

ALGORITEM ZA PRETVORBO TELESA PREDSTAVLJENEGA Z OVOJNICO V PREDSTAVITEV Z OSMIŠKIM DREVESOM

INFORMATICA 2/91

Keywords: geometric modeling, boundary representation, octree representation, computer graphics

Natalija Trstenjak,
Borut Žalik, Nikola Guid
Tehniška fakulteta Maribor
Smetanova 17, 62000 Maribor
Slovenija, Jugoslavija

Predstavitev teles z ovojnico je široko uporabljena v računalniški grafiki in predstavlja tudi primarno predstavitev v Eksperimentalnem modelirniku teles, ki smo ga razvili. Vendar pa je izvajanje Boolovih operatorjev, ki so standardno orodje vsakega modelirnika, zelo težavno nad telesi predstavljenimi z ovojnico. Zato smo kot sekundarno predstavitev v modelirniku uvedli predstavitev z osmiškimi drevesi, ki je za izvajanje Boolovih operatorjev prav idealna. V tem članku obravnavamo pretvorbo predstavitev z ovojnico poljubnega telesa v predstavitev z osmiškim drevesom. Podrobno smo predstavili iskanje sečišč med oktantom in poljubnim telesom, ki je najbolj kritičen del pretvorbe.

An Algorithm for Boundary to Octree Conversion

The boundary representation is widely used in computer graphics and it represents a primary representation in our Experimental geometric modeler. An implementation of the regularized Boolean operators which are the standard tools in the solid modelers is very complicated in the case of boundary representation. For this reason the octree representation is included as a secondary representation and it is almost ideal for the implementation of Boolean operators. In this paper, we consider the problem of converting the boundary representation of a polyhedron to its octree representation. The critical work in the subdivision process is to determine how an octant intersects the polyhedron and this problem is described in details.

1 Uvod

Osnovni problem geometrijskega modeliranja je v opisovanju tridimenzionalnega zveznega telesa v digitalnem računalniku tako, da je predstavitev nedvoumna [HILL82]. Znanih je več nedvoumnih predstavitev, najpogostejše pa so predstavitev z ovojnico ("boundary representation", B-rep), predstavitev s temeljnimi gradniki ("constructive solid

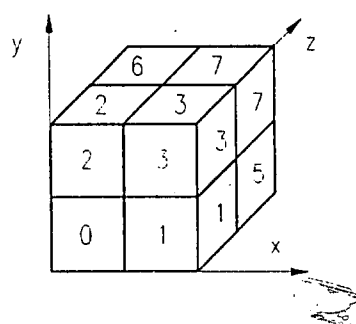
geometry", CSG) in z osmiškimi drevesi ("octree representation") ([REQU82], [REQU83], [MORT85]). Vsaka izmed predstavitev ima svoje slabosti in prednosti. Zato večina komercialnih modelirnikov temelji na takolimenovani hibridni predstavitvi, ki jo dobimo s kombinacijo vsaj dveh različnih predstavitev. Idealno arhitekturo bi tvorile vse tri prej omenjene predstavitve [MILL89]. Za hibridno predstavitev so izjemno pomembni učinkoviti pretvorbeni algoritmi iz

ene predstavitev v drugo. V prispevku bomo podali pretvorbeni algoritem iz predstavitev z ovojnico v predstavitev z osmiškim drevesom.

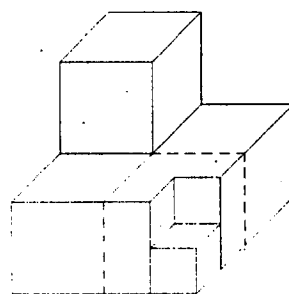
Predstavitev z ovojnico hrani podatke o površju telesa in je sestavljena iz oglišč, robov in ploskev. Podatkovna struktura mora hraniti tako sosednostne relacije kot geometrijske podatke, kar zahteva pazljivo načrtovanje podatkovne strukture. Ker je vsaka ploskev predstavljena eksplicitno, omogoča dodajanje specifičnih informacij vsaki ploskvi telesa (npr. barva, kvaliteta materiala), prav tako pa v principu ni težav pri vgradnji poljubno oblikovanih ploskev [CHIY87]. Pri vizualizaciji predstavljenega telesa večjih težav ni, saj lahko žični model prikazemo zelo hitro, mnogo klasičnih algoritmov računalniške grafike za odstranjevanje zakritih robov, senčenje in uporabljanje pa temelji na tej predstavitvi [PRAT83]. Telesa lahko modeliramo z nizko - nivojskimi operatorji, npr. Eulerjevimi operatorji, z uporabo višje - nivojskih lokalnih operatorjev in z Boolovimi operatorji ([MANT82], [WILS85], [CHIY85], [MANT88]). Prav implementacija Boolovih operatorjev, ki so standardno orodje vsakega geometrijskega modelirnika, pa je v predstavitvi z ovojnico zelo težavno [MANT83].

Predstavitev z osmiškim drevesom temelji na principu rekurzivne delitve prostora. Rezultat rekurzivnega delitvenega procesa je predstavljen z drevesom osme stopnje, katerega vozlišča so ali listi ali sinovi. Korensko vozlišče v izhodnem drevesu predstavlja vhodni prostor. Končna vozlišča-listi predstavljajo oktante, ki vsebujejo enotske kocke iste barve in so označena s črna ali bela, odvisno od tega ali se nahajajo znotraj ali zunaj telesa. Vmesna vozlišča imenujemo notranja vozlišča in so označena s siva. Ta predstavljajo oktante v prostoru osmiškega drevesa, ki ležijo delno znotraj delno zunaj telesa in jih v rekurzivnem postopku delimo na podoktante. Vsak oktant ima predpisano osmiško kodo, ki določa njegov položaj glede na oktanta očeta. Osmiške kode oktantov so prikazane na sliki 1a, primer telesa in njegove predstavitev pa kažeta sliki 1b in 1c.

