Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2018) 26: 1651 – 1661

doi:10.3906/elk-1701-193

Özyeğin Biopsy Robot: System integration architecture and motion compensation of a moving target

Awais AHMAD[1], **Özkan BEBEK**[2,*]
[1]Department of Electrical and Electronics Engineering, Özyeğin University, İstanbul, Turkey
[2]Department of Mechanical Engineering, Özyeğin University, İstanbul, Turkey

Received: 18.01.2017 • **Accepted/Published Online:** 03.04.2018 • **Final Version:** 30.05.2018

Abstract: This paper presents the complete system architecture of a robotic biopsy system for real-time operations. The system has individual functional blocks working simultaneously including a 5 DoF parallel robot, ultrasound (US) imaging machine, two robotic manipulators, and a motion capturing system. Details of the real-time functionality of the robotic system components working with spatial and computational synchronization are presented. This paper also deals with a scenario in which the target tissue is moving due to the breathing of the patient. The motion of the needle tip and the target is tracked using US images as feedback. Two types of control laws for target tracking are discussed: traditional feedback control and an optimal control method. Motion compensation is demonstrated by tracking a moving target with the needle tip motion with an RMS error of 0.25 mm.

Key words: Robotic biopsy, robotic system architecture, predictive control

1. Introduction

Image-guided core needle biopsies are performed to collect tissue samples from the patient's body for analysis and diagnosis. Core needle biopsies can be used to take samples from a patient's muscles, bones, and organs like the liver and prostate. Imaging technologies do not always provide sufficient information about tumors, so in some cases needle biopsies are suggested. Physicians and surgeons have to be careful while taking samples, as wrong samples can lead to a wrong diagnosis. In order to be sure, physicians normally take multiple samples, which are painful for the patients and can create postprocedure complications by damaging healthy tissue. Recent developments in robotic technologies have given the courage to introduce robotic systems in medicine. In percutaneous operations, robotic systems have provided valuable improvements in outcomes. They provide high precision and consistency in these complex operations.

Needle biopsy is a type of minimally invasive procedure. Minimally invasive surgery has also found popularity among surgical techniques due to the reduced postsurgical recovery time. Custom robotic systems have been developed for specific tissues like brain [1], breast [2], kidney [3], lung [4], and prostate [5]. In addition to these, there are systems developed for small animals [6–9] for medical studies. The common design purpose of these systems is to reduce physician error and fatigue and to get the required accuracy and reliability in the process. In a clinical setup for a biopsy, the patient's breathing causes the target tissue to move from its position inside the body. A motion compensation scheme is also needed in order to reduce errors due to the respiratory motion, which is lacking in the systems discussed above.

*Correspondence: ozkan.bebek@ozyegin.edu.tr

An imaging system is used for visual feedback during these procedures. Some commonly used imaging systems are computed tomography (CT), magnetic resonance (MR) [10], and ultrasound (US) technologies. In our study, 2D US imaging is used. A robotic system with intraoperative image guidance can be used to help physicians maintain motion stability in long and tiring operations and also make the procedure less painful for the patient.

The robots discussed in the literature are designed to insert biopsy needles with accurate orientation and position. They follow manual or predefined instructions and lack the ability to compensate the target's motion. This paper presents the Özyeğin Biopsy Robot (OBR), which is designed to perform in vivo needle biopsies, mostly focusing on breast, liver, and kidney biopsy. The OBR is a 5 DOF robotic system and uses US imaging to detect the motion of the needle tip as well as the moving target tissue in real time. The main contribution of the paper is demonstrating that the designed robotic system is able to cancel the relative motion between the target and the needle while performing the needle insertion. A target position is selected by a physician, such as an anomaly, using the US device. The robot can position itself with the desired pose and the needle insertion is performed by the robot autonomously. A complex task like this would require many components to function together. A complete system architecture that would combine seamless operation of the system components, and the design and implementation of feedback and feedforward controllers to compensate the motion of a moving target, are also presented.

2. System architecture of the OBR

Architecture is the backbone of any complex system. It is the base that defines how a system is divided into subsystems and how these subsystems communicate with each other to execute tasks. Just like any complex system, architecture is also important in the design of any robotic system, such as the robotic needle placement system presented in [11]. The right choice of architecture can lead to meeting system requirements and smooth implementation while a bad choice can cause frustrating problems in the development and can also cause restrictions [12].

Robotic systems, especially real-time systems, interact with a dynamic environment and have physical components. That is why they require more care when being designed as compared to a software system. Most systems are designed as a hierarchical system [13], a behavioral system [14], or a hybrid of both systems [15, 16]. The OBR's system components have modules just like a behavioral system and work as subsystems. As opposed to hierarchical systems, these subsystems have the ability to perform high-level tasks independently. The system architecture adopted for the OBR is the hybrid type because it has many subsystems working as independent modules, but controlled by a master module. In this approach, a modular behavioral system and also a hierarchical control with a direct link between the master module and the slave modules was achieved. All the subsystems or system components are connected as a client-server dependency via UDP networking. The modularity of the system gives it flexibility and more security as each module can be programmed to handle exceptions independently.

