

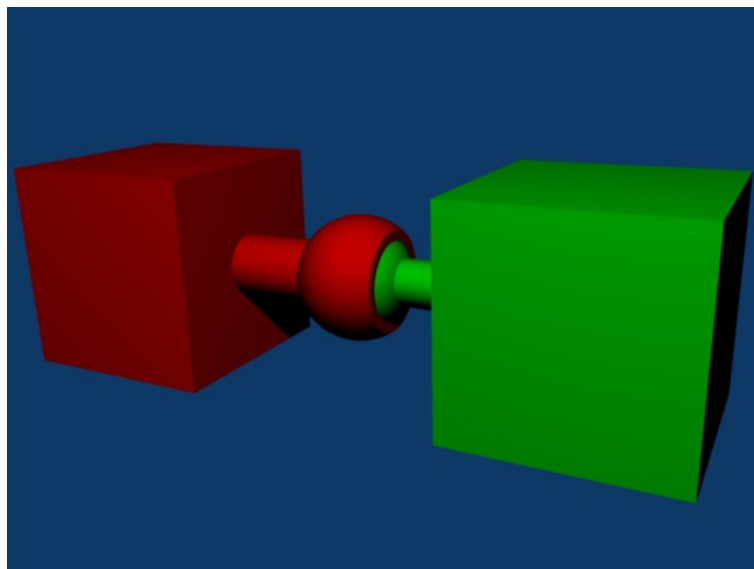
Wiązania w BGE (011; 31.07.2009; blender)

Artykuł ten ma na celu opis zastosowania wiązań dla elementów typu *Rigid Body* w silniku gry Blendera. Szczegółowy opis wiązań i metody ich ustalania uzupełniony został wieloma przykładami.

Wiązanie jest to połączenia dwóch elementów – ograniczenie ich wzajemnego ruchu. Patrząc w ujęciu mechaniki elementy te tworzą parę kinematyczną, a wiązanie to w istocie więzy geometryczne. Jednak wracając do Blendera, wiązania te dotyczyć będą elementów typu *Rigid Body* i określane są mianem *Rigid Body Joints*. Należą one do jednej z grupy wiązań (*constraints*) możliwych do zastosowania w Blenderze.

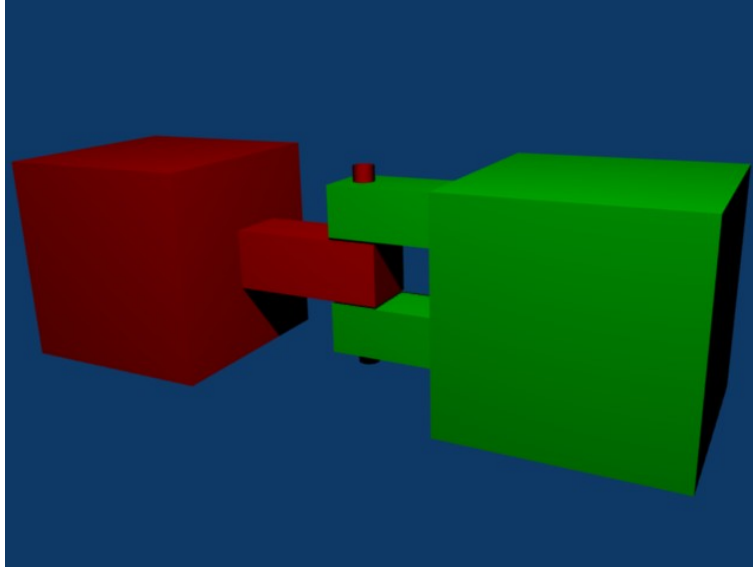
Przegub kulisty.

Wiązanie kuliste (*Ball*) łączy dwa elementy w jednym punkcie tak, że oba obiekty mogą obracać się tylko wokół tego punktu, a punkt ten jest punktem wspólnym dla obu obiektów i może się z nimi przemieszczać. Schematycznie można przedstawić to następująco:



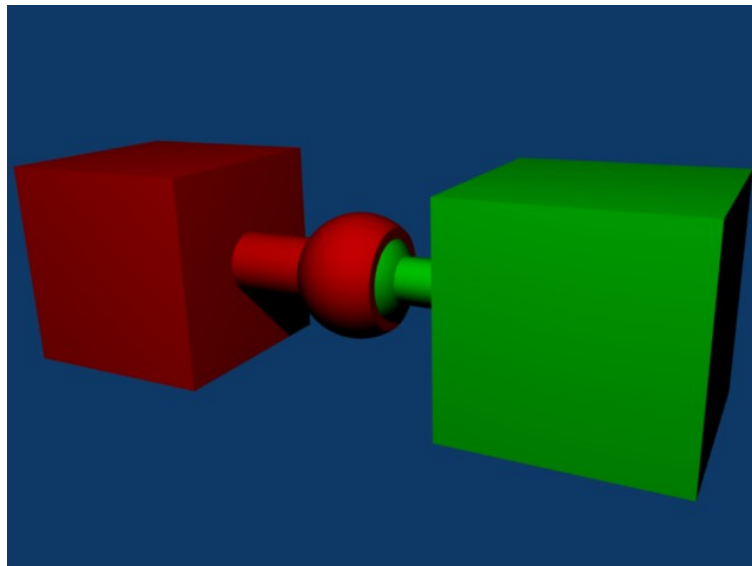
Zawias.

Wiązanie zawiasowe (przegubowe, *Hinge*) łączy dwa elementy w jednej osi tak, że oba mogą obracać się względem niej, oś ta jest wspólna dla obu obiektów i może się razem z nimi poruszać. Schematycznie można przedstawić to następująco:

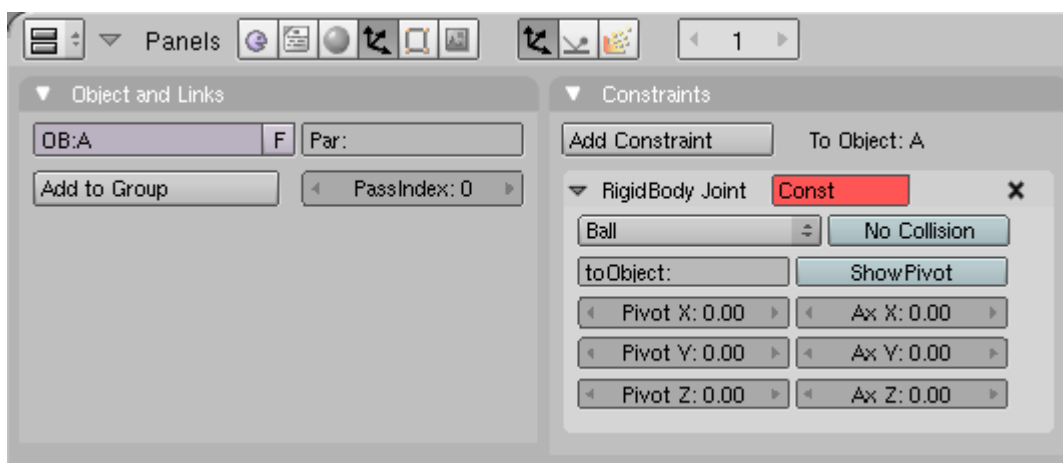


Tworzenie wiązania kulistego.

