



# **Szerkezetek lökésszerű hidrodinamikai igénybevételének numerikus modellezése SPH-val**

**Tudományos Diákköri Konferencia**

Budapesti Műszaki és Gazdaságtudományi Egyetem  
2012

**Készítette:**

Tóth Balázs, MSc hallgató

Gépészmérnöki Kar

**Konzulens:**

Dr. Szabó Kálmán Gábor, egyetemi docens

Építőmérnöki Kar, Vízépítési és Vízgazdálkodási Tanszék

# Tartalom

1	Bevezetés.....	3
2	Folyadékok mechanikája .....	4
2.1	Az Euler-féle numerikus folyadékmodellezés .....	4
2.2	A Lagrange-féle numerikus folyadékmodellezés.....	5
3	Az SPH – Smoothed Particle Hydrodynamics .....	6
3.1	Az SPH alapjai .....	7
3.2	Simító kernelfüggvények.....	9
3.3	A sűrűség meghatározása.....	11
3.4	A nyomás meghatározása .....	12
3.4.1	Hagyományos SPH.....	12
3.4.2	WCSPH (Weakly Compressible SPH).....	13
3.4.3	PCISPH (Predictive-Corrective Incompressible SPH) .....	13
3.5	A külső és belső erők meghatározása .....	15
3.5.1	A nyomásból származó erő.....	15
3.5.2	A belső súrlódás.....	16
3.5.3	Külső erők számítása .....	18
3.6	Időbeli integrálás .....	18
3.7	Szomszédok keresése .....	20
4	A végelem-módszer (Finite Element Method - FEM) .....	21
4.1	A mozgásegyenlet megoldása .....	24
5	Kapcsolt FSI szimuláció (SPH - FEM) .....	24
6	A modellfejlesztés folyamata .....	25
6.1	Modellfejlesztés MATLAB környezetben.....	25
6.2	Sűrűségmodellek implementálása .....	26
6.3	Gátszakadás modellezése.....	27
6.4	Egy újfajta peremfeltétel, mint FSI feladat.....	29
7	Eredmények, következtetések .....	31
8	További feladatok, lehetőségek .....	32
9	Felhasznált irodalom .....	33

## 1 Bevezetés

Az elmúlt évtizedek számítógépek terén nyújtott fejlődésének köszönhetően, a már korábban kidolgozott elméleteken alapuló mérnöki számítások elérhetővé tették a műszaki fejlődés felgyorsítását, költségeinek csökkentését. Mára számos, numerikus módszereken alapuló szimulációs rendszert alkalmaznak a műszaki tervezések és fejlesztések terén. Az alkalmazott programok túlnyomórészt Euler szemléletén, végeselem-, végestérfogat-, illetve végesdifferencia-hálón alapuló modellek segítségével közelítik a parciális differenciálegyenletek egzakt megoldását [3, 14, 15, 16]. Ezek a módszerek sok esetben kielégítő hatékonysággal alkalmazhatók, ám számos korláttal rendelkeznek. Ilyen például a szilárdtest- mechanikában a végeselemes háló nagy torzulásával járó számítási hiba növekedése, az anyag szakadásának modellezése pedig gyakorlatilag lehetetlen. Áramlástan számítások esetében a véges térfogatok módszere szintén jelentős elméleti és gyakorlati problémát vet fel. Ezen a módszeren alapuló számítások során a turbulencia-modellek alkalmazása elkerülhetetlen, többfázisú, illetve nyílt felszínű áramlások modellezése pedig a függvények, fázishatárok mentén fellépő ugrásszerű megváltozásának leírása a tér- és időbeli felosztás nagysága miatt igen nagy számítási teljesítményt igényel [1].

Az SPH (Smoothed Particle Hydrodynamics) részecskéken alapuló hálófüggetlen numerikus módszer, melyet Robert A.Gingold és Joe J.Monaghan dolgozott ki 1977-ben [2]. Kezdetben csillagászati számításokra használták, és csak később kezdték áramlások modellezésére alkalmazni. A módszer lagrange-i szemléletéből kifolyólag igen nagy hatékonysággal alkalmazható többfázisú, nyílt felszínű áramlások modellezésére. Lényeges hátránya azonban az igen elterjedt végestérfogatok módszerével szemben a peremfeltételek kezelésének nehézsége. Dolgozatomban végeselemes módszeren alapuló peremfeltételek megvalósítására teszek kísérletet. Ennek segítségével az áramlás és a deformálható szilárd testek közötti kétirányú kapcsolat modellezése (FSI – Fluid-Structure Interaction) lehetővé válik, melyre jelentős igény mutatkozik mind építőmérnöki, mind gépészmérnöki területeken. A deformálható szilárd testek és folyadékok kölcsönhatásának modellezésével lehetőség nyílik szökőárak, hullámtörők, úszó testek viselkedésének megértésére [20].

Dolgozatomban az általam MATLAB környezetben fejlesztett kétdimenziós SPH és végeselemes modell működését, és az összekapcsolásukkal végzett FSI számításokat mutatom be.

## 2 Folyadékok mechanikája

Egy viszkozus, izoterm folyadék áramlásának alapvető fizikai jellemzői az  $\mathbf{u}$  sebesség, a  $\rho$  sűrűség és  $p$  nyomás, melyek a folyadék belsejében folytonos mennyiségek. Inkompresszibilis, viszkozus közeg – az ún. Stokes-féle folyadék –, mozgását a Navier–Stokes egyenlet (1) és a kontinuitási egyenlet (2) írja le,

$$\rho \left( \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \nabla \mathbf{u} \right) = -\nabla p + \mu \nabla (\nabla \mathbf{u}) + \mathbf{f}, \quad (1)$$

$$\frac{\partial \rho}{\partial t} + \nabla \rho \mathbf{u} = \mathbf{0}, \quad (2)$$

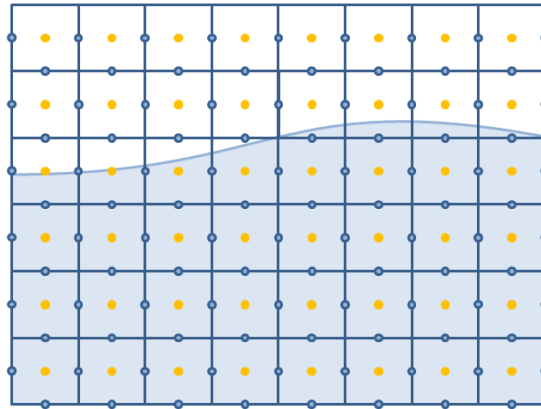
ahol  $\mu$  a viszkozitás,  $f$  a külső erők összegének fajlagos értéke,  $t$  pedig az idő [3]. A sebesség, és a külső erők reprezentációja vektormezőik segítségével történik, míg a sűrűség és a nyomás vektor-skalár függvény segítségével. A térbeli, háromdimenziós problémák helyett gyakran állítunk fel egyszerűsített, alacsonyabb dimenziós modelleket. Az (1–2) egyenletrendszer egy- és kétdimenzióban is érvényes marad, ha a benne szereplő mennyiségeket megfelelő fizikai dimenziójúnak választjuk. Általános  $D$  térbeli dimenzió esetén a  $\rho$  sűrűséget, a  $p$  nyomást, a  $\mu$  a dinamikai viszkozitást és az  $f$  erőssűrűséget  $\text{kg/m}^D$ ,  $\text{Nm}^{1-D}$ ,  $\text{Nm}^{2-D}$ s, illetve  $\text{N/m}^D$  SI mértékegységekkel fejezhetjük ki. A dolgozatban az SPH általános összefüggéseit a szabadon választható  $D$  térbeli dimenzióban írjuk fel, míg saját munkám és síkbeli modelljeim bemutatásánál csak a konkrét  $D=2$  esetre szorítkozom.

A Navier–Stokes-egyenleteknek csak nagyon speciális esetekben létezik analitikus megoldása, ezért általános esetben a folyadékok áramlásának vizsgálatára hagyományosan kísérleti, de újabban numerikus módszereket alkalmazunk. A numerikus modellezésben két lényegesen eltérő megközelítést különböztetünk meg, az Euler- és a Lagrange-féle szemléletet.

### 2.1 Az Euler-féle numerikus folyadékmodellelés

Az Euler-féle leíráson alapuló számítások szinte minden esetben hálólapúak, míg a Lagrange-féle esetben többnyire hálófüggetlen módszerekről beszélünk. Az Euler-féle megközelítés szerint a teret véges számú cellára osztjuk fel (1. ábra), melyekben a mezőváltozókat tároljuk (FVM: Finite Volume Method). Véges térfogatok módszere szerint a Navier–Stokes-egyenletek megoldása a cellák között felírt transzportegyenletek megoldásával történik [4]. Ahogy az ábrán látható, nyílt felszínű áramlások esetében a felszín elfogadható közelítéssel történő kidolgozása a cellák alakja miatt nehézkes, a felszín közelében a cellaméretet igen kicsinek kell választani a pontos eredmények elérése

érdekében. Természetesen többfázisú áramlások, illetve szakadozott, nyílt felszínű áramlások esetében gyakorlatilag a teljes víztér igen finom hálót igényel, ami a számítás tárigényét és hosszát is jelentősen megnöveli.

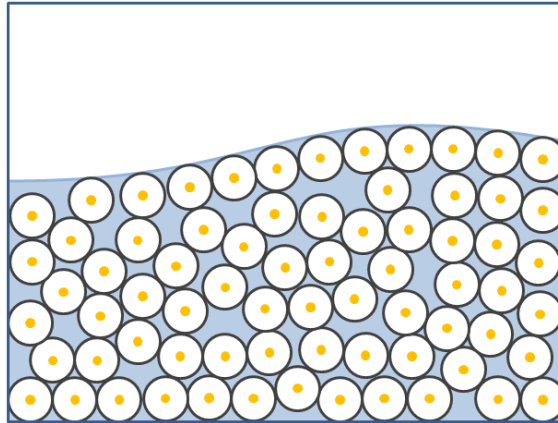


1. ábra: Euler féle diszkretizáció

A hálóalapú módszerek további igen fontos hátránya az, hogy a folyadék csak a hálón belül van értelmezve, ezért sok esetben – például nagy deformációk, törések esetén, amitől a víztér deformálódik – nehezen, vagy egyáltalán nem modellezhetők. A folyadék csak a háló által lefedett térben tud szabadon mozogni. Ennek a problémának a megelőzésére alkalmazható deformálódó, alkalmanként újragenerált háló, mellyel elérhető az, hogy a közeg változó geometriájú víztérben áramoljon [5].

## 2.2 A Lagrange-féle numerikus folyadékmodellelés

Az itt tárgyalt két, folyadékok mozgását modellező módszer legalapvetőbb eltérése a mezőváltozók tárolásának módjában keresendő. Ahogy említettem, az Euler-féle modellek a térben rögzített háló cellában tárolják a folyadék jellemzőit, míg egy Lagrange-féle módszer a folyadékot véges számú, a folyadékkal együtt mozgó részecske segítségével diszkretizálja. A modell sematikus felépítése az 2. ábrán látható.



2. ábra: Lagrange féle diszkretizáció

A részecskealapú folyadékmodellezés számos előnnyel rendelkezik az Euler-féle modellekkel szemben, ilyen a bevezetésben már említett többfázisú és nyílt a felszínű áramlások vizsgálata. Fontos azonban tudni, hogy az euléri szemléleten alapuló szimulációs rendszerek elterjedése a részecske-alapú módszerek néhány sarkalatos nehézsége miatt alakult ki. A Lagrange-i szemléletre alapozott numerikus módszerekkel igen nehezen érhető el az műszaki megoldások szempontjából fontos inkompresszibilitás, és a fal peremfeltételek, források, nyelők realiztikus definiálása szintén komoly nehézségekbe ütközik.

