UNIVERZA V LJUBLJANI
FAKULTETA ZA MATEMATIKO IN FIZIKO

Finančna matematika – 2. stopnja

Tadej Šnajder

Metode ADI za reševanje Sylvestrove enačbe

Magistrsko delo

Mentor: prof. dr. Bor Plestenjak

Ljubljana, 2018

Zahvala

Magistrskega dela ne bi bilo brez pomoči mentorja prof. dr. Bora Plestenjaka, ki se mu zahvaljujem za vso pomoč, čas in usmeritve s strokovnimi napotki.

Posebej se zahvaljujem tudi svoji družini in punci Veroniki, ki so me podpirali in mi brezpogojno stali ob strani.

Zahvala tudi vsem prijateljem, sostanovalcem in profesorjem, ki ste me spremljali na moji študijski poti.

Najlepša hvala.

Kazalo

Kazalo slik

Kazalo tabel

Program dela

Navodila:

Sylvestrovo matrično enačbo lahko v primeru, ko matrike niso prevelike, učinkovito numerično rešimo npr. z Bartels-Stewartovo metodo. V primeru večjih dimenzij in razpršenih matrik pa iščemo metode za čim boljšo aproksimacijo rešitve z matriko nizkega ranga. Ena izmed takšnih je metoda ADI. Za reševanje Sylvestrove enačbe je bila prvič predstavljena leta 2008. Potrebno je predstaviti metodo in jo testirati na numeričnih primerih. Predstavljene naj bodo tudi nekatere izboljšave predstavljene metode, ki so se pojavile v kasnejši literaturi.

prof. dr. Bor Plestenjak

Povzetek

V magistrskem delu smo se osredotočili na reševanje Sylvestrove enačbe in enačbe Ljapunova, kot poseben primer Sylvestrove enačbe, z metodami ADI. Če dimenzije matrik v Sylvestrovi matrični enačbi niso prevelike, jo lahko rešimo s pomočjo direktnih algoritmov, kot je na primer Bartels-Stewartova metoda. Ko imamo v Sylvestrovi enačbi razpršene matrike velikih dimenzij, namesto direktnih algoritmov raje uporabimo iteracijske metode, med katere spadajo tudi metode ADI.
V magistrskem delu so najprej predstavljene povezave med teorijo upravljanja linearnih kontrolnih sistemov in enačbo Ljapunova, kot poseben primer Sylvestrove enačbe. Hkrati so navedene tudi predpostavke, ki jih uporabljamo v magistrskem delu. Sledi predstavitev Smithove metode, metode ADI in nekaj njenih najpomembnejših razširitev.
Nato je predstavljen problem izbire premikov, ki vplivajo na hitrost konvergence metod ADI, podane so ocene za konvergenco metod ADI ter nekateri pristopi, s katerimi rešujemo problem izbire premikov.
Predstavljene so tudi implementacije metod ADI v Matlabu. Narejena je bila primerjava premikov in primerjava metod na nekaterih primerih iz spletne zbirke Slicot.

Math. Subj. Class. (2018): 15A06, 15A24, 40C05, 65F10, 65F30, 65F50, 65H10, 93C05

Ključne besede: metoda ADI, Sylvestrova enačba, enačba Ljapunova, razpršene matrike, Smithova metoda, metoda ADI nizkega ranga s faktorji Choleskega, faktorizirana metoda ADI, iterativne metode

Abstract

In master's thesis we focused in solving the Sylvester equation and the Lyapunov equation, as a special case of the Sylvester equation, by using ADI methods. If the matrix dimensions in the Sylvester matrix equation are not too large, then it can be solved by means of direct algorithms, such as the Bartels-Stewart method. When we are solving Sylvester equation with sparse matrices of large dimensions, iterative methods, such as ADI methods, are preferred over direct algorithms.
In the thesis the connections between the theory of linear control systems and the Lyapunov equation, as a special case of the Sylvester equation, are first presented. At the same time, the assumptions used in the thesis are also presented. Then the Smith method, the ADI method and some of the most important extensions of the ADI method are presented.
Next, the selection of shifts, which determine the rate of convergence of ADI methods, is presented. Some approaches to select the shifts are given.
Implementations of algorithms from the previous chapters in Matlab are presented. Comparison of shifts and comparison of methods was obtained for some test cases from the online benchmark collection Slicot.

Math. Subj. Class. (2018): 15A06, 15A24, 40C05, 65F10, 65F30, 65F50, 65H10, 93C05

Keywords: ADI method, Sylvester equation, Lyapunov equation, sparse matrices, Smith method, low rank Cholesky ADI method, factored ADI method, iterative methods

1. Uvod

Za teorijo upravljanja linearnih kontrolnih sistemov je izrednega pomena zvezna enačba Ljapunova, ki je poseben primer splošnejše Sylvestrove enačbe. Na začetku magistrskega dela so zato predstavljeni nekateri ključni pojmi in postopki, s katerimi se srečamo v teoriji upravljanja linearnih kontrolnih sistemov in njihove povezave z rešitvijo zvezne enačbe Ljapunova. Podani so privzetki za matrike, s katerimi je zagotovljeno učinkovito reševanje enačbe Ljapunova in Sylvestrove enačbe z metodami ADI. Predstavljena je tudi predhodnica metode ADI, Smithova metoda, njena razširitev na metodo ADI in izboljšave metode ADI, ki omogočajo iskanje rešitve Sylvestrove enačbe v faktoriziranih oblikah. Sledi predstavitev problema iskanja premikov, ki vplivajo na hitrost konvergence metode ADI. Na koncu je predstavljena implementacija metod ADI in algoritmov za iskanje premikov v Matlabu in testiranje le-teh na zbirki podatkov Slicot.

2. Motivacija in privzetki

V tem poglavju bo predstavljena motivacija za raziskovanje ADI metod, predvsem s strani teorije upravljanja linearnih kontrolnih sistemov. Navedeni bodo tudi privzetki, ki se bodo uporabljali skozi celotno magistrsko nalogo.

Dan imamo zvezen **linearni časovno invariantni dinamičen sistem(LTI)**

$$\begin{aligned} &\dot{x}(t) = Ax(t) + Bu(t),\ x(t_0) = x_0,\ t \geq t_0, \\ &\dot{y}(t) = Cx(t) + Du(t) \end{aligned} \tag{1}$$

z matrikami $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{r\times n}$, $D \in \mathbb{R}^{r\times m}$. Vektorju $x(t) \in \mathbb{R}^n$ rečemo **vektor stanja**, $u(t) \in \mathbb{R}^m$ **vhodni signal** in $y(t) \in \mathbb{R}^r$ **izhodni signal**. Ko sta $r = 1$ in $m = 1$ sistemu (1) pravimo **SISO sistem** (ang. *single-input single-output*), če pa sta oba $r > 1$ in $m > 1$ pa sistemu (1) pravimo **MIMO sistem** (ang. *multiple-input multiple-output*).
Če na sistemu (1) uporabimo Laplaceovo transfomacijo, dobimo

$$\begin{aligned} &s\mathcal{L}(x)(s) = A\mathcal{L}(x)(s) + B\mathcal{L}(u)(s), \\ &\mathcal{L}(y)(s) = C\mathcal{L}(x)(s) + D\mathcal{L}(u)(s). \end{aligned} \tag{2}$$

Rešitev transformiranega sistema (2), z začetnim pogojem $X(0) = 0$, je

$$\mathcal{L}y(s) = G(s)\mathcal{L}(u)(s),$$

kjer funkciji

$$G(s) = C(sI_n - A)^{-1}B + D \tag{3}$$

pravimo **prenosna funkcija** (ang. *transfer function*). Z I_n smo označili identično matriko dimenzije $n \times n$. Z vidika teorije matrika D ni pomembna, zato bomo privzeli, da je $D = 0$.
Za dani LTI sistem pravimo, da je **vodljiv**, če za poljubno začetno stanje x_0 in končno stanje x_1 obstaja končen t_1 in vhod $u(t)$, za $0 \leq t \leq t_1$, da pride sistem iz začetnega stanja x_0 v končno stanje x_1.
Izkaže se, da je vodljivost sistema odvisna samo od para matrik A in B, zato govorimo o vodljivosti para (A, B). Za vodljivost sta pomembna PBH kriterija (Popov-Belevitch-Hautus).

Izrek 2.1. *Naslednje trditve so ekvivalentne:*

(1) *Par* (A, B) *je vodljiv.*
(2) *Za vsako lastno vrednost* λ *matrike* A *velja* $\operatorname{rang}([A - \lambda I\ B]) = n$.
(3) *Za vsak lastni par* (λ, x) *matrike* A^T *velja* $x^T B \neq 0$.

Lastnim vrednostim λ matrike A, ki zadoščajo pogoju (3) izreka 2.1, pravimo, da so **vodljive**, v nasprotnem primeru pa pravimo, da niso vodljive.

Drug pomemben pojem, s katerim se srečamo pri proučevanju linearnih kontrolnih sistemov (1), je spoznavnost. Za sistem (1) pravimo, da je **spoznaven**, če obstaja tak $t_1 \geq 0$, da iz poznavanja $u(t)$ in $y(t)$, $0 \leq t \leq t_1$, lahko določimo začetno stanje $x(0)$. Spoznavnost sistema (1) je, podobno kot pri vodljivosti, odvisna samo od para matrik A in C, zato govorimo o spoznavnosti para (A, C).

Med vodljivostjo in spoznavnostjo velja povezava, da je spoznavnost para (A, C) ekvivalentna vodljivosti para (A^T, C^T). Z obema pojmoma se srečamo, ko poskušamo sistem (1) stabilizirati preko povratne zanke. Več o teoriji upravljanja linearnih sistemov lahko najdemo v [13].
Tipično nas zanima stabilnost dinamičnega sistema. Za homogeni sistem

$$\dot{x}(t) = Ax(t),\ x(0) = x_0, \tag{4}$$

pravimo, da je **asimptotično stabilen**, če gre $x(t)$ proti 0, ko gre $t \to \infty$, za vsako začetno stanje x_0. Hitro se lahko prepričamo, da je homogeni sistem (4) stabilen natanko tedaj, ko za vsako lastno vrednost matrike A, ki jo označimo z λ, velja $\mathbf{Re}(\lambda) < 0$. Takšno matriko potem definiramo kot **stabilno matriko**.
Če se želimo izogniti računanju lastnih vrednosti matrike A, si lahko pomagamo tudi s kriterijem Ljapunova. Za homogene sisteme oblike (4) se kriterij Ljapunova izrazi preko matrične enačbe Ljapunova, ki je poseben primer Sylvestrove matrične enačbe.
Zvezna Sylvestrova matrična enačba je matrična enačba oblike

$$AX + XB = C, \tag{5}$$

kjer so $A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{m\times m}$ in $C \in \mathbb{R}^{n\times m}$. Kot poseben primer Sylvestrove enačbe imamo **zvezno matrično enačbo Ljapunova**

$$AX + XA^T = -Q, \tag{6}$$

kjer sta A, $Q \in \mathbb{R}^{n\times n}$. Za matriko Q se velikokrat zahteva, da je simetrična. Če je Q simetrična matrika, potem velja naslednji izrek:

Trditev 2.2. *Rešitev enačbe Ljapunova je simetrična pozitivno definitna, če je A stabilna matrika in Q simetrična pozitivno definitna matrika.*

Dokaz trditve lahko ponovno najdemo v [13]. Če imamo opravka s kompleksnimi matrikami, v enačbi (6) namesto simetričnosti Q zahtevamo, da je Q hermitska matrika, transponiranje matrike A se zamenja s hermitiranjem, z A^H pa označimo konjugirano transponiranko oziroma hermitsko matriko matrike A.

V magistrski nalogi bomo zahtevali, da sta A in B hermitski (simetrični za realne matrike) in stabilni. Ta zahteva je sicer za neposredno reševanje Sylvestrove enačbe (5) odveč, vendar bomo potrebovali stabilnost za enoličnost rešitve in za določitev problemov, ki jih lahko rešujemo z metodami ADI, hermitskost matrik pa bomo potrebovali za učinkovito iskanje parametrov, ki določajo konvergenco in hitrost konvergence metod.
Pokazali bomo, da lahko Sylvestrovo enačbo zapišemo kot sistem linearnih enačb. Za to potrebujemo naslednji dve definiciji.

Definicija 2.3. ***Kroneckerjev produkt*** *matrik $A \in \mathbb{R}^{n\times m}$ z elementi a_{ij} in $B \in \mathbb{R}^{s\times t}$ definiramo kot matriko dimenzije $ns \times mt$, s predpisom*

$$A \otimes B = \begin{bmatrix} a_{11}B & \dots & a_{1m}B \\ \vdots & & \vdots \\ a_{n1}B & \dots & a_{nm}B \end{bmatrix}. \tag{7}$$

Definicija 2.4. ***Vektorizacija matrike*** $A \in \mathbb{R}^{n\times m}$ *je vektor velikosti* nm*, bločne oblike*

$$\text{vec}(A) = \begin{bmatrix} a_1 \\ \vdots \\ a_m \end{bmatrix}, \tag{8}$$

kjer smo z a_i*,* $i = 1, 2, \ldots, m$*, označili stolpce matrike* A*.*

S pomočjo Kroneckerjevega produkta in vektorizacije matrik lahko zapišemo Sylvestrovo enačbo (5) v obliki sistema linearnih enačb

$$(I_n \otimes A + B^T \otimes I_m)\text{vec}(X) = \text{vec}(C). \tag{9}$$

Lema 2.5. *Naj bosta dani matriki* $A \in \mathbb{R}^{n\times n}$ *in* $B \in \mathbb{R}^{m\times m}$*. Označimo lastne vrednosti matrike* A *z* λ_i*,* $i = 1, 2, \ldots, n$*, in lastne vrednosti matrike* B *z* μ_j*,* $j = 1, 2, \ldots, m$*. Potem velja*

(1) *lastne vrednosti* $A \otimes B$ *so* $\lambda_i\mu_j$*,*
(2) *lastne vrednosti* $I_n \otimes A + B^T \otimes I_m$ *so* $\lambda_i + \mu_j$*.*

Izpeljavo pretvorbe Sylvestrove enačbe (5) na sistem linearnih enačb in dokaz leme 2.5 najdemo v [13].
Nabor vseh lastnih vrednosti matrike A definiramo kot **spekter matrike** A in ga označimo s $\sigma(A)$. Za rešitev Sylvestrove enačbe (5) je pomembna naslednja posledica leme 2.5.

Izrek 2.6. *Sylvestrova enačba (5) ima enolično rešitev natanko tedaj, ko velja*

$$\sigma(A) \cap \sigma(-B) = \emptyset,$$

tj. ko A *in* B *nimata nobene skupne lastne vrednosti.*

Dokaz.
Sylvestrovo enačbo (5) lahko zapišemo v obliki sistema linearnih enačb (9), ki je enolično rešljiv natanko tedaj, ko so lastne vrednosti matrike $I_n \otimes A + B^T \otimes I_m$ neničelne.
Označimo lastne vrednosti matrike A z λ_i, $i = 1, 2, \ldots n$, in z μ_j, $j = 1, 2, \ldots m$, označimo lastne vrednosti matrike B. Ker so po lemi 2.5 lastne vrednosti matrike $I_n \otimes A + B^T \otimes I_m$ enake $\lambda_i + \mu_j$, lahko rešimo (9) natanko tedaj, ko

$$\lambda_i + \mu_j \neq 0$$

za vsak par i, j, kar pa lahko kompaktneje zapišemo kot

$$\sigma(A) \cap \sigma(-B) = \emptyset.$$

□

Posledica 2.7. *Če je* A *stabilna matrika, potem je enačba Ljapunova (6) enolično rešljiva.*

Dokaz.
Po izreku 2.6 je enačba Ljapunova enolično rešljiva natanko tedaj, ko A in $-A^T$ nimata nobene skupne lastne vrednosti. Ker je A stabilna, imajo vse njene lastne vrednosti negativen realni del. Posledično, ker imata A in A^T enake lastne vrednosti, morajo imeti tudi vse lastne vrednosti A^T negativen realni del, lastne vrednosti $-A^T$ pa morajo potem imeti pozitiven realni del. Sledi, da morajo biti lastne vrednosti A in $-A^T$ različne in je enačba Ljapunova res enolično rešljiva.

□

V magistrski nalogi bomo privzeli, da za spektra matrik A in B velja

$$\sigma(A) \cap \sigma(-B) = \emptyset.$$

Tako imamo zagotovljeno rešitev Sylvestrove enačbe oz. enačbe Ljapunova (kjer je $B = A^T$), ki je enolična in simetrična pozitivno definitna.

S Sylvestrovo enačbo in enačbo Ljapunova se srečamo marsikje v uporabni matematiki in igrata ključno vlogo v teoriji upravljanja linearnih kontrolnih sistemov. Naslednja izreka nam povesta povezavo med stabilnostjo matrike A, rešitvijo enačbe Ljapunova ter spoznavnostjo in vodljivostjo sistema (1).

Izrek 2.8. *Naj bo X rešitev zvezne enačbe Ljapunova, kjer je matrika $Q = C^TC$. Potem velja:*

(1) *če je X simetrična pozitivno definitna in je par (A, C) spoznaven, potem je A stabilna,*
(2) *če je A stabilna in je par (A, C) spoznaven, potem je X simetrična pozitivno definitna,*
(3) *če je A stabilna in je X simetrična pozitivno definitna, potem je par (A, C) spoznaven.*

Izrek 2.9. *Naj bo X rešitev zvezne enačbe Ljapunova, kjer je matrika $Q = BB^T$. Potem velja:*

(1) *če je X simetrična pozitivno definitna in je par (A, B) vodljiv, potem je A stabilna,*
(2) *če je A stabilna in je par (A, B) vodljiv, potem je X simetrična pozitivno definitna,*
(3) *če je A stabilna in je X simetrična pozitivno definitna, potem je par (A, B) vodljiv.*

Dokaza obeh izrekov najdemo v [13]. Rešitvi enačbe Ljapunova za matriki A in $Q = BB^T$ pravimo **vodljivostna Gramova matrika** in jo označimo z X_c, rešitvi enačbe Ljapunova za matriki A in $Q = C^TC$ pa pravimo **spoznavnostna Gramova matrika** in jo označimo z X_o.
Vodljivost in spoznavnost sta pomembni tudi, ko poskušamo stabilizirati sistem preko matrike povratne zveze K tako, da bo zaprtozančni sistem stabilen ali ko poskušamo poiskati takšno povratno matriko K, da bo imel zaprtozančni sistem točno predpisane pole.

Izrek 2.10. *Za realni matriki A in B linearnega sistema $\dot{x}(t) = Ax(t) + Bu(t)$ obstaja realna matrika K, da ima $A - BK$ predpisan spekter, zaprt za konjugiranje, natanko tedaj, ko je par (A, B) vodljiv.*

Če A in B pripadata SISO sistemu, bo K enolična, sicer pa imamo neskončno rešitev. Pripadajočo matriko povratne zveze K, ki stabilizira sistem, lahko izračunamo tudi s pomočjo enačbe Ljapunova.

Izrek 2.11. *Naj bo par (A, B) vodljiv in $\beta > |\lambda_{max}|$, kjer je λ_{max} lastna vrednost matrike A, ki ima po absolutni vrednosti največji realni del. Če vzamemo za matriko povratne zveze $K = B^TZ^{-1}$, kjer je Z simetrična pozitivno definitna rešitev enačbe Ljapunova*

$$-(A + \beta I_n)Z + Z(-(A + \beta I_n))^T = -2BB^T, \tag{10}$$

potem je matrika $A - BK$ stabilna.

Dokaza izrekov 2.10 in 2.11 najdemo v [13]. Za reševanje enačb (5) in (6) poznamo kar nekaj direktnih algoritmov kot so na primer Hammarlingov algoritem, Bartels-Stewartov algoritem ali Hessenberg-Schurova metoda, ki najprej reducirajo sistem s pomočjo razcepa Choleskega, Schurove forme ali Hessenberg-Schurove forme in rešijo sistem enačb. Težava le-teh algoritmov je, da so računsko zahtevni (reda $\mathcal{O}(n^3+m^3)$ za velike m in n ter porabijo ogromno pomnilnika. Zato jih je v primeru velikih matrik bolje zamenjati z algoritmi, ki iterirajo k iskani rešitvi enačbe (5).
Alternativno se poskuša zmanjšati dimenzijo problema in zmanjšan sistem nato rešiti s pomočjo direktnih algoritmov. S tem se ukvarja **redukcija modela** (ang. *model order reduction*), ki je povezana z rešitvijo problema

$$AX + XA^T = -BC, \tag{11}$$

kjer so matrike A, B in C iz sistema (1). Rešitvi (11) pravimo **križna Gramova matrika** (ang. *cross gramian*) in jo označimo X_{co}. Popularen pristop k redukciji modela je metoda **uravnoteženega rezanja** (ang. *balanced truncation*). Uravnoteženo rezanje si bomo ogledali v podpoglavju 3.4 kot motivacijo za razvoj metod ADI v faktorizirani obliki.

Podobno kot v [2] se bomo v magistrskem delu ukvarjali s Sylvestrovo enačbo (5) za velike matrike A in B ter bomo privzeli, da sta matriki A in B razpršeni. Za matriko $A \in \mathbb{R}^{n\times n}$ pravimo, da je **razpršena** (ang. *sparse*), če je množenje A z vektorjem časovne zahtevnosti $\mathcal{O}(n)$. Za matriko C bomo privzeli, da je nizkega ranga in jo lahko zapišemo kot

$$C = GF^T,$$

dimenzij $G \in \mathbb{R}^{n\times p}$, $F \in \mathbb{R}^{m\times p}$, kjer je $p \ll m, n$. Če imamo opravka s kompleksnimi matrikami, se F^T zamenja s F^H. Če sta izpolnjena zgornja pogoja, se izkaže, da metode v večini primerov hitro konvergirajo k rešitvi.
Iteracijski algoritmi, ki se uporabljajo za reševanje takšnih problemov, so tipično osnovani ali na podprostorih Krilova ali na ADI (Alternative Directional Implicit) iteraciji. Prednost prvih je, da ni potrebno vedeti ničesar o spektru matrik A in B, medtem ko ADI iteracije pogosto konvergirajo hitreje, če nam uspe učinkovito poiskati ustrezen nabor (sub)optimalnih premikov za A in B [2].

