UNIVERSITY OF MISKOLC
FACULTY OF MECHANICAL ENGINEERING AND INFORMATICS

DEVELOPMENT OF A MACHINE VISION SYSTEM FOR ANTI-SWAY CONTROL OF OVERHEAD CRANE

PHD THESES

Prepared by

Marouane Hmoumen
Electrical Engineering (BSc)
Renewable Energy and Storage (MSc)

ISTVÁN SÁLYI DOCTORAL SCHOOL OF MECHANICAL ENGINEERING SCIENCES
DESIGN MACHINES AND STRUCTURES
DESIGN OF MECHATRONICS SYSTEMS

Head of Doctoral School

Vadászné Dr. Gabriella Bognár
DSc, Full Professor

Head of Topic Group

Dr. Tamás Szabó

Scientific Supervisor

Dr. Tamás Szabó

Miskolc
2021

CONTENTS

SUPERVISOR'S RECOMMENDATIONS

Date: 03.02.2021 Miskolc

Supervisor: Dr. Tamás Szabó

Marouane Hmoumen asked for his admission to the István Sályi Doctoral School in January of 2017 after he had spent the previous semester in Szent István University in Gödöllő. He had no supervisor's support and no subjects to learn in Gödöllő. Therefore he lost one semester out of eight ones. This fact explains why he had to extend his PhD studies.

The topic of machine vision related to mechatronics was offered him to investigate in Miskolc. Since he had a university degree in electrical engineering the topic was close to him. The scientific problems of machine vision were induced in the Robert Bosch Energy and Body Systems Ltd. The company has more than one hundred machine vision applications. One of the hottest problem they was interested is the character recognition in noisy circumstances, i.e., when the letters and numbers are scratched. From the very beginning he made researches in machine vision literatures and he learned to use the OpenCV library. During the last three years he regularly attended the Robert Bosch Energy and Body Systems Ltd. to resolve the problems of character recognitions. In aware of the theory he could find the appropriate parameters which were available in the machine vision systems. Then all the systems recognized the characters correctly without further research development. Therefore his research topic was modified to "Development of a machine vision system for anti-sway control of overhead crane".

After the complex exam, he progressed very well on his modified research topic. The machine vision displacement sensor was successfully developed and built into a model of an experimental overhead crane. In 2019, he participated in a conference in Pécs where he reported on his PhD research and we jointly wrote a Q3 article for the Pollack Periodica journal, which was already published. Unfortunately, his research work came to a temporaly halt during the spring due to the COVID-19 emergency.

As for the feasibility of the research plan, we have already prepared an article on overhead crane control, which provides the backbone of his PhD research. He already have submitted a Q2 article to the Journal of Facta Universitatis, Series: Mechanical Engineering and its acceptance is in advanced state.

During his doctoral studies, Marouane Hmoumen proved that he was suitable for independent scientific research in the field of mechatronics. The preparation of his dissertation reflects careful work, the results of the thesis value achieved so far show well his commitment to the topic of mechatronics.

Dr. Tamás Szabó
Associate Professor
Supervisor

LIST OF SYMBOLS AND ABBREVIATIONS

SCALAR QUANTITIES

V_t	Volume of the structure at time t	
M	Mass of the payload	Kg
g	Gravity acceleration	m/s^2
e_{11}^t	Linear strain measured in axial direction at t	[-]
$\bar{u}$	Displacement in axial x local coordinate	[mm]
$\bar{v}$	Displacement in axial y local coordinate	[mm]
u	Displacement in axial x Cartesian coordinate	[mm]
v	Displacement in axial y Cartesian coordinate	[mm]
L^t	Deformed length of the element	
I, j	Node numbers	
$\bar{u}^e(\xi)$	Axial displacement along the element e	
$\bar{v}^e(\xi)$	y displacement along the element e	
$\bar{u}_i, \bar{u}_j, \bar{v}_i, \bar{v}_j$	Local nodal displacement increments	
u_i, u_j, v_i, v_j	Global Cartesian node displacement increments	
$\ddot{u}_i, \ddot{u}_j, \ddot{v}_i, \ddot{v}_j$	Global nodal accelerations	
N^t	Axial force of the truss element	
tol	Tolerance prescribed by the user	[-]
V_0	Volume of the structure at time 0	
$\ddot{u}_M^t$	Acceleration of mass M	
σ_{11}^0	Function of the axial stress at time 0	
R^t	Virtual work of the external force	
u	Prescribed position of the trolley (Fig. 3.1)	
L	Length of the pendulum	

Θ	Angle of the pendulum	[degree]
x_M	Horizontal absolute coordinate of the payload	[mm]
x_r	Relative positions of the payload	[mm]
T	Kinetic energy of the system	
V	Potential energy of the system	
Δt	Time step	[s]
v	Velocity of the load (Chapter 3)	[m.s]
L_g	Lagrangian function	
x_T	Target position of the payload	[mm]
τ	Time delay	[s]
k_1, k_2, k_3, k_4	Gain parameters	[-]
m_L	Mass of the chain	[kg]
$F(y)$	Force in the chain due to gravity	[N]
μ	Linear mass density of the chain	[Kg.m]
ξ	Local coordinate of element e	[-]
L^e	Length of one element (Fig. 3.5)	[mm]
x_2	Lateral displacement of the middle of the chain	
Ne	Number of element the chain is subdivided	[-]
y[k]	Lateral displacement of the payload	[mm]
e_x	Error of the position of the payload	[-]
I2	Position of the payload in pixel at t+1	Pixel]
I0	Position of the payload in pixel at t	Pixel]

VECTOR QUANTITIES

$\bar{\mathbf{M}}_e$	Element mass matrix
$\bar{\mathbf{K}}_{Le}$	Linear matrix
$\bar{\mathbf{K}}_{Ge}$	Geometric element stiffness matrices
$\mathbf{I}$	Unit matrix
$\delta \boldsymbol{u}_M$	2D vector of the virtual displacements of mass
$\bar{\mathbf{f}}_{Ge}$	External load vectors in element local coordinate system

$\bar{\mathbf{f}}_{\sigma e}$	Internal load vectors in element local coordinate system
$\bar{\mathbf{f}}_{\sigma e}$	Internal load vectors in element local coordinate system
$\mathbf{M}^{t+\Delta t}$	Structural mass matrix
ρ^{t}	Density of the material in the deformed configuration
σ_{11}^{t}	Cauchy stress of the one-dimensional structure at time t
$\delta\eta_{11}$	Variation of the quadratic portion of the nonlinear strain at t
ρ_0	Density of the material at time zero
δu_M	Virtual lateral displacements of mass
$\mathbf{K}_L^t$	Structural linear
$\mathbf{K}_G^t$	Structural geometric stiffness matrices
$\mathbf{f}_g^{t+\Delta t}$	Structural column vector of external loads
$\mathbf{f}_\sigma^t$	Structural column vector of internal loads
$\ddot{\mathbf{q}}$	Structural column vector of nodal accelerations
$\Delta\mathbf{q}$	Structural column vector of the nodal displacements
$\hat{\mathbf{f}}$	Effective load vector at t+Δt
$\tilde{\mathbf{x}}[k]$	Column vector of state variables
$\mathbf{A}_D, \mathbf{b}_D, \mathbf{l}_D, c$	Coefficient matrices
$\ddot{\boldsymbol{u}}^{t+\Delta t}$	2D vector of the accelerations at t+Δt
$\delta\boldsymbol{u}$	2D vector of the virtual displacements

OTHER SYMBOL

PLC	Programmable logic controller
OpenCV	Open-source computer vision library
NCTF	Nominal characteristic trajectory following
DPTOC	Double-pendulum-type overhead crane
4DoF	Four degree of freedom
IIR	Infinite impulse response
FIR	Finite impulse response

LQR	Linear quadratic response
MPC	Model predictive control
NN	Neural network
SMC	Sliding model control
DoR	Degree of freedom
RGB	Red-green-blue
HSV	Hue-saturation-value
SCL	Serial clock
SDA	Serial data
CFC	Continuous function chart
IDE	Integrated development environment
D/A	Digital-analog
RAM	Random-access memory
CPU	Computer central processing
I2C	Inter-integrated circuit
SPI	Serial peripheral interface
rpm	Revolution per minute
I/O	Input/output
MP	Megapixel camera
Fps	Frames per second
PWM	Pulse width modulation
ADRC	Active disturbance rejection control
PID	Proportional integral derivative
PDTSFC	Proportional derivative Takagi-Sugeno fuzzy control
PD	Proportional derivative
LQG	Linear quadratic Gaussian
GPC	Generalized predictive control
FEM	Finite element method
ZVDDD	Zero vibration and triple derivative

DZV	Distributed zero vibration
ZV	Zero vibration
NC	Network controller
PSO	Particle swarm optimization
GA	Genetic algorithm
RNN	Recurrent neural network
HEA	Hybrid evolutionary algorithm
LPV	Linear parameter varying

1. INTRODUCTION

In every level of economic activities, each industrial and commercial, we reach the handling equipment present in the transitional phase of the development cycles of materials and products. Also, they are significantly crucial in transportation [1].

Dealing with loads requires elite lifting gear that works securely, precisely, and safely. The accessibility, unwavering quality, and productivity of this gear are pivotal components whatever the application [1].

Cranes structure is one of the most broadly utilized solutions in the field of dealing with and transshipment. They take numerous forms and are truly versatile to multiple conditions [2].

The substantial development of the transporting tool market increases load volume to be dealt with, forcing progressively strict necessities regarding overhead cranes performance. These performances are expressed in the form of a compromise between the criteria of speed, execution, precision, and operator safety. Therefore, handling gantry crane or overhead cranes requires a lot of involvement from an experienced operator. In general, the computerization of production systems has been one reaction to this trade-off. This automation had a double target: to increase the system efficiency (cost reduction, reliability, accessibility, quality) and improved direct administrator security [3, 4, 5].

Overhead cranes and gantry cranes are handling equipment found mainly in the industry for the movement of loads, the distribution of manufactured materials, and the manufacturing industry for the assembly of equipment [2, 6, 7].

In industry, the smallest delay can be very costly. Speeding up an overhead crane is probably going to energize undesirable load swaying. If the load is swaying during the movement, it may be challenging to follow the desired path or avoid an obstacle. Once the system has reached the desired position, swaying will degrade positioning accuracy and cause a delay in operation. For these effects, the crane control functions must currently manage the load displacement and control the unwanted swaying of the load [8]. In addition, the robustness of the controllers to changes in load and external disturbances must be included [7].

In this work, we design a controller based on a linearized model of overhead cranes.

A simple mathematical model is investigated considering the time delay. The stability domains are expressed by the gain parameters for different time delays, and then a taut string model is also developed extending the pendulum model taking into consideration the vibration of the suspending chain. Its gain parameters are determined by the pole placement method utilizing the results of the simple mathematical pendulum model. An observer is designed based on the extended model to define the state variables related to the vibration of the chain.

In the proposed method, the vibration of the chain suspending the payload and its time delay is considered by an observer, which is programmed in an industrial Programmable Logic Controller (PLC). The motion of the payload is measured by a vision system that uses a Webcam, Visual Studio C++ and OpenCV library. The resulting controller has been verified on a laboratory overhead crane.

1.1 Control methods of the cranes

Cranes are essential for handling equipment in many industrial sectors. Hydraulic power stations, shipbuilding, armaments, cement factories, the iron and steel industry, household waste incineration plants are all examples that testify to the usefulness of these lifting and transshipment means. It moves or transfers a load at any point of a parallelepiped volume. They are designed to meet the needs of medium and heavy industrial lifting. These devices allow handling the most varied loads with the possible help of certain interchangeable accessories such as lifting beams, buckets, rotators, and forks.

Several types of cranes are employed in numerous industrial sectors. These embrace overhead cranes, tower cranes, and boom cranes. These cranes are generally classified into two groups, in line with their structures and motions: overhead cranes Fig. 1.1, boom cranes Fig. 1.2, and tower cranes Fig. 1.3.

Fig.1. 1 Overhead crane.

Fig.1. 2 Boom crane.

Fig.1. 3 Tower crane.

Generally, a crane consists of a hoisting mechanism (hoisting line and a hook) and a support mechanism (trolley). The payload hook cable gathering is attached from a point on the support mechanism. The support moves the payload around the crane workspace. At the same time, the raising system lifts and brings the load down to maintain a strategic distance from impediments in the way and store the payload at the objective point without causing an enormous swing [9].

Many researchers have suggested solutions by designing multiple controls scheme for a crane system to solve the control problems discussed. However, the majority of the current methods are not appropriate for realistic implementation. For that reason, most commercial cranes are not automatic and nevertheless rely on operators, who fail to atone for the swing. This failure can also additionally put the load and the surroundings in danger. Another crane automation problem is the crane surroundings character, which is frequently unstructured in shipyards and manufacturing facility floors. In the following, we focus on reviewing the overall techniques used in the controlling topic.

The crane operation steps automation is feasible, and some research studies have been pointed at this mission [10].

The most consuming task is to move the payload from the initial position to the target position and needs a skillful operator to perform it. Suitable techniques to facilitate moving the payload

fast without inducing large swings are focal of several researchers [6]. An overhead crane basic motion can be described as object hoisting or lowering, trolley travel, and lifting. To increase the production, the crane movements must take place at high speeds without generating undesirable sway of the payload and being safe to the surrounding. There are two main approaches for crane automation: the first is that the operator is retained in the loop, and the payload dynamics are adjusted to make his job more manageable by feeding back the payload sway angle and its ratio [11, 12]. This response provides a different trajectory to that generated by the operator. Another way is to not excite the payload close to its natural frequency by adding a filter to eliminate the undesirable input frequency [13]. This causes a delay between the crane input and the operator action, which may affect the system negatively. Another method is installing a mechanical absorber to the crane structure, which requires a large amount of energy, making it impractical.

The second approach is to take away the operator from the cycle, making the operation fully computerized. Different control methods can be used to achieve the above goal. Numerous researchers have commonly used open-loop control schemes Fig. 1.4, in particular for manipulating the payload swinging. There are different open-loop techniques: The input-shaping crane control method is one of the widely used open-loop methods in the literature [14, 15, 16]. It consists of a sequence of acceleration and deceleration pulses. These sequences are generated such that there is no residual swing at the end of the transfer operation. Cutforth et al. [17] designed an adaptive input shaping based upon flexible mode frequency variations to handle the uncertainties of the parameters. Also, an adaptive discrete-time control was proposed in [18], where the timing of impulses and amplitude are adapted to cater for the system uncertainties.

Vaughan et al. have presented a command shaping algorithm to adapt to the nonlinearities caused by the nonlinear slewing motion of a tower crane [19]. The authors introduce an input shaping control technic to limit the residual vibration. The method was experimentally tested for several cases, and its robustness to variations in suspension cable length and nonlinear effects during slewing was presented.

Another input shaping method was presented based on a distributed delay by Vyhlídal et al. [20] referred to as Distributed Zero Vibration (DZV) shaper, while other input shapers were developed based on a lump type delay. Numerous features of the shaper have been examined and compared with the ZV shaper. However, the DZV shaper offers a slower vibration suppression performance than the classical ZV shaper. The DZV shaper delivers smoother command and superior robustness at the high-frequency range.

Rhim et al. proposed a time-domain adaptive command shaping by a direct method. The desired command shaper coefficients can be estimated directly without getting the system parameters information [21]. The same author presented an adaptive time delay command shaping for multi-mode and single systems in another work [22].

Singhose et al. [23] have studied an input shaping model using an average travel length method for gantry cranes to control the hoisting effect. Other works have included an adaptive input shaping for 3D overhead cranes [24]. The input-shaping method implementation for sway control has been proposed in [25, 26, 27] for gantry cranes, overhead cranes [26] for bridge cranes [28], rotary cranes [28–31], and container cranes [32, 33, 34].

Ahmed et al. developed hybrid control schemes based on NCTF control with IIR low-pass filter techniques for input tracking and sway suppression of a DPTOC for overhead cranes [35], and gantry cranes [36]. Glossiotis et al. proposed a method based on the preprocessing of all the commanded crane inputs by designing a finite impulse response (FIR) filter for a four-degrees-of-freedom (4DoF) model of rotary crane system [37]. An FIR filter was implemented too for the control of the gantry crane. However, investigation results show that the IIR filters have no accurate phase and are frequently hard to control. Yet, the FIR filters have every time a linear stage and are simple to manage.

Command smoothing is another open-loop technique, which can significantly attenuate the vibrations via smoothing the original command [38]. The smoother is aimed to approximate the actual frequency and the amortization ratio of the system. Xie et al. proposed a technique that suppresses vibration in a flexible systems [39]. The method can be more insensitive to modeling error at higher frequencies while compared to the ZVDDD shaper, which has more insensitivity at lower frequencies.

Alghanim et al. [40] introduced a method to produce an optimal discrete-time shaped acceleration profile for concurrent hoist maneuvers and crane travel. The effectiveness of the technique was confirmed using a laboratory model of an overhead crane.

The open-loop techniques are simple to incorporate because there is no need for extra sensors to measure the payload displacement angle, which saves in terms of costs [41].

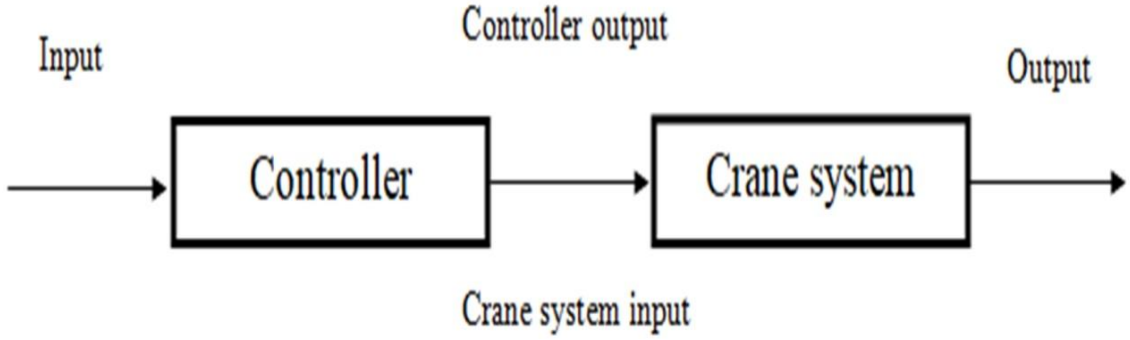

Fig.1. 4 Block diagram of open loop control system for a crane system.

However, this technique negative point is that it is susceptible to parameter variation and to external disturbances negatively affecting its performance [42]. Some of these external factors, as an example, are ocean waves and wind.

The closed-loop technique is a good control scheme that allows the crane system to regulate its performance based upon the preferred output response, see Fig. 1.5. Feedback structures utilize the system states measurement and approximation to decrease the oscillations and achieve precise positioning. Therefore, the closed-loop control schemes are not that sensitive to parameter variations and turbulences [43].

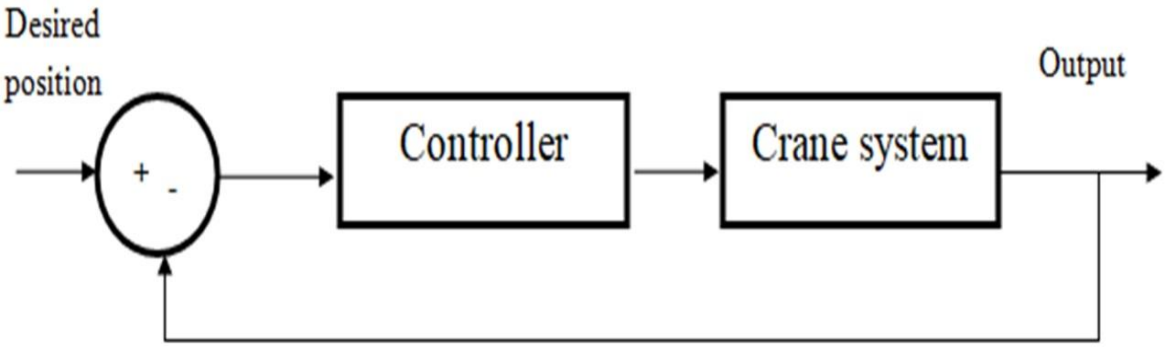

Fig.1. 5 Block diagram of open loop control system for a crane system.

The optimal control strategy MPC was used for controlling gantry cranes and overhead cranes [44–48]. The adaptive control method was proposed by numerous researchers in [49–54]. Lee et al. [55] proposed a combination of an NN and an SMC to achieve a precise trolley position and eliminate the payload sway angles. Nakazono et al. [56] proposed a three-layered neural network as a controller (NC) with genetic algorithm-based (GA-based) training that was able to control the load swing in both the circumferential and the radial directions, simultaneously for the rotary crane system. Furthermore, the proposed method was efficient and make the controller design simple than a traditional one. Duong et al. [57] proposed a method of controlling an underactuated three-dimensional tower crane system adopting a recurrent neural network (RNN) which is improved by

an evolutionary algorithm. The hybrid evolutionary algorithm (HEA) was developed by implanting the genetic operators of the GA (recombination and mutation) into the particle swarm optimization (PSO) to create an offspring. Simulation results illustraten that the proposed HEA has higher performance comparing with the canonical algorithms and that the control system works efficiently. Li et al. [58] have proposed a combination of an NN and a fuzzy on a special crane system which has combination of industrial arm and crane system worked on the ship. The learning algorithm that was used for the neural fuzzy controller was founded on an ant colony optimization. It displayed a rapid convergence of the performance compared to the back propagation algorithm.

Hilhorst et.al [59] proposed a numerical approach to design reduced-order multi-objective H-infinity controllers for discrete-time linear parameter varying (LPV) systems. The resulting method address any prefixed controller order, permits polynomial parameter dependencies of all system matrices, can consider several design objectives, and offers intuitive guide lines for the selection of an initial controller. The effectiveness of the approach with varying cable length is carried on lab scale overhead crane.

Panuncio et al. [60] proposed a neural PID control that solves large integral and derivative gains in the classical PID, PD control for the anti-swing of the overhead crane. An LQR method was applied to the anti-sway controller design of the overhead crane [61]. To make the designed linear optimal system, a parametric formula method of solutions to LQ inverse problems to get the weighting matrix Q. In [62], a Linear Quadratic Gaussian (LQG) control has been developed for controlling the load swing. In the proposed method, the quadratic derivative of a state variable term was combined with the regular standard performance indexes of the LQG for estimation and control. The extra term provides an additional weighting function to be considered to reduce the given performance indexes. The results showed that the load swinging could be decreased. In [63, 64], the Generalized Predictive Control (GPC) and LQG were well applied and compared for controlling the pendulation of the payload for an offshore crane. Roman et al. [65] proposed fusion control technics of a second-order data-driven Active Disturbance Rejection Control (ADRC) with Proportional-Derivative Takagi-Sugeno Fuzzy Control. The control structure is referred to as ADRC–PDTSFC. The data-driven controller was validated through real-time experiments. The structure improves ADRC–PDTSFC the overall control system performance by compensating for shortcomings of ADRC. An anti-swing crane control method is developed in [66]. The authors built a novel manifold and the corresponding analysis, ensuring that the system state variables would converge to the equilibrium point asymptotically when they were on this manifold.

Shengzeng et al. [67] developed a partially saturated nonlinear controller for under-actuated overhead cranes which is based on passivity. In the proposed approach, a storage function is designed with the desired inertia matrix and potential energy function, which results in a nonlinear coupling controller, thus introducing additional damping terms to the sway angle.

In order to acquire the most realistic model of the crane system, several researchers have included other parameters in their models like damping and elasticity of the structure [68], air resistance, and variable length of cable that hold the payload [69]. It has been known that time delay may produce significant damping of the oscillations [70, 71]. A control technique was founded on time-delayed position feedback of the payload cable angles where the vibration of the payload is reduced by obliging the trolley to follow inertial reference coordinates. The objective of this control technique is to add damping to the system.

The majority of research works are based on the contact detection of the payload sway displacement [72, 73]. The sway angle or the pendulum displacement is measured with different sensors like Hall-effect switches [74], inductive sensors, etc. Machine vision systems are being developed very fast in the last two decades [75]. Researchers start working on the contactless detection of the payload sway performed using machine vision applications [76- 79]. In our system, the vision system is based on a web-camera and a personal laptop as hardware, and for software, a Visual Studio C++ with OpenCV library as a programming tool for image processing.

A delayed reference non-collocated control approach for container cranes was developed by Sano et al. By taking into consideration the delay due to the vision sensor, which usually has a negative impact on the controlling, two novel swing angle observer-based control methods is developed in [80]. The investigated model in this mentioned research is only two degrees of freedom, and the suspending element is a rigid bar. Also, the parameters P and I of these controllers were not given. While in our system, the model has a multi-degree of freedom due to the suspending element, which is a chain.

1.2 Aim of research

The main aim of this research work is to design and implement a robust, fast, and practical controller for overhead cranes. The crane controllers are made to transfer the load from the starting position to the target position without consuming too much time, without overshooting, and to make the swinging of the payload as small as possible during the transfer and to make them completely disappears when the payload reaches its target position.

To target a new two-level hierarchical approach. At the first level, a simple mathematical pendulum model will be analyzed considering the time delay due to the use of a vision system. In the second level, a chain model is developed, extending the previous pendulum model considering the vibration of the suspending chain. The gain parameters associated with the payload are used from the first level model, and the rest of the parameters related to the state variables of the chain are determined by the pole placement method. The unmeasured state-variables will be determined by a collocated observer.

