



## Zadání bakalářské práce

**Název:** Simulování vodního povrchu  
**Student:** Hong Son Ngo  
**Vedoucí:** Ing. Petr Pauš, Ph.D.  
**Studijní program:** Informatika  
**Obor / specializace:** Webové a softwarové inženýrství, zaměření Počítačová grafika  
**Katedra:** Katedra softwarového inženýrství  
**Platnost zadání:** do konce letního semestru 2022/2023

### Pokyny pro vypracování

Voda a vodní hladina se vyskytuje ve spoustě grafických aplikací a hrách a jejich realistická real-time simulace je žádoucí. Cílem této práce je vytvořit simulátor vodního povrchu a jeho případných světelných efektů na tělesa pod vodou.

1. Analyzujte možnosti real-time simulace vodní hladiny.
2. Analyzujte vhodné nástroje pro její simulaci (např. OpenGL, Unity, atd.).
3. Na základě analýzy vyberte vhodný nástroj pro simulaci a navrhňte prototyp simulace.
4. Ve zvoleném nástroji implementujte.
5. Vytvořte testovací scénu, která simulaci demonstruje.





**FAKULTA  
INFORMAČNÍCH  
TECHNologiÍ  
ČVUT V PRAZE**

Bakalářská práce

## **Simulování vodního povrchu**

*Hong Son Ngo*

Katedra softwarového inženýrství  
Vedoucí práce: Ing. Petr Pauš, Ph.D.

13. března 2022



---

## Poděkování

Doplňte, máte-li komu a za co děkovat. V opačném případě úplně odstraňte tento příkaz.



---

# Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principu při přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 2373 odst. 2 zákona č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů, tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu) licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 13. března 2022

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2022 Hong Son Ngo. Všechna práva vyhrazena.

*Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí a nad rámec oprávnění uvedených v Prohlášení na předchozí straně, je nezbytný souhlas autora.*

### **Odkaz na tuto práci**

Ngo, Hong Son. *Simulování vodního povrchu*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2022.



---

## Abstrakt

V několika větách shrňte obsah a přínos této práce v češtině. Po přečtení abstraktu by se čtenář měl mít čtenář dost informací pro rozhodnutí, zda chce Vaši práci číst.

**Klíčová slova** Nahradte seznamem klíčových slov v češtině oddělených čárkou.

---

## Abstract

Sem doplňte ekvivalent abstraktu Vaší práce v angličtině.

**Keywords** Nahradte seznamem klíčových slov v angličtině oddělených čárkou.



---

# Obsah

<b>Úvod</b>	<b>1</b>
<b>1 Cíl práce</b>	<b>3</b>
<b>2 Základ teorie mechaniky tekutin</b>	<b>5</b>
2.1 Navierovy–Stokesovy rovnice . . . . .	5
2.1.1 Vysvětlení . . . . .	5
2.2 Využití . . . . .	7
2.3 Popisy tekutiny . . . . .	7
2.3.1 Lagrangeův popis . . . . .	7
2.3.2 Eulerův popis . . . . .	7
<b>3 Vlastnosti vodního povrchu</b>	<b>9</b>
3.1 Dynamické vlastnosti . . . . .	9
3.1.1 Procedurální metody . . . . .	10
3.1.1.1 Vlnění za pomoci sinusoid . . . . .	11
3.1.1.2 Vlnění za pomoci Gerstnerovy vlny . . . . .	12
3.1.1.3 Vlnění za pomoci Fourierovy transformace . . . . .	14
3.2 Světelné vlastnosti . . . . .	15
<b>4 Realizace</b>	<b>17</b>
<b>Závěr</b>	<b>19</b>
<b>A Seznam použitých zkratk</b>	<b>21</b>
<b>B Obsah přiloženého CD</b>	<b>23</b>