A Lagrange-féle szemléletben a Navier—Stokes-egyenlet jelentősen egyszerűbb formában írható fel, erre példát a 3.6-os alfejezetben fogunk látni. Mivel a részecskék száma, és tömege az általam elvégzett szimulációk során változatlan, ezért a kontinuitási egyenlet automatikusan teljesül.

### 3 Az SPH – Smoothed Particle Hydrodynamics

Az SPH módszer teljes mértékben hálófüggetlen, Lagrange-féle módszer [6]. Kezdetben az asztrofizikában alkalmazták – kitalálója, Joe J. Monaghan is asztrofizikus [2] – csillagközi gázok viselkedésének modellezésére, azonban a módszer a diszkretizáció jellegéből adódóan igen jól alkalmazható összenyomható közegek áramlásának modellezésére. Az SPH segítségével folytonos függvények, és deriváltjaik értéke diszkrét pontok felhasználásával közelíthető. Ezen pontok rendelkeznek bizonyos tulajdonságokkal, melyeket magukkal szállítanak – például sűrűség, nyomás, tömeg stb. –, valamint térbeli elhelyezkedésük és aktuális sebességük minden időpillanatban ismert. Egy adott részecske a tulajdonságait a szomszédos részecskék pozíciójából és azok tulajdonságai alapján kapja. Ellentétben az Euler szemléletű módszerekkel, ahol egy függvény differenciáljának kiszámításához egy előre definiált rácsra van szükség, melynek csomópontjaiban ismert helyettesítési értékek alapján közelítjük a deriváltakat, addig az SPH módszer ezt a közelítést analitikus deriválás

segítségével végzi el tetszőlegesen elhelyezkedő részecskék segítségével. Ezen közelítés pontossága természetesen a részecskék sűrűségének növelésével érhető el [5].

### 3.1 Az SPH alapjai

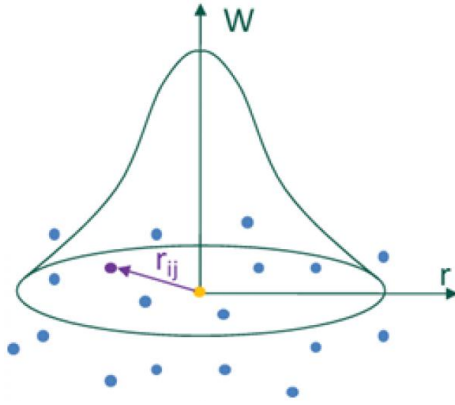
Az SPH egy interpolációs módszer. Alapja az integrál-interpoláció, melyet ún. kernel- (mag-) függvények segítségével valósítunk meg. A kernelfüggvények Dirac delta függvényét közelítik, egyéb tulajdonságaikra dolgozatomban későbbi szakaszában térek rá. Egy tetszőleges  $A(\mathbf{r})$  függvény integrál-interpoláltja egy  $\Omega$  tartományon a következő:

$$A_I(\mathbf{r}) = \int_{\Omega} A(\mathbf{r}') W(\mathbf{r} - \mathbf{r}', h) d\mathbf{r}', \quad (3)$$

ahol  $\mathbf{r}$  egy  $\Omega$ -n belüli pont,  $W$  a simító kernelfüggvény,  $h$  pedig a kernelfüggvény sugara. Ez az érték határozza meg a kernel, tehát a simítás *hatósugarát*. A (3) kifejezés numerikus megfelelője az integrál diszkrét közelítéseként a következőképpen írható:

$$A_S(\mathbf{r}) = \sum_j A_j \frac{m_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h), \quad (4)$$

ahol  $j$  a szomszédos, azaz a hatósugáron belüli, részecskék indexei,  $m_j$  és  $\rho_j$  a  $j$ -edik részecske tömege és sűrűsége,  $A_j$  pedig a függvény értéke a  $j$ -edik részecskében [6]. A (4) egyenlet tehát egy tetszőlegesen választott részecske helyén meghatározza az  $A$  függvény értékét úgy, hogy a tőle  $h$  távolságon belül lévő részecskékhez tartozó  $A$  értékeket a kernel függvénnyel súlyozva megszorozza a saját térfogatukkal, majd ezen értékeket összegzi valamennyi részecskére a hatósugáron belül (3. ábra).



3. ábra: A kernel függvénnyel súlyozott szomszéd részecskék kezelése [7]

Áramlások modellezése során azonban sokszor szükség van skalárfüggvények térbeli deriváltjaira. Az SPH módszer kényelmessége az, hogy a deriváltak előállítására igen egyszerű, amiatt, hogy a deriválás a függvényről átvihető a kernelfüggvényre [5]:

$$\begin{aligned}
 \nabla A_s(\mathbf{r}) &= \nabla \sum_j A_j \frac{m_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h) = \\
 &= \sum_j A_j \frac{m_j}{\rho_j} \nabla W(\mathbf{r} - \mathbf{r}_j, h)
 \end{aligned}
 \tag{5}$$

és

$$\begin{aligned}
 \nabla^2 A_s(\mathbf{r}) &= \nabla^2 \sum_j A_j \frac{m_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h) = \\
 &= \sum_j A_j \frac{m_j}{\rho_j} \nabla^2 W(\mathbf{r} - \mathbf{r}_j, h)
 \end{aligned}
 \tag{6}$$



### 3.2 Simító kernelfüggvények

Többféle kernelfüggvény létezik, a különböző kernelfüggvények használata a véges differencia módszer esetében a különböző differenciálsémák alkalmazásával analóg. Egy kernelfüggvény számos kritériumnak kell, hogy eleget tegyen. A kernelfüggvény deriválható kell, hogy legyen a  $[0, h]$  intervallumon, illetve teljesülnie kell a következő egyenlőségeknek [5]:

$$\int_r W(\mathbf{r}, h) d\mathbf{r} = 1, \quad (7)$$

$$\lim_{h \rightarrow 0} W(\mathbf{r}, h) = \delta(\mathbf{r}), \quad (8)$$

$$W(\mathbf{r}, h) = 0, \text{ ha } \|\mathbf{r}\| > h, \quad (9)$$

$$W(\mathbf{r}, h) = W(-\mathbf{r}, h), \quad (10)$$

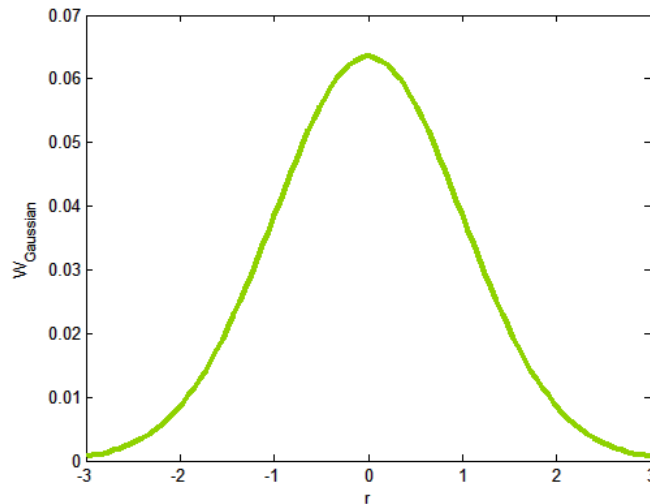
$$W(\mathbf{r}, h) \geq 0, \quad (11)$$

ahol  $\delta(\mathbf{r})$  a Dirac-delta:

$$\begin{aligned} \delta(\mathbf{r}) &= 0 \quad \text{ha } \|\mathbf{r}\| > 0 \\ \int_{-\infty}^{\infty} \delta(\mathbf{r}) d\mathbf{r} &= 1 \end{aligned} \quad (12)$$

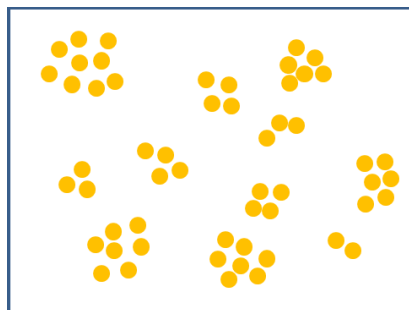
Az említett kritériumoknak eleget tesz, és igen jó matematikai tulajdonságokkal rendelkezik például a Gauss-görbe:

$$W_{Gaussian}(\mathbf{r}, h) = \frac{1}{(2\pi h^2)^{\frac{D}{2}}} e^{-\|\mathbf{r}\|^2 / 2h^2}, \quad h > 0, \quad (13)$$



4. ábra: Gauss kernel, ha  $h=1$

Bár igen nagy pontosság érhető el ezen Gauss-kernel segítségével (4. ábra), a gyakorlatban mégsem előnyös. Ennek oka az, hogy hatósugara végtelen, így számítási igénye igen nagy, amit tovább növel a függvény exponenciális jellege. Egy 10 000 részecskéből álló modellben, átlagosan 20 szomszédos részecskével számolva 1 000 időlépéshez, a kernel függvény helyettesítési értékének kiszámítását 20 millió alkalommal kell elvégeznünk! Ebből kifolyólag a kernelfüggvények egyszerűsítése nagy mértékben járul hozzá a számítási idő csökkentéséhez. A Gauss-kernel másik problémája, hogy a  $[0, h]$  tartományon inflexiós pontja van, amely a deriválást követően szélsőértékként jelenik meg. Emiatt a részecskék – ha egymáshoz nagyon közel kerülnek – a szélső értéket átlépve azt tapasztalják, hogy a kérdéses részecskével szemben egyre kisebb mértékű kölcsönhatásban vannak. Ez természetesen nem valós, ezért a számítások nagy nyomás esetén hibásak lehetnek (ez az ún. particle clustering – részecskefürtök kialakulása), ezt mutatja be a 5. ábra.



5. ábra: A részecskék „fürtösödése” instabilitási probléma nagy nyomások fellépésekor

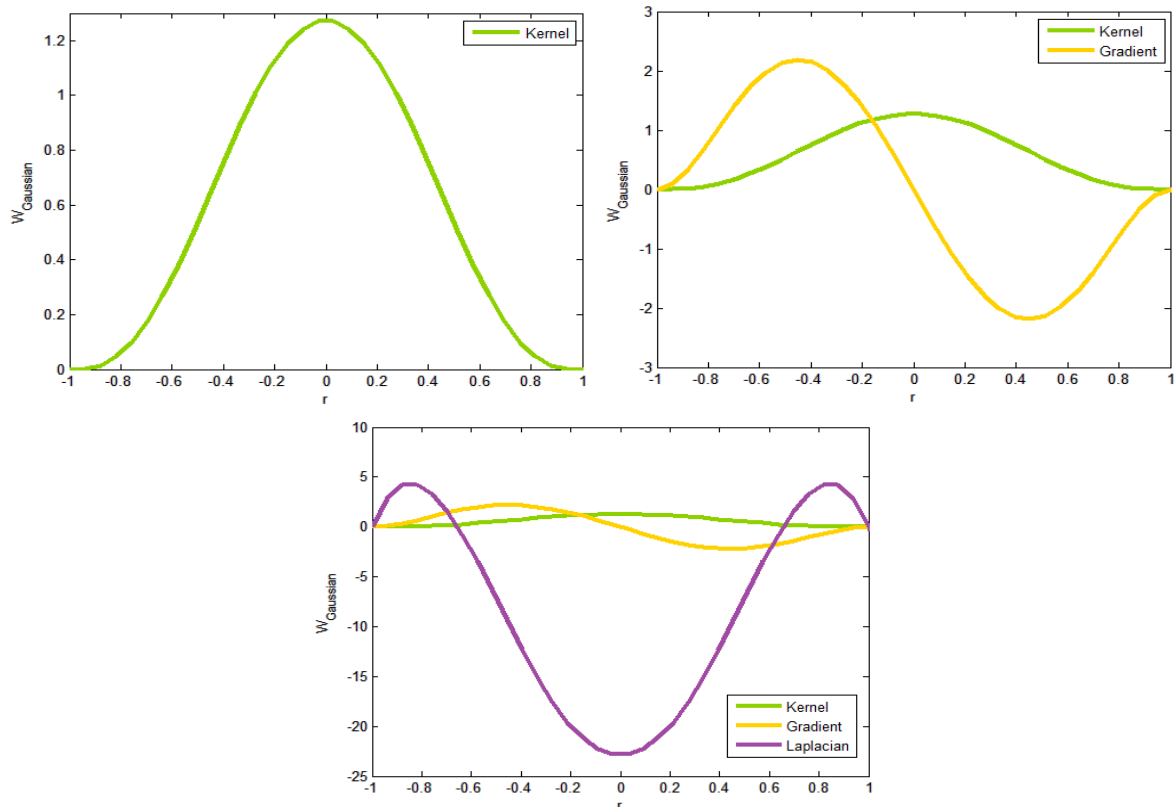
A Gauss-kernel alkalmazásából fakadó nagy számítási igény csökkentésére, és instabilitási problémák elkerülésére számos numerikus szempontból előnyösebb kernelfüggvényt javasoltak. A következő alfejezetben bemutatom az SPH módszer működését és az általam választott kernelfüggvényeket. Itt érdemes megjegyezni, hogy a (7) normálási feltétel miatt fellépő normálási tényező a (13) Gauss-kernelben függ a geometriai tér  $D$  dimenziójától. Így van ez a többi kernel esetén is, de a modellemben felhasznált kernelek esetén csak a  $D=2$  dimenziós normálási tényezőket adom majd meg.

