

مشروع الحسابات العلمية

محاكاة قارب بمحرك بسيط باستخدام Three JS

2024

إعداد الطلاب

آلاء محمد عدنان المرجي

نغم نضال السهو

علي محمد زيتون

أمل أحمد فتحية

بثينة محمد سعيد شكاكي

نسيبة عمر بعلة

بإشراف الأستاذ

إسماعيل الكريان

3	صفات القارب.....
5	القوة المؤثرة.....
5	قوة النقل.....
5	قوة الطفو.....
6	قوة الدفع.....
9	قوة السحب Force resulting from centerboard-water interaction.....
9	قوة التيارات المائية Current induced forces.....
10	قوة الرياح.....
10	دراسة الحالات الحركية للقارب.....
10	الحالات التي تحقق قانون نيوتن الأول.....
12	الحالات التي تحقق قانون نيوتن الثاني:.....
13	عزم الدوران الناتج عن المحركات.....
13	العزم الزاوي الكلي الناتج عن المحركين.....
14	التسارع الزاوي.....
14	(Moment of Inertia) عزم القصور الذاتي $I\Delta$
14	السرعة الزاوية وزاوية الدوران.....
15	القوانين الفيزيائية المؤثرة.....
15	معادلة القوة والتسارع الخطي.....
15	العلاقة بين التسارع الخطي والتسارع الزاوي.....
15	معادلة عزم الدوران والتسارع الزاوي.....
16	الدراسة الخوارزمية والبرمجية.....
19	التصميم التقني (شرح مكونات البرنامج – الصفوف):.....
27	المخطط التدفقي.....
28	مخطط الصفوف.....
28	المشاكل وحلها.....
29	صور توضيحية لمحاكاة لحركة القارب:.....
29	توابع القوى.....
32	آلية تقسيم العمل.....
32	المشكلات والتطويرات المستقبلية:.....
32	المراجع:.....

صفات القارب



الطول: حوالي 4 متر مما يوفر استقرار الصيد ومساحة لأنشطة الصيد.

العرض (البيم): تقريباً 5.5 متر، مما يوفر منصة مريحة للصيد

والحركة على متن القارب.

قوة المحرك: يمكن تجهيزه بمحرك خارجي يصل إلى 175 حصاناً، مما يوفر قوة كافية للإبحار والوصول إلى أماكن

الصيد بسرعة.

تصميم الهيكل: يتم تصميمه بقاع على شكل حرف V عميق للتعامل مع ظروف المياه المتقلبة، مما يجعله مناسباً

لبعثات الصيد المختلفة.

مقر القيادة: منطقة واسعة مع مساحات لتخزين معدات الصيد وآبار الطعم وصناديق الأسماك .

مقصورة القيادة: توفر الحماية من الرذاذ والرياح، مع مقاعد وأدوات التحكم للملاحة.

التنوع: مصمم ليكون متعدد الاستخدامات لكل من الصيد الترفيهي والرحلات العائلية .

الدفة (Rudder) : هي لوحة متحركة تُثبت عادةً في مؤخرة القارب دورها هو توجيه حركة القارب من خلال دفع

الماء.

و هي الجزء المتحرك من القارب الذي يُمكن تحريكه لضبط اتجاه القارب تُثبت الدفة في مؤخرة القارب تُتحكم

في الدفة عن طريق "عجلة التوجيه" (Steering Wheel) تُسبب حركة الدفة ضغطاً غير متساوي على المياه حول

القارب ، مما يؤدي إلى دوران القارب

تم تصميم هذه القوارب مع مراعاة الصياد، حيث تحتوي على حاملات للصنارات ومساحات لتخزين الأدوات وغالباً

ما تشمل الإلكترونيات المتقدمة مثل أجهزة تحديد المواقع وأجهزة الكشف عن الأسماك لتحديد موقع الأسماك. تم

تصميم تخطيط السطح ليكون مفتوحاً لتعظيم مساحة الصيد، والمواد المستخدمة مختارة للمتانة وسهولة الصيانة.

للحصول على معلومات أكثر تحديداً ستحتاج إلى النظر في الطرازات وماركات معينة حيث يمكن أن تختلف التصاميم

والمواصفات بشكل كبير اعتماداً على الغرض من القارب وفلسفة تصميم الشركة.

أنواع المحركات المستخدمة في القوارب ذات المحركات العادية تتنوع بين عدة خيارات، وتشمل:

1- محركات خارجية (Motors Outboard) :

- هذه المحركات معلقة على ذيل القارب وتحتوي على وحدة دفع مدمجة .

- سهولة التركيب والإزالة وتستخدم بشكل شائع في القوارب الصغيرة والمتوسطة .

2- محركات داخلية /خارجية (Inboard/Outboard Motors or Sterndrives):

- تجمع بين محرك داخلي ووحدة دفع خارجية .

- توفر توازناً جيّداً بين الاداء والقدرة على المناورة.

3- محركات داخلية (Motors Inboard):

- محركات مثبتة داخل هيكل القارب، وتتصل بمروحة دفع عبر عمود.

- توفر مركز ثقل منخفض وتستخدم في القوارب الكبيرة واليخوت .

- تتطلب فتحة في قاع القارب لعمود الدفع مما يجعلها اقل شيوعاً في القوارب الصغيرة.

4- محركات النفاثة (Drives Jet) :

- تستخدم قوة الدفع النفاثة لتحريك القارب.

- لا تحتوي على أجزاء خارجية متحركة تحت الماء، مما يجعلها مناسبة للمياه الضحلة .

5- محركات الدفع الكهربائية (Drives Electric) :

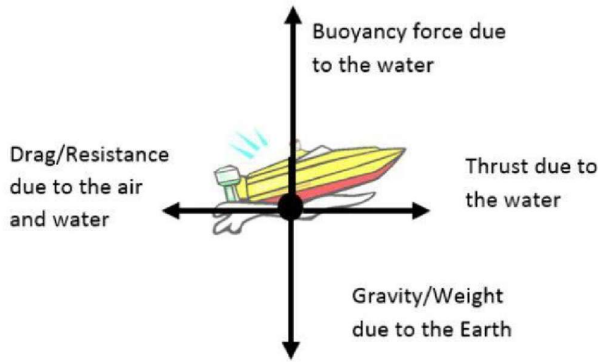
- تعمل بالطاقة الكهربائية وتكون صديقة للبيئة.

- مناسبة للقوارب الصغيرة والمتوسطة وتستخدم بشكل متزايد في المناطق التي تفرض فيها قيود على الانبعاثات.

6- محركات الديزل (Engines Diesel) :

- تستخدم بشكل رئيسي في القوارب الكبيرة والتجارية بسبب كفاءتها في استهلاك الوقود وعمرها الطويل.

القوة المؤثرة



بشكل عام كل القوى المؤثرة على السفينة تصنف إلى:

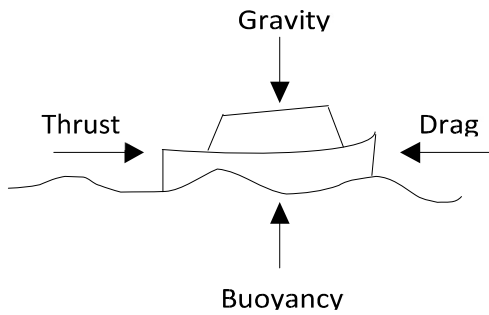
- قوى ساكنة:

- قوة الثقل.

