

IDŹ DO

PRZYKŁADOWY ROZDZIAŁ



SPIS TREŚCI

KATALOG KSIĄŻEK

KATALOG ONLINE

ZAMÓW DRUKOWANY KATALOG

TWÓJ KOSZYK

DODAJ DO KOSZYKA

CENNIK I INFORMACJE

ZAMÓW INFORMACJE
O NOWOŚCIACH

ZAMÓW CENNIK

CZYTELNIA

FRAGMENTY KSIĄŻEK ONLINE

Fizyka dla programistów gier

Autor: David M. Bourg

Tłumaczenie: Krzysztof Masłowski

ISBN: 83-7361-152-5

Tytuł oryginału: [Physics for Game Developers](#)

Format: B5, stron: 360



Stosowanie praw fizyki pomaga realistycznie i wiarygodnie modelować w grach, symulacjach i animacjach niemal wszystko, co wokół podskakuje, lata, toczy się, ześlizguje; ogólnie mówiąc, nie pozostaje w spoczynku. „Fizyka dla programistów gier” ma być punktem startowym dla osób zainteresowanych zapewnieniem stworzonym grom realistycznego działania opartego na stosowaniu zasad fizyki.

Część pierwsza to przegląd podstawowych koncepcji mechaniki łącznie z pewnymi aspektami i problemami dynamiki ciał sztywnych, w tym kinematyki, sił i kinetyki. W części drugiej te koncepcje zostaną zastosowane do wybranych zagadnień świata rzeczywistego, np. lotu pocisków i samolotów oraz ruchu łodzi i samochodów. Część trzecia wprowadza symulacje w czasie rzeczywistym i pokazuje ich zastosowanie w grach komputerowych. Oto kilka elementów gier w których realizm znacznie wzrasta po zastosowaniu prawa fizyki:

- trajektorie raket i pocisków, włączając w to efekt zużycia paliwa,
- zderzenia obiektów, na przykład kul bilardowych,
- stabilność samochodów na ostrych zakrętach podczas wyścigów,
- dynamika łodzi i innych pojazdów pływających,
- tor lotu piłki baseballowej po uderzeniu kijem,
- charakterystyki lotu samolotów,
- tor lotu karty do gry wrzucanej do kapelusza.

„Fizyka dla programistów gier” nie jest przeznaczona dla ekspertów w dziedzinie fizyki, jednakże autor założył podstawową znajomość tej dziedziny na poziomie szkoły średniej. Potrzebna będzie także znajomość trygonometrii, rachunku wektorowego i macierzowego (informacje na ten temat zostały zawarte w dodatkach na końcu książki). Lektura książki wymaga także podstawowej znajomości klasycznego rachunku różniczkowego i całkowego.



Spis treści

<i>Wstęp</i>	7
<i>Rozdział 1. Koncepcje podstawowe</i>	15
Prawa ruchu Newtona	15
Jednostki i miary	16
Układ współrzędnych	18
Wektory	19
Masa, środek masy i moment bezwładności.....	20
Drugie prawo ruchu Newtona	30
Tensor momentu bezwładności.....	34
<i>Rozdział 2. Kinematyka</i>	41
Wstęp	41
Prędkość i przyspieszenie	42
Przyspieszenie stałe.....	44
Przyspieszenie zmienne.....	46
Kinematyka punktu materialnego w przestrzeni dwuwymiarowej.....	48
Kinematyka punktu materialnego w przestrzeni trójwymiarowej	50
Eksplozja punktu materialnego.....	59
Kinematyka ciała sztywnego	65
Lokalny układ współrzędnych	66
Prędkość kątowa i przyspieszenie kątowe	66

Rozdział 3. Siła	73
Wstęp	73
Pola sił	74
Tarcie	75
Opór cieczy lepkiej	77
Uwaga o ciśnieniu	78
Wypór hydrostatyczny	79
Sprężyny i tłumienie drgań.....	81
Siła i moment siły	82
Rozdział 4. Kinetyka.....	87
Kinetyka punktu materialnego w dwóch wymiarach	88
Kinetyka punktu materialnego w trzech wymiarach	93
Kinetyka ciała sztywnego.....	100
Rozdział 5. Zderzenia	105
Zależność pędu i popędu (impulsu).....	106
Zderzenie	107
Popęd (impuls) liniowy i kątowy.....	114
Tarcie	117
Rozdział 6. Pociski.....	121
Proste trajektorie	122
Opór środowiska	126
Efekt Magnusa.....	134
Zmienna masa	139
Rozdział 7. Samolot.....	141
Geometria.....	142
Siła nośna i siła oporu	145
Inne siły	149
Sterowanie.....	150
Modelowanie	152
Rozdział 8. Statki.....	167
Pływalność	168
Objętość	170
Opór	180
Masa wirtualna	183

Rozdział 9. Poduszkowiec	185
Zasada działania	185
Opór	187
Rozdział 10. Samochody.....	191
Opór	191
Moc.....	192
Droga hamowania	193
Przechylenie drogi.....	194
Rozdział 11. Symulacje w czasie rzeczywistym	197
Całkowanie równań ruchu.....	198
Metoda Eulera	200
Inne metody.....	205
Rozdział 12. Dwuwymiarowy symulator ciała sztywnego.....	211
Model.....	212
Całkowanie	219
Sterowanie lotem	222
Rendering.....	226
Rozdział 13. Implementacja odpowiedzi zderzenia	233
Odpowiedź zderzenia liniowego	234
Efekty związane z ruchem obrotowym	239
Rozdział 14. Obrót ciała sztywnego.....	251
Macierze obrotu	252
Kwaterniony	255
Rozdział 15. Trójwymiarowy symulator ciała sztywnego.....	259
Model.....	260
Całkowanie	264
Sterowanie lotem	267
Rendering.....	271
Rozdział 16. Wiele ciał w przestrzeni trójwymiarowej.....	279
Model.....	280
Całkowanie	294
Odpowiedź zderzenia	295
Tuning	299

Rozdział 17. Układ wielu punktów materialnych	301
Model	302
Całkowanie równań ruchu	311
Odpowiedź zderzenia	312
Tuning	313
Dodatek A Rachunek wektorowy	315
Klasa Vector	315
Funkcje i operatory wektorowe	321
Dodatek B Rachunek macierzowy	325
Klasa Matrix3x3	325
Funkcje i operatory macierzowe	330
Dodatek C. Rachunek kwaternionowy	335
Klasa Quaternion	335
Funkcje i operatory kwaternionowe	338
Bibliografia	347
Skorowidz	353

