# Evolutionary Algorithms for Medical Simulations – A Case Study in Minimally–Invasive Vascular Interventions

Peter. A.N. Bosman
Centre for Mathematics and Computer Science
P.O. Box 94079
1090 GB Amsterdam
The Netherlands
Peter.Bosman@cwi.nl

Tanja Alderliesten
Department of Radiology
The Netherlands Cancer Institute
Antoni van Leeuwenhoek Hospital
Plesmanlaan 121
1066 CX Amsterdam
The Netherlands
t.alderliesten@nki.nl

## ABSTRACT

To obtain the expertise to correctly perform minimally–invasive vascular interventions thorough training is required. Training using simulation systems are increasingly becoming an accepted methodology. Recently, a minimally–invasive vascular intervention simulation (MIVIS) system has been developed. At the heart of this system lies an optimization problem to be solved repeatedly. In this paper, we investigate the advantages and disadvantages of using an evolutionary algorithm (EA) to solve the optimization problem instead of a problem–specific first–order analytical approximation algorithm. The results show that the use of the EA as optimization algorithm is favorable. A substantial reduction in time can be obtained while the RMS error associated with the simulation result differs only slightly.

## Categories and Subject Descriptors

G.1.6 [**Numerical Analysis**]: Optimization—*Gradient methods*; I.2 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search; I.6 [**Simulation and Modeling**]: Applications; J.3 [**Life and Medical Sciences**]: Health

## General Terms

Algorithms, Performance, Experimentation

## Keywords

Evolutionary Algorithms, Numerical Optimization, Gradients, Medical Simulation, Minimally Invasive, Vascular Intervention, Training, Guide wire

## 1. INTRODUCTION

In minimally–invasive vascular interventions instruments manipulated outside the body (e.g. guide wire and catheter) are navigated towards the vascular abnormality to be treated while visual feedback is provided through intra–operative imaging. Although compared to traditional open vascular surgery, minimally–invasive vascular interventions have some important advantages for patients, it is hard to master the required skills. Therefore, thorough training is needed. Simulation is becoming an accepted and established possibility for training.

Several groups have developed simulation systems to serve as a training possibility for minimally–invasive vascular interventions over the last years [1, 7, 9, 17]. One system, called the MIVIS (Minimally–Invasive Vascular Intervention Simulation) system [2, 3, 4], is of particular interest to us due to its clear modelling and definition in terms of an optimization problem to be solved. This system simulates the propagation of a guide wire inside the vasculature. The guide wire relaxes into a state of minimal energy after a given translation or rotation. Hence, in a practical implementation this means that the optimization problem to be solved can be defined as the minimization of the total energy. The energy is composed of an external component due to vessel geometry and an internal component related to the stiffness of the guide wire.

For a simulation system to be used in practice, two requirements should be met. The system should be sufficiently realistic and it should be real–time. Our goal in this paper is to compare the primarily developed problem–specific algorithm for the MIVIS system with an EA [8] and see whether the simulation system can benefit from the use of an EA with respect to the requirements. To this end we have performed experiments with an implementation of the problem–specific algorithm [12] and an EA for real–valued optimization, specifically a hybrid EDA [6].

## 2. OPTIMIZATION PROBLEM

The simulation is founded upon an optimization problem: a configuration of the guide wire should be found that corresponds to minimal total energy. The energy consists of two sources: an internal energy term is associated with the flexibility of the guide wire and an external energy term is associated with the deformation of the vasculature. The objective is to remain in a state of minimal energy.

The simulation is based on quasi–static mechanics, which means that the velocity and acceleration aspects associated with the propagation of the guide wire are ignored. This is a valid approximation, as much of human movement is performed at moderate speed, and is therefore essentially quasi–static [11].


*GECCO'05,* June 25–29, 2005, Washington, DC, USA.

## 2.1 Guide wire and vasculature modelling

The guide wire is modelled using a set of joint positions $(\boldsymbol{x}_0, \ldots, \boldsymbol{x}_{k-1})$, where joint 0 is the tip of the guide wire and joint $k-1$ the bottom. Between subsequent joints a straight, not bendable or compressible segment is defined with a predefined constant length $(\lambda_i)$. Guide–wire thickness is taken into account using sample points on the outer hull of the guide wire instead of using the joint positions when calculating the interaction with the vasculature (Figure 1).

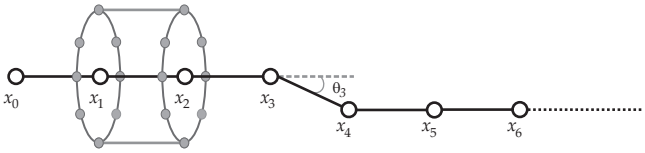


**Figure 1: Representation of a guide wire. Angle $(\theta_i)$ between the two segments connected by joint $i$ as used in Equation 1 is illustrated for joint 3. For joint 1 and 2 the sample points on the outer hull of the guide wire are drawn.**

### 2.1.1 Internal energy: bending of the guide wire

Hooke's law [5] is used as a basis for modelling the flexibility of the guide wire (internal bending energy) as well as for modelling the deformation of the vasculature (external vessel–wall energy). The bending energy $(U_{\mathrm{b}}(i))$ per joint $i$ as well as the total bending energy $(U_{\mathrm{Tb}})$ for the guide wire, are defined as:

$$U_{\mathrm{b}}(i) = \frac{1}{2} c_i \theta_i^2, \quad U_{\mathrm{Tb}} = \sum_{i=1}^{k-2} U_{\mathrm{b}}(i) \tag{1}$$

where $\theta_i$ denotes the angle between two segments connected by joint $i$ and $c_i$ is a spring constant related to the flexibility of the joint. The dependency between the flexibility of the guide wire and the segment length has been investigated previously [3, 12] and can be formulated as $c_i = \frac{EI}{\lambda_i}$. Here, $EI$ denotes the flexural rigidity of the guide wire, which is defined as the product of elasticity $(E)$ and the second moment of area $(I)$.