- قوة الطفو.

- قوى ديناميكية: فتنقسم إلى عدة قوى منها:

قوى تصادم الرياح واهتزاز الهيكل وتأثير الرياح ومقاومة حركة السفينة والقوى الناتجة عن شغل الرفاس



- قوة الدفع (المحرك)

- قوة السحب

- قوة التيارات المائية

- قوة الرياح

قوة الثقل

تؤثر رأسياً إلى أسفل في اتجاه عجلة الجاذبية الأرضية وتكون نقطة تأثير هذه القوة هي نقطة مركز ثقل السفينة (CG)

$$W = M * g = (9.8)M$$

- M هو الكتلة المجمعة للقارب والبحارة Kg

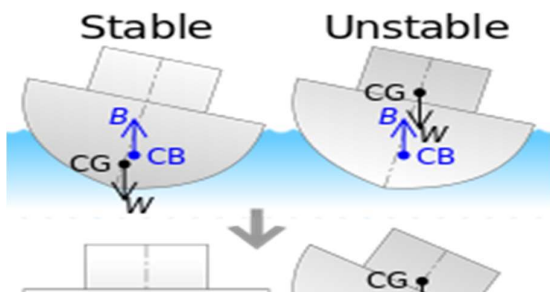
- g هي تسارع الجاذبية الأرضية (9.81 m/s)

- تعمل القوة W للأسفل في الاتجاه السلبي للمحور Z

قوة الطفو

وهي قوة دفع الماء وتؤثر رأسياً إلى الأعلى ويكون تأثيرها على

نقطة تعرف بمركز الطفو (CB) وفقاً لقاعدة أرخميدس



(إذا غمر جسم جزئياً أو كلياً في سائل ، فإنه يكون مدفوعاً

من أسفل إلى أعلى بقوة دفع تساوي وزن السائل المزاح من الجسم

؛ وحجم هذا السائل يساوي حجم الجسم المغمور في هذا السائل)

وهي تنطبق مع المركز الهندسي لهيكل السفينة المغمور في الماء.

قوة الطفو تساوي وزن الجسم المغمور في الماء. أي أن وزن الأجسام في الماء تفقد وزنها الطبيعي فهذا يؤثر على حركة

الأجسام وقوة دفعها .

فيمكن حساب قوة الدفع على الوزن الظاهري فقط

حيث أن وزن الجسم الطبيعي - وزن الجسم المغمور في الماء = وزن السائل = حجم السائل معا مضروب في الكثافة

المخصصة للسائل.

$$B = P_{\text{water}} * g * V$$

- V هو الحجم المغمور (يساوي حجم الماء المزاح بواسطة القارب) m^3

- P_{water} كثافة ماء البحر تبلغ في المتوسط (1025 كجم / متر مكعب ماء البحر و 1000 للماء العذب)

- g هي تسارع الجاذبية الأرضية ($9.81m/s^{-1}$)

قوة الدفع

هي القوة التي تدفع القارب إلى الأمام ويتم توليدها من خلال دوران محرك القارب مما يولد رد فعل بالاتجاه المعاكس تدفع

القارب إلى الأمام.

يوجد للقارب محركين ايمن وايسر لكل واحد قوة دفع خاصة به

معادلة قوة الدفع النظرية:

$$F_{thrust} = \rho_{water} * A_{propeller} * \eta_{propeller} * \omega_{propeller} * 0.5$$

- F_{thrust} : قوة الدفع الناتجة عن المروحة.
- ρ_{water} : كثافة الماء، وهي ثابتة (1025 m/kg).
- $A_{propeller}$: مساحة شفرات المروحة التي تتفاعل مع الماء (متر مربع).
- $\eta_{propeller}$: كفاءة المروحة، وهي نسبة تعبر عن مدى فعالية تحويل الطاقة إلى قوة دفع.
- $\omega_{propeller}$: سرعة دوران المروحة (الراديان/الثانية).

هذه المعادلة تعبر عن القوة التي تولدها المروحة بناءً على السرعة التي تدور بها. كلما زادت سرعة دوران المروحة زادت قوة الدفع المولدة، بشرط أن تكون بقية المتغيرات ثابتة.

1- الحد الأقصى لقوة الدفع (Max Thrust Force):

$$\rho_{engine} \geq F_{thrust}$$

ρ_{engine} : القوة القصوى للمحرك (نيوتن).

قوة الدفع الفعلية لا يمكن أن تتجاوز القوة القصوى التي يمكن أن ينتجها المحرك، حتى لو زادت السرعة أو العوامل الأخرى.

حساب الدفع الكلي (Total Thrust) :

$$F_{total} = F_{thrust,L} + F_{thrust,R}$$

F_{total} : قوة الدفع الكلية المؤثرة على القارب.

$F_{thrust,L}$: قوة الدفع الناتجة عن المحرك واليسر.

$F_{thrust,R}$: قوة الدفع الناتجة عن المحرك اليمين.

المعادلة تجمع قوى الدفع من المروحتين للمحركين (اليمين واليسر) للحصول على القوة الكلية التي تؤثر على حركة القارب. إذا كان أحد المحركات معطلاً، فإن القوة من هذا المحرك تكون صفراً، وبالتالي يؤثر فقط المحرك الآخر على الحركة.

كيف تؤثر هذه المعادلات على القارب؟

- **قوة الدفع: (Thrust Force)** هي القوة الأساسية التي تدفع القارب للأمام. كلما زادت قوة الدفع، زادت سرعة القارب.
- **كفاءة المروحة: (Propeller Efficiency)** تؤثر بشكل مباشر على مقدار القوة الناتجة لكل وحدة طاقة تُستهلك. مراوح ذات كفاءة عالية توفر قوة دفع أكبر لنفس السرعة، مما يحسن أداء القارب ويقلل استهلاك الوقود.
- **سرعة دوران المروحة: (Propeller Speed)** سرعة الدوران العالية تزيد من قوة الدفع، ولكنها أيضاً تزيد من استهلاك الوقود بشكل غير خطي.

قوة السحب Force resulting from centerboard-water interaction

القوة الناتجة عن التفاعلات بين اللوح المركزي والماء ومعامل السحب

$$F_{cbw} = 0.5 * C * \rho * A_{cb} * v^2$$

— حيث C هو معامل السحب،

— ρ هو كثافة الماء 1025 (كجم/متر مكعب)،

— A_{cb} هو مساحة اللوح المركزي (المساحة الطولية المتوقعة للقارب) m^2 ،

— v هو سرعة القارب ms^{-1}

قوة التيارات المائية Current induced forces

تتأثر قوة التيارات بسرعة التيار على المحورين x و z وبزاوية هبوب التيار .

$$F_c = \frac{1}{2} \rho_w \times v_c^2 \times L_w \times T \times C_c \times \sin(\theta)$$

حيث F_c هي القوة العرضية الناتجة عن التيار N ،

— ρ_w هو كثافة الماء،

— v_c يكون سرعة التيار ms^{-1} ،

— L_w هو طول خط المياه m ،

— T هو الغاطس (المسافة من خط الماء الى اسفل القارب) m ،

— C_c هو معامل السحب،

— θ هي زاوية اتجاه التيار بالنسبة للقارب

قوة الرياح

تؤثر الرياح في حركة القارب وتضيف قوّة جديدة إلى محصلة القوى التي يخضع لها القارب.

