

Iterirani Ritzov postupak: počela, trenutačno stanje, budući razvoj

Josip Dvornik¹, Damir Lazarević¹, Antonia Jaguljnjak Lazarević²

¹Sveučilište u Zagrebu, Građevinski fakultet

²Sveučilište u Zagrebu, Rudarsko – geološko – naftni fakultet

Sažetak

U članku je opisan razvoj novoga iteracijskog postupka za rješavanje sustava linearnih algebarskih jednadžbi, utemeljen na primjeni diskretnoga Ritzova postupka u svakom koraku. Pogodan je za izrazito velike sustave slabo popunjenih, čak loše uvjetovanih, matrica. Osim vlastitih obilježja posjeduje i svojstvo općenitosti, jer su mnogi iteracijski postupci samo poseban slučaj ovoga pristupa. To pomaže drugačijoj interpretaciji tih postupaka, što doprinosi razumijevanju njihovih prednosti i nedostataka, a time i zamisliva poboljšanja. Algoritam je realiziran samostalno, a potom je pridružen programu otvorena koda FEAP. Provedene su raznolike provjere, posebice na praktičnim modelima. Postupak je tek djelomice istražen, ali već pokazuje dobre rezultate.

Ključne riječi: Ritzov postupak, metoda konjugiranih gradijenata, postupak uzastopne prekomjerne relaksacije, preduvjetovanje, točna aritmetika

Iterated Ritz Method: fundamentals, current state and future development

Abstract

Development of a novel iterative solver for linear systems of algebraic equations, based on a discrete Ritz method, is described. It is suitable for extremely large, even ill-conditioned, sparse systems. In addition to its own characteristics, it also has a feature of generality, as many iterative methods are only special cases of this approach. It also helps in different interpretation of these methods, which contributes to understanding of their advantages and limitations, and hence gives motivation for their improvement. The algorithm was developed independently, and then implemented into the open source finite element program FEAP. Also, various checks were conducted, especially on practical models. Although the method has been only partially studied, good results have already been obtained.

Key words: Ritz method, conjugate gradient method, Method of successive overrelaxation, preconditioning, exact arithmetic

1. Kratki teorijsko – povijesni uvod

Poznato je, rješavanje realnoga, linearnog sustava

$$\mathbf{Ax} = \mathbf{b} \quad (1)$$

sa simetričnom i pozitivno definitnom (SPD) matricom, ekvivalentno je minimizaciji pripadajućega funkcionala energije (kvadratne forme)

$$f(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{Ax} - \mathbf{x}^T \mathbf{b}. \quad (2)$$

Godine 1978. J. Dvornik je objavio iteracijski postupak minimizacije izraza utemeljen na rješavanju maloga sustava jednadžbi unutar svakoga koraka [1]. Ukratko, ako prirast rješenja u (i)-tom koraku prikažemo diskretiziranim Ritzovim postupkom

$$\mathbf{p}_{(i)} = \Phi_{(i)} \mathbf{a}_{(i)} \quad (3)$$

gdje je $\Phi_{(i)} = [\phi_{1,(i)} \phi_{2,(i)} \dots \phi_{m,(i)}]$ matrica linearno nezavisnih koordinatnih vektora, a $\mathbf{a}_{(i)}$ vektor pripadajućih koeficijenata, smanjenje energije nakon prirasta (3) također je kvadratna forma

$$\Delta f(\mathbf{a}_{(i)}) = \frac{1}{2} \mathbf{a}_{(i)}^T \bar{\mathbf{A}}_{(i)} \mathbf{a}_{(i)} - \mathbf{a}_{(i)}^T \bar{\mathbf{r}}_{(i)}, \quad (4)$$

gdje je $\bar{\mathbf{A}}_{(i)} = \Phi_{(i)}^T \mathbf{A} \Phi_{(i)}$ poopćena (Ritzova) SPD matrica, a $\bar{\mathbf{r}}_{(i)} = \Phi_{(i)}^T \mathbf{r}_{(i)}$ poopćeni (Ritzov) vektor reziduala, oboje reda m . Minimizacijom (4) dobivamo sustav jednadžbi

$$\bar{\mathbf{A}}_{(i)} \mathbf{a}_{(i)} = \bar{\mathbf{r}}_{(i)}, \quad (5)$$

koji treba riješiti u svakom koraku. Rješenjem odredimo prirast (3), a potom i trenutačnu aproksimaciju $\mathbf{x}_{(i+1)} = \mathbf{x}_{(i)} + \omega_{(i)} \mathbf{p}_{(i)}$, gdje je $\omega_{(i)} \in (0,2)$ faktor relaksacije, poznat iz postupka uzastopne prekomjerne relaksacije (engl. *successive overrelaxation*) koji može ubrzati konvergenciju. Određivanje optimalnoga $\omega_{(i)}$ u svakom koraku nije isplativo (koraka je manje, ali predugo traju), pa se prema numeričkim pokusima i iskustvu bira konstantan iznos na početku proračuna. Rezidual obično definiramo rekursivno kao $\mathbf{r}_{(i+1)} = \mathbf{r}_{(i)} + \omega_{(i)} \mathbf{A} \mathbf{p}_{(i)}$ i povremeno ga (svakih k koraka), zbog gomilanja pogrešaka zaokruživanja, obnavljamo iz uvjeta ravnoteže $\mathbf{r}_{(i+1)} = \mathbf{b} - \mathbf{A} \mathbf{x}_{(i)}$. Iteracijski slijed završava ako je $\|\mathbf{r}_{(i)}\|_2 \leq \varepsilon \|\mathbf{r}_{(0)}\|_2$, gdje je \mathbf{r}_0 početni rezidual ($\mathbf{r}_0 = \mathbf{b} - \mathbf{A} \mathbf{u}_0 = \mathbf{0}$), a ε je vrlo mali pozitivan broj.

Znači, u svakom koraku tvorimo koordinatne vektore koji razapinju potprostor unutar kojega smanjujemo energiju sistema. Zato treba riješiti sustav (5). Ako je $\omega = 1$, riječ je o najvećem smanjenju (lokalnom minimumu), što za globalnu konvergenciju

nije nužno optimalno. Dimenzija sustava je mala, puno manja od broja nepoznanica (n), jer ga nekim (najčešće izravnim) postupkom treba riješiti u svakom koraku. Time postupno reduciramo energiju (2), do pronalaska globalnoga minimuma. Očito je riječ o kombinaciji iteracijskoga (preciznije, gradijentnog) i izravnoga postupka rješavanja sustava. Pseudokod je prikazan u nastavku.

Iterirani Ritzov postupak

- 1: **Potrebno:** \mathbf{A} , \mathbf{b} , ω , k , ε
 - 2: **Rezultat:** $\mathbf{x}_{(i)}$ približno rješenje
 - 3: $i \leftarrow 0$ brojač koraka
 - 4: $\mathbf{x}_{(i)} \leftarrow \mathbf{0}$ početno rješenje nul-vektor
 - 5: $\mathbf{r}_{(i)} \leftarrow \mathbf{b}$ rezidual jednak opterećenju
 - 6: **repeat**
 - 7: $\Phi_{(i)} \leftarrow [\phi_{1,(i)} \ \phi_{2,(i)} \ \dots \ \phi_{m,(i)}]$ definiranje koordinatnih vektora
 - 8: $\bar{\mathbf{A}}_{(i)} \leftarrow \Phi_{(i)}^T \mathbf{A} \Phi_{(i)}$ tvorba „male” matrice sustava
 - 9: $\bar{\mathbf{r}}_{(i)} \leftarrow \Phi_{(i)}^T \mathbf{r}_{(i)}$ tvorba „maloga” vektora desne strane
 - 10: $\mathbf{a}_{(i)} \leftarrow \bar{\mathbf{A}}_{(i)}^{-1} \bar{\mathbf{r}}_{(i)}$ rješavanje „maloga” sustava
 - 11: $\mathbf{p}_{(i)} \leftarrow \Phi_{(i)} \mathbf{a}_{(i)}$ određivanje prirasta rješenja
 - 12: $\mathbf{x}_{(i+1)} \leftarrow \mathbf{x}_{(i)} + \omega \mathbf{p}_{(i)}$ proračun novoga pomaka
 - 13: **if** $i \bmod k \neq 0$ **then**
 - 14: $\mathbf{r}_{(i+1)} \leftarrow \mathbf{r}_{(i)} - \omega \mathbf{A} \mathbf{p}_{(i)}$ rezidual rekurzivno
 - 15: **else**
 - 16: $\mathbf{r}_{(i+1)} \leftarrow \mathbf{b} - \mathbf{A} \mathbf{x}_{(i+1)}$ rezidual iz ravnoteže
 - 17: **end if**
 - 18: $i \leftarrow i + 1$ povećanje brojača koraka
 - 19: **until** $\|\mathbf{r}_{(i)}\|_2 / \|\mathbf{r}_{(0)}\|_2 < \varepsilon$
-

U radu [1] istaknuta je i općenitost algoritma. Pokazano je, postupci Gauß–Seidela, najstrmijega silaska i konjugiranih gradijenata mogu se interpretirati kao posebni slučajevi pristupa. Predloženo je nekoliko načina tvorbe koordinatnih vektora, napravljene su usporedbe među postupcima, a riješeni su i karakteristični primjeri. Rad je zaključen osvrtom na mogućnosti rješavanja nelinearnih sustava. Članak je

1979. godine u izvornom obliku otisnut u časopisu *Computers & Structures* [2]. Početak prve stranice vidi se na sljedećoj slici.

Computers & Structures Vol. 10, pp. 217-223
© Pergamon Press Ltd., 1979. Printed in Great Britain

0045-7949/79/0401-0217/\$02.00/0

GENERALIZATION OF THE CG METHOD APPLIED TO LINEAR AND NONLINEAR PROBLEMS

JOSIP DVORNIK

Gradjevinski Institut, Janka Rakuse 1, 41000 Zagreb, Yugoslavia

(Received 19 May 1978)

Abstract—A method of solving a set of linear equations through iteration process is described. In every step of the process the Ritz method is applied. With a suitably chosen procedure for generating coordinate vectors, the process is efficient when applied to nonlinear and in some cases even to linear problems. Present experience is limited, and no objective criteria have been developed for an *a priori* judgment of coordinate vectors what would very probably contribute to the efficiency. Some standard iterative algorithms can be interpreted as special cases of this procedure.

Na žalost, rad je praktički ostao bez odjeka. Jedino je, prema šturim podacima i saznanjima J. Dvornika, osamdesetih godina postupak primijenilo nekoliko istraživačkih ustanova, koje nisu publicirale područje i način primjene postupka. Prema našim saznanjima, članak je tih godina citiran tri puta [3-5].

