

21. Szczecinski N. S. et al. Design process and tools for dynamic neuromechanical models and robot controllers. *Biological cybernetics*, 2017, vol. 111, no.1, pp. 105–127.
22. Tedeschi F. et al. Design issues for hexapod walking robots. *Robotics*, 2014, vol. 3, no. 2, pp. 181–206.
23. Vukobratovic M. et al. ZMP: A review of some basic misunderstandings. *International Journal of Humanoid Robotics*, 2006, vol. 3, no. 2, pp. 153–175.
24. Xiong X. et al. Adaptive and energy efficient walking in a hexapod robot under neuromechanical control and sensorimotor learning. *IEEE transactions on cybernetics*, 2016, vol. 46, no. 11, pp. 2521–2534.

УДК 004.896, 629.584

## ИНТЕЛЛЕКТУАЛЬНОЕ УПРАВЛЕНИЕ ПОДВОДНЫМ РОБОТОМ НА ОСНОВЕ ИСКУССТВЕННОЙ НЕЙРОННОЙ СЕТИ

Статья поступила в редакцию 10.11.2018, в окончательном варианте – 13.02.2019.

**Свищев Николай Дмитриевич**, Астраханский государственный университет, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,  
магистрант, e-mail: svishchev.nikolai@gmail.com

**Рыбаков Алексей Владимирович**, Астраханский государственный университет, 414056, Российская Федерация, г. Астрахань, ул. Татищева, 20а,  
кандидат физико-математических наук, директор физико-математического института, e-mail: rybakov\_alex@mail.ru

В последнее время для выполнения работ в сложных условиях при ремонте подводных частей инженерных сооружений, визуальном осмотре оснований буровых платформ, корпусов судов, подводных трубопроводов и кабелей все чаще применяются подводные роботы. Как правило, это полуавтономные аппараты, в которых основными задачами автоматического управления являются контроль положения и/или скорости робота. При этом актуальность приобретают задачи точного позиционирования робота в условиях изменений скоростей течений, удержания робота в перпендикулярном осматриваемой плоскости положении на строго определенной глубине, минимизации ошибок при перемещении аппарата по заданной траектории. Использование нейронных сетей для построения адаптивных регуляторов позволяет решить сформулированные задачи. Задача позиционирования в условиях изменяющейся внешней среды может быть решена с использованием нейросетевого регулятора. В статье представлено описание конструкции, кинематики и динамики робота для выполнения визуальных осмотров подводных сооружений, разработана его математическая модель, основанная на теории управления подводных аппаратов, и приведены результаты симуляции в Matlab. Результаты, полученные при использовании модели нейросетевого регулятора, сравниваются с результатами для одной из самых распространенных моделей – ПИД регулятора.

**Ключевые слова:** нейронная сеть, подводный робот, интеллектуальное управление, позиционирование, динамика подводных аппаратов, нейросетевой регулятор, корректирующий фильтр, кинематика вязкой среды

Графическая аннотация (Graphical annotation)



## INTELLECTUAL CONTROL OF UNDERWATER ROBOT BASED ON ARTIFICIAL NEURAL NETWORK

*The article was received by editorial board on 10.11.2018, in the final version – 13.02.2019.*

**Svishev Nikolay D.**, Astrakhan State University, 20a Tatischev St., Astrakhan., 414056, Russian Federation, student, e-mail: timurkyun@mail.ru

**Rybakov Alexey V.**, Astrakhan State University, 20a Tatischev St., Astrakhan., 414056, Russian Federation, Cand. Sci. (Physics and Mathematics), Director of the Institute for Research and Solution of Technological Problems, e-mail: rybakov\_alex@mail.ru

Recently, underwater robots have been increasingly used to perform work in difficult conditions in the repair of underwater parts of engineering structures, visual inspection of the bases of drilling platforms, ship hulls, underwater pipelines and cables. As a rule, these are semi-Autonomous devices in which the main tasks of automatic control are to control the position and/or speed of the robot. At the same time, the problems of precise positioning of the robot in the conditions of changes in the flow velocities, keeping the robot perpendicular to the inspected plane position at a strictly defined depth, minimizing errors when moving the device along a given trajectory become relevant. The use of neural networks for the construction of adaptive controllers allows to solve the formulated problems. The problem of positioning in a changing environment can be solved using a neural network controller. The article presents a description of the design, kinematics and dynamics of the robot to perform visual inspections of underwater structures, developed its mathematical model based on the theory of control of underwater vehicles, and the results of simulation in Matlab. The results obtained using the neural network controller model are compared with the results for one of the most common models – pid controller.

