JEAS
Journal of Economics and Administrative Sciences
E-ISSN: 2148-1792 P- ISSN: 1302-2024
Volume 5, Supplement Issue 1
www.jeasweb.org

Integrated Modeling of Mechatronic Systems Integrated Mechatronic Modeling of Servo-Mechanical Systems

Nejat Darijani
Master of Mechatronics Engineering, Department of Electrical Engineering Majlesi Branch Islamic Azad University Esfahan Iran

ABSTRACT

The ultimate purpose of this study is to integrate mechatronic modeling of servo-mechanical systems. Matlab / Simulink software was used for mechatronic modeling. In this study, a foot system was designed for the robot that provides the robot with maneuverability. Evaluating foot movement in the field of robotics is very important because the foot crosses rough terrain. Therefore, designing a foot system for robots capable of moving on uneven surfaces is critical. In this study, the CARSI code is used for the design process of the foot mechatronic system. Also, to analyze the data, a library called SimMechanics has been used, which simulates the dynamics of a rigid body based on a three-dimensional graphical model. In this software, three types of implementation are supported by the RTW toolbox: Real-Time Windows Target, XPC target, and Real-Time Embedded Target. In the first two implementations, real-time system metrics were used. Therefore, the functionalities or space is considered in sufficient detail in real-time. In this study, two simulation and experimental models with similar scales were used to evaluate foot movement. Module "ICP_i8438" is considered for integrated mechatronic modeling for mechanical foot systems. The results of this study reflect that the experimental and simulation results are very close to each other. Based on the results, it can be seen that the idea of integrated DFC design and fast implementation of CARSI, which are proposed in this paper, provide the conditions for robotic foot movement. The DFC model is highly desirable for designing the mechanical structure of a mechatronic foot system by fully exploring the physical properties. The design of the controller and the implementation of the considered control measures are faced with the least hardware limitations. Finally, the "ICP_i8438" module based on DFC codes assists the mechatronic system to satisfy low driving power and helps to direct the system easily. In addition, the CARSI approach realized structural design, controller design, and system execution simultaneously in the design environment and significantly reduced mechatronic foot system development time.

Keywords: Integrated mechatronics, Mechanical servo systems, Matlab / Simulink software, Robotic system

1-Introduction

The mechatronic systems usually show a high complexity due to the strong interconnection of various engineering disciplines such as mechanics, electronics, and computers (Ali, 2017). This complexity is due to the high number of couplings in different levels of elements and auxiliary components. The problem of a design engineer in his daily work is that these couplings must be considered in the first stage of the design process. By shortening the production increase cycle, the design managers continuously try to identify tools for producing better products in shorter periods (Akram, 2011).

Therefore, the domain of mechatronics includes high speed, high precision, and high efficiency. The issue in the mechatronic approach is that it needs a systemic perspective: the system interactions are important, modeling is needed, and feedback control systems can be unstable. The mechatronic design concepts include direct drive mechanisms, simple mechanics, basic mechanics, system complexity, controls precision and speed, performance and reliability of electronics, and performance of microcomputers (Klimchuk, 2013). Mechatronic designs, with the initiation of designing and continuing the production, optimize a combination of the available technologies to produce precise and high-quality products and systems in time, with features the customer intends. The actual advantages of the mechatronic design for the industry are shorter development cycles, lower costs, and an increase in quality, reliability, and performance (Zhan, 2006).

In addition, to evaluate the concepts produced in the design process, without producing and testing each, the mechatronics engineer must be proficient in modeling, analyzing, and controlling dynamic systems and understand the key points in hardware execution. Thus, as Figure 1 shows, the basic characteristic of a mechatronics engineer and the key to success in the mechatronic system is the balance between the two sets of skills (Shan et al., 2019).

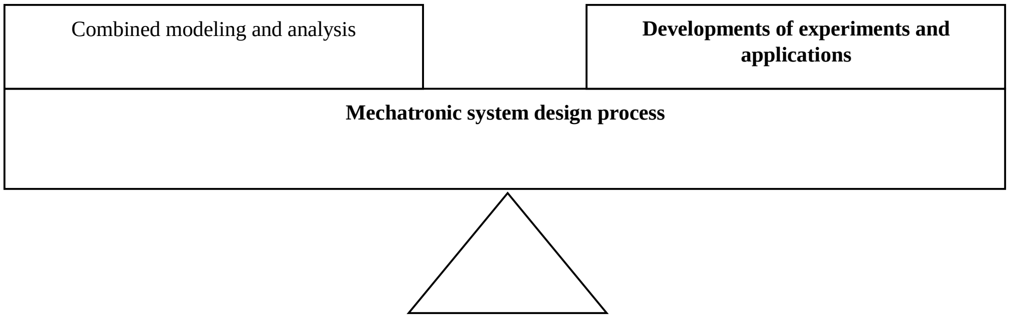


Figure 1: Mechatronic design process balance

1-1-Integrated Modelling and Analysis of Dynamic Mechatronic Systems:

During the mechatronic system's design, it is important to evaluate the changes in the mechanical structure and controller simultaneously (Kia et al., 2020). Although a proper controller makes the construction of a mechatronic system cheaper and feasible, a mechanical system with a bad design can never show a good performance with the addition of a complex controller. Therefore, it is important to have, in the initial phases of designing, a proper choice based on the mechanical properties needed for the achievement of good controlled system performance. On the other hand, the knowledge of controller capabilities to compensate for the mechanical issues allows the construction of cheaper mechanical structures. It requires the availability of a simple integrated model in the initial phases of designing, which show the limiting factors of the mechatronic system performance (Heming et al., 2019).

As a result, to help the mechanical structure and mechatronic system modeling controller, simultaneously, the mechatronic system design methods should be integrated. In this regard, some integrated design strategies based on some mechatronic systems have been initially suggested for some fields such as aerospace, robotics, and manufacturing systems. However, the dynamic models derived from the integrated models above are usually highly disciplined. An important position in mechanical structure and control modeling with an integrated design approach is the difficulty of each area (Kristin et al., 2015).

Therefore, for complex multi-body mechatronic systems, the graphical modeling program is useful for the automatic formulation of motion equations from a high-rank description. Among the computerized modeling methods, the symbolic methods allow the creation of motion equations in a symbolic format, while the numerical methods of motion equations create complex numerical procedures. The symbolic format has portability and efficiency advantages and provides an interesting vision in equations analysis structure. Yet, the numerical methods can confront a more general category of problems and are especially suitable for dynamic modeling a flexible mechanism with complex typology in a systematic method. After this clarification, let us determine the modeling prerequisites in the design process that are directly related to the objectives of the current study.