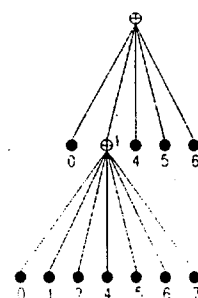
Predstavitev z osmiškimi drevesi je prav idealna za izvajanje Boolovih operacij. Vsa izračunavanja pri Boolovih operacijah temeljijo na celoštevilski aritmetiki (vse potrebne zaokrožitve smo namreč odpravili že v fazi tvorbe osmiškega drevesa), zato so algoritmi izredno robustni. Glavna slabost je v tem, ker meja telesa ni eksplicitno znana, zato ni možno hitro prikazati npr. žičnega modela objekta; direktno jih lahko vizualiziramo samo s tehnikami sledenja žarku ("ray tracing"). Prav tako predstavitev zahteva mnogo pomnilnika, s tem pa je omejena tudi resolucija objekta [CLIF87]. Za predstavitev z osmiškimi drevesi razvijajo zaradi njenih dobrih lastnosti tudi specialno namensko aparaturno opremo [KAUF88].



a) Osmiške kode oktantov (oktant 4 je skrit)



b) Primer telesa



c) Njegova predstavitev z osmiškim drevesom

Slika 1 Osmiške kode oktantov ter predstavitev telesa z osmiškim drevesom

Naš Eksperimentalni geometrijski modelirnik (EGM) ima primarno predstavitev z ovojnico, sekundarno pa z osmiškimi drevesi. Predstavitev z ovojnico podpira 14 vgrajenih Eulerjevih operatorjev, domena teles pa vključuje telesa z ravnimi ploskvami [ŽALI89a]. Višje nivojski operatorji omogočajo tvorbo teles s translacijskim pomikanjem in z operatorji, ki upoštevajo značilnosti teles [ŽALI90]. Vgrajeni so tudi Boolovi operatorji, ki pa so uspešni le nad konveksni telesi z lici, ki ne vsebujejo prstanov [ŽALI89b]. Zaradi tega smo se odločili, da bomo izdelali algoritem za pretvorbo predstavitev z ovojnico v predstavitev z osmiškimi drevesi.

2 Tehnike za tvorbo osmiškega drevesa iz predstavitev z ovojnico

Poznamo različne tehnike za tvorbo osmiškega drevesa, ki se razlikujejo med seboj po obliki vhodnih podatkov. Vhodni podatki so različne predstavitve 3D teles, kot so na primer 3D polja, zaporedne sekcije, slike obrisa objekta, globinske slike in modeli objektov. Med modele objektov spada tudi predstavitev z

ovojnico. Poznamo dve tehniki za tvorbo osmiškega drevesa telesa, predstavljenega z ovojnico: tehnika označevanja komponent in tehnika deli-in-vladaaj.

Tehnika označevanja komponent temelji na principu polnjenja notranjosti zaprte meje, zato ne potrebuje testiranja presečišč. Algoritem je sestavljen iz dveh delov. V prvem delu pretvorimo mejo objekta v delno 3D binarno drevo. Črni listi drevesa predstavljajo celice, ki sekajo mejo objekta. V drugem delu delno binarno drevo prečno obiščemo in tiste nedoločene bele liste, ki odgovarjajo notranjosti ali zunanosti telesa, klasificiramo v črna ali bela vozlišča, skladno s tehniko označevanja. Algoritem generira binarno verzijo osmiškega drevesa, ki jo lahko enostavno pretvorimo v osmiško drevo.

Druga tehnika za pretvorbo poljubnega telesa, predstavljenega z ovojnico v predstavitev z osmiškim drevesom, je tehnika deli_in_vladaaj, ki jo bomo v tem članku podrobno predstavili. Ta tehnika velja za grob pristop in pričenja z enojnim črnim vozliščem - korenom drevesa kot začetno aproksimacijo poljubnega telesa in nato nadaljuje z rekurzivno delitvijo prostora osmiškega drevesa. Algoritem je predstavljen na sliki 2.

```

Deli_in_vladaaj ( vozlišče )
( * M : meja - obris telesa * )

Begin
  Case PRESEČIŠČE ( vozlišče, M ) of
    BELA : opusti vozlišče in vkoreninjeno
           poddrevo
    ČRNA : vozlišče.barva = ČRNA
    SIVA : begin
             vozlišče.barva = SIVA
             for i = 0 to 7
               Deli_in_vladaaj ( vozlišče.sin[i] )
             end
           end
  end
End

```

Slika 2 Algoritem tehnike deli-in-vladaaj

Kritičen del v procesu delitve je določiti, kako oktant seka poljubno telo. To lahko opravimo v dveh korakih: v prvem koraku pregledamo ali oktant seka katero koli ploskev poljubnega telesa. Če seka, potem je oktant delno zunaj, delno znotraj telesa, pripadajoče vozlišče v osmiškem drevesu označimo s siva in oktant delimo na podoktante. Drugače je oktant ali zunaj ali znotraj telesa. V drugem koraku, katerega bistvo je vsebnostni test, določimo, katera od trditev je pravilna. Če za katero koli oglišče oktanta ugotovimo, da je obdano s površino poljubnega telesa, se oktant nahaja znotraj telesa, vozlišče osmiškega drevesa označimo s črna, drugače pa se nahaja zunaj telesa in vozlišče osmiškega drevesa označimo z bela [HOME88].

3 Ugotavljanje presečišča med oktantom in poljubnim telesom

Testiranje presečišča med oktantom in poljubnim telesom je v splošnem počasen proces. Da bi povečali hitrost testiranja presečišč, uvedemo izločevalni postopek, ki sestoji iz petih korakov oziroma testov:

- testa minimax,
- testiranja ravnin ploskev telesa,
- testiranja oglišč ploskev telesa,
- testiranja presečnega mnogokotnika,
- testiranja lukenj ploskve telesa.

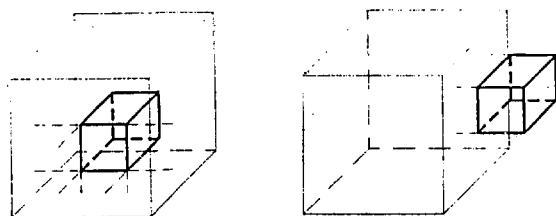
Namen tega postopka je v prvih dveh korakih zmanjšati število ploskev telesa, ki bodo prišle v naslednji korak. Izločamo ploskve, ki zagotovo ležijo zunaj oktanta in ga zato ne morejo sekati. Če je število vseh izločenih ploskev enako številu vseh ploskev telesa, v naslednji korak ne vstopimo, saj lahko v takem primeru določimo barvo vozlišča oziroma presečišča z vsebnostnim testom. V tretjem koraku želimo ugotoviti, ali oktant seka katero od ploskev telesa, ki jih nismo izločili in s tem določiti barvo vozlišča. Vsak nadaljnji korak obravnava presečišče bolj podrobno in v zadnjem, petem koraku, barvo vozlišča osmiškega drevesa, ki predstavlja pripadajoč oktant v prostoru osmiškega drevesa, zagotovo določimo.