2.1. Experimental setup and system components of the OBR

In a clinical situation, the patient is lying down on an operating table. The physician then marks the incision points on the skin according to prior knowledge of the location of the target tissue. If prior reports are not available, the physician locates the anomaly manually using US imaging. The physician then makes a small incision on the skin and prepares to insert the needle. The needle is inserted when the patient exhales and the physician tries to keep the needle in an optimal orientation to reach the target tissue.

The experimental setup to implement and evaluate the proposed system is shown in Figure 1. The figure shows all the system components in a functional arrangement. In order to make it possible for the OBR to perform autonomous biopsies, these components must work together with the OBR to make a whole system. Each component has a function of either an actuator or a sensor in the system. In this section, each system component is discussed briefly as a module and their mutual functions are also discussed [17]. The biopsy procedures using the OBR are performed on gel or water phantoms. A six DOF industrial robotic arm, Robot I (UR3 Universal Robots A/S, Odense, Denmark), is used to hold the US probe in place and align it with the target and the needle when it is inserted. Another six DOF industrial robot, Robot II (Kuka KR6 R900, KUKA Roboter GmbH, Augsburg, Germany), is used to mimic the respiratory motion of the target tissue by moving the target point in the phantom. At the start of the procedure, the US probe is aligned with the target. Then the OBR is initialized and then the needle goes to an insertion pose. Once the needle has started to move towards the target, Robot I aligns the probe with the needle and the OBR starts to track the motion of the target tissue.

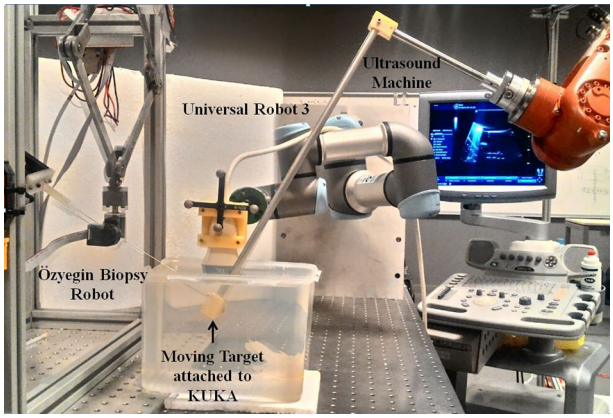

Figure 1. Özyeğin Biopsy Robot (OBR) system components.

A motion capturing system (OptiTrack, Naturalpoint, Inc., Corvallis, OR, USA) is used to track the motion of all the moving parts of the project in 3D space, such as the OBR, US the probe, and the needle mechanism. The motion capturing system is also used to calibrate the 2D US images in 3D space to align the needle axis with the US imaging plane as discussed in [17]. In the experiments, a GE LOGIQ P5 2D US machine and a GE 11L linear 2D US probe were used to acquire US images. The EURESYS PICOLO HD 3G frame grabber is used for data acquisition purposes. The US machine is used to get the visual feedback for the needle interventions.

Figure 2 shows the data flow diagram of the system components. It describes how each component is connected to each other within the system. As seen in the figure, the workstation has two Ethernet ports, one of which connects to the motion capturing system (MoCap) and the other to Robot I or Robot II for TCP/IP communications. The US imaging machine is connected to the workstation through the frame grabber, which streams video at the rate of 15 FPS.

Robot I and II communicate with the workstation using the TCP/IP protocol. A communicator can be programmed in any language that supports UDP communication. Python was used to implement UDP communicator. OBR has two-way communications with the CPU through the Quanser Q8 DAQ Board at 500 Hz. The controller for the OBR is developed using Simulink according to the control law discussed in [18].

