2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

A Model for Optimising the Size of Climbing Robots for Navigating Truss Structures

Wesley Au*, Tomoki Sakaue and Dikai Liu

Abstract— **Truss structures can be found in many buildings and civil infrastructure such as bridges and towers. But as these architectures age, their maintenance is required to keep them structurally sound. A legged robotic solution capable of climbing these structures for maintenance is sought, but determining the size and shape of such a robot to maximise structure coverage is a challenging task. This paper proposes a model in which the size of a multi-legged robot is optimised for coverage in a truss structure. A detailed representation of a truss structure is presented, which forms the novel framework for constraint modelling. With this framework, the overall truss structure coverage is modelled, given a robot's size and its climbing performance constraints. This is set up as an optimisation problem, such that its solution represents the optimum size of the robot that satisfies all constraints. Three case studies of practical climbing applications are conducted to verify the model. By intuitive analysis of the model's output data, the results show that the model accurately applies these constraints in a variety of truss structures.**

I. INTRODUCTION

The size and shape of a robot is usually chosen based the task and characteristics of its intended environment, and actuator constraints to maximise its performance and efficiency. For a known robot topology, the size of the robot can be modelled based on the environment it is deployed in, and the navigational constraints it must satisfy. However, striking the right balance between the robot's size and meeting operational constraints such as structural coverage, locomotive speeds and actuator load limits is a difficult, multi-dimensional with no unique solution.

Empirical methods for determining the overall topology and optimum design for a robot are common practice in the absence of numerical optimisation. For naturalistic tasks such as walking or climbing, we can mechanically replicate the several aspects from the topology of living organisms. It has been shown that a robot may be better equipped to handle real-world environments and tasks, using this method [1]. The bio-inspired robotic design paradigm [2] has been applied to several small-scale climbing robots such as in [3] and [4]. However, unlike the evolution of larger organisms, robots do not scale well with increasing size due to technical limitations of current-day actuators [5].

Wesley Au wesley.au@uts.edu.au and Dikai Liu dikai.liu@uts.edu.au are with the Faculty of Engineering and IT, Centre for Autonomous Systems, University of Technology Sydney, Ultimo, New South Wales, Australia

Tomoki Sakaue sakaue.tomoki@tepco.co.jp is with Tokyo Electric Power Company Holdings, Inc. 4-1 Egasaki-cho, Tsurumi-ku, Yokohama, Japan

* Corresponding author

There are several methods that exist for optimising robot design and selection, which take into account the end effector's kinematics [6], [7], dynamics [8], and purpose [9]. However, these works only consider a static robot base and is unsuitable for a robot that may be in constant motion. Several robots were proposed specifically for climbing truss structures [10]–[14], and planes [15]–[20]. However, the optimisation of the design of these robots were never documented. Only the ROMA robot reports the use of the ADAMS simulation package, but all design parameters were determined based on torque constraints only.

It is evident that no generalised framework exists for the optimisation of these robots in truss-like structures when precise coverage constraints are required. Hence we propose a model for calculating overall truss structure coverage of a multi-limbed robot, when deployed in such a scenario. The proposed model focuses on detailed coverage constraints within a truss structure, which supplements typical kinematic and dynamic constraints. The model is implemented as a discrete optimisation problem, such that the solution represents the smallest robot possible that satisfies coverage, stability and gait constraints.

This paper will first present the setup of the model in Section II, then define the calculation of truss structure coverage, and all constraint equations in Section III. Section III-B shows how the model is used, and verifies the result of this model with optimised solutions on various truss structures, and Section V concludes this paper with a discussion of the derived model, its applications and future improvements.

II. METHODS OF VIRTUAL REPRESENTATION

The method for the virtual representation of the model dictates the ability to apply detailed coverage constraints in the optimisation problem. Three main components aside from the robot that make up the model require virtual representation: the truss structure, the environment and the interaction space. This space defines the area at which the robot can observe, touch or grasp. The choice of representing these components in a continuous or discretised space depends on the model's complexity, the computational efficiency and memory footprint when running the optimisation.

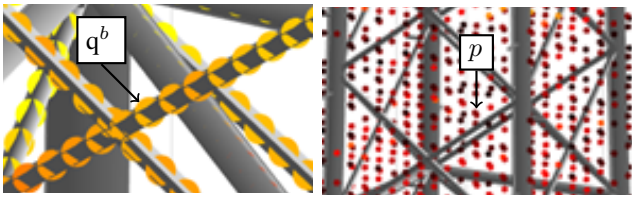
A discretised representation carries the advantage of model simplicity over a continuous model, hence our proposed model implements a discretised representation of the interaction space and environment. The truss structure is represented by parameterised vector functions. The aim is to model calculate truss structure coverage as a function of the robot's size, hence discretised interaction points are parented to the

truss structure. This has the same effect as discretising the entire truss structure as a series of interaction points. As each robot pose and interaction point is a unique and discrete point in the environment, the constraint modelling can be expressed in set notation. This is a logic-based approach to modelling, in which constraint equations can be simplified to logic statements based on inequalities. This allows for the application of highly detailed coverage constraints such as coverage redundancy, limb-specific coverage and grasping constraints. Finally, truss structure coverage is a simple calculation of the number interaction points that are reachable by the robot when constraints are applied. The model utilises this calculation to form the discretised optimisation problem.