### 3.3 A sűrűség meghatározása

Az  $i$ -edik részecske sűrűségének meghatározása a (4) egyenlet segítségével a következőképpen történik:

$$\rho_i(\mathbf{r}) = \sum_j \rho_j \frac{m_j}{\rho_j} W_{poly6}(\mathbf{r} - \mathbf{r}_j, h) = \sum_j m_j W_{poly6}(\mathbf{r} - \mathbf{r}_j, h), \quad (14)$$

ahol  $W_{poly6}$  a Gauss-kernel egy polinomiális közelítése. A 6. ábra mutatja a *poly6* kernelt és első két deriváltját. Ez a kernel lényegesen gyorsabb futást eredményez, mert a számítógép könnyebben kezeli a polinomiális függvényeket az exponenciális függvényeknél.



6. ábra: A poly6 kernel és első két deriváltja

Ez a kernelfüggvény és deriváltai a következőképpen adható meg [5]:

$$W_{poly6}(\mathbf{r}, h) = \frac{4}{\pi h^8} (h^2 - |\mathbf{r}|^2)^3, \quad (15)$$

$$\nabla W_{poly6}(\mathbf{r}, h) = -\frac{24}{\pi h^8} \mathbf{r} (h^2 - |\mathbf{r}|^2)^2, \quad (16)$$

$$\Delta W_{poly6}(\mathbf{r}, h) = -\frac{48}{\pi h^8} (h^2 - |\mathbf{r}|^2)(3h^2 - 7|\mathbf{r}|^2). \quad (17)$$

### 3.4 A nyomás meghatározása

A nyomás számításánál lényegesen eltérő megoldásokkal találkozhatunk. Ettől eltekintve a különböző modellváltozatok azonos módszerekkel operálnak. A következőkben három eltérő módszert fogok ismertetni a nyomás meghatározására.

#### 3.4.1 Hagyományos SPH

A sűrűség ismeretében a nyomás meghatározható. Hagyományos esetben a nyomás meghatározása az ideális gáz állapotegyenletének segítségével történik:

$$\begin{aligned} pV &= nRT \\ p &= k\rho \end{aligned} \quad (18)$$

ahol  $p$  a nyomás,  $V$  a térfogat,  $n$  a gáz mólszáma,  $R$  a gázállandó,  $T$  a hőmérséklet,  $\rho$  a sűrűség,  $k$  pedig a gáz merevségét reprezentáló konstans. A (18) minden esetben pozitív nyomást fog eredményezni minden részecske esetében, ami a szokásos gázok esetén fizikailag helytálló, azonban a cseppfolyós fázis esetében kohéziós hatások eredményeként felléphetnek vonzó irányú erők, tehát negatív nyomások. Ennek figyelembe vételére számítására a következő, módosított ideális gáz-állapotegyenlet használható [5]:

$$\begin{aligned} (p + p_0)V &= k \\ p + k\rho_0 &= k\rho \\ p &= k(\rho - \rho_0) \end{aligned} \quad (19)$$

ahol  $p_0$  és  $\rho_0$  a nyugalmi nyomás és sűrűség,  $V$  pedig a fajtérfogat. Ebben a módosított – folyadék nyugalmi állapota körül linearizált – gáz-állapotegyenletben a  $k$  merevség tetszőleges növelésével segíthetjük a folyadékot a konstans sűrűségeloszlású állapotba, tehát az inkompesszibilitás irányába terelni.

### 3.4.2 WCSPH (Weakly Compressible SPH)

A WCSPH módszer a nyomás meghatározásában tér el a hagyományos SPH módszertől. A módosítás célja a nagyobb merevségű folyadék elérése. Ennek megvalósítására használható az ún. Tait-egyenlet [8]:

$$p = B \left( \left( \frac{\rho}{\rho_0} \right)^\gamma - 1 \right), \text{ ahol } B = \frac{\rho_0 c_s^2}{\gamma}, \gamma = 7 \quad (20)$$

és  $c_s$  a közegben terjedő lökéshullámok sebessége. A közeg merevsége  $\gamma$  értékének növelésével fokozható, ezért a folyadék merevsége igen nagy lehet, mert a nyomás az aktuális sűrűség, és a nyugalmi sűrűség hányadosának jelen esetben a hetedik hatványával arányos. Ezen módszer esetében azonban, a nagy merevség következtében az időlépést igen alacsonyra kell választanunk [8].

### 3.4.3 PCISPH (Predictive-Corrective Incompressible SPH)

Nagyobb időlépés, numerikusan stabil számítás mellett a PCISPH [9] módszer segítségével érhető el gyakorlatilag inkompesszibilis áramlások esetére. A nyomások kiszámítása a PCISPH módszer esetében lényegesen eltér az SPH és a WCSPH módszerektől. Míg az említett két módszer gáz-állapotegyenletek segítségével határozza meg a részecskék nyomásait, addig a PCISPH nem használ explicit állapotegyenletet, ehelyett a nyomást iteratív úton határozza meg a célzott konstans sűrűség eléréséhez.

Az inkompesszibilitás lényege, hogy a közeg sűrűségének eloszlása homogén. Ezt a kritériumot felhasználva, a részecskék sűrűségének a névleges állapottól való eltérése alapján határoz meg iteratív úton olyan nyomáseloszlást, amely a közeget az egyenletes sűrűségeloszlás irányába tereli [9].

Az alábbi algoritmus sematikusan mutatja be a PCISPH működését, és összehasonlítását a hagyományos, SPH és a WCSPH módszerekkel [9].

---

**Algorithm 1** SPH / WCSPH

---

```
1 while animating do
2   for all i do
3     find neighborhoods  $N_i(t)$ 
4   for all i do
5     compute density  $\rho_i(t)$ 
6     compute pressure  $p_i(t)$ 
7   for all i do
8     compute forces  $\mathbf{F}^{p,v,g,ext}(t)$ 
9   for all i do
10    compute new velocity  $\mathbf{v}_i(t+1)$ 
11    compute new position  $\mathbf{x}_i(t+1)$ 
```

---

---

**Algorithm 2** PCISPH

---

```
1 while animating do
2   for all i do
3     find neighborhoods  $N_i(t)$ 
4   for all i do
5     compute forces  $\mathbf{F}^{v,g,ext}(t)$ 
6     initialize pressure  $p(t) = 0.0$ 
7     initialize pressure force  $\mathbf{F}^p(t) = 0.0$ 
8   while ( $\rho_{err}^*(t+1) > \eta$ ) || ( $iter < minIterations$ ) do
9     for all i do
10      predict velocity  $\mathbf{v}_i^*(t+1)$ 
11      predict position  $\mathbf{x}_i^*(t+1)$ 
12     for all i do
13      predict density  $\rho_i^*(t+1)$ 
14      predict density variation  $\rho_{err}^*(t+1)$ 
15      update pressure  $p_i(t) += f(\rho_{err}^*(t+1))$ 
16     for all i do
17      compute pressure force  $\mathbf{F}^p(t)$ 
18   for all i do
19     compute new velocity  $\mathbf{v}_i(t+1)$ 
20     compute new position  $\mathbf{x}_i(t+1)$ 
```

---

A PCISPH a WCSPH alacsony időlépés miatti nagy számításigényét igyekszik orvosolni. Algoritmus szerint a részecskék helyzetét a nyomásból származó erők elhanyagolásával (7. sor) időben ideiglenesen előrevetítjük a hagyományos SPH eljárását követve (10, 11. sor), majd a kialakult konfiguráció sűrűségeloszlását meghatározzuk (13. sor). A kapott sűrűségeloszlás minden részecske esetében eredményez egy sűrűséghibát (14. sor): a nyugalmi, azaz a kívánt sűrűségtől való eltérést. A kapott hiba ismeretében meghatározható az a részecskékre ható nyomásgradiens, amellyel a sűrűség hibáját le lehetne csökkenteni (15–17. sor). Ezt a nyomást korrekcióként figyelembe vesszük. Ezt követően, már a kialakult nyomásmező figyelembevételével ismét kiszámoljuk a részecskék új pozícióját, amellyel a számítást megismételjük. Ezt iteratív eljárásként addig ismételjük, amíg a sűrűség hibájának az összes részecskére vett maximális értéke kisebbé nem válik egy általunk definiált  $\eta$  küszöbértéknél. Ez utóbbi az inkompresszibilitás mértékével áll kapcsolatban, pl. 2%-os összenyomhatóság és  $1000 \text{ kg/m}^2$  esetén  $\eta=20 \text{ kg/m}^2$ . A minimális iterációk száma [9] szerint háromra választandó, hogy a szimuláció stabil legyen. A nyomás meghatározására szolgáló összefüggés a PCISPH módszer legnehezebb része. Az algoritmus 15. sorának részletesebb alakja:

$$\tilde{p}_i = \tilde{p}_i + \delta \rho_{err}^* \quad (21)$$

ahol:

$$\delta = \frac{\rho_0^2}{2\Delta t^2 m^2 (\sum_j \nabla W_{ij} \sum_j \nabla W_{ij} + \sum_j \nabla W_{ij} \nabla W_{ij})}, \text{ ahol } \nabla W_{ij} = \nabla W(|\mathbf{r}_i - \mathbf{r}_j|, h). \quad (22)$$

A (22) részletes levezetése a [9] cikkben olvasható. A nyomáseloszlás az iteráció során összegződik (2. algoritmus, 15. sor), így a részecskék sűrűsége, megfelelő paraméterezés

esetén, az egyenletes sűrűségeloszlás felé konvergál. A PCISPH előnye az SPH-val és a WCSPH-val szemben az, hogy lényegesen nagyobb időlépéssel képes numerikusan stabil számításokat eredményezni, emellett a közeg összenyomhatósága sokkal jobban tartható.

A [10] szerint az időlépés akár két nagyságrenddel(!) is nagyobb lehet, mint a WCSPH esetében, a számítási igény időlépésenként azonban nem sokkal haladja meg a WCSPH esetében szükséges számításokét. Emiatt a számítási idő lényegesen alacsonyabb [10].

