In-Band Autonomous Maintenance of Mobile Free-Space-Optical Links: A Prototype

Mahmudur Khan*, Garrett Winkelmaier†, and Murat Yuksel‡
* ‡Department of Computer Science and Engineering
†Department of Electrical and Biomedical Engineering
University of Nevada, Reno, NV 89557
Email: *mahmudurk@nevada.unr.edu, †winkg14@gmail.com, ‡yuksem@cse.unr.edu

***Abstract*—Free-Space-Optical (FSO) communication has the potential to be a significant part of mobile wireless networks. It provides wireless communication at rates of tens of GHz, spatial reuse and enhanced signal security. Despite its various advantages, FSO communication requires strict maintenance of line-of-sight (LOS) and is prone to mobility. We consider two autonomous mobile nodes, each with one FSO transceiver mounted on a movable head capable of rotating 360^o. We present a prototype implementation of such mobile node with FSO transceivers. The nodes use a simple LOS alignment protocol to maintain an FSO link between themselves. The effectiveness of the alignment protocol is evaluated by analyzing the results obtained from experiments conducted using the prototype. The results show that, by using such mechanically steerable transceivers and a simple auto-alignment mechanism, it is possible to maintain optical wireless links in a mobile setting with nominal disruption.**

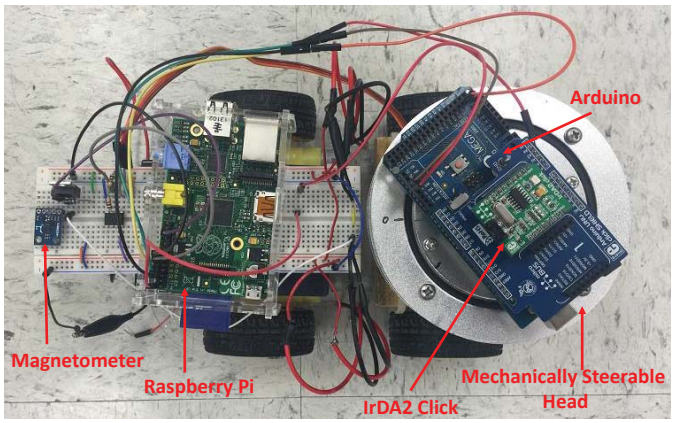


Fig. 1: Bird's eye view of the prototype

I. Introduction

Free-space-optical (FSO), a.k.a. optical wireless, communications is a promising complementary approach to the traditional radio frequency (RF) networks. FSO communication (FSOC) is attracting significant interest from telecommunication research and industry, mainly due to the increasing capacity crunch faced by the RF wireless technologies [1]. An FSO transceiver consists of an optical transmitter such as an LED or a laser diode and a receiver such as a photodetector (PD). Such FSO transceivers are low cost, small in size and lightweight. They can be densely integrated and efficient in terms of power consumption (100 μWatts for 10-100 Mbps). Moreover, they have a long lifetime (10+ years) and can be modulated at high speeds (1 GHz for LEDs/VCSELs and higher for laser diodes). FSO transceivers operate in large ribbons of unlicensed spectrum amenable to wavelength-division multiplexing (infrared/visible) [2]. The high directionality of FSO transceivers improve spatial reuse. Moreover, they can provide connectivity in conditions that are unfavorable for RF communication, e.g., presence of RF jamming or interception. A highly useful feature of FSOC is its inherent signal security due to the containment of FSO signals behind walls [3].

There are many different applications of FSOC including both civil and military scenarios. Observation behind enemy lines, sharing of high-resolution imagery and guidance data in next-generation air-traffic control, airborne Internet, monitoring of a traffic jam or disaster area are a few examples of these scenarios [4]. FSOC is useful not only for networks with stationary nodes but also with mobile nodes. Fixed FSOC techniques have been studied to remedy small vibrations [5], swaying of the buildings using mechanical auto-tracking or beam steering, and interference and noise [6]. Line-of-sight (LOS) scanning, tracking and alignment have also been studied for years in satellite FSOCs [7]. Yet, the major challenge for FSOC is still its vulnerability against mobility.

