

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 2159

**VIZUALIZACIJA SIMULACIJE KRETANJA PLOVILA PO
VODI**

Nikola Nađ

Zagreb, lipanj 2020.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 2159

**VIZUALIZACIJA SIMULACIJE KRETANJA PLOVILA PO
VODI**

Nikola Nađ

Zagreb, lipanj 2020.

DIPLOMSKI ZADATAK br. 2159

Pristupnik: **Nikola Nađ (0036491265)**
Studij: Računarstvo
Profil: Računarska znanost
Mentor: prof. dr. sc. Željka Mihajlović

Zadatak: **Vizualizacija simulacije kretanja plovila po vodi**

Opis zadatka:

Proučiti načine simulacije širenja valova na vodenoj površini. Razraditi model plovila i fizikalno temeljen utjecaj interakcije s vodenom površinom. Razraditi i ostvariti simulacijski model kretanja plovila uz prigodno ostvarivanje interakcije plovila i površine vode. Ostvariti trodimenzionalni prikaz dobivene simulacije. Diskutirati utjecaj različitih parametara. Načiniti ocjenu rezultata i implementiranih algoritama. Izraditi odgovarajući programski proizvod. Koristiti programsko razvojno okruženje Unity 3D te programski jezik C# i razvojno programsko okruženje VisualStudio. Rezultate rada načiniti dostupne putem Interneta. Radu priložiti algoritme, izvorne kodove i rezultate uz potrebna objašnjenja i dokumentaciju. Citirati korištenu literaturu i navesti dobivenu pomoć.

Rok za predaju rada: 30. lipnja 2020.

Sadržaj

Uvod	1
1. Ambijentalni valovi	2
2. Utjecaj vode na brod	4
2.1. Sila uzgona	4
2.2. Sila otpora.....	5
2.3. Otpor viskoznosti	6
3. Utjecaj broda na vodu	7
3.1. Valne čestice	7
3.2. Oblikovanje valne fronte.....	10
3.2.1. Dijeljenje valnih čestica	10
4. Implementacija.....	13
4.1. Određivanje uronjene površine	13
4.2. Računanje sila	17
4.2.1. Sila uzgona	18
4.2.2. Viskozni otpor	18
4.2.3. Sila otpora tlaka.....	19
4.2.4. Udarajuća sila.....	20
4.2.5. Motorski pogon	22
4.3. Implementacija valnih čestica.....	22
4.3.1. Sustav valnih čestica	22
4.3.2. Generator valnih čestica	23
4.3.3. Generiranje mape pomicanja.....	27
5. Rezultati.....	29
6. Zaključak	31
Literatura	32
Sažetak.....	33
Abstract	33

Uvod

Simulacija i prikaz vodenih površina interesantno je područje računalne grafike koje se još uvijek istražuje. Ovisno o primjeni, simulacije se mogu izvoditi u stvarnom vremenu (engl. *real-time, online*) ili se simulacija vode može najprije izračunati te kasnije prikazivati (engl. *offline*). Oba načina imaju svoje prednosti i mane – simulacija u stvarnom vremenu može reagirati na korisnikove ulaze, međutim zahtjev za izvođenjem u stvarnom vremenu ograničava složenost simulacije. S druge strane, *offline* simulacije mogu biti znatno složenije s obzirom na to da izvođenje nije ograničeno vremenskim rokom, ali ih više nije moguće mijenjati nakon izvođenja simulacije, već je potrebno cijelu simulaciju pokrenuti iz početka. Simulacije u stvarnom vremenu imaju svoju primjenu primjerice u interaktivnim simulacijskim sustavima i video igrama, dok se *offline* simulacije koriste za prikaze gdje interaktivnost ne igra ulogu, već je bitan što realističniji prikaz vode. Ove simulacije se koriste primjerice u filmovima.

U ovom radu opisana je i implementirana simulacija vode u stvarnom vremenu koja podržava interakciju plovila s vodom. U prvom poglavlju ukratko je opisana simulacija ambijentalnih valova. Drugo poglavlje opisuje sile koje djeluju na brod uslijed interakcije s vodom, a treće poglavlje opisuje tehniku simulacije interaktivnih valova sustavom valnih čestica. U četvrtom poglavlju opisana je implementacija sustava, dok je u petom poglavlju dan pregled rezultata implementiranog sustava.

1. Ambijentalni valovi

Ambijentalni valovi su, za razliku od interaktivnih valova, valovi koji se prirodno javljaju na vodenoj površini, a najčešće su uzrokovani vjetrom. Jedna od tehnika simulacije ambijentalnih valova je uporaba inverzne brze Fourierove transformacije (engl. *Inverse Fast Fourier Transform*, IFFT) u dvije dimenzije nad spektrom valova, a u ovom radu korišten je Phillipsov spektar, statistički temeljen model dobiven empirijskim oceanografskim istraživanjem. U nastavku poglavlja dan je kratak pregled formula korištenih u implementaciji ambijentalnih valova.

Phillipsov spektar definiran je kao

$$P_h(\mathbf{k}) = A \frac{\exp\left(\frac{-1}{(kl)^2}\right)}{k^4} |\hat{\mathbf{k}} \cdot \hat{\mathbf{w}}|^2 \quad (1.1)$$

gdje je \mathbf{k} smjer propagacije vala, \mathbf{w} brzina vjetra, $l = |\mathbf{w}|^2/g$ najveći mogući val kao rezultat vjetra konstantne brzine $|\mathbf{w}|$, a A je konstanta kojom se regulira amplituda valova.

Polazna ideja simulacije ambijentalnih valova je definirati visinu vodene površine $h(\mathbf{x}, t)$ u točki $\mathbf{x} = (x, z)$ i trenutku t kao sumu sinusoida s kompleksnom, vremenski ovisnom amplitudom $\tilde{h}(\mathbf{k}, t)$:

$$h(\mathbf{x}, t) = \sum_{\mathbf{k}} \tilde{h}(\mathbf{k}, t) e^{i\mathbf{k} \cdot \mathbf{x}} \quad (1.2)$$

gdje je \mathbf{k} dvodimenzionalni vektor koji određuje smjer propagacije vala, definiran kao:

$$\mathbf{k} = \left(\frac{2\pi n}{L}, \frac{2\pi m}{L} \right) \quad (1.3)$$

pri čemu su m i n iz intervala $[-N/2, N/2]$. Ovime se dobiva visinska mapa (engl. *heightmap*) dimenzija $N \times N$ piksela (N je broj uzoraka Fourierove transformacije po jednoj dimenziji) koja odgovara vodenoj površini dimenzija $L \times L$, tako da su dva susjedna uzorka udaljena za N/L .

Polazne Fourierove amplitude računaju se prema izrazu:

$$\tilde{h}_0(\mathbf{k}) = \frac{1}{\sqrt{2}} (\xi_r + i\xi_i) \sqrt{P_h(\mathbf{k})} \quad (1.4)$$

gdje su ξ_r i ξ_i slučajne varijable sa standardnom normalnom razdiobom. Na temelju polaznih amplituda i relacije disperzije $\omega(k)$ dobivaju se Fourierove amplitude u trenutku t :

$$\tilde{h}(\mathbf{k}, t) = \tilde{h}_0(\mathbf{k})e^{i\omega(k)t} + \tilde{h}_0^*(-\mathbf{k})e^{-i\omega(k)t}. \quad (1.5)$$

Relacija disperzije predstavlja vezu između norme valnog vektora \mathbf{k} (engl. *wavevector*) i valne frekvencije, a za duboku vodu ona glasi $\omega(k) = \sqrt{gk}$.

Polazne Fourierove amplitude potrebno je izračunati jednom, a na temelju tih polaznih amplituda se u proizvoljnom trenutku t određuju amplitude koje se koriste u inverznoj brznoj Fourierovoj transformaciji u dvije dimenzije kako bi se dobila visinska mapa za taj trenutak. Dobivena visinska mapa je periodička, pa je moguće dobiti vodenu površinu proizvoljnih dimenzija, međutim periodičnost se može uočiti ako se vodena površina pogleda iz dovoljno velike daljine. Ambijentalni valovi generirani opisanom tehnikom prikazani su na slici (Slika 1.1).



Slika 1.1 Ambijentalni valovi generirani IFFT tehnikom

2. Utjecaj vode na brod

Sile koje utječu na brod koji se nalazi u vodi mogu se podijeliti na hidrostatičke i hidrodinamičke. Statičke sile uvijek su prisutne, dok su dinamičke sile uzrokovane kretanjem plovila relativno u odnosu na vodu. Sile koje se javljaju interakcijom plovila i fluida vrlo su komplicirane u stvarnosti te mnoge nisu još analitički određene (kao na primjer viskozni otpor opisan u nastavku rada) već su dobivene regresijom rezultata empirijskih istraživanja.

