Stochastic Design and Analysis for Lattice-Based Self-Rearranging Robots

Cole Johnson

Abstract

In this write-up, we detail our method for computing the stochastic guarantees on the holding force of electro-permanent magnets (EPMs) in a self-rearranging robotic lattice system. We derive the reliability-based design optimization (RBDO) formulation, the lattice geometry constraints (using a triangular mesh decomposition), and the steps we took to numerically solve for the necessary force margins under manufacturing and operational uncertainties.

1 Introduction

Our self-rearranging robotic system relies on a network of nodes and edges with embedded electro-permanent magnets (EPMs). Each node contains power and actuation elements, and each edge can rotate, extend, and flex. The magnets must hold the system together robustly in the presence of uncertain manufacturing tolerances, environmental factors, and motor torque demands.

To guarantee that these magnets can reliably hold even under unknown loads, we developed a stochastic design-optimization framework. In particular, we imposed reliability constraints to ensure that the forces generated by the magnets meet or exceed the required holding forces with high probability.

2 Lattice Model: Composite Triangular Mesh

Consider a lattice comprised of N *nodes* and E *edges*, each edge having an EPM connection at both ends. To capture the load paths, we model the lattice as a triangular mesh, where each triangle is formed by three edges and three nodes. Define a set of triangles:

$$\mathcal{T} = \{ \triangle_1, \triangle_2, \ldots, \triangle_T \},$$

with T denoting the total number of triangles. Each $\triangle_i$ is a set of nodes (n_1, n_2, n_3) and edges connecting them.

We let $\mathbf{F}_i$ be the vector of internal forces in triangle i due to tension or compression in its edges. If $\mathbf{q} \in \mathbb{R}^{2N}$ (or $\mathbb{R}^{3N}$ in 3D) is the collection of node positions, then the force-balance equations at every node n_j take the form:

$$\sum_{i\,:\,n_j \in \triangle_i} \mathbf{F}_i(\mathbf{q}, \boldsymbol{\omega}) \;-\; \mathbf{W}_j(\boldsymbol{\omega}) \;=\; \mathbf{0}, \tag{1}$$

where $\mathbf{W}_j(\boldsymbol{\omega})$ is the (random) weight and external load at node n_j. These loads arise from battery mass, motor torque, payload, etc., all subject to uncertainty $\boldsymbol{\omega}$.

To remain connected and stable, each magnet in the lattice must provide a holding force sufficient to prevent any edge from failing under these nodal loads.

3 Stochastic Reliability Formulation

3.1 Design Variables

Let $\mathbf{x} \in \mathbb{R}^d$ be our *design vector* specifying magnet attributes:

$$\mathbf{x} = \begin{bmatrix} N \\ I \\ d_c \\ l_c \\ \vdots \end{bmatrix},$$

where N is the number of coil windings, I is the coil current (for magnetization/demagnetization), and d_c, l_c characterize coil geometry (diameter/length). Additional parameters (magnet material constants, enclosure dimensions) may also be included.

3.2 Random Variables

We define $\boldsymbol{\omega}$ as a vector of random parameters capturing uncertainties in:

$$\boldsymbol{\omega} = \left[\, M_n,\ M_e,\ T_m,\ f, \ldots \right],$$

where M_n, M_e are node/edge masses, T_m is motor torque or torque requirements, and f may be friction or alignment uncertainties. All of these distributions can be empirically derived or estimated from manufacturing data.

3.3 Magnet Force Model

Let $F_{\text{mag}}(\mathbf{x}, \boldsymbol{\omega})$ be the resulting holding force of a single magnet. A simplified form might be:

$$F_{\text{mag}}(\mathbf{x}, \boldsymbol{\omega}) \;=\; \alpha(\omega)\, \kappa\, N\, I\, \frac{A_c}{l_c^2} \;-\; \delta(\omega), \tag{2}$$

where κ is a constant involving material permeability, A_c is coil cross-sectional area, $\alpha(\omega)$ captures misalignment or partial contact effects, and $\delta(\omega)$ represents further random losses.

3.4 Required Lattice Holding Force

The required force $L_{\text{req}}(\boldsymbol{\omega})$ for an EPM (on any given edge) depends on the total load that edge must sustain in the triangular mesh. Denote by $L_{\text{req},i}(\boldsymbol{\omega})$ the force required to keep triangle i intact. This might be computed via:

$$L_{\text{req},i}(\boldsymbol{\omega}) \;=\; \varphi_i\Big(\mathbf{F}_i(\mathbf{q}, \boldsymbol{\omega})\Big), \tag{3}$$

where $\varphi_i(\cdot)$ is a scalar measure of tension or compression in edges associated with triangle i. To ensure *all* triangles remain stable, we require:

$$F_{\text{mag}}(\mathbf{x}, \boldsymbol{\omega}) \;\geq\; \max_{i \in \mathcal{T}} L_{\text{req},i}(\boldsymbol{\omega}) \;\;=\; L_{\text{req}}(\boldsymbol{\omega}). \tag{4}$$

In many cases, we take the worst-case triangle load from among the T triangles as the critical load for the magnet design.