FSOC has a strict requirement of establishing clear line-of-sight (LOS) and then maintaining the LOS alignment [2]. The alignment must be maintained to compensate for any sway or movement. The alignment protocol proposed in our earlier work [8] showed that it is possible to maintain optical wireless communication in a mobile setting with minimal disruption. We considered two autonomous nodes/robots moving in random directions, each initially unaware of the location of the other. We focused on the case where the mobiles have an FSO transceiver each, mounted on a mechanically steerable head (which could as well be a simple arm) capable of rotating 360^o. Through simulations we showed that using such mechanical steering capability to control the rotation of the transceivers; the problem of LOS detection, establishment and maintenance can be effectively tackled *by support from only a compass without requiring global positioning system (GPS)*. In this paper, we present a prototype implementation of such mobile FSO nodes, shown in Figure 1. Through analysis of experimental results, we show that it is possible to maintain a

978-1-5090-0448-5/16/$31.00 ©2016 IEEE

highly directional FSO link between two autonomous mobiles with high degrees of accuracy.

The rest of the paper is organized as follows: In Section II, we review the literature for wireless link maintenance between multiple mobiles or in robot teams. Then, in Section III, we discuss our proposed method for maintaining communication link between two autonomous mobiles using FSOC and in Section IV we describe the prototype in detail. In Section V, we provide some initial experimental results. Finally, we conclude in Section VI.

II. Related Work

There are several application areas ranging from robotics to vehicular networks that require maintenance of directional wireless communication links between two or more mobiles within LOS each other. It has been an attractive problem to both research and industry. In [9], the authors illustrated an experiment which is representative of various prominent stages in a group-formation task such as formation-achievement, maintenance, and response of formation movement to the presence of obstacles among multiple robots. In terms of localizing and tracking, [10] presented an approach for deploying a team of mobiles to form a sensor network in indoor environments. In this approach the mobile sensors use different techniques such as acoustic sensing, laser scanning and a vision system like camera for localization. Another localization work was presented in [11] as a method for measuring the relative position and orientation between two mobile robots using a dual binaural ultrasonic sensor system. In [12], a laser-based pedestrian tracking system in outdoor environments is presented using multiple GPS-enabled mobile robots.

A hybrid RF-FSO system is presented in [13]. In this work, the authors developed a system consisting of an MRR (Modulating Retro-Reflector)-based FSO link with a tracking optical terminal, a conventional RF link and a deployable pod to provide a relay node bridging the FSO link to the operator and the RF link to the robot. The MRR-FSO link provides the capability to operate the robot in the presence of jamming while the RF link allows short range non-LOS operation. Methods for establishing and maintaining an FSO link among nearby balloons with the aid of GPS, RF, camera, and communication with a ground station have recently been shown in [14]. In [15], maintaining an FSO link between two UAVs hovering in a given location and orientation was considered.

These prior works assume support from a positioning system and availability of an out-of-band communication channel between the mobile nodes. Our approach [8] assumes *no availability of GPS or other positioning systems* and *uses the FSOC link itself to maintain it*, i.e., it is an in-band approach. It further assumes autonomy for the two mobiles and hence works in a completely distributed manner for calculating the mechanical steering amounts.

III. FSO Link Maintenance Protocol

We consider two autonomous mobile nodes in a GPS-free environment. The nodes are equipped with only a compass to give them the sense of direction. Each node has a mechanically steerable head that is mounted with an FSO transceiver. Our algorithm [8] has two main stages: (i) Neighbor discovery or LOS detection and (ii) FSO link maintenance.

A. Discovery

The nodes are initially unaware of each other's position and move in random directions. But they are within the communication range of each other. Neighbor discovery and, in particular, LOS discovery with its orientation over a directional link is hard to do in an entirely in-band manner. Most of the existing solutions involve omni-directional beacons so that the nodes can exchange their orientation (e.g., w.r.t. a compass) and then steer their heads towards each other. One possible in-band approach could be as follows: The nodes rotate their steerable heads continuously in clockwise but with different angular velocities. While rotating, both nodes keep sending search signals (SYN) through the FSO transceivers. Once a node receives a search signal it responds with an acknowledgement (SYN_ACK). The sender of the search signal also responds with an acknowledgment (ACK) once it gets the SYN_ACK. At this stage, both nodes stop rotating their heads and complete the FSO link establishment. Then they move to the next stage, i.e., preserving an existing FSO link between two mobiles, which is the focus of this paper.