### 3.5 A külső és belső erők meghatározása

#### 3.5.1 A nyomásból származó erő

A nyomásból származó belső erő kiszámítható, ha ismerjük a nyomáseloszlást a közegben, vagyis ha minden részecske nyomása ismert, akkor (5) alapján:

$$f_i^p(\mathbf{r}) = \sum_{j \neq i} p_j \frac{m_j}{\rho_j} \nabla W(\mathbf{r} - \mathbf{r}_j, h). \quad (23)$$

A fenti kifejezés azonban inkorrekt eredményhez vezet. Ennek oka két részecske esetén szemléltethető. Az első részecskére ható erő csak a másik részecske nyomásától függ. Mivel a szomszédos részecskék esetében a nyomás általános esetben nem egyezik meg, ezért a nyomásból származó erő aszimmetrikus lesz, és nem teljesül Newton harmadik törvénye. Ennek elkerülésére a következő – szimmetrizált – kifejezés használható [5]:

$$f_i^p(\mathbf{r}) = -\rho_i \sum_{j \neq i} \left( \frac{p_i}{\rho_i} + \frac{p_j}{\rho_j} \right) m_j \nabla W_{spiky}(\mathbf{r} - \mathbf{r}_j, h). \quad (24)$$

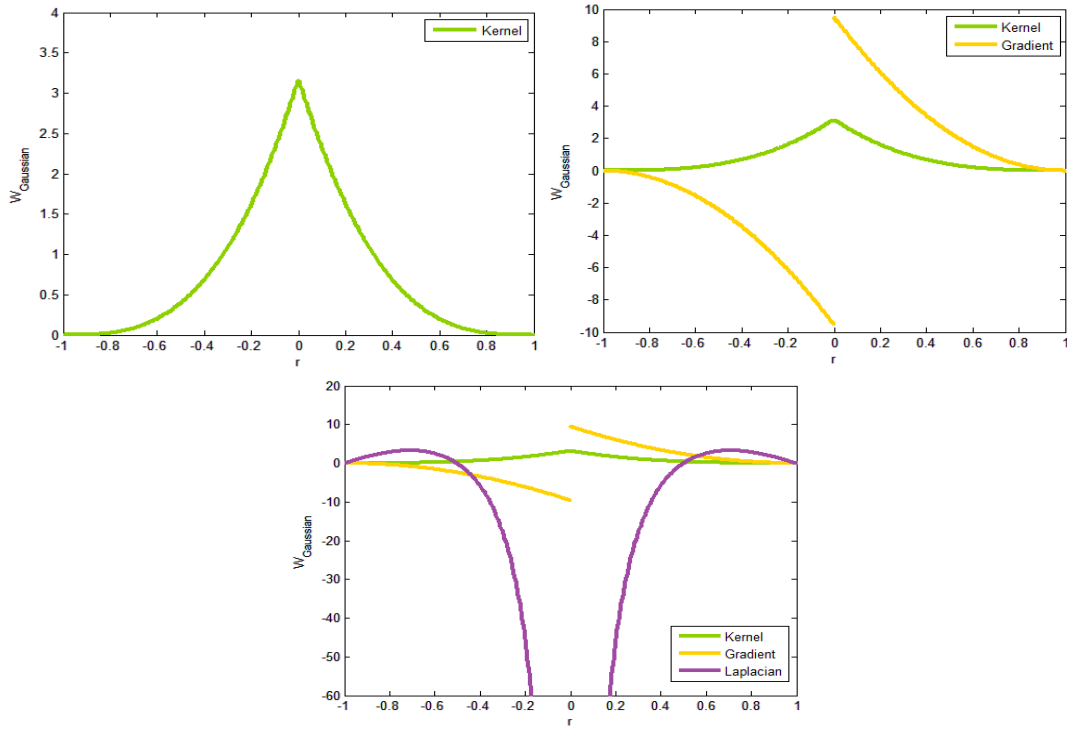
Itt már azt a kernelfüggvényt írtam fel, melyet a nyomásból származó erők korrigálásához használtunk [5]:

$$W_{spiky}(\mathbf{r}, h) = \frac{10}{\pi h^5} (h - |\mathbf{r}|)^3, \quad (25)$$

$$\nabla W_{spiky}(\mathbf{r}, h) = -\frac{30}{\pi h^5} \frac{\mathbf{r}}{|\mathbf{r}|} (h - |\mathbf{r}|)^2, \quad (26)$$

$$\Delta W_{spiky}(\mathbf{r}, h) = -\frac{60}{\pi h^5} (h - |\mathbf{r}|)(h - 2|\mathbf{r}|). \quad (27)$$

A 7. ábra mutatja ezt a *spiky* – azaz hegyes – kernelfüggvényt és első két deriváltját.



7. ábra: A *spiky* kernel és első két deriváltja

Az ábrákon látható, hogy ennek nincs inflexiós pontja, ezért a deriváltjának sincs szélsőértéke. Ebből kifolyólag a részecskék fűrtösödése még nagy nyomások esetén sem jelentkezik, és megakadályozható a 3.2-es alfejezetben említett instabilitás.

### 3.5.2 A belső súrlódás

A valós folyadékokban fellépő másik belső erő a viszkozitás. Ez az erő a folyadék rész deformációjának sebességétől függ, és disszipatív jellegű. Numerikus szempontból tehát előnyös a viszkozitás figyelembe vétele, mert minden esetben stabilizálja a számítást. Az SPH-ban a viszkozitás a szomszédos részecskék relatív sebességétől függ [5]:

$$f_i^v(\mathbf{r}) = -\mu \sum_{j \neq i} (u_j - u_i) \frac{m_j}{\rho_j} \nabla^2 W_{viscosity}(\mathbf{r} - \mathbf{r}_j, h), \quad (28)$$



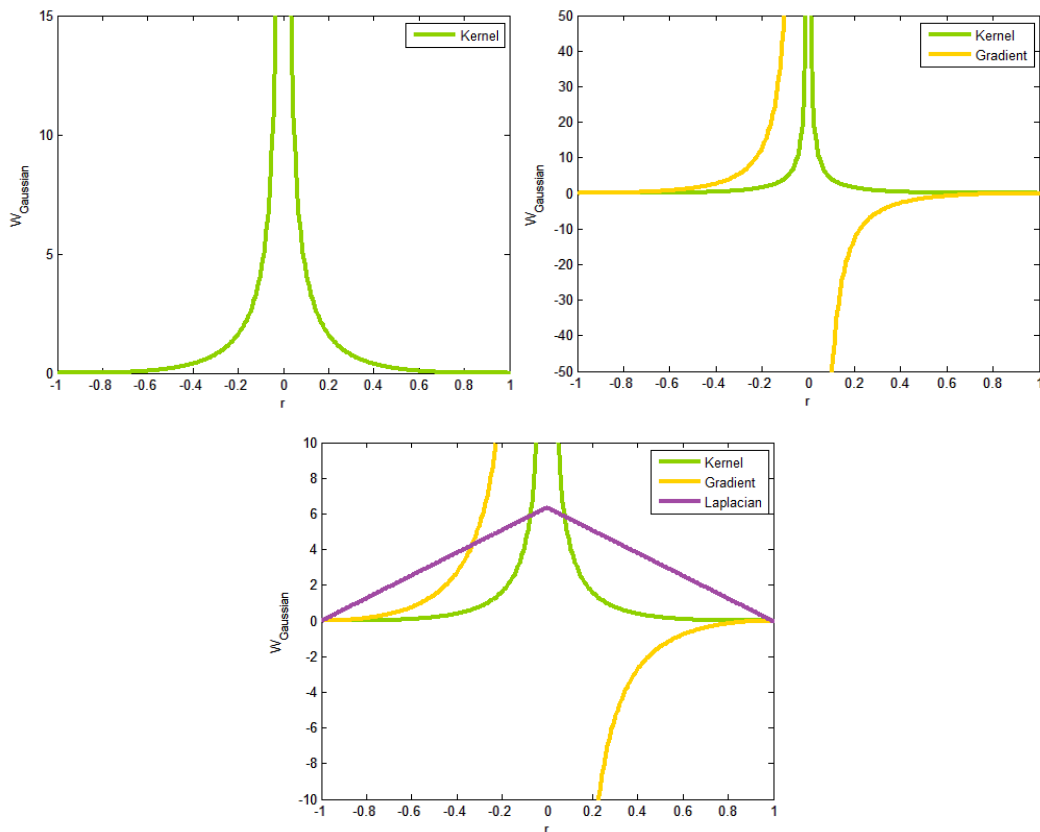
Modellfuttatásainkhoz [5] alapján a viszkozitás modellezéséhez az alábbi kernelfüggvényt választottuk:

$$W_{viscosity}(\mathbf{r}, h) = -\frac{10}{3\pi h^2} \left( -\frac{|\mathbf{r}|^3}{2h^3} + \frac{|\mathbf{r}|^2}{h^2} + \frac{h}{2|\mathbf{r}|} - 1 \right), \quad (29)$$

$$\nabla W_{viscosity}(\mathbf{r}, h) = -\frac{10}{3\pi h^2} |\mathbf{r}| \left( -\frac{3|\mathbf{r}|}{2h^3} + \frac{2}{h^2} + \frac{h}{2|\mathbf{r}|^3} \right), \quad (30)$$

$$\Delta W_{viscosity}(\mathbf{r}, h) = -\frac{40}{\pi h^4} (h - |\mathbf{r}|). \quad (31)$$

A  $W_{viscosity}$  kernel és első két deriváltja a 8. ábrán látható:



8. ábra: A viszkozitás kernel és első két deriváltja

### 3.5.3 Külső erők számítása

A közegben ébredő belső erőkön kívül az áramlást külső erők is befolyásolják. Ezek a külső erők a gravitáció, a falakkal való kölcsönhatás és a szabad felszínen fellépő felületi feszültség. Dolgozatomban a felületi feszültséget elhanyagoltam.

A *gravitáció* figyelembe vétele az  $i$ -edik részecskére egészen egyszerű [5]:

$$f_i^g = \rho_i g \quad (32)$$

Mint említettem, a *fali erők* definiálása az SPH modellekben komoly problémát okoz. Dolgozatomban a merev falakat a számítások mindegyikében rögzített, ún. ghost – azaz „kísértet” – részecskék segítségével oldottam meg. Ezek a részecskék ugyanolyan tulajdonságokkal rendelkeznek, mint a folyadékot alkotó részecskék, azonban helyzetük időben állandó, sebességük minden időpillanatban nulla [11].

### 3.6 Időbeli integrálás

A folyadék mozgásának időbeli modellezéséhez első lépésként ki kell számolnunk minden részecske gyorsulását, valamennyi időlépésben. A gyorsulások meghatározása az alábbi módon történik [5]:

$$\begin{aligned} \dot{\mathbf{r}}_i &= \mathbf{u}_i, \\ \dot{\mathbf{u}}_i &= \mathbf{a}_i \equiv \frac{1}{\rho_i} \mathbf{F}_i', \end{aligned} \quad (33)$$

ahol  $\mathbf{F}_i$  az  $i$ -edik részecskére ható külső és belső erőssűrűségek – nyomásból (24), viszkozitásból (28), gravitációs gyorsulásból (32) és fali hatásokból származó erőssűrűségek – összege. Az egyenletek kiszámításához természetesen szükség van a részecskék sűrűségére is, melyeket a (14) alapján határozzunk meg minden időlépésben. Ez az euleri (1–2) parciális differenciálegyenlet-rendszer helyébe tehát (33, 14) közös differenciálegyenlet-rendszer lép. A lagrange-i leírás tehát matematikailag valóban egyszerűbb struktúrájú.

Az általam választott, és kielégítő pontossággal és stabilitással működő, mégis igen egyszerű módszer a fél-implicit Euler-féle megoldóséma:

$$\mathbf{u}_{t+\Delta t} = \mathbf{u}_t + \Delta t \mathbf{a}_t \quad (34)$$

$$\mathbf{r}_{t+\Delta t} = \mathbf{r}_t + \Delta t \mathbf{u}_{t+\Delta t} \quad (35)$$

Ez a sebességet a sűrűség segítségével explicit módon határozza meg, azonban a következő időpillanatbeli pozíció számításához ezt a sebességet már fel tudjuk használni, így a pozíciókat már impliciten számoljuk.