3. Metode ADI

V tem poglavju bodo predstavljene metode ADI. Najprej si bomo pogledali Smithovo metodo in nato metodo ADI. Nato si bomo ogledali metodi ADI, ki jih dobimo s faktorizacijo rešitve enačbe Ljapunova in Sylvestrove enačbe. Na koncu si bomo pogledali še, kako lahko izboljšamo hitrost konvergence z Galerkinovim pogojem.

3.1. Smithova iteracija

Prvotno sta Peaceman in Rachford razvila ADI iteracijo pri reševanju eliptičnih parcialnih diferencialnih enačb, kjer se srečamo z naslednjim problemom:
Za simetrično pozitivno definitno matriko $C \in \mathbb{R}^{n\times n}$ in vektor s dimenzije n rešujemo sistem linearnih enačb

$$Cu = s. \tag{12}$$

Na sistemu linearnih enačb (12) lahko uporabimo ADI iteracijo, če lahko C zapišemo kot vsoto matrik H in V, za kateri znamo učinkovito rešiti linearna sistema

$$\begin{aligned}(H + \alpha I_n)v &= r \text{ in}\\ (V + \alpha I_n)w &= t,\end{aligned} \tag{13}$$

za primerno izbran parameter α in dana vektorja s in t. Če sta matriki H in V simetrični pozitivno definitni, potem obstaja nabor pozitivnih parametrov α_r, $r = 1, 2, \ldots,$ za katere iteracija

$$\begin{aligned}(H + \alpha_r I_n)u^{(r-\frac{1}{2})} &= (\alpha_r I_m - V)u_{r-1} + s,\\ (V + \alpha_r I_n)u_r &= (\alpha_r I_m - H)u^{(r-\frac{1}{2})} + s\end{aligned} \tag{14}$$

konvergira. Če H in V še komutirata in lahko uporabimo optimalen nabor parametrov, potem je konvergenca iteracije (14) hitrejša od vseh predhodnih metod za reševanje problema (12) [20].

Definicija 3.1. *Za dani realni, simetrični, komutirajoči matriki H in $V \in \mathbb{R}^{n\times n}$, katerih vsota je simetrična pozitivno definitna matrika, in dani vektor $v \in \mathbb{R}^n$ iščemo vektor $u \in \mathbb{R}^n$, ki reši linearen sistem*

$$Cu = (H + V)u = v.$$

*Zgornji problem definiramo kot **problem modela ADI** (ang. ADI model problem).*

Definicijo problema modela ADI lahko naravno razširimo, če v iteracijo (14) uvedemo še en nabor parametrov β_r, $r = 1, 2, \ldots,$ tako, da namesto (14) dobimo

$$\begin{aligned}(H + \alpha_r I_n)u^{(r-\frac{1}{2})} &= (\alpha_r I_m - V)u_{r-1} + s,\\ (V + \beta_r I_n)u_j &= (\beta_r I_m - H)u^{(r-\frac{1}{2})} + s.\end{aligned} \tag{15}$$

V tem primeru lahko zahtevo v definiciji 3.1, da sta H in V simetrični pozitivno definitni, sprostimo na pogoj, da za poljubni lastni vrednosti $\lambda \in \sigma(H)$ in $\mu \in \sigma(V)$ velja

$$\lambda + \mu > 0. \tag{16}$$

Učinkovitost metode je motivirala iskanje posplošitev na podobne probleme, med katerimi se nahaja tudi reševanje Sylvestrove enačbe (5) [20]. Zapišimo Sylvestrovo enačbo kot

$$(I_n \otimes A + B^T \otimes I_m)\mathrm{vec}(X) = \mathrm{vec}(C). \tag{17}$$

Če vzamemo

$$\begin{aligned} H &= I_n \otimes A \text{ in} \\ V &= B^T \otimes I_m, \end{aligned} \tag{18}$$

potem H in V vedno komutirata, ker velja $HV = B^T \otimes A = VH$. Če sta A in B simetrični pozitivno definitni, sledi, da sta H in V simetrični pozitivno definitni. Potem je tudi $H + V$ simetrična pozitivno definitna, saj je vsota simetričnih matrik simetrična matrika, po lemi 2.5 pa imamo pozitivno definitnost. Sedaj lahko uporabimo isto idejo za $-A$ in $-B$, kjer sta A in B simetrični in stabilni matriki. Posledično vidimo, da reševanje Sylvestrove enačbe (5) ustreza problemu modela ADI.

Smith [15] je za reševanje Sylvestrove enačbe (5) razvil naslednji iterativen postopek, kjer je uporabil samo en parameter α. Sylvestrovo enačbo (5) zapišemo v obliki

$$(\alpha I_n + A)X(\alpha I_m + B) - (\alpha I_n - A)X(\alpha I_m - B) = 2\alpha C, \tag{19}$$

kjer $I_n \in \mathbb{R}^{n\times n}$ označuje identiteto, analogno za $I_m \in \mathbb{R}^{m\times m}$.
Prepričajmo se, da enačba (19) za $\alpha \neq 0$ res ustreza enačbi (5). Razpišimo najprej izraz za drugi člen enačbe $(\alpha I_n - A)X(\alpha I_m - B)$. Dobimo

$$\begin{aligned} &(\alpha I_n - A)X(\alpha I_m - B) \\ &= \alpha I_n X(\alpha I_m - B) - AX(\alpha I_m - B) \\ &= \alpha^2 X - \alpha XB - \alpha AX + AXB. \end{aligned} \tag{20}$$

Analogno dobimo za prvi člen enačbe (19)

$$\alpha^2 X + \alpha XB + \alpha AX + AXB.$$

Ko vstavimo dobljene izraze v (19), dobimo

$$2\alpha XB + 2\alpha AX = 2\alpha C.$$

Če sedaj pomnožimo obe strani enačbe z $\frac{1}{2\alpha}$, kjer je $\alpha \neq 0$, dobimo ravno (5).

Pomnožimo enačbo (19) z leve z $(\alpha I_n + A)^{-1}$ in z desne z $(\alpha I_m - B)^{-1}$ ter definiramo matrike

$$\begin{aligned} G &= (\alpha I_n + A)^{-1}(\alpha I_n - A), \\ H &= (\alpha I_m - B)(\alpha I_m + B)^{-1}, \\ K &= 2\alpha(\alpha I_n + A)^{-1}C(\alpha I_m - B)^{-1}. \end{aligned} \tag{21}$$

Potem lahko enačbo (19) poenostavimo v

$$X - GXH = K. \tag{22}$$

Rešitev enačbe (22) je

$$X = \sum_{i=1}^{\infty} G^{i-1} K H^{i-1}. \tag{23}$$

Pogoj za konvergenco vsote je $\rho(G)\rho(H) < 1$, kjer je $\rho(G)$ po absolutni vrednosti največja lastna vrednost oz. spektralni radij matrike G. Tedaj obstaja $\rho(G)\rho(H) < r < 1$ in taka konstanta M, da v spektralni normi velja $\|G^{i-1}\|_2\|H^{i-1}\|_2 < Mr^{i-1}$ za $i = 1, 2, \ldots$ [15]. Posledično je (23) majorizirana z

$$M\|K\|_2 \sum_{i=1}^{\infty} r^{i-1} = M\|K\|_2 \sum_{j=0}^{\infty} r^j = \frac{M\|K\|_2}{1-r}$$

za primerno izbran pozitiven $r < 1$.
Posebej za enačbo Ljapunova (6), s simetrično matriko A, velja $H = G^T$. Ker preslikava

$$x \mapsto \frac{\alpha - x}{\alpha + x} = \frac{1 - \frac{x}{\alpha}}{1 + \frac{x}{\alpha}}$$

za $\alpha < 0$ preslika $\mathbb{R}_- \mapsto [0, 1)$, je konvergenca pri tem pogoju izpolnjena. To vidimo tako, da pogledamo diagonalizacijo matrike A. Ker je A simetrična jo lahko diagonaliziramo kot $A = PDP^{-1}$ in velja, da so lastne vrednosti A realne. Ker je matrika A stabilna, so njene lastne vrednosti negativne. Sledi

$$\begin{aligned} G &= (\alpha I_n + A)^{-1}(\alpha I_n - A) \\ &= (\alpha I_n + PDP^{-1})^{-1}(\alpha I_n - PDP^{-1}) \\ &= P(\alpha I_n + D)^{-1}(\alpha I_n - D)P^{-1}, \end{aligned} \tag{24}$$

torej $\rho(G) < 1$. V poglavju 4 bomo videli, da sta konvergenca in hitrost konvergence k rešitvi Sylvestrove enačbe (5) odvisni od izbire parametrov. Zaenkrat pa privzemimo, da smo izbrali primeren parameter α in da vrsta (23) konvergira.
Preverimo še stacionarnost.

$$\sum_{i=1}^{\infty} G^{i-1}KH^{i-1} - G\Big(\sum_{i=1}^{\infty} G^{i-1}KH^{i-1}\Big)H = K,$$

$$(K + GKH + G^2KH^2 + \cdots) - (GKH + G^2KH^2 + \cdots) = K.$$

Na levi se vsi členi razen K odštejejo, torej je (23) stacionarna točka (22). Posledično zaporedje

$$X^{(0)} = K$$

$$X^{(r)} = X^{(0)} + GX^{(r-1)}H,\ r = 1, 2, \ldots \tag{25}$$

konvergira k rešitvi enačbe (5).
Smith je tudi ugotovil, da lahko povečamo red konvergence iz linearnega na kvadratičnega, če na vsakem koraku "kvadriramo" G in H:

$$\begin{aligned} G_0 &= G, \\ H_0 &= H, \\ G_r &= G_{r-1}^2,\ r = 1, 2, \ldots \\ H_r &= H_{r-1}^2,\ r = 1, 2, \ldots \end{aligned}$$

in

$$\tilde{X}^{(r)} = \tilde{X}^{(r-1)} + G_{r-1}\tilde{X}^{(r-1)}H_{r-1},\ r = 1, 2, \ldots \tag{26}$$

Metodo posledično imenujemo **kvadrirana Smithova metoda** (ang. *squared Smith method*).

Pokažimo še, da r-ti korak iteracije (26) ustreza 2^{r-1}-ti iteraciji enačbe (25). Vidimo, da r-ta iteracija (25) ustreza r-ti delni vsoti

$$X^{(r)} = \sum_{i=1}^{r} G^{i-1} K H^{i-1}.$$

Če sedaj privzamemo, da je

$$\tilde{X}^{(r)} = X^{(2^{r-1})},$$

velja

$$\begin{aligned}
\tilde{X}^{(r+1)} &= \tilde{X}^{(r)} + G_r \tilde{X}^{(r)} H_r \\
&= X^{(2^{r-1})} + G_{k-1} X^{(2^{r-1})} H_{k-1} \\
&= \sum_{i=1}^{2^{r-1}} G^{i-1} K H^{i-1} + G^{2^{r-1}} \Big(\sum_{i=1}^{2^{r-1}} G^{i-1} K H^{i-1} \Big) H^{2^{r-1}} \\
&= \sum_{i=1}^{2^{r-1}} G^{i-1} K H^{i-1} + \sum_{i=2^{r-1}+1}^{2^r} G^{i-1} K H^{i-1} \\
&= \sum_{i=1}^{2^r} G^{i-1} K H^{i-1} = X^{(2^r)}.
\end{aligned} \tag{27}$$

Kljub izboljšanemu redu konvergence se je slednja metoda izkazala za neprivlačno, saj kvadriranje v splošnem zgosti matrike, kar poveča računsko zahtevnost ter prinese dodatno numerično nestabilnost [19].

Zapišimo še oba algoritma v psevdokodi:

Algoritem 1: Smithova metoda za reševanje Sylvestrove enačbe

VHOD: Matrike A, B in C, α, število korakov i
IZHOD: Približek X_i za rešitev Sylvestrove enačbe X

$G = (\alpha I_n + A)^{-1}(\alpha I_n - A)$
$K = 2\alpha(\alpha I_n + A)^{-1} C (\alpha I_m - B)^{-1}$
$H = (\alpha I_m - B)(\alpha I_m + B)^{-1}$
$X_0 = K$
for $r = 0, 1, 2, \ldots, i$
    $X_{r+1} = K + G X_r H$
end

Algoritem 2: Kvadrirana Smithova metoda za reševanje Sylvestrove enačbe

---

VHOD: Matrike A, B in C, α, število korakov i
IZHOD: Približek X_i za rešitev Sylvestrove enačbe X

$G = (\alpha I_n + A)^{-1}(\alpha I_n - A)$
$K = 2\alpha(\alpha I_n + A)^{-1} C (\alpha I_m - B)^{-1}$

$H = (\alpha I_m - B)(\alpha I_m + B)^{-1}$
$X_0 = K$
for $r = 0, 1, 2, \ldots, i$
$\quad G = G^2$
$\quad H = H^2$
$\quad X_{r+1} = X_r + GX_rH$
end

3.2. **Posplošitev na metodo ADI**

Naslednjo lemo bomo potrebovali pri razširitvi Smithove metode na metodo ADI.

Lema 3.2. *Naj bo dana matrika B in števili $\alpha, \beta \in \mathbb{C}$, kjer je $\alpha \notin \sigma(-B)$. Velja komutativnost matrik*

$$(\alpha I_m + B)^{-1}(\beta I_m - B) = (\beta I_m - B)(\alpha I_m + B)^{-1}. \tag{28}$$

Dokaz.
Pomnožimo enačbo (28) z desne in z leve z $\alpha I_m + B$, dobimo

$$(\beta I_m - B)(\alpha I_m + B) = (\alpha I_m + B)(\beta I_m - B).$$

Razpišemo

$$\alpha\beta I_m + \beta I_m B - \alpha I_m B - B^2 = \alpha\beta I_m + \beta I_m B - \alpha I_m B - B^2$$

in vidimo, da sta obe strani res enaki.

□

Wachspress [19] je originalno Smithovo metodo posplošil v drugo smer. Namesto da bi uporabljal samo en parameter α, je dodal več parametrov. Za G, H in K je vzel matrike

$$\begin{aligned} G &= (\beta I_n + A)^{-1}(\alpha I_n - A), \\ H &= (\alpha I_m + B)^{-1}(\beta I_m - B), \\ K &= (\alpha + \beta)(\beta I_n + A)^{-1} C (\alpha I_m + B)^{-1}. \end{aligned} \tag{29}$$

Pripadajoča iteracijska shema

$$\begin{aligned} X_r - (\beta I_n + A)^{-1}(\alpha I_n - A) X_{r-1} (\alpha I_m + B)^{-1}(\beta I_m - B) \\ = (\alpha + \beta)(\beta I_n + A)^{-1} C (\alpha I_m + B)^{-1} \end{aligned} \tag{30}$$

je po lemi 3.2 ekvivalentna

$$\begin{aligned} X_r - (\beta I_n + A)^{-1}(\alpha I_n - A) X_{r-1} (\beta I_m - B)(\alpha I_m + B)^{-1} \\ = (\alpha + \beta)(\beta I_n + A)^{-1} C (\alpha I_m + B)^{-1}. \end{aligned}$$

Še več, s to formulacijo lahko na vsakem koraku spreminjamo α in β in bo metoda, s primerno izbiro parametrov, še vedno konvergirala k rešitvi, kot bomo videli v podpoglavju 3.3 in poglavju 4.
Wachspress [19] je ugotovil, da s takšno izbiro G, H in K dosežemo večjo učinkovitost kot pri Smithovi metodi, ko imata matriki A in B bistveno različna spektra.

Lema 3.3. *Naj bo dana matrika A in naj bo za $\beta \in \mathbb{C}$ matrika $\beta I_n + A$ obrnljiva ter $\alpha \in \mathbb{C}$. Potem velja*

$$(\beta I_n + A)^{-1}(-\alpha I_n + A) = I_n - (\alpha + \beta)(\beta I_n + A)^{-1}. \tag{31}$$

Dokaz.
Pomnožimo enačbo (31) z leve z $\beta I_n + A$ in dobimo

$$\begin{aligned} &(\beta I_n + A)(\beta I_n + A)^{-1}(-\alpha I_n + A) = (\beta I_n + A)(I_n - (\alpha + \beta)(\beta I_n + A)^{-1}) \\ &I_n(-\alpha I_n + A) = (\beta I_n + A)I_n - (\alpha + \beta)I_n, \\ &(-\alpha I_n + A) - (\beta I_n + A) = -(\alpha + \beta)I_n, \\ &-\alpha I_n - \beta I_n = -(\alpha + \beta)I_n. \end{aligned} \tag{32}$$

□

Lema 3.3 nam bo pomagala pri izpeljavi Wachspressovega iterativnega postopka, ki je bistvenega pomena za vse nadaljnje razširitve.

Iteracijsko enačbo (30) lahko zapišemo v naslednji obliki

$$\begin{aligned} X_r(\alpha I_m + B) - (\beta I_n + A)^{-1}(\alpha I_n - A)X_{r-1}(\alpha I_m + B)^{-1}(\beta I_m - B)(\alpha I_m + B) \\ = (\alpha + \beta)(\beta I_n + A)^{-1}C. \end{aligned} \tag{33}$$

Uporabimo lemi 3.2 in 3.3

$$X_r(\alpha I_m + B) - (\beta I_n + A)^{-1}(\alpha I_n - A)X_{r-1}(\beta I_m - B) = (I_n - (\beta I_n + A)^{-1}(-\alpha I_n + A))C,$$

preuredimo

$$X_r(\alpha I_m + B) = C - (\beta I_n + A)^{-1}(-\alpha I_n + A)C + (\beta I_n + A)^{-1}(\alpha I_n - A)X_{r-1}(-\beta I_m + B)$$

in končno izpostavimo $(\beta I_n + A)^{-1}(-\alpha I_n + A)$ ter uporabimo 3.2

$$X_r(\alpha I_m + B) = C - (-\alpha I_n + A)(\beta I_n + A)^{-1}(C - X_{r-1}(-\beta I_m + B)).$$

Sedaj lahko enačbo razbijemo na dva dela

$$\begin{aligned} (A + \beta I_n)X^{(r-\frac{1}{2})} &= C - X^{(r-1)}(B - \beta I_m), \\ X^{(r)}(B + \alpha I_m) &= C - (A - \alpha I_n)X^{(r-\frac{1}{2})}, \end{aligned} \tag{34}$$

$r = 1, 2, \ldots, t$, kjer smo umetno uvedli novo iteracijsko spremenljivko $X^{(r-\frac{1}{2})}$. Še več, na vsakem koraku lahko izbiramo drug par α in β. Dobimo

$$\begin{aligned} (A + \beta_r I_n)X^{(r-\frac{1}{2})} &= C - X^{(r-1)}(B - \beta_r I_m), \\ X^{(r)}(B + \alpha_r I_m) &= C - (A - \alpha_r I_n)X^{(r-\frac{1}{2})}. \end{aligned} \tag{35}$$

Iteracijska shema nam omogoča reševanje Sylvestrove enačbe (5) brez računanja inverzov in slabosti, ki jih slednji prinašajo. Iz samih enačb je razvidno, da lahko vsak korak izvajamo paralelno, tj. matrični enačbi pretvorimo na n in m linearnih sistemov tako, da rešujemo linearen sistem enačb za vsak stolpec matrike X, vrstico na naslednjem koraku, posebej. Slednje je zelo pomembno, saj z množenjem matrik z vektorji, ko sta matriki A in B razpršeni, prihranimo ogromno računanja v primerjavi z direktnimi algoritmi. Če matriki A in B nista razpršeni je časovna zahtevnost podobna kot za direktne algoritme [19].
Wachspress [19] je še ugotovil, če so matrike dovolj razpršene, da je število operacij v vsaki iteraciji pri metodi ADI za polovico manjše kot pri Smithovi metodi, potem

je metoda ADI bolj učinkovita, če dopuščamo relativno napako večjo od 10^{-8}.
V naslednjem podpoglavju si bomo podrobneje ogledali ADI iteracijo.

3.3. **Metoda ADI**

Iterativen sistem enačb, ki torej reši Sylvestrovo enačbo (5) z začetno ničelno matriko $X^{(0)} = 0$, je sestavljen iz dveh korakov.
Najprej rešimo enačbo

$$(A + \beta_r I_n)X^{(r-\frac{1}{2})} = C - X^{(r-1)}(B - \beta_r I_m) \tag{36}$$

za $X^{(r-\frac{1}{2})}$, nato nov približek poračunamo preko enačbe

$$X^{(r)}(B + \alpha_r I_m) = C - (A - \alpha_r I_n)X^{(r-\frac{1}{2})}, \tag{37}$$

$r = 1, 2, \ldots, i$.