B. Maintaining The Link

In this stage, we assume that an FSO link has already been established between the two mobiles, and the goal is to maintain this link. While establishing the link, a node conveys the information about its velocity, the direction in which it is moving and the orientation of its head to the other node. This information is used by the other node to set the angular velocity and the direction of rotation (clockwise/counterclockwise) of its head to maintain the link. The nodes exchange their velocity and signal quality values periodically over the FSO link itself. These periodic exchanges allows the nodes to recalculate how fast they should turn their heads so that the FSO link stays up. Every time the nodes exchange their information, the receiver node checks if it has deviated greater than a preset threshold. If so, the receiver node recalculates the angular speed of its head.

Let t_x be the time period of information exchanges between the two nodes. Further let α be the ratio between the angle of deviation, θ_d, of the receiver from the normal of the other node's beam and the divergence angle, θ, and α_{max} be the maximum allowed α before recalculation of rotational speed is performed. At every t_x, α is checked, and if $\alpha > \alpha_{max}$ then the angle of rotation, X, for the mobile FSO nodes are recalculated and updated as follows:

$$\angle X = \angle X \left(1 \pm \frac{\theta_d}{\theta}\right) \tag{1}$$

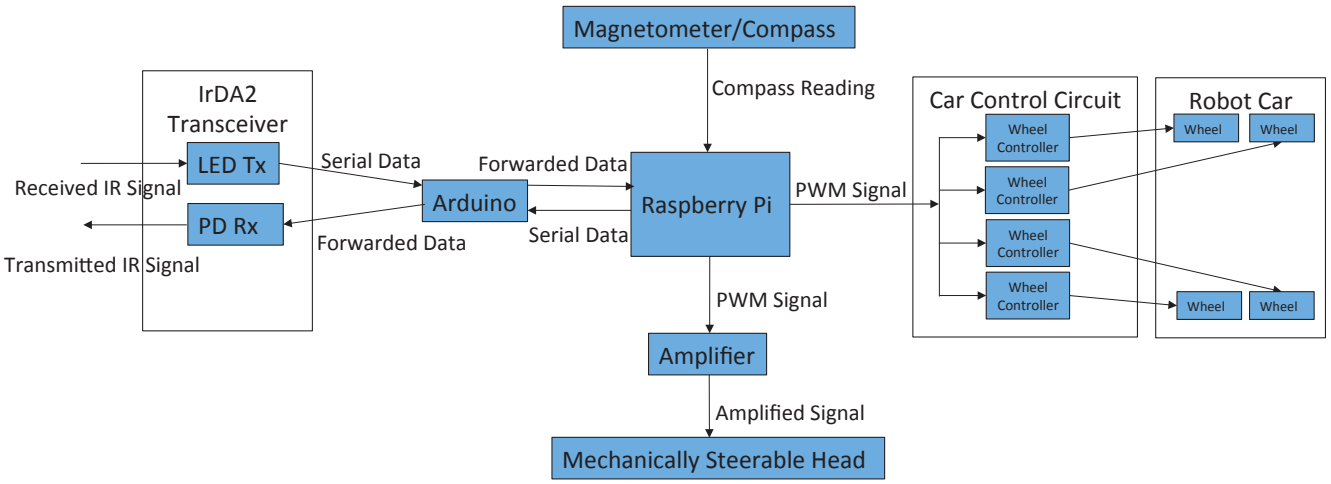
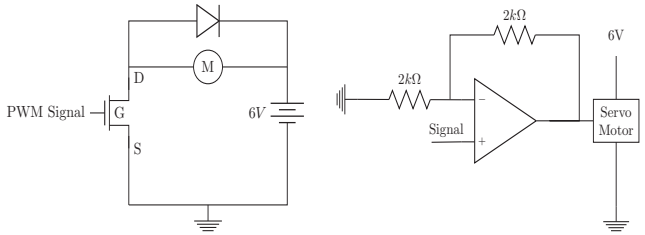