*Extension for guide wires with intrinsic curvature*

Not only do guide wires vary in thickness, flexibility and rigidity, they often also have intrinsically curved tips to allow for enhanced navigation (i.e. turning left or right). To allow for the modelling of such guide wires, two additional aspects are introduced into the model:

- An intrinsic flexion angle $\gamma_i$ indicating the strength of the intrinsic curvature for joint $i$ (the segment between joints $i$ and $i-1$) relative to the segment between joints $i$ and $i+1$.
- A unit vector $\hat{\boldsymbol{e}}_i^{(\gamma)}$ indicating the direction of intrinsic curvature at joint $i$.

The calculation of the angle $\theta_i$ between two adjacent segments now needs to be adapted. The former segment can be used as reference segment for completely straight guide wires, but for guide wires with intrinsic curvature this reference segment needs to be adapted according to the intrinsic curvature as defined by $\gamma_i$ and $\hat{\boldsymbol{e}}_i^{(\gamma)}$. Specific details are provided in the Appendix (Section 7).

### 2.1.2 External energy: deformation of the vessel wall

The external vessel–wall energy per joint $(U_{\mathrm{vw}}(i))$ and the external vessel–wall energy for the complete guide wire $(U_{\mathrm{Tvw}})$ are given by:

$$U_{\mathrm{vw}}(i) = \frac{1}{2} l d_i^2, \quad U_{\mathrm{Tvw}} = \sum_{i=0}^{k-1} U_{\mathrm{vw}}(i) \tag{2}$$

The vessel–wall elasticity constant is denoted by $l$ and the vessel–wall deformation by $d_i$. The total energy now to be minimized is just $U_{\mathrm{Tb}} + U_{\mathrm{Tvw}}$.

## 2.2 Processing of actions: performing optimization

A continuous propagation action is discretized into a sequence of small, forced motions. To simulate the outcome of each small forced translation of size $\xi_{\mathrm{internal}}$, first each joint position is displaced in the propagation direction by a length of $\xi_{\mathrm{internal}}$, i.e. as if no vasculature was present. Then, the influence of the vasculature is taken into account by minimizing $U_{\mathrm{Tb}} + U_{\mathrm{Tvw}}$ as a function of joint positions $\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_{k-3}$. The bottom two joint positions are fixed as they define the propagation direction.

# 3. OPTIMIZATION METHODS

The problem dimensionality can be reduced using a 2D parameterization (instead of the 3D joint positions). Such a parameterization is possible because by construction the distance between two neighbouring joints always remains constant and hence each subsequent joint must lie on the surface of a sphere. The actual mapping can be constructed in various ways (see for instance [16]).

It is important to realize that the goal is not to globally minimize the value of $U_{\mathrm{Tb}} + U_{\mathrm{Tvw}}$. The reason for this is that the current guide–wire position depends on the history of propagation; the guide wire cannot simply switch between vascular branches. For this reason, the problem–specific optimization algorithm is a local–search method, which seems a reasonable approach, especially if $\xi_{\mathrm{internal}}$ is not too large and the nearest local optimum is the one that would have been obtained if a continuous motion had been used.

## 3.1 First–order analytical solution

An explicit analytical solution in the first–order approximation has been derived to minimize the energy per joint [12] using a problem–specific 2D parameterization. An algorithm has been formulated that applies this analytical solution to each joint sequentially, starting at the tip and moving towards the bottom of the guide wire. As a result of displacing joint $i$ however, all joints $j \in \{i-1, i-2, \ldots, 0\}$ must also be displaced to preserve the segment lengths, possibly moving them away from their optimal position. Therefore, to ensure convergence, this process is repeated several times. These repetitions are called global iterations. In the derivation of the first–order solution, a number of assumptions and approximations have been made which restrict the combinations of values that can be chosen for the different parameters (e.g. segment length and internal stepsize). Moreover, the many involved computations that are required entail a risk of numerical instability.

## 3.2 Evolutionary algorithm

Evolutionary algorithms (EAs) are generally applicable optimizers. The optimization problem at hand is a real–valued one and may grow to quite a large dimensionality as the guide wire is advanced further into the vasculature. An EA that has recently been shown to be a good generally–applicable global optimizer for problems of various dimensionalities is the Iterated Density–Estimation Evolutionary Algorithm (IDEA) [6]. IDEA actually represents a similar type of algorithms indicated by the more commonly used acronym of EDA (Estimation–of–Distribution Algorithm). The main difference with traditional EAs is that in EDAs a probabilistic model is learned using the selected solutions. The probabilistic model can capture various structural properties of the optimization problem at hand. By drawing new solutions from the probabilistic model these structural properties can be exploited to obtain more efficient optimization. Especially if the probabilistic model is a low–complexity, highly generalizing probability distribution such as the normal distribution, EDAs tend to be good global optimizers. To speed up convergence it has been shown that the hybridization of such algorithms with local optimizers can be beneficial. For instance the Gradient–Leveraged IDEA (GLIDE–EA) in which the conjugate–gradients algorithm [15, 10] is applied to a small percentage of solutions in the population has been shown to give superior results. In this paper we performed experiments with GLIDE based on the normal distribution in which each variable is taken to be independent of all the other variables. This means that for each variable the mean and standard deviation of a one–dimensional normal distribution is estimated from the selected solutions and that a new solution is constructed by sampling one value per variable from the associated one–dimensional normal distribution. This approach was previously observed to give good results [6]. The specific EA used in this paper can be written in pseudo–code as follows:

GLIDE-EA-RELAX

1 $l \leftarrow 2k$
  → Genotype encodes 2 parameters per joint position
2 $\boldsymbol{\mathcal{P}} \leftarrow$ **new vector** of length $n$
3 **for** $i \leftarrow 0$ **to** $|\boldsymbol{\mathcal{P}}| - 1$ **do**
  3.1 $\boldsymbol{\mathcal{P}}_i \leftarrow$ random genotypes of length $l$
  3.2 Compute energy $U_{\mathrm{Tb}} + U_{\mathrm{Tvw}}$ for $\boldsymbol{\mathcal{P}}_i$ (after decoding genotype information in $\boldsymbol{\mathcal{P}}_i$ into joint positions)
4 Iterate until termination
  4.1 $\boldsymbol{\mathcal{S}} \leftarrow \lfloor \tau n \rfloor$ solutions from $\boldsymbol{\mathcal{P}}$ with lowest energy
  4.2 **for** $i \leftarrow 0$ **to** $|\boldsymbol{\mathcal{S}}| - 1$ **do**
    4.2.1 $\mu_i \leftarrow \frac{1}{|\boldsymbol{\mathcal{S}}|} \sum_{j=0}^{|\boldsymbol{\mathcal{S}}|-1} (\boldsymbol{\mathcal{S}}_i)_j$
    4.2.2 $\sigma_i^2 \leftarrow \frac{1}{|\boldsymbol{\mathcal{S}}|} \sum_{j=0}^{|\boldsymbol{\mathcal{S}}|-1} ((\boldsymbol{\mathcal{S}}_i)_j - \mu_i)^2$
  4.3 $\boldsymbol{\mathcal{O}} \leftarrow$ **new vector** of length $n - |\boldsymbol{\mathcal{S}}|$
  4.4 **for** $i \leftarrow 0$ **to** $|\boldsymbol{\mathcal{O}}| - 1$ **do**
    4.4.1 **for** $j \leftarrow 0$ **to** $l - 1$ **do**
      4.4.1.1 $(\mathcal{O}_i)_j \leftarrow$ random sample drawn from $\mathcal{N}(\mu_i, \sigma_i^2)$
  4.5 **for** $i \leftarrow 0$ **to** $|\boldsymbol{\mathcal{O}}| - 1$ **do**
    4.5.1 $\boldsymbol{\mathcal{P}}_{i+|\boldsymbol{\mathcal{S}}|} \leftarrow \boldsymbol{\mathcal{O}}_i$
    4.5.2 Compute energy $U_{\mathrm{Tb}} + U_{\mathrm{Tvw}}$ for $\boldsymbol{\mathcal{P}}_{i+|\boldsymbol{\mathcal{S}}|}$
  4.6 Apply conjugate gradients search to $\lfloor \tau_{\mathrm{G}} n \rfloor$ randomly chosen solutions in $\boldsymbol{\mathcal{P}}$ (and update energy accordingly)

Since EAs are by construction capable of escaping local optima to a certain extent, additional precautions need to be taken to prevent physically incorrect behavior of the simulation when using the EA for optimization. Therefore, we have added a penalty term to the total energy to be minimized. The penalty term increases exponentially if the guide wire is displaced by more than a predefined distance that is based on the vessel diameter.

# 4. EXPERIMENTS AND RESULTS

The accuracy of the simulation has been determined by comparing physical experiments with simulated outcomes. Two phantoms, manufactured by Shelley Medical Imaging Technologies, Ontario, Canada, have been used: a carotid anthropomorphic vascular phantom and an elastrat normal abdominal aorta phantom. Shelleys Carotid Anthropomorphic Vascular Phantoms are designed to very accurately mimic complex physiological vascular geometries and are made of acrylic. The elastrat normal abdominal aorta phantom is derived from cadaveric specimens and made of silicone. The vessels were extracted from a CT data set of these phantoms by means of voxel–based segmentation. The distance of a joint to the vessel wall as required in Equation 2 is thus only available in a discretized fashion. This may pose a problem for algorithms based on continuous gradients such as conjugate gradients. The first–order analytical approximation (FOA) algorithm internally also works with gradient information. Therefore, this algorithm is also hampered. Preliminary experiments have shown however that using trilinear interpolation provides a sufficiently smooth representation for these algorithms to work.

Simulation results are evaluated based on a root–mean–square (RMS) error measure. The RMS error is computed from distances between the joint positions in the simulated guide wire and corresponding reference points that have been extracted from physical experiments. For a guide wire with $k$ joints the RMS error measure is defined as:

$$\mathrm{RMS} = \sqrt{\frac{1}{k}\sum_{i=0}^{k-1}(d_i^{\mathrm{SR}})^2}, \quad d_i^{\mathrm{SR}} = \|\boldsymbol{x}_i^{\mathrm{S}} - \boldsymbol{x}_i^{\mathrm{R}}\| \qquad (3)$$

where $\boldsymbol{x}_i^{\mathrm{S}}$ denotes the simulated joint position and $\boldsymbol{x}_i^{\mathrm{R}}$ the corresponding reference point. All experiments were run on a 1.7 GHz Pentium with 512 Mb of RAM. Reported runtimes do not involve online visualization.

## 4.1 Abdominal–aorta phantom

### 4.1.1 First–order analytical approximation algorithm

Experiments with various values for the internal stepsize $\xi_{\mathrm{internal}}$ and for the segment length $\lambda_i$ have been performed, exploring the parameter space with respect to the constraints used to derive the analytical first–order approximation. For selected values for $\xi_{\mathrm{internal}}$ and $\lambda_i$ results are tabulated in Table 1. A global iteration number of 8 was used. For various parameter settings the first–order analytical approximation (FOA) algorithm supplies a guide– wire configuration with an RMS error around 1 $mm$. The use of segment lengths larger than 3 $mm$ resulted in erroneous results.

The best result is achieved for a segment length of 3 $mm$, combined with an internal stepsize of 0.3 $mm$. The required time to obtain this result was 92.040 seconds. This can be reduced to 20.359 seconds by using only 2 global iterations. However, the RMS error then increases to 1.232 $mm$. This simulation result is illustrated in Figure 3.