V podglavljih, ki sledijo, je podrobno obdelan vsak korak in zaključki, ki sledijo iz rezultatov testiranja v omenjenem koraku.

3.1 Test minimax

S testom minimax izločimo tiste ploskve telesa, ki zagotovo ne sekajo oktanta. Test je razširjen na 3D prostor, z njim pa ugotavljamo prekrivanje omejitvenih škatel ("bounding box") ploskev telesa in oktanta (omejitvena škatla ploskve je pravokotna škatla, katere robovi so vzporedni z osmi prostora osmiškega drevesa in omejuje ploskev). To je prva groba selekcija ploskev, ki ne sekajo oktanta, saj zaradi narave testa minimax ne moremo trditi, da smo izločili vse takšne ploskve iz nadaljnjega postopka [GUID88]. Kljub temu pa s tem testom izločimo veliko število ploskev, še posebej pri večji globini osmiškega drevesa, ko je lahko oktant napram telesu zelo majhen.

Če s testom minimax izločimo vse ploskve telesa, kar pomeni, da vse ploskve ležijo zunaj oktanta, se lahko oktant nahaja strogo znotraj telesa ali strogo zunaj telesa. Primer je podan na sliki 3, kjer je oktant narisan z močnejšo in telo s tanjšo črto. Da bi določili, katera od trditev je pravilna, uporabimo vsebnostni test oglišč oktanta v telesu.



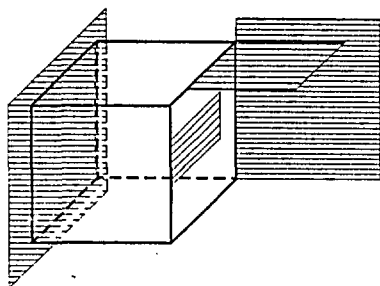
a) Oktant strogo znotraj telesa b) Oktant strogo zunaj telesa

Slika 3 S testom minimax smo izločili vse ploskve telesa

Če s testom minimax nismo izločili vseh ploskev telesa, oziroma ne moremo trditi, da ležijo vse zunaj oktanta, ne moremo določiti barve vozlišča osmiškega drevesa. Ploskve, ki jih nismo izločili, testiramo v drugem koraku, ki vključuje test ravnin ploskev.

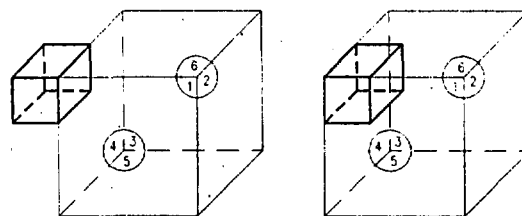
3.2 Test ravnin ploskev telesa

S tem testom izločimo ploskve telesa, ki se oktanta dotikajo. Za vsako ploskev telesa izračunamo enačbo ravnine, v kateri leži, in jo primerjamo z enačbami ravnin, v katerih ležijo ploskve oktanta. Če ravnina ploskve telesa sovpada s katero od ravnin ploskev oktanta, leži ploskev telesa na eni od ploskev oktanta ali pa se oktanta dotika v enem od njegovih robov. Primer je podan na sliki 4, kjer so ploskve telesa označene s vodoravno šrafuro. V tem primeru se ploskev in oktant zagotovo ne sekata in ploskev izločimo iz nadaljnjega postopka.



Slika 4 Ploskve telesa, katerih ravnine sovpadajo z ravninami ploskev oktanta

Če je vsota ploskev, ki smo jih izločili v prvem koraku, in ploskev, katere smo izločili v drugem koraku, enaka številu vseh ploskev telesa, se lahko oktant nahaja le zunaj ali znotraj telesa (slika 5). Uporabimo vsebnostni test, da ugotovimo, katera od navedenih trditev je pravilna in lahko skladno s tehniko označevanja določimo barvo vozlišča osmiškega drevesa.



a) Oktant zunaj telesa b) Oktant znotraj telesa

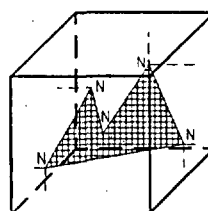
Slika 5 Ploskve telesa 2, 3, 5 smo izločili s testom minimax, ravnine ploskev 1, 4, 6 sovpadajo z ravninami oktanta

Če vsota do sedaj izločenih ploskev ni enaka številu vseh ploskev telesa, je barva presečišča odvisna od lege ploskev, ki jih še nismo izločili. Te ploskve testiramo v naslednjem koraku, ki vključuje test oglišč ploskev.

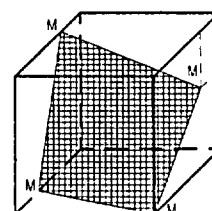
3.3 Test oglišč ploskev telesa

V tem koraku ne izločamo ploskev, ampak želimo ugotoviti, ali katera od ploskev zagotovo seka oktant. Ploskve telesa, ki jih testiramo v tem koraku, lahko:

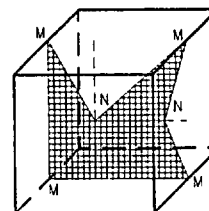
- ležijo znotraj oktanta,
- sekajo oktant,
- ležijo zunaj oktanta.



a) $OP=NP$; $MP=0, ZP=0$



b) $OP=MP$; $NP=0, ZP=0$



c) $OP=MP+NP$; $MP \neq 0, NP \neq 0, ZP=0$

Slika 6 Ploskev leži znotraj oktanta

Ploskev zagotovo seka oktant, če leži v celoti znotraj oktanta, ali pa seka oktant, ne glede na to ali je konveksna ali konkavna in ali ima luknje ali ne. Ta dva položaja ploskve lahko enostavno določimo na podlagi lege njenih oglišč.

Naj je OP število vseh oglišč ploskve telesa. Število oglišč ploskve zunaj oktanta označimo z ZP, število oglišč znotraj oktanta z NP in število oglišč ploskve na meji oktanta z MP.