Fig. 2: Prototype block diagram

IV. Prototype

We designed and built a prototype of the mobile node with a mechanically steerable FSO transceiver by employing commercially available off-the-shelf electronic components. The prototype and a block diagram of the proposed system are shown in Figure 1 and Figure 2, respectively. The main parts of the prototype are: a robot car, a mechanically steerable head, a magnetometer, and IR transceiver. All of these parts are controlled by a Raspberry Pi [16] using separate threads: head control, car control, compass readings, and transmit or receive data. Due to UART compatibility issues of Raspberry Pi, we added an Arduino to handle the transmission and reception of the serial data (via UART) to/from the IR transceiver.



(a) Robot wheel controller (b) Amplifier circuit for the head

Fig. 3: Circuit diagrams for robot car wheel controller and mechanically steerable head

A. Robot Car

We used the Emgreat 4-wheel Robot Smart Car Chassis Kits car [17] as the mobile node. The car is four wheel drive with dimensions 25×15cm and has carrying capacity of about 1kg. Its maximum speed is 40m/min. Figure 3a shows the circuit diagram for Robot wheel controller. It utilizes a Metal Oxide Semiconductor Field Effect Transistor (MOSFET) as a switching mechanism to control the rotational speed of the DC motor. There are four motors each attached to a wheel and propel the car forward. The gate of the MOSFET is connected to a General Purpose Input-Output (GPIO) pin on the Raspberry Pi where a pulse width modulated (PWM) signal is sent. By varying the duty cycle of this signal, the DC voltage drop across the motor can be varied which results in a changing rotational velocity.

B. Mechanically Steerable Head

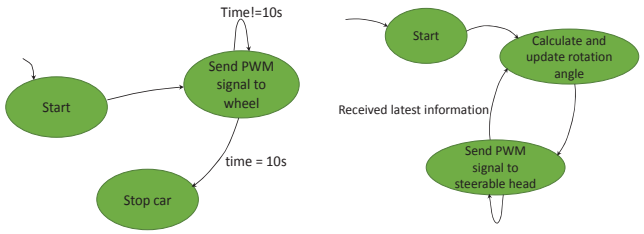
We used the Aluminum Robot Turntable Swivel Base [18] as the steerable head on which we mount the IR transceiver. It is run by a servo motor. The swivel base consists of an outer stationary ring while the inner ring is free to move. The servo is bolted to the outer ring with the gear mounted to the inner ring allowing for a controlled rotation of the inner ring. An op-amp (Figure 3b) is used to amplify the signal between the Raspberry Pi and the servo motor to control the head rotation. The servo motor works off of a PWM signal with a high voltage of 4.5-6V. Since the Raspberry Pi GPIO pins can source a maximum of 3.3V, the op-amp was added to increase the voltage of the signal.

C. Transceiver Circuit

We used IrDA2 Click [19] as the transceiver. The IrDA2 Click is an accessory board in mikroBUS form factor. The board features the MCP2120 which is a low-cost, high performance, fully-static infrared encoder/decoder. This device sits between a UART and an infrared (IR) optical transceiver. The board also features TFDU4101 diode which is an infrared transceiver module compliant with the latest IrDA physical layer standard for fast infrared data communication. It supports IrDA speeds up to 115.2Kbit/s. Integrated within the transceiver module are a photo pin diode, an infrared emitter (IRED), and a low-power control IC to provide a total front-end solution in a single package. This device covers the full IrDA range of 3m using the internal intensity control. The IRED has peak emission wavelength of 900nm and the angle of half intensity is $\pm 24^o$.

D. Magnetometer/Compass

The Raspberry Pi reads three separate magnetic readings produced by a digital compass [20]. The readings correspond to each rectangular coordinate direction: X, Y, and Z. These analog outputs are connected to the analog pins on the Raspberry Pi. An open source library function named HMC5883L [21] is used to convert the readings into a heading. For this prototype the X-axis on the breakout board is aligned with the forward direction on the car and is considered its heading. This library must be manually calibrated for different areas as the magnetic declination changes geographically. Without the calibration, a true north cannot be determined, although the heading can still be used. By taking the difference between each nodes' headings the direction of movement between the nodes can be determined and used for calculation. Fig. 1 shows the location of the digital compass with the X direction aligned with the forward direction of the node.