Do powyżej pokazanych par obiektów utworzymy odpowiednie wiązania. Plik z obiektami do pobrania tutaj: <http://myinventions.pl/011/Wiazania.blend>

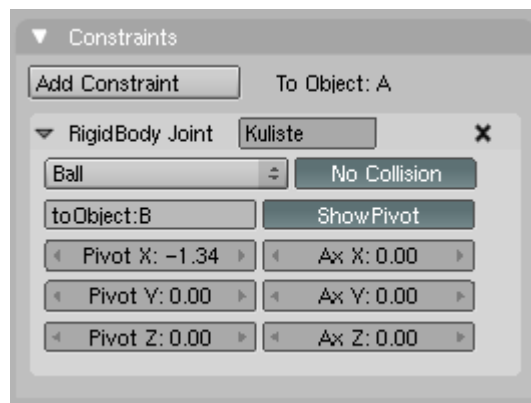


- Dla obu obiektów używamy kombinacji klawiszy *Ctrl + A* i wybieramy polecenie *Scale and Rotation to ObData* (jest to bardzo ważna operacja, bez niej m. in. BGE może mieć błędne dane o lokalizacji i orientacji obiektów).
- Zaznaczmy jeden z obiektów, np. obiekt *A* (zielony).
- Przejdźmy do panelu *Object* (F7).
- W zakładce *Constraints* klikamy na *Add Constraint* a z listy wybieramy *Rigid Body Joint*.

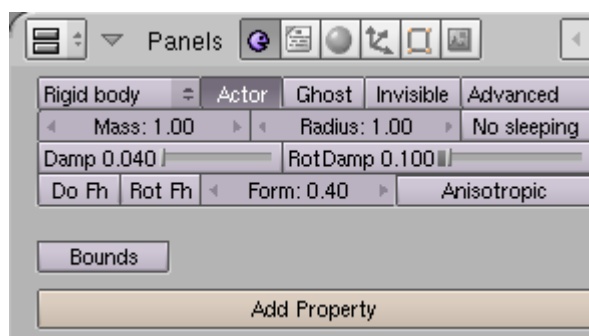


- Na czerwono podświetlona widnieje nazwa wiązania (kolor ten oznacza błąd niezdefiniowania). Zmieńmy nazwę na *Kuliste*.

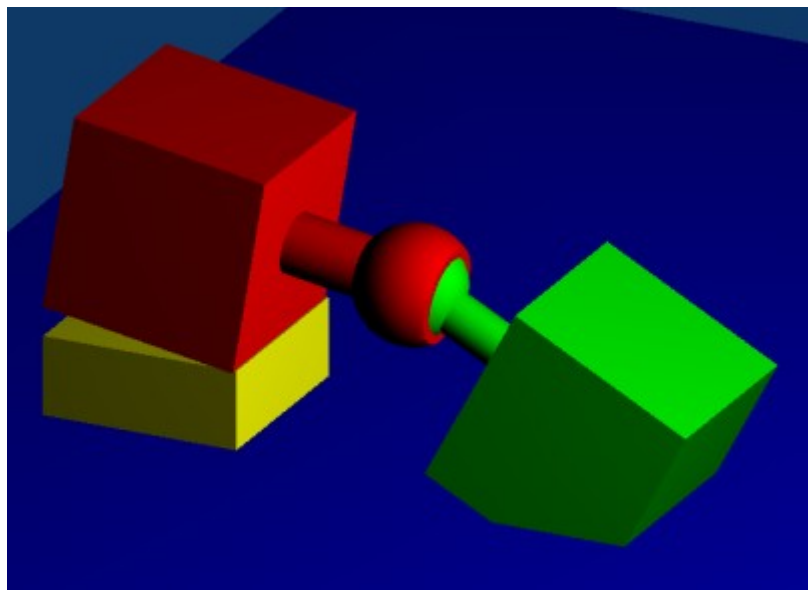
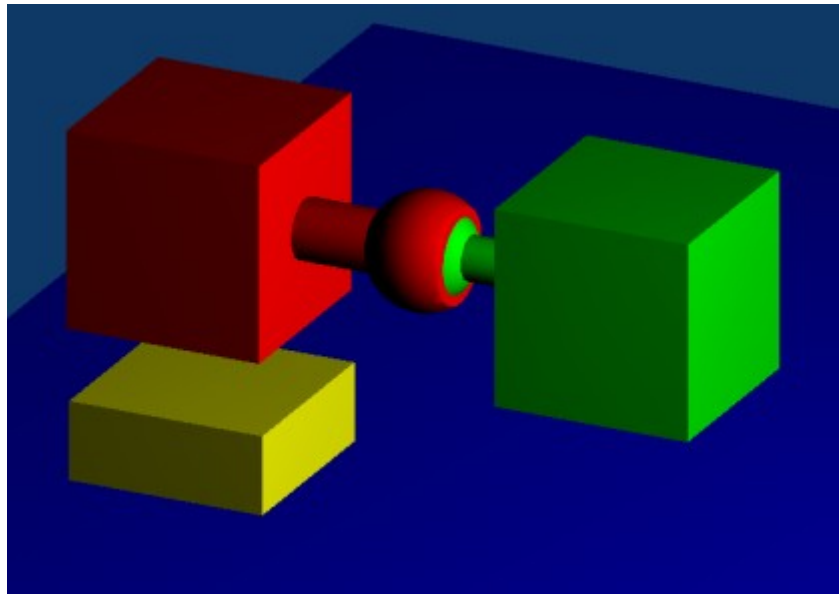
- Poniżej widnieje wybrany typ wiązania *Ball* z listy *Joint Types*.
- W polu z treścią *toObject*: wpisujemy nazwę obiektu do związania z obiektem *A*, czyli obiekt *B* (gaśnie czerwone podświetlenie nazwy wiązania).
- Wciskamy przycisk *No Collision* co powoduje, że obiekty te nie będą ulegać kolizji podczas symulacji i nie nastąpi rozstrzelenie wiązania z powodu przenikających się obszarów obiektów.
- Wciskamy przycisk *ShowPivot* co spowoduje pokazanie się dodatkowego układu współrzędnych pokrywającego się z układem współrzędnych obiektu (widać go po przejściu do wyświetlania siatkowego – *Draw Type/Wireframe* – klawisz *z*). Jeśli układy współrzędnych nie pokrywają się, oznacza to nie zastosowanie się do pierwszego punktu.
- Początek opisanego wyżej układu współrzędnych jest punktem wspólnym obu obiektów i będą mogły one wykonywać tylko obroty względem tego punktu. Musimy zatem przesunąć go w środek kuli symbolizującej wiązanie. Do przesuwania układu wzdłuż osi *px*, *py* i *pz* służą odpowiednio pola: *Pivot X*, *Pivot Y*, *Pivot Z*. Do obrotu układu wokół osi *px*, *py* i *pz* służą odpowiednio pola: *Ax X*, *Ax Y*, *Ax Z* (wartości obrotu podane w stopniach).
- Dla podanego przykładu odpowiednia jest wartość *Pivot X* równa *-1.335*. Obroty nie mają wpływu na wiązanie. Warto zauważyć, że środki lokalnych układów współrzędnych obu obiektów połączone są przerywaną linią – symbolizuje ona wiązanie.



