

Міністерство освіти і науки України
Одеський національний політехнічний університет

Кваліфікаційна наукова
праця на правах рукопису

Горошко Василь Володимирович

УДК 621.313+ 681.5.01: 517.97

ДИСЕРТАЦІЯ

Оптимізація динамічних режимів електричних машин з використанням дробово-інтегральних регуляторів

Спеціальність: 141 Електроенергетика, електротехніка та електромеханіка

Галузь знань: 14 Електрична інженерія

Подається на здобуття наукового ступеня доктора філософії.

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Горошко

Горошко В. В.

Науковий керівник (консультант)
доктор технічних наук, професор
Бушер Віктор Володимирович

Одеса – 2020 р.

АНОТАЦІЯ

Горошко В.В. Оптимізація динамічних режимів електричних машин з використанням дробово-інтегральних регуляторів. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 141 - Електроенергетика, електротехніка та електромеханіка. – Одеський національний політехнічний університет, Одеса, 2020.

Об'єкти дослідження в дисертаційній роботі – електромеханічні системи, характерною особливістю яких є наявність суттєвої ступеневі залежності в математичному описі. Через це виникають проблеми при виборі структури та параметрів регуляторів. Зокрема, в двигуні постійного струму з послідовним збудженням, вентильно-реактивному двигуні та електромагнітних гальмах може відбуватися насичення магнітної системи в статичних і динамічних режимах. Застосований в роботі апарат дробово-інтегрального числення дозволяє такі нелінійні об'єкти описати з високою точністю лінійними передавальними функціями дробового порядку. Так, для якірного ланцюга двигуна постійного струму з послідовним збудженням така апроксимація дає найменшу середньоквадратичну похибку. Комбінація звичайного PID-регулятора з дробово-інтегральними складовими порядку 0.35 і 1.35 забезпечує найкращу якість перехідного процесу – струм досягає заданого значення максимально швидко без перерегулювання. По-друге, вентильно-реактивний двигун, в моделі якого необхідно враховувати ступеневі залежності, при описі перехідних процесів швидкості при стрибку напруги апроксимований аперіодичною функцією порядку 0.7. З сімейства досліджених регуляторів традиційний PI-регулятор з додатковими дробово-інтегральними складовими порядку 0.7 і 1.7 забезпечив астатизм контуру швидкості порядку 1.7 і найменше перерегулювання. Третій досліджений об'єкт – електромагнітні гальма, що встановлюються на ведучі колеса автомобіля для

налаштування двигуна внутрішнього згорання – також після тестування найбільш точно описано передавальною функцією дробового порядку. За допомогою PIDIT^m-регулятора, що забезпечив астатизм замкненого контуру порядку 1.63, досягнуто стабілізацію швидкості обертання двох коліс без протифазних коливань і точне відпрацювання трикутної тахограми. Таким чином, завдяки апарату дробово-інтегрального числення, забезпечується більш точна ідентифікація параметрів об'єкта, математичний опис зводиться до лінійних передавальних функцій дробового порядку, в замкнених системах можна забезпечити астатизм дробового порядку 1.3...1.7 і домогтися кращої якості перехідних процесів, ніж при використанні класичних методів.

Наукова новизна одержаних результатів полягає в наступному:

1. Вперше виконана ідентифікація якірного ланцюга двигуна послідовного збудження передавальною функцією з дробовим порядком, що в порівнянні з моделями цілочисельного порядку забезпечує мінімальну похибку при розрахунку перехідних процесів при струмі якоря вище номінального, що дозволило вибрати структуру регулятора з дробовим порядком астатизму 1.35 і отримати замкнену систему з найкращими динамічними і статичними показниками.

2. Отримав подальший розвиток метод ідентифікації контуру швидкості в вентильно-реактивному двигуні на базі дробово-аперіодичної функції при різних прикладених напругах з постійними кутами включення і виключення, що дозволило зменшити похибку ідентифікації, синтезувати регулятор з дробовим порядком, при якому отримано мінімальний час першого узгодження з перерегулювання не більше 2%.

3. Вперше синтезовано регулятор з дробовим порядком інтегрування в системі керування двома електромагнітними гальмами в стенді для налаштування паливної системи двигуна внутрішнього згорання автомобілів, що дозволило забезпечити мінімальне перерегулювання з кращою динамічною та статичною точністю, внаслідок чого підвищується точність налаштування паливної системи

та вимірювання максимальної потужності та моменту двигунів.

Практичне значення одержаних результатів:

1. Запропоновано модель і метод ідентифікації параметрів якірного ланцюга двигуна послідовного збудження з подальшим синтезом регуляторів, які враховують ефект насичення магнітної системи двигуна.

2. Запропоновано модель і метод ідентифікації параметрів контуру швидкості вентильно-реактивного двигуна при різних прикладених напругах з подальшим синтезом регуляторів з урахуванням нелінійних залежностей потокозчеплення та моменту від струму фази та положення ротора.

3. Створено систему керування на основі дробового регулятора в стенді налаштування паливної системи двигуна внутрішнього згорання, при яких перехідні процеси мали мінімум перерегулювання і більший порядок астатизму, що дозволяє відпрацьовувати сигнал завдання з точністю $\pm 0.2\%$.

Дослідження двигуна постійного струму з послідовним збудженням, вентильно-реактивного двигуна та електромагнітного гальма проведено за загальноприйнятою методикою. На підставі тестування об'єктів шляхом подачі стандартних сигналів виконувалася ідентифікація параметрів моделей з подальшим синтезом оптимальних регуляторів і перевіркою замкнених систем. На стадії ідентифікації, крім загальноприйнятих, розглянуто моделі, що включають в себе дробово-диференціальні рівняння. Такий вибір обумовлений тим, що в основі апарату дробово-інтегрального числення лежать степеневі функції, а в розглянутих електромеханічних об'єктах крива намагнічування також близька до степеневій функції. Саме це дозволяє стверджувати, що регулятори з інтегральними і диференціальними складовими нецілого порядку можуть забезпечити найкращі динамічні і статичні показники систем.

У вступі обґрунтовано актуальність роботи, сформульовано мету та задачі досліджень, зазначено основні наукові і практичні положення, що виносяться на

захист. Наведено інформацію про апробацію роботи та її структуру, а також про публікації автора за темою дисертації.

У першому розділі надаються короткі відомості про історію розвитку дробового числення, його область застосування (зокрема в системах керування) та огляд різноманітних методів для апроксимаційного розрахунку дробового інтеграла або похідної.

У другому розділі досліджується двигун постійного струму з послідовним збудженням при врахуванні кривої намагнічування в динамічних процесах з подальшою апроксимацією контуру струму передавальною функцією з дробовим порядком та синтезом відповідних регуляторів з різним порядком астатизму.

У такому двигуні при струмі вище номінального перехідні процеси суттєво відрізняються від рішень диференціальних рівнянь першого або другого порядків. При математичному описі перехідного процесу розглянуто наступні передавальні функції об'єкта керування:

$$W_{co1} = \frac{K}{a_0 s^\mu + 1}, W_{co2} = \frac{K}{a_1 s + a_0 s^\mu + 1}, W_{co3} = \frac{K}{a_1 s^{1+\mu} + a_0 s^\mu + 1} \quad (1)$$

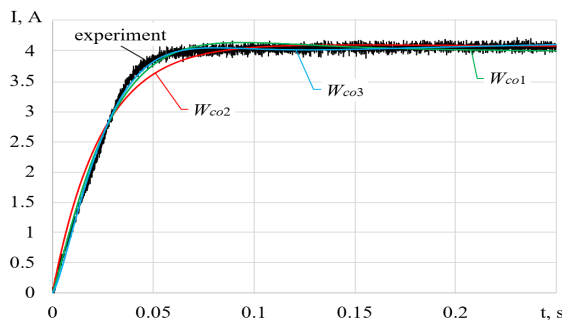


Рис. 1. Графіки перехідних процесів передавальних функцій W_{co1} , W_{co2} , W_{co3} та експериментальні дані

Передавальна функція W_{co3} з дробовим порядком $\mu+1$ має

мінімальну середньоквадратичну похибку серед розглянутих. Експериментальні та розрахункові графіки зіставлені на рис. 1.

В ході експериментальних досліджень найкращу якість перехідного процесу отримано з PIDIT^μ-регулятором (рис. 2, а). У цьому випадку струм досягає заданого значення максимально швидко та без перерегулювання.

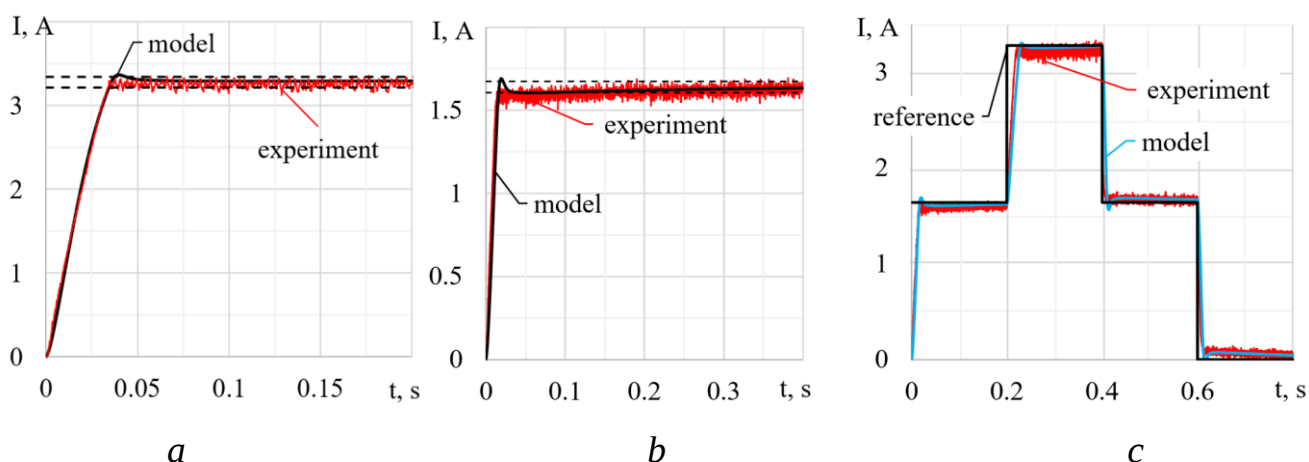


Рис. 2. Перехідний процес при $PID I^{\mu}$ -регуляторі при сигналі завдання струму 3.2 А (a) і 1.6 А (b) та при ступінчастій зміні сигналу (c)

Виявилося, що такий регулятор ефективний і при сигналах завдання вдвічі нижче максимального, а також при ступінчастій зміні завдання (рис. 2, b, c). Видно, що якісні показники системи залишаються незмінними і відповідними оптимальним. Отримані результати є новими, що дозволило сформулювати 1-й пункт наукової новизни.

У третьому розділі досліджується модель вентиляно-реактивного двигуна з врахуванням нелінійної залежності поточкозчеплення та моменту з подальшою апроксимацією контуру швидкості та синтезом регуляторів з різним порядком астатизму.

Для ідентифікації замкненого контуру швидкості вентиляно-реактивного двигуна обрано дробово-аперіодичну передавальну функцію наступного виду:

$$W_{col} = \frac{K}{a_0 s^{\mu} + 1}. \quad (2)$$

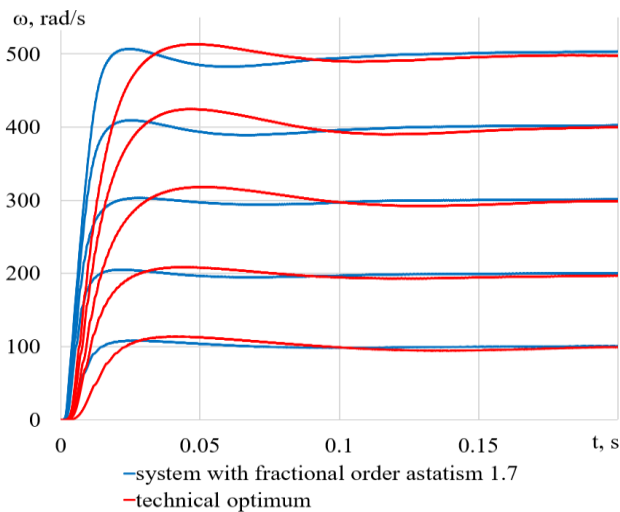
Результати апроксимації ілюструє рис. 3. Як видно, при різних

прикладених напругах характер перехідного процесу змінюється, але використання передавальної функції дробового порядку (2) забезпечує найменше відхилення. Також це дозволяє при синтезі регуляторів відмовитися від складної моделі двигуна, яка будується на системі

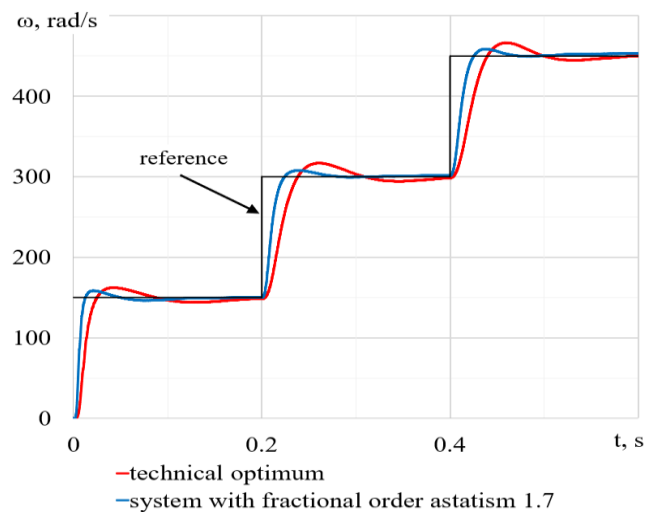
нелінійних диференціальних рівнянь.

Відповідно, істотно спрощується налаштування замкнутого контуру – модульний оптимум забезпечує Π^γ -регулятор, а дробовий порядок астатизму 1.7 досягається за допомогою $\text{РП}^\gamma\Gamma^\mu$ -регулятора.

На рис. 4 показано результати моделювання перехідних процесів в контурі керування швидкістю вентильно-реактивного двигуна. На рис. 4, *a* – графіки перехідних процесів при стрибку завдання, на рис. 4, *b* – при ступінчастій зміні сигналу завдання. У обох випадках



a



b

Рис. 4. Графіки перехідних процесів швидкості вентильно-реактивного двигуна: *a* – при стрибках завдання, *b* – при ступінчастій зміні сигналу завдання

система з дробовим порядком астатизму характеризується кращими динамічними показниками.

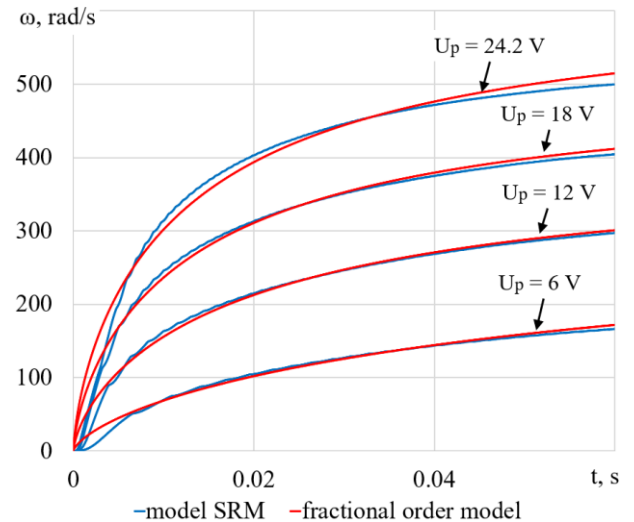
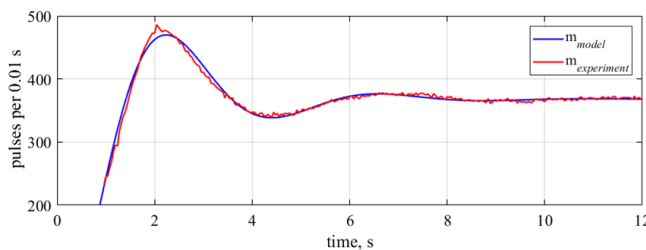


Рис. 3. Графіки перехідних процесів швидкості в розімкненій системі при різних значеннях напруги

У четвертому розділі досліджено систему керування двома електромагнітними гальмами. За принципом дії електромагнітне гальмо має нелінійні залежності, що ускладнює синтез системи керування. Також на відміну від двох інших об'єктів електромагнітні гальма вбудовані в стенд для налаштування паливної системи двигуна внутрішнього згоряння автомобілів. Тому спочатку підібрано PID-регулятор, при якому отримано перехідний процес з найменшими коливальністю, перерегулюванням та протифазними коливаннями швидкостей коліс. Це дозволило ідентифікувати структурну схему і параметри динамічної моделі об'єкта керування.

З розглянутих варіантів математичного опису найкращий і практично однаковий збіг розрахункових і експериментальних характеристик забезпечили дві передавальні функції – інерційна ланка 3-го порядку:

$$W_{co1}(s) = \frac{K}{(a_1s + 1)(a_2s + 1)(a_3s + 1)} \quad (3)$$

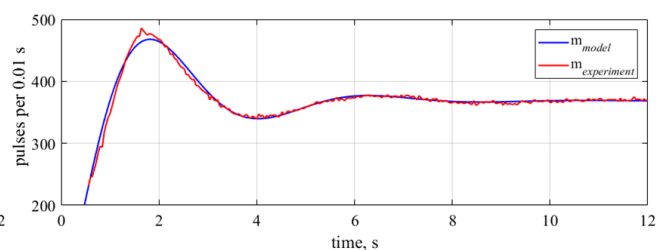


a

та модель дробового порядку:

$$W_{co2}(s) = \frac{K}{a_1s^{1+\mu_{co}} + a_2s^{\mu_{co}} + 1} \quad (4)$$

Перехідні процеси за передавальними функціями (3), (4) та отримані експериментально зіставлені на рис. 5.



b

Рис. 5. Графіки перехідних процесів моделі m_{model} при передавальних функціях W_{co1} (a), W_{co2} (b) та експерименту $m_{experiment}$

На основі передавальних функцій W_{co1} (3) та W_{co2} (4) отримано PID-, D^γT^μI- та PID^μT^γ-регулятор. Проведені експериментальні дослідження дозволили зіставити

швидкодію та перерегулювання за швидкістю в системі з синтезованими регуляторами (рис. 6).

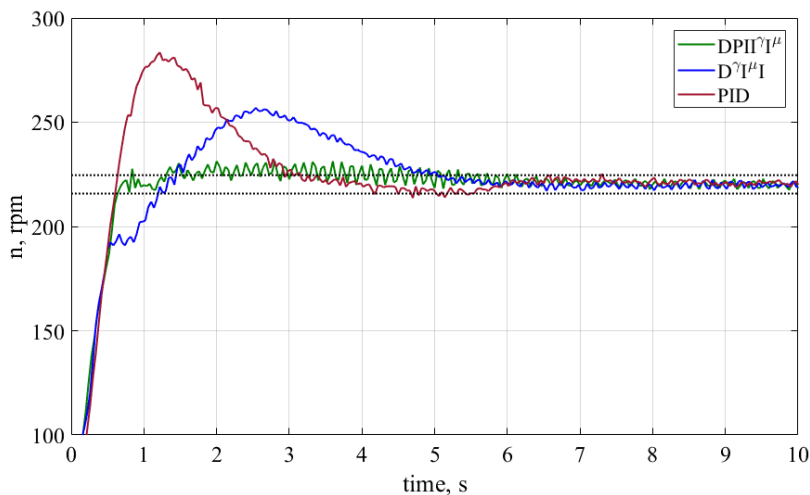


Рис. 6. Графіки переходних процесів середньої швидкості коліс автомобіля при накиду навантаження від ДВЗ з різними регуляторами

PID-регулятор забезпечив кращу швидкодію (2.55 s), але з найбільшим перерегулюванням, яке склало $\delta_1 = 27.6 \%$. З $D^{\gamma}I^{\mu}$ -регулятором перерегулювання склало $\delta_2 = 16.3 \%$, а час переходного процесу – 4.42 s. Найменше перерегулювання отримано з $DPID^{\gamma}I^{\mu}$ -регулятором – $\delta_3 = 3.3 \%$ при тривалості переходного процесу 3.8 s.

Проведено експериментальне дослідження при вимірюванні потужності і максимального моменту при лінійному збільшенні швидкості (рис. 7, a) та трикутній тахограмі (рис. 7, b). Результати такого тесту з найшвидшим (PID) і найточнішим ($DPID^{\gamma}I^{\mu}$) регулятором наведені на рис. 7, a.

Обидва регулятори забезпечують стабілізацію швидкості у всьому діапазоні вимірювань потужності, але на початку розгону (рис. 7, a) $DPID^{\gamma}I^{\mu}$ -регулятор забезпечує найменше перерегулювання, що дозволило прийняти остаточне рішення вибору структури регулятора в його користь.

Визначено швидкісну помилку для двох регуляторів з лінійного завдання (рис. 7, a) на часовому діапазоні з 18 до 50 s. Таким чином показано, що швидкісна

помилка при $PIDIT^{\mu}$ -регуляторі зменшується майже в три рази швидше, ніж при PID -регуляторі.

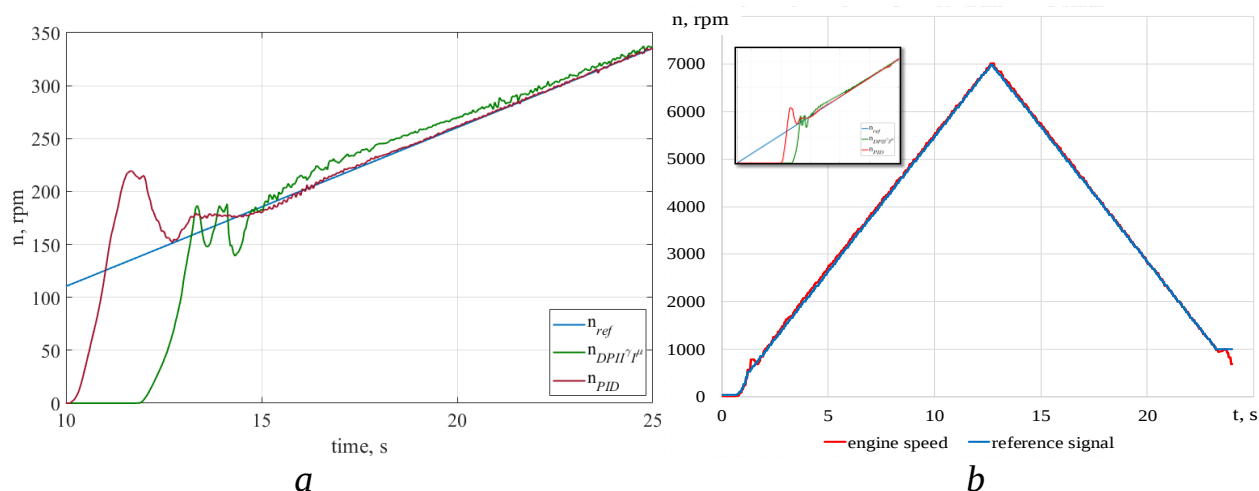


Рис. 7. Перехідні процеси по середній швидкості між напіввісями диференціала при вимірюванні потужності, де a – на початку розгону, b – на трикутній тахограмі

Отже, отримані результати в другому, третьому та четвертому розділі повністю відповідають вище сформульованій науковій новизні та на підставі досліджень трьох типів електричних машин можна зробити загальний висновок, що дробово-інтегральне числення надає можливість опису перехідних процесів в системах зі степеневими властивостями (у разі електричних машин причиною їх існування є насичення магнітної системи) як лінійних об'єктів з диференціальними рівняннями дробового порядку, що, в свою чергу, дозволяє використовувати класичний апарат теорії автоматичного керування для лінійних систем для аналізу таких об'єктів та синтезу регуляторів, які забезпечують оптимізацію динамічних режимів різних за принципом роботи електричних машин за мінімумом перерегулювання та часом першого узгодження. Так, з використанням регуляторів з приблизно однаковою структурою отримано оптимальні показники перехідних процесів в контурі струму двигуна послідовного збудження отримано з $PIDIT^{\mu}$ -регулятором, в контурі швидкості вентильно-реактивного двигуна – з $PIIT^{\mu}$ -регулятором.

регулятором та при керуванні електромагнітними гальмами – з PID^{PI}-регулятором.

Ключові слова: дробове числення, двигун постійного струму з послідовним збудженням, регулятор з дробовим порядком, генетичний алгоритм, вентильно-реактивний двигун, дробовий астатизм, електромагнітне гальмо, двигун внутрішнього згоряння, мікропроцесорна система керування.

ABSTRACT

Horoshko V. V. Optimization of dynamic modes of electric machines using fractional-order integral controllers. – Qualifying scientific work on the rights of the manuscript.

Dissertation for the degree of Doctor of Philosophy in the specialty 141 - Electrical Power Engineering, Electrical Engineering and Electromechanics. – Odessa National Polytechnic University, Odessa, 2020.

Objects of research in the dissertation – electromechanical systems, a characteristic feature of which is the presence of a significant power dependence in the mathematical description. Because of this, there are problems in choosing the structure and parameters of the controllers. In particular, in a DC motor with sequential excitation, a switched reluctance motor and electromagnetic retarders, the magnetic system can be saturated in static and dynamic modes. The fractional-integral calculus used in the work allows to describe such nonlinear objects with high accuracy by linear transfer functions of fractional order. Thus, for the armature circuit of a DC motor with sequential excitation, such an approximation gives the least root mean square error. The combination of a conventional PID controller with fractional-integral components of the order of 0.35 and 1.35 provides the best quality of the transient process – the current reaches the reference value as quickly as possible without overshoot. Secondly, the switched reluctance motor, in the model of which it is necessary to take into account power dependences, is described by the aperiodic function of the order of 0.7 when

describing the transient processes of the speed during a voltage jump. From the family of studied controllers, the traditional PI-controller with additional fractional-integral components of the order of 0.7 and 1.7 provided the astaticism of the speed loop of the order 1.7 and the smallest overshoot. The third object studied – electromagnetic retarders mounted on the drive wheels of the car to tune the internal combustion engine – also is most accurately described by the transfer function of the fractional order. With the help of the $PIDIT^\mu$ -controller, which provided the closed-loop astaticism of the order of 1.63, the stabilization of the speed of rotation of the two wheels without antiphase oscillations and the exact trajectory of the triangular tachogram were achieved. Thus, thanks to the apparatus of fractional-integral calculus, more accurate identification of object parameters is provided, the mathematical description is reduced to linear transfer functions of fractional order, in closed systems it is possible to provide astaticism of fractional order 1.3 ... 1.7 to achieve better quality of transients than using classical methods.

The scientific novelty of the obtained results is as follows:

1. *For the first time* perform the identification of the armature circuit in the series DC motor by the transfer function with fractional order, which in comparison with integer order models provides minimal error in calculating transients at armature current above nominal, which allowed to choose the structure of the controller with fractional order of astaticism with the best dynamic and static qualities.
2. The method of speed identification in a switched-reluctance motor based on fractional-aperiodic function at different applied voltages with constant on and off angles *was further developed*, which allowed to reduce the identification error and to synthesize a controller with fractional order, with the minimum rise time and overshoot not more than 2%.
3. *For the first time* a controller with a fractional order of integral in the control system of two electromagnetic retarders in the stand for tune the fuel system of the car's

internal combustion engine was synthesized, which allowed to provide more accuracy for measuring torque and electromechanical power.

The practical significance of the obtained results:

1. A model and a method for identifying the parameters of the armature circuit of a series DC motor with subsequent synthesis of controllers that take into account the saturation effect of the motor magnetic system is proposed.

2. The model and method of identification of parameters in the speed loop of the switched-reluctance motor at different voltages with the subsequent synthesis of controllers taking into account nonlinear dependences of flux linkage and the torque on a phase current and position of a rotor are offered.

3. A control system based on a fractional controller was applied in the stand for tune the fuel system of the internal combustion engine, in which the transients had a minimum of overshoot and a higher order of astaticism, which allows to reproduce the reference signal with an accuracy of $\pm 0.2\%$.

The study of each of the listed objects was carried out according to the generally accepted methodology. Based on the testing of objects by supplying standard signals, identification of model parameters was carried out with the subsequent synthesis of optimal controllers and verification of closed systems. However, at the identification stage, in addition to the generally accepted ones, models are considered that include fractional differential equations. This choice is due to the fact that the fractional calculus is based on power functions, and in the considered electromechanical objects the magnetization curve is also close to the power function. This allows us to say that controllers with fractional integral and differential components can provide the best dynamic and static performance of control systems.

The introduction proves the relevance of the work, formulates the purpose and objectives of research, indicates the main scientific and practical provisions submitted for defense. The information on approbation of work and its structure, and also provide the publications of the author on a dissertation theme.

The first chapter provides brief information about the history of fractional calculus, its scope (particularly in control systems) and an overview of various methods for approximate calculation of the fractional integral and derivative.

The second chapter investigates a series DC motor taking into account the magnetization curve in dynamic processes with subsequent approximation of the current circuit by the transfer function with fractional order and synthesis of corresponding controllers with different order of astaticism.

In such a motor at a current value above the nominal transients are significantly different from the solutions of differential equations of the first or second order. In the mathematical description of the transition process, the following transfer functions of the control object are considered:

$$W_{co1} = \frac{K}{a_0 s^\mu + 1}, W_{co2} = \frac{K}{a_1 s + a_0 s^\mu + 1}, W_{co3} = \frac{K}{a_1 s^{1+\mu} + a_0 s^\mu + 1} \quad (2)$$

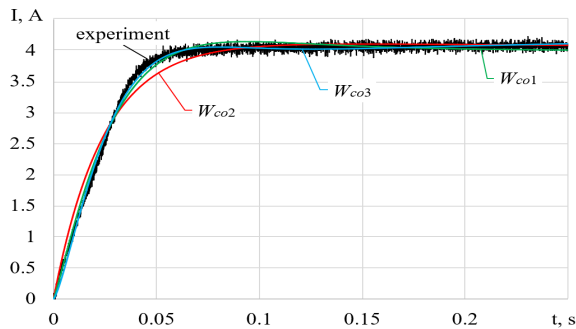


Fig. 1. Transient for transfer functions W_{co1} , W_{co2} , W_{co3} and experimental data

The transfer function W_{co3} with fractional order $\mu + 1$ has a minimum root square mean error. Experimental and calculation diagrams are compared in fig. 1.

In experimental studies, the best quality of the transient was obtained with a

PID I^μ -regulator (Fig. 2, a). In this case, the current reaches the set value as quickly as possible and without overshoot.

It turned out that such a controller is effective for task signals twice below the maximum, as well as for a step-wise change of reference signal (fig. 2, b, c). The quality of the system remains unchanged and correspond to optimal. The obtained results are new, which allowed to formulate the 1st point of scientific novelty.

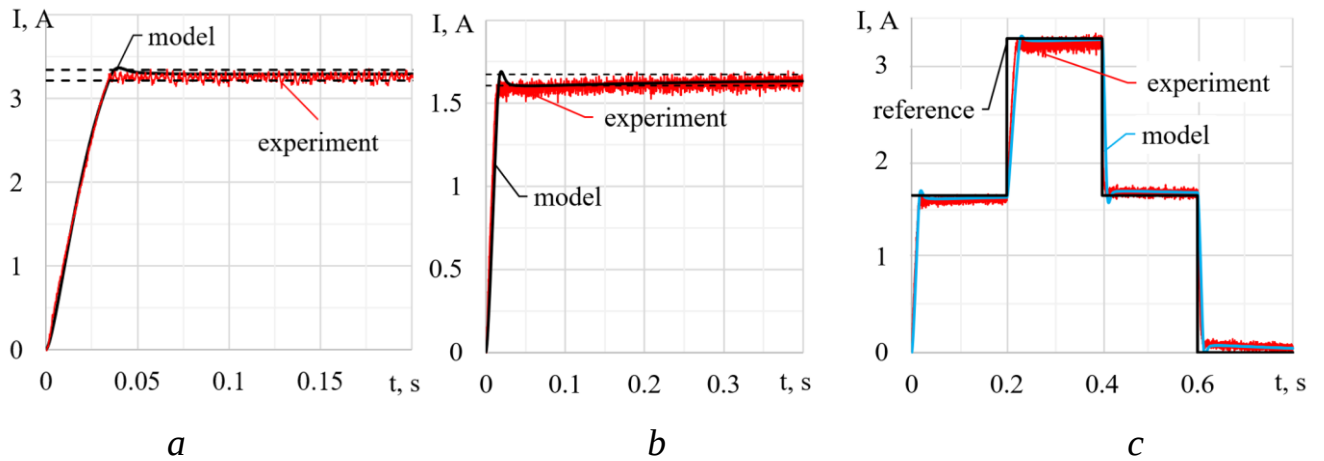


Fig. 2. Transients with $\text{PIDI}^{\gamma}\text{I}^{\mu}$ -controller at the current reference signal 3.2 A (a), 1.6 A (b) and at a step-wise change in the reference signal (c)

In **the third chapter** the model of switched-reluctance motor is investigated. In model taken into account nonlinear dependence of flux linkage and the torque with the subsequent approximation of speed and synthesis controllers with a different order of astaticism.

To identify the speed closed loop of switched-reluctance motor, a fractional-aperiodic transfer function of the following type is selected:

$$W_{col} = \frac{K}{a_0 s^{\mu} + 1}. \quad (2)$$

The results of the approximation are illustrated in fig. 3. As can be seen, at different applied voltages, the nature of the transient process changes, but the use of the transfer function of the fractional order (2) provides the smallest

deviation. It also allows the synthesis of controllers without considering the complex model of the motor, which is based on a system of nonlinear differential equations.

Accordingly, the tuning of the closed loop is significantly simplified – the modular optimum is provided by the II^{γ} -controller, and the fractional order of astaticism 1.7 is achieved with the $\text{PII}^{\gamma}\text{I}^{\mu}$ -regulator.

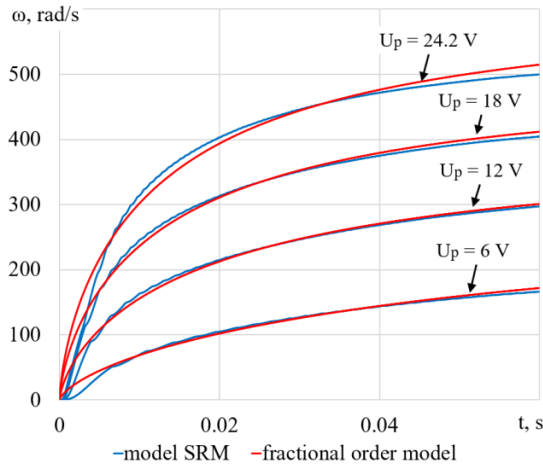


Fig. 3. Transients of speed in an open loop at different voltage

In fig. 4 shows the results of simulation of transients in the speed control loop of the switched-reluctance motor. In fig. 4, *a* – diagram of transients during the step reference, in Fig. 4, *b* – at a step-wise change in the reference signal. In both cases, the system with a fractional order of astaticism is characterized by better dynamic performance.

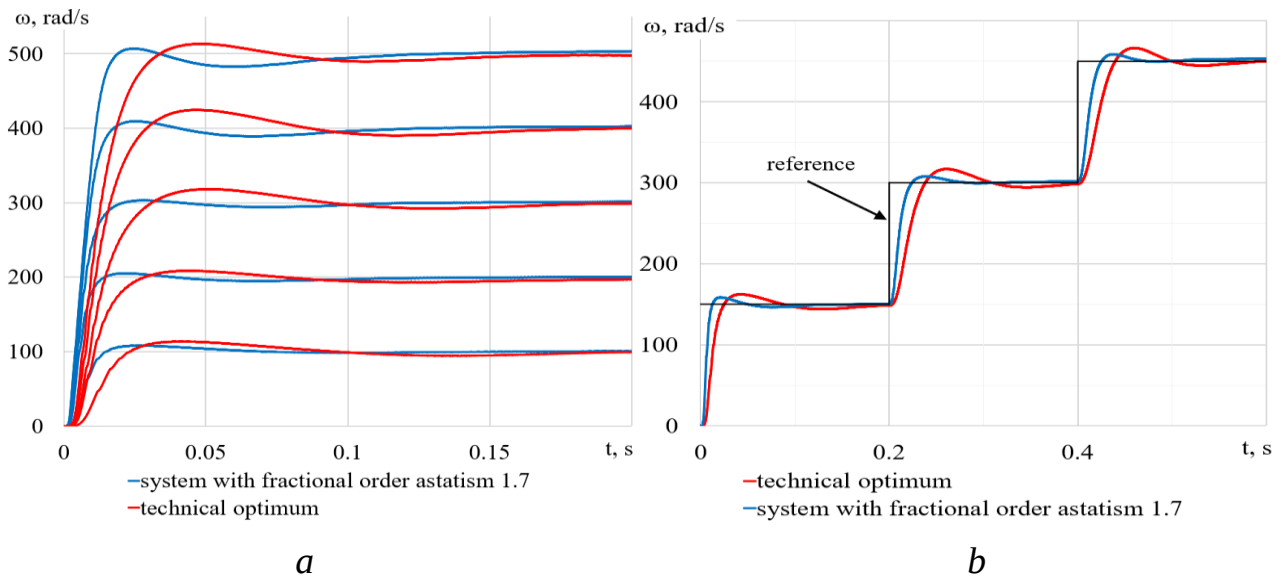


Fig. 4. Transients of speed switched-reluctance motor: *a* – at different value of speed reference signal, *b* – at a step-wise change in the reference signal

The fourth chapter investigates the control system of two electromagnetic retarders. According to the principle of operation, the electromagnetic brake has nonlinear dependences, which make hard to synthesis the control system. Also, unlike the other two objects, electromagnetic retarders are built into the stand to tune the fuel system of the car's internal combustion engine. Therefore, it was selected PID-controller

for obtain a transient with the least oscillations, overshoot and antiphase oscillations of wheel speeds. This allowed us to identify the block diagram and parameters of the dynamic model in the control object.

Of the considered variants of the mathematical description, the best and almost identical coincidence of the calculated and experimental characteristics was provided by two transfer functions – the inertial link of the 3rd order:

$$W_{co1}(s) = \frac{K}{(a_1s + 1)(a_2s + 1)(a_3s + 1)} \quad (3)$$

and fractional order model:

$$W_{co2}(s) = \frac{K}{a_1s^{1+\mu_{co}} + a_2s^{\mu_{co}} + 1} \quad (4)$$

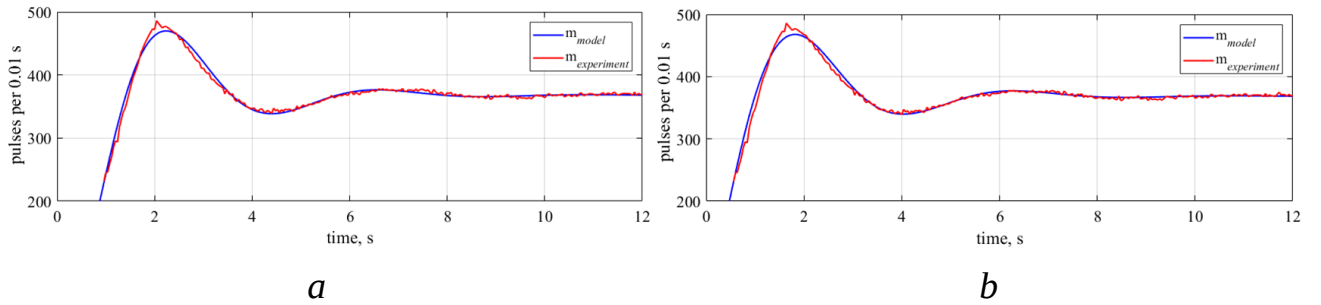


Fig. 5. Transients in model m_{model} by transfer functions W_{co1} (a), W_{co2} (b) and experimental data $m_{experiment}$

The PID controller provided better performance (2.55 s), but with the largest overshoot, which value $\delta_1 = 27.6\%$. With the D^γI^μI controller, the overshoot value $\delta_2 = 16.3\%$, and the settling time 4.42 s. The smallest overshoot was obtained with the PID^γI^μ controller – $\delta_3 = 3.3\%$ with a settling time 3.8 s.

Transients in the transfer functions (3), (4) and obtained experimental data compared in fig. 5.

Based on the transfer functions W_{co1} (3) and W_{co2} (4), the PID-, D^γI^μI- and PIDI^μI^γ-controller are obtained. The experimental researches allowed to compare the speed and speed overshoot in the system with the synthesized controllers (Fig. 6).

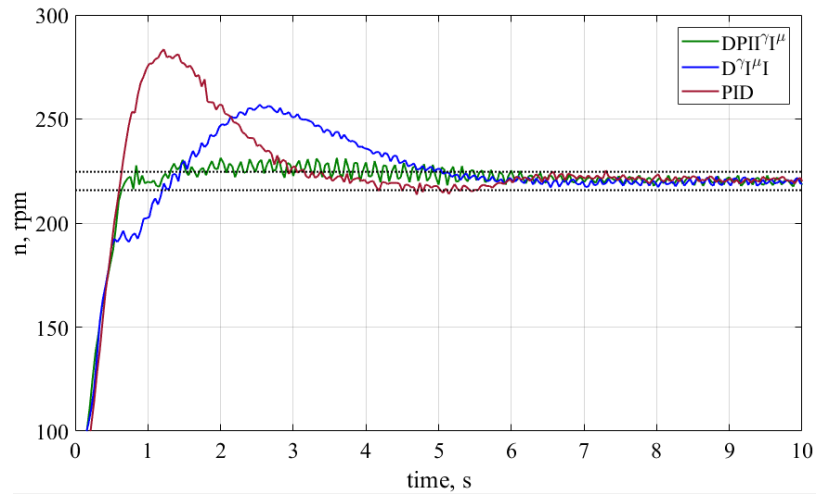


Fig. 6. Transients in average speed of wheels car during stabilization

An experimental study was performed by measuring power and maximum torque with a linear increase in speed (fig. 7, *a*) and a triangular tachogram (fig. 7, *b*). The results of such a test with the fastest (PID) and most accurate ($\text{PIDI}^\mu\text{T}^\mu$) controller are shown in fig. 7, *a*.

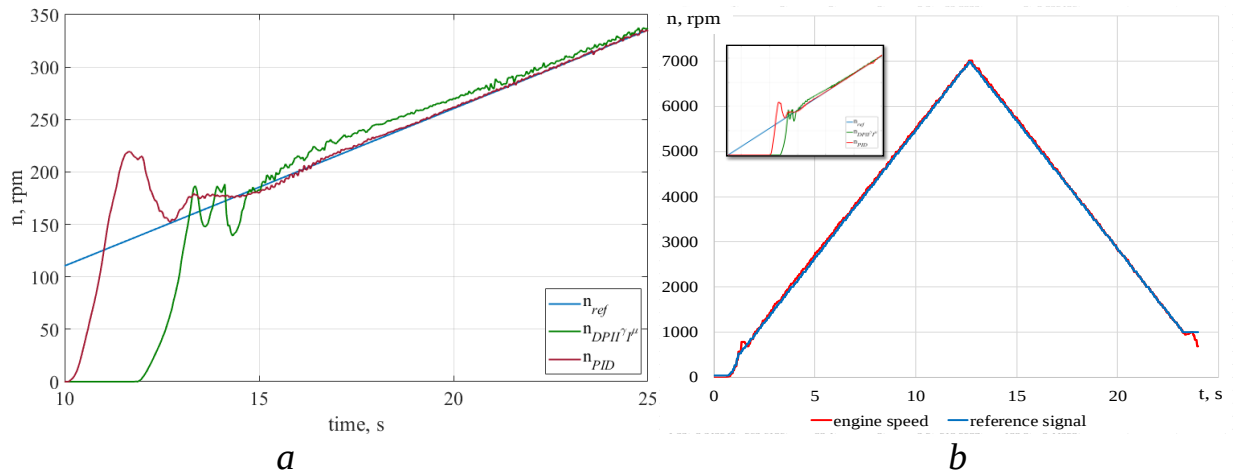


Fig. 7. Transients on the average speed between the semi-axes of the open differential then measuring power, where *a* – at the beginning of acceleration, *b* – at a triangular tachogram

Both controllers provide speed stabilization in the entire range of power measurements, but at the beginning (fig. 7, *a*) $\text{PIDI}^\mu\text{T}^\mu$ -controller provides the least

overshoot, which allowed to make the final decision on the choice of controller structure in its favor.

The speed error for two controllers from the linear reference (fig. 7, *a*) in the time range from 18 to 50 s is determined. Thus, it is shown that the speed error at the $PID^{I^{\mu}I^{\nu}}$ controller decreases almost three times faster than at the PID controller.

The results obtained in the second, third and fourth chapters fully correspond to the above scientific novelty. Based on studies of three types of electric machines it can be concluded that fractional-integral calculus provides a description of transients in systems with power function properties (in the case of electric machines the reason for its existence is the saturation of the magnetic system) as linear objects with differential equations of fractional order, which, in turn, allows to use the classical apparatus of automatic control theory for linear systems to analyze such objects and synthesize controllers that optimize dynamic modes different on the principle of operation in electric machines with a minimum of overshoot and the rise time. Thus, with the use of controllers with approximately the same structure, the optimal transients in the current circuit of the series DC motor were obtained with a $PID^{I^{\mu}I^{\nu}}$ -controller, in the speed loop of switched-reluctance motor with a $PIII^{I^{\mu}I^{\nu}}$ -controller and when controlling electromagnetic retarders – $PID^{I^{\mu}I^{\nu}}$ -controller.

Keywords: fractional calculus, series DC motor, fractional order controller, genetic algorithm, switched-reluctance, fractional astaticism, electromagnetic retarders, internal combustion engine, microprocessor control system.

Список публікацій здобувача

1. Бушер В. В. Модифицированный метод управления моментом вентильно-реактивного двигателя [Текст] / В. В. Бушер, В. В. Горошко, К. А. Хандакжи // Электротехнические и компьютерные системы, ISSN Print 2221-3937, ISSN Online 2221-3805 – ELTEKS-2017. – Киев: Техника, 2017. – Вып. 25(101). – С. 44–52. Access mode: <http://etks.opu.ua/core/getfile.php?id=2185>, <http://dx.doi.org/10.15276/eltecs.25.101.2017.05>, **Фах.вид.кат. В**
2. Бушер, В., Хандакжи, К., Горошко, В., Кравченко, В. Техническая реализация дробных интегрально-дифференцирующих регуляторов в микропроцессорных системах управления. [Текст] // International conference Ships' Electrical Engineering, Electronics And Automation, 05.12.2018 – 06.12.2018 – Odessa: NU «ОМА». – 2019. – Р. 180–186. Access at: http://femire.onma.edu.ua/docs/conf/electro_conf_50_60ka_228str.pdf
3. Busher, V. Synthesis and implementation of fractional-order controllers in a current circuit of the motor with series excitation [Text] / V. Busher, L. Melnikova, V. Horoshko // Eastern-European Journal of Enterprise Technologies ISSN 1729–3774. Industry Control System. – Kharkov: PC Technology Center, 2019. – Vol. 2, No 2 (98), (2019). – Р. 63–72. Access mode: <http://journals.uran.ua/eejet/article/view/161352> DOI: 10.15587/1729-4061.2019.161352, **SCOPUS – Q2, Фах.вид.кат. А**
4. Бушер В. В. Дробные интегрально-дифференцирующие регуляторы в системе управления двигателем постоянного тока с последовательным возбуждением [Текст] / В. В. Бушер, В. В. Горошко // Електромеханічні та енергозберігаючі системи. Методи моделювання та оптимізації. XXVII МНТК 11-12 квітня 2019 р. ISSN 2079-5106. – Кременчук: КрНУ, 2019. – С. 39–41. Access at: http://esmo.kdu.edu.ua/publ/ESMO_2019.pdf, **Фах.вид.кат. Б**
5. Busher V.V., Goroshko V.V. Fractional Integral-Differentiating Control in Speed Loop of Switched Reluctance Motor [Текст] // Problemele Energeticii Regionale – vol. 1, No 2(42) – 2019 – Р. 46–54. – ELTEKS-2019. http://journal.ie.asm.md/assets/files/05_12_41_2019.pdf <https://doi.org/10.5281/zenodo.3239166> <http://journal.ie.asm.md/en/contents/electronii-jurnal-1241-2019>, **WEB OF SCIENCE**
6. Busher V. Dual Electromagnetic Retarder Control System for Tuning Internal Combustion Engines. / Busher V. V., Horoshko V. V. // Modern Electrical and Energy Systems (MEES–2019) September 23-25, 2019 – ISBN 978-1-7281-2569-

- 5, IEEE Cat. No.: CFP19K83-ART. – Kr.: M.Ostrogradsky NU – 2019. – P.26–29. Access at: <https://ieeexplore.ieee.org/document/8896526>, **SCOPUS / WEB OF SCIENCE**
7. Horosko, V. Control optimization of electromechanical systems by fractional-integral controllers / Technology Audit and Production Reserves. Information and Control Systems. p-ISSN 2664-9969, e-ISSN 2706-5448. – Kharkov: PC Technology Center, 2020. – Vol. 3, No 2 (53), (2020). – P. 56–59. DOI: <https://doi.org/10.15587/2706-5448.2020.207037>, **Фах.вид.кат. Б**
8. Горошко, В. В. Применение аппарата дробного исчисления в системах управления электрическими машинами / Theoretical and Practical Foundations of Social Process Management. XXIII International Scientific and Practical Online Conference, San Francisco, USA, 29–30 June, 2020. – с. 172–176. <https://doi.org/10.46299/ISG.2020.XXIII>, ISBN 978-1-64871-431-3,
9. Simulation of Internet of Things based Systems. Practicum / O.V. Drozd, D. A. Maevsky, O. J. Maevskaya, O.M. Martynyuk, G.V. Tabunshchyk, V.V. Goroshko and other, D. A. Maevsky (ed.) – Ministry of Education and Science of Ukraine, Odessa National Polytechnic University, 2019. – 130 p. – ISBN 978-617-7361-95-3. https://alioi.eu.org/wp-content/uploads/2019/10/ALIOT_PC1_Sim-of-IoT-Sys_web.pdf

ЗМІСТ

ВСТУП.....	25
1. ДРОБОВЕ ЧИСЛЕННЯ В СИСТЕМАХ КЕРУВАННЯ.....	35
1.1 Коротка історія розвитку дробового числення.....	35
1.2 Області застосування дробового числення.....	42
1.3 Дробові регулятори в системах керування.....	45
1.4 Підлегле регулювання координат з налаштуванням на дробовий порядок астатизму.....	53
1.5 Апроксимаційні методи розрахунку дробових похідних та інтегралів.....	58
2. РЕГУЛЯТОРИ З ДРОБОВИМ ПОРЯДКОМ ІНТЕГРУВАННЯ ТА ДИФЕРЕНЦІЮВАННЯ В СИСТЕМІ КЕРУВАННЯ СТРУМОМ ДВИГУНА ПОСТІЙНОГО СТРУМУ З ПОСЛІДОВНИМ ЗБУДЖЕННЯМ.....	68
2.1 Аналіз моделі двигуна постійного струму з послідовним збудженням.....	68
2.2 Моделювання замкнутої системи керування за струмом.....	73
2.3 Експериментальні дослідження контуру струму двигуна послідовного збудження.....	81
ВИСНОВКИ ПО РОЗІЛУ 2.....	98
3. РЕГУЛЯТОРИ З ДРОБОВИМ ПОРЯДКОМ ІНТЕГРУВАННЯ ТА ДИФЕРЕНЦІЮВАННЯ В СИСТЕМІ КЕРУВАННЯ ШВИДКІСТЮ ВЕНТИЛЬНО-РЕАКТИВНОГО ДВИГУНА.....	100
3.1 Аналіз об'єкта.....	100
3.2 Математична модель контуру швидкості вентильно-реактивного двигуна.....	102
3.3 Ідентифікація параметрів об'єкта керування передавальною функцією з дробовим порядком.....	111
3.4 Синтез регуляторів швидкості за моделлю з дробовим порядком.....	116
ВИСНОВКИ ПО РОЗДІЛУ 3.....	126

4. РЕГУЛЯТОРИ З ДРОБОВИМ ПОРЯДКУ ІНТЕГРУВАННЯ ТА ДИФЕРЕНЦІЮВАННЯ В СИСТЕМІ КЕРУВАННЯ ЕЛЕКТРОМАГНІТНИМ ГАЛЬМОМ ДЛЯ НАЛАШТУВАННЯ ДВИГУНІВ ВНУТРІШНЬОГО ЗГОРЯННЯ.....	128
4.1 Аналіз об'єкта і попередній синтез системи керування.....	128
4.2 Вимірювальний стенд ДВЗ з двома електромагнітними ретардерами.....	138
4.3 Експериментальні дослідження PID-регулятора за середнім значенням швидкостей напіввісей диференціала автомобіля.....	147
4.4 Експериментальні дослідження двоконтурної системи керування.....	151
4.5 Синтез системи керування електромагнітними ретардерами на базі дробово-інтегральних регуляторів.....	154
ВИСНОВКИ ПО РОЗДІЛУ 4.....	166
ВИСНОВКИ.....	167
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	170
ДОДАТОК А. Матеріали впровадження результатів дисертаційної роботи.....	182
ДОДАТОК Б. Код розробленого програмного забезпечення	188

Перелік умовних позначень

IGBT – insulated-gate bipolar transistor;

ВРД – вентильно-реактивний двигун;

ДВЗ – двигун внутрішнього згоряння;

ДПС ПЗ – двигун постійного струму з послідовним збудженням;

ДС – датчик струму;

ЛАЧХ – логарифмічна амплітудно-частотна характеристика;

ЛФЧХ – логарифмічна фазо-частотна характеристика;

МК – мікроконтролер;

МСЕ – метод скінчених елементів;

ПК – персональний комп'ютер;

ПП – перехідний процес;

СУ – система управління;

ТАК – теорія автоматичного керування;

ШІМ – широтна-імпульсна модуляція;

I_n – номінальний струм;

T_μ – мала постійна часу;

σ – перерегулювання;

t_f – час першого узгодження;

μ, γ – порядок дробового диференціювання або інтегрування.

ВСТУП

Електричні машини широко використовуються в різних сферах діяльності людини, а динамічні режими є невід'ємною частиною їх роботи. Тому оптимізація даних режимів безпосередньо впливає на ефективність і продуктивність таких машин.

Для оптимізації перехідних процесів потрібний найбільш точний опис об'єкта керування. Зазвичай це диференційні рівняння з цілочисельним порядком. Але для деяких об'єктів характерною особливістю є наявність степеневих залежностей в математичному описі, що призводить до похибок при моделюванні динамічних процесів. В цьому випадку більшу точність дають диференціальні рівняння з дробовим порядком. Насичення магнітної системи (залежність потоку від струму в обмотках) в електричних машинах має степеневий характер. Двигуни постійного струму з послідовним збудженням, вентильно-реактивні двигуни і електромагнітні гальма – це найбільш яскраві представники електричних машин, які часто або майже завжди працюють в режимі магнітного насичення. Тому ідентифікація математичних моделей з дробовим порядком інтегрування і диференціювання дасть більш точний результат, що надалі дозволить оптимізувати динамічні режими.

Роботи Васильєва В. В. і Симак Л. О. пов'язані з напрямком моделювання динамічних систем, за допомогою апроксимаційно-операційних методів зі змішаними порядками диференціальних рівнянь. Розглядаються динамічні процеси в електроніці та енергетиці.

Напрямок робіт Бушера В. В. пов'язано з замкнутими системами управління з налаштуванням на дробовий порядок астатизму від 0.5 до 2 з оптимальними перехідними процесами. Приділено увагу ідентифікації об'єктів моделями з дробовим порядком за допомогою генетичного алгоритму, зокрема кліматичних

установок, суперконденсаторів і т. д. Також в роботах запропоновано удосконалений метод розрахунку дробового інтеграла.

Марущак Я. Ю. в своїх роботах приділяє увагу теорії синтезу регуляторів з дробовим порядком, за допомогою апроксимації передавальних функцій з більш високим цілочисельним порядком, а також робастності таких систем керування.

Роботи Копчака Б. Л. мають здебільшого практичні застосування в електромеханічних системах. Наприклад, в його роботах досліджуються регулятори з дробовим порядком в системах управління напругою автономного асинхронного генератора та в перетворювачах частоти. Досліджуються застосування генетичних алгоритмів і метод рою частинок для синтезу таких регуляторів.

У роботах Чорного О. П. і Титюка В. К. розглядається процес ідентифікації вентильно-реактивного двигуна за перехідним процесом швидкості математичною моделлю з дробовим порядком, а також досліджено вплив моменту інерції на параметри цієї моделі.

Сергієнко С. А. в своїх роботах приділяє увагу пошуку оптимальних параметрів PID-регулятора з дробовим порядком в системі управління двигуном з незалежним збудженням за допомогою регресійної моделі.

Напрямок Ушакова П. А. пов'язано з синтезом елементів (RLC ланцюга), фільтрів, регуляторів з дробовим порядком. Застосування пристроїв з фрактальним імпедансом в радіофізиці та електроніці.

Роботи Індраніла Пана і Саптарші Даса спрямовані на дослідження інтелектуальних методів контролю, таких як нейро-фаззі-регуляторів із застосуванням еволюційних алгоритмів і методів рою частинок, в сукупності з регуляторами з дробовим порядком.

Напрямок Ін Ло (Ying Luo) має зв'язок з теорією автоматичного управління із застосуванням PID-регулятора дробового порядку в мікропроцесорних

системах. Велику увагу приділено системам управління швидкістю на базі дробового числення.

Ігор Подлубний в своїх роботах досліджує чисельні методи розрахунку диференціальних рівнянь з дробовим порядком, а також моделювання та управління систем, які описуються такими рівняннями.

Роботи Іво Петраса пов'язані з напрямком синтезу регуляторів з дробовим порядком для забезпечення необхідних якостей перехідних процесів. Є роботи присвячені реалізації дрібних регуляторів в аналогових і цифрових пристроях, а також досліджені хаотичні системи дробового порядку.