يمكننا حساب القوة الناتجة عن رياح ذات سرعة محددة عن طريق القانون التالي

$$F_{wind} = C_d \times P \times A$$

نلاحظ أنه مشابه لقانون مقاومة السحب

P ضغط الرياح

$$P = \rho \times V^2$$

$$P = 0.00256 * V^2$$

ρ هي كثافة الهواء، و A هي المساحة التي يؤثر عليها الهواء و C_d هو معامل السحب و v هو سرعة الهواء ms^{-1} .

دراسة الحالات الحركية للقارب

تكون فيه محصلة القوى المؤثرة في القارب مساوية للصفر وسرعته إما معدومة أو ثابتة بحيث يتحقق

قانون نيوتن الاول $\Sigma \vec{F} = 0$: الذي ينص على أن الجسم إذا كان في حالة سكون أو حالة حركة بسرعة

ثابتة في خط مستقيم، فإنه سيبقى في حالة سكون أو يستمر في التحرك في خط مستقيم بسرعة ثابتة

ما لم تؤثر به قوة خارجية تجبره على تغيير ذلك، وتكون محصلة القوى المؤثرة به معدومة. وهي الحالة

البداية للقارب.

الحالات التي تحقق قانون نيوتن الأول

1- حالة السكون (التوازن)

تتحقق عندما تصبح قوة الطفو مساوية لقوة الثقل يخضع حينها القارب لقوتين فقط ، الثقل و الرفع

(باعتبار عدم وجود رياح)

$$\vec{F} = \vec{W} + \vec{B} = 0$$

2- حالة السير بسرعة ثابتة

يكون القارب في هذه الحالة خاضعاً لجميع القوى المؤثرة في حركته او بعضها وتكون محصلة هذه القوة معدومة وسرعة القارب ثابتة .

$$\sum \vec{F} = \vec{W} + \vec{B} + \vec{F}_{cbw} + \vec{F}_{wind} + \vec{F}_c + \vec{F}_T = \vec{0}$$

3- حالة وجود قوة تؤثر على القارب

بوجود قوة خارجية تؤثر على القارب، يكتسب القارب تسارعا محددا وتصبح محصلة القوى التي تؤثر فيه غير معدومة ويتحقق قانون نيوتن الثاني:

$$\sum \vec{F} = m \times \vec{a}$$

الذي ينص على أن الجسم إذا أثرت عليه قوة ما فإنها تكسبه تسارعا، وعكسيا مع كتلته ويكون باتجاه تلك القوة. وهو القانون الذي س نحدد من خلال سرعة القارب وبالتالي تغيرات احداثياته ومكانه. نقوم بإيجاد محصلة القوى المؤثرة في القارب

$$\sum \vec{F} = \vec{W} + \vec{B} + \vec{F}_{cbw} + \vec{F}_{wind} + \vec{F}_c + \vec{F}_T = m \times \vec{a}$$

ثم نقوم بإيجاد التسارع عن طريق قسمة شعاع القوة على كتلة القارب:

$$\vec{a} = \frac{\sum \vec{F}}{m}$$

ثم يمكننا حساب السرعة اللحظية للقارب عن طريق القانون:

$$\vec{v} = \vec{a}t + \vec{v}_0$$

وعن طريق السرعة نستطيع حساب الموقع الجديد

$$\vec{x} = \vec{v}t + \vec{x}_0$$

يمكن حسابها مباشرة من القانون

$$\vec{x} = \frac{1}{2} \vec{a}t^2 + \vec{v}t + \vec{x}_0$$

ولكن محصلة القوى ستستمر بالتناقص تدريجياً بسبب قوة مقاومة الماء التي تتناسب طردياً مع سرعة القارب وتعاكس اتجاهه حتى تصبح معدومة وبالتالي ينعدم التسارع وتثبت السرعة ويتحقق قانون نيوتن الأول فتصبح الحركة حركة مستقيمة ذات سرعة ثابتة وتسارع معدوم.

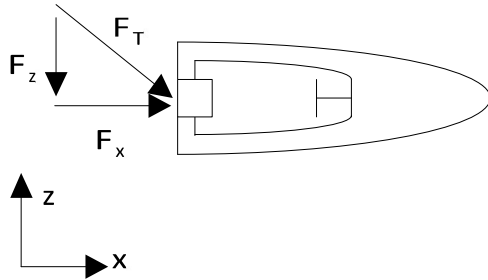
الحالات التي تحقق قانون نيوتن الثاني:

1- حالة السير نحو الأمام

ينطلق القارب نحو الأمام بفعل قوة الدفع التي ينتجها المحرك والمروحة المتصلة به، حيث تؤثر على القارب ويكون عندها خاضعاً:

$$\sum \vec{F} = \vec{W} + \vec{B} + \vec{F}_{cbw} + \vec{F}_{wind} + \vec{F}_c + \vec{F}_T = m \times \vec{a}$$

2- حالة الدوران يميناً ويساراً



عندما يتم تدوير متجه قوة الدفع، تنشأ مكونات قوة عمودية ومتعامدة مع اتجاه سير القارب. كما هو موضح في الشكل، يؤدي إلى F_z ، مكون القوة العمودية توليد عزم دوران حول مركز كتلة القارب. هذا العزم يؤدي إلى تسارع زاوي للقارب، مما يغير اتجاه القارب أثناء انعطافه.

آلية الدوران: القارب الذي اخترناه يعتمد على الدفع بواسطة محركين (- يتم التحكم بدوران القارب بتعديل سرعة

المحركين) وقمنا بمعالجة حالة تعطل أحد المحركين

1. حالة عمل المحركين: عند تشغيل محركين، تُنتج كل مروحة قوة دفع مختلفة، مما يؤدي إلى توليد Torque مختلف حول محور الدوران.

- التشغيل المتساوي للمحركين: عندما يعمل المحركان بنفس السرعة، يتحرك القارب للأمام بشكل مستقيم،

القوة الناتجة عن كلا المحركين تعمل على دفع القارب للأمام، مما يؤدي إلى زيادة السرعة.

- زيادة سرعة محرك واحد (مثل اليمين): يزيد Torque المروحة اليمنى، مما يؤدي إلى دوران القارب في اتجاه

عقارب الساعة، وكذلك الامر في حال زيادة المحرك الايسر سيدور باتجاه عكس عقارب الساعة

2. حالة تعطل أحد المحركات: يتم التحكم بدوران القارب باستخدام الدفة (Rudder):

تعتمد آلية دوران القارب باستخدام الدفة على مبدأ دفع الماء جانبياً، مما يؤثر على اتجاه حركة القارب.

- عند دوران الدفة إلى جانب واحد (مثل اليمين) يضغط الماء على الدفة بشكل أكبر على الجانب الأيمن تؤدي هذه

القوة الجانبية إلى دفع مؤخرة القارب إلى اليسار نتيجة لذلك، يدور القارب حول محوره الرأسي (vertical axis) في

اتجاه عقارب الساعة.

- العكس: عند دوران الدفة إلى الجانب الآخر (اليسار)، يدور القارب في عكس اتجاه عقارب الساعة.

عزم الدوران الناتج عن المحركات .

