

**Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Дагестанский государственный технический университет»**

**УДК 621.396**

*На правах рукописи*



**Абдуллаева Залина Мусаевна**

**РАЗРАБОТКА И РЕАЛИЗАЦИЯ  
МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ ДВИЖЕНИЯ  
СУДНА НА МЕЛКОВОДЬЕ ПРИ ПЕРЕМЕННОЙ  
ГЛУБИНЕ**

**Специальность:**

**05.13.18 - Математическое моделирование, численные методы и  
комплексы программ**

**диссертация  
на соискание ученой степени  
кандидата технических наук**

**Научный руководитель:  
доктор технических наук, профессор  
Асланов Гайдарбек Кадырбекович**

**Махачкала - 2017**

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
ГЛАВА 1. ТЕОРЕТИЧЕСКИЙ АНАЛИЗ ДВИЖЕНИЯ СУДНА .....	11
1.1 Математические модели движения судна на глубокой воде.....	11
1.2 Математические модели движения судна на мелководье .....	17
1.3 Влияние мелководья на гидродинамические характеристики корпуса судна	19
1.4 Влияние мелководья на коэффициенты нелинейности поперечной силы и момента.....	23
Выводы по первой главе.....	27
ГЛАВА 2. АНАЛИТИЧЕСКОЕ ОПИСАНИЕ КРИВЫХ ВЛИЯНИЯ МЕЛКОВОДЬЯ НА ГИДРОДИНАМИЧЕСКИЕ КОЭФФИЦИЕНТЫ .....	28
2.1 Аналитическое описание влияния мелководья на гидродинамические характеристики корпуса судна.....	28
2.2 Аналитическое описание влияния мелководья на коэффициенты нелинейности поперечной силы и момента .....	34
2.3 Математическая модель движения судна на мелководье .....	39
Выводы по второй главе .....	45
ГЛАВА 3. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ДВИЖЕНИЯ СУДНА НА МЕЛКОВОДЬЕ ПО КРИВОЛИНЕЙНОЙ ТРАЕКТОРИИ.....	46
3.1. Постановка задачи, допущения и ограничения, принимаемые при моделировании движения судна на мелководье по заданной криволинейной траектории.....	46
3.2. Геометрические соотношения, возникающие при движении судна по криволинейной траектории .....	47
3.3. Выбор параметров авторулевого .....	52
3.4 Задание формы дна .....	60
Выводы по третьей главе.....	63
ГЛАВА 4. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ МАТЕМАТИЧЕСКОЙ МОДЕЛИ ДВИЖЕНИЯ СУДНА НА МЕЛКОВОДЬЕ .....	64

4.1. Общая характеристика программного комплекса .....	64
4.2 Выбор среды разработки .....	66
4.3 Разработка блок-схемы алгоритма моделирования движения судна на мелководье по заданной криволинейной траектории .....	69
4.4 Разработка программы, моделирующей движение судна на мелководье по криволинейной траектории .....	72
4.5 Анализ результатов моделирования движения судна по заданной траектории .....	74
Выводы по четвертой главе.....	91
ЗАКЛЮЧЕНИЕ .....	92
СПИСОК ЛИТЕРАТУРЫ.....	94
Приложение А .....	103
Приложение Б.....	111
Приложение В.....	185
Приложение Г .....	203
Приложение Д.....	224
Приложение Е.....	245

## ВВЕДЕНИЕ

**Актуальность темы исследования.** Одним из основных путей повышения эффективности работы морского флота является определение условий безопасного плавания различных судов на мелководье с переменной глубиной. Другими словами, обеспечение безопасного плавания, например, в акваториях портов, при прохождении рек и каналов в сложных гидрометеорологических условиях является одной из актуальных проблем судовождения. Это обусловлено тем, что по статистике в перечисленных выше условиях происходит около 90% навигационных аварий, и, в первую очередь, крупнотоннажных судов. Основная причина таких аварий связана с выбором тактики маневрирования, базирующейся, в основном, на опыте судоводителей, приобретенном ими, как правило, в процессе плавания на глубокой воде, т.е. с тем, что судоводитель, фактически не обладая априорными достоверными данными, проводит маневрирование на основе субъективной оценки возникающих ситуаций [7,50,52,80,85].

Однако, на мелководье, необходимо объективно учитывать соотношение между глубиной и осадкой судна, т.к. в этом случае возникают силы гидродинамического взаимодействия его корпуса и грунта, приводящие к ухудшению маневренности судов, повышению уровня рыскливости, а при неизменном значении угла перекладки руля к увеличению радиуса установившейся циркуляции. Таким образом, при плавании на мелководье, особенно в условиях с изменяющейся глубиной, т.е. с изменяющимся рельефом дна необходима повышенная точность счисления траектории движения и высокая точность определения координат местоположения судна в заданной системе отчета [44].

Следует отметить, что решение перечисленных выше задач, связанных с безопасностью судовождения, может быть выполнено путем проведения сложных натурных экспериментов. Однако, реализация таких экспериментов в естественных условиях порой невозможна или экономически нецелесообразна. Обойти указанную трудность можно путем математического моделирования

процессов движения судов на мелководье. Необходимо также отметить, что замена натурного эксперимента математическим моделированием для исследования влияния мелководья на динамику судна, кроме снижения колоссальных затрат времени и средств, позволяет обеспечить повторяемость результатов проводимых экспериментов при различных условиях плавания и обобщить получаемые в результате данные.

Таким образом, математическое моделирование является одним из эффективных способов получения достоверных данных, необходимых для принятия эффективных навигационных решений судоводителями, в процессе маневрирования судов в условиях мелководья.

Результаты математического моделирования движения судов на мелководье при переменной глубине и ходовых испытаний, позволяют получить достоверные данные в общем виде безотносительно к конкретным условиям плавания и использовать их с учетом коррекции на фактические условия различных видов мелководья для обеспечения требуемой точности и безопасности маневров, проводимых судоводителями.

Следовательно, возникает объективная необходимость в разработке адекватных математических моделей и их программная реализация для получения достоверных данных, используемых судоводителями для принятия эффективных решений в процессе маневрирования судов на мелководье с переменной глубиной, что и определяет актуальность темы настоящей диссертационной работы.

**Степень разработанности исследуемой проблемы.** Объективная необходимость математического моделирования поведения судов на мелководье за последние десятилетия существенно повысилась в связи с возрастанием интенсивности мореплавания, а также с повышением объемов перевозок опасных грузов, в частности, на танкерном флоте. Автор, при проведении своего исследования опирался на результаты, полученные в рассматриваемой области следующими учеными: Аслановым Г.К., Басиным А.М., Войткунским Я.М., Гофманом А.Д., Павленко В.Г., Першицем Р.Ж., Соболевым Г.В., Федяевским

К.К., Шлейером Г.Э., Острцовым Г. Э. и мн. др. [7,19, 20, 31, 32, 34, 44, 73, 74, 77, 84, 85, 86].

Однако использование известных линейных или частично линеаризованных моделей движения судна, разработанных различными авторами и описанных в большом количестве литературных источников по данной тематике, в настоящее время является не достаточно эффективным. Это связано с тем, что в данных математических моделях движения различных судов, при наличии возмущающих воздействий внешней среды, практически не учитывается переменная глубина акватории плавания, существенным образом влияющая на динамику их движения на мелководье.

Необходимо также отметить, что в настоящее время хотя и разработаны математические модели движения судов по криволинейной траектории на мелководье, но в них не учитываются возмущения, связанные с переменной глубиной района плавания, существенно влияющие на их маневрирование. Другими словами, вопросы, связанные с исследованием влияния изменяющегося рельефа морского дна на мелководье на гидродинамические характеристики движения судов практически остаются открытыми.

Отмеченные выше обстоятельства и определили цели, задачи и направление настоящего диссертационного исследования.

**Целью диссертационного исследования** является разработка математических моделей движения судов по заданной криволинейной траектории на мелководье при переменной глубине, а также разработка на их основе пакета прикладных программ для проведения компьютерных экспериментов.

В соответствии с целью исследования в диссертации поставлены и решены следующие **задачи**:

- получить аналитические зависимости влияния мелководья на гидродинамические коэффициенты уравнений гидродинамики судна, определяемые как функции отношения осадки судна к глубине акватории;
- разработать математическую модель, используя аналитические зависимости влияния мелководья на гидродинамические коэффициенты в

уравнениях гидродинамики судна при его движении на мелководье с учетом переменной глубины района плавания;

- синтезировать математическую модель движения судна на мелководье при переменной глубине района плавания по заданной криволинейной траектории;

- разработать пакет прикладных программ, обеспечивающий возможность проведения исследования гидродинамики судов на мелководье с переменной глубиной, на основе разработанных математических моделей.

**Направление исследования:**

- *объектом исследования* являются процессы маневрирования и движение морских судов на мелководье с переменной глубиной безотносительно к конкретным условиям плавания;

- *предметом исследования* является система компьютерного моделирования гидродинамики судов и инструментальные средства управления их движением на мелководье с переменной глубиной района плавания;

- *исследование проведено* в рамках специальности 05.13.18 – математическое моделирование, численные методы и комплексы программ. Паспорт специальности ВАК (технические науки): п.4. Реализация эффективных численных методов и алгоритмов в виде комплексов проблемно-ориентированных программ вычислительного эксперимента; п.5. Комплексные исследования научных и технических проблем с применением современных технологий математического моделирования и вычислительного эксперимента; п.8. Разработка систем компьютерного и имитационного моделирования.

**Научная новизна проведенного исследования** заключается в разработке математических моделей движения судов на мелководье с переменной глубиной и криволинейной траекторией плавания на основе аналитических зависимостей его влияния на гидродинамические коэффициенты уравнений гидродинамики судна, определяемые как функции отношения их осадки к глубине акватории с помощью кривых третьего порядка, с обеспечением максимальной среднеквадратической погрешности аппроксимации  $\delta = 2\%$ .

**К новым научным результатам, выносимым на защиту, следует отнести:**

- путем аппроксимации кривых влияния мелководья на гидродинамику судна, получены закономерности изменения коэффициентов при переменных в уравнениях гидродинамики движения судна на мелководье в соответствии с изменениями рельефа дна в районе плавания. Это позволяет значительным образом повысить адекватность формируемых математических моделей и сократить объем материальных и временных затрат на проведение исследований по изучению динамики судов на мелководье;

- разработаны оригинальные математические модели движения судов на мелководье при циркуляции и заданной криволинейной траектории, отличающиеся от известных моделей учетом влияния изменения глубины фарватера на коэффициенты уравнений их гидродинамики, что позволяет определить условия безопасного маневрирования судов в условиях с переменной глубиной района плавания;

- разработана программная реализация предложенных математических моделей движения судов на мелководье с изменяющимся рельефом дна, что позволяет провести исследования и на основе полученных результатов дать рекомендации судоводителям по безопасному маневрированию судов на мелководье с переменной глубиной акватории плавания.

**Теоретическая значимость диссертационного исследования** заключается в том, что проведенные исследования вносят вклад в получение новых знаний в области математического моделирования движения судов на мелководье при циркуляции с переменной глубиной района плавания, а также при их движении на мелководье с переменной глубиной района плавания по заданной криволинейной траектории. Программная реализация разработанных математических моделей и построенных на их основе инструментальных средств управления маневрированием морских судов на мелководье с переменной глубиной района плавания позволяет проводить дальнейшие исследования их динамических характеристик, например, в условиях с волнения морской



поверхности.

**Практическая значимость результатов исследования** заключается в том, что разработанные математические модели и построенные на их основе инструментальные средства управления движением морских судов позволяют:

- значительным образом сократить объемы материальных и временных затрат на проведение исследований по изучению динамических характеристик различного класса судов на мелководье при переменной глубине района плавания и на этой основе дать практические рекомендации по их безопасному маневрированию;

- обеспечить возможность прогнозирования результатов совершения заданного маневра в текущих внешних условиях плавания;

- получить необходимые данные для разбора причин аварий в процессе судовождения на морском транспорте;

- дать практические рекомендации судоводителям по безопасному маневрированию на мелководье.

***Результаты диссертационной работы внедрены:***

- в учебный процесс в ФГБОУ ВПО «Дагестанский государственный технический университет» на кафедре управления и информатики в технических системах в методических указаниях к выполнению лабораторной работы «Моделирование движения судна на мелководье» по дисциплине «Моделирование и анализ сложных систем» для магистров направления подготовки 220200.68-Управление в технических системах;

- ряд теоретических положений и практических рекомендаций диссертационного исследования приняты к внедрению Морским филиалом ФГБУ «АМП Каспийского моря».

**Методы исследований, достоверность, обоснованность и апробация результатов исследования:**

- *методологические основы проведенного исследования* базируются на применении математических моделей и методов описания поведения динамических систем в нестабильной окружающей среде;

- **достоверность полученных в работе научных результатов** обоснована на эвристическом уровне строгости и подтверждается корректным использованием принципов математического моделирования динамических систем, известных уравнений гидродинамики судов, численных методов решения дифференциальных уравнений, методов Д-разбиения и аппроксимации кривых полиномами третьего порядка;

- **основные положения диссертации** докладывались и получили одобрение на: VI Всероссийской конференции по актуальным проблемам внедрения и развития сектора IT-технологий «Современные информационные технологии в проектировании, управлении и экономике» (28 – 29 сентября 2011 г., Махачкала); II Всероссийской научно-практической конференции «Молодежь, наука, инновации» (10-12 ноября 2013 года, Грозный); Всероссийской научно-технической конференции «Фундаментальные и прикладные проблемы математики, информатики в современной науке: теория и практика актуальных исследований» (19-21 октября 2016, Махачкала) и X Всероссийской научно-практической конференции «Современные информационные технологии в проектировании, управлении и экономике» - Махачкала, 2016; XXX, XXXIV итоговых научно-технических конференциях преподавателей, сотрудников, аспирантов и студентов ДГТУ «Неделя науки – 2009» и «Неделя науки – 2013».

- **оригинальность и работоспособность** разработанного на основе предложенных математических моделей комплекса программ подтверждается государственной регистрацией в реестре следующих программ для ЭВМ: «Моделирование движения судна на мелководье», регистрационный номер №2014662089 и «Моделирование движения судна на мелководье по криволинейной траектории», регистрационный номер №2016611225, а также на основе проведенных вычислительных экспериментов.

**Объем и структура диссертации.** Работа состоит из введения, четырех глав, заключения, списка литературы из 96 наименований и приложений. Основная часть работы изложена на 93 страницах машинописного текста, включая 52 рисунка, 3 таблицы, а также 6 приложений на 167 страницах.

## ГЛАВА 1. ТЕОРЕТИЧЕСКИЙ АНАЛИЗ ДВИЖЕНИЯ СУДНА

### 1.1 Математические модели движения судна на глубокой воде

Вопросам описания динамики судна посвящено достаточное количество работ, авторами которых являются: А.М. Басин, Я.М. Войткунский, А.Д. Гофман, В.Г. Павленко, Р.Ж. Першиц, Г.В. Соболев, К.К. Федяевский, Г. К. Асланов, и др. [6, 7, 21, 23, 29, 33, 40, 44, 51, 87]. Рассмотрим некоторые из них.

Система безразмерных дифференциальных уравнений для малых отклонений судна от криволинейной траектории в связанной с судном системе координат согласно [86] с учетом следующих допущений:

- судно считают твердым телом, симметричным относительно диаметральной плоскости, с центром тяжести, расположенным в плоскости миделя шпонгоута;

- пренебрегают влиянием крена и дифферента на величину гидродинамических сил и моментов, действующих на судно;

- предполагают, что движение жидкости безвихревое и зависит исключительно от движения судна;

- при определении сил и моментов инерционной природы, действующих на корпус судна, исходят из гипотезы стационарности, имеет вид (1.1):

$$\left. \begin{aligned} m_{22}\dot{\beta} + \bar{n}_y^{\beta} \beta - \bar{n}_y^{\omega} \omega + n\beta\omega \operatorname{sign}(\beta, \omega) &= n_y^{\delta} \delta \\ m_{66}\dot{\omega} - \bar{m}_y^{\beta} \beta - \bar{m}_y^{\omega} \omega - m\beta\omega \operatorname{sign}(\beta, \omega) &= n_y^{\delta} \delta \bar{l}_p \\ \left( \frac{\dot{\bar{V}}}{\bar{V}} \right) + \beta \left( \frac{m_{22}}{m_{11}} \omega - \dot{\beta} \right) &= \frac{1}{m_{11}} (n_p - n_x) \end{aligned} \right\} \quad (1.1)$$

где  $\bar{V}$  - линейная (приведенная) скорость судна  $\bar{V} = \frac{V}{V_0}$ ;

$V$  - текущее значение линейной скорости судна;

$V_0$  - начальное значение линейной скорости судна;

$\beta$  - угол дрейфа;

$\bar{\omega}$  - безразмерная угловая скорость судна  $\bar{\omega} = \frac{\Omega L}{V_0}$ ;

$L$  - длина судна между перпендикулярами;

$\Omega$  - угловая скорость судна;

$m_{11}, m_{22}, m_{66}$  - безразмерные гидродинамические коэффициенты;

$\tau$  - безразмерное время  $\tau = \frac{V_0}{L}$ ;

$n_x$  - безразмерный коэффициент сопротивления движению судна;

$n_p$  - безразмерный коэффициент тяги движителей;

$\delta$  - угол перекладки руля, выраженный в радианах и отсчитываемый от диаметральной плоскости судна, причем перекладка руля на правый борт соответствует  $\delta > 0$ , а на левый борт - значению  $\delta < 0$ ;

$\bar{l}_p = \frac{l_p}{L}$  - приведенное отстояние баллера руля от центра тяжести судна (безразмерное плечо руля);

$l_p$  - отстояние баллера руля от центра тяжести судна;

$n, m$  - коэффициенты нелинейности поперечной силы и момента.

$n_y^\delta$  - коэффициент, характеризующий эффективность установленных на судне средств управления;

$\bar{n}_y^\beta, \bar{m}_y^\beta$  - позиционные производные поперечной силы и момента рыскания;

$\bar{n}_y^\omega, \bar{m}_y^\omega$  - вращательные производные поперечной силы и момента рыскания;

Влияние мелководья на позиционные  $\bar{n}_y^\beta, \bar{m}_y^\beta$  и вращательные производные  $\bar{n}_y^\omega, \bar{m}_y^\omega$  поперечной силы и момента рыскания зависит от размерений судна

$$\lambda = \frac{2T}{H},$$

где,  $T$ -осадка судна на мидели;

$H$ -глубина воды.

Рисунок 1.1 поясняет обозначения, принятые в системе уравнений (1.1).

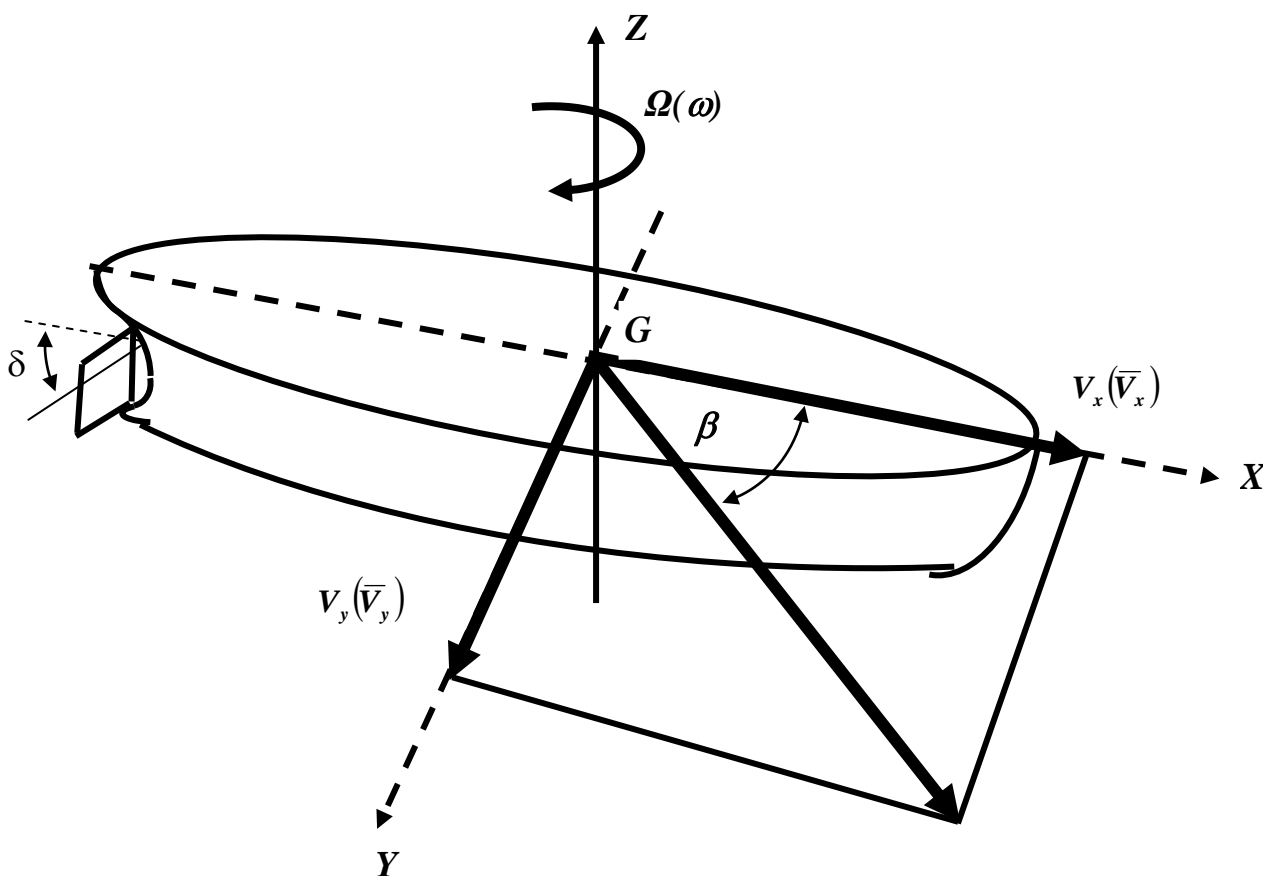


Рисунок 1.1 - Кинематические параметры плоского движения судна

В работе [45] описаны уравнения движения судна на глубокой воде (1.2) с учетом того, что гидродинамические силы  $X_G$ ,  $Y_G$ ,  $M_G$  определены с помощью ротативной установки, также считается, что центр тяжести судна совпадает с его миделем:

$$\begin{aligned}
 \frac{d\bar{V}}{d\tau} &= \left[ \frac{m_{26}}{\bar{M}} \bar{C}_m \sin\beta - \frac{m_{66}}{\bar{M}} \bar{C}_y \sin\beta - \frac{1}{m_{11}} \bar{C}_x \cos\beta \right] \bar{V}^2 \\
 \frac{d\beta}{d\tau} &= \left[ \frac{m_{26}}{\bar{M}} \bar{C}_m \cos\beta - \frac{m_{66}}{\bar{M}} \bar{C}_y \cos\beta - \frac{1}{m_{11}} \bar{C}_x \sin\beta \right] \bar{V} \\
 \frac{d\bar{\omega}}{d\tau} &= \left[ \frac{m_{26}}{\bar{M}} \bar{C}_m - \frac{m_{66}}{\bar{M}} \bar{C}_y \right] \bar{V}^2,
 \end{aligned} \tag{1.2}$$

где

$$\bar{C}_x = C_{x_r} + \frac{\rho_1 S_y}{\rho L T} \bar{V}_k^2 C_{x_a} + \bar{m} \bar{\omega} \sin \beta - x(1 - \psi')^2 \frac{F_p}{L T} \sigma_e (1 - t') k_v^2;$$

$$\bar{C}_y = C_{y_r} + \frac{\rho_1 S_y}{\rho L T} \bar{V}_k^2 C_{y_a} + \bar{m} \bar{\omega} \cos \beta - x(1 - \psi')^2 \frac{F_p}{L T} k_p \sigma_y k_v^2;$$

$$\bar{C}_m = C_{m_r} + \frac{\rho_1 S_y}{\rho L T} \bar{V}_k^2 C_{m_a} - x(1 - \psi')^2 \frac{F_p}{L T} k_p k_\varepsilon \bar{l}_p \sigma_y k_v^2;$$

$$\bar{M} = m_{22} m_{66} - m_{26}^2.$$

Аналитическое решение система (1.2) не имеет, так как входящие в нее величины  $C_{x_r}, C_{y_r}, C_{m_r}, C_{x_a}, C_{y_a}, C_{m_a}, \sigma_e, \sigma_y, t', \psi'$  являются сложными нелинейными функциями искомых параметров  $\bar{V}, \bar{\omega}, \beta$ .

$X_G, Y_G, M_G$  — экспериментально определенные гидродинамические силы;

$C_{x_r}, C_{y_r}, C_{m_r}$  — безразмерные коэффициенты экспериментально определенных гидродинамических сил;

$C_{x_a}, C_{y_a}, C_{m_a}$  — безразмерные коэффициенты аэродинамических сил;

$\sigma_y$  — безразмерный коэффициент по поперечной силе;

$\sigma_e$  — безразмерный коэффициент по тяге;

$\psi'$  — коэффициент попутного потока;

$\bar{l}_p = \frac{l_p}{L}$  — относительное отстояние движительно-рулевого комплекса

судна от миделя ( $\bar{l}_p < 0$  в корму);

$F_p$  — гидравлическое сечение движительно-рулевого комплекса;

$t'$  — коэффициент засасывания при криволинейном движении;

$k_v$  — коэффициент, характеризующий изменение скорости в районе комплекса по сравнению со скоростью в центре тяжести судна;

$k_\varepsilon$  — коэффициент смещения точки приложения поперечной силы движительно-рулевого комплекса при его работе за корпусом;

$\bar{m}, \bar{m}_{11}, \bar{m}_{22}, \bar{m}_{26}, \bar{m}_{66}$  — безразмерные коэффициенты масс и моментов;

$\rho$  – массовая плотность жидкости (воды);

$\rho_1$  – плотность воздуха;

$S_y$  – площадь парусности (боковой проекции надводной части);

$V_k$  – скорость ветра относительно судна.

В [78] при составлении уравнений движения судна использованы следующие допущения:

- движение судна происходит в плоскости ватерлинии.
- наклона судна в плоскости шпангоута малы и они не влияют на инерционные и гидродинамические характеристики.
- изменение скорости в процессе маневра не влияет на деферент и среднюю осадку судна.
- при неустановившемся движении все гидродинамические характеристики судна в каждый момент времени совпадают с гидродинамическими характеристиками, совершающего установившееся движение с теми же элементами (скоростью, угловой скоростью и углом дрейфа), т. е. используется гипотеза стационарности.

С учетом этих допущений уравнения движения судна в безразмерном виде имеет вид (1.3):

$$\begin{aligned} \frac{d\beta}{d\tau} + q_{21}\beta + r_{21}\bar{\omega} + s_{21}\delta + h_1\beta|\beta| &= 0 \\ \frac{d\bar{\omega}}{dt} + q_{31}\beta + r_{31}\bar{\omega} + s_{31}\delta &= 0 \end{aligned} \quad (1.3)$$

$q_{21}, r_{21}, s_{21}, q_{31}, r_{31}, s_{31}, h_1$  – коэффициенты, характеризующие динамику судна и изменяющиеся при изменении внешних факторов.

Входящая в уравнения величина  $\bar{\omega}$  равна (1.4):

$$\bar{\omega} = \omega \frac{L}{V_0} = \left( \frac{V}{R} + \frac{d\beta}{dt} \right) \frac{L}{V_0} = \Omega + \frac{d\beta}{d\tau} \quad (1.4)$$

где  $\bar{\omega}$  – безразмерная угловая скорость;

$R$  – радиус кривизны траектории;

$V_0$  – начальная скорость судна,

которая при установившемся движении преобразуется в относительную кривизну, а при неустановившемся – отличается от нее на малую величину

$$\frac{d\beta}{d\tau}.$$

Система уравнений (1.3) нелинейна, так как содержит произведение неизвестных в виде  $\beta|\beta|$ .

При выводе уравнений в [20] приняты следующие допущения:

- движение происходит в горизонтальной плоскости. Пренебрегают влиянием крена и дифферента на гидродинамические моменты и силы, действующие на судно;

- силы инерционной природы, действующие на корпус судна, определяют в предположении, что судно симметрично относительно диаметральной плоскости и его центр тяжести располагается в плоскости мидель-шпангоута. Пренебрегает инерционными силами, действующими на движительно-рулевой комплекс;

- при определении моментов и сил неинерционной природы, действующих на корпус судна, пользуется гипотезой стационарности.

Уравнения движения судна, принимая, что скорость хода его постоянна, т. е.  $\bar{V} = const$ , будет иметь вид:

$$\begin{aligned} \frac{d\beta}{d\tau} + r_{22} \frac{d\bar{\omega}}{d\tau} + q_{21}\beta + r_{21}\bar{\omega} + h_1\beta|\beta| + h_2\beta|\bar{\omega}| + s_{21}\delta = 0 \\ \frac{d\bar{\omega}}{d\tau} + q_{32} \frac{d\beta}{d\tau} + q_{31}\beta + r_{31}\bar{\omega} + s_{31}\delta = 0 \end{aligned} \quad (1.5)$$

где  $r_{22}$ ,  $q_{21}$ ,  $r_{21}$ ,  $h_1$ ,  $h_2$ ,  $s_{21}$ ,  $q_{32}$ ,  $q_{31}$ ,  $r_{31}$ ,  $s_{31}$  – коэффициенты, характеризующие динамику судна и изменяющиеся при изменении внешних факторов.



## 1.2 Математические модели движения судна на мелководье

Вопросы влияния мелководья на гидродинамические силы и моменты рассматриваются в работах В. Г. Павленко, А. Д. Гофмана, А. М. Басина, В. И. Когана и др. [20, 21, 45, 53].

В работе [45] система уравнений для описания влияния мелководья на гидродинамику судна имеет вид (1.6):

$$\begin{aligned}
 -C_{x_r} - \bar{m} \bar{\omega} \sin \beta + x(1 - \psi') \frac{F_p}{LT} q_e \sigma_e (1 - t) k_v^2 &= 0 \\
 C_{y_r} - \bar{m} \bar{\omega} \cos \beta - x(1 - \psi')^2 \frac{F_p}{LT} k_p \sigma_y k_v^2 &= 0 \\
 C_{m_r} - x(1 - \psi')^2 \frac{F_p}{LT} k_p k_\varepsilon \bar{l}_p \sigma_y k_v^2 &= 0 \\
 -K_\varepsilon \bar{l}_p C_{y_r} + C_{m_r} + k_\varepsilon \bar{l}_p \bar{m} \bar{\omega} \cos \beta &= 0
 \end{aligned} \tag{1.6}$$

Наличие мелководья искажает динамические характеристики судна. На мелководье возрастает сопротивление движению судна, и при одной и той же частоте вращения винтов скорость судна, по сравнению с его скоростью на глубокой воде, уменьшается. При этом возрастает коэффициент нагрузки движителя.

Изменение гидромеханических характеристик корпуса на мелководье приводит к изменению характеристик его криволинейного движения. Наиболее наглядно эти изменения прослеживаются на установившейся циркуляции.

Для изучения влияния мелководья на характеристики управляемости судна в [21] использовали следующую систему уравнений:

$$\begin{aligned}
 -m_{11} \frac{d\bar{V}}{d\tau} \cos \beta + \bar{m}_{11} V \frac{d\beta}{d\tau} \sin \beta - V^2 (C_x - m \bar{\omega} \sin \beta) + \varphi_\rho n_p (1 - t) V^2 &= 0 \\
 m_{22} \frac{dV}{d\tau} \sin \beta + \bar{m}_{22} \bar{V} \frac{d\beta}{d\tau} \cos \beta - \bar{m}_{26} \frac{d\bar{\omega}}{d\tau} + \bar{V}^2 (C_y - m \bar{\omega} \cos \beta) - \varphi_\rho C_y \bar{V}^2 &= 0 \\
 \bar{m}_{66} \frac{d\bar{\omega}}{d\tau} + m_{26} \left( \frac{dV}{d\tau} \sin \beta + \bar{V} \frac{d\beta}{d\tau} \cos \beta \right) + V^2 C_m - \varphi_\rho \bar{l}_p \sigma_y \bar{V}^2 &= 0
 \end{aligned} \tag{1.7}$$

Влияние мелководья, описываемое этими уравнениями, проявляется в том, что коэффициенты гидродинамических сил  $C_x$ ,  $C_y$ ,  $C_m$ ,  $\bar{m}_{ik}$  зависят от отношения глубины к осадке судна  $\frac{H}{T}$ .

В работе [7] получена система дифференциальных уравнений (1.8), учитывающих влияние мелководья на гидродинамику судна в следующем виде:

$$\left. \begin{aligned}
 \frac{d\beta}{dt} &= \frac{A'_1}{A'_2 + \left(\frac{T}{H}\right)^2} \frac{V_0^2}{V} \delta - \frac{A'_3 \left(\frac{T}{H}\right)^2 + A'_4 V_0^2}{A'_2 + \left(\frac{T}{H}\right)^2} \frac{V_0^2}{V} \beta - \frac{A'_5 \left(\frac{T}{H}\right)^2 + A'_6 V_0^2}{A'_2 + \left(\frac{T}{H}\right)^2} \frac{V_0^2}{V} \Omega + \\
 &+ \frac{A'_7 \left(\frac{T}{H}\right)^2 + A'_8 V_0^2}{A'_2 + \left(\frac{T}{H}\right)^2} \frac{V_0^2}{V} \Omega - A'_9 \frac{V_0}{V} \beta \Omega \text{sign}(\beta, \Omega) \\
 \frac{d\Omega}{dt} &= \frac{B'_1}{B'_2 + \left(\frac{T}{H}\right)^2} V_0^2 \delta + \frac{B'_3 \left(\frac{T}{H}\right)^2 + B'_4}{B'_2 + \left(\frac{T}{H}\right)^2} V_0^2 \beta + \frac{B'_5 \left(\frac{T}{H}\right)^2 + B'_6}{B'_2 + \left(\frac{T}{H}\right)^2} V_0^2 \Omega + B'_7 V_0 \beta \Omega \text{sign}(\beta, \Omega) \\
 \frac{dV}{dt} &= \frac{C'_1 V_0^4 V^{-2}}{C'_2 + \left(\frac{T}{H}\right)^2} - \frac{C'_1 V_0 V}{C'_2 + \left(\frac{T}{H}\right)^2} - \frac{C'_3 V_0 V \delta^2}{C'_2 + \left(\frac{T}{H}\right)^2} - \left( \frac{A'_2 + \left(\frac{T}{H}\right)^2}{C'_2 + \left(\frac{T}{H}\right)^2} - \frac{A'_7 \left(\frac{T}{H}\right)^2 + A'_8}{A'_2 + \left(\frac{T}{H}\right)^2} \right) \\
 V\beta\beta &+ \frac{A'_1 V_0^2 \beta \delta}{A'_2 + \left(\frac{T}{H}\right)^2} - \frac{A'_3 \left(\frac{T}{H}\right)^2 + A'_4}{A'_2 + \left(\frac{T}{H}\right)^2} V_0^2 \beta^2 - \frac{A'_5 \left(\frac{T}{H}\right)^2 + A'_6}{A'_2 + \left(\frac{T}{H}\right)^2} V_0^2 \Omega \beta - \\
 &- A'_9 \beta^2 \Omega V_0 \text{sign}(\beta, \Omega)
 \end{aligned} \right\} (1.8)$$

где  $A_i$ ,  $B_i$ ,  $C_i$  – новые гидродинамические коэффициенты, зависящие от  $T$  и  $H$ .

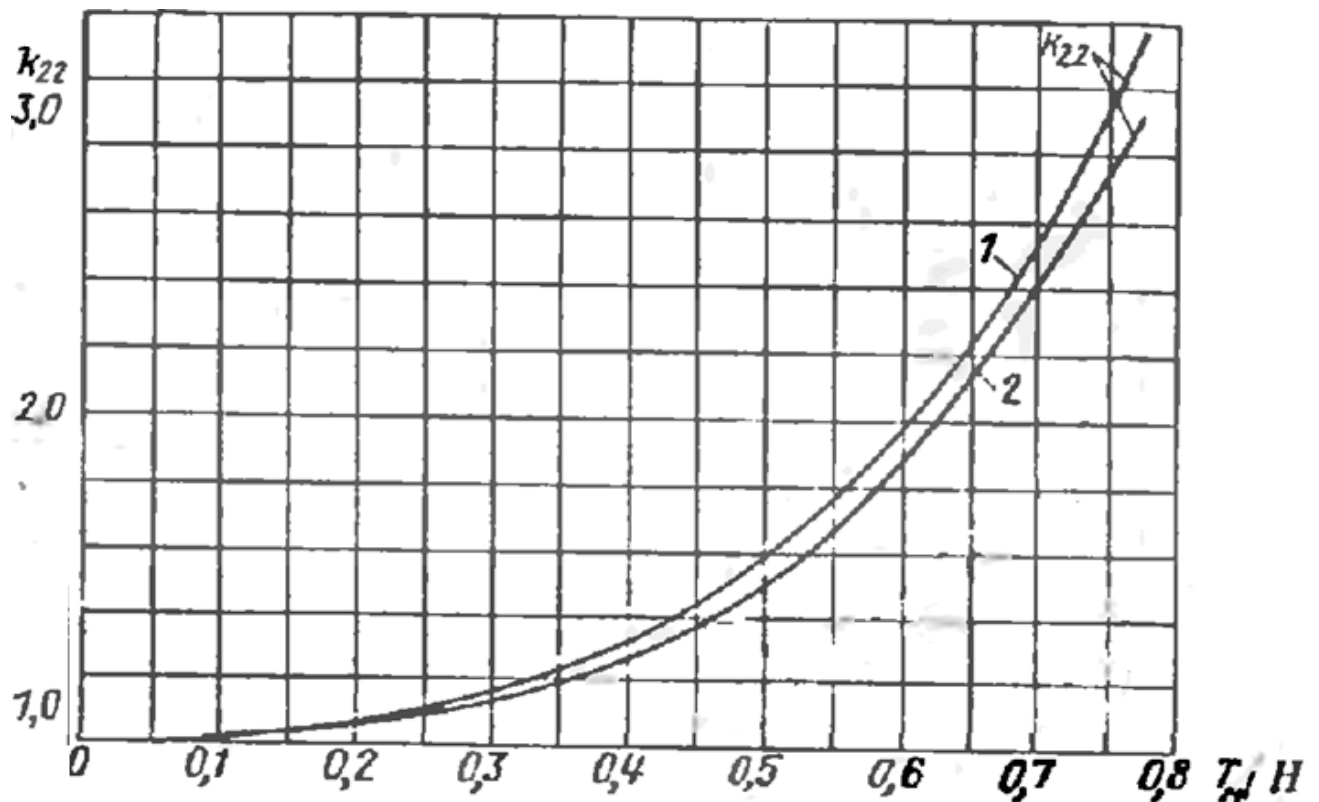
Недостатком этой модели является относительно низкая точность аппроксимации коэффициентов влияния мелководья на присоединенные массы и инерции, влияния мелководья на вращательные и позиционные производные. Кроме того, в этой модели не осуществлена аппроксимация коэффициентов нелинейности поперечной силы и момента.

### 1.3 Влияние мелководья на гидродинамические характеристики корпуса судна

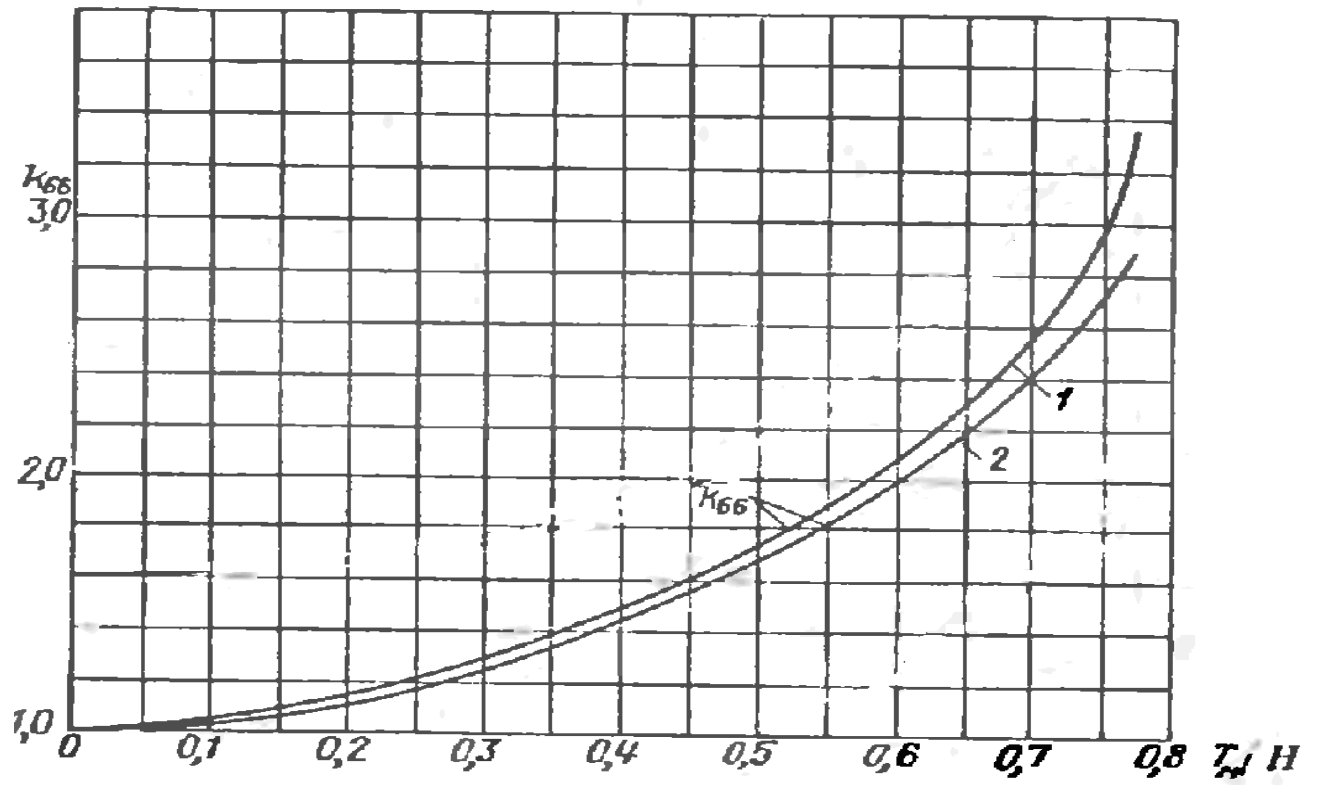
В работе [7] выведены эмпирические зависимости вращательных и позиционных производных  $k_1, k_2, k_3, k_4$  от отношения  $T/H$  для значения  $\lambda = 0,093$ , а также для коэффициента влияния мелководья на кинематические характеристики судна, определяемые в функции  $T/H$   $k_{11}, k_{22}, k_{66}$  аппроксимированные уравнением 2-го порядка.

Проведенный анализ показал, что при такой аппроксимации максимальная средняя квадратическая погрешность аппроксимации составляет 5,8 %, что неприемлемо.

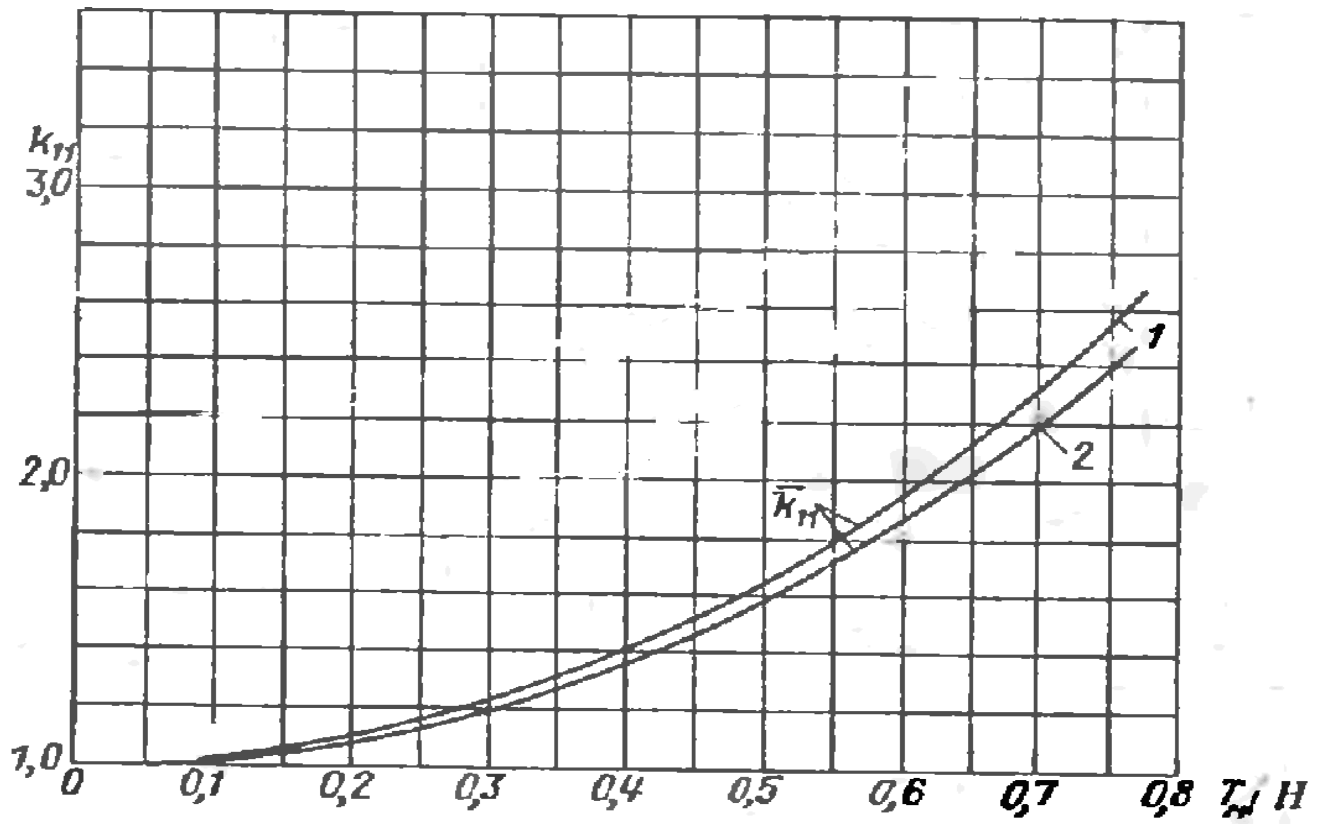
На рисунке 1.2 приведена зависимость влияния мелководья на присоединенные массы и инерции для эллипсоида и корпуса судна, а на рисунках 1.3 и 1.4 приведены зависимости влияния мелководья на вращательные производные: а) для момента рыскания, б) для поперечной силы. I - для  $\lambda = 0,2$ . II - для  $\lambda = 1$ , взятые из [45].



а)

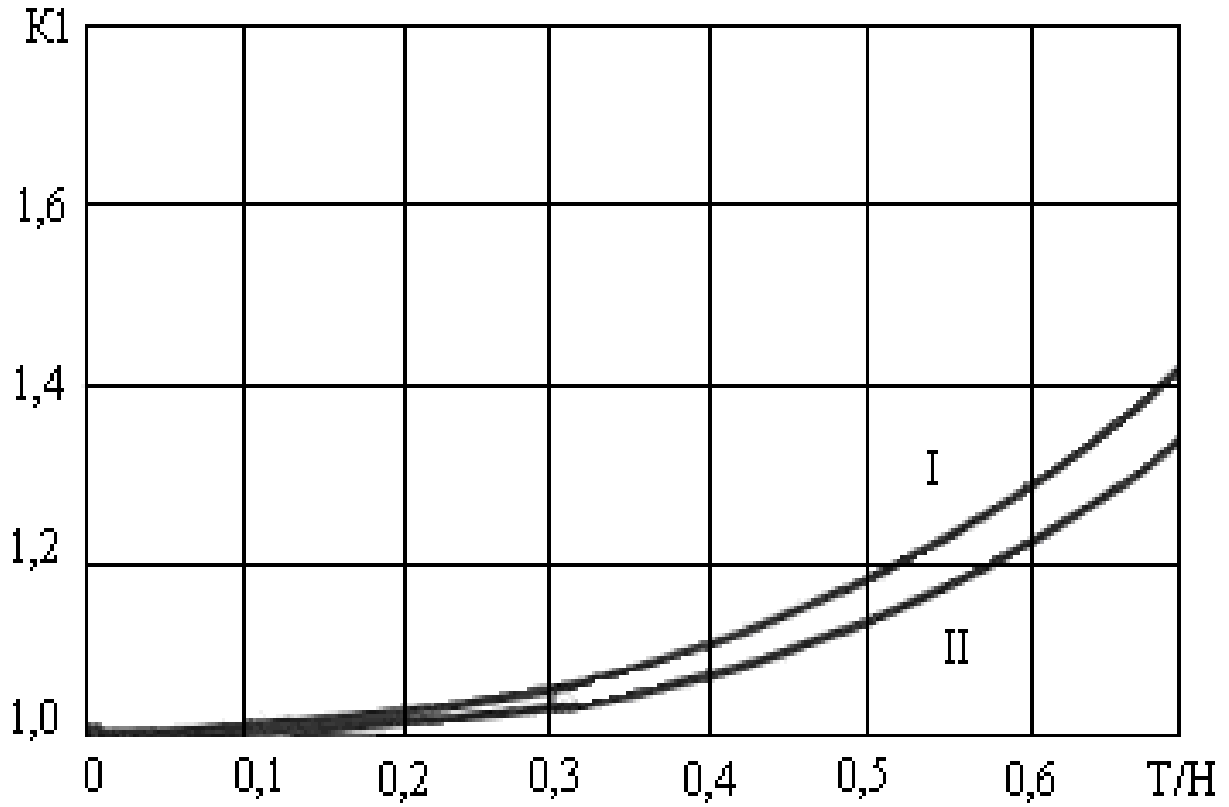


б)

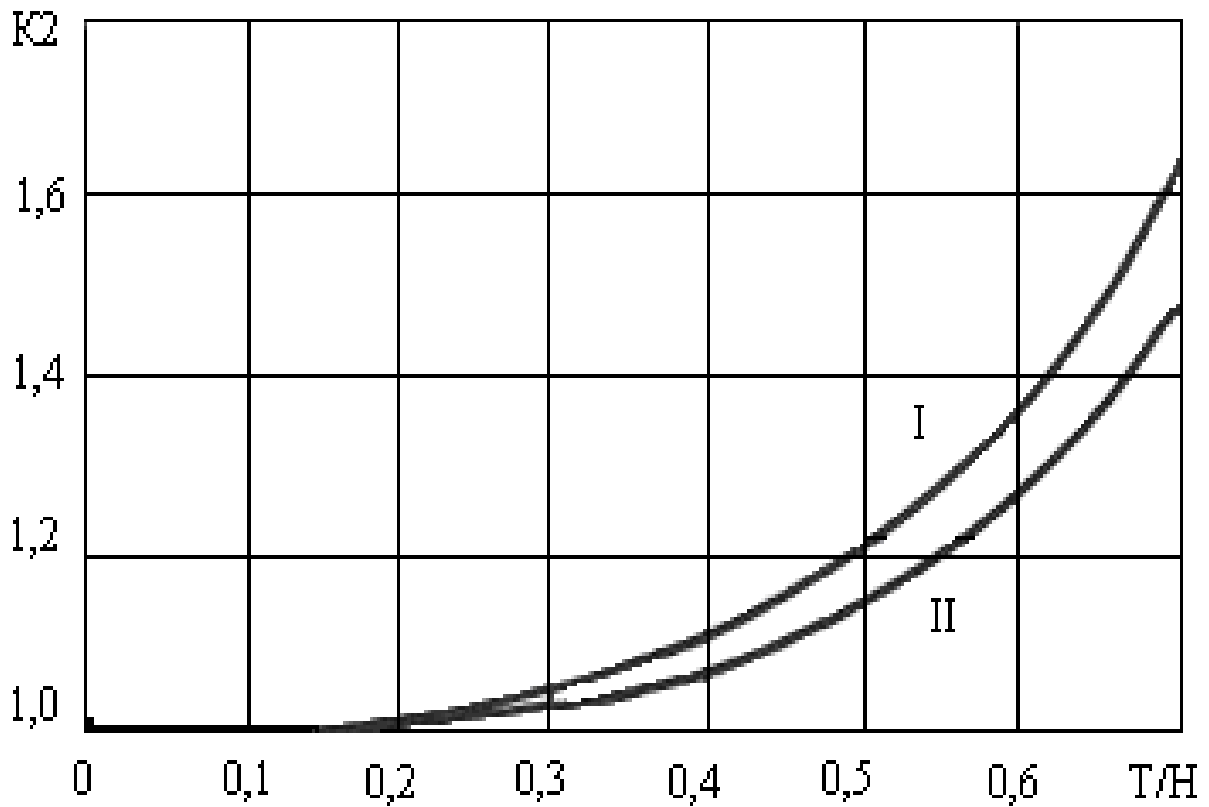


в)

Рисунок 1.2 - Влияние мелководья на присоединенные массы и инерции:  
I-эллипсоида, II-корпуса судна



а)



б)

Рисунок 1.3 - Влияние мелководья на вращательные производные: а) для момента рыскания, б) для поперечной силы.

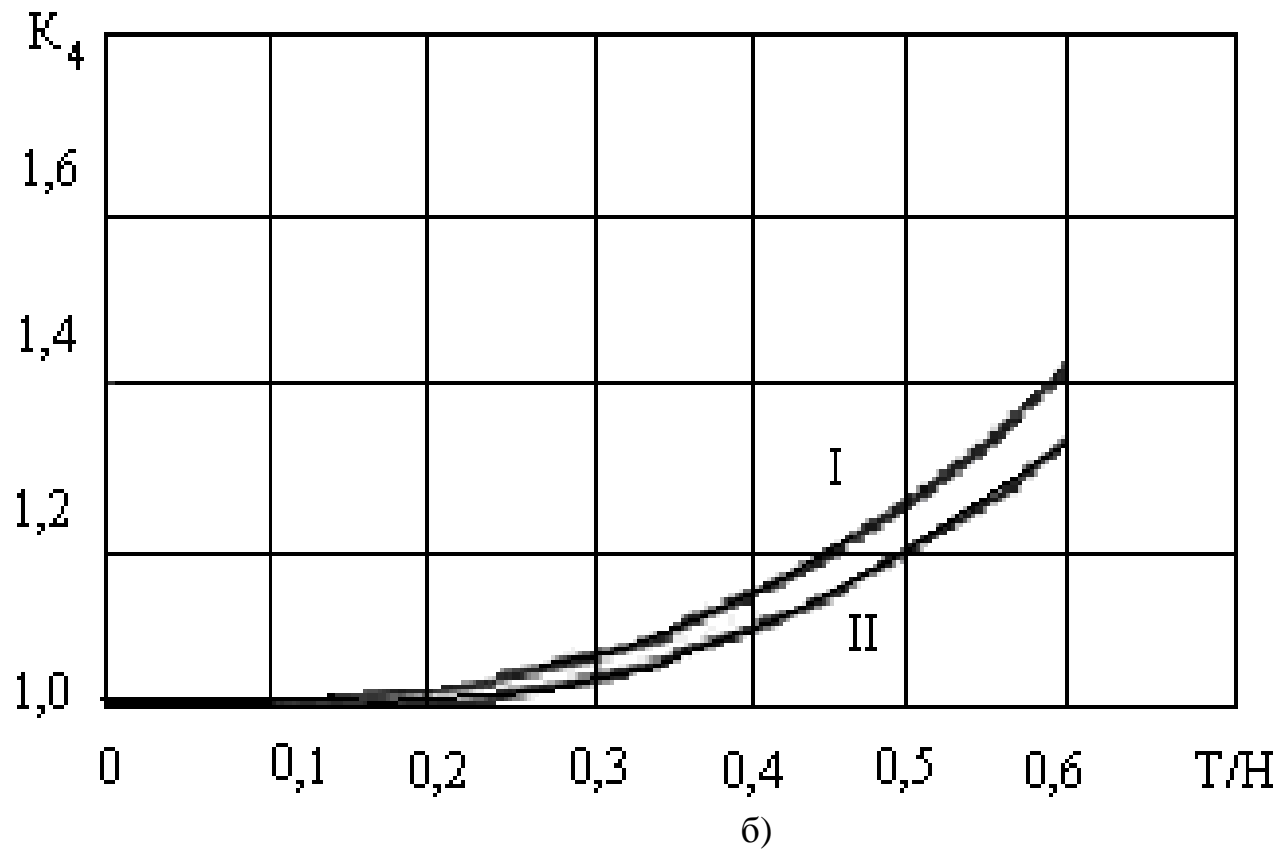
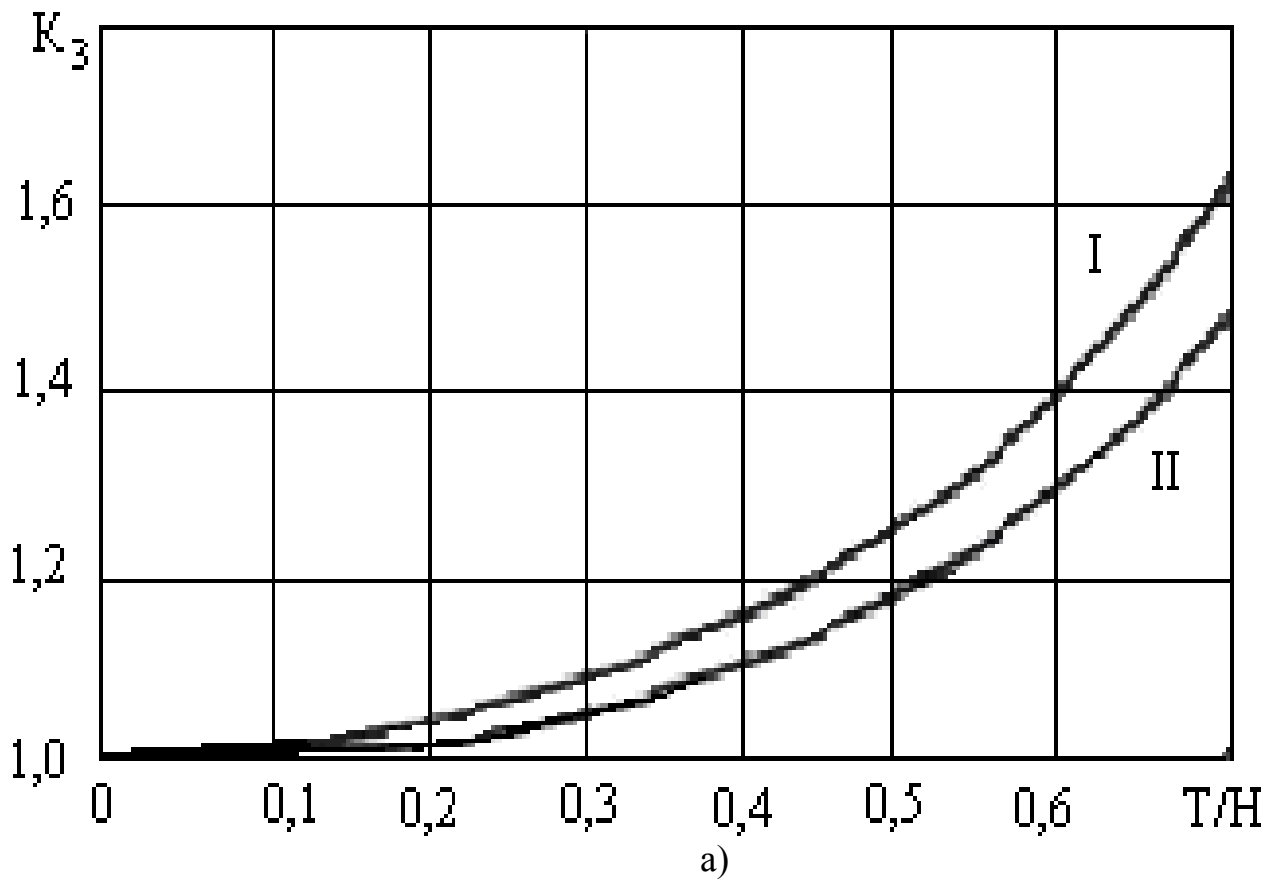


Рисунок 1.4 - Влияние мелководья на позиционные производные (I - для  $\lambda = 0,2$ . II - для  $\lambda = 1$ ): а) для момента рыскания, б) для поперечной силы.

## 1.4 Влияние мелководья на коэффициенты нелинейности поперечной силы и момента

Одним из основных факторов, определяющим управляемость судна, являются гидродинамические силы и моменты, действующие на корпус в плоскости его ватерлинии при движении судна по криволинейной траектории.

Важнейшим частным случаем криволинейного движения судна является движение на установившейся циркуляции.

Согласно [45] структурные выражения для коэффициента поперечной силы и момента, общее для глубокой воды и мелководья, имеет вид (1.9):

$$n = a_1\beta + a_2\beta|\beta| + a_3\omega + a_4\beta|\omega| \quad (1.9)$$

$$m = b_1\beta - b_2\omega \quad (\text{при } \beta < 20^\circ; \omega < 0,65).$$

Коэффициенты  $a_1, \dots, a_4; b_1, b_2$ , входящие в выражения являются функциями основных параметров корпуса судна.

$$a_{1H} = a_1\bar{a}_1; \text{ где } a_1 = \pi \frac{T}{L}$$

Коэффициент влияния мелководья  $\bar{a}_1$  определяется по графику [45], приведенному на рисунке 1.5 в функции  $\frac{T}{L}$  и  $\frac{T}{H}$ .

