

1. Praćenje putanje

Za putanju po kojoj ćemo pomicati naš objekt odabrat ćemo **aproksimacijsku uniformnu B-splajn kubnu krivulju** zbog njenog jednostavnog oblika, različitih poželjnih svojstava, te jednostavnog načina određivanja derivacije. Derivacija određuje tangentu na krivulju, a koristit ćemo ju za određivanje orijentacije objekta.

1.1 Aproximacijska uniformna B-splajn kubna krivulja

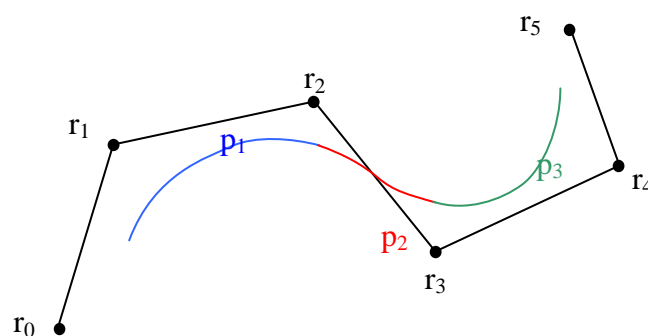
Aproximacijska B-splajn krivulja definirana je:

$$\vec{p}(u) = \sum_{i=0}^n \vec{r}_i N_{i,3}(u) \quad (1.1)$$

Za aproksimacijsku B-splajn kubnu krivulju možemo imati uniformno zadan vektor uzlova $u_{i+1} - u_i = \text{konstanta}$. To znači da su uzlovi jednoliko raspoređeni. Ako je razmak jediničan vrijedi $u_{i+1} - u_i = 1$. U tom slučaju krivulju možemo načiniti upotrebom samo periodičkog segmenta. Periodički segment aproksimacijske kubne B-splajn krivulje određen je sa četiri točke, a u tom slučaju i -ti segment krivulje zadan je:

$$\vec{p}_i(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \frac{1}{6} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix} \begin{bmatrix} \vec{r}_{i-1} \\ \vec{r}_i \\ \vec{r}_{i+1} \\ \vec{r}_{i+2} \end{bmatrix} = \mathbf{T}_3 \mathbf{B}_{i,3} \mathbf{R}_i, \quad (1.2)$$

gdje se za svaki segment parametar t mijenja $0 \leq t < 1$. **Sa četiri točke određen je jedan segment krivulje**, a općenito s n točaka određeno je $n-3$ segmenata krivulje (Slika 1).



Slika 1. Početni segment p_1 (plavi) krivulje određen je točkama r_0, r_1, r_2, r_3 , slijedeći segment p_2 (crveni) određen je točkama r_1, r_2, r_3, r_4 , a završi segment p_3 (zeleni) određen je točkama r_2, r_3, r_4, r_5 .

1.2 Tangenta na aproksimacijsku uniformnu B-splajn kubnu krivulju

Tangenta na krivulju zadanu jednažbom (1.2) određena je parametarskom derivacijom po parametru t . Tangenta $\vec{p}'_i(t)$ u točki krivulje određenoj parametrom t određena je istim kontrolnim točkama i matičnim oblikom:

$$\frac{\partial \vec{p}_i(t)}{\partial t} = \vec{p}'_i(t) = [3t^2 \quad 2t \quad 1 \quad 0] \mathbf{B}_{i,3} \mathbf{R}_i \quad (1.3)$$

odnosno, jednažbu (1.3) možemo zapisati:

$$\vec{p}'_i(t) = [t^2 \quad t \quad 1] \frac{1}{2} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 2 & -4 & 2 & 0 \\ -1 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \vec{r}_{i-1} \\ \vec{r}_i \\ \vec{r}_{i+1} \\ \vec{r}_{i+2} \end{bmatrix} = \mathbf{T}_2 \mathbf{B}'_{i,3} \mathbf{R}_i \quad (1.4)$$

Treba primijetiti da izraz (1.2) određuje točku krivulje $\vec{p}_i(t)$, a izraz (1.3) odnosno (1.4) smjer tangente $\vec{p}'_i(t)$, odnosno **vektor** u toj točki.