---

## Seznam obrázků

3.1	Ukázka off-line simulace Eulerovy tekutiny [Guendelman05] . . . .	10
3.2	Simulace vlnění pomocí jedné sinusoidy . . . . .	11
3.3	Simulace vlnění pomocí sčítání sinusoid . . . . .	11
3.4	Porovnání mezi vlnami funkcí 3.1 a 3.3 . . . . .	12
3.5	Gerstnerova vlna s fázovou rychlostí $c$ , vlnovou délkou $\lambda$ a rozdílem hřebenu a údolí $H$ <a href="https://commons.wikimedia.org/wiki/File:Trochoidal_wave.svg">https://commons.wikimedia.org/wiki/File: Trochoidal_wave.svg</a> . . . . .	13
3.6	Porovnání hodnot $s$ ve dvourozměrném prostoru . . . . .	13
3.7	Implementace Gerstnerových vln pro simulování vodního povrchu <a href="https://catlikecoding.com/unity/tutorials/flow/waves/">https://catlikecoding.com/unity/tutorials/flow/waves/</a> . .	14



---

# Úvod

Kvalita herních titulů za poslední let ohromně vzrostla. Jejich úspěch lze připsat nejen zajímavému příběhovému obsahu, ale také i jejich vizuálnímu zpracování. Právě v grafickém provedení můžeme vidět největší pokrok. Díky hardwarovým zlepšením ve výpočetních technologiích zejména v grafických kartách se obraz grafických aplikací čím dál více blíží fotorealistickým výsledkům a s příchodem zařízení pro virtuální a rozšířenou realitu je o realistický obraz ještě větší zájem.

Co rozlišuje scény napříč historií her, jsou speciální efekty. Na základě kvality jejich provedení je hráč hlouběji vnořen do virtuálního světa a následně i do jeho příběhu. Jedním z takových efektů jsou např. přírodní jevy jako pohyb plamene, vlnění hladiny vody, proudění větru...

Fyzikálně korektní chování přírodních jevů však zůstává do dnešního dne mezi komplexnějšími problémy. Mezi těmi výpočetně nejobtížnějšími je považováno proudění tekutin, do kterých patří jak plynné, tak i kapalná skupenství. Pro vizualizaci se většina metod zabývá zejména kapalinami kvůli její povaze (jsou lidským okem viditelné), ale v některých případech lze postupy pro simulování proudění kapalin využít i pro zobrazení plynů jako např. kouře nebo plamene.

Kvůli vysoce dynamickému chování tekutin se však v současnosti nejrealističtějších výsledků dosáhne hlavně off-line metodami. Real-time aplikace využívají stejných principů, ale s podstatnými kompromisy jak paměťovými, tak i výpočetními. Vzhledem k tomu, že některé aplikace nevyužijí přesné a hlavně výpočetně drahé simulace, tak napodobují jen výsledné efekty, které se nijak neopírají o fyzikální pravidla. Ačkoli off-line metody zobrazují věrohodně chování kapalin, resp. plynů, nejsou nijak zastoupeny v hrách, neboť je interakce s uživatelem a dynamické prostředí scén nedovolí využít.

Obsah této bakalářské práce se zabývá real-time simulací vodního povrchu a jeho případnými světelnými efekty jak na hladině, tak i na tělese pod ní, jejíž aplikace by bylo možné využít k fotorealistickému zobrazení virtuálních

## Úvod

---

scén v hrách.



---

## Cíl práce

Cílem teoretické části této bakalářské práce je prozkoumat možnosti real-time simulace vodního povrchu a případných vlastností, které interagují se světelnými paprskami osvětelující vodní plochu. Na základě analýzy současných metod prozkoumat vhodné nástroje pro její zobrazení v grafických aplikacích jako jsou počítačové hry.

Následně v praktické části je cílem vybrat vhodné simulační metody pro real-time grafické aplikace, navrhnout prototyp simulátoru a ve zvoleném nástroji ho implementovat a nakonec vytvořit testovací scénu, která simulaci demonstruje.



# Základ teorie mechaniky tekutin

V této kapitole je shrnutí teorie mechaniky tekutin, která je nezbytnou součástí pro simulování fyzikálně korektního chování vody. Teorie je zde vyložena takovým způsobem, aby byl čtenář s ní seznámem a co nejrychleji pochopil principy chování tekutin, neobsahuje žádné rigorózní vysvětlení problematiky.