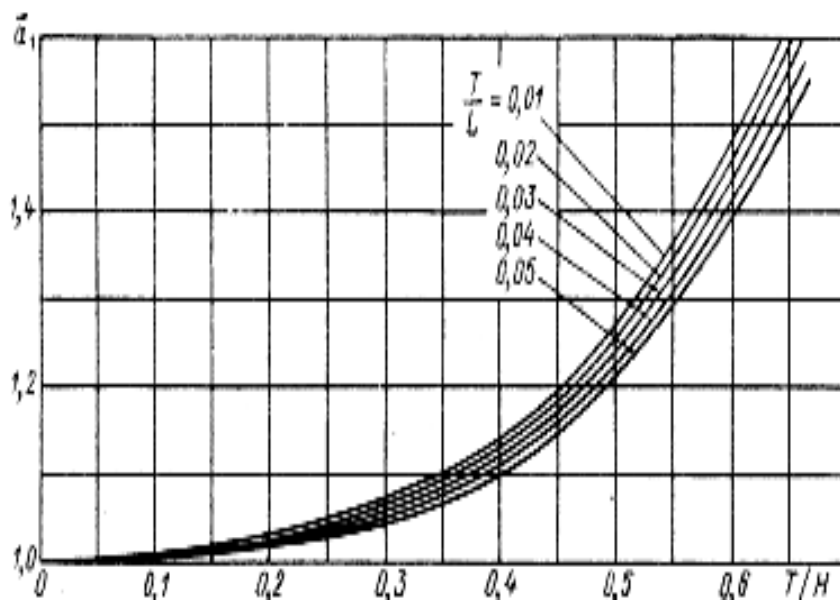


Рисунок 1.5 - График изменения коэффициента  $\bar{a}_1$

$$a_{2H} = a_2 \bar{a}_2,$$

$$\text{где } a_2 = 1,055 + 0,251 \frac{B}{T} + 0,021 \left( \frac{B}{T} \right)^2 + 12,49 \frac{T}{L} + 0,0244 \frac{L}{T} - a'_k,$$

$$\bar{a}_2 = 3,34 \left( \frac{T}{H} \right)^2 - 0,234 \left( \frac{T}{H} \right) + 1$$

$a'_k = 1,096$  - для судов с ложкообразными обводами,

$a'_k = 0,949$  - для судов с санообразными обводами.

$$\bar{a}_2 = 1,0 - 0,234 \frac{T}{H} + 3,34 \left( \frac{T}{H} \right)^2.$$

$a_3$  для глубокой и мелкой воды определяется по графику [45], приведенному на рисунке 1.6 в функции  $\bar{s}_k = 2S_k/LT$  и  $T/H$

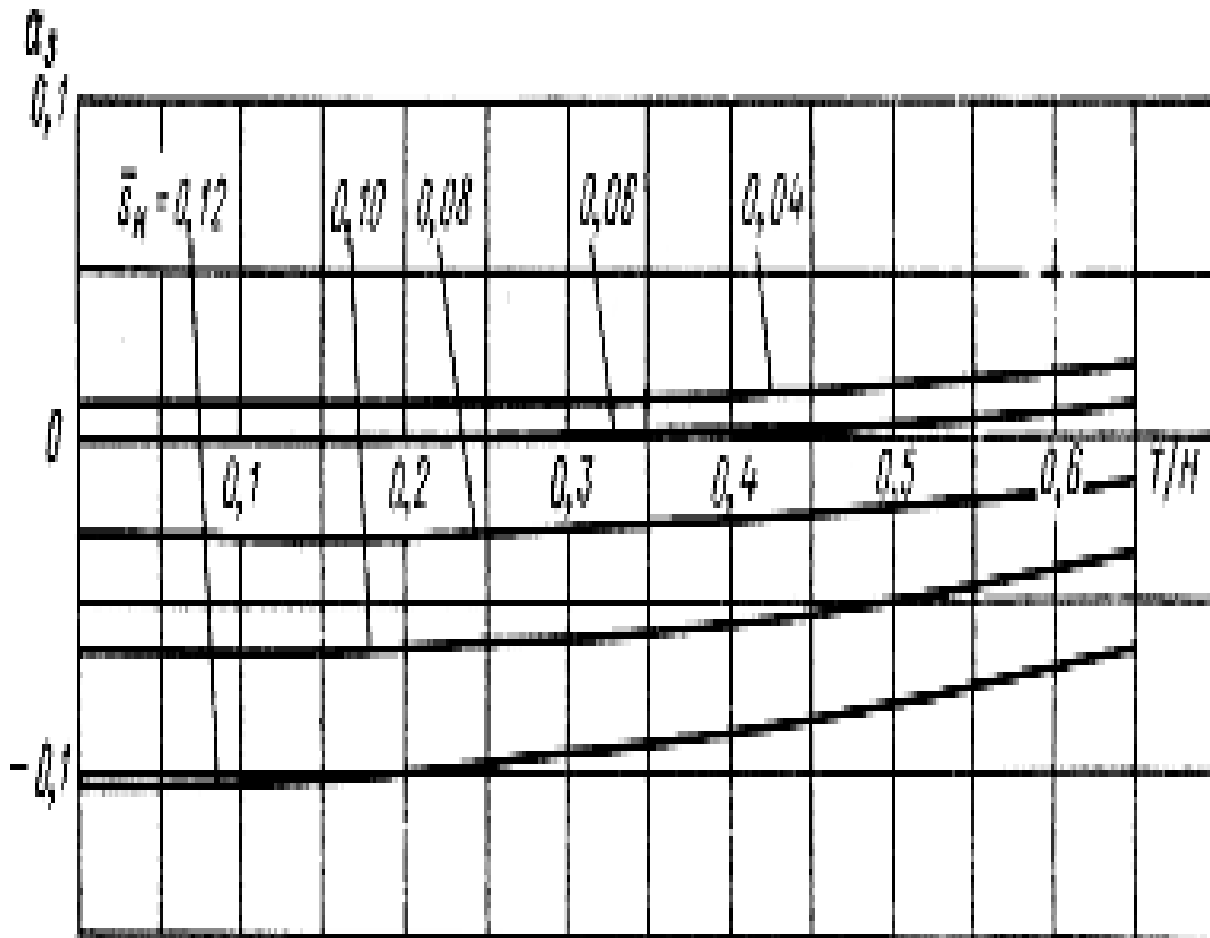


Рисунок 1.6 - График изменения коэффициента  $a_3$



$a_4$  для глубокой и мелкой воды определяется по графику [45], приведенному на рисунке 1.7 в функции  $\bar{s}_k = 2S_k/LT$  и  $T/H$

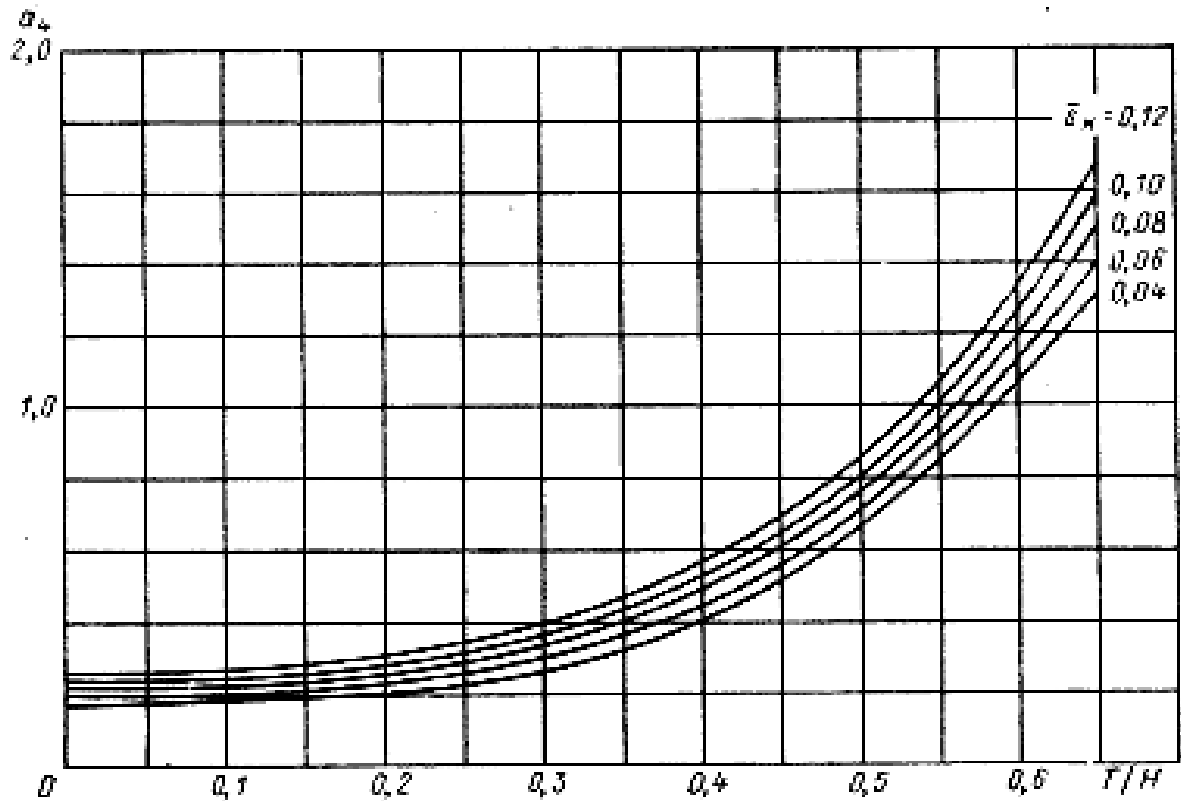


Рисунок 1.7 - График изменения коэффициента  $a_4$

$$b_{1H} = b_1 \bar{b}_1; \quad b_1 = 3,36 \left( \frac{T}{L} + 0,0136 \right) \times (\bar{s}_k + a_k)$$

$a_k = 0,58$  - для судов с ложкаобразными обводами,

$a_k = 0,48$  - для судов с санообразными обводами.

Коэффициент  $\bar{b}_1$  определяется по графику, приведенному на рисунке 1.8 [45] в функции  $\bar{s}_k = 2S_k/LT$  и  $T/H$ .

$$b_{2H} = b_2 \bar{b}_2;$$

$$b_2 = 0,1595 - 0,0469 \frac{B}{T} + 0,00375 \left( \frac{B}{T} \right)^2$$

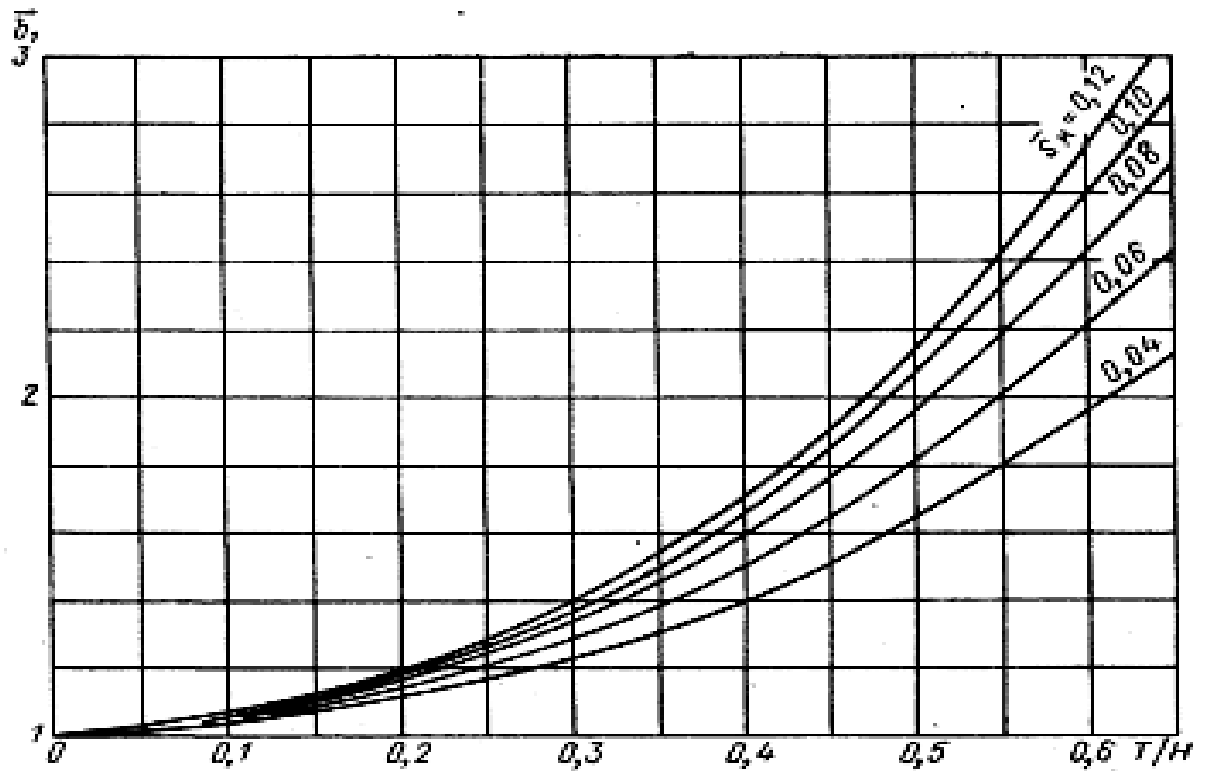


Рисунок 1.8 - График изменения коэффициента  $\bar{b}_1$

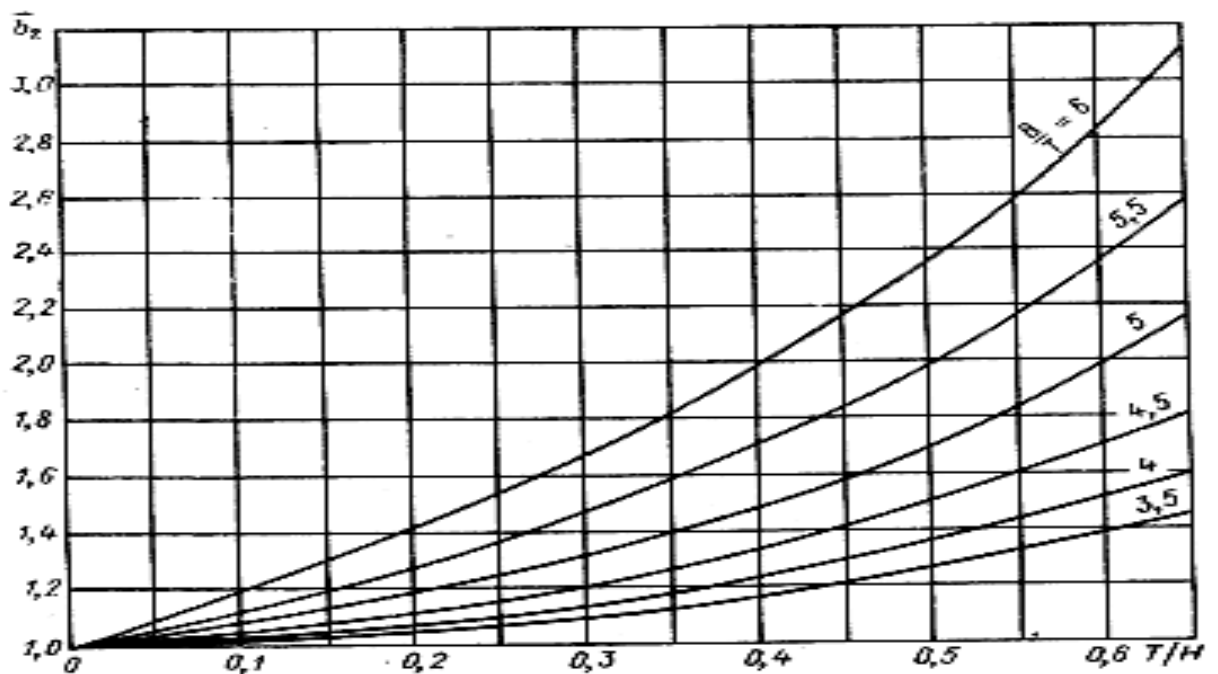


Рисунок 1.9 - График изменения коэффициента  $\bar{b}_2$

Коэффициент влияния мелководья  $\bar{b}_2$  определяется по графику [45], приведенному на рисунке 1.9 в функции  $V/T$  и  $T/H$ .

Как видим, численные значения этих коэффициентов можно получить с помощью интерполяционных кривых, часть которых аппроксимирована соответствующими формулами, а часть представлена непосредственно в виде графиков.

Ставится задача аппроксимировать те интерполяционные кривые, представленные в виде графиков влияния мелководья на соответствующие коэффициенты.

### **Выводы по первой главе**

1. Проведен анализ математических моделей движения судна на глубокой воде.
2. Проведен анализ математических моделей движения судна на мелководье.
3. В качестве модели движения судна на глубокой воде взята система безразмерных дифференциальных уравнений для малых отклонений судна от криволинейной траектории в связанной с судном системе координат согласно [86]
4. Проведен анализ влияния мелководья на гидродинамические характеристики корпуса судна.
5. Проведен анализ влияния мелководья на коэффициенты нелинейности поперечной силы и момента.

## ГЛАВА 2. АНАЛИТИЧЕСКОЕ ОПИСАНИЕ КРИВЫХ ВЛИЯНИЯ МЕЛКОВОДЬЯ НА ГИДРОДИНАМИЧЕСКИЕ КОЭФФИЦИЕНТЫ

### 2.1 Аналитическое описание влияния мелководья на гидродинамические характеристики корпуса судна

Влияние мелководья на позиционные  $\bar{n}_y^\beta, \bar{m}_y^\beta$  и вращательные производные  $\bar{n}_y^\omega, \bar{m}_y^\omega$  поперечной силы и момента рыскания зависят от размерений судна и глубины мелководья, в котором происходит плавание

$$\lambda = \frac{T}{H}.$$

В работе [7] выведены эмпирические зависимости вращательных и позиционных производных  $k_1, k_2, k_3, k_4$  от отношения  $T/H$  для значения  $\lambda = 0,093$ , а также для коэффициентов влияния мелководья на кинематические характеристики судна  $k_{11}, k_{22}, k_{66}$ , аппроксимированные уравнением 2-го порядка.

Проведенный анализ показал, что при такой аппроксимации максимальная средняя квадратическая погрешность аппроксимации составляет 5,8 %, что неприемлемо.

Ставилась задача повышения точности аппроксимации кривых с точностью не хуже 2 %.

Для аппроксимации кривых влияния мелководья на гидродинамические коэффициенты использован пакет программ MatLab.

На рисунках 2.1- 2.3 приведены экранные формы аппроксимации кривых влияния мелководья на кинематические характеристики судна, в частности, для  $k_{11}, k_{22}, k_{66}$  для значения  $\lambda = 0,093$  (судно №2), определяемые в функции отношения осадки судна к глубине мелководья, аппроксимированные уравнением 3-го порядка [4].

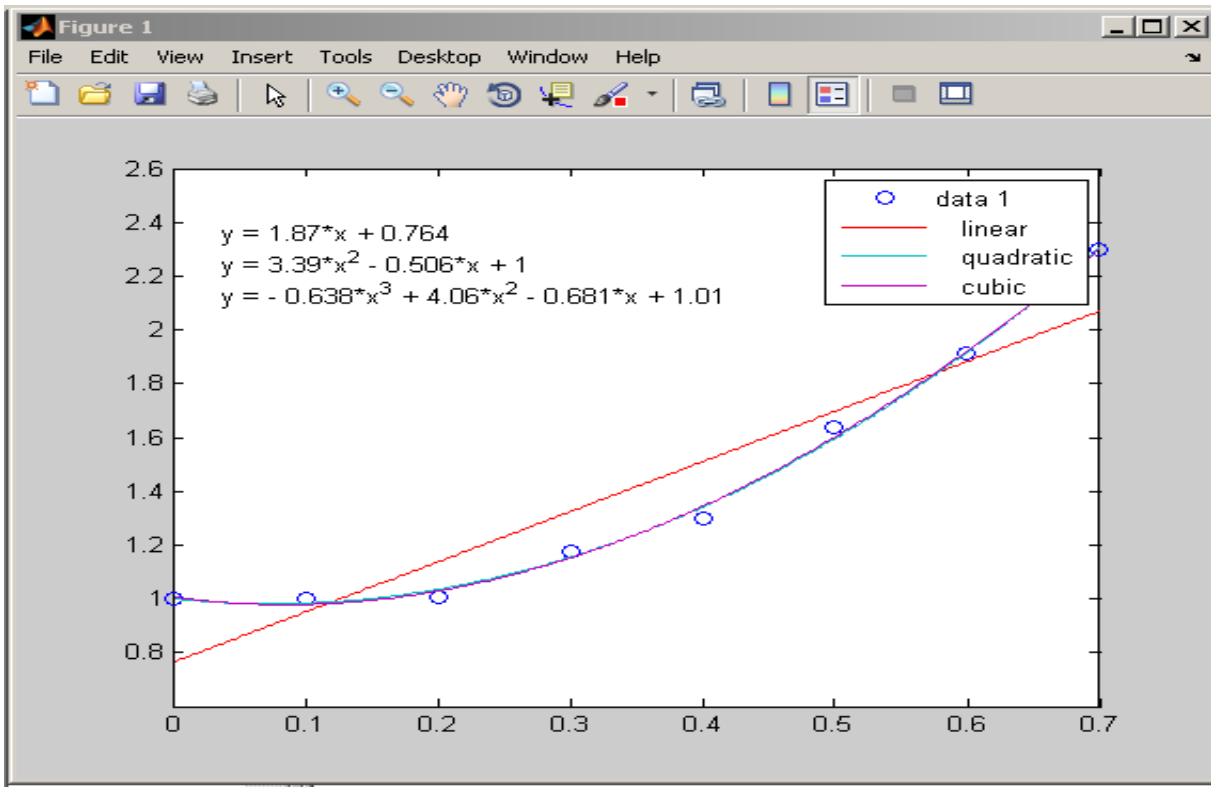


Рисунок 2.1 - Экранная форма аппроксимации кривой влияния мелководья на коэффициент  $k_{11}$  для значения  $\lambda = 0,093$

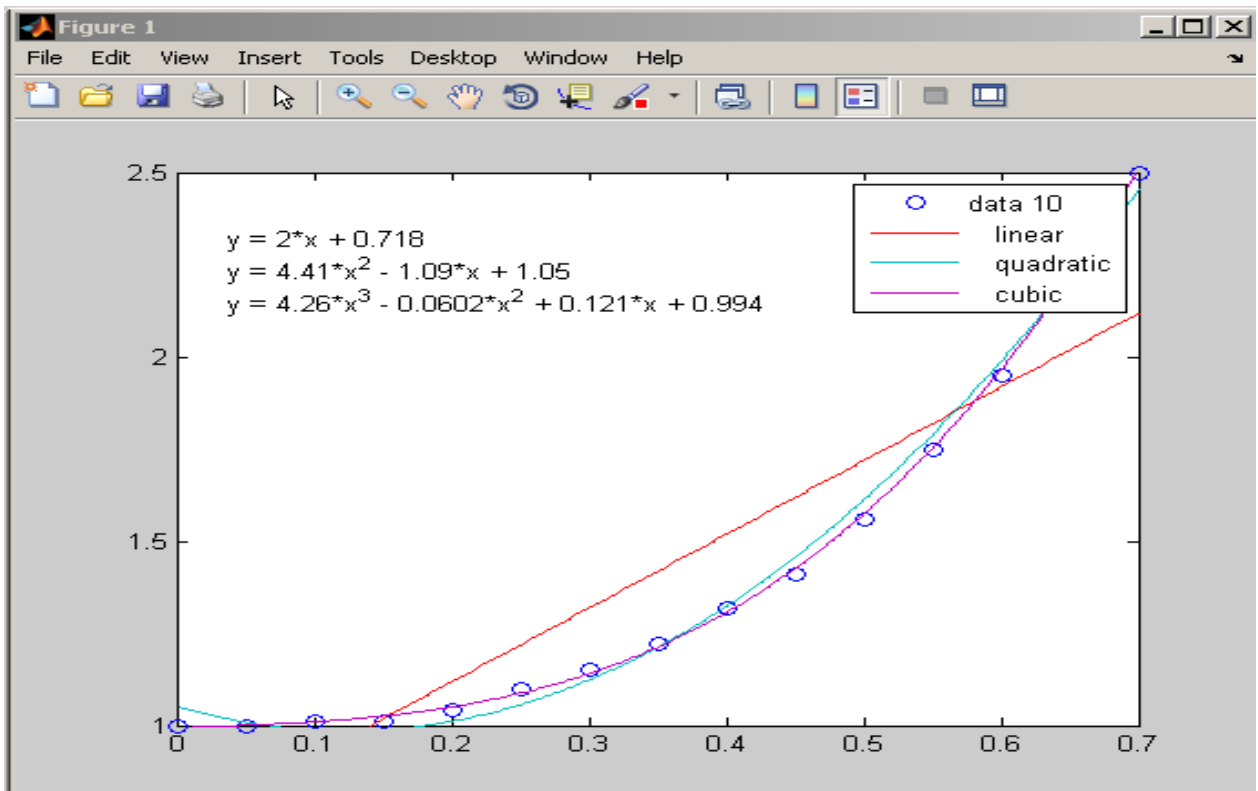


Рисунок 2.2 - Экранная форма аппроксимации кривой влияния мелководья на коэффициент  $k_{22}$  для значения  $\lambda = 0,093$ .

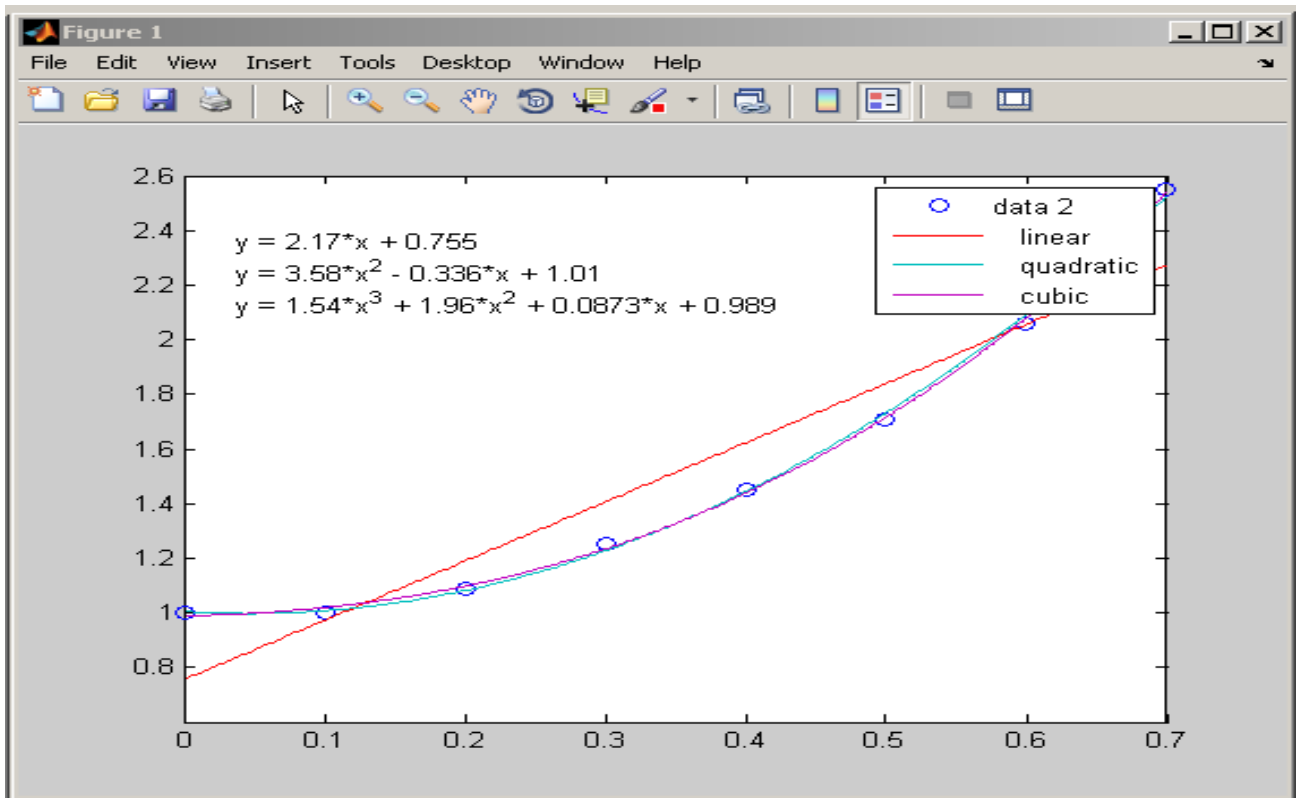


Рисунок 2.3 - Экранная форма аппроксимации кривой влияния мелководья на коэффициент  $k_{66}$  для значения  $\lambda = 0,093$ .

На рисунках 2.4-2.7 приведены экранные формы аппроксимации кривой влияния мелководья на вращательные и позиционные производные судна, в частности для  $k_1, k_2, k_3, k_4$ , определяемые в функции отношения осадки судна к глубине акватории, аппроксимированные уравнением 3-го порядка.

Как известно, с повышением порядка уравнения аппроксимируемой кривой увеличивается точность аппроксимации.

Была проведена аппроксимация кривых влияния мелководья на коэффициенты при уравнениях движения судна уравнением 3-го порядка.

$$k_i = A \left( \frac{T}{H} \right)^3 + B \left( \frac{T}{H} \right)^2 + C \left( \frac{T}{H} \right) + D \quad (2.1)$$

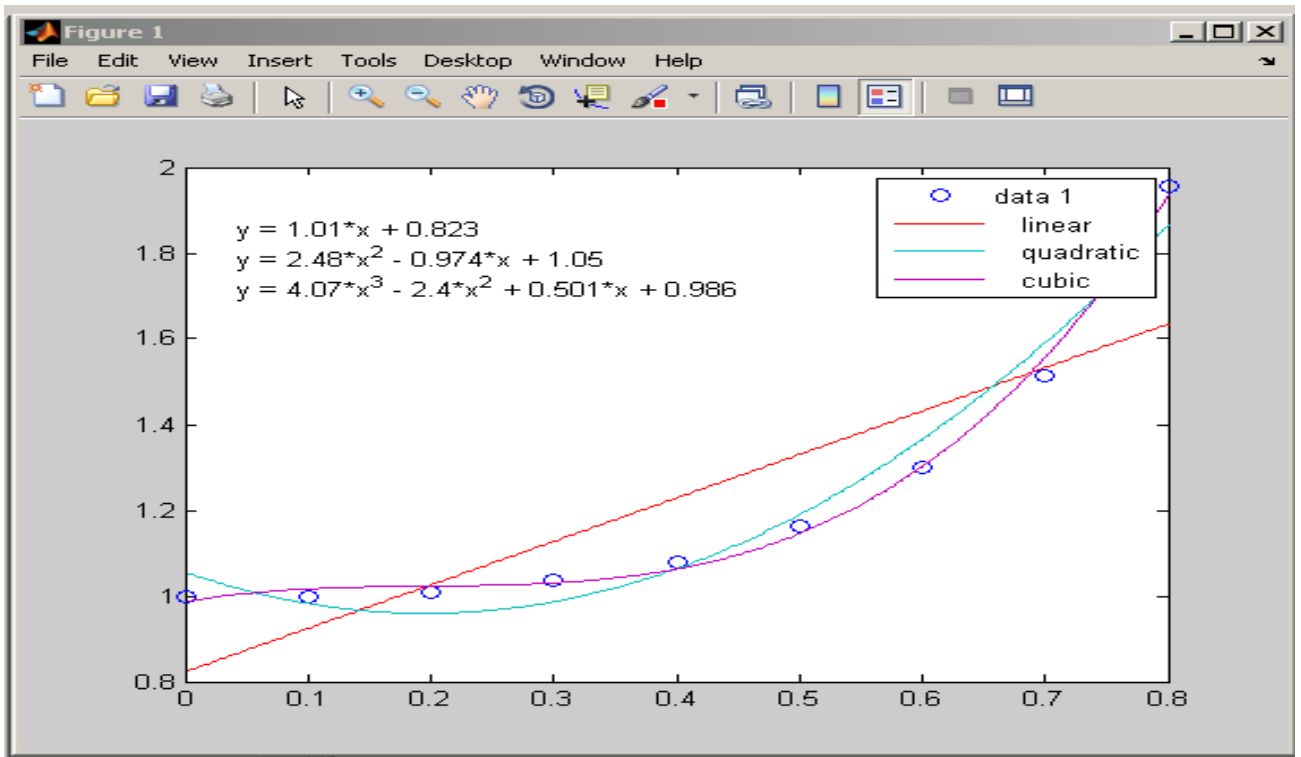


Рисунок 2.4 - Экранная форма аппроксимации кривой влияния мелководья на вращательные производные коэффициента момента рыскания судна  $k_1$

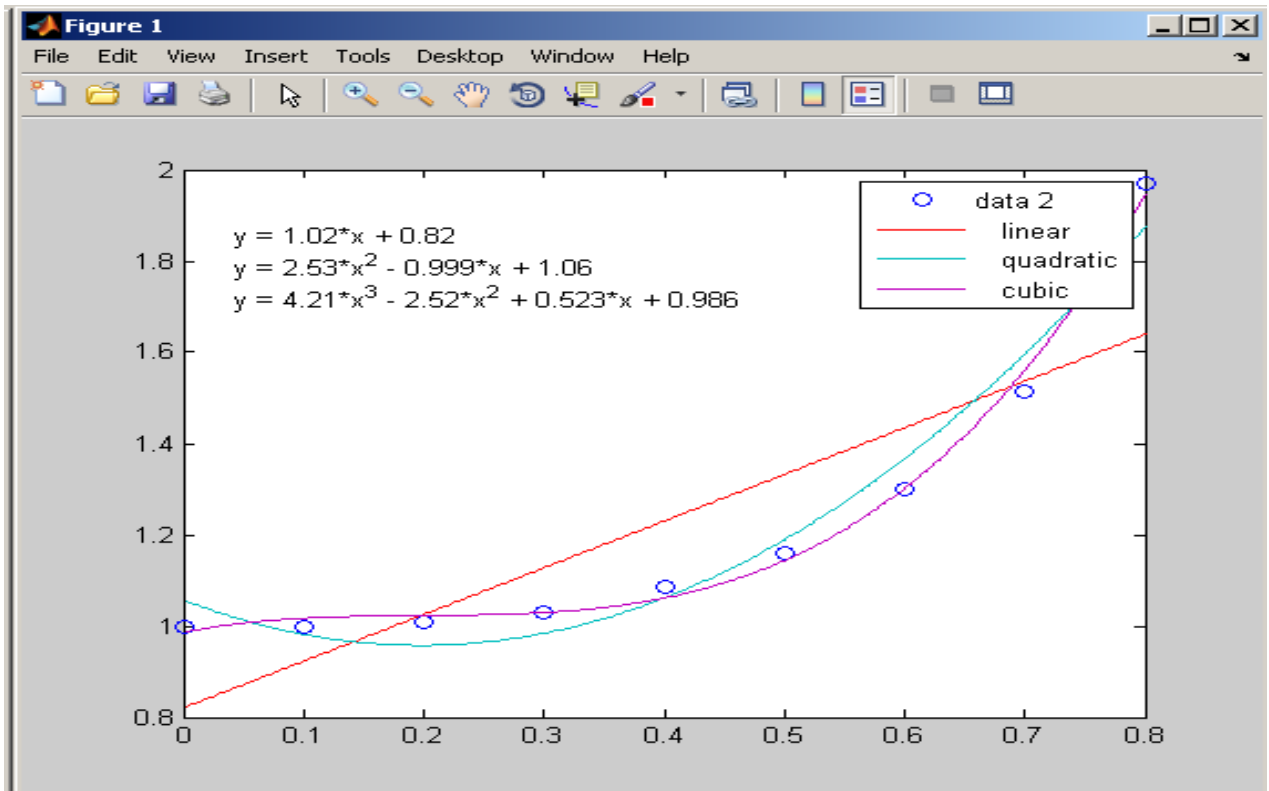


Рисунок 2.5 - Экранная форма аппроксимации кривой влияния мелководья на вращательные производные коэффициента поперечной силы судна  $k_2$

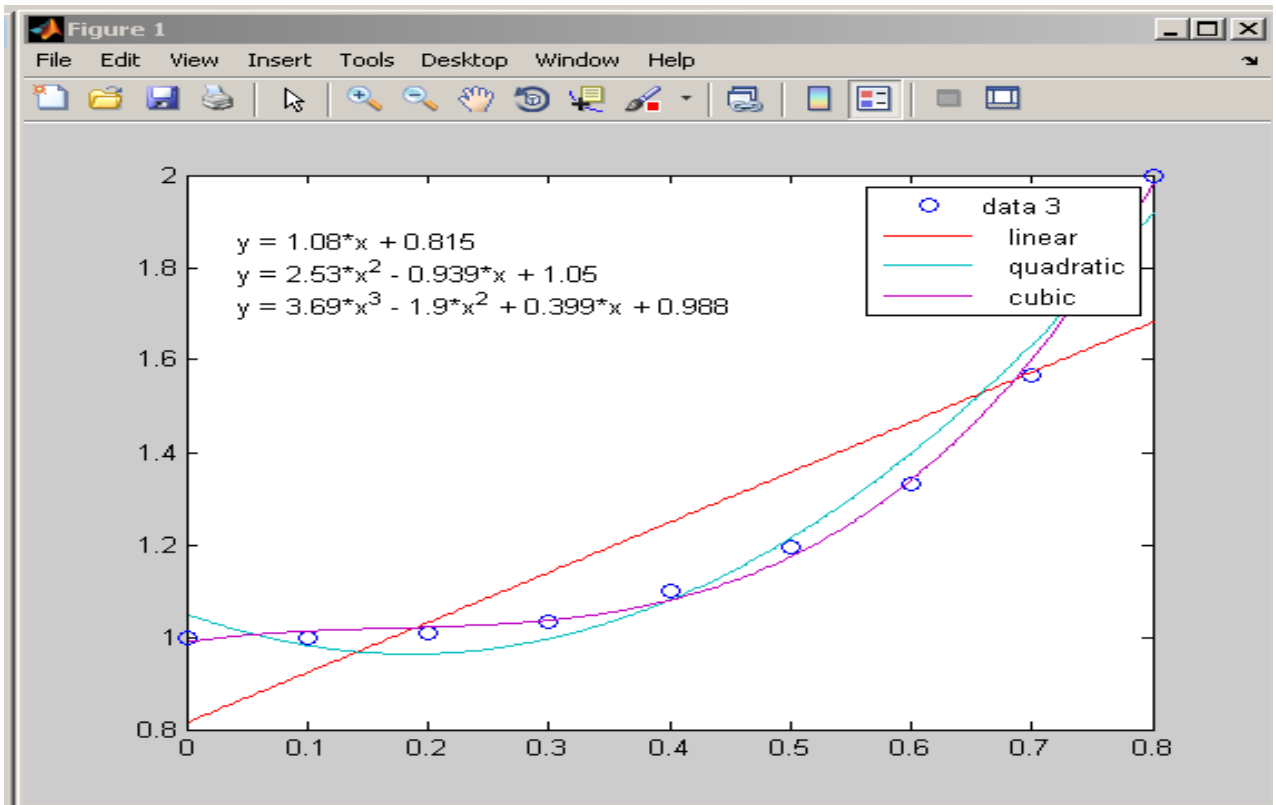


Рисунок 2.6 - Экранная форма аппроксимации кривой влияния мелководья на позиционные производные коэффициента момента рыскания судна  $k_3$

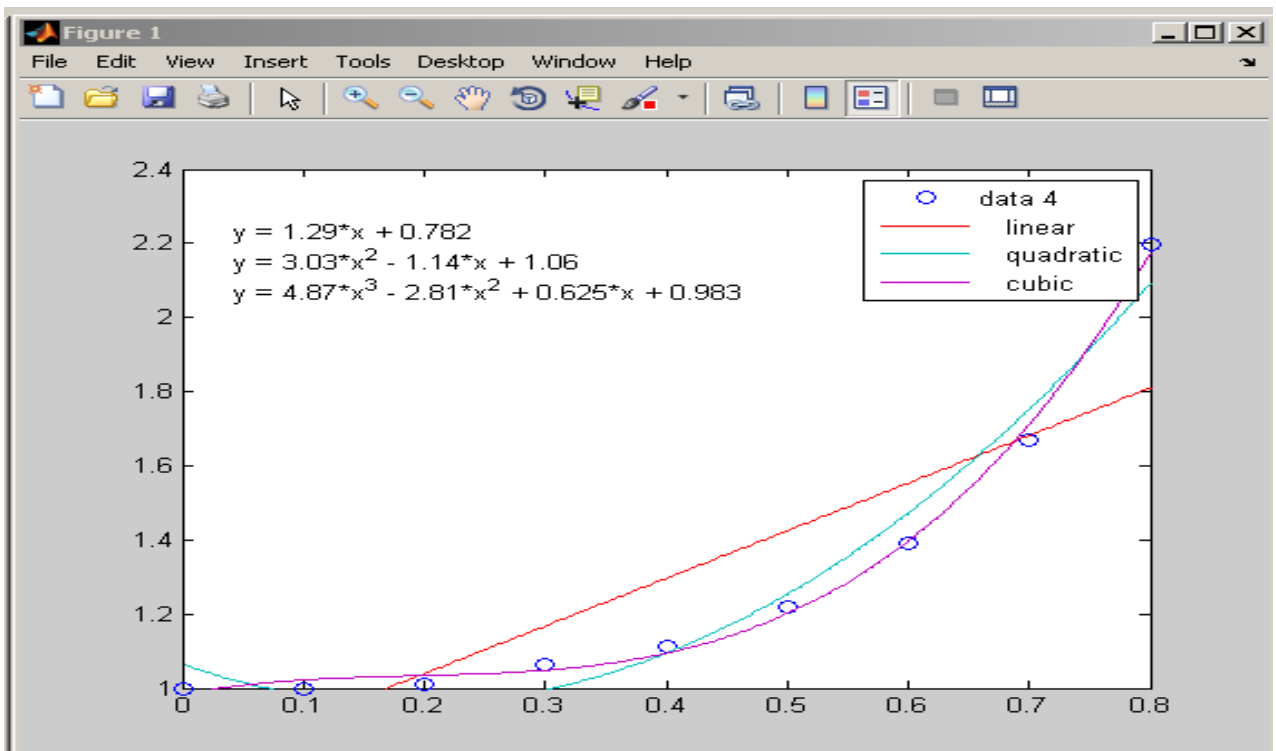


Рисунок 2.7 - Экранная форма аппроксимации кривой влияния мелководья на позиционные производные коэффициента поперечной силы судна  $k_4$



Ниже приводятся аналитические зависимости полученные аппроксимацией кривых влияния мелководья на гидродинамические коэффициенты при уравнениях гидродинамики судна, определяемые в функции  $T/H$  [8]:

$$k_{11} = A_{k11} \left( \frac{T}{H} \right)^3 + B_{k11} \left( \frac{T}{H} \right)^2 + C_{k11} \left( \frac{T}{H} \right) + D_{k11} \quad (2.2)$$

$$k_{22} = A_{k22} \left( \frac{T}{H} \right)^3 + B_{k22} \left( \frac{T}{H} \right)^2 + C_{k22} \left( \frac{T}{H} \right) + D_{k22} \quad (2.3)$$

$$k_{66} = A_{k66} \left( \frac{T}{H} \right)^3 + B_{k66} \left( \frac{T}{H} \right)^2 + C_{k66} \left( \frac{T}{H} \right) + D_{k66} \quad (2.4)$$

$$k_1 = A_{k1} \left( \frac{T}{H} \right)^3 + B_{k1} \left( \frac{T}{H} \right)^2 + C_{k1} \left( \frac{T}{H} \right) + D_{k1} \quad (2.5)$$

$$k_2 = A_{k2} \left( \frac{T}{H} \right)^3 + B_{k2} \left( \frac{T}{H} \right)^2 + C_{k2} \left( \frac{T}{H} \right) + D_{k2} \quad (2.6)$$

$$k_3 = A_{k3} \left( \frac{T}{H} \right)^3 + B_{k3} \left( \frac{T}{H} \right)^2 + C_{k3} \left( \frac{T}{H} \right) + D_{k3} \quad (2.7)$$

$$k_4 = A_{k4} \left( \frac{T}{H} \right)^3 + B_{k4} \left( \frac{T}{H} \right)^2 + C_{k4} \left( \frac{T}{H} \right) + D_{k4} \quad (2.8)$$

В таблице 1 приведены значения поправочных коэффициентов при уравнениях гидродинамики судна для судна №2 из [86].

Таблица 1 - Значения поправочных коэффициентов при уравнениях гидродинамики судна для судна №2 из [86]

Поправочные коэффициенты при уравнениях гидродинамики судна	Значения поправочных коэффициентов	Поправочные коэффициенты при уравнениях гидродинамики судна	Значения поправочных коэффициентов
$A_{k22}$	4,26	$C_{k66}$	0,0873
$B_{k22}$	-0,0602	$A_{k3}$	3,69
$C_{k22}$	0,121	$B_{k3}$	-1,9
$A_{k4}$	4,87	$C_{k3}$	0,399
$B_{k4}$	-2,81	$A_{k1}$	4,07
$C_{k4}$	0,625	$B_{k1}$	-2,4
$A_{k2}$	4,21	$C_{k1}$	0,501
$B_{k2}$	-2,52	$A_{k11}$	-0,638
$C_{k2}$	0,523	$B_{k11}$	4,06
$A_{k66}$	1,54	$C_{k11}$	-0,681
$B_{k66}$	1,96	$D$	1

Примечание: При расчетах значения  $D_i$  получились близкие к единице ( $D_{k22}=0,994$ ,  $D_{k4}=0,983$ ,  $D_{k2} =0,986$ ,  $D_{k66}=0,989$ ,  $D_{k3}=0,988$ ,  $D_{k1} =0,986$ ,  $D_{k11}=1,01$ ), а они должны быть равными единице, т.к. при глубокой воде ( $H \rightarrow \infty$ )  $k_i$  должно быть равно 1. В связи с этим, в полученных аналитических зависимостях значения  $D_i$  приняты равными единице.

## 2.2 Аналитическое описание влияния мелководья на коэффициенты нелинейности поперечной силы и момента

В работах [4, 12, 75] рассмотрены вопросы аппроксимации кривых влияния мелководья на коэффициенты нелинейности поперечной силы и момента. На рисунках 2.8-2.12 приведены экранные формы результатов обработки информации при вычислении коэффициентов  $\bar{a}_1, a_3, a_4, \bar{b}_1, \bar{b}_2$ .

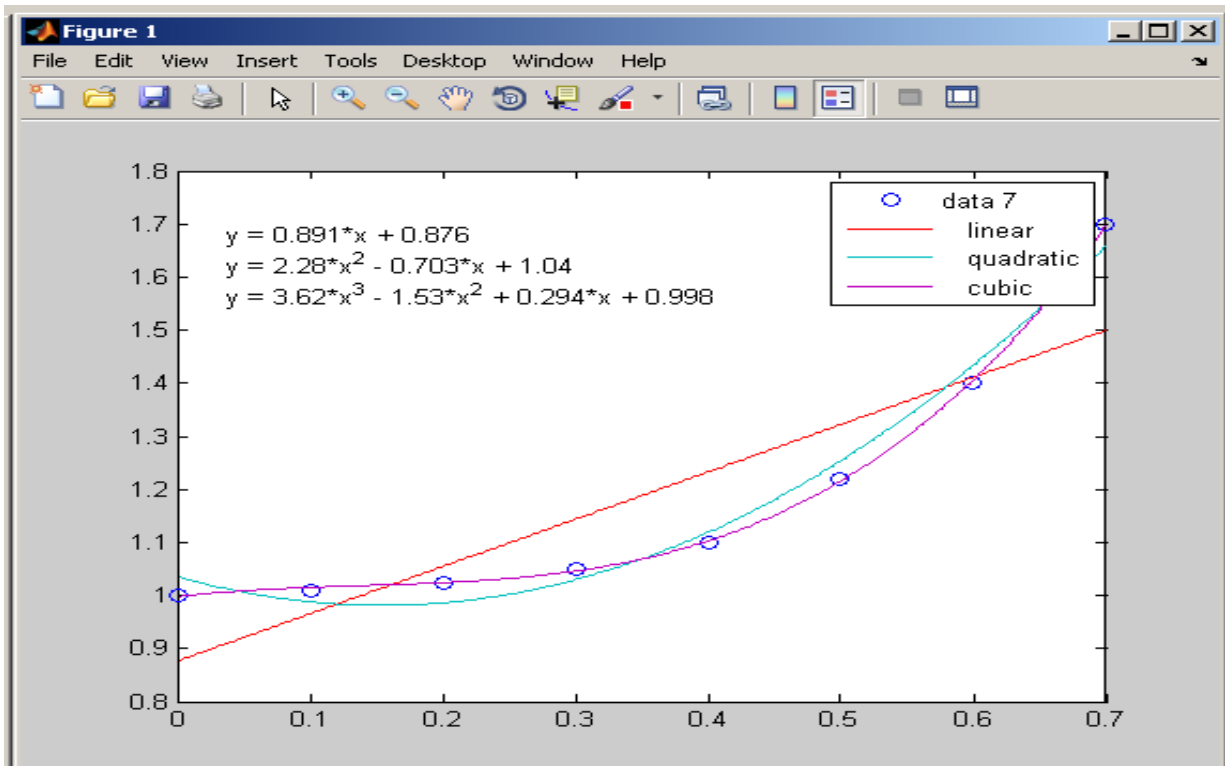


Рисунок 2.8 - Экранная форма результатов обработки информации при вычислении коэффициента  $\bar{a}_1$

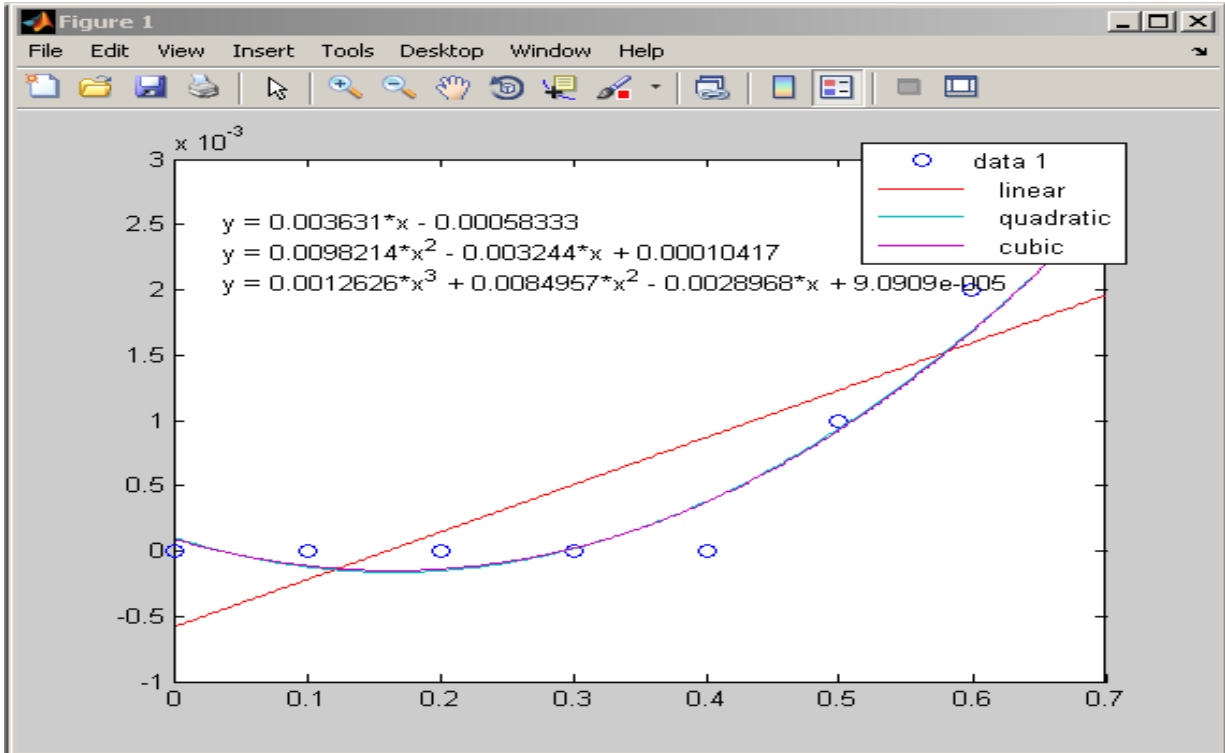


Рисунок 2.9 - Экранная форма результатов обработки информации при вычислении коэффициента  $a_3$

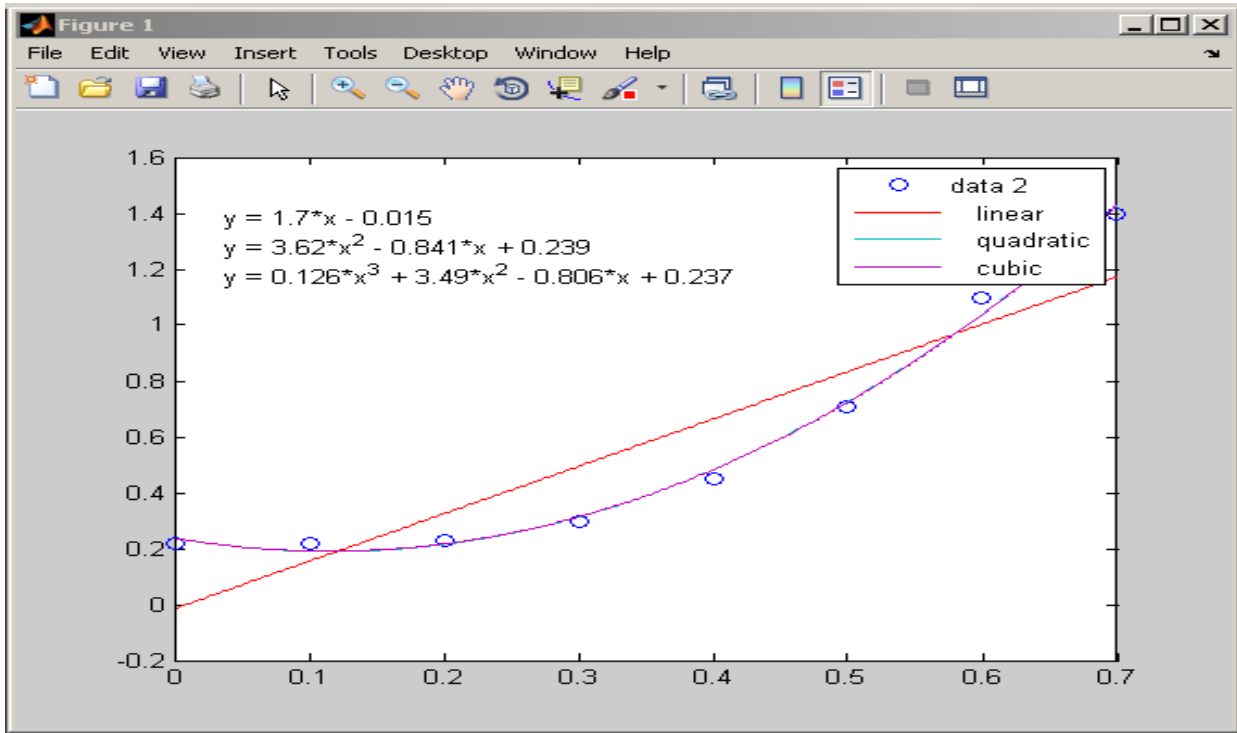


Рисунок 2.10 - Экранная форма результатов обработки информации при вычислении коэффициента  $a_4$

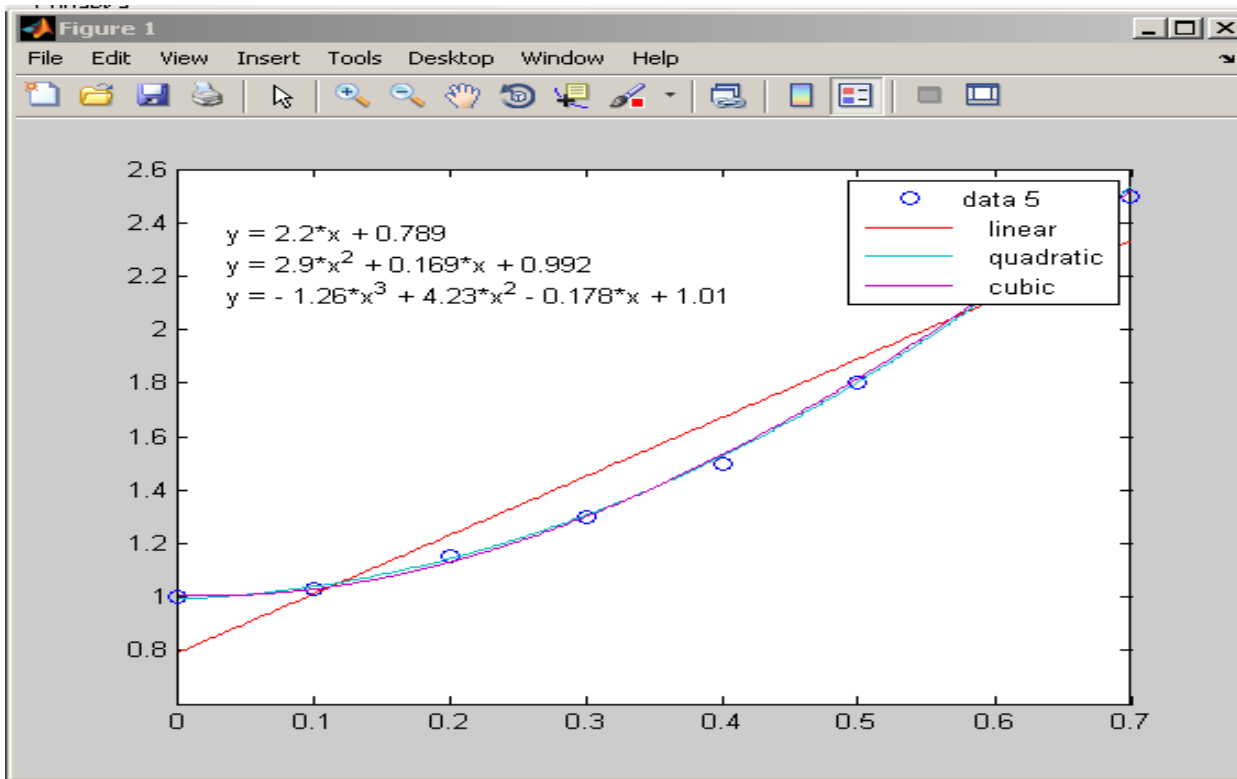


Рисунок 2.11 - Экранная форма результатов обработки информации при вычислении коэффициента  $\bar{b}_1$

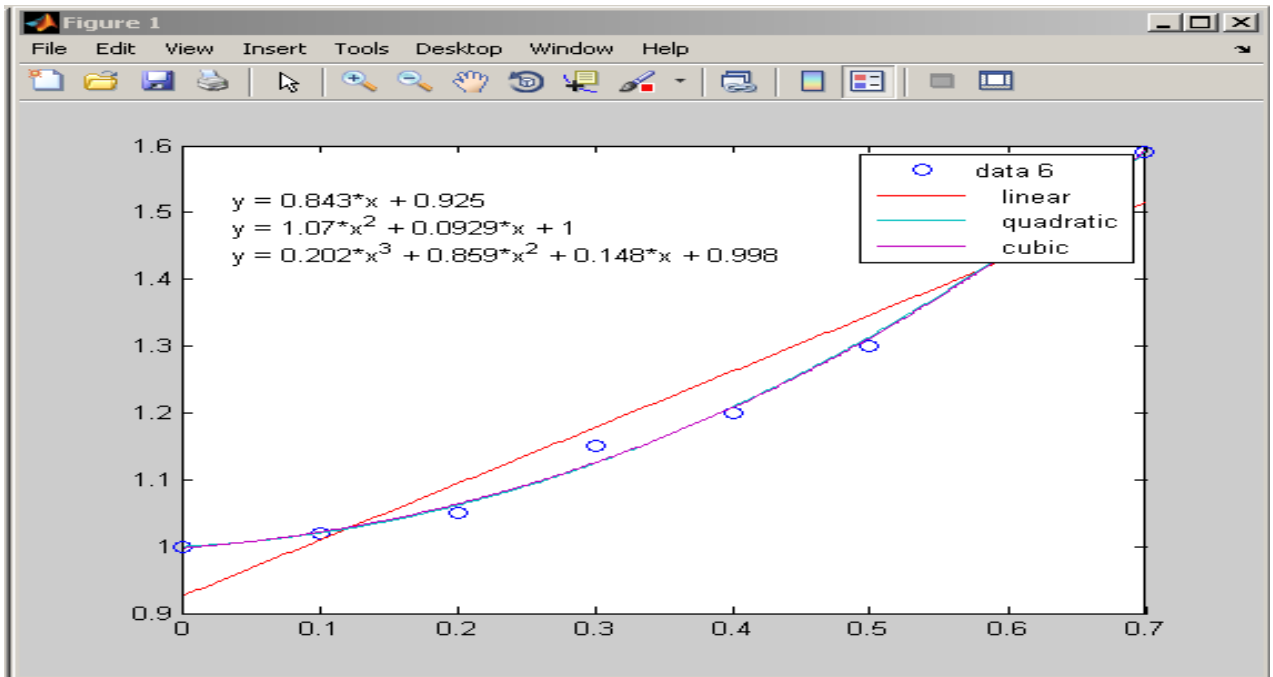


Рисунок 2.12 - Экранная форма результатов обработки информации при вычислении коэффициента  $\bar{b}_2$

Ниже приводятся аналитические зависимости, полученные аппроксимацией кривых влияния мелководья на гидродинамические коэффициенты  $\bar{a}_1, a_3, a_4, \bar{b}_1, \bar{b}_2$ , определяемые в функции  $T/H$  (2.9)-(2.13):

$$\bar{a}_1 = A_{\bar{a}1} \left( \frac{T}{H} \right)^3 + B_{\bar{a}1} \left( \frac{T}{H} \right)^2 + C_{\bar{a}1} \left( \frac{T}{H} \right) + D_{\bar{a}1} \quad (2.9)$$

$$a_3 = A_{a3} \left( \frac{T}{H} \right)^3 + B_{a3} \left( \frac{T}{H} \right)^2 + C_{a3} \left( \frac{T}{H} \right) + D_{a4} \quad (2.10)$$

$$a_4 = A_{a4} \left( \frac{T}{H} \right)^3 + B_{a4} \left( \frac{T}{H} \right)^2 + C_{a4} \left( \frac{T}{H} \right) + D_{a4} \quad (2.11)$$

$$\bar{b}_1 = A_{\bar{b}1} \left( \frac{T}{H} \right)^3 + B_{\bar{b}1} \left( \frac{T}{H} \right)^2 + C_{\bar{b}1} \left( \frac{T}{H} \right) + D_{\bar{b}1} \quad (2.12)$$

$$\bar{b}_2 = A_{\bar{b}2} \left( \frac{T}{H} \right)^3 + B_{\bar{b}2} \left( \frac{T}{H} \right)^2 + C_{\bar{b}2} \left( \frac{T}{H} \right) + D_{\bar{b}2} \quad (2.13)$$

Значения коэффициентов при полиноме  $A_1 \dots A_{12}$  и  $B_1 \dots B_8$  сведены в таблицу 2.

Таблица 2- Значения коэффициентов при полиноме

Коэффициенты при полиноме	Значения коэффициентов при полиноме	Коэффициенты при полиноме	Значения коэффициентов при полиноме
$A_{\bar{a}_1}$	3,62	$C_{a_4}$	-0,806
$B_{\bar{a}_1}$	-1,53	$D_{a_4}$	0,237
$C_{\bar{a}_1}$	0,294	$A_{\bar{b}_1}$	-1,26
$D_{\bar{a}_1}$	0,998	$B_{\bar{b}_1}$	4,23
$A_{a_3}$	0,00126	$C_{\bar{b}_1}$	-0,178
$B_{a_3}$	0,0085	$D_{\bar{b}_1}$	1,01
$C_{a_3}$	-0,0029	$A_{\bar{b}_2}$	0,202
$D_{a_3}$	0,0000909	$B_{\bar{b}_2}$	0,859
$A_{a_4}$	0,126	$C_{\bar{b}_2}$	0,148
$B_{a_4}$	3,49	$D_{\bar{b}_2}$	0,998

С учетом аппроксимированных гидродинамических коэффициентов  $\bar{a}_1, a_3, a_4, \bar{b}_1, \bar{b}_2$ , структурные выражения для коэффициента поперечной силы и момента для мелководья (1.9) можно записать в виде:

$$\begin{aligned}
n = & \beta\pi \frac{T}{L} \left( A_{\bar{a}1} \left( \frac{T}{H} \right)^3 + B_{\bar{a}1} \left( \frac{T}{H} \right)^2 + C_{\bar{a}1} \left( \frac{T}{H} \right) + D_{\bar{a}1} \right) + \\
& + \beta|\beta| \left( \left( 1,055 - 0,251 \frac{B}{T} + 0,021 \left( \frac{B}{T} \right)^2 + 12,49 \frac{T}{L} + 0,0244 \frac{L}{T} - a'_k \right) \times \right. \\
& \times \left. \left( 3,34 \left( \frac{T}{H} \right)^2 - 0,234 \left( \frac{T}{H} \right) + 1 \right) \right) + \\
& \omega \left( A_{a3} \left( \frac{T}{H} \right)^3 + B_{a3} \left( \frac{T}{H} \right)^2 + C_{a3} \left( \frac{T}{H} \right) + D_{a4} \right) + \\
& + \beta|\omega| \left( A_{a4} \left( \frac{T}{H} \right)^3 + B_{a4} \left( \frac{T}{H} \right)^2 + C_{a4} \left( \frac{T}{H} \right) + D_{a4} \right);
\end{aligned} \tag{2.14}$$

$$\begin{aligned}
m = & \beta \left( 3,36 \left( \frac{T}{L} + 0,0136 \right) \times (\bar{s}_k + a_k) \right) \times \\
& \times \left( A_{\bar{b}1} \left( \frac{T}{H} \right)^3 + B_{\bar{b}1} \left( \frac{T}{H} \right)^2 + C_{\bar{b}1} \left( \frac{T}{H} \right) + D_{\bar{b}1} \right) - \\
& - \left( 0,1595 - 0,0469 \frac{B}{T} + 0,00375 \left( \frac{B}{T} \right)^2 \right) \times \\
& \times \left( A_{\bar{b}2} \left( \frac{T}{H} \right)^3 + B_{\bar{b}2} \left( \frac{T}{H} \right)^2 + C_{\bar{b}2} \left( \frac{T}{H} \right) + D_{\bar{b}2} \right)
\end{aligned} \tag{2.15}$$

Таким образом, получены аналитические зависимости влияния мелководья на гидродинамические коэффициенты при уравнениях гидродинамики судна, определяемые в функции отношением осадки судна к глубине акватории ( $T/H$ ), с помощью кривых третьего порядка с максимальной средней квадратической погрешностью  $\delta = 2\%$ .

### 2.3 Математическая модель движения судна на мелководье

Для моделирования движения судна на мелководье с использованием аналитических зависимостей гидродинамических коэффициентов от отношения осадки судна к глубине акватории необходимо безразмерные уравнения движения судна привести к размерному виду [1, 9].

В соответствии с [20, 86] производные размерных и безразмерных величин связаны соотношениями:

$$\frac{d\beta}{d\tau} = \frac{d\beta}{dt} \frac{L}{V_0} \quad (2.16)$$

$$\frac{d\omega}{d\tau} = \frac{d\Omega}{dt} \left( \frac{L}{V_0} \right)^2 \quad (2.17)$$

$$\frac{d\bar{V}}{d\tau} = \frac{dV}{dt} \frac{L}{V_0^2} \quad (2.18)$$

$$\bar{n}_y^\beta = n_y^\beta + n_y^\delta \chi_{\beta} I_2 + I_3 \quad (2.19)$$

$$\bar{m}_y^\beta = m_y^\beta - n_y^\delta \frac{l_p}{L} \chi_{\beta} I_2 - I_3 \frac{l_g}{L} \quad (2.20)$$

$$\bar{n}_y^\omega = n_y^\omega + n_y^\delta \frac{l_p}{L} \chi_{\omega} I_2 + I_4 \frac{l_g}{L} \quad (2.21)$$

$$\bar{m}_y^\omega = m_y^\omega - n_y^\delta \left( \frac{l_p}{L} \right)^2 \chi_{\omega} I_2 - I_4 \left( \frac{l_g}{L} \right)^2 \quad (2.22)$$

$$\hat{n}_y^\omega = m_{11} - \bar{n}_y^\omega \quad (2.23)$$

где:

$$I_2 = \left[ 1 - \frac{F'_{p+pn}}{F_{p+pn}} \left( \frac{\sqrt{1+\sigma_p} - 1}{1 + \sqrt{1+\sigma_p}} \right) \right], \quad I_3 = \sum_{\epsilon} n_{y\epsilon}^\beta \chi_{\beta\epsilon}, \quad I_4 = \sum_{\epsilon} n_{y\epsilon}^\beta \chi_{\omega\epsilon},$$

$m_y^\omega, n_y^\beta, n_y^\omega, m_y^\beta$  - гидродинамические коэффициенты корпуса судна,

причем:

$m_y^\omega$  - гидродинамический коэффициент вращательной производной

горизонтального момента;

$n_y^\beta$  - гидродинамический коэффициент позиционной производной

поперечной силы;

$n_y^\omega$  - гидродинамический коэффициент вращательной производной

поперечной силы;

$m_y^\beta$  - гидродинамический коэффициент позиционной производной



горизонтального момента.

$n_y^\omega, m_y^\beta$  характеризуют линейные компоненты зависимости поперечной силы и горизонтального момента на корпусе от угловой скорости и называются вращательными производными.

По аналогии с ними значения  $n_y^\beta, m_y^\beta$  называются позиционными производными.

$l_\epsilon$  – отстояние диска винта от центра тяжести корабля;

$\chi_\beta$  – поправка, учитывающая справляющее действие корпуса на скос у руля, обусловленный углом дрейфа;

$\chi_\omega$  – поправка, учитывающая справляющее действие корпуса на скос у руля, обусловленный кривизной траектории;

$F'_{p+pn}$  - площадь руля и рудерпоста, попадающая в струю винта;

$F_{p+pn}$  - площадь руля и рудерпоста, соответствующая осадке судна кормой ( $T_k$ );

$\sigma_p$  - коэффициент нагрузки винта;

$\sum_6$  - число винтов;

$n_{y\epsilon}^\beta$  - позиционная производная безразмерного коэффициента поперечной силы от винта;

$\chi_{\beta\epsilon}$  - поправка, учитывающая спрямляющее действие корпуса на скос у винта от угла дрейфа;

$\chi_{\omega\epsilon}$  - поправка, учитывающая спрямляющее действие корпуса на скос у винта от кривизны траектории;

$\bar{n}_y^\omega$  - гидродинамический коэффициент вращательной производной поперечной силы.