**Key words:** neural network, underwater robot, intelligent control, positioning, dynamics of underwater vehicles, neural network controller, corrective filter, viscous medium kinematics

**Введение.** Использование подводных роботов (ПР) имеет очень широкий спектр применений в судостроении, строительстве гидротехнических сооружений, буровых платформ, в области мониторинга экологической обстановки морей, океанов и др. Обеспечить точное перемещение ПР по заданной пространственной траектории – важная цель, которая нуждается в использовании систем контроля высокого качества. При этом ПР можно рассматривать как многомерный нелинейный и неопределенный контролируемый объект, процедура разработки законов управления которым представляется сложной задачей. Именно из-за отсутствия полноты знания об управляемой системе использование стандартных регуляторов не представляется возможным.

Зачастую большинство ПР не способно контролировать свое перемещение в полном трехмерном пространстве. Так, например, подводный аппарат «ГНОМ» оснащен всего тремя маршевыми двигателями и двумя вспомогательными. Этот аппарат способен перемещаться по любому направлению в пространстве, но не способен контролировать углы тангажа и рысканья (рис. 1). Эти явления могут возникнуть вследствие сильного течения или удара аппарата, при этом единственное, что удерживает аппарат в равновесном состоянии, – это поплавок, компенсатор плавучести.



Рисунок 1 – Подводный аппарат «ГНОМ» (<https://gnomrov.ru/products/super-gnom/>)

Однако в то же время без качественно разработанного контроля управления ПР может быть излишним использование двигателей для шестистепенного позиционирования аппарата, так как математическая модель робота начинает требовать более серьезного и полного подхода в описании системы. К примеру, ПР BlueROV2 Heavy (рис. 2) компании Blue Robotics оснащен 8 двигателями, что делает данную конструкцию громоздкой и энергозатратной.



Рисунок 2 – Подводный аппарат BlueROV2 Heavy (<https://www.bluerobotics.com/store/rov/bluerov2-upgrade-kits/brov2-heavy-retrofit-r1-rp/>)

В дополнение к вышесказанному также необходимо уделить отдельное внимание корпусу аппарата. Как правило, корпуса ПР делятся на два типа: реберный и обтекаемый. В первом случае все просто – необходимо обеспечить минимальную площадь вязкого сопротивления ПР с внешней средой. Таким образом, при разработке контроллера мы сможем минимизировать диапазон внешнего возмущения или приравнять его к нулю. Второй вариант подходит больше к пилотируемым аппаратам, но также встречаются и ПР (рис. 3). Как правило, они используются в коммерческих целях и предназначены для работы на небольшой глубине.