## 2.1 Navierovy–Stokesovy rovnice

Proud tekutin se v reálném světě řídí podle Navierovými–Stokesovými rovnicemi (NSE), soustavou nelineárních diferenciálních rovnic:

$$\nabla \cdot \vec{u} = 0 \quad (2.1)$$

$$\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \nabla \vec{u} + \frac{1}{\rho} \nabla p = \vec{g} + \nu \nabla \cdot \nabla, \vec{u} \quad (2.2)$$

kde vektor  $\vec{u} = (u, v, w)$  označuje rychlost tekutiny,  $\rho$  označuje hustotu tekutiny,  $p$  tlak, kterým působí tekutina na své okolí,  $\vec{g} = (x, y, z)$  je gravitační zrychlení,  $\nu$  je značení kynematické viskozity tekutiny. Symboly  $\nabla$ ,  $\nabla \cdot$ ,  $\nabla \cdot \nabla$  označují diferenciální operátory nabra, divergence a Laplaceův operátor.

### 2.1.1 Vysvětlení

Před vysvětlení jednotlivých rovnic je dobré zmínit, že většina teorie mechaniky tekutin je založena na odvětví matematiky vektorové analýzy a pracují s vektorovými poli, resp. skalárními poli. NSE např. pracují s vektorovým poli, resp. se skalárními poli, kde jednotlivým bodům prostoru přiřazují vektor určující rychlost proudu tekutiny, resp. hodnotu tlaku.

Na první pohled vypadají rovnice těžce uchopitelné, ale myšlenka za nimi je velmi jednoduchá. První rovnice 2.1 popisuje zákon o zachování hmotnosti, tím je myšleno, že není možné, aby hmota tekutiny na některém místě z ničeho vznikla nebo zanikla.

Druhá rovnice 2.2 popisuje zákon o zachování hybnosti a je ve své podstatě Newtonův druhý zákon.

$$\vec{F} = m\vec{a} \quad (2.3)$$

Zrychlení  $\vec{a}$  lze přepsat je derivací rychlosti podle času.

$$\vec{F} = m \frac{D\vec{u}}{Dt} \quad (2.4)$$

Pro odvození celé rovnice je třeba rozvést sílu  $\vec{F}$ , které na tekutinu působí [Bridson2006]. Jedna z nich je samozřejmě gravitace, jejíž hodnota je vyčíslena jako  $m\vec{g}$ .

Další síly vytváří tekutina sama na sobě. První z nich je síla způsobena rozdílem tlaků v tekutině. Tekutina se v oblastech s vyšším tlakem přesouvá do oblastí s nižším tlakem. Její hodnotu můžeme zapsat jako  $-V\nabla p$ <sup>1</sup>.

Další síla působící na tekutinu je ovlivněna její viskozitou. Viskozita působí při každém pohybu částic tekutiny a snaží se vyrovnat rychlost částic rychlostí svých sousedních částic. Její hodnotu můžeme vyjádřit jako  $V\mu\nabla \cdot \nabla\vec{u}$ <sup>2</sup>, kde  $\mu$  označuje koeficient dynamické viskozity.

$$m\vec{g} - V\nabla p + V\mu\nabla \cdot \nabla\vec{u} = m \frac{D\vec{u}}{Dt} \quad (2.5)$$

Rovnice 2.3, 2.4 a 2.5 předpokládají, že tekutinu lze rozložit na konečně mnoho malých částí, tímto způsobem je do vyčíslení představena výpočetní chyba. Řešením toho problému je celou rovnici 2.5 je vydělit  $V$ , aby zachytila pohyb nekonečně mnoho nekonečně malých částic tekutin. Dobré je připomenout, že hustota  $\rho$  lze vyjádřit jako  $\frac{m}{V}$

$$\rho\vec{g} - \nabla p + \mu\nabla \cdot \nabla\vec{u} = \rho \frac{D\vec{u}}{Dt} \quad (2.6)$$