Здесь гидродинамические коэффициенты корпуса, включающие влияние рулей и винтов, отмечены черточками.

Кроме того, в соответствии с [86], безразмерные коэффициенты инерции связаны с характеристиками судна следующими выражениями:

$$m_{11} = 2 \frac{C_b B}{\sigma L} \left( \frac{\rho W + \lambda_{11}}{\rho W} \right) \quad (2.24)$$

$$m_{22} = 2 \frac{C_b B}{\sigma L} \left( \frac{\rho W + \lambda_{22}}{\rho W} \right) \quad (2.25)$$

$$m_{66} = 2 \frac{C_b B}{\sigma L} \rho_z^{-2} \left( \frac{\rho W \rho_z^2 + \lambda_{66}}{\rho W \rho_z^2} \right) \quad (2.26)$$

где  $C_b$  – коэффициент общей полноты;

$B$  – ширина судна по действующую ватерлинию;

$\sigma$  - коэффициент полноты площади диаметральной плоскости;

$\rho$  - массовая плотность жидкости (воды);

$\rho_z$  – радиус инерции массы корпуса относительно вертикальной оси;

$W$  – объемное водоизмещение, рассчитываемое через произведение главных размерений и коэффициентов общей полноты  $C_b$ :  $W = C_b \times L \times B \times T$ .

Значения безразмерных коэффициентов тяги и силы сопротивления движению имеют вид:

$$n_p = n_{x0} \frac{1}{\bar{V}^3} + 0,3 \omega^2 \frac{D_e^2}{S} = n_{x0} \left( \frac{V_0}{V} \right)^3 + 0,3 \left( \frac{\Omega L}{V_0} \right)^2 \frac{D_e^2}{\sigma L T} \quad (2.27)$$

$$n_x = n_{x0} + 0,4 n_y^\delta \delta^2 \quad (2.28)$$

где  $D_e$  – диаметр гребневого винта.

После всех преобразований и подстановок размерные уравнения движения судна на мелководье примут вид [9, 13]:

$$\begin{aligned}
\frac{d\beta}{dt} = & \frac{I_1}{\left[ A_{k22} \left( \frac{T}{H} \right)^3 + B_{k22} \left( \frac{T}{H} \right)^2 + C_{k22} \left( \frac{T}{H} \right) + D \right]} \times \\
& \times \left( n_y^\delta \delta - \beta \left[ n_y^\beta \left[ A_{k4} \left( \frac{T}{H} \right)^3 + B_{k4} \left( \frac{T}{H} \right)^2 + C_{k4} \left( \frac{T}{H} \right) + D \right] + n_y^\delta \chi_\beta I_2 + I_3 \right) + \right. \\
& + \Omega \frac{L}{V_0} \left( -n_y^\omega \left[ A_{k2} \left( \frac{T}{H} \right)^3 + B_{k2} \left( \frac{T}{H} \right)^2 + C_{k2} \left( \frac{T}{H} \right) + D \right] - n_y^\delta \frac{l_p}{L} \chi_\omega I_2 + I_4 \frac{l_a}{L} \right) + \quad (2.29.1) \\
& + I_5 \left[ A_{k11} \left( \frac{T}{H} \right)^3 + B_{k11} \left( \frac{T}{H} \right)^2 + C_{k11} \left( \frac{T}{H} \right) + D \right] - \\
& - I_7 \left[ \beta \pi \frac{T}{L} \left( A_{a1} \left( \frac{T}{H} \right)^3 + B_{a1} \left( \frac{T}{H} \right)^2 + C_{a1} \left( \frac{T}{H} \right) + D_{a1} \right) + I_8 I_9 + \right. \\
& + \frac{\Omega L}{V_0} \left( A_{a3} \left( \frac{T}{H} \right)^3 + B_{a3} \left( \frac{T}{H} \right)^2 + C_{a3} \left( \frac{T}{H} \right) + D_{a4} \right) + \\
& \left. + \beta \left| \frac{\Omega L}{V_0} \left( A_{a4} \left( \frac{T}{H} \right)^3 + B_{a4} \left( \frac{T}{H} \right)^2 + C_{a4} \left( \frac{T}{H} \right) + D_{a4} \right) \right| \right];
\end{aligned}$$

$$\begin{aligned}
\frac{d\Omega}{dt} = & \frac{I_6}{\left[ A_{k66} \left( \frac{T}{H} \right)^3 + B_{k66} \left( \frac{T}{H} \right)^2 + C_{k66} \left( \frac{T}{H} \right) + D \right]} \times \\
& \times \left( n_y^\delta \frac{l_p}{L} + \beta \left( m_z^\beta \left[ A_{k3} \left( \frac{T}{H} \right)^3 + B_{k3} \left( \frac{T}{H} \right)^2 + C_{k3} \left( \frac{T}{H} \right) + D \right] - n_y^\delta \frac{l_p}{L} \chi_\beta I_2 - I_3 \frac{l_a}{L} \right) + \right. \\
& + \Omega \frac{L}{V_0} \left( m_y^\omega \left[ A_{k1} \left( \frac{T}{H} \right)^3 + B_{k1} \left( \frac{T}{H} \right)^2 + C_{k1} \left( \frac{T}{H} \right) + D \right] - n_y^\delta \left( \frac{l_p}{L} \right)^2 \chi_\omega I_2 - I_4 \left( \frac{l_a}{L} \right)^2 \right) + \quad (2.29.2) \\
& + I_7 \left[ I_{10} \left( A_{\bar{b}1} \left( \frac{T}{H} \right)^3 + B_{\bar{b}1} \left( \frac{T}{H} \right)^2 + C_{\bar{b}1} \left( \frac{T}{H} \right) + D_{\bar{b}1} \right) - \right. \\
& \left. - I_{11} \left( A_{\bar{b}2} \left( \frac{T}{H} \right)^3 + B_{\bar{b}2} \left( \frac{T}{H} \right)^2 + C_{\bar{b}2} \left( \frac{T}{H} \right) + D_{\bar{b}2} \right) \right]
\end{aligned}$$

$$\frac{dV}{dt} = \frac{VV_0}{L} \left[ \frac{1}{I_5 \left( \left[ A_{k11} \left( \frac{T}{H} \right)^3 + B_{k11} \left( \frac{T}{H} \right)^2 + C_{k11} \left( \frac{T}{H} \right) + D \right] \right)} I_{12} - \right. \\ \left. - \beta \frac{\left( \rho W + \lambda_{22} \left[ A_{k22} \left( \frac{T}{H} \right)^3 + B_{k22} \left( \frac{T}{H} \right)^2 + C_{k22} \left( \frac{T}{H} \right) + D_{k22} \right] \right)}{\left( \rho W + \lambda_{11} \left[ A_{k11} \left( \frac{T}{H} \right)^3 + B_{k11} \left( \frac{T}{H} \right)^2 + C_{k11} \left( \frac{T}{H} \right) + D_{k11} \right] \right)} \frac{\Omega L}{V_0} - \frac{d\beta}{dt} \frac{L}{V_0} \right]; \quad (2.29.3)$$

где:  $I_1 = \frac{V_0^2 \sigma \rho W}{2C_b B V (\rho V + \lambda_{22})}$ ,  $I_5 = \frac{2C_b B}{\sigma L} \left( \frac{\rho W + \lambda_{11}}{\rho W} \right)$ ,

$$I_6 = \frac{V_0^2 \sigma \rho W L}{2C_b B (\rho W \rho_z^2 + \lambda_{66})}, \quad I_7 = \beta \Omega \frac{L}{V_0} \text{sign}(\beta, \Omega),$$

$$I_8 = \beta |\beta| \left( \left( 1,055 + 0,251 \frac{B}{T} + 0,021 \left( \frac{B}{T} \right)^2 + 12,49 \frac{T}{L} + 0,0244 \frac{L}{T} - a'_k \right) \right)$$

$$I_9 = \left( 3,34 \left( \frac{T}{H} \right)^2 - 0,234 \left( \frac{T}{H} \right) + 1 \right), \quad I_{10} = \beta \left( 3,36 \left( \frac{T}{L} + 0,0136 \right) \times (2S_k / LT + a_k) \right),$$

$$I_{11} = \left( 0,1595 - 0,0469 \frac{B}{T} + 0,00375 \left( \frac{B}{T} \right)^2 \right), \quad I_{12} = \left( n_{x0} \left( \left( \frac{V_0}{V} \right)^3 - 1 \right) + 0,3 \frac{D_B^2}{\sigma L T} \left( \frac{\Omega L}{V_0} \right)^2 - 0,4 n_y^\delta \delta^2 \right)$$

По этим уравнениям могут быть вычислены угол дрейфа, угловая и линейная скорости судна при маневрировании на мелководье.

В таблице 3 приведены размерные значения параметров судна № 2, взятые из [86].

Таблица 3- Размерные значения параметров судна №2, взятые из [86].

Параметры судна	Значения параметров	Параметры судна	Значения параметров	Параметры судна	Значения параметров
$V_0$	5,144м	$m_y^\omega$	-0,026	$F_{p+pn}$	10м <sup>2</sup>
$L$	100м	$\lambda_{11}$	109,457	$F'_{p+pn}$	0м <sup>2</sup>
$C_b$	0,64	$\lambda_{22}$	2476,755	$\sigma_p$	0,289

$T$	4,6м	$\lambda_{66}$	1338600	$l_p$	48м
$W$	5004,8	$m$	-0,120	$l_e$	47м
$B$	17м	$n$	0,432	$\rho_z$	25
$n_y^\beta$	0,249	$\chi_\beta$	0,30	$S_k$	0,06
$n_y^\omega$	0,014	$\chi_\omega$	0,50	$D_e$	3
$n_y^\delta$	0,035	$\chi_{\beta e}$	0,30	$\sum_e$	2
$n_{y\beta}^\beta$	0,011	$\chi_{\omega e}$	0,85	$\sigma$	0,971
$m_y^\beta$	0,084	$n_{x0}$	0,009	$\rho$	1

### Выводы по второй главе

1. Выполнена аппроксимация кривых влияния мелководья на гидродинамические коэффициенты с использованием пакета программ MatLab (точность аппроксимации 2 %).
2. Система безразмерных дифференциальных уравнений гидродинамики судна приведена к размерному виду.
3. Получены аналитические зависимости коэффициентов гидродинамики в зависимости от отношения осадки судна к глубине.
3. Разработана математическая модель движения судна на мелководье, в которой аналитически учитываются изменения коэффициентов гидродинамики судна в зависимости от отношения осадки судна к глубине.

## **ГЛАВА 3. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ДВИЖЕНИЯ СУДНА НА МЕЛКОВОДЬЕ ПО КРИВОЛИНЕЙНОЙ ТРАЕКТОРИИ**

### **3.1 Постановка задачи, допущения и ограничения, принимаемые при моделировании движения судна на мелководье по заданной криволинейной траектории**

Качество системы автоматического управления судном определяется алгоритмом управления и управляемостью судна. При ухудшении условий плавания понижается управляемость судна. Это объясняется влиянием возмущающих воздействий, из которых наиболее сильное влияние оказывает мелководье, т.к. с уменьшением запаса воды под килем наблюдается рост сопротивления движению судна и уменьшение скорости движения.

Большой интерес представляет исследование движения судна на мелководье при переменной глубине по криволинейной траектории. Вопросы управления судами по криволинейной траектории рассматриваются в работах [7, 10, 15, 52, 53, 81, 82].

Следует отметить, что в этих работах, в частности, рассматривается управление движением судна для специальных систем навигации. Так, в работе [7] рассматривается движение судна по криволинейной траектории, вид которой определяется траекторией укладки ведущего кабеля по морскому или речному дну. При моделировании движения судна по ведущему кабелю траектория укладки кабеля аппроксимируется кусочно-линейными уравнениями и окружностями, что заведомо снижает возможности моделирования.

При плавании в стесненных районах и управлении судном судоводителем по прямолинейным участкам пути контроль местоположения судна обеспечивается с довольно высокой точностью. В то же время при движении по криволинейной траектории судоводителем моменты начала перекладки руля определяются глазомерно.

При этом на судно действуют различные возмущающие факторы: ветер, течение, волнение и т.д. Кроме того, по сравнению с ветровым дрейфом, в

узкостях более существенным является воздействие течения [55].

Обычно, скорость и курс течения представляются стационарными случайными процессами. При составлении модели движения судна на мелководье обычно ограничиваются детерминированным подходом и полагают [69]:

$$V_T = 0, \quad \text{при } t \leq 0,$$

$$V_T = \text{const}, \quad K_T = \text{const}, \quad \text{при } t > 0$$

где  $V_T$  – скорость течения,

$K_T$  – курс течения

Как известно [7], управление судном по заданной траектории осуществляется при допущении, что отклонения курса и бокового смещения судна от заданной траектории являются малыми.

Кроме того, обычно центр тяжести судна не совпадает с центром поворота судна. Однако для упрощения решения задачи считается, что поворот судна происходит вокруг центра тяжести.

### **3.2 Геометрические соотношения, возникающие при движении судна по криволинейной траектории**

В связи с тем, что в настоящее время в составе систем управления судном используются ЭВМ, то траектория движения судна может быть задана аналитически.

Однако, судоводитель чаще всего траекторию движения корабля задает графически [2, 3, 7, 10, 52, 53, 81].

С другой стороны, при моделировании движения судна с привязкой к конкретному географическому району, траектория движения также задается графически.

В связи с этим заданная траектория движения судна аппроксимируется аналитической зависимостью, по аналогии с алгоритмом аппроксимации кривых влияния гидродинамических коэффициентов в пакете программ MatLab, приведенным в главе 2.

Отметим, что при управлении судном по заданной траектории, координаты судна необходимо определять в маршрутной, подвижной системе координат.

Сказанное поясняется рисунком 3.1, где приведены геометрические соотношения, возникающие при движении судна по криволинейной траектории.

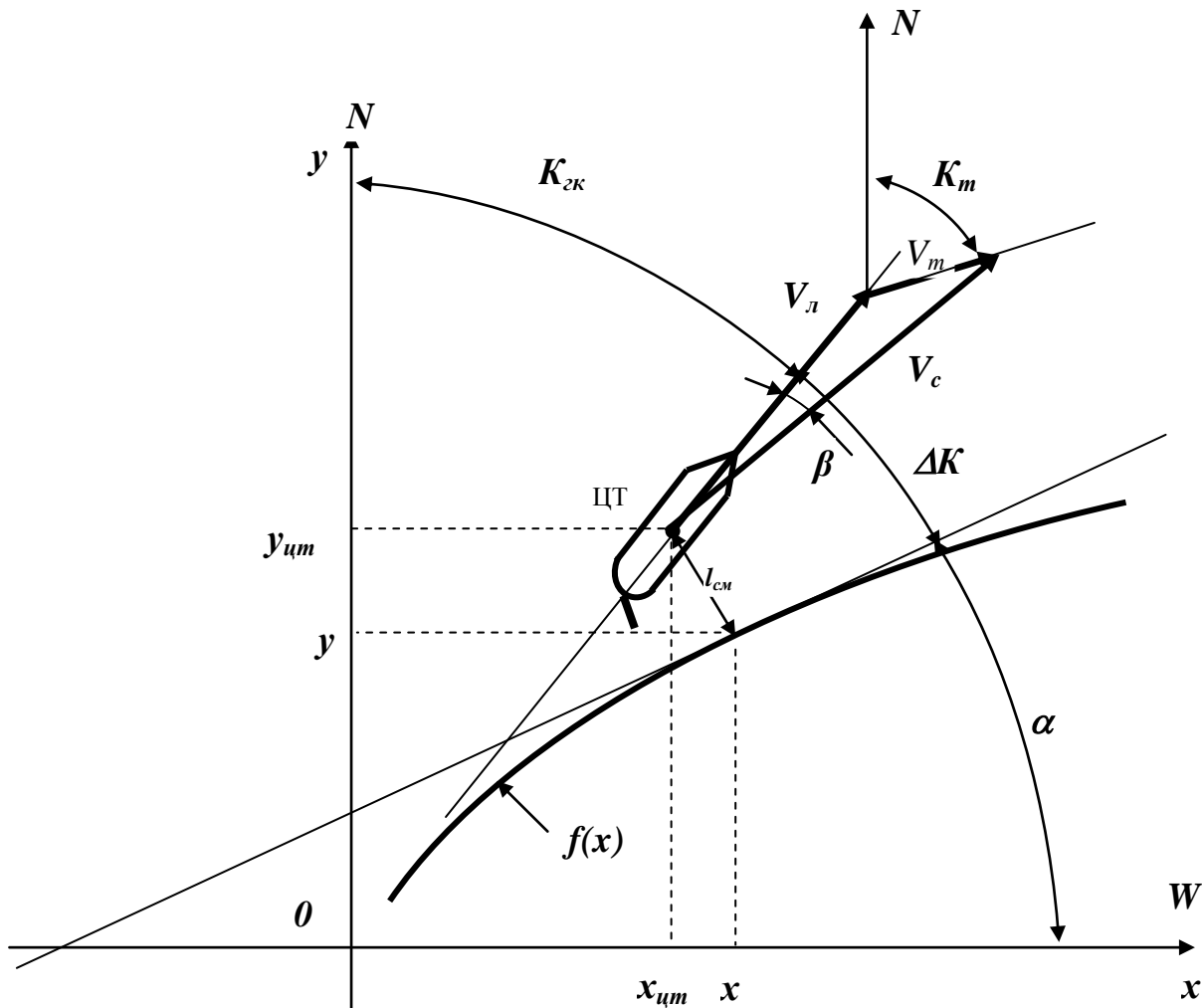


Рисунок 3.1 - Геометрические соотношения, возникающие при движении судна по криволинейной траектории

На рисунке обозначены:  $\alpha$  - заданное направление судна, которое является касательной к кривой  $y = f(x)$ , в точке пересечения перпендикуляра, опущенного из центра тяжести судна на кривую,  $\alpha = \arctg f'(x)$ .  $N$  - направление на север;  $W$  - направление на восток;  $K_{зк}$  - курс судна по гирокомпасу;  $K_m$  - курс течения;  $V_l$  - скорость судна по лагу;  $V_c$  - скорость судна с учетом течения;  $V_m$  - скорость течения;  $\Delta K_{ГК}$  - курс судна в маршрутной системе координат;  $l_{см}$  - смещение судна относительно заданной траектории; ЦТ - центр тяжести судна;  $x_{цт}$  -



абсцисса центра тяжести судна;  $y_{цт}$  – ордината центра тяжести судна;  $x, y$  – соответственно абсцисса и ордината точки пересечения нормали, опущенной из центра тяжести судна с заданной траекторией движения судна;  $\beta$  - угол дрейфа.

Для моделирования управления судном по заданной криволинейной траектории необходимо иметь координаты местоположения судна в неподвижной системе координат, которые могут быть определены по выражениям:

$$x_{ум} = x_0 + \int_0^t V_n \sin K_{zk} dt + V_T \sin K_T t \quad (3.1)$$

$$y_{ум} = y_0 + \int_0^t V_n \cos K_{zk} dt + V_T \cos K_T t \quad (3.2)$$

$$K_{zk} = K_{zk0} + \int_0^t \Omega dt \quad (3.3)$$

где:  $x_0$  – начальное значение абсциссы центра тяжести судна;

$y_0$  – начальное значение ординаты центра тяжести судна;

$K_{zk0}$  - начальное значение истинного курса судна.

Для управления судном по криволинейной траектории необходимо иметь значение бокового смещения судна  $l_{см}$  и отклонение от заданного курса  $\Delta K$ .

Боковое смещение судна  $l_{см}$  определяется как расстояние от точки нахождения центра тяжести судна до точки пересечения нормали, опущенной из центра тяжести судна на заданную траекторию движения судна.

Отклонение от заданного курса  $\Delta K$  находится как угол между диаметральной плоскостью судна и касательной к точке пересечения нормали, проведенной от центра тяжести судна до пересечения нормали с заданной линией криволинейного движения

Боковое смещение судна находится по формуле:

$$l_{см} = \sqrt{(x_{ум} - x)^2 + (y_{ум} - y)^2} \quad (3.4)$$

Смещение судна по формуле (3.4) находится с неоднозначностью. Неоднозначность разрешается по формуле:

$$l_{cm} = \text{sign}(y_{um} - y) |l_{cmr}| \quad (3.5)$$

где  $l_{cmr}$  – значение  $l_{cm}$ , вычисленное по формуле (3.4).

Для определения отклонений судна от заданной криволинейной траектории необходимо найти уравнение касательной к точке с координатами  $x$ ,  $y$ , которая может быть найдена по формуле:

$$y_{um} - y = f'(x)(x_{um} - x) \quad (3.6)$$

Из рисунка 3.1 видно, что отклонение судна от заданного курса будет равно:

$$\Delta K = \pi/2 - K_{zk} - \alpha = \pi/2 - K_{zk} - \arctg f'(x) \quad (3.7)$$

На рисунке 3.2 приведена структурная схема предполагаемой системы управления судном по заданной траектории. Здесь по заданной траектории, координатам местоположения и курсу судна вычислительное устройство определяет ошибку по курсу и смещению от заданной траектории.

Выше получены математические описания всех устройств, входящих в систему, кроме авторулевого и рулевого устройства.

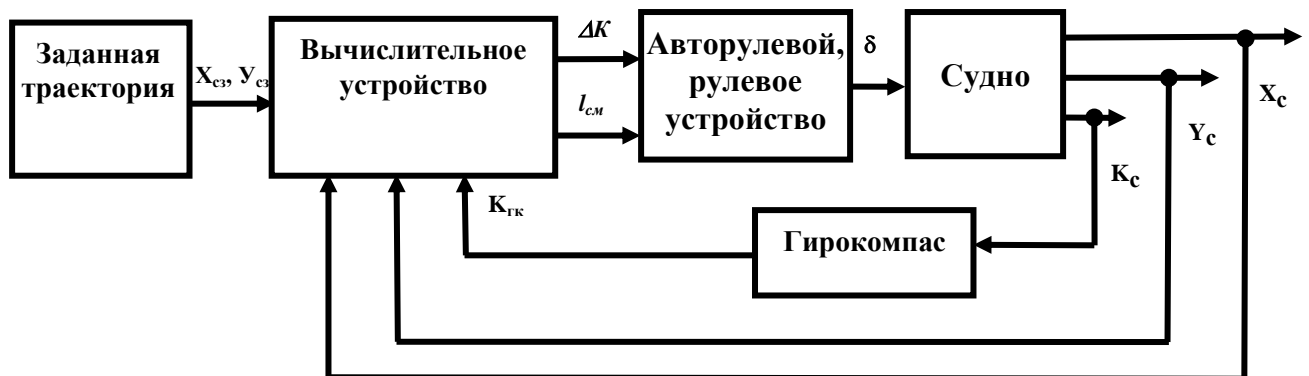


Рисунок 3.2 - Система управления судном по заданной криволинейной траектории

Исполнительной частью системы автоматического управления судном по заданной траектории является рулевое устройство, которое может быть описано как апериодическое звено первого порядка [11], то есть предполагается, что

между управляющим воздействием и управляющим сигналом  $U(t)$  существует связь, описываемая выражением:

$$\frac{d\delta}{dt} = \frac{U(t) - \delta}{T} \quad (3.8)$$

где  $T$  - инерционная постоянная времени, значения которой колеблются в пределах 3÷4 сек.

При этом на скорость и величину угла кладки руля налагаются ограничения.

Обычно на морских транспортных судах

$$|\delta| \leq 35^\circ = 0,61 \text{ рад} \quad (3.9)$$

$$\left| \frac{d\delta}{dt} \right| \leq 2 \div 4 \text{ рад/с} = 0,035 \div 0,07 \frac{1}{c} \quad (3.10)$$

Рулевой привод имеет зону нечувствительности, которая обычно равна:

$$\Delta\delta = 0,5 \div 1^\circ$$

В современных авторулевых формируется пропорционально-интегрально-дифференциальный закон управления вида:

$$U(t) = \frac{1}{K_{oc}} \Delta K + \frac{K_{mz}}{K_{oc}} \frac{d\Delta K}{dt} + K_4 \int_0^t \Delta K dt + K_5 I_{cm} \quad (3.11)$$

где  $K_{oc}$  – коэффициент обратной связи;

$K_{mz}$  – коэффициент тахогенератора, равный отношению скорости вращения тахогенератора к его выходному напряжению в установившемся режиме;

$K_4$  – коэффициент при интегральной составляющей закона управления по курсу судна;

$K_5$  – коэффициент при пропорциональной составляющей закона управления по боковому смещению судна.

В работах [7, 10] для управления судном по величине бокового смещения и углу отклонения от заданного курса предлагается следующий закон управления рулём:

$$U(t) = K_1 \Delta K + K_2 \frac{d\Delta K}{dt} + K_3 l_{cm} + K_4 \frac{dl_{cm}}{dt} \quad (3.12)$$

Для устранения статической ошибки в закон управления может быть введен сигнал, пропорциональный интегралу от бокового смещения. Таким образом, необходимый закон управления будет вида:

$$U(t) = K_1 \Delta K + K_2 \frac{d\Delta K}{dt} + K_3 l_{cm} + K_4 \frac{dl_{cm}}{dt} + K_5 \int_0^t l_{cm} dt \quad (3.13)$$

Однако в работе [7] показано, что введение в закон управления интегральной составляющей приводит к появлению значительных ошибок.

### 3.3 Выбор параметров авторулевого

В связи с вышеизложенным, при моделировании системы управления по заданной траектории будет использован закон управления вида:

$$U(t) = K_1 \Delta K + K_2 \frac{d\Delta K}{dt} + K_3 l_{cm} + K_4 \frac{dl_{cm}}{dt}, \quad (3.14)$$

рекомендованный в работах [7, 10]

При этом коэффициенты  $K_1 \dots K_4$  должны быть выбраны из областей устойчивости системы.

Обычно выбор параметров при исследовании устойчивости систем осуществляется построением областей устойчивости в плоскости одного или двух параметров.

На практике чаще всего применяют наиболее общий метод построения областей устойчивости, предложенный Ю. И. Неймарком и названный им методом  $D$ -разбиения.

Найдем без учета влияния мелководья передаточную функцию судна №2 из [86]. При этом в системе уравнений 2.29 не будем учитывать изменение скорости. Исключим также из уравнений 2.29.1 и 2.29.2. нелинейные

составляющие.

Получим:

$$\left. \begin{aligned} \frac{d\beta}{dt} &= I_1 \left( (n_y^\delta \delta - \beta (n_y^\beta + n_y^\delta \chi_\beta I_2 + I_3)) + \right. \\ &\quad \left. + \Omega \frac{L}{V_0} \left( (-n_y^\omega - n_y^\delta \frac{l_p}{L} \chi_\omega I_2 + I_4 \frac{l_e}{L}) + I_5 \right) \right) \\ \frac{d\Omega}{dt} &= I_6 \left( (n_y^\delta \delta \frac{l_p}{L} + \beta (m_y^\beta - n_y^\delta \frac{l_p}{L} \chi_\beta I_2 - I_3 \frac{l_e}{L}) + \right. \\ &\quad \left. + \Omega \frac{L}{V_0} (m_y^\omega - n_y^\delta \left( \frac{l_p}{L} \right)^2 \chi_\omega I_2 - I_4 \left( \frac{l_e}{L} \right)^2) \right) \end{aligned} \right\} (3.15) \quad (53)$$

Для нахождения передаточной функции судна, уравнения (3.15) напишем в символической форме, при этом исключим из уравнений (3.15) переменную  $\beta$ .

Не приводя промежуточных расчетов, приведем окончательный результат, который имеет вид:

$$\begin{aligned} \Omega(p^2 + pI_1(n_y^\beta + n_y^\delta \chi_\beta I_2 + I_3) - I_2 \frac{L}{V_0} \left( (-n_y^\omega - n_y^\delta \frac{l_p}{L} \chi_\omega I_2 + I_4 \frac{l_e}{L}) + I_5 \right) - pI_6 \frac{L}{V_0} \left( m_y^\omega - I_2 n_y^\delta \frac{l_p}{L} \chi_\omega - \right. \\ \left. - I_4 \left( \frac{l_e}{L} \right)^2 \right) + I_6 I_1 \frac{L}{V_0} (n_y^\beta + n_y^\delta \chi_\beta I_2 + I_3) \left( m_y^\omega - n_y^\delta \frac{l_p}{L} \chi_\omega I_2 - I_4 \left( \frac{l_p}{L} \right)^2 \right) \Big) = \\ = \delta \left( I_6 n_y^\delta \frac{l_p}{L} p + I_6 n_y^\delta \frac{l_p}{L} I_1 (n_y^\beta + n_y^\delta \chi_\beta I_2 + I_3) + n_y^\delta I_6 I_1 \right) \end{aligned} \quad (3.16)$$

В полученном уравнении входной величиной является  $\delta$  – угол перекадки руля, а выходной  $\Omega$  – угловая скорость вращения судна.

Тогда передаточная функция судна будет иметь вид:

$$W\left(\frac{\Omega}{\delta}\right) = \frac{I_6 n_y^\delta \frac{l_p}{L} p + I_6 n_y^\delta \frac{l_p}{L} I_1 (n_y^\beta + n_y^\delta \chi_\beta I_2 + I_3) + n_y^\delta I_6 I_1}{p^2 + pI_1(n_y^\beta + n_y^\delta \chi_\beta I_2 + I_3) - I_2 \frac{L}{V_0} \left( (-n_y^\omega - n_y^\delta \frac{l_p}{L} \chi_\omega I_2 + I_4 \frac{l_e}{L}) + I_5 \right) - \dots} \quad (3.17)$$

$$\dots - pI_6 \frac{L}{V_0} \left( m_y^\omega - I_2 n_y^\delta \frac{l_p}{L} \chi_\omega - I_4 \left( \frac{l_e}{L} \right)^2 \right) + I_6 I_1 \frac{L}{V_0} (n_y^\beta + n_y^\delta \chi_\beta I_2 + I_3) \left( m_y^\omega - n_y^\delta \frac{l_p}{L} \chi_\omega I_2 - I_4 \left( \frac{l_p}{L} \right)^2 \right)$$

Подставляя вместо параметров судна №2 из [86] их числовые значения, получаем передаточную функцию судна :

$$W_c = \frac{0,000704 (58,83p + 1)}{(51,34p^2 + 7,23p + 1)} \quad (3.18)$$

На рисунке 3.3а приведена структурная схема системы управления судном по заданной траектории. Для удобства синтеза и выбора параметров системы управления судном  $K1$ ,  $K2$ ,  $K3$ ,  $K4$  на рисунке 3.3б приведена та же, но преобразованная схема.

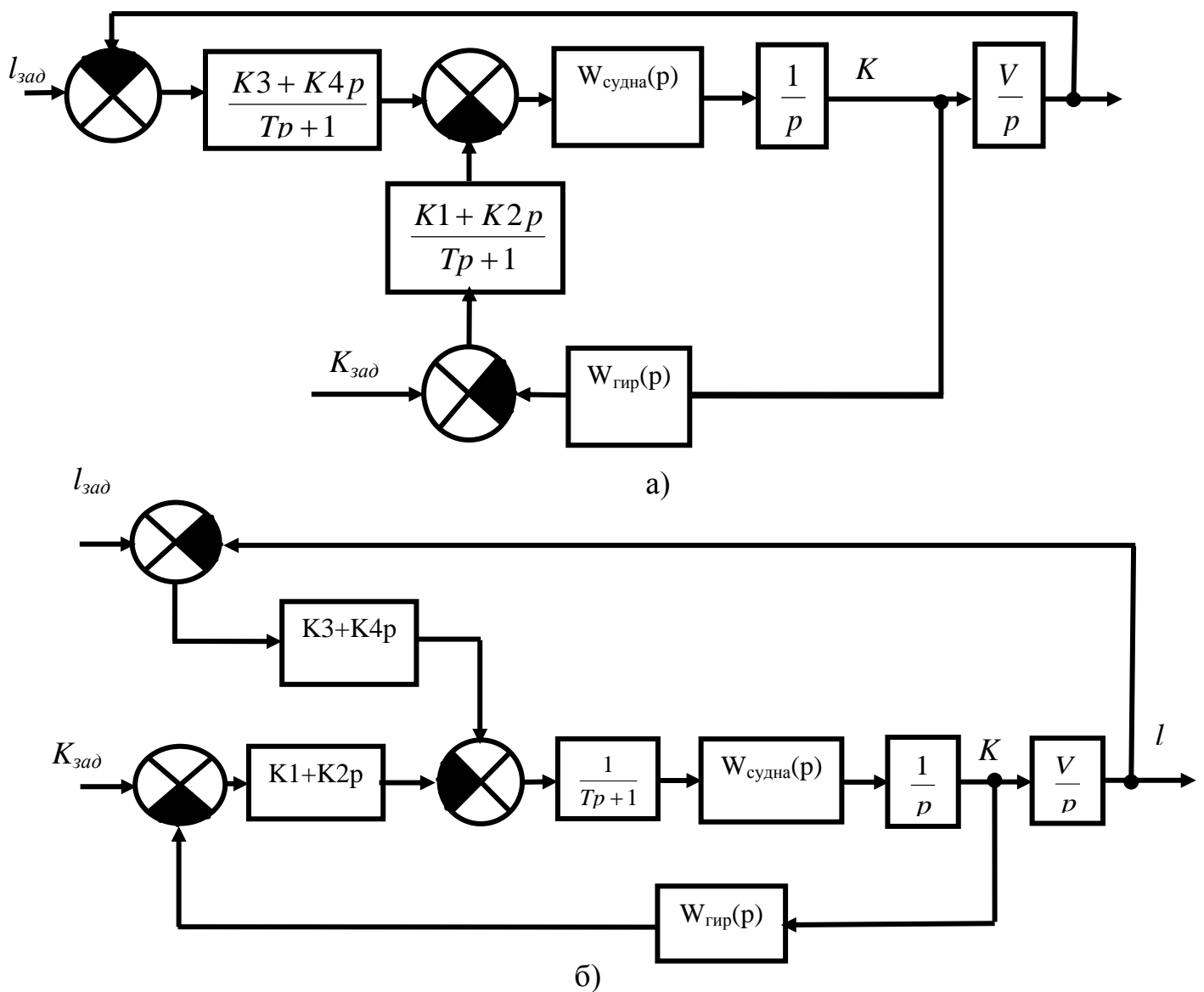
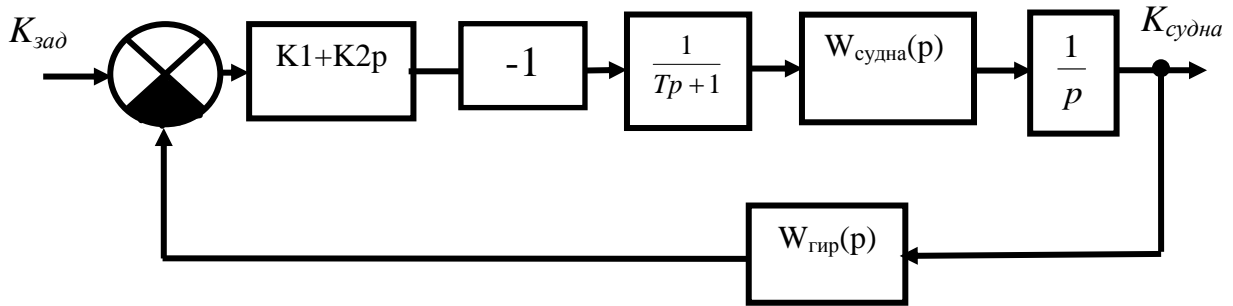
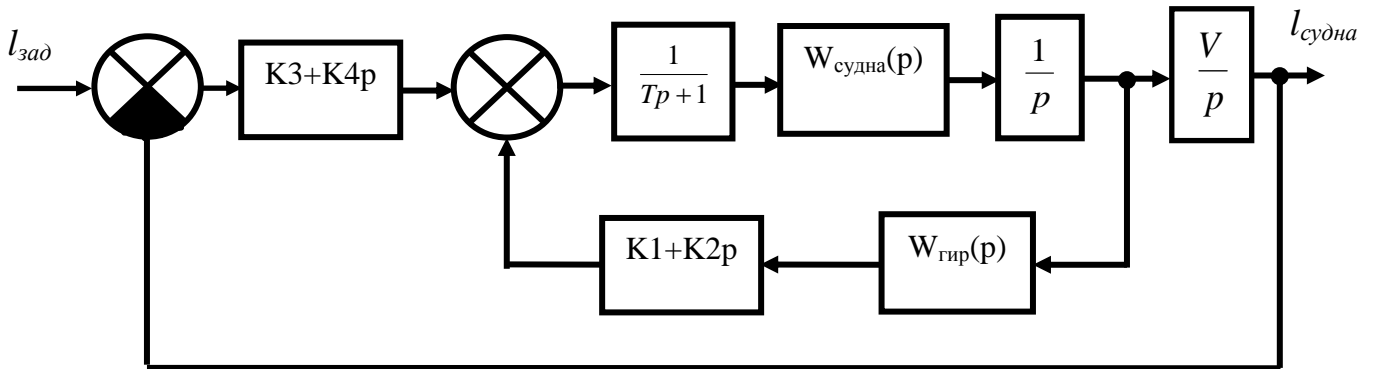


Рисунок 3.3 - Структурная схема системы управления судном по заданной траектории

Тогда структурная схема системы управления по курсу может быть представлена в виде, приведенной на рисунке 3.4а, а по боковому смещению в виде, приведенной на рисунке 3.4б.



а)



б)

Рисунок 3.4 - Структурная схема системы управления судном по заданной траектории: а) по отклонению от заданного курса  $\Delta K$ , б) по боковому смещению судна  $l_{см}$ .

Для реализации модели необходимо иметь конкретные значения коэффициентов  $K1$ ,  $K2$ ,  $K3$ ,  $K4$ , которые должны быть выбраны из области устойчивости системы.

Найдем передаточную функции системы по контуру управления курсом:

$$W_K = \frac{(58,83p+1)(K1+K2P)}{-205,36p^4 - 80,26p^3 - 11,23p^2 - p + \dots} \quad (3.19)$$

$$\dots + 0,0137pK1 + 0,00023K1 + 0,0137K2p^2 + 0,00023K2p$$

Характеристическое уравнение по контуру управления курсом имеет вид:

$$D(p) = K1(0,0137p + 0,00023) + K2(0,0137p^2 + 0,00023p) - 205,36p^4 - 80,26p^3 - 11,23p^2 - p \quad (3.20)$$

Заменяя в характеристическом уравнении  $p = j\omega$  получаем выражение для определения границ  $D$ -разбиения.

$$D(j\omega) = K1N(j\omega) + K2M(j\omega) + L(j\omega) = 0 \quad (3.21)$$

где  $N(j\omega) = N_1(\omega) + jN_2(\omega) = 0,00023 + j0,00137\omega$ ;

$$M(j\omega) = M_1(\omega) + jM_2(\omega) = -0,0137\omega^2 + j0,00023\omega$$
;

$$L(j\omega) = L_1(\omega) + jL_2(\omega) = (205,36\omega^4 + 11,23\omega^2) + j(80,26\omega^3 - \omega),$$

что приведет к следующей системе уравнений:

$$0,00023K1 - 0,0137\omega^2 K2 + 205,36\omega^4 + 11,23\omega^2 = 0;$$

$$0,00137\omega K1 + 0,00023\omega K2 + 80,26\omega^3 - \omega = 0;$$

Решая эти уравнения относительно  $K1$  и  $K2$ , получим:

$$K1 = \frac{\Delta 1}{\Delta}; \quad K2 = \frac{\Delta 2}{\Delta}; \quad (3.22)$$

где

$$\Delta = \begin{vmatrix} N_1(\omega) & M_1(\omega) \\ N_2(\omega) & M_2(\omega) \end{vmatrix} = \begin{vmatrix} 0,00023 & -0,0137\omega^2 \\ 0,00137\omega & 0,00023\omega \end{vmatrix} =$$

$$= 0,000000054\omega - 0,0001865\omega^3;$$



$$\Delta 1 = \begin{vmatrix} -L_1(\omega) & M_1(\omega) \\ -L_2(\omega) & M_2(\omega) \end{vmatrix} = \begin{vmatrix} -(205,36\omega^4 + 11,23\omega^2) & -19,4139\omega^2 \\ -(80,26\omega^3 - \omega) & 0,00023\omega \end{vmatrix} =$$

$$= 1,0486\omega^5 + 0,1106\omega^3;$$

$$\Delta 2 = \begin{vmatrix} N_1(\omega) & -L_1(\omega) \\ N_2(\omega) & -L_2(\omega) \end{vmatrix} = \begin{vmatrix} 0,00023 & -(205,36\omega^4 + 11,23\omega^2) \\ 0,00137\omega & -(80,26\omega^3 - \omega) \end{vmatrix} =$$

$$= 0,2805\omega^5 + 0,005489\omega^3 + 0,00023\omega$$

$$K1 = \frac{\Delta 1}{\Delta} = \frac{1,0486\omega^5 + 0,1106\omega^3}{0,000000054\omega - 0,0001865\omega^3};$$

$$K2 = \frac{\Delta 2}{\Delta} = \frac{0,2805\omega^5 + 0,005489\omega^3 + 0,00023\omega}{0,000000054\omega - 0,0001865\omega^3}$$

На рисунке 3.5 приведена кривая D-разбиения в области параметров  $K1$  и  $K2$ .

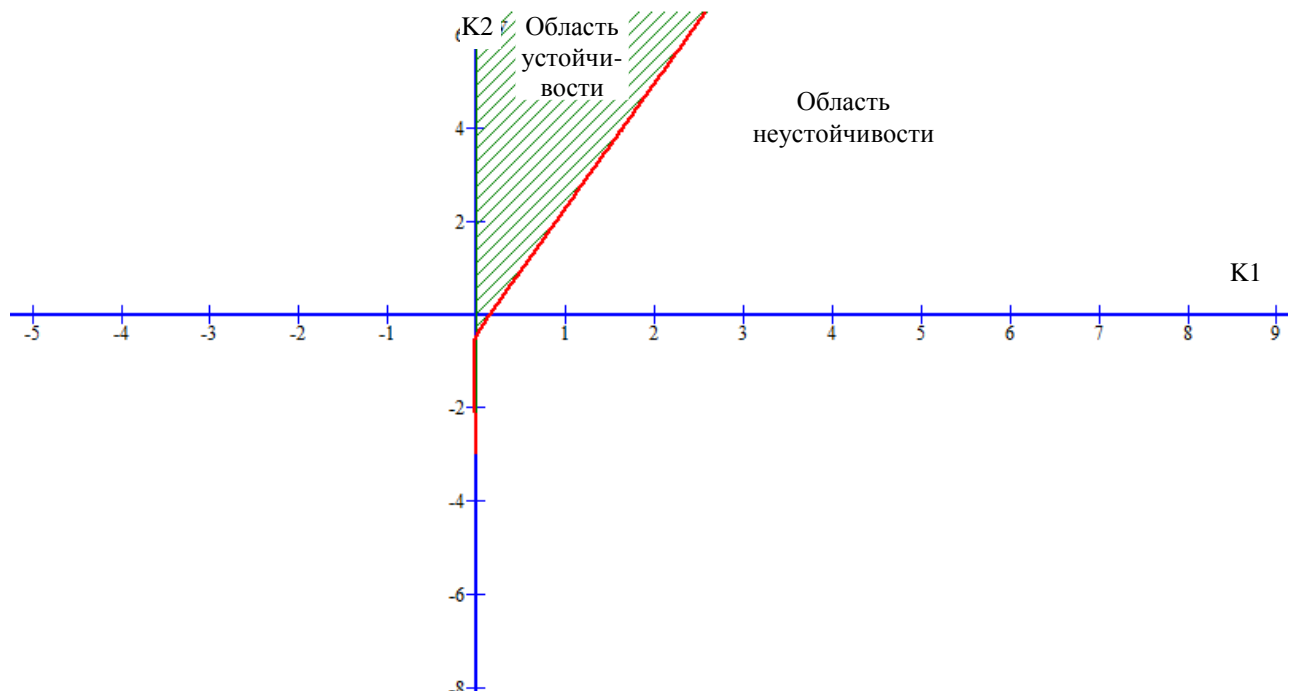


Рисунок 3.5 - Кривая D-разбиения в области параметров  $K1$  и  $K2$

Определим также область устойчивости по боковому смещению судна от заданной траектории  $l_{см}$ .

Как видно из рисунка 3.1, боковое смещение от заданной траектории может быть определено по выражению  $l_{см} = l_0 + \int_0^{\infty} V \sin \Delta K dt$ , которое для малых отклонений после линеаризации (замены  $\sin \Delta K$  на  $\Delta K$ ) в операторной форме может быть представлена в виде  $l_{см} = \frac{V}{p} \Delta K$ .

Найдем передаточную функцию системы по контуру бокового смещения.

$$W(p)_{l_{см}} = \frac{0,2117K3p + 0,2117K4p + 0,0036K3 + 0,0036K4p^2}{205,36p^5 + 80,26p^4 + 11,23p^3 - 4,6p^2 - 0,092p - 0,08538 + \dots} \quad (3.23)$$

$$\dots + 0,2117pK3 - 0,2117K4p + 0,0036K3 + 0,0036K4p^2$$

После замены  $p=j\omega$  характеристический комплекс замкнутой системы управления судном по заданной криволинейной траектории при  $K1=6$  и  $K2 = 400$  будет иметь вид:

$$D(j\omega) = K3(0,0036 + 0,2117\omega) + K4(-0,0036\omega^2 - j0,21174\omega) + (-j205,36 \omega^5 + 80,26\omega^4 - j11,23\omega^3 + 4,6\omega^2 - j0,092\omega - 0,08538). \quad (3.24)$$

$$D(j\omega) = K3N(j\omega) + K4M(j\omega) + L(j\omega) = 0 \quad (3.25)$$

где  $N(j\omega) = N_1(\omega) + jN_2(\omega) = 0,0036 + j0,2117\omega$ ;

$M(j\omega) = M_1(\omega) + jM_2(\omega) = 0,0036\omega^2 - j0,2117\omega$ ;

$L(j\omega) = L_1(\omega) + jL_2(\omega) = (80,26\omega^4 + 4,6\omega^2 - 0,08538) + j(-205,36\omega^5 - 11,23\omega^3 - 0,092\omega)$ .

Откуда получим:

$$0,0036K3 + 0,0036\omega^2 K4 + (80,26\omega^4 + 4,6\omega^2 - 0,08538) = 0$$

$$0,2117\omega K3 + (-0,2117\omega)K4 + (-205,36\omega^5 - 11,23\omega^3 - 0,092\omega) = 0$$

Решая эти уравнения относительно  $K3$  и  $K4$  получим:

$$K3 = \frac{\Delta 1}{\Delta}; \quad K4 = \frac{\Delta 2}{\Delta}; \quad (3.26)$$

где:

$$\Delta = \begin{vmatrix} N_1(\omega) & M_1(\omega) \\ N_2(\omega) & M_2(\omega) \end{vmatrix} = \begin{vmatrix} 0,0036 & 0,0036\omega^2 \\ 0,2117\omega & (0,2117\omega) \end{vmatrix} =$$

$$= -0,00076\omega - 0,00076\omega^3$$

$$\Delta 1 = \begin{vmatrix} -L_1(\omega) & M_1(\omega) \\ -L_2(\omega) & M_2(\omega) \end{vmatrix} = \begin{vmatrix} -(80,26\omega^4 + 4,6\omega^2 - 0,08538) & 0,0036\omega^2 \\ -(-205,36\omega^5 - 11,23\omega^3 - 0,092\omega) & -0,2117\omega \end{vmatrix} =$$

$$= -0,74\omega^7 + 16,95\omega^5 + 0,9734\omega^3,$$

$$\Delta 2 = \begin{vmatrix} N_1(\omega) & -L_1(\omega) \\ N_2(\omega) & -L_2(\omega) \end{vmatrix} =$$

$$= \begin{vmatrix} 0,0036 & -(-80,26\omega^4 + 4,6\omega^2 - 0,08538) \\ 0,2117\omega & -(-205,36\omega^5 - 11,23\omega^3 - 0,092\omega) \end{vmatrix} =$$

$$= -16,252\omega^5 + 1,01\omega^3 - 0,018\omega$$

$$K3 = \frac{\Delta 1}{\Delta} = \frac{-0,74\omega^7 + 16,95\omega^5 + 0,9734\omega^3}{-0,00076\omega - 0,00076\omega^3};$$

$$K4 = \frac{\Delta 2}{\Delta} = \frac{-16,252\omega^5 + 1,01\omega^3 - 0,018\omega}{-0,00076\omega - 0,00076\omega^3}$$

Кривая D-разбиения в области параметров  $K3$  и  $K4$  имеет вид приведенный на рисунке 3.6.

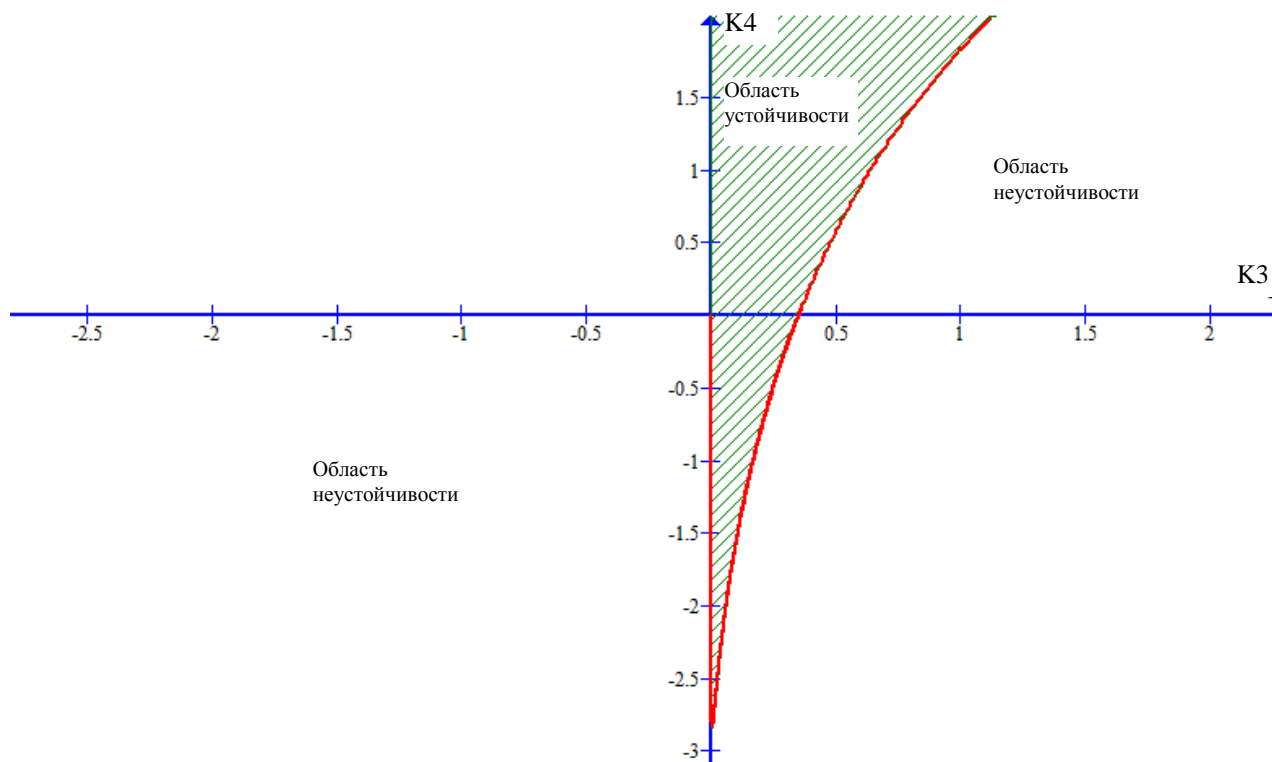


Рисунок 3.6 - Кривая D-разбиения в области параметров  $K3$  и  $K4$

### 3.4 Задание формы дна

При моделировании движения судна при неровном дне нужно будет задать параметры формы дна.

На рисунке 3.7 приведены геометрические соотношения для описания формы дна акватории плавания.

В работе форма дна задается в виде синусоидальной поверхности:

$$\text{по оси X, } \text{map [i j]} = \text{Sin}\left(\frac{3\pi}{2} + \frac{b}{3T} * j * 1.0546875\right);$$

$$\text{по оси Y, то } \text{map [i j]} = \text{Sin}\left(\frac{3\pi}{2} + \frac{b}{3T} * i * 1.0546875\right);$$

$b$  – длина оси синусоида.

$T$  – период синусоиды.

Причем:  $b = \frac{\pi}{C}$ ;  $C = \frac{T}{2}$

На рисунке 3.7  $y_0$  – смещение оси синусоиды относительно оси абсцисс,  $z$  – расстояние от оси абсцисс до поверхности воды

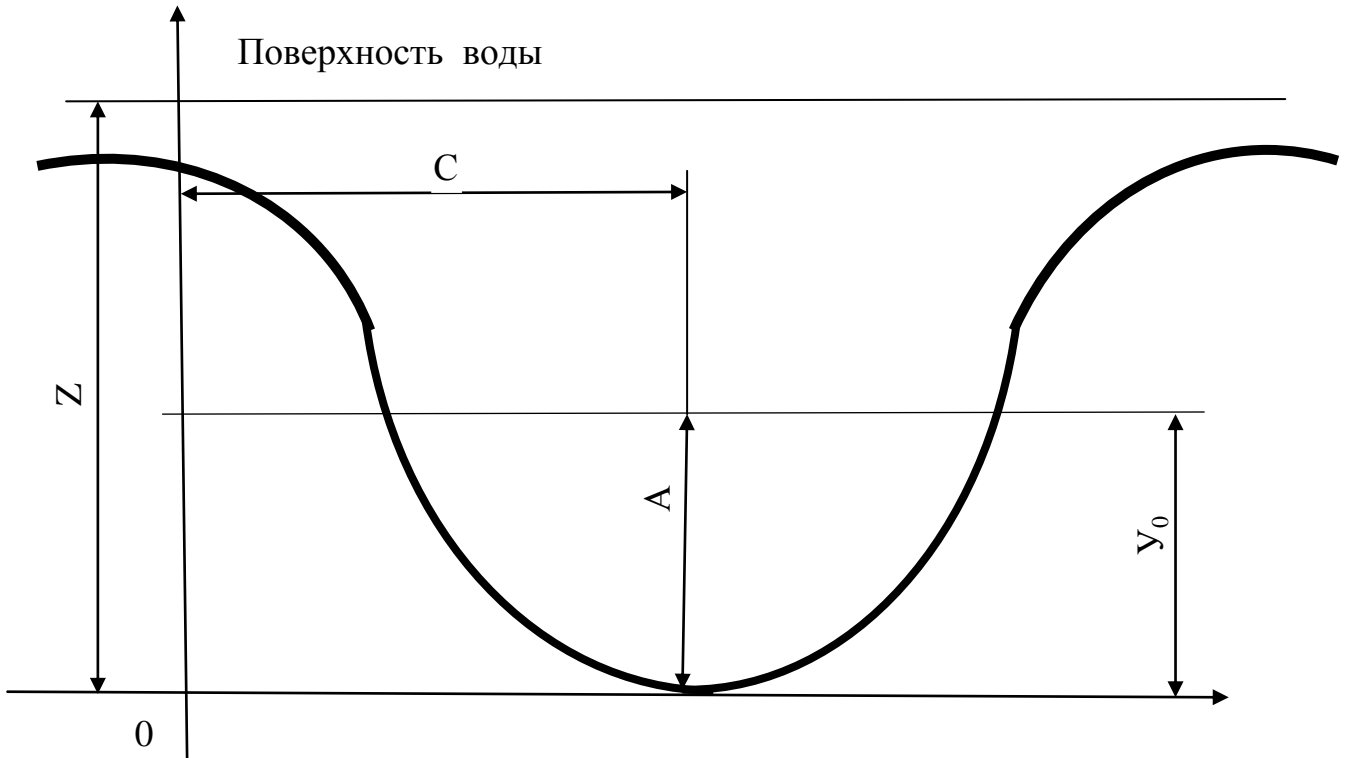


Рисунок 3.7 - Геометрические соотношения для описания формы дна

Ось абсцисс проходит через минимумы синусоиды.

Форма дна может быть задана также в виде матрицы, где каждому элементу матрицы соответствует определенная глубина. Для примера на рисунке 3.8 приведено задание формы дна для матрицы размером  $M[3, 3]$ :

$$M = \begin{vmatrix} 4 & 7 & 4 \\ 4 & 8 & 5 \\ 6 & 4 & 5 \end{vmatrix}$$

В программе для преобразования определенного участка дна в элемент матрицы применяют маску  $M$ . В работе, для задания формы дна, применена матрица размерности  $A[5,5]$ . Вся рассматриваемая площадь 5000м x 5000м.

Маску применяют для масштабирования всей площади.

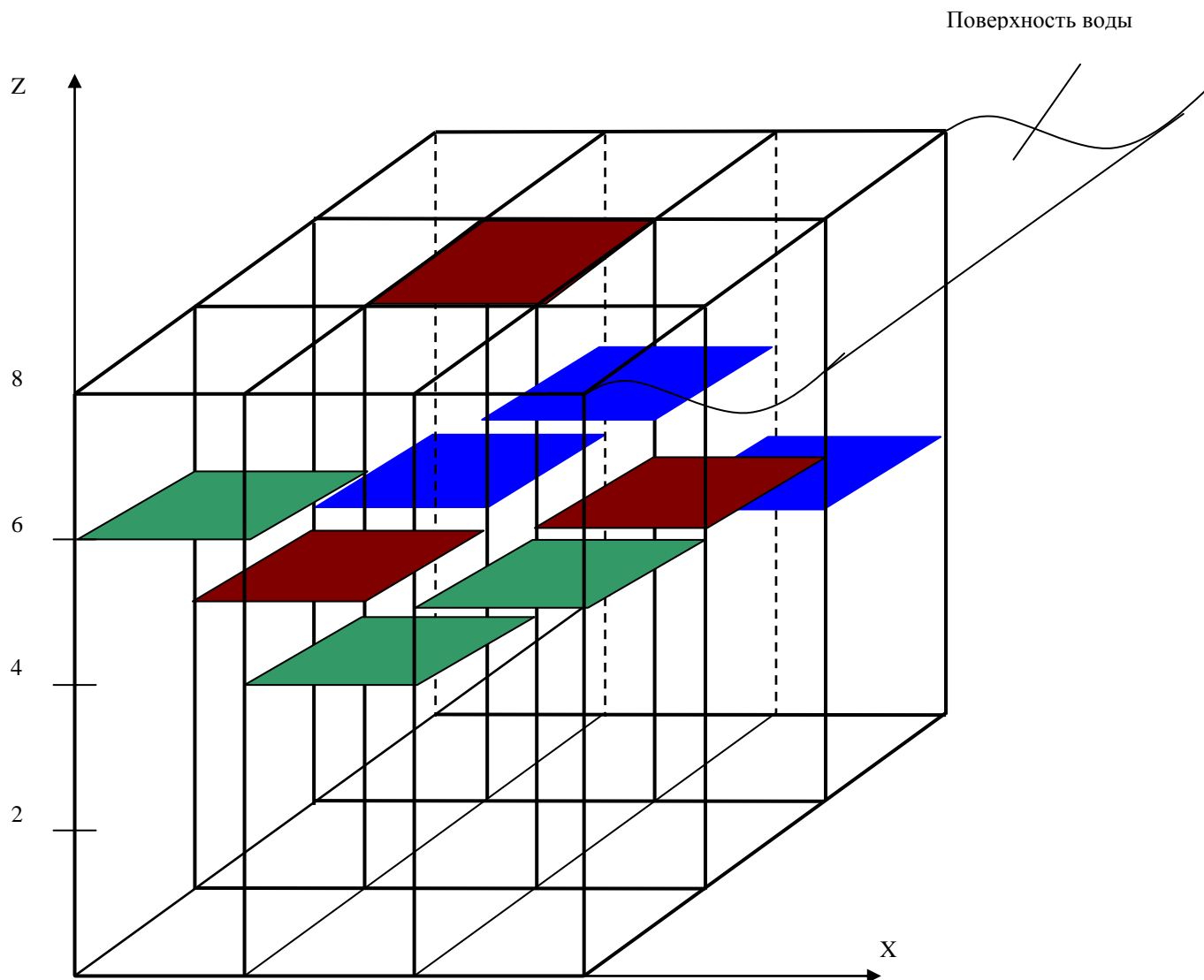


Рисунок 3.8 - Задание формы дна в виде матрицы  $M[3,3]$

На рисунке 3.8 по оси  $Z$  откладываются дискретные значения глубины  $H$ .

В диссертационной работе  $M=1024$ , т.е. если  $k=1024/5 \approx 200$  – значение для масштабирования значения точек дна по горизонтали,  $I=1024/5 \approx 200$  – значение для масштабирования значений точек дна по ветиркали, то в нашем случае получается, что одному элементу матрицы соответствует площадь дна 200м x 200 м. В результате такая матрица покроет всю поверхность дна.

Для увеличения детализации дна достаточно увеличить размер матрицы. Чем больше размер матрицы, тем меньше будут площади, на которые делится площадь дна.

### **Выводы по третьей главе**

1. Определены допущения и ограничения, принимаемые при моделировании движения судна на мелководье по заданной траектории.
2. Разработана математическая модель движения судна на мелководье при переменной глубине акватории.
3. Параметры авторулевого выбраны методом D - разбиения из области устойчивости системы.
4. Разработана математическая модель движение судна на мелководье по заданной криволинейной траектории при постоянной и переменной глубинах.

## ГЛАВА 4. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ МАТЕМАТИЧЕСКОЙ МОДЕЛИ ДВИЖЕНИЯ СУДНА НА МЕЛКОВОДЬЕ

### 4.1 Общая характеристика программного комплекса

Разработанная в работе программа позволяет визуально наблюдать за движением модели судна на мелководье. Программа функционирует на IBM-PC – совместимом компьютере стандартной конфигурации со средой Windows и может быть использована для исследования движения судна на мелководье на его модели.

В программном пакете предусмотрена возможность ввода параметров различных судов, имеется возможность визуального наблюдения за траекторией движения судна при различных скоростях и углах кладки руля.

Решение системы дифференциальных уравнений осуществлено с применением метода Рунге-Кутты.

На рисунке 4.1 приведена укрупненная блок-схема алгоритма моделирования движения судна на мелководье.

Блок 1 обеспечивает увеличение текущего времени при каждом шаге решения дифференциальных уравнений на величину  $\Delta t$ .

Блок 2 предназначен для решения размерных уравнений динамики судна, а блок 4 осуществляет расчет координат местоположения и курса судна.

Блок 5 определяет глубину на текущем местоположении судна и производит перерасчет коэффициентов при уравнениях гидродинамики судна для текущей глубины.

Логический блок 6 предназначен для завершения расчетов после поворота судна вокруг вертикальной оси на  $360^0$ , т.е. после завершения циркуляции, а логический блок 7 предназначен для завершения расчетов по требованию оператора.

Решающий блок 3 производит операцию присваивания для увеличения времени на  $\Delta t$  для выполнения вычислений на следующем шаге.



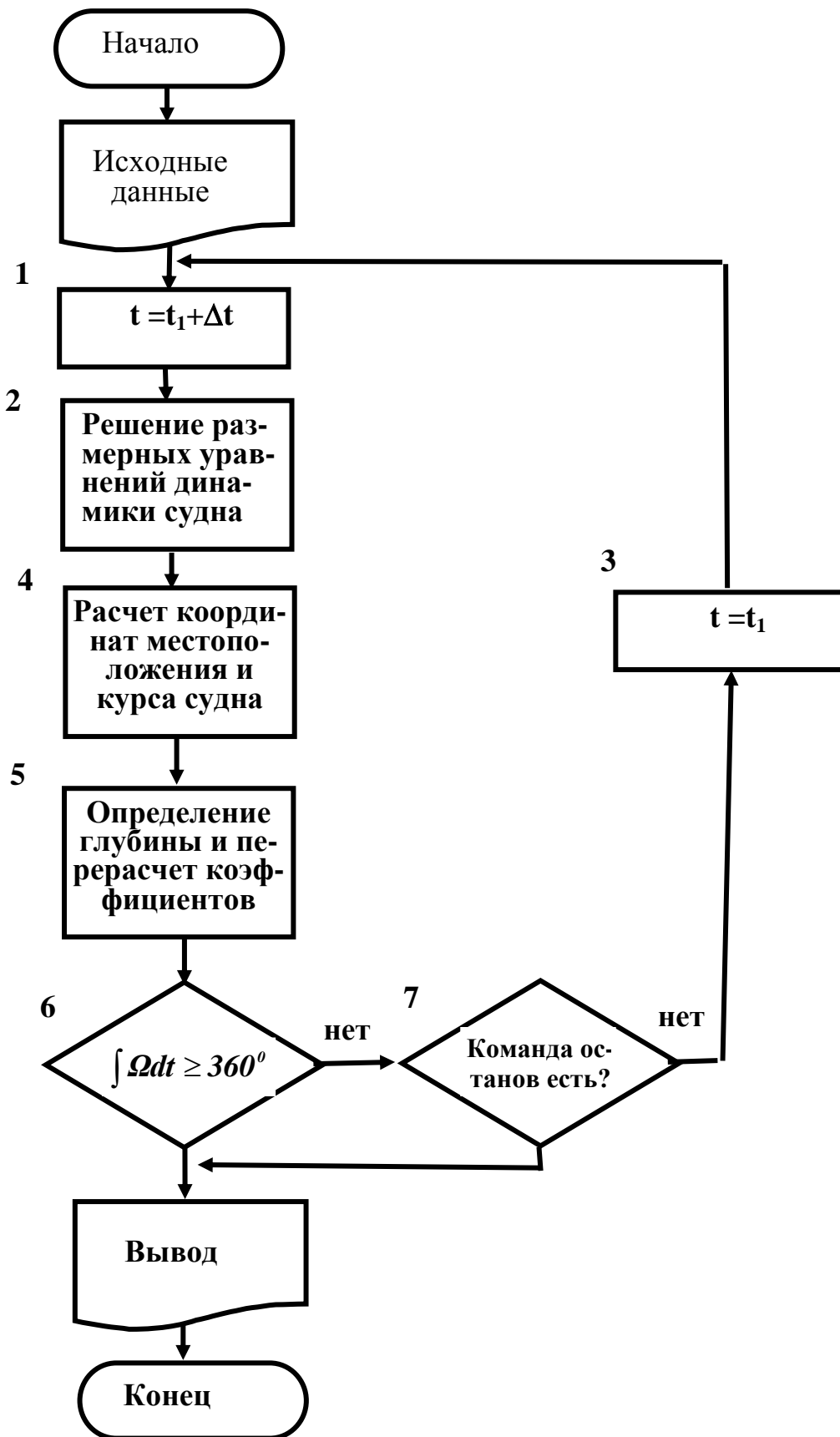


Рисунок 4.1 - Блок-схемы алгоритма моделирования циркуляции судна на мелководье

Назначение остальных блоков понятно из их графического обозначения.

