

Specyfikacja funkcjonalna symulatora układu N-ciał: **nsim**

Mateusz Czarnecki, Paweł Jędrzejczyk

27.02.2019

Cel projektu

Program **nsim** ma na celu numeryczne wyznaczanie trajektorii obiektu, na który wpływa N-1 innych ciał. Program działa w trybie nieinteraktywnym/wsadowym. Wygenerowane zbiory danych pozwalają na ich wizualizację w programie **gnuplot**. Program umożliwia analizę wielu ciał poprzez umożliwienie pominięcia nieistotnych oddziaływań. W **nsim** zaimplementowano różne sposoby obsługi kolizji. Użytkownik może wybrać też metodę całkowania numerycznego.

Dane wejściowe

Do swojego działania program potrzebuje danych o śledzonych ciałach. Każde z ciał musi być opisane następującymi parametrami:

- etykieta/nazwa: nie jest używana w symulacji, jednak jest zapisywana do pliku wyjściowego w celach wizualizacyjnych; etykieta może zawierać znaki **a-zA-Z0-9** - oraz nie może być dłuższa niż 32 znaki;
- masa ciała (M): podana w kg, dopuszczalne wartości $M > 0$;
- położenie w przestrzeni 3D (x, y, z): niezbędne jest zdefiniowanie położenia x i y . W przypadku niezdefiniowanego położenia z , program założy $z=0$; położenie podane w metrach, względem pewnego początku układu odniesienia;
- bieżąca prędkość w przestrzeni 3D (v_x, v_y, v_z): analogicznie jak w przypadku położenia, niezdefiniowana prędkość v_z będzie zinterpretowana jako $v_z=0$; prędkości podane są w m/s;
- promień obiektu (R): podany w metrach, dopuszczalne tylko $R > 0$.

Parametry symulacji numerycznej są modyfikowane za pomocą argumentów wywołania programu.

Każde z ciał jest zdefiniowane w jednej linii, jego parametry oddzielone są od siebie znakiem `;`. Program wczytuje kolejne ciała aż do napotkania linii zawierającej tylko znak `..`

Przykład danych wejściowych:

```
object_a;10; 1.5 2.0;15.0 0.0 0.0;0.2
object_b; 4;-0.5 0.0;-15.0 0.0;0.2
..
```

W powyższym pliku zdefiniowane są dwa ciała `object_a` i `object_b` o masach (odpowiednio) 10 kg i 4 kg. Położenie obiektu `object_a` to [1,5m; 2m; 0m], `object_b` to [-0,5m; 0m; 0m]. Prędkości obiektów to:

- `object_a` <15m/s, 0m/s, 0m/s> ,
- `object_a` <-15m/s, 0m/s, 0m/s>.

Argumenty wywołania programu

Program `nsim` akceptuje następujące argumenty wywołania:

- `--timestep dt` precyzuje krok czasowy wykorzystywany podczas całkowania numerycznego; podany w sekundach; domyślnie `dt = 60`;
- `--iterations n` określa liczbę analizowanych kroków czasowych; domyślnie `n = 1000`, liczba iteracji nie może być mniejsza od 1;
- `--integrator euler|verlet` pozwala na wybór algorytmu całkowania numerycznego: `euler` określa metodę całkowania Eulera, `verlet` określa metodę całkowania Verleta; domyślna wartość to `euler`;
- `--collision none|remove|momentum-exchange|sticky` określa sposób obsługi kolizji między ciałami: `none` wyłącza kolizje, `remove` usuwa obydwie ciała, `momentum-exchange` odpowiada zderzeniu sprężystemu, `sticky` odpowiada zderzeniu niesprężystemu;
- `--relative-mass ratio` próg stosunku mas ciał poniżej którego pomija się wpływ ciał lżejszych; domyślna wartość `ratio=0.01`;
- `--radius-of-influence R` odległość powyżej której nie uwzględniamy wpływu ciała; domyślna wartość `R=1e8`;
- `--bodies filename` nazwa pliku, z którego wczytane będą parametry ciał (opisane w *Dane wejściowe*); domyślnie jest to wartość pusta, oznaczająca pobieranie danych ze standardowego wejścia;
- `--output filename` nazwa pliku, do którego zapisane zostaną dane;

Przykładowe wywołania programu:

- `./nsim --timestep 3600 --bodies solar.dat --output solar.out` , efektem będzie analiza pliku `solar.dat` z krokiem czasowym 1 godzina, dane wynikowe zostaną zapisane do pliku `solar.out`. Program nie będzie analizować kolizji. Zostanie zastosowana metoda całkowania Eulera.

Teoria

[tutaj opisałbym jak to jest liczone -- patrz tablica]

Komunikaty błędów

Program `nsim` stara się kontynuować pracę mimo napotkania nieprawidłowych danych w celu poprawienia diagnostyki danych wejściowych.

1. Masa ciała w danych wejściowych mniejsza/równa 0: Linia 31: Masa ciała musi być większa od 0 kg. Wczytano: "-2.1". Ciało zostało pominięte. Program wykrył ujemną masę (o wartości -2.1) w linii 31 danych wejściowych. Program ignoruje nieprawidłowe ciało.
2. Brak poprawnie wczytanych ciał: W podanych danych wejściowych brak poprawnie zdefiniowanych ciał. Przerywam działanie.

Komunikat pojawia się, gdy w podanych danych wejściowych program nie znajdzie poprawnie zdefiniowanych ciał. Może to wynikać z tego, że plik jest pusty, albo wszystkie wpisy zawierają błędy (wtedy pojawią się stosowne komunikaty jak w 1.).