(a) Car control thread (b) Head control thread

(c) Transmitter thread (d) Receiver thread

Fig. 4: Different threads running in the prototype

E. Implementing The LOS Maintenance Protocol

In this paper, we kept one of the nodes stationary (Node A) and the other node (Node B) mobile. Node A can only receive data and Node B can only transmit. As stated earlier, we focus only on the maintenance phase in this paper. So, we assume the nodes know the distance between them and are initially pointing their transceivers toward each other. Upon launching the programs on both Nodes' Raspberry Pis, the GPIO pins are initialized. This is done by initializing the BCM2835 (micro-controller in Raspberry Pi) to accept GPIO manipulation and then configuring specific pins to be used as output/input for the various sensors and controller signals. Node A then enters into a waiting period where it waits for a UART signal (sent from Node B) to be available for reading. Node B, after initialization, sends the first control packet which includes all of its initial information (speed, direction) required for head rotation calculations. Both programs then enter into

Control Packet Identifier	Packet Number	Car Speed	Car Heading	Data
1 byte	2 bytes	1 byte	1 byte	3 bytes

Control Packet

Data Packet Identifier	Data
1 byte	7 bytes

Data Packet

Fig. 5: Packet structure

a delay period so that the initial starting time is the same on either node. Node A then generates two threads: one for head control (Figure 4b) and another one for data reception (Figure 4d). Node B generates three separate threads: one data transmission (Figure 4c), one for car control (Figure 4a) and one for head control. Node B's car control thread periodically sends a PWM signal to the GPIO pins that runs the wheels. The head control thread performs the head rotation. The data transmission thread continuously sends data packets to Node B with control packets mixed in to the stream. Node B's reading thread continuously reads packets being sent from Node A. At Node A, the data reception thread takes care of receiving the data sent from Node B.

The head rotation is controlled by the head control thread. The head control thread on both nodes calculate the required rotation angle using the available information about speed and direction. As imposed by the protocol, Node B sends its speed and direction information to Node A at every t_x. This information is used by both nodes to calculate their respective head rotation angles.

The data transmission and reception in the two nodes are not solely performed by the Raspberry Pis. An Arduino is introduced between the IrDA2 transceiver and the Raspberry Pi. The IrDA2 is connected to an Arduino using an Arduino shield. The Arduino along with the IrDA2 transceiver are mounted on the inner ring of the rotating head. The Arduino is introduced as a buffer between the IrDA chip and the Raspberry Pi's GPIO pins. While building the prototype, we found out that the Raspberry Pi has a hardware glitch when trying to communicate using UART. Every transmitting byte was being preceded by an unintentional high bit. When using the UART directly with the IrDA chip, this unintentional bit was being interpreted as a start signal. The byte being read by the IrDA from the Raspberry Pi was then incorrect. Due to the short duration of the bit, the Arduino does not read the bit as a start bit and ignores the unintentional start bit. By using the Arduino as a buffer between IRDA and the Raspberry Pi, the unintentional start bit is filtered out and the correct byte is transmitted and received through the IR transceivers.

V. Experimental Results

We performed some initial experiments using the prototype to gain insight about the effectiveness of our proposed FSO link maintenance protocol in [8]. We considered one node stationary and the other mobile. Also, we considered half-