2.1. Sila uzgona

Hidrostatička sila ili sila uzgona je sila koja djeluje na objekt djelomično ili u potpunosti uronjen u fluid. Sila uzgona prouzročena je djelovanjem hidrostatičkog tlaka

$$p_h = \rho gh \quad (2.1)$$

na površinu

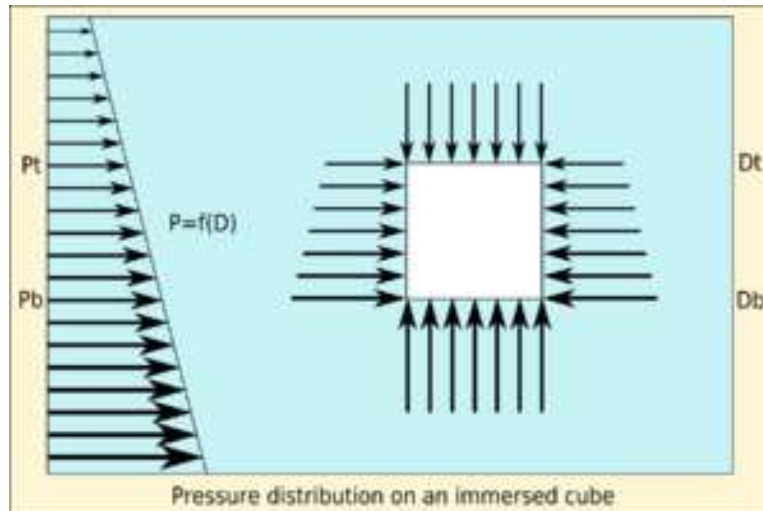
$$d\vec{F}_U = -p_h \vec{n} dS \quad (2.2)$$

gdje je \vec{n} vanjska normala na površinu. Minus se javlja zbog orijentacije sile, koja djeluje prema unutrašnjosti objekta. Tlak djeluje na objekt sa svih strana, pri čemu se sile uzrokovane tim tlakom poništavaju na horizontalnim stranama objekta. Na donju stranu objekta djeluje veći tlak u odnosu na gornju stranu (jer se nalazi na većoj dubini) te je ta razlika uzrok sile uzgona. Slika 2.1 ilustrira promjenu tlaka s dubinom.

Ukupna sila uzgona dobije se integriranjem po površini uronjenog dijela objekta

$$F_U = \oint \rho gh dS \quad (2.3)$$

gdje je ρ gustoća fluida, g ubrzanje sile teže, a h visina stupca fluida.



Slika 2.1 Ilustracija promjene tlaka s dubinom na primjeru kocke

2.2. Sila otpora

Sila otpora je hidrodinamička sila i rezultat je relativnog kretanja objekta u fluidu, a uzrokuje ju tlak fluida, kao i trenje između fluida i objekta. Jednadžba otpora glasi

$$F_D = \frac{1}{2} \rho C_D S u^2 \quad (2.4)$$

gdje je ρ gustoća fluida, C_D koeficijent otpora, u brzina fluida u odnosu na objekt, a S efektivna površina objekta, koja se najčešće definira kao ortografska projekcija objekta na ravninu okomitu na smjer kretanja. Sila otpora djeluje u smjeru gibanja fluida, odnosno u suprotnom smjeru od gibanja objekta. Koeficijent otpora C_D značajno ovisi o obliku objekta i nije konstantan, već ovisi i o Reynoldsovom broju, koji pak ovisi o relativnoj brzini kretanja fluida u odnosu na objekt. S obzirom na to da kretanje fluida u blizini objekta može biti komplicirano, ne postoji analitičko rješenje za određivanje tog koeficijenta, već se određuje mjerenjima.

2.3. Otpor viskoznosti

Otpor viskoznosti (engl. *Skin friction drag*) je sila otpora koja se javlja zbog viskoznosti fluida u kojem se objekt kreće. Sloj fluida se „lijepi“ za objekt kako se on kreće, zatim se drugi sloj fluida „zalijepi“ za prvi sloj i tako dalje te na taj način objekt svojom površinom povlači dio fluida u smjeru gibanja, što uzrokuje otpor. Ova sila se računa po jednadžbi otpora, gdje je koeficijent otpora C_f definiran kao

$$C_f = \frac{\tau_w}{\frac{1}{2}\rho v^2} . \quad (2.5)$$

Ovdje je τ_w smično naprezanje. Ukupan otpor može se izračunati integriranjem smičnog stresa po površini objekta u fluidu

$$F_D = \int \tau_w dS = \int C_f \frac{\rho v^2}{2} dS . \quad (2.6)$$

C_f se određuje empirijskim formulama dobivenim eksperimentima, a 1957. godine je ITTC odredio formulu

$$C_f = \frac{0.075}{(\log_{10} R_n - 2)^2} \quad (2.7)$$

kao dobru aproksimaciju prikupljenih podataka. Ovdje je R_n Reynoldsov broj, koji se definira kao

$$R_n = \frac{uL}{\nu} \quad (2.8)$$

gdje je u brzina fluida, L duljina površine objekta po kojoj se fluid kreće, a ν kinematička viskoznost fluida.

3. Utjecaj broda na vodu

Rezultat interakcije broda i vode su interaktivni valovi. Za razliku od ambijentalnih valova, ovi valovi ne mogu se unaprijed izračunati s obzirom na to da njihovo generiranje ovisi o kretanju broda. Interaktivni valovi modelirani su metodom valnih čestica (engl. Wave Particles) [5].

3.1. Valne čestice

Svaka valna čestica predstavlja lokalnu devijaciju vodene površine na svojoj poziciji. Pritom se čestice u sustavu kreću isključivo po horizontalnoj ravnini. Valne čestice su međusobno nezavisne – ne postoji interakcija među njima te nisu „svjesne“ drugih susjednih čestica, što omogućava implementaciju linearne složenosti. Svaka valna čestica ima nekoliko svojstava kojima je definirana, a to su (1) točka podrijetla (engl. origin point) \mathbf{O} , (2) točka rođenja (engl. birth point) \mathbf{B} , (3) vrijeme rođenja t_b (4) amplituda a , (5) orijentacija d , (6) radijus r te (7) brzinu v . Uloga ovih svojstava opisana je u nastavku poglavlja.

Lokalna vertikalna devijacija čestice predstavljena je funkcijom D , dok je globalna vertikalna devijacija vodene površine na danoj poziciji $\mathbf{x} = (x, y)$ u trenutku t jednaka superpoziciji lokalnih devijacija svih čestica:

$$\delta_z = \sum_i D_i(\mathbf{x}, t) . \quad (3.1)$$

Funkcija lokalne devijacije čestice može se formulirati kao

$$D_i(\mathbf{x}, t) = a_i W(\mathbf{x} - \mathbf{x}_i(t)) \quad (3.2)$$

gdje je a_i amplituda čestice, $\mathbf{x}_i(t)$ pozicija čestice u trenutku t u horizontalnoj ravnini, a W konstantna funkcija kojom je određen oblik svih valnih čestica. Pritom se W uzima tako da je $W > 0$ na ograničenom dijelu domene u relativnoj blizini ishodišta, dok je 0 za ostatak domene kako bi se osigurao lokalni utjecaj čestice.

U dvodimenzionalnom prostoru dobar odabir za funkciju W bila bi sinusoida, s obzirom na to da po obliku odgovara većini valova koji nastaju na vodenoj površini:

$$W(x) = \frac{1}{2} \left(\cos\left(\frac{2\pi x}{\lambda}\right) + 1 \right) \Pi\left(\frac{x}{\lambda}\right) \quad (3.3)$$

gdje je Π jedinična pravokutna funkcija u rasponu $[-1/2, 1/2]$, a λ valna duljina čestice. Ova funkcija je, osim zbog oblika, povoljna i zbog jednostavnosti generiranja kontinuiranih valova fiksne valne duljine uzastopnim nizanjem valnih čestica s pozitivnom i negativnom amplitudom. Također, ova funkcija ima C^1 kontinuitet, pa će i globalna devijacija također imati isti kontinuitet.

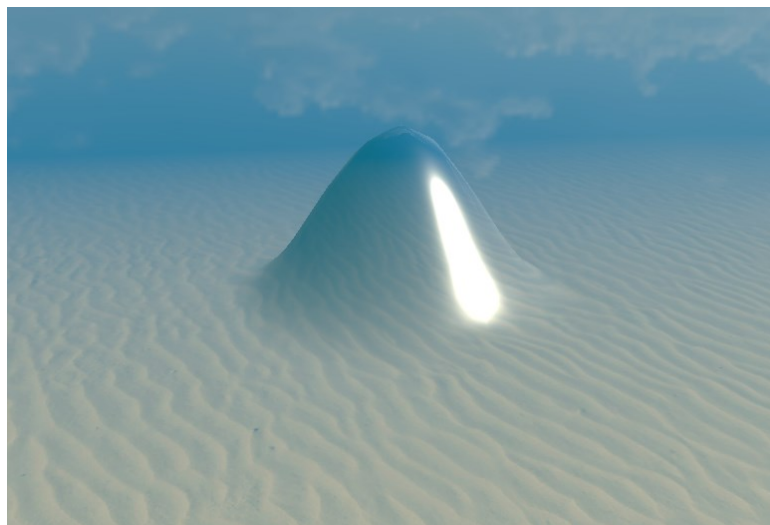
Proširenjem gornje funkcije na trodimenzionalni prostor dobiva se funkcija:

$$W(\mathbf{x}) = \frac{1}{2} \left(\cos\left(\frac{\pi|\mathbf{x}|}{r}\right) + 1 \right) \Pi\left(\frac{|\mathbf{x}|}{2r}\right) \quad (3.4)$$

gdje je r radijus valne čestice. Konačno se za vertikalnu devijaciju čestice može pisati

$$D_i(\mathbf{x}, t) = \frac{a_i}{2} \left(\cos\left(\frac{\pi|\mathbf{x} - \mathbf{x}_i(t)|}{r}\right) + 1 \right) \Pi\left(\frac{|\mathbf{x} - \mathbf{x}_i(t)|}{2r}\right) \quad (3.5)$$

Slika 3.1 prikazuje oblik jedne valne čestice s vertikalnom devijacijom.



Slika 3.1 Oblik valne čestice

Međutim, stvarni valovi imaju pomalo drugačiji oblik – nisu savršeno glatki poput sinusoide, već su im vrhovi zašiljeni kao što je istaknuto na slici (Slika 3.2).