Následně po vydělení hustoty  $\rho$ , vyjádření derivace  $\frac{D\vec{u}}{Dt}$  řetízkovým pravidlem a prohození sčítanců získáme druhou Navierovu–Stokesovu rovnici. Kynematická viskozita  $\nu$  je rovna  $\frac{\mu}{\rho}$ .

$$\frac{\partial\vec{u}}{\partial t} + \vec{u} \cdot \nabla\vec{u} + \frac{1}{\rho}\nabla p = \vec{g} + \nu\nabla \cdot \nabla\vec{u} \quad (2.7)$$

---

<sup>1</sup>Operace  $\nabla p$  vyjadřuje vektorové pole, ve kterém vektory směřují, z jednotlivých bodů na své sousední body, aby jeho hodnota ve skalárním poli nejrychleji vzrostla.

<sup>2</sup>Operace  $\nabla \cdot \nabla\vec{u}$  vyjadřuje míru deviace rychlosti částice okolo svého okolí.

## 2.2 Využití

Výhodou NSE je, že je lze aplikovat na téměř jakékoliv tekutiny. V praxi se využívají pro modelování počasí, podnebí pro předpověď počasí, proudění v oceánech, výpočtu aerodynamických vlastností vozidel. Přestože využití rovnic je nespočetné, mají zásadní problém, neboť do dnes nevíme, zdali mají pro náhodný vstup nějaké řešení. Pro zjednodušení výpočtu existují několik aproximací, které je umožňují aplikovat se zanedbatelnými chybami v modelovacích systémech.

## 2.3 Popisy tekutiny

V teorii mechaniky tekutin existují dva různé pohledy popisu tekutin. Na základě těchto popisů jsou následně založeny algoritmy pro simulaci kapalin, resp. plynů, které jich využívají pro diskretizaci NSE.

### 2.3.1 Lagrangeův popis

Lagrangeův popis je jeden z popisů kontinua, který si většina nejspíše vybaví. Na tekutinu nahlíží jako na systém částic. Na jednotlivé body tekutiny nahlíží jako částice, které mohou být na základě potřeby různě velké, např. jako molekule nebo části tekutiny o nějakém objemu. Ke každé z nich Lagrangeův popis přiřazuje její pozici  $\vec{x}$  a rychlost  $\vec{u}$  a sleduje ho, jak se tyto hodnoty v čase mění.

### 2.3.2 Eulerův popis

Eulerův popis kontinua je na první pohled neintuitivní. Eulerův postup diskretizuje prostor tekutiny na pevně dané oblasti, ve kterých měří vlastnosti tekutiny jako např. tlak, rychlost proudu, teplotu, jak se v čase mění. Přestože tento postup vypadá omezující a složitý kvůli sledování veličin tekutin jen v pevných bodech, je v metodách pro simulování tekutin preferovaným popisem kontinua.



## Vlastnosti vodního povrchu

Tato kapitola obsahuje popis vizuálních vlastností vodní hladiny, které v reálném světě lze pozorovat, a k jejím hlavním rysům jsou vypsány algoritmy, které je simulují. Z pohledu počítačové grafiky lze vlastnosti vody rozdělit do dvou kategorií:

**dynamické vlastnosti** popisující pohyb vodní hladiny,

**světelné vlastnosti** popisující interakce povrchu vody se světelnými prsky.

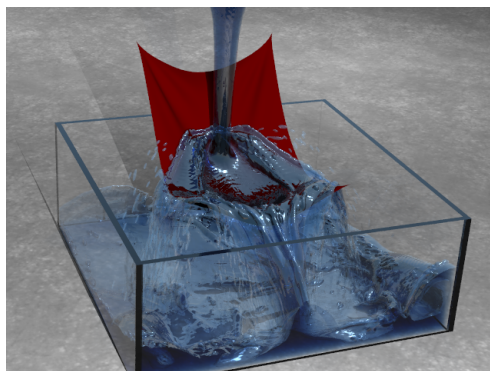
### 3.1 Dynamické vlastnosti

Dynamické vlastnosti zachycují, jak se vodní hladina pohybuje a jak reaguje na dynamické prostředí.