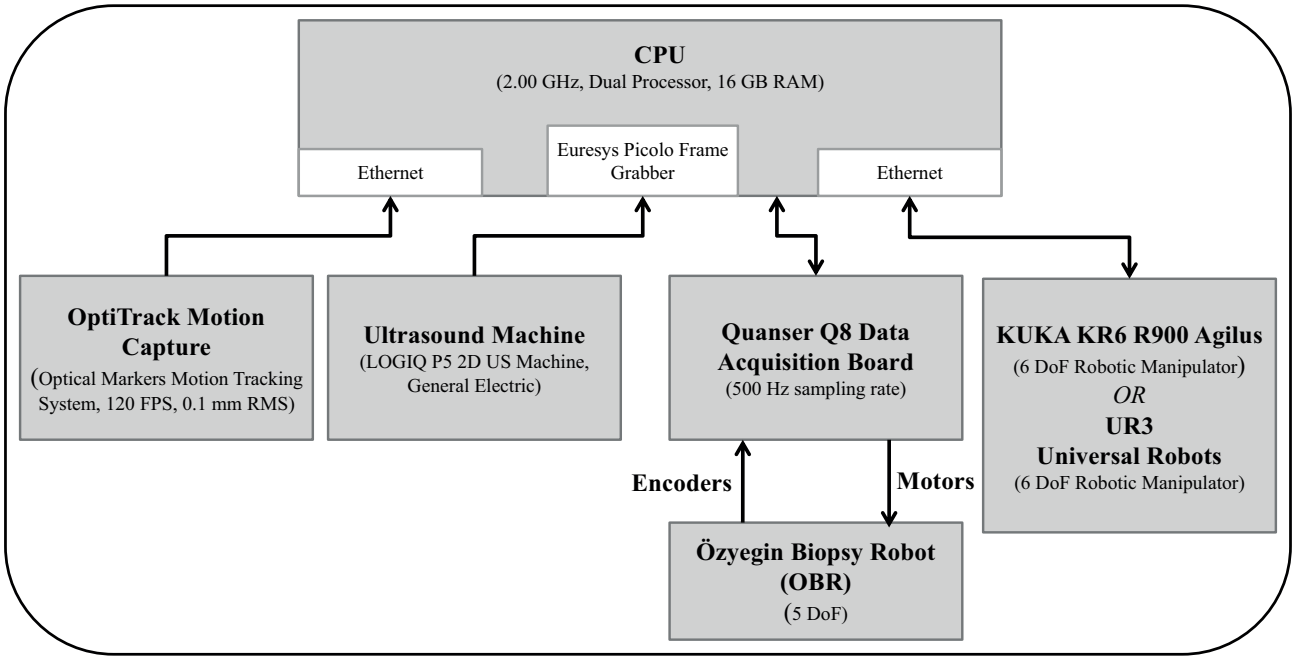

Figure 2. Data flow diagram. System components and their connection topology.

3. System integration

In this section, we will discuss how all the components are integrated together in the system and are running in parallel. The MoCap system runs on proprietary software called Motive (OptiTrack, Naturalpoint, Inc.). The cameras are calibrated in the software using a calibration wand, and then the rigid bodies are created from the optical markers. Once the software starts tracking the rigid bodies, data streaming is enabled to stream the tracking data in real time to the local or external host. The data can be listened to through opening a UDP socket in MATLAB.

A Python application, henceforth referred to as *server.py*, is programmed. This piece of code opens two UDP sockets, one to communicate with Robot I or Robot II through the Ethernet port and the other one to receive correction values from another program, which will behave as the master program, henceforth named as *Main_Module*. These applications are programmed to transmit zeroes as default correction values whenever the external link is not available. It allows them to run in parallel to a program at a different frequency of data rates and stay idle while waiting for it to initialize or send a packet of new information.

When Robot I/Robot II and the motion tracking are ready, the biopsy robot's controller is started in Simulink. It has the same mechanism of listening to a UDP socket for correction values from an external program. The controller first finds the joint angles of all the links from the inverse kinematics and then the calculated torque controller combined with the PD controller computes the motor currents.

Once all the subsystems are open and running, *Main_Module* manages the operation of these components and runs the control algorithms for the autonomous needle biopsies. The control flow diagram of this *Main_Module* is shown in Figure 3.

After initializing, *Main_Module* connects to the UDP sockets that are opened for communicating with Motive, OBR, and *server.py*. In order to run the main at a constant rate, a timer interrupt is initialized, which will trigger the main loop at constant intervals of time. The frequency of this timer is set manually and depends

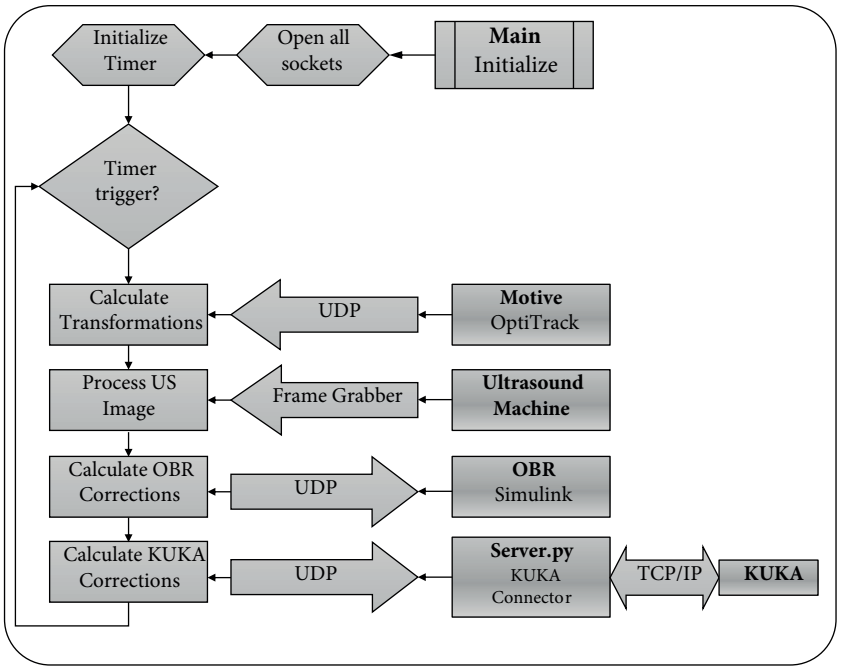

Figure 3. Control flow diagram of Main_Module.