Slika 3.2 Šiljasti oblik valova na moru

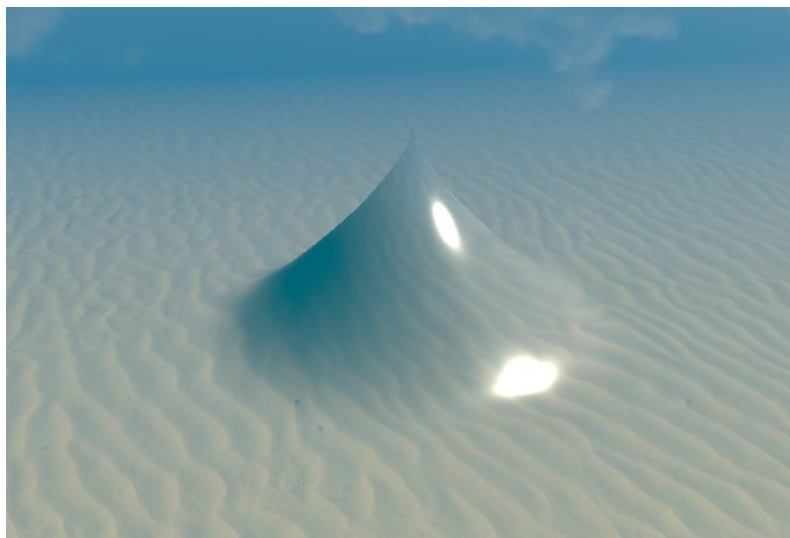
Da bi se modelirao takav oblik vala, osim vertikalne devijacije uvodi se i devijacija u horizontalnoj ravnini. Ova devijacija definira se kao

$$\mathbf{H}_i(\mathbf{x}, t) = -\frac{a_i\sqrt{2}}{2} \sin\left(\frac{\pi|\mathbf{p}|}{r}\right) \left(\cos\left(\frac{\pi|\mathbf{p}|}{r}\right) + 1\right) \Pi\left(\frac{|\mathbf{p}|}{2r}\right) \hat{\mathbf{p}} \quad (3.6)$$

gdje je $\mathbf{p} = \mathbf{x} - \mathbf{x}_i(t)$ vektor od pozicije čestice u trenutku t do dane pozicije \mathbf{x} . Sada ukupna devijacija čestice postaje trodimenzionalni vektor, čija je vertikalna komponenta D_i , dok su horizontalne komponente devijacije komponente vektora \mathbf{H}_i . Globalna devijacija također postaje trodimenzionalni vektor te se može zapisati po komponentama kao

$$\boldsymbol{\delta} = \sum_i (H_{ix}, H_{iy}, D_i) . \quad (3.7)$$

Konačni oblik valne čestice s vertikalnom i horizontalnom devijacijom prikazan je na slici (Slika 3.3).



Slika 3.3 Oblik valne čestice s horizontalnom devijacijom

3.2. Oblikovanje valne fronte

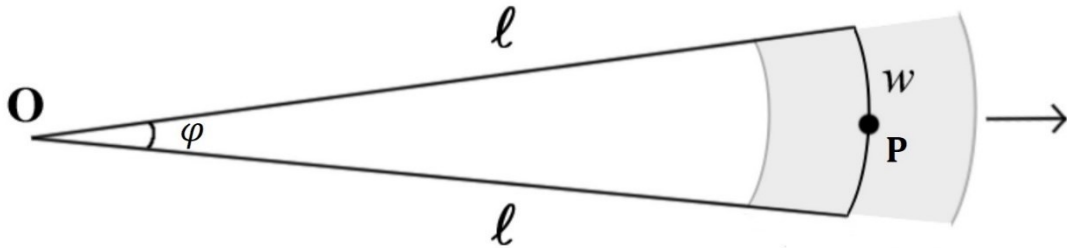
Valna fronta se u ovom prikazu modelira nizanjem čestica jednu pored druge, pri čemu je važno pripaziti da su valne čestice dovoljno blizu jedna drugoj kako bi valna fronta bila kontinuirana. Rezultat ovog načina modeliranja je zapravo aproksimacija valne fronte, međutim greška nastala aproksimacijom je gotovo neprimjetna ako se čestice nalaze na udaljenosti manjoj od pola radijusa.

S obzirom na to da su valne fronte uglavnom zakrivljene, odnosno proširujuće, valne čestice se propagacijom sve više udaljavaju jedne od drugih te nakon nekog vremena valna fronta gubi kontinuitet. Da bi valna fronta ostala kontinuirana, potrebno je podijeliti valne čestice na veći broj manjih čestica kada udaljenost između njih postane prevelika.

3.2.1. Dijeljenje valnih čestica

Točno određivanje udaljenosti između susjednih čestica narušilo bi njihovu nezavisnost i povećalo bi složenost izvođenja. Kako bi se to izbjeglo, uvodi se novo svojstvo čestice – kut disperzije φ . Kut disperzije predstavlja kut pod kojim se širi dio valne fronte predstavljen česticom. Također, uvodi se i točka podrijetla čestice (engl. origin point) te se na temelju te točke, kuta disperzije i

trenutne pozicije čestice može odrediti širina w dijela valne fronte koju čestica predstavlja. Odnos ovih svojstava prikazan je na slici (Slika 3.4).



Slika 3.4 Odnos kuta disperzije (φ), točke podrijetla (**O**) i širine dijela valne fronte (w)

Nadalje, uvodi se pretpostavka da susjedne čestice koje pripadaju istoj valnoj fronti te da imaju istu točku podrijetla i kut disperzije – tada se udaljenost između čestica d može aproksimirati kao duljina luka, odnosno širina čestice w kad je kut disperzije mali. Tada vrijedi

$$\varphi \approx \frac{w}{l} \approx \frac{d}{l} \quad (3.8)$$

Ako je za trenutak t_0 poznata udaljenost čestica d_0 , a u proizvoljnom trenutku t ta udaljenost označena kao d_t , vrijedi

$$\varphi \approx \frac{d_0}{l_0} = \frac{d_t}{l_t} = \frac{d_t}{l_0 + v(t - t_0)} \quad (3.9)$$

iz čega se može izraziti d_t

$$d_t \approx d_0 + \varphi v(t - t_0) \quad (3.10)$$

Ovdje je v brzina propagacije čestice. Sada je lako odrediti trenutak kada je potrebno dijeliti česticu na manje dijelove – u trenutku t kada d_t postane veći od pola radijusa čestice. Ovaj trenutak moguće je odrediti već pri stvaranju same čestice tako da se za t_0 uzme vrijeme rođenja čestice, a za l_0 udaljenost između točke rođenja i točke podrijetla.

Kod dijeljenja čestica, svaka čestica podijeli se na tri manje čestice, svaka s trećinom amplitude originalne čestice kako bi se očuvao volumen vala.

Također, svaka čestica nakon dijeljenja ima trećinu originalne disperzije kuta jer sada predstavlja trećinu širine dijela valne fronte koju je predstavljala originalna čestica, dok je točka podrijetla istovjetna originalnoj. Da bi se odredila točka rođenja novonastalih čestica, potrebno je rotirati poziciju originalne čestice u trenutku dijeljenja za $\varphi/2$ oko točke podrijetla u oba smjera, dok treća čestica ostaje na istoj poziciji kao originalna čestica. Smjer propagacije čestice određuje se kao vektor dobiven oduzimanjem točke podrijetla od točke rođenja.

4. Implementacija

Simulacija je implementirana u razvojnoj okolini Unity, skripte su pisane u programskom jeziku C#, računski sjenčari (engl. compute shader) pisani su u jeziku HLSL (engl. High-Level Shading Language), dok je sjenčar za prikaz vode razvijen u *Shader Graph*-u, Unity-jevom vizualnom alatu za izradu sjenčara.

Najprije je opisan postupak kojim se određuje dio površine broda koji je uronjen u vodu. Zatim je opisano računanje hidrostatskih i hidrodinamičkih sila koje djeluju na uronjeni dio broda. Na posljertku je opisana implementacija interaktivnih valova pomoću valnih čestica.

4.1. Određivanje uronjene površine

Da bi se odredila uronjena površina objekta definiranog triangularnom mrežom, potrebno je iterirati po svim trokutima objekta te odrediti nalazi li se trokut u vodi. Ako se trokut ne nalazi u vodi, odbacuje se, a ako se u potpunosti nalazi u vodi, onda se cijeli trokut pridodaje uronjenoj površini. Ako je trokut samo djelomično uronjen u vodu, tada je potrebno odrediti presjek trokuta s vodenom površinom. Za ispitivanje uronjenosti trokuta koristi se visinska mapa (engl. height map) površine vode.