Ahogy a 3.4.2-es alfejezetben említettem, a WCSPH a folyadék nagy merevsége miatt igen kicsiny időlépést követel meg. Ez a gyakorlatban megnehezíti az alkalmazhatóságát. Ezen a problémán *adaptív időlépés* alkalmazásával segíthetünk. Az adaptív időlépés alapfeltevése az, hogy a számítás csak abban az esetben maradhat numerikusan stabil, ha a részecskék egy időlépés alatt megtett úthossza lényegesen kisebb, mint a hatósugaruk hossza [8]. Ezzel elkerülhető a részecskék egymás közötti és falakkal való hatásainak hibás számítása. Ha ugyanis az időlépés nagysága megengedné a részecskék hatósugárnál nagyobb távolság megtételét egy időlépésen belül, akkor előfordulhatna olyan eset, hogy egy részecske egy adott időlépésben nincs kapcsolatban azzal a fallal, amelyik felé halad, a következő időlépésben pedig már áthaladt rajta. Legrosszabb esetként két részecske szemből történő ütközése tekinthető, ilyen esetben a sebességek összeadódnak, tehát az egy időlépés alatt maximálisan engedhető út a hatósugár felénél kisebb kell, hogy legyen. Az adaptív időlépés kritériumainak felállítása ennek megfelelően:

$$s = \frac{a}{2} \Delta t^2 + v_0 \Delta t < \lambda \cdot \Delta t, \quad (36)$$

ahol  $s$  az egy időlépés alatt megtett út hossza,  $a$  a gyorsulás,  $v_0$  a sebesség az aktuális időlépésben,  $\Delta t$  pedig az időlépés nagysága,  $\lambda=1/2$ . A gyorsulás helyére (33)-et helyettesítve:

$$s = \frac{F}{2\rho_0} \Delta t^2 + v_0 \Delta t < \lambda \cdot \Delta t. \quad (37)$$

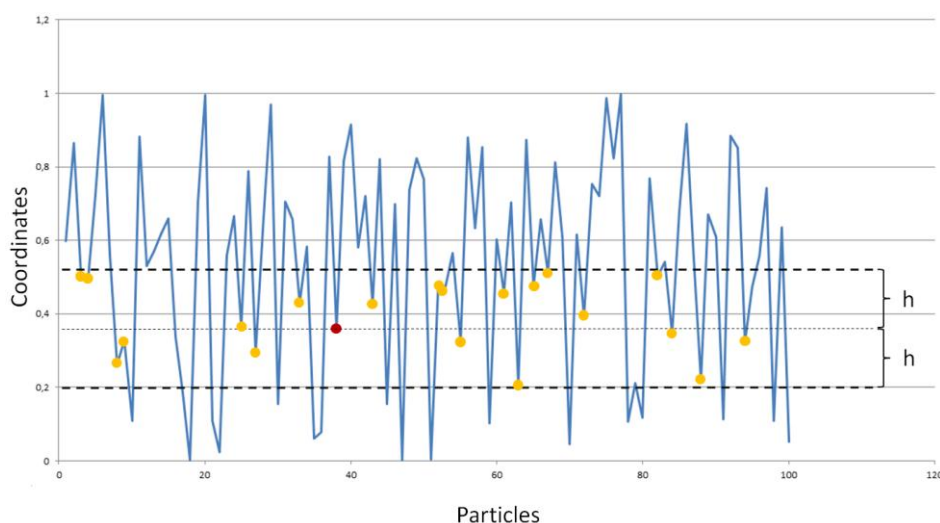
A (37) fizikai tartalma az, hogy a részecske egy időlépésben megtett úthossza két részből tevődik össze, az egyik a sebesség az időlépés elején, amely  $v_0 \Delta t$  elmozdulást idézne elő, ha nem hatna erő a részecskére, a másik tag pedig a gyorsulásból származó sebességváltozást veszi figyelembe. Az adaptív időlépés kritériuma tehát a két tag összegére vonatkozik. Ez a következő Courant–Freidrich–Levy (CFL) számhoz vezet [12]:

$$\Delta t = \min \left( \lambda_v \frac{h}{v_{\max}}, \lambda_F \sqrt{\frac{h \rho_0}{F_{\max}}} \right), \quad (38)$$

ahol  $\lambda_v$  és  $\lambda_F$  konstansok, melyek az időlépés nagyságát közvetlenül befolyásolják. Értékükre [13] a  $\lambda_v=0.4$ ,  $\lambda_F=0.25$  értékeket javasolja. A számítások során mi is ezeket az értékeket használtuk.

### 3.7 Szomszédok keresése

Az SPH módszer elméleti háttéréből fakadóan, egy adott részecske mozgásának számításához, minden, vele kapcsolatban álló más részecske hatását figyelembe kell vennünk. Az a megoldó, amelyik minden részecske hatását megvizsgálja egy adott részecskére, rendkívül sok számítást végez feleslegesen. Ez a számításigény a szimuláció futási sebességét drasztikusan rontja, ami gyakorlati szempontból a megoldót használhatatlanná teszi. Emiatt a szomszédok keresése az egyik legsarkalatosabb része az SPH megoldóknak. A probléma kezelésére igen sok megoldás született, [5] szerint a részecskék hash táblával történő kezelése nagyon hatékonyan növeli a számítás gyorsaságát. Dolgozatomban én egy ennél lényegesen egyszerűbb szomszédkeresési módszert alkalmaztam. A részecskék koordinátáit  $x$  és  $y$  vektorokban tároltam. A vektorok bármely elemére meg tudom határozni, hogy az a részecske egy adott részecskétől  $x$  és  $y$  szerint  $h$  távolságon belül van-e, ez a teszt egy logikai igaz vagy hamis értéket ad eredményül. A tesztet minden részecskére elvégezve  $x$  és  $y$  koordináták szerint két bináris vektort kapok, melyekben az egyes azt jelenti, hogy az adott indexű elem a kérdéses részecske potenciális szomszédja (9.ábra).

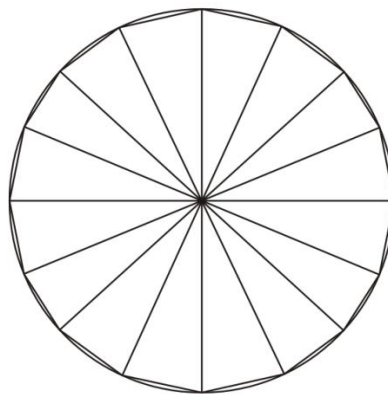


9. ábra: Szomszédok keresésének sematikus ábrázolása

A két vektoron elemenként elvégezve az ésműveletet, egy harmadik vektort kapok, melyben már csak azon részecskék szerepelnek egyessel, melyek a kérdéses részecske hatókörét befoglaló  $2h$  oldalélű négyzetbe esnek. Ezen részecskék ugyan nem mind tényleges szomszédok, de így a számítás során, egy részecske esetén végigsöpört részecskék számát nagy mértékben csökkenthettem. Ez az eljárás kihasználja a MATLAB környezet magas szintű rutinjait, és az általam végzett kétdimenziós számítások esetében lényegesen hatékonyabbnak bizonyult a szakirodalomban szokásos egyéb szomszédkeresési algoritmusoknál.

#### 4 A végeelem-módszer (Finite Element Method - FEM)

A végeelem-módszer alapötlete az ókorból származik. A matematika fejlődésének korai szakaszában az összetett, magas szintű analitikus módszerek ismerete nélkül, egyszerű műveletek segítségével közelítették a keresett mennyiség egzakt értékét [14]. Egy egyszerű példát mutat a 10. ábra.



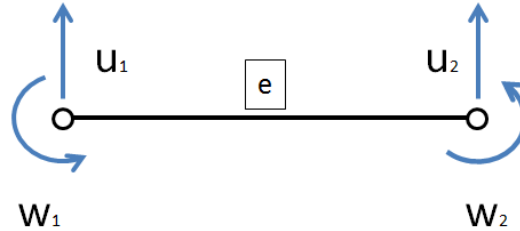
10. ábra: A kör területének közelítő számítása háromszögek segítségével [14]

Az ismeretlen területű kör felosztása egyszerű módszerrel számítható véges számú területre, lehetővé teszi a teljes terület, közelítő értékének kiszámítását  $\pi$  ismerete nélkül. A feladat így egyszerű számítások sorával megoldható, a közelítés pontossága pedig a részterületek számának növelésével minden határon túl növelhető.

A modern szilárdságtani végeelemes módszerek a fent említett módszerhez hasonlóan egy összetett mechanikai feladat megoldását – az analitikus módszerek bonyolultságát megkerülve – igen nagyszámú, de egyszerű számítások sorozatával közelítik.

Dolgozatomban a végeelem-módszer elméletét a teljesség igénye nélkül, csak egy igen egyszerű feladat megoldásához szükséges mélységben ismertetem. Az általam készített végeelem-módszeren alapuló szilárdságtani számítás egy prizmatikus, homogén, lineáris anyagtulajdonsággal rendelkező rúd deformációját határozza meg a rá ható erők

ismeretében. A számítás során a rudat véges számú, egyenlő hosszúságú részre osztottam fel. Az általam használt elemtípus, a négy szabadságfokú síkbeli rúdelem (11. ábra).



11. ábra: négy szabadsági fokú kétdimenziós rúdelem

Az ábrán feltüntetett  $u_{1,2}$  és  $w_{1,2}$  a csomópontok elmozdulásait és szögelfordulásait, tehát az elem szabadságfokait mutatják.

Az ideálisan rugalmas szilárd testet alkotó elemek statikus elmozdulásait leíró alapegyenlet [15]:

$$\mathbf{K}_e \mathbf{u}_e = \mathbf{F}_e, \quad (39)$$

ahol  $\mathbf{K}_e$  az elem merevségi mátrixa,  $\mathbf{u}_e$  az elmozdulásokat, és szögelfordulásokat,  $\mathbf{F}_e$  pedig a csomópontokra ható erőket és nyomatékokat tartalmazó vektor. A végeelem-modellben a rudat alkotó elemek mindegyikére felírható a (39) egyenlet, ám a szomszédos elemek közös csomópontjaiban ébredő erők, nyomatékok, elmozdulások és szögelfordulások mind a két rúdelem szempontjából azonosak kell, hogy legyenek, ezért a teljes rendszerre érvényes globális egyenlet az egyes elemek egyenleteinek segítségével előállítható. A rendszer szabadságfokainak száma az említett elemtípus felhasználásával:

$$d = 2(n+1), \quad (40)$$

ahol  $n$  a rudat alkotó elemek száma. A globális merevségi mátrix előállításához két elem esetén [16]:

$$\mathbf{K} = \begin{bmatrix} k_{11}^{(1)} & k_{12}^{(1)} & k_{13}^{(1)} & k_{14}^{(1)} & 0 & 0 \\ k_{21}^{(1)} & k_{22}^{(1)} & k_{23}^{(1)} & k_{24}^{(1)} & 0 & 0 \\ k_{31}^{(1)} & k_{32}^{(1)} & k_{33}^{(1)} + k_{11}^{(2)} & k_{34}^{(1)} + k_{12}^{(2)} & k_{13}^{(2)} & k_{14}^{(2)} \\ k_{41}^{(1)} & k_{42}^{(1)} & k_{43}^{(1)} + k_{21}^{(2)} & k_{44}^{(1)} + k_{22}^{(2)} & k_{23}^{(2)} & k_{24}^{(2)} \\ 0 & 0 & k_{31}^{(2)} & k_{32}^{(2)} & k_{33}^{(2)} & k_{34}^{(2)} \\ 0 & 0 & k_{41}^{(2)} & k_{42}^{(2)} & k_{43}^{(2)} & k_{44}^{(2)} \end{bmatrix} \quad (41)$$

A rendszer peremfeltételei az  $\mathbf{u}$  vektor elemeinek, megfelelő értékadásával történik. Az általam modellezett rúd egyik vége befogott, ezért ebben a csomópontban két szabadsági fokot kell megkötni, tehát a vektor ezen két eleme a számítás során nulla értéket kap, mert az elmozdulás és szögelfordulás nulla. A rúd másik vége szabadon leng, ezért az ahhoz tartozó elemekre nincs megszorítás. Az említett globális egyenlet megoldásával a rúd csomópontjainak elmozdulásai és elfordulásai a csomópontokban ható erők és nyomatékok segítségével kiszámolhatók.