on the processing time taken to complete one cycle. Once the timer is triggered, first of all, the MoCap data are received and transformations are calculated. After that, the US video object is triggered to get the latest frame and this frame is processed to find the location of the target point using the calibration factors found in [17]. At this point, the information from both these sensory inputs is used to generate the corresponding correction values for both the OBR and Robot I/Robot II and transmitted through the respective sockets, and then the program waits here for the next trigger of the timer.

4. Motion compensation

The OBR is designed to perform biopsies in the abdominal region. The organs of the patient's body move due to respiratory or voluntarily movements of the patient. Tumorous tissue can also move from its place due to US probe pressure on the skin and needle insertion. In order to be autonomous, the system must be able to interact with its environment and behave accordingly. In our case, the needle must reach a target tissue that is a dynamic environment due to constantly changing position [19].

This study suggests a method in which the robot moves the needle with the moving target in order to cancel the relative motion, hence reducing the error and increasing the accuracy of correct tissue extraction. Sharma et al. [20] and Schwiekard et al. [21] concluded in their studies that motion compensation of a respiratory motion is achievable. Riviere et al. [22] presented an adaptive controller that was able to model and predict the breathing motion of a target during needle interventions. Trejos et al. [23] discussed the ability of a platform that allowed a surgeon to perform tasks on a motion-cancelled target.

The motion compensation problem has many challenges. It is hard to track the moving target and the exact needle tip position in a US image. Reaching a target tissue while avoiding any obstruction requires a significant number of degrees of freedom. The OBR has 5 degrees of freedom, which makes it effective in reaching a target from any angle. A visual tracking method is used in this study to simultaneously track the

needle tip and the target. The visual tracking method is an image-based tracking method and it uses motion information in consecutive frames.

In this section, two different control techniques to track a moving target are presented. One is a traditional PD controller and the other is receding horizon model predictive control [24], or simply called model predictive control (MPC).

A 20 × 20 mm piece of rubber was used as the moving target for these experiments because its appearance in the US image was good for the segmentation algorithm. This rubber target is attached to the Robot II end effector with an extension rod, which can move it according to a breathing reference signal. The US probe is manipulated by Robot I and the probe is attached to the robot through a 3D printed probe holder. The OBR is used to manipulate the biopsy needle. Figure 4 shows the real target and the US image of the target and the needle being tracked by the image processing software.

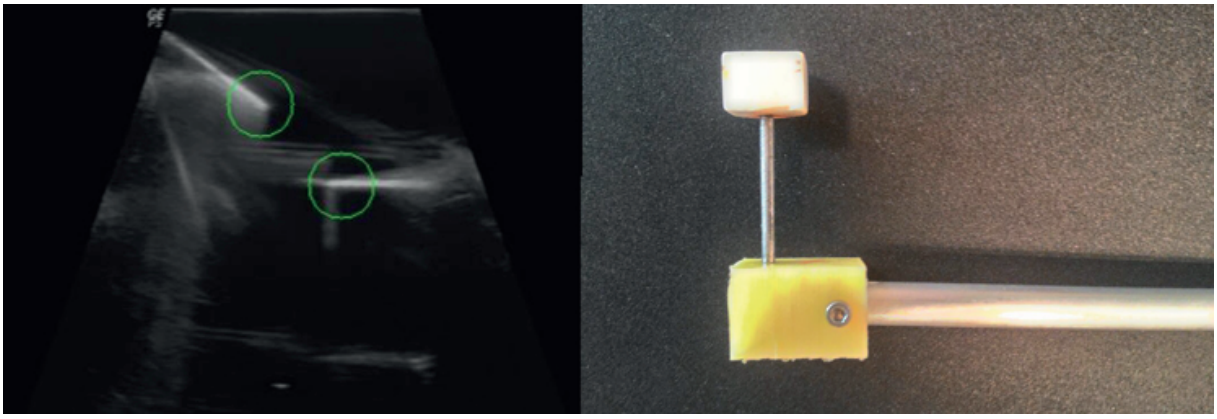

Figure 4. Left: Rubber target and needle in US image. Right: Photo of the rubber target attached to the extension rod at the end effector of the KUKA robotic arm.