كل محرك يولد عزم دوران يؤثر على القارب بناءً على القوة الناتجة عنه والمسافة من المحور المركزي للقارب. يتم

حساب عزم الدوران باستخدام المعادلة:

$$\tau = F_{traction} \times d$$

τ هو عزم الدوران الناتج عن المحرك

$F_{traction}$ هي قوة الجر الناتجة عن المحرك

d هي المسافة بين المحرك والمحور المركزي للقارب

العزم الزاوي الكلي الناتج عن المحركين .

يتم حساب العزم الزاوي الكلي الناتج عن المحركين عبر الفرق بين العزمين الناتجين عن المحرك الأيمن والمحرك

الأيسر:

$$\tau_{dual} = \tau_{right} - \tau_{left}$$

إذا كانت القوتان متساويتين، فلن يكون هناك دوران .

أما إذا كانت قوة أحد المحركات أكبر، فإن ذلك يؤدي إلى دوران القارب.

وفي حالة تعطل احد المحركات يحسب τ_{dual} عن طريق قوة الدفع للمحرك السليم مضروب ب $d \sin()$

$$\tau_{dual} = d \times F_{traction} \times \sin(\text{angleRudder})$$

التسارع الزاوي .

باستخدام قانون نيوتن الثاني للحركة الدورانية

$$\alpha = \frac{\tau_{dual}}{I}$$

I_{Δ} (Moment of Inertia) عزم القصور الذاتي

والذي يعتمد على شكل القارب وتوزيع الوزن فيه، يمكن استخدام تقريب يعتمد على تحديد عزم القصور الذاتي

على شكل أسطواني

$$I = \frac{1}{12} m l^2 + \frac{1}{4} m \left(\frac{b}{2}\right)^2$$

m هي كتلة القارب

l هو طول القارب

b هو عرض القارب

السرعة الزاوية وزاوية الدوران

بناءً على التسارع الزاوي، يمكن حساب السرعة الزاوية الجديدة وزاوية الدوران للقارب باستخدام المعادلات

التالية:

$$\omega = \omega_0 + \alpha \cdot \Delta t$$

$$\theta = \theta_0 + \omega \times \Delta t + \frac{1}{2} \times \alpha \times \Delta t^2$$

- الخلاصة

يعتمد انعطاف القارب على توليد عزم الدوران الناتج عن المحركات، والذي يؤدي إلى تسارع زاوي يؤدي بدوره إلى

، تغيير اتجاه القارب. يتم تحديد مدى تأثير هذه القوى والعزوم باستخدام القوانين الفيزيائية المرتبطة بالقوة

العزم، والتسارع الزاوي.

القوانين الفيزيائية المؤثرة

تعتمد حركة القارب على مجموعة من القوانين الفيزيائية التي تحكم القوة والعزم والتسارع. ومن هذه القوانين:

معادلة القوة والتسارع الخطي

$$F = m \times a$$

F هي القوة

M هي كتلة القارب

a هو التسارع الخطي

العلاقة بين التسارع الخطي والتسارع الزاوي

$$a = r \times \alpha$$

a هو التسارع الخطي

r هو نصف القطر (المسافة من المحور إلى نقطة تطبيق القوة)

α هو التسارع الزاوي

معادلة عزم الدوران والتسارع الزاوي

في العلاقة السابقة r من خلال ضرب الطرفين في نصف القطر:

$$rF = m \times r^2 \times \alpha$$

rF هو العزم الناتج عن القوة المؤثرة










$I = mr^2$ هو عزم القصور الذاتي (العطالة)

وعليه:

$$\Gamma = I \alpha$$

Γ هو عزم الدوران

α هو التسارع الزاوي

Shape	Axis	Equation	Image
Slender Rod	through center	$I = \frac{1}{12} \cdot M \cdot L^2$	
Slender Rod	through end	$I = \frac{1}{3} \cdot M \cdot L^2$	
Rectangular Plane	through center	$I = \frac{1}{12} \cdot M \cdot (a^2 + b^2)$	
Rectangular Plane	along edge	$I = \frac{1}{3} \cdot M \cdot a^2$	
Sphere	thin-walled hollow	$I = \frac{2}{3} \cdot M \cdot R^2$	
Sphere	solid	$I = \frac{2}{5} \cdot M \cdot R^2$	
Cylinder	hollow	$I = \frac{1}{2} \cdot M \cdot (R_1^2 + R_2^2)$	
Cylinder	solid	$I = \frac{1}{2} \cdot M \cdot R^2$	
Cylinder	thin-walled hollow	$I = M \cdot R^2$	

الدراسة الخوارزمية والبرمجية

مقدمة:

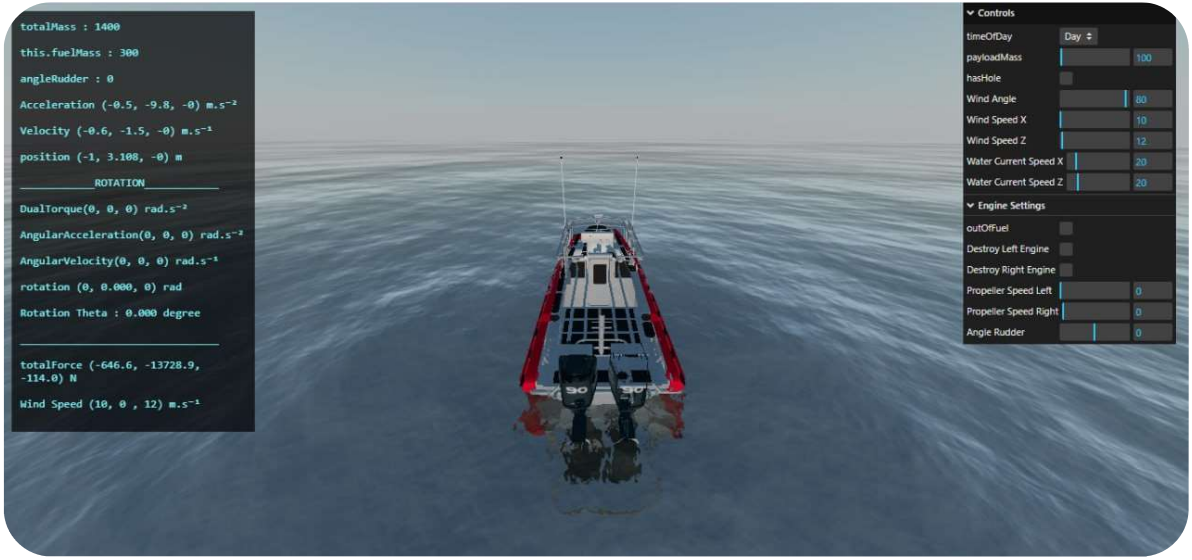
تهدف الدراسة الخوارزمية لمشروع الحسابات العلمية إلى فهم وتحديد آلية العمل ضمن البرنامج وتحديد المتغيرات وآلية الإدخال والإخراج وكيف سيقوم البرنامج بأداء عمله قبل تطوير أي تطبيق برمجي يخص الظاهرة الفيزيائية المعنية. في هذه الدراسة سنهدف ونتطرق إلى شرح مكونات المشروع بشكل برمجي بما فيها من صفوف ومتغيرات وتوابع وعلاقاتها بين بعضها البعض مع توضيح شكل التطبيق وكيف يمكن للمستخدم التعديل على بعض المتغيرات.