- Przechodzimy do panelu Logic (F4) i ustalamy dla obiektów *A* i *B* *Object Type: Rigid Body*.



- Dodajemy do sceny podłoże oraz bryłę pod jednym z obiektów, aby wymusić kontakt i ukazać działanie wiązania.
- Uruchamiamy symulację (klawisz P). Dwie przykładowe klatki symulacji:



Wiązanie działa prawidłowo, pozostał jeszcze do rozważenia problem przenikania się obiektów (wywołałem go celowo, aby dokładniej opisać to zagadnienie).

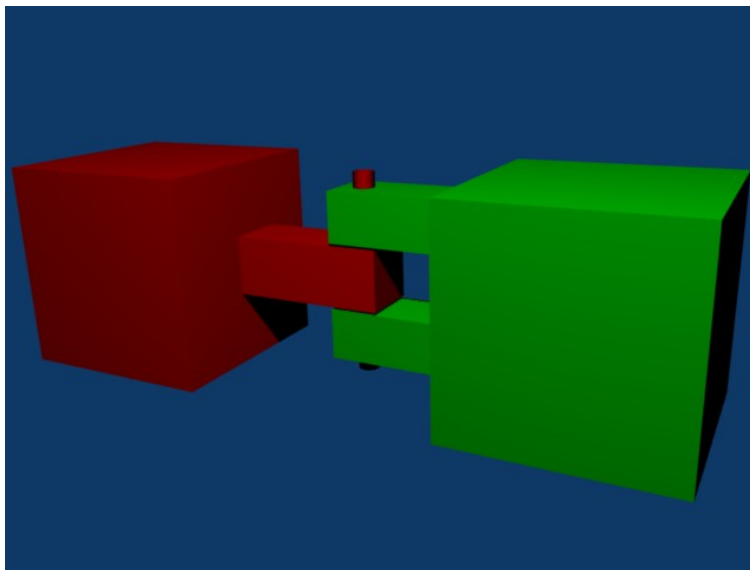
Przyjrzyjmy się dokładnie siatce jaka została utworzona dla obiektów na potrzeby symulacji (a raczej gry, bo to w końcu silnik gry blendera). Zaznaczamy w menu *Game* opcję *Show Physics Visualization* oraz przechodzimy do wyświetlania typu *Wireframe*. Po uruchomieniu symulacji zauważymy białe bryły o środkach w środkach lokalnych układów współrzędnych obu brył nijak nie oddające ich kształtu. Istotnie, nie ustaliliśmy dla obiektów żadnego rodzaju granic (*Bounds*), zostały zatem dobrane domyślne o średnicy ustalonej parametrem *Radius*.

Najprościej jest zazwyczaj wcisnąć przycisk *Bounds* i wybrać *Convex Hull*. Jednak dla tej opcji, wiele niepotrzebnych szczegółów obiektu jest dokładnie odwzorowywane, warto czasem zastanowić się nad zastosowaniem prostszego obszaru kolizji (np. dla kół samochodu użyć typu *Cylinder*).

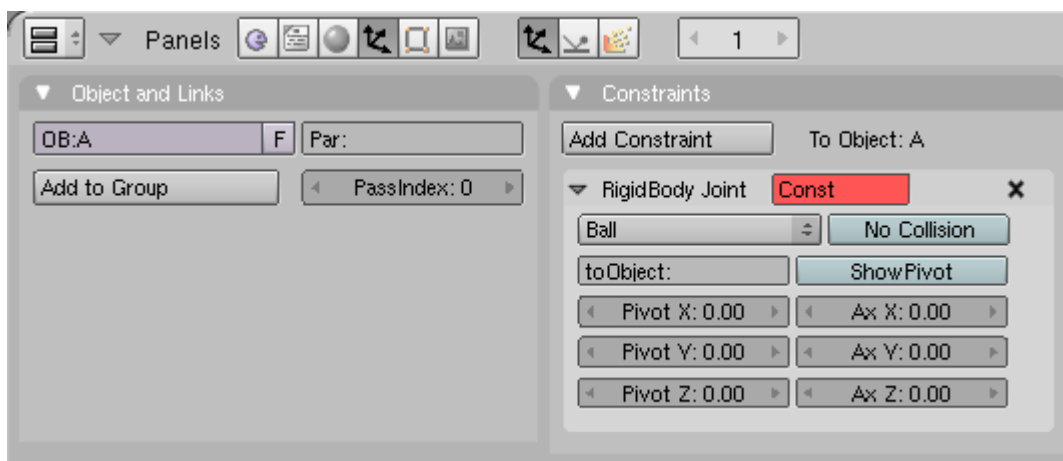
Przy okazji nadmienię, że w wizualizacji obiektów do symulacji rozpoznać można kilka kolorów linii. Linie białe tworzą siatkę granicy kolizji obiektów. Gdy obiekt przejdzie w stan uśpienia zmieniają one kolor na zielony (uśpienie następuje, gdy prędkość obiektu będzie dostatecznie mała i nie ma już potrzeby przeliczania jego fizyki; opcję tą można wyłączyć wciskając dla obiektu przycisk *No sleeping*).

Plik z gotowym wiązaniem: <http://myinventions.pl/011/WiazanieAB.blend>

Tworzenie wiązania zawiasowego.