#### 4.1.2 GLIDE

The maximum allowed number of evaluations was set to $c_{\text{eval}}k$ where $k$ is the number of joints that the guide wire currently consists of. If convergence occurred earlier, termination was enforced also. For selection we set $\tau$ to 0.3, conforming to the rule–of–thumb for EDAs by Mühlenbein and Mahnig [14]. Furthermore, we fixed $\tau_{\text{G}}$ to ...

$\mathcal{T}$, $\mathcal{T}_{\text{G}}$

Since the EA uses a lot less problem–specific information, the expected room for improvement lies in the use of settings where the FO$\mathbb{A}$ algorithm failed. Indeed, using the same settings as for the FO$\mathbb{A}$ algorithm the EA required more time, although similar RMS errors were found. Hence, we performed experiments with more coarse–grained settings.

First, we performed experiments in which we varied the population size and the value of $c_{\text{eval}}$. However, this time we only performed experiments with a much larger internal stepsize. As $c_{\text{eval}}k$ increases, the algorithm is allowed more precision and more time respectively to find a local optimum. It is to be expected that precision comes at the cost of time. The results as shown in Figure 2 indeed clearly indicate a trade–off between the two ultimate objectives of minimizing the required time and minimizing the RMS error. Based on these experiments we experimented further with population sizes of 20 and 50 and $c_{\text{eval}} \in \{50, 10000\}$ because these population sizes were prominently present on the Pareto front. The results of these further experiments are tabulated in Table 1. The results show that an RMS error of 1.118 $mm$ can be obtained in 2.984 seconds ($\lambda = 3, \xi_{\text{int}} = 14$). Furthermore, for a variety of parameter settings the runtime is reduced to less than 1 second while the RMS error increases to around 2 and 3 $mm$.

From the tabulated results we can conclude that GLIDE is capable of quick and reliable optimization. Lower RMS errors can be found if $c_{\text{eval}}$ is set large enough. However, this greatly comes at the expense of time. Since the final objective is to build a real–time simulator, these settings are not preferable. However, by choosing for instance a population size of 20 combined with a $c_{\text{eval}}$ of 50, the EA is still able to come up with good results. Moreover, the EA can still find good solutions for extreme cases. In these cases, only a few joints are required because the segment length is relatively large. Therefore, as the guide wire is propagated further, the scalability of the EA is much better as the optimization problem to solve contains less variables. This is illustrated in Figure 2 where required simulation time is plotted against the propagated length for different algorithms on a logarithmic scale. Although the asymptotic runtime complexity is the same for all algorithms, the constant different is extremely significant. An illustration of the best obtained result is supplied in Figure 3.

### 4.2 Carotid–bifurcation phantom

To test the performance of the algorithms in the case of guide wires with intrinsic curvature, the propagation over a distance of 102 mm of a guide wire with an intrinsically–curved tip in a Carotid–bifurcation phantom was also simulated. For both algorithms the simulation system behaved as desired when orienting the curved tip to turn either left or right at the bifurcation (see Figure 4).

The FO$\mathbb{A}$ algorithm was run with $\lambda = 3$mm, $\xi_{\text{int}} = 0.3$mm and a global iteration number of 2. The GLIDE algorithm was run with $\lambda = 3$mm, and $\xi_{\text{int}} = 14$mm. The RMS errors associated with the turning–left and turning–right simulations were 1.1 mm and 1.3 mm respectively for FO$\mathbb{A}$ and 0.47 mm and 0.89 mm respectively for GLIDE. The associated runtimes were 5.7 and 5.6 seconds respectively for FO$\mathbb{A}$ and 4.3 and 4.5 seconds respectively for GLIDE. The main reason for the smaller difference in runtime for this phantom is the smaller amount of interaction of the guide wire with the vessel wall. The FO$\mathbb{A}$ algorithm performs less computations if such interaction is not present, whereas the GLIDE algorithm does not use such problem–specific information.

## 5. DISCUSSION

Simulations are an approximation of reality. We have seen that when comparing simulated results with results from reality, RMS errors that are significantly smaller than 1 $mm$ could not be obtained by both optimization techniques. Most likely the reason for this is thus that the actual minimum of the optimization problem does not correspond to the minimum of physical reality, caused by the simplified model of energy and kinetics. Therefore, an RMS error around 1 $mm$ is likely to be close to the optimally obtainable result using this model.

The experiments show that both optimization algorithms can provide results with which a comparable RMS error is associated. Nevertheless, the FO$\mathbb{A}$ algorithm turns out to be less appropriate to use in a real–time application. GLIDE still provides good solutions in the case of large segment lengths. This allows for the advantage of keeping the runtime small when the guide wire is propagated over a larger distance as a result of improved scalability.

The deformation of the vessel wall $d_i$ in Equation 2 is computed using the distance transform from the segmented vasculature. The problem–specific optimization algorithm uses a vectorial image in which the vessel–wall energy gradient is stored. GLIDE works directly by evaluating Equation 2. The distance transform can be refined by preprocessing methods. Preliminary experiments have shown that the problem–specific optimization algorithm does not benefit from this refinement as it is based on the gradient of the distance transform. However, GLIDE does benefit from such a refinement. The refinement can be obtained by preprocessing methods, therefore no additional running time is associated with the use of this refinement.

## 6. CONCLUSIONS

We have investigated the applicability of different optimization algorithms in the MIVIS system; a system developed for the simulation of minimally–invasive vascular interventions. We compared the use of a problem–specific optimization algorithm with the use of an an EA, specifically the GLIDE algorithm.

The results show that GLIDE is favorable. Compared to the use of the first–order analytical approximation algorithm a substantial reduction in simulation time is obtained while the RMS error associated with the result remains acceptable. Hence, the use of EAs in medical simulation systems can bring these systems significant step closer to real–time without significant loss of accuracy.

