



MÁRTON LÓRINC FEHÉR ÁRON

IRÁNYÍTÁSTECHNIKA

LABORATÓRIUMI ÚTMUTATÓ

MÁRTON LŐRINC-FEHÉR ÁRON

IRÁNYÍTÁSTECHNIKA
LABORATÓRIUMI ÚTMUTATÓ



SAPIENTIA ERDÉLYI MAGYAR TUDOMÁNYEGYETEM
MAROSVÁSÁRHELYI KAR
VILLAMOSMÉRNÖKI TANSZÉK

MÁRTON LŐRINC–FEHÉR ÁRON

IRÁNYÍTÁSTECHNIKA
LABORATÓRIUMI ÚTMUTATÓ

| Scientia Kiadó |
| Kolozsvár · 2019 |

Lektor

Gincsiné Szádeczky-Kardoss Emese (Budapest)

Borítóterv

Tipotéka Kft.

Első magyar nyelvű kiadás: 2019

©Scientia, 2019

Minden jog fenntartva, beleértve a sokszorosítás, a nyilvános előadás, a rádió- és televízióadás, valamint a fordítás jogát, az egyes fejezeteket illetően is.

ISBN 978-606-975-023-0

TARTALOM

1. Bevezető	13
2. Grafikus folyamatmonitorizálás	18
3. Hőmérséklet-szabályozás kétállású szabályozóval	24
4. Proporcionális hőmérséklet-szabályozás	32
5. Hőmérsékletprofil-követés PI-szabályozóval	38
6. Ipari kemencék PID-szabályozása	45
7. Önhangoló PID-szabályozás	51
8. Vegyioldat-koncentráció szabályozása	57
9. Egyenáramú motor modellezése	65
10. Egyenáramú motor kaszkádszabályozása	72
11. PI- és PD-szabályozók paramétereinek hangolása	80
12. Előreccatolást alkalmazó szabályozás	89
13. Lejtőn guruló golyó nemlineáris irányítása	97
Függelék	106
Szakirodalom	109
Kivonat	111

Rezumat	113
Abstract	115
A szerzőkről	117

CONTENTS

1. Introduction	13
2. Graphical process modeling	18
3. Temperature control using ON/OFF controllers	24
4. Proportional temperature control	32
5. Reference profile tracking using PI controllers	38
6. PID control of industrial furnaces	45
7. Self-tuning PID controllers	51
8. Chemical concentration control	57
9. Modeling of direct current motors	65
10. Cascade control of direct current motors	72
11. PI and PD controller parameter tuning	80
12. Feed-forward control	89
13. Nonlinear control of the ball and beam system	97
Annexes	106
References	109
Abstract	115

About the authors

117

CUPRINS

1. Introducere	13
2. Monitorizarea grafică a proceselor	18
3. Controlul temperaturii cu reglatoare bipoziționale	24
4. Controlul temperaturii cu regulator proporțional	32
5. Controlul temperaturii cu referință variabilă folosind regulator PI	38
6. Comanda PID a cuptoarelor industriale	45
7. Reglatoare PID auto-acordabile	51
8. Reglarea concentrației unei soluții chimice	57
9. Modelarea motorului de curent continuu	65
10. Reglarea în cascadă a motorului de curent continuu	72
11. Acordarea parametrilor pentru reglatoare PI și PD	80
12. Reglatoare cu elemente de reacție directă (feed-forward)	89
13. Reglarea nelineară a sistemului bilă-rampă	97
Anexe	106
Bibliografie	109

Rezumat	113
Despre autori	117

BEVEZETŐ

1.1. Elméleti bevezető

Az irányítástechnika: A korszerű termelési és szolgáltatási folyamatokban elengedhetetlen a tervszerű működtetés biztosítása. Az ismétlődő tevékenységet igénylő folyamatok esetén, mint például a sorozatgyártás, vagy olyan folyamatok esetében, amelyek az ember számára veszélyes környezetben valósulnak meg, célszerű az embert felszabadítani a termelési tevékenység végrehajtása alól. Korszerű termelési folyamatok esetében az ember feladata a termelés elindítása, felügyelete; a folyamat minimális emberi beavatkozással kell hogy működjön. Ennek eléréséhez a folyamatainkat önműködővé kell tennünk.

Az irányítástechnika az önműködő rendszerek törvényszerűségeivel, tervezésével, megvalósításával foglalkozó mérnöki tudományág. Napjaink korszerű irányítástechnikai módszerei feltételezik a számítógépes támogatást, az irányítást megvalósító eljárásokat számítógépes program részeként implementáljuk. A jelen laboratóriumi útmutató is a számítógépes irányításra fekteti a hangsúlyt.

Laboratóriumi gyakorlatok összefoglalója: A laboratóriumi útmutató célja bevezetőt nyújtani a klasszikus irányítási eljárások tervezésébe és megvalósításába.

Az első négy laboratóriumi gyakorlat a hőmérséklet-szabályozási feladattal foglalkozik. A diákoknak meg kell valósítaniuk egy egybemenetű, egykimenetű szabályozási kör irányítását biztosító számítógépes programot, amely magába foglalja a folyamat felügyeletét biztosító grafikus kijelzést, a folyamat kétállású szabályozását, a folyamat proporcionális szabályozását, valamint a profilkövetést megvalósító szabályozást.

Az ötödik és hatodik laboratóriumi gyakorlat a holtidős folyamatok irányításának tervezésével foglalkozik, ipari kemencékre bemutatva, illetve az önhangoló szabályozás módszerével.

A referenciamodell-alapú tervezést egy oldat koncentrációsabályozási feladatán keresztül mutatja be a hetedik laboratóriumi gyakorlat (vegyi folyamat szabályozása).

Az irányításhoz elengedhetetlen az irányított folyamat modellezése. A nyolcadik laboratóriumi gyakorlat az egyenáramú motor modellezésével foglalkozik. A kilencedik laboratóriumi gyakorlat az egyenáramú motor kaszkád szabályozását mutatja be.

A tizedik laboratóriumi gyakorlat a Proporcionális-Integráló-Deriváló (PID) szabályozók paramétereinek hangolásának hatását mutatja be a szabályozási minőségi jellemzőkre. A tizenegyedik laboratóriumi gyakorlat bemutatja az előreccsatolást alkalmazó szabályozókat, összehasonlítva őket az integrátor tagot tartalmazó szabályozókkal.

A tizenkettedik laboratóriumi gyakorlat betekintőt nyújt a nemlineáris irányítások tervezésébe a lejtőn guruló golyó példáján keresztül.

A feladatok megoldásához programozói ismeretek (C, C#, Matlab/Simulink), rendszerelméleti alapfogalmak, elektronikai, mérés-technikai alapismeretek szükségesek.

Vezérlés vs. szabályozás: Számos irányítási feladat esetében a cél a folyamat kimenetének egy előírt értékre eljuttatása és annak biztosítása, hogy a kimenet az előírt értéken maradjon (értéktartó szabályozás). Például az, hogy egy robot végberendezését adott térbeli pontba eljuttassuk, vagy egy ipari kemence hőmérsékletét adott értékre beállítsuk és ezen az értéken tartsuk.

Amennyiben feltételezzük, hogy a folyamat viselkedését tökéletesen ismerjük, meghatározhatjuk a folyamat bemenetét úgy, hogy a kimenet eljusson az előírt értékre és azt tartsa. Ebben az esetben vezérlésről beszélünk: A folyamat bemenetét előre meghatározzuk, az irányítás elindítása és futása alatt az előre megtervezett bemenetet alkalmazzuk a folyamatra.

Tipikus vezérlési példa az önműködő öntözőberendezés: minden napszakban egyszer elindul az öntözés és adott ideig locsolja a kijelölt területet. Itt a folyamat bemenete az az időtartam, amíg a locsolás aktív.

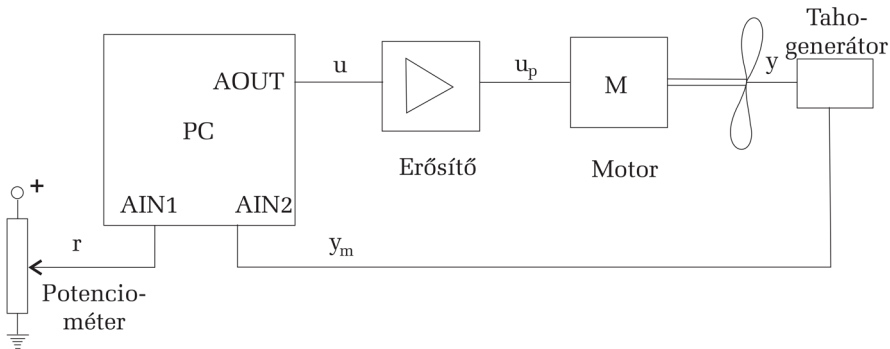
Ez a megoldás nem veszi figyelembe a folyamatra ható zavarokat, például azt, hogy egy esetleges eső, illetve az erős napsütés is számottevően befolyásolja a talaj nedvességtartalmát. Jobb módszer lenne, ha mérnénk a talaj nedvességtartalmát, és a mérés alapján számítanánk ki, akár működés közben az öntözés intenzitását. Abban az esetben, ha a folyamat kimenetét mérjük, és ennek alapján határozzuk meg a bemenetet, szabályozásról beszélünk. A szabályozás figyelmeztet az eltérést az előírt kimenet és a mért kimenet között, és ennek alapján végzi el a beavatkozást a folyamat bemenetén.

1.2. Példák irányítási feladatokra

Fordulatszám-szabályozás: Az 1.1. ábrán egy ventilátor fordulatszámának szabályozási köre látható. A ventilátort villamos gép hajtja, amelyet egy teljesítményerősítőn keresztül irányíthatunk. A teljesítményerősítőt az jellemzi, hogy a bemeneti feszültség függvényében lineárisan változik a motorra adott teljesítmény. A ventilátor fordulatszámát tahogenerátorral mérjük. A fordulatszámot egy betáplált potenciométerrel írjuk elő. A rendszert egy analóg be- és kimenetekkel felszerelt adatbegyűjtő kártyával rendelkező számítógéppel irányítjuk. Határozzuk meg a szabályozási hurokban megjelenő jeleket és a hurok elemeit.

A szabályozókör elemei:

- Folyamat: a ventilátor; a bemeneti jel a ventilátorra alkalmazott forgatónyomaték, és a kimeneti jel a ventilátor szögsebessége
- Érzékelő: tahogenerátor
- Beavatkozó: a teljesítményerősítő + motor
- Szabályozó: a számítógépen futó irányítószoftver (NEM a számítógép maga)



1.1. ábra. Fordulatszám-szabályozási kör

A szabályozókör jelei:

- u_p – a folyamat bemenete – a motor által kifejtett nyomaték;
- y – a folyamat kimenete – ventilátor fordulatszáma;
- y_m – mért kimenet – a tahogenerátor által létrehozott, számítógépbe beolvasott analóg feszültség;
- r – a betáplált potenciométerről leolvasott, számítógépbe beolvasott feszültség;

e – szabályozási hiba – a jel, a számítógépes irányítószoftverben számoljuk;

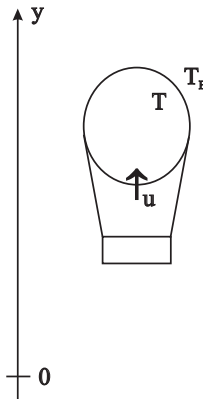
u – az irányítószoftver által kiszámolt, a számítógép analóg kimenetén kiadott jel, ami a teljesítményerősítő bemenete;

d – zavaró jel – mérési zajok, a ventilátor csapágyaiban fellépő nem viszkózus súrlódási komponensek stb.

Példa folyamatok modellezésére: A szabályozási hurokban az alkalmazott szabályozási eljárás kiválasztásához, illetve a szabályozóparaméterek meghatározásához elengedhetetlen az irányított folyamat ismerete. Szükséges azt ismerni, hogy a folyamat kimenete hogyan reagál egy adott bemenetre. A folyamat be- és kimenete közötti összefüggést (matematikai modellt) fizikai, kémiai törvények alapján határozhatjuk meg.

Hőlégballon-modell irányítástechnikai alkalmazásokhoz: Legyen egy hőlégballon, amelynek

- tömege: m ;
- a ballon hőáteresztési állandója: λ ;
- a ballon hőtárolási kapacitása: C ;
- a gázégető modellje: $E = k_u u$, ahol u a gázcsap nyitási szöge, k_u pedig a beavatkozó erősítése;
- pozíciója a függőleges tengelyen: y .



1.2. ábra. Hőlégballon-vázlat modellezéshez

Az irányítási probléma: határozzuk meg az u nyitási szöget úgy, hogy a hőlégballon y_{ref} előírt (referencia) magasságba emelkedjen.

A feladat megoldásához keressük az összefüggést a hőlégballon magassága (y – szabályozott kimenet), valamint a gázcsap nyitási szöge (u – beavatkozójel) között.

A hőlégballont a felhajtóerő emeli fel, ami arányos a ballon belsejében levő levegő hőmérséklete (T), valamint a környezeti levegő hőmérséklete (T_E) különbségének arányával:

$$F_b = k_b(T - T_E). \quad (1.1)$$

A modellezéshez használt fizikai egyenletek:

1. Hőegyensúly: a gázégető által leadott hő egyenlő a ballon által tárolt hő, valamint a ballomból a környezetbe kisugárzott hő összegével:

$$C\dot{T} + \lambda(T - T_E) = k_u u. \quad (1.2)$$

2. Newton második törvénye:

$$m\ddot{y} = k_b(T - T_E) - mg. \quad (1.3)$$

Feltételezzük, hogy T_E konstans.

$$\begin{aligned} m\ddot{y} &= k_b\dot{T} \\ \frac{Cm}{k_b}\ddot{y} + \frac{\lambda}{k_b}\dot{y} &= k_u u - \frac{\lambda}{k_b}g \\ Cm\ddot{y} + \lambda m\dot{y} &= k_u k_b u - \lambda g. \end{aligned} \quad (1.4)$$

A hőlégballon-modell átviteli függvénye, alkalmazva a Laplace-transzformáltat:

$$\begin{aligned} Cms^3Y(s) + \lambda ms^2Y(s) &= k_u k_b(U(s) - D(s)) \\ H(s) = \frac{Y(s)}{U(s) - D(s)} &= \frac{k_u k_b}{Cms^3 + \lambda ms^2}. \end{aligned} \quad (1.5)$$

A modellben a $D = \frac{\lambda g}{k_u k_b}$ a gravitációs hatás miatt megjelenő konstans zavarbemenet.

Az egybemenetű, egykimenetű rendszerekre kidolgozott szabályozótervezési módszerek jelentős része a folyamatok átviteli függvényekkel megadott modelljén alapulnak.

GRAFIKUS FOLYAMATMONITORIZÁLÁS

2.1. A gyakorlat célja

Hőmérséklet-szabályozási feladatot megvalósító számítógépes program fejlesztése, lassú folyamatok grafikus monitorizálásának megvalósítása.

2.2. Elméleti bevezető

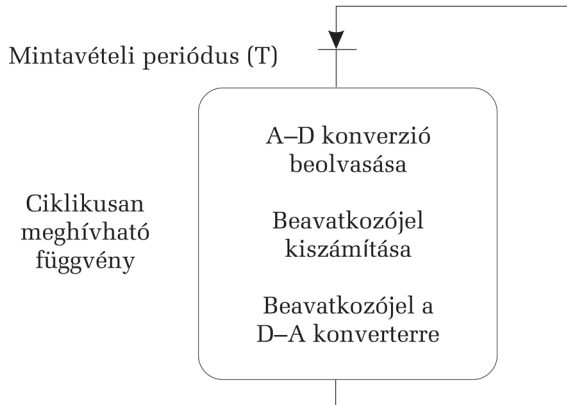
Az irányítási algoritmusok megvalósításának legelterjedtebb módja a szoftveres implementálás. Az irányítási algoritmust megvalósító függvények, eljárások általában egy nagyobb, ipari folyamatok felügyeletére, irányítására kialakított szoftver részei. Az irányítási algoritmus a folyamat mért kimenete és az előírt kimenet (alapjel) alapján számítja ki a beavatkozójelet. A folyamat kimenetét analóg-digitális átalakítón keresztül olvassuk be. Az alapjelet ugyancsak analóg-digitális átalakítón keresztül olvashatjuk be vagy megadhatjuk számítógépes periférián keresztül is (például billentyűzet segítségével). A második esetben nincs szükség analóg-digitális átalakításra az alapjel esetében. A kiszámolt beavatkozójelet digitális-analóg átalakítón keresztül küldjük a beavatkozó bemenetére.

A mintavételi periódus: A mintavételes megvalósításnál szükségünk van egy ciklikusan (konstans periódusonként) meghívható függvényre. A konstans periódust *mintavételi periódusnak* nevezzük. Az ipari szoftver főszála ciklikusan megszakítódik, és végrehajtódik az irányítási algoritmust megvalósító eljárás. Az eljárás főbb részei a következők:

- a folyamat bemenetének beolvasása analóg-digitális átalakítóról;
- a beavatkozójel kiszámítása az alapjel és a folyamat bemenetének függvényében;
- a beavatkozójel kiküldése a digitális-analóg átalakítón keresztül.

A mintavételi periódus megválasztása kritikus a szabályozás helyes működésének biztosításához. Az irányítási feladatokban *fix, konstans mintavételi periódust* alkalmazunk. Mivel a mintavételi periódus gyakoriságával

olvassuk be a folyamat kimenetét, ami alapján az irányítási algoritmus kiszámítja a vezérlőjelet, a periódust úgy kell megválasztani, hogy a beolvasott mintavételezett jel tartalmazza a folytonos jel jellegzetességeit. Így a periódus megválasztása minden esetben alkalmazásfüggő. Gyors válaszú rendszerek esetében kis mintavételi periódust választunk, lassú válaszú rendszerek esetében nagyobb mintavételi periódus alkalmazható.



2.1. ábra. Mintavételes szabályozás megvalósítása analóg be- és kimeneti jel esetén

Az irányítás mellett fellépő feladatok: Az irányítást megvalósító számítógépes programnak a beavatkozójel kiszámítása mellett több feladatot is el kell végeznie. A feladatokat két csoportba sorolhatjuk:

– *Nagy prioritású feladatok:* Tipikusan biztonsági problémák, amelyek prioritása nagyobb a beavatkozójel számításánál, megszakíthatják a vezérlőjel számolását. Például fordulatszám-szabályozás esetén, ha a beavatkozó szervként használt motor károsodik, a motor védelme ezt egy digitális bemeneten keresztül jelzi az irányítást megvalósító számítógépes programnak. Ebben az esetben az irányítási algoritmus a beavatkozójel számítását nem kell hogy elvégezze, a futása felfüggesztődik.

– *Kis prioritású feladatok:* Ezek a feladatok ugyancsak párhuzamosan futnak az irányítást megvalósító, ciklikusan meghívható függvénnyel, de nem szakíthatják meg a beavatkozójel számítását. Ilyen feladatok például a megjelenítés és a naplózás.

– *Megjelenítés (monitorizálás)* – a beolvasott, kiküldött, számított jelek numerikus, grafikus kijelzése.

– Naplózás – a rendszer viselkedésének (beolvasott, kiküldött, számított jelek) archiválása adattárolókra az utólagos feldolgozás érdekében (irányítás kiértékelése, rendszerhibák megtalálása). A hasznos adatok elmentése mellett általában tároljuk a mentés idejét is (óra, dátum), ezért az eljárást naplózásnak is nevezzük.

2.3. A mérés menete

A ciklikusan meghívható függvény megvalósítása Windows operációs rendszerben: A laboratóriumi gyakorlat során a *Temperature_Control* szoftverprojektben dolgozunk. A program vázát a függelékben található F2. ábra mutatja.

A ciklikus meghívható függvények implementálásához a Windows szoftverek fejlesztésére alkalmas programozási környezetek egy külön eseményt biztosítanak. Ez az úgynevezett *Timer* esemény. A *Timer* eseményt a program futása során engedélyezni vagy letiltani lehet, illetve módosítható a periódusa.

C# környezetben az időzítőt a grafikus felületen rendeljük a programunkhoz, amely létrehoz egy *Timer Tick* metódust. Ezen metódus tulajdonságainak beállítását, illetve az elindítást a grafikus felület betöltésekor tesszük meg a *Temperature_Control_Form_Load* metódusban:

```
controlTimer.Interval = 1000; // millisecond  
controlTimer.Start();
```

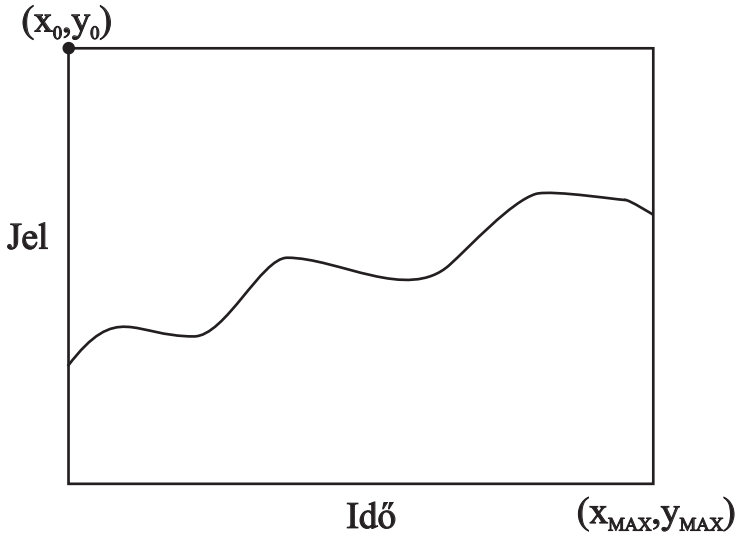
Ebben az esetben az időzítő neve *ControlTimer*. Az *Interval* tulajdonsággal adjuk meg a periódust ezredmásodpercben, a *Start()* metódussal indítjuk el az időzítőt.

Az időzítő megállítását a *Stop()* metódussal tehetjük, amikor a grafikus felületet eltüntettük (a *Temperature_Control_FormClosed* metódusban).

Feladat: Készítsünk egy C# alapú programot, ciklikusan (1 másodperc periódussal) jelenít meg véletlenszerűen generált pontokat egy grafikonon.

- Az értékgenerálás akkor kezdődjön el, amikor a grafikus kezelőfelület megjelenik.
- A száz legfrissebb értéket tárolja el egy lista (ami ciklikusan frissüljön).
- A lista tartalma grafikusan jelenjen meg a kezelőfelületen – alkalmazva a beépített Windows grafikon kezelő metódusait.

Folyamatok grafikus monitorizálása: Ahhoz, hogy a folyamat aktuális és múltbeli állapotáról információnk legyen, a folyamat mért jeleinek grafikus megjelenítését alkalmazhatjuk. Ez egy időfüggvény grafikus megjelenítése, amelyben az utolsó N mintavételi periódusban begyűjtött mintákat ábrázoljuk grafikusan. Ahhoz, hogy mindig a legutolsó mérések eredményét lássuk a képernyőn, a grafikont folyamatosan el kell tolni.



2.2. ábra. Mérési adatok grafikus kijelzése

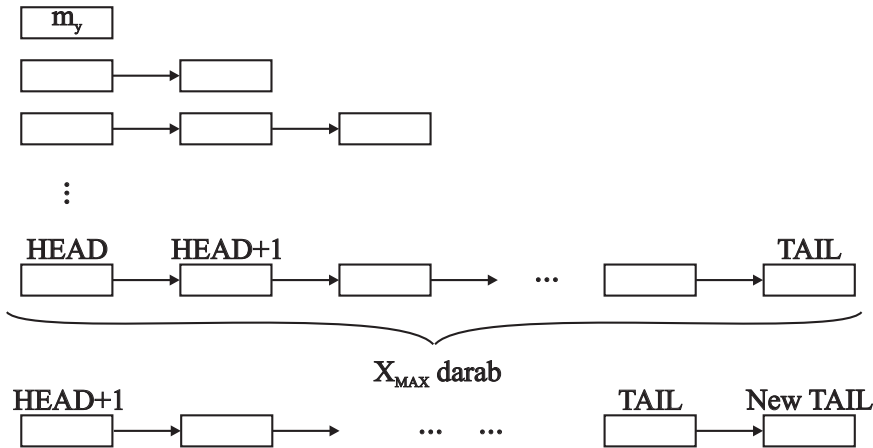
A kijelzéshez feltételezzük, hogy a képernyőn X_{MAX} , Y_{MAX} pixelméretű terület (grafikus ablak) áll rendelkezésünkre a grafikus kijelzéshez. Az egyszerűség kedvéért a grafikus ablak bal felső sarkának a pozíciója $x_0 = 0$, $y_0 = 0$.

Grafikus kalibrálás: Feltételezzük, hogy a méréseket egy m változóba kapjuk. A mérések tartománya: $m_{MIN} \dots m_{MAX}$. Ezt a tartományt kell hogy transzformáljuk a $0 \dots Y_{MAX}$ grafikus tartományba. Figyelembe kell venni, hogy a grafikus kijelzőkön az Y függőleges tengely iránya fentről lefele néz. Így, ha m_y -vel jelöljük a kijelzendő értéket:

$$m_y = \left[Y_{MAX} - \frac{Y_{MAX}}{m_{MAX} - m_{MIN}} \cdot m \right]_{INT} \quad (2.1)$$

INT a kapott érték egész részét jelenti.

Az adatok tárolása a kijelzéshez: Mivel egyszerre több adatot kell ábrázolni, a grafikus kalibráció után a méréseket érdemes egy listában tárolni. A lista végére mindig az új mérést tesszük. Ha a lista hossza elérte az X_{MAX} értéket, a lista fejét levágjuk. Így a listában mindig a legutolsó X_{MAX} mérés lesz.



2.3. ábra. Adatok tárolása grafikus kijelzéshez

Az aktuális listában található mérési adatok ábrázolásához *Series* változókat alkalmazunk. A listák számozása nulláról indul. A listában található adatszámot a *.Points.Count* változón keresztül kérhetjük le. Adatot *.Points.RemoveAt(i)* módszerrel távolítunk el, ahol i az adat sorszáma. Új adatot a *.Points.Add(data)* módszerrel csatolhatunk a lista végére.

A grafikon megjelenítése előtt az olvashatóság érdekében fel kell konfigurálni a grafikonkezelő felületet: *Sample_chart.Titles.Add("Title")* módszer címet ad a grafikonnak;

Sample_chart.ChartAreas[i].AxisX.Title = "Time (s)" az i -ik X tengelyt nevezi el;

Sample_chart.ChartAreas[i].AxisX.Minimum és *.Maximum* statikus méretet ad az X tengelynek – Ha az adataink kiesnek az adott intervallumból, a grafikus felület levágja. Hasonlóan érvényesek a műveletek az Y tengelyre is.

A grafikon adatlistáját a *Sample_chart.Series.Clear()* utasítással töröljük. Ha egy listát akarunk a grafikonhoz rendelni, akkor a *randomSeries = Sample_chart.Series.Add("RNG")* utasítással tehetjük meg

– ebben az esetben a *Sample_chart* nevű grafikonhoz csatolja a *randomSeries* listát, ami a jelmagyarázatban az “RNG” névvel fog megjelenni. Az adathalmaz színét a *randomSeries.Color* tulajdonságon keresztül adjuk meg. A típusát pedig a *.ChartType* tulajdonsággal állítjuk be (ez lehet *Pie*, *Histogram*, *Line* stb., de a jelen feladatnak a *FastLine* vagy a *Spline* típus felel meg).