Če se vsa oglišča ploskve telesa nahajajo znotraj ali na meji oktanta, oziroma če velja

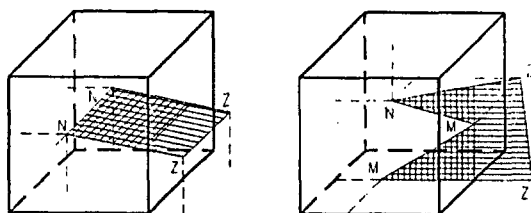
$$OP = NP + MP ; ZP = 0 \quad (1)$$

lahko sklepamo, da se ploskev telesa nahaja znotraj oktanta (slika 6). Ploskev ne more ležati na eni izmed ploskev oktanta, kljub temu, da lahko vsa oglišča ploskve telesa ležijo na meji oktanta, saj bi ploskev izločili že v drugem koraku, s testom ravnin ploskev telesa (slika 4).

Če se vsaj eno oglišče ploskve telesa nahaja zunaj oktanta in vsaj eno oglišče znotraj oktanta, oziroma če velja

$$OP = NP + MP + ZP ; ZP \neq 0, NP \neq 0 \quad (2)$$

ploskev telesa seka oktant, ne glede na to ali je konveksna ali konkavna in ali ima luknje ali ne (slika 7).



$$\begin{aligned} \text{a) } OP &= NP + ZP \\ MP &= 0, NP \neq 0, ZP \neq 0 \end{aligned}$$

$$\begin{aligned} \text{b) } OP &= NP + MP + ZP \\ NP &\neq 0, MP \neq 0, ZP \neq 0 \end{aligned}$$

Slika 7 Ploskev seka oktant

Ko ugotovimo, da se vsaj ena od ploskev telesa nahaja znotraj oktanta ali seka oktant, je barva vozlišča osmiškega drevesa siva ne glede na položaj ostalih ploskev telesa glede na oktant.

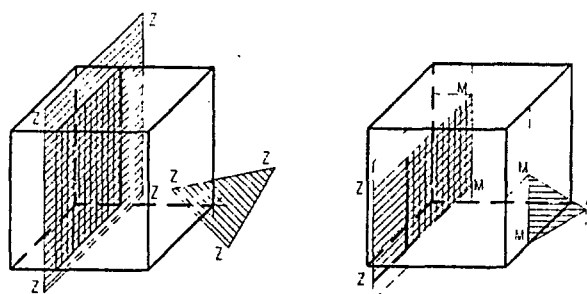
Če za nobeno od ploskev telesa, ki smo jih testirali v tem koraku ne velja enačba 1 ali enačba 2, ne moremo trditi, da katera od ploskev seka oktant, ali da so vse ploskve zunaj oktanta, zato nadaljujemo s četrtim korakom, ki vključuje test presečnega mnogokotnika.

3.4 Test presečnega mnogokotnika

Za ploskve telesa, ki jih testiramo v tem koraku velja, da imajo vsaj eno oglišče zunaj oktanta in nobenega oglišča znotraj oktanta oziroma velja

$$OP = MP + ZP ; NP = 0, ZP \neq 0 \quad (3)$$

Ploskve telesa lahko ležijo zunaj oktanta ali sekajo oktant (slika 8), na rezultat testa pa lahko vpliva tudi konkavnost ploskve.



$$\text{a) } OP = ZP ; MP = 0, NP = 0$$

$$\begin{aligned} \text{b) } OP &= MP + ZP \\ NP &= 0, ZP \neq 0, MP \neq 0 \end{aligned}$$

Slika 8 Ploskve telesa ležijo zunaj oktanta ali sekajo oktant

Cilj tega testa je isti kot v tretjem koraku, le da je postopek ugotavljanja presečišč bolj natančen, saj moramo upoštevati, da ima ploskev lahko luknje in ni nujno konveksna. Ta test sestoji iz štirih korakov - testov, ki se nanašajo na posamezno ploskev.

Da bi ugotovili položaj ploskve telesa, si pomagamo s presečnim mnogokotnikom, ki ga določajo točke, v katerih ravnina ploskve telesa seka oktant. Ravnina ploskve telesa seka ravnine šestih ploskev oktanta v šestih premicah. Točke, v katerih se teh šest premic seka, določajo oglišča presečnega mnogokotnika, ki leži v ravnini ploskve telesa. Ploskev telesa seka oktant le takrat, ko seka presečni mnogokotnik.

V prvem koraku testiramo število oglišč presečnega mnogokotnika, ki je odvisno od položaja ravnine ploskve telesa glede na oktant. Na podlagi rezultatov tega testa, izločimo ploskve, ki ležijo zunaj oktanta.

Naj je OPM število vseh oglišč presečnega mnogokotnika. Ko je število oglišč presečnega mnogokotnika manjše ali enako dva, oziroma če velja

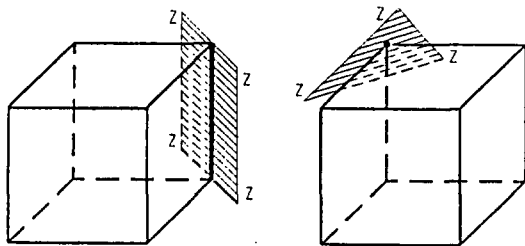
$$OPM \leq 2 ; OPM \neq 0 \quad (4)$$

se ravnina ploskve telesa dotika oktanta. Ko je število oglišč presečnega mnogokotnika enako nič, oziroma če velja

$$OPM = 0$$

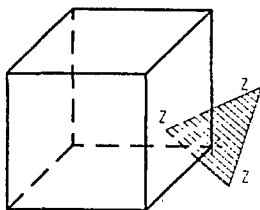
(5)

leži ravnina ploskve telesa zunaj oktanta. Na podlagi tega sklepamo, da ploskev ne seka oktanta, temveč zagotovo leži zunaj oktanta (slika 9).



a) OPM = 2

b) OPM = 1



c) OPM = 0

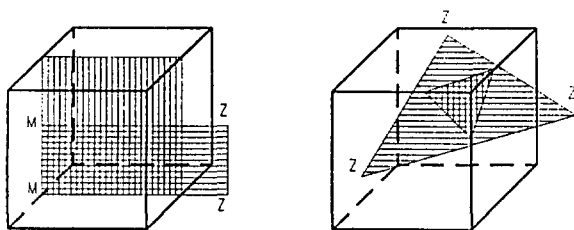
Slika 9 Ploskev telesa leži zunaj oktanta

Ko je število oglišč presečnega mnogokotnika večje od dva, oziroma velja:

$$OPM > 2$$

(6)

ravnina ploskve seka oktant in preidemo na drugi korak (slika 10).