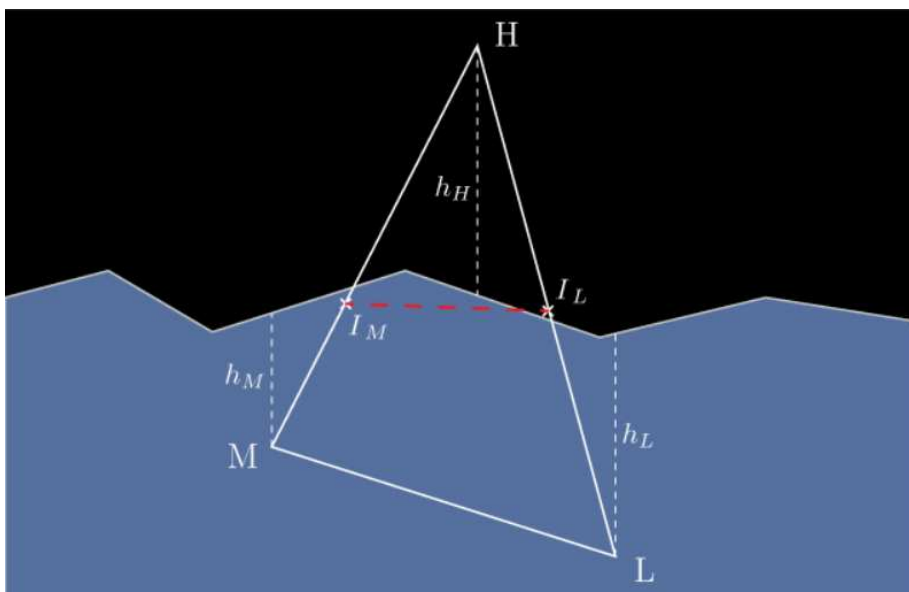
Određivanje presjeka trokuta s vodenom površinom relativno je složen problem, s obzirom na to da postoji velik broj načina na koji trokut može presijecati površinu, stoga je algoritam pojednostavljen kako bi se poboljšale performanse. Performanse su ovdje bitne jer je algoritam potrebno izvoditi svaki okvir (engl. frame), s obzirom na to da se i vodena površina i objekt mijenjaju, odnosno pomiču u svakom okviru. Zbog pojednostavljenja algoritma, dobiveni rezultat je samo aproksimacija pravoga presjeka, međutim pogreška prouzrokovana aproksimacijom u praksi ne dolazi do izražaja ako trokuti mreže objekta nisu preveliki u odnosu na amplitudu i valnu duljinu valova. Utjecaj greške smanjuje se povećanjem broja trokuta u mreži. Prvo

korišteno pojednostavljenje je da se trokut smatra u potpunosti uronjenim ako su sva tri njegova vrha ispod površine vode, odnosno smatra se u potpunosti izvan vode ukoliko su mu sva tri vrha iznad površine vode. Primjer u kojem ova aproksimacija daje pogrešan rezultat je slučaj kada vodena površina probija trokut u njegovom središtu, međutim sva tri vrha i dalje ostaju iznad površine. Ovaj slučaj prikazan je na slici (Slika 4.1).



Slika 4.1 Primjer slučaja u kojem algoritam daje pogrešan rezultat – središnji dio trokuta je uronjen, međutim algoritam ovaj trokut smatra neuronjenim

Nadalje, slučaj kada je trokut djelomično uronjen u vodu detektira se kada su jedan ili dva vrha trokuta ispod razine vode. Za svaki vrh trokuta odredi se udaljenost od vodene površine, pri čemu pozitivna vrijednost znači da je vrh iznad vode, dok negativna vrijednost predstavlja uronjeni vrh. Vrhovi trokuta sortiraju se po toj udaljenosti te su imenovani H , M i L kao najviši (engl. *highest*), srednji (engl. *middle*) i najniži (engl. *lowest*), a pripadajuće udaljenosti označene su s h_H , h_M i h_L . Slučaj kada su dva vrha uronjena prepoznaje se kada su ispunjeni uvjeti $h_M, h_L \leq 0, h_H > 0$. Ovaj slučaj prikazan je na slici (Slika 4.2).



Slika 4.2 Slučaj kad je jedan vrh trokuta iznad površine vode, a dva ispod (preuzeto iz [3])

Ovdje se koriste pretpostavke da vodena površina presijeca bridove \overline{HM} i \overline{HL} u samo jednoj točki te da je brid \overline{ML} u potpunosti uronjen. Točke presjeka na bridovima označene su s I_M i I_L , a moguće ih je izraziti u parametarskom obliku:

$$\overline{MI_M} = t_M \overline{MH} \quad (4.1)$$

i

$$\overline{LI_L} = t_L \overline{LH} \quad (4.2)$$

Parametri t_M i t_L aproksimiraju se na sljedeći način:

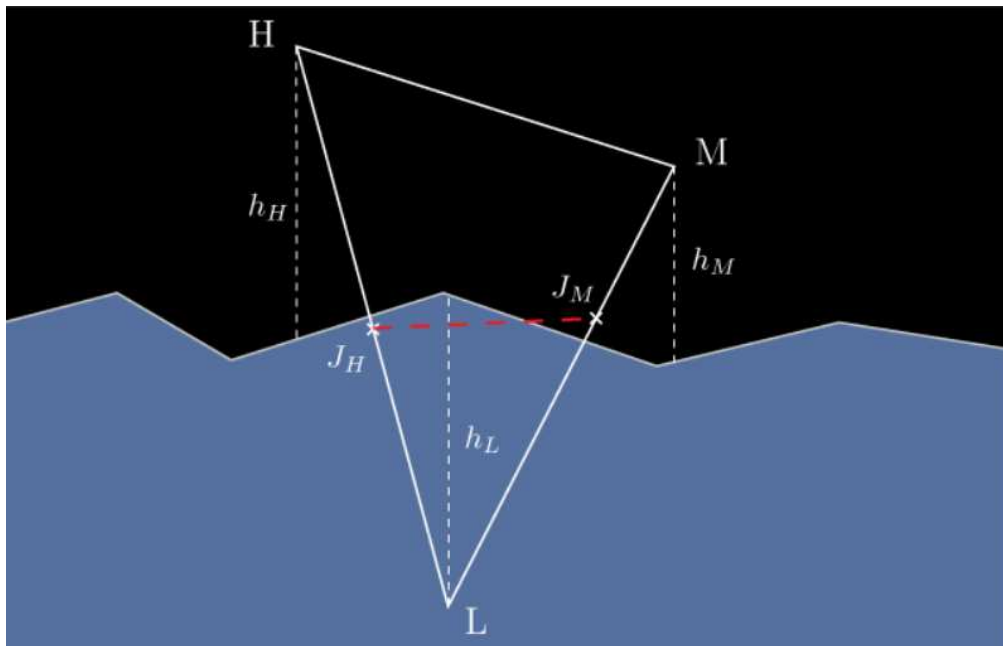
$$t_M = \frac{-h_M}{(h_H - h_L)} \quad (4.3)$$

i

$$t_L = \frac{-h_L}{(h_H - h_L)} \quad (4.4)$$

Uronjenim dijelom trokuta smatra se četverokut definiran vrhovima M , L , I_L i I_M , kojeg je još potrebno triangulirati, pa se u listu uronjenih trokuta dodaju trokuti M, L, I_M te L, I_L, I_M .

Slučaj kada je samo jedan vrh uronjen prepoznaje se kada su ispunjeni uvjeti $h_L \leq 0, h_M, h_H > 0$. Ovaj slučaj prikazan je na slici (Slika 4.3).



Slika 4.3 Slučaj kad su dva vrha trokuta iznad površine vode, a dva ispod (preuzeto iz [3])

Ovdje su korištene slične pretpostavke kao i u prethodnom slučaju, odnosno da bridovi \overline{HL} i \overline{LM} imaju po jedno sjecište te da je brid \overline{HM} u potpunosti izvan vode. Točke presjeka označene su s J_H i J_M te ih ponovno izražavamo parametarski:

$$\overrightarrow{LJ_H} = t_H \overrightarrow{LH} \quad (4.5)$$

i

$$\overrightarrow{LJ_M} = t_M \overrightarrow{LM} \quad (4.6)$$

pri čemu su parametri

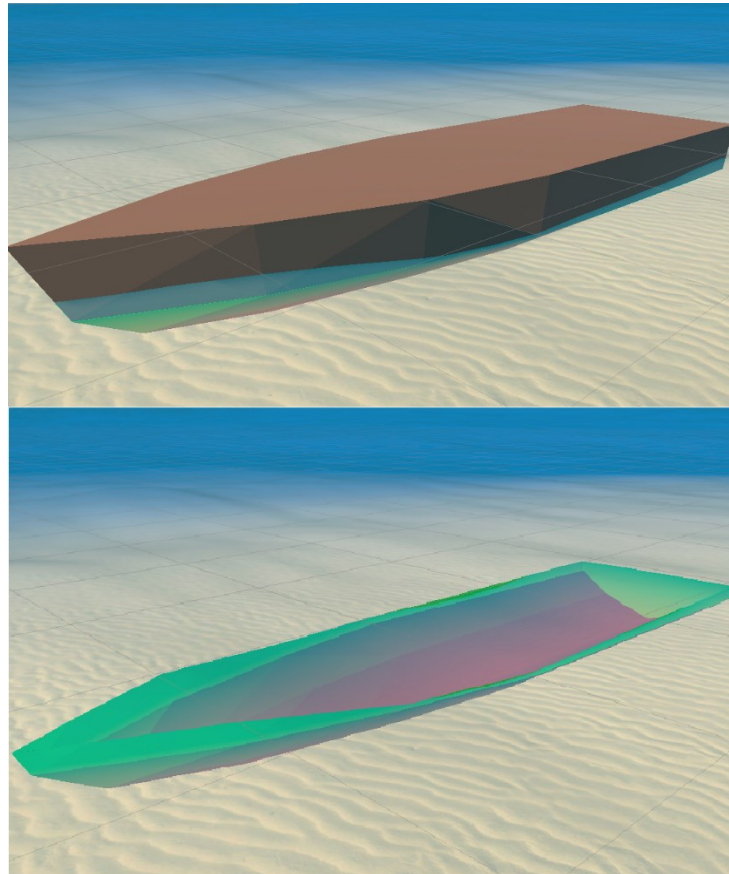
$$t_M = \frac{-h_L}{(h_M - h_L)} \quad (4.7)$$

i

$$t_H = \frac{-h_L}{(h_H - h_L)} \quad (4.8)$$

Dio polaznog trokuta koji se smatra uronjenim je trokut definiran vrhovima L , J_M , J_H te se taj trokut dodaje listi uronjenih trokuta.

Konačni rezultat algoritma je lista trokuta koji su uronjeni u vodu, a prikaz se može vidjeti na slici (Slika 4.4).