Another aim is to validate the successful use of the web-camera as capturing sensor, Visual Studio C++ with OpenCV library as a programming tool for image processing, and industrial Programmable Logic Controller as a programming tool for crane control on a laboratory overhead crane.

1.3 Dissertation guide

This work is organized as follows:

Chapter 1 is an introduction to crane systems with a literature review of crane automation followed by motivation and objectives.

Chapter 2, a nonlinear and linear dynamical model of an overhead crane with chain, is formulated, analyzed, and compared. In addition to the payload, the inertia and independent DoF of the suspending chain is considered.

Chapter 3 introducing a new two-level hierarchical approach. At the first level, a simple mathematical pendulum model is investigated considering the time delay due to the use of a vision system. D-subdivision method [81] is applied to determine the stability regions expressed by gain parameters for different time delays. In order to design the controller, the gain parameters associated with the state-space variables of the payload are selected within the stability region. In the second level, a chain model is developed, extending the previous pendulum model considering the vibration of the suspending chain. Only the relative displacement of the payload is measure with a vision sensor, The gain parameters associated with the payload are used from the first level model, and the rest of the parameters related to the state variables of the chain are determined by the pole placement method. The unmeasured state-variables will be determined by a collocated observer.. The objective of this approach is to reduce the load swinging as fast as possible.

Chapter 4 machine vision programs are developed to measure the displacement and swinging angle of the payload using a web-camera as capturing sensor and Visual Studio C++ and OpenCV

library as a programming tool for image processing. The layout of the experimental setup is described.

Chapter 5, an overhead crane model is used to test the proposed control technique. The results of the controlling system for different parameters are analyzed and discussed.

Chapter 6 summary

2. LINEAR AND NONLINEAR DYNAMICAL ANALYSIS OF A CRANE MODEL

This chapter deals with the dynamical analysis of a crane model. A nonlinear and linear dynamical model of an overhead crane with chain is formulated, analyzed, and compared. In addition to the payload, the inertia of the independent DoF of the suspending chain is considered. The finite element method is applied for the formulation of the nonlinear and linear models. The flexible suspending chain has no bending stiffness. The nonlinear model can be derived with the help of truss elements, which can transfer only axial force [82] and perform 2D motions. The linear model practically is a taut string [83], which is also discretized with linear two node elements approximating only lateral motions. The solutions of the linear models are compared to the nonlinear counterpart.

2.1 Equations of motions of the crane models

The picture of a crane model is shown in Fig. 2.1. It is assumed that the displacements of the two parallel chains are the same; therefore, a single chain with double mass is a good substitution in a dynamical analysis. The chain is regarded as an elastic, one-dimensional structure and can transmit only axial force along its x_1 coordinate. The motion of the structure in a plane (x, y) is investigated with nonlinear and linear approaches.

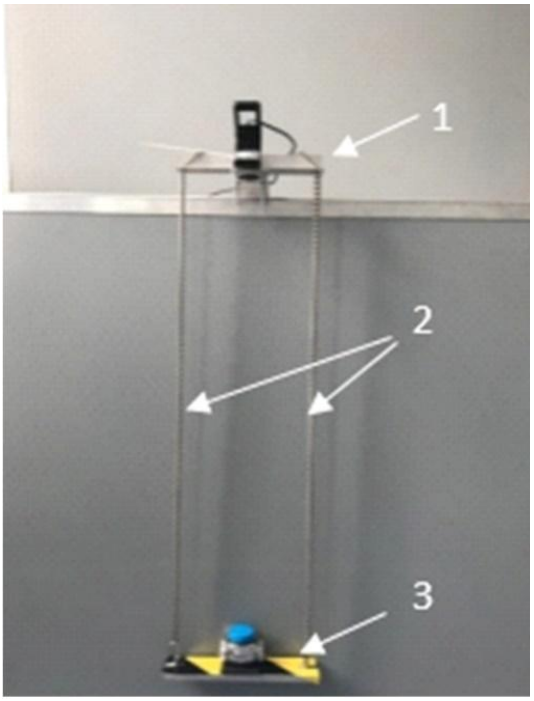

Fig. 2. 1 Part of overhead lab crane model: 1. Trolley; 2. Chains; 3. Payload.

2.1.1 Nonlinear Crane Model

In order to treat the nonlinear model of the overhead crane, an incremental form of the principle of the virtual displacements with updated Lagrangian formulation [82] is used:

$$\int_{V_t} \rho^t \ddot{\boldsymbol{u}}^{t+\Delta t} \delta \boldsymbol{u}\, dV_t + M\, \ddot{\boldsymbol{u}}_M^{t+\Delta t} \delta \boldsymbol{u}_M + \int_{V_t} E\, e_{11}^t\, \delta e_{11}\, dV_t + \int_{V_t} \sigma_{11}^t\, \delta \eta_{11}\, dV_t = \int_{V_t} \rho^t \boldsymbol{g}\, \delta \boldsymbol{u}\, dV_t - M g \delta v_M - \int_{V_t} \sigma_{11}^t\, \delta e_{11}\, dV_t, \tag{2.1}$$

where ρ^t is the density of the material in the deformed configuration; $\ddot{\boldsymbol{u}}^{t+\Delta t}$ is a 2D vector of the accelerations at time $t+\Delta t$; $\delta \boldsymbol{u}$ is the 2D vector of the virtual displacements; V_t is the volume of the structure at time t; M is the mass of the payload; $\delta \boldsymbol{u}_M$ is the 2D vector of the virtual displacements of mass M; $\boldsymbol{g}$ is the 2D vector of the gravity acceleration; E is the Young modulus; $e_{11}^t = d\bar{u}/dx_1$ is the linear strain measured in the axial direction in deformed configuration at time t; δe_{11} is the variation of the linear strain at the configuration time t; σ_{11}^t is the Cauchy stress of the one-dimensional structure at time t; $\delta \eta_{11}$ is the variation of the quadratic portion of the nonlinear strain at the configuration time t. The quadratic portion of the nonlinear strain increment is written as:

$$\delta\eta_{11} = \frac{1}{2}\delta\left[\left(\frac{\partial\bar{u}}{\partial x_1}\right)^2 + \left(\frac{\partial\bar{v}}{\partial x_1}\right)^2\right], \tag{2.2}$$

where $\bar{u}$ and $\bar{v}$ are the increments in displacements in axial x_1 and its perpendicular direction, respectively. It is noted that (2.2) can also be expressed with global Cartesian displacement increments *u*, *v*

$$\delta\eta_{11} = \frac{1}{2}\delta\left[\left(\frac{\partial u}{\partial x_1}\right)^2 + \left(\frac{\partial v}{\partial x_1}\right)^2\right]. \tag{2.3}$$

Truss elements are used to discretize (2.1). A truss element is a two-node straight line member capable of transferring axial force only. The deformed length of the element is L^t, coordinate ξ measures the distance along with the truss element, positive from node *i* to node *j*. The approximations for the axial displacement $\bar{u}^e(\xi)$ and its perpendicular displacement $\bar{v}^e(\xi)$ along the element *e* are given as:

$$\bar{u}^e(\xi) = \left(1 - \frac{\xi}{L^t}\right)\bar{u}_i + \frac{\xi}{L^t}\bar{u}_j, \tag{2.4}$$

$$\bar{v}^e(\xi) = \left(1 - \frac{\xi}{L^t}\right)\bar{v}_i + \frac{\xi}{L^t}\bar{v}_j. \tag{2.5}$$

where $\bar{u}_i, \bar{u}_j$ and $\bar{v}_i, \bar{v}_j$ are the increments in nodal displacements at element local coordinate system see Fig. 2.2, it is noted that $dx_1 = d\xi$.

The local nodal displacement increments $\bar{u}_i, \bar{u}_j$ and $\bar{v}_i, \bar{v}_j$ can be related to the global Cartesian node displacement increments u_i, u_j and v_i, v_j see Fig. 2.3, via the following transformation:

$$\begin{bmatrix} \bar{u}_i \\ \bar{u}_j \\ \bar{v}_i \\ \bar{v}_j \end{bmatrix} = \begin{bmatrix} \frac{x_j - x_i}{L^t} & \frac{y_j - y_i}{L^t} & 0 & 0 \\ -\frac{y_j - y_i}{L^t} & \frac{x_j - x_i}{L^t} & 0 & 0 \\ 0 & 0 & \frac{x_j - x_i}{L^t} & \frac{y_j - y_i}{L^t} \\ 0 & 0 & -\frac{y_j - y_i}{L^t} & \frac{x_j - x_i}{L^t} \end{bmatrix} \begin{bmatrix} u_i \\ u_j \\ v_i \\ v_i \end{bmatrix}. \tag{2.6}$$

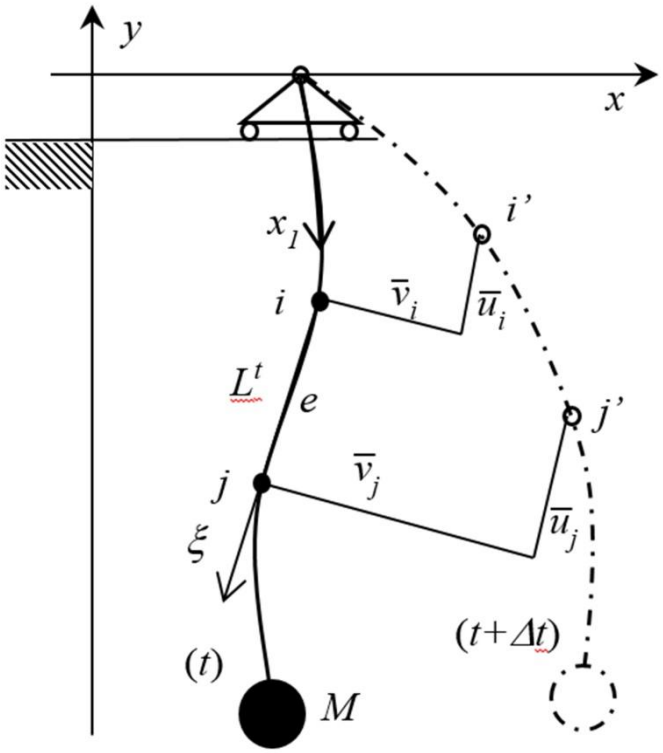

Fig. 2. 2 Increments in nodal displacements of the nonlinear model given by local coordinates.

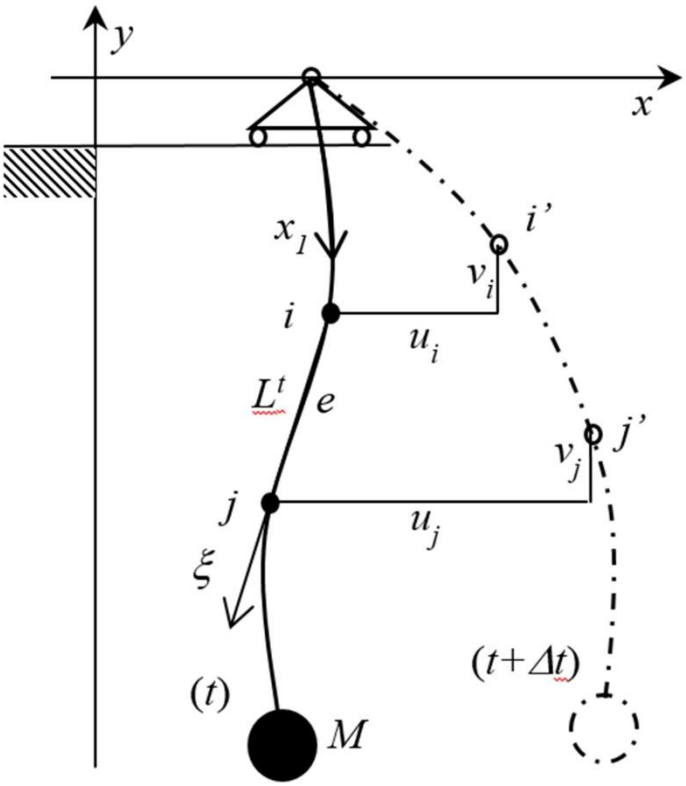

Fig. 2. 3 Increments in nodal displacements of the nonlinear model given by global coordinates.

It is also noted that the relation between the nodal forces in local and global coordinates are expressed similar way as (2.6), only the nodal displacements are replaced with nodal forces. The same transformation is also valid for the nodal accelerations.

The global Cartesian displacements in (2.3) and accelerations in (2.1) are approximated by the same linear interpolation functions as in (2.4) and (2.5):

$$u^e(\xi) = \left(1 - \frac{\xi}{L^t}\right) u_i + \frac{\xi}{L^t} u_j, \tag{2.7}$$

$$v^e(\xi) = \left(1 - \frac{\xi}{L^t}\right) v_i + \frac{\xi}{L^t} v_j, \tag{2.8}$$

$$\ddot{u}^e(\xi) = \left(1 - \frac{\xi}{L^t}\right) \ddot{u}_i + \frac{\xi}{L^t} \ddot{u}_j, \tag{2.9}$$

$$\ddot{v}^e(\xi) = \left(1 - \frac{\xi}{L^t}\right) \ddot{v}_i + \frac{\xi}{L^t} \ddot{v}_j, \tag{2.10}$$

where $\ddot{u}_i, \ddot{u}_j, \ddot{v}_i, \ddot{v}_j$ are nodal accelerations

The coordinates of the vector of displacement variation are $\delta \boldsymbol{u}$ approximated in the same way as the functions of (2.4), (2.5) and (2.7), (2.8).

Using (2.2)-(2.10) to determine the integrals in (2.1), it provides element mass matrix:

$$\bar{\mathbf{M}}_e = \frac{\rho^t A^t L^t}{6} \begin{bmatrix} 2 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \\ 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 2 \end{bmatrix}, \tag{2.11}$$

Linear stiffness matrix $\bar{\mathbf{K}}_{Le}$:

$$\bar{\mathbf{K}}_{Le} = \frac{A^t E}{L^t} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \tag{2.12}$$

Geometric element stiffness matrices $\bar{\mathbf{K}}_{Ge}$:

$$\bar{\mathbf{K}}_{Ge} = \frac{A^t \sigma_{11}}{L^t} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}. \tag{2.13}$$

The external and internal load vectors $\bar{\mathbf{f}}_{Ge}$, $\bar{\mathbf{f}}_{\boldsymbol{\sigma}e}$ in element local coordinate system:

$$\bar{\mathbf{f}}_{Ge} = -\frac{g\boldsymbol{\rho}^t A^t L^t}{2}\begin{bmatrix} \dfrac{y_j - y_i}{L^t} \\ \dfrac{x_j - x_i}{L^t} \\ \dfrac{y_j - y_i}{L^t} \\ \dfrac{x_j - x_i}{L^t} \end{bmatrix}, \tag{2.14}$$

$$\bar{\mathbf{f}}_{\boldsymbol{\sigma}e} = A^t \boldsymbol{\sigma}_{11}^t \begin{bmatrix} -1 \\ 0 \\ 1 \\ 0 \end{bmatrix}. \tag{2.15}$$

where $A^t\boldsymbol{\sigma}_{11}^t = N^t$ is the axial force of the truss element.

The element matrices are transformed to the global Cartesian coordinate system with the help of (2.6) then assembled according to the usual FEM process. The obtained equation of motion associated to (1) is written in matrix form as follows:

$$\mathbf{M}^{t+\Delta t}\ddot{\mathbf{q}} + \left(\mathbf{K}_L^t + \mathbf{K}_G^t\right)\Delta\mathbf{q} = \mathbf{f}_g^{t+\Delta t} - \mathbf{f}_{\boldsymbol{\sigma}}^t, \tag{2.16}$$

where $\mathbf{M}^{t+\Delta t}$ is the structural mass matrix; $\mathbf{K}_L^t$ and $\mathbf{K}_G^t$ are the structural linear and geometric stiffness matrices; $\mathbf{f}_g^{t+\Delta t}$ is the structural column vector of external loads; $\mathbf{f}_{\boldsymbol{\sigma}}^t$ is the structural column vector of internal loads; $\ddot{\mathbf{q}}$ is the structural column vector of nodal accelerations; $\Delta\mathbf{q}$ is the structural column vector of the increments in nodal displacements. It is noted that the mass matrix is a constant matrix, i.e., it is time-invariant $\mathbf{M} = \mathbf{M}^{t+\Delta t}$, since the mass of the structure does not change in time.

The equation of motion (2.16) can be integrated numerically, e.g., by the Newmark method. The steps of this method [82] are summarized as follows:

I. Initially, the stiffness matrices $\mathbf{K}_L^t$ $\mathbf{K}_G^t$ and the mass matrix $\mathbf{M}$ are formed and set the initial value of the displacements $\mathbf{q}^0$, the velocity $\dot{\mathbf{q}}^0$, and the accelerations $\ddot{\mathbf{q}}^0$.

II. Form of the effective stiffness matrix at time *t*

$$\hat{\mathbf{K}} = \mathbf{K}_L^t + \mathbf{K}_G^t + \frac{4}{\Delta t^2}\mathbf{M}. \tag{2.17}$$

III. Form of the effective load vector $\hat{\mathbf{f}}$ at time t+Δt

$$\hat{\mathbf{f}} = \mathbf{f}_g^{t+\Delta t} + \mathbf{M}\left(\frac{4}{\Delta t}\mathbf{q}^t + \dot{\mathbf{q}}^t\right) - \mathbf{f}_{\boldsymbol{\sigma}}^t. \tag{2.18}$$

IV. Solve the linear equation for the displacement increment $\Delta\mathbf{q}_{(0)}$

$$\hat{\mathbf{K}}\,\Delta\mathbf{q}_{(i)} = \hat{\mathbf{f}}, \qquad \text{where } i = 0. \tag{2.19}$$

V. Iteration for dynamic equilibrium

(a) $i = i + 1$

(b) Calculate $(i-1)st$ approximation to accelerations, velocities, and displacements

$$\ddot{\mathbf{q}}_{(i-1)}^{t+\Delta t} = \frac{4}{\Delta t^2}\Delta\mathbf{q}_{(i-1)} - \frac{4}{\Delta t}\dot{\mathbf{q}}^t - \ddot{\mathbf{q}}^t, \tag{2.20}$$

$$\dot{\mathbf{q}}_{(i-1)}^{t+\Delta t} = \frac{4}{\Delta t}\Delta\mathbf{q}_{(i-1)} - \dot{\mathbf{q}}^t, \tag{2.21}$$

$$\mathbf{q}_{(i-1)}^{t+\Delta t} = \Delta\mathbf{q}_{(i-1)} + \mathbf{q}^t. \tag{2.21}$$

(c) Calculate $(i-1)st$ effective out of balance load vector

$$\hat{\mathbf{f}}_{(i-1)}^{t+\Delta t} = \mathbf{f}_g^{t+\Delta t} - \mathbf{M}\ddot{\mathbf{q}}_{(i-1)}^{t+\Delta t} - \mathbf{f}_{\boldsymbol{\sigma}(i-1)}^{t+\Delta t}. \tag{2.23}$$

(d) Solve for the i'th correction to displacement increment

$$\hat{\mathbf{K}}\,\Delta\hat{\mathbf{q}}_{(i)} = \hat{\mathbf{f}}_{(i-1)}^{t+\Delta t}. \tag{2.24}$$

(e) Calculate new displacement increments

$$\Delta\mathbf{q}_{(i)} = \Delta\mathbf{q}_{(i-1)} + \Delta\hat{\mathbf{q}}_{(i)}. \tag{2.25}$$

(f) Iteration is terminated if $\dfrac{\left\|\Delta\hat{\mathbf{q}}_{(i)}\right\|_2}{\left\|\mathbf{q}_{(i)}^{t+\Delta t}\right\|_2} \le tol$ otherwise back to step (a). The value of tol is prescribed by the user. Its magnitude is 10^{-6}

VI. If convergence $\Delta\mathbf{q} = \Delta\mathbf{q}_{(i)}$, then accelerations, velocities, and displacements are determined as:

$$\ddot{\mathbf{q}}^{t+\Delta t} = \frac{4}{\Delta t^2}\Delta\mathbf{q} - \frac{4}{\Delta t}\dot{\mathbf{q}}^t - \ddot{\mathbf{q}}^t, \tag{2.26}$$

$$\dot{\mathbf{q}}^{t+\Delta t} = \mathbf{q}^t + 0.5\Delta t\left(\ddot{\mathbf{q}}^t + \ddot{\mathbf{q}}^{t+\Delta t}\right), \tag{2.27}$$

$$\mathbf{q}^{t+\Delta t} = \mathbf{q}^{t} + \Delta\mathbf{q}\,. \tag{2.28}$$

Then increase the time with Δt and jump to step II.

2.1.2 Linear crane model

Usually, the linear counterpart of the nonlinear equation of motion is needed in order to design a controller for a dynamic system. The crane model made of chain shown in Fig. 2.4 can be regarded as a taut string, which suffers relatively small lateral displacements. The axial force of the string due to gravity force is calculated in the rest position and assumed to be constant in the course of its motion.

The linear equation (1D model) of motion of the crane can be derived by the help of the virtual displacements written for a taut string

$$\int_{V_0} \rho_0 \ddot{u}^t \delta u \, dV_0 + M\ddot{u}_M^t \, \delta u + \int_{V_0} \sigma_{11}^0 \delta\left(\frac{1}{2}u_t^2\right) dV_0 = R^t\,, \tag{2.29}$$

where ρ_0 is the density of the material at time zero; $\ddot{u}^t$ is the acceleration function in lateral direction; i.e. in horizontal direction at time t; δu is the virtual lateral displacement function; V_0 is the volume of the structure at time t=0; M is the mass of the payload; $\ddot{u}_M^t$ and δu_M is the acceleration and virtual lateral displacements of mass M; σ_{11}^0 is the function of the axial stress due to gravity force determined in the rest position of the crane; and R^t is the virtual work of the external force obtained from displacement control.

One dimensional linear line elements are used to discretize (2.29). The horizontal displacement is approximated along an element in local coordinate system:

$$u^e(\xi) = \left(1 - \frac{\xi}{L^0}\right)u_i + \frac{\xi}{L^0}u_j\,, \tag{2.30}$$

where u_i, u_j are the horizontal nodal displacements of the element e.

The horizontal acceleration along an element is also approximated with the same interpolation functions

$$\ddot{u}^e(\xi) = \left(1 - \frac{\xi}{L^0}\right)\ddot{u}_i + \frac{\xi}{L^0}\ddot{u}_j\,, \tag{2.31}$$

where $\ddot{u}_i, \ddot{u}_j$ are horizontal nodal accelerations of the element e.

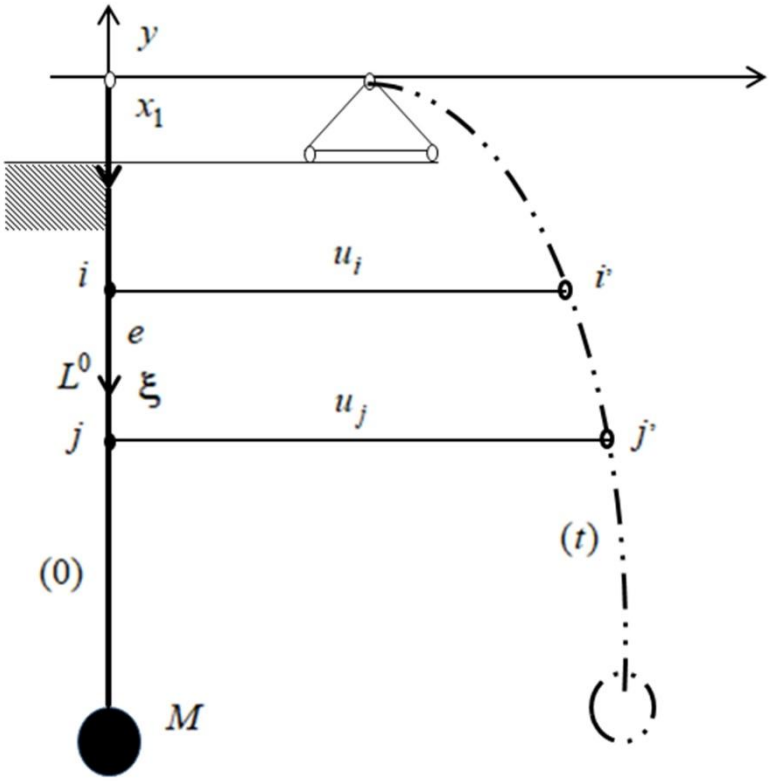

Fig. 2. 4 Nodal displacements of the linear model.

Substituting (2.30) and (2.31) into (2.29) the element mass matrix $\mathbf{M}^e$ and stiffness matrix $\mathbf{K}^e$ can be derived

$$\mathbf{M}^e = \frac{\rho^0 A^0 L^0}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}, \tag{2.32}$$

$$\mathbf{K}^e = A^0 \sigma_{11}\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \tag{2.33}$$

$$\mathbf{q}^e = \begin{bmatrix} u_i \\ u_j \end{bmatrix},\ \ddot{\mathbf{q}}^e = \begin{bmatrix} \ddot{u}_i \\ \ddot{u}_j \end{bmatrix}, \tag{2.34}$$

where $\mathbf{q}^e$ is the element nodal displacement vector and $\ddot{\mathbf{q}}^e$ is the element nodal acceleration vector, which is generated directly by the global Cartesian horizontal displacement approximation; therefore, no transformation is needed before assembling the structural matrices.