## 7. APPENDIX

For any configuration of the guide wire the two angles $\gamma_i$ and $\delta_i$ tell us the position of minimal bending energy for

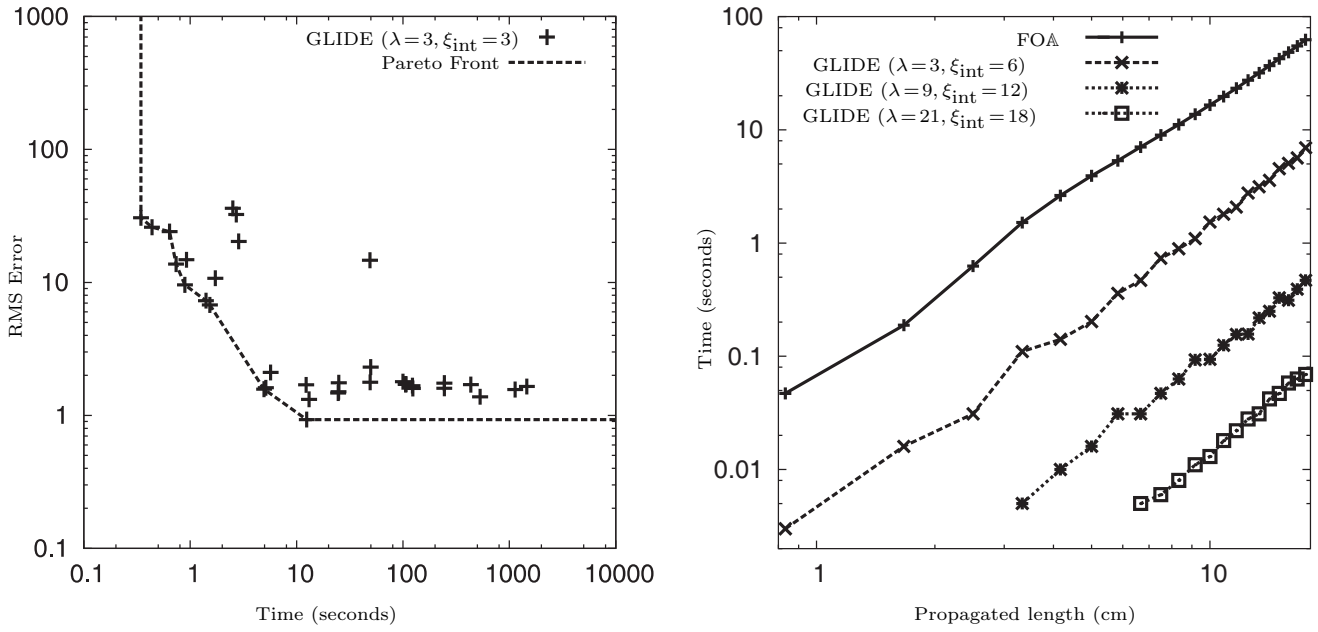


**Figure 2:** ***(Left)*** **GLIDE results in a scatterplot for different simulation settings.** ***(Right)*** **Scale-up of algorithms with propagated guide–wire length. Both $\lambda$ and $\xi_{\text{int}}$ are in mm.**

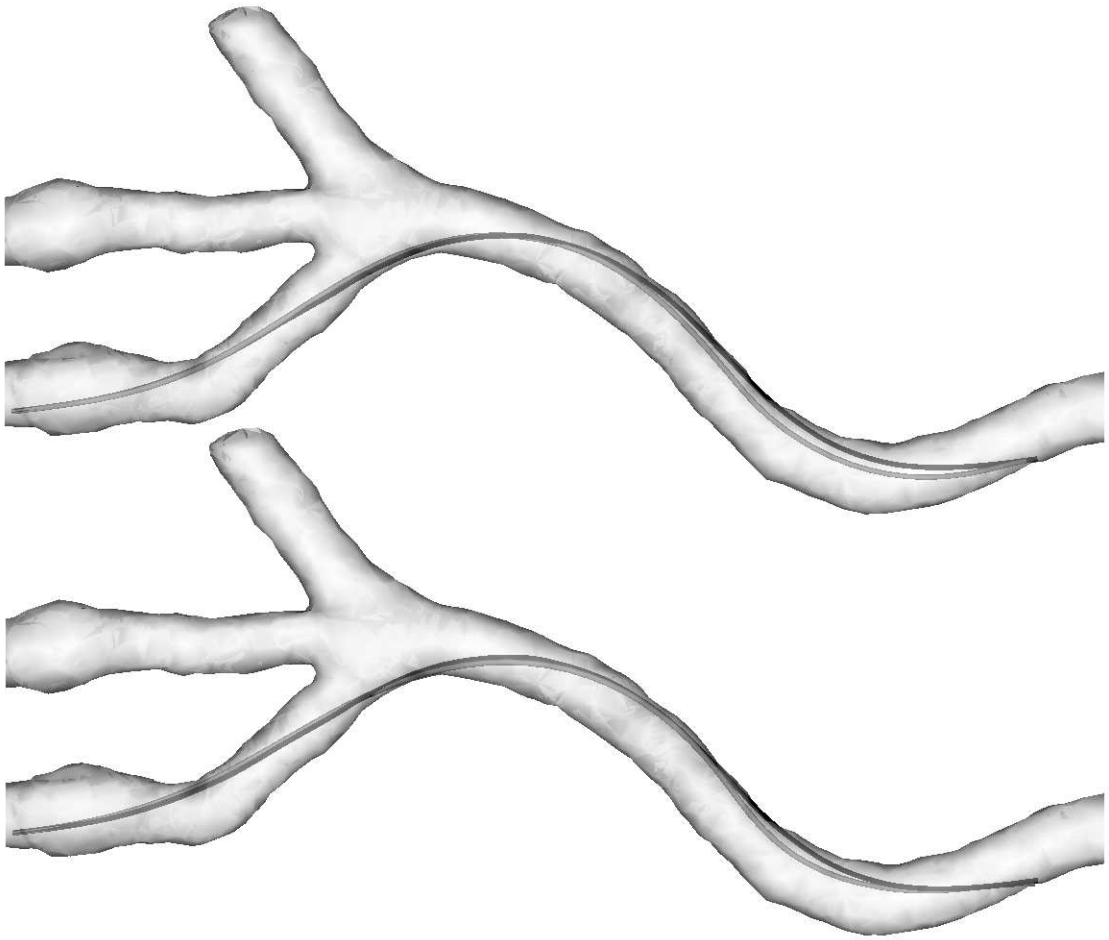

**Figure 3: Illustration of the reference guide wire configuration (black) and the simulated guide wire configuration (grey) using the FOA algorithm (top) and GLIDE (bottom). Settings for GLIDE: population size $= 20$, $c_{\text{eval}} = 50$, $\lambda_i = 3\ mm$ and $\xi_{\text{int}} = 14\ mm$.**

