

Swept laser based 3D pose detection of the swinging robot Acroboter

Roland Reginald Zana
Department of Applied Mechanics
Budapest University of Technology and Economics
Budapest, Hungary
zana_r@mm.bme.hu

Ambrus Zelei
MTA-BME Research Group on
Dynamics of Machines and Vehicles
Budapest, Hungary
zelei@mm.bme.hu

Abstract—Domestic robots have been attracted growing interest. The Acroboter is a ceiling based crane-like robot concept, which utilizes the entire cubic volume of indoor environments. The benefits of cranes and drones are combined in our concept: the fan actuators make the robot agile in the horizontal direction, and the hoist rope provides the suspending force without power input when the robot is still. The lightweight structure is composed of 3D printed components and carbon balsa composite sandwich base plate. Our novel pose detection approach is to attach the HTC Vive measurement unit rigidly to the swinging unit of the robot. The working principle is based on sweeping beams of laser and time measurement providing high accuracy. We demonstrate in experiments that this novel pose detection concept is feasible for robots and it is accurate and fast enough for achieving stable trajectory tracking control of our crane-like manipulator.

Keywords—cable suspended robot, model predictive control, trajectory tracking, underactuation, multibody dynamics, pose estimation

I. INTRODUCTION

Robots and robot control algorithms are widespread and greatly developed: the structure and the control of industrial robots haven't change in the past few decades. However, new robots with specific purposes are continuously developed, such as legged robots, flexible robots, human-friendly lightweight robots, cable tethered robots, ceiling based robots and flying/underwater robots [1].

Model predictive control (MPC) is an efficient approach for the motion control of these state-of-the-art robots because of its accuracy, good trajectory tracking performance and relatively low computational effort. In case of mechanical systems, MPC basically means that the torques of the servo drivers are computed in such a way that results the prescribed output motion. The complete dynamical model of the controlled robotic structure is included in the core of the computation algorithms. This approach is often referred as computed torque control (CTC) method in robotics [1], [2].

A great portion of robots are underactuated; the development of their MPC is mathematically more challenging than the control of fully actuated robots. By definition, underactuated dynamical systems have less number of independent control inputs than degrees-of-freedom (DoF) [3], [4], [5]. One of the most expressive underactuated examples is the mathematical pendulum model of a gantry crane, where the position of the upper mounting point of the suspension cable can only be directly controlled, while the hanging payload performs a swinging motion [6]. Even so, the positioning of the payload is the control task. The problem of underactuation occurs in several other real-life applications, such as flexible robots,

unmanned air and underwater vehicles, robotic hands, legged locomotion systems and tethered robots.

Necessarily, underactuated systems have internal dynamics of which the motion is not specified by the control task. The model predictive control of underactuated systems cannot be achieved without the calculation of the internal dynamics. Thus, the dynamics of the full controlled mechanical system has to be considered and the resulting system of differential-algebraic equations (DAE) must be handled in the motion control algorithm [7], [8]. This is in contrast with fully actuated systems, where the control inputs can be expressed on purely algebraic way.

Besides underactuation, the complexity of the mechanical structure of robots can grow high. Multibody systems may possess high DoF and closed kinematic loops. A widely used approach for the mathematical description of these systems is to use redundant descriptor coordinates and equation of motion in DAE form [9]. The simulation techniques of multibody systems are elaborated in the literature; however their efficient MPC algorithms can be further developed.

Our goal is the further development of the underactuated test robot shown in Fig. 1 [10]. The robot is applicable for the test and quantitative benchmark of the novel underactuated DAE model based motion control algorithms. Our robot possesses similar structure of cranes; employs fan actuators and fast cable winches for actuation; and uses sweeping laser beams and inertial measurement units (IMU) for pose estimation.