Sistem (36), (37) imenujemo **ADI iteracija**. Uporablja se tudi izraz **metoda ADI**.
Parametrom α_r in $\beta_r, r = 1, 2, \ldots, i$, pravimo **premiki** (ang. *shifts*).
Prepričajmo se, da je rešitev Sylvestrove enačbe (5) fiksna točka sistema (36), (37).
Predpostavimo, da je $X = X^{(r-1)}$. Potem po enačbi (30) in identiteti (3.2) velja

$$\begin{aligned} X^{(r)} =& (\beta_r I_n + A)^{-1}(\alpha_r I_n - A)X(\alpha_r I_m + B)^{-1}(\beta_r I_m - B) \\ &+ (\beta_r + \alpha_r)(\beta_r I_n + A)^{-1}(AX + XB)(\alpha_r I_m + B)^{-1} \\ =& (\beta_r I_n + A)^{-1})\Big[(\alpha_r I_n - A)X(\beta_r I_m - B) \\ &+ (\beta_r + \alpha_r)(AX + XB)\Big](\alpha_r I_m + B)^{-1} \\ =& (\beta_r I_n + A)^{-1})\Big[(\beta_r I_n + A)X(\alpha_r I_m + B)\Big](\alpha_r I_m + B)^{-1} = X. \end{aligned} \tag{38}$$

□

Sledi neposredna implikacija, da bomo potem v vseh nadaljnjih iteracijah ostali v X.
Še več, X je fiksna točka, ki je neodvisna od izbire premikov na vsaki iteraciji, če ti zagotavljajo konvergenco metode. To bomo potrebovali pri oceni napake približka.
Ker je X fiksna točka (30), sledi, da lahko izrazimo razliko $X - X^{(r)}$ kot

$$X - X^{(r)} = (\beta_r I_n + A)^{-1}(\alpha_r I_n - A)\Big[X - X^{(r-1)}\Big](\alpha_r I_m + B)^{-1}(\beta_r I_m - B). \tag{39}$$

Napaka je odvisna od premikov na trenutni iteraciji in prejšnjega približka. Za korak k, $0 \leq k \leq r$, lahko s pomočjo rekurzivne formule (39) zapišemo razliko med rešitvijo X in k-tim približkom (spomnimo se, da je $X^{(0)} = 0$):

$$\begin{aligned} X - X^{(k)} &= \left[\prod_{i=1}^{k}(\beta_i I_n + A)^{-1}(\alpha_i I_n - A)\right](X - X^{(0)})\left[\prod_{i=1}^{k}(\alpha_i I_m + B)^{-1}(\beta_i I_m - B)\right] \\ &= s_k(A)X s_k(-B)^{-1}, \end{aligned}$$

kjer z s_k označimo racionalno funkcijo oblike

$$s_k(x) = \prod_{i=1}^{k}\frac{x - \alpha_i}{x + \beta_i}.$$

Po trikotniški neenakosti dobimo še oceno v poljubni matrični normi

$$\|X - X^{(k)}\| = \|s_k(A)Xs_k(-B)^{-1}\| \leq \|s_k(A)\|\|X\|\|s_k(-B)^{-1}\|, \tag{40}$$

oziroma oceno za relativno napako

$$\frac{\|X - X^{(k)}\|}{\|X\|} \leq \|s_k(A)\|\|s_k(-B)^{-1}\|. \tag{41}$$

Tipično se uporablja spektralna ali Frobeniusova matrična norma. Enačbi (40) in (41) pa nista pomembni samo zaradi ocene relativne napake, ampak nam omogočata tudi vpogled v izbiro parametrov $\{\alpha_k\}$ in $\{\beta_k\}$. Spomnimo se najprej Cayley-Hamiltonovega izreka, ki pravi, da za karakteristični polinom matrike A, $p(\lambda) = \det(A - \lambda I)$, velja

$$p(A) = 0.$$

Po Cayley-Hamiltonovemu izreku potem velja: če $\{\alpha_k\}_{k=1}^r$ vsebuje vse lastne vrednosti A in $\{\beta_k\}_{k=1}^r$ vsebuje vse lastne vrednosti matrike B (večkratne z njihovo algebraično večkratnostjo), tedaj je $X - X^{(r)} = 0$.
Iz enačbe (41) je razvidno, da optimalna izbira premikov reši minimizacijski problem

$$\min_{\substack{\alpha_1,\ldots\alpha_r\in\mathbb{C}\\ \beta_1,\ldots\beta_r\in\mathbb{C}}} \|s_r(A)\|.\|s_r(-B)^{-1}\|. \tag{42}$$

Posledično (42) pravimo **ADI minimaks problem**. Optimalna nabora $\{\alpha_k\}_{k=1}^r$, $\{\beta_k\}_{k=1}^r$ za dovolj velik r v obeh primerih vsebujeta lastne vrednosti A ter lastne vrednosti matrike B. Ker je problem iskanja lastnih vrednosti matrik A in B podobne zahtevnosti kot direktno reševanje Sylvestrove enačbe, tipično poskušamo poiskati aproksimacije za spektra matrik.
V splošnem je problem (42) zahteven in zaenkrat ne poznamo rešitve [2]. Za enačbo Ljapunova kot tudi Sylvestrovo enačbo, ko sta A in B hermitski, pa lahko poiščemo dobre aproksimacije za rešitev problema (42) s pomočjo eliptičnih integralov. To problematiko bomo podrobneje obravnavali v poglavju 4.
Zapišimo postopek še kot algoritem:

Algoritem 3: Metoda ADI za reševanje Sylvestrove enačbe

VHOD: Matrike A, B in C, urejena nabora premikov α_r in β_r, število korakov i
IZHOD: Približek X_i za rešitev Sylvestrove enačbe X

$X_1 = 0$
for $r = 1, 2, \ldots, i$
  reši $(\beta_r I_n + A)Y = C - X_r(-\beta_r I_m + B)$
  reši $X_{r+1}(B + \alpha_r I_m) = C - (A - \alpha_r I_n)Y$,
end

3.4. **Metoda LRCF-ADI**

Preden se lotimo naslednje razširitve si oglejmo motivacijo za njen nastanek. Ko imamo opravka z velikimi zveznimi LTI sistemi jim velikokrat poskušamo zmanjšati dimenzijo tako, da sta si reduciran sistem in originalen sistem čimbolj podobna. V uvodu smo povedali, da je popularen pristop k zmanjšanju dimenzije sistema uravnoteženo rezanje. Povedali smo tudi, da je problem redukcije modela povezan s križno Gramovo matriko. Izkaže se, da je križna Gramova matrika nesingularna

natanko tedaj, ko je par (A, B) vodljiv in par (A, C) spoznaven [4]. Če obstaja še matrika J tako, da je

$$\begin{aligned} AJ &= JA^T, \\ B &= JC^T, \end{aligned} \tag{43}$$

potem velja, da je prenosna funkcija $G(s)$ LTI sistema (1) simetrična. Pri teh pogojih za križno Gramovo matriko le-tega sistema velja

$$X_{co}^2 = X_c^2 X_o^2.$$

Za matrike A, B in C iz enačbe (11) takšna matrika J obstaja, če je A simetrična in $B = C^T$ [4]. Za lastne vrednosti matrike X_{co} posledično velja

$$|\sigma(X_{co})| = \sqrt{\sigma(X_c^2 X_0^2)},$$

kjer računska operacija na množici pomeni računsko operacijo na vsakem elementu množice. Vrednostim $|\sigma(X_{co})|$ pravimo **Hanklove singularne vrednosti**. Po privzetkih, ko je A stabilna in $Q = BC = BB^T$, je X_{co} simetrična pozitivno definitna in lahko absolutne vrednosti izpustimo.
Privzemimo, da sta para (A, B) in (A, C) vodljiva in spoznavna. Uravnoteženo rezanje sistema (1) poteka po naslednjem postopku:

- Izračunata se vodljivostna Gramova matrika X_c in spoznavnostna Gramova matrika X_o za enačbi Ljapunova, ki pripadata zveznemu LTI sistemu ter njuna razcepa Choleskega $X_c = Z_c Z_c^T$ in $X_o = Z_o Z_o^T$.
- Izračuna se singularni razcep $U\Sigma V^T = Z_c^T Z_o$,

  $$\Sigma = \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix},$$

  kjer so singularne vrednosti Σ urejene nenaraščajoče in so singularne vrednosti Σ_2 manjše od vnaprej določene tolerančne meje.
- Naj bo $T = \Sigma^{1/2} U^H Z_c^{-1} = \Sigma^{-1/2} V^T Z_o^T$, potem je $T^{-1} = Z_c (U^H)^{-1} \Sigma^{-1/2} = (Z_o^T)^{-1} (V^{-T})^{-1} \Sigma^{1/2}$. Označimo $\tilde{x} = Tx$ in naslednje matrike razdelimo bločno skladno z matriko Σ

  $$TAT^{-1} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \; Tx = \begin{bmatrix} \tilde{x} \\ \tilde{x}_k \end{bmatrix}, \; TB = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \; CT^{-1} = \begin{bmatrix} C_1 & C_2 \end{bmatrix}.$$

- Dobimo nov LTI sistem manjše dimenzije

  $$\frac{d}{dt}\tilde{x}(t) = A_{11}\tilde{x}(t) + B_1 u(t), \; y = C_1 \tilde{x}(t).$$

Ime *uravnoteženo* pride iz lastnosti, da sta za reduciran sistem obe, vodljivostna in spoznavnostna Gramova matrika, diagonalni in enaki Σ [8, 12], tj. sistem zmanjša uravnoteženo glede na spoznavnost in vodljivost sistema.
Metoda je privlačna, ker velja, da je matrika A_{11} stabilna, če je bila A stabilna in ker za prenosno funkcijo originalnega sistema $G(s)$ in prenosno funkcijo reduciranega sistema, ki jo označimo z $\tilde{G}(s)$, pri pogoju $\sigma_k > \sigma_{k+1}$, velja

$$\|G - \tilde{G}\|_{\mathcal{H}_\infty} = \sup_{s \in \mathbb{R}} \|G(is) - \tilde{G}(is)\|_2 \leq 2 \sum_{r=k+1}^{n} \sigma_r,$$

kjer so $\{\sigma_i\}_{i=1}^n$ nenaraščajoče urejene Hanklove vrednosti originalnega sistema, i imaginarna enota, k je rang matrike Σ_1 in z $\|.\|_2$ označimo spektralno normo [8],

[12]. Dokaz ocene najdemo v [8], več informacij o redukciji velikih zveznih LTI sistemov najdemo v [12].

Privzemimo torej, da imamo opravka z enačbo Ljapunova (6), kjer je $B = A^T$ in $Q = GG^T$ oziroma $B = A^H$ in $Q = GG^H$, če sta A in G kompleksni. Q dobimo v takšni obliki na primer, ko rešujemo enačbi za spoznavnostno Gramovo matriko ali vodljivostno Gramovo matriko.
V tem primeru je dovolj vzeti samo en nabor parametrov $\{\alpha_r\}_{r=1}^{i}$, za premike β_r vzamemo konjugirane vrednosti α_r, tj. $\beta_r = \overline{\alpha_r}$, $r = 1, 2, \ldots, i$. Postopke bomo zapisali za splošen primer, ko A ni nujno simetrična. Če je A simetrična, potem v naslednjih izpeljavah velja $\overline{\alpha_r} = \alpha_r$. Privzemimo, da je rešitev X nizkega ranga in jo lahko zapišemo kot $X = ZZ^H$, kjer je posledično tudi faktor Z nizkega ranga. To lahko privzamemo, če je desna stran enačbe Ljapunova nizkega ranga, saj v tem primeru razmerje $\frac{\lambda_r}{\lambda_1}$, kjer sta λ_1 po absolutni vrednosti največja lastna vrednost rešitve enačbe Ljapunova in λ_r r-ta največja lastna vrednost rešitve, hitro pada proti 0. Dejstvi sledita iz ocen pri reševanju ADI minimaks problema, ki si jih bomo ogledali v poglavju 4.
Videli bomo, da lahko bistveno prihranimo pri prostorski in časovni zahtevnosti. Končno iteracijo metode ADI (33) lahko pretvorjeno za enačbo Ljapunova zapišemo v obliki

$$\begin{aligned} Z_r Z_r^H =& (\overline{\alpha_r} I_n + A)^{-1}(\alpha_r I_n - A) Z_{r-1} Z_{r-1}^H (\alpha_r I_n + A^H)^{-1}(\overline{\alpha_r} I_n - A^H) \\ & - 2\mathbf{Re}(\alpha_r)(\overline{\alpha_r} I_n + A)^{-1} G G^H (\alpha_r I_n + A^H)^{-1}. \end{aligned} \tag{44}$$

Sedaj lahko zapišemo rekurzivno enačbo za faktor Z_r za $r = 1, 2, \ldots, i$

$$Z_r = \left[\sqrt{-2\mathbf{Re}(\alpha_r)}(\overline{\alpha_r} I_n + A)^{-1} G \quad (\overline{\alpha_r} I_n + A)^{-1}(\alpha_r I_n - A) Z_{r-1} \right], \tag{45}$$

z začetno vrednostjo

$$Z_1 = \sqrt{-2\mathbf{Re}(\alpha_r)}(\overline{\alpha_r} I_n + A)^{-1} G.$$

Pri tem smo upoštevali, da lahko zamenjamo vrstni red inverza in hermitskega operatorja.
Če v enačbi (45) obrnemo vrstni red premikov $\tilde{\alpha_r} = \alpha_{i-(r-1)}$ za $r = 1, 2, \ldots, i$, lahko nato sestavimo novo metodo

$$Z_r = \left[Z_{r-1} \quad Z^{(r)} \right], \tag{46}$$

kjer $Z^{(r)}$ računamo kot

$$\begin{aligned} Z^{(r)} &= \sqrt{\frac{\mathbf{Re}(\tilde{\alpha}_r)}{\mathbf{Re}(\tilde{\alpha}_{r-1})}} (\tilde{\alpha}_r I_n + A)^{-1}(\overline{\tilde{\alpha}_{r-1}} I_n - A) Z^{(r-1)} \\ &= \sqrt{\frac{\mathbf{Re}(\tilde{\alpha}_r)}{\mathbf{Re}(\tilde{\alpha}_{r-1})}} ((\tilde{\alpha}_r + \overline{\tilde{\alpha}_{r-1}})(\tilde{\alpha}_r I_n + A)^{-1} - I_n) Z^{(r-1)}, \end{aligned} \tag{47}$$

z začetnim približkom

$$Z_1 = Z^{(1)} = \sqrt{-2\mathbf{Re}(\tilde{\alpha_1})}(\tilde{\alpha_1} I_n + A)^{-1} G.$$

Posledično lahko Z_r zapišemo kot

$$Z_r = \left[Z^{(1)} \quad Z^{(2)} Z^{(1)} \quad Z^{(3)} Z^{(2)} Z^{(1)} \quad \ldots \quad Z^{(r)} Z^{(r-1)} \ldots Z^{(2)} Z^{(1)} \right].$$

Prvotno je obliko (45) razvil Penzl [11] v metodi low rank ADI, dodatek (47), ki zagotavlja, da na vsaki iteraciji spreminjamo blok fiksne dimenzije, pa je bil objavljen

neodvisno v [7], kjer je bila razvita metoda Cholesky factored ADI. Posledično se za metodo uporablja skupno ime **metoda ADI nizkega ranga s faktorji Choleskega** (ang. *Low rank Cholesky factored ADI method*), faktorju Z pa rečemo **faktor Choleskega nizkega ranga** (ang. *Low rank Cholesky factor*). Za metodo se uporablja kratica **LRCF-ADI**.

Zapišimo še postopek v algoritmični obliki kot v [9]:

Algoritem 4: LRCF-ADI metoda za reševanje enačbe Ljapunova

VHOD: Matriki A in G, nabor urejenih premikov $\{\alpha_r\}_{r=1}^{i}$
IZHOD: Približek za matriko Z, da je $ZZ^H \approx X$

$V_1 = \sqrt{-2\mathbf{Re}(\alpha_1)}(\overline{\alpha_1}I_n + A)^{-1}G$
$Z_1 = V_1$
for $r = 1, 2, \ldots, i$
$\quad V_r = \sqrt{\mathbf{Re}(\alpha_r)/\mathbf{Re}(\alpha_{r-1})}((\alpha_r + \overline{\alpha_{r-1}})(\overline{\alpha_r}I_n + A)^{-1}V_{r-1} - V_{r-1})$
$\quad Z_r = [Z_{r-1}\ V_r]$
end for

Pri uravnoteženem rezanju smo videli, da ne potrebujemo celotne rešitve enačbe Ljapunova, ampak je dovolj, da poznamo zgornjo trikotno matriko Z. Metoda LRCF-ADI nam omogoča, da direktno izračunamo aproksimacijo matrike Z, ki je tipično uporabnejša kot približek rešitve X. Dodatno lahko prihranimo pri prostorski in časovni zahtevnosti, če metoda konvergira hitro, tj. če lahko najdemo tako aproksimacijo Z z malo stolpci, da je ZZ^H dober približek za X. To se zgodi pogosto, ko je matrika Q nizkega ranga [10]. V nasprotnem primeru, ko metoda ne konvergira hitro, število stolpcev Z_r na vsaki iteraciji hitro narašča, kar bistveno poveča količino spomina, ki ga potrebujemo za shranjevanje matrik.
Razpišimo (44) še en korak nazaj.

(48)
$$\begin{aligned} Z_r Z_r^H =& (\overline{\alpha_r}I_n + A)^{-1}(\alpha_r I_n - A) \\ & \Big[(\overline{\alpha_{r-1}}I_n + A)^{-1}(\alpha_{r-1}I_n - A)Z_{r-2}Z_{r-2}^H(\alpha_{r-1}I_n + A^H)^{-1} \\ & \cdot (\overline{\alpha_{r-1}}I_n - A^H) - 2\mathbf{Re}(\alpha_{r-1})(\overline{\alpha_{r-1}}I_n + A)^{-1}GG^H(\alpha_{r-1}I_n + A^H)^{-1}\Big] \\ & \cdot (\alpha_r I_n + A^H)^{-1}(\overline{\alpha_r}I_n - A^H) - 2\mathbf{Re}(\alpha_r)(\overline{\alpha_r}I_n + A)^{-1}GG^H(\alpha_r I_n + A^H)^{-1}. \end{aligned}$$

Enačbo preuredimo tako, da dobimo

(49)
$$\begin{aligned} Z_r Z_r^H =& (\overline{\alpha_r}I_n + A)^{-1}(\overline{\alpha_{r-1}}I_n + A)^{-1}\Big[(\alpha_r I_n - A)(\alpha_{r-1}I_n - A)Z_{r-2}Z_{r-2}^H \\ & (\alpha_r I_n - A)^H(\alpha_{r-1}I_n - A)^H - 2(\alpha_r + \alpha_{r-1})(AGG^HA^H + \alpha_r\alpha_{r-1}GG^H)\Big] \\ & (\overline{\alpha_r}I_n + A)^{-H}(\overline{\alpha_{r-1}}I_n + A)^{-H}. \end{aligned}$$

Iz enačbe (49) lahko vidimo naslednje. Če vzamemo za zaporedna premika kompleksni par, tj. $\alpha_r = \overline{\alpha_{r-1}}$, sledi

$$(\alpha_{r-1}I_n + A)(\overline{\alpha_{r-1}}I_n + A) = |\alpha_r|I_n + 2\mathbf{Re}(\alpha_{r-1})A + A^2$$

in

$$(\alpha_{r-1}I_n - A)(\overline{\alpha_{r-1}}I_n - A) = |\alpha_r|I_n - 2\mathbf{Re}(\alpha_{r-1})A + A^2.$$

Če upoštevamo, da je inverz realne matrike realna matrika, lahko tako zagotovimo, da so faktorji Z_r realne matrike [7].

Slabost tega postopka je izračun $(|\alpha_r|I_n + \mathbf{Re}(\alpha_r)I_n + A^2)$ in $(|\alpha_r|I_n - \mathbf{Re}(\alpha_r)I_n + A^2)$, saj računamo inverze prejšnjih izrazov in A^2, ki zgostijo matrike.

3.5. **Faktorizirana metoda ADI**

Velikokrat se zgodi, da lahko matriko C iz Sylvestrove enačbe zapišemo kot $C = GF^H$, $G \in \mathbb{R}^{n\times p}, F \in \mathbb{R}^{m\times p}$, $p \ll \min\{n, m\}$. Prej smo že omenili, da lahko kot poseben primer matriko C dobimo v takšni obliki, če iščemo spoznavostno, vodljivostno ali križno Gramovo matriko.
Privzemimo, da lahko rešitev Sylvestrove enačbe zapišemo kot $X_k = Z_k D_k Y_k^H$. Potem lahko ADI iteracijsko enačbo (30) iz oblike (33) zapišemo v matrični obliki

$$\begin{aligned} X_r = &\begin{bmatrix}(\beta_r I_n + A)^{-1}G & (\beta_r I_n + A)^{-1}(\alpha_r I_n - A)Z_{r-1}\end{bmatrix} \\ &\cdot \begin{bmatrix}(\alpha_r + \beta_r)I_p & \\ & D_{r-1}\end{bmatrix} \\ &\cdot \begin{bmatrix}F^H(\alpha_r I_m + B)^{-1} \\ Y_{r-1}^H(\alpha_r I_m + B)^{-1}(\beta_r I_m - B)\end{bmatrix} \\ &= Z_r D_r Y_r^H, \end{aligned} \tag{50}$$

$r = 1, 2, \ldots, i$. Tako dobimo rešitev X_r v faktorizirani obliki $X_r = Z_r D_r Y_r^H$, kjer so dimenzije matrik po vrsti $n \times p, p \times p$ in $p \times m$. Če uporabimo lemo 3.3, potem lahko posamezne faktorje računamo iterativno po formulah

$$\begin{aligned} Z_r &= \begin{bmatrix}(\beta_r I_n + A)^{-1}G & (\alpha_r + \beta_r)(\beta_r I_n + A)^{-1}Z_{r-1} - Z_{r-1}\end{bmatrix}, \\ D_r &= \begin{bmatrix}(\alpha_r + \beta_r)I_p & \\ & D_{r-1}\end{bmatrix}, \\ Y_r^H &= \begin{bmatrix}F^H(\alpha_r I_m + B)^{-1} \\ (\alpha_r + \beta_r)Y_{r-1}^H(\alpha_r I_m + B)^{-1} - Y_{r-1}^H\end{bmatrix}. \end{aligned} \tag{51}$$

Za začetni vrednosti Z_0 in Y_0^H lahko vzamemo kar ničelni matriki primernih velikosti. Ker rešitev dobimo v faktorizirani obliki, pravimo metodi **faktorizirana ADI iteracija**. Uporablja se oznaka **fADI** [2].
Prednost metode je, da lahko vse tri sisteme računamo paralelno, kar zmanjša količino časa, ki ga računalnik potrebuje za reševanje Sylvestrove enačbe. Velikokrat se zgodi tudi, da je faktorizirana rešitev priročnejša za nadaljnjo analizo.
Zapišimo postopek še v psevdokodi kot v [2]:

Algoritem 5: Faktorizirana metoda za reševanje Sylvestrove enačbe

---

VHOD: Matrike A, B, G in F, nabor urejenih premikov α_r ter število korakov i
IZHOD: Matrike Z, D in Y, tako da ZDY^H aproksimira X

$Z_1 = (\beta_1 I_n + A)^{-1} G$
$Y_1^H = F^H(\alpha_1 I_m + B)^{-1}$
for $r = 2, \ldots, i$
$\quad Z_r = (\alpha_r + \beta_r)(\beta_r I_n + A)^{-1} Z_{r-1} - Z_{r-1}$
$\quad Y_r^H = (\alpha_r + \beta_r) Y_{r-1}^H (\alpha_r I_m + B)^{-1} - Y_{r-1}^H$
end for
$D = \mathrm{diag}((\alpha_1 + \beta_1) I_p, (\alpha_2 + \beta_2) I_p, \ldots, (\alpha_i + \beta_i) I_p)$