8

Statki

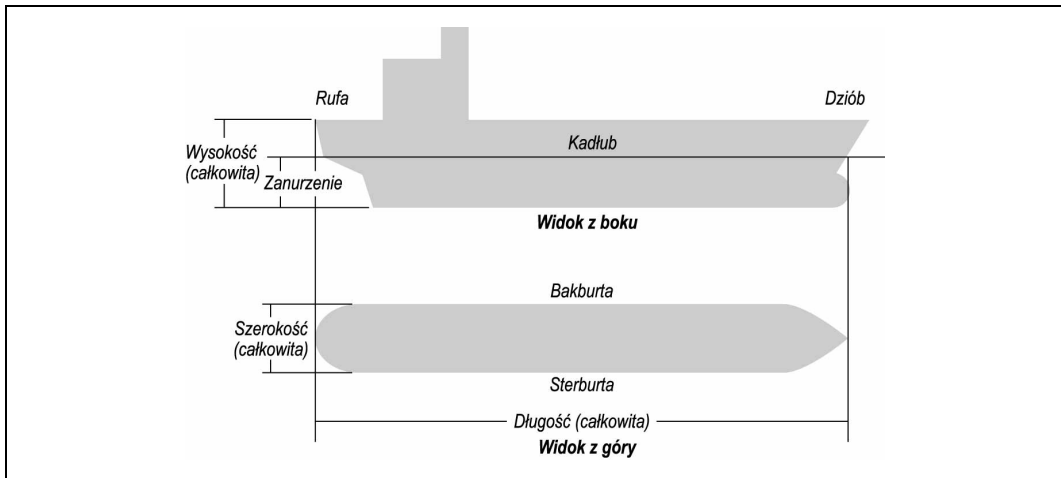
Celem tego rozdziału nie jest nauczenie projektowania statków, lecz jedynie wytłumaczenie na przykładach znaczenia kilku podstawowych pojęć fizycznych — takich jak wypór hydrostatyczny, stabilność, masa wirtualna i opór. Ich znajomość może okazać się potrzebna do tworzenia gier lub symulacji opartych na prawach fizycznych. Pojęcia te wyjaśnimy na przykładzie statku pływającego po powierzchni, lecz znaczna część tego, co powiemy, będzie się odnosiła także do obiektów zanurzonych lub częściowo zanurzonych w cieczy, jak łódź podwodna w wodzie lub balon w powietrzu. Pamiętajmy, że mówiąc o sile wyporu, mamy też na myśli powietrze, które także możemy uważać za ciecz.

Cechą wspólną statków pływających po powierzchni wody (na styku dwóch ośrodków: wody i powietrza) oraz obiektów w pełni zanurzonych, takich jak łodzie podwodne (zanurzone w wodzie) i balony (zanurzone w powietrzu), jest działająca na nie siła wyporu hydrostatycznego, nazywana krócej siłą wyporu lub po prostu wyporem. Jednakże w naturze fizycznej tych obiektów występuje wiele różnic, które omówimy w tym rozdziale. Są one przyczyną odmiennego zachowania się tych obiektów, więc musimy być ich świadomi, jeżeli chcemy, aby nasza symulacja odpowiadała rzeczywistości.

Ponieważ przykłady omówione w tym rozdziale będą dotyczyły statków, zaczniemy od zdefiniowania kilku pojęć dotyczących ich budowy i geometrii. Ułatwi to późniejsze jednoznaczne rozumienie tekstu. Zajmiemy się typowym statkiem *wypornościowym*¹. Termin „wypornościowy” oznacza tu, że statek jest utrzymywany na powierzchni wyłącznie przez siłę wyporu hydrostatycznego, bez udziału dynamicznych sił nośnych, które mają znaczenie w przypadku szybkich łodzi i poduszkowców. Słowo „wyporność” odnosi się do objętości wody wypartej, tzn. przesuniętej w inne miejsce, przez płynący statek (w następnym rozdziale można znaleźć więcej informacji na ten temat).

¹ Takie jest tłumaczenie „displacement ship”, ale w języku polskim nazwa, choć logiczna, nie jest używana, więc dalej będę pisał po prostu „statek”. Jeżeli tekst będzie dotyczył statków innego rodzaju, działających na innej zasadzie, zostanie to zaznaczone — *przyp. tłum.*

Kadłub to część statku bezpośrednio stykająca się z wodą i tę wodę wypierająca. Częściowo zanurzony w wodzie kadłub obejmuje wszystko, co jest w statku i na statku. Długość statku to odległość od dziobu do rufy. W praktyce istnieje kilka różnie określanych długości statku, ale tutaj będziemy się zajmować *długością całkowitą* kadłuba. *Dziób* to część najbardziej wysunięta do przodu, a *rufa* to część najbardziej wysunięta do tyłu. Gdy стоимy na statku z twarzą zwróconą ku dziobowi, po lewej stronie mamy bakburkę, a po prawej — sterburkę². Poza tym mamy *całkowitą wysokość* kadłuba i *całkowitą szerokość* statku, które w skrócie będziemy nazywali wysokością i szerokością statku. Gdy statek pływa po powierzchni wody, odległość od powierzchni wody do dna kadłuba nazywamy *zanurzeniem*. Znaczenie omówionych terminów zostało pokazane na rysunku 8.1.



Rysunek 8.1. Geometria statku

Pływalność

Dla statku najważniejszą rzeczą jest zdolność do pływania. Nie tylko samo unoszenie się na wodzie — statek musi pływać prosto (w pozycji pionowej).

Już w rozdziale 3. mówiliśmy o wyporności i stwierdziliśmy, że siła wyporności działająca na zanurzone ciało jest funkcją objętości zanurzonej części tego ciała. Prawo Archimedeasa mówi, że ciężar ciała pływającego po powierzchni jest równy ciężarowi cieczy wypartej przez to ciało (cieczy o objętości równej objętości zanurzonej części ciała). Jest to bardzo ważne prawo, które stwierdza, że statek o danym ciężarze musi mieć objętość wystarczającą do wyparcia takiej ilości wody, aby jej ciężar pozwolił na utrzymanie statku na powierzchni. Ponadto prawo to pozwala na łatwe określenie ciężaru statku: wystarczy obliczyć objętość wypartej wody i obliczyć jej ciężar. W słowniku morskim wyporność jest równoważna ciężarowi.

² Pochodzenie tych nazw wyjaśniłem w przypisie na początku poprzedniego rozdziału — *przyp. tłum.*

Siła wyporności działająca na dowolny obiekt może być obliczona według wzoru:

$$F_B = \rho g \nabla$$

gdzie ∇ jest objętością zanurzonej części obiektu, ρ — gęstością cieczy, w której obiekt jest zanurzony, a g — przyspieszeniem grawitacyjnym. Ponieważ wypór jest siłą, ma wielkość, kierunek i zawsze działa wprost do góry wzdłuż prostej przechodzącej przez środek wyporu. Środek wyporu jest geometrycznym środkiem części zanurzonej.

Jeżeli statek pływający po powierzchni jest w stanie równowagi, jego środek wyporu musi leżeć dokładnie pod środkiem ciężkości. Ciężar statku, siła skierowana wprost w dół i zaczepiona w środku ciężkości, działa przeciwnie do wyporu. Gdy statek jest w równowadze, obie te siły — wypór i ciężar (dokładniej: siła wyporu i siła ciężkości) — mają jednakową wielkość i przeciwny zwrot.