The linear equation of motion of the crane model is written as:

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{f}^t, \tag{2.35}$$

where **M** and **K** are the structural mass and stiffness matrices, $\mathbf{f}^t$ is the column vector of the external forces due to displacement control, **q** is the structural vector of nodal displacements, $\ddot{\mathbf{q}}$ is the structural vector of nodal accelerations.

The linear differential equation (2.35) can also be given in state-space form introducing new variables $\mathbf{q} = \mathbf{x}_1$ and $\dot{\mathbf{q}} = \mathbf{x}_2$:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u\,, \tag{2.36}$$

where $\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix}$, $\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & \mathbf{0} \end{bmatrix}$, $\mathbf{b}u = \begin{bmatrix} 0 \\ -\mathbf{M}^{-1}\mathbf{f}^t \end{bmatrix}$, and **I** is a unit matrix.

Numerical integration of (2.36) can be performed with a number of standard methods, e.g., Runge-Kutta method or trapezoid rule, using Matlab or Scilab software system.

2.2 Numerical examples

Nonlinear and linear FEM programs have been developed using the above theory. The length of the chains is 0.8 m, and it is subdivided into ten uniform finite elements. The linear version will also be analyzed with an FE mesh of two elements chain models. The mass of the chain is 0.22 kg, and two different payloads are exerted on the crane. The heavy one is 0.42 kg, the light one is 0.07 kg.

In order to compare their performances, a simple crane motion will be simulated. In the beginning, the trolley is moving with constant velocity $v_1 = 0.8\,\mathrm{m/s}$ along the length $x_1 = 0.8\,\mathrm{m}$, and then it stops suddenly. After that, the payload and the suspending chain will swing freely.

Motions of the cranes obtained for heavy payload using linear and nonlinear models are displayed in Fig. 2.5 and Fig. 2.6. It is clearly seen that the payload performs only horizontal motions in Fig. 2.6. However, the deformed shapes of the chains for both models are comparable in both Figures.

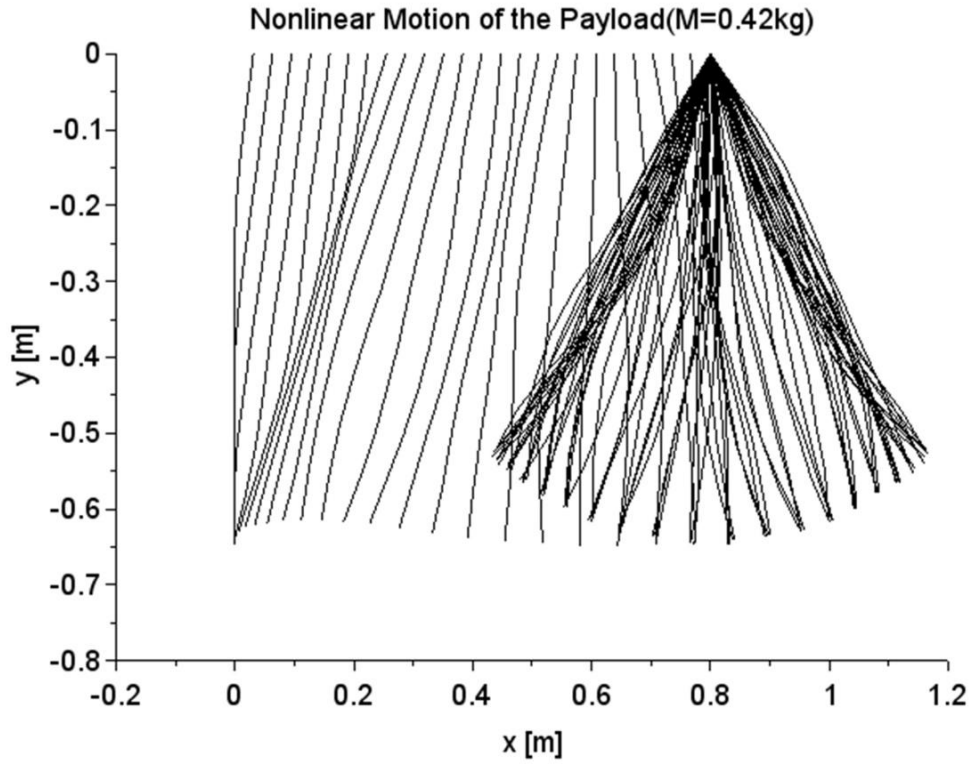

Fig. 2. 5 Position and shape of the chain during the motion for a nonlinear model.

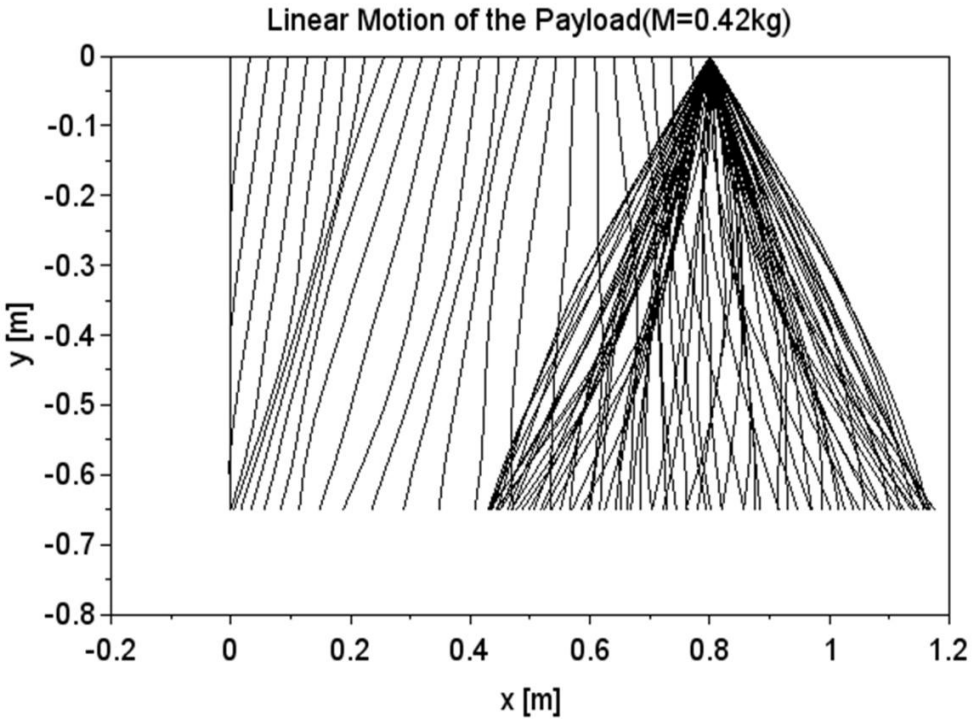

Fig. 2. 6 Position and shape of the chain during the motion of a linear model.

The computations of the next experiment have been performed for heavy and light payloads. The results are given in reference coordinate system and in relative coordinate system attached to the trolley, which is called in the sequel absolute and relative motions, respectively. The absolute and relative motions of the payloads and the relative motions of the middle of the chain are shown in Fig. 2.7 - Fig. 2.12.

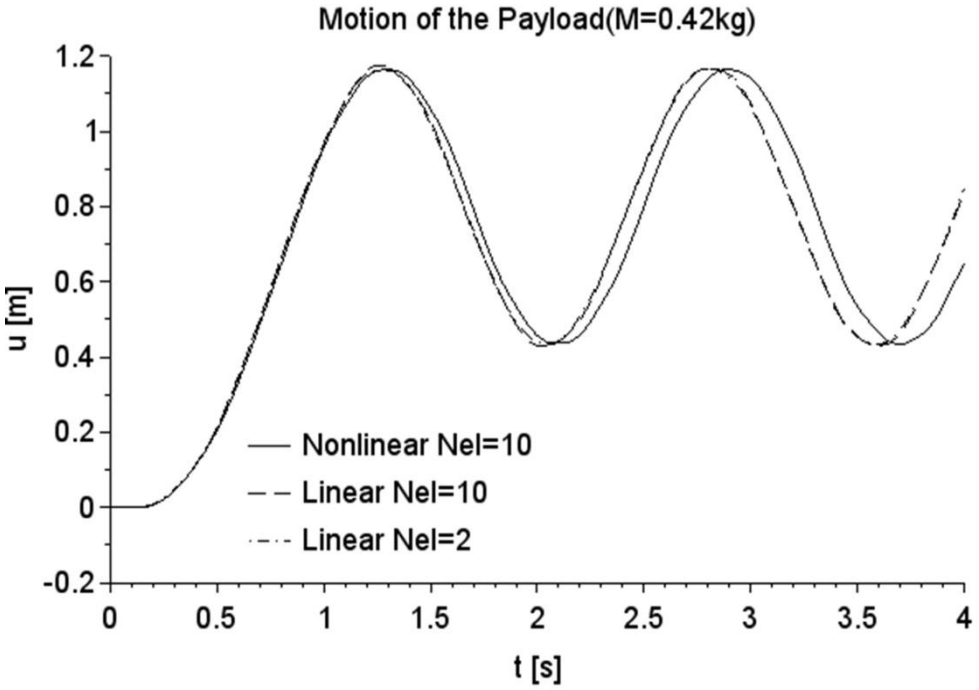

Fig. 2. 7 Absolute motion of heavy payload.

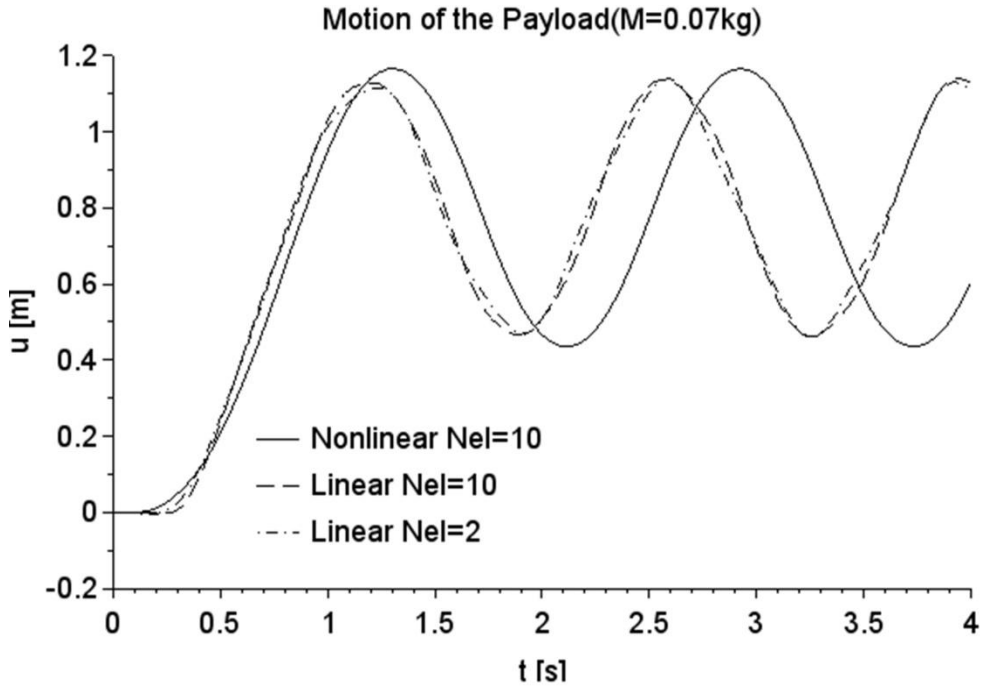

Fig. 2. 8 Absolute motion of light payload.

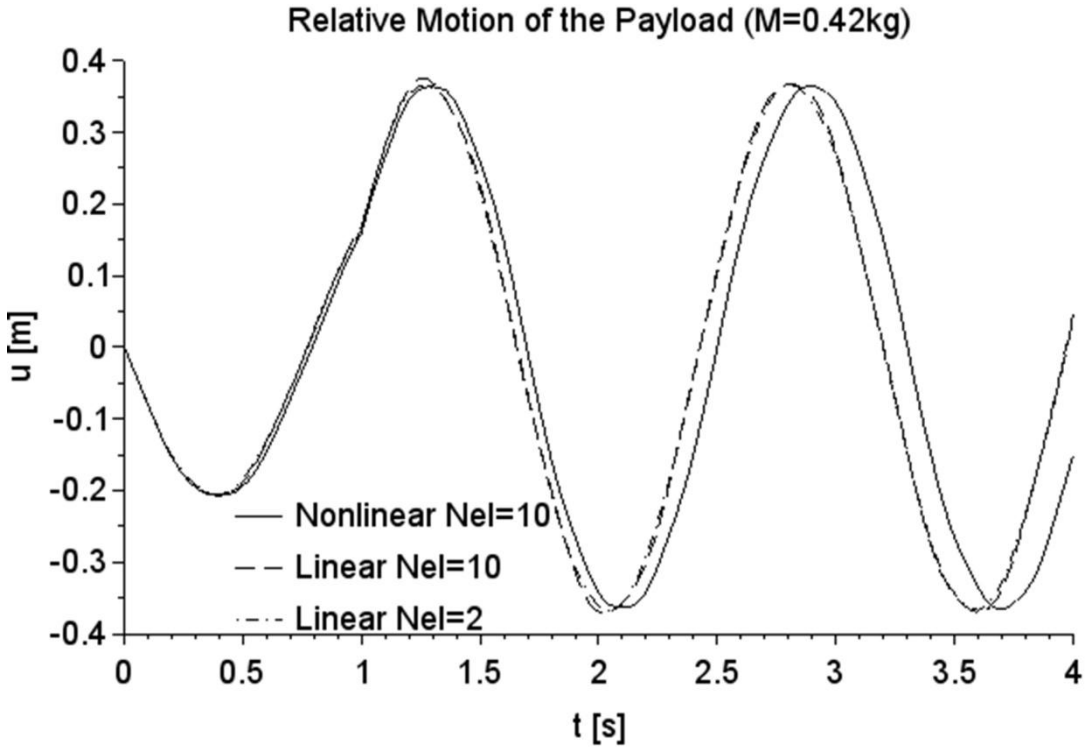

Fig. 2. 9 Relative motion of heavy payload.

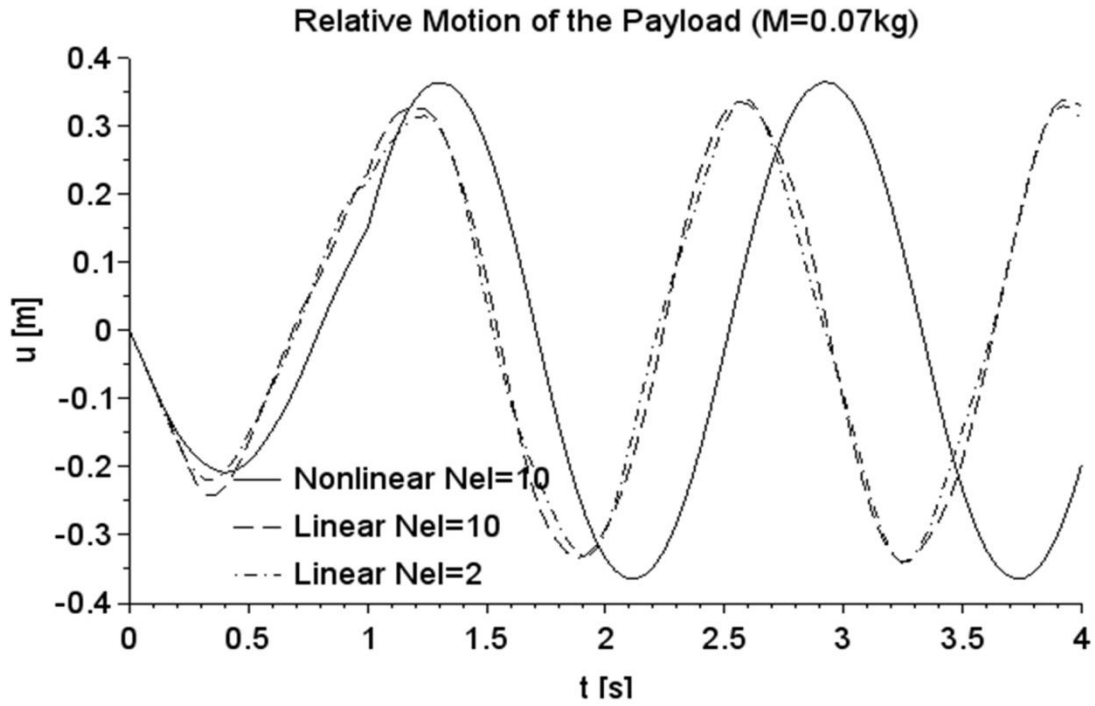

Fig. 2. 10 Relative motion of the light payload.

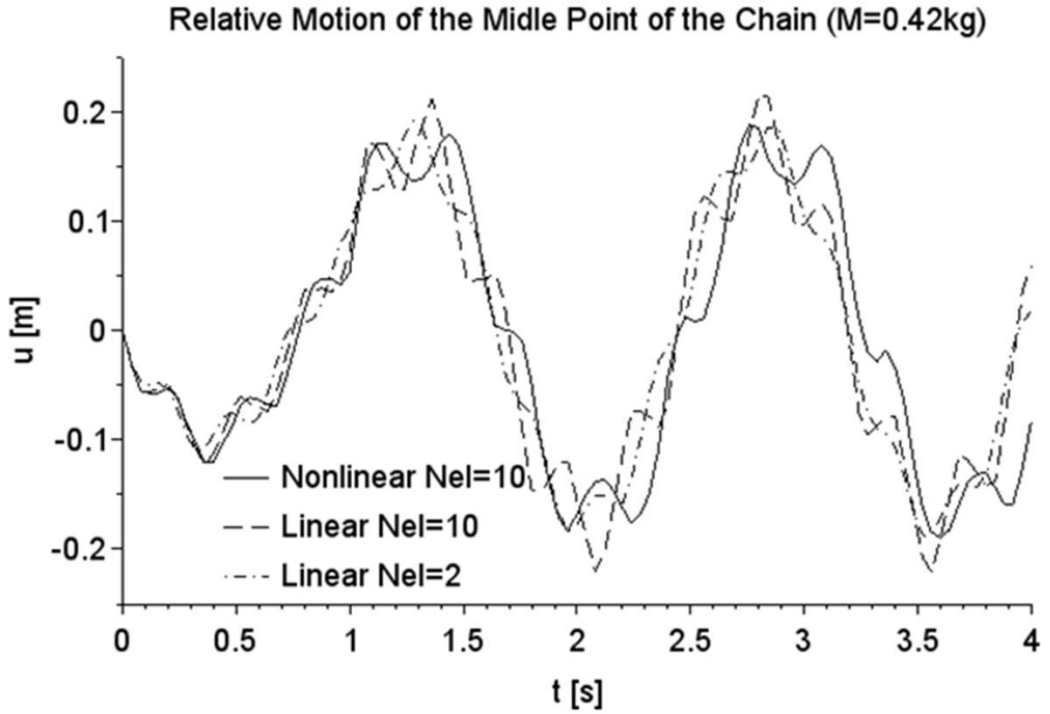

Fig. 2. 11 Relative motions of the middle point of the chain for heavy payload model.

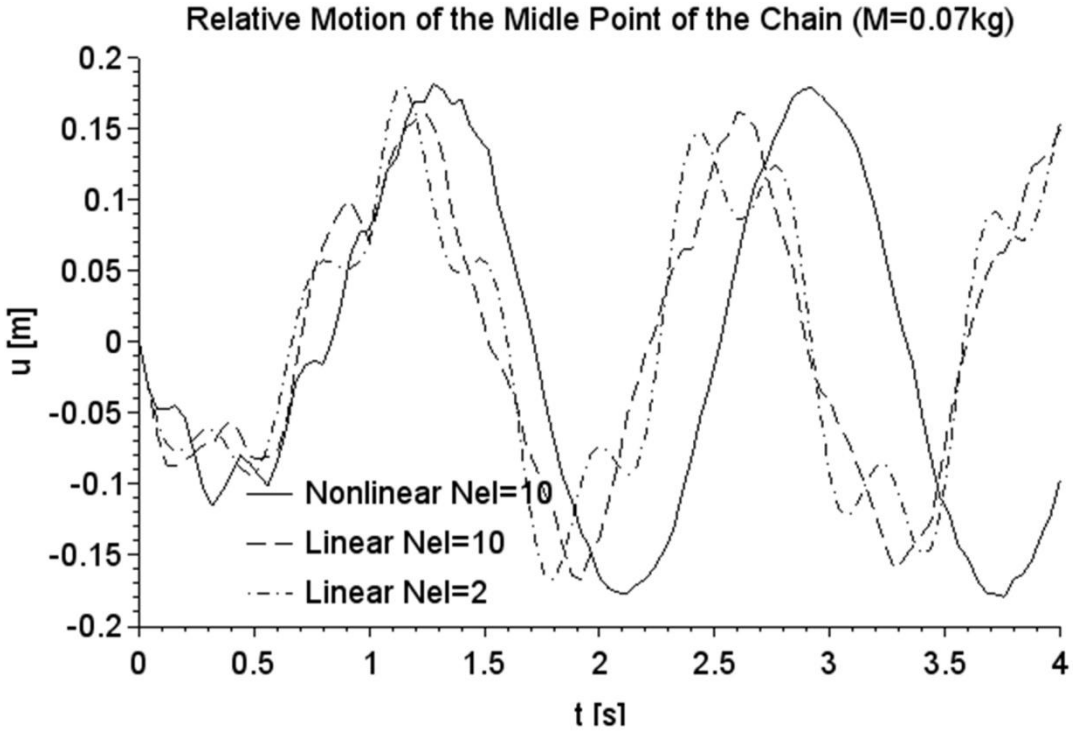

Fig. 2. 12 Relative motions of the middle point of the chain for light payload model.

The displacements of nonlinear and linear models for heavy payload show good agreement in Fig. 2.7 and Fig. 2.9, while the discrepancies are more significant for the light one in Fig. 2.8 and Fig. 2.10.

It is clear from Fig. 2.11 and Fig. 2.12 that the chain vibration is significant. The discrepancies between the nonlinear and linear models are negligible for the heavy payload, in the beginning, see Fig. 2.11, and then only a relatively small shift can be detected later on. Except, in the beginning, the results of chain vibrations are less similar for light payload see Fig. 2.12. However, during the controlling of the crane, the displacement of the payload is enforced to be close to the real motion of the crane. Therefore, a small shift in the solution is not a significant problem. The linear model with two finite elements also provided acceptable results compared to ten element mesh; therefore, it is an excellent candidate to use as an observer.

3. CONTROLLING OF TROLLEY POSITION AND PAYLOAD SWINGING OF AN OVERHEAD CRANE

3.1 Mathematical pendulum model of the overhead crane

The model of the first level of the proposed hierarchical controller is described using a simple mathematical pendulum.

The motion of the crane payload in Fig. 3.1 can be written as:

$$x_{\mathrm{M}} = u + x_r \tag{3.1}$$

$$x_M = u + L\sin\theta \tag{3.2}$$

$$y_M = -L\cos\theta \tag{3.3}$$

where u is the prescribed position of the trolley, L, Θ are the length and the angle of the pendulum, respectively, $x_{M,}$ x_r are the horizontal absolute and relative positions of the payload M, respectively.

The Lagrangian approach is used to derive the mathematical model. The kinetic energy T and the potential energy V of the system:

$$T = \frac{1}{2} M v^2 \tag{3.4}$$

$$V = -M g L\cos\theta \tag{3.5}$$

where g is the gravity acceleration, and the velocity v of the load is calculated as:

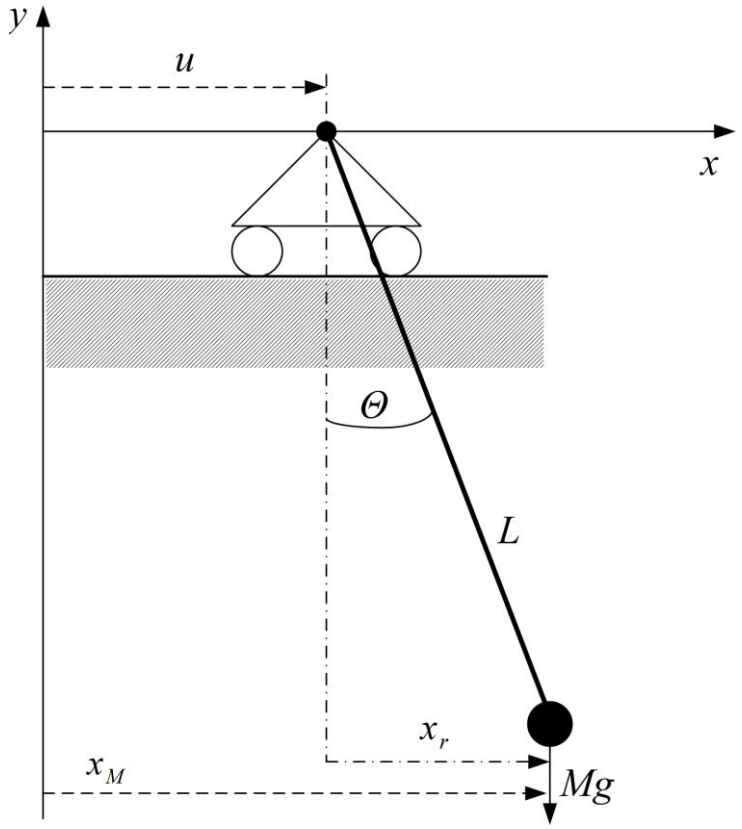

Fig.3. 1 Overhead-crane model (L=0.8 m, M=0.419 kg).

$$v^2 = \dot{x}_M^2 + \dot{y}_M^2 \tag{3.6}$$

The Lagrangian function *Lg* is written as:

$$L_g = \frac{1}{2} M \left(L^2 \dot{\theta}^2 + 2\dot{u} L \dot{\theta} \cos\theta + \dot{u}^2 \right) + M g L \cos\theta \tag{3.7}$$

then it is inserted into the Euler-Lagrange's equation of the second kind:

$$\frac{d}{dt}\left(\frac{\partial L_g}{\partial \dot{\theta}} \right) - \frac{\partial L_g}{\partial \theta} = 0 \tag{3.8}$$

which gives a nonlinear equation of motion for the overhead crane model:

$$ML^2\ddot{\theta} + M\ddot{u}L\cos\theta + MgL\sin\theta = 0 \tag{3.9}$$

The equation (3.9) can be simplified assuming $\sin\Theta \approx \Theta$; $\cos\Theta \approx 1$:

$$\ddot{\theta} + \frac{g}{L}\theta = -\frac{\ddot{u}}{L} \tag{3.10}$$

Since the position of the payload is controlled by the motion u of trolley, the following approximation is substituted in equation (3.10):

$$\theta \cong \sin\theta = \frac{x_r}{L} = \frac{x_M - u}{L} \tag{3.11}$$

The resulting equation of motion is expressed by the position x of the payload

$$\ddot{x}_M + \frac{g}{L} x_M = \frac{g}{L} u \tag{3.12}$$

The main objective is to determine *u* the position of the trolley in equation (3.12) as a function of time to decrease the oscillation of the pendulum considering the delay of the system.