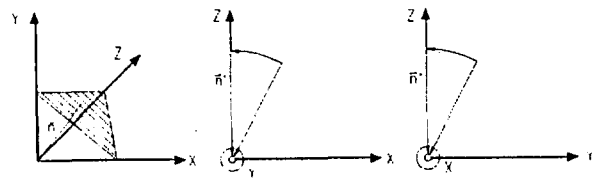


a) OPM = 4

b) OPM = 3

Slika 10 Ploskev telesa seka oktant

V drugem koraku moramo določiti položaj presečnega mnogokotnika glede na ploskev telesa. Da bi poenostavili vsebnostni test, s katerim določamo položaj oglišč presečnega mnogokotnika, zavrtimo ravnino, v kateri ležita ploskev in presečni mnogokotnik tako, da je vzporedna s koordinatno ravnino XY. To opravimo tako, da zavrtimo normalni vektor ploskve telesa v koordinatno os Z (slika 11).



Slika 11 Rotacija normale ploskve telesa v koordinatno os Z

Sedaj opravimo vsebnostni test, da ugotovimo položaj presečnega mnogokotnika glede na ploskev telesa. Presečni mnogokotnik je na vseh nadaljnjih slikah označen z navpično šrafuro, ploskev telesa pa z vodoravno šrafuro.

Naj bo ZPM število oglišč presečnega mnogokotnika zunaj ploskve, MPM naj bo število oglišč na meji ploskve in NPM število oglišč znotraj ploskve telesa. Presečni mnogokotnik lahko leži:

- strogo znotraj ploskve telesa,
- seka ploskev telesa,
- znotraj ploskve telesa,
- zunaj ploskve telesa.

Če so vsa oglišča presečnega mnogokotnika znotraj ploskve, oziroma če velja

$$OPM = NPM ; MPM = 0, ZPM = 0$$

(7)

potem je presečni mnogokotnik strogo znotraj ploskve telesa (slika 12), vendar barve vozlišča ne moremo določiti, saj je ta odvisna od medsebojne lege presečnega mnogokotnika in lukenj v ploskvi.

Če ploskev nima lukenj, potem ploskev seka oktant (slika 12a) in je barva vozlišča osmiškega drevesa siva. V primeru, ko ima ploskev luknje (slika 12b), uporabimo test lukenj ploskve telesa in določimo položaj ploskve glede na oktant.

V primeru, ko enačba 7 ni izpolnjena, preidemo na tretji korak. V tem koraku želimo ugotoviti, ali presečni mnogokotnik seka ploskev telesa.

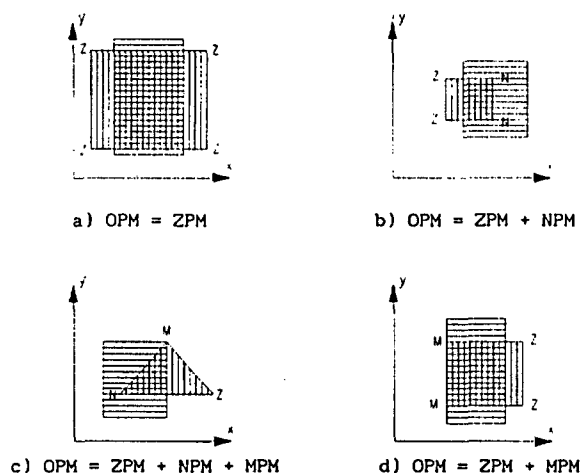


a) Ploskev nima luknje

b) Ploskev z luknjo

Slika 12 Presečni mnogokotnik strogo znotraj ploskve telesa

Če vsaj en rob presečnega mnogokotnika seka kateri koli rob ploskve telesa, presečni mnogokotnik seka ploskev (slika 13), ne glede na to ali je ploskev konkavna ali konveksna in ali ima luknje ali ne.



Slika 13 Presečni mnogokotnik seka ploskev telesa

Ploskev telesa seka oktant, barva vozlišča osmiškega drevesa je siva in prekinemo s testiranjem presečišč.

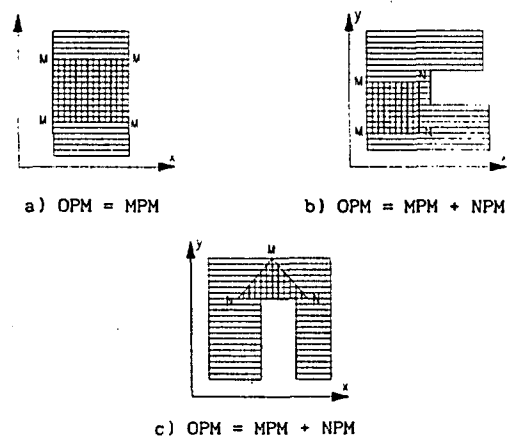
Če noben rob presečnega mnogokotnika ne seka katerega koli roba ploskve telesa, preidemo na četrti korak.

V četrtem koraku ugotavljamo položaj presečnega mnogokotnika glede na ploskev telesa na osnovi lege njegovih oglišč.

Če so vsa oglišča presečnega mnogokotnika na meji ploskve telesa, oziroma če velja

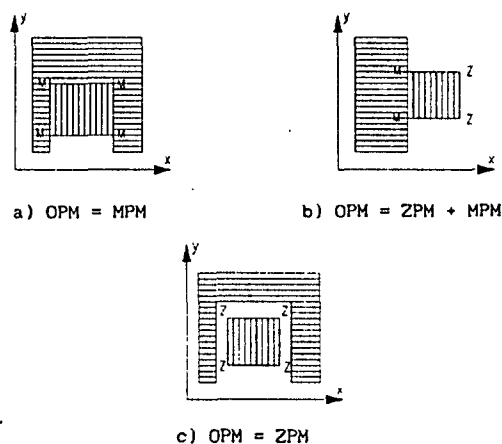
$$OPM = MPM ; NPM = 0, ZPM = 0 \quad (8)$$

Je rezultat testa odvisen od lege središčne točke presečnega mnogokotnika (slika 14a, 15a).



Slika 14 Presečni mnogokotnik znotraj ploskve telesa

Če središčna točka presečnega mnogokotnika leži znotraj ploskve telesa, leži tudi presečni mnogokotnik znotraj ploskve in ploskev seka oktant (slika 14a). Barva presečišča je siva in prekinemo s testiranjem presečišč. Če leži središčna točka presečnega mnogokotnika zunaj ploskve, leži tudi presečni mnogokotnik zunaj ploskve telesa in ploskev ne seka oktanta (slika 15a).