A rugalmas rúd végeselem-módszer segítségével számolt deformációit az SPH megoldó által számolt erők felhasználásával végeztem el. A fent említett egyenlet azonban csak statikus, egyensúlyi állapot modellezésére alkalmas, ezért az egyenlet időbeli reprezentációja szükséges, ami két további tag figyelembevételét követeli meg [15]:

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{D}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{F}, \quad (42)$$

ahol  $\mathbf{M}$  a tömegmátrix,  $\mathbf{D}$  a csillapítási mátrix,  $\ddot{\mathbf{u}}$  és  $\dot{\mathbf{u}}$  a deformációk idő szerinti második, illetve első deriváltja. Így egy tetszőleges szabadságfokú lengőrendszer mozgását leíró differenciálegyenlet-rendszert kapunk, melynek ismeretében a megoldást numerikus módszerek segítségével kereshetjük. Az általam használt  $\mathbf{K}_e$  lokális merevségi mátrixot,  $\mathbf{M}$  globális tömegmátrixot és  $\mathbf{D}$  globális csillapítási mátrixot [21] alapján a következőképpen definiáltam:

$$\mathbf{K}_e = \frac{EI_z}{l^3} \begin{bmatrix} 12 & 6l & -12 & 6l \\ 6l & 4l^2 & -6l & 2l^2 \\ -12 & -6l & 12 & -6l \\ 6l & 2l^2 & -6l & 4l^2 \end{bmatrix}, \quad (43)$$

$$\mathbf{M} = \begin{bmatrix} m & 0 & 0 & \dots & 0 \\ 0 & m\frac{l^2}{12} & 0 & \dots & 0 \\ 0 & 0 & m & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & m\frac{l^2}{12} \end{bmatrix} \quad (44)$$

$$\mathbf{D} = \begin{bmatrix} d & 0 & 0 & \cdots & 0 \\ 0 & d & 0 & \cdots & 0 \\ 0 & 0 & d & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & d \end{bmatrix}, \quad (45)$$

ahol  $E$  és  $I_z$  a rúd anyagának rugalmassági modulusza és a rúd keresztmetszetének főtengelyére számolt másodrendű nyomatéka,  $m$  egy elem tömege,  $l$  egy elem hossza,  $d$  pedig a csillapítási tényező. A végelem-módszer területén széles körben alkalmazott *Reyleigh-féle* csillapítási mátrix [21] az én számításaim során nem adott kielégítő eredményt, ezért az igen egyszerű (45)-ben szereplő mátrixot alkalmaztam. (A további számítások során természetesen egy pontosabb csillapítási modell implementálása szükséges lesz.) A globális merevségi mátrix létrehozása a (41)-ben ismertetett módon történt.

#### 4.1 A mozgásegyenlet megoldása

A kapcsolatot az SPH és a végeleemes megoldó között a (42) egyenlet jobb oldalán álló erővektor valósítja meg, ezért az egyenlet megoldása során ismertnek tekinthető. Az egyenlet megoldását idő szerinti numerikus integrálással kaphatjuk meg. Ennek megvalósítására implicit Euler-módszert használtam [17]:

$$\dot{\mathbf{u}}_{n+1} = \dot{\mathbf{u}}_n + \Delta t \ddot{\mathbf{u}}_{n+1} \quad (46)$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \dot{\mathbf{u}}_{n+1} \quad (47)$$

Az implicit megoldás segítségével lényegesen nagyobb időlépés érhető el, mint az explicit módszerekkel, azonban a számítási igénye nagyobb.

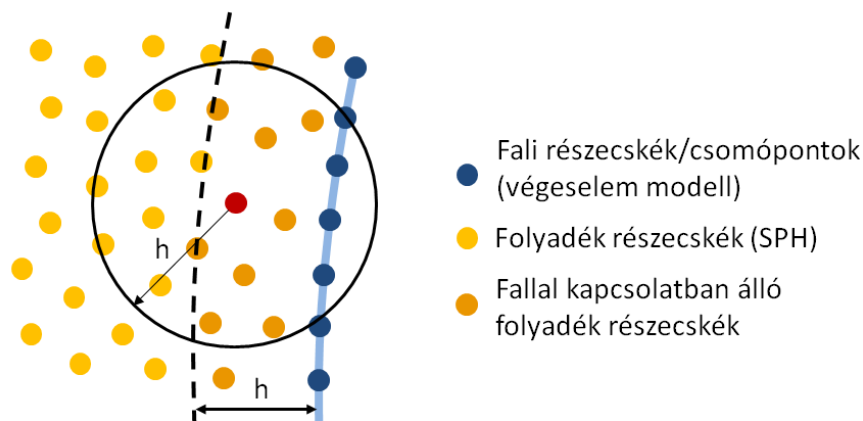
## 5 Kapcsolt FSI szimuláció (SPH - FEM)

A folyadékok és rugalmas testek kapcsolatának modellezése napjainkban is nagy nehézségeket okoz. A létező szimulációs programok többsége nem képes a kétirányú, vagy időfüggő FSI szimulációkra. További hátrány az Euler-féle modellek használatakor a víztér deformációja, ami időlépésenként a háló módosítását teszi szükségessé. Ez jelentősen növeli a feladat megoldásához szükséges számítási időt.



A probléma másik megközelítése az ALE (Arbitrary Lagrangian—Eulerian – Kapcsolt Lagrange- és Euler-féle megoldók) szemlélet, melynek segítségével a folyadék és a szilárd test modellezése az eltérő viselkedés miatt, alapjaiban eltérő módszerek segítségével modellezhető. Az ilyen kapcsolt számításokban a folyadék modellezése lagrange-i, a szilárd testé pedig euleri szemléletben történik. Ez természetesen két, egymástól független megoldó összekapcsolását teszi szükségessé, melyben az egyik megoldó futása közben a másik megoldó változóit konstansként kezeljük [18].

A szilárd, deformálható rúdra ható erők ismeretéhez szükség van a – végeelem-módszerrel modellezett – rudat alkotó elemek és az SPH megoldó részecskéinek kapcsolatára. Ez a kapcsolat a kapcsolt szimuláció megvalósításához kétirányú kell, hogy legyen, tehát a folyadék részecskéinek hatása a rúdra és a rúd hatása a folyadékra egyaránt figyelembe veendő. A végeselemes csomópontokat az SPH megoldóban a falhoz hasonlóan rögzített részecskék segítségével modelleztem, mozgásukat a végeselemes megoldó valósítja meg a csomópontokban (rögzített részecskékben) ható erők segítségével.



12. ábra: Az SPH és a végeelem-modell kapcsolata

## 6 A modellfejlesztés folyamata

### 6.1 Modellfejlesztés MATLAB környezetben

Áramlások modellezésére számos részecskealapú, főleg látványtervek és animációk készítésére alkalmas szoftver létezik (pl. ilyen a Blender). Ezek műszaki alkalmazása azonban nem lehetséges. Ennek oka az, hogy a számítások fizikai tartalma, alkalmazott módszerei sokszor nem ismertek, továbbá a műszaki szempontból fontos áramlástanai jelenségek feltárása a mezőváltozók (sűrűség, nyomás, stb.) ismeretének hiányában nem lehetséges. Az animációk készítésére íródott szoftverek a jelenségek kvantitatív modellezésére általában

nem alkalmasak, céljuk az áramlások látványos megjelenítése, fizikai hűség igénye nélkül. A kvalitatív és kvantitatív részecskealapú modellezés megvalósítására saját készítésű SPH megoldó fejlesztésébe kezdtem MATLAB környezetben. Választásom a MATLAB-ra annak rugalmassága és programozástechnikai egyszerűsége miatt esett. Segítségével a modell fejlesztése során annak tartalmi részletei könnyebben érthetőek, a beépített grafikai eszközök segítségével működése könnyebben figyelemmel követhető és kézben tartható. További előnye, hogy eszköztárának használatával összetett számítások implementálása rövidebb programot eredményez (így a kód alig több mint 400 sor) mint egy alacsonyabb szintű programnyelven. MATLAB környezetben a modell paramétereinek finomhangolása nagyfokú hatékonysággal lehetséges. Hátránya azonban, hogy a számítások – alacsonyabb szintű nyelvekhez hasonlítva – lényegesen lassabbak, ezért alkalmazása csak kétdimenziós áramlás modellezésére célszerű.

## 6.2 Sűrűségmodellek implementálása

Először a hagyományos SPH módszert implementáltam MATLAB környezetben, annak sűrűségmodelljével együtt. Szeretném hangsúlyozni, hogy a MATLAB környezet nagy segítséget nyújtott a programhibák feltárásában és elhárításában. Ekkor került sor a számítások grafikai megjelenítésének és az eredmények dokumentálásának kidolgozására is. Ez a munkafázis a nagy mennyiségű próba futtatás miatt nagyon időigényes volt, a hibák elhárítását követően azonban egy stabil fejlesztési környezet állt elő.

A várakozásoknak megfelelően a hagyományos SPH által használt állapotegyenlet nem adott kielégítő eredményt, a folyadék rugalmassága miatt alacsony frekvenciás lengések alakultak ki.

Ezek elkerülésére adaptáltam először a WCSPH módszert, mely lényegesen nagyobb merevségű folyadékok áramlását képes modellezni. Ahogy a 3.4.2-es alfejezetben említettem, ez a módszer a nagy merevség következtében igen kicsiny időlépés alkalmazása esetén eredményez numerikusan stabil eredményt, ami a számítási időt nagymértékben megnövelte. Ennek a korlátozó tényezőnek a lehető legkisebbre szorítását úgy értem el, hogy a 3.6. alfejezetben említett *adaptív időlépést* alkalmaztam az ott említett paraméterekkel. Ezt a WCSPH modellváltozatban alkalmaztam először.

A számítási sebesség jelentős növelését azonban a PCISPH [9] módszerváltozat segítségével értem el. Ennek megvalósítása, iteratív volta miatt, lényegesen összetettebb feladat volt. Külön problémát okozott, hogy [9] lényeges technikai részleteket nem tartalmazott. A konvergenciaproblémák megoldása a paraméterek pontos hangolását követelte meg. A módszert kidolgozó, [9] egyik szerzőjével történt konzultációt követően sikerült csak stabilizálnom a PCISPH eljárást. Ennek köszönhetően a kétdimenziós áramlásokat kvalitatívan helyesen modellező PCISPH megoldó készült el.