## 4.2 Выбор среды разработки

При разработки программного комплекса использовался инструмент Unity с использованием языка C#.

На рисунке 4.2 приведено дерево программ, используемых для моделирования движения судна на мелководье.

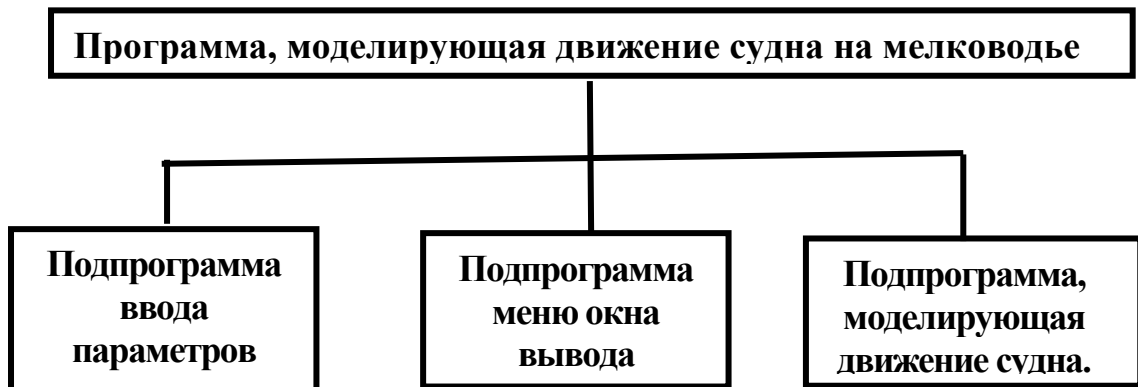


Рисунок 4.2 - Дерево программ, используемых при моделировании движения судна на мелководье

**Подпрограмма ввода параметров судна** (приложение Б) используется для ввода параметров судна. Параметры разнесены по трем вкладкам. На рисунке 4.3 приведена экранная форма (окно) для ввода параметров судна.

Кнопка **Сохранить** позволяет сохранить введенные параметры. Данные сохраняются в xml файл: **muxmx.xml**.

Кнопка **По умолчанию** позволяет вернуть значения по умолчанию, значения берутся из xml файла: **default.xml**.

$V_0$	5.144	$L$	100	$\rho$	1
$\lambda_{11}$	109.457	$l_\epsilon$	47	$\rho_z^2$	25
$\lambda_{22}$	2476.755	$l_p$	48	$\sigma$	0.971
$\lambda_{66}$	1338600	$m_y^\beta$	0.084	$\sigma_p$	0.289
$B$	17	$m_y^\omega$	-0.026	$\sum_\epsilon$	2
$C_b$	0.64	$n_{x0}$	0.009	$T$	4.6
$\chi_\beta$	0.3	$n_y^\beta$	0.249	$W$	5004.8
$\chi_{\beta\epsilon}$	0.3	$n_{y\epsilon}^\beta$	0.011	$F_{p+pn}$	10
$\chi_\omega$	0.5	$n_y^\delta$	0.035	$F'_{p+pn}$	0
$\chi_{\omega\epsilon}$	0.85	$n_y^\omega$	0.014		

Сохранить      По умолчанию

Рисунок 4.3 - Экранная форма для ввода параметров судна

**Подпрограмма моделирования циркуляции судна** (Приложение В, Приложение Г)

При запуске подпрограммы (Приложение В) открывается окно, где можно выбрать разрешение экрана, качество графики.

При нажатии кнопки **Play** параметры судна считываются из xml файла: **myxml.xml**.

После считывания файла открывается окно программы, у которого в правом верхнем углу отображается окно **Menu**, ввода начальных условий для запуска программы (рисунок 4.4)

В данном окне можно задать:

- 1) координаты [X;Y] судна, в пределах от -450 до 1500 метров;
- 2) курс судна в пределах от 0 до 360 градусов;
- 3) угол перекладки руля в пределах от -40 до 40 градусов;

- 4) глубину под килем судна от 5 до 500 метров;
- 5) время моделирования от 0,001 до 0,6;
- 6) параметры дна;
- 7) время наблюдения;
- 8) начальную скорость судна;
- 9) параметры дна;



Рисунок 4.4 - Меню ввода начальных условий для запуска программы при циркуляции

- 10) кнопка **Ровное дно** для генерации плоского морского дна и запуска движения судна;
- 11) кнопка **Кривое дно** для генерации рельефного дна и запуска движения судна;
- 12) кнопка **Резкие перепады** для генерации рельефного дна и запуска движения судна.

При пуске хода судна по п.10 на экране дисплея отображается траектория движения судна при плоском морском дне (Приложение Г )

При пуске хода судна по п.11 на экране дисплея отображается траектория движения судна на мелководье при неровном дне (Приложение Г ).

В левом верхнем углу окна программы отображается значок **бегунок**, двигая его влево или вправо можно увеличить или уменьшить масштаб соответственно.

В левом верхнем углу ( под значком **бегунок** ) окна программы отображается тип морского дна

Вверху, в центре, отображается глубина под килем судна (Н), текущие координаты судна (X,Y), курс судна (К) и время (Т).

В данной подпрограмме при нажатии клавиши:

**R** – судно стартует при задании начальных значений координат, курса и угла кладки руля.

**F** - камера «следит» за судном, т.е. движется за судном.

**ESC** – выводит окно **Menu**, прерывая процесс движения судна. Повторное нажатие закрывает окно, и судно продолжает движение.

Задание и отображение рельефа дна осуществляется с помощью двумерного массива, заполнение которого осуществляется по исходной алгебраической функции, которая задается в программе *menuSelectScene* в подпрограмме рисования кривого дна, *terrainHeights[i,j]*.

На рисунке 4.5 приведена экранная форма задания рельефа дна.

### **4.3 Разработка блок-схемы алгоритма моделирования движения судна на мелководье по заданной криволинейной траектории**

На рисунке 4.6 приведена блок-схема алгоритма управления судном по криволинейной траектории при переменной глубине.

Здесь блок 1 обеспечивает увеличение текущего времени с каждым шагом решения дифференциальных уравнений на величину  $\Delta t$ .

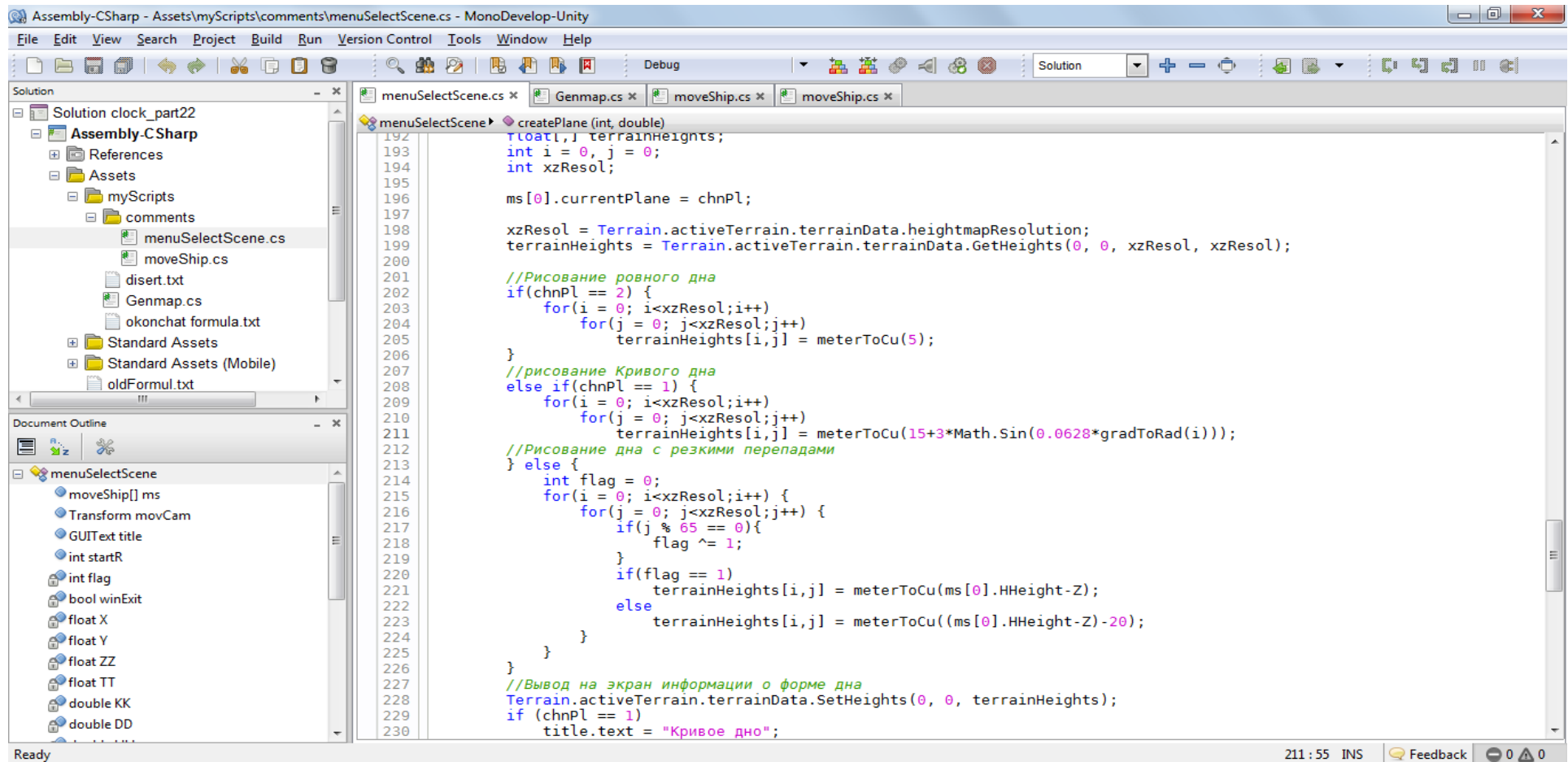


Рисунок 4.5 - Экранная форма задания рельефа дна

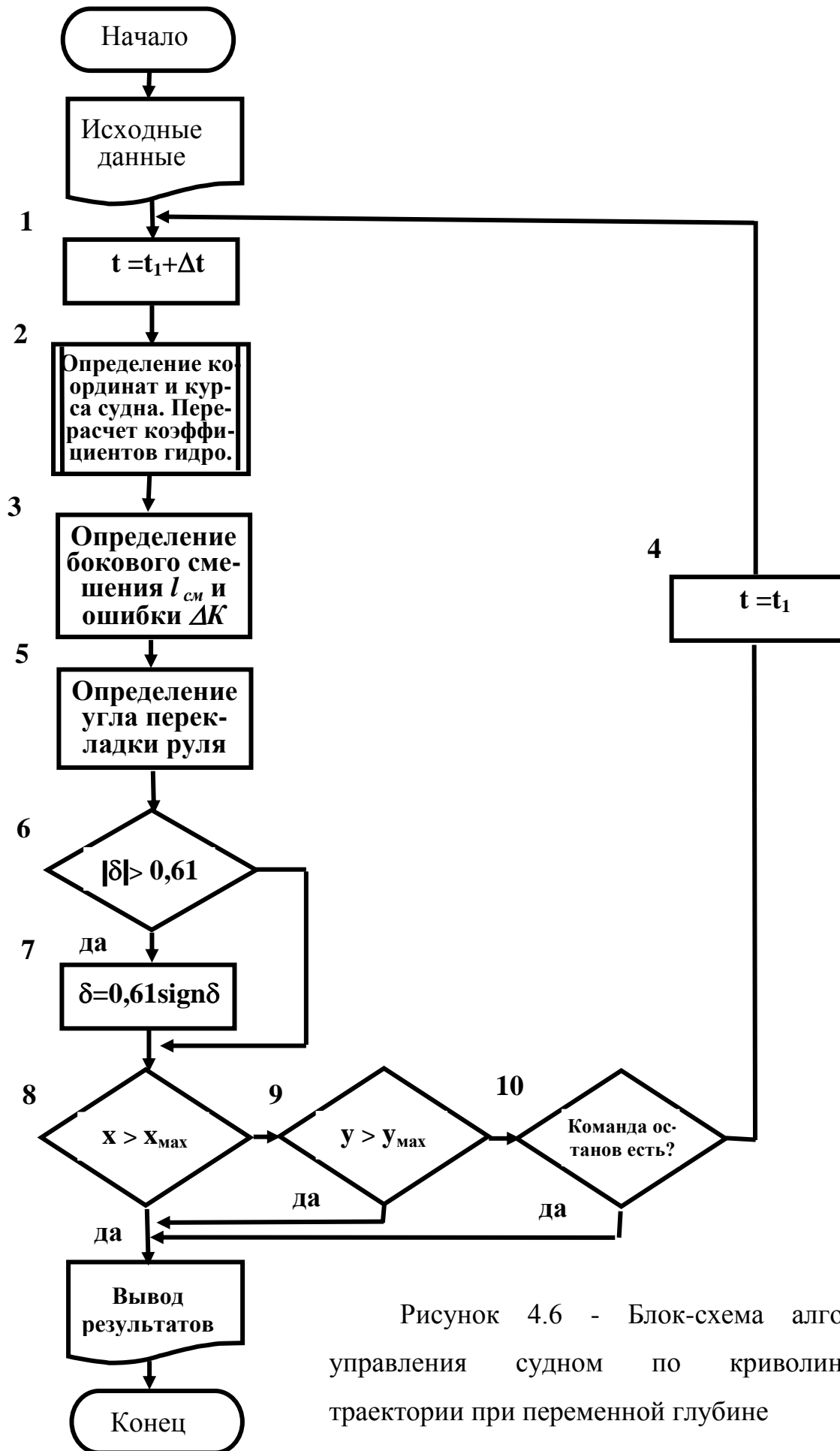


Рисунок 4.6 - Блок-схема алгоритма управления судном по криволинейной траектории при переменной глубине

Блок 2 осуществляет определение координат и курса судна, а также перерасчет коэффициентов гидродинамики судна в соответствии с алгоритмом, описанным в главе 3.

Блок 3 предназначен для определения бокового смещения  $l_{см}$  и ошибки по курсу  $\Delta K$ .

Блок 5 по известному закону регулирования определяет угол перекладки руля.

Ограничения на угол перекладки руля  $|\delta| < 0,61$  реализуются блоками 6 и 7.

Логические блоки 8 и 9 завершают расчет при достижении судном одного из конечных координат  $x_{max}$  или  $y_{max}$

Логический блок 10 предназначен для завершения расчетов по требованию оператора.

Решающий блок 4 производит операцию присваивания для увеличения времени на  $\Delta t$  для выполнения вычислений на следующем шаге.

Назначение остальных блоков понятно из их графического обозначения.

#### 4.4 Разработка программы, моделирующей движение судна на мелководье по криволинейной траектории

На рис. 4.7 - приведено дерево программ, используемых для моделирования движения судна на мелководье по криволинейной траектории.

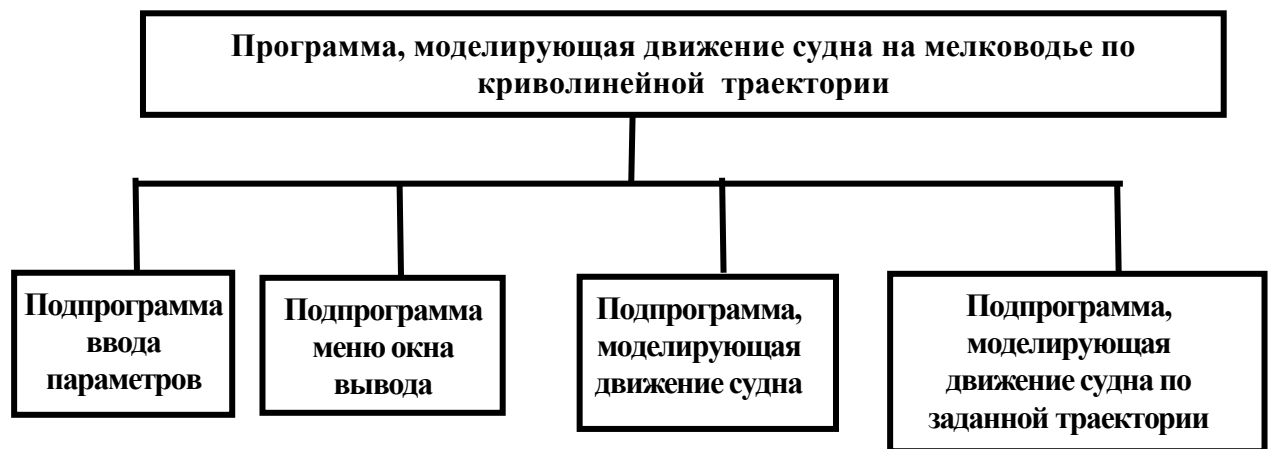


Рисунок 4.7 - Дерево программ, используемых при моделировании движения судна на мелководье по криволинейной траектории



В качестве подпрограммы ввода параметров судна при управлении по заданной криволинейной траектории используются экранные формы приведенные на рисунках 4.3. После запуска подпрограммы открывается окно, где можно выбрать разрешение экрана, качество графики.

При нажатии кнопки **Play** параметры судна считываются из xml файла: **myxml.xml**.

После считывания файла открывается окно программы, у которого в правом верхнем углу отображается окно **Menu**, ввода начальных условий для запуска программы (Рисунок 4.8) (Приложение Д).

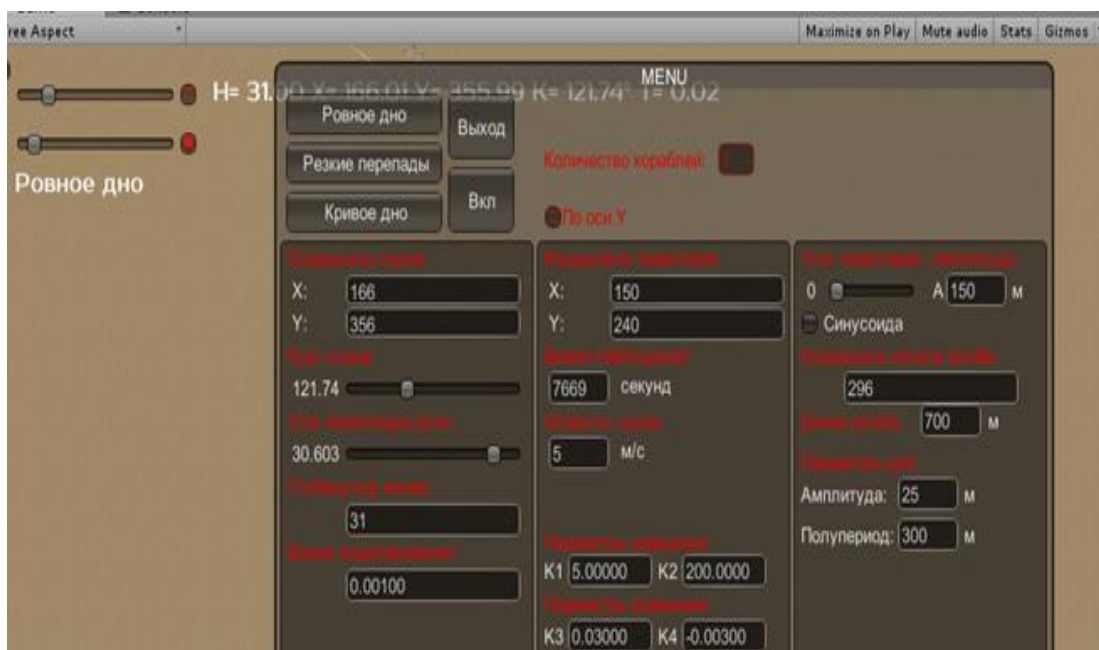


Рисунок 4.8 - Меню ввода начальных условий для запуска программы движения судна на мелководье по заданной траектории

Назначение одинаково названных на рисунках 4.7 и 4.8 органов управления и порядок пользования ими аналогичны.

В данном окне можно задать:

- 1) координаты [X;Y] судна, в пределах от -450 до 1500 метров;
- 2) курс судна в пределах от 0 до 360 градусов;
- 3) угол перекадки руля в пределах от -40 до 40 градусов;
- 4) глубину под килем судна от 5 до 500 метров;
- 5) время моделирования от 0,001 до 0,6;

- 6) координаты [X;Y] заданной траектории, в пределах от 0 до 1500 метров;
- 7) время наблюдения;
- 8) начальную скорость судна;
- 9) коэффициенты K1, K2, K3, K4 регулятора;
- 10) параметры дна;
- 11) кнопка **Ровное дно** для генерации плоского морского дна и запуска движения судна;
- 12) кнопка **Кривое дно** для генерации рельефного дна и запуска движения судна;
- 13) кнопка **Резкие перепады** для генерации рельефного дна и запуска движения судна.

При пуске хода судна по п.11 на экране дисплея отображается траектория движения судна при плоском морском дне.(Приложение E)

При пуске хода судна по п.12 на экране дисплея отображается траектория движения судна на мелководье по криволинейной траектории при кривом дне. (Приложение E)

#### **4.5 Анализ результатов моделирования движения судна по заданной траектории**

На рисунках 4.9 и 4.10 приведены результаты моделирования циркуляции судна при различных параметрах неровности дна, которая задается в виде синусоидальной поверхности [5,13, 14, 15,16].

Из рисунков 4.9 и 4.10 видно, что в отличие от циркуляций на глубокой воде, на мелководье при переменной глубине кривая циркуляции имеет совершенно другой вид.

В этом случае траектория движения судна приобретает вид циклоиды.

На рисунках, для наглядности, более глубоким участкам соответствует более темный фон.

В программе, при движении судна на мелководье с переменной глубиной, перед каждым шагом вычислений определяется глубина, для которой

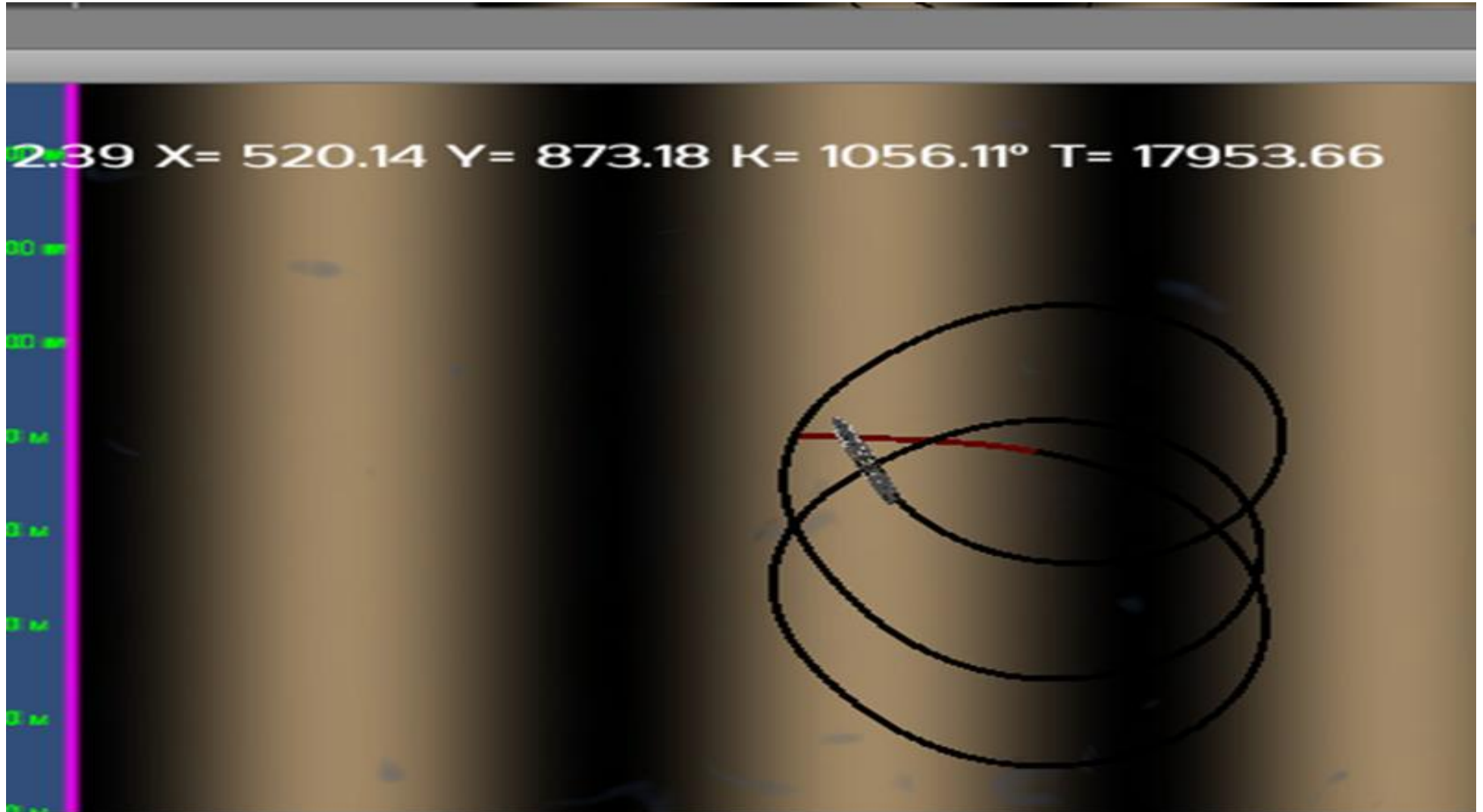


Рисунок 4.9 – Результаты моделирования циркуляции судна при неровном дне при  $H=27,330\text{ м.}$ ,  $X=474\text{ м.}$ ,  $Y=899\text{ м.}$ ,  $K=91,9830$ ,  $\delta=30.4570$ ,  $h=0.001$ ,  $V=5\text{ м/с}$ . Параметры дна: амплитуда  $25\text{ м.}$ , полупериод  $300\text{ м.}$

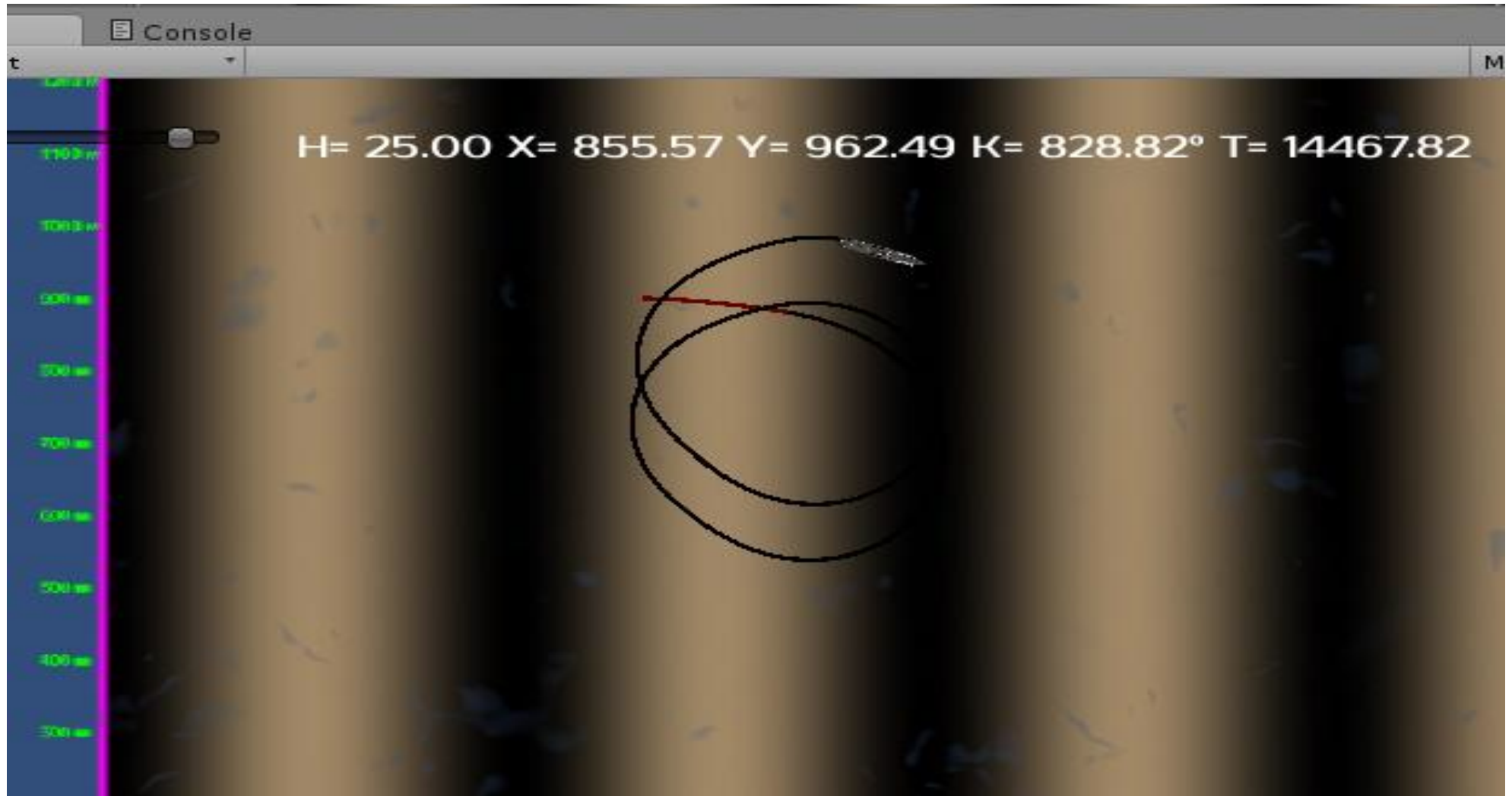


Рисунок 4.10 – Результаты моделирования циркуляции судна при неровном дне при  $H=27,330$ м.,  $X=474$  м.,  $Y=899$ м.,  $K=91,9830$ ,  $\delta=30.4570$ ,  $h=0.001$ ,  $V=5$ м/с. Параметры дна: амплитуда 25м., полупериод 350м.

производится перерасчет коэффициентов в уравнениях гидродинамики судна. Затем по этим уравнениям определяются координаты положения судна. Данная операция повторяется на каждом шаге проводимых вычислений.

На рисунке 4.11 приведены экранные формы движение судна при ровном дне по заданной траектории, при этом скорость движения судна  $V=5\text{м/с}$ , шаг моделирования равен 0,04.

Самый большой шаг моделирования, которая может быть задана в программе  $h=0.6$ .

На рисунке 4.12 приведена экранная форма движение судна при ровном дне по заданной траектории.

Анализ переходных процессов рисунков 4.11 и 4.12 показывает, что качественно они мало отличаются друг от друга. Поэтому для качественной оценки переходных процессов с меньшими затратами машинного времени для моделирования может быть использован шаг  $h=0.04$  сек. Однако для количественной оценки переходных процессов использование большого шага моделирования недопустимо.

На рисунке 4.13 приведена экранная форма движение судна при неровном дне по заданной траектории.

Анализ рисунков 4.12 и 4.13 показывает, что система является склонной к колебательности, т. е имеет место большая интегральная квадратичная ошибка регулирования. Из теории управления известно, что для уменьшения ошибки регулирования необходимо увеличить общий коэффициент усиления системы. Изменим коэффициент  $K_1$  при составляющем курса. Вместо  $K_1=5$ , возьмем  $K_1=10$ . Результаты моделирования, при  $K_1=10$ , приведены на рисунке 4.14 (шаг моделирования  $h=0,6$ ) и на рисунке 4.15 (шаг моделирования  $h=0,001$ )

Как видно из рисунка 4.15, интегральная ошибка и ошибка регулирования, с увеличением коэффициента усиления  $K_1$  значительно уменьшились.

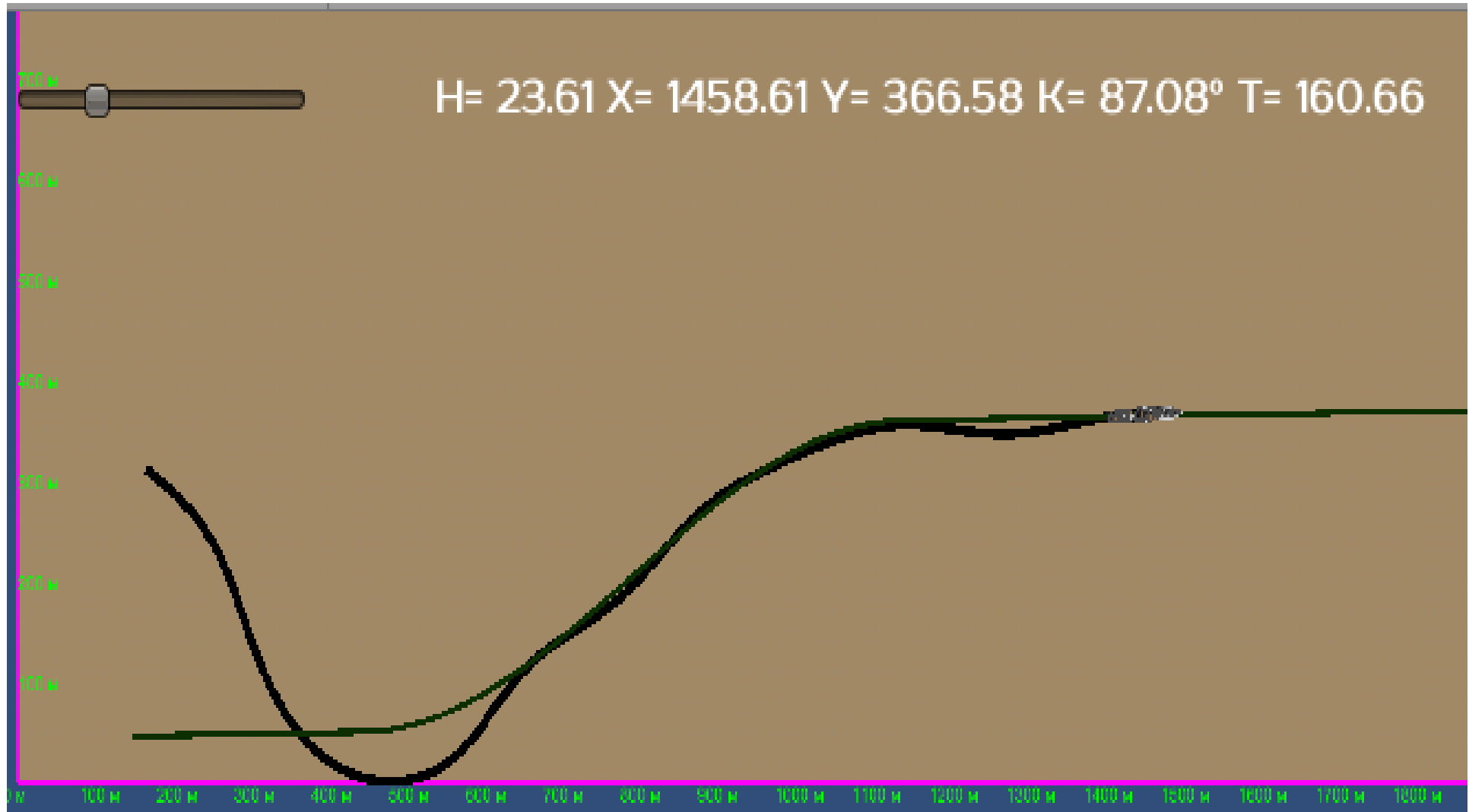


Рисунок 4.11 - Движение судна при ровном дне по заданной траектории:  $H=23.609$  м.,  $X=166$  м.,  $Y=312$  м.,  $K=121.74^0$ ,  $\delta=31.206^0$ ,  $h=0.04$ ,  $V=5$  м/с.,  $K_1=5$ ,  $K_2=400$ ,  $K_3=0.03$ ,  $K_4=-0.003$

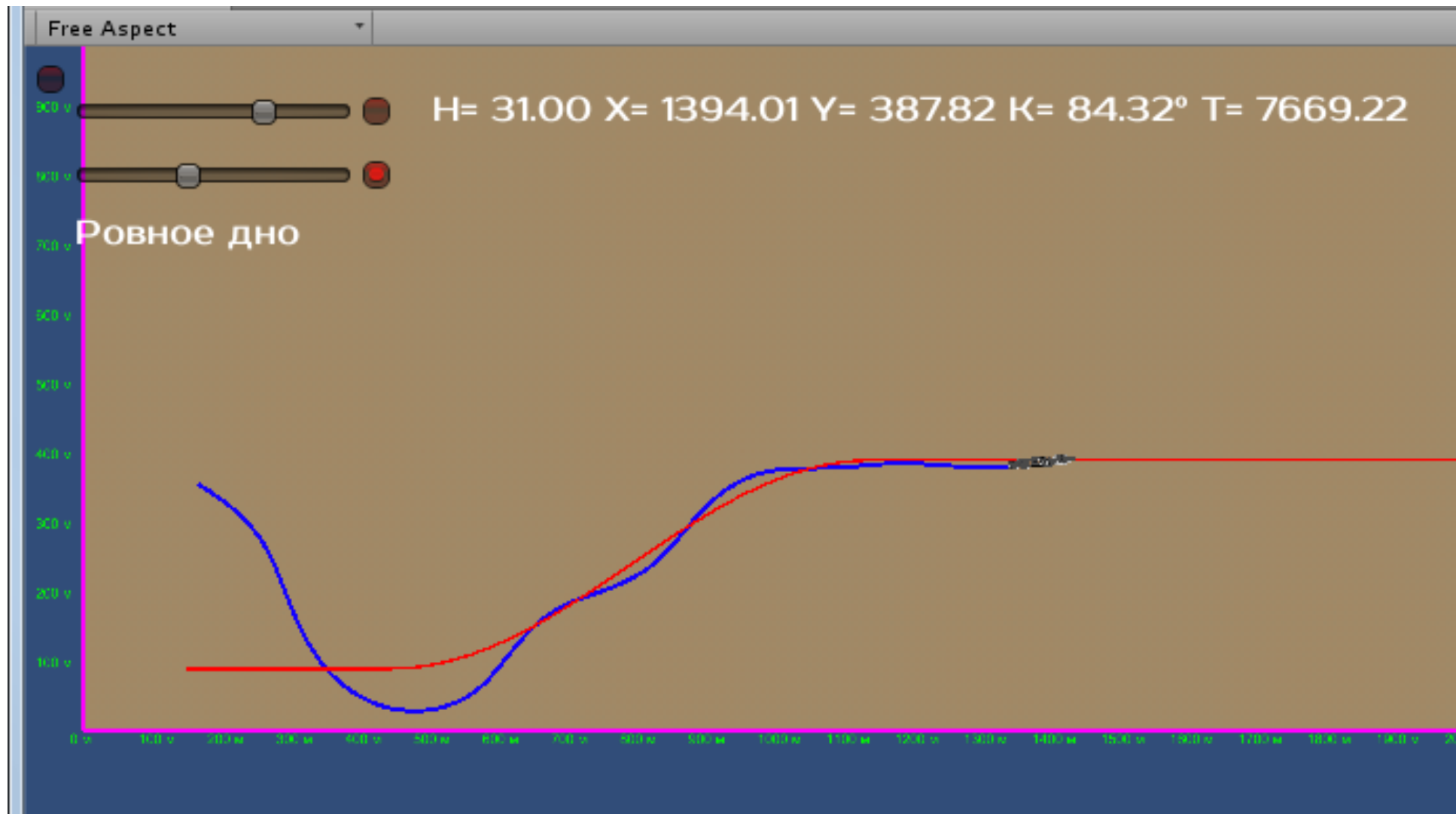


Рисунок 4.12 - Движение судна при ровном дне по заданной траектории:  $H=31$  м.,  $X=166$  м.,  $Y=356$  м.,  $K=121.74^{\circ}$ ,  
 $\delta=30^{\circ}$ ,  $h=0.001$ ,  $V=5$  м/с.,  $K_1=5$ ,  $K_2=400$ ,  $K_3=0.03$ ,  $K_4=-0.003$

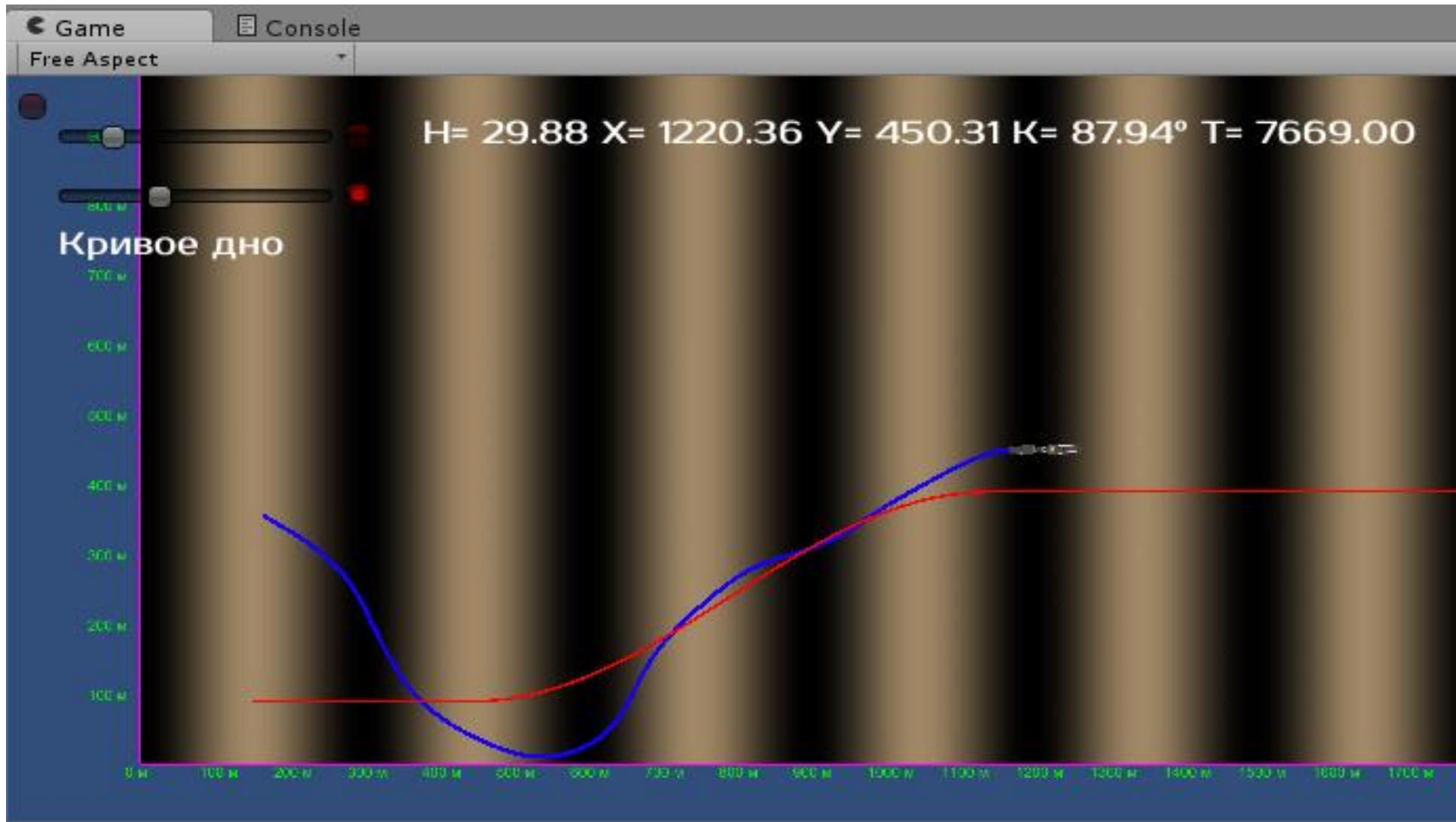


Рисунок 4.13 - Движение судна по заданной траектории при неровном дне:  $H=31$  м.,  $X=166$  м.,  $Y=356$  м.,  
 $K=121.74^\circ$ ,  $\delta=30^\circ$ ,  $h=0.001$ ,  $V=5$  м/с.,  $K_1=5$ ,  $K_2=400$ ,  $K_3=0.03$ ,  $K_4=-0.003$



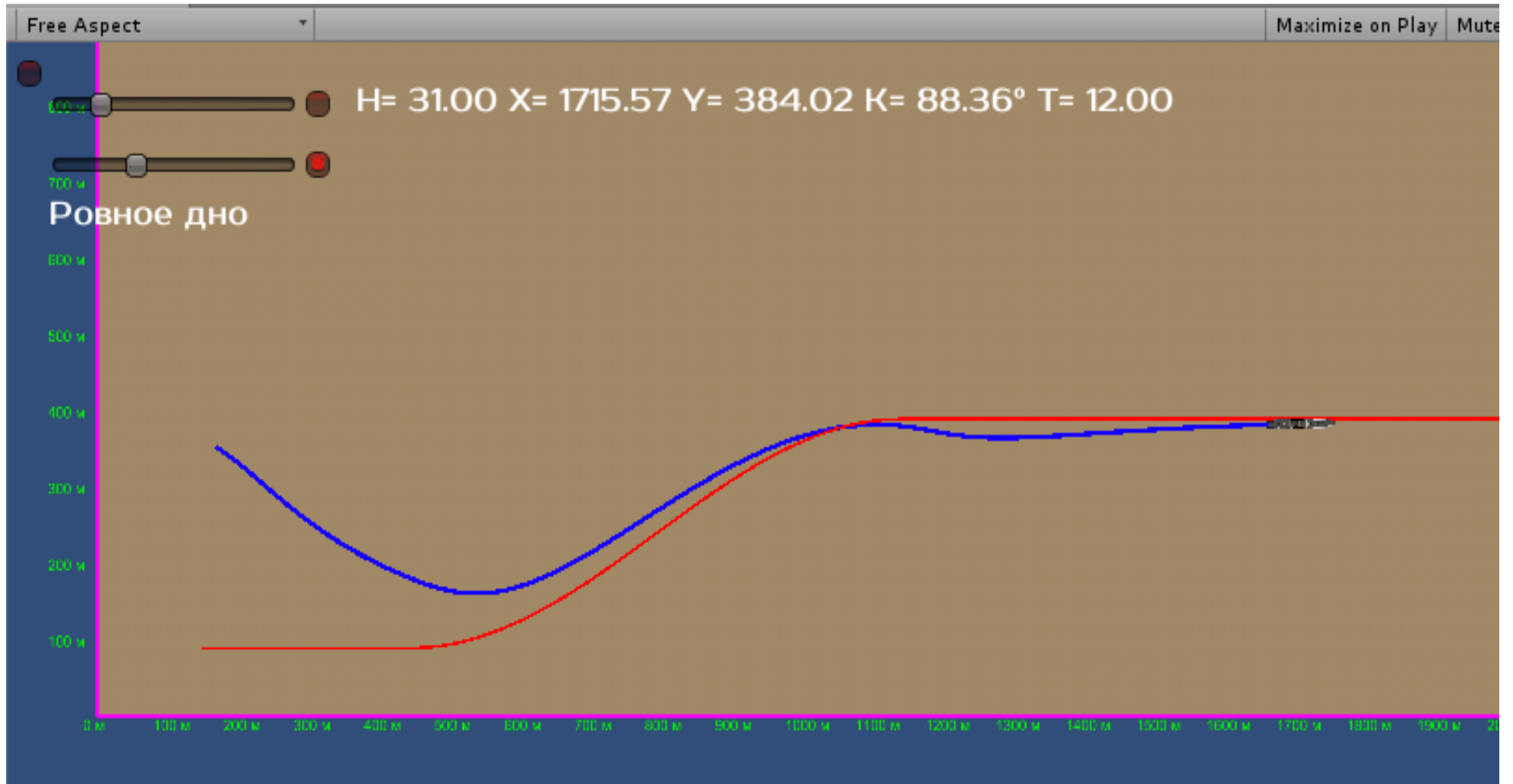


Рисунок 4.14 - Движение судна при ровном дне по заданной траектории:  $H=31$  м.,  $X=166$  м.,  $Y=356$  м.,  $K=121.74^\circ$ ,  
 $\delta=30^\circ$ ,  $h=0.6$ ,  $V=5$  м/с., при  $K_1=10$ ,  $K_2=400$ ,  $K_3=0.03$ ,  $K_4=-0.003$

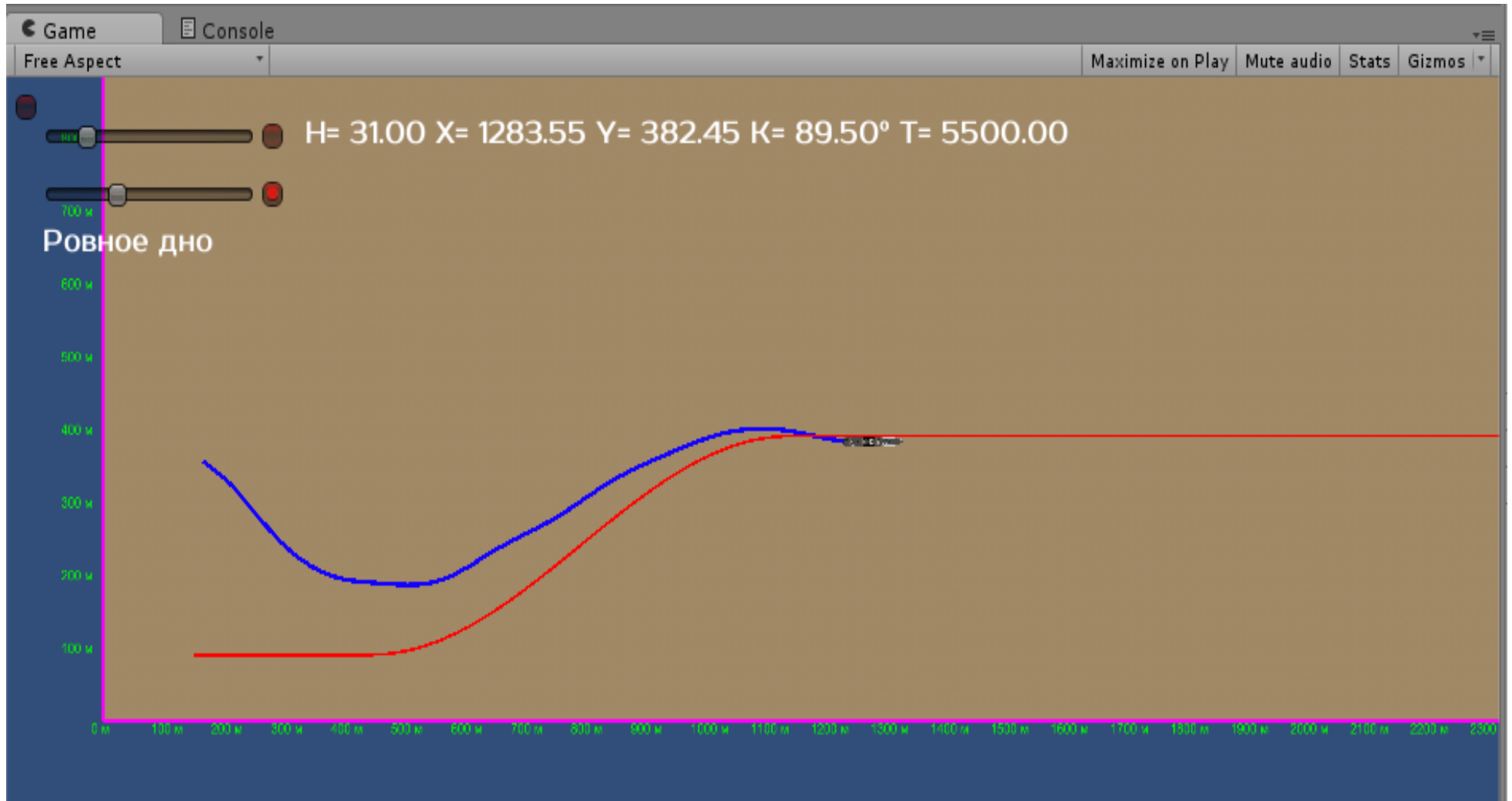


Рисунок 4.15 - Движение судна при ровном дне по заданной траектории:  $H=31$  м.,  $X=166$  м.,  $Y=356$  м.,  $K=121.74^{\circ}$ ,  $\delta=30^{\circ}$ ,  $h=0.001$ ,  $V=5$  м/с., при  $K_1=10$ ,  $K_2=400$ ,  $K_3=0.03$ ,  $K_4=-0.003$

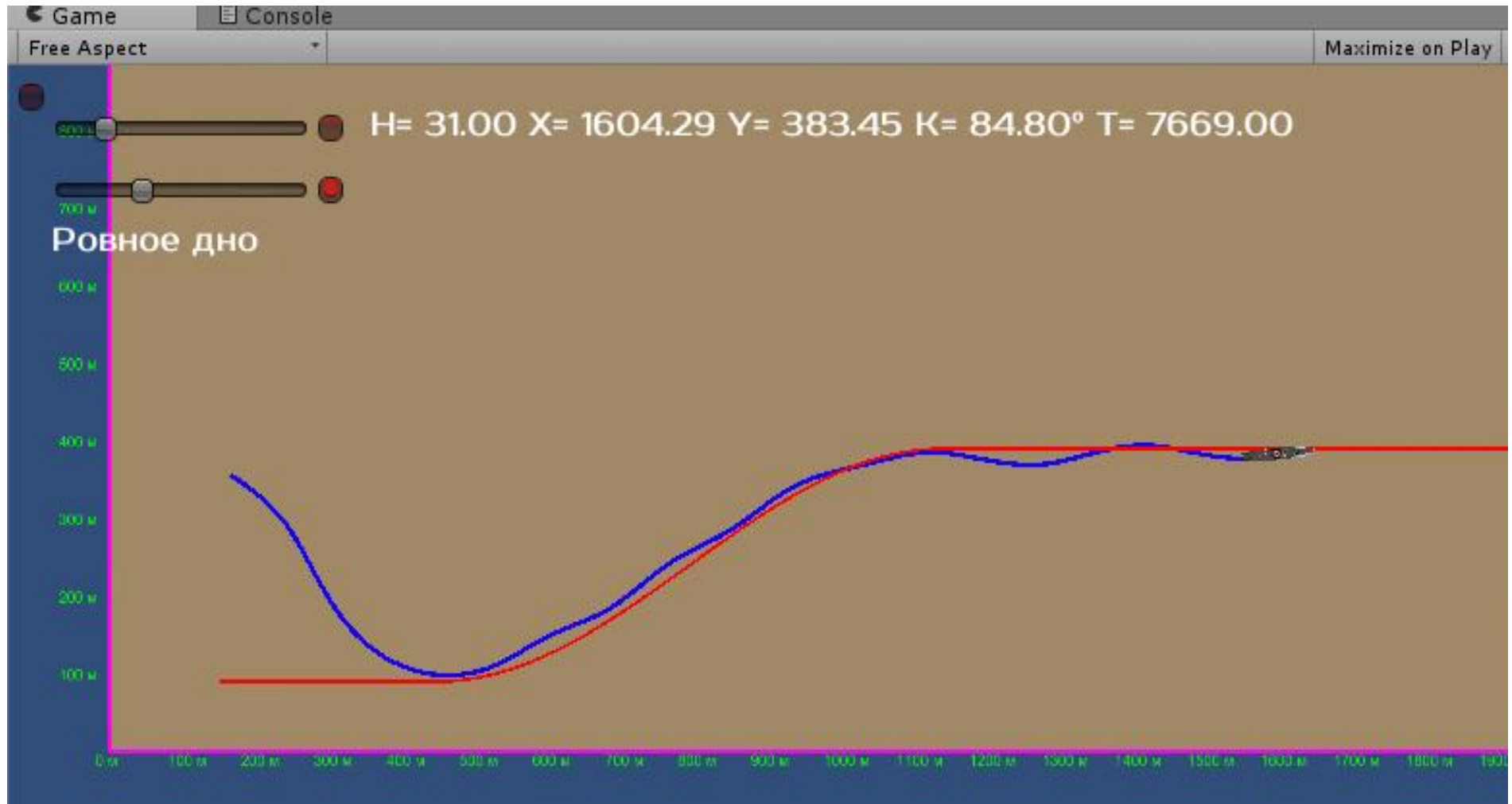


Рисунок 4.16 - Движение судна по заданной траектории при ровном дне:  $H=31$  м.,  $X=166$  м.,  $Y=356$  м.,  $K=121.74^0$ ,  $\delta=30^0$ ,  $h=0.001$ ,  $V=5$  м/с., при  $K1=10$ ,  $K2=650$ ,  $K3=0.045$ ,  $K4=-0.003$

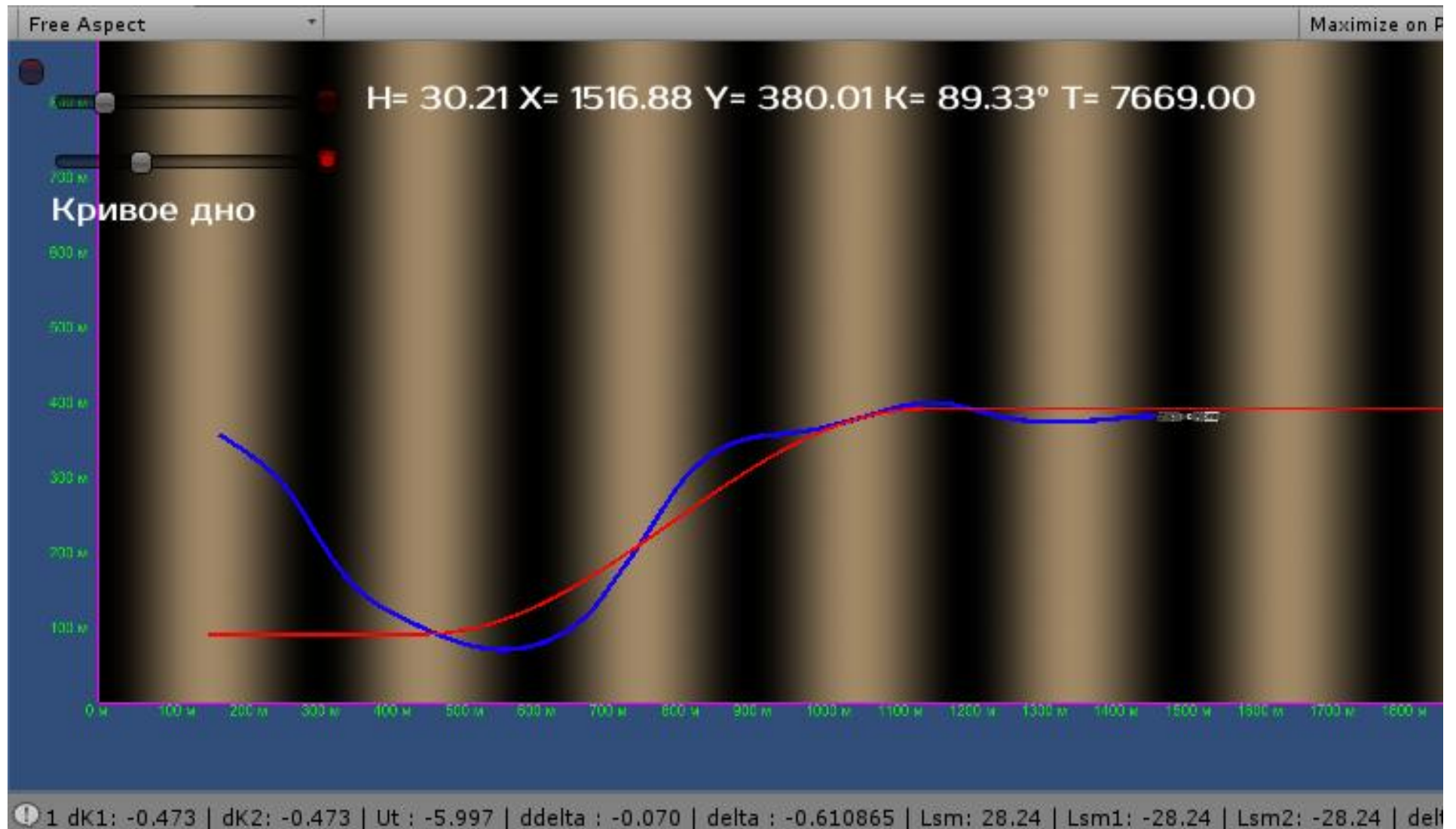


Рисунок 4.17 - Движение судна по заданной траектории при неровном дне:  $H=31$  м.,  $X=166$  м.,  $Y=356$  м.,  $K=121.74^0$ ,  $\delta=30^0$ ,  $h=0.001$ ,  $V=5$  м/с., при  $K1=10$ ,  $K2=650$ ,  $K3=0.045$ ,  $K4=-0.003$

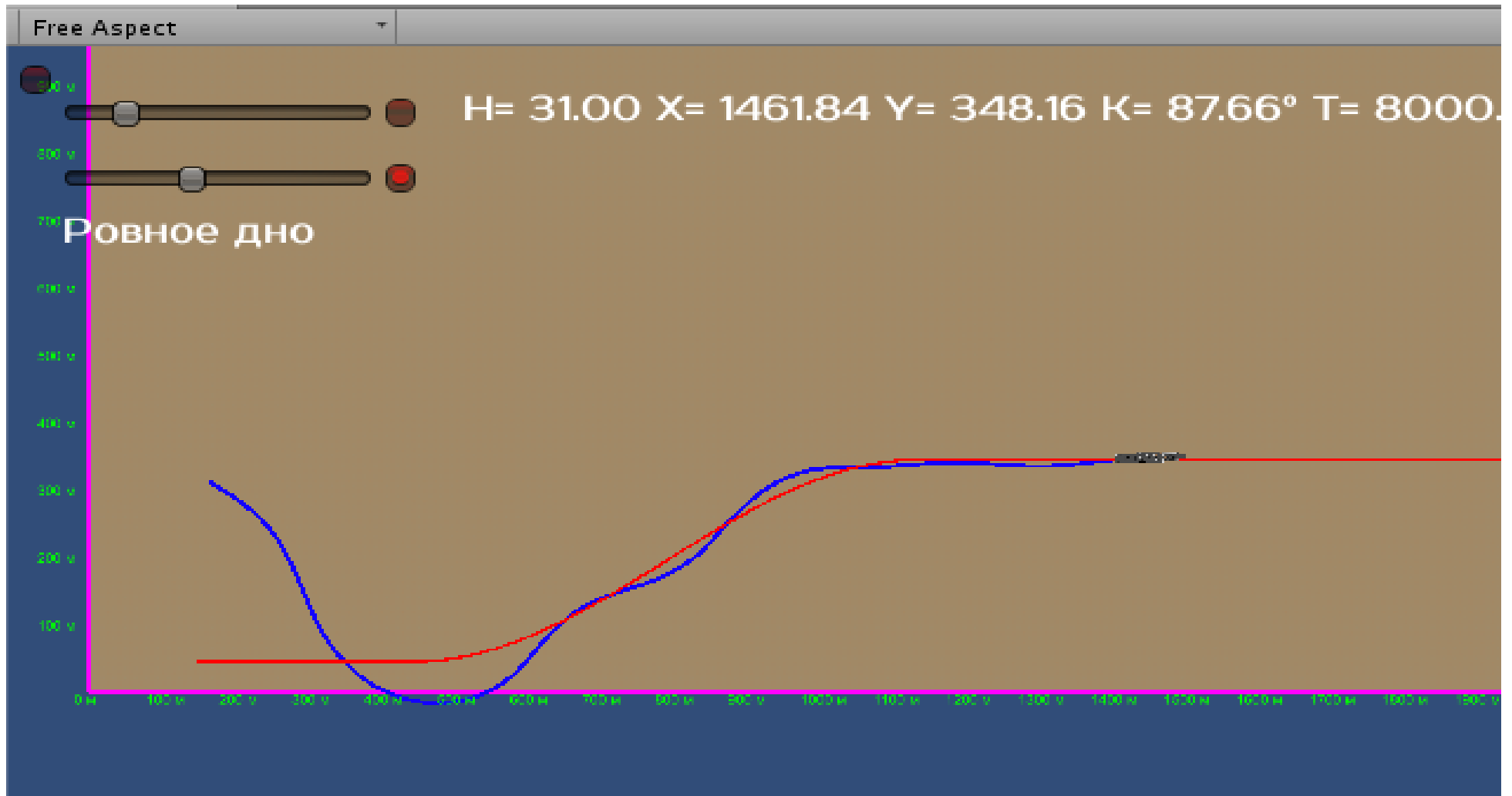


Рисунок 4.18 - Движение судна при ровном дне по заданной траектории :  $H=31$  м.,  $X=166$  м.,  $Y=312$  м.,  $K=121.74^\circ$ ,  
 $\delta=30^\circ$ ,  $h=0.001$ ,  $V=5$  м/с., при  $K_1=5$ ,  $K_2=200$ ,  $K_3=0,03$ ,  $K_4=-0.003$