Slika 4.4 Prikaz broda u vodi (gore) te uronjenog dijela mreže tog broda (dolje)

Za svaki uronjeni trokut još se bilježi nekoliko podataka pri njihovom stvaranju, koji su potrebni za određivanje sila nad tim trokutima te ubrzavanje računanja tih sila. Tako se za svaki trokut u strukturu zapisuje (1) centar trokuta, (2) normala trokuta, (3) brzina trokuta, (4) površina trokuta te (5) indeks trokuta mreže broda od kojeg je nastao taj uronjeni trokut.

4.2. Računanje sila

Nakon što se odredi lista trokuta koji su pod vodom, nad njima se računaju četiri sile kojima se modelira utjecaj vode na brod: sila uzgona, viskozni otpor,

otpor tlaka te sila udaranja. Sila uzgona i viskozni otpor temeljeni su na pravim, fizikalno zasnovanim silama koje se javljaju u stvarnosti, dok su otpor tlaka i sila udaranja aproksimativne sile kojima se modelira ciljani aspekt ponašanja broda te služe kao zamjena za stvarne sile. Implementacija tih sila opisana je u nastavku.

4.2.1. Sila uzgona

Silu uzgona moguće je implementirati na dva načina, preko volumena ili preko površine. Metoda volumena svodi se na određivanje volumena uronjenog dijela objekta te traženje njegovog centroida. Sila uzgona zatim se odredi po Arhimedovom zakonu te se sila primijeni na objekt u pronađenom centroidu. U metodi površine se određuje uronjena površina objekta te se sila uzgona računa nad tom površinom. Obje metode su korektne te rezultiraju jednakom silom uzgona (ukoliko je implementacija egzaktna), a u implementaciji ovog rada korištena je metoda površine.

Sila uzgona računa se nad svakim trokutom iz liste uronjenih trokuta po formuli

$$\vec{F}_U = -\rho g h_c S n_y \hat{y} \quad (4.9)$$

gdje je h_c dubina na kojoj se nalazi centar trokuta, S površina trokuta, n_y y komponenta normale trokuta, a \hat{y} jedinični vektor u smjeru y osi (pokazuje prema gore). Ovdje je važno napomenuti da se sila uzgona računa samo na vertikalnoj osi, s obzirom na to da se horizontalne komponente pokrate. Također, rezultat ove formule nije egzaktna jer je uvedena pretpostavka da se cijeli trokut nalazi na istoj dubini h_c , što u većini slučajeva nije istina. Međutim, greška se smanjuje smanjenjem površine trokuta mreže (odnosno povećavanjem broja trokuta u mreži objekta) te nestaje u potpunosti kada površina teži nuli. Dobivena sila se primjenjuje na brod na poziciji koja odgovara centru uronjenog trokuta.

4.2.2. Viskozni otpor

Viskozni otpor računa se za svaki uronjeni trokut broda, po formuli

$$\vec{F}_D = \frac{1}{2} \rho C_f S v_t \vec{v}_t \quad (4.10)$$

gdje je C_f koeficijent viskoznog otpora izračunat po formuli (2.7), S površina trokuta, a \vec{v}_t relativna tangencijalna brzina toka fluida kod centra trokuta. S obzirom na to da se brzina broda mijenja svaki okvir, potrebno je nanovo računati Reynoldsov broj svaki okvir, a time i koeficijent viskoznog otpora jer on ovisi o Reynoldsovom broju. Reynoldsov broj računa se po formuli (2.8), gdje se duljina površine L po kojoj se fluid kreće aproksimira duljinom obujmice (engl. bounding box) uronjenog dijela broda po lokalnoj z osi (pretpostavka je da je model broda orijentiran u smjeru z osi). Tangencijalna brzina računa se kao

$$\vec{v}_t = v \hat{u}_f \quad (4.11)$$

gdje je v brzina centra trokuta, a \hat{u}_f jedinični vektor definiran kao

$$\hat{u}_f = - \frac{\vec{n} \times (\vec{v} \times \vec{n})}{|\vec{n} \times (\vec{v} \times \vec{n})|} \quad (4.12)$$

Utjecaj ove sile najbolje se može primijetiti pri vožnji broda ravno na mirnoj vodi. U tom slučaju je ova hidrodinamična sila dominantna, s obzirom na to da su otpor tlaka i udarajuća sila (opisani u nastavku) relativno mali.

4.2.3. Sila otpora tlaka

Sila otpora tlaka (engl. Pressure Drag Force) je aproksimativna sila, međutim uvedena je u model utjecaja vode na brod jer je jednostavna za izračunati, a dostatna je zamjena za stvarne, znatno kompliciranije sile otpora koje utječu na brod, kao što su otpor stvaranja valova (engl. wave-making resistance), otpor prelamajućih valova (engl. breaking wave resistance), otpor raspršenja (engl. spray resistance) [3]. Sila se računa za svaki uronjeni trokut broda, a računa se prema formuli

$$\vec{F}_D = -(C_{PD1} v + C_{PD2} v^2) S (\cos\theta)^{f_p} \vec{n} \quad (4.13)$$

ako je $\cos\theta$ pozitivan, odnosno

$$\vec{F}_D = (C_{PS1}v + C_{PS2}v^2)S(\cos\theta)^{f_s}\vec{n} \quad (4.14)$$

ako je $\cos\theta$ negativan. Pritom je S površina trokuta, θ kut između vektora brzine i normale trokuta, \vec{n} normala trokuta, a C_{PD1} , C_{PD2} , C_{PS1} i C_{PS2} konstante koje definira korisnik. Ako je kosinus kuta pozitivan, trokut se „zabija“ u vodu te otpor ima smjer suprotan od normale, a ako je kosinus negativan, tada sila djeluje u smjeru normale te se time modelira „usisni“ otpor zbog pada tlaka uslijed odmicanja broda. Sila ima linearni i kvadratni član, a pomoću konstanti moguće je kontrolirati iznos sile pri različitim brzinama, s obzirom na to da je pri manjim brzinama linearni član relevantan, dok pri velikim brzinama kvadratni član dominira.

Utjecaj ove sile najbolje se može se primijetiti pri skretanju jer ograničava zanošenje broda te pri većim brzinama, gdje modelira podizanje broda zbog hidrodinamičkog podizanja (engl. Hydrodynamic lift).

4.2.4. Udarajuća sila

Ova sila je također aproksimativna, a uvedena je u model kako bi se jednostavnije moglo kontrolirati i podesiti ponašanje broda pri naglom ulasku u vodu velikom brzinom. Naime, pri udaru u vodu velikom brzinom, potrebno bi bilo istisnuti veliki volumen vode u kratkom vremenu, a tome se protivi inercija vode te se javlja sila koja zaustavlja brod – udarajuća sila.

Da bi se ova sila mogla izračunati, potrebno je mjeriti podatke o uronjenosti trokuta mreže broda u vodu u trenutnom i u prethodnom okviru (engl. frame). Koristi se dvostruki spremnik (engl. *buffer*) $[A_j^s(t), A_j^s(t - dt)]$, s kapacitetom broja trokuta u mreži broda. Ovdje $A_j^s(t)$ predstavlja uronjeni dio površine trokuta j u trenutku t , a dt je vremenski korak između dva uzastopna okvira. U svakom okviru aktivan je jedan spremnik te se u njega upisuju vrijednosti izračunate za trenutni okvir, dok neaktivni spremnik sadrži vrijednosti prethodnoga okvira. U svakom okviru spremnici mijenjaju uloge, tako da spremnik koji je bio aktivan prošli okvir postaje neaktivan, a spremnik koji je prošli okvir bio neaktivan postaje aktivan te se u njega zapisuju vrijednosti za

trenutni okvir. Podaci o uronjenosti trokuta u vodu zapisuju se pri samom određivanju uronjenog dijela mreže, na način da svaki novotvoreni uronjeni trokut T_i^s pribroji svoju površinu u spremnik na indeksu originalnog trokuta mreže broda od kojeg uronjeni trokut potječe. Također, potrebno je mjeriti podatke i o brzini trokuta u trenutnom i prethodnom okviru te se za to također koristi dvostruki spremnik.

Podaci o uronjenosti i brzini trokuta koriste se za određivanje volumena vode koji taj trokut pomiče po sekundi:

$$dV_i(t) = A_i^s(t)v_i(t) \quad (4.15)$$

Razlikom tih volumena za trenutni i prošli okvir te dijeljenjem s ukupnom površinom trokuta S_i i vremenskim korakom dt dobiva se ekvivalent akceleraciji

$$\Gamma_i(t) = \frac{V_i(t) - V_i(t - dt)}{S_i dt} \quad (4.16)$$

Da bi se lakše shvatio značaj ove mjere, proučit ćemo dva slučaja: (1) ako je trokut u oba okvira u potpunosti uronjen, akceleracija će iznositi

$$\Gamma_i(t) = \frac{S_i v_i(t) - S_i v_i(t - dt)}{S_i dt} = \frac{v_i(t) - v_i(t - dt)}{dt} \quad (4.17)$$

što je akceleracija centra trokuta, te (2) ako je trokut u prethodnom okviru u potpunosti bio izvan vode, a u trenutnom je u potpunosti uronjen, tada je

$$\Gamma_i(t) = \frac{S_i v_i(t) - 0}{S_i dt} = \frac{v_i(t)}{dt} \quad (4.18)$$

što odgovara akceleraciji potrebnoj da bi se trokut u potpunosti zaustavio u jednom vremenskom koraku. Dakle, ako trokut naglo uranja u vodu, iznos ove akceleracije će biti velik, dok će iznos biti malen ako se uronjenost ne mijenja znatno. Ova akceleracija koristi se za određivanje u kojoj mjeri je potrebno primijeniti udarajuću silu prema sljedećem izrazu

$$\vec{F}_{slam} = \text{clamp}\left(\frac{\Gamma_i(t)}{\Gamma_{max}}, 0, 1\right)^p \cos \theta \vec{F}_{stop} \quad (4.19)$$

gdje su Γ_{max} i p parametri koje definira korisnik, θ kut između brzine trokuta i njegove normale, a \vec{F}_{stop} sila zaustavljanja broda. Parametrom Γ_{max} određuje se akceleracija pri kojoj će se u potpunosti primijeniti sila zaustavljanja, dok se parametrom p postiže nelinearnost – ako se ovaj parametar postavi na

vrijednost veću od 1, tada će vrijednost regulatornog člana $\text{clamp}\left(\frac{\Gamma_i(t)}{\Gamma_{max}}, 0, 1\right)^p$ imati značajan utjecaj tek kada se $\Gamma_i(t)$ približava Γ_{max} , dok će za manje vrijednosti akceleracije biti bliži nuli. Sila zaustavljanja definirana je kao

$$\vec{F}_{stop} = m \frac{v_i}{dt} \frac{2A_i^s(t)}{S_b} \quad (4.20)$$

gdje je m masa broda, a S_b ukupna površina broda.