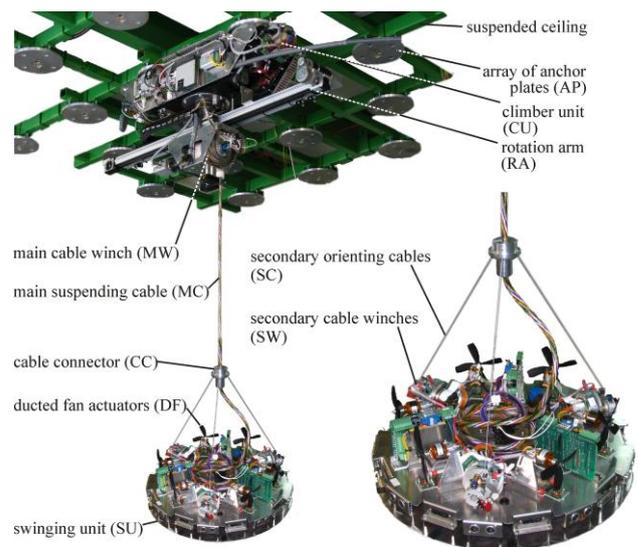


Fig. 1. Prior prototype of the Acroboter robot platform

II. BENCHMARKED CONTROL ALGORITHMS

Our proposed prototype is not only a promising design of a domestic robot, but it will be applied in scientific research too. This section surveys the control algorithms which will be subjected to benchmark tests on our robot prototype.

The benchmarked control algorithms will be compared in terms of a) computational effort (achievable digital sampling frequency); b) stability; c) robustness; d) trajectory tracking accuracy; e) control torque and/or force requirements; f) applicability for various type of controlled systems.

A. Overview

The CTC method can be used for motion control when the end-effector trajectory is prescribed. The CTC method requires the inverse kinematic and dynamic calculation [1], [2]. In our research work we apply the CTC method for underactuated multibody systems and we perform benchmark tests of the existing control approaches.

The application of the CTC method for underactuated multibody systems leads to a control law that is formulated in DAE form because of the following two reasons:

i) For underactuated systems, the application of the CTC leads to a DAE problem since the generalized coordinates of the system appears as differential variables and the control inputs appears as algebraic variables in the joined inverse dynamical and kinematical calculation [6], [7] and [11].

ii) Multibody systems, especially those which contain closed kinematic chain, cannot be efficiently modelled by the most commonly used minimum set of generalized coordinates. Instead, redundant set of descriptor coordinates are usually used together with the corresponding geometric constraints. Such numerically effective computation approach has been developed based on the so-called natural coordinates [9]. The geometric constraints between the redundant coordinates are represented by algebraic equations in the resulting DAE equation of motion.

B. Problem formulation with redundant coordinates

Let us consider the following general form of DAE equations of motion of multibody systems [3], [9]:

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C} + \mathbf{\Phi}_q^T \boldsymbol{\lambda} = \mathbf{H}\mathbf{u}, \quad (1)$$

$$\boldsymbol{\varphi} = \mathbf{0}, \quad (2)$$

where $\mathbf{q}(t) \in \mathfrak{R}^n$ is the vector of descriptor coordinates $\mathbf{M}(\mathbf{q}) \in \mathfrak{R}^{n \times n}$ is the mass matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathfrak{R}^n$ is the vector of the inertial forces, $\boldsymbol{\lambda}(t) \in \mathfrak{R}^m$ is the vector of Lagrange-multipliers and $\mathbf{\Phi}_q(\mathbf{q}) = \partial \boldsymbol{\varphi} / \partial \mathbf{q} \in \mathfrak{R}^{m \times n}$ is the constraint Jacobian associated with the geometric constraints $\boldsymbol{\varphi}(\mathbf{q}) \in \mathfrak{R}^m$. $\mathbf{H}(\mathbf{q}) \in \mathfrak{R}^{n \times l}$ is the control input matrix and $\mathbf{u} \in \mathfrak{R}^l$ is the vector of independent control inputs. We assume that the dimension of the control input l is less than the $n - m$ number of DoFs.

Unique solution exists for the inverse kinematic and dynamic calculations if l equals to the dimension of the task [7]. This yields to the assumption that the task is defined by l number of servo-constraint equations (control-constraints) $\boldsymbol{\sigma}(\mathbf{q}, t) \in \mathfrak{R}^l$ [12], [13], [14]:

$$\boldsymbol{\sigma} = \mathbf{0}. \quad (3)$$

We assume that the servo-constraints can be satisfied with bounded control forces.