| $\lambda_i$ | $\xi_{\text{int}}$ | FOA | |
|---|---|---|---|
| | | RMS | Time |
| 1 | 0.1 | 0.996 | |
| 1 | 0.05 | 1.217 | |
| 1 | 0.025 | 1.141 | |
| 1 | 0.0125 | 1.276 | |
| 1 | 0.00625 | 1.215 | |
| 2 | 0.2 | 1.066 | |
| 2 | 0.1 | 0.995 | |
| 2 | 0.05 | 1.117 | |
| 2 | 0.025 | 1.120 | |
| 2 | 0.0125 | 1.130 | |
| 3 | 0.3 | 0.983 | |
| 3 | 0.15 | 1.075 | |
| 3 | 0.075 | 1.097 | |
| 3 | 0.0375 | 1.140 | |
| 3 | 0.01875 | 1.151 | |

| $\lambda_i$ | $\xi_{\text{int}}$ | GLIDE $_{20\text{–}50}$ | | GLIDE $_{20\text{–}10000}$ | | GLIDE $_{50\text{–}50}$ | | GLIDE $_{50\text{–}10000}$ | |
|---|---|---|---|---|---|---|---|---|---|
| | | RMS | Time | RMS | Time | RMS | Time | RMS | Time |
| 3 | 3 | 1.697 | 12.344 | 1.715 | 106.328 | 0.930 | 12.360 | 1.381 | 531.140 |
| 3 | 6 | 1.289 | 6.281 | 1.678 | 78.562 | 1.619 | 6.313 | 1.652 | 256.360 |
| 3 | 8 | 1.808 | 5.000 | 1.618 | 81.704 | 4.043 | 5.016 | 1.666 | 270.438 |
| 3 | 12 | 1.701 | 3.469 | 2.416 | 6.344 | 1.273 | 3.500 | 1.628 | 126.187 |
| 3 | 14 | 1.118 | 2.984 | 1.787 | 55.328 | 4.753 | 3.015 | 1.395 | 179.578 |
| 3 | 21 | 12.424 | 2.172 | 14.360 | 91.187 | 14.308 | 2.187 | 12.991 | 273.234 |
| 9 | 3 | 1.894 | 1.344 | 1.970 | 12.688 | 1.552 | 1.375 | 2.037 | 70.907 |
| 9 | 6 | 2.892 | 0.688 | 2.275 | 8.765 | 2.987 | 0.703 | 2.036 | 35.313 |
| 9 | 8 | 1.923 | 0.547 | 2.433 | 6.547 | 2.496 | 0.546 | 2.424 | 36.329 |
| 9 | 12 | 2.773 | 0.391 | 1.654 | 37.031 | 2.707 | 0.391 | 2.417 | 21.328 |
| 9 | 14 | 2.656 | 0.328 | 2.397 | 3.672 | 4.055 | 0.344 | 2.235 | 19.875 |
| 9 | 21 | 6.602 | 0.235 | 2.458 | 3.156 | 6.648 | 0.250 | 2.344 | 12.438 |
| 18 | 3 | 2.333 | 0.344 | 3.375 | 6.953 | 3.464 | 0.359 | 3.383 | 25.703 |
| 18 | 6 | 123.961 | 0.172 | 3.265 | 3.406 | 38.017 | 0.187 | 3.049 | 14.297 |
| 18 | 8 | 2.855 | 0.140 | 3.013 | 2.594 | 2.911 | 0.140 | 3.037 | 12.265 |
| 18 | 12 | 2.987 | 0.110 | 3.329 | 1.969 | 5.541 | 0.110 | 3.039 | 6.625 |
| 18 | 14 | 2.855 | 0.094 | 2.996 | 1.281 | 4.049 | 0.094 | 3.018 | 6.234 |
| 18 | 21 | 3.432 | 0.063 | 3.070 | 1.141 | 2.986 | 0.078 | 3.085 | 5.797 |

**Table 1: Results of both the FOA algorithm (left) and of GLIDE (right). For different combinations of segment length ($\lambda_i$) and internal stepsize ($\xi_{\text{internal}}$) the RMS error ($mm$) is given. The index for GLIDE represents: population size–$c_{\text{eval}}$. Both the RMS error in $mm$ and the required time in seconds are supplied.**

