

Self Assembly of Modular Manipulators with Active and Passive modules

Seung-kook Yun*, Yeoreum Yoon, Daniela Rus[†]

*Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology, Cambridge, Massachusetts, USA*

*yunsk@mit.edu, [†]rus@csail.mit.edu

Abstract—We propose a vision of self-assembling robot systems composed of active modular robots and passive bars. As the first step of realizing the long-term vision, 3DOF robots with rotating grippers and passive struts with embedded IR LEDs are designed and working in 3D structural bars. We describe hardware experiment with two active modules and one passive bar in a distributed fashion. They interact with each other and are combined into a self-assembled 6DOF manipulator, which executes the tasks that are hardly achievable by only one robot.

I. INTRODUCTION

Recently, a modular robot has had great attention because of mass productivity, robustness to a fault, and its high adaptability to an uncertain environment. This adaptability requires a variety of configurations from many modules, in which modular robots are physically connected and may communicate with each other. Our long-term vision is a active structure consisting of heterogeneous modules with active robots and passive structural modules.

In this paper, as the first stage of the long-term goal, we focus on the particular idea of building a 6-DOF manipulator with two active modules and a passive bar, with an emphasis on the possibility only producing the latter—much simpler—units from the environment. The main challenges include simple and robust hardware design of the active and passive modules, and control and planning algorithm.

Our work on self-assembling modular manipulators is closely related with systems based on rotary DOF and mechanical connection mechanisms. The following works are good examples:

Murata, Kurokawa, et al’s “3D Fracta” [1]; Kotay and Rus’ “Molecule” [2]; Unsal, Kiliccote, and Khosla’s bi-partite “I-Cubes” [3] system; Duff, Yim, et al’s PolyBot [4]; and Lund, Beck, Dalgaard, Støy et al’s ATRON [5].

Our proposing system has a big difference in that the modular manipulators consist of not only active modules—robots— but also passive structural elements. On the contrary, all of the above referenced systems are either homogeneous (all modules identical and actuated) or are heterogeneous but still require actuation in all modules.

Many truss climbing robot systems have been developed as following examples:

e.g. Staritz et al’s “Skyworker” [6], Amano et al’s handrail-gripping robot for firefighting [7], Ripin et al’s pole climbing

robot [8], Nechba, Xu, Brown et al’s “mobile space manipulator SM2” [9], Kotay and Rus’ “Inchworm” [10], and Almonacid et al’s parallel mechanism for climbing on pipe-like structures [11].

Our proposed system has a particularity in that our Shady3D is a self-reconfiguring/self-assembling module and it extends its functionality by self-assembly.

In this research, we show a vision of building self-assembled active linkages with active and passive modules. We propose a simple design of a truss climbing robot with the minimal design for 3D movement, and a passive strut that is able to notify itself to a robot. Configurations from the active and passive modules are discussed to build a 6DOF manipulator with minimal number of units, and their inverse kinematics is briefly analyzed. To control the robots in a distributed way, an intuitive algorithm is proposed and implemented with the hardware system, given a simple pick and drop task.

II. SELF-RECONFIGURING MODULAR MANIPULATOR: SHADY3D

As a modular manipulator for self-assembly, we have been developing a robot, named Shady3D, which moves on the trusses. It has three joints for 3-d motion and two grippers on each side as shown in Figure 1. The number of joints is chosen to be minimal for moving on the 3-D trusses. Unlike Shady [12] that only works on a 2-D plane, the middle joint enables a robot to switch from one plane to another as Figure 1(b). Note that a robot can only move on the specific points on the trusses and every robot has the identical structure and functions.

A. Self-assembly to an active tower

Like fascinating 2-dimensional self-assembly by multi-module of Shady [12], we want multi-units of the Shady3Ds to co-operate and to construct a self-assembling truss structure including active and passive components in the 3-D space. Figure 2 and 3 illustrate the concept of the self-assembly structure, which is composed of several Shady3Ds and short trusses (passive bars) to connect them.

Figure 2 shows snapshots of the self-assembly of a truss tower. Twelve active modules and eight passive bars are employed to build a three-dimensional tower. The tower building is performed through the following steps.

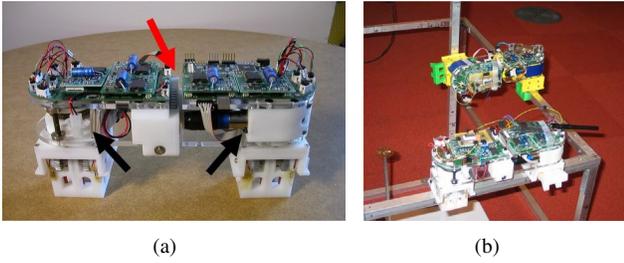


Fig. 1. (a) Shady3D and its structure: 3-joints and 2 grippers (b) Shady3Ds are moving on the trusses

- Two active modules form a six-degree-of-freedom manipulator by connecting to each other using a passive bar. Eight active modules form four manipulators. These manipulators can reach arbitrary position and orientation.
- Four 6-DOF manipulators move to the base location of the tower and approach remaining four active modules to pick them up. Remaining modules hold passive bars to connect to the 6-DOF manipulators.
- The 6-DOF manipulators connect to four remaining active modules. The active modules held by the manipulators release the grippers gripping the ground trusses. Thus, each 6-DOF manipulator becomes a structure consisting of three active modules and two passive bars.
- The four structures formed in the previous stage arrange themselves in the desired poses. Then, they connect to their neighbors to complete the tower structure.

Figure 3 demonstrates the tower moving procedure. The tower built in the previous simulation is moved to the neighbor grid on the ground trusses. This is done through similar procedures with the tower building.

- The upper part of the tower tilts toward the direction in which it moves. For this, two legs in diagonal position rotate so that all gripper joints in the middle of legs have parallel joint vectors.
- The legs of the tower move to their desired position on the next grid. One leg is moved at a time to maximize the stability of the structure during the process.
- After all legs have moved to the next grid, gripper joints in the middle of legs are rotated synchronously to recover the original shape of the tower.

B. 6DOF from two robots and one passive bar

To realize such simulations as the active tower, we examine what we get from combining multi-robots and how to effectively control each joint of the modules when more than one modules are combined. In this paper, we focus on making a 6-DOF manipulator out of multi Shady3Ds because 6-DOF is the minimum to span an arbitrary posture in 3D. We will prove that the developed structure of the Shady3D has a minimal design to build a 6-DOF manipulator out of two modules. We also propose a new inverse kinematics of the self-assembled manipulator, modified from a damped least square method, to control a self-assembled modules in

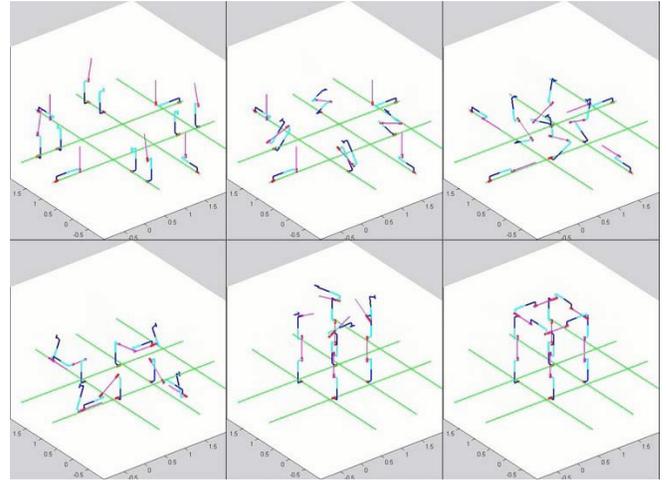


Fig. 2. Six snapshots of the tower building simulation. Shady3D modules are drawn as an elongated U-shape with light and dark halves; free bars and the grid are drawn as straight segments.

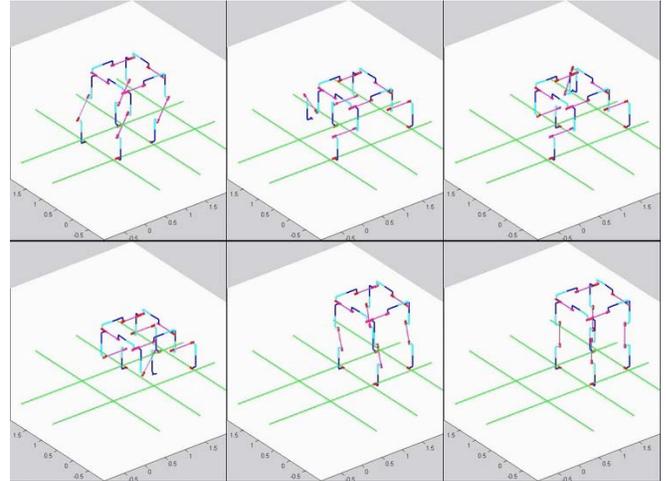


Fig. 3. Six snapshots of the tower moving simulation.

3D. As we have seen, a Shady3D has three joints which are chosen as the minimal number to work on the 3D trusses. There is another reason to choose the 3joints, for it also turns out the least number of joints that is required for 6-DOF when two modules and a passive strut are combined. As shown in Figure 4(a), simply connecting two Shady3Ds gives only 5-DOF because of one DOF reduction at the connection of two grippers. The reason is that the axes of two gripper joints lie on the same line. This problem can be solved by introducing a free truss element as a medium of connection as in Figure 4(b). In this configuration, two robots are grasping the different point of the free truss and the reduction of DOF does not happen. We have compared a few design models with the various DOF [13], and it tells that two 4-DOF models result in redundancy while 2-DOF ones gives only 4-DOF motion even if they are combined with a passive bar.

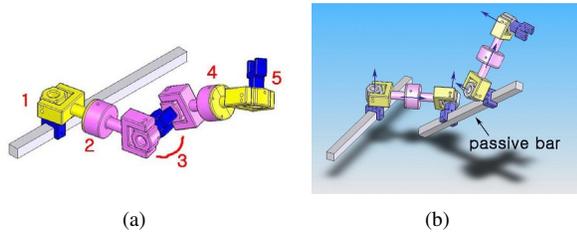


Fig. 4. (a) A 5-DOF manipulator with directly connected two Shady3Ds (b) A 6-DOF one with inserting a passive bar between two robots [13]

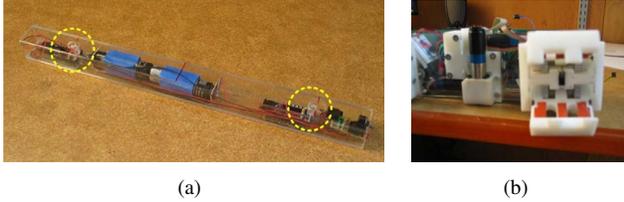


Fig. 5. (a) A passive bar with embedded IR LEDs (b) an IR sensor attached beneath the gripper

III. HARDWARE IMPLEMENTATION

To prove the proposed concepts, self-assembly and a few exemplary tasks are implemented with two Shady3Ds. Two robots and one passive bar with embedded LEDs are combined into a 6DOF manipulator by active sensing. Upon self-assembly, a pick and drop task is given to the manipulator and executed in a distributed way.

A. Set-up

As we mentioned in the early sections, a passive component is essential to build a 6DOF manipulator. We design a passive bar to emit IR signal by the IR LED embedded in the bar as shown in Figure 5(a). Two LEDs are located at each side of the bar (denoted by yellow dotted circles), and they inform a robot about existence of the bar. It includes two AAA batteries as a power source. A Shady3D has two IR sensors on each gripper bed like Figure 5(b) so that it can check whether a bar exists just below its gripper or not. In our experiment, it can sense the bar located about 50(cm) below the gripper. Note that this sensing range matters when a combined 6-DOF manipulator also try to find the bar, because it is capable of moving any direction while a single module can not.

To sense the bar, we assume the each robot knows information about the passive bars. The data is composed of two approachable nodes from where a robot can grasp the bar, and an approach vector to which a robot can find the bar at the approachable nodes. Therefore, a robot is able to know where to find a bar and it can confirm it by sensing the IR signal. Note that it cannot be sure there is a bar even if it knows its location, because every robot is supposed to work in a distributed fashion and another robot might have got the bar and be gone to somewhere before it reached there.

B. Distributed Control Algorithm

In the following tasks, the whole information is given to the both robots so that two robots move in a distributed fashion. Each robot decides which role it should play based on the task specification and its location.

All the tasks starts from building a 6DOF manipulator by self-assembly. The way they build it is addressed in Algorithm 1, and its implementation is shown in Figure 6(a, b, c). Firstly, given a specified position for the passive bar within the Shady3D experimental environment, each Shady3D module positions itself optimally so as to be able to reach the bar. Details of the optimal deploying algorithm is addressed in [14]. In the first step of the algorithm each Shady3D module moves independently and in parallel to reach and grasp the bar. The bar is detected using the LED sensors within the Shady3D grippers. Upon grasping the bar, the Shady3D modules signal to each other to coordinate the completion of the grasping step and the self-assembly of a 6DOF manipulator

Each robot always has its *status* to show other robots what it is doing. For self-assembly and tasks we have implemented, we use the following states:

- *Idle*: waiting for a command
- *Moving*: motors are working
- *Assembling*: doing self-assembly now
- *Assembled*: assembled with other robots

Algorithm 1 Self-assembly to a 6DOF manipulator

```

1: State = Moving
2: Move to the nearer approachable node
3: State = Idle
4: while the other approachable node is empty do
5:   Communicate with the adjacent robots
6: end while
7: State = Assembling
8: Rotate to check the passive bar
9: if IR sensor = ON then
10:  Grasp the bar
11:  State = Idle
12:  while The other robot's State = Assembling do
13:    Delay
14:  end while
15:  State = Assembled
16: else
17:  Rotate back to the original position
18:  State = Idle
19: end if

```

After self-assembly to a 6DOF manipulator, we assign two roles for the robots: a root and a leaf, where the former has a fixed gripper at the truss while the latter is only supported by the other robot and a free bar. The fixed gripper of a root is called an anchor and the free gripper of a leaf is an end-effector.

Each task is a stack of the command sets for the two robots, and how a robot execute the task is shown in

Algorithm 2. Each robot reads the next queue of the stack, which has a command set of the following information:

- *root node*: a node on the truss where the root should be anchored
- *displacement*: 6 joint movements of two robots/end-effector displacement in 3D
- *grasp*: grasping/release of the end-effector

Based on the command, a robot finds out if it is a root. The root robot calculates the joint displacement of two robots directly or indirectly by the inverse kinematics, while the leaf one is waiting for a command. The root sends the corresponding joint displacements to the leaf robot, and they both execute their own command. The root checks if they have finished the commands, and they pops the next command set until the stack is empty. Actual communication is implemented via Bluetooth.

Algorithm 2 Task execution

```

1: while Task Stack not empty do
2:   Pop the next queue
3:   if Anchor = Root then
4:     Get the commands from the queue
5:     Send the command for the leaf
6:     State = Moving
7:     Execute my command
8:     while The leaf's State = Moving do
9:       Delay
10:      State = Assembled
11:    end while
12:  else
13:    Wait for the command from the root
14:    State = Moving
15:    Execute my command
16:    State = Assembled
17:  end if
18: end while

```

C. pick and drop by forward kinematic control

The first task is a *pick and drop*, which requires a 6DOF manipulator. The locations of pick and drop are given by sets of the joint angles so that Shady3Ds moves by forward kinematic control.

The task stack is composed of 7 command sets each of which has one joint displacement or grasping/release. As the task starts, one of the modules releases its grasp of the environment. This 6DOF manipulator is controlled using forward kinematics by providing joint angles in the command sets. The two modules are controlled independently and in parallel to demonstrate the movement of the arm (see Figures 6 (d, e, f).) Finally, an additional bar is manually presented to the free gripper of the 6DOF manipulator, grasped, transported, and dropped at a specified location (see Figure 6 (g, h, i).) We have performed this experiment 10 times in a row during the course of one hour. Each experiment consisted of 9 joint movements and 5 grasping/release operations and it takes

about 4 minutes. All the control steps succeeded for all the experiments. However, due to a hardware failure at the end of the 7th experiment one of the gripper motors had to be replaced.

IV. CONCLUSIONS AND FUTURE WORKS

In this research, we discussed our vision of building a self-assembly robot composed of passive components as well as modular manipulators. As the first stage of a proof of the concept, we have designed a module with the minimal number of joints for 3D movement and building a 6DOF manipulator, which can produce arbitrary motion, by combining two modules and one passive bar with embedded IR LEDs. We have also proposed the inverse kinematics to control multi-robot in 3-D space without serious position error and long convergence time. Hardware implementation of building a 6-dof manipulator and the pick and drop task shows how the proposed self-assembly works in the real world.

There are much works to be done in future, as we need a better hardware design and a high-level distributed planning algorithm.

V. ACKNOWLEDGMENTS

This work was supported by NSF and Samsung Scholarship to Mr. Yun. We are grateful for this support.

REFERENCES

- [1] Satoshi Murata, Haruhisa Kurokawa, Eiichi Yoshida, Kohji Tomita, and Shigeru Kokaji, "A 3-d self-reconfigurable structure," in *Proceedings of the 1998 IEEE International Conference on Robotics and Automation*, Leeuven, Belgium, May 1998, pp. 432–439.
- [2] Keith Kotay, "Self-reconfiguring robots: Designs, algorithms, and applications," Ph.D. dissertation, Dartmouth College, Dec. 2003.
- [3] Cem Unsal, Han Kiliccote, and Pradeep Khosla, "A modular self-reconfigurable bipartite robotic system: Implementation and motion planning," *Autonomous Robots*, vol. 10, no. 1, pp. 23–40, Jan. 2001.
- [4] David G. Duff, Mark Yim, and Kimon Roufas, "Evolution of polybot: A modular reconfigurable robot," in *Proceedings of the Harmonic Drive International Symposium*, Nagano, Japan, Nov. 2001.
- [5] Kasper Støy, "The ATRON self-reconfigurable robot: challenges and future directions," Presentation at the Workshop on Self-reconfigurable Robotics at the Robotics Science and Systems Conference, July 2005.
- [6] Peter J. Staritz, Sarjoun Skaff, Chris Urmsen, and William Whittaker, "Skyworker: A robot for assembly, inspection and maintenance of large scale orbital facilities," in *IEEE ICRA*, Seoul, Korea, 2001, pp. 4180–4185.
- [7] Hisanori Amano, Koichi Osuka, and Tzyh-Jong Tarn, "Development of vertically moving robot with gripping handrails for fire fighting," in *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Maui, HI, 2001, pp. 661–667.
- [8] Zaidi Mohd Ripin, Tan Beng Soon, A.B. Abdullah, and Zahurin Samad, "Development of a low-cost modular pole climbing robot," in *TENCON*, vol. I, Kula Lumpur, Malaysia, 2000, pp. 196–200.
- [9] Michael Nechyba and Yangsheng Xu, "Human-robot cooperation in space: SM2 for new space station structure," *IEEE Robotics and Automation Magazine*, vol. 2, no. 4, pp. 4–11, Dec. 1995.
- [10] Keith D. Kotay and Daniela L. Rus, "Navigating 3d steel web structures with an inchworm robot," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, 1996.
- [11] M. Almonacid, R. J. Salterén, R. Aracil, and O. Reinoso, "Motion planning of a climbing parallel robot," *IEEE Transactions on Robotics and Automation*, vol. 19, no. 3, pp. 485–489, 2003.
- [12] C. Detweiler, M. Vona, K. Kotay, and D. Rus, "Hierarchical control for self-assembling mobile trusses with passive and active links," in *IEEE Intl. Conf. on Robotics and Automation*, 2006, pp. 1483–1490.
- [13] Y. Yoon, "Modular robots for making and climbing 3-d trusses," Master's thesis, Massachusetts Institute of Technology, Cambridge, 2006.

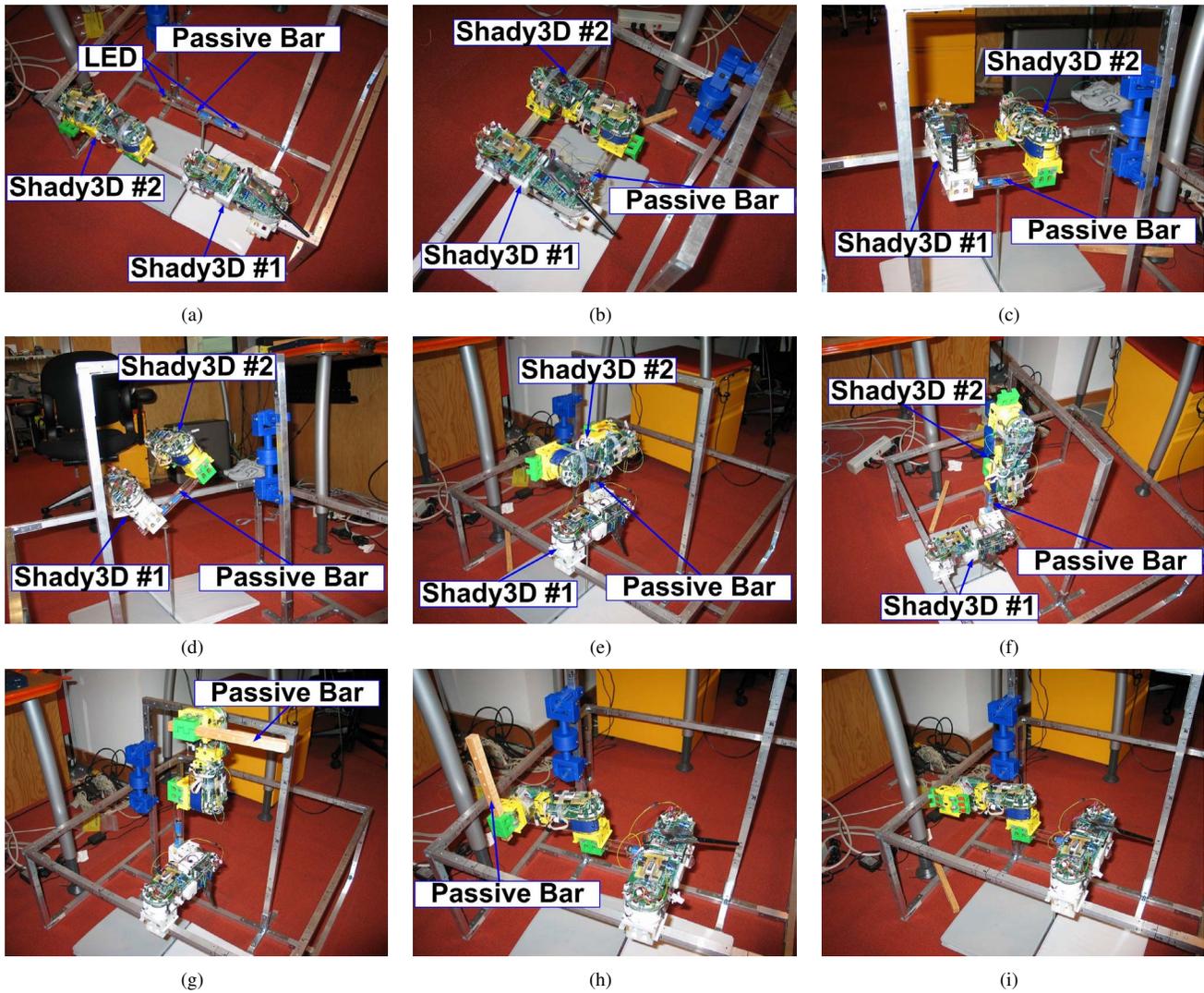


Fig. 6. Implementation of self-assembly of 6DOF modular arm and an example of moving a bar. (a) Two robots have moved to the approachable nodes. (b) They are swinging their body to find the bar. (c) They have grasped each side of the bar. (d) A 6-dof manipulator combined by two modules and the base module pulls the other upward. (e) The base module has fully pulled the other. (f) The manipulator is fully stretched to the maximum height. (g) It is gripping a manually given bar to be moved. (h) The bar is moved to the dropping position. (i) The manipulator has dropped the bar.

[14] S. kook Yun and D. Rus, "Optimal distributed planning of multi-robot placement on a 3d truss," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2007.