بيئة البرنامج ومبدأ عمله:

في مشروعنا سيتم اعتماد مكتبة **Three.js** لرسم البيئة والسفينة وربط الفيزياء بالبرمجة. وهي مكتبة بلغة **JavaScript** تستخدم مع متصفحات الويب وواجهة برمجة تطبيقات **API** وتستخدم لإنشاء رسومات كمبيوتر متحركة ثلاثية الأبعاد وعرضها في مستعرض ويب باستخدام **WebGL**. إن نموذج الواجهة

الرسومية GUI الخاص بالمشروع بسيط للغاية بحيث يحقق المطلوب. سيكون عبارة عن واجهة متصفح تقوم تلقائياً بعرض المشهد سيتم عرض حركة السفينة حيث يمكن مشاهدة سيرها وحركتها ضمن البيئة الخارجية. كما سيكون في واجهة المشهد الرئيسية شيء شبيه ب خاصية بالمتغيرات الفيزيائية القابلة للتعديل من قبل المستخدم (والتي سيتم ذكرها لاحقاً لوحة تحكم) **Panel Control** من خلالها يتم عرض قيم افتراضية لها حسب حركة القارب ونوعها ويمكن للمستخدم تغيير أي قيمة بحيث يتم عرض الخرج **Output** الخاص بحالة القارب ككل بشكل تلقائي عند تغيير القيم وهذا هو الهدف من ربط الفيزياء الخاصة بالظاهرة الفيزيائية مع البرمجة.

نموذج السفينة المعتمد



وطبعاً تتميز هذه السفينة بالعديد من الخصائص والمتغيرات الفريدة الخاصة بها والتي يمكن أن تساعد البعض منها في دراستنا بتحديد القيم الابتدائية ومنها:

1. الخصائص العامة للسفينة:

هي الخصائص التي تصف البنية الأساسية للسفينة ولا تتغير أثناء التشغيل.

- طول السفينة

(lengthBoat): this.lengthBoat = 4

- عرض السفينة

(wBoat): this.wBoat = 1

- ارتفاع السفينة

(heighBoat): this.heighBoat = 1.5

- كتلة السفينة

boatWeight = 1000

- كثافة الماء

(p_w): this.p_w = 1025

- المتغيرات القابلة للتعديل:

هي المتغيرات التي يمكن تعديلها أو تغييرها لتغيير سلوك السفينة في سيناريوهات مختلفة

- زاوية التيار المائي (theta): this.theta

- سرعة التيار المائي على محور Y: (v_c_y): this.v_c_y

- سرعة التيار المائي على محور X: (v_c_x): this.v_c_x

- سرعة الرياح على محور X: (windSpeedX): this.windSpeedX

- سرعة الرياح على محور Z: (windSpeedZ): this.windSpeedZ

- سرعة دوران مروحة المحرك الأيمن: propellerSpeedR

- سرعة دوران مروحة المحرك الأيسر: propellerSpeedL

- تعطل المحرك الأيمن: destoryRightEngine

- تعطل المحرك الأيسر: destoryLeftEngine

- زاوية دوران الدفة: angleRudder

- حمولة القارب: payloadMass

- حالة حدوث ثقب في القارب: hasHole

- حالة نفاذ الوقود

التصميم التقني (شرح مكونات البرنامج – الصفوف):

❖ المكاتب المستخدمة:

OIMO: مكتبة فيزيائية لمحاكاة الاصطدامات والحركة الفيزيائية.

THREE: مكتبة لإنشاء الرسومات ثلاثية الأبعاد.

InputManager: وحدة لإدارة إدخال المستخدم من خلال لوحة المفاتيح (المفاتيح المضغوطة).

لسهولة العمل ضمن المشروع يمكن تقسيم المشروع ك كل إلى عدة صفوف Classes تحوي واصفات Attributes منها

الثابتة Constants التي ال تتغير قيمتها طوال تشغيل المشهد ومنها المتغير Variables والتي يمكن للمستخدم

التعديل عليها ضمن لوحة التحكم للواجهة الرسومية GUI الخاصة بالبرنامج. كما تحتوي الصفوف على توابع

Functions من خلالها يتم التفاعل بين الصفوف وتستخدم المتغيرات الفيزيائية طالما المشروع يعمل. ومن هذه

الصفوف هي:

Boat -

- هذا الكلاس يمثل القارب في المحاكاة. يقوم بتحميل نموذج ثلاثي الأبعاد للقارب باستخدام مكتبة

GLTFLoader، ومن ثم يضيفه إلى المشهد (**scene**) يتم تحديث موقع القارب ودورانه في كل إطار بناءً

على القيم المحسوبة من الكلاس الفيزيائي (**Ph**)

الوظائف الرئيسية:

- **constructor()**

يقوم بتحميل نموذج القارب وإعداداته في المشهد. يتم ضبط حجم القارب، موقعه، ودورانه الابتدائي.

يتم حفظ المرجع إلى النموذج المحمل في المتغير **boatModel**

- **update()**

- يتحقق ما إذا كان النموذج (**boatModel**) قد تم تحميله بالفعل.

- يقوم بتحديث دوران القارب باستخدام الوظائف المتاحة في الكلاس الفيزيائي (**Ph**).

- يتم تحديث موقع القارب في الفضاء ثلاثي الأبعاد بناءً على القيم المحسوبة من الكلاس الفيزيائي.

- يتم تحديث اتجاه القارب بناءً على دورانه المحسوب.
- يتم رسم أشعة القوة (**force vectors**) التي تمثل القوى المؤثرة على القارب (مثل قوة الجبر، الرياح، التيارات المائية).

- Ph

- هذا الكلاس يمثل المحاكاة الفيزيائية لحركة القارب. يحسب القوى المختلفة التي تؤثر على القارب، مثل قوة الجاذبية، الطفو، السحب، الدفع، تأثير التيارات المائية، والرياح. ثم يستخدم هذه القوى لتحديث موقع القارب، سرعته، وتسارعه في الفضاء ثلاثي الأبعاد.

الوظائف والخصائص الرئيسية:

- **constructor()**
 - إعداد الخصائص الفيزيائية الأساسية مثل الكتلة (**mass**) ، الموقع (**position**) ، السرعة (**velocity**) ، التسارع (**acceleration**) ، وغيرها.
 - إعداد بعض الخصائص الإضافية المتعلقة بالقارب مثل معامل السحب، كثافة الماء، كثافة الهواء، حجم القارب، وعزم القصور الذاتي (**moment of inertia**) .
- **update()**
 - حساب القوى الكلية (**totalForce**) المؤثرة على القارب وتحديث التسارع بناءً على هذه القوى.
 - تحديث السرعة والموقع بناءً على التسارع المحسوب.
 - التحكم في ارتفاع القارب لضمان عدم تجاوزه حدًا معينًا.
 - زيادة الكتلة في حالة وجود ثقب في القارب (**hasHole**) .

• **totalForce()**

- يجمع جميع القوى المؤثرة على القارب (الجاذبية، الطفو، السحب، الدفع، الرياح، التيارات) ليحسب القوة الكلية.

- الدوال المتعلقة بالقوى:

- قوة التيارات `CurrentInducedForce()`: تحسب القوة الناتجة عن التيارات المائية بناءً على سرعة التيار وزاوية الاتجاه.

- قوة الرياح `windForce()`: تحسب القوة الناتجة عن الرياح بناءً على سرعة الرياح وزاوية الاتجاه.
- قوى الدفع:

- `tractionForceRight()`: حساب قوة الدفع من المحرك الأيمن.

