the velocity and acceleration of each node.

A different method was developed to achieve better quasi-static behaviour by using A Finite element method introduced by Cotin et al. [4],[5] (Figure 2 d). This method is a physics-based model which consists of a set of connected beam elements while each segment has bending twisting abilities in six DOF. To maintain the validity of linear elasticity theory, we assume the catheter temporally moves with small displacements. In this method, need to calculate 12x12 matrixes of the forces and moments which apply on the base and end of each node. The first and main drawback from this method is related to the complex matrix method which requires also transform the matrix of each segment into global coordinate system in each iteration, it adds calculation time and makes this method to be very sensitive to cumulative errors from the base to the tip. In addition, since it leans on the elasticity theory of beam while thin rods are not incorporate the classic theory, the local errors generated during the calculation are translated incrementally through the nodes and create a relative accurate deformation while applications such as GW is needs for higher bending abilities. Another drawback from this model is that it could return
only to its previous state and not the initial state of the GW. There are some solutions that could be done to overcome this issue but still computation time is high and therefore this model is not preferred. The main high-light of this method is that lower number of elements is needed to represent the GW, still there are issues with contact handling representation and numerical solutions are needed to deal with high order matrix. **Dynamic spline model** (see for example [06] and [16]) (Figure 2 c) a continuous physics-based model, based on Lagrange equations combined with spline geometry (Catmull-Rom method which pass through points [06]). This model objects are continuous spatial curves defined as successions of spline segments collectively controlled by a set of n mobile control points. This model could deal with high curvatures but since it not use continuous energies and its spring representation method is based on the wire special position, is it lacks the right formulation of bending and twisting energies which are needed for representing accurately the deformation of the GW. In addition, since the contact points with tissue changes, this model has difficulties to present smooth and realistic GW movements.

**Cosserat theory (Strands)** [08], [13] (Figure 2 e) model is a general beam deformation theory for beams that could used to formulate static or dynamic deformation behaviour. This GW simulation method was first introduced by Pai and later by Wang [11], Cosserat was used only for static model. This model takes into account all possible deformations (6 DOF) of a one dimensional object, however contact handling with such a model is difficult: in order to have ODE rather the PDE one simplifies the model by considering only a relative coordinate frames (i.e. one computes a frame relative to its prior). Still this yields a boundary value ODE, and having such multiple conditions makes the problem hard to solve. For catheter navigation, where collisions occur continuously along the length of the device, this is a critical issue. Computation times for a large number of points can lead to non-interactive simulation times. It should be noted that Cosserat theory is best suited if torsional motion is essential (here it is assumed that torsional movements are small) and also could be used to simulate the GW tip which has usually different properties. Another application for Cosserat is to describe ropes and materials with low bending
resistance and also plastic deformation behaviour of wires. Cosserat could be used also to model tissues sewing and to find materials and techniques for this cause. Since our we are dealing with GW and the elastic behaviour is one of the keys for accurate modelling we have preferred to use a different model which incorporate the high bending curvatures and keeps the thin rods elastic behaviour.

A relative accurate model was introduced by Koning and Alderliesten [01], [02] and it is using Multi-link model together Minimum energy principle and forms a semi-analytical theory which solves the minimum energy problem with relation to the internal and external forces. This multi-body system is composed by rigid bodies, joints, angular springs and which needs to keep minimum energy in this close system. This model can be seen as a macroscopic approximation of a continuous model at some level. External forces, on the GW tip and collision areas, are transmitted to all rigid bodies via the links. Internal elastic forces are represented by angular springs placed between consecutive rigid links. This discrete model for the catheter permits a good approximation of a catheter but requires many small links to represent a high degree of flexibility, thus leading to increased computation cost which is least significant to us.

Some of the recent researches have based on a combination of the method above and developed new combined methods such as (Shun Li and Jing Qin [09]) which used minimum energy concept introduced by Koning and Alderliesten but with a changed the energy formulation. The energy terms has a discrete form for each node creating of energy equations as functions of the vectors which connecting them, Solving those equations using FEM method gives the special wire representation. Another research of Wang et al [10] have used a new representation of mass point model which is based on finding forces by using Jacobian matrices of the force with respect to position and velocities. In addition they have added a friction model for static and dynamic friction and also added a feature of re-meshing the GW in cases of obstacles such as blood clot the GW. In this area, researchers are still searching for ways to improvement the GW performances and get more realistic behaviour for the GW insertion and the interaction with body tissue. Those researches are mainly to improve catheterization trainer applications in VR environment. Some of the new methods are combining
few physical based simulations such as Cosserat, Minimum Energy and FEM method to achieve better RT performances also in the area of force feedback.

We shall also use the method of minimum energy as a benchmark for this study and introduce novel algorithm which take the physical based simulation and add new method for forces contact handling and calculation of collision with the surrounding tissues. These enabled a stable GW movement and improved accuracy while substantially reducing the number of links needed for computation and enhancing the maximum curvature constraint. The calculation time was not considered here as important factor. Therefore we will be analyzing the results by comparing accuracies of the models and real GW threading.

2 Physical Model and simulation

In the following chapter we shall describe the physical model and algorithm for computing the guide-wire’s shape. In sections 2.1, 2.2 and 2.3 we shall review the Multi-link Minimum Energy GW model and follow [1] [2] for semi-analytical solution of minimal invasive endovascular interventions. In section 2.4 we shall introduce our contribution with an additional Local Forces formalism since this model fails to overcome small radius of curvatures and also get unstable in multiple forces modes.

2.1 Basic Model

The model is based on minimizing the total energy of the GW threading process. The GW is deformed during the threading while it collides with a confined tissue (taken here to be the respiratory system or the blood vessels). We assume threading in quasi-static steps therefore the dynamic nature of the movements is not modeled. We will take into account are only Potential Energies: (1) from the GW’s elastic deformation (Internal energy) and (2) of the tissue’s elastic deformation during the collision (External energy).

\[ U_i^{tot} = U_i^{Wall} + U_i^{Bend} = Const \]
We have defined vectors $\mathbf{\lambda}_i$ between each node (which has the same length $\lambda$). After a forced translation of $\mathbf{\varepsilon}_0$, each segment $\mathbf{\lambda}_i$ will move to a new position by adding $\mathbf{\alpha}_i$, see Figure 3.

![Figure 3: Basic model of the segments after translation](image)

The vector $\mathbf{\alpha}_i$ represents a relative change between the movements of two sequential links. We assume that the movement of a segment $\mathbf{\lambda}_i$ will effect the following consecutive segments (and by doing so it “share” its potential energy). The translation $\delta\mathbf{\lambda}_i$ at each joint $i$ can be derived from the following formula, which accumulates each segment's influence on the total deformation and the initial quasi-static displacement:

$$
\delta\mathbf{\lambda}_i = \mathbf{\varepsilon}_0 + \sum_{j=0}^{i-1} \mathbf{\alpha}_j
$$

One need to find a set of displacements vectors $\mathbf{\alpha}_i$ for each translation $\mathbf{\varepsilon}_0$ in order to solve this problem. To do so, we will use the energy conservation law because the total energy of the system remains unchanged.

**GW Elasticity model** is dividing the wire into discrete concatenated rigid link elements with Torsional spring attaching them. Each spring follow Hook’s Law and the elastic angular deformation $\theta_i$ is proportional to the applied torque $M_i$.

$$
M_i = -c_i \theta_i
$$
Integrating the torque equation by $\theta$ will give an expression of discrete potential energy as a result of GW bending.

$$U_{\text{bend},i} = -\int_{0}^{\theta} M_i d\tilde{\theta} = \frac{1}{2} c_i \theta_i^2$$ \hspace{1cm} 2.3

Total discrete potential energy in segment $i$ is a sum of all energies from the base to the segment

$$U_{\text{bend},i}^{\text{Tot}} = \sum_{k=1}^{i} U_{\text{bend},k} = \frac{1}{2} c_i \sum_{k=1}^{i} \theta_k^2$$

On the other hand, this energy could be also formulated as continues energy of a beam with flexural rigidity $EI$ from the base to segment $\lambda_i$:

$$U_{\text{Continues}} = \int_{0}^{\lambda_i} \frac{1}{2} EI (\psi')^2 dx$$

We will integrate this expression and use the relation $\psi' = 1/R_i$ (curvature radius) finally having an expression for Continues interval Energy:

$$U_{\text{Continues}} = \frac{\lambda_i EI}{2R_i^2}$$ \hspace{1cm} 2.4

Using an assumption of small angles $\theta << 1$ will define $\theta_i \approx \lambda_i / R_i$ and inserting this relation to Eq. 2.4 we will get:

$$U_{\text{ContinuesInterval}} = \frac{\theta_i^2 EI}{2\lambda_i}$$ \hspace{1cm} 2.5

Comparing the Bending Energy of a segment from Eq. 2.3 with Continues Interval Energy Eq. 2.5 will give us an expression of Segment’s Torsional spring coefficient:
\[ c_i = \frac{EI}{\lambda} \]  

To have an expression of GW potential energy, we will representation \( \theta_i^2 \) of Eq. 2.3 using trigonometric identities (see Figure 4):

\[
\begin{align*}
\cos(\theta) &= \hat{\lambda}_{i-1} \cdot \hat{\lambda}_i \\
\hat{\lambda}_{i-1} &= \frac{1}{\lambda}(x_i - x_{i-1}) \quad ; \quad \hat{\lambda}_i = \frac{1}{\lambda}(x_{i+1} - x_i) \\
\cos(\theta) &= \frac{1}{\lambda}(x_i - x_{i-1}) \cdot \frac{1}{\lambda}(x_{i+1} - x_i) \\
\sin^2(\theta) &= 1 - \cos^2(\theta) = 1 - \frac{1}{\lambda^2}[(x_i - x_{i-1}) \cdot (x_{i+1} - x_i)]^2
\end{align*}
\]

Note that if \( \theta \) is small we may substitute it into the equation above:

\[
\theta_i^2 = 1 - \frac{1}{\lambda^2}[(x_i - x_{i-1}) \cdot (x_{i+1} - x_i)]^2
\]

Using this expression of \( \theta_i^2 \) into the bending potential energy will give a discrete expression for bending energy based on 3 points:

\[
U_{i,\text{Bend}} = \frac{C}{2} \left\{ 1 - \frac{1}{\lambda^2}[(x_i - x_{i-1}) \cdot (x_{i+1} - x_i)]^2 \right\}
\]

2.7

The arterial wall’s elasticity is modelled as linear springs and follows Hook’s Law therefore the force \( \vec{F} \) is proportional, with a spring coefficient \( g \), to the wall displacement \( \Delta \vec{x} \) (measured from rest position).

\[ \vec{F} = g \Delta \vec{x} \]

We shall integrating this equation to get the segment’s potential energy:

\[
U_{w_i} = \frac{1}{2} g \Delta \vec{x}_i^2
\]

2.8

Finally summing it all up from the segment till the tip:

\[
U_{w,\text{Total}} = \sum_{i=1}^{N} \frac{1}{2} g \cdot \Delta \vec{x}_i^2
\]

Now we will define the vector \( \vec{G}_i \) which is the sum of all wall potential energy gradients from segment \( i \) to the GW tip:

\[
\vec{G}_i = \sum_{j=i+1}^{N} \nabla U_{i,\text{Wall}}^j
\]

