Műszaki Tudományos Közlemények vol. 17. (2022) 37–41.
DOI: Hungarian: https://doi.org/10.33895/mtk-2022.17.08
English: https://doi.org/10.33894/mtk-2022.17.08

THE SIMULATION OF A PID CONTROLLED ROBOTIC DRIVE BASED ON BOND GRAPH MODELLING

Gyula KORSOVECZKI,[1] Patrik PÁL,[2] Géza HUSI[3]

University of Debrecen, Mechatronics Department, Debrecen, Hungary

[1] *korsoveczki.gyula@gmail.com*

[2] *palpatrik5@gmail.com*

[3] *husigeza@eng.unideb.hu*

Abstract

During the study, the simulation of a robotic drive was realised using the 20-sim simulation software. The assumed robotic drive consists of an ideal permanent magnetic DC motor, a gear pair to realise the speed reduction, a torque transfer torsion shaft and a differential gear pair to change the direction and inertia. The drive is controlled by a speed control loop using a PID controller. In line with the purpose and result of the study, the simulation proved the effectiveness of Bond Graph-based modelling.

Keywords*: mechatronics, robotic drive, Bond Graph, block diagram, PID, 20-sim.*

1. Introduction

Today, engineering has attained an increasingly complex level. As the available technology expands, so does the complexity of engineering systems. The process itself is aimed at being able to provide increasingly more functions. As a result, the implementation of systems requires a more detailed design or modelling process. It is necessary to find modelling language which is not resource-intensive and provides the same response as the real physical system in cases of different signals.

2. Bond Graph modelling

2.1. The model of Multi-Domain systems

Every system has its description language. Systems that operate only in one physical domain are called Single Domain systems. Modern engineering focuses on complex combined systems. There is physical or informational connection between the contained Single Domain systems and the effect of variables is transferred from one system to the other. These systems are called Multi-domain systems. Mechatronic systems are typically Multi-domain systems **[1]**.

For Multi-Domain systems, the biggest challenge is modelling their operations. It is required to establish connections between different domains and variables to create a dynamic model. It is difficult to find connections between the description languages. The key is a physical variable that is present in all systems. This physical variable is energy **[1]**.

2.2. The principles of the Bond Graph

A Bond graph is a special visual and mathematical description language that uses energy as its basic variable. The modelling is an effective method for Multi-Domain systems, as energy is a physical variable that is present in all Single Domain systems. The method was developed by Henry Paynter in 1959. According to the method, the two-way flow of energy can be tracked through different elements of the language. As a result, it is possible to define the equations that describe each domain, as well as the connections between them **[1, 2]**.

2.3. The description language of the Bond Graph

The use of the Bond Graph is based on 2 power variables, effort and flow. The multiplication of these variables gives the transferred power, which is [1–3]:

$$P(t) = e(t) \cdot f(t) \tag{1}$$

Their role is defined and is not interchangeable. The Bond Graph is based on the Bonds, which represents the physical or informational relation between the elements. The elements can be [3–5]:

– passive one-port elements (*R, C, I*)
– active one-port elements (*Se, Sf*)
– two/three-port junctions (*TF*, *GY*, 0, 1)

Passive elements are used to store or dissipate energy. No new energy is produced only the existing energy is used. They can be *R* elements (e.g. damping, bearings, electrical resistance). Energy can be stored in the form of potential (*C* elements) or kinetic energy (*I* elements). Passive *C* elements are spring, shaft, capacitor and *I* elements are mass, inertia, circuit coil. The one-port active elements can add new power variables to the system. The power variable can be effort or flow, and accordingly, there is Source of effort (e.g. voltage generator, gravity force) and Source of flow (current generator, applied velocity.)The third large group consists of junctions such as basic two-port (Transformers *TF*, Gyrator *GY*) and three/more-port (0 and 1 junctions) junctions. Transformers (e.g. electrical transformator, rigid lever, gear pair) convert energy from one form to another while strictly keeping the role of power variables. Gyrators (e.g. ideal PMDC motor) reverse the roles of power variables. Another large group of junctions is the three/more-port junctions. The 0 junction (e.g. serial mechanical system or a parallel electrical circuit) is an effort junction that has constant effort value. In the case of the 1 junction (e.g. parallel mechanical system or the serial electric circuit) the flow is constant and the sum of the effort is zero [4, 5]. Causality is the most important aspect of Bond Graph-based system modelling. Causality determines the direction of effort and flow along with the Bonds since it relates to the cause and consequence relations in time. The causal stroke shows the direction of effort and flow. To correctly define causality, it is necessary to know the most important rules regarding different types of elements [6–8]. The basic Bond Graph elements can be seen in Figure 1.