3.6. **Modificirana metoda fADI**

Za dva dana nabora parametrov $\{\alpha_i\}$ in $\{\beta_i\}$ metoda fADI reši enačbo (5). Če razpišemo Z_r iz enačbe (50) še en korak nazaj, dobimo

$$Z_r = \begin{bmatrix} (\beta_r I_n + A)^{-1} G \\ (\beta_r I_n + A)^{-1}(\alpha_r I_n - A)(\beta_{r-1} I_n + A)^{-1} G \\ (\beta_r I_n + A)^{-1}(\alpha_r I_n - A)(\beta_{r-1} I_n + A)^{-1}(\alpha_{r-1} I_n - A) Z_{r-2} \end{bmatrix}^T.$$

Analogno lahko naredimo tudi s faktorji Y_r^H

$$Y_r^H = \begin{bmatrix} F^H(\alpha_r I_m + B)^{-1} \\ F^H(\alpha_{r-1} I_m + B)^{-1}(\alpha_r I_m + B)^{-1}(\beta_r I_m - B) \\ Y_{r-2}^H(\alpha_{r-1} I_m + B)^{-1}(\beta_{r-1} I_m - B)(\alpha_r I_m + B)^{-1}(\beta_r I_m - B) \end{bmatrix},$$

matrika D_r pa je oblike

$$D_r = \begin{bmatrix} (\alpha_r + \beta_r) I_p & & \\ & (\alpha_{r-1} + \beta_{r-1}) I_p & \\ & & D_{r-2} \end{bmatrix}.$$

Iz tega je neposredno razvidno, da lahko podobno kot pri metodi LRCF-ADI matriko Z_r zapišemo kot

$$Z_r = \begin{bmatrix} Z^{(1)} & Z^{(2)} & Z^{(3)} & \ldots & Z^{(r)} \end{bmatrix},$$

kjer je začetna matrika

$$Z^{(1)} = (\beta_r I_n + A)^{-1} G,$$

preostale $Z^{(i)}$ izračunamo s pomočjo enačb (faktorje smo preuredili po lemi 3.2)

$$\begin{aligned} Z^{(i+1)} &= (\beta_{r-i} I_n + A)^{-1}(\alpha_{r-(i-1)} I_n - A) Z^{(i)} \\ &= (\beta_{r-i} + \alpha_{r-(i-1)})(\beta_{r-i} I_n + A)^{-1} Z^{(i)} - Z^{(i)},\ i = 1, 2, \ldots, r. \end{aligned} \tag{52}$$

Analogno velja za faktor Y_r^H, za katerega dobimo

$$Y_r = \begin{bmatrix} Y^{(1)} & Y^{(2)} & \ldots & Y^{(r)} \end{bmatrix},$$

kjer je

$$Y^{(1)H} = F^H(\alpha_r I_m + B)^{-1}$$

in

$$\begin{aligned} Y^{(i+1)H} &= Y^{(i)H}(\alpha_{r-i} I_m + B)^{-1}(\beta_{r-(i-1)} I_m - B) \\ &= (\beta_{r-(i-1)} + \alpha_{r-i}) Y^{(i)H}(\alpha_{r-i} I_m + B)^{-1} - Y^{(i)H},\ i = 1, 2, \ldots, r. \end{aligned} \tag{53}$$

Za konsistentnost preimenujmo indekse premikov v $k = r - i$, kot pri metodi LRCF-ADI, tako da lahko zapišemo postopek bolj kompaktno v obliki:

$$Z_k = \begin{bmatrix} Z^{(1)} & Z^{(2)} & \dots & Z^{(k)} \end{bmatrix}$$

za $k = 1, 2, \dots, r$, kjer je

$$\begin{aligned} Z^{(1)} &= (\beta_1 I_n + A)^{-1} G, \\ Z^{(k+1)} &= (\beta_{k+1} I_n + A)^{-1} (\alpha_k I_n - A) Z^{(k)} \\ &= (\beta_{k+1} + \alpha_k)(\beta_{k+1} I_n + A)^{-1} - Z^{(k)} \end{aligned} \tag{54}$$

in

$$Y_k = \begin{bmatrix} Y^{(1)} & Y^{(2)} & \dots & Y^{(k)} \end{bmatrix},$$

kjer je

$$\begin{aligned} Y^{(1)H} &= F^H (\alpha_1 I_m + B)^{-1}, \\ Y^{(k+1)} &= Y^{(k)H} (\alpha_{k+1} I_m + B)^{-1} (\beta_k I_m - B) \\ &= (\beta_k + \alpha_{k+1}) Y^{(k)H} (\alpha_{k+1} I_m + B)^{-1} - Y^{(k)H} \end{aligned} \tag{55}$$

in

$$D_{k+1} = \begin{bmatrix} (\alpha_1 + \beta_1) I_p & 0 & \cdots & 0 \\ 0 & (\alpha_2 + \beta_2) I_p & & \vdots \\ \vdots & & \ddots & \\ 0 & \cdots & & (\alpha_k + \beta_k) I_p \end{bmatrix} \tag{56}$$

ter

$$X_{k+1} = Z_{k+1} D_{k+1} Y_{k+1}^H. \tag{57}$$

Enačbe (54-57) predstavljajo novo različico metode fADI, ki so jo razvili Brenner, Li in Truhar [2]. Modifikacija metode fADI prinaša dve prednosti pred metodo fADI:

(1) Ker velikost Z_k narašča linearno se na vsakem koraku tudi povečuje število izračunov, saj na vsakem koraku metode fADI rešujemo večji sistem linearnih enačb, medtem ko na vsakem koraku modificirane metode fADI rešujemo blok fiksne velikosti. Analogno velja tudi za faktorja Y_k^H in $(Y^{(k)})^H$.

(2) Na vsakem koraku modifikacije metode fADI izračunamo novi matriki $Z^{(k)}$ in $Y^{(k)}$ in ju dodamo Z_k ter Y_k, medtem ko se pri metodi fADI na vsaki iteraciji spreminjata celotni matriki Z_k in Y_k. To tudi izboljša numerično stabilnost, saj lahko na Z_k in Y_k izvedemo ortonormalizacijo, vendar se s tem poveča čas, potreben za izračun aproksimacije X.

Poglejmo še, kako je z enačbo Ljapunova.
V oznakah metode fADI za enačbo Ljapunova velja $Y^{(k)} = Z^{(k)}$, zato računamo rešitev enačbe X na naslednji način:

$$Z_k = \begin{bmatrix} Z^{(1)} & Z^{(2)} & Z^{(3)} & \dots & Z^{(k)} \end{bmatrix},$$

kjer je

$$Z^{(1)} = (A + \overline{\alpha_1} I_n)^{-1} G,$$

preostale člene pa iterativno računamo kot

$$\begin{aligned} Z^{(k+1)} &= (\overline{\alpha_{k+1}} I_n + A)^{-1}(\alpha_k I_n - A)Z^{(k)} \\ &= Z^{(k)} - (\overline{\alpha_{k+1}} + \alpha_k)(A + \alpha_k I_m)^{-1} Z^{(k)}. \end{aligned} \tag{58}$$

Označimo še diagonalno matriko D_k

$$D_k = \begin{bmatrix} \mathbf{Re}(\alpha_1) I_p & & \\ & \ddots & \\ & & \mathbf{Re}(\alpha_k) I_p \end{bmatrix}.$$

Potem lahko k-ti približek rešitve ponovno računamo kot

$$X_k = Z_k D_k Z_k^H.$$

Iz tega je neposredno razvidno, da je za enačbo Ljapunova modifikacija metode fADI v bistvu enaka LRCF-ADI, samo da je pri metodi LRCF-ADI diagonalna matrika zapisana v faktorijh Z. Torej lahko na modifikacijo metode fADI gledamo kot na razširitev metode LRCF-ADI iz enačbe Ljapunova na Sylvestrovo enačbo. Kot tudi v prejšnjih primerih so premiki bistveni za hitrost konvergence metode fADI.

3.7. Metoda podprostorov ADI

Za dane premike α_r in β_r definiramo **r-ti ADI stolpični podprostor** kot prostor, ki ga razpenjajo stolpci $X_r = Z_r D_r Y_r$ oziroma ekvivalentno prostor, ki ga razpenjajo stolpci Z_r in **r-ti ADI vrstični podprostor** kot prostor, ki ga razpenjajo vrstice X_r^H oziroma ekvivalentno prostor, ki ga razpenjajo vrstice Y_r^H.
Namesto približka X_r iz metode fADI uporabimo približek

$$\tilde{X}_r = U_r W_r V_r^H.$$

Stolpci U_r razpenjajo isti prostor kot stolpci Z_r ter V_r^H isti vrstični prostor kot Y_r^H. Privzemimo, da imata U_r in V_r ortonormirane stolpce (sicer lahko to dosežemo z Gram-Schmidtovim postopkom).
Ostanek pri aproksimaciji $\tilde{X}_r$ označimo z

$$R_r = A\tilde{X}_r + \tilde{X}_r B - C. \tag{59}$$

Za minimizacijo ostanka imamo na voljo več različnih pogojev, npr. Galerkinov pogoj, minimizacija ostanka v Frobeniusovi normi, dobre približke za U_r in V_r lahko poiščemo tudi s podprostori Krilova. Metodi rečemo **metoda podprostorov ADI** (ang. *projection ADI subspace method*) v skladu z literaturo [2].
Prednost metode je predvsem to, da lahko vzamemo slabše premike in nato dobljene približke izboljšamo z izbiro stolpičnih in vrstičnih ADI podprostorov.
V nadaljevanju si bomo pogledali primer minimizacije za Galerkinov pogoj.
Za Galerkinov pogoj dobimo

$$U_r^H R_r V_r = 0. \tag{60}$$

Če torej pomnožimo (59) z U_r^H z leve ter V_r z desne in upoštevamo (60), dobimo

$$(U_r^H A U_r) W_r - W_r (V_r^H B V_r) = U_r^H C V_r. \tag{61}$$

Enačba (61) je še vedno Sylvestrova enačba, vendar bistveno manjše velikosti. Velikokrat so dobljene matrike goste. Tako jo lahko rešimo s pomočjo direktnih metod, npr. z Bartels-Stewartovim algoritmom.

4. **Izbira premikov**

V tem poglavju bomo pogledali, kako poiskati premike za metode ADI iz prejšnjega poglavja. Ker je izbira premikov neposredno povezana z oceno napake med r-to iteracijo in dejansko rešitvijo Sylvestrove enačbe (5) oziroma enačbe Ljapunova (6), bomo v tem poglavju hkrati pogledali tudi različne ocene za napake.

Če sta matriki A in B hermitski in za normo vzamemo operatorsko 2-normo, potem je (42) ekvivalenten

$$\begin{aligned}&\min_{\substack{\alpha_1,\ldots,\alpha_r\in\mathbb{C}\\ \beta_1,\ldots,\beta_r\in\mathbb{C}}} \max_{\substack{x\in\sigma(A)\\ y\in\sigma(B)}} \prod_{j=1}^{r}\left|\frac{(\alpha_j-x)(\beta_j-y)}{(\beta_j+x)(\alpha_j+y)}\right| \\ &= \min_{\substack{\alpha_1,\ldots,\alpha_r\in\mathbb{C}\\ \beta_1,\ldots,\beta_r\in\mathbb{C}}} \max_{\substack{x\in\sigma(A)\\ y\in\sigma(B)}} \prod_{j=1}^{r}\left|\frac{(x-\alpha_j)(y-\beta_j)}{(\beta_j+x)(\alpha_j+y)}\right|.\end{aligned} \tag{62}$$

Prepričajmo se, da to res drži. Po predpostavki sta A in B simetrični oziroma hermitski, torej ju lahko diagonaliziramo. Naj bosta prehodni matriki P_1 za A in P_2 za B unitarni. Velja

$$\begin{aligned}&\min_{\substack{\alpha_1,\ldots,\alpha_r\in\mathbb{C}\\ \beta_1,\ldots,\beta_r\in\mathbb{C}}} \|s_r(A)\|_2.\|s_r(-B)^{-1}\|_2 \\ &= \min_{\substack{\alpha_1,\ldots,\alpha_r\in\mathbb{C}\\ \beta_1,\ldots,\beta_r\in\mathbb{C}}} \|s_r(P_1D_1P_1^{-1})\|_2.\|s_r(P_2D_2P_2^{-1})^{-1}\|_2 \\ &= \min_{\substack{\alpha_1,\ldots,\alpha_r\in\mathbb{C}\\ \beta_1,\ldots,\beta_r\in\mathbb{C}}} \|P_1s_r(D_1)P_1^{-1}\|_2.\|P_2s_r(D_2)^{-1}P_2^{-1}\|_2 \\ &= \min_{\substack{\alpha_1,\ldots,\alpha_r\in\mathbb{C}\\ \beta_1,\ldots,\beta_r\in\mathbb{C}}} \left(\max_{x\in\sigma(D_1)}|s_r(x)|\max_{y\in\sigma(D_2)}|s_r(y)^{-1}|\right) \\ &= \min_{\substack{\alpha_1,\ldots,\alpha_r\in\mathbb{C}\\ \beta_1,\ldots,\beta_r\in\mathbb{C}}} \left(\max_{x\in\sigma(A)}|s_r(x)|\max_{y\in\sigma(B)}|s_r(y)^{-1}|\right) \\ &= \min_{\substack{\alpha_1,\ldots,\alpha_r\in\mathbb{C}\\ \beta_1,\ldots,\beta_r\in\mathbb{C}}} \max_{\substack{x\in\sigma(A)\\ y\in\sigma(B)}} \left|\frac{s_r(x)}{s_r(y)}\right|.\end{aligned} \tag{63}$$

Iz izpeljave same opazimo, da je dovolj, če namesto pogoja, da sta A in B hermitski matriki, vzamemo pogoj, da sta A in B normalni matriki (po karakterizaciji sta tedaj unitarno podobni diagonalni matriki).
Če pri enačbi Ljapunova (6) dopuščamo kompleksne normalne matrike, ki niso nujno hermitske, se enačba (62) poenostavi v

$$\min_{\alpha_1,\ldots\alpha_k\in\mathbb{C}} \max_{x\in\sigma(A)} \prod_{j=1}^{r}\left|\frac{x-\alpha_j}{\overline{\alpha_j}+x}\right|, \tag{64}$$

sicer, če je A hermitska matrika, dobimo

$$\min_{\alpha_1,\dots\alpha_k\in\mathbb{C}} \max_{x\in\sigma(A)} \prod_{j=1}^{r} \left|\frac{x-\alpha_j}{\alpha_j+x}\right|. \tag{65}$$

Iz (64) in (65) potem sledi, da je pri reševanju enačbe Ljapunova s katerokoli izmed metod ADI iz prejšnjega poglavja dovolj samo en nabor parametrov.

Za hitrost konvergence je bistvenega pomena izbira premikov α_r in β_r. Če je število iteracij $r \geq m, n$, potem optimalni premiki vsebujejo kar lastne vrednosti A in B, vendar teh navadno ne poznamo. Zato namesto problema (62) rešujemo

$$\min_{\substack{\alpha_1,\dots\alpha_r\in\mathbb{C}\\ \beta_1,\dots\beta_r\in\mathbb{C}}} \max_{\substack{x\in E\\ y\in F}} \prod_{j=1}^{r} \left|\frac{(\alpha_j-x)(\beta_j-y)}{(\beta_j+x)(\alpha_j+y)}\right| = \min_{\substack{\alpha_1,\dots\alpha_r\in\mathbb{C}\\ \beta_1,\dots\beta_r\in\mathbb{C}}} \max_{\substack{x\in E\\ y\in F}} \prod_{j=1}^{r} \left|\frac{(x-\alpha_j)(y-\beta_j)}{(\beta_j+x)(\alpha_j+y)}\right|, \tag{66}$$

kjer sta E in F (kompaktni) intervalski oceni spektrov matrik A in B. Za izračun ocen premikov bomo potrebovali meje za E in F. Slednje skladno z literaturo [2] tipično ocenimo z Arnoldijevim iteracijskim algoritmom oziroma Lanczosevo metodo, ko rešujemo enačbo Ljapunova, če je A hermitska matrika.

4.1. **Penzlova ocena**

Penzl [10] je omejil singularne vrednosti rešitve Ljapunove enačbe, če je A simetrična (hermitska za kompleksno matriko A) matrika z lastnimi vrednostmi v $\mathbb{R}_-$ in $Q = GG^T$. Tedaj je Q simetrična pozitivno semidefinitna matrika.
Uredimo lastne vrednosti rešitve Ljapunove enačbe po velikosti nenaraščujoče z

$$\lambda_1(X) \geq \lambda_2(X) \geq \cdots \geq \lambda_n(X). \tag{67}$$

Penzl se je ukvarjal z zgornjo mejo za izraz

$$\frac{\lambda_{r+1}(X)}{\lambda_1(X)} \tag{68}$$

za $r = 1, 2, \dots, i-1$.
Preden nadaljujemo potrebujemo naslednji izrek.

Izrek 4.1. *(Eckart–Young–Mirsky) Naj bo $A \in \mathbb{R}^{n\times m}$ in naj bo*

$$A = U\Sigma V^T$$

njen singularni razcep, kjer so singularne vrednosti na diagonali razporejene nenaraščujoče (67). Razdelimo matrike U, Σ in V na naslednji način

$$U = \begin{bmatrix} U_1 & U_2 \end{bmatrix},\ \Sigma = \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix},\ V = \begin{bmatrix} V_1 & V_2 \end{bmatrix}, \tag{69}$$

kjer so $\Sigma_1 \in \mathbb{R}^{r\times r}$, $U \in \mathbb{R}^{m\times r}$ in $V \in \mathbb{R}^{n\times r}$. Potem je

$$A_r = U_1\Sigma_1 V_1^T$$

matrika ranga r takšna, da velja

$$\min_{rang(\tilde{A}_r)\leq r} \|A - \tilde{A}_r\|_2 = \|A - A_r\|_2 = \sigma_{r+1}(A),$$

$\sigma_{r+1}(A)$ pa označuje $(r+1)$-vo največjo singularno vrednost matrike A.

Dokaz.
Trdimo, da je najboljša aproksimacija problema $\min_{\text{rang}(\tilde{A}_r)\leq r} \|A - \tilde{A}_r\|_2$

$$A_r = \sum_{k=1}^{r} \sigma_k(A) u_k v_k^T,$$

kjer sta u_k in v_k k-ta stolpca matriki U in V.
Hitro opazimo, da velja

$$\|A - A_r\|_2 = \| \sum_{k=r+1}^{n} \sigma_k(A) u_k v_k^T \|_2 = \sigma_{r+1}(A).$$

Dovolj je dokazati, če $B_r = XY^T$, kjer imata X in Y r stolpcev, potem za poljubna tako izbrana X in Y velja

$$\|A - A_r\|_2 = \sigma_{r+1}(A) \leq \|A - B_r\|_2.$$

Ker ima Y r stolpcev mora obstajati linearna kombinacija $r+1$ stolpcev matrike V, označimo jo z $v = a_1 v_1 + \cdots + a_{r+1} v_{r+1}$ tako, da je $Y^T v = 0$. Vektor v lahko skaliramo, zato lahko privzamemo, da je $\|v\|_2 = 1$ oziroma $\alpha_1^2 + \cdots + \alpha_{r+1}^2 = 1$. Sledi

$$\|A - B_r\|_2^2 \geq \|(A - B_r)v\|_2^2 = \|Av\|_2^2 = a_1^2\sigma_1^2(A) + \cdots + a_{r+1}^2\sigma_{r+1}^2(A) \geq \sigma_{r+1}^2(A).$$

Korenimo obe strani neenakosti in dobimo željeni rezultat.

□

Meja iz izreka 4.1 je zelo tesno povezana s problemom najboljše aproksimacije matrike X nizkega ranga, saj po izreku 4.1 velja

$$\min_{\tilde{X}\in\mathbb{R}^{n\times n},\ \text{rang}(\tilde{X})\leq r} \frac{\|X - \tilde{X}\|_2}{\|X\|_2} = \frac{\lambda_{r+1}(X)}{\lambda_1(X)}, \tag{70}$$

kjer $\|.\|_2$ označuje spektralno normo kot prej. Za dane predpostavke, ko je A stabilna matrika in Q simetrična pozitivno semidefinitna singularne vrednosti X sovpadajo z lastnimi vrednostmi X. Rezultat bomo potrebovali pri naslednjem izreku [10].

Izrek 4.2. *(Penzl) Naj bo $A \in \mathbb{R}^{n\times n}$ simetrična negativno definitna matrika s pogojenostnim številom $\kappa = \kappa(A) = \|A^{-1}\|_2\|A\|_2$, naj bo matrika $B \in \mathbb{R}^{n\times m}$ neničelna matrika ter $\lambda_r(X)$ nenaraščajoče urejene lastne vrednosti, kot v (67), rešitve zvezne enačbe Ljapunova X za $r = 1, 2, \ldots, n$. Potem velja ocena*

$$\frac{\lambda_{mr+1}(X)}{\lambda_1(X)} \leq \left(\prod_{j=0}^{r-1} \frac{\kappa^{(2j+1)/(2r)} - 1}{\kappa^{(2j+1)/(2r)} + 1} \right)^2, \tag{71}$$

za $1 \leq mr \leq n$.