A. Truss Structure and Environment Representation

The physical modelling of the truss structure in its environment is expressed in the discretised space in three sets:

- B, a set of trusses that make up the structure,
- Q, a set of discrete *interaction* points on each beam in the structure (Fig. 1a), and
- T, a set of discrete points in the environment (Fig. 1b).



(a) Interaction points q^b (b) Robot poses p

Fig. 1: The interaction space Q and environment space T of a truss structure B.

This representation allows constraints can be applied to individual beams, grasp points on the beams and position around the truss structure within the environment.

1) Parameterised truss structure: The truss structure is made up of individual but connected trusses. We define this structure as set B,

$$B = \{\text{all trusses in the environment} : b \in B \text{ is a data structure}\}, \quad (1)$$

where each element b is a data structure representing a parameterised vector with a known diameter or profile that makes up a truss.

2) Discretised interaction space: The interaction space is the set of all points that lie on beams $q \in B$ in which the robot can *interact* with, such as observing, touching or grasping. This set is defined as Q,

$$Q = \{q^b : q \in \mathbb{R}^3 \text{ represents a discrete point on an object } b \in B\}, \quad (2)$$

where each element represents an ordered pair: $q^b = (q, b)$. Point $q \in \mathbb{R}^3$ represents a point in 3D space, constrained to lie on a beam $b \in B$, that indicates that part of a beam can be interacted with. Therefore by definition, a beam without any $q \in Q$ indicates that the beam is not accessible by a robot. Fig. 1a shows an example of the interaction space Q on a truss structure B, where each dot represents a valid interaction points q^b on a beam b. Thicker beams shown in the figure are inaccessible due to the absence of interaction points attached to them.

3) Discretised environment space: The environment space defines all possible robot body poses in the environment enveloping the truss structure. This is defined as set T,

$$T = \{p \in SE(3) : g_t(p, B)\}, \quad (3)$$

where g_t provides the discretisation scheme for pose p, describing the position and orientation of the robot's centre of mass. Fig. 1b shows a possible layout of the environment space T, where each dot represents a possible body pose of the robot p, constrained to a grid set by g_t. Body poses that are too far from the truss structure are removed.

B. Robot

This work assumes a multi-legged climbing robot is kinematically capable of navigating a truss structure and to carry out maintenance work. Inchworm-style robots [21] [22] are capable of climbing similar structures, but lack the capability of carrying maintenance tools due to the absence of a stable robot body during locomotion. The gait stability [23] observed by four-limbed organisms in nature [24], [25], forms the basis of a bio-inspired four-limbed robotic solution for the navigation of truss structures. Hence any further reference to multi-limbed robots is assumed to be a robot of at least four limbs.

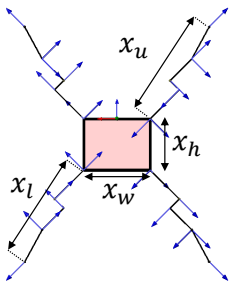


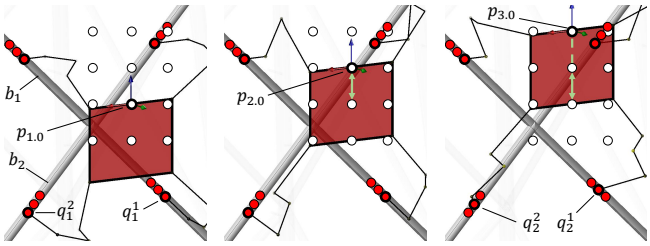
Fig. 2: A four-limbed robot with size parameters. Joint arrows indicate axes of rotation.

The proposed model will calculate truss structure coverage based on constraints applied to sets B, Q, and T, for a given multi-limbed robot (Fig. 2) with size parameters $\mathbf{x} \in \mathbb{R}^n$. For example in the figure, a four-limbed robot can have four size parameters defined as $\mathbf{x} = [x_u, x_l, x_w, x_h]$, representing the length of the upper limbs, lower limbs, body width and body height respectively ($n = 4$). A discretised optimisation problem will then be set up to optimise $\mathbf{x}$, based on the application of truss structure coverage constraints.

C. Robot-Truss Structure Interaction

Fig. 3 shows how the the robot interacts with the interaction points Q on a truss structure B in the environment space T. The layout of environment space T generated around the truss structure represents a discretised transform $p \in SE(3)$, set by a user-defined discretisation method in g_t.