C. ODE reformulation

Most of the control algorithms are applicable for mathematical models given in the form of ordinary differential equations (ODE). In order to reach this, one may choose to transform the DAE equations of motion (1) and (2) into ODE form. In such case, the constraint forces, mathematically the Lagrange-multipliers, are eliminated. This is also called as DAE index-reduction [9], [15], [16].

The core idea of the method of Lagrange-multipliers is to reformulate the geometric constraints (2) on acceleration level by differentiating them twice with respect to time:

$$\mathbf{\Phi}_q \ddot{\mathbf{q}} + \dot{\mathbf{\Phi}}_q \dot{\mathbf{q}} = \mathbf{0}. \quad (4)$$

We use (4) instead of the position level constraint equation (2). The Lagrange-multipliers are then expressed in closed form:

$$\boldsymbol{\lambda} = (\mathbf{\Phi}_q \mathbf{M}^{-1} \mathbf{\Phi}_q^T)^{-1} (\mathbf{\Phi}_q \mathbf{M}^{-1} (\mathbf{H}\mathbf{u} - \mathbf{C}) + \dot{\mathbf{\Phi}}_q \dot{\mathbf{q}}). \quad (5)$$

After substituting (5) back into the equation of motion (1), the acceleration $\ddot{\mathbf{q}}$ can be expressed explicitly. However, we notice that the acceleration level constraints in the resulting ODE are unstable. Therefore it cannot be used for time integration; only for the calculation of the control input in a certain time instant.

An alternative possibility to transform the equation of motion into ODE form is the projection of the equation of motion (1) into the subspace of motion which is admissible by the constraints (2) [15]. The projection below results the equation of motion admitted by the geometric constraints:

$$\mathbf{P}_a^T (\mathbf{M}\ddot{\mathbf{q}} + \mathbf{C} - \mathbf{H}\mathbf{u}) = \mathbf{0}. \quad (6)$$

After the projection, we get rid of constraint forces and Lagrange-multipliers. Projection matrix \mathbf{P}_a is calculated as:

$$\mathbf{P}_a = \mathbf{I} - \mathbf{\Phi}_q^\dagger \mathbf{\Phi}_q, \quad (7)$$

where $\mathbf{\Phi}_q^\dagger$ is the Moore-Penrose pseudo-inverse of the constraint Jacobian and \mathbf{I} is the identity. Paper [15] introduces a pseudo-inverse calculation which avoids inconsistency in terms of the dimensions of the descriptor coordinates:

$$\mathbf{\Phi}_q^\dagger = \mathbf{L}^{-1} (\mathbf{\Phi}_q \mathbf{L}^{-1})^\dagger, \quad (8)$$

where \mathbf{L} is the Cholesky-decomposition of the mass matrix \mathbf{M} .

The ODE reformulation makes possible to use the control methods which are developed for underactuated systems of which the mathematical representation uses non-redundant set of general coordinates.

D. Partial feedback linearization

An alternative control approach for underactuated systems requires their partial feedback linearization (PFL). By a specific transformation, the original nonlinear system is partially substituted with an equivalent linear system. For the

application of the PFL, the controlled dynamical system has to be formulated as equations (9) and (10) shows [5], [17]. Therefore the Lagrange-multipliers are eliminated from the equation of motion before the application of PFL, as it has been explained in section II/C.

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}, \quad (9)$$

$$\mathbf{y} = \mathbf{h}(\mathbf{x}), \quad (10)$$

where \mathbf{x} is the vector of state variables, \mathbf{u} is the control input and \mathbf{y} is the output vector. The control input can be formulated as after PFL:

$$\mathbf{u} = \mathbf{a}(\mathbf{x}) + \mathbf{b}(\mathbf{x})\mathbf{v}, \quad (11)$$

which results a linearized system as a cascade of integrators, and a synthetic input \mathbf{v} . The synthetic input can be chosen arbitrarily, e.g. linear compensator [17].

E. Computed Desired Computed Torque Control method

The generalization of the CTC method for underactuated systems is available in [18] for dynamical systems described by non-redundant set of generalized coordinates. For the application of the CDCTC, again, the Lagrange-multipliers have to be eliminated from the equation of motion as section II/C explains.