2.9
We will use a 3D transformation to represent $\alpha_i$, the movement vector, using only two variables $a_i, \psi_i$ (see Figure 6). This transformation creates a spherical representation

$$\bar{\alpha}_i = \frac{a_i^2}{2\lambda} \hat{e}_i^{(p)} + \frac{a_i}{2\lambda} \sqrt{4\lambda^2 - a_i^2} \cos \psi_i \hat{e}_i^{(q)} + \frac{a_i}{2\lambda} \sqrt{4\lambda^2 - a_i^2} \sin \psi_i \hat{e}_i^{(s)}$$

Where

$\alpha_i$ - Total Movement vector of segment $i$

$a_i$ - Translation vector of segment $i$

$\psi_i$ - Rotation vector of segment $i$

We will differentiate the energy equation and receive a partial differential equation:
\[
\delta U_i^{\text{tot}} = \delta \vec{a}_i \cdot F_i^{\text{tot}} = \left( \frac{\delta \vec{a}_i}{\partial a_i} \delta a_i + \frac{\delta \vec{a}_i}{\partial \psi_i} \delta \psi_i \right) \cdot F_i^{\text{tot}} = 0
\]

2.11

This implies that for each segment the following holds:

\[
\frac{\partial U_{\text{tot}}}{\partial a_i} = \frac{\delta \vec{a}_i}{\partial a_i} F_i^{\text{tot}} = 0
\]

2.12

\[
\frac{\partial U_{\text{tot}}}{\partial \psi_i} = \frac{\delta \vec{a}_i}{\partial \psi_i} F_i^{\text{tot}} = 0
\]

2.13

We will take into consider three forces, external force created from the wall potential energy gradient and internal forces, form each side of the link, determined by the GW elasticity energy.

\[
F_i^{\text{tot}} = \vec{G}_i + \vec{F}_i^{\text{Bend}} + \vec{F}_{i+1}^{\text{Bend}}
\]

Therefore we will separate the expression of energy:

\[
\delta U_i^{\text{tot}} = \delta \vec{a}_i \left( \vec{G}_i + \vec{F}_i^{\text{Bend}} + \vec{F}_{i+1}^{\text{Bend}} \right) = 0
\]

\[
\delta U_i^{\text{Wall}} + \delta U_i^{\text{Bend}} + \delta U_{i+1}^{\text{Bend}} = \delta \vec{a}_i \cdot \vec{G}_i + \delta \vec{a}_i \cdot \vec{F}_i^{\text{Bend}} + \delta \vec{a}_i \cdot \vec{F}_{i+1}^{\text{Bend}}
\]

2.14

First we will deal with \( \delta U_i^{\text{Wall}} \):

\[
\delta U_i^{\text{Wall}} = \delta \vec{a}_i \sum_{j=i+1}^{\text{link}} \nabla U_{\text{wall}}(x_j) = \delta \vec{a}_i \cdot \vec{G}_i
\]

While \( \vec{G}_i \) is the accumulation of all the gradient vectors created form the wall form the link \( i \) to the tip and create the vector. This calculation will be done using the potential energy algorithm which will be described later, in section 4.2.

Now, we will formulate an expression for \( \delta U_i^{\text{Bend}} \), which is influence by the forces \( \vec{F}_i^{\text{Bend}} \), \( \vec{F}_{i+1}^{\text{Bend}} \) Inserting the GW with quasi-static movement \( \vec{e}_0 \), will cause a deformation of the GW during collision with the walls. Since the GW has an elastic behavior, each link will find to the next minimum energy position. We will add \( \delta \vec{x}_i \), which is defined in Eq.2.2, to each segment position. So translating all segments with \( \vec{e}_0 \) makes the product expression which defines the angle between segments \((\vec{x}_i - \vec{x}_{i-1}) \cdot (\vec{x}_{i+1} - \vec{x}_i)\) in Eq. 2.7 become \((\vec{x}_i + \delta \vec{x}_i - \vec{x}_{i-1} - \delta \vec{x}_{i-1}) \cdot (\vec{x}_{i+1} + \delta \vec{x}_{i+1} - \vec{x}_i - \delta \vec{x}_i)\)
recalling Eq.2.2.2 one may write \( \delta \bar{x}_i - \delta \bar{x}_{i-1} = \bar{\alpha}_{i-1} \), \( \delta \bar{x}_{i+1} - \delta \bar{x}_i = \bar{\alpha}_i \) and using for the definition of \( \bar{\lambda}_i \), one gets:

\[
(\bar{x}_i - \bar{x}_{i-1} + \bar{\alpha}_{i-1}) (\bar{x}_{i+1} - \bar{x}_i + \bar{\alpha}_i) = (\bar{x}_{i-1} + \bar{\alpha}_{i-1}) (\bar{x}_i + \bar{\alpha}_i) = \bar{\lambda}_{i-1} \bar{x}_i + \bar{\lambda}_i \bar{x}_{i+1} + \bar{\lambda}_{i+1} \bar{x}_i + \bar{\lambda}_i \bar{\alpha}_i
\]

Substituting (Eq. 2.15) into the bending energy expression (Eq. 2.7) yields:

\[
U_{i\text{Bend}} = C \left\{ 1 - \frac{1}{\lambda_i^2} \left[ \bar{\lambda}_{i-1} \bar{x}_i + \bar{\lambda}_i \bar{x}_{i+1} + \bar{\lambda}_{i+1} \bar{x}_i + \bar{\lambda}_i \bar{\alpha}_i \right] \right\}
\]

\[
\delta U_{i\text{Bend}} = \delta \left[ \frac{C}{2} - \frac{C}{2\lambda_i^4} (\bar{\lambda}_{i+1} \bar{x}_i + \bar{\lambda}_i \bar{x}_{i+1} + \bar{\lambda}_{i-1} \bar{x}_i + \bar{\lambda}_i \bar{\alpha}_i)^2 \right]
\]

\[
\delta U_{i\text{Bend}} = \delta \left[ \frac{C}{2} - \frac{C}{2\lambda_i^4} (\bar{\lambda}_{i+1} \bar{x}_i + \bar{\lambda}_i \bar{x}_{i+1} + \bar{\lambda}_{i+1} \bar{x}_i + \bar{\lambda}_i \bar{\alpha}_i)^2 \right]
\]

Differentiating the equation for bending energy results with respect to \( \delta \alpha_i \):

\[
\delta U_{i\text{Bend}} = -\frac{C}{\lambda_i^4} \left( \bar{\lambda}_{i+1} \bar{x}_i + \bar{\lambda}_i \bar{x}_{i+1} + \bar{\lambda}_{i-1} \bar{x}_i + \bar{\lambda}_i \bar{\alpha}_i \right) \left( \bar{\lambda}_{i-1} + \bar{\alpha}_i \right) \delta \alpha_i
\]

\[
\delta U_{i\text{Bend}} = -\frac{C}{\lambda_i^4} \left( \bar{\lambda}_{i+1} \bar{x}_i + \bar{\lambda}_i \bar{x}_{i+1} + \bar{\lambda}_{i+1} \bar{x}_i + \bar{\lambda}_i \bar{\alpha}_i \right) \left( \bar{\lambda}_{i+1} + \bar{\alpha}_i \right) \delta \alpha_i
\]

Expanding the differential expression and neglecting high order terms one have:

\[
\delta U_{i\text{Bend}} = -\frac{C}{\lambda_i^4} \left( \bar{\lambda}_{i+1} \bar{x}_i^2 + \bar{\lambda}_i \bar{x}_{i+1}^2 + 2\bar{\alpha}_i \bar{\lambda}_{i+1} \bar{x}_i + \bar{\lambda}_{i+1} \bar{x}_i^2 + \bar{\lambda}_{i+1} \bar{x}_i + \bar{\lambda}_i \bar{x}_{i+1}^2 \right) \delta \alpha_i
\]

Since \( \alpha_i \) is perpendicular to \( \bar{\lambda}_i \) and \( |\alpha_i| \ll |\bar{\lambda}_i| \) the expression reduces to:

\[
\delta U_{i\text{Bend}} = -\frac{C}{\lambda_i^4} \left( \bar{\lambda}_{i+1} \bar{x}_i^2 + \bar{\lambda}_i \bar{x}_{i+1}^2 \right) \delta \alpha_i \quad \text{And} \quad \delta U_{i\text{Bend}} = -\frac{C}{\lambda_i^4} \left( \bar{\lambda}_{i+1} \bar{x}_i^2 + \bar{\lambda}_i \bar{x}_{i+1}^2 \right) \delta \alpha_i \quad \text{so the total bending energy of } i\text{-th segment is approximately:}
\]

\[
\delta U_{i\text{Bend}} + \delta U_{i-1\text{Bend}} \approx -\frac{C}{\lambda_i^4} \left( \bar{\lambda}_{i+1} \bar{x}_i^2 + \bar{\lambda}_i \bar{x}_{i+1}^2 + \bar{\lambda}_{i+1} \bar{x}_i + \bar{\lambda}_i \bar{x}_{i+1}^2 \right) \delta \alpha_i
\]

Rearranging the terms yields:

\[
\frac{C}{\lambda_i^4} \left[ (\bar{\lambda}_{i+1} \bar{x}_i + (\bar{\lambda}_{i+1} - \bar{\lambda}_i)) + \bar{\alpha}_i (\bar{\lambda}_i + (\bar{\lambda}_{i+1} - \bar{\lambda}_i))^2 \right] + \left[ \bar{\lambda}_i (\bar{x}_i + (\bar{\lambda}_{i+1} - \bar{\lambda}_i)) + \bar{\alpha}_i (\bar{x}_i + (\bar{\lambda}_{i+1} - \bar{\lambda}_i))^2 \right] \delta \alpha_i
\]

Finally incorporating \( \lambda \) into the parenthesis yields:
\[ \delta U^{\text{Bend}}_i + \delta U^{\text{Bend}}_{i-1} \approx \frac{C}{\lambda^2} \left\{ \left[ \frac{\alpha_i}{\lambda} - \frac{\alpha_{i-1}}{\lambda} \right] + \frac{1}{\lambda} \left( \alpha_i + \alpha_{i-1} \right) \right\} \delta \alpha_i \]

Denoting the terms:

\[ \eta_i = \frac{1}{\lambda} (\alpha_i - \alpha_{i-1}) \quad \varepsilon_i = \frac{1}{\lambda} (\alpha_i + \alpha_{i-1}) \quad k_i = \alpha_i \cdot \alpha_{i-1} \quad k'_i = \alpha_i \cdot \alpha_{i-1} \quad \dot{\varepsilon}_i^{(p)} = \frac{\alpha_i}{\lambda} \]

Incorporating them in the bending energy expression (Eq. 2.16), we will get an expression of the bending force \( F_i^{\text{Bend}} \). Adding \( \dot{G}_i \) from Eq. 2.14, will give the next expression to the total force:

\[ \frac{\partial U^{\text{int}}}{\partial \alpha_i} = G_i - \frac{C}{\lambda^2} \left\{ \left[ k_i + \alpha_i \left( \dot{\varepsilon}_i^{(p)} + \dot{\varepsilon}_i \right) \right] + \left[ k'_i + \alpha_i \left( \dot{\varepsilon}_i^{(p)} + \dot{\varepsilon}_i \right) \right] \left( \dot{\varepsilon}_i^{(p)} + \dot{\varepsilon}_i \right) \right\} = F_i^{\text{Bend}} \]

Now, we will partially derive \( \partial \alpha_i \) (see Eq. 2.11) with regards to \( \partial \psi \) and \( \partial a \)

\[ \frac{\partial \alpha_i}{\partial \psi} \cdot \nabla = \left( \frac{\partial \alpha_i}{\partial \alpha} \cdot \frac{\partial \alpha_i}{\partial \psi} \right) \]