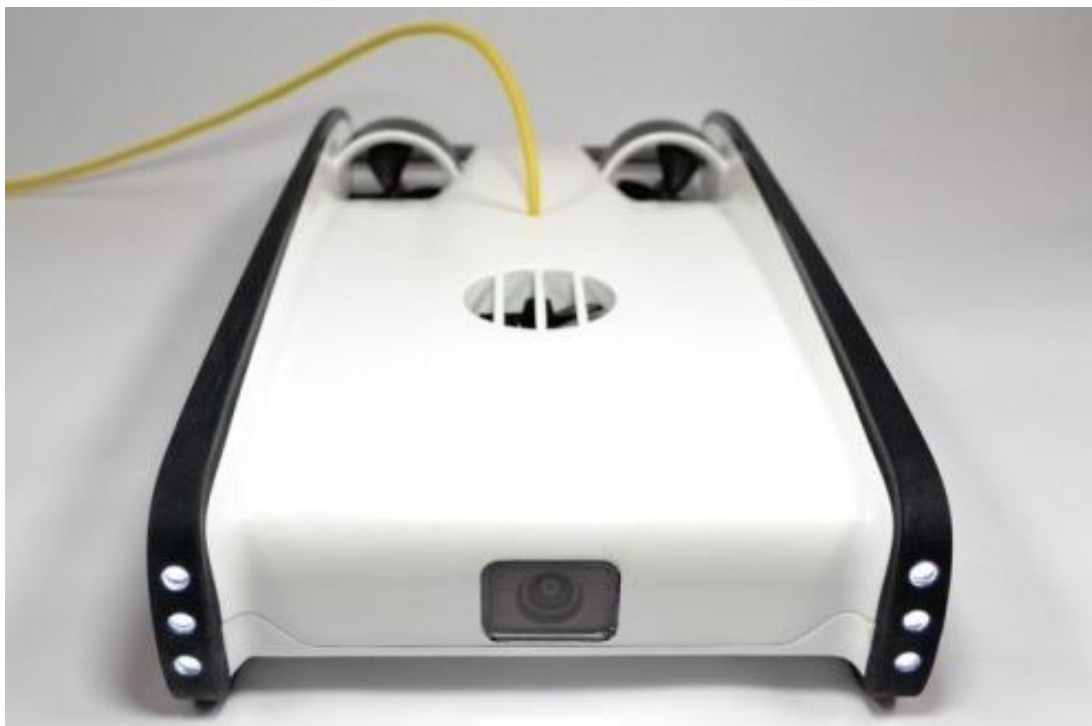


Рисунок 3 – Аппарат OpenROV Trident (<https://www.openrov.com/products/trident/>)

Таким образом, необходимо найти баланс между степенью управляемости ПР, зависящей от качественных характеристик, и сложностью описания его математической модели, которая зависит от конструктивных параметров аппарата.

*Современная теория управления выработала множество методов и подходов для решения этой проблемы, такие как нелинейная линеаризация обратной связи, адаптивный контроль системы с переменной структурой и т.д.*

Возможный способ преодоления проблем синтеза законов управления можно найти в классе систем искусственного интеллекта. В частности, можно использовать нейронные сети, для которых нет необходимости полного знания об управляемой системе.

**Материал и методики проведения вычислительных экспериментов.** Многие публикации были посвящены проблемам идентификации нейронных сетей (НС) и управления в самых различных отраслях, начиная с основополагающей работы [11]. Большинство публикаций связаны, в частности, с применением НС к задачам управления ПР [5, 6, 7]. Обычные применения многослойных НС основаны на предварительном обучении сети. Как правило, этот процесс сводит к минимуму критерий, который выражает сводку (сумму) отклонений выходов НС от желаемых значений, за короткий срок. В большинстве случаев работы подводных аппаратов, наиважнейшим критерием, является именно быстрая реакция на непредсказуемое изменение внешней среды, таких как: термоклин, донные течения и пр. Использование нейронных сетей для выбора оптимальной траектории движения роботов проанализировано в работе [3]. Там же предложена структура нейронной сети для минимизации функции ошибки следования по траектории.

Сетевое обучение приводит к корректировке весовых коэффициентов НС. Такой подход предполагает знание обучающих пар ввода-вывода [6]. Особенность использования НС как контроллера заключается в том, что желаемый сигнал контроллера заранее неизвестен. Заданная траектория движения (программный сигнал) может определяться только для всей системы управления [5, 6]. Таким образом, применение НС для управления сложным динамическим объектом является логичной и необходимой задачей в корректном позиционировании ПР. На рисунке 4 представлена структура ПР на примере робота Gnom-Pro, управляемого с поверхности с помощью оператора.

