Software Requirements Specification for Optimal EM Placement: Convex Optimization for Optimal Positioning of Electromagnetic Actuators

Hussein Saad

April 12, 2025

Contents

Revision History

Date	Version	Notes
April 12, 2025	1.2	Implement instructor suggestions
April 11, 2025	1.1	Implement Domain Expert suggestions
February 8, 2025	1.0	First draft

1 Reference Material

This section records information for easy reference.

1.1 Table of Units

Throughout this document SI (Système International d'Unités) is employed as the unit system. In addition to the basic units, several derived units are used as described below. For each unit, the symbol is given followed by a description of the unit and the SI name.

symbol	unit	SI
m	length	metre
A	current	ampere
N	force	newton
T	magnetic flux density	tesla

1.2 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units. The choice of symbols was made to be consistent with existing literature around magnetic actuation systems.

symbol	unit	description
N	-	number of turns in a coil
i	A	current received by one coil
A	m^2	cross-sectional area of one coil
M	-	sample size
K	-	desired number of electromagnets
V	m^3	allocated under-the-table volume
t	m	target location

1.3 Abbreviations and Acronyms

symbol	description
A	Assumption
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
TM	Theoretical Model
EM	ElectroMagnet

1.4 Mathematical Notation

Vectors and matrices are bolded to fit the IEEE typesetting guide.

2 Introduction

Microrobots hold immense promise for a variety of clinical and surgical procedures. Their size and wireless controllability allow them to navigate complex and tortuous area of the human body with minimal collateral damage. However, clinical adoption of these devices faces the hurdle of by complexities in the actuation systems that control them (Bozuyuk et al., 2024). Magnetic actuation is perhaps the most common method of manipulating these devices, as magnetic fields can safely penetrate the human body and offer a high level of controllability. Magnetic fields can be generated through either EMs or permanent magnets, and while the latter offer stronger magnetic fields, EM coils are generally preferred because their magnetic field strength can be regulated by adjusting their power input (Hwang et al., 2020). However, EMs tend to be bulky and can potentially obstruct a clinician's access to the workspace. Even with recent designs that place the actuation system under the operating table, generating magnetic fields large enough to cover a patient-sized workspace remains a challenge.

We propose a convex optimization algorithm to solve the optimal EM arrangement problem for under-the-table actuation systems. Our program solves for the positions of EM actuators that yield the highest system manipulability.

This section explains the purpose of the document, the scope of requirements, the characteristics of the intended reader and the organization of the document.

2.1 Purpose of Document

The purpose of this document is to describe the mathematical methods used in solving the optimal EM arrangement problem. Thus, it is to be used as a reference to all information necessary for people who wish to verify or contribute to the software.

2.2 Scope of Requirements

The scope of the requirements is to present the mathematical models used in determining the optimal poses of electromagnetics in an under-the-table actuation system. The models used do not claim the optimality of any metrics other than positional and orientational metrics of EMs.

2.3 Characteristics of Intended Reader

The Intended Reader of this document must possess an understanding of magnetism at the first-year university level. In addition, a solid foundation in linear algebra at the freshman level is required. The reader should also have cursory knowledge of optimization basics, which are covered in crash-courses available online.

2.4 Organization of Document

The SRS follows a standard pattern of presenting background knowledge, goals, definitions, assumptions, theories, and instance models as proposed by (Smith and Lai, 2005; Smith et al., 2007; Smith and Koothoor, 2016). Readers who are familiar with the information presented in theoretical models may want to skip to the instance models section and refer back to earlier information as needed.

3 General System Description

This section provides general information about the system. It identifies the interfaces between the system and its environment, describes the user characteristics and lists the system constraints.

3.1 System Context

As the system context shows in the below figure, the user, represented by the circle, is responsible for both program inputs and outputs. The program itself is represented by the box, and the flow of information is indicated by the direction of the arrows.

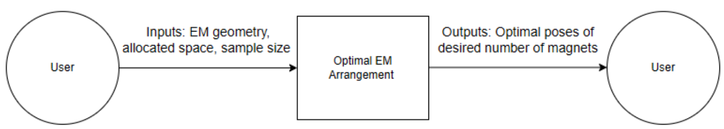


Figure 1: System Context

- User Responsibilities:
  - Provide the required input including EM geometry, under-the-table dimensions, sample size and desired number of EMs.
  - Ensure all inputs are valid.
- Optimal EM Placement Responsibilities:
  - Detect data type mismatch, such as a string of characters instead of a floating point number.
  - Determine the most optimal poses of the desired number of EMs within the under-the-table volume.
  - Ensure that the arrangement is physically feasible.