1.3 Orijentacija objekta

Položaj objekta u prostoru određen je pozicijom objekta u radnom prostoru koju određuju tri prostorne koordinate $\vec{p}_i(t) = [x(t) \quad y(t) \quad z(t)]$, te orijentacijom objekta koja je određena vektorom orijentacije. Objekt je potrebno rotirati oko koordinatnih osi kao bi postigao položaj zadane orijentacije, no nije svejedno kojim redoslijedom i oko kojih osi (izvornih ili rotiranih) ćemo obavljati rotaciju. Orijentacija se može zapisati i preko kutova obzirom na koordinatni sustav scene ili koordinatni sustav objekta, te definiranog redoslijeda koordinatnih osi oko kojih se obavlja rotacija. Ovi kutovi zovu se Eulerovi kutovi. U koordinatnom sustavu objekta promjene kutova (Slika 2.) određuju se po nagibu/dubini/smjeru (engl. roll/pitch/yaw).



Slika 2. Nagib dubina i smjer u koordinatnom sustavu objekta.

Promjena orijentacije zahtijeva interpolaciju u nizu zadanih orijentacijskih položaja. U postupku promjene orijentacijskog položaja kako u mehaničkim tako i u numeričkim postupcima može doći do problema kod podudaranja koordinatnih osi. Taj problem se naziva zastoj žiroskopskih osi (engl. gimbal lock). Problem se rješava tako da se dodaje još jedan stupanj slobode. Mehanički se to izvodi dodavanjem još

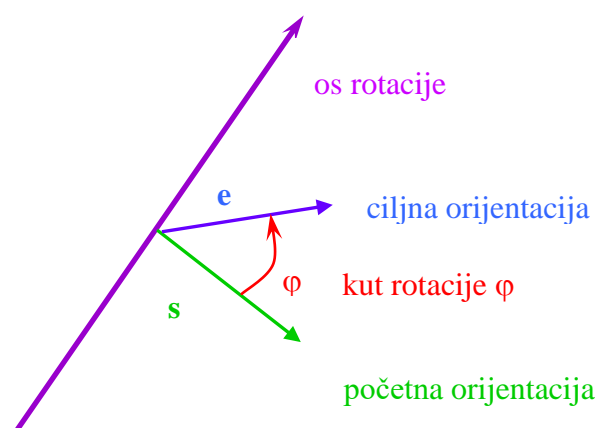
jednog prstena na žiroskop. U OpenGL-u rješenje je ostvareno uvođenjem još jedne varijable. Upotrebom kvaterniona za zapis i interpolaciju orijentacije objekta postiže se kvalitetno rješenje.

1.4 Rotacija objekta u OpenGL-u

Rotacija objekta u OpenGL zadaje se upotrebom četiri parametra. To su os rotacije (tri parametra) i kut rotacije (Slika 3).

Početna i ciljna orijentacija objekta svojim vektorskim produktom određuju os rotacije. Početna orijentacija objekta neka je zadana vektorom \mathbf{s} , a orijentacija koju želimo postići vektorom \mathbf{e} . Početnu orijentaciju možemo proizvoljno odabrati. Ako je nos aviona orijentiran u smjeru z-osi odabrat ćemo za vektor $\mathbf{s}=(0\ 0\ 1)$. Vektor \mathbf{os} određuje os oko koje je potrebno rotirati objekt.

$$\mathbf{os} = \mathbf{s} \times \mathbf{e} = \begin{vmatrix} \vec{i} & \vec{j} & \vec{k} \\ s_x & s_y & s_z \\ e_x & e_y & e_z \end{vmatrix} = \begin{bmatrix} \vec{i} & \vec{j} & \vec{k} \end{bmatrix} \cdot \begin{bmatrix} s_y e_z - e_y s_z \\ -(s_x e_z - e_x s_z) \\ s_x e_y - s_y e_x \end{bmatrix} \quad (1.5)$$

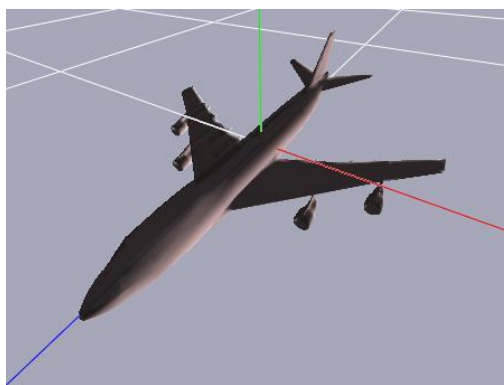


Slika 3. Os rotacije i kut rotacije.