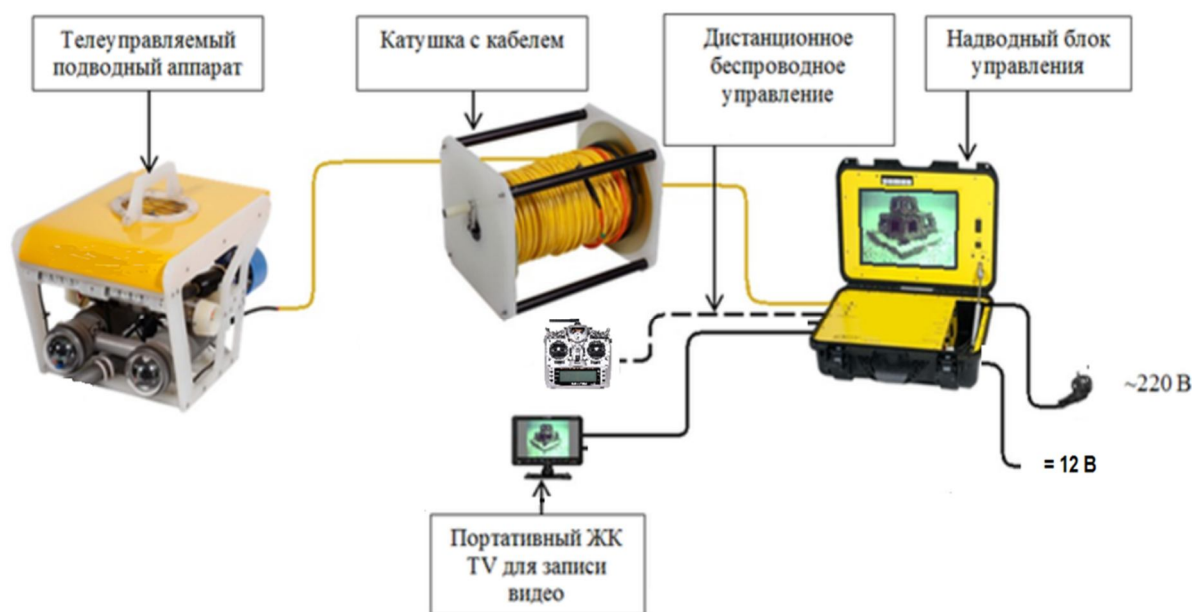


Рисунок 4 – Структурная схема ПР, управляемого с поверхности

Проектирование ПР имеет ряд особенностей, позволяющих выделить для таких аппаратов уникальные стадии проектирования. Пространственное движение подводного робота традиционно описывается в 6-ти степенях свободы, включая независимые координаты, необходимые для описания расположения и ориентации, и записывается в виде системы нелинейных векторно-матричных дифференциальных уравнений [1]. Для формирования математической модели управляемого движения применяются различные методы анализа и синтеза, разработанные в рамках динамики и кинематики твердого тела и теории автоматического управления. Для большинства ПР пространственную траекторию движения можно представить как совокупность участков движения в горизонтальной и вертикальной плоскостях, с постоянной проверкой качества перемещения с помощью датчиков. В разрабатываемом авторами аппарате (рис. 5) для отслеживания перемещения используются следующие датчики: трехосевой гироскоп, акселерометр, магнетометр и датчик давления для определения глубины погружения ПР.

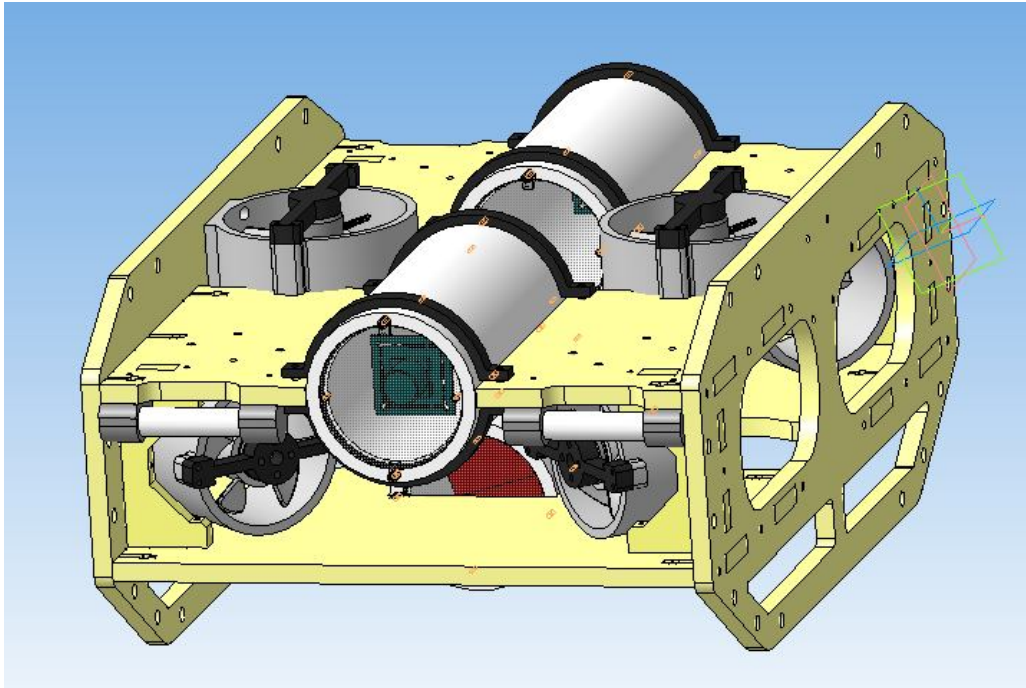


Рисунок 5 – Модель разрабатываемого авторами аппарата

Совокупность данных МЭМС сенсоров в реальном времени передают положение в микроконтроллер аппарата. Там они проходят обработку, и на выходе мы получаем координаты, углы наклона, глубину, скорость и ускорение аппарата (рис. 6).