Dokaz.
Naj bo $r > 1$ poljubno izbran, fiksen, tako, da je $mr < n$. Definiramo racionalno funkcijo

$$f_p(t) = \frac{p - t}{p + t} \tag{72}$$

in produkt

$$f_{p_1,\ldots,p_r}(t) = \prod_{k=1}^{r} f_{p_i}(t), \tag{73}$$

za parametre $p, p_1, \ldots, p_r, t < 0$. Označimo še zaporedje ADI približkov $(X_r)_{r=1}^{\infty}$, dobljenih z začetno matriko X_0 in predpisom

$$X_r = f_{p_r}(A)X_{r-1}f_{p_r}(A) - 2p_r(A - p_rI_n)^{-1}BB^T(A - p_rI_n)^{-1}. \tag{74}$$

Videli smo, če je $X_{r-1} = X$ potem prav tako velja $X_r = X$ in dobimo

$$X - X_r = f_{p_r}(A)(X - X_{r-1})f_{p_r}(A). \tag{75}$$

Privzemimo, da je $X_0 = 0$. Z rekurzijo (75) potem dobimo oceno

$$X - X_r = f_{p_1,\ldots,p_r}(A)Xf_{p_1,\ldots,p_r}(A) \tag{76}$$

Po (74) za rang X_r velja, da je $\text{rang}(X_r) \leq \text{rang}(X_{r-1}) + m$ in posledično je $\text{rang}(X_r)$ največ mr.
Če združimo dejstva, dobimo

$$\frac{\lambda_{mr+1}(X)}{\lambda_1(X)} \leq \frac{\|X - X_r\|_2}{\|X\|_2} \leq \frac{\|f_{p_1,\ldots,p_r}(A)Xf_{p_1,\ldots,p_r}(A)\|_2}{\|X\|_2} \leq \|f_{p_1,\ldots,p_r}(A)\|_2^2. \tag{77}$$

Ocenimo sedaj $\|f_{p_1,\ldots,p_r}(A)\|_2$.

$$\begin{aligned} \|f_{p_1,\ldots,p_k}(A)\|_2 &= \max\{\sigma(f_{p_1,\ldots,p_r}(A))\} \\ &= \max\{|f_{p_1,\ldots,p_r}(\lambda)|; \lambda \in \sigma(A)\}. \end{aligned} \tag{78}$$

Enakost (78) velja, ker je $f_{p_1,\ldots,p_r}(A)$ simetrična matrika. Če zaporedoma označimo $\lambda_1(A) = \beta$ in $\lambda_n(A) = \alpha$ največjo in najmanjšo lastno vrednost matrike A, potem lahko izraz (78) omejimo navzgor z

$$\max\{|f_{p_1,\ldots,p_k}(\lambda)|; \lambda \in [\alpha, \beta]\}. \tag{79}$$

Preden nadaljujemo si podrobneje oglejmo funkcijo $f_p(t)$. Funkcija je na domeni $\mathbb{R}_-$ monotono naraščujoča za vsak $p \in \mathbb{R}_-$. To lahko preprosto preverimo z odvajanjem, saj je v tem primeru odvod pozitiven.
Velja še $|f_p(t)| < 1$ za poljubna $p, t \in \mathbb{R}_-$.
Izberemo $\tilde{\alpha}, \tilde{\beta} \in \mathbb{R}$ tako, da velja $\tilde{\beta} < \tilde{\alpha} < 0$, sicer poljubna in definiramo $\tilde{\kappa} = \frac{\tilde{\beta}}{\tilde{\alpha}}$ in

$$\tilde{p} = -(\tilde{\alpha}\tilde{\beta})^{1/2}. \tag{80}$$

Potem velja

$$\begin{aligned} 0 < -f_{\tilde{p}}(\tilde{\beta}) &= -\frac{-(\tilde{\alpha}\tilde{\beta})^{1/2} - \tilde{\beta}}{-(\tilde{\alpha}\tilde{\beta})^{1/2} + \tilde{\beta}} \\ &= f_{\tilde{p}}(\tilde{\alpha}) = \frac{\sqrt{\tilde{\kappa}} - 1}{\sqrt{\tilde{\kappa}} + 1} < 1. \end{aligned} \tag{81}$$

Iz prve vrstice dobimo drugo tako, da števec in imenovalec pomnožimo z $-\frac{1}{\tilde{\beta}} \neq 0$.
Zaradi monotonosti $f_{\tilde{p}}(t)$ mora veljati

$$|f_{\tilde{p}}(t)| \leq \frac{\sqrt{\tilde{\kappa}} - 1}{\sqrt{\tilde{\kappa}} + 1} \tag{82}$$

za vsak $t \in [\tilde{\beta}, \tilde{\alpha}]$. Končno neenakost lahko sedaj uporabimo za oceno (79).
Tvorimo zaporedje $t_0 = \alpha$ in $t_i = t_0(\frac{\beta}{\alpha})^{i/r} = t_0\kappa^{i/r}$ za $i = 0, \pm 1, \pm 2, \ldots$. Dobljeno geometrijsko zaporedje, za katerega velja $t_{i+1} < t_i$ in $t_r = \beta$, tvori particijo intervala $[\beta, \alpha]$ na r podintervalov. Za točke p_r sedaj vzamemo geometrično sredino posameznega podintervala pomnoženo z -1, tj. $p_i = -\sqrt{t_{i-1}t_i}$. Velja še

$p_i = -\sqrt{t_i t_{i-1}} = -\sqrt{t_{i+j} t_{i-1-j}}$, za $j = 0, 1, 2, \ldots$.
Za boljšo preglednost uvedemo nove (pomožne) spremenljivke

$$\kappa_{r,j} = \frac{t_{i+|j|}}{t_{i-1-|j|}} = \kappa^{(2|j|+1)/r} \tag{83}$$

in

$$R_{r,j} = \frac{\sqrt{\kappa_{r,j}} - 1}{\sqrt{\kappa_{r,j}} + 1} = \frac{\kappa^{(2|j|+1)/2r} - 1}{\kappa^{(2|j|+1)/2r} + 1} \tag{84}$$

za $j = 0, \pm 1, \pm 2, \ldots$. Če vzamemo $\tilde{\alpha} = t_{i-1-j}$ in $\tilde{\beta} = t_{i+j}$, lahko po (82) ocenimo

$$|f_{\tilde{p}}(t)| \leq R_{r,j},$$

kjer je $t \in [t_{i+j}, t_{i-1+j}] \subseteq [t_{i+|j|}, t_{i-1+|j|}]$, $i = 1, 2, \ldots, k$ in $j = 0, \pm 1, \pm 2, \ldots$. V posebnem desna stran ni odvisna od i.
Če sedaj privzamemo, da $[t_l, t_{l-1}]$ za $l = 1, 2, \ldots, r$ tvorijo poljubne podintervale $[\beta, \alpha]$, $t \in [t_l, t_{l-1}]$ in zmnožimo neenakosti, dobimo oceno

$$|f_{p_1,\ldots,p_r}(t)| \leq \prod_{i=1}^{r} R_{r,l-i} \leq \prod_{i=1}^{r-1} R_{r,j}. \tag{85}$$

Desna neenakost velja, ker je

$$0 < R_{r,0} < R_{r,-1} = R_{r,1} < R_{r,-2} = R_{r,-2} < \ldots.$$

To vidimo, ker je $R_{r,j}$ monotono odvisna od $\kappa_{r,j}$ slednja pa monotono odvisna od j. Desna stran neenakosti (85) ni odvisna od l, torej predstavlja zgornjo mejo za poljuben $t \in [\beta, \alpha]$.

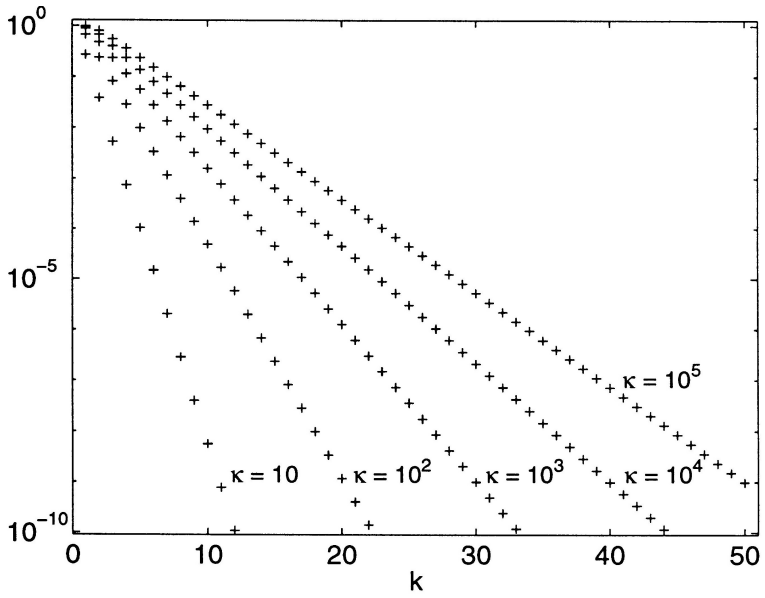
□

Penzlove meje zagotavljajo, da imajo rešitve (6) dobro aproksimacijo nizkega ranga, tudi če je A slabo pogojena.
Druga ključna lastnost, ki jo je predstavil Penzl [10], je, da v smiselnih mejah pogojenosti, tj. takšnih ki jih srečamo v praksi, za simetrične stabilne matrike A vrednost $\frac{\|X - X_r\|_2}{\|X\|_2}$ hitro pada proti 0.

Za Smithovo metodo dobimo optimalno izbiro premika $\alpha = -\sqrt{ab}$, kot v (80), kjer sta a in b najmanjša in največja lastna vrednost matrike A.

Penzl [11] je, v primeru ko rešujemo enačbo Ljapunova (6), ugotovil, da heuristično dobimo boljše rezultate, če za premike uporabimo ocene prvih k_+ Ritzevih vrednosti matrike A in prvih k_- Ritzevih vrednosti matrike A^{-1}. Slednji seveda aproksimirajo k_- najmanjših lastnih vrednosti matrike A. Alternativno lahko torej poskusimo heuristično oceniti premike pri reševanju Sylvestrove enačbe (5) z lastnimi vrednostmi matrik A in B. Tako na primer ocenimo največjih $k/2$ in najmanjših $k/2$ lastnih vrednosti matrike A in največjih $k/2$ in najmanjših $k/2$ lastnih vrednosti matrike B ter jih uporabimo za premike. Lahko poiščemo tudi več Ritzevih vrednosti in izberemo k vrednosti, ki se najmanj razlikujejo od lastnih vrednosti.

Premike, ki dosežejo največje zmanjšanje norme ostanka navadno želimo uporabiti najprej. Poglejmo si še heuristično izbiro premikov, ki so jo predlagali Brenner, Li



Slika 1. Odvisnost Penzlove ocene od κ in k, povzeto po [10]

in Truhar [2], ki se uporablja tudi, ko matriki A in B nista hermitski (simetrični). V tem primeru vzamemo za premike urejena nabora $\{\alpha_r\}$ in $\{\beta_r\}$ na naslednji način:

Algoritem 6: Izbira premikov s pomočjo Ritzevih vrednosti

VHOD: Matrike A, B, G in F ($C = GF^H$) ter število korakov i
IZHOD: Premiki α_r in β_r za metodo ADI

Izvedemo Arnoldijev algoritem za A na G, da dobimo množico $\mathbb{E}_A^+$ Ritzevih vrednosti
Izvedemo Arnoldijev algoritem za A^{-1} na G, da dobimo množico $\mathbb{E}_A^-$ Ritzevih vrednosti
$\mathbb{E} = \mathbb{E}_A^+ \cup (1/\mathbb{E}_A^-)$
Izvedemo Arnoldijev algoritem za B^H na F, da dobimo množico $\mathbb{F}_B^+$ Ritzevih vrednosti
Izvedemo Arnoldijev algoritem za $(B^H)^{-1}$ na F, da dobimo množico $\mathbb{F}_B^-$ Ritzevih vrednosti
$\mathbb{F} = \mathbb{F}_B^+ \cup (1/\mathbb{F}_B^-)$
Poiščemo $\alpha_1 \in \mathbb{E}$ in $\beta_1 \in \mathbb{F}$ tako, da minimizirata

$$\min_{\substack{\alpha\in\mathbb{E},\\ \beta\in\mathbb{F}}} \max_{\substack{x\in\mathbb{E},\\ y\in\mathbb{F}}} \left| \frac{(x-\alpha)(y-\beta)}{(x-\beta)(y-\alpha)} \right|$$

for $r = 2, \ldots i$
$\mathbb{E}'$ je množica $\mathbb{E}$ brez vrednosti $\alpha_1, \alpha_2, \ldots, \alpha_{r-1}$
$\mathbb{F}'$ je množica $\mathbb{F}$ brez vrednosti $\beta_1, \beta_2, \ldots, \beta_{r-1}$
Poiščemo $\alpha_r \in \mathbb{E}'$ in $\beta_r \in \mathbb{F}'$, ki minimizirata

$$\min_{\substack{\alpha\in\mathbb{E}',\\ \beta\in\mathbb{F}'}} \max_{\substack{x\in\mathbb{E},\\ y\in\mathbb{F}}} \left| \frac{(x-\alpha)(y-\beta)}{(x-\beta)(y-\alpha)} \right| \prod_{j=1}^{r-1} \left| \frac{(x-\alpha_j)(y-\beta_j)}{(x-\beta_j)(y-\alpha_j)} \right|$$

end for

Če sta G in F vektorja, potem izvedemo standarden Arnoldijev algoritem, če sta G in F matriki pa izvedemo bločno verzijo Arnoldijevega algoritma, opisano v [6].
Za dano matriko A in vektor b definiramo podprostor Krilova kot

$$K_r(A,b) = \mathrm{Lin}(b, Ab, \ldots, A^{r-1}). \tag{86}$$

Naj stolpci Q_k tvorijo ortonormirano bazo $K_k(A,b)$ za $k \leq r$. Potem pripadajočim lastnim vrednostim matrike $H_k = Q_k^H A Q_k$ pravimo Ritzeve vrednosti, ki jih uporabimo za približke lastnih vrednosti matrike A. Ko povečujemo k, se povečuje razsežnost podprostora Krilova in Ritzeve vrednosti postajajo vedno boljši približki za lastne vrednosti matrike A. Arnoldijev algoritem zgenerira matriki H_k in Q_k, s katerima lahko nato poiščemo Ritzeve vrednosti in hkrati poda napako. Če je A hermitska matrika, potem algoritmu pravimo Lanczoseva metoda. Več o obeh postopkih najdemo v [14].
Na podoben način bločni Arnoldijev algoritem da zgenerira ortonormirano bazo za $\mathrm{Lin}(B, AB, \ldots A^{k-1}B)$, ki jo zapišemo v $\tilde{Q}_k$ in $\tilde{H}_k$ tako, da velja $\tilde{H}_k = \tilde{Q}_k^H A \tilde{Q}_k$. Lastnim vrednostim $\tilde{H}_k$ ponovno rečemo Ritzeve vrednosti in prav tako predstavljajo približke za lastne vrednosti A. Več o bločnem Arnoldijevem algoritmu najdemo v [5].
Arnoldijev algoritem izvedemo $\min\{2i,\, i+10\}$ korakov in nato tipično izberemo i Ritzevih vrednosti z najmanjšo napako. E in F imata tedaj vsaka po $2i$ vrednosti. Izberemo lahko tudi manj vrednosti, do najmanj $i/2$ oz. $i/2+1$, če je i lih. S tem zmanjšamo število operacij, ki jih potrebuje algoritem za urejanje. Dodatne informacije o sami implementaciji tega algoritma najdemo v [2]. Algoritem pošiče Ritzeve vrednosti in jih razvrsti tako, da poskuša čimbolj zmanjšati normo ostanka. Podroben pregled še drugih različnih algoritmov za razvrščanje premikov najdemo v [16].

4.2. **Ocena Antoulas, Sorensen in Zhou**

Drugo oceno, ki si jo bomo pogledali, so razvili Antoulas, Sorensen in Zhou [1] za primerne diagonalizabilne matrike. S tem so dobili natančnejšo oceno za enačbo Ljapunova kot Penzl, ko je A simetrična oziroma hermitska matrika. Najprej si bomo pogledali primer, ko je $B = b \in \mathbb{R}^n$ vektor, $Q = bb^T$. Primer takšnih zveznih LTI sistemov so npr. SISO sistemi. Nato ocene posplošijo na standarden primer. Pomembno je, da z njihovo pomočjo dobimo ocene tudi za primer, ko A ni nujno simetrična.
Naj bo

$$A = VDV^{-1}$$

diagonalizacija matrike A, lastne vrednosti matrike A pa označimo z λ_k, $k = 1, 2, \ldots, n$. Vrstni red lastnih vrednosti bomo določili kasneje. Označimo

$$A_c = \begin{bmatrix} 0 & 0 & \ldots & 0 \\ 1 & 0 & & \\ & \ddots & \ddots & \\ 0 & \ldots & 1 & 0 \end{bmatrix} - \begin{bmatrix} \lambda_0 & \lambda_1 & \ldots & \lambda_{n-1} \end{bmatrix} \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}, \tag{87}$$

kjer so $\lambda_0, \lambda_1, \ldots, \lambda_{n-1}$ koeficienti karakterističnega polinoma matrike A. Označimo $e_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T$ in z G rešitev enačbe Ljapunova

$$A_c G + G A_c^T + e_1 e_1^T = 0.$$

Po predpostavki je par (A, b) vodljiv. Sklicali se bomo na dejstvo iz teorije upravljanja linearnih kontrolnih sistemov [13], da je matrika

$$K = [b \; Ab \; \ldots \; A^{n-1}b],$$

ki ji rečemo **vodljivostna matrika**, za par (A, b) nesingularna natanko tedaj, ko je par (A, b) vodljiv. Ker je K nesingularna, velja $Ke_1 = b$. Po Cayley-Hamiltonovemu izreku velja še

$$AK = KA_c. \tag{88}$$

Če sedaj X reši enačbo Ljapunova

$$AX + XA^T + bb^T = 0,$$

potem je $X = KGK^T$, saj po (88) velja

$$AKGK^T + KGK^T A^T + bb^T = K(A_c G + G A_c^T + e_1 e_1^T)K^T = 0.$$

Ob privzetku, da je A stabilna matrika, imamo še enoličnost.
Če lahko A diagonaliziramo, potem pa velja tudi

$$YA_c = \Lambda Y,$$

kjer je $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$ in so vrstice matrike Y enake $[1 \; \lambda_j \; \lambda_j^2 \ldots \; \lambda_j^{n-1}]$ za $j = 1, 2, \ldots, n$. Matriki Y rečemo **Vandermondova matrika**.
Definiramo še **Cauchyjevo jedro**

$$C := YGY^H.$$

Lema 4.3. *Naj bo $A \in \mathbb{R}^{n\times n}$ in $b \in \mathbb{R}^n$ tako, da je par (A, b) vodljiv. Privzemimo, da lahko matriko A diagonaliziramo. Naj bo P matrika (desnih) lastnih vektorjev matrike A, ki so dolžine 1, tako da velja*

$$AP = P\Lambda,$$

kjer je Λ diagonalna matrika. Naj bo X rešitev enačbe Ljapunova

$$AX + XA^T + bb^T = 0.$$

Potem je

$$X = P_b C P_b^H,$$

kjer je $P_b = P\mathrm{diag}(P^{-1}b)$ in matrika C Cauchyjevo jedro. Matrika C je še hermitska pozitivno definitna in njeni elementi so

$$c_{i,j} = \frac{-1}{\lambda_i + \overline{\lambda_j}}$$

Dokaz.
Najprej vidimo

$$\begin{aligned} K &= [b \; Ab \; A^2b \; \ldots \; A^{n-1}b] \\ &= [P\hat{b} \; P\Lambda\hat{b} \; P\Lambda^2\hat{b} \; \ldots \; P\Lambda^{n-1}\hat{b}] \\ &= P_b Y. \end{aligned} \tag{89}$$

Videli smo, da velja

$$X = KGK^T = P_bYG(P_bY)^H = P_bCP_b^H.$$

Ker je par (A, b) vodljiv, b ne more biti ortogonalen na noben levi lastni vektor matrike A, sicer je K singularna. Posledično $\hat{b} = P^{-1}b$ nima ničel in je matrika $\mathrm{diag}(\hat{b})$ nesingularna. Še več,

$$AP_b = P_b\Lambda \text{ in } P_b^{-1}b = [1\ 1\ \ldots\ 1]^T.$$

Torej,

$$0 = P_b^{-1}(AX + XA^T + bb^T)(P_b^{-1})^H = \Lambda C + C\Lambda^H + ee^T,$$

kjer je $e = [1\ 1\ \ldots\ 1]^T$. Če razpišemo, dobimo, da so elementi matrike C enaki $\frac{-1}{\lambda_i+\overline{\lambda_i}}$. Ker je A stabilna matrika je tudi Λ stabilna matrika in posledično je C hermitska pozitivno definitna matrika.