Ahhoz, hogy a lista elemeinek frissítése a grafikonon is megjelenjen, a *Sample_chart.ResetAutoValue()* metódust kell meghívni.

Feladat: A *Temperature_Control_Form* tervben oldja meg egy periodikusan generált véletlenszerű számsorozat grafikus kalibrálását, listába mentését és ennek ábrázolását (maximum 100 értékkel).

A megoldását a *controlTimer_Tick* metódusban implementáljuk az alábbi módon:

- Generálunk két véletlenszerű számot a *rand.Next()* függvénnyel – egyik jel [0, 100], másik pedig [-100, 100] intervallumból vegye az értékeket.
- Létrehozunk egy-egy listát, megvizsgáljuk, hogy az elemek száma ne haladja meg a maximumot, majd hozzáadjuk az új generált számokat.
- A listák elemeit egy-egy külön grafikonon ábrázoljuk, ezen grafikonokat a kezelőfelület betöltésekor konfiguráljuk.

2.4. Kérdések és feladatok

1. Vizsgálja meg, hogyan kell a *Timer*-szerű függvényt alkalmazni Linux operációs rendszer alatt C++ programozási nyelven.
2. Oldja meg a grafikus kijelzést úgy, hogy a mérési adatok tárolásához nem listát, hanem tömböt alkalmaz.
3. Módosítsa a programot úgy, hogy a listában nem egy, hanem több mérési adatot tárol, és mindegyiket kiteszi ugyanarra a grafikus ablakra.

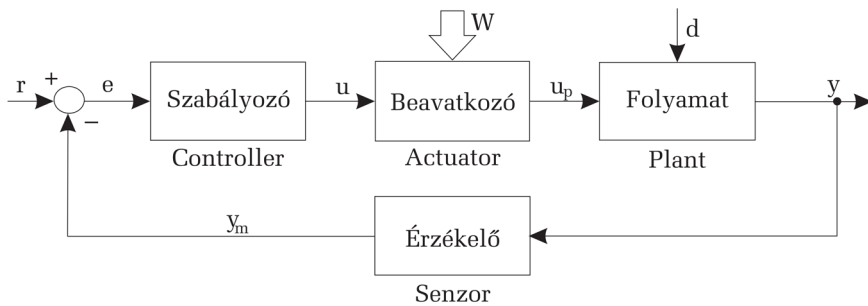
HŐMÉRSÉKLET-SZABÁLYOZÁS KÉTÁLLÁSÚ SZABÁLYOZÓVAL

3.1. A gyakorlat célja

A hőmérséklet-szabályozási hurok elemeinek megismerése, a folyamat és számítógép csatolásának (érzékelés, beavatkozás) megvalósítása. Kétállású (ON-OFF) szabályozó megvalósítása és tanulmányozása.

3.2. Elméleti bevezető

A szabályozási hurok: A szabályozási hurok a folyamatból és annak irányítását biztosító technikai eszközökből álló zárt rendszer (lásd 3.1. ábra).



3.1. ábra. A szabályozási hurok felépítése

A hurokban megjelenő jelek elnevezései:

- u_p – a folyamat bemenete – *plant input* (módosított jellemző)
- y – a folyamat kimenete – *plant output* (szabályozott jellemző)
- y_m – mért kimenet – *measured output* (ellenőrzőjel)
- r – előírt érték – *reference signal* (alapjel)
- e – szabályozási hiba – *control error* (rendelkezőjel)

u – beavatkozójel – *control signal*

d – zavarójel – *disturbance*

w – a beavatkozó által felhasznált külső energiaforrás

A szabályozási hurok elemeinek főbb jellemzői az irányítási algoritmus szempontjából:

Az irányított folyamat: Bemenete tipikusan nagy energiájú jel, amelyet a szabályozási hurkon belül úgy kell meghatározni, hogy a folyamat kimenete kövesse az előírt értéket.

Érzékelő: a folyamat kimenetét alakítja az irányítási algoritmus által feldolgozható értékre. A bemenetük lehet nem elektromos jel (például hőmérséklet, fordulatszám), a kimenete tipikusan elektromos jel. Célszerű, ha az érzékelő kimenete egyezményesített, szabványos tartományú jel. Ilyen tartományok például a 0-5 V, 0-10 V, 0-20 mA, 4-20 mA. Természetesen nem minden jelet lehet vagy érdemes egyezményesített tartományba átalakítani.

Beavatkozó: A beavatkozó szerepe, hogy a bemenetén megjelenő kis energiájú jelet (amit a szabályozó határoz meg) a folyamat bemenetének megfelelő szintre átalakítsa, tehát tipikusan kis energiájú jelből nagy energiájú jelet kell hogy létrehozson. Jellemző a beavatkozóra, hogy ezen az elemen keresztül áramlik be az energia a szabályozási körbe. Hasznos, ha a bemenete egyezményesített jel. Elvárható, hogy kisebb időállandókkal rendelkezzen, mint az irányított folyamat.

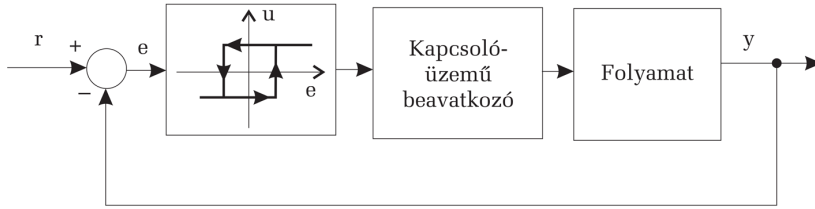
Az irányítási algoritmus feladata, hogy az érzékelő által mért kimenet és az előírt érték függvényében meghatározza a beavatkozó bemenetét úgy, hogy a folyamat kimenete kövesse az alapjelet. Az esetek többségében először képezzük az előírt érték és a mért kimenet különbségét, vagyis meghatározzuk a szabályozási hibát:

$$e = r - y_m. \quad (3.1)$$

A szabályozó bemenetét így a szabályozási hiba képezi. A szabályozónak mind a bemenete, mind a kimenete kis energiájú jel.

Kétállású (ON-OFF) szabályozás: Számos irányítástechnikai alkalmazásnál célszerű kapcsoló üzemmódú szabályozót alkalmazni, vagy a szabályozót kibővíteni nemfolytonos elemekkel.

A kétállású szabályozó a hiba függvényében két fix értéket képes kiadni ($u = +1$; $u = -1$ vagy $u = +1$; $u = 0$). Ennek megfelelően a beavatkozó szerv is két üzemmódban működhet a kiadott beavatkozójel függvényében. A túl gyors kapcsolás elkerülésére a szabályozót (h szélességű) hiszterézissel módosíthatjuk (lásd 3.2. ábra).



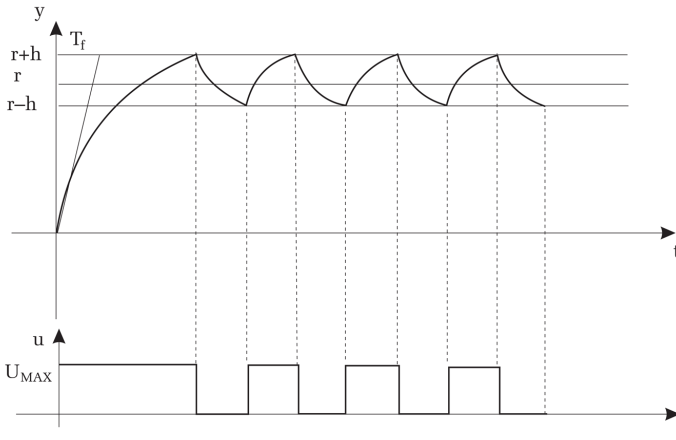
3.2. ábra. Szabályozási rendszer kétállású szabályozóval

Vizsgáljuk meg azt az esetet is, amikor az irányított folyamat *elsőfokú átviteli* függvénnyel modellezhető és a rendszer *tartalmaz holtidőt*.

$$H_f(s) = \frac{K_f}{T_f s + 1} e^{-s\tau} \quad (3.2)$$

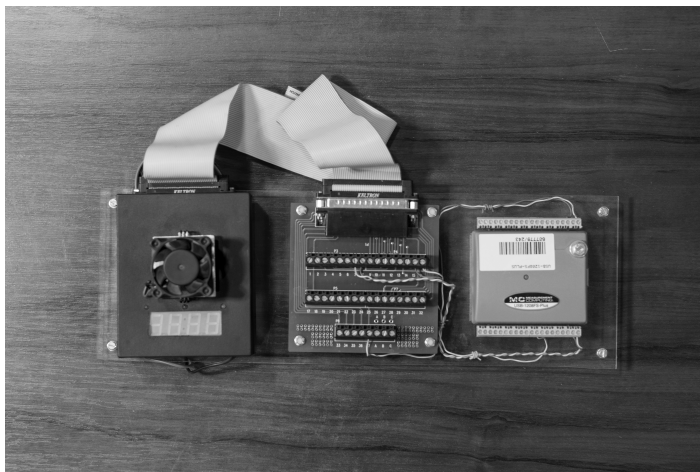
τ a holtidőt jelöli.

Az irányított folyamat válasza hasonló lesz az előző esethez, de a holtidő miatt a lengések periódusa megnő, és a maximális eltérés az előírt értéktől ugyancsak nagyobb lesz (lásd 3.3. ábra). Az irányítási rendszer válaszá-
nak tulajdonságait a hiszterézis szélessége, valamint a rendszer időállandói nagymértékben meghatározzák. A hiszterézis szélessége a legnagyobb eltérésre és a lengések tartományára is hatással van. A hiszterézis szélességének csökkentése nagyobb kapcsolási frekvenciát von maga után.



3.3. ábra. Kétállású szabályozóval irányított holtidős rendszer válasza

3.3. A mérés menete



3.4. ábra. Mérőstand hőmérséklet-szabályozásra

Feladat: Az irányítás célja egy félapka hőmérsékletének konstans értékre szabályozása. Tehát az irányított folyamat bemenete a félapkának leadott hőmennyiség, kimenete a félapka hőmérséklete.

A mérőstand a 3.4. ábrán látható, a folyamat-számítógép illesztést megvalósító áramkört pedig a Függelék F1. ábrája mutatja.

A laboratóriumi gyakorlat során a *Temperature_Control* szoftverprojektben dolgozunk. A program vázát a függelékben található F2. ábra mutatja. A mérés feltételezi, hogy a diák elvégezte a 2. laboratóriumi gyakorlatot.

Folyamat illesztése a számítógéphez: A félapka hőmérsékletét egy LM135 típusú kalibrált félvezető alapú hőmérsékletszenzorral mérjük (érzékelő). Az érzékelő $-40 \dots +135$ Celsius fokos hőmérséklet-tartományban alkalmazható, és kisebb mint 1 Celsius fok a mérési pontossága.

Működésének feltétele, hogy a rajta átfolyó áram $0,5 \dots 5$ milliamper tartományban legyen. Ezt úgy oldhatjuk meg, hogy az egyenfeszültséggel történő betápláláskor az érzékelővel párhuzamosan egy ellenállást helyezünk el. Ha a tápfeszültség U_T és az érzékelőn eső feszültség $U_E = 0,5$ mV, akkor az ellenállás értékét $R = (U_T - U_E)/I$ alapján kell megválasztani, ahol I az érzékelőn átfolyó (elvárt) áram.

Az érzékelő Kelvinben van kalibrálva. 0 Kelvin foknak 0 V felel meg, és ha a hőmérséklet 1 fokot változik, akkor az érzékelőn mért feszültség 10 millivoltot változik. Így, ha a hőmérséklet 0 Celsius fok, az érzékelőn 2,73 Voltot mérünk.

A fémlapnak a hőt egy kapcsolóüzemű teljesítménytranzisztorral adjuk le. Ugyanakkor a fémlap hűtésére egy ventilátort alkalmazunk. Tehát a beavatkozó a tranzisztorból és a ventilátorból álló együttes.

A számítógéphez a rendszert egy *Measurement Computing USB-1208FS*¹ adatbegyűjtő kártya segítségével csatoljuk. Az érzékelő által adott feszültséget az adatbegyűjtő kártya egyik analóg bemenetére kötjük. A tranzisztort és a ventilátort a kártya egy-egy digitális kimenetére kapcsoljuk.

A hőmérséklet-szabályozási szoftver megvalósítása:

I. Az adatbegyűjtő kártya meghajtójának illesztése a programhoz:

A kártya funkcióit C# környezetben az *MCCDaq.dll* állomány segítségével érjük el. A Reference Manager segítségével hozzáadjuk a *dll* állományt a *Temperature_Control* tervezet (a program interfészét lásd a 3.5. ábrán). Ahhoz, hogy a *Temperature_Control_Form.cs* állományban a kártyafüggvényeket alkalmazni tudjuk, be kell illeszteni a *using MccDaq* parancsot, illetve példányosítani kell a kártya osztályt a konstruktorban a *MccDaq.MccBoard DaqBoard = new MccDaq.MccBoard(0)* paranccsal.

II. A hőmérséklet beolvasása az *.AIn* metódussal történik. A visszatérített érték a beolvasott AD konverzió eredménye 16 bites előjeles egészként (*short*). A metódus szintaxisa:

DaqBoard.AIn(int Channel, Range range, out short value)

Channel – az analóg csatorna száma, amire a beolvasott értéket kötötük (jelen esetben 0)

range – a mérési tartomány, amelyben beolvassuk a jeleket (jelen esetben +/- 10 V, az ennek megfelelő konstans *Bip10Volts*)

value – short típusú változó, amibe a konverzió eredménye kerül

III. A hőmérséklet kalibrálása és szűrése:

A hőmérséklet kalibrálásához először átalakítjuk az analóg-digitális konverzió eredményét feszültségértékké. A kártyán az AD konverzió 12 bites, tehát a beolvasott értékek tartománya 0...4095 egység. A mérési tartomány -10 V...+10 V, tehát a feszültséget az alábbi módon számíthatjuk:

$$U = (10 - (-10))/4095 * (ADConversionResult - 2048,5)$$

1 <https://www.mccdaq.com/pdfs/manuals/USB-1208FS.pdf>

A második lépésben a kiszámított feszültséget átalakítjuk hőmérsékletté (Celsius fokba). Mivel 0 fok Celsiusra a feszültség 2,73 V és 1 fok hőmérséklet-változásra a feszültség 10 mV-ot változik, a mért feszültségnek megfelelő hőmérsékletérték Celsius fokban:

$$T_k = 100 * U - 273$$

A harmadik lépésben a kapott hőmérsékletértéket szűrjük. Ehhez a (3.3) összefüggést alkalmazzuk. A T_{k-1} , T_{k-2} értékeket statikus változóként kell elmenteni, és az értékeket a szűrt hőmérséklet (T) számolása után frissíteni kell:

$$\begin{aligned} T &= (T_k + T_{k-1} + T_{k-2})/3 \\ T_{k-2} &= T_{k-1} \\ T_{k-1} &= T_k \end{aligned} \quad (3.3)$$

IV. Kézi vezérlés: Ebben az üzemmódban a felhasználó indítja el a melegítést/hűtést a szoftver felhasználói interfészéről. Döntsük el, hogy manuális üzemmódban vagyunk-e vagy sem a *Manual_radioButton.Checked* értéke alapján. Ha igen, akkor vizsgáljuk meg, hogy melegíteni akarunk (*Heat_checkBox.Checked*), vagy hűteni akarunk (*Cool_checkBox.Checked*), és mentsük el a kívánt műveleteknek megfelelő értéket (1 fűtésre, 2 hűtésre).

V. Automata üzemmód esetén a szabályozási hibát a (3.1) összefüggés alapján számoljuk, ahol a referenciaértéket a *Reference_textBox*-ból olvassuk ki.

VI. Kétállású szabályozó: Kétállású szabályozó esetén csak a hiba előjele szükséges a szabályozáshoz. Ha a hiba pozitív, akkor fűtjük a rendszert, ha a hiba negatív, akkor hűtjük a rendszert.

VII. A beavatkozájel számolása és kiküldése: Függetlenül attól, hogy a kimenetet kézi vagy automata esetben számoltuk ki, ugyanazt a két digitális kimenetet használjuk. A fűtésre az A1 port 1-es bitje, hűtésre pedig az A1 port 2-es bitje van felhasználva, az alábbi paranccsal:

DaqBoard.DOut(DigitalPortType port, short value)

port – a digitális port, ahova a jelet szeretnénk kiküldeni (jelen esetben a port neve *FirstPortA*)

value – a digitális portra kiküldendő 16b-es érték

Mivel a fűtő tranzisztor a nullás bitjére, a hűtést biztosító ventilátor a port első bitjére van kötve, ezért ON esetén 0X01-et, OFF esetén 0X02-t kell kiküldeni a portra. A szabályozási rendszer jeleinek numerikus

kiírása szövegdobozok segítségével történik. A *Temperature_textBox* a mért és kalibrált hőmérsékletet, a *ControlError_textBox* pedig a számolt hőmérséklet hibát tartalmazza. A szabályozás során a *ControlSignal_textBox* dobozba tegyük ki a “Heating”, illetve “Cooling” szöveget a megfelelő esetekben. A grafikus kiíráshoz a *temperatureSeries* listát kell feltölteni a mért értékekkel, frissítve a grafikon interfészt.



3.5. ábra. A hőmérséklet-szabályozást megvalósító program felhasználói interfésze

3.4. Kérdések és feladatok

1. A grafikusán és numerikusan kijelzett hőmérséklet alapján állapítsuk meg, hogy mennyi a rendszer holtideje.
2. A grafikusán és numerikusan kijelzett mérések alapján állapítsuk meg, hogy mennyi a hőmérséklet jel lengésének amplitúdója.
3. Módosítsuk a szabályozást úgy, hogy a szabályozóban helyezzünk el hiszterézis elemet is. A hiszterézis szélessége legyen 2 fok. Vizsgáljuk meg, hogyan módosul a szabályozási rendszer kimenete (hőmérsékletválasza).

PROPORCIONÁLIS HŐMÉRSÉKLET-SZABÁLYOZÁS

4.1. A gyakorlat célja

Az impulzusszélesség modulált jelek szoftverrel történő generálása. Hőmérséklet-szabályozás megvalósítása proporcionális szabályozóval.

4.2. Elméleti bevezető

A proporcionális szabályozás: Az aktuálisan mért hiba függvényében számítjuk ki a beavatkozójelet. A beavatkozójel arányos a mért hibával. Minél nagyobb az e szabályozási hiba (a folyamat y mért kimenete minél távolabb van az r alapjeltől), annál nagyobb lesz a kiszámított beavatkozójel (u). Folytonos időben a beavatkozójel számítása P szabályozó esetén:

$$u(t) = K_P \cdot e(t), \quad K_P > 0 \quad (4.1)$$

$$e(t) = r(t) - y(t) \quad (4.2)$$

K_P proporcionális erősítés, a szabályozó erősítése.

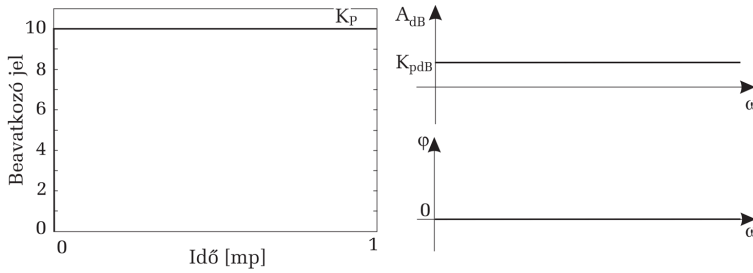
A szabályozó átviteli függvénye megegyezik az ideális erősítő átviteli függvényével.

$$u(s) = K_p \cdot e(s) \quad (4.3)$$
$$H_P(s) = \frac{u(s)}{e(s)} = K_p.$$

Mintavételes megvalósításnál a k -dik mintavételben mért alapjel és kimenet alapján a beavatkozójel számítása:

$$u_k = K_P \cdot e_k = K_P(r_k - y_k). \quad (4.4)$$

A szabályozó egységugrásra adott válasza és Bode-diagramja a 4.1. ábrán látható.



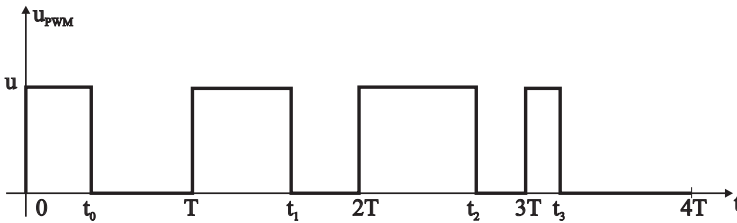
4.1. ábra. Az ideális P szabályozó egységugrásra adott válasza és Bode-diagramja

Impulzus szélesség modulált (PWM) jel generálása: Az impulzusszélesség-modulált jel két diszkrét állapotot felvevő ($0 - u$ vagy $-u - u$ konstans periódusú (T) jel, amelyben egy periódus alatt az impulzus szélessége változtatható (lásd 4.2. ábra).

Határozzuk meg a PWM jel átlagértékét egy T periódus alatt. Feltételezzük, hogy a PWM jel szélessége (kitöltése) $0 < t < T$.

$$\tilde{u}_{PWM} = \frac{1}{T} \int_0^T u_{PWM} d\tau = \frac{1}{T} \left[\int_0^t u d\tau + \int_t^T 0 d\tau \right] = \frac{1}{T} u \int_0^t d\tau = \frac{u}{T} t \quad (4.5)$$

Tehát a PWM jel átlagértéke egyenesen arányos a kitöltéssel.



4.2. ábra. PWM jel

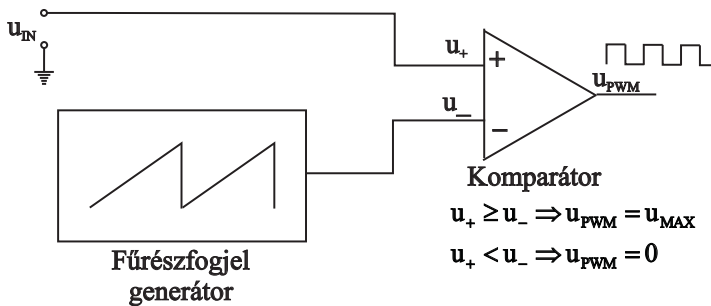
A *PWM jel generátorok* olyan rendszerek, amelyeknek a bemenete a PWM kitöltésének megfelelő jel, a kimenete pedig a PWM jel.

PWM jel generálása megvalósítható analóg elektronikai eszközökkel, digitális elektronikai eszközökkel, de könnyen le is programozható, ha a kívánt jelfrekvencia alacsony.



4.3. ábra. PWM jel generátor

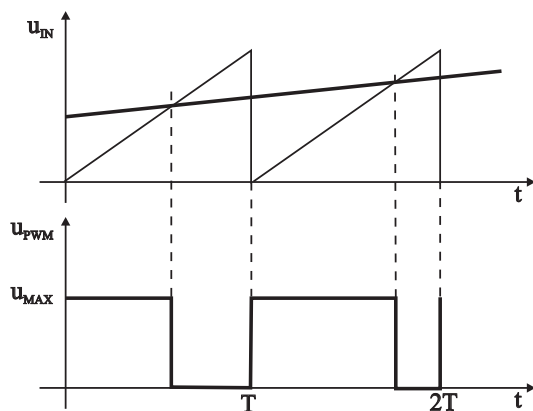
Az *analóg megoldásnál* szükségünk van egy fűrészfog-jel generátorra, egy analóg komparátorra. Ebben az esetben a kitöltés egy u_{IN} feszültség. Ezt hasonlítjuk össze a fűrészfog-jel generátor kimenetével. Ha ez a feszültség szint nagyobb, mint a fűrészfog-jel értéke, akkor a komparátor kimenete 1, másként a kimenet 0. A komparátor kimenete egy PWM jel lesz, aminek a kitöltése u_{IN} feszültséggel lesz arányos.



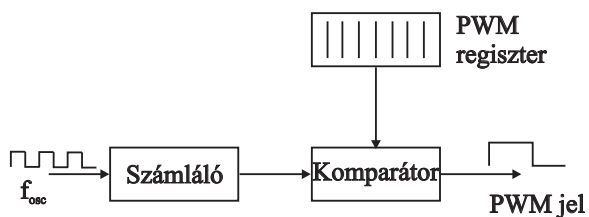
4.4. ábra. Analóg PWM jel generátor

Digitális áramkörökkel megvalósított PWM jel esetén a kitöltési tényezőt egy regiszterben tároljuk. Ugyanakkor szükségünk van egy regiszterrel megegyező bithosszúságú számlálóra is. A számláló bemenete egy órajel. Így a számláló értéke folyamatosan nő, majd ha eléri a maximumot, akkor nullázódik, majd ismét nő. A számláló kimenetét és az órajelet összehasonlítjuk egy digitális komparátorral. A komparátor kimenete PWM jel lesz, aminek a kitöltése egyenlő a regiszterbe beírt értékkel. A PWM jel periódusa: az órajel periódusa szorozva a számláló bithossza.

Szoftveres megoldás esetén egy ciklikus megszakítást (*Timer*) alkalmazhatunk. Minden ciklusban növelünk egy változót (*counter*). Tároljuk az u változóban a PWM kitöltését. Ha a *counter* elért egy maximális értéket (*MAXCOUNTER*), akkor nullázzuk. Ha a *counter* kisebb, mint u értéke, akkor a DOUT digitális kimeneten egyet, máskülönben nullát küldünk ki.



4.5. ábra. A PWM jel generálása



4.6. ábra. Digitális PWM jel generátor

Egy szoftver megoldás PWM jel generálására:

```
const int MAXCOUNTER=100;
unsigned int counter=0, u;
Timer(){
  if (counter==0)
  read (u);
  if(counter<u)
  DOUT=1;
  else
    DOUT=0;
    counter++;

  if (counter == MAXCOUNTER)
  counter=0;
}
```

A PWM jelek segítségével egy digitális kimeneten folytonos módon végezhetünk szabályozást. A PWM jel periódusát úgy kell megválasztani, hogy nagyságrenddel kisebb legyen, mint az irányított folyamat legkisebb időállandója. Így a folyamat bemenetén levő PWM jelnek a hatása a kimeneten csak átlagolva jelenik meg. Ha motorvezérlésre szeretnénk használni a PWM jelet, akkor magas jelfrekvenciát kell választani (például 20 kHz), hőmérséklet-szabályozás esetén a PWM frekvenciája lehet akár 0,1–1 Hz tartományban is. Kemence hőmérséklet-szabályozása esetén a PWM jellel egy kapcsolót vezérelhetünk, amely a melegítő ellenállásra vagy rákapcsolja, vagy lekapcsolja a feszültséget.

4.3. A mérés menete

A laboratóriumi gyakorlat során a *Temperature_Control* szoftverprojektben dolgozunk. A program vázát a függelékben található F2. ábra mutatja. A mérés feltételezi, hogy a hallgató megoldotta a 2. és 3. laboratóriumi gyakorlatokat.

1. A proporcionális szabályozó megvalósítása:

- A *controlTimer_Tick* metódusban megírt automata szabályozó részben dolgozunk, és felhasználjuk az előző laboron kiszámolt hiba értéket, majd a beavatkozójelet is kiszámoljuk a (4.1) összefüggés alapján.
- A beavatkozójelet korlátozzuk a $\pm 100\%$ tartományra.

```
if (u > 100) u = 100
```

```
if (u < -100) u = -100
```

2. A PWM jel szoftveres generálása:

Az elméleti részben bemutatott szoftvermegoldás kibővített formáját alkalmazzuk a PWM jel generálására.

