

# Проектиране на обобщено предсказващо управление

Евгени Динев

Технически университет – София

Проектиране на обобщено предсказващо  
управление

Дипломна работа

Факултет "Автоматика"

Катедра "Системи и управление"

Дипломант: Евгени Ст. Динев, фак. № 01021867

Ръководител: Емил М. Гарипов

София, 14 юни 2002 г

## **КОМИСИЯ**

---

Председател:

проф. Петко Петков

Членове:

проф. дтн. Никола Маджаров

проф. д-р Гео Гатев

доц. д-р Елена Хараланова

доц. д-р Емил Гарипов

доц. дтн Камен Ищев

доц. д-р Никола Христов

ТУ — София, зала 2114<sup>Б</sup>

28 юни 02 г., 9:00 часа



## Рецензия

Дипломната работа е интересна т.к. е посветена на предсказващото управление, един съвременен подход за управление, който намира все по-широко приложение с увеличаване на бързодействието на изчислителната техника. Интересът се засилва и от факта, че в нашата страна все още няма много разработки в тази област. От тази гледна точка направеният обзор на методите за предсказващо управление е полезен, като е разгледано развитието на методите. Като забележка бих отбелязал омаловажаването на достойнствата на доказали се методи за управление, както и на други подходи за предсказващо управление, намерили широко място в литературата (например, MPC метод и да се покаже връзката му с GPC).

Дипломната работа е посветена на една модификация на метода за предсказващо управление. Този метод разглежда входно изходната постановка на задачата, докато в литературата се разглежда предимно описание в пространство на състоянията. Това може да се счита както за предимство, така и за недостатък с оглед на запознаването с предсказващото управление на евентуалните студенти, за които се твърди, че разработката е предназначена. Съставена е програма за решаване на задачата и е направено сравнение с програма разработена в Испания, реализираща същия метод. Направени са експерименти с цел да се установи влиянието на параметрите, върху характеристиките на системата. Според мен от първите експерименти е направен грешен извод относно влиянието на хоризонта на предсказване ( $N_2$ ). Аз смятам, че това се дължи на факта, че при избора му са използвани десетократно по-малки стойности от препоръчаните в литературата. В експеримент 4 са взети правилни стойности за параметрите, като се казва, че те са най-добрите, но за жалост не е дадена методика за тяхното намиране. Представени са и серия от 4 упражнения, които са заимствани отново от Испания, но са преработени за нашите условия и дипломанта се е запознал основно с тях. Считаю че дипломната работа е в достатъчен обем и че дипломанта се е справил с поставените му цели. Поради това предлагам на комисията да даде висока оценка на работата.

Искам да дам и някои препоръки за бъдеща работа: Трябва да бъдат разгледани важни въпроси като i) устойчивост — за разлика от конвенционалните регулатори при предсказващите регулатори устойчивостта не е гарантирана, ii) решаването на задачата с ограничения, което е едно от съществените достойнства на предсказващите регулатори (в литературата се доказва, че задачата без ограничения се свежда до просто умножение, а не както идентификация и оптимизационна процедура). Това значително ускорява работата на алгоритъма, iii) следва да се обърне по-голямо внимание и на особено важния въпрос за настройка параметрите на регулатора. Предлагам да се разгледа и стандартния GPC метод в който не се използва идентификация на всяка стъпка, а разгледаният проблем с шума на изхода се решава с филтър. Това ще ускори метода. Интерес представлява да се направи сравнение със курса, упражненията и разработеният toolbox в Холандия.

Инж. Александър Ищев

## Въведение

Предсказващото управление се основава на предсказване на бъдещото поведение на управляваната система. Предсказаните стойности се изчисляват от модел на системата, който е съществена част на предсказващите регулатори. Това управление е единствената напреднала методология, която е успяла да намери най-широко приложение в индустрията.

Целта на тази дипломна работа е да представи един от последните варианти на обобщеното предсказващо управление. Паралелно с теоретичната формулировка на алгоритъма е изведен и начинът за програмното му осигуряване, което е същинската цел на настоящата работа. Методиката на предсказващото управление дава възможност да се изведат различни алгоритми за решаване на регулатора, като се поставят конкретни изисквания към него.

От значение е, да се докаже ефективността на предсказващия регулатор, в която трябва да се убедят изучаващите го студенти. За целта са проведени експериментални симулации, които показват неговия потенциал при различни стойности на параметрите му. За по-добро запознаване с функционирането на обобщения предсказващ регулатор, са предложени упражнения, основаващи се на изведеното програмно осигуряване. С тези упражнения студентите в лабораторните часове по “Цифрови системи за управление” могат да придобият много по-ясна представа за работата на регулатора и неговите ключови особености, без да се изискват задълбочени познания на неговата същност. Това може да бъде осъществено чрез изучаване на синтезирания материал представен в тази дипломна работа.

## Означения

Тук са представени означенията, които се използват като математически символи, оператори и съкращения.

### СИМВОЛИ

$A, B, C$	Полиноми на модела на обекта
$R, S, T$	Полиноми на затворената система
$E, F$	Полиноми необходими за решаване на Диофантовото уравнение
$G$	Тегловна матрица
$\Gamma$	Матрица съдържаща предходните стойности на управляващия сигнал
$J(u, t)$	Целева функция на управлението
$N_1$	Начало на хоризонта на предсказване
$N_2$	Край на хоризонта на предсказване
$N_u$	Хоризонт на управление
$d$	Чисто закъснение
$f$	Предсказан възстановим изход
$m, n$	Ред на числителя и знаменателя на обекта
$r_t$	Еталонен сигнал на входа във време $t$
$u_t$	Вход на обекта във време $t$
$\hat{u}_t + j k$	Предсказано управление във време $k$
$y_t$	Изход на обекта във време $t$
$\hat{y}_t + j k$	Предсказан изход във време $k$
$\alpha$	Филтър на заданието
$\lambda$	Тегловен коефициент
$\xi_t$	Стъпаловидно смущение във време $t$



---

**ОПЕРАТОРИ**

---

$A^T$	Транспонирана матрица
$q^{-1}$	Чисто закъснение в обратната връзка
$\Delta = 1 - q^{-1}$	Диференциален оператор

---

**СЪКРАЩЕНИЯ**

---

ARIMAX	Авто-регресия с интегрална пълзяща средна стойност
ARMAX	Авто-регресия с пълзяща средна стойност
GPC	Обобщено предсказващо управление
MV	Управление с минимална дисперсия
GMV	Обобщено управление с минимална дисперсия
LQ	Линеен квадратичен регулатор

# Съдържание

<b>Глава 1</b>	Теоретична постановка	1–1
1.1.	Функциониране на предсказващия регулатор .....	1–1
1.1.1.	Основен принцип на управлението .....	1–2
1.1.2.	Общ преглед на GPC регулатора.....	1–3
1.1.3.	Важни елементи в схемата на GPC регулатора.....	1–7
1.2.	Обзор на методите за проектиране на предсказващи регулатори .....	1–11
1.2.1.	Какво е предсказващо управление? .....	1–11
1.2.2.	Предназначение и цели .....	1–12
1.2.3.	Преглед на различните видове предсказващи регулатори .....	1–13
1.2.4.	Разширени модификации на GPC алгоритъма .....	1–15
1.2.5.	Кратък литературен преглед.....	1–17
1.3.	Обобщение.....	1–18
1.3.1.	Стъпки при изграждане на алгоритъма .....	1–19
<b>Глава 2</b>	Проектиране на обобщено предсказващо управление по метода на Clarke <i>et al.</i>	2–1
2.1.	Формулиране на задачата на предсказващото управление	2–1
2.2.	Начални данни .....	2–2
2.3.	Алгоритъм на управление .....	2–3
2.3.1.	Формулиране на управлението в отворения контур	2–3
2.3.2.	Формулиране на затворения контур .....	2–12
2.3.3.	Систематизиране на алгоритъма.....	2–15
2.3.4.	Избор на структурни параметри .....	2–16
2.4.	Програмно осигуряване .....	2–18
2.4.1.	Изграждане на програмно управление .....	2–18
2.4.2.	Приложение .....	2–23
<b>Глава 3</b>	Изследване на ефективността на GPC регулатора	3–1

---

3.1.	Цел на изследването .....	3-1
3.2.	Осъществяване на изследването .....	3-1
3.2.1.	Параметри на симулиране .....	3-1
3.2.2.	Резултати от изследванията.....	3-3
3.3.	Обобщение на резултатите.....	3-9
<b>Глава 4</b>	<b>Предложения за лабораторни упражнения с използване на GPC регулатора</b>	<b>4-1</b>
4.1.	Цели.....	4-1
4.2.	Осъществяване на упражнението .....	4-1
4.3.	Пример.....	4-2
4.3.1.	Симулация .....	4-3
4.3.2.	Резултати .....	4-4
4.4.	Предложения за упражнения.....	4-5
<b>Глава 5</b>	<b>Заключение</b>	<b>5-1</b>
5.1.	Резюме.....	5-1
5.2.	Литература.....	5-2

# Глава 1

## Теоретична постановка

### 1.1. Функциониране на предсказващия регулатор

Предсказващите регулатори се отнасят към самонастройващите се адаптивни регулатори. При тях параметрите на управление са променливи величини, а механизмът за настройка на тези параметри е базиран в реално време на сигналите на системата.

Основната идея на тези регулатори е да изчисляват последователност от променливи управляващи сигнали, за да оптимизират бъдещото поведение на обекта. Те се прилагат най-вече за процеси със сложна динамика, бързо-променливи процеси и т.н.

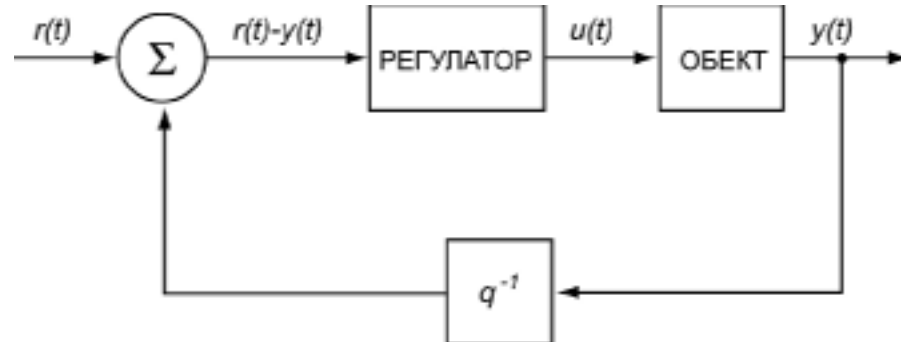
Предсказващите регулатори използват динамичен модел, за да предсказват управляващите променливи. Предсказаните управляващи променливи се оптимизират, като се минимизира съответен функционал, за да се определи действителната им стойност. Процедурата се повтаря за всеки такт, като се използват обновените данни. Обикновено, стойността на функционала зависи от квадратичната грешка между бъдещия настройваем параметър и бъдещата управляема променлива ограничени от хоризонта на предсказване.

Независимо, че GPC алгоритъма е разяснен с проста формулировка на адаптивното оптимално управление, той показва повечето характерни особености на прекалено усложнения адаптивен LQG (Linear Quadratic Gaussian Optimal Control) регулатор — тъй като и двата произлизат от оптимално-квадратичния критерий за управление. Твърде пестеливата структура на GPC критерия е постигнала бърза практическа реализация поради осъществените изисквания към по-сложните модификации и разширения.

### 1.1.1. Основен принцип на управлението

В традиционното управление принципа на отрицателната обратна връзка се използва за получаване на информация за поведението на системата в предходните моменти. Ако разполагаме със сигналите определени във фиг. 1, достъпната информация във време  $t$  е:

$$\begin{aligned} r(s) &\text{ където } s \leq t, \\ u(s) &\text{ където } s \leq t, \\ y(s) &\text{ където } s \leq t. \end{aligned}$$

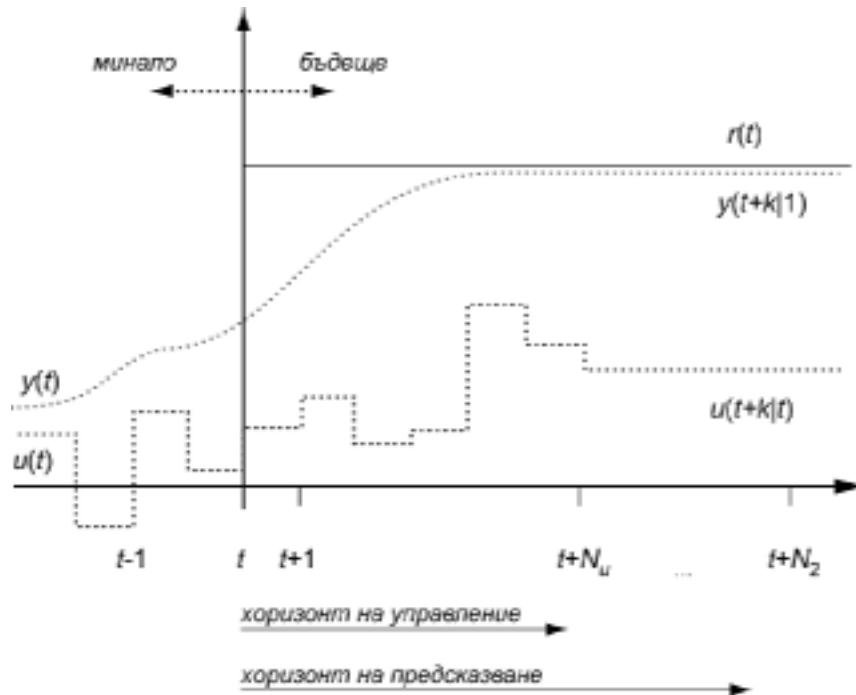


фиг. 1 Стандартна схема за управление

Като използваме тази информация и принципа на отрицателната обратна връзка можем да получим повече данни за състоянието на системата. Принципа на предсказващото управление се състои в това — да се извлече и използва повече информация.

Предсказващото управление използва вътрешен модел на реалната система, чрез който бъдещото поведение на системата може да бъде предсказано в краен интервал от време. Той обикновено се нарича *хоризонт на предсказване* и започва от текущото време  $t$ . Предсказаните стойности са функции на бъдещите входни променливи  $\hat{u}(t + k|t)$ , където  $k = 0, 1, \dots, N_u - 1$  и на текущото състояние на системата.  $N_u$  служи за обозначаване края на хоризонта на управление. Накрая се избира оптималната управляваща последователност от входящи променливи и първата стойност  $\hat{u}(t|t)$  се прилага към реалната система. В следващият период от време  $t + 1$  този процес се повтаря

отново, но вече хоризонтът на предсказване е изместен с една стъпка. Това е причината, поради която този начин на управление на системата често се нарича *отдалечен хоризонт на управление* Maciejowski (2001). На фиг. 2 е представена основната идея на предсказващото управление.



фиг. 2 Отдалечен хоризонт на управлението, основна идея

С пунктирна линия са показани предсказаните стойности на  $y(t)$  и  $u(t)$ , които се отнасят за бъдещите моменти  $t > k$ .

### 1.1.2. Общ преглед на GPC регулатора

За да бъде разбрано по-добре функционирането на GPC, е добре да се разгледа повърхностно MPC (Model-based Predictive Control) алгоритъма. В MPC, модела на обекта се използва за предсказване на ефекта на бъдещия вход върху изхода на системата в границите на хоризонта на предсказване. Управляващият вход се определя чрез минимизиране на желания критерий. Повечето MPC алгоритми се различават по структурата на използвания модел и критерия за формиране на управлението, но методите по принцип са еднакви. Изследванията могат да

бъдат намерени в De Keyser *et al* (1988), Garcia *et al* (1989), Soeterboek (1992).

По-долу е разгледано популярното *обобщено предсказващо управление* (GPC) и са представени някои от основните концепции.

Един от предшествениците на GPC е MV (Minimum-Variance) регулаторът с минимална дисперсия, описан най-напред от Åström (1970), който е поставил основите на известния самонастройващ се регулатор на Åström & Wittenmark (1990). Този регулатор минимизира разликата с  $d$ -стъпки напред, предсказвайки грешката на управление според критерия

$$J(u, t) = E\{(y_{t+1} - r_{t+1})^2\} \quad (1.1)$$

където  $y_t$  е изход на системата,  $r_t$  е задание, а с  $E(\cdot)$  е обозначено очакването. Този критерий е минимизиран във време  $t$  след избора на управляващ сигнал  $u_t$ . Във време  $t + 1$  новата задача е решена за  $u_{t+1}$ . Критерият (1.1) е основан на косвеното предположение, че моделът на обекта има чисто закъснение. Ще е безсмислено да се минимизира грешката във време  $t + 1$ , ако управлението  $u_t$  въздейства върху входа на обекта само във време  $t + d$  при  $d > 1$ . Ако обекта има закъснение  $d$ , грешката в (1.1) се замества с  $y_{t+d} - r_{t+d}$ . Ще разгледаме само случая, когато обекта е с чисто закъснение, тъй като манипулирането със закъснения не внася голяма сложност, при условие, че закъснението е известно и няма опасност от лоши последствия.