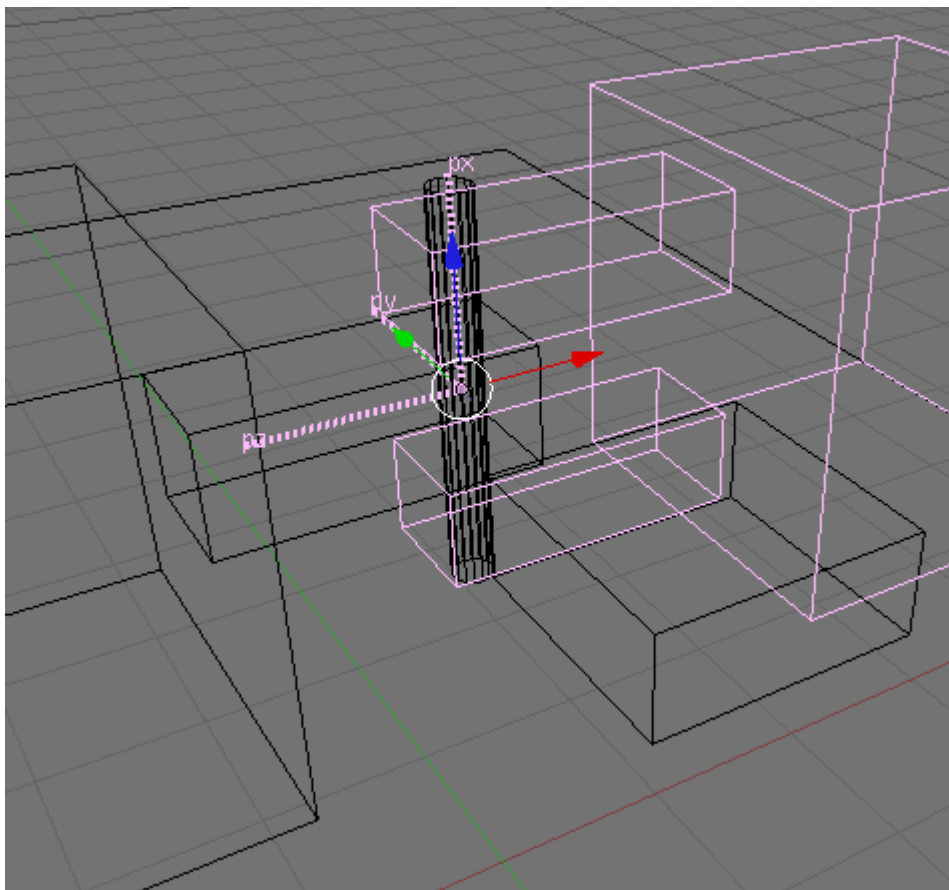
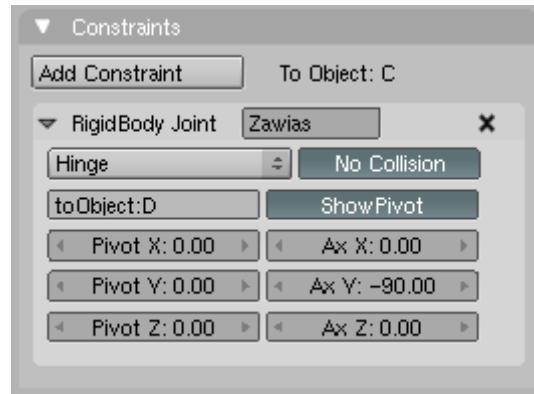


- Dla obu obiektów używamy kombinacji klawiszy *Ctrl + A* i wybieramy polecenie *Scale and Rotation to ObData*.
- Zaznaczymy jeden z obiektów, np. obiekt *C* (zielony).
- Przejdźmy do panelu *Object* (F7).
- W zakładce *Constraints* klikamy na *Add Constraint* a z listy wybieramy *Rigid Body Joint*.



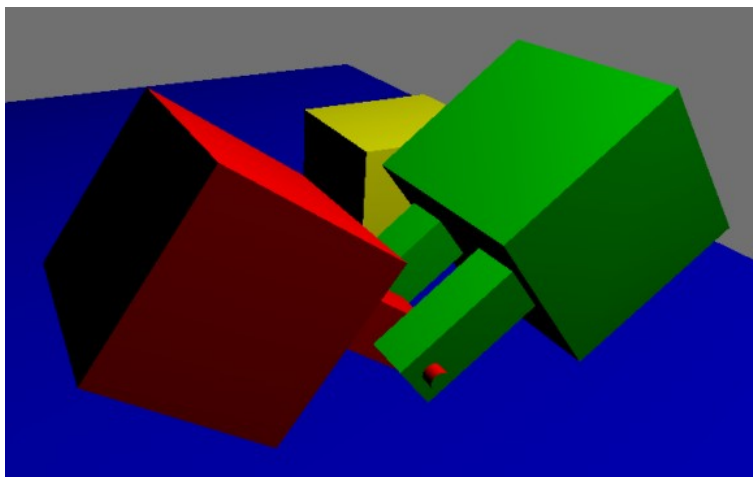
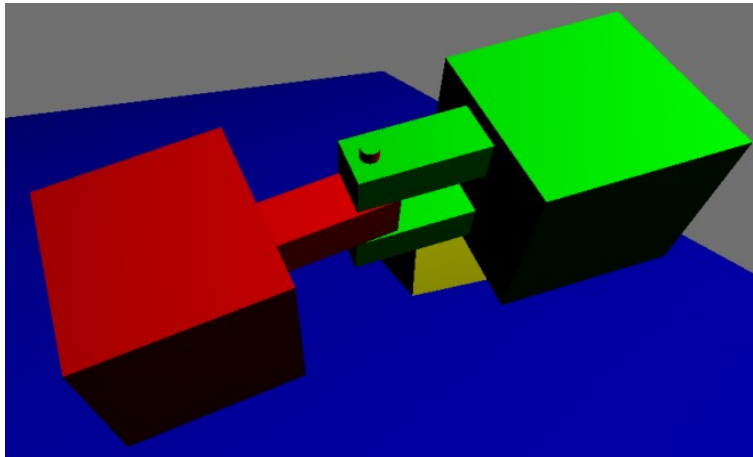
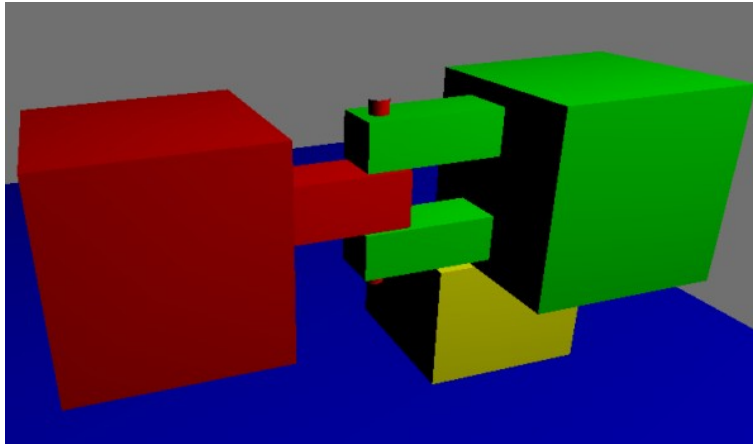
- Zmieńmy nazwę wiązania na *Zawias*.
- Wybieramy typ wiązania *Hinge* z listy *Joint Types*.
- W polu z treścią *toObject:* wpisujemy nazwę obiektu do połączenia, czyli obiekt *D*.
- Wciskamy przycisk *No Collision*.
- Wciskamy przycisk *ShowPivot* i przechodzimy do wyświetlania siatkowego *Wireframe*.

- Dla wiązania zawiasowego osią obrotu jest zawsze oś px . Musimy zatem tak ustawić położenie i orientację układu wiązania, aby oś px znalazła się w pożądanej osi obrotu. Zamiast przesuwać układ współrzędnych parametrami wiązania, możemy również przesunąć lokalny układ współrzędnych obiektu (stawiamy kursor w osi lewym kliknięciem myszy, w panelu *Editing* (F9) w zakładce *Mesh* klikamy przycisk *Center Cursor*). Pozostaje już tylko odpowiedni obrót układu o -90 stopni wokół osi py .