The approach is called Computed Desired Computed Torque Control method (CDCTC), where the expression ‘‘computed desired’’ refers to the fact that the so called uncontrolled coordinates cannot arbitrarily be prescribed, since they depend on the internal dynamics. The approach necessitates the separation of general coordinates into the set of controlled and uncontrolled coordinates. The controlled coordinates are directly prescribed by the control task, while the uncontrolled ones are not specified.

The ODE equation of motion is projected into the subspace of the uncontrolled motion by the null-space of the control input matrix \mathbf{H} . The projected ODE set is solved for the desired values of the uncontrolled coordinates and the control input is then calculated from the original equation of motion.

F. Method of Lagrange-multipliers with servo-constraint stabilization

The servo-constraint [12], [13], [14], which has been introduced in (3), is handled similarly to the geometric constraint (2). Both are expressed in acceleration level, on which the servo-constraint reads:

$$\mathbf{G}_q \ddot{\mathbf{q}} + \dot{\mathbf{G}}_q \dot{\mathbf{q}} + \dot{\mathbf{c}} = \mathbf{0}, \quad (12)$$

where $\mathbf{G}_q \in \mathfrak{R}^{kn}$ is the Jacobian of the servo-constraint σ and \mathbf{c} is the time derivative of the explicitly time dependent part of σ . We extend the acceleration level servo-constraint equation (12) with the Baumgarte-stabilization terms [9], [19] as follows:

$$\mathbf{G}_q \ddot{\mathbf{q}} + \dot{\mathbf{G}}_q \dot{\mathbf{q}} + \dot{\mathbf{c}} + K_D(\mathbf{G}_q \dot{\mathbf{q}} + \mathbf{c}) + K_P \sigma = \mathbf{0}. \quad (13)$$

We incorporate the dynamic equation (1), the acceleration level geometric constraint equation (4) and the

acceleration level servo-constraint equation (13) in the following hyper-matrix form:

$$\begin{bmatrix} \mathbf{M} & \Phi_q^T & -\mathbf{H} \\ \Phi_q & \mathbf{0} & \mathbf{0} \\ \mathbf{G}_q & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \lambda \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} -\mathbf{C} \\ -\dot{\Phi}_q \dot{\mathbf{q}} \\ -\dot{\mathbf{G}}_q \dot{\mathbf{q}} - \dot{\mathbf{c}} - K_D(\mathbf{G}_q \dot{\mathbf{q}} + \mathbf{c}) - K_P \sigma \end{bmatrix}. \quad (14)$$

The state given by \mathbf{q} and $\dot{\mathbf{q}}$ is measured in each digital time sample and the control input is calculated by means of equation (14).

The advantage of this method is that it is directly applicable for the DAE equation of motion and the servo-constraints without any transformation. The disadvantage is that the coefficient hyper-matrix of the unknowns $\ddot{\mathbf{q}}$, λ and \mathbf{u} is not invertible if the system is non-collocated. The reference [18] gives definition for collocated and non-collocated underactuated systems.

G. Direct discretization

The direct discretization of the DAE system using the backward Euler scheme is an alternative method, which is directly applicable for the control problem formulated by (1), (2) and (3). The discretization is applied for the first order form of (1) and the acceleration level reformulation of the servo constraints given by (13):

$$\frac{\mathbf{q}_i - \mathbf{q}_{i-1}}{h} = \mathbf{y}_i, \quad (15)$$

$$\frac{\mathbf{y}_i - \mathbf{y}_{i-1}}{h} = -\mathbf{M}^{-1}(\mathbf{C} + \Phi_q^T \lambda_i - \mathbf{H} \mathbf{u}_i), \quad (16)$$

$$\boldsymbol{\varphi} = \mathbf{0}, \quad (17)$$

$$\mathbf{G}_q \frac{\mathbf{y}_i - \mathbf{y}_{i-1}}{h} + (\dot{\mathbf{G}}_q + K_D \mathbf{G}_q) \frac{\mathbf{q}_i - \mathbf{q}_{i-1}}{h} + \dot{\mathbf{c}} + K_D \mathbf{c} + K_P \sigma = \mathbf{0}, \quad (18)$$