The program is typically used in two settings: engineering and scientific research. In engineering, it is primarily used during the planning phase and is not integrated into live systems. Thus, it is neither safety-critical nor mission-critical.

3.2 User Characteristics

The end user of the system is expected to be familiar with elementary magnetism concepts covered at the freshman science/engineering level.

3.3 System Constraints

This project has no system constraints.

4 Specific System Description

This section first presents the problem description, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the assumptions, theories, definitions and finally the instance models.

4.1 Problem Description

Optimal EM Placement is intended to solve the optimal magnet arrangement problem for under-the-table EM actuation systems.

4.1.1 Terminology and Definitions

This subsection provides a list of terms that are used in the subsequent sections and their meaning, with the purpose of reducing ambiguity and making it easier to correctly understand the requirements:

- Microrobot: A robot with dimensions less than 1mm.
- Pose: A position in 3D space together with an angular configuration.
- Workspace: The space encompassed by the magnetic fields generated by the EMs.
- Magnetic actuation: The use of magnetic fields to wirelessly manipulate an object.
- Solenoid: A type of EM formed by a helical coil of wire whose length is significantly greater than its diameter.

4.1.2 Physical System Description

The physical system of Optimal EM Placement includes the following elements:

PS1: An operating table.

PS2: A set amount of cubic volume available under the operating table.

PS3: Some number of EMs.

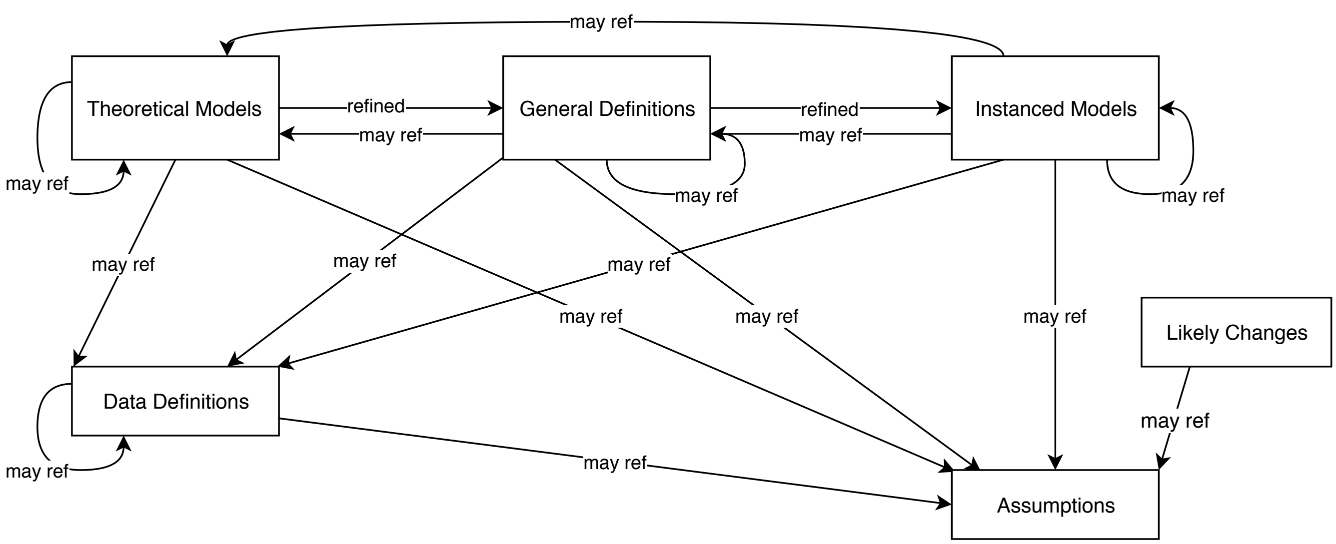
4.1.3 Goal Statements

Given the volume of the under-the-table space, the size of the EMs (number of turns, area), the current received by the system, a target location, the desired number of EMs, and sample size, the goal statements are:

GS1: Calculate the positions of the EMs that maximize the manipulability over the workspace.