- Przechodzimy do panelu Logic (F4) i ustalamy dla obiektów *A* i *B* *Object Type: Rigid Body*. Wciskamy przycisk *Bounds* i wybieramy *Convex Hull*.

- Dodajemy do sceny podłoże oraz bryłę pod jednym z obiektów, aby wymusić kontakt i ukazać działanie wiązania.
- Uruchamiamy symulację (klawisz P). Trzy przykładowe klatki symulacji:



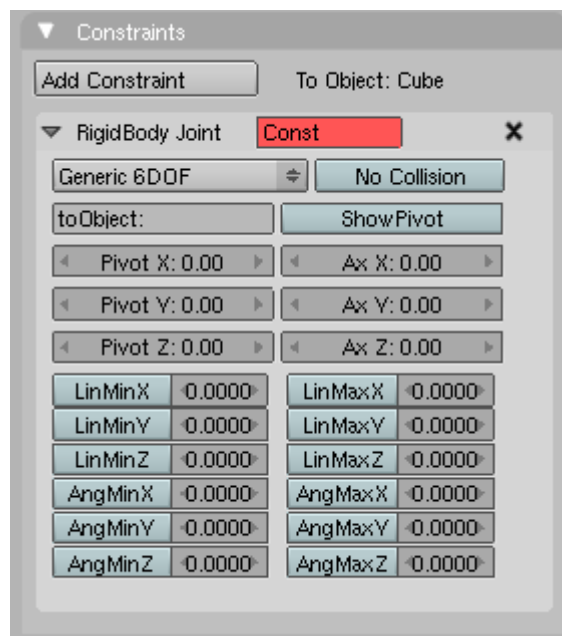
Polecam sprawdzić zachowanie obiektów, gdy wyłączymy w opcjach wiązania przycisk *No collision*. Jeśli chcemy, aby obiekty ulegały kolizji, musimy koniecznie zadbać, aby kolizja nie występowała w miejscu połączenia oraz dla ruchu w pożądanym zakresie.

Plik z gotowym wiązaniem: <http://myinventions.pl/011/WiazanieCD.blend>

Wiązania 6DOF.

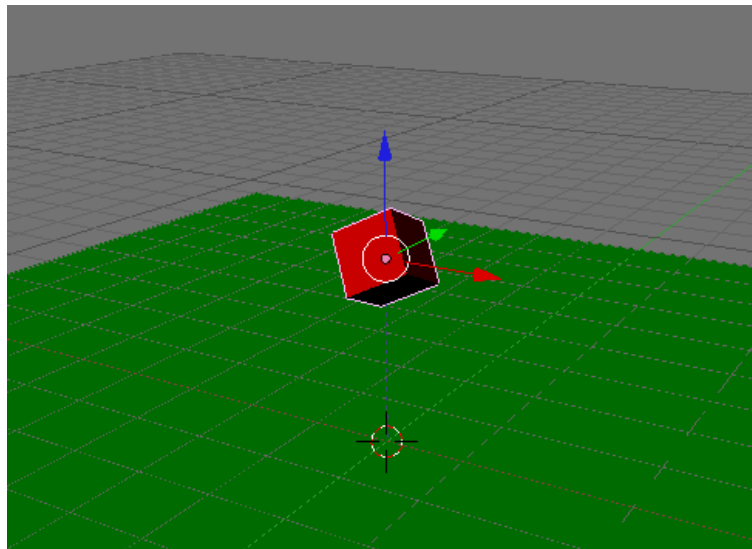
Wiązanie to jest wiązaniem, które daje ogromne możliwości, właściwie daje wszystko czego potrzebujemy, aby stworzyć każde wiązanie. Po rozszyfrowaniu nazwa oznacza 6 stopni swobody (*6 Degrees of Freedom*).

Obiektowi, któremu nadajemy wiązanie ustalmy nieruchomy układ współrzędnych. Nasz obiekt w ogólności może przemieszczać się wzdłuż trzech osi układu odniesienia oraz obracać się wokół tych trzech osi, obiekt ma zatem 6 stopni swobody. Wiązanie 6DOF pozwala nam ograniczyć dowolny ze stopni swobody obiektu zadając wartość maksymalną i minimalną przemieszczenia lub kąta obrotu.



6DOF - przykład.

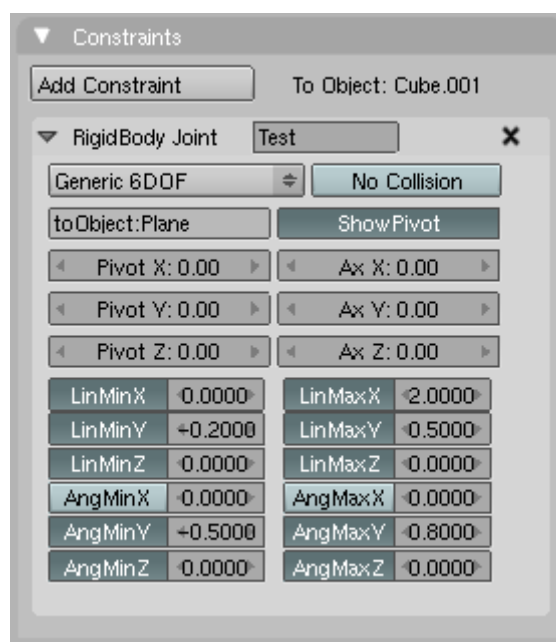
Tworzymy następującą scenę:



W panelu *Logic* zmieniamy typ obiektu na *Rigid Body*, klikamy *Bounds* i wybieramy *Box*.

Uwaga: Dla obszaru kolizji typu *Box* użycie polecenia *Scale and Rotation to ObData* (*ctrl+A*) powoduje, że obszar sześcianu kolizji jest zawsze zorientowany równoległe do osi układu współrzędnych, a nie jest to pożądane w naszym przykładzie. Polecam sprawdzić jak wygląda obszar kolizji przy włączonej opcji *Show Physics Visualization* i trybie *Wireframe* oraz przetestować swobodny spadek obiektu i kolizję z podłożem.

