

DU 10

April 28, 2023

Vizualizujte pomocou OpenGL rzptyl biliardovych gul vzajomne medzi sebou a ich odrazy od stien biliardoveho stola.

Parametre programu:

- rozmer stola: $L_x = 2,5$; $L_y = 1,5$
- $R = 0,075$
- $dt_0 = 0.025$ je pociatocna dlzka kroku v sekundach.

Stred stola je v bode $(0,0)$. Program by mal fungovat pre lubovolny pocet gul (pokial sa zmestia na stol), ale pouzite ich iba 3. Pociatocne suradnice gul a rychlosti vygenerujte. Dbajte na to, aby sa gule na zaciatku neprekryvali a jednotlivé zložky rychlosti každej gule dajte v rozsahu $(-0.5, +0.5)$. Najskor ku rozptylu. Vstupné veličiny pri zrazke sú vektory rychlosti a polohové vektory gul pri zrazke (označené bez ciariiek). Potom zložky rychlosti po zrazke (toto je prípad pre gule 1 a 2) budú:

$$v'_{1x} = v_{1x} + \frac{r_{12x}(r_{12x}v_{12x} + r_{12y}v_{12y})}{4R^2}; \quad (1)$$

$$v'_{1y} = v_{1y} + \frac{r_{12y}(r_{12x}v_{12x} + r_{12y}v_{12y})}{4R^2}; \quad (2)$$

$$v'_{2x} = v_{2x} - \frac{r_{12x}(r_{12x}v_{12x} + r_{12y}v_{12y})}{4R^2}; \quad (3)$$

$$v'_{2y} = v_{2y} - \frac{r_{12y}(r_{12x}v_{12x} + r_{12y}v_{12y})}{4R^2}; \quad (4)$$

$$\vec{r}_{12} = \vec{r}_2 - \vec{r}_1 \Rightarrow r_{12x} = r_{2x} - r_{1x} \quad (5)$$

$$r_{12y} = r_{2y} - r_{1y} \quad (6)$$

$$\vec{v}_{12} = \vec{v}_2 - \vec{v}_1 \Rightarrow v_{12x} = v_{2x} - v_{1x} \quad (7)$$

$$v_{12y} = v_{2y} - v_{1y} \quad (8)$$

kde r_{1x} a r_{1y} su x -ova a y -ova suradnica prvej gule, podobne pre druhu gulu r_{2x} a r_{2y} .

Co sa tyka odrazu od stien, ten je trivialny, iba v pripade dotyku so stenou otocite smer zlozky rychlosti kolmej na stenu. Napriklad pre x -ovu zlozku, ked ide teleso doprava:

$$if(x + R) \geq (0.5 * Lx) \Rightarrow v_x * = -1 \quad (9)$$

samozrejme to treba osetrit podla smeru pohybu.

Teraz k vypoctu najblizsej kolizie (to je kvoli stabilite simulacie, mozete to osetrit aj inym sposobom, ked viete ako). Treba najst najmensi cas z moznosti: kolizia jednotlivych gul so stenami t_i (teda najblizsi cas narazu i -tej gule na stenu), cas vzajomnej zrazky i -tej a j -tej gule t_{ij} . Pre vypocet t_i (tuto je len cas z x -ovej zlozky pohybu, ak ide doprava, treba spravit test na vsetky smery a vybrat najmensi cas):

$$t_i = \frac{0.5 * Lx - (x + R)}{v_x} \quad (10)$$

Podmienky pre vypocet t_{ij} (tieto musia byt splnene, aby sa gule realne mohli zrazit):

$$\vec{r}_{ij} \cdot \vec{v}_{ij} < 0 \quad (11)$$

$$(\vec{r}_{ij} \cdot \vec{v}_{ij})^2 - v_{ij}^2 (r_{ij}^2 - 4R^2) > 0 \quad (12)$$

Ak su obe splnene, potom samotny cas:

$$t_{ij} = \frac{1}{v_{ij}^2} \left(-\vec{r}_{ij} \cdot \vec{v}_{ij} - \sqrt{(\vec{r}_{ij} \cdot \vec{v}_{ij})^2 - v_{ij}^2 (r_{ij}^2 - 4R^2)} \right) \quad (13)$$

Potom zoberieme za cas najblizsej zrazky:

$$t_z = \min\{t_i, t_{ij}\} \quad (14)$$

pre vsetky kombinacie i a j . Z tejto hodnoty vypocitame novu dlzku kroku:

$$n = 1 + \text{int} \left(\frac{t_z}{dt_0} \right) \quad (15)$$

$$dt_{new} = \frac{t_z}{n} \quad (16)$$

kde dt_0 je dlzka kroku, ktoru sme si nastavili defaultne na zaciatku (tu je to tych 0.025 s) a n je pocet krokov do najblizsej zrazky. Pre kontrolu novy cas kroku dt_{new} vypisujte do konzoly.

Po zistení t_z a nastavenia dt_{new} nechajte simuláciu bezat (tych vypočítaných n krokov), až kým sa aktuálny čas nebude rovnať času zrazky $t = t_z$, potom spočítajte danú zrazku (nové rýchlosti) a znova určite čas do ďalšej zrazky a tak dookola. Keďže sú zrazky dokonale pružné a neuvažujeme trenie, tak by sa mal systém hybať donekonečna, nemala by sa strácať kinetická energia. Fotky tabule s odvodeniami všetkých vzťahov sú priložené.