where \mathbf{q}_{i-1} and \mathbf{y}_{i-1} are the measured or estimated values of the descriptor coordinates and velocities in the current time-stamp respectively. The time step size is h . Matrices \mathbf{M} , Φ_q , \mathbf{H} , \mathbf{G}_q , $\dot{\mathbf{G}}_q$ and vectors $\boldsymbol{\varphi}$, σ , \mathbf{c} , and $\dot{\mathbf{c}}$ are evaluated with \mathbf{q}_{i-1} and \mathbf{y}_{i-1} . The resulting set of nonlinear algebraic equations (15)-(18) are solved by the Newton-Raphson method for the desired coordinates \mathbf{q}_i , the velocities \mathbf{y}_i , the Lagrange-multipliers λ_i and for the control inputs \mathbf{u}_i , which will be valid in the upcoming time-stamp [6], [11]. The Newton-Raphson method gives accurate result in maximum of three iterations, because the initial guess is given by the calculated values of the previous time-stamp.

H. Least square error based predictive method

A recently developed least square error based semi-analytic predictive control algorithm [20], [21] will be included in the benchmark. The control input is formulated in the form of polynomial functions of which the coefficients are tuned along the motion such that the difference between the realized and prescribed motion is minimized. The predictive control algorithm finds the motion of the controlled system in polynomial form:

$$\mathbf{q}_c = \mathbf{P}\boldsymbol{\gamma}, \quad (19)$$

where $\mathbf{q}_c(t) \in \mathfrak{R}^l$ is the vector of controlled coordinates, $\mathbf{P} \in \mathfrak{R}^{l \times n}$ is the matrix of unknown coefficients and $\gamma(t) \in \mathfrak{R}^n$ contains n number of polynomial functions of time. The main idea is to determine the control input by means of the variational principle, with which the control task is satisfied most accurately. The time integral of the error is minimized over a certain time horizon from t_0 to t_e :

$$J = \int_{t_0}^{t_e} \mathbf{E}^T \mathbf{E} dt, \quad (20)$$

where \mathbf{E} is the error between the realized and the desired trajectories. In each time-stamp, the control input is determined in the form of an analytical function for which J in (20) is minimal. In the upcoming digital sample, the time interval $[t_0, t_e]$ is shifted by the time step size h , which is significantly smaller than the interval $[t_0, t_e]$.

I. Experimental analysis of digital effects

In case of every digitally controlled dynamic system, the issue of digital effects such as quantization in time and space and time delay are important issues that affect stability [22], [23]. The dry friction, which emerges in servo drives and bearings, together with the digital effect often cause unexpected instability and unwanted vibrations [24], [25], which demand thorough insight if one aims to avoid it. These phenomena will be experimentally studied on the Acroboter robot platform.

III. ROBOT PROTOTYPE

The Acroboter platform consists of two main units: the Swinging Unit (SU) and the Climber Unit (CU), as it is shown in Fig. 1.

The new prototype of the SU, which is shown in Fig. 2, is optimized to minimum weight. The goal was to reduce the mass from 10 kg to about ~3 kg, which was achieved. Many components, which were integrated to the previous version of the SU, have been simplified or removed. In the current setup, the power electronics has been moved to the CU, which provides 24 V for power electronics and 5 V for the logic circuits. The ultrasonic distance sensors for obstacle avoidance, the photodiodes together with phototransistors for the RPM measurement of the propellers, and the related electronics have been removed from the SU. The base plate material is changed from aluminium to a lightweight carbon balsa composite sandwich material, see Fig. 2. The suspensions of the fans are manufactured with rapid prototyping (3D printing) from PLA with medium fill density, see Fig. 4. Through these modifications, the desired mass is achievable.

A. Mechanical structure and actuation

The mechanical structure is depicted in Fig. 1 and Fig. 2. The length of the main suspending cable and the secondary orienting cables of the SU are varied by servo motors, which means the system has 4 actuators. The positioning is compensated by six fan actuators. The fans provide an arbitrary direction resultant force in the plane of the SU base plate and a resultant torque around the axis perpendicular to the base plate SU. These represent 3 further independent control input. Despite the 7 actuators, the system is still underactuated, because the number of actuators is lower than the total number of DoF, which is 9, because the cable

connector, which is modelled as a particle, represents 3 DoF and the SU, which is modelled as a rigid body in the space, represents another 6 DoF.