1-2-Empirical Validation and Design Hardware Implementation:

In an industrial process, the controller design, including formulation of models with logical precision from the power plant under control, can be done by control rules design based on the derived models, and simulation of control rules by the use of available simulation tools such as MATLAB/Simulink, while implementation is done with the modification of control rules designed with native code of target systems, which usually have an architecture based on microchips or PCs with embedded analog and digital connectors. The controllers can be designed in continuous, discontinuous, or combined time, while the implementation is done more in the field of discontinuous time since most of the modern controllers are implemented in digital machines. The huge difference between the design and implementation of control applications is due to different concepts in the field of control engineering and computer science. Therefore, changing the controller design into implementation suggests the probability of errors and unreliable behaviors. In some cases, these errors cannot be identified by the precise implementation experiments, so they would lead to a system failure which can ultimately cause a serious disaster (Kleanthis et al., 2019).

In addition, the conventional controller design duty requires the choice of controller strategies, structures, and parameters' values. Before implementation, the engineers should be tested by the use of the real-time -time values of the workshop or during the implementation of the prototype with the physical and measured inputs and the produced outputs. This stage is necessary for empirical validation of the model's simplifications and other assumptions made during controller design. On the other hand, the rapid simulation creates the best conditions for modification of this capability performance. However, the reverse state occurs when the workshop's model replaces the real-time workshop while the controller may be completely implemented. This approach of simulation of rapid controller prototyping is called RCP (Ali, 2017). For this technique, the engineers have actuators, sensors, and other physical components which are linked the rapid simulation. In addition, the RCP techniques allow the controlling strategies implementation and validation during the development process to enable the operators to work in the same environment of demand analysis to the controller design and implementation stage (Kenway et al., 2020).

Regarding these two sets of skills, mechatronic design in a computerized design environment can be generally categorized into three stages as the design problem, perception of the mechatronic system through the demand analysis, and production of the initial solution. Through the conceptual design, solution correction, and finalization by precise multi-disciplinary design. In the computerized support for engineering design, there is little support for the two first stages of the design process, initially due to the complexity and various demands of these designing activities during the three stages. The final stage in the design process is currently the main zone, which enjoys logical computer support, and can be used as an aid for the engineer designers to improve their designs or products. This stage of computer support can be divided into component modeling, adaptation and measurement of components, and behavior simulation and comparison for learner decision-making (Kleanthis, 2019). This division facilitates further evaluation of any design support activity.

What should be noted about an ordinary problem with many of the current computerized modeling methods is that they are highly domain-dependent. In the mechatronic system design processes, which include structure designing, controller designing, and implementation in three areas, the interactions between several areas such as integrated design, rapid prototyping, and animation technology are also considered (Figure 2).

Therefore, the mechatronic system design engineers must be trained for use in different practical domains to be skillful in using all technologies dependent on the skill domain. This duty itself is very challenging. As a result, to solve the problems of mechatronic systems, the mechatronics engineers always use a dynamic equation that includes all parameters in the structural and control domains. Unfortunately, one of the most important problems in using equation-based mechatronic modeling is the tabular data of modeling which should be analyzed during the process. Due to this huge mass of data and the numerical algorithm that should be used, this modeling and simulation method is usually very slow and prone to errors. In addition, this method requires a high knowledge of numerical solving methods and programming principles (Akram, 2011).

Based on these reasons, for integrated design and easy simulation of a mechatronic system for different fields, the graphic environment named Computer-Aided Rapid System Integration (CARSI) technology will be developed in the current study to achieve the structure design controller design and implementation (Rosenberg).

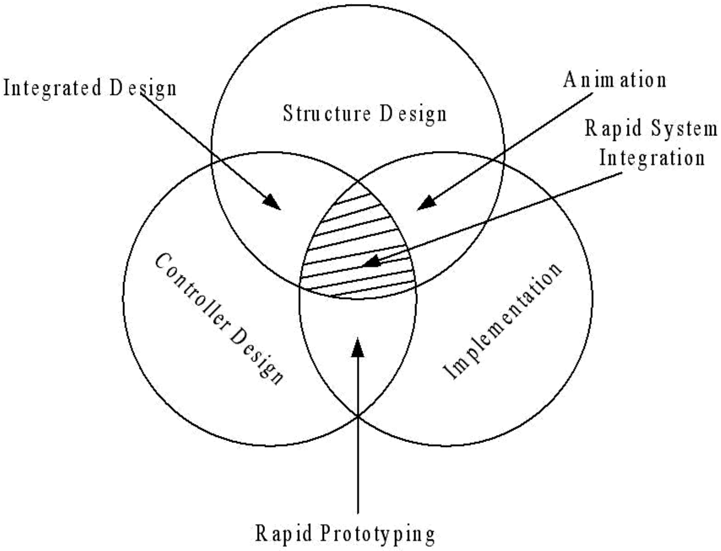


Figure 2: Mechatronic system designing skill

In this study, the next section explains the integrated design strategy by the use of consecutive, repetitive, and simultaneous methods. In section 3, the DFC integrated designing method for the development of a mechatronic system with foot movement was used. Following section 4 and in section 5, CARSI technology is provided to put together design, simulation, and implementation in an environment.

2- Integrated Design Strategy:

A large complex optimization problem is divided into a hierarchy of smaller secondary optimization problems with a multi-level division approach. This hierarchy can be considered as levels of increasing details. At the high level, the secondary problem is expressed per local values and limitations, which have only a little effect on the whole system. Any secondary problem uses local design variables to reduce violation of limitations, which is exclusive to that secondary problem. Each level of the optimization problem is multi-purpose, which is identified with the vector of objective functions, limitations, and design variables. Therefore, considering the structure and two-level problem of control for a mechatronic system, the multi-level analysis method can be written as follows (Ptolemaeus, 2014).

Equation 1:

min.YNi(XN),i=1,....,nNs.t.gNk(XN)≤0,k=1,....,ncN∑i=1NDVN∂YRj∂XNiΔXNi≤ε2j,j= 1,......,nRXNiL≤XNi≤XNiU,i=1,.......,NDVNXRjL≤XRj*+∑i=1NDVN∂XRj*∂XNiΔXNi≤XRjU,j=1,.. .,NDVR

in which YN and YR are the objective functions for structure and control levels, respectively. gNA and gR are the related limitation vectors. XNand XR is the related design variable vectors. ε2ji is tolerance in the transformation of j objective of control level during the optimization in structure level. L and U are the lower and upper bounds of design vectors. ∂YRj*/∂XNiand ∂XRj*/∂XNi are the optimized sensitivity parameters of the objective vector of control level and design variable vectors, based on the construct level design variables. nNand nR is the number of objective functions for each level. ncN is the number of limitations for the structure level. NDVN and NDVR are the number of design variables for the structure and control levels (Paredis, 2019).