V současnosti nejrealističtější výsledků je dosaženo simulací Eulerovy vody (nahlíží na vodu Eulerovým popisem), jejíž prostor je rozdělen do alespoň  $512^3$  buněk. Takto řídké rozdělení obsahuje přes 100 milion neznámých pro vyřešení a s užitím globálních zobrazovacích metod jako ray-tracing pro realistickou vizualci odrazů, refrakcí a kaustik je výpočetně nemožné simulovací provést v reálném čase [MuellerGDC08].

Proto real-time simulace vody vhodné pro aplikace jako hry musí nutně splňovat tyto podmínky [MuellerGDC08]:

- být výpočetně rychlé – zlomek 15 ms, který je třeba pro vykreslení jednoho snímku,
- být paměťově nenáročné,
- být stabilní – korektně reagovat i na nefyzikálně pohybující se objekty.



Obrázek 3.1: Ukázka off-line simulace Eulerovy tekutiny [Guendelman05]

Jeden přístup, jak se nejvíce přiblížit off-line simulaci a stále dodržet podmínky pro real-time simulaci ve hrách, je zachovat stejný algoritmus jako při off-line simulaci, ale zmenšit rozlišení prostoru simulace. Tento postup ale spíše ubírá na realitě vodního povrchu, neboť se voda nefyzikálně shlukuje a detaily z obrazu jsou vynechány.

Další z možností, jak splnit výše zmíněné podmínky, je omezit míru interakce s vnějším prostředím, která tvoří největší výpočetní překážku. Na základě významnosti vodních útvarů ve scéně může simulace reagovat na všechny rigidní tělesa nebo u případů, ve kterých voda slouží jako pozadí, opominout jakoukoliv interakci. Podle tohoto principu můžeme dělit metody simulace vodní plochy na [MuellerGDC08]:

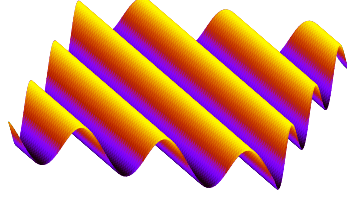
- procedulární
- částicové
- hybridní

#### 3.1.1 Procedulární metody

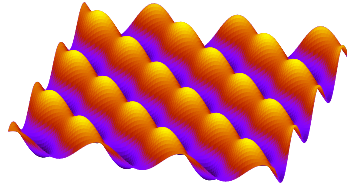
Procedulární metody simulují konečné efekty jako vlnění vodního povrchu, které nejsou vyvolány fyzikální činiteli. Největší výhodou této metody je výpočetní rychlost a versetelita, na druhou stranu ale vodní útvary nereagují na dynamické prostředí.

Na základě těchto vlastností se procedulární metody používají pro vizualizaci rozsáhlé vodní plochy, které ve scéně hraje malou roli, např. jako oceán v pozadí scény.





Obrázek 3.2: Simulace vlnění pomocí jedné sinusoidy



Obrázek 3.3: Simulace vlnění pomocí sčítání sinusoid

#### 3.1.1.1 Vlnění za pomoci sinusoid

Mezi prvními, kteří zkoumali procedurální metody pro zobrazení vodního povrchu, byl Nelson. L. Max ve své práci „Vectorized Procedural Models for Natural Terrain: Wave and Islands in the Sunset“, ve které modeluje vlnění povrchu pomocí sinusoid. Na základě horizontální pozici a času Max manipuloval výšku vrcholů roviny [Max81]. Výšku jednotlivých vrcholů roviny vyjádřil jako funkci