□

Sedaj imamo vse potrebno za oceno. Če je torej $b \in \mathbb{R}^n$ in je par (A, b) **dosegljiv** (ang. *reachable*), tj. za LTI sistem je par $(A,\ b)$ je vodljiv in $x_0 = 0$, potem lahko rešitev enačbe Ljapunova za A in $Q = bb^T$ zapišemo v obliki

$$X = P_bCP_b^T,$$

kjer sta $P_b = V\mathrm{diag}(P^{-1}b)$ in matrika C z elementi $c_{i,j} = -\frac{1}{\lambda_i+\overline{\lambda_j}}$.
Po privzetku, da je A stabilna, tj. lastne vrednosti A imajo negativen realni del, je C hermitska pozitivno definitna matrika. Potem lahko na C naredimo razcep Choleskega tako, da dobimo razcep $C = L\tilde{D}L^H$, kjer je $\tilde{D}$ diagonalna in L spodnja trikotna matrika, diagonalne elemente $\tilde{D}$ pa označimo z δ_k, $k = 1, 2, \ldots, n$.
Velja, da lahko elemente δ_k zapišemo kot

$$\delta_k = -\frac{1}{2\mathbf{Re}(\lambda_k)} \prod_{j=1}^{k-1} \left| \frac{\lambda_k - \lambda_j}{\overline{\lambda_k} + \lambda_j} \right|^2. \tag{90}$$

Sedaj določimo, da naj bodo lastne vrednosti λ_k razporejene tako, da velja

$$\delta_1 \geq \delta_2 \geq \cdots \geq \delta_n.$$

Ker je $\tilde{D}$ diagonalna matrika, jo lahko zapišemo kot

$$X = \sum_{j=1}^{n} \delta_j (P_bLe_j)(P_bLe_j)^H, \tag{91}$$

kjer z e_j označimo enotski vektor, ki ima na j-tem mestu 1, drugje pa ničle. Privzemimo še, da stolpci Le_j zadoščajo $\|Le_j\|_\infty = 1$. Potem lahko ocenimo $\|Le_j\|_2 \leq (n - j + 1)^{1/2}$. Če označimo $z_k = P_bLe_k$ in seštejemo prvih r členov (91), kot pri dokazu izreka 4.1, dobimo aproksimacijo X_r ranga r za X in oceno

$$\begin{aligned}
\sigma_{r+1} \leq \|X - X_r\|_2 &= \| \sum_{k=r+1}^{n} \delta_k z_k \overline{z_k} \|_2 \\
&\leq (n-r)\delta_{r+1} \max_{k>r} \|z_j\|_2^2 \\
&\leq (n-r)\delta_{r+1} \|P_b\|_2^2 \|\hat{b}\|_2^2 \|Le_j\|_2^2 \\
&\leq (n-r)\delta_{r+1} \kappa {P_b}^2 \|b\|_2^2 (n-k+1) \\
&\leq (n-r)^2 \delta_{r+1} \kappa^2(P_b) \|b\|^2.
\end{aligned} \tag{92}$$

Naj bo sedaj $B = [b_1\ b_2\ \dots\ b_m]$, kjer so $b_k \in \mathbb{R}^n$ stolpci matrike B za $k = 1, 2, \dots, m$. Potem lahko zapišemo enačbo Ljapunova (6) kot

$$AX + XA^T + \sum_{k=1}^{m} b_k b_k^T = 0. \tag{93}$$

Enačbo lahko razbijemo na m enačb kot v prejšnjem primeru. Za rešitev k-te enačbe Ljapunova velja $X^{(k)} = P_{b_k}^{(k)} C (P_{b_k}^{(k)})^T = P_{b_k}^{(k)} L \tilde{D} L^H (P_{b_k}^{(k)})^H$. Naj bo

$$Z_k = [P_{b_1}^{(1)} L e_j\ P_{b_2}^{(2)} L e_j\ \dots P_{b_m}^{(m)} L e_j] = [z_{1j}\ z_{2j}\ \dots z_{mj}].$$

Po linearnosti potem velja

$$X = \sum_{l=1}^{m} X^{(k)} = \sum_{k=1}^{n} \delta_k Z_k Z_k^H.$$

Izrek 4.4. *Naj bo $\hat{X}_{rm} = \sum_{k=1}^{r} \delta_k Z_k Z_k^H$. Če je $\frac{\delta_{r+1}}{\delta_1} < \epsilon$, potem je $\hat{X}_{rm}$ aproksimacija X ranga največ km, ki zadošča oceni*

$$\|X - \hat{X}_{rm}\|_2 \leq \epsilon \delta_1 m (n-r)^2 (\kappa(P)\|B\|_2)^2, \tag{94}$$

kjer je $\delta_1 \approx \|X\|_2$.

Dokaz.
Ker je $Z_k Z_k^H = \sum_{i=1}^{m} z_{ij} z_{ij}^H$, sledi

$$\begin{aligned} Z_k Z_k^H &\leq \sum_{i=1}^{m} \|z_{ij} z_{ij}^H\|_2 \\ &\leq m \max_i \|z_{ij} z_{ij}^H\|_2 \\ &\leq m \max_i (n-r)(\kappa(P)\|b_i\|_2)^2 \\ &\leq m \max_i (n-r)(\kappa(P)\|B\|_2)^2. \end{aligned} \tag{95}$$

Sedaj upoštevamo, da je $\delta_{r+1} < \delta_1 \epsilon$ in

$$\|X - \hat{X}_{rm}\|_2 = \| \sum_{k=r+1}^{n} \delta_k Z_k Z_k^H \|_2$$

ter združimo ocene v iskano oceno.

□

Vendar je pri uporabi te ocene potrebna previdnost. Penzl je pokazal, da ni mogoče omejiti $\frac{\sigma_r(X)}{\sigma_1(X)}$ samo z lastnimi vrednostni matrike A ali z Jordanovo formo matrike A in rangom vektorja b, $Q = bb^T$, kjer je X rešitev enačbe Ljapunova (6), ko A ni hermitska matrika. Vendar v praksi, ko občutljivost $\kappa(A)$ ni prevelika in zanemarimo preostale faktorje, lahko uporabimo

$$\frac{\delta_{r+1}}{\delta_1} = \frac{\mathbf{Re}(\lambda_1)}{\mathbf{Re}(\lambda_{r+1})} \prod_{j=1}^{r} \left| \frac{\lambda_{r+1} - \lambda_j}{\overline{\lambda_{r+1}} + \lambda_j} \right|^2.$$

za oceno $\frac{\sigma_r(X)}{\sigma_1(X)}$ [16].

4.3. **Občutljivost modificirane metode fADI**

V tem poglavju si bomo pogledali ocene za Sylvestrovo enačbo, kjer bomo predpostavili, da lahko matriki A in B diagonaliziramo, matriko C pa lahko zapišemo kot produkt GF^H. Rezultati tega podpoglavja so povzeti po [17].

Privzemimo, da lahko matriki A in B diagonaliziramo. Naj bosta

$$\begin{aligned} A &= S\Lambda S^{-1},\ \Lambda = \operatorname{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n) \text{ in} \\ B &= T\Omega T^{-1},\ \Omega = \operatorname{diag}(\mu_1, \mu_2, \ldots, \mu_m) \end{aligned} \tag{96}$$

diagonalizaciji matrik A in B. Potem lahko faktorja $Z^{(r)}$ in $Y^{(r)}$ modificirane metode fADI zapišemo kot

$$\begin{aligned} Z^{(r+1)} &= S(\Lambda + \beta_{r+1} I_n)^{-1}(\Lambda - \alpha_r I_n) S^{-1} Z^{(r)}, \\ Y^{(r+1)H} &= Y^{(r)H} T(\Omega + \alpha_{r+1} I_m)^{-1}(\Omega - \beta_r I_m) T^{-1}. \end{aligned} \tag{97}$$

Označimo

$$\begin{aligned} \Delta_r &= (\Lambda + \beta_r I_n)^{-1}(\Lambda - \alpha_r I_n), \\ \Theta_r &= (\Omega - \beta_r I_m)(\Omega + \alpha_{r+1} I_m)^{-1}. \end{aligned} \tag{98}$$

V teh oznakah sta

$$\begin{aligned} Z^{(r+1)} =& S(\Lambda - \alpha_r I_n)(\Lambda + \beta_{r+1} I_n)^{-1}(\Lambda - \alpha_{r-1} I_n) \cdots \\ & \cdots (\Lambda + \beta_2 I_n)^{-1}(\Lambda + \beta_1 I_n)^{-1} S^{-1} G \\ =& S(\Lambda + \beta_{r+1} I_n)^{-1} \Delta_r \Delta_{r-1} \cdots \Delta_1 S^{-1} G, \end{aligned} \tag{99}$$

kjer smo člene preuredili po lemi 3.2, in

$$\begin{aligned} Y^{(r+1)H} =& F^H T(\Omega + \alpha_1 I_m)^{-1}(\Omega + \alpha_2 I_m)^{-1}(\Omega - \beta_1 I_m) \cdots \\ & \cdots (\Omega + \alpha_{r+1} I_m)^{-1}(\Omega - \beta_r I_m) T^{-1} \\ =& F^H T \Theta_1 \Theta_2 \cdots \Theta_r (\Omega + \alpha_{r+1} I_m)^{-1} T^{-1}, \end{aligned} \tag{100}$$

kjer smo vmesne člene produkta ponovno preuredili. Opazimo, da sta

$$(\Lambda + \beta_{r+1} I_n)^{-1} \Delta_r \Delta_{r-1} \cdots \Delta_1$$

in

$$\Theta_1 \Theta_2 \cdots \Theta_r (\Omega + \alpha_{r+1} I_m)^{-1}$$

diagonalni matriki. Diagonalne vrednosti $(\Lambda + \beta_{r+1} I_n)^{-1} \Delta_r \Delta_{r-1} \cdots \Delta_1 S^{-1}$ so enake

$$\frac{1}{\lambda_j + \beta_{r+1}} \prod_{i=1}^{r} \frac{\lambda_j - \alpha_i}{\lambda_j + \beta_i},\ j = 1, 2, \ldots, r,$$

diagonalne vrednosti $\Theta_1 \Theta_2 \cdots \Theta_r (\Omega + \alpha_{r+1} I_m)^{-1}$ pa so enake

$$\frac{1}{\mu_j + \alpha_{r+1}} \prod_{i=1}^{r} \frac{\mu_j - \beta_i}{\mu_j + \alpha_i},\ j = 1, 2, \ldots, r.$$

Če sedaj za parametre izberemo lastne vrednosti (večkratne z njihovo algebraično večkratnostjo) $\alpha_i = \lambda_{p_i}$ za $i = 1, 2, \ldots, n$ in $\beta_j = \mu_{q_j}$ za $j = 1, 2, \ldots, m$, kjer sta p_i in

q_j indeksa permutacij, ki določata vrstni red v katerem uporabimo premike, lahko rešitev Sylvestrove enačbe X zapišemo kot

$$X = X_{n_0} = S\Big(\sum_{j=1}^{n_0}(\mu_{q_j} + \lambda_{p_j})\Phi^{(j)}\Psi^{(j)}\Big)T^{-1},\ n_0 = \min\{n, m\}, \tag{101}$$

kjer smo uporabili oznake

$$\Phi^{(j)} = \mathrm{diag}\Big(\frac{\sigma(1, j-1)}{\lambda_1 + \mu_{q_j}}, \cdots, \frac{\sigma(n, j-1)}{\lambda_n + \mu q_j}\Big)S^{-1}G, \tag{102}$$

kjer je

$$\sigma(r, j) = \prod_{i=1}^{j-1}\frac{\lambda_r - \lambda_{p_i}}{\lambda_r + \mu_{q_i}},$$

$\sigma(r, 0) = 1$ za $r = 1, 2, \ldots, n$ in

$$\Psi^{(j)} = F^H T\mathrm{diag}\Big(\frac{\tau(1, j-1)}{\mu_{p_j} + \lambda_1}, \cdots, \frac{\tau(m, j-1)}{\mu_{p_j} + \lambda_m}\Big), \tag{103}$$

kjer je

$$\tau(r, j) = \prod_{i=1}^{j-1}\frac{\mu_r - \mu_{q_i}}{\mu_r + \lambda_{p_i}},$$

$\sigma(r, 0) = 1$ za $r = 1, 2, \ldots, m$.
S pomočjo oznak lahko sestavimo oceno za normo X.

Izrek 4.5. *Privzemimo, da lahko matriki A in B diagonaliziramo in je njuna diagonalizacija enaka (96). Naj bo X rešitev Sylvestrove enačbe (5). Potem za X velja*

$$\|X\| \leq \|S\|\cdot\|T^{-1}\| \sum_{j=1}^{n_0}|\mu_{q_j} + \lambda_{p_j}| \sum_{i=1}^{n}\frac{|\sigma(i, j-1)|\cdot\|\hat{g}_i\|}{|\lambda_i + \mu_{q_j}|}\sum_{l=1}^{m}\frac{|\tau(l, j-1)|\cdot\|\hat{f}_l\|}{|\mu_l + \lambda_{p_j}|}, \tag{104}$$

kjer sta σ in τ kot v (102) in (103), $n_0 = \min\{m, n\}$, $\hat{g}_i$ označuje i-to vrstico matrike $S^{-1}G$ in $\hat{f}_l$ označuje l-ti stolpec matrike $F^H T$.

Dokaz.
Na X iz (101) uporabimo trikotniško neenakost ter upoštevamo definiciji Φ in Ψ.

□

Za oceno občutljivosti potrebujemo še naslednji izrek.

Izrek 4.6. *Privzemimo, da lahko matriki A in B diagonaliziramo in je njuna diagonalizacija enaka (96). Naj bo X rešitev Sylvestrove enačbe (5) in X_r r-ta aproksimacija, ki jo dobimo z modificirano metodo fADI, kjer smo za premike vzeli urejene lastne vrednosti matrik A in B, tj. $\alpha_i = \lambda_{p_i}$, $i = 1, 2, \ldots, r$ in $\beta_j = \mu_{q_j}$, $j = 1, 2, \ldots, r$. Potem velja*

$$\|X - X_r\| \leq \|S\|\cdot\|T^{-1}\| \sum_{j=r+1}^{n_0}|\mu_{q_j} + \lambda_{p_j}| \sum_{i=1}^{n}\frac{|\sigma(i, j-1)|\cdot\|\hat{g}_i\|}{|\lambda_i + \mu_{q_j}|}\sum_{l=1}^{m}\frac{|\tau(l, j-1)|\cdot\|\hat{f}_l\|}{|\mu_l + \lambda_{p_j}|}, \tag{105}$$

kjer sta σ in τ kot v (102) in (103), $n_0 = \min\{m, n\}$, $\hat{g}_i$ označuje i-to vrstico matrike $S^{-1}G$ in $\hat{f}_j$ označuje j-ti stolpec matrike $F^H T$ kot v prejšnjem izreku.

Dokaz.
X in X_r lahko zapišemo kot

$$X_r = \sum_{j=1}^{r} (\mu_{q_j} + \lambda_{p_j}) Z^{(j)} Y^{(j)H} \tag{106}$$

in

$$X = \sum_{j=1}^{n_0} (\mu_{q_j} + \lambda_{p_j}) Z^{(j)} Y^{(j)H}, \tag{107}$$

kjer je $n_0 = \min\{n, m\}$. Sledi, da lahko razliko zapišemo kot

$$X - X_r = \sum_{j=r+1}^{n_0} (\mu_{q_j} + \lambda_{p_j}) Z^{(j)} Y^{(j)H} = S\Big(\sum_{j=r+1}^{n_0} (\mu_{q_j} + \lambda_{p_j}) \Phi^{(j)} \Psi^{(j)} \Big) T^{-1}. \tag{108}$$

Podobno kot v prejšnjem izreku vzamemo normo $\|X - X_r\|$, uporabimo trikotniško neenakost in upoštevamo oznake.

□

Poglejmo si sedaj zmoteno Sylvestrovo enačbo

$$(A + \delta A)(X + \delta X) + (X + \delta X)(B + \delta B) = (G + \delta G)(F + \delta F)^H. \tag{109}$$

Če zanemarimo napake drugega reda (109)in odštejemo eksaktno Sylvestrovo enačbo (5), potem dobimo

$$A\delta X + \delta X B \approx G\delta F^H + \delta G F^H - \delta A X - X \delta B. \tag{110}$$

δX iz (110) lahko potem ocenimo z $\delta X \approx \delta X_1 + \delta X_2 + \delta X_3$, kjer δX_1, δX_2, δX_3 ustrezajo Sylvestrovim enačbam

$$\begin{aligned} A\delta X_1 + \delta X_1 B &= -\delta A X, \\ A\delta X_2 + \delta X_2 B &= -X \delta B, \\ A\delta X_3 + \delta X_3 B &= -G\delta F^H + \delta G F^H. \end{aligned} \tag{111}$$

Ponovno vzamemo za premike lastne vrednosti eksaktnih matrik A in B in dobimo

$$\delta X_1 = S\Big(\sum_{j=1}^{n_0} (\mu_j + \lambda_j) \hat{\Phi}^{(j)} \hat{\Psi}^{(j)} \Big) T^{-1},\ n_0 = \min\{n, m\}, \tag{112}$$

kjer sta

$$\begin{aligned} \hat{\Phi}^{(j)} &= \mathrm{diag}\Big(\frac{\sigma(1, j-1)}{\lambda_1 + \mu_{q_j}}, \cdots, \frac{\sigma(n, j-1)}{\lambda_n + \mu_{q_j}} \Big) S^{-1} \delta A, \\ \hat{\Psi}^{(j)} &= XT\mathrm{diag}\Big(\frac{\tau(1, j-1)}{\mu_{p_j} + \lambda_1}, \cdots, \frac{\tau(m, j-1)}{\mu_{p_j} + \lambda_m} \Big), \end{aligned} \tag{113}$$

δ in τ sta enaka kot prej. Označimo

$$\delta^{(j)}_{\max} = \max_i \frac{|\sigma(i, j-1)|}{|\lambda_i + \mu_j|} \text{ in } \tau^{(j)}_{\max} = \max_i \frac{|\tau(i, j-1)|}{|\mu_i + \lambda_j|}. \tag{114}$$

Po (112) in trikotniški neenakosti je potem

$$\|\delta X_1\| \leq \sum_j^l |\mu_j + \lambda_j| \|S\| \delta_{max}^{(j)} \|S^{-1}\delta A\| \|X\| \|T\| \tau_{\max}^{(j)} \|T^{-1}\|, \tag{115}$$

oziroma

$$\frac{\|\delta X_1\|}{\|X\|} \leq \kappa(T)\|S\|\|S^{-1}\delta A\| \sum_j^l |\mu_j + \lambda_j| \delta_{\max}^{(j)} \tau_{\max}^{(j)}. \tag{116}$$

Analogno izpeljemo ocene za δX_2 in δX_3

$$\frac{\|\delta X_2\|}{\|X\|} \leq \kappa(S)\|\delta B T\|\|T^{-1}\| \sum_j^l |\mu_j + \lambda_j| \delta_{\max}^{(j)} \tau_{\max}^{(j)}, \tag{117}$$

$$\frac{\|\delta X_3\|}{\|X\|} \leq \|S\|\|S^{-1}(G\delta F^H + \delta G F^H)T\|\|T\| \sum_j^l |\mu_j + \lambda_j| \delta_{\max}^{(j)} \tau_{\max}^{(j)}. \tag{118}$$

Izrek 4.7. *Privzemimo, da lahko matriki A in B diagonaliziramo in je njuna diagonalizacija enaka (96). Naj bo X rešitev Sylvestrove enačbe (5) in $X + \delta X$ rešitev*

$$(A + \delta A)(X + \delta X) + (X + \delta X)(B + \delta B) = (G + \delta G)(F + \delta F)^H. \tag{119}$$

Za dovolj majhen $\epsilon = \max\{\|\delta A\|, \|\delta B\|, \|\delta G\|, \|\delta F\|\}$ velja ocena

$$\begin{aligned} \frac{\|\delta X\|}{\|X\|} \leq & \Big(\kappa(T)\|\|S\|\|S^{-1}\| + \kappa(S)\|T^{-1}\|\|\delta B T\| \\ & \|S\|\|T^{-1}\| \frac{\|S^{-1}(G\delta F^H + \delta G F^H)T\|}{\|X\|} \Big) \gamma + \mathcal{O}(\epsilon^2), \end{aligned} \tag{120}$$

kjer je $\gamma = \sum_j^l |\mu_j + \lambda_j| \tau_{\max}^{(j)} \sigma_{\max}^{(j)}$, $\tau_{\max}^{(j)}$ in $\sigma_{\max}^{(j)}$ pa sta definirana kot prej.

Dokaz.
δX ocenimo kot vsoto $\delta X \approx \delta X_1 + \delta X_2 + \delta X_3$. Sledi

$$\frac{\|\delta X\|}{\|X\|} \approx \frac{\|\delta X_1 + \delta X_2 + \delta X_3\|}{\|X\|} \leq \frac{\|\delta X_1\|}{\|X\|} + \frac{\|\delta X_2\|}{\|X\|} + \frac{\|\delta X_2\|}{\|X\|}. \tag{121}$$

Na vsakem členu vsote uporabimo pripadajoče ocene (116), (117) in (118) ter dobimo željen rezultat.

□

Podobno kot pri oceni, ki so jo razvili Antoulas, Sorensen in Zhou, tudi tu velja previdnost pri uporabi ocene.

4.4. **Ocena premikov preko eliptičnih funkcij**

V nadaljevanju bomo potrebovali teorijo o Jacobijevih eliptičnih funkcijah. Jacobijeve eliptične funkcije lahko formuliramo na podoben način kot trigonometrične funkcije [16]. Tako, na primer, za zvezno monotono funkcijo $\tilde{u} : [0, 1] \to \mathbb{R}$ s predpisom

$$\tilde{u}(x) = \int_0^x \frac{dt}{\sqrt{1 - t^2}}$$

lahko definiramo inverzno funkcijo

$$x(\tilde{u}) = \sin(\tilde{u}).$$

Vzemimo sedaj za fiksen $0 < k < 1$ funkcijo $u : [0, 1] \to \mathbb{R}$ s predpisom

$$u(x) = \int_0^x \frac{dt}{\sqrt{(1-t^2)(1-(kt)^2)}}.$$

Funkcija u je kot funkcija zgornje meje zvezna, nenegativnost integranda pa nam zagotavlja, da je tudi monotona. Torej ima inverz, ki ga označimo z

$$x(u) = \mathrm{sn}(u).$$

Definiramo še dve funkciji $\mathrm{cn}(u)$ in $\mathrm{dn}(u)$, tako, da za vsak $u \in [0, 1]$ velja

$$cn^2(u) = 1 - \mathrm{sn}^2(u),$$

$$dn^2(u) = 1 - k^2\mathrm{sn}^2(u).$$

Parametru k rečemu **modus**. Potrebujemo še konstanto $K = u(1)$, ki ji rečemo **popoln eliptični integral** (ang. *complete elliptic integral*). K je odvisna od modusa k, ni pa odvisna od x, zato jo obravnavamo kot konstanto. Analogno definiramo še **komplementarni eliptični integral** (ang. *complementary elliptic integral*) K', kjer v funkciji u namesto k vzamemo modus $k' = \sqrt{1-k^2}$. S pomočjo K in K' lahko definiramo **nome** (ang. *nome*)

$$q = e^{-\pi \frac{K}{K'}}.$$

V Octave-u kot tudi v Matlabu se za izračune eliptičnih integralov uporablja tudi parameter $m = k^2 = 1 - (k')^2$. Prav tako, če poznamo vrednost poljubnega parametra, so vsi ostali enolično določeni.