Accordingly, the control process is written as below:

Equation 2:

minYRj(XN*,XR),j=1,...,nRs.t.gRk(XN*,XR)≤0,k=1,...,ncRXRiL≤XRi≤XRiU,i=1,.. .,NDVR

n which XN is the vector of optimization design variable from the structure level, and it should be constant in the control level during the optimization.

Sub (equation 1) and (equation 2); the integrated design methodology can be divided into three strategies as consecutive, repetitive, and simultaneous:

In the consecutive strategy, usually, the mechanical structure is designed first (equivalent 1). It is then installed with off-the-shelf electric motors and electronic actuators. Finally, a controller is designed and set for the available physical system until the target is archived (equivalent 2). Therefore, it is called DTC (Design Time Control) strategy. In this method, the construct is assumed to be constant, and it is not changeable by the omission of dynamic and control considerations. As a result, this approach leads to a system with a non-optimal dynamic performance (Fantuzzi, 2011).

In this regard, to improve the system performance, the repetitive strategy is discussed. For this method, the construct design is initially done based on the loading considerations (equivalent 1). The size and mass of components related to the mission are estimated, and a structure is designed that maintains links between the intended components. In the next stage, a controller is designed for the constant structure to achieve the required dynamic performance (equivalent 2). The control design must also guarantee the strength and stability of the closed-loop. If the nominal system does not provide a proper performance, the design process must return to the structural line for modification (equivalent 1). After modification, the structure parameters return to the control line for a redesign (equivalent 2). This repetitive process continues until a satisfactory agreement between the mission and control requirements is found. Now assume that you want to simplify the formula of (equivalent 1) and (equivalent 2) as much as possible. Probably, the problem can be simplified with the assumption that all the objective functions and limitations are convex in both sub-spaces of structure design and the controller. In other words, it can probably be assumed that when the XN structure design variable is constant, all the objective functions and limitations in the above problem will be convex and vice versa. However, this assumption is not a sufficient guarantee for the convexity of the system-level optimization problem (Kleanthis, 2019). Therefore, to achieve the optimization problem at the system level, the simultaneous design strategy should be considered.

As seen in (equivalent 1) and (equivalent 2), regarding the combined problem of structure and controller optimization for the mechatronic system, the system's level is mostly non-convex, even if the secondary sub-problems of structure and control optimization are convex (individual design problem for (equivalent 1) and (equivalent 2)). The main reason is that the static optimization problem and the easy changes are involved in the repetitive design process. Therefore, use the special values of the closed-loop, Design For Control (DFC), and convex integrated design [8] to improve the structure and control problem simultaneously (Drath, 2008).

Therefore, as Figure 3 shows, comparing the three above strategies, the system performance is even increased during consecutive, repetitive, and simultaneous strategies.

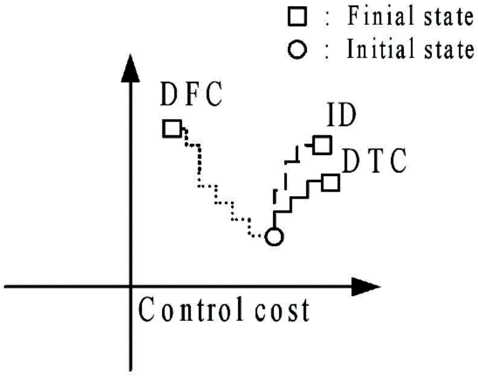


Figure 3: Cost control in the repetitive process

3- Legged Motion Mechatronics System Design:

Most of the robots are equipped with wheels. A wheel can be easily guided and controlled. A leg system (pelvic) allows the ability for the robot to maneuver on, and it is easy to build. However, one of the main drawbacks of a wheel is the limitation it creates on the ground where it can be successfully traversed. Thus, research on legged locomotion is important since the feet can cross uneven ground. That is why the creation of a legged system motion mechanism has become the main objective in robotics [13-15]. In this regard, in the current study, the CARSI will be used in the rapid mechatronic system design process of a legged system, and its chart is shown in Figure 4.

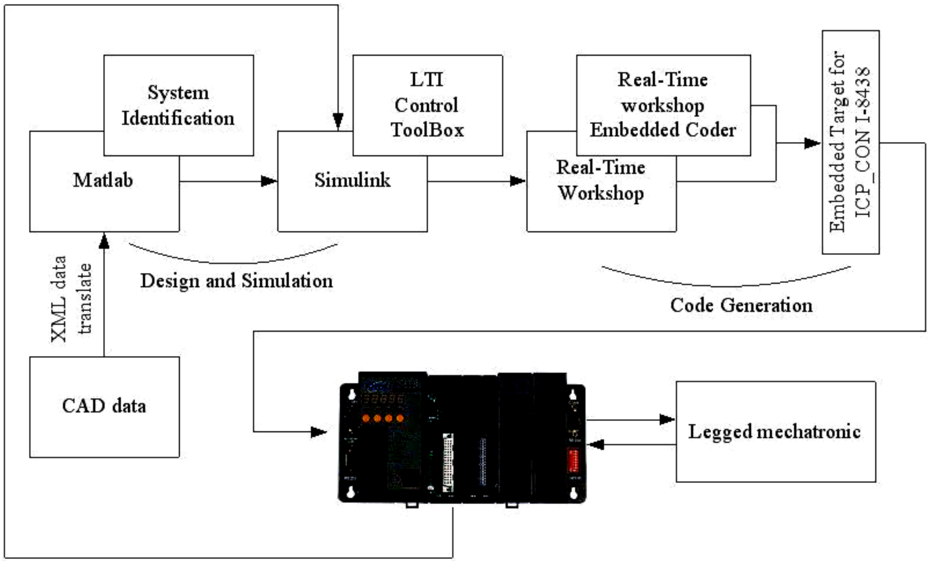


Figure 4: Chart of a legged motion (pelvic) mechatronic system design

3-1-Legged (Pelvic) System Structure:

The basic considerations for legged system (pelvic) design are as follows: the legged system (pelvic) must provide an almost direct path for the leg (pelvic) system with the body. It should have a simple mechanical design, and if required, it should have the least number of DOFs to ensure reliability. Therefore, the basic principle in this study is the creation of a walking machine through linkage with symmetrical coupling curves to combine the function of a four-bar link and the leg (pelvic) system of the netograph in the structure of a leg (pelvic) system.

Based on the embedded legged (pelvic) system (Figure 5), an embedded path P is first designed by a 4-bar link, and then it is magnified with a scale of n(B0E=nB0D) to obtain gait profile G. Based on the design specifications (Table 1), the link parameters of the four bars are obtained. In addition, all the design processes are based on the below assumptions:

There are no transitional losses between the input effect and the legged (pelvic) system of this mechanism.