Početak devedesetih matematičar S. Polić radio je na volumnom pristupu pločama umjerene debljine primjenom metode konačnih razlika. Dobivao je (za tadašnje prilike) velike sustave, široke vrpce, pa su 1994. godine za njegove potrebe J. Dvornik i D. Lazarević napravili poboljšanu inačicu postupka. Tada je, primjenom štednoga zapisa matrice po redcima, riješeno pristojnih jednadžbi. Rezultati nikada nisu objavljeni. Početak programa koji je primijenio Polić priložen je u nastavku.

PROGRAM GCGMETHOD

```
c#####  
c   iterativni algoritam za rjesavanje velikih sustava  
c   linearnih algebarskih jednadzbi oblika 'Ax=b' sa  
c   simetricnom i pozitivno definitnom matricom sustava 'A'  
  
c   izradili:      Prof.dr. Josip Dvornik   dipl.ing.gradj.  
c                Damir Lazarevic dipl.ing.gradj.  
c#####
```

IMPLICIT DOUBLE PRECISION (A-H,O-Z)

```
c#####
c      NM - broj koordinatnih vektora
c      sve potrebne vektore smjestamo u
c      dinamički alocirani vektor 'L'
c#####
```

```
PARAMETER (NM=5, NMAX=100000000)
COMMON L(NMAX)
CHARACTER*12 NAME
```

```
WRITE(*,*)' upisi ime input file-a: '
READ(*,*) NAME
OPEN(1,FILE=NAME)
OPEN(2,FILE='gcsolve.par')
```

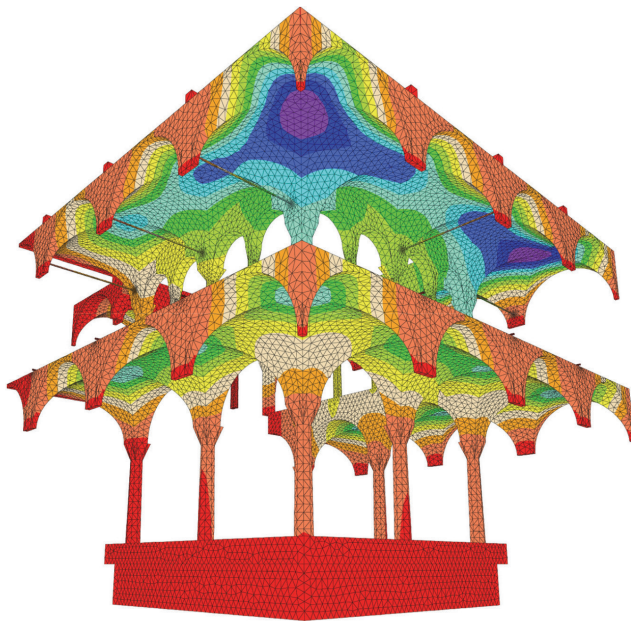
```
c#####
c      N - broj nepoznanica
c      K - broj elemenata gornje trokutaste matrice 'A'
c      razlicitih od nule
c#####
```

```
READ(1,*)N,K
WRITE(*,*)' upisi ime output file-a: '
READ(*,*) NAME
OPEN(3,FILE=NAME)
```

```
c#####
c      odredjivanje pocetaka svih vektora
c#####
```

```
I1=2*K+1
I2=I1+K
I3=I2+N+1
I4=I3+2*N
I5=I4+2*N
I6=I5+2*N
I7=I6+2*N
I8=I7+2*N*NM
I9=I8+2*N*NM
I10=I9+2*N
I11=I10+2*NM*NM
I12=I11+2*NM
I13=I12+2*NM
I14=I13+NM-1
```

Autori su se postupku vratili 2014. godine i na temelju dotadašnjih iskustava, ali i novih zamisli, 2015. godine dobili istraživački projekt Hrvatske zaklade za znanost *Novi, učinkoviti iteracijski postupak poračuna konstrukcija – poopćenje suvremenih postupaka*, u trajanju od četiri godine. Istraživanjima su se pridružili N. Bićanić (koji je iznenada preminuo 2016. godine), zatim A. Jaguljnjak Lazarević, K. Fresl, M. Uroš, P. Gidak, E. Šamec, M. Šavor Novak i M. Demšić. Poboljšani postupak, skraćena naziva IRM (od engl. *Iterated Ritz Method*), pridružen je programu otvorena koda za realizaciju metode konačnih elemenata FEAP [6, 7]. Sredstvima zaklade nabavljeno je moćno računalo pa su riješeni problemi reda veličine od [8], do čak nepoznanica [9]. Dio novijih, kontrolnih proračuna atrija *Kneževa dvora* u Dubrovniku napravljen je suvremenom inačicom postupka [10].



2. O izvornosti, konvergenciji i općenitosti postupka

Lako je uočiti prirodnu i ne odveć tešku zamisao IRM-a. Unatoč tomu, niti nakon četrdeset godina, pristup i interpretacija nisu naišli na širu primjenu među istraživačima koji se bave iteracijskim postupcima rješavanja sustava [11, 12].

Neslužbeno, prema još jednoj povijesnoj crtici, riječ je o gotovo pedeset godina. Naime, 1970. godine J. Dvornik prvi je put čuo o metodi konjugiranih gradijenata od matematičara N. Ivančića [13], s kojim je radio u tadašnjem *Institutu građevinarstva Hrvatske*. Nije bio zadovoljan svojim shvaćanjem postupka, pa ga je pokušao razjasniti na drugačiji, više interpretativan, inženjerski način. U traganju za zornijim

predočenjem zapazio je ekvivalenciju algoritma s rješavanjem sustava reda dva u svakom koraku, što se podudara s lokalnom minimizacijom energije u ravnini razapetoj s $\mathbf{r}_{(i)}$ i $\mathbf{p}_{(i-1)}$. Potom je uočio, i drugi osnovni iteracijski postupci mogu se slično interpretirati, a dovoljan je jedan vektor, rješavanje samo jedne jednadžbe u svakom koraku. Uvidjevši ove analogije postavio je logično pitanje: Ako imamo postupke s jednim ili dva koordinatna vektora, zašto ne pokušati s tri, četiri, pa i više? Tako je nastao IRM.

Postupak je samo površno sličan iteraciji po potprostorima u dinamici konstrukcija. Tamo niz potprostora konvergira prema potprostoru vlastitih vektora i nakon ispunjenja kriterija podudara se s približnim rješenjem. Kod IRM-a približno konvergentno rješenje vektor je izvan potprostora, a potprostor ne treba konvergirati. Drugim riječima, niz potprostora nema niti treba svojstvo konvergencije.

Za konvergenciju postupka dovoljan je (jedan) vektor potprostora koji nije ortogonalan na trenutni rezidual. To može biti baš $\mathbf{r}_{(i)}$ ili pomnožen nekom pozitivno definitnom matricom. Ostali vektori biraju se slobodno (i različito su učinkoviti), ali moraju biti linearno nezavisni. Uzima se i $\mathbf{p}_{(i-1)}$, doduše ortogonalan na rezidual, ali doprinosi brzini konvergencije. Poznat je od ranije, pa je "besplatan", za razliku od ostalih vektora koje treba nekako generirati. Ako je proračun konvergentan, suma rješenja malih sustava teži prema rješenju velikoga sustava, a suma prirasta energije malih sustava monotono pada i teži prema minimumu energije (2) velikoga sustava. Prema matematičkoj interpretaciji, postupak bi spadao u skup projekivnih postupaka [14, 15] kod kojih je projekcija -dimenzijskog prostora na tzv. potprostor pretraživanja (engl. *search subspace*) dimenzije definirana uvođenjem dodatnih uvjeta. Najčešće je riječ o ograničenju Petrov–Galerkin: ortogonalizaciji reziduala na linearno nezavisnih vektora koji tvore tzv. potprostor ograničenja (engl. *subspace of constraints*). Potprostori pretraživanja i ograničenja mogu se podudarati, pa je riječ o ortogonalnom projekivnom postupku.¹ To je slučaj IRM-a, ali samo ako je $\omega = 1$. Izvodom IRM-a iz Ritzova rastava (3), uz energijsku interpretaciju iteracije, uvjet ortogonalnosti reziduala na potprostor nije potreban – proizlazi izravno iz postupka [8]:

$$\begin{aligned}
 \Phi_{(i)}^T \mathbf{r}_{(i+1)} &= \Phi_{(i)}^T (\mathbf{b} - \mathbf{A} \mathbf{x}_{(i+1)}) \\
 &= \Phi_{(i)}^T \mathbf{b} - \Phi_{(i)}^T \mathbf{A} (\mathbf{x}_{(i)} + \mathbf{p}_{(i)}) \\
 &= \Phi_{(i)}^T \mathbf{b} - \Phi_{(i)}^T \mathbf{A} \mathbf{x}_{(i)} - \Phi_{(i)}^T \mathbf{A} \mathbf{p}_{(i)} \\
 &= \Phi_{(i)}^T (\mathbf{b} - \mathbf{A} \mathbf{x}_{(i)}) - \Phi_{(i)}^T \mathbf{A} \Phi_{(i)} \mathbf{a}_{(i)} \\
 &= \Phi_{(i)}^T \mathbf{r}_{(i)} - \bar{\mathbf{A}}_{(i)} \mathbf{a}_{(i)} = \Phi_{(i)}^T \mathbf{r}_{(i)} - \bar{\mathbf{r}}_{(i)} \\
 &= \Phi_{(i)}^T \mathbf{r}_{(i)} - \Phi_{(i)}^T \mathbf{r}_{(i)} = \mathbf{0}.
 \end{aligned} \tag{6}$$

1 U protivnom, projekivni postupak nazivamo posrednim (engl. *oblique*).

Ako je $\omega \neq 1$, ortogonalnost ne vrijedi. U člancima o projektivnim postupcima raspravlja se o konvergenciji, uz određena svojstva sustava i potprostora, ali se za tvorbu koordinatnih vektora upotrebljavaju rekurzije i ortogonalizacije (općenito transformacije), kojima se nepotrebno sužava podskup dobrih vektora i izbjegava rješavanje maloga sustava. Time, prema našim numeričkim testovima, iskustvu na praktičnim primjerima i strogim dokazima na posebnim slučajevima, postupci postaju manje stabilni od IRM-a. Za opći dokaz ove tvrdnje (i stabilnosti IRM-a općenito), trebalo bi tražiti pomoć matematičara.

Ovisno o izboru matrice $\Phi_{(i)}$ razlikujemo brojne iteracijske postupke. Analogije ne služe bržoj realizaciji tih postupaka u odnosu na uvriježene načine [16], nego doprinose njihovu pojašnjenju i ističu općenitost ovoga iteracijskog algoritma. I jednostruko ili višestruko preduvjetovanje (engl. *preconditioning*) sustava [17, 18] možemo interpretirati primjenom (jednoga ili više) koordinatnih vektora [8, 9]. Za IRM preduvjetovanje ne znači nikakvu bitnu promjenu algoritamskoga slijeda.