Kosinus kuta rotacije određen je skalarnim produktom vektora početne i ciljne orijentacije objekta podijeljenim s normama tih vektora:

$$\cos \varphi = \frac{\mathbf{s} \cdot \mathbf{e}}{|\mathbf{s}| |\mathbf{e}|} \quad (1.6)$$

U naredbi OpenGL-a `glRotatef(φ, osx, osy, osz)` prvi parametar određuje kut rotacije izražen u stupnjevima gledano iz pozitivnog smjera osi suprotno od kazaljke na satu, a slijedeća tri parametra određuju pojedine komponente vektora osi rotacije. Obratiti pažnju da funkcija `acos()` obično vraća kut u radijanima a `glRotate()` prihvaća kut u stupnjevima. Početna orijentacija \mathbf{s} objekta može se odabrati proizvoljno, a odredimo ju tako da početna orijentacija objekta bude u smjeru u kojem želimo da se objekt giba (Slika 4).



Slika 4. Za početni položaj objekta u koordinatnom sustavu x-crvena, y-zelena, z-plava, početnu orijentaciju s objekta odabrat ćemo tako da se podudara sa z osi.

Ako početna orijentacija (pozicija) objekta nije onakva kakvu bi željeli, potrebno je inicijalno rotirati objekt (trenslatirati) kako bi se središte objekta i početna orijentacija podudarila s putanjom.

1.5 Akumulacija pogreške

Prilikom transformacije objekta mijenjaju se izračunate koordinate objekta. Računanje novih koordinata na osnovi starih nije poželjno zbog nakupljanja numeričke pogreške. Izvorne koordinate objekta u lokalnom koordinatnom sustavu objekta se čuvaju, a novi položaj objekta određuje se iz izvornih koordinata objekta i zadanog ciljnog položaja koji objekt treba zauzeti. To znači da ćemo za svaki izračunati položaj objekta i orijentaciju objekta množiti izvorne (početne) koordinate objekta matricom rotacije i translacije uz odgovarajući redoslijed, a zatim ćemo iscrtati objekt.

1.6 Reprezentacije orijentacije DCM matricom

Orijentaciju objekta možemo postići ako znamo poziciju i koordinatne osi ciljnog koordinatnog sustava. U našem primjeru pomak na zadano mjesto krivulje određuje poziciju. Koncentrirat ćemo se sada samo na orijentaciju, tako da ćemo promatrati poziciju fiksiranu uz ishodište koordinatnog sustava $O(0, 0, 0)$.

Od koordinatnog sustava koji određuje orijentaciju našeg objekta znamo jednu os koja je određena tangentom (1.4). Potrebno je odrediti i druge dvije koordinatne osi, odnosno normalu i binormalu kao bi imali potpuno određen koordinatni sustav objekta. Normala je određena vektorskim umnožkom prve i druge derivacije, odnosno vektorom tangente kojeg imamo (1.4) i drugom derivacijom koji dobijemo ako (1.4) još jednom deriviramo po parametru t :

$$\vec{u} = \vec{p}'_i(t) \times \vec{p}''_i(t) \quad ? \quad (1.7)$$

a binormala je određena vektorom tangente $\vec{w} = \vec{p}'_i(t)$ i dobivenim vektorom normale \vec{u} , te tako ovi vektori formiraju desni koordinatni sustav:

$$\vec{v} = \vec{w} \times \vec{u} \quad ? \quad (1.8)$$

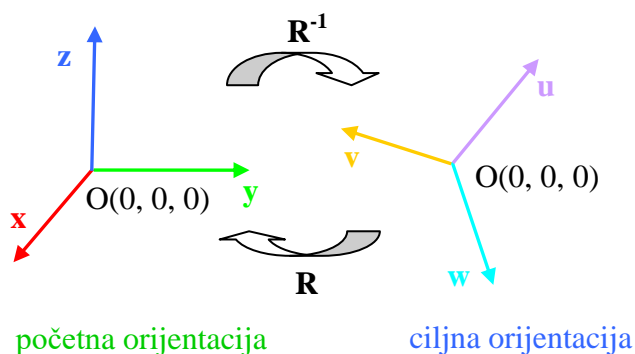
Treba pripaziti na drugu derivaciju koja može biti nula u slučaju kada je segment krivulje ravan (zadane četiri točke su na pravcu). U tom slučaju obično se odabire neka orijentacija.