The ground reaction force on the legged system is constant.

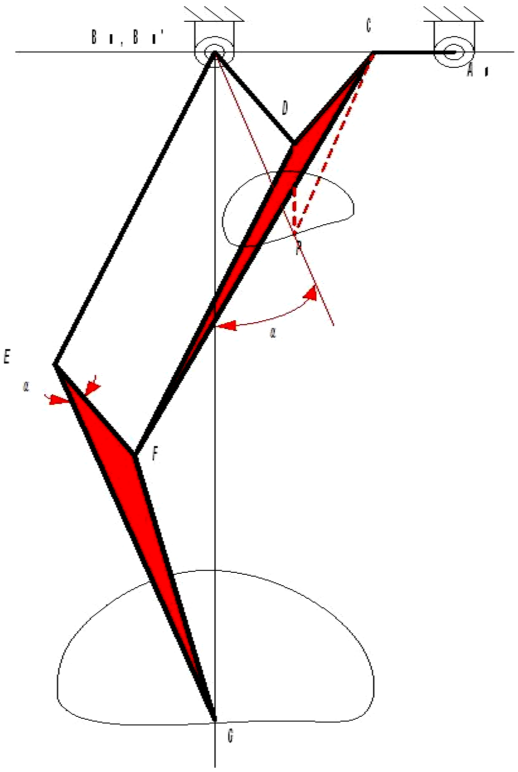


Figure 5: Legged system (pelvic) structure

3-2-Optimal Multi-Variable Design for Gait Profile:

As was discussed, the gait profile can be designed by the use of an embedded four-bar linkage and magnify it by a netograph legged system (pelvic) to achieve the target. In addition, to reduce legged (pelvic) system size (or minimize the n scale ratio) and obtain the increased height of the walking path, the design objective function is formulated as (equivalent 3) (Fantuzzi, 2011).

Equation 3:

$I1=min(\beta(ls)-1+\gamma(lh)-1)$

Equation 4:

$\varphi\geq2cos-1cos\beta1sin\beta2$

$12cm\leq A0B0¯\leq14cm2cm\leq A0C¯\leq(2CD¯-A0B0¯)cm\pi<\mu+\varphi'<2\pi45\circ\leq\mu\leq135\circ E5$

In which:

β,γ is the weighted coefficient, β=1 ‹γ=0.2ls: is the length of the four embedded bars link: path height of legged system link with the four embedded bars, and α is the angle of deviation.

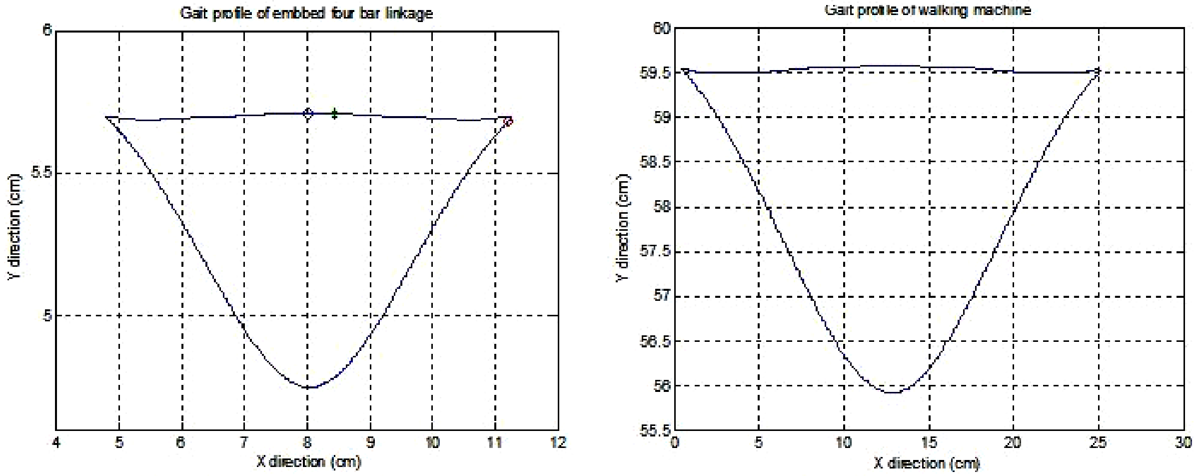
equation 5:

$\angle CDF=\phi+\alpha=\phi'$

Table 1 lists the optimal results based on (Equivalent 3) and those constraints. In addition, Figure 6 shows the gait profile of the 6-bar machine and embedded four-bar linkage profile.

Table 1: Integrated design results:

Components	DTC	DFC	Var. %
Structure parameters			
A0B0¯)cm(	12.8	12.8	-
A0C¯)cm(	2.6	2.6	-
B0D¯=EF¯)cm(	8	8	-
B0E¯=DF¯)cm(	30.8	30.8	-
mass A0C¯)Kg(	0.050	0.080	60
mass B0D¯)Kg(	0.035	0.07	100
mass B0E¯)Kg(	0.134	0.134	-
mass ΔCDF)Kg(	0.368	0.215	-41.5
mass ΔEFG)Kg(	0.316	0.177	-44.0
r2 (cm) / δ2)degree(	1.3 / 0	0 / 0	- / -
r3 (cm) / δ3)degree(	16.0 / 35.6	14.6 / 38.0	-8.6 / 6.3
r4 (cm) /δ4)degree(	4 / 0	0 / 0	- / -
r5 (cm) /δ5)degree(	12.5 / 42.0	11.6 / 36.6	-7.5 / -13
r6 (cm) / δ6)degree(	15.4 / 0	15.4 /0	- / -
α)degree(	51.2	51.2	-
N	3.8	3.8	-
ϕ′)degree(	128.8	128.8	-
Controller parameters			
Kp	4.3	3.5	-18.6
Chi	4000	4800	20
Kpp	150	180	20
maximum |τ|(N-m) without acc/dec (0.5 steps per second	0.14	0.12	-10
maximum |τ|(N-m) without acc/dec (2 phase per second	0.5	0.2	-60



(a) Embedded (with skew angle) (b) End effect

Figure 6: gait profile

3-3-Controller Design:

When the kinematic design of the walking machine was completed, the controller design was considered. Therefore, for merging and modeling a walking machine mechatronic system in the design process, the Lagrange equation was formulated as (equivalent 4), which is used to extract all parameters in the controller design process.

Equation 7:

ddt∂K∂θ˙2−∂K∂θ2+∂P∂θ2=τ

In which K is kinetic energy, P is potential energy, τ is the control torque, and θ2 is the input linkage angle. Figure 7 shows the precise parameters of a dynamic system for a walking machine. Therefore, the initial parameters K and P can be expressed with (equivalent 5) and (equivalent 6), and control torque τ is reformulated as (equivalent 7) (Kia, 2020).