First we will use the coordinate system defined in Eq. 2.10 to present the vector \( \alpha \):

\[ \alpha = \left[ \begin{array}{c} \frac{a}{2\lambda} \\ H(a) \cdot \left( \begin{array}{c} \cos(\psi) \\ \sin(\psi) \end{array} \right) \end{array} \right] \]

One will derive \( \alpha \) by \( \psi \) and multiply it by the total force from \( 2.18 \). This yields the equation below which need to solve in order to satisfy the minimum energy equation.

\[ \frac{\partial \alpha}{\partial \psi} \cdot \nabla = \left[ \begin{array}{c} 0 \\ H(a) \cdot \left( \begin{array}{c} -\sin(\psi) \\ \cos(\psi) \end{array} \right) \end{array} \right] \cdot \nabla \left[ \begin{array}{c} \dot{\varepsilon}_i^{(p)} \\ \dot{\varepsilon}_i^{(s)} \\ \dot{\varepsilon}_i^{(0)} \end{array} \right] = 0 \]

We will combine the forces into the equation:

\[ \left[ \begin{array}{c} 0 \cdot \dot{\varepsilon}_i^{(p)} \\ H(a) \cdot \left( \begin{array}{c} -\sin(\psi) \cdot \dot{\varepsilon}_i^{(s)} \\ \cos(\psi) \cdot \dot{\varepsilon}_i^{(0)} \end{array} \right) \end{array} \right] \cdot \left[ \begin{array}{c} G_i - \frac{C}{\lambda^2} \dot{\varepsilon}_i^{(p)} \left[ \frac{k_i}{\lambda} + \frac{k'_i}{\lambda} + \alpha_i \left( 2\dot{\varepsilon}_i^{(p)} + \dot{\varepsilon}_i + \dot{\varepsilon}_i \right) \right] \\ - \frac{C}{\lambda^2} \left[ \dot{\varepsilon}_i \left( \frac{k_i}{\lambda} + \alpha_i^{(p)} \right) + \dot{\varepsilon}_i \left( \frac{k'_i}{\lambda} + \alpha_i^{(p)} \right) \right] \end{array} \right] = 0 \]

16
Since the movement of the segment in \( \psi \) direction will not apply any change in the vector \( e^{(P)} \) (which is parallel to segment) we can write the equation in a next form.

\[
H(a) = -\sin(\psi) \left[ \tilde{G}^{(s)}_i - \frac{C}{\lambda^2} \left( \varepsilon^{(s)}_i k_i - \eta^{(s)}_i k_i \right) \right] \\
\cos(\psi) \left[ \tilde{G}^{(q)}_i - \frac{C}{\lambda^2} \left( \varepsilon^{(q)}_i k_i + \eta^{(q)}_i k_i \right) \right] = 0
\]

Knowing that the movements of the joints in each iteration will be smaller than the segment size \( a^2 \ll \lambda^2 \) will get a simplify equation.

\[
H(a) = -\sin(\psi) \left[ \tilde{G}^{(s)}_i - \frac{C}{\lambda^2} \left( \varepsilon^{(s)}_i k_i + \eta^{(s)}_i k_i \right) \right] = 0
\]

Calculate the vector \( \bar{B}^{(SQ)}_i = \tilde{G}^{(SQ)}_i - \frac{C}{\lambda^2} \left( \varepsilon^{(SQ)}_i k_i - \eta^{(SQ)}_i k_i \right) \) by using the vectors defined in Eq.2.17 and finding the angle \( \psi \) with makes a perpendicular vector, solve the equation. \( \sin(\psi), \cos(\psi) \) here represent the whole 2D vector space, there are two vectors which will be perpendicular to the segment(\( \psi \) and \( \psi \pm \pi \)). Finding the angle \( \beta \), which is the angle of force applied on segment \( i \), projected on segment’s SQ is plane. Finally selecting \( \psi = \beta + \pi \) (this solution represents the minimum energy because it promote the GW from areas with high wall energy gradient), will satisfy the equation and give the minimal energy of segment \( i \).

Now we will derive the vector \( \bar{a} \) by \( a \) in order to find the last variable:

\[
\frac{\partial \bar{a}}{\partial a} = \left[ \begin{array}{c}
\frac{-a}{\lambda} \\
H(a) \cdot \left( \begin{array}{c}
\cos(\psi) \\
\sin(\psi)
\end{array} \right)
\end{array} \right]^T \left[ \begin{array}{c}
e^{(P)} \\
e^{(S)} \\
e^{(Q)}
\end{array} \right]
\]

Multiply it by the force as we did with the derivative of \( \psi \):
\[
\frac{\partial \alpha}{\partial a} \cdot \mathbf{F} = \left[ \begin{array}{c}
- \frac{a}{\lambda} \mathbf{e}^{(p)} \\
H'(a) \left\{ \cos(\psi) \mathbf{e}^{(s)} + \sin(\psi) \mathbf{e}^{(q)} \right\}
\end{array} \right] = 0
\]

\[
\begin{align*}
\hline
\hline
\hline
\end{align*}
\]

\[
\begin{align*}
\frac{\partial ^{2} \alpha}{\partial a^{2}} \cdot \mathbf{F} = \left[ \begin{array}{c}
- \frac{a}{\lambda^2} \mathbf{e}^{(p)} \\
H''(a) \left\{ \cos(\psi) \mathbf{e}^{(s)} + \sin(\psi) \mathbf{e}^{(q)} \right\}
\end{array} \right] = 0
\end{align*}
\]

\[
\begin{align*}
\hline
\hline
\hline
\end{align*}
\]

Figure 7: Segment geometrical constrains.

Using the geometrical constrains from Figure 7, \(x^2 + (\lambda - y) = \lambda^2\) and \(x^2 + y^2 = a^2\), we have found an expression for \(\alpha^{(p)} = -a^2 / 2\lambda\). Together with the assumption that \(\theta_i\) is small we know that \(a^2 / \lambda^2 < 1\), therefore \(\alpha_i^{(p)}\) will be much smaller than the expressions \(\alpha_i^{(p)} < k' / \lambda\) and \(\alpha_i^{(p)} < k / \lambda\).

Using the assumptions, and neglecting the high order terms, we can simplify the expression for \(\frac{\partial \alpha}{\partial a} \cdot \mathbf{F}\) and get the next equation:

\[
\begin{align*}
- \frac{a}{\lambda} \left\{ \mathbf{G}_i^{(p)} - \frac{C}{\lambda^2} \left[ k_i + k_i' + \alpha_i \left( 2\mathbf{e}^{(p)} + \mathbf{e} + \mathbf{n} \right) \right] \right\} + H'(a) \left\{ \cos(\psi) \mathbf{G}_i^{(s)} - \frac{C}{\lambda^2} \left[ \mathbf{e}^{(s)} k_i + \mathbf{n}^{(s)} k_i \right] \right\} + H''(a) \left\{ \sin(\psi) \mathbf{G}_i^{(q)} - \frac{C}{\lambda^2} \left[ \mathbf{e}^{(q)} k_i + \mathbf{n}^{(q)} k_i \right] \right\} = 0
\end{align*}
\]

This is:

\[
\begin{align*}
\frac{\partial ^{2} \alpha}{\partial a^{2}} \cdot \mathbf{F} = \left[ \begin{array}{c}
- \frac{a}{\lambda^2} \mathbf{e}^{(p)} \\
\cos(\psi) \mathbf{G}_i^{(s)} - \frac{C}{\lambda^2} \left[ \mathbf{e}^{(s)} k_i + \mathbf{n}^{(s)} k_i \right] \\
\sin(\psi) \mathbf{G}_i^{(q)} - \frac{C}{\lambda^2} \left[ \mathbf{e}^{(q)} k_i + \mathbf{n}^{(q)} k_i \right]
\end{array} \right] = 0
\end{align*}
\]

Using the vector \(\mathbf{b}^{(30)}\), we already found from Eq. Error! Reference source not found.2.19, and deriving the function \(H(a)\):
\[ H(a) = \left( \frac{a}{2\lambda} \right) \sqrt{4\lambda^2 - a^2} \]

\[ H'(a) = \frac{\partial H(a)}{\partial a} = \frac{1}{2\lambda} \sqrt{4\lambda^2 - a^2} - \frac{a^2}{2\lambda} \left( \frac{1}{\sqrt{4\lambda^2 - a^2}} \right) \begin{cases} a^2 \ll 1 \end{cases} \]

\[ H'(a) = \frac{\partial H(a)}{\partial a} = \frac{1}{2} \sqrt{4 - \frac{a^2}{\lambda^2}} - \frac{a^2}{2\lambda^2} \left( \frac{1}{\sqrt{4 - \frac{a^2}{\lambda^2}}} \right) \equiv 1 \]

We will get a simplified expression

\[
\begin{align*}
&\left[ \cos(\psi) \left\{ \bar{G}^{(s)}_i - \frac{C}{\lambda^2} \left( \bar{\varepsilon}^{(s)}_i + \bar{\eta}^{(s)}_i \right) \right\} \right] = -a_i \left\{ \bar{G}^{(p)}_i - \frac{C}{\lambda^2} \left[ \frac{k_i + k^\prime_i}{\lambda} + \bar{\varepsilon}_i \left( 2\bar{\varepsilon}^{(p)}_i + \bar{\eta}_i + \bar{\eta}^{(p)}_i \right) \right] \right\} \\
&\sin(\psi) \left\{ \bar{G}^{(q)}_i - \frac{C}{\lambda^2} \left( \bar{\varepsilon}^{(q)}_i k_i + \bar{\eta}^{(q)}_i k^\prime_i \right) \right\} = -a_i \left\{ \bar{G}^{(p)}_i - \frac{C}{\lambda^2} \left( \frac{k_i + k^\prime_i}{\lambda} + \bar{\varepsilon}_i \left( 2\bar{\varepsilon}^{(p)}_i + \bar{\eta}_i + \bar{\eta}^{(p)}_i \right) \right\} \\
&a_i = -\frac{\lambda}{\bar{G}^{(p)}_i - \frac{C}{\lambda^2} \left( \frac{k_i + k^\prime_i}{\lambda} + \bar{\varepsilon}_i \left( 2\bar{\varepsilon}^{(p)}_i + \bar{\eta}_i + \bar{\eta}^{(p)}_i \right) \right)} \\
&\bar{G}_i^{(q)} = \bar{G}^{(q)}_i - \frac{C}{\lambda^2} \left( \bar{\varepsilon}^{(q)}_i k_i - \bar{\eta}^{(q)}_i k^\prime_i \right) \\
&B_i^{(q)} = \bar{G}_i^{(q)} - \frac{C}{\lambda^2} \left( \bar{\varepsilon}^{(q)}_i k_i - \bar{\eta}^{(q)}_i k^\prime_i \right)
\end{align*}
\]

Finally, one gets the expression for a joint translation. Since \( \alpha_i \) is approximately perpendicular to the wire one may neglect \( \bar{\varepsilon}_i \cdot (2\bar{\varepsilon}^{(p)}_i + \bar{\eta}_i + \bar{\eta}^{(p)}_i) \) so:

\[
\begin{align*}
&a_i = -\lambda \frac{\left| B_i^{(q)} \right|}{\bar{G}_i^{(q)} - \frac{C}{\lambda^2} \left( \frac{k_i + k^\prime_i}{\lambda} + \bar{\varepsilon}_i \left( 2\bar{\varepsilon}^{(p)}_i + \bar{\eta}_i + \bar{\eta}^{(p)}_i \right) \right)} \\
&\text{Where} \\
&\left| B_i^{(q)} \right| = \bar{G}_i^{(q)} - \frac{C}{\lambda^2} \left( \bar{\varepsilon}^{(q)}_i k_i - \bar{\eta}^{(q)}_i k^\prime_i \right)
\end{align*}
\]

The main assumptions we used in order to simplify the model calculations and to disregard the least significant factors. (1) The basic assumption that \( \theta \ll 1 \) is also the main drawback of this model since it limits the wire curvature for a given number of segments. Still, for a reasonable number of junctions and reasonable curvatures this is turning out sufficient (2) another assumption is a quasi-static motion, in each calculation we will insert the GW in small displacement \( \bar{\varepsilon}_0 \) into the body. The movement causes changes the wall potential energy and consequently changes the shape of the GW to keep the minimum energy equations. This assumption is the reason that we neglected the kinetic energy and also dynamic friction. Since we
wanted to find the static behaviour of GW deformation this was sufficient for our usage (3) another assumption, derived from the rigid links model, is that the total length of the guide wire doesn't change throughout the procedure. This will simplify the calculations and reduce the DOF, however since the bending coefficient is much lower than axial deformation ability of GW this assumption won’t damage the final results (4) The last assumption is that we will neglect the static friction forces since they apply in parallel to the links which are not comprisable. Still the results were sufficient so for cases when higher accuracy level is needed, for a dynamic model where static and dynamic friction factors are changing and cases where hysteresis phenomena is important we recommend to use also the friction forces.