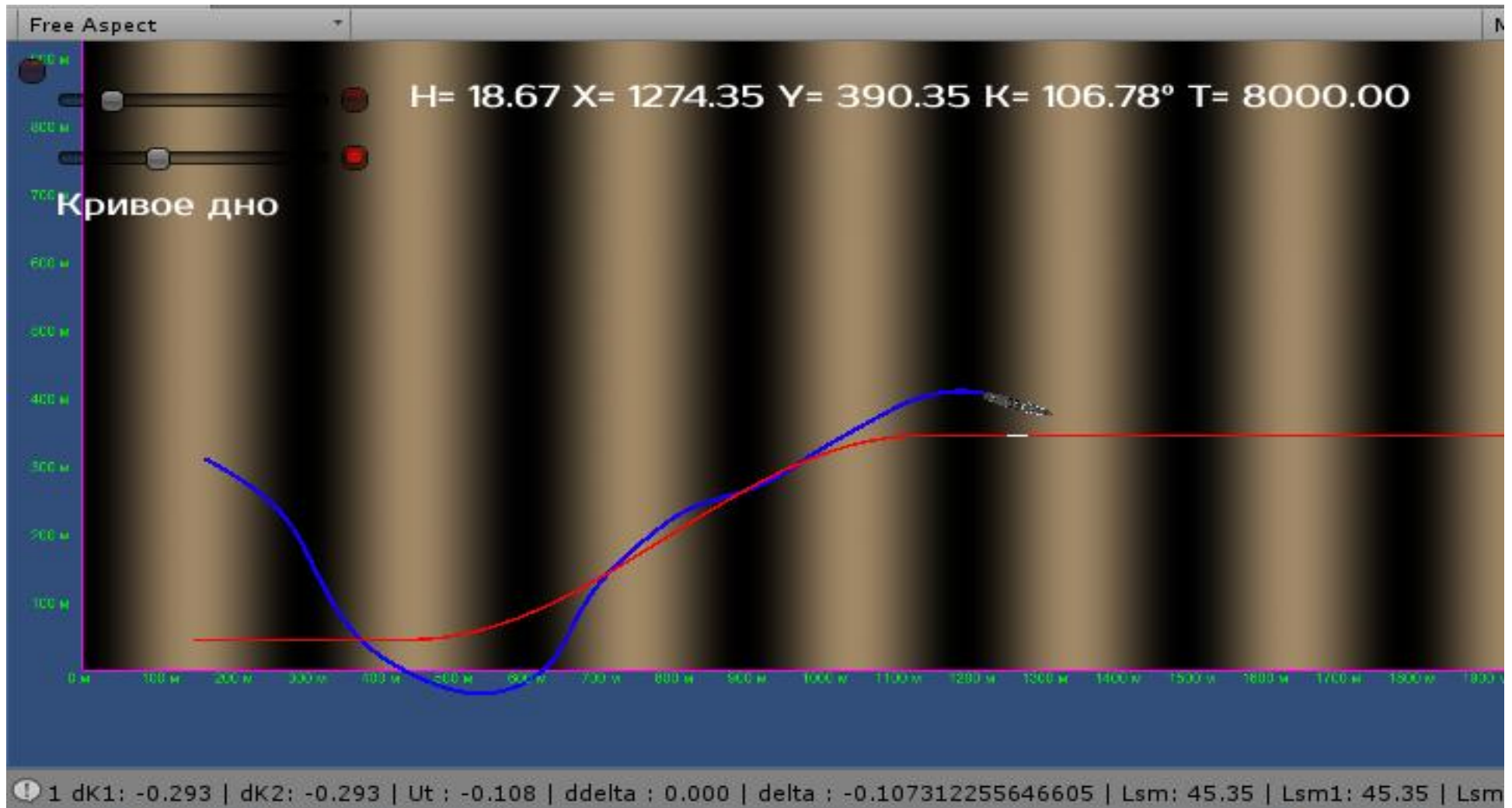


Рисунок 4.19 - Движение судна по заданной траектории при неровном дне:  $H=31$  м.,  $X=166$  м.,  $Y=312$  м.,  $K=121.74^0$ ,  
 $\delta=30^0$ ,  $h=0.001$ ,  $V=5$  м/с., при  $K1=5$ ,  $K2=200$ ,  $K3=0,03$ ,  $K4=-0.003$

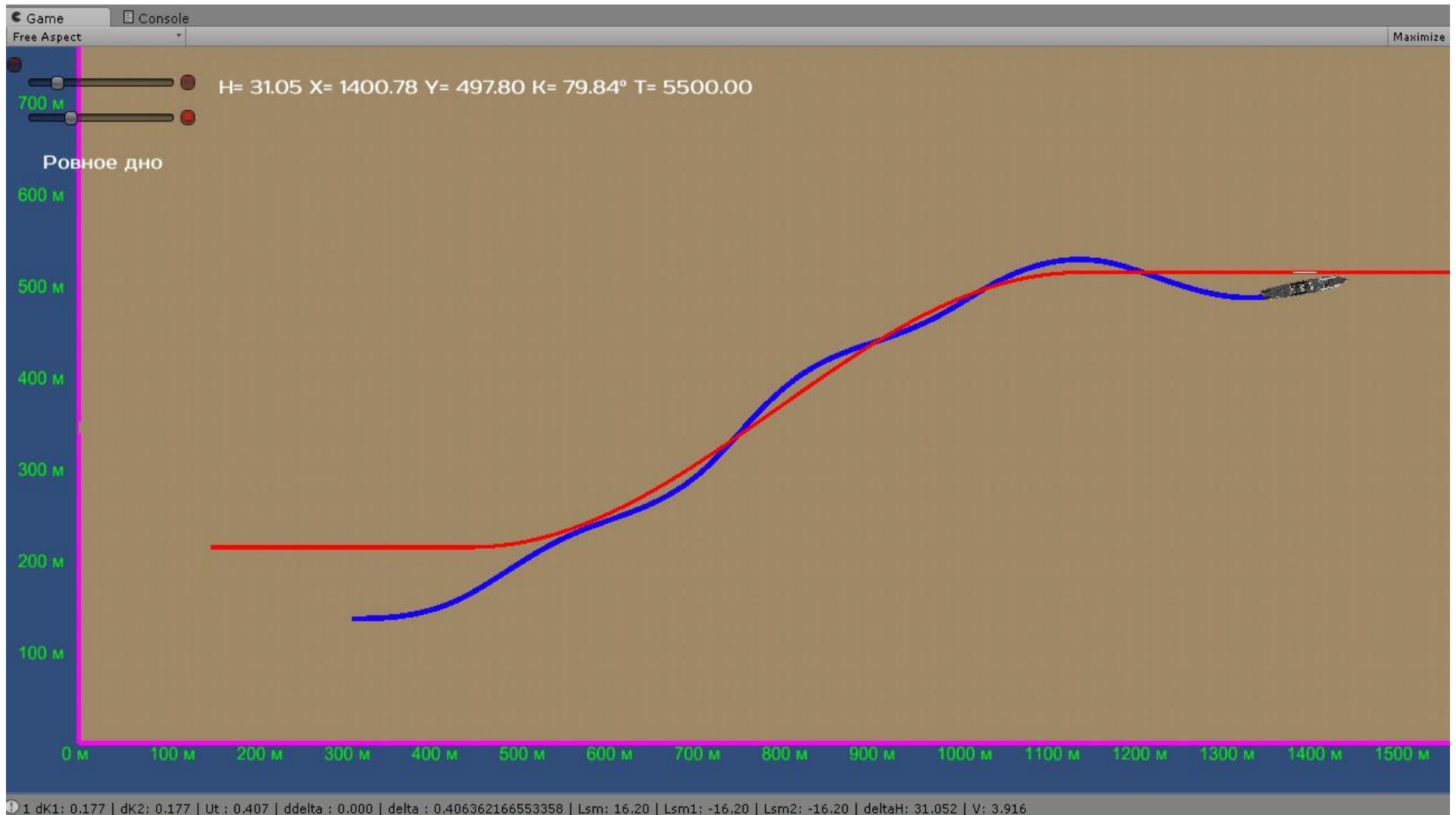


Рисунок 4.20 - Движение судна по заданной траектории при ровном дне:  $H=31.052$  м.,  $X=312$  м.,  $Y=136$  м.,  
 $K=91.983^0$ ,  $\delta=31.206^0$ ,  $h=0.001$ ,  $V=5$  м/с,  $K1=5$ ,  $K2=400$ ,  $K3=0,03$ ,  $K4=-0.003$

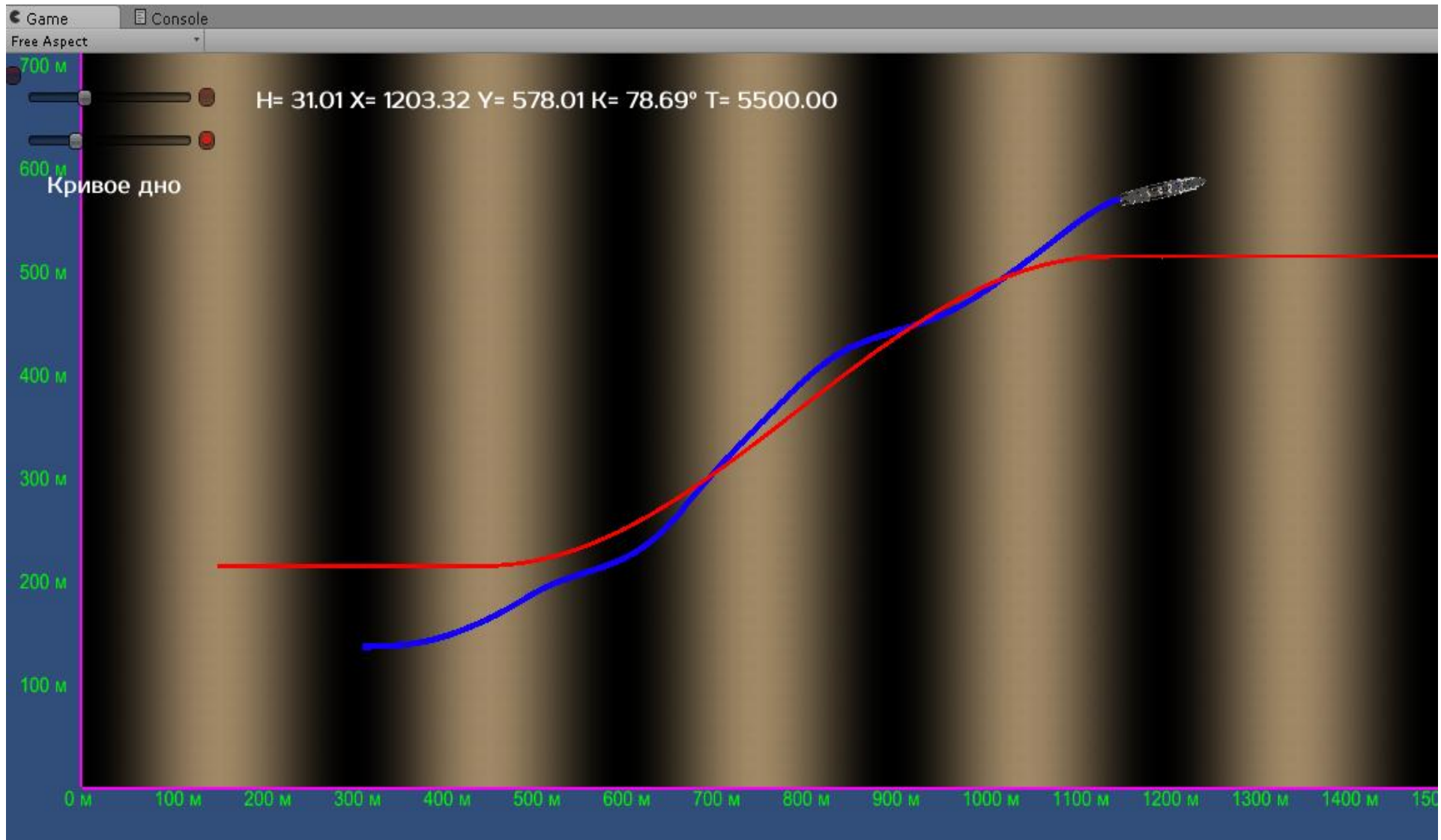


Рисунок 4.21 - Движение судна по заданной траектории при неровном дне:  $H=31.052$  м.,  $X=312$  м.,  $Y=136$  м.,  
 $K=91.983^0$ ,  $\delta=31.206^0$ ,  $h=0.001$ ,  $V=5$  м/с,  $K1=5$ ,  $K2=400$ ,  $K3=0.03$ ,  $K4=-0.003$



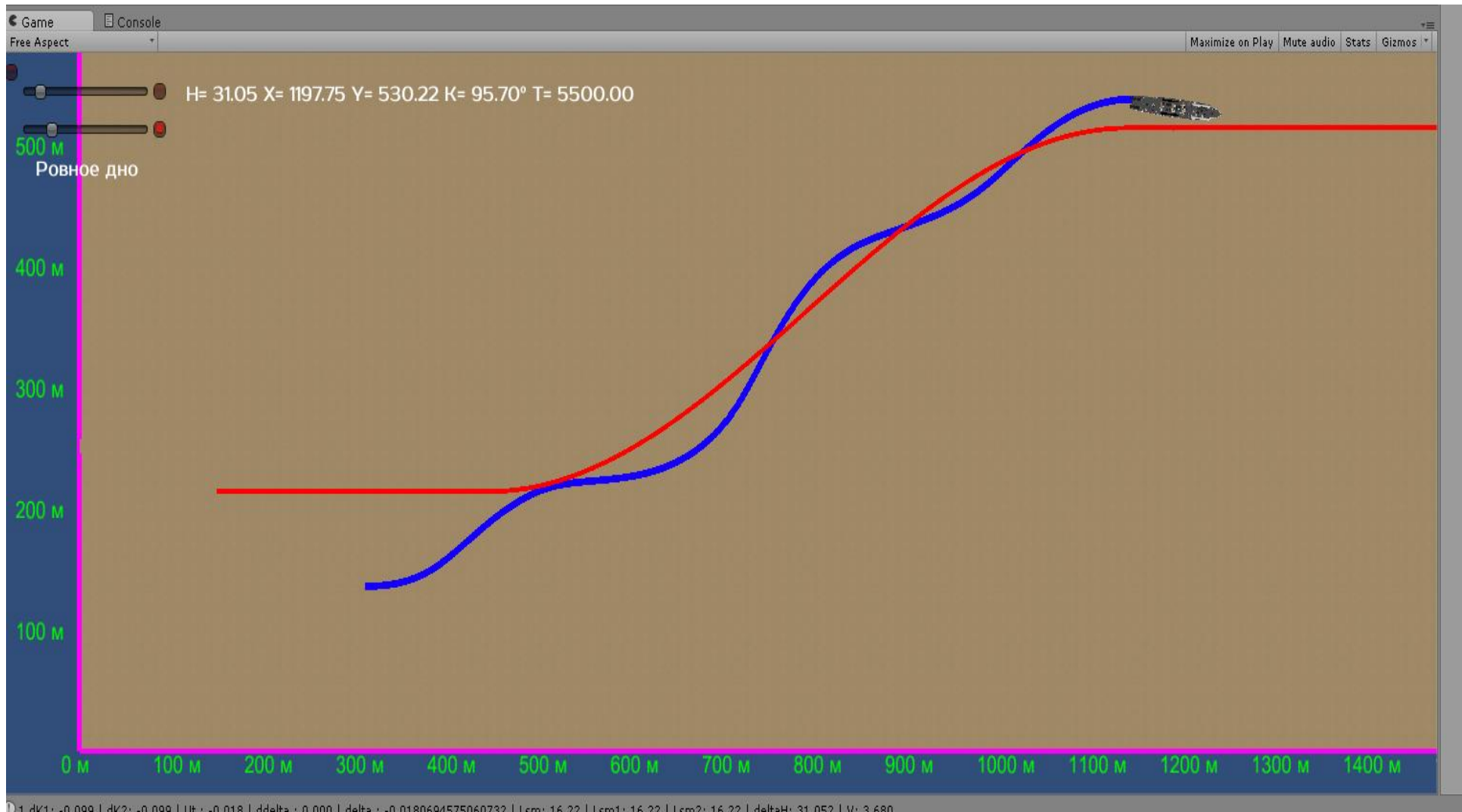


Рисунок 4.22 - Движение судна по заданной траектории при ровном дне:  $H=31.052$  м.,  $X=312$  м.,  $Y=136$  м.,  $K=90^0$ ,  
 $\delta=31.809^0$ ,  $h=0.001$ ,  $V=3$  м/с.,  $K1=5$ ,  $K2= 400$ ,  $K3=0,03$ ,  $K4=-0.003$

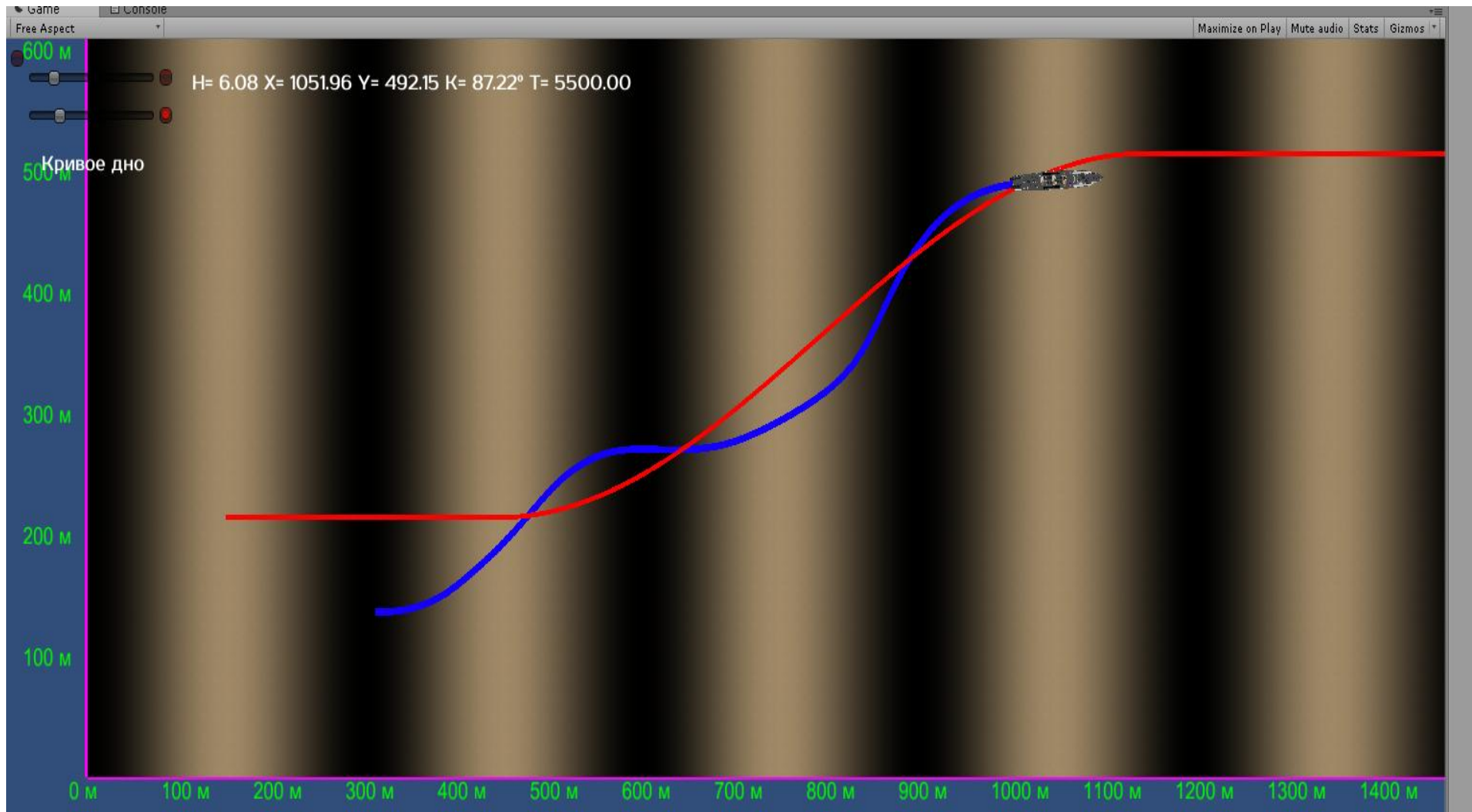


Рисунок 4.23 - Движение судна по заданной траектории при неровном дне:  $H=31.052$  м.,  $X=312$  м.,  $Y=136$  м.,  $K=90^0$ ,  
 $\delta=31.809^0$ ,  $h=0.001$ ,  $V=3$  м/с.,  $K1=5$ ,  $K2= 400$ ,  $K3=0,03$ ,  $K4=-0.003$

Так же проведен сравнительный анализ движения судна при ровном дне при тех же условиях, что и для рисунков 4.11, 4.12, но при этом, в одном случае взяты коэффициенты  $K_1=10$ , а  $K_2=650$  (рисунки 4.16, 4.17), а в другом случае  $K_1=5$ , а  $K_2=200$  (рисунки 4.18, 4.19).

Сравнительный анализ рисунков показывает, что увеличение  $K_1$  или  $K_2$  приводит к уменьшению интегральной ошибки, а их уменьшение увеличивает ошибку.

На рисунках 4.20, 4.21 и 4.22, 4.23 приведены переходные процессы при одинаковых условиях, но при различных скоростях движения судна.

Рисункам 4.20 и 4.21 соответствует скорость движения судна равная 5 м/с, а рисункам 4.22 и 4.23 - 3 м/с. Анализ графиков показывает, что с уменьшением скорости интегральная квадратичная ошибка увеличивается. Как видно из сравнения рисунков 4.9-4.23 наличие неровности дна приводит к увеличению ошибки регулирования и к ухудшению управляемости судна.

### **Выводы по четвертой главе**

1. Разработана программа, позволяющая визуально наблюдать за движением модели судна на мелководье, которая может быть использована как навигационный тренажер, а также для исследования движения судна на мелководье на его модели.

2. Для решения системы дифференциальных уравнений движения судна использован метод Рунге-Кутты.

3. Разработка программного комплекса осуществлена с помощью инструмента Unity с использованием языка C#.

4. В отличие от существующих алгоритмов и программ, разработанная программа позволяет моделировать движение судна по криволинейной траектории на мелководье при неровном рельефе дна.

## ЗАКЛЮЧЕНИЕ

При проведении исследования автором получены следующие основные научные результаты и даны рекомендации по их практическому применению:

1. Обоснована необходимость учета в разрабатываемых математических моделях движения судов по заданной криволинейной траектории на мелководье изменяющейся поверхности дна. Программная реализация таких моделей позволяет значительным образом сократить объем материальных и временных затрат, связанных с «ручным» проведением исследования поверхности дна для обеспечения адекватных динамических характеристик судов на мелководье с переменной глубиной его расположения.

2. Проведен анализ математических моделей, по результатам которого определены основные их достоинства и недостатки, связанные с моделированием процесса движения судов на мелководье.

3. Получены аналитические зависимости (представленные в виде кривых третьего порядка) влияния мелководья на гидродинамические коэффициенты в уравнениях гидродинамики судна, определяемые как функции отношения осадки судна к глубине расположения дна, что позволяет осуществить программную реализацию математической модели движения судна на мелководье при переменной глубине района плавания, без многократного перерасчета значений этих коэффициентов.

4. Разработана математическая модель циркуляции судна на мелководье при переменной глубине и выполнена ее программная реализация, что обеспечивает более широкие возможности моделирования, в частности, позволяет заменить натурные и физические эксперименты математическим моделированием и проведением исследований на ЭВМ и, как следствие, сократить на этой основе материальные и временные затраты.

5. Разработана математическая модель движения судна на мелководье при переменной глубине района плавания, с учетом аналитических

зависимостей влияния изменения глубины мелководья на гидродинамические коэффициенты в уравнениях гидродинамики судна, определяемые как функции отношения осадки судна к глубине мелководья, что позволяет проводить исследования динамических характеристик судов на мелководье при переменной глубине района плавания.

6. Разработана программа расчета для построения кривых  $D$ -разбиения, что позволило выбирать параметры авторулевого в области устойчивости системы автоматического управления движением судов.

7. Выполнена программная реализация разработанных математических моделей на мелководье с переменной глубиной. Это позволяет проводить дальнейшие исследования по изучению динамических характеристик судов на мелководье с неровным рельефом дна, а также использовать разработанный комплекс программ для выяснения поведения проектируемого судна в той или иной ситуации и при выполнении того или иного маневра, а также определять причины аварий, связанных с неэффективным маневрированием на морском и речном транспорте.

8. Дальнейшее развитие предложенных математических моделей гидродинамики движения судов на мелководье с переменной глубиной может быть связано с учетом волнения водной поверхности.

## СПИСОК ЛИТЕРАТУРЫ

1. Абдуллаева, З. М. Моделирование движения судна на мелководье / З. М. Абдуллаева, М. Л. Яхьяев // Свидетельство о государственной регистрации программы для ЭВМ №2014662089. Заявл. №2014614843 от 23.05.2014г: опубл. 21.11.2014г.
2. Абдуллаева, З.М. Моделирование движения судна на мелководье по заданной траектории при переменной глубине / З.М Абдуллаева // Вестник Дагестанского государственного технического университета. 2017. № 3 (44). – С.93-102.
3. Абдуллаева, З. М. Моделирование движения судна на мелководье по криволинейной траектории / Г.К. Асланов, З. М. Абдуллаева, М. Л. Яхьяев// Свидетельство о государственной регистрации программы для ЭВМ №2016611225. Заявл. №2015615208: опубл.28.01.2016г.
4. Абдуллаева, З. М. Влияние мелководья на гидродинамические характеристики корпуса судна / З. М.Абдуллаева // Системные технологии. – 2013.-№9.-С.66-72.;
5. Абдуллаева, З.М. Результаты моделирования движения судна на мелководье / Г.К.Асланов, З.М. Абдуллаева., М. Л. Яхьяев // Вестник Дагестанского государственного технического университета. 2015. № 1 (36). – С.45-53.
6. Антонов, В. А. Теоретические вопросы управления судном: Учебное пособие / В.А.Антонов - Владивосток: Дальневосточный университет, 1988 — 111 с.
7. Асланов, Г. К. Исследование систем автоматического вождения судов по ведущему кабелю: дис. ...канд. тех. наук: 05.22.16 / Асланов Гайдарбек Кадырбекович. - Л.: 1981. - 129с.
8. Асланов, Г. К. Аналитическое описание кривых влияния мелководья на гидродинамику судна / Г. К Асланов., З.М. Магомедова // Современные информационные технологии в проектировании, управлении и

экономике: материалы шестой Всероссийской конференции по актуальным проблемам внедрения и развития сектора IT-технологий, 28 – 29 сентября 2011 г. – Махачкала: ДГТУ, 2011. – С.83-85.;

9. Асланов, Г. К. Математическая модель движения судна на мелководье / Г. К. Асланов, З. М. Абдуллаева // Сборник тезисов докладов XXXIV итоговой научно-технической конференции преподавателей, сотрудников, аспирантов и студентов ДГТУ. – Махачкала: ДГТУ, 23 – 27 апреля 2013. – Махачкала: ДГТУ, 2013. – С.59-60.

10. Асланов, Г. К. Математическая модель движения судна по криволинейной траектории / Г. К. Асланов, З. М. Абдуллаева // Молодежь, наука, инновации: материалы II Всероссийской научно-практической конференции. – Грозный: ГГНТУ имени М. Д. Миллионщикова, 2013. – С. 27-34.;

11. Асланов, Г. К. Анализ результатов моделирования движения судна на мелководье / Г. К. Асланов, З. М. Абдуллаева // Молодежь, наука, инновации: материалы II Всероссийской научно-практической конференции. – Грозный: ГГНТУ имени М. Д. Миллионщикова, 2013. – С. 44-50.;

12. Абдуллаева, З.М. Моделирование влияния мелководья на гидродинамические коэффициенты при уравнениях гидродинамики судна / Г.К. Асланов, З.М. Абдуллаева // Вестник Дагестанского государственного технического университета. 2011. № 3 (22). – С.54-58.;

13. Абдуллаева, З.М. Математическая модель движения судна на мелководье / Г. К. Асланов З. М. Абдуллаева // Вестник Дагестанского государственного технического университета. 2012. № 4 (27). – С.36-41.

14. Абдуллаева, З.М. Результаты моделирования движения судна на мелководье / Г.К.Асланов, З.М. Абдуллаева., М. Л. Яхьяев // Вестник Дагестанского государственного технического университета. 2015. № 1 (36). – С.45-53.

15. Абдуллаева, З. М. Особенности циркуляции судна на мелководье при неровном дне // Фундаментальные и прикладные проблемы математики,

информатики в современной науке: теория и практика актуальных исследований: материалы всероссийской научно-технической конференции. – Махачкала: ДГТУ, 19-21 октября 2016.- С. 89-92.;

16. Абдуллаева, З. М. Разработка и реализация методов управления движения судна на мелководье по заданной траектории при неровном дне / З. М. Абдуллаева // Современные информационные технологии в проектировании, управлении и экономике: материалы X всероссийской научно-практической конференции. – Махачкала: ДГТУ, 2016.- С.113-119.

17. Атурин, В.В. Высшая математика. Учебное пособие для вузов / В. В. Атурин, В. В. Годин. - М.: Академия, 2010.- 304 с.

18. Баврин, И.И. Высшая математика / И. И. Баврин. - М.: Академия, 2010. – 616 с.

19. Баранов Ю.К. Навигация: учебник для вузов - 3-е издание, переработанное и дополненное / Ю. К. Баранов, М. И. Гаврюк, В. А. Логиновский, Ю. А. Песков. - СПб.: Лань, 1997. - 512 с.

20. Басин, А. М. Ходкость и управляемость судов: учебное пособие для вузов водного транспорта / А.М. Басин. - М.: Транспорт, 1977. — 457 с.

21. Басин, А. М. Гидродинамика судов на мелководье / А. М Басин, И. О. Веледницкий, А. Г Ляховицкий. - Л.: Судостроение, 1976- 318с.

22. Бахвалов, Н. С. Численные методы (анализ, алгебра, обыкновенные дифференциальные уравнения) /Н. С. Бычев. - М.: Наука, 1975. – 632 с.

23. Березин, С.Я. Системы автоматического управления движением судов по курсу / С .Я. Березин, В. А. Тетюев - JL: Судостроение, 1974. - 264 с.

24. Бесекерский, В.А. Теория систем автоматического регулирования / В.А. Бесекерский, Е.П. Попов - М.: Наука, 1975. — 767 с.

25. Большаков, В. П. К теории управляемости корабля / В. П Большаков.- Тр. ВМАКВ им. акад. А. Н. Крылова, 1959. - Вып. XIX. - С. 3-19.

26. Буслов, В.А. Введение в численный анализ / В.А. Буслов, С.Л. Яковлев. - СПб: 1999. - 99 с.

27. Буслов, В.А. Численные методы, в 2 -х ч. / В.А. Буслов, С.Л. Яковлев -



СПб, 2001.

28. Васильев, А. В. Управляемость судов / А. В. Васильев - Л.: Судостроение, 1989. - 328 с.

29. Васильев, А. В. Гидромеханика судов внутреннего плавания : учеб. пособие : в 2 ч. / А. В. Васильев, В. Н. Савинов, П. Н. Ершов ; Нижегород. гос. техн. ун-т. - 2-е изд., перераб. и доп. - Н. Новгород : НГТУ. - 1997. -2 ч.

30. Васильев, А. В. Управляемость винтового судна / А. В. Васильев, В. И. Белоглазов. - М.: Транспорт, 1966. - 167 с.

31. Войткунский, Я. И. Сопротивление движению судов / Я. И. Войткунский . - Л.: Судостроение, 1988.-288 с.

32. Войткунский, Я. И. Справочник по теории корабля: В трех томах. Том 1. Гидромеханика. Сопротивление движению судов. Судовые движители / Я. И. Войткунский. - Л.: Судостроение, 1985.- 768 с.

33. Войткунский, Я. И. Справочник по теории корабля: В трех томах. Том 3. Управляемость водоизмещающих судов. Гидродинамика судов с динамическими принципами поддержания / Я. И. Войткунский. - Л.: Судостроение, 1985.- 544 с.

34. Войткунский, Я.И. Управляемость водоизмещающих судов: Справочник по теории корабля [текст] / Я.И. Войткунский. - Л.: Судостроение, 1985.- Т.3.- 544 с.

35. Войткунский, Я. И. Справочник по теории корабля. Судовые движители и управляемость / Я. И. Войткунский, Р. Я. Першиц, И. А. Титов. - Изд. 2-е, перераб. и доп. - Л.: Судостроение, 1973. - 512 с.

36. Войткунский, Я. И. Справочник по теории корабля: Справочник для науч. раб., инженер.- проектир., студент, вузов / Я.И. Войткунский. - Л.: Судостроение, 1973 -511с.

37. Выгодский, М. Я. Справочник по высшей математике / М. Я. Выгодский. – Москва: АСТ: Астрель, 2010. – 703 с.

38. Высшая математика: учебник / К. В. Балдин, В. Н. Башлыков, А. В. Рукосуев. – Москва: Флинта: МПСИ, 2010. – 359 с.

39. Вьюгов, В. В. Приближенный метод расчета присоединенных масс речных судов / В. В. Вьюгов, В. Г. Павленко // Вопросы гидродинамики речных судов и составов : труды / НИИВТ. - Н. Новгород, 1989. - С. 4-7.
40. Вьюгов, В. В. Об адекватности математической модели движения судна В. В. Вьюгов, Б. В. Палагушкин // Проблемы комплексного развития регионов Казахстана : тез. док. науч. конф. / КазгосИНТИ. - Алматы, 1996. - С. 217-222.
41. Вьюгов В. В. Приближенный метод расчета присоединенных масс речных судов В. В. Вьюгов, В. Г Павленко // Вопросы гидродинамики речных судов и составов : труды / НИИВТ. - Н. Новгород, 1989. - С. 4-7.
42. Вьюгов, В. В., Палагушкин Б. В. Присоединенные массы транспортных судов при криволинейном движении В. В. Вьюгов, В. Г Павленко// Проблемы комплексного развития регионов Казахстана : тез. докл. науч. конф. / КазгосИНТИ. - Алматы, 1996. - С. 211-216.
43. Герберт Шилдт. С# 2.0. Полное руководство. ЭКОМ Паблишерз, 2007 г.
44. Гофман, А. Д. Движительно-рулевой комплекс и маневрирование судна. Справочник / А.Д.Гофман. - Л.: Судостроение. 1988.-360 с.
45. Гофман, А. Д. Теория и расчет поворотливости судов внутреннего плавания / А.Д.Гофман. - Л: Судостроение, 1971. -182 с.
46. Основы вычислительной математики. (3-е изд.) / б.п Демидович, И.А Марон. - М.: Наука, 1966.-С
47. Численные методы анализа. Приближение функций, дифференциальные и интегральные уравнения / Б.П. Демидович, И.А Марон, Э.З. Шувалова. - М.: Наука, 1967 - С.
48. Джеффри Рихтер, Мартен ван де Боспурт. WinRT. Программирование на С# для профессионалов. Вильямс, 2014. - 368 с.
49. Джон Скит. С# для профессионалов. Тонкости программирования. Третье издание. Вильямс, 2014. - 608 с.
50. Дорогостайский, Д. В. Теория и устройство судна / Д.В. Дорогостайский. - Л.: Судостроение, 1976-413 с.
51. Драчев, В. Н. Расчет точки перекладки руля при повороте на мелковод-

дь: / В. Н Драчев// Безопасность судоходства в Дальневосточном бассейне: сборник докладов научно-практической конференции. - Владивосток: Мор. гос. ун-т, 2007. - С. 294 - 303.

52. Драчев, В. Н. Расчет точки перекладки руля при повороте с учетом циркуляции, Сборник докладов научно-практической конференции. — Безопасность судоходства в дальневосточном бассейне / В. Н Драчев — Владивосток: Мор. гос. ун-т, 2007. - С. 282 - 294.

53. Драчев, В. Н. Учет влияния мелководья при маневрировании судном: дис. ... канд. тех. наук: 05.13.06 / Драчев Владимир Николоевич. - Владивосток, 2009. – 170 с.

54. Драчев, В.Н. Определение точки перекладки руля при учете циркуляции/ В.Н. Драчев // Вестник МГУ, сер. Судовождение, 2004, Вып. 2/2004. - С. 159 -161.

55. Драчев, В.Н. Учет циркуляции при плавании в стесненных водах: монография / В.Н. Драчев - Владивосток: Мор. Гос. Ун-т, 2008 - 97 с.

56. Зубов, В. И. Лекции по теории управления / В. И. Зубов. - М.: Наука, 1975.-496 с.

57. Касьянов, В.И.: Руководство к решению задач по высшей математике / В.И.Касьянов. - М.: Юрайт, 2011-С

58. Кацман, Ф. М. Теория и устройство судов: Учеб. Для вузов / Ф. М. Кацман. - Л.: Судостроение, 1991. —416 с.

59. Короткое, С. Н. Двумерная теория движения судна на мелководье / С.Н. Короткое. - Новосибирск : НГАВТ, 1995.-112 с.

60. Костюков, А. А. Соппротивление воды движению судов / А. А Костюков. - Л.: Судостроение, 1966.-448 с.

61. Красовский, Н.Н. Теория управления движением. Линейные системы / Н.Н. Красовский - М.: Наука, 1968. - 475 с.

62. Крылов, В.И., Бобков В.В., Монастырный П.И. Вычислительные методы, том II. - Главная редакция физико-математической литературы изд-ва «Наука» М., 1977

63. Киреев, В.И. Численные методы в примерах и задачах // В. И. Киреев, А.В.Пантелеев— М.: Высш. шк. , 2008 - 480 с.
64. Курлян, О.С. Автоматическое управление курсом судна (авторулевые)/ О.С. Курлян.- Судостроение за рубежом. - 1989. - №10. - С.73-84.
65. Лапчик М.П. Численные методы /М. П. Лапчик. - М.: Академия, 2008.
66. Лентарёв, А. А. Навигация: в 3 ч., ч. III: курс лекций / А. А. Лентарев. - Владивосток: Мор. Гос. Ун-т, 2005 - 157 с.
67. Лесков, М. М. Навигация. Учеб. для вузов - Изд. 2-е, перераб. И доп. / М. М. Лесков. - М.: Транспорт, 1986. — 360 с.
68. Лихачев, А. В. Управление судном: учебник для морских вузов / А. В. Лихачев. - СПб.: Изд-во Политехи, ун-та, 2004. 504 с.
69. Малыхин, В. И. Высшая математика: учебное пособие / В. И. Малыхин. – Москва: Инфра-М, 2010. – 363 с.
70. Мамедов Л.К. Математическое моделирование сложных систем, на примере радиопеленгатора / Л.К Мамедов, З. М. Магомедова. // Сборник тезисов докладов XXX итоговой научно-технической конференции преподавателей, сотрудников, аспирантов и студентов ДГТУ. – Махачкала: ДГТУ, 2009. – Ч.1 – С.93-95.
71. Мартынюк, Г. И. Учет ветра в математической модели судна с целью оценки его влияния на маневренные характеристики / Мартынюк Г. И., Юдин Ю. И., Юдин А. Ю. Мурманск - Вестник МГТУ, том 7, №3, 2004. - 375 - 380 с.
72. Марчук, Г. И., Методы вычислительной математики. М., Наука, 1977, 456 с.
73. Острецов, Г. Э. Испытания системы автоматического управления движением судна по заданной траектории / Г. Э. Острецов, Э. В.Дюжев, А. Л Клячко - Судостроение, 2000. - № 4. - С. 37-39.
74. Острецов, Г. Э. Метод прогнозирования фазового состояния судна специального назначения. Судостроение. - 2001. - № 6. - С. 43-44.
75. Острецов, Г. Э. Методика определения коэффициентов математической модели движения судна. Сборник трудов XXVII Всесоюз. конф. по управлению

движением / ИПУ РАН. - М., 2000. - С.3.

76. Павловская, Т.А. С#. Программирование на языке высокого уровня / Т. А. Павловская. - Питер, 2014 – 432 с.

77. Палагушкин, Б. В. Приближенный теоретический метод определения присоединенных масс транспортных судов: труды / Б. В. Палагушкин. - Новосибирск: НГАВТ, 1996. - 61 с.

78. Перщиц, Р. Я. Управляемость и управление судном - Л.: Судостроение, 1983. - 272 с.

79. Пирумов, У. Г. Численные методы / У. Г. Пирмов. М.: Изд-во МАИ, 1998. - 188с.

80. Воронов, А. А. Теория автоматического управления / Н.А. Бабаков, А.А. Воронов, А. А. Воронова, Г. А. Дидук, Н. Д. Дмитриева, Д. П. Ким, Б. М. Менский, П. Н. Попович. -М.: Высшая школа, 1986.-367 с.

81. Погосов, С. Г., Обеспечение безопасности движения судов в портовых водах. ЦБНТИ ММФ. Обзорная информация / Е. В. Борисов, В. П. Королева. "Судовождение и связь". - М.: Транспорт, 1974. - 42 с

82. Поселенов, Е. Н. Обоснование и разработка адаптивного алгоритма управления движением речного водоизмещающего судна: дис. ... канд. тех.наук: 05.13.06 / Поселенов Евгений Николаевич. - Нижний Новгород – 2010. – 128 с.

83. Поттер Д. Вычислительные методы в физике - М.: Мир, 1975. -392 с.

84. Снопков, В. И. Управление судном: Учеб. для вузов / В. И. Снопков. - М.: Транспорт, 1991-359 с.

85. Снопков, В. И. Управление судном: Учеб. для вузов / В.И. Снопков - СПб.: «Профессионал», 2004 — 535 с.

86. Соболев, Г. В. Управляемость корабля и автоматизация судовождения: Учебник для вузов / Г. В. Соболев. - Л.: Судостроение. 1976. – 479 с.

87. Сотников, И. И. Математические модели, вычислительные схемы анализа и компьютерное моделирование движения судна: дис. ... канд. тех.наук: 05.13.18 / Сотников Игорь Игоревич. - Великий Новгород, 2007. - 200 с.

88. Сотников, И. И. Об одной постановке задачи математического моделирования движения судна. Разработка программного комплекса моделирования движения судна в реальных условиях плавания / И. И. Сотников // Наука и образование - 2005 : материалы междунар. научно-техн. конф.(Мурманск, 14. апреля 2005г.) : в 7 ч. Мурман. гос. техн. ун-т. - Мурманск,2005,-4.4.-С. 260-263.

89. Красковский, А. А. Справочник по теории автоматического управления/ А.А. Красковский. - М.: Наука, Гл. ред. физ.- мат. Лит., 1987. – 712 с.

90. Нетушила, А. В. Теория автоматического управления / А. В. Нетушила, Л.С. Гольдфарб, А.В. Балтрушевич, А.В. Круг и др. - М.: Высшая школа, 1972. - ч.2. - 432 с.

91. Тумашик, А. П. Расчет гидродинамических характеристик судна при маневрировании / А. П. Тумашик - Судостроение, 1978. -№ 5. - С. 13-15.

92. Управляемость корабля / К.К. Федяновский, Г. В. Соболев. - Л.: Судпромгиз, 1963.- 376 с.

93. Фильчаков, П. Ф. Справочник по высшей математике / П. Ф.Фильчаков. – Киев: Наукова думка, 1974-744 с.

94. Хемминг, Р. В. Численные методы для научных работников и инженеров / Р. В. Хемминг - М.: Наука, 1972.- 399 с.

95. Цурбан, А.И. Определение маневренных элементов судна / А. И. Цурбан. - М.: Транспорт, 1977-126 с.

96. Юдин, Ю. И. Математические модели плоскопараллельного движения судна. Классификация и критический анализ / Ю. И. Юдин, И. И. Сотников // Вестник МГТУ -2006. - Т. 9. - №2. - С. 200-208.

Таблица условных обозначений, принятый в программах

№ п/п	Обозначение в работе	Обозначение в программе	Единица измерения	Величина
1.	$V_0$	V0	м/с	Начальное значение линейной скорости судна
2.	V	V	м/с	Текущая скорость судна
3.	W	W	м <sup>3</sup>	Объемное водоизмещение, рассчитываемое через произведение главных размерений и коэффициентов общей полноты
4.	T	T	м	Осадка судна на миделе
5.	$C_b$	$C_b$	-	Коэффициент общей полноты
6.	B	BB	м	Ширина судна по действующую ватерлинию
7.	$D_e$	$D_e$	м	Диаметр гребневого винта
8.	L	L	м	Длина судна между перпендикулярами
9.	$\beta$	B	радиан	Угол дрейфа
10.	$\Omega$	omega	радиан	Размерная угловая скорость судна
11.	$F'_{p+pn}$	Fspnp	м <sup>2</sup>	Площадь руля и рудерпоста, попадающая в струю винта

	$F_{p+pn}$	Fppn	м <sup>2</sup>	Площадь руля и рудерпоста, соответствующая осадке судна кормовой
12.	$\sum_{\epsilon}$	summab	-	Число винтов
13.	$\sigma_p$	sigmap	-	Коэффициент нагрузки винта
14.	$\sigma$	sigma	-	Коэффициент полноты площади диаметральной плоскости
15.	$\rho_z$	poz	-	Радиус инерции массы корпуса относительно вертикальной оси
16.	$\rho$	po	-	Массовая плотность жидкости (воды)
17.	$n_y^{\omega}$	nyw	-	Гидродинамический коэффициент вращательной производной поперечной силы
18.	$n_y^{\delta}$	nyd	-	Коэффициент, характеризующий эффективность установленных на судне средств управления
19.	$n_{y\epsilon}^{\beta}$	nybb	-	Позиционная производная безразмерного коэффициента поперечной силы от винта



20.	$n_y^\beta$	nyb	-	Гидродинамический коэффициент позиционной производной поперечной силы
21.	$n_{x0}$	nx0	-	Безразмерный коэффициент сопротивления движению судна
22.	$m_y^\omega$	myw	-	Гидродинамический коэффициент вращательной производной горизонтального момента
23.	$m_y^\beta$	myb	-	Гидродинамический коэффициент позиционной производной горизонтального момента
24.	$l_p$	lp	м	Отстояние баллера руля от центра тяжести корабля
25.	$l_b$	lb	м	Отстояние диска винта от центра тяжести корабля
26.	$\kappa_{\omega b}$	ksiwb	-	Поправка, учитывающая спрямляющее действие корпуса на скос у винта от кривизны траектории
27.	$\kappa_\omega$	ksiw	-	Поправка, учитывающая справляющее действие корпуса на скос у руля, обусловленный кривизной

				траектории
28.	$\kappa_{\beta e}$	ksibb	-	Поправка, учитывающая спрямляющее действие корпуса на скос у винта от угла дрейфа
29.	$\delta$	delta	Радян	Угол перекладки руля
30.	$S_k$	Sk	м	Площадь, дополняющая диаметрально плоскость в корме судна до прямоугольника
31.	$\Omega$	omega	радиан	Размерная угловая скорость вращения судна
32.	$\sigma$	sigma		Коэффициент полноты площади диаметральной плоскости
33.	h	h		Шаг моделирования
34.	$\bar{n}_y^\beta$	n_yb		Безразмерный гидродинамический коэффициент позиционной производной поперечной силы
35.	$\bar{n}_y^\omega$	n_yw		Безразмерный гидродинамический коэффициент вращательной производной поперечной силы
36.	$\bar{m}_y^\beta$	m_yb		Безразмерный гидродинамический

				коэффициент позиционной производной горизонтального момента
37.	$\bar{m}_y^\omega$	m_yw		Безразмерный гидродинамический коэффициент вращательной производной горизонтального момента
38.	$\hat{n}_y^\omega$	_nyw		$\hat{n}_y^\omega = m_{11} - \bar{n}_y^\omega$
39.	U(t)	U(t)		Закон управления рулем
40.	$\delta$	delta		Угол перекадки руля
41.	d $\delta$ /dt	ddelta		Приращение угла перекадки руля
42.	$l_{cm}$	$l_{cm}$		Отклонение судна от заданий криволинейной траектории
43.	$\Delta K$	$\Delta K$		Отклонение судна от заданного курса
44.	K1-K4	K1-K4		Коэффициенты при регуляторе
<i>Безразмерные коэффициенты инерции</i>				
45.	$m_{11}$	$m_{11}$	-	
46.	$m_{22}$	$m_{22}$	-	
47.	$m_{66}$	$m_{66}$	-	
<i>Коэффициенты влияния мелководья на присоединенные массы и инерции</i>				
48.	$k_{11}$	$k_{11}$	-	
49.	$k_{22}$	$k_{22}$	-	
50.	$k_{66}$	$k_{66}$	-	

51.	$K_1 - K_4$	$K1 - K4$	-	Коэффициенты влияния мелководья на вращательные производные для момента рыскания и для поперечной силы
<i>Коэффициенты, используемые в методе Рунге-Кутты</i>				
52.	$k_{11}$	$k_{11}$	-	
53.	$k_{12}$	$k_{12}$	-	
54.	$k_{13}$	$k_{13}$	-	
55.	$k_{14}$	$k_{14}$	-	
56.	$k_{21}$	$k_{21}$	-	
57.	$k_{22}$	$k_{22}$	-	
58.	$k_{23}$	$k_{23}$	-	
59.	$k_{24}$	$k_{24}$	-	
60.	$k_{31}$	$k_{31}$	-	
61.	$k_{32}$	$k_{32}$	-	
62.	$k_{33}$	$k_{33}$	-	
63.	$k_{34}$	$k_{34}$	-	
<b>Поправочные коэффициенты при уравнениях гидродинамики судна</b>				
<i>Присоединенные массы (инерции)</i>				
64.	$\lambda_{66}$	alfa66	-	
65.	$\lambda_{22}$	alfa22	-	
66.	$\lambda_{11}$	alfa11	-	
<i>Значения коэффициентов при полиномах, аппроксимирующих кривых влияния отношения осадки судна к глубине на гидродинамические коэффициенты.</i>				
67.	$C_{kII}$	css1	-	
68.	$B_{kII}$	bss1	-	
69.	$A_{kII}$	ass1	-	

70.	$C_{k3}$	cs3	-	
71.	$C_{k2}$	cs2	-	
72.	$C_{k1}$	cs1	-	
73.	$B_{k3}$	bs3	-	
74.	$B_{k2}$	bs2	-	
75.	$B_{k1}$	bs1	-	
76.	$A_{k3}$	as3	-	
77.	$A_{k2}$	as2	-	
78.	$A_{k1}$	as1	-	
79.	$D$	d	-	
80.	$C_{k2}$	c3	-	
81.	$C_{k4}$	c2	-	
82.	$C_{k22}$	c1	-	
83.	$B_{k2}$	b3	-	
84.	$B_{k4}$	b2	-	
85.	$B_{k22}$	b1	-	
86.	$A_{k2}$	a3	-	
87.	$A_{k4}$	a2	-	
88.	$A_{k22}$	a1	-	
89.	$a'_k$	ask	-	
90.	$a_k$	ak	-	
91.	$A_{\bar{a}1}$	A1	-	
92.	$B_{\bar{a}1}$	A2	-	
93.	$C_{\bar{a}1}$	A3	-	
94.	$D_{\bar{a}1}$	A4	-	
95.	$A_{\bar{a}3}$	A5	-	
96.	$B_{\bar{a}3}$	A6	-	
97.	$C_{a3}$	A7	-	
98.	$D_{a3}$	A8	-	

99.	$A_{a4}$	A9	-	
100.	$B_{a4}$	A10	-	
101.	$C_{a4}$	A11	-	
102.	$D_{a4}$	A12	-	
103.	$A_{\bar{b}1}$	B1	-	
104.	$B_{\bar{b}1}$	B2	-	
105.	$C_{\bar{b}1}$	B3	-	
106.	$D_{\bar{b}1}$	B4	-	
107.	$A_{\bar{b}2}$	B5	-	
108.	$B_{\bar{b}2}$	B6	-	
109.	$C_{\bar{b}2}$	B7	-	
110.	$D_{\bar{b}2}$	B8	-	

**Подпрограмма запуска Form1\_Designer.cs и Form1.cs.**

```
using System;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows.Forms;

namespace inputData
{
    static class Program
    {
        /// <summary>
        /// Главная точка входа для приложения.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

**Подпрограмма для ввода параметров судна. Расположение элементов  
ввода, надписей и кнопок (Form1\_Designer.cs)**

```

using SWF = SWF;
using TbCl = System.Windows.Forms.TabControl;
using TbPg = System.Windows.Forms.TabPage;
using PcBx = System.Windows.Forms.PictureBox;
using TxBx = System.Windows.Forms.TextBox;
using SysComIS = System.ComponentModel.ISupportInitialize;

namespace inputData
{
    partial class inFrm
    {
        private System.ComponentModel.IContainer components = null;
        override protected void Dispose(bool disp)
        {
            if ((components != null) && disp)
            {
                components.Dispose();
            }
            base.Dispose(disp);
        }
        #region
        private void InitializeComponent()
        {
            System.ComponentModel.ComponentResourceManager resources = new
System.ComponentModel.ComponentResourceManager(typeof(inFrm));
            this.tbCl1 = new SWF.TabControl();

```



```
this.tbPg1 = new TbPg;  
this.txBx29 = new TxBx;  
this.pcBx29 = new PcBx;  
this.txBx28 = new TxBx;  
this.pcBx28 = new PcBx;  
this.txBx27 = new TxBx;  
this.pcBx27 = new PcBx;  
this.txBx26 = new TxBx;  
this.pcBx26 = new PcBx;  
this.txBx25 = new TxBx;  
this.pcBx25 = new PcBx;  
this.txBx24 = new TxBx;  
this.pcBx24 = new PcBx;  
this.txBx23 = new TxBx;  
this.pcBx23 = new PcBx;  
this.txBx22 = new TxBx;  
this.pcBx22 = new PcBx;  
this.txBx21 = new TxBx;  
this.pcBx21 = new PcBx;  
this.txBx20 = new TxBx;  
this.pcBx20 = new PcBx;  
this.txBx19 = new TxBx;  
this.pcBx19 = new PcBx;  
this.txBx18 = new TxBx;  
this.pcBx18 = new PcBx;  
this.txBx17 = new TxBx;  
this.pcBx17 = new PcBx;  
this.txBx16 = new TxBx;  
this.pcBx16 = new PcBx;  
this.txBx15 = new TxBx;
```

```
this.pcBx15 = new PcBx;  
this.txBx14 = new TxBx;  
this.pcBx14 = new PcBx;  
this.txBx13 = new TxBx;  
this.pcBx13 = new PcBx;  
this.txBx12 = new TxBx;  
this.pcBx12 = new PcBx;  
this.txBx11 = new TxBx;  
this.pcBx11 = new PcBx;  
this.txBx10 = new TxBx;  
this.pcBx10 = new PcBx;  
this.txBx9 = new TxBx;  
this.pcBx9 = new PcBx;  
this.txBx8 = new TxBx;  
this.pcBx8 = new PcBx;  
this.txBx7 = new TxBx;  
this.pcBx7 = new PcBx;  
this.txBx6 = new TxBx;  
this.pcBx6 = new PcBx;  
this.txBx5 = new TxBx;  
this.pcBx5 = new PcBx;  
this.txBx4 = new TxBx;  
this.pcBx4 = new PcBx;  
this.txBx3 = new TxBx;  
this.pcBx3 = new PcBx;  
this.txBx2 = new TxBx;  
this.pcBx2 = new PcBx;  
this.txBx1 = new TxBx;  
this.pcBx1 = new PcBx;  
this.tbPg2 = new TbPg;
```

```
this.txBx51 = new TxBx;  
this.txBx50 = new TxBx;  
this.txBx40 = new TxBx;  
this.txBx41 = new TxBx;  
this.txBx42 = new TxBx;  
this.txBx43 = new TxBx;  
this.txBx44 = new TxBx;  
this.txBx45 = new TxBx;  
this.txBx46 = new TxBx;  
this.txBx47 = new TxBx;  
this.txBx48 = new TxBx;  
this.txBx49 = new TxBx;  
this.txBx39 = new TxBx;  
this.txBx38 = new TxBx;  
this.txBx37 = new TxBx;  
this.txBx36 = new TxBx;  
this.txBx35 = new TxBx;  
this.txBx34 = new TxBx;  
this.txBx33 = new TxBx;  
this.txBx32 = new TxBx;  
this.txBx31 = new TxBx;  
this.txBx30 = new TxBx;  
this.pcBx51 = new PcBx;  
this.pcBx50 = new PcBx;  
this.pcBx40 = new PcBx;  
this.pcBx41 = new PcBx;  
this.pcBx42 = new PcBx;  
this.pcBx43 = new PcBx;  
this.pcBx44 = new PcBx;  
this.pcBx45 = new PcBx;
```

```
this.pcBx46 = new PcBx;  
this.pcBx47 = new PcBx;  
this.pcBx48 = new PcBx;  
this.pcBx49 = new PcBx;  
this.pcBx39 = new PcBx;  
this.pcBx38 = new PcBx;  
this.pcBx37 = new PcBx;  
this.pcBx36 = new PcBx;  
this.pcBx35 = new PcBx;  
this.pcBx34 = new PcBx;  
this.pcBx33 = new PcBx;  
this.pcBx32 = new PcBx;  
this.pcBx31 = new PcBx;  
this.pcBx30 = new PcBx;  
this.tbPg3 = new TbPg;  
this.txBx72 = new TxBx;  
this.pcBx72 = new PcBx;  
this.txBx73 = new TxBx;  
this.pcBx73 = new PcBx;  
this.txBx74 = new TxBx;  
this.pcBx74 = new PcBx;  
this.txBx52 = new TxBx;  
this.pcBx52 = new PcBx;  
this.txBx53 = new TxBx;  
this.pcBx53 = new PcBx;  
this.txBx54 = new TxBx;  
this.pcBx54 = new PcBx;  
this.txBx55 = new TxBx;  
this.pcBx55 = new PcBx;  
this.txBx56 = new TxBx;
```

```
this.pcBx56 = new PcBx;  
this.txBx57 = new TxBx;  
this.pcBx57 = new PcBx;  
this.txBx58 = new TxBx;  
this.pcBx58 = new PcBx;  
this.txBx59 = new TxBx;  
this.pcBx59 = new PcBx;  
this.txBx60 = new TxBx;  
this.pcBx60 = new PcBx;  
this.txBx61 = new TxBx;  
this.pcBx61 = new PcBx;  
this.txBx62 = new TxBx;  
this.pcBx62 = new PcBx;  
this.txBx63 = new TxBx;  
this.pcBx63 = new PcBx;  
this.txBx64 = new TxBx;  
this.pcBx64 = new PcBx;  
this.txBx65 = new TxBx;  
this.pcBx65 = new PcBx;  
this.txBx66 = new TxBx;  
this.pcBx66 = new PcBx;  
this.txBx67 = new TxBx;  
this.pcBx67 = new PcBx;  
this.txBx68 = new TxBx;  
this.pcBx68 = new PcBx;  
this.txBx69 = new TxBx;  
this.pcBx69 = new PcBx;  
this.txBx70 = new TxBx;  
this.pcBx70 = new PcBx;  
this.txBx71 = new TxBx;
```

```
this.pcBx71 = new PcBx;
this.butn1 = new SWF.Button();
this.butn2 = new SWF.Button();
this.tbC11.SuspendLayout();
this.tbPg1.SuspendLayout();
((SysComIS)(this.pcBx29)).BeginInit();
((SysComIS)(this.pcBx28)).BeginInit();
((SysComIS)(this.pcBx27)).BeginInit();
((SysComIS)(this.pcBx26)).BeginInit();
((SysComIS)(this.pcBx25)).BeginInit();
((SysComIS)(this.pcBx24)).BeginInit();
((SysComIS)(this.pcBx23)).BeginInit();
((SysComIS)(this.pcBx22)).BeginInit();
((SysComIS)(this.pcBx21)).BeginInit();
((SysComIS)(this.pcBx20)).BeginInit();
((SysComIS)(this.pcBx19)).BeginInit();
((SysComIS)(this.pcBx18)).BeginInit();
((SysComIS)(this.pcBx17)).BeginInit();
((SysComIS)(this.pcBx16)).BeginInit();
((SysComIS)(this.pcBx15)).BeginInit();
((SysComIS)(this.pcBx14)).BeginInit();
((SysComIS)(this.pcBx13)).BeginInit();
((SysComIS)(this.pcBx12)).BeginInit();
((SysComIS)(this.pcBx11)).BeginInit();
((SysComIS)(this.pcBx10)).BeginInit();
((SysComIS)(this.pcBx9)).BeginInit();
((SysComIS)(this.pcBx8)).BeginInit();
((SysComIS)(this.pcBx7)).BeginInit();
((SysComIS)(this.pcBx6)).BeginInit();
((SysComIS)(this.pcBx5)).BeginInit();
```

```
((SysComIS)(this.pcBx4)).BeginInit();
((SysComIS)(this.pcBx3)).BeginInit();
((SysComIS)(this.pcBx2)).BeginInit();
((SysComIS)(this.pcBx1)).BeginInit();
this.tbPg2.SuspendLayout();
((SysComIS)(this.pcBx51)).BeginInit();
((SysComIS)(this.pcBx50)).BeginInit();
((SysComIS)(this.pcBx40)).BeginInit();
((SysComIS)(this.pcBx41)).BeginInit();
((SysComIS)(this.pcBx42)).BeginInit();
((SysComIS)(this.pcBx43)).BeginInit();
((SysComIS)(this.pcBx44)).BeginInit();
((SysComIS)(this.pcBx45)).BeginInit();
((SysComIS)(this.pcBx46)).BeginInit();
((SysComIS)(this.pcBx47)).BeginInit();
((SysComIS)(this.pcBx48)).BeginInit();
((SysComIS)(this.pcBx49)).BeginInit();
((SysComIS)(this.pcBx39)).BeginInit();
((SysComIS)(this.pcBx38)).BeginInit();
((SysComIS)(this.pcBx37)).BeginInit();
((SysComIS)(this.pcBx36)).BeginInit();
((SysComIS)(this.pcBx35)).BeginInit();
((SysComIS)(this.pcBx34)).BeginInit();
((SysComIS)(this.pcBx33)).BeginInit();
((SysComIS)(this.pcBx32)).BeginInit();
((SysComIS)(this.pcBx31)).BeginInit();
((SysComIS)(this.pcBx30)).BeginInit();
this.tbPg3.SuspendLayout();
((SysComIS)(this.pcBx72)).BeginInit();
((SysComIS)(this.pcBx73)).BeginInit();
```

```
((SysComIS)(this.pcBx74)).BeginInit();
((SysComIS)(this.pcBx52)).BeginInit();
((SysComIS)(this.pcBx53)).BeginInit();
((SysComIS)(this.pcBx54)).BeginInit();
((SysComIS)(this.pcBx55)).BeginInit();
((SysComIS)(this.pcBx56)).BeginInit();
((SysComIS)(this.pcBx57)).BeginInit();
((SysComIS)(this.pcBx58)).BeginInit();
((SysComIS)(this.pcBx59)).BeginInit();
((SysComIS)(this.pcBx60)).BeginInit();
((SysComIS)(this.pcBx61)).BeginInit();
((SysComIS)(this.pcBx62)).BeginInit();
((SysComIS)(this.pcBx63)).BeginInit();
((SysComIS)(this.pcBx64)).BeginInit();
((SysComIS)(this.pcBx65)).BeginInit();
((SysComIS)(this.pcBx66)).BeginInit();
((SysComIS)(this.pcBx67)).BeginInit();
((SysComIS)(this.pcBx68)).BeginInit();
((SysComIS)(this.pcBx69)).BeginInit();
((SysComIS)(this.pcBx70)).BeginInit();
((SysComIS)(this.pcBx71)).BeginInit();
this.SuspendLayout();
//
// tbC11
//
this.tbC11.Controls.Add(this.tbPg1);
this.tbC11.Controls.Add(this.tbPg2);
this.tbC11.Controls.Add(this.tbPg3);
this.tbC11.Location = new System.Drawing.Point(12, 12);
this.tbC11.Name = "tbC11";
```



```
this.tbC11.SelectedIndex = 0;
this.tbC11.Size = new System.Drawing.Size(665, 510);
this.tbC11.TabIndex = 60;
//
// tbPg1
//
this.tbPg1.BorderStyle = SWF.BorderStyle.FixedSingle;
this.tbPg1.Controls.Add(this.txBx29);
this.tbPg1.Controls.Add(this.pcBx29);
this.tbPg1.Controls.Add(this.txBx28);
this.tbPg1.Controls.Add(this.pcBx28);
this.tbPg1.Controls.Add(this.txBx27);
this.tbPg1.Controls.Add(this.pcBx27);
this.tbPg1.Controls.Add(this.txBx26);
this.tbPg1.Controls.Add(this.pcBx26);
this.tbPg1.Controls.Add(this.txBx25);
this.tbPg1.Controls.Add(this.pcBx25);
this.tbPg1.Controls.Add(this.txBx24);
this.tbPg1.Controls.Add(this.pcBx24);
this.tbPg1.Controls.Add(this.txBx23);
this.tbPg1.Controls.Add(this.pcBx23);
this.tbPg1.Controls.Add(this.txBx22);
this.tbPg1.Controls.Add(this.pcBx22);
this.tbPg1.Controls.Add(this.txBx21);
this.tbPg1.Controls.Add(this.pcBx21);
this.tbPg1.Controls.Add(this.txBx20);
this.tbPg1.Controls.Add(this.pcBx20);
this.tbPg1.Controls.Add(this.txBx19);
this.tbPg1.Controls.Add(this.pcBx19);
this.tbPg1.Controls.Add(this.txBx18);
```

```
this.tbPg1.Controls.Add(this.pcBx18);
this.tbPg1.Controls.Add(this.txBx17);
this.tbPg1.Controls.Add(this.pcBx17);
this.tbPg1.Controls.Add(this.txBx16);
this.tbPg1.Controls.Add(this.pcBx16);
this.tbPg1.Controls.Add(this.txBx15);
this.tbPg1.Controls.Add(this.pcBx15);
this.tbPg1.Controls.Add(this.txBx14);
this.tbPg1.Controls.Add(this.pcBx14);
this.tbPg1.Controls.Add(this.txBx13);
this.tbPg1.Controls.Add(this.pcBx13);
this.tbPg1.Controls.Add(this.txBx12);
this.tbPg1.Controls.Add(this.pcBx12);
this.tbPg1.Controls.Add(this.txBx11);
this.tbPg1.Controls.Add(this.pcBx11);
this.tbPg1.Controls.Add(this.txBx10);
this.tbPg1.Controls.Add(this.pcBx10);
this.tbPg1.Controls.Add(this.txBx9);
this.tbPg1.Controls.Add(this.pcBx9);
this.tbPg1.Controls.Add(this.txBx8);
this.tbPg1.Controls.Add(this.pcBx8);
this.tbPg1.Controls.Add(this.txBx7);
this.tbPg1.Controls.Add(this.pcBx7);
this.tbPg1.Controls.Add(this.txBx6);
this.tbPg1.Controls.Add(this.pcBx6);
this.tbPg1.Controls.Add(this.txBx5);
this.tbPg1.Controls.Add(this.pcBx5);
this.tbPg1.Controls.Add(this.txBx4);
this.tbPg1.Controls.Add(this.pcBx4);
this.tbPg1.Controls.Add(this.txBx3);
```