GS2: Calculate the magnetic force and torque values at the target location.

4.2 Solution Characteristics Specification



The instance models that govern Optimal EM Placement are presented in Subsection 4.2.4. The information to understand the meaning of the instance models and their derivation is also presented, so that the instance models can be verified.

4.2.1 Assumptions

This section simplifies the original problem and helps in developing the theoretical model by filling in the missing information for the physical system. The numbers given in the square brackets refer to the theoretical model [TM], general definition [GD], data definition [DD], or instance model [IM], in which the respective assumption is used.

A1: All EMs are receiving the same, constant current [TM1, TM2].

A2: All EMs are identical in geometric properties [TM1, TM2].

A3: All EMs are perfectly cylindrical [TM1, TM2].

A4: All EMs follow the solenoid model [TM1].

4.2.2 Theoretical Models

This section focuses on the general equations and laws that Optimal EM Placement is based on.

Number	TM1
Label	**Magnetic Moment of a Solenoid**
Equation	$\boldsymbol{m} = NiA$
Description	The above equation calculates the magnetic moment $\boldsymbol{m}$ (A$\cdot m^2$) of a solenoid EM. N is the number of turns in the coil. i is the current received by the solenoid (A). A is the cross-sectional area of the solenoid (m^2).
Notes	-
Sources	https://en.wikipedia.org/wiki/Magnetic_moment
Ref. By	TM2

Number	TM2
Label	**Magnetic Field of a Magnetic Moment**
Equation	$\boldsymbol{B}(\boldsymbol{p}) = \frac{\mu_0 \|\|\boldsymbol{m}\|\|}{4\pi \|\|\boldsymbol{r}\|\|^3}(3\hat{\boldsymbol{r}}\hat{\boldsymbol{r}}^\top - \mathbb{I})\hat{\boldsymbol{m}}$
Description	The magnetic field $\boldsymbol{B}$ (T) at some point $\boldsymbol{p}$ can be calculated using the equation above. $\mu_0 = 4\pi \times 10^{-7}$ Tm/A is the permeability of free space. $\boldsymbol{r}$ is the position of the point relative to the center of the magnet (m). $\hat{\boldsymbol{r}}$ is the unit vector of $\boldsymbol{r}$. $\boldsymbol{m}$ is the magnetic moment of magnet generating the field $(\mathrm{A} \cdot m^2)$. $\mathbb{I}$ is the identity matrix.
Notes	-
Sources	https://en.wikipedia.org/wiki/Magnetic_moment
Ref. By	TM3, IM1

Number	TM3
Label	**Magnetic Force on a Magnetized Object**
Equation	$\boldsymbol{F} = \nabla(\boldsymbol{m} \cdot \boldsymbol{B})$
Description	The force $\boldsymbol{F}$ experienced by some magnetized object with magnetic moment m can be found as presented. ∇ is the field gradient. $\boldsymbol{m}$ is the magnetic moment of the object $(\mathrm{A} \cdot m^2)$. $\boldsymbol{B}$ is the magnetic field the object is in (T).
Notes	-
Sources	https://en.wikipedia.org/wiki/Magnetic_field
Ref. By	IM1

Number	TM4
Label	**Manipulator Jacobian**
Equation	$\boldsymbol{J}(x) = \begin{bmatrix} \boldsymbol{J}_v(x) \\ \boldsymbol{J}_w(x) \end{bmatrix}$
Description	The sensitivity of the outputs of a robotic system to changes in input x, can be described by a Jacobian $\boldsymbol{J}$. $\boldsymbol{J}_{v,w}$ are the Jacobians describing an output v, w respectively.
Notes	-
Sources	Spong et al. (2020).
Ref. By	IM1

Number	TM5
Label	**Robotic Manipulability**
Equation	$\mu \triangleq \prod_{i=1}^{n} \sigma_i \propto \sqrt{det(\boldsymbol{J}\boldsymbol{J}^\top)}$
Description	The manipulability μ of a robotic system is given by the product of singular values of its Jacobian. σ_i is the i-th singular value of the system. $\boldsymbol{J}$ is the Jacobian of the system with respect to some input.
Notes	-
Sources	Spong et al. (2020).
Ref. By	IM2

4.2.3 Data Definitions

This section collects and defines all the data needed to build the instance models. The dimension of each quantity is also given.