2.2 Geometrical interpretation

We used a local 3D auto-normal coordinate frames on each segment \((\hat{e}^{(p)}_i, \hat{e}^{(s)}_i, \hat{e}^{(q)}_i)\), where \(\hat{e}^{(p)}_i\) a unit is vector from \(X_i\) to \(X_{i+1}\) and \(\hat{e}^{(s)}_i,\hat{e}^{(q)}_i\) are arbitrary unit vectors which satisfy the conditions \((\hat{e}^{(p)}_i \perp \hat{e}^{(s)}_i \perp \hat{e}^{(q)}_i)\). In order to calculate \(\hat{e}^{(p)}_i\) we used the following equation:

\[
\hat{e}^{(p)}_i = \frac{X_{i+1} - X_i}{|X_{i+1} - X_i|}
\]

\(\hat{e}^{(s)}_i\) is calculated as the cross product of a general vector \(\hat{e}_i = (0,0,1)\) together with \(\hat{e}^{(p)}_i\), later on we calculate \(\hat{e}^{(q)}_i\). For each segment the following parameters are extracted and presented in each segment internal coordinate system:

\[
\begin{align*}
\eta_i &= \frac{1}{|\lambda_i|} (\lambda_{i+1} - \lambda_i) \\
\varepsilon_i &= \frac{1}{|\lambda_i|} (\lambda_{i+1} - \lambda_i) \\
k_j &= \lambda_i \cdot \lambda_{i+1} \\
k_j' &= \lambda_{i-1} \cdot \lambda_i
\end{align*}
\]

Next, using Eq. 2.22 in equations 2.192.202.21 2.21 gives us the required \(a_i\) and \(\psi_i\) needed for the iteration calculation.
2.3 Guide wire algorithm I

Implementing the above from section 2.1 and 2.2 we devised an algorithm that can calculate the guide wire's deformation as a function of the control translation and rotation parameters:

| Table 1: Navigation algorithm

---

1. Define for each segment \( i \) local coordinate unit vectors \( \hat{\lambda}_{i}^{(P)} \) and \( \hat{\lambda}_{i}^{(S)}, \hat{\lambda}_{i}^{(Q)} \) which defines SQ plane.

2. Add to each \( x_{i} \) a constant vector \( \varepsilon_{0} \) for quasi-static movement.

3. Calculate the potential vector \( G_{i} \) for each segment (Eq. 2.9).

4. Project \( G_{i} \) onto the segment's SQ plane:
   \[
   G_{i}^{(SQ)} = G_{i} - \left( G_{i} \cdot \hat{\lambda}_{i}^{(P)} \right) \cdot \hat{\lambda}_{i}^{(P)}
   \]

5. Calculate \( \eta_{i}, \varepsilon_{i}, k_{i}, k'_{i} \) for each segment \( i \) (Eq. 2.22).

6. Project the vectors \( \varepsilon_{i}, \eta_{i} \) onto the segment's SQ plane:
   \[
   \varepsilon_{i}^{(SQ)} = \varepsilon_{i} - \left( \varepsilon_{i} \cdot \hat{\lambda}_{i}^{(P)} \right) \cdot \hat{\lambda}_{i}^{(P)}
   \]
   \[
   \eta_{i}^{(SQ)} = \eta_{i} - \left( \eta_{i} \cdot \hat{\lambda}_{i}^{(P)} \right) \cdot \hat{\lambda}_{i}^{(P)}
   \]

7. Calculate \( B_{i}^{(SQ)} \) for each segment \( i \) (Eq. 2.19).

8. Calculate translation and rotation \( a_{i}, \psi_{i} \) (Eq. 2.20, 2.212.21) and apply it on each segment.

---

1 Since each segment's calculation effects its consecutive segment (Eq. 2.2), we need to start each iteration from the tip of the guide wire (segment \( N \)). find \( a^{N}, \psi^{N} \), add the vector \( \hat{\lambda}^{N}_{i} \) to the segment's position \( x^{N}_{i} \) and then continue on to the next segment \( N - 1 \). This method ensures that the tip position will be updated with each change in the previous segment. The iteration scheme will be performed in the following manner: \( (i^{N}),(i^{N-1},i^{N}),(i^{N-2},i^{N-1},i^{N}), (i^{2},i^{N-2},i^{N-1},i^{N}), \ldots) \).
2.4 Local forces model

This chapter describes our contribution. In Figure 8 we exemplify the limitation of the aforementioned algorithm. Clearly, the guide-wire fails to remain in the confinement of the artery, in other words the guide-wire deviates from the arteries boundary which has an unlimited strength comparing the GW. Note that, the algorithm is sequential in the sense that (if simply put), it relocates the distal segments according to the forces exerted on the proximal ones. So, a naive solution for this problem cannot be simply to enhance the potential field, since in that case, the wire makes “wild” turns (large proximal forces yield large translations in the distal segments).

Figure 8: GW model needs fine adjustment in high radius curvature

In order to get better accuracy with the same physical rigid-body technique, we have developed a new method for force refinement which is base on the assumption that the GW cannot be outside of the vessel/lungs walls since the model is a rigid one, therefore we adjusted the forces coming out from the potential field with the next algorithm.

2.4.1 Forces matrix

Recall the following formulas taken from Euler-Bernuli beams for two forces kinds applied on the GW. We shell use the superposition principle and add them one by one following the next two kind of forces which the walls apply on the GW. Since

This way the segments closest to the tip, will be subjected to the most iteration. We calculate the vectors \( a_i, \psi_j \) for the last segment \( N(N+1)/2 \) times.
we can use the superposition principle we will add them on after the other and accumulate their influence.

![Figure 9: Basic beam deformation formulas](image)

We shall define the forces and the displacement of the GW on the contact points \( \{i\} \) of the wire with its surroundings at points which distance from the base point are \( l_i \).

We shall do so by first summing up the forces along the beam (wire):

\[
\delta_i = \frac{F_1 l_1^2}{6EI} (3l_i - l_1) + \frac{F_2 l_2^2}{6EI} (3l_i - l_2) + \ldots + \frac{F_n l_n^2}{6EI} (3l_i - l_n)
\]

\[\text{2.23}\]
Here \( \delta_i \) denotes the deflection caused by applying forces \( F_i \), caused by the potential energy gradient and located before and after the contact force. The desired deflection is the one which will re-locate a segment \( i \) (which deviated from the arteries boundary) back into the artery. We also took into an account the distributed moment within the GW which corresponds to the radius of curvature. That is, in order to compute the desired forces, one should consider the forces applied to the artery walls by the wire itself. Empirically, we noticed that the radii of curvature should be much larger than those we achieved in simulation by applying forces alone (which in turn made it difficult to maintain the wire within the artery).

For a uniformly distributed moment one has:

\[
M = \int_{x} m(x)dx = EIv^*
\]

Where \( m(x) \) is the distributed moment (measured in [Nm]). Integrating twice with respect to \( x \) yields:

\[
\int_{x}^{L} \int_{0}^{L} m(x)dx dx = \frac{EI}{2.24} \nu_i
\]

While \( \nu_i \) is the current GW deflection which was caused by the external forces before modulation. We need this parameter in order to estimate the difference between the
current and the desired deflection. Note that \( m(x) \) and the current radius of curvature at \( x \) are related as follows:

\[
\left( \frac{EI}{\lambda} \sum_{i} \theta(i) \right) = \int_{x} m(x) dx
\]

Recalling from Chapter 2 that \( \cos(\theta(x)) = \hat{\lambda}_{j-1} \cdot \hat{\lambda}_{j} \) yields:

\[
\theta(x) = \sqrt{1 - \frac{1}{\lambda_{j-1}^{4}} \left( (x - x_{j-1}) \cdot (x_{j+1} - x_{j}) \right)^{2}}
\]

So Eq. 2.24 takes the form:

\[
\frac{EI}{\lambda} \left( \sum_{0}^{L} \sum_{0}^{L} \sum_{x} \theta(x) \right) = v_{i}
\]

Summing up the differences between the needed and current deflections using Eq. 2.26 and 2.23 for all forces and moments one have the following set of equations:

\[
A^{-1} \begin{bmatrix} \delta_{1} - v_{1} \\ \delta_{2} - v_{2} \\ \vdots \\ \delta_{n} - v_{n} \end{bmatrix} = \begin{bmatrix} f_{1} \\ f_{2} \\ \vdots \\ f_{n} \end{bmatrix}
\]

One calculates the matrix \( A \) taken from the force equations defined by Eq. 2.23. This matrix is defining the forces relations of Euler-Bernuli forces applied on beams:

\[
A = \frac{1}{6EI} \begin{bmatrix} 2l_{i}^{3} & l_{i}^{2}(3l_{i} - l_{i}) & \cdots & l_{i}^{2}(3l_{n} - l_{i}) \\ l_{i}^{2}(3l_{i} - l_{i}) & 2l_{i}^{3} & \cdots & l_{i}^{2}(3l_{n} - l_{i}) \\ \vdots & \vdots & \ddots & \vdots \\ l_{i}^{2}(3l_{n} - l_{i}) & l_{i}^{2}(3l_{2} - l_{i}) & \cdots & 2l_{i}^{3} \end{bmatrix}
\]

Eq. 2.27 formulates the relations between the desired shape and the forces need to be applied upon a beam to maintain that shape. However, one needs to remember that this is correct only for an Euler-Bernuli beam which in our case is not true so the solution only an approximated one therefore we need to be refine it in order to introduce the desired shape. One should also note that the desired deflections \( \delta_{i} \) found in Eq. 2.23 is usually defined as a perpendicular distance between the “X” axis to the deflected beam (see Figure 10), while in our case this is again is true. Again our answer to this is a numerical convergence which we shall describe below.
2.4.2 Translation

Original model have used a fixed translation vector for all segments. When the wire is threaded into a curved artery such a translation vector would cause the wire to transversely exit the artery wall. Needless to say, such a deviation is hard to fix, all the more so, when a multiple of deviations occur. Moreover, when the artery is an ‘>’ shaped (the wire acquire the same shape) translating the whole wire as a rigid body upwards causes the lower diagonal to exit to the right hand side of the artery while the upper diagonal exits to the right. In such a scenario (which is quite common) the wire will wildly swing as a result of the right and left side forces and their (temporal) non-continuities (that is, swinging from right to left makes forces switch sides). A much ‘smoother’ option would be to ‘slide’ the wire in a tangent manner, and this won’t cause big changes in the potential field gradient and will enable a stable solution to the algorithm during the threading into the arteries.