- Egy új időzítőt helyezünk el a grafikus felületen – nevezzük el *pwm-Timer*-nek, a konstruktorban állítsuk be a periódust 10 ms-ra, és indítsuk el.
- A *pwmTimer* időzítő *Tick* metódusában csak akkor végezzünk valamilyen feladatot, ha kiválasztottuk az automata üzemmódot.
- Minden meghíváskor növeljünk egy számlálót, amely ha eléri a százzal, nullázzuk, és átmásoljuk a beavatkozójelet egy új, PWM kitérésnek megfelelő változóba.
- Ha az elmentett szabályozójel előjele pozitív és abszolút értéke nagyobb, mint a számláló, akkor fűtjük a rendszert.

- Ha az elmentett szabályozójel előjele negatív és abszolút értéke nagyobb, mint a számláló, akkor hűtjük a rendszert.
- Minden más esetben nullát adunk ki a digitális portokra.

3. Kiértékelés:

- Ábrázoljuk grafikusan a hőmérsékletet és a referenciajelet ugyanazon a *Temperature_chart* grafikonon. Ábrázoljuk a beavatkozójelet a *ControlSignal_chart* grafikonon.
- Vizsgáljuk meg a hőmérséklet-kimenet és a beavatkozójel változását egymáshoz képest.
- Vizsgáljuk meg a szabályozási rendszer válaszát különböző K_P értékekre.

4.4. Kérdések és feladatok

1. Módosítsuk a programban a PWM jel felbontását (számláló maximális értékét) 100-ról 20-ra. Vizsgáljuk a hőmérséklet válaszmódosulását.
2. Módosítsuk a P szabályozót PD szabályozásra (a deriválási idő legyen 1-nél kisebb). Milyen változást látunk a szabályozott hőmérsékletjelen?
3. Írjuk át a programot olyan formában, hogy a hűtés-kimenetet nem alkalmazzuk. Hogyan változik a rendszer válasza?

HŐMÉRSÉKLETPROFIL-KÖVETÉS PI-SZABÁLYOZÓVAL

5.1. A gyakorlat célja

Referenciaprofil-generálás implementálása, alkalmazás hőmérséklet-szabályozásra. Mintavételes PI-szabályozás megvalósítása és alkalmazása hőmérséklet-szabályozásra.

5.2. Elméleti bevezető

A *PI-szabályozó*: A PI-szabályozó figyelembe veszi a szabályozási hiba múltbeli alakulását is. A múltbeli hiba összegzett hatását integrálócatornával számítjuk. A folytonos idejű PI-szabályozó esetében a beavatkozójelet az alábbi formában kapjuk:

$$u(t) = K_P \cdot \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau \right), \quad K_P, T_i > 0. \quad (5.1)$$

A $T_i > 0$ paraméter az integrálási idő.

Az átviteli függvényt az alábbi formában kapjuk:

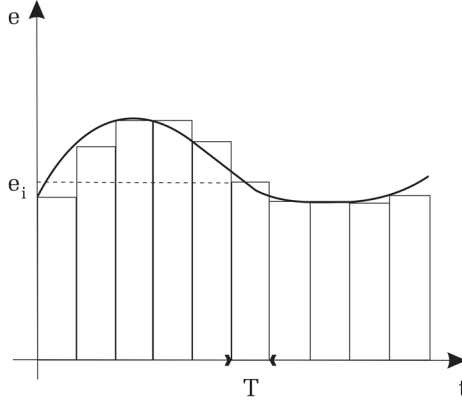
$$\begin{aligned} u(s) &= K_P \cdot \left(e(s) + \frac{1}{T_i s} e(s) \right) \\ H_{PI}(s) &= \frac{u(s)}{e(s)} = K_P \cdot \left(1 + \frac{1}{T_i s} \right). \end{aligned} \quad (5.2)$$

A PI-szabályozó kauzális, az integrátor csatorna miatt a pólusa a zéróban van.

A szabályozó mintavételes megvalósításánál az integráló tag megközelítésére a téglalap módszert lehet alkalmazni, az $e(t)$ hibafüggvény alatti területet T szélességű téglalapokkal közelítjük meg (lásd 5.1. ábra). Az integrál megközelítőleg egyenlő a téglalapok területének összegével a k -dik

mintavételig:

$$\int_0^{kT} e(t)dt = \int_{i=0}^k T e_i. \quad (5.3)$$



5.1. ábra. Az integrál megközelítése téglalapokkal

Felhasználva az (5.3) megközelítést, a beavatkozójel számítása a k -dik mintavételben:

$$u_k = K_P \left(e_k + \frac{T}{T_i} \sum_{i=0}^k e_i \right). \quad (5.4)$$

Egyszerűbb implementálási forma kapható, ha a beavatkozójel számítása rekurzívan történik. Felírva a beavatkozójelet a k és $k - 1$ mintavételben, majd egymásból kivonva:

$$\begin{aligned} u_k &= K_P \left(e_k + \frac{T}{T_i} \sum_{i=0}^k e_i \right) \\ u_{k-1} &= K_P \left(e_{k-1} + \frac{T}{T_i} \sum_{i=0}^{k-1} e_i \right) \\ u_k - u_{k-1} &= K_P \cdot \left(e_k - e_{k-1} + \frac{T}{T_i} e_k \right). \end{aligned} \quad (5.5)$$

Így a beavatkozójel a k -dik mintavételben:

$$u_k = u_{k-1} + K_P \left(e_k - e_{k-1} + \frac{T}{T_i} e_k \right). \quad (5.6)$$

Látszik, hogy a beavatkozójel értéke mindaddig változni fog, amíg a szabályozási hiba nem válik nullává.

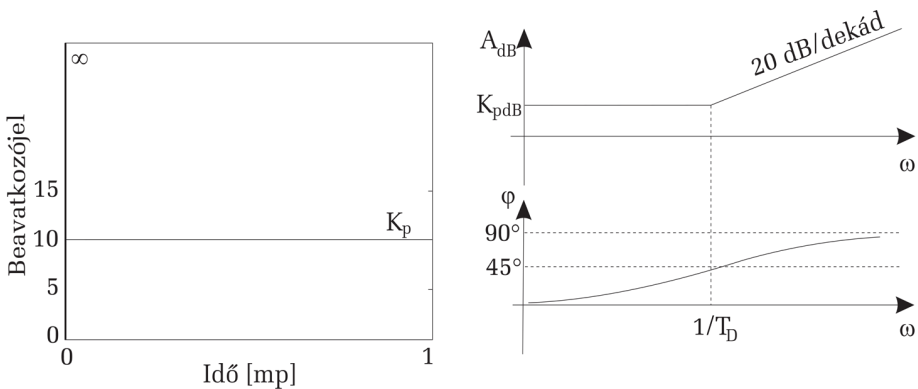
A mintavételes megközelítés átviteli függvénye a Z transzformált alkalmazásával számítható:

$$u_k = u_{k-1} + q_0 \cdot e_k + q_1 \cdot e_{k-1} \quad q_0 = K_p \left(1 + \frac{T}{T_i} \right) \quad q_1 = -K_p \quad (5.7)$$

$$u(z) = u(z)z^{-1} + q_0 \cdot e(z) + q_1 \cdot e(z)z^{-1} \quad (5.8)$$

$$H_{PI}(z) = \frac{u(z)}{e(z)} = \frac{q_0 z + q_1}{z - 1}.$$

A PI-szabályozó egységugrásra adott válasza és Bode-diagramja az 5.2. ábrán látható. Mivel az egységugrás hiba bemenet konstans bármely $t > 0$ pillanatban, a beavatkozójel az integráló tag miatt sebességugrás-szerűen nőni fog.



5.2. ábra. Az ideális PI-szabályozó egységugrásra adott válasza és Bode-diagramja

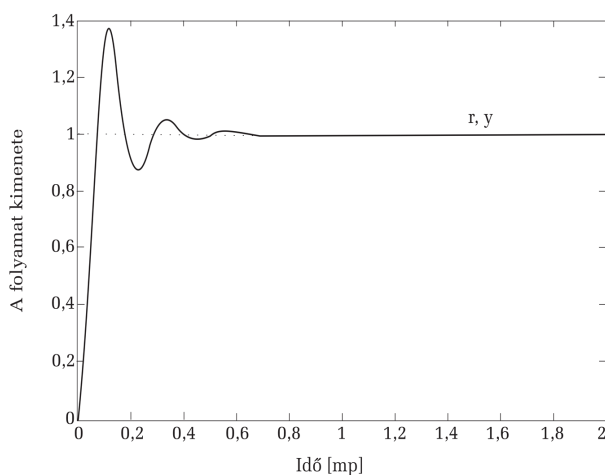
Az integrálás hatása az alapjelkövetésre: Feltételezzük, hogy a nyílt rendszer modellje az alábbi formában írható fel:

$$H_N(s) = K_P K_F \frac{1}{s} \frac{\left(\frac{1}{z_1} s + 1 \right) \dots \left(\frac{1}{z_m} s + 1 \right)}{\left(T_1 s + 1 \right) \dots \left(T_n s + 1 \right)}. \quad (5.9)$$

Feltételezve, hogy az *alapel* *egységugrás*, $r(s) = 1/s$, az állandósult állapotbeli hiba:

$$\begin{aligned} \lim_{t \rightarrow \infty} e(t) &= \lim_{s \rightarrow 0} s \frac{1}{1 + H_N(s)} \cdot \frac{1}{s} \\ &= \lim_{s \rightarrow 0} s \frac{1}{1 + \frac{K_P K_F}{s}} \cdot \frac{1}{s} \\ &= \lim_{s \rightarrow 0} \frac{s}{s + K_P K_F} = 0. \end{aligned} \quad (5.10)$$

Ha van integrátor akár a folyamatban, akár a szabályzóban, *egységugrás* *alapelre* az állandósult állapotbeli hiba nullához konvergál (lásd 5.3. ábra).

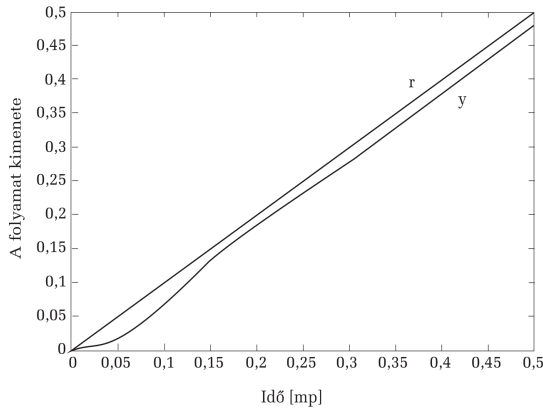


5.3. ábra. Állandósult állapotbeli hiba egységugrásra (integrátorral)

Feltételezve, hogy az *alapel* *sebességugrás*, $r(s) = 1/s^2$, az állandósult állapotbeli hiba:

$$\lim_{t \rightarrow \infty} e(t) = \lim_{s \rightarrow 0} s \frac{1}{1 + \frac{K_P K_F}{s}} \cdot \frac{1}{s^2} = \lim_{s \rightarrow 0} s \frac{1}{1 + \frac{K_P K_F}{s}} \cdot \frac{1}{s} = \frac{1}{K_P K_F}. \quad (5.11)$$

Ha az *alapel* *sebességugrás*, az állandósult állapotbeli hiba nem válik zéróvá, de csökkenthető a szabályzó erősítésének növelésével (lásd 5.4. ábra).



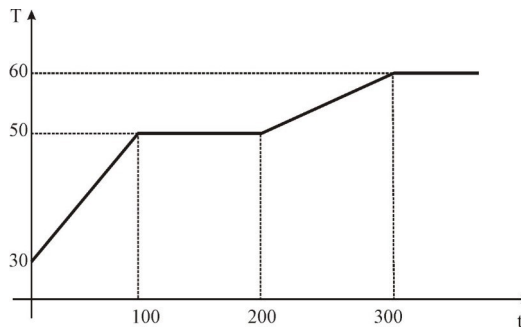
5.4. ábra. Állandósult állapotbeli hiba sebességugrásra (integrátorral)

5.3. A mérés menete

Feladat: A *Temperature_Control* tervben állítsunk elő egy hőmérsékletprofil (időben változó alapjelet), majd PI-szabályozóval szabályozzuk rá a folyamat hőmérséklet-kimenetét a profilra.

A laboratóriumi gyakorlat során a *Temperature_Control* szoftverprojektben dolgozunk. A program vázát a függelékben található F2. ábra mutatja. A mérés feltételezi, hogy a hallgató megoldotta a 2., 3. és 4. laboratóriumi gyakorlatokat.

1. *Profilgenerálás:* Valósítsuk meg az 5.5. ábrán látható hőmérsékletprofil a *GetControlReference* függvényben. Az ábrán a T hőmérséklet Celsius fokban, az idő pedig másodpercben van megadva.



5.5. ábra. Referencia-hőmérsékletprofil

– Először az időalapot generáljuk ki. Ismerve, hogy a mintavételi periódus $TM = 1$ másodperc, az idő változót (t) minden mintavételi periódusban növeljük: $t = t + TM$;

– A profilban vannak konstans hőmérsékletszakaszok és konstans meredekségű hőmérsékletszakaszok. Feltételezve, hogy a konstans meredekségű szakaszok kezdeti hőmérséklete T_k , véghőmérséklete T_v , kezdeti időpillanata t_k , végső időpillanata t_v , akkor a hőmérséklet alapjelet az alábbi inkremenssel kell növelni minden mintavételi periódusban:

$$DT = \frac{T_v - T_k}{t_v - t_k} TM. \quad (5.12)$$

Az 5.5. ábrán levő hőmérsékletprofilat az alábbi módon programozhatjuk le:

```

theta = 30
if (t < 100)
    theta += DT1
else if (t < 200)
    theta = 50
else if (t < 300)
    theta += DT2
else
    theta = 60

```

2. A PI-szabályozó megvalósítása: A *controlTimer_Tick* metódusban:

- A megírt proporcionális szabályozót cseréljük le PI-szabályozóra.
- Az (5.6) összefüggés alapján kiszámoljuk a beavatkozójelet. T a mintavételi periódust jelöli. Válasszuk $K_P = 5,16$, $T_I = 21,3$ s.
- A beavatkozójelet korlátozzuk a +/-100 tartományra.

3. Kiértékelés:

- Ábrázoljuk grafikusán a beavatkozójelet és a hőmérsékletet a *GetTemperatureList* és a *GetControlList* függvényekben.
- Vizsgáljuk meg a hőmérséklet-kimenet és a beavatkozójel változását egymáshoz képest.
- Vizsgáljuk meg a hőmérséklet-kimenetet különböző K_P és T_I értékekre.

5.4. Kérdések és feladatok

1. Programozzuk le az alapjelkövetés feladatát általánosítva, feltételezve, hogy N pont áll rendelkezésünkre: $(t_1, T_1), (t_2, T_2), \dots (t_N, T_N)$.
2. Írjuk át a programot olyan formában, hogy a hűtés-kimenetet nem alkalmazzuk. Hogyan változik a rendszer válasza?
3. Módosítsuk az irányítási algoritmust PI- helyett P-szabályozóra és végezzük el a profilkövetést. Hogyan változik a követési pontosság?

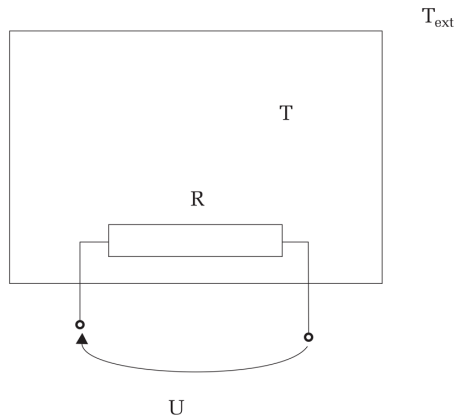
IPARI KEMENCÉK PID-SZABÁLYOZÁSA

6.1. A gyakorlat célja

Az ellenállással melegített ipari kemencék modelljének meghatározása. Az Oppelt PID tervezési módszer alkalmazása ipari kemencék hőmérséklet-szabályozására. Az ipari kemencék szabályozási hurkának szimulációja.

6.2. Elméleti bevezető

A *kemence modellje*: Ellenállással melegített kemencék egyszerűsített matematikai modelljét a kemence hőegyensúlya alapján lehet felírni: az ellenállás által termelt hő egy része eltárolódik a kemence tömegében, a többi pedig a hővezetés jelensége alapján átadódik a környezetnek.



6.1. ábra. Ellenállással melegített kemence vázlata modellezéshez

Az ellenálláson termelődött hő dt idő alatt:

$$\Delta Q_i = \frac{U^2}{R} dt \quad (6.1)$$

ahol R az ellenállás értéke, U az ellenállásra kapcsolt feszültség.

A környezetnek leadott hő dt idő alatt:

$$\Delta Q_S = \lambda S(T - T_{\text{ext}})dt \quad (6.2)$$

ahol S a kemence felülete, λ a felület hővezetési állandója, T a hőmérséklet a kemencében, T_{ext} a környezet hőmérséklete. A kemence által tárolt hőmennyiség:

$$\Delta Q_m = McdT \quad (6.3)$$

ahol M a kemence tömege, c a kemence fajhője.

Az (6.1), (6.2), (6.3) alapján a kemence működését az alábbi egyenlet írja le:

$$Mc \frac{dT}{dt} = -\lambda S(T - T_{\text{ext}}) + \frac{U^2}{R}. \quad (6.4)$$

Amint látható, a modell nem lineáris (a hőmérséklet-változás négyzetesen függ a feszültségtől). Az ellenállás értéke a hőmérséklet függvényében változhat. Ugyanakkor ez a modell nem veszi figyelembe a rendszer holtidejét, ami nagy kemencék esetében nem elhanyagolható.

A rendszermodell linearizálása: Feltételezzük, hogy a bemeneti feszültség egy négyzetgyökönő elemen keresztül éri el a kemencét, és tekintsük ezt a rendszer bemenetének:

$$U := \sqrt{U}. \quad (6.5)$$

A megközelítő értéket behelyettesítve a (6.4) képletbe kapjuk a következő összefüggést:

$$Mc \frac{dT}{dt} = -\lambda S(T - T_{\text{ext}}) + \frac{U}{R}. \quad (6.6)$$

Legyen a rendszer kimenete (T) a kemence és a környezet közötti hőmérséklet-különbség:

$$T := T - t_{\text{ext}}. \quad (6.7)$$

Feltételezve, hogy T_{ext} konstans, kapjuk:

$$\frac{Mc}{\lambda S} \frac{dT}{dt} + T = \frac{1}{R\lambda S} U. \quad (6.8)$$

Vezessük be az alábbi paramétereket: T_k a kemence időállandója, K_k a rendszer erősítése:

$$T_k = \frac{Mc}{\lambda S} \quad (6.9)$$

$$K_k = \frac{1}{R\lambda S}. \quad (6.10)$$

Figyelembe véve a (6.7), (6.8) jelöléseket, a következő egyenletet kapjuk:

$$T_k \frac{dT}{dt} + T = K_k U. \quad (6.11)$$

A (6.11) differenciálegyenletből következik, hogy:

$$H_k(s) = \frac{T(s)}{U(s)} = \frac{K_k}{T_k s + 1}. \quad (6.12)$$

A kemence modellezésénél a holtidőt általában nem tudjuk elhanyagolni. A holtidő az alábbi módon jelenik meg a rendszer modelljében:

$$H_k(s) = \frac{K_k}{T_k s + 1} e^{-\tau s} \quad (6.13)$$

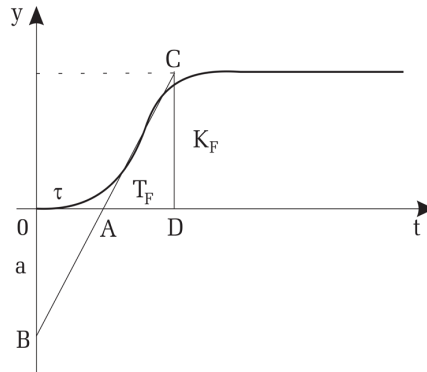
ahol τ a holtidő.

Az *Oppelt tervezési módszer*: Számos olyan szabályozó tervezési módszer létezik, amely a rendszer egységugrásra adott válasza alapján adja meg a szabályozó paramétereit. Ilyen hangolási módszer az Oppelt-módszer, amely feltételezi, hogy az irányított folyamat elsőfokú stabil rendszer, amely holtidővel is rendelkezik. Ebben az esetben az irányított folyamatot három paraméterrel jellemezhetjük: K_F erősítés, T_F időállandó, τ holtidő. Az Oppelt-módszer lényege, hogy a folyamat egységugrásra adott válasza alapján határozzuk meg ezen paramétereket, majd a folyamat paramétereinek ismeretében hangoljuk be a PID-szabályozót.

A stabil rendszer egységugrásra adott válaszát könnyen megkaphatjuk, hiszen ehhez csak arra van szükség, hogy a folyamatnak konstans egységugrás-bemenetet biztosítsunk, miközben mérjük a kimenetet. Bizonyos folyamatoknál problémát jelenthet, hogy a K_F értéke túlságosan nagy, nem a mérhető tartományban van az egységugrásra adott válasz állandósult állapota, amely körül a szabályozás történik, a kimenet sohasem éri el a K_F értékét. Ebben az esetben a K_F/T_F érték közelítőlegesen meghatározására a rendszer válaszát egyenesekkel közelítjük. A 6.2. ábra alapján az OAB háromszög hasonló az ACD háromszöggel, tehát:

$$OAB_{\Delta} \sim ACD_{\Delta} \Rightarrow \frac{a}{K_F} = \frac{\tau}{T_F} \Rightarrow \frac{K_F}{T_F} = \frac{a}{\tau}. \quad (6.14)$$

Nagy K_F értékek esetén a válasz alapján legkönnyebb a τ és az a paramétereket mérni.



6.2. ábra. Stabil elsőfokú holtidős rendszer egységugrásra adott válasza

Ezért az Oppelt-módszer esetén ezeket a paramétereket használjuk a PID-paraméterek meghatározására. A különböző struktúrájú szabályozók esetén az alábbi paraméter-választások javasoltak:

6.1. táblázat. Oppelt-módszer – hangolás

	K_P	T_i	T_D
P	$1/a$	–	–
PI	$0,8/a$	3τ	–
PID	$1,2/a$	2τ	$0,42\tau$

Csak a P szabályozó nem garantálja a zérus állandósult állapotbeli hibát egységugrás alapjelre, ezért ha nagy pontosságú szabályozást szeretnénk, integrátort kell elhelyezni a szabályozóba.

6.3. A mérés menete

Folyamatparaméterek: Legyen a (6.13) modell által leírt folyamat az alábbi paraméterekkel:

$$K_F = 0,05$$

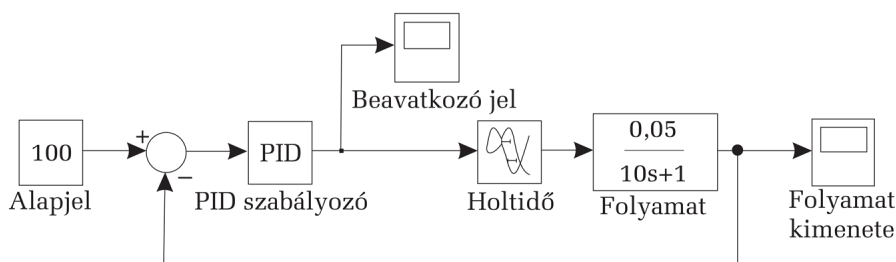
$$T_F = 10 \text{ másodperc (mp)}$$

$$\tau = 3 \text{ mp}$$

Szabályozótervezési feladatok:

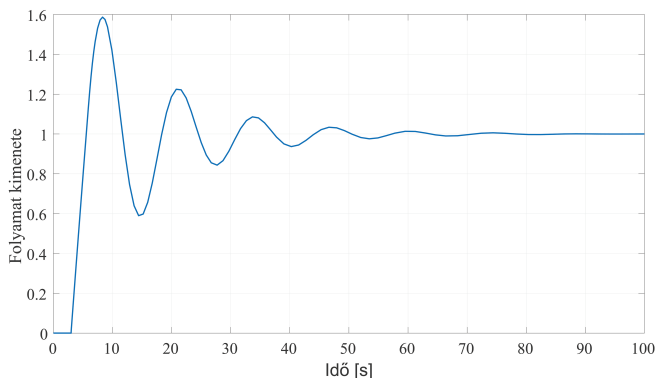
1. Tervezzük P folyamatnak az Oppelt-módszer alapján (6.1. táblázat)
2. Tervezzük PI folyamatnak az Oppelt-módszer alapján (6.1. táblázat)
3. Tervezzük P folyamatnak az Oppelt-módszer alapján (6.1. táblázat)
4. Teszteljük a kapott szabályozási rendszert úgy, hogy az alapjel 100 legyen.

Szimulációs vizsgálatok: A PID-szabályozás Matlab/Simulink diagramját a 6.3. ábra mutatja. A holtidős rendszer viselkedését a sorban elhelyezett 'Transport Delay' és 'Transfer Function' blokkokkal szimuláljuk.



6.3. ábra. Holtidős rendszer PID-szabályozásának Simulink diagramja

- Számítsuk ki az állandósult állapotbeli hibát és a túllövést a három tervezési esetben. Látható, hogy P-szabályozó esetén az állandósult állapotbeli hiba jelentős, tehát nem alkalmazható a feladat megoldására. A PID-szabályozó garantálja a zérus állandósult állapotbeli hibát. Mindkét esetben jelentős, 40% körüli túllövésre számíthatunk.
- Ezt elkerülhetjük, ha az alapjelet nem egységugrásnak, hanem a szabályozás indításakor korlátosan növekvő sebességugrásnak választjuk, majd amikor elérjük az előírt értéket, az alapjelet konstans értéken tartjuk. Módosítsuk úgy az alapjelet, hogy az alapjel 25 másodperc alatt lineárisan növekedjen az előírt értékig.



6.4. ábra. Szimulációs eredmény PID-szabályozóval

6.4. Kérdések

1. Változtassuk a szabályozó erősítését PID-szabályozás esetén. Milyen hatással van a szabályozási kör stabilitására? Milyen KP-értékre válik instabillá a rendszer?
2. P-szabályozás esetén változtassuk a szabályozási kör erősítését. Milyen hatással van az állandósult állapotbeli hibára?
3. Hasonlítsuk össze a szabályozási minőségi jellemzőket holtidős és nem holtidős rendszer esetén ugyanazzal a PID-szabályozóval.

ÖNHANGOLÓ PID-SZABÁLYOZÁS

7.1. A gyakorlat célja

A Ziegler–Nichols-hangolás és az önhangoló PID-szabályozó elvének bemutatása. A hangolás elvégzése szimulációs környezetben. A módszer demonstrálása valós időben kemence hőmérséklet-szabályozásának keretében.

7.2. Elméleti bevezető

A *Ziegler–Nichols-módszer*: A Ziegler–Nichols-módszert csak olyan folyamatoknál lehet alkalmazni, amelyeknél a technológia megengedi, hogy a szabályozási kört a stabilitás határán működtessük.

A hangolási módszernél nem szükséges ismerni a rendszer válaszát.