Equation 8:

K=∑i=26[12mi(Vix2+Viy2)+12Jiθ˙i2]

Equation 9:

P=(m2r2sin(θ2+δ2)+m3(L2sinθ2+r3sin(θ3+δ3))+m4r4sin(θ4+δ4)+m5(L2sinθ2+L3′sin(θ3+ρ3)+r5sin(θ5+δ5)+m6r6sin (θ6+δ6))g

In which mi is the mass of each linkage. Vix and Viy are the speed in directions x and y, respectively. The length is increased from the central mass to the reference position. δi is the length of each linkage. The central mass angle for each linkage.

gis;L3′=(L32+L62−2L3L6cosφ′)0.5.

On the other hand, using simple controllers such as the PD/PID is prevalent in industrial manipulators and servo systems, which work based on the position loop control. In this process, to improve the positioning performance for speed and position simultaneously, the IP controller is used for the speed loop, and the P controller is used for the position loop. The control power equation τ can be formulated in (Equivalent 7) (Kristin, 2015).

Equation 10:

$$\tau(t)=Ki\int(e\theta Kpp-\omega m)dt-Kp\omega m$$

In which, θ, Kp, Kpp, and Ki are positioning error, increase in position loop ratio, increase in speed loop ratio, and speed loop merging ratio, respectively. Following the Integral Time Absolute Error (ITAE), the design objective for the controller was written as (Equivalent 8), in which η and ς are weighted factors, and $\eta=1$ and $\varsigma=0.1$. The results are listed in Table 1.

Equation 11:

$$I=\min(\eta\int t|e\theta(t)|dt+\varsigma|\tau(t)|)s.t \quad \tau(t)\leq 5Nme\theta\geq 0$$

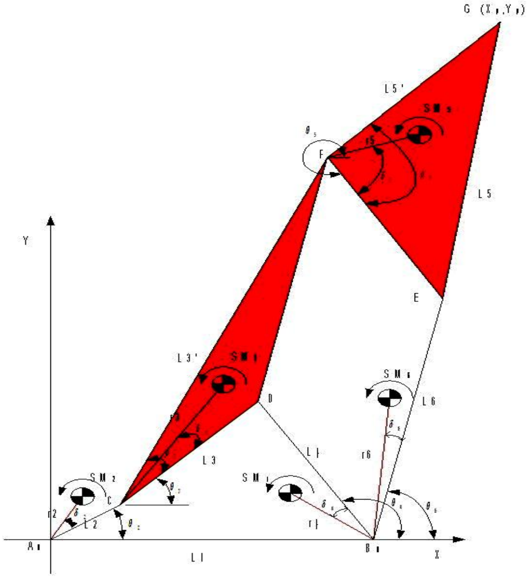


Figure 7: linkage dynamic model for legged (pelvic) system

4 -System Optimization Using Design For Control (DFC) Approach:

As Figure 8 shows the repetitive process DFC [5] [6] [23], if the system performance is not satisfactory, the design process returns to the structural area. The structural correction process exits the used domain limitations, and the general conditions of the dynamic system are replaced with main conditions, and it is transferred to the control area for obtaining a new controller solution. Therefore, not only DFC uses a simultaneous (parallel) integrated design process to obtain the system performance, but also it is used to increase control requirements for the easy control system in the design approach (Kleanthis, 2019).

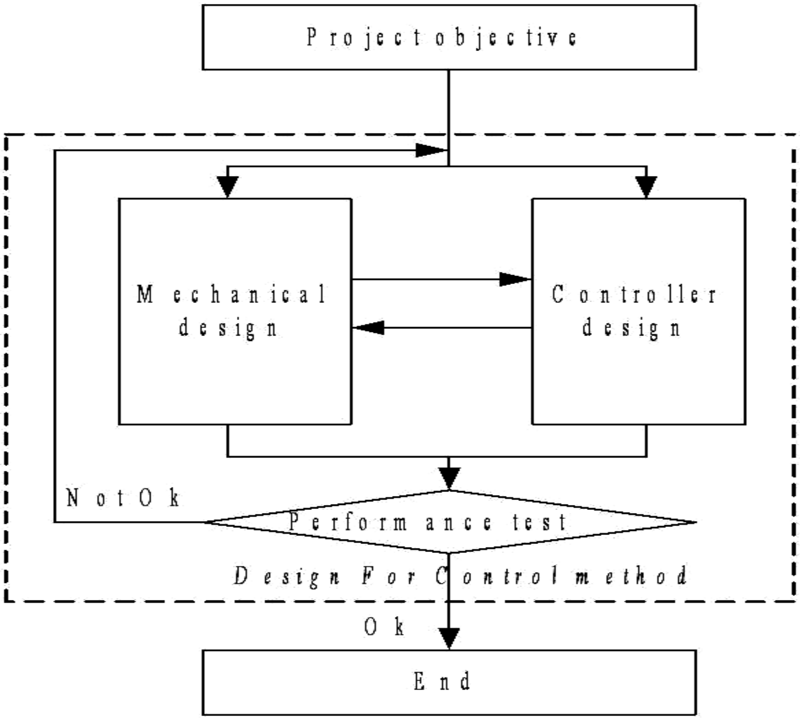


Figure 8: Integrated design of the mechatronic system using DFC

After using the DFC concept and Lagrange equation, the objective of the design process is the reduction of the kinetic and potential energy, which is interactional at the beginning of the design process. Therefore, correction of legged linkage system parameters for the walking machine should be considered, and the system performance is simultaneous based on the structural results of (equivalent) and (equivalent 6) in adjusting the controller parameters (equivalent 7). Thus, following (equivalent 5) and (equivalent 6), two methods can be used for the improvement of the system's performance, i.e., variable input speed [18], which reduces the kinetic energy for (equivalent 5), and redistribution of mass [5-6], both interpretations reduce (equivalent 5) and (equivalent 6), simultaneously. Therefore, the "complete force balancing" method based on the redistribution of mass was used to increase the system performance. So, the initial objective in this interactional process can be formulated as (equivalent 9).

Based on this objective, when (equivalent 9) equals zero, the legged system gait mechatronic dynamic equation can be reformulated as (equivalent 10). From this equation, the dynamic behavior of legged system mechatronic can also be reduced to a simple equation, i.e., the control power is considered only as close-to-constant kinetic and potential energy in this interactional control process. As was mentioned before, the main idea of the DFC approach is saving the efforts for control design and rapid performance improvement by the provision of the simple dynamic model through logical, mechanical design. As a result, the key parameters of the internal moment, δi, ri, and mi are improved (Table 1). In addition, Table 1 also lists the optimal control achievements.