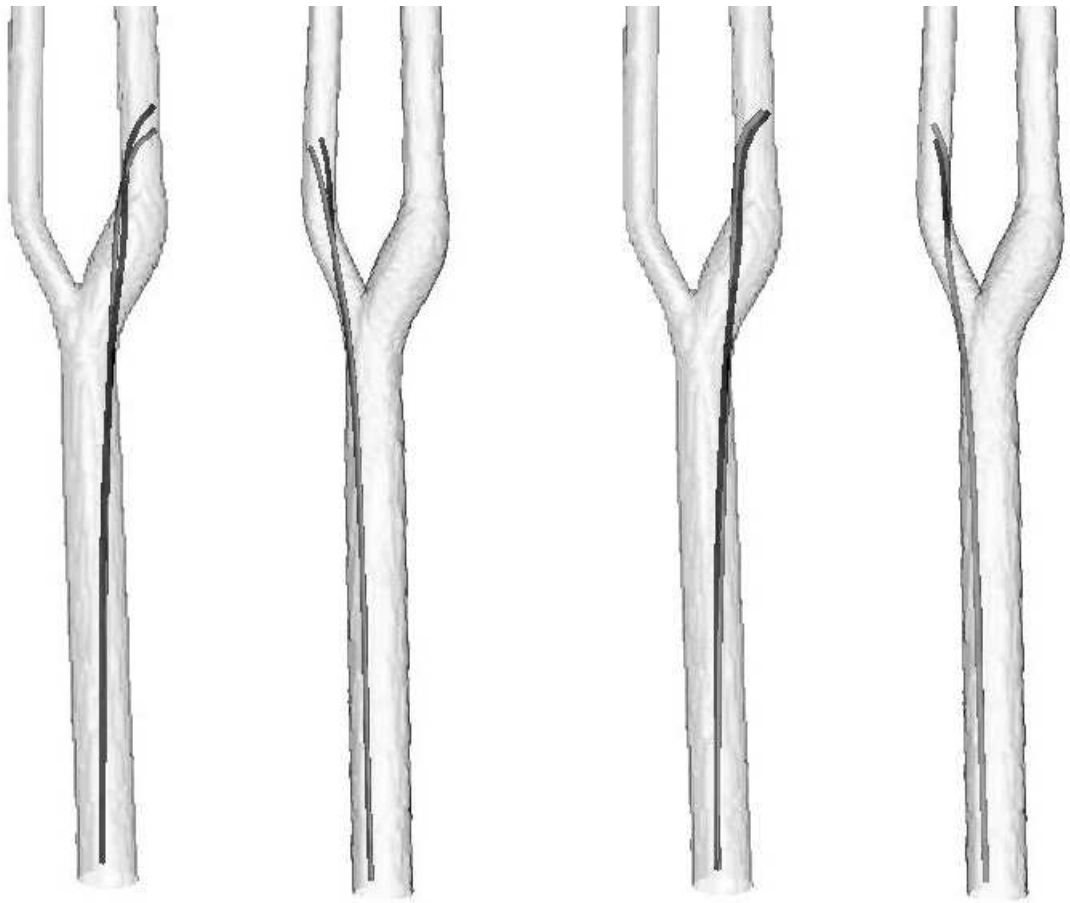

**Figure 4: Illustrations of two simulations of the propagation of an intrinsically–curved tip guide wire inside the carotid bifurcation phantom. The reference guide–wire configuration is rendered in black and the simulated guide–wire configuration in grey. The two visualizations on the left illustrate the simulation results obtained with the problem–specific optimization algorithm. The two visualizations on the right illustrate the simulation results obtained with GLIDE.**

the segment between joints $i-1$ and $i$. In the following all vectors are assumed to be unit vectors.

Initially $\delta_i$ is zero. Knowing the value of $\delta_i$ the vector $\hat{\boldsymbol{e}}_i^{(\gamma)}$ can be derived. The vector $\hat{\boldsymbol{e}}_i^{(\gamma)}$ is continuously defined in a straightforward manner. Vector $\boldsymbol{r}_i$ is the reference vector that needs to be updated:

$$\boldsymbol{r}_i = (\boldsymbol{r}_i^x, \boldsymbol{r}_i^y, \boldsymbol{r}_i^z) = \frac{1}{||\boldsymbol{x}_i - \boldsymbol{x}_{i+1}||}(\boldsymbol{x}_i - \boldsymbol{x}_{i+1}) \qquad (4)$$

We calculate the angle $\rho_i$ between the z–axis and $\boldsymbol{r}_i$. Subsequently the x–axis is rotated around the normal vector of the plane spanned by $\boldsymbol{z}$ and $\boldsymbol{r}_i$ with angle $\rho_i$. Since the dot product always returns the smallest angle between two vectors, a test is built in to determine whether $\rho_i$ or $2\pi - \rho_i$ should be used to obtain a vector perpendicular to $\boldsymbol{r}_i$: $\hat{\boldsymbol{e}}_i^{(\gamma)}$.

Summarizing, the mathematical definitions involved in the definition of $\hat{\boldsymbol{e}}_i^{(\gamma)}$ are as follows:

$$\begin{aligned}
\boldsymbol{r}_i &= \tfrac{(\boldsymbol{x}_i - \boldsymbol{x}_{i+1})}{||\boldsymbol{x}_i - \boldsymbol{x}_{i+1}||} \\
\rho_i &= \arccos(\boldsymbol{r}_i^z) \\
\zeta_i &= \begin{cases} \rho_i & \text{if } \frac{\sin(\rho_i)}{\sqrt{\boldsymbol{r}_i^{x\,2} + \boldsymbol{r}_i^{y\,2}}} = 1 \\ 2\pi - \rho_i & \text{otherwise} \end{cases}
\end{aligned}$$

$$\begin{aligned}
\hat{\boldsymbol{e}}_i^{(\gamma)} = \Big( & \tfrac{\boldsymbol{r}_i^{y\,2}}{\boldsymbol{r}_i^{x\,2} + \boldsymbol{r}_i^{y\,2}}(1 - \cos(\zeta_i)) + \cos(\zeta_i)\ , \\
& - \tfrac{\boldsymbol{r}_i^x \boldsymbol{r}_i^y}{\boldsymbol{r}_i^{x\,2} + \boldsymbol{r}_i^{y\,2}}(1 - \cos(\zeta_i)), \\
& - \tfrac{\boldsymbol{r}_i^x}{\sqrt{\boldsymbol{r}_i^{x\,2} + \boldsymbol{r}_i^{y\,2}}} \sin(\zeta_i) \Big)
\end{aligned}$$

Subsequently, $\hat{\boldsymbol{e}}_i^{(\gamma)}$ needs to be rotated with angle $\delta_i$ around $\boldsymbol{r}_i$. The reference vector $\boldsymbol{r}_i$ and $\hat{\boldsymbol{e}}_i^{(\gamma)}$ now define the plane in which the reference vector needs to be updated. To adapt the reference vector according to the intrinsic curvature, the reference vector needs to be rotated around the normal vector of the plane spanned by the reference vector and vector $\hat{\boldsymbol{e}}_i^{(\gamma)}$ with the angle $\gamma_i$. This results in an updated reference vector $\boldsymbol{r}_i^{\text{updated}}$, see Figure 5.