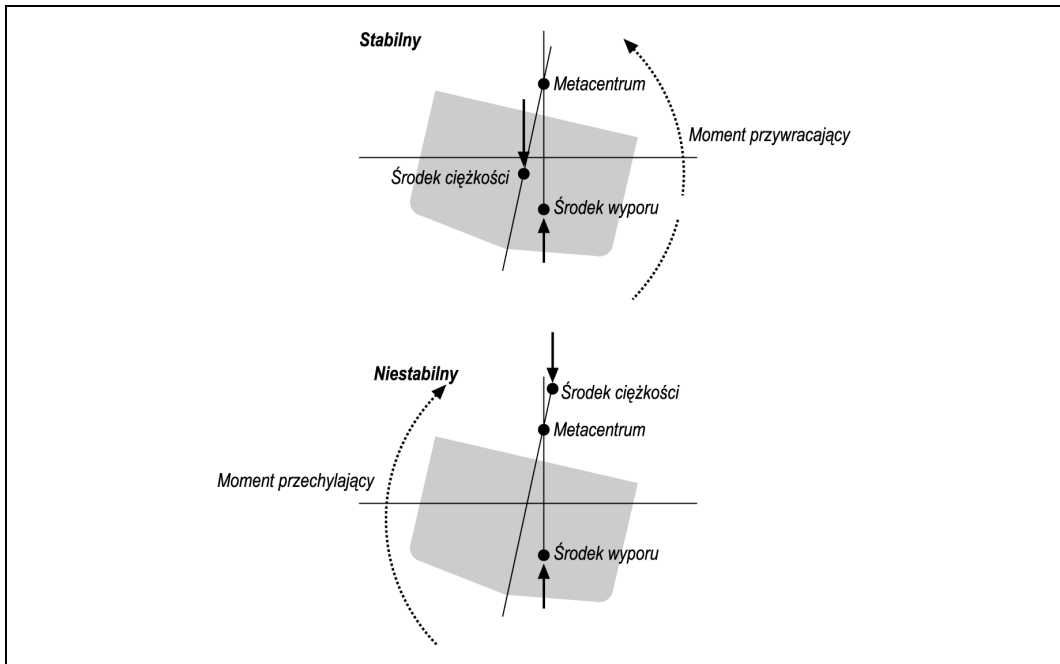
Jeżeli statek się nachyla lub przechyla³ (czyli ma odpowiednio: nachylenie wzdłużne lub przechył poprzeczny), zmienia się położenie i wielkość części zanurzonej, a więc również zmienia się miejsce środka wyporu. Na przykład, jeżeli statek przechyla się na sterburtę, jego środek wyporu przesuwają się także w stronę tej burty. Wówczas siły wyporu i ciężkości przestają działać wzdłuż jednej prostej, co powoduje powstanie momentu siły działającego na statek. Ten moment jest równy iloczynowi siły ciężkości przez odległość między prostymi, wzdłuż których działają obie siły.

Teraz wróćmy do sprawy prostego (pionowego) ustawienia statku podczas pływania. Kiedy statek się przechyla, nie chcemy, aby wywrócił się dnem do góry. Chcemy, aby gdy przestanie działać siła, która spowodowała przechył, łagodnie powrócił do pozycji pionowej. Powrót ten powinien nastąpić niezależnie od rodzaju siły, będącej przyczyną przechyłu, a mógł to być na przykład wiatr. Mówiąc w skrócie, chcemy, aby statek był stabilny. Aby tak było, punkt, w którym linia akcji siły wyporu przecina linię środkową kadłuba — zwany punktem metacentrycznym (w skrócie metacentrum), musi znajdować się powyżej środka ciężkości. Wówczas moment działających sił powstający przy przechyle dąży do przywrócenia położenia pionowego. Jeżeli punkt metacentryczny położony jest poniżej środka ciężkości, moment sił spowodowany przechylem wywołuje zwiększenie tego przechyłu, co doprowadza do wywrócenia statku. Oba scenariusze zostały pokazane na rysunku 8.2.

Każdy, kto ma jakieś pojęcie o żeglarskim, wie, że bardzo ważne jest utrzymanie położenia środka ciężkości łodzi jak najniżej. Pomaga to zwiększyć wysokość położenia metacentrum nad środkiem ciężkości, co z kolei zwiększa stabilność.

Sytuacja jest inna w przypadku obiektów zanurzonych całkowicie, na przykład łodzi podwodnych i balonów. Siła wyporu zawsze działa w punkcie geometrycznego środka obiektu, zwanym tu środkiem wyporu. Dla zachowania stabilności punkt ten musi się znajdować powyżej środka ciężkości. Jeżeli wówczas nastąpi wychylenie, linie akcji siły ciężkości

³ Porównaj rysunek 7.4, gdzie te pojęcia zostały wyjaśnione na przykładzie samolotu — *przyp. tłum.*



Rysunek 8.2. Stabilność statku

i siły wyporu zostaną rozdzielone, a powstały w wyniku tego moment sił będzie działał tak, by przywrócić wyprostowaną pozycję obiektu. W przeciwnym razie obiekt będzie niestabilny — jak przy ułożeniu jednej piłki na drugiej. Jeżeli środek wyporu znajdzie się poniżej środka ciężkości, moment sił powstający przy przechyle będzie dążył do wywrócenia obiektu do góry nogami, tak aby wzajemne położenie środka ciężkości i środka wyporu zostało odwrócone.

Problemem jest obliczenie objętości części zanurzonej i położenia jej geometrycznego środka, co nie sprawia kłopotów jedynie w przypadku ciał o najprostszej geometrii. Na przykład kadłuby statków mają zwykle kształty skomplikowane, pełne zakrzywień, wgłębień i wypukłości. W rezultacie obliczenie objętości wypieranej wody wymaga zwykle stosowania technik całkowania numerycznego. Jedną z metod poznamy w następnym podrozdziale.

Objętość

Istnieją różne techniki i algorytmy obliczania objętości wywodzące się z różnych dziedzin nauki i techniki. Ich celem jest optymalne wykonywanie szczególnych zadań, a ich natura i forma zależy geometrii obiektów, których objętość należy obliczyć. Na przykład, w świecie grafiki komputerowej obiekty są zwykle reprezentowane przez trójkątowane (triangu-

lowane) wielościany i figury wielowierzchołkowe⁴. Istnieje wiele algorytmów służących do obliczania ich objętości, lecz w zasadzie sprowadzają się one do dzielenia figur na wiele czworościanów, obliczania ich objętości i sumowania (ta technika zostanie wkrótce pokazana). Istnieje również odmienna technika, wywodząca się z chemii, gdzie potrzebne jest obliczanie objętości molekuł. Polega ona na obliczaniu objętości wielu odpowiednio dobranych i ustawionych kul.

Metody te znajdują zastosowanie również w dziedzinie budowy statków. Tradycyjne techniki obliczania objętości statków polegały na całkowaniu powierzchni przekrojów wzdłuż długości kadłuba. Warto zauważyć, że choć techniki te różnią się sposobami implementacji, wszystkie opierają się na całkowaniu numerycznym polegającym na dyskretyzacji obiektu przez podzielenie go na mniejsze elementy o prostszej geometrii, pozwalającej na obliczanie objętości i sumowaniu tych objętości cząstkowych w celu obliczenia objętości całkowitej.