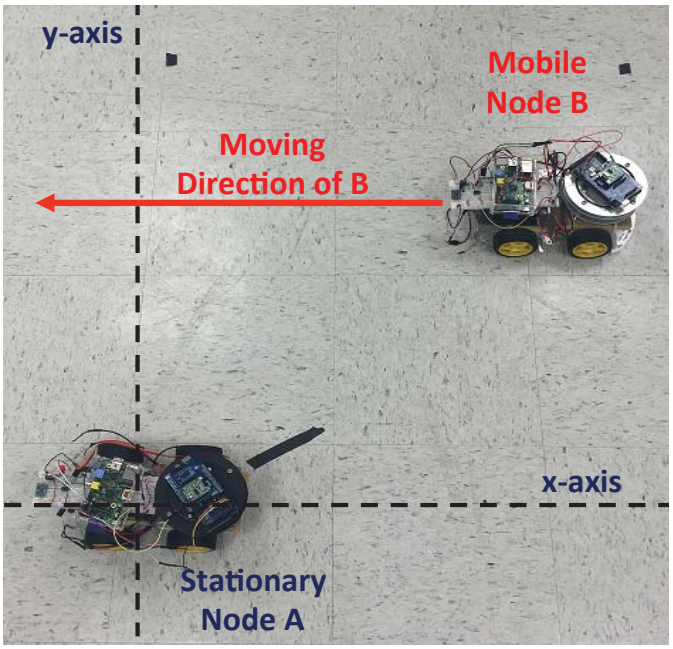


Fig. 6: Experiment scenario I

duplex communication mode since the available IrDA2 click transceivers do not work in full-duplex mode. The mobile node worked as a transmitter and the stationary node was the receiver. For data transmission, we used the maximum baud rate of 115.2 Kbps of the IrDA2 click. We considered two types of packets: control packets and data packets. The control packet contained the speed and direction information of the mobile node. The data packet was randomly generated. The packet structures are shown in Figure 5.

We performed two experiments on the setup shown in Figure 6. For both experiments, we kept Node A stationary and Node B moving. We varied the message exchange interval t_x (i.e., the time interval between sending the control packets) and Node B's speed v_x. We measured the throughput of the FSO link as the performance metric.

A. Experiment I: Angular Change

The goal of this experiment is to show that the protocol can effectively maintain the FSO link while there is an angular change in the relative position of the two nodes. In this experiment, Node B moved on a straight line parallel to the x-axis as shown in Figure 6. As Node B moves, both nodes have to tune their heads' angular speeds in order to maintain the link. Node B was stopped after it traveled 8 feet (or 2.44m), half of it towards the y-axis and the other half away from it. This parallel movement requires changing the heads' angular speeds continuously.

We performed the experiment for $t_x = [1, 2750]$ms and $v_x = [0.375, 0.75]$m/s. Figure 7 shows how the throughput (in $Kbps$) behaves as t_x varies. We can observe that the throughput was ≈68Kbps for tx of up to 2,500ms at speeds 0.375–0.75m/s. The throughput drops significantly for larger t_x. We can also observe that the maximum value of t_x to

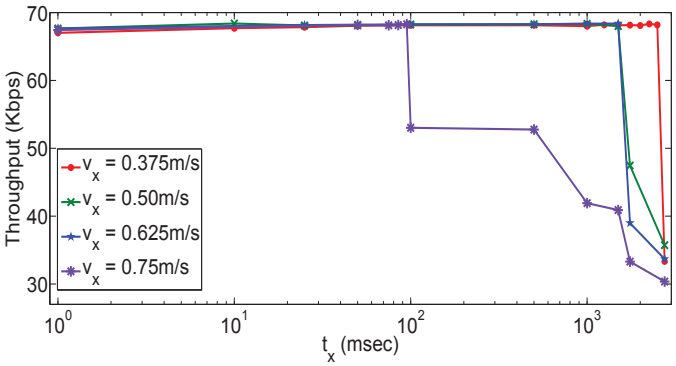


Fig. 7: Effect of t_x on throughput for different car speed (x-axis in log scale)

achieve a throughput of ≈68Kbps reduces with increase in speed. As expected, this means that higher speeds require more frequent (smaller t_x) information exchange between the nodes to maintain the same level of throughput. For example, for v_x=0.375m/s, the throughput was ≈68Kbps for 1ms≤ $t_x \leq$ 2,500ms. Likewise, for v_x=0.50m/s, the throughput was ≈68Kbps for 1ms≤ $t_x \leq$1600ms.