In this example, all $p_{i,0}$ poses represent the position of the robot's body parallel to the truss plane. Through gripper continuity analysis, poses $p_{1.0}$ and $p_{2.0}$ are connected because gripper positions between them are fixed, marked with a bi-directional arrow. However, poses $p_{2.0}$ and $p_{3.0}$ are



(a) Pose 1.0 (b) Pose 2.0 (c) Pose 3.0

Fig. 3: A robot at three possible poses in the environment space (white markers p) with interaction points (red markers q^b) on trusses b.

not connected because the lower grippers require a different grasping positions. Pose connectivity analysis is an important step in calculating truss structure coverage constraints, which will be explained in Section III-B.3.

III. MODEL FORMULATION

To determine the minimum size robot $\mathbf{x}$ that satisfies truss structure coverage constraints, a discrete optimisation problem is set up:

$$\begin{aligned} \min_{\mathbf{x}} \quad & f_{obj}(\mathbf{x}) \\ s.t. \quad & lb_i \leq x_i \leq ub_i \qquad i = 1, ..., |\mathbf{x}| \\ & G_i(R(\mathbf{x}, B, Q, T)) \leq 0 \qquad i = 1, ..., m \end{aligned} \tag{4}$$

where

$\mathbf{x}$ = Parameters to optimise
f_{obj} = Objective function
lb_i, ub_i = Lower and upper bound constraints
R = The set of interaction points as a function of $\mathbf{x}$, subject to collision, grasp, stability and connectivity constraints on sets B, Q and T
G_i = Truss structure coverage constraints on R
m = Number of constraints

A. Calculating Truss Structure Coverage

The set $R(\mathbf{x})$ models truss structure coverage with robot parameters $\mathbf{x}$. Henceforth, we will reference $R(\mathbf{x})$ as the model for calculating truss structure coverage. In order to apply constraints that are meaningful to the accuracy of model, a *configuration space* of the robot covering all possible configurations around the truss structure is generated. The proposed configuration space V is a set of all configurations represented by the triplet

$$u = (p, U^G, U^R). \tag{5}$$

where u is the configuration of a robot at pose $p \in P$, and U^G and U^R are the sets of all graspable and reachable interaction points for that pose, respectively. An interaction point is *graspable* when the robot's gripper is able to apply a stable attachment point to it, and an interaction point is *reachable* when the gripper is able to observe or touch it. Finally, we define the set of all u as the configuration space,

$$V = \left\{u_1, u_2, ...u_{|P|}\right\}. \tag{6}$$

Our definition of coverage is the robot's ability to *reach* an interaction point. Therefore, overall truss structure coverage h is defined as

$$h = \frac{|q^b \in \bigcup Q^R|}{|Q|}, \tag{7}$$

where Q^R is the set of all U^R from the triplets in the configuration space V. The elements in Q^R are dependent on $\mathbf{x}$, i.e. $h(\mathbf{x})$. We can then express this as a coverage constraint equation, where h_{min} is the target truss structure coverage quotient

$$G(\mathbf{x}) = h_{min} - h(\mathbf{x}). \tag{8}$$

B. Modelling Constraints

This section defines all functions and constraints that are applied to B, Q and T to generate $R(\mathbf{x})$. Constraint functions g are non-linear and difficult to express as inequality equations, and are represented like functions of a program. Constraint functions f return values for use in inequality constraints between the robot and truss structure, and can be easily tuned to as per the user's requirements.

1) Physical Constraints:

a) Body-truss structure collisions: Let g_s be the mapping of the robot body's pose $p \in P$, to a set of limb base points, such that $g_s : (p, \mathbf{x}) \rightarrow S$, where the ordered pair (p, S) represents the shoulder and hip positions in the world frame ($S \in \mathbb{R}^3$) for a particular body pose $p \in P$ and robot size $\mathbf{x}$.

In addition, let $g_c(S, B)$ be a function that evaluates whether the robot's body is in collision with any truss in B, given hip and shoulder points S at pose p. Then the set of all poses $p \in T$ that is not in collision with any trusses $b \in B$ is

$$P = \{p : p \in T,\ g_c\left(g_s(p, \mathbf{x}), B\right)\}. \tag{9}$$

b) Kinematic constraints: Let g_{kg}^j be a function for determining the existence of an inverse kinematic solution for the j-th limb with gripper attached. The set of all graspable points for the j-th limb at pose p_i is therefore

$$Q_{i,j}^G = \left\{q^b \in Q : g_{kg}^j(p_i, \mathbf{x}, Q)\right\}. \tag{10}$$

These elements are combined to obtain the set of graspable points for all limbs for pose p_i

$$U_i^G = \left\{Q_{i,1}^G, Q_{i,2}^G, Q_{i,3}^G, Q_{i,4}^G\right\}. \tag{11}$$

Note that constraint function g_{kg}^j can also be used to set constraints on grasping poses; the interaction between a gripper and beams.