Određivanjem vektora tangenta, normale i binormale dobili smo desni koordinatni sustav određen koordinatnim osima \vec{w} , \vec{u} , \vec{v} . Ako je zadan objekt u koordinatno sustavu \vec{w} , \vec{u} , \vec{v} a želimo ga transformirati u \vec{x} , \vec{y} , \vec{z} tako da se navedenim redom podudare koordinatne osi, potrebno je pomnožiti sve točke zadane u sustavu \vec{w} , \vec{u} , \vec{v} rotacijskom matricom:

$$\mathbf{R} = [\vec{w} \quad \vec{u} \quad \vec{v}] = \begin{bmatrix} w_x & u_x & v_x \\ w_y & u_y & v_y \\ w_z & u_z & v_z \end{bmatrix} \quad (1.9)$$

kako bi dobili koordinate u sustavu \vec{x} , \vec{y} , \vec{z} (Slika 5). Ova matrica se zove rotacijska ili DCM matrica (*direction cosine matrix*). Neka svojstva ove matrice su da ima realne parametre, ortogonalna je, determinata joj je 1, svojstveni vektor čija je pripadna svojstvena vrijednost 1 određuje os rotacije.

Nama je potrebna obrnuta transformacija, odnosno objekt je zadan u \vec{x} , \vec{y} , \vec{z} a potrebno je transformirati objekt u \vec{w} , \vec{u} , \vec{v} , odnosno potrebno je pomnožiti točke objekta iz lokalnog koordinatnog sustava objekta \vec{x} , \vec{y} , \vec{z} inverznom matricom \mathbf{R} (1.9) kako bi dobili koordinate u sustavu \vec{w} , \vec{u} , \vec{v} .



Slika 5. Promjena orijentacije x, y, z u w, u, v .

Ovaj način određivanja orijentacije treba povezati još s pomakom, odnosno **prvo odredimo orijentaciju, a zatim pomaknemo objekt na traženu poziciju u prostoru**. Ovisno o inicijalnoj orijentaciji objekta u ovoj vježbi, tj. je li trup aviona (Slika 4) u smjeru x, y, z osi bit će potrebno promijeniti redoslijed podudaranja osi ili inicijalno rotirati objekt tako da se tangenta krivulje podudara sa željenom orijentacijom pri kretanju po krivulji.

1.7 Radni zadatak

Zadan je jednostavan primjer u kojem se koristi samo dio zapisa objekta Wavefront (.obj). Zapis sadrži popis vrhova i njihovih koordinata te popis poligona s pripadnim indeksima vrhova.

1. Iz datoteke učitati zadano tijelo (dio laboratorijske vježbe 4. iz Interaktivne računalne grafike). Dozvoljeno je korištenje biblioteka za učitavanje objekata, npr. [Assimp](#).
2. Iz datoteke učitati niz točaka koje određuju aproksimacijsku uniformnu kubnu B-splajn krivulju.
3. Za svaki segment krivulje mijenjati parametar t od 0 do 1:
 - 3.1. Prema formuli (1.3) ili (1.4) odrediti ciljnu orijentaciju objekta. Na osnovi početne orijentacije i ciljne orijentacije odrediti os rotacije i kut rotacije po formulama (1.5) i (1.6). Prema formuli (1.2) odrediti potrebnu translaciju objekta.
 - 3.2. Odrediti transformirane koordinate objekta.
 - 3.3. Iscrtati krivulju (putanju) i pripadnu tangentu. Tangente nacrtati tako da se od pojedine točke krivulje nacrtaju kratka dužina u smjeru tangente (ciljne orijentacije) za parametar t . Znači početna točka dužine je točka krivulje, a završna se dobije tako da se na početnu točku zbroji vektor tangente skaliran s proizvoljnim faktorom.
 - 3.4. Iscrtati objekt.
 - 3.5. Usporediti načine određivanja orijentacije opisane u poglavlju 1.4 i 1.6.
4. Definirati putanju u obliku spirale npr.

$$V_1=(0\ 0\ 0)\ V_2=(0\ 10\ 5)\ V_3=(10\ 10\ 10)\ V_4=(10\ 0\ 15)$$

$$V_5=(0\ 0\ 20)\ V_6=(0\ 10\ 25)\ V_7=(10\ 10\ 30)\ V_8=(10\ 0\ 35)$$

$$V_9=(0\ 0\ 40)\ V_{10}=(0\ 10\ 45)\ V_{11}=(10\ 10\ 50)\ V_{12}=(10\ 0\ 55)$$
 animirati objekt.