Slika 15 Presečni mnogokotnik zunaj ploskve telesa

Če je vsaj eno oglišče presečnega mnogokotnika znotraj ploskve, oziroma če velja

$$OPM = NPM + MPM ; ZPM = 0, NPM \neq 0 \quad (9)$$

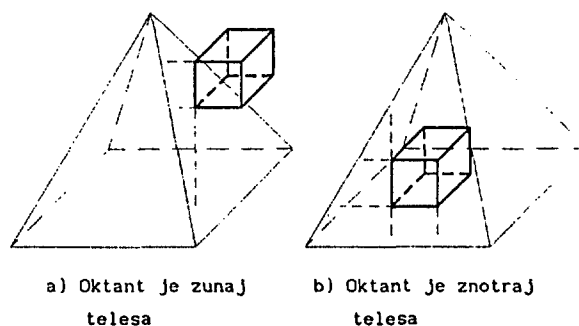
leži presečni mnogokotnik znotraj ploskve in ploskev seka oktant (slika 14b,c). Barva presečišča, oziroma vozlišča osmiškega drevesa je siva in prekinemo s testiranjem presečišč.

Če je vsaj eno oglišče presečnega mnogokotnika zunaj ploskve telesa, oziroma če velja

$$OPM = ZPM + MPM ; ZPM \neq 0, NPM = 0 \quad (10)$$

leži presečni mnogokotnik zunaj ploskve (slika 15b,c) in ploskev ne seka oktanta.

Rezultat testa presečnega mnogokotnika nam vrne barvo presečišča oziroma vozlišča osmiškega drevesa, če vsaj ena ploskev telesa seka oktant. V primeru, ko nismo dobili barve presečišča, so vse ploskve, ki smo jih testirali v tem koraku, zunaj oktanta (slika 16) in oktant lahko leži znotraj telesa, ali zunaj telesa.

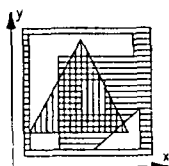


Slika 16 Medsebojna lega telesa in oktanta

Uporabimo vsebnostni test oglišč oktanta v telesu, da bi ugotovili, katera od trditev je pravilna. Nato skladno s tehniko označevanja predpišemo vozlišču osmiškega drevesa črno ali belo barvo.

3.5 Test lukenj ploskve telesa

Presečni mnogokotnik je strogo znotraj ploskve, ko velja enačba 7. Samo v tem primeru lahko luknja ploskve prekriva presečni mnogokotnik (slika 12b) in ploskev ne seka oktanta. Luknja v ploskvi se ne more dotikati roba ploskve niti sekati ploskve, zato v primerih, ko enačba 7 ne velja, presečni mnogokotnik ne more biti v celoti prekrit z luknjo (slike 13, 14, 15). Ploskev ima lahko več lukenj (slika 17), zato test ponavljamo za vsako luknjo v ploskvi, dokler ne določimo barve presečišča.



Slika 17 Ploskev telesa z luknjami in lega presečnega mnogokotnika

Ugotoviti moramo ali luknja v celoti prekriva presečni mnogokotnik, saj v vseh ostalih primerih ploskev vsaj delno prekriva presečni mnogokotnik, kar pomeni, da ploskev seka oktant.

Presečni mnogokotnik lahko:

- seka luknjo,
- leži znotraj luknje,
- prekriva luknjo,
- leži strogo zunaj luknje.

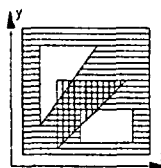
Določanje medsebojne lege luknje ploskve in presečnega mnogokotnika poteka v treh korakih.

V prvem koraku uporabimo test minimax, s katerim določimo prekrivanje omejitvenih pravokotnikov presečnega mnogokotnika in luknje ploskve. Če se omejitvena pravokotnika ne prekrivata, nadaljujemo z naslednjo luknjo ploskve. Če za vse luknje ploskve velja, da se njihovi omejitveni pravokotniki ne prekrivajo z omejitvenim pravokotnikom presečnega mnogokotnika, potem presečni mnogokotnik seka, oziroma leži na ploskvi telesa in ta seka oktant. Če se omejitvena pravokotnika luknje ploskve in presečnega mnogokotnika prekrivata, preidemo na naslednji korak.

V drugem koraku preverjamo, ali kateri od robov presečnega mnogokotnika seka kateri koli rob luknje ploskve. Če je test pritrdilen, leži presečni mnogokotnik delno na ploskvi telesa in ploskev seka oktant ne glede na to, ali ima ploskev še kakšno luknjo in kje na ploskvi se ta luknja nahaja (slika 18). Zato lahko v tem primeru zaključimo s testom lukenj ploskve.

Če je test sekanja robov negativen, nadaljujemo s tretjim korakom.

V tretjem koraku želimo ugotoviti, ali luknja ploskve popolnoma prekriva presečni mnogokotnik



Slika 18 Presečni mnogokotnik seka luknjo ploskve

oziroma, ali leži presečni mnogokotnik v celoti znotraj luknje ploskve. V tem primeru ploskev ne seka oktanta, ne glede na ostale luknje ploskve in lahko zaključimo s testiranjem. Če omejitveni pravokotnik presečnega mnogokotnika omejuje omejitveni pravokotnik luknje ploskve, ugotavljamo vsebnost luknje ploskve v presečnem mnogokotniku, drugače pa ugotavljamo vsebnost presečnega mnogokotnika znotraj luknje ploskve. Ta vsebnostni test je enak v obeh primerih, zato ga lahko posplošimo na ugotavljanje vsebnosti mnogokotnika M1 v mnogokotniku M2.

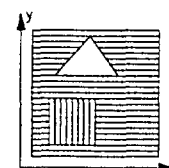
Naj bo OM1 število oglišč mnogokotnika M1 in OM2 število oglišč mnogokotnika M2. Število oglišč mnogokotnika M1 zunaj mnogokotnika M2 označimo z ZM1, število oglišč M1 znotraj M2 označimo z NM1, število oglišč, ki ležijo v ogliščih M2 označimo z VM1 in število oglišč, ki ležijo na robovih M2 označimo z RM1. Če vsa oglišča M1 ležijo v ogliščih M2, oziroma če velja

$$OM1 = VM1 ; RM1 = 0, ZM1 = 0, NM1 = 0 \quad (11)$$

in

$$OM1 = OM2 \quad (12)$$

sta mnogokotnika enaka in se popolnoma prekrivata. V tem primeru leži presečni mnogokotnik v celoti znotraj luknje ploskve, ne glede na to kateri mnogokotnik označuje presečni mnogokotnik in kateri luknjo ploskve (slika 19). Ploskev telesa ne seka oktanta in prekinemo s testiranjem.