Add to each $X_i$ a small sliding movement, starting $\epsilon_0$ on base of the GW. Afterwards we will find a vector form each segment’s tip that has a direction of the next segment and keeps the segment length. See the differences in the pictures below.

![Figure 11: GW constant translation (left) versus sliding translation (right)](image)

In order to check the validity of this sliding movement, we will check the influence on a the equation which defines $\delta X_i$ and is its influence on the final equations for $a_i$ and $\psi_i$. 

Recall that for a fixed movement of $\vec{e}_0$ for each link from Eq. 2.2
\[
\delta \vec{x}_i = \vec{e}_i + \sum_{j=0}^{i-1} \vec{a}_j = \vec{e}_0 + \vec{a}_0 + \ldots + \vec{a}_{i-2} + \vec{a}_{i-1}.
\]
So translating all segments with $\vec{e}_0$ vectors makes the product expression $(\vec{x}_i - \vec{x}_{i-1}) \cdot (\vec{x}_{i+1} - \vec{x}_i)$ in Eq. 2.7 become

\[
(\vec{x}_i + \delta \vec{x}_i - \vec{x}_{i-1} - \delta \vec{x}_{i-1}) \cdot (\vec{x}_{i+1} + \delta \vec{x}_{i+1} - \vec{x}_i - \delta \vec{x}_i)
\]

after movement. Recalling the definitions for $\vec{a}_i$, one writes $\delta \vec{x}_i = \vec{x}_{i-1}$, $\delta \vec{x}_{i+1} = \vec{a}_i$ yield:

\[
(\vec{x}_i - \vec{x}_{i-1} + \vec{a}_i) \cdot (\vec{x}_{i+1} - \vec{x}_i + \vec{a}_i) = (\vec{\alpha}_{i-1} + \vec{a}_i) \cdot (\vec{\alpha}_i + \vec{a}_i) = 2.29
\]

There are differences between constant $\vec{e}_0$ and sliding $\vec{e}_i$ movements. In the last, each segment has a different translation vector. Therefore, defining a sliding movement $\vec{e}_i = \tau (\vec{x}_i - \vec{x}_{i-1})$ where the coefficient $\tau < 1$ and indicates how fast the sliding is. The total translation is defined as $\delta \vec{x}_i = \sum_{j=0}^{i-1} \vec{e}_j + \sum_{j=0}^{i-1} \vec{a}_j$, So rearranging Eq. 2.29 which is after the sliding movement:

\[
(\vec{x}_i - \vec{x}_{i-1})(\vec{x}_{i+1} - \vec{x}_i) =
\]

\[
= \left[ \vec{x}_i + \sum_{j=0}^{i-1} \vec{e}_j + \sum_{j=0}^{i-1} \vec{a}_j - \vec{x}_{i-1} - \sum_{j=0}^{i-1} \vec{e}_j - \sum_{j=0}^{i-2} \vec{a}_j \right] \left[ \vec{x}_{i+1} + \sum_{j=0}^{i-1} \vec{e}_j + \sum_{j=0}^{i-1} \vec{a}_j - \vec{x}_i - \sum_{j=0}^{i-1} \vec{e}_j - \sum_{j=0}^{i-1} \vec{a}_j \right]
\]

Becomes $(\vec{x}_i + \vec{e}_i + \vec{a}_{i-1} - \vec{x}_{i-1})(\vec{x}_{i+1} + \vec{e}_{i+1} + \vec{a}_i - \vec{x}_i)$ which sums up to

\[
(\vec{x}_{i-1} + \vec{e}_{i-1}) (\vec{x}_{i+1} + \vec{e}_{i+1} + \vec{a}_i - \vec{x}_i)
\]

Finally the dot product for calculating the GW bending energy becomes:

\[
(\vec{x}_i - \vec{x}_{i-1})(\vec{x}_{i+1} - \vec{x}_i) =
\]

\[
= \vec{x}_{i-1} \vec{x}_i (1 + \tau)^2 + \vec{e}_{i-1} \vec{e}_i (1 + \tau) + \vec{a}_{i-1} \vec{a}_i (1 + \tau) + \vec{a}_i \vec{a}_{i-1}
\]

Substituting Eq. 2.30 into the bending energy expression (Eq. 2.7) and Differentiating it yields:

\[
\delta U_{i \rightarrow i+1}^{Bend} = -\frac{C}{A^2} \left[ \vec{x}_{i-1} \vec{a}_i (1 + \tau)^2 + \vec{e}_{i-1} \vec{e}_i (1 + \tau) + \vec{a}_{i-1} \vec{a}_i (1 + \tau) + \vec{a}_i \vec{a}_{i-1} \right] \delta \alpha_i
\]

\[
\delta U_{i+1 \rightarrow i}^{Bend} = -\frac{C}{A^2} \left[ \vec{x}_{i+1} \vec{a}_{i+1} (1 + \tau)^2 + \vec{e}_{i+1} \vec{e}_{i+1} (1 + \tau) + \vec{a}_{i+1} \vec{a}_{i+1} (1 + \tau) + \vec{a}_{i+1} \vec{a}_i \right] \delta \alpha_i
\]

Expanding the differential expression and neglecting high order terms one have:
\[
\delta U_{i}^{\text{Bend}} = -\frac{C}{\lambda^4} \left( \sum_{i=1}^{2} \left( \tilde{L}_{i}^{-1} \tilde{L}_{i+1} (1+\tau)^3 + \alpha_{i-1} \tilde{L}_{i} \tilde{L}_{i+1} (1+\tau)^2 + \alpha_{i} \tilde{L}_{i+1} (1+\tau)^2 + 2\alpha_{i-1} \tilde{L}_{i+1} (1+\tau) \right) \delta \alpha_{i} \right)
\]

Since \( \alpha_{i} \) is perpendicular to \( \tilde{L}_{i} \) and \( |\alpha_{i}| \ll |\tilde{L}_{i}| \) the expression reduces and the total GW bending energy of i-th segment is approximately:

\[
\delta U_{i}^{\text{Bend}} + \delta U_{i-1}^{\text{Bend}} \approx -\frac{C}{\lambda^4} \left[ \alpha_{i} \tilde{L}_{i} \tilde{L}_{i+1} (1+\tau)^2 + \tilde{L}_{i+1} (1+\tau) + \tilde{L}_{i+1}^2 \alpha_{i} \right] (1+\tau)^2 \delta \alpha_{i}
\]

The new derivative as the expression of the force \( F_{i} \) (as in Eq. 2.11 for constant translation)

\[
\frac{\delta U_{i}^{\text{int}}}{\delta \alpha_{i}} = \mathcal{G}_{i} - \frac{C}{\lambda^2} \left[ \frac{k_{i} (1+\tau)}{\lambda} + \alpha_{i} \left( \tilde{e}_{i}^{(p)} + \tilde{\eta}_{i}^{(p)} \right) \right] \left( \tilde{e}_{i}^{(p)} + \tilde{\eta}_{i}^{(p)} \right) = \tilde{F}_{i}
\]

Therefore the final equations which give the deflation of the GW have changed with the additional \( \tau \) expression leading to new \( a_{i} \) and \( \psi_{i} \) which will compensate the change in the GW equations made by the sliding movements.

\[
a_{i} = -\frac{|B_{i}^{(SQ)}|}{\mathcal{G}_{i}^{(p)} - \frac{C}{\lambda^3} \left( k_{i} + k_{i+1} + \tilde{e}_{i}^{(p)} k_{i} + \tilde{\eta}_{i}^{(p)} k_{i} \right)}
\]

\[
|\tilde{B}_{i}^{(SQ)}| = \mathcal{G}_{i}^{(p)} - \frac{C}{\lambda^3} \left( \tilde{e}_{i}^{(SQ)} k_{i} + \tilde{\eta}_{i}^{(SQ)} k_{i} \right)
\]

We have added the sliding movement to prevent large non-linearity in the potential filed algorithm and found that during the sliding translation, adding this \( \tau \) expression will have a minor influence on the result. However, using this sliding GW method increased the stability of the simulation and partially solves the stability issue occur in constant translation. Only using all the improvement mentioned in this section enabled us to have a stable GW threading in the simulation.

### 2.4.3 Forces calculation

Since contact handling of GW collision with body is very sensitive because of non-linear changes in the potential field. We have searched for a robust method to get the contact areas and calculate the applied forces. Therefore to have the GW
special representation we have defined search on the link using a random spherical vector around each GW node to find all possible contact points of the GW with the arteries and to locate the critical areas which could be contact points in the next iteration (see Figure 12, it gives volume to the GW). The critical points are points that are considered to be out of the arteries model or in areas which are closer to the wall. For each segment we will calculate a “Position error” which represents the distance of each point form the tissue surface (if the point inside the arteries the value is zero).

Using the Position error vector, we will divide the wire into segments by the contact areas and contact local forces, in the segments which have the highest position error. The force direction is calculated from the average direction of all segments which are out of the lung in the same area. This way we will have less impacts of the non-linearity potential field during the GW movements since we will filter the changes by using this average. In the images of Figure 13 you can see the force act on each are segment during the threading. Those forces are calculated inorder to push the wire inside the lung’s bounderies. Using the forces matrics form section 2.4.1 to push the GW into the bounderies and together repeatative algorithm (which has a treashold of the summery of max position error) will reduce the GW

Figure 12: Possible contact points with lungs.
Left – one random iteration, Right – Critical areas total random search.
transverse exits through the boundaries and find a minimum energy solution for the problem inside the arteries.

Figure 13: Force modulation vectors in few insertions positions.

we can also see that the direction of the forces is perpendicular to the wire location on the lungs.

2.4.4 Numerical Relaxation

After the forces have been calculated we will use a numerical relaxation mode on the GW to make sure that the forces which act on it, will provide the GW final deformation. We need this mode in order to overcome one of the basic limitations of this method which is that the segment changes in small angles in order to have valid equations. In addition, in the naive potential field operation (see the algorithm in section 2.3 and the potential field calculation 4.2), the value of the force is changing as a result of the displacement. Therefore, the forces which act on the
segment always change since the GW changes its position to the new location which is closer to the wall and has a smaller gradient change. The Normal forces which are causing the GW deflation hence could be reduced sometime up to zero when the GW is inside the arteries. The change in the force will lead to change in the deflection of the GW and sometime we see that the wire is swinging from side to side trying to reach equilibrium.

We have implemented this change on the new algorithm by using the same forces until reaching a final GW deformation. When the forces don’t make any further deformation on the GW, we know that we have found the minimum energy solution. The force acting on the GW and the internal Torsional stiffness elastic forces have reached equilibrium. This way, we have increased the model stability and add more flexibility to each movement. Although we are limited with small angles, the numerical relaxation mode will provide the maximum extended angle by summing the deflections over all iterations. We have continued to slide the GW only after we make sure that the forces have applied in maximum deformation.

Figure 14: Relaxation and sliding moments.
2.5 Local Forces Algorithm

Finally we will incorporate the force matrix, sliding movement and numerical relaxation mode in our algorithm to improve the result we got in the previous method in section 2.3. Our technique is using the naïve model as a platform to represent the GW elasticity together with new techniques which we have introduced to improve the simulation results and gain a stable solution.