In order to design anti-swing control of the payload, the state variables must be measured. The experimental setup contains a machine vision system that is mounted on the trolley, and it determines the relative displacement of the payload. The measured data is sent to a PLC via Arduino Nano and D/A converter with a sample rate. The position of the trolley is updated in each time step Δt from the initial to a target position x_T . It means that the state feedback has a delay τ, which must be considered.

The control signal u is defined by the state feedback:

$$u(t) = -k_1\left(x_M(t-\tau) - x_T\right) - k_2\dot{x}_r(t-\tau) + x_T \tag{3.13}$$

Substituting equation (3.13) into equation (3.12) and taking into consideration equation (3.1), the equation of motion with delay is obtained:

$$\ddot{x}_M(t) + \frac{g}{L} x_M(t) + \frac{g}{L} k_1 x_M(t-\tau) + \frac{g}{L} k_2 \dot{x}_M(t-\tau) = \frac{g}{L}\left[k_2\dot{u}(t) + \left(k_1+1\right)x_T\right]. \tag{3.14}$$

Characteristic of equation (3.14) is written in Laplacian domain:

$$s^2 + \frac{g}{L} + \frac{g}{L} k_1 e^{-s\tau} + \frac{g}{L} k_2 s e^{-s\tau} = 0 \tag{3.15}$$

In order to use D-subdivision method, we substitute s=(iω) into equation (3.15), after straightforward manipulations, the gain parameters can be separated and expressed as a functions of ω:

$$k_1 = \left(\frac{L}{g}\omega^2 - 1\right)\cos(\omega\tau) \tag{3.16}$$

$$k_2 = \left(\frac{L}{g}\omega^2 - 1\right)\sin(\omega\tau) \tag{3.17}$$

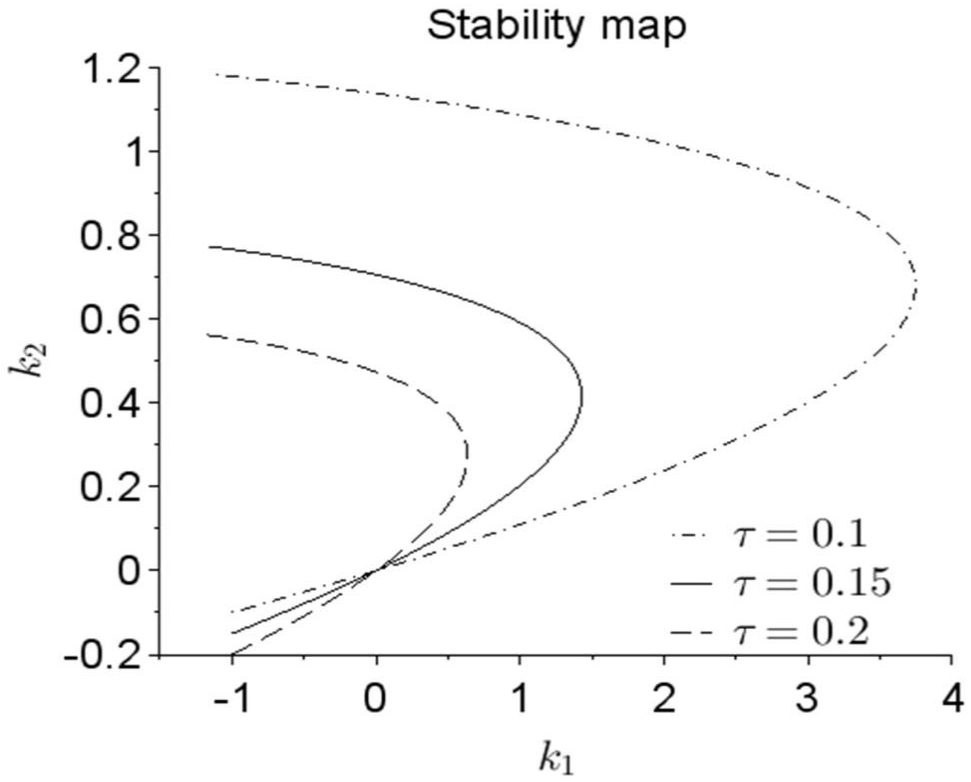

Fig.3. 2 Stability regions for different delays.

The gain parameters in equations (3.16) and (3.17) determine stability regions of the system for different time delays τ (0.1 s; 0.15 s; 0.2 s), as shown in Fig. 3.2. The higher the time delay, the smaller the stability region.

Equation (3.12) can be written also in state space form with state variables $x_1=x$ and $x_2=\dot{x}$

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{g}{L} & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{g}{L} \end{bmatrix} u \tag{3.18}$$

where the gravity acceleration g=9.81 m.s^2.

It can be seen from Fig. 3.2 that the curves of the stability regions are intersecting each other at points $k_1=0$, $k_2=0$. The size of the stability region is shrinking as the time delay is increasing. Therefore, choosing the parameters approximately at the vicinity $-2<K_1<0$ and $K_2=0$ will make the system stable and robust since it remains in a stable state with a high range of time delay. The gains of a non-delayed linear system can be determined, e.g., using the pole placement method with MATLAB or Scilab software. According to the control theory, if the poles of the transfer function are located on the left half-space of the complex domain, the linear system is stable. Prescribing the poles as $p_{1,2}$ =-0.12±i2.5, it provides gain parameters k_1=-0.5, k_2=0.02, which are within the stable region of Fig. 3.2 for different time delays.

This way the stability of the system is granted. Equation (3.18) can be solved with its discrete-time state-space equation with time step Δt, which is proportional with the delay of feedback:

$$\boldsymbol{x}[n+1] = \boldsymbol{A}_D x[n] + \boldsymbol{b}_D u[n]; \quad \boldsymbol{y}[n] = \boldsymbol{c}^T \boldsymbol{x}[n] \tag{3.19}$$

It is noted that in equation (3.19) the state variables of the feedback are given in time step n-1 in order to consider the delay $\tau = \Delta t$.

$$u[n] = -\left[k_1\left(x_M[n-1] - x_T\right) + k_2 \dot{x}_r[n-1]\right] + x_T \tag{3.20}$$

where $\boldsymbol{x}[k] = \begin{bmatrix} x_1[n] \\ x_2[n] \end{bmatrix}$; $\boldsymbol{A}_D = \begin{bmatrix} 0.8651897 & 0.1431969 \\ -1.7559517 & 0.8651897 \end{bmatrix}$; $\boldsymbol{b}_D = \begin{bmatrix} 0.1348103 \\ 1.7559517 \end{bmatrix}$

are determined using Scilab software with Δt=0.15 s.

In the right-hand side of the equation (3.20) the variables are taken in time step *n-1* instead of n due to time delay τ=0.15 s. The results obtained for the control $u[n]$ and $x_r[n]$ are shown in Fig. 3.3 and Fig. 3.4. The displacement of the trolley is shown in Fig. 3.3, while the Fig. 3.4 displays the displacement of the payload relative to the trolley. The simulation shows that the proposed controller provides good active damping; the vibration of the payload is suppressed effectively within 5 seconds.

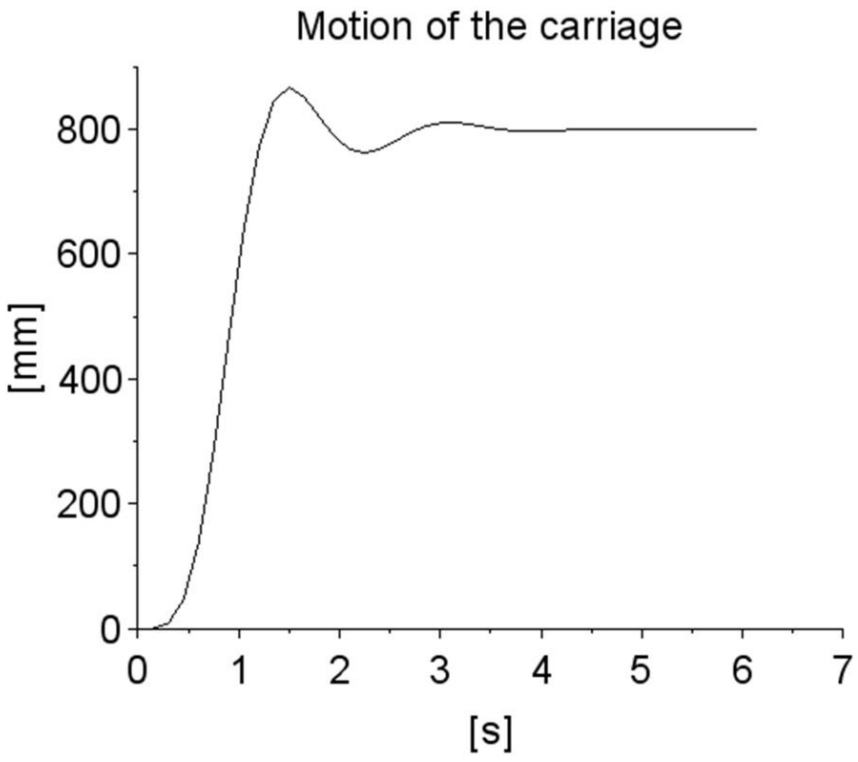

Fig.3. 3 Simulation of anti-swing control of overhead crane model motion of the carriage.

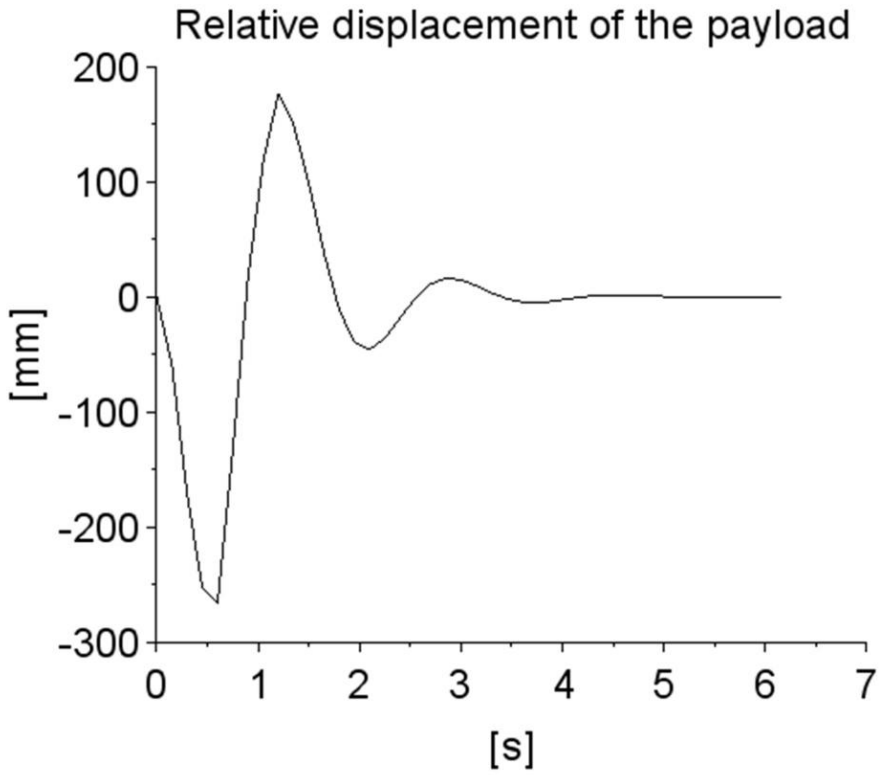

Fig.3. 4 Simulation of anti-swing control of overhead crane model displacement of the payload.

The payload will inevitably be dominant in an extended model, where the suspending chain will be considered in the next section. Therefore, the gain parameters k_1=-0.5, k_2=0.02 related to the payload state variables will be used in the extended model.

3.2 Formulation of chain model

The second level of the hierarchical approach will be described. In the extended model, the vibrations of the payload and the suspending chains are considered with a linear model of a taut string. The mass of the chain is m_L=0.22 kg. It is assumed that the displacement of the payload is negligible in the vertical direction compared to the horizontal one.

The force in the chain due to gravity is calculated as:

$$F(y) = (M + \mu L)g - y\mu g \tag{3.21}$$

where $\mu = m_L.L$ is the linear mass density of the chain, y is the vertical coordinate of the payload.

The Lagrangian of the whole structure is written as:

$$L_g = T - V = \frac{1}{2}\int_0^{-L} \mu \left[\frac{\partial x(y,t)}{\partial t}\right]^2 dy + \frac{1}{2}M\dot{x}_M^2 - \frac{1}{2}\int_0^{-L} F(y)\left[\frac{\partial x(y,t)}{\partial y}\right]^2 dy \tag{3.22}$$

The chain in Fig. 3.5 can be discretized with two-node linear line elements. The lateral displacement is approximated with the following linear interpolation:

$$x^e(\xi) = \left(1 - \frac{\xi}{L^e}\right) x_i^e + \frac{\xi}{L^e} x_j^e \tag{3.23}$$

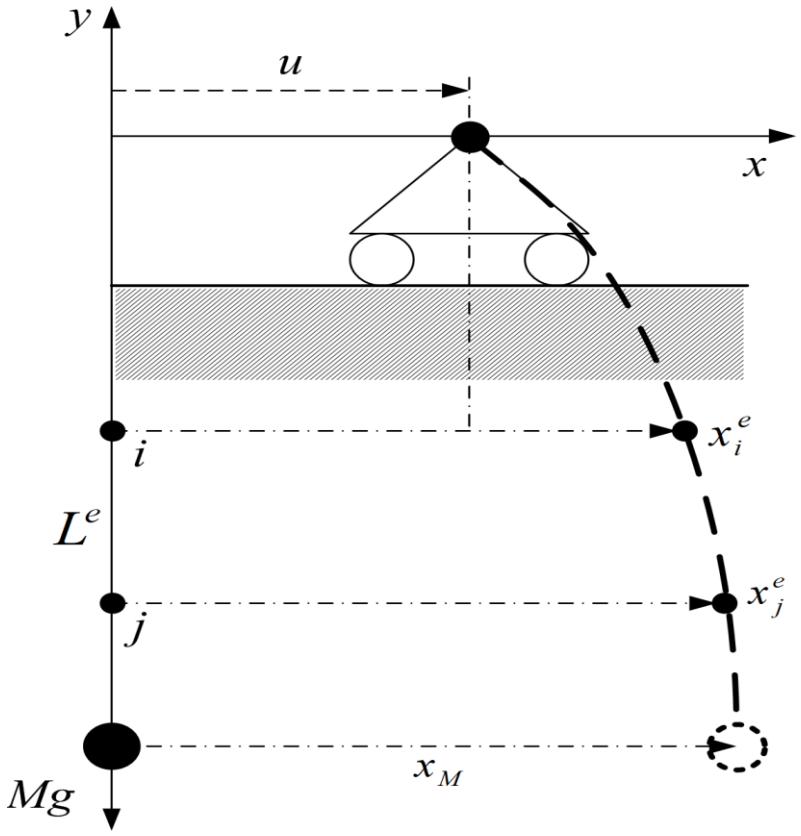

Fig.3. 5 Overhead-crane model (L=0.8 m, M=0.419 kg, m2=0.22 kg).

The chain where ξ is a local coordinate of element *e*, L^e is the length of one element, x_i^e x_j^e are its lateral nodal displacements. It is noted that the local coordinate ξ is measured in direction *y* and their elementary lengths are equal $d\xi = dy$.

Substituting equation (3.23) into equation (3.22), the Lagrangian can be expressed with finite element matrices:

$$\begin{aligned} L_g &= \frac{1}{2}\sum_{e=1}^{Ne}\int_0^{L^e} \mu\left[\frac{\partial x^e(\xi,t)}{\partial t}\right]^2 d\xi + \frac{1}{2}M\dot{x}_M^2 - \frac{1}{2}\sum_{e=1}^{Ne}\int_0^{L} F^e(\xi)\left[\frac{\partial x(\xi,t)}{\partial \xi}\right]^2 d\xi \\ &= \frac{1}{2}\sum_{e=1}^{Ne}\dot{\mathbf{q}}^{eT}\mathbf{M}^e\dot{\mathbf{q}}^e + \frac{1}{2}M\dot{x}_M^2 - \frac{1}{2}\sum_{e=1}^{Ne}\mathbf{q}^{eT}\mathbf{K}^e\mathbf{q}^e \end{aligned} \tag{3.24}$$

where $\boldsymbol{q}^e$ is the vector of element nodal displacement, $\mathbf{M}^e$ is the element mass matrix, and $\mathbf{K}^e$ is the element stiffness matrix:

$$\mathbf{q}^e = \begin{bmatrix} x_i \\ x_j \end{bmatrix}; \quad \mathbf{M}^e = \mu L^e \begin{bmatrix} \frac{1}{3} & \frac{1}{6} \\ \frac{1}{6} & \frac{1}{3} \end{bmatrix}; \quad \mathbf{K}^e = \frac{F^e}{L^e} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \tag{3.25}$$

The chain is subdivided into elements N_e=2; therefore, x_M=x_3, x_1=u, and x_2 denote the lateral displacement of the middle of the chain. Performing the Finite Element Method assembly for the two finite elements and for the payload, the Lagrangian is obtained in matrix form:

$$L_g = \frac{1}{2}\dot{\mathbf{q}}^T \mathbf{M} \dot{\mathbf{q}}^T - \frac{1}{2}\mathbf{q}^T \mathbf{K} \mathbf{q} \tag{3.26}$$

where $\mathbf{q} = \begin{bmatrix} u \\ x_2 \\ x_3 \end{bmatrix}$; $\quad \dot{\mathbf{q}} = \begin{bmatrix} \dot{u} \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix}$

Substituting equation (3.24) to Lagrange equation

$$\frac{d}{dt}\left(\frac{\partial L_g}{\partial \dot{x}_i}\right) - \frac{\partial L_g}{\partial x_i} = 0 \quad i = 2,3 \tag{3.27}$$

The following equation of motion is obtained in matrix form

$$\bar{\mathbf{M}}\ddot{\mathbf{q}} + \bar{\mathbf{K}}\bar{\mathbf{q}} = \mathbf{b}_1 u + \mathbf{b}_2 \ddot{u} \tag{3.28}$$

where $\bar{\mathbf{M}} = \begin{bmatrix} \frac{2\mu L^e}{3} & \frac{\mu L^e}{6} \\ \frac{\mu L^e}{6} & \frac{\mu L^e}{3} + M \end{bmatrix}$; $\bar{\mathbf{K}} = \begin{bmatrix} \frac{F^1 + F^2}{L^e} & -\frac{F^1}{L^e} \\ -\frac{F^2}{L^e} & \frac{F^2}{L^e} \end{bmatrix}$; $\mathbf{b}_1 = \begin{bmatrix} -\frac{F^1}{L^e} \\ 0 \end{bmatrix}$; $\quad \mathbf{b}_2 = \begin{bmatrix} -\frac{\mu L^e}{6} \\ 0 \end{bmatrix}$

For the sake of simplicity, the consistent mass matrix in equation (3.28) can be replaced by a so-called lumped mass matrix, which is a diagonal matrix:

$$\bar{\mathbf{M}} = \begin{bmatrix} \mu \mathrm{L}^e & 0 \\ 0 & \frac{\mu \mathrm{L}^e}{2} + \mathrm{M} \end{bmatrix} \text{ and } \mathbf{b}_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{3.29}$$

In control theory, the state space form of the differential equation (3.28) is preferred, which is shown as:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}\mathbf{u} \tag{3.30}$$

where $\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\bar{\mathbf{M}}^{-1}\bar{\mathbf{K}} & \mathbf{0} \end{bmatrix}$; $\mathbf{b} = \begin{bmatrix} \mathbf{0} \\ \mathbf{M}^{-1}\mathbf{b}_1 \end{bmatrix}$ and $\mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix}$.

In this model, the state vector contains four state variables.

The objective of controlling the motion of the trolley is to damp the excessive swing of the payload within a short time. To achieve this, the following state feedback is used:

$$u = -\left[\hat{k}_1\left(x_2 - x_T\right) + \hat{k}_2\left(x_M - x_T\right) + \hat{k}_3\dot{x}_2 + \hat{k}_4\dot{x}_M\right] + x_T \tag{3.31}$$

where $\hat{k}_1, \hat{k}_2 \ldots, \hat{k}_4$ are gain parameters, which can be represented in a row vector $\mathbf{k}_D^T = \begin{bmatrix}\hat{k}_1 & \hat{k}_2 & \hat{k}_3 & \hat{k}_4\end{bmatrix}$.

The payload parameters $\hat{k}_2$ and $\hat{k}_4$ in (3.31) have the same role as k_1 and k_2 in equation (3.13), respectively. Therefore, the values $\hat{k}_2 = -0.5$, $\hat{k}_4 = 0.02$ are also used in this controller. Using pole placement function of Scilab 6.0.2 software and prescribing different stable poles and gain parameters, with the condition of keeping the previous values of $\hat{k}_2$ and $\hat{k}_4$, the rest of the parameters $\hat{k}_1$ and $\hat{k}_3$ are associated to the chain vibration and they could be determined: $\hat{k}_1 = -0.07$, $\hat{k}_2 = -0.5$, $\hat{k}_3 = 0.003$, $\hat{k}_4 = 0.02$ by these poles p_1=-0.0878902+2.3587906i; p_2=-0.0878902-2.3587906i; p_3=0.2896223+16.901392i; p_4=-0.2896223-16.901392i.

3.3 Controller design

The observer determines the unknown state variables of the middle of the chain: position and velocity. Only the lateral displacement of the payload is measured by the vision system. The rest of the state variables are determined by the observer in every time step *Δt*. The equation of the observer is a discrete-time state-space model, which is determined by the help of equation (3.30) and (3.31)

$$\mathbf{x}[k+1] = \mathbf{A}_D\ \mathbf{x}[k] + \mathbf{b}_D u[k] + \mathbf{l}_D(y[k] - \tilde{y}[k]); \quad \tilde{y}[k] = \mathbf{c}^T\mathbf{x}[k] \tag{3.32}$$

where $y[k]=x_M[k]$ denotes the lateral displacement of the payload, is the column vector of state variables, the term e_x is the error of the position of the payload, and the output matrix $\mathbf{c}^T$=[0 1 0 0], the coefficient matrices are calculated as follows:

$$\begin{aligned} u[k] &= -\left[k_1\left(x_2[k-1] - x_T\right) + k_2\left(x_M[k-1] - x_T\right) + k_3\dot{x}_2[k-1] + k_4\dot{x}_M[k-1]\right] + x_T \\ &= -\left[k_1\left(x_2[k-1] - x_T\right) + k_2 e_x + k_3\dot{x}_2[k-1] + k_4\dot{x}_M[k-1]\right] + x_T \end{aligned} \tag{3.34}$$

$$\mathbf{A}_D = e^{\mathbf{A}\Delta t} \tag{3.34}$$

$$\mathbf{b}_D = \mathbf{A}^{-1}\left[e^{\mathbf{A}\Delta t} - \mathbf{I}\right]\mathbf{b} \tag{3.35}$$

$$\mathbf{l}_D = \mathbf{A}^{-T}\left[e^{\mathbf{A}^T\Delta t} - \mathbf{I}\right] \tag{3.36}$$

A uniform time step Δt is used in equations (3.32) and (3.33). It is noted that in equation (3.33), the state variables of the feedback are given in time step k-1 in order to consider the delay $\tau = \Delta t$.

The anti-swing control scheme of the crane model is illustrated in Fig. 3.6. The desired position of the trolley is given in equation (3.33). The measured position of the trolley is slightly different from the desired one. This difference is gradually decreased in the controlling process.

