SOLL-E: A Module Transport and Placement Robot for Autonomous Assembly of Discrete Lattice Structures

In-Won Park[1], Damiana Catanoso[1], Olivia Formoso[2], Christine Gregg[2], Megan Ochalek[2], Taiwo Olatunde[1], Frank Sebastianelli[1], Pascal Spino[3], Elizabeth Taylor[2], Greenfield Trinh[2], and Kenneth Cheung[2]

***Abstract*— This paper presents the design and development of a transport and placement robot that demonstrates autonomous assembly of structural building blocks. The robots are intended to serve as a critical component of automated structural assembly and maintenance systems. The Scaling Omni-directional Lattice Locomoting Explorer (SOLL-E) uses a 5-DoF bipedal inchworm locomotion architecture with locking foot and cargo grippers. The locomotion system employs large magnet gap diameter BLDC motors with moderate timing belt gearing for primary joints, and DC planetary gearmotors for turning. Foot and cargo grippers are identical, with servo-actuated locking mechanisms. Three modular controller boards are used to control these actuators in real-time, with command and telemetry data transferred between the server and each controller board via WiFi. Functionality and performance were evaluated in a ground demonstration.**

I. Introduction

Autonomous robotic assembly of large-scale structures has been envisioned as a solution to sustainable infrastructure, disaster relief, and construction in extreme environments. Autonomous in-space assembly and construction of large-scale instrumentation and infrastructure (on the moon and on-orbit) is a particularly compelling application, since assembly could overcome launch shroud limits to enable larger space structures and more exploration, without needlessly endangering human astronauts.

Multiple approaches have been proposed to achieve automated construction, often focusing on automated assembly of truss structures due to their high specific mass performance. Doggett et al. offer a thorough review of NASA's research into truss assembly using robotic arms [1]. To make large-scale structures, these systems typically use turn-tables, arms mounted on tracks, or take advantage of low-gravity to make very long reach tensegrity arms. Other concepts of on-orbit large assembly and servicing use a walking modular robot arm that can locomote on spacecraft [2]. Large-format 3D printing has been demonstrated but can face scalability challenges, especially for non-terrestrial applications. 3D printing requires that the gantry be larger than the structure being printed, or the gantry be mobile and capable of appropriate alignment. Additionally, printed structures typically require secondary outfitting or finishing with reinforcement, secondary structural elements like windows, or even hybrid elements to support roofs, all of which can be challenging to automate.

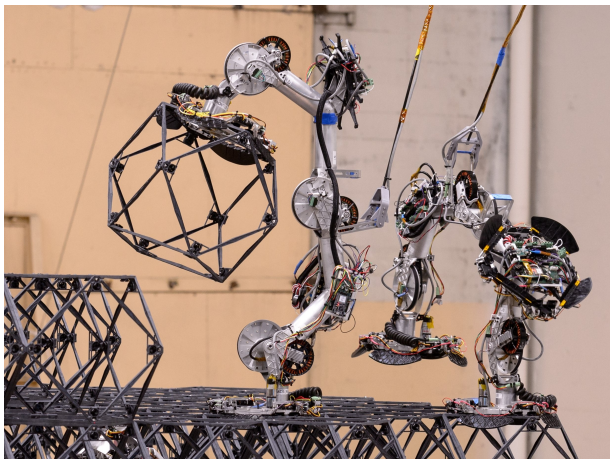


Fig. 1: Two SOLL-E robots perform assembly operations on the lattice structure. The left robot is placing the unit cell at a build location, and the right robot is returning to the supply depot for an additional building block.

To address scalability challenges, the NASA Ames Automated Reconfigurable Mission Adaptive Digital Assembly Systems (ARMADAS) project has developed a structural assembly system based on a programmable matter approach. This approach features light-weight lattice building blocks (termed "voxels") and mobile task-specific robots that locomote on and within the structure to build high-performance trusses. These robots are termed "relative robots" because they exist only relative to the highly structured and periodic lattice environment, allowing the robots to leverage the structure for metrology [3]. In this manner, small and relatively imprecise "blind" robots can build structures larger and more precise than themselves. Based on a previously published architecture study [4], the ARMADAS system features two types of robots that collaborate to assemble the structure: an external material transport/placement robot and an internal bolting robot. The transport/placement robot, also known as Scaling Omni-directional Lattice Locomoting Explorer (SOLL-E), walks on the outside of the structure in an inch-worm fashion as shown in Fig. 1. The bolting robot, also known as Mobile Metamaterial Internal Co-integrator (MMIC-I) [5][6], climbs through the structure to bolt each unit cell building block (voxel) to its neighbors [7][8].