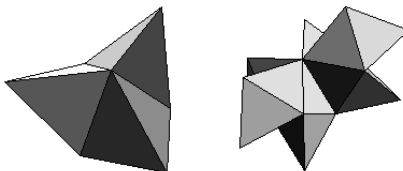
Przyjrzyjmy się prostemu przykładowi znalezienia położenia środka i obliczenia objętości triangularyzowanego sześciąnu. Sześciąn ten został pokazany na rysunku 8.3.

Jako przykładu użyjemy sześciąnu, gdyż w tym przypadku wyniki obliczeń możemy w prosty sposób zweryfikować. Należy pamiętać, że metoda, którą zastosujemy, może być używana do figur o geometriach o wiele bardziej skomplikowanych, jeżeli tylko całość zostanie podzielona na *proste czworościany*. Oznacza to, że muszą być spełnione następujące warunki:

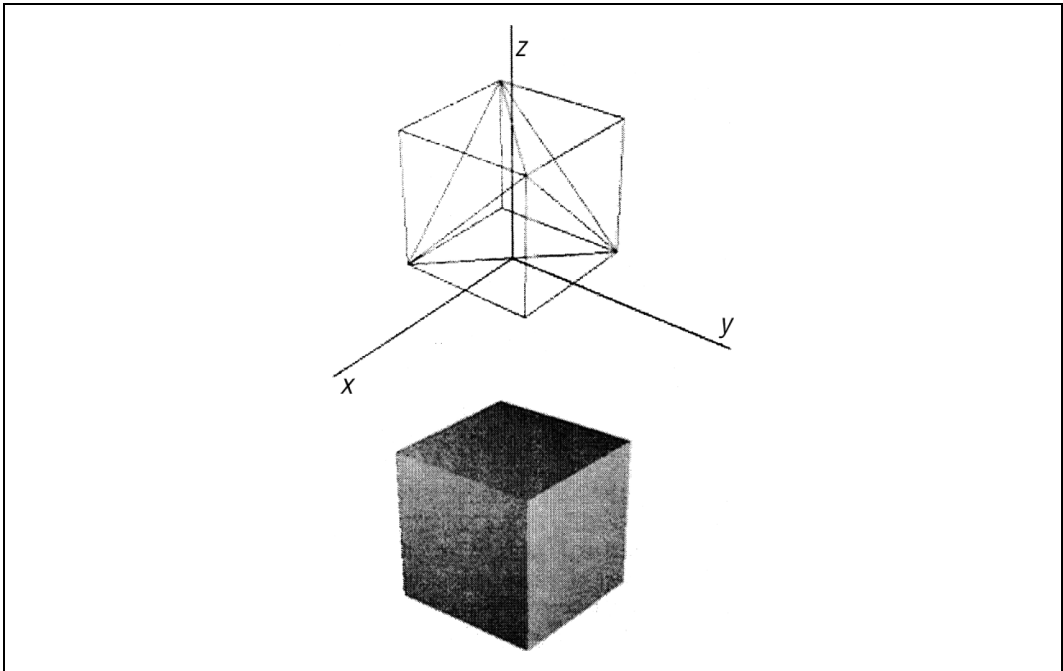
- Wszystkie ściany tworzące obiekt muszą być trójkątami.
- Obiekt nie może mieć dziur.
- Obiekt musi obejmować zamkniętą przestrzeń. Oznacza to, że nie może być żadnych swobodnych ścian ani krawędzi. Każda krawędź musi być połączona z dwoma wierzchołkami i musi być częścią wspólną dokładnie dwóch ścian.
- Obiekt musi spełniać *wzór Eulera*, według którego liczba wierzchołków minus liczba krawędzi plus liczba ścian musi być równa 2: $W - K + S = 2$.

Jak już zostało wcześniej powiedziane, podstawą tej metody jest podział obiektu na określoną liczbę czworościanów, obliczenie objętości każdego z nich i zsumowanie objętości składowych w celu otrzymania objętości całkowitej. Zastosowany podział na czworościany może także posłużyć do znalezienia położenia środka geometrycznego obiektu (środka

⁴ Warto to przyjrzeć się takim twórcom. Oto kilka przykładów:

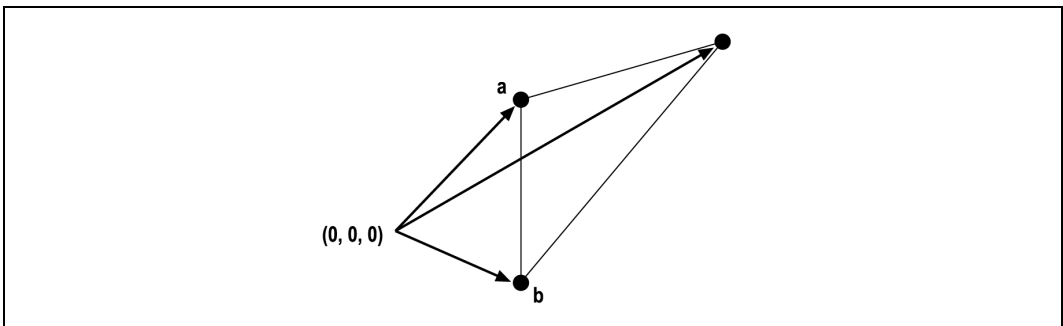


— przyp. tłum.



Rysunek 8.3. Striangulowany sześcian

objętości) metodą podobną do stosowanej w celu obliczenia położenia środka masy (co zostało omówione w rozdziale 1.). W tym przypadku masy powinny zostać zastąpione przez objętości czworościanów. Rysunek 8.4 pokazuje sposób konstruowania czworościanu.



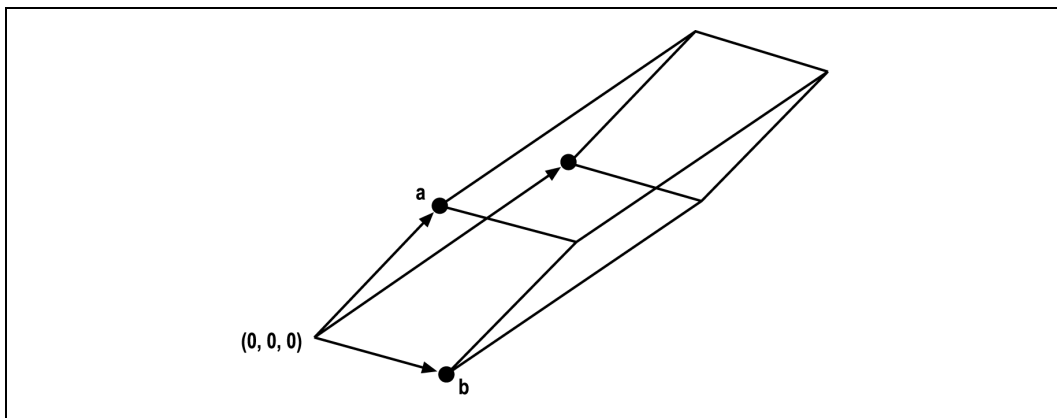
Rysunek 8.4. Czworoscian

Tutaj punkt początkowy $(0, 0, 0)$ i trzy wierzchołki służą do zdefiniowania czworościanu. O krawędziach możemy myśleć jako o wektorach o wspólnym początku w punkcie $(0, 0, 0)$ i końcach w poszczególnych wierzchołkach. Zauważmy, że wierzchołki każdej ściany są określane w porządku przeciwnym do kierunku obrotu wskazówek zegara, gdy patrzymy na ścianę z zewnątrz.

