MONTHLY REPORTING CHECKLIST

Submitted by: University of Miochigan   Report Month: DECEMBER 2002

TTP No.: ALO-7-C1-61 (UMichigan)

EARNED VALUE ANALYSIS DATA. *Check* ***one*** *box in Line A and* ***one*** *box in Line B.*

A. How is your project's schedule doing compared to your TTP baseline?

☐ 1. As planned. ☐ ☐

B. How is your project's total cost doing compared to your TTP baseline?

☐ 2. As planned ☐ ☐

PTS NARRATIVE INPUT CHECKLIST. *Check that you have prepared the following narrative inputs:*

☐ 1. SIGNIFICANT ISSUES/PROBLEMS/CONCERNS:
Note if there are any problems; otherwise, state "None." Report only problems considered "showstoppers" or fatal flaws (i.e., a lack of funding will cause the project to be shut down).

None

☐ 2. CORRECTIVE ACTION:
If a significant issue/problem/concern in Section 1 above is described, this section is required; otherwise state "None needed."

None needed

☐ 3. SUMMARY ASSESSMENT:
This should be a BRIEF paragraph summarizing the overall status of the project. This section is a synopsis of the entire report.

During this past month, work has continued in robust navigation, sensing, and radiation imaging. Development of the novel OmniPede and OmniTread vehicles is progressing well. Work using the Sick laser range scanner for obstacle avoidance will result in a better understanding of sources of artifacts from scans. Work to instantiate a new generation gamma camera is proceeding rapidly with the receipt of the readout devices, and new efforts are exploring more efficient imaging techniques.

☐ **4. COST VARIANCE:**
If you checked number 4 or 6 in the Earned Value Analysis section, ***you must provide an explanation here.*** **Explain funding issues such as variances, carryover, commitments, incorrect FIS data. Avoid using only the words "Within budget." Some narrative is preferred.**

Within budget.

☐ **5. SCHEDULE VARIANCE:**
If you checked number 3 or 5 in the Earned Value Analysis section, ***you must provide an explanation here.*** **Note if the project is on schedule, ahead of schedule, or behind schedule. If behind, explain what is being done to bring the project back on schedule.**

On schedule.

☐ 6. TECHNICAL STATUS:
This is likely to be the longest section of the narrative and describes the technical accomplishments during the reporting period. Provide enough detail to inform, yet avoid extensive details that can confuse the reader.

6.1 Obsacle avoidance

6.1.1 Outdoor mapping experiments

We conducted several outdoor mapping experiments with the Sick laser rangefinder (LRF) but only one set of the experiments produced useful data. This is because the control computer is unable to initiate the Sick LRF reliably, due to the cold (Michigan) weather. Sick sells special outdoor models that are equipped with heating plates, but our Sick model LMS 200 does not have those. We will have to defer most outdoor experiments until the weather gets a little warmer, around March or April.

6.1.2 Range and Reflectivity data

We are interested in obtaining both range and reflectivity data continuously from the Sick LRF. Until now we worked only with range data. Since the Sick documentation is quite inadequate and the company is not very responsive to inquiries, we tried to decode the format of the data stream provided by the LRF ourselves. We were successful at doing so and are now able to command a range and reflectivity mode and then read the range and reflectivity data stream.

One important insight we gained from analyzing this data is that it may be possible to use reflectivity information to identify and reject erroneous readings due to the so-called “mixed pixels” phenomenon. The mixed pixel problem is well-known in connection with LRFs and it can be the largest source of erroneous data under certain indoor operating conditions. In order to investigate this possibility we will conduct further characterization experiments on our linear motion table testbed. Testing the incoming data stream for discontinuities in reflectivity, which is an indicator for mixed pixels, is simple and computationally less expensive than our filtering algorithms. However, the penalty is that the combined range and reflectivity data exceeds the 500 Kbaud bandwidth of the serial LRF-to-computer interface. As a result, we have to limit the horizontal scan range to less than the nominal 180°.

6.2 Position Estimation

During most of December we spent most of our efforts on the migration of our existing software from off-line, post processing code (written as Matlab interpreted scripts) to real-time, object oriented code written in C++. This migration effort is detailed below in Section 6.4.1 Software Migration and it is mostly completed.