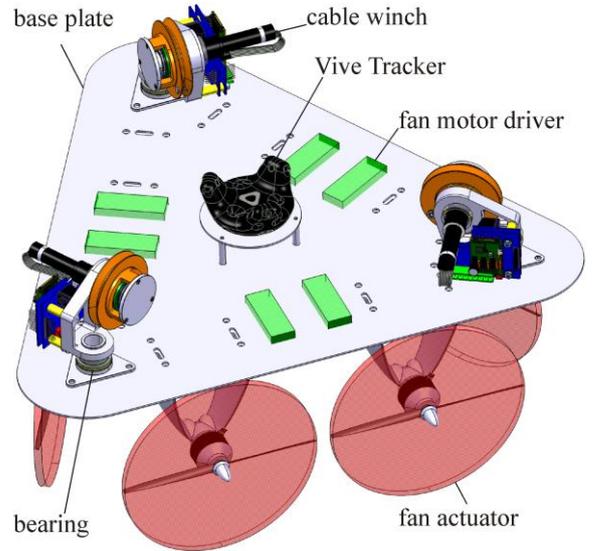


Fig. 2. Lightweight design of the Swinging Unit prototype

The Acroboter is kinematically redundant. We note that underactuated robots are necessarily kinematically redundant [26]. Fully actuated manipulators equipped with more internal DoFs than required to perform a specified task are called kinematically redundant [2]. For such systems the inverse kinematic calculation is not unique. However, the joined inverse kinematic and inverse dynamic calculation of underactuated systems is unique if the dimension of the task is equal to the number of independent control inputs. Both are less than the number of DoF; therefore underactuated systems are kinematically redundant too and the kinematic redundancy is resolved by the consideration of the internal dynamics. However, the Acroboter platform consists further kinematically redundant DoFs.

B. Actuators

The secondary cable winches are fixed via bearings (see Fig. 3); therefore they are able to rotate around the axis perpendicular to the SU base plate. Brushless EC16 60 W Maxon motors with planetary gear-head are used for the winches.

The fan motors, which are shown in Fig. 4, are changed from BL 2212/10 180W to MN3110-17 700kv type Tiger Motors which has more than twice maximum continuous power: 466 W. High efficiency two bladed carbon fibre propellers are applied with 8 inch diameter and 5.5 inch pitch. The propellers are clockwise – counter clockwise pairs and rotates in opposite directions in order to eliminate the resulting rotational torque on the base plate and gyroscopic effects.

C. Sensing and pose estimation

The position tracking of the base plate is realized by one of the HTC Vive's [27] virtual reality system's accessory: the Vive Tracker. The Tracker is attached to the upper side of the base plate via vibration isolation. The positioning of the Tracker provides better field of view. Two laser emitters, which are called Lighthouses, alternately send out

horizontal and vertical infrared laser sweeps spanning 120° in each direction. On the surface of the Tracker there are photodiodes that indicate when the laser hits them. The difference in time at which the various photodiodes are hit allows recovery of the position and orientation. The operation of the Tracker is aided by IMU. The details and the working principle of the Vive Tracker are summarized in the followings based on [28] and [29].

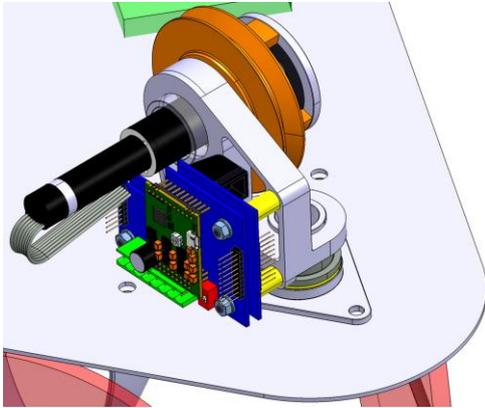


Fig. 3. Cable winch mechanism combined with angular position detection.



Fig. 4. Fan actuator.

The alternating laser sweeps of base stations A and B (Lighthouses) operate in the following sequence:

- the vertical laser of station A sweeps from left to right,
- half a revolution or 8.333 ms later, the horizontal laser of station A sweeps from the bottom to the top,
- 8.333 ms later, lasers of station A turn off and the vertical laser of station B sweeps from left to right,
- 8.333 ms later, the horizontal laser of station B sweeps from the bottom to the top,
- lasers of station B are turned off, and lasers of A turn on, and the entire sequence begins again.