A hangoláshoz a szabályozási körből kiiktatjuk az integráló és deriváló csatornát a ($T_i = \infty$, $T_d = 0$ paraméterezéssel). Így a szabályozó egy erősítőre redukálódik (P szabályozó). A szabályozó K_p erősítését nulláról kell növelni addig, amíg a zárt rendszer eléri a stabilitás határát. A stabilitás határán állandósult állapotban a folyamat kimenete szinuszosan leng az alapjel körül. Jelölje $K_{P_{krit}}$ a kritikus erősítést, vagyis a szabályozó erősítését a konstans amplitúdójú lengések bekövetkeztekor. Jelölje T_{krit} a kritikus periódust, a konstans amplitúdójú lengések periódusát. A szabályozó hangolása ezen értékek alapján történik.

7.1. táblázat. Ziegler–Nichols-hangolás

P	$K_P = 0,45 K_{P_{krit}}$	–	–
PI	$K_P = 0,45 K_{P_{krit}}$	$T_i = 0,85 T_{krit}$	–
PID	$K_P = 0,60 K_{P_{krit}}$	$T_i = 0,50 T_{krit}$	$T_D = 0,12 T_{krit}$

A szabályozási kör csillapítása a 7.1. táblázat alapján is $\zeta = 0,25$, ami 40% körüli túllövést eredményez. Mintavételes megvalósításnál a mintavételi periódust $T \cong (0,1 \dots 0,3)T_{krit}$ körüli értékre kell választani.

A hangolás előnye, hogy nincs szükség az egységugrásra adott válaszra, az egyetlen paraméter, amit a folyamatról ismerni kell, a T_{krit} . A hátránya egyrészt az, hogy a szabályozási rendszert el kell vinni a stabilitás határára, másrészt, hogy ha az irányított folyamat lassú, a hangolás időigényes. Másik előnye, hogy a kidolgozott táblázat használható az önhangoló szabályozások megvalósításánál.

Önhangoló PID-szabályozók: Az önhangoló szabályozók képesek a saját paramétereik meghatározására, ami jelentősen megkönnyíti használatukat. A szabályozó maga határozza meg azokat a folyamatra jellemző paramétereket, amelyek alapján a szabályozóparaméterek kiszámíthatóak. Így az önhangoló szabályozók esetében induláskor van egy úgynevezett *hangolási üzemmód* (*Tuning*), amikor a szabályozó megméri/kiszámítja a folyamat azon paramétereit, amelyek szükségesek a szabályozóparaméterek meghatározásához, majd meg is határozza ezeket. Ezután átkapcsol *önműködő üzemmód*ba (*Automat*), amely során a felparaméterezett szabályozó irányítja a folyamatot zárt hurokban.

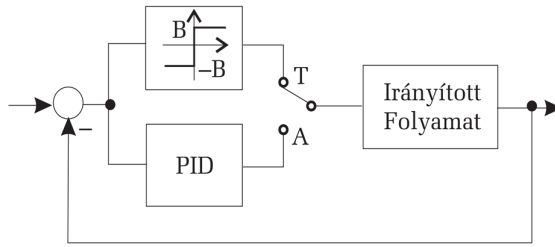
Az önhangoló PID-szabályozók egy típusa a Ziegler–Nichols-módszer alapján határozza meg a szabályozóparamétereket. E módszer esetében a hangolás is zárt rendszerben történik. A hangolás alatt egy P-szabályozó erősítését kell növelni, amíg a folyamat kimenete leng, a zárt rendszer stabilitás határára jut, majd a szabályozó erősítése és a lengések periódusa alapján meghatározható.

Az önhangoló PID-szabályozó ugyancsak a kritikus erősítés és periódus alapján számolja ki a szabályozóparamétereket, de anélkül, hogy eljuttatná a rendszert a stabilitás határára. Ehhez egy kétállású (*ON-OFF*) szabályozót alkalmaz. Így a szabályozóban párhuzamosan egy PID- és egy kétállású szabályozó van (lásd 7.1. ábra).

A hangolás üzemmódban a kétállású szabályozó aktív. Ezzel a szabályozóval a folyamat kimenete állandósult állapotban is lengeni fog az előírt érték körül a nagy energiájú kapcsoló üzemmódú szabályozás és a folyamat tehetetlensége miatt. Ugyanakkor a rendszer nincs a stabilitás határán, a kétállású szabályozó biztosítja, hogy a szabályozási kör ne váljon instabillá.

A hangoláshoz először meg kell határozni azt az *ekvivalens P-szabályozót*, amely a stabilitás határán, a kétállású szabályozó által generált lengéseket képes létrehozni. Feltételezzük, hogy a kétállású szabályozó kimenete $\pm B$ értékeket vehet fel. A folyamat kimenetén az állandósult állapotbeli lengések amplitúdója a . A kritikus erősítés:

$$K_{P_{\text{krit}}} = \frac{4B}{a\pi}. \quad (7.1)$$



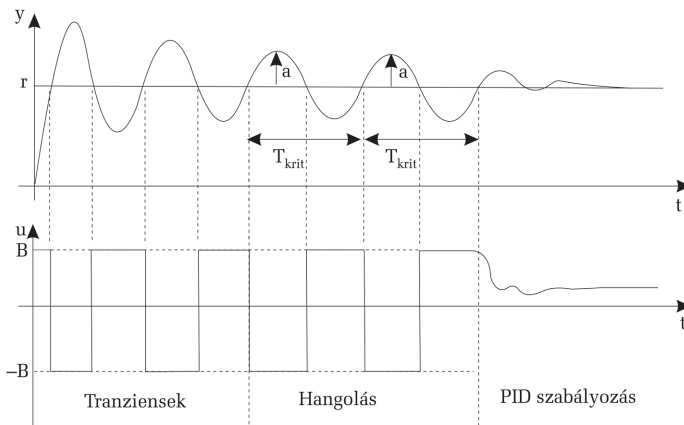
7.1. ábra. Önhangoló PID-szabályozás

Az önhangoló PID működése:

- A szabályozás kétállású szabályozóval indul.
- A tranziensek lecsengése után a szabályozó megméri a lengések periódusát (T_{krit}) és a lengések amplitúdóját (a).
- A (7.1) összefüggés alapján kiszámolja a kritikus erősítést ($K_{P_{krit}}$).
- A kritikus erősítés és kritikus periódus alapján a 7.1. táblázat felhasználásával meghatározza a PID-szabályozó paramétereit.
- Átkapcsolás PID-szabályozásra.

Mivel mérési zajokra számíthatunk, a lengések periódusának és amplitúdójának meghatározásánál fontos, hogy a mért jelet szűrjük, valamint a mérést több lengéscikluson keresztül ismételjük meg, az amplitúdót és periódust több mérés átlagaként számítsuk.

A 7.2. ábrán az önhangoló PID-szabályozó tipikus válasza látható.



7.2. ábra. Önhangoló szabályozás (folyamat kimenete, beavatkozójel)

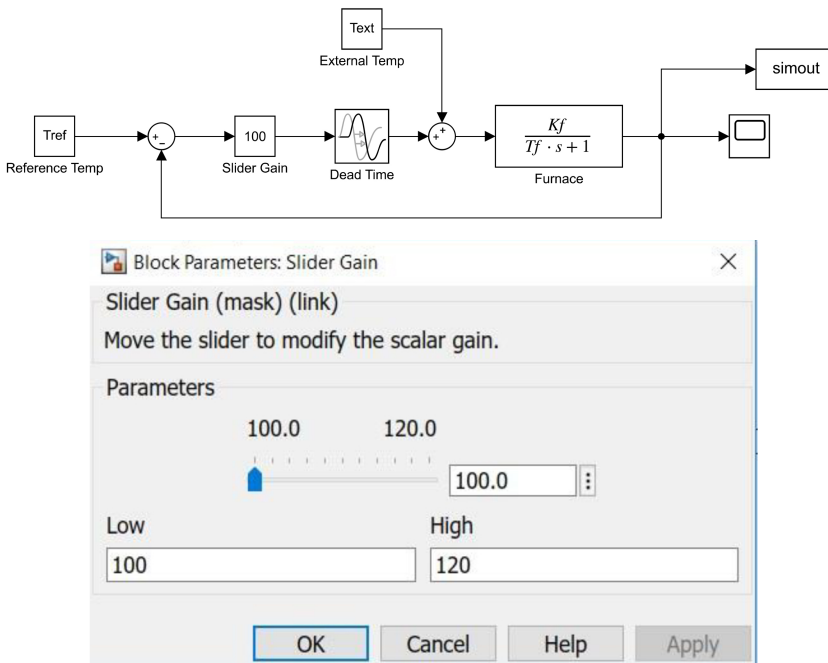
7.3. A mérés menete

Szimulációs feladatok: Legyen az alábbi elsőfokú rendszer holtidős rendszer:

$$H_F(s) = \frac{K_F}{T_F s + 1} e^{-\tau s}.$$

A folyamat paramétereit: A folyamat erősítése $K_F = 0,05$, a folyamat időállandója $T_F = 10$ másodperc (mp), a folyamat holtideje $\tau = 3$ mp.

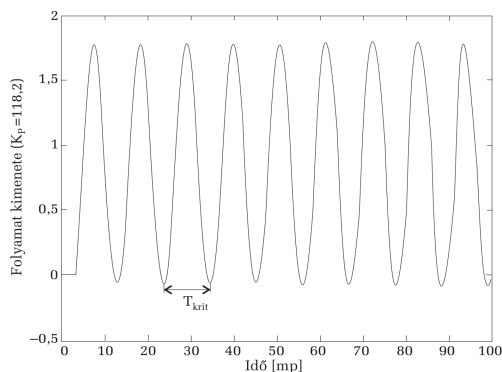
– Készítsük el a 7.3. ábrán látható szimulációs diagramot. T_{ext} konstans bemeneti zavart jelöl. A *Slider Gain* állítható erősítésű P-szabályozóként alkalmazzuk. A *Slider Gain* paramétereit az alábbi módon válasszuk meg: MIN: 100, MAX: 120. A szimuláció során állítsuk a szimulációs megállási időt végtelenre (*inf*).



7.3. ábra. Szimulációs diagram a Ziegler–Nichols-hangoláshoz

– Határozzuk meg a kritikus erősítést és periódust: folyamatosan növeljük az erősítést, és amikor azt tapasztaljuk, hogy a folyamat kimenete

konstans amplitúdóval leng, állítsuk meg a szimulációt. Olvassuk le a folyamat kimenetéről az amplitúdót és a periódust.



7.4. ábra. P-szabályozás – a folyamat válasza a stabilitás határán

– Szabályozó hangolása: A kritikus periódus és erősítés felhasználásával a 7.1. táblázat alapján hangoljuk be a PID-szabályozót. Cseréljük le a *Slider Gain* blokkot PID-szabályozóra, és teszteljük a szabályozási rendszert P-, PI-, PID-szabályozóra.

Az önhangoló szabályozó vizsgálata valós időben: Az önhangoló szabályozást kemence hőmérséklet-szabályozásának keretében vizsgáljuk. A kemence a 7.5. ábrán látható, a hőmérséklet-szabályozási rendszer kapcsolási rajzát a függelék F3. ábrája mutatja.



7.5. ábra. Kemence önhangoló szabályozás tesztelésére

A kemencét a falában elhelyezett ellenállások melegítik. Az ellenállásra a feszültséget egy szilárdtest-relével kapcsoljuk. A kapcsolást impulzus szélesség modulált (PWM) jellel végezzük, így analóg módon tudunk beavatkozni a kemence hőmérséklet-szabályozásába. A hőmérsékletet egy K típusú hőelemmel mérjük.

A kemence szabályozását egy HAGA PID-szabályozóval végezzük. A szabályozó egy PIC 16-os családú mikrovezérlővel van megoldva: A mikrovezérlő szoftverében egy önhangoló PID-algoritmus van implementálva, ugyanakkor a szabályozó egy interaktív menürendszert is szolgáltat a felhasználónak.

Feladatok:

- Kösse be a szabályozóba az érzékelőt (hőelem) és a beavatkozót (szilárdtest-relé). A kemence hőmérséklet-szabályozási rendszerének a kapcsolási rajza a Függelék F3. ábráján látható.
- Ismerkedjen meg a Haga PID-szabályozó menürendszerével. Állítsa be az érzékelő típusát (K hőelem). Olvassa ki az aktuális szabályozóparamétereket.
- Indítsa el az önhangolást a szabályozóval és figyelje meg az önhangolás folyamatát.

7.4. Kérdések és feladatok

1. Dolgozza ki az önhangoló PID-szabályozó algoritmusát, ha az önhangolás nem a Ziegler–Nichols-, hanem az Oppelt-módszer alapján történik.
2. Hasonlítsa össze a Ziegler–Nichols-módszerrel hangolt szabályozási rendszer választ az Oppelt-módszerrel hangolt szabályozási rendszer válaszával.
3. Milyen más hőmérséklet-érzékelőt lehetne használnia 0–1000 °C hőmérséklet-tartományban a kemence hőmérséklet-szabályozásához?

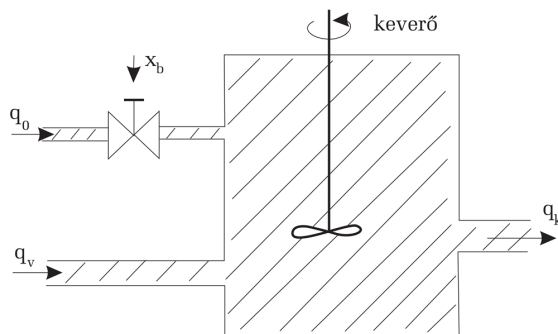
VEGYIOLDAT-KONCENTRÁCIÓ SZABÁLYOZÁSA

8.1. A gyakorlat célja

A vegyikoncentráció-szabályozás folyamatmodelljének megismerése. Szabályozótervezés előírt tranziens minőségi jellemzők alapján. A szabályozás szimulációja, eredmények kiértékelése.

8.2. Elméleti bevezető

A *folyamat modellezése*: Számos vegyipari folyamat esetén adott vízmennyiségben egy oldat koncentrációját kell konstans értéken tartani. A rendszerbe áramló folyadékok mennyiségét hozammal (egységnyi idő alatt be/kiáramló térfogattal) jellemezzük. A rendszer vázlatát a 8.1. ábra mutatja. c_0 koncentrációjú oldatot vízzel keverünk, hogy c_k koncentrációjú oldatot kapjunk. A beáramló víz hozama q_v állandó, a beáramló oldat q_0 hozamát proporcionális szeleppel szabályozzuk. A koncentrációt jellemezhetjük például egységnyi térfogatra eső oldott anyag mennyiségével.



8.1. ábra. Vegyi folyamat vázlatja modellezéshez

A rendszer bemenete a szelepszár x_b pozíciója, kimenete a kapott oldat c_k koncentrációja. A beáramló oldat mennyisége a szelepszár pozíciójával arányos:

$$q_0 = K_B x_b. \quad (8.1)$$

Feltételezve, hogy a tartályban a folyadékszint állandó, a kiáramló oldat mennyisége a beáramló oldat és a víz mennyiségének összege:

$$q_k = q_0 + q_v. \quad (8.2)$$

Elemi idő alatt $q_0 c_0 dt$ tömegű oldat kerül a V térfogatú tartályba, ugyanakkor $q_k c_k dt$ tömegű oldott anyag hagyja el azt. Feltételezve, hogy a vízben az oldat koncentrációja 0, a koncentrációváltozás:

$$dc_k = \frac{(q_0 c_0 + q_v \cdot 0) - q_k c_k}{V} dt. \quad (8.3)$$

A folyamat kimenetének dinamikája:

$$\frac{dc_k}{dt} + \frac{q_k}{V} c_k = \frac{c_0 K_B}{V} x_B. \quad (8.4)$$

Ha feltételezzük, hogy $q_0 \ll q_v$, akkor $q_k \approx q_v$ állandó, tehát paraméternek tekinthetjük. Alkalmazva a Laplace-transzformáltat, a rendszer átviteli függvénye:

$$\begin{aligned} sc_k(s) + \frac{q_v}{V} c_k(s) &= \frac{c_0 K_B}{V} x_B(s) \\ H(s) = \frac{c_k(s)}{x_B(s)} &= \frac{K}{Ts + 1} \\ K = \frac{c_0 K_B}{q_v}, \quad T &= \frac{V}{q_v}. \end{aligned} \quad (8.5)$$

Ha a folyamat beavatkozója integráló jellegű (K_α/s), például szervomotor, akkor a folyamat modellje a beavatkozóval:

$$\begin{aligned} H(s) = \frac{c_k(s)}{x_B(s)} &= \frac{K}{(Ts + 1)s} \\ K = \frac{c_0 K_B K_\alpha}{q_v}, \quad T &= \frac{V}{q_v}. \end{aligned} \quad (8.6)$$

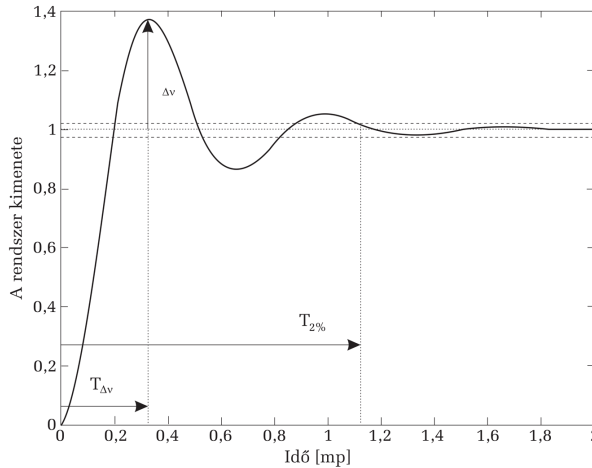
Másodfokú referenciarendszer, időtartománybeli minőségi jellemzők:
Irányítástechnikai alkalmazásoknál kiemelt jelentőségű rendszermodell a

másodfokú lengőrendszer. Az irányítás tervezésénél abból indulhatunk ki, hogy az irányított rendszer úgy viselkedjen, mint egy előírt referenciarendszer. Tipikusan ilyen rendszernek választható a másodfokú lengőrendszer:

$$H(s) = \frac{1}{T^2 s^2 + 2\xi T s + 1} \stackrel{(\omega_n = \frac{1}{T})}{=} \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}. \quad (8.7)$$

$\xi > 0$ jelöli a rendszer csillapítását, $\omega_n > 0$ a rendszer saját körfrekvenciáját.

$\xi < 1$ feltétel mellett a pólusok komplexek, ami lengő viselkedésre utal: ha a rendszer bemenete egységugrásszerű, a kimeneten csillapított, lengő választ kapunk (lásd 8.2. ábra).



8.2. ábra. Másodfokú lengőrendszer tipikus válasza egységugrás-bemenetre

A rendszer válaszának legfontosabb jellemzőit *időtartománybeli minőségi jellemzőknek* nevezzük:

1. *Túllövés*: az egységugrásra adott válasz legnagyobb pozitív irányú eltérése az egységugrástól, százalékban kifejezve. Az alábbi képlet alapján számíthatjuk:

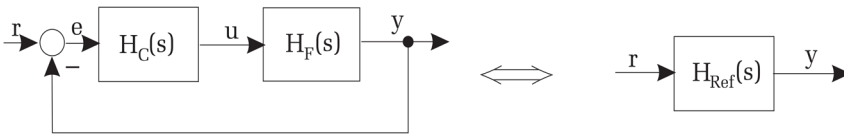
$$\Delta v = \exp\left(-\frac{\pi\xi}{\sqrt{1-\xi^2}}\right) \quad (8.8)$$

2. *Szabályozási idő*: az az időtartam, amelynek elteltével a rendszer egységugrásra adott válasza csak maximum 2%-kal tér el az egységtől. A 8.2. ábrán a 2%-os sávot a vízszintes szaggatott vonalak jelölik.

$$T_{2\%} \cong \frac{4}{\xi \cdot \omega_n}. \quad (8.9)$$

Fontos eset a $\xi = \frac{\sqrt{2}}{2}$ csillapítás, ugyanis erre az értékre a válasz túllövése $\Delta v = \exp(\pi) = 0,043 \Rightarrow \Delta v = 4,3\%$. Tehát a $\xi = \frac{\sqrt{2}}{2}$ csillapítási érték kis túllövést biztosít.

Előírt mintarendszer-alapú tervezés: Az előírt mintarendszer-alapú tervezési módszer esetében meg kell keresni azt a dinamikus rendszert, amely az előírt tranziens és állandósult állapotbeli követelményeknek eleget tesz. A mintarendszer meghatározása után a szabályozót úgy kell megválasztani, hogy a visszacsatolt zárt rendszer ugyanolyan viselkedést mutasson, mint az előírt mintarendszer (lásd 8.3. ábra).



8.3. ábra. Előírt mintarendszer-alapú tervezés

Legyen a folyamat átviteli függvénye $H_F(s)$, az előírt mintarendszer átviteli függvénye $H_{\text{Ref}}(s)$. Keressük a szabályozó $H_C(s)$ átviteli függvényét. A zárt rendszer modellje:

$$H_0(s) = \frac{H_C(s) \cdot H_F(s)}{1 + H_C(s) \cdot H_F(s)}. \quad (8.10)$$

A zárt rendszer modellje meg kell hogy egyezzen a mintarendszer átviteli függvényével $H_0(s) = H_{\text{Ref}}(s)$, így:

$$H_{\text{Ref}}(s) = \frac{H_C(s) \cdot H_F(s)}{1 + H_C(s) \cdot H_F(s)}. \quad (8.11)$$

A (8.11) összefüggés alapján a szabályozót leíró dinamikus modell automatikusan következik:

$$\begin{aligned} H_{\text{Ref}}(s) + H_{\text{Ref}}(s) \cdot H_C(s) \cdot H_F(s) &= H_C(s) \cdot H_F(s) \\ H_{\text{Ref}}(s) &= H_C(s) \cdot H_F(s)(1 - H_{\text{Ref}}(s)) \\ H_C(s) &= \frac{1}{H_F(s)} \cdot \frac{H_{\text{Ref}}(s)}{1 - H_{\text{Ref}}(s)}. \end{aligned} \quad (8.12)$$

Mintavételes rendszerek esetében, ha a mintarendszer és a folyamatmodell diszkrét átviteli függvényekkel van megadva ($H_F(z)$, $H_{\text{Ref}}(s)$), a szabályozó

modelljét ugyancsak a (8.12) formában kapjuk:

$$H_C(z) = \frac{1}{H_F(z)} \cdot \frac{H_{\text{Ref}}(z)}{1 - H_{\text{Ref}}(z)}. \quad (8.13)$$

Fokszámfeltétel előírt mintarendszer-alapú tervezés esetén: Feltevődik a kérdés, hogy a folyamat és a referenciarendszer fokszámai (pólusainak és zérusainak számai) között milyen összefüggés kell fennálljon ahhoz, hogy a szabályozó megvalósítható (kauzális) legyen. Jelölje $\text{gr}\{\cdot\}$ általában egy polinom fokszámát. Legyenek a folyamat és a mintarendszer fokszámai:

$$H_F(s) = \frac{Q_F(s)}{P_F(s)} \quad \text{gr}\{Q_F(s)\} \leq \text{gr}\{P_F(s)\} \quad (8.14)$$

$$H_{\text{Ref}}(s) = \frac{Q_{\text{Ref}}(s)}{P_{\text{Ref}}(s)} \quad \text{gr}\{Q_{\text{Ref}}(s)\} \leq \text{ge}\{P_{\text{Ref}}(s)\}. \quad (8.15)$$

A szabályozó modell fokszámát a (8.12) összefüggés alapján határozhatjuk meg:

$$\begin{aligned} H_C(s) &= \frac{Q_C(s)}{P_C(s)} = \frac{P_F(s)}{Q_F(s)} \cdot \frac{\frac{Q_{\text{Ref}}(s)}{P_{\text{Ref}}(s)}}{1 - \frac{Q_{\text{Ref}}(s)}{P_{\text{Ref}}(s)}} \\ &= \frac{P_F(s)}{Q_F(s)} \cdot \frac{Q_{\text{Ref}}(s)}{P_{\text{Ref}}(s) - Q_{\text{Ref}}(s)}. \end{aligned} \quad (8.16)$$

A (8.16) alapján látszik, hogy annak a feltétele, hogy a szabályozó kauzális legyen – $\text{gr}\{P_C(s)\} \geq \text{gr}\{Q_C(s)\}$:

$$\text{gr}\{P_{\text{Ref}} - Q_{\text{Ref}}\} - \text{gr}\{Q_{\text{Ref}}\} \geq \text{gr}\{P_F\} - \text{gr}\{Q_F\}. \quad (8.17)$$

A szabályozó megvalósíthatóságának feltétele, hogy a mintarendszer és a folyamat fokszámai között a (8.17) feltétel teljesüljön.

8.3. A mérés menete

A folyamat paramétereit: Legyenek a (8.6) modellel megadott rendszer paramétereit:

$$V = 2 \text{ m}^2$$

$$q_V = 1 \text{ l/s}$$

$$K_B = 0,1 \text{ l/rad}$$

$$K_a = 0,2 \frac{\text{rad/sec}}{\sqrt{\text{m}^3}}$$

$$C_0 = 100 \text{ kg/m}^3$$

Szabályozótervezési feladat: Tervezzünk a folyamatnak PD-szabályozót úgy, hogy egységugrásra az állandósult állapotbeli hiba nulla legyen, 20 másodperc alatt érjük el a 2% szabályozási pontosságot, a túllövés 10% legyen.

A szabályozó tervezése:

1. Határozzuk meg a szabályozó tervezéséhez alkalmazható referenciarendszert. A (8.7), (8.8) és (8.9) összefüggéseket alkalmazhatjuk:

$$\ln \Delta v = -\frac{\pi \xi}{\sqrt{1-\xi^2}} \Rightarrow \left(\frac{\ln \Delta v}{\pi}\right)^2 = \frac{\xi^2}{1-\xi^2}$$

$$\left(\frac{\ln \Delta v}{\pi}\right)^2 = \left(1 + \left(\frac{\ln \Delta v}{\pi}\right)^2\right) \xi^2$$

$$\left\{ \begin{array}{l} \omega_n = \frac{4}{\xi T_{2\%}} \\ \xi = \pm \frac{\sqrt{\left(\frac{\ln \Delta v}{\pi}\right)^2}}{1 + \left(\frac{\ln \Delta v}{\pi}\right)^2} = \pm \frac{\ln \Delta v}{\pi \sqrt{1 + \left(\frac{\ln \Delta v}{\pi}\right)^2}} \end{array} \right. \quad (8.18)$$

Vizsgáljuk Matlab környezetben az egységugrásra adott válasz alapján (*step*), hogy az így számított paraméterekkel a kapott referenciarendszer teljesíti-e a szabályozási követelményeket.