4.1. PD controller

One of the simplest controllers that can be used in this situation is a PD controller. The feedback of both the needle tip and the target from the US imaging is available, also shown in Figure 4. The image processing algorithm developed in [19, 25] is used to get the position of both the needle tip and the moving target in real time. The error between the position of the target and the needle tip is sent to the OBR controller via the UDP connection as discussed in Section 3.

The algorithm is tested at different frequencies of 0.2 Hz to 1 Hz and peak to peak amplitudes of 1 cm to 3 cm. Figure 5 shows the tracking of a target motion with 0.2 Hz and 1 cm amplitude. It shows plots for both the x- and y-direction.

The RMS errors are 0.42 and 0.81 mm for the x- and y-direction, respectively. The reason for this high RMS error is the processing time of images between the updates to the controller effort. This is also one of the reasons why this controller is not suitable for higher frequencies. Sensor delay, which is discussed in the following section, is also contributing to introduce lag in tracking.

4.2. Sensor time delay – US imaging latency

All the sensors have some intrinsic latency in them, which is the time delay between the actual event and the time when the sensor reports the event. In the motion compensation application of the OBR, US imaging is our

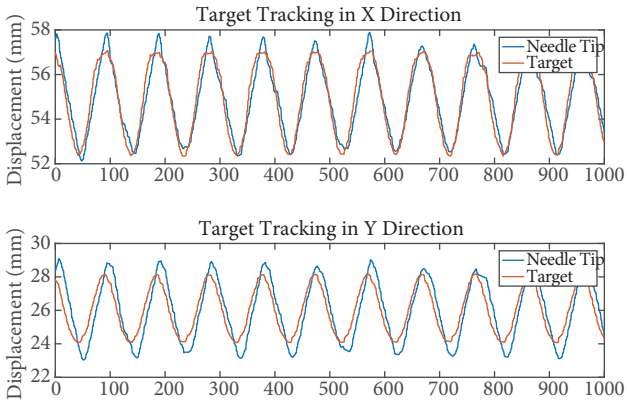

Figure 5. Motion tracking with PD control.

feedback system. The US machine takes some time to process the information received from the piezoelectric sensor array and send it to the external system. Ignoring this time delay can cause large tracking errors. Bowthorpe et al. in [26] presented a similar problem. They found the sensor time delay and implemented a Smith predictor to compensate sensor delay. An experiment is designed to measure the sensor delay and incorporate it in our control scheme.

The OBR is controlled with a joystick and moved randomly. US imaging is used to record the motion of the needle tip. Both of these events are time-stamped using the same universal clock. The commands to the OBR are stored with a time stamp, after the processing, right before they are sent to the robot, and the time for the US image is stamped right when a new frame is triggered, before any processing is performed. In this way, the computational delays of MATLAB are eliminated and the offset between two sets of data should purely be caused by the US machine's processing time. The delay between the two signals is calculated as the difference of the discrete timestamps and is 150 ms on average.

4.3. Receding horizon model predictive control

The results of tracking from PD control were above the acceptable error level of 1 mm RMS for high frequencies. The OBR is designed to have the capability of tracking a target moving with 2 Hz frequency and 10 mm amplitude. The OBR should meet the benchmark of less than 1 mm RMS tracking error under these conditions.

The sensor latency found in the previous section introduces a phase lag in the tracking. Design of a new controller is needed to handle these problems. This section presents the design of MPC [24, 27], which utilizes the information of the breathing model and sensor delay optimizing tracking problem.

MPC optimizes the current state of the system taking the future states in account. A number of samples, for instance N, are selected as a design parameter, also called the horizon, and the current state of the system is optimized for this time-horizon. The current state is optimized for $[kk + N]$ samples, k being the current sample, but only the current position is implemented. For the next position in time, the horizon is moved forward and the state is optimized over $[k + 1, k + N + 1]$, which gives this technique the name of receding horizon model predictive control (RHMPC) [28]. The length of the horizon is finite for periodic systems. The breathing model is a periodic signal and hence one cycle can be used as a feedforward signal for the next cycle.

RHMPC is an iterative and multivariable process that uses the dynamic model of the system, the history of previous control efforts, and an optimization cost function over the length of the horizon. The dynamic model

of each joint of the OBR is available as calculated during the system identification presented by Orhan et al. in [18]. Using the breathing model of the previous cycle for prediction, a cost function can be defined and optimal control can be designed to minimize the cost function.

$$J\left[k\right] = \sum_{k=k_0}^{k_0+T} \left(\left(x\left[k\right] - x_{est}[k]\right)^T Q\left(x\left[k\right] - x_{est}\left[k\right]\right) + u^T\left[k\right] R\, u[k] \right) \tag{1}$$

The optimal tracking problem goal is to find optimal control u for the system that will minimize the given cost function in Eq. (1). This will result in y tracking the signal y_est. When solved as optimal feedback regulator [27, 29], the solution to this control problem is derived from [30] and becomes as follows:

$$u\left[k\right] = u_{fk}\left[k\right] + u_{ff}\left[k\right] \tag{2}$$

The resulting control algorithm is composed of feedback and feedforward parts.