Figure 8 displays the effect of α_{max} (described in III-B) on the maximum value of t_x to maintain a throughput of ≈68Kbps. For our experiments, we used α_{max} of 0.125 and 0.25. We can observe that, for both cases, maximum t_x reduces with increase in velocity. We can also see that the maximum t_x is larger for $\alpha_{max} = 0.25$ than that for α_{max}=0.125. We also performed experiments with $\alpha_{max} = 0.375$, but for this case even with t_x=1ms, the throughput was ≈50Kbps.

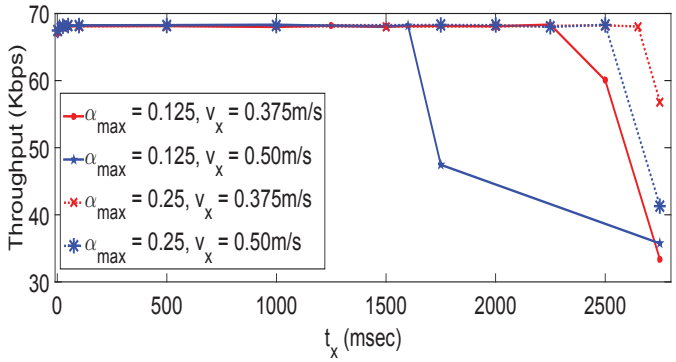
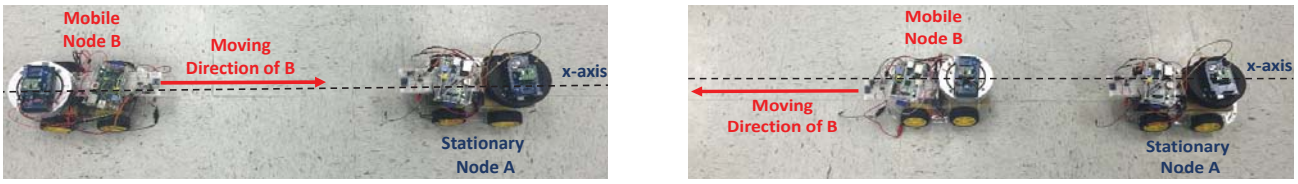


Fig. 8: Effect of α_{max} on t_x and throughput

B. Experiment II: No Angular Change

The goal of this experiment is to show the sanity of the proposed alignment protocol. In this scenario also, we kept Node A stationary on the x-axis and Node B mobile but there was no angular change in the relative positions of the nodes. First, we performed experiments with Node B moving along the x-axis towards Node A (Figure 9a). For all values of t_x and v_x, both nodes correctly calculated the head rotation angles to be 0^o and successfully maintained the FSO link. The throughput remained ≈ $68Kbps$ for all the trials.



(a) B moving towards A

(b) B moving away from A

Fig. 9: Experiment scenario II

Then, we performed experiments with Node B moving on the x-axis away from Node A as shown in Figure 9b. In this case also, all the packets sent by Node B was received by Node A correctly.

VI. Summary and Future Work

We presented a prototype implementation that deals with the problem of LOS detection and maintenance for maintaining an FSO link between two autonomous mobile nodes. Each of the nodes is equipped with a mechanically steerable head (or arm) on which an FSO transceiver is mounted. We showed through experimental results that using a simple protocol to control the mechanically steerable head, the FSO transceivers on the nodes can maintain a communication link successfully. The current communication speed is limited by the IrDA2 Click (maximum 115.2 Kbps). This limits the speed of the data transfer between the nodes regardless of the IR transceivers capabilities. Also, the maximum range of the IR transceiver is limited to 3 meters. We considered one node stationary and one node mobile in this work. For implementing the proposed FSO link maintenance protocol between two mobiles, transceivers capable of full duplex communication are required. As future work, we plan to develop an algorithm for the discovery phase and test it using the developed prototype. We also plan to develop full duplex optical transceivers for improving the prototype.

Acknowledgement

This work is supported in part by NSF CNS award 1422354, ARO DURIP award W911NF-14-1-0531, and NASA NV Space Consortium.