We also worked on integrating the new C++ code with the onboard sensors and data acquisition components. We then began experimentation with the new C++ software running on the Pioneer. We still need to address some minor issues, such as defining the encoder reading interface and probably some changes in the serial communication protocol.

6.3 Novel mobility concepts

We are continuing to develop in parallel the OmniPede and the OmniTread (collectively called “Omnis”) versions of our innovative hyper-mobility platform. We recall that the newer OmniTread is covered by treads instead of legs to provide large and constantly moving surfaces all around the vehicle.

6.3.1 New OmniTread design

We completed machining the chassis of one OmniTread segment and are now beginning to manufacture drive train components and assembling the gearbox. The chassis has all the center holes drilled but the holes still need to be drilled to the appropriate size for the screws they are made for.

We obtained two double-sided belts that we will use as treads for the prototype segment. Once completed, the prototype segment will have a total of eight treads. Because the treads are somewhat costly we wanted to verify that they work well before ordering more. Having one working side is enough to demonstrate that the drive system works and adding the others later will not take long.

After careful consideration we decided to change the pulley-axles from a round cross section to a hexagonal one. The hexagonal axle locks the pulleys onto it so they cannot slip. And, as long as all the other components that have to fit the axle have a hexagonal key in the center, they will not all have to be interconnected or set-screwed into place on the axle. This greatly increases the strength and durability of the design. We have ordered the shaft and investigated ordering pulleys with the hexagonal key machined into them.

We expect to have a working prototype segment moving by mid-January, although it will not have all the parts completed. This will leave for next month the machining of the parts for the idler pulleys (the small ‘wheels’ that roll along the bottom of the tread) and the parts that hold the joint actuators. However, it may not be necessary to add the idler pulleys at all, depending on our observations once the prototype starts moving. Also, beginning in January a team of three undergraduates will help with the machining.

6.3.2 OmniPede design

We continue the design of small mechanical improvements for the OmniPede. Specifically we wanted to simplify the transmission mechanism and redesign the joint angle feedback. As a solution for the latter we found a low profile rotary position sensor that is only 2.1 mm thick and which fits perfectly on the joint. The thus freed-up space can be used to mount bellows, with which we are also experimenting at this time as a possible replacement for the heavier pneumatic cylinders that actuate the OmniPede’s joints.

6.3.3 Joint actuation strategies

One large issue we have so far not addressed is the control strategy for the joints of the two Omnis. We expect this to be a very complex matter, given that the Omnis are hyper-redundant mechanisms with two actuated joints for any two adjacent segments.

As a first step toward gaining a better understanding of the joint actuation problem we have begun the development of dynamic simulations of the OmniPede that included activation of the joints. The software package we selected for this purpose is called Working Model. This package proved to be intuitive yet powerful. The only shortcoming with Working Model is that we are limited to 2-dimensional analysis. In its current implementation our simulated OmniPede (called “SimuPede” in the remainder of this report) consists of four segments with two legs each. All legs are driven with constant velocity and with a permanent walking pattern. The simulated pneumatic actuators for the joints are currently operated in open loop, although we are planning to add closed-loop control to them soon.

The first two working simulations we produced so far show the behavior of the SimuPede when it reaches an obstacle higher than the step-height of a single leg (about 15 mm). When all joints are locked, SimuPede’s feet slip on the ground as it reaches the step. This behavior matches that of the OmniPede. However, as the second simulation shows, proper activation of the first joint allows the SimuPede to traverse the obstacle.

We will continue the development of SimuPede over the next few months. The immediate next improvement will be the provision of binary force feedback from the feet, so as to determine when a SimuPede foot touches the ground.