Ustalamy dla obiektu wiązanie o następujących parametrach:

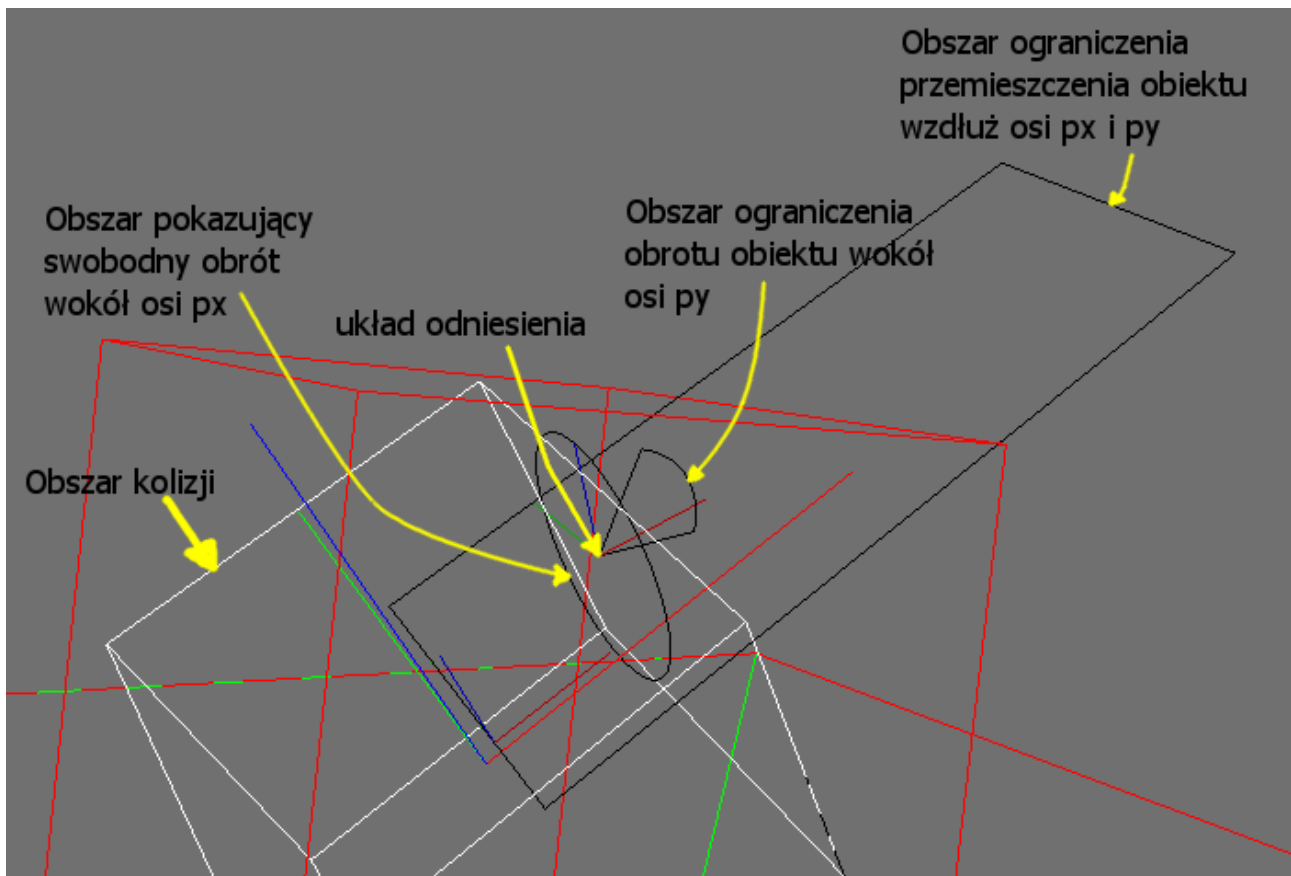


Układ współrzędnych pokrywa się z lokalnym układem obiektu i po uruchomieniu symulacji zostanie w tym miejscu, gdyż jest związany z obiektem *Plane* (nieruchomym podłożem).

W ustawieniach dokonaliśmy:

- ograniczenia położenia obiektu wzdłuż osi px tylko do wartości z przedziału $(0, 2)$,
- ograniczenia położenia obiektu wzdłuż osi py tylko do wartości z przedziału $(-0.2, 0.5)$,
- całkowitego ograniczenia położenia obiektu wzdłuż osi pz ,
- ograniczenia obrotu obiektu wokół osi py tylko do wartości z przedziału $(-0.5, 0.8)$,
- całkowitego ograniczenia obrotu obiektu wokół osi pz ,
- nie ograniczenia obrotu obiektu wokół osi px .

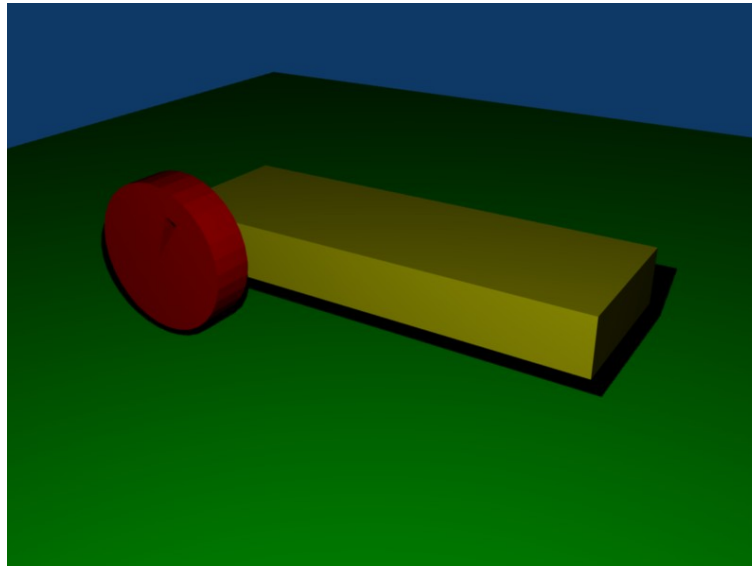
Wszystkie ograniczenia mają swoje graficzne odzwierciedlenie po uruchomieniu symulacji:



Plik z gotowym wiązaniem: http://myinventions.pl/011/6dof_p1.blend

Wiązania w prostym pojeździe.

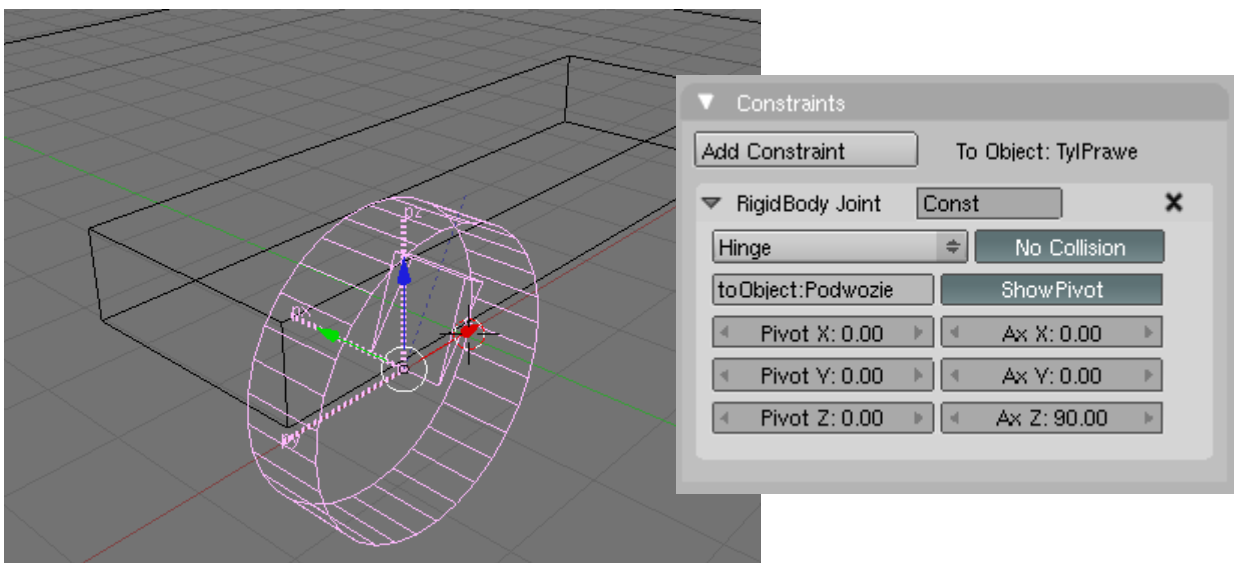
Stwórzmy następującą scenę:



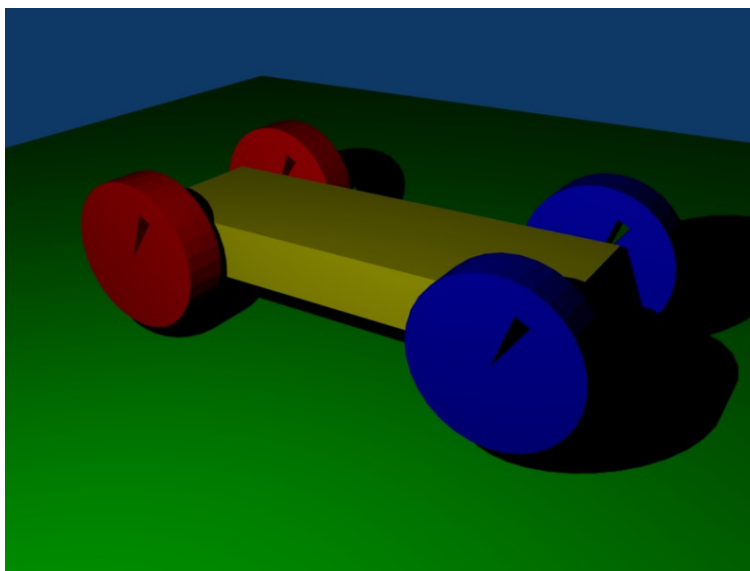
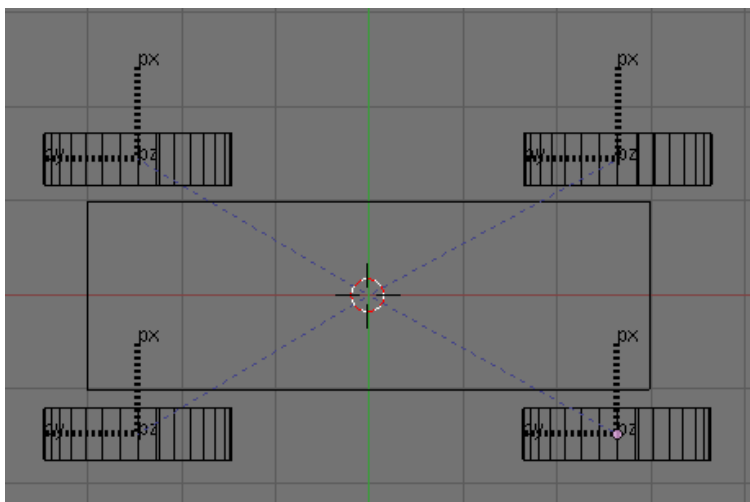
do pobrania: <http://myinventions.pl/011/PojazdModel.blend>

Dla obu obiektów używamy kombinacji klawiszy *Ctrl + A* i wybieramy polecenie *Scale and Rotation to ObData*.

Używamy wiązania zawiasowego (*Hinge*) do przymocowania koła do podwozia, pamiętając o odpowiednim ustawieniu osi *px*.



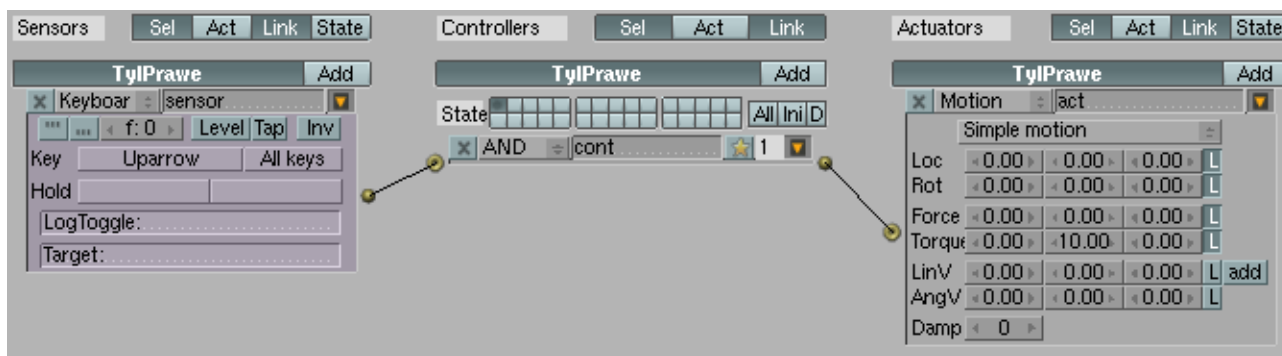
Duplikując koło (Shift+d) otrzymujemy pozostałe koła od razu z gotowymi wiązaniami. Odpowiednio nazywamy koła.



Wszystkie obiekty ustawiamy jako *Rigid Body*. Dla podwozia wybieramy obszar kolizji typu *Box*, dla kół ustalamy typ *Triangle Mesh*. Gdy użyjemy kolizji obszaru kół typu *Convex Hull* w czasie symulacji występują jakieś nieprzewidziane bezpodstawne kolizje obiektów.

Możemy już wstępnie przetestować pojazd – upada z niewielkiej wysokości na podłoże, wiązania trzymają koła.

Zastosujemy napęd do prawego koła tylnego tylnego (tylne koła – czerwone). W panelu logiki obiektu dodajemy sensor typu *Keyboard*, klikamy pole obok napisu *key* i wciskamy na klawiaturze strzałkę do góry. Dodajemy kontroler typu *AND* i łączymy go z sensorem. Dodajemy aktuator typu *Motion* i w drugim polu wiersza *Torque* (moment siły) wpisujemy wartość *10* (moment przykładamy wokół osi y lokalnego układu koła, a nie układu wiązania). Przycisk L w tymże wierszu musi być wciśnięty (ponieważ interesuje nas lokalny a nie globalny układ współrzędnych).



Powtarzamy operację dla napędu wstecznego, przy czym wybieramy klawisz strzałka w dół i wpisujemy wartość momentu napędowego -10.