Truss structure coverage h (Eqn. 7) is dependent on reachable interaction points, therefore the set of reachable points for all limbs for pose p_i is also be calculated. Let g_{kr}^j be a function for determining the inverse kinematic solution for the j-th limb without grasping constraints. Then

$$Q_{i,j}^R = \left\{q^b \in Q : g_{kr}^j(p_i, \mathbf{x}, Q)\right\}, \tag{12}$$

so the set of reachable points for all limbs for pose p_i is

$$U_i^R = \left\{Q_{i,1}^R, Q_{i,2}^R, Q_{i,3}^R, Q_{i,4}^R\right\}. \tag{13}$$

Eqns. (11) and (13) make up the configuration triplet in Eqn. (5) in index form

$$u_i = (p_i, U_i^G, U_i^R), \tag{14}$$

where u_i is the configuration of the robot at the i-th pose p_i in the configuration space V defined in Eqn. (6). Note that the generation elements u is dependent on $\mathbf{x}$. Therefore we can assume $u = u(\mathbf{x})$ when defining constraint equations.

2) Stability Constraint: There are two constraints to consider for static stability of a robot at a given pose: gait stability and torque.

a) Gait stability: Let f_s^g be a function that calculates the number of grasping points per pose, such that

$$f_s^g(\mathbf{x}) = |\left\{Q^G \in U^G : Q^G \neq \emptyset\right\}|, \tag{15}$$

where for a configuration $u_i = (p_i, U_i^G, U_i^R)$, $U_i^G = \left\{Q_{i,1}^G, Q_{i,2}^G, Q_{i,3}^G, Q_{i,4}^G\right\}$. When expressed as a constraint where a minimum of C_s^g grasping points is required per pose, this can be written as

$$G(\mathbf{x}) = C_s^g - f_s^g(\mathbf{x}). \tag{16}$$

Based on [23], we recommend $C_s^g = 3$, which maintains a minimum of three contact points for stability.

b) Torque constraint: Let f_s^t be a function that calculates torque for a given configuration such that $f_s^t = g_a(u)$, where g_a is a user-defined function which calculates torque requirements based on the robot's configuration u. An estimation of torque, based on calculation of the grasping area using the *Shoelace formula*[26] is proposed in this model to significantly improve calculation times. Hence g_a can represent a function that calculates the maximum grasp area among all possible grasp combinations at configuration u. This constraint can then be written as

$$G(\mathbf{x}) = C_s^t - f_s^t(\mathbf{x}) \tag{17}$$

where a suitable value of C_s^t (m^2) should be chosen based on the capability of the robot's actuators. Based on simulated experiments on four-limbed robot grasps, a grasping area greater than 0.3 m^2 is utilised in this paper to ensure static stability of a configuration. Both constraints f_s^g and f_s^t are applied to all configurations in V. The resultant set is

$$V' = \left\{u \in V : f_s^g(u) \text{ and } f_s^t(u)\right\}, \tag{18}$$

where V' is the set of all robot configurations around the truss structure that are statically stable when grasping.

3) Connectivity Constraints: Two configurations in V', u_i and u_{i+1} are considered adjacent when their poses satisfy $|p_i - p_{i+1}| = 1$ unit. Let g_d be a function that returns a set of adjacent configurations, then for configuration u_i,

$$U_i^a = \{u_i \in V' : g_d(u_i, V')\} \tag{19}$$

where U_i^a is the set of all adjacent configurations to u_i.

a) Connectivity via valid gait: In order for locomotion to occur, adjacent configurations should have common grasping points. Consider adjacent configurations u_i and u_{i+1}, and let $U_i^{G'}$ be a set containing the sets of common grasping points for each limb between these configurations

$$u_i^C = \left\{Q_{i,1}^G \cap Q_{i+1,1}^G, ..., Q_{i,n_l}^G \cap Q_{i+1,n_l}^G\right\}, \tag{20}$$

where n_l is the number of limbs on the robot. Let g_g be a function that performs this operation for all elements in U_i^a, such that

$$U_i^C = \left\{u^C : g_g(u_i, U_i^a)\right\}, \tag{21}$$

where each element in U_i^C is the set of common grasping points for each limb, at adjacent configurations to u_i. Now let f_c^g calculate the number of limbs with common grasping points on a single element u_i^C, such that

$$f_c^g(u_i^C) = \left|\left\{Q \in u_i^C : Q \neq \emptyset\right\}\right|. \tag{22}$$

We can constrain gait connectivity by imposing a C_c^g-limb grasp between common grasping points. This is defined as

$$G(\mathbf{x}) = C_c^g - f_c^g(\mathbf{x}), \tag{23}$$

where C_c^g is the number of limbs required to stay attached to the truss structure during locomotion. Similarly, it is recommended that $C_c^g = 3$ for gait stability.

b) Connectivity under torque constraints: In addition, these common grasp points should be considered stable by having contact points exceed C_c^t m^2 in grasp area. Let f_c^t apply this constraint. Then

$$f_c^t : g_a(u_i^C) \geq C_c^t, \tag{24}$$

where g_a is the same function used in the static configuration case. A grasp area of 0.09 m^2 was chosen to ensure torque constraints are met during locomotion. This value was verified using stability analysis with multiple experiments.