```
this.tbPg1.Controls.Add(this.pcBx3);
this.tbPg1.Controls.Add(this.txBx2);
this.tbPg1.Controls.Add(this.pcBx2);
this.tbPg1.Controls.Add(this.txBx1);
this.tbPg1.Controls.Add(this.pcBx1);
this.tbPg1.Location = new System.Drawing.Point(4, 22);
this.tbPg1.Name = "tbPg1";
this.tbPg1.Padding = new SWF.Padding(3);
this.tbPg1.Size = new System.Drawing.Size(657, 484);
this.tbPg1.TabIndex = 0;
this.tbPg1.Text = "1";
this.tbPg1.UseVisualStyleBackColor = true;
//
// txBx29
//
this.txBx29.Location = new System.Drawing.Point(544, 396);
this.txBx29.Name = "txBx29";
this.txBx29.Size = new System.Drawing.Size(82, 20);
this.txBx29.TabIndex = 117;
this.txBx29.Tag = "Fsppn";
this.txBx29.Text = "0";
//
// pcBx29
//
this.pcBx29.Image = global::inputData.Properties.Resources.Fsppn;
this.pcBx29.Location = new System.Drawing.Point(474, 386);
this.pcBx29.Name = "pcBx29";
this.pcBx29.Size = new System.Drawing.Size(64, 40);
this.pcBx29.TabIndex = 116;
this.pcBx29.TabStop = false;
```

```
//  
// txBx28  
//  
this.txBx28.Location = new System.Drawing.Point(544, 350);  
this.txBx28.Name = "txBx28";  
this.txBx28.Size = new System.Drawing.Size(82, 20);  
this.txBx28.TabIndex = 115;  
this.txBx28.Tag = "Fppn";  
this.txBx28.Text = "10";  
//  
// pcBx28  
//  
this.pcBx28.Image = global::inputData.Properties.Resources.Fppn;  
this.pcBx28.Location = new System.Drawing.Point(474, 340);  
this.pcBx28.Name = "pcBx28";  
this.pcBx28.Size = new System.Drawing.Size(64, 40);  
this.pcBx28.TabIndex = 114;  
this.pcBx28.TabStop = false;  
//  
// txBx27  
//  
this.txBx27.Location = new System.Drawing.Point(544, 304);  
this.txBx27.Name = "txBx27";  
this.txBx27.Size = new System.Drawing.Size(82, 20);  
this.txBx27.TabIndex = 113;  
this.txBx27.Tag = "W";  
this.txBx27.Text = "5008";  
//  
// pcBx27  
//
```

```
this.pcBx27.Image = global::inputData.Properties.Resources.W;
this.pcBx27.Location = new System.Drawing.Point(498, 294);
this.pcBx27.Name = "pcBx27";
this.pcBx27.Size = new System.Drawing.Size(40, 40);
this.pcBx27.TabIndex = 112;
this.pcBx27.TabStop = false;
//
// txBx26
//
this.txBx26.Location = new System.Drawing.Point(544, 258);
this.txBx26.Name = "txBx26";
this.txBx26.Size = new System.Drawing.Size(82, 20);
this.txBx26.TabIndex = 111;
this.txBx26.Tag = "T";
this.txBx26.Text = "4.6";
//
// pcBx26
//
this.pcBx26.Image = global::inputData.Properties.Resources.T;
this.pcBx26.Location = new System.Drawing.Point(498, 248);
this.pcBx26.Name = "pcBx26";
this.pcBx26.Size = new System.Drawing.Size(40, 40);
this.pcBx26.TabIndex = 110;
this.pcBx26.TabStop = false;
//
// txBx25
//
this.txBx25.Location = new System.Drawing.Point(544, 212);
this.txBx25.Name = "txBx25";
this.txBx25.Size = new System.Drawing.Size(82, 20);
```

```
this.txBx25.TabIndex = 109;
this.txBx25.Tag = "summab";
this.txBx25.Text = "2";
//
// pcBx25
//
this.pcBx25.Image = global::inputData.Properties.Resources.summab;
this.pcBx25.Location = new System.Drawing.Point(498, 202);
this.pcBx25.Name = "pcBx25";
this.pcBx25.Size = new System.Drawing.Size(40, 40);
this.pcBx25.TabIndex = 108;
this.pcBx25.TabStop = false;
//
// txBx24
//
this.txBx24.Location = new System.Drawing.Point(544, 166);
this.txBx24.Name = "txBx24";
this.txBx24.Size = new System.Drawing.Size(82, 20);
this.txBx24.TabIndex = 107;
this.txBx24.Tag = "sigmap";
this.txBx24.Text = "0.289";
//
// pcBx24
//
this.pcBx24.Image = global::inputData.Properties.Resources.sigmapo;
this.pcBx24.Location = new System.Drawing.Point(498, 156);
this.pcBx24.Name = "pcBx24";
this.pcBx24.Size = new System.Drawing.Size(40, 40);
this.pcBx24.TabIndex = 106;
this.pcBx24.TabStop = false;
```

```
//  
// txBx23  
//  
this.txBx23.Location = new System.Drawing.Point(544, 120);  
this.txBx23.Name = "txBx23";  
this.txBx23.Size = new System.Drawing.Size(82, 20);  
this.txBx23.TabIndex = 105;  
this.txBx23.Tag = "sigma";  
this.txBx23.Text = "0.971";  
//  
// pcBx23  
//  
this.pcBx23.Image = global::inputData.Properties.Resources.sigma;  
this.pcBx23.Location = new System.Drawing.Point(498, 110);  
this.pcBx23.Name = "pcBx23";  
this.pcBx23.Size = new System.Drawing.Size(40, 40);  
this.pcBx23.TabIndex = 104;  
this.pcBx23.TabStop = false;  
//  
// txBx22  
//  
this.txBx22.Location = new System.Drawing.Point(544, 74);  
this.txBx22.Name = "txBx22";  
this.txBx22.Size = new System.Drawing.Size(82, 20);  
this.txBx22.TabIndex = 103;  
this.txBx22.Tag = "poz";  
this.txBx22.Text = "2";  
//  
// pcBx22  
//
```

```
this.pcBx22.Image = global::inputData.Properties.Resources.poz;
this.pcBx22.Location = new System.Drawing.Point(498, 64);
this.pcBx22.Name = "pcBx22";
this.pcBx22.Size = new System.Drawing.Size(40, 40);
this.pcBx22.TabIndex = 102;
this.pcBx22.TabStop = false;
//
// txBx21
//
this.txBx21.Location = new System.Drawing.Point(544, 28);
this.txBx21.Name = "txBx21";
this.txBx21.Size = new System.Drawing.Size(82, 20);
this.txBx21.TabIndex = 101;
this.txBx21.Tag = "po";
this.txBx21.Text = "1";
//
// pcBx21
//
this.pcBx21.Image = global::inputData.Properties.Resources.po;
this.pcBx21.Location = new System.Drawing.Point(498, 18);
this.pcBx21.Name = "pcBx21";
this.pcBx21.Size = new System.Drawing.Size(40, 40);
this.pcBx21.TabIndex = 100;
this.pcBx21.TabStop = false;
//
// txBx20
//
this.txBx20.Location = new System.Drawing.Point(304, 442);
this.txBx20.Name = "txBx20";
this.txBx20.Size = new System.Drawing.Size(82, 20);
```



```
this.txBx20.TabIndex = 99;
this.txBx20.Tag = "nyw";
this.txBx20.Text = "0.014";
//
// pcBx20
//
this.pcBx20.Image = global::inputData.Properties.Resources.nyw;
this.pcBx20.Location = new System.Drawing.Point(258, 432);
this.pcBx20.Name = "pcBx20";
this.pcBx20.Size = new System.Drawing.Size(40, 40);
this.pcBx20.TabIndex = 98;
this.pcBx20.TabStop = false;
//
// txBx19
//
this.txBx19.Location = new System.Drawing.Point(304, 396);
this.txBx19.Name = "txBx19";
this.txBx19.Size = new System.Drawing.Size(82, 20);
this.txBx19.TabIndex = 97;
this.txBx19.Tag = "nyd";
this.txBx19.Text = "0.035";
//
// pcBx19
//
this.pcBx19.Image = global::inputData.Properties.Resources.nyd;
this.pcBx19.Location = new System.Drawing.Point(258, 386);
this.pcBx19.Name = "pcBx19";
this.pcBx19.Size = new System.Drawing.Size(40, 40);
this.pcBx19.TabIndex = 96;
this.pcBx19.TabStop = false;
```

```
//  
// txBx18  
//  
this.txBx18.Location = new System.Drawing.Point(304, 350);  
this.txBx18.Name = "txBx18";  
this.txBx18.Size = new System.Drawing.Size(82, 20);  
this.txBx18.TabIndex = 95;  
this.txBx18.Tag = "nybb";  
this.txBx18.Text = "0.011";  
//  
// pcBx18  
//  
this.pcBx18.Image = global::inputData.Properties.Resources.nyBb;  
this.pcBx18.Location = new System.Drawing.Point(258, 340);  
this.pcBx18.Name = "pcBx18";  
this.pcBx18.Size = new System.Drawing.Size(40, 40);  
this.pcBx18.TabIndex = 94;  
this.pcBx18.TabStop = false;  
//  
// txBx17  
//  
this.txBx17.Location = new System.Drawing.Point(304, 304);  
this.txBx17.Name = "txBx17";  
this.txBx17.Size = new System.Drawing.Size(82, 20);  
this.txBx17.TabIndex = 93;  
this.txBx17.Tag = "nyb";  
this.txBx17.Text = "0.249";  
//  
// pcBx17  
//
```

```
this.pcBx17.Image = global::inputData.Properties.Resources.nyB;
this.pcBx17.Location = new System.Drawing.Point(258, 294);
this.pcBx17.Name = "pcBx17";
this.pcBx17.Size = new System.Drawing.Size(40, 40);
this.pcBx17.TabIndex = 92;
this.pcBx17.TabStop = false;
//
// txBx16
//
this.txBx16.Location = new System.Drawing.Point(304, 258);
this.txBx16.Name = "txBx16";
this.txBx16.Size = new System.Drawing.Size(82, 20);
this.txBx16.TabIndex = 91;
this.txBx16.Tag = "nx0";
this.txBx16.Text = "0.009";
//
// pcBx16
//
this.pcBx16.Image = global::inputData.Properties.Resources.nx0;
this.pcBx16.Location = new System.Drawing.Point(258, 248);
this.pcBx16.Name = "pcBx16";
this.pcBx16.Size = new System.Drawing.Size(40, 40);
this.pcBx16.TabIndex = 90;
this.pcBx16.TabStop = false;
//
// txBx15
//
this.txBx15.Location = new System.Drawing.Point(304, 212);
this.txBx15.Name = "txBx15";
this.txBx15.Size = new System.Drawing.Size(82, 20);
```

```
this.txBx15.TabIndex = 89;
this.txBx15.Tag = "myw";
this.txBx15.Text = "-0.026";
//
// pcBx15
//
this.pcBx15.Image = global::inputData.Properties.Resources.myw;
this.pcBx15.Location = new System.Drawing.Point(258, 202);
this.pcBx15.Name = "pcBx15";
this.pcBx15.Size = new System.Drawing.Size(40, 40);
this.pcBx15.TabIndex = 88;
this.pcBx15.TabStop = false;
//
// txBx14
//
this.txBx14.Location = new System.Drawing.Point(304, 166);
this.txBx14.Name = "txBx14";
this.txBx14.Size = new System.Drawing.Size(82, 20);
this.txBx14.TabIndex = 87;
this.txBx14.Tag = "myb";
this.txBx14.Text = "0.084";
//
// pcBx14
//
this.pcBx14.Image = global::inputData.Properties.Resources.myB;
this.pcBx14.Location = new System.Drawing.Point(258, 156);
this.pcBx14.Name = "pcBx14";
this.pcBx14.Size = new System.Drawing.Size(40, 40);
this.pcBx14.TabIndex = 86;
this.pcBx14.TabStop = false;
```

```
//  
// txBx13  
//  
this.txBx13.Location = new System.Drawing.Point(304, 120);  
this.txBx13.Name = "txBx13";  
this.txBx13.Size = new System.Drawing.Size(82, 20);  
this.txBx13.TabIndex = 85;  
this.txBx13.Tag = "lp";  
this.txBx13.Text = "48";  
//  
// pcBx13  
//  
this.pcBx13.Image = global::inputData.Properties.Resources.lp;  
this.pcBx13.Location = new System.Drawing.Point(258, 110);  
this.pcBx13.Name = "pcBx13";  
this.pcBx13.Size = new System.Drawing.Size(40, 40);  
this.pcBx13.TabIndex = 84;  
this.pcBx13.TabStop = false;  
//  
// txBx12  
//  
this.txBx12.Location = new System.Drawing.Point(304, 74);  
this.txBx12.Name = "txBx12";  
this.txBx12.Size = new System.Drawing.Size(82, 20);  
this.txBx12.TabIndex = 83;  
this.txBx12.Tag = "lb";  
this.txBx12.Text = "47";  
//  
// pcBx12  
//
```

```
this.pcBx12.Image = global::inputData.Properties.Resources.lb;
this.pcBx12.Location = new System.Drawing.Point(258, 64);
this.pcBx12.Name = "pcBx12";
this.pcBx12.Size = new System.Drawing.Size(40, 40);
this.pcBx12.TabIndex = 82;
this.pcBx12.TabStop = false;
//
// txBx11
//
this.txBx11.Location = new System.Drawing.Point(304, 28);
this.txBx11.Name = "txBx11";
this.txBx11.Size = new System.Drawing.Size(82, 20);
this.txBx11.TabIndex = 81;
this.txBx11.Tag = "L";
this.txBx11.Text = "100";
//
// pcBx11
//
this.pcBx11.Image = global::inputData.Properties.Resources.L;
this.pcBx11.Location = new System.Drawing.Point(258, 18);
this.pcBx11.Name = "pcBx11";
this.pcBx11.Size = new System.Drawing.Size(40, 40);
this.pcBx11.TabIndex = 80;
this.pcBx11.TabStop = false;
//
// txBx10
//
this.txBx10.Location = new System.Drawing.Point(64, 442);
this.txBx10.Name = "txBx10";
this.txBx10.Size = new System.Drawing.Size(82, 20);
```

```
this.txBx10.TabIndex = 79;
this.txBx10.Tag = "ksiwb";
this.txBx10.Text = "0.85";
//
// pcBx10
//
this.pcBx10.Image = global::inputData.Properties.Resources.ksiwb;
this.pcBx10.Location = new System.Drawing.Point(18, 432);
this.pcBx10.Name = "pcBx10";
this.pcBx10.Size = new System.Drawing.Size(40, 40);
this.pcBx10.TabIndex = 78;
this.pcBx10.TabStop = false;
//
// txBx9
//
this.txBx9.Location = new System.Drawing.Point(64, 396);
this.txBx9.Name = "txBx9";
this.txBx9.Size = new System.Drawing.Size(82, 20);
this.txBx9.TabIndex = 77;
this.txBx9.Tag = "ksiw";
this.txBx9.Text = "0.5";
//
// pcBx9
//
this.pcBx9.Image = global::inputData.Properties.Resources.ksiw;
this.pcBx9.Location = new System.Drawing.Point(18, 386);
this.pcBx9.Name = "pcBx9";
this.pcBx9.Size = new System.Drawing.Size(40, 40);
this.pcBx9.TabIndex = 76;
this.pcBx9.TabStop = false;
```

```
//  
// txBx8  
//  
this.txBx8.Location = new System.Drawing.Point(64, 350);  
this.txBx8.Name = "txBx8";  
this.txBx8.Size = new System.Drawing.Size(82, 20);  
this.txBx8.TabIndex = 75;  
this.txBx8.Tag = "ksibb";  
this.txBx8.Text = "0.3";  
//  
// pcBx8  
//  
this.pcBx8.Image = global::inputData.Properties.Resources.ksiBb;  
this.pcBx8.Location = new System.Drawing.Point(18, 340);  
this.pcBx8.Name = "pcBx8";  
this.pcBx8.Size = new System.Drawing.Size(40, 40);  
this.pcBx8.TabIndex = 74;  
this.pcBx8.TabStop = false;  
//  
// txBx7  
//  
this.txBx7.Location = new System.Drawing.Point(64, 304);  
this.txBx7.Name = "txBx7";  
this.txBx7.Size = new System.Drawing.Size(82, 20);  
this.txBx7.TabIndex = 73;  
this.txBx7.Tag = "ksib";  
this.txBx7.Text = "0.3";  
//  
// pcBx7  
//
```



```
this.pcBx7.Image = global::inputData.Properties.Resources.ksiB;
this.pcBx7.Location = new System.Drawing.Point(18, 294);
this.pcBx7.Name = "pcBx7";
this.pcBx7.Size = new System.Drawing.Size(40, 40);
this.pcBx7.TabIndex = 72;
this.pcBx7.TabStop = false;
//
// txBx6
//
this.txBx6.Location = new System.Drawing.Point(64, 258);
this.txBx6.Name = "txBx6";
this.txBx6.Size = new System.Drawing.Size(82, 20);
this.txBx6.TabIndex = 71;
this.txBx6.Tag = "Cb";
this.txBx6.Text = "0.64";
//
// pcBx6
//
this.pcBx6.Image = global::inputData.Properties.Resources.Cb;
this.pcBx6.Location = new System.Drawing.Point(18, 248);
this.pcBx6.Name = "pcBx6";
this.pcBx6.Size = new System.Drawing.Size(40, 40);
this.pcBx6.TabIndex = 70;
this.pcBx6.TabStop = false;
//
// txBx5
//
this.txBx5.Location = new System.Drawing.Point(64, 212);
this.txBx5.Name = "txBx5";
this.txBx5.Size = new System.Drawing.Size(82, 20);
```

```
this.txBx5.TabIndex = 69;
this.txBx5.Tag = "BB";
this.txBx5.Text = "17";
//
// pcBx5
//
this.pcBx5.Image = global::inputData.Properties.Resources.BB;
this.pcBx5.Location = new System.Drawing.Point(18, 202);
this.pcBx5.Name = "pcBx5";
this.pcBx5.Size = new System.Drawing.Size(40, 40);
this.pcBx5.TabIndex = 68;
this.pcBx5.TabStop = false;
//
// txBx4
//
this.txBx4.Location = new System.Drawing.Point(64, 166);
this.txBx4.Name = "txBx4";
this.txBx4.Size = new System.Drawing.Size(82, 20);
this.txBx4.TabIndex = 67;
this.txBx4.Tag = "alfa66";
this.txBx4.Text = "0.092";
//
// pcBx4
//
this.pcBx4.Image = global::inputData.Properties.Resources.alfa66;
this.pcBx4.Location = new System.Drawing.Point(18, 156);
this.pcBx4.Name = "pcBx4";
this.pcBx4.Size = new System.Drawing.Size(40, 40);
this.pcBx4.TabIndex = 66;
this.pcBx4.TabStop = false;
```

```
//  
// txBx3  
//  
this.txBx3.Location = new System.Drawing.Point(64, 120);  
this.txBx3.Name = "txBx3";  
this.txBx3.Size = new System.Drawing.Size(82, 20);  
this.txBx3.TabIndex = 65;  
this.txBx3.Tag = "alfa22";  
this.txBx3.Text = "0.092";  
//  
// pcBx3  
//  
this.pcBx3.Image = global::inputData.Properties.Resources.alfa22;  
this.pcBx3.Location = new System.Drawing.Point(18, 110);  
this.pcBx3.Name = "pcBx3";  
this.pcBx3.Size = new System.Drawing.Size(40, 40);  
this.pcBx3.TabIndex = 64;  
this.pcBx3.TabStop = false;  
//  
// txBx2  
//  
this.txBx2.Location = new System.Drawing.Point(64, 74);  
this.txBx2.Name = "txBx2";  
this.txBx2.Size = new System.Drawing.Size(82, 20);  
this.txBx2.TabIndex = 63;  
this.txBx2.Tag = "alfa11";  
this.txBx2.Text = "0.092";  
//  
// pcBx2  
//
```

```
this.pcBx2.Image = global::inputData.Properties.Resources.alfa11;
this.pcBx2.Location = new System.Drawing.Point(18, 64);
this.pcBx2.Name = "pcBx2";
this.pcBx2.Size = new System.Drawing.Size(40, 40);
this.pcBx2.TabIndex = 62;
this.pcBx2.TabStop = false;
//
// txBx1
//
this.txBx1.Location = new System.Drawing.Point(64, 28);
this.txBx1.Name = "txBx1";
this.txBx1.Size = new System.Drawing.Size(82, 20);
this.txBx1.TabIndex = 61;
this.txBx1.Tag = "V0";
this.txBx1.Text = "5.144";
//
// pcBx1
//
this.pcBx1.Image = global::inputData.Properties.Resources.V0;
this.pcBx1.Location = new System.Drawing.Point(18, 18);
this.pcBx1.Name = "pcBx1";
this.pcBx1.Size = new System.Drawing.Size(40, 40);
this.pcBx1.TabIndex = 60;
this.pcBx1.TabStop = false;
//
// tbPg2
//
this.tbPg2.Controls.Add(this.txBx51);
this.tbPg2.Controls.Add(this.txBx50);
this.tbPg2.Controls.Add(this.txBx40);
```

this.tbPg2.Controls.Add(this.txBx41);  
this.tbPg2.Controls.Add(this.txBx42);  
this.tbPg2.Controls.Add(this.txBx43);  
this.tbPg2.Controls.Add(this.txBx44);  
this.tbPg2.Controls.Add(this.txBx45);  
this.tbPg2.Controls.Add(this.txBx46);  
this.tbPg2.Controls.Add(this.txBx47);  
this.tbPg2.Controls.Add(this.txBx48);  
this.tbPg2.Controls.Add(this.txBx49);  
this.tbPg2.Controls.Add(this.txBx39);  
this.tbPg2.Controls.Add(this.txBx38);  
this.tbPg2.Controls.Add(this.txBx37);  
this.tbPg2.Controls.Add(this.txBx36);  
this.tbPg2.Controls.Add(this.txBx35);  
this.tbPg2.Controls.Add(this.txBx34);  
this.tbPg2.Controls.Add(this.txBx33);  
this.tbPg2.Controls.Add(this.txBx32);  
this.tbPg2.Controls.Add(this.txBx31);  
this.tbPg2.Controls.Add(this.txBx30);  
this.tbPg2.Controls.Add(this.pcBx51);  
this.tbPg2.Controls.Add(this.pcBx50);  
this.tbPg2.Controls.Add(this.pcBx40);  
this.tbPg2.Controls.Add(this.pcBx41);  
this.tbPg2.Controls.Add(this.pcBx42);  
this.tbPg2.Controls.Add(this.pcBx43);  
this.tbPg2.Controls.Add(this.pcBx44);  
this.tbPg2.Controls.Add(this.pcBx45);  
this.tbPg2.Controls.Add(this.pcBx46);  
this.tbPg2.Controls.Add(this.pcBx47);  
this.tbPg2.Controls.Add(this.pcBx48);

```
this.tbPg2.Controls.Add(this.pcBx49);
this.tbPg2.Controls.Add(this.pcBx39);
this.tbPg2.Controls.Add(this.pcBx38);
this.tbPg2.Controls.Add(this.pcBx37);
this.tbPg2.Controls.Add(this.pcBx36);
this.tbPg2.Controls.Add(this.pcBx35);
this.tbPg2.Controls.Add(this.pcBx34);
this.tbPg2.Controls.Add(this.pcBx33);
this.tbPg2.Controls.Add(this.pcBx32);
this.tbPg2.Controls.Add(this.pcBx31);
this.tbPg2.Controls.Add(this.pcBx30);
this.tbPg2.Location = new System.Drawing.Point(4, 22);
this.tbPg2.Name = "tbPg2";
this.tbPg2.Padding = new SWF.Padding(3);
this.tbPg2.Size = new System.Drawing.Size(657, 484);
this.tbPg2.TabIndex = 1;
this.tbPg2.Text = "2";
this.tbPg2.UseVisualStyleBackColor = true;
//
// txBx51
//
this.txBx51.Location = new System.Drawing.Point(544, 74);
this.txBx51.Name = "txBx51";
this.txBx51.Size = new System.Drawing.Size(82, 20);
this.txBx51.TabIndex = 105;
this.txBx51.Tag = "css1";
this.txBx51.Text = "-0.406";
//
// txBx50
//
```

```
this.txBx50.Location = new System.Drawing.Point(544, 28);
this.txBx50.Name = "txBx50";
this.txBx50.Size = new System.Drawing.Size(82, 20);
this.txBx50.TabIndex = 103;
this.txBx50.Tag = "bss1";
this.txBx50.Text = "3.62";
//
// txBx40
//
this.txBx40.Location = new System.Drawing.Point(304, 442);
this.txBx40.Name = "txBx40";
this.txBx40.Size = new System.Drawing.Size(82, 20);
this.txBx40.TabIndex = 101;
this.txBx40.Tag = "ass1";
this.txBx40.Text = "-0.559";
//
// txBx41
//
this.txBx41.Location = new System.Drawing.Point(304, 396);
this.txBx41.Name = "txBx41";
this.txBx41.Size = new System.Drawing.Size(82, 20);
this.txBx41.TabIndex = 99;
this.txBx41.Tag = "cs3";
this.txBx41.Text = "0.555";
//
// txBx42
//
this.txBx42.Location = new System.Drawing.Point(304, 350);
this.txBx42.Name = "txBx42";
this.txBx42.Size = new System.Drawing.Size(82, 20);
```

```
this.txBx42.TabIndex = 97;
this.txBx42.Tag = "cs2";
this.txBx42.Text = "0.545";
//
// txBx43
//
this.txBx43.Location = new System.Drawing.Point(304, 304);
this.txBx43.Name = "txBx43";
this.txBx43.Size = new System.Drawing.Size(82, 20);
this.txBx43.TabIndex = 95;
this.txBx43.Tag = "cs1";
this.txBx43.Text = "2.35";
//
// txBx44
//
this.txBx44.Location = new System.Drawing.Point(304, 258);
this.txBx44.Name = "txBx44";
this.txBx44.Size = new System.Drawing.Size(82, 20);
this.txBx44.TabIndex = 93;
this.txBx44.Tag = "bs3";
this.txBx44.Text = "-2.43";
//
// txBx45
//
this.txBx45.Location = new System.Drawing.Point(304, 212);
this.txBx45.Name = "txBx45";
this.txBx45.Size = new System.Drawing.Size(82, 20);
this.txBx45.TabIndex = 91;
this.txBx45.Tag = "bs2";
this.txBx45.Text = "-2.254";
```



```
//  
// txBx46  
//  
this.txBx46.Location = new System.Drawing.Point(304, 166);  
this.txBx46.Name = "txBx46";  
this.txBx46.Size = new System.Drawing.Size(82, 20);  
this.txBx46.TabIndex = 89;  
this.txBx46.Tag = "bs1";  
this.txBx46.Text = "-6.8";  
//  
// txBx47  
//  
this.txBx47.Location = new System.Drawing.Point(304, 120);  
this.txBx47.Name = "txBx47";  
this.txBx47.Size = new System.Drawing.Size(82, 20);  
this.txBx47.TabIndex = 87;  
this.txBx47.Tag = "as3";  
this.txBx47.Text = "4.04";  
//  
// txBx48  
//  
this.txBx48.Location = new System.Drawing.Point(304, 74);  
this.txBx48.Name = "txBx48";  
this.txBx48.Size = new System.Drawing.Size(82, 20);  
this.txBx48.TabIndex = 85;  
this.txBx48.Tag = "as2";  
this.txBx48.Text = "3.756";  
//  
// txBx49  
//
```

```
this.txBx49.Location = new System.Drawing.Point(304, 28);
this.txBx49.Name = "txBx49";
this.txBx49.Size = new System.Drawing.Size(82, 20);
this.txBx49.TabIndex = 83;
this.txBx49.Tag = "as1";
this.txBx49.Text = "10";
//
// txBx39
//
this.txBx39.Location = new System.Drawing.Point(64, 442);
this.txBx39.Name = "txBx39";
this.txBx39.Size = new System.Drawing.Size(82, 20);
this.txBx39.TabIndex = 81;
this.txBx39.Tag = "d";
this.txBx39.Text = "1";
//
// txBx38
//
this.txBx38.Location = new System.Drawing.Point(64, 396);
this.txBx38.Name = "txBx38";
this.txBx38.Size = new System.Drawing.Size(82, 20);
this.txBx38.TabIndex = 79;
this.txBx38.Tag = "c3";
this.txBx38.Text = "0.516";
//
// txBx37
//
this.txBx37.Location = new System.Drawing.Point(64, 350);
this.txBx37.Name = "txBx37";
this.txBx37.Size = new System.Drawing.Size(82, 20);
```

```
this.txBx37.TabIndex = 77;
this.txBx37.Tag = "c2";
this.txBx37.Text = "0.479";
//
// txBx36
//
this.txBx36.Location = new System.Drawing.Point(64, 304);
this.txBx36.Name = "txBx36";
this.txBx36.Size = new System.Drawing.Size(82, 20);
this.txBx36.TabIndex = 75;
this.txBx36.Tag = "c1";
this.txBx36.Text = "0.541";
//
// txBx35
//
this.txBx35.Location = new System.Drawing.Point(64, 258);
this.txBx35.Name = "txBx35";
this.txBx35.Size = new System.Drawing.Size(82, 20);
this.txBx35.TabIndex = 73;
this.txBx35.Tag = "b3";
this.txBx35.Text = "-2.399";
//
// txBx34
//
this.txBx34.Location = new System.Drawing.Point(64, 212);
this.txBx34.Name = "txBx34";
this.txBx34.Size = new System.Drawing.Size(82, 20);
this.txBx34.TabIndex = 71;
this.txBx34.Tag = "b2";
this.txBx34.Text = "-2.024";
```

```
//  
// txBx33  
//  
this.txBx33.Location = new System.Drawing.Point(64, 166);  
this.txBx33.Name = "txBx33";  
this.txBx33.Size = new System.Drawing.Size(82, 20);  
this.txBx33.TabIndex = 69;  
this.txBx33.Tag = "b1";  
this.txBx33.Text = "-1.78";  
//  
// txBx32  
//  
this.txBx32.Location = new System.Drawing.Point(64, 120);  
this.txBx32.Name = "txBx32";  
this.txBx32.Size = new System.Drawing.Size(82, 20);  
this.txBx32.TabIndex = 67;  
this.txBx32.Tag = "a3";  
this.txBx32.Text = "3.964";  
//  
// txBx31  
//  
this.txBx31.Location = new System.Drawing.Point(64, 74);  
this.txBx31.Name = "txBx31";  
this.txBx31.Size = new System.Drawing.Size(82, 20);  
this.txBx31.TabIndex = 65;  
this.txBx31.Tag = "a2";  
this.txBx31.Text = "3.442";  
//  
// txBx30  
//
```

```
this.txBx30.Location = new System.Drawing.Point(64, 28);
this.txBx30.Name = "txBx30";
this.txBx30.Size = new System.Drawing.Size(82, 20);
this.txBx30.TabIndex = 63;
this.txBx30.Tag = "a1";
this.txBx30.Text = "5.88";
//
// pcBx51
//
this.pcBx51.Image =
((System.Drawing.Image)(resources.GetObject("pcBx51.Image")));
this.pcBx51.Location = new System.Drawing.Point(498, 64);
this.pcBx51.Name = "pcBx51";
this.pcBx51.Size = new System.Drawing.Size(40, 40);
this.pcBx51.TabIndex = 104;
this.pcBx51.TabStop = false;
//
// pcBx50
//
this.pcBx50.Image =
((System.Drawing.Image)(resources.GetObject("pcBx50.Image")));
this.pcBx50.Location = new System.Drawing.Point(498, 18);
this.pcBx50.Name = "pcBx50";
this.pcBx50.Size = new System.Drawing.Size(40, 40);
this.pcBx50.TabIndex = 102;
this.pcBx50.TabStop = false;
//
// pcBx40
//
this.pcBx40.Image =
```

```
((System.Drawing.Image)(resources.GetObject("pcBx40.Image")));
    this.pcBx40.Location = new System.Drawing.Point(258, 432);
    this.pcBx40.Name = "pcBx40";
    this.pcBx40.Size = new System.Drawing.Size(40, 40);
    this.pcBx40.TabIndex = 100;
    this.pcBx40.TabStop = false;
    //
    // pcBx41
    //
    this.pcBx41.Image =
((System.Drawing.Image)(resources.GetObject("pcBx41.Image")));
    this.pcBx41.Location = new System.Drawing.Point(258, 386);
    this.pcBx41.Name = "pcBx41";
    this.pcBx41.Size = new System.Drawing.Size(40, 40);
    this.pcBx41.TabIndex = 98;
    this.pcBx41.TabStop = false;
    //
    // pcBx42
    //
    this.pcBx42.Image =
((System.Drawing.Image)(resources.GetObject("pcBx42.Image")));
    this.pcBx42.Location = new System.Drawing.Point(258, 340);
    this.pcBx42.Name = "pcBx42";
    this.pcBx42.Size = new System.Drawing.Size(40, 40);
    this.pcBx42.TabIndex = 96;
    this.pcBx42.TabStop = false;
    //
    // pcBx43
    //
    this.pcBx43.Image =
```

```
((System.Drawing.Image)(resources.GetObject("pcBx43.Image")));
    this.pcBx43.Location = new System.Drawing.Point(258, 294);
    this.pcBx43.Name = "pcBx43";
    this.pcBx43.Size = new System.Drawing.Size(40, 40);
    this.pcBx43.TabIndex = 94;
    this.pcBx43.TabStop = false;
    //
    // pcBx44
    //
    this.pcBx44.Image =
((System.Drawing.Image)(resources.GetObject("pcBx44.Image")));
    this.pcBx44.Location = new System.Drawing.Point(258, 248);
    this.pcBx44.Name = "pcBx44";
    this.pcBx44.Size = new System.Drawing.Size(40, 40);
    this.pcBx44.TabIndex = 92;
    this.pcBx44.TabStop = false;
    //
    // pcBx45
    //
    this.pcBx45.Image =
((System.Drawing.Image)(resources.GetObject("pcBx45.Image")));
    this.pcBx45.Location = new System.Drawing.Point(258, 202);
    this.pcBx45.Name = "pcBx45";
    this.pcBx45.Size = new System.Drawing.Size(40, 40);
    this.pcBx45.TabIndex = 90;
    this.pcBx45.TabStop = false;
    //
    // pcBx46
    //
    this.pcBx46.Image =
```

```
((System.Drawing.Image)(resources.GetObject("pcBx46.Image")));
    this.pcBx46.Location = new System.Drawing.Point(258, 156);
    this.pcBx46.Name = "pcBx46";
    this.pcBx46.Size = new System.Drawing.Size(40, 40);
    this.pcBx46.TabIndex = 88;
    this.pcBx46.TabStop = false;
    //
    // pcBx47
    //
    this.pcBx47.Image =
((System.Drawing.Image)(resources.GetObject("pcBx47.Image")));
    this.pcBx47.Location = new System.Drawing.Point(258, 110);
    this.pcBx47.Name = "pcBx47";
    this.pcBx47.Size = new System.Drawing.Size(40, 40);
    this.pcBx47.TabIndex = 86;
    this.pcBx47.TabStop = false;
    //
    // pcBx48
    //
    this.pcBx48.Image =
((System.Drawing.Image)(resources.GetObject("pcBx48.Image")));
    this.pcBx48.Location = new System.Drawing.Point(258, 64);
    this.pcBx48.Name = "pcBx48";
    this.pcBx48.Size = new System.Drawing.Size(40, 40);
    this.pcBx48.TabIndex = 84;
    this.pcBx48.TabStop = false;
    //
    // pcBx49
    //
    this.pcBx49.Image =
```



```
((System.Drawing.Image)(resources.GetObject("pcBx49.Image")));
    this.pcBx49.Location = new System.Drawing.Point(258, 18);
    this.pcBx49.Name = "pcBx49";
    this.pcBx49.Size = new System.Drawing.Size(40, 40);
    this.pcBx49.TabIndex = 82;
    this.pcBx49.TabStop = false;
    //
    // pcBx39
    //
    this.pcBx39.Image =
((System.Drawing.Image)(resources.GetObject("pcBx39.Image")));
    this.pcBx39.Location = new System.Drawing.Point(18, 432);
    this.pcBx39.Name = "pcBx39";
    this.pcBx39.Size = new System.Drawing.Size(40, 40);
    this.pcBx39.TabIndex = 80;
    this.pcBx39.TabStop = false;
    //
    // pcBx38
    //
    this.pcBx38.Image =
((System.Drawing.Image)(resources.GetObject("pcBx38.Image")));
    this.pcBx38.Location = new System.Drawing.Point(18, 386);
    this.pcBx38.Name = "pcBx38";
    this.pcBx38.Size = new System.Drawing.Size(40, 40);
    this.pcBx38.TabIndex = 78;
    this.pcBx38.TabStop = false;
    //
    // pcBx37
    //
    this.pcBx37.Image =
```

```
((System.Drawing.Image)(resources.GetObject("pcBx37.Image")));
    this.pcBx37.Location = new System.Drawing.Point(18, 340);
    this.pcBx37.Name = "pcBx37";
    this.pcBx37.Size = new System.Drawing.Size(40, 40);
    this.pcBx37.TabIndex = 76;
    this.pcBx37.TabStop = false;
    //
    // pcBx36
    //
    this.pcBx36.Image =
((System.Drawing.Image)(resources.GetObject("pcBx36.Image")));
    this.pcBx36.Location = new System.Drawing.Point(18, 294);
    this.pcBx36.Name = "pcBx36";
    this.pcBx36.Size = new System.Drawing.Size(40, 40);
    this.pcBx36.TabIndex = 74;
    this.pcBx36.TabStop = false;
    //
    // pcBx35
    //
    this.pcBx35.Image =
((System.Drawing.Image)(resources.GetObject("pcBx35.Image")));
    this.pcBx35.Location = new System.Drawing.Point(18, 248);
    this.pcBx35.Name = "pcBx35";
    this.pcBx35.Size = new System.Drawing.Size(40, 40);
    this.pcBx35.TabIndex = 72;
    this.pcBx35.TabStop = false;
    //
    // pcBx34
    //
    this.pcBx34.Image =
```

```
((System.Drawing.Image)(resources.GetObject("pcBx34.Image")));
    this.pcBx34.Location = new System.Drawing.Point(18, 202);
    this.pcBx34.Name = "pcBx34";
    this.pcBx34.Size = new System.Drawing.Size(40, 40);
    this.pcBx34.TabIndex = 70;
    this.pcBx34.TabStop = false;
    //
    // pcBx33
    //
    this.pcBx33.Image =
((System.Drawing.Image)(resources.GetObject("pcBx33.Image")));
    this.pcBx33.Location = new System.Drawing.Point(18, 156);
    this.pcBx33.Name = "pcBx33";
    this.pcBx33.Size = new System.Drawing.Size(40, 40);
    this.pcBx33.TabIndex = 68;
    this.pcBx33.TabStop = false;
    //
    // pcBx32
    //
    this.pcBx32.Image =
((System.Drawing.Image)(resources.GetObject("pcBx32.Image")));
    this.pcBx32.Location = new System.Drawing.Point(18, 110);
    this.pcBx32.Name = "pcBx32";
    this.pcBx32.Size = new System.Drawing.Size(40, 40);
    this.pcBx32.TabIndex = 66;
    this.pcBx32.TabStop = false;
    //
    // pcBx31
    //
    this.pcBx31.Image =
```

```
((System.Drawing.Image)(resources.GetObject("pcBx31.Image")));
    this.pcBx31.Location = new System.Drawing.Point(18, 64);
    this.pcBx31.Name = "pcBx31";
    this.pcBx31.Size = new System.Drawing.Size(40, 40);
    this.pcBx31.TabIndex = 64;
    this.pcBx31.TabStop = false;
    //
    // pcBx30
    //
    this.pcBx30.Image =
((System.Drawing.Image)(resources.GetObject("pcBx30.Image")));
    this.pcBx30.Location = new System.Drawing.Point(18, 18);
    this.pcBx30.Name = "pcBx30";
    this.pcBx30.Size = new System.Drawing.Size(40, 40);
    this.pcBx30.TabIndex = 62;
    this.pcBx30.TabStop = false;
    //
    // tbPg3
    //
    this.tbPg3.Controls.Add(this.txBx72);
    this.tbPg3.Controls.Add(this.pcBx72);
    this.tbPg3.Controls.Add(this.txBx73);
    this.tbPg3.Controls.Add(this.pcBx73);
    this.tbPg3.Controls.Add(this.txBx74);
    this.tbPg3.Controls.Add(this.pcBx74);
    this.tbPg3.Controls.Add(this.txBx52);
    this.tbPg3.Controls.Add(this.pcBx52);
    this.tbPg3.Controls.Add(this.txBx53);
    this.tbPg3.Controls.Add(this.pcBx53);
    this.tbPg3.Controls.Add(this.txBx54);
```

```
this.tbPg3.Controls.Add(this.pcBx54);
this.tbPg3.Controls.Add(this.txBx55);
this.tbPg3.Controls.Add(this.pcBx55);
this.tbPg3.Controls.Add(this.txBx56);
this.tbPg3.Controls.Add(this.pcBx56);
this.tbPg3.Controls.Add(this.txBx57);
this.tbPg3.Controls.Add(this.pcBx57);
this.tbPg3.Controls.Add(this.txBx58);
this.tbPg3.Controls.Add(this.pcBx58);
this.tbPg3.Controls.Add(this.txBx59);
this.tbPg3.Controls.Add(this.pcBx59);
this.tbPg3.Controls.Add(this.txBx60);
this.tbPg3.Controls.Add(this.pcBx60);
this.tbPg3.Controls.Add(this.txBx61);
this.tbPg3.Controls.Add(this.pcBx61);
this.tbPg3.Controls.Add(this.txBx62);
this.tbPg3.Controls.Add(this.pcBx62);
this.tbPg3.Controls.Add(this.txBx63);
this.tbPg3.Controls.Add(this.pcBx63);
this.tbPg3.Controls.Add(this.txBx64);
this.tbPg3.Controls.Add(this.pcBx64);
this.tbPg3.Controls.Add(this.txBx65);
this.tbPg3.Controls.Add(this.pcBx65);
this.tbPg3.Controls.Add(this.txBx66);
this.tbPg3.Controls.Add(this.pcBx66);
this.tbPg3.Controls.Add(this.txBx67);
this.tbPg3.Controls.Add(this.pcBx67);
this.tbPg3.Controls.Add(this.txBx68);
this.tbPg3.Controls.Add(this.pcBx68);
this.tbPg3.Controls.Add(this.txBx69);
```

```
this.tbPg3.Controls.Add(this.pcBx69);
this.tbPg3.Controls.Add(this.txBx70);
this.tbPg3.Controls.Add(this.pcBx70);
this.tbPg3.Controls.Add(this.txBx71);
this.tbPg3.Controls.Add(this.pcBx71);
this.tbPg3.Location = new System.Drawing.Point(4, 22);
this.tbPg3.Name = "tbPg3";
this.tbPg3.Padding = new SWF.Padding(3);
this.tbPg3.Size = new System.Drawing.Size(657, 484);
this.tbPg3.TabIndex = 2;
this.tbPg3.Text = "3";
this.tbPg3.UseVisualStyleBackColor = true;
//
// txBx72
//
this.txBx72.Location = new System.Drawing.Point(544, 120);
this.txBx72.Name = "txBx72";
this.txBx72.Size = new System.Drawing.Size(82, 20);
this.txBx72.TabIndex = 147;
this.txBx72.Tag = "Sk";
this.txBx72.Text = "13.3";
//
// pcBx72
//
this.pcBx72.Image = global::inputData.Properties.Resources.Sk;
this.pcBx72.Location = new System.Drawing.Point(498, 110);
this.pcBx72.Name = "pcBx72";
this.pcBx72.Size = new System.Drawing.Size(40, 40);
this.pcBx72.TabIndex = 146;
this.pcBx72.TabStop = false;
```

```
//  
// txBx73  
//  
this.txBx73.Location = new System.Drawing.Point(544, 74);  
this.txBx73.Name = "txBx73";  
this.txBx73.Size = new System.Drawing.Size(82, 20);  
this.txBx73.TabIndex = 145;  
this.txBx73.Tag = "ask";  
this.txBx73.Text = "1.096";  
//  
// pcBx73  
//  
this.pcBx73.Image = global::inputData.Properties.Resources.ask;  
this.pcBx73.Location = new System.Drawing.Point(498, 64);  
this.pcBx73.Name = "pcBx73";  
this.pcBx73.Size = new System.Drawing.Size(40, 40);  
this.pcBx73.TabIndex = 144;  
this.pcBx73.TabStop = false;  
//  
// txBx74  
//  
this.txBx74.Location = new System.Drawing.Point(544, 28);  
this.txBx74.Name = "txBx74";  
this.txBx74.Size = new System.Drawing.Size(82, 20);  
this.txBx74.TabIndex = 143;  
this.txBx74.Tag = "ak";  
this.txBx74.Text = "0.58";  
//  
// pcBx74  
//
```

```
this.pcBx74.Image = global::inputData.Properties.Resources.ak;
this.pcBx74.Location = new System.Drawing.Point(498, 18);
this.pcBx74.Name = "pcBx74";
this.pcBx74.Size = new System.Drawing.Size(40, 40);
this.pcBx74.TabIndex = 142;
this.pcBx74.TabStop = false;
//
// txBx52
//
this.txBx52.Location = new System.Drawing.Point(304, 442);
this.txBx52.Name = "txBx52";
this.txBx52.Size = new System.Drawing.Size(82, 20);
this.txBx52.TabIndex = 141;
this.txBx52.Tag = "B8";
this.txBx52.Text = "1";
//
// pcBx52
//
this.pcBx52.Image = global::inputData.Properties.Resources.BB8;
this.pcBx52.Location = new System.Drawing.Point(258, 432);
this.pcBx52.Name = "pcBx52";
this.pcBx52.Size = new System.Drawing.Size(40, 40);
this.pcBx52.TabIndex = 140;
this.pcBx52.TabStop = false;
//
// txBx53
//
this.txBx53.Location = new System.Drawing.Point(304, 396);
this.txBx53.Name = "txBx53";
this.txBx53.Size = new System.Drawing.Size(82, 20);
```



```
this.txBx53.TabIndex = 139;
this.txBx53.Tag = "B7";
this.txBx53.Text = "1.89";
//
// pcBx53
//
this.pcBx53.Image = global::inputData.Properties.Resources.BB7;
this.pcBx53.Location = new System.Drawing.Point(258, 386);
this.pcBx53.Name = "pcBx53";
this.pcBx53.Size = new System.Drawing.Size(40, 40);
this.pcBx53.TabIndex = 138;
this.pcBx53.TabStop = false;
//
// txBx54
//
this.txBx54.Location = new System.Drawing.Point(304, 350);
this.txBx54.Name = "txBx54";
this.txBx54.Size = new System.Drawing.Size(82, 20);
this.txBx54.TabIndex = 137;
this.txBx54.Tag = "B6";
this.txBx54.Text = "0.56";
//
// pcBx54
//
this.pcBx54.Image = global::inputData.Properties.Resources.BB6;
this.pcBx54.Location = new System.Drawing.Point(258, 340);
this.pcBx54.Name = "pcBx54";
this.pcBx54.Size = new System.Drawing.Size(40, 40);
this.pcBx54.TabIndex = 136;
this.pcBx54.TabStop = false;
```

```
//  
// txBx55  
//  
this.txBx55.Location = new System.Drawing.Point(304, 304);  
this.txBx55.Name = "txBx55";  
this.txBx55.Size = new System.Drawing.Size(82, 20);  
this.txBx55.TabIndex = 135;  
this.txBx55.Tag = "B5";  
this.txBx55.Text = "2.22";  
//  
// pcBx55  
//  
this.pcBx55.Image = global::inputData.Properties.Resources.BB5;  
this.pcBx55.Location = new System.Drawing.Point(258, 294);  
this.pcBx55.Name = "pcBx55";  
this.pcBx55.Size = new System.Drawing.Size(40, 40);  
this.pcBx55.TabIndex = 134;  
this.pcBx55.TabStop = false;  
//  
// txBx56  
//  
this.txBx56.Location = new System.Drawing.Point(304, 258);  
this.txBx56.Name = "txBx56";  
this.txBx56.Size = new System.Drawing.Size(82, 20);  
this.txBx56.TabIndex = 133;  
this.txBx56.Tag = "B4";  
this.txBx56.Text = "1.01";  
//  
// pcBx56  
//
```

```
this.pcBx56.Image = global::inputData.Properties.Resources.BB4;
this.pcBx56.Location = new System.Drawing.Point(258, 248);
this.pcBx56.Name = "pcBx56";
this.pcBx56.Size = new System.Drawing.Size(40, 40);
this.pcBx56.TabIndex = 132;
this.pcBx56.TabStop = false;
//
// txBx57
//
this.txBx57.Location = new System.Drawing.Point(304, 212);
this.txBx57.Name = "txBx57";
this.txBx57.Size = new System.Drawing.Size(82, 20);
this.txBx57.TabIndex = 131;
this.txBx57.Tag = "B3";
this.txBx57.Text = "0.917";
//
// pcBx57
//
this.pcBx57.Image = global::inputData.Properties.Resources.BB3;
this.pcBx57.Location = new System.Drawing.Point(258, 202);
this.pcBx57.Name = "pcBx57";
this.pcBx57.Size = new System.Drawing.Size(40, 40);
this.pcBx57.TabIndex = 130;
this.pcBx57.TabStop = false;
//
// txBx58
//
this.txBx58.Location = new System.Drawing.Point(304, 166);
this.txBx58.Name = "txBx58";
this.txBx58.Size = new System.Drawing.Size(82, 20);
```

```
this.txBx58.TabIndex = 129;
this.txBx58.Tag = "B2";
this.txBx58.Text = "-0.524";
//
// pcBx58
//
this.pcBx58.Image = global::inputData.Properties.Resources.BB2;
this.pcBx58.Location = new System.Drawing.Point(258, 156);
this.pcBx58.Name = "pcBx58";
this.pcBx58.Size = new System.Drawing.Size(40, 40);
this.pcBx58.TabIndex = 128;
this.pcBx58.TabStop = false;
//
// txBx59
//
this.txBx59.Location = new System.Drawing.Point(304, 120);
this.txBx59.Name = "txBx59";
this.txBx59.Size = new System.Drawing.Size(82, 20);
this.txBx59.TabIndex = 127;
this.txBx59.Tag = "B1";
this.txBx59.Text = "6.39";
//
// pcBx59
//
this.pcBx59.Image = global::inputData.Properties.Resources.BB1;
this.pcBx59.Location = new System.Drawing.Point(258, 110);
this.pcBx59.Name = "pcBx59";
this.pcBx59.Size = new System.Drawing.Size(40, 40);
this.pcBx59.TabIndex = 126;
this.pcBx59.TabStop = false;
```

```
//  
// txBx60  
//  
this.txBx60.Location = new System.Drawing.Point(304, 74);  
this.txBx60.Name = "txBx60";  
this.txBx60.Size = new System.Drawing.Size(82, 20);  
this.txBx60.TabIndex = 125;  
this.txBx60.Tag = "A12";  
this.txBx60.Text = "0.287";  
//  
// pcBx60  
//  
this.pcBx60.Image = global::inputData.Properties.Resources.AA12;  
this.pcBx60.Location = new System.Drawing.Point(258, 64);  
this.pcBx60.Name = "pcBx60";  
this.pcBx60.Size = new System.Drawing.Size(40, 40);  
this.pcBx60.TabIndex = 124;  
this.pcBx60.TabStop = false;  
//  
// txBx61  
//  
this.txBx61.Location = new System.Drawing.Point(304, 28);  
this.txBx61.Name = "txBx61";  
this.txBx61.Size = new System.Drawing.Size(82, 20);  
this.txBx61.TabIndex = 123;  
this.txBx61.Tag = "A11";  
this.txBx61.Text = "0.103";  
//  
// pcBx61  
//
```

```
this.pcBx61.Image = global::inputData.Properties.Resources.AA11;
this.pcBx61.Location = new System.Drawing.Point(258, 18);
this.pcBx61.Name = "pcBx61";
this.pcBx61.Size = new System.Drawing.Size(40, 40);
this.pcBx61.TabIndex = 122;
this.pcBx61.TabStop = false;
//
// txBx62
//
this.txBx62.Location = new System.Drawing.Point(64, 442);
this.txBx62.Name = "txBx62";
this.txBx62.Size = new System.Drawing.Size(82, 20);
this.txBx62.TabIndex = 121;
this.txBx62.Tag = "A10";
this.txBx62.Text = "-1.07";
//
// pcBx62
//
this.pcBx62.Image = global::inputData.Properties.Resources.AA10;
this.pcBx62.Location = new System.Drawing.Point(18, 432);
this.pcBx62.Name = "pcBx62";
this.pcBx62.Size = new System.Drawing.Size(40, 40);
this.pcBx62.TabIndex = 120;
this.pcBx62.TabStop = false;
//
// txBx63
//
this.txBx63.Location = new System.Drawing.Point(64, 396);
this.txBx63.Name = "txBx63";
this.txBx63.Size = new System.Drawing.Size(82, 20);
```

```
this.txBx63.TabIndex = 119;
this.txBx63.Tag = "A9";
this.txBx63.Text = "6.39";
//
// pcBx63
//
this.pcBx63.Image = global::inputData.Properties.Resources.AA9;
this.pcBx63.Location = new System.Drawing.Point(18, 386);
this.pcBx63.Name = "pcBx63";
this.pcBx63.Size = new System.Drawing.Size(40, 40);
this.pcBx63.TabIndex = 118;
this.pcBx63.TabStop = false;
//
// txBx64
//
this.txBx64.Location = new System.Drawing.Point(64, 350);
this.txBx64.Name = "txBx64";
this.txBx64.Size = new System.Drawing.Size(82, 20);
this.txBx64.TabIndex = 117;
this.txBx64.Tag = "A8";
this.txBx64.Text = "0.11";
//
// pcBx64
//
this.pcBx64.Image = global::inputData.Properties.Resources.AA8;
this.pcBx64.Location = new System.Drawing.Point(18, 340);
this.pcBx64.Name = "pcBx64";
this.pcBx64.Size = new System.Drawing.Size(40, 40);
this.pcBx64.TabIndex = 116;
this.pcBx64.TabStop = false;
```

```
//  
// txBx65  
//  
this.txBx65.Location = new System.Drawing.Point(64, 304);  
this.txBx65.Name = "txBx65";  
this.txBx65.Size = new System.Drawing.Size(82, 20);  
this.txBx65.TabIndex = 115;  
this.txBx65.Tag = "A7";  
this.txBx65.Text = "0.033";  
//  
// pcBx65  
//  
this.pcBx65.Image = global::inputData.Properties.Resources.AA7;  
this.pcBx65.Location = new System.Drawing.Point(18, 294);  
this.pcBx65.Name = "pcBx65";  
this.pcBx65.Size = new System.Drawing.Size(40, 40);  
this.pcBx65.TabIndex = 114;  
this.pcBx65.TabStop = false;  
//  
// txBx66  
//  
this.txBx66.Location = new System.Drawing.Point(64, 258);  
this.txBx66.Name = "txBx66";  
this.txBx66.Size = new System.Drawing.Size(82, 20);  
this.txBx66.TabIndex = 113;  
this.txBx66.Tag = "A6";  
this.txBx66.Text = "0.0693";  
//  
// pcBx66  
//
```



```
this.pcBx66.Image = global::inputData.Properties.Resources.AA6;
this.pcBx66.Location = new System.Drawing.Point(18, 248);
this.pcBx66.Name = "pcBx66";
this.pcBx66.Size = new System.Drawing.Size(40, 40);
this.pcBx66.TabIndex = 112;
this.pcBx66.TabStop = false;
//
// txBx67
//
this.txBx67.Location = new System.Drawing.Point(64, 212);
this.txBx67.Name = "txBx67";
this.txBx67.Size = new System.Drawing.Size(82, 20);
this.txBx67.TabIndex = 111;
this.txBx67.Tag = "A5";
this.txBx67.Text = "-0.0208";
//
// pcBx67
//
this.pcBx67.Image = global::inputData.Properties.Resources.AA5;
this.pcBx67.Location = new System.Drawing.Point(18, 202);
this.pcBx67.Name = "pcBx67";
this.pcBx67.Size = new System.Drawing.Size(40, 40);
this.pcBx67.TabIndex = 110;
this.pcBx67.TabStop = false;
//
// txBx68
//
this.txBx68.Location = new System.Drawing.Point(64, 166);
this.txBx68.Name = "txBx68";
this.txBx68.Size = new System.Drawing.Size(82, 20);
```

```
this.txBx68.TabIndex = 109;
this.txBx68.Tag = "A4";
this.txBx68.Text = "0.998";
//
// pcBx68
//
this.pcBx68.Image = global::inputData.Properties.Resources.AA4;
this.pcBx68.Location = new System.Drawing.Point(18, 156);
this.pcBx68.Name = "pcBx68";
this.pcBx68.Size = new System.Drawing.Size(40, 40);
this.pcBx68.TabIndex = 108;
this.pcBx68.TabStop = false;
//
// txBx69
//
this.txBx69.Location = new System.Drawing.Point(64, 120);
this.txBx69.Name = "txBx69";
this.txBx69.Size = new System.Drawing.Size(82, 20);
this.txBx69.TabIndex = 107;
this.txBx69.Tag = "A3";
this.txBx69.Text = "0.22";
//
// pcBx69
//
this.pcBx69.Image = global::inputData.Properties.Resources.AA3;
this.pcBx69.Location = new System.Drawing.Point(18, 110);
this.pcBx69.Name = "pcBx69";
this.pcBx69.Size = new System.Drawing.Size(40, 40);
this.pcBx69.TabIndex = 106;
this.pcBx69.TabStop = false;
```

```
//  
// txBx70  
//  
this.txBx70.Location = new System.Drawing.Point(64, 74);  
this.txBx70.Name = "txBx70";  
this.txBx70.Size = new System.Drawing.Size(82, 20);  
this.txBx70.TabIndex = 105;  
this.txBx70.Tag = "A2";  
this.txBx70.Text = "-0.885";  
//  
// pcBx70  
//  
this.pcBx70.Image = global::inputData.Properties.Resources.AA2;  
this.pcBx70.Location = new System.Drawing.Point(18, 64);  
this.pcBx70.Name = "pcBx70";  
this.pcBx70.Size = new System.Drawing.Size(40, 40);  
this.pcBx70.TabIndex = 104;  
this.pcBx70.TabStop = false;  
//  
// txBx71  
//  
this.txBx71.Location = new System.Drawing.Point(64, 28);  
this.txBx71.Name = "txBx71";  
this.txBx71.Size = new System.Drawing.Size(82, 20);  
this.txBx71.TabIndex = 103;  
this.txBx71.Tag = "A1";  
this.txBx71.Text = "3.01";  
//  
// pcBx71  
//
```

```
this.pcBx71.Image = global::inputData.Properties.Resources.AA1;
this.pcBx71.Location = new System.Drawing.Point(18, 18);
this.pcBx71.Name = "pcBx71";
this.pcBx71.Size = new System.Drawing.Size(40, 40);
this.pcBx71.TabIndex = 102;
this.pcBx71.TabStop = false;
//
// butn1
//
this.butn1.Location = new System.Drawing.Point(12, 528);
this.butn1.Name = "butn1";
this.butn1.Size = new System.Drawing.Size(188, 34);
this.butn1.TabIndex = 61;
this.butn1.Text = "Сохранить";
this.butn1.UseVisualStyleBackColor = true;
this.butn1.Click += new System.EventHandler(this.butn1_Click);
//
// butn2
//
this.butn2.Location = new System.Drawing.Point(555, 528);
this.butn2.Name = "butn2";
this.butn2.Size = new System.Drawing.Size(122, 34);
this.butn2.TabIndex = 62;
this.butn2.Text = "По умолчанию";
this.butn2.UseVisualStyleBackColor = true;
this.butn2.Click += new System.EventHandler(this.butn2_Click);
//
// inFrm
//
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
```

```
this.AutoScaleMode = SWF.AutoScaleMode.Font;
this.BackColor = System.Drawing.SystemColors.Window;
this.ClientSize = new System.Drawing.Size(685, 570);
this.Controls.Add(this.butn2);
this.Controls.Add(this.butn1);
this.Controls.Add(this.tbC11);
this.Name = "inFrm";
this.Text = "inFrm";
this.tbC11.ResumeLayout(false);
this.tbPg1.ResumeLayout(false);
this.tbPg1.PerformLayout();
((SysComIS)(this.pcBx29)).EndInit();
((SysComIS)(this.pcBx28)).EndInit();
((SysComIS)(this.pcBx27)).EndInit();
((SysComIS)(this.pcBx26)).EndInit();
((SysComIS)(this.pcBx25)).EndInit();
((SysComIS)(this.pcBx24)).EndInit();
((SysComIS)(this.pcBx23)).EndInit();
((SysComIS)(this.pcBx22)).EndInit();
((SysComIS)(this.pcBx21)).EndInit();
((SysComIS)(this.pcBx20)).EndInit();
((SysComIS)(this.pcBx19)).EndInit();
((SysComIS)(this.pcBx18)).EndInit();
((SysComIS)(this.pcBx17)).EndInit();
((SysComIS)(this.pcBx16)).EndInit();
((SysComIS)(this.pcBx15)).EndInit();
((SysComIS)(this.pcBx14)).EndInit();
((SysComIS)(this.pcBx13)).EndInit();
((SysComIS)(this.pcBx12)).EndInit();
((SysComIS)(this.pcBx11)).EndInit();
```

```
((SysComIS)(this.pcBx10)).EndInit();
((SysComIS)(this.pcBx9)).EndInit();
((SysComIS)(this.pcBx8)).EndInit();
((SysComIS)(this.pcBx7)).EndInit();
((SysComIS)(this.pcBx6)).EndInit();
((SysComIS)(this.pcBx5)).EndInit();
((SysComIS)(this.pcBx4)).EndInit();
((SysComIS)(this.pcBx3)).EndInit();
((SysComIS)(this.pcBx2)).EndInit();
((SysComIS)(this.pcBx1)).EndInit();
this.tbPg2.ResumeLayout(false);
this.tbPg2.PerformLayout();
((SysComIS)(this.pcBx51)).EndInit();
((SysComIS)(this.pcBx50)).EndInit();
((SysComIS)(this.pcBx40)).EndInit();
((SysComIS)(this.pcBx41)).EndInit();
((SysComIS)(this.pcBx42)).EndInit();
((SysComIS)(this.pcBx43)).EndInit();
((SysComIS)(this.pcBx44)).EndInit();
((SysComIS)(this.pcBx45)).EndInit();
((SysComIS)(this.pcBx46)).EndInit();
((SysComIS)(this.pcBx47)).EndInit();
((SysComIS)(this.pcBx48)).EndInit();
((SysComIS)(this.pcBx49)).EndInit();
((SysComIS)(this.pcBx39)).EndInit();
((SysComIS)(this.pcBx38)).EndInit();
((SysComIS)(this.pcBx37)).EndInit();
((SysComIS)(this.pcBx36)).EndInit();
((SysComIS)(this.pcBx35)).EndInit();
((SysComIS)(this.pcBx34)).EndInit();
```

```
((SysComIS)(this.pcBx33)).EndInit();
((SysComIS)(this.pcBx32)).EndInit();
((SysComIS)(this.pcBx31)).EndInit();
((SysComIS)(this.pcBx30)).EndInit();
this.tbPg3.ResumeLayout(false);
this.tbPg3.PerformLayout();
((SysComIS)(this.pcBx72)).EndInit();
((SysComIS)(this.pcBx73)).EndInit();
((SysComIS)(this.pcBx74)).EndInit();
((SysComIS)(this.pcBx52)).EndInit();
((SysComIS)(this.pcBx53)).EndInit();
((SysComIS)(this.pcBx54)).EndInit();
((SysComIS)(this.pcBx55)).EndInit();
((SysComIS)(this.pcBx56)).EndInit();
((SysComIS)(this.pcBx57)).EndInit();
((SysComIS)(this.pcBx58)).EndInit();
((SysComIS)(this.pcBx59)).EndInit();
((SysComIS)(this.pcBx60)).EndInit();
((SysComIS)(this.pcBx61)).EndInit();
((SysComIS)(this.pcBx62)).EndInit();
((SysComIS)(this.pcBx63)).EndInit();
((SysComIS)(this.pcBx64)).EndInit();
((SysComIS)(this.pcBx65)).EndInit();
((SysComIS)(this.pcBx66)).EndInit();
((SysComIS)(this.pcBx67)).EndInit();
((SysComIS)(this.pcBx68)).EndInit();
((SysComIS)(this.pcBx69)).EndInit();
((SysComIS)(this.pcBx70)).EndInit();
((SysComIS)(this.pcBx71)).EndInit();
this.ResumeLayout(false);
```

```
}  
#endregion  
private SWF.TabControl tbCtl;  
private SWF.TabPage tbPg1;  
private SWF.TextBox txBx29;  
private SWF.PictureBox pcBx29;  
private SWF.TextBox txBx28;  
private SWF.PictureBox pcBx28;  
private SWF.TextBox txBx27;  
private SWF.PictureBox pcBx27;  
private SWF.TextBox txBx26;  
private SWF.PictureBox pcBx26;  
private SWF.TextBox txBx25;  
private SWF.PictureBox pcBx25;  
private SWF.TextBox txBx24;  
private SWF.PictureBox pcBx24;  
private SWF.TextBox txBx23;  
private SWF.PictureBox pcBx23;  
private SWF.TextBox txBx22;  
private SWF.PictureBox pcBx22;  
private SWF.TextBox txBx21;  
private SWF.PictureBox pcBx21;  
private SWF.TextBox txBx20;  
private SWF.PictureBox pcBx20;  
private SWF.TextBox txBx19;  
private SWF.PictureBox pcBx19;  
private SWF.TextBox txBx18;  
private SWF.PictureBox pcBx18;  
private SWF.TextBox txBx17;  
private SWF.PictureBox pcBx17;
```