Equation 12;

$I=\min\partial P\partial\theta 2$

$ddt\partial K\partial\dot{\theta}2-\partial K\partial\theta 2=\tau$

4-1- Merging Multi-Domain Graphical Model:

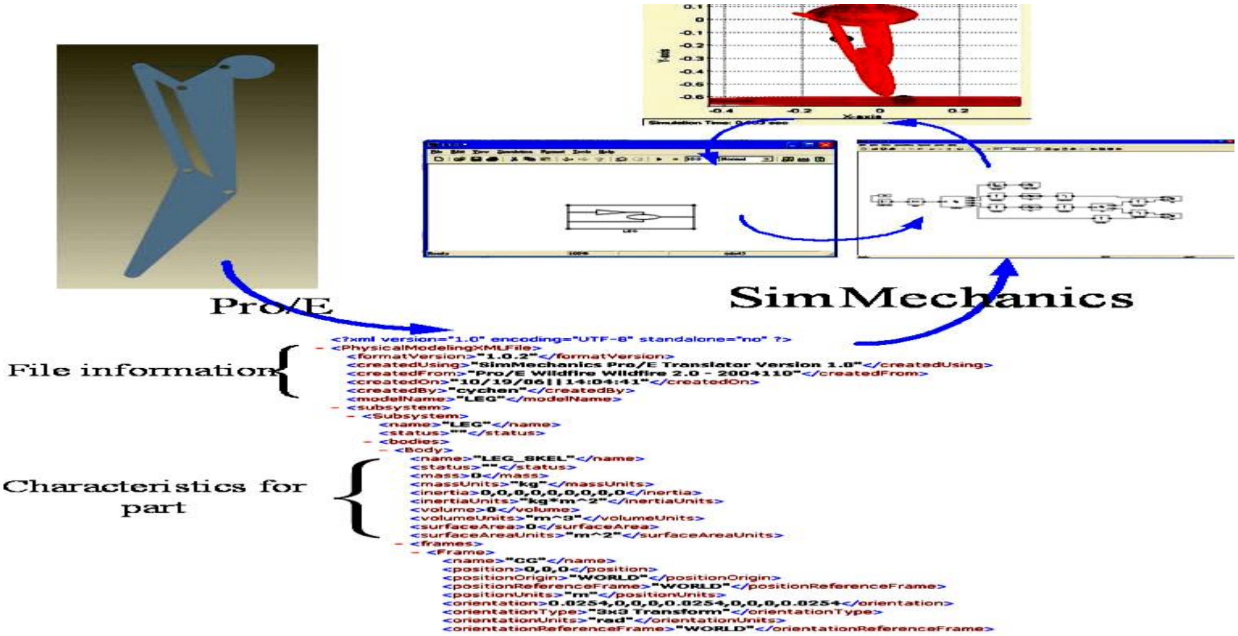
With rapid advances in computer science over the past 20 years, engineering software such as Pro/Engineer, Solidworks, Ansys, and Matlab have been extensively used in the areas of structure and control. Therefore, file format standards such as the Initial Graphics Exchange Specifications (IGES), ISO-10303, STEP, and DFX were developed to deal with the issue of incompatibility of different CAD/CAM systems. This standard allows the efficient and precise exchange of product definition data in almost all CAD/CAM systems.

Since any computer-aided engineering program package uses a unique method to describe geometry both mathematically and structurally, when translating the data from a systemic data format to another, some of the data is always lost. The middle file formats are also limited in the cases they can describe and can be interpreted differently by the ending and receiving systems. When transferring the data between the systems, identification of what should be translated is important. In addition, the translation of the middle file is always focused on one engineering field. Thus, in the mechatronic system, the formats or parameters of the middle file should be considered partial to be accepted by each domain, i.e., different systemic fields are possible in a time model, in which the language used for model description can be developed, and includes several standard libraries for different fields. It helps operators because they can use modeling tools they know for different purposes.

XML (Extensible Markup Language) is a general-purpose markup language recommended by the World Wide Web Consortium (W3C) that supports a wide range of applications [19]. XML language and its dialects can be designed by anybody and can be processed using the proper software. XML has also been designed in a way to be logically readable for humans.

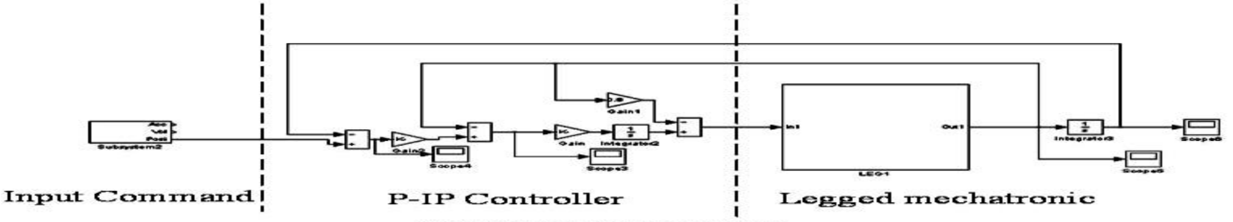
According to the kinematic synthesis of legged (pelvic) system linkage (Table 1), the 3-D graphical model for leg mechatronic was first designed using Pro/E (Figure 9a). In addition, based on (equivalent 4), (equivalent 5), and (equivalent 6), the linkage parameters such as mass, length, position, the center of gravity, unit, volume, and limitation (or linkage type) were extracted from these CAD data. Further in this stage, the XML-based graphical model was obtained with parameters needed for control. Based on this model and the parameters (Table 1), the embedded controller was also created using this graphical model (Figure 9b). The technology was used for construct design, controller design, and implementation in the same design environment.