c) Connected configuration space: A connectivity graph is generated by considering all statically stable poses in $p \in V'$, and generating sets of adjacently connected poses for each of them after considering constraints. For pose p_i,

$$C_i = \left\{u \in V' : f_c^g(U_i^C),\ f_c^t(U_i^C) \text{ and } u_i \in V'\right\}, \tag{25}$$

where C_i is a set of configurations adjacent to u_i that satisfies connectivity constraints, and $U_i^C = g_g(u_i, g_d(u_i, V'))$. Once the connected pose sets are generated for all $p \in V$, each element is expressed as an ordered pair in set W. Let f_c be a function that generates the set C_i for each configuration u_i in Eqn. (25). Then set W is defined as

$$W = \{(u, C) : u \in V', f_c : u \to C,\ C \subset V'\}. \tag{26}$$

The set of statically-stable and connected poses W is called the *connected configuration space*. From this set, we can obtain graphs of connected configurations, represented as sets of configurations $G \subseteq V'$. Let Γ be the set of all sets of connected configurations in the environment, then

$$\Gamma = \{G_1, G_2, ..., G_k\}. \tag{27}$$

C. Deployment constraint

Let g_d be a a function such that

$$P_d = \{p \in P : g_d(\mathbf{p}_r)\} \tag{28}$$

where $\mathbf{p}_r$ is the proposed deployment area for the robot given as a range in Cartesian co-ordinates, and P_d is the set of poses $p \in P$ that represents this area. Any connected graphs of configurations in Γ that contain any starting poses in P_d are therefore reachable from the starting poses. Hence we can calculate the total reachable set of robot configurations $u \in V$ around the truss structure

$$R = \left\{ u \in \bigcup R' \right\}, \tag{29}$$

where

$$R' = \{G \in \Gamma : P_d \subset G\}. \tag{30}$$

R is the set of all connected and statically stable configurations, for robot size $\mathbf{x}$. The total truss structure coverage is the arbitrary union of all interaction points contained in R. Table I summarises all functions and constraints needed to model truss structure coverage.

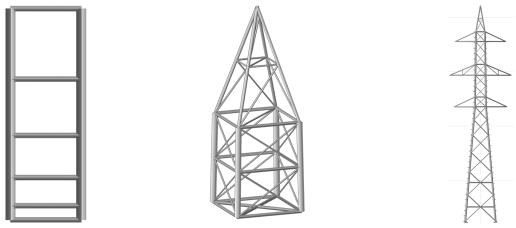
Fcn.	Const.	Summary
g_t		Discretisation scheme for T
g_c		Body-truss collision constraint
g^j_{kg}		Robot IK constraint for the j-th limb
g^j_{kr}		Same as g^j_{kr}, no grasp requirement
g_a		Torque model based on grasp area
f^g_s	C^g_s	Gait stability constraint
f^t_s	C^t_s	Torque constraints
f^g_c	C^g_c	Connectivity under gait constraint
f^t_c	C^t_c	Connectivity under torque constraint
g_d	$\mathbf{p}_r$	Deployment constraint range $\mathbf{p}_r$

TABLE I: List of functions and constants used in the model.

IV. CASE STUDIES

Three case studies on different truss structures were used to verify the model. The first study considers a simple ladder structure (Fig. 4a) that is 1 m wide and whose rungs are progressively spaced apart at 0.2, 0.4, 0.6, 0.8 and 1.0 m from the bottom up. With a simple structure, this tests the intuitiveness of the result of the model. The second case study focusses on the top section of a truss structure that tapers to a single point (Fig. 4b). Due to a narrow grasp profile, this case study tests application of stability constraints, and optimises the robot's size accordingly. The final case study covers the modelling and optimisation on a transmission tower of approximately 26 m high (Fig. 4c). We used a mathematically-generated model based on a 66 kV power transmission tower in Japan. This complex structure contains varying gaps between trusses which will test connectivity and stability constraints.

For all case studies in this section, we fix the body width and height ($x_w = w_h = 0.5$ m) such that the optimisation problem is two-dimensional. The optimisation vector for the upper and lower limbs respectively is



(a) Ladder (b) Top of a tower (c) Transmission tower

Fig. 4: Truss structures used in the following case studies.

$$\mathbf{x} = \begin{bmatrix} x_u & x_l \end{bmatrix}, \tag{31}$$

with the objective function

$$f_{obj}(\mathbf{x}) = {x_u}^2 + {x_l}^2 \tag{32}$$

to minimise the overall size of the robot, subject to bound constraints

$$0.4 \leq x_u \leq 0.8 \qquad 0.4 \leq x_l \leq 0.8 \tag{33}$$

and

$$G_1(\mathbf{x}) = C^g_s - f^g_s(\mathbf{x}) \qquad G_2(\mathbf{x}) = C^t_s - f^t_s(\mathbf{x}) \tag{34}$$

$$G_3(\mathbf{x}) = C^g_c - f^g_c(\mathbf{x}) \qquad G_4(\mathbf{x}) = C^t_c - f^t_c(\mathbf{x}). \tag{35}$$

where

Constraint	Summary
$C^g_s = 3$	Minimum grasping points at static pose
$C^t_s = 0.3$	m^2 minimum grasping area
$C^g_c = 3$	Minimum grasping points for gaits
$C^t_c = 0.09$	m^2 minimum grasping area for gaits
$\mathbf{p}_r$	$0 \leq z \leq 1$, deploy from ground to 1 m