3.5 Reliability Constraint

Our central reliability condition is:

$$P\Big[F_{\text{mag}}(\mathbf{x}, \boldsymbol{\omega}) \;\geq\; L_{\text{req}}(\boldsymbol{\omega})\Big] \;\;\geq\; 1 - \alpha, \tag{5}$$

where α is the acceptable probability of failure (e.g., $\alpha = 0.01$). Equivalently, we define a *limit-state function*:

$$G(\mathbf{x}, \boldsymbol{\omega}) \;=\; F_{\text{mag}}(\mathbf{x}, \boldsymbol{\omega}) \;-\; L_{\text{req}}(\boldsymbol{\omega}), \tag{6}$$

and require

$$P\big[G(\mathbf{x}, \boldsymbol{\omega}) \;\geq\; 0\big] \;\;\geq\; 1 - \alpha.$$

4 Deterministic Constraints and Objective

We also impose:

$$d_c \;\leq\; d_{\max}, \quad l_c \;\leq\; l_{\max}, \tag{7}$$

$$P_{\text{coil}}(N, I) \;=\; I^2\, R(N) \;\leq\; P_{\max}, \tag{8}$$

$$m_{\text{EPM}}(\mathbf{x}) \;\leq\; m_{\max}, \tag{9}$$

reflecting physical geometry, power limits, and mass constraints.

We then aim to minimize some cost function:

$$\min_{\mathbf{x}} \quad m_{\text{EPM}}(\mathbf{x}) \quad \text{or} \quad \min_{\mathbf{x}} \big[\beta_1\, m_{\text{EPM}}(\mathbf{x}) \;+\; \beta_2\, P_{\text{coil}}(N, I)\big], \tag{10}$$

subject to the reliability constraint in (5) and all deterministic constraints.

5 Solution Strategy

We solve the above as a *reliability-based design optimization* (RBDO) problem:

$$\begin{aligned} \min_{\mathbf{x}} \quad & m_{\text{EPM}}(\mathbf{x}) \\ \text{subject to} \quad & P\big[G(\mathbf{x}, \boldsymbol{\omega}) \geq 0\big] \;\;\geq\; 1 - \alpha, \\ & d_c \;\leq\; d_{\max}, \;\; l_c \;\leq\; l_{\max}, \;\; P_{\text{coil}}(N, I) \;\leq\; P_{\max}, \\ & \mathbf{x} \in \mathcal{X}, \end{aligned} \tag{11}$$

where $\mathcal{X}$ is our feasible design set.

5.1 Monte Carlo Sample-Average Approximation

A common approach is to approximate the probability constraint via sampling:

- Draw K samples $\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_K$ from the distribution of $\boldsymbol{\omega}$.
- Define $G(\mathbf{x}, \boldsymbol{\omega}_k)$ for $k = 1, \ldots, K$.

- Approximate the failure probability as

$$\hat{p}(\mathbf{x}) \;=\; \frac{1}{K}\sum_{k=1}^{K} \mathbf{1}\Big[G(\mathbf{x}, \boldsymbol{\omega}_k) \;<\; 0\Big].$$

- Impose $\hat{p}(\mathbf{x}) \leq \alpha$ in place of the exact probability.

We then iteratively adjust $\mathbf{x}$ to meet this approximate constraint while minimizing $m_{\text{EPM}}(\mathbf{x})$.

5.2 FORM, SORM, or Surrogate Methods

If $G(\cdot, \boldsymbol{\omega})$ is smooth and partially analytic, we can employ known reliability methods instead:

- **First-Order Reliability Method (FORM)**: Finds the most probable point (MPP) of failure and linearizes G around it.
- **Second-Order Reliability Method (SORM)**: Incorporates curvature in G.
- **Surrogate / Polynomial Chaos**: Builds an approximate model of G and performs reliability analysis on that surrogate.

6 Conclusion

By systematically incorporating the triangular lattice mechanics and uncertain parameters, we arrived at a *stochastic guarantee* on magnet holding force. The resulting design ensures:

$$P\big[F_{\text{mag}}(\mathbf{x}, \boldsymbol{\omega}) \;\geq\; \max_i L_{\text{req},i}(\boldsymbol{\omega})\big] \;\geq\; 1 - \alpha.$$

This condition provides high-confidence that the self-rearranging robotic structure will remain intact under real-world loads, friction, misalignments, and motor torques. At the same time, we minimize mass or power usage, enabling each node to remain as light and energy-efficient as possible.