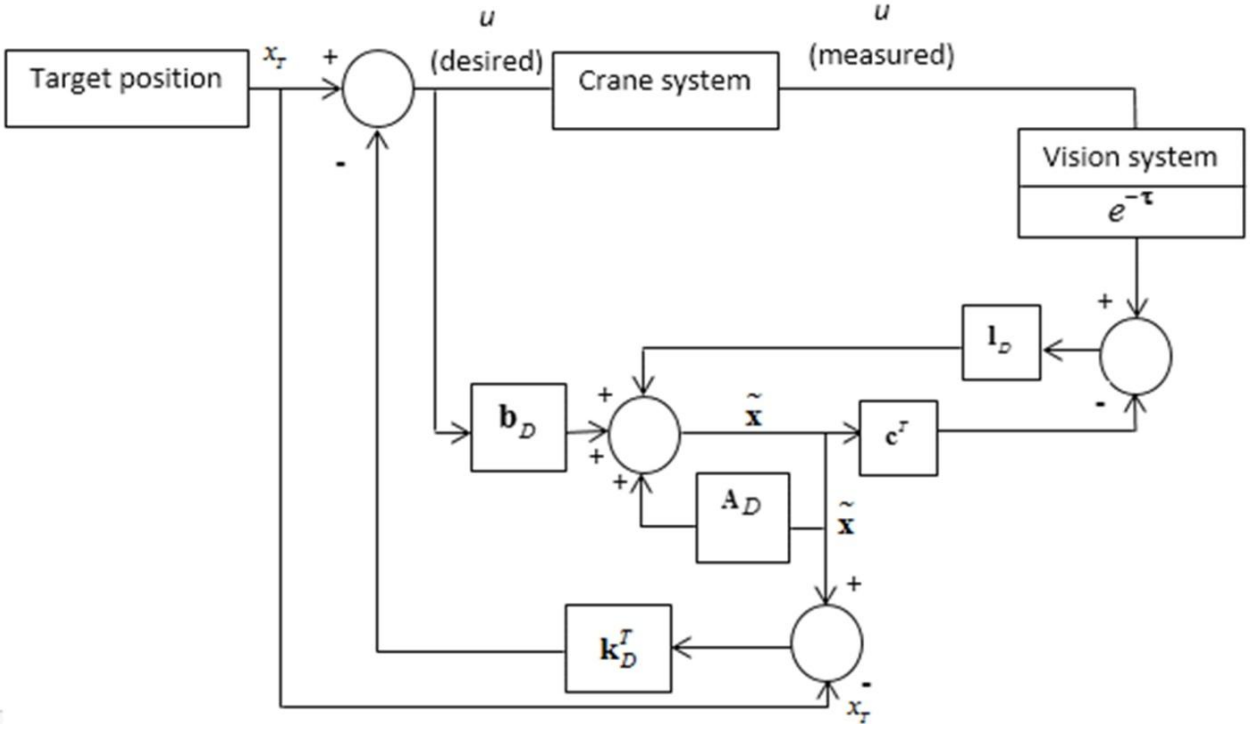

Fig.3. 6 Control scheme of the overhead crane model.

4. MACHINE VISION SYSTEM FOR AN OVERHEAD CRANE MODEL

The main goal of this chapter is to describe a position/angle measurement system of a payload using an image processing system to detect without contact the sway of a payload. In the proposed technique, the camera will capture images in a real-time environment. They will proceed to detect an object with a specified color by using the open-source computer vision library OpenCV with Visual Studio C++. The vision system measures the displacement/angle of a colored object set in the middle of the payload. By having a minimum of two frames of the desired object and the chain length, the payload swing can be estimated.

4.1 Model of a laboratory overhead crane

The scheme of the model is shown in Fig. 4.1. The model consists of :

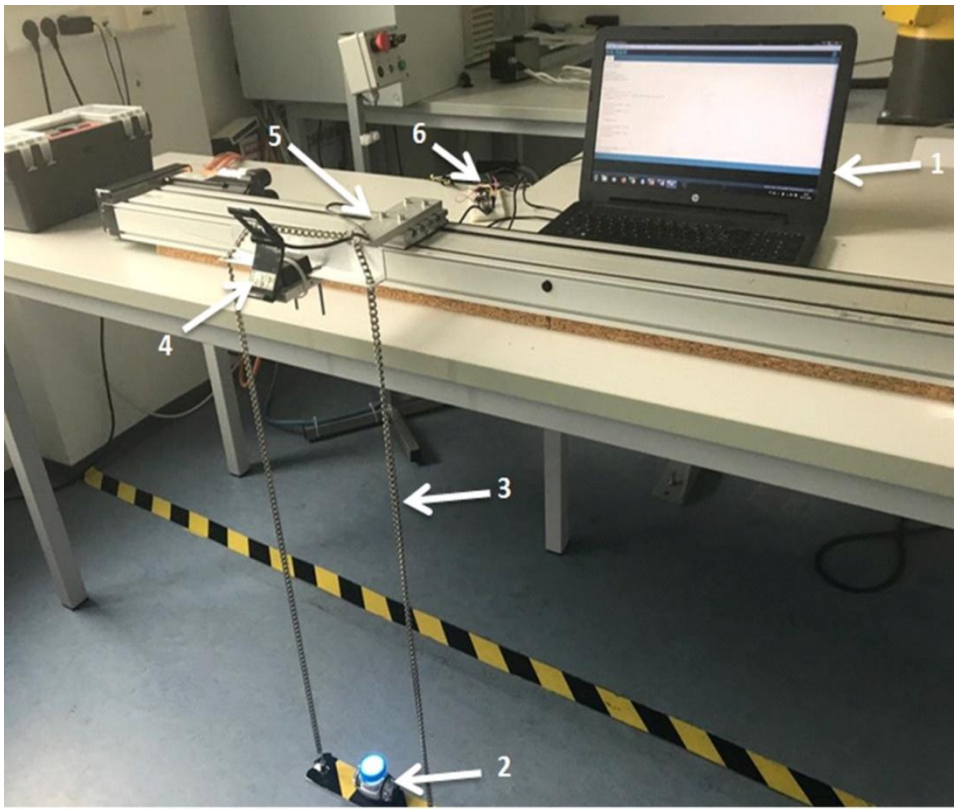

Fig.4. 1 System description: 1. Laptop, 2. Payload with blue light, 3. Chain, 4. Camera, 5. Trolley, 6. Arduino and converter.

-Trolley,

-Compact module CKK with servo motor by Rexroth see Fig. 4.2. It is a precision, ready to install linear motion system characterized by its high performance, aluminum profile, and compact design.

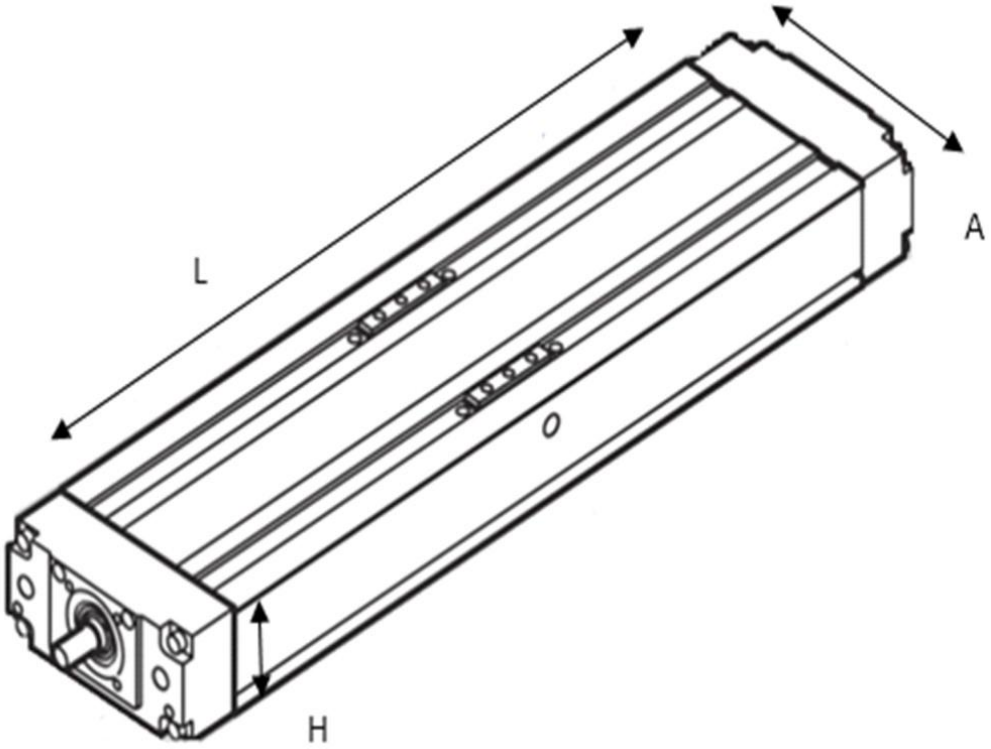

Fig.4. 2 . Compact module.

Compact module	CKK 15-110
Dimension A x H x L	110 x 50 x 1500 mm
Dynamic load capacity	15600 N
Ball screw d_0 x P	16 x 10
Max velocity V	45 m.min

Table 4. 1 Compact module characteristics.

-Personal Laptop,

CPU	I3 4005U 1.70GHz
Ram	4 GB
Hard drive	250GB 5400 rpm

Table 4. 2 Laptop characteristics.

-Web-camera,

Lens aperture	F:2.6
Lens view angle	50°
Resolution	1.3 MP
Frame rate	30 fps

Table 4. 3 web-camera characteristics.

- Visual software studio C++,

-The OpenCV (Open Source Computer Vision Library) is an open-source computer vision and machine learning software library. It has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms [84]. It has C++, Python, Java, and MATLAB interfaces and supports Windows, Linux, Android, and Mac OS.

-Payload with lighting,

-Chains,

-Arduino Uno, see Fig.4.3, is an open-source platform. It comports of a programmable circuit board (often referred to as a microcontroller) founded on the Atmega328P and software, or IDE (Integrated Development Environment) see Fig. 4.4, made to upload and write a computer program to the board [85].

Fig.4. 3 Arduino Uno.

General caracteristcs	
CPU	ATMega328
Flash memory	32 Ko
SRAM	2 Ko
EEPROM	1 Ko
Clock speed	16 MHz
Input.Output	

Input voltage	7-12 V
USB-B	For programing
Digital I.O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
Other I.O	Serie, I2C, SPI

Table 4. 4 Characteristics of Arduino Uno [85].

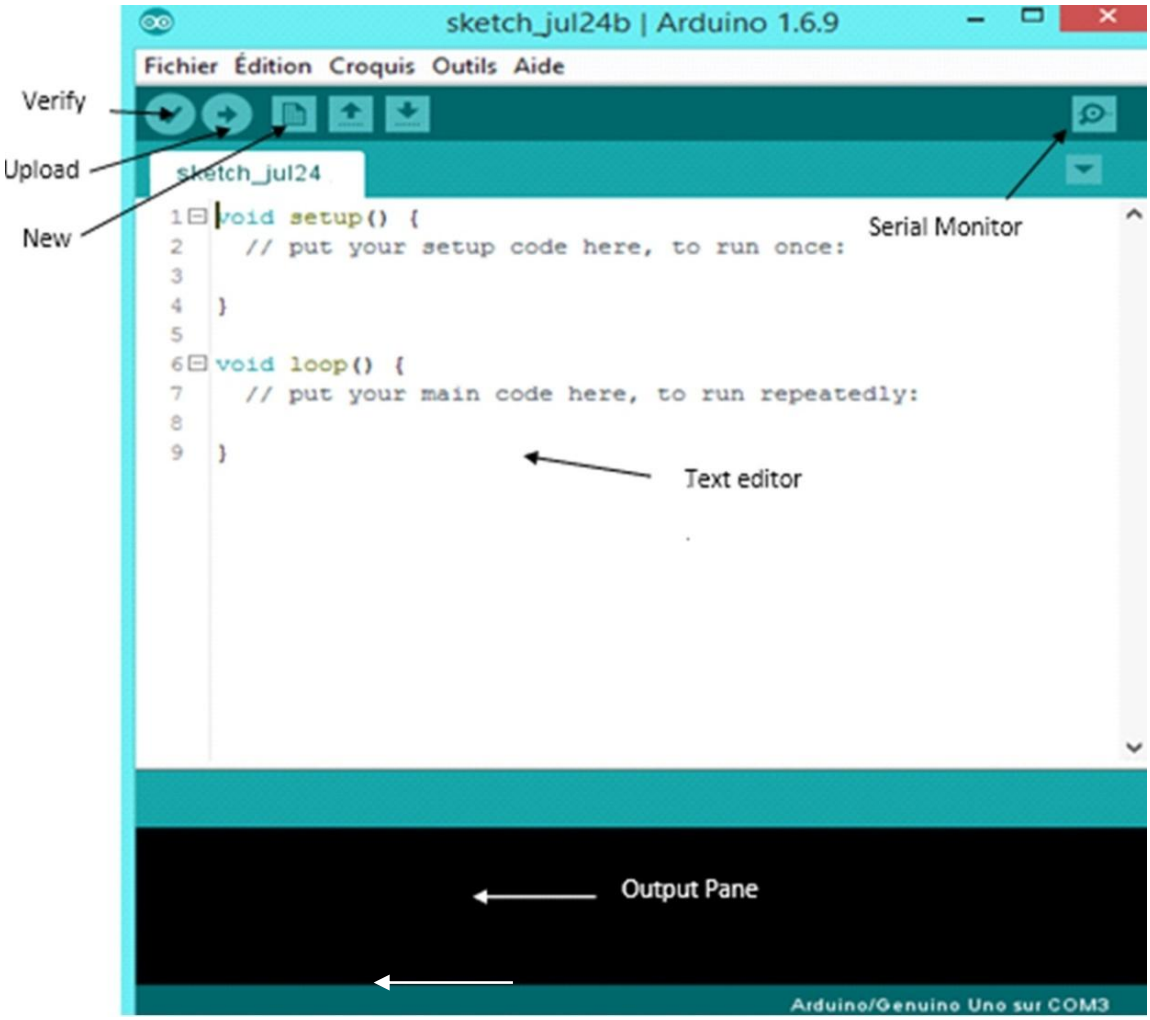

Fig.4. 4 Arduino Uno IDE interface.

The Integrated Development Environment or IDE consists of two main parts: Editor and Compiler, where the former is utilized for writing the appropriate code and then used to compile and upload it into the Arduino Module. This environment accepts both C++ and C languages. The compiler contains two parts: *void setup ()* and the *void loop ()* instructions section. The instructions written in the former part of the program will run only once after booting the microcontroller, while in the second one, the instructions will be executed in an infinite cycle. The primary code, also called a sketch, build on the IDE platform, will create a Hex File, which is sent and uploaded to the board controller.

-D/A converter PCF8591 see Fig. 4.5. The module is with a four-channel converter analog-digital (A/D). It also has one output of the digital-analog (D/A) converter with 8-bit precision, which allows only 256 steps in the analog output value or the digital output value. It works with voltage from 2.5 V to 6.0 V; it communicates via the I2C interface. Onboard are also: pulling up resistors, bus lines of 10 kΩ and potentiometer, a photoresistor, and the thermistor connected to input channels using configurable jumpers.

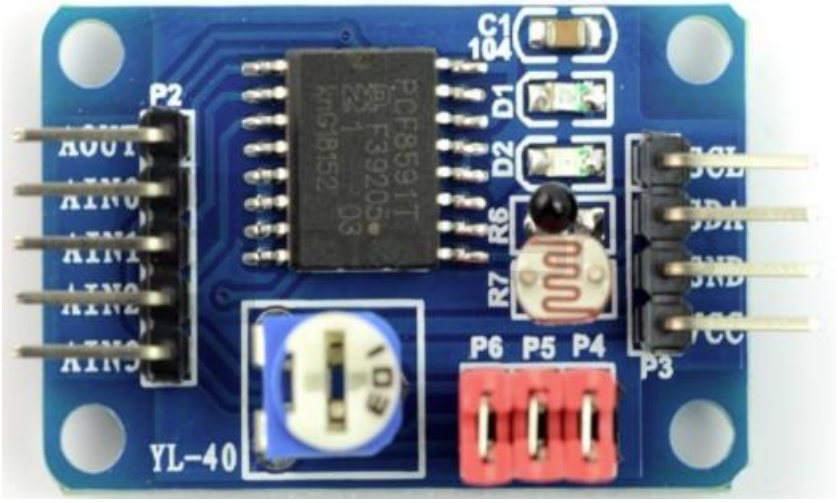

Fig.4. 5 A/D and D/A converter.

-PLC CSH01.1C type Bosch-Rexroth IndraMotion MLD.

The trolley travelling over a girder of 800 mm length. The trolley is moved via a compact module driven by an electric servo motor, mounted on the girder. The software for the image analysis of the presented vision-based sway system was developed under Visual Studio 2017 using C++ language in addition to the use of open-source computer vision library OpenCV. The hardware of the vision system is a web-camera see Table.4.3, and a personal laptop see Table. 4.2, which contains the vision program. The camera was mounted on the crane trolley in a perpendicular position, a blue light was also mounted on the middle of the payload, and its motion is within the field of view of the desktop camera.

The measured displacement of the payload is sent to an Arduino UNO, which transmits the data to a Rexroth Programmable Logic Controller (PLC) by a D/A converter. The Arduino platform is wired to the laptop by Mini-B USB cable.

4.2 Machine vision program and conversion process

The aim of the vision software is to detect and track the payload in each frame. The payload is characterized by a blue light as shown in Fig. 4.1. Each frame is analyzed and filtered to detect all the edges and then find the payload and calculate the payload displacement. The steps of the

payload detection are camera calibration, filtering and detecting the object, payload displacement calculation, and the last step is to send the data to the Arduino Uno.

-Camera calibration:

The web camera that is used in this experiment introduces distortion to images. Therefore a camera calibration should be made. This calibration is nothing but approximating the parameters of a camera to get an accurate relationship between a 3D point in the real world and its corresponding 2D projection (pixel) in the image captured by that calibrated camera. The parameters are distortion coefficients and camera matrix. The steps of camera calibration are [86].

1. Define real-world coordinates of 3D points using a checkerboard pattern of known size.
2. Capture the images of the checkerboard from different viewpoints.
3. Use *findChessboardCorners* method in OpenCV to find the pixel coordinates for each 3D point in different images.
4. Find camera parameters using *calibrateCamera* method in OpenCV, the 3D points, and the pixel coordinates.

The Camera matrix:

$$\begin{bmatrix} 9.1714031297051474e+02 & 0 & 3.2230623256829608e+02 \\ 0 & 9.1820061593023615e+02 & 2.7172433858333937e+02 \\ 0 & 0 & 1 \end{bmatrix}$$

Distortion coefficients:

$$\begin{bmatrix} -1.78807e-01 & 2.06882e+00 & 1.18571e-03 & 3.17260e-03 & -7.223233e+00 \end{bmatrix}$$

The next step is to use in the main program the camera matrix and distortion coefficient with the OpenCV function *initUndistortRectifyMap* once since it is an expensive method and the parameters of the camera are fixed and to call the function *remap* of openCV on each frame [86]. Using this way will improve the performance of the program significantly see Fig. 4.6.

```
Mat cameraMatrix = (Mat1d(3, 3) << 8.4934882687830418e+02, 0., 3.0533519545834918e+02,
    0., 8.4552604109224899e+02, 2.7226934023759043e+02,
    0., 0., 1.);
Mat distortionCoefficients = (Mat1d(1, 5) << -2.4659367461769788e-01, 2.2388409002544090e+00,
    -3.2064569929815363e-03, -6.5504274270249377e-05, -6.9838556726560199e+00);
Mat imgOriginal, map1, map2;
int i = 0, bb = 0;

//arduino.is Connected (inside while)
while (1)
{
    cap >> image_frame1; // get a new frame from the camera

    if (bb == 0)
    {
      initUndistortRectifyMap(cameraMatrix, distortionCoefficients, cv::Mat(), cv::Mat(),
          image_frame1.size(), CV_16SC2, map1, map2);
      bb = bb + 1;
    }
    cv::remap(image_frame1, image_frame1, map1, map2, cv::INTER_LINEAR, cv::BORDER_CONSTANT);
    cv::flip(image_frame1, image_frame1, 1);  // flliping a camera to get mirror effect
    if (image_frame1.empty()) break;
```

Fig.4. 6 Camera calibration program.

-Filtering and detecting:

After the calibration of the image, the image is converted from RGB color to HSV using the function *cvtColor*, which is included in the OpenCV library. The reason for this is that HSV separate color components from the intensity, unlike RGB, the advantages of this is its robustness to light changes and removing shadows. The so-called *inRange* function is used to filtrate with a minimum (0, 0, 255) and maximum (81, 54, 255) threshold value, which is set and then using a combination of morphological operations like erode and dilate for blurring and noise reduction to make the payload visible for positions estimation as shown in Fig. 4.7. After noise reduction, a round contour (payload) with the specified area is searched using a *findcontours* and *drawContours* functions to obtain the coordinates x and y of the payload. Then the position of the center of mass (centroid) of the payload is calculated and detected using the as seen in Fig. 4.8, a portion of the C++ program of filtering and detection is shown in Fig. 4.9. The implementation of blue light on top of the payload will allows the system to measure the displacement/angle in different environment lights

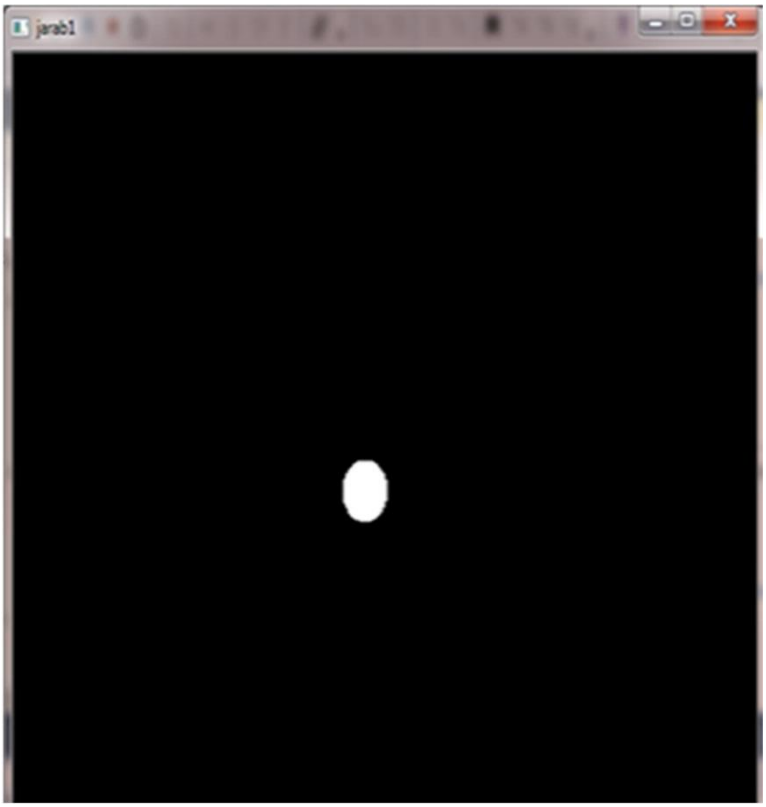

Fig.4. 7 Colour threshold and noise removal.

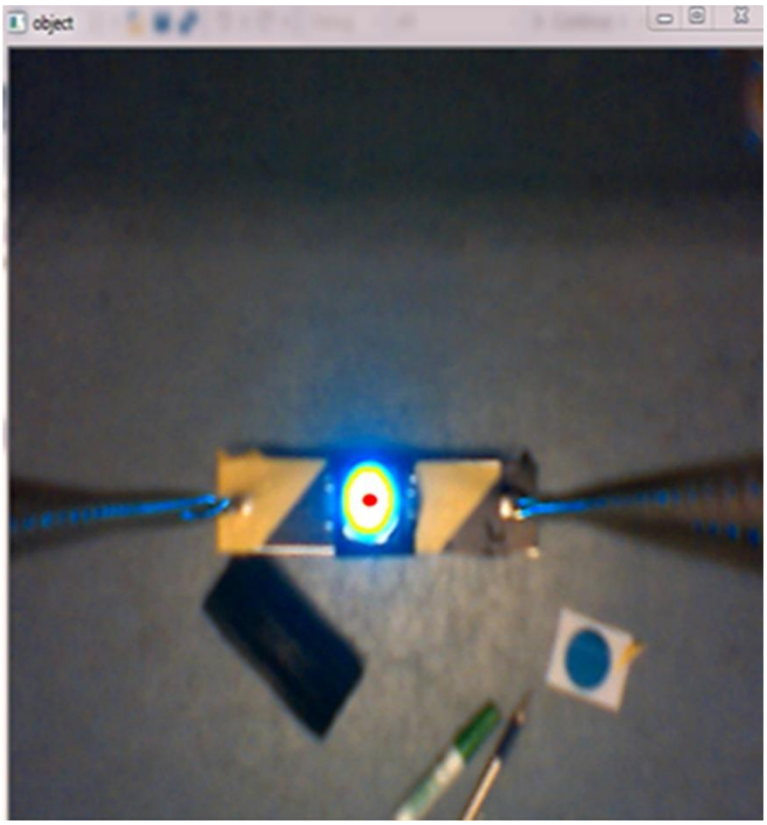

Fig.4. 8 Centroid location of the payload.

```
cv::cvtColor(image_frame1, image_HSV, CV_BGR2HSV);
cv::inRange(image_HSV, cv::Scalar(0, 0, 255), cv::Scalar(81, 54, 255), image_Color2);

erode(image_Color2, image_Color2, getStructuringElement(MORPH_ELLIPSE, Size(5, 5)));
dilate(image_Color2, image_Color2, getStructuringElement(MORPH_ELLIPSE, Size(5, 5)));
dilate(image_Color2, image_Color2, getStructuringElement(MORPH_ELLIPSE, Size(5, 5)));
erode(image_Color2, image_Color2, getStructuringElement(MORPH_ELLIPSE, Size(5, 5)));

findContours(image_Color2, contours_2, hierarchy_2, CV_RETR_EXTERNAL, CV_CHAIN_APPROX_SIMPLE,
    Point(0, 0));
vector<Moments> mu_2(contours_2.size()); //init V.M called mu, V size the Nbr of contours
for (int i = 0; i < contours_2.size(); i++)
{
    mu_2[i] = moments(contours_2[i], false);
}
vector<Point2f> mc_2(contours_2.size()); //vector to store all the center points of the contours
for (int i = 0; i < contours_2.size(); i++)
{
    mc_2[i] = cv::Point2f(mu_2[i].m10 / mu_2[i].m00, mu_2[i].m01 / mu_2[i].m00);
}
for (int i = 0; i< contours_2.size(); i++)
{
    if (mu_2[i].m00 > 1000) {
        center_2 = mc_2[i];
        drawContours(image_frame1, contours_2, i, color_2, 2, 8, hierarchy_2, 0, cv::Point());
        circle(image_frame1, center_2, 4, color_2, -1, 8, 0);
        line(image_frame1, center_2, Point(324, 291), color_2, 4, 8, 0);        }
}
```

Fig.4. 9 Filtring and detection program.