Objętość takiego czworościanu możemy obliczyć za pomocą iloczynu mieszanego wektorów (przykładowy kod został umieszczony w dodatku A). Dla wektorów \mathbf{a} , \mathbf{b} i \mathbf{c} z rysunku 8.4 iloczyn mieszany ma postać:

$$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})$$

Ma on prostą interpretację fizyczną — jest równy objętości równoległościanu o podstawie czworokątnej⁵, o długościach boków a , b i c , pokazanego na rysunku 8.5.⁶



Rysunek 8.5. Równoległościan

Dla nas ważną rzeczą jest fakt, że objętość czworościanu rozpiętego na trzech wektorach (jak na rysunku 8.4) jest równa jednej szóstej objętości równoległościanu rozpiętego na tych samych wektorach. Zatem objętość czworościanu dana jest wzorem:

$$[\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})]/6$$

Znalezienie geometrycznego środka czworościanu jest stosunkowo proste: należy po prostu obliczyć średnią wartość współrzędnych wszystkich czterech wierzchołków. Należy pamiętać, że nawet gdy jeden z wierzchołków leży w początku układu współrzędnych, należy go uwzględnić przy wyliczaniu średniej. Przyjmując oznaczenia z rysunku 8.4 i stosując notację wektorową, geometryczny środek czworościanu znajdujemy, stosując wzór:

$$\mathbf{d} = (\mathbf{a} + \mathbf{b} + \mathbf{c})/4$$

Założono tu, że czwarty wierzchołek znajduje się w początku układu współrzędnych $(0, 0, 0)$. Dlatego w liczniku widzimy sumowanie jedynie trzech wektorów, choć suma jest potem dzielona przez 4.

⁵ Równoległościan o podstawie czworokątnej jest bryłą mającą trzy pary ścian równoległych. Jeżeli jego ściany są prostopadłe, nazywamy go prostopadłościanem. Sześcian jest prostopadłościanem, którego krawędzie są równej długości.

⁶ Warto wiedzieć, że: $|\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})| = |\mathbf{b} \cdot (\mathbf{c} \times \mathbf{a})| = |\mathbf{c} \cdot (\mathbf{a} \times \mathbf{b})|$ — przyp. tłum.

Na potrzeby tego przykładu przygotowano prostą klasę o nazwie `Body3D`, która przechowuje dane o wierzchołkach i ścianach i jest implementacją metody wczytywania danych o obiekcie z pliku oraz metody obliczania objętości i położenia środka geometrycznego obiektu:

```
#define MAX_NUM_VERTICES 100
#define MAX_NUM_FACES 100

typedef struct VertexTag {
    float x; // współrzędna x wierzchołka
    float y; // współrzędna y wierzchołka
    float z; // współrzędna z wierzchołka
} TVertex;

typedef struct FaceTag {
    // wierzchołki definiujące ścianę są ułożone przeciwnie do
    // ruchu wskazówek zegara, jeżeli patrzymy na ścianę
    // z zewnątrz obiektu

    int a; // 1-szy wierzchołek (jego indeks na liście wierzchołków)
    int b; // 2-gi wierzchołek (jego indeks na liście wierzchołków)
    int c; // 3-ci wierzchołek (jego indeks na liście wierzchołków)
} TFace;

//-----//
// klasa Body3D reprezentuje prosty triangulowany wielościan
//-----//
class Body3D {
public:
    int nFaces; // liczba trójkątnych ścian
    int nVertices; // liczba wierzchołków
    TVertex Vertex[MAX_NUM_VERTICES]; // lista wierzchołków
    TFace Face[MAX_NUM_FACES]; // lista ścian
    float Volume; // objętość całkowita
    Vector Centroid; // położenie środka obiektu

    Body3D(void); // konstruktor

    void ReadData(char *filename); // czyta dane wierzchołków/ścian
    void CalculateProperties(void); // oblicza objętość i środek obiektu
};
```

Każdy składnik tej klasy można zidentyfikować, czytając komentarze umieszczone w kodzie, więc nie będziemy się tym dokładniej zajmować. Jednakże przyjrzyjmy się dwóm użytym metodom (część nazwana konstruktorem nie jest ciekawa, gdyż po prostu zeruje wszystko):

`ReadData` po prostu odczytuje dane obiektu z pliku tekstowego:

```
void Body3D::ReadData(char *filename)
{
    FILE *fptr;
    int i;

    fptr = fopen(filename, "r");
```

```

fscanf(fp, "%d\n", &nVertices );
for(i=0; i< nVertices; i++)
{
    fscanf( fp,
            "%f %f %f\n",
            &(Vertex[i].x),
            &(Vertex[i].y),
            &(Vertex[i].z));
}

fscanf(fp, "%d$\n", &nFaces );
for(i=0; i< nFaces; i++)
{
    fscanf( fp,
            "%d %d %d\n",
            &(Face[i].a),
            &(Face[i].b),
            &(Face[i].c)); // przeciwnie do ruchu wskazówek zegara
}

fclose(fp);
}

```

Liczba całkowita w pierwszej wierszu pliku określa liczbę wierzchołków. W następnych wierszach są zapisane współrzędne x , y , z (float — zmiennoprzecinkowe) kolejnych wierzchołków, po jednym wierzchołku w wierszu. Po wszystkich wierzchołkach wczytywana jest znów liczba całkowita, określająca liczbę ścian. W kolejnych wierszach mamy dane ścian — w każdym wierszu trzy numery wierzchołków tworzących ścianę (w porządku przeciwnym do ruchu wskazówek zegara). Oto przykładowy plik definiujący sześcian o długości krawędzi równej dwóm jednostkom z jednym wierzchołkiem umieszczonym w początku układu współrzędnych i z podstawą położoną na płaszczyźnie xy (patrz rysunek 8.3).

```

8
-1.000000 -1.000000 0.000000
-1.000000 -1.000000 2.000000
1.000000 -1.000000 0.000000
1.000000 -1.000000 2.000000
-1.000000 1.000000 0.000000
1.000000 1.000000 0.000000
1.000000 1.000000 2.000000
-1.000000 1.000000 2.000000
12
2 3 1
2 1 0
4 5 2
4 2 0
6 3 2
6 2 5
6 7 1
6 1 3
6 5 4
6 4 7
1 7 4
1 4 0