---

### Navigation algorithm II Pseudo-code

| Given: The GW position: \( x_i \in \mathbb{R}^3 \) for each segment. |
| Output: The GW position after one step using new forces modulation. |

1. Add to each \( x_i \) a small sliding movement, starting \( \varepsilon_0 \) on base of the GW.
2. Random searching for critical spots with high position error in force areas.
3. Dividing the GW into segments using the force spots.
4. Calculate for mean force and direction each force spot.
5. Initial force estimation based on adding the forces matrix to push the GW inside the arteries and minimize the segments position error
6. Applying the new forces on the GW
7. Relaxation until the GW deform to the final position (if position error is higher than threshold continue to section 8 else follow section 1).
8. Using bisection method to make additional adjustment on forces to have less GW point outside the wire (returning to section 6).

---

### Table 2: Navigation algorithm with improvements

Later, on section 5 we will explore the results and compare the two algorithms introduced in sections 2.3 and 2.5.
3 Overall System description

Following the above we recognize a need for a full automatic solution which will insert the GW into the body taking into account all the external and internal forces which apply on the wire. We have designed a prototype of a robotic manipulator which can hold the GW, navigate it into the body using a 2 DOF mechanisms (translation and rotation), sensor system (with GW identification) and GW simulation which is used to predict the GW behaviour inside the body. This robotic system will navigate the GW through the blood circulatory system, the respiratory system or other internal cavity which currently using the treatment of catheter, stent or bronchoscope.

One of the basic goals of this thesis is to design robust systems that given a target point, will autonomously lead the guide wire to the desired location.

![Diagram](image)

**Figure 15: The process Flow chart**

In order to collect the vast amount of geometrical (internal cavity space) data needed for the simulation one may acquire the data (prior to the procedure) using an MRI 3D scan. Once collected, data may be used for inverse kinematics computation to set the manipulator’s motors. Using the simulation, one could also calculate the forces applied on the internal tissues thus avoiding expected damage to the tissue.
Closing the loop, that is, acquiring a current wire configuration (via an X-ray/3D-CT scan) and applying it back into the simulation will minimize inaccuracy (see Figure 16). After reaching the target multiple treatments could be applied such as Catheterization, Stent insertion, tissue biopsy or scoping the area.

In order to successfully insert the wire to target, the following subsystems should be fabricated: Scanning sensors; navigation manipulator and GW simulation software:

![Diagram](image)

**Figure 16: The process block diagram.**

In this block diagram we can see that computation system will incorporates the data collected from the sensors back into the simulation making RT corrections, simulate the next steps and manipulate the wire to the next position.

### 3.1 Mechanical system – GW manipulator

In order to perform the medical procedures described above, Torsional movement from the GW base, and a linear translation is sufficient. Using these, we can manipulate the GW from its base to the target configuration. When reaching a junction, the system may choose a desired path by manipulating both rotation and translation modes. Note that in order to overcome buckling of the GW (which
commonly accrues) we require that both modes could be operated simultaneously. Therefore we could actually “thread” the GW using them together.

![Figure 17: The manipulator system design](image)

**Linear movement** of the GW is possible by using a step motor together with spur gear transmission which includes three spur gears. The first gear is mounted on the motor while remaining connected to the tubes with a ratio of 2:1 to the motor. The tubes are mounted on the axis and roll in opposite directions. Two rollers hold the GW by pressing it from both sides. We used this mechanism in order to execute symmetrical forces on the GW, pushing it into the body.

**Torsional movements** of the GW, we used second step motor with lead screw which promotes one of the tubes relative to the other executing a rolling movement on the GW from its base. The shaft, which holds the upper tube, has hexagon profile to enable both rotational and linear movements.
We have selected two low-voltage Bipolar step motors, first rotary stepper motor (PK244M-02AA, Torque 0.26[Nm]), second is an integrated stepper with lead screw (11HY302-102N, Force 100[N]). The Controller is “Arduino-Duemilanove” unit with “Adafruit” motor shield. These actuators work in low speed and high torque and can be controlled without encoder which simplifies the complexity of the system.

Figure 18: The manipulator system manufactured prototype

Figure 19: Stepper Motors control and power board
3.2 Sensor system – GW and space identification

This system is the process identification which close loop on the GW position and makes real-time correction to improve model position accuracy and helps monitoring the GW movements during the medical process. There are known technologies to determine the GW real-time deformation most of them are using fluoroscopic images coming from CT and graphically projecting it on a 2D or 3D image. Since we are using a theoretical model to calculate the navigation of the GW into the body, there will be a difference between the calculation and actual movement. We will use the GW identification inside the body and update the GW simulation. The simulation could work without any updates, but we assume that using this mode will improve process safety and GW navigation performances especially when need to get into high angle curves.

In our preliminary tests, we have found the GW position using data collected also by using video camera. Applying Motion detection algorithm on this raw data, we have identified the GW while moving. One of the disadvantages of this method is that the wire needs to constantly move GW in order to be identified.

![The Video Motion detection algorithm](image)

Later, we have started to use only few frames which were acquired during the GW motion, after manipulating the GW. To acquire the GW position we have used a frame technique which take one frame and calculate the 2D position over the picture. The identification algorithm is described in Table 3.
GW Identification Algorithm

Given: RGB image after GW movement.
Output: GW coordinates \( x_i \in \mathbb{R}^2 \).

1) Acquiring the image

2) Cropping only A4 borders and reshaping to overcome registrations issues

3) Highlighting all pixels leaving only the black pixels of the GW on the image.

4) Finding all GW points (marked with red dots)

5) Using least square to build smooth mathematical function.

Table 3: GW identification algorithms
Applying this algorithm we have succeed to compare the data from the simulation and getting better understanding of how we need to improve the GW simulation. In real time process, we need the ability to identify only the GW tip since it has a major role in GW navigation through curvatures. The tip is usually curved to enable the operator to select the needed path by using torsional movements on the base of the GW. This mode impacts directly on the free tip and enable to select a path, therefore identifying the tip with pattern recognition or a magnetic field is essential to real time navigation.

![Diagram of GW navigation](image)

**Figure 21: Three steps for GW navigation using pattern recognition**

### 3.3 Process simulation – GW deformation

In the process the simulation is being used in two modes: The first mode is used prior the procedure and made mainly to create an initial simulation from insertion point to the target. In this mode we have a theoretical data that will be used to understand whether it is possible to reach the target location and also to determine what are the risks performing this process by calculating all the stress in the tissues which deform the GW. The second mode is a real time mode which uses the current GW location and simulate in the next steps of insertion from the current GW location.

In this thesis we will deal with the first mode and compare the simulation results to experimental behaviour of a GW. The physical model of the simulation is described was described in section 12.
4 Experimentation

In order to check the validation of the algorithm and to see our improvement, we will thread the GW into a known geometry and overlay the simulations and check the deviation of simulation from real GW experiment. The experiment we have performed is navigation into a desired location using the GW manipulator, in order to move the GW accurately. The geometry which simulates the arteries/lungs branches was designed for this cause and was printed using 3D printer. We have used the following setup which is also described in section 3.

![Experiment setup](image)

**Figure 22: Experiment setup**

Using collision detection function, which will be discussed on section 4.2, the algorithm calculates the potential energy of the wire points in each location. Using the naïve model form section 2.3 together with non-linear forces correction from section 2.5, will get the GW deformation after each insertion inside the phantom. We have developed also GW identification algorithms to calculate the differences between the simulation and real GW from the experiment. For this cause we have used a CCD sensor of a camera which gives a 2D representation of a GW therefore we have produced the phantom with “cut-offs” from above to be able to identify major parts from the GW shape and to re-construct it. The simulation could be used
also for 3D shapes and to simulate a GW deformation, but since the identification process is much complicated and need CT/X-Ray equipment we choose only to check the algorithm in 2D environment. We choose also a simplified phantom of branches made from non-deformable material (comparing the GW) therefore we didn’t want the GW to exit from the simulation wall. In this phantom we can make few different experiments and check different forces and bending abilities. We produced its branches’ diameter to be narrower as the GW inserted deeper in order to have more realistic shape. The phantom has a similarity to lungs but could represent any system of vessels toward the heart as the aorta branches, which become successively smaller in diameter, down to the arteriole.

4.1 Experiment phantom

The phantom we used will simulate the first branches of the lungs and it will enable us to simulate a medical procedure inside the lungs. GW Navigation in the respiratory system is unique and could be used for multiple of operation inside the Trachea into the repertory branches. The branches phantom is open to enable us to use standard camera instead of X-ray system. The camera will be mounted from above and will be used to identify major parts of the GW. We believe that combining the GW deformation algorithm with the inputs coming from the GW identification algorithm will give better accuracy to the GW deformation algorithm still in this thesis we haven’t used this mode. This phantom model is a 3D model which lies on 2D surface, mainly designed to simplify the GW identification for the preliminary tests. This model was built using real diameters of the trachea and in each segment the diameter changes from 20 to 5 [mm] with the following dimensions of inner and outer wall diameters.
4.2 Potential field algorithm

One of the great challenges, in this thesis, is to have a good representation of the body tissues in order to calculate the potential energy correctly and use it with navigation algorithms. We have created a function that takes one point from the GW special representation and calculates the distance to closest tissue surface. The body tissue is divided to cylinder shaped segments and for the model illustrated in Figure 23 we have used 11 segments. To simulate real body tissues, it is possible to divide each segment into smaller segments with different radii to achieve better representation of real body tissues which has usually much complicated shape. The force each segment applies on the GW is simulated as linear spring $\vec{F} = g \Delta \vec{x}$ with spring constant $g$ as a function of the tissue deformation $\Delta \vec{x}$ caused by interaction with the GW. Each of the GW nodes $i$ has a potential energy $U_i = g \Delta x_i^2 / 2$ from the collision with the tissue, the gradient change is creating a wall force with direction is $\hat{d}_i$ (a unit vector of $\vec{d}$ normal to the body tissue) therefore we will define the local segment force as the potential energy multiply with wall normal $\vec{g}_i = U_i \hat{d}_i$. Since we are searching for the total wall vector force $\vec{G}_i$ (see Table 2) we will collect all the
forces applied on the GW segments and accumulate all consecutive points, from point $i$ to last point $n$ (which is the GW tip), finally we will sum all the vectors and receive $\vec{G} = \sum_{i}^{n} U_i \hat{d}_i$

The GW deformation algorithm is capable simulating the GW deformation in a 3 DOF curvature. The deformation algorithm calculates potential field of a curved pipe with multiple junction given by three $s$ functions.

\[
X = f(s) \\
Y = f(s) \\
Z = f(s) = 0 \\
R = f(s)
\]

In order to calculate the GW deformation to the desired end point location, we will build a path to the trajectory via all the needed braches which the GW needs to pass through by using FBS algorithm in Table 4. Therefore the first step is dividing the model to branches, using search algorithm to find the needed path through the branches, building curvature from the selected path through the desired end location.

\[\text{Figure 24: Lungs branches segmentation by levels}\]

In order to navigate inside the lungs we have build a 3D model of lungs potential that can be used to reach any needed location. The target defines the needed route inside the lungs and for each insertion. We have defined 3 different routes which will are tests we would like to compare between the GW simulation and the RT threading.
To get a better understanding of the potential field function, we have made two images of normalized 2D slice potential field we used in order to navigate inside the lungs model. This way one can see the potential behaviour caused by squaring the distance from the lungs walls.