3. Sve je u koordinatnim vektorima

Mi bismo htjeli odrediti matricu $\Phi_{(i)}$ tako da postupak bude brži od postojećih. Premda želimo da broj koordinatnih vektora m ostane malen, čini nam se da je samo jedan vektor (kao kod Jacobija, Gauß – Seidela, uzastopnoga prekoračenja i najstrmijega silaska) ili dva (kao kod konjugiranih gradijenata) manje od optimuma. Jasno, broj je vektora puno manji od n , pa približni prirast $\mathbf{p}_{(i)}$ može u ranoj fazi postupka samo s vrlo malom vjerojatnošću pronaći rješenje \mathbf{x} .

Zamislimo da dvama koordinatnim vektorima metode konjugiranih gradijenata dodamo treći. Time potprostor proširujemo na tri vektora, pa je u usporedbi s onim koji sadrži samo dva razumno očekivati veće smanjenje energije po koraku. U najgorem slučaju, doprinos dodatnoga vektora može biti jednak nuli.² Tako razmišljajući možemo dodati četvrti, peti i sljedeće vektore, te očekivati još veće smanjenje energije po koraku. Na taj bi način trebalo statičkom sistemu u par koraka oduzeti većinu energije i “prigušiti ga” prema najnižoj točki – rješenju. Očito je zamisao o proširenju potprostora privlačna, ali samo do neke mjere. S jedne strane, generiranje vektora ne smije biti vremenski zahtjevno, a s druge strane, pretjerivanje s njihovim brojem povećava mali sustav kojega treba riješiti u svakom koraku. U krajnjem slučaju, ako bi dimenzija potprostora bila jednaka broju nepoznanica, minimizirali bismo ukupnu energiju i odredili rješenje u prvom koraku (zbog pogrešaka zaokruživanja možda u drugom), ali po “cijeni” jednako ili većoj od one za izravno rješavanje sustava.

2 Možemo zaključiti i ovako: minimum funkcije energije u širem potprostoru ne može biti veći (više razine) od minimuma u manjem potprostoru.

Mala matrica $\mathbf{A}_{(j)}$ može biti singularna (ili skoro singularna – loše uvjetovana), ako su neki koordinatni vektori točno (ili približno) linearno zavisni. Ako ih je više od dva, nije jednoznačno definirano koji treba odbaciti. Tada bismo mogli razmišljati o točnoj ili približnoj ortogonalizaciji takvih vektora, što “poskupljuje” korak.

Zato treba razmišljati o postupku tvorbe kojim nije (ili je rijetko) moguće generirati (gotovo) zavisne vektore. Radi lakše realizacije zamisli i tijekom proračuna možemo mijenjati način generiranja vektora. Ako se zavisnost ipak dogodi, postoji “spas u zadnji čas”, jer u postupku dekompozicije neke uporišne komponente (engl. *pivots*) od $\mathbf{A}_{(j)}$ postaju jednake (bliske) nuli. To možemo prepoznati i upotrijebiti za izbacivanje pripadajućih jednadžbi iz maloga sustava. Time smanjujemo dimenziju potprostora, ali mala matrica postaje regularna i bolje uvjetovana. Ovaj se pristup pokazao brzim i jednostavnim rješenjem problema linearne zavisnosti.

Skup iz kojega možemo odabrati dobre koordinatne vektore iznimno je velik. Na žalost, nisu nam unaprijed poznati (i koliko smo upućeni nitko ne zna), kriterije za izbor općeučinkovitih vektora. Niti je dovoljno poznata pozadinska teorija koja bi olakšala biranje. Mogućnosti je mnogo i možemo se (za sada) zadovoljiti provedbom i usporedbom numeričkih pokusa na brojnim primjerima. U pravilu, određeni podskup vektora za neke modele radi dobro, a za druge ne. U ovim uvjetima bili bismo zadovoljni izborom dobrih vektora, a najbolji podskup (nekoliko brzih i o modelima neovisnih vektora), bio bi ravan velikom otkriću.

Dva su temeljna pristupa tvorbi koordinatnih vektora: opći i posebni. U prvomu nisu potrebni nikakvi dodatni podaci o modelu. Matrica sustava i vektor desne strane dovoljni su za generiranje vektora. Ako je izbor vektora dobar, u većini smo praktičnih slučajeva zadovoljni s konvergencijom postupka. U drugom pristupu upotrebljavamo posebnosti koje vrijede za model koji rješavamo. Tada očekujemo odličnu konvergenciju, ali samo za taj model (i možda za mali skup njemu bliskih modela). Postoji i mješoviti postupak generiranja, kombinacija temeljnih pristupa. Neki načini realizacije opisani su u [8] i [9].

4. Trenutačno stanje postupka

Za sada su najbolji rezultati postignuti koordinatnim vektorima generiranim primjenom simetričnoga postupka prekomjerne relaksacije (engl. *symmetric successive overrelaxation*) [19]. Dakle, prvi je vektor određen kao

$$\phi_{1,(j)} = \mathbf{L}_{\Omega}^{-1} \mathbf{D} \mathbf{U}_{\Omega}^{-1} \mathbf{r}_{(j)}, \quad (7)$$

a za ostale smo upotrijebili rekurzivnu formulu

$$\phi_{j,(j)} = \mathbf{L}_{\Omega}^{-1} \mathbf{D} \mathbf{U}_{\Omega}^{-1} (\mathbf{A} \phi_{j-1,(j)}), \quad j = 2, \dots, m, \quad (8)$$

gdje su \mathbf{L}_Ω i \mathbf{U}_Ω donja i gornja trokutna matrica od \mathbf{A} s dijagonalnim elementima pomnoženim faktorom Ω . Riječ je o lokalnom faktoru relaksacije koji se ne mora podudarati s globalnim ω , niti mora ležati unutar intervala $(0,2)$. Zapravo, upravo istaknute trokutne matrice dobivamo ako napravimo prvi korak postupka obične prekomjerne relaksacije odozgo i odozdo, počevši od nul-vektora. Ovako generiranim vektorima pridružen je i prethodni prirast rješenja $\mathbf{p}_{(i-1)}$. Upotrebljavali smo od dva do deset koordinatnih vektora, uz $\omega = 1$ i Ω jednak 1 ili 1,65 (pogledajte odjeljak 6).

Pojasnimo malo umnožak u zagradi. Budući da jedan korak u smjeru vektora ϕ_1 daje trenutačni prirast pomaka $\alpha\phi_1$, gdje je α neki broj, rekursivna formula za rezidual daje $\mathbf{r}_{(i)} - \alpha\mathbf{A}\phi_1$. Ako na izraz primijenimo simetričnu prekomjernu relaksaciju, zbog (7) dobivamo drugi vektor kao $\phi_2 = \phi_1 - \alpha\mathbf{L}_\omega^{-1}\mathbf{D}\mathbf{U}_\omega^{-1}(\mathbf{A}\phi_1)$. Vektor ϕ_1 već sudjeluje u tvorbi potprostora, a α utječe samo na duljinu novoga vektora (ne mijenja potprostor koji taj vektor proširuje), pa ϕ_1 i α možemo izostaviti. Time dobivamo oblik (8). Brža realizacija ovoga pristupa, koju bi trebalo dodatno istražiti, jest umjesto \mathbf{A} upotrijebiti \mathbf{D} .

5. Ukratko o realizaciji i testiranju

Priloženi pseudokod realiziran je programskim jezikom gfortran [20]. Upotrijebljena je 64 bitna Ubuntu inačica 5.3.1 i OS X inačica 6.1.0. Nakon provjere programa na malim sustavima, generirali smo matrice krutosti i opterećenja velikoga broja ravninskih i prostornih rešetkastih nosača. S jedne strane, štapove smo postavljali klasično, tako da rešetke tvore dobar statički sistem. S druge strane, radi svjesnoga pogoršanja uvjetovanosti sustava, broja $\kappa(\mathbf{A})$, štapovima velikih razlika u krutostima nepravilno smo spajali udaljene čvorove. Time smo tvorili nelogične rešetke koje ne možemo smatrati konstrukcijama. Na taj smo način program testirali na, s numeričkoga gledišta, dobrim i lošim modelima. Upotrijebili smo potpuno knjiženje matrice po stupcima, a probali smo i po redcima [21]. Slični načini spremanja postoje i u FEAP-u, što je olakšalo spajanje koda s tim programom. Upotrijebili smo inačicu 8.4.1 [7]. Spajanje, programsko prevođenje i povezivanje s FEAP-om također je realizirano gfortran-om.

6. Rezultati proračuna praktičnih modela

Nakon temeljnih provjera, analizirali smo nekoliko modela iz konstruktorske prakse na kojima smo radili prijašnjih godina (slike 1. do 8.). Radi jasnoće, na slikama su izostavljeni prikazi opterećenja i ležajeva. Nepoznanice su pomaci, a ponegdje i kutovi zaokreta. Popunjenost matrice krutosti određena je kao omjer broja članova štednoga zapisa i broja svih elemenata matrice. Osnovni podaci o modelima prilože-

ni su u tablici. Istaknimo još, najveći $\kappa(\mathbf{A})$, određen kao omjer ekstremnih uporišnih komponentata pri dekompoziciji matrice \mathbf{A} , iznosi 10^{11} (primjer 6.), a najmanji 10^3 (primjer 4.).³

| Broj slike | Broj čvorova | Broj elemenata | Broj nepoznanica | Broj članova štednog zapisa | Popunjenost matrice |
|--------------------|----------------------|------------------------|----------------------|-----------------------------|--|
| 1. lijevo desno | 1 030 301 276 244 | 1 000 000 1 461 134 | 3 060 300 820 446 | 123 026 091 17 723 235 | $1,31 \cdot 10^{-5}$ $2,63 \cdot 10^{-5}$ |
| 2. lijevo desno | 71 3078 11 844 | 278 499 11 664 | 206527 69 984 | 3 826 156 1 635 876 | $8,97 \cdot 10^{-5}$ $3,34 \cdot 10^{-4}$ |
| 3. lijevo desno | 3 018 960 486 | 2 918 728 1 782 | 8 955 164 2 754 | 358 190 300 54 594 | $4,47 \cdot 10^{-6}$ $7,20 \cdot 10^{-3}$ |
| 4. | 132 651 | 125 000 | 397 947 | 15 692 116 | $9,90 \cdot 10^{-5}$ |
| 5. | 10 329 | 16 104 | 61 776 | 1 434 447 | $3,76 \cdot 10^{-4}$ |
| 6. | 21 420 | 27 411 | 117 936 | 3 133 980 | $2,25 \cdot 10^{-4}$ |
| 7. | 43 815 | 43 072 | 258 954 | 7 053 951 | $1,10 \cdot 10^{-4}$ |
| 8. | 79 162 | 74 247 | 289 986 | 10 251 174 | $1,22 \cdot 10^{-4}$ |

Modeli sadrže štapne, plošne i volumne elemente velikih razlika u krutostima, što je za statičke sisteme u građevinarstvu uobičajeno. Primjerice, često spajamo elemente velike uzdužne krutosti (debelih armiranobetonskih zidova) i male fleksijske krutosti (tankih čeličnih limova) u ne baš uvijek optimalan oblik mreže. Zato je veliki $\kappa(\mathbf{A})$ za modele građevinskih konstrukcija neizbježan. Također, razdioba je vlastitih vrijednosti modela jednolična, što je svojstvo dobro osmišljene konstrukcije, jer gomilišta vlastitih vrijednosti nepovoljno utječu na dinamički odziv i stabilnost (osjetljivost) statičkoga sistema.