-Payload displacement calculation:

The principal of payload displacement estimating in the individual step is shown in Fig. 4.10. Before starting, an assumption is made that in a specified time interval, the chain length is constant, and the swing of the payload is relatively small. Taking into account those conditions, the movement of the payload is not performed as an arc. The study of the geometry shape of the given measurement system shows a right triangle, where i0-i2 movement is the payload displacement in pixel, and the i2-a distance is the hypotenuse see Fig. 4.11. Finally, we can approximate the payload displacement in pixel by having a minimum of two frames of the payload and substituting their coordinates. In aware of the base position, the length of the chain and using the Pythagoras theorem, we can get the sway of the payload:

$$Displacement = i0 - i2 \tag{3.1}$$

$$\tan\beta = \frac{Displacement}{Chain\ legth} \tag{3.2}$$

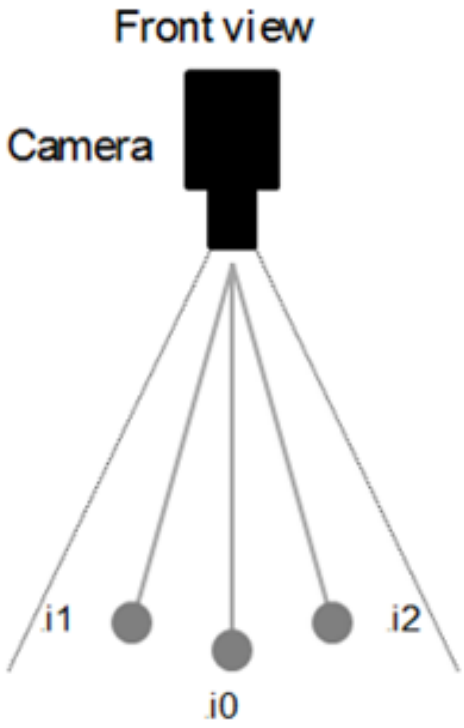

Fig.4. 10 Payload position measurement system.

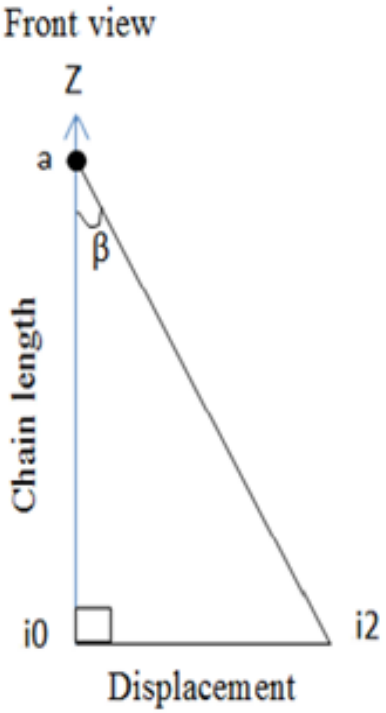

Fig.4. 11 The geometry of the payload displacement measurement system.

Fig. 4.12 shows the displacement equation implementation to the main program, which is multiplied by the size of the pixel in cm. To distinguish the swinging of the payload is right or left, we multiplied the displacement result by a positive value +1 in case the payload is on the right side of the initial axial position; otherwise, multiply it by negative value -1. Then displacement is

multiplied by (& 0xff) to effectively masks the variable, so it leaves only the value in the last 8 bits and ignores all the rest of the bits. The final step is to send the measured value to the Arduino UNO via *Arduino.writeSerialPort().*

```
int data = (sqrt(pow((324.0 - center_2.x), 2))*0.06031);
if (center_2.x >= 324)
{
    data = data * 1;
}
else
{
    data = data*(-1);
}
std::cout << "  Point y " << center_2.y << "  Point x " << center_2.x << "  Distance :" << data << " cm " << '\n';
data = data + 122;
char LSB;
LSB = data & 0xff;
char *charArray = &LSB;
arduino.writeSerialPort(charArray, 1);
delete[] charArray;
```

Fig.4. 12 Arduino UNO program.

The vision system was optimized to give 20 measurements per one second. The measured data will be sent to the PLC for analysis via Arduino UNO and D/A converter. Between the Arduino UNO and the D/A converter, the I^2C communication protocol is performed by SDA (Serial Data) and SCL (Serial Clock) pins, and the output voltage is 5 V. A program code was written in Arduino IDE software, a portion of the code shown in Fig. 4.13.

```
#include "Wire.h"
#include "EEPROM.h"
#define PCF8591 (0x90 >> 1)
int MSBLSB;
char info;
int voltage;
int ledPin = 6;
//int seged=0;
void setup() {
Serial.begin(9600);
 Wire.begin(); }
void loop() {
if (Serial.available ()>0)  {
info = Serial.read();  }
MSBLSB= info & 0xFF;
voltage=MSBLSB*(200./245.)+10;
Wire.beginTransmission(PCF8591);
Wire.write(0x40);
Wire.write(voltage);
Wire.endTransmission();
    }
```

Fig.4. 13 Arduino UNO program.

5. RESULTS OF SIMULATIONS AND EXPERIMENTS

The displacement of the lighted payload is detected by a web camera using a machine vision system based on Visual Studio C++ and OpenCV library. The measured data of the displacements are transferred to a PLC (CSH01.1C type Bosch-Rexroth IndraMotion MLD) via Arduino Nano and a D/A converter with a frequency of 20 Hz.

The positioning of the trolley can be prescribed by four parameters: relative distance, velocity, acceleration, and deceleration. Those are the input parameters of a PLC function that controls the position of the trolley. In general, the execution process of the function consists of three subsequent phases: acceleration, constant velocity and deceleration, written by the following equations, respectively:

$$v_{i+1} = v_i + a_i \Delta t_i, \quad s_{i+1} = s_i + v_i \Delta t_i + \frac{1}{2} a_i \left(\Delta t_i\right)^2 \tag{5.1}$$

$$v_{i+2} = v_{i+1}, \quad s_{i+2} = s_{i+1} + v_{i+1} \Delta t_{i+1} \tag{5.2}$$

$$v_{i+3} = v_{i+2} - a_{i+1} \Delta t_{i+2}, \quad s_{i+3} = s_{i+2} + v_{i+2} \Delta t_{i+2} - \frac{1}{2} a_{i+1} \left(\Delta t_{i+2}\right)^2 \tag{5.3}$$

Only the first phase given in (5.1) will be allowed to be performed by recalling the PLC position controller at every Δt_i to achieve the desired motion of the trolley.

The positioning of the payload can be considered as a tracking problem with minimizing the swing of the payload. The proposed closed-loop control system integrates an observer, which is integrated into the PLC program.

The main PLC program is written in Sequential Function Charts (SFC) see Fig. 5.1. It has five steps. The first step is related to the initialization of the parameters, and the second is to position the trolley to the desired starting point. The third contains the Continuous Function Chart (CFC) program of the controlling and observer of the system Fig. 5.2. The last one is to stop after the swinging of the payload is removed.

When the program start and the base position is reached, a timer TON_1 starts counting up to a specified value. After the controlling process start and the condition of the process to be complete is the trolley reaches the desired position without swinging. An emergency stop is available in the program where we can stop it anytime. The controlling portion of the program is a PLC program is developed using Continuous Function Chart (CFC) to perform experiments. Due to the size of the program, the block diagram is subdivided into initialization and the main control blocks. The

initialization of the CFC code is shown in Fig. 5.2, and the blocks of the main control code are displayed in Fig. 5.3.

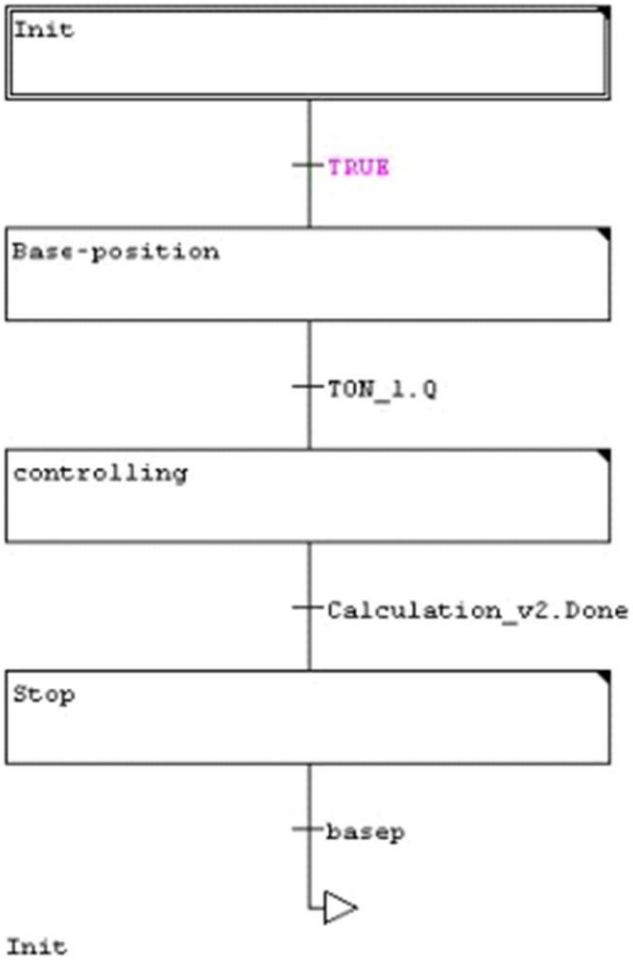

Fig.5. 1 Main PLC program.

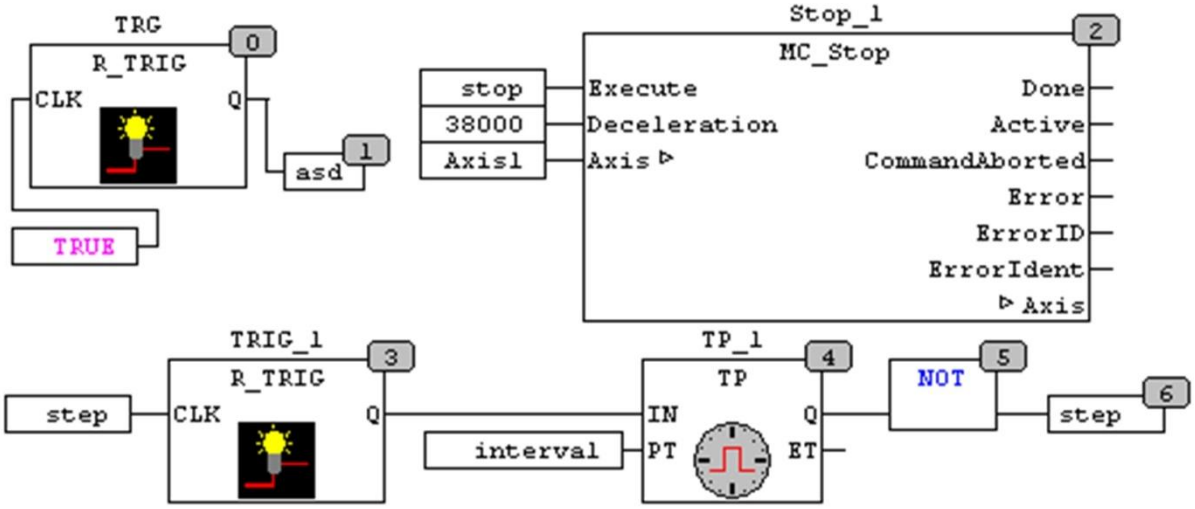

Fig.5. 2 Initialization of the controlling code.

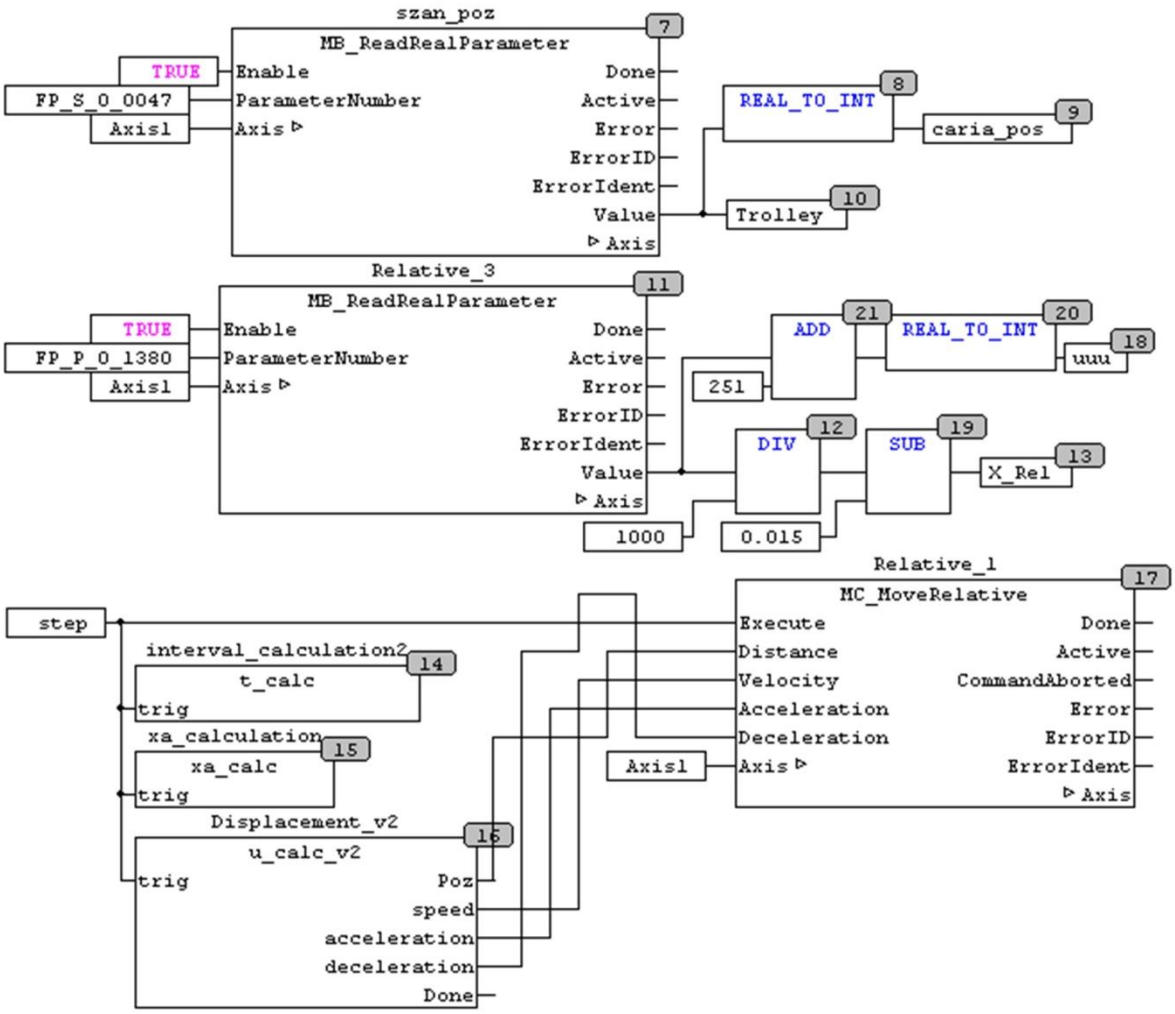

Fig.5. 3 Controlling step of the main PLC program.

The PLC program contains blocks numbered from 0 to 21 at the upper right corners. The blocks 0 to 2 are responsible for the start/stop of the system. The sampling time *dt* is generated in blocks from 3 to 6. Blocks from 7 to 10 determine the position *u* (measured) of the trolley. The relative position x_r of the payload is given by the blocks from 11 to 13. The observer and the execution of the position *u* (desired) are given in blocks from 14 to 17. The rest of the blocks perform only data conversion.

-First experiment: the vision system without controlling will be tested:

Controlling	No
Chain length	71 mm
Payload weight	0,419 kg
Trajectory length	800 mm

Trolley speed	0.8 m.s

Table 5. 1 Experiment parameters.

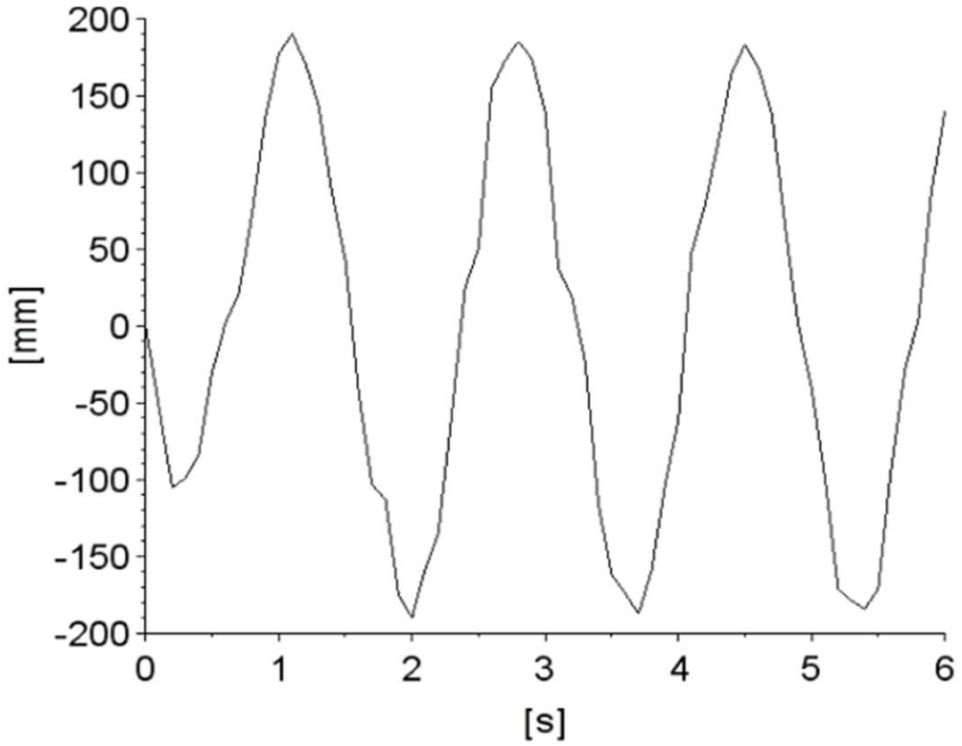

Fig.5. 4 Displacement of the payload in x direction.

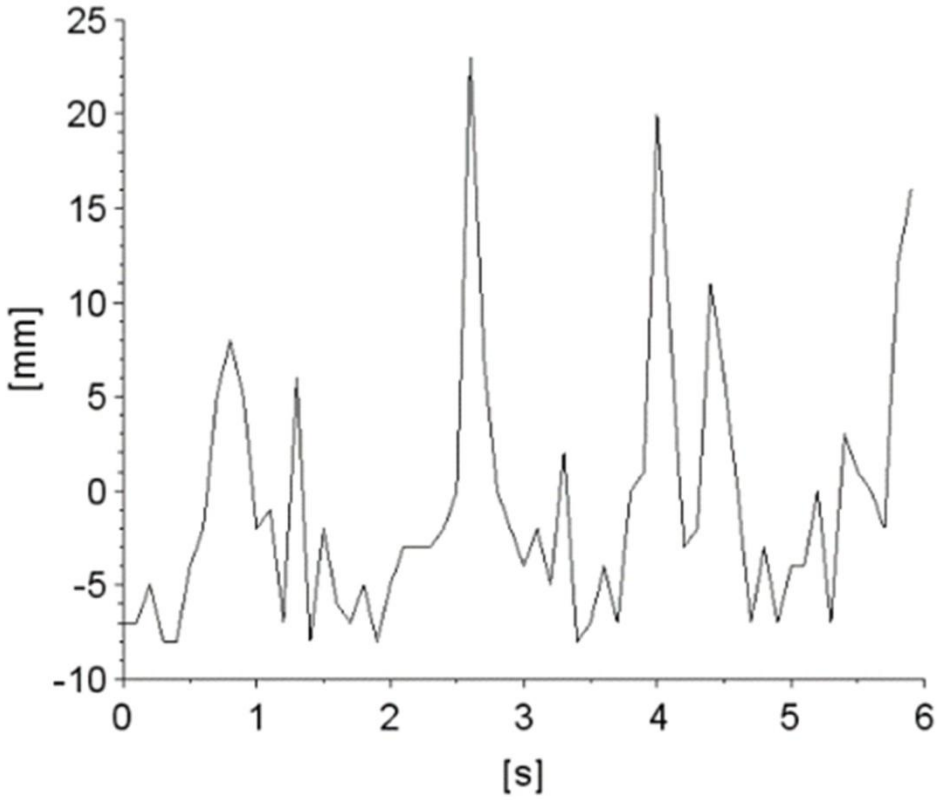

Fig.5. 5 Displacement of the payload in y direction.

The results in Fig. 5.1 and Fig. 5.2 demonstrate that the proposed vision system for payload displacement measurement under daylight condition can provide coordinates with relatively small sampling time. One of the advantages of this solution is that only one camera is used, and it can measure the displacement of the payload in both directions, *x* and *y*.

In the next experiment for controlling validation, only the displacement in *x* direction will be considered due to its high magnitude compared to the displacement in y direction.

-Second experiment:

The parameters of the experiment are the same as the first one, but this time with controlling is on. The time increment *dt* is equal to 0.15 s and for the controlling the gain parameters and the matrixes of the observer are as follows:

$$\hat{k}_1 = -0.07, \hat{k}_2 = -0.5, \hat{k}_3 = 0.003, \hat{k}_4 = 0.02$$

$$\mathbf{A}_D(\Delta t = 0.15s) = \begin{bmatrix} -0.7371381 & 0.7629382 & 0.036768 & 0.0506495 \\ 0.1479837 & 0.7887384 & 0.0098243 & 0.1380671 \\ -8.9574487 & 3.4067666 & -0.7460993 & 0.7663399 \\ 0.6607952 & -2.1439156 & 0.1486435 & 0.7865806 \end{bmatrix}$$

$$\mathbf{b}_D(\Delta t = 0.15s) = \begin{bmatrix} 0.9741999 \\ 0.0632779 \\ 5.5506821 \\ 1.4831203 \end{bmatrix}, \mathbf{l}_D(\Delta t = 0.15s) = \begin{bmatrix} -0.4415241 \\ -0.010011 \\ 25.405014 \\ -3.4306935 \end{bmatrix}$$

A simulation program has been developed under Scilab 6.0.2 software based on equations (3.32)-(3.36). The results of the simulation and the experiment in Fig. 5.6 and Fig. 5.7 show a good agreement of the trolley motions.

The simulation model correlates with the experimental results closely. However, results obtained for the relative motion of the payload display higher discrepancies in the beginning, but after four seconds, the results are converging. A very small fluctuation is seen after four seconds can be explained by the digital circuit employed in the experimental setup. The overshooting for the experiment and the model for trolley motion are 87 mm and 98 mm, respectively. The relative motions of the payload converge to 0 between 5-6 seconds.