Двигуни постійного струму з послідовним збудженням (або їх ще називають універсальними двигунами) – одні з найбільш поширених електричних машин. Це пов'язано з тим, що такі двигуни мають просту конструкцію, мають високу частоту обертання (можуть розвивати швидкість до 20 000 rpm), вартість виробництва дешевше (у порівнянні з двигунами з постійними магнітами), а також можуть працювати, як від джерела постійної напруги, так і від змінного. Тому універсальні двигуни широко застосовуються в побутовій техніці. Наприклад, вони використовуються в пральних машинах, дрелях, кутових шліфувальних машинах, пирососах і т. д. Також двигуни постійного струму з послідовним збудженням мають високий пусковий момент, що робить їх придатними в електротранспорті, ліфтах і кранах.

Вентильно-реактивні двигуни (ВРД) не так поширені, як універсальні двигуни, але в останні десятиліття до них звертають все більшу увагу. Це пов'язано з тим, що виробництво даних двигунів економить ресурси – в роторі немає обмоток, а в статорі міді використовуються на 30–40% менше, ніж в асинхронних двигунах. Також технологія виготовлення простіше. ВРД має в 4 рази меншу вартість, більш широкий діапазон швидкостей обертання, більш високий діапазон робочих температур, ніж вентильні двигуни з постійними магнітами. Однією з причин таких переваг є явнополісна конструкція цих

двигунів. Але це ж одночасно стає недоліком, так як електромеханічний перетворювач працює в імпульсному режимі, що призводить до акустичних шумів, а магнітне насичення ускладнює керування через нелінійні залежності.

Індуктивні гальма знаходять широке застосування в сфері вантажного і пасажирського транспорту, промисловості (вітрові генератори, підйомники), системах аварійного гальмування, а також в випробувальних стендах. У багатьох сучасних вантажних автомобілях і автобусах, крім основної гальмівної системи можна зустріти додаткову систему гальмування (гірське гальмо), що використовує індукційну гальмівну систему (ретардер). Принцип роботи такого гальма заснований на розсіюванні механічної енергії обертання в тепло за рахунок генерації вихрових струмів. Вихрові струми виникають в масивному металевому провіднику при його русі в магнітному полі, яке створюється котушками. Ретардер складається з статора і пари роторів, жорстко з'єднаних між собою. Статор виготовлений зі спеціального провідного матеріалу і грає роль індуктора. Він являє собою послідовно з'єднані електромагніти, в яких при виникненні електричного струму створюється потужне електромагнітне поле та при досягненні відповідного значення насичується магнітна система ретардера.

Дробово-інтегральне числення має довгу історію розвитку (з кінця XVII століття), але найбільш активний розвиток припав на останні 30–35 років. Це безпосередньо пов'язано з розвитком обчислювальної техніки, так як для розрахунків в цій області потрібно оперувати великими масивами даних. Але з появою різних методів апроксимацій розрахунку дробового інтеграла дозволяє використовувати регулятори з таким порядком в швидкодіючих системах. Це дає можливість поліпшити систему керування на основі уточнених моделей з урахуванням насичення магнітної системи за допомогою регуляторів з дробовим порядком.

Тому підвищення якості управління в динамічних режимах замкнених систем з метою приросту продуктивності і ефективності в електромеханічних системах є актуальною.

Зв'язок роботи з науковими програмами, планами, темами. Дисертаційна робота виконана на кафедрі електромеханічної інженерії Одеського національного політехнічного університету. Тематика роботи відповідає Закону України "Про пріоритетні напрямки розвитку науки і техніки", зокрема, п. 6 статті 7 "Нові технології та ресурсозберігаючі технології в енергетиці, промисловості та агропромисловому комплексі". Автор брав участь як виконавець в науково-дослідних робіт кафедри: №165-52 (номер державної реєстрації 0119U000466) «Розробка і дослідження електромеханічних систем автоматизації з поліпшеними динамічними та енергетичними характеристиками»; №94-52 «Розробка і дослідження електромеханічних систем робочих машин промислового та побутового призначення для поліпшення енергетичних показників, усталених та перехідних режимів».

На підставі вищевикладеного сформульовано мету та завдання роботи.

Мета роботи – синтез замкнутих систем керування електричними машинами, математичні моделі яких описуються передавальними функціями з дробовим порядком, спрямований на оптимізацію динамічних режимів шляхом підвищення якості з мінімумом перерегулювання та найменшим часом першого узгодження за допомогою дробово-інтегральних регуляторів.

Для досягнення поставленої мети необхідно вирішити такі **завдання**.

1. Провести експериментальні дослідження двигуна послідовного збудження і знайти адекватну математичну модель контуру струму, засновану на дробово-диференціальному рівнянні. За отриманою моделлю синтезувати регулятор для забезпечення оптимальних показників динамічного режиму з мінімумом перерегулювання та часом першого узгодження.

2. Провести моделювання процесів пуску вентильно-реактивного двигуна з урахуванням нелінійної залежності потокозчеплення та електромагнітного моменту з різними значеннями напруги живлення. За отриманими даними апроксимувати даний перехідний процес дробово-аперіодичною функцією. Синтезувати регулятор, який дозволить отримати найменше перерегулювання та час першого узгодження перехідного процесу.
3. Розробити систему керування двома електромагнітними гальмами в стенді для прецизійного налаштування паливної системи двигуна внутрішнього згоряння автомобілів. На підставі математичної моделі, отриманої методами ідентифікації параметрів об'єкта за експериментальними даними, налаштувати регулятор таким чином, щоб отримати найкращу динамічну точність з мінімальним перерегулюванням.
4. Дати огляд отриманих дробово-інтегральних регуляторів, оцінити їх вплив на динамічні режими в системі керування трьома різними електричними машинами, узагальнити отримані результати.

Об'єктом дослідження є динамічні процеси, які описуються диференціальними рівняннями дробового порядку, зокрема, процеси перетворення електричної і механічної енергії в двигунах постійного струму з послідовним збудженням, вентильно-реактивних двигунах і електромагнітних гальмах.

Предметом дослідження є синтез системи керування з дробовими інтегральними та диференціальними регуляторами для таких електричних машин, як двигун постійного струму з послідовним збудженням, вентильно-реактивний двигун та електромагнітне гальмо.

Методи дослідження.

Для вирішення поставлених в роботі завдань використано методи теорії автоматичного управління (вибір структури систем і визначення оптимальних показників), теорії електроприводу, методи чисельного рішення диференціальних рівнянь цілого і дробового порядку (для пошуку наближених рішень), методи

апроксимації, генетичні алгоритми ідентифікації параметрів динамічних об'єктів, методи чисельного рішення рівнянь електромагнітного поля вентильно-реактивного двигуна методом кінцевих елементів в двомірній постановці завдання.

Наукова новизна одержаних результатів полягає в наступному:

1. Вперше виконана ідентифікація якірного ланцюга двигуна послідовного збудження передавальною функцією з дробовим порядком, що в порівнянні з моделями цілочисельного порядку забезпечує мінімальну похибку при розрахунку перехідних процесів при струмі якоря вище номінального, що дозволило вибрати структуру регулятора з дробовим порядком астатизму 1.35 і отримати замкнену систему з найкращими динамічними і статичним показниками.
2. Отримав подальший розвиток метод ідентифікації контуру швидкості в вентильно-реактивному двигуні на базі дробово-аперіодичної функції при різних прикладених напругах з постійними кутами включення і виключення, що дозволило зменшити похибку ідентифікації, синтезувати регулятор з дробовим порядком, при якому отримано мінімальний час першого узгодження з перерегулювання не більше 2%.
3. Вперше синтезовано регулятор з дробовим порядком інтегрування в системі керування двома електромагнітними гальмами в стенді для налаштування паливної системи двигуна внутрішнього згоряння автомобілів, що дозволило забезпечити мінімальне перерегулювання з кращою динамічною та статичною точністю, внаслідок чого підвищується точність налаштування паливної системи та вимірювання максимальної потужності та моменту двигунів.

Практичне значення одержаних результатів:

1. Запропоновано модель і метод ідентифікації параметрів якірного ланцюга двигуна послідовного збудження з подальшим синтезом регуляторів, які враховують ефект насичення магнітної системи двигуна.
2. Запропоновано модель і метод ідентифікації параметрів контуру швидкості вентильно-реактивного двигуна при різних прикладених напругах з подальшим синтезом регуляторів з урахуванням нелінійних залежностей потокозчеплення та моменту від струму фази та положення ротора.
3. Створено систему керування на основі дробового регулятора в стенді налаштування паливної системи двигуна внутрішнього згорання, при яких перехідні процеси мали мінімум перерегулювання і більший порядок астатизму, що дозволяє відпрацьовувати сигнал завдання з точністю $\pm 0.2\%$.

Результати дисертаційної роботи використано в сервісному автосалоні, а саме в системі керування стендом вимірювання потужності (моменту) спортивних автомобілів та у навчальному процесі при викладенні дисциплін «Програмування мікропроцесорів і промислових контролерів для керування ЕМС та ЕТК», «Мікропроцесорна техніка та її програмування», «Моделювання ЕМС та пристроїв» на кафедрі електромеханічної інженерії

Особистий внесок здобувача.

Дисертаційна робота містить результати, які отримані самостійно. Серед них ідентифікація параметрів об'єктів управління, які описуються диференціальними рівняннями з дробовим порядком; синтез регуляторів з дробовим порядком інтегрування і диференціювання з астатизмом від 0 до 2; поліпшені моделі і структурні схеми регуляторів в системах керування двигуном постійного струму з послідовним збудженням, вентильно-реактивних двигунів і індукційних гальм.

У друкованих працях, опублікованих у співавторстві, дисертанту належать: в [1] – отримання нелінійних залежностей моменту і потокозчеплення від струму та положення ротора в ВРД для моделювання, а також створення модифікованого методу керування моментом; в [2] – випробування розрахунку дробових

інтегрально-диференційних регуляторів на процесорі STM32F429I; в [3] – проведення експериментів та за запропонованим математичним моделям ідентифікація параметрів якірного ланцюга двигуна постійного струму з подальшим синтезом регуляторів на різні налаштування; в [4] – моделювання з урахуванням залежності потоку двигуна від струму і застосування дробово-інтегральних регуляторів в системі керування; [5] – за моделлю в [1] апроксимація контуру швидкості ВРД при різних прикладених напругах дробово-аперіодичною передавальною функцією з подальшим синтезом регуляторів, в [6] – проведено попередні розрахунки для системи керування індукційними гальмами.

Роботи [7, 8] написано та опубліковано автором самостійно.

Апробація результатів роботи.

Результати дослідження та основні положення дисертаційної роботи доповідалися і обговорювалися на наступних семінарах, конференціях:

- міжнародна науково-практична конференція «Електротехнічні та комп'ютерні системи: теорія і практика» (ЕЛТЕКС-2017), Одеса, 26–28 червня, 2017 р.;
- VIII науково-методична конференція «Суднова електроінженерія, електроніка і автоматика», Одеса, 5–6 грудня, 2018 р.;
- XVII міжнародна науково-технічна конференція молодих вчених і спеціалістів «Електромеханічні та енергетичні системи. Методи моделювання та оптимізації », Кременчук, 11–12 квітня, 2019 р.;
- міжнародна науково-практична конференція «Електротехнічні та комп'ютерні системи: теорія і практика» (ELTECS-2019), Одеса, 24–26 червня, 2019 р.;
- міжнародна конференція «Сучасні електричні та енергетичні системи», Кременчук, 23–25 вересня, 2019 р.;

- XXIII міжнародна науково-практична онлайн-конференція "Theoretical and Practical Foundations of Social Process Management", Сан-Франциско, США, 29–30 червня 2020 р.;
- семінарах НАН України «Проблеми динаміки автоматизованих електромеханічних систем змінного струму», 2016–2019 рр.

Публікації.

За результатами досліджень опубліковано 8 статей та навчальний посібник, з них 4 – у спеціалізованих виданнях, рекомендованих департаментом атестації кадрів вищої кваліфікації, і 3 – у виданнях, які індексуються в наукометричних базах SCOPUS і WEB OF SCIENCE. Без співавторів опубліковано 2 наукові роботи.

Структура і обсяг дисертації.

Дисертаційна робота загальним обсягом 223 сторінок з основною частиною 144, яка складається зі вступу, чотирьох розділів, висновків, та анотації і списку використаних джерел з 101 найменування і 2 додатків на 79 сторінках; містить 93 рисунка і 16 таблиць.

РОЗДІЛ 1

ДРОБОВЕ ЧИСЛЕННЯ В СИСТЕМАХ КЕРУВАННЯ

1.1. Коротка історія розвитку дробового числення.

Перша історична згадка, пов'язане з дробовим численням, була трохи більше три століття тому. Тоді в листах між вченими Гійомом Франсуа Лопіталем і Готфрідом Вільгельмом Лейбніцем обговорювалася можливість порядку диференціювання не цілочисельного, а проміжного значення, яке дорівнює $\frac{1}{2}$ [1].

У 1738 році Л. Ейлер зауважив, що результату обчислення похідної $\frac{d^\beta y}{dx^\beta}$ від степеневі функції $y = x^m$ можна надати значення при нецілому β . Це помітно з серії диференціювання з порядком 1, 2, ... n :

$$\begin{aligned}\frac{dy}{dx} &= mx^{m-1}, \\ \frac{d^2 y}{dx^2} &= m(m-1)x^{m-2}, \\ &\dots \\ \frac{d^n y}{dx^n} &= m(m-1)\dots(m-n+1)x^{m-n} = \frac{m!}{(m-n)!}x^{m-n}.\end{aligned}\tag{1.1}$$

Після проміжку трохи менше 100 років, а точніше 1819 році з'являється публікація С. Лякруа, де повторена думка Ейлера і вже приведена явна формула обчислення похідної від степеневі функції $y = x^m$ [2]:

$$\frac{d^n y}{dx^n} = \frac{\Gamma(m+1)}{\Gamma(m-n+1)} x^{m-n},\tag{1.2}$$

де $\Gamma()$ – гамма функція и $m \geq n$.

У цій формулі використовується Гамма-функція, яка узагальнює факторіальну функцію на випадок нецілочисельних аргументів. Зокрема – для степеневій функції першого порядку $y = x$ та дробового диференціювання $n = \frac{1}{2}$, він отримав:

$$\frac{d^{\frac{1}{2}}x}{dx^{\frac{1}{2}}} = \frac{\Gamma(1+1)}{\Gamma\left(1-\frac{1}{2}+1\right)} x^{1-\frac{1}{2}} = \frac{\Gamma(2)}{\Gamma(1.5)} x^{1-\frac{1}{2}} = \frac{2\sqrt{x}}{\sqrt{\pi}}. \quad (1.3)$$

Через пару років в 1822 р. Ж. Фур'є вивів формулу розрахунку дробової похідної для майже будь-якій (не обмежуючись степеневою) функції [3]. Він отримав її, інтегруючи по частинах формулу зворотного перетворення Фур'є:

$$f(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(t) e^{ixt} dt. \quad (1.4)$$

В результаті:

$$f^n(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} (ix)^n F(t) e^{ixt} dt. \quad (1.5)$$

Замінивши цілочисельне значення n на дробове γ отримаємо формулу розрахунку дробової похідної:

$$f^{\gamma}(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} (ix)^{\gamma} F(t) e^{ixt} dt. \quad (1.6)$$

Всі перераховані вище події можна віднести до передісторії дробового числення. Одна з перших найбільш повних робіт до цього розділу математики вніс Ж. Ліувілл в 1832-1837 рр. Хоч подальші дослідження розвинули його ідеї, але вони дали необхідну основу. Його визначення дробового диференціала засноване на похідній степеневій функції [4], яка подається у вигляді $f(x) = \sum_{k=0}^{\infty} c_k e^{a_k x}$. Для цієї функції Ж. Ліувілл вивів свою першу (1.7) і другу (1.8) формулу дробової похідної:

$$D^{\gamma} f(x) = \sum_{k=0}^{\infty} c_k a_k^{\gamma} e^{a_k x}. \quad (1.7)$$

$$D^{\gamma} x^{-\alpha} = \frac{(-1)^{\gamma} \Gamma(\alpha + \gamma)}{\Gamma(\alpha)} x^{-\alpha-\gamma}. \quad (1.8)$$

Недолік цих двох визначень дробової похідної є те, що вони застосовуються до строгого класу функцій. Наприклад, формула (1.7) може бути застосовна до функцій, які можна розкласти в тригонометричні послідовності, а другу (1.8) – до функцій $x^{-\alpha}$ з $\alpha > 0$.

В інших його роботах [5–9] розширюється область понять, застосування дробового числення до геометрії, фізики, механіки і т. д. Серед отриманих результатів слід зробити акцент на ідею визначення дробової похідної через межу різницевого відношення $\Delta_h^{\gamma} f / h^{\lambda}$, де $\Delta_h^{\gamma} f$ – різниця дробового порядку. Це поняття буде використано іншими математиками.

Не менш важливими були роботи Б. Рімана в 1847 році. Вони полягали у використанні ряду Тейлора і за допомогою визначеного інтеграла, які були застосовані до рядів з нецілим порядком [10–11]. Завдяки цьому він прийшов до наступної форми дробового інтегрування:

$$I^{\gamma} f(x) = \frac{1}{\Gamma(\gamma)} \int_c^x (x-t)^{\gamma-1} f(t) dt + \Psi(x). \quad (1.9)$$

Поява додаткової функції $\Psi(x)$ викликало проблеми, які не могли вирішити ні Ріман ні Ліувуль. Але вони заклали основи для форми дробового інтегродиференціювання, яка була названа на їхню честь – *форма Рімана-Ліувілля*.

Першим кроком до цієї форми зробив Н. Я. Сонін в своїй статті 1869 році [12], де він використав інтегральну формулу Коші. А. В. Летніков також розширює цю ідею [13–14]. Однак були помилки з переходом до нецілочисленного значення, при якому інтеграл давав похибку. Вирішив цю проблему Лоран [15] в 1884 році і завдяки цьому з'явився дробовий інтеграл Рімана-Ліувілля:

$$D^{-\gamma} f(x) = \frac{1}{\Gamma(\gamma)} \int_c^x (x-t)^{\gamma-1} f(t) dt. \quad (1.10)$$

При нижній межі $c=0$ формула (1.10) стає ідентичною формулою Рімана (1.9), але без проблематичної додаткової функції $\Psi(x)$. При $c=-\infty$ формула (1.10) еквівалентна першій формі дробовій похідної Ліувілля (1.7). Ці два факти пояснюють чому формулу (1.10) стали називати формою дробового інтеграла *Рімана-Ліувілля*. Нижче представлена форма дробової похідної Рімана-Ліувілля (1.11).

$$D^\alpha f(x) = D^{n-\beta} f(x) = D^n f(x) D^{-\beta} f(x) = \frac{d^n}{dx^n} \left(\frac{1}{\Gamma(\beta)} \int_c^x (x-t)^{\beta-1} f(t) dt \right). \quad (1.12)$$

У цій формулі n це найменше ціле число, але більше α з умовою $0 < n - \alpha < 1$.

У цей час майже одночасно Грюнвальд [16] і Летніков [13] вивели іншу формулу дробового диференціала, яку також часто використовують донині. Вирішивши позбутися від обмежень в формулах похідних Ліувілля, вони почали роботу над визначенням дробового диференціала, як межа різницевого відношення $\Delta_{\Delta t}^\gamma f / \Delta t^\gamma$:

$$D^\gamma f(x) = \lim_{\Delta t \rightarrow 0} \frac{\Delta_{\Delta t}^\gamma f}{\Delta t^\gamma} = \lim_{\Delta t \rightarrow 0} \frac{\sum_{k=0}^{\infty} (-1)^k \binom{\gamma}{k} f(x - k\Delta t)}{\Delta t^\gamma}, \quad (1.13)$$

де $\binom{\gamma}{k}$ – узагальнені біноміальні коефіцієнти.

Вираз (1.13) в наш час стали називати формою дробової похідної *Грюнвальда-Летнікова*. Дану форму стали в основному використовувати для чисельних методів розрахунку з обмеженням суми для апроксимації дробового диференціала.

Також слід згадати роботи О. Хевісайда [17], який використовував операційне числення для вирішення певних проблем в електромагнітній теорії. Ця діяльність пов'язана з дробовими похідними, так як він в формулах використовував показники ступеня p довільного порядку, але в основному $\sqrt{p}$ для вирішення інженерних завдань.

При досягненні кінця XIX століття можна було зупинитися, так як роботи Л. Ліувілля, Б. Рімана, А. Грюнвальда і А. В. Летнікова принесли фундаментальний внесок у розвиток дробового числення. Але варто згадати ще пару робіт, які були не настільки великі, але мають свою користь.

Одна з таких праць була публікація Г. Вейля [18] в 1917 році, яке визначає дробове інтегрування, пристосоване для періодичних функцій $\varphi(x)$:

$$\sum_{k=-\infty}^{\infty} \varphi_k e^{ikx}, \quad \varphi_k = \frac{1}{2\pi} \int_0^{2\pi} \varphi(t) dt, \quad \varphi_0 = 0, \quad (1.14)$$

яка реалізується у вигляді згортки

$$I_{\pm}^{\gamma} \varphi(x) = \frac{1}{2\pi} \int_0^{2\pi} \psi_{\pm}^{\gamma}(x-t) \varphi(t) dt, \quad (1.15)$$

з деякою спеціальною функцією $\psi_{\pm}^{\gamma}$. Він показав, що дробовий інтеграл (1.14) і (1.15) можна записати у вигляді

$$I_{+}^{\gamma} \varphi(x) = \frac{1}{\Gamma(\lambda)} \int_{-\infty}^x (x-t)^{\gamma-1} \varphi(t) dt, \quad I_{-}^{\gamma} \varphi(x) = \frac{1}{\Gamma(\lambda)} \int_x^{\infty} (t-x)^{\gamma-1} \varphi(t) dt. \quad (1.16)$$

Так дробові інтеграли по нескінченному проміжку інтегрування (1.16) стали називатися інтегралами Вейля. Хоча назва обрана історично невдало, так як нескінченне інтегрування на інтервалі визначено і в формі Рімана-Ліувілля (1.10).

Пізніше, в 1927 А. Маршо [19], розробив інтегральну версію форми дробової похідної Грюнвальда-Летнікова:

$$D^{\gamma} f(x) = c \int_0^{\infty} \frac{(\Delta_t^l f)(x)}{t^{1+\gamma}} dt, \quad \gamma > 0, \quad (1.17)$$

де $(\Delta_t^l f)(x)$ – кінцева різниця порядку $l > \gamma$, $l = 1, 2, \dots, c$ – деяка константа.

Ця форма збігається з ліувівівською формою:

$$D^{\gamma} f(x) = \frac{1}{\Gamma(\gamma - n)} \frac{d^n}{dx^n} \int_{-\infty}^x \frac{f(t) dt}{(x - t)^{\gamma - n - 1}}, \quad n = \gamma + 1. \quad (1.18)$$

Але головна відмінність (1.17) в тому, що вона може бути застосовна до функцій зі значно «поганою» поведінкою на нескінченності. Наприклад, для $f(x)$ допускається зростання на нескінченності (порядки менше γ), що неможливо в (1.18). Формулу (1.17) прийнято називати *дробовою похідною Маршо*.

Однією з істотних робіт в теорії дробового числення відносно недавнього часу можна віднести результати М. Капуто [20]. Одне з головних недоліків форми дробового диференціала Рімана-Ліувілля є те, що дробові диференціальні рівняння зі своїм оператором вимагає дещо «дивний» набір з початкових значень. Зокрема, значення деяких дробових інтегралів та похідних повинно бути уточнено в початковому часі для вирішення рівнянь з дробовими диференціалами. М. Капуто [21] переформулював визначення дробової похідної Рімана-Ліувілля для використання більш «класичних» початкових значень. Для даної функції $f(x)$ з порядком $(n - 1)$ цілочисельних безперервних похідних, М. Капуто отримав наступну форму дробового диференціала:

$$D_*^{\gamma} f(x) = \frac{1}{\Gamma(n - \gamma)} \int_0^t (t - s)^{n - \gamma - 1} \frac{d^n}{dx^n} f(s) ds. \quad (1.19)$$

Формулу (1.19) стали називати дробовою похідною Капуто, яка широко використовується донині.

Математична область інтегралів і похідних з дробовим ступенем настільки розширилася, що в другій половині ХХ століття в 1974 році в Нью Хейвені була проведена перша конференція присвячена дробовому численню [22]. У тому ж році була опублікована перша книга Олдхема і Спаньє по дробовому численню [23]. З тих пір з'явилося кілька додаткових книжок, найбільш популярні з яких Міллер і Росс [24], Самко [25] і Подлубний [26]. У 1998 році вийшов перший номер математичного журналу "Дробове числення та прикладний аналіз". Цей журнал присвячений виключно темам теорії дробового числення і її додатків. В 2004 році в Бордо була проведена велика конференція «Дробове диференціювання та її застосування», де було проведено не менше 104 доповідей в області фракційного обчислення.

Все вищесказане є лише найбільш коротким описом історії розвитку дробового числення, в процесі його становлення було написано значну кількість публікацій. Але все ж основними формулами для розрахунку дробових інтегро-диференціалів вважають форми Рімана-Ліувілля та Грюнвальда-Летнікова. Перша форма використовується для аналітичного рішення, а друга – при численних розрахунках. Безліч інших форм дробових інтегро-диференціалів мають свої особливості і в деяких випадках спрощують процедуру розрахунку або мають свої переваги (наприклад, вищезгадані форми Вейля та Капуто).

1.2. Области застосування дробового числення.

З розвитком теорії дробового числення публікувалися безліч робіт (за останні два десятиліття понад три тисячі), пов'язані з рішенням різних наукових і інженерних проблем із застосуванням цього математичного апарату [27]. Ці роботи пов'язані з наступними областями науки і математичного моделювання [28, 29]:

- Зворотні механічні проблеми.

- Стохастична кінетика і динамічний хаос.
- Рух в в'язкої рідини.
- Поширення теплового потоку.
- Електрохімія електродів.
- Динаміка турбулентного середовища.
- Фільтрація через пористе середовище.
- Реологія в'язкопружних матеріалів.
- Електротехніка та радіотехніка.
- Фізика плазми.
- Квантова оптика і нанофізика.
- Лазерне охолодження атомів.
- Астрофізика і космологія.
- Біофізика та медицина.

Наявність дробово-часової похідної в рівнянні інтерпретується як спеціальна властивість даного процесу, яка називається пам'яттю або наслідком. Дробові похідні по координатах відображають середу з неоднорідностями якогось особливого виду, званого самоподібними неоднорідностями або фракталами [30]. Ми зустрічаємо такі структури в турбулентних потоках, плазмі та міжзоряних середовищах.

Для прикладу розглянемо окремі застосування дробового числення в фундаментальних законах фізики. У 1991 р С. Вестерлунд [31] запропонував використовувати дробові похідні для опису процесу поширення плоских електромагнітних хвиль в ізотропній однорідній діелектричній середовищі з втратами. Рівняння для напруженості електричного поля в одновимірному випадку має вигляд:

$$\mu_0 \varepsilon_0 \frac{d^2 E}{dt^2} + \mu_0 \varepsilon_0 \chi_0 E^{(\gamma)} + \frac{d^2 E}{dx^2} = 0, \quad (1.20)$$

де E – напруженість електричного поля; $\mu_0 \varepsilon_0 \chi_0$ – константи; $E^{(\gamma)}$ – дробова похідна напруженості електричного поля за часом порядку γ ($0 < \gamma < 2$).

Пізніше С. Вестерлундом [32] було запропоновано в рівняннях Максвелла співвідношення $D = \varepsilon E$ та $B = \mu H$ (D – електричне зміщення, B – магнітна індукція) замінити їх на дробовими узагальненнями: $D = \varepsilon E^{(\gamma-1)}$ і $B = \mu H^{(\gamma-1)}$, що фактично запроваджувала в систему рівнянь Максвелла інтеграли дробового порядку від E і H при $0 < \gamma < 1$.

М. Капуто в 1993 р. в роботі по дослідженню електролітично поляризованому середовищу [33] запропонував більш загальне вираження для залежності між D і E :

$$\gamma D^{(\gamma)} + \alpha D = \sigma E + \varepsilon E^{(\gamma)}, \quad (1.21)$$

де $\gamma, \alpha, \sigma, \varepsilon$ – константи, γ – нецілий порядок диференціювання за часом.

Це дозволило при деяких припущеннях звести рівняння Максвелла в одновимірному (по простору) випадку до наступної системи двох рівнянь:

$$\begin{cases} \frac{d^2 E}{dx^2} = -\mu \frac{d^2 D}{dt^2}, \\ \gamma D^{(\gamma)} + \alpha D = \sigma E + \varepsilon E^{(\gamma)}. \end{cases} \quad (1.22)$$

Також С. Вестерлунд [32] дав узагальнення другого закону Ньютона і показав, що закон Гука в теорії пружності ($F = kx$), Ньютоновская модель в'язкої рідини ($F = kx^{(1)}$) і другий закон Ньютона ($F = kx^{(2)}$) можуть розглядатися як окремі випадки більш загального співвідношення виду: $F = kx^{(\gamma)}$, де порядок похідної γ може бути будь-яким дійсним числом.

Дробове числення можна застосувати та в теорії відносності Ейнштейна і пов'язаний з нею відомий «парадокс близнюків». Відомому співвідношенню теорії відносності $E = mc^2$ зіставляється вираз для каузальної моделі маси, яка є функцією часу: $\frac{d^{-2\gamma}W(t)}{dt^{-2\gamma}} = m(t)c_0^2$, де $0 < \gamma < 1$. Таким чином, доповненням до рівнянь теорії відносності є, по суті, зв'язок маси з дробовим інтегралом енергії [34]. Вестерлунду належить також крилата фраза: «Нежива матерія має пам'ять. Інакше кажучи, ми можемо сказати, що Природа працює з дробовими похідними за часом» [31].

Як ми бачимо дробове числення можна застосувати в фундаментальних законах фізики, не говорячи, що існують тисячі публікацій в вузькопрофільних областях.

1.3. Дробові регулятори в системах керування.

Застосування дробового числення в автоматичному керуванні можна поділити на дві групи [35, 36]. Першу утворюють методи математичного та комп'ютерного моделювання систем дробового порядку [37], в яких виявляються властивості дробової динаміки (зазвичай це пов'язано з наявністю у сигналів степеневі залежності в часовій та (або) частотній областях). До другої відносяться методи використання дробового числення для синтезу систем керування динамічними системами як цілого, так і дробового порядків, зокрема, синтезу контролерів нецілого порядку.

Математична модель лінійної динамічної системи з постійними параметрами дробового порядку в разі однієї змінної має вигляд:

$$a_0 D^{\gamma_0} y(t) + a_1 D^{\gamma_1} y(t) + \dots + a_n D^{\gamma_n} y(t) = b_0 D^{\beta_0} u(t) + b_1 D^{\beta_1} u(t) + \dots + b_m D^{\beta_m} u(t), \quad (1.23)$$

де: a_i, b_j – коефіцієнти рівняння, γ_i, β_j – дробові порядки диференціальних операторів, $y(t)$ – функція виходу динамічної системи (функція стану), $u(t)$ – функція входу динамічної системи (функція управління).

Перетворення Лапласа, яка для дробового порядку може бути визначена наступним чином:

$$L\{D^\gamma f(t)\} = s^\gamma F(s). \quad (1.24)$$

У разі нульових початкових умов передавальна характеристика динамічної системи в області перетворення по Лапласу має вигляд:

$$G(s) = \frac{U(s)}{Y(s)} = \frac{b_m s^{\beta_m} + \dots + b_1 s^{\beta_1} + b_0 s^{\beta_0}}{a_n s^{\gamma_n} + \dots + a_1 s^{\gamma_1} + a_0 s^{\gamma_0}}. \quad (1.25)$$

Рішення класичних проблем теорії автоматичного управління (стійкість, робастність і т. д.) в разі систем дробового порядку виявляються схожими з цілочисельними.

У теорії автоматичного керування широко поширені і застосовуються елементи системи управління, що отримали назву PID-регуляторів. Вони являють собою сукупність пропорційних, інтегруючих і диференціальних ланок.

Математична модель класичного PID-регулятора є зваженою сумою вхідного сигналу, його інтеграла і похідною:

$$u(t) = K_p \cdot e(t) + T_i \cdot \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt}, \quad (1.26)$$

де $e(t), u(t)$ – вхідний і вихідний сигнал, відповідно; T_p, T_i, T_d – коефіцієнти пропорційної, інтегральної та диференціальної частини регулятора, відповідно. Передавальна функція контролера при перетворенні за Лапласом має вигляд:

$$G(s) = \frac{U(s)}{E(s)} = K_p + T_i \cdot s^{-1} + T_d \cdot s, \quad (1.27)$$

де s – оператор Лапласа.

У найбільш простому представленні контур системи управління з такими регуляторами можна показати у вигляді двох ланок з передавальною функцією регулятора і об'єкта управління (plant) охоплених одиничним негативним зворотним зв'язком (рис. 1.1).

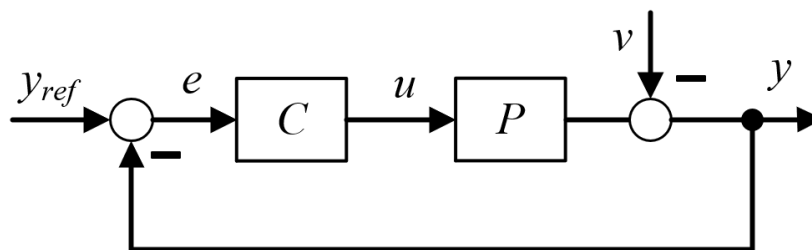


Рис. 1.1. Система керування із замкнутим контуром і регулятором, де y_r, e, u, v, y – сигнали завдання, помилки, управління від регулятора, збурення та вихідна координати, відповідно

У 1989 році було проведено опитування [38] Японською асоціацією виробників приладів і згідно їх результатами понад 90 відсотків контурів управління були регулятори типу PID. В роботі [39] розглянута паперова фабрика в Канаді, яка має понад 2000 контурів і при цьому 97 відсотків використовують регулятори з типом PI. Завдяки цим даним можна зробити висновок про те, що в переважній більшості в промислових контурах управління використовується PI /

PID регулятори. Наявність обширної теорії [40, 41] для налаштування даних регуляторів з необхідними якостями перехідних процесів робить їх як найбільш очевидним вибором при синтезі системи управління технологічним процесом.

Зрозуміло, що при використанні диференціала та інтеграла в (1.27) з дробовим порядком дозволить отримати більш гнучкий в налаштуванні регулятор. Такі регулятори позначаються аббревіатурою $PI^\gamma D^\mu$ з інтегралом і похідною порядку γ та μ відповідно, їх математичний опис від часу має вигляд:

$$u(t) = K_p \cdot e(t) + T_i \cdot D_t^{-\gamma} e(t) + T_d \cdot D_t^\mu e(t). \quad (1.28)$$

Передавальна функція дробового регулятора при перетворенні Лапласа має вигляд:

$$G(s) = \frac{U(s)}{E(s)} = K_p + T_i \cdot s^{-\gamma} + T_d \cdot s^\mu. \quad (1.29)$$

Додаткова свобода налаштування регулятора у вигляді дробового порядку інтеграла та диференціала дозволяє отримати більш широкий діапазон бажаних перехідних процесів на відміну від цілочисельного PID. Тобто дробовий $PI^\gamma D^\mu$ -регулятор може замінити контролер зі значно вищим цілочисельним порядком при цьому зберігаючи аналогічні якості перехідного процесу [42–44].

Може здатися, що причиною такого досягнення в прирості продуктивності дробових $PI^\gamma D^\mu$ -регуляторів є додаткові параметри для налаштування. Але насправді, відповідно до теорії, $PI^\gamma D^\mu$ -регулятор є нескінченно-розмірним фільтром через дробовий порядок похідної та інтеграла [45].

На даний момент немає конкретної методики для розрахунку параметрів для $PI^\gamma D^\mu$ -регулятора, на відміну від традиційного PID. Але можна скористатися простою схемою підбору параметрів. Вона заснована на розподілі полюсів

характеристичного рівняння системи управління в комплексній площині, які повинні бути розташовані в певних областях відповідно до вимоги якості перехідних процесів вихідній координаті [46, 47]. Припустимо, що головна пара комплексно-сполучених коренів має вигляд:

$$p_{1,2} = -r \pm j\omega. \quad (1.30)$$

Дані корені характеристичного рівняння системи управління безпосередньо впливають на стабільність S_t і коефіцієнт демпфірування ξ . У цій спрощеній ситуації підбір параметрів для $PI^{\gamma}D^{\mu}$ -регулятора можна розділити на два етапи.

По-перше, потрібно підібрати коефіцієнт посилення пропорційної частина регулятора K_p . Цей параметр впливає на статичну похибку δ [%], час регулювання t_r [s] та перерегулювання σ_r [%]. У загальному випадку, чим більше підсилення K_p , тим менше час регулювання t_r , як і статична помилка δ . Отже, пошук K_p може бути спрощений через вираз:

$$K_p \geq (100/\delta). \quad (1.31)$$

По-друге, необхідно підібрати параметри порядку дробового інтеграла γ та похідною μ , також їх коефіцієнти T_i і T_d . З парних комплексно-спряжених коренів (1.30) можна розрахувати стабільність через $S_t = r$ та демпфірування $\xi = r/\omega$. З отриманими S_t і ξ , використовуючи метод кореневих локусів (root locus method), можна чисельно розрахувати T_i, γ, T_d, μ з характеристичного рівняння з регулятором дробового порядку $C(s)$, що визначається таким чином:

$$C(s)P_0(s)+1=0, \quad (1.32)$$

де $P_0(s)$ – номінальна модель $P(s)$ (рис. 1.1).

Крім вищезгаданого $PI^{\gamma}D^{\mu}$ -регулятора в теорії автоматичного керування також застосовуються інші типи регуляторів з дробовим порядком – TID-регулятор, CRONE-регулятор та fractional lead-lag compensator.

TID-регулятор [48] аналогічний PID за винятком того, що у нього замінені пропорційна складова на передавальну функцію $s^{-\frac{1}{n}}$ (рис. 1.2), де n – це не нульове дійсне число, зазвичай в межах між 2 і 3.

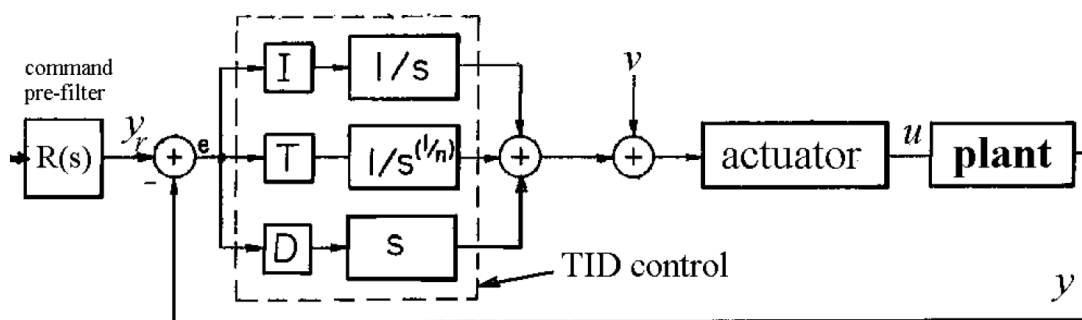


Рис. 1.2. Структурна схема TID-регулятора [48]

Дана заміна дозволяє домогтися більш кращого перехідного процесу, так як результуюча ЛАЧХ ближче до ідеальної Бode (рис. 1.3). В цьому випадку регулятор краще апроксимує оптимальну передавальну функцію, отже, поліпшується управління зі зворотним зв'язком.

Так як складова з $s^{-\frac{1}{n}}$ усуває статичну помилку, тому інтегральна складова при s^{-1} може бути встановлена в нуль для деяких об'єктів керування, що спростить регулятори. Пропонована процедура налаштування TID-регулятора полягає в наступному:

1. Встановити $T_i = 0, T_d = 0$ і коефіцієнт K_T для контуру з посиленням 0 dB на бажаній частоті зрізу f_c .
2. Встановити коефіцієнту T_d таке значення, при якому запас стійкості по фазі буде на 5 градусів більше, ніж необхідно.
3. Встановити коефіцієнт $T_i = 0.25K_T f_c^{(1-1/n)}$.

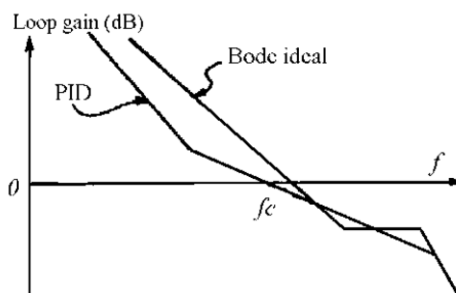


Рис. 1.3. ЛАЧХ системи управління з PID-регулятором та ідеальна Боде [46]

У порівнянні з традиційним PID, TID-регулятор дозволяє спростити налаштування параметрів, покращити коефіцієнт відхилення перешкод (disturbance rejection ratio) і менший вплив зміни параметрів об'єкта керування на перехідний процес.

CRONE-регулятор був запропонований Оусталоупом в спробі досягти «фрактальної робастності» (fractal robustness) [49]. CRONE походить від французької аббревіатури "Contr'ole Robuste d'Ordre Non Entier" (що перекладається як робастне управління з нецілочисельним порядком). В [50, 51] «фрактальна робастність» використовується для опису наступних двох характеристик: ізодемпінга і вертикальної ковзаючої форми частотного шаблону на діаграмі Ніколса. Ця бажана робастність мотивувала використання контролера дробового порядку в класичних системах управління для підвищення їх продуктивності.

Дробовий регулятор випередження (Fractional Lead-Lag Compensator) заснований на своїй цілочисельній версії, але вже з використанням дробового числення [52]. Такий регулятор розраховується за формулою:

$$C_r(s) = C_0 \left(\frac{1 + s/\omega_b}{1 + s/\omega_h} \right)^r, \quad (1.33)$$

де $0 < \omega_b < \omega_h, C_0 > 0$ и $r \in (0,1)$.

За аналогією зі звичайним PID-регулятором необхідно знайти таку передавальну функцію $C(s)$ (рис. 1.1), яка гарантуватиме робастний Q-фактор (коефіцієнт збільшення амплітуди на резонансній частоті) для передавальної функції від y_y до y та від v до y для об'єктів управління $P = \alpha P_0$, P_0 де – номінальна модель об'єкта управління і $\alpha \in [\alpha_m, \alpha_M]$. Ідеальним рішенням буде зробити номінальну передавальну функцію контуру, яка дорівнює:

$$L_0(s) = C(s)P_0(s) = \left(\frac{s}{\omega_0} \right)^n, \quad (1.34)$$

де $1 < n < 2$.

Отже, запас по фазі та добротність (Q-фактор) для замкнутого контуру передавальної функції від y_r до y і від v до y буде не залежати від α , а буде визначатися n . Крім того, так як передавальна функція контуру може бути апроксимована ланкою другого порядку, то перерегулювання перехідного процесу (викликаним стрибком завдання) практично не залежить від α . Отже, α буде впливати на повільне або швидке загасання сигналу, але з однаковим демпфуванням. Така робастність (стійкість) бажана для деяких систем.

Перераховані вище методи показують, що дробове числення розширює область налаштувань регуляторів за рахунок дробового порядку диференціала і інтеграла. Це дозволяє більш тонше налаштовувати їх значення, тим самим отримуючи найбільш прийнятну якість перехідних процесів (ПП), які могли б отримати лише в тому випадку якби регулятор був би більш високого порядку. Наприклад, для $PI^{\nu}D^{\mu}$ -регулятора можна підбирати 5 значень. Три з яких – це коефіцієнти, а два – дробовий порядок похідної та інтеграла. Але подібна свобода вибору має свою ціну – ускладнений пошук параметрів регулятора для перехідного процесу з бажаними якостями. Тому якщо необхідно швидко налаштувати регулятор, то краще скористатися TID-регулятором. Тобто при виборі регулятора потрібно зробити вибір – швидкий пошук параметрів і низьку якість ПП або повільний пошук параметрів і високу якість ПП. CRONE-регулятор і регулятор на випередження дають схожі результати із застосуванням $PI^{\nu}D^{\mu}$ -регулятора. Для пошуку коефіцієнтів регуляторів також можуть використовувати оптимізаційні алгоритми, наприклад, генетичний алгоритм, метод рою частинок і т.д.

1.4. Підлегле регулювання координат з налаштуванням на дробовий порядок астатизму.

Одним з методів щодо спрощення пошуку потрібної структури регулятора і його параметрів є корекція за допомогою підлеглого регулювання. На даний момент найбільш поширені налаштування є модульний (технічний) і симетричний оптимум. Сенс даних налаштувань для кожної координати полягає в тому, щоб зменшити час першого узгодження перехідного процесу необхідно компенсувати вплив великих постійних часу, а для усунення статичної помилки вводять інтегруючі ланки, від яких залежить коливальність замкнутої системи керування. Малі постійні часу не компенсуються, так як вони істотно не впливають на якість ПП і помітно обмежують високочастотні перешкоди. Таким

чином, мета даної корекції полягає в тому, щоб включити в прямий тракт послідовно коригуючу ланку – регулятор (аналогічно PID), структуру і параметри якого визначаємо виходячи з того, щоб вихідний контур мав астатизм першого порядку (модульний оптимум) і при цьому забезпечував високу швидкодію. Для модульного оптимуму в розімкнутому стані його передавальна функція повинна мати вигляд:

$$W_{op.}^1(s) = \frac{1}{2T_v s \cdot (T_v s + 1)} = W_{co}(s)W_{reg}(s), \quad (1.35)$$

де T_v – мала постійна часу, s ; $W_{co}(s)$ – передавальна функція об'єкта керування; $W_{reg}(s)$ – передавальна функція регулятора.

У разі симетричного оптимуму замкнутий контур має астатизм другого порядку і в розімкнутому стані має вигляд передавальної функції:

$$W_{op.}^2(s) = \frac{4T_v s + 1}{8T_v^2 s^2 \cdot (T_v s + 1)} = W_{co}(s)W_{reg}(s). \quad (1.36)$$

Розвиток дробового числення дозволило розширити значення порядку астатизму на нецілочисельний порядок. Це в свою чергу дало початок нового налаштування на дробовий порядок астатизму [53]. Така система повинна в розімкнутому стані мати такі передавальні функції:

$$W_{op.}^\mu(s) = \frac{1}{aT_v^\mu s^\mu \cdot (T_v s + 1)}, 0 < \mu < 1 \quad (1.37)$$

$$W_{op.}^{1+\mu}(s) = \frac{1}{aT_v^{\mu-1} s^{\mu-1}} \frac{bT_v s + 1}{bT_v s} \frac{1}{(T_v s + 1)}, 1 < \mu < 2, b > 1 \quad (1.38)$$

де a, b – параметри налаштування, T_v – мала постійна часу.

Порівняння логарифмічної амплітудно-частотної та логарифмічної фазо-частотної характеристики при однакових частотах зрізу (рис. 1.4) на модульний або симетричний оптимум показує, що запас стійкості по фазі при зменшенні μ збільшується на величину $(1-\mu)\frac{\pi}{2}$, а це знижує коливальність в системі, що дає можливість збільшити частоту зрізу та прискорити швидкодію замкнутого контуру на початок перехідного процесу.

Для контуру з функцією передачі (1.37) було встановлені співвідношення [53] між параметрами a, b, μ при яких перехідний процес характеризується максимально можливою швидкодією та з заданим допустимим перерегулюванням $\delta = 0.05$:

$$a \approx \frac{\mu}{4.683 - 5.897\mu + 1.595\mu^2}, 0.5 \leq \mu \leq 0.9 \quad (1.39)$$

та параметрів a, b для контуру з передаточною функцією (1.38):

$$\begin{cases} (ba) \approx \exp(-10.27 + 7.831\mu), \\ b \approx 7.336 + 0.792(ba) + 3.83\ln(ba). \end{cases} 1.3 \leq \mu \leq 1.9 \quad (1.40)$$

Дані залежності встановлені при пошуку екстремуму функції $F(a, b, \mu, \delta)$:

$$F = \sum_{i=1}^N \begin{cases} (1 - Y_i), & \text{де } Y_i \leq 1 \\ \left(\frac{1}{\delta} (1 - Y_i) \right)^2, & \text{де } Y_i > 1 \end{cases} \rightarrow \min, \quad (1.41)$$

де δ – допустиме перерегулювання.

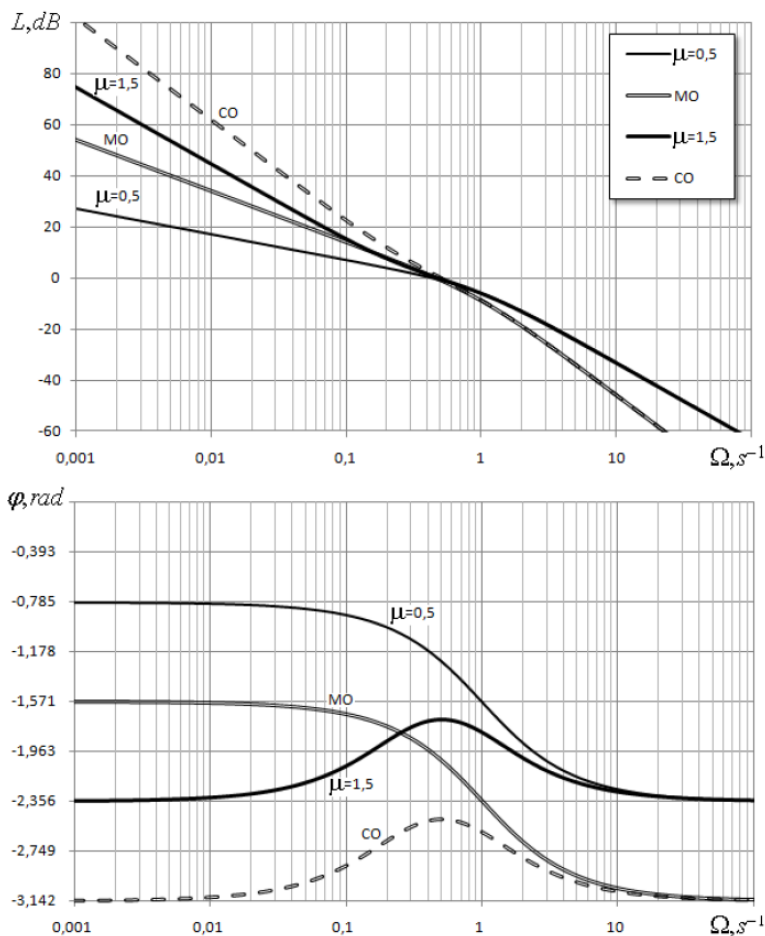


Рис. 1.4. ЛАЧХ і ЛФЧХ при однакових частотах зрізу [53]

Для забезпечення обраних налаштувань регулятор, включений послідовно з об'єктом керування, повинен компенсувати небажані властивості об'єкта. Загальна передавальна функція регулятора визначається зі співвідношення

$$W_{reg}(s) = \frac{W_{op}(s)}{W_{obj}(s)}.$$

Регулятор може складатися з дробових інтегральних, диференціальних, аперіодичних ланок і може бути використаний для об'єкта як з дробовим, так і з цілим порядком похідних у складі його математичної моделі. По відношенню до об'єкту проводять класифікацію:

- цілочисельний регулятор – цілочисельний об'єкт керування;
- регулятор з дробовим порядком – цілочисельний об'єкт керування;

- цілочисельний регулятор – об'єкт керування з дробовим порядком;
- регулятор з дробовим порядком – об'єкт керування з дробовим порядком.

При забезпеченні оптимальних налаштувань замкнутий контур характеризується передавальною функцією виду:

$$W_{op.close}^{\mu}(s) = \frac{1}{aT_v^{\mu+1}s^{\mu+1} + aT_v^{\mu}s^{\mu} + 1}, \quad 0 < \mu < 1, \quad (1.42)$$

$$W_{op.close}^{1+\mu}(s) = \frac{1}{baT_v^{\mu+1}s^{\mu+1} + baT_v^{\mu}s^{\mu} + bT_vs + 1}, \quad 1 < \mu < 2. \quad (1.43)$$

Для синтезу зовнішнього контуру передавальні функції (1.42) і (1.43) можуть бути спрощені на підставі аналізу частотних і перехідних характеристик. ЛАЧХ і ЛФЧХ (рис. 1.5) дозволяють визначити смугу пропускання контуру Ω_{bw} і постійну часу $T_{bw} = \frac{1}{\Omega_{bw}}$, коли $L \geq -3dB$ або $\varphi \geq -\frac{\pi}{4}$, а оцінка перехідних функцій дозволяє описати замкнутий контур інерційною ланкою першого порядку з одиничним коефіцієнтом і деякою постійною часу T_{eq} .

Під час налаштування зовнішнього контуру величина T_{eq} відповідає некомпенсованій постійної часу об'єкта керування.

Також є інший підхід до синтезу дробових регуляторів в підпорядкованих системах керувань. Наприклад, в роботі [54] запропоновані наступні передавальні функції при яких система в розімкненому стані має вигляд:

$$W_{d.1}(s) = \frac{\omega_{oc}}{s^q + \omega_{oc}}, \quad (1.44)$$

або

$$W_{d.2}(s) = \frac{\omega_{oc}^q}{(s + \omega_{oc})^q}, \quad (1.45)$$

де q – дробове число, ω_{oc} – бажана частота зрізу.

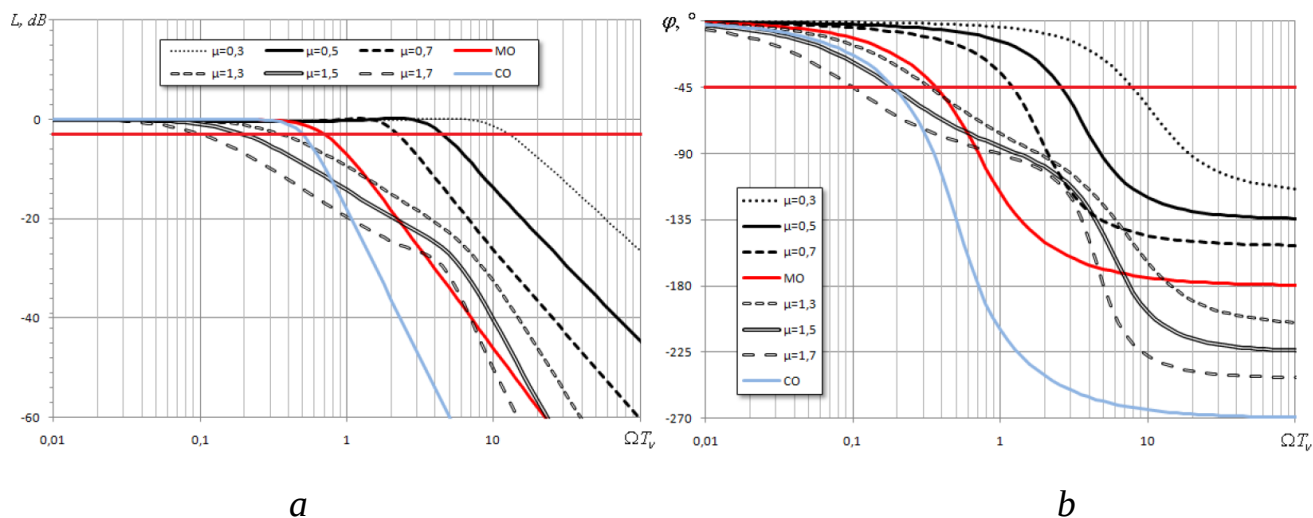


Рис. 1.5. ЛАЧХ (а) та ЛФЧХ (b) налаштованого контуру [53]

1.5 Апроксимаційні методи розрахунку дробових похідних та інтегралів

Незважаючи на переваги, одержувані від використання дробового числення в системах керування, їх реалізація є проблематичною. Це пов'язано з тим, що з теоретичної точки зору оператор s^α , з дробовим порядком α , можна уявити як передавальну функцію з цілочисельним нескінченним порядком. Тому на практиці в основному використовують наближені рішення в обраному частотному діапазоні. Дані методи можна розділити по виду реалізації: аналогові і цифрові [55]. Як виходить з назви при аналоговій реалізації [56] використовуються операційні підсилювачі з різними R, L, C комбінаціями. Цифрові ж безпосередньо використовуються при розрахунках на мікроконтролерах (МК).

На даний момент існує досить багато методик для отримання апроксимуючої передавальної функції, але неможливо сказати яка з них краще. Так як одні з них показують хороші результати на певних характеристиках, відносні переваги кожного методу залежать від порядку апроксимації, вибір де буде менше похибка в часовому діапазоні або частотному, наскільки трудомістким буде розрахунок і т. д.

Апроксимації можуть застосовуватися для отримання функцій в безперервній s -області або дискретної z -області [57]. Тому розрізняють непряму дискретизацію, коли отриману передавальну функцію переводять через z -перетворення, і пряму дискретизацію, коли її відразу реалізують на МК.

Один з найбільш популярних методів апроксимації дробового порядку, який належить до непрямої дискретизації, заснований на Осталоуповському рекурсивному апроксимаційному алгоритмі [58, 59]. Він заснований на функції такої форми:

$$W(s) = s^\gamma, \quad \gamma \in R, \quad \gamma \in [-1; 1]. \quad (1.46)$$

Для обраного частотного діапазону $[\omega_l; \omega_h]$, передавальна функція фільтра має вигляд:

$$W_f(s) = K \prod_{k=1}^N \frac{s + \omega'_k}{s + \omega_k}. \quad (1.47)$$

Нулі ω'_k , полюса ω_k та коефіцієнт посилення розраховуються за формулами:

$$\begin{aligned}
\omega'_k &= \omega_l \cdot \omega_m^{\frac{2k-1-\lambda}{N}}, \\
\omega_k &= \omega_l \cdot \omega_m^{\frac{2k-1+\lambda}{N}}, \\
K &= \omega_h^\gamma,
\end{aligned} \tag{1.48}$$

де ω_h, ω_l – високі та низькі частоти і $\omega_m = \sqrt{\frac{\omega_h}{\omega_l}}$.

Кількість нулів та полюсів N впливає на точність апроксимації і трудомісткість обчислень. Чим більше N , тим точніше обчислення дробової похідної або інтеграла, але потрібно більше часу на виконання операції і навпаки – чим менше N , тим збільшується похибка і на характеристиках ЛАЧХ і ЛФЧХ з'являються пульсації, але обчислення виконуються за менший час.

Апроксимація, заснована на методі Карлсона [60], виходить від рівняння передавальної функції:

$$W(s) = s^\gamma \Rightarrow W^{\frac{1}{\gamma}}(s) = s. \tag{1.49}$$

Використовуючи ітеративний метод Ньютона, вирішуємо рівняння (1.49).