4.4. Implementation

A breathing motion model is used as the reference signal and is fed to both the target and the RHMPC. The breathing motion model was obtained from the abdomen of a breathing human using a motion capture system. Then the motion is modeled with two sinusoids and then repeated tests are conducted using the same model. The same target rubber square is used in these experiments as described in Section 4 and KUKA moves the target with the reference signal. The same reference signal is also fed to the RHMPC as a feedforward signal. As moving target is isolated from the biopsy system, both signals must be synchronized by phase. Bebek [27] presented a technique where the phase difference is estimated on the basis of biological signals of the heart. In our study, the phase is estimated using visual feeds from US imaging.

For the first 14 s when started, the encoders of the OBR are being calibrated and no control algorithm is enabled. This time is utilized to find the phase offset of the target motion and added to the input signal of RHMPC. A parallel processing thread, henceforth named *Phase_Fit*, waits for an enabling signal from the OBR controller in order to synchronize the clock. Figure 6 shows how the phase is calculated and added to the input signal.

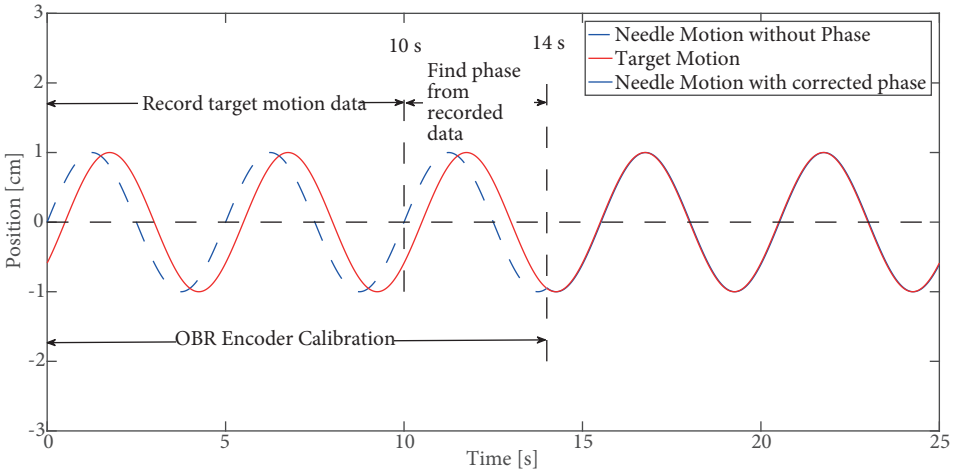

Figure 6. Timeline of phase offset detection.

The *Phase_Fit* thread stores the motion of the target for 10 s. A sinusoidal function can be fitted to the recorded data using least square fit. From this fit, the phase of the target motion can be found. The mean position of target motion to align the needle tip with it can also be found. This information is then shared with the OBR controller using a UDP connection before the 14-s mark hits. Hence, at the 14th second, the input signal phase to the RHMPC is corrected and it starts to track the target perfectly.

The KUKA robot is programmed to move the target in a sinusoidal pattern. The algorithm is tested at frequencies of 0.2 Hz to 1 Hz and peak to peak amplitudes of 1 cm to 2 cm. Figure 7 shows the tracking of a target motion with 1 Hz and 1 cm amplitude with an RMS error of 0.267 mm.

5. Comparison of RHMPC with PD

The RMS error of the PD controller in motion compensation is above our acceptable threshold value of 1 mm for high frequencies, as shown in the Table. On the other hand, RHMPC performed better in motion compensation and the RMS error was well below the threshold value of 1 mm. Figure 8 shows the comparison of the errors of these tracking control solutions. It can clearly be seen that the errors in the case of RHMPC are contained well under the safe value of 1 mm with mean very close to zero, which indicates perfect phase synchronization. On the other hand, the PD controller has a constant lag while tracking, which causes the mean of error to move away from the zero value.

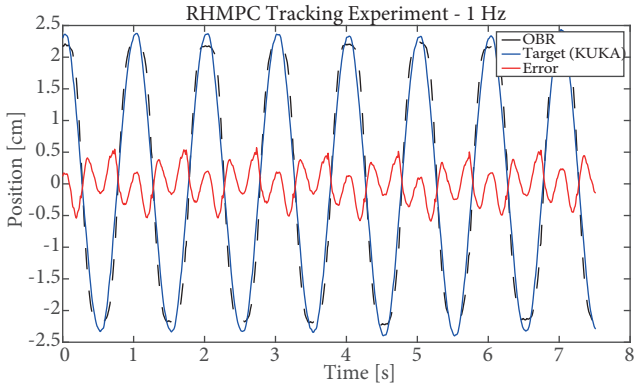

Figure 7. RHMPC target tracking at 1 Hz and 10 mm amplitude.