According to the CARSI method, Figures 10a and 10b present the results obtained from DTC and DFC methods. Comparing the results obtained from these methods, the mechanism performance was greatly improved after the redistribution of mass. The maximum control torque in low and high speeds was decreased from 10%, ±0.14 to ±0.12N-m, and 10%, ±0.5 to ±0.2N-m. Based on these analytical results, when the center of mass and walking machine mass is equal to zero, the potential energy would almost have no effects, even when the speed is changed. On the contrary, based on the DTC result, with the change in gait speed, the control power is increased.



```
<?xml version="1.0" encoding="UTF-8" standalone="no" ?>
- <PhysicalModelingXMLFile>
    <formatVersion>"1.0.2"</formatVersion>
    <createdUsing>"SimMechanics Pro/E Translator Version 1.0"</createdUsing>
    <createdFrom>"Pro/E Wildfire Wildfire 2.0 - 2004110"</createdFrom>
    <createdOn>"10/19/06||14:04:41"</createdOn>
    <createdBy>"cychen"</createdBy>
    <modelName>"LEG"</modelName>
  - <subsystem>
    - <Subsystem>
        <name>"LEG"</name>
        <status>""</status>
      - <bodies>
        - <Body>
            <name>"LEG_SKEL"</name>
            <status>""</status>
            <mass>0</mass>
            <massUnits>"kg"</massUnits>
            <inertia>0,0,0,0,0,0,0,0,0</inertia>
            <inertiaUnits>"kg*m^2"</inertiaUnits>
            <volume>0</volume>
            <volumeUnits>"m^3"</volumeUnits>
            <surfaceArea>0</surfaceArea>
            <surfaceAreaUnits>"m^2"</surfaceAreaUnits>
          - <frames>
            - <Frame>
                <name>"CG"</name>
                <position>0,0,0</position>
                <positionOrigin>"WORLD"</positionOrigin>
                <positionReferenceFrame>"WORLD"</positionReferenceFrame>
                <positionUnits>"m"</positionUnits>
                <orientation>0.0254,0,0,0,0.0254,0,0,0,0.0254</orientation>
                <orientationType>"3x3 Transform"</orientationType>
                <orientationUnits>"rad"</orientationUnits>
                <orientationReferenceFrame>"WORLD"</orientationReferenceFrame>
```

(a) Multi-domain transforms



(b) Simulation model of legged mechatronic.

Figure 9: Multi-domain graphical model

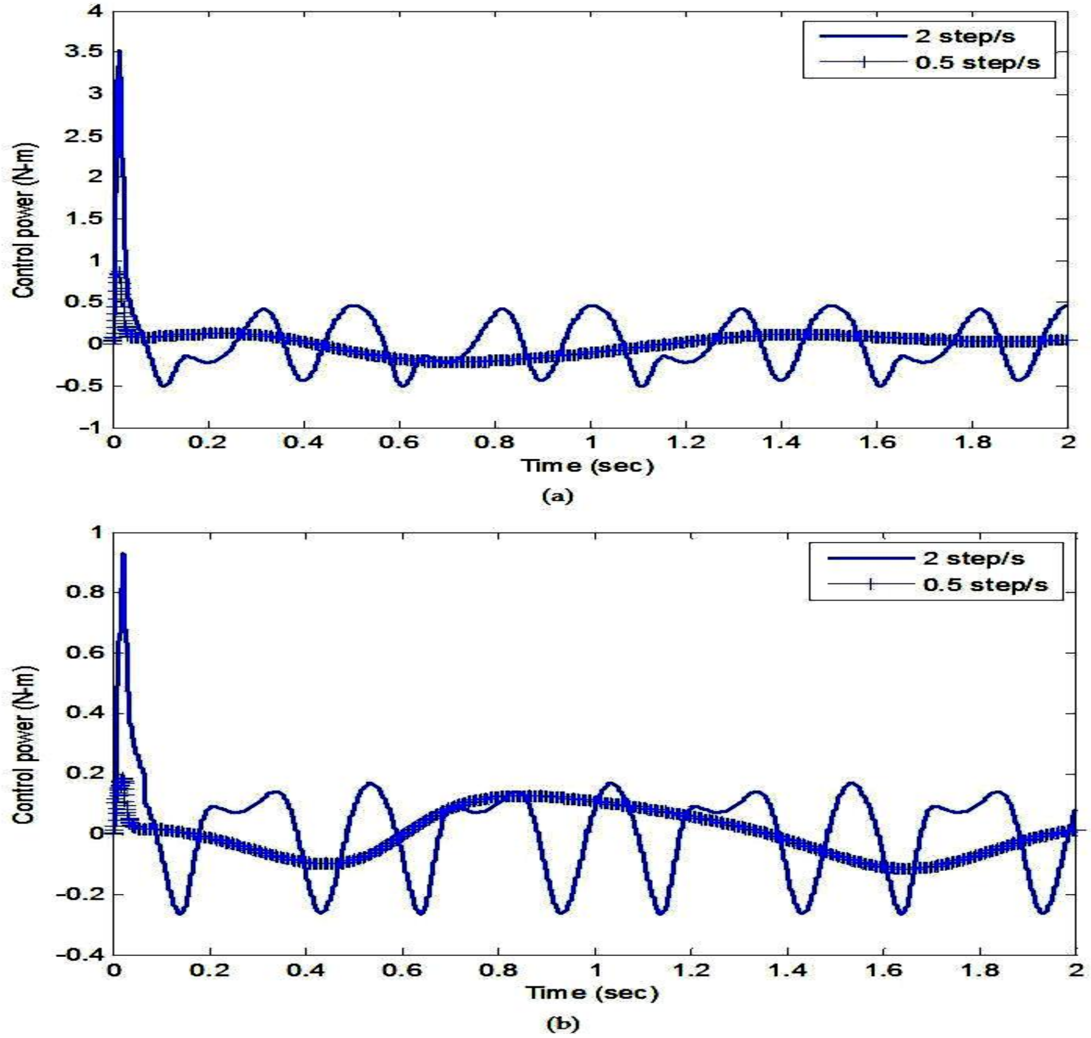


Figure 10: Control power for DTC and DFC methods

5- Rapid Control Modelling:

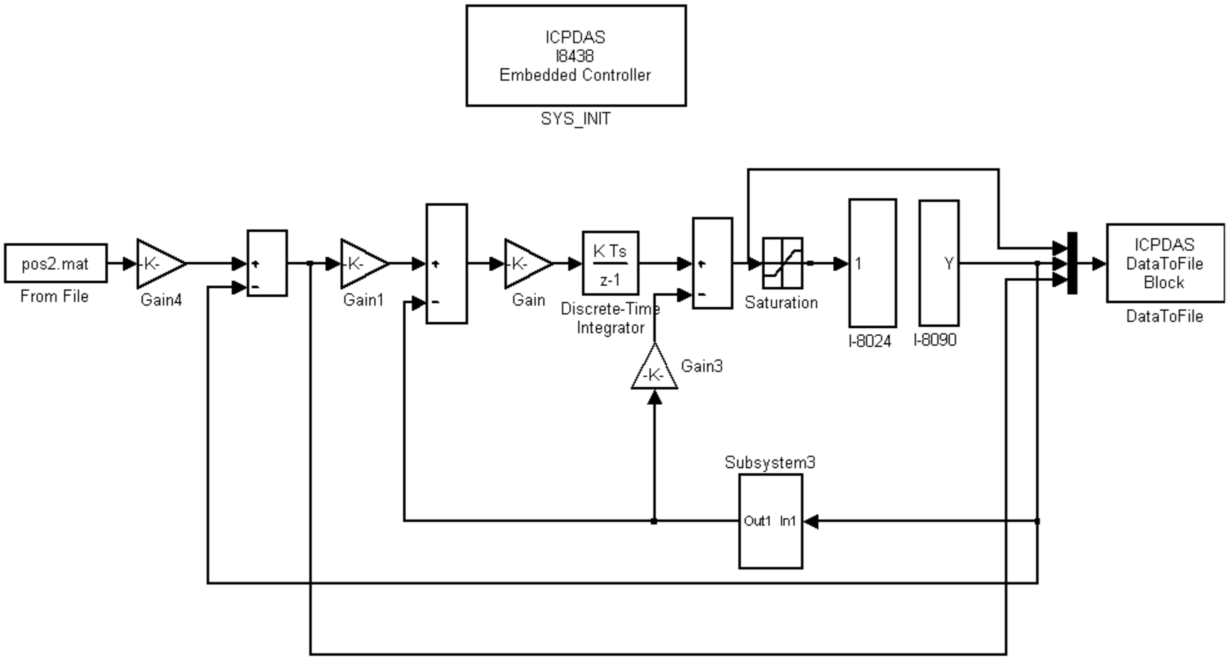
As the control systems get more complicated with the development of control algorithms and controller designing techniques, the manual interpretation and designing of a control system using differential equations or numerical formulae get more difficult and time-consuming. In addition, different user-friendly graphs and interfaces, as well as complex calculations, are necessary. Moreover, due to repetitive operations when designing a control system is required, normal manual programming is not an easy task, and it would be inefficient when facing it.