This is only a 2D slice of the potential field, Z is defined as zero, however it is also possible to use this method in order to navigate in 3D potential field. The red symbolize zero potential energy as in the model, and the topography symbolize the potential energy at those points.
To calculate this potential field we have used a function which will give a potential vector over the whole simulation range. This field is which showed in the figure above is made from all segments of lungs. We have defined each lung’s segment using its first point $P_{1_{Segment_i}}$ and its last points $P_{2_{Segment_i}}$.

$$P_{1_{Segment_i}} = \begin{bmatrix} x_1 & y_1 & z_1 \end{bmatrix}^T$$
$$P_{2_{Segment_i}} = \begin{bmatrix} x_2 & y_2 & z_2 \end{bmatrix}^T$$

We will make from those vectors two main vectors which define the whole lungs model.

$$P_1 = \begin{bmatrix} P_{1_{Segment_1}} & P_{1_{Segment_2}} & \ldots & P_{1_{Segment_n}} \end{bmatrix}$$
$$P_2 = \begin{bmatrix} P_{2_{Segment_1}} & P_{2_{Segment_2}} & \ldots & P_{2_{Segment_n}} \end{bmatrix}$$

Finally, we will select only the active segment needed to reach the target from insertion point, in order to reduce the calculation time. For each random point $P = \begin{bmatrix} x & y & z \end{bmatrix}^T$, which is located of the GW, we will calculate the distances from each segment separately from segment 1 to last n performing all next calculation over the selected segments and creating two vectors, one with distance and the other with a unit vector from the point to each segment. We will define the vector $V$ (see Figure 27) from each segment to the random point.

$$V_i = \left( P_{2_{Segment_i}} - P_{1_{Segment_i}} \right)$$
$$V'_{i_1} = \left( P - P_{1_{Segment_i}} \right)$$
$$V'_{i_2} = \left( P_{2_{Segment_i}} - P \right)$$
$$\hat{V'}_i = \frac{V'_i}{|V'_i|}$$

![Figure 27: Segment’s Vector definitions](image)
Now we will check the next condition

\[ A = (\hat{V}'_2 \cdot \hat{V}')(\hat{V}'_1 \cdot \hat{V}') \]

We will follow the next algorithm to define \( P'_v \) and \( D' \):

If \( A > 0 \) we will define another condition: \( B = |V''_2| - |V''_1| \) and find \( P'_v \) and \( D' \):

\[
D' = \min \left( |V''_1|, |V''_2| \right)
\]

\[
\begin{align*}
B > 0 : & \quad P'_v = \frac{V''_1}{|V''_1|} \\
B < 0 : & \quad P'_v = \frac{V''_2}{|V''_2|}
\end{align*}
\]

If \( A < 0 \) we will define \( V''_n = V''_2 - (V''_2 \cdot \hat{V}') \hat{V} \) and find \( P'_v \) and \( D' \):

\[
D' = |V''_n|
\]

\[
P'_v = \frac{V''_n}{|V''_n|}
\]

Making this calculation of distance and potential vectors to all segments and choosing the closest segment to the point which has a minimum distance.

\[
P_v = \begin{bmatrix} P'_{v1} & P'_{v2} & \cdots & P'_{vn} \end{bmatrix}
\]

\[
D = \begin{bmatrix} D'_1 & D'_2 & \cdots & D'_3 \end{bmatrix}
\]

\[
P_v = P_v \mid_{D_{\text{min}}}
\]

See in the picture below 3 points which shows potential vector selection in arbitrary points near the desired path.

Figure 28: Possible potential vector calculation for three different points
We have created a mashed grid to check the validity of this algorithm. We can see that the vectors direction is logical and lead us to the normal of the closest wall.

![2D vector map for potential field](image1)

**Figure 29: 2D vector map for potential field**

One of the first steps we have check if refinement of the map by averaging the mesh grid. We can see that the vectors direction gradient is smoother in the left figure and we have adopted this method in section 2.4 with the adaptive forces calculation.

![Potential map refinement](image2)

**Figure 30: Potential map refinement**

### 4.3 FBS Algorithm

We will find the right path using FBS (First breath search). This method is the natural method for using in such navigation because the path to a desired location pass through multiple junctions and could be found in a single way. The junctions and the branches will be stored in data base as a navigation map and when we will
change the trajectory we recharge the path and create new potential field. For example I will show the algorithm for Figure 24.

Table 4: FBS algorithm for route selection.
5 Results

In order to understand the improvement using the new algorithm, we used the mechanical setup of the GW (see Figure 22) and made offline simulation using the two algorithms introduced in section 2.3 and 2.5. We saved the simulations outputs and analyze them versus the set of images that were taken during the experiment (see image processing in Table 3). In order to understand the differences we choose few analysis methods which are usually used in such experiments with GW model. This way we could see the performances differences between each of the methods. The methods we will use are:

(1) Overlaid pictures of the two simulations on top of each experiment. We will capture the GW in few of the main steps during the GW insertion to get a better visual understanding of the special 2D differences. In those next figures we will mark the GW as follow:

Green - Local Forces Model
Blue - Naive Model
Black - Experimental results

(2) The total RMS error of the differences between all links. RMS error calculation was made using the expression below:

\[ E_{\text{RMS}} = \sqrt{\frac{1}{N_{\text{segment}}} \sum_{j=1}^{\text{num}} |X_{\text{real}}^j - X_{\text{simulation}}^j|^2} \]

(3) The Position Errors of each link from real GW in the last insertion mode

(4) DSC coefficient (Dice similarity coefficient) which represent overlay percentage of the two GW. The DSC is a common similarity measure and is defined as the ratio of twice the common area to the sum of the individual areas of the two objects, in this case the real (R) and the Virtual (V) GW. For this propose we have used the equation below:

\[ DSC = \frac{2|R \cap V|}{|R| + |V|} \]

We are looking for a DSC coefficient which is above 80% in order to have a persisted estimation of a GW.
5.1 First Target Navigation

**Green** - New Algorithm, **Blue** – Naive Algorithm, **Black** – GW experimental results

Table 5: First Experiment results of simulations VS real GW. Insertions of the GW of 12,13,14,16 [cm] into the model.
Figure 31: First target simulation
Naïve (Blue) and Force-Loci (Green) in comparison to experiment GW results (Black) of 17 [cm] insertions into the phantom.
**Figure 32: First simulation Position Errors**

**X axis:** Number of the element (0 is the base N is the tip).  **Y axis:** Overall segment position deviation from the GW in [mm]. This error was taken from the last iteration with total GW length of 22 [cm]. **Left-** Force loci, **Right - Naïve model**

**Figure 33: Total segments RMS Error [mm]**

**X axis:** Number of iteration (each iteration is a 5 [mm] insertion).  **Y axis:** RMS Error [mm]. This error is calculating all simulation segments’ deviation from the real GW. **Left-** Force loci, **Right - Naïve model**
Figure 34: First experiment DSC (%) coefficient for two simulations

**X axis:** Number of iteration (each iteration is a 5 [mm] insertion). **Y axis:** DSC (%) coefficient. The DCS coefficient expresses the similarity of the simulation versus real GW position. **Green**- Force loci, **Blue** - Naïve model
5.2 Second Target Navigation


Table 6: Second Experiment results of simulations VS real GW. Insertions of the GW inside the phantom in 16, 17, 18.5, 19.5 [cm]
Figure 35: Second target simulation

Naïve (Blue) and Force-Loci (Green) in comparison to experiment GW results (Black) of 20 [cm] insertion into the phantom.
Figure 36: Second simulation Position Errors

**X axis:** Number of the element (0 is the base N is the tip). **Y axis:** Overall segment position deviation from the GW in [mm]. This error was taken from the last iteration with total GW length of 22 [cm]. **Left:** Force loci, **Right:** Naïve model

Figure 37: Total segments RMS Error [mm]

**X axis:** Number of iteration (each iteration is a 5 [mm] insertion). **Y axis:** RMS Error [mm]. This error is calculating all simulation segments’ deviation from the real GW. **Left:** Force loci, **Right:** Naïve model
Figure 38: Second experiment DSC (%) coefficient for two simulations

X axis: Number of iteration (each iteration is a 5 [mm] insertion). Y axis: DSC (%) coefficient. The DCS coefficient expresses the similarity of the simulation versus real GW position. Green - Force loci, Blue - Naïve model
5.3 Third Target Navigation

Green - New Algorithm, Blue – Naive Algorithm, Black – GW experimental results

Table 7: Third Experiment results of simulations VS real GW. Insertions of the GW inside the phantom in 14.5, 16, 18, 20.5 [cm]
Figure 39: Third target simulation

Naïve (Blue) and Force-Loci (Green) in comparison to experiment GW results (Black) of 22 [cm] insertion into the phantom.
Figure 40: Third simulation Position Errors

X axis: Number of the element (0 is the base N is the tip). Y axis: Overall segment position deviation from the GW in [mm]. This error was taken from the last iteration with total GW length of 22 [cm]. **Left-** Force loci, **Right -** Naïve model

Figure 41: Total segments RMS Error [mm]

X axis: Number of iteration (each iteration is a 5 [mm] insertion). Y axis: RMS Error [mm]. This error is calculating all simulation segments’ deviation from the real GW. **Left-** Force loci, **Right -** Naïve model
Figure 42: Third experiment DSC (%) coefficient for two simulations

X axis: Number of iteration (each iteration is a 5 [mm] insertion). Y axis: DSC (%) coefficient. The DCS coefficient expresses the similarity of the simulation versus real GW position. Green- Force loci, Blue - Naïve model
5.4 Results Summary

The main reasons for evaluating a new method for local forces was (1) to gain stability of the GW in the simulation and (2) to have a better representation of the wires and reduce the deviation of the GW form the wall boundaries (see section 2.4). We took all the experiments’ figures from sections 5.1 to 5.3 and reviewed them using a summary table to analyse them one versus the other and to get an overall review of those methods. In this table below, we can see 3 main methods: RMS (Mean and STD), Max Position error, and DSC which are been used to understand accuracy and bending abilities and ‘Number of elements’ which is important to understand the stability.

<table>
<thead>
<tr>
<th></th>
<th>First Experiment</th>
<th>Second Experiment</th>
<th>Third Experiment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Naïve</td>
<td>FL</td>
<td>Naïve</td>
</tr>
<tr>
<td>Number of Elements</td>
<td>20</td>
<td>20</td>
<td>50</td>
</tr>
<tr>
<td>Max Position error [mm]</td>
<td>5</td>
<td>3.5</td>
<td>30</td>
</tr>
<tr>
<td>Mean of RMS[mm]</td>
<td>1.72</td>
<td>1.42</td>
<td>2.27</td>
</tr>
<tr>
<td>STD of RMS[mm]</td>
<td>1.95</td>
<td>1.38</td>
<td>0.65</td>
</tr>
<tr>
<td>Mean DSC</td>
<td>82%</td>
<td>83%</td>
<td>60%</td>
</tr>
</tbody>
</table>

Table 8: Summary - Naïve algorithm VS Force-loci

5.4.1 Element number

In the GW simulations it is possible to use different lengths of segments to represent the GW model. Since we encountered stability and accuracy issues with the naive model, we increased the numbers of segments by reducing their size and by doing so, gain more flexibility. One of the main reasons that cause those issues was the assumption of small theta angles between each segment. This is true in a case that we will use miniature segments. However, small segment leads to high computational time which makes this method to be more complicates and not usable for RT environment. The number of segment ratio we used in the simulation (not including the first experiment), were 20/50 and 30/100, which is 2 to3 times less segments with the new algorithm. Need to remember that with the same number of elements, in experiment 2 and 3, we have encountered an unstable solution. Therefore, we have found the minimum number of element which needed to simulate those tests with the naive solution.
5.4.2 Bending ability

The other reasons that we have searched for improvement in the naive algorithm was because of the high deviations from the boundaries wall. If we look on the second experiment results, we can clearly see an issue with the final GW position using the naive model (see blue GW in Figure 35). The GW has finished the second navigation test with a big position error of 30[mm] (deviation in the last segment see Figure 36) while in the new method with the final position error was 5.5[mm]. This error is critical because the total diameter of this branch is 9 [mm] at this area, therefore an error of 30[mm] is 3 times more than the segment’s diameter at this area. Using the new method have improved the simulation and reduced the error to be 6 times lower and of course inside the boundaries of the model. Looking at the position error we can also see another improvement in the third experiment, looking at the last iteration, the error of the GW tip was reduced from 5[mm] to 1[mm]. In this case we should also consider that the branch diameter is 4[mm] therefore a 5[mm] error again takes the GW to be out of the model area.