- `tractionForceLeft()`: حساب قوة الدفع من المحرك الأيسر.

- `updateTraction()`: حساب القوة الناتجة عن المحركات.

- قوة السحب `dragForce()`: حساب قوة السحب.

- قوة الطفو `buoyancyForce()`: حساب قوة الطفو.

- قوة الجاذبية `gravityForce()`: حساب قوة الجاذبية.

- الدوال المتعلقة بالدوران:

- `calculate_I()`: حساب العزم الذاتي للقارب.

- `updateRightTorque()`: حساب العزم الناتج عن المحرك الأيمن.

- `updateLeftTorque()`: حساب العزم الناتج عن المحرك الأيسر.

- `updateDualTorque()`: حساب العزم الناتج عن كلا المحركين.

- `updateAngularAcceleration()`: حساب التسارع الزاوي.

- `updateAngularVelocity()`: حساب السرعة الزاوية.

- `updateTheta()`: حساب الزاوية الناتجة عن الدوران.

- `updateRotation()`: تحديث دوران القارب بناءً على الزاوية.

طريقة حساب قوة الدفع :

```
1  const enginePowerR = 10000;
2  const propellerEfficiencyR = 0.1
3  const propellerAreaR = 0.1;
4
5  const enginePowerL = 10000;
6  const propellerEfficiencyL = 0.1
7  const propellerAreaL = 0.1;
8
9  const waterDensity = 1025;
```

دفع المحرك الأيمن (Right Traction Force) :

```
1  tractionForceRight() {
2    const force =
3      !this.destoryRightEngine * 0.5 * waterDensity * propellerAreaR *
4      Math.pow(this.propellerSpeedR, 2) * propellerEfficiencyR;
5    const maxForce = enginePowerR;
6    const thrust = Math.min(force, maxForce);
7    if (this.propellerSpeedR > 30) {
8      this.fuelMass -= 0.003;
9    } else if (this.propellerSpeedR > 2) {
10     this.fuelMass -= 0.001;
11   }
12   return thrust;
13 }
```

دفع المحرك الأيسر (Left Traction Force) :

```
1  tractionForceLeft() {
2    const force =
3      !this.destoryLeftEngine * 0.5 * waterDensity * propellerAreaL *
4      Math.pow(this.propellerSpeedL, 2) * propellerEfficiencyL;
5    const maxForce = enginePowerL;
6    const thrust = Math.min(force, maxForce);
7    if (this.propellerSpeedL > 30) {
8      this.fuelMass -= 0.003;
9    } else if (this.propellerSpeedL > 2) {
10     this.fuelMass -= 0.001;
11   }
12   return thrust;
13 }
```

هذا الجزء يتحقق مما إذا كان المحرك الأيمن معطلاً (destoryRightEngine) ، فإذا كان كذلك، فإن القوة تصبح صفرًا.

```
1 const force =  
2   !this.destoryRightEngine * 0.5 * waterDensity * propellerAreaR * Math.pow(this.propellerSpeedR, 2) * propellerEfficiencyR;
```

```
const maxForce = enginePowerR;
```

القوة القصوى التي يمكن أن يولدها المحرك الأيمن، وهي محددة مسبقًا بـ 10000 نيوتن.

```
const thrust = Math.min(force, maxForce);
```

يتم تحديد قيمة قوة الدفع (thrust) كأقل قيمة بين القوة المحسوبة (force) والقوة القصوى (maxForce).

هذا لضمان عدم تجاوز القوة الناتجة حدود القوة القصوى للمحرك.

```
return thrust;
```

يعيد القيمة النهائية لقوة الدفع التي يولدها المحرك الأيمن وكذلك الأمر للمحرك الأيسر

من أجل حساب قوة الدفع الكلية

```
1 updateTraction() {  
2   this.direction = this.compute_direction(this.rotation.y);  
3   const temp = this.tractionForceRight() + this.tractionForceLeft();  
4   const T = this.direction.clone().multiplyScalar(temp);  
5   return T;  
6 }
```

```
this.direction = this.compute_direction(this.rotation.y);
```

يتم حساب اتجاه الدفع بناءً على زاوية دوران السفينة (rotation.y) .

وأيضاً الجزء المتعلق بدوران القارب :

`updateRightTorque` تقوم بحساب العزم الناتج عن المحرك الأيمن

```
1  updateRightTorque() {  
2    const z = this.tractionForceRight() * d;  
3    return new THREE.Vector3(0, 0, z);  
4  }
```

`updateLeftTorque` مثل العزم الناتج عن المحرك الأيمن، ولكن هنا للمحرك الأيسر.

```
1  updateLeftTorque() {  
2    const z = this.tractionForceLeft() * d;  
3    return new THREE.Vector3(0, 0, z);  
4  }
```

`updateDualTorque`

يحسب العزم الناتج عن تشغيل كلا المحركين. إذا كان أحد المحركين معطلاً، يتم حساب العزم بناءً على المحرك

الآخر وزاوية الدفة (`angleRudder`) إذا كان كلا المحركين يعملان، يتم حساب الفرق بين العزم الناتج عن

المحرك الأيمن والمحرك الأيسر

```
1  updateDualTorque() {  
2    if (this.destoryLeftEngine) {  
3      const z =  
4        this.updateRightTorque().z * d *  
5        Math.sin((this.angleRudder * Math.PI) / 180);  
6      return new THREE.Vector3(0, 0, z);  
7    } else if (this.destoryRightEngine) {  
8      const z =  
9        this.updateLeftTorque().z *  
10       d *  
11       Math.sin((this.angleRudder * Math.PI) / 180);  
12      return new THREE.Vector3(0, 0, z);  
13    }  
14    return this.updateRightTorque().sub(this.updateLeftTorque());  
15  }
```


updateRotation تقوم بتحديث زاوية دوران القارب (**rotation.y**) بناءً على عزم الدوران الناتج عن

المحركات. يتم تعديل متجه السرعة بناءً على هذا التغيير في الزاوية باستخدام الكواتيرنيون (**Quaternion**) للتأكد من أن جميع المتجهات متناسقة مع الدوران.

```
1  updateRotation() {
2    this.rotation.y += this.updateTheta();
3    const q1 = new THREE.Quaternion().setFromAxisAngle(
4      new THREE.Vector3(0, 1, 0),
5      this.updateTheta()
6    );
7    this.velocity.applyQuaternion(q1);
8  }
9
```

وأيضاً عالجنا في الكود حالات نفاذ الوقود وإذا كانت السفينة تحوي على ثقب

```
if (this.propellerSpeedR > 30) {
  this.fuelMass -= 0.003;
} else if (this.propellerSpeedR > 2) {
  this.fuelMass -= 0.001;
}
```

```
this.fuelMass = 300;
```

fuelMass يمثل كمية الوقود المتبقية في القارب. في كل مرة يتم فيها تشغيل المحركات، ينقص الوقود

أثناء حساب قوة الدفع لكل من المحرك الأيمن والأيسر، يُخصم الوقود بناءً على سرعة المروحة.

```
if (this.hasHole && this.totalMass() <= 5800) {
  this.holeMass += 15;
}
```

```
this.hasHole = false
```

```
this.holeMass = 0
```

hasHole متغير بولياني يدل على وجود ثقب في السفينة أم لا. إذا كان true، فهذا يعني أن هناك ثقبًا يتسبب في

دخول الماء وزيادة وزن السفينة.

holeMass يمثل وزن الماء الذي يدخل إلى السفينة بسبب الثقب.