Raznovrsne su provjere iteracijskih algoritama nužne, jer je poznata njihova ovisnost o naravi problema. Odnosno, mogu se prilagoditi za izrazito brzo rješavanje tipičnih primjera, s unaprijed poznatim podacima kojima namjestimo ključne parametre postupka. Međutim, čim problem odstupa od očekivanoga, učinkovitost pada.

Na slikama 1. do 3. priložene su razdiobe vertikalnih pomaka modela [pod a)], te ovisnosti logaritma omjera reziduala $\|\mathbf{r}_i\|_2 / \|\mathbf{r}_0\|_2$ i energije $(\Pi_0 - \sum_i \Delta \Pi_i) / \Pi_0$ o broju koraka [pod b) i c)]. Crvena boja označava područje najmanjih, a ljubičasta najvećih pomaka. Provedeni su proračuni IRM-om s dva, četiri, šest i deset koordinatnih vektora (argument uz IRM) i $\Omega = 1,65$ te metodom konjugiranih gradijenata bez dijagonalnoga preduvjetovanju (CG) i s njim (CGD). Na slikama pod b) i c) možemo uočiti: s povećanjem potprostora opada broj potrebnih koraka, odnosno veće je smanjenje reziduala i energijske norme po koraku. Čak i za dva koordinatna vektora postupak brže konvergira od CG i CGD (koje također interpretiramo s dva vektora).

3 Dekompozicija matrice sustava napravljena je samo radi procjene .

| Broj slike | Broj koraka za $\varepsilon < 10^{-8}$ ($\omega = 1$ i $\Omega = 1,65$) | | | | | |
|------------|---|--------|---------|--------|--------|---------|
| | CG | CGD | IRM(2) | IRM(4) | IRM(6) | IRM(10) |
| 1. lijevo | 580 | 567 | 243 | 102 | 67 | 38 |
| desno | $> 10^4$ | 5 083 | 2 381 | 1 097 | 682 | 410 |
| 2. lijevo | $> 10^5$ | 53 002 | 24 9995 | 8 608 | 5 166 | 2 871 |
| desno | 11 091 | 8 966 | 4 124 | 1 381 | 888 | 498 |
| 3. lijevo | 2 157 | 1 936 | 623 | 208 | 126 | 71 |
| desno | 2 769 | 987 | 703 | 269 | 169 | 94 |

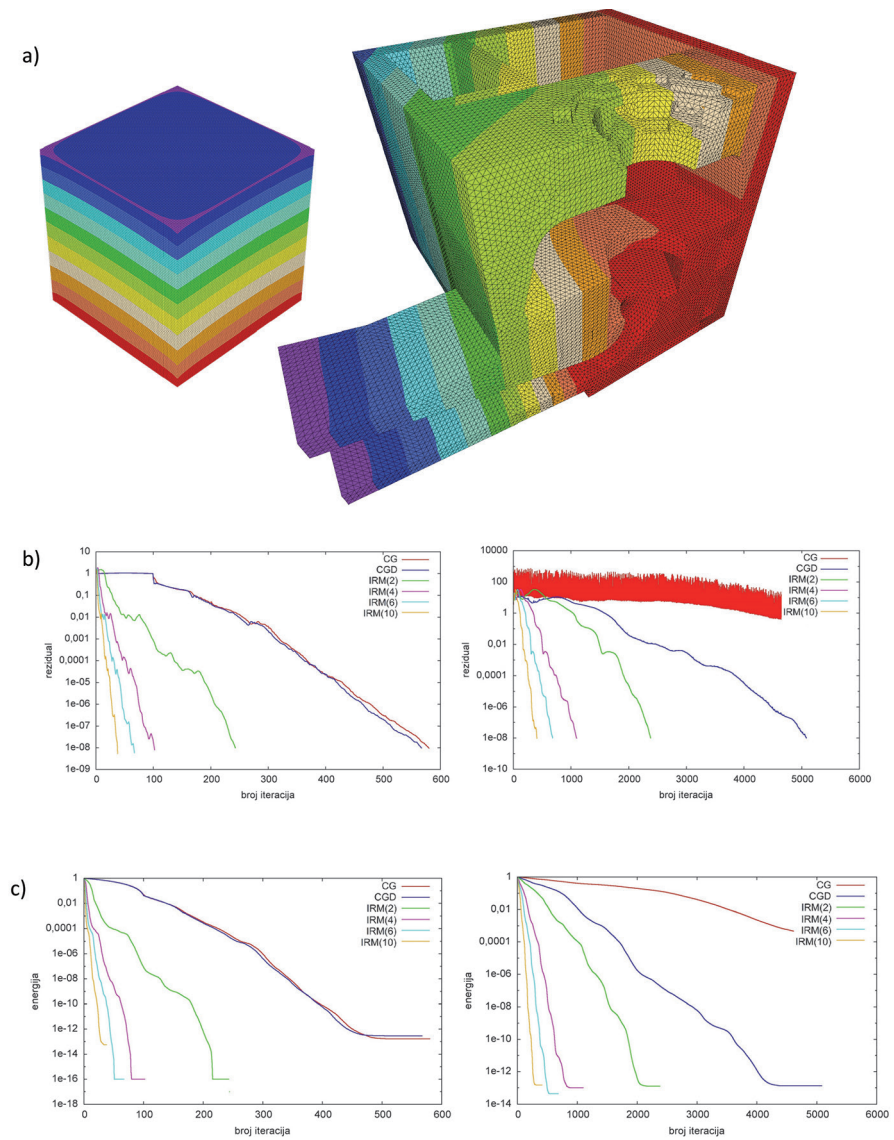
Na slikama 4. do 8. dodani su rezultati proračuna postupkom najstrmijega silaska (SD), Jacobijevim postupkom (JAC) i metodom konjugiranih gradijenata s blok–čvor-nim (engl. *block nodal*) preduvjetovanjem (CGBN). IRM proračuni napravljeni su za $\Omega = 1$ i prema broju iteracija postupak ostaje u prednosti. Opadanje energije slično je kao na slikama 1. do 3., pa smo funkcije izostavili. Za odabrane primjere proračuni postupcima SD, JAC i CG presporo konvergiraju – potreban broj koraka nismo prilagali. Istaknimo još, rezultati svih primjera podudaraju se s rješenjima dobivenima izravnim i iteracijskim postupcima koji su standardni dio FEAP-a.

| Broj slike | Broj koraka za $\varepsilon < 10^{-8}$ ($\omega = \Omega = 1$) | | | | | |
|------------|--|-------|--------|--------|--------|---------|
| | CG | CGD | IRM(2) | IRM(4) | IRM(6) | IRM(10) |
| 4. | 1 396 | 1 394 | 456 | 305 | 233 | 159 |
| 5. | 2 049 | 757 | 704 | 235 | 141 | 79 |
| 6. | 4 155 | 3 114 | 1 303 | 502 | 324 | 167 |
| 7. | 4 682 | 4 040 | 1 509 | 501 | 300 | 166 |
| 8. | 5 708 | 5 155 | 1 559 | 565 | 313 | 191 |

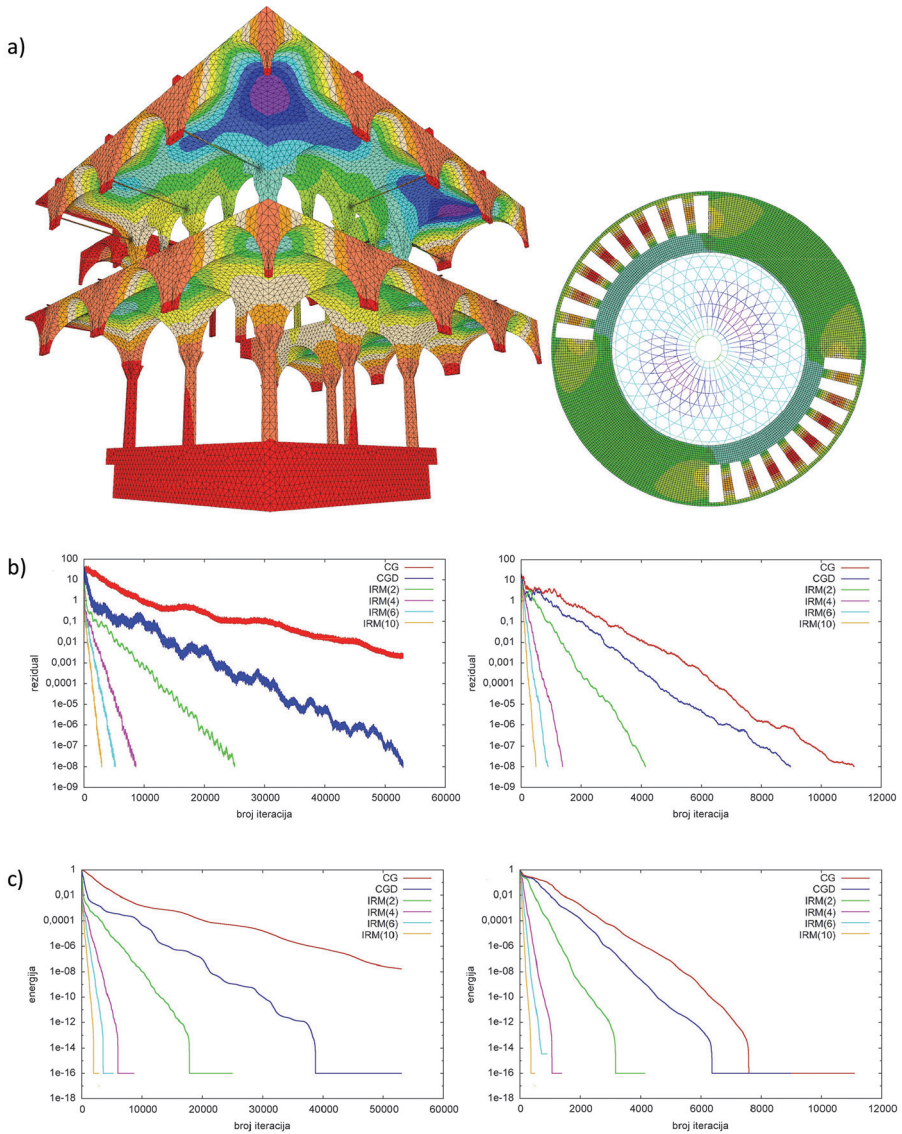
Treba priznati, postoji bolje preduvjetovanje metode konjugiranih gradijenata, primjerice (ne uvijek stabilnim) nepotpunim rastavom Choleskoga (engl. *incomplete Cholesky factorization*) [22], zatim primjenom algebarskoga višemrežnog (engl. *algebraic multigrid*) postupka [23] ili približne inverzije (engl. *approximate inverse*) [24].