Zaznaczamy drugie tylne koło, następnie z wciśniętym shiftem zaznaczmy koło, któremu nadaliśmy już napęd. Wciskamy kombinację klawiszy Ctrl+c i wybieramy polecenie *Logic Bricks*. Tym samym dokonaliśmy skopiowania kostek logiki dla drugiego koła – napęd zadziała jednocześnie na oba.

Można już przetestować pojazd. Zauważamy, że pojazd rozpędza się dość wolno, koła doznają poślizgu. Masa podwozia nie może być taka sama jak koła, zwiększmy ją do *200*. Aby uniknąć poślizgu zwiększmy dla każdego koła skalowanie momentu bezwładności względem masy ustawiając *Form: 10*.

Gotowy model do pobrania: <http://myinventions.pl/011/PojazdGotowy.blend>

Artykuł o tym jak zamodelować bardziej zaawansowany pojazd wkrótce.

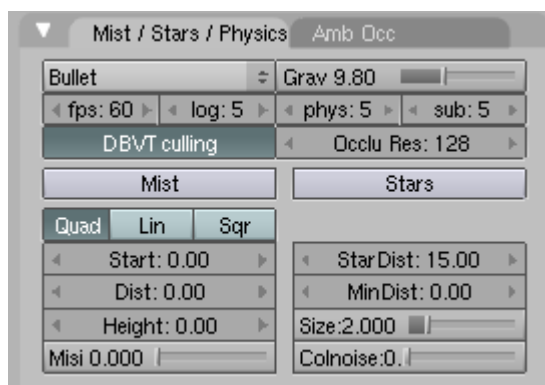
Błędy wyświetlania.

Gdy spotkamy się z sytuacją, w której niektóre ściany obiektów są nie widoczne, bądź 'wywracają się' na drugą stronę przechodzimy do trybu *Edit Mode* (klawisz 'tab'), zaznaczamy cały obiekt (klawisz 'a') i używamy kombinacji klawiszy 'Ctrl+n' i zatwierdzamy polecenie *Recalculate normals outside*.

Sztywność wiązań a dokładność obliczeń.

Temat ten rozważałem już na forum troman.pl. Skorzystajmy w wcześniej zamodelowanego pojazdu i zmienmy masę podwozia na 500. Po włączeniu symulacji zauważamy, że opadając i uderzając o podłoże pojazd zachowuje się jakby miał miękkie zawieszenie. Z racji dużej masy we wiązaniach powstają duże siły, które je 'rozciągają'. Może to być korzystny efekt (właśnie w przypadku zawieszenia), jednak gdy niezbędna jest lepsza sztywność wiązań musimy zwiększyć dokładność obliczeń silnika Bullet.

Zaznaczamy kamerę znajdującą się w scenie, przechodzimy do panelu *Shading* (F5) i w *World Buttons* odnajdujemy zakładkę *Mist/Stars/Physics*. Parametr *Sub* określa ilość kroków symulacji przypadających na klatkę fizyki. Parametr *Phys* określa liczbę klatek fizyki przypadających na klatkę gry. Zatem dla zwiększenia dokładności obliczeń i tym samym zwiększenia sztywności wiązań zwiększamy paramer *Sub*.



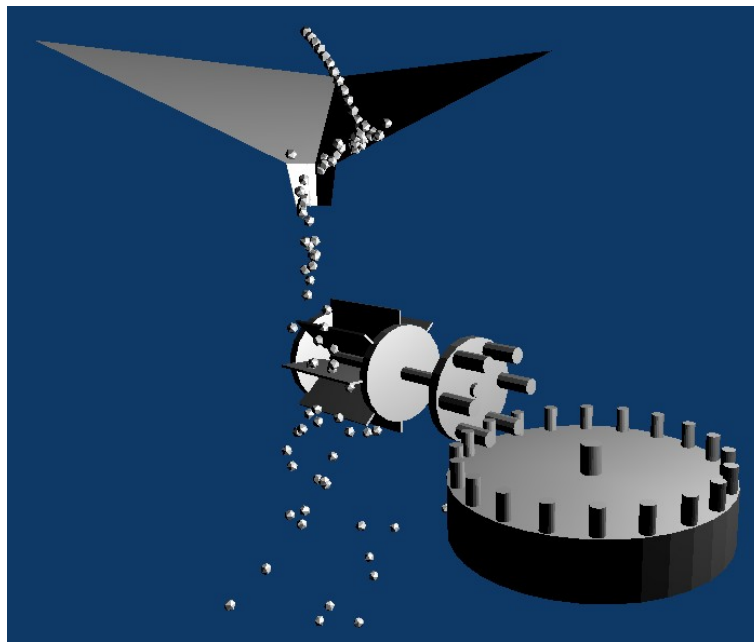
Gdyby maksymalna możliwa do ustawienia wartość parametru *Sub* równa 5 była niewystarczająca, istnieje możliwość ustalenia większych wartości za pomocą skryptu pythona. Treść skryptu (dla *Sub*: 6), który należy jednorazowo uruchomić na początku symulacji za pomocą kontrolera Python wygląda następująco:

```
import PhysicsConstraints
PhysicsConstraints.setNumTimeSubSteps(6)
```

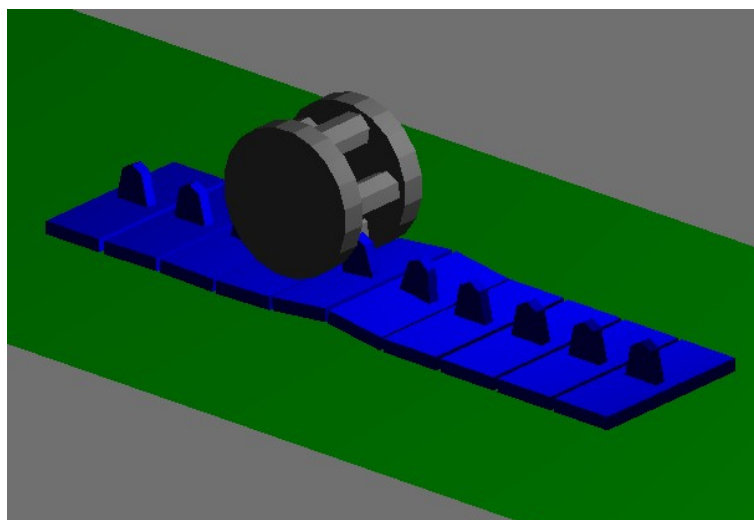

Dodatkowe uwagi.

Dla każdego obiektu typu Rigid Body możemy zablokować obliczenia wzdłuż osi globalnego układu współrzędnej lub wokół tej osi. Dokonać możemy tego w panelu Logic po wciśnięciu przycisku Advanced (przyciski Lock X Axis, Lock X Rot Axis itd.).

Inne przykładowe zastosowaniem wiązań.



<http://myinventions.pl/011/Kola.blend>



<http://myinventions.pl/011/Gasienica.blend>