Слід зазначити, що рішення справедливо тільки для $\frac{1}{\gamma} \in \mathbb{Z}$, тобто тільки для цілих

чисел, тому γ може бути рівним $\pm\frac{1}{2}, \pm\frac{1}{3}, \pm\frac{1}{4} \dots$ і так далі. В результаті

отримуємо наступну послідовність апроксимуючих передавальних функцій $F_i(s)$:

$$F_i(s) = F_{i-1}(s) \frac{\left(\frac{1}{\gamma} - 1\right) F_{i-1}^{\frac{1}{\gamma}}(s) + \left(\frac{1}{\gamma} + 1\right) s}{\left(\frac{1}{\gamma} + 1\right) F_{i-1}^{\frac{1}{\gamma}}(s) + \left(\frac{1}{\gamma} - 1\right) s}, \quad F_0(s) = 1, \quad i \in \mathbb{N}. \quad (1.50)$$

Кожна ітерація зменшує пульсації на ЛАЧХ і ЛФЧХ, також розширює діапазон частот апроксимації. Дану операцію можна виконувати до тих пір, поки не будуть отримані задовільні характеристики. За замовчуванням ЛАЧХ і ЛФЧХ відцентровані щодо частоти 1 rad / s. Це значення можна змінити, якщо помножити нулі і полюси в (1.50) на новий коефіцієнт для зсуву центру частот.

Безперервне розширення дробу (continued fraction expansion – CFE) часто розглядається як корисний тип наближення раціональної функції до заданої функції $W(s) = s^\gamma$. Він зазвичай має кращу збіжність, ніж функції степеневих рядів, такі як розкладання в ряд Тейлора. Для оператора дробового порядку γ розкладання по неперервному дробу CFE можна записати у вигляді:

$$W(s) = \frac{b_0(s)}{a_0(s) + \frac{b_1(s)}{a_1(s) + \frac{b_2(s)}{a_2(s) + \dots}}}, \quad (1.51)$$

де $a_i(s)$ та $b_i(s)$ можуть бути виражені через раціональну функцію s .

Щоб знайти апроксимаційну функцію за цим методом спочатку потрібно розбити вихідний дробовий оператор по CFE, а потім отримати передавальну функцію цілочисельного порядку.

Ключовим способом реалізації на МК апроксимаційних функцій для дробового оператора s^γ є дискретна передавальна функція змінної z . Як було вже вище сказано поділяють пряму дискретизацію і непряму. Існуючі методи з прямою

дискретизацією можуть використовувати застосування прямого розкладання в степеневі ряди (power series expansion) оператора Ейлера [61], безперервне розкладання дробу (CFE) оператора Тустіна [62, 63]. Комбінований оператор Аль-Алауї [64] заснований на змішуванні операторів Ейлера і Тустіна. Цей метод також використовує безперервне розкладання дробів. Всі перераховані вище методи по дискретизації, отримують на виході фільтри з нескінченною імпульсною характеристикою (IIR filter). Також є роботи присвячені апроксимації дробового порядку за допомогою фільтра з кінцевою імпульсною характеристикою (FIR filter) [65]. Але такі фільтри мають більш високий порядок, тому менш ефективні з точки часу розрахунку. Також є підходи із застосуванням функції В-сплайнів [66] і матричний підхід Підлубного [67].

Перераховані вище методи засновані на апроксимації в заданій області частот, що дає деяку похибка в обчисленнях. Для більш детальних задач, наприклад симуляції фізичних процесів використовуються чисельні методи [68]. В [69] описуються такі розрахунки, як процеси, описувані рівняннями інтеграла Абеля, рівняння дробових дифузійних хвиль і т. д. Виходячи з форми дробової похідної Грюнвальда-Летникова (1.13) можна вивести формулу для розрахунку з постійним кроком:

$$D_k^\gamma f(t) \approx \frac{1}{\Delta t^\gamma} \sum_{i=0}^k K_i^\gamma f_{k-i}. \quad (1.52)$$

Однак в замкнутих системах керування домогтися астатизму дозволяє інтегральні складові регулятора, а диференційні складові з'являються лише в тому випадку, якщо об'єкту керування притаманне інерційність 2-го або вище порядку і потрібно їх компенсувати. Тому при реалізації на МК краще використовувати апроксимацію через дробовий порядок інтегрування. А якщо необхідно обчислити дробову похідну, то можна зробити через значення інтеграла:

$$D^{1-\mu} f = \frac{d}{dt} I^\mu f \approx \frac{I^\mu f_i - I^\mu f_{i-1}}{\Delta t}. \quad (1.53)$$

Для компенсації помилки при розрахунку дробового інтеграла в формі Рімана-Ліувілля можна ввести коригуюче зміщення в знаменнику. Це дозволить підвищити точність обчислення [70].

Часова характеристика ланки з дробовим порядком інтегрування μ , яка є реакцією на стрибок завдання $x(t) = 1$:

$$y(t) = \frac{1}{\Gamma(1+\mu)} t^\mu. \quad (1.54)$$

Тоді

$$\frac{1}{\Gamma(\mu)} \sum_{j=1}^i \frac{1}{(t_i - j\Delta t + C_j)^{1-\mu}} \Delta t = \frac{1}{\Gamma(1+\mu)} (i\Delta t)^\mu. \quad (1.55)$$

Після серії перетворення, значення дробового інтеграла:

$$I_{t_i}^\mu f(t) = \Delta t^\mu \sum_{j=1}^i \frac{(j^\mu - (j-1)^\mu)}{\Gamma(1+\mu)} f_{i-j+1} = \Delta t^\mu \sum_{j=1}^i k_j f_{i-j+1}, \quad (1.56)$$

де $k_j = \frac{(j^\mu - (j-1)^\mu)}{\Gamma(1+\mu)}$ – масив констант.

У виразі (1.56) через зсув в знаменнику, помилка була усунена без застосування коригувальних обчислень.

Однак, незважаючи на практично відсутню помилку при обчисленні дробового інтеграла, необхідно на кожному періоді квантування розраховувати суму попарного добутку коефіцієнтів k_j та попередніх значень вхідного сигналу. Це означає, що на МК необхідно зберігати вектор констант і історію зміни сигналу, що накладає додаткові витрати з боку пам'яті і часу для розрахунку дробового інтеграла. Максимальна кількість членів ряду визначається обсягом оперативної пам'яті. Якщо зменшити розмір масивів, то це призведе до втрати точності, але позначиться поліпшенням продуктивності МК. Необхідний розмір масиву (1.56) для розрахунку з допустимою відносною помилкою ε , визначається за виразом

$$N = \left(\frac{1}{\varepsilon} \frac{\Gamma(1+\mu)}{\Gamma(\mu) \Delta t^\mu} \right)^{\frac{1}{\mu}}. \quad (1.57)$$

Так, наприклад, для $\mu = 0.5$ і $\varepsilon = 0.05$ необхідно пам'ятати 315 точок, а для забезпечення $\varepsilon = 0.02$ – вже 1964. Зберігання такої кількості даних перевищує можливості багатьох однокристальних процесорів.

Аналіз коефіцієнтів в [70] показав, що, починаючи з 50 ... 100 члена ряду коефіцієнти змінюються по залежності, на частки відсотка відрізняються від геометричній прогресії, що при побудові графіків в логарифмічних осях відповідає прямим лініям (рис. 1.6).

Це дозволяє істотно спростити процедуру розрахунку дробового інтеграла і звести її до декількох сотень поєднань операцій множення і додавання протягом одного періоду квантування. Найважливіше полягає в тому, що в пам'яті процесора необхідно зберігати лише останні значення вхідного сигналу і таку ж кількість коефіцієнтів [71].

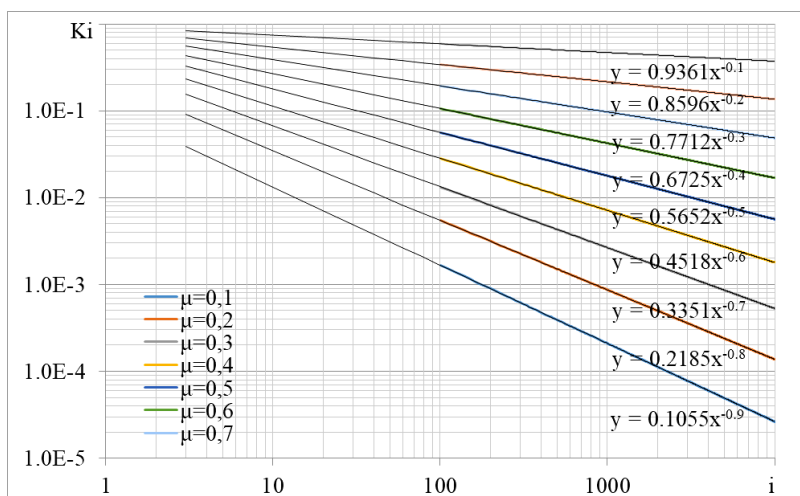


Рис. 1.6. Залежності коефіцієнтів у формул дробового інтегрування від порядкового номера [70]

Для оптимізації обчислень доцільно останні значення вхідного сигналу зберігати в організованому кільцевому масиві з нумерацією від 0 до $n_{\text{dim}} - 1$. Якщо задати $n_{\text{dim}} = 2^k$ і вибрати k від 6 до 8, резервуючи масиви з 64, 128 або 256 елементів, то процедура зміни номера j елемента кільцевого масиву описується простим виразом $j = (j + 1) \& \text{mask}$, де $\text{mask} = (n_{\text{dim}} - 1) = 0b0\dots0011\dots11$ є двійковим числом, в якому k молодших розрядів рівні 1. На рис. 1.7 наведено алгоритм розрахунку дробового інтеграла, в якому можна виділити дві основні частини.

1-й блок виконується одноразово при запуску системи і складається з n_{dim} циклів. У ньому розраховується точне значення дробового інтеграла і заповнюється масив вхідних значень першими точками. Далі в 2-му блоці організовується «ковзний» по кільцевому масиву покажчик C_i і цикл розрахунку складової дробового інтеграла S для останнього виміряного часу вхідного сигналу X з лічильником C_j . А розраховане в попередньому періоді квантування значення «старої» складової дробового інтеграла S_s множиться на коефіцієнт геометричний

прогресії q і доповнюється новою складовою $X[Ci] * k[ndim]$. Результуюче значення – сума S і Ss .

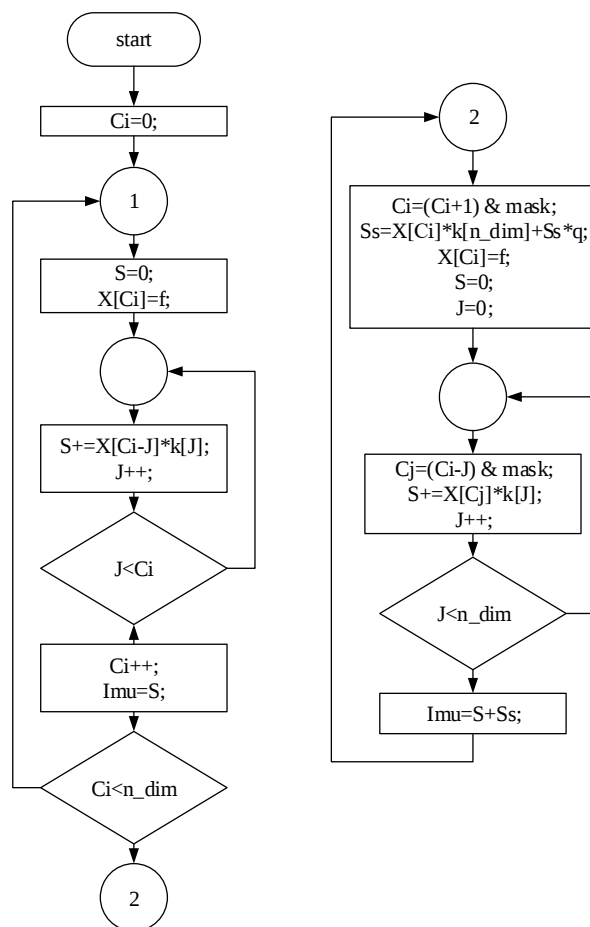


Рис. 1.7. Алгоритм розрахунку дробового інтеграла [70]

Даний алгоритм наближеного розрахунку дробових інтегралів, заснований на апроксимації старших коефіцієнтів, розкладанням в ряд геометричної прогресії, що забезпечує скорочення необхідних обсягів пам'яті для зберігання масивів коефіцієнтів і історії вхідного сигналу в сотні-тисячі разів і в стільки ж разів скорочує витрати часу процесора для обчислення сигналу регулятора. Це дозволяє на базі сучасних мікроконтролерів, таких, як Intel® Quark™ SoC X1000, STM32F4,

FPGA Altera Cyclone V, реалізувати дробові інтегрально-диференційні регулятори з високою швидкістю.

Отже, в даному розділі проведено короткий огляд історії розвитку дробового числення та наведено приклади застосування цього математичного апарату. Аналіз літературних джерел показує широке застосування дробових $PI^{\gamma}D^{\mu}$ - та подібних типів регуляторів при оптимізації перехідних процесів, а наявність широкого спектру методів для апроксимації розрахунку дробового інтегралу дає можливість простій реалізації на мікроконтролерах та практичного застосування в сучасних системах керування електричними машинами.

РОЗДІЛ 2

РЕГУЛЯТОРИ З ДРОБОВИМ ПОРЯДКОМ ІНТЕГРУВАННЯ ТА ДИФЕРЕНЦЮВАННЯ В СИСТЕМІ КЕРУВАННЯ СТРУМОМ ДВИГУНА ПОСТІЙНОГО СТРУМУ З ПОСЛІДОВНИМ ЗБУДЖЕННЯМ

2.1 Аналіз моделі двигуна постійного струму з послідовним збудженням

Характеристики двигуна послідовного збудження нелінійні через те, що магнітний потік полюсів цих машин непостійний і залежить не тільки від струму послідовної обмотки, що є загальним з якорем, а й від магнітної системи, яка, насичуючись, порушує пропорційність між потоком та струмом [72].

Математичний опис динамічного процесу електромеханічного перетворення енергії в двигуні постійного струму з послідовним збудженням має такий вигляд [73]:

$$\begin{cases} 0 = i_{ec}(t) \cdot R_{ec} + w_{ec} \cdot \frac{d\Phi(t)}{dt}, \\ u_a(t) = R_{a\Sigma} \cdot i_a(t) + L_{a\Sigma} \cdot \frac{di_a(t)}{dt} + k \cdot \Phi(t) \cdot \omega(t) + w_f \cdot \frac{d\Phi(t)}{dt}, \\ M(t) = k \cdot \Phi(t) \cdot i_a(t), \end{cases} \quad (2.1)$$

де $R_{a\Sigma} = R_a + R_{ap} + R_f$ – сумарний опір якірного ланцюга; w_f – число витків обмотки збудження; $L_{a\Sigma}$ – індуктивність розсіювання якірного ланцюга двигуна; $i_{ec}(t)$, R_{ec} , w_{ec} – умовний струм, опір, число витків для врахування впливу вихрових струмів на динамічні процеси; $\Phi(t)$ – магнітний потік двигуна; k – конструктивний коефіцієнт.

Індуктивність розсіювання $L_{a\Sigma}$ набагато менше індуктивності обмотки збудження $L_{f\Sigma}$. Але нехтувати їй при дослідженні динамічних процесів не слід, так як при $L_{a\Sigma} = 0$ стрибкоподібна зміна напруги двигуна згідно (2.1) повинна викликати миттєву зміну струму двигуна, що в принципі неможливо.

На рис. 2.1 зображена структурна схема двигуна по рівнянням (2.1) без урахування втрат на вихрові струми.

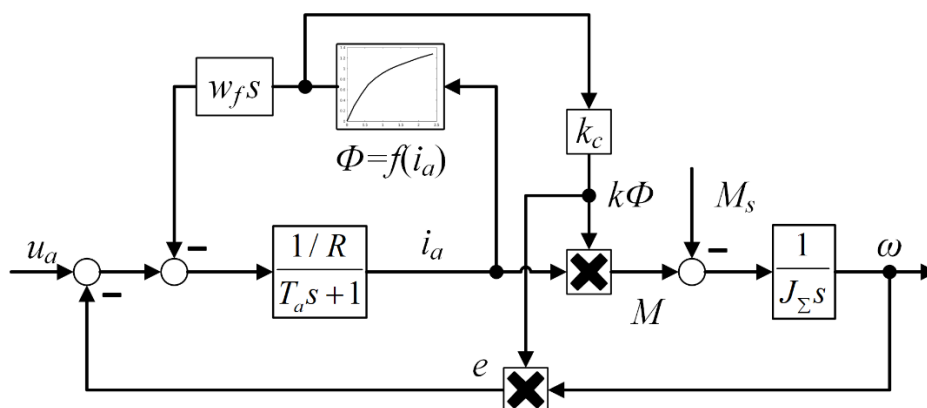


Рис. 2.1. Структурна схема математичної моделі двигуна послідовного збудження

Як об'єкт дослідження було обрано двигун серії Д-12 з номінальними параметрами $P_n=2.5$ kW, $\omega_n=115$ rad/s, $U_n=220$ V, $I_n=16$ A (більш детально в табл. 2.1).

Таблиця 2.1. Параметри двигуна серії Д-12

Параметр	Значення
Номінальна потужність P_n , kW	2.5
Номінальна частота обертання n_n , rpm	1100
Номінальна напруга U_n , V	220
Номінальний струм I_n , A	16
Опір обмотки якоря R_a , Ω	1.43
Опір послідовної обмотки R_f , Ω	0.59

Магнітний потік полюса Φ_n , mWb	5.2
Число витків полюса послідовної обмотки w_f	83
Число активних провідників якоря N_a	990
Число паралельних гілок якоря $2a$	2
Число пар полюсів $2p$	4
Момент інерції двигуна J_m , kg·m ²	0.05

Крива намагнічування зображена на рис. 2.2. За вихідними даними з табл. 2.1 і відомими формулами знайдемо необхідні коефіцієнти для моделювання.

Конструктивний коефіцієнт двигуна:

$$k_c = \frac{pN_a}{2\pi a} = 315.1 \quad (2.2)$$

Номінальний момент двигуна:

$$M_n = k_c \Phi_n I_n = 26.22 \text{ N} \cdot \text{m} \quad (2.3)$$

Номінальна кутова частота:

$$\omega_n = \frac{2\pi n_n}{60} = 115.2 \text{ rad} / \text{s} \quad (2.4)$$

Індуктивність якоря двигуна за формулою Уманського-Ленвілля:

$$L_a = \frac{kU_n}{p\omega_n I_n} = 14.99 \text{ mH} \quad (2.5)$$

де $k = 0.25$ – поправочний коефіцієнт для компенсаційних двигунів.

Загальний опір ланцюга якоря з урахуванням нагріву:

$$R_{\Sigma} = (R_a + R_f)(1 + \alpha_{Cu}(T_{\max} - T_e)) = 3.313 \, \Omega \quad (2.6)$$

де $\alpha_{Cu} = 0.004 \frac{1}{^{\circ}\text{C}}$ – температурний коефіцієнт опору міді, $T_{\max}$ – максимальна

температура якоря, $^{\circ}\text{C}$; T_e – температура навколишнього середовища, $^{\circ}\text{C}$.

Постійна часу обмотки якоря:

$$T_a = \frac{L_a}{R_{\Sigma}} = 4.497 \, \text{ms} \quad (2.7)$$

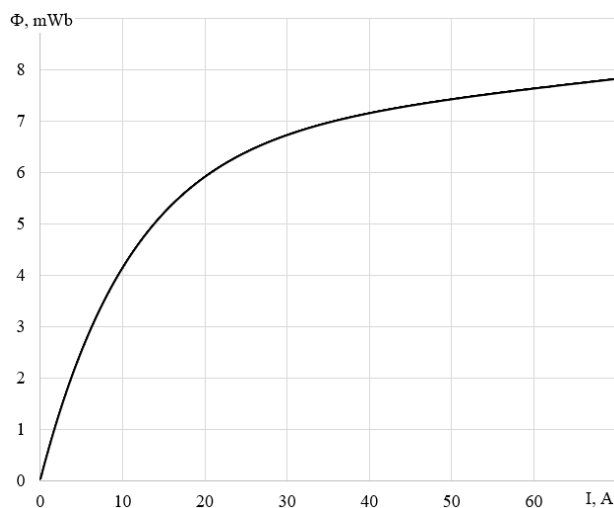


Рис. 2.2. Крива намагнічування $\Phi = f(i_a)$ двигуна послідовного збудження серії Д-12

Виділимо структурну схему ланцюга якоря з рис. 2.1 (рис. 2.3) і проведемо моделювання з різними значеннями напруги живлення (рис. 2.4) – 30 V, 60 V, 90 V і 120 V.

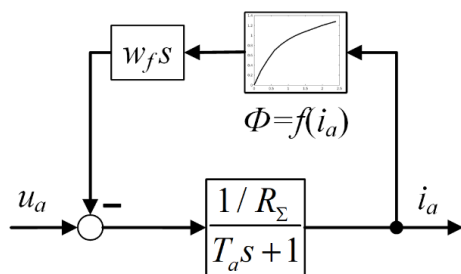


Рис. 2.3. Структурна схема якоря двигуна з послідовним збудженням

Як видно з рис. 2.4 при підвищенні напруги перехідний процес за струмом видозмінюється і стає менше схожим на експоненту, при цьому еквівалентна постійна часу стає менше. Порівняння перехідних процесів експоненти з постійною часу $T_\Sigma = 36\text{ ms}$ та ланцюга якоря при стрибкоподібній напрузі в 53 V показано на рисунку 2.5.

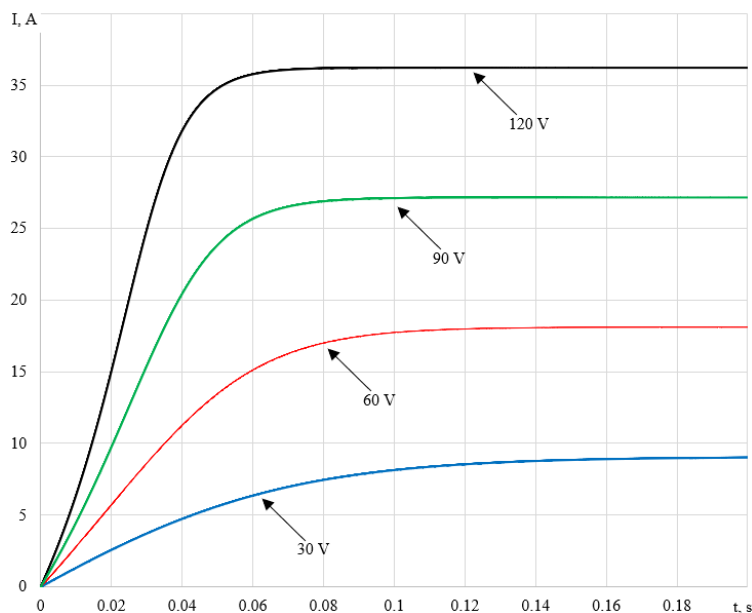


Рис. 2.4. Перехідні процеси за струмом з чотирма різними значеннями напруги живлення

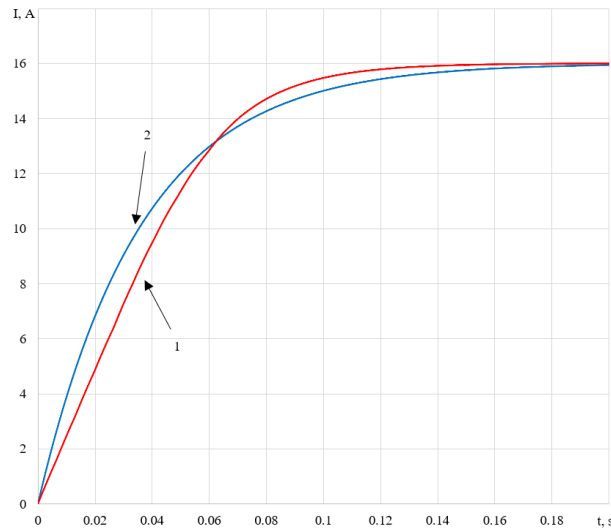


Рис. 2.5. Перехідні процеси за струмом з урахуванням кривої намагнічування (1) і апроксимуючої аперіодичною функцією (2)

2.2 Моделювання замкнутої системи керування за струмом

Після визначення основних параметрів перейдемо до моделювання системи керування для даного двигуна. Вона буде складатися з двох контурів – струму. Виділимо контур керування струмом на рис. 2.6. На ній зображені $W_{cc}(s)$ – передавальна функція регулятора, $\frac{k_{conv}}{T_{\mu c}s + 1}$ – передавальна функція перетворювача,

$\frac{1/R_{\Sigma}}{T_{\Sigma}s + 1}$ – передавальна функція ланцюга якоря, k_{conv} – коефіцієнт посилення перетворювача, $T_{\mu c}$ – мала некомпенсована постійна часу, k_{cs} – коефіцієнт посилення датчика струму, u_{refC} – сигнал завдання струму, u_{cur} – сигнал зворотного зв'язку за струмом, Δu_{cur} – сигнал помилки між завданням і зворотного зв'язку за струмом, u_a – напруга якоря, e – ЕРС двигуна, яке в даному випадку дорівнює нулю.

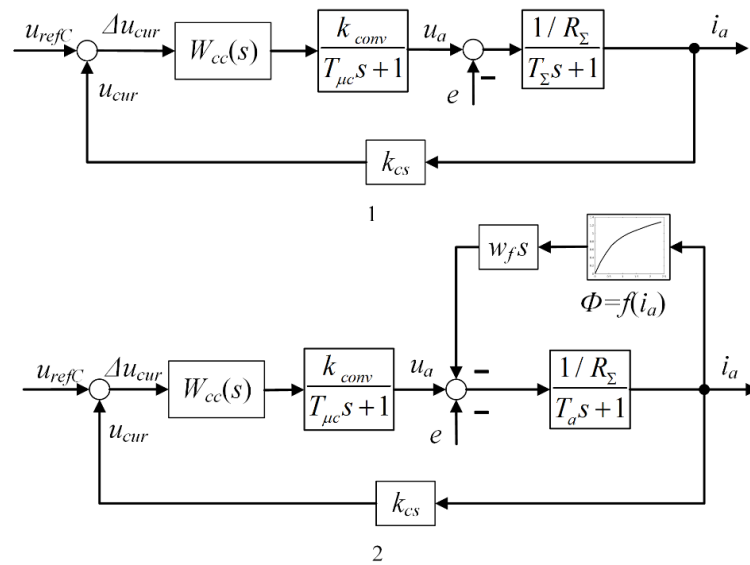


Рис. 2.6. Структурна схема САК струму двигуна з послідовним збудженням, де 1 – без урахування кривої намагнічування, 2 – з урахуванням кривої намагнічування

Коефіцієнт посилення перетворювача:

$$k_{conv} = \frac{U_n}{10} = 22 \quad (2.8)$$

Коефіцієнт посилення зворотного зв'язку за струмом:

$$k_{cs} = \frac{10}{2I_n} = 0.313 \quad (2.9)$$

2.2.1 Синтез РІ-регулятора струму

Передавальну функцію регулятора $W_{cc}(s)$ знайдемо виходячи з налаштувань на модульний оптимум без урахування кривої намагнічування

$$W_{op.c}^{mo}(s) = W_{cc}^{mo}(s) \cdot \frac{k_{conv}}{T_{\mu c}s + 1} \cdot \frac{1/R_{\Sigma}}{T_{\Sigma}s + 1} \cdot k_{cs} = \frac{1}{2T_{\mu c}s \cdot (T_{\mu c}s + 1)} \quad (2.10)$$

звідки

$$W_{cc}^{mo}(s) = \frac{1}{2T_{\mu c}s \cdot (T_{\mu c}s + 1)} : \frac{k_{conv}k_{cs}}{(T_{\mu c}s + 1)(T_{\Sigma}s + 1)R_{\Sigma}} \quad (2.11)$$

Після перетворень отримаємо

$$W_{cc}^{mo}(s) = K_1 + K_2s^{-1} \quad (2.12)$$

де коефіцієнти

$$K_1 = \frac{T_{\Sigma}R_{\Sigma}}{2T_{\mu c}k_{conv}k_{cs}} = \frac{0.36 \cdot 3.641}{2 \cdot 0.01 \cdot 22 \cdot 0.313} = 0.867,$$

$$K_2 = \frac{R_{\Sigma}}{2T_{\mu c}k_{conv}k_{cs}} = \frac{3.641}{2 \cdot 0.01 \cdot 22 \cdot 0.313} = 24.093$$

Після розрахунку параметрів проведемо по структурній схемі (рис. 2.7) моделювання з урахуванням кривої намагнічування та без.

Як бачимо з рис. 2.7 сильні розбіжності між об'єктом керування та її апроксимацією викликають погіршення якості перехідного процесу, так перерегулювання зросло з 4.3% до $\frac{18.01-16}{16} \cdot 100\% = 12.56\%$, але час першого узгодження зменшився з 44.5 ms до 36.8 ms.

Зменшимо перерегулювання шляхом зміни коефіцієнтів регулятора і приймемо їх рівними $K_1 = 0.667$, $K_2 = 18.53$. Результат моделювання на рис. 2.8.

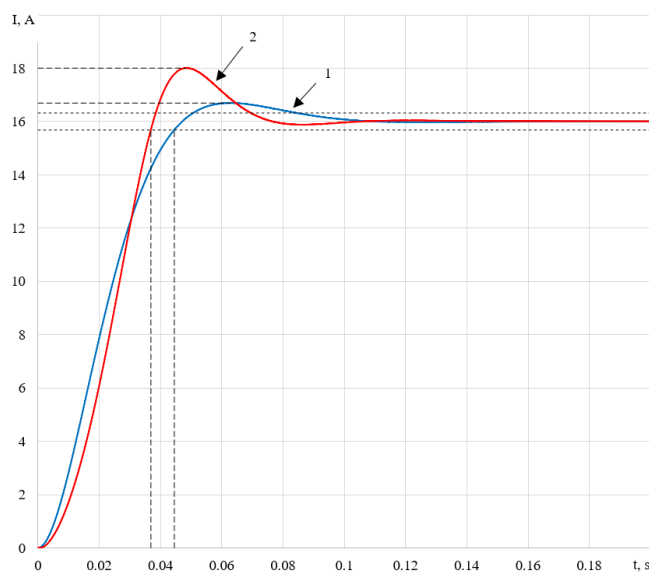


Рис. 2.7. Перехідні процеси за струмом з РІ-регулятором без урахування кривої намагнічування (1) та з її урахуванням (2)

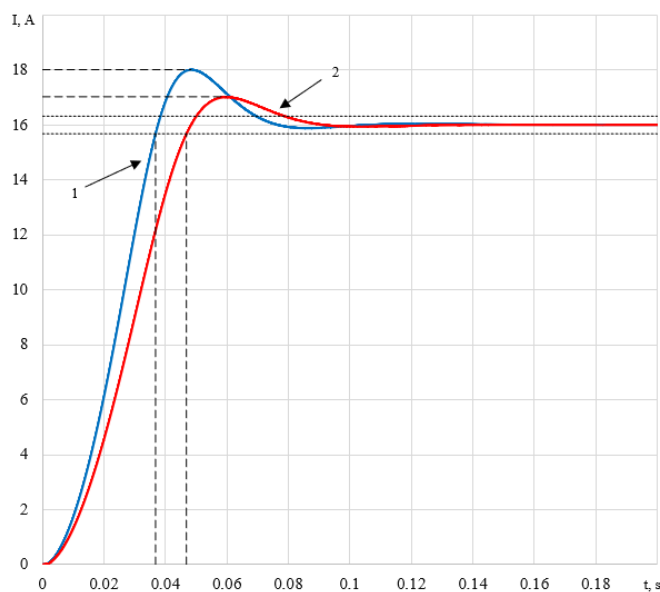


Рис. 2.8. Перехідні процеси за струмом з урахуванням кривої намагнічування зі старими (1) та новими (2) коефіцієнтами

Це дозволило зменшити перерегулювання з 12.56 % до $\frac{17.03-16}{16} \cdot 100\% = 6.4\%$, але при цьому збільшився час першого узгодження з 36.8 ms до 46.8 ms.

Перевіримо роботу регулятора на інших завданнях струму (рис. 2.9).

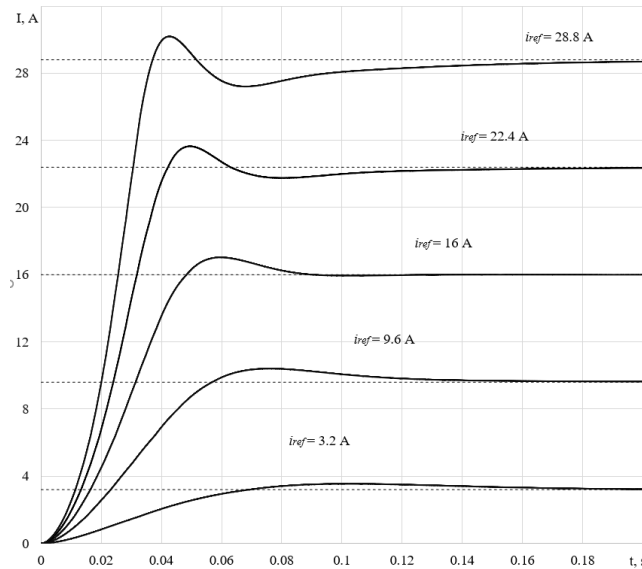


Рис. 2.9. Перехідні процеси за струмом з PI-регулятором з урахуванням кривої намагнічування із ступінчастим завданням струму 3.2 А, 9.6 А, 16 А, 22.4 А і 28.8 А

Як видно з рисунка 2.9, якщо порівнювати при завданні 16 А, то чим нижче завдання, тим перехідний процес стає затягнутим. А якщо завдання вище – перехідний процес характеризується підвищеним перерегулюванням, коливальністю та «дотягуванням» до заданого значення.

2.2.2 Синтез D^μPI^μ-регулятора струму

Далі знайдемо передавальну функцію регулятора $W_{cc}(s)$ виходячи з налаштувань на дробовий порядок астатизму без урахування кривої намагнічування:

$$W_{op.c}^{fo}(s) = W_{cc}^{fo}(s) \frac{k_{conv}}{T_{\mu c}s + 1} \frac{1/R_{\Sigma}}{T_{\Sigma}s + 1} k_{cs} = \frac{1}{aT_{\mu c}^{\mu-1}s^{\mu-1}} \frac{bT_{\mu c}s + 1}{bT_{\mu c}s} \frac{1}{(T_{\mu c}s + 1)}, \quad (2.13)$$

де значення a і b наближено вибираються за наступними співвідношеннями:

$$\begin{cases} (ba) \approx \exp(-10.27 + 7.831\mu), \\ b \approx 7.336 + 0.792(ba) + 3.83\ln(ba), \end{cases}$$

звідки

$$\begin{aligned} W_{cc}^{fo}(s) &= \frac{1}{aT_{\mu c}^{\mu-1}s^{\mu-1}} \frac{bT_{\mu c}s+1}{bT_{\mu c}s} \frac{1}{(T_{\mu c}s+1)} \frac{(T_{\mu c}s+1) \cdot (T_{\Sigma}s+1)R_{\Sigma}}{k_{conv}k_{cs}} = \\ &= \frac{R_{\Sigma}}{abT_{\mu c}^{\mu}k_{conv}k_{cs}} \left(bT_{\mu c}T_{\Sigma}s^{2-\mu} + (T_{\Sigma} + bT_{\mu c})s^{1-\mu} + s^{\mu} \right) \end{aligned} \quad (2.14)$$

Тобто регулятор, налаштований на дробовий порядок з астатизмом 1.5 з об'єктом керування – аперіодичною ланкою, має три складові – дробова похідна порядку 0.5, дробовий інтеграл порядку 1.5 та дробовий інтеграл порядку 0.5:

$$W_{cc}^{fo}(s) = K_1s^{0.5} + K_2s^{-0.5} + K_3s^{-1.5} \quad (2.15)$$

де

$$K_0 = K_3 = \frac{R_{\Sigma}}{abT_{\mu c}^{\mu}k_{conv}k_{cs}} = 110.1,$$

$$K_1 = K_0bT_{\mu c}T_{\Sigma} = 0.667,$$

$$K_2 = K_0(T_{\Sigma} + bT_{\mu c}) = 22.5.$$

Проведемо моделювання за структурною схемою зображеною на рис. 2.6 використовуючи отриманий регулятор. Результат зображений на рис. 2.10.

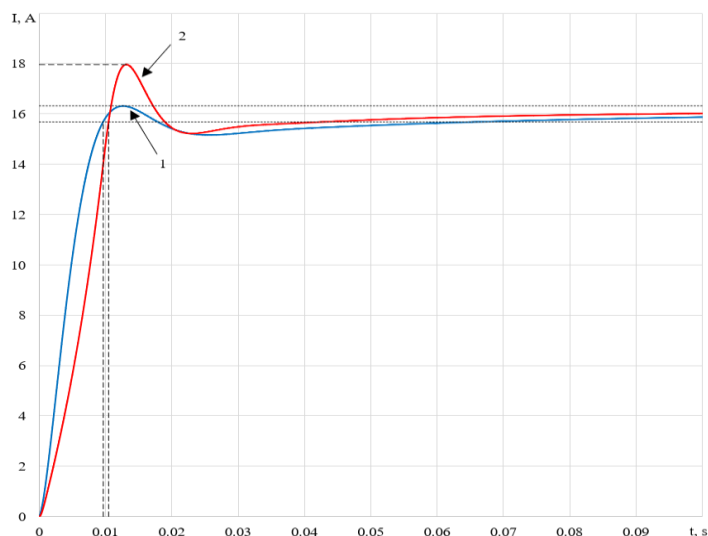


Рис. 2.10. Перехідні процеси за струмом з D^{IT}І^μ-регулятором без урахування кривої намагнічування (1) та з її урахуванням (2)

Отримані перехідні процеси показують аналогічний результат, як і в випадку з налаштуванням на модульний оптимум – при врахуванні в моделюванні кривої намагнічування якість перехідного процесу відрізняється від бажаного. Так перерегулювання зросло с 0 % до $\frac{17.96-16}{16} \cdot 100\% = 12.25\%$ і збільшився час першого узгодження з 9.7 ms до 10.5 ms.

Зменшимо перерегулювання шляхом зміни коефіцієнтів регулятора та приймемо їх рівними $K_1 = 0.2253$, $K_2 = 11.44$, $K_3 = 55.04$. Результат моделювання на рис. 2.11.

Це дозволило зменшити перерегулювання з 12.25% до 0%, але при цьому збільшився час першого узгодження з 10.5 ms до 22 ms. Перевіримо роботу D^{IT}І^μ-регулятора на інших завданнях струму і порівняємо з PI-регулятором (рис. 2.12).

Як видно з рис. 2.12 D^{IT}І^μ-регулятор дає істотне зниження часу першого узгодження, а також зменшує перерегулювання практично до нуля. Але є невеликий недолік у вигляді «дотягування» до заданого значення. Виведемо параметри якості перехідного процесу в таблицю 2.2.

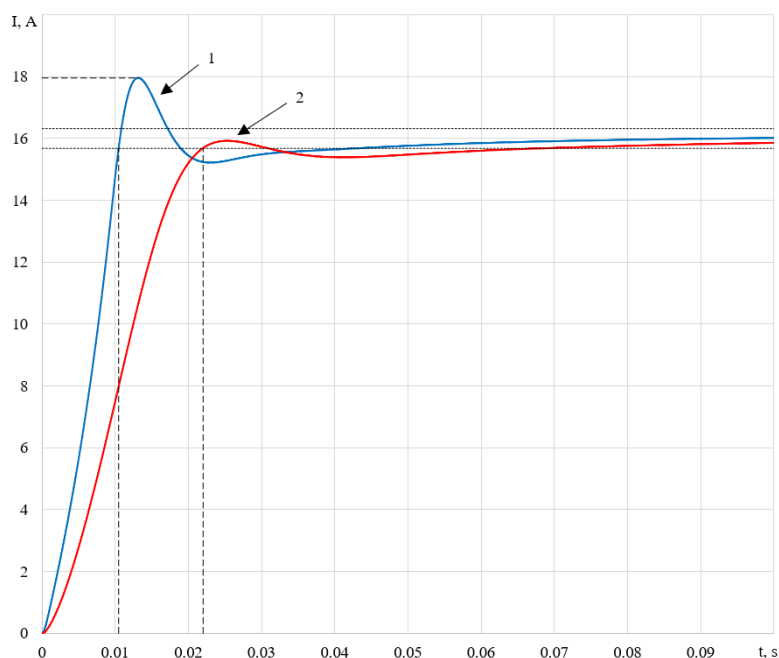


Рис. 2.11. Перехідні процеси за струмом з урахуванням кривої намагнічування зі старими (1) і з новими (2) коефіцієнтами $D^{\gamma}T^{\mu}$ -регулятора

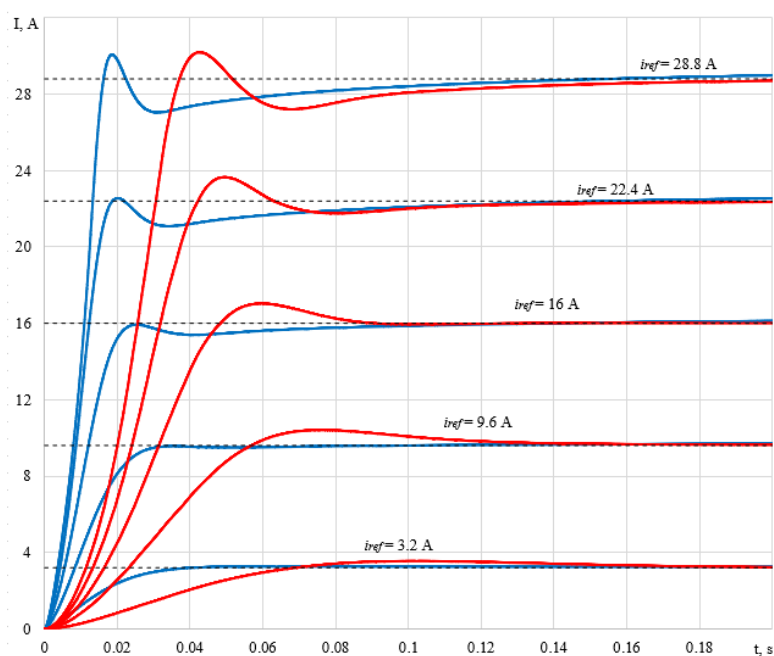


Рис. 2.12. Перехідні процеси за струмом з PI -регулятором (червоний) і з $D^{\gamma}T^{\mu}$ -регулятором (синій) з урахуванням кривої намагнічування із ступінчастим завданням струму 3.2 A, 9.6 A, 16 A, 22.4 A і 28.8 A

Таблиця 2.2. Час першого узгодження та перерегулювання перехідних процесів за струмом з PI- та D^УІ^У-регулятором

Завдання i_{ref} , А	PI-регулятор		D ^У І ^У -регулятор	
	Час першого узгодження t_f , ms	Перерегулювання σ , %	Час першого узгодження t_f , ms	Перерегулювання σ , %
3.2	66.8	10.71	36.5	1.84
9.6	55	8.36	28.6	0
16	46.8	6.42	22	0
22.4	41	5.55	17.4	0.6
28.8	36.2	4.8	15.8	4.4

З таблиці 2.2 видно, що в обох випадках обидва регулятора з підвищенням завдання, час перехідного процесу зменшується, але перерегулювання з PI-регулятором зменшується, а з D^УІ^У-регулятором – збільшується. Також ця таблиця показує незрівнянно кращі показники з регулятором при налаштуванні на дробовий порядок астатизму 1.5 [74].

2.3 Експериментальні дослідження контуру струму двигуна послідовного збудження

Як об'єкт експериментального дослідження виступив двигун постійного струму з послідовним збудженням серії MCA 38/64-148/AD8, з номінальною частотою $\omega_n = 12800$ rpm, струмом $I_n = 1.6$ А, опором $R = 6$ Ω та потужністю $P_n = 370$ W (рис. 2.13). Для вимірювання струму використано аналоговий датчик ACS712, заснований на ефекті Холу. 12-ти бітове аналогово-цифрове перетворення і запис даних виконувала програма, яка працювала на платі STM32F4DISCOVERY з тактовою частотою 168 MHz. До двигуна прикладалася напруга, що подається від блоку живлення Moeller SN3-100-BV8 24 V, а її

регулювання проводилося за допомогою широтно-імпульсного чотириквadrантного перетворювача.

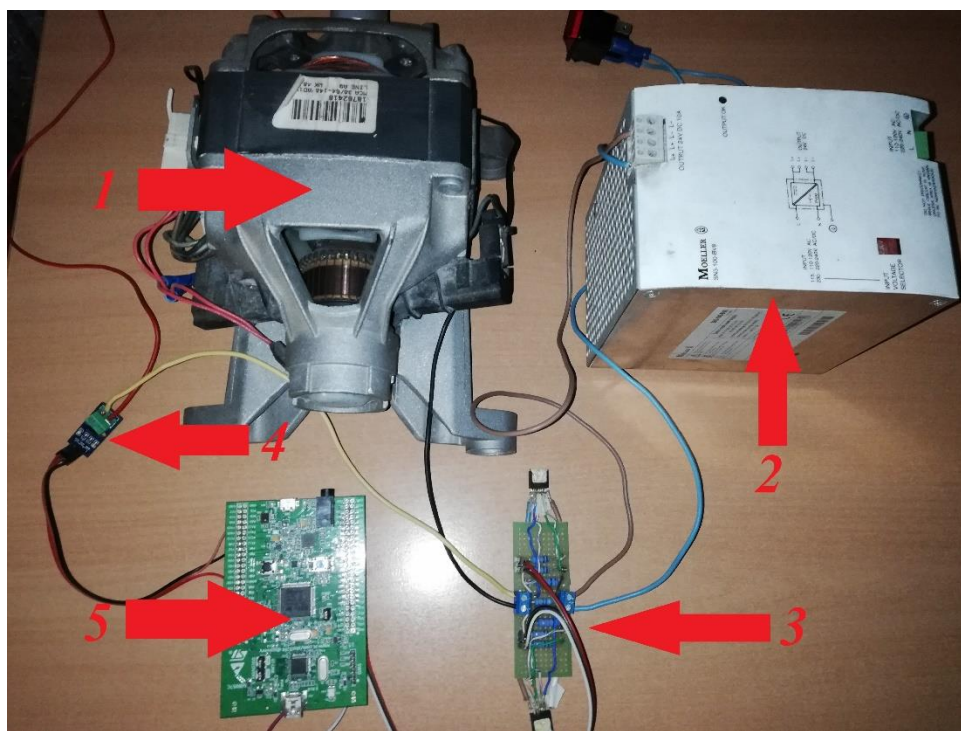


Рис. 2.13. Універсальний двигун (1), датчик струму ACS712 (4), плата STM32F4DISCOVERY (5), блок живлення SN3-100-BV8 (2), макетна плата з транзисторним мостом (3)

Мостова схема паялася на макетній платі (рис. 2.14). Вона в основному складається з двох полумостових інтегральних мікросхем BTN7960, яка може витримати струм до 40 А.

Якщо струм двигуна порівняно малий, то крива намагнічування в цій області майже лінійна, отже, контур струму можна описати інерційною ланкою першого порядку. Визначимо його постійну часу за перехідним процесом при подачі напруги 10 В, при якому не відбувається насичення в обмотці збудження (рис. 2.15). В результаті обробки даних визначена постійна часу – $T_a=0.0267$ с.

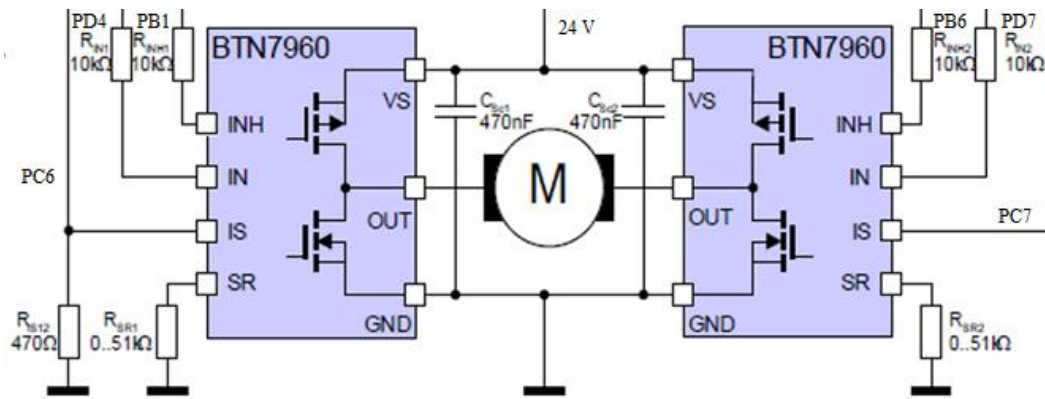


Рис. 2.14. Схема з'єднання мікросхеми BTN7960

Побудуємо замкнуту систему керування за цим об'єктом. За малу постійну часу приймемо $T_{\mu c} = 0.01 \text{ s}$. Коефіцієнт перетворювача дорівнює:

$$k_{conv} = \frac{U_n}{10} = 2.4 \quad (2.16)$$

Коефіцієнт зворотного зв'язку:

$$k_{cs} = \frac{I_{\max}}{10} = 0.4 \quad (2.17)$$

Коефіцієнти PI-регулятора:

$$K_1 = \frac{T_a R_a}{2T_{\mu c} k_{conv} k_{cs}} = 8.34, \quad (2.18)$$

$$K_2 = \frac{R_a}{2T_{\mu c} k_{conv} k_{cs}} = 312.5.$$

При завданні струму нижче номінального (1.6 А) перехідні процеси відповідають теоретичним. Однак при великих завданнях спостерігаються помітні розбіжності (рис. 2.16).

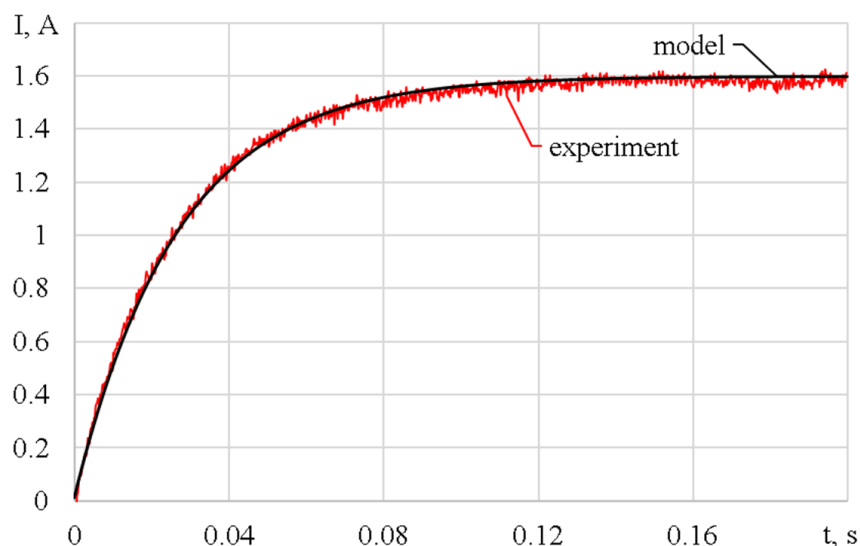


Рис. 2.15. Графік перехідних процесів інерційної ланки першого порядку з постійною часу $T_a = 0.0267$ s та експерименту при напрузі 10 V

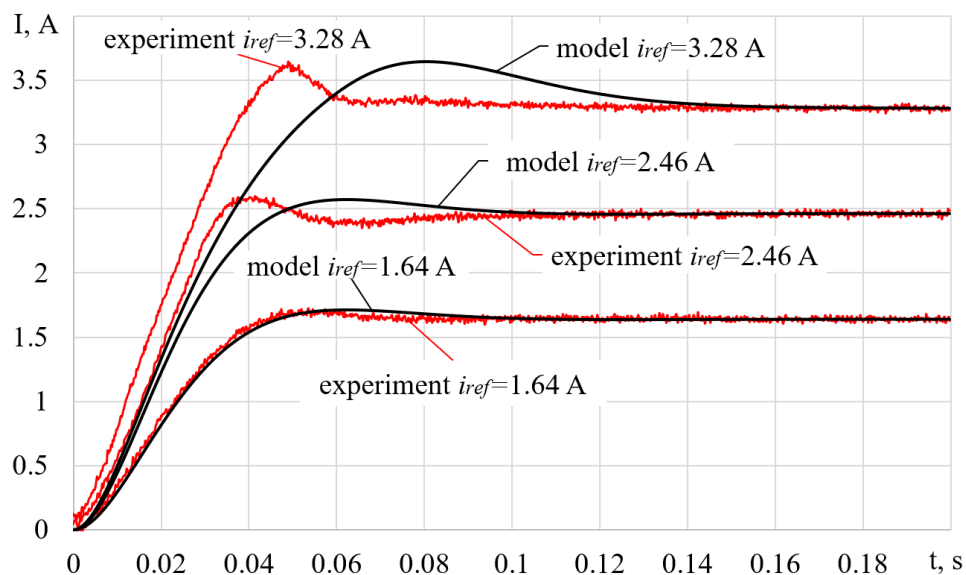


Рис. 2.16. Графіки перехідних процесів моделі та експерименту з ПІ-регулятором

Для синтезу оптимізованої системи виконаємо ідентифікацію об'єкта керування при подачі стрибком максимально допустимої напруги (рис. 2.17).

При математичному описі перехідного процесу були розглянуті наступні передавальні функції об'єкта керування:

$$W_{co1} = \frac{K}{a_0 s^\mu + 1}. \quad (2.19)$$

$$W_{co2} = \frac{K}{a_1 s + a_0 s^\mu + 1}. \quad (2.20)$$

$$W_{co3} = \frac{K}{a_1 s^{1+\mu} + a_0 s^\mu + 1}. \quad (2.21)$$

Для ідентифікації невідомих коефіцієнтів в (2.19), (2.20) и (2.21) використано генетичний алгоритм, в якому при селекції обрано турнірний метод, нові особини успадковували гени батьків шляхом рівномірного схрещування, ймовірність мутації в хромосомі – 20%. Оцінкою пристосованості особини служила середньоквадратична помилка F . Результати ідентифікації передавальних функцій зведені в табл. 2.3.

Таблиця 2.3. Результати ідентифікації передавальних функцій

Параметр	Передавальна функція об'єкта управління		
	W_{co1}	W_{co2}	W_{co3}
μ	1.14	0.6374	0.3533
a_0	0.01414	0.04951	0.1271
a_1	0	0.02439	0.006193
K	0.1623	0.1705	0.1928
F	0.0079	0.0163	0.0039

Експериментальні та розрахункові графіки зіставлені на рис. 2.17.

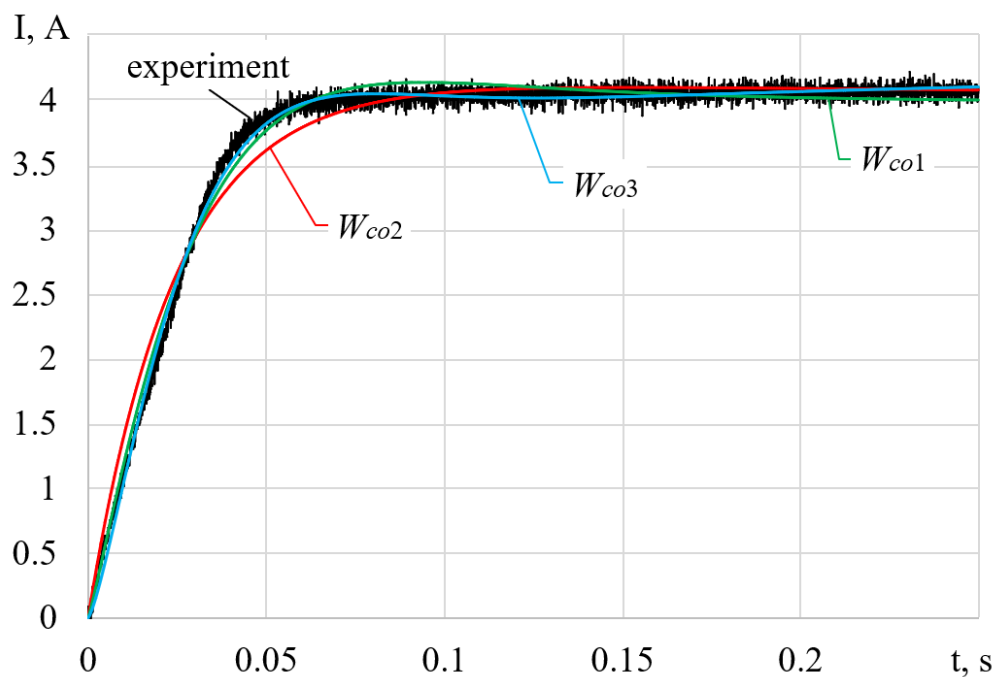


Рис. 2.17. Графіки перехідних процесів за передавальними функціями W_{co1} , W_{co2} , W_{co3} та за експериментальними даними

Очевидно, що найкращою передавальною функцією з найменшою похибкою є W_{co3} , яка буде використана в подальшому для синтезу регуляторів [75].

2.3.1 Синтез замкнутого контуру струму з оптимальними перехідними характеристиками

Проаналізуємо замкнуту систему керування струмом ДПС ПЗ при різних налаштуваннях.

Структурна схема замкнутої системи регулювання струму зображена на рис. 2.18. На ньому u_{ref} , u_{cs} , Δu_r , u_a , i_a – сигнали завдання струму та зворотного зв'язку, помилка між ними, напруга і струм якоря відповідно. Також $W_{cr}(s)$,

$W_{conv}(s) = \frac{k_{conv}}{T_{\mu} s + 1}$, $W_{cs}(s) = k_{cs}$, $W_{co}(s)$ – передавальні функції регулятора струму,

перетворювача, датчика струму та об'єкта керування відповідно.

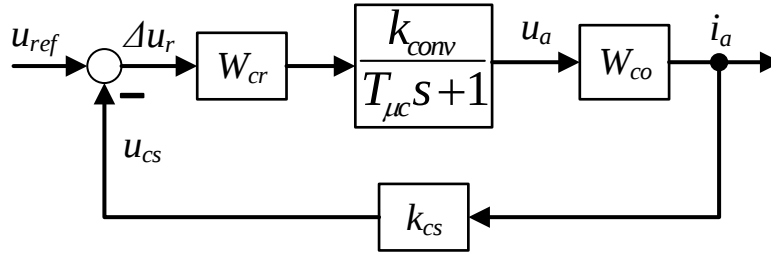


Рис. 2.18. Структурна схема системи керування струмом

Визначимо передавальну функцію регулятора струму $W_{cr}(s)$ виходячи з налаштувань на модульний оптимум з урахуванням кривої намагнічування:

$$W_{op.c}(s) = \frac{1}{2T_{\mu c}s(T_{\mu c}s + 1)} = W_{cr}(s) \frac{k_{conv}}{(T_{\mu c}s + 1)} \frac{K}{(a_1 s^{1+\mu} + a_0 s^{\mu} + 1)} k_{cs}, \quad (2.22)$$

звідки

$$W_{cr}(s) = \frac{1}{2T_{\mu c}s(T_{\mu c}s + 1)} \cdot \frac{(T_{\mu c}s + 1)(a_1 s^{1+\mu} + a_0 s^{\mu} + 1)}{k_{conv} k_{cs} K}. \quad (2.23)$$

Після перетворень отримаємо

$$W_{cr}(s) = K_1 s s^{\mu_{co}-1} + K_2 s^{\mu_{co}-1} + K_3 s^{-1}, \quad (2.24)$$

де коефіцієнти

$$K_1 = \frac{a_1}{2T_{\mu c} k_{conv} k_{cs} K}, K_2 = \frac{a_0}{2T_{\mu c} k_{conv} k_{cs} K}, K_3 = \frac{1}{2T_{\mu c} k_{conv} k_{cs} K}.$$

Для досліджуваного об'єкта отримано регулятор D^μI^νI (K₁=0.27, K₂=5.539, K₃=43.58), графік перехідного процесу струму показаний на рис. 2.19 і точно відповідає показникам модульного оптимуму.

Виконаємо налаштування системи з дробовим астатизмом порядку $\mu = \mu_{co} = 0.3533$ та $\mu = 0.6$. Це дозволить підвищити швидкодію системи керування та при цьому зменшити перерегулювання.

$$W_{op.c}(s) = \frac{1}{aT_{\mu c}^{\mu} s^{\mu} (T_{\mu c} s + 1)} = W_{cr}(s) \frac{k_{conv}}{(T_{\mu c} s + 1)} \frac{K}{(a_0 s^{1+\mu_{co}} + a_1 s^{\mu_{co}} + 1)} k_{cs}, \quad (2.25)$$

звідки

$$W_{cr}(s) = \frac{1}{aT_{\mu c}^{\mu} s^{\mu} (T_{\mu c} s + 1)} \cdot \frac{(T_{\mu c} s + 1)(a_1 s^{1+\mu_{oy}} + a_0 s^{\mu_{oy}} + 1)}{k_{conv} k_{cs} K}. \quad (2.26)$$

Після перетворень отримаємо

$$W_{cr}(s) = K_1 s s^{\mu_{co}-\mu} + K_2 s^{\mu_{co}-\mu} + K_3 s^{-\mu}. \quad (2.27)$$

Отже, коефіцієнти дорівнюють:

$$K_1 = \frac{a_1}{aT_{\mu c}^{\mu} k_{conv} k_{cs} K}, \quad K_2 = \frac{a_0}{aT_{\mu c}^{\mu} k_{conv} k_{cs} K}, \quad K_3 = \frac{1}{aT_{\mu c}^{\mu} k_{conv} k_{cs} K}, \quad (2.28)$$

де параметр налаштування $a \approx \frac{\mu}{4.683 - 5.897\mu + 1.595\mu^2}$ забезпечує найкраще співвідношення між швидкодією та перерегулюванням.

Результати моделювання контуру струму з порядком астатизму $\mu=\mu_{co}$ (регулятор $PI^{\mu}D$, $K_1=0.218$, $K_2=4.465$, $K_3=35.14$) та $\mu=0.6$ (регулятор $D^{\mu}PI^{\nu}$, $K_1=0.245$, $K_2=5.03$, $K_3=39.58$) також зображені на рис. 2.19.

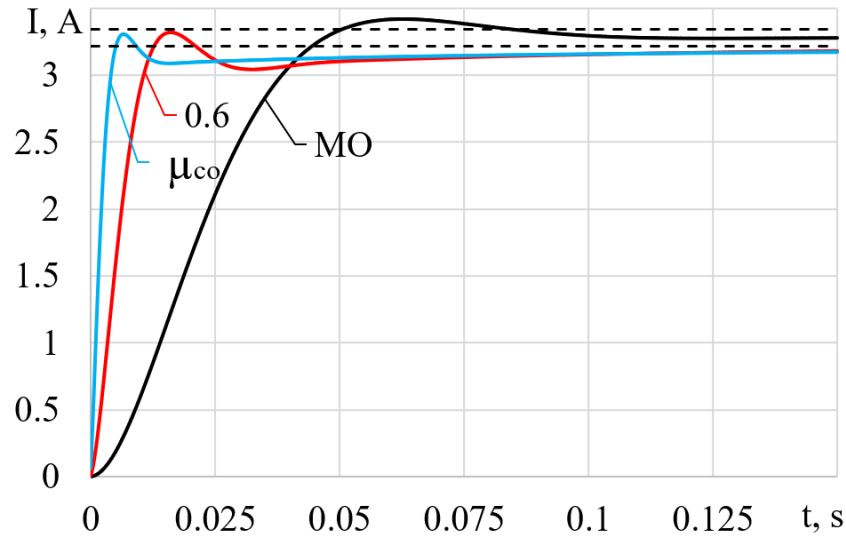


Рис. 2.19. Графіки перехідних процесів при налаштуванні на модульний оптимум та дробовий порядок астатизму $\mu=\mu_{co}$, $\mu=0.6$

Недоліком останніх налаштувань є тривале наближення струму до заданого значення.

Найбільший інтерес представляє налаштування на дробовий порядок астатизму $1 + \mu$, $\mu \in (0; 1)$, при якому підвищується не тільки статична, але й динамічна точність системи.

Для цього визначимо передавальну функцію регулятора:

$$W_{op.c}(s) = \frac{1}{aT_{\mu c}^{\mu-1}s^{\mu-1}} \frac{bT_{\mu c}s+1}{bT_{\mu c}s} \frac{1}{(T_{\mu c}s+1)} = W_{cr}(s) \frac{k_{conv}}{T_{\mu c}s+1} \frac{K}{a_0s^{1+\mu_{co}} + a_1s^{\mu_{co}} + 1} k_{cs}, \quad (2.29)$$

де значення a та b наближено вибираються за наступним співвідношенням:

$$\begin{cases} (ba) \approx \exp(-10.27 + 7.831\mu), \\ b \approx 7.336 + 0.792(ba) + 3.83\ln(ba), \end{cases}$$

звідки

$$W_{cr}(s) = \frac{1}{aT_{\mu c}^{\mu-1}s^{\mu-1}} \frac{bT_{\mu c}s+1}{bT_{\mu c}s} \frac{1}{(T_{\mu c}s+1)} \frac{(T_{\mu c}s+1) \cdot (a_1s^{1+\mu_{co}} + a_0s^{\mu_{co}} + 1)}{k_{conv}k_{cs}K}. \quad (2.30)$$

Після перетворень отримаємо передавальну функцію регулятора, що складається з шести ланок:

$$W_{cr}(s) = K_0 s^{-(\mu-1)} [K_1 s^2 s^{\mu_{co}-1} + K_2 s s^{\mu_{co}-1} + K_3 s^{\mu_{co}-1} + K_4 s^{-1} + K_5], \quad (2.31)$$

де

$$K_0 = \frac{1}{aT_{\mu c}^{\mu-1}}, K_1 = \frac{a_1}{k_{conv}k_{cs}K}, K_2 = \frac{a_0 b T_{\mu c} + a_1}{b T_{\mu c} k_{conv} k_{cs} K}, K_3 = \frac{a_0}{b T_{\mu c} k_{conv} k_{cs} K},$$

$$K_4 = \frac{1}{b T_{\mu c} k_{conv} k_{cs} K}, K_5 = \frac{1}{k_{conv} k_{cs} K}.$$

Для $\mu=1.5$ отримуємо $K_0=37.04$, $K_1=0.0054$, $K_2=0.144$, $K_3=0.683$, $K_4=5.376$, $K_5=0.872$, а при $\mu = 1 + \mu_{co}$ – $K_0=35.09$, $K_1=0.0054$, $K_2=0.167$, $K_3=1.158$, $K_4=9.11$, $K_5=0.872$. Результати моделювання наведені на рис. 2.20. Очевидно, що швидкодія збільшилася, а перерегулювання в порівнянні з налаштуванням на модульний оптимум зменшено більш ніж в 2 рази.

Перехідні процеси, показані на рис. 2.19, 2.20, отримані без урахування обмеження напруги джерела живлення. Однак наруга експериментального стенду дозволяє подати тільки 24 вольт. В результаті перехідні характеристики не

відповідають розрахунковим, якісні показники процесу істотно погіршуються (рис. 2.21).

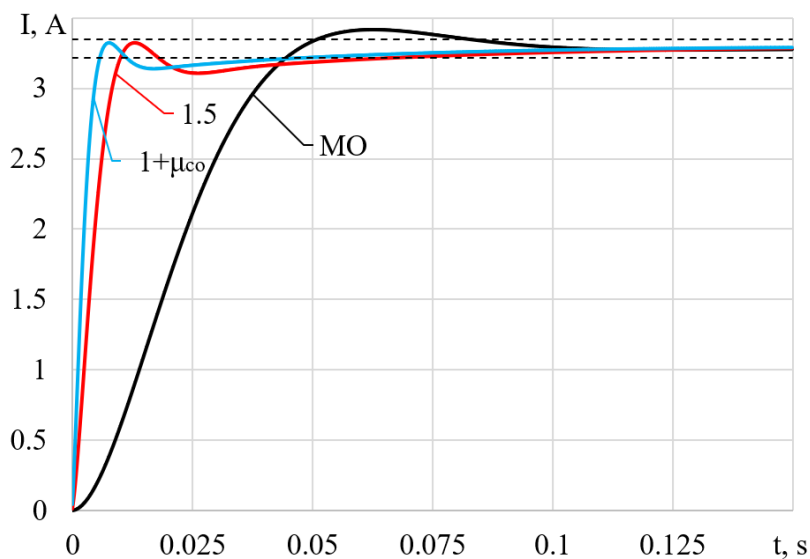


Рис. 2.20. Графіки перехідних процесів при налаштуванні на модульний оптимум та дробовий астатизм порядку $\mu=1+\mu_{co}$, $\mu=1.5$

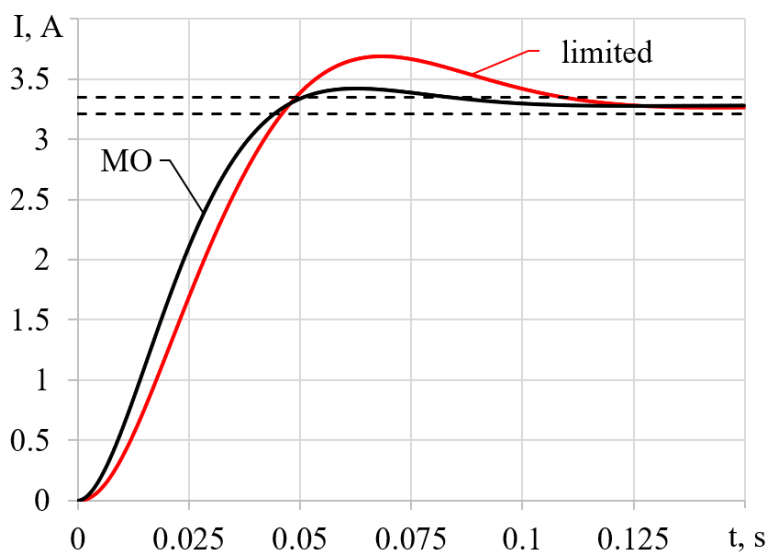


Рис. 2.21. Графіки перехідних процесів з обмеженням напруги живлення та без

Змінимо структуру регуляторів і визначимо їх параметри для забезпечення оптимальних за швидкодією та перерегулюванню перехідних процесів.

2.3.2 Пошук оптимальних параметрів регуляторів при обмеженій напрузі джерела живлення

Для пошуку оптимальних коефіцієнтів використаний один з варіантів генетичного алгоритму, в якому змінювалася опорна функція, по якій оцінювалася функція пристосованості кожної особини. Також обрані діапазони коефіцієнтів, в яких проводився пошук значень – для регулятора зі структурою D^μПІ $K_1 \in [1; 20]$, $K_2 \in [1; 80]$, $K_3 \in [1; 60]$. Потім шляхом схрещування і селекції особин знайдено найкраще рішення. Наприклад, для D^μПІ-регулятора (рис. 2.22) отримані коефіцієнти $K_1=1.746$, $K_2=40.75$, $K_3=12.43$ (замість $K_1=0.27$, $K_2=5.539$, $K_3=43.58$, визначених раніше) і відповідні графіки показані на рис. 2.23.

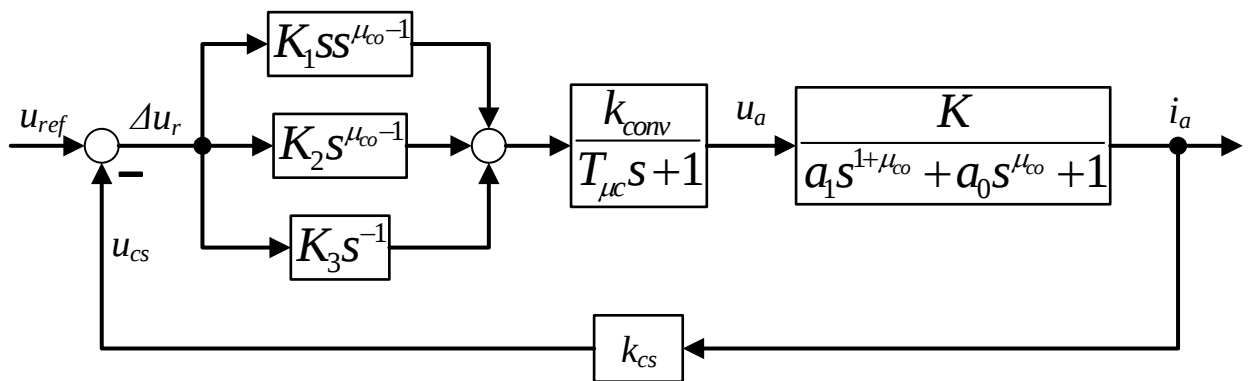


Рис. 2.22. Структурна схема замкнутої системи керування з D^μПІ-регулятором

Видно, що динамічні показники системи краще, ніж в будь-якому з випадків на рис. 2.21.

Підбір значень коефіцієнтів для систем керування, налаштованих на дробовий порядок астатизму $\mu=0.6$ та $\mu=\mu_{co}$, виконувався за структурними схемами регуляторів на рис. 2.24, а і рис. 2.24, б відповідно. В регуляторі на рис. 2.24, а додано фільтр з постійною часу $T_f=0.0006$ s для подавлення шумів від диференційної складової. Коефіцієнти при $\mu = 0.6$ – $K_1=0.2$, $K_2=37.246$, $K_3=1.288$,

при $\mu = \mu_{co}$ – $K_1 = 0.0068$, $K_2 = 29.24$, $K_3 = 25.242$. Результати експериментальних досліджень та моделювання наведені на рис. 2.25, а і рис. 2.25, б.

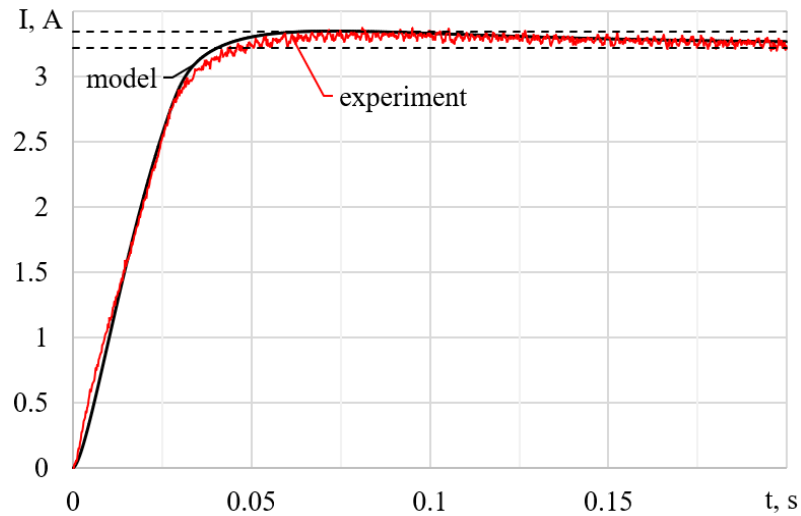


Рис. 2.23. Графіки перехідних процесів з оптимізованим $D^{\mu}PI$ -регулятором

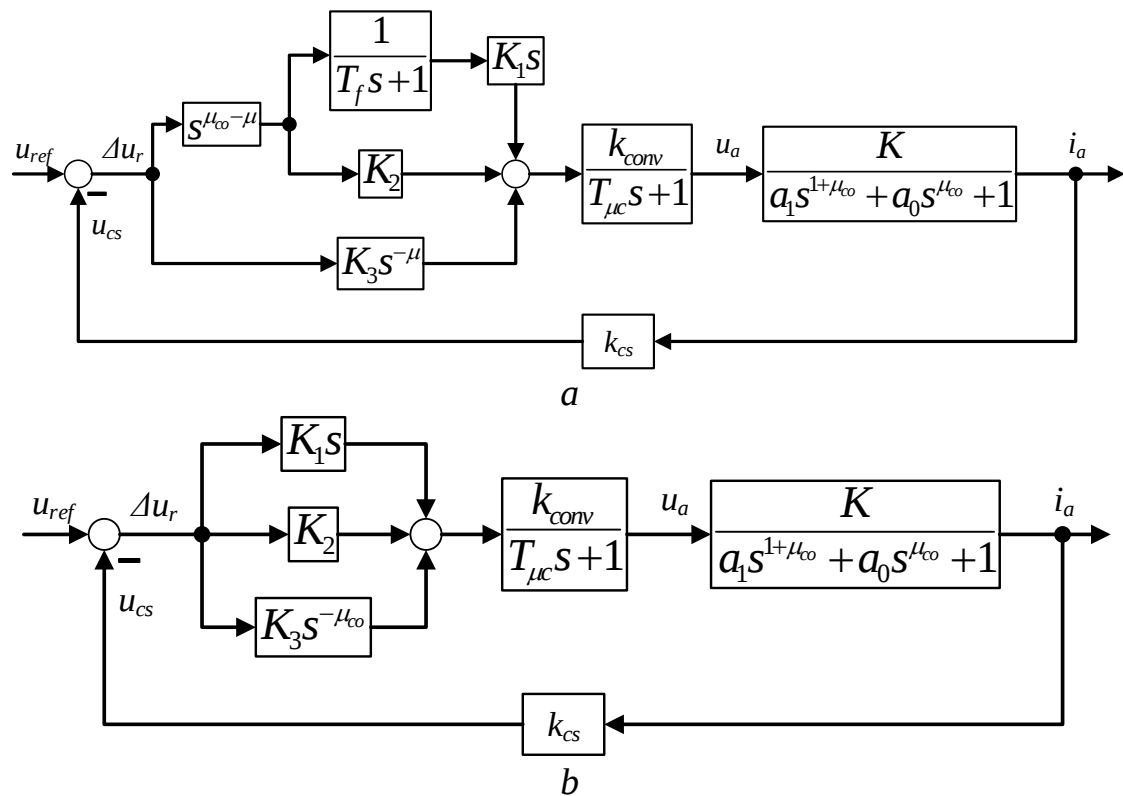


Рис. 2.24. Структурна схема замкнутої системи керування з астатизмом дробового порядку: а – $\mu = 0.6$; б – $\mu = \mu_{co}$

У порівнянні з модульним оптимумом рівень шумів підвищується, а стале значення дещо менше заданого. Час першого узгодження $t_c=0.0344$ s, що робить такі системи більш швидкодіючими. Але кращими якостями перехідного процесу має варіант $\mu=\mu_{co}$ – рівень шумів та перерегулювання зменшено. Крім того, в підпрограмі регулятора немає необхідності розраховувати сигнали двох дробових інтеграторів різного порядку.

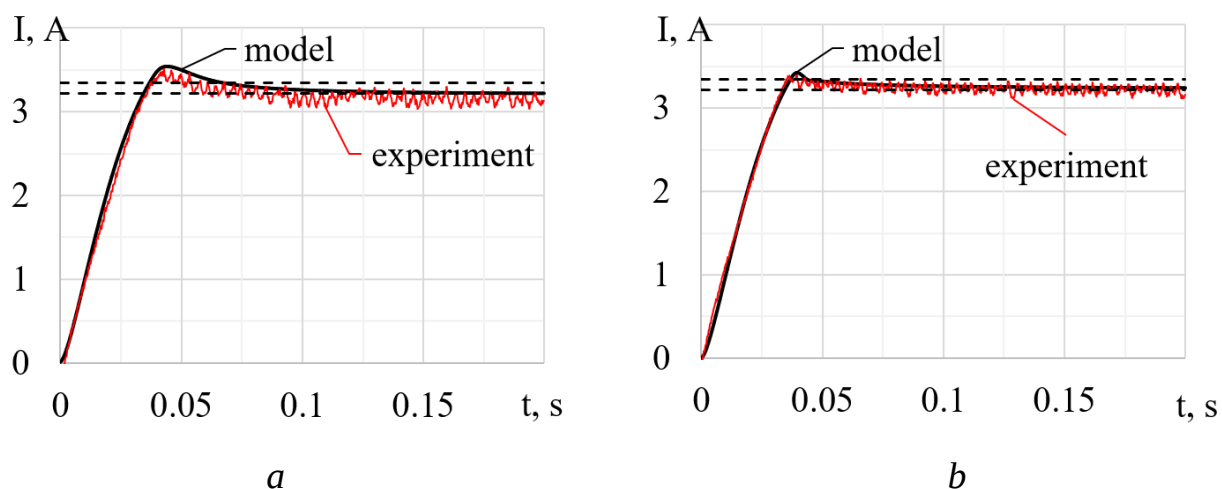


Рис. 2.25. Графіки перехідних процесів при налаштуванні з дробовим порядком астатизму: $a - \mu=0.6$; $b - \mu=\mu_{co}$

Проведено аналогічні дослідження для системи з порядками астатизму $\mu=1+\mu_{co}$ и $\mu=1.5$, отримані регулятори зі структурними схемами, показаними на рис. 2.26 і відповідні перехідні процеси (рис. 2.27).

Коефіцієнти регуляторів при $T_f=0.0006$ s: для $\mu=1.5$ – $K_0=0.393$, $K_1=0.118$, $K_2=29.31$, $K_3=66.22$, $K_4=170.54$, $K_5=1.969$; для $\mu=1+\mu_{co}$ – $K_1=0.0193$, $K_2=16.63$, $K_3=44.19$, $K_4=26.52$, $K_5=4.452$. Видно, що налаштування на $\mu=1.5$ має декілька затягнутий процес в області $\pm 2\%$ від заданого значення і на відміну від моделі відсутнє перерегулювання. У разі налаштуванні на $\mu=1+\mu_{co}$ фактично відразу досягається стале значення з часом першого узгодження $t_c=0.0344$ s без перерегулювання.

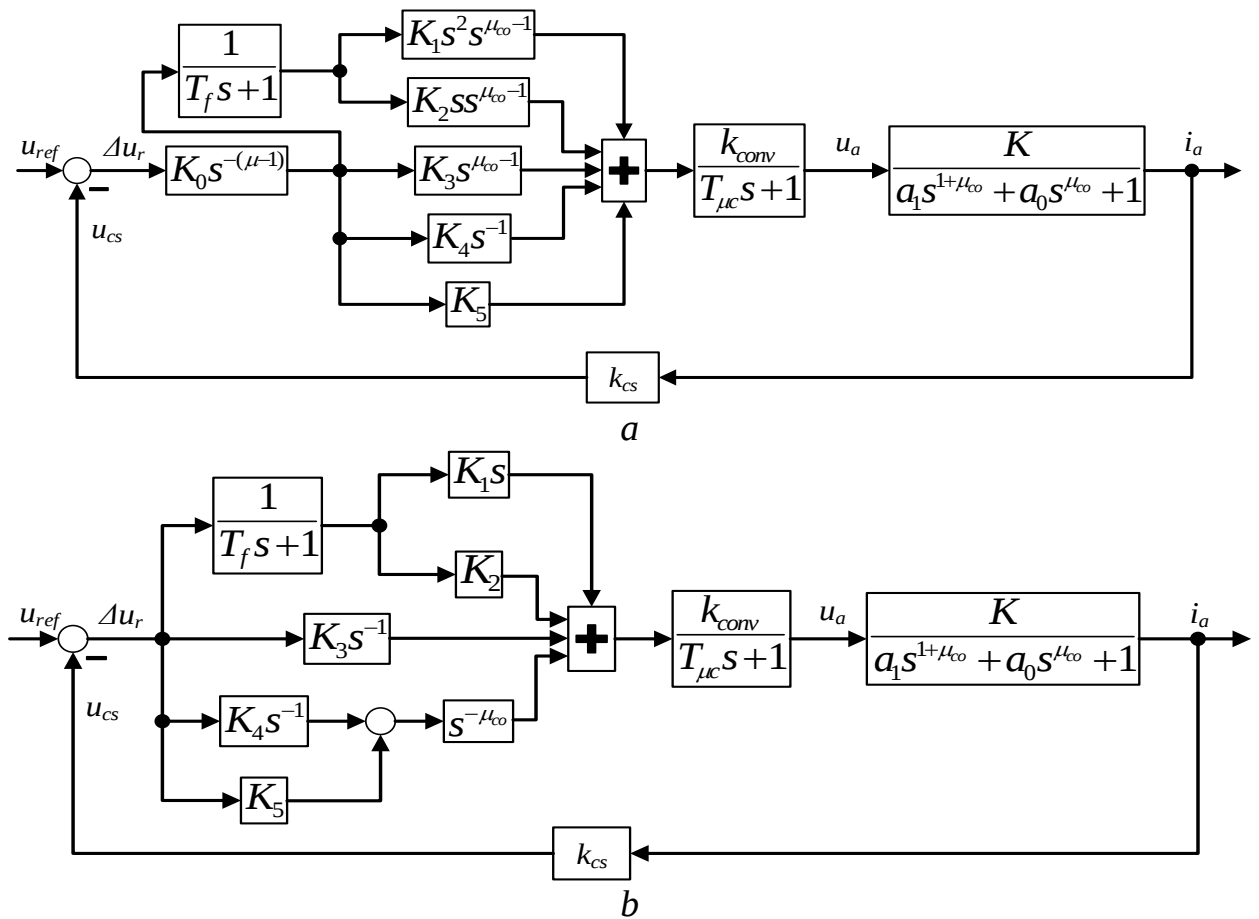


Рис. 2.26. Структурна схема замкнутої системи керування з дробовим порядком астатизму: $a - \mu = 1.5$; $b - \mu = 1 + \mu_{co}$

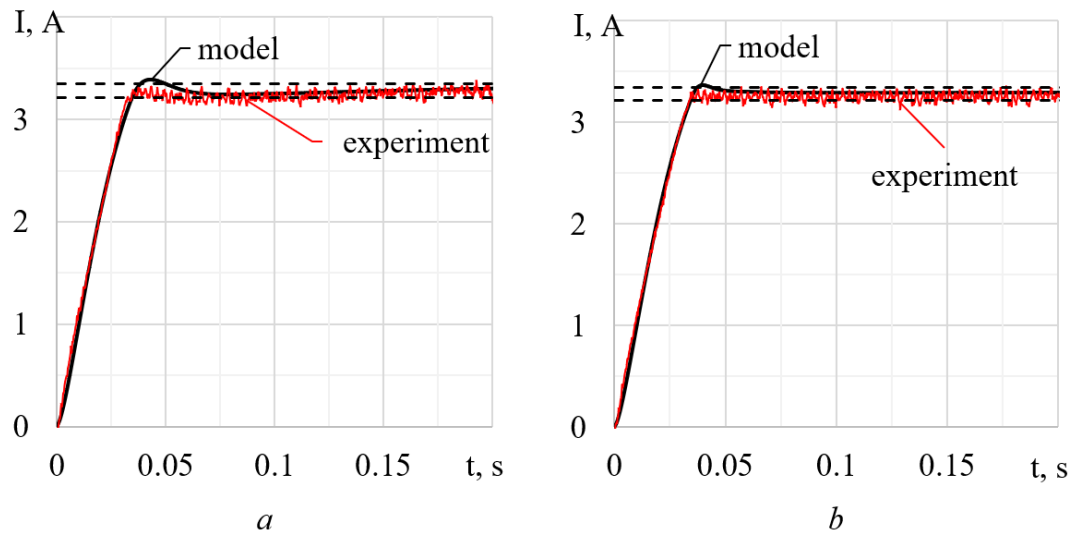


Рис. 2.27. Графіки перехідних процесів замкнутої системи керування з дробовим порядком астатизму: $a - \mu = 1.5$; $b - \mu = 1 + \mu_{co}$

Всі перераховані вище результати ідентифікації, моделювання та експериментів проводилися при відносно великому струмі ($2I_n$), щоб за допомогою дробового числення врахувати нелінійність, викликану насиченням. Але що станеться, якщо двигун буде працювати в лінійній області, де струм менше або дорівнює номінальному 1.6 А. Щоб це перевірити, було проведено моделювання та експеримент з сигналом завдання вдвічі меншим (рис. 2.28, а). Структура регулятора, його параметри були незмінні та відповідали налаштуванню при $\mu=1+\mu_{co}$. Також перевірена реакція системи на ступінчасті зміни завдання (рис. 2.28, b).

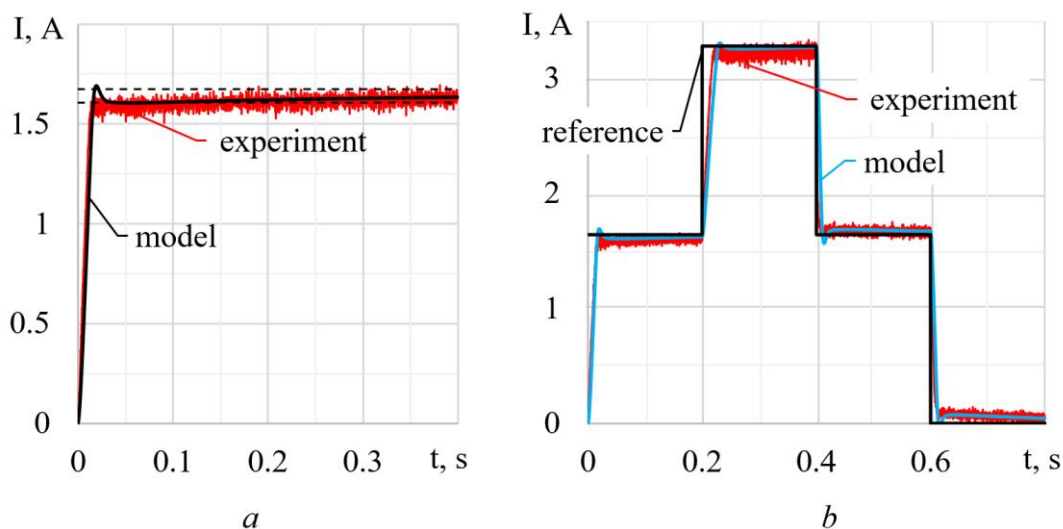


Рис. 2.28. Графіки перехідних процесів з регулятором для $\mu=1+\mu_{co}$: *a* – при сигналі завдання 1.6 А; *b* – при ступінчастій зміні сигналу завдання

Як видно, при моделюванні та в експериментах якісні показники системи залишаються незмінними і відповідають бажаним налаштуванням. У цьому полягає найважливіша відмінність від лінійної системи з РІ-регулятором (див. рис. 2.16).

Таким чином досліджено різні налаштування замкнутого контуру струму з цілочисельним та дробовим порядком астатизму для двигуна послідовного збудження. Зіставлені результати моделювання та експериментальних досліджень

при налаштуваннях на модульний оптимум та з дробовими порядками астатизму μ_{co} , 0.6, $1+\mu_{co}$, 1.5 (рис. 2.23, 2.25, 2.27, 2.28). Отримані дані свідчать, по-перше, про те, що обрана математична модель об'єкта управління добре узгоджується з експериментом, особливо при струмі вище номінального в 2 рази [75]. Навіть в лінійній області (до рівня номінального струму) розбіжності відносно невеликі – в моделі системи виникає невелике перерегулювання, що не виявлене в експериментах (рис. 2.28, *a*).

По-друге, кожне налаштування має свої переваги і недоліки [75]:

- при структурі регулятора D^PI (рис. 2.23) перехідний процес практично не має перерегулювання – менше 2%, але затягнутий у часі в області $\pm 2\%$ від заданого значення;
- при налаштуванні на дробовий порядок астатизму 0.6 і μ_{co} через наявність в регуляторі диференційної ланки посилюються шуми, що викликало при певних значеннях коефіцієнтів нестійкість і підвищену коливальність. Це вимагає ввести обмеження цих коефіцієнтів для генетичного алгоритму. Найкращий результат для регуляторів під ці налаштування показані на рис. 2.25, *a* і рис. 2.25, *b*. Час першого узгодження менше на 0.006 s, ніж в попередньому випадку. Але спостерігається коливальність, перерегулювання, в сталому режимі з'являється помилка;
- при налаштуванні на дробовий порядок астатизму 1.5 структура регулятора виходить складніше (рис. 2.26, *a*). Однак це дозволяє компенсувати ті недоліки, які виходили у випадку з дробовим астатизмом 0.6 і μ_{co} . Перерегулювання невелике, час першого узгодження 0.0344 s, але є затягування перехідного процесу в області $\pm 2\%$ від заданого значення (рис. 2.27, *a*);
- при налаштуванні на дробовий порядок астатизму $1 + \mu_{co}$ перехідний процес має найкращі показники серед попередніх налаштувань (рис. 2.27, *b*), до недоліків можна віднести тільки декілька підвищених пульсацій струму в порівнянні з регулятором D^PI .

Таким чином, можна стверджувати, що серед розглянутих варіантів регулятор з налаштуванням на дробовий порядок астатизму $1 + \mu_{co}$ дає найкращі результати в замкнутому контурі струму двигуна послідовного збудження. Експеримент з сигналами завдання $0 \rightarrow 1.64 \text{ А} \rightarrow 3.28 \text{ А} \rightarrow 1.64 \text{ А} \rightarrow 0$ показує, що дане налаштування також може бути застосована у всьому діапазоні регулювання струму.

ВИСНОВКИ ПО РОЗІЛУ 2

1. При моделюванні контуру струму двигуна Д-12 показано, що апроксимація інерційною ланкою не дає бажаних якостей перехідних процесів при подальшому синтезі регуляторів, тому необхідно скорегувати значення коефіцієнтів відповідних регуляторів. Налаштування на дробовий порядок астатизму дало кращий результат, ніж класичне налаштування на модульний оптимум.
2. Незважаючи на різну потужність двигунів при моделюванні та при експериментальному дослідженні перехідний процес з $2I_n$ має схожий характер (рис. 2.4 та 2.17).
3. Експериментальні дослідження ДПС ПЗ при струмі $2I_n$ і подальша обробка даних показали, що найбільш точно якірне коло двигуна описується дробово-диференціальним рівнянням порядку $1 + \mu_{co}$ ($\mu_{co} \approx 0.35$) і, відповідно, передавальною функцією:
$$W_{co3} = \frac{K}{a_1 s^{1+\mu} + a_0 s'' + 1}.$$
4. Проведено дослідження замкнутої системи з налаштуваннями на модульний оптимум, на оптимум з порядками астатизму 0.6, μ_{co} , 1.5, $1 + \mu_{co}$. Визначено відповідні структурні схеми та параметри регуляторів. Найкращими

динамічними і статичними показниками характеризується система з порядком $1 + \mu_{co}$.

5. Показано, що така замкнута система зберігає оптимальні властивості при різних рівнях завдання від 0 до $2I_n$, що характеризує її як самоподібну. Відповідно, стає допустимим аналіз і синтез класичними методами теорії автоматичного керування для лінійних систем.

РОЗДІЛ 3

РЕГУЛЯТОРИ З ДРОБОВИМ ПОРЯДКОМ ІНТЕГРУВАННЯ ТА ДИФЕРЕНЦЮВАННЯ В СИСТЕМІ КЕРУВАННЯ ШВИДКІСТЮ ВЕНТИЛЬНО-РЕАКТИВНОГО ДВИГУНА

3.1. Аналіз об'єкта

Область застосування вентильно-реактивних двигунів (ВРД) в останні роки істотно розширилася, про що свідчить активне зростання патентів, пов'язаних з цим напрямком досліджень [76, 77]. Причини такої активності пояснюються тим, що виробництво ВРД і використовувані матеріали виявляються дешевше навіть виробництва асинхронних двигунів (в роторі немає обмоток, в обмотці статора використовується міді на 40-50% менше), технологія виготовлення простіше. В середньому, при однакових електричних та вагогабаритних характеристиках, ВРД має в 4 рази меншу вартість, більш широкий діапазон швидкостей обертання, більш високий діапазон робочих температур, ніж вентильні двигуни з постійними магнітами [78]. Причиною даних переваг є явнополюсна структура цих двигунів. Але це ж одночасно стає недоліком, так як електромеханічний перетворювач працює в імпульсному режимі, що призводить до акустичних шумів, а магнітне насичення ускладнює управління через нелінійні залежності [77].

Використання класичних лінійних PID-регуляторів не забезпечує бажану якість перехідних процесів, так як диференціальні рівняння, що описують динамічні процеси в ВРД, нелінійні і змінюються в залежності від кута повороту і швидкості обертання [79].

Публікацій, пов'язаних з керуванням ВРД, досить багато. Наприклад, в контурі моменту (струму) найбільш перспективними вважаються методи TSF (torque shared function) і DITC (direct instantaneous torque control) [80 – 82]. У

контурі швидкості використовуються нелінійні PID-регулятори [83], а також інтелектуальні методи контролю, такі як fuzzy-регулятори [84], нейронні мережі [85] та комбінації цих методів [86 – 88].

В останні десятиліття більш активно застосовують дробове числення в техніці та в теорії автоматичного керування (ТАК). Однією з причин є те, що дробовий порядок диференціальних рівнянь точніше описує деякі фізичні процеси (акустика, електроніка, термодинаміка і багато інших) [89 – 91], особливо ті, в математичної моделі яких присутні ступеневі залежності. Також $PI^{\gamma}D^{\mu}$ -регулятори з дробовим порядком інтегрування та диференціювання дають можливість вибрати додаткові опції γ і μ . Іншою перевагою таких регуляторів є підвищений запас стійкості в порівнянні з цілочисельними PID. Так як синтез регуляторів з дробовим ступенем схожий з традиційними методами, то це спрощує пошук коефіцієнтів для окремих складових цих регуляторів.

Мета цього розділу дисертації – синтез системи керування вентильно-реактивним двигуном на базі дробово інтегрально-пропорційних регуляторів, що забезпечують підвищення якісних показників роботи в динамічних і статичних режимах.

Для досягнення цієї мети необхідно вирішити задачу ідентифікації вентильно-реактивного двигуна моделлю, яка побудована на базі диференціальних рівнянь дробового порядку, що дозволить врахувати нелінійні залежності магнітного потоку, обертового моменту та вентиляторний характер навантаження. Це дозволить вирішити друге завдання – синтезувати регулятори з дробовим порядком інтегрування і диференціювання на підставі стандартних методів теорії автоматичного управління.

3.2. Математична модель контуру швидкості вентиляно-реактивного двигуна

Вентильно-реактивний двигун в основному характеризується нелінійними залежностями із-за його роботи в області магнітного насичення та дискретної роботи фаз. Немає точного аналітичного рівняння для опису перетворення електричної енергії в механічну в цьому двигуні. Тому характеристики магнітного потоку і статичного моменту від струму в фазі і положення ротора для кожного ВРД є індивідуальними даними. Ці дані широко використовуються, як для моделювання, так і з метою керування [77]. Вони можуть бути отримані чисельним методом або експериментальним [79]. Один з найбільш поширених чисельних методів отримав назву метод скінчених елементів (МСЕ). При розрахунку магнітної системи електродвигуна вона розбивається на області, число яких не обмежується. І вже в цій області вирішуються рівняння Максвелла:

$$\begin{cases} \nabla \times H = J + \frac{\partial D}{\partial t}, \\ \nabla \times E = -\frac{\partial B}{\partial t}, \\ \nabla \cdot D = \rho, \\ \nabla \cdot B = 0. \end{cases} \quad (3.1)$$

де ∇ – диференційний оператор набла, H – напруженість електричного поля, V/m; J – щільність електричного струму, A/m²; D – електрична індукція, C/m²; E – напруженість електричного поля, V/m; B – магнітна індукція, Т; ρ – об'ємна щільність електричного заряду, C/m³.

Для проведення польових розрахунків була використана програма ANSYS Maxwell. Вона дозволяє проводити моделювання електромагнітного поля для проектування і аналізу різних типів електродвигунів, приводів, датчиків,

трансформаторів і інших електромагнітних і електромеханічних пристроїв.

Програма дозволяє вирішувати наступні види завдань:

- магнітні перехідні процеси з рухом (Magnetic transient with rigid motion);
- магнітостатика;
- електростатика;
- змінні магнітні поля (AC electromagnetic);
- провідність постійного струму;
- електричні перехідні процеси;
- інтерфейс для проектування електричних машин і трансформаторів (Expert design interfaces for electric machines and transformers);

Основна геометрія ВРД будувалася в програмі для креслення AutoCAD (рис. 3.1), а далі імпортувалася в середовище ANSYS.

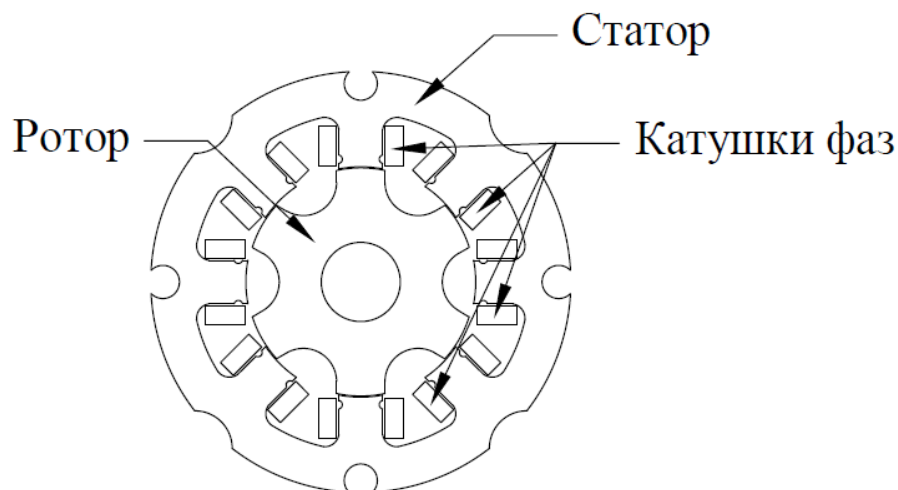


Рис. 3.1. Конструкція вентильно-реактивного двигуна

Детальні дані про основні параметри вентильно-реактивного двигуна представлені в табл. 3.1.

Таблиця 3.1. Основні параметри ВРД

Параметр	Значення
Кількість полюсів статора	8
Кількість полюсів ротора	6
Діаметр ротора, mm	28.7
Активна довжина ротора, mm	30
Діаметр розточки статора, mm	29
Внутрішній діаметр вікна статора, mm	42.5
Зовнішній діаметр статора, mm	53
Ширина зубця статора, mm	5.5
Ширина зубця ротора, mm	6.1
Число витків обмотки фази	60
Напруга живлення, V	24.2
Сумарна маса сталі, kg	0.28
Момент інерції ротора, $\text{kg} \cdot \text{m}^2$	$7.3 \cdot 10^{-6}$
Активний опір фази постійного струму, Ω	0.68
Сталь марки	2211
Питомий опір сталі, $\Omega \cdot \text{m}$	$250 \cdot 10^9$
Товщина листа, mm	0.5

Для завдання властивостей магнітного матеріалу використовувалася характеристика намагнічування листової ізотропної сталі марки 2211 (рис. 3.2).

На рисунку 3.3 зображена модель ВРД, яка вже імпортована в ANSYS Maxwell.

При польову модель розбито приблизно на 100000 елементів. Це дозволяє з достатнім ступенем точності апроксимувати поперечний розріз електромеханічного перетворювача. Рисунок кінцевих елементів (трикутників), магнітних ліній і щільність магнітної індукції при узгодженому, неузгодженому і проміжному положенні ротора показані на рис. 3.4 зі струмом в фазі $i_{ph} = 5 \text{ A}$.

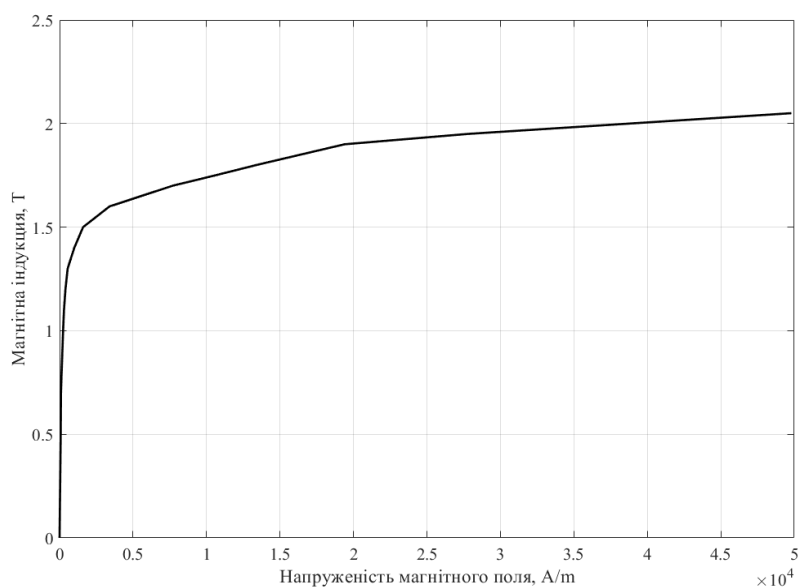


Рис. 3.2. Крива намагнічування для сталі марки 2211

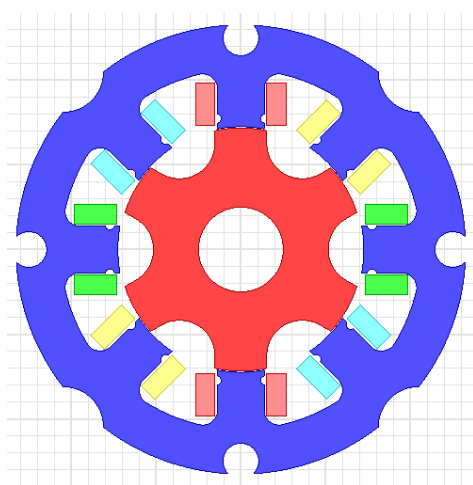


Рис. 3.3. Модель ВРД в ANSYS Maxwell

Нелінійна залежність по індуктивності розраховувалася в діапазоні струму фази i_{ph} від 0 А до 20 А з кроком 0.5 А і кута повороту ротора Θ від 0° до 60° з кроком 0.5° . При цьому зміщення ротора починалося від неузгодженого положення 0° до узгодженого 30° і заново до неузгодженого 60° , але вже проти годинникової стрілки. В результаті розрахунку польовій моделі вентильно-реактивного двигуна отримані залежності:

$$\begin{aligned} L_{ph} &= f(i_{ph}, \Theta) \\ M_{ph} &= f(i_{ph}, \Theta) \end{aligned} \quad (3.2)$$

де i_{ph} , L_{ph} , M_{ph} – струм, індуктивність і момент фази відповідно; Θ – кутове положення ротора.

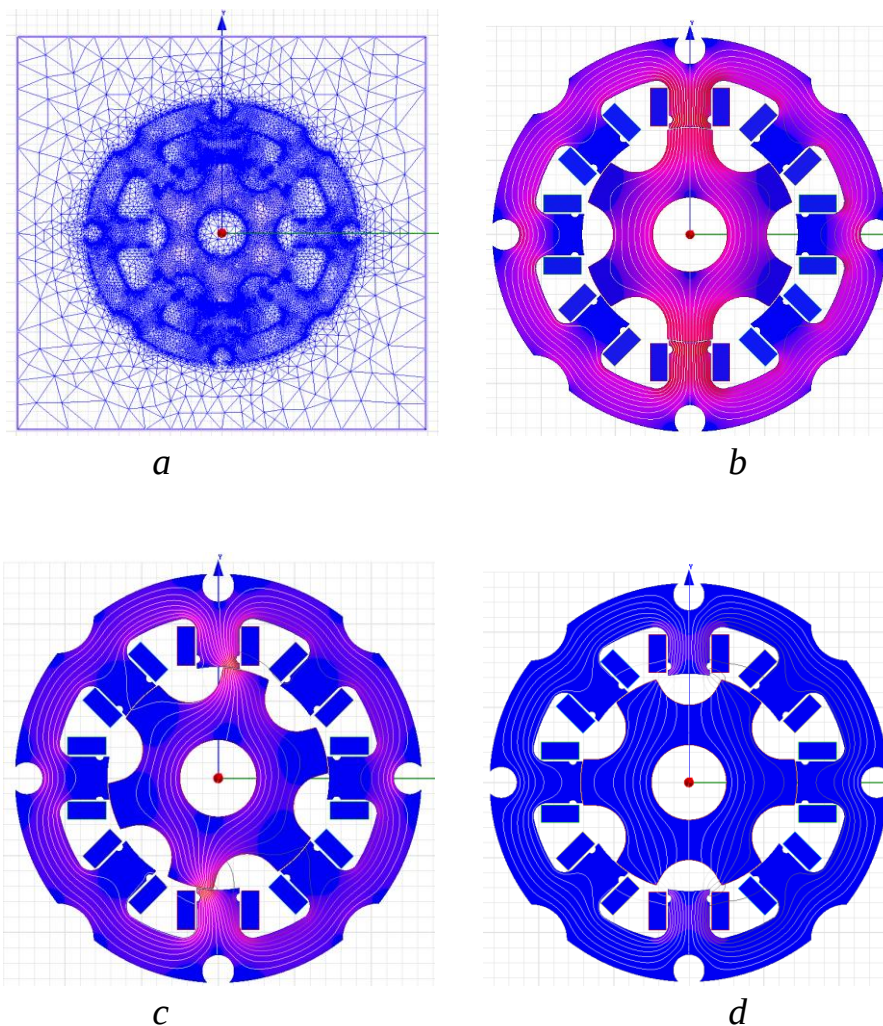


Рис. 3.4. Апроксимація трикутниками поперечної геометрії двигуна (а), магнітні лінії і щільність магнітної індукції при узгодженому положенні (b), проміжному (c) та неузгодженому (d)

Графіки залежностей (3.2) представлені на рисунку 3.5 і 3.6. На основі отриманих даних будується динамічна модель ВРД.

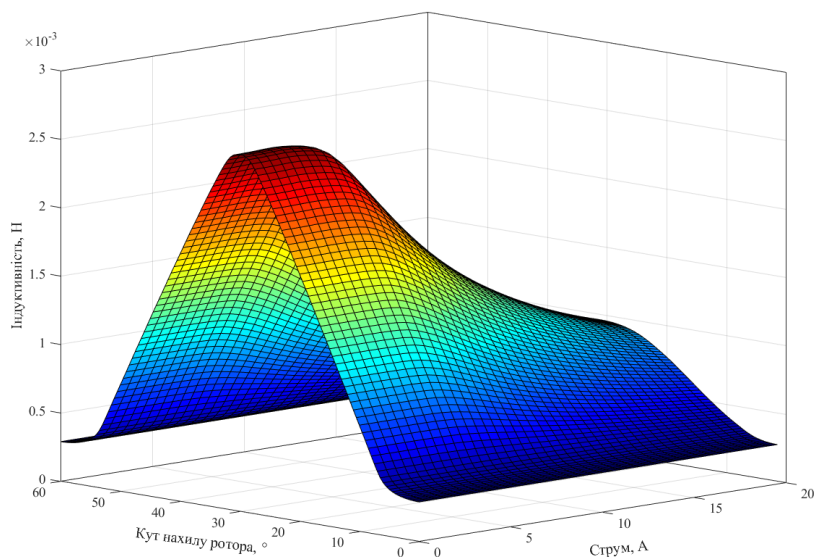


Рис. 3.5. Графік залежності $L_{ph}=f(i_{ph}, \Theta)$

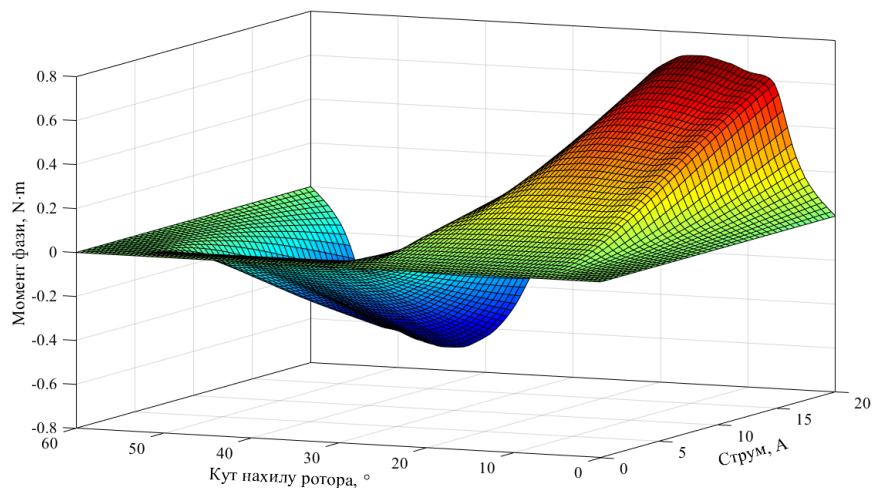


Рис. 3.6. Графік залежності $M_{ph}=f(i_{ph}, \Theta)$

З рисунку 3.5 і 3.6 видно, що чим ближче ротор до узгодженого стану і вище значення струму, тим характеристики $L_{ph}=f(i_{ph}, \Theta)$ та $M_{ph}=f(i_{ph}, \Theta)$ приймають істотно нелінійний вигляд, який пояснюється насиченням магнітної системи двигуна [92].

Визначення нелінійних залежностей дозволяють побудувати найбільш точну модель для вентиляно-реактивного двигуна. Для спрощення опису процесу перетворення енергії були зроблені наступні допущення в моделі електродвигуна:

- фази магніто-ізолювані один від одного;
- при визначенні сумарного моменту, що діє на ротор, застосовується принцип суперпозиції;
- ідеалізовані силові ключі;
- внутрішній опір джерела дорівнює нулю;
- зниженням моменту ротора від опору повітря нехтуємо.

Дані припущення дозволяють розглядати кожну фазу окремо і створюваний момент кожної з них підсумовувати з іншими. Диференціальні рівняння для окремої фази:

$$\begin{cases} u_{ph} = i_{ph} R_{ph} + \frac{d\psi_{ph}}{dt} \\ M_{ph} = M_s + M_{idle} + J_{\Sigma} \frac{d\omega}{dt} \\ \psi_{ph} = f(i_{ph}, \Theta) \\ M_{ph} = f(i_{ph}, \Theta) \end{cases} \quad (3.3)$$

де u_{ph} , i_{ph} , R_{ph} , ψ_{ph} , M_{ph} – напруга, струм, опір, потокозчеплення, момент фази, відповідно; M_s – момент опору на валу двигуна; M_{idle} – момент холостого ходу; J_{Σ} – сумарний момент інерції, приведений до валу; Θ – кутове положення; ω – кутова частота обертання; t – час.

Так як $\psi_{ph} = L_{ph} \cdot i_{ph}$ і нас цікавить тільки контур моменту, тому можна записати:

$$\begin{cases} \frac{d\psi_{ph}}{dt} = u_{ph} - i_{ph} R_{ph} \\ i_{ph} = \frac{\psi_{ph}}{L_{ph}} \\ L_{ph} = f(i_{ph}, \Theta) \\ M_{ph} = f(i_{ph}, \Theta) \end{cases} \quad (3.4)$$

Виходячи з системи рівнянь (3.4), можна побудувати структурну схему для однієї фази, зображену на рис. 3.7.

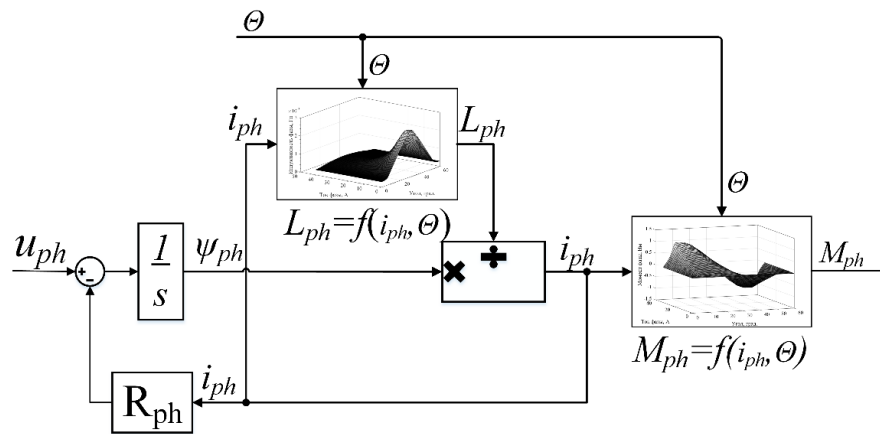


Рис. 3.7. Структурна схема однієї фази ВРД

З принципу суперпозиції моменту, структурна схема моделі ВРД складена з чотирьох таких блоків, а також доповнена блоком визначення кута для кожної фази відносно ротора (рис. 3.8). Момент холостого ходу двигуна приймався рівним 5% від номінального значення:

$$M_{idle} = 0.05 \cdot M_n = 0.05 \cdot 0.05 = 2.5 \text{ mN} \cdot \text{m} \quad (3.5)$$

На вхід блоків фази (рис. 3.8) подається напруга від інвертора. Це значення залежить від керуючих сигналів транзисторів VT1 і VT2 полумостового інвертора і напрямку струму. Варіанти стану силового інвертора показані на рис. 3.9.

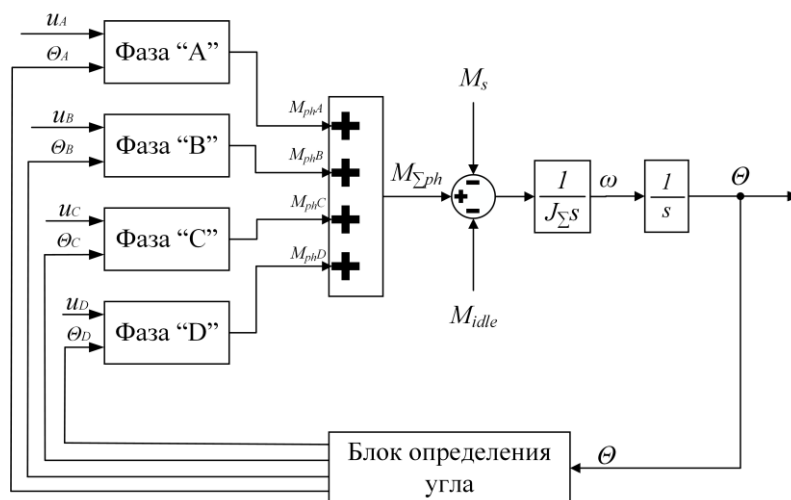


Рис. 3.8. Структурна схема вентильно-реактивного двигуна

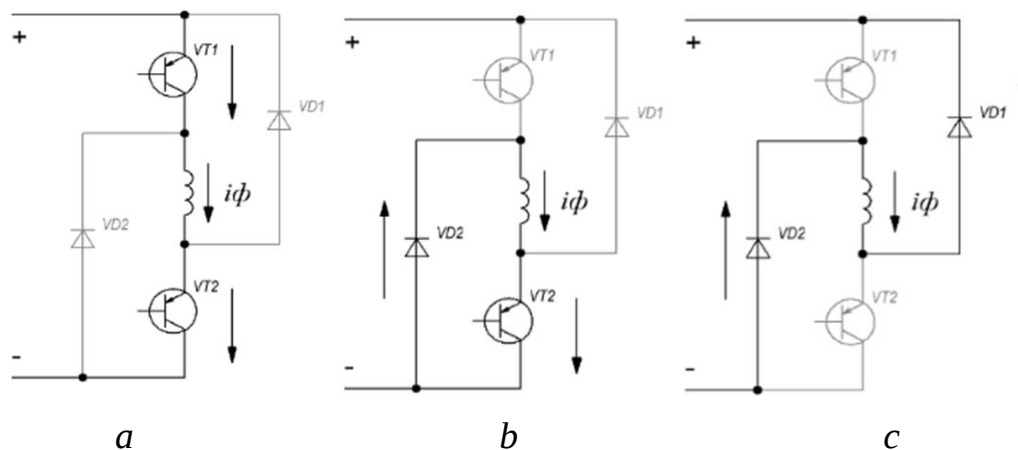


Рис. 3.9. Діаграми роботи інвертора при різних сигналах керування: *a* – режим «1», *b* – режим «0», *c* – режим «-1»

У режимі «1» обидва транзистора VT1 і VT2 включені та до котушки прикладається позитивна напруга від джерела живлення, струм зростає. Коли задається режим «0», відкривається один транзистор VT2 і запасена енергія

магнітного поля за рахунок електрорушійної сили (ЕРС) самоіндукції фази призводить до виникнення струму, який замикається через діод VD2 – і буде повільно зменшуватися (к фазі двигуна прикладається нульова напруга). Запасена в фазі енергія розсіюється на активному опорі, силовому транзисторі VT2 і зворотному діоді VD2. У разі режиму «-1» обидва транзистора VT1 і VT2 закриті і струм спадає через зворотні діоди VD1 і VD2. До котушці прикладається негативна напруга – струм швидко зменшується, при цьому збережена енергія магнітного поля в фазі повертається в джерело живлення.

Структурній схемі на рисунку 3.8 цілком відповідає математична модель Switched Reluctance Machines з бібліотеки Simscape / Power Systems / Simscape Components / Machines / Reluctance Machines. Внутрішня структура моделі відповідає системі рівнянь (3.3) і використовує табличні залежності (3.2) з лінійною апроксимацією при обчисленні проміжних значень. Але в даній роботі використовувався інший підхід – була написана програма в середовищі MATLAB. Диференціальні рівняння вирішувалися чисельними методами, а інтерполяція в нелінійних функціях $L_{ph}=f(i_{ph}, \Theta)$ та $M_{ph}=f(i_{ph}, \Theta)$ виконувалася через функцію **interp2()**.

Очевидно, що модель включає в себе нелінійні залежності, які істотно ускладнюють процедуру синтезу регуляторів.

3.3. Ідентифікація параметрів об'єкта керування передавальною функцією з дробовим порядком

Проведемо моделювання з різною прикладеною до двигуна напругою і вентиляторним навантаженням з моментом інерції збільшеним в два рази. При цьому кути включення і відключення рівні $\alpha_{on} = 7.5^\circ$, $\alpha_{off} = 22.5^\circ$. На рисунку 3.10 зображені перехідні процеси за швидкістю вентилярно-реактивного двигуна при напругах 4, 8, 12, 16, 20 и 24 V. На рисунку 3.11 показана механічна

характеристика вентиляторного навантаження, при якій проводилося моделювання.

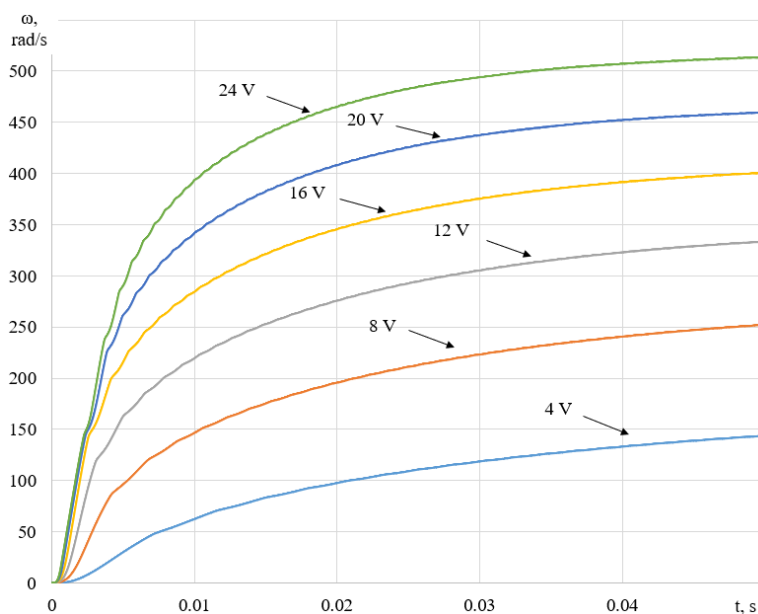


Рис. 3.10. Перехідні процеси за швидкістю з різними прикладеними напругами

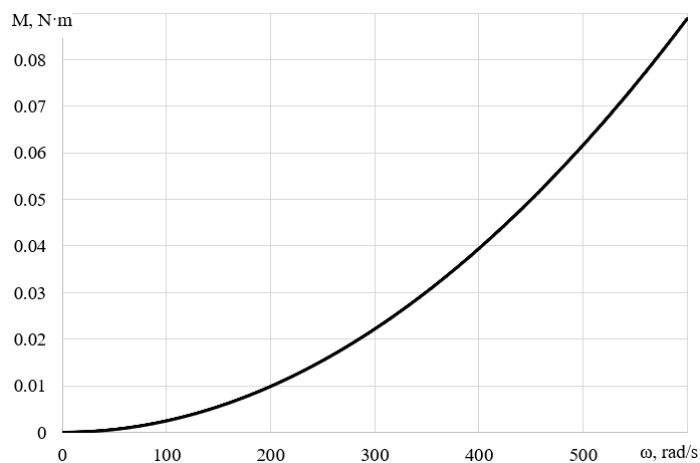


Рис. 3.11. Вентиляторне навантаження на валу двигуна

З рисунку 3.10 видно, що в залежності від величини напруги, яка прикладена до двигуна, змінюється характер його перехідного процесу. Виконаємо ідентифікацію об'єкта керування за отриманими ПП з рис. 3.10. Для

математичного опису перехідного процесу була розглянута наступна передавальна функція:

$$W_{co1} = \frac{K}{a_0 s^\mu + 1} \quad (3.6)$$

Для ідентифікації невідомих коефіцієнтів (3.6) використано генетичний алгоритм, в якому при селекції обрано турнірний метод, нові особини успадковували гени батьків шляхом рівномірного схрещування, ймовірність мутації в хромосомі – 20%. Оцінкою пристосованості особини служила середньоквадратична помилка F .

Результат ідентифікації передавальної функції W_{co1} при різних напругах двигуна зведені в табл. 3.2. Пошук коефіцієнта a_0 проводився від 0.001 до 0.5 з кроком 0.0001, для K – з 10.0 до 100 з кроком 0.01 і для μ – з 0.3 до 1.8 з кроком 0.0001.

Таблиця 3.2. Результати ідентифікації передавальної функцій W_{co1}

Параметр	Передавальна функція об'єкта керування W_{co1}					
	4 V	8 V	12 V	16 V	20 V	24 V
μ	0.9548	0.7098	0.6602	0.629	0.7392	0.7849
a_0	0.0487	0.1141	0.1043	0.0938	0.0457	0.0366
K	40.82	42.86	36.7	31.45	25.64	23.58
F	0.0193	0.0234	0.0264	0.0298	0.0249	0.0246

Графіки моделі ВРД і результати апроксимації передавальною функцією W_{co1} зіставлені на рис. 3.12.

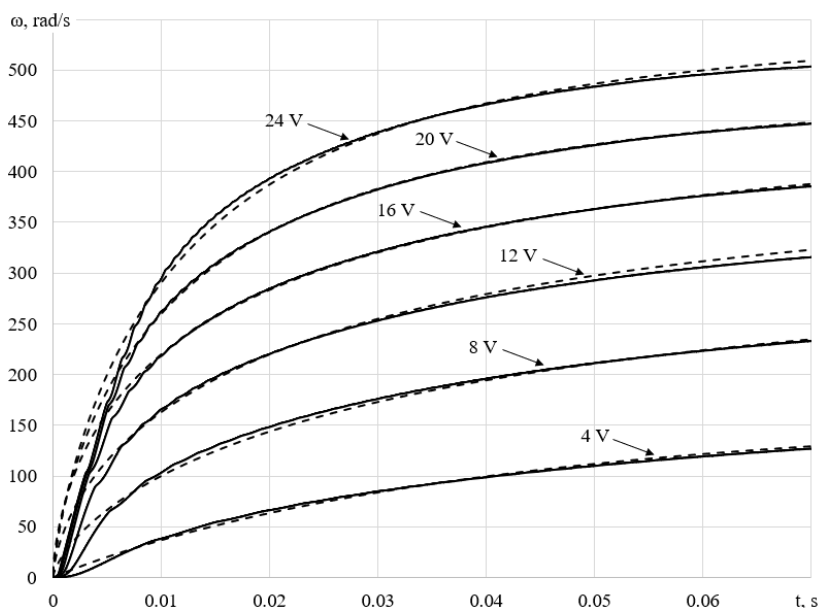


Рис. 3.12. Графіки перехідних процесів швидкості за передавальною функцією W_{col} (пунктир) і за моделлю ВРД (суцільна лінія)

Як видно з таблиці 3.2 при різних прикладених напругах об'єкт керування описується з високою точністю за допомогою дробово-аперіодичній функції, порядок якої лежить в межах від 0.629 до 0.9548. Візьмемо середню величину дробового порядку, що спростить синтез регуляторів:

$$\mu_{ave} = \frac{0.629 + 0.6602 + 0.7098 + 0.7392 + 0.7849}{5} = 0.705 \approx 0.71 \quad (3.7)$$

Далі використовуючи генетичний алгоритм, де у особин будуть постійний дробовий порядок $\mu_{ave} = 0.71$, знайдемо значення коефіцієнтів a_0 і K , які будуть скориговані таким чином, щоб відповідати отриманим перехідним процесам (табл. 3.3).

Як видно з рис. 3.12, отримані перехідні процеси за моделлю двигуна та за передавальною функцією (3.6) досить близькі, головними відмінностями є в початковому інтервалі часу. З табл. 3.3 можна помітити, що в передавальній

функції (3.6) параметри (a_0 , K) змінюються в залежності від прикладеної до обмотки напруги. Для компенсації цього при синтезі регуляторів будемо також змінювати їх коефіцієнти залежно від сигналу завдання.

Таблиця 3.3. Результати ідентифікації передавальної функцій W_{co1} з постійним значенням дробового порядку $\mu_{ave} = 0.71$

Параметр	Передавальна функція об'єкта керування W_{co1}					
	4 V	8 V	12 V	16 V	20 V	24 V
a_0	0.2951	0.1072	0.0751	0.0602	0.0585	0.058
K	80.84	41.73	33.28	28.84	26.94	25.5
F	0.0299	0.0227	0.0205	0.0215	0.0258	0.0318

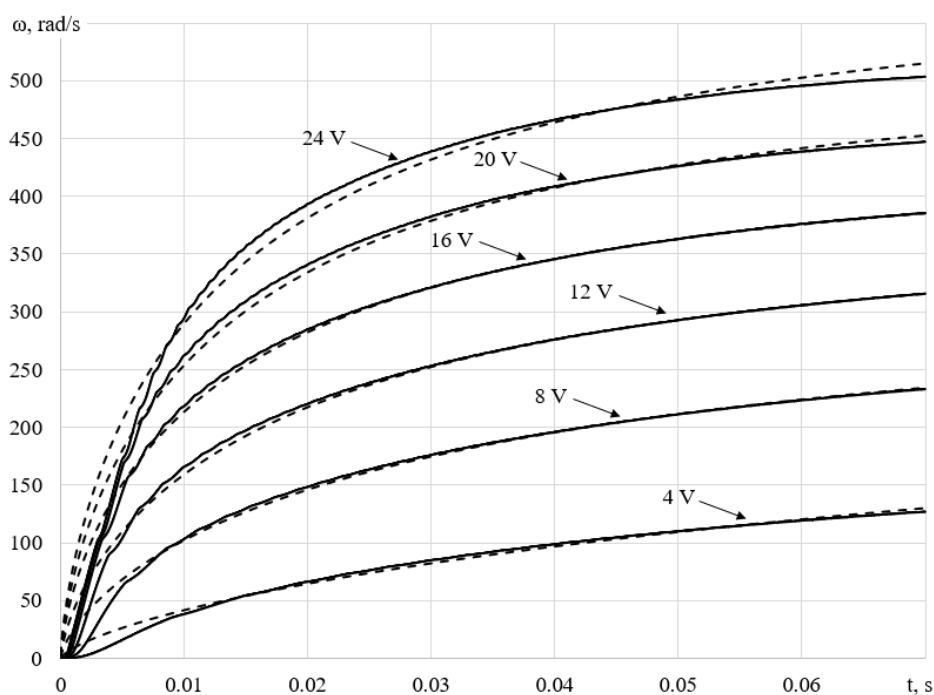


Рис. 3.13. Графіки перехідних процесів струму за передавальною функцією W_{co1} з постійним дробовим порядком $\mu_{ave} = 0.71$ (пунктир) і за моделлю ВРД (суцільна лінія)

3.4 Синтез регуляторів в контурі швидкості за моделлю з дробовим порядком

Структурна схема замкнутої системи регулювання швидкості зображена на рис. 3.14. На ній u_{ref} , u_{ss} , Δu_r , u_{fd} , ω – сигнали завдання швидкості і зворотного зв'язку, помилка між ними, напруга обмотки збудження і кутова швидкість двигуна відповідно. Також $W_{sr}(s)$, $W_{conv}(s) = \frac{k_{conv}}{T_{fr}s + 1}$, $W_{ss}(s) = k_{ss}$, $W_{co}(s)$ – передавальні функції регулятора швидкості, перетворювача, датчика швидкості та об'єкта керування, відповідно.

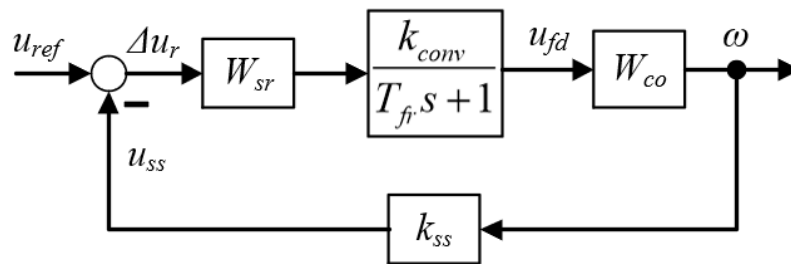


Рис. 3.14. Структурна схема системи керування швидкістю

3.4.1. Синтез регулятора при налаштуванні на модульний оптимум

Визначимо передавальну функцію регулятора швидкості $W_{sr}(s)$, виходячи з налаштувань на модульний оптимум, з об'єктом керування, що описуються передавальною функцією W_{co1} (при цьому врахуємо, що $T_{\mu s} = T_{fr}$):

$$W_{op.s}(s) = \frac{1}{2T_{\mu s}(T_{\mu s}s + 1)} = W_{sr}(s) \frac{k_{conv}}{(T_{fr}s + 1)} \frac{K}{(a_0s^\mu + 1)} k_{ss}, \quad (3.8)$$

звідки

$$W_{sr}(s) = \frac{1}{2T_{\mu s} s (T_{\mu s} s + 1)} \cdot \frac{(T_{fr} s + 1)(a_0 s^\mu + 1)}{k_{conv} k_{ss} K} = \frac{a_0 s^\mu + 1}{2T_{\mu s} k_{conv} k_{ss} K s}. \quad (3.9)$$

Після перетворень отримаємо ПІ-регулятор

$$W_{sr}(s) = K_1 s^{\mu-1} + K_2 s^{-1}, \quad (3.10)$$

де коефіцієнти

$$K_1 = \frac{a_0}{2T_{\mu s} k_{conv} k_{ss} K}, K_2 = \frac{1}{2T_{\mu s} k_{conv} k_{ss} K}.$$

Знайдемо точні значення коефіцієнтів при різних значеннях прикладеної напруги. Для цього визначимо невідомі параметри структурної схеми. За малу некомпенсовану постійну часу контуру швидкості (в той же час і фільтра) прийmemo $T_{\mu s} = T_{fr} = 0.005 \text{ s}$.

Коефіцієнт посилення перетворювача:

$$k_{conv} = \frac{U_{\max}}{10} = 2.4 \quad (3.11)$$

Коефіцієнт посилення зворотного зв'язку за швидкістю

$$k_{ss} = \frac{\omega_{\max}}{10} = 50 \quad (3.12)$$

Коефіцієнти регулятора показані в таблиці 3.4.

Таблиця 3.4. Коефіцієнти ПІ-регулятора
в різних діапазонах завдання швидкості

Параметр	Діапазон завдання U_{ref} , V					
	[0; 1.67)	[1.67; 3.33)	[3.33; 5)	[5; 6.67)	[6.67; 8.33)	[8.33; 10)
Коефіцієнт при дробовому інтегралі, K_1	7.542	5.308	4.661	4.31	4.486	4.697
Коефіцієнт при цілочисленому інтегралі, K_2	25.557	49.518	62.08	71.65	76.69	81.03

Перехідні процеси з даними регуляторами представлені на рисунку 3.15.

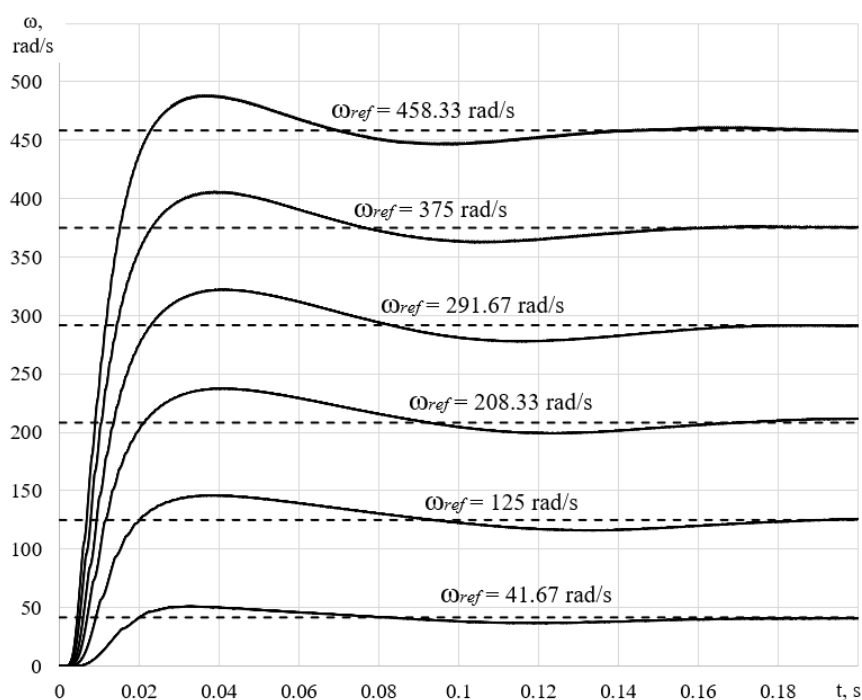


Рис. 3.15. Перехідні процеси з використанням ПІ-регулятора при розрахунку коефіцієнтів при налаштуванні на модульний оптимум

Як видно з рисунку 3.15, якість перехідних процесів не відповідає налаштуванню на модульний оптимум. Детальніші параметри ПП представлені в таблиці 3.5.

Таблиця 3.5. Якість перехідного процесу
з ПІ-регулятором при різних завданнях

Якість перехідного процесу	Завдання швидкості ω_{ref} , rad /s					
	41.67	125	208.33	291.61	375	458.33
Час першого узгодження t_{fm} , s	0.0197	0.0194	0.0204	0.0212	0.0213	0.0214
Перерегулювання δ , %	22.8	16.42	12.9	11.11	8.09	6.35

Незважаючи на незадовільні результати, з розрахунку отримано приблизні значення коефіцієнтів регулятора. Щоб покращити якість перехідного процесу підберемо інші значення для коефіцієнта при цілочисельному інтегралі (3.10). Нові його значення представлені в таблиці 3.6.

Таблиця 3.6. Значення підбраного коефіцієнта K_2 при цілочисельному інтегралі
в ПІ-регуляторі в різних діапазонах завдання швидкості

Параметр	Діапазон завдання U_{ref} , V					
	[0; 1.67)	[1.67; 3.33)	[3.33; 5)	[5; 6.67)	[6.67; 8.33)	[8.33; 10)
Коефіцієнт K_2	6.389	19.045	34.975	47.765	61.35	73.667

Результати даного коригування показані на рисунку 3.16, де порівнюються результати перехідного процесу при старому і новому значенні коефіцієнта K_2 .

Як видно з табл. 3.7, зі зміненим коефіцієнтом K_2 при цілочисельному інтегралі вдалося зменшити перерегулювання на заданому діапазоні, але при цьому збільшився час першого узгодження в середньому на 5 ms.

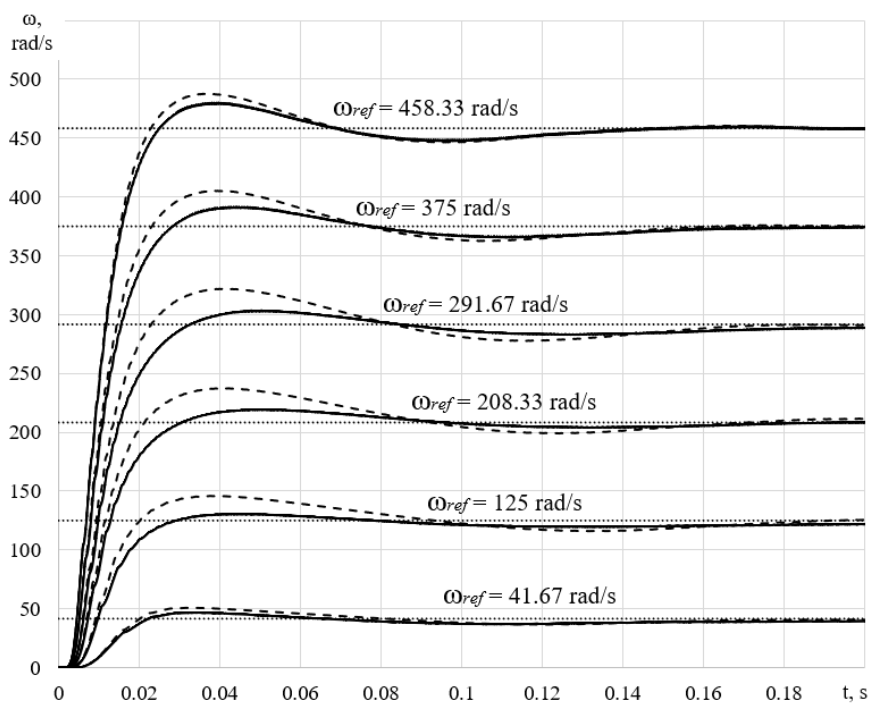


Рис. 3.16. Перехідні процеси з використанням ПІ-регулятора зі старим (пунктир) і новим (суцільна лінія) значенням коефіцієнта K_2

Таблиця 3.7. Якість перехідного процесу з ПІ-регулятором при різних завданнях зі скоригованим коефіцієнтом K_2

Якість перехідного процесу	Завдання швидкості ω_{ref} , rad /s					
	41.67	125	208.33	291.61	375	458.33
Час першого узгодження t_{fm} , s	0.0218	0.0267	0.0292	0.029	0.0259	0.0235
Перерегулювання δ , %	12.64	4.32	4.37	4.48	4.24	4.18

3.4.2. Синтез регулятора при налаштуванні на дробовий порядок астатизму

Також налаштуємо систему керування на дробовий астатизм порядку $1 + \mu$, $\mu \in (0;1)$, при якому підвищується не тільки статична, а й динамічна точність контуру. Передавальна функція розімкнутої системи має мати наступний вигляд:

$$W_{op.s}^{fo}(s) = \frac{bT_{\mu s}s+1}{abT_{\mu s}^{\mu}s^{\mu}} \frac{1}{(T_{\mu s}s^{\mu}+1)} = W_{sr}^{fo}(s) \frac{k_{conv}}{(T_{\mu s}s+1)} \frac{K}{(a_0s^{\mu_{co}}+1)} k_{ss}, \quad (3.13)$$

де значення a і b наближено вибираються за наступними співвідношеннями:

$$\begin{cases} (ba) \approx \exp(-10.27 + 7.831\mu), \\ b \approx 7.336 + 0.792(ba) + 3.83\ln(ba). \end{cases}$$

звідки

$$W_{sr}^{fo} = \frac{bT_{\mu s}s+1}{abT_{\mu s}^{\mu}s^{\mu}} \frac{(T_{\mu s}s+1)}{(T_{\mu s}s+1)} \cdot \frac{(a_0s^{\mu_{co}}+1)}{k_{ss}k_{conv}K} = \frac{ba_0T_{\mu s}s^{1+\mu_{co}} + bT_{\mu s}s + a_0s^{\mu_{co}} + 1}{abT_{\mu s}^{\mu}k_{ss}k_{conv}Ks^{\mu}}. \quad (3.14)$$

Після перетворень отримаємо наступний регулятор

$$W_{sr}^{fo}(s) = K_1s^{\mu_{co}+1-\mu} + K_2s^{1-\mu} + K_3s^{\mu_{co}-\mu} + K_4s^{-\mu}. \quad (3.15)$$

При $\mu_{co} = 0.71$ та $\mu = 1.71$ вираз спрощується:

$$W_{sr}^{fo}(s) = K_1 + K_2s^{-\mu_{co}} + K_3s^{-1} + K_4s^{-1}s^{-\mu_{co}}, \quad (3.16)$$

де коефіцієнти

$$K_1 = \frac{bT_{\mu s}a_0}{abT_{\mu s}^{\mu}k_{conv}k_{ss}K}, K_2 = \frac{bT_{\mu s}}{abT_{\mu s}^{\mu}k_{conv}k_{ss}K}, K_3 = \frac{a_0}{abT_{\mu s}^{\mu}k_{conv}k_{ss}K}, K_4 = \frac{1}{abT_{\mu s}^{\mu}k_{conv}k_{ss}K}.$$

Результати розрахунку коефіцієнтів показані в таблиці 3.8.

Таблиця 3.8. Коефіцієнти $PI^{\mu}I$ -регулятора
в різних діапазонах завдання швидкості

Коефіцієнти	Діапазон завдання U_{ref} , V					
	[0; 1.67)	[1.67; 3.33)	[3.33; 5)	[5; 6.67)	[6.67; 8.33)	[8.33; 10)
K_1	2.95	2.076	1.823	1.686	1.755	1.837
K_2	9.996	19.37	24.28	28.03	30	31.7
K_3	28.63	20.15	17.69	16.36	17.03	17.83
K_4	97.01	189	235.7	272	291.1	307.6

Перехідні процеси з $PI^{\mu}I$ -регулятором представлені на рисунку 3.17.

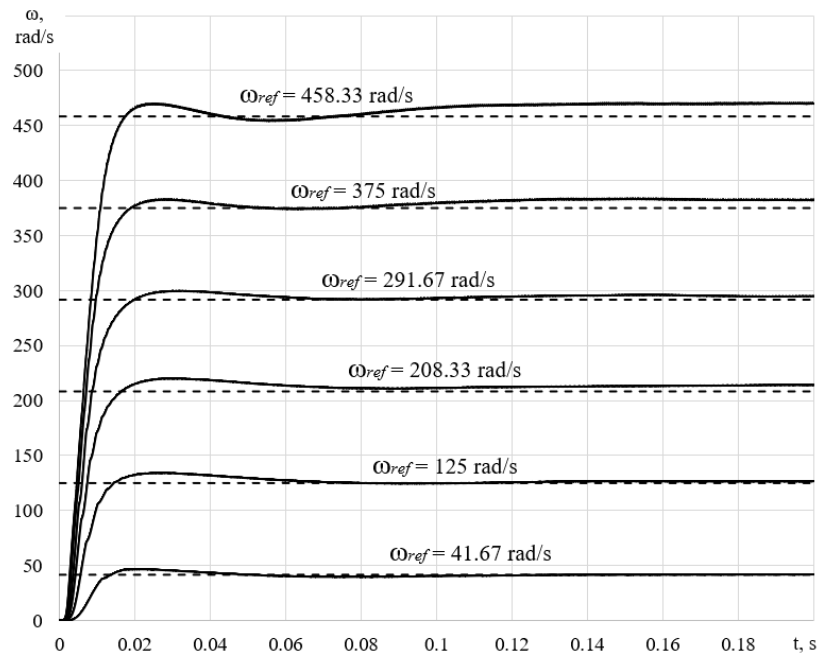


Рис. 3.17. Перехідні процеси з використанням $PI^{\mu}I$ -регулятора при розрахунку коефіцієнтів на дробовий астатизм порядку $1 + \mu$

Як видно з рис. 3.17, якість перехідних процесів точно не відповідає оптимальним. Наприклад, після 0.09 s швидкість двигуна має більше значення, ніж задане на 3 % та повільно зменшується. Час першого узгодження та перерегулювання перехідного процесу представлені в таблиці 3.9. Аналіз

складових ПІДІ-регулятора показав, що причиною суттєво повільного «дотягування» є завищений коефіцієнт при дробовому інтегралі порядку $1 + \mu$.

Таблиця 3.9. Якість перехідного процесу з ПІДІ-регулятором при різних завданнях

Якість перехідного процесу	Завдання швидкості ω_{ref} , rad /s					
	41.67	125	208.3	291.6	375	458.3
Час першого узгодження t_{fm} , s	0.0132	0.0138	0.0156	0.017	0.0166	0.0161
Перерегулювання δ , %	12.26	7.31	4.74	3.32	2.19	2.19

Зменшимо коефіцієнт K_4 для свого діапазону завдання, при цьому скорегуємо значення a і b , щоб компенсувати цей вплив. Значення нових коефіцієнтів представлені в таблиці 3.10.

Таблиця 3.10. Коефіцієнти ПІДІ-регулятора в різних діапазонах завдання швидкості з новими значеннями

Коефіцієнти	Діапазон завдання U_{ref} , V					
	[0; 1.67)	[1.67; 3.33)	[3.33; 5)	[5; 6.67)	[6.67; 8.33)	[8.33; 10)
K_1	2.028	1.631	1.543	1.613	1.679	1.713
K_2	6.873	15.22	20.55	26.81	28.69	29.55
K_3	22.02	14.39	17.69	19.63	27.25	30.31
K_4	49.75	33.57	58.91	81.59	116.4	130.7

При порівнянні старих (табл. 3.10) і нових значень коефіцієнтів можна зробити наступний аналіз: значення K_1 і K_2 не зазнали змін, у K_3 значення збільшилося в області завдання 6.67 V до 10 V, а K_4 приблизно зменшилася в 2 – 3 рази на всій області завдання. Зіставлення перехідних процесів зі старими і новими коефіцієнтами представлені на рис. 3.18.

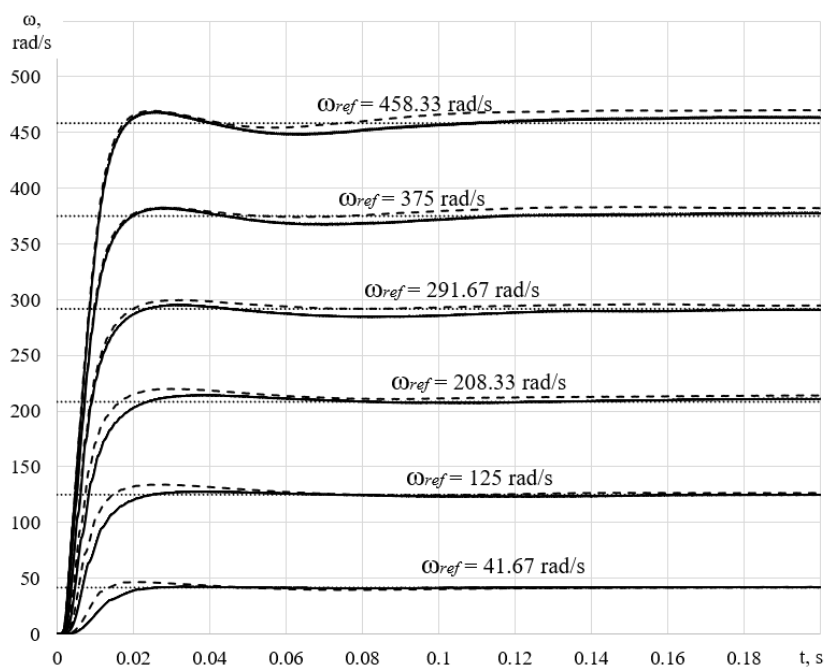


Рис. 3.18. Перехідні процеси з використанням РІПІ-регулятора зі старими (пунктир) і новими значеннями коефіцієнтів (суцільна лінія)

Як видно з рис. 3.18 скориговані значення дозволили зменшити вплив «дотягування» в кінці перехідного процесу, а також встановити перерегулювання на рівні 2% на всьому діапазоні завдання. Але як недолік в області середніх і низьких швидкостей час першого узгодження дещо збільшився. Більш докладні дані в таблиці 3.11.

Таблиця 3.11. Якість перехідного процесу з РІПІ-регулятором при різних завданнях з новими значеннями коефіцієнтів

Якість перехідного процесу	Завдання швидкості ω_{ref} , rad /s					
	41.67	125	208.33	291.61	375	458.33
Час першого узгодження t_{fm} , s	0.0237	0.0221	0.0217	0.0188	0.0173	0.0168
Перерегулювання δ , %	1.967	2.051	1.968	1.756	1.817	1.708

На рисунку 3.19 порівняємо для більшої наочності результати роботи ΓI -регулятора (3.10) і $PI^{\mu}\Gamma I$ -регулятора (3.16) при завданнях швидкості 41.67, 125, 208.33, 291.67, 375, 458.33 rad/s.

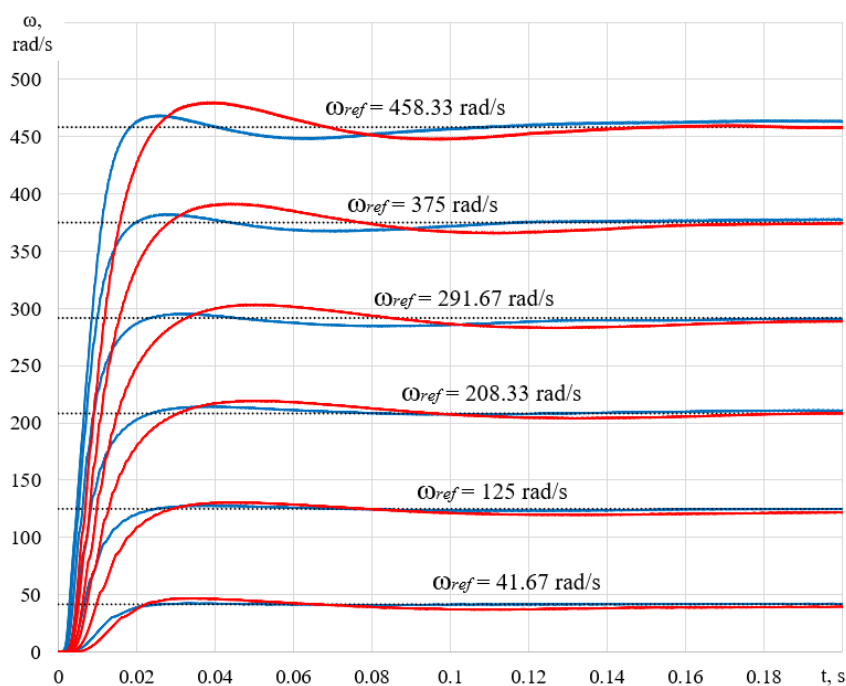


Рис. 3.19. Перехідні процеси з використанням ΓI -регулятора (червоний) і $PI^{\mu}\Gamma I$ -регулятора (синій)

Видно, що форма і якісні показники перехідних процесів системи зберігаються приблизно однаковими при різних завдання швидкості в великому діапазоні регулювання, тобто система поводить себе подібно лінійній – це підтверджує можливість застосування методів дробово-інтегрального числення для такого об'єкта керування, як ВРД. Якщо порівнювати налаштування регулятора на модульний оптимум і дробовий астатизм 1.71, то саме другий варіант дає найкращий час першого узгодження і менше перерегулювання. Для оптимізації перехідного процесу необхідно змінювати параметри регуляторів (3.10) або (3.16) в залежності від завдання швидкості, причому досить враховувати

кілька піддіапазонів, змінюючи коефіцієнти в регуляторах залежності від сигналу завдання (табл. 6 і 10).

Також з метою перевірки проведено моделювання реакції системи на ступеневу зміну сигналу завдання: $0 \rightarrow 150 \rightarrow 300 \rightarrow 450 \text{ rad/s}$ (рис. 3.20).

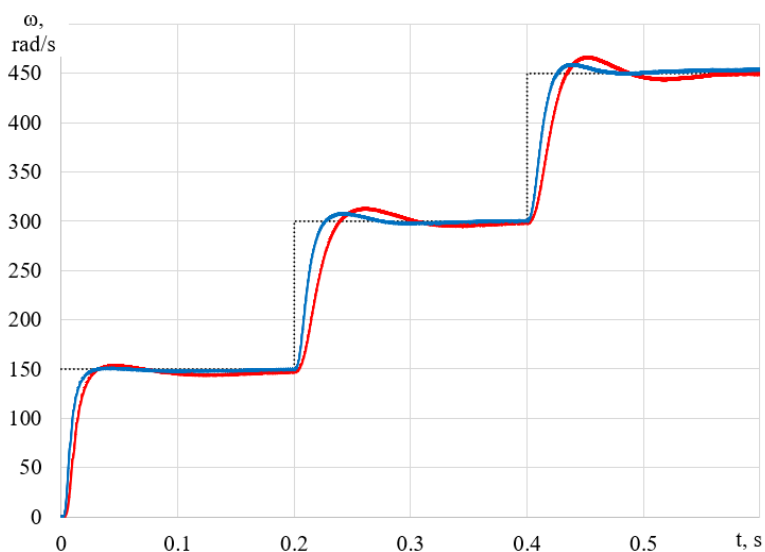


Рис. 3.20. Перехідні процеси при ступінчастій зміні завдання з використанням ПІ-регулятора (червоний) і ПІД-регулятора (синій)

Графіки на рис. 3.20 підтверджують, що якісні показники перехідних процесів зберігаються і відповідають еталонним (рис. 3.19). Кращі результати забезпечуються при налаштуванні контуру з дробовим порядком астатизму [93].

ВИСНОВКИ ПО РОЗДІЛУ 3

1. Проведено моделювання вентильно-реактивного двигуна при різних величинах джерела живлення та за отриманими даними проведена апроксимація ВРД передавальною функцією дробово-аперіодичної ланки

$$W_{co}(s) = \frac{K}{a_0 s^\mu + 1}.$$

2. Показано, що при різних значеннях напруги параметри апроксимуючої функції змінюються, тому коефіцієнти регуляторів необхідно перераховувати в залежності від завдання швидкості.
3. Найкращі показники перехідних процесів (мінімальний час першого узгодження, перерегулювання не перевищує 2%) забезпечують регулятори з дробово-інтегральними складовими при налаштуванні системи з дробовим порядком астатизму $1 + \mu$ (1.71).

РОЗДІЛ 4

РЕГУЛЯТОРИ ДРОБОВОГО ПОРЯДКУ ІНТЕГРУВАННЯ ТА ДИФЕРЕНЦІЮВАННЯ В СИСТЕМІ КЕРУВАННЯ ЕЛЕКТРОМАГНІТНИМИ ГАЛЬМАМИ ДЛЯ НАЛАШТУВАННЯ ДВИГУНІВ ВНУТРІШНЬОГО ЗГОРЯННЯ

4.1 Аналіз об'єкта і попередній синтез системи керування

Індуктивні гальма (рис. 4.1) знаходять широке застосування в сфері вантажного і пасажирського транспорту, промисловості (вітрові генератори, підйомники), системах аварійного гальмування, а також в випробувальних стендах [94]. У багатьох сучасних вантажних автомобілях і автобусах, крім основної гальмівної системи можна зустріти додаткову електричну машину – ретардер. Принцип роботи ретардери заснований на розсіюванні механічної енергії обертання в тепло за рахунок генерації вихрових струмів, які виникають в масивному металевому провіднику при його русі в магнітному полі, яке створюється котушками. Індуктивне гальмо складається зі статора та пари роторів, жорстко з'єднаних між собою. Статор виготовлений зі спеціального провідного матеріалу і грає роль індуктора. Він являє собою послідовно з'єднані електромагніти, в яких при протіканні електричного струму виникає потужне електромагнітне поле.

Магнітне поле створює момент [95], який прагне зупинити ротор, при цьому нагріваючи його. Спеціальна форма роторів допомагає відводити надлишки тепла за допомогою потоку повітря, який вони створюють. Тому при розміщенні ретардера дуже важливо забезпечити хорошу вентиляцію для відводу гарячого повітря.

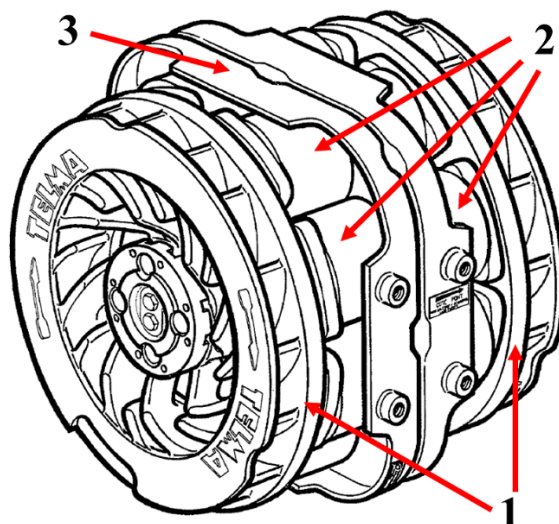


Рис. 4.1. Електромагнітне гальмо, де 1 – металевий диск (ротор), 2 – електромагнітні, 3 – корпус

Завдяки своїм перевагам електромагнітне гальмо може застосовуватися не тільки в системах, де потрібно уповільнити швидкість обертання валу. Так в сервісному центрі з обслуговування спортивних автомобілів різних класів, що беруть участь в міжнародних ралі, використовуються два ретардера. Їх призначення полягає в тому, що на відміну від більшості серійних автомобілів, налаштування двигуна спортивних машин повинно бути виконано з високою точністю для всього діапазону швидкості та обертового моменту двигуна внутрішнього згоряння (ДВЗ) для кожної точки тривимірної поверхні в залежності кута випередження запалювання від швидкості та подачі палива (рис. 4.2). Одночасно необхідно підбирати склад паливної суміші (співвідношення між кількістю палива і повітря) так, щоб забезпечити максимальний обертовий момент та мінімальні витрати палива [96, 97].

Для забезпечення такого налаштування майстри сервісного центру створили стенд, що включає в себе поки дві, а в подальшому чотири електромагнітні гальмівні муфти TELMA, що закріплюються на маточинах ведучих коліс автомобіля. Знизу ретардера вбудовані тензодатчики, що виконують вимір

обертового моменту до 10 разів на секунду. Також було використано інфрачервоні датчики швидкості муфт з роздільною здатністю 10 імпульсів на оберт. Система управління здійснювала синхронізацію швидкостей коліс за рахунок регулювання струму муфт. Однак, випробування показали неефективність системи. Синхронізація відбувалася довго (десятки секунд), незначні зміни моменту і швидкості ДВЗ приводили до виникнення коливань.

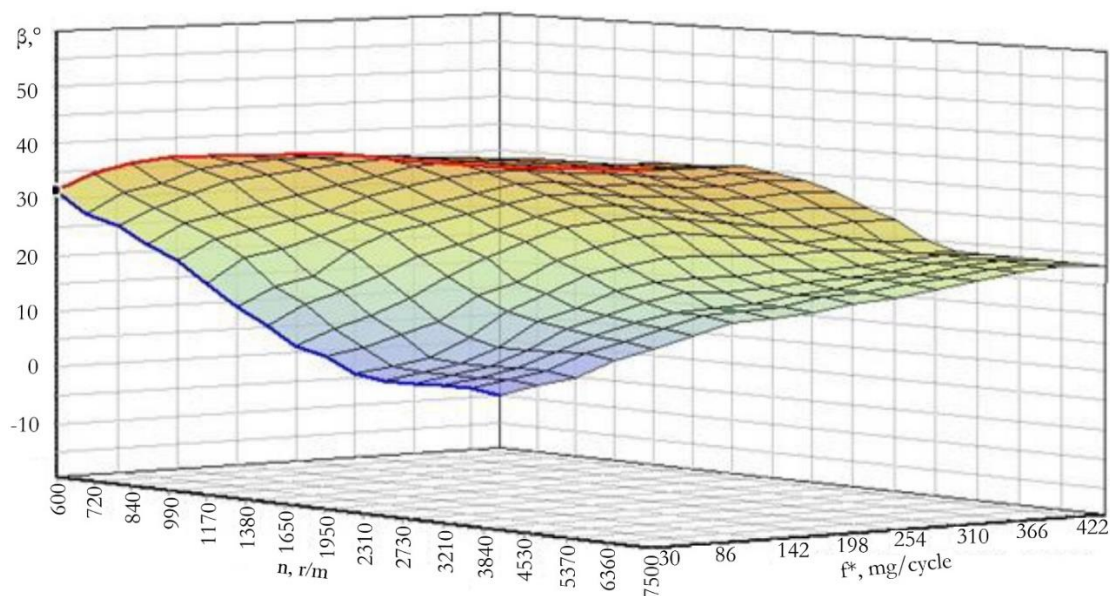


Рис. 4.2. Залежність кута випередження запалювання від швидкості та подачі палива за цикл

У зв'язку з цим постало завдання дослідження системи і синтезу регуляторів, які забезпечать швидко (не довше ніж 1–2 с) синхронізацію частоти обертання коліс та швидкі переходи з однієї робочої точки в іншу.

На рис. 4.3 зображено функціональну схему стенду вимірювання потужності та обертового моменту двигуна внутрішнього згорання, де представлено P_e , T_e , ω_e – потужність, обертовий момент та кутова швидкість ДВЗ, відповідно; T_{EL} , T_{ER} , ω_L , ω_R – момент, кутова швидкість лівого та правого колеса (напіввісей диференціала), відповідно; T_{bL} , T_{bR} – гальмівні моменти від лівого та правого ретардери, відповідно;

U_{PWM_L} , U_{PWM_R} – щільність напруги ШІМ лівого та правого ретардери; ω_{ref} – сигнал завдання швидкості. При збільшенні кута нахилу педалі газу автомобіля до двигуна внутрішнього згоряння вприскується певна кількість палива, при цьому створена механічна енергія передається через трансмісію та диференціал на колеса автомобіля, які починають розвивати швидкість. Для налаштування ДВЗ необхідно стабілізувати цю швидкість заданому рівні при максимальному куту нахилу педалі газу, тобто при максимальній потужності двигуна. Це досягається завдяки гальмівному моменту ретардери, значення якого регулюється системою керування.

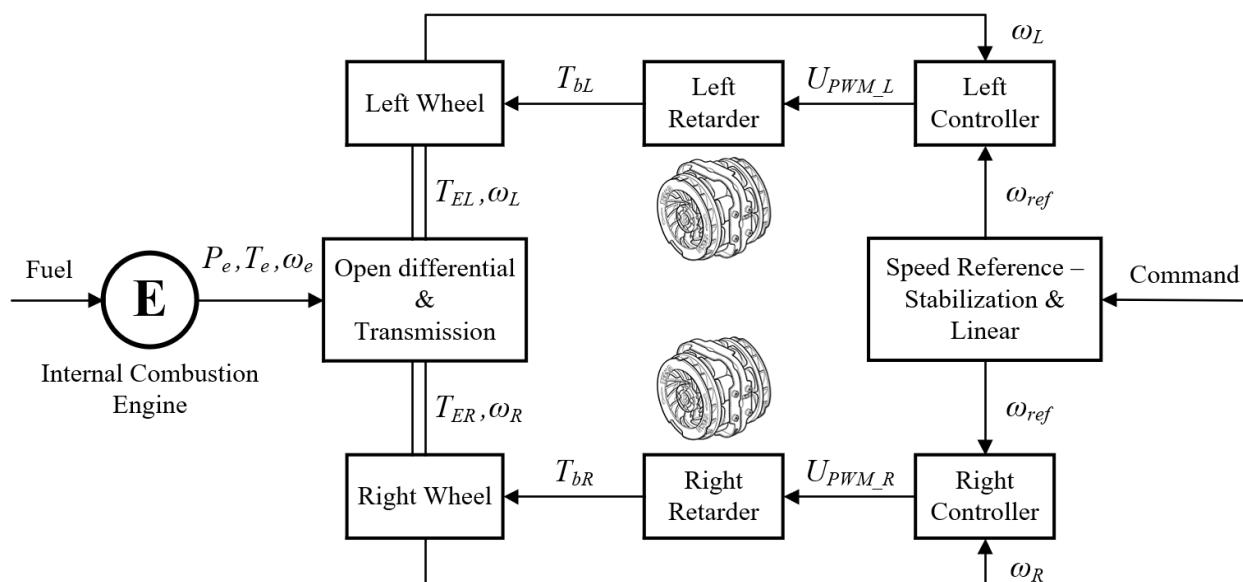


Рис. 4.3. Функціональна схема стенду вимірювання потужності та обертового моменту двигуна внутрішнього згоряння

Для попереднього дослідження промодельовано систему керування стендом. З огляду на те, що головною метою даної роботи не є вивчення внутрішніх процесів в двигуні внутрішнього згоряння, тому його математичне представлення було замінено на асинхронний двигун з пакету SimPower System зі збільшеним опором ротора, яке забезпечить приблизно схожі механічні характеристики.

Також в математичній моделі механічної частини стенду і автомобіля прийняті наступні допущення:

- пружні зв'язку і втрати в зчепленні, коробці передач та карданом валі об'єднано в один елемент;
- втрати потужності і пружні зв'язку в диференціалі не враховано.

З запропонованими припущеннями було побудовано структурну схему (рис. 4.3) і по ній модель в Simulink (рис. 4.4). Вона складається з чотирьох основних блоків: виконавчого механізму (асинхронний двигун), диференціала і двох систем керування ретардерами.

Від валу двигуна потужність передається до симетричного диференціалу, який описується системою рівнянь [98, 99]:

$$\left\{ \begin{array}{l} \frac{d\omega_L}{dt} J_L = T_L - T_{bL} + \mu T_d, \\ \frac{d\omega_R}{dt} J_R = T_R - T_{bR} - \mu T_d, \\ \frac{T_d}{2} = T_L = T_R, \\ \frac{dT_{ru}}{dt} c_u = \omega_h - \omega_L - \omega_R, \\ T_{rv} = b_v (\omega_h - \omega_L - \omega_R), \\ T_d = T_{ru} + T_{rv}. \end{array} \right. \quad (4.1)$$

де $J_L, J_R, \omega_L, \omega_R, T_L, T_R, T_{bL}, T_{bR}$ – моменти інерції, кутові швидкості, обертовий момент і моменти опору лівого і правого коліс, відповідно; μ – коефіцієнт асиметрії навантаження; T_{rv} – момент в'язкого тертя карданного валу; T_{ru} – момент пружності від двигуна до диференціалу; T_d – момент, який передається диференціалом на напіввісі; c_u, b_v – коефіцієнти пружних зв'язків і в'язкого тертя відповідно.

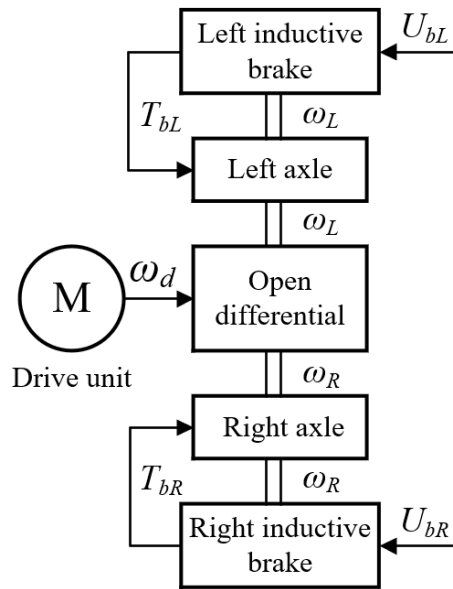


Рис. 4.3. Структурна схема моделі

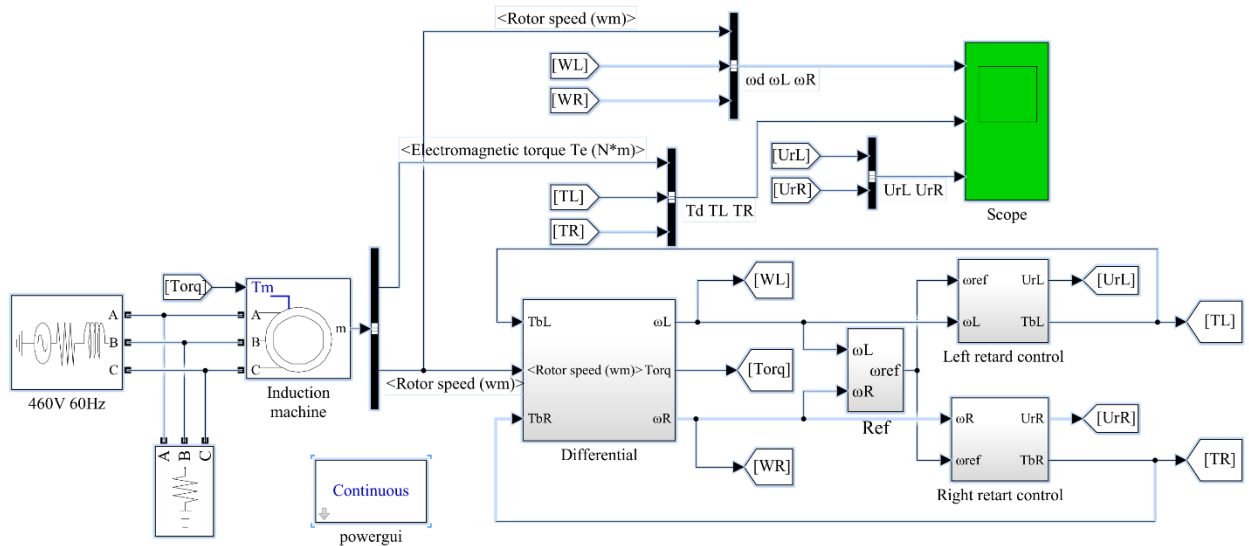


Рис. 4.4. Модель системи керування ретардером в Simulink

Так як напіввісі ведучих коліс ніяк не пов'язані між собою, то вони можуть обертатися з різною швидкістю. Ця асиметрія задана в моделі постійною величиною $\pm \mu T_d$ в системі рівнянь (4.1), хоча при реальній роботі диференціала ця величина і знак можуть бути різними.

За системою рівнянь (4.1) було побудовано блок "Differential" (рис. 4.5), який описує передачу моменту на вали коліс. На його вхід подається кутова швидкість двигуна і моменти опору від індукційного гальма "TbL" і "TbR". Виходами є кутові швидкості лівого та правого коліс " ω_L " та " ω_R ", а також момент диференціала "Torq".

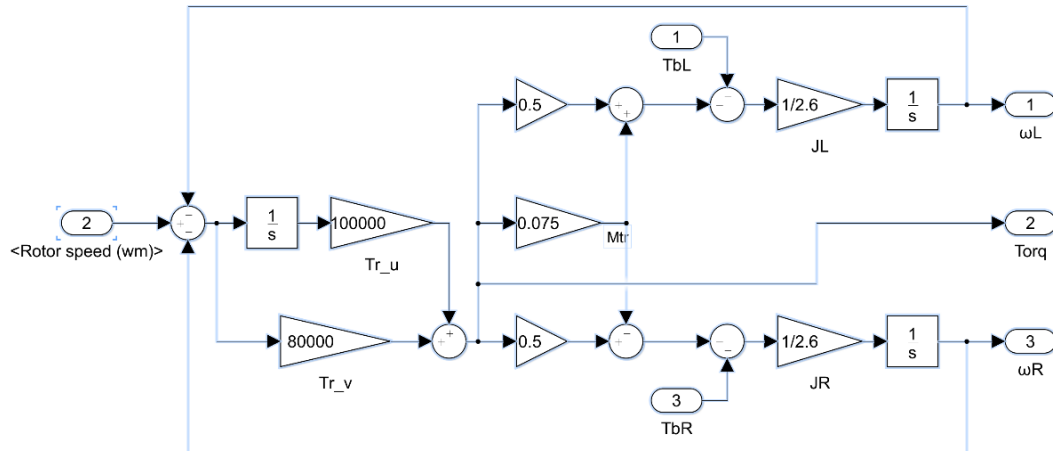


Рис. 4.5. Модель диференціала

Величинами гальмівних моментів керує система, яка представлена блоками "Left retard control" і "Right retard control" (рис. 4.6, а і 4.6, b). На вхід цього блоку подаються поточні значення швидкості коліс (" ω_L " або " ω_R ") та сигнал завдання " ω_{ref} ". Виходами є сигнали "UrL" або "UrR", які визначають величини моментів гальмування ("TbL" або "TbR").

За допомогою даної моделі зпівставлено різні системи стабілізації швидкості. Для керування лівої напіввісі застосовується одноконтурна система з РІ-регулятором швидкості, а для правої – двоконтурна з РІ-регулятором швидкості та релейним регулятором струму. У вимірах датчиків швидкості враховано квантування за часом та за рівнем блоками «Quantizer» і «Zero-Order hold».

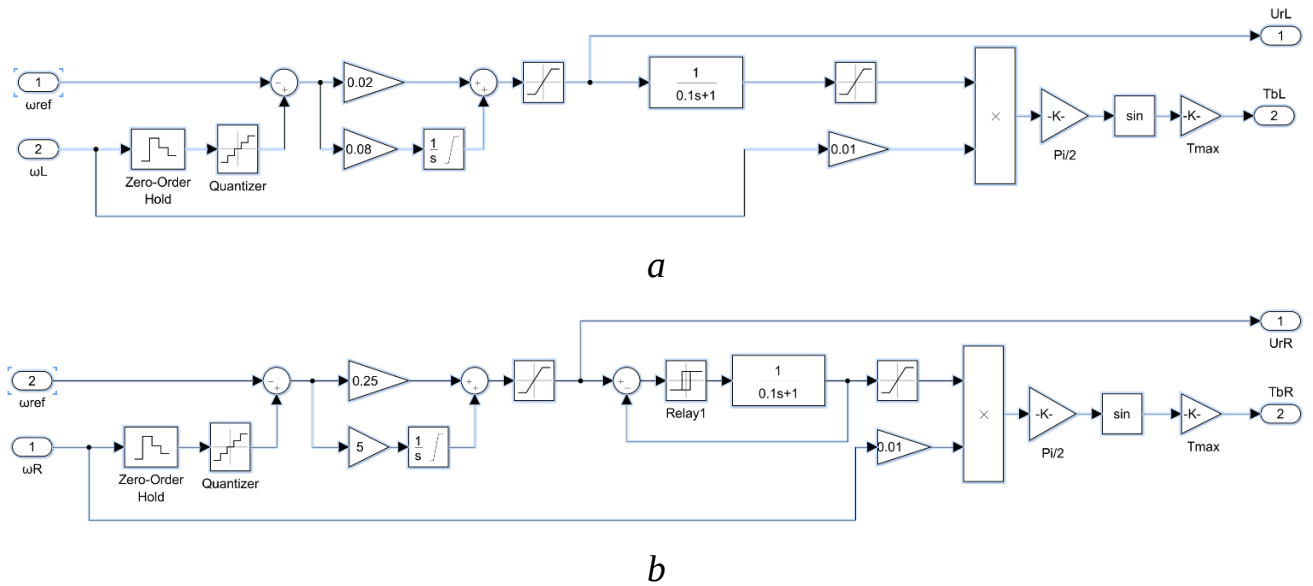


Рис. 4.6. Система керування лівим (а) і правим (b) індукційним гальмом

Перехідні процеси струму в котушці індукційного гальма апроксимовані аперіодичною ланкою:

$$W_c(s) = \frac{1}{0.1s + 1}. \quad (4.2)$$

Нелінійну залежність моменту індукційного гальма від струму та швидкості наближено описано функцією:

$$T_b = T_{\max} \sin\left(\frac{\pi}{2} i_b k \omega_b\right). \quad (4.3)$$

де $T_{\max}$ — максимальний момент гальма; i_b — миттєве значення струму гальма; ω_b — кутова швидкість гальма.

Блок "Ref" (рис. 4.7) служить для завдання швидкості. Він забезпечує два режими: калібрування та стабілізації.

В режимі калібрування двигун розганяється оператором, але при цьому швидкість обертання на двох колесах повинна бути приблизно однаковою. Для цього генерується сигнал завдання:

$$\omega_{ref} = \frac{\omega_L + \omega_R}{2} + \Delta. \quad (4.4)$$

де ω_{ref} – сигнал завдання швидкості; Δ – постійна випередження, підбирається в залежності від прискорення коліс так, щоб до закінчення розгону швидкості були приблизно однакові, а моменти – близькі до нуля.

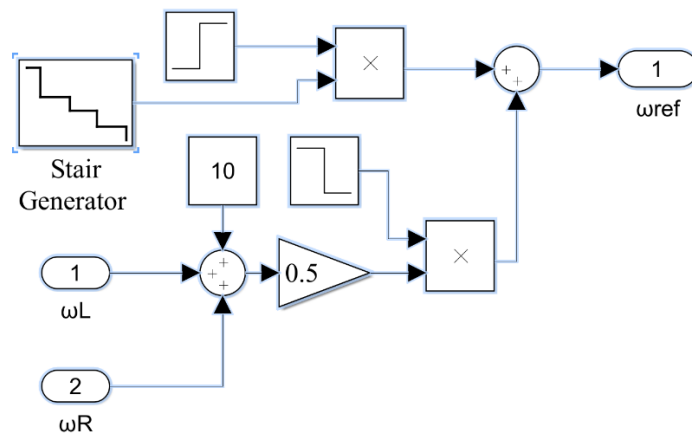


Рис. 4.7. Сигнал завдання швидкості

У режимі стабілізації швидкості сигнал завдання подається від блоку "Stair Generator" і далі на відповідні системи керування. Кутові швидкості коліс повинні підтримуватися постійними незалежно від кута нахилу дросельної заслінки карбюратора, тобто від потужності двигуна.

Результати моделювання представлено на рис. 4.8. На ньому показані перехідні процеси швидкостей для лівої напіввісі – " ω_L ", правої напіввісі – " ω_R " та двигуна – " ω_d ", момент лівої напіввісі – " T_L ", правої напіввісі – " T_R " і двигуна

– " T_d ", а також сигнал завдання струму " U_{rR} " для правого ретардера і значення щільності напруги ШІМ " U_{rL} " для лівого ретардера.

Як видно з рис. 4.8, протягом першої секунди відбувається розгін двигуна в режимі калібрування – в цьому режимі система керування забезпечує приблизно рівні кутові швидкості на обох напіввісях. Так як на ліву напіввісь передається на 15% більший момент, то вона розганяється швидше. При певному значенні швидкості, лівий ретардер починає пригальмовувати напіввісь. Правий ретардер генерує момент, коли кутова швидкість лівої напіввісі стає менше, ніж правої.

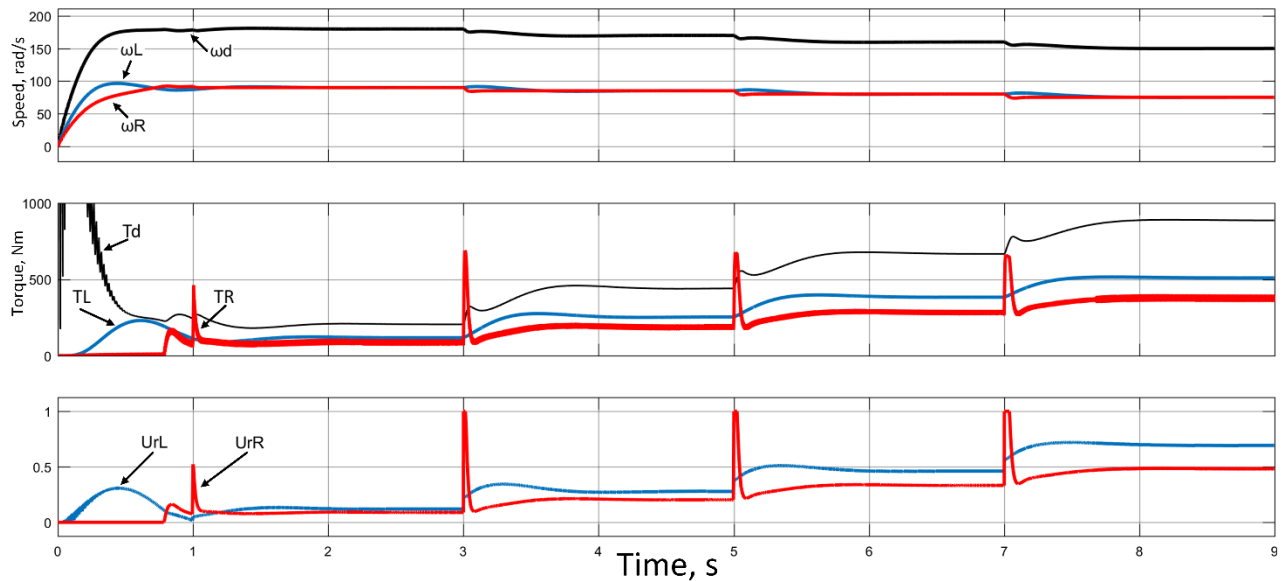


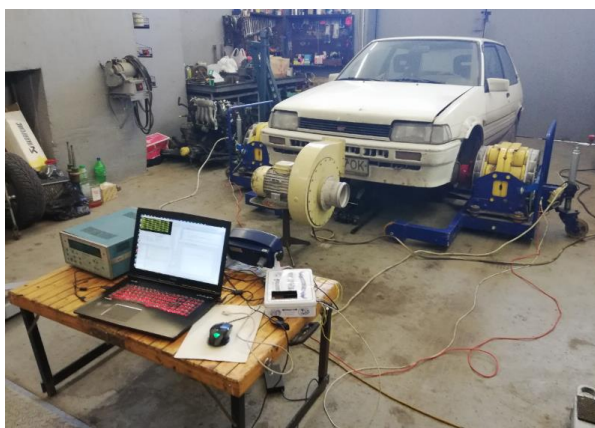
Рис. 4.8. Перехідні процеси за кутовою швидкістю, моменту і сигналами керування

Починаючи з першої секунди система переходить в режим стабілізації з заданими значеннями швидкості 90 rad / s, далі – 85 rad / s при 3 s, 80 rad / s при 5 s і 75 rad / s при 7 s. У всіх випадках двоконтурна система правого гальма забезпечує кращу якість перехідного процесу, ніж одноконтурна лівого гальма. Так, час першого узгодження правого колеса становить 0.04 s, лівого – 0.5 s.

Таким чином, в обох випадках отримано задовільні з точки зору поставленої задачі результати. Однак, остаточне рішення про структуру регуляторів можна прийняти тільки після випробування системи керування.

4.2 Вимірювальний стенд ДВЗ з двома електромагнітними ретардерами

Тести проводилися з автомобілем фірми GT. З фронтального боку знаходився вентилятор для охолодження двигуна внутрішнього згоряння. Ліве і праве колесо машини знімалися і безпосередньо до напіввісі приєднувалися індукційні гальма в спеціальних рамах з роликами (рис. 4.9).



(a)



(b)

Рис. 4.9. Загальний вигляд стенду (a) та індукційного ретардеру (b)

Регулювання струму в електромагнітному гальмі здійснювалося за рахунок зміни щільності ШІМ постійної напруги за одноключовою схемою. В якості силового ключа використовувався IGBT-транзистор серії G4PH40UD. Для захисту від перенапруги обрано зворотний діод серії 30EPH06. Діодний міст KBPC5010 випрямляє напругу від однофазної мережі і подає постійну напругу на транзистори. Керуючі сигнали подаються на базу від гальванічної розв'язки – оптопари HCPL3120. Живлення оптопари забезпечує DC/DC перетворювач

TMA1515D, який підвищував напругу із зовнішнього джерела живлення 5 V до 15 V. На вхід оптопари надходить сигнал від порту мікроконтролера PC6 і при логічній «1» (для МК STM32F429ZI дорівнює 3.3 V) подається 15 V, тим самим відкриваючи IGBT-транзистор. Точна схема для одного ретардера представлена на рисунку 4.10. Для іншого гальма схема аналогічна, але на вхід оптопари сигнал йде від порту PC7.

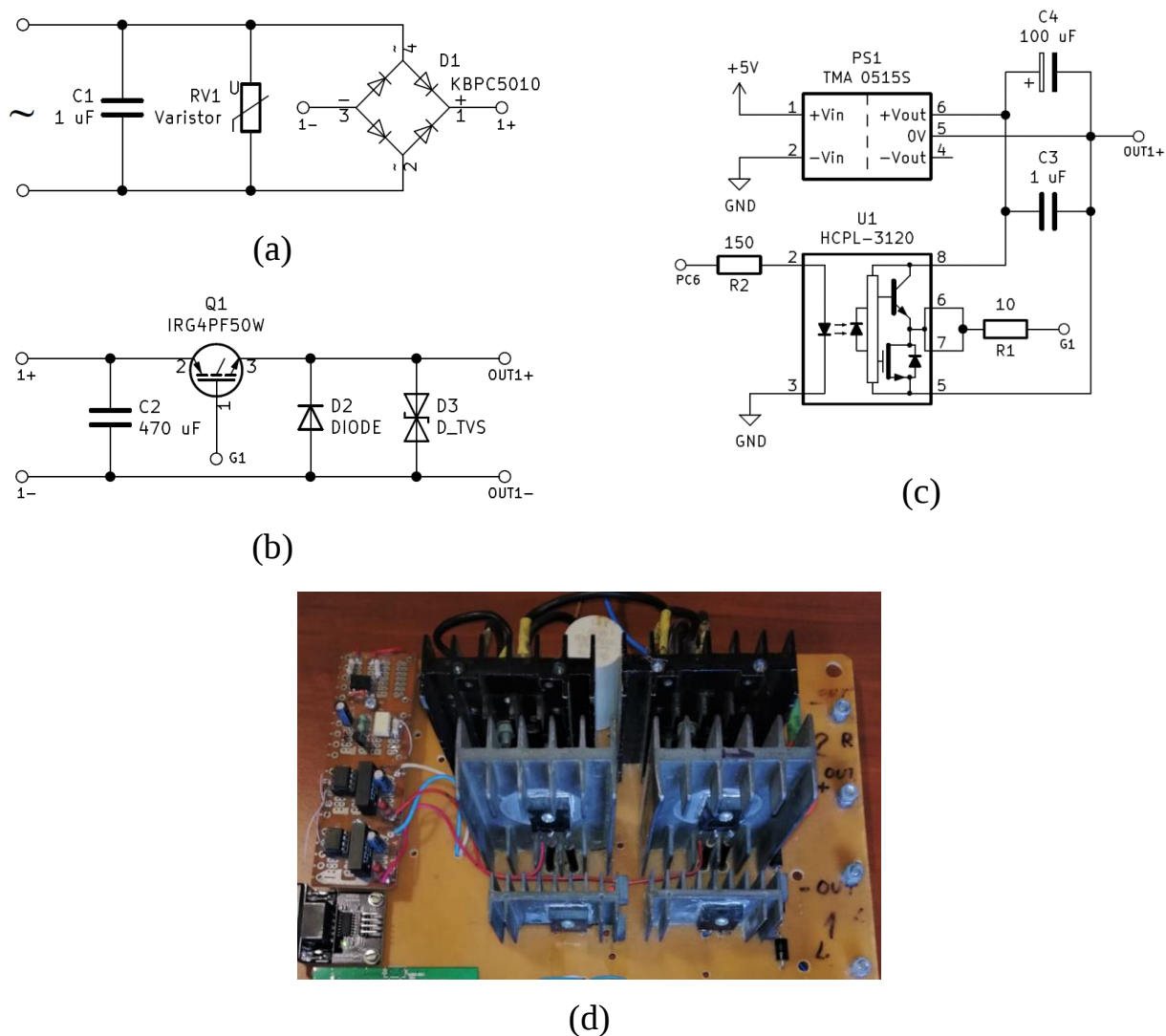


Рис. 4.10. Схеми випрямляча напруги (a), широтно-імпульсного перетворювача (b), оптопари (c) та розташування електроніки на монтажній платі (d)

Керування ключами відбувалося за допомогою плати 32F429IDISCOVERY (рис. 4.11) з вбудованим мікроконтролером STM32F429ZI. Програма для МК розроблялася на мові С за допомогою бібліотеки STM HAL (Hardware Abstraction Layer) в інтегрованому середовищі розробки «System Workbench for STM32».

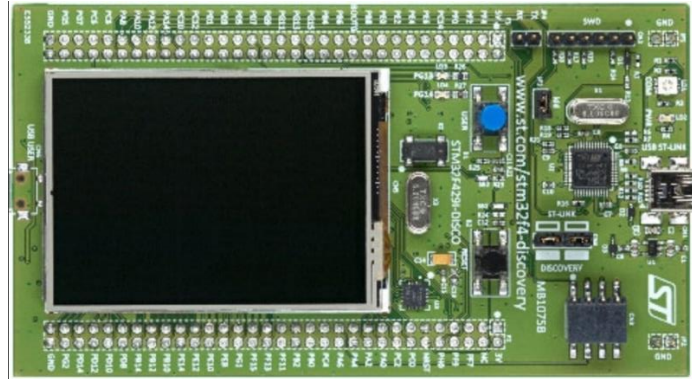


Рис. 4.11. Плата 32F429IDISCOVERY

Тактова частота мікроконтролера – 168 MHz. Частота широтно-імпульсної модуляції на порту PC6 і PC7:

$$f_{PWM} = \frac{f_{APB2}}{(ARR + 1)(PSC + 1)} = 4kHz \quad (4.5)$$

де $ARR = 41999$ (AutoReload Register) – значення регістру лічильника періоду, $PSC = 0$ (Prescaler) – значення регістру преддільника таймера, $f_{APB2} = 168$ MHz – частота APB (Advanced Peripheral Bus).

Як видно з налаштувань таймера, щільність може змінюватися в межах цілих чисел від 0 до 41999.

Обрано наступний час періоду квантування для контуру швидкості:

$$T_{TIM7} = \frac{(ARR + 1)(PSC + 1)}{f_{APB1}} = 10\ ms, \quad (4.6)$$

де $ARR = 8399$, $PSC = 99$, $f_{APB1} = 84 \text{ MHz}$.

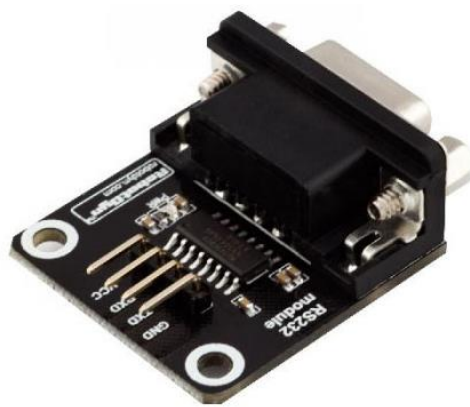
Також налаштовано периферію для прийому даних за протоколом UART з параметрами:

- швидкість передачі 9600 bits / s;
- довжина повідомлення 8 bit;
- без парності;
- один стоп біт.

Таке з'єднання необхідно для отримання даних про величину моменту електромагнітного гальма за допомогою електронних ваг ХК3118Т1 (рис. 4.12а), тензодатчик яких встановлено в корпусі ретардера.



(a)



(b)

Рис. 4.12. Індикатор ваги ХК3118Т1

Індикатор ХК3118Т1 передає інформацію через протокол RS-232, тому необхідно використовувати перетворювач інтерфейсів RS-232 – UART (рис. 4.12b). Модуль побудований на мікросхемі MAX3232.

Для вимірювання швидкості обертання напіввісєй обрано енкодер фірми OMRON E6B2-CWZ6C з 2500 імпульсів на оберт (рис. 4.13а). Внутрішня схема енкодера показана на рис. 4.13b. Блок живлення для датчика – WX-DC2412 24V /

4А. Для узгодження логічних рівнів з МК використано дільник напруги (рис. 4.13с).

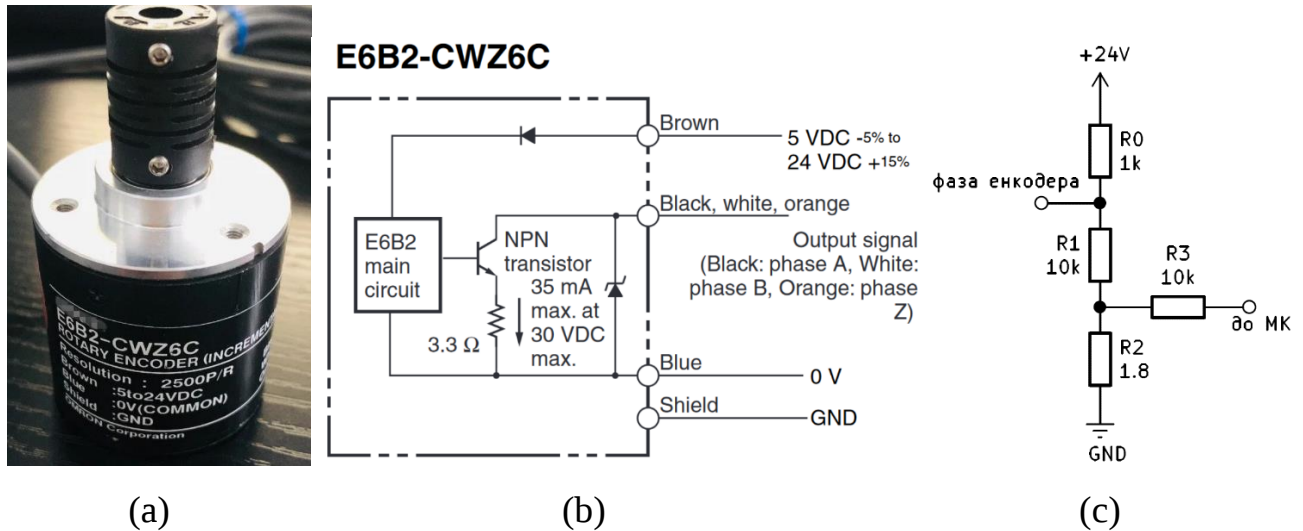


Рис. 4.13. Енкодер E6B2-CWZ6C (а), його функціональна схема (b) і дільник напруги для сполучення логічних рівнів (с)

У мікроконтролері налаштовано два таймера для роботи в режимі квадратурного декодера з визначенням переднього і заднього фронту імпульсу від енкодера, що дозволяє отримувати 10000 імпульсів на оберт, підвищуючи в 4 рази точність вимірювання швидкості.

Для перших тестів в якості інтерфейсу керування було розроблено програма для взаємодії з 2.4 "QVGA TFT LCD дисплеєм через RGB протокол. Драйвером в платі для цього дисплея є мікросхема ILI9341. Також на екрані розміщено резистивне сенсорне скло, а обробкою сигналів від нього займається контролер STMPE811 за протоколом I²C виконується прийом даних на МК STM32F429ZI. Інтерфейс показаний на рис. 4.14. На ньому зображені наступні поля:

- "REF. SPEED" – величина завдання швидкості на ліву і праву напіввісь (в імпульсах за 10 ms);

- "SPEED LEFT" і "SPEED RIGHT" – поточна швидкість лівої та правої напіввісі, відповідно (в імпульсах за 10 ms);
- "GAMMA LEFT" і "GAMMA RIGHT" – поточна щільність лівого і правого індукційного гальма, відповідно (максимальне значення 42000);
- "TORQ. LEFT" і "TORQ. RIGHT" – поточний обертовий момент лівої і правої напіввісі, відповідно (в kg / m);

Кнопки "CALIBR" і "STABLE" включають режим калібрування та режим стабілізації.



Рис. 4.14. Зображення інтерфейсу для керування індукційними гальмами

При натисканні на "DATA" дані відправляються через послідовний інтерфейс USB версії 1.1 Full-Speed зі швидкістю передачі 12 Mbit / s. Через термінальну програму "RealTerm: Serial Capture Program 3.0.1.44" виконувалося відкриття порту для читання та запису отриманих даних в текстовий файл формату ".csv". Передача даних на МК проводилася кожні 40 ms.

Такий інтерфейс має кілька обмежених функціонал (екран невеликого розміру, графіки будуються вручну на комп'ютері, дані у вигляді цифр мають малу наочність, потреба в двох операторів – один за кермом, а другий дає команди і т. д.). Тому розроблена для персонального комп'ютера програма на мові

програмування Java для зв'язку з МК по мережі Ethernet. Інтерфейс створеної програми показаний на рис. 4.15.

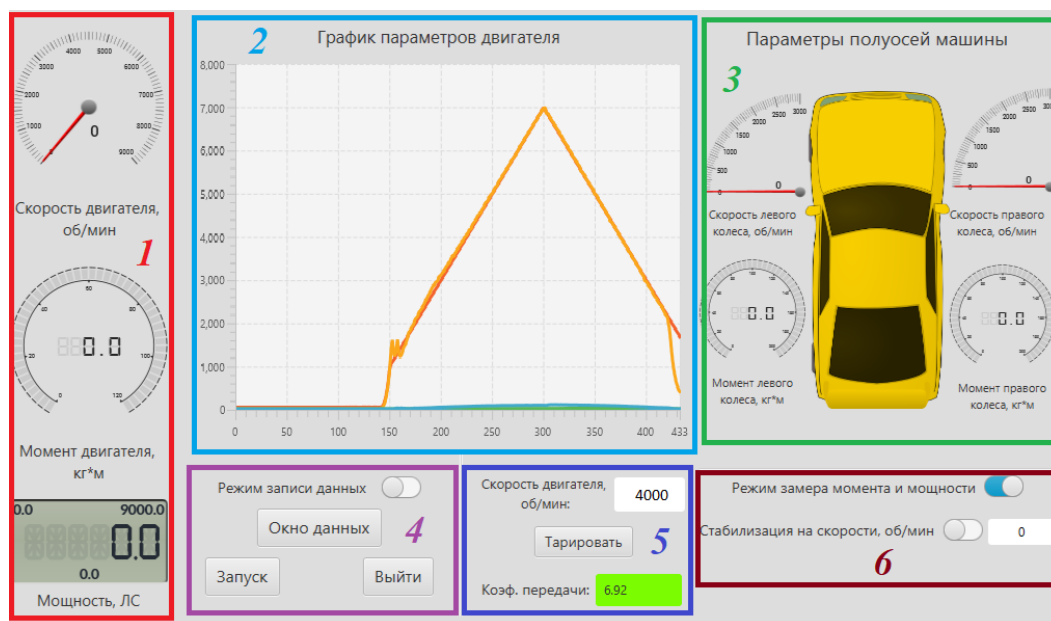


Рис. 4.15. Інтерфейс програми для зняття даних з двигуна внутрішнього згоряння

У червоному прямокутному блоці під номером "1" розміщені: спідометр для вимірної швидкості двигуна, спідометр, що відображає момент двигуна без врахування втрат та дисплей з даними потужності двигуна.

У блакитному прямокутному блоці під номером "2" відображаються осцилограми чотирьох параметрів: завдання швидкості, швидкості двигуна, момент двигуна та потужність двигуна.

У зеленому прямокутному блоці під номером "3" відображається інформація в спідометрах про двох, лівої і правої, напіввісях машини – їх швидкість в оборотах на хвилину та момент в кілограмах на метр.

В фіолетовому прямокутному блоці під номером "4" знаходиться один перемикач та три кнопки. Перемикач "Режим записи данных" робить активним запам'ятовування показників датчиків стенду в масив.

В синьому прямокутному блоці під номером "5" виконуються дії, пов'язані з таруванням. Під цим терміном розуміється знаходження передавального числа між двигуном і напіввісю, де кріпиться колесо. Також для зручності було зроблено так, щоб цей коефіцієнт можна було ставити оператором в цьому ж полі "Коеф. передачі" при натисканні клавіші ENTER.

У коричневому прямокутному блоці під номером "6" розміщені два перемикача, які відповідають за два режими системи керування. Перед роботою в даному блоці необхідно виконати тарування (блок 5) для визначення передаточного числа трансмісії.

Кнопка "Запуск" запускає з'єднання з мікроконтролером, а кнопка "Вийти" зупиняє з'єднання та закриває програму.

Кнопка "Окно данных" викликає інше вікно для відображення статичних характеристик і даних у вигляді таблиці для аналізу показників двигуна внутрішнього згоряння (рис. 4.16).

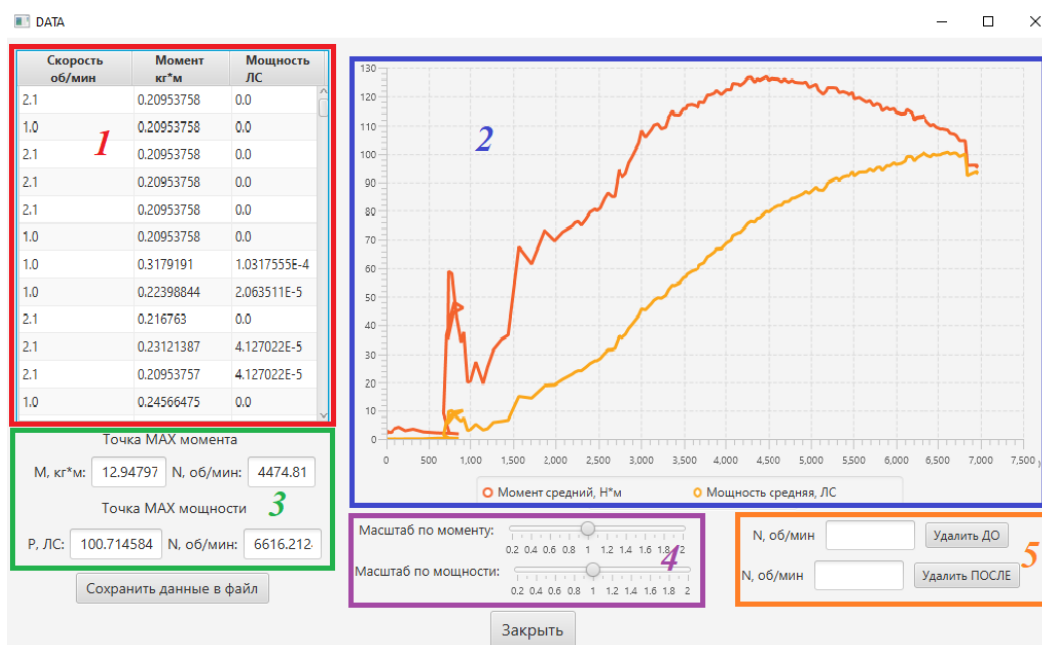


Рис. 4.16. Графічне вікно для відображення отриманих статичних даних потужності та моменту

У червоному прямокутному блоці під номером "1" розташовано таблицю з наступними колонками: кутова швидкість двигуна автомобіля (rpm), моменту ($\text{kg} \cdot \text{m}$) та потужність (HP).

У синьому прямокутному блоці під номером "2" знаходиться графік по попередній таблиці, де потужність та момент побудовані в залежності від швидкості.

У зеленому прямокутному блоці під номером "3" відображено поля для двох точок – максимального моменту та потужності на заміряному діапазоні та при якій швидкості вони отримані.

У зеленому прямокутному блоці під номером "4" розташовано повзунки для зміни масштабу для графіку моменту або потужності, якщо вони необхідні для коректного відображення.

У жовтому прямокутному блоці під номером "5" перебувають поля та кнопки для видалення зайвих даних до або після встановленого значення кутової швидкості двигуна.

З'єднання для обміну інформацією між мікроконтролером і персональним комп'ютером виконано за допомогою мережі Ethernet з кабелем UTP (вита пара) і роз'ємом 8P8C. Сам процес передачі здійснюється через стек протоколів TCP/IP. Для цього роль TCP сервера виступає програма на ПК, а TCP клієнтом є підпрограма в МК.

Пакети даних передаються на сокет сервера у вигляді однобайтового масиву з розміром в 14 елементів (по два байта на кожен величину), де вони перетворюються в 16-ти бітну змінну, а потім розшифровуються до фізичної величини та відображаються в графічній оболонці. Цей пакет даних містить інформацію про сигнали завдання швидкості в МК, правої та лівої кутової швидкості напіввісі машини, правий і лівий обертовий момент ретардери. Останні 4 байта зарезервовані для даних струму в індукційних гальмах.

Відправлення пакета даних від сервера на сокет клієнта складається з одnobайтового масиву розмірністю в 6 елементів. Перший елемент дає інформацію про режим роботи системи керування – лінійне завдання, стабілізація на заданій швидкості та нейтральний режим. Ці 3 режими кодуються через ASCII символами 'r', 's' і 'i'. Наступні 4 байта становлять 32-бітну змінну передавального числа. Останні два байта відповідають за значення швидкості, на якій стабілізувати напіввісі диференціала автомобіля.

4.3 Експериментальні дослідження PID-регулятора за середнім значенням швидкостей напіввісей диференціала автомобіля

У перших експериментальних тестах використовували PID-регулятор за середнім значенням для керування лівим і правим індукційним гальмом, структурну схему якого зображено на рис. 4.18. На ній зображені ω_{left} , ω_{right} , ω_{ref} – значення швидкостей лівої і правої напіввісі, а також сигнал завдання; K_p , K_i , K_d – коефіцієнти PID-регулятора; K_m – пропорційний коефіцієнт, який посилює чутливість сигналу по "допоміжним" гілкам; U_0 – константа для позитивного зсуву загального (основного) сигналу, щоб дати ретардеру базове навантаження; $[-4200; 4200]$, $[0; 42000]$ – мінімальне і максимальне обмеження на відповідні значення; γ_{left} , γ_{right} – вихідні сигнали щільності широтно-імпульсної модуляції на силові транзистори для керування струмом в лівому або правому ретардері.

Таку структуру регулятора було обрано для виключення виникнення коливань швидкостей в протифазі. Основна величина щільності визначається за середньою швидкістю між лівою та правою напіввісю. Це значення обмежувалося нижнім значенням 0 і максимальним 42000. Дві допоміжні гілки створюють сигнали, безпосередньо спрямовані на керування відповідним гальмом. Значення цих сигналів значно обмежено (мінімальне і максимальне значення – -4200 та 4200), і основна його роль полягає в регулюванні біля середнього значення. Після

підсумовування основного сигналу регулятора і відповідного допоміжного, результуюче значення записується в таймер для створення ШІМ з певною щільністю. Коефіцієнти в трьох окремих регуляторах мають однакові величини.

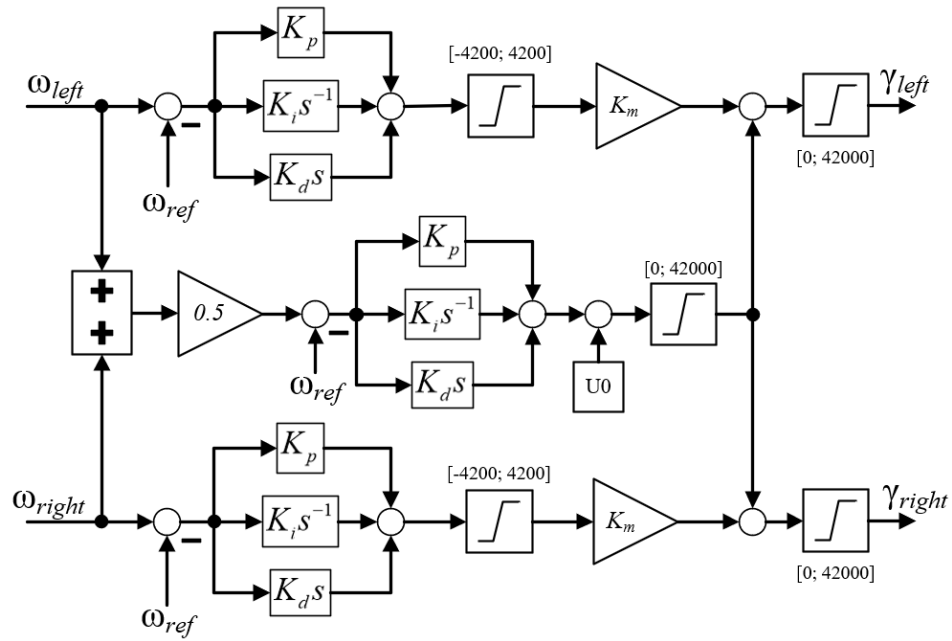


Рис. 4.18. Регулятор за середнім значенням швидкості з PID-складової індукційного гальма

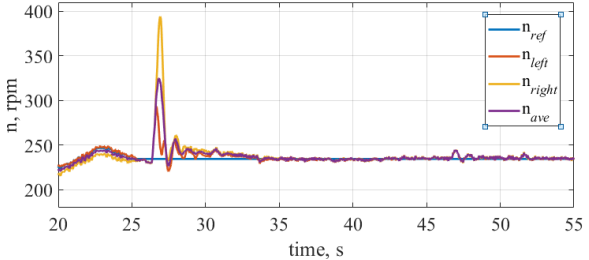
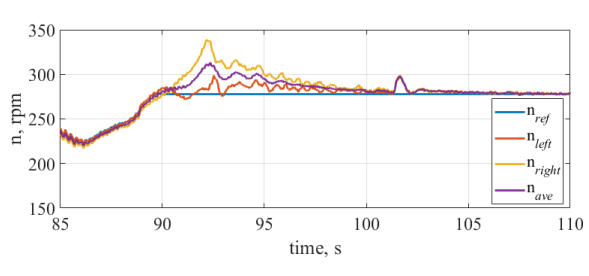
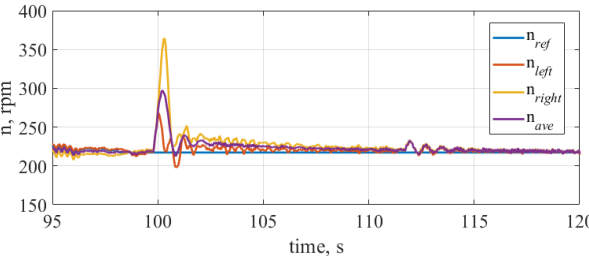
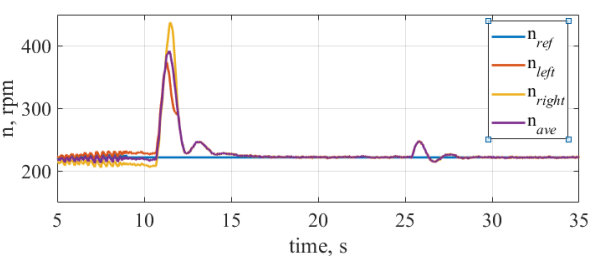
Розрахувати відразу коефіцієнти PID-регулятора важко, так як точно не відомі постійні часу двигуна внутрішнього згорання, його коефіцієнт посилення, індуктивності котушок і моментів інерції електромагнітного гальма. Тому в перших експериментах підбиралися значення, що найбільш підходять для отримання перехідних процесів з кращими показниками.

Через людський фактор – зміна характеру натискання на педаль газу – графіки з різними регуляторами неточно відповідають один одному.

Результати роботи PID-регулятора с різними коефіцієнтами представлено в таблиці 4.1. На рисунках зображені – n_{ref} , n_{left} , n_{right} , n_{ave} – сигнал завдання, частота обертання лівої і правої напіввісі, а також їх середня величина. Постійна випередження 12 грм.

Таблиця 4.1. Результати експериментальних досліджень з регулятором за середнім значенням

№ п/п	Рисунок	Опис
1		$K_p = 25, K_i = 50, K_d = 0.5$ $U_0 = 1000, K_m = 1.5$ До 23.6 с система працює в режимі калібрування. Ближче до 400 rpm спостерігається коливальний процес. Хоча якщо судити по середній частоті обертання, то вона близька до завдання. При переключенні в режим стабілізації і накиду навантаження з'являються коливання трохи більшої частоти, але з меншою амплітудою і з плином часу загасають. Ближче до кінця ПП спостерігаються ті ж коливання.
2		$K_p = 50, K_i = 25, K_d = 0.25$ $U_0 = 1000, K_m = 1.5$ Тривалість перехідного процесу склала 7 секунд, а перерегулювання склало 7.88%. До негативних якостей можна віднести пульсації швидкості в сталому режимі.
3		$K_p = 25, K_i = 12.5, K_d = 0.125$ $U_0 = 1000, K_m = 1.5$ Коливання стало менше, але перехідний процес став більш затягнутий. Час перехідного процесу зайняло 10 секунд, а перерегулювання склало 5.99%.
4		$K_p = 75, K_i = 25, K_d = 0.5$ $U_0 = 500, K_m = 2.5$ Частота обертання лівої напіввісі майже відразу зрівнюється із завданням, але права напіввісь має перерегулювання. В обох випадках притаманне коливання сигналу. Час перехідного процесу по середній швидкості 6 с та величина його перерегулювання 4.71%.

5		$K_p = 75, K_i = 25, K_d = 0.5$ $U_0 = 3000, K_m = 5$ Збільшення коефіцієнта K_m призвело до того, що амплітуда коливань під час перехідного процесу стала більше. Перерегулювання по середній швидкості склало 24.1%.
6		$K_p = 50, K_i = 16.6, K_d = 0.3$ $U_0 = 500, K_m = 5$ Зменшення коефіцієнтів дозволило знизити амплітуду коливань, але перехідний процес став більш затягнутий. Перерегулювання з таким регулятором прийняло значення 12.3%, а час регулювання 8 s.
7		$K_p = 100, K_i = 16.6, K_d = 0.33$ $U_0 = 500, K_m = 5$ Збільшення пропорційного коефіцієнта підвищило перерегулювання, яке склало 36.3%, а час регулювання 7 s.
8		$K_p = 25, K_i = 16.6, K_d = 0.33$ $U_0 = 500, K_m = 5$ Регулятор з подібними коефіцієнтами дає перехідний процес з перерегулюванням в 76.6%, але час регулювання 5 s. Також спостерігається фактична відсутність пульсацій.

Як видно з табл. 4.1 з усіх розглянутих результатів при різних коефіцієнтах регуляторів не вдалося домогтися перехідного процесу з прийнятною якістю за перерегулюванням та за часом регулювання. Так, наприклад, в 1, 4 на початку ПП спостерігаються коливання в протифазі і шуми в сталому режимі; в 5, 7, 8 – перерегулювання від 24% до 77%; в 2, 3, 6 – затягнутий перехідний процес від 7 s до 10 s. Тому було вирішено випробувати двоконтурну систему керування.

4.4 Експериментальні дослідження двоконтурної системи керування

Так як мікроконтролер вже обробляє дані з датчика моменту, то було вирішено створити на його основі внутрішній контур. Основним недоліком такого контуру є велике квантування за часом, яке становить 100 ms.

PI-регулятор швидкості зовнішнього контуру та PID-регулятору моменту внутрішнього контуру зображено на рис. 4.19, де K_p^s, K_i^s – коефіцієнти PI-регулятора швидкості; K_p^t, K_i^t, K_d^t – коефіцієнти PID-регулятора моменту; $T_{ref. left}, T_{ref. right}$ – сигнали завдання лівого та правого моменту; T_{left}, T_{right} – сигнали лівого та правого тензодатчиків; $[-10; 120]$ – мінімальне і максимальне обмеження на момент; $[0; 1]$ – мінімальне і максимальне відносне значення щільності; K_γ – коефіцієнт посилення для переводу з відносного значення щільності в параметр регістру таймера, для створення відповідного сигналу ШІМ.

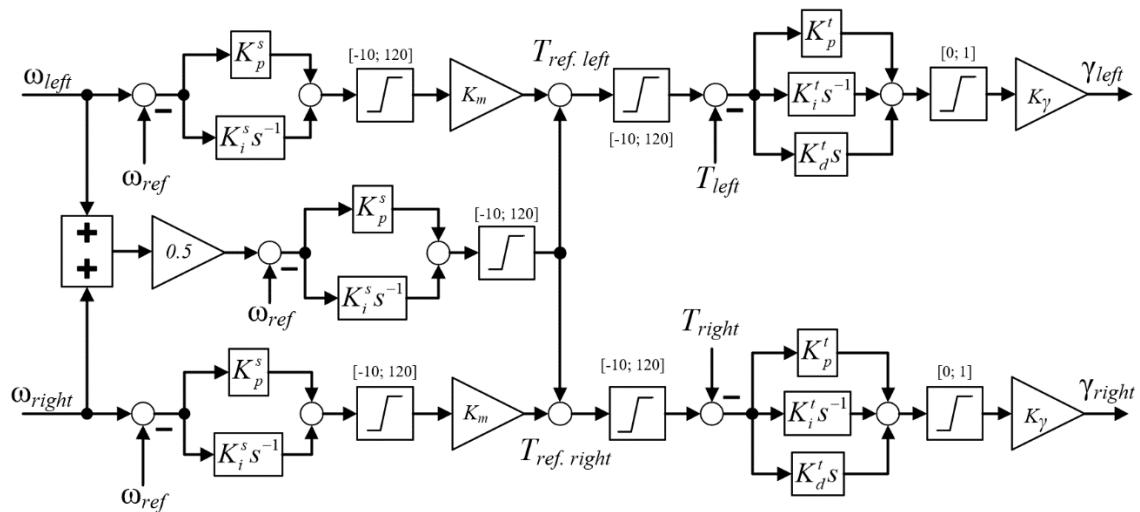


Рис. 4.19. PI-регулятори швидкості і PID-регулятори моменту в двоконтурній системі керування

Загальна структура зовнішнього регулятора швидкості аналогічна попередньому (рис. 4.18) за винятком того, що використовується тільки

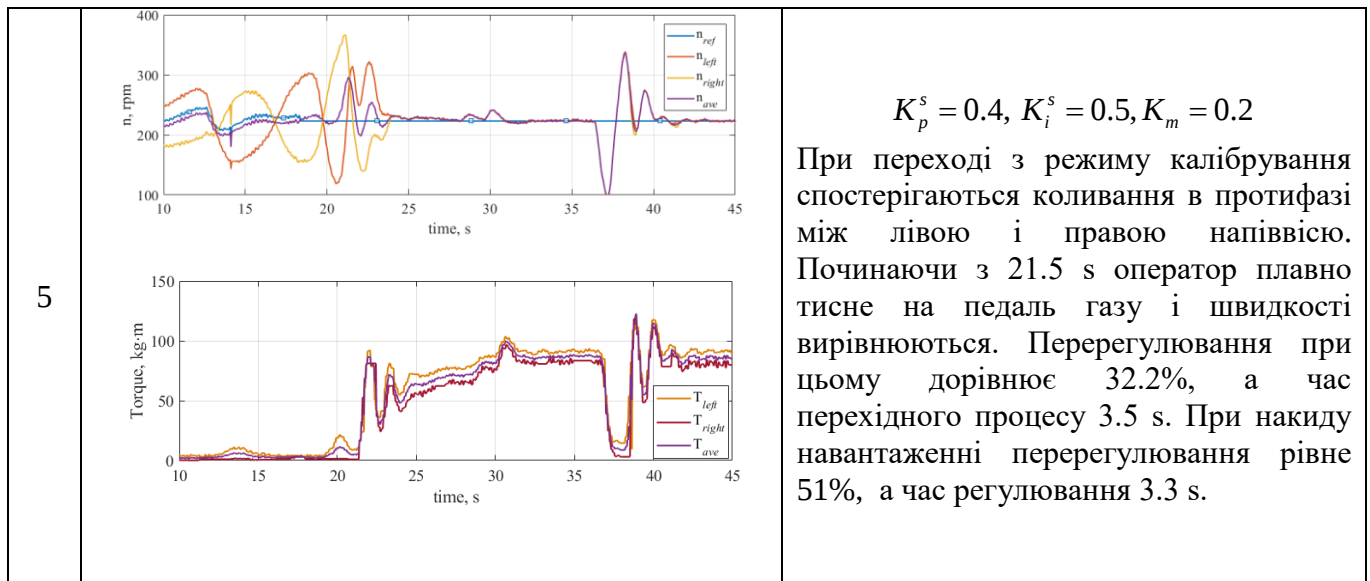
пропорційна і інтегруюча складова для кожної з трьох гілок. Величина обмеження сигналу завдання для внутрішнього контуру на рівні нижньої межі $-10 \text{ kg}\cdot\text{m}$ і верхньої $120 \text{ kg}\cdot\text{m}$. Негативне значення моменту не має фізичного застосування так, як індукційне гальмо не може створювати момент для руху, тому негативне значення тільки необхідно для зменшення інтегральної складової під час перехідного процесу. Величину верхньої межі було визначено експериментальним шляхом з 20% запасом. Структура внутрішнього регулятора простіше, ніж зовнішнього, та керування за моментом лівим і правим ретардером проводиться незалежно за допомогою PID-регулятора. Його коефіцієнти обрано таким чином $K_p^t = 4.42 \cdot 10^{-3}$, $K_i^t = 8.66 \cdot 10^{-3}$, $K_d^t = 17.33 \cdot 10^{-6}$.

Результати роботи двоконтурної системи керування з різними коефіцієнтами PI-регулятора швидкості можна спостерігати в табл. 4.2. На рисунках зображено T_{left} , T_{right} , T_{ave} – це сигнали лівого і правого датчиків моменту, а також їх середнє значення.

Таблиця 4.2. Результати експериментальних досліджень
з двоконтурною системою керування

№ п/п	Рисунок	Опис
1		$K_p^s = 0.1, K_i^s = 0.26, K_m = 0.2$ Починаючи з 18.4 с система керування переходить з режиму калібрування в режим стабілізації. Оператор повільно натискає на педаль газу і при 39 с подає повну потужність на напіввісі. На початку цього проміжку видно, як напіввісь обертається в протифазі, але через 10 секунд швидкість вирівнюються. На 39 с спостерігається перерегулювання 19.9% з подальшим коливальним ПП з повільно згасаючим характером. При зниженні швидкості і потім накиду навантаження оператором (59 s) викликаються коливання з тієї ж частотою, але з значною початковою амплітудою (перерегулювання 65.8%).

2		$K_p^s = 0.4, K_i^s = 0.26, K_m = 0.2$ В 9.2 s система керування переходить в режим стабілізації, 12.4 s – оператор повільно тисне на газ, 25.8 s – різка подача палива, 36.3 s – накид моменту з боку двигуна. При 25.8 s перерегулювання склало 13.5%, а час регулювання 3.4 s. Як видно частота коливань збільшилася приблизно в два рази (в порівнянні з попереднім випадком), що призвело до більш швидкого темпу затухання. В 36.3 s перерегулювання дорівнює 45.2% з часом перехідного процесу 5 s.
3		$K_p^s = 0.2, K_i^s = 0.5, K_m = 0.2$ В цьому експерименті оператор вже фактично відразу тиснув на газ до кінця при переході в режим стабілізації і повторно при накиду навантаження. Як в 1-му випадку на початку діапазону спостерігаються коливання напіввісі в протифазі. Перерегулювання за середньою величиною склало 46.1%, а час перехідного процесу 5 s. При накиду навантаженні перерегулювання дорівнює 71.1%, а час регулювання 4.1 s.
4		$K_p^s = 0.1, K_i^s = 0.5, K_m = 0.2$ При даних коефіцієнтах регулятора швидкості оператор плавно тиснув на педаль газу до 35.2 s. Як і у випадках 1 і 3 спостерігаються коливання в протифазі лівої і правої напіввісі, але з меншою частотою. Перерегулювання на початку складає 31.1%. При досягненні максимальної потужності від ДВЗ викликаються автоколивання.



Як видно з табл. 4.2 двоконтурна система керування в порівнянні з регулятором за середнім значенням швидкості (табл. 4.1) в загальному випадку (1, 2, 3 і 5 з табл. 4.2) також призводить до протифазних коливань, а при деяких параметрах РІ-регулятора з підвищенням потужності від ДВЗ викликаються автоколивання (4 з табл. 4.2). При відповідних коефіцієнтах в регуляторах 3 і 5 час регулювання тримається в діапазоні від 3.3 s до 4.1, але при цьому перерегулювання становить від 32.2% до 71.1%.

4.5 Синтез системи керування електромагнітними ретардерами на базі дробово-інтегральних регуляторів

Наступним кроком була спроба спростити систему керування – відмовитися від внутрішнього контуру і від взаємозв'язаного регулятора за середнім значенням. Керування кожної з двох напіввісей виконується незалежно зі своїм PID-регулятором (рис. 4.32). Подібне спрощення дозволило зменшити число параметрів системи і спростити підбір коефіцієнтів.

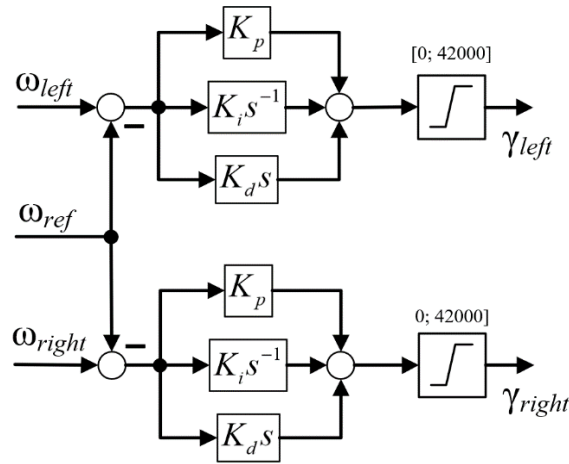


Рис. 4.32. Структура регулятора з незалежним керуванням по лівій і правій напіввісі диференціала

4.5.1 Ідентифікація структурної схеми і параметрів об'єкта керування

Після декількох експериментів один з найбільш вдалих варіантів підбору коефіцієнтів мав наступні параметри – $K_p = 50$, $K_i = 50$, $K_d = 9$. Для більш точного налаштування ідентифіковано модель об'єкта керування за отриманим перехідним процесом. В якості математичного опису перехідного процесу розглянуті наступні передавальні функції об'єкта керування:

$$W_{co1}(s) = \frac{K}{(a_1 s + 1)(a_2 s + 1)(a_3 s + 1)} \quad (4.7)$$

$$W_{co2}(s) = \frac{K}{(a_1 s + 1)(a_2 s + 1)} \quad (4.8)$$

$$W_{co3}(s) = \frac{K}{(a_1 s + 1)(a_2 s + 1)s} \quad (4.9)$$

$$W_{co4}(s) = \frac{K}{a_1 s^{1+\mu_{co}} + a_2 s^{\mu_{co}} + 1} \quad (4.10)$$

Для ідентифікації невідомих коефіцієнтів (4.7), (4.8), (4.9) и (4.10) використано генетичний алгоритм, в якому при селекції обрано турнірний метод, нові особини успадковували гени батьків шляхом рівномірного схрещування, ймовірність мутації в хромосомі – 20%. Оцінкою пристосованості особини служила середньоквадратична помилка F . В якості вихідних даних використані дискретні значення швидкостей (кількість імпульсів енкодера за 0.01 s) тієї частини перехідного процесу, коли стенд переходив в режим стабілізації фіксованій швидкості.

Результати ідентифікації параметрів передавальних функції W_{co1} , W_{co2} , W_{co3} та W_{co4} зведені в табл. 4.3. А графіки експериментальних і розрахункових перехідних процесів зіставлені на рис. 4.33.

Таблиця 4.3. Результати ідентифікації передавальних функцій

Параметр	Передавальна функція об'єкта керування			
	W_{co1}	W_{co2}	W_{co3}	W_{co4}
K	0.03729	0.06666	0.04069	0.11514
a_1	0.7445	1.286	0.2042	2.895
a_2	0.3208	0.8804	0.04813	1.899
a_3	0.7252	—	—	—
μ_{co}	—	—	—	0.6261
F	0.0130	0.0142	0.0275	0.0132

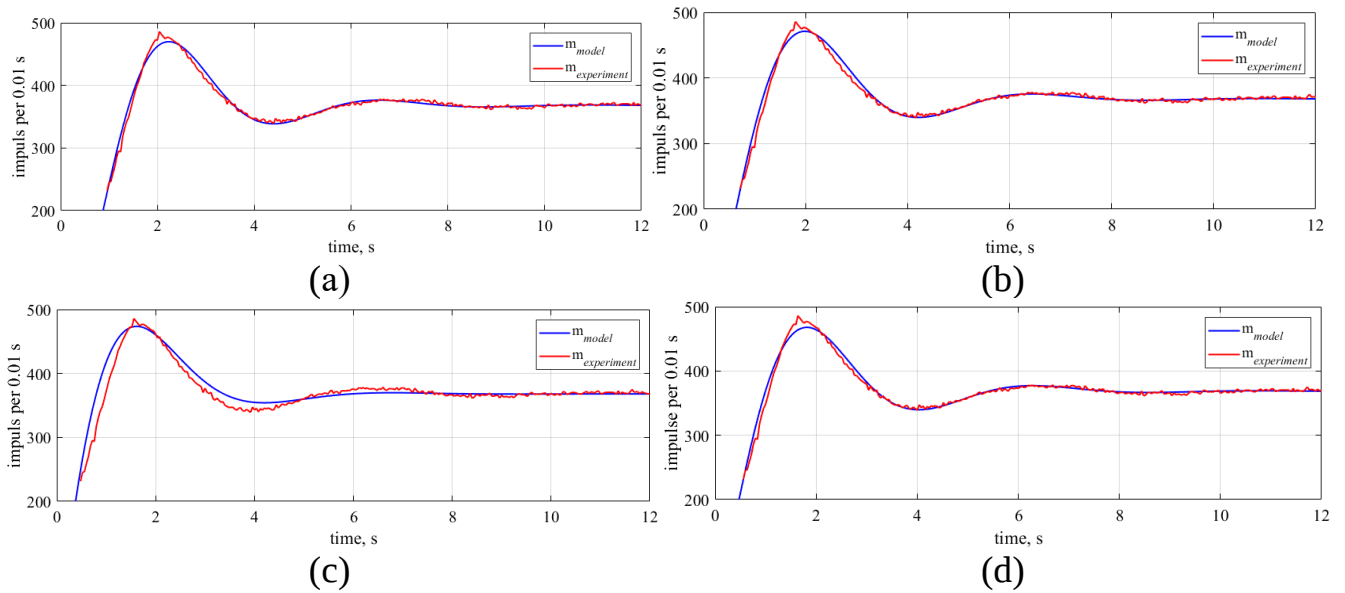


Рис. 4.33. Порівняння перехідного процесу експерименту $m_{\text{experiment}}$ і передавальної функції моделі об'єкта m_{model} , де (a) – W_{co1} , (b) – W_{co2} , (c) – W_{co3} , (d) – W_{co4}

Передавальна функція W_{co1} має найменшу середньоквадратичну помилку серед цілочисельних передавальних функцій, а передавальна W_{co4} – серед функцій дробового порядку.

4.5.2 Синтез регуляторів

На підставі функції W_{co1} синтезовано PID-регулятори для лівого та правого каналів керування. Сам процес синтезу виконувався в Matlab Simulink за допомогою функції "Tune" в блоці "PID Controller". В результаті для PID-регулятора обрано такі коефіцієнти – $K_p = 65$, $K_i = 50$, $K_d = 15$.

Для синтезу регулятора з моделлю дробового порядку налаштуємо розімкнутий контур на модульний оптимум з наступною еквівалентною передавальною функцією:

$$W_{op.s}(s) = \frac{1}{2T_f s(T_f s + 1)} = W_{sr}(s) \frac{1}{(T_f s + 1)} \frac{K}{(a_1 s^{1+\mu_{co}} + a_2 s^{\mu_{co}} + 1)}, \quad (4.11)$$

де $T_f = 0.05$ s – постійна часу фільтра, яка еквівалентна малій постійній часу.

Тоді отримуємо необхідну передавальну функцію регулятора:

$$W_{sr}(s) = \frac{\cancel{(T_f s + 1)}}{2T_f s \cancel{(T_f s + 1)}} \cdot \frac{(a_1 s^{1+\mu_{co}} + a_2 s^{\mu_{co}} + 1)}{K} = \frac{a_1 s^{\mu_{co}}}{2T_f K} + \frac{a_2 s^{\mu_{co}-1}}{2T_f K} + \frac{s^{-1}}{2T_f K}. \quad (4.12)$$

яка відповідає D^{VI}I-регулятору:

$$W_{sr}(s) = K_1 s s^{\mu_{co}-1} + K_2 s^{\mu_{co}-1} + K_3 s^{-1}, \quad (4.13)$$

$$\text{де } K_1 = \frac{a_1}{2T_f K} = 251.4, \quad K_2 = \frac{a_2}{2T_f K} = 164.9, \quad K_3 = \frac{1}{2T_f K} = 86.85.$$

Методи дробово-інтегрального числення дозволяють синтезувати регулятор для налаштування системи на дробовий порядок астатизму $\mu \in (1; 2)$. Беручи до уваги, що об'єкт має апроксимовану передавальну функцію дробового порядку $1 + \mu_{co}$, визначимо передавальну функцію регулятора зі співвідношення:

$$W_{op.s}(s) = \frac{1}{a T_f^{\mu-1} s^{\mu-1}} \frac{b T_f s + 1}{b T_f s} \frac{1}{(T_f s + 1)} = W_{sr}(s) \frac{1}{(T_f s + 1)} \frac{K}{(a_0 s^{1+\mu_{co}} + a_1 s^{\mu_{co}} + 1)} \quad (4.14)$$

де значення a і b наближено вибираються по наступних співвідношеннях:

$$\begin{cases} (ba) \approx \exp(-10.27 + 7.831\mu), \\ b \approx 7.336 + 0.792(ba) + 3.83 \ln(ba). \end{cases}$$

Тоді отримуємо:

$$W_{sr}(s) = \frac{1}{aT_f^{\mu-1}s^{\mu-1}} \frac{bT_f s + 1}{bT_f s} \frac{1}{(T_f s + 1)} \frac{(T_f s + 1) \cdot (a_1 s^{1+\mu_{co}} + a_0 s^{\mu_{co}} + 1)}{K} \quad (4.15)$$

Після перетворень отримаємо передавальну функцію регулятора з п'яти складових:

$$W_{sr}(s) = K_0 s^{-(\mu-1)} \left[K_1 s^2 s^{\mu_{co}-1} + K_2 s s^{\mu_{co}-1} + K_3 s^{\mu_{co}-1} + K_4 s^{-1} + K_5 \right] \quad (4.16)$$

де коефіцієнти розраховуються за виразами:

$$K_0 = \frac{1}{aT_f^{\mu-1}}, \quad K_1 = \frac{a_1}{k_f K}, \quad K_2 = \frac{a_2 b T_f + a_1}{b T_f K}, \quad K_3 = \frac{a_2}{b T_f K}, \quad K_4 = \frac{1}{b T_f K}, \quad K_5 = \frac{1}{K}.$$

Прийmemo $\mu = 1 + \mu_{co}$, що дозволить спростити структуру регулятора:

$$\begin{aligned} W_{sr}(s) &= K_0 s^{-(\mu-1)} \left[K_1 s^2 s^{\mu_{co}-1} + K_2 s s^{\mu_{co}-1} + K_3 s^{\mu_{co}-1} + K_4 s^{-1} + K_5 \right] = \\ &= N_1 s + N_2 + N_3 s^{-1} + N_4 s^{-(1+\mu_{co})} + N_5 s^{-\mu_{co}}, \end{aligned} \quad (4.17)$$

де $N_1 = K_0 K_1, N_2 = K_0 K_2, N_3 = K_0 K_3, N_4 = K_0 K_4, N_5 = K_0 K_5$.

Отриманий регулятор (4.17) складається з традиційного PID-регулятора та дробово-інтегральних складових порядків $1 + \mu_{co}$ та μ_{co} .

Для розрахунку коефіцієнтів прийmemo постійну фільтра $T_f = 0.1$ с. Тоді отримаємо:

$$K_0 = \frac{1}{aT_f^{\mu-1}} = 9.39, N_1 = K_0 \frac{a_1}{K} = 236.2, N_2 = K_0 \frac{a_2 b T_f + a_1}{b T_f K} = 222.4,$$

$$N_3 = K_0 \frac{a_2}{b T_f K} = 44.26, N_4 = K_0 \frac{1}{b T_f K} = 23.31, N_5 = K_0 \frac{1}{K} = 81.6.$$

Результати експериментальних досліджень з синтезованими регуляторами представлені нижче.

4.5.3 Експериментальні дослідження

Перехідні процеси з трьома різними регуляторами показані на рис. 4.34. Досліджено переходи з режиму калібрування в режим стабілізації (на 10 ... 15 s) та реакція на збільшення подачі палива після короткочасного зниження швидкості (на 35 ... 45 s) з різними регуляторами.

Так, з PID-регулятором (рис. 4.34, а, б) оператор плавно збільшував подачу палива з 10 до 20 s, а з D^γI^μI- та PIDI^μI^γ-регуляторами (рис. 4.34, с, d, і 5, е, f) оператор діяв ривками.

При проведенні випробувань із-за підвищеного рівня шумів коефіцієнт посилення дробово-диференціальної складової D^γI^μI-регулятора був зменшений до $K_1 = 160$.

Слід також зазначити, що на якість перехідних процесів впливає точність установки візків з ретардерами, положення керма, симетрія навантаження в машині. Тому на графіках показані середні швидкості і середні моменти (n_{ave} и T_{ave}). З отриманих графіків також виділені і суміщені в часі та за середніми швидкостями обох коліс ділянки накиду моменту від двигуна внутрішнього згоряння (рис. 4.35).

Проведені експериментальні дослідження дозволили зіставити швидкодію (як час від першого узгодження до входу в смугу $\pm 2\%$ від заданого значення

швидкості) та перерегулювання за швидкістю в системі з синтезованими регуляторами.

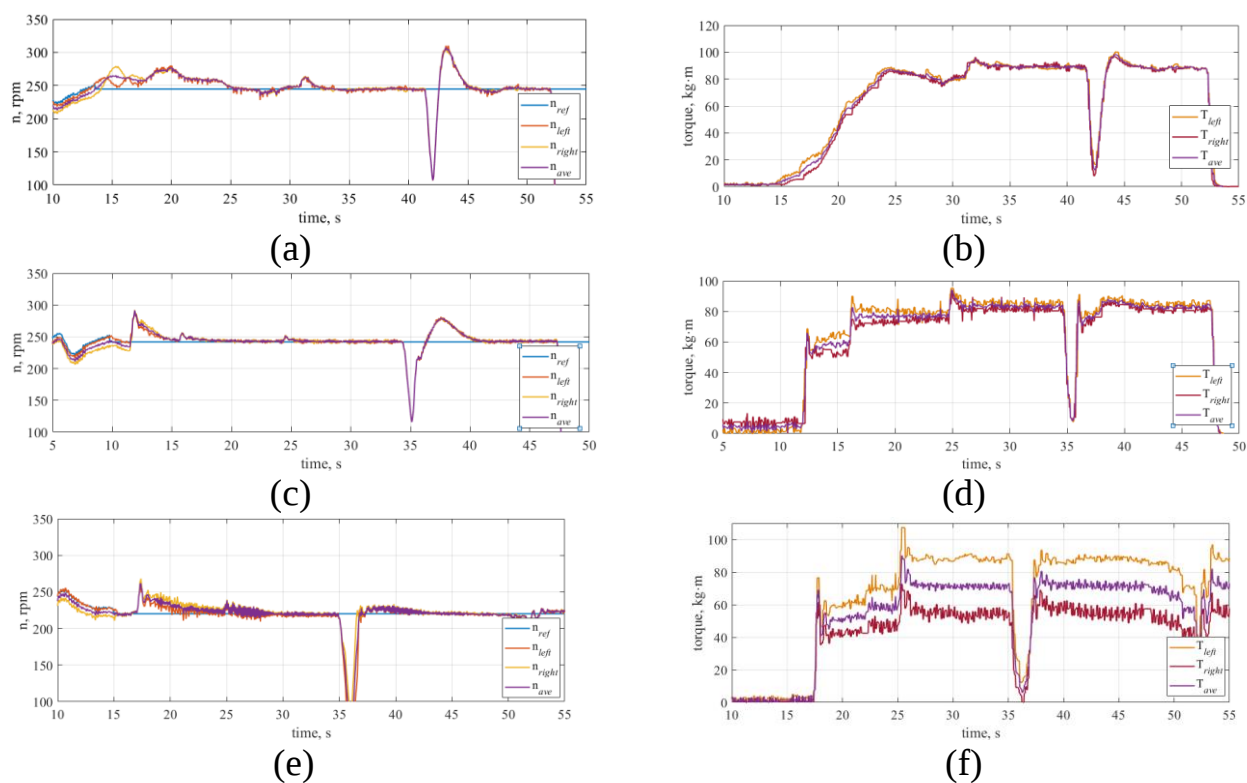


Рис. 4.34. Перехідні процеси швидкості та моменту при різних регуляторах: (а), (b) – PID-регулятор, (с), (d) – $D^{\gamma}I^{\mu}$ -регулятор, (е), (f) – $PID^{\gamma}I^{\mu}$ -регулятор

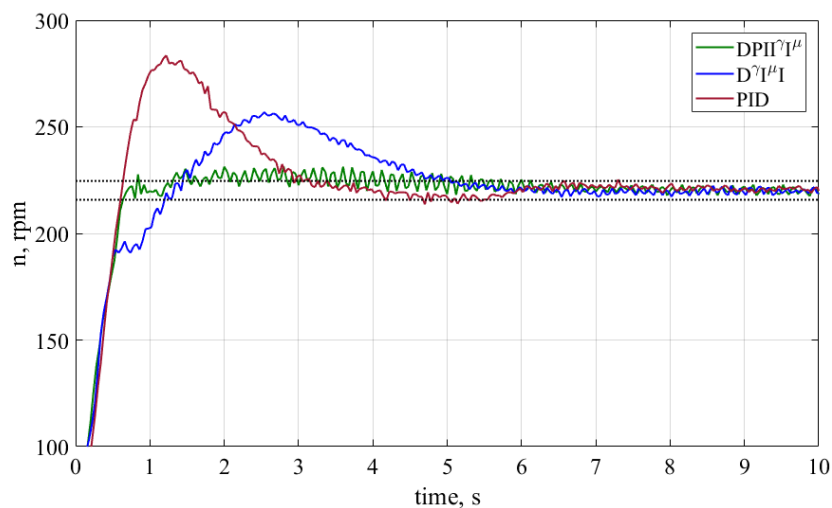


Рис. 4.35. Перехідні процеси швидкості напіввісі двигуна при накиду навантаження з різними регуляторами

PID-регулятор забезпечив кращу швидкодію (2.55 s), але з найбільшим перерегулюванням, яке склало $\delta_1 = 27.6\%$. З D^{PI}I-регулятором перерегулювання склало $\delta_2 = 16.3\%$, а час перехідного процесу – 4.42 s. Найменше перерегулювання отримано з PID^{PI}-регулятором – $\delta_3 = 3.3\%$ при тривалості перехідного процесу 3.8 s.

Важливим етапом перевірки результатів налаштування ДВЗ є вимірювання потужності і максимального моменту при лінійному збільшенні швидкості. Результати такого тесту з найшвидшим (PID) і найточнішим (PID^{PI}) регуляторами наведені на рис. 4.36.

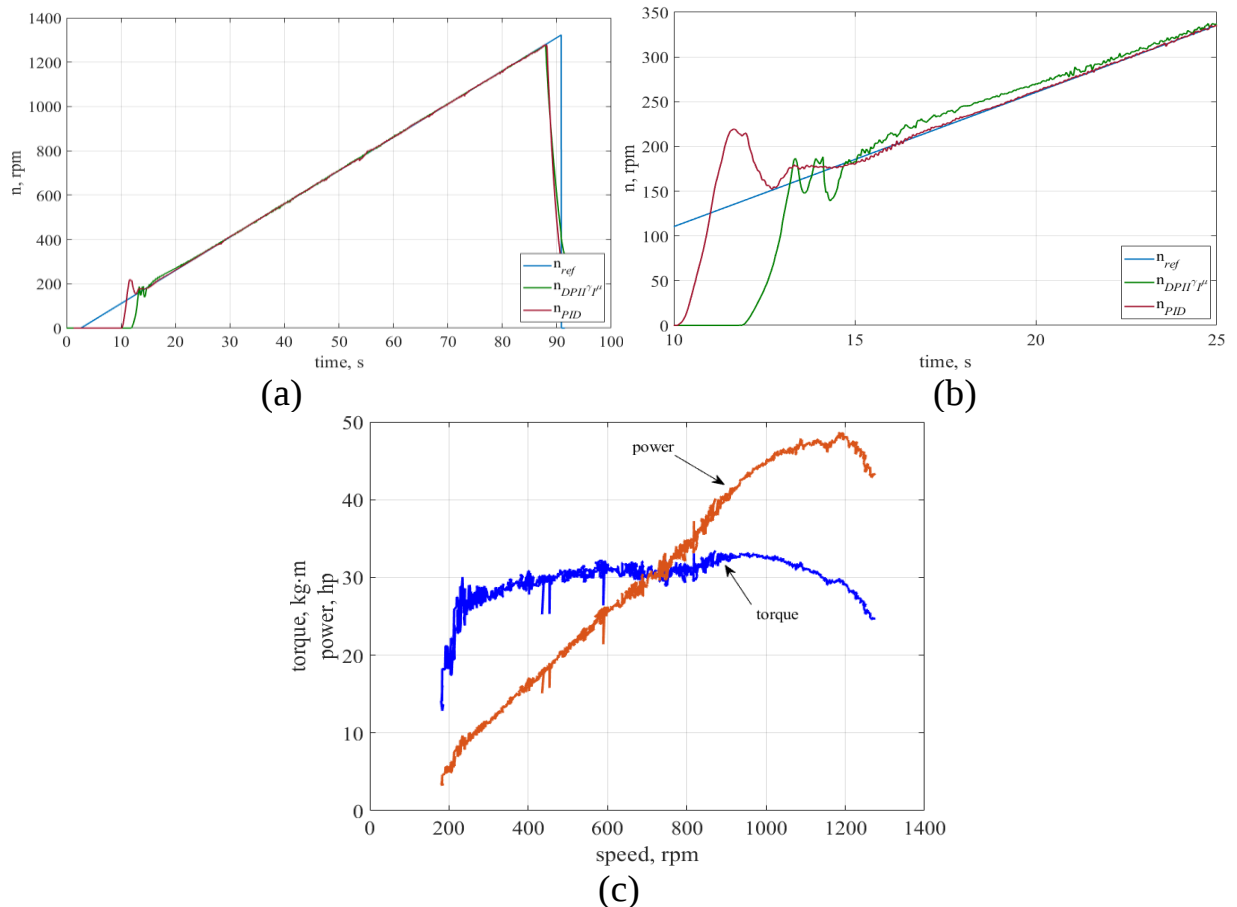


Рис. 4.36. Перехідні процеси по середній швидкості між напіввісями диференціала при вимірюванні потужності, де (а) – весь діапазон, (b) – на початку розгону, (c) – залежність середньої швидкості від середнього моменту на напіввісях диференціала

Обидва регулятори забезпечують стабілізацію швидкості у всьому діапазоні вимірювань потужності, але на початку розгону (рис. 4.36, b) PIDTM-регулятор забезпечує найменше перерегулювання, що дозволило прийняти остаточне рішення вибору структури регулятора в його користь. Крім того, аналіз рис. 4.34–4.36 дозволив також перекоонатися в достатності одноконтурній системи керування без використання датчиків струму [99, 101].

Вимірювання обертового моменту двигуна внутрішнього згоряння при лінійному завданні швидкості з 1000 до 7000 rpm та навпаки дає різні характеристики із-за впливу динамічного моменту. Тому для компенсації помилки формується трикутна діаграма та розраховується середнє значення виміру між моментами при однаковому завданні «прискорення» та «сповільнення» (рис. 4.37).

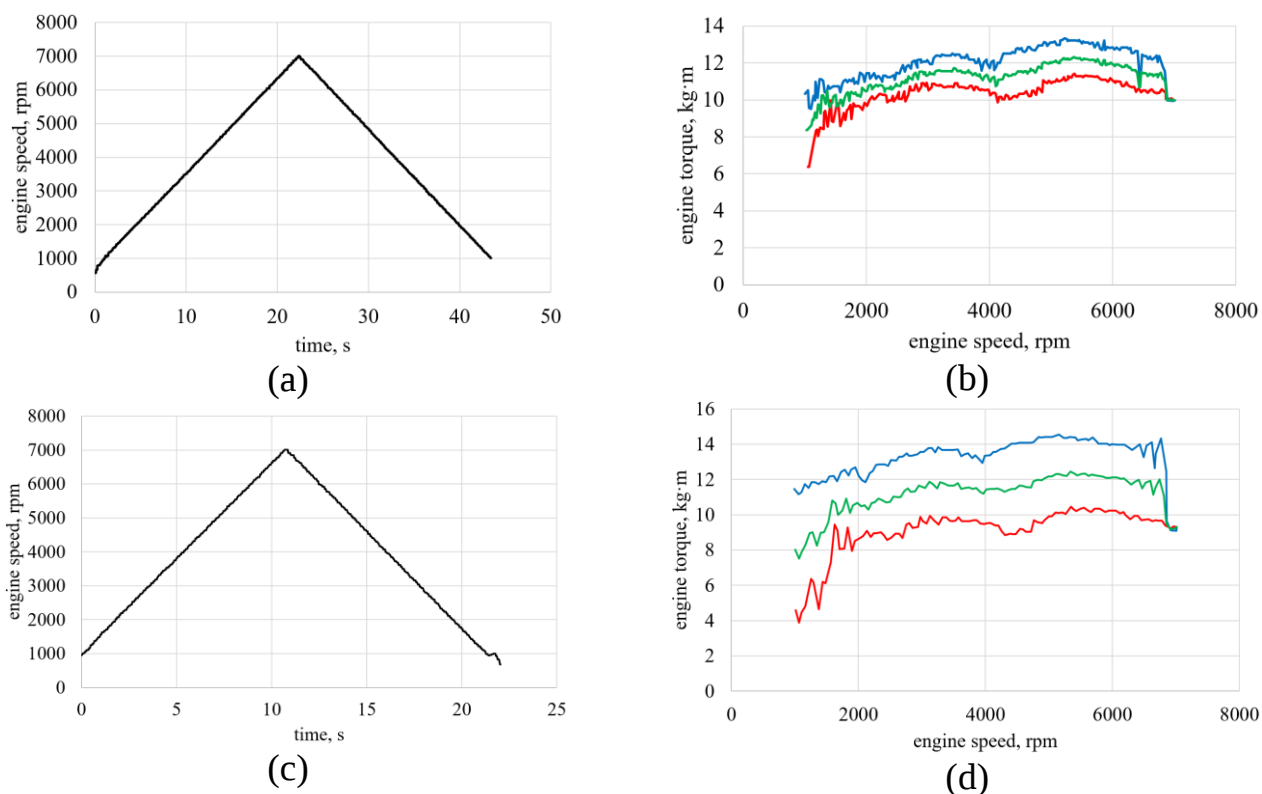


Рис. 4.37. Графіки тахограми з різними прискореннями (a, b) та відповідні механічні характеристики ДВЗ, де червоний – вимірювання при завданні «прискорення», синій – вимірювання при завданні «сповільнення», зелений – середня величина

На рис. 4.37 видно, що середнє значення обертового моменту майже не залежить від прискорення абсолютної величини завдання.

Таким чином, виконано розробку стенда для налаштування паливної системи двигуна внутрішнього згоряння спортивних автомобілів на базі двох електромагнітних гальм, що з'єднуються з ведучими колесами автомобіля, та мікропроцесорної системи керування моментом кожного гальма. Проведений раніше попередній аналіз системи з урахуванням особливостей трансмісії і нелінійних характеристик індукційних гальм та наступні експериментальні дослідження дозволили отримати два види найбільш точної моделі об'єкта керування – інерційний ланка 3-го порядку та дробово-аперіодична ланка порядку 1.6. На підставі отриманих даних синтезовано сімейство регуляторів, що забезпечують роботу стенду в декількох режимах – синхронізація швидкостей обертання коліс в процесі розгону двигуна; стабілізація заданих швидкостей при зміні моменту двигуна від мінімального до максимального; вимірювання максимальної потужності та моменту ДВЗ при лінійно-зростаючої швидкості. Найбільшою швидкістю 2.55 s, але з помітним перерегулюванням 27.6% володіє система з PID-регулятором, настроєним в пакеті «Tune» MATLAB. Система з $D^{\mu}PI^{\mu}$ -регулятором, що забезпечує налаштування на модульний оптимум, характеризується часом перехідного процесу 4.42 s і перерегулюванням 16.3%. Але найменше перерегулюванням і кращу динамічну точність забезпечив дрібно-інтегральний регулятор $PID^{\mu}I^{\mu}$, з яким система характеризується дробовим порядком астатизму 1.6. Таке налаштування стало основою мікропроцесорної системи керування, збору і відображення даних. Представлені графіки (рис. 4.34, 4.36) відображаються на екрані комп'ютера та дозволяють судити про якість налаштування ДВЗ.

Визначимо швидкісну помилку для двох регуляторів з лінійного завдання швидкості (рис. 4.38). Дані для лінійної апроксимації були взяті з часового діапазону з 18 s по 50 s.

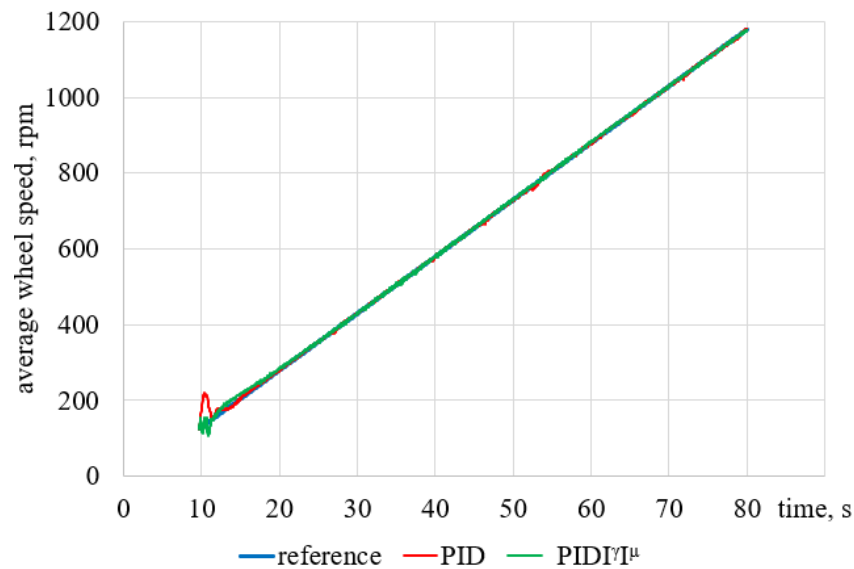


Рис. 4.38. Графік перехідного процесу за швидкістю при вимірюванні потужності з двома регуляторами

Рівняння для сигналу завдання швидкості буде в обох випадках однаково:

$$n_{ref}(t) = 15 \cdot t - 20.4 \quad (4.18)$$

Апроксимація лінійним рівнянням при PID-регуляторі:

$$n_{PID}(t) = 14.978 \cdot t - 19.242 \quad (4.19)$$

Апроксимація лінійним рівнянням при $PIDIT^\mu$ -регуляторі:

$$n_{PIDIT^\mu}(t) = 14.92 \cdot t - 16.49 \quad (4.20)$$

Знайдемо різницю між рівнянням завдання (4.18) та середньої швидкістю при PID-регуляторі (4.19):

$$n_{\Delta PID}(t) = (15 - 14.978) \cdot t + (-20.4 - (-19.242)) = 0.022 \cdot t - 1.158 \quad (4.21)$$

Знайдемо різницю між рівнянням завдання (4.18) та середньої швидкістю при $PIDIT^{\mu}$ -регуляторі (4.20)

$$n_{\Delta PIDIT^{\mu}}(t) = (15 - 14.92) \cdot t + (-20.4 - (-16.49)) = 0.08 \cdot t - 3.91 \quad (4.22)$$

Як видно з (4.21) та (4.22) швидкісна помилка при $PIDIT^{\mu}$ -регуляторі зменшується майже в три рази швидше, ніж при PID-регуляторі.

ВИСНОВКИ ПО РОЗДІЛУ 4

1. Виконано розробку системи керування стенда для налаштування паливної системи двигуна внутрішнього згоряння автомобілів на базі двох індукційних ретардерів, що з'єднуються з ведучими колесами автомобіля, та мікропроцесорної системи керування моментом кожного гальма.
2. Експериментальні дослідження дозволили отримати два види найбільш точних моделей об'єкта керування – інерційна ланка 3-го порядку та ланка порядку 1.6.
3. Найбільшу швидкодію 2.55 s, але з помітним перерегулюванням 27.6% має система з PID-регулятором, налаштованим в пакеті «Tune» MATLAB. Система з DIT^{μ} -регулятором, що забезпечує налаштування на модульний оптимум, характеризується часом перехідного процесу 4.42 s і перерегулюванням 16.3%. Але найменше перерегулювання та кращу динамічну точність забезпечив дробово-інтегральний регулятор $PIDIT^{\mu}$, з яким система характеризується дробовим порядком астатизму 1.6. Таке налаштування стало основою мікропроцесорної системи керування.

ВИСНОВКИ

Основні результати виконаної дисертаційної роботи полягають в наступному:

1. На підставі експериментальних досліджень виконано ідентифікацію якірного ланцюга двигуна послідовного збудження передавальною функцією з дробовим порядком. Серед розглянутих динамічних моделей найменшу похибку забезпечила передавальна функція порядку $1 + \mu_{co}$ ($\mu_{co} \approx 0.35$):

$$W_{co3} = \frac{K}{a_1 s^{1+\mu} + a_0 s^{\mu} + 1}. \text{ Для синтезу регуляторів з об'єктом керування з дробовим}$$

порядком 1.35 контур струму налаштовано на модульний оптимум ($D^{\mu}I^{\gamma}$) та оптимумами з дробовим порядком астатизму 0.35 ($PI^{\gamma}D$), 0.6 ($D^{\mu}I^{\gamma}I^{\gamma}$), 1.35 ($PID I^{\mu}I^{\gamma}$) та 1.5 ($D^{\epsilon}D^{\rho}D^{\delta}I^{\gamma}$). Показано, що при обмеженому значенні напруги живлення налаштування не дають бажаного перехідного процесу. Тому застосовано генетичний алгоритм для пошуку оптимальних коефіцієнтів в регуляторах. Найкращий результат отримано при налаштуванні контуру на дробовий порядок астатизму $1 + \mu_{co}$ (1.35), тобто з $PID I^{\mu}I^{\gamma}$ -регулятором, при якому відсутнє перерегулювання та забезпечується мінімальний час першого узгодження. Експериментальні дослідження з різними значеннями сигналу завдання струму з таким регулятором показав, що це налаштування характеризує систему, як самоподібну. Тому стає допустимим аналіз і синтез регуляторів для контуру струму таких двигунів класичними методами теорії автоматичного керування для лінійних систем.

2. Виконано моделювання перехідних процесів швидкості в вентильно-реактивному двигуні з урахуванням нелінійної залежності моменту та потокозчеплення від струму та положення ротора, які отримано в програмному забезпеченні Ansys MAXWELL, та при різних прикладених напругах з постійними кутами комутації. Показано, що найменшу похибку забезпечує дробово-

аперіодична функція $W_{co}(s) = \frac{K}{a_0 s^\mu + 1}$. Показано, що в залежності від напруги статора коефіцієнти K та a_0 змінюють своє значення, тому для компенсації цієї нелінійності необхідно змінювати коефіцієнти регуляторів від значення сигналу завдання швидкості. Синтезовано два види регуляторів, один при налаштуванні контуру швидкості на модульний оптимум (ΓI -регулятор) та на оптимум з дробовим порядком астатизму 1.71 ($PI^{\mu}\Gamma$ -регулятор). Кращий результат з меншим перерегулюванням (до 2 %) та часом першого узгодження має система з $PI^{\mu}\Gamma$ -регулятором.

3. Виконано налаштування системи керування двома електромагнітними гальмами в стенді прецизійного регулювання паливної системи двигуна внутрішнього згоряння автомобілів. За експериментальними даними об'єкт керування апроксимовано аперіодичною функцією 3-го порядку та передавальною функцією з дробовим порядком 1.6. На основі першої динамічної моделі синтезовано PID-регулятор, на основі дробової моделі – $D^{\mu}\Gamma I$ - та $PI D^{\mu}\Gamma$ -регулятори. Показано, що найменше перерегулювання та кращу динамічну точність забезпечує дробово-інтегральний регулятор $PI D^{\mu}\Gamma$, з яким система характеризується дробовим порядком астатизму 1.6. Такий регулятор став основою мікропроцесорної системи керування та дозволив виконувати налаштування паливної системи в режимі стабілізації швидкості та вимірювати максимальні значення моменту та потужності двигуна при формуванні трикутної тахограми.

4. На підставі досліджень трьох типів електричних машин можна зробити загальний висновок, що дробово-інтегральне числення надає можливість опису перехідних процесів в системах зі степеневими властивостями (у разі електричних машин причиною їх існування є насичення магнітної системи) як лінійних об'єктів з диференціальними рівняннями дробового порядку, що, в свою чергу, дозволяє використовувати класичний апарат теорії автоматичного керування для лінійних

систем для аналізу таких об'єктів та синтезу регуляторів, які забезпечують оптимізацію динамічних режимів різних за принципом роботи електричних машин за мінімумом перерегулювання та часом першого узгодження. Так, з використанням регуляторів з приблизно однаковою структурою отримано оптимальні показники перехідних процесів в контурі струму двигуна послідовного збудження отримано з $PID^{PI^{\gamma}}$ -регулятором, в контурі швидкості вентильно-реактивного двигуна – з $PI^{PI^{\gamma}}$ -регулятором та при керуванні електромагнітними гальмами – з $PID^{PI^{\gamma}}$ -регулятором.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Kimeu M. J. Fractional Calculus: Definitions and Applications [Електронний ресурс] / М. J. Kimeu. – 2009. – Режим доступу до ресурсу: <https://digitalcommons.wku.edu/theses/115/>
2. Самко С. Г. Интегралы и производные дробного порядка и некоторые их приложения / С. Г. Самко, А. А. Килбас, О. И. Маричев. – Минск: Наука и техника, 1987. – 688 р.
3. Miller D. A. Fractional Calculus [Електронний ресурс] / D. A. Miller. – 2004. – Режим доступу до ресурсу: https://www.researchgate.net/publication/258109711_FRACTIONAL_CALCULUS, doi: 10.13140/RG.2.1.2473.0966
4. Lioville J. Memoire sur quelques questions de gemetrie et de mecanique, et sur un nouveau genre de calcui pour resoudre ces questions / J. Lioville. // J.L'Ecole Roy Polytechn. – 1832. – P. 69.
5. Lioville J. Memoire sur le calcut des differentielles a indices quelconques / J. Lioville. // Ibit. – 1832. – P. 71–168.
6. Lioville J. Memoire sur le theorem des fonctions complementaires / J. Lioville. // Math. Bd. – 1834. – №11. – P. 1–19.
7. Lioville J. Memoire sur une formule d'analyse / J. Lioville. // Ibid. – 1835. – Bd12, №4. – P. 273–287.
8. Lioville J. Memoire sur le changement de la variable independante dans le calcul des differentielles a indices quelconques / J. Lioville. // J.L'Ecole Roy Polytechn. – 1835. – T. 15, sect. 24. – P. 17–65.
9. Lioville J. Memoire sur l'integration des equations differentielles a indices fractionnaires / J. Lioville. // Ibid. – 1837. – T. 15, N 55. – P. 58–84.
10. Lazarević M. P. Advanced Topics on Applications of Fractional Calculus on Control Problems, System Stability and Modelin [Електронний ресурс] / М. P. Lazarević,

- M. R. Rapaić, T. B. Sekara // WSEAS Press. – 2014. – Режим доступа до ресурсу: <https://www.researchgate.net/publication/312137269> Introduction to Fractional Calculus with Brief Historical Background
11. B. Riemann. Versuch einer allgemeinen Auffassung der Integration und Differentiation / B. Riemann. // Gesammelte Mathematische Werke. – 1876. – P. 331–344.
 12. Sonin N. Y. On differentiation with arbitrary index / N. Y. Sonin. // Moscow Matem. Sbornik. – 1869. – №6 (1). – С. 1–38.
 13. Letnikov A. V. Theory of differentiation with an arbitrary index / A. V. Letnikov. // Moscow Matem. Sbornik. – 1868. – №3. – P. 1–66.
 14. Letnikov A. V. An explanation of the concepts of the theory of differentiation of arbitrary index / A. V. Letnikov. // Moscow Matem. Sbornik. – 1872. – №6. – P. 413–445.
 15. Laurent H. Sur le calcul des derivees a indices quelconques / H. Laurent. // Nouv. Annales de Mathematiques. – 1884. – №3 (3). – P. 240–252.
 16. Grunwald A. K. Ueber “begrenzte” Derivationen und deren Anwendung [Электронний ресурс] / A. K. Grunwald // Niedersächsische Staats- und Universitätsbibliothek Göttingen. – 1867. – Режим доступа до ресурсу: <https://www.deutsche-digitale-bibliothek.de/item/7OWDE57YKUI6KP2N2ZI3SDEW2B2EZ6AP>.
 17. Heaviside O. Electrical papers / O. Heaviside. // The Macmillan Company. – 1892. – №8. – Режим доступа до ресурсу: <https://www.cambridge.org/core/books/electrical-papers/E8DA42E7CD2341B67DFAF486F4027EE0>.
 18. Weyl H. Bemerkungen zum Begriff des Differentialquotienten gebrochener Ordnung. Vierteljahr. Naturforsch. Gesellsch [Электронний ресурс] / H. Weyl. // Zürich – 1917. – № 62 – P. 296–302. – Режим доступа до ресурсу: https://www.ngzh.ch/archiv/1917_62/62_1-2/62_21.pdf

19. Marchaud M. A. Sur les dérivées et sur les différences des fonctions de variables réelles / M. A. Marchaud. // *J. math. pures et appl* – № 6 (4) – 1927. – P. 371–382 – Режим доступа до ресурсу: http://www.numdam.org/item/THESE_1927__78__1_0/.
20. Caputo M. Linear Models of Dissipation whose Q is almost Frequency Independent-II / M. Caputo. // *Geophys. J. Royal astr. Soc.* – № 13 – 1967. – P. 529–539 – Режим доступа до ресурсу: <https://academic.oup.com/gji/article/13/5/529/592934>.
21. Caputo M. Linear models of dissipation in anelastic solids / M. Caputo, F. Mainardi. // *Rivista del Nuovo Cimento*. – 1971. – №1. – P. 161–198.
22. Ross B. Fractional Calculus and Its Applications / B. Ross // *Proceedings of the International Conference held at the University of New Haven*. – vol. 457 – 1974. – Режим доступа до ресурсу: <https://www.springer.com/gp/book/9783540071617>.
23. Oldham K. B. The fractional calculus / K. B. Oldham, J. Spanier. – New York-London: Academic Press, 1974. – Режим доступа до ресурсу: <https://www.sciencedirect.com/bookseries/mathematics-in-science-and-engineering/vol/111/suppl/C>.
24. Miller K. S. An introduction to the fractional calculus and fractional differential equations / K. S. Miller, B. Ross. – New York: A Wiley-Interscience Publication, 1993. – (John Wiley & Sons Inc).
25. Samko S. Fractional Integrals and Derivatives: Theory and Applications / S. Samko, A. A. Kilbas, O. I. Marichev. – Yverdon: Gordon and Breach Science Publishers, 1993. – 1006 p.
26. Podlubny I. An introduction to fractional derivatives, fractional differential equations, to methods of their solution and some of their applications / I. Podlubny. – San Diego: Academic Press Inc., 1999.
27. Uchaikin V. V. Fractional Derivatives for Physicists and Engineers / V. V. Uchaikin. – Berlin: Springer, Berlin, Heidelberg, 2013. – 385 p.

28. Учайкин В. В. Метод дробных производных / В. В. Учайкин. – Ульяновск: Артишок, 2008. – 512 с.
29. Kimeu J. M. Fractional calculus: Definitions and Applications [Электронный ресурс] / Joseph M. Kimeu // Masters Theses & Specialist Projects. – 2009. – Режим доступа до ресурсу: <https://digitalcommons.wku.edu/theses/115>
30. Нахушев А. М. Дробное исчисление и его применение / А. М. Нахушев. – Москва: Физматлит, 2003. – 272 с.
31. Westerlund S. Dead matter has memory! / S. Westerlund. // Physica Scripta. – 1961. – №43. – Р. 174–179. – Режим доступа до ресурсу: <https://iopscience.iop.org/article/10.1088/0031-8949/43/2/011>
32. Westerlund S. Causality / S. Westerlund. // University of Kalmar, Tech. Rep. – 1994. – №940426.
33. Caputo M. Free modes splitting and alterations of electrochemically polarizable media / M. Caputo. // Rend. Fis. Acc. Lincei. – 1993. – №9(4). – С. 89–98. – Режим доступа до ресурсу: <https://link.springer.com/article/10.1007%2F03001421>
34. Westerlund S. Einstein's Relativity – and what it really is / S. Westerlund. // Preprint SE-39351, Kalmar Sweden. – 2003. – №5.
35. Васильев В. В. Дробное исчисление и аппроксимационные методы в моделировании динамических систем / В. В. Васильев, Л. А. Симак. – Киев: НАН Украины, 2008. – 256 р.
36. Fractional order systems: Modeling and Control Applications / R.Caponetto, G. Dongola, L. Fortuna, I. Petras. – Singapore: World Scientific, 2010. – 196 p.
37. Podlubny I. Fractional Differential Equations / I. Podlubny. – New York: Academic Press, 1999. – 365 p.
38. Yamamoto, S. Present Status and Future Needs: The View from Japanese Industry / S. Yamamoto, I. Hashimoto // Proc. 4th International Conf. on Chemical Process Control, Texas. – 1991.

39. Bialkowski W. L. Dreams versus reality: A view from both sides of the gap / W. L. Bialkowski. // Pulp and Paper Canada. – 1994. – №11. – P. 19–27.
40. Datta A. Structure and Synthesis of PID Controllers / A. Datta, H. Ming-Tzu, S. P. Bhattacharyya. – London: Springer Science & Business Media, 2013. – 235 p.
41. Tan K. K. Advances in PID Control / K. K. Tan, Q. Wang, C. C. Hang. – London: Springer-Verlag London Limited, 1999. – 264 p.
42. Podlubny I. Fractional-order systems and $PI^{\alpha}D^{\mu}$ -controllers / I. Podlubny. // IEEE Transactions on Automatic Control. – 1999. – №1 (44). – P. 208 – 214.
43. Petras I. The fractional-order controllers: methods for their synthesis and application / I. Petras. // J. of Electrical Engineering. – 1999. – №50. – P. 284–288.
44. Lozynskyy O. Synthesis and research of electromechanical systems described by fractional order transfer functions / O. Lozynskyy, A. Lozynskyy, B. Kopchak, Y. Paranchuk, P. Kalenyuk and Y. Marushchak// 2017 International Conference on Modern Electrical and Energy Systems (MEES), Kremenchuk – 2017 – P. 16-19. doi: 10.1109/MEES.2017.8248877
45. Chen Y. Fractional Order Control - A Tutorial / Y. Chen, I. Petras, D. Xue. // American Control Conference, Hyatt Regency Riverfront, St. Louis, MO, USA June 10-12. – 2009.
46. Xue D. A Comparative Introduction of Four Fractional Order Controllers / D. Xue, Y. Q. Chen. // Proc. of the 4th World Congress on Intelligent Control and Automation, Shanghai, China. – 2002. – P. 3228–3235.
47. Chen Y. Q. Robust controllability of interval fractional order linear time invariant systems / Y. Q. Chen, H. S. Ahn, D. Xue. // Signal Processing. – 2006. – №86. – P. 2794–2802.
48. Lurie B. J. Three-parameter tunable Tilt-Integral-Derivative (TID) controller [Электронный ресурс] / Boris J. Lurie // Jet Propulsion Lab. California Inst. of Tech., Pasadena, CA., United States. – 1994. – Режим доступа до ресурсу: <https://ntrs.nasa.gov/citations/19950011898>.

49. Oustaloup A. The CRONE control of resonant plants: application to a flexible transmission / A. Oustaloup, B. Mathieu, P. Lanusse. // European Journal of Control. – 1995. – №1 (2). – Режим доступа до ресурсу: https://www.academia.edu/17607234/The_CRONE_Control_of_Resonant_Plants_Application_to_a_Flexible_Transmission
50. Oustaloup A. La commande CRONE / Alain Oustaloup. – Paris: Hermes Science Publications, 1991. – 495 p.
51. Oustaloup A. La Derivation Non Entiere: Theorie, Synthese et Applications / A. Oustaloup. – Paris: Hermès, 1995. – 508 p.
52. Auto-tuning of Fractional Lead-Lag Compensators [Электронний ресурс] / [C. A. Monje,, B. M. Vinagre, A. J. Calderón та ін.] // IFAC Proceedings Volumes. – 2005. – Режим доступа до ресурсу: <https://doi.org/10.3182/20050703-6-CZ-1902.00453>.
53. Бушер В. В. Повышение качества электротехнических систем с дробными интегрально-дифференцирующими звеньями. Дисс.докт.техн.наук: 05.09.03 – Электротехнические комплексы и системы. – Одеса: ОНПУ, 2013. – 270 p.
54. Kopchak B. Devising a procedure for the synthesis of electromechanical systems with cascade-enabled fractional-order controllers and their study [Text] / B. Kopchak, Y. Marushchak, A. Kushnir // Eastern-European Journal of Enterprise Technologies ISSN 1729–3774. Industry Control System. – Kharkov: PC Technology Center – 2019. – Vol 5, No 2 (101) – P. 65–71. Access mode: <https://doi.org/10.15587/1729-4061.2019.177320>
55. Fractional-order Systems and Controls. Fundamentals and Applications / [C. Monje, Y. Chen, M. Vinagre and others]. – London: Springer, London, 2010. – 430 p.
56. Charef A. Modeling and Analog Realization of the Fundamental Linear Fractional Order Differential Equation / A. Charef. // Nonlinear Dynamics. – 2006. – №46. – P. 195–210. – Режим доступа до ресурсу: <https://link.springer.com/article/10.1007%2Fs11071-006-9023-2>

57. Calderón A. J. Fractional order control strategies for power electronic buck converters / A. J. Calderón, B. M. Vinagre, V. Feliu. // Signal Processing. – 2006. – №86. – P. 2803–2819. – Режим доступу до ресурсу: <https://doi.org/10.1016/j.sigpro.2006.02.022>
58. An overview of the CRONE approach in system analysis, modeling and identification, observation and control / [A. Oustaloup, J. Sabatier, P. Lanusse та ін.]. // in Proc. of the 17th World Congress IFAC, Soul, Korea, July 6-11. – 2008. – P. 14254–14265. – Режим доступу до ресурсу: <https://doi.org/10.3182/20080706-5-KR-1001.02416>.
59. Марущак Я. Ю. Побудова регуляторів дробового порядку для електромеханічних систем / Я. Ю. Марущак, Б. Л. Копчак // Електротехнічні та комп'ютерні системи. – 2017. – № 25. – P. 26-33. http://nbuv.gov.ua/UJRN/etks_2017_25_6.
60. Valério D. D. Fractional Robust System Control : дис. канд. техн. наук / Valério Duarte Pedro Mata de Oliveira – Universidade Técnica de Lisboa Instituto Superior , 2005. – 217 p.
61. Two digital realizations of fractional controllers: application to temperature control of a solid / B. M. Vinagre, I. Petras, P. Merchan, L. Dorcak. // European Control Conference (ECC), IEEE. – 2001. – P. 1764–1767. – Режим доступу до ресурсу: <https://ieeexplore.ieee.org/document/7076176>
62. Chen Y. Discretization schemes for fractional-order differentiators and integrators / Y. Chen, K. L. Moore. // IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications. – 2002. – №49. – P. 363–367. – Режим доступу до ресурсу: <https://ieeexplore.ieee.org/document/7076176>
63. Vinagre B. M. Two direct Tustin discretization methods for fractional-order differentiator/integrator / B. M. Vinagre, Q. Y. Chen, I. Petras. // Journal of Franklin Institute. – 2003. – №340. – P. 349–362. – Режим доступу до ресурсу: <https://www.sciencedirect.com/science/article/abs/pii/S0016003203000498>

64. Al-Alaoui M. A. Novel digital integrator and differentiator / M. A. Al-Alaoui. // Electronics Letters. – 1993. – №29. – P. 376–378. – Режим доступа до ресурсу: <https://ieeexplore.ieee.org/document/4602529>
65. Tseng C. C. Design of fractional order digital FIR differentiator / C. C. Tseng. // IEEE Signal Processing Letters. – 2001. – №8. – P. 77–79. – Режим доступа до ресурсу: <https://ieeexplore.ieee.org/document/905945>
66. Hwang C. A note on time-domain simulation of feedback fractional-order systems / C. Hwang, J. F. Leu, S. Y. Tsay. // IEEE Transactions on Automatic Control. – 2002. – №47. – P. 625–631. – Режим доступа до ресурсу: <https://ieeexplore.ieee.org/document/995039>
67. Matrix approach to discrete fractional calculus II: Partial fractional differential equations / [I. Podlubny, A. Chechkin, T. Skovranek та ін.]. // Journal of Computational Physics. – 2009. – №228. – P. 3137–3153. Режим доступа до ресурсу: <https://doi.org/10.1016/j.jcp.2009.01.014>
68. Dorcak L. Numerical Models for Simulation the Fractional-Order Control Systems [Електронний ресурс] / L. Dorcak // Slovak Academy of Sciences, Inst. of Experimental Physic. – 1994. – Режим доступа до ресурсу: <https://arxiv.org/pdf/math/0204108.pdf>
69. Damarla S. K. Fractional Order Processes. Simulation Identification and Control / S. K. Damarla, M. Kundu. – New York: CRC Press, 2019. – 349 p.
70. Busher V. Synthesis and technical realization of the control systems with the digital fractional integral-differentiating regulators / V. Busher, A. Aldairi. // Eastern-European Journal of Enterprise Technologies. – 2018. – №4. – P. 63–71. doi: 10.15587/1729-4061.2018.139892
71. Техническая реализация дробных интегрально-дифференцирующих регуляторов в микропроцессорных системах управления / В.Бушер, К. Хандакжи, В. Горошко, В. Кравченко. // International conference SHIPS'

ELECTRICAL ENGINEERING, ELECTRONICS AND AUTOMATION, Odessa: NU «ОМА». – 2018. – С. 180–186.

72. Вешеневский С. Н. Характеристики двигателей в электроприводе / С. Н. Вешеневский. – Москва: Энергия, 1977. – 431 с.
73. Кувшинов А. А. Теория электропривода / А. А. Кувшинов, Э. Л. Греков. – Оренбург: ГОУ ОГУ, 2009. – 197 с.
74. Бушер В. В. Дробные интегрально-дифференцирующие регуляторы в системе управления двигателем постоянного тока с последовательным возбуждением [Текст] / В. В. Бушер, В. В. Горошко // Електромеханічні та енергозберігаючі системи. Методи моделювання та оптимізації. XXVII МНТК 11-12 квітня 2019 р. ISSN 2079-5106. – Кременчук: КрНУ, 2019. – С. 39–41. Access at: http://esmo.kdu.edu.ua/publ/ESMO_2019.pdf
75. Busher, V. Synthesis and implementation of fractional-order controllers in a current circuit of the motor with series excitation [Text] / V. Busher, L. Melnikova, V. Horoshko // Eastern-European Journal of Enterprise Technologies ISSN 1729–3774. Industry Control System. – Kharkov: PC Technology Center, 2019. – Vol. 2, No 2 (98), (2019). – P. 63–72. Access mode: <http://journals.uran.ua/eejet/article/view/161352> doi: 10.15587/1729-4061.2019.161352
76. Kuznetsov V. Switched reluctance motor: schoolbook / V. Kuznetsov, V. Kuz'michev. – Moscow: Publisher MEI, 2003. – 70 p.
77. Miller T. J. Electronic control of switched reluctance motors / T. J. Miller. – Glasgow: SPEED Laboratory, 2001. – 272 p.
78. Savvinov P. V. Overview of Switched Reluctance / P. V. Savvinov, A. S. Semenov. // Modern High Technologies. – 2003. – №8. – Available at: <https://top-technologies.ru/pdf/2013/8-2/169.pdf>. (accessed 11.03.2019)
79. Krishnan R. Switched Reluctance Motor Drives: Modeling, Simulation, Analysis, Design / R. Krishnan. – New York: CRC Press, 2001. – 432 p.

80. Srinivas P. Direct torque control of 4 phase 8/6 switched reluctance motor drive for constant torque load / P. Srinivas, P. Prasad, N. Venkata. // World Journal of Modelling and Simulation. – 2005. – №3. – P. 185–193.
81. Inderka R. B. DITC-direct instantaneous torque control of switched reluctance drives / R. B. Inderka, R. W. Doncker. // IEEE Transactions on Industry Applications. – 2003. – №4. – P. 1046–1051.
82. Dee J. Instantaneous Torque Control of SRM with a Logical Torque Sharing Method / J. Dee, Z. Lee, J. Ahn. // IEEE Annual Power Electronics Specialists Conference. – 2007. – P. 1784–1789. doi: 10.1109/PESC.2007.4342271
83. Cetin E. Genetic PI controller for a switched reluctance motor drive / E. Cetin, Y. Tuncay. // International XII Turkish Symposium on Artificial Intelligence and Neural Networks, Tainn. – 2003.
84. Xiu J. An application of adaptive fuzzy logic controller for switched reluctance motor drive / J. Xiu, C. Xia. // in Proc. IEEE Fuzzy Syst. Knowl. Disc. – 2007. – P. 154–158.
85. Salah A. Speed Control of Switched Reluctance Motor Using Artificial Neural Network Controller / A. Salah, H. M. Hany, G. Yasser. // Computational Intelligence and Information Technology: First International Conference, CIIT, Pune, India. – 2011. – P. 6–14.
86. Xia C. Single Neuron PID Control for Switched Reluctance Motors O Based on RBF Neural Network / C. Xia, M. Wang. // Proceedings of the CSEE. – 2005. – №15. – P. 161–165.
87. Xiu J. GA-Based Adaptive Fuzzy Logic Controller for Switched Reluctance Motor / J. Xiu, C. Xia. // Transactions of China Electrotechnical Society. – 2007. – №11. – P. 69–73.
88. Hari Prabhu M. Fuzzy Logic Based PI Closed Loop Control of Switched Reluctance Motor Drives Using Z-Source Inverter / M. Hari Prabhu, S. Mahendran, V.

- Rangarajan. // International Journal of Engineering Research & Technology. – 2013. – №9. – P. 2330–2340.
89. Anomalous relaxation in dielectrics. Equations with fractional derivatives. / V. Novikov, K. Wojciechowski, O. Komkova, O. Thiel. // Materials Science-Poland. – 2005. – №4. – P. 977–984.
 90. Tarasov V. E. Fractional Dynamics. Applications of Fractional Calculus to Dynamics of Particles, Fields and Media / V. E. Tarasov. – Berlin: Higher Education Press, 2010. – 505 p. doi: 10.1007/978-3-642-14003-7
 91. Hilfer R. Applications of Fractional Calculus in Physics / R. Hilfer. – New York: World Scientific, 2000. – 472 p. doi: 10.1142/3779
 92. Бушер В. В. Модифицированный метод управления моментом вентильно-реактивного двигателя [Текст] / В. В. Бушер, В. В. Горошко, К. А. Хандакжи // Электротехнические и компьютерные системы, ISSN Print 2221-3937, ISSN Online 2221-3805 – ELTEKS-2017. – Киев: Техника, 2017. – Вып. 25(101). – С. 44–52. Access mode: <http://etks.opu.ua/core/getfile.php?id=2185>, <http://dx.doi.org/10.15276/eltecs.25.101.2017.05>
 93. Busher V.V., Goroshko V.V. Fractional Integral-Differentiating Control in Speed Loop of Switched Reluctance Motor [Текст] // Problemele Energeticii Regionale – vol. 1, No 2(42) – 2019 – P. 46–54. – ELTEKS-2019. http://journal.ie.asm.md/assets/files/05_12_41_2019.pdf <https://doi.org/10.5281/zenodo.3239166> <http://journal.ie.asm.md/en/contents/electronnii-jurnal-1241-2019>,
 94. Liu C. Y. Design and Use of an Eddy Current Retarder In an Automobile / C. Y. Liu, K. J. Jiang, Y. Zhang. // International Journal of Automotive Technology. – 2011. – №4. – P. 611–616. doi: 10.1007/s12239-011-0071-3
 95. Anwar S. A parametric model of an eddy current electric machine for automotive braking applications / S. Anwar. // IEEE Transactions on Control Systems Technology. – 2004. – №3. – P. 422–427. doi: 10.1109/TCST.2004.824293

96. V Kalmakov V. A. The Ignition System of a Car / V. A. Kalmakov, A. A. Andreev. – Cheljabinsk, 2014.
97. Kakaee A. H. Sensitivity and Effect of Ignition Timing on the Performance of a Spark Ignition Engine: An Experimental and Modeling Study / A. H. Kakaee, M. H. Shojaeefard, J. Zareei. // Journal of Combustion. – 2011. – P. 1–8. doi: 10.1155/2011/678719
98. Morselli R. Detailed and Reduced Dynamic Models of Passive and Active Limited-slip Car Differentials / R. Morselli, R. Zanasi, G. Sandoni. // Mathematical and Computer Modelling of Dynamical Systems. – 2004. – P. 1–20.
99. Tarasik V. P. Differential drives modeling of mobile machines driving wheels / V. P. Tarasik, O. V. Puzanova, V. I. Kurstak. // Vestnik Belorussko-Rossiyskogo universiteta. – 2009. – №24. – P. 1–12.
100. Horosko, V. Control optimization of electromechanical systems by fractional-integral controllers / Technology Audit and Production Reserves. Information and Control Systems. p-ISSN 2664-9969, e-ISSN 2706-5448. – Kharkov: PC Technology Center, 2020. – Vol. 3, No 2 (53), (2020). – P. 56–59. doi: <https://doi.org/10.15587/2706-5448.2020.207037>
101. Горошко, В. В. Применение аппарата дробного исчисления в системах управления электрическими машинами / Theoretical and Practical Foundations of Social Process Management. XXIII International Scientific and Practical Online Conference, San Francisco, USA, 29–30 June, 2020. – с. 172–176. doi: <https://doi.org/10.46299/ISG.2020.XXIII>, ISBN 978-1-64871-431-3.

ДОДАТОК А

Матеріали впровадження результатів дисертаційної роботи

«УТВЕРЖДАЮ»

Директор станции технического
обслуживания

«СПД Грушевой В. С.»

Грушевой В. С.

2 августа 2020 г.

АКТ ВНЕДРЕНИЯ

результатов диссертационной работы Горошко Василия Владимировича
«Оптимизация динамических режимов электрических машин с
использованием дробно-интегральных регуляторов»
представленной на соискание ученой степени доктора философии по
специальности 141 – Электроэнергетика, электротехника и
электромеханика

Настоящим актом подтверждаем, что на станции технического обслуживания «СПД Грушевой В. С.» при непосредственном участии аспиранта Горошко В. В. разработана и запущена в эксплуатацию система управления стендом для настройки и тестирования двигателей внутреннего сгорания (ДВС) на базе двух электромагнитных тормозов TELMA с регулируемым моментом.

Система управления, состоящая из отладочной платы STM32DISCOVERY с микропроцессором серии STM32F429I и одноключевых широтно-импульсных преобразователей, обеспечивает три режима работы, соответствующие технологическим требованиям к стенду:

- режим **калибровки**;
- режим **стабилизации** скоростей ведущих колес;
- режим **измерения момента и мощности** ДВС.

Режим **калибровки** предназначен для синхронизации вращения ведущих колес при разгоне ДВС до заданной скорости с рассогласованием не

более 5 %. В режиме стабилизации система управления обеспечивает поддержание заданной скорости вращения вала ДВС во всем диапазоне момента при отсутствии автоколебаний и противофазных колебаний скоростей колес и точностью не хуже $\pm 0,5$ %, что необходимо для настройки угла зажигания и состава топливно-воздушной смеси. В режиме измерения момента и мощности ДВС при максимальной подаче топлива формируется треугольная тахограмма с заданным темпом разгона/торможения (125-250 об/мин/с) с точностью не менее $\pm 0,2$ %, что позволяет скомпенсировать динамическую погрешность и получить при равных скоростях в процессах разгона и торможения средние величины момента и мощности ДВС.

В системе управления электромагнитными тормозами использованы PID^{IT}-регуляторы, обеспечивающие дробный порядок астатизма 1.6, соответствующий полученным в результате обработки экспериментальных данных параметрам и порядку дробно-аперидической передаточной функции, описывающей объект управления со среднеквадратичной погрешностью не более 1,3 %. Это обеспечивает отсутствие статической ошибки в режиме стабилизации и динамической ошибки при линейно-изменяющемся задающем сигнале, перерегулирование в переходных процессах при стабилизации скорости не более 3,3 %, время первого согласования 0,8 с и длительность переходного процесса 3,8 с.

Разработанный графический интерфейс предоставляет оператору возможность переключения режимов работы, задания или определения передаточного числа трансмиссии, отображения текущих значений измеряемых координат и сохранения данных в файлах для последующего формирования отчетных документов, графиков, таблиц.

Технический директор СТО

Кочетков Ю. В.

ЗАТВЕРДЖУЮ

проректор з науково-педагогічної та виховної
роботи ОНПУ

д.т.н., проф. Нестеренко С.А.

« 03 » серпня 2020 р.

ДОВІДКА

про науково-педагогічну діяльність

Горошка Василя Володимировича

в процесі підготовки роботи

«Оптимізація динамічних режимів електричних машин з використанням дробово-інтегральних регуляторів»

на здобуття наукового ступеня доктора філософії за спеціальністю 141 –

«Електроенергетика, електротехніка та електромеханіка»

Під час навчання в аспірантурі Горошко В. В. пройшов педагогічну підготовку – проводив на кафедрі електромеханічної інженерії лабораторні роботи з дисциплін «Основи автоматизованого проектування ЕМС», «Програмування мікропроцесорів і промислових контролерів для керування ЕМС та ЕТК», «Мікропроцесорна техніка та її програмування», «Моделювання ЕМС та пристроїв», «Теорія дискретних систем автоматичного керування» та «Комплектні електроприводи».

Горошко В. В. брав участь у весняній школи AIST-2017, яка проходила з 08.05.2017 по 12.05.2017 в м. Миколаїв в рамках ERASMUS+ проекту «Internet of Things: Emerging Curriculum for Industry and Human Application» (ALIOT), де було представлено прототип макету розумного будинку, який став початком циклу лабораторних робіт на кафедрі. Особистий вклад Горошко В. В. – створення основної програми плати Arduino MEGA 2560 для керування підсистемами клімат контролю, освітлення, аварійної безпеки і т. д., Горошко В. В. є співавтором практикуму з симуляції інтернету речей:

Simulation of Internet of Things. Based Systems. Practicum / O. V. Drozd, D. A. Maevsky, O. J. Maevskaya, V. V. Goroshko and other, D. A. Maevsky (ed.) – Ministry of Education and Science of Ukraine, Odessa National Polytechnic University, 2019. – 130 p. – ISBN 978-617-7361-95-3. https://aliot.eu.org/wp-content/uploads/2019/10/ALIOT_PC1_Sim-of-IoT-Sys_web.pdf

Матеріали дисертаційної роботи буде використано при підготовці оновлених версій конспекту лекцій та методичних вказівок до лабораторних робіт з дисципліни «Прикладні задачі і методи оптимізації складних електромеханічних систем» для аспірантів третього (освітньо-наукового) рівня вищої освіти спеціальності 141 «Електроенергетика, електротехніка та електромеханіка»

Директор ІЕЕ
доктор технічних наук, професор

Бойко А.О.

Завідувач кафедри
«Електромеханічна інженерія»,
доктор технічних наук, професор

Маєвський Д.А.



Funded by the
Erasmus+ Programme
of the European Union



CERTIFICATE



is presented to



Vasyl Goroshko

for participation in
International Spring Training School AIST 2017
for students, early-career researchers and lecturers



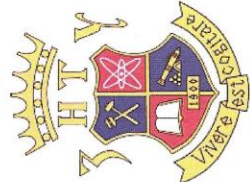
“Internet of Things: Emerging Curriculum for Industry and Human Applications”



co-organized by

Petro Mohyla Black Sea National University
National Aerospace University named after N.E. Zhukovsky “KhAI”

Course duration: lectures, training and independent studies 20 hours



Mykolaiv, Ukraine, 2017 May 8-12

Prof. Chris Phillips

Prof. Chris Phillips
Grantholder of Erasmus+ ALIoT project,
Newcastle University



Prof. Leonid Klymenko
Rector of

Petro Mohyla Black Sea National University



Університет аеронавтики та космонавтики
імені Юрія Тютюнника



University of Chernivsi



ДОДАТОК Б

Код розробленого програмного забезпечення для керування, моніторингу та аналізу даних стенда для налаштування паливної системи двигунів внутрішнього згоряння автомобілів на базі двох електромагнітних гальм.

Проект складається з п'ятих папок – controllers, fxml, main, model, threads.

В папці controllers знаходяться класи – DataResult.java, MainController.java, MessageBox.java. В fxml розміщені файли розмітки інтерфейсу – result-data.fxml, telma.fxml. В main – Main.java. В model – Const.java, RowTelma.java, TableTelma.java, Telma.java, TriangleData.java. В threads – RequestHandler.java, SimpleSocketServer.java.

DataResult.java

```
package controllers;
import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.fxml.FXML;
import javafx.fxml.Initializable;
import javafx.scene.chart.LineChart;
import javafx.scene.chart.XYChart;
import javafx.scene.control.*;
import javafx.scene.control.Button;
import javafx.scene.control.TextField;
import javafx.scene.control.cell.PropertyValueFactory;
import javafx.scene.input.KeyCode;
import javafx.scene.input.KeyEvent;
import javafx.scene.input.MouseEvent;
import javafx.stage.FileChooser;
import javafx.stage.Stage;
import model.*;

import javax.imageio.ImageIO;
import java.awt.*;
import java.awt.image.BufferedImage;
import java.io.*;
import java.net.URL;
import java.sql.Timestamp;
import java.text.SimpleDateFormat;
import java.util.*;

public class DataResult implements Initializable {

    @FXML
    private LineChart lineChart;
    @FXML
    private TableView<RowTelma> tblStaticTelma;
```

```

@FXML
private TableColumn<RowTelma, Float> clmSpeed;
@FXML
private TableColumn<RowTelma, Float> clmTorque;
@FXML
private TableColumn<RowTelma, Float> clmPower;
@FXML
private TextField tfMaxTorq;
@FXML
private TextField tfSpeedTorq;
@FXML
private TextField tfMaxPower;
@FXML
private TextField tfSpeedPower;
@FXML
private TextField tfEfficiency;
@FXML
private TextField tfDelBefore;
@FXML
private TextField tfDelAfter;
@FXML
private TextField tfFileName;
@FXML
private TextField tfFileToLoad;
@FXML
private Button btnDelBefore;
@FXML
private Button btnDelAfter;
@FXML
private Button btnFileChooser;

@FXML
private Button btnSaveToFile;
@FXML
private Button btnSaveScreen;
@FXML
private Button btnClose;

private static final SimpleDateFormat sdf = new
SimpleDateFormat("yyyy.MM.dd.HH.mm.ss");

private XYChart.Series<Number, Number> serTorqueAve;
private XYChart.Series<Number, Number> serPowerAve;

private XYChart.Series<Number, Number> serTorqueF, serPowerF;

private ObservableList<RowTelma> table;

private Map<controllers.DataResult.T_FIELD, List> dataTelma;

private final char DELIMETER = ',';

private enum FIELD {
    REF_SPEED("Ref speed rpm"),

```

```

    AVE_SPEED("Ave speed rpm"),
    LEFT_SPEED("Left speed rpm"),
    RIGHT_SPEED("Right speed rpm"),
    AVE_TORQUE("Ave torque kg*m"),
    LEFT_TORQUE("Left torque kg*m"),
    RIGHT_TORQUE("Right torque kg*m"),
    POWER("Power HP");

    private String name;
    FIELD(String name){
        this.name = name;
    }

    @Override
    public String toString(){
        return this.name;
    }
}

private enum T_FIELD{
    UP_REF_SPEED("Up Ref speed rpm"),
    DOWN_REF_SPEED("Down Ref speed rpm"),

    UP_AVE_SPEED("Up Ave speed rpm"),
    UP_LEFT_SPEED("Up Left speed rpm"),
    UP_RIGHT_SPEED("Up Right speed rpm"),

    DOWN_AVE_SPEED("Down Ave speed rpm"),
    DOWN_LEFT_SPEED("Down Left speed rpm"),
    DOWN_RIGHT_SPEED("Down Right speed rpm"),

    UP_AVE_TORQUE("Up Ave torque kg*m"),
    UP_LEFT_TORQUE("Up Left torque kg*m"),
    UP_RIGHT_TORQUE("Up Right torque kg*m"),

    DOWN_AVE_TORQUE("Down Ave torque kg*m"),
    DOWN_LEFT_TORQUE("Down Left torque kg*m"),
    DOWN_RIGHT_TORQUE("Down Right torque kg*m"),

    UP_POWER("Up Power HP"),
    DOWN_POWER("Down Power HP"),

    AVE_SPEED("Ave speed rpm"),
    AVE_TORQUE("Ave torque kg*m"),
    AVE_POWER("Ave Power HP");

    private String name;
    T_FIELD(String name){
        this.name = name;
    }

    @Override
    public String toString(){
        return this.name;
    }
}

```

```

private enum LIMIT{
    BEFORE,
    AFTER
}

private float EFFICIENCY = 1.0f;

@Override
public void initialize(URL location, ResourceBundle resources) {
    //set test data

    // set marker off
    lineChart.setCreateSymbols(false);

    // create new symbols
    serTorqueUp.setName("Момент при подъеме, Н*м");
    serTorqueDown = new XYChart.Series<>();
    serTorqueDown.setName("Момент при спуске, Н*м");*/
    serTorqueAve = new XYChart.Series<>();
    serTorqueAve.setName("Момент средний, Н*м");

    serPowerUp.setName("Мощность при подъеме, ЛС");
    serPowerDown = new XYChart.Series<>();
    serPowerDown.setName("Мощность при спуске, ЛС");*/
    serPowerAve = new XYChart.Series<>();
    serPowerAve.setName("Мощность средняя, ЛС");

    serTorqueF = new XYChart.Series<>();
    serTorqueF.setName("Момент средний (из файла), Н*м");
    serPowerF = new XYChart.Series<>();
    serPowerF.setName("Мощность средняя (из файла), ЛС");

    // add series
    lineChart.getData().add(serTorqueAve);

    lineChart.getData().add(serPowerAve);

    lineChart.getData().add(serTorqueF);
    lineChart.getData().add(serPowerF);

    TriangleData triangleData =
        new TriangleData(TableTelma.getLeftSpeedList(),
            TableTelma.getRightSpeedList(), TableTelma.getAveSpeedList(),
            TableTelma.getLeftTorqueList(),
            TableTelma.getRightTorqueList(), TableTelma.getAveTorqueList(),
            TableTelma.getPowerList(),
            TableTelma.getRefSpeedList());

    //=====

    int sizeR = TableTelma.getRefSpeedList().size();
    if(sizeR == 0 ){
        return;
    }

```

```

        dataTelma = new HashMap<>();

        dataTelma.put(DataResult.T_FIELD.UP_REF_SPEED,
triangleData.getUpRefSpeed());
        dataTelma.put(DataResult.T_FIELD.DOWN_REF_SPEED,
triangleData.getDownRefSpeed());

        dataTelma.put(DataResult.T_FIELD.UP_LEFT_SPEED,
triangleData.getUpSpeedLeft());
        dataTelma.put(DataResult.T_FIELD.UP_RIGHT_SPEED,
triangleData.getUpSpeedRight());
        dataTelma.put(DataResult.T_FIELD.UP_AVE_SPEED,
triangleData.getUpSpeedAve());

        dataTelma.put(DataResult.T_FIELD.DOWN_LEFT_SPEED,
triangleData.getDownSpeedLeft());
        dataTelma.put(DataResult.T_FIELD.DOWN_RIGHT_SPEED,
triangleData.getDownSpeedRight());
        dataTelma.put(DataResult.T_FIELD.DOWN_AVE_SPEED,
triangleData.getDownSpeedAve());

        dataTelma.put(DataResult.T_FIELD.UP_LEFT_TORQUE,
triangleData.getUpTorqueLeft());
        dataTelma.put(DataResult.T_FIELD.UP_RIGHT_TORQUE,
triangleData.getUpTorqueRight());
        dataTelma.put(DataResult.T_FIELD.UP_AVE_TORQUE,
triangleData.getUpTorqueAve());

        dataTelma.put(DataResult.T_FIELD.DOWN_LEFT_TORQUE,
triangleData.getDownTorqueLeft());
        dataTelma.put(DataResult.T_FIELD.DOWN_RIGHT_TORQUE,
triangleData.getDownTorqueRight());
        dataTelma.put(DataResult.T_FIELD.DOWN_AVE_TORQUE,
triangleData.getDownTorqueAve());

        dataTelma.put(DataResult.T_FIELD.UP_POWER, triangleData.getUpPower());
        dataTelma.put(DataResult.T_FIELD.DOWN_POWER, triangleData.getDownPower());

        dataTelma.put(DataResult.T_FIELD.AVE_SPEED, triangleData.getAveSpeed());
        dataTelma.put(DataResult.T_FIELD.AVE_TORQUE, triangleData.getAveTorque());
        dataTelma.put(DataResult.T_FIELD.AVE_POWER, triangleData.getAvePower());

//=====

        setLine(serTorqueAve.getData(), dataTelma.get(T_FIELD.AVE_SPEED),
dataTelma.get(T_FIELD.AVE_TORQUE), 9.8f);

        setLine(serPowerAve.getData(),
dataTelma.get(DataResult.T_FIELD.AVE_SPEED), dataTelma.get(T_FIELD.AVE_POWER),
1.0f);

//=====

```

```

        clmSpeed.setCellValueFactory(new PropertyValueFactory<>("speed"));
        clmTorque.setCellValueFactory(new PropertyValueFactory<>("torque"));
        clmPower.setCellValueFactory(new PropertyValueFactory<>("power"));

        table = getRows();
        tblStaticTelma.setItems(table);

//=====

        // set date as file name
        Timestamp timestamp = new Timestamp(System.currentTimeMillis());
        String dateFile = sdf.format(timestamp);
        tfFileName.setText(dateFile);

//=====

        // choose file
        FileChooser fileChooser = new FileChooser();
        fileChooser.setTitle("Выберите файл с предыдущем замером мощности");
        fileChooser.getExtensionFilters().addAll(
            new FileChooser.ExtensionFilter("CSV files", "*.csv")
        );

        fileChooser.setInitialDirectory(new File("."));

//=====

        btnFileChooser.setOnAction(new EventHandler<ActionEvent>() {
            @Override
            public void handle(ActionEvent event) {
                File selectedFile =
fileChooser.showOpenDialog(btnFileChooser.getScene().getWindow());
                if(selectedFile != null) {
                    tfFileToLoad.setText(selectedFile.getName());
                    List<Float> listAveSpeed = getListFromFile(selectedFile,
T_FIELD.AVE_SPEED.toString());
                    List<Float> listAveTorque = getListFromFile(selectedFile,
T_FIELD.AVE_TORQUE.toString());
                    List<Float> listPower = getListFromFile(selectedFile,
T_FIELD.AVE_POWER.toString());

                    if(listAveSpeed == null || listAveTorque == null || listPower
== null){
                        return;
                    }

                    serTorqueF.getData().clear();
                    serPowerF.getData().clear();

                    EFFICIENCY = 1.0f;
                    setLine(serTorqueF.getData(), listAveSpeed, listAveTorque,
9.8f);
                    setLine(serPowerF.getData(), listAveSpeed, listPower, 1.0f);

```

```

    }

    });

    btnClose.setOnAction(event -> {
        Stage stage = (Stage) btnClose.getScene().getWindow();
        stage.close();
    });

    btnSaveToFile.setOnAction(event -> {
        String fileName = tfFileName.getText().trim();

        try {
            FileWriter writer = new FileWriter("./" + fileName + ".csv");

            String sRow;

            int rowSize = dataTelma.size();
            String[] arrRow = new String[rowSize];

            //=====
            // set name column
            int num = 0;
            for(Map.Entry<T_FIELD, List> entry: dataTelma.entrySet()){
                arrRow[num] = entry.getKey().toString();

                ++num;
            }

            sRow = concat(arrRow);
            writer.write(sRow);
            writer.write('\n'); // newline

            //=====
            // set through all data list
            int listSize = dataTelma.get(T_FIELD.AVE_SPEED).size();

            for(int i = 0; i < listSize; ++i){

                num = 0;
                for (Map.Entry<T_FIELD, List> entry: dataTelma.entrySet()) {

                    //if list is empty
                    if(entry.getValue().size() == 0){
                        arrRow[num] = "0.0";
                        ++num;
                        continue;
                    }

                    arrRow[num] = String.valueOf(entry.getValue().get(i));
                    ++num;
                }
            }
        }
    });

```

```

        sRow = concat(arrRow);
        writer.write(sRow);
        writer.write('\n'); // newline
    }

    writer.close();

    } catch (IOException e) {
        e.printStackTrace();
    }
});

btnDelBefore.setOnAction(event -> {
    float delSpeedBefore = Float.parseFloat(tfDelBefore.getText());

    deleteData(delSpeedBefore, LIMIT.BEFORE);
});

btnDelAfter.setOnAction(event -> {
    float delSpeedAfter = Float.parseFloat(tfDelAfter.getText());

    deleteData(delSpeedAfter, LIMIT.AFTER);
});

btnSaveScreen.setOnAction(new EventHandler<ActionEvent>() {
    @Override
    public void handle(ActionEvent event) {
        String fileName = tfFileName.getText().trim();

        // saves screenshot to desired path
        try {
            Robot r = new Robot();

            String path = "./" + fileName + ".png";

            Rectangle capture = new
Rectangle(Toolkit.getDefaultToolkit().getScreenSize());
            BufferedImage image = r.createScreenCapture(capture);
            ImageIO.write(image, "png", new File(path));
            System.out.println("Screenshot saved");

        } catch (IOException | AWTException e) {
            e.printStackTrace();
        }
    }
});

tfEfficiency.setOnKeyPressed(new EventHandler<KeyEvent>() {
    @Override
    public void handle(KeyEvent event) {
        if (event.getCode().equals(KeyCode.ENTER)) {
            EFFICIENCY = Float.parseFloat(tfEfficiency.getText());

            serPowerAve.getData().clear();

```

```

        setLine(serTorqueAve.getData(),
dataTelma.get(T_FIELD.AVE_SPEED), dataTelma.get(T_FIELD.AVE_TORQUE), 9.8f);

        setLine(serPowerAve.getData(),
dataTelma.get(DataResult.T_FIELD.AVE_SPEED), dataTelma.get(T_FIELD.AVE_POWER),
1.0f);

        tblStaticTelma.getItems().clear();
        table = getRows();
        tblStaticTelma.setItems(table);

        findMaxTorquePoint();

        tfEfficiency.setStyle("-fx-background-color: #7CFC00;");
    }
}

});

tfEfficiency.setOnMouseClicked(new EventHandler<MouseEvent>() {
    @Override
    public void handle(MouseEvent event) {
        tfEfficiency.setStyle("-fx-background-color: white;");
    }
});

findMaxTorquePoint();
}

private void setLine(ObservableList<XYChart.Data<Number, Number>> list, List
listX, List listY, float xK){
    if(listX == null || listY == null){
        return;
    }

    for(int i = 0; i < listX.size(); ++i){
        float x = (float) listX.get(i);
        float y = (float) listY.get(i);
        y = y * xK / EFFICIENCY;
        list.add(new XYChart.Data<Number, Number>(x, y));
    }
}

private ObservableList<RowTelma> getRows(){
    ObservableList<RowTelma> rows = FXCollections.observableArrayList();

    List lSpeed = dataTelma.get(T_FIELD.AVE_SPEED);
    List lTorque = dataTelma.get(T_FIELD.AVE_TORQUE);
    List lPower = dataTelma.get(T_FIELD.AVE_POWER);

    float speedVal = 0.0f;
    float torqueVal = 0.0f;
    float powerVal = 0.0f;

```

```

        for(int i = 0; i < lSpeed.size(); ++i){
            speedVal = Math.round((float)lSpeed.get(i) * 10) / 10.0f;
            torqueVal = ((float)lTorque.get(i)) / EFFICIENCY;
            powerVal = ((float)lPower.get(i)) / EFFICIENCY;

            rows.add(new RowTelma(speedVal, torqueVal, powerVal));
        }

        return rows;
    }

    private void scaleData(ObservableList<XYChart.Data<Number, Number>> list,
                           List originList,
                           float scale){

        int size = list.size();

        for (int i = 0; i < size; ++i) {
            float yOldVal = (float) originList.get(i);
            list.get(i).setYValue(yOldVal * scale);
        }
    }

    private String concat(String[] arrS) {
        StringBuilder row = new StringBuilder(arrS[0]);
        row.append(DELIMETER);

        for(int i = 1; i < arrS.length; ++i){
            row.append(arrS[i]).append(DELIMETER);
        }

        return row.toString();
    }

    private void setTestData(){
        File file = new File("C:/Users/kings/Documents/idea-
        projects/telmaServerTCP2/2020.06.03.18.14.51 tiangle.csv");

        List<Float> listAveSpeed = getListFromFile(file,
        FIELD.AVE_SPEED.toString());
        List<Float> listLeftSpeed = getListFromFile(file,
        FIELD.LEFT_SPEED.toString());
        List<Float> listRightSpeed = getListFromFile(file,
        FIELD.RIGHT_SPEED.toString());

        List<Float> listAveTorque = getListFromFile(file,
        FIELD.AVE_TORQUE.toString());
        List<Float> listLeftTorque = getListFromFile(file,
        FIELD.LEFT_TORQUE.toString());
        List<Float> listRightTorque = getListFromFile(file,
        FIELD.RIGHT_TORQUE.toString());

        List<Float> listPower = getListFromFile(file, FIELD.POWER.toString());
        List<Float> listRefSpeed = getListFromFile(file,
        FIELD.REF_SPEED.toString());
    }

```

```

        TableTelma.setListAveSpeed(listAveSpeed);
        TableTelma.setListLeftSpeed(listLeftSpeed);
        TableTelma.setListRightSpeed(listRightSpeed);

        TableTelma.setListAveTorque(listAveTorque);
        TableTelma.setListLeftTorque(listLeftTorque);
        TableTelma.setListRightTorque(listRightTorque);

        TableTelma.setListPower(listPower);
        TableTelma.setListRefSpeed(listRefSpeed);
    }

    private void findMaxTorquePoint() {
        List lSpeed = dataTelma.get(T_FIELD.AVE_SPEED);
        List lTorque = dataTelma.get(T_FIELD.AVE_TORQUE);
        List lPower = dataTelma.get(T_FIELD.AVE_POWER);

        int size = lSpeed.size();

        float maxTorque = 0;
        float speedTorq = 0;

        float maxPower = 0;
        float speedPower = 0;

        for (int i = 0; i < size; ++i) {
            float currSpeed = (float)lSpeed.get(i);
            float currTorque = ((float)lTorque.get(i)) / EFFICIENCY;
            float currPower = ((float)lPower.get(i)) / EFFICIENCY;

            if(currTorque > maxTorque){
                maxTorque = currTorque;
                speedTorq = currSpeed;
            }

            if(currPower > maxPower){
                maxPower = currPower;
                speedPower = currSpeed;
            }
        }

        tfMaxTorq.setText(String.valueOf(maxTorque));
        tfSpeedTorq.setText(String.valueOf(speedTorq));

        tfMaxPower.setText(String.valueOf(maxPower));
        tfSpeedPower.setText(String.valueOf(speedPower));
    }

    private void deleteData (float fLimit, LIMIT type){
        boolean isFind = false;

```

```

int index = 0;

for (int i = 0; i < dataTelma.get(T_FIELD.AVE_SPEED).size(); ++i) {
    float currSpeed =(float) dataTelma.get(T_FIELD.AVE_SPEED).get(i);
    if(currSpeed >= fLimit){
        isFind = true;
        index = i;
        break;
    }
}

if(!isFind){
    return;
}

Iterator<Map.Entry<T_FIELD, List>> itMap =
dataTelma.entrySet().iterator();

while (itMap.hasNext()){
    Map.Entry<T_FIELD, List> entry = itMap.next();

    Iterator<Float> itList = entry.getValue().iterator();
    int curInd = 0;
    while (itList.hasNext()){
        itList.next();

        if(type.equals(LIMIT.AFTER)) {
            if (curInd > index) {
                itList.remove();
            }
        }else{
            if (curInd < index) {
                itList.remove();
            }
        }

        ++curInd;
    }
}

findMaxTorquePoint();
serTorqueAve.getData().clear();
serPowerAve.getData().clear();
setLine(serTorqueAve.getData(), dataTelma.get(T_FIELD.AVE_SPEED),
dataTelma.get(T_FIELD.AVE_TORQUE), 9.8f);
setLine(serPowerAve.getData(),
dataTelma.get(DataResult.T_FIELD.AVE_SPEED), dataTelma.get(T_FIELD.AVE_POWER),
1.0f);
tblStaticTelma.getItems().clear();
table = getRows();
tblStaticTelma.setItems(table);
}

private List<Float> getListFromFile(File file, String nameColumn){
    List<Float> records = new LinkedList<>();
    BufferedReader br = null;

```

```

try {
    br = new BufferedReader(new FileReader(file));

    // first row
    String line = br.readLine();
    String[] row;

    //find number of column by name
    int numColumn = -1;
    if(line != null){
        row = line.split(Const.COMMA_DELIMITER);

        for(int i = 0; i < row.length; ++i){
            if(nameColumn.equals(row[i])){
                numColumn = i;
            }
        }

    }

    if(numColumn == -1){
        br.close();
        return null;
    }

    while ((line = br.readLine()) != null) {
        row = line.split(Const.COMMA_DELIMITER);
        records.add(Float.parseFloat(row[numColumn]));
    }

    br.close();

} catch (IOException e) {
    e.printStackTrace();
}

return records;
}
}

```

MainController.java

```

package controllers;

import eu.hansolo.medusa.Gauge;
import javafx.beans.value.ChangeListener;
import javafx.beans.value.ObservableValue;
import javafx.collections.ObservableList;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.fxml.Initializable;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.chart.LineChart;

```

```

import javafx.scene.chart.NumberAxis;
import javafx.scene.chart.XYChart;
import javafx.scene.control.Button;
import javafx.scene.control.ProgressIndicator;
import javafx.scene.control.TextField;
import javafx.scene.input.KeyCode;
import javafx.scene.input.KeyEvent;
import javafx.scene.input.MouseEvent;
import javafx.scene.layout.HBox;
import javafx.stage.Stage;
import model.Const;
import model.TableTelma;
import model.Telma;
import org.controlsfx.control.ToggleSwitch;
import threads.SimpleSocketServer;

import java.io.IOException;
import java.net.*;
import java.util.ResourceBundle;

public class MainController implements Initializable {

    @FXML
    private HBox mainPane;
    @FXML
    private ProgressIndicator progressIndicator;

    @FXML
    private Gauge ggEngineSpeed;
    @FXML
    private Gauge ggEngineTorque;
    @FXML
    private Gauge ggEnginePower;

    @FXML
    private Gauge ggRightSpeed;
    @FXML
    private Gauge ggLeftSpeed;

    @FXML
    private Gauge ggRightTorque;
    @FXML
    private Gauge ggLeftTorque;

    @FXML
    private ToggleSwitch tglRun;
    @FXML
    private ToggleSwitch tglStable;

    @FXML
    private TextField tfRefEngine;
    @FXML
    private Button btnCalc;
    @FXML
    private TextField tfGearRatio;

```

```

@FXML
private TextField tfStable;

@FXML
private TextField tfStartSpeed;
@FXML
private TextField tfFinalSpeed;
@FXML
private TextField tfRiseTime;

@FXML
private ToggleSwitch tglBtnWriteData;

@FXML
private Button btnStartServer;
@FXML
private Button btnExit;
@FXML
private Button btnWriteData;

@FXML
private LineChart<Number, Number> lineChart;

@FXML
private NumberAxis xAxis;
@FXML
private NumberAxis yAxis;

private XYChart.Series<Number, Number> serSpeed, serTorque, serPower,
serRefSpeed;

private SimpleSocketServer serverTCP;

private boolean isWriteData = false;
private boolean isActionMode = false;
private boolean isRun = false;
private boolean isStable = false;

private int refSpeedStable = 0;

private int stepVal = -1;

@Override
public void initialize(URL location, ResourceBundle resources) {
    // set marker off
    lineChart.setCreateSymbols(false);

    // create new symbols
    serSpeed = new XYChart.Series<>();
    serSpeed.setName("Скорость двигателя, об/мин");
    serTorque = new XYChart.Series<>();
    serTorque.setName("Момент, кг*м");
    serPower = new XYChart.Series<>();
    serPower.setName("Мощность, ЛС");
    serRefSpeed = new XYChart.Series<>();
    serRefSpeed.setName("Задание, об/мин");
}

```

```

// add series
lineChart.getData().add(serRefSpeed);
lineChart.getData().add(serSpeed);
lineChart.getData().add(serToque);
lineChart.getData().add(serPower);

//set gear ratio text
tfGearRatio.setText("1.0");

progressIndicator.setVisible(false);
tglBtnWriteData.setDisable(false);

btnStartServer.setOnAction(new EventHandler<ActionEvent>() {
    @Override
    public void handle(ActionEvent event) {
        startServer();
    }
});

btnExit.setOnAction(new EventHandler<ActionEvent>() {
    @Override
    public void handle(ActionEvent event) {
        closeProgram();
    }
});

btnCalc.setOnAction(new EventHandler<ActionEvent>() {
    @Override
    public void handle(ActionEvent event) {
        int refSpeedEngine = Integer.parseInt(tfRefEngine.getText());
        Telma.setGearRatio(refSpeedEngine / Telma.getAveSpeedRPM());

        tfGearRatio.setText(String.valueOf(Telma.getGearRatio()));
    }
});

tglBtnWriteData.selectedProperty().addListener(new
ChangeListener<Boolean>() {
    @Override
    public void changed(ObservableValue<? extends Boolean> observable,
Boolean oldValue, Boolean newValue) {
        isWriteData = tglBtnWriteData.isSelected();
        System.out.println(isWriteData + "");
        if(isWriteData){
            serSpeed.getData().clear();
            serToque.getData().clear();
            serPower.getData().clear();
            serRefSpeed.getData().clear();

            TableTelma.clearRefSpeed();
            TableTelma.clearAveSpeed();

            TableTelma.clearLeftSpeed();
            TableTelma.clearRightSpeed();
        }
    }
});

```

```

        TableTelma.clearAveTorque();
        TableTelma.clearLeftTorque();
        TableTelma.clearRightTorque();

        TableTelma.clearPower();

        stepVal = -1;
    }
}
});

tglRun.selectedProperty().addListener(new ChangeListener<Boolean>() {
    @Override
    public void changed(ObservableValue<? extends Boolean> observable,
        Boolean oldValue, Boolean newValue) {
        int startSpeed =
Math.abs(Integer.parseInt(tfStartSpeed.getText()));
        int finalSpeed =
Math.abs(Integer.parseInt(tfFinalSpeed.getText()));
        float riseTime = Math.abs(Float.parseFloat(tfRiseTime.getText()));

        isRun = tglRun.isSelected();

        if(startSpeed < 500 && isRun){
            MessageBox.showError("Стартовая скорость должна быть больше 500
об/мин!");

            tglRun.setSelected(false);
            return;
        }

        if((finalSpeed - startSpeed) < 0 && isRun){
            MessageBox.showError("Начальная скорость должна быть меньше
финальной!");

            tglRun.setSelected(false);
            return;
        }

        if(riseTime < 2.0f && isRun){
            MessageBox.showError("Темп нарастания должен быть больше 2 с!");
            tglRun.setSelected(false);
            return;
        }

        Telma.setStartSpeedEngine(startSpeed);
        Telma.setFinalSpeedEngine(finalSpeed);
        Telma.setRiseTime(riseTime);

        if(isRun && isStable){
            tglStable.setSelected(false);
        }
        isActionMode = true;
    }
});

tglStable.selectedProperty().addListener(new ChangeListener<Boolean>() {
    @Override

```

```

        public void changed(ObservableValue<? extends Boolean> observable,
Boolean oldValue, Boolean newValue) {
            isStable = tglStable.isSelected();
            if(isStable && isRun){
                tglRun.setSelected(false);
            }
            isActionMode = true;

            if(isStable){
                refSpeedStable =
Math.abs(Integer.parseInt(tfStable.getText()));
            }else{
                refSpeedStable = 0;
                tfStable.setText("0");
            }
        }
    });

    btnWriteData.setOnAction(new EventHandler<ActionEvent>() {
        @Override
        public void handle(ActionEvent event) {
            Parent root = null;
            try {
                root = FXMLLoader.load(getClass().getResource("/fxml/result-
data.fxml"));

                Stage mainStage = new Stage();
                mainStage.setTitle("DATA");

                Scene scene = new Scene(root);

                mainStage.setScene(scene);
                mainStage.show();

            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    });

    tfGearRatio.setOnKeyPressed(new EventHandler<KeyEvent>() {
        @Override
        public void handle(KeyEvent event) {
            if (event.getCode().equals(KeyCode.ENTER)) {
                float gearR = Float.parseFloat(tfGearRatio.getText());
                Telma.setGearRatio(gearR);
                tfGearRatio.setStyle("-fx-background-color: #7CFC00;");
            }
        }
    });

    tfGearRatio.setOnMouseClicked(new EventHandler<MouseEvent>() {
        @Override
        public void handle(MouseEvent event) {

```

```

        tfGearRatio.setStyle("-fx-background-color: white;");
    });
}

//-----

private void startServer(){
    activateProgrInd(true);

    btnStartServer.setDisable(true);

    serverTCP = new SimpleSocketServer(Const.PORT_SRC, this);
    serverTCP.startServer();
}

private void stopServer(){
    if(serverTCP != null) {
        serverTCP.stopServer();
    }
}

private void closeProgram(){
    stopServer();
    System.exit(0);
}

public void activateProgrInd(boolean isActive){
    if(isActive){
        mainPane.setDisable(true);
        progressIndicator.setVisible(true);
    }else{
        mainPane.setDisable(false);
        progressIndicator.setVisible(false);
    }
}

public void uiLineChartUpdate(Const.TYPE_LINE type){
    ObservableList<XYChart.Data<Number, Number>> list = null;

    switch (type){
        case SPEED: list = serSpeed.getData(); break;
        case REF: list = serRefSpeed.getData(); break;
        case TORQUE: list = serToque.getData(); break;
        case POWER: list = serPower.getData(); break;
        default: break;
    }

    float yValue = 0.0f;

    switch (type){
        case SPEED: yValue = Telma.getEngineSpeedRPM(); break;
        case REF: yValue = Telma.getRefSpeedRPM(); break;
        case TORQUE: yValue = Telma.getEngineTorque(); break;
    }
}

```

```

        case POWER: yValue = Telma.getPowerHP(); break;
        default: break;
    }

    if(list == null) return;

    list.add(new XYChart.Data<>(stepVal, yValue));

    int lowBound = stepVal - 1000;
    int tickUnit = 100;
    if(lowBound < 0){
        lowBound = 0;
        tickUnit = 50;
    }

    xAxis.setLowerBound(lowBound);
    xAxis.setUpperBound(stepVal);

    xAxis.setTickUnit(tickUnit);
}

public void incValStep(){
    ++stepVal;
}

public boolean isActionMode() {
    return isActionMode;
}

public void setActionMode(boolean actionMode) {
    isActionMode = actionMode;
}

public boolean isRun() {
    return isRun;
}

public boolean isStable() {
    return isStable;
}

public int getRefSpeedStable() {
    return refSpeedStable;
}

public Gauge getGgEngineSpeed() {
    return ggEngineSpeed;
}

public Gauge getGgEnginePower() {
    return ggEnginePower;
}

public Gauge getGgEngineTorque() {
    return ggEngineTorque;
}

```

```

    public Gauge getGgRightSpeed() {
        return ggRightSpeed;
    }

    public Gauge getGgLeftSpeed() {
        return ggLeftSpeed;
    }

    public Gauge getGgRightTorque() {
        return ggRightTorque;
    }

    public Gauge getGgLeftTorque() {
        return ggLeftTorque;
    }

    public boolean isWriteData() {
        return isWriteData;
    }
}

```

MessageBox.java

```

package controllers;

import javafx.application.Platform;
import javafx.scene.control.Alert;

public class MessageBox {
    static public void showError(String content){
        Platform.runLater(new Runnable() {
            @Override
            public void run() {
                Alert alert = new Alert(Alert.AlertType.ERROR);
                alert.setTitle("Ошибка!");
                //alert.setHeaderText(header);
                alert.setContentText(content);

                alert.showAndWait();
            }
        });
    }
}

```

Main.java

```

package main;

import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.stage.Stage;

import java.io.File;

```

```

import java.io.IOException;
import java.io.PrintWriter;

public class Main extends Application {

    @Override
    public void start(Stage primaryStage) throws Exception{
        Parent root = FXMLLoader.load(getClass().getResource("/fxml/telma.fxml"));
        primaryStage.setTitle("TELMA");
        primaryStage.setScene(new Scene(root));
        primaryStage.show();
    }

    public static void main(String[] args) {
        try {
            launch(args);
        }
        catch (Exception e) {
            try {
                PrintWriter pw = new PrintWriter(new File("error.txt"));
                e.printStackTrace(pw);
                pw.close();
            }
            catch (IOException e1) {
                e1.printStackTrace();
            }
        }
    }
}

```

Const.java

```

package model;

public final class Const {
    public static final int PORT_SRC = 52041;
    public static final long SLEEP_TIME = 20;
    public static final long LIST_SIZE = 5000;
    public static final int TO_I_RATIO = 10000000;
    public static final int TO_TIME_xK = 100;
    public static final String COMMA_DELIMITER = ",";

    public static enum TYPE_LINE {
        SPEED,
        REF,
        TORQUE,
        POWER
    }
}

```

RowTelma.java

```

package model;

public class RowTelma {
    private float speed;
    private float torque;
}

```

```

private float power;

public RowTelma(float speed, float torque, float power) {
    this.speed = speed;
    this.torque = torque;
    this.power = power;
}

public float getSpeed() {
    return speed;
}

public void setSpeed(int speed) {
    this.speed = speed;
}

public float getTorque() {
    return torque;
}

public void setTorque(float torque) {
    this.torque = torque;
}

public float getPower() {
    return power;
}

public void setPower(float power) {
    this.power = power;
}
}

```

TableTelma.java

```

package model;

import java.util.LinkedList;
import java.util.List;

public class TableTelma {
    private static List<Float> listAveSpeed = new LinkedList<Float>();
    private static List<Float> listLeftSpeed = new LinkedList<>();
    private static List<Float> listRightSpeed = new LinkedList<>();

    private static List<Float> listAveTorque = new LinkedList<Float>();
    private static List<Float> listLeftTorque = new LinkedList<>();
    private static List<Float> listRightTorque = new LinkedList<>();

    private static List<Float> listPower = new LinkedList<Float>();
    private static List<Float> listRefSpeed = new LinkedList<Float>();

    public static void setListLeftSpeed(List<Float> listLeftSpeed) {
        TableTelma.listLeftSpeed = listLeftSpeed;
    }
}

```

```

public static void setListRightSpeed(List<Float> listRightSpeed) {
    TableTelma.listRightSpeed = listRightSpeed;
}

public static void setListLeftTorque(List<Float> listLeftTorque) {
    TableTelma.listLeftTorque = listLeftTorque;
}

public static void setListRightTorque(List<Float> listRightTorque) {
    TableTelma.listRightTorque = listRightTorque;
}

public static void setListRefSpeed(List<Float> listRefSpeed) {
    TableTelma.listRefSpeed = listRefSpeed;
}

//=====
public static void addAveSpeed(){
    if(listAveSpeed == null){
        listAveSpeed = new LinkedList<Float>();
    }

    listAveSpeed.add(Telma.getEngineSpeedRPM());
}

public static void clearAveSpeed(){
    if(listAveSpeed != null){
        listAveSpeed.clear();
    }
}

public static List<Float> getAveSpeedList(){
    return listAveSpeed;
}

//=====

public static List<Float> getLeftSpeedList(){
    return listLeftSpeed;
}

public static void addLeftSpeed(){
    if(listLeftSpeed == null){
        listLeftSpeed = new LinkedList<Float>();
    }

    listLeftSpeed.add(Telma.getLeftSpeedRPM());
}

public static void clearLeftSpeed(){
    if(listLeftSpeed != null){
        listLeftSpeed.clear();
    }
}

//=====

```

```

public static List<Float> getRightSpeedList(){
    return listRightSpeed;
}

public static void addRightSpeed(){
    if(listRightSpeed == null){
        listRightSpeed = new LinkedList<Float>();
    }

    listRightSpeed.add(Telma.getRightSpeedRPM());
}

public static void clearRightSpeed(){
    if(listRightSpeed != null){
        listRightSpeed.clear();
    }
}

//=====================================================

public static void addAveTorque(){
    if(listAveTorque == null){
        listAveTorque = new LinkedList<Float>();
    }
    listAveTorque.add(Telma.getEngineTorque());
}

public static void clearAveTorque(){
    if(listAveTorque != null){
        listAveTorque.clear();
    }
}

public static List<Float> getAveTorqueList(){
    return listAveTorque;
}

//=====================================================

public static void addLeftTorque(){
    if(listLeftTorque == null){
        listLeftTorque = new LinkedList<Float>();
    }
    listLeftTorque.add(Telma.getLeftTorque());
}

public static void clearLeftTorque(){
    if(listLeftTorque != null){
        listLeftTorque.clear();
    }
}

public static List<Float> getLeftTorqueList(){
    return listLeftTorque;
}

```

```
//=====

public static void addRightTorque(){
    if(listRightTorque == null){
        listRightTorque = new LinkedList<Float>();
    }
    listRightTorque.add(Telma.getRightTorque());
}

public static void clearRightTorque(){
    if(listRightTorque != null){
        listRightTorque.clear();
    }
}

public static List<Float> getRightTorqueList(){
    return listRightTorque;
}

//=====

public static void setListAveSpeed(List<Float> list) {
    listAveSpeed = list;
}

public static void setListAveTorque(List<Float> list) {
    listAveTorque = list;
}

public static void setListPower(List<Float> list) {
    listPower = list;
}

//=====

public static void addPower(){
    if(listPower == null){
        listPower = new LinkedList<Float>();
    }
    listPower.add(Telma.getPowerHP());
}

public static void clearPower(){
    if(listPower != null){
        listPower.clear();
    }
}

public static List<Float> getPowerList(){
    return listPower;
}

//=====

public static void addRefSpeed(){
```

```

        if(listRefSpeed == null){
            listRefSpeed = new LinkedList<Float>();
        }

        listRefSpeed.add(Telma.getRefSpeedRPM());
    }

    public static void clearRefSpeed(){
        if(listRefSpeed != null){
            listRefSpeed.clear();
        }
    }

    public static List<Float> getRefSpeedList(){
        return listRefSpeed;
    }
}

```

Telma.java

```

package model;

import java.util.LinkedList;
import java.util.List;

public final class Telma {
    public static final float TO_RPM = 0.6f;
    public static final float TO_TRQ = 10.0f;

    private static float gearRatio = 1.0f;

    private static float refSpeedRPM = 0.0f;
    private static float leftSpeedRPM = 0.0f;
    private static float rightSpeedRPM = 0.0f;
    private static float aveSpeedRPM = 0.0f;

    private static float engineSpeedRPM = 0.0f;

    private static float leftTorque = 0.0f;
    private static float rightTorque = 0.0f;
    private static float aveTorque = 0.0f;
    private static float engineTorque = 0.0f;

    private static float powerHP = 0.0f;

    private static float leftCurrent = 0.0f;
    private static float rightCurrent = 0.0f;

    // for triangle reference
    private static int startSpeedEngine = 0;
    private static int finalSpeedEngine = 0;
    private static float riseTime = 0.0f;

    public static void setSpeed(int refSpeedImp, int leftSpeedImp, int
rightSpeedImp){
        refSpeedRPM = refSpeedImp * TO_RPM * gearRatio;
    }
}

```

```

    leftSpeedRPM = leftSpeedImp * TO_RPM;
    rightSpeedRPM = rightSpeedImp * TO_RPM;
    aveSpeedRPM = (leftSpeedImp + rightSpeedImp) / 2.0f * TO_RPM;
}

public static void setTorque(int torqInLeft, int torqInRight){
    leftTorque = torqInLeft / TO_TRQ;
    rightTorque = torqInRight / TO_TRQ;
    aveTorque = (torqInLeft + torqInRight) / TO_TRQ;
}

public static void setCurrent(int currInLeft, int currInRight){
    leftCurrent = currInLeft / TO_TRQ;
    rightCurrent = currInRight / TO_TRQ;
}

public static void setGearRatio(float gearR){
    if(gearR <= 0) return;

    gearRatio = gearR;
}

public static float getGearRatio() {
    return gearRatio;
}

public static float getLeftSpeedRPM() {
    return leftSpeedRPM;
}

public static float getRightSpeedRPM() {
    return rightSpeedRPM;
}

public static float getEngineSpeedRPM() {
    return aveSpeedRPM * gearRatio;
}

public static float getLeftTorque() {
    return leftTorque;
}

public static float getRightTorque() {
    return rightTorque;
}

public static float getAveTorque() {
    return aveTorque;
}

public static float getEngineTorque() {
    return aveTorque / gearRatio;
}

public static float getRefSpeedRPM(){
    return refSpeedRPM;
}

```

```

    }

    public static float getAveSpeedRPM(){
        return aveSpeedRPM;
    }

    public static float getPowerHP() {
        return (aveTorque * 9.8f) * (aveSpeedRPM * 2.0f * 3.14159f / 60.0f) /
746.0f;
    }

    public static int getStartSpeedEngine() {
        return startSpeedEngine;
    }

    public static void setStartSpeedEngine(int startSpeedEngine) {
        if(startSpeedEngine > 60000){
            Telma.startSpeedEngine = 60000;
            return;
        }
        Telma.startSpeedEngine = startSpeedEngine;
    }

    public static int getFinalSpeedEngine() {
        return finalSpeedEngine;
    }

    public static void setFinalSpeedEngine(int finalSpeedEngine) {
        if(finalSpeedEngine > 60000){
            Telma.finalSpeedEngine = 60000;
            return;
        }
        Telma.finalSpeedEngine = finalSpeedEngine;
    }

    public static float getRiseTime() {
        return riseTime;
    }

    public static void setRiseTime(float riseTime) {
        Telma.riseTime = riseTime;
    }
}

```

TriangleData.java

```

package model;

import java.util.LinkedList;
import java.util.List;

public class TriangleData {
    private List<Float> upSpeedLeft;
    private List<Float> upSpeedRight;
    private List<Float> upSpeedAve;

    private List<Float> downSpeedLeft;

```

```

private List<Float> downSpeedRight;
private List<Float> downSpeedAve;

private List<Float> upTorqueLeft;
private List<Float> upTorqueRight;
private List<Float> upTorqueAve;

private List<Float> downTorqueLeft;
private List<Float> downTorqueRight;
private List<Float> downTorqueAve;

private List<Float> upRefSpeed;
private List<Float> downRefSpeed;

private List<Float> upPower;
private List<Float> downPower;

private List<Float> aveSpeed;
private List<Float> aveTorque;
private List<Float> avePower;

public TriangleData(List<Float> allLeftSpeed, List<Float> allRightSpeed,
List<Float> allAveSpeed,
                    List<Float> allLeftTorque, List<Float> allRightTorque,
List<Float> allAveTorque,
                    List<Float> allPower, List<Float> allRefSpeed){
    int listSize = allRefSpeed.size();

    if(listSize == 0){
        return;
    }

    //find max value and it index
    float maxRefVal = 0;
    int maxRefInd = 0;

    for(int i = 0; i < listSize; ++i){
        float val = allRefSpeed.get(i);
        if(val > maxRefVal){
            maxRefVal = val;
            maxRefInd = i;
        }
    }

    upSpeedLeft = getUpList(allLeftSpeed, maxRefInd, listSize);
    upSpeedRight = getUpList(allRightSpeed, maxRefInd, listSize);
    upSpeedAve = getUpList(allAveSpeed, maxRefInd, listSize);

    downSpeedLeft = getDownList(allLeftSpeed, maxRefInd, listSize);
    downSpeedRight = getDownList(allRightSpeed, maxRefInd, listSize);
    downSpeedAve = getDownList(allAveSpeed, maxRefInd, listSize);

    upTorqueLeft = getUpList(allLeftTorque, maxRefInd, listSize);
    upTorqueRight = getUpList(allRightTorque, maxRefInd, listSize);
    upTorqueAve = getUpList(allAveTorque, maxRefInd, listSize);

```

```

        downTorqueLeft = getDownList(allLeftTorque, maxRefInd, listSize);
        downTorqueRight = getDownList(allRightTorque, maxRefInd, listSize);
        downTorqueAve = getDownList(allAveTorque, maxRefInd, listSize);

        upPower = getUpList(allPower, maxRefInd, listSize);
        downPower = getDownList(allPower, maxRefInd, listSize);

        upRefSpeed = getUpList(allRefSpeed, maxRefInd, listSize);
        downRefSpeed = getDownList(allRefSpeed, maxRefInd, listSize);

        int size = upSpeedAve.size();
        aveSpeed = new LinkedList<>();
        aveTorque = new LinkedList<>();
        avePower = new LinkedList<>();
        for(int i = 0; i < size; ++i){
            aveSpeed.add((upSpeedAve.get(i) + downSpeedAve.get(i)) / 2.0f);
            aveTorque.add((upTorqueAve.get(i) + downTorqueAve.get(i)) / 2.0f);
            avePower.add((upPower.get(i) + downPower.get(i)) / 2.0f);
        }
    }
}

```

RequestHandler.java

```

package threads;

import controllers.MainController;
import javafx.application.Platform;
import javafx.scene.control.Tab;
import model.Const;
import model.TableTelma;
import model.Telma;

import java.io.InputStream;
import java.io.OutputStream;
import java.net.Socket;

public class RequestHandler extends Thread{
    byte[] bufIn = new byte[16];
    byte[] bufOut = new byte[16];
    private Socket socket;
    private MainController mainCtrl;

    RequestHandler( Socket socket, MainController mainCtrl) {
        this.socket = socket;
        this.mainCtrl = mainCtrl;
    }

    @Override
    public void run() {

        if(mainCtrl.isActionMode()){
            boolean isRun = mainCtrl.isRun();
            boolean isStable = mainCtrl.isStable();
            int refSpeedStable = mainCtrl.getRefSpeedStable();

            setUpOutBuffer(isRun, isStable, refSpeedStable);
        }
    }
}

```

```

        mainCtrl.setActionMode(false);
    }

    try {

        InputStream input = socket.getInputStream();
        int is = input.read(bufIn);

        readData();

        OutputStream out = socket.getOutputStream();
        out.write(bufOut);
        out.flush();

        // Close our connection
        out.close();
        input.close();
        socket.close();

        clearData();

    }
    catch( Exception e ) {
        e.printStackTrace();
    }
}

private void readData(){
    int refSpeed = 0;
    int speedLeft = 0;
    int speedRight = 0;

    int torqLeft = 0;
    int torqRight = 0;

    int currLeft = 0;
    int currRight = 0;

    // speed
    refSpeed = ((bufIn[0] & 0xff) << 8) | (bufIn[1] & 0xff);
    speedLeft = ((bufIn[2] & 0xff) << 8) | (bufIn[3] & 0xff);
    speedRight = ((bufIn[4] & 0xff) << 8) | (bufIn[5] & 0xff);
    Telma.setSpeed(refSpeed, speedLeft, speedRight);

    // torque
    torqLeft = ((bufIn[6] & 0xff) << 8) | (bufIn[7] & 0xff);
    torqRight = ((bufIn[8] & 0xff) << 8) | (bufIn[9] & 0xff);
    Telma.setTorque(torqLeft, torqRight);

    // current
    currLeft = ((bufIn[10] & 0xff) << 8) | (bufIn[11] & 0xff);
    currRight = ((bufIn[12] & 0xff) << 8) | (bufIn[13] & 0xff);
    Telma.setCurrent(currLeft, currRight);

    TableTelma.addRefSpeed();
}

```

```

TableTelma.addAveSpeed();

TableTelma.addLeftSpeed();
TableTelma.addRightSpeed();

TableTelma.addAveTorque();
TableTelma.addLeftTorque();
TableTelma.addRightTorque();

TableTelma.addPower();

Platform.runLater(() -> //switches to GUI Thread
{
    uiSpeedUpdate();
    uiTorqueUpdate();
    uiPowerUpdate();

    if(mainCtrl.isWriteData()) {
        mainCtrl.incValStep();
        mainCtrl.uiLineChartUpdate(Const.TYPE_LINE.SPEED);
        mainCtrl.uiLineChartUpdate(Const.TYPE_LINE.TORQUE);
        mainCtrl.uiLineChartUpdate(Const.TYPE_LINE.POWER);
        mainCtrl.uiLineChartUpdate(Const.TYPE_LINE.REF);
    }
});

}

private void setUpOutBuffer(boolean isRun, boolean isStable, int
refSpeedStable){

    // send to the client work MODE
    if(isRun){
        bufOut[0] = 'r';
    }else if(isStable){
        bufOut[0] = 's';
    }else{
        bufOut[0] = 'i';
    }
}

// send to the client gear ratio
int iRatio = (int)(Math.round(Telma.getGearRatio() * Const.TO_I_RATIO));
bufOut[1] = (byte) (iRatio >> 24);
bufOut[2] = (byte) ((iRatio & 0x00FF0000) >> 16);
bufOut[3] = (byte) ((iRatio & 0x0000FF00) >> 8);
bufOut[4] = (byte) (iRatio & 0x000000FF);

// send to the client reference speed in impulses
int iSpeed = (int) (Math.round(refSpeedStable / Telma.getGearRatio() /
Telma.TO_RPM));
if(iSpeed > 10000){
    iSpeed = 0;
}

bufOut[5] = (byte) ((iSpeed & 0x0000FF00) >> 8);
bufOut[6] = (byte) (iSpeed & 0x000000FF);

```

```

        // send start speed engine
        int startSpeed = Telma.getStartSpeedEngine();
        bufOut[7] = (byte) ((startSpeed & 0x0000FF00) >> 8);
        bufOut[8] = (byte) (startSpeed & 0x000000FF);

        // send final speed engine
        int finalSpeed = Telma.getFinalSpeedEngine();
        bufOut[9] = (byte) ((finalSpeed & 0x0000FF00) >> 8);
        bufOut[10] = (byte) (finalSpeed & 0x000000FF);

        // send rise time in /100 seconds
        int riseTimeI = (int) Math.round(Telma.getRiseTime() * Const.TO_TIME_xK);
        bufOut[11] = (byte) ((riseTimeI & 0x0000FF00) >> 8);
        bufOut[12] = (byte) (riseTimeI & 0x000000FF);

    }

    private void uiSpeedUpdate(){
        Platform.runLater(() -> //switches to GUI Thread
        {
            mainCtrl.getGgLeftSpeed().setValue(Telma.getLeftSpeedRPM());
            mainCtrl.getGgRightSpeed().setValue(Telma.getRightSpeedRPM());
            mainCtrl.getGgEngineSpeed().setValue(Telma.getEngineSpeedRPM());
        });
    }

    private void uiTorqueUpdate(){
        Platform.runLater(() -> //switches to GUI Thread
        {
            mainCtrl.getGgLeftTorque().setValue(Telma.getLeftTorque());
            mainCtrl.getGgRightTorque().setValue(Telma.getRightTorque());
            mainCtrl.getGgEngineTorque().setValue(Telma.getEngineTorque());
        });
    }

    private void uiPowerUpdate(){
        mainCtrl.getGgEnginePower().setValue(Telma.getPowerHP());
    }

    private void clearData(){
        for(int i = 0; i < bufIn.length; ++i){
            bufIn[i] = 0;
        }

        for(int i = 0; i < bufOut.length; ++i){
            bufOut[i] = 0;
        }
    }
}

```

SimpleSocketServer.java

```

package threads;

import controllers.MainController;
import javafx.application.Platform;

```

```

import java.io.IOException;
import java.net.InetAddress;
import java.net.ServerSocket;
import java.net.Socket;
import java.net.UnknownHostException;

public class SimpleSocketServer extends Thread {
    private ServerSocket serverSocket;
    private int port;
    private boolean running = false;
    private MainController mainController;

    public SimpleSocketServer( int port, MainController mainController ) {
        this.port = port;
        this.mainController = mainController;

        try {
            System.out.println("ip address of server is - " +
InetAddress.getLocalHost());
        } catch (UnknownHostException e) {
            e.printStackTrace();
        }
    }

    public void startServer() {
        try {

            serverSocket = new ServerSocket( port );

            this.start();

        } catch (IOException e) {
            e.printStackTrace();
        }
    }

    public void stopServer() {
        running = false;
        this.interrupt();
    }

    @Override
    public void run() {
        running = true;

        boolean flag = true;
        while( running ) {

            try {

                // Call accept() to receive the next connection
                Socket socket = serverSocket.accept();

                // Pass the socket to the RequestHandler thread for processing
                RequestHandler requestHandler = new RequestHandler( socket,
mainController );

```

```

        requestHandler.start();

    } catch (IOException e) {
        e.printStackTrace();
    }

    if(flag){
        flag = false;
        Platform.runLater(() -> //switches to GUI Thread
        {
            mainController.activateProgrInd(false);
        });
    }
}
}
}

```