```

Następna metoda `CalculateProperties` jest prawdziwym „sercem” omawianego przykładu. Przeprowadza proces dyskretyzacji obiektu, dzieląc go na czworościany potrzebne do obliczenia całkowitej objętości i położenia środka obiektu. Najpierw zobaczymy kod, a potem przystąpimy do jego omówienia:

```
void Body3D::CalculateProperties(void)
{
    Vector a;
    Vector b;
    Vector c;
    int i;
    float dv = 0;
    float vol = 0;
    Vector d;
    Vector dmom;

    for(i=0; i < nFaces; i++)
    {
        a.x = Vertex[Face[i].a].x;
        a.y = Vertex[Face[i].a].y;
        a.z = Vertex[Face[i].a].z;

        b.x = Vertex[Face[i].b].x;
        b.y = Vertex[Face[i].b].y;
        b.z = Vertex[Face[i].b].z;

        c.x = Vertex[Face[i].c].x;
        c.y = Vertex[Face[i].c].y;
        c.z = Vertex[Face[i].c].z;

        dv = (TripleScalarProduct(a, b, c)) / 6.0f;
        vol += dv;
        d = ((a + b + c) / 4);
        dmom += (d * dv);
    }
    Volume = vol;
    Centroid = dmom / vol;
}
```

Zauważmy, że funkcja definiuje kilka zmiennych lokalnych `a`, `b` i `c` typu `Vector`, które posłużą do reprezentowania wektorów prowadzonych od początku układu do wierzchołków wszystkich ścian czworościanu. Typ `Vector` został omówiony w dodatku A. Zmienna `i` typu `int` jest zmienną licznikową. Zmienne `dv` i `vol` typu `float` to odpowiednio: objętość pojedynczego czworościanu i bieżąca objętość całkowita obiektu. Zmienne `d` i `dmom` typu `Vector` to odpowiednio: współrzędne pojedynczego czworościanu i bieżący sumaryczny pierwszy moment objętości⁷ wszystkich czworościanów.

W tej metodzie po nadaniu wartości początkowych wszystkim zmiennym wykonywana jest iteracja przez listę wszystkich ścian obiektu, podczas której następuje tworzenie wektorów `a`, `b` i `c` dla każdego czworościanu uformowanego przez punkt początkowy układu i bieżącą ścianę `Face[i]`. Następnie jest obliczany iloczyn mieszany wektorów i wynik zostaje podzielony przez 6. Rezultatem tego obliczenia jest objętość czworościanu `dv`, która

⁷ Przez analogię do pierwszego momentu masy — patrz podrozdział „Masa, środek masy i moment bezwładności” w rozdziale 1. — *przyp. tłum.*

zostaje dodana do bieżącej całkowitej objętości vol . Następnym etapem obliczeń jest znalezienie środka czworoscianu d , pomnożenie go przez objętość dv i dodanie otrzymanego iloczynu do $dmom$ (bieżącej całkowitej wartości pierwszego momentu objętości). Na zakończenie iteracji po przejściu przez wszystkie ściany całkowita objętość jest równa vol , a położenie środka jest sumą pierwszych momentów podzieloną przez całkowitą objętość — $dmom/vol$.

To wszystko na ten temat. Dla przetestowania tej klasy został przygotowany program do obsługi konsoli, który korzysta z `Body3D`, wczytuje dane sześcianu i wylicza jego objętość oraz położenie środka. Jego funkcja główna wygląda następująco:

```
int main(int argc, char* argv[])
{
    Body3D body = Body3D();
    float volume = 0;
    int i;
    Vector centroid;

    // wczytanie danych obiektu
    body.ReadData("cube.txt");

    // echo danych na konsolę
    printf("Liczba wierzchołków = %d\n", body.nVertices);
    for(i=0; i< body.nVertices; i++)
        printf(" Wierzchołek %d: x=%f y=%f z=%f\n",
            i,
            body.Vertex[i].x,
            body.Vertex[i].y,
            body.Vertex[i].z);

    printf("Liczba ścian = %d\n", body.nFaces);
    for(i=0; i<body.nFaces; i++)
        printf(" Ściana %d: a=%d b=%d c=%d\n",
            i,
            body.Face[i].a,
            body.Face[i].b,
            body.Face[i].c);
    // obliczanie objętości i środka
    body.CalculateProperties();

    // wyświetlanie wyników na konsoli
    printf("\n");
    printf("Objętość = %f\n", body.Volume);
    printf("\n");
    printf("Środek:\n");
    printf("x=%f y=%f z=%f\n", body.Centroid.x, body.Centroid.y,
    body.Centroid.z);
    printf("\n");

    printf("Koniec\n");

    return 0;
}
```

Po uruchomieniu programu, okazuje się, że objętość sześcianu wynosi 8 jednostek sześciennych, a środek znajduje się w punkcie (0, 0, 1).

Bardzo interesującym testem jest użycie tego programu do obliczenia objętości i środka obiektu przypominającego (choć w znacznym uproszczeniu) kadłub łodzi.

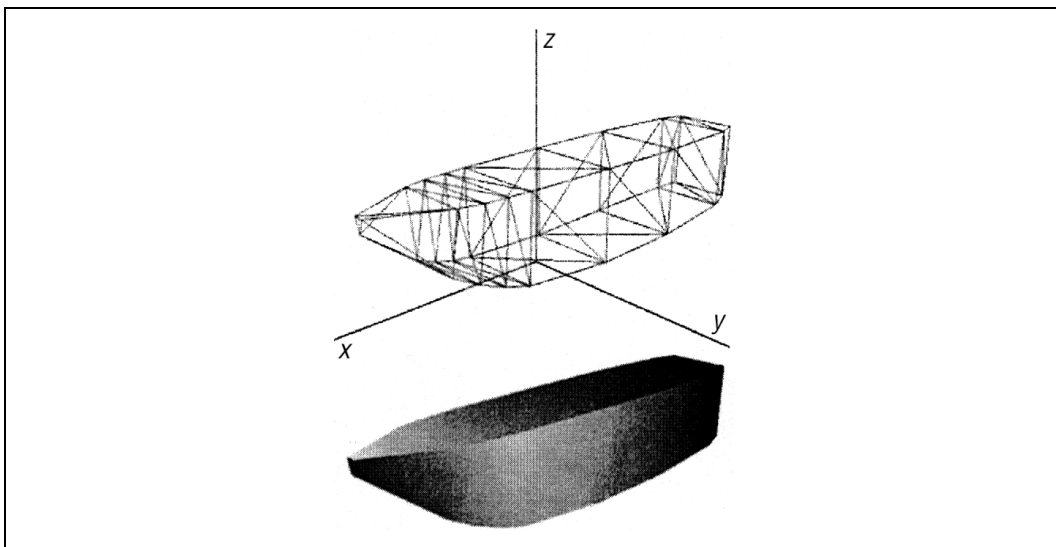
Plik danych opisujących kadłub łodzi, pokazany na rysunku 8.6, wygląda następująco:

```

36
-5.500000 -0.693775 0.281525
-5.500000 -0.693775 2.000000
3.888562 0.000000 1.700000
3.888562 -0.100000 1.991344
-5.500000 0.693775 0.281525
3.888562 0.000000 1.700000
3.888562 0.100000 1.991344
-5.500000 0.693775 2.000000
1.000000 0.900000 0.105572
1.000000 0.950000 2.000000
1.000000 -0.950000 2.000000
1.000000 -0.900000 0.105572
1.500000 0.794000 0.219941
1.500000 0.900000 2.000000
1.500000 -0.900000 2.000000
1.500000 -0.793988 0.219941
2.000000 0.680000 0.387096
2.000000 0.870000 2.000000
2.000000 -0.870381 2.000000
2.000000 -0.680000 0.387096
2.500000 0.538000 0.633431
2.500000 0.747000 2.000000
2.500000 -0.747214 2.000000
2.500000 -0.538416 0.633431
-1.000000 -1.000000 0.000000
-1.000000 -1.000000 2.000000
-1.000000 1.000000 2.000000
-1.000000 1.000000 0.000000
-3.000000 -1.000000 0.070381
-3.000000 -1.000000 2.000000
-3.000000 1.000000 2.000000
-3.000000 1.000000 0.070381
-5.000000 -0.864029 0.211143
-5.000000 -0.864029 2.000000
-5.000000 0.864029 2.000000
-5.000000 0.864029 0.211143
68

2 3 22
2 22 23
20 5 23
5 2 23
6 3 2
6 2 5
6 21 3
3 21 22
6 5 20
6 20 21
1 7 4
1 4 0
34 35 4
34 4 7
34 7 1
34 1 33
32 33 1