6.3.4 Active pneumatic suspension vehicle

We completed the assembly of our active suspension test-bed and began performing dynamic experiments with it. The control system is currently implemented on an off-board PC and input/output signals are transmitted through a multifunction A-to-D and D-to-A converter board. The test-bed itself consists of four wheels that are mounted to the distal ends of the piston rods of vertically installed pneumatic cylinders. Each cylinder is controlled by two digital valves. Feedback is provided by four potentiometers that measure the vertical position of the wheels for proprioceptive position feedback. Two accelerometers mounted in the center of the vehicle measure tilt by measuring deviations from gravitational acceleration. The goal of the pneumatic suspension system is to keep the vehicle chassis horizontal, keep all wheels on the ground at all times, and to try and keep the pneumatic actuators near their center position. The control loops are implemented as follows:

- two P-I-D controllers are preserving the horizontal orientation of the vehicle's chassis,
- two P-D controllers try to keep the four pneumatic actuators in their mid-stroke position

Both control loops work against each other but as only one has the integration (I) component the system in its entirety performs as desired. When any number of wheels are pushed upward (i.e., moving over a bump or climbing up a slope) the controllers are successful at preserving the horizontal orientation of the chassis. They do so by contracting the actuators of the lifted wheels and extending the others. When the vehicle is on a flat and horizontal surface all actuators assume their mid-stroke position.

Our experiments so far were limited to steady state conditions (i.e., slow movements of wheels). In order to perform dynamic tests, however, the current test-bed needs to be expanded. This is because the accelerometers currently used to measure tilt are affected by accelerations in longitudinal direction. We are currently investigating options for implementing a fast acting tilt measuring system.

6.4 Infrastructure

6.4.1 Software migration

After completing the common software structure for all our projects (as reported in last month's progress report), we focused on software migration from Matlab to 'C++'. This migration has become necessary because in the past, when we were focusing on the development of a basic position estimation system, we ran only about 10% of the code in real-time and on-board the robot. We did so only for the purpose of controlling the robot and for the acquisition of real-time data. The remaining 90% of the position estimation code was written in Matlab interpreted script. This large body of code was used to analyze the real-time data off line, and to test diverse algorithms for extracting useful position estimation data from the raw data. Now, however, several of our projects require completely functioning real-time position estimation modules. For example, our obstacle negotiation work with the Sick laser rangefinder requires a pose estimation system for the Sick LRF, and so does the work with the active pneumatic suspension system.

We spent most of the month planning and implementing the migration of our positioning estimation algorithms from Matlab interpreted script code to object oriented C++ code. Considering that Matlab is a vector-oriented language, converting the Matlab scripts into C++ was not a simple task since there is no one-to-one correspondence between the two languages.

One of the most critical parts of this conversion was the development of a Fuzzy Logic Inference System module, which is the heart of our FLEXnav positioning system. Considering its importance, we made special efforts to ensure that the code was fast and accurate. We tested this module extensively and compared the results with those from the Matlab toolbox.

In addition we created and incorporated a few tools that will allow us to do any future development directly in C++. Among these tools is a simulation mode, which allows offline testing of future code.

Initial tests show that the C++-based positioning system algorithm is very fast. Our full FLEXnav implementation can be completed in less than 500 µs on our onboard 166-MHz Pentium-based Libretto and in less than 50 µs on a 1-GHz Pentium III computer.

6.4.2 High-end Analog-to-digital Converter

We developed low-level routines to interface with the vendor-supplied device drivers for initializing and reading data from the National Instruments data acquisition card. The driver reads sequential samples from eight channels to get data from three KVH gyros as well as two accelerometers. This interface is working and transferring data but we have yet to complete the debugging under field conditions.

6.5 Radiation Sensing and Imaging

Form the beginning of December 2002, we have been concentrating on setting up the PSPMT/VAMCR radiation imaging system and making preliminary measurements. This includes building a detector housing, setting up and testing electronics and making measurements with a small crystal array on the PSPMT. The VA Multichannel Readout system is also being checked carefully since it represents a FOAK high density readout scheme that has not been previously characterized. We have also begun a preliminary study on using coded apertures with a pre-existing medical gamma camera.

MAJOR ACCOMPLISHMENTS:

- The University of Michigan Intellectual Properties Office filed a formal “Continuation-in-Part” application with the US Patent Office. If granted, this filing will extend the coverage of our soon-to-be-issued OmniPede patent to the OmniTread invention.
- We submitted two journal papers:
    1. Ye, C. and Borenstein, J., “A Novel Filter for Terrain Mapping.” Submitted to the *IEEE Transactions on Robotics and Automation*.
    2. Ojeda, L. and Borenstein, J., “Methods for the Reduction of Odometry Errors in Over-constrained Mobile Robots.” Submitted to *Autonomous Robots* (Journal).