Както е известно, този метод на управление работи само с минималнофазови системи (системи с устойчиви нули) и задължително изисква мощен входен сигнал. При неминималнофазови обекти управлението достига своите стойности, като унищожават нулите на обекта (включвайки неустойчиви нули), което води до загуба на вътрешна устойчивост в затворения контур. Има подход, който кара тази стратегия да работи с неминималнофазови системи, като модифицира леко критерия (1.1), с пресмятане на корекция на управляващия сигнал, а също и на изхода. Това е показано в новия критерий

$$J(u, t) = E\{(y_{t+1} - r_{t+1})^2 + \lambda u_t^2\} \quad (1.2)$$

където  $\lambda$  е положителен тегловен коефициент. Тази стратегия е известна като GMV (Generalized Minimum Variance). Докато тя има потенциала да води до вътрешна устойчивост на управлението, това все още не е гарантирано за специфичния избор на тегловния коефициент  $\lambda$ .

Друга често срещана модификация с GMV е осъществяване на задачата за оптимизация не по  $u_t$ , а по  $\Delta u_t$ , където  $\Delta u_t$  е нарастващото управление на входа на системата,

$$\begin{aligned} \Delta u_t &= u_t - u_{t-1} \\ J(u, t) &= E\{(y_{t+1} - r_{t+1})^2 + \lambda (\Delta u_t)^2\} \end{aligned} \quad (1.3)$$

Във време  $t$ , този критерий се минимизира по отношение на  $\Delta u_t$ . Причината за използване на нарастващи входни сигнали в този критерий е, че (1.2) не допуска нулева статична грешка в случай на ненулево константно задание, освен ако отворения контур на обекта не съдържа интегратор, който би позволил  $y_t$  да остане ненулева константна стойност с управляващ вход започващ от нула. Стандартният GMV алгоритъм винаги коригира ненулевото  $u_t$ , дори в случай на поддържане на ненулево задание. Една от причините за използване на  $\Delta u_t$  в закона за управление е, че определеното ограничаване на  $u_t$  може да бъде трудно получено. Идеята е главно с цел да запази  $\lambda$  колкото е възможно по-малко за да поддържа изходните променливи минимални докато все още затворения контур е устойчив. Положителното тегло  $\lambda$  се включва с цел да се избегне рязкото нарастване на управляващите сигнали.

Тъй като това управление често се проваля при някои неустойчиви и неминималнофазови системи и често при системи с променливо закъснение, е направено допълнително разширение от Clarke *et al*, наречено *обобщено предсказващо управление* (GPC), което се минимизира със следния критерий:

$$J(u, t) = E\left\{ \sum_{j=N_1}^{N_2} [y_{t+j} - r_{t+j}]^2 + \lambda \sum_{j=1}^{N_u} [\Delta u_{t+j-1}]^2 \right\} \quad (1.4)$$



като  $\Delta u_{t+j} = 0, j = N_u, \dots, N_2$

Тази минимизация извежда  $\Delta u_t, \Delta u_{t+1}, \dots, \Delta u_{t+N_u-1}$ , но всъщност само  $\Delta u_t$  се прилага действително. Във време  $t + 1$  минимизацията се решава отново. Тази процедура се нарича *отдалечен хоризонт на управление*.

Причината за тази модификация на управлението може да се интерпретира като подобрене на MV и GMV методите, като се опитва да наложи решение предоставящо както адекватно действие, така и асимптотична устойчивост за широк клас обекти. Оптимизацията се извършва от няколко  $N_2 - N_1 + 1$  последователни бъдещи изходни стойности, като се вземат предвид и  $N_u$  бъдещите нарастващи управляващи действия. По презумпция, този алгоритъм ще работи в случай че “реалното” закъснение е включено в интервала между  $N_1$  и  $N_2$ . Обикновено  $N_1$  се избира като долна граница и в действителност не се използва като структурен параметър. Включването на няколко бъдещи управляващи стойности се прави за да може по-продължителното наблюдение на сигнала да намалее в случай, че стане неограничен. Прибавянето на ограничение върху нарастващото управление след определено време се рационализира, като се насърчи регулатора да достигне характеристиките си бързо, но на базата на бъдещите изчисления на решението.

Забележителното в критерия (1.4) е, че притежава значимо ниво на сложност достатъчно, да направи ефективно управлението на широк клас обекти, тъй като критерият зависи само от три главни параметъра  $N_2, N_u$  и  $\lambda$ .

Тези особености дават възможност на GPC регулатора да се справи ефективно с неизвестно или променливо закъснение, а също и да осигури устойчива работа при неминималнофазови дискретни модели. Ето част от ситуацияите, с които GPC може да се справи без да са му необходими специфични модификации:

- Неминималнофазови обекти — при достатъчно бързи честоти и при непрекъсната предавателна функция;
- Неустойчивост в отворения контур или обекти с лошо амплитудно колебание;

- Неизвестно или променливо закъснение — много традиционни самонастройващи методи като MV са прекалено чувствителни към оценяване на чистото закъснение;
- Обекти с по-висок ред — GPC се справя с анулирането на нулите и полюсите без проблеми.

Тази хронологията ясно показва връзката между GPC и MV управлението и обяснява защо във всяко следващо използване на този алгоритъм малките стойности на  $\lambda$  се препоръчват. За адаптивната същност на този алгоритъм е казано съвсем малко. Това не е изненадващо, когато се разглежда MV регулатора, който фактически е представен като неадаптивен, но той е повече присъщ за другите два алгоритъма, които са специализирани за адаптивно управление. Както беше казано по-горе, адаптивната същност на GMV и GPC няма особена роля. За по-лесно обяснение, GPC алгоритъма се описва като неадаптивен, но се има предвид, че той е с адаптивна същност в завършения алгоритъм.

Поради тези особености GPC има такава популярност, за което повечето нови предсказващи методи произлизат от него.

### 1.1.3. Важни елементи в схемата на GPC регулатора

#### 1.1.3.1. Модел на системата

GPC методът може да бъде приложен както към SISO (един вход, един изход), така и към MIMO (много входове, много изходи) системи. За опростяване на описанието ще бъде разгледано действието на алгоритъма върху SISO системи. Основният GPC алгоритъм се базира на ARIMAX модел, представен по следния начин:

$$A(q^{-1})y_t = B(q^{-1})u_{t-1} + \frac{C(q^{-1})}{\Delta} \xi_t \quad (1.5)$$

където  $u_t$ ,  $y_t$  и  $\xi_t$  са вход на обекта, изход на обекта и стъпаловидно смущение, еквивалентно във време  $t$ , а  $A$ ,  $B$  и  $C$  са полиноми с оператор на чистото закъснение  $q^{-1}$ .

Ролята на делта оператора  $\Delta = 1 - q^{-1}$  е да осигури интегрално действие на регулатора, за да се предотврати (компенсира) влиянието на нежеланите ефекти от появата на стъпални (товарни) смущения върху обекта. Стъпаловидното смущение  $\xi_t$  може да бъде детерминиран или стохастичен сигнал, но заради делта оператора, неговата средна стойност се приема, че е нула. При реализацията на GPC алгоритъма, се приема също, че  $\xi_t$  е стохастичен сигнал, а полиномът  $C(q^{-1})$  се смята, че е устойчив, тъй като само спектралната характеристика на сигнала  $(C(q^{-1})/\Delta)\xi_t$  влияе върху предсказването на изхода  $y_t$ . Често полинома на шума  $C(q^{-1})$  се разглежда като структурна променлива и затова е наречен наблюдателен полином  $T(q^{-1})$ . Този полином влияе върху смущението и робастността по отношение на немоделираната динамика на регулатора. Ролята на наблюдателния полином  $T(q^{-1})$  е анализирана от Robinson & Clark (1991), Mohtadi (1988), Ordys (1993). Съставката на шума в това описание не е стационарна. Причината за избора на такова описание е за да се подсили интегралното действие в регулатора с цел да се справи с товарните смущения, които са неделима част от индустриалните приложения.

Същността на (1.5) е да моделира изхода сякаш е изопачен от въздействието на добавъчния случаен шум. Въпреки, че това не е реалистичен модел, ефектът от произлизащите от него управляващи въздействия е да увеличи способността да елиминира стъпаловидните смущения на изхода. Това се наблюдава при процесите на управление в промишлеността, където тези закони се прилагат успешно. GPC включва този модел на шума в своята формулировка, за да формира адекватен отговор за конкретните обстоятелства. Това е принципът на *вътрешния модел*.

### 1.1.3.2. Критерий за формиране на оптималното управление

С това описание на модела, отчасти ограничения квадратичен критерий е поставен в рамките на нарастващите входове и изходи (а не само по състоянието) на обекта. Целевата функция (1.4) се минимизира, където  $r_t$  е заданието, към което желаем да се доближава изхода на

системата, а  $N_1$ ,  $N_2$  и  $N_u$  са хоризонтите на предсказване и управление, в които ще бъдат изчислени предсказаните стойности на входния и изходния сигнал на системата. Положителният тегловен коефициент усилва управлението и увеличава грешката. Ползата от (1.4) е да покаже, че избраните стойностите на управлението са изчислени въз основа на наличната информация във време  $t$  и предполагат наличието на модел на стохастичния шум. Поради тази структура на GPC управлението, се достига да решаване на стандартната оптимална задача на *отдалечения хоризонт*.

### 1.1.3.3. Хоризонт на управление

Важен елемент от GPC е хоризонтът на управление  $N_u$ . Управляващите сигнали във време  $t + (N_2 - 1)T$  имат ефект върху изхода във време  $t + N_2T$  (без в случая да се отчита закъснението). Вместо да се допусне всички тези управляващи сигнали да бъдат свободни, се приема, че след интервала  $N_u$  нарастването на управлението е нула. Това се отразява върху сложността на минимизационната процедура, като броя на свободните оптимални променливи се редуцира до  $N_u + 1$ . В случая, при  $N_u = 0$  имаме само един свободен управляващ сигнал  $u_t$ .

Както е дискутирано в Clark *et al* (1987),  $N_u$  е изключително важен структурен параметър. Общо казано, колкото по-сложен е обекта, толкова по-висока стойност е необходимо да има  $N_u$ , за да бъде приемливо управлението. За по-прости системи, изборът на  $N_u = 0$  е най-адекватното решение.

### 1.1.3.4. Бъдещо задание

Важна особеност на предсказващото управление е възможността за задаване на определена траектория на движение. При изчисляване на текущото управление на входа е възможно да се използва информацията в бъдещото задание върху хоризонта на предсказване. Такива случаи се срещат често при управление на работи или компютърно управление на машини, манипулатори, а също и химични и топлинни процеси.

Ако бъдещото задание не е известно, трябва да бъде генерирана подходяща еталонна траектория. Има няколко начина:

- Предсказване на заданието — основава се на модел на заданието, който може да бъде предсказан;
- Генериране на изгладена траектория към заданието посредством еталонния модел;
- Пряко предположение за бъдещо задание — общото предположение е, че бъдещето задание е константа и е равна на текущото задание.

Нататък ще допуснем, че бъдещето задание е известно.

#### 1.1.3.5. Стратегия на отдалечения хоризонт

GPC регулатора, както и повечето предсказващи регулатори са изградени според така наречения хоризонт на предсказване, по следния ред:

- Изчисляване на последователността на бъдещото задание;
- Намиране на предсказаните изходи изразени посредством неизвестната бъдеща входна последователност. Тоест, намиране на  $\hat{y}(t + N_1), \dots, \hat{y}(t + N_2)$ , като функция на минали стойности и бъдеща последователност от сигнали на управлението  $u_t, u_{t+1}, \dots, u_{t + N_u}$
- Чрез минимизацията на критерия на управление, се намира оптималната последователност от бъдещи входове;
- Прилага се първия елемент  $u_t$  към обекта.

Като се използва отдалечения хоризонт на предсказване може да се види начина на коригиране на предсказания изход по отношение на шума или грешките. Това дава възможност да се извлече полза от новата информация, получена за всеки модел главно, чрез измерване на изхода на системата. Друга информация като детерминирани смущения, промени в заданието или твърдите ограничения на входа или изхода също така могат да бъдат коригирани.

С независим от времето обект, ефективният GPC алгоритъм е стационарен, когато е приложен принципът на отдалечения хоризонт (с линеен обект и без твърди ограничения). Това е в разрез с например, LQ регулатора с краен хоризонт, при който състоянието по обратна връзка е време–зависимо. Следователно, като се използва отдалечен хоризонт на управление, се получават време–независими регулатори, със стационарен безкраен хоризонт на управление, но с критерий на краен хоризонт. Виж Bitmead *et al* (1990) за повече информация.

Следва че, GPC управлението е управление в отворения контур, тъй като  $N_u$  бъдещите нарастъци на управлението са изчислени изключително, чрез минимизацията (1.4). Във време  $t$  единствено решение на оптималната задача с критерий  $J(u, t)$  на управлението  $\{\Delta u_{t+j}, j = 0, \dots, N_u - 1\}$  се прилага само върху първия елемент  $u_t = u_{t-1} + \Delta u_t$ . Новата задача е решена във време  $t + 1$  с критерий  $J(u, t + 1)$  и решение  $\{\Delta u_{t+1+j}, j = 0, \dots, N_u - 1\}$ , като е използвано само управлението  $u_{t+1}$ . Това пререшаване на оптимизационната задача във всеки момент показва как тази частична стратегия в отворения контур, всъщност е осъществена в затворения контур. Механизмът за затварянето на контура е включването на всички предишни входове на обекта и изходна информация в генерирането на предсказването.

## 1.2. Обзор на методите за проектиране на предсказващи регулатори

### 1.2.1. Какво е предсказващо управление?

Предсказващите методи в адаптивното управление се отнасят за набор от формулировки на структурата на управление, които поставят критериите на управление от гледна точка на предсказването на бъдещите входове и изходи на обекта за дадено време. Тъй като това предсказване е много трудно при голяма разлика във времето, този критерий се ограничава, чрез *оптимален хоризонт* на продължителността на управлението.

Историята на предсказващите методи в адаптивното управление е примесена с много предложения за тяхното използване в различни независими модификации. Всеки един от тези методи, като цяло има положителни особености, които се отличават по своята структурна същност от предшестващите ги методи — решение на задачата за краен хоризонт, като задача за отдалечен хоризонт, включване на предсказани изходи на обекта, задаване на по-малък брой структурни параметри обединени до различна степен с динамиката на затворения контур.

Най-разпространената версия на GPC алгоритъма, е изведената от Clarke, Mohtadi & Tuffs (1987). Съкращението GPC е прието, като всеобщо название за цял клас предсказващи регулатори в адаптивното управление. GPC може да се разглежда, както специфичен алгоритъм, така и като название на цял клас от свързани методи.

## 1.2.2. Предназначение и цели

Предсказващите регулатори са широко използвани в промишлеността още в края на 70-те години. В началото те са били разработени, за да изпълняват специализирани потребности в управлението на електростанции и нефтени рафинерии. Сега те намират много голямо приложение в редица области — химическа, хранителна промишленост, автомобилостроене, космически технологии и др.

Най-важните задачи на предсказващия регулатор са:

- Да предотвратява смущенията на входа и изхода;
- Да привежда управлението към устойчиви оптимални стойности (динамична оптимизация на изхода);
- Да привежда настройващите променливи към устойчиви оптимални стойности (динамична оптимизация на входа);
- Да предотвратява случайните процеси, когато сигналите са некоректни и да се справя с управлението на обекта доколкото е възможно.

Съществена особеност на предсказващите регулатори е, че те могат да се справят успешно със синтеза на оптимални управляващи въздействия при предварително зададено изменение на заданието. Чрез използване на отдалечен хоризонт на предсказване могат да се генерират изпреварващи управляващи сигнали, които да осигуряват плавно изменение на изходния сигнал в предходните режими, за да се приближи системата до желаното поведение.

### 1.2.3. Преглед на различните видове предсказващи регулатори

Съществуват много видове предсказващи регулатори, основани на общи идеи, като:

- CGPC (Continuous-time Predictive Control)
- CRHPC (Constrained Receding Horizon Predictive Control)
- DMC (Dynamic Matrix Control)
- EHAC (Extended Horizon Adaptive Control)
- EPSAC (Extended Prediction Self-Adaptive Control)
- GMV (General Minimum-Variance Controller)
- GPC (Generalized Predictive Control)
- GPCW (Generalized Predictive Control with end-point state weighting)
- IDCOM (Identification and COMmand)
- IMC (Internal Model Control)
- MAC (Model Algorithmic Control)
- MPC (Model-based Predictive Control)
- MURHAC (Multipredictor Receding Horizon Adaptive Control)
- MUSMAR (Multistep Multi-variable Adaptive Control)
- MV (Minimum-Variance Controller)



- PFC (Predictive Functional Control)
- PSMRC (Partial State Model Reference Control)
- RHIMC (Receding Horizon Internal Model Control)
- SIORHC (Stabilizing I/O Receding Horizon Control)
- UPC (Unified Predictive Control) и др.

Всеки от тези методи има особености, с които се различава от другите, например: решаване на проблема за оптимизацията във всеки текущ момент, чрез стратегията на отдалечения хоризонт; обединяване на предсказания изход на обекта; доставянето на по-малък брой стойности на изчислените параметри и т.н.