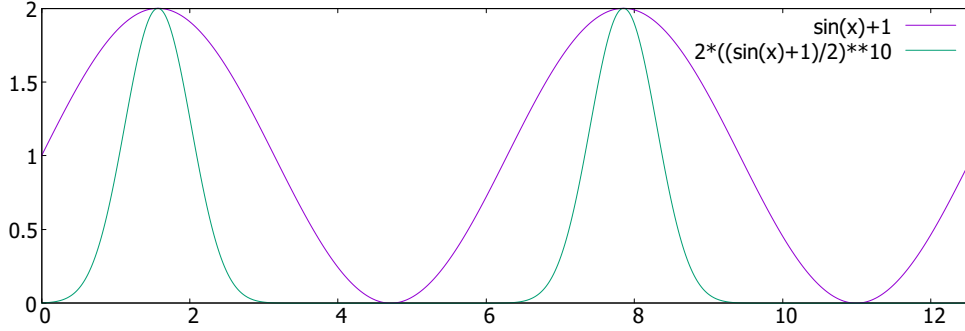
$$h(x, z, t) = -y_0 + A \sin(k_x x + k_z z - \omega t + \varphi), \quad (3.1)$$

kde  $(x, z)$  je horizontální pozice vrcholu roviny (v celé práci se bude považovat, že kladná osa  $y$  směřuje nahoru),  $t$  je čas,  $y_0$  je výška hladiny vody v klidovém stavu (při nulovém výskytu vln),  $A$  reprezentuje amplitudu roviny,  $\vec{k} = (k_x, k_z)$  je vlnový vektor reprezentující směr a rychlost propagace vln,  $\omega$  je úhlová frekvence a  $\varphi$  je fáze vlny.

Tento model se ale vlní jen v jednom směru a výsledné vlny vypadají nerealisticky hladce. K dosažení větších detailů lze k funkci přičíst další sinusoidami s odlišnými parametry amplitudy, vlnového vektoru nebo úhlové frekvence:

$$h(x, z, t) = -y_0 + \sum_{i=1}^{N_w} A_i \sin(k_{x_i} x + k_{z_i} z - \omega_i t + \varphi_i), \quad (3.2)$$

kde  $N_w$  je celkový počet vln.



Obrázek 3.4: Porovnání mezi vlnami funkcí 3.1 a 3.3

Realistického řešení však za pomoci jen obyčejných sinusoid nelze dosáhnout. Max si všiml, že vlnění oceánu s vyššími amplitudami mají užší hřeben a mělké údolí, zatímco vrchol a údolí sinusoidy je stejně oblý. Mark Finch a Cyan Worlds přišly s řešením toho problému, které stále využívá jednoduchých sinusoid. Kromě obyčejných funkcí sinusoid přičítá navíc funkci

$$f_i(x, z, t) = -y_0 + 2 \left( \frac{\sin(k_{x_i}x + k_{z_i}z - \omega_i t + \varphi_i) + 1}{2} \right)^k, \quad (3.3)$$

kde  $k \in \mathbb{R}^+$  určuje míru, jak má být hřeben úzký [GPUGems1].

#### 3.1.1.2 Vlnění za pomoci Gerstnerovy vlny

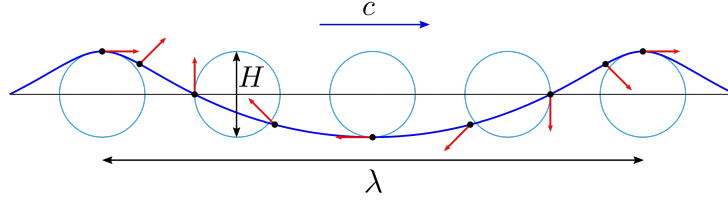
Limitující faktor metody založené na transformaci vrcholů pomocí sinusoid je, že manipuluje s jediným parametrem vrcholu, tj. výškou, a pro realističtější výsledky bylo třeba manipulovat vrcholy i v horizontální rovině. Tento problém řeší cyklické křivky, tzv. trochoidy, podle kterých Franz Josef Gerstner, německý fyzik, modeloval vlnění hladiny v hlubokých vodách [Gerstner1802]. Trochoidou nazýváme trajektorii bodu pevně spojeného s „kotálející se“ kružnice po nehybné přímce [https://www.fa.cvut.cz/studium/predmety/deskriptivni-geometrie-i-ii/dg\\_elskripta/krivky/cyklicke\\_krivky\\_1.pdf](https://www.fa.cvut.cz/studium/predmety/deskriptivni-geometrie-i-ii/dg_elskripta/krivky/cyklicke_krivky_1.pdf).