4.2.5. Motorski pogon

Kako bi se brod mogao kretati, dodana je još i pogonska sila motora koja se na brod uvijek primjenjuje na stražnji dio broda, a njezin smjer po horizontalnoj osi kontrolira korisnik. Smjer sile prikazan je jednostavnim kormilom koje se rotira u skladu s kontrolama korisnika.

4.3. Implementacija valnih čestica

Interaktivni valovi implementirani su u obliku sustava koji se može podijeliti na tri osnovne komponente: (1) sustav valnih čestica određenog kapaciteta, (2) generator čestica te (3) generator mape pomicanja (engl. displacement map). Sustav valnih čestica je u suštini kolekcija u koju se pohranjuju sve valne čestice, a zadužen je za ažuriranje i dijeljenje čestica. Generator čestica na temelju interakcije broda s vodom generira nove čestice te ih ubacuje u sustav čestica, dok je generator mape pomicanja zadužen za generiranje mape pomicanja na temelju živih čestica u sustavu. Ova mapa se dalje koristi u sjenčaru (engl. shader) za prikaz vode kako bi se interaktivni valovi prikazali na površini vode.

4.3.1. Sustav valnih čestica

Čestice su u sustavu valnih čestica predstavljene strukturom podataka u kojoj se nalaze sva svojstva čestica opisana u poglavlju 3.1, a sam sustav čestica predstavljen je poljem takvih struktura. Polje je ograničenog kapaciteta, pa pri dodavanju novih čestica u nekom trenutku može doći do prepisivanja čestica. Za dodavanje novih čestica u polje čestica prati se sljedeći slobodni indeks u polju. Na početku izvođenja programa indeks je 0, a svakim dodavanjem nove čestice u sustav on se inkrementira tako da pokazuje na prvo slobodno mjesto u polju. Jednom kada indeks dođe do kraja polja, ponovno se postavlja na 0 te će se pri sljedećem dodavanju čestice u sustav prepisati stara čestica. Ovdje je bitno osigurati da je kapacitet sustava čestica dovoljno velik tako da kada dođe do prepisivanja čestica u polju, stara čestica već ima malenu amplitudu uslijed većeg broja dijeljenja, te uklanjanje takve čestice ima neprimjetan

utjecaj na prikaz valova. U suprotnom bi valovi mogli naglo nestajati, što bi narušilo vjernost prikaza.

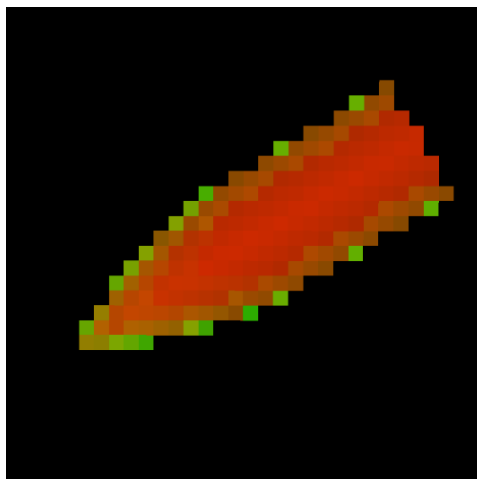
Sustav valnih čestica ima zadaću u svakom okviru odrediti čestice koje je potrebno podijeliti te zatim i provesti dijeljenje nad njima. S obzirom na to da je moguće izračunati trenutak u kojem je potrebno podijeliti česticu već pri samom stvaranju čestice, koristi se mapa dijeljenja čestica kako bi se izbjeglo iteriranje po svim česticama sustava. Ključevi ove mape su brojevi diskretnih vremenskih trenutaka od početka izvođenja programa, dok su vrijednosti liste valnih čestica koje je potrebno podijeliti u tom trenutku. U svakom okviru provjerava se postoji li ključ u mapi koji odgovara trenutnom vremenu – ako postoji, iz mape se dohvaća lista čestica. Zatim se iterira po toj listi i svaka čestica se dijeli na tri nove čestice s trećinom amplitude i trećinom kuta disperzije. Kada se ne bi koristila mapa dijeljenja čestica, u svakom okviru bilo bi potrebno iterirati po svim česticama u sustavu te provjeravati je li potrebno dijeliti ih u trenutnom okviru. U većini slučajeva tek nekolicina čestica bi se trebala podijeliti, pa ovakav pristup ne bi bio učinkovit.

4.3.2. Generator valnih čestica

Generator valnih čestica zadužen je za generiranje valnih čestica uslijed interakcije broda s vodom. Čestice se generiraju u svakom okviru u kojem se brod djelomično ili u potpunosti nalazi u vodi, u nekoliko koraka:

- određivanje siluete
- određivanje valnih utjecaja (engl. wave effects) uronjenih trokuta
- određivanje granice siluete i smjerova propagacije
- distribucija valnih utjecaja na rub siluete
- generiranje čestica na temelju distribuiranih valnih utjecaja

Prvi korak u procesu generiranja čestica je određivanje siluete dijela broda uronjenog u vodu te zapisivanje te siluete u teksturu. Silueta se generira uporabom ortogonalne projekcije uronjenog dijela broda na horizontalnu plohu te se koristi poseban sjenčar fragmenata koji za svaki piksel u crveni kanal pohranjuje dubinu na kojoj se nalazi odgovarajući dio broda, dok u zeleni kanal pohranjuje y (vertikalnu) komponentu normale broda u globalnom koordinatnom sustavu, skaliranu na raspon $[0, 1]$ kako bi se mogla zapisati u teksturu. Primjer tako generirane siluete prikazan je na slici (Slika 4.5).



Slika 4.5 Generirana silueta uronjenog dijela broda rezolucije 32x32 piksela

Pritom je važno da je rezolucija siluete niska (npr. 32x32) jer će se u zadnjem koraku tekstura čitati na CPU kako bi se mogle generirati čestice, što je relativno sporo, a s obzirom na to da je to potrebno činiti svaki okvir, veće rezolucije mogu znatno narušiti broj sličica u sekundi. Također, važno je osigurati da tekstura ima jednaku širinu i visinu te da su njene dimenzije potencije broja 2, kako bi četvrti korak izvođenja (distribucija valnih utjecaja) radila ispravno.

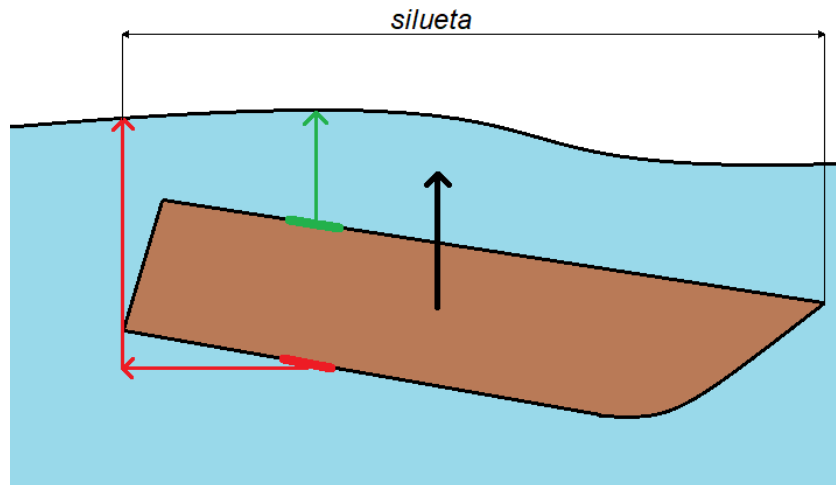
Drugi korak je određivanje i zapisivanje valnih utjecaja (engl. wave effects) za svaki trokut mreže uronjenog dijela broda. Valni utjecaj za i -ti trokut računa se prema

$$E_i = S_i(\vec{v}_i \cdot \vec{n}_i)dt \quad (4.21)$$

gdje je S_i površina trokuta, \vec{v}_i brzina centra trokuta (odnosno brzina broda u centru trokuta), \vec{n}_i normala trokuta, a dt vremenski korak između dva okvira. Valni utjecaj predstavlja volumen vode koji trokut pomakne u jednom diskretnom vremenskom koraku. Pritom je on pozitivan ako se trokut „zabija“ u vodu, a negativan ako se odmiče od vode. Valni utjecaj odgovara amplitudi valnih čestica koje će se generirati – što veći volumen vode trokut pomakne, to će biti veća amplituda generiranih čestica. Ako je valni utjecaj, odnosno amplituda čestica pozitivna, na površini vode nastat će povišenje, a ako je amplituda negativna, nastat će udubljenje.