إذا كان هناك ثقب (**hasHole = true**) ولم يتجاوز الوزن الإجمالي للسفينة 5800، يزداد وزن الماء داخل السفينة

بمقدار 15 وحدة في كل دورة.

Date GUI

Output

```
totalMass : 1400
this.fuelMass : 300
angleRudder : 0
Acceleration (0.3, 21.5, 0) m.s-2
Velocity (-0.6, 4.8, -0) m.s-1
position (-2, 0.504, -0) m
_____
ROTATION
DualTorque(0, 0, 0) rad.s-2
AngularAcceleration(0, 0, 0) rad.s-2
AngularVelocity(0, 0, 0) rad.s-1
rotation (0, 0.000, 0) rad
Rotation Theta : 0.000 degree
_____
totalForce (408.4, 26278.8, 72.0) N
Wind Speed (10, 0, 12) m.s-1
```

Controls

▼ Controls

timeOfDay

Day ↕

payloadMass

100

hasHole

☐

Wind Angle

80

Wind Speed X

10

Wind Speed Z

12

Water Current Speed X

20

Water Current Speed Z

20

▼ Engine Settings

outOfFuel

☐

Destroy Left Engine

☐

Destroy Right Engine

☐

Propeller Speed Left

0

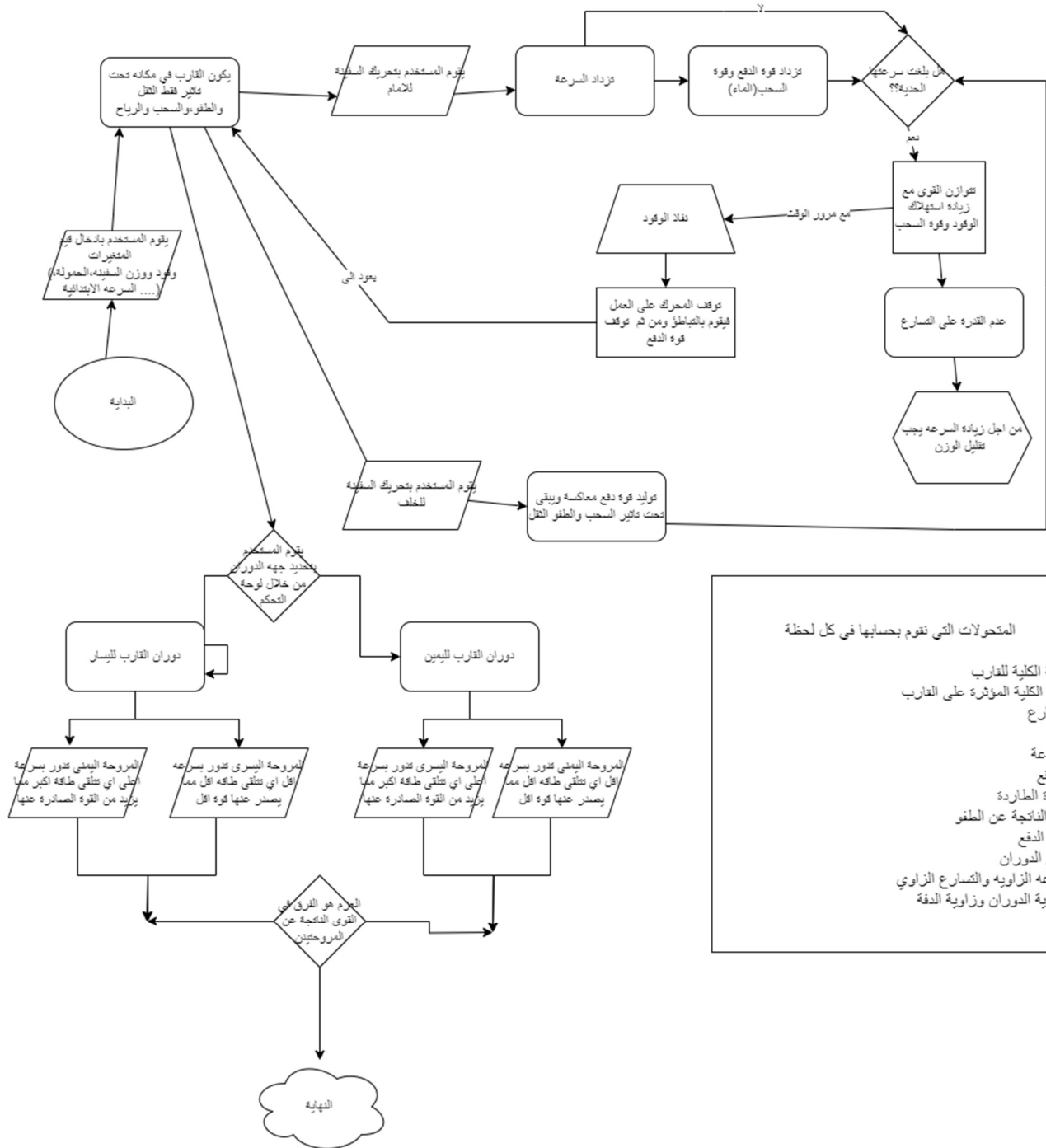
Propeller Speed Right

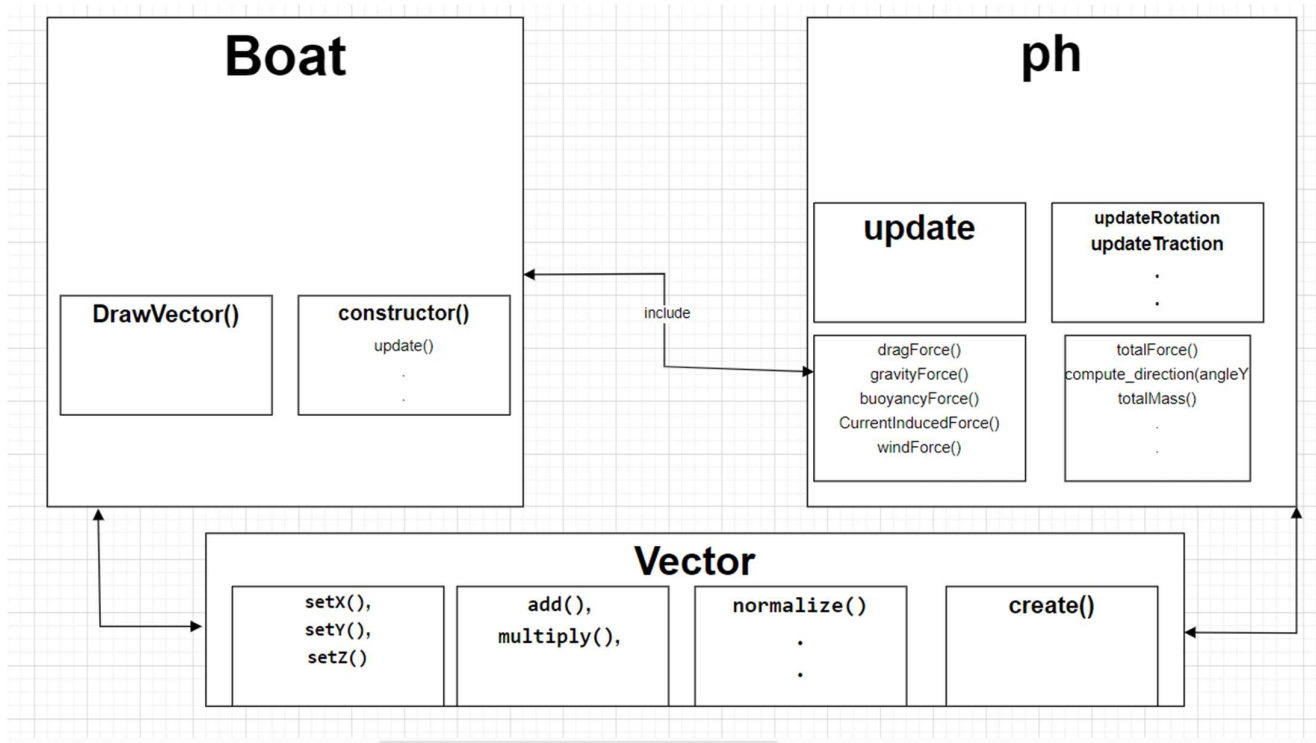
0

Angle Rudder

0

المخطط التدفقي





المشاكل وحلها

❖ واجهتنا بعض الصعوبات عند محاولة تدوير القارب حول المحاور ودمج الحركة الانسحابية مع الدورانية وضبط

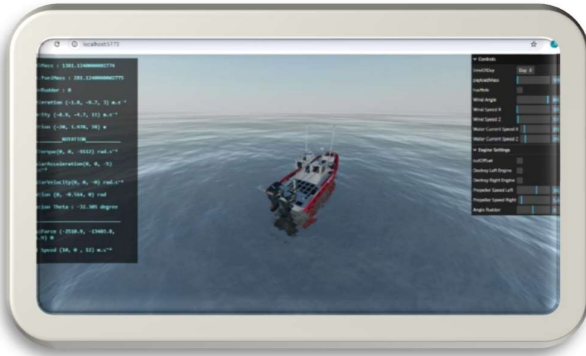
حركة القارب

❖ بعض القيم العلمية لم تنجح اثناء المحاكاة لذلك اضطررنا إعطاء قيم تجريبية

❖ بعض القوانين كان من الصعب تطبيقها بشكل دقيق على جسم القارب

مباشرة لذلك قربنا بعض أجزاء القارب إلى أجسام متناظرة مثل المستطيل

صور توضيحية لمحاكاة لحركة القارب :



Rudder right



Rudder left

توابع القوى

```

1 totalMass() {
2   const total = boatWeight + this.payloadMass + this.fuelMass + this.holeMass;
3   return total;
4 }
5
6 update() {
7   if (this.hasHole && this.totalMass() <= 5800) {
8     this.holeMass += 15;
9   }
10  var totalF = this.totalForce();
11
12  this.acceleration = totalF.divideScalar(this.totalMass());
13
14  this.velocity = this.velocity
15    .clone()
16    .add(this.acceleration.clone().multiplyScalar(0.02));
17  this.vlength = this.velocity.length();
18
19  this.position = this.position
20    .clone()
21    .add(this.velocity.clone().multiplyScalar(0.02));
22
23  if (this.position.y > 5) {
24    this.position.y = 5;
25  }
26 }

```

```

1 totalForce() {
2   var tf = new THREE.Vector3();
3   tf = tf.add(this.gravityForce());
4   tf = tf.add(this.buoyancyForce());
5
6   if (this.velocity.x !== 0 || this.velocity.z !== 0) {
7     tf = tf.add(this.dragForce());
8   }
9   tf = tf.add(this.updateTraction());
10  tf = tf.add(this.CurrentInducedForce());
11  tf = tf.add(this.windForce());
12  return tf;
13 }
14
15 compute_direction(angleY) {
16   return new Vector3(Math.sin(angleY), 0, Math.cos(angleY));
17 }

```




```
1  buoyancyForce() {
2    var buoyancy = this.volume() * waterDensity * this.gravity.length();