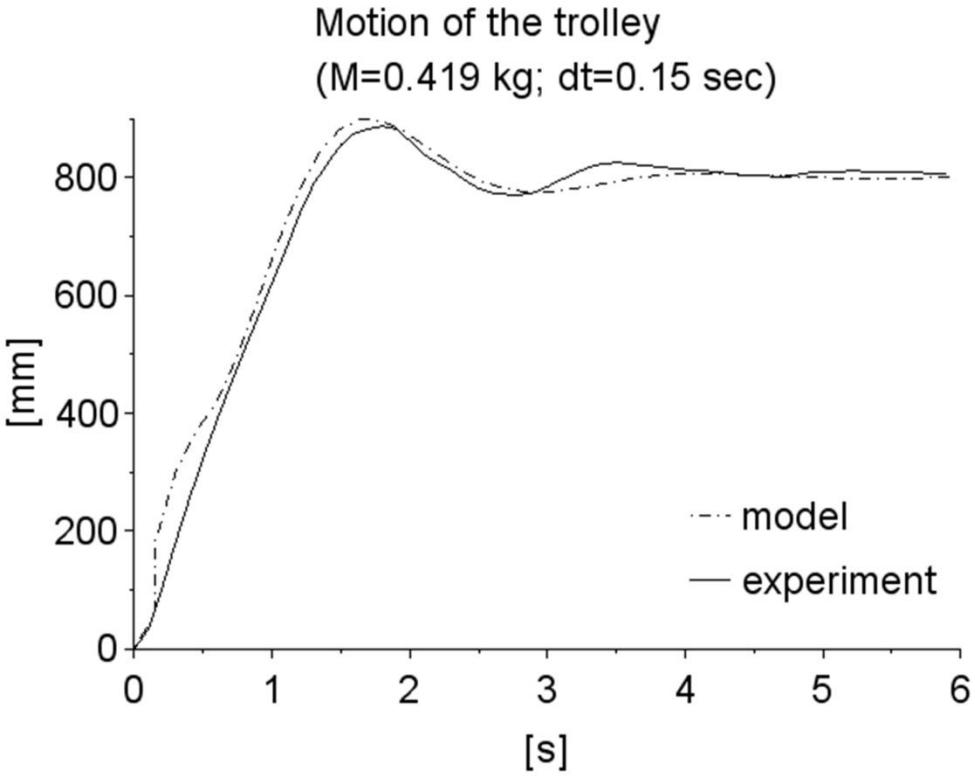

Fig.5. 6 Comparison of the trolley motion simulation model and the experimental.

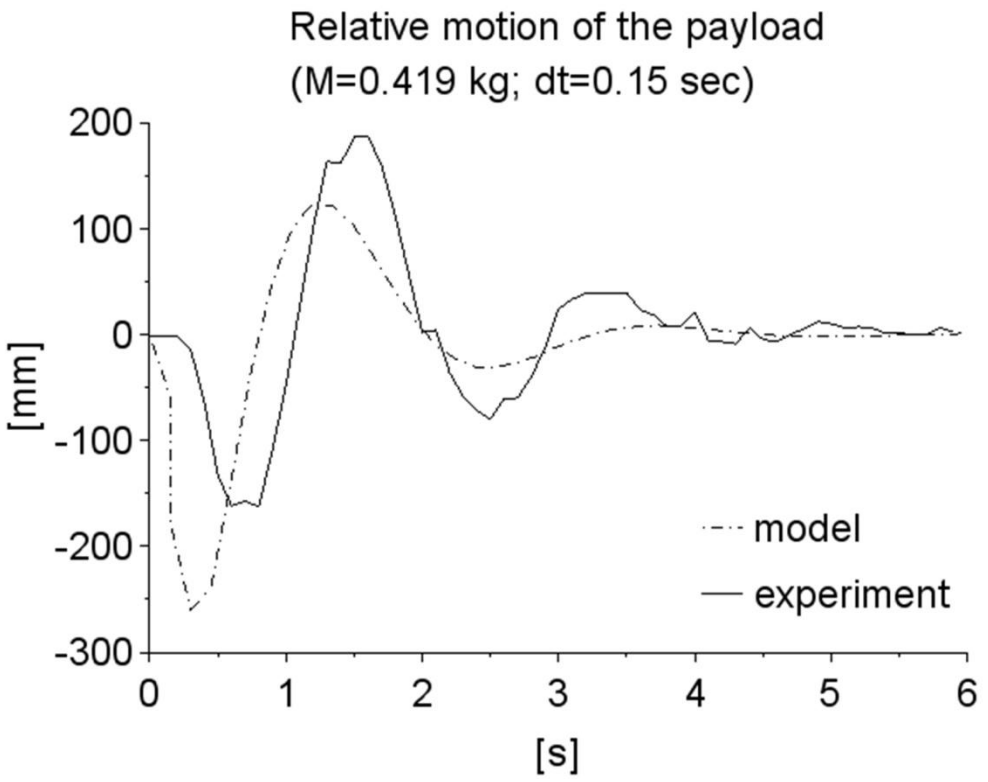

Fig.5. 7 Comparison of the relative motion of the payload simulation model and the experimental.

-Second experiment:

The responses of the proposed controller have been tested for different gain parameters of the payload $\hat{k}_2, \hat{k}_4$, with time step *dt*=0.15 s. The previous gain parameters $\hat{k}_1, \hat{k}_3$ will be maintained.

The parameter $\hat{k}_4 = 0.02$ is constant while different values $\hat{k}_2$ were tested, as shown in Fig. 5.8 and Fig. 5.9. The experimental results show that the trolley movement is not sensitive to the parameter change while the payload is a little bit more.

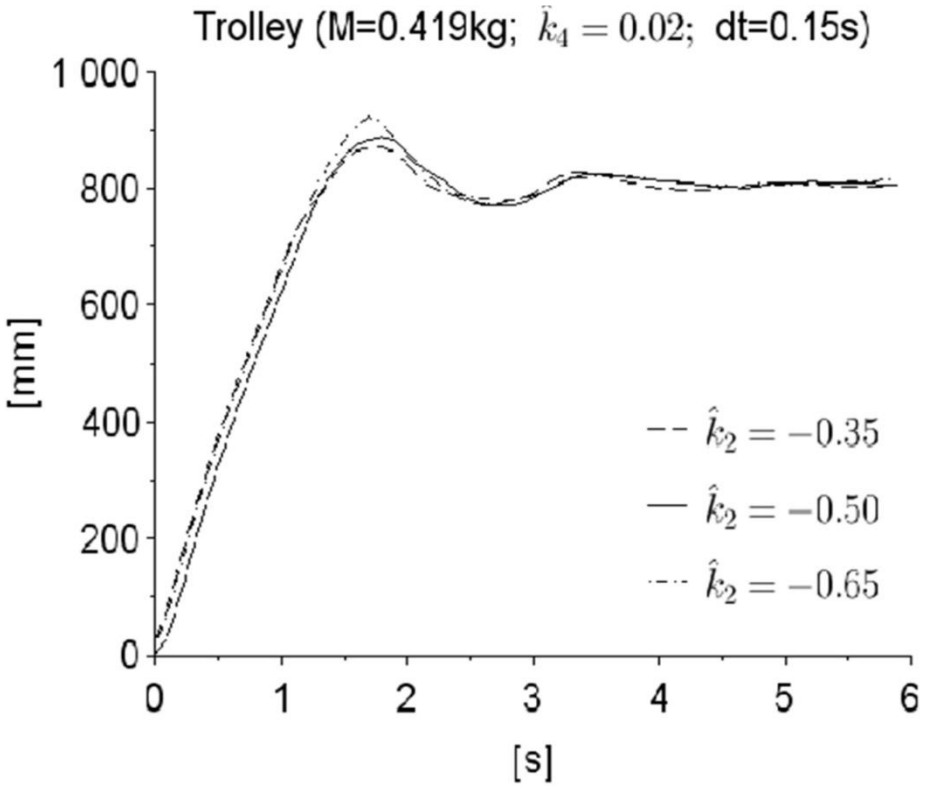

Fig.5. 8 Comparison of the trolley motion of anti-swing control for different $\hat{k}_2$ *parameters.*

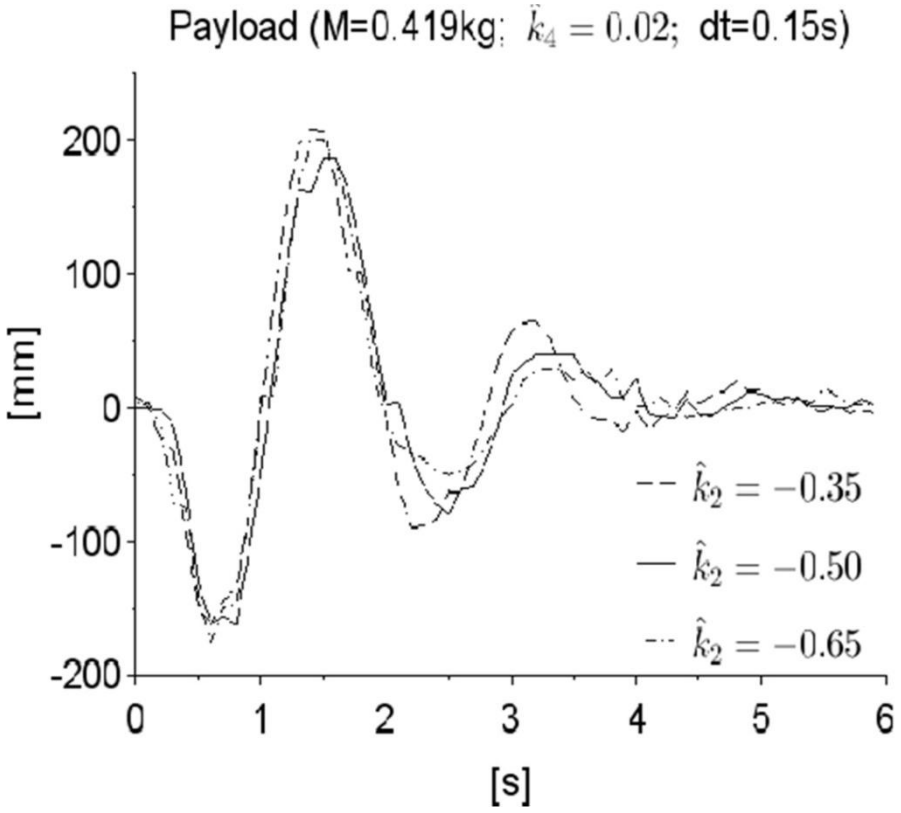

Fig.5. 9 Comparison of the payload results of anti-swing control for different $\hat{k}_2$ *parameters.*

-Third experiment:

In the following experiment, the parameter $\hat{k}_2 = -0.5$ is constant while different values $\hat{k}_4$ were tested, as shown in Fig. 5.10 and Fig. 5.11.

The experimental results in Fig. 5.10 and Fig. 5.11 show that the motion of the system is more sensitive to the changes of this gain parameter compared to $\hat{k}_2$.

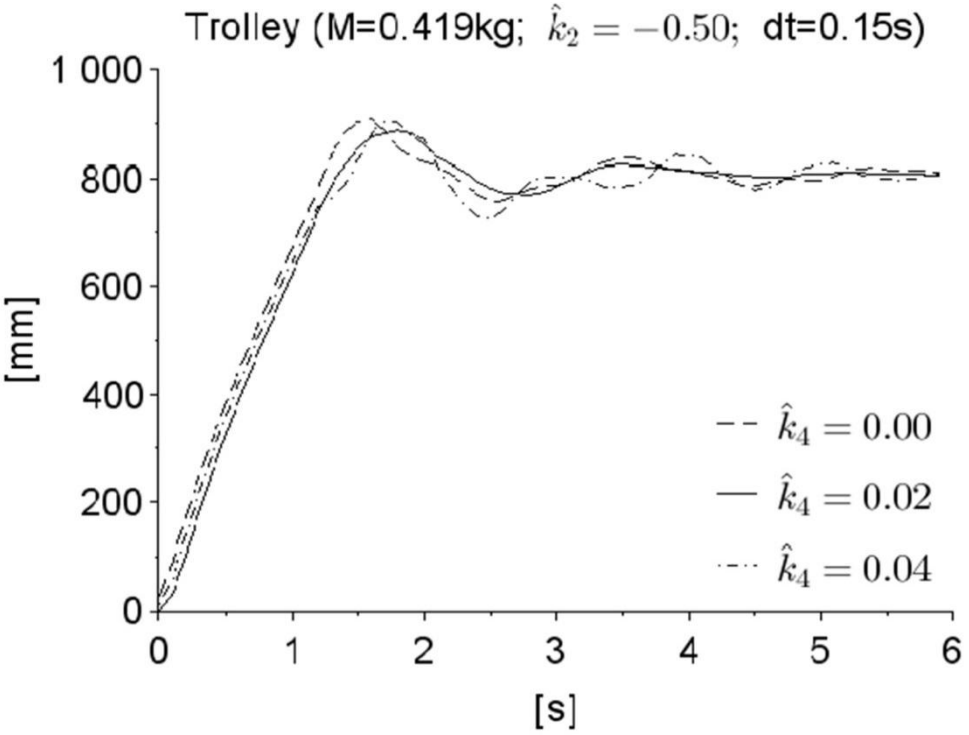

Fig.5. 10 Comparison of the trolley motion of anti-swing control for different $\hat{k}_4$ parameters.

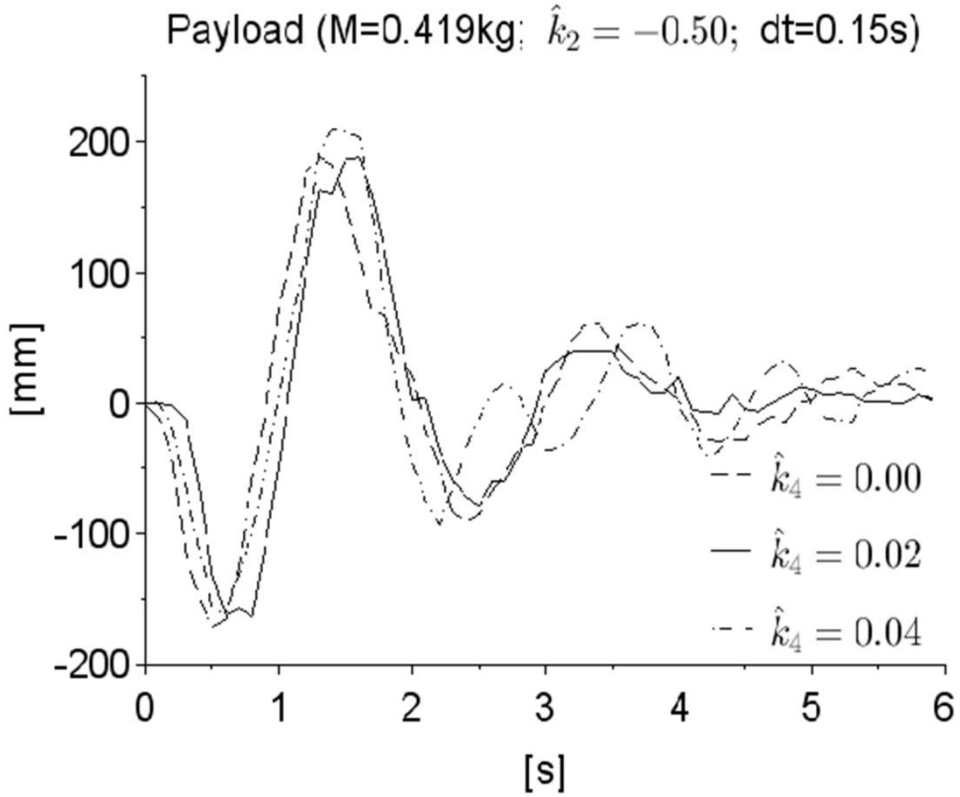

Fig.5. 11 Comparison of the payload results of anti-swing control for different $\hat{k}_4$ parameters.

-Fourth experiment:

Finally different time increments Δt[0.10 s; 0.15 s; 0.20 s] have been tested.

The matrices of the observers are given as follows:

$$\mathbf{A}_D(\Delta t = 0.10s) = \begin{bmatrix} -0.0742061 & 0.4830562 & 0.0591938 & 0.0184559 \\ 0.0936962 & 0.8919063 & 0.0035798 & 0.0961056 \\ -15.968022 & 7.0318315 & -0.0901801 & 0.4900862 \\ 1.3639328 & -1.9043587 & 0.0950598 & 0.8899923 \end{bmatrix}$$

$$\mathbf{b}_D(\Delta t = 0.10s) = \begin{bmatrix} 0.5911499 \\ 0.0143975 \\ 8.9361901 \\ 0.5404258 \end{bmatrix}; \quad \mathbf{l}_D(\Delta t = 0.10s) = \begin{bmatrix} -0.7987121 \\ 0.0580099 \\ 16.063697 \\ -2.2193061 \end{bmatrix}$$

$$\mathbf{A}_D(\Delta t = 0.20s) = \begin{bmatrix} -0.869268 & 0.7760892 & -0.006247 & 0.0904905 \\ 0.1505345 & 0.6829104 & 0.017552 & 0.1747341 \\ 3.9522234 & -3.0091483 & -0.8653151 & 0.773071 \\ -0.583671 & -2.0660731 & 0.1499491 & 0.6808269 \end{bmatrix}$$

$$\mathbf{b}_D(\Delta t = 0.20s) = \begin{bmatrix} 1.0931789 \\ 0.166555 \\ -0.9430665 \\ 2.6497433 \end{bmatrix}; \quad \mathbf{l}_D(\Delta t = 0.20s) = \begin{bmatrix} 0.1984911 \\ -0.1208698 \\ 26.689188 \\ -3.8673205 \end{bmatrix}$$

The experimental results are displayed in Fig. 5.12 and Fig. 5.13. For the time increments Δt equal to 0.15s and 0.20s, the solutions convergent. However, for Δt=0.10s it gives undesirable oscillations at the vicinity of the target position.

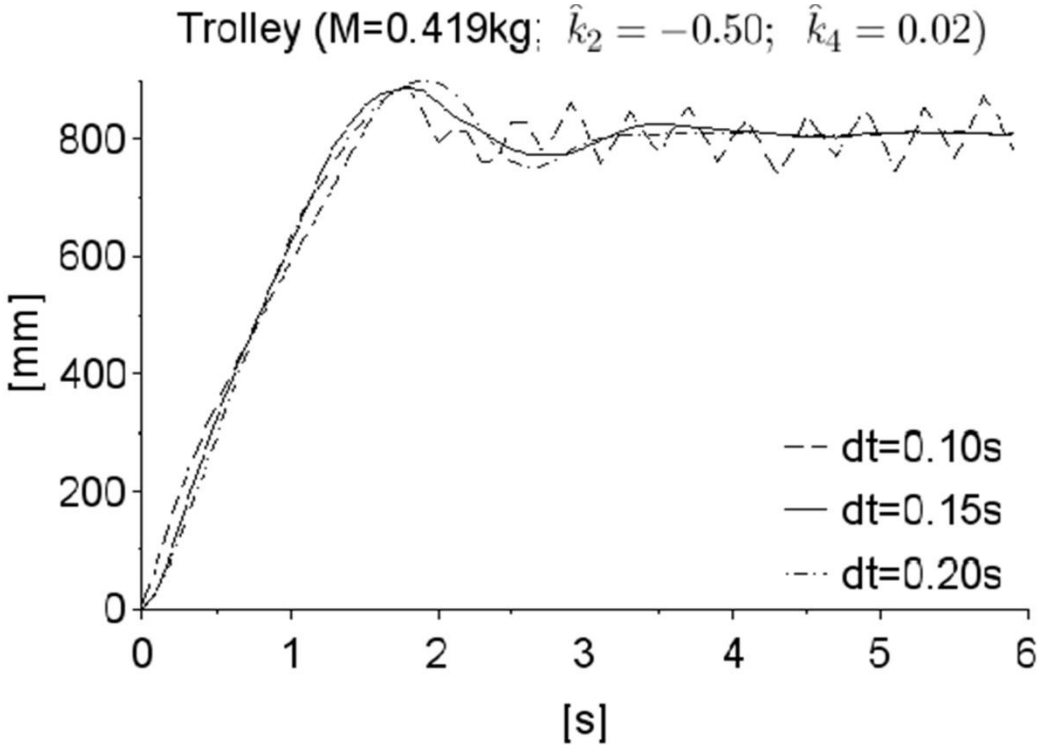

Fig.5. 12 Comparison of the trolley motion results of anti-swing control for different time steps.

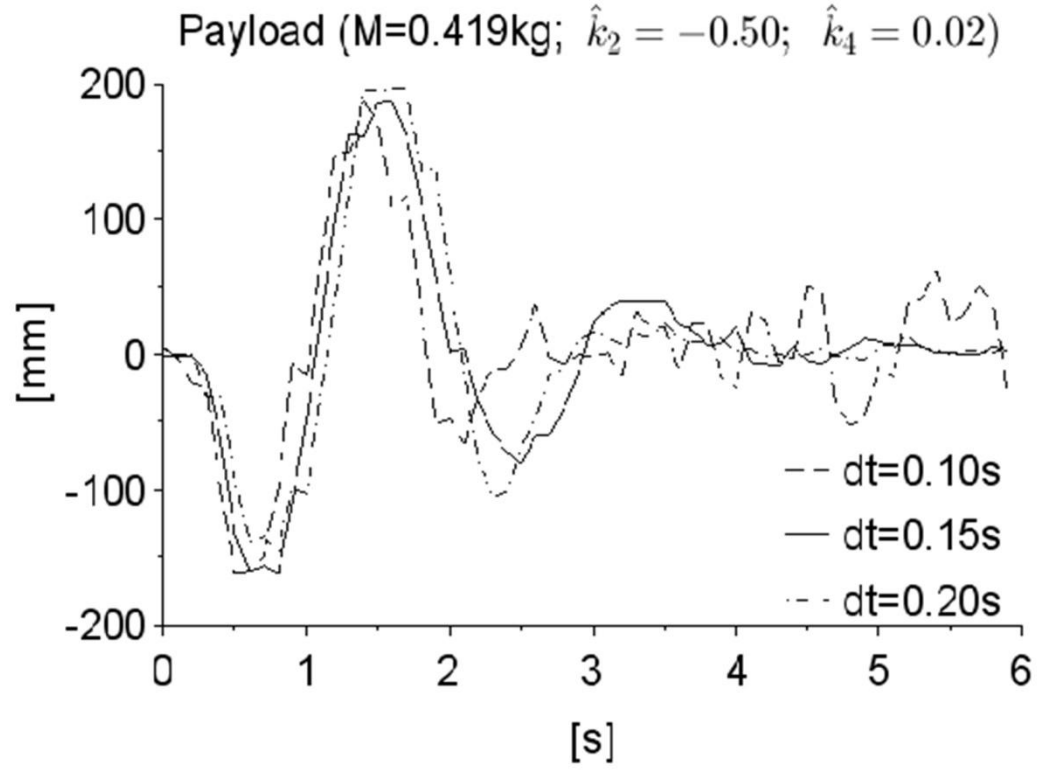

Fig.5. 13 Comparison of the payload results of anti-swing control for different time steps.

The simulation results show that the displacement of the trolley and the swinging of the payload can be damped in approximately 4 to 6 seconds. The trolley quickly arrives at its target position

within 2 seconds with no excessive overshooting. Then an additional 2 to 3 seconds are needed to damp the vibration of the payload thoroughly.

The results of the newly proposed method could be compared with the paper [80]. However, the presented experimental setup is slightly bigger in sizes and weights, and the suspending element of the payload is a chain instead of a straight, rigid bar in this paper. Also, the investigated model in this paper has multi-degrees freedoms, while in the previously mentioned paper, the model has only two degrees. Also, the trolley reaches the target position much slower than in the presented work. Though this fact, the swinging of the payload was damped around at the same time.

6. THESES – NEW SCIENTIFIC RESULTS

T1. A linear and nonlinear model of the overhead crane has been developed considering the vibration of the payload and also the suspending chain. The simulation results of both models, linear and nonlinear, show a good agreement, making the linear model a good candidate to be used as an observer in order to provide the non-measurable state-variables for the feedback. Related publication: [S2]

T2. It is assumed that the mass of the payload is significantly higher than the suspending chain of the crane system. The stability region of the dynamic system has been expresed by two gain parameters using a D-subdivision method of the simple pendulum model. The drawn curves of the stability regions are shrinking as the time delay is increasing (see Fig. 3.2). The boundary curves are intersecting each other at point $k_1=0$, $k_2=0$. Therefore, choosing the parameters within the interval of $(-2<k_1<0)$ at the vicinity of $k_2=0$, will make the system stable and robust, since it remains in always in stable state with a high range of time delay. Related publication: [S4]

T3. A non-contact position measurement system for the payload of the overhead crane was developed. In the proposed technique a web-camera is used as an image sensor for real-time measurement of the displacement/angle of the payload. Visual Studio C++ with open-source computer vision library OpenCV is used for image analysis and calculations. The measurement system is robust due to the implementation of blue light source on top of the payload, which allows the system to measure the displacement/angle in different light circumstances. The vision system is optimized to give 20 measurements per one second. The experimental results proved the effectiveness of the developed vision system. Related publication: [S1, S3]

T4. The newly designed controller is based on a two-level hierarchical approach. At the first level, the time delay due to the vision system is considered by the use of the D-subdivision method of T2, which provides the gain parameters assosiated with the state-variables of the payload. In the second level, the extended model is also considering the vibration of the suspending

chain, and its gain parameters are determined with the use of pole placement method. The anti-swing control of the overhead crane system is performed by a PLC program, which contains the collocated observer of T1. The robustness of the proposed controller has been validated by experimental measurements and it proved to be competitive with other methods published in literatures.

Related publication: [S4]

7. SUMMARY

The aim of this work was to design a controller for an overhead crane i.e. robust, quick, and feasible. The controller could quickly transfer the payload from the start point until the target position and without excessive overshooting of the payload and the trolley. A vision system is developed for the measurement of the relative displacement payload. The delay due to the vision system is considered in the design of the controller. Furthermore, the vibration of the chain is also included. An observer was developed to determine the unmeasured state variables.

To reach our objective, we have formulated a linear and a nonlinear mathematical model of an overhead crane using a finite element method. In addition to the payload, the inertia and independent degree of freedom (DoF) of the suspending chains are also considered. The full linear and nonlinear equations are utilized in the computer simulations. A simple crane motion is simulated to compare the two models. The solution of the linear model shows a good agreement with the nonlinear one. Therefore we decided to use the linear model as an observer.