```
private SWF.TextBox txBx16;  
private SWF.PictureBox pcBx16;  
private SWF.TextBox txBx15;  
private SWF.PictureBox pcBx15;  
private SWF.TextBox txBx14;  
private SWF.PictureBox pcBx14;  
private SWF.TextBox txBx13;  
private SWF.PictureBox pcBx13;  
private SWF.TextBox txBx12;  
private SWF.PictureBox pcBx12;  
private SWF.TextBox txBx11;  
private SWF.PictureBox pcBx11;  
private SWF.TextBox txBx10;  
private SWF.PictureBox pcBx10;  
private SWF.TextBox txBx9;  
private SWF.PictureBox pcBx9;  
private SWF.TextBox txBx8;  
private SWF.PictureBox pcBx8;  
private SWF.TextBox txBx7;  
private SWF.PictureBox pcBx7;  
private SWF.TextBox txBx6;  
private SWF.PictureBox pcBx6;  
private SWF.TextBox txBx5;  
private SWF.PictureBox pcBx5;  
private SWF.TextBox txBx4;  
private SWF.PictureBox pcBx4;  
private SWF.TextBox txBx3;  
private SWF.PictureBox pcBx3;  
private SWF.TextBox txBx2;  
private SWF.PictureBox pcBx2;
```

```
private SWF.TextBox txBx1;
private SWF.PictureBox pcBx1;
private SWF.TabPage tbPg2;
private SWF.TextBox txBx51;
private SWF.PictureBox pcBx51;
private SWF.TextBox txBx50;
private SWF.PictureBox pcBx50;
private SWF.TextBox txBx40;
private SWF.PictureBox pcBx40;
private SWF.TextBox txBx41;
private SWF.PictureBox pcBx41;
private SWF.TextBox txBx42;
private SWF.PictureBox pcBx42;
private SWF.TextBox txBx43;
private SWF.PictureBox pcBx43;
private SWF.TextBox txBx44;
private SWF.PictureBox pcBx44;
private SWF.TextBox txBx45;
private SWF.PictureBox pcBx45;
private SWF.TextBox txBx46;
private SWF.PictureBox pcBx46;
private SWF.TextBox txBx47;
private SWF.PictureBox pcBx47;
private SWF.TextBox txBx48;
private SWF.PictureBox pcBx48;
private SWF.TextBox txBx49;
private SWF.PictureBox pcBx49;
private SWF.TextBox txBx39;
private SWF.PictureBox pcBx39;
private SWF.TextBox txBx38;
```

```
private SWF.PictureBox pcBx38;  
private SWF.TextBox txBx37;  
private SWF.PictureBox pcBx37;  
private SWF.TextBox txBx36;  
private SWF.PictureBox pcBx36;  
private SWF.TextBox txBx35;  
private SWF.PictureBox pcBx35;  
private SWF.TextBox txBx34;  
private SWF.PictureBox pcBx34;  
private SWF.TextBox txBx33;  
private SWF.PictureBox pcBx33;  
private SWF.TextBox txBx32;  
private SWF.PictureBox pcBx32;  
private SWF.TextBox txBx31;  
private SWF.PictureBox pcBx31;  
private SWF.TextBox txBx30;  
private SWF.PictureBox pcBx30;  
private SWF.TabPage tbPg3;  
private SWF.TextBox txBx52;  
private SWF.PictureBox pcBx52;  
private SWF.TextBox txBx53;  
private SWF.PictureBox pcBx53;  
private SWF.TextBox txBx54;  
private SWF.PictureBox pcBx54;  
private SWF.TextBox txBx55;  
private SWF.PictureBox pcBx55;  
private SWF.TextBox txBx56;  
private SWF.PictureBox pcBx56;  
private SWF.TextBox txBx57;  
private SWF.PictureBox pcBx57;
```

```
private SWF.TextBox txBx58;  
private SWF.PictureBox pcBx58;  
private SWF.TextBox txBx59;  
private SWF.PictureBox pcBx59;  
private SWF.TextBox txBx60;  
private SWF.PictureBox pcBx60;  
private SWF.TextBox txBx61;  
private SWF.PictureBox pcBx61;  
private SWF.TextBox txBx62;  
private SWF.PictureBox pcBx62;  
private SWF.TextBox txBx63;  
private SWF.PictureBox pcBx63;  
private SWF.TextBox txBx64;  
private SWF.PictureBox pcBx64;  
private SWF.TextBox txBx65;  
private SWF.PictureBox pcBx65;  
private SWF.TextBox txBx66;  
private SWF.PictureBox pcBx66;  
private SWF.TextBox txBx67;  
private SWF.PictureBox pcBx67;  
private SWF.TextBox txBx68;  
private SWF.PictureBox pcBx68;  
private SWF.TextBox txBx69;  
private SWF.PictureBox pcBx69;  
private SWF.TextBox txBx70;  
private SWF.PictureBox pcBx70;  
private SWF.TextBox txBx71;  
private SWF.PictureBox pcBx71;  
private SWF.TextBox txBx72;  
private SWF.PictureBox pcBx72;
```

```

private SWF.TextBox txBx73;
private SWF.PictureBox pcBx73;
private SWF.TextBox txBx74;
private SWF.PictureBox pcBx74;
private SWF.Button btn1;
private SWF.Button btn2;

}
}

```

**Подпрограмма для ввода параметров судна. Основная часть.( Form1.cs.)**

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Xml;
using System.Threading;
namespace inputData
{
    public partial class inFrm : Form
    {
        private Dictionary<string, double> dic = new Dictionary<string, double>();
        private Dictionary<string, double> dic2 = new Dictionary<string, double>();
        public inFrm()
        {
            InitializeComponent();

```

```

    recVar();
}
private void btn1_Click(object sender, EventArgs e)
{
    createXmlDoc("myxml.xml");
    inputInXml("myxml.xml");
}
void createXmlDoc(string path)
{
    XmlTextWriter txtWrtr = new XmlTextWriter(path, Encoding.UTF8);
    txtWrtr.WriteStartDocument(); //step1
    txtWrtr.WriteStartElement("head"); //step2
    txtWrtr.WriteEndElement(); //step3
    txtWrtr.Close(); //step4
}
void inputInXml(string path)
{
    XmlDocument d0cum = new XmlDocument();
    d0cum.Load(path);
    for (int i = 1; i <= 3; i++)
    {
        XmlNode elem = d0cum.CreateElement("tab");
        d0cum.DocumentElement.AppendChild(elem); // указываем родителя
        XmlAttribute attribute = d0cum.CreateAttribute("number"); // создаём
атрибут
        attribute.Value = i.ToString(); // устанавливаем значение атрибута
        elem.Attributes.Append(attribute); // добавляем атрибут

        foreach (TextBox txBx in tbC11.TabPages[i-
1].Controls.OfType<TextBox>())

```

```

        {
            XmlNode subElement = d0cum.CreateElement(txBx.Tag.ToString());
            subElement.InnerText = txBx.Text;
            elem.AppendChild(subElement);
        }
    }
    d0cum.Save(path);
}
private void btn2_Click(object sender, EventArgs e)
{
    inputInXml("default.xml");
    for (int i = 0; i < 3; i++)
        foreach (TextBox txBx in
tbC11.TabPages[i].Controls.OfType<TextBox>())
            {
                txBx.Text = (dic[txBx.Tag.ToString()].ToString()).Replace(",", ".");
            }
}
public void recVar()
{
    string strName;
    for (int i = 0; i < 3; i++)
        foreach (TextBox txBx in
tbC11.TabPages[i].Controls.OfType<TextBox>())
            {
                strName = txBx.Tag.ToString();
                dic.Add(strName, ParseEx(txBx.Text));
            }
}
public double ParseEx(string str)

```

```
{  
  double value;  
  return  
    double.TryParse(str.Replace(",", "."), out value)  
    ? value  
    : double.Parse(str.Replace(".", ","));  
}  
}  
}
```



**Подпрограмма вывода меню для задания начальных значений параметров судна при движении на циркуляции (menuSelectScene)**

```
using UnityEngine;
using System.Collections;
using System;
using System.Xml;

public class menuSelectScene : MonoBehaviour
{
    //Объявление открытых переменных
    public moveShip[] ms;
    //public moveShip[] ms2;
    public Transform movCam;
    public GUIText title;
    public int startR = 0;
    //Объявление закрытых переменных
    private int flag = 1, countShips = 1;
    private bool winExit = false;
    private float sX = 200f, sY = 200f, tX = 200f, tY = 200f, ZZ = 22.5f, TT = 30f,
    amp = 20f, T2 = 300f;
    private double KK = Math.PI / 2, DD = Math.PI / 6, HH = 0.005, VV = 10, alfa =
    0f,
        m = 0.3, n = 0.3;
    private string statusWater = "Вкл";
    private bool sin = false, sinY = false;
    private bool perspective = true;
```

```

private bool hideMenu = false;
private bool existConfigFile = false;

public GameObject axisX, axisY;
//Функция вызываемая первый раз после запуска программы
//Построение dna
//Установка начальных значений
void Start()
{
    //Получение доступа к открытым переменным из файла moveShip.cs
    loadConfig();
    initiateAxis(axisX, true);
    initiateAxis(axisY, false);
    ms = GameObject.Find("ship").GetComponents<moveShip>();
    //Построение dna
    createPlane(2, 20);
    ms[0].currentPlane = 1;
    ms[0].nullVar(sX, sY, KK, DD, ZZ, HH, TT, VV);
}

private void initiateAxis(GameObject axis, bool X)
{
    int i = 0;
    if (!X)
        i = 100;
    for (; i <= 5000; i = i + 100)
    {
        GameObject gameObject = new GameObject();
        gameObject.AddComponent<TextMesh>();
        gameObject.AddComponent<MeshRenderer>();
    }
}

```

```

string value = i.ToString() + " м";
if (i == 5000 && X)
    value = "X";
else if (i == 5000 && !X)
    value = "Y";
gameObject.GetComponent<TextMesh>().text = value;
if (X)
    gameObject.GetComponent<TextMesh>().transform.position = new
Vector3(i - 20, 0, 0);
else
    gameObject.GetComponent<TextMesh>().transform.position = new
Vector3(-70, 0, i + 10);
gameObject.GetComponent<TextMesh>().fontSize = 20;
gameObject.GetComponent<TextMesh>().color = Color.green;
gameObject.GetComponent<TextMesh>().transform.localScale = new
Vector3(10, 10, 10);
gameObject.GetComponent<TextMesh>().transform.Rotate(90, 0, 0);
gameObject.transform.SetParent(axis.transform);
}
}

//Вызывается один раз за кадр
void Update()
{
    //Если нажата клавиша Escape то вызывается меню
    //т.е. значение переменной flag меняется на противоположный
    if (Input.GetKeyDown("escape") && (winExit == false))
    {
        flag ^= 1;
    }
}

```



```

        i.SetValue(this, Convert.ToSingle(param.GetAttribute("value")));
    if (i.Name == "sin")
        i.SetValue(this,
Convert.ToBoolean(param.GetAttribute("value")));
    }
    }
    }
    }
}
}
private void saveConfig()
{
    existConfigFile = System.IO.File.Exists("config.xml");
    if (existConfigFile)
    {
        System.IO.File.Delete("config.xml");
    }
    XmlTextWriter xmlConfigWriter = new XmlTextWriter("config.xml",
System.Text.Encoding.UTF8);
    xmlConfigWriter.WriteStartDocument();
    xmlConfigWriter.WriteStartElement("head");
    xmlConfigWriter.WriteEndElement();
    xmlConfigWriter.Close();
    XmlDocument saveXmlConfig = new XmlDocument();
    saveXmlConfig.Load("config.xml");
    System.Reflection.FieldInfo[] fi =
this.GetType().GetFields(System.Reflection.BindingFlags.NonPublic |
System.Reflection.BindingFlags.Instance);
    foreach (System.Reflection.FieldInfo i in fi)
    {

```

```

    if (i.FieldType == typeof(float) || i.FieldType == typeof(double) || i.Name ==
"sin")
    {
        XmlNode tag = saveXmlConfig.CreateElement(i.Name);
        saveXmlConfig.DocumentElement.AppendChild(tag);
        XmlAttribute value = saveXmlConfig.CreateAttribute("value");
        value.Value = i.GetValue(this).ToString();
        tag.Attributes.Append(value);
    }
}
saveXmlConfig.Save("config.xml");
}
//Объявление переменных размера окна меню
private Rect winRect1 = new Rect(Screen.width - 700, 10, 650, 400);
private Rect winRect2 = new Rect((Screen.width / 2) - 105, (Screen.height / 2) -
40, 215, 80);
private Rect winRect = new Rect(Screen.width - 700, 10, 650, 400);
//стандартное значение бегунка приближения камеры
private float hSliderValue = 50.0f;
private float hValue = 50.0f;
//Функция отвечающая за прорисовку окна меню
void OnGUI()
{
    //Если значение переменной равно true, то ...
    if (flag == 1)
    {
        //остановить моделирование
        Time.timeScale = 0;
        //и вывести меню
        if (!hideMenu)

```

```

    GUI.Window(0, winRect, myFunction, "MENU");
}
else
{
    // иначе продолжить моделирование
    Time.timeScale = 1;
}
//присвоение переменной hSliderValue текущего значения приближения
камеры
hSliderValue = movCam.GetComponent<Camera>().fieldOfView;
//Если hSliderValue меньше 20 то присвоить ей 20
//если hSliderValue больше 165 то присвоить 165
if (hSliderValue <= 20f)
    hSliderValue = 20f;
else if (hSliderValue >= 178f)
    hSliderValue = 178f;
//Присвоение камере измененного приближения
hSliderValue = GUI.HorizontalSlider(new Rect(25, 25, 130, 30), hSliderValue,
20.0f, 178.0f);
hValue = GUI.HorizontalSlider(new Rect(25, 55, 130, 30), hValue, 1.0f,
1500.0f);
GUI.color = Color.red;
perspective = GUI.Toggle(new Rect(160, 20, 20, 20), perspective, "");
perspective = GUI.Toggle(new Rect(160, 50, 20, 20), !perspective, "");
hideMenu = GUI.Toggle(new Rect(5, 5, 20, 20), hideMenu, "");
//Присвоение камере измененного приближения
movCam.GetComponent<Camera>().fieldOfView = hSliderValue;
movCam.GetComponent<Camera>().orthographic = perspective;
movCam.GetComponent<Camera>().orthographicSize = hValue;
}

```

```

//Прорисовка элементов меню и изменение значений параметров через меню
void myFunction(int winID)
{
    if (winExit != true)
    {
        winRect = winRect1;
        if (GUI.Button(new Rect(10, 20, 130, 25), "Ровное дно"))
        {
            saveConfig();
            Time.timeScale = 1;
            flag ^= 1;
            createPlane(2, ZZ);
            ms[0].nullVar(sX, sY, KK, DD, ZZ, HH, TT, VV);
            ms[0].currentPlane = 2;
            if (countShips > 1) createShips(countShips);
            startR = 1;
        }
        if (GUI.Button(new Rect(10, 50, 130, 25), "Резкие перепады"))
        {
            saveConfig();
            if (ZZ < 21f)
                ZZ = 21f;
            Time.timeScale = 1;
            flag ^= 1;
            createPlane(1, ZZ);
            ms[0].nullVar(sX, sY, KK, DD, ZZ, HH, TT, VV);
            ms[0].currentPlane = 1;
            startR = 1;
        }
        if (GUI.Button(new Rect(10, 80, 130, 25), "Кривое дно"))
    }
}

```



```
{
    saveConfig();
    if (ZZ < 21f)
        ZZ = 21f;
    Time.timeScale = 1;
    flag ^= 1;
    createPlane(0, ZZ);
    ms[0].nullVar(sX, sY, KK, DD, ZZ, HH, TT, VV);
    ms[0].currentPlane = 0;
    startR = 1;
}
if (GUI.Button(new Rect(145, 20, 55, 40), "ВЫХОД"))
{
    saveConfig();
    winExit = true;
}
if (GUI.Button(new Rect(145, 65, 55, 40), statusWater))
{
    ms[0].Water.active ^= true;
    if (ms[0].Water.active)
        statusWater = "ВЫКЛ";
    else
        statusWater = "ВКЛ";
}
ms[0].focus = true;
//Расстановка элементов меню
GUI.Box(new Rect(5, 110, 211, 267), "");
GUI.Box(new Rect(219, 110, 211, 267), "");
GUI.Box(new Rect(433, 110, 211, 267), "");
```

```
GUI.color = Color.red;

GUI.Label(new Rect(11, 110, 150, 20), "Координаты судна:");
GUI.color = Color.white;
GUI.Label(new Rect(15, 130, 15, 20), "X:");
try
{
    sX = Convert.ToSingle(GUI.TextField(new Rect(60, 131, 145, 18),
sX.ToString()));
}
catch
{
    sX = 1;
}
GUI.Label(new Rect(15, 150, 15, 20), "Y:");
try
{
    sY = Convert.ToSingle(GUI.TextField(new Rect(60, 151, 145, 18),
sY.ToString()));
}
catch
{
    sY = 1;
}
GUI.color = Color.red;

GUI.Label(new Rect(11, 170, 100, 20), "Курс судна:");
GUI.color = Color.white;
GUI.Label(new Rect(15, 190, 40, 20), (KK * 180 / Math.PI).ToString());
KK = GUI.HorizontalSlider(new Rect(60, 195, 145, 20), (float)KK, 0.0f,
```

6.28f);

```
GUI.color = Color.red;
```

```
GUI.Label(new Rect(11, 210, 190, 20), "Угол перекладки руля:");
```

```
GUI.color = Color.white;
```

```
GUI.Label(new Rect(15, 230, 40, 20), (DD * 180 / Math.PI).ToString());
```

```
DD = GUI.HorizontalSlider(new Rect(60, 235, 145, 20), (float)DD, -0.7f,
```

0.7f);

```
GUI.color = Color.red;
```

```
GUI.Label(new Rect(11, 250, 190, 20), "Глубина под килем:");
```

```
GUI.color = Color.white;
```

```
try
```

```
{
```

```
    ZZ = Convert.ToSingle(GUI.TextField(new Rect(60, 271, 145, 20),
```

ZZ.ToString()));

```
}
```

```
catch
```

```
{
```

```
    ZZ = 1;
```

```
}
```

```
GUI.color = Color.red;
```

```
GUI.Label(new Rect(11, 290, 190, 20), "Время моделирования:");
```

```
GUI.color = Color.white;
```

```
try
```

```
{
```

```
    HH = Convert.ToSingle(GUI.TextField(new Rect(60, 311, 145, 20),
```

HH.ToString(), 7));

```
}
```

```
catch
```

```
{
```

```
        HH = 0.002;
    }

    GUI.color = Color.red;

    GUI.Label(new Rect(225, 50, 200, 20), "Количество кораблей:");
    try
    {
        countShips = Convert.ToInt16(GUI.TextField(new Rect(370, 50, 30, 20),
countShips.ToString(), 1));
    }
    catch
    {
        countShips = 1;
    }
    GUI.color = Color.red;
    sinY = GUI.Toggle(new Rect(225, 85, 200, 20), sinY, "По оси Y");

    GUI.color = Color.red;
    GUI.Label(new Rect(225, 110, 200, 20), "Параметры dna:");
    GUI.color = Color.white;
    GUI.Label(new Rect(229, 135, 90, 20), "Амплитуда:");
    try
    {
        amp = Convert.ToSingle(GUI.TextField(new Rect(310, 135, 50, 20),
amp.ToString(), 20));
    }
    catch
    {
        amp = 20f;
    }
}
```

```
}
GUI.Label(new Rect(365, 135, 10, 20), "м");
GUI.Label(new Rect(229, 160, 90, 20), "Полупериод:");
try
{
    T2 = Convert.ToSingle(GUI.TextField(new Rect(310, 160, 50, 20),
T2.ToString(), 300));
}
catch
{
    T2 = 300f;
}
GUI.Label(new Rect(365, 160, 10, 20), "м");
GUI.color = Color.red;
GUI.Label(new Rect(225, 180, 190, 20), "Время Наблюдения:");
GUI.color = Color.white;
GUI.Label(new Rect(289, 200, 50, 20), "секунд");
try
{
    TT = Convert.ToSingle(GUI.TextField(new Rect(229, 200, 50, 20),
TT.ToString(), 30));
}
catch
{
    TT = 20f;
}
GUI.color = Color.red;
GUI.Label(new Rect(225, 220, 190, 20), "Скорость судна:");
GUI.color = Color.white;
GUI.Label(new Rect(289, 240, 50, 20), "м/с");
```

```

try
{
    VV = Convert.ToSingle(GUI.TextField(new Rect(229, 240, 50, 20),
VV.ToString(), 10));
}
catch
{
    VV = 10;
}
//меню выхода из программы
}
else if (winExit == true)
{
    winRect = winRect2;
    GUI.Label(new Rect(10, 15, 200, 20), "Вы действительно хотите выйти?");
    if (GUI.Button(new Rect(40, 45, 55, 25), "Да"))
    {
        Application.Quit();
    }
    if (GUI.Button(new Rect(115, 45, 55, 25), "Нет"))
    {
        winExit = false;
    }
}
}

```

```

private void createShips(int countShips)
{
    System.Random random = new System.Random();
    for (int i = 1; i < countShips; i++)

```

```

{
    moveShip tempShip = Instantiate(ms[0]);
    tempShip.line3d = Instantiate(ms[0].line3d);
    tempShip.line3d.material.shader = Shader.Find("Unlit/Color");
    tempShip.line3d.material.SetColor("_Color", new
Color((float)random.NextDouble(), (float)random.NextDouble(),
(float)random.NextDouble()));
    tempShip.nullVar(sX + i * (random.Next(4) + 1), sY - i * (random.Next(4) +
1), KK, DD, ZZ, HH, TT, VV);
    tempShip.currentPlane = 2;
}
}
public float meterToCu(double meter)
{
    return (float)(meter / ms[0].HHeight);
}

public float gradToRad(float grad)
{
    return (float)(grad * Math.PI / 180f);
}
//Функция для рисования дна
public void createPlane(int chnPl, double Z)
{
    float[,] terrainHeights;
    int i = 0, j = 0;
    int xzResol, width = 5000, height = 5000;
    System.Random random = new System.Random();
    ms[0].currentPlane = chnPl;
    float[,] array = new float[5, 5] { { 2f, 1f, 2f, 2f, 0.5f }, { 2f, 1f, 1f, 2f, 2f }, { 1f,

```

```

1f, 2f, 0.5f, 1f }, { 0.5f, 2f, 0f, 2f, 2f }, { 1f, 2f, 1f, 0.5f, 2f } };
    Terrain.activeTerrain.terrainData.alphamapResolution = 1025;
    Terrain.activeTerrain.terrainData.heightmapResolution = 1024;
    xzResol = Terrain.activeTerrain.terrainData.heightmapResolution;
    terrainHeights = Terrain.activeTerrain.terrainData.GetHeights(0, 0, xzResol,
xzResol);
    var maps = Terrain.activeTerrain.terrainData.GetAlphamaps(0, 0,
Terrain.activeTerrain.terrainData.alphamapWidth,
Terrain.activeTerrain.terrainData.alphamapHeight);
    int k = xzResol / array.GetLength(0);
    int l = xzResol / array.GetLength(1);
    int x = 0;
    //Рисование ровного дна
    if (chnPl == 2)
    {
        for (i = 0; i < xzResol; i++)
            for (j = 0; j < xzResol; j++)
                terrainHeights[i, j] = meterToCu(0);
    }
    //рисование Кривого дна
    else if (chnPl == 1)
    {
        for (i = 0; i < xzResol; i++)
        {
            for (j = 0; j < xzResol; j++)
            {
                terrainHeights[i, j] = meterToCu(1 * array[i / (k + 1), j / (l + 1)]);
            }
        }
    }
}

```



```

else
{
    int flag = 0;
    for (i = 0; i < xzResol; i++)
    {
        for (j = 0; j < xzResol; j++)
        {
            if (sinY)
                x = i;
            else
                x = j;
            terrainHeights[i, j] = meterToCu(1 + System.Math.Sin(3 *
System.Math.PI / 2 + (height / (T2 * 3)) * gradToRad(x * 1.0546875f)));
        }
    }
}
//Вывод на экран информации о форме дна
Terrain.activeTerrain.terrainData.SetHeights(0, 0, terrainHeights);
terrainHeights = Terrain.activeTerrain.terrainData.GetHeights(0, 0, xzResol,
xzResol);
for (i = 0; i < xzResol; i++)
    for (j = 0; j < xzResol; j++)
    {
        var a0 = maps[i, j, 0];
        if (chnPl == 2)
            a0 = 1;
        else
            a0 = terrainHeights[i, j];
        maps[i, j, 0] = a0;
    }
}

```

```
}

Terrain.activeTerrain.terrainData.SetAlphamaps(0, 0, maps);
Terrain.activeTerrain.terrainData.size = new Vector3(width, amp, height);
if (chnPl == 1)
    title.text = "Резкие перепады";
else
    if (chnPl == 2)
        title.text = "Ровное дно";
    else
        title.text = "Кривое дно";
}
}
```

**Подпрограмма управления судном, а также рисования траектории движения судна на мелководье при движении на циркуляции (moveShip)**

```

using UnityEngine;
using System.Collections;
using System.Collections.Generic;
using System;
using System.IO;
using System.Xml;

public class moveShip : MonoBehaviour {
    //Объявление открытых переменных
    public Terrain terra;
    public GameObject Water;
    public LineRenderer line3d, laser;
    public GameObject dragCam;
    public TextMesh textKurs;
    public GUIText guiText, guiTime, guiData;
    public int currentPlane;
public float HHeight = 2;
    public menuSelectScene[] mss;
    public double h, prevK, K, delta;
    public bool focus = false, followAngle = false, btntm = false;
    public float x0 = 250f, y0 = 250f, time, timeGeneral;
    //Объявление закрытых переменных
    private int counter = 0, miniCount;
    private float x, y, deltaH, Z, endTime, dx, dy;
    private double B, omega, V, Q, X, Y;

```

```

private Dictionary<string, double> dic = new Dictionary<string, double>();
//Функция вызываемая первый раз после запуска программы
//Установка начальных значений
void Start() {
    //Получение доступа к открытым переменным из файла
menuSelectScene.cs
    mss =
GameObject.Find("Camera").GetComponent<menuSelectScene>();
    //Запуск функции считывания значений переменных из xml файла
mxml.xml
    inputData();
    //Установка корабля по заданным координатам
transform.position = new Vector3(x0, Z, y0);
    //Рисование луча от киля судна до дна
laser.SetVertexCount(2);
    //Задание уровня относительно дна, на котором находится судно
deltaH = Z;
    //Получить объект камеры
dragCam = GameObject.Find("Camera");
}
//Функция установки требуемых значений для параметров судна
//Функция с параметрами
public void nullVar(float x00, float y00, double K00, double D00, float Z00,
double H00, float T00, double V00){
    //Задание значений параметров судна
    K = K00;
    prevK = K00;
    endTime = T00;
    V = V00;
    Z = Z00;
}

```

```

h = H00;
delta = D00;
x0 = x00;
y0 = y00;
counter = 0; x = 0f; y = 0f; //deltaH = 0;
B = 0; omega = 0; Q = 0; X = 0; Y = 0;
//Толщина линии троектории движения судна
line3d.SetWidth(5.75f, 5.75f);
//Установка корабля по заданным координатам
transform.position = new Vector3(x0, Z, y0);
//Рисование луча от киля судна до дна
laser.SetVertexCount(2);
//Задание уровня относительно дна, на котором находится судно
deltaH = Z;
//Задание уровня водной глади
Water.transform.position = new Vector3(253.3115f, Z+0.00782f,
254.1541f);
//Сохранение времени в переменной time
time = Time.fixedTime;
//Обнуление переменной, хранящей интервал времени от начала до
окончания моделирования
timeGeneral = 0f;
}
double function(double x, double dx, double dy) {
return (50*Math.Sin(0.01*x)+(dy-( 50*Math.Sin(0.01*dx) ));
}
//функция для разработчика
//необходима для вывода в файл, значений переменных, для проверки и
контроля
void outtofile(string text)

```

```

{
    using (System.IO.StreamWriter file = new
System.IO.StreamWriter("xyt.txt", true)) {
        file.WriteLine(text);
    }
}
//Функция запускаемая по таймеру
void FixedUpdate() {
    //Если время от начала моделирования не больше времени
моделирования endTime то,
    if (timeGeneral < endTime) {
        //Курс до решения уравнений, и курс после решения
уравнений, соответственно
        double prevK,nextK;
        prevK = K;
        //Функция решения диф уравнений
        melkovodie(deltaH);
        nextK = K;
        //если предыдущий курс больше чем следующий то вывести
сообщение
        if(prevK >= nextK)
            mss[0].title.text = "urrra";
        //Координаты судна относительно начального расположения
        x = x0+(float)X;
        y = y0+(float)Y;
        //Установка корабля по координатам полученным выше
        transform.position = new Vector3(x, Z, y);
        //Если переменная focus равна true то камера будет следить
за судном, т.е. камере будут присвоены координаты судна
        //Иначе камера получит те координаты которые имела до

```

приостановки слежки

```

        if (focus)
            dragCam.GetComponent<Camera>().transform.position =
new Vector3(x, Z+22.5f, y);
            //Если focus равна true и followAngle равна true то камера
будет следить и поворачиваться вслед
            //за судном, т.е. камере будет присвоен курс судна,
            //Иначе камера будет иметь постоянный курс направленный
на север
            if (followAngle && focus)
                dragCam.transform.rotation = Quaternion.Euler(90f,
(float)(K*180/Math.PI), 0f);
            else
                dragCam.transform.rotation = Quaternion.Euler(90f, 0, 0f);
            //Присвоение судну курса полученного после решения
уравнений
            transform.localRotation = Quaternion.Euler(0f,
(float)(K*180/Math.PI), 0f);
            //Рисование луча от киля судна...
            laser.SetPosition(0, new Vector3(x, Z-1, y));
            //до дна
            laser.SetPosition(1, new Vector3(x, 0, y));
            //Глубина под килем судна
            Vector3 pos = transform.position;
            pos.y = Terrain.activeTerrain.SampleHeight(transform.position);
            deltaH = Z - pos.y;
            //Текстовая метка отображающая глубину под килем судна
            guiText.text = deltaH.ToString("0.00") + " м";
            //Время прошедшее от начала моделирования
            timeGeneral = Time.time-time;

```

```

//Текстовая метка отображающая Время, прошедшее от
начала моделирования
guiTime.text = timeGeneral.ToString("0.00");
//Текстовая метка отображающее курс судна
textKurs.text = (K*180/Math.PI).ToString("0.00°") + " " +
x.ToString("0.0") + " " + y.ToString("0.0");
guiData.text = "H= " + deltaH.ToString("0.00") + " X= " + x.ToString("0.00")
+ " Y= " + y.ToString("0.00") + " K= " + (K * 180 / Math.PI).ToString("0.00°") + "
T= " + timeGeneral.ToString("0.00");
Debug.Log(" | H: " + deltaH.ToString("0.000") + " | x: " +
x.ToString("0.000") + " | y: " + y.ToString("0.000") + " | V: " +
V.ToString("0.000"));

using (System.IO.StreamWriter file = new System.IO.StreamWriter("xyt.txt",
true))
{
file.WriteLine("-----");
file.WriteLine("X: " + x.ToString("0.00") + " Y: " + y.ToString("0.00") + "
H: " + deltaH.ToString("0.00") + " K: " + (K * 180 / Math.PI).ToString("0.00°") + "
T: " + timeGeneral.ToString("0.00"));
}

//Счетчик используемый для отсчитывания шага рисования
точек, линии траектории, между которыми проводится эта линия
miniCount++;
//Если счетчик равен одному то...
if(miniCount == 1) {
//рисовать следующую точку и провести к нему линию
от предыдущей
line3d.SetVertexCount(counter+1);
line3d.SetPosition(counter, new Vector3(x, Z+1, y));

```



```

        //Обнуляем счетчик
        miniCount = 0;
        //Счетчик количества точек
        counter++;
    }
}
}
//Вызывается один раз за кадр
void Update() {

    float x,z,fow;
    //Координаты судна
    x = dragCam.GetComponent<Camera>().transform.position.x;
    z = dragCam.GetComponent<Camera>().transform.position.z;
    //Приближение камеры
    fow = dragCam.GetComponent<Camera>().fieldOfView;
    //Если нажата клавиша со стрелкой вверх, то переместить камеру на
0.5 единиц на север
    if (Input.GetKey (KeyCode.UpArrow)) {
        z += 5f;
        dragCam.GetComponent<Camera>().transform.position = new
Vector3(x, Z+22.5f, z); //change
    }
    //Если нажата клавиша со стрелкой вниз, то переместить камеру на
0.5 единиц на юг
    if (Input.GetKey (KeyCode.DownArrow)) {
        z -= 5f;
        dragCam.GetComponent<Camera>().transform.position = new
Vector3(x, Z+22.5f, z); //change
    }
}

```

//Если нажата клавиша со стрелкой влево, то переместить камеру на 0.5 единиц на запад

```

    if (Input.GetKey(KeyCode.LeftArrow)) {
        x -= 5f;
        dragCam.GetComponent<Camera>().transform.position = new
Vector3(x, Z+22.5f, z); //change
    }

```

//Если нажата клавиша со стрелкой вправо, то переместить камеру на 0.5 единиц на восток

```

    if (Input.GetKey(KeyCode.RightArrow)) {
        x += 5f;
        dragCam.GetComponent<Camera>().transform.position = new
Vector3(x, Z+22.5f, z); //change
    }

```

```

    if (Input.GetKey(KeyCode.UpArrow) && Input.GetKey(KeyCode.LeftShift))
    {
        z += 20f;
        dragCam.GetComponent<Camera>().transform.position = new Vector3(x, Z
+ 22.5f, z); //change
    }

```

```

    if (Input.GetKey(KeyCode.DownArrow) &&
Input.GetKey(KeyCode.LeftShift))
    {
        z -= 20f;
        dragCam.GetComponent<Camera>().transform.position = new Vector3(x, Z
+ 22.5f, z); //change
    }

```

```

    if (Input.GetKey(KeyCode.LeftArrow) && Input.GetKey(KeyCode.LeftShift))
    {
        x -= 20f;

```

```

dragCam.GetComponent<Camera>().transform.position = new Vector3(x, Z
+ 22.5f, z); //change
}
if (Input.GetKey(KeyCode.RightArrow) && Input.GetKey(KeyCode.LeftShift))
{
x += 20f;
dragCam.GetComponent<Camera>().transform.position = new Vector3(x, Z
+ 22.5f, z); //change
}

//Если нажата клавиша со знаком минус, то отдалить камеру на 2
единиц от судну
if (Input.GetKey (KeyCode.KeypadMinus)) {
fow += 10;
dragCam.GetComponent<Camera>().fieldOfView = fow;
} else
//иначе если нажата клавиша со знаком плюс, то приблизить камеру
на 2 единиц к судну
if (Input.GetKey (KeyCode.KeypadPlus)) {
fow -= 10;
dragCam.GetComponent<Camera>().fieldOfView = fow;
}

//Если нажата клавиша F, то сменить значение переменной focus на
противоположное(true->>false или false->>true)
if (Input.GetKeyDown (KeyCode.F))
focus ^= true;

//Если нажата клавиша G, то сменить значение переменной
followAngle на противоположное(true->>false или false->>true)
if (Input.GetKeyDown (KeyCode.G))
followAngle ^= true;

//Если нажата клавиша R, то начать моделирование сначала

```

```

    if (Input.GetKeyDown (KeyCode.R)) {
        //Прорисовка дна
        mss[0].createPlane(currentPlane, Z);
        //Установка предыдущих значений
        nullVar(x0, y0, prevK, delta, Z, h, endTime, V); //change
    }
}
//Функция считывания значений параметров судна из xml файла:
myxml.xml
void inputData()
{
    string fieldName = "",fieldValue = "", doubleStr = "";
    XmlTextReader reader = new XmlTextReader ("myxml.xml");
    while (reader.Read())
    {
        switch (reader.NodeType)
        {
            case XmlNodeType.Element:
                fieldName = reader.Name;
                fieldValue = "";
                break;
            case XmlNodeType.Text:
                fieldValue = reader.Value;
                break;
        }
        if ((fieldName != "tab") && (fieldName != "head") &&
            (fieldName != "") && (doubleStr != fieldName)&&(fieldValue != ""))
        {
            dic.Add(fieldName, ParseEx(fieldValue));
            doubleStr = fieldName;
        }
    }
}

```

```

    }
}
}
//Функция используемая в inputData() для преобразования чисел с точкой
в числа с запятой или наоборот
public double ParseEx(string str)
{
double value;
return
double.TryParse(str.Replace(",", "."), out value)
? value
: double.Parse(str.Replace(".", ","));
}
// функция решающая диф уравнения
void melkovodie(double H) {
//Объявление коэффициентов в методе Рунге-Кутты
double k11 = 0.0, k12 = 0.0, k13 = 0.0, k14 = 0.0,
k21 = 0.0, k22 = 0.0, k23 = 0.0, k24 = 0.0,
k31 = 0.0, k32 = 0.0, k33 = 0.0, k34 = 0.0,
k41 = 0.0, k42 = 0.0, k43 = 0.0, k44 = 0.0,
k51 = 0.0, k52 = 0.0, k53 = 0.0, k54 = 0.0,
k61 = 0.0, k62 = 0.0, k63 = 0.0, k64 = 0.0,
k71 = 0.0, k72 = 0.0, k73 = 0.0, k74 = 0.0;
//Вспомогательные переменные
double Db = 3;
double Bs1, Bs2, Bs3, Bs4;
double n1=0,m1=0,n2=0,m2=0,n3=0,m3=0,n4=0,m4=0;
double DDb1, DDb2, DDb3, DDb4;
//первая стадия метода Рунге-Кутты 4-го порядка
DDb1 =

```

$0.3 * \text{Math.Pow}((\text{omega} * \text{dic}["L"] / \text{dic}["V0"]), 2) * ((\text{Db} * \text{Db}) / (\text{dic}["\text{sigma}"] * \text{dic}["L"] * \text{dic}["T"]));$

$n1 =$

$(B * \text{Math.PI} * (\text{dic}["T"] / \text{dic}["L"]) * (\text{dic}["A1"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A2"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A3"] * (\text{dic}["T"] / H) + \text{dic}["A4"])) + B * \text{Math.Abs}(B) * ((1.055 + 0.251 * (\text{dic}["BB"] / \text{dic}["T"]) + 0.021 * \text{Math.Pow}((\text{dic}["BB"] / \text{dic}["T"]), 2) + 12.49 * (\text{dic}["T"] / \text{dic}["L"]) + 0.0244 * (\text{dic}["L"] / \text{dic}["T"])) - \text{dic}["ask"]) * (3.34 * \text{Math.Pow}((\text{dic}["T"] / H), 2) - 0.234 * (\text{dic}["T"] / H) + 1)) + (\text{omega} * \text{dic}["L"] / \text{dic}["V0"]) * (\text{dic}["A5"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A6"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A7"] * (\text{dic}["T"] / H) + \text{dic}["A8"]) + B * \text{Math.Abs}(\text{omega} * \text{dic}["L"] / \text{dic}["V0"]) * (\text{dic}["A9"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A10"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A11"] * (\text{dic}["T"] / H) + \text{dic}["A12"]));$

$m1 =$

$(B * (3.36 * ((\text{dic}["T"] / \text{dic}["L"]) + 0.0136) * (\text{dic}["Sk"] + \text{dic}["ak"]))) * (\text{dic}["B1"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["B2"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["B3"] * (\text{dic}["T"] / H) + \text{dic}["B4"]) - (0.1595 - 0.0469 * (\text{dic}["BB"] / \text{dic}["T"]) + 0.00375 * \text{Math.Pow}((\text{dic}["BB"] / \text{dic}["T"]), 2)) * (\text{dic}["B5"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["B6"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["B7"] * (\text{dic}["T"] / H) + \text{dic}["B8"]));$

$Bs1 = (\text{dic}["nyd"] * \text{delta} -$

$\text{formul}("n\_yb", 0, H) * B + \text{formul}("_nyw", 0, H) * (\text{omega} * \text{dic}["L"] / \text{dic}["V0"] -$

$n1 * (B * \text{Math.Sign}(\text{omega})) * (\text{omega} * \text{dic}["L"] / \text{dic}["V0"])) / \text{formul}("m", 22, H);$

$k11 = h * (Bs1 * (\text{Math.Pow}(\text{dic}["V0"], 2) / (\text{dic}["L"] * V));$

$k21 = h * ($

$((\text{dic}["nyd"] * \text{delta} * (\text{dic}["lp"] / \text{dic}["L"]) + \text{formul}("m\_yb", 0, H) * B + \text{formul}("m\_yw", 0, H) * (\text{omega} * \text{dic}["L"] / \text{dic}["V0"] + m1 * (B * \text{Math.Sign}(\text{omega})) * (\text{omega} * \text{dic}["L"] / \text{dic}["V0"]))) / \text{formul}("m", 66, H)) * \text{Math.Pow}((\text{dic}["V0"] / \text{dic}["L"]), 2);$

$k31 = h * ($

$((1 / \text{formul}("m", 11, H)) * ((\text{dic}["nx0"] * \text{Math.Pow}((\text{dic}["V0"] / V), 3) + \text{DDb1}) - (\text{dic}["nx0"] + 0.4 * \text{dic}["nyd"] * \text{delta} * \text{delta})) -$

$B * ((\text{formul}("m", 22, H) / \text{formul}("m", 11, H)) * (\omega * \text{dic}["L"] / \text{dic}["V0"]) - k_{11} * \text{dic}["L"] / \text{dic}["V0"])) * (V / \text{dic}["V0"]) * (\text{Math.Pow}(\text{dic}["V0"], 2) / \text{dic}["L"]));$

$$k_{41} = h * (K + B);$$

$$k_{51} = h * (\omega);$$

$$k_{61} = h * (V * \text{Math.Sin}(K));$$

$$k_{71} = h * (V * \text{Math.Cos}(K));$$

//Вторая стадия метода Рунге-Кутты 4-го порядка

$$DDb2 = 0.3 * \text{Math.Pow}((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"], 2) * ((Db * Db) / (\text{dic}["sigma"] * \text{dic}["L"] * \text{dic}["T"]));$$

$$n2 = ((B + 0.5 * k_{11}) * \text{Math.PI} * (\text{dic}["T"] / \text{dic}["L"]) * (\text{dic}["A1"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A2"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A3"] * (\text{dic}["T"] / H) + \text{dic}["A4"]) + (B + 0.5 * k_{11}) * \text{Math.Abs}((B + 0.5 * k_{11}) * ((1.055 + 0.251 * (\text{dic}["BB"] / \text{dic}["T"]) + 0.021 * \text{Math.Pow}((\text{dic}["BB"] / \text{dic}["T"]), 2) + 12.49 * (\text{dic}["T"] / \text{dic}["L"]) + 0.0244 * (\text{dic}["L"] / \text{dic}["T"]) - \text{dic}["ask"]) * (3.34 * \text{Math.Pow}((\text{dic}["T"] / H), 2) - 0.234 * (\text{dic}["T"] / H) + 1)) + ((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]) * (\text{dic}["A5"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A6"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A7"] * (\text{dic}["T"] / H) + \text{dic}["A8"]) + (B + 0.5 * k_{11}) * \text{Math.Abs}((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]) * (\text{dic}["A9"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A10"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A11"] * (\text{dic}["T"] / H) + \text{dic}["A12"]));$$

$$m2 = ((B + 0.5 * k_{11}) * (3.36 * ((\text{dic}["T"] / \text{dic}["L"]) + 0.0136) * (\text{dic}["Sk"] + \text{dic}["ak"])) * (\text{dic}["B1"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["B2"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["B3"] * (\text{dic}["T"] / H) + \text{dic}["B4"]) - (0.1595 - 0.0469 * (\text{dic}["BB"] / \text{dic}["T"]) + 0.00375 * \text{Math.Pow}((\text{dic}["BB"] / \text{dic}["T"]), 2)) * (\text{dic}["B5"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["B6"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["B7"] * (\text{dic}["T"] / H) + \text{dic}["B8"]));$$

$$Bs2 = (\text{dic}["nyd"] * \text{delta-formul}("n_yb", 0, H)) * (B + 0.5 * k_{11}) * \text{Math.Abs}((B + 0.5 * k_{11}) * ((1.055 + 0.251 * (\text{dic}["BB"] / \text{dic}["T"]) + 0.021 * \text{Math.Pow}((\text{dic}["BB"] / \text{dic}["T"]), 2) + 12.49 * (\text{dic}["T"] / \text{dic}["L"]) + 0.0244 * (\text{dic}["L"] / \text{dic}["T"]) - \text{dic}["ask"]) * (3.34 * \text{Math.Pow}((\text{dic}["T"] / H), 2) - 0.234 * (\text{dic}["T"] / H) + 1)) + ((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]) * (\text{dic}["A5"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A6"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A7"] * (\text{dic}["T"] / H) + \text{dic}["A8"]) + (B + 0.5 * k_{11}) * \text{Math.Abs}((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]) * (\text{dic}["A9"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A10"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A11"] * (\text{dic}["T"] / H) + \text{dic}["A12"]));$$

$k_{11}) + \text{formul}(\text{"_nyw"}, 0, H) * ((\omega + 0.5 * k_{21}) * \text{dic}[\text{"L"}] / \text{dic}[\text{"V0"}]) - n_2 * ((B + 0.5 * k_{11}) * \text{Math.Sign}(\omega + 0.5 * k_{21})) * ((\omega + 0.5 * k_{21}) * \text{dic}[\text{"L"}] / \text{dic}[\text{"V0"}]) / \text{formul}(\text{"m"}, 22, H);$

$$k_{12} = h * (Bs_2 * (\text{Math.Pow}(\text{dic}[\text{"V0"}], 2) / (\text{dic}[\text{"L"}] * (V + 0.5 * k_{31})))$$

);

$$k_{22} = h * ($$

$((\text{dic}[\text{"nyd"}] * \delta * (\text{dic}[\text{"lp"}] / \text{dic}[\text{"L"}]) + \text{formul}(\text{"m\_yb"}, 0, H) * (B + 0.5 * k_{11}) + \text{formul}(\text{"m\_yw"}, 0, H) * ((\omega + 0.5 * k_{21}) * \text{dic}[\text{"L"}] / \text{dic}[\text{"V0"}]) + m_2 * ((B + 0.5 * k_{11}) * \text{Math.Sign}(\omega + 0.5 * k_{21})) * ((\omega + 0.5 * k_{21}) * \text{dic}[\text{"L"}] / \text{dic}[\text{"V0"}])) / \text{formul}(\text{"m"}, 66, H) * \text{Math.Pow}((\text{dic}[\text{"V0"}] / \text{dic}[\text{"L"}]), 2) );$

$$k_{32} = h * ($$

$((1 / \text{formul}(\text{"m"}, 11, H)) * ((\text{dic}[\text{"nx0"}] * \text{Math.Pow}((\text{dic}[\text{"V0"}] / (V + 0.5 * k_{31})), 3) + \text{DDb}_2) - (\text{dic}[\text{"nx0"}] + 0.4 * \text{dic}[\text{"nyd"}] * \delta * \delta)) - (B + 0.5 * k_{11}) * ((\text{formul}(\text{"m"}, 22, H) / \text{formul}(\text{"m"}, 11, H)) * ((\omega + 0.5 * k_{21}) * \text{dic}[\text{"L"}] / \text{dic}[\text{"V0"}]) - k_{12} * \text{dic}[\text{"L"}] / \text{dic}[\text{"V0"}])) * ((V + 0.5 * k_{31}) / \text{dic}[\text{"V0"}])) * (\text{Math.Pow}(\text{dic}[\text{"V0"}], 2) / \text{dic}[\text{"L"}]) );$

$$k_{42} = h * ((K + 0.5 * k_{51}) + (B + 0.5 * k_{11}));$$

$$k_{52} = h * ((\omega + 0.5 * k_{21}));$$

$$k_{62} = h * ((V + 0.5 * k_{31}) * \text{Math.Sin}((K + 0.5 * k_{51})));$$

$$k_{72} = h * ((V + 0.5 * k_{31}) * \text{Math.Cos}((K + 0.5 * k_{51})));$$

//третья стадия метода Рунге-Кутты 4-го порядка

$$\text{DDb}_3 = 0.3 * \text{Math.Pow}(((\omega + 0.5 * k_{22}) * \text{dic}[\text{"L"}] / \text{dic}[\text{"V0"}]), 2) * ((\text{Db} * \text{Db}) / (\text{dic}[\text{"sigma"}] * \text{dic}[\text{"L"}] * \text{dic}[\text{"T"}]));$$

);

$$n_3 = ((B + 0.5 * k_{12}) * \text{Math.PI} * (\text{dic}[\text{"T"}] / \text{dic}[\text{"L"}]) * (\text{dic}[\text{"A1"}] * \text{Math.Pow}((\text{dic}[\text{"T"}] / H), 3) + \text{dic}[\text{"A2"}] * \text{Math.Pow}((\text{dic}[\text{"T"}] / H), 2) + \text{dic}[\text{"A3"}] * (\text{dic}[\text{"T"}] / H) + \text{dic}[\text{"A4"}]) + (B + 0.5 * k_{12}) * \text{Math.Abs}((B + 0.5 * k_{12}) * ((1.055 + 0.251 * (\text{dic}[\text{"BB"}] / \text{dic}[\text{"T"}]) + 0.021 * \text{Math.Pow}((\text{dic}[\text{"BB"}] / \text{dic}[\text{"T"}]), 2) + 12.49 * (\text{dic}[\text{"T"}] / \text{dic}[\text{"L"}]) + 0.0244 * (\text{dic}[\text{"L"}] / \text{dic}[\text{"T"}]) - \text{dic}[\text{"ask"}]) * (3.34 * \text{Math.Pow}((\text{dic}[\text{"T"}] / H), 2) - 0.234 * (\text{dic}[\text{"T"}] / H) + 1)) + ((\omega + 0.5 * k_{22}) * \text{dic}[\text{"L"}] / \text{dic}[\text{"V0"}]), 2) * ((\text{Db} * \text{Db}) / (\text{dic}[\text{"sigma"}] * \text{dic}[\text{"L"}] * \text{dic}[\text{"T"}]));$$

$n_3 = ((B + 0.5 * k_{12}) * \text{Math.PI} * (\text{dic}[\text{"T"}] / \text{dic}[\text{"L"}]) * (\text{dic}[\text{"A1"}] * \text{Math.Pow}((\text{dic}[\text{"T"}] / H), 3) + \text{dic}[\text{"A2"}] * \text{Math.Pow}((\text{dic}[\text{"T"}] / H), 2) + \text{dic}[\text{"A3"}] * (\text{dic}[\text{"T"}] / H) + \text{dic}[\text{"A4"}]) + (B + 0.5 * k_{12}) * \text{Math.Abs}((B + 0.5 * k_{12}) * ((1.055 + 0.251 * (\text{dic}[\text{"BB"}] / \text{dic}[\text{"T"}]) + 0.021 * \text{Math.Pow}((\text{dic}[\text{"BB"}] / \text{dic}[\text{"T"}]), 2) + 12.49 * (\text{dic}[\text{"T"}] / \text{dic}[\text{"L"}]) + 0.0244 * (\text{dic}[\text{"L"}] / \text{dic}[\text{"T"}]) - \text{dic}[\text{"ask"}]) * (3.34 * \text{Math.Pow}((\text{dic}[\text{"T"}] / H), 2) - 0.234 * (\text{dic}[\text{"T"}] / H) + 1)) + ((\omega + 0.5 * k_{22}) * \text{dic}[\text{"L"}] / \text{dic}[\text{"V0"}]), 2) * ((\text{Db} * \text{Db}) / (\text{dic}[\text{"sigma"}] * \text{dic}[\text{"L"}] * \text{dic}[\text{"T"}]));$



\*

$k22) * \text{dic}["L"] / \text{dic}["V0"] * (\text{dic}["A5"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A6"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A7"] * (\text{dic}["T"] / H) + \text{dic}["A8"])] + (B + 0.5 * k12) * \text{Math.Abs}((\omega + 0.5 * k22) * \text{dic}["L"] / \text{dic}["V0"] * (\text{dic}["A9"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A10"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A11"] * (\text{dic}["T"] / H) + \text{dic}["A12"]));$

$m3 = (B + 0.5 * k12) * (3.36 * ((\text{dic}["T"] / \text{dic}["L"]) + 0.0136) * (\text{dic}["Sk"] + \text{dic}["ak"])) * (\text{dic}["B1"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["B2"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["B3"] * (\text{dic}["T"] / H) + \text{dic}["B4"]) - (0.1595 - 0.0469 * (\text{dic}["BB"] / \text{dic}["T"]) + 0.00375 * \text{Math.Pow}((\text{dic}["BB"] / \text{dic}["T"]), 2)) * (\text{dic}["B5"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["B6"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["B7"] * (\text{dic}["T"] / H) + \text{dic}["B8"]));$

$Bs3 = (\text{dic}["nyd"] * \text{delta-formul}("n\_yb", 0, H) * (B + 0.5 * k12) + \text{formul}("_nyw", 0, H) * ((\omega + 0.5 * k22) * \text{dic}["L"] / \text{dic}["V0"]) - n3 * ((B + 0.5 * k12) * \text{Math.Sign}(\omega + 0.5 * k22)) * ((\omega + 0.5 * k22) * \text{dic}["L"] / \text{dic}["V0"])) / \text{formul}("m", 22, H);$

$k13 = h * (Bs3 * (\text{Math.Pow}(\text{dic}["V0"], 2) / (\text{dic}["L"] * (V + 0.5 * k32)))));$

$k23 = h * ((\text{dic}["nyd"] * \text{delta} * (\text{dic}["lp"] / \text{dic}["L"]) + \text{formul}("m\_yb", 0, H) * (B + 0.5 * k12) + \text{formul}("m\_yw", 0, H) * ((\omega + 0.5 * k22) * \text{dic}["L"] / \text{dic}["V0"]) + m3 * ((B + 0.5 * k12) * \text{Math.Sign}(\omega + 0.5 * k22)) * ((\omega + 0.5 * k22) * \text{dic}["L"] / \text{dic}["V0"])) / \text{formul}("m", 66, H) * \text{Math.Pow}((\text{dic}["V0"] / \text{dic}["L"]), 2));$

$k33 = h * (((1 / \text{formul}("m", 11, H)) * ((\text{dic}["nx0"] * \text{Math.Pow}((\text{dic}["V0"] / (V + 0.5 * k32)), 3) + \text{DDb3}) - (\text{dic}["nx0"] + 0.4 * \text{dic}["nyd"] * \text{delta} * \text{delta})) - (B + 0.5 * k12) * ((\text{formul}("m", 22, H) / \text{formul}("m", 11, H)) * ((\omega + 0.5 * k22) * \text{dic}["L"] / \text{dic}["V0"]) - k13 * \text{dic}["L"] / \text{dic}["V0"])) * ((V + 0.5 * k32) / \text{dic}["V0"]) * (\text{Math.Pow}(\text{dic}["V0"], 2) / \text{dic}["L"]);$

$k33 = h * (((1 / \text{formul}("m", 11, H)) * ((\text{dic}["nx0"] * \text{Math.Pow}((\text{dic}["V0"] / (V + 0.5 * k32)), 3) + \text{DDb3}) - (\text{dic}["nx0"] + 0.4 * \text{dic}["nyd"] * \text{delta} * \text{delta})) - (B + 0.5 * k12) * ((\text{formul}("m", 22, H) / \text{formul}("m", 11, H)) * ((\omega + 0.5 * k22) * \text{dic}["L"] / \text{dic}["V0"]) - k13 * \text{dic}["L"] / \text{dic}["V0"])) * ((V + 0.5 * k32) / \text{dic}["V0"]) * (\text{Math.Pow}(\text{dic}["V0"], 2) / \text{dic}["L"]);$

$k33 = h * (((1 / \text{formul}("m", 11, H)) * ((\text{dic}["nx0"] * \text{Math.Pow}((\text{dic}["V0"] / (V + 0.5 * k32)), 3) + \text{DDb3}) - (\text{dic}["nx0"] + 0.4 * \text{dic}["nyd"] * \text{delta} * \text{delta})) - (B + 0.5 * k12) * ((\text{formul}("m", 22, H) / \text{formul}("m", 11, H)) * ((\omega + 0.5 * k22) * \text{dic}["L"] / \text{dic}["V0"]) - k13 * \text{dic}["L"] / \text{dic}["V0"])) * ((V + 0.5 * k32) / \text{dic}["V0"]) * (\text{Math.Pow}(\text{dic}["V0"], 2) / \text{dic}["L"]);$

```

k43 = h * ((K + 0.5 * k52) + (B + 0.5 * k12));
k53 = h * ((omega + 0.5 * k22));
k63 = h * ((V + 0.5 * k32) * Math.Sin((K + 0.5 * k52)));
k73 = h * ((V + 0.5 * k32) * Math.Cos((K + 0.5 * k52)));
//четвертая стадия метода Рунге-Кутты 4-го порядка
DDb4 = 0.3*Math.Pow(((omega +
k23)*dic["L"]/dic["V0"]),2)*((Db*Db)/(dic["sigma"]*dic["L"]*dic["T"]));
n4 = ((B +
k13)*Math.PI*(dic["T"]/dic["L"])*(dic["A1"]*Math.Pow((dic["T"]/H),3)+dic["A2"]
*Math.Pow((dic["T"]/H),2)+dic["A3"]*(dic["T"]/H)+dic["A4"])+(B +
k13)*Math.Abs((B +
k13))*((1.055+0.251*(dic["BB"]/dic["T"])+0.021*Math.Pow((dic["BB"]/dic["T"]),2)
+12.49*(dic["T"]/dic["L"])+0.0244*(dic["L"]/dic["T"])-
dic["ask"])*(3.34*Math.Pow((dic["T"]/H),2)-0.234*(dic["T"]/H)+1))+((omega +
k23)*dic["L"]/dic["V0"])*(dic["A5"]*Math.Pow((dic["T"]/H),3)+dic["A6"]*Math.Po
w((dic["T"]/H),2)+dic["A7"]*(dic["T"]/H)+dic["A8"])+(B + k13)*Math.Abs((omega
+
k23)*dic["L"]/dic["V0"])*(dic["A9"]*Math.Pow((dic["T"]/H),3)+dic["A10"]*Math.P
ow((dic["T"]/H),2)+dic["A11"]*(dic["T"]/H)+dic["A12"]));
m4 = ((B +
k13)*(3.36*((dic["T"]/dic["L"])+0.0136)*(dic["Sk"]+dic["ak"])*(dic["B1"]*Math.P
ow((dic["T"]/H),3)+dic["B2"]*Math.Pow((dic["T"]/H),2)+dic["B3"]*(dic["T"]/H)+di
c["B4"])-(0.1595-
0.0469*(dic["BB"]/dic["T"])+0.00375*Math.Pow((dic["BB"]/dic["T"]),2))*((dic["B5"
]*Math.Pow((dic["T"]/H),3)+dic["B6"]*Math.Pow((dic["T"]/H),2)+dic["B7"]*(dic["
T"]/H)+dic["B8"]));
Bs4 = (dic["nyd"]*delta-formul("n_yb",0,H)*(B +
k13)+formul("_nyw",0,H)*((omega + k23)*dic["L"]/dic["V0"])-n4*((B +
k13)*Math.Sign(omega + k23))*((omega +
k23)*dic["L"]/dic["V0"]))/formul("m",22,H);

```

```

k14 = h * ( Bs4*(Math.Pow(dic["V0"], 2)/(dic["L"] * (V + k33))) );
k24 = h * (
((dic["nyd"]*delta*(dic["lp"]/dic["L"])+formul("m_yb",0,H)*(B +
k13)+formul("m_yw",0,H)*((omega + k23)*dic["L"]/dic["V0"])+m4*((B +
k13)*Math.Sign(omega + k23))*((omega +
k23)*dic["L"]/dic["V0"]))/formul("m",66,H))*Math.Pow((dic["V0"]/dic["L"]),2) );
k34 = h * (
(((1/formul("m",11,H))*((dic["nx0"]*Math.Pow((dic["V0"]/(V + k33)),3)+DDb4)-
(dic["nx0"]+0.4*dic["nyd"]*delta*delta))-(B +
k13)*((formul("m",22,H)/formul("m",11,H))*((omega + k23)*dic["L"]/dic["V0"])-
k14*dic["L"]/dic["V0"]))*((V + k33)/dic["V0"])*(Math.Pow(dic["V0"],2)/dic["L"])
);

k44 = h * ((K + k53) + (B + k13));
k54 = h * ((omega + k23));
k64 = h * ((V + k33) * Math.Sin((K + k53)));
k74 = h * ((V + k33) * Math.Cos((K + k53)));

double dB = (k11 + (2 * k12) + (2 * k13) + k14) / 6;
B = B + dB;
double domega = (k21 + (2 * k22) + (2 * k23) + k24) / 6;
omega = omega + domega;
double dV = (k31 + (2 * k32) + (2 * k33) + k34) / 6;
V = V + dV;

K = K + h*omega;
Q = K + B;
X = X + h*V*Math.Sin(Q);
Y = Y + h*V*Math.Cos(Q);
}

```

//Подстановка нужной формулы в функцию выше, формула выбирается

по параметрам введенным в формулу

```

double formul(string type, int num, double H)
{
    double res = 0;
    double cstFppn = (1-
(dic["Fsppn"]/dic["Fppn"])*(Math.Sqrt(1+dic["sigmap"])-
1)/(Math.Sqrt(1+dic["sigmap"])+1));
    if((type == "k")&&(num == 11))
        res =
(dic["ass1"]*Math.Pow((dic["T"]/H),3)+dic["bss1"]*Math.Pow((dic["T"]/H),2)+dic["
css1"]*(dic["T"]/H)+dic["d"]);
    else
        if((type == "k")&&(num == 22))
            res =
(dic["a1"]*Math.Pow((dic["T"]/H),3)+dic["b1"]*Math.Pow((dic["T"]/H),2)+dic["c1"
]*dic["T"]/H)+dic["d"]);
    else
        if((type == "k")&&(num == 66))
            res =
(dic["as1"]*Math.Pow((dic["T"]/H),3)+dic["bs1"]*Math.Pow((dic["T"]/H),2)+dic["c
s1"]*(dic["T"]/H)+dic["d"]);
    else
        if((type == "m")&&(num == 11))
            res =
(2*dic["Cb"]*dic["BB"]*(dic["po"]*dic["W"]+dic["alfa11"]*formul("k",11,H)))/(dic
["sigma"]*dic["L"]*dic["po"]*dic["W"]);
    else
        if((type == "m")&&(num == 22))
            res =
(2*dic["Cb"]*dic["BB"]*(dic["po"]*dic["W"]+dic["alfa22"]*formul("k",22,H)))/(dic

```

```

["sigma"]*dic["L"]*dic["po"]*dic["W"]);
    else
        if((type == "m")&&(num == 66))
            res =
(2*dic["Cb"]*dic["BB"]*Math.Pow(dic["poz"]/dic["L"],2)*(dic["po"]*dic["W"]*Mat
h.Pow(dic["poz"],2)+dic["alfa66"]*formul("k",66,H)))/(dic["sigma"]*dic["L"]*dic["
po"]*dic["W"]*Math.Pow(dic["poz"],2));
        else
            if((type == "K")&&(num == 1))
                res =
(dic["as3"]*Math.Pow((dic["T"]/H),3)+dic["bs3"]*Math.Pow((dic["T"]/H),2)+dic["c
s3"]*(dic["T"]/H)+dic["d"]);
            else
                if((type == "K")&&(num == 2))
                    res =
(dic["a3"]*Math.Pow((dic["T"]/H),3)+dic["b3"]*Math.Pow((dic["T"]/H),2)+dic["c3"
]*dic["T"]/H)+dic["d"]);
            else
                if((type == "K")&&(num == 3))
                    res =
(dic["as2"]*Math.Pow((dic["T"]/H),3)+dic["bs2"]*Math.Pow((dic["T"]/H),2)+dic["c
s2"]*(dic["T"]/H)+dic["d"]);
            else
                if((type == "K")&&(num == 4))
                    res =
(dic["a2"]*Math.Pow((dic["T"]/H),3)+dic["b2"]*Math.Pow((dic["T"]/H),2)+dic["c2"
]*dic["T"]/H)+dic["d"]);
            else
                if(type == "n_yb")
                    res =

```

```

dic["nyb"]*formul("K",4,H)+dic["nyd"]*dic["ksib"]*cstFppn+dic["summab"]*dic["n
ybb"]*dic["ksibb"];
    else
        if(type == "m_yb")
            res = dic["myb"]*formul("K",3,H)-
dic["nyd"]*(dic["lp"]/dic["L"])*dic["ksib"]*cstFppn-
dic["summab"]*dic["nybb"]*dic["ksibb"]*(dic["lb"]/dic["L"]);
        else
            if(type == "n_yw")
                res =
dic["nyw"]*formul("K",2,H)+dic["nyd"]*(dic["lp"]/dic["L"])*dic["ksiw"]*cstFppn+d
ic["summab"]*dic["nybb"]*dic["ksiwb"]*(dic["lb"]/dic["L"]);
            else
                if(type ==
"m_yw")
                    res =
dic["myw"]*formul("K",1,H)-
dic["nyd"]*Math.Pow((dic["lp"]/dic["L"]),2)*dic["ksiw"]*cstFppn-
dic["summab"]*dic["nybb"]*dic["ksiwb"]*Math.Pow((dic["lb"]/dic["L"]),2);
                else
                    if(type ==
"_nyw")
                        res =
formul("m",11,H) - formul("n_yw",0,H);
                    else
                        if((type == "k")&&(num == 111))
                            res = (-
dic["ass1"]*Math.Pow((dic["T"]/H),3)+dic["bss1"]*Math.Pow((dic["T"]/H),2)+dic["
css1"]*(dic["T"]/H)+dic["d"]);
                        else

```

```
if((type == "m")&&(num == 111))
    res =
(2*dic["Cb"]*dic["BB"]*(dic["po"]*dic["W"]+dic["alfa11"]*formul("k",111,H)))/(di
c["sigma"]*dic["L"]*dic["po"]*dic["W"]);
    return res;
}
}
```

**Подпрограмма вывода меню для задания начальных значений параметров судна при движении судна на мелководье по криволинейной траектории (menuSelectScene)**

```

using UnityEngine;
using System.Collections;
using System;
using System.Xml;

public class menuSelectScene : MonoBehaviour {
    //Объявление открытых переменных
    public moveShip[] ms;
    public Transform movCam;
    public GUIText title;
    public int startR = 0;
    //Объявление закрытых переменных
    private int flag = 1, countShips = 1;
    private bool winExit = false;
    private float sX = 200f, sY = 200f, tX = 200f, tY = 200f, ZZ = 22.5f, TT = 30f,
amp = 20f, T2 = 300f;
    private double KK = Math.PI / 2, DD = Math.PI / 6, HH = 0.005, VV = 10, alfa =
0f,
        m = 0.3, n = 0.3, K1 = 20, K2 = 200, K4 = 1, K3 = 1, A = 50, bendX =
100f, bendLength = 100;
    private string statusWater = "Вкл";
    private bool sin = false, sinY = false;
    private bool perspective = true;
    private bool hideMenu = false;
    private bool existConfigFile = false;