The rotation of a vector $\boldsymbol{v}$ around another vector $\boldsymbol{a}$ can be accomplished in the following way [13]:

$$\boldsymbol{v}_{\text{r}} = (1 - \cos(\iota))(\boldsymbol{a} \cdot \boldsymbol{v})\boldsymbol{a} + \cos(\iota)\boldsymbol{v} + \sin(\iota)(\boldsymbol{a} \times \boldsymbol{v}) \qquad (5)$$

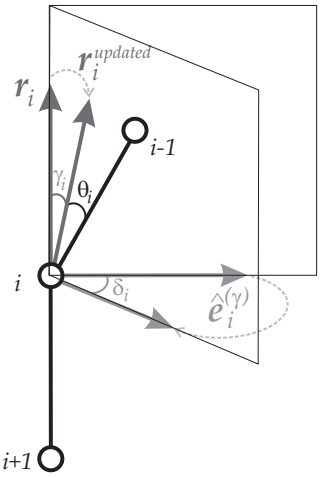


**Figure 5: An illustration of the update of $\hat{\boldsymbol{e}}_i^{(\gamma)}$ as well as the rotation of the reference vector $\boldsymbol{r}_i$. The angle to be used in the bending energy formula ($\theta_i$) is also shown.**

## 8. REFERENCES


[1] G. Abdoulaev, S. Cadeddu, G. Delussu, M. Donizelli, L. Formaggia, A. Giachetti, E. Gobbetti, A. Leone, C. Manzi, P. Pili, A. Scheinine, M. Tuveri, A. Varone, A. Veneziani, G. Zanetti, and A. Zorcolo. ViVa: The virtual vascular project. *IEEE Transactions on Information Technology in Biomedicine*, 22(4):268–274, 1998.

[2] T. Alderliesten, M.K. Konings, and W.J. Niessen. Simulation of guide wire propagation for minimally invasive vascular interventions. In T. Dohi and R. Kikinis, editors, *Medical Image Computing and Computer–Assisted Intervention – MICCAI 2002, Lecture Notes in Computer Science, vol. 2489, part II*, pages 245–252, Berlin, 2002. Springer–Verlag.

[3] T. Alderliesten, M.K. Konings, and W.J. Niessen. Simulation of minimally invasive vascular interventions for training purposes. *Computer Aided Surgery*, 9(1/2):3–15, 2004.

[4] Tanja Alderliesten. *Simulation of Minimally–Invavise Vascular Interventions for Training Purposes*. PhD thesis, Universiteit Utrecht, Utrecht, 2004.

[5] G. Arfken. *Mathematical methods for physicists*. Academic Press, Inc., San Diego, California, 1985.

[6] Peter A. N. Bosman. *Design and Application of Iterated Density–Estimation Evolutionary Algorithms*. PhD thesis, Universiteit Utrecht, Utrecht, 2003.

[7] S. L. Dawson, S. Cotin, D. Meglan, D. W. Shaffer, and M. A. Ferrell. Designing a computer–based simulator for interventional cardiology training. *Catheterization and Cardiovascular Interventions*, 51(4):522–527, 2000.

[8] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learing*. Addison Wesley, Reading, Massachusetts, 1989.

[9] J. K. Hahn, R. Kaufman, A. B. Winick, T. Carleton, Y. Park, R. Lindeman, K. M. Oh, N. Al-Ghreimil, R. J. Walsh, M. Loew, and S. Sankar. Training environment for inferior vena caval filter placement. In J.D. Westwood, H.M. Hoffman, S.J. Weghorst, and D. Stredney, editors, *Medicine Meets Virtual Reality – MMVR 1998, Studies in Health Technology and Informatics*, volume 50, pages 291–297, Amsterdam, 1998. IOS Press.

[10] M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. of Research of the National Bureau of Standards*, 6(49):409–436, 1952.

[11] N. Hogan and J.M. Winters. Principles underlying movement organization: upper limb and single joint. In J.M. Winters and S.L-Y. Woo, editors, *Multiple Muscle Systems: Biomechanics and Movement Organization*, pages 182–194. Springer–Verlag, New York, 1990.

[12] M.K. Konings, E.B. van de Kraats, T. Alderliesten, and W.J. Niessen. Analytical guide wire motion algorithm for simulation of endovascular interventions. *Medical & Biological Engineering & Computing*, 41:689–700, 2003.

[13] A. Mendoza. Tutorial on rotations about axes. Departamento de Matemáticas, Universidad Simón Bolívar, Venezuela. http://www.ma.usb.ve/ jacob/povgraph/ rot1.html.

[14] H. Mühlenbein and T. Mahnig. FDA – a scalable evolutionary algorithm for the optimization of additively decomposed functions. *Evolutionary Computation*, 7:353–376, 1999.

[15] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipies In C: The Art Of Scientific Computing*. Cambridge University Press, Cambridge, Massachusetts, 1992.

[16] F. Wattenberg. Spherical coordinates. Department of Mathematics, Montana State University, Bozeman, MT 59717. http://www.math.montana.edu/frankw/ccp/ multiworld/multipleIVP/spherical/body.htm, 1997.

[17] A. Zorcolo, M. Tuveri, G. Zanetti, and A. Zorcola. Catheter insertion simulation with co–registered direct volume rendering and haptic feedback. In J.D. Westwood, H.M. Hoffman, G.T. Mogel, A. Robb, and D. Stredney, editors, *Medicine Meets Virtual Reality – MMVR 2000, Health Technology and Informatics*, volume 70, pages 96–98, Amsterdam, 2000. IOS Press.