Each base station contains a flashing LED array in order to allow them to be synchronized with each other. The LED arrays flash a wide-angle synchronization pulse at the beginning of each 8.333 ms laser sweep period.

The position and orientation estimation is aided by IMUs. The high sampling rate data of the IMUs are integrating with the measured values via dead reckoning. When a laser sweep occurs, the base stations merely correct the positional and orientation drift. It is unavoidable in integrating noisy and biased measurements.

References [28] report that measurements shown that the jitter of the system, was about 0.3 mm in the case of the headset. The jitter is the imprecision of the measurements

when the measured object is standing completely still. It may become 2.1 mm when the tracked object is seen by only one Lighthouse, which has less probability in the case of the Vive Tracker because of the placement of its sensors. Measurements, which were also done related to the systems accuracy, resulted about RMS 2 mm precision.

Since, each base station must be placed facing the other at a maximum distance of 5 m, this is the maximum size of the working area, which is satisfactory for our measurements.

For higher, possibly sub millimetre precision, an OptiTrack camera kit [30] could be a more expensive but possible alternative solution instead of HTC Vive.



Fig. 5. Vive tracker for pose estimation [27].

The cable winch motors are equipped with hall sensors for angular position detection. They can provide 6 different signal combinations per turn. Furthermore digital rotary incremental encoders with 512 counts per turn provide the data for velocity measurement. The cable length data are utilized for the enhancement of the pose estimation.

D. Control implementation and communication

The electronic equipments are chained on a half-duplex RS422 bus, including the controllers of the fans, the controller unit of the winch motors. The RS422 bus ends in a PC on which the high level trajectory tracking control runs.

The Tracker is connected by wireless interface to a dongle. This dongle is used to transfer tracking data from the VIVE Tracker to a PC.

The control algorithms, which were summarized in Section II, have been implemented in MatLab. MatLab provides a convenient environment for control development.

IV. CONCLUSION

The further development of the already existing Acroboter robot prototype was summarized. The development steps were taken towards lightweight construction and towards more precise and easy-to-use sensory system. Several underactuated control approaches were surveyed, which will be benchmarked on the Acroboter robot platform in future work.

ACKNOWLEDGMENT

This research was supported by the Hungarian National Research, Development and Innovation Office (Project

identifier: NKFI-FK18 128636) and the MTA-BME Research Group on Dynamics of Machines and Vehicles.