Nakon što se izračunaju valni efekti za sve uronjene trokute, oni se zapisuju u teksturu sa siluetom. Pritom je bitno razlikovati direktne od indirektnih utjecaja. Naime, ako se uronjeni trokut nalazi na gornjoj strani broda, njegov utjecaj je direktan te se valovi stvaraju direktno iznad trokuta. Slika 4.6 prikazuje dvodimenzionalni primjer broda u potpunosti uronjenog u vodu, a direktan utjecaj trokuta naznačen je zelenom strelicom na toj slici. S druge strane, ako se trokut nalazi na donjoj strani broda, njegov utjecaj više neće biti direktan pa

se valovi neće stvarati iznad samog trokuta, već će se stvarati na rubu siluete broda. Ova situacija prikazana je crvenim strelicama na slici (Slika 4.6).



Slika 4.6 Prikaz direktnog (zeleno) i indirektnog (crveno) valnog utjecaja

Zapisivanje se izvršava u računskom sjenčaru (engl. compute shader) kojem se proslijeđuju polje s valnim efektima, polje s pozicijama centara trokuta te polje dubina na kojima se centri nalaze. Svaki valni efekt zapisuje se na piksel koji odgovara poziciji centra trokuta koji ga je prouzrokovao. Pritom se čita vrijednost y komponente normale iz zelenog kanala te ako je ta komponenta negativna (odnosno manja od 0.5 nakon skaliranja), radi se o indirektnom utjecaju. Naime, ako je y komponenta normale negativna, to znači da brod na tom mjestu nije u potpunosti uronjen, što znači da se iznad trokuta ne nalazi voda te je valni utjecaj potrebno pomaknuti na rub siluete. Ako je pak komponenta pozitivna, to znači da je brod na tom mjestu u potpunosti uronjen u vodu te je potrebno napraviti dodatnu provjeru usporedbom dubine zapisane u siluetu s dubinom na kojoj se nalazi centar trokuta. U slučaju kad je normala pozitivna, potrebno je još napraviti dodatnu provjeru usporedbom dubine zapisane u siluetu s dubinom na kojoj se nalazi centar trokuta. Ako je centar trokuta na većoj dubini od zapisane, radi se o trokutu koji se nalazi s donje strane broda te je utjecaj tada indirektan, a ako dubina odgovara zapisanoj dubini, tada se radi o trokutu s gornje strane te je utjecaj direktan. Direktni utjecaji zapisuju se u plavi kanal, dok se indirektni utjecaji zapisuju u alfa kanal teksture.

Treći korak je određivanje granice siluete i smjerova propagacije valova na toj granici. Zapravo se određuje vanjska granica siluete, odnosno oni pikseli koji nisu dio siluete, ali barem jedan susjedni piksel jest. Vanjska granica siluete određuje se u računskom sjenčaru koji kao ulaz prima siluetu sa zapisanim valnim efektima, a izlaz mu je tekstura granice. Za svaki piksel provjerava se najprije je li on sam dio siluete. Ako je, tada se on ne nalazi na vanjskoj granici te se na njegovo mjesto zapisuje crna boja. Ako piksel nije dio siluete, provjerava se njegovih 8 susjeda te se provjerava nalazi li se jedan od njih na

silueti. Ako se barem jedan susjed dio siluete, u plavi kanal se zapisuje vrijednost 1 kako bi se indiciralo da se piksel nalazi na granici. Usput se određuje i smjer propagacije čestica na način da se definira dvokomponentni vektor \vec{d} te se za svaki susjedni piksel koji se nalazi na silueti vektoru \vec{d} pridodaje jedinični vektor od susjednog piksela prema pikselu na granici. Nakon prolaska po svim susjedima, vektor smjera \vec{d} se normalizira te se njegove komponente zapisuju u crveni i zeleni kanal. Također, u alfa kanal teksture granice se prepisuju sve vrijednosti iz alfa kanala teksture siluete, odnosno svi indirektni efekti koje je potrebno distribuirati na granicu siluete.

Četvrti korak u procesu generiranja čestica je distribucija indirektnih valnih utjecaja na granicu siluete. Uz distribuciju utjecaja, provodi se i zaglađivanje smjerova propagacije čestica. Ovaj korak se provodi u dva podkoraka:

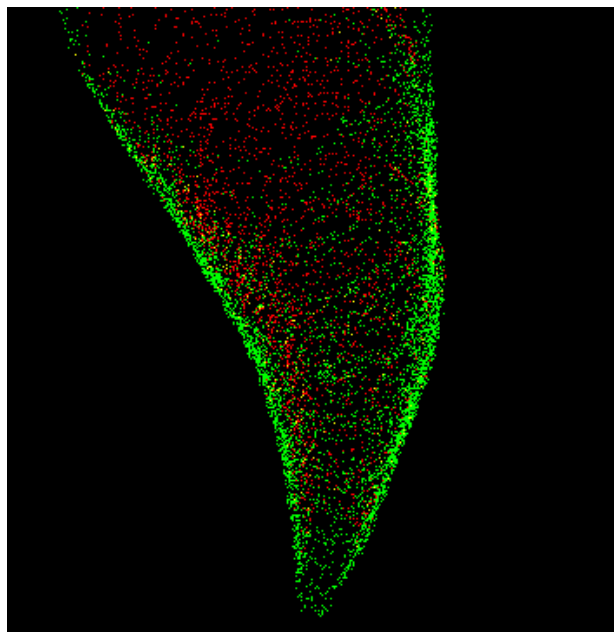
- najprije se provodi smanjivanje rezolucije teksture granice s indirektnim utjecajima u n iteracija, gdje je $n = \log_2 N$, a N je dimenzija teksture granica siluete (koja je jednaka dimenziji teksture siluete). S obzirom na to da se u svakoj iteraciji dimenzije smanjuju 2 puta, n je broj iteracija potreban da bi se rezolucija smanjila na 1×1 . Svaka iteracija izvodi se računskim sjenčarom koji kao ulaz prima rezultat prethodne iteracije (u prvoj iteraciji ulaz je tekstura granice), a izlaz mu je nova tekstura dvostruko manjih dimenzija. U svaki piksel izlazne teksture upisuje se zbroj crvenih, zelenih i alfa kanala četiriju piksela ulazne teksture koji odgovaraju tom pikselu, dok se u plavi kanal upisuje broj korespondirajućih piksela ulazne teksture koji pripadaju granici (odnosno imaju vrijednost plavog kanala veću od 0). Tada će u crvenom i zelenom kanalu biti zbrojeni smjerovi propagacije, u plavom kanalu nalazit će se broj korespondirajućih piksela koji su dio granice (vrijednost 0-4), dok će se u alfa kanalu nalaziti zbroj indirektnih utjecaja.
- u drugom podkoraku se provodi povećavanje rezolucije, ponovno u n iteracija kako bi se opet došlo do početne rezolucije prije prvog podkoraka. Svaka iteracija izvodi se računskim sjenčarom koji kao ulaz prima dvije teksture – (1) izlaz prethodne iteracije (u prvoj iteraciji to je 1×1 tekstura iz prvog podkoraka) i (2) tekstura iz prvog podkoraka koja ima dimenzije dvostruko veće od prvog ulaza, a izlaz mu je nova tekstura dvostruko većih dimenzija od prvog ulaza. U crveni i zeleni kanal izlazne teksture upisuje se prosjek smjerova zapisanih u ulaznim teksturama na odgovarajućim pikselima. Indirektni utjecaji upisuju se samo za one piksele koji su dio granice siluete u drugoj ulaznoj teksturi (odnosno imaju vrijednost u plavom kanalu veću od 0), a računaju se kao zbroj utjecaja ulaznih tekstura podijeljen s brojem zapisanim u plavom kanalu druge ulazne teksture. Plavi kanal ostaje neiskorišten.

Nakon što se indirektni utjecaji distribuiraju, na posljednju izlaznu teksturu drugog podkoraka još se prekopiraju direktni valni utjecaji u plavi kanal iz teksture siluete s valnim utjecajima, čime se dobiva konačna tekstura sa

zaglađenim smjerovima propagacije čestice, direktnim valnim utjecajima te distribuiranim indirektnim valnim utjecajima. Na temelju ove teksture se provodi peti i posljednji korak – generiranje čestica. Rezultantna tekstura čita se piksel po piksel te se za svaki piksel koji ima definiran direktni i/ili indirektni valni utjecaj generira valna čestica. Ako je utjecaj direktan, generirana čestica nema definiran smjer i ima kut disperzije 2π – dakle širi se u svim smjerovima te nastaje kružni val na površini. Ako je utjecaj indirektan, generira se čestica koja ima smjer zapisan u teksturi, dok se njen kut disperzije računa kao prosjek kutova između smjerova trenutnog piksela i susjednih piksela koji se nalaze na granici. Pritom su amplitude generiranih čestica valni utjecaji iz plavog, odnosno alfa kanala teksture, ovisno o kakvom utjecaju se radi.