And finally, the non-linear function definitions are

Function	Summary
g_t	Body Position: Grid at $\delta 0.1$ m Orientation: Planar to truss ($\delta \frac{\pi}{2}$ rad) No inverted poses
g_c	Collisions calculated via geometry
g^j_{kg}	IK for full 7-DoF limb with spherical wrist
g^j_{kr}	Same as g^j_{kg} but without grasp constraint
g_a	Grasp area via Shoelace formula

We also consider that only the upper limbs can carry a maintenance tool. This requires a slight redefinition of the truss structure coverage Eqn. (7) is modified such that it only counts the interaction points for limbs 1 and 2

$$Q'^R_{1,2} = \left\{ \left\{ Q^R_1 \in R \right\} \cup \left\{ Q^R_2 \in R \right\} \right\}. \tag{36}$$

This means

$$Q^R_{1,2} = \left\{ q^b \in \bigcup Q'^R_{1,2} \right\}, \tag{37}$$

and therefore

$$h_{1,2} = \frac{\left| Q^R_{1,2} \right|}{|Q|} \tag{38}$$

where $h_{1,2}$ is the coverage as observed from limbs 1 and 2.

Any optimisation algorithm that handles discrete models can be used to solve this problem. However, the *exhaustive-search* method was used in this paper to visualise how an optimisation algorithm will converge to a solution.

A. Ladder

We seek an optimally-sized robot that is able to cover 100% of a ladder-like structure (Fig. 4a). To enforce this constraint, we define

$$G_5(\mathbf{x}) = 1 - h_{1,2}(\mathbf{x}). \tag{39}$$

Using the optimisation model, the optimally-sized robot achieves full coverage of this ladder structure is

$$\mathbf{x} = \begin{bmatrix} 0.8 & 0.6 \end{bmatrix}. \tag{40}$$

To confirm this result, we can refer to a contour map, where objective function and truss structure coverage $h_{1,2}$ are overlaid on each other as a contour map, and plotted against the optimisation variables x_u and x_l (Fig. 5a). The optimised solution is closest to the bottom-left most point of the plot (the origin), which is observed to be correct. The result shows that the total arm span (height) of the robot required to bridge the 1 m gap between the rungs is $0.8 + 0.5 + 0.6 = 1.9$ m.

B. Top Section of a Tower

For case study of the top of a tower (Fig. 4b), the same assumptions apply from the ladder case study. The optimally-sized robot that achieves full coverage of this structure is

$$\mathbf{x} = \begin{bmatrix} 0.46 & 0.28 \end{bmatrix}. \tag{41}$$

Again to confirm this result, we refer to the contour map for this problem (Fig. 5b). This shows it is more efficient to increase the size of the upper limbs for increasing truss structure coverage, due to the tapering nature at the top, decreasing grasp area and therefore grasp stability of the robot. This information is useful when determining the cost-benefits of sizing limbs during the design phase of a robot.

C. Whole Transmission Tower

In this case study of a whole power transmission tower (Fig. 4c), the model is analysed in two parts: coverage modelling, and optimisation results.

1) Coverage modelling: With the model $R(x)$, all reachable interaction points on the structure are determined at varying lengths of the upper and lower limbs ($x_u = x_l$ in the figure). In addition, the coverage redundancy can be calculated, as observed by the colouring of the interaction points on the tower. Darker regions indicate interaction points that are reachable but highly constrained.

In this tower model, the beam positions are less dense as height increases, hence there are larger gaps between beams in the middle of the tower below the cage. This is why in this result, we see that at shorter limb lengths, the robot is incapable of bridging the gaps between beams in the middle of the tower. In addition, we observe that coverage does not increase significantly beyond 40 cm limb length because grasping area decreases at the top of the tower. This indicates that cost-effectiveness for robot size increase versus coverage peaks at approximately this point.

2) Optimisation: In this example, we apply a 99.5% coverage constraint with only upper limbs can carrying the maintenance tool

$$G_5(\mathbf{x}) = 0.995 - h_{1,2}(\mathbf{x}). \tag{42}$$

With these constraints, the optimised robot size is

$$\mathbf{x} = \begin{bmatrix} 0.55 & 0.33 \end{bmatrix}, \tag{43}$$

This results in a total robot arm span (height) of 1.6 m when including the 0.5 m tall body. Fig. 5c shows the contour map of truss structure coverage $h_{1,2}(\mathbf{x})$, which shows the optimum solution to be correct.

V. CONCLUSION

We have proposed a model that can optimise the overall coverage of a multi-limbed robot navigating a truss structure. Utilising the representation methods, the model was broken down into fundamental equations which can be expressed in set notation. The simple, yet detailed representation of the problem allows the application of specialised truss structure coverage constraints in addition to other standard constraints, such as static and gait stability, grasp redundancy and targetted area coverage. The model is applied as an optimisation problem, such that when solved, represents the most optimised robot that satisfies all constraints for a truss structure. This is an alternate approach to the optimisation of a the design of a robot, where the proposed model specialises in detailed coverage of truss structure.