A new designed controller is proposed, which is based on a two-level hierarchical approach. Due to the domination of the payload on the motion of the crane system, a simple pendulum model was analyzed in the first level and a more complex model considering the chain vibration in the second level. A the first level, a D-subdivision method is applied for the simple mathematical pendulum model, which considers the time delay due to the use of a vision system to determine the stability regions expressed by gain parameters. In order to design the controller, the gain parameters associated with the state-space variables of the payload are selected within the stability region. In the second level, the extended model considering the vibration of the suspending chain. The previous gain parameters related to the relative displacement of the payload are maintained, and by using a pole placement method, the rest of the gain parameters associated with the chain vibration are determined. The rest of the unmeasured state variables are determined by the observer in every time step. The positioning of the trolley is prescribed by four parameters, relative distance, velocity, acceleration, and deceleration.

A position measurement system of the payload using an image processing system to detect without contact the swinging of the payload. In the proposed technique, the camera will capture

images in a real-time environment, and they will proceed to detect an object with a specified color by using the open-source computer vision library OpenCV with Visual Studio C++. The results will be sent to a Programmable logic controller via an Arduino Uno.

The computer simulations show that the designed controller using the two-level hierarchical approach effectively transfers the payload from the starting point until the target position without residual oscillations and without excessive overshooting. The proposed controller provides good active damping, the vibration of the payload is suppressed effectively within a short time.

The effectiveness of the proposed controller is also verified experimentally on a laboratory test bench. The designed controller displayed good results in the computer simulations. The robustness of the designed controller was tested for different gain parameters and sampling times. The designed anti-swing controller with the tracking controller effectively reduces payload oscillations in a reasonable time, and its performance is comparable to the results of a theoretical model. The presented method is competitive with the existing methods.

FUTURE WORK

The designed controller could be improved by adding a cut-off function for the acceleration of the trolley to make the swinging of the payload at the beginning of the transfer minimum. It can also be optimized to work with different cable lengths and different load sizes. The model can be extended to work with three dimensional.

The vision system can be improved to include smart functions, where it can detect obstacles, distance and height and change the trajectory to reach the target position successfully.

ACKNOWLEDGEMENTS

This thesis is a long term job, a challenge that we give to ourselves. But above all it is a great story of relationships, meetings, and friendships. This research project could not have been completed without the rich collaboration that I have been able to have with many people as well as without the invaluable help and unfailing support they have been able to give me. I want to show them my gratitude through these few lines.

It will be very difficult for me to thank everyone because it is due to the help of many people that I was able to complete this thesis.

I would like to warmly thank Dr. Tamás Szabó, Associate Professor, institute head of department, for the confidence he has shown in me by accepting the scientific direction of my work. I am grateful to him for allowing me throughout this work to benefit from his great competence, his intellectual rigor, his dynamism, and his sure effectiveness that I will never forget. Be assured of my attachment and my deep gratitude.

I would like to thanks József Lénárt, Assistant Professor at the University of Miskolc, for always giving me advice and help their support during my research.

I would like to thanks László Rónai, Senior Lecturer at the University of Miskolc, for always giving me advices and help their support during my research.

I am also thankful to all the staff of the Robert Bosch Department of Mechatronics.

I would like to thank Prof. Dr. Gabriella Vadászné Bognár, the head of the István Sályi Doctoral School of Mechanical Sciences; also Prof. Dr. Miklós Tisza, the previous head of the doctoral school, he was my professor during the 1st semester who taught me (Fundamentals of Materials Science).

I would like to express my gratitude to all my respected teachers of the University of Miskolc who taught me different subjects during the first two years of the program.

I also thank my teachers at the University of Hassan II and The University of Rabat, who gave me a good foundation that helped me and still.

Last but not least, I would like to thank my parents, my sisters, and my wife, and the rest of my family for their non-stop support during my studies and preparation of my Ph.D. dissertation.

I want to thank Stipendium Hungaricum Scholarship for this chance by covering the academic courses fee and the living costs and for their cooperation to receive admission at the University of Miskolc to complete my Ph.D. study in Hungary.

The publications were carried out as part of the EFOP-3.6.1-16-2016-00011 “Younger and Renewing University – Innovative Knowledge City – institutional development of the University of Miskolc aiming at intelligent specialization” project implemented in the framework of the Széchenyi 2020 program. The realization of this project is supported by the European Union, co-financed by the European Social Fund.

DEDICATION

To:

My parents,

My sisters and wife,

And for everyone who was on my side.

REFERENCES

[1] C. Pelletier: *Appareils de levage - Généralités*, Techniques de l'Ingénieur, 2000.

[2] C. Pelletier: *Ponts roulants et portiques*, Techniques de l'Ingénieur, 2000.

[3] S. Kang, E. Miranda: *Toward fully automated robotic crane for construction erection*, Conference: Proceedings of The Future of the AEC Industry, Las Vegas, 2005.

[4] E. Fadier: Automatisation et sécurité, Techniques de l'Ingénieur, 2009.

[5] https:..www.inrs.fr.media.html?refINRS=ED%206105.

[6] E. M. Abd-el-arahmen, A. H. Nayef, Z. N. Messoud: *Dynamics and control of cranes: a review*, Journal of Vibration and Control, Vol. 9, No. 2, pp. 863-908, 2003.

[7] O. Sawodnya, H. Aschemannb, S. Lahresc: *An automated gantry crane as a large workspace robot*, Control Engineering Practice, Vol. 10, No. 12, pp. 1323–1338, 2002.

[8] N. Zrnić, Z. Petković, S. Bošnjak: *Automation of ship-to-shore container cranes: a review of state of- the-Art*, FME Transactions, Vol. 33, No. 3, pp. 111-121, 2005.

[9] A. Bhatia: *Overview of Electric Overhead Traveling (EOT) Cranes*, CreateSpace, NY. 2014. (ISBN1502515938)

[10] P. Vaha, A. Pieska, E. Timonen: *Robotization of an offshore crane*, Proceedings of the 19th ISIR International Symposium, pp. 637–648. 1988.

[11] R. J. Henry, Z. N Masoud, A. H. Nayfeh, D. T. Mook: *Cargo pendulation reduction on ship-mounted cranes via boom-luff angle actuation*. Journal of Vibration and Control, Vol. 7, No. 8, pp. 1253-1264, 2001.

[12] Z. N Masoud, A. H. Nayfeh, A. Al-Mousa: *Delayed position-feedback controller for the reduction of payload pendulations of rotary cranes*. Journal of Vibration and Control, Vol. 9, No. 1-2, pp. 257-277, 2003.

[13] R. D. Robinett, G. G. Parker, J. Feddma, C. R. Dohrmann, J. Petterson: *Sway control method and system for rotary crane*, USA Patent No. 5908122, 1999.

[14] B. H. Karnopp, F. F. Fisher, B. O. Yoon: *A strategy for moving a mass from one point to another*, Journal of the Franklin Institute, Vol. 329, No. 5, pp. 881-892, 1992.

[15] C. L. Teo, C. J. Ong, M. Xu: *Pulse input sequences for residual vibration reduction*, Journal of Sound and Vibration, Vol. 211, No. 2, pp. 157–177, 1998.

[16] W. E. Singhose, L. J. Porter, W. Seering: *Input shaped of a planar gantry crane with hoisting*, Proceedings of the 1997 American Control Conference, Vol. 1, USA, pp. 97–100. DOI: 10.1109.ACC.1997.611762

[17] C. F. Cutforth, L. Y. Pao: *Adaptive input shaping for maneuvering fexible structures*, Automatica, Vol. 40, No. 1, pp. 685-693, 2004.

[18] O. T. C. Matthew: *A discrete-time approach to impulse-based adaptive input shaping for motion control without residual vibration*, Automatica, Vol. 47, No. 11, pp. 2504-2510, 2011.

[19] J. Vaughan, D. Kim, W. Singhose: Control of tower cranes with double-pendulum payload dynamics, IEEE Transactions on Control Systems Technology, Vol. 18, No. 6, pp. 1345-1358, 2010. DOI: 10.1109/TCST.2010.2040178.

[20] T. Vyhlídal, V. Kucera, M. Hromčík: *Zero vibration shapers with distributed delays of various types*, 52nd IEEE Conference on Decision and Control, Florence, pp. 940-945, 2013. DOI: 10.1109/CDC.2013.6760003.

[21] S. Rhim, W. J. Book: *Noise effect on adaptive command shaping methods for flexible manipulator control*, IEEE Transactions on Control Systems Technology, Vol. 9, No. 1, pp. 84-92, 2001, DOI: 10.1109/87.896749.

[22] S. Rhim, W. J. Book: *Adaptive time-delay command shaping filter for flexible manipulator control*, IEEE/ASME Transactions on Mechatronics, Vol. 9, No. 4, pp. 619-626, 2004. DOI: 10.1109/TMECH.2004.839046.

[23] W. Singhose, L. Porter, M. Kenison, E. Kriikku: Effects of hoisting on the input shaping control of gantry cranes, Control Engineering Practice, Vol. 8, pp. 1159–1165, 2000.

[24] J. Stergiopoulos, A. Tzes: *An Adaptive Input Shaping Technique for the Suppression of Payload Swing in Three-Dimensional Overhead Cranes with Hoisting Mechanism*, 2007 IEEE Conference on Emerging Technologies and Factory Automation (EFTA 2007), Patras, pp. 565-568, 2007. DOI: 10.1109/EFTA.2007.4416819.

[25] W. Xu, B. Liu, J. Chu, X. Zhou: *An anti-swing and positioning controller for overhead cranes based on multi-sliding mode method*, Advanced Materials Research, Vol. 468–471, pp. 328–334, 2012. DOI: 10.4028.www.scientific.net.AMR.468-471.328

[26] M. Z. M. Tumari, L. Shabudin, M. A. Zawawi, L.H. A. Shah: *Active sway control of a gantry crane using hybrid input shaping and PID control schemes*, IOP Conference Series: Materials Science and Engineering, Malaysia, Vol. 50, No. 1, pp. 1-10, 2013.

[27] Jeslin Thalapil: *Input shaping for sway control in gantry cranes*, IOSR Journal of Mechanical and Civil Engineering, , Vol. 1, pp. 36–46, 2012. DOI: 10.9790.1684-0123646

[28] X. Xie, J. Huang, Z. Liang: *Vibration reduction for flexible systems by command smoothing*, Mechanical Systems and Signal Processing, Vol. 39, No. 1-2, pp. 461–470, 2013.

[29] D. Blackburn, J. Lawrence, J. Danielson, W. Singhose, T. Kamoi, A. Taura: *Radial-motion assisted command shapers for nonlinear tower crane rotational slewing*, Control Engineering Practice, Vol. 18, No. 5, pp. 523–531, 2010.

[30] J. Lawrence, W. Singhose, *Command shaping slewing motions for tower cranes*, Journal of Vibration and Acoustics, Vol. 132, No. 1, pp. 11, 2010.

[31] R. E. Samin, Z. Mohamed, J. Jalani, R. Ghazali: *Input shaping techniques for anti-sway control of a 3-DOF rotary crane system*, 2013 1st International Conference on Artificial Intelligence, Modelling and Simulation, Kota Kinabalu, pp. 184–189, 2014. Doi: 10.1109.AIMS.2013.36.

[32] W. Singhose, J. Lawrence, K. Sorensen, D. Kim: *Applications and educational uses of crane oscillation control*, FME Transactions, Vol. 34, No. 5, pp. 175–183, 2006.

[33] K. T. Hong, C. Do Huh, K. S. Hong: *Command shaping control for limiting the transients way angle of crane systems*, International Journal of Control Automation and Systems, Vol. 1, No. 1, pp. 43–53, 2003.

[34] Z. N. Masoud, M. F. Daqaq: *A graphical design of an input-shaping controller for quay-side container cranes with large hoisting: theory and experiments*, Jordan Journal of Mechanical and Industrial Engineering, Vol. 1, pp. 57–67, 2007.

[35] M. A. Ahmad, R. M. T. Raja Ismail, M. S. Ramli, N. Hambali: *Analysis of IIR filter with NCTF-PI control for sway and trajectory motion of a DPTOC System*, 2010 International Conference on Electronic Devices, Systems and Applications, Malaysia, pp. 54–58, 2010. DOI: 10.1109.ICEDSA.2010.5503102

[36] M. A. Ahmad, F. R. Misran, M. S. Ramli, R. M. T. Raja Ismail: *Experimental investigations of low pass filter techniques for sway control of a gantry crane system*, 2010 2nd International Conference on Electronic Computer Technology, Kuala Lumpur, Malaysia, pp. 1–4, 2010. DOI: 10.1109.ICECTECH.2010.5480005

[37] G. Glossiotis, I. Antoniadis: *Payload sway suppression in rotary cranes by digital filtering of the commanded inputs*, Proceedings of the Institution of Mechanical Engineers Part K Journal of Multi-body Dynamics, Vol. 217, No. 2, pp. 99–109, 2003.

[38] J. Huang, X. Xie, Z. Liang: *Control of bridge cranes with distributed-mass payload dynamics*, IEEE.ASME Transactions on Mechatronics, Vol. 20, No. 1, pp. 481–486, 2015.

[39] X. Xie, J. Huang, Z. Liang: *Vibration reduction for flexible systemsby command smoothing*, Mechanical Systems and Signal Processing, Vol. 39, No. 1-2, pp. 461–470, 2013.

[40] K. A. Alghanim, K. A. Alhazza, Z. N. Masoud: *Discrete-time command profile for simultaneous travel and hoist maneuvers of overhead*, Journal of Sound and Vibration, Vol. 345, pp. 47–57, 2015.

[41] H. Saeidi, M. Naraghi, A. A. Raie: *A neural network self tuner based on input shapers behavior for anti sway system of gantry cranes*, Journal of Vibration and Control. Vol. 19, No. 13, pp. 1936–1949, 2013.

[42] H. M. Omar: *Control of gantry and tower cranes*, PhD Thesis, Virginia Polytechnic Institute and State University, USA, 2003.

[43] A. A. Al-mousa: *Control of rotary cranes using fuzzy logic and time-delayed position feedback control*, Master Thesis, Virginia Polytechnic Institute and State University, USA, 2000.

[44] Z. Wu, X. Xia, B. Zhu: *Model predictive control for improving operational efficiency of overhead cranes*, Nonlinear Dynamics, Vol. 79, pp. 2639–2657, 2015.

[45] D. Jolevski, O. Bego: *Model predictive control of gantry.bridge crane with anti-sway algorithm*, Journal of Mechanical Science and Technology, Vol. 29, No. 2, pp. 827–834, 2015.

[46] B. Kapernick, K. Graichen: *Model predictive control of an overhead crane using constraint substitution*, 2013 American Control Conference, Washington, USA, pp. 3973–3978, 2013. DOI: 10.1109.ACC.2013.6580447

[47] A. Khatamianfar, A. V. Savkin: *A new tracking control approach for 3D overhead crane systems using model predictive control*, 2014 European Control Conference (ECC), Strasbourg, France, pp. 796–801, 2014. DOI: 10.1109.ECC.2014.6862298

[48] M. Vukov, W. Van, L. Boris, H. Hans, J. Ferreau, J. Swevers, M. Diehl: *Experimental validation of nonlinear MPC on an overhead crane using automatic code generation*, 2012

American Control Conference (ACC), Montreal, Canada, pp. 6264–6269, 2012. DOI: 10.1109.ACC.2012.6315390

[49] Q. C. Nguyen, H. Ngo, W. Kim: *Nonlinear adaptive control of a 3D overhead crane*, 2015 15th International Conference on Control, Automation and Systems (ICCAS), Busan, Korea, pp. 41–47, 2015. DOI: 10.1109.ICCAS.2015.7364876

[50] H. C. Cho, K. S. Lee: *Adaptive control and stability analysis of nonlinear crane systems with perturbation*, Journal of Mechanical Science and Technology, Vol. 22, No. 6, pp. 1091–1098, 2008.

[51] Y. Fang, B. Ma, P. Wang, X. Zhang: *A motion planning-based adaptive control method for an underactuated crane system*, in IEEE Transactions on Control Systems Technology, Vol. 20, No. 1, pp. 241-248, 2012. DOI: 10.1109.TCST.2011.2107910

[52] S. Ning, F. Yongchun, C. He: *Adaptive control of underactuated crane systems subject to bridge length limitation and parametric uncertainties*, Proceedings of the 33rd Chinese Control Conference, Nanjing, China, pp. 3568–3573, 2014. DOI: 10.1109.ChiCC.2014.6895532

[53] N. Sun, Y. Fang, H. Chen: *Adaptive antiswing control for cranes in the presence of rail length constraints and uncertainties*, Nonlinear Dynamics, Vol. 81, No. 1-2, pp. 41–51, 2015.

[54] N. Sun, Y. Fang, C. He, B. He: *Adaptive nonlinear crane control with load hoisting.lowering and unknown parameters: design and experiments*, in IEEE.ASME Transactions on Mechatronics, Vol. 20, No. 5, pp. 2107-2119, 2015. DOI: 10.1109.TMECH.2014.2364308

[55] L. Lee, P. Huang, Y. Shih, T. Chiang, C. Chang: *Parallel neural network combined with sliding mode control in overhead crane control system*, Journal of Vibration and Control, Vol. 20, No. 5, pp. 749–760, 2014.

[56] K. Nakazono and K. Ohnishi and H. Kinjo and T. Yamamoto:*Vibration control of load for rotary crane system using neural network with GA-based training*, Artificial Life and Robotics, Vol. 13, pp. 98-101, 2008. DOI: 10.1007/s10015-008-0586-5

[57] S. C. Duong, E. Uezato, H. Kinjo, T. Yamamoto: *A hybrid evolutionary algorithm for recurrent neural network control of a three-dimensional tower crane*, Automation in Construction, Vol. 23, pp.55-63, 2012.

[58] P. Li, Z. Li, Y. Yang: *The Application Research of Ant Colony Optimization Algorithm for Intelligent Control on Special Crane*, 2012 Second International Conference on

Instrumentation, Measurement, Computer, Communication and Control, Harbin, 2012, pp. 999-1004, doi: 10.1109/IMCCC.2012.238.

[59] G. Hilhorst, G. Pipeleers, W. Michiels, R. C. L. F. Oliveira, P. L. D. Peres, J. Swevers, : *Reduced-order $\mathcal{H}2/\mathcal{H}\infty$ control of discrete-time LPV systems with experimental validation on an overhead crane test setup*, 2015 American Control Conference (ACC), Chicago, IL, 2015, pp. 125-130, doi: 10.1109/ACC.2015.7170723.

[60] F. Panuncio, W. Yu, X. Li: *Stable neural PID anti-swing control for an overhead crane*, 2013 IEEE International Symposium on Intelligent Control (ISIC), Hyderabad, pp. 53-58, 2013. DOI: 10.1109.ISIC.2013.6658616

[61] B. Yang, B. Xiong: *Application of LQR techniques to the anti-sway controller of overhead Crane*, Advanced Materials Research, Vol. 139-141, pp. 1933-1936, 2010.

[62] P. Pannil, K. Smerpitak, V. La-orlao, T. Trisuwannawat: *Load swing control of an overhead crane*, ICCAS 2010, Gyeonggi-do, pp. 1926-1929, 2010. DOI: 10.1109.ICCAS.2010.5670121

[63] M. P. Spathopoulos, D. Fragopoulos: *Pendulation control of an offshore crane*, International Journal of Control, Vol. 77, No. 7, pp. 654–670, 2004.

[64] M. P. Spathopoulos, D. Fragopoulos: *Control design of a crane for offshore lifting operations*, in: Isidori, A., Lamnabhi-Lagarrigue, F., Respondek, W., (Eds.), Nonlinear Control in the Year 2000 Volume 2, Springer, pp. 469-486, 2001. DOI: 10.1007.BFb0110321

[65] R. C. Roman, R. E. Precup, E. M. Petriu: *Hybrid data-driven fuzzy active disturbance rejection control for tower crane systems*, European Journal of Control, 2020. DOI: 10.1016.j.ejcon.2020.08.001

[66] N. Sun, Y. Fang, H. Chen: *A new antiswing control method for underactuated cranes with unmodeled uncertainties: theoretical design and hardware experiments*, IEEE Transactions on Industrial Electronics, Vol. 62, No. 2, pp. 453–465, 2015. DOI: 10.1109.TIE.2014.2327569

[67] S. Zhang, X. He, H. Zhu, Q. Chen, Y. Feng: *Partially saturated coupled-dissipation control for underactuated overhead cranes*, Mechanical Systems and Signal Processing, Vol. 136, 2020. DOI: 10.1016.j.ymssp.2019.106449

[68] A. Aksjonov, V. Vodovozov, E. Petlenkov: *Three-dimensional crane modelling and control using euler-lagrange state-space approach and anti-swing fuzzy logic*, Electrical Control and Communication Engineering, Vol. 9, No. 1, pp. 5-13, 2015.

[69] K. S. Hong, B. J. Park, M. H. Lee: *Two stage control for container cranes*, JSME International Journal. Series C, Vol. 43, No. 2, pp. 273-282, 2000.

[70] N. Minorsky: *Self-excited oscillations in dynamical systems possesing retarded actions*, Journal of Applied Mechanics, Vol. 9, pp. 65-71, 1942.

[71] N. Minorsky: *Nonlinear Oscillations*, D. Van Nostrand Company, London, 714 p, 1962.

[72] M. Wahyudi, J. Jamaludin, R. Muhida, M. Salami: *Control Strategy for Automatic Gantry Crane Systems*: A Practical and Intelligent Approach. International Journal of Advanced Robotic Systems, Vol. 4, No. 4, pp. 447-456, 2007.

[73] N. B. Almutairi, M. Zribi: *Fuzzy Controllers for a Gantry Crane System with Experimental Verifications*, Mathematical Problems in Engineering, No. 14, pp 1-17, 2016. DOI: 10.1155.2016.1965923

[74] A. Aksjonov, V. Vodovozov, E. Petlenkov: *Three-Dimensional Crane Modelling and Control Using Euler-Lagrange State-Space Approach and Anti-Swing Fuzzy Logic*, Electrical Control and Communication Engineering, Vol. 9, No. 1, pp. 5-13, 2015.

[75] K. S. Hong, B. J. Park, M. H. Lee: *Two stage control for container cranes*, JSME International Journal, Series C, Vol. 43, No, 2, pp. 273-282, 2000.

[76] O. Itoh, H. Migita, J. Itoh, Y. Irie: *Application of fuzzy control to automatic crane operation*, Proceedings of IECON '93 - 19th Annual Conference of IEEE Industrial Electronics, Maui, Vol. 1, pp. 161-164, 1993. DOI: 10.1109.IECON.1993.339088

[77] L. H. Lin, P. D. Lawrence, R. Hall: *Stereo vision based swing angle sensor for mining rope shovel*, Proceedings of the RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, pp. 1714-1721, 2010. DOI: 10.1109.IROS.2010.5650457

[78] K. Yong-Seok, H. Yoshihara, N. Fujioka, H. Kasahara, S. Hyungbo, S. Seung-Ki: *A new vision-sensorless anti-sway control system for container cranes*, 38th IAS Annual Meeting on Conference Record of the Industry Applications Conference, Salt Lake City, Vol. 1, pp. 262-269, 2003. DOI: 10.1109.IAS.2003.1257512

[79] Y. Yoshida, K. Tsuzuki: *Visual tracking and control of a moving overhead crane load*, 9th IEEE International Workshop on Advanced Motion Control, Istanbul, pp. 630-635, 2006. DOI: 10.1109.AMC.2006.1631733

[80] H. Sano, K. Ohishi, T. Kaneko, H. Mine: *Anti-sway crane control based on dual state observer with sensor-delay correction*, 2010 11th IEEE International Workshop on Advanced Motion Control (AMC), Nagaoka, Niigata, pp. 679-684, 2010. DOI: 10.1109.AMC.2010.5464050

[81] D. Bachrathy, J. M. Reith, G. Stepan: *Algorithm for robust stability of selayed multi-degree-of-freedom systems*, in: Insperger, T., Ersal, T., Orosz, G., (Eds.), Time Delay Systems Theory, Numerics, and Experiments, Springer International Publishing, pp. 141-154, 2017. DOI: 10.1007.978-3-319-53426-8_10

[82] K. J. Bathe: *Finite Element Procedures*, Prentice Hall, Pearson Education Inc, 2006.

[83] L. Meirovich: *Fundamentals of Vibrations*, McGraw-Hill, New York, 2001.

[84] https:..opencv.org.about.

[85] https:..store.arduino.cc.arduino-uno-rev3

[86] https:..docs.opencv.org.master.dc.dbb.tutorial_py_calibration.html

LIST OF PUBLICATIONS RELATED TO THE TOPIC OF THE RESEARCH FIELD

[S1] Marouane Hmoumen: *A Review of Optical Character Recognition System, Design of Machine and Structures,* Vol. 7, No. 2 (2017), pp. 5-12.

[S2] Marouane Hmoumen and Tamás Szabó: *Linear and nonlinear dynamical analysis of a crane model*, Pollack Periodica, Vol. 15, No. 2 (2020), pp. 82-93.

[S3] Marouane Hmoumen and Tamás Szabó*: Crane payload position measurement vision based system,* Doctoral Forum 2019.

[S4] Marouane Hmoumen and Tamás Szabó: *Controlling of trolley position and payload swinging of an overhead crane,* (Its acceptance is in progress)

LIST OF FIGURES

LIST OF TABLES