### 6.3 Gátszakadás modellezése

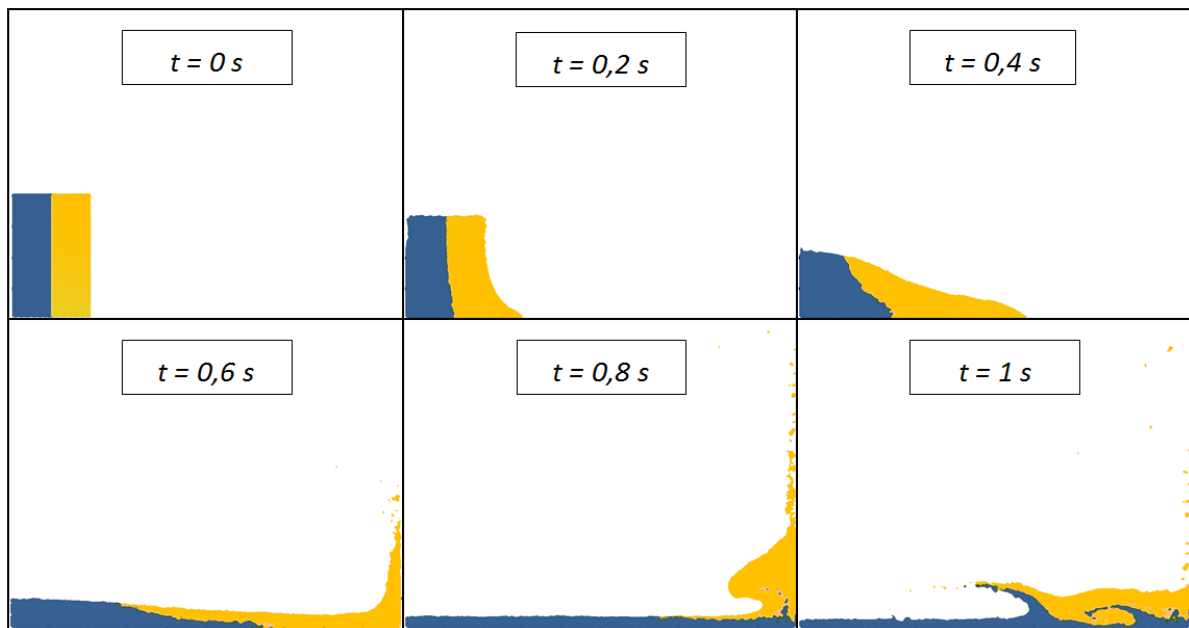
A program és a modellváltozatok tesztelésére gátszakadás (dam break) 2D szimulációját választottam. Ez az SPH módszer egyik mintapéldája, több cikkben is szerepel ezzel kapcsolatos demonstráció. Az SPH algoritmusok fejlesztése során a validált algoritmusok és mérések kvalitatív összehasonlítási alapot adnak. Az általam végzett számítások során azért alkalmaztam a gátszakadás modellezését, mert építőmérnöki szempontból – például a vörösiszap-katasztrófa – ez a teszteset a legrelevánsabb. A gátszakadás modellezése során számos áramlási jelenség lefolyása ellenőrizhető a numerikus modellben. Ilyen például a kezdetben kialakuló függőleges folyadékfal kvázi-parabolikus profilja, vagy a falhoz való ütközés után előre haladó és a visszacsapódó rétegek között kialakuló örvények. A 13, 14, 15. ábrákon mutatom be a gátszakadási számítások eredményét mindhárom SPH variáns esetén. A folyadék kezdeti méretei valamennyi számítás során 0,1 m és 0,2 m volt. Az 1. táblázat mutatja a számítások paramétereit.

Paraméter neve	Jele	Értéke
Víztér mérete [m]	$L$	0,4×0,4
Referenciasűrűség [kg/m <sup>3</sup> ]	$\rho_0$	1000
Dinamikai viszkozitás [Pa·s]	$\mu$	0,5
Részecskék tömege [kg]	$m$	0,011
Közeg merevségi állandója [m <sup>2</sup> /s <sup>2</sup> ] (SPH)	$k$	10000
Hang terjedési sebessége [m/s] (WCSPH)	$c_s$	4

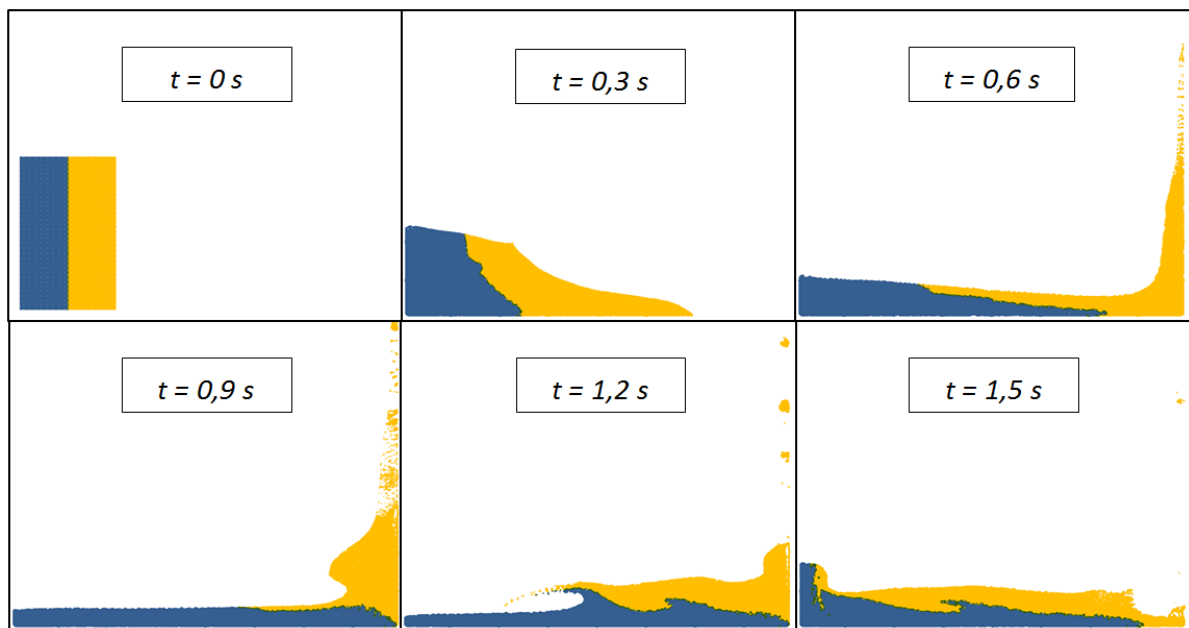
1. táblázat: A számítások paramétereit

A dinamikai viszkozitás viszonylag magas értékét az indokolja, hogy így – mint örvényviszkozitást – próbálom modellezni az áramláskor fellépő turbulenciát. Numerikus szempontból ennek előnyös mellékhatása a 3.5.2-es alfejezetben említett számítást stabilizáló effektus.

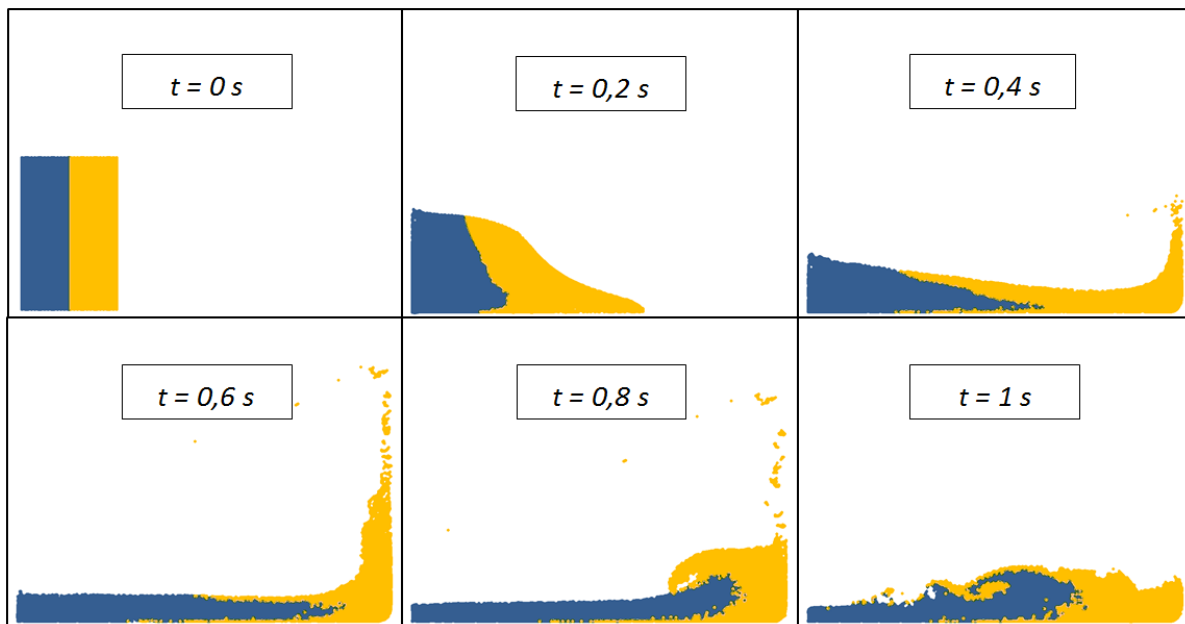
A folyadék mozgásának szemléltetésére eltérő színeket alkalmaztam, a kezdeti állapotban a részecskék felét kék, a másik felét sárga színnel jelöltem meg. A részecskék a szimuláció során megtartják eredeti színüket, egyébként a két folyadék rész minden tulajdonsága azonos. Ezt a módszert a lagrange-i áramlásmodellezésben „festékfolttérjedési” vizsgálatnak nevezik [22].



13. ábra: Gátszakadás modellezése hagyományos SPH-val. Részecskék száma: 5 000



14. ábra: Gátszakadás modellezése WCSPH-val. Részecskék száma: 5 000

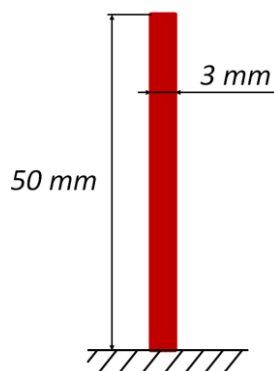


15. ábra: Gátszakadás modellezése PCISPH-val. Részecskék száma: 1 800

A különböző SPH változatok közötti különbségek a paraméterek hangolásával csökkenthető. A PCISPH módszer fali peremfeltételének problémája [12] nagyobb részecskeszám esetén már instabilitást okoz, ezért az általam készített számítás csak kisebb részecskeszám működőképes.

#### 6.4 Egy újfajta peremfeltétel, mint FSI feladat

A peremfeltételként alkalmazott fali részecskék végeelem-moddellel történő helyettesítése lehetővé teszi a modellezett áramlás rugalmas, szilárd testekkel való kapcsolatának vizsgálatát. Számításaim során az áramlási térbe helyezett szilárd testre ható erőket az 5. fejezetben ismertetett módszer segítségével vettem figyelembe. A kapcsolt szimulációk során a folyadék és a víztér kezdeti geometriája a 6.3-as alfejezetben bemutatott számításokkal azonos, a szilárd test geometriája pedig a 16. ábrán látható.



16. ábra: Az áramlási térbe helyezett test geometriája

Paraméter neve	Jele	Értéke
Rúd tömege [kg]	$M$	0,165
Rugalmassági modulusz [GPa]	$E$	0,5
Részecskék tömege [kg]	$m$	0,011
Csillapítási tényező	$d$	0.01

2. táblázat: A végelelemes modell paraméterei

A 2. táblázatban láthatók a kapcsolt számításban alkalmazott végelelemes modell paraméterei.