5.4.3 Accuracy

Looking at the RMS and DSC criteria to check the overall accuracy of this model, we can see also an improvement mainly in the second experiment. The RMS error was 25% lower (2.25[mm] Naive, 1.7 [mm] New) and had 50% less variations during the simulation (STD: 0.6[mm] Naive, 0.3 [mm] New). In addition we can see an improvement of 20% in the accuracy taken from of the overall mean RMS of all three experiments. The mean of all iterations in the new method is around 1.5[mm] while the naive model has an average of 1.9[mm]. Usually RMS error of 1[mm] defines good simulation results therefore we still need to improve this model, however, we should mention that few of the iterations with the new method had an error which was lower than 1[mm]. Therefore we can say that the accuracy of the new model is slightly better than naive model.

Looking at the results of the DSC coefficient in the second experiment we can see that in the final position of the new method has 80% similarity while in the naive algorithm we had only 30%. The average DSC has slightly improved in the second and third experiments, it means that the new model provides better similarity to the
real GW movements (Second Experiment: 60% - Naive, 75% - New, Third Experiment: 72% - Naive, 85% - New).

Note that we are using less segments in the new method comparing the naive, therefore, achieving better accuracy results with lower element number is an improvement. Reducing element number will reduce the calculation time and which is very important in RT systems. However, since we haven’t searched for time optimisation, it is not presented in this thesis.

5.5 Conclusions

We shall divide the conclusions for two, at first we will deal with the simulation and then with the mechanical system.

From the simulation results, as we can see in section 5.4, we overcome the issues which were caused by the small angles assumption. This assumption was the main drawback of our model since it limits the wire curvature for a given number of segments. Using the local forces model made it possible to handle the non-linear behaviour of GW collisions with the walls. Together with a reasonable number of junctions and reasonable curvatures this is turn out sufficient for this prototype testing system in position the GW tip in the phantom. However, it seems like we reached the limitation of this method and only with our modifications we succeed reaching the targets properly.

In order to achieve higher accuracy using this simulation we believe that it is important to add also the friction forces which are caused by the pressure of the GW on the wall in the contact areas. From the manual and automatically experiments, we can see that the GW has hysteresis behaviour which is caused by stick-slip conditions that are related to dynamic and static friction. Some of the differences between the simulation and the experiments (see the first experiment in Figure 31) are caused by those forces and could easily resolve by adding also the lateral forces instead of using only normal forces. Now when we have the normal forces we can multiply t with the friction coefficient and simulate it more accurately.

Another thing which should be added into this simulation is the behaviour of the GW tip which is impotent to us especially to navigate the GW and to be able to manipulate the GW and choose the needed section to continue. We can use the
current method also to simulate the GW Tip by changing the spring coefficient of those segments and to use rotational movement along the tip base, the tip will have a constant deformation to give the rounded shape and we need to assume that the torsional movement from the base pass along the GW. This assumption is quite accurate and could be enough for this cause since the GW has a thin diameter and it is has a minimum friction with the body tissue. In order to add it to the simulation we should test it first and to use a new set up where the 3D movements could be analyzed using different equipment and creating new algorithm to detect the tip of the GW.

Regarding the mechanical system, we have seen that insertions and exertion movements are accurate and the tests were performed with high accuracy and repeatability (maximum to 1[mm] error) using the current stepper motor and mechanical parts. We have used the linear system in the 3 tests and each time we got repeatable results. However, the Torsional movements were not accurate enough and have suffered because of design issues of this part in the manipulator prototype. One of the reasons for those issues is freedom between the bearing and the shafts that caused the shafts moving during the rotation movements. In addition, the ABS parts were not stiff enough and caused the nut to clamp on the lead screw. We should use linear bearing to support better the upper tube during the linear movement. We can improve this part easily by re-manufacturing the upper tube holder. Since we showed in this thesis only GW movements without the tip, the current performances of only the linear system were sufficient to use. It is possible to fix those issues in case that further research will be needed.

From this research, we could see that there are many obstacles we should pass in order to have an autonomous GW navigation system. We should remember that this system need to replace a highly trained doctor therefore we should add forces sensors to the base of the GW to sense accurately the reaction forces during the threading process and to control the maximum force which is applied to translate the wire into the body. We should consider checking this model with real elastic tissues like blood vessels but in this case we will need again to use other imaging systems to see the wire inside the tissue. In general, the results of the simulation and the performances of mechanical system were good and satisfied this research need, we
believe that desired autonomous treatment will be part of our life in the near future but still there are many challenges that need to be resolved.
6 References


סימולציה והנעה של צנתר על תיל-מוליך

ליאור קבסה
סימולציה והנעה של צנתר על תיל-מוליך

חובר על מחקר

לשם תרומת חלקי של הדרישות לקבלת התואר מגיסטר למדעים בהנדסת מכוניות

ליאור קבשה

הוגש במסלול למענה – מכון טכנולוגי לישראל

תשמ"ג, חיפה, אוקטובר 2012
המחבר נועש בהנחיית:

פרופ' משה שטם, הפקולטה להנדסת מכונות, בטכניון.

דר' ניר שוולב, הפקולטה להנדסת מכונות, במרצה האוניברסיטת אריאל.
Techniques like the introduction of the catheter and the introduction of the guide wire have become common in recent years for the treatment of different heart diseases. Nevertheless, there are still risks associated with the catheter, in which damage may be caused due to the introduction of the catheter with too much force inside the patient.

We present a new technique of catheter introduction through an "external" robotic system and the use of simulation to track the catheter movement and the system model. In order to enable accurate and continuous movement of the catheter through the simulation, we study all the relevant factors to the system model, such as environmental conditions and mechanical properties of the catheter and the guide wire.

This study focuses primarily on the development of a physical model for the deformation of the catheter when it is introduced into the body, and the initial planning for the system's navigation of the catheter during medical treatment.

We present a way to use the catheter model with the robotic system to make the medical treatment automatic or semi-automatic. In the first chapter, we briefly summarize the medical background that led to the need for simulation and an automatic system, and it can be seen from the studies done in this field that one of the main causes of injuries during introduction is friction between the catheter and the inner lining of the body, therefore it is crucial to analyze the forces and find the shape of the catheter during the treatment in addition to the difficulties associated with the current technique of catheter introduction, which depends on monitoring the guide wire movement on a screen of a visualization device. This method also complicates the treatment and requires high skills and experience mainly in critical areas like the brain.

In the continuation of the chapter, we perform a literature study that looks at the evolution of the models and a review of some of the most popular methods for catheter deformation calculation. There are many models developed to give response mainly to training systems and practicing catheterization, as they answer different requirements, such as desired degrees of freedom, model accuracy, and computational time in each iteration.

The main models that we have chosen are: Cosserat, FEM, Dynamic Spline, Mass-spring and Rigid links. Among them, we chose the solid model because it is suitable for fast implementation with simple implementation and accuracy at the end point.

The solid model of the guide wire (over which the catheter is placed) presented in this thesis is a solid model of hard plates that move one after another and between each plate there is a spring to describe the elasticity of the guide wire in the form. In order to describe the interaction of the guide wire with the body, we created a software platform that can create external boundary conditions for the model to receive the deformation of the guide wire inside the body.

We assume that the guide wire is hard and does not change its length and we assume that during the simulation we move the guide wire in small movements, quasi-statically, to achieve the goal. In the second chapter, we devoted to the development of the physical model of the guide wire based on the model of Eshelby and King[1][2] of hard plates while finding the point of equilibrium where there is a minimum energy in the system. For this we modeled the relationship between the plates and the spring, and the points of contact with the wall...
We consider the potential energy of a coiled spring in its initial state. When the spring changes position from an initial state, the potential energy changes, and this change is what generates the force of the spring. We use these forces, which are functions of the initial conditions, and set them to zero to obtain finally a set of equations involving the initial conditions.

After using this model, we found some problems related to the stability of the solution and the exit of the mollicule from the model's boundary, since it was impossible to continue with the model as developed. Therefore, we developed a few modifications in the algorithm to improve the stability of the solution and control the velocity of the mollicule.

The main modifications we made were to divide the mollicule into several segments, calculate the forces acting on each segment, and use these forces to control the direction of motion of the mollicule until the solution stabilizes. In this way, we stabilized the motion of the mollicule and brought it to a continuous movement within the model's boundary.

In order to allow a large number of experiments to be performed, we developed a robot that can move the mollicule from the base and back. The robot is composed of two axes, one of which is driven by a DC motor and assisted by a system of gears and wheels that push the mollicule forward and backward, while the other is a motor and arrangement that causes the mollicule to lean from its base, allowing us to choose the direction of motion of the mollicule.

The robot is controlled by an expert two-axis control that receives commands from the computer according to the position of the mollicule. In order to close a real cycle on the mollicule, we needed a system of simulation that uses a simple camera to capture the mollicule's position, since we found a way to use a simple camera and image processing to obtain the position of the mollicule in the space.

Additional details about the design of the robot and architecture for an autonomous system are described in detail in Chapter 3. We present the methods used and the system in which we performed the experiments in Chapter 4, where we present the system we built to test the algorithms.

For each experiment selected, we built an algorithm that allows the movement of the robot in the model for different paths. Finally, we present the results of three experiments in Chapter 5, and for each experiment, we present images that we have captured while moving the mollicule in the model, along with the results of the initial simulation and the improvement of the algorithms.

We present various measures for checking the accuracy of the simulation such as: number of cells, root mean square (RMS), and computational efficiency (DSC) and the accuracy of the model compared to the initial simulation shows improvement.

We can see that for a similar number of cells, the problems of stability found in the initial model were solved, and in order to achieve the same results, we used the method we developed.
החדשה עם ה改善ים, עלינו להכפיל את כמות החוליות במודל הראשוני כדי להשיג את הגמישות הרצויה. בדרך זו אנו מגדילים את זמן החישוב כך שהשיטה הראשונית לא תוכל לתפקד בזמן אמת ולהתאונה למערכת.

ניתן לראות שיפור מתבטא בעיקר ביכולות הכפיפה של המודל המשופר אל מול השיטה הישנה, שיפור זה יernes את תקופת זמן ביצועים בשפה השיפור הגודל יותר.

בניסויי הדי-מוליך, בסימולציה של המודל הראשוני, היוצא אל מחוץ לדגם עם שגיאה של 31 מ"מ במודל החדש, לעומת 5 מ"מ במודל הדהיד. גם במדדים נוספים רואים שיפורים אך עדיין לא קיבלנו התאמה מלאה לבעיה לכן אנו ממליצים להוסיף את כוחות החיכוך לחישובים על מנת廖אר במדויק התהליך ויתר.

לבצע כל זה необходимо לחקור את התוך המדויק determin עולם ש-dismissible תאור רוב מודול ויתר.