Number	DD1
Label	**Permeability of Free Space**
Symbol	μ_0
SI Units	$\mathrm{T\,m\,A^{-1}}$
Equation	$\mu_0 = 4\pi \times 10^{-7}\ \mathrm{T\,m\,A^{-1}}$
Description	Permeability is the measure of magnetization produced in a material in response to an applied magnetic field. Above the permeability of material in free space (vacuum).
Sources	https://en.wikipedia.org/wiki/Vacuum_permeability
Ref. By	TM2

4.2.4 Instance Models

This section transforms the problem defined in Section 4.1 into one which is expressed in mathematical terms. It uses concrete symbols defined in Section 4.2.3 to replace the abstract symbols in the models identified in Sections 4.2.2. The goal G1 is solved by IM2.

Number	IM1
Label	**Actuation Matrix**
Input	$\boldsymbol{B}$, $\boldsymbol{F}$
Output	$\begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{F} \end{bmatrix} \in \mathbb{R}^{6\times 1}$
Equation	$\mathcal{U} = \begin{bmatrix} \boldsymbol{B} \\ \boldsymbol{F} \end{bmatrix}$
Description	The actuation matrix $\mathcal{U}$ is an example of a Jacobian manipulator the describes the sensitivity of magnetic field and force values. $\boldsymbol{B}$ and $\boldsymbol{F}$ are the matrices containing magnetic field and force values.
Source	-
Ref. By	IM2

Number	IM2
Label	**Objective Function** $f(x)$
Input	$\mathcal{U}$, M, K
Output	$\boldsymbol{x} \in \{0,1\}^K$ s.t. $\mathbf{1}_M^\top \boldsymbol{x} = K$ and $f(x) = \sum_{i=1}^{K} \boldsymbol{x}_i \boldsymbol{U}_i \boldsymbol{U}_i^\top$ is maximized
Description	$\mathcal{U}$ is the actuation matrix. M and K are the sample size and desired number of EMs. $\boldsymbol{x}$ is the binary that indicates which EM positions were selected.
Sources	-
Ref. By	-

4.2.5 Input Data Constraints

Table 2 shows the data constraints on the input output variables. The column for physical constraints gives the physical limitations on the range of values that can be taken by the variable. The column for software constraints restricts the range of inputs to reasonable values. The software constraints will be helpful in the design stage for picking suitable algorithms. The constraints are conservative, to give the user of the model the flexibility to experiment with unusual situations. The column of typical values is intended to provide a feel for a common scenario. The uncertainty column provides an estimate of the confidence with which the physical quantities can be measured. This information would be part of the input if one were performing an uncertainty quantification exercise.

Table 2: Input Variables

Var	Physical Constraints	Software Constraints	Typical Value	Uncertainty
N	$N > 0$	$N > 0$	100	0%
I	$I > 0$	$0 < I \leq$ MAXCURR	10A	10%
A	$A > 0$	$A > 0$	6m^2	5%
M	$M > 0$	$M > 0$	10000	0%
K	$0 < K < M$	$K > 0$	8	0%
V	$V > 0$	$V > 0$	1m^3	0%
t	$t > 0$	$t > 0$	0.1m	10%

4.2.6 Properties of a Correct Solution

A correct solution must exhibit the properties stated in Table 4

Table 4: Output Variables

Var	Physical Constraints
$\boldsymbol{x}$	$\sum_{i=1}^{M} \boldsymbol{x}_i = K$

5 Requirements

This section provides the functional requirements, the tasks that the software is expected to complete, and the non-functional requirements, the qualities that the software is expected to possess.

5.1 Functional Requirements

R1: Load the inputs specific to EMs (number of turns, cross-sectional area, current), and actuation system (number of samples and actuators, volume) and a target location.

R2: Verify the given inputs satisfy the constraints in Table 2.

R3: Calculate the magnetic field and forces at the given distance t.

R4: Construct the actuation matrix $\mathcal{U}$.

R5: Compute the singular values of $\sum_{i=1}^{K} \boldsymbol{x}_i \mathcal{U}_i \mathcal{U}_i^\top$

R6: Output the $\boldsymbol{x}$ vector from IM2.