7. Budući razvoj

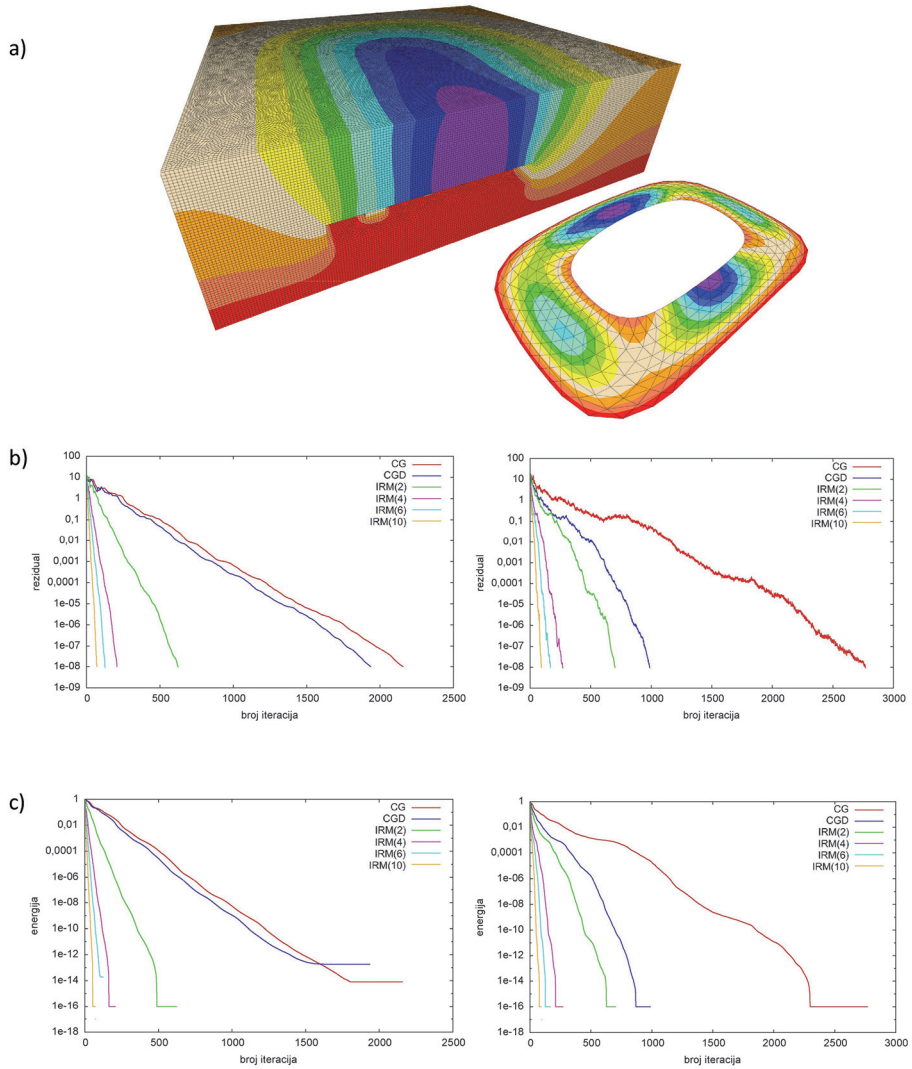
Međutim, sigurno postoji bolji potprostor za IRM. Koordinatne vektore možemo generirati bilo kojim od istaknutih ili drugim iteracijskim postupcima, pobrojavanjem nepoznanica prema naprijed (engl. *forward*), unatrag (engl. *backward*), u skladu s opadanjem komponenata reziduala, ili drugim obećavajućim redosljedom. Potonja dva pristupa iziskuju sortiranje što “poskupljuje” korak. Zato se može razmišljati o povremenoj obnovi reziduala, općenito redosljeda obilaženja nepoznanica.



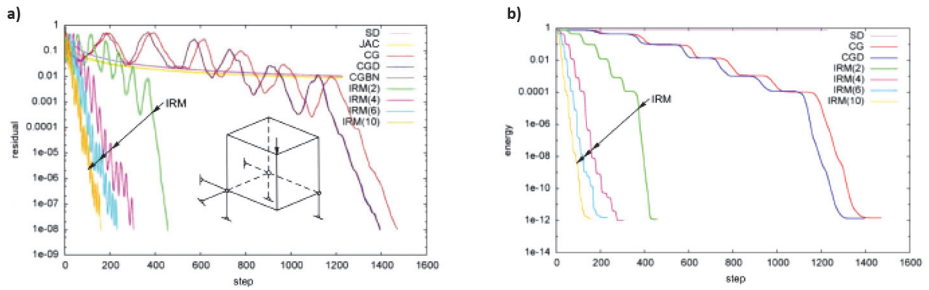
Slika 1. Proračun modela kocke (lijevo) i strojarnice HE Rama (desno; prikazana je samo polovina modela) [28]: a) razdioba vertikalnih pomaka, b) opadanje reziduala, c) smanjenje energije



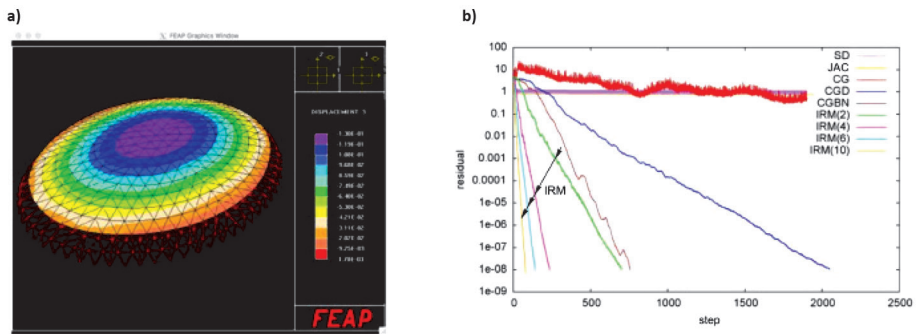
Slika 2. Proračun atrija Kneževa dvora u Dubrovniku (lijevo) [29] i kupole dvorane “Krešimir Ćosić” u Zadru (desno) [30]: a) razdioba vertikalnih pomaka, b) opadanje reziduala, c) smanjenje energije



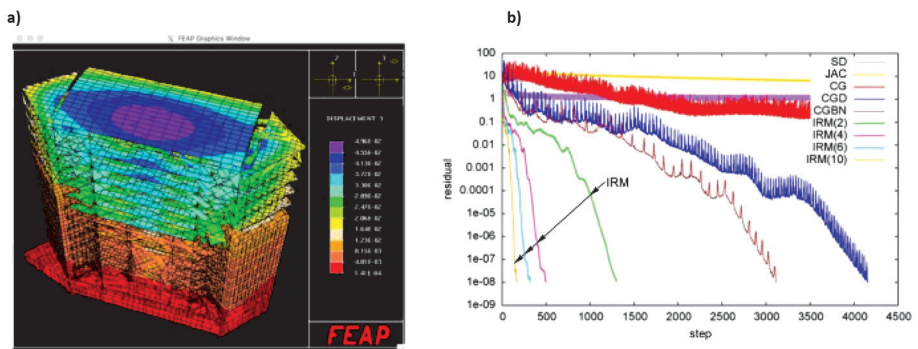
Slika 3. Proračun podzemnoga kamenoloma Kanfanar s okolnim područjem (lijevo; unutarnje prostori i stupovi nisu vidljivi) [31] i krovišta budućega stadiona Kantrida na Rijeci (desno) [32]: a) razdioba vertikalnih pomaka, b) opadanje reziduala, c) smanjenje energije



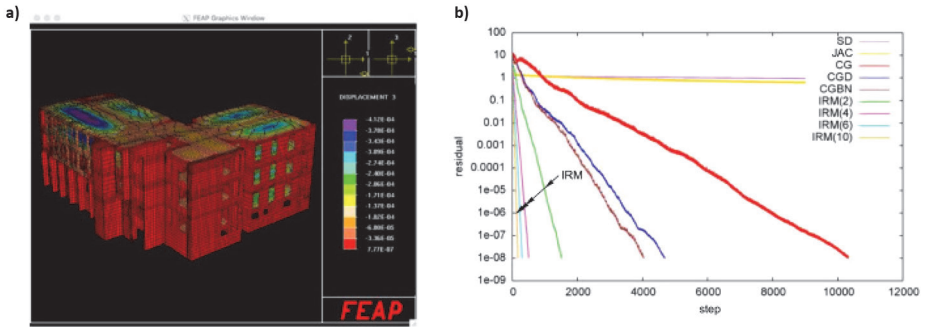
Slika 4. Jednostavan test: a) model, b) opadanje reziduala



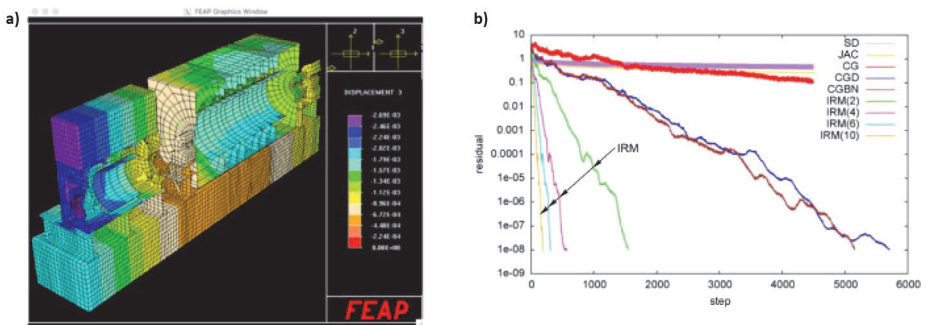
Slika 5. Košarkaška dvorana Cibona [33]: a) model, b) opadanje reziduala



Slika 6. Nadogradnja zgrade Euroherc osiguranja u Zagrebu [37]: a) model, b) opadanje reziduala



Slika 7. Dio bolnice *Sveti duh* u Zagrebu [35]: a) model, b) opadanje reziduala



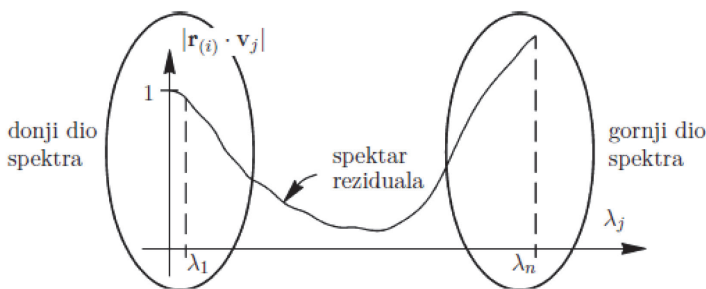
Slika 8. Temelj i generator u tvornici *SMIT transformers* u Nijmegenu (presjek) [36]: a) model, b) opadanje reziduala

Možemo raditi i više ciklusa za tvorbu jednoga vektora. Primjerice, nekoliko puta naprijed–nazad s različitim iznosima Ω . Time izgladujemo koordinatne vektore pa i rezidual, što doprinosi brzini konvergencije.