```



```

public GameObject axisX, axisY;
    //Функция вызываемая первый раз после запуска программы
    //Построение dna
    //Установка начальных значений
    void Start () {
        //Получение доступа к открытым переменным из файла moveShip.cs
        loadConfig();
        initiateAxis(axisX, true);
        initiateAxis(axisY, false);
            ms = GameObject.Find("ship").GetComponent<moveShip>();
            //Построение dna
            createPlane(2,20);
            ms[0].currentPlane = 1;
            Arguments arguments = new Arguments(m, n, K1, K2, K3, K4, alfa, A,
sin, bendX, bendLength);
            ms[0].nullVar(sX, sY, tX, tY, KK, DD, ZZ, HH, TT, VV, arguments);
        }
private void initiateAxis(GameObject axis, bool X)
{
    int i = 0;
    if (!X)
        i = 100;
    for (; i <= 5000; i = i + 100)
    {
        GameObject gameObject = new GameObject();
        gameObject.AddComponent<TextMesh>();
        gameObject.AddComponent<MeshRenderer>();
        string value = i.ToString() + " м";
        if (i == 5000 && X)
            value = "X";
    }
}

```

```

else if (i == 5000 && !X)
    value = "Y";
gameObject.GetComponent<TextMesh>().text = value;
if(X)
    gameObject.GetComponent<TextMesh>().transform.position = new
Vector3(i - 20, 0, 0);
else
    gameObject.GetComponent<TextMesh>().transform.position = new
Vector3(-70, 0, i + 10);
gameObject.GetComponent<TextMesh>().fontSize = 20;
gameObject.GetComponent<TextMesh>().color = Color.green;
gameObject.GetComponent<TextMesh>().transform.localScale = new
Vector3(10, 10, 10);
gameObject.GetComponent<TextMesh>().transform.Rotate(90, 0, 0);
gameObject.transform.SetParent(axis.transform);
}
}
//Вызывается один раз за кадр
void Update() {
    //Если нажата клавиша Escape то вызывается меню
    //т.е. значение переменной flag меняется на противоположный
    if (Input.GetKeyDown("escape") && (winExit == false)){
        flag ^= 1;
    }
}
private void loadConfig()
{
    existConfigFile = System.IO.File.Exists("config.xml");
    if (existConfigFile)
    {

```

```

XmlDocument loadXmlConfig = new XmlDocument();
loadXmlConfig.Load("config.xml");
System.Reflection.FieldInfo[] fi =
this.GetType().GetFields(System.Reflection.BindingFlags.NonPublic |
System.Reflection.BindingFlags.Instance);
XmlNode root = loadXmlConfig.DocumentElement;
IEnumerator ienum;
XmlElement param;
foreach (System.Reflection.FieldInfo i in fi)
{
    if (i.FieldType == typeof(float) || i.FieldType ==
typeof(double) || i.Name == "sin")
    {
        ienum = root.GetEnumerator();
        while(ienum.MoveNext())
        {
            param = (XmlElement)ienum.Current;
            if (i.Name.Equals(param.Name))
            {
                if (i.FieldType == typeof(double))
                    i.SetValue(this,
Convert.ToDouble(param.GetAttribute("value")));
                if (i.FieldType == typeof(float))
                    i.SetValue (this, Convert.ToSingle
(param.GetAttribute ("value")));
                if (i.Name == "sin")
                    i.SetValue (this,
Convert.ToBoolean(param.GetAttribute ("value")));
            }
        }
    }
}

```

```

        }
    }
}
private void saveConfig()
{
    existConfigFile = System.IO.File.Exists("config.xml");
    if (existConfigFile)
    {
        System.IO.File.Delete("config.xml");
    }
    XmlTextWriter xmlConfigWriter = new XmlTextWriter("config.xml",
System.Text.Encoding.UTF8);
    xmlConfigWriter.WriteStartDocument();
    xmlConfigWriter.WriteStartElement("head");
    xmlConfigWriter.WriteEndElement();
    xmlConfigWriter.Close();
    XmlDocument saveXmlConfig = new XmlDocument();
    saveXmlConfig.Load("config.xml");
    System.Reflection.FieldInfo[] fi =
this.GetType().GetFields(System.Reflection.BindingFlags.NonPublic |
System.Reflection.BindingFlags.Instance);
    foreach(System.Reflection.FieldInfo i in fi) {
        if (i.FieldType == typeof(float) || i.FieldType == typeof(double) ||
i.Name == "sin")
        {
            XmlNode tag = saveXmlConfig.CreateElement(i.Name);
            saveXmlConfig.DocumentElement.AppendChild(tag);
            XmlAttribute value = saveXmlConfig.CreateAttribute("value");
            value.Value = i.GetValue(this).ToString();

```

```

        tag.Attributes.Append(value);
    }
}
saveXmlConfig.Save("config.xml");
}
//Объявление переменных размера окна меню
private Rect winRect1 = new Rect(Screen.width-700,10,650,400);
private Rect winRect2 = new Rect((Screen.width/2)-105,(Screen.height/2)-
40,215,80);
private Rect winRect = new Rect(Screen.width-700,10,650,400);
//стандартное значение бегунка приближения камеры
private float hSliderValue = 50.0f;
private float hValue = 50.0f;
//Функция отвечающая за прорисовку окна меню
void OnGUI(){
    //Если значение переменной равно true, то ...
    if(flag == 1) {
        //остановить моделирование
        Time.timeScale = 0;
        //и вывести меню
        if(!hideMenu)
            GUI.Window(0, winRect, myFunction, "MENU");
    } else {
        // иначе продолжить моделирование
        Time.timeScale = 1;
    }
    //присвоение переменной hSliderValue текущего значения
приближения камеры
    hSliderValue = movCam.GetComponent<Camera>().fieldOfView;
    hValue = movCam.GetComponent<Camera>().orthographicSize;

```

```

//Если hSliderValue меньше 20 то присвоить ей 20
//если hSliderValue больше 165 то присвоить 165
if (hSliderValue <= 20f)
    hSliderValue = 20f;
else if (hSliderValue >= 178f)
    hSliderValue = 178f;
hSliderValue = GUI.HorizontalSlider(new Rect(25, 25, 130, 30),
hSliderValue, 20.0f, 178.0f);
hValue = GUI.HorizontalSlider(new Rect(25, 55, 130, 30), hValue, 1.0f,
1500.0f);
GUI.color = Color.red;
perspective = GUI.Toggle(new Rect(160, 20, 20, 20), perspective, "");
perspective = GUI.Toggle(new Rect(160, 50, 20, 20), !perspective, "");
hideMenu = GUI.Toggle(new Rect(5, 5, 20, 20), hideMenu, "");
//Присвоение камере измененного приближения
movCam.GetComponent<Camera>().fieldOfView = hSliderValue;
movCam.GetComponent<Camera>().orthographic = perspective;
movCam.GetComponent<Camera>().orthographicSize = hValue;
}
//Прорисовка элементов меню и изменение значений параметров через
МЕНЮ
void myFunction(int winID) {
    if(winExit != true){
        winRect = winRect1;
        if(GUI.Button(new Rect(10,20,130,25), "Ровное дно")){
saveConfig();
            Time.timeScale = 1;
            flag ^= 1;
            createPlane(2,ZZ);
            Arguments arguments = new Arguments(m, n, K1, K2, K3,

```

```

K4, alfa, A, sin, bendX, bendLength);
    ms[0].nullVar(sX, sY, tX, tY, KK, DD, ZZ, HH, TT, VV, arguments);
        ms[0].outtofile("-----Ровное дно-----");
ms[0].currentPlane = 2;
if (countShips > 1) createShips(countShips);
    startR = 1;
    }
    if(GUI.Button(new Rect(10,50,130,25), "Резкие перепады")){
saveConfig();
    if(ZZ < 21f)
        ZZ = 21f;
    Time.timeScale = 1;
    flag ^= 1;
    createPlane(1,ZZ);
    Arguments arguments = new Arguments(m, n, K1, K2, K3,
K4, alfa, A, sin, bendX, bendLength);
    ms[0].nullVar(sX, sY, tX, tY, KK, DD, ZZ, HH, TT, VV, arguments);
        ms[0].outtofile("-----Резкие перепады-----");
    ms[0].currentPlane = 1;
    startR = 1;
    }
    if(GUI.Button(new Rect(10,80,130,25), "Кривое дно")){
saveConfig();
    if(ZZ < 21f)
        ZZ = 21f;
    Time.timeScale = 1;
    flag ^= 1;
    createPlane(0,ZZ);
    Arguments arguments = new Arguments(m, n, K1, K2, K3,
K4, alfa, A, sin, bendX, bendLength);

```

```

ms[0].nullVar(sX, sY, tX, tY, KK, DD, ZZ, HH, TT, VV, arguments);
    ms[0].outtofile("-----Кривое дно-----");
    ms[0].currentPlane = 0;
    startR = 1;
}
if(GUI.Button(new Rect(145,20,55,40), "Выход")){
saveConfig();
    winExit = true;
}
if(GUI.Button(new Rect(145,65,55,40), statusWater)){
    ms[0].Water.active ^= true;
    if(ms[0].Water.active)
        statusWater = "Выкл";
    else
        statusWater = "Вкл";
}
ms[0].focus = true;
//Расстановка элементов меню
GUI.Box(new Rect(5,110,211,267), "");
GUI.Box(new Rect(219,110,211,267), "");
GUI.Box(new Rect(433,110,211,267), "");
GUI.color = Color.red;
GUI.Label(new Rect(11,110,150,20), "Координаты судна:");
GUI.color = Color.white;
GUI.Label(new Rect(15,130,15,20), "X:");

try
{
    sX = Convert.ToSingle(GUI.TextField(new Rect(60, 131, 145, 18),
sX.ToString()));
}

```



```

catch
{
    sX = 1;
}

    GUI.Label(new Rect(15,150,15,20), "Y:");

try
{
    sY = Convert.ToSingle(GUI.TextField(new Rect(60, 151, 145, 18),
sY.ToString()));
}
catch
{
    sY = 1;
}

    GUI.color = Color.red;
    GUI.Label(new Rect(11,170,100,20), "Курс судна:");
    GUI.color = Color.white;
    GUI.Label(new Rect(15,190,40,20), (KK * 180 / Math.PI).ToString());
    KK = GUI.HorizontalSlider(new Rect(60, 195, 145, 20), (float)KK, 0.0f,
6.28f);

    GUI.color = Color.red;
    GUI.Label(new Rect(11,210,190,20), "Угол перекладки руля:");
    GUI.color = Color.white;
    GUI.Label(new Rect(15,230,40,20),
(DD*180/Math.PI).ToString());
    DD = GUI.HorizontalSlider(new Rect(60, 235, 145, 20),
(float)DD, -0.7f, 0.7f);

    GUI.color = Color.red;
    GUI.Label(new Rect(11,250,190,20), "Глубина под килем:");
    GUI.color = Color.white;

```

```
try
{
    ZZ = Convert.ToSingle(GUI.TextField(new Rect(60, 271, 145, 20),
ZZ.ToString()));
}
catch
{
    ZZ = 1;
}

    GUI.color = Color.red;
    GUI.Label(new Rect(11,290,190,20), "Время моделирования:");
    GUI.color = Color.white;

try
{
    HH = Convert.ToSingle(GUI.TextField(new Rect(60, 311, 145, 20),
HH.ToString(), 7));
}
catch
{
    HH = 0.002;
}

    GUI.color = Color.red;
    GUI.Label(new Rect(225, 50, 200, 20), "Количество кораблей:");

try
{
    countShips = Convert.ToInt16(GUI.TextField(new Rect(370, 50, 30, 20),
countShips.ToString(), 1));
}
catch
{
```

```

    countShips = 1;
}

    GUI.color = Color.red;
    sinY = GUI.Toggle (new Rect(225, 85, 200, 20), sinY, "По оси
Y");
    GUI.color = Color.red;
    GUI.Label(new Rect(225, 110, 200, 20), "Координаты траектории:");
    GUI.color = Color.white;
    GUI.Label(new Rect(229, 130, 15, 20), "X:");
    try
    {
        tX = Convert.ToSingle(GUI.TextField(new Rect(280, 131, 145, 18),
tX.ToString()));
    }
    catch
    {
        tX = 1;
    }
    GUI.Label(new Rect(229, 150, 15, 20), "Y:");
    try
    {
        tY = Convert.ToSingle(GUI.TextField(new Rect(280, 151, 145, 20),
tY.ToString()));
    }
    catch
    {
        tY = 1;
    }

    GUI.color = Color.red;

```

```

GUI.Label(new Rect(225,170,190,20), "Время Наблюдения:");
GUI.color = Color.white;
GUI.Label(new Rect(289,188,50,20), "секунд");

try {
    TT = Convert.ToSingle(GUI.TextField(new Rect(229, 190, 50, 20),
TT.ToString(), 30));
    }
    catch {
        TT = 20f;
    }

    GUI.color = Color.red;
    GUI.Label(new Rect(225,210,190,20), "Скорость судна:");
    GUI.color = Color.white;
    GUI.Label(new Rect(289,228,50,20), "м/с");
    try {
        VV = Convert.ToSingle(GUI.TextField(new Rect(229, 230,
50, 20), VV.ToString(), 10));
    }
    catch {
        VV = 10;
    }

    GUI.color = Color.red;
    GUI.Label(new Rect(225, 284, 190, 20), "Параметры уравнения:");
    GUI.color = Color.white;
    GUI.Label(new Rect(225, 302, 50, 20), "K1");
    GUI.Label(new Rect(320, 302, 50, 20), "K2");
    try
    {
        K1 = Convert.ToSingle(GUI.TextField(new Rect(245, 302, 70, 20),
K1.ToString("0.00000"), 8));

```

```

        K2 = Convert.ToSingle(GUI.TextField(new Rect(340, 302, 70, 20),
K2.ToString("0.00000"), 8));
    }
    catch
    {
        K1 = 20;
        K2 = 200;
    }
    GUI.color = Color.red;
    GUI.Label(new Rect(225, 322, 190, 20), "Параметры уравнения:");
    GUI.color = Color.white;
    GUI.Label(new Rect(225, 342, 50, 20), "K3");
    GUI.Label(new Rect(320, 342, 50, 20), "K4");
    try
    {
        K3 = Convert.ToSingle(GUI.TextField(new Rect(245, 342, 70, 20),
K3.ToString("0.00000"), 8));
        K4 = Convert.ToSingle(GUI.TextField(new Rect(340, 342, 70, 20),
K4.ToString("0.00000"), 8));
    }
    catch
    {
        K3 = 1;
        K4 = 1;
    }

        GUI.color = Color.red;
        GUI.Label(new Rect(439, 110, 190, 20), "Угол траектории:");
        GUI.color = Color.white;
        GUI.Label(new Rect(444, 130, 20, 20), (alfa * 180 /
Math.PI).ToString("0"));

```

```

        alfa = GUI.HorizontalSlider(new Rect(463, 135, 70, 20),
(float)alfa, 0.0f, (float)Math.PI / 4);
        GUI.color = Color.red;
        GUI.Label(new Rect(549, 110, 100, 20), "Амплитуда:");
        GUI.color = Color.white;
        GUI.Label(new Rect(549, 130, 20, 20), "A");
        try
        {
                A = Convert.ToSingle(GUI.TextField(new Rect(560,
130, 50, 20), A.ToString(), 3));
        }
        catch
        {
                A = 50;
        }
        GUI.Label(new Rect(615, 130, 10, 20), "м");
        sin = GUI.Toggle (new Rect(439, 150, 90, 20), sin, "
Синусоида");
        GUI.color = Color.red;
        GUI.Label(new Rect(439, 170, 190, 20), "Координата начала
изгиба:");
        GUI.color = Color.white;
        try
        {
                bendX = Convert.ToSingle(GUI.TextField(new Rect(475, 190, 145, 20),
bendX.ToString(), 4));
        }
        catch
        {
                bendX = 1;

```

```

}

    GUI.color = Color.red;
    GUI.Label(new Rect(439, 210, 100, 20), "Длина изгиба:");
    GUI.color = Color.white;
    try
    {
        bendLength = Convert.ToSingle(GUI.TextField(new
Rect(539, 210, 50, 20), bendLength.ToString(), 3));
    }
    catch
    {
        bendLength = 100;
    }
    GUI.Label(new Rect(595, 210, 10, 20), "м");
    GUI.color = Color.red;
    GUI.Label(new Rect(439, 235, 200, 20), "Параметры дна:");
    GUI.color = Color.white;
    GUI.Label(new Rect(439, 255, 90, 20), "Амплитуда:");
    try
    {
        amp = Convert.ToSingle(GUI.TextField(new Rect(520, 255, 50, 20),
amp.ToString(), 20));
    }
    catch
    {
        amp = 20f;
    }
    GUI.Label(new Rect(575, 255, 10, 20), "м");
    GUI.Label(new Rect(439, 280, 90, 20), "Полупериод:");
    try

```

```

    {
        T2 = Convert.ToSingle(GUI.TextField(new Rect(520, 280, 50, 20),
T2.ToString(), 300));
    }
    catch
    {
        T2 = 300f;
    }
    GUI.Label(new Rect(575, 280, 10, 20), "M");
    GUI.color = Color.red;
        //МЕНЮ ВЫХОДА ИЗ ПРОГРАММЫ
} else if(winExit == true) {
    winRect = winRect2;
    GUI.Label(new Rect(10,15,200,20), "Вы действительно хотите
выйти?");
    if(GUI.Button(new Rect(40,45,55,25), "Да")){
        Application.Quit();
    }
    if(GUI.Button(new Rect(115,45,55,25), "Нет")){
        winExit = false;
    }
}
}
private void createShips(int countShips) {
    System.Random random = new System.Random();
    for (int i = 1; i < countShips; i++)
    {
        moveShip tempShip = Instantiate(ms[0]);
        tempShip.line3d = Instantiate(ms[0].line3d);
        tempShip.line3d.material.shader = Shader.Find("Unlit/Color");
    }
}

```



```

tempShip.line3d.material.SetColor("_Color", new
Color((float)random.NextDouble(), (float)random.NextDouble(),
(float)random.NextDouble()));

Arguments arguments2 = new Arguments(m, n, K1, K2, K3, K4,
alfa, A, sin, bendX, bendLength);

tempShip.nullVar(sX + i * (random.Next(4) + 1), sY - i * (random.Next(4) +
1), tX, tY, KK, DD, ZZ, HH, TT, VV, arguments2);

tempShip.outtofile("-----Ровное дно-----");

tempShip.currentPlane = 2;
}
}

public float meterToCu(double meter){
return (float)(meter / ms[0].HHeight);
}

public float gradToRad(float grad){
return (float)(grad * Math.PI/180f);
}

//Функция для рисования дна
public void createPlane(int chnPl, double Z){
float[,] terrainHeights;
int i = 0, j = 0;
int xzResol, width = 5000, height = 5000;
System.Random random = new System.Random();
ms[0].currentPlane = chnPl;
float[,] array = new float[5, 5] {{2f, 1f, 2f, 2f, 0.5f}, {2f, 1f, 1f, 2f, 2f}, {1f, 1f,
2f, 0.5f, 1f}, {0.5f, 2f, 0f, 2f, 2f}, {1f, 2f, 1f, 0.5f, 2f}};
Terrain.activeTerrain.terrainData.alphamapResolution = 1025;
Terrain.activeTerrain.terrainData.heightmapResolution = 1024;
xzResol = Terrain.activeTerrain.terrainData.heightmapResolution;
terrainHeights = Terrain.activeTerrain.terrainData.GetHeights(0, 0,

```



```

        x = j;
        terrainHeights[i, j] = meterToCu(1 +
System.Math.Sin(3 * System.Math.PI / 2 + (height / (T2 * 3)) * gradToRad(x *
1.0546875f)));
    }
}
}
//Вывод на экран информации о форме дна
Terrain.activeTerrain.terrainData.SetHeights(0, 0, terrainHeights);
terrainHeights = Terrain.activeTerrain.terrainData.GetHeights(0, 0,
xzResol, xzResol);
for (i = 0; i < xzResol; i++)
    for (j = 0; j < xzResol; j++)
    {
        var a0 = maps[i, j, 0];
            if (chnPl == 2)
                a0 = 1;
            else
                a0 = terrainHeights[i, j];
        maps[i, j, 0] = a0;
    }
Terrain.activeTerrain.terrainData.SetAlphamaps(0, 0, maps);
Terrain.activeTerrain.terrainData.size = new Vector3(width, amp,
height);
if (chnPl == 1)
    title.text = "Резкие перепады";
else
    if (chnPl == 2)
        title.text = "Ровное дно";
    else

```

```
        title.text = "Кривое дно";
    }
}
public class Arguments
{
    public double Ut = 0f, E = 0f, K1 = 0f, K2 = 0f, K3 = 0f, K4 = 0f,
        alfa = 0f, A = 0f, bendX = 0f, bendLength = 0f;
    public bool sin = false;
    public Arguments(double m, double n, double K1, double K2, double
K3,
        double K4, double alfa, double A, bool sin, double bendX,
double bendLength)
    {
        Ut = m;
        E = n;
        this.K1 = K1;
        this.K2 = K2;
        this.K3 = K3;
        this.K4 = K4;
        this.alfa = alfa;
        this.A = A;
        this.sin = sin;
        this.bendX = bendX;
        this.bendLength = bendLength;
    }
    public Arguments() { }
}
```

**Подпрограмма управления судном на мелководье по криволинейной траектории (moveShip)**

```

using UnityEngine;
using System.Collections;
using System.Collections.Generic;
using System;
using System.IO;
using System.Xml;

public class moveShip : MonoBehaviour {
    //Объявление открытых переменных
    public Terrain terra;
    public GameObject Water;
    public LineRenderer line3d, laser, trajectory;
    public GameObject dragCam;
    public GameObject point1, point2, strela;
    public TextMesh textKurs;
    public GUIText guiText, guiTime, guiData;
    public float HHeight = 2;
    public int currentPlane;
    public menuSelectScene[] mss;
    public double h, prevK, K, delta, Ut = 0, dK1, dK2, Lsm, Lsm1, Lsm2, kurs;
    double tX1 = 0, tX2 = 0;
    Boolean key1 = false, key2 = false, key3 = false;
    int firstDraw = 0;
    public bool focus = false, followAngle = false, btntm = false;
    public float x0 = 250f, y0 = 250f, tX0 = 0f, tY0 = 0f, time, timeGeneral;

```

```

//Объявление закрытых переменных
private int counter = 0, miniCount, recount = 0;
private float x, y, deltaH, Z, endTime, dx, dy;
private double B, omega, V, Q, X, Y;
private Dictionary<string, double> dic = new Dictionary<string, double>();
private Arguments arguments;
public const int length = 4500;
private int i = 0;

//Функция вызываемая первый раз после запуска программы
//Установка начальных значений
void Start() {
    //Получение доступа к открытым переменным из файла
menuSelectScene.cs
    mss =
GameObject.Find("Camera").GetComponent<menuSelectScene>();
    //Запуск функции считывания значений переменных из xml файла
mxml.xml
    inputData();
    //Установка корабля по заданным координатам
transform.position = new Vector3(x0, Z, y0);
    //Рисование луча от киля судна до дна
laser.SetVertexCount(2);
    //Задание уровня относительно дна, на котором находится судно
deltaH = Z;
    //Получить объект камеру
dragCam = GameObject.Find("Camera");
}
//Функция установки требуемых значений для параметров судна
//Функция с параметрами
public void nullVar(float sX, float sY, float tX, float tY, double K00, double

```

```

D00, float Z00, double H00, float T00, double V00, Arguments arguments){
    //Задание значений параметров судна
    K = K00; //Курс
    prevK = K00; //~~Предыдущий курс
    endTime = T00; //Время Наблюдения
    V = V00; //Скорость
    Z = Z00; //Глубина под килем
    h = H00; //Время моделирования
    delta = D00; //Угол перекладки руля
    x0 = sX; // X
    y0 = sY; // Y
    counter = 0; x = 0f; y = 0f; //deltaH = 0;
    B = 0; omega = 0; Q = 0; X = 0; Y = 0;
    //Толщина линии троектории движения судна
    line3d.SetWidth(6f, 6f);
    trajectory.SetWidth(4f, 4f);
    //Установка корабля по заданным координатам
    transform.position = new Vector3(x0, Z, y0);
    //Рисование луча от киля судна до дна
    laser.SetVertexCount(2);
    //Задание уровня относительно дна, на котором находится судно
    deltaH = Z;
    //Задание уровня водной глади
    //Сохранение времени в переменной time
    time = Time.fixedTime;
    //Обнуление переменной, хранящей интервал времени от начала до
окончания моделирования
    timeGeneral = 0f;
    tX0 = tX;
    tY0 = tY;

```

```

this.arguments = arguments;
drawTrajectory(tX, tY);
}

//Функция расчета y по заданному x, при постоянных dx и dy,
принадлежащих синусоиде
double function(double x, double dx, double dy) {
    if (arguments.sin) {
        return (arguments.A * Math.Sin (1 /
arguments.bendLength * x) + (dy - (arguments.A * Math.Sin (1 /
arguments.bendLength * dx)))));
    } else {
        if (x >= (arguments.bendX + dx) && x < (arguments.bendX
+ arguments.bendLength + dx))
            return Math.Tan (arguments.alfa) * x + (dy +
arguments.A * Math.Sin (Math.PI * (3 - (x - (arguments.bendX + dx)) /
(arguments.bendLength / 2)) / 2));
        else if (x >= (arguments.bendX + arguments.bendLength +
dx))
            return (arguments.A + dy) + Math.Tan
(arguments.alfa) * x;
        return (-arguments.A + dy) + Math.Tan (arguments.alfa) *
x;
    }
}

bool minDistance(double x, double y) {
    double tx = dx, ty = 0, l = 0, min = 2000, minX = 0, minY = 0; /*key1 = false;
key2 = false; key3 = false;*/
    bool keyEnter = false;
    //~~ Метод вычисляет минимальное расстояние(гипотенузу) от судна(x, y)
до точки на синусоиде заданной траектории

```



```

while (tx < (dx + length))
{
    ty = function(tx, dx, dy);
    l = (float)Math.Sqrt((double)((tx - x) * (tx - x) + (ty - y) * (ty - y)));
//Вычисление гипотенузы двух катетов x-tx и y-ty
    if (l < min) { //Сравнение полученной длины l и заданного min.
Далее до тех пор, пока l, не станет увеличиваться
        min = l; //Сохранение текущей длины, при заданных
значений tx и ty, до тех пор пока l уменьшается
        minX = tx; //Сохранение точек tx и ty в minX
        minY = ty; // и minY
        keyEnter = true;
    } //Сохранение значений в minX и в minY. Т.о. находится
ближайшая точка на заданной траектории движения к судну
        tx+=0.5f;
    }
    Lsm = min; //Сохранение минимального значения гипотенузы
    strela.transform.position = new Vector3((float)minX, Z + 2, (float)minY);
//Устанавливает координаты для объекта "Стрела"
    kurs = deltaKurs(minX, minY); //Вычисляется касательная и угол
между касательной и оси ординат
    strela.transform.rotation = Quaternion.Euler(0f,
(float)(kurs*180/Math.PI), 0f); //Вычисляет направление куда смотрит объект
"Стрела"
    if (minY > y)
    {
        if ((K > 0) && (K < Math.PI))
            return false;
        else
            return true;
    }
}

```

```

}
else
{
    if ((K > 0) && (K < Math.PI))
        return true;
    else
        return false;
}
}

```

//Задаёт направление движение корабля.

//С помощью двух объектов point вычисляет угол направления движения.

Курс

```

double deltaKurs(double x0, double y0) {
    double px, py, p, k1, k2, alfa, kursD = 0;
    px = x0-2f; //Определяем координату X предыдущей точки для
текущей на 2
    py = function(px, dx, dy); //Определяем координату Y предыдущей точки
для текущей на 2
    point2.transform.position = new Vector3((float)x0, Z+2, (float)y0);
    point1.transform.position = new Vector3((float)px, Z+2, (float)py);
    k1 = Math.Abs(px-x0); //Определяем катет dX
    k2 = Math.Abs(py-y0); //Определяем катет dY
    p = Math.Sqrt(k1*k1+k2*k2); //Определяем гипотенузу
    alfa = Math.Asin(k2/p); //Находим угол
    if ((px>=x0)&&(py>=y0)) {
        kursD = -(Math.PI/2)-alfa;
    } else
        if ((px<=x0)&&(py>=y0)) {
            kursD = (Math.PI/2)+alfa;
        } else

```

```

        if ((px<=x0)&&(py<=y0)) {
            kursD = (Math.PI/2)-alfa;
        } else
            if ((px>=x0)&&(py<=y0)) {
                kursD = -(Math.PI/2)+alfa;
            }

        return kursD;
    }

//Отрисовка траектории движения
//Аргументы x и y. Начальные координаты траектории
void drawTrajectory(float x, float y){
    int localCounter = 0, localMiniCounter = 0;
    double tX = x, tY;
    dy = y;
    dx = x;

//1500 длина траектории
while (localCounter < length)
{
    tX = (x + localCounter);

//расчет функцией следующей точки в траектории
    tY = function(tX, x, y);
    localMiniCounter++;
    if(localMiniCounter == 10) {
        trajectory.SetVertexCount(localCounter+1);
        trajectory.SetPosition(localCounter, new Vector3((float)tX,
Z+2, (float)tY));

        localMiniCounter = 0;
        localCounter++;
    }
}
}

```

```

firstDraw++;
}
//функция для разработчика
//необходима для вывода в файл, значений переменных, для проверки и
контроля
public void outtofile(string text)
{
    using (System.IO.StreamWriter file = new
System.IO.StreamWriter("xyt1.txt", true)) {
        file.WriteLine(text);
    }
}
//Функция запускаемая по таймеру
void FixedUpdate() {
    bool znak;
    //Если время от начала моделирования не больше времени
моделирования endTime то,
    if (timeGeneral < endTime) {
        i++;
        //Курс до решения уравнений, и курс после решения
уравнений, соответственно
        double prevK,nextK;
        prevK = K;
        minDistance(x, y);
        dK1 = -(K-kurs);/*180/Math.PI;
        Lsm1 = Lsm;
        //Функция решения диф уравнений
        melkovodie(deltaH-5);
        nextK = K;
        //Координаты судна относительно начального расположения

```

```

x = x0+(float)X;
y = y0+(float)Y;
znak = minDistance(x, y);
dK2 = -(K-kurs);/*180/Math.PI;
Lsm2 = Lsm;
if (!znak) Lsm2 *= -1;
if (Lsm2 > 0 && Lsm1 < 0 || Lsm2 < 0 && Lsm1 > 0) Lsm1 *= -1;
Ut = arguments.K1 * dK2 + arguments.K2 * ((dK2-dK1)) + arguments.K3 *
Lsm2 + arguments.K4 * ((Lsm2-Lsm1));
double ddelta = 0;
double dddelta = 0;
dddelta = delta;
ddelta = (Ut - delta) / 3;
if (Math.Abs(ddelta) > 0.07) ddelta = Math.Sign(ddelta) * 0.07;
delta += ddelta;
if (Math.Abs(delta) > 0.610865) delta = Math.Sign(delta) *
0.610865;
Debug.Log(mss[0].startR + " dK1: " + dK1.ToString("0.000") + " | dK2: " +
dK2.ToString("0.000") + " | Ut : " + Ut.ToString("0.000") + " | ddelta : " +
ddelta.ToString("0.000") + " | delta : " + delta.ToString() +
" | Lsm: " + Lsm.ToString("0.00") + " | Lsm1: " +
Lsm1.ToString("0.00") + " | Lsm2: " + Lsm2.ToString("0.00") + " | deltaH: " +
deltaH.ToString("0.000") + " | V: " + V.ToString("0.000"));
//Установка корабля по координатам полученным выше
transform.position = new Vector3(x, Z, y);
//Если переменная focus равна true то камера будет следить
за судном, т.е. камере будут присвоены координаты судна
//Иначе камера получит те координаты которые имела до
приостановки слежки
if (focus)

```

```

dragCam.GetComponent<Camera>().transform.position =
new Vector3(x, Z+22.5f, y);
//Если focus равна true и followAngle равна true то камера
будет следить и поворачиваться вслед
//за судном, т.е. камере будет присвоен курс судна,
//Иначе камера будет иметь постоянный курс направленный
на север
if (followAngle && focus)
dragCam.transform.rotation = Quaternion.Euler(90f,
(float)(K*180/Math.PI), 0f);
else
dragCam.transform.rotation = Quaternion.Euler(90f, 0, 0f);
//Присвоение судну курса полученного после решения
уравнений
transform.localRotation = Quaternion.Euler(0f,
(float)(K*180/Math.PI), 0f);
//Рисование луча от киля судна...
laser.SetPosition(0, new Vector3(x, Z-1, y));
//до дна
laser.SetPosition(1, new Vector3(x, 0, y));
//Глубина под килем судна
Vector3 pos = transform.position;
pos.y =
Terrain.activeTerrain.SampleHeight(transform.position);
deltaH = Z - pos.y; /*terra.SampleHeight(new
Vector3(x, 0, y));*/
//Текстовая метка отображающая глубину под килем судна
guiText.text = deltaH.ToString("0.00") + " м";
//Время прошедшее от начала моделирования
timeGeneral = Time.time-time;

```

```

//Текстовая метка отображающая Время, прошедшее от
начала моделирования
guiTime.text = timeGeneral.ToString("0.00");
using (System.IO.StreamWriter file = new
System.IO.StreamWriter("xyt.txt", true))
{
    file.WriteLine(" X: " + x + " y: " + y + " t: " +
timeGeneral);
}
//Текстовая метка отображающее курс судна
textKurs.text = (K*180/Math.PI).ToString("0.00°") + " " +
x.ToString("0.00") + " " + y.ToString("0.00");
guiData.text = "H= " + deltaH.ToString("0.00") + " X= " + x.ToString("0.00")
+ " Y= " + y.ToString("0.00") + " K= " + (K * 180 / Math.PI).ToString("0.00°") + "
T= " + timeGeneral.ToString("0.00");

using (System.IO.StreamWriter file = new System.IO.StreamWriter("xyt.txt",
true))
{
    file.WriteLine("-----");
    file.WriteLine("X: " + x.ToString("0.00") + " Y: " + y.ToString("0.00") + "
H: " + deltaH.ToString("0.00") + " K: " + (K * 180 / Math.PI).ToString("0.00°") + "
T: " + timeGeneral.ToString("0.00"));
}

//Счетчик используемый для отсчитывания шага рисования
точек, линии траектории, между которыми проводится эта линия
miniCount++;
//Если счетчик равен одному то...
if(miniCount == 1) {
    //рисовать следующую точку и провести к нему линию

```

от предыдущей

```

        line3d.SetVertexCount(counter+1);
        line3d.SetPosition(counter, new Vector3(x, Z+1, y));
        //Обнуляем счетчик
        miniCount = 0;
        //Счетчик количества точек
        counter++;
    }
}
}
//Вызывается один раз за кадр
void Update() {
    float x,z,fow;
    //Координаты судна
    x = dragCam.GetComponent<Camera>().transform.position.x;
    z = dragCam.GetComponent<Camera>().transform.position.z;
    //Приближение камеры
    fow = dragCam.GetComponent<Camera>().fieldOfView;
    //Если нажата клавиша со стрелкой вверх, то переместить камеру на
0.5 единиц на север
    if (Input.GetKey (KeyCode.UpArrow)) {
        z += 5f;
        dragCam.GetComponent<Camera>().transform.position = new
Vector3(x, Z+22.5f, z); //change
    }
    //Если нажата клавиша со стрелкой вниз, то переместить камеру на
0.5 единиц на юг
    if (Input.GetKey (KeyCode.DownArrow)) {
        z -= 5f;
        dragCam.GetComponent<Camera>().transform.position = new

```



```

Vector3(x, Z+22.5f, z); //change
    }
    //Если нажата клавиша со стрелкой влево, то переместить камеру на
0.5 единиц на запад
    if (Input.GetKey (KeyCode.LeftArrow)) {
        x -= 5f;
        dragCam.GetComponent<Camera>().transform.position = new
Vector3(x, Z+22.5f, z); //change
    }
    //Если нажата клавиша со стрелкой вправо, то переместить камеру
на 0.5 единиц на восток
    if (Input.GetKey (KeyCode.RightArrow)) {
        x += 5f;
        dragCam.GetComponent<Camera>().transform.position = new
Vector3(x, Z+22.5f, z); //change
    }
    if (Input.GetKey(KeyCode.UpArrow) && Input.GetKey(KeyCode.LeftShift))
    {
        z += 20f;
        dragCam.GetComponent<Camera>().transform.position = new Vector3(x, Z
+ 22.5f, z); //change
    }
    //Если нажата клавиша со стрелкой вниз, то переместить камеру на 0.5
единиц на юг
    if (Input.GetKey(KeyCode.DownArrow) &&
Input.GetKey(KeyCode.LeftShift))
    {
        z -= 20f;
        dragCam.GetComponent<Camera>().transform.position = new Vector3(x, Z
+ 22.5f, z); //change

```

```

}
//Если нажата клавиша со стрелкой влево, то переместить камеру на 0.5
единиц на запад
if (Input.GetKey(KeyCode.LeftArrow) && Input.GetKey(KeyCode.LeftShift))
{
    x -= 20f;
    dragCam.GetComponent<Camera>().transform.position = new Vector3(x, Z
+ 22.5f, z); //change
}
//Если нажата клавиша со стрелкой вправо, то переместить камеру на 0.5
единиц на восток
if (Input.GetKey(KeyCode.RightArrow) && Input.GetKey(KeyCode.LeftShift))
{
    x += 20f;
    dragCam.GetComponent<Camera>().transform.position = new Vector3(x, Z
+ 22.5f, z); //change
}

//Если нажата клавиша со знаком минус, то отдалить камеру на 2
единиц от судну
if (Input.GetKey (KeyCode.KeypadMinus)) {
    fow += 5;
    dragCam.GetComponent<Camera>().fieldOfView = fow;
} else
//иначе если нажата клавиша со знаком плюс, то приблизить камеру
на 2 единицы к судну
if (Input.GetKey (KeyCode.KeypadPlus)) {
    fow -= 5;
    dragCam.GetComponent<Camera>().fieldOfView = fow;
}
//Если нажата клавиша F, то сменить значение переменной focus на

```

```

противоположное(true->>false или false->>true)
    if (Input.GetKeyDown (KeyCode.F))
        focus ^= true;
        //Если нажата клавиша G, то сменить значение переменной
followAngle на противоположное(true->>false или false->>true)
    if (Input.GetKeyDown (KeyCode.G))
        followAngle ^= true;
        //Если нажата клавиша R, то начать моделирование сначала
    if (Input.GetKeyDown (KeyCode.R)) {
        //Прорисовка дна
        System.Random random = new System.Random();
        mss[0].createPlane(currentPlane,Z);
        LineRenderer line = new LineRenderer();
        line = Instantiate(line3d);
        line.material.shader = Shader.Find("Unlit/Color");
        line.material.SetColor("_Color", new Color((float)random.NextDouble(),
(float)random.NextDouble(), (float)random.NextDouble()));
        //Установка предыдущих значений
        nullVar(x0, y0, tX0, tY0, prevK, delta, Z, h, endTime, V,
arguments); //change
    }
}
//Функция считывания значений параметров судна из xml файла:
myxml.xml
void inputData()
{
    string fieldName = "",fieldValue = "", doubleStr = "";
    XmlTextReader reader = new XmlTextReader ("myxml.xml");
    while (reader.Read())
    {

```

```

switch (reader.NodeType)
{
    case XmlNodeType.Element:
        fieldName = reader.Name;
        fieldValue = "";
        break;
    case XmlNodeType.Text:
        fieldValue = reader.Value;
        break;
}
if ((fieldName != "tab") && (fieldName != "head") &&
(fieldName != "") && (doubleStr != fieldName)&&(fieldValue != ""))
{
    dic.Add(fieldName, ParseEx(fieldValue));
    doubleStr = fieldName;
}
}
}

```

//Функция используемая в inputData() для преобразования чисел с точкой  
в числа с запятой или наоборот

```

public double ParseEx(string str)
{
    double value;
    return
        double.TryParse(str.Replace(",", "."), out value)
        ? value
        : double.Parse(str.Replace(".", ","));
}

```

// функция решающая диф уравнения  
void melkovodie(double H){

//Объявление коэффициентов в методе Рунге-Кутты

```
double k11 = 0.0, k12 = 0.0, k13 = 0.0, k14 = 0.0,
      k21 = 0.0, k22 = 0.0, k23 = 0.0, k24 = 0.0,
      k31 = 0.0, k32 = 0.0, k33 = 0.0, k34 = 0.0,
      k41 = 0.0, k42 = 0.0, k43 = 0.0, k44 = 0.0,
      k51 = 0.0, k52 = 0.0, k53 = 0.0, k54 = 0.0,
      k61 = 0.0, k62 = 0.0, k63 = 0.0, k64 = 0.0,
      k71 = 0.0, k72 = 0.0, k73 = 0.0, k74 = 0.0;
```

//Вспомогательные переменные

```
double Db = 3;
double Bs1, Bs2, Bs3, Bs4;
double n1 = 0, m1 = 0, n2 = 0, m2 = 0, n3 = 0, m3 = 0, n4 = 0, m4 = 0;
double DDb1, DDb2, DDb3, DDb4;
```

//первая стадия метода Рунге-Кутты 4-го порядка

DDb1 =

$0.3 * \text{Math.Pow}((\omega * \text{dic}["L"] / \text{dic}["V0"]), 2) * ((\text{Db} * \text{Db}) / (\text{dic}["\sigma"] * \text{dic}["L"] * \text{dic}["T"]));$

n1 =

$(B * \text{Math.PI} * (\text{dic}["T"] / \text{dic}["L"]) * (\text{dic}["A1"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A2"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A3"] * (\text{dic}["T"] / H) + \text{dic}["A4"])) + B * \text{Math.Abs}(B) * ((1.055 + 0.251 * (\text{dic}["BB"] / \text{dic}["T"])) + 0.021 * \text{Math.Pow}((\text{dic}["BB"] / \text{dic}["T"]), 2) + 12.49 * (\text{dic}["T"] / \text{dic}["L"]) + 0.0244 * (\text{dic}["L"] / \text{dic}["T"])) - \text{dic}["ask"]) * (3.34 * \text{Math.Pow}((\text{dic}["T"] / H), 2) - 0.234 * (\text{dic}["T"] / H) + 1)) + (\omega * \text{dic}["L"] / \text{dic}["V0"]) * (\text{dic}["A5"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A6"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A7"] * (\text{dic}["T"] / H) + \text{dic}["A8"])) + B * \text{Math.Abs}(\omega * \text{dic}["L"] / \text{dic}["V0"]) * (\text{dic}["A9"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["A10"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["A11"] * (\text{dic}["T"] / H) + \text{dic}["A12"]));$

m1 =

$(B * (3.36 * ((\text{dic}["T"] / \text{dic}["L"]) + 0.0136) * (\text{dic}["Sk"] + \text{dic}["ak"]))) * (\text{dic}["B1"] * \text{Math.Pow}((\text{dic}["T"] / H), 3) + \text{dic}["B2"] * \text{Math.Pow}((\text{dic}["T"] / H), 2) + \text{dic}["B3"] * (\text{dic}["T"] / H) + \text{dic}["B4"] * 1));$

["B4"])-(0.1595-  
 0.0469\*(dic["BB"]/dic["T"])+0.00375\*Math.Pow((dic["BB"]/dic["T"]),2))\*(dic["B5"]  
 ]\*Math.Pow((dic["T"]/H),3)+dic["B6"]\*Math.Pow((dic["T"]/H),2)+dic["B7"]\*(dic["  
 T"]/H)+dic["B8"]));

Bs1 = (dic["nyd"]\*delta-  
 formul("n\_yb",0,H)\*B+formul("\_nyw",0,H)\*(omega\*dic["L"]/dic["V0"])-  
 n1\*(B\*Math.Sign(omega))\*(omega\*dic["L"]/dic["V0"]))/formul("m",22,H);

k11 = h \* ( Bs1\*(Math.Pow(dic["V0"], 2)/(dic["L"] \* V)) );

k21 = h \* (

((dic["nyd"]\*delta\*(dic["lp"]/dic["L"])+formul("m\_yb",0,H)\*B+formul("m\_yw",0,H)  
 )\*(omega\*dic["L"]/dic["V0"])+m1\*(B\*Math.Sign(omega))\*(omega\*dic["L"]/dic["V  
 0"]))/formul("m",66,H))\*Math.Pow((dic["V0"]/dic["L"]),2) );

k31 = h \* (

((1/formul("m",11,H))\*((dic["nx0"]\*Math.Pow((dic["V0"]/V),3)+DDb1)-  
 (dic["nx0"]+0.4\*dic["nyd"]\*delta\*delta))-

B\*((formul("m",22,H)/formul("m",11,H))\*(omega\*dic["L"]/dic["V0"])-

k11\*dic["L"]/dic["V0"]))\*((V/dic["V0"]))\*((Math.Pow(dic["V0"],2)/dic["L"]));

k41 = h \* (K + B);

k51 = h \* (omega);

k61 = h \* (V \* Math.Sin(K));

k71 = h \* (V \* Math.Cos(K));

//Вторая стадия метода Рунге-Кутты 4-го порядка

DDb2 = 0.3\*Math.Pow(((omega + 0.5 \*

k21)\*dic["L"]/dic["V0"],2)\*((Db\*Db)/(dic["sigma"]\*dic["L"]\*dic["T"]));

n2 = ((B + 0.5 \*

k11)\*Math.PI\*(dic["T"]/dic["L"])\*(dic["A1"]\*Math.Pow((dic["T"]/H),3)+dic["A2"]

\*Math.Pow((dic["T"]/H),2)+dic["A3"]\*(dic["T"]/H)+dic["A4"])+(B + 0.5 \*

k11)\*Math.Abs((B + 0.5 \*

k11))\*((1.055+0.251\*(dic["BB"]/dic["T"])+0.021\*Math.Pow((dic["BB"]/dic["T"]),2)

+12.49\*(dic["T"]/dic["L"])+0.0244\*(dic["L"]/dic["T"]))-

$\text{dic}["ask"] * (3.34 * \text{Math.Pow}((\text{dic}["T"]/H), 2) - 0.234 * (\text{dic}["T"]/H) + 1) + ((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]) * (\text{dic}["A5"] * \text{Math.Pow}((\text{dic}["T"]/H), 3) + \text{dic}["A6"] * \text{Math.Pow}((\text{dic}["T"]/H), 2) + \text{dic}["A7"] * (\text{dic}["T"]/H) + \text{dic}["A8"])) + (B + 0.5 * k_{11}) * \text{Math.Abs}((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]) * (\text{dic}["A9"] * \text{Math.Pow}((\text{dic}["T"]/H), 3) + \text{dic}["A10"] * \text{Math.Pow}((\text{dic}["T"]/H), 2) + \text{dic}["A11"] * (\text{dic}["T"]/H) + \text{dic}["A12"]));$

$$m_2 = (B + 0.5 * k_{11}) * (3.36 * ((\text{dic}["T"] / \text{dic}["L"]) + 0.0136) * (\text{dic}["Sk"] + \text{dic}["ak"])) * (\text{dic}["B1"] * \text{Math.Pow}((\text{dic}["T"]/H), 3) + \text{dic}["B2"] * \text{Math.Pow}((\text{dic}["T"]/H), 2) + \text{dic}["B3"] * (\text{dic}["T"]/H) + \text{dic}["B4"]) - (0.1595 - 0.0469 * (\text{dic}["BB"] / \text{dic}["T"]) + 0.00375 * \text{Math.Pow}((\text{dic}["BB"] / \text{dic}["T"]), 2)) * (\text{dic}["B5"] * \text{Math.Pow}((\text{dic}["T"]/H), 3) + \text{dic}["B6"] * \text{Math.Pow}((\text{dic}["T"]/H), 2) + \text{dic}["B7"] * (\text{dic}["T"]/H) + \text{dic}["B8"]));$$

$\text{Bs}_2 = (\text{dic}["nyd"] * \text{delta-formul}("n\_yb", 0, H) * (B + 0.5 * k_{11}) + \text{formul}("_nyw", 0, H) * ((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]) - n_2 * ((B + 0.5 * k_{11}) * \text{Math.Sign}(\omega + 0.5 * k_{21}) * ((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]))) / \text{formul}("m", 22, H);$

$$k_{12} = h * ( \text{Bs}_2 * (\text{Math.Pow}(\text{dic}["V0"], 2) / (\text{dic}["L"] * (V + 0.5 * k_{31}))) );$$

$k_{22} = h * ( ((\text{dic}["nyd"] * \text{delta} * (\text{dic}["lp"] / \text{dic}["L"]) + \text{formul}("m\_yb", 0, H) * (B + 0.5 * k_{11}) + \text{formul}("m\_yw", 0, H) * ((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"])) + m_2 * ((B + 0.5 * k_{11}) * \text{Math.Sign}(\omega + 0.5 * k_{21}) * ((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]))) / \text{formul}("m", 66, H) * \text{Math.Pow}((\text{dic}["V0"] / \text{dic}["L"]), 2) );$

$$k_{32} = h * ( (((1 / \text{formul}("m", 11, H)) * ((\text{dic}["nx0"] * \text{Math.Pow}((\text{dic}["V0"] / (V + 0.5 * k_{31})), 3) + \text{DDb}_2) - (\text{dic}["nx0"] + 0.4 * \text{dic}["nyd"] * \text{delta} * \text{delta})) - (B + 0.5 * k_{11}) * ((\text{formul}("m", 22, H) / \text{formul}("m", 11, H)) * ((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]))) - k_{12} * \text{dic}["L"] / \text{dic}["V0"]))) * (V + 0.5 * k_{31}));$$

$k_{33} = h * ( (((1 / \text{formul}("m", 11, H)) * ((\text{dic}["nx0"] * \text{Math.Pow}((\text{dic}["V0"] / (V + 0.5 * k_{31})), 3) + \text{DDb}_2) - (\text{dic}["nx0"] + 0.4 * \text{dic}["nyd"] * \text{delta} * \text{delta})) - (B + 0.5 * k_{11}) * ((\text{formul}("m", 22, H) / \text{formul}("m", 11, H)) * ((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]))) - k_{12} * \text{dic}["L"] / \text{dic}["V0"]))) * (V + 0.5 * k_{31}));$

$$k_{34} = h * ( (((1 / \text{formul}("m", 11, H)) * ((\text{dic}["nx0"] * \text{Math.Pow}((\text{dic}["V0"] / (V + 0.5 * k_{31})), 3) + \text{DDb}_2) - (\text{dic}["nx0"] + 0.4 * \text{dic}["nyd"] * \text{delta} * \text{delta})) - (B + 0.5 * k_{11}) * ((\text{formul}("m", 22, H) / \text{formul}("m", 11, H)) * ((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]))) - k_{12} * \text{dic}["L"] / \text{dic}["V0"]))) * (V + 0.5 * k_{31}));$$

$k_{35} = h * ( (((1 / \text{formul}("m", 11, H)) * ((\text{dic}["nx0"] * \text{Math.Pow}((\text{dic}["V0"] / (V + 0.5 * k_{31})), 3) + \text{DDb}_2) - (\text{dic}["nx0"] + 0.4 * \text{dic}["nyd"] * \text{delta} * \text{delta})) - (B + 0.5 * k_{11}) * ((\text{formul}("m", 22, H) / \text{formul}("m", 11, H)) * ((\omega + 0.5 * k_{21}) * \text{dic}["L"] / \text{dic}["V0"]))) - k_{12} * \text{dic}["L"] / \text{dic}["V0"]))) * (V + 0.5 * k_{31}));$

```

k31)/dic["V0"]))*Math.Pow(dic["V0"],2)/dic["L"] ) );
    k42 = h * ((K + 0.5 * k51) + (B + 0.5 * k11));
    k52 = h * ((omega + 0.5 * k21));
    k62 = h * ((V + 0.5 * k31) * Math.Sin((K + 0.5 * k51)));
    k72 = h * ((V + 0.5 * k31) * Math.Cos((K + 0.5 * k51)));
    //третья стадия метода Рунге-Кутты 4-го порядка
    DDb3 = 0.3*Math.Pow(((omega + 0.5 *
k22)*dic["L"]/dic["V0"],2)*((Db*Db)/(dic["sigma"]*dic["L"]*dic["T"]));
    n3 = ((B + 0.5 *
k12)*Math.PI*(dic["T"]/dic["L"])*(dic["A1"]*Math.Pow((dic["T"]/H),3)+dic["A2"]
*Math.Pow((dic["T"]/H),2)+dic["A3"]*(dic["T"]/H)+dic["A4"])+(B + 0.5 *
k12)*Math.Abs((B + 0.5 *
k12))*((1.055+0.251*(dic["BB"]/dic["T"])+0.021*Math.Pow((dic["BB"]/dic["T"]),2)
+12.49*(dic["T"]/dic["L"])+0.0244*(dic["L"]/dic["T"])-
dic["ask"])*(3.34*Math.Pow((dic["T"]/H),2)-0.234*(dic["T"]/H)+1))+((omega + 0.5
* k22)*dic["L"]/dic["V0"])* (dic["A5"]*Math.Pow((dic["T"]/H),3)+dic["A6"] *
Math.Pow((dic["T"]/H),2)+dic["A7"]*(dic["T"]/H)+dic["A8"])+(B + 0.5 *
k12)*Math.Abs((omega + 0.5 * k22)*dic["L"]/dic["V0"])* (dic["A9"]*Math.Pow
((dic["T"]/H),3)+dic["A10"]*Math.Pow((dic["T"]/H),2)+dic["A11"]*(dic["T"]/H)+di
c["A12"]));
    m3 = ((B + 0.5 *
k12)*(3.36*((dic["T"]/dic["L"])+0.0136)*(dic["Sk"]+dic["ak"]))* (dic["B1"]*Math.P
ow((dic["T"]/H),3)+dic["B2"]*Math.Pow((dic["T"]/H),2)+dic["B3"]*(dic["T"]/H)+di
c["B4"])-(0.1595-0.0469*(dic["BB"]/dic["T"])+0.00375*Math.Pow
((dic["BB"]/dic["T"]),2))* (dic["B5"]*Math.Pow((dic["T"]/H),3)+dic["B6"]*Math.Po
w((dic["T"]/H),2)+dic["B7"]*(dic["T"]/H)+dic["B8"]));
    Bs3 = (dic["nyd"]*delta-formul("n_yb",0,H)*(B + 0.5 *
k12)+formul("_nyw",0,H)*((omega + 0.5 * k22)*dic["L"]/dic["V0"])-n3*((B + 0.5 *
k12)*Math.Sign(omega + 0.5 * k22))*((omega + 0.5 * k22)*dic["L"]/dic ["V0"]))
/formul("m",22,H);

```



$k13 = h * (Bs3 * (Math.Pow(dic["V0"], 2) / (dic["L"] * (V + 0.5 * k32))))$   
);

$k23 = h * ($   
((dic["nyd"] \* delta \* (dic["lp"] / dic["L"]) + formul("m\_yb", 0, H) \* (B + 0.5 \*  
k12) + formul("m\_yw", 0, H) \* ((omega + 0.5 \* k22) \* dic["L"] / dic["V0"]) + m3 \* ((B + 0.5  
\* k12) \* Math.Sign(omega + 0.5 \* k22)) \* ((omega + 0.5 \*  
k22) \* dic["L"] / dic["V0"]))) / formul("m", 66, H) \* Math.Pow((dic["V0"] / dic["L"]), 2) );

$k33 = h * ($   
(((1 / formul("m", 11, H)) \* ((dic["nx0"] \* Math.Pow((dic["V0"] / (V + 0.5 \* k32)), 3) +  
DDb3) - (dic["nx0"] + 0.4 \* dic["nyd"] \* delta \* delta)) - (B + 0.5 \* k12) \* ((formul  
("m", 22, H) / formul("m", 11, H)) \* ((omega + 0.5 \* k22) \* dic["L"] / dic["V0"]) - k13 \*  
dic["L"] / dic["V0"]))) \* ((V + 0.5 \* k32) / dic["V0"]) \* (Math.Pow(dic["V0"],  
2) / dic["L"]));

$k43 = h * ((K + 0.5 * k52) + (B + 0.5 * k12));$

$k53 = h * ((omega + 0.5 * k22));$

$k63 = h * ((V + 0.5 * k32) * Math.Sin((K + 0.5 * k52)));$

$k73 = h * ((V + 0.5 * k32) * Math.Cos((K + 0.5 * k52)));$

//четвертая стадия метода Рунге-Кутты 4-го порядка

$DDb4 = 0.3 * Math.Pow(((omega +$   
k23) \* dic["L"] / dic["V0"]), 2) \* ((Db \* Db) / (dic["sigma"] \* dic["L"] \* dic["T"]));

$n4 = ((B +$   
k13) \* Math.PI \* (dic["T"] / dic["L"]) \* (dic["A1"] \* Math.Pow((dic["T"] / H), 3) + dic["A2"]  
\* Math.Pow((dic["T"] / H), 2) + dic["A3"] \* (dic["T"] / H) + dic["A4"]) + (B +  
k13) \* Math.Abs((B + k13)) \* ((1.055 + 0.251 \* (dic["BB"] / dic["T"]) + 0.021 \*  
Math.Pow((dic["BB"] / dic["T"]), 2) + 12.49 \* (dic["T"] / dic["L"]) + 0.0244 \* (dic["L"] / dic[  
"T"]) - dic["ask"]) \* (3.34 \* Math.Pow((dic["T"] / H), 2) - 0.234 \* (dic["T"] / H) + 1)) + ((omega  
+ k23) \* dic["L"] / dic["V0"]) \* (dic["A5"] \* Math.Pow((dic["T"] / H), 3) +  
dic["A6"] \* Math.Pow((dic["T"] / H), 2) + dic["A7"] \* (dic["T"] / H) + dic["A8"]) + (B +  
k13) \* Math.Abs((omega + k23) \* dic["L"] / dic["V0"]) \* (dic["A9"] \* Math.Pow  
((dic["T"] / H), 3) + dic["A10"] \* Math.Pow((dic["T"] / H), 2) + dic["A11"] \* (dic["T"] / H) + di

c["A12"]));

$$m4 = ((B + k13) * (3.36 * ((dic["T"] / dic["L"]) + 0.0136) * (dic["Sk"] + dic["ak"]))) * (dic["B1"] * Math.Pow((dic["T"] / H), 3) + dic["B2"] * Math.Pow((dic["T"] / H), 2) + dic["B3"] * (dic["T"] / H) + dic["B4"]) - (0.1595 - 0.0469 * (dic["BB"] / dic["T"]) + 0.00375 * Math.Pow((dic["BB"] / dic["T"]), 2))) * (dic["B5"] * Math.Pow((dic["T"] / H), 3) + dic["B6"] * Math.Pow((dic["T"] / H), 2) + dic["B7"] * (dic["T"] / H) + dic["B8"]));$$

$$Bs4 = (dic["nyd"] * delta - formul("n_yb", 0, H) * (B + k13) + formul("_nyw", 0, H) * ((omega + k23) * dic["L"] / dic["V0"]) - n4 * ((B + k13) * Math.Sign(omega + k23)) * ((omega + k23) * dic["L"] / dic["V0"])) / formul("m", 22, H);$$

$$k14 = h * (Bs4 * (Math.Pow(dic["V0"], 2) / (dic["L"] * (V + k33))));$$

$$k24 = h * (((dic["nyd"] * delta * (dic["lp"] / dic["L"]) + formul("m_yb", 0, H) * (B + k13) + formul("m_yw", 0, H) * ((omega + k23) * dic["L"] / dic["V0"]) + m4 * ((B + k13) * Math.Sign(omega + k23)) * ((omega + k23) * dic["L"] / dic["V0"]))) / formul("m", 66, H) * Math.Pow((dic["V0"] / dic["L"]), 2));$$

$$k34 = h * (((1 / formul("m", 11, H)) * ((dic["nx0"] * Math.Pow((dic["V0"] / (V + k33)), 3) + DDb4) - (dic["nx0"] + 0.4 * dic["nyd"] * delta * delta)) - (B + k13) * ((formul("m", 22, H) / formul("m", 11, H)) * ((omega + k23) * dic["L"] / dic["V0"]) - k14 * dic["L"] / dic["V0"]))) * ((V + k33) / dic["V0"]) * (Math.Pow(dic["V0"], 2) / dic["L"]));$$

$$k44 = h * ((K + k53) + (B + k13));$$

$$k54 = h * ((omega + k23));$$

$$k64 = h * ((V + k33) * Math.Sin((K + k53)));$$

$$k74 = h * ((V + k33) * Math.Cos((K + k53)));$$

//итерационные формулы для нахождения следующих значений

$$\text{double } dB = (k11 + (2 * k12) + (2 * k13) + k14) / 6;$$

$$B = B + dB;$$

$$\text{double } \text{domega} = (k21 + (2 * k22) + (2 * k23) + k24) / 6;$$

```

omega = omega + domega;
double dV = (k31 + (2 * k32) + (2 * k33) + k34) / 6;
V = V + dV;
K = K + h*omega;
Q = K + B;
X = X + h*V*Math.Sin(Q);
Y = Y + h*V*Math.Cos(Q);

```

```

}

```

//Подстановка нужной формулы в функцию выше, формула выбирается по параметром введенным в формулу

```

double formul(string type, int num, double H)
{
    double res = 0;
    double cstFppn = (1-
(dic["Fsppn"]/dic["Fppn"])*(Math.Sqrt(1+dic["sigmap"])-
1)/(Math.Sqrt(1+dic["sigmap"])+1));

    if((type == "k")&&(num == 11))
        res =
(dic["ass1"]*Math.Pow((dic["T"]/H),3)+dic["bss1"]*Math.Pow((dic["T"]/H),2)+dic["
css1"]*(dic["T"]/H)+dic["d"]));
    else
        if((type == "k")&&(num == 22))
            res =
(dic["a1"]*Math.Pow((dic["T"]/H),3)+dic["b1"]*Math.Pow((dic["T"]/H),2)+dic["c1"
]*dic["T"]/H)+dic["d"]);
    else
        if((type == "k")&&(num == 66))
            res =
(dic["as1"]*Math.Pow((dic["T"]/H),3)+dic["bs1"]*Math.Pow((dic["T"]/H),2)+dic["c

```

```

s1"*(dic["T"]/H)+dic["d"]);

else

if((type == "m")&&(num ==
11))

res =

(2*dic["Cb"]*dic["BB"]*(dic["po"]*dic["W"]+dic["alfa11"]*formul("k",11,H)))/(dic
["sigma"]*dic["L"]*dic["po"]*dic["W"]);

else

if((type == "m")&&(num == 22))

res =

(2*dic["Cb"]*dic["BB"]*(dic["po"]*dic["W"]+dic["alfa22"]*formul("k",22,H)))/(dic
["sigma"]*dic["L"]*dic["po"]*dic["W"]);

else

if((type == "m")&&(num == 66))

res =

(2*dic["Cb"]*dic["BB"]*Math.Pow(dic["poz"]/dic["L"],2)*(dic["po"]*dic["W"]*Mat
h.Pow(dic["poz"],2)+dic["alfa66"]*formul("k",66,H)))/(dic["sigma"]*dic["L"]*dic["
po"]*dic["W"]*Math.Pow(dic["poz"],2));

else

if((type == "K")&&(num == 1))

res =

(dic["as3"]*Math.Pow((dic["T"]/H),3)+dic["bs3"]*Math.Pow((dic["T"]/H),2)+dic["c
s3"]*(dic["T"]/H)+dic["d"]);

else

if((type == "K")&&(num == 2))

res =

(dic["a3"]*Math.Pow((dic["T"]/H),3)+dic["b3"]*Math.Pow((dic["T"]/H),2)+dic["c3"
]*dic["T"]/H)+dic["d"]);

else

if((type == "K")&&(num == 3))

```

```

res =
(dic["as2"]*Math.Pow((dic["T"]/H),3)+dic["bs2"]*Math.Pow((dic["T"]/H),2)+dic["cs2"]*(dic["T"]/H)+dic["d"]);
else
if((type == "K")&&(num == 4))
res =
(dic["a2"]*Math.Pow((dic["T"]/H),3)+dic["b2"]*Math.Pow((dic["T"]/H),2)+dic["c2"]*(dic["T"]/H)+dic["d"]);
else
if(type == "n_yb")
res =
dic["nyb"]*formul("K",4,H)+dic["nyd"]*dic["ksib"]*cstFppn+dic["summab"]*dic["nybb"]*dic["ksibb"];
else
if(type == "m_yb")
res = dic["myb"]*formul("K",3,H)-
dic["nyd"]*(dic["lp"]/dic["L"])*dic["ksib"]*cstFppn-
dic["summab"]*dic["nybb"]*dic["ksibb"]*(dic["lb"]/dic["L"]);
else
if(type == "n_yw")
res =
dic["nyw"]*formul("K",2,H)+dic["nyd"]*(dic["lp"]/dic["L"])*dic["ksiw"]*cstFppn+
dic["summab"]*dic["nybb"]*dic["ksiwb"]*(dic["lb"]/dic["L"]);
else
if(type ==
"m_yw")
res =
dic["myw"]*formul("K",1,H)-
dic["nyd"]*Math.Pow((dic["lp"]/dic["L"]),2)*dic["ksiw"]*cstFppn-
dic["summab"]*dic["nybb"]*dic["ksiwb"]*Math.Pow((dic["lb"]/dic["L"]),2);

```

```

else
    if(type ==
        "_nyw")
            res =
                formul("m",11,H) - formul("n_yw",0,H);
            else
                if((type == "k")&&(num == 111))
                    res = (-
                        dic["ass1"]*Math.Pow((dic["T"]/H),3)+dic["bss1"]*Math.Pow((dic["T"]/H),2)+dic["
                        css1"]*(dic["T"]/H)+dic["d"]);
                else
                    if((type == "m")&&(num == 111))
                        res =
                            (2*dic["Cb"]*dic["BB"]*(dic["po"]*dic["W"]+dic["alfa11"]*formul("k",111,H)))/(di
                            c["sigma"]*dic["L"]*dic["po"]*dic["W"]);
                    return res;
                }
            }
    }

```