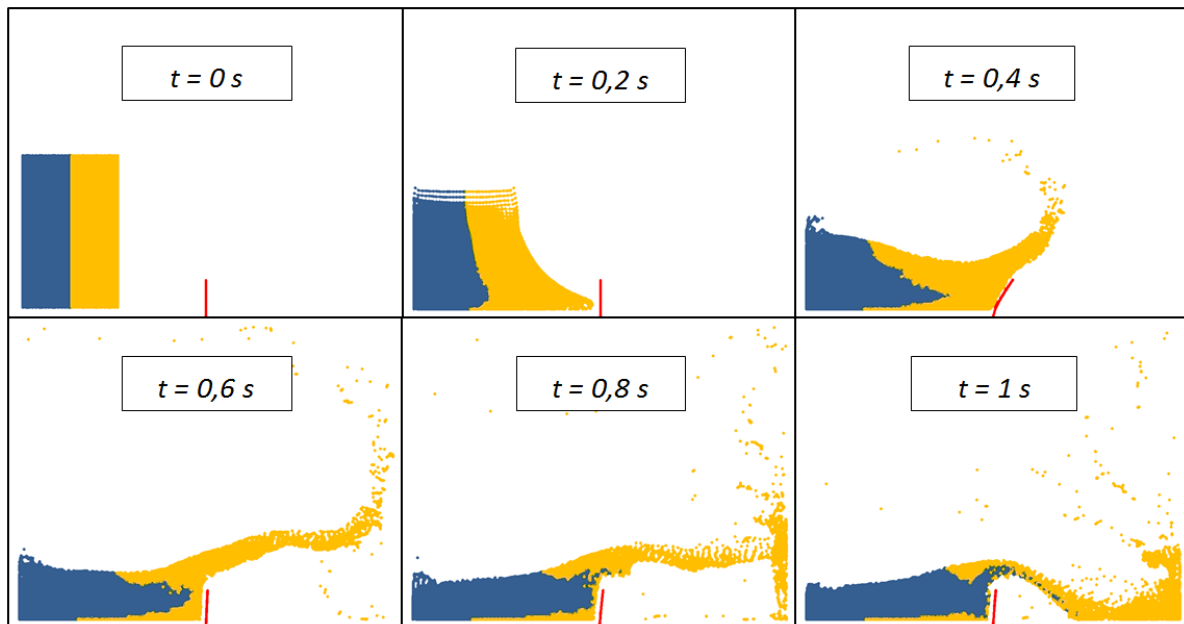
A két megoldó összekapcsolása megköveteli az időlépések összehangolását. A számításaimat csak a WCSPH variáns alkalmazásával végeztem el, ezért az ebben használt adaptív időlépés a végelelem-modellre is hatással volt. A végelelem-modell időlépésének számítását az alábbiak szerint végeztem:

$$n_t = \text{int} \left( \frac{\Delta t^{SPH}}{\Delta t_{Max}^{FEM}} \right) + 1, \quad (48)$$

$$\Delta t^{FEM} = \frac{\Delta t^{SPH}}{n_t}, \quad (49)$$

ahol  $n_t$  az SPH számítás egy időlépése alatt elvégzett végelelem-időlépések száma, int pedig az osztás egész része. Ezzel elérhető volt, hogy az SPH modell adaptív időlépése ellenére a végelelem-modell időlépése minden esetben az előre definiált maximális időlépés alatt maradjon. Mivel az SPH időlépése általában kisebb, mint a végelelem-modell maximális időlépése, ezért a lépések túlnyomó részében  $\Delta t^{FEM} = \Delta t^{SPH}$  volt.

A kapcsolt számítás eredményét mutatja a 17. ábra.



17. ábra: FSI szimuláció WCSPH és végelelemes módszer összekapcsolásával.  
Részecskék száma: 1 800 (SPH), 100 (FEM)

A kapcsolt számítások elvégzését követően találtam a [20] konferenciakiadványt. Tudomásom szerint SPH és FEM megoldók összekapcsolását tárgyaló más publikáció nem jelent még meg, ezért egy kimondottan újszerű modellezési eljárásról van szó.

## 7 Eredmények, következtetések

- *SPH modell implementálása MATLAB-ban.*  
A különböző SPH modellek implementálása által egy olyan nyílt felszínű, inkompresszibilis áramlást modellező saját fejlesztésű szoftvereszközt hoztam létre, amely továbbfejlesztést követően kvantitatív számítások elvégzését teheti lehetővé mind építő-, mind gépészmérnöki területeken adódó feladatok megoldásában.
- *Három állapotegyenlet-változat implementálása és összehasonlítása.*  
A különböző állapotegyenletek használatával az implementált modellek alkalmazásorientált felhasználása válik lehetővé. A gátszakadás tesztesetén végzett számítások igazolták a változatokkal kapcsolatos szakirodalmi tapasztalatokat: mindhárom modell helyesen írta le az áramlási mezőt gátszakadás során. A WCSPH alkalmazásával a hagyományos SPH modellben jelentkező, cseppfolyós közegek esetében irreális sűrűségkülönbségeket sikerült elkerülni, de a futási idő még adaptív időlépés alkalmazásával is jelentősen megnőtt. Ezt a problémát a PCISPH módszer implementálásával sikerült orvosolni.
- *FEM+SPH peremfeladat prototípusának kidolgozása.*  
A részecskealapú megoldók áramlás- és szilárdságtani számítások esetében egyaránt

alkalmazhatók. A szilárdságtani számításokra jól kidolgozott végeelemes modellek alkalmazása előnyös a rugalmas testek igénybevételeinek meghatározására a terhelések ismeretében. Az áramlástanai számítások részecskealapú modellezése azonban sok esetben előnyösebb az Euler-féle modelleknél. Ilyen esetekben az ALE eljárás alkalmazása hatékonyan alkalmazható. Ennek megvalósítására készítettem egy egyszerű végeelem-módszeren alapuló időfüggő szilárdságtani modellt, melyet összekapcsoltam a WCSPH módszeren alapuló áramlástanai modellel. Ennek segítségével újszerű megközelítésben végeztem FSI számításokat. Ez a kutatási irány nemzetközi kutatásban is kifejezetten újszerűnek számít.

## 8 További feladatok, lehetőségek

A további fejlesztéseknek három főirányát látom: a modell további finomhangolása és fejlesztése, újabb fizikai effektusok figyelembevétele; a futtatási hatékonyság növelése; konkrét műszaki alkalmazások, esettanulmányok vizsgálata.

A fejlesztés során az széleskörű alkalmazhatóságot szem előtt tartva szükségessé válhat a felületi feszültség modellezése [5], a be- és kilépő peremfeltételek, valamint turbulenciamodellek megvalósítása [11]. A PCISPH modellváltozat alkalmazása nagyfokú hatékonyságot tehet lehetővé [10], ennek fejlesztése viszont iteratív volta miatt nagy nehézségekbe ütközik, ám az adaptív időlépés ebben az esetben is alkalmazható [12]. További jelentős teljesítménynövekedés érthető el az iterációs lépések során történő részecskeszelektálással, melynek segítségével az egyes iterációs lépések időigénye csökkenthető [19].

Az algoritmus finomhangolását és optimalizálását követően alacsonyabb szintű programnyelven való implementálással a futási idő jelentősen csökkenthető. Az SPH számítások, a módszer explicit jellegéből adódóan, igen hatékonyan párhuzamosíthatók GPU-n (Graphics Processing Unit) történő számításokhoz, melynek segítségével a számítási idő a töredékére csökken.

Az általam készített, ALE-szemléletű FSI algoritmus lehetővé teszi kétdimenziós áramlások rugalmas testekkel való kapcsolatának modellezését. Az algoritmus felhasználhatósági körének szélesítéséhez a modell háromdimenziós implementálása elengedhetetlen. A további fejlesztések lehetőséget adnak egy olyan eszköz kifejlesztésére, amely építő- és gépészmérnöki területeken fellépő problémák megoldása során nyújthat segítséget. Az építőmérnöki területen ilyen feladatok például a szökőár-jelenségek vizsgálata, hullámtörők hatékonyságának maximalizálása, folyómeder eróziójának előrejelzése, hidak és épületek víz-, és szélterhelésének modellezése. Gépészmérnöki területen például turbinák, hullámerőművek tervezése és fejlesztése, többfázisú áramlások modellezése, során válhat hasznos eszközzé.



## 9 Felhasznált irodalom

- [1] Dr. Szabó Kálmán Gábor. „Multiphase and Reactive Flow Modelling”. Előadásjegyzet, BME 2012.
- [2] Gingold RA, Monaghan JJ. „Smoothed particle hydrodynamics”. Mon. Not. R. Astron. Soc. **181**, 375–89, 1977.
- [3] Lajos Tamás. „Az áramlástan alapjai”. Egyetemi tankönyv, Budapest, 2008, ISBN 978 963 06 6382 3.
- [4] Kristóf Gergely. „The solution of pressure velocity coupling”. Computational Fluid Dynamics előadásjegyzet, BME 2012.
- [5] M. Kelager. „Lagrangian Fluid Dynamics Using Smoothed Particle Hydrodynamics”. (Department of Computer Science, University of Copenhagen, 2006).
- [6] Marcus Vesterlund. „Simulation and Rendering of a Viscous Fluid using Smoothed Particle Hydrodynamics”. Master thesis, Virtual Reality Laboratory, Umea Universitet, 2004.
- [7] [http://www.ged.rwth-aachen.de/index.php?cat=Tools\\_n\\_Methods&subcat=Geomechanical\\_modeling&page=Smoothed\\_Particle\\_Hydrodynamics](http://www.ged.rwth-aachen.de/index.php?cat=Tools_n_Methods&subcat=Geomechanical_modeling&page=Smoothed_Particle_Hydrodynamics)
- [8] M. Becker, M. Teschner. „Weakly compressible SPH for free surface flows”. In: „Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation” 209–217 (Eurographics Association, 2007, ISBN 978-1-59593-624-0).
- [9] B. Solenthaler, R. Pajarola. „Predictive-Corrective Incompressible SPH” ACM Trans. Graph. **28** (3) 40 (6 pp), 2009, DOI 10.1145/1531326.1531346.
- [10] B. Solenthaler, R. Pajarola. „Performance Comparison of Parallel PCISPH and WCSPH”. Technical Report No. IFI-2010.0003 (Department of Informatics, University of Zurich, 2009).
- [11] Mostafa Safdari Shadloo, Amir Zainali, Samir H. Sadek, Mehmet Yildiz. „Improved Incompressible Smoothed Particle Hydrodynamics method for simulating flow around bluff bodies”, Computer methods in applied mechanics and engineering, **200**(9) 1008-1020, 2011
- [12] Markus Ihmsen, Nadir Akinci, Marc Gissler, Matthias Teschner. „Boundary handling and adaptive time-stepping for PCISPH”, Workshop on Virtual Reality Interaction and Physical Simulation VRIPHYS, 2010.
- [13] MONAGHAN J. „Smoothed particle hydrodynamics”. Ann. Rev. Astron. Astrophys. **30**, 543–574, 1992.

- [14] Dr. Kovács Ádám, Dr. Szekrényes András. "Végeselem-módszer". Creative Commons NonCommercial-NoDerivs 3.0 (CC BY-NC-ND 3.0), ISBN 978-963-279-539-3.
- [15] Bojtár Imre, Gáspár Zsolt. „Végeselem módszer építőmérnököknek” (TERC Kereskedelmi és Szolgáltató Kft., Budapest, 2003, ISBN 963 86303 2 9).
- [16] David Roylance. „FiniteElementAnalysis” (Department of Materials Science and Engineering, Massachusetts Institute of Technology, Cambridge, 2001).
- [17] Kristóf Gergely. „Numerical approximations of derivatives and integrals” Computational Fluid Dynamics előadásjegyzet, BME 2012.
- [18] Lars Aaes Jakobsen, Erik Lund, and Henrik Møller. „Shape Sensitivity Analysis of Time Dependent Fluid-Structure Interaction Problems Using the ALE Method”. Institute of Mechanical Engineering, Aalborg University, Dánia, 2001
- [19] Karthik Raveendran, Chris Wojtan, Greg Turk. „Hybrid Smoothed Particle Hydrodynamics”, Eurographics/ ACM SIGGRAPH Symposium on Computer Animation 2011
- [20] G. Fourey, G. Oger, D. Le Touzé, B. Alessandrini. „SPH-FEM coupling to simulate Fluid-Structure Interactions with complex free-surface flows”, Proceedings of the 5<sup>th</sup> Spheric Workshop (University of Manchester, 2010).
- [21] Páczelt István, Szabó Tamás, Baksa Attila. „A végeselem-módszer alapjai”, Értékünk az ember, humánerőforrás-fejlesztési operatív program, 2007
- [22] Tél Tamás, Gruiz Márton. „Kaotikus dinamika”, Nemzeti Tankönyvkiadó, Budapest, 2002, ISBN 963 19 3280 X