Postupci koji služe generiranju ne moraju biti optimizirani, kao za slučaj neovisne primjene. Vrijede čak i oni koji se ne upotrebljavaju samostalno, jer nisu dovoljno brzi, robusni, numerički stabilni i slično. Za IRM su dovoljni razmjerno “grubi” koordinatni vektori koji se međusobno nadopunjuju u smislu smanjenja reziduala uzduž različitih dijelova spektra.

Preciznije, ako $\mathbf{r}_{(i)}$ rastavimo na vlastite vektore \mathbf{v}_j od \mathbf{A} ($j = 1, \dots, n$), trebalo bi formirati dva ili tri koordinatna vektora koji izgladuju donji dio spektra reziduala (malih vlastitih vrijednosti λ_j) i dodati još dva, tri, specijalizirana za gornji dio spektra, velikih λ_j (slika 9.). Ne mora pri tome pojedini vektor dobro aproksimirati $\mathbf{A}^{-1} \mathbf{r}_{(i)}$, nego potprostor koji razapinju svi vektori treba sadržavati što bolju aproksimaciju toga umnoška. Njima bi trebalo dodati i $\mathbf{p}_{(i-1)}$ – bit uspješnosti metode konjugiranih gradijenata. Slično učinku većega broja ciklusa kod tvorbe jednoga vektora, ovdje svi

vektori izgladuju potprostor. S porastom njihova broja (i/ili ciklusa za jedan vektor), glatkoća reziduala raste, a broj koraka pada.



Slika 9. Skica čestoga oblika spektra reziduala modela građevinskih konstrukcija

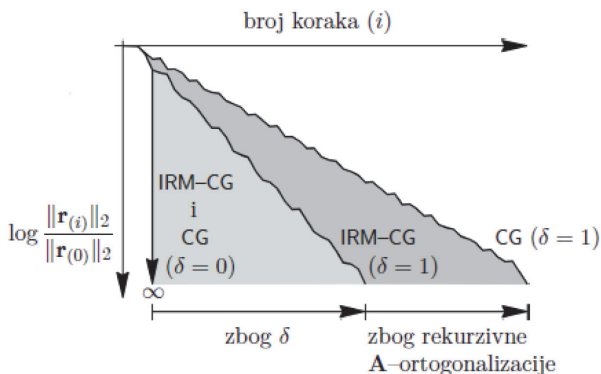
Ako se netko sjeti boljega skupa (s jednim ili više) koordinatnih vektora, bez teškoća može formirati matricu $\Phi_{(i)}$ i upotrijebiti opisani postupak. Na taj način IRM možemo smatrati općim iteracijskim algoritmom, kojemu kao ulazne podatke (osim nužnih koje trebaju svi iteracijski postupci), zadajemo i koordinatne vektore, odnosno potprostor.

Primjena egzaktno aritmetike. U raspravama o stabilnosti iteracijskih postupaka, pa i IRM-a, programi s egzaktnom aritmetikom mogu pomoći, jer eliminiramo pogrešku zaokruživanja. Samo trebamo točnu prezentaciju ulaznih podataka (primjerice u obliku racionalnih brojeva) i sve se veličine računaju točno. Promotrimo jednostavan cjelobrojni primjer:

$$\mathbf{A} = \begin{bmatrix} 4 & -1 & -1 \\ -1 & 3 & -1 \\ -1 & -1 & 2 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 1 \\ 2 \\ 5 \end{bmatrix}. \quad (9)$$

Broj uvjetovanosti je malen, $\kappa(\mathbf{A}) = \lambda_1/\lambda_3 \approx 5,95$, gdje su λ_i nul-točke vlastitoga polinoma. Rješenje potražimo postupkom CG i IRM-om s dva koordinatna vektora, $\mathbf{r}_{(i)}$ i $\mathbf{p}_{(i-1)}$, što je (istaknuli smo) ekvivalentno CG-u. Postupak smo nazvali IRM-CG [25]. Oba pristupa daju točan rezultat nakon tri koraka, odnosno $\mathbf{x}_{(3)} = [31/13 \ 42/13 \ 69/13]^T$. To je u skladu s poznatim teoremom M. R. Hestensa i E. Stiefela [27], jer sustav sadrži tri nepoznanice. Početna relativna norma reziduala jednaka je (cjelobrojnoj) jedinici, a ostale su redom $\sqrt{179}/16$, $19/567\sqrt{358/15}$ i (cjelobrojna) nula. Unesimo (samo) u prvom koraku cjelobrojni poremećaj \mathbf{A} - ortogonalnosti za oba postupka, nazovimo ga δ . Primjerice, poremetimo treću komponentu prirasta rje-

šenja $\mathbf{p}_{3,(1)} \leftarrow \mathbf{p}_{3,(1)} + \delta$. To u realnomu numeričkom okruženju odgovara gubitku ortogonalnosti zbog gomilanja pogrešaka zaokruživanja, izraženoga kod velikih i slabo uvjetovanih sustava. Tada se IRM–CG ponaša stabilnije od CG-a (slika 10.). Zbog poremećaja više ne vrijede pretpostavke teorema iz [27] i premda je riječ o egzaktnoj aritmetici, nisu dovoljna tri koraka niti se može pronaći egzaktno rješenje. Kriterij konvergencije može biti neubičajeno malen (uzmimo $\varepsilon = 10^{-50}$), jer sustav rješavamo bez pogreške zaokruživanja, ali trebamo više koraka i vremena za proračun. To je razumljivo, jer rastu cijeli brojevi, zatim onih pod drugim korijenom, te brojnici i nazivnici koji određuju točne vrijednosti svih varijabli. Međutim, manji ε znači sve veću prednost IRM–CG-a.



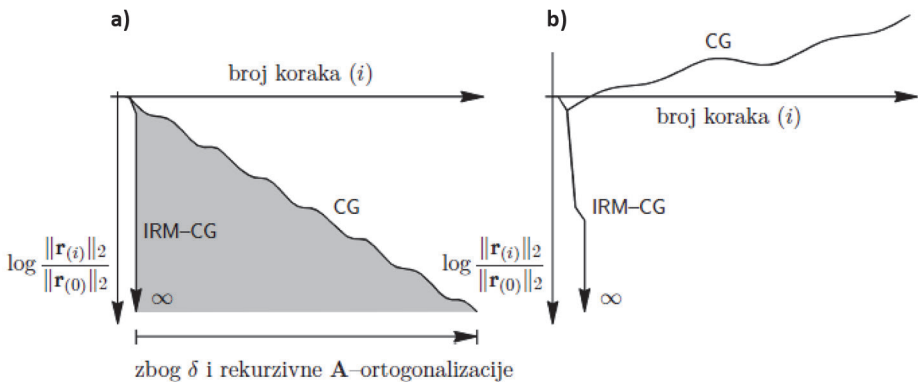
Slika 10. Egzaktno opadanje reziduala uz poremećaj $\mathbf{p}_{3,(1)} + 1$

Razlika nastaje, jer u istaknutoj ravnini IRM–CG nalazi točno rješenje unutar koraka (zato se rješava sustav reda dva), a CG samo približno rješenje (određeno rekurzivnom \mathbf{A} – ortogonalizacijom, premda je provedena egzaktno). Drugim riječima, zbog δ , CG rekurzijom dobiva egzaktnu grešku koja u svakom koraku egzaktno raste. Kod IRM–CG-a ona postoji samo u prvom koraku (gdje je i uvedena), a budući da se rekurzija ne upotrebljava, te greške u ostalim koracima nema (ne nasljeđuje se). Zato se IRM–CG bolje “oporavlja” od (možemo reći i perturbacije) δ .

Leži li poremećaj ortogonalnosti u ravnini razapetoj vektorima $\mathbf{p}_{(i-1)}$ i $\mathbf{r}_{(i)}$, primjerice u prvom koraku unesemo $\mathbf{p}_{(1)} \leftarrow \mathbf{p}_{(1)} + \mathbf{r}_{(2)}$, IRM–CG daje egzaktno rješenje u tri koraka, kao da poremećaja nema, a CG “osjeća” teškoću (slika 11.a). Ako zadamo poremećaj ortogonalan na ravninu, uzmimo $\mathbf{p}_{(1)} \leftarrow \mathbf{p}_{(1)} \times \mathbf{r}_{(2)}$, IRM–CG daje točno rješenje u četiri koraka (posljednja relativna norma reziduala je cjelobrojna nula), a CG ne konvergira (slika 11.b).

Promotrimo jedan veći, realističniji primjer – kocku s minimalno potrebnim brojem ležajeva, opterećenu jediničnom silom na vrhu (slika 12.). Diskretizirana je Lagran-geovim C^0 konačnim elementom sa 192 stupnja slobode. Unutarnji čvorovi nisu

kondenzirani, a matrica krutosti određena je egzaktno, racionalnim brojevima [37]. Vektor opterećenja također je točan, zadan cijelim brojevima. Kocka je pridržana minimalnim brojem opruga cjelobrojnih krutosti, kojima smo utjecali na $\kappa(\mathbf{A})$. Sustav ima 192 nepoznanice. Problem je riješen postupcima IRM–CG i CG, egzaktnom aritmetikom (indeks E) i primjenom dvostruke točnosti (indeks DT), uz $\varepsilon = 10^{-10}$. Bez obzira na $\kappa(\mathbf{A})$, tijekom točnoga proračuna podudara se za oba postupka i završava nakon 188 koraka. Riječ je o broju nepoznanica umanjenom za četiri, koliko je “neaktivnih” λ_j , onih kojima pripada $\mathbf{v}_j \cdot \mathbf{b}$ jednak (u egzaktnoj aritmetici cjelobrojnoj) nuli [25, 26]. Ili drugačije, potreban broj koraka jednak je broju “aktivnih” vlastitih vrijednosti, za koje je $\mathbf{v}_j \cdot \mathbf{b} \neq 0$.⁴