2. Mivel a folyamat tartalmaz integrátort, így garantálva egységugrásra a nulla állandósult állapotbeli hibát a szabályozási hurokban, a szabályozáshoz alkalmazzunk PD-szabályozót. Számítsuk ki a szabályozási követelményeknek megfelelő PD-szabályozó paramétereit (K_p, T_d, T):

$$H_{\text{PD}}(s) = K_P \left(1 + \frac{T_d \cdot s}{T \cdot s + 1}\right). \quad (8.19)$$

A nyílt rendszer:

$$H_N(s) = K_P \left(\frac{(T_d + T) \cdot s + 1}{T \cdot s + 1}\right) \cdot \frac{K_F}{s(T_F \cdot s + 1)}. \quad (8.20)$$

Válasszuk a deriválási időt:

$$T_F = T_d + T \Rightarrow T_d = T_F - T. \quad (8.21)$$

A zárt rendszer átviteli függvényét felírhatjuk a nyílt rendszer segítségével a (8.20) és (8.21) összefüggések alapján:

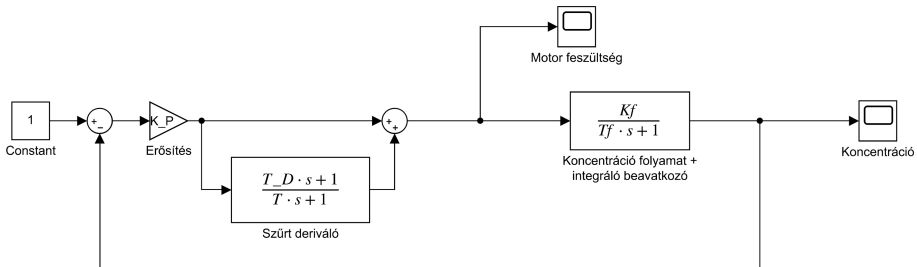
$$H_N(s) = \frac{K_P \cdot K_F}{s(T \cdot s + 1)} \quad (8.22)$$

$$H_0(s) = \frac{\frac{K_P \cdot K_F}{s(T \cdot s + 1)}}{1 + \frac{K_P \cdot K_F}{s(T \cdot s + 1)}} = \frac{K_P \cdot K_F}{T \cdot s^2 + s + K_P \cdot K_F} = \frac{\frac{K_P \cdot K_F}{T}}{s^2 + \frac{s}{T} + \frac{K_P \cdot K_F}{T}}$$

Ahhoz, hogy a zárt rendszer úgy viselkedjen, mint az előírt referencia-rendszer, a szabályozó paraméterei:

$$\begin{cases} \omega_n^2 = \frac{K_p K_f}{T} \\ \frac{1}{T} = 2\xi\omega_n \end{cases} \Rightarrow \begin{cases} T = \frac{1}{2\xi\omega_n} \\ K_p = \frac{\omega_n^2 T}{K_f} \end{cases} \quad (8.23)$$

Szimulációs vizsgálatok: Készítsük el a 8.4. ábrán látható szimulációs diagramot. A K_P , T_D , T paramétereket a (8.21), (8.23) összefüggések alapján számítsuk. Ellenőrizzük le, hogy teljesíti-e a szabályozási rendszer a követelményeket (túllövés, szabályozási idő).



8.4. ábra. A koncentrációsabályozás modellje

8.4. Kérdések és feladatok

1. Válasszunk a szabályozási követelménynek kisebb túllövést (5%). Hogyan módosul a beavatkozójel?

2. Válasszuk a szabályozási követelménynek nagyobb szabályozási időt. Hogyan módosul a beavatkozójel?
3. Mikor szükséges, hogy a szabályozó is tartalmazzon integrátort?

EGYENÁRAMÚ MOTOR MODELLEZÉSE

9.1. A gyakorlat célja

Az egyenáramú szervomotor működését leíró modell meghatározása. A modell ellenőrzése számításokkal és szimulációs vizsgálattal.

9.2. Elméleti bevezető

Az egyenáramú motor: Az egyenáramú motor napjainkban az egyik legelterjedtebb beavatkozó szerv nagy pontosságú pozicionálási feladatok megoldására. A szervomotorok fő jellemzői a kis mechanikai és elektromos időállandók (gyors dinamika), kiterjedt lineáris működési tartomány, könnyű vezérelhetőség. Használatosak például ipari robotkarok, mobilis robotok, $X - Y$ pozicionáló asztalok stb. meghajtására. Ahhoz, hogy ezekkel a motorokkal nagy pontosságú szabályozást tudjunk végezni, elengedhetetlen, hogy a motor viselkedését leíró matematikai modellt ismerjük.

Kis teljesítményű szervomotoroknál az állórész állandó mágnesből készülhet, míg a forgórész tekercselt. A forgórészre kapcsolt feszültség (U) hatására a rotor forgómozgást végez.

A modellezéshez a fizika és elektronika jól ismert alaptörvényeit használhatjuk. A Biot–Savart-törvény következményeként a motor által kifejtett forgatónyomaték (τ) arányos a rotoron átfolyó árammal (i). A Lenz-törvény következményeként a rotorban visszaindukált feszültség (e) pedig arányos a rotor fordulatszámával (ω).

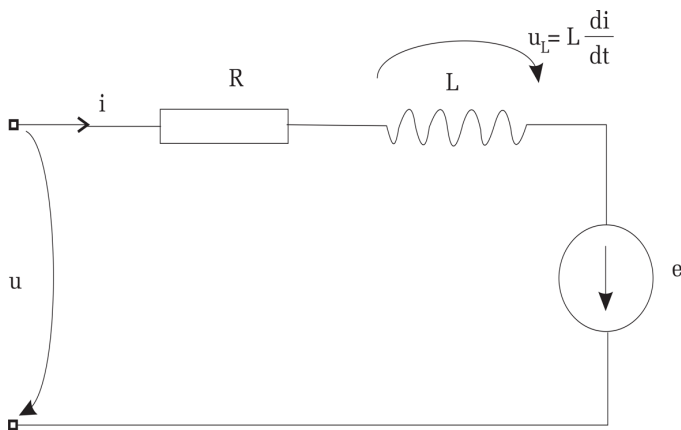
$$\begin{cases} \text{Lenz :} & e = c_1 \omega \\ \text{Biot-Savart :} & \tau = c_2 \cdot i_n \end{cases}$$

ahol c_1 , c_2 konstansok.

A rotor egy sorba kötött L induktivitású ideális tekercssel és R ellenállással modellezhető. Figyelembe véve a külső rákapcsolt feszültséget (U) és

a Lenz-törvényt, a motor elektromos egyenlete:

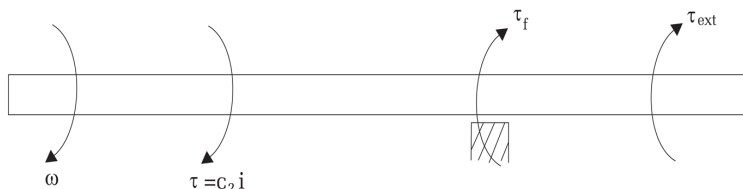
$$u = i \cdot R + L \frac{di}{dt} + e. \quad (9.1)$$



9.1. ábra. A rotor mint elektromos áramkör

A mechanikai mozgásegyenlet felírásához ismernünk kell a rotorra ható forgatónyomatékokat (9.2. ábra):

- a motor által kifejtett nyomaték, amely arányos az árammal,
- a motorra ható külső nyomaték, amit a motor által mozgatott munkagép fejt ki (τ_{ext}),
- a motor belsejében fellépő súrlódási erők (a rotor felfüggesztése és a szénkefék miatt). Ezt Coulomb súrlódási modellel írhatjuk le. ($F_f = -F_C \text{sign}(\omega)$, ahol F_C a Coulomb súrlódási tényező).



9.2. ábra. A rotorra ható erők

A Newton-mozgástörvény értelmében:

$$J_R \frac{d\omega}{dt} = c_2 \cdot i - F_f - \tau_{\text{ext}} \quad (9.2)$$

ahol J_R a rotor inerciája.

A motor dinamikus modellezéséhez a mechanikai (9.2) és elektromos viselkedést (9.1) leíró egyenleteket használjuk:

$$\begin{cases} L \frac{di}{dt} + i \cdot R + c_1 \cdot \omega = u \\ J_R \frac{d\omega}{dt} = c_2 \cdot i - F_f - \tau_{\text{ext}} \\ \frac{d\alpha}{dt} = \omega \end{cases} \quad (9.3)$$

α jelöli a rotor szögpozícióját.

Az egyenáramú motor irányításra alkalmazható állapotterezes modellje és átviteli függvénye: Amint láthattuk, a motor viselkedését lineáris differenciálegyenletekkel írhatjuk le, ezért a motor sebesség- vagy pozíciószabályozásának tervezéséhez a lineáris rendszerek irányítástechnikáját alkalmazhatjuk. Ahhoz, hogy a lineáris szabályozótervezési algoritmusokat alkalmazzni tudjunk, célszerű a motor modelljét állapotterezes alakba átírni vagy meghatározni a motor átviteli függvényét.

Általában az *állapotterezes modell* alakja: $\begin{cases} \dot{x} = Ax + Bu \\ Y = Cx + Du \end{cases}$, ahol x az állapotok vektora, u a bemenetek vektora, az A, B, C, D pedig a rendszer-mátrixok, amelyek tartalmazzák a motor paramétereit.

Az egyenáramú szervomotor esetében definiáljuk az alábbi állapot, valamint bemeneti vektort:

$$x = \begin{pmatrix} i \\ \omega \\ \alpha \end{pmatrix} \quad u = \begin{pmatrix} F_f \\ \tau_{\text{ext}} \\ u \end{pmatrix}. \quad (9.4)$$

Feltételezzük, hogy csak a motor szögelfordulása mérhető. Ebben az esetben a (9.3) egyenlet az alábbi alakba írható át:

$$\begin{pmatrix} \frac{di}{dt} \\ \frac{d\omega}{dt} \\ \frac{d\alpha}{dt} \end{pmatrix} = \underbrace{\begin{pmatrix} -\frac{R}{L} & -\frac{c_1}{L} & 0 \\ \frac{c_2}{J_R} & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}}_A \begin{pmatrix} i \\ \omega \\ \alpha \end{pmatrix}$$

$$\begin{aligned}
 & + \underbrace{\begin{pmatrix} \frac{1}{L} & 0 & 0 \\ 0 & -\frac{1}{J_R} & -\frac{1}{J_R} \\ 0 & 0 & 0 \end{pmatrix}}_B \begin{pmatrix} u \\ F_f \\ \tau_{\text{ext}} \end{pmatrix} \\
 & y = \underbrace{\begin{pmatrix} 0 & 0 & 1 \end{pmatrix}}_C \begin{pmatrix} i \\ \omega \\ \alpha \end{pmatrix} + \underbrace{\begin{pmatrix} 0 & 0 & 1 \end{pmatrix}}_D.
 \end{aligned} \tag{9.5}$$

A (9.5) egyenletek a motor állapotteres modelljét adják meg. Vegyük észre, hogy a súrlódási erő meg a külső nyomaték zavaró bemenetekként jelennek meg a modellben.

Az átviteli függvény meghatározására a súrlódási erő meg a külső nyomaték hatását elhanyagoljuk ($F_f = \tau_{\text{ext}} = 0$). Az átvitelt az u vezérlőjelről a szögsebességre (ω) határozzuk meg. Induljunk ki a (9.2) mozgásegyenletből:

$$J_R \frac{d\omega}{dt} = c_2 \cdot i. \tag{9.6}$$

Ezt differenciálva és alkalmazva a (9.1) egyenletet:

$$\begin{aligned}
 J_R \frac{d^2\omega}{dt^2} &= c_2 \frac{di}{dt} = c_2 \left(-\frac{R}{L}i - \frac{c_1}{L}\omega + \frac{1}{L}u \right) = c_2 \left(-\frac{R}{L} \frac{J_R}{c_2} \frac{d\omega}{dt} - \frac{c_1}{L}\omega + \frac{1}{L}u \right) \\
 & \tag{9.7} \\
 J_R \frac{d^2\omega}{dt^2} + J_R \frac{R}{L} \frac{d\omega}{dt} + \frac{c_1 c_2}{L} \omega &= \frac{1}{L} u \\
 \tag{9.8} \\
 \frac{d^2\omega}{dt^2} + \frac{R}{L} \frac{d\omega}{dt} + \frac{c_1 c_2}{L J_R} \omega &= \frac{1}{L J_R} u.
 \end{aligned}$$

A Laplace-transzformáltat a fenti differenciálegyenletre alkalmazva kapjuk a motor átviteli függvényét:

$$H(s) = \frac{\omega(s)}{u(s)} = \frac{\frac{1}{L J_R}}{s^2 + \frac{R}{L}s + \frac{c_1 c_2}{L J_R}}. \tag{9.9}$$

9.3. A mérés menete

A motor paramétereit: A MAXON cég nagy pontosságot és jó szabályozási jellemzőket megkövetelő alkalmazásokhoz gyárt kis teljesítményű szervomotorokat. Vegyük példának az A-max 32 24V/15W¹ típusú motort.

A motor paramétereit SI mértékegységekben:

- A rotor ellenállása (*terminal resistance*): $R = 7,13 \Omega$
- A rotor induktivitása (*terminal inductance*): $L = 1,05 \cdot 10^{-3} \text{ H}$
- A sebességállandó inverze (*1/speed constant*):

$$c_1 = 1/250 [\text{V/rpm}] = 1/26,16 \left[\frac{\text{V}}{\text{rad/sec}} \right].$$

Részletezve a mértékegység-átszámolásokat:

$$1 \text{ rpm} = 1 \text{ ford}/60 \text{ sec};$$

$$1 \text{ ford} \dots 2\pi \text{ rad};$$

$$\frac{2\pi \text{ rad}}{60 \text{ sec}} \dots 1 \text{ rpm};$$

$$x \frac{\text{rad}}{\text{sec}} \dots 250 \text{ rpm};$$

- A nyomatékállandó (*torque constant*): $c_2 = 38,2 \cdot 10^{-3} [\text{Nm/A}]$.
- A rotor inerciája (*rotor inertia*): $J_R = 41,9 \text{ g} \cdot \text{cm}^2 = 41,9 \cdot 10^{-3} \cdot 10^{-4} \text{ kgm}^2 = 41,9 \cdot 10^{-7} \text{ kgm}^2$.

Az állapotterezes modell meghatározása: A (9.5) modellben az A, B állapotmátrixokban az alábbi paraméterek jelennek meg:

$$\frac{R}{L} = \frac{7,13}{1,05 \cdot 10^{-3}} = 6,79 \cdot 10^3$$

$$\frac{c_1}{L} = \frac{1/26,16}{1,06 \cdot 10^{-3}} = 36,4$$

$$\frac{c_2}{J_R} = \frac{38,2 \cdot 10^{-3}}{41,9 \cdot 10^{-7}} = 9,11 \cdot 10^3$$

$$\frac{1}{L} = \frac{1}{1,05 \cdot 10^{-3}} = 0,95 \cdot 10^{-3}$$

$$\frac{1}{J_R} = \frac{1}{41,9 \cdot 10^{-7}} = 23,8 \cdot 10^4.$$

A motor belsejében fellépő *Coulomb súrlódási együttható* nem katalógusadat, azonban számolással meghatározhatjuk. Vegyük észre, hogy adott

1 <https://maxonmotor.com/maxon/view/product/236659>

a rotor tekercsén külső terhelés nélkül folyó áram (*no load current*). Stacionárius állapotban a (9.2) egyenlet: $c_2 \cdot i_{\text{NO_LOAD}} = \underbrace{FC}_0 + \underbrace{\tau_{\text{ext}}}_0 + \underbrace{J_R \frac{d\omega}{dt}}_0$.

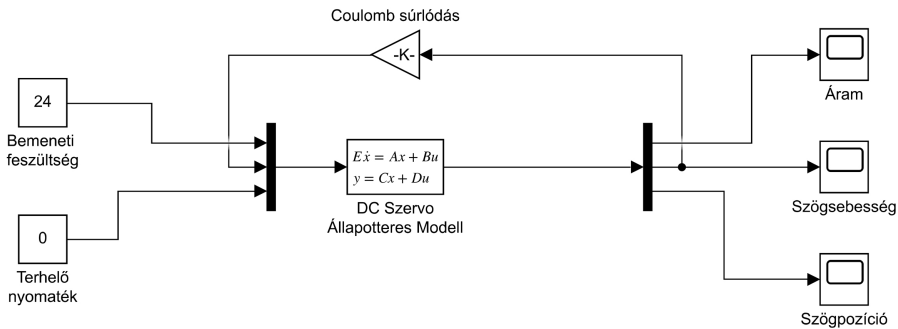
Innen a Coulomb súrlódási együttható:

$$F_f = c_2 \cdot i_{\text{NO_LOAD}} \Rightarrow F_f = 38,2 \cdot 10^{-3} \cdot 74 \cdot 10^{-3} = 2,826 \cdot 10^{-3} \text{ Nm.}$$

Modellezés Simulink környezetben: A kiszámított paraméterek alapján írjuk fel a motor állapotteres modelljét és építsük fel az alábbi Simulink diagramot, amely tartalmazza: a motor állapotteres modelljét, bemeneteket (vezérlőjel, külső nyomaték), a Coulomb súrlódási erő modelljét. A három állapotot oszcilloszkópon figyeljük meg (válasszuk a C kimeneti mátrixot egységmátrixnak). A szimulációs időnek válasszunk 0,5 mp-t.

Figyeljük meg az áram-, sebesség- és pozícióállapotokat.

A modell ellenőrzéséhez válasszuk a külső terhelő nyomatékot zérónak. Ebben az esetben a szögsebesség kimeneten a tranziens lejártá után meg kell hogy kapjuk a motorterhelés nélküli sebességét (*no load speed*), ami katalógusadat $\omega_{\text{NO_LOAD}} = 5860 \cdot \frac{2\pi}{60} = 613,38 \text{ rad/sec}$.



9.3. ábra. Simulink diagram a motor dinamikus viselkedésének vizsgálatához

9.4. Kérdések és feladatok

1. Tanulmányozzuk a (9.5) állapotteres modell irányíthatóságát és megfigyelhetőségét.

2. Határozzuk meg a motor átviteli függvényét a vezérlőjelről a szögpozíció-kimenetre.
3. Tanulmányozzuk a nem zéró külső terhelőnyomaték hatását a rotoráramra és a rotor szögsebességére.

EGYENÁRAMÚ MOTOR KASZKÁDSZABÁLYOZÁSA

10.1. A gyakorlat célja

Az egyenáramú motor modellje alapján kaszkádszabályozó tervezése. A szabályozási kör megvalósítása Simulink környezetben. A szimulációs eredmények feldolgozása.

10.2. Elméleti bevezető

Ha az irányítási folyamat fokszáma nagy, az irányítást megvalósító szabályozó is bonyolult struktúrájú lesz. Ha a folyamat sok pólust tartalmaz, a pólus-zérus kiejtést alkalmazó szabályozó sok zérust kell tartalmazzon, a szabályozó elveszítheti kauzalitását. Ilyen esetekben használhatjuk a kaszkád szabályozás elvét: a folyamatot felosztjuk több, egymással sorban levő részfolyamatra, és az összes részfolyamatra szabályozót tervezünk.

A *motor modellezése*: Az egyenáramú motorral végzett pozíciószabályozás esetén a szabályozó tervezéséhez a motor dinamikus modelljéből indulhatunk ki. A motor dinamikáját leíró egyenletek:

$$\begin{cases} L \frac{di}{dt} + i \cdot R = U - c_1 \cdot \omega \\ J \frac{d\omega}{dt} + F_v \cdot \omega = c_2 \cdot i - \tau_{\text{ext}} \\ \frac{d\alpha}{dt} = \omega \end{cases} \quad (10.1)$$

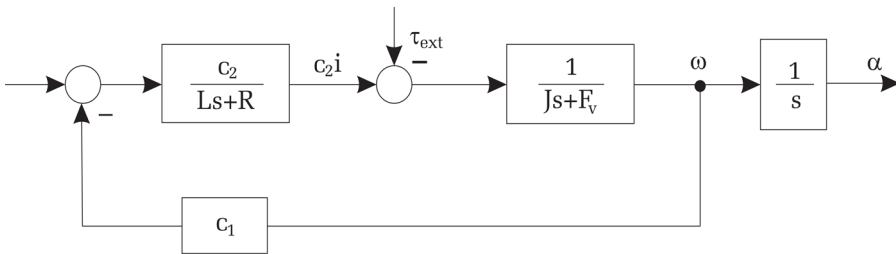
A folyamat bemenete a rotorra adott feszültség (U), kimenete a rotor szögpozíciója (α). A motor belső állapotai a rotor szögsebessége (ω) és a rotoron átfolyó áram (i). τ_{ext} a motorra ható külső nyomatékot jelöli. A motor viselkedését leíró (1) rendszer első egyenlete a rotor elektromos, a második egyenlet a rotor mechanikus viselkedését írja le. A harmadik egyenletben a szögpozíció-változás egyenlő a szögsebességgel. A paraméterek:

- R – a rotor ellenállása
- L – a rotor induktivitása
- c_1 – a rotor sebességállandójának inverze
- J – a rotor inerciája
- c_2 – a nyomatékállandó
- F_v – a viszkózus súrlódási együttható.

A (10.1) egyenletekre alkalmazva a Laplace-transzformáltat, a következőt kapjuk:

$$\begin{cases} \frac{i(s)}{U(s) - c_1 \cdot \omega(s)} = \frac{1}{L \cdot s + R} \\ \frac{\omega(s)}{c_2 \cdot i(s) - \tau_{\text{ext}}(s)} = \frac{1}{J \cdot s + F_v} \\ \frac{\alpha(s)}{\omega(s)} = \frac{1}{s} \end{cases} \quad (10.2)$$

A 10.1. ábrán a motor (10.2) összefüggés alapján kapott tömbvázlata látható.



10.1. ábra. Egyenáramú motor tömbvázlata

A pozíciósabályozási feladat megvalósításához a kaszkádszabályozási struktúrát alkalmazzuk. Habár a cél a szögpozíció előírt értékre történő szabályozása, az irányítás tervezésénél figyelembe vesszük a motor áramát és szögsebességét is. Az irányításnak teljesítenie kell a szervoszabályozási követelményeket (zérus állandósult állapotbeli pozícióhiba, gyors, túllövésmentes válasz, érzéketlenség külső zajokra, terhelésváltozásra).

A *belső áramszabályozási hurok*: Főleg nagy elektromos időállandójú (nagy L/R érték) motoroknál célszerű kialakítani az áramszabályozási hurkot. A belső áramszabályozás célja a rendszer gyorsítása. Ezt egy, a visszacsatolásban elhelyezett P szabályozóval valósíthatjuk meg. Legyen a P szabályozó erősítése K_i . Képezzük úgy a beavatkozójelet, hogy tartalmazza a külső hurok beavatkozójelet és a K_i erősítőn keresztül kialakított

áram visszacsatolást:

$$U = u - K_i i \quad (10.3)$$

ahol u tagját a beavatkozójelnek a külső sebesség és pozíciószabályozásnál határozzuk meg.

A (10.3) beavatkozójelet alkalmazva, a belső, áramszabályozási hurkot mint zárt rendszert az alábbi modell írja le:

$$\begin{aligned} H_i(s) &= \frac{i(s)}{u - c_i \omega(s)} = \frac{\frac{1}{L \cdot s + R}}{1 + K_i \cdot \frac{1}{L \cdot s + R}} \\ &= \frac{1}{L \cdot s + R + K_i} = \frac{\frac{1}{R + K_i}}{\frac{L}{R + K_i} s + 1}. \end{aligned} \quad (10.4)$$

A nem szabályozott ($K_i = 0$), nyílt rendszer ($\frac{1}{L \cdot s + R}$) időállandója L/R , erősítése $1/R$. A visszacsatolás kialakításával látható, hogy a szabályozott rendszer időállandója ($\frac{L}{R+K_i}$) kisebb lesz bármely $K_i > 0$ értékre, tehát a belső hurok gyorsabb, mint a nem szabályozott rendszer. Ugyanakkor a szabályozott rendszer erősítése ($\frac{1}{R+K_i}$) is kisebb, az u külső beavatkozójel változásánál kisebb áramugrásokra számíthatunk.

Tehát célszerű a K_i értéket minél nagyobbra választani. Az erősítés növelésének határt szab a motort vezérlő áramkör. Impulzusszélesség-modulációt alkalmazva a motor beavatkozójének változtatásához, a belső hurok időállandója nagyobb kell legyen, mint a feszültségbemenetre alkalmazott impulzusszélesség-modulált jel periódusa.

Mivel az áramszabályozó csak egy erősítőt tartalmaz, akár elektronikus áramkörrel (műveleti erősítővel) is megvalósítható.

A külső sebesség és pozíciószabályozási hurok tervezése: A külső hurok irányítási algoritmusának az alábbi követelményeket kell megvalósítania:

- egységugrásra nulla állandósult állapotbeli hiba
- 0% túllövés
- gyors válasz
- külső zajok, bemenetre ható terhelés hatásának elnyomása
- nem modellezett, a mechanikai rész rugalmas alakváltozásai miatt fellépő, mechanikai rezgések elkerülése.

A követelmények teljesítéséhez a referenciamodell alapú tervezést alkalmazzuk. Legyen a referenciarendszer (előírt mintarendszer):

$$H_{0\text{ref}}(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}. \quad (10.5)$$

A referenciarendszer erősítése 1, tehát egységugrásra nulla állandósult állapotbeli hibát biztosít.

Közismert, hogy ha a ξ csillapítás értéke 1-nél nagyobb, a rendszer válasza aperiodikus, tehát nem tartalmaz túllövést. Ugyanakkor a leggyorsabb aperiodikus választ $\xi = 1$ értékre érjük el. Tehát a ξ paraméterpértéke 1-nél nagyobb, de 1-hez közeli kell hogy legyen.

Legyen a motorral meghajtott mechanikai rendszer sajátrezgéseinek frekvenciája ω_M . Ahhoz, hogy a szabályozó által kiszámított beavatkozási jel ne gerjessze a mechanikai rész rezgéseit, a referenciamodell saját frekvenciáját $\omega_n \leq 0,5 \cdot \omega_{0M}$ összefüggés alapján kell megválasztani.

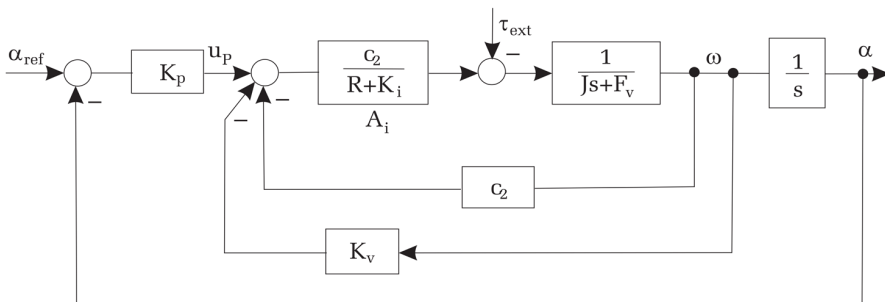
A jó zajelnyomás biztosítására a szabályozónak minél nagyobb erősítést kell választani. Számítsuk a külső szabályozási hurok beavatkozási jelét az alábbi formában:

$$u = K_P(\alpha_{\text{ref}} - \alpha) - K_V \cdot \dot{\alpha}. \quad (10.6)$$

A belső áramszabályozási hurok időállandója nagy K_i értékek esetén elhanyagolható a külső hurok időállandói mellett ($\frac{L}{R+K_i} \cong 0$). A belső áramszabályozási hurok sorban a c_2 taggal ideális erősítővel közelíthető meg:

$$A_i = \frac{c_2}{R + K_i}. \quad (10.7)$$

A visszacsatolt külső hurok tömbvázlata a 10.2. ábrán látható.