Като най-успешни предсказващи регулатори, могат да споменат GPC, MUSMAR, MURHAC, CRHPC, SIORHC.

Различните алгоритми са формулирани в зависимост от — тип на разглежданото смущение, критерий на формулиране, използване на филтри за постигане на необходимата работна характеристика, предположение за бъдещото действие на управляващите сигнали и др.

Въпреки, че GPC е много мощен алгоритъм за управление, той има и сериозни недостатъци, тъй като в процеса на предсказване регулатора смесва детерминирани и стохастични предсказани стойности. Тъй като модела следва предавателната функция  $G(q^{-1})$  и внася смущения в Тейлоровия (наблюдаващ) полином  $T(q^{-1})$ , се използват различни методи за разрешаване на този проблем.

UPC (Unified Predictive Control), например, добавя към GPC динамично усилване на управлението. От друга страна, характеристиката на управление на GPC започва с много настройващи параметри, които също влияят и на UPC. Ето защо PSMRC (Partial State Model Reference Control) е създаден, за да добави към UPC и GPC възможност за напълно независими параметри. Предсказващите регулатори на базата на IMC (Internal Model Control) и DMC (Dynamic Matrix Control) имат преимущество при отделянето на оценката на смущенията от изчислените стойности на управлението.

При оценяването на различните предсказващи регулатори, може да се състави следния ред на по-важните им особености:

- Създаване на структура, която напълно да отделя заданието и измерва смущенията;
- Отделно определяне на желаните характеристики на следене, предсказване и регулиране;
- Отделяне на стохастичния предиктор от регулатора;
- Използване на вътрешния модел на управляващата схема за отделяне на оценката на смущенията от отклонението;
- Използване на пространство на състоянията за по-лесно унифициране на двата едно–променливи и много–променливи регулатора, за да е възможно използването на параметрични модели на процесите и неизмеримите състояния.

## 1.2.4. Разширени модификации на GPC алгоритъма

По-долу са описани някои от по-важните модификации на GPC алгоритъма.

### 1.2.4.1. Гарантиране на устойчивост

Основният GPC алгоритъм в общия случай не може да осигури номинална устойчивост. Съществуват различни модификации, които могат да се справят с този проблем. Така нареченият *Stable GPC* (SGPC) алгоритъм, предложен от Kouvaritakis *et al.*, работи като прилага GPC управление след затварянето и стабилизирането на обратната връзка около управляваната система. *Multivariable* (MUSMAR) регулатора предложен от Kouvaritakis & Rossiter (1993) и стабилизиращият *I/O Receding Horizon* (SIORHC) регулатор осъществяват устойчивост, чрез добавяне на нулево ограничение на крайното състояние, т.е. крайната стойност на изхода трябва да достигне до желаното задание. Подобна идея е използвана в *Constrained Receding Horizon Predictive Control* (CRHPC).

#### 1.2.4.2. Твърди ограничения

Едно от предимствата на предсказващото управление е, че твърдото ограничение може да бъде управлявано по систематичен начин в задачата за оптимално управление, като използва принципа на отдалечения хоризонт. QP (Quadratic Programming) е разширен метод на GPC, който се доближава към решение, чрез ограничение на скоростта на входните сигнали.

#### 1.2.4.3. Адаптивно предсказващо управление

MPC (Model-based Predictive Control) разчита на съществуването на оценен модел на системата, в смисъл, че предсказването на изходите може да бъде изчислено. Следователно, такива алгоритми са директно приложими в косвени адаптивни схеми, в които оценяващия алгоритъм е комбиниран със структурната схема на управление, вж. Åström & Wittenmark (1989).

Някои предположения в разработването на адаптивно обобщено предсказващо управление, където управляващите параметри се оценяват пряко са описани от Wang (1994), Wang & Henriksen (1994), Wang & Li (1995).

Различен подход за адаптивно предсказващо управление е MUSMAR (Multi Step Multi-variable Adaptive Control) алгоритъма, развит от Mosca *et al* (1989), Mosca (1995), Chisci *et al* (1994). MUSMAR се състои от поредица паралелни RLS идентификатори, като всеки от тях има предсказващ модел на изхода на системата за хоризонта на предсказване. Използването на такава мултипредикторна структура има предимството да бъде по-малко чувствителна към не-моделирани динамики, тъй като предсказаните стойности не са изчислени косвено от единичния модел.

#### 1.2.4.4. Нелинейно предсказващо управление

Концепцията на предсказващото управление може да бъде приложена успешно и върху нелинейни модели на обекта. Оптимизационната задача също се явява нелинейна и изчислителният подход зависи от структурата на модела. Алгоритъмът базиран на модела на Hammerstein е

представен в Zhu *et al* (1999). Теоретичен анализ на основните проблеми може да бъде намерен в Henson & Seborg (1993).

#### 1.2.4.5. Непрекъснатото предсказващо управление

Непрекъснатата версия на GPC, също така наречена CGPC е представена от Demircioğlu & Gawthrop (1991). CGPC алгебрично е подобен на дискретния GPC алгоритъм, но интерпретацията на структурните променливи е различна. Двете модифицирани версии на CGPC – CGPC\_C и CGPC\_W са представени от Demircioğlu & Clark (1992) CGPC\_C включва ограничение подобно на дискретния CRHPC алгоритъм. CGPC\_W включва претегляне на крайната точка на състояние в целевата функция подобно на GPCW алгоритъма. Двата алгоритъма имат свойства, които гарантират устойчивост.

#### 1.2.5. Кратък литературен преглед

Зараждането на идеята за предсказващо управление взема място в края на 70-те години. В литературата работите на Richalet *et al* (1978), често се отнасят към първия алгоритъм на предсказващото управление. През 1979 г. друг алгоритъм наречен Dynamic Matrix Control или DMC е бил предложен от Cutler *et al*, (1983). Освен тези два алгоритъма съществува голямо разнообразие от различни варианти на концепцията за предсказващото управление, които бяха описани по-напред в тази глава.

Всеки от тези алгоритми използва една и съща основна идея на предсказващото управление – ясно формулиран вътрешен модел, стратегия на отдалечения хоризонт, изчисляване на управляващите сигнали, чрез оптимизиране на предсказания изход и т.н. Разликите между споменатите методи са детайлни, например, използването на различни модели на реалната система.

В литературата съществува разнообразно и богато описание на методите за предсказващо управление. Из между най-скоро публикуваните трудове са тези на: Mayne *et al* (2000), Rawlings (2000), Allgöwer *et al* (1999), Morari & Lee (1999).

Отношение към различните методи на предсказващото управление може да бъде открито почти във всяко въведение на статиите третиращи тази задача.

Съществуват и други книги, които могат да бъдат обсъдени. В “*Predictive Control with Constraints*”, Maciejowski (2001), чрез трактуване на темата дава вероятно най-добрата книга, с обяснение на общите идеи на предсказващото управление. В едно от приложенията към тази книга е даден интересен преглед на съществуващите комерсиални продукти за предсказващо управление, които показват важните методи използвани в индустрията. Друга книга, която покрива повечето важни идеи е Samacho & Bordons (1998), тъй като тя е фокусирана главно върху GPC управлението.

### 1.3. Обобщение

Главните идеи илюстрирани дотук могат да бъдат обобщени, както следва:

- GPC алгоритъма е бил разработен като разширение на MV регулатора;
- Този метод за управление се използва в различни индустриални приложения и е показал добри характеристики, като осигурява добра степен на робастност по отношение на свръх параметризация или към променливо закъснение;
- Тези алгоритми са нищо друго освен краен хоризонт на предсказване с квадратичен критерий за минимизация, който може да бъде преработен в LQG форма на управление;
- Теоретическият анализ е с цел да се оцени влиянието на структурните променливи (хоризонти на предсказване и управление, тегловни коефициенти) върху устойчивост на затворената система.

### 1.3.1. Стъпки при изграждане на алгоритъма

В аналитичната част на тази дипломна работа е разгледан процедурата за осъществяване на програмно управление на GPC регулатора. Последователността, в която е изведен алгоритъма е следната:

1. Дефинира се някаква желана траектория, до която трябва максимално да се доближава изходът на системата. Най-често (както и тук) се приема, че заданието е единица;
2. Дефинира се ARIMAX модел на обекта (2.1) и се преобразува в ARMAX (2.2);
3. Формира се критерий за определяне на оптималното управление на обекта (2.3);
4. Прилага се стратегията на отдалечения хоризонт, при което се налага решаването на Диофантовото уравнение (2.6) чрез използване на рекурсивна процедура (гл. 2.3.1.4);
5. Формулира се затворения контур (гл. 2.3.2);
6. Задачата се решава програмно. За по-добра на демонстрация на алгоритъма, задачата е решена теоретично паралелно с програмното описание (гл. 2.4).

## Глава 2

### Проектиране на обобщено предсказващо управление по метода на Clarke *et al.*

#### 2.1. Формулиране на задачата на предсказващото управление

В предната глава идеите на предсказващото управление бяха разгледани твърде общо. Тази глава ще даде по-ясна формулировка на GPC метода, като използва математически подход. Най-напред ще бъде представена стратегията на отдалечения хоризонт, изразяваща се със следната последователност:

1. Във време  $t$  предсказваме изхода от системата  $\hat{y}_{t+j|k}$ , където  $k = N_1, N_1 + 1, \dots, N_2$ . Тези изходи ще зависят от бъдещите управляващи сигнали  $\hat{u}_{t+j|k}, j = 0, \dots, N_u$ ;
2. Избира се критерий базиран на тези променливи и се оптимизира по отношение на  $\hat{u}_{t+j|k}, j = 0, \dots, N_u$ ;
3. Прилага се  $u_t = \hat{u}_{t|k}$ ,
4. Във време  $t + 1$  алгоритъмът се повтаря отново от точка 1.

Оптимизационният критерий в алгоритъма може да приеме различни форми, но най-често се използва квадратичен критерий, който минимизира дисперсията между предсказаните изходи и заданието за хоризонт на предсказване, който започва от  $t + N_1$  и завършва в  $t + N_2$ . Той също минимизира и промените в управляващия сигнал на базата на *хоризонта на управление*, който започва от  $t$  време до  $t + N_u$ . Подходящият вид на тази минимизация може да бъде постигнат, чрез използване на тегловната матрица  $G$ , която в повечето случаи е диагонална.

За избора на модел на обекта се ползват параметрични модели от типа ARIMAX. Това се налага, поради по-малкият им брой параметри, което облекчава изчислителната процедура, както при оценяването, така и при използването им за синтез на управлението. Единствената трудност, свързана с проектирането им, е необходимостта от решаване на Диофантовото уравнение.

## 2.2. Начални данни

Общата характеристика на предсказващите регулатори може да се представи със следните показатели:

- Входно–изходни данни от обекта, достъпни за измерване до текущия момент  $k$  от времето:

$$[u_{k-1}, u_{k-2}, \dots], [y_k, y_{k-1}, y_{k-2}, \dots];$$

- Предсказването се извършва в бъдещ интервал от време  $[k, k+N_2]$  наречен хоризонт на предсказване, който надхвърля чистото закъснение на обекта  $d$ -такта.
- Чрез предсказващия регулатор могат да се генерират изпреварващи управляващи сигнали  $[u_k, u_{k+1}, u_{k+2}, \dots]$ , ако предварително е известно поведението на заданието в бъдещия интервал  $[r_k, r_{k+1}, r_{k+2}, \dots]$ , или на желаната траектория за отработване на заданието, която най-често се създава в качеството на филтър за формиране на плавно отработване на заданието, като звено от първи ред с единичен коефициент за усилване:

$$W_R(q^{-1}) = \frac{(1-\alpha)q^{-1}}{1-\alpha q^{-1}}$$

- Проектираното оптимално управление минимизира функционал  $J$  на грешката между предсказаната стойност на изходната величина и заданието  $r$ .



- Предполага се, че заданието се отработва след  $k + N_u$  такта,  $\Delta u_{k+i} = 0$  за  $i > N_u$
- Всички предсказващи регулатори използват стратегията на отдалечения хоризонт, изразен с алгоритъма (гл 2.1).

Многократното решаване на оптимизационната задача във всеки момент на бъдещия интервал показва, как тази стратегия на оптимизация в отворен контур на практика се извършва в затворен контур, чрез обратна връзка от грешката между и.

## 2.3. Алгоритъм на управление

### 2.3.1. Формулиране на управлението в отворения контур

#### 2.3.1.1. Модел на обекта на управление

Както вече беше споменато, за проектирането на този тип регулатори се използват параметрични модели от типа на ARIMAX, който се описва по следния начин

$$A(q^{-1})y_t = B(q^{-1})u_{t-1} + \frac{C(q^{-1})}{\Delta} \xi_t \quad (2.1)$$

Делта операторът  $\Delta = (1 - q^{-1})$  осигурява интегрално действие на регулатора, като премахва нежеланите ефекти и появата на товарни смущения върху обекта.

Моделът описан в уравнението (2.1) се преобразува в ARMAX по отношение не на самите сигнали, а на техните първи разлики

$$A(q^{-1})\Delta y_t = B(q^{-1})q^{-d}\Delta u_{t-1} + C(q^{-1})\xi_t \quad (2.2)$$

тъй като

$$\begin{aligned}\Delta y_t &= (1 - q^{-1})y_t = y_t - y_{t-1} \\ \Delta u_t &= (1 - q^{-1})u_t = u_t - u_{t-1}\end{aligned}$$

### 2.3.1.2. Целева функция на управлението

Минимизацията на целевата функция става, чрез следния критерий

$$J(u, t) = E \left\{ \sum_{j=N_1}^{N_2} [y_{t+j} - r_{t+j}]^2 + \lambda \sum_{j=1}^{N_u} [\Delta u_{t+j-1}]^2 \right\} \quad (2.3)$$

Тегловният коефициент  $\lambda$  ( $\lambda_{\max} < \lambda < 0$ ) осъществява компромиса между постигане на точност при отследяване на желаната траектория ( $\lambda \rightarrow 0$ ) и ограничаване на поведението на управляващия сигнал ( $\lambda \rightarrow \lambda_{\max}$ ). Освен това при наличие на чисто закъснение  $d > 1$ , нулевата му стойност осигурява и устойчива работа на системата.

За да бъде решена оптимизационната задача е необходимо да се определят предсказаните стойности на изходната променлива  $[y_{t+N_1}, y_{t+N_1+1}, \dots, y_{t+N_2}]$ .

### 2.3.1.3. Формулиране на предсказания изход

За да се реши задачата поставена при минимизацията на целевата функция (2.3), трябва да се изчислят с  $j$ -стъпки напред предсказаните стойности на изхода  $y_{t+j}$ , за  $j = N_1, \dots, N_2$ , основаващи се на информацията известна във време  $t$  и на бъдещите стойности на управлението, които по-нататък трябва да бъдат избрани, така че GPC критерия  $J$  да е оптимизиран.

Нека (2.1) се запише във вида

$$y_t = \frac{B(q^{-1})q^{-d}}{A(q^{-1})}u_{t-1} + \frac{C(q^{-1})}{A(q^{-1})\Delta} \xi_t \quad (2.4)$$

Полиномите пред  $\xi_t$  в (2.4) се представят с

$$\frac{C(q^{-1})}{A(q^{-1})\Delta} = E_j(q^{-1}) + \frac{q^{-j}F_j(q^{-1})}{A(q^{-1})\Delta} \quad (2.5)$$

Предсказването се осъществява, чрез многократно решаване на Диофантовото уравнение:

$$C(q^{-1}) = E_j(q^{-1})A(q^{-1})\Delta + q^{-j}F_j(q^{-1}) \quad (2.6)$$

Където всички променливи са полиноми в  $q^{-1}$ , а  $\deg(E_j) = j - 1$ . Въз основа на (2.2), след като преместим определените аргументи в  $q^{-1}$ , се получава

$$y_{t+j} = \frac{B}{A}u_{t+j-1} + E_j\xi_{t+j} + \frac{F_j}{A\Delta}\xi_t \quad (2.6.a)$$

Ако заместим  $\xi_t$  от израза (2.2), уравнението приема вида

$$y_{t+j} = \frac{F}{C}y_t + \frac{E_jB}{C}\Delta u_{t+j-1} + E_j\xi_{t+j} \quad (2.6.b)$$

където последният член съдържа информация, която е независима от сигналите изчислени във време  $t$ . Очевидно е, че предсказването с минимална дисперсия (MV) на  $y_{t+j}$  даващо данните известни във време  $t$  е получено чрез заместване на последния член с нула в новото уравнение,

$$\hat{y}_{t+j} = \frac{F_j}{C} y_t + \frac{E_j B}{C} \Delta u_{t+j-1} \quad (2.7)$$

В този израз  $\hat{y}_{t+j}$  е функция на известните стойности на сигнала във време  $t$ , а също и на бъдещите стойности на управлението, които вече са били изчислени.