Dnes v teorii dynamiky tekutin vlnění založené na trochoidních křivkách nazýváme Gerstnerovými vlnami. Mezi prvními, kteří se Gerstnerovými vlnami zabývali, byli Alain Fournier a William T. Reeves, kteří transformovali vrcholy roviny podle těchto parametrických rovnic [Fournier86]:

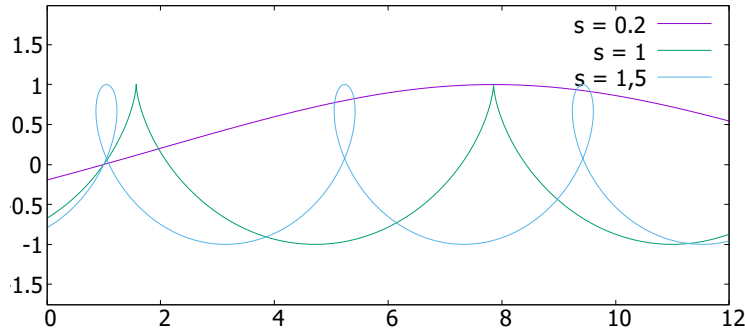
$$x = x_0 + r \cos(k_x x_0 + k_z z_0 - \omega t) \quad (3.4)$$

$$y = r \sin(k_x x_0 + k_z z_0 - \omega t) \quad (3.5)$$

$$z = z_0 + r \cos(k_x x_0 + k_z z_0 - \omega t), \quad (3.6)$$



Obrázek 3.5: Gerstnerova vlna s fázovou rychlostí  $c$ , vlnovou délkou  $\lambda$  a rozdílem hřebenu a údolí  $H$  [https://commons.wikimedia.org/wiki/File:Trochoidal\\_wave.svg](https://commons.wikimedia.org/wiki/File:Trochoidal_wave.svg)



Obrázek 3.6: Porovnání hodnot  $s$  ve dvourozměrném prostoru

kde vektor  $(x_0, z_0)$  reprezentuje klidovou pozici vrcholu roviny,  $r$  je délka opírajícího bodu od středu kružnice trochoidní křivky, vektor  $\vec{k} = (k_x, k_z)$  určuje rychlost a směr propagace vlnění,  $\omega$  označuje úhlovou frekvenci a  $t$  je čas.

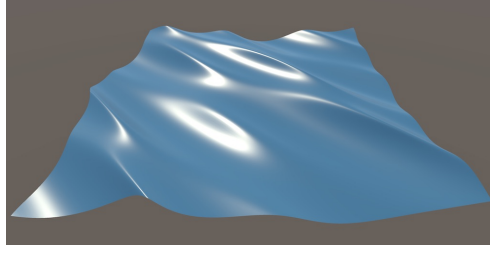
Hodnota  $s = r|\vec{k}|$  určuje, jak strmá bude vlna. Pro hodnotu  $s = 0.2$  má vlna tvar jako sinusoida, pro  $s = 1$  má tvar cykloidy a pro  $s > 1$  dochází samoprotínání trochoidy, a proto by se hodnotám vyšší než jedna mělo vyhýbat pro vizualizaci vodního povrchu. V obrázku 3.6 je porovnání hodnot  $s$  pro vlnění ve dvourozměrném prostoru.

Detailnějších výsledků lze získat podobně jako u sinusoid pomocí skládání Gerstnerových vln:

$$x = x_0 + \sum_{i=1}^{N_w} r_i \cos(k_{x_i} x_0 + k_{z_i} z_0 - \omega_i t) \quad (3.7)$$

$$y = \sum_{i=1}^{N_w} r_i \sin(k_{x_i} x_0 + k_{z_i} z_0 - \omega_i t) \quad (3.8)$$

$$z = z_0 + \sum_{i=1}^{N_w} r_i \cos(k_{x_i} x_0 + k_{z_i} z_0 - \omega_i t). \quad (3.9)$$



Obrázek 3.7: Implementace Gerstnerových vln pro simulování vodního povrchu  
<https://catlikecoding.com/unity/tutorials/flow/waves/>