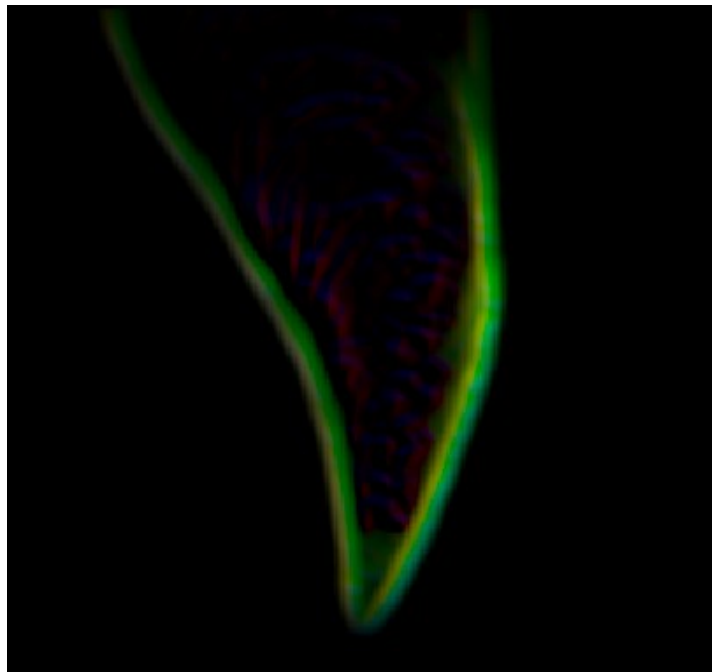
4.3.3. Generiranje mape pomicanja

Mapa pomicanja generira se na temelju čestica koje su trenutno u sustavu. Najprije se pomoću računskog sjenčara čestice u određenom području iscrtaju kao točke u crveni kanal, gdje je intenzitet njihova amplituda. Radi jednostavnosti, to područje se odabire tako da se brod nalazi u sredini područja te se specificira širina i visina (kvadratnog) područja. U teksturu se iscrtavaju samo čestice koje se nalaze u tom području, dok se ostale čestice zanemaruju. Slika 4.7 prikazuje valne čestice iscrtane kao točke. Pritom su amplitude čestica zamijenjene najvećim iznosom kako bi se čestice mogle vidjeti (amplitude su relativno male, pa tekstura izgleda gotovo potpuno crno). Također, na slici su crvenom bojom prikazane čestice s negativnom amplitudom, dok su zelenom prikazane one s pozitivnom te je prikazan samo dio teksture od interesa kako bi se povećala vidljivost (ostatak teksture je crn).



Slika 4.7 Prikaz valnih čestica s pozitivnom (zeleno) i negativnom (crveno) amplitudom kao točke

Nakon što se sve čestice u području iscrtaju, nad dobivenom teksturom se vrši konvolucija s tri filtra koji predstavljaju lokalnu devijaciju čestice te se rezultat sprema u novu teksturu – konačnu mapu pomicanja. Filtri su definirani prema jednažbama (3.5) (za vertikalnu devijaciju po y osi) i (3.6) (za horizontalnu devijaciju po x i z osima). Za svaki piksel izlazne teksture prolazi se kroz kvadratno područje ulazne teksture koje po širini i visini odgovara radijusu čestice te se za svaki piksel iz tog područja računa njegova udaljenost od centra filtra (nakon pretvorbe u globalni koordinatni sustav), na temelju te udaljenosti odrede se vrijednosti jednažbi filtara te se rezultat pribraja trenutnom izlaznom pikselu. Slika 4.8 prikazuje tako generiranu mapu pomicanja. U crveni kanal zapisan je pomak po x osi, u zeleni po y osi, a u plavi pomak po z osi (na slici su vidljivi samo pozitivni pomaci).



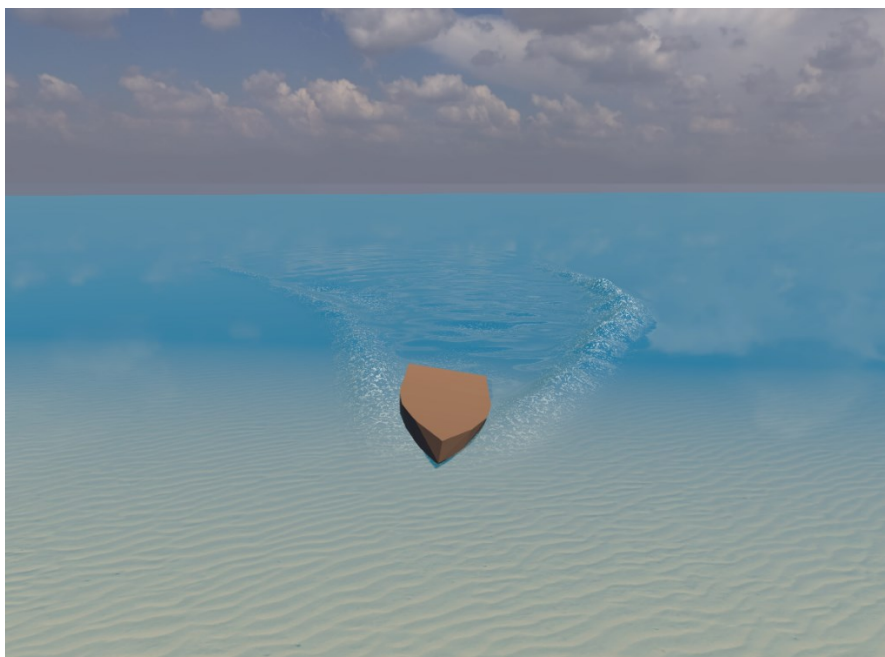
Slika 4.8 Konačna generirana mapa pomicanja

5. Rezultati

Rezultati implementacije s ambijentalnim valovima prikazan je na prvoj slici (Slika 5.1), dok je na drugoj slici (Slika 5.2) prikaz rezultata bez ambijentalnih valova – ovdje su prisutni samo valovi nastali interakcijom broda s vodom. Kako bi se interaktivni valovi dodatno istaknuli, u prikazu im je dodana tekstura morske pjene čiji je intenzitet proporcionalan s visinom vala.



Slika 5.1 Prikaz simulacije u izvođenju



Slika 5.2 Prikaz simulacije u izvođenju bez ambijentalnih valova

Korisniku je omogućeno upravljanje brodom i kamerom, a podržane kontrole su navedene u tablici (Tablica 5.1).

Tablica 5.1 Korisničke kontrole

KONTROLA	OPIS KONTROLE
W	<i>dodaje pogon motora u pozitivnom smjeru (naprijed)</i>
S	<i>dodaje pogon motora u negativnom smjeru (natrag)</i>
A	<i>rotira kormilo ulijevo, pa brod skreće ulijevo</i>
D	<i>rotira kormilo udesno, pa brod skreće udesno</i>
Pomak miša	<i>rotira kameru oko broda, koji je uvijek u središtu</i>
Kotačić na mišu	<i>približava/udaljava kameru od broda</i>
Desni klik mišem	<i>Zaključava/otključava kontrole kamere</i>
Space	<i>pauzira/nastavlja izvođenje simulacije</i>
Escape	<i>prekid rada simulacije</i>

Simulacija se izvodi u prosjeku sa 115 sličica u sekundi na računalu s grafičkom karticom Nvidia RTX 2080S 8Gb GDDR6, procesorom i7 9700k @3.6GHz, 32Gb DDR4 radne memorije i rezolucijom 2560x1440. Pritom je rezolucija visinske mape ambijentalnih valova 512x512, rezolucija siluete broda 32x32, rezolucija mape pomicanja interaktivnih valova 512x512, a kapacitet sustava čestica 100000. Isključivanjem simulacije ambijentalnih valova u prosjeku se dostiže 220 sličica u sekundi.

6. Zaključak

Simulacija interakcije plovila i vode vrlo je zahtjevan zadatak za koji još uvijek ne postoji analitičko rješenje. Dok *offline* simulacije mogu postići vrlo vjerne rezultate, simulacije u stvarnom vremenu znatno su ograničenije što se tiče performansi te je potrebno uvoditi razne aproksimacije i pojednostavljenja kako bi se ta ograničenja mogla zadovoljiti. Međutim, iako simulacija nije u potpunosti realistična, moguće je postići prihvatljive rezultate za potrebe primjerice interaktivnih simulacijskih sustava ili video igara, gdje greške prouzrokovane aproksimacijama ne igraju veliku ulogu, dok simulacija pridonosi uživljavanju u virtualno okruženje.

U ovom radu dan je samo jedan od više mogućih načina simulacije interakcije plovila s vodom u stvarnom vremenu. Problem je i dalje otvoren te još uvijek ne postoji apsolutno najbolje rješenje koje bi u svim aspektima dominiralo nad ostalima.

Literatura

- [1] Tessendorf, J., *Simulating Ocean Water*, (2001)
- [2] Flügge, F. J., *Realtime GPGPU FFT Ocean Water Simulation*, (2017)
- [3] Kerner, J., *Water Interaction Model for Boats in Video Games*, (2015)
- [4] Batchelor, G. K., *An Introduction to Fluid Dynamics*, Cambridge University Press (1967)
- [5] Yuksel, C., *Real-Time Water Waves With Wave Particles*, (2010)

Sažetak

Naslov: Vizualizacija simulacije kretanja plovila po vodi

Rad opisuje jedan od načina simulacije interakcije plovila s vodom u stvarnom vremenu. Opisane su hidrostatičke i hidrodinamičke sile kojima se simulira utjecaj vode na plovilo, kao što su sila uzgona i sile otpora. Opisan je i sustav valnih čestica, kojim se simulira stvaranje interaktivnih valova prouzročenih utjecajem plovila na vodu. Navedene su aproksimacije i pojednostavljenja koja su ugrađena u implementaciju, kako bi se poboljšale performanse cjelokupne simulacije te omogućilo izvođenje u stvarnom vremenu.

Ključne riječi: računalna grafika, simulacija vode, interakcija plovila s vodom, valovi, valne čestice, sustav čestica, hidrostatske sile, hidrodinamičke sile

Abstract

Title: Visualization of Water Interaction Model for Boats

This paper describes one of the ways to simulate the interaction of a boat with water in real time. Hydrostatic and hydrodynamic forces used to simulate the influence of water on a boat are described, such as buoyancy and drag forces. The wave particle system is also described, which is used to simulate the interactive waves created by the influence of boat on the water. Approximations and simplifications integrated into the implementation are listed, which improve the performance of the simulation, enabling execution in real time.

Keywords: computer graphics, water simulation, boat-water interaction, waves, wave particles, particle system, hydrostatic forces, hydrodynamic forces