3. Drive chain of industrial robots

Industrial robots are movable mechanisms that consist of joints (constraints) and segments and have at least 3 degrees of freedom. Their motion is

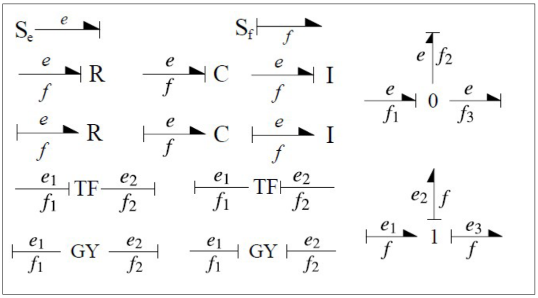

Figure 1. *The basic Bond Graph elements.* [4]

a constrained motion, which must be performed with exceptional precision since the repeatability of the robots is in the hundredths of millimetres. The segments are moved by the drive chain, with the following units [9]:

– energy supply or motor,
– drive unit,
– motion transforming unit,
– controlling unit.

Industrial robots can have pneumatic, hydraulic or electric drive chains. Pneumatic drives use compressed air. It is clean, explosion-proof with favourable power/weight ratio and low installation space. Disadvantages are poor signal transmission ability of compressed air, high noise levels, low positioning accuracy and difficulty in proportional control. Hydraulic drives use special liquid with higher forces. Advantages are the favourable power-to-weight ratio and the good signal transmission ability of the oil. Its disadvantages are explosion and fire hazards, noise problems, oil pollution and strong temperature dependency. The most commonly used solutions are electric drives. Their spread has been helped by the success of semiconductor technology. The requirement of low motor weight is important as the motors are part of the kinematic chain, so space and weight requirements are limited. An important criterion is the need for high torques, as industrial robots must be moved with the same dynamics throughout the load range. The motor needs to produce a wide speed range. The output typically does not have an adjustable gear ratio, so changing the speed ensures a wide range of motion. Stepper motors, DC or AC servo motors are used. The goal is to move low-speed robotic arms with low-weight but high-speed motors with high torque. Drives are used to achieve low output speed. The function of the drives is to fit the speed, fit the torque and convert motion forms. Gears (gear pairs, differential gear), drive pulley (maximum 1/60 ratio), planetary gears and harmonic drives are used. With harmonic drives the transmission of speed up to (1/200) is available. Ball screw spindle, belt drives, chain drives and crank mechanisms are also used for this purpose [9–11].

4. The modelled drive chain

4.1. The construction of the drive chain

In the project, a drive chain was modelled using Bond graph. Its essential element is an ideal DC motor. The motor has a R_a armature resistance, a L_a armature inductance, the U_a input voltage, and an U_m electric motor voltage. The equivalent circuit is a serial circuit that has i armature current. The ideal DC motor converts the electric domain into the rotary mechanical domain. The motor produces τ_1 torque and ω_1 angular velocity values. The motor output shaft has bearing with a b_1 damping constant. Since the motor is essentially capable of moving in a wide speed range, but axes require a stable high torque and low speed, the desired speed is achieved with a gear pair. The N_1 is the input, and the N_2 gear is the output, which is supported by b_2 bearings. The torque transmission is carried out by a k torsion shaft. The axial direction of the motor is perpendicular to the axis of the motion. This is achieved by an N_3/N_4 differential gear pair. The N_3 gear is the input, the N_4 gear is the output. The differential gear pair allows us to achieve 90° motion conversion. Bearings are also on both the input and output sides with b_3 and b_4 gains. The axis to be moved is represented by inertia in the system that has J value.

4.2. The Bond Graph model of the drive chain

It is important to make the correct choice of elements and variables. The serial circuit of the motor can be modelled with 1 junction, where constant flow (current) and variable effort (voltage) is present. The voltage source can be modelled as *Se* the armature resistance as an *R* element, and the armature inductance as *I* element. Since the motor is an ideal conversion between the electric and rotary mechanical domain, it can be modelled as a GY element with k_m motor constant ratio. The mechanical part of the motor output can also be modelled with 1 junction, which includes an *R* bearing element. The speed-reducing gear pair is modelled with the *TF* element. The modulus of the TF element is the ratio of N_1/N_2 gear pair. The output bearing of the N_2 axis is modelled with 1 junction, and the b_2 bearing modelled as an *R* element. The torsion shaft for torque transmission can be modelled as a *C* element and its value is k. This connection is responsible for the torque transmission, so it can be modelled with a 0 junction. The output bearing is modelled with a 1 junction through an R element with the b_3 gain. The differential gear pair can also be modelled as a *TF* element. Its modulus value is the N_3/N_4 ratio number. Finally, the output side is modelled with 1 junction, the bearing is an *R* element and the axis is represented by *J* inertia as an *I* element. The drive chain construction and its Bond Graph model can be seen in Figure 2.

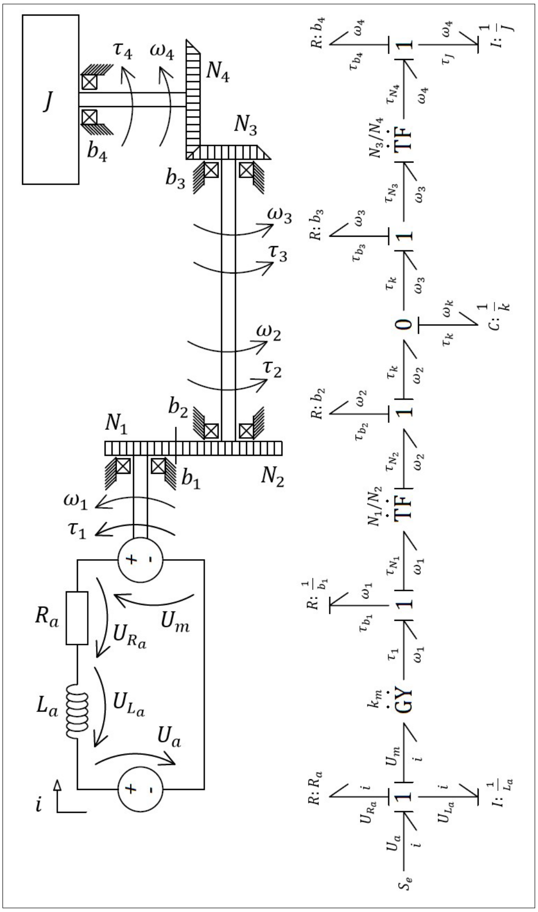

Figure 2. *The drive chain and its Bond Graph model.*

4.3. The system equations

The advantage of Bond Graph-based modelling is the possibility to define equations directly. The equations of the DC motor based on the Bond Graph are the following:

$$U_{L_a} = U_a - U_{R_a} - U_m \quad (2)$$

$$L_a \cdot \frac{di}{dt} = U_a - R_a \cdot i - k_m \cdot \omega_1 \tag{3}$$

$$\tau_{b_1} = \tau_1 - \tau_{N_1} \tag{4}$$

$$b_1 \cdot \omega_1 = k_m \cdot i - \frac{N_1}{N_2} \cdot \left(b_2 \cdot \omega_2 + \frac{1}{k} \cdot \int \omega_k dt\right) \tag{5}$$

The relations of effort and flow in the *GY* element with k_m ratio are the following:

$$U_m = k_m \cdot \omega_1 \tag{6}$$

$$\tau_m = k_m \cdot i \tag{7}$$

The equations of the *TF* element with $m_1 = N_1/N_2$ modulus are the following:

$$\tau_{N_1} = \frac{N_1}{N_2} \cdot \tau_{N_2} \tag{8}$$

$$\omega_2 = \frac{N_1}{N_2} \cdot \omega_1 \tag{9}$$

Based on these the equations, the k torsion shaft and the b_3 bearing have the following relations:

$$\tau_{N_2} = \tau_{b_2} + \tau_k \tag{10}$$

$$\frac{N_2}{N_1} \cdot \tau_{N_1} = b_2 \cdot \omega_2 + \frac{1}{k} \cdot \int \omega_k dt \tag{11}$$

$$\omega_k = \omega_2 - \omega_3 \tag{12}$$

$$k \cdot \frac{d\tau_k}{dt} = \frac{N_1}{N_2} \cdot \omega_1 - \frac{\tau_{b_3}}{b_3} \tag{13}$$

$$\tau_{N_3} = \tau_k + \tau_{b_3} \tag{14}$$

$$\frac{N_3}{N_4} \cdot \tau_4 = \frac{1}{k} \cdot \int \omega_k dt - b_3 \cdot \omega_3 \tag{15}$$

The differential gear pair represented by the *TF* element with N_3/N_4 modulus affects the drive by the following:

$$\tau_{N_4} = \frac{N_4}{N_3} \cdot \tau_{N_3} \tag{16}$$

$$\omega_3 = \frac{N_4}{N_3} \cdot \omega_4 \tag{17}$$

Finally, the equations of the J inertia and the b_4 bearing are the following:

$$\tau_J = \tau_{N_4} - \tau_{b_4} \tag{18}$$

$$J \cdot \frac{d\omega_4}{dt} = \frac{N_4}{N_3} \cdot \tau_3 - b_4 \cdot \omega_4 \tag{19}$$

4.4. Design of speed control using PID controller

To model the speed conditions of the drive chain, speed control loop was designed using a PID controller. PID controllers are widely used for linear systems. It operates based on parallel compensation. The PID controller possesses 3 components, which are proportional (*P*), integrating (*I*) and derivative (*D*) parts [12]. PID controllers need an error signal, which is the difference between the reference signal and the real output detected by a suitable sensor. In parallel compensation, the proportional part (*P*) creates an executive signal proportional to the error signal, the integral part (*I*) is proportional to the integration of the error signal, and the derivative part (*D*) creates an executive signal proportional to the derived of the error signal until the error signal reaches the switch-off threshold. PID controllers use hysteresis control **[12]**.

The equations of the PID controller are the following:

$$e(t) = K_p \cdot e(t) + K_i \cdot \int_0^t e(t)dt + K_d \cdot \frac{de(t)}{dt} \tag{20}$$

$$e(t) = K_p \cdot \left(e(t) + \frac{1}{T_i} \cdot \int_0^t e(t)dt + T_d \cdot \frac{de(t)}{dt}\right) \tag{21}$$

5. The simulation results

After the simulation model was created, the addition of values was the next step. These values determine the behaviour of the drive chain and they are also necessary to tune the PID controller. The set values during the simulation can be seen in the **Table 1** :

Table 1. *The set values of the simulation*

R_a = 0.45 Ω	k = 0.09 Nm/rad
L_a = 0.1 mH	J = 0,0011 kgm^2
k_m = 0.067 Nm/A, Vs/rad	N_1/N_2 = 0.1
$b_1 = b_3 = b_4$ = 0.1 Nms	N_4/N_3 = 10
b_2 = 0.028 Nms	

If the drive chain operates without a controller it is opened loop and results in unstable behaviour. To solve this, a PID controller was designed and implemented. After implementation, the controller was tuned by using the Ziegler-Nichols tuning method, which is used to determine values in a closed loop. The first step of tuning is to determine the critical amplification amplitude and period time using the proportional (*P*) part. The integration time (T_i) is infinite, while the derivative time (T_d) is zero. Resonance with constant amplitude is produced at the critical amplification, since this is the border of the stability. The K_p, T_i és T_d parameters can be set based on pre-defined formulas of the Ziegler-Nichols method **[12]**.

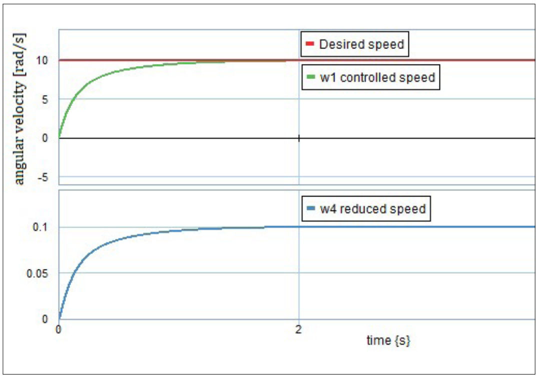

Figure 3. *The displayed angular velocity values.*

The displayed angular velocity results of the simulation can be seen in **Figure 3**:

The first value is the desired reference angular velocity (ω_{ref}), which is represented by a unit step signal. This is the required angular velocity of the DC motor. To present the behaviour of the PID controller the direct output angular velocity of the motor shaft was also displayed. The third displayed value was the angular velocity of the *J* inertia. This value needs to be changed according to the applied gear ratio (1/100), so the angular velocity of the inertia is a hundredth part of the output angular velocity of the motor. After a short transient state, the required value of 10 m/s should appear on the output of the PMDC motor as a result of the PID controller. The angular velocity of the *J* inertia is 1/100 of the angular velocity of the motor, thus this value is 0.1 rad/s.

6. Conclusions

During the study, a Multi-Domain system was modelled, which is an industrial robotic drive. It contains both electrical and rotating mechanical domains. It was important to ensure that the simulation should be dynamically variable. Thus, the choice of the description language of the Bond Graph, proved that it is suitable for this purpose. The description language allowed the graphical notations to translate directly into mathematical equations without a software environment. Since simulation-based visualization was also a purpose, a special software called 20-sim was used. The advantage is the low resource requirement, and the wide toolkit.

References

[1] G. Husi, A. H. Abdulkareem: *Mechatronics Systems in the Cyber-physical Space*. Debreceni Egyetem, Műszaki Kar, 2018.

[2] R. H. Bishop: *The Mechatronics Handbook*. CRC Press LLC, Florida, U.S., 2002.

[3] *Bond Graph Modelling*. 1st International and 16th National Conference on Machines and Mechanisms, iNaCoMM 2013, IIT Roorke, December 18-20, 2013.

[4] J. A. Kypuros: *System Dynamics and Control with Bond Graph Modeling*. CRC Press Taylor & Francies Group LLC, U.S., 2013.

[5] S. Das: *Mechatronic Modeling and Simulation Using Bond Graphs*. CRC Press Taylor & Francis Group LLC, Boca Raton, 2009.

[6] W. Borutzky: *Bond Graphs for Modelling, Control and Fault Diagnosis of Engineering Systems*. Second edition, Springer International Publishing, Switzerland, 2017.

[7] D. C. Karnopp: *System Dynamics – Modeling, Simulation, and Control of Mechatronics System*. Fifth Edition, John Wiley & Sons, Inc., Hoboken, New Jersey, 2012.

[8] R. C. Dorf, R. H. Bishop: *Modern Control Systems*. Thirtheen Edition, Pearson Education Inc., Hoboken, New Jersey, 2017.

[9] J. Pintér: *Ipari robotok hajtása.* Robottechnika, 4. előadás, [Online]. https://docplayer.hu/11254587-Ipari-robotok-hajtasa.html. [Accessed: 20.01.2022.].

[10] A. K. Gupta, S. K. Arora, J. R. Westcott: *Industrial Automation and Robotics: An Introduction*. Mercury Learning & Information Publishing, 2016.

[11] Z. Luo: *Robotics, Automation, and Control in Industrial and Service Settings*. Engineering Science Reference, United States of America, 2015.

[12] S. Duman, U. Guvenc: *Determination of the PID Controller Parameters for Speed and Position Control of DC Motor Using Gravitational Search Algorithm*, January 2011. https://www.researchgate.net/publication/254048079_Determination_of_the_PID_controller_parameters_for_speed_and_position_control_of_DC_motor_using_Gravitational_Search_Algorithm. [Accessed: 20.01.2022.].