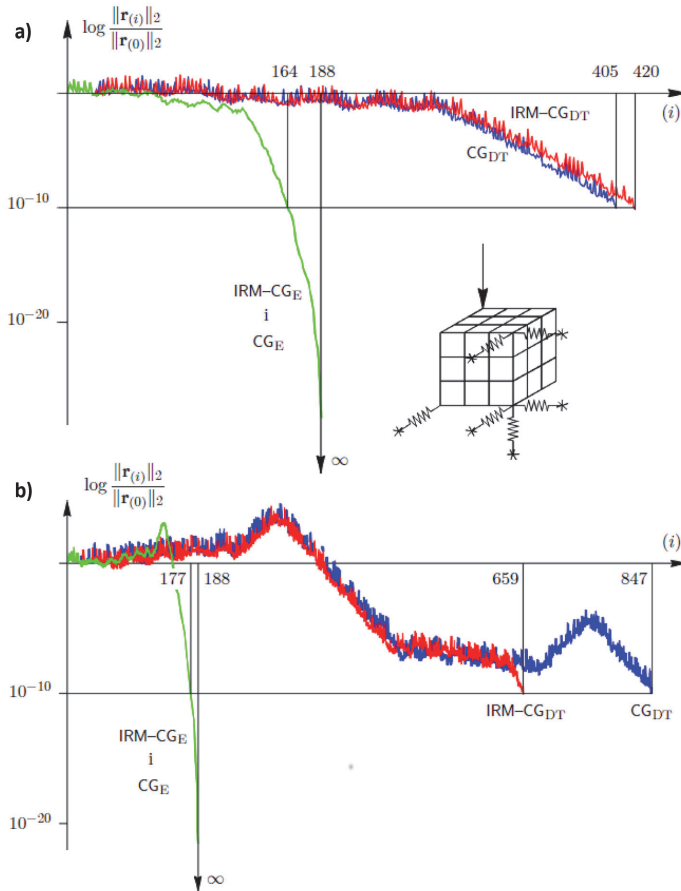
Slika 11. Egzaktno (ne)opadanje reziduala uz poremećaj: a) $\mathbf{p}_{(1)} + \mathbf{r}_{(2)}$, b) $\mathbf{p}_{(1)} \times \mathbf{r}_{(2)}$

Zanimljiva je velika razlika u broju koraka između točnoga i numeričkih pristupa, čak i za dobro uvjetovani primjer. A osim numeričkog, drugoga praktičnog načina proračuna nemamo. Očito, povećanje točnosti na više od šesnaest znamenaka može biti opravdano.⁵ I dok se u numeričkom okruženju za dobro uvjetovani model funkcije opadanja reziduala gotovo podudaraju (slika 12.a), za lošije uvjetovani model IRM–CG je stabilniji (slika 12.b). Naravno, u oba slučaja numerička su rješenja bliska i zanemarivo odstupaju od točnoga. Ističemo, riječ je o malim primjerima. Ako bi rezultati sa slike 12. vrijedili za velike sustave, postupci bi bili neučinkoviti. A znamo iz prakse, upravo je suprotno – broj koraka puno je manji od broja nepoznanica.

4 Ako opterećenje možemo prikazati linearnom kombinacijom ℓ vlastitih vektora, primjenom egzaktne aritmetike postupci daju točno rješenje nakon ℓ koraka.

5 S porastom broja značajnih znamenaka krivulje numeričkih pristupa međusobno se približavaju i teže prema krivulji egzaktnog pristupa. Radi preglednosti slika, rezultate proračuna s povećanom točnošću nismo priložili.

Zašto je tako vidi se na slici 13. Skicirano je tipično ponašanje reziduala za veliki broj stupnjeva slobode (aktivnih vlastitih vrijednosti ℓ).



Slika 12. Opadanje reziduala uz egzaktne i numeričke pristupe CG-u i IRM-CG-u za različite uvjetovanosti modela: a) $\kappa(\mathbf{A}) = 7,7 \cdot 10^5$, b) $\kappa(\mathbf{A}) = 6,4 \cdot 10^{12}$

Minimizacija kvadrata reziduala. Rješenje sustava (1) ne minimizira samo (2) nego i kvadrat duljine reziduala:

$$\mathbf{r}^T \mathbf{r} = (\mathbf{b} - \mathbf{A}\mathbf{x})^T (\mathbf{b} - \mathbf{A}\mathbf{x}) = \mathbf{b}^T \mathbf{b} - 2\mathbf{x}^T \mathbf{A}\mathbf{b} + \mathbf{x}^T \mathbf{A}^2 \mathbf{x}. \quad (10)$$

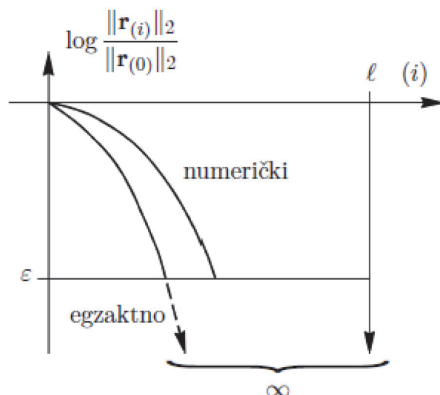
Pronalaskom egzaktne rješenja vrijednost funkcionala je točno, a u slučaju numeričkoga rješenja, približno jednaka nuli. Ako izraz pomnožimo s 1/2 i zanemarimo konstantu $\mathbf{b}^T \mathbf{b}$ (jer će derivacijom iščeznuti) dobivamo kvadratnu formu

$$g(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{A}^2 \mathbf{x} - \mathbf{x}^T \mathbf{A} \mathbf{b}, \quad (11)$$

koja podsjeća na (2), samo \mathbf{A}^2 i $\mathbf{A} \mathbf{b}$ zamijenimo s \mathbf{A} i \mathbf{b} . Minimizacijom (10) ili (11) dobivamo

$$\mathbf{A}^2 \mathbf{x} = \mathbf{A} \mathbf{b}, \quad (12)$$

što je zapravo (1), loše preduvjetovan matricom \mathbf{A} , jer je \mathbf{A}^2 više popunjena i kvadratno slabije uvjetovana od \mathbf{A} , odnosno $\kappa(\mathbf{A}^2) = \kappa(\mathbf{A})^2$. Matrice imaju iste vlastite vektore, a vlastite vrijednosti od \mathbf{A}^2 jednake su kvadratu vlastitih vrijednosti od \mathbf{A} . Zato su familije elipsoida $f(\mathbf{x}) = c$ i $g(\mathbf{x}) = c$ (c je konstanta) međusobno centrične, a druga je familija izduljenija od prve, jer su duljine poluosi jednake vlastitim vrijednostima. Naravno, elipsoidima pripada zajedničko središte – rješenje (slika 14.a).

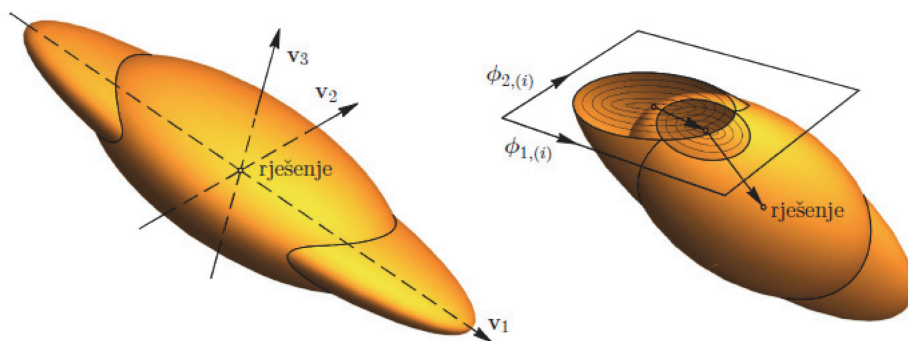


Slika 13. Skica tipičnoga opadanja reziduala pri rješavanju velikih sustava

Ako problem rješavamo minimizacijom $f(\mathbf{x})$ ili $g(\mathbf{x})$, primjenom IRM-a rješenju prilazimo na različite načine, jer se lokalni minimumi energije i kvadrata reziduala u potprostoru koordinatnih vektora ne poklapaju (slika 14.b). U prvom slučaju norma energije monotono pada, a norma reziduala je vrlo nepravilna funkcija koja tijekom postupka može značajno rasti. Tada je trenutačno rješenje grubo i konvergencija je spora. U drugom slučaju obje su funkcije glatke i monotono padaju, što postupak čini stabilnijim i često bržim od prvoga pristupa. Zato očekujemo dobro ponašanje i pri rješavanju nelinearnih, posebice o putu ovisnih (engl. *path dependent*) problema. Primijetite još, postupak vrijedi i za nesimetrične matrice, samo u gornjim izrazima umjesto \mathbf{A}^2 treba upotrijebiti nesimetrični umnožak $\mathbf{A}^T \mathbf{A}$.

Zašto je ponašanje dobro, a minimizacijom rješavamo sustav lošije uvjetovan od izvornoga? Naime, budući da ne primjenjujemo rezidual sustava (12), jednak $\mathbf{A}\mathbf{r}$, nego \mathbf{r} , sustava (1), to je istoznačno preduvjetovanju (12) matricom \mathbf{A}^{-1} , čime vraćamo uvjetovanost izvornog sustava.

Tijekom istraživanja uočili smo i mješoviti način koji može ubrzati IRM: u nekim koracima minimizirati $f(\mathbf{x})$, a u nekim $g(\mathbf{x})$ (slika 14.b). Koordinatni vektori ne moraju biti isti za obje minimizacije. Prelazak s jedne minimizacije na drugu ovisi o obliku funkcija reziduala i energije, a kriterije prelaska, načine njihove automatizacije i prikladne vektore upravo istražujemo. Ovaj je pristup zapravo poopćenje postupka Barzilai–Borwein [38].



Slika 14. Grafički prikaz mješovitoga postupka IRM s dva koordinatna vektora u primjeni na sustav od tri nepoznanice: a) elipsoidi $f(\mathbf{x}) = c$ i $g(\mathbf{x}) = c$, b) presjek ravninom koordinatnih vektora

8. Zaključak

U skladu s opisanim svojstvima i našim iskustvima, iterirani Ritzov postupak može imati prednosti kod rješavanja velikih linearnih sustava. Neki naši primjeri imali su preko 10^8 nepoznanica, popunjenosti oko 10^{-7} . Tumačenje postupka blisko je inženjerskomu načinu razmišljanja, Ritzovoj energijskoj interpretaciji, za razliku od primjerice metode konjugiranih gradijenata koja se uobičajeno pojašnjava geometrijski, u apstraktnomu n -dimenzijskom prostoru.

Ovaj postupak ne bi smio biti lošiji od metode konjugiranih gradijenata (s preduvjetovanjem i bez njega). Odnosno, dobro proširenje potprostora ne donosi pogoršanje konvergencije. Pri tomu možemo upotrijebiti više načina generiranja koordinatnih vektora. Kao da radimo s nekoliko iteracijskih postupaka istodobno. Dobrim izborom vektora konvergencija je puno brža od konvergencije pojedinačnoga postupka. Ponovno pokretanje (engl. *restart*), poznato iz metode konjugiranih gradijenata, kojim se zbog gubitka ortogonalnosti ponovno započinje postupak, ovdje nije op-

ravdano. Rekurzivna ortogonalizacija se ne upotrebljava, a iz prethodnoga koraka nasljeđujemo samo prirast pomaka koji poboljšava trenutnačno rješenje.

Preduvjetovanje postupka nije potrebno, ali se takvi algoritmi mogu uspješno primijeniti za tvorbu vektora, bez transformiranja sustava koje prati preduvjetovanje.