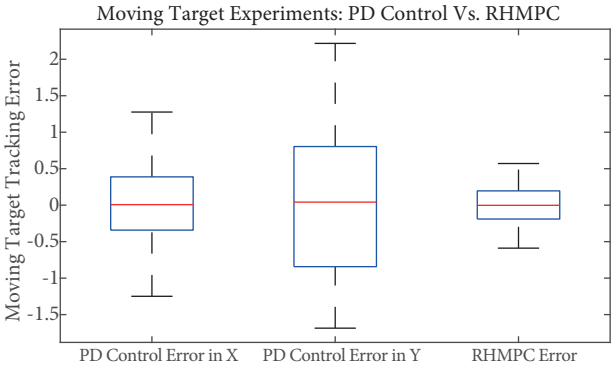

Figure 8. Tracking results of RHMPC and PD controller.

Table. RMS errors of PD and RHMPC controllers with different reference signals.

Controller type	Frequency (Hz)	Amplitude (mm)	RMS error
PD	(0.2, 0.05)	(8, 1.5)	0.95
	0.5	20	1.27
	1	10	1.82
RHMPC	(0.2, 0.05)	(8, 1.5)	0.25
	0.5	20	0.26
	1	10	0.28

The Table represents the results of both of the controllers for different types of reference signals. It can be seen from the table that the PD controller is not robust at higher frequencies but responds reasonably to signals based on breathing motion data. Meanwhile, RHMPC performs well within the safe boundary at all frequencies.

6. Conclusion

This paper presented the design and implementation of the architecture of a robotic system for autonomous biopsy on moving targets. The OBR system architecture is designed in such a way that all the system components exist as modules and one master program controls the execution of tasks. The master program is connected to the subsystem modules over a client-server dependency via UDP networking. The subsystems have the ability to run independently in the case of master failure, which makes the system secure and flexible. The OBR is light-weight and compact, which makes it easier to be used in an operating room. The OBR has enough motion bandwidth to perform fast autonomous tasks [18].

The system was tested to perform a biopsy on a target moving with different frequencies and amplitudes. In order to track a moving target, both feedback and feedforward controllers were designed. Feedback control with PD gains performed just around the minimum design requirement with RMS error of 0.95 mm considering the frequencies of the breathing motion model. To handle the higher frequencies and dynamic behavior of the operation environment, MPC is implemented. Sensor time latency was found through experiments and breathing motion was modeled to design the RHMPC. The performance of the RHMPC was found to be very promising with a tracking RMS error of 0.25 mm at 1 Hz.

Acknowledgment

This research was supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under Grant No. 112E312.

References

[1] Zaidi H, Chao Y, Lei Z,Wang Y. Design and optimization analysis of open-MRI compatible robot for neurosurgery. In: International Conference on Bioinformatics and Biomedical Engineering; 2008; Athens, Greece. New York, NY, USA: IEEE. pp. 1777-1776.

[2] Yang B, Tan U, McMillan A, Gullapalli R, Desai JP. Design and implementation of a pneumatically-actuated robot for breast biopsy under continuous MRI. In: IEEE International Conference on Robotics and Automation; 2011; Shanghai, China. New York, NY, USA: IEEE. pp. 674-679.

[3] Majewicz SPM, vanVledder MG, Lin M, Choti MA, Song DY, Okamura AM. Behavior of tip-steerable needles in ex vivo and in vivo tissue. IEEE T Bio-Med Eng 2012; 59: 2705-2715.

[4] Moon Y, Choi J. A compliant parallel mechanism for needle intervention. In: Annual International Conference of the IEEE Engineering in Medicine and Biology Society; 2013; Osaka, Japan. New York, NY, USA. IEEE. pp. 4875-4878.

[5] Fischer GS, Iordachita I, Csoma C, Tokuda J, DiMaio SP, Tempany CM, Hata N, Fichtinger G. MRI-compatible pneumatic robot for transperineal prostate needle placement. IEEE-ASME T Mech 2008; 13: 295-305.

[6] Chung J, Cha HJ, Yi BJ, Kim WK. Implementation of a 4-DOF parallel mechanism as a needle insertion device. In: IEEE International Conference on Robotics and Automation; 2010; Anchorage, AK, USA. New York, NY, USA: IEEE. pp. 662-668.

[7] Goffin L, Bour G, Martel F, Nicolau S, Gangloff J, Egly JM, Bayle B. Design and in vivo evaluation of a robotized needle insertion system for small animals. IEEE T Bio-Med Eng 2013; 60: 2193-2204.

[8] Bebek O, Hwang MJ, Cavusoglu MC. Design of a parallel robot for needle-based interventions on small animals. IEEE-ASME T Mech 2013; 18: 62-73.