V nadaljevanju bomo rešitev problema (66) predstavili v obliki eliptičnih funkcij, ki jo najdemo v prilogi vira [18].
Najprej zamenjamo x in y z novimi spremenljivkami u in v tako, da velja

$$x = \frac{u-h}{f-gu},\ y = \frac{v+h}{f+gv}, \tag{122}$$

kjer bomo f, g in h določili kasneje.
Potem postane problem izbire premikov na intervalskih ocenah spektrov (66) minimaks funkcije

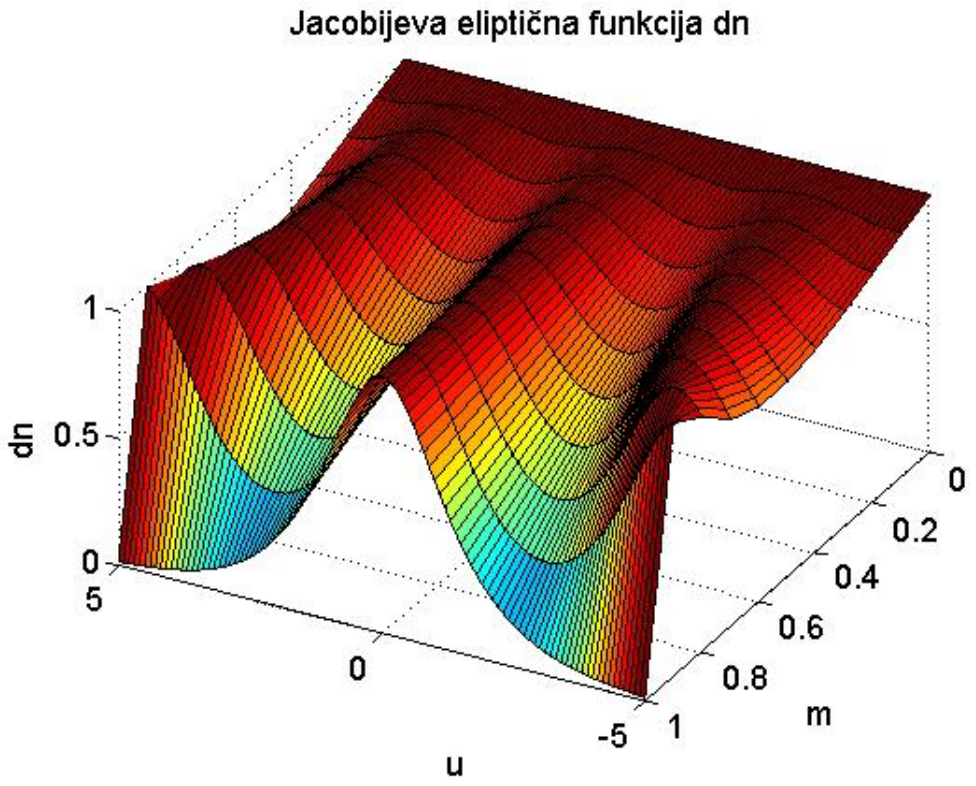
$$Q_r = \prod_{j=1}^{r} \left(\frac{u-\tilde{\beta}_j}{u+\tilde{\alpha}_j}\right)\left(\frac{v-\tilde{\alpha}_j}{v+\tilde{\beta}_j}\right), \tag{123}$$

nove vrednosti $\tilde{\alpha}_j, \tilde{\beta}_j$ pa dobimo iz enačb

$$\tilde{\alpha_j} = \frac{f\alpha_j - h}{1-g\alpha_j} \text{ in } \tilde{\beta}_j = \frac{f\beta_j + h}{1+g\beta_j}.$$

Vrednosti f, g in h želimo izbrati tako, da nam $x = x(u)$ preslika $[a, b] \mapsto [k', 1]$ in $y = y(v)$ preslika $[c, d] \mapsto [k', 1]$. To lahko naredimo, če je k' ničla kvadratne funkcije

$$(k')^2 - 2\left(1 + 2\frac{(b-a)(d-c)}{(a+c)(b+d)}\right)k' + 1 = (k')^2 - 2(1+\eta)k' + 1,$$



SLIKA 2. Prikaz eliptične funkcije dn

kjer smo z η označili

$$\eta = 2\frac{(b-a)(d-c)}{(a+c)(b+d)}. \tag{124}$$

Ker sta A in B stabilni matriki, velja $b-a>0$, $d-c>0$, $a+c<0$, $b+d<0$ in mora biti $\eta>0$. Tedaj so ničle kvadratne enačbe realne in pozitivne ter dobimo rešitvi

$$k_1' = \frac{-2(\eta+1)+2\sqrt{(\eta+1)^2-1}}{2} = (\eta+1)+\sqrt{\eta(\eta+2)}$$

in

$$k_2' = (\eta+1)-\sqrt{\eta(\eta+2)} = \frac{1}{(\eta+1)+\sqrt{\eta(\eta+2)}}. \tag{125}$$

Izberemo manjšo ničlo $k'=k_2'$ tako, da je $0<k'<1$ in dobimo smiseln interval $[k',1]$. Sedaj lahko poračunamo še d, g in h:

- $$g = \frac{2(k'(b+d)-(a+c))}{(a+c)(b-d)+k'(b+d)(c-a)}, \text{ ali } g=0, \text{ če } a=b \text{ in } c=d,$$

- $$f = \frac{2+g(b-d)}{b+d},$$

- $$h = \frac{k'(c - a + 2acg)}{a + c}.$$

Iz enoličnosti, ker smo oba intervala preslikali na $[k', 1]$, potem sledi, da morata biti nabora parametrov $\{\tilde{\alpha_j}\}$ in $\{\tilde{\beta_j}\}$ enaka in lahko Q_r zapišemo kot

$$Q_r = P(u)P(v), \tag{126}$$

kjer je

$$P(u) = \prod_{j=1}^{r} \frac{u - \tilde{\alpha_j}}{u + \tilde{\alpha_j}}. \tag{127}$$

Posledično, če P izpolnjuje minimaks pogoj (66), ga mora tudi Q_r in je dovolj poiskati rešitev P.

Navedimo dva izreka, ki nam bosta pomagala pri določitvi iskanih parametrov. Dokaze obeh izrekov najdemo v [20].

Izrek 4.8. *Če P zavzame vrednosti $\tau_1, -\tau_2, \tau_3, \ldots, (-1)^r\tau_{r+1}$ v točkah x_j, ki predstavljajo monotono naraščajočo delitev intervala $[a, b]$ in je P zvezna na $[a, b]$, potem je rešitev minimaks problema*

$$H = \min_{\tilde{\alpha} \in \mathbb{C}^n} \max_{x \in [a,b]} |P(u)|, \tag{128}$$

$\tilde{\alpha} = (\tilde{\alpha}_1, \tilde{\alpha}_2, \ldots \tilde{\alpha}_r)$, navzdol omejena z najmanjšim x_j.

Izrek 4.9. *Funkcija $P(u)$, ki se na intervalu $[a, b]$ najmanj razlikuje od 0, doseže svoje maksimalno odstopanje na $[a, b]$ alternirajoče (z alternirajočim predznakom), $(r + 1)$-krat.*

Posledica 4.10. *Ključna posledica obeh izrekov je, če $P(u)$ doseže maksimalno odstopanje na $[a, b]$ (v našem primeru je interval $[k', 1]$) $(r + 1)$-krat, potem mora biti le-ta maksimum najmanjša možna vrednost in nam reši problem H.*

Za določitev parametrov $\tilde{\alpha_j}$ vzemimo vrednosti

$$u = \mathrm{dn}(Kz),\ \tilde{k} = \sqrt{1 - (k')^2} \text{ in } \tau = i\frac{K'}{K}, \tag{129}$$

tako, da $P(u)$ (127) postane

$$P = \prod_{j=1}^{r} \frac{\mathrm{dn}(Kz) - \tilde{\alpha_j}}{\mathrm{dn}(Kz) + \tilde{\alpha_j}}.$$

Za P velja, da je kot ulomek dveh eliptičnih funkcij tudi sam eliptična funkcija v spremenljivki z in ko u preteče interval $[k', 1]$, spremenljivka z preteče $[0, 1]$. Oglejmo si sedaj funkcijo

$$P' = (-1)^{r-1}\frac{\mathrm{dn}(rK_1z) - \sqrt{\tilde{k'}}}{\mathrm{dn}(rK_1z) + \sqrt{\tilde{k'}}}, \tag{130}$$

s parametri modus $= \tilde{k}$, $\tilde{\tau} = r\tau$. Vidimo, da ima P' naslednje lastnosti

(1) P' doseže svoj maksimum $\frac{1-\sqrt{\tilde{k'}}}{1+\sqrt{\tilde{k'}}}$ pri vrednosti $z = 0$,

(2) P' doseže svoj minimum, ki je enak $-\frac{1-\sqrt{\tilde{k}'}}{1+\sqrt{\tilde{k}'}}$, pri $z = 1/r$,
(3) P' ima realno periodo $2/r$ in alternira $(r+1)$-krat v spremenljivki z na $[0,1]$,
(4) P' ima tudi imaginarno periodo $4\tilde{\tau}/r = 4\tau$,
(5) v točki $z = \tilde{\tau}$ ima P' vrednost 1,
(6) po (1), (2) in (3) P' zadošča minimaks pogoju, kot smo videli v posledici 4.4 prejšnjih dveh izrekov, tj. najbolje aproksimira 0 na $[0,1]$ glede na maksimalno vrednost.
(7) ker P tudi zadošča prejšnjima točkama (4) in (5) bosta P in P' enaka, če se ujemata v vseh polih in v vseh ničlah (Liouvileov izrek iz kompleksne analize).

Ničle P' so v $z = \frac{2j-1}{2r}$ in poli so v $z = 2\tau + \frac{2j-1}{2r}$. Funkciji bosta potem enaki, če vzamemo

$$\tilde{\alpha_j} = \mathrm{dn}\left(\frac{(2j-1)K}{2r}\right) \text{ in } k = \sqrt{1-(k')^2}. \tag{131}$$

Pripadajoče α_j in β_j dobimo iz enačb

$$\alpha_j = \frac{\tilde{\alpha_j} + h}{f + g\tilde{\alpha_j}} \text{ in } \beta_j = \frac{\tilde{\alpha_j} - h}{f - g\tilde{\alpha_j}}. \tag{132}$$

Ker P' doseže svoj maksimum $\frac{1-\sqrt{k'}}{1+\sqrt{k'}}$ pri vrednosti $z = 0$, velja

$$\max_{x\in[a,b], y\in[c,d]} \prod_{j=1}^{k} \left|\frac{(x-\alpha_j)(y-\beta_j)}{(x+\beta_j)(y+\alpha_j)}\right| = \left(\frac{1-\sqrt{k'}}{1+\sqrt{k'}}\right)^2. \tag{133}$$

Povzemimo ključne ugotovitve v naslednjem izreku [13].

Izrek 4.11. *Naj bosta $A \in \mathbb{C}^{n\times n}$ in $B \in \mathbb{C}^{m\times m}$ stabilni hermitski matriki. Naj bosta $[a,b]$ in $[c,d]$ intervala, ki vsebujeta vse lastne vrednosti za matriki A in B. Naj bo*

$$\eta = 2\frac{(b-a)(d-c)}{(a+c)(b+d)}.$$

Naj bosta še matriki $G \in \mathbb{C}^{n\times p}$ in $F \in \mathbb{C}^{p\times m}$, kjer je $p \ll \min(n,m)$. Singularne vrednosti $\sigma_i(X)$ rešitve Sylvestrove enačbe

$$AX + XB = -GF^H$$

zadoščajo

$$\frac{\sigma_{pr+1}(X)}{\sigma_1(X)} \le \left(\frac{1-\sqrt{k'}}{1+\sqrt{k'}}\right)^2. \tag{134}$$

kjer je $k' = 1/(1+\eta+\sqrt{\eta(\eta+2)})$, $k = \sqrt{1-k'^2}$.

Zapišimo še psevdokodo, kako poiskati premike s pomočjo eliptičnh integralov.

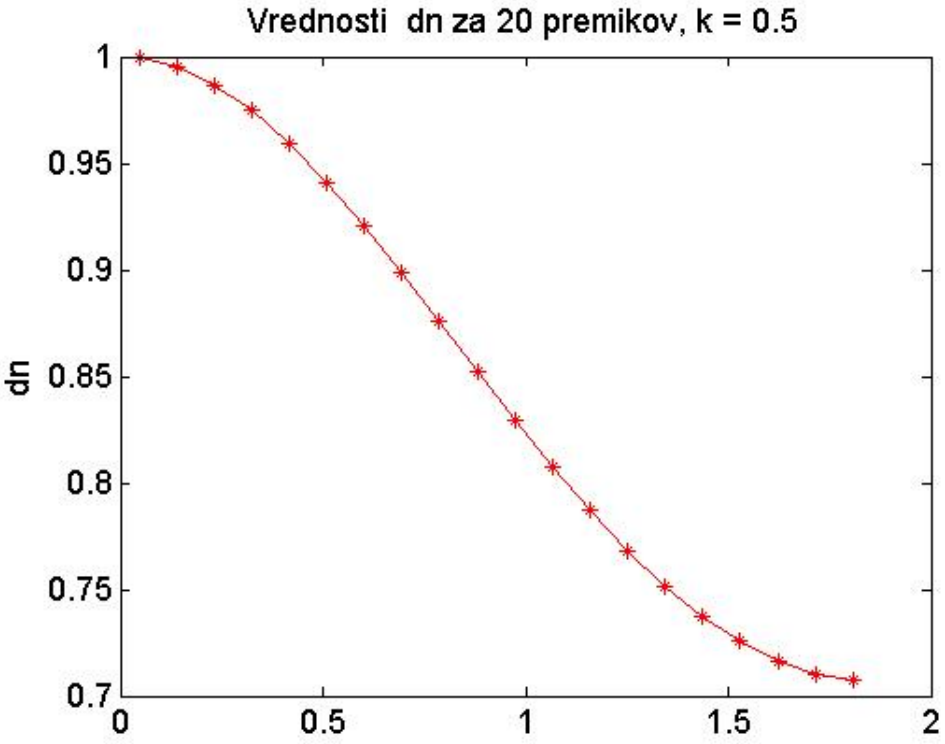
Algoritem 7: Izračun premikov s pomočjo eliptičnih integralov

---

VHOD: Vrednosti a, b, c, d za intervala $[a,b]$ in $[c,d]$, $b, d < 0$ in število premikov i
IZHOD: Premiki $\{\alpha_r\}$ in $\{\beta_r\}$, ki rešijo (66)
$\eta = 2(b-a)(d-c)/((a+b)(c+d))$

$k' = 1/(1+\eta+\sqrt{\eta(\eta+2)})$
if$(a = c$ **and** $b = d)$
    g=0
else
    $g = (2k'(b+d) - (a+c))/((a+c)(b-d) + k'(b+d)(c-a))$
end if
$f = (2 + g(b-d))/(b+d)$
$h = k'(c - a + 2acg)/(a+c)$
for $r = 1, 2, \ldots, i$
    $\omega_j = \mathrm{dn}((2r-1)K/2i)$, z modus $k = \sqrt{1-(k')^2}$
    $\alpha_r = -(\omega_r + h)/(f + g\omega_r)$
    $\beta_r = -(\omega_r - h)/(f - g\omega_r)$
end for

Podroben pregled različnih pristopov k iskanju dobrih premikov najdemo v [16]. Na sliki je prikazan izbor 20 premikov za vrednost $k = 0.5$.



Slika 3. Prikaz premikov

5. Numerični izračuni in primerjava

V tem poglavju si bomo ogledali implementacijo premikov in metod ADI iz poglavij 3 in 4. Metode so implementirane v Matlabu. Primeri so vzeti iz spletne zbirke Slicot [3], ki jo sestavlja več različnih primerov za redukcijo LTI sistemov. Več o zbirki najdemo v [3], kjer najdemo tudi uporabljene podatke.

Prvi primer, ki ga bomo podrobneje pogledali je primer heat-cont. Primer predstavlja diskretizacijo parcialne diferencialne enačbe za prevajanje toplote v eni dimenziji. Podani so podatki za delno diskretiziran LTI sistem in popolnoma diskretiziran sistem. Uporabili smo delno diskretiziran primer. V tem primeru imamo simetrično tridiagonalno matriko A dimenzije 200×200, B in C^T sta vektorja. Primer smo izbrali zaradi simetričnosti matrike A.

Drug primer, na katerem smo uporabili metode ADI, je fom. Primer fom predstavlja večji dinamični sistem, kjer je matrika A dimenzije 1006×1006, B in C^T sta ponovno vektorja. LTI sistem je podan z matrikami

$$A = \begin{bmatrix} A_1 & & & \\ & A_2 & & \\ & & A_3 & \\ & & & A_4 \end{bmatrix}, \; A_1 = \begin{bmatrix} -1 & 100 \\ -100 & -1 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} -1 & 200 \\ -200 & -1 \end{bmatrix}, \; A_3 = \begin{bmatrix} -1 & 400 \\ -400 & -1 \end{bmatrix}$$

$$\text{in } A_4 = \operatorname{diag}(-1, -2, \ldots, -1000).$$

Vektorja B in C^T sta enaka

$$B^T = C = [\underbrace{10 \ldots 10}_{6} \; \underbrace{1 \ldots 1}_{1000}].$$

Iz definicije matrike A lahko vidimo, da $\sigma(A)$ vsebuje kompleksne lastne vrednosti.

Za premike smo implementirali dva algoritma. Prvi za matriki A in B iz Sylvestrove enačbe (5) poišče $2k$ Ritzevih vrednosti s pomočjo Arnoldijevega algoritma, izbere $\min\{2k, k + 10\}$ Ritzevih vrednosti z najmanjšo napako in jih potem uredi. Arnoldijev algoritem izvedemo na slučajnem začetnem vektorju. Če je nabor Ritzevih vrednosti iz Arnoldijevega algoritma vseboval kakšno nenegativno vrednost, smo nabor zavrgli in ponovno zagnali Arnoldijev algoritem na novem slučajnem vektorju. Tem premikom pravimo heuristični premiki.
Drugi algoritem s pomočjo eliptičnih integralov poračuna optimalne premike, če je A simetrična oziroma hermitska matrika. Algoritem je implementiran tako kot je opisano v algoritmu 8.
Za primerjavo hitrosti smo poračunali tudi primer z enim samim premikom za Smithovo in kvadrirano Smithovo metodo. Premik je geometrična sredina največje in najmanjše lastne vrednosti matrike A. Vrednost premika je enaka, kot če bi poračunali premik z eliptičnimi funkcijami. Za primerjavo smo, kjer je bilo to možno, poračunali lastne vrednosti z Matlabovo funkcijo eig, saj v primeru velikih matrik

te primerjave ni možno narediti.

Za merjenje napake med rešitvijo enačbe in približkom, ki ga dobimo s katerokoli izmed opisanih metod, smo uporabili normalizirano normo ostanka.

Definicija 5.1. *Za aproksimacijo X, ki aproksimira rešitev Sylvestrove enačbe (5), definiramo* ***normalizirano normo ostanka*** *(ang. normalized residual norm) kot*

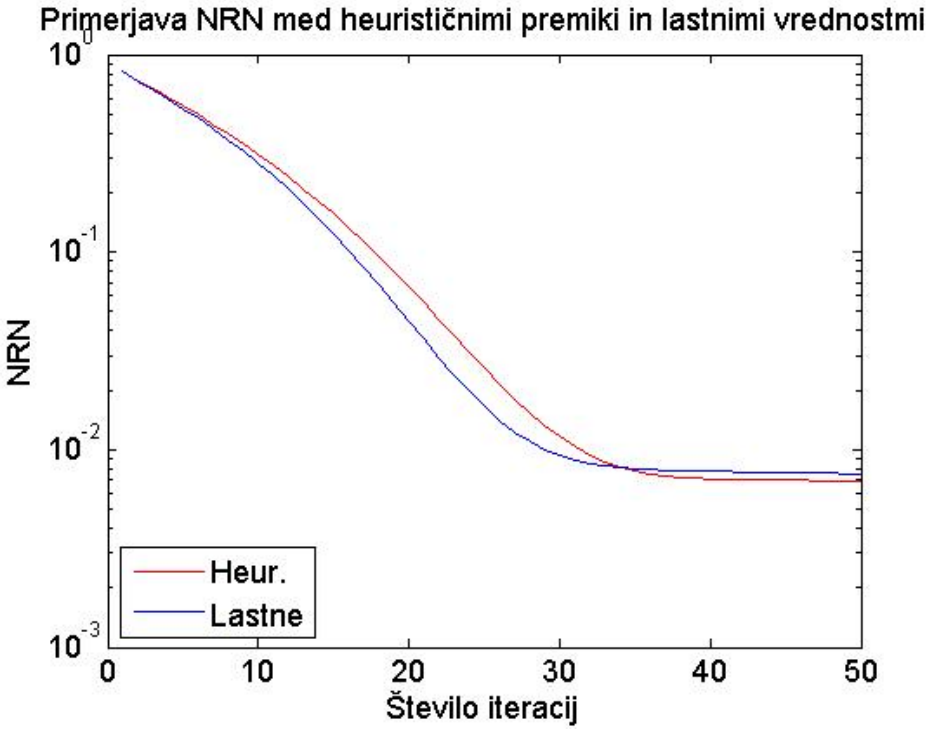
$$\mathrm{NRN}(X) = \frac{\|AX + XB - C\|_F}{\|C\|_F}. \tag{135}$$

Algoritmi so implementirani tako, da izvedejo vnaprej predpisano število korakov. Alternativno bi lahko kot v [9] implementirali tudi naslednje zaustavitvene pogoje:

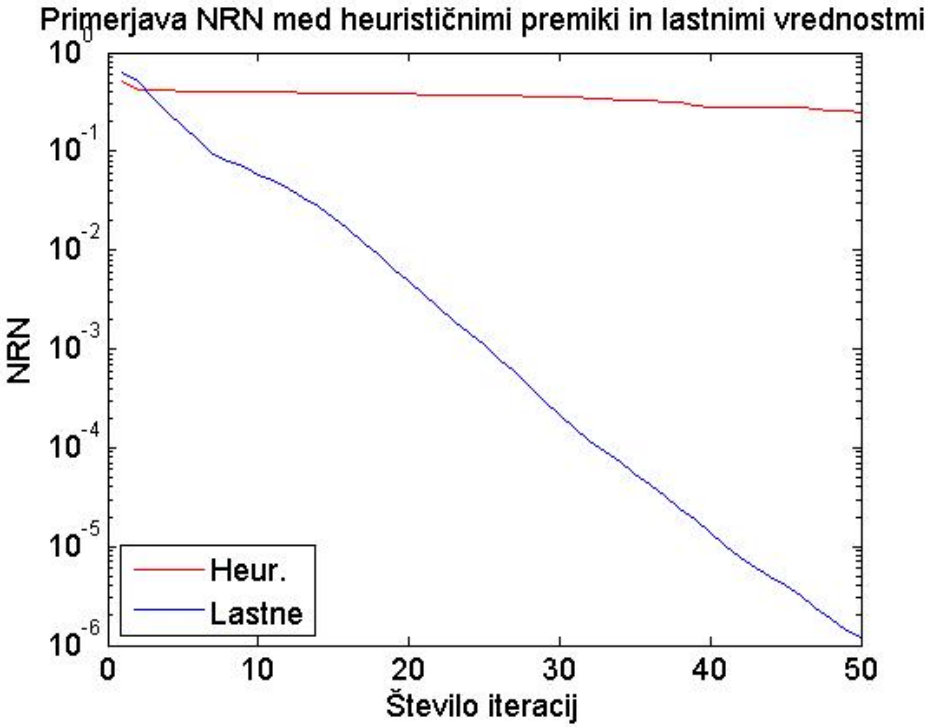
(1) minimalno toleranco za normalizirano normo ostanka,
(2) minimalno toleranco za spremembo normalizirano normo ostanka iz koraka r na $r + 1$,
(3) minimalno normo novega bloka $Z^{(r)}$ (npr. za LRCF ADI) oziroma minimalno toleranco za norme posameznih novih blokov (npr. pri fADI in modifikaciji fADI).