```



Rysunek 8.6. Kadłub łodzi

```
32 1 0
4 35 0
0 35 32
13 12 8
13 8 9
13 9 14
14 9 10
15 14 10
15 10 11
8 12 15
8 15 11
17 16 12
17 12 13
17 13 18
18 13 14
19 18 14
19 14 15
12 16 19
12 19 15
21 20 16
21 16 17
21 17 22
22 17 18
23 22 18
23 18 19
16 20 23
16 23 19
11 10 25
11 25 24
9 26 10
10 26 25
9 8 27
9 27 26
27 8 11
27 11 24
24 25 29
24 29 28
```

```

26 30 29
26 29 25
26 27 31
26 31 30
31 27 24
31 24 28
28 29 33
28 33 32
30 34 33
30 33 29
30 31 35
30 35 34
35 31 32
32 31 28

```

W wyniku tego testu otrzymamy objętość równą 28,67 jednostek sześciennych i położenie środka określone wektorem $(-1,43; 0,00; 1,08)$.

Należy zaznaczyć, że w podanych kodach przykładowych brakuje procedur sprawdzających błędy. Zostały pominięte w celu przedstawienia demonstrowanej metody w prostszej postaci. W rzeczywistym kodzie zawsze należy pamiętać o uzupełnieniu tego braku. Jeżeli chcemy rzeczywiście wykorzystać ten kod, trzeba przede wszystkim dodać procedurę sprawdzającą dane wejściowe, aby być pewnym, że obiekt jest prostym, triangulowanym wielościanem oraz że w funkcji `CalculateProperties` nigdzie nie wystąpi dzielenie przez zero.

Opór

W rozdziale 6. omawialiśmy siły oporu działające na obiekt poruszający się w cieczy. Szczególnie zwróciliśmy uwagę na opór wywołany przez siły tarcia i przez ciśnienie. Na statki poruszające się po powierzchni wody, oprócz tych sił oporu, działają jeszcze inne, występujące na styku wody i powietrza. Wzór, w którym całkowity opór zostanie przedstawiony w rozbiciu na trzy główne składowe, będzie wyglądał mniej więcej tak:

$$R_{\text{całk}} = R_{\text{tarcia}} + R_{\text{ciśn}} + R_{\text{fal}}$$

Omówimy wszystkie składniki tej sumy i poznamy przydatne do ich obliczania wzory empiryczne. Zaczniemy jednak od stwierdzenia, że te wzory są jedynie bardzo ogólnym przybliżeniem i stosuje się je jedynie wówczas, gdy niewiele wiadomo o geometrii statku. W praktyce projektowania statków używa się ich tylko do przybliżonego obliczania oporu we wstępnej fazie projektowania. Są przydatne do szacowania wpływu zmian poszczególnych głównych parametrów na wielkość oporu.

Pierwszą składową jest opór pochodzący od tarcia powierzchni podwodnej części kadłuba podczas przesuwania się w wodzie. Jest to to samo tarcie, o którym mówiliśmy w rozdziale 6. Do jego obliczania w przypadku statków możemy korzystać z zestawu wzorów empirycznych (R_{tarcia} oznaczymy w skrócie przez R_t):

$$R_t = (1/2)\rho V^2 SC_t$$

W tym wzorze ρ jest gęstością wody, V — szybkością statku, S — powierzchnią przekroju poprzecznego podwodnej części kadłuba, a C_t — współczynnikiem oporu tarcia:

$$C_t = 0,075 / (\log_{10} R_n - 2)^2$$

gdzie R_n jest liczbą Reynoldsa zdefiniowaną w rozdziale 6. i zależną od długości kadłuba. Ten wzór, zaakceptowany w roku 1957 przez *International Towing Tank Conference (ITTC)*⁸, jest szeroko stosowany w obliczeniach konstrukcyjnych statków do obliczania współczynnika oporu tarcia.

Aby stosować wzór na R_t , trzeba również znać S — powierzchnię przekroju poprzecznego podwodnej części kadłuba. Można ją wyliczyć wprost za pomocą technik całkowania numerycznego, podobnych do tych, które posłużyły nam do obliczania objętości, lub za pomocą kolejnego wzoru empirycznego:

$$S = C_{ws} \sqrt{\nabla L}$$

W tym wzorze ∇ jest objętością wypartej wody, L — długością statku, a C_{ws} — współczynnikiem powierzchni zwilżonej. Jest on funkcją kształtu statku, jego stosunku szerokości kadłuba do zanurzenia. Statystycznie jego wartość dla typowych kształtów kadłuba waha się od 2,6 do 2,9.

Opór ciśnieniowy działający na statki jest taki sam jak omówiony w rozdziale 6. opór ciśnieniowy działający na pociski. Jak pamiętamy, przyczyną jego powstawania jest lepkość cieczy powodująca powstawanie za obiektem obszaru o mniejszym ciśnieniu. W przypadku statków o dowolnej geometrii obliczenie wartości tej siły jest trudne. Przybliżone jej wartości są obliczane za pomocą komputerowych algorytmów dynamiki cieczy, których użycie wymaga jednak szczegółowych informacji o geometrii kadłuba i pochłania dużo czasu obliczeniowego. Można też wykonywać testy na modelach wykonanych w skali, po czym ekstrapolować wyniki na statek normalnej wielkości.

Opór falowy, tak samo jak ciśnieniowy, jest trudny do obliczenia i w praktyce często są stosowane testy na modelach. Opór falowy jest spowodowany przez przekaz energii lub momentu ze statku do cieczy. Innymi słowy, jest funkcją pracy wykonywanej przez statek nad otaczającą cieczą, co powoduje powstawanie fal. Dowodem występowania tego zjawiska jest duża fala powstająca przy dziobie przecinającym wodę oraz fale, które statek pozostawia za rufą. Te fale wpływają na rozkład ciśnień wokół statku, a zatem również na opór ciśnieniowy i z tego powodu trudno w obliczeniach oddzielić składowe pochodzące od obu typów oporu.