3    var buoyancyVector = new Vector3(0, buoyancy, 0);
4    return buoyancyVector;
5  }
6
7  gravityForce() {
8    var gravityForce = -(this.totalMass() * this.gravity.length());
9    var gravityForceVector = new Vector3(0, gravityForce, 0);
10   return gravityForceVector;
11 }
12 // قوة التيارات
13 CurrentInducedForce() {
14   // تحويل الزاوية من درجات إلى راديان
15   var thetaRad = (this.theta * Math.PI) / 180;
16
17   var F_c_x =
18     0.5 *
19     waterDensity *
20     Math.pow(this.v_c_x, 2) *
21     0.5 *
22     this.lengthBoat *
23     (this.heighBoat - this.position.y) *
24     this.dragCoefficient;
25   var F_c_z =
26     0.5 *
27     waterDensity *
28     Math.pow(this.v_c_z, 2) *
29     0.5 *
30     this.lengthBoat *
31     (this.heighBoat - this.position.y) *
32     this.dragCoefficient;
33
34   var F_c_Vector = new Vector3(
35     F_c_x * Math.sin(thetaRad),
36     0,
37     F_c_z * Math.cos(thetaRad)
38   );
39   return F_c_Vector;
40 }
41
42 windForce() {
43   // تحويل الزاوية من درجات إلى راديان
44   var thetaRad = (this.theta * Math.PI) / 180;
45   var windForceX =
46     this.air_rho *
47     Math.pow(this.windSpeedX, 2) *
48     (this.wBoat * this.lengthBoat) *
49     this.dragCoefficient;
50   var windForceZ =
51     this.air_rho *
52     Math.pow(this.windSpeedZ, 2) *
53     (this.wBoat * this.lengthBoat) *
54     this.dragCoefficient;
55
56   var windForceVector = new Vector3(
57     windForceX * Math.sin(thetaRad),
58     0,
59     windForceZ * Math.cos(thetaRad)
60   );
61
62   return windForceVector;
63 }
```

رقم المرحلة	اسم المرحلة	المشاركون
1	الدراسة الفيزيائية	كل المجموعة
2	الدراسة الخوارزمية	كل المجموعة
3	تطبيق البرمجة	كل المجموعة
4	رسم البيئة	كل المجموعة

المشكلات والتطويرات المستقبلية :

كان من الممكن معالجة حالة الصدم ولكن لم يسعفنا الوقت فاكثفينا بذكر كل ما هو أعلاه.

المراجع:

1. Fundamentals of Aerodynamics by John D. Anderson Jr.
2. https://en.wikipedia.org/wiki/Boeing_AH-64_Apache#Development
3. Aerodynamics for Engineering Students by E. L. Houghton and P. W. Carpenter
4. Serway, R. A., & Jewett, Jr., J. W. (2013). Physics for scientists and engineers with modern physics. Cengage Learning .
5. Young, H. D., & Freedman, R. A. (2012). University physics with modern physics. Pearson Education .
6. Federal Aviation Administration. (2016). Helicopter Flying Handbook. U.S. Department of Transportation .
7. Principles of Helicopter Aerodynamics by J. Gordon Leishman
8. Introduction to Flight by John D. Anderson Jr .
9. Helicopter Performance, Stability, and Control by Raymond W. Prouty.
10. Understanding Helicopter Aerodynamics by R. W. Prouty