10.2. ábra. Külső sebesség és pozíciószabályozási hurok

A 10.2. ábra alapján a belső sebességszabályozási hurok átviteli függvénye:

$$H_\omega(s) = \frac{\omega(s)}{u_p(s)} = \frac{\frac{A_i}{J \cdot s + F_v}}{1 + \frac{A_i(c_1 + K_v)}{J \cdot s + F_v}} = \frac{A_i}{J \cdot s + F_v + A_i(c_1 + K_v)}. \quad (10.8)$$

A teljes szabályozási kör átvitelét leíró modell:

$$\begin{aligned} H_0(s) &= \frac{\alpha(s)}{\alpha_{\text{ref}}(s)} = \frac{\frac{K_p}{s} \cdot \frac{A_i}{J \cdot s + F_v + A_i(c_2 + K_v)}}{1 + \frac{K_p}{s} \cdot \frac{A_i}{J \cdot s + F_v + A_i(c_2 + K_v)}} \\ &= \frac{K_p \cdot A_i}{J \cdot s^2 + (F_v + A_i(c_1 + K_v)) \cdot s + K_p \cdot A_i} \\ H_0(s) &= \frac{\frac{K_p \cdot A_i}{J}}{s^2 + \frac{(F_v + A_i(c_2 + K_v))}{J} \cdot s + \frac{K_p \cdot A_i}{J}}. \end{aligned} \quad (10.9)$$

Mivel a zárt rendszer modellje meg kell hogy feleljen a referenciarendszer modelljének ($H_0(s) = H_{0\text{ref}}(s)$), a (10.5) és (10.9) összefüggések alapján kapjuk:

$$\begin{cases} \omega_n^2 &= \frac{K_p \cdot A_i}{J} \\ 2\xi\omega_n &= \frac{F_v + A_i(c_2 + K_v)}{J}. \end{cases} \quad (10.10)$$

A (10.10) összefüggés alapján kapjuk az irányítási algoritmus paramétereit.

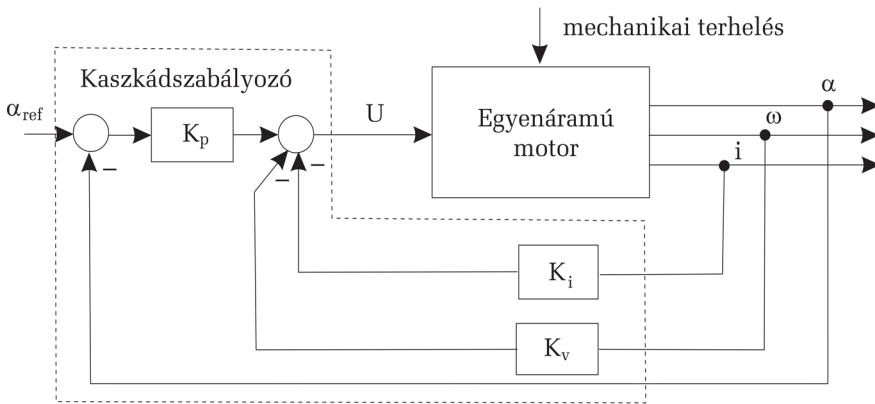
$$\begin{cases} K_p &= \frac{J \cdot \omega_n^2}{A_i \left(\leq \frac{J}{A_i} \cdot (0.5 \cdot \omega_{0M})^2 \right)} \\ K_v &= \frac{2\xi\omega_n J - F_v}{A_i} - c_1. \end{cases} \quad (10.11)$$

Látszik, hogy a K_p erősítés értéke nem lehet tetszőlegesen nagy, mivel ez gerjesztené a mechanikai rendszer sajátfrekvenciáját. Amennyiben a proporcionális zajelnyomás nem elégséges a kaszkádszabályozónak a pozícióhibát figyelembe vevő ágában, a proporcionális tag mellé integráló tagot

kell elhelyezni, és ezt is figyelembe kell venni a tervezésnél. A (10.6) és (10.3) összefüggések alapján következik az irányítás végső formája:

$$U = K_p(\alpha_{\text{ref}} - \alpha) - K_v \cdot \omega - K_i i. \quad (10.12)$$

A kapott irányítási algoritmus a kaszkádszabályozás elvét kihasználva teljesíti az előírt követelményeket, ugyanakkor nagyon egyszerű struktúrájú. Nem tartalmaz dinamikus (integráló, deriváló) elemeket, megvalósítása akár analóg, akár mintavételes módon egyszerű. Mivel csak három erősítéssparamétert tartalmaz, már működés közben is könnyen finomhangolható.



10.3. ábra. Egyenáramú motor kaszkádszabályozása

10.3. A mérés menete

A folyamat paramétereit: Legyen egy egyenáramú motor az alábbi paraméterekkel:

$$R = 7,13 \, \Omega$$

$$L = 1,05 \, \text{mH}$$

$$c_1 = 1/26,6 \, \text{V/rad/sec}$$

$$c_2 = 0,0382 \, \text{Nm/A}$$

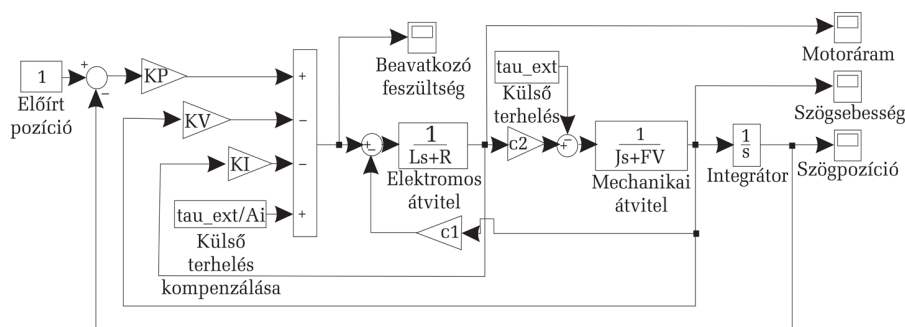
$$J = 0,0001 \, \text{Nm}^2$$

$$F_V = 0,001795 \, \text{Nm/rad/sec.}$$

A kaszkádszabályozó tervezése:

- Tervezzük a motornak kaszkádszabályozót pozíciószabályozásra, amely nulla túllövést biztosít, a zárt rendszer saját körfrekvenciája $\omega_n = 10$ rad/sec és $\tau_{\text{ext}} = 0,01$ Nm külső terhelés mellett nulla az állandósult állapotbeli hiba. Mivel a zárt rendszer egységugrásra adott válasza aperiodikus kell hogy legyen, a csillapítás értékét a (10.5) mintarendszerben egynél nagyobbra kell választani. A minél gyorsabb válasz biztosításához válasszuk a csillapítás értékét $\xi = 1,1$ -nek. Írjuk fel az előírt mintarendszer átviteli függvényét.
- Határozzuk meg a K_I erősítést a (10.4) összefüggés alapján úgy, hogy az áramszabályozási hurok válasza kétszer olyan gyors legyen, mint az eredeti rendszeré.
- Határozzuk meg a szabályozó K_p és K_v paramétereit a (10.11) összefüggés alapján.
- A konstans külső terhelés kompenzálására bővítsük ki a (10.12) beavatkozójelet egy extra előreecsatoló taggal. Ahhoz, hogy a külső terhelés hatását ki tudjuk ejteni, az U értékét additív taggal kell bővíteni: $U := U + \tau_{\text{ext}}/A_i$.

Szimulációs vizsgálatok: Építsük fel a 10.4. ábrán látható szimulációs modellt:



10.4. ábra. Egyenáramú szervomotor kaszkádszabályozásának Simulink diagramja

Vizsgáljuk meg, hogy a rendszer válasza (pozíciókimenet) egyezik-e a referenciamodell válaszával.

10.4. Kérdések és feladatok

1. Vizsgáljuk meg a rendszer válaszát $K_i = 0$ értékkel. Hasonlítsuk össze az eredeti rendszer válaszait az így kapott válaszokkal.
2. Vizsgáljuk meg a rendszer válaszát a τ_{ext}/A_i beavatkozójel kibővítése nélkül. Mekkora az állandósult állapotbeli hiba?
3. Milyen tagot kellene elhelyezni a szabályozóba ahhoz, hogy ismeretlen külső terhelést kompenzáljunk?

PI- ÉS PD-SZABÁLYOZÓK PARAMÉTEREINEK HANGOLÁSA

11.1. A gyakorlat célja

A gyakorlat célja annak a vizsgálata, hogy miként befolyásolják a szabályozóparaméterek a legfontosabb tranziens minőségi jellemzőket (szabályozási idő, túllövés). Feltételezzük, hogy az irányított folyamat egy, illetve két integrátort tartalmazó modellel írható le.

11.2. Elméleti bevezető

A *szabályozás minőségi jellemzői*: Irányítástechnikai alkalmazásoknál kiemelt jelentőségű rendszermodell a másodfokú lengőrendszer. Az irányítás tervezésénél abból indulhatunk ki, hogy az irányított rendszer úgy viselkedjen, mint egy előírt referenciarendszer. Tipikusan ilyen rendszernek választható a másodfokú lengőrendszer:

$$H(s) = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \quad (11.1)$$

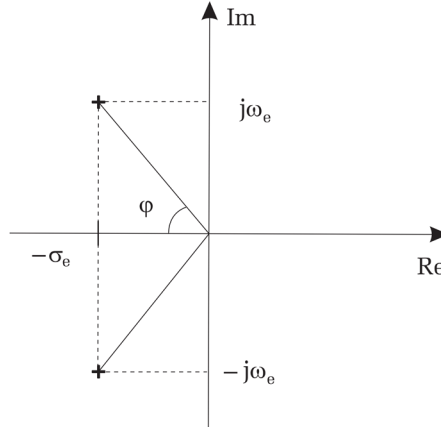
$\xi > 0$ jelöli a rendszer csillapítását, $\omega_n > 0$ a rendszer saját körfrekvenciáját.

A karakterisztikus polinom gyökei (pólusai) könnyen meghatározhatók, mivel a rendszer másodfokú:

$$\begin{aligned} s^2 + 2\xi\omega_n s + \omega_n^2 &= 0 \\ \Delta &= 4\xi^2\omega_n^2 - 4\omega_n^2 \\ s_{1,2} &= -\xi\omega_n \pm j\sqrt{1 - \xi^2}\omega_n \\ s_{1,2} &= -\sigma_e \pm j\omega_e \end{aligned} \quad (11.2)$$

$\sigma_e = \xi\omega_n$ a pólus valós részének, $\omega_e = \sqrt{1 - \xi^2}\omega_n$ a pólus komplex részének $\xi\omega_n > 0$ abszolút értékét jelöli.

A két konjugált komplex pólust komplex térben ábrázolva (11.1. ábra) látható, hogy a rendszer $\xi\omega_n > 0$ feltétel mellett stabil.



11.1. ábra. Másodfokú lengőrendszer póluselozlása

$\xi < 1$ feltétel mellett a pólusok komplexek, ami lengő viselkedésre utal: ha a rendszer bemenete egységugrásszerű, a kimeneten csillapított, lengő választ kapunk (11.2. ábra).

A rendszer válaszában legfontosabb jellemzőit *időtartomány-beli minőségi jellemzőknek* nevezzük:

1. *Túllövés*: az egységugrásra adott válasz legnagyobb pozitív irányú eltérése az egységugrástól, százalékban kifejezve. Az alábbi képlet alapján számíthatjuk:

$$\Delta x = \exp\left(-\frac{\pi\xi}{\sqrt{1-\xi^2}}\right) = \exp\left(-\frac{\pi\sigma_e}{\omega_e}\right). \quad (11.3)$$

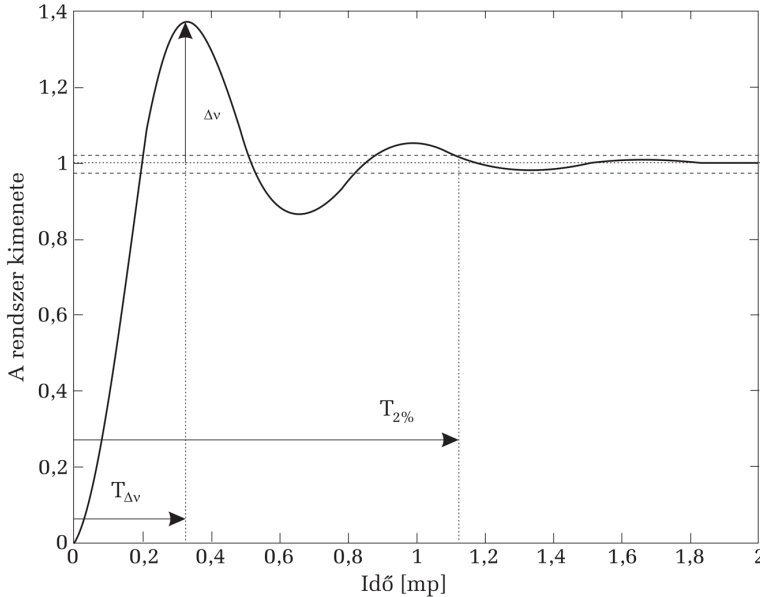
2. *Belengési idő*: a túllövés bekövetkezésének ideje.

$$T_{\Delta v} = \frac{\pi}{\omega_n \sqrt{1-\xi^2}} = \frac{\pi}{\omega_e}. \quad (11.4)$$

3. *Szabályozási idő*: az az időtartam, amelynek elteltével a rendszer egységugrásra adott válasza csak maximum 2%-kal tér el az egységtől. A 11.2. ábrán a 2%-os sávot a vízszintes szaggatott vonalak jelölik.

$$T_{2\%} \cong \frac{4}{\xi \cdot \omega_n} = \frac{4}{\sigma_e}. \quad (11.5)$$

Fontos eset a $\xi = \frac{\sqrt{2}}{2}$ csillapítás, ugyanis erre az értékre a válasz túllövése $\Delta v = \exp(-\pi) = 0.043 \Rightarrow \Delta v = 4.3\%$. Tehát a $\xi = \frac{\sqrt{2}}{2}$ csillapítási érték kis túllövést biztosít.



11.2. ábra. Másodfokú lengőrendszer tipikus válasza egységugrás-bemenetre

Másodfokú lengőrendszer pólusainak meghatározása időtartománybeli minőségi jellemzők alapján: Az irányítás tervezésekor követelményként adottak az időtartománybeli minőségi jellemzők vagy azok korlátai. Így a tervezés fontos lépése a rendszer modelljének meghatározása, amelynek válasza teljesíti az előírt jellemzőket.

Legyenek adottak a következők:

- I. maximálisan megengedett túllövés: $\Delta v \leq \Delta v_{MAX}$
- II. maximálisan megengedett belengési idő: $T_{\Delta v} \leq T_{\Delta v_{MAX}}$
- III. maximális szabályozási idő: $T_{2\%} \leq T_{2\%_{MAX}}$
- IV. maximálisan megengedett saját körfrekvencia (maximálisan megengedett nagyfrekvenciás lengés): $\omega_n \leq \omega_{n_{MAX}}$.

Az I. feltételből következik:

$$\begin{aligned}
 \Delta v = \exp\left(-\frac{\pi\sigma_3}{\omega_e}\right) \leq \Delta v_{\text{MAX}} &\Rightarrow -\frac{\pi\sigma_e}{\omega_e} \leq \ln \Delta v_{\text{MAX}} \\
 \Rightarrow \frac{\sigma_e}{\omega_e} \geq -\frac{\ln \Delta v_{\text{MAX}}}{\pi} &\Rightarrow \frac{\omega_e}{\sigma_e} \leq -\frac{\pi}{\ln \Delta v_{\text{MAX}}} \\
 \Rightarrow \text{tg}\left(\frac{\omega_e}{\sigma_e}\right) \leq \text{tg}\left(-\frac{\pi}{\ln \Delta v_{\text{MAX}}}\right) &\text{ vagyis} \\
 \text{tg}\left(\frac{\omega_e}{\sigma_e}\right) \leq \text{tg}\left(\frac{\omega_e}{\sigma_e}\right)_{\text{MAX}} &
 \end{aligned} \tag{11.6}$$

Ugyancsak az I. feltételből következik:

$$\exp\left(-\frac{\pi\xi}{\sqrt{1-\xi^2}}\right) \leq \Delta v_{\text{MAX}} \Rightarrow \xi \geq \xi_{\text{MIN}} \left(= \frac{1}{\sqrt{1+\pi^2/\ln^2 \Delta v_{\text{MAX}}}}\right). \tag{11.7}$$

A II. feltételből következik:

$$\frac{\pi}{\omega_e} \leq T_{\Delta v_{\text{MAX}}} \Rightarrow \omega_e \geq \omega_{e_{\text{MIN}}} = \frac{\pi}{T_{\Delta v_{\text{MAX}}}}. \tag{11.8}$$

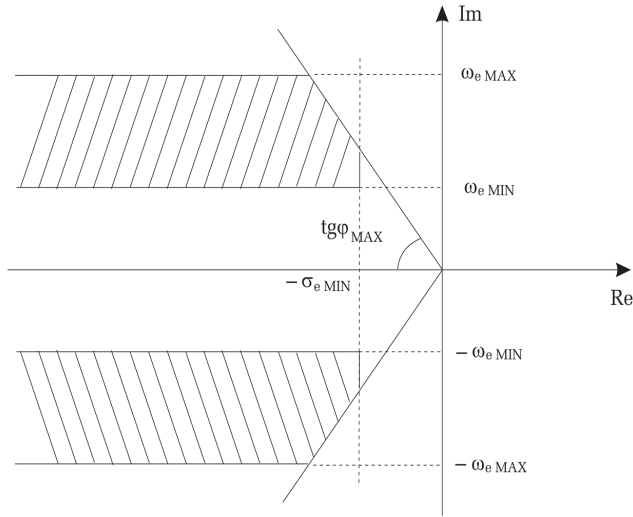
A III. feltételből következik:

$$\frac{4}{\sigma_e} \leq T_{2\%_{\text{MAX}}} \Rightarrow \sigma_e \geq \sigma_{e_{\text{MIN}}} = \frac{4}{T_{2\%_{\text{MAX}}}}. \tag{11.9}$$

Felhasználva a IV. feltételt kapjuk:

$$\begin{aligned}
 \omega_e = \sqrt{1-\xi^2} \cdot \omega_n &\leq \sqrt{1-\xi_{\text{MIN}}^2} \cdot \omega_{n_{\text{MAX}}} \\
 \omega_e \leq \omega_{e_{\text{MAX}}} &= \sqrt{1-\xi_{\text{MIN}}^2} \cdot \omega_{n_{\text{MAX}}}.
 \end{aligned} \tag{11.10}$$

A fenti összefüggések alapján a rendszer pólusait egyértelműen behatárolhatjuk a komplex térben. A 11.3. ábrán látható sávokon belül bárhonnán választjuk a konjugált komplex pólusokat, teljesíteni fogják az előírt követelményeket. A pólusok alapján pedig könnyen megkaphatjuk a megtervezett lengőrendszerünk csillapítását és saját frekvenciáját.



11.3. ábra. A pólusok helye a dinamikus minőségi jellemzők egyidejű előírása esetén

11.3. A mérés menete

PD-szabályozás hatása a tranziens viselkedésre dupla integrátor jellegű folyamatok esetén: Legyen egy mechanikai rendszer mozgását leíró modell:

$$m\ddot{x} = u \quad (11.11)$$

ahol m a mozgó test tömege, x a mozgó test elmozdulása, u a beavatkozóerő.

Számítsa a beavatkozójelet egy PD-szabályozó:

$$u = K_p(x_{\text{ref}} - x) - K_D\dot{x} \quad (11.12)$$

ahol x_{ref} a konstans előírt pozíció (tehát $\dot{x}_{\text{ref}} = 0$), K_P a proporcionális tag erősítése, K_D a deriváló tag erősítése.

Amennyiben $x_{\text{ref}} = 0$, akkor a szabályozott rendszer dinamikája:

$$m\ddot{x} + K_D\dot{x} + K_Px = 0. \quad (11.13)$$

A másodfokú lengőrendszer esetén használt jelöléseket alkalmazva:

$$\ddot{x} + 2\xi\omega_n\dot{x} + \omega_n^2x = 0. \quad (11.14)$$

A fenti összefüggésekből számítható a szabályozott rendszer csillapítása és saját körfrekvenciája az alábbi egyenletek alapján:

$$2\xi\omega_n = \frac{K_D}{m} \quad (11.15)$$

$$\omega_n^2 = \frac{K_P}{m}. \quad (11.16)$$

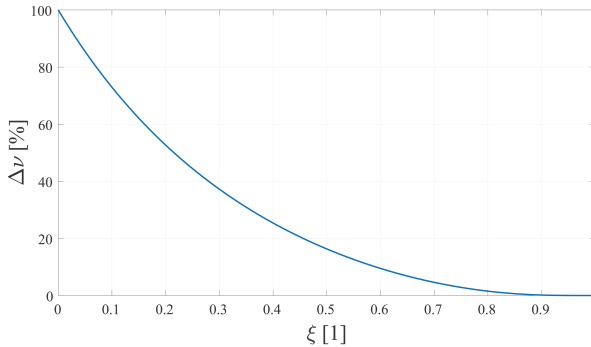
Feltételezzük, hogy a csillapítás értéke 0 és 1 között van. Innen számítható a szabályozott rendszer túllövése

$$\Delta v = \exp\left(-\frac{\pi\xi}{\sqrt{1-\xi^2}}\right), \quad \xi = \frac{K_D}{2m\omega_n} = \frac{K_D}{2\sqrt{mK_P}} \quad (11.17)$$

és szabályozási ideje

$$T_{2\%} \cong \frac{4}{\xi \cdot \omega_n} = \frac{8m}{K_D}. \quad (11.18)$$

A túllövés és a csillapítás közötti összefüggést a 11.4. ábra mutatja:



11.4. ábra. A túllövés a csillapítás függvényében

A fenti összefüggésekből látszik, hogy PD-szabályozás esetén:

- Ha növeljük a proporcionális tag erősítését, csökken a csillapítás, nő a túllövés.
- Ha növeljük a deriváló tag erősítését, nő a csillapítás, csökken a túllövés.
- A proporcionális tagnak nincs számottevő hatása a szabályozási időre.
- A deriváló tag erősítésének növelésével csökken a szabályozási idő.

PI-szabályozás integrátor jellegű folyamatok esetén: Legyen a mechanikai rendszer dinamikáját leíró modell:

$$m\dot{v} = u \quad (11.19)$$

ahol v a mozgó test sebessége, u a beavatkozóerő.

Számítsa a beavatkozójelet egy PI-szabályozó:

$$u = K_P(v_{\text{ref}} - v) + K_I \int_0^t (v_{\text{ref}} - v) d\tau \quad (11.20)$$

ahol v_{ref} az előírt sebesség, K_P a proporcionális tag erősítése, K_I az integráló tag erősítése.

Amennyiben $v_{\text{ref}} = 0$, akkor a szabályozott rendszer dinamikája:

$$m\ddot{v} + K_P\dot{v} + K_I v = 0. \quad (11.21)$$

A szabályozott rendszer csillapítása és saját körfrekvenciája:

$$2\xi\omega_n = \frac{K_P}{m} \quad (11.22)$$

$$\omega_n^2 = \frac{K_I}{m}. \quad (11.23)$$

Feltételezzük, hogy a csillapítás értéke 0 és 1 között van. Innen számítható a szabályozott rendszer túllövése

$$\Delta v = \exp\left(-\frac{\pi\xi}{\sqrt{1-\xi^2}}\right), \quad \xi = \frac{K_P}{2m\omega_n} = \frac{K_P}{2\sqrt{mK_I}} \quad (11.24)$$

és szabályozási ideje

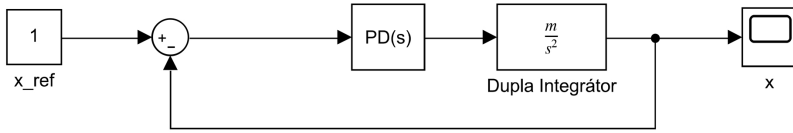
$$T_{2\%} \cong \frac{4}{\xi \cdot \omega_n} = \frac{8m}{K_P}. \quad (11.25)$$

A fenti összefüggésekből látszik, hogy PI-szabályozás esetén:

- Ha növeljük a proporcionális tag erősítését, nő a csillapítás, csökken a túllövés.
- Ha növeljük az integráló tag erősítését, csökken a csillapítás, nő a túllövés.
- Az integráló tagnak nincs számottevő hatása a szabályozási időre.
- A proporcionális tag erősítésének növelésével csökken a szabályozási idő.

Szimulációs vizsgálatok:

– Készítse el a PD-szabályozás szimulációs modelljét Simulink környezetben a 11.5. ábra alapján.



11.5. ábra. PD-szabályozás szimulációs diagram

– $m = 1$, $K_D = 1$ mellett olvassa le a szimulációs modell *Scope* eleméről a túllövést és a szabályozási időt, és töltsse ki az alábbi táblázatot: – $m = 1$,

11.1. táblázat. K_P hatása a tranziens minőségi jellemzőkre (PD-szabályozás)

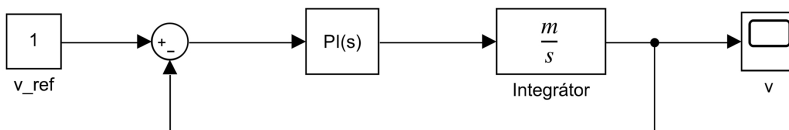
K_P	1	10	50	100
Δ_v				
$T_{2\%}$				

$K_P = 1$ mellett olvassa le a szimulációs *Scope* eleméről a túllövést és a szabályozási időt, és töltsse ki az alábbi táblázatot:

11.2. táblázat. K_D hatása a tranziens minőségi jellemzőkre (PD-szabályozás)

K_D	0,1	0,25	0,5	1
Δ_v				
$T_{2\%}$				

– Készítse el a PI-szabályozás szimulációs modelljét Simulink környezetben a 11.6. ábra alapján.



11.6. ábra. PI-szabályozás szimulációs diagram

– $m = 1$, $K_P = 1$ mellett olvassa le a szimulációs modell *Scope* eleméről a túllövést és a szabályozási időt, és töltsse ki az alábbi táblázatot:

11.3. táblázat. K_I hatása a tranziens minőségi jellemzőkre (PI-szabályozás)

K_I	0,1	0,25	0,5	1
Δ_v				
$T_{2\%}$				

– $m = 1$, $K_I = 1$ mellett olvassa le a szimulációs modell *Scope* eleméről a túllövést és a szabályozási időt, és töltsse ki az alábbi táblázatot:

11.4. táblázat. K_P hatása a tranziens minőségi jellemzőkre (PI-szabályozás)

K_P	1	10	50	100
Δ_v				
$T_{2\%}$				

– Ábrázolja grafikusán a túllövést és szabályozási időt a szabályozó-paraméterek függvényében mind a négy esetben.

11.4. Kérdések és feladatok

1. Egészítse ki az irányított folyamat modelljét egy lineáris csillapító taggal, tehát az irányított rendszer modellje: $m\ddot{x} + K_f\dot{x} = u$. Milyen hatással van a $K_f > 0$ paraméter a szabályozás minőségi jellemzőire?
2. Egészítse ki az irányított folyamat modelljét egy lineáris rugó hatással, tehát az irányított rendszer modellje: $m\ddot{x} + K_r x = u$. Milyen hatással van a $K_r > 0$ paraméter a szabályozás minőségi jellemzőire?
3. Sebességszabályozás esetén alkalmazzon PI-szabályozó helyett PID-szabályozót. Milyen hatása van a K_D paraméternek a szabályozás minőségi jellemzőire?