5.2 Nonfunctional Requirements

Due to the reasons stated in Section 3.1, there need not be an emphasis on time performance. However, due to the scientific nature of the program, some non-functional requirements must be met.

NFR1: **Accuracy** The minimum singular value returned by the program must be greater than that returned by the random and greedy solutions.

NFR2: **Usability** A user with the background described in Section 3.2 should understand the physical implications of the computations performed by the program e.g. EM positions, field and force values, etc., as described in Section 4.2.2 of the VnV Plan.

NFR3: **Maintainability** The effort required to make any of the likely changes listed for Optimal EM Placement should be less than $\frac{1}{10}$ of the original development time.

NFR4: **Portability** The program should run on any Windows, MacOS, or Linux machines.

5.3 Rationale

The program is designed for use in the preliminary stages of EM actuation system development. Typically, users will run it once or a few times before advancing to the physical development phase. EMs are assumed to be identical in properties to reduce the complexity of our formulation and implementation, as well as to remain consistent with the actuation system designs used in current literature. The cylindrical assumption is made since modeling the EM's shape in detail will over-complicate our equations but add little to the accuracy of our program. The solenoid model is chosen as it is a simple method to calculate all the information required for the solution.

6 Likely Changes

Given the scope of the project, there are no likely changes.

7 Unlikely Changes

LC1: Use a different model (dipole approximation, distributed current model, etc.) as opposed to a solenoid to model the EM [A4].

LC2: Permit different EMs in the same system [A1, A2].

8 Traceability Matrices and Graphs

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Every time a component is changed, the items in the column of that component that are marked with an "X" may have to be modified as well. Table 6 shows the dependencies of theoretical models, general definitions, data definitions, and instance models with each other. Table 7 shows the dependencies of instance models, requirements, and data constraints on each other. Table 8 shows the dependencies of theoretical models, general definitions, data definitions, instance models, and likely changes on the assumptions.

	TM1	TM2	TM3	TM4	TM5	DD1	IM1	IM2
TM1								
TM2								
TM3								
TM4							X	
TM5								X
DD1		X						
IM1								X
IM2								

Table 6: Traceability Matrix Between Items of Different Sections

	IM1	IM2
R1	X	X
R2		
R3	X	
R4	X	
R5	X	X
R6		X

Table 7: Traceability Matrix Between Requirements and Instance Models

	A1	A2	A3	A4
TM1			X	X
TM2				
TM3				
TM4				
TM5				
DD1				
IM1	X	X		
IM2	X	X		

Table 8: Traceability Matrix Between Assumptions and Other Items

9 Values of Auxiliary Constants

MAXCURR = 20000A.

References

Ugur Bozuyuk, Paul Wrede, Erdost Yildiz, and Metin Sitti. Roadmap for clinical translation of mobile microrobotics. *Advanced Materials*, page 2311462, 2024.

Junsun Hwang, Jin-young Kim, and Hongsoo Choi. A review of magnetic actuation systems and magnetically actuated guidewire-and catheter-based microrobots for vascular interventions. *Intelligent Service Robotics*, 13:1–14, 2020.

W. Spencer Smith and Nirmitha Koothoor. A document-driven method for certifying scientific computing software for use in nuclear safety analysis. *Nuclear Engineering and Technology*, 48(2):404–418, April 2016. ISSN 1738-5733. doi: http://dx.doi.org/10.1016/j.net.2015.11.008. URL http://www.sciencedirect.com/science/article/pii/S1738573315002582.

W. Spencer Smith and Lei Lai. A new requirements template for scientific computing. In J. Ralyté, P. Ȧgerfalk, and N. Kraiem, editors, *Proceedings of the First International Workshop on Situational Requirements Engineering Processes – Methods, Techniques and Tools to Support Situation-Specific Requirements Engineering Processes, SREP'05*, pages 107–121, Paris, France, 2005. In conjunction with 13th IEEE International Requirements Engineering Conference.

W. Spencer Smith, Lei Lai, and Ridha Khedri. Requirements analysis for engineering computation: A systematic approach for improving software reliability. *Reliable Computing, Special Issue on Reliable Engineering Computation*, 13(1):83–107, February 2007.

Mark W Spong, Seth Hutchinson, and M Vidyasagar. Robot modeling and control. *John Wiley*, 2020.