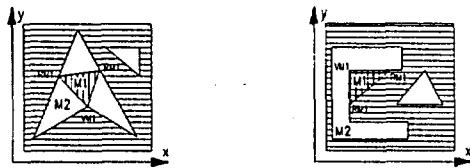


Slika 19 Mnogokotnika se popolnoma prekrivata

Če vsa oglišča mnogokotnika M1 ležijo v ogliščih M2 in na robovih M2, oziroma če velja

$$OM1 = VM1 + RM1 ; ZM1 = 0, NM1 = 0 \quad (13)$$

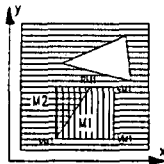
je mnogokotnik M1 lahko zunaj ali znotraj mnogokotnika M2. Uporabimo vsebnostni test središčne točke mnogokotnika M1 v mnogokotniku M2, da ugotovimo, katera trditev je pravilna.



a) Presečni mnogokotnik znotraj luknje b) Presečni mnogokotnik zunaj luknje

Slika 20 M1 presečni mnogokotnik, M2 luknja ploskve

V primeru, ko M1 označuje presečni mnogokotnik in M2 luknjo ploskve (slika 20), imamo naslednjo rešitev: Če presečni mnogokotnik leži znotraj luknje (slika 20a), ploskev ne seka oktanta. Če leži presečni mnogokotnik zunaj luknje ploskve, ploskev seka oktant (slika 20b).



Slika 21 M1 luknja ploskve, M2 presečni mnogokotnik

V primeru, ko M1 označuje luknjo ploskve in M2 presečni mnogokotnik, je rešitev naslednja: luknja ploskve lahko leži le znotraj presečnega mnogokotnika, ker je oblika presečnega mnogokotnika omejena. Zato presečni mnogokotnik leži delno na ploskvi telesa in ta seka oktant (slika 21).

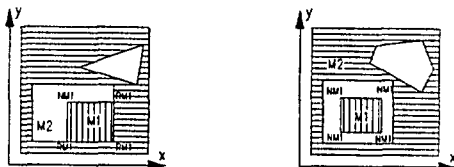
Če vsaj eno oglišče mnogokotnika M1 leži znotraj mnogokotnika M2, oziroma če velja

$$OM1 = NM1 + RM1 + VM1 ; NM1 \neq 0, ZM1 = 0 \quad (14)$$

leži mnogokotnik M1 znotraj mnogokotnika M2.

Če leži presečni mnogokotnik (M1) znotraj luknje ploskve (M2), luknja ploskve prekriva presečni mnogokotnik in ploskev ne seka oktanta (slika 22).

Če leži luknja ploskve (M1) znotraj presečnega mnogokotnika (M2), leži presečni mnogokotnik delno na ploskvi telesa in ploskev seka oktant (slika 23).

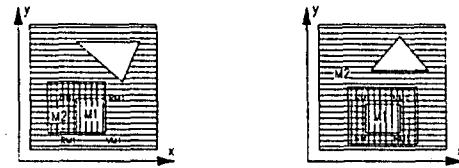


a) $RM1 \neq 0, VM1 \neq 0$ b) $RM1 = 0, VM1 = 0$

Slika 22 Presečni mnogokotnik (M1) znotraj luknje ploskve (M2)

Če vsaj eno oglišče mnogokotnika M1 leži zunaj mnogokotnika M2, oziroma če velja

$$OM1 = ZM1 + RM1 + VM1 ; ZM1 \neq 0, NM1 = 0 \quad (15)$$



a) $RM1 \neq 0, VM1 \neq 0$ b) $RM1 = 0, VM1 = 0$

Slika 23 Luknja ploskve (M1) znotraj presečnega mnogokotnika (M2)

in

$$RM1 \neq 0 \text{ ali } VM1 \neq 0 \quad (16)$$

leži mnogokotnik M1 zunaj mnogokotnika M2 in mnogokotnika se dotikata. V tem primeru, ne glede na to, kateri mnogokotnik določa luknjo ploskve in kateri presečni mnogokotnik, ugotovimo, da presečni mnogokotnik leži zunaj luknje ploskve, vendar ga nobena druga luknja ploskve ne more popolnoma prekrivati. Ploskev seka oktant in prekinemo s testiranjem lukenj (slika 24).



a) Presečni mnogokotnik (M1) luknja ploskve (M2) b) Luknja ploskve (M1) presečni mnogokotnik (M2)

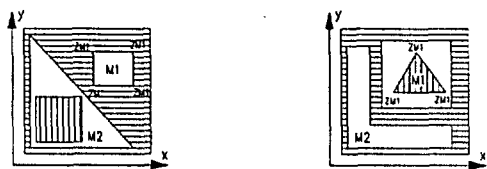
Slika 24 Mnogokotnik M1 zunaj mnogokotnika M2

Če vsa oglišča mnogokotnika M1 ležijo zunaj mnogokotnika M2 oziroma, če velja

$$OM1 = ZM1 ; VM1 = 0, RM1 = 0, NM1 = 0 \quad (17)$$

leži mnogokotnik M1 strogo zunaj mnogokotnika M2 (slika 25).

Ne glede na to, kateri mnogokotnik označuje presečni mnogokotnik in kateri luknjo ploskve, leži presečni mnogokotnik strogo zunaj luknje ploskve, kar pomeni, da lahko ostale luknje ploskve vplivajo na rešitev. Če ploskev nima več lukenj oziroma, če smo obdelali vse luknje ploskve, ploskev seka oktant, drugače pa ponovimo test lukenj za naslednjo luknjo ploskve.