⁸ *Międzynarodowa Konferencja Basenów Holowniczych* — dobrowolne światowe stowarzyszenie organizacji normalizacyjnych i odpowiedzialnych za rezultaty obliczeń hydrodynamicznych dotyczących statków i różnego rodzaju instalacji morskich. Członkiem tego stowarzyszenia jest np. polskie *Centrum Techniki Okrętowej* — *przyp. tłum.*

Przy wykonywaniu testów modelowych opór ciśnieniowy i opór falowy zwykle są traktowane jako jeden wspólny opór resztkowy. Analogicznie do współczynnika oporu tarcia możemy wprowadzić współczynnik oporu resztkowego:

$$R_r = R_{\text{ciśn}} + R_{\text{fal}} = (1/2)\rho V^2 S C_r$$

gdzie R_r jest całkowitym oporem resztkowym, a C_r — współczynnikiem oporu resztkowego.

Istnieje wiele metod obliczeniowych pozwalających na oszacowanie współczynnika oporu resztkowego statków, ale zwykle są one przeznaczone do obliczeń dla statków specyficznego typu. Na przykład jedną metodą z zastosowaniem określonych wzorów empirycznych oblicza się C_r dla niszczycieli, zaś dla wielkich tankowców stosowana jest inna metoda i inne wzory empiryczne. Problem polega na wybraniu metody odpowiedniej dla statku, którego ruch analizujemy⁹. Ogólnie możemy powiedzieć, że C_r wzrasta wraz z wypornością i szybkością. Typowy zakres wartości C_r dla statków o dużej wyporności wynosi od $1,0 \cdot 10^{-3}$ do $3,0 \cdot 10^{-3}$.

Choć te trzy składowe oporu, spowodowane przez tarcie, ciśnienie i fale, w przypadku statków o typowej wyporności są najważniejsze, to jednak nie są jedyne. Ponieważ statki poruszają się na granicy wody i powietrza, znaczna ich część wystaje ponad wodę i jest wystawiona na działanie powietrza. Dla statku oznacza to dodatkowy opór, który możemy w przybliżeniu obliczyć za pomocą wzoru:

$$R_{\text{pow}} = (1/2)\rho V^2 A_p C_{\text{pow}}$$

gdzie C_{pow} jest współczynnikiem oporu, a ρ — gęstością powietrza, V — szybkością statku, a A_p — powierzchnią przekroju poprzecznego nadwodnej części statku. Typowe wartości C_{pow} mieszczą się w przedziale od 0,6 do 1,1, zależnie od typu statku. Dla tankowców i wielkich statków towarowych wartości C_{pow} są bliskie górnej granicy, zaś dla okrętów wojennych zbliżają się do dolnej granicy. Jeżeli brakuje informacji pozwalających na dokładne obliczenie przekroju poprzecznego, można go oszacować za pomocą wzoru:

$$A_p = B^2/2$$

gdzie B jest szerokością statku.

Na statek działają jeszcze inne siły oporu, zależne od wieku statku, stanu morza i rodzaju pełnionej służby. Na przykład, jeżeli statek długo pływa po wodach morskich bez oczyszczania kadłuba, nastąpi narośnięcie na nim grubej warstwy porostów morskich, które spowodują zwiększenie siły oporu powodowanej przez tarcie. Jeżeli statek pływa na wodach płytkich lub np. w kanałach o ograniczonych rozmiarach, ograniczenie przepływu wokół niego może spowodować głębsze zanurzenie i w efekcie większy opór. Trudne warunki morskie, silne wiatry i duże fale również zwiększają opór działający na statek.

⁹ Metody te są bardzo zaawansowane i nie jest możliwe omówienie ich tutaj, więc w bibliografii zostały podane tytuły odpowiednich prac.

W przypadku niektórych statków liczne urządzenia wystające w wodzie zwiększają opór od 10% do 15% w porównaniu do oporów gładkiego kadłuba. Wszystko to są bardzo specyficzne warunki, które trzeba rozważać w poszczególnych przypadkach.

Masa wirtualna

Pojęcie *masy wirtualnej* jest ważne przy obliczaniu przyspieszenia statku w symulatorze działającym w czasie rzeczywistym. Masa wirtualna jest równa sumie masy statku i masy wody przyspieszanej wraz ze statkiem.

W rozdziale 6. mówiliśmy o powstającej z powodu lepkości granicznej warstwie wody przywierającej do powierzchni ciała poruszającego się w cieczy. Prędkość względna wody (względem poruszającego się ciała) był równa zero na powierzchni ciała i wzrastała wraz ze zwiększaniem się odległości od tej powierzchni aż do osiągnięcia prędkości strumienia swobodnego. Część cieczy przyklejała się do ciała, przesuwała się i przyspieszała wraz z nim. Skoro szybkość cieczy zmienia się w obrębie warstwy granicznej, zmienia się także przyspieszenie. *Masa dodana*, czyli masa wody przyspieszanej jest wynikiem scałkowania ważonego całej masy cieczy, na którą ma wpływ przyspieszanie ciała.

Dla statku lepka warstwa graniczna może być całkiem gruba i w pobliżu końca kadłuba (zależnie od jego długości) sięgać nawet kilku stóp¹⁰, co powoduje, że masa dodana ma istotną wielkość. Dlatego musi być brana pod uwagę przy wykonywaniu wszelkiego rodzaju analiz dotyczących przyspieszenia. Jednakże obliczanie masy dodanej wykracza poza zakres tej książki. Warto jednak zaznaczyć, że w przeciwieństwie do zwykłej masy masa dodana jest tensorem, czyli zależy od kierunku przyspieszenia. Ponadto masa dodana ma znaczenie zarówno dla przyspieszenia liniowego, jak i kąтового.

Zwykle uwzględniamy ją za pomocą tak zwanego współczynnika masy dodanej, który jest równy stosunkowi masy dodanej do masy statku. Choć nie zajmujemy się tu choćby przybliżonym obliczaniem masy dodanej, zaznaczymy, że w niektórych metodach całkowanie przeprowadza się po całej powierzchni kadłuba, zaś w innych przybliża się kadłub przez elipsoidę o proporcjach dopasowanych do rozmiarów statku.

Przy tym przybliżaniu długość elipsoidy jest zbliżona do długości, a szerokość do szerokości statku. Dla ruchu wzdłużnego, tzn. ruchu postępowego o kierunku wyznaczonym przez oś równoległą do kadłuba, współczynnik masy dodanej zmienia się niemal liniowo od zera przy stosunku szerokości do długości równym zero (dla statku nieskończenie cienkiego) do niemal 0,5 przy stosunku szerokości do długości równym 1 (dla kuli).

Jeżeli współczynnik masy dodanej jest wyrażony w procentach masy statku, masę wirtualną można obliczyć ze wzoru $m_w = m(1 + x_d)$, gdzie m jest masą statku, a x_d — współczynnikiem masy dodanej po przeliczeniu procentów na liczbę, np. 0,2 dla 20%. Dla statków o typowej wyporności wartość wzdłużnej masy dodanej waha się od 4% do 15% masy statku. W ostrożnych obliczeniach przyjmuje się zwykle 20%.

¹⁰1 stopa = 0,305 m — *przyj. tłum.*