Instead of ordinary low-level programming languages, graphical model-based programming is increasingly being used for rapid simulation and Hardware-in-the-Loop (HIL) to obtain rapid prototyping from different electrical and mechanical systems. Compared to ordinary manual low-level programming, the most important feature of the advanced control programs is the function the program codes automatically create through some user-friendly graphical modules so that the time needed for system development is reduced.

As was mentioned, Matlab/Simulink software is a design and simulation tool that is most used in the field of control. This software allows the users to easily create models for dynamic systems through connecting the blocks from the given libraries, and also, it includes a library named SimMechanics, which simulates rigid body dynamics by the use of the 3-D graphical model. Some of Matlab/Simulink blocks execute the linear systems provided as transition functions or realization of the state space in the continuous and discontinuous time. When is a simulation is completed, the RTW (Real-Time Workshop) creates code C from the model without the need for programming knowledge. Therefore, rapid controller prototyping techniques facilitate the implementation and validation of control strategies during the development process. The operators can work in the same space of analysis of structure demands to design and implementation of controllers. Based on this software, three types of execution are supported by the RTW toolbox, which are Real-Time Windows target, XPS target, and Real-Time Embedded Target. For the first two techniques, the rapid target devices are based on the PC. Therefore, the real-time space or performance must be considered in detail.

As mentioned before, Figure 11 shows module “ICP_i8438,” which is based on a microchip, providing the auxiliary modules such as scale output (I-8024) and coder feedback (I-8090). According to this model and legged (pelvic) gait mechatronic system (Figure 12a), the simulation and empirical results are presented in Figure 12b.

With these results, the constant friction torque is considered as 0.3 N.m. The empirical and simulation results are very close. However, an evident issue with these results is that the dynamic friction during the acceleration phase has been ignored. With reinterpretation, merging a graphics-based model and an equation-based model for simulation of a mechatronic system can easily predict and correct the system model parameters to achieve the target for a real-time system.



6- Conclusion:

An integrated design concept (DFC) and rapid execution (CARSI) for a walking machine were suggested in this study. DFC was used for designing the mechanical structure of a legged mechatronic system with a complete exploration of general system physical characteristics while it considered controller design and controlling measures execution with the minimum hardware limitations. Again, the DFC not only helped the mechatronic system to fulfill low driving power but also facilitated the system control. In addition, the CARSI approach obtained simultaneous structural designing, controller designing, and system execution in the design environment so that the legged mechatronic system development time was reduced.

References

Al-Ali, A. R., Negm, M. M., & Kassas, M. (2017). A PLC based power factor controller for a 3-phase induction motor. In Industry Applications Conference, 2017. Conference Record of the 2017 IEEE (Vol. 2, pp. 1065-1072). IEEE.

Akram, Hossian. (2011)-352: Integrating Servomotor Concepts into Mechatronics Engineering Technology Curriculum Emphasizing High Speed Packaging Machinery

Merzouki, R., Davila, J. A., Fridman, L., & Cadiou, J. C. (2007). Backlash phenomenon observation and identification in electromechanical system. Control Engineering Practice, 15(4), 447-457.

Zhan, Y., & Makis, V. (2006). A robust diagnostic model for gearboxes subject to vibration monitoring. Journal of Sound and vibration, 290(3), 928-955.

Klimchuk, M. R., & Krasovec, S. A. (2013). Packaging design: Successful product branding from concept to shelf. John Wiley & Sons.

Barbieri, G., Fantuzzi, C., and Borsari, R. (2014). A Model-Based Design Methodology for the Development of Mechatronic Systems. Mechatronics Journal

Drath, R., Luder, A., Peschke, J., and Hundt, L. (2008). AutomationML - the glue for seamless automation engineering. In Emerging Technologies and Factory Automation (ETFA). Hamburg, Germany.

Fantuzzi, C., Bonf´e, M., Fanfoni, F., and Secchi, C. (2011). A Design Pattern for translating UML software models into IEC 61131-3 Programming Languages. In 18th World Congress of International Federation of Automation Control (IFAC).

Paredis, C.J., Bernard, Y., Burkhart, R.M., de Koning, H.P., Friedenthal, S., Fritzson, P., Rouquette, N.F., and Schamai, W. (2019). An Overview of the SysMLModelica Transformation Specification. In INCOSE International Symposium

Ptolemaeus, C. (ed.) (2014). System Design, Modeling, and Simulation using Ptolemy II. Ptolemy.org. http://ptolemy.org/books/Systems.

Rosenberg, D. and Mancarella, S. (2019). Embedded Systems Development using SysML. OMG: Systems Modeling Language

Shah, A.A., Kerzhner, A.A., Schaefer, D., and Paredis, C.J.J. (2019). Multi-View Modeling to Support Embedded Systems Engineering in SysML. In Graph Transformations and Model-Driven Engineering

Kia, S. H., Henao, H., & Capolino, G. A. (2020). Torsional vibration effects on induction machine current and torque signatures in gearbox-based electromechanical system. IEEE Transactions on Industrial Electronics, 56(11), 4689-4699.

Akram, Hossian. (2011)-352: Integrating Servomotor Concepts into Mechatronics Engineering Technology Curriculum Emphasizing High Speed Packaging Machinery