The case studies show how the model can be applied in different scenarios of varying complexity and size. In each study, the optimised solution for each robot is verified with contour maps. By visualising the truss structure coverage model, the design process can be optimised as observed in all case studies. This information is valuable for evaluating the cost-effectiveness of design changes regarding limb sizing.

Future work for this model includes minimising the effects of discretised optimisation by improving the efficiency of the model. This can be achieved by reducing the memory footprint in higher-dimensional problems, and improving the area function g_a, as this function is computationally expensive for truss structures with many interaction points.

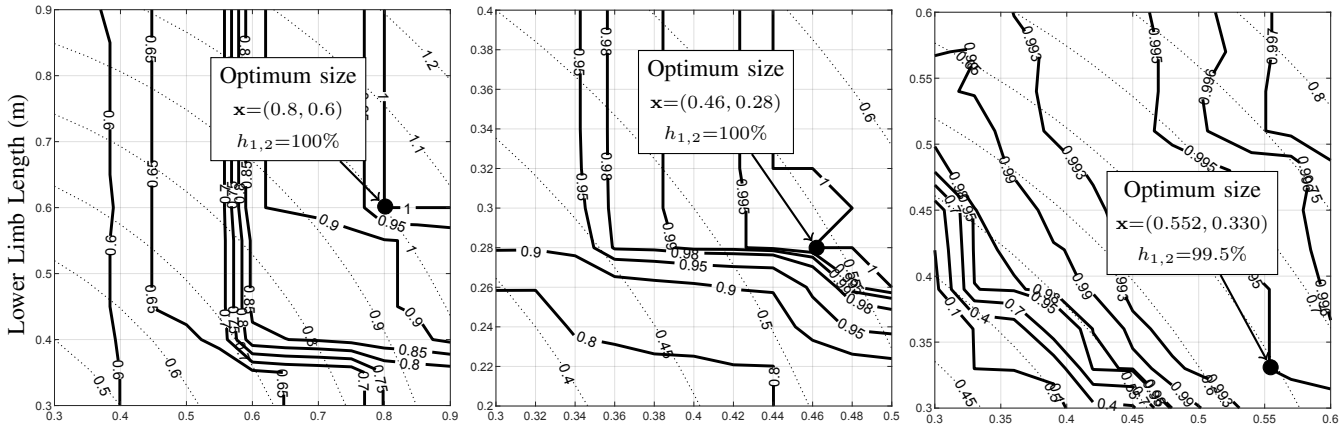
ACKNOWLEDGEMENTS

We thank Dr. Lakshitha Dantanarayana, Mr. Thomas Hudson and Mr. Tristan Robinson for their input and testing of this model. This work was supported by Tokyo Electric Power Company Holdings, Inc.

REFERENCES

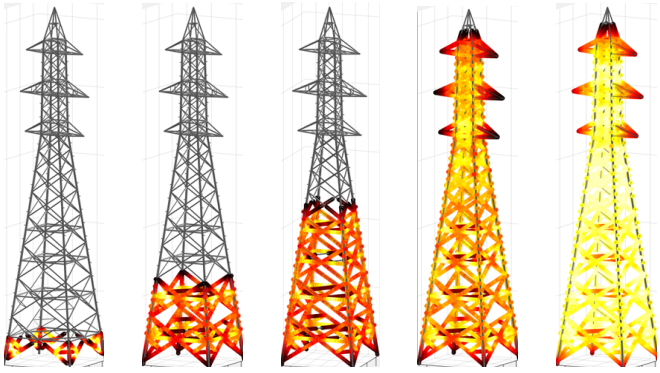
[1] Cecilia Laschi and Barbara Mazzolai. Lessons from animals and plants: the symbiosis of morphological computation and soft robotics. *IEEE Robotics & Automation Magazine*, 23(3):107–114, 2016.

[2] Rolf Pfeifer, Max Lungarella, and Fumiya Iida. Self-organization, embodiment, and biologically inspired robotics. *science*, 318(5853):1088–1093, 2007.



(a) Ladder structure (b) Top of a tower (c) Power transmission tower

Fig. 5: Contour maps of the case studies. Thick contours represent $h_{1,2}$ and the thin dotted contours represent f_{obj}.



(a) 30 cm $h = 5.3\%$ (b) 35 cm 30.7% (c) 36 cm 50.3% (d) 40 cm 98.4% (e) 60 cm 98.6%

Fig. 6: Coverage maps of a transmission tower with varying limb lengths. Bright areas indicate higher grasp redundancy.

[3] Carlo Menon and Metin Sitti. Biologically inspired adhesion based surface climbing robots. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 2715–2720. IEEE, 2005.