### 3.1.1.3 Vlnění za pomoci Fourierovy transformace

Předešlé dvě metody jsou kvalitními nástroji pro simulaci vodního povrchu a za jejich pomoci lze i získat fotorealistických výsledků pro vysoký počet vln  $N_w$  v řádech tisíců nebo více. Vysoký počet představuje však výpočetní překážku. Obvykle by výpočet posunutí podle rovnic 3.2 nebo 3.7–3.9 herní engine provedly ve vertex shaderech, pro které je však výpočet funkcí sinus a cosinus náročné [<https://paroj.github.io/glutut/Positioning/Tut03%20n%20Vertex%20Shader%20Performance.html>].

Johanson řešil výpočetní náročnost adaptivní metodou, která na základě vzdálenosti kamery a vodní plochy vyřazuje vlny [Johanson2004]. V případě, že je kamera dostatečně vzdálená od vodní plochy, vyřazuje Johanson vlnění s vysokými frekvencí. Efektivnějšího výsledku dosáhl Lee [LeeGL2006], který před vykreslení vodní plochy navíc zahazuje vrcholy geometrie hladiny, které jsou mimo rozsah kamery.

V současnosti nejlepším řešením tohoto problému je užití rychlé Fourierovy transformace (FFT), resp. inverzní Fourierovy transformace (IFFT) podle Jerryho Tessendorfa [Tessendorf01]. Sčítání sinusoid v rovnici 3.2 je ve podstatě inverzní Fourierova transformace, kde jednotlivé sinusoidy přispívají ke konečnému vlnění. Tessendorf podle stastických dat vlnění oceánů, např. ze satalitních snímků, modeluje výšku vrcholů hladiny jako inverzní Fourierovu transformaci funkce

$$h(x, z, t) = \sum_k \tilde{h}(k_x, k_z, t) e^{i(k_x, k_z) \cdot (x, z)}, \quad (3.10)$$

$(x, z)$  je pozice vrcholů roviny, funkce  $\tilde{h}$  závislá na čase  $t$  obsahuje informaci o amplitudě a fázi sinusoidy  $e^{i(k_x, k_z) \cdot (x, z)}$  s vlnovým vektorem  $(k_x, k_z)$ . Tessendorf následně podle oceánografických dat vhodně volí funkci  $\tilde{h}$ , aby bylo výsledné vlnění nejvíce realistické.

Časová komplexita triviálního sčítání sinusoid v rovnici 3.2, tj. inverzní Fourierovy transformace, je  $\mathcal{O}(n^2)$ , kdežto užitím algoritmů pro IFFT se redukuje časová komplexita na  $\mathcal{O}(n \log n)$ .

## 3.2 Světelné vlastnosti



## **Realizace**





---

## Závěr

Jedním z cílů práce bylo provést analýzu současných možností pro simulaci vodního povrchu. Rozbor obsahuje několik metod založených na různých přístupech k simulaci kapalin. Na základě typu scén her a míry interakce vodní plochy s prostředím jsou jednotlivé metody různě vhodné.

Dalším cílem bylo vytvořit testovací scénu s vybraným algoritmem pro simulace vodní plochy z analýzy. Testovací scéna je naimplementována pomocí frameworku OpenGL, ale principy simulace jsou lehce přenositelné na různé nástroje. Vodní plocha reaguje jak na prostředí testovací scény, tak i na manipulace hladiny uživatelem.



## Seznam použitých zkratk

**NSE** Navier–Stokes equations

**FFT** Fast Fourier transform

**IFFT** Inverse fast Fourier transform



## Obsah přiloženého CD

	readme.txt .....	stručný popis obsahu CD
	exe.....	adresář se spustitelnou formou implementace
	src	
	impl.....	zdrojové kódy implementace
	thesis.....	zdrojová forma práce ve formátu L <sup>A</sup> T <sub>E</sub> X
	text .....	text práce
	thesis.pdf.....	text práce ve formátu PDF
	thesis.ps.....	text práce ve formátu PS