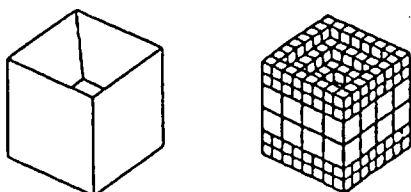
a) Presečni mnogokotnik M1 luknja ploskve M2 b) Luknja ploskve M1 presečni mnogokotnik M2

Slika 25 Mnogokotnik M1 strogo zunaj mnogokotnika M2

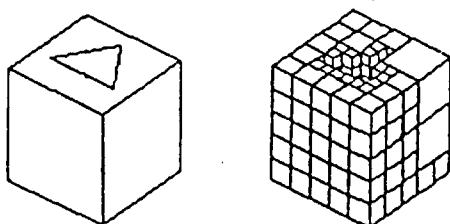
4 Zaključek

Testiranje presečišč med elementi telesa in osmiškega drevesa je časovno in prostorsko zelo zahtevno. Ta zahtevnost je odvisna predvsem od oblike telesa oziroma od števila vozlišč v osmiškem drevesu in od morebitne konkavnosti in luknjavosti telesa. Naše nadaljnje raziskovalno delo bo temeljilo na izboljšanju, oziroma optimiranju algoritma, kot tudi na realizaciji algoritma za pretvorbo iz predstavitve z osmiškim drevesom v predstavitev z ovojnico.

Program je napisan v pascalu in vključen v Eksperimentalni geometrijski modelirnik teles (EGM) [ŽAL189b]. Razvili smo ga na računalniku VAX-8800, za vizualizacijo pa smo uporabili grafični terminal TEK 4207. Rezultati programa so predstavljeni na slikah 26 in 27, kjer smo telesa prikazali z žičnim modelom. Za prikaz telesa in osmiškega drevesa z odstranjenimi zakritimi robovi smo uporabili razširjen Robertsov algoritem za odstranjevanje zakritih robov in ploskev ([ŠOB088], [ŠOB089]).



Slika 27 Primer konkavnega telesa



Slika 26 Primer telesa z luknjo

5 Literatura

- [CHIY85] Chiyokura, H., F. Kimura, "A Method of Representing the Solid Design Process", IEEE CG&A, Vol. 5, No. 4, 1985, pp. 32 - 41.
- [CHIY87] Chiyokura, H., "An Extended Rounding Operation for Modeling Solids with Free-Form Surfaces", IEEE CG&A, Vol. 7, No. 12, 1987, pp. 27 - 36.
- [CLIF87] Clifford, A. S. and H. Samet, "Optimal Quadtree Construction Algorithms", Computer Vision, Graphics, and Image Processing, Vol. 37, No. 3, March 1987, pp. 402 - 419.
- [GUID88] Guid, N., "Računalniška grafika" Tehniška fakulteta Maribor, Maribor 1988.
- [HILL82] Hillyard, R., "The Build Group of Solid Modelers", IEEE CG&A, Vol. 2, No. 2, 1982, pp. 43 - 52.
- [HOME88] Homer, H. C. and T. S. Huang, "A Survey of Construction and Manipulation of Octree", Computer Vision, Graphics, and Image Processing, Vol. 43, No. 3, September 1988, pp. 409 - 431.
- [KAUF88] Kaufman, A. and R. Bakalash, "Memory and Processing Architecture for 3D Voxel - Based Imagery", IEEE CG&A, Vol. 8, No. 6, 1988, pp. 10 - 23.
- [MÄNT82] Mäntylä, M. and R. Sulonen, "GWB: A Solid Modeler with Euler Operators", IEEE CG&A Vol. 2, No. 7, 1982, pp. 17 - 31.
- [MÄNT83] Mäntylä, M., "SET operations of GWB", Graphics Forum, No. 2 - 3, 1983, pp. 122 - 134.
- [MÄNT88] Mäntylä, M., "An Introduction to Solid Modeling", Computer Science Press, Rockville, MD (1988).
- [MILL89] Miller, R. J., "Architectural Issues in Solid Modelers", IEEE CG&A, Vol. 9, No. 5, 1989, pp. 72 - 87.
- [MORT85] Mortenson, M. E., "Geometric Modeling", John Wiley & Sons, New York, 1985.

- [PRAT83] Pratt, M. J., "Interactive geometric modelling for CAD/CAM", Eurographics '83 Tutorials, Zagreb, 1983.
- [REQU82] Requicha, A. A. G., "Solid Modeling: A Historical Summary and Contemporary Assessment", IEEE CG&A, Vol. 2, No. 2, 1982, pp. 9 - 24.
- [REQU83] Requicha, A. A. G. and H. B. Völcker, "Solid Modeling: Current Status and Research Directions", IEEE CG&A, Vol. 3, No. 5, 1983, pp. 25 - 37.
- [ŠOB088] Šobot, P., B. Žalik, N. Guid, "Extended Robert's hidden line / hidden surface algorithm for computer graphics", X. International Symposium Computer at the University, Cavtat 1988, pp. 7.5/1-4.
- [ŠOB089] Šobot, P., B. Žalik, N. Guid, "Analiza in uporaba razširjenega Robertsovega algoritma, ki rešuje problem zakritih robov pri mnogokotniških pravilnih telesih", XIII. Simpozijum o informacionim tehnologijama, Sarajevo 1989, pp. 227/1-10.
- [WEIL85] Weiler, K., "Edge-Based Data Structures for Solid Modeling in Curved-Surface Environmenst", IEEE CG&A, Vol. 5, No. 1, 1985, pp. 21 - 40.
- [WILS85] Wilson, P. R., "Euler Formulas and Geometric Modeling", IEEE CG&A, Vol. 5, No. 8, August 1985, pp. 24 - 36.
- [ŽALI89a] Žalik, B., N. Guid, and P. Šobot, "Experimental geometric modeler based on boundary representation", XI. International Symposium Computer at the University, Cavtat 1989, pp. 7.6/1-6.
- [ŽALI89b] Žalik, B., "Geometrijski modelirnik z Eulerjevimi operatorji", Magistrsko delo, Tehniška fakulteta, VTO ERI, Maribor, 1990.
- [ŽALI90] Žalik, B., N. Guid, "Vgradnja Eulerjevih operatorjev in njihova uporaba pri tvorbi teles s translacijskim pomikanjem", Informatica, Ljubljana, Vol. 14, No. 2, 1990, pp. 32 - 38.

* * * * *

Knjiga

Antona P. Železnikarja

On the Way to Information

(Napotik informaciji)

(recenziji na strani 79 in 80)
odpira izvirne poglede in možnosti razvoja

teorije informacije,
umetne inteligence
in nove poglede na področje informacije.

Profesor Branko Souček
in
doc. dr. Valter Motaln

podajata svoja stališča in ocene o knigi.

Knjigo lahko naročite pri
Slovenskem društvu Informatika,
po telefonu (061) 214 455 ali na naslov Tržaška c. 2, Ljubljana,
s plačilom din 270.

* * * * *