Budući da ne zahtijevamo ortogonalnost među iteracijskim veličinama, dodatno ubrzanje treba očekivati u primjeni na nelinearne probleme. Naime, tada se gube svojstva ortogonalnosti na kojima ustrajavaju brojni iteracijski algoritmi. Zbog toga se postupak može uspješno primijeniti i u području optimizacije.

Jasno je, izrazito velike sustave nije ekonomično rješavati jednim procesorom. Tada paralelan pristup postaje nužnost. Zanimljivo bi bilo svakomu koordinatnom vektoru pridružiti njegov procesor.

Radi brojnih mogućnosti, još smo u intenzivnoj istraživačkoj fazi algoritma (već samo izmjene iznosa Ω i ω izrazito utječu na brzinu konvergencije, a o boljim koordinatnim vektorima i načinima njihova generiranja da ne raspravljamo), još nema smisla uspoređivati IRM s drugim izravnim i iteracijskim postupcima. To je moguće napraviti tek nakon brižljivoga programiranja, programskoga prevođenja i povezivanja konačne inačice programa, s uključenim optimizacijskim opcijama koje doprinose brzini izvođenja koda [39].

Zahvale

Ovaj je rad financirala Hrvatska zaklada za znanost projektom IP – 2014 – 09 – 2899. Zahvaljujemo prof. dr. sc. Sonji Gorjanc na pomoći oko geometrijske interpretacije postupka.

Literatura

- [1] Dvornik, J.: *Generalization of the CG Method Applied to Linear and Nonlinear Problems*, Trends in computerized structural analysis and synthesis; Proceedings of the Symposium (urednici Noor, A. K. i McComb, H. G. Jr.), NASA, Langley Research Center, Washington D. C., 30. X – 1. XI 1978., Pergamon Press, str. 217. – 223.
- [2] Dvornik, J.: *Generalization of the CG Method Applied to Linear and Nonlinear Problems*, Computers & Structures **10** (1979) 1/2, str. 217.–223.
- [3] Dobláré, M.; Alarcon, E.: *A Three-Dimensional B.I.E.M. Program*, Finite Element Systems (urednik Brebbia, C. A.), Springer, Berlin, Heidelberg, 1982., str. 325. – 345.
- [4] Ryu, Y.-S.; Arora, J. S.: *Review of Nonlinear FE Methods with Substructures*, Journal of Engineering Mechanics, **111** (1985) 11, str. 1361. – 1379.
- [5] Dobláré, M.: *Computational Aspects of the Boundary Element Method*, Topics in Boundary Element Research, Computational Aspects, (urednik Brebbia, C. A.), Vol 3., Springer, Berlin, Heidelberg, 1987., str. 51. – 131.

- [6] Taylor, R. L.: *FEAP – A Finite Element Analysis Program*, Version 8.4, User Manual, Department of Civil and Environmental Engineering, University of California at Berkeley, Berkeley, 2013.
- [7] Taylor, R. L.: *FEAP – A Finite Element Analysis Program*, Version 8.4, Programmer Manual, Department of Civil and Environmental Engineering, University of California at Berkeley, Berkeley, 2014.
- [8] Dvornik, J.; Lazarević, D.: *Iterirani Ritzov postupak za rješavanje sustava linearnih algebarskih jednadžbi*, *Građevinar*, **69** (2017) 7, str. 521. – 535.
- [9] Dvornik, J.; Lazarevic, D.; Uros, M.; Savor Novak, M.: *The Iterated Ritz Method: Basis, implementation and further development*, *Coupled Systems Mechanics*, **6** (2018) 7, str. 755. – 774.
- [10] Atalić, J.; Uroš, M.; Šavor Novak, M.: *Sanacija tradicijskih kamenih stupova olovnom umetkom*, *Građevinar*, **70** (2018) 9, str. 741. – 756.
- [11] Saad, Y.: *Iterative Methods for Sparse Linear Systems*, Siam, Society for Industrial and Applied Mathematics, Philadelphia, 2003.
- [12] Olshanskii, M. A.; Tyrtshnikov, E. E.: *Iterative Methods for Linear Systems. Theory and Application*, Siam, Society for Industrial and Applied Mathematics, Philadelphia, 2014.
- [13] Ivančić, N.: *Metoda konjugiranih gradijenata*, *Građevinar*, **24** (1972) 9–10, str. 369. – 374.
- [14] Maess, G.: *Projection methods solving rectangular systems of linear equations*, *Journal of Computational and Applied Mathematics*, **24** (1988) 1 – 2, str. 107. – 119.
- [15] Brezinski, C.: *Projection methods for linear systems*, *Journal of Computational and Applied Mathematics*, **77** (1997) 1 – 2, str. 35. – 51.
- [16] Barrett, R.; Berry, M.; Chan, T. F.; Demmel, J.; Donato, J.; Dongarra, J.; Eijkhout, V.; Pozo, R.; Romine, C.; van der Vorst, H.: *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, 1987.
- [17] Benzi, M.: *Preconditioning techniques for large linear systems: A survey*, *Journal of Computational Physics*, **182** (2002), str. 418. – 477.
- [18] Korneev, V. G.; Langer, U.: *Domain Decomposition Methods and Preconditioning*, u: *Encyclopedia of Computational Mechanics* (Editors: Erwin Stein, René de Borst, Thomas J. R. Hughes), Chapter 22, Volume 1: Fundamentals, John Wiley & Sons, Ltd, Chichester, 2004., str. 617. – 649.
- [19] Young, D. M.: *Iterative Solution of Large Linear Systems*, Dover Publications, Inc., Mineola, New York, 2003.

- [20] The gfortran team: *Using GNU Fortran*, For GCC version 7.0.0 (pre – release), (GCC), Free Software Foundation, Boston, 2016.
- [21] Nour – Omid, B.; Taylor, R. L.: *An Algorithm for Assembly of Stiffness Matrices into a Compacted Data Structure*, Report No. UCB/SESM – 84/06, Structural Engineering and Structural Mechanics, Department of Civil Engineering, University of California, Berkeley, 1984.
- [22] Van’t Wout, E.; Van Gijzen, M. B.; Ditzel, A.; van der Ploeg, A.; Vuik, C.: *The Deflated Relaxed Incomplete Cholesky CG method for use in a real-time ship simulator*, *Procedia Computer Science* **1** (2010) 1, str. 249. – 257.
- [23] Iwamura, C.; Costa, F. S.; Sbarski, I.; Easton, A.; Li, N.: *An efficient algebraic multigrid preconditioned conjugate gradient solver*, *Computer Methods in Applied Mechanics and Engineering*, **192** (2003) 20–21, 2299.–2318.
- [24] Pereira, F. H.; Verardi, S. L. L.; Nabeta, S. I.: *A fast algebraic multigrid preconditioned conjugate gradient solver*, *Applied Mathematics and Computation*, **179** (2006) 1, str. 344. – 351.
- [25] Dvornik, J.; Lazarevic, D.; Jaguljnjak Lazarevic, A.; Demsic, M.: *Nonrecursive Equivalent of the Conjugate Gradient Method without the Need to Restart*, *Advances in Civil Engineering*, (2019), str. 1. – 5.
- [26] Van der Vorst, H. A.: *Iterative Krylov Methods for Large Linear Systems*, Cambridge University Press, Cambridge, 2003.
- [27] Hestens, M. R.; Stiefel, E.: *Methods of Conjugate Gradients for Solving Linear Systems*, *Journal of Research of the National Bureau of Standards*, **49** (1952) 6, str. 409. – 436.
- [28] *Statički i dinamički proračun postojećih temelja agregata na nova opterećenja i djelovanja*, knjiga G3, glavni projekt (projektant konstrukcije za ovu projektnu obradu Milan Crnogorac), Javno poduzeće Elektroprivreda hrvatske zajednice Herceg bosne d. d. Mostar, Zagreb, 2012.
- [29] Lazarević, D.; Dvornik, J.; Fresl. K.: *Analiza oštećenja atrija Kneževa dvora u Dubrovniku*, *Građevinar*, **56** (2004) 10, str. 601. – 612.
- [30] Lazarević, D.; Anđelić, M.; Uroš. M.: *Oblikovanje i proračun kupole dvorane “Krešimir Čosić” u Zadru*, *Građevinar*, **62** (2010) 10, str. 875. – 886.
- [31] Hrženjak, P.; Petzel, M.; Vujec. S.: *Kontrolna mjerenja i numeričke analize za dimenzioniranje stupova i komora pri podzemnom otkopavanju arhitektonsko–građevnog kamena*, Znanstvenostručno savjetovanje s međunarodnim sudjelovanjem “Mehanika stijena i tuneli”, Zagreb 30. 09.–2. 10. 1999. (urednici Jašarević, I.; Hudec, M.; Vujec, S.), Vol. 1, str. 127. – 133.

- [32] Uroš, M.; Gidak, P.; Lazarević, D.: *Optimization of stadium roof structure using force density method*, Proceedings of the third international conference on structures and architecture (ICSA2016) – Structures and Architecture – Beyond their Limits (editor Paulo J.S. Cruz), CRC Press/Balkema, Taylor & Francis Group, Guimaraes, 2016. str. 693. – 700.
- [33] Anđelić, M.: *Konstrukcija košarkaške dvorane "Cibona"*, Zbornik II. kongresa Društva građevinskih konstruktora Hrvatske, Split, 1986., str. 19. – 23.
- [34] Lazarević, D.; Anđelić, M.; Atalić, J.: *Projekt konstrukcije nadogradnje zgrade Euroherca u Zagrebu*, Građevinar, **63** (2011) 12, str. 1021. – 1032.
- [35] *Studija za saniranje posljedica potresa. 5. faza*, (voditelj projekta J. Atalić), Sveučilište u Zagrebu, Građevinski fakultet, Zavod za tehničku mehaniku, Zagreb, 2017., str. 87. – 111.
- [36] Anđelić, M.; Lazarević, D.; Uroš, M.: *Analiza postojećeg stanja i prijedlog sanacije temelja agregata u tvornici "SMIT transformers" u Nijmegenu, Nizozemska*, Končar, inženjering za energetiku i transport, d. d., Zagreb, 2014.
- [37] Jaguljnjak Lazarević, A.; Dvornik, J.; Frgić, L.: *Utjecaj pogreške zaokruživanja na točnost proračuna konstrukcije*, Građevinar, **63** (2011) 11, str. 911. – 921.
- [38] Barzilai, J.; Borwein, J. M.: *Two-Point Step Size Gradient Methods*, IMA Journal of Numerical Analysis, **8** (1988) 1, str. 141. – 148.
- [39] Fresl, K.; Šamec, E.; Gidak, P.: *O optimizaciji programskoga kôda iteriranoga Ritzova postupka*, Zbornik mini-simpozija "Novi, učinkoviti iteracijski postupak proračuna konstrukcija – poopćenje suvremenih postupaka" (ovaj zbornik), Građevinski fakultet Sveučilišta u Zagrebu, Zagreb, 2019., str. 39.-52.