REFERENCES

- [1] B. Siciliano, O. Khatib, Springer Handbook of Robotics. Springer-Verlag Berlin, Heidelberg, 2007.
- [2] M. W. Spong and M. Vidyasagar, Robot Dynamics and Control, John Wiley & Sons, 1989.
- [3] H. Goldstein, Classical Mechanics, 2nd ed. Addison-Wesley, 1980.
- [4] A. De Luca, R. Mattone, G. Oriolo: Control of underactuated mechanical systems: Application to the planar 2r robot. In: 35th IEEE Conf Decision Contr, Kobe, Japan, pp 1455–1460, 1996.
- [5] M. W. Spong, “Underactuated mechanical systems”, Control Problems in Robotics and Automation. Lecture Notes in Control and Information Sciences, vol 230. Springer, Berlin, Heidelberg, 1998, doi: /10.1007/BFb0015081.
- [6] A. Zelei, L. L. Kovács, G. Stépán, “Computed torque control of an under-actuated service robot platform modeled by natural coordinates,” Commun Nonlinear Sci Numer Simulat, 16(5):2205–2217, 2011.
- [7] W. Blajer, K. Kolodziejczyk, “Modeling of underactuated mechanical systems in partly specified motion,” Journal of Theoretical and Applied Mechanics, 46(2):383–394, 2008.
- [8] R. Seifried, Dynamics of Underactuated Multibody Systems: Modeling, Control and Optimal Design, Springer, 2014.
- [9] J.G. de Jalón, E. Bayo, Kinematic and dynamic simulation of multibody systems: the realtime challenge, Springer-Verlag, 1994.
- [10] G. Stépán et al, “A ceiling based crawling, hoisting and swinging service robot platform,” in Proceedings of Beyond Gray Droids: Domestic Robot Design for the 21st Century Workshop at HCI 2009, 2009.
- [11] L. L. Kovács, A. Zelei, L. Bencsik, J. Turi, and G. Stépán, “Motion control of an under-actuated service robot using natural coordinates,” in Proceedings of ROMANSY 18 Robot Design, Dynamics and Control, 2010, pp. 331–338.
- [12] W. Blajer and K. Kolodziejczyk, “A geometric approach to solving problems of control constraints: Theory and a dae framework,” Multibody System Dynamics, 11(4):343–364, 2004.
- [13] W. Blajer and K. Kolodziejczyk, “Control of underactuated mechanical systems with servoconstraints,” Nonlinear Dynamics, 50(4):781–791, 2007.
- [14] P. Masarati, M. Morandini, A. Fumagalli, “Control Constraint of Underactuated Aerospace Systems”, ASME J. Comput. Nonlinear Dyn., 9(2):021014, April 2014, doi:10.1115/1.4025629.
- [15] J. Kövecses, J.-C. Piedoboeuf, and C. Lange, “Dynamic modeling and simulation of constrained robotic systems,” IEEE/ASME Transactions on mechatronics, vol. 8, no. 2, pp. 165–177, 2003.
- [16] L.L. Kovács and L. Bencsik, “Stability case study of the acroboter underactuated service robot,” Theoretical and Applied Mechanics Letters, 2(4):043004, 2012.
- [17] A. Isidori, Nonlinear Control Systems, 2nd ed. Springer, 1999.
- [18] I. M. M. Lammerts, Adaptive Computed Reference Computed Torque Control, Ph.D. thesis, Eindhoven University of Technology, 1993.
- [19] J. Baumgarte, “Stabilization of constraints and integrals of motion in dynamical systems,” Computer methods in applied mechanics and engineering, 1(1):1–16, 1972.
- [20] L. Bencsik, B. Bodor, L. L. Kovács, “Predictive trajectory tracking of under-actuated systems”, in proc.: Joint International Conference on Multibody Dynamics, Montreal, Canada, 2016., paper a305.
- [21] D. J. Nagy, L. Bencsik, T. Insperger, “Identification of the model of stick balancing using the cepstral analysis”, in proc.: DSTA 2017: Mathematical and Numerical Aspects of Dynamical System Analysis, Lodz, Poland, 2017., pp. 405-412.
- [22] G. Stépán, A. Steven, and L. Maunder. Design principles of digitally controlled robots. Mechanism and Machine Theory, 25(5):515-527, 1990.
- [23] G. Stépán. Vibrations of machines subjected to digital force control. International Journal of Solids and Structures, 38(10-13):2149-2159, 2001.
- [24] Budai, C., Kovács, L.L., Kövecses, J., Stépán, G.: Effect of dry friction on vibrations of sampled-data mechatronic systems. Nonlinear Dynamics, 88(1): 349-361, 2017. <https://doi.org/10.1007/s11071-016-3246-7>.
- [25] Csaba Budai, László L. Kovács and József Kövecses: Combined Effect of Sampling and Coulomb Friction on Haptic Systems Dynamics. J. Comput. Nonlinear Dynam 13(6), 061005 (10 pages), 2018. Doi: 10.1115/1.4039962.
- [26] A. Zelei, L. Bencsik, L. L. Kovács, G. Stépán: Redundancy Resolution of the Underactuated Manipulator ACROBOTER. In proc. RoManSy 2012 - 19th CISM-IFTToMM Symposium on Robot Design, Dynamics and Control, June 12-15, 2012, Paris, France, pp. 233-240.
- [27] Official Vive homepage: <https://www.vive.com/us/vive-tracker/>. Last accessed: 10. September, 2018.
- [28] Doc-Ok homepage: <http://www.doc-ok.org/?p=1478>. Last accessed: 10. September, 2018.
- [29] D.C. Niehorster, L. Li, M. Lappe, “The Accuracy and Precision of Position and Orientation Tracking in the HTC Vive Virtual Reality System for Scientific Research”. Iperception 8(3): 2041669517708205, 2017. Doi: 10.1177/2041669517708205.
- [30] OptiTrack official homepage: optitrack.com. Last accessed: 10. September, 2018