#### 2.3.1.4. Рекурсивна процедура за формиране на неизвестните полиноми

Възможно е използването на рекурсивна процедура за формиране на неизвестните полиноми на Диофантовото уравнение, при която полиномите  $E_{i+1}$  и  $F_{i+1}$  се получават от полиномите  $E_i$  и  $F_i$ , като се постига значително съкращаване на изчисленията. За извеждане на рекурсията, Диофантовото уравнение (2.6) се записва последователно за  $i$  и за  $i + 1$

$$\begin{aligned} 1 &= E_i + A\Delta + q^{-i} F_i, \\ 1 &= E_{i+1} A\Delta + q^{-(i+1)} F_{i+1} \end{aligned} \quad (2.8)$$

като  $i = 1, 2, \dots, N_u$ . Като извадим двете уравнения се получава

$$0 = A\Delta(E_{i+1} - E_i) + q^{-i}(q^{-1}F_{i+1} - F_i) \quad (2.9)$$

Полиномът  $R = E_{i+1} - E_i$  е от  $i$ -та степен и е равен

$$R = R_1 + r_i q^{-i} \quad (2.10)$$

Където  $r_i = e_{i+1,i}$  е коефициентът пред най-високата отрицателна степен на  $E_{i+1}$ , а  $i$  е номер на рекурсията. Като се отчете (2.10), изказа (2.9) може да се запише

$$0 = A\Delta R_1 + q^{-i}(q^{-1}F_{i+1} - F_i + r_i A\Delta) \quad (2.11)$$

За да бъдат първите  $i-1$  отрицателни степени в дясната страна на (2.11) равни на нула при произволен полином  $A\Delta$ , полиномът  $R_1$  също трябва да е равен на нула, което означава, че всички коефициенти на  $E_i$  са коефициенти и на  $E_{i+1}$ , т.е.

$$E_{i+1} = E_i + r_i q^{-i} \quad (2.12)$$

Понеже изразът в скобите на (2.11) също трябва да е равен на нула, за  $F_{i+1}$  се получава

$$F_{i+1} = q(F_i - r_i A\Delta) \quad (2.13)$$

Формулите (2.12) и (2.13) дават рекурентните съотношения за определяне на полиномите  $E_i$  и  $F_i$ , като остава само да се определи коефициентът  $r_i$ . Стойностите му се получават, като се има предвид, че полиномът  $F_{i+1}$  не трябва да има положителна степен на  $q$ , т.е.  $f_{i0} - r_i = 0$ , тъй като  $a_0\Delta = 1$ . Следователно

$$r_i = f_{i0} \quad (2.14)$$

За съставяне на рекурсивната схема са необходими  $E_1$  и  $F_1$ . Те са:  $E_1 = 1$  и  $F_1 = q(1 - A\Delta)$ . Следващите стъпки  $i = 1, 2, \dots, N_u - 1$  се получава, чрез алгоритъма

$$\begin{aligned} e_{i+1,j} &= e_{i,j}, & j < i, \\ e_{i+1,j} &= f_{i,0}, \\ f_{i+1,j} &= f_{i,j+1} - e_{i+1,i} a_{j+1} \Delta, & j = 0, 1, \dots, n \end{aligned} \quad (2.15)$$

От (2.15) се извежда процедурата за оценяване на  $F_i G_i = E_i B$ .

След заместване на (2.5) в (2.4) се достига до предсказващия модел

$$y_{i+j} = \frac{B(q^{-1})q^{-d}}{A(q^{-1})} u_{i+j-1} + \frac{F_j(q^{-1})}{A(q^{-1})} \xi_t + E_j \xi_{t+j} \quad (2.16)$$

В който процесът на грешката  $\xi_t$  може да се замени с изведения от (2.1) израз

$$\xi_t = \frac{A(q^{-1})\Delta}{C(q^{-1})} y_t + \frac{B(q^{-1})q^{-d}\Delta}{C(q^{-1})} u_{t-1} \quad (2.16.a)$$

така че

$$y_{t+j} = \frac{F_i(q^{-1})}{C(q^{-1})} y_k + \frac{B(q^{-1})q^{-d}}{A(q^{-1})} \left[ 1 - \frac{q^{-i}F_i(q^{-1})}{C(q^{-1})} \right] u_t + E_j \xi_{t+j} \quad (2.16.б)$$

Изразът се обработва, така че

$$y_{t+j} = \frac{F_i(q^{-1})}{C(q^{-1})} y_k + \frac{E_i(q^{-1})B(q^{-1})}{C(q^{-1})} \Delta u_{t+j-d-1} + E_j \xi_{t+j} \quad (2.16.в)$$

Исходната величина може да се представи с две компоненти

$$y_{t+j} = \hat{y}_{t+j|k} + \tilde{y}_{t+j} \quad (2.17)$$

Първата компонента е така наречения *възстановим изход*,

$$\hat{y}_{t+j|k} = \frac{F_i(q^{-1})}{C(q^{-1})} y_k + \frac{E_i(q^{-1})B(q^{-1})}{C(q^{-1})} \Delta u_{t+j-d-1} \quad (2.18)$$

може да се предскаже въз основа на входно–изходните данни от момента  $t$  посредством синтезирания управляващ сигнал след време  $t$ .

Втората компонента е така наречения *невъзстановим изход*,

$$\tilde{y}_{t+j} = E_j \xi_{t+j}$$

който не зависи от данните, т.е. на нея не може да се повлияе, чрез оптималния управляващ сигнал. Поради тази причина показателя (2.3) се редуцира до вид, в който минимизацията му зависи само от управлението

$$J(u, t) = \sum_{j=N_1}^{N_2} \left[ \hat{y}_{t+j|k} - r_{t+j} \right]^2 + \lambda \sum_{j=1}^{N_u} \left[ \Delta u_{t+j-1} \right]^2 \quad (2.19)$$

При положение, че

$$N_1 = d + 1, N_2 = d + n \text{ и } \Delta u_{t+j} = 0 \text{ при } N_u \leq i \leq n$$

### 2.3.1.5. Формулиране на предсказания възстановим изход

За да се отделят миналите и бъдещите стойности на управлението, се извежда второто Диофантово уравнение

$$E_j(q^{-1})B(q^{-1}) = G_j(q^{-1})C(q^{-1}) + q^{-j} \Gamma_j(q^{-1}) \quad (2.20)$$

където се съдържа израза за предсказването,

$$\hat{y}_{t+1} = G_j \Delta u_{t+j-1} + \Gamma_j \Delta u_{t-1}^f + F_j y_t^f \quad (2.21)$$

където  $u_t^f$  и  $y_t^f$  са филтрираните сигнали на  $\Delta u_t$  и  $y_t$

$$u_t^f = C^{-1}(q^{-1}) \Delta u_t$$

$$y_t^f = C^{-1}(q^{-1}) y_t$$

еквивалентно,

$$\hat{y}_{t+j} = G_j(q^{-1})\Delta u_{t+j-1} + y_{t+j|t} \quad (2.22)$$

където  $\hat{y}_{t+j|t}$  е предсказвания възстановим изход на  $y_{t+j}$ , като се приеме, че бъдещите нараствания на управлението след време  $t - 1$  ще са равни на нула

$$\hat{y}_{t+j|t} = G_j(q^{-1})\Delta u_{t-1}^f + F_j(q^{-1})y_t^f \quad (2.23)$$

Първият запис на диофантовото уравнение (2.6) е класически подход за определяне на оптималното предсказване. Може да се покаже, че полиномът  $C$  записан в (2.7) определя динамиката на наблюдението. Идеята е, че вместо да бъде оставен независим от идентификационната процедура на реалния процес полиномът  $C$  може да бъде избран отделно, като чрез него се определя динамиката на наблюдението.

Ако полиномите (2.6) и (2.20) се разделят с  $j$ . За израза (2.6) се получава

$$C(A\Delta)^{-1} = E_j + q^{-j}F_j(A\Delta)^{-1} \quad (2.24)$$

оттук заместваме  $E_j$  в (2.20),

$$\begin{aligned} CB(A\Delta)^{-1} &= BE_j + q^{-j}BF_j(A\Delta)^{-1} = G_jC + q^{-j}\Gamma_j + q^{-j}BF_j(A\Delta)^{-1} \\ B(A\Delta)^{-1} &= G_j + q^{-j}\Gamma_jC^{-1} + q^{-j}BF_j(A\Delta C)^{-1} \end{aligned} \quad (2.25)$$

Ддиофантовото уравнение (2.20) може да бъде решено с рекурсивна процедура, т.е. чрез обновяване на изразите, чиито резултати от многото изчисления се запазват върху техните индивидуални и единични изчисления.

Определяме вектора на предсказвания възстановим изход  $\mathbf{f}$

$$\mathbf{f} = [\hat{y}_{t+1|t}, \hat{y}_{t+2|t}, \dots, \hat{y}_{t+N2|t}]^T \quad (2.27)$$



т.е., предсказването на  $\hat{y}_{t+k}$ ,  $k = 1, \dots, N_2$ , дава  $\{u_{s-1}, y_s; s \leq t\}$  допускайки, че  $\{u_{t+k} = 0, k = 0, \dots, N_2 - 1\}$ . Следващото определяне на вектора на бъдещото управление увеличава  $\hat{\mathbf{u}}$  (като разполагаме с  $\Delta u_{t+j} = 0$  за  $j \geq N_u$ )

$$\hat{\mathbf{u}} = [\Delta u_t, \Delta u_{t+1}, \dots, \Delta u_{t+N_u-1}]^T \quad (2.28)$$

определяме вектора на предсказаното управление на изходите на обекта

$$\hat{\mathbf{y}} = [\hat{y}_{t+1}, \hat{y}_{t+2}, \dots, \hat{y}_{t+N_2}]^T \quad (2.29)$$

От уравнението (2.22), предсказаното входно–изходно отношение на обекта може да бъде записана като векторно уравнение

$$\hat{\mathbf{y}} = \mathbf{G}\hat{\mathbf{u}} + \mathbf{f} \quad (2.30)$$

където матрицата  $\mathbf{G}$  съдържа стойностите на дискретната преходна функция  $g_i$  от модела на обекта  $B/\Delta\Delta$ .

$$\mathbf{G} = \begin{pmatrix} g_0 & 0 & \dots & 0 \\ g_1 & g_0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{N_u-1} & g_{N_u-2} & \dots & g_0 \\ \vdots & \vdots & & \vdots \\ g_{N_2-1} & g_{N_2-2} & \dots & g_{N_2-N_u} \end{pmatrix} \quad (2.31)$$

следователно, коефициентите на полинома  $G_j(q^{-1})$  са първите (ненулевите)  $j$  стойности на описаната в (2.25) преходна функция. Размерността на матрицата  $\mathbf{G}$  е  $N_2 \times N_u$ , като сме отчели ограниченията на  $\Delta u_{t+j}$  за  $j \geq N_u$  и приемаме  $N_1$  равно на единица за простота в (2.31). Ефектът от промяната на  $N_1$  е за да се нулират горните редове на  $\mathbf{G}$ . Квадратичната минимизация от (2.3) сега може да се представи така

$$\mathbf{J} = (\hat{\mathbf{y}} - \mathbf{r})^T (\hat{\mathbf{y}} - \mathbf{r}) + \lambda \hat{\mathbf{u}}^T \hat{\mathbf{u}} \quad (2.32)$$

Минимумът на този израз се намира при положение, че няма наложени ограничения върху управляващият сигнал, като се положи градиентът на  $J$  да е равен на нула. Това води до оптималното решение на бъдещето нарастване на вектора на управлението  $\tilde{\mathbf{u}}$

$$\tilde{\mathbf{u}} = (\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T (\mathbf{r} - \mathbf{f}) \quad (2.33)$$

където  $\mathbf{r}$  е се определя от заданието като

$$\mathbf{r} = [r_{t+1}, r_{t+2}, \dots, r_{t+N_2}]^T \quad (2.34)$$

Уравнението (2.32) съдържа бъдещите стойности на управлението за време  $t + N_u - 1$  базирани на информацията налична във време  $t$ . Механизмът използван в затворената система се осъществява само от първия елемент на  $\mathbf{u}$ ,  $\Delta u_t \equiv u(1)$ , след което управлението се изчислява отново за следващата стъпка, като се използват данните налични във време  $t + 1$  определени в  $\mathbf{f}$ . Тази процедура се нарича *отдалечен хоризонт на управление*. Ползата от осъществяването на тази процедура е, че управляващият *gain* изчислен в (2.33) остава фиксиран и само векторите  $\mathbf{f}$  и  $\mathbf{r}$  се обновяват.

### 2.3.2. Формулиране на затворения контур

Формулирането на затворения контур за GPC ще бъде представено в неадаптивния случай с цел да се покаже, как структурните параметри  $N_1$ ,  $N_2$ ,  $N_u$  и  $\lambda$  могат да влияят върху устойчивостта на управлявания обект. Първо ще определим  $\rho_i$ ,  $i = 1, \dots, N_2$ , като коефициенти от първия ред на матрицата (2.32).

$$(\rho_1 \quad \dots \quad \rho_{N_2}) = (1 \quad 0 \quad \dots \quad 0)(\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T \quad (2.35)$$

Лесно е да се види, че първата компонента на векторното уравнение (2.32) може да бъде записана така

$$\sum_{i=1}^{N_2} \rho_i \frac{\Gamma_i}{C} \Delta u_{t-1} - \sum_{i=1}^{N_2} \rho_i \frac{F_i}{C} y_t + \sum_{i=1}^{N_2} \rho_i r_{t+i} \quad (2.36)$$

Това води до уравнението на управлението

$$\left(C + \sum_{i=1}^{N_2} \rho_i \Gamma_i q^{-1}\right) = \left(C \sum_{i=1}^{N_2} \rho_i q^{-N_2+i}\right) r_{t+N_2} - \left(\sum_{i=1}^{N_2} \rho_i F_i\right) y_t \quad (2.37)$$

или с явното определение на полиномите  $R$ ,  $S$  и  $T$ ,

$$R \Delta u_t = C T r_{t+N_2} - S y_t \quad (2.38)$$

Това показва, че GPC е начин за синтезиране на линейно управление посредством обратна връзка, предвид на критерия за оптимизация. Разбира се всеки линеен регулатор каквато и техника за синтез да има, винаги ще притежава структура подобна на (2.38).

Затвореният контур се получава, чрез комбиниране на модела на обекта (2.2) с уравнението на управлението (2.38). Това непосредствено ще даде следния затворен контур,

$$(A \Delta R + B S q^{-1}) y_t = B C T r_{t+N_2-1} + C R \xi_t \quad (2.39)$$

Използвайки определените  $R$ ,  $S$  и  $T$  след няколко изчисления, се вижда, че:

$$\begin{aligned} A \Delta R + B S q^{-1} &= A \Delta C + \sum_{i=1}^{N_2} \rho_i (A \Delta \Gamma_i + B F_i) q^{-1} \\ &= C \left( A \Delta + \sum_{i=1}^{N_2} \rho_i (B - A \Delta G_i) q^{i-1} \right) \\ &= C A_c \end{aligned} \quad (2.40)$$

Оттук уравнение (2.39) може да се запише така:

$$y_t = \frac{BT}{A} r_{t+N_2-1} + \frac{R}{A_c} \xi_t \quad (2.41)$$

както се вижда, полинома  $C$  се анулира в предавателната функция на затворения контур (което е обикновено в случаите на динамичен наблюдател), като устойчивостта и състоянието на затворената система се управляват от корените на полинома  $A_c$  който влияе, чрез различния избор на структурните параметри  $N_1$ ,  $N_2$ ,  $N_u$  и  $\lambda$ . На практика този подход е направен за да оптимизира в реално време стойностите на структурния параметър  $\lambda$  с цел да вземе съответни стойности за корените на  $A_c$ , но това не винаги е възможно. Трябва да се отбележи също, че  $A_c(1) = \sum_{i=1}^{N_2} \rho B(1)$  и

$$(BT)(1) = \sum_{i=1}^{N_2} \rho B(1) \quad \text{за да може статичното усилване на}$$

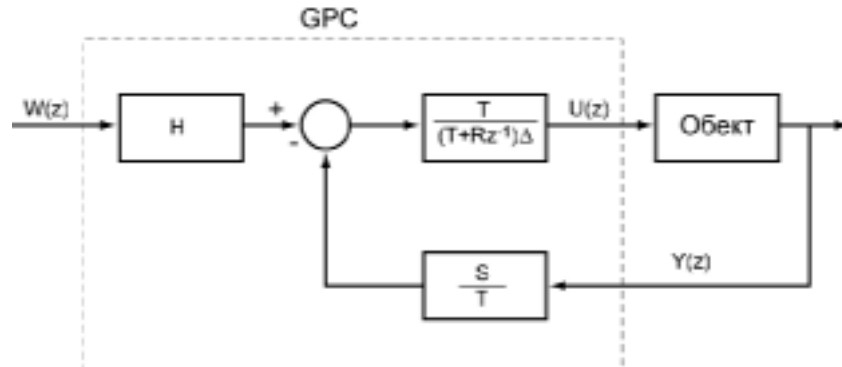
предавателната функция на затворения контур от заданието към изхода винаги да е 1.

Ако се върнем отново към ролята на полинома  $C$ , като наблюдателен полином, ще подчертаем, че ако модел е избран за обект с  $C = 1$  и ако наблюдателния полином  $T$  е избран от потребителя, като се използва вместо  $C$  в Диофантовото уравнение (2.6) и (2.20), тази динамика на наблюдение ще се изгуби в предавателната функция на затворения контур от  $r_t$  към  $y_t$  но не и в шума към изходната предавателна функция. Много повече за ролята на наблюдателния полином  $T$  и неговото използване може да бъде намерено в Mohtadi (1988).