[4] Matthew J Spenko, G Clark Haynes, JA Saunders, Mark R Cutkosky, Alfred A Rizzi, Robert J Full, and Daniel E Koditschek. Biologically inspired climbing with a hexapedal robot. *Journal of Field Robotics*, 25(4-5):223–242, 2008.

[5] Kenneth J Waldron and Christopher Hubert. Scaling of robotic mechanisms. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 1, pages 40–45. IEEE, 2000.

[6] Ou Ma and Jorge Angeles. Optimum architecture design of platform manipulators. In *Fifth International Conference on Advanced Robotics' Robots in Unstructured Environments*, pages 1130–1135. IEEE, 1991.

[7] Leo Stocco, SE Salcudean, and Farrokh Sassani. Matrix normalization for optimal robot design. In *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No. 98CH36146)*, volume 2, pages 1346–1351. IEEE, 1998.

[8] Vincent Nabat, Sébastien Krut, Philippe Poignet, François Pierrot, et al. On the design of a fast parallel robot based on its dynamic model. In *Experimental Robotics*, pages 409–419. Springer, 2008.

[9] VP Agrawal, V Kohli, and S Gupta. Computer aided robot selection: the 'multiple attribute decision making'approach. *The International Journal of Production Research*, 29(8):1629–1644, 1991.

[10] C Balaguer, A Gimenez, and CM Abderrahim. ROMA robots for inspection of steel based infrastructures. *Industrial Robot: An International Journal*, 29(3):246–251, 2002.

[11] Yeoreum Yoon and Daniela Rus. Shady3d: A robot that climbs 3d trusses. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 4071–4076. IEEE, 2007.

[12] Anirban Mazumdar and H Asada. An underactuated, magnetic-foot robot for steel bridge inspection. *Journal of Mechanisms and Robotics (Transactions of the ASME)*, 2(3):031007 (9)–031007 (9), 2010.

[13] Mahmoud Tavakoli, Lino Marques, et al. 3dclimber: Climbing and manipulation over 3d structures. *Mechatronics*, 21(1):48–62, 2011.

[14] Yisheng Guan, Li Jiang, Haifei Zhu, Xuefeng Zhou, Chuanwu Cai, Wenqiang Wu, Zhanchu Li, Hong Zhang, and Xianmin Zhang. Climbot: A modular bio-inspired biped climbing robot. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1473–1478. IEEE, 2011.

[15] BL Luk, AA Collie, and J Billingsley. Robug II: An intelligent wall climbing robot. In *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*, pages 2342–2347. IEEE, 1991.

[16] BL Luk, AA Collie, V Piefort, and GS Virk. Robug III: a tele-operated climbing and walking robot. In *UKACC International Conference on Control. Control '96*, pages 347–352. IET, 1996.

[17] Shigeo Hirose, Akihiko Nagakubo, and Ryosei Toyama. Machine that can walk and climb on floors, walls and ceilings. In *Advanced Robotics, 1991.'Robots in Unstructured Environments', 91 ICAR., Fifth International Conference on*, pages 753–758. IEEE, 1991.

[18] Manuel Armada, M Prieto, T Akinfiev, Roemi Fernández, P González de Santos, E Garcia, H Montes, S Nabulsi, R Ponticelli, J Sarria, et al. On the design and development of climbing and walking robots for the maritime industries. *Journal of Maritime Research*, 2(1):9–32, 2005.

[19] Aaron Saunders, Daniel I Goldman, Robert J Full, and Martin Buehler. The RiSE climbing robot: body and leg design. In *Unmanned Systems Technology VIII*, volume 6230, page 623017. International Society for Optics and Photonics, 2006.

[20] Tim Bretl, Stephen Rock, Jean-Claude Latombe, Brett Kennedy, and Hrand Aghazarian. Free-climbing with a multi-use robot. In *Experimental Robotics IX*, pages 449–458. Springer, 2006.

[21] PK Ward, Palitha Manamperi, Philip Brooks, Peter Mann, W Kaluarachchi, L Matkovic, G Paul, C Yang, P Quin, D Pagano, et al. Climbing robot for steel bridge inspection: Design challenges. In *Austroads Bridge Conference*. ARRB Group, 2014.

[22] David Pagano, Dikai Liu, and Kenneth Waldron. A method for optimal design of an inchworm climbing robot. In *2012 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 1293–1298. IEEE, 2012.

[23] Robert B McGhee and Andrew A Frank. On the stability properties of quadruped creeping gaits. *Mathematical Biosciences*, 3:331–351, 1968.

[24] Lilian K Takahashi. Morphological basis of arm-swinging: multivariate analyses of the forelimbs of hylobates and ateles. *Folia Primatologica*, 54(1-2):70–85, 1990.

[25] Fana Michilsens, Evie E Vereecke, Kristiaan D'août, and Peter Aerts. Functional anatomy of the gibbon forelimb: adaptations to a brachiating lifestyle. *Journal of Anatomy*, 215(3):335–354, 2009.

[26] Bart Braden. The surveyor's area formula. *The College Mathematics Journal*, 17(4):326–337, 1986.