This work was supported by a NASA's Game Changing Development (GCD) Program, Space Technology Mission Directorate

[1]Authors are with KBR Inc., NASA Ames Research Center, Moffett Field, CA 94035 USA in.w.park@nasa.gov

[2]Authors are with NASA Ames Research Center, Moffett Field, CA 94035 USA kenny@nasa.gov

[3]Author is with Massachusetts Institute of Technology, Cambridge, MA 02139 USA

This paper describes the modular architecture, mechanical design, and avionics of the SOLL-E robot. Many modular robot architectures for assembly have utilized inch-worm type robots [9] [10]. Small-scale examples of inch-worm robots such as BILL-E [10] demonstrated efficient and reliable locomotion in a three-dimensional lattice, as well as lattice unit cell transport and assembly. However, BILL-E relies on unit cells with magnets to provide attachment force and alignment during assembly. To meet project structural requirements (specific strength, stiffness, and ease of manufacturing), the ARMADAS project requires a robot capable of aligning and placing its larger-scale voxels without the help of embedded magnets. We discuss the design of the robot to meet these requirements, as well as operational modes and trajectories for locomotion and voxel placement. Data from experimental characterization of custom drive modules is presented, as well as lessons learned from a 164-voxel autonomous build experiment.

II. SOLL-E Architecture

Previous work established the overall ARMADAS system architecture, separating the tasks of material transport and material attachment between two different robots [4]. As such, SOLL-E's primary function in the ARMADAS system is to transport voxels from a depot to the intended placement location. Therefore, the robot must be capable of navigating a 3D grid while carrying a voxel. It must also be able to unload a voxel and place that voxel into the correct grid position with sufficient precision that an internal bolting robot [5] can perform final voxel alignment and attachment to the existing structure.

SOLL-E is a bipedal inchworm robot that can walk along the surface of a voxel structure. The robot is described in Fig. 2. It consists of three main actuators (a "knee" joint and two "ankle" joints), two yaw stage actuators to provide in-plane rotation, two foot modules to grip the lattice structure, and a voxel carrying module (a "cargo backpack" formed by a foot module attached to the upper portion of the robot). Though only one yaw module is strictly necessary to navigate 3D space, a yaw module at both feet allows the robot more planning flexibility as well as more efficient routing across the lattice structure; for example, by moving along diagonals. To maximize the efficiency of design-fabrication-repair life cycles by reducing the number of parts, SOLL-E hardware and mechanisms are highly modular. Components and alignment features are reused across the robot.

Unlike previous lattice-traversing inchworm robot examples that had a dedicated mechanism to carry and place voxels [10], SOLL-E is intended to work in pairs such that the main locomotion degrees of freedom are reused to perform voxel placement. One SOLL-E robot retrieves a voxel from the cargo module of another SOLL-E robot to then function as a mobile robot arm to position the voxel in the correct build orientation. This design was chosen to reduce the individual robot mass, reduce robot degrees of freedom, and improve placement reliability. Unlike previous literature examples, the ARMADAS voxels contain structural alignment features on the voxel face that extend out of plane from the face-to-face mating surface. To avoid catching these features on each other during placement, a dedicated placement mechanism would have required several degrees of freedom and the associated added mass. In 1g, this added mass would have dramatically increased both the torque requirements for the main joints and the torque that must be reacted by the lattice structure during robot locomotion.

III. SOLL-E Hardware

A. Leg Length and Shape Design

Link lengths are designed such that SOLL-E can effectively walk up and down single voxel steps and move between orthogonal planes while minimizing the size, and thus mass, of the robot. If u is the voxel length as shown in Fig. 3, the minimal practical approach has leg lengths approaching $(\sqrt{5}/2)u$ with rotational hinges $u/2$ from the attached voxel face. The link length may be maintained with offset to compensate for the nonzero characteristic diameter of the leg. When compared with a straight leg design, the leg length of the bent design is only minimally longer ($<$20 mm) and preserves robot symmetry with respect to the feet in static poses, which balances mass distribution. The bent leg design also provides the ability to reach around single voxels for additional locomotion and placement flexibility (Fig. 3). The link lengths for the chosen bent leg geometry are listed in Table I.

B. Main Joint Torque Requirements and Module Design

SOLL-E is designed to operate with the ARMADAS voxel, which has a pitch length of 0.3048 m (12 in) and a mass of approximately 380 g. This is larger than previous examples of inchworm lattice crawler robots, necessitating an assessment of the main locomotion module torque density requirements and the availability of commercial off the shelf (COTS) actuators to meet these requirements.

The upper bound on static joint torque, τ_{MAX}, can be determined as the greatest joint torque that is required to support the robot in the worst-case configuration. This torque occurs when the robot is attached to the structure at one end, and stretched out perpendicular to gravity to support a voxel at the distal end. The joint closest to the foot attached to the structure experiences the greatest torque, represented as follows:

$$\tau_{MAX} = ug(\sum k_{link}m_{link} + \sum k_{module}m_{module} + \sum k_{gripper}m_{gripper} + k_{voxel}m_{voxel}) \quad (1)$$

where k is the component type lever factor ($1/u$) and m is the component mass. Based on the link length calculated using the universal inching architecture and the predicted mass of joint links (m_{link}), joint modules (m_{module}), grippers ($m_{gripper}$), and a voxel (m_{voxel}), τ_{MAX} is calculated to be 16 Nm. This static torque requirement applies to both Joint 1 and Joint 3. The torque requirement for Joint 2 is less, but the same actuator is used for all three joints for modularity of components.

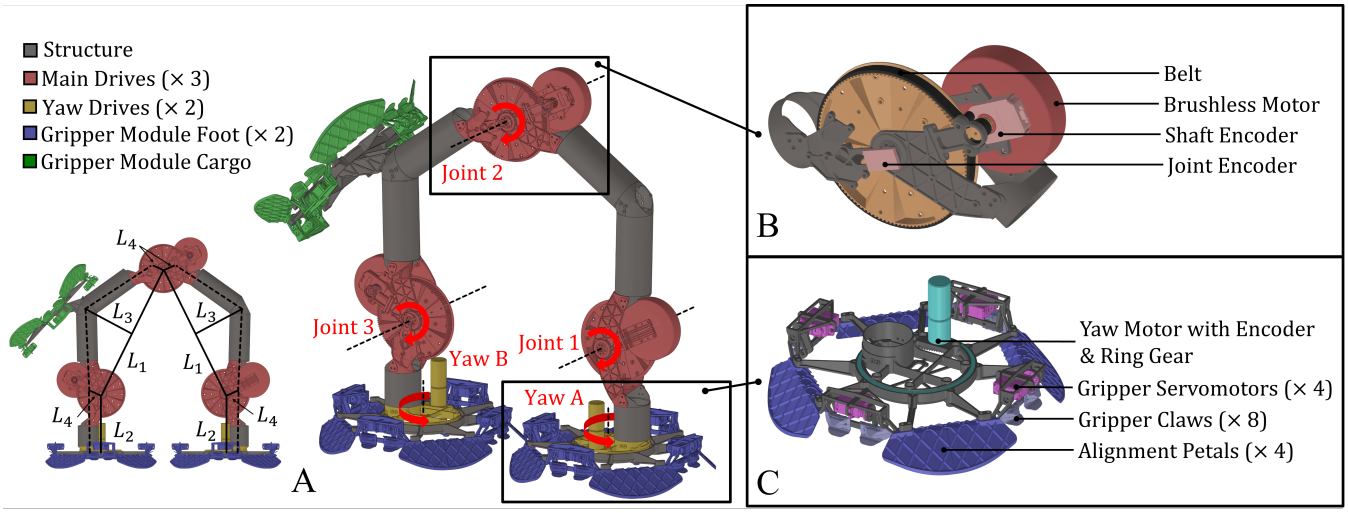


Fig. 2: Diagram of the SOLL-E robot and key subsystems. (A) Color coded view of the robot in its initial (stand) pose with labels for the five rotational axes and description of linkage lengths for Table I. (B) Isolated view of the main drive actuator with key components labelled - these actuators are custom-built for high torque density and identical across the three joints. (C) Isolated view of the foot module with key components labelled - this subsystem is capable of passive alignment and active gripping of the voxel structure and contains a yaw stage actuator allowing the robot to pivot on either foot.

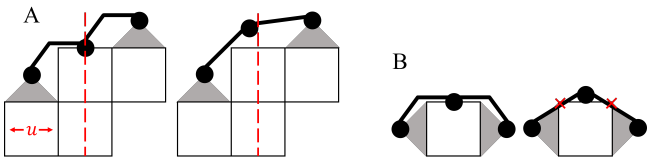


Fig. 3: For legs of approximately equal length, (A) the bent leg preserves a certain geometric and mass distribution symmetry that the straight leg does not, and (B) the bent leg allows for certain robot poses that are unfeasible for the straight leg due to collision with voxel edges. Note that the voxels have open faces, allowing Joint 2 to move into their envelope without collision.

The symmetry of the robot necessitates high torque actuators at the least ideal locations; the distal ends. To operate outside of the quasi-static domain and achieve reasonable structure build times, actuators with a low gear ratio in addition to a high torque density are desirable. An approach similar to that of the MIT cheetah was used [11]. The actuators for Joints 1, 2, and 3 consist of large gap radius brushless motors with small gear ratio transmission; the most easily sourced motor that met our specifications required a gear ratio of 10.8:1. A single stage belt system is built into the joint to perform this reduction. The drive assembly provides a holding torque of 21.1 Nm, which is above the minimum required torque of 16 Nm. The drive assembly also provides momentary torques of approximately 28 Nm for periods up to three seconds. This design provides sufficient performance from standstill (low RPM) while also providing margin for control authority over dynamic effects. With the total actuator module weighing just 557 g including control electronics, this system presents an acceptably high torque density of 50 Nm/kg.

TABLE I: SOLL-E Hardware Specification

Element	**Specification**
Mass	cargoSOLL-E: 4.8 kg craneSOLL-E: 4.4 kg
Link Length	L_1: 0.341 m L_2: 0.152 m L_3: 0.116 m L_4: 0.031 m
Main Drive Motor	Pole Count: 42 Motor KV: 100 kV Operating Voltage: 48 V Rotor Diameter: 86.8 mm
Main Drive Assembly	Max Holding Torque: 21.1 Nm Max Momentary Torque: ∼28 Nm <3 sec Gearing: 10.8:1
Range of Motion (from initial configuration)	YawA θ_A: $-180°$ to $180°$ Joint1 θ_1: $-55°$ to $135°$ Joint2 θ_2: $-128°$ to $35°$ Joint3 θ_3: $-58°$ to $132°$ YawB θ_B: $-180°$ to $180°$

The belts were sized based on COTS industrial components. A 6 mm belt width specified to 516 N of strength can provide 21.3 Nm of output torque from a 130 tooth output pulley driven by 2.96 Nm of input from a 12 tooth pulley. Although the belt drive is susceptible to wear and thermal expansion, a belt drive configuration was chosen over a more conventional planetary gear arrangement for its low mass and ability to provide some compliance during the early development phase.

C. Mechanical

Table I summarizes the mass, link length, main drive motor, main drive assembly, and range of motion due to mechanical limits for SOLL-E. The cargoSOLL-E and

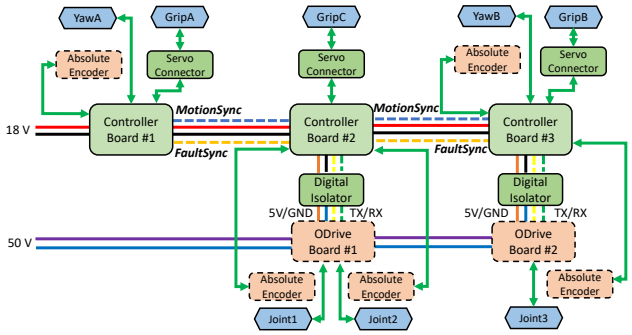


Fig. 4: Block diagram of SOLL-E avionics power and data distribution. Components with a solid border are custom designed; components with a dashed border are COTS.

craneSOLL-E designations correspond to versions of the robots with and without cargo modules, respectively. The total mass of a SOLL-E robot is 4.8 kg including all avionics and harness components. A tether is used to power the system for ease of early experimentation, but a 1 kg lithium battery integration is within scope for a 1g environment given the torque density of the drive systems.

Each leg has a foot mechanism (GripA and GripB) which consists of four Hitec D89MW servos and machined aluminum and injection-molded thermoplastic composite parts that align and attach the robot to the voxel structure. This gripper allows SOLL-E to connect and disconnect legs for motion along arbitrarily oriented voxel surfaces. The voxel-holding "backpack" (GripC) uses an identical mechanism, which can grip and ungrip to pick-up and drop-off voxels. The ankle module allows rotation in both pitch and yaw directions. For yaw movement (YawA and YawB), a ServoCity 26 RPM DC motor with a relative encoder, bearing, and ring gear are integrated with a coiled harness to rotate $\pm 360^{\circ}$ from the initial configuration.

For pitch movement (Joint1 and Joint3), a SunnySky M8 BLDC motor, CUI AMT102-V relative encoder, bearing, and belt are integrated. Two legs from each ankle module meet at a knee module (Joint2) that provides a single degree of freedom in the pitch direction and has an identical mechanism to the pitch movement of the ankle modules. For all DC (YawA and YawB) and BLDC (Joint1, Joint2, and Joint3) motors, AS5048 absolute encoders are integrated at the output stage for position sensing and to detect possible gear and belt slip, respectively.

D. Avionics

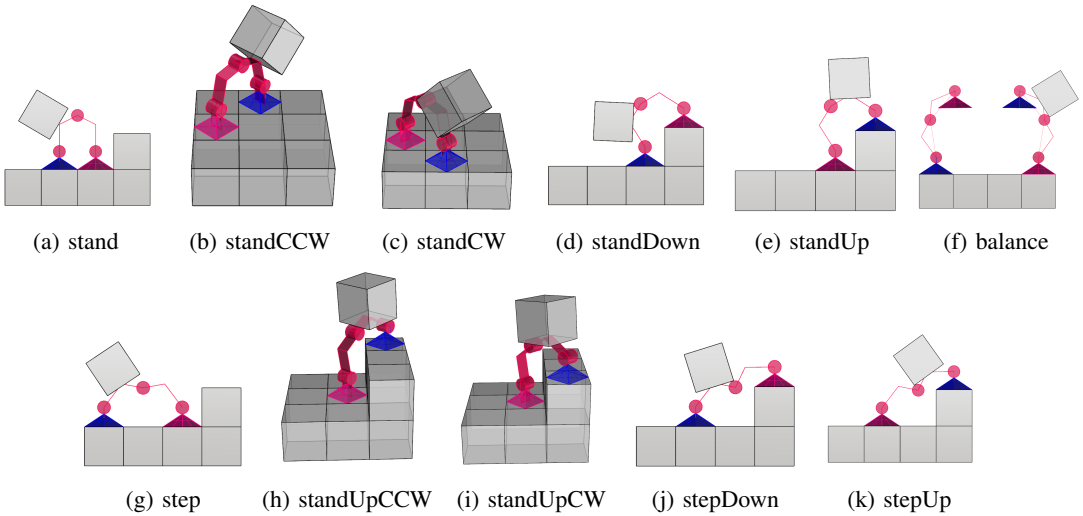
Fig. 4 shows the data and power systems architecture of the SOLL-E robot. Solid rectangles, including the controller board, servo connector board, and digital isolator board, are custom designed. COTS products are represented by dotted rectangles. The DC motors (YawA and YawB) and servos (GripA, GripB, and GripC) are powered by an 18 V power supply via regulators on the controller boards. Since the BLDC motors (Joint1, Joint2, and Joint3) are controlled using COTS ODrive boards, they are directly powered by a 48 V power supply. The power bus between each controller board and ODrive board is daisy-chained. To avoid any ground loops, digital isolator boards are integrated in between the controller board and the ODrive board. Ferrite rings are added to the BLDC motor phase harness to reduce capacitively coupled noise, and shielded cables are used on all absolute and relative encoder harnesses to mitigate EMI noise.

TABLE II: SOLL-E Fault Monitoring

Fault	**Trigger**
Stop robot	When the stop command is sent from the server
Gripper over current	When the current draw of gripper servo exceeds nominal operational level
Yaw motor over current	When the current draw of yaw DC motor exceeds nominal operational level
Yaw motor out of range	When the command value for yaw DC motor exceeds nominal operational value
Joint 1/2/3 motor response timeout	When the Joint 1/2/3 motor does not receive feedback from the external motor driver within the expected time frame
Joint 1/2/3 motor target mismatch	When the Joint 1/2/3 motor does not return the commanded target within the expected time frame
Joint 1/2/3 Absolute encoder mismatch	When the discrepancy between absolute joint and incremental shaft encoders for Joint 1/2/3 exceed an allowable threshold
Synchronization fault	When the board synchronization is indeterminate

Each robot is controlled by three identical controller boards, with all of the actuators and sensors divided between them. An ESP32 micro-controller is used on each controller board and communicates independently to the base station using WiFi. The ESP32 is a dual-core micro-controller where one core is dedicated to wifi communication handling and the other controls actuators and processes sensor data in real-time. The controller board is designed to control any two motors and one set of gripper servos. The board has three voltage regulators to handle its power. Different voltage buses exist for the DC motors, servos, and microcontrollers. A voltage monitoring circuit is integrated to disable actuator power when a voltage drop from the input source is detected. Two current sensors are also included, which monitor the current consumption of the servos.

The relative encoders are necessitated by the ODrive v3.6 controller and are connected directly to the ODrive. For the controller boards, the absolute encoder signals are read through I^2C protocol, and communication with the ODrive board uses UART. Since each controller board communicates to the base station separately, the boards are connected to one another using GPIOs in order to synchronize both the start of motions and any fault alerts. "Controller Board #1" acts as the leader among the three boards, deciding when to send this information to the two follower boards. The firmware architecture is identical in all three boards, except this 'MotionSync' and 'FaultSync' handling.



(a) stand (b) standCCW (c) standCW (d) standDown (e) standUp (f) balance

(g) step (h) standUpCCW (i) standUpCW (j) stepDown (k) stepUp

Fig. 5: List of all SOLL-E locomotion configurations (GripA = purple, GripB = blue). All robot locomotion sequences can be attained by moving between some subset of these configurations.

IV. Operation Modes

SOLL-E has standby, operational, and safed modes that are automatically or manually triggered depending on the condition. These three modes of operation allow for robot initialization, operations, and fault recovery.

Standby mode is the default startup mode and allows for initialization and calibration of robot parameters as well as minimal power consumption. Once initial calibration has been completed, the robot can be commanded to enter operational mode.

In operational mode, the robot is fully powered and can receive individual commands that contain the target angle for each joint, or execute macro commands that move multiple joints with pre-stored trajectory motions. Every 25 ms, the controller board controls each actuator and processes the external absolute encoder sensor readings, ODrive board readings, and actuator current readings. The robot can receive a command to enter back into standby mode in order to disable torque on the BLDC motors.

Safed mode is triggered when the robot receives a stop command from the base station or an internal system fault is triggered. In this mode, all motors are halted in their current position to prevent collisions with the structure. The system faults, shown in Table II, are designed to prevent damage to the robot and structure from errors in planning, communications, hardware, or operations. Sources of faults can include: system over current, BLDC communication issues, or excessive sensor noise. In each case, there are procedures for response recovery and debugging.

V. Trajectory

The transitions of the SOLL-E robot between different configurations are defined by trajectories. Three different types of trajectories are used for carrying out needed operations: locomotion, grab, and placement trajectories. The sequence of joint angles and yaw rotations that make up a complete trajectory has been determined in an analytical manner. Inverse kinematics were used to create the baseline intermediate points between starting and ending configurations. For the grab and placement trajectories, some waypoints were fine tuned due to compliance in the robot and structure. Once the trajectory is fully tested from the base station, the sequence of values are stored in the robot firmware such that the base station can only send a macro command to the robot to execute this sequence automatically.

A. Locomotion

Fig. 5 shows the complete set of achievable locomotion configurations with the current SOLL-E robot. The "stand" configuration indicates the grippers are on x-axis or y-axis adjacent voxels, but can be separated in the z-axis one unit up or down. The "step" configuration indicates a space of $1u$ in the x-axis or y-axis with the same convention for the z-axis. The "balance" configuration is added to allow the robot to balance with one foot on a single voxel. This configuration minimizes main joint holding torques when the robot performs a yaw motion.

Based on these locomotion configurations we generate a sequence that moves from one configuration to another as *solle0_stand_to_step_wA*, where *solle0* is the robot designation, *stand* is the initial configuration, *step* is the target configuration, and *wA* represents which foot moves. This sequence includes the gripper commands: gripA, ungripA, gripB, ungripB, gripC, and ungripC.

B. Grab

Grab trajectories utilize two SOLL-Es: cargoSOLL-E carries the voxel from the base station to the build front, then hands it off to craneSOLL-E, which places it at the build location. CargoSOLL-E moves into position by performing a series of locomotion trajectories which bring it in proximity to craneSOLL-E. CraneSOLL-E maneuvers, also via locomotion trajectories, into a balanced position in readiness

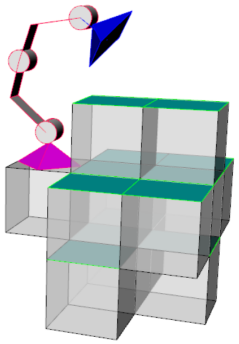


Fig. 6: CraneSOLL-E placement map from the balance configuration. Each cube with a green face represents a possible placement location of a voxel relative to craneSOLL-E's location. (GripA = purple, GripB = blue).

to initiate a grab. CraneSOLL-E is then commanded through intermediate joint angles needed to position the suspended gripper on the cargoSOLL-E voxel and secure it with the gripper. There are seven voxel grab configurations based on the handoff position of the cargoSOLL-E. After grabbing the voxel off of cargoSOLL-E's backpack, craneSOLL-E returns to the balance configuration while holding the voxel in its foot.

C. Placement

Fig. 6 shows all possible craneSOLL-E placement locations. For the placement trajectories, three approach directions – straight, CW, and CCW – must be considered due to the presence of adjacent voxels. The initial configuration of a placement trajectory starts in a balanced position with the voxel gripped by the suspended gripper. The joints are then commanded through the waypoints needed to set the voxel down in the desired location.

VI. Experimental Results

Ground testing experiments involved assembling a shelter structure that utilized two SOLL-Es and one internal bolting robot (MMIC-I). CargoSOLL-E continuously transported voxels to craneSOLL-E, which grabbed and placed them according to the build sequence. The initial seed structure was composed of a line of five voxels, where two SOLL-Es were resting in the stand configuration spaced one voxel apart. The operator manually loaded each new voxel onto cargoSOLL-E at the depot, but the remaining operations, including locomotion, grab, placement, and bolting, were completed autonomously. It took on the order of 20 seconds for SOLL-E to move one unit step forward.

The ground testing involved building a structure composed of 164 voxels ($5u$ height) to increase the confidence in autonomous surface or orbital space assembly. The build plan of the final structure had 4712 cargoSOLL-E moves and 1484 craneSOLL-E moves. The current ARMADAS system is capable of building arbitrarily sized structures autonomously.

During the experiment, cargoSOLL-E performed under high workloads during numerous stair case climbing motions. This caused excess wear on the belts of Joint1 and Joint3, eventually necessitating the replacement of the belts. Although the expected belt torque of 16 Nm was under the maximum load of 21 Nm, dynamic motions from compliance in the robot and voxel induced inertial loads exceeding the belt force limits. In addition, the current sensing of the gripper servos was found to be unreliable in detecting a successful gripped status. Since false gripping could lead to crashes and falls, a limit switch was integrated onto each gripper to determine if the gripper successfully closed.

VII. Conclusions

This paper presented the design of the SOLL-E robot, which was used to demonstrate autonomous assembly of lightweight building blocks for the ARMADAS project. SOLL-E robots are able to walk on the outside of the structure to transport voxels from a depot to the build front and can grab and place voxels according to the build sequence. The ground experiment of building a 164 unit structure showed that ARMADAS overcomes the scalability issue in assembly automation systems. Future work includes integration of a battery to remove the tethered power line, designing mechanisms to load voxels autonomously from the base station, and advancing the high-level autonomous path-planning algorithm for multiple sets of robots.

References

[1] W. Doggett, J. Dorsey, and D. Kang, "State of the Profession Considerations: NASA Langley Research Center Capabilities and Technologies for Large Space Structures, In-Space Assembly and Modular Persistent Assets." in AIAA Journal, 2019.

[2] S. Scheringer, J. Weinland, R. Wilbrandt, P. Backer, A. Roennau, and R. Dillmann, "A walking space robot for on-orbit satellite servicing: the ReCoBot," in *Proc. 2022 IEEE Int. Conf. Autom. Sci. Eng.*, Mexico City, Mexico, Aug. 2022, pp. 2231–2237.

[3] M. Carney and B. Jenett, "Relative Robots: Scaling Automated Assembly of Discrete Cellular Lattices," in*Int. Manu. Sci. and Eng. Conf.* vol. 49903, Virginia, USA, Jun. 2016. doi: 10.1115/MSEC2016-8837.

[4] B. Bernus, G, Trinh, C. Gregg, O. Formoso, and K. Cheung. "Robotic specialization in autonomous robotic structural assembly." in *IEEE Aero. Conf.*, Big Sky, USA, Mar. 2020.

[5] O. Formoso, G. Trinh, D. Catanoso, I.-W. Park, C. Gregg, and K. Cheung, "MMIC-I: a robotic platform for assembly integration and internal locomotion through mechanical meta-material structures," in *2023 IEEE Int. Conf. on Robots. and Autom.*, London, UK, May 2023, pp. 7303-7309.

[6] D. Catanoso, I.-W. Park, . Olatunde, O. Formoso, G. Trinh, C. Gregg, E. Taylor, M. Ochalek, and K. Cheung, "Rapid lightweight firmware architecture of the mobile metamaterial internal co-integrator robot," in *IEEE Aero. Conf.*, Big Sky, USA, Mar. 2023.

[7] C. Gregg, J. Kim, and K. Cheung, "Ultra-light and scalable composite lattice materials," Adv. Eng. Mater., vol. 20, 2018, doi: 10.1002/adem.201800213.

[8] M. Ochalek, B. Jenett, O. Formoso, C. Gregg, G. Trinh, and K. Cheung, "Geometry systems for lattice-based reconfigurable space structures." in *IEEE Aero. Conf.*, Big Sky, USA, Mar. 2019.

[9] Y. Terada and S. Murata, "Automatic assembly system for a large-scale modular structure," in *Proc. 2004 IEEE/RSJ Int. Conf. Intell. Robots Sys.*, Sendai, Japan, Sep. 2004, pp. 2349–2355.

[10] B. Jenett, A. Abdel-Rahman, K. Cheung, and N. Gershenfeld, "Material-robot system for assembly of discrete cellular structures," IEEE Robotics and Autom. Letters, vol 4, no. 4, 2019, pp. 4019–4026.

[11] P. M. Wensing, A. Wang, S. Seok, D. Otten, J. Lang, and S. Kim, "Proprioceptive actuator design in the MIT cheetah: impact mitigation and high-bandwidth physical interaction for dynamic legged robots," IEEE Trans. Robotics, vol. 33, no. 3, 2017, pp. 509–522.