След като сме определили

$$\begin{aligned} H &= \sum_{j=1}^{N_2} (G^T G + \lambda I)^{-1} G^T, \\ R &= (C + \sum_{j=1}^{N_2} \rho_j \Gamma_j q^{-1}), \\ S &= \sum_{j=1}^{N_2} \rho_j F_j \end{aligned} \quad (2.42)$$

можем да определим вида на структурната схема, която ще използваме при осъществяване на управлението



фиг. 3 Принципно блок–диаграма на GPC регулатора

### 2.3.3. Систематизиране на алгоритъма

1. Въз основа на информацията, с която се разполага в  $k$ -тия момент се оценяват параметрите на полиномите  $A$  и  $B$  в модела (2.1). За целта се приема, че  $C \equiv 1$ ;
2. Решава се рекурсивно Диофантовото уравнение (2.6) и се определят полиномите  $E_i$  и  $F_i$  за  $i = 1, 2, \dots, N_2$ , след което се определят и  $G_i = E_i B$ ;
3. От (2.17) се определя се вектора на възстановимия изход  $u_{t+j}$ ;
4. Образува се матрицата  $G$  и векторът  $\mathbf{f}$ ;
5. Решава се системата уравнения (2.32) и се намира векторът  $\tilde{\mathbf{u}}$ ;
6. Прилага се управлението  $u_t = u_{t-1} - \Delta u_t$ , полага се  $k = k + 1$  и алгоритъмът се повтаря в следващия такт от стъпка 1.

## 2.3.4. Избор на структурни параметри

Приложението на този клас регулатори изисква задаването на редица параметри. Това от една страна дава голяма гъвкавост при проектирането, но от друга, изисква добро познаване на неговите свойства. Параметрите, които трябва да се зададат предварително, са свързани с модела на обекта и с минимизирувания функционал.

### 2.3.4.1. Параметри на модела

При използването на параметрични модели е необходимо да се зададат степените  $n$  и  $m$  на полиномите, както и чистото закъснение  $d$ . Обикновено се избира  $n = m$  не по-голямо от 2–3. Ако чистото закъснение е неизвестно или променливо, може да се зададе стандартната стойност  $d = 1$  и да се увеличи степента на числителя в полинома до  $m + d_{\max}$ . Това увеличава значително необходимия изчислителен ресурс при оценяването. Изборът на  $m$ ,  $n$  и  $d$  не е критичен за работата на управляващите алгоритми. Чистото закъснение обикновено се поглъща от полинома  $B$  и, ако се избере достатъчно отдалечен хоризонт на предсказване  $N_2$ , то не оказва съществено влияние. От друга страна, понеже в Диофантовото уравнение не участва полиномът  $B$ , препараметризирането на модела също не е опасно

### 2.3.4.2. Параметри на регулатора

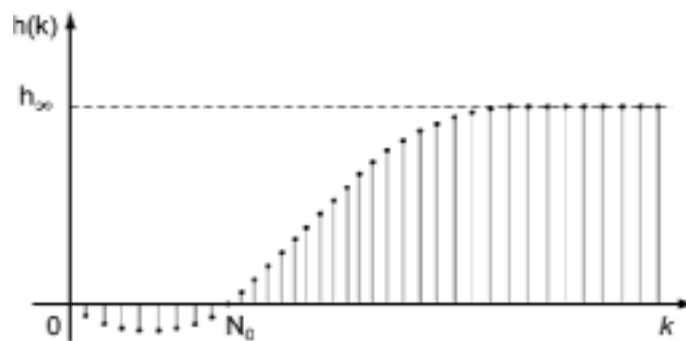
Основните структурни параметри на регулатора са три: минимален  $N_1$  и максимален  $N_2$  хоризонти на предсказване на изхода и хоризонт на управление  $N_u$ . Към тях следва да прибавим параметрите на филтрите  $W_R(q^{-1})$  и  $Q(q^{-1})$ . Обикновено се избира от вида

$$y_r(k+i) = \alpha y_r(k+i-1) + (1-\alpha)r(k+i-1), \quad i = 1, \dots, N_2 \quad (2.43)$$

и параметърът, който се задава е  $\alpha$ , а  $Q$  има вида  $\lambda(1-q^{-1})$  и се избира  $\lambda$ .

**Минималният хоризонт** на предсказване е  $N_1 = d$ . Ако  $d$  е неизвестно се поглъща от полинома  $B$ , то  $N_1 = 1$  по подразбиране.

**Максималният хоризонт на предсказване на изхода** е добре да бъде по-голям, например от порядъка на  $N_2 = T_3/T_0$ . В случаите на програмирано задание, по-голямата стойност на  $N_2$  може да доведе до твърде ранна реакция на изхода, което не винаги е желателно. Във всички случаи той трябва да е по-голям от максимално възможното чисто закъснение  $d_{\max}$ . Най-често се задава  $N_2 > t$ , като се вземе предвид възможната неминималнофазовост на непрекъснатия обект и това налага  $N_2$  да се избере така, че  $N_2 > N_0$  (фиг. 4). В противен случай съществува опасност от подаване на управляващи сигнали, които ще разколебават системата.



фиг. 4 Неминималнофазов обект

**Хоризонтът на управление**  $N_u$  се избира в интервал  $[1, N_2]$ . По-голямата му стойност води, като правило, до по-добро управление, но увеличава в значителна степен броя на изчисленията. В повечето случаи избора на  $N_u$  в порядъка на  $N_u = 1 \div 3$  дава добри резултати. Препоръчителни стойности са  $N_u \geq p + 1$ , където  $p$  е броят на лошо демпфираните или неустойчиви полюси на обекта. По-малките стойности на  $N_u$  водят до малки амплитуди на управляващите сигнали, което в някои случаи е от съществено значение, а освен това осигуряват регуляризиращо действие при  $\lambda = 0$ .

Параметърът  $\alpha$  в еталонния модел (2.1) задава динамиката на следене на заданието и се избира в интервала  $[0, 1]$ . По-големите стойности на  $\alpha$  осигуряват по-мекото управление и обратно, подобна роля играе и параметърът  $\lambda$ , който осъществява компромиса между точността на следене на

желаната траектория и мекотата на управлението. Освен това при наличие на чисто закъснение  $d > 1$  и избор на  $N_1 = 1$  и  $N_2 = N_u$ , ненулевата му стойност осигурява и устойчива работа на системата, тъй като в противен случай матрицата  $G^T G$  може да стане лошо обусловена. Препоръчителни стойности са  $0.01 \leq \lambda \leq 0.1$ .

За всеки конкретен случай, оптималните стойности на структурните параметри на регулатора са строго индивидуални. При липса на информация за обекта по подразбиране могат да се използват следните стойности на параметрите:  $N_1 = 1$ ,  $N_2 = 10$ ,  $N_u = 2$ ,  $\alpha = 0$ ,  $\lambda = 0.1$ .

## 2.4. Програмно осигуряване

### 2.4.1. Изграждане на програмно управление

За да се види действието описания по-горе алгоритъм на базата на аналитично решен пример ще бъде формулирано програмното управление на дискретния GPC регулатор в отворения контур. Всички изчисления се провеждат в средата на MATLAB®.

Приема се, че GPC регулатора управлява обект описан с ARIMAX модела от вида (2.1)

$$y_t = 1.7y_{t-1} - 0.7y_{t-2} + 0.9\Delta u_{t-1} - 0.6\Delta u_{t-2} + \xi_t \quad (2.44)$$

където  $\xi_t$  е с нулева средна стойност на белия шум. След преобразуване на ARIMAX модела в ARMAX се определят полиномите на обекта

$$\begin{aligned} A(q^{-1}) &= 1 - 0.7q^{-1} \\ B(q^{-1}) &= 0.9 - 0.6q^{-1} \\ C(q^{-1}) &= 1 \end{aligned}$$



Засега се приема, че параметрите на регулатора са  $N_1 = 1$ ,  $N_2 = 3$ ,  $N_u = 3$  и  $\lambda = 0.1$ .

За програмно решение на тази задача в средата на MATLAB в нов *m*-файл *grc\_1.m* се записват параметрите на регулатора и полиномите на модела:

---

```
N1=1; N2=10; Nu=2; lambda=0.1; alpha=0.2;

A = [1 -0.7];
B = [0.9 -0.6];
C = 1
```

---

Определя се  $A(q^{-1})\Delta$ , като се има предвид, че  $\Delta = 1 - q^{-1}$ :

---

```
delta = [1 -1];
Ad     = [conv(A,delta) zeros(1,N2-(length(A)+N1))];
```

---

Задават се началните стойности на полиномите  $E_1$  и  $F_1$ , където  $F_1 = q(1 - A\Delta)$ :

---

```
E = 1;
F1 = eye(1,N2)-Ad;
for k=N1:N2-1
    F2(k) = F1(k+1);
end
```

---

Предсказват се стойности на изхода, чрез модел на обекта (2.1) за хоризонт на предсказване  $N_2 = 3$ , при условие, че  $\xi_{t+i} = 0$ ,  $i \geq 1$ . От преобразуваното уравнение се установяват зависимостите:

$$\begin{aligned}\hat{y}_{t+1} &= 1.7y_t - 0.7y_{t-1} + 0.9\Delta u_t - 0.6\Delta u_{t-1} \\ \hat{y}_{t+2} &= 1.7\hat{y}_{t+1} - 0.7y_t + 0.9\Delta u_{t+1} - 0.6\Delta u_t \\ &= 2.19y_t - 1.19y_{t-1} + 0.9\Delta u_{t+1} + 0.93\Delta u_t - 1.02\Delta u_{t-1} \\ \hat{y}_{t+3} &= 1.7\hat{y}_{t+2} - 0.7y_{t+1} + 0.9\Delta u_{t+2} - 0.6\Delta u_{t+1} \\ &= 2.533y_t - 1.533y_{t-1} + 0.9\Delta u_{t+2} + 0.93\Delta u_{t+1} + 0.951\Delta u_t - \\ &\quad - 1.314\Delta u_{t-1}\end{aligned}$$

Решава се първото Диофантово уравнение (2.6) от  $i = 1 \div 3$

$$i = t, \quad \left| \begin{array}{l} 1 = E_t(q^{-1})A(q^{-1})\Delta + q^{-1}F_t(q^{-1}) \end{array} \right.$$

$$i = 1, \quad \left| \begin{array}{l} 1 = 1(1 - 1.7q^{-1} + 0.7q^{-2}) + q^{-1}(1.7 + 0.7q^{-1}) \\ E1 = 1, \\ F1 = 1.7 - 0.7q^{-1} \end{array} \right.$$

$$i = 2, \quad \left| \begin{array}{l} 1 = (1 + 1.7q^{-1})(1 - 1.7q^{-1} + 0.7q^{-2}) + q^{-2}(2.19 - 1.19q^{-1}) \\ E2 = 1 + 1.7q^{-1}, \\ F2 = 2.19 - 1.19q^{-1} \end{array} \right.$$

$$i = 3, \quad \left| \begin{array}{l} 1 = (1 + 1.7q^{-1} + 2.19q^{-2})(1 - 1.7q^{-1} + 0.7q^{-2}) + \\ \quad + q^{-3}(2.533 - 1.533q^{-1}) \\ E3 = 1 + 1.7q^{-1} + 2.19q^{-2}, \\ F3 = 2.533 - 1.533q^{-1} \end{array} \right.$$

Прави се рекурсивна процедура за намиране на неизвестните стойности на  $E_j$  и  $F_j$  (т. 3.1.4):

---

```
for i=N1:N2-1
    E(i+1,i+1) = F2(i,1);
    for j=1:N2
        if j<=i
            E(i+1,j) = E(i,j);
```

```

end
if j<N2-1
    F2(i+1,j) = F2(i,j+1)-E(i+1,i+1)*Ad(j+1);
end
if j==N2-1
    F2(i+1,j) = -E(i+1,i+1)*Ad(j+1);
end
end
end
F2 = F2(N1:N2,N1:length(A));

```

---

Решава се второто диофантово уравнение (2.20), за да се намерят стойностите на тегловната матрица  $G_j(q^{-1})$  и вектора  $\Gamma_j(q^{-1})$ :

$$i = 1, \begin{cases} E_1 B = 0.9 - 0.6q^{-1} \\ G_1 = 0.9, \\ \Gamma_1 = -0.6 \end{cases}$$

$$i = 2, \begin{cases} E_2 B = 0.9 + 0.93q^{-1} - 1.02q^{-2} \\ G_2 = 0.9 - 0.6q^{-1}, \\ \Gamma_2 = -1.02 \end{cases}$$

$$i = 3, \begin{cases} E_3 B = 0.9 + 0.93q^{-1} + 0.951q^{-2} - 1.314q^{-3} \\ G_3 = 0.9 - 0.93q^{-1} - 0.951q^{-2}, \\ \Gamma_3 = -1.314 \end{cases}$$

Отново, чрез рекурсивна процедура се намират неизвестните стойности на  $G_j(q^{-1})$  и  $\Gamma_j(q^{-1})$  за хоризонта на предсказване  $N_2 = 3$ :

```

for i=N1:N2
    EB(i,:) = conv(E(i,:),B);
end

for i=N1:N2-1
    G2(i,1) = EB(i,i);
    for j=N1:N2
        if j<=i
            G2(i+1,1) = EB(i+1,j+1);
            G2(i+1,j+1) = G2(i,j);
        end
    end
end
J2 = diag(EB,1);

```

---

След като матриците  $G_j(q^{-1})$ ,  $F_j(q^{-1})$  и  $\Gamma_j(q^{-1})$  са вече намерени, може да се запише матричният вид на предсказания изход:

$$\hat{y}_{t+1} = \begin{pmatrix} 0.9 & 0 & 0 \\ 0.93 & 0.9 & 0 \\ 0.951 & 0.93 & 0.9 \end{pmatrix} \begin{pmatrix} \Delta u_t \\ \Delta u_{t+1} \\ \Delta u_{t+2} \end{pmatrix} + \begin{pmatrix} 1.7y_t & -0.7y_{t-1} & -0.6\Delta u_{t-1} \\ 2.19y_t & -1.19y_{t-1} & -1.02\Delta u_{t-1} \\ 2.533y_t & -1.533y_{t-1} & -1.314\Delta u_{t-1} \end{pmatrix}$$

Тегловната матрица, вече може да бъде записана:

$$(G^T G + \lambda I_3)^{-1} G^T = \begin{pmatrix} 0.8947 & 0.0929 & 0.0095 \\ -0.8316 & 0.8091 & 0.0929 \\ -0.0766 & -0.8316 & 0.8947 \end{pmatrix}$$

---

```
I = lambda*eye(N2);
Gu = inv(G2'*G2+I)*G2';
```

---

Оттук може да се изведе решението:

$$\Delta u_t = -0.644\Delta u_{t-1} + 1.7483y_t - 0.7513y_{t-1} + 0.8947r_{t+1} + 0.0929r_{t+2} + 0.0095r_{t+3}$$

Изчисляват се полиномите на регулатора за хоризонт на управление  $N_u$

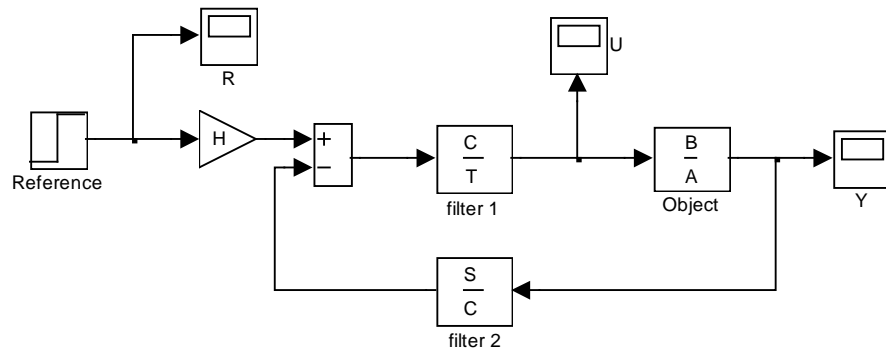
---

```
H = Gu(1, :);
J = H*J2;
F = H*F2;
G = sum(H); Ju = [0 J];

Ru = [0 R];
x3 = max(length(C), length(Ru));
Rd = [C zeros(1, x3-length(C))] + [Ru zeros(1, x3-length(Ru))];
T = conv(Jd, [1 -1]);
```

---

След като разполагаме с необходимите полиноми в средата на SIMULINK® построяваме схема необходима за симулиране на управлението:



фиг. 5 Блок–схема за симулиране на управлението (SIMULINK)

Чрез това аналитично решение на задачата, се вижда начинът за формулиране на програмното управление на GPC регулатора. Естествено, съществуват и други идеи за осъществяването му. Освен представения по-горе алгоритъм, в *приложението* (гл. 2.4.2) е показан още един програмен начин за решаване (гл. 2.4.2.2), при който изчислителната процедура е сравнително по-съкратена, за сметка на обема на програмния код. В “Neural Network Control Toolbox” на MATLAB, също съществува функция *appcon.m* за решаване на GPC регулатора, която е представена от Nørgaard *et al* (2000).