References

[1] A. Sevincer, A. Bhattarai, M. Bilgi, M. Yuksel, and N. Pala, "Lightnets: Smart lighting and mobile optical wireless networks – a survey," Communications Surveys & Tutorials, IEEE, pp. 1–22, 2013.

[2] A. Sevincer, M. Bilgi, M. Yuksel, and N. Pala, "Prototyping multi-transceiver free-space-optical communication structures," in *Proceedings of IEEE International Conference on Communications (ICC)*, Cape Town, South Africa, May 2010.

[3] D. Zhou, P. G. LoPresti, and H. H. Refai, "Enlargement of beam coverage in fso mobile network," *Journal of Lightwave Technology*, vol. 29, pp. 1583–1589, 2011.

[4] ——, "Evaluation of fiber-bundle based transmitter configurations with alignment control algorithm for mobile fso nodes," *J. of Lightwave Tech.*, vol. 31, pp. 249–356, 2013.

[5] S. Arnon and N. S. Kopeika, "Performance limitations of free-space optical communication satellite networks due to vibrations-analog case," *SPIE Optical Engineering*, vol. 36, no. 1, pp. 175–182, January 1997.

[6] E. Bisaillon, D. F. Brosseau, T. Yamamoto, M. Mony, E. Bernier, D. Goodwill, D. V. Plant, and A. G. Kirk, "Free-space optical link with spatial redundancy for misalignment tolerance," *IEEE Photonics Technology Letters*, vol. 14, pp. 242–244, February 2002.

[7] S. G. Lambert and W. L. Casey, *Laser Communications in Space*. Artech House, 1995.

[8] M. Khan and M. Yuksel, "Maintaining a free-space-optical communication link between two autonomous mobiles," in *Proc. of IEEE WCNC*, 2014.

[9] K. N. Krishnanand and D. Ghose, "Formations of minimalist mobile robots using local-templates and spatially distributed interactions," *Robotics and Autonomous Systems*, vol. 53, pp. 194–213, Dec 2005.

[10] L. E. Parker, B. Kannan, X. Fu, and Y. Tang, "Heterogeneous mobile sensor net deployment using robot herding and line-of-sight formations," in *Proceedings of the IEEE IROS*, 2003, pp. 2488–2493.

[11] S. Shoval and J. Borenstein, "Measuring the relative position and orientation between two mobile robots with binaural sonar," in *Proceedings of the ANS 9th International Topical Meeting on Robotics and Remote Systems, Seattle, Washington*, 2003.

[12] J. Jiang and L. Tuan-jie, "A method for distributed multi-robot simultaneous localization and mapping," in *P. of IEEE ICICEE*, 2012, pp. 358–362.

[13] J. L. Murphy, M. S. Ferraro, W. S. Rabinovich, P. G. Goetz, M. R. Suite, and S. H. Uecke, "Control of a small robot using a hybrid optical modulating retro-reflector/rf link," vol. 9080, 2014, pp. 90 801F–90 801F–12. [Online]. Available: http://dx.doi.org/10.1117/12.2052942

[14] R. DeVaul, E. Teller, C. Biffle, and J. Weaver, "Using predicted movement to maintain optical-communication lock with nearby balloon," 2013, uS Patent App. 14/108,542.

[15] A. Kaadan, D. Zhou, H. H. Refai, and P. G. LoPresti, "Modeling of aerial-to-aerial short-distance free space optical links," in *Proc. of IEEE ICNS*, 2013, pp. 1–12.

[16] "Raspberry pi," https://www.arduino.cc/en/reference/servo.

[17] "Emgreat 4-wheel robot smart car," http://www.amazon.com/Emgreat-4-wheel-Chassis-Encoder-Arduino/.

[18] "Aluminium robot turntable swivel base," http://www.ebay.com/itm/Aluminium-Robot-Turntable-Swivel-Base-2-DOF-PTZ-2pcs-Servo-Set-for-Arduino-/271254609757.

[19] "Irda2 click," http://www.mikroe.com/click/irda2/.

[20] "3-axis digital compass ic hmc5883l," https://www.parallax.com/downloads/hmc5883l-3-axis-digital-compass-ic-datasheet.

[21] "Hmc5883l," https://github.com/sleemanj.