Zgornji zaustavitveni pogoji zahtevajo, da na vsakem koraku izračunamo normalizirano normo ostanka.

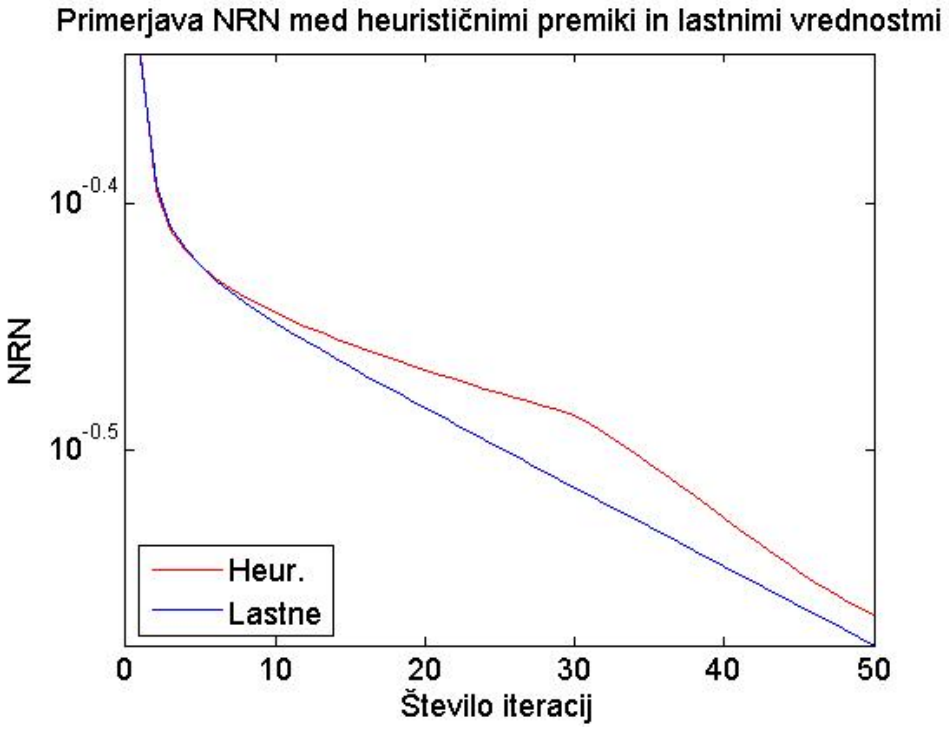
Poglejmo primerjavo NRN, ko za premike vzamemo lastne vrednosti urejene tako, da dosežejo čim manjši NRN, in med heurističnimi premiki. Ker so primeri majhni, smo poračunali vse lastne vrednosti za dan sistem, jih uredili in nato izbrali prvih k, ki zmanjšajo minimaks problem. Dobljene nabore vrednosti smo nato uporabili za premike.



Slika 4. Primerjava med heurističnimi premiki in lastnimi vrednostmi, primer heat-cont.



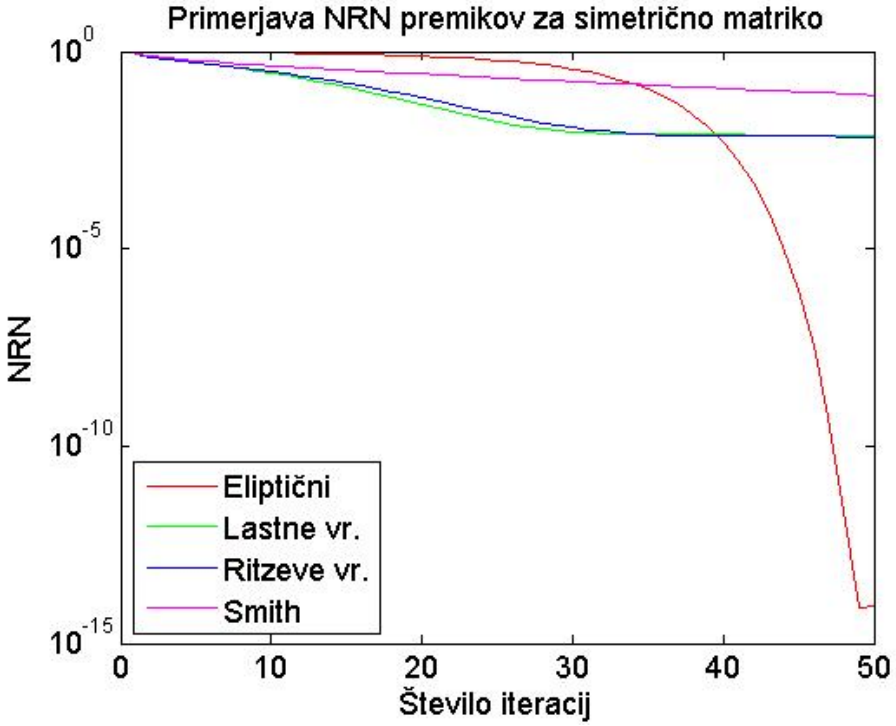
Slika 5. Primerjava med heurističnimi premiki in lastnimi vrednostmi, primer eady.



Slika 6. Primerjava med heurističnimi premiki in lastnimi vrednostmi, primer fom.

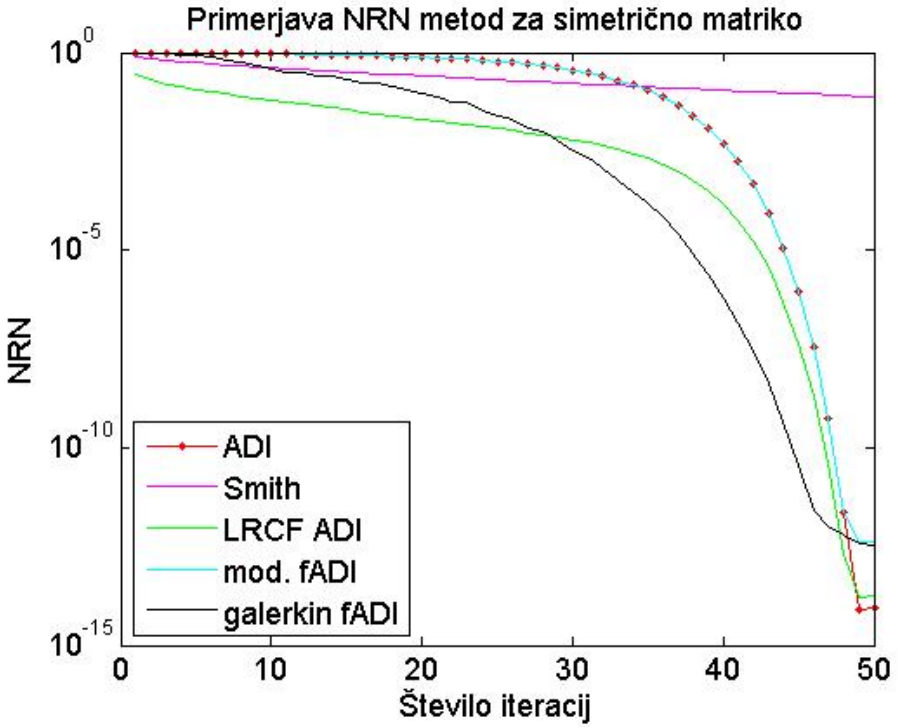
Slike 4, 5 in 6 prikazujejo primerjavo premikov za primere heat-con, eady in fom. V vseh primerih smo iskali rešitev enačbe Ljapunova za matriki A in $Q = BB^T$,

kjer sta matrika A in vektor B vzeti iz danega primera. Grafi prikazujejo razliko v NRN, ki jo dobimo zaradi zamenjave lastnih vrednosti z Ritzevimi vrednostmi. Iz grafov je razvidno, da Ritzeve vrednosti velikokrat dajo podobne rezultate kot lastne vrednosti, včasih, kar prikazuje graf 5, pa NRN za Ritzeve vrednosti pada bistveno počasneje.



Slika 7. Primerjava med različnimi izbirami premikov za aproksimacijo vodljivostne Gramove matrike, primer heat-cont.

Slika 7 prikazuje primerjavo premikov na podatkih heat-cont, kjer sta podana matrika A in vektor B. V vseh primerih smo reševali enačbo Ljapunova, kjer smo iskali vodljivostno Gramovo matriko za sistem določen z A in B. V tem primeru je matrika A simetrična negativno definitna in imamo izpolnjene pogoje, ki nam omogočajo oceno premikov z eliptičnimi funkcijami. Primerjali smo premike, ki smo jih dobili z eliptičnimi funkcijami, Ritzeve vrednosti, premik za Smithovo metodo in lastne vrednosti. Za vsak premik smo uporabili metodo ADI, tako da lahko lažje primerjamo rezultate. Po 50 korakih je najmanj zmanjšala NRN izbira premika za Smithovo metodo. Ritzeve in lastne vrednosti so dosegle zmerno zmanjšanje NRN. Ko smo za premike vzeli lastne vrednosti je bila končna NRN 0.00756, ko smo za premike vzeli Ritzeve vrednosti pa 0.00697.
Iz grafa je razvidno, da so se znatno bolje obnesli premiki, ki smo jih poračunali z eliptičnimi funkcijami, za katere je metoda skonvergirala, če smo izbrali tolerančno mejo večjo od 10^{-14}.
Poglejmo še primerjavo metod na primeru heat-cont.
Graf 8 prikazuje primerjavo Smithove metode, metode ADI, metode LRCF-ADI in modificirane metode fADI. Dobljene rezultate smo primerjali tudi z rezultati Smithove kvadrirane metode, ki niso prikazani na grafu. V primeru Smithove in kvadrirane Smithove metode smo za premik vzeli $-\sqrt{ab}$, kjer sta a največja lastna vrednost A in b največja lastna vrednost A^{-1}. Lastni vrednosti smo poračunali z Matlabovo

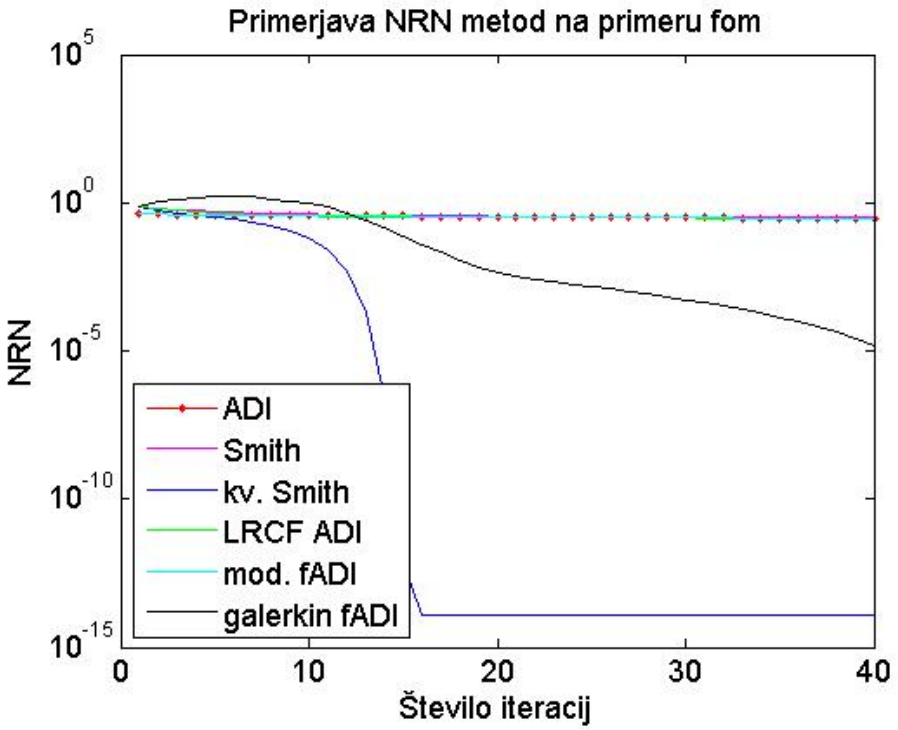


Slika 8. Primerjava med metodami za aproksimacijo vodljivostne Gramove matrike, primer heat-cont.

funkcijo eigs. Za preostale metode smo vzeli premike, dobljene z eliptičnimi funkcijami, saj so se ti najbolje obnesli pri izbiri parametrov.
Znatno najboljši rezultat smo dobili s Smithovo kvadrirano metodo, ki skonvergira v 10 korakih, če zahtevamo tolerančno mejo večjo od 10^{-12}. Kljub rezultatom Smithove kvadrirane metode še vedno ne priporočamo za zelo velike matrike, kjer je potrebnih več iteracij, saj vsaka nadaljnja iteracija pomeni več računskih operacij zaradi zgoščevanja matrik. Za metode ADI, LRCF-ADI, Galerkin fADI in modificirano metodo fADI dobimo primerljive rezultate. Iz grafa je razvidno tudi, da najbolj enakomerno NRN pada pri metodi podprostorov ADI z Galerkinovim pogojem.

Graf 9 prikazuje primerjavo metod na primeru fom. V tem primeru niso bile izpolnjene predpostavke za izbiro premikov, opisano v algoritmu 8, zato smo namesto le-teh uporabili Ritzeve vrednosti. Za Smithovo in kvadrirano Smithovo metodo smo poračunali premik na enak način kot pri prejšnjem grafu. Najboljši rezultat smo dobili s kvadrirano Smithovo metodo. Iz grafa 9 je razvidno tudi, da tako za metode ADI kot tudi za Smithovo metodo NRN pada zelo počasi. Izjema je metoda podprostorov ADI, za katero NRN pada veliko hitreje kot za ostale metode, razen za Smithovo kvadrirano metodo.

Na primeru heat-cont smo za iskanje vodljivostne Gramove matrike naredili tudi časovno primerjavo. Primerjali smo Matlabovo funkcijo lyap, ki za reševanje Ljapunove enačbe uporablja direkten, algoritem z implementiranimi algoritmi. Rezultati so povzeti v tabeli 1. Za izračun smo uporabili $k = 30$ iteracij.
Iz tabele je razvidno, da je za probleme manjših dimenzij količina časa med direktnimi metodami in metodo LRCF-ADI primerljiva, če imamo dober nabor premikov.



Slika 9. Primerjava med metodami za aproksimacijo vodljivostne Gramove matrike, primer mod.

Tabela 1. Primerjava porabljenega časa za aproksimacijo X

Primerjava	Čas v sekundah	NRN
Funkcija lyap()	0.059872	NA
Heur. premiki + ADI	0.270843	0.051106338668110
Samo ADI	0.181052	0.051106338668110
Elip. premiki + ADI	0.412170	0.000000000005101
Samo LRCF-ADI	0.064797	0.000000000005100

V nasprotnem primeru, ko imamo matrike manjših dimenzij, je bolje uporabiti direktne metode.

6. Zaključek

V magistrski nalogi smo obravnavali, kako je enačba Ljapunova kot poseben primer Sylvestrove enačbe povezana s teorijo upravljanja linearnih kontrolnih sistemov. Nadaljnje smo obravnavali metodo ADI za reševanje Sylvestrove enačbe in pokazali, da je konvergenca metode odvisna od izbire premikov. Obravnavali smo več metod, ki izvirajo iz metode ADI. Predstavili smo predhodnico metode ADI, Smithovo metodo, kot tudi kvadrirano Smithovo metodo. Za reševanje enačbe Ljapunova smo predstavili metodo ADI nizkega ranga s faktorji Choleskega. Nato smo obravnavali faktorizirano metodo ADI, ki predstavlja razširitev metode ADI nizkega ranga s faktorji Choleskega na Sylvestrovo enačbo. Pokazali smo, kako lahko zmanjšamo časovno in prostorsko zahtevnost na vsakem koraku iteracije s pomočjo modifikacije faktorizirane metode ADI in izboljšamo njeno konvergenco z metodo podprostorov ADI, kjer smo uporabili Galerkinov pogoj. Nato smo predstavili problem izbire primernih premikov za metode ADI. Podali smo nekaj ocen za konvergenco metode ADI in pokazali, kako lahko poiščemo premike heuristično. Pokazali smo tudi, kako najdemo najboljšo izbiro premikov, ko sta matriki A in B hermitski in stabilni matriki. V empiričnem delu smo uporabili implementacije algoritmov, ki smo jih predstavili v teoretičnem delu, na primerih iz zbirke Slicot. Metode smo primerjali z normalizirano normo ostanka. Naredili smo primerjavo premikov na primeru heat-cont iz zbirke Slicot in naredili primerjavo metod na primerih heat-cont in fom.
Raziskovanje ADI se je v zadnjem desetletju razširilo na reševanje matričnih enačb, kot so zvezna algebraična Ricattijeva enačba, s katero se srečamo pri optimalnem vodenju linearnih kontrolnih sistemov, posplošena Sylvestrova enačba ter posplošena enačba Ljapunova, s katerima se srečamo pri preučevanju deskriptorskih sistemov.
Ključna problematika metod ADI ostaja učinkovita izbira premikov, ki bi za splošne matrike zagotovila hitro konvergenco.

Literatura

[1] A. C. Antoulas, D. C. Sorensen, Y. Zhou, *On the decay of Hankel singular values and related issues*, Systems and Control Letters 46 (2002) 323–342.

[2] P. Brenner, N. Truhar, R-C. Li, *On ADI method for Sylvester equations*, Journal of Computational and Applied Mathematics 233 (2009) 1034–1045.

[3] Y. Chahlaoui, P. Van Dooren, *A collection of benchmark examples for model reduction of linear time invariant dynamical systems*, SLICOT Working Note 2002-2, 2002.
Dosegljivo na: `http://slicot.org/objects/software/reports/SLWN2002-2.ps.gz` [Dostop: 4. november 2018].

[4] L. Fortuna, M. Frasca, *Optimal and Robust Control Advanced Topics with MATLAB*, CRC Press, Florida, 2012, 81–85.

[5] M. Gutknecht, *Block Krylov space methods for linear systems with multiple right-hand sides: An introduction*, verzija 28. februar 2006.
Dosegljivo na: `http://www.sam.math.ethz.ch/~mhg/pub/delhipap.pdf` [Dostop: 13. 12 2018].

[6] Z. Jia, *A refined iterative algorithm based on the block Arnoldi process for large unsymmetric eigenproblems*, Linear Algebra and its Applications 270 (1998) 171–189.

[7] J. Li, J. White, *Low rank solution of Lyapunov equations*, SIAM Journal on Matrix Analysis and Applications 24 (2002) 260–280.

[8] A. Megretski, *Model Reduction*, MIT, 2004.
Dosegljivo na: `web.mit.edu/6.242/www/images/lec5_6242_2004.pdf` [Dostop: 12. avgust 2018].

[9] T. Penzl, *LYAPACK: A MATLAB Toolbox for Large Lyapunov and Riccati Equation, Model Reduction Problems, and Linear Quadratic Optimal Control Problems*, SFB 393 Fakultät für Mathematik, TU Chemnitz, 1999.
Dosegljivo na: `https://www.tu-chemnitz.de/sfb393/lyapack/guide.pdf` [Dostop: 4. november 2018].

[10] T. Penzl, *Eigenvalue decay bounds for solutions of Lyapunov equations: the symmetric case*, Systems and Control Letters 40 (2000) 139–144.

[11] T. Penzl, *A cyclic low-rank Smith method for large sparse Lyapunov equations*, SIAM Journal on Scientific Computing 21 (2000) 1401–1418.

[12] T. Penzl, *Algorithms for model reduction of large dynamical systems*, Linear Algebra and its Applications 415 (2006) 322–343.

[13] B. Plestenjak, *Numerične metode za linearne sisteme upravljanja*, skripta, verzija: 3. april 2012.
Dosegljivo na: `https://www.fmf.uni-lj.si/~plestenjak/Vaje/NMLSU/skriptanmlks.pdf`, [Dostop: 20. julij 2018].

[14] B. Plestenjak, *Iterativne numerične metode v linearni algebri*, skripta, verzija: 5. februar 2018.
Dosegljivo na: `https://www.fmf.uni-lj.si/~plestenjak/Vaje/INMLA/Predavanja/Skripta.pdf`, [Dostop: 13. december 2018].

[15] R. Smith, *Matrix equation* $XA + BX = C$, SIAM Journal on Applied Mathematics 16 (1968) 198–201.

[16] J. Sabino, *Solution of Large-Scale Lyapunov Equations via the Block Modified Smith Method*, doktorska disertacija, Rice University (2007).

[17] N. Truhar, Z. Tomljanović, R-C. Li, *Analysis of the solution of the Sylvester equation using low-rank ADI with exact shifts*, Systems and Control Letters 59 (2010) 248–257.

[18] E. L. Wachspress, *Extended application of alternating direction implicit iteration model problem theory*, Journal of the Society for Industrial and Applied Mathematics 11 (1963) 994–1016.

[19] E. L. Wachspress, *Iterative solution of the Lyapunov matrix equation*, Applied Mathematics Letters 1 (1988) 87–90.

[20] E. Wachspress, *The ADI Model Problem*, Springer-Verlag, Berlin, 2013.