## 2.4.2. Приложение

Тук е представен пълният код на два програмни алгоритъма за решаване на задачата на GPC управлението, както и схемата за симулиране на управлението. Програмното осигуряване е осъществено в средата на MATLAB 5.2.0, със стандартни функции достъпни в “System Control Toolbox”. За рекурсивната процедура необходима за решаване на диофантовото уравнение са използвани операторите за цикъл и условен преход *for* и *if*. При векторните и

матричните операции — функциите *max*, *length*, *zeros*, *ones*, *eye*, *diag*, *sum* и *conv*. За графично изчертаване на резултатите — функцията *plot*.

## АЛГОРИТЪМ 1

Този алгоритъм е разработен въз основа на пример (2.44). Програмата е представена в *gpc\_1.m*, по начина описан в (гл. 2.3.3), за да бъде по-лесно разбрано нейното функциониране.

```
% ИНИЦИАЛИЗИРАНЕ ПАРАМЕТРИТЕ НА РЕГУЛАТОРА И ОБЕКТА

% Параметри на регулатора
N1=1; N2=3; Nu=3; lambda=0.1;

% ARIMAX модел на обекта
A = [1 -0.7];
B = [0.9 -0.6];
C = 1;

% Делта оператор
delta = [1 -1];
Ad = [conv(A,delta) zeros(1,N2-(length(A)+N1))];

% Първите елементи на E1 и F1
E = 1;
F1 = eye(1,N2)-Ad;
for k=N1:N2-1
    F2(k) = F1(k+1);
end

% РЕКУРСИВНО ИЗЧИСЛЯВАНЕ НА ПОЛИНОМИТЕ E, F, G и J N2

% Определяне на полиномите Ej, Fj
for i=N1:N2-1
    E(i+1,i+1) = F2(i,1);
    for j=1:N2
        if j<=i
            E(i+1,j) = E(i,j);
        end
        if j<N2-1
            F2(i+1,j) = F2(i,j+1)-E(i+1,i+1)*Ad(j+1);
        end
        if j==N2-1
            F2(i+1,j) = -E(i+1,i+1)*Ad(j+1);
        end
    end
end
end

F2 = F2(N1:N2,N1:length(A));

% Определяне на полинома EiB
for i=N1:N2
    EB(i,:) = conv(E(i,:),B);
end
```

```

% Тепловна матрица G
for i=N1:N2-1
    G2(i,1) = EB(i,i);
    for j=N1:N2
        if j<=i
            G2(i+1,1) = EB(i+1,j+1);
            G2(i+1,j+1) = G2(i,j);
        end
    end
end

% Вектор Г - Г=ΔUt-1
J2 = diag(EB,1);

% ИЗЧИСЛЯВАНЕ НА ПОЛИНОМИТЕ НА РЕГУЛАТОРА ЗА Nu

% Матрица на усилване
I = lambda*eye(N2);
Gu = inv(G2'*G2+I)*G2';

% Параметри на регулатора
G = Gu(1,:);
R = G*J2;
S = G*F2;
H = sum(G);

% Изчисляване на полинома T
Ru = [0 R];
x3 = max(length(C),length(Ru));
Rd = [C zeros(1,x3-length(C))] + [Ru zeros(1,x3-
    ...length(Ru))];
T = conv(Rd,[1 -1]);

```

---

## АЛГОРИТЪМ 2

Освен изведения по-горе програмен алгоритъм за решаване на GPC регулатора в отворения контур на управление могат да бъдат осъществени и други начини за представяне на програмното управление. За да се направи сравнение е показан *алгоритъм 2*, който е потвърждение за еднозначност на резултатите при изчисляване на GPC регулатора. Той на практика дава абсолютно идентични резултати както и *алгоритъм 1*, макар и да използва друга технология за решаване на регулатора.

Функциите, които са показани по-надолу са взети от интернет мрежата на Испанския университет “Мигел Ернандес”, който се намира на адрес <http://lorca.umh.es/isa/es/asignaturas/cas/>. Всички величини

и функции са преработени, според изискванията на конкретната задача.

В този алгоритъм рекурсивната процедура е разделена на отделни функции за изчисляване на полиномите необходими за Диофантовото уравнение. Като начало, параметрите на модела и регулатора се записват в

*param.m*,

---

```
N1=1; N2=3; Nu=3; lambda=0.1;
param = [N1 N2 Nu lambda];

A = [1 -0.7];
B = [0.9 -0.6];
C = 1;

[G, J, F, T]=c_gpc(A, B, C, param)
```

---

след което се изпълнява основната функция

*gpc\_2.m*

---

```
function [H, R, S, T]=c_gpc(A, B, C, param);

N1      = param(1);
N2      = param(2);
Nu      = param(3);
lambda  = param(4);

lambda  = lambda*diag(ones(1, Nu));

% Натрупва се числителя на обекта, така че полиномът B
% да съдържа всички закъснения с изключение на своето

if all(B)
    delay=0;
else
    delay=min(find(B))-1;
end

if delay>=1
    B0=B(2:length(B));
else
    B0=B;
end

b0 = B(1);
c0 = C(1);

% ИНИЦИАЛИЗИРАНЕ НА G, Г и F НЕОБХОДИМИ ЗА РЕКУРСИЯТА
```



```

G1 = b0;

% Ред на Ji, n=max(ord(C)-1,ord(B)-1)

J1 = max(length(C),length(B));

J1 = [c0.*B zeros(1,J1-length(B))]-[b0.*C zeros(1,J1-
...length(C))];
J1 = J1(2:length(J1)); % умножение по z
if length(J1)==0
    J1 = 0;
end

% Ред на Fi, n=max(ord(A),ord(C)-1)

Ad = conv([1 -1],A);
F1 = max(length(Ad),length(C));

F1 = [C zeros(1,F1-length(C))]-[c0.*Ad zeros(1,F1-
...length(Ad))];
F1 = F1(2:length(F1)); % умножение по z
if length(F1)==0
    F1 = 0;
end

% ЗАПЪЛВАНЕ С НУЛИ МАТРИЦИТЕ СЪДЪРЖАЩИ ПОЛИНОМИТЕ G,Г, F

G0 = zeros(N2,Nu);
G0(1,1) = G1;

J0 = zeros(N2,length(J1));
J0(1,:) = J1;

F0 = zeros(N2,length(F1));
F0(1,:) = F1;

% РЕКУРСИВНО ИЗЧИСЛЯВАНЕ НА ПОЛИНОМИТЕ G, Г и F

for i=2:N2,
    gi = f_gi(c0,F0(i-1,1),b0,J0(i-1,1));
    G0(i,:) = f_G(G0(i-1,:),gi,Nu);
    F0(i,:) = f_F(F0(i-1,:),Ad);
    J0(i,:) = f_J(J0(i-1,:),B,F0(i-1,:),gi,C);
end

% ОБРАЗУВАНЕ НА МАТРИЦИТЕ G, Г И F ЗА ХОРИЗОНТ N2

G2 = G0(N1:N2,:);
J2 = J0(N1:N2,:);
F2 = F0(N1:N2,:);

% ИЗЧИСЛЯВАНЕ НА ПОЛИНОМИТЕ ЗА ХОРИЗОНТ Nu

H = inv(G2'*G2+lambda)*G2'; % gain matrix
J = H(1,:)*J2; % polinomi na "Gama"
F = H(1,:)*F2; % polinomi na "F"
G = sum(H(1,:)); % gain

% ИЗЧИСЛЯВАНЕ НА ПОЛИНОМА T

Ju = [0 J];
x3 = max(length(C),length(Ju));
Jd = [C zeros(1,x3-length(C))]+[Ju zeros(1,x3-
...length(Ju))];
T = conv(Jd,[1 -1]);

```

Тук са показани функциите за рекурсивно изчисляване на полиномите  $G_j(q^{-1})$ ,  $\Gamma_j(q^{-1})$  и  $F_j(q^{-1})$

### *f\_gi.m*

---

```
% Рекурсивно изчисляване на полиномите G(i+1) и Г(i+1)
function [g_i] = f_gi(c0,f0,b0,j0)
g_i = (f0*b0+j0)/c0;
```

---

### *f\_G.m*

---

```
% Рекурсивно изчисляване на полинома G'(i+1)
function [Gp] = f_G(Gi,gi,Nu)
Gp = [gi Gi(1:Nu-1)];
```

---

### *f\_F.m*

---

```
% Рекурсивно изчисляване на полинома F(i+1)
function [Fp] = f_F(Fi,Ad)
x1 = Fi(1).*Ad;
x3 = max(length(Fi),length(x1));
Fp = [Fi zeros(1,x3-length(Fi))]-[x1 zeros(1,x3-
...length(x1))];
Fp = Fp(2:length(Fp));
```

---

### *f\_J.m*

---

```
% Рекурсивно изчисляване на полинома гама(i+1)
function [Jp] = f_J(Ji,B,Fi,gi,C)
x1 = Fi(1).*B;
x2 = gi.*C;
x3 = max(length(Ji),max(length(x1),length(x2)));
Jp = [Ji zeros(1,x3-length(Ji))]+[x1 zeros(1,x3-
...length(x1))]-[x2-zeros(1,x3-length(x2))];
Jp = Jp(2:length(Jp));
if length(Jp)==0
```

```
end      Jp = 0;
```

---

### 2.4.2.1. Схема за симулиране на управлението

За построяване на схемата за управление в затворения контур (фиг. 5) е използвана графичната среда SIMULINK на MATLAB

В нова схема “model”, от библиотеката “library” на SIMULINK се избират и поставят блоковете необходими за изграждане на блок–схемата за управление:

---

#### ПАРАМЕТРИ НА БЛОКОВЕТЕ

---

В менюто параметри “Parameters” на отделните блокове се записват стойностите, както следва:

- Задание (reference) — От “Sources” се избира блок “Step” с параметри *Step time* = 0, *Initial value* = 0, *Final value* = 1;
- Усилвател (gain) — От “Linear” се избира блок “Gain” с параметър *Gain* = H;
- Суматор (sum) — От “Linear” се избира блок “Sum” с параметри *List of signs* = + -;
- Наблюдател (filter 1) — От “Discrete” се избира блок “Discrete Filter” с параметри *Numerator* = C, *Denominator* = T;
- Предсказан изход (filter 2) — От “Discrete” се избира блок “Discrete Filter” с параметри *Numerator* = F, *Denominator* = C;
- Модел на обекта (object) — От “Discrete” се избира блок “Discrete Transfer Fcn” с параметри *Numerator* = B, *Denominator* = A.

За визуализиране на резултатите може да бъде използван блокът “scope” или да се изчертаят с командата PLOT в средата на MATLAB по следния начин

---

```
figure(1); clf;  
t=1:length(sy);  
plot(t,sr(:,2),'b',t,sy(:,2),'g',t,su(:,2),'r'),grid on
```

---

## Глава 3

### Изследване на ефективността на GPC регулатора

#### 3.1. Цел на изследването

За проверка на ефективността на разработеното програмно осигуряване (гл. 2.4), ще бъде демонстрирано действието на предсказващия регулатор посредством функцията *gpc\_2* от алгоритъм 2 (гл. 2.4.2.1) и схемата за управление (гл. 2.4.2.3).

Изследването на дискретния *обобщен предсказващ регулатор* (GPC) се прави с цел, за да се покаже как се справя той с управлението при промяна на параметрите — хоризонти на предсказване и управление  $N_1$ ,  $N_2$  и  $N_u$  и тегловния коефициент  $\lambda$ . Използван е модел на обекта описан в (2.44).

#### 3.2. Осъществяване на изследването

##### 3.2.1. Параметри на симулиране

За осъществяване на изчисленията се използват полиномите на преработения ARIMAX в ARMAX модел на обекта (2.44), след което се задават коефициентите на хоризонтите на предсказване и управление  $N_1$ ,  $N_2$  и  $N_u$ , тегловният коефициент  $\lambda$  и ако е необходимо (в този случай не е) се включва филтър на заданието  $\alpha$ .

В работната среда на MATLAB се записват параметрите на модела и регулатора:

---

```
A = [1 -0.7];
B = [0.9 -0.6];
C = 1;

N1=1; N2=10; Nu=2; lambda=0.1;
```

---

Посредством *алгоритъм 1* (2.4.2), се изпълнява програмата (*m*-файла) *grc\_1* за да се стартира изчисляването на полиномите  $F_2$ ,  $G_2$  и  $J_2$  за хоризонт на предсказване  $N_2$ , след което се пресмятат полиномите на затворения контур  $H$ ,  $R$ ,  $S$  и  $T$  за хоризонт на управление  $N_u$ .

В това изследване, заданието  $r$  е единично стъпално въздействие “step” с нулева начална стойност. Схемата за симулиране на управлението (фиг. 5) съдържа блоковете необходими за изчислението (гл. 2.4.2.1).

След като бъде изпълнено симулирането, в блоковете  $R$ ,  $U$  и  $Y$  (блокове за визуализация “scope”) се записват променливите  $sr$ ,  $su$  и  $sy$ , които са достъпни в работната среда на MATLAB. От там с командата PLOT тези променливи се визуализират за период  $t$  равен на дължината на вектора  $sy$ :

---

```
figure(1); clf;
t=1:length(sy);
plot(t, sr(:,2), '-.', t, sy(:,2), 'g', t, su(:,2), 'm'); grid on
```

---

**Забележка:** Характеристиките на сигналите са обозначени в цветовете, както следва — задание  $r$  (син цвят), управление  $u$  (червен цвят), изход на обекта  $y$  (зелен цвят).

## 3.2.2. Резултати от изследванията

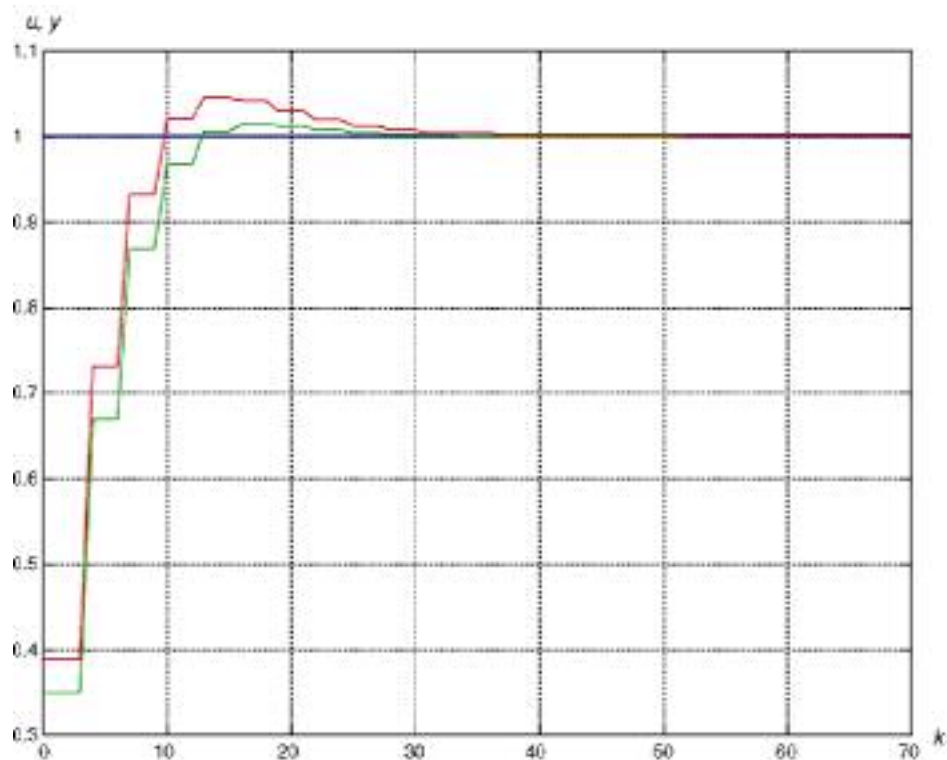
### 3.2.2.1. Обща схема за управление

#### ЕКСПЕРИМЕНТ 1

*Постановка на експеримента:* Използват се стойностите на параметрите от аналитично решената задача (гл. 2.4.1).

Параметри:  $N1=1$ ;  $N2=3$ ;  $Nu=3$ ;  $\lambda=0.1$

Резултати:  $H = 0.997$   
 $R = -0.644$   
 $S = 1.7483 \quad -0.7513$   
 $T = 1.0000 \quad -1.6440 \quad 0.644$



*Общи наблюдения:* Предсказващият регулатор се справя сравнително добре с управлението на обекта. Въпреки скока в началото, той се установява достатъчно бързо.

**ЕКСПЕРИМЕНТ 2**

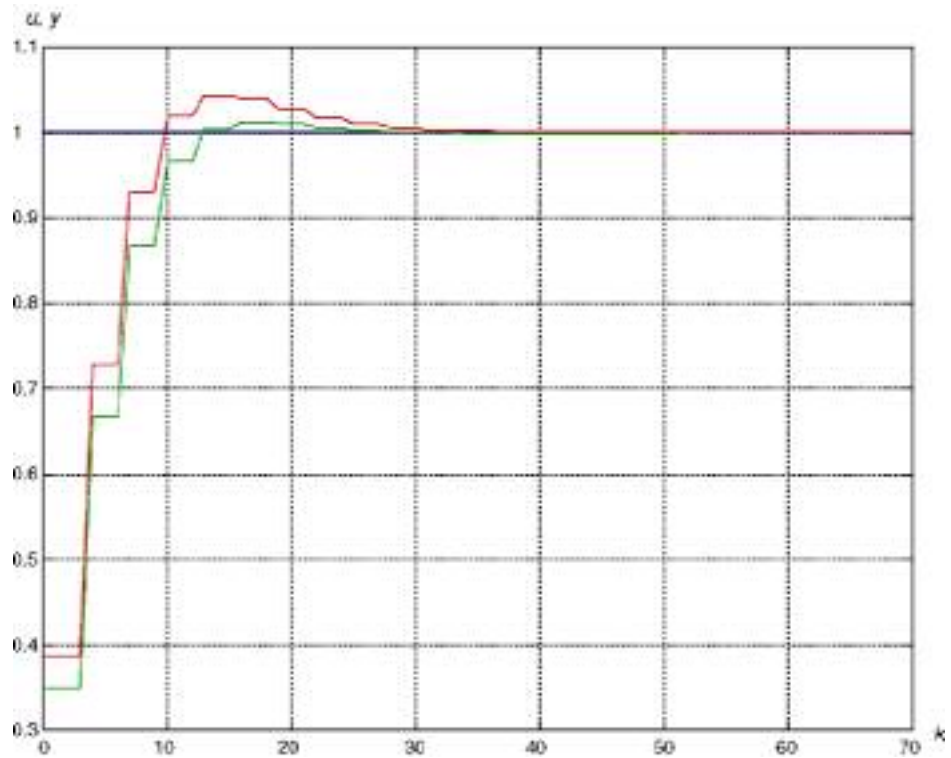
*Постановка на експеримента:* Параметрите се подбират в случай, че липсва достатъчно информация за обекта (гл. 2.3.4.2).

---

Параметри:  $N_1=1$ ;  $N_2=2$ ;  $N_u=10$ ;  $\lambda=0.1$

Резултати:  $H = 0.9944$   
 $R = -0.6832$   
 $S = 1.7915 \quad -0.7971$   
 $T = 1.0000 \quad -1.6832 \quad 0.6832$

---



*Общи наблюдения:* Характеристиките почти не се различават с тези от *експеримент 1*. Вижда се, че увеличаването на крайния хоризонт на предсказване  $N_2$  с няколко степени не влияе особено върху управлението и предсказания изход.

**ЕКСПЕРИМЕНТ 3**

*Постановка на експеримента:* Както се вижда в *експеримент 1* и *2*, увеличаването на крайната стойност на



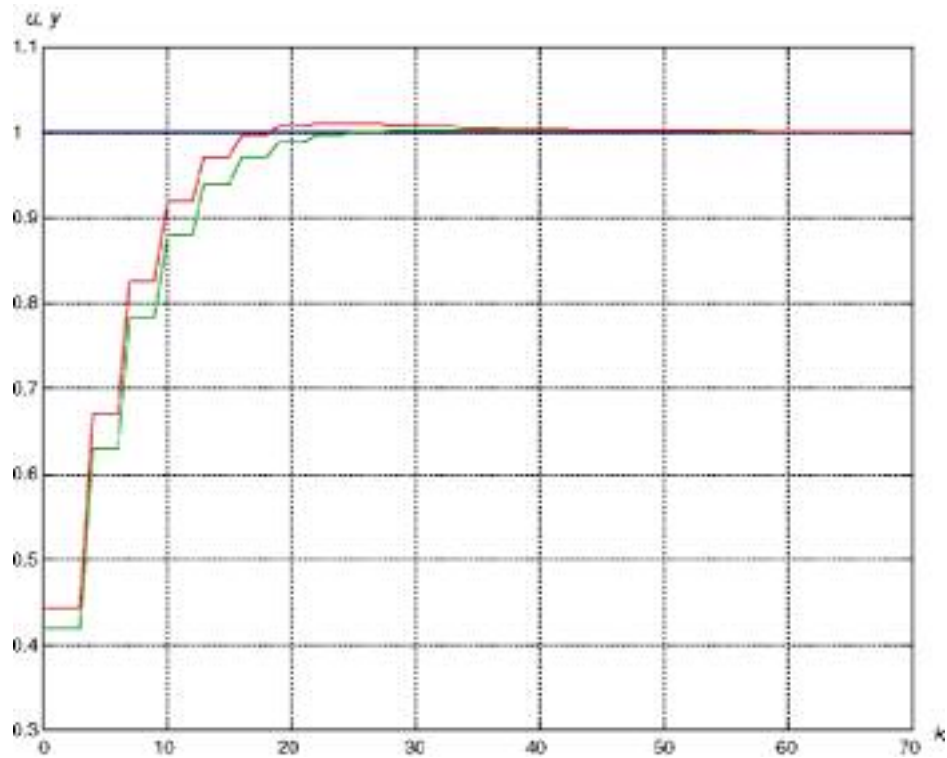
хоризонта на предсказване  $N_2$ , не оказва съществено влияние. В този експеримент ще увеличим тегловния коефициент  $\lambda$  до неговата крайна допустима стойност 1, като оставим останалите параметри същите, както в предния пример.

---

Параметри:  $N_1=1$ ;  $N_2=10$ ;  $N_u=2$ ;  $\lambda=1$

Резултати:  $H = 0.6749$   
 $R = -0.7611$   
 $S = 1.5629 \quad -0.8880$   
 $T = 1.0000 \quad -1.7611 \quad 0.7611$

---

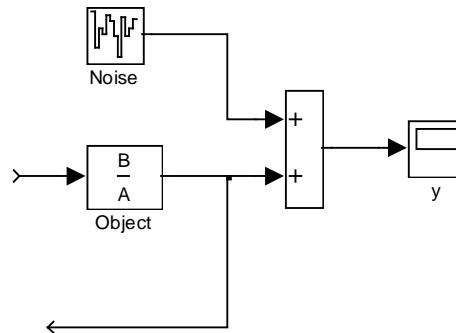


*Общи наблюдения:* Скокът в началото е съвсем малък, при което характеристиките са по-плавни. Оказва се, че параметърът  $\lambda$  играе съществена роля за изрязване на нежеланите амплитуди в началото на процеса, за сметка на по-бавното му установяване. Както се вижда от *експеримент 2*, системата се установява приблизително в границите между  $t = 32-36$ , докато в този случай границите са между  $t = 36-38$ .



### 3.2.2.2. Схема с добавен бял гаусов шум на изхода на обекта

За да се оцени поведението на системата при наличието на бял гаусов шум, ще бъде добавено ново звено “Band-Limited White Noise” към изхода на обекта, с параметри Noise Power = 0.00001. Ето как изглежда схемата:



фиг. 6 Схема с добавен бял гаусов шум на изхода на обекта

## ЕКСПЕРИМЕНТ 5

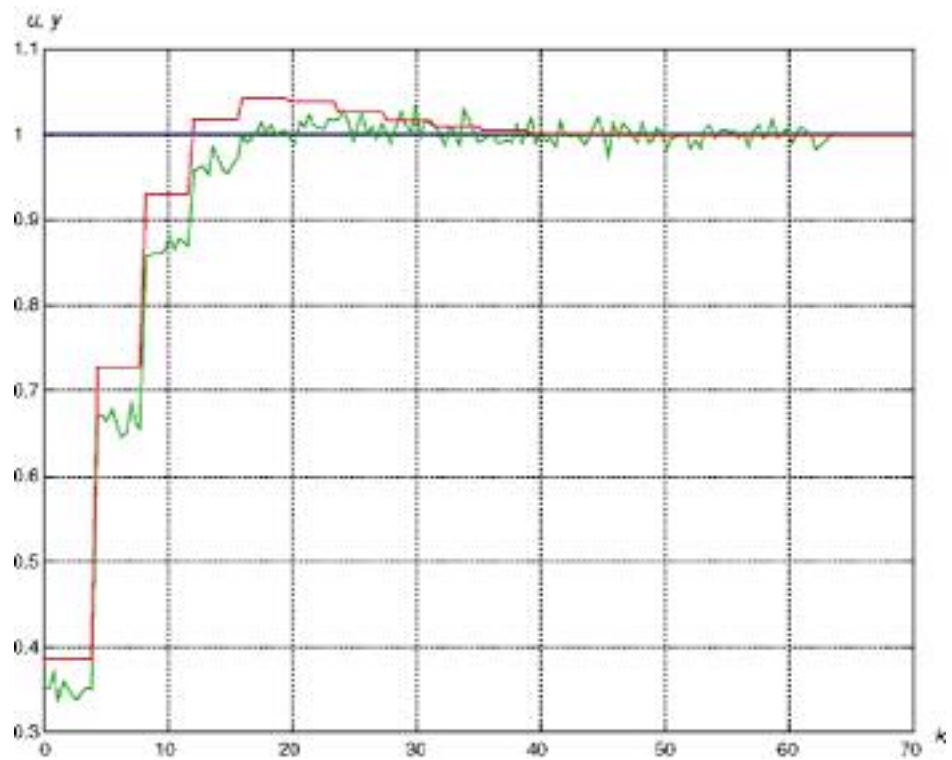
*Постановка на експеримента:* Параметрите са същите, като в експеримент 1, както са формулирани в (гл. 2.4.1).

---

Параметри: N1=1; N2=3; Nu=3; lambda=0.1

Резултати: H = 0.997  
 R = -0.644  
 S = 1.7483 -0.7513  
 T = 1 -1.644 0.644

---



*Общи наблюдения:* Регулятора се справя без проблеми с установяването на управляващия сигнал към траекторията на заданието. Характеристиката на управлението се доближава максимално към тази от *експеримент 1*.

#### **ЕКСПЕРИМЕНТ 6**

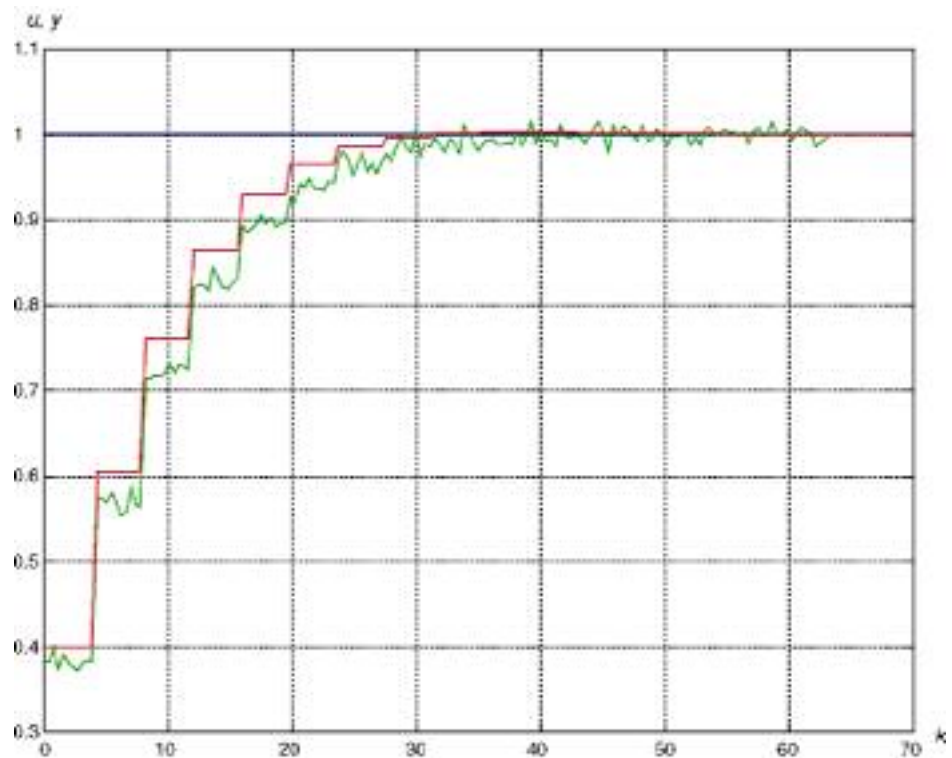
*Постановка на експеримента:* Същите параметри както в *експеримент 4*.

---

Параметри:  $N1=2$ ;  $N2=100$ ;  $Nu=5$ ;  $\lambda=0.2$

Резултати:  $H = 0.4560$   
 $R = -0.5050$   
 $S = 1.0452 \quad -0.5892$   
 $T = 1.0000 \quad -1.5050 \quad 0.5050$

---



*Общи наблюдения:* Резултатите са почти същите както в схемата без шум. Характеристиката на управлението остава незасегната от случайния процес.

### 3.3. Обобщение на резултатите

Като се направи сравнение на характеристиките при различните коефициенти на параметрите на регулатора, се вижда, че в *експеримент 4* се получават най-плавни характеристики. Това се дължи, както на високата начална стойност хоризонта на предсказване  $N_1$ , така и на тегловния коефициент  $\lambda$ , който колкото е по-близо до единица, толкова характеристиката на системата е по-гладка, но за сметка на това схожда по-бавно. При изменение на хоризонта на управление  $N_u$  почти не се наблюдава разлика.

В схемата с добавен шум (фиг. 6), регулатора се справя с изчистването на шума от управляващия сигнал, което до голяма степен се дължи от използването на ARIMAX модела, който позволява да се отработват стъпаловидни въздействия с достатъчно малка честота.

## Глава 4

### Предложения за лабораторни упражнения с използване на GPC регулатора

#### 4.1. Цели

В развитието на темата се стигна до извода, че действието на GPC се осъществява на основата на няколко структурни параметри — хоризонти на предсказване и управление и балансиращи параметри (тегловен коефициент, филтър на заданието). Тези параметри обуславят поведението на регулатора.

С осъществяването на това упражнение трябва бъдат разрешени следните въпроси:

- **Структура:** Да разберем как влияе всеки един от тези параметри върху регулатора и какъв е ефекта му върху управлението на системата.
- **Симулация:** Да успеем да докажем, че във всеки един момент регулатора се държи адекватно, дори при процеси, които възникват в реалния обект (разминаване между модел и реален обект).

#### 4.2. Осъществяване на упражнението

Упражнението се осъществява на базата на примера представен в (гл. 2.4.1). За решаване на GPC регулатора се използва *алгоритъм 2*, тъй като неговите качества са изпитани и в други университети, а описание му във вид на функции предоставя по-добра възможност за съвместимост в учебната компютърна мрежа. Програмното описание на този алгоритъм е преработено, така че да съответства напълно със структурата на разглежданото управление.

За да се изчисли GPC регулатора, на студента се предоставя следната функция в MATLAB:

*gpc\_2.m*

---

```
[H,R,S,T] = gpc_2(A,B,C,param)
```

---

като:

- $V$  е вектор, който съдържа коефициентите на числителя на обекта (включително структурното забавяне на обекта);
- $A$  е вектор, който съдържа коефициентите на знаменателя на обекта;
- $param$  е вектор, който съдържа:

---

```
param(1) = N1
param(2) = N2
param(3) = Nu
param(4) =  $\lambda$ 
```

---

Стойностите, които връща функцията са:

- $H$  е скалар, който съдържа коефициентите на полинома в нарастващите стойности на  $q$ ;
- $R$  е вектор, който съдържа коефициентите на полинома  $R(q^{-1})$  в нарастващите стойности на  $q^{-1}$ ;
- $S$  е вектор, който съдържа коефициентите на полинома  $S(q^{-1})$  в нарастващите стойности на  $q^{-1}$ ;
- $T$  е вектор, който съдържа коефициентите на  $(C(q^{-1}) + R(q^{-1})\Delta)$ .

### 4.3. Пример

Зададен е модел на обекта:

$$y_t = 1.7y_{t-1} - 0.7y_{t-2} + 0.9\Delta u_{t-1} - 0.6\Delta u_{t-2} + \xi_t$$

Да се изчисли GPC регулатора за хоризонт на предсказване  $N_1 = 1$  до  $N_2 = 40$  и хоризонт на управление  $N_u = 5$ . Тегловният коефициентите нека бъде  $\lambda = 0.2$ .

В MATLAB записваме:

---

```
B = [0.9 -0.6];  
A = [1 -0.7];  
C = 1;  
param = [1 40 5 0.2];
```

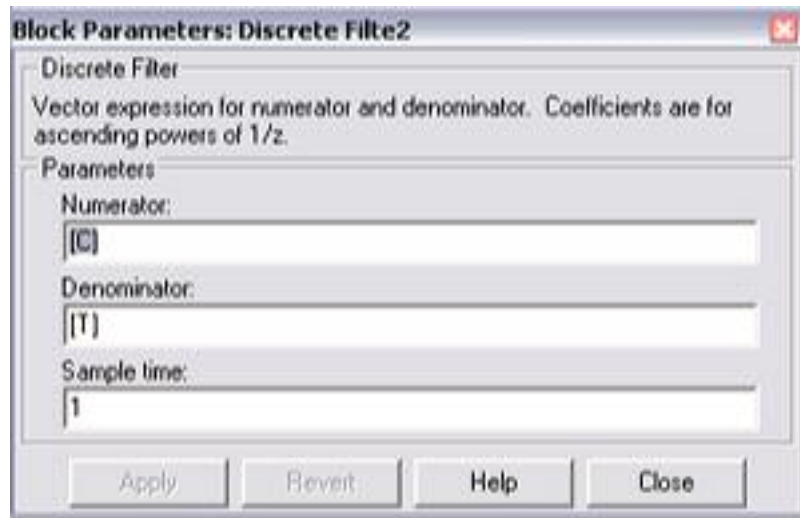
---

### 4.3.1. Симулация

За да се симулира контура на управлението, трябва да се построи блок схема на регулатора (фиг. 3), към която ще се прибави останалата част от системата (фиг. 5).

За да се попълни всеки един от параметрите, които се изискват в различните блокове на схемата – числител, знаменател, и т.н. могат да се използват имената на променливите, които са определени в работната среда на MATLAB. Така променливите, които съдържат изходните полиноми на функцията *gpc\_2* могат да бъдат описани в съответните прозорци на блоковете (фиг. 7).





фиг. 7 Свойства на SIMULINK блока “Discrete Filter”

### 4.3.2. Резултати

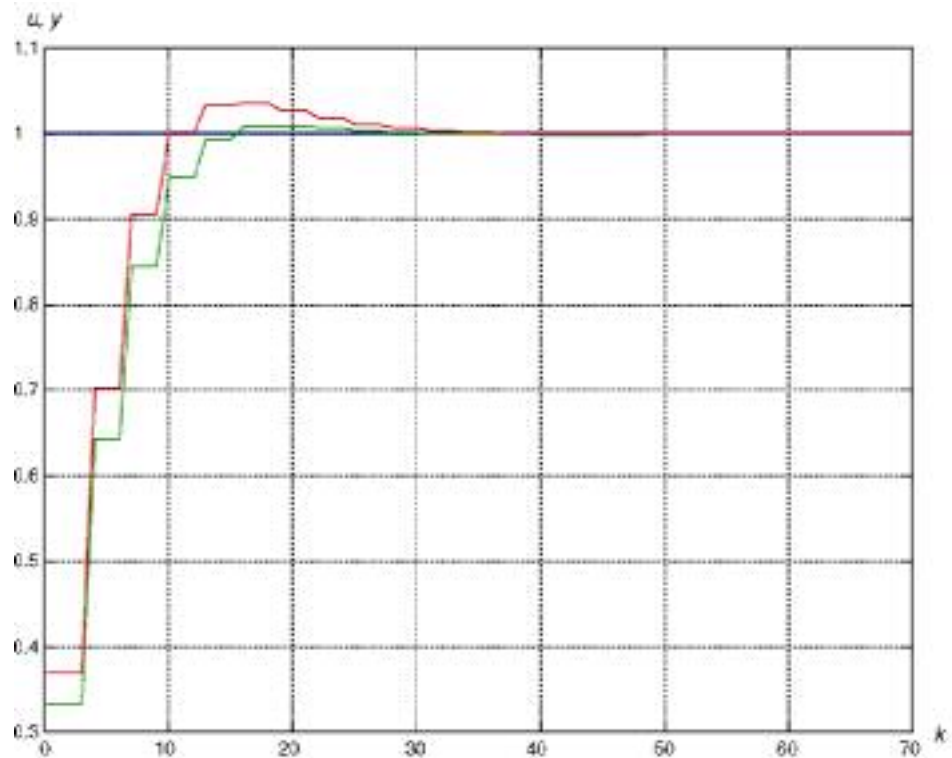
След решаване на примера, резултатите трябва да съответстват на по-долу приведените:

*Резултати:*

---

H = 0.9183  
R = -0.6213  
S = 1.6432 -0.7248  
T = 1.0000 -1.6213 0.6213

---



задание ( $r$ ) — син цвят,  
управление ( $u$ ) — червен цвят,  
предсказан изход ( $\hat{y}$ ) — зелен цвят

## 4.4. Предложения за упражнения

Целта на тези упражнения е да даде на студентите по-широка и обективна представа за поведението на GPC регулатора при различни модели на обекта, а също и какво влияние оказват неговите параметри при промяна на всеки един от тях. Въз основа на наученото от “Теория на управлението” и другите специални предмети, студентите могат да затвърдят познанията си на практика.

Четирите упражнения са развити от проф. Puerto R. в университета “Мигел Ернандес”, Испания, където се преподават. Накрая на всяка задача има по няколко въпроса, целящи да установят до каква степен е усвоен материала. Умишлено са подбрани упражнения, които са доказали своята ефективност в лабораторните часове третиращи изучаването на дискретни предсказващи регулатори.

Тези упражнения ще бъдат само представени, без да се описва начина за тяхното решаване. Целта на тази дипломна работа е да покаже начина за разработване на програмно осигуряване на GPC регулатора, а не последствията от приложението му. Това е тема, която може да бъде осъществена именно в часовете по “Цифрови системи за управление” в ТУ – София.

### УПРАЖНЕНИЕ 1

Имаме модел на обекта дефиниран, чрез

$$G(q) = \frac{0.6(q + 2)}{(q + 0.507)(q + 0.968)}$$

Да се дискретизира тази предавателна функция за период на дискретизация 0.5 s. Да се изчисли и симулира GPC за всеки един от следните случаи:

Пример	N1	N2	Nu	$\lambda$
1	1	40	1	0
2	1	40	3	0.1
3	1	40	1	1
4	1	20	1	1500
5	1	20	1	0.6

#### ВЪПРОС

От гледна точка на предизвиканото действие на управление и на изхода на системата:

- Какъв е ефекта от нарастване на хоризонта на управление  $N_u$ ?
- Какъв би бил ефекта при промяна на балансиращите параметри?

### УПРАЖНЕНИЕ 2

При модел дефиниран от

$$G(q) = \frac{7.3e^{-0.08q}}{0.0361q^2 + 0.66q + 9}$$

да се дискретизира предавателната функция с период на дискретизация 0.001 s. Да се изчисли и симулира GPC регулатор с модел на обекта със следните структурни параметри:

$C(q^{-1})$	$N_1$	$N_2$	$N_d$	$\lambda$
1	1	10	1	0

Да предположим, че реалния обект не е идентифициран правилно, чрез предавателната функция  $G(q)$  и че реалното поведение на системата се определя от следната предавателна функция

$$G^*(q) = \frac{7.3e^{-0.08q}}{0.0361q^2 + 0.55q + 9.44}$$

Сега нека симулираме затворения контур за предварително изчисления регулатор, но с реалния обект  $G^*(q)$  вместо  $G(q)$ . За да включим продължителното закъснение използвайте блока “transport delay”. Забележете как се отразяват на управлението съществените разлики, които се появяват, когато вместо да управляваме модела на обекта изчислен от регулатора, управляваме реалния обект.

Да се структурира и симулира GPC за  $G(q)$  с предните структурни параметри, но като изберем  $C(q^{-1})$  по начина

$$C(q^{-1}) = 1 - 0.9q^{-1}$$

Да се наблюдава поведението на изхода на системата.

ВЪПРОС

Предвид резултатите:

- Каква е основната цел на полинома  $C(q^{-1})$ ?

**УПРАЖНЕНИЕ 3**

Като предположим, че  $G(q)$  моделира перфектно обекта и като имаме бял гаусов шум на изхода, както е показано на фиг. 6, да се сравнят резултатите при симулация на предните два регулатора (за различни стойности на  $C(q^{-1})$ ) и да се наблюдава как влияе смущението върху изхода.

**УПРАЖНЕНИЕ 4**

В този случай имаме синусоидално смущение с приблизителна честота 100 rad в секунда и с амплитуда 0.25. Този вид смущения са много често срещани и могат да се получат на изхода на обекта. Например, в една система за управление на височината на течност в резервоар, двигател който се върти с постоянна скорост и който се намира достатъчно близо до обекта, би предизвикал електромагнитно поле, което влияе върху изхода на системата.

Да се симулира контур на управление с предните два регулатора (с двете стойности на  $C(q^{-1})$ ) и да се наблюдава поведението на изхода.

А сега нека структурираме и симулираме GPC със следните параметри

$C(q^{-1})$	$N_1$	$N_2$	$N_d$	$\lambda$
$(1-0.99q^{-1}) (1-0.95q^{-1})$	1	10	1	0

**ВЪПРОС**

- Каква друга основна цел има полинома  $C(q^{-1})$ ?
- Какво би се случило ако структурния полином  $C(q^{-1})$  се фиксира на  $1 - 1.1q^{-1}$ ? Обосновайте числово отговора.

*Забележка:* Тези лабораторни упражнения се водят в курса “Съвременни системи за управление” в университета “Мигел Ернандес”, Испания от проф. Puerto R.

## Глава 5

### Заклучение

#### 5.1. Резюме

Същинската цел на тази дипломна работа е да покаже начина на програмно осигуряване на GPC регулатора. За да бъде по-ясно разбрано функционирането му е събран и изложен материал от различни научни статии, дисертации, книги, включително и от материали публикувани в глобалната мрежа (Internet). Тук ще представя различните аспекти на този труд.

В началото е направен общ преглед на действието на GPC регулатора, разгледан е накратко принципа на управление, показана е необходимостта от разработването на такъв тип регулатори и е направено съпоставяне с предшестващите го алгоритми, от които произлиза той. Също така е представен обзор на различните методи за предсказващо управление и са разгледани разширените модификации на GPC регулатора, които имат за цел, например, да гарантират устойчивост, да работят с твърди ограничения и т.н.

Следва математическа формулировка на алгоритъма и начин за неговото програмно осигуряване. Приложената теоретична част включва анализ на GPC в отворения и затворения контур на управление. В отделни точки е описано решаването на алгоритъма, като се тръгне от избор на модел на обекта, мине се през рекурсивната процедура (решаване на Диофантовото уравнение) и се стигне до формулиране на управлението. Разгледано е как става избирането на параметрите на регулатора и по какъв начин влияе всеки един от тях върху работата на системата.

Най-важната част от тази дипломна работа е изграждането на програмното осигуряване (гл. 2.4) или по-точно на алгоритъма за изчисляване на GPC, осъществен на базата на дискретен модел на обекта (гл. 2.3.3). В края на главата е разгледан втори начин за програмно решаване на GPC

регулатора. Описана е и схемата за осъществяване на управлението (фиг. 5).

По-нататък е представена изчислителната част. За да бъде показана ефективността на програмния алгоритъм на GPC управлението, са проведени експерименти с различни параметри на регулатора. Направен е кратък анализ на всеки експеримент, както и обща оценка за влиянието на параметрите.

Тъй като предсказващите регулатори са тема, която се изучава в курса “Цифрови системи за управление”, са предложени и лабораторни упражнения. Те дават възможност на студентите да се запознаят по-отблизо с начина на функциониране на GPC регулатора, който е най-популярен в този клас методи за управление. Познаването на GPC предоставя възможност за развиване на нови алгоритми, базирани на него.

Предложеният пример е във формата на функция, която се извиква от работната среда на MATLAB. Студентът трябва да зададе полиномите на модела на обекта (2.44) и параметрите на регулатора. За да симулира управлението на обекта реализира блок–схемата (фиг. 5). Резултатите трябва да отговарят на тези от гл. 4.3.2.

За да се насърчи по-доброто усвояване на темата, са предложени няколко упражнения (Puerto R., 2002), които трябва да бъдат развити от студента. Тези лабораторни упражнения са изпитани в лабораторните часове в специалността “Съвременни системи за управление”, в Университета “Мигел Ернандес”, Испания, от където са заимствани. В края на всяко упражнение има конкретен въпрос.

## 5.2. Литература

1. Велев К., Самонастройващи се регулатори. Адаптивни системи, 168–195.
2. Гарипов Е., (2002), Проектиране на съвременни предсказващи регулатори.

Материал за лекции по цифрови системи за управление, страници 268–281.

3. Allgöwer, F., Badgwell, T. A., Qin, J. S., Rawlings, J. B., and Wright, S. J. (1999). *Advances in Control – Highlights of ECC'99*, chapter 12. *Nonlinear Predictive Control and Moving Horizon Estimation*, pages 391–449. Springer, London.
4. Åström, J. (1970). *Introduction to Stochastic Control Theory*, Academic Press.
5. Åström, J. and Wittenmark, B. (1990), *Computer-Controlled Systems: Theory and Design*, Prentice Hall, Englewood Cliffs NJ, USA, second edition.
6. Bitmead, R., Gevers, M., and Wertz, V. (1990). *Adaptive Optimal Control: The Man's GPC*, Prentice Hall. Australia.
7. Camacho, F. and Bordons, C. (1998). *Model Predictive Control*. Springer, Berlin.
8. Chisci, L., Giarré, L. and Moca, E. (1994). Indirect and implicit adaptive predictive control of the benchmark plant, *Automatica*, 30, 577–584.
9. Clarke, D. W., Mohtadi, C., and Tuffs, P. S. (1987). Generalized predictive control. *Automatica*, 23(2):137–148.
10. Cutler, C. R. (1983). *Dynamic Matrix Control: An Optimal Multivariable Control Algorithm With Constraints*. PhD thesis, University of Houston.
11. Demircioğlu, H. and Clarke D., W. (1993). Generalized predictive control with end-point state weighting. *IEE Proceeding–D*, 140, 275–282.
12. Demircioğlu, H. and Gawthrop P., J. (1991). Continuous-time generalized predictive control (CGPC), *Automatica*, 28, 55–74.
13. Garcia, E., Prett, M., and Morari, M. (1989). Model predictive control: Theory and practice—a survey. *Automatica*, 25(3):335–348.
15. Greco, C., Menga, G., Mosca, E., and Zappa, G. (1984). Performance improvements of self–



- tuning controllers by multistep horizons: The musmar approach. *Automatica*, 20(5):681–699.
16. Henson, M. A. and Seborg D., E. (1993). Theoretical analysis of unconstrained nonlinear model predictive control. *International Journal of Control*, 58, 1053–1080.
  17. Kouvaritakis, B. and Rossiter, J. A., (1993). Multivariable stable generalized predictive control, *IEE Proceeding—D*.
  18. Lauritsen M. B. (1997) Delta-domain predictive control and identification for control, pages 106–115. Department of Mathematical Modelling (IMM), Technical University of Denmark.
  19. Maciejowski, J. M. (2001). *Predictive Control with Constraints*. Addison–Wesley Pub Co, 1 edition.
  20. Mayne, D. Q., Rawlings, J. B., Rao, C. V., and Scolaert, P. O. M. (2000). Constrained model predictive control: Stability and optimality. *Automatica*, 36:789–814.
  21. Mohtadi, C. (1988) On the role on prefiltering in parameter estimation and control, In *Adaptive Control Strategy for Industrial Use*.
  22. Morari, M. and Lee, J. H. (1999). Model predictive control: past, present and future. *Computers and Chemical Engineering*, 23:667–682.
  23. Morari, M. and Ricker, N. L. (1998). *Model Predictive Control Toolbox*. The MathWorks, Inc., Natick, MA. USA.
  24. Mosca, E. (1995). *Optimal Predictive and Adaptive Control*, Prentice–Hall, Engelwood Cliffs NJ, USA.
  25. Mosca, E., Zappa, G., and Lemos, J. M. (1989). Robustness of multipredictor adaptive regulators: Musmar. *Automatica*, 25(4):521–529.
  26. Nørgaard M., (2000). Control by Instantaneous Linearization. Neural network based control system design. *Toolkit for MATLAB*, pages 2\_17–2\_21. Department for Automation, Technical University of Denmark.

27. Ordys, A. W. (1993). Model–system parameter mismatch in GPC control, *International Journal of Adaptive Control and Signal Processing*, 7.
28. Pannocchia, G. and Rawlings, J. B. (2001). The velocity algorithm LQR: a survey. Technical Report 2001–01, Department of Chemical Engineering, Madison, WI.
29. Puerto R., (2002). Diseño de reguladores predictivos. Control avanzado de sistemas, Practica 3. Ingeniería Industrial. Madrid, España.
30. Rawlings, J. B. (2000). Tutorial overview of model predictive control. *IEEE Control Systems Magazine*, (3):38–52.
31. Richalet, J., Rault, A., L. Testud, J., and Papon, J. (1978). Model predictive heuristic control. *Automatica*, 14:413–428.
32. Robinson, B. D. and Clarke, D. W. (1991). Robustness effects of a prefilter in Generalized Predictive Control, *IEE Proceedings–D*, 138, 2–8.
33. Shoön T., (2001). Identification for Predictive Control – A Multiple Model Approach. Master’s Degree Thesis, pages 9–14. Department of Electrical Engineering, Linköping University, Sweden.
34. Soeterboek, R. (1992). *Predictive Control: A Unified Approach*. Prentice Hall, New York.
35. Wang, W., (1994). A direct adaptive generalized predictive control algorithm for MIMO systems. *International Journal of Control*.
36. Wang, W. and Henriksen, R. (1994). A direct adaptive generalized predictive controller and some of its global convergence properties. *Journal of Modeling, Identification and Control*.
37. Wang, W. and Li, G., Y. (1995). A new direct generalized predictive control algorithm. In *ECC 95*, pp. 61–64, Rome, Italy.
39. Zhu, Y. (1998). Multivariable process identification for MPC: the asymptotic method and its applications. *J. Proc. Cont.*