[9] Huang J, Triedman JK, Vasilyev NV, Suematsu Y, Cleveland RO, Dupont PE. Imaging artifacts of medical instruments in ultrasound-guided interventions. J Ultras Med 2007; 26: 1303-1322.

[10] Pondman KM, Fütterer JJ, ten Haken B, Kool LJS, Witjes JA, Hambrock T, Macura KJ, Barentsz JO. MR-guided biopsy of the prostate: an overview of techniques and a systematic review. Eur Urol 2008; 54: 517-527.

[11] DiMaio S, Pieper S, Chinzei K, Hata N, Haker S, Kacher D, Fichtinger G, Tempany C, Kikinis R. Robot-assisted needle placement in open MRI: system architecture, integration and validation. Comput Aided Surg 2007; 12: 15-24.

[12] Coste-Maniere E, Simmons R. Architecture, the backbone of robotic systems. In: IEEE International Conference on Robotics and Automation; 2000; San Francisco, CA, USA. New York, NY, USA: IEEE. pp. 67-72.

[13] Albus JS, Lumia R, McCain H. Hierarchical control of intelligent machines applied to space station telerobots. IEEE T Aero Elec Sys 1988; 24: 535-541.

[14] Brooks R. A robust layered control system for a mobile robot. IEEE T Robotic Autom 1986; 2: 14-23.

[15] Musliner DJ, Durfee EH, Shin KG. CIRCA: A cooperative intelligent real-time control architecture. IEEE T Syst Man Cyb 1993; 23: 1561-1574.

[16] Bellingham J, Leonard J. Task configuration with layered control. In: IARP Workshop on Mobile Robots for Subsea Environments; 1994; Monterey, CA, USA.

[17] Ahmad A, Cavusoglu MC, Bebek O. Calibration of 2D ultrasound in 3D space for robotic biopsies. In: International Conference on Advanced Robotics; 2015; İstanbul, Turkey. New York, NY, USA: IEEE. pp. 40-46.

[18] Ahmad A, Orhan SO, Yildirim MC, Bebek O. Development and 3D spatial calibration of a parallel robot for percutaneous needle procedures with 2D ultrasound guidance. Journal of Medical Robotics Research 2017; 2: 1750007.

[19] Kaya M, Senel E, Ahmad A, Orhan O, Bebek O. Real-time needle tip localization in 2D ultrasound images for robotic biopsies. In: International Conference on Advanced Robotics; 2015; İstanbul, Turkey. New York, NY, USA: IEEE. pp. 47-52.

[20] Sharma K, Newman WS, Weinhous M, Glosser G, Macklis R. Experimental evaluation of a robotic image-directed radiation therapy system. In: IEEE International Conference on Robotics and Automation; 2000; San Francisco, CA, USA. New York, NY, USA: IEEE. pp. 2913-2918.

[21] Schweikar GG, Bodduluri M, Murphy MJ, Adler JR. Robotic motion compensation for respiratory movement during radiosurgery. Comput Aided Surg 2000; 5: 263-277.

[22] Riviere C, Thakral A, Iordachita I, Mitroi G, Stoianovici D. Predicting respiratory motion for active canceling during percutaneous needle insertion. In: Proceedings of the 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society; 2011; İstanbul, Turkey. New York, NY, USA: IEEE. pp. 3477-3480.

[23] Trejos L, Salcudean S, Sassani F, Lichtenstein S. On the feasibility of a moving support for surgery on the beating heart. In: Conference on Medical Image Computing and Computer-Assisted Intervention; 1999; Cambridge, UK. Berlin, Germany: Springer. pp. 1088-1097.

[24] Anderson D, Moore JB. Linear Optimal Control. Upper Saddle River, NJ, USA: Prentice Hall, 1971.

[25] Kaya M, Senel E, Ahmad A, Bebek O. Visual tracking of biopsy needles in 2D US images. In: IEEE International Conference on Robotics and Automation; 2016; Stockholm, Sweden. New York, NY, USA: IEEE. pp. 4386-4391.

[26] Bowthorpe M, Tavakoli M, Becher H, Howe R. Smith predictor-based robot control for ultrasound-guided teleoperated beating-heart surgery. IEEE J Biomed Health Inform 2014; 18: 157-166.

[27] Bebek O. Robotic-assisted beating heart surgery. PhD, Case Western Reserve University, Cleveland, OH, USA, 2008.

[28] Kwon WH, Han SH. Receding Horizon Control: Model Predictive Control for State Models. London, UK: Springer Science & Business Media, 2006.

[29] Camacho EF, Alba CB. Model Predictive Control. London, UK: Springer, 2013.

[30] Bebek O, Cavusoglu MC. Intelligent control algorithms for robotic-assisted beating heart surgery. IEEE T Robot 2007; 23: 468-480.