ELŐRECSATOLÁST ALKALMAZÓ SZABÁLYOZÁS

12.1. A gyakorlat célja

Az előreccsatolást alkalmazó szabályozók tervezésének és megvalósításának tanulmányozása. Az előreccsatolást alkalmazó szabályozás összehasonlítása az integráló szabályozással. Előreccsatoló szabályozó tervezés gravitációkompenzálásra egyszerű repülő járműmodell esetén.

12.2. Elméleti bevezető

Az irányítási rendszerek célja annak biztosítása, hogy a szabályozott folyamat az elvárt módon viselkedjen (a kimenete elérje az előírt értéket előírt tranziensekkel), valamint az, hogy a külső zavarok hatása ne vagy csak kismértékben befolyásolja a szabályozott kimenetet (zavarok hatásának kompenzálása).

Ezt a két feladatot általában visszacsatolást alkalmazó szabályozó segítségével oldjuk meg (például PID-irányítást alkalmazva). Az integráló tag a szabályozóban képes kompenzálni az ismeretlen, konstans bemeneti zaj hatását, a PID-szabályozó paramétereit pedig referenciamodell alapú tervezéssel határozhatjuk meg.

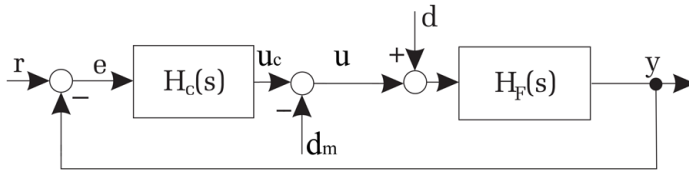
Az előreccsatolás alkalmazásával a két feladatot (modellkövetés, valamint zavarelnyomás) kettéválaszthatjuk. A módszer előnye, hogy hatékonyabb szabályozást kapunk, hátránya, hogy a megvalósításához szükséges mérni a zavarhatást.

Zavarelnyomás (zavarkompenzálás) előreccsatolással

Feltételezzük, hogy a folyamat bemenetére ható zavar (d) mérhető, a mért értéke d_m .

Szabályozó által kiszámított beavatkozójelet az alábbi formában választhatjuk, lásd 12.1. ábra.

$$u = u_c - d_m. \quad (12.1)$$



12.1. ábra. Direkt zavarelnyomás

Amennyiben $d \cong d_m$, a H_c szabályozót úgy tervezhetjük, hogy nem vesszük figyelembe a d zavart.

A zavar hatása a szabályozás pontosságára: A 12.1. ábra alapján:

$$y(s) = H_F(s)(d(s) - d_m(s)) + H_F(s)H_C(s)(r(s) - y(s))$$

$$y(s) = \frac{H_F(s)}{1 + H_F(s)H_C(s)}(d(s) - d_m(s)) + \frac{H_F(s)H_C(s)}{1 + H_F(s)H_C(s)}r(s). \quad (12.2)$$

Feltételezzük, hogy a folyamat tartalmaz integrátort, a szabályozó pedig nem. A zavar- és a referenciabemenetet egységugrás-alakúnak feltételezzük: $d(s) = d/s$, $r(s) = r/s$. Legyen a szabályozó erősítése K_P . Ebben az esetben a zavar hatása a kimenetre az alábbi módon számítható:

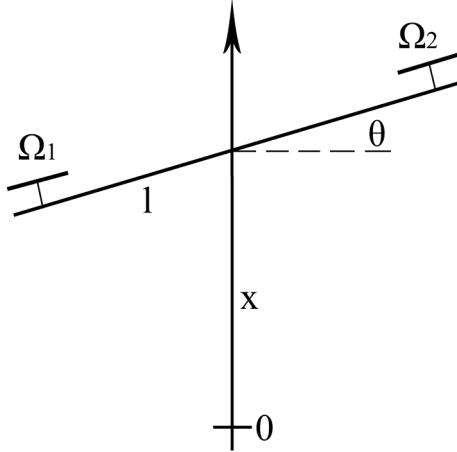
$$y = \lim_{s \rightarrow 0} \frac{\frac{K_F}{s}}{1 + \frac{K_F}{s}K_P}(d - d_m) + \frac{\frac{K_F}{s}K_P}{1 + \frac{K_F}{s}K_P}r$$

$$y = \frac{1}{K_P}(d - d_m) + r. \quad (12.3)$$

Látható, hogy az előrecsatolás megvalósításával nulla állandósult állapotbeli hiba csak akkor érhető el, ha $d = d_m$, vagyis a zavar pontosan ismert. Nagy szabályozóerősítéssel ugyanakkor a $d - d_m$ különbség hatása a szabályozás pontosságára csökkenthető.

Az irányított folyamat modellezése: A 12.2. ábrán egy repülő jármű egyszerűsített, síkbeli modellje látható. A jármű két rotorral van meghajtva.

Az ábrán $0x$ a függőleges tengely. Feltételezzük, hogy a jármű az $x0y$ síkra merőlegesen van stabilizálva.



12.2. ábra. Irányított rendszer vázlata modellezéshez

Beavatkozás: A jármű mozgását az $x0y$ síkban két rotor biztosítja, amelyek az ábrán látható módon vannak elhelyezve a járműven. Legyen l a jármű tengelyének félhossza. A rotorok szögsebessége (Ω_1, Ω_2) előírható.

A rotor által kifejtett felhajtóerő arányos a rotor szögsebesség négyzetével ($b > 0$ beavatkozó-függő konstans paraméter).

$$\begin{aligned} f_1 &= b\Omega_1^2 \\ f_2 &= b\Omega_2^2 \end{aligned} \quad (12.4)$$

A rotorerők által létrehozott felhajtóerő és forgatónyomaték:

$$\begin{aligned} F &= f_1 + f_2 \\ \tau &= lf_1 - lf_2 \end{aligned} \quad (12.5)$$

Mátrixos alakban felírva:

$$\begin{pmatrix} F \\ \tau \end{pmatrix} = b \underbrace{\begin{pmatrix} 1 & 1 \\ l & -l \end{pmatrix}}_L \begin{pmatrix} \Omega_1^2 \\ \Omega_2^2 \end{pmatrix}. \quad (12.6)$$

Mivel az L mátrix determinánsa nem nulla ($\det(L) = -2l$), invertálható. Tehát kiszámítható, hogy adott felhajtóerő és forgatónyomaték eléréséhez milyen rotorsebességek szükségesek:

$$\begin{pmatrix} \Omega_1^2 \\ \Omega_2^2 \end{pmatrix} = \frac{1}{b} L^{-1} \begin{pmatrix} F \\ \tau \end{pmatrix} = \frac{1}{b} \begin{pmatrix} \frac{1}{2} & \frac{1}{2l} \\ \frac{1}{2} & -\frac{1}{2l} \end{pmatrix} \begin{pmatrix} F \\ \tau \end{pmatrix}. \quad (12.7)$$

Figyelembe véve, hogy a beavatkozójelek mindig pozitívak kell legyenek, ellenkező esetben a rotorok a gravitációs gyorsulással megegyező irányú gyorsulás komponenszt generálnak:

$$\begin{aligned} \omega_1 &= \sqrt{\max\left(0, \frac{1}{2b} \left(F + \frac{\tau}{l}\right)\right)} \\ \omega_2 &= \sqrt{\max\left(0, \frac{1}{2b} \left(F - \frac{\tau}{l}\right)\right)}. \end{aligned} \quad (12.8)$$

A jármű dinamikus modelljét kis dőlésszögek ($\theta \cong 0$) esetén a Newton-mozgástörvény alapján számolhatjuk:

$$\begin{aligned} m\ddot{x} + mg &= u_1 \\ I\ddot{\theta} &= u_2. \end{aligned} \quad (12.9)$$

Ebben a modellben a bemenetnek az F erőt és τ forgatónyomatékat tekinthetjük. A paraméterek: m – tömeg, I – inercia.

A jármű szimulációs modelljének felállításához a fenti modellt írjuk fel az alábbi alakban:

$$\begin{aligned} x &= \int_0^t \int_0^t \left(\frac{1}{m} F - g \right) d\zeta d\zeta \\ \theta &= \int_0^t \int_0^t \frac{1}{I} \tau d\zeta d\zeta. \end{aligned} \quad (12.10)$$

12.3. A mérés menete

Feladat: Tervezzünk szabályozót, amely biztosítja, hogy $\theta = 0$ és $x = x_{\text{ref}}$ állandósult állapotban. Írjuk elő a szabályozás tranziens minőségi jellemzőit (szabályozási idő – $T_{2\%}$, túllövés – Δv).

A *szabályozótervezés*: A pozicionálás tervezéséhez válasszuk az alábbi másodfokú referenciarendszert:

$$H_{\text{ref}}(s) = \frac{x(s)}{x_{\text{ref}}(s)} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}. \quad (12.11)$$

Az előírt túllövés és a szabályozási idő alapján a szabályozó tervezéséhez határozzuk meg a másodfokú referenciarendszer csillapítását és saját körfrekvenciáját (ξ, ω_n) .

$$\Delta v = \exp\left(-\frac{\pi\xi}{\sqrt{1-\xi^2}}\right) \quad (12.12)$$

$$T_{2\%} \cong \frac{4}{\xi \cdot \omega_n}. \quad (12.13)$$

A referenciarendszer időtartományban:

$$\ddot{x} + 2\xi\omega_n\dot{x} + \omega_n^2x = \omega_n^2x_{\text{ref}}. \quad (12.14)$$

PD+FF szabályozás: Válasszunk egy PD-szabályozót, kiegészítve előreccsatolással (FF – Feed-Forward)

$$u = K_P(x_{\text{ref}} - x) - K_D\dot{x} + m_m g \quad (12.15)$$

m_m a mechanikai rendszer mért/ismert tömegparamétere.

A szabályozott rendszer:

$$\begin{aligned} m\ddot{x} &= K_P((x_{\text{ref}} - x) - K_D\dot{x} + (m_m - m)g) \\ \ddot{x} + \frac{K_D}{m}\dot{x} + \frac{K_P}{m}x &= \frac{K_P}{m}x_{\text{ref}} + (m_m - m)\frac{g}{m}. \end{aligned} \quad (12.16)$$

Összehasonlítva a referenciarendszerrel a szabályozóparamétereket kapjuk:

$$K_D = 2m\xi\omega_n, \quad K_P = m\omega_n^2.$$

Ugyancsak látszik, hogy az állandósult állapotbeli kimenet:

$$x = x_{\text{ref}} + (m_m - m)\frac{g}{K_P}. \quad (12.17)$$

PID-szabályozás: Előreccsatolás helyett válasszunk a rendszer irányításához PID-szabályozót

$$u = K_P(x_{\text{ref}} - x) + K_I \int_0^t (x_{\text{ref}} - x)d\zeta - K_D\dot{x}. \quad (12.18)$$

A szabályozott rendszer:

$$m\ddot{x} = K_P(x_{\text{ref}} - x) + K_I \int_0^t (x_{\text{ref}} - x)d\zeta - K_D\dot{x} - mg. \quad (12.19)$$

Figyelembe véve, hogy m és x_{ref} konstansak, a fenti modellt deriválva kapjuk:

$$\ddot{x} + \frac{K_D}{m}\dot{x} + \frac{K_P}{m}x = \frac{K_I}{m}x_{\text{ref}}. \quad (12.20)$$

Mivel az irányított rendszert leíró folyamat harmadfokú, legalább harmadfokú referenciamodellt kell hogy válasszunk:

$$H_{\text{ref}}(s) = \frac{x(s)}{x_{\text{ref}}(s)} = \frac{\omega_n^2}{(Ts + 1)(s^2 + 2\xi\omega_n s + \omega_n^2)}. \quad (12.21)$$

A PID-szabályozó paramétereit összehasonlítással számolhatjuk ki, hasonlóan az előző esethez. Ugyanakkor a harmadfokú referenciarendszerben az extra időállandó ($T > 0$) miatt lassúbb válaszra számíthatunk.

Integráló szabályozás vs. előreccatolás:

- Általában is kijelenthető, hogy az előreccatolást alkalmazó szabályozással kisebb szabályozási idő érhető el, mint az integráló szabályozással.
- Az előreccatolás megvalósításához pontosan ismernünk kell a zavart, integráló szabályozáshoz a zavart nem kell ismerni.
- Az integráló zavarkompenzálás csak konstans zavarok esetében alkalmazható, az előreccatolás esetében az ismert zavar lehet időben változó is.
- Az előreccatoló szabályozás energiahatékonyabb, mivel pontosan kompenzálja az előreccatolást, nem kell „túlméretezni” a szabályozóparamétereket.

A tervezés menete:

– Tervezze meg az előreccatolást alkalmazó szabályozót és a PID-szabályozót az alábbi paraméterekkel.

– Folyamatparaméterek:

$$m = 1 \text{ kg}$$

$$l = 0,2 \text{ m}$$

$$J = 0,1 \text{ kgm}^2$$

$$g = 9,81 \text{ m/s}^2$$

$$b = 0,001$$

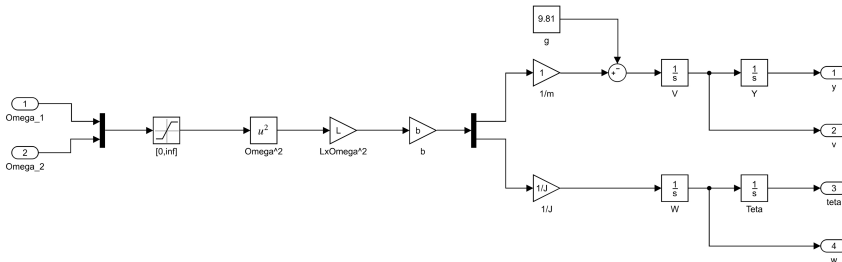
– Előírt tranziens minőségi jellemzők:

$$T_{2\%} = 1 \text{ s}$$

$$\Delta v = 4,2\%$$

$$T = 0,1 \text{ s}$$

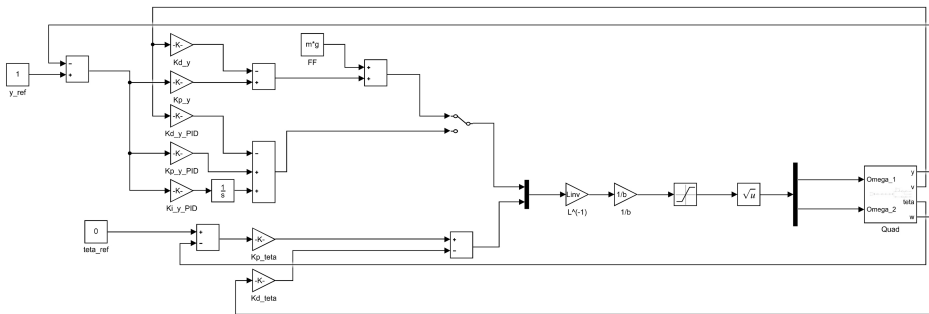
Szimulációs vizsgálatok: – Készítse el a repülő járműmodell Simulink diagramját a 12.3. ábrán látható módon.



12.3. ábra. A járműmodell Simulink diagramja

– Készítse el a jármű irányítási rendszerének a modelljét a 12.4. ábra alapján.

– Az egységugrásra adott válaszok alapján ellenőrizze a szabályozás tranziens minőségi jellemzőit mind a két szabályozási stratégia esetén.



12.4. ábra. A jármű irányítási rendszerének Simulink diagramja

12.4. Kérdések és feladatok

1. Számoljuk ki, hogy ha a jármű tömege 10%-ot változik, mennyivel változik az állandósult állapotbeli hiba előreccsatolást alkalmazó szabályozók esetén? Ellenőrizzük szimulációval a számítást.
2. Milyen hatása van a b beavatkozó-paraméter 10%-os változása a szabályozás minőségi jellemzőire?
3. Mikor van értelme olyan szabályozót alkalmazni, amely tartalmaz integráló tagot és előreccsatoló ágat is?

LEJTŐN GURULÓ GOLYÓ NEMLINEÁRIS IRÁNYÍTÁSA

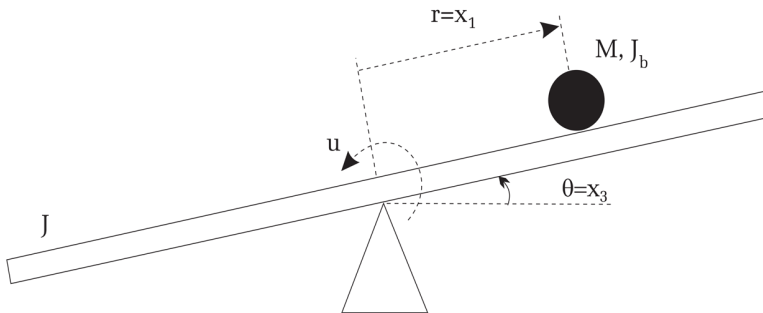
13.1. A gyakorlat célja

Lyapunov-technikákon alapuló szabályozótervezési módszerek elsajátítása, alkalmazása a lejtő-golyó rendszerre. A nemlineáris szabályozási rendszer vizsgálata szimulációkkal. Az s -függvények technikájának megismerése.

13.2. Elméleti bevezető

A lejtőn guruló golyó dinamikus modellje: A lejtőn guruló golyó, mint irányított rendszer, egy vezérelhető dőlésszögű lejtőből és egy rajta szabadon guruló golyóból áll. Mivel az egyedüli beavatkozó a rendszerbe a lejtő dőlésszögének szabályozását végző motor és a rendszer szabadságfoka 2 (a dőlésszög és a golyó pozíciója a lejtőn), a mechanikai rendszer alulirányított.

Az irányítási feladat: A golyót a lejtő egy adott pontjába szabályozni vagy azt elérni, hogy a golyó egy előre megadott pályát kövessen a lejtőn. Az irányításhoz feltételezzük, hogy a lejtő dőlésszöge, valamint szögsebessége, a golyó pozíciója és sebessége mérhető.



13.1. ábra. Lejtőn guruló golyó rendszer vázlatja modellezéshez

Jelölje ϑ a lejtő szögelfordulását, J a lejtő (forgástengelyére vonatkozó) tehetetlenségi nyomatékát, M a golyó tömegét, R a golyó sugarát, J_b a golyónak (a középpontjára vonatkozó) tehetetlenségi nyomatékát és r a golyó távolságát a lejtő tengelyétől. Ha a golyó csúszásmentesen gördül, akkor a golyó \dot{r} sebessége és a gördülő ω golyó szögsebessége között fennáll: $\dot{r} = R\omega \Rightarrow \omega = \dot{r}/R$. Jelölje M a golyó tömegét, akkor a golyó potenciális energiája:

$$P = Mgr \sin \vartheta. \quad (13.1)$$

A kinetikus energia több komponensből áll. A forgó lejtő kinetikus energiája $J\dot{\vartheta}^2/2$. A nem gördülő, de a lejtővel együtt forgó golyó kinetikus energiája $(J_b + Mr^2)\dot{\vartheta}^2/2$, ahol J_b a golyó középpontjára vonatkozó tehetetlenségi nyomaték. A gördülő golyó kinetikus energiája $J_b(\dot{r}/R)^2/2$, a tömegpont sebességének hatása $Mr^2/2$. A teljes kinetikus energia:

$$K = \frac{1}{2}J\dot{\vartheta}^2/2 + \frac{1}{2}(J_b + Mr^2)\dot{\vartheta}^2 + \frac{1}{2}J_b \left(\frac{\dot{r}}{R}\right)^2 + \frac{1}{2}Mr^2. \quad (13.2)$$

A lejtőn guruló golyó dinamikus modelljének meghatározására a Lagrange-egyenleteket fogjuk használni.

$$\frac{d}{dt} \frac{\partial K}{\partial \dot{q}_i} - \frac{\partial K}{\partial q_i} + \frac{\partial P}{\partial q_i} = \tau_i, \quad i = 1, \dots, n. \quad (13.3)$$

Általánosított koordinátákként választható $q = (q_1, q_2)^T := (r, \vartheta)^T$. A Lagrange-egyenletben szereplő $\frac{\partial}{\partial q_i}$, $\frac{d}{dt} \frac{\partial}{\partial q_i}$, $\frac{\partial}{\partial \dot{q}_i}$, deriválások eredményei a következők:

$$\begin{aligned} \frac{\partial K}{\partial \dot{r}} &= M\dot{r} + J_b \left(\frac{\dot{r}}{R}\right) \frac{1}{R} \\ \frac{d}{dt} \frac{\partial K}{\partial \dot{r}} &= \left(M + \frac{J_b}{R^2}\right) \ddot{r} \\ \frac{\partial K}{\partial r} &= Mr\dot{\vartheta}^2 \\ \frac{\partial P}{\partial r} &= Mg \sin \vartheta \\ \frac{\partial K}{\partial \dot{\vartheta}} &= (J + J_b + Mr^2)\dot{\vartheta} \end{aligned}$$

$$\begin{aligned}
\frac{d}{dt} \frac{\partial K}{\partial \dot{\vartheta}} &= (J + J_b + Mr^2)\ddot{\vartheta} + 2Mr\dot{r}\dot{\vartheta} \\
\frac{\partial K}{\partial \vartheta} &= 0 \\
\frac{\partial P}{\partial \vartheta} &= Mgr \cos \vartheta.
\end{aligned} \tag{13.4}$$

Ezek felhasználásával a következő mozgásegyenleteket kapjuk:

$$\begin{aligned}
\frac{d}{dt} \frac{\partial K}{\partial \dot{r}} - \frac{\partial K}{\partial r} + \frac{\partial P}{\partial r} &= \left(M + \frac{J_b}{R^2} \right) \ddot{r} - Mr\dot{\vartheta}^2 + Mgr \sin \vartheta = 0 \\
\frac{d}{dt} \frac{\partial K}{\partial \dot{\vartheta}} - \frac{\partial K}{\partial \vartheta} + \frac{\partial P}{\partial \vartheta} &= (J + J_b + Mr^2)\ddot{\vartheta} + 2Mr\dot{r}\dot{\vartheta} + Mgr \cos \vartheta = \tau.
\end{aligned} \tag{13.5}$$

Ha beavatkozójelnek az

$$u := \ddot{\vartheta} = -\frac{2Mr\dot{r}\dot{\vartheta} + Mgr \cos \vartheta}{J + J_b + Mr^2} + \frac{1}{J + J_b + Mr^2} \tau \tag{13.6}$$

szöggyorsulást választjuk, akkor a rendszer nemlineáris dinamikus modellje az $x = (r, \dot{r}, \vartheta, \dot{\vartheta})^T$ állapotválasztás és $B = M/(M + J_b/R^2)$ jelölés mellett:

$$\frac{d}{dt} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} x_2 \\ B(x_1 x_4^2 - g \sin x_2) \\ x_4 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u. \tag{13.7}$$

A *Lyapunov-tétel*: A Lyapunov-stabilitástétel alkalmazható irányítások tervezésére is.

Legyen a vizsgált nemlineáris dinamikus rendszer: $\dot{\underline{x}} = f(\underline{x}, t)$. Először vezessük be az egyensúlyi állapot fogalmát. \underline{x}^* a rendszer *egyensúlyi állapota*, ha a t^* pillanatban a rendszer állapota \underline{x}^* , akkor bármely $t > t^*$ pillanatban az állapot \underline{x}^* marad.

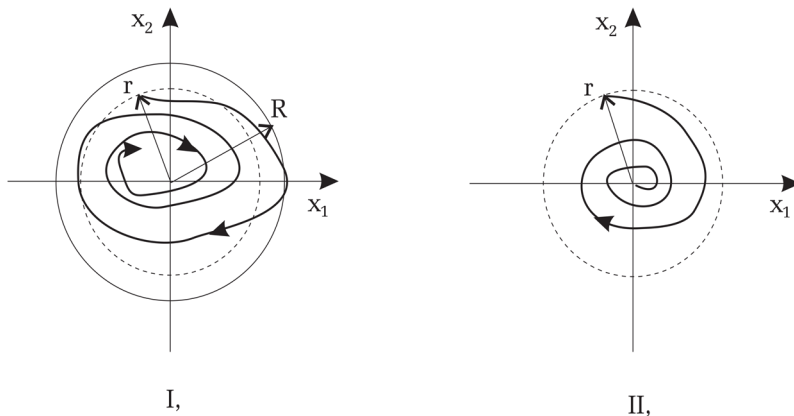
Mivel egyensúlyban az állapotok nem változnak, ezért $\dot{\underline{x}}^* = f(\underline{x}^*, t) = \underline{0}$. Ezt a feltételt használhatjuk a rendszer egyensúlyi állapotainak feltérképezésére.

A nemlineáris rendszerek esetén az egyensúlyi pontok stabilitását vizsgáljuk. Feltételezzük, hogy a rendszer állapotterének origója ($\underline{x} = \underline{0}$) egyensúlyi állapot. A rendszer $\underline{x} = \underline{0}$ egyensúlyi állapota stabil, ha az állapotok

nem távolodnak el végtelenül az origótól. Ha az állapotok nullába konvergálnak, akkor azt mondjuk, hogy a rendszer aszimptotikusan stabil. A pontos matematikai megfogalmazáshoz vezessük be a norma fogalmát, vagyis az állapot távolságát az origótól: $\|\underline{x}\| = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$.

Stabilitás: $\underline{x} = \underline{0}$ stabil egyensúlyi állapot a $t = t_0$ időpillanatban, ha bármely $R > 0$ -ra létezik $r(R, t_0) > 0$ úgy, hogy ha $\|\underline{x}(t_0)\| < r$ akkor $\|\underline{x}(t)\| < R$ bármely $t > t_0$.

Aszimptotikus stabilitás: $\underline{x} = \underline{0}$ stabil egyensúlyi állapot a $t = t_0$ időpillanatban, ha létezik $r(t_0) > 0$ úgy, hogy ha $\|\underline{x}(t_0)\| < r$ akkor $\|\underline{x}(t)\| \rightarrow 0$, ha $t \rightarrow \infty$.



13.2. ábra. Stabilitás (I) és aszimptotikus stabilitás (II) fázistérben

A definíciók illusztrálása a 13.2. ábrán látható. Az ábrán egy másodfokú stabil rendszer és egy másodfokú aszimptotikusan stabil rendszer fázistere látható. Stabil esetben, ha a kezdeti időpillanatban az állapotok véges távolságra vannak az állapottér origójától, akkor bármely kezdeti állapot utáni időpillanatban is véges távolságra maradnak az origótól. Aszimptotikusan stabil esetben, ha a kezdeti időpillanatban az állapotok véges távolságra vannak az állapottér origójától, akkor konvergálnak az origóhoz.

A nemlineáris dinamikus rendszerek stabilitásának vizsgálatához a *Lyapunov-tételt* alkalmazhatjuk:

Ha létezik egy olyan $V(\underline{x}, t)$ függvény, amely deriválható $\underline{x} = \underline{0}$ egyensúlyi pont körül, és legyenek az alábbi feltételek:

- I. $V(\underline{0}, t) = 0$ és $V(\underline{x}, t) > 0$, ha $\underline{x} \neq \underline{0}$ ($V(\underline{x}, t)$ pozitív definit)
- II. a $\dot{V}(\underline{x}, t) \leq 0$ ($\dot{V}(\underline{x}, t)$ negatív szemidefinit)

II. b $\dot{V}(\underline{x}, t) < 0$ ($\dot{V}(\underline{x}, t)$ (negatív definit) Ha az I. és a II. a feltétel teljesül, akkor a rendszer stabil, ha az I. és a II. b feltétel teljesül, akkor a rendszer aszimptotikusan stabil.

A Lyapunov-tételben szereplő V függvény a vizsgált rendszerhez rendelt energiafüggvény. Az I. feltétel szerint értéke mindig pozitív, így a legkisebb érték, amit felvehet: 0. A II. a feltétel értelmében a V függvény nem növekvő, tehát mindig véges értékű, a II. b függvény értelmében pedig a V csökkenő függvény.

Ha V pozitív és csökkenő, akkor nullába kell hogy konvergáljon.

Modellalapú irányítás: Ahhoz, hogy pályakövetést is megvalósítani képes irányítási algoritmust dolgozzunk ki, a (13.7) kifejezéssel leírt modellt írjuk át az alábbi alakra:

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= B\xi_B(\underline{x}) \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= K_1\xi_1(\underline{x}) + K_2\tau\end{aligned}\tag{13.8}$$

ahol:

$$\begin{aligned}K_1 &= -\frac{M}{(Mx_1^2 + J + J_b)} & K_2 &= \frac{1}{(Mx_1^2 + J + J_b)} \\ \xi_B = (\underline{x}) &= x_1x_4^2 - g \sin x_3 & \xi_1(\underline{x}) &= 2x_1x_2x_4 + gx_1 \cdot (x_3).\end{aligned}\tag{13.9}$$

Vegyük észre, hogy K_2 felülről korlátos, vagyis $K_2 < K_{2m}$, ahol $K_{2m} = 1/(J + J_b)$.

Az irányítási algoritmus kidolgozásához vezessük be az alábbi összetett sebesség- és pozícióhibát:

$$S = \alpha S_1 + S_2 = \alpha((x_2 - \dot{x}_{1d}) + \lambda_1(x_1 - x_{1d})) + (x_4 + \lambda_2x_3)\tag{13.10}$$

ahol $\alpha, \lambda_1, \lambda_2 > 1$, x_d korlátos, kétszer differenciálható előírt pálya. Az első tag (S_1) a golyó szabályozási hibája, a második tag (S_2) a lejtő túlságosan nagy dőlésszögét bünteti annak érdekében, hogy elkerüljük a lejtő nagy amplitúdójú oszcillációját, ami akár instabilitáshoz is vezethet.

Természetesen az irányítás eredményeként nem várhatjuk el, hogy $S \rightarrow 0$, amikor $t \rightarrow \infty$, mivel a golyó mozgása feltételezi, hogy a második tag sohasem lesz 0. Definiáljuk az elérhető pontosságot Φ -vel és vezessük be az alábbi függvényt $S\Delta = S - \Phi\text{sat}(S/\Phi)$, ahol $\text{sat}()$ a telítődési függvényt jelöli. Amennyiben az irányítási törvény garantálja, hogy $S_\delta \rightarrow 0$, abban az esetben S konvergál a Φ határréteg belsejébe.

Könnyen belátható, hogy S_Δ kielégíti az alábbi összefüggéseket:

$$\begin{aligned} \dot{S}_\Delta = \dot{S} \text{ ha } |S_\Delta| > \Phi; \quad \dot{S}_\Delta = 0 \text{ máskülönben} \\ S_\Delta \text{sat}(S/\Phi) = \|S_\delta\| \text{ ha } |S_\Delta| > \Phi. \end{aligned} \quad (13.11)$$

Definiáljuk az alábbi Lyapunov-függvényt:

$$V = \frac{1}{2} S_\Delta^2. \quad (13.12)$$

Amennyiben a zaj hatását elhanyagoljuk a modellből, a Lyapunov-függvény deriváltját az alábbi módon írhatjuk fel:

$$\dot{V} = S_\Delta \dot{S}_\Delta = S_\Delta (\alpha((B\xi_B(\underline{x}) - \ddot{x}_{1d}) + \lambda_1(x_2 - x_{2d})) + (K_1\xi_1(\underline{x}) + K_2\tau + \lambda_2x_4)). \quad (13.13)$$

Ha $|S_\Delta| \leq \Phi$ akkor $\dot{V} = 0$.

Válasszuk az irányítási törvényt:

$$u = \frac{1}{K_2} (-\alpha((B\xi_B(\underline{x}) - \ddot{x}_{1d}) + \lambda_1(x_2 - x_{2d})) - K_1\xi_1(\underline{x}) - \lambda_2x_4 - K_S S_\Delta) \quad (13.14)$$

ahol $K_S > 0$.

Az irányítási törvény biztosítja, hogy $\dot{V} = -K_S S_\Delta^2 < 0$, ha $S_\Delta \neq 0$, tehát garantálja S konvergenciáját a Φ határréteg belsejébe.

13.3. A mérés menete

Folyamatparaméterek: Legyen a mérlegkaron guruló golyó rendszer, amelyet a (13.8) és (13.9) összefüggésben megadott modellel írhatunk le. A modell paraméterei:

$$\begin{aligned} M &= 0,05 \text{ kg}; \\ R &= 0,01 \text{ m}; \\ J &= 0,01 \text{ kgm}^2; \\ J_b &= 1E - 6 \text{ kgm}^2; \\ g &= 9,81 \text{ m/s}^2. \end{aligned}$$

Valósítsuk meg a rendszer nemlineáris szabályozását úgy, hogy az alapjel egy 0,5 Hz frekvenciájú, 0,4 m amplitúdójú szinuszos jel legyen.

Az s-függvények: A (13.7) és (13.6)-ban megadott modellt és a (13.14) összefüggésben megadott szabályozót *s-function* formájában építjük fel. Az *s-függvény* általános formája:

```

function [sys, x0]= model(t, x, u, flag)

if (flag==0)
    \%Initialization
    sys = [ ,      \% number of continuous states
           ,      \% number of discrete states
           ,      \% number of outputs
           ,      \% number of inputs
           0,     \% reserved must be zero
           ];     \% direct feedthrough flag
    x0 = [];

if (flag==1)                               \% continous states
    sys=
elseif (flag==2)                           \% output equation
    sys=
elseif (flag==3)                           \% model constants
    sys=
else
    sys=[];
end

```

Egy s -függvény bemenetei az idő (t), a rendszer állapotai (x), a rendszer bemenetei (u) és egy kapcsoló ($flag$), amely az s -függvény állapotát adja meg. A visszaterítési érték (sys) a kapcsoló értékétől függ.

Ha a kapcsoló értéke 0, akkor a rendszer dimenzióit és kezdő állapotait (x_0) adjuk meg. Az sys utolsó paramétere 0, ha a bemenet hatása egyenes úton jelentkezik a kimeneten (statikus előreccatolás is van a rendszerben).

Ha a kapcsoló értéke 1, akkor a rendszer folytonos állapotainak változását kell visszatéríteni (dx/dt).

Ha a kapcsoló értéke 2, akkor a rendszer diszkrét állapotainak változását kell visszatéríteni (x_{k+1}).

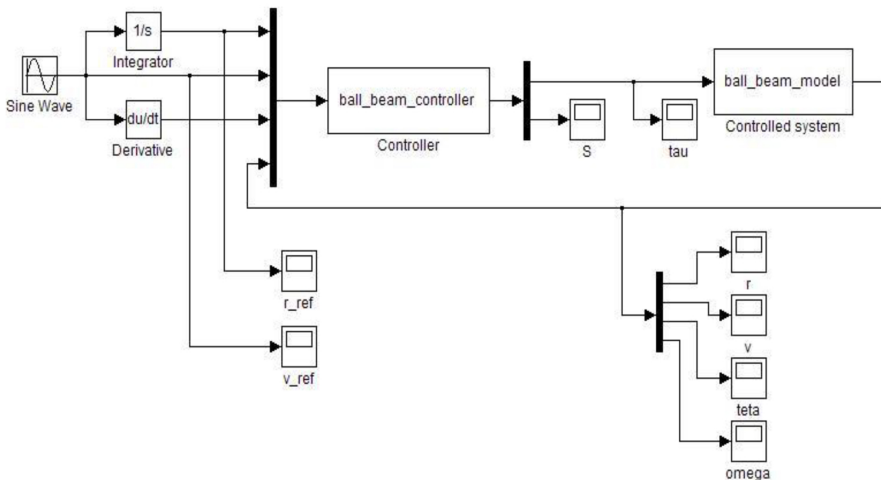
Ha a kapcsoló értéke 3, akkor a rendszer kimeneteit kell visszatéríteni (y).

A mérlegkar-golyó szabályozó implementálása: A modell esetén a rendszernek négy állapota van, a kimenetei az állapotok; a rendszernek egy bemenete van. Kezdeti állapotnak a kezdőpozíciója a rendszernek legyen 1, az összes többi kezdőállapot pedig nulla.

A rendszer folytonos állapotainak változását a (13.8), (13.9) összefüggések adják, mintavételes állapotai pedig nincsenek.

A szabályozási algoritmus esetén a rendszernek hét bemenete (az előírt pozíció, sebesség és gyorsulás, valamint a rendszer négy visszamért állapota) és két kimenete van (a beavatkozójel és az S összetett hibametrika). A szabályozónak nincs folytonos, sem diszkrét dinamikus állapota, a bemenetek és a kimenet között statikus leképzés van. A beavatkozójel számítását a (13.10) és (13.14) összefüggés adja.

A szabályozó paramétereit válasszuk: $\lambda_1 = 10$; $\lambda_2 = 100$; $\alpha = 10$; $K_S = 100$. Készítsük el a rendszer Simulink diagramját a 13.3. ábra alapján.



13.3. ábra. A szabályozási rendszer Simulink diagramja

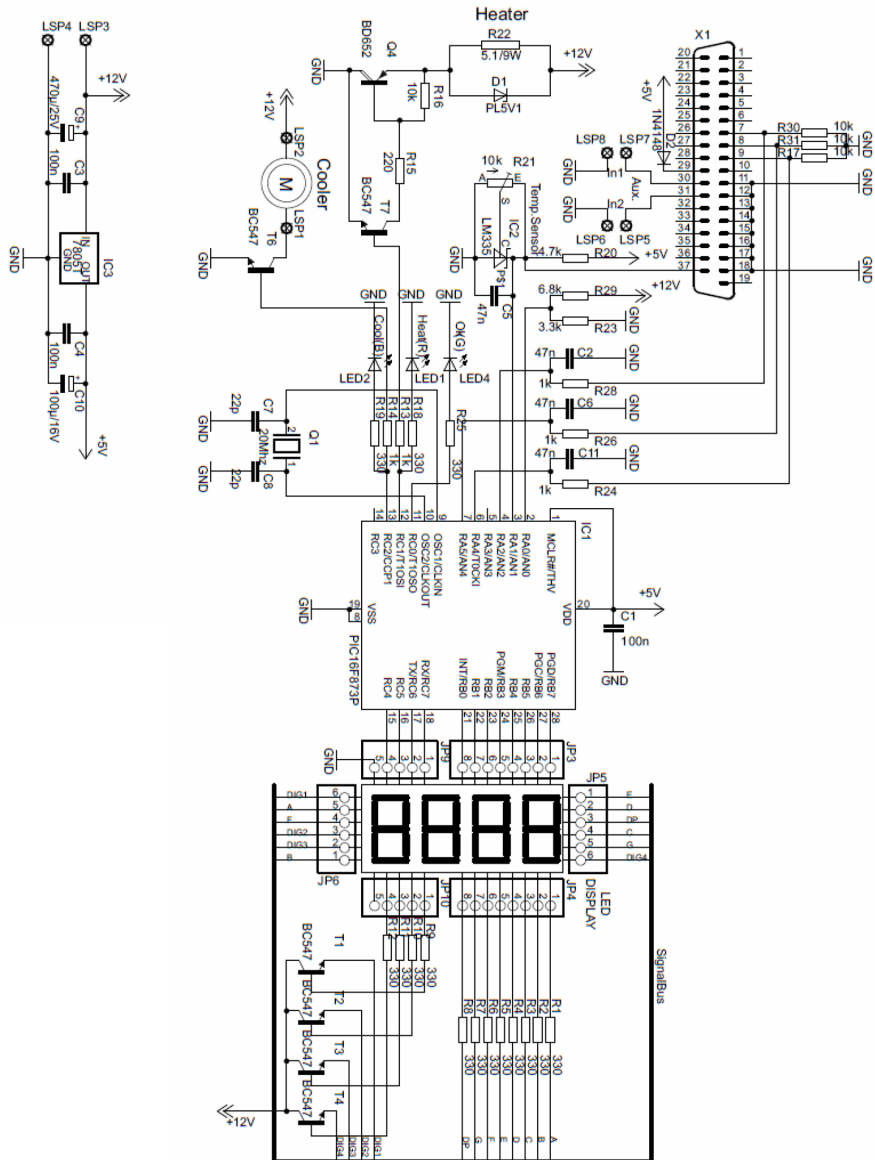
Az eredmények kiértékelése: A 13.3. ábra alapján figyeljük meg a rendszer előírt pályáját és a kimeneteit.

- Figyeljük meg az összefüggést a lejtő dőlésszöge és a beavatkozójel között.
- Hogyan befolyásolja a K_s erősítés értéke a szabályozás tranzienseit?
- Hogyan befolyásolják a λ_1, λ_2 paraméterek a szabályozás tranzienseit?
- Vizsgáljuk a szabályozás robusztosságát: Módosítsuk a golyó tömegét a rendszermodellben. Hogyan változik meg a szabályozás minősége nehezebb, illetve könnyebb golyóval?

13.4. Kérdések és feladatok

1. Módosítsuk úgy a Simulink diagramot, hogy a szabályozót és a folyamatot egy s -függvényben valósítsuk meg.
2. Hogyan valósítaná meg C programozási nyelven a bemutatott irányítási algoritmust?
3. Keresse meg az inverz inga rendszer dinamikus modelljét (Internet). Alkalmazható-e a bemutatott irányítási algoritmus elve az inverz inga irányítására?

FÜGGELÉK



F1. ábra. Hőmérséklet-szabályozás – vezérlő áramkör

```

namespace Temperature_Control
{
    public partial class Temperature_Control_Form : Form
    {
        public Temperature_Control_Form()
        {
            InitializeComponent();
        }

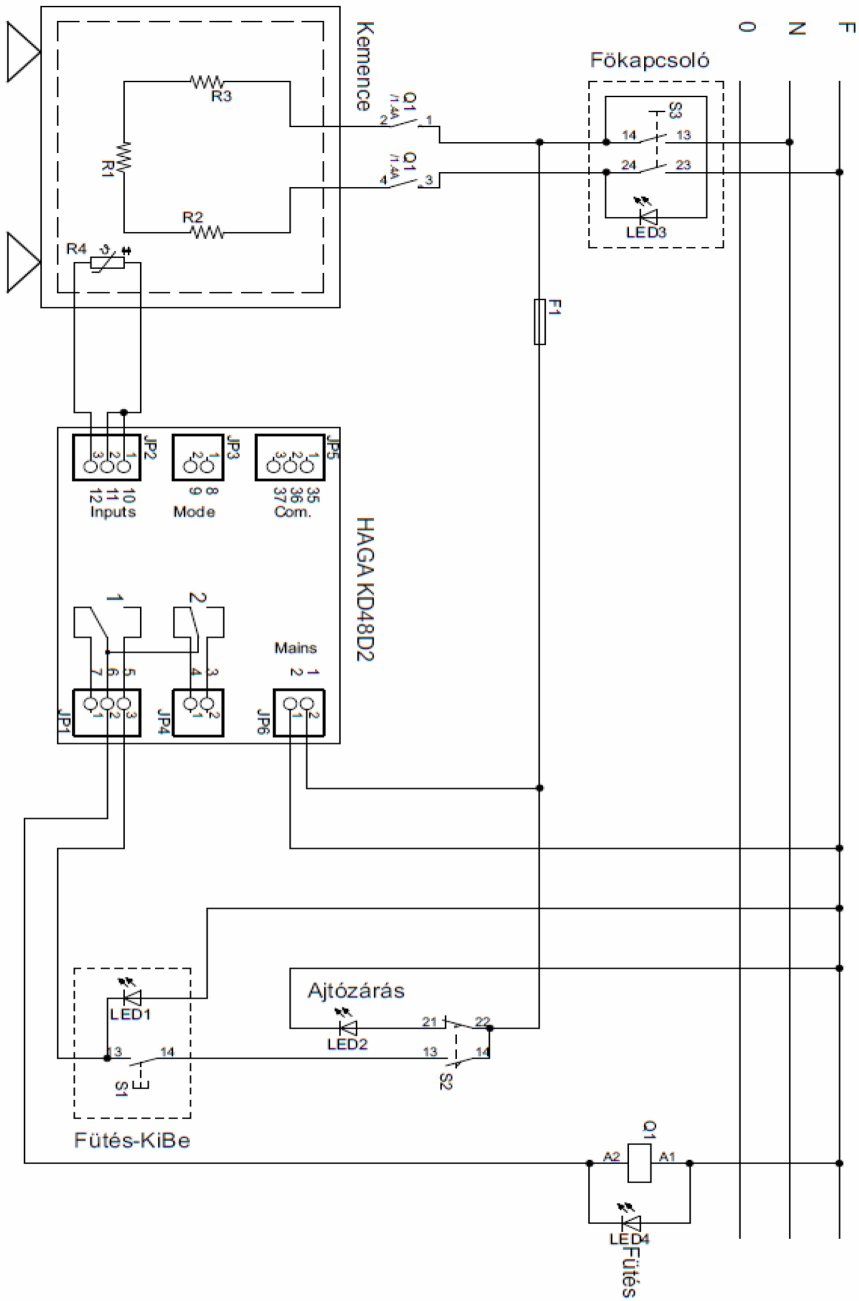
        private void Temperature_Control_Form_Load
        (object sender, EventArgs ea)
        { // . . . }

        private void controlTimer_Tick
        (object sender, EventArgs ea)
        { // . . . }

        private void Temperature_Control_Form_FormClosed
        (object sender, FormClosedEventArgs ea)
        { // . . . }
    }
}

```

F2. ábra. Hőmérséklet-szabályozás – A C# program váza



F3. ábra. Kemence hőmérséklet-szabályozása – Kapcsolási rajz

SZAKIRODALOM

- KULAKOWSKI, Bohdan T. – GARDNER, John F. – SHEARER, J. Lowen 2014 *Dynamic modeling and control of engineering systems*. Cambridge, Cambridge University Press.
- KING, Myke 2016 *Process control: a practical approach*. Chichester, John Wiley&Sons.
- KUO, Benjamin C. – GOLNARAGHI Farid 2003 *Automatic control systems*. New York, John Wiley.
- KUTASI Dénes Nimród – MÁRTON László Ferenc 2010 *Rendszerelmélet: laboratóriumi gyakorlatok*. Kolozsvár, Scientia.
- LANTOS Béla 2002 *Irányítási rendszerek elmélete és tervezése*. Budapest, Akadémiai Kiadó.
- MÁRTON Lőrinc 2009 *Irányítástechnika*. Kolozsvár, Scientia.
- MÁRTON László Ferenc 2006 *Jelek és rendszerek*. Kolozsvár, Scientia.
- PÉREZ López, César 2014 *Matlab control systems engineering*. Berkeley, Apress.
- TROELSEN, Andrew 2009 *A C# 2008 és a .NET 3.5*. Bicske, Szak Kiadó.
- VINATORU, Matei 2011 *Fundamente de sisteme automate: teorie și aplicații*. Craiova, Sistech.

KIVONAT

A laboratóriumi útmutató célja bevezetőt nyújtani a klasszikus irányítási eljárások tervezésébe és megvalósításába.

Az első négy laboratóriumi gyakorlat a hőmérséklet-szabályozási feladattal foglalkozik. A diákok meg kell valósítsanak egy egybemenetű, egykimenetű szabályozási kör irányítását biztosító számítógépes programot, amely magába foglalja a folyamat felügyeletét biztosító grafikus kijelzést, a folyamat kétállású szabályozását, a folyamat proporcionális szabályozását, valamint a profilkövetést megvalósító szabályozást.

Az ötödik és hatodik laboratóriumi gyakorlat a holtidős folyamatok irányításának tervezésével foglalkozik, ipari kemencékre bemutatva, és az önhangoló szabályozás módszerével.

A referenciamodell-alapú tervezést egy oldat koncentrációsabályozási feladatán keresztül mutatja be a hetedik laboratóriumi gyakorlat (vegyi folyamat szabályozása).

Az irányításhoz elengedhetetlen az irányított folyamat modellezése. A nyolcadik laboratóriumi gyakorlat az egyenáramú motor modellezésével foglalkozik. A kilencedik laboratóriumi gyakorlat az egyenáramú motor kaszkádszabályozását mutatja be.

A tizedik laboratóriumi gyakorlat a Proporcionális-Integráló-Deriváló (PID) szabályozók paramétereinek hangolásának hatását mutatja be a szabályozási minőségi jellemzőkre. A tizenegyedik laboratóriumi gyakorlat bemutatja az előreccsatolást alkalmazó szabályozókat, összehasonlítva őket az integrátor tagot tartalmazó szabályozókkal.

A tizenkettedik laboratóriumi gyakorlat betekintőt nyújt a nemlineáris irányítások tervezésébe a lejtőn guruló golyó példáján keresztül.

REZUMAT

Scopul acestui îndrumar de laborator este de a introduce studenții în proiectarea și implementarea algoritmilor de reglare convenționali și a unor regulatoare avansate.

Primele patru laboratoare se ocupă de problema buclei de reglare a temperaturii. Studenții trebuie să implementeze un program de calculator pentru controlul unui proces cu o intrare și o ieșire. Programul include monitorizarea grafică a procesului, implementarea interfeței proces-calculator, controlul cu un regulator bipozițional, comanda procesului cu un regulator proporțional, respectiv urmărirea unui profil de temperatură.

Lucrările de laborator cinci și șase au ca scop prezentarea proiectării reguletoarelor pentru procesele cu timp mort. Sunt prezentate metodele de acordare convenționale a reguletoarelor și reguletoarele cu auto-acordare. Implementarea reguletoarelor este exemplificată pe un cuptor industrial de încălzire.

Proiectarea reguletoarelor pe baza unui sistem de referință este ilustrată prin controlul concentrației chimice.

Pentru controlul avansat este inadmisibil obținerea modelului matematic al procesului condus. A opta lucrare se ocupă cu modelarea unui motor de curent continuu, care este un element de execuție folosit în multe bucle de reglare. A noua lucrare are ca scop proiectarea reguletorului în cascadă a motorului de curent continuu folosind modelul obținut în lucrarea precedentă.

A zecea lucrare de laborator prezintă influența acordării parametrilor reguletoarelor PID (Proporțional-Integral-Derivativ) pe performanțele de reglare. Lucrarea a unsprezecea prezintă reguletoarele care au în structură element feed-forward. Aceste reguletoare sunt comparate cu reguletoare care conțin element de integrare.

A douăsprezecea lucrare de laborator prezintă o introducere în controlul neliniar ai sistemelor mecanice prin exemplul procesului rampă-bilă.

ABSTRACT

This laboratory exercise guidebook aims to provide a deeper understanding of the design and implementation of classic control methods.

The first four exercises cover the problem of temperature control. The students have to implement a software-based control loop for a single-input single-output system. The software have to contain the graphical process monitoring, a hysteresis controller, proportional control, and a reference tracking algorithm.

The fifth and sixth exercises deal with the control of systems with dead time. In these exercises, the students will learn the controller tuning methods with the help of an industrial furnace application, and the principle of self-tuning control.

The reference model-based controller design is shown in the seventh exercise through a chemical concentration control problem.

Most control methods require the model of the controlled system. In the eighth exercise, the students will study the model of a DC motor. In the ninth activity, the previously designed model will be used for cascade position control algorithm design.

The tenth exercise shows the effect of the Proportional Integrative Derivative (PID) controller parameter tuning of the controller on control loop performances. The eleventh exercise presents the control algorithms which use feed-forward terms and compares them with control methods containing integrator terms.

The twelfth exercise provides an insight into the nonlinear control design methods through the model-based control of the ball and beam system.

A SZERZŐKRŐL

Márton Lőrinc (www.ms.sapientia.ro/~martonl) középiskolai tanulmányait a marosvásárhelyi Bolyai Farkas Elméleti Líceumban végezte 1990–1994 között, matematika–fizika szakon. A marosvásárhelyi Petru Maior Egyetem Automatizálás és Ipari Informatika szakán 1999-ben szerzett egyetemi oklevelet. 2000–2003 között doktorandusz a Budapesti Műszaki és Gazdaságtudományi Egyetemen, 2006-ban doktori címet szerez. 2007–2010 között posztdoktori kutatói munkáját a Magyar Tudományos Akadémia Bolyai János kutatási ösztöndíjának támogatásával végzi. 2010–2012 között a Német Űrkutatási Hivatal (DLR) Robotika és Mechatronika Intézetének kutatója az Alexander von Humboldt Alapítvány támogatásával. 2003-tól a Sapientia Erdélyi Magyar Tudományegyetem Marosvásárhelyi Karának oktatója, ugyanitt 2019-től egyetemi professzor. Főbb kutatási területei az irányításelmélet, hálózatok, robotirányítások.

Fehér Áron (www.ms.sapientia.ro/~fehera) középiskolai tanulmányait a marosvásárhelyi Bolyai Farkas Elméleti Líceumban végezte 2007–2011 között, természettudomány szakon. A marosvásárhelyi Sapientia Erdélyi Magyar Tudományegyetemen szerzett mérnöki oklevelet Automata és Alkalmazott Informatika szakon 2015-ben, majd mesterszintű oklevelet Számítógépes Irányítási rendszerek szakon 2017-ben. 2017-től doktorandusz a veszprémi Pannon Egyetem Műszaki Informatika Karán. 2019-től tanársegéd a Sapientia Erdélyi Magyar Tudományegyetem Marosvásárhelyi Karán. Főbb kutatási területei folytonos és hibrid rendszerek modellezése és szabályozása, késleltetett rendszerek tranzienst viselkedésének vizsgálata, analóg és digitális jelfeldolgozási algoritmusok, biológiai jelfeldolgozás, biológiai rendszerek modellezése és szabályozása.

Scientia Kiadó

400112 Kolozsvár (Cluj-Napoca)

Mátyás király (Matei Corvin) u. 4. sz.

Tel./fax: +40-364-401454

E-mail: scientia@kpi.sapientia.ro

www.scientiakiado.ro

Korrektúra

Szenkovics Enikő

Műszaki szerkesztés

Csizmadia Erzsébet

Tipográfia

Könczey Elemér

Az Irányítástechnika laboratóriumi útmutató gyakorlati segédanyag a műszaki oktatásban részt vevő hallgatók számára. Tizenkét gyakorlatot tartalmaz, amelyek hozzásegítik a hallgatókat a szabályozástechnikai alapfogalmak megismeréséhez, begyakorlásához és alkalmazásához. Külön gyakorlatok foglalkoznak a mintavételes szabályozások megvalósítási problémáival, illetve a klasszikus szabályozó tervezési módszerekkel. A szabályozástechnikai feladatokat gyakorlati problémákon keresztül mutatjuk be – ilyenek a hőmérsékletszabályozás, az egyenáramú motor kaszkád szabályozása, az ipari kemencék szabályozása, illetve mechanikai rendszerek és vegyi folyamatok PID szabályozása.

ISBN 978-606-975-023-0



9 786069 750230