Evaluation In Vitro of a Treatment Planning Algorithm for an Epicardial Crawling Robot

Brina E. Goyette, Brian C. Becker, Marco A. Zenati and Cameron N. Riviere

Abstract— **HeartLander is a small, mobile robot designed to assist with surgical procedures on the surface of the heart. It crawls within the pericardial sac surrounding the heart. Numerous potential clinical uses for HeartLander involve injections or other interventions at multiple locations on the epicardial surface. To minimize treatment time, we have developed an algorithm that optimizes a plan for reaching a given set of treatment targets. Results from *in vitro* evaluation on a beating heart model show improvement over a simple greedy technique.**

I. INTRODUCTION

HEARTLANDER is a small, mobile robot that has been designed to assist with surgical procedures on the surface of the heart (the epicardium). It is inserted through a subxiphoid incision (below the sternum), and then through a small incision in the pericardium, the sac surrounding the heart. This technique is less invasive than traditional laparoscopic techniques, which require the left lung to be collapsed in order to access the heart behind it. By attaching to the epicardium, HeartLander passively compensates for the movement of the heartbeat. These advantages allow the patient to breathe normally, and obviate cardiopulmonary bypass, potentially obviating general anesthesia as well.

HeartLander has two body segments, or feet, which attach to the epicardium using suction. Using flexible push-wires connected to offboard stepper motors to modulate the distance between the feet, and alternating suction between them, HeartLander achieves an inchworm-like locomotion [1]. This allows HeartLander to move to various treatment sites on the heart, where it can administer treatments such as injection, pacemaker lead placement, and tissue ablation for cardiac resynchronization [2]. Such treatments often involve reaching multiple sites on the heart.

A magnetic tracker provides information about HeartLander's position and orientation with six degrees of freedom, allowing movement around the heart to be coordinated. A tube running from outside the body through the front foot provides a channel through which the treatments are deployed.

HeartLander can coordinate its motion in two ways. In regular locomotion, the back foot attaches to the heart using suction, and the front foot is left free. The stepper motors push the front foot forward. The front foot applies suction, and the back foot is released. This allows the stepper motors to pull the back foot up to the front foot. This completes one step. In order to turn, one wire is extended farther than the other. Using this locomotion, HeartLander can travel to any position on the heart surface.

Because of the friction forces HeartLander faces inside the pericardial sac, some amount of slippage is encountered during regular locomotion, which affects efficiency and accuracy. In Fine Positioning locomotion, the back foot remains adhered to the epicardium, while the front foot reaches for a treatment site. The adhered back foot adds greater stability, and allows HeartLander to reach nearby treatment sites with greater accuracy [1].

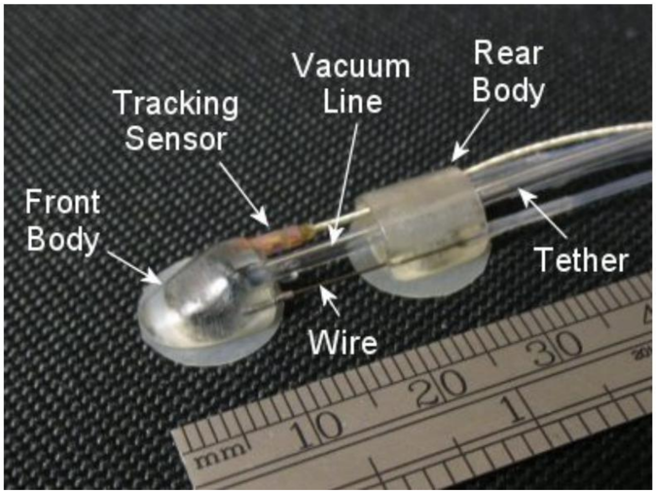

Fig. 1. A view of the HeartLander crawling robot

II. METHODS

The slippage experienced with regular locomotion increases procedure times, which increases treatment cost and influences patient outcomes. Combining both types of locomotion, and maximizing the use of Fine Positioning can decrease procedure times. We have developed an algorithm that first finds the smallest set of points, referred to as Base Locations, on the heart surface that allows HeartLander to use Fine Positioning to reach all the required treatment points, and secondly finds the shortest path from the apex of the heart (where HeartLander is inserted), to each point in the set, and back to the apex for removal. To address these problems, we look to the literature on the Facility Location Problem [3] and the Traveling Salesman Problem [4]

This work was supported by the NIH under Grant R01 HL078839.

B. E. Goyette, B. C Becker, and C. N. Riviere are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: camr@ri.cmu.edu).

M. A. Zenati is with the Division of Cardiothoracic Surgery, University of Pittsburgh, Pittsburgh, PA 15213 USA.

Preprint submitted to 32nd Annual International IEEE EMBS Conference.
Received April 1, 2010.

respectively, using the linear programming solver *lpsolve* [5] to find the solutions.

A. Selection of Base Locations

The Facility Location Problem was solved in order to determine the base locations from which HeartLander used Fine Positioning to reach each treatment site. The problem deals with the problem of designating optimal warehouse (or facility) locations for a set of given stores (or sites). There are many possible facility locations, each with an associated building cost and service cost to each individual site. The Facility Location Problem selects the set of facilities that serve each site exactly once, with the lowest combination of building and service costs.

The Facility Location Problem can be solved as a Mixed Integer Linear Program (MILP) [6]. Given a set of variables and a set of constraints, a MILP solver finds the optimal values for each of the variables. To define the Facility Location Problem as a MILP, we define a set of variables $F = (f_1, f_2, ..., f_n)$ to represent our possible facilities, and a set of values $Y=(y_1, y_2, ..., y_n)$ to be the associated building costs for each facility. The MILP solver will set $f_x = 1$ if facility x is to be created, and $f_x = 0$ if it is not. The cost for facility i to serve site j is defined as c_{ij}. The MILP solver will set $x_{ij} = 1$ if facility i is to serve site j.

The constraints are defined to find the cheapest solution such that each site is served by exactly one facility, and that facility is one that will be created. This is defined formally as follows.

$$\begin{gathered}\min \sum_{i\le m} f_i y_i + \sum_{i\le m}\sum_{j\le n} c_{ij} x_{ij} \\ s.t. \\ \sum_{i\le m} x_{ij} = 1, \forall j \le n \\ x_{ij} \le y_i, \forall i \le m, \forall j \le n \\ x_{ij}, y_i \in \{0,1\}, \forall i \le m, \forall j \le n\end{gathered}$$

In the case of HeartLander, the sites are the treatment sites required for the procedure. The facility locations are the places on the heart surface that HeartLander can stop and use Fine Positioning from. In reality, those locations are continuous, but for the purposes of the algorithm, the locations are discretized. The fine positioning motion is the method of serving a site. The cost of reaching site j from facility i is determined by an equation based on both the distance HeartLander much stretch to reach the site from the facility point (d_{ij}), and the angle with which it much reach (θ_{ij}). Based on empirical observations of HeartLander's fine positioning, we used the cost function $c_{ij} = 7.5\theta_{ij} + 2.5d_{ij}$. The angle is weighted more heavily because is has a stronger influence on the speed and accuracy of the fine positioning than the distance. If either the angle or the distance is so great that it is beyond HeartLander's physical limitations, $c_{ij} = \infty$, ensuring that facility i will never be chosen to serve site j. To reduce the amount of regular locomotion HeartLander must use, we want to minimize the total number of facilities created. To achieve this, we set the values in Y to be high enough that it is not preferable to create extra facilities over using a high service cost. Each facility is given the same cost.

B. Order of Base Locations

Once the facility locations are established, the problem of determining the order in which HeartLander will reach them is formulates as a Traveling Salesman Problem. This problem looks to find the optimal tour from a salesman's home, to each city in which he has business, and back to his home. The cost of the tour can be travel time, distance, or price. It can also be solved as a MILP. We define the cost between city i and city j to be c_{ij}, for $i, j < n$, where n is the number of cities. The MILP solver sets the variable $x_{ij} = 1$ if we choose to travel from city i to city j. The variable u is used to ensure that our tour is continuous, rather than a number of small, separate loops. The other constraints are defined such that we find the cheapest solution, while ensuring that we visit each city exactly once.

$$\begin{gathered}\min \sum_i \sum_j c_{ij} x_{ij} \\ s.t. \\ x_{ij} \ge 0, \forall i, j \\ x_{ij} \le 1, \forall i, j \\ \sum_{j \ne i} x_{ij} = 1, \forall i \\ u_i - u_j + n x_{ij} \le n - 1, \forall i \in \{0,1,...,n-1\}, j \in \{1,2,...,n-1\}\end{gathered}$$

In the case of HeartLander, the cities are the base locations that were determined with the Facility Location Problem. The start and end location is the apex, the bottom of the heart where HeartLander is inserted. Because horizontal motion with HeartLander is more difficult and requires more time, tours involving much horizontal motion needed to have a higher cost then those that did not. Therefore, the cost function is defined as $c_{ij} = d_{ij} + 10\ h_{ij}$, where d_{ij} is the distance between city i and city j, and h_{ij} is the horizontal component of that distance.

C. Other Considerations

By attaching to the surface of the heart, HeartLander moves along with the heart, passively compensating the beating motion. This is advantageous when administering treatments, because HeartLander does not need to actively predict the motion of the heart beat and move along with it. However, the signal from the magnetic tracker that is used to coordinate HeartLander's locomotion on the heart reflects the movement of the heartbeat. Because the treatment sites on the heart are defined statically, this artifact must be removed in order for HeartLander to reach targets accurately.

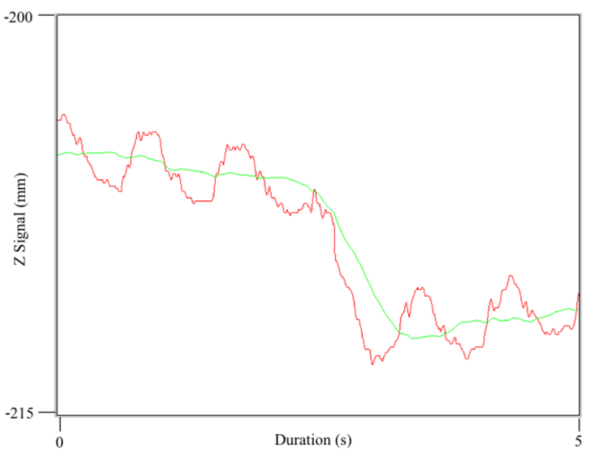

Fig. 2 The Z signal from the magnetic tracker. HeartLander begins at rest of the heart, then takes a step, and is at rest again. The unfiltered signal is in red, and the filtered signal is in green.

To remove the noise from the heartbeat artifact from the tracker signal, we applied a third-order Chebyshev Type II low-pass filter, with a stopband cutoff of 20 dB at a frequency of 1.0 Hz to the tracker signal. The filter can attenuate the noise from the heartbeat to levels similar to the noise from the tracker itself, and introduces a minimal delay.

Experiments were conducted on a Chamberlain Group no. 1008 heart model, which uses compressed air to create a simulated heartbeat. This model is fitted with a fabric cover to simulate the pericardium. Treatment points were defined on a 3D model generated using CT data from a scan of the heart model [7].

We evaluated our algorithm against a simple greedy approach. This approach did not take advantage of fine positioning, or weighted distances. It simply chose the closest site to the previously chosen site, until all sites had been visited, starting and ending with the apex.

The treatment sites were defined in three patterns. In the first pattern, a set of treatment points was randomly defined on the surface of the heart. This pattern is not typical of a real surgical procedure. The second pattern simulated the treatment of the perimeter of an area of damaged tissue, while the third simulated the treatment of the entire surface of an area of damaged tissue. Each pattern was attempted both with the greedy algorithm and our algorithm, for a total of six experiments per set, and three full sets of experiments were run. Simulations of these experiments had been run previously, suggesting that procedure times would decrease with our algorithm [8].

TABLE I
PLANNING ALGORITHM VS. GREEDY APPROACH

	Random	Perimeter	Area
Number of Treatment Sites	13	21	26
Number of Facilities	10	13	12
Error: Plan (mm)	2.4 ± 0.0	1.8 ± 0.1	2.1 ± 0.2
Error: Greedy (mm)	2.1 ± 0.0	2.1 ± 0.4	2.1 ± 0.5
Time: Plan (min)	19.3 ± 3.3	17.8 ± 3.1	19.3 ± 5.3
Time: Greedy (min)	25.3 ± 11.8	20.9 ± 2.7	37.7 ± 10.8
Mean Time Decrease	31.11%	20.97%	31.05%

The time for each experiment was measured from the beginning of locomotion at the apex until HeartLander's return to the apex after reaching each treatment site. Time to administer a treatment was not included.

III. RESULTS

The results in Table I show that the algorithm does not improve procedure times for the random pattern. This is expected, as most of the treatment sites are spread farther apart. HeartLander can only take advantage of fine positioning locomotion when treatment sites are within one step of each other. Because the random pattern is of little clinically relevance, this is not of great concern. In the pattern that simulates treatment of the perimeter of an area of damaged tissue, the treatment sites are closer together, and the advantage of the algorithm begins to emerge. This advantage is even more apparent for the pattern that simulates treatment of the entire surface of an area of damaged tissue.

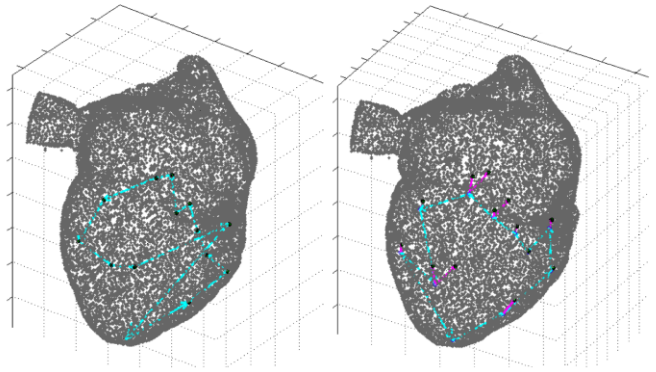

Fig. 3. Path plans for treating a random pattern. The plan created by the greedy algorithm is on the left, and the plan created by our algorithm is on the right. Grey shows all potential base locations, regular locomotion is shown in cyan, and fine positioning is shown in magenta.

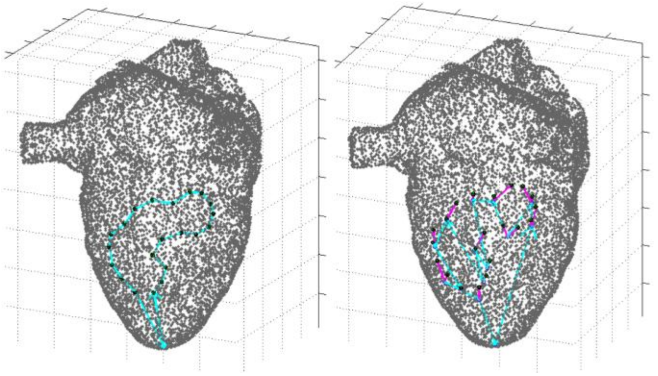

Fig. 4. Path plans for treating a pattern simulating the treatment of the perimeter of an area of damaged tissue. The plan created by the greedy algorithm is on the left, and the plan created by our algorithm is on the right. Grey shows all potential base locations, regular locomotion is shown in cyan, and fine positioning is shown in magenta.

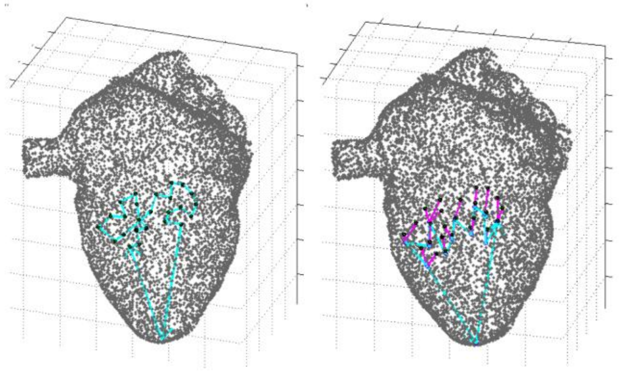

Fig. 5. Path plans for treating a pattern simulating the treatment of the entire surface of an area of damaged tissue. The plan created by the greedy algorithm is on the left, and the plan created by our algorithm is on the right. Grey shows all potential base locations, regular locomotion is shown in cyan, and fine positioning is shown in magenta.

IV. DISCUSSION

Decreasing procedure times results in faster post-operative recovery, and decreased operating room costs. By optimizing the treatment plan using a combination of the Facility Location Problem and the Traveling Salesman Problem, the algorithm presented herein achieved shorter treatment times were attained with a simply greedy approach, while maintaining roughly equal positioning accuracy. Future work will involve experimental verification *in vivo* in a porcine model.

REFERENCES

[1] N. A. Patronik, T. Ota, M. A. Zenati, and C. N. Riviere, "A miniature mobile robot for navigation and positioning on the beating heart," *IEEE Tran Robotics*, vol. 25, pp. 1109-1124, 2009.

[2] M. Rivero-Averza, D. A. Theuns, H. M. Gargia-Garcia, E. Boersma, M. Simoons, and L. J. Jordaens, "Effects of cardiac resynchronization therapy on overall mortality and mode of death: a meta-analysis of randomized controlled trials," *Eur. Heart J.*, vol. 27, pp. 2682-8, Nov. 2006.

[3] R. Z. Farahani and M. Hekmatfar, *Facility Location: Concepts, Models, Algorithms and Case Studies*, Heidelberg, Germany: Springer, 2009, pp. 96-99.

[4] R. A. Schweikert, W. I. Saliba, G. Tomassoni, N. F. Marrouche, C. R. Cole, T. J. Dresing, P. J. Tchou, D. Bash, S. Beheiry, C. Lam, L. Kanagaratnam, and A. Natale, "Percutaneous pericardial instrumentation for endo-epicardial mapping of previously failed ablations," *Circulation*, vol. 108, pp. 1239-35, 2003.

[5] F. Rossi, P. V. Beek, and T. Walsh, *Handbook of Constraint Programming*, Amsterdam, The Netherlands: Elsevier, 2006, pp. 542-544.

[6] M. Berkelaar, K. Eikland, and P. Notebaert, *lpsolve*, 2004.

[7] B. E. Goyette, "CT visualization and treatment planning for a surgical robot," M.S. thesis, Robotics Institute, Carnegie Mellon University, 2009.

[8] B. E. Goyette and C. N. Riviere, "Reducing operating time of a crawling robot for epicardial surgery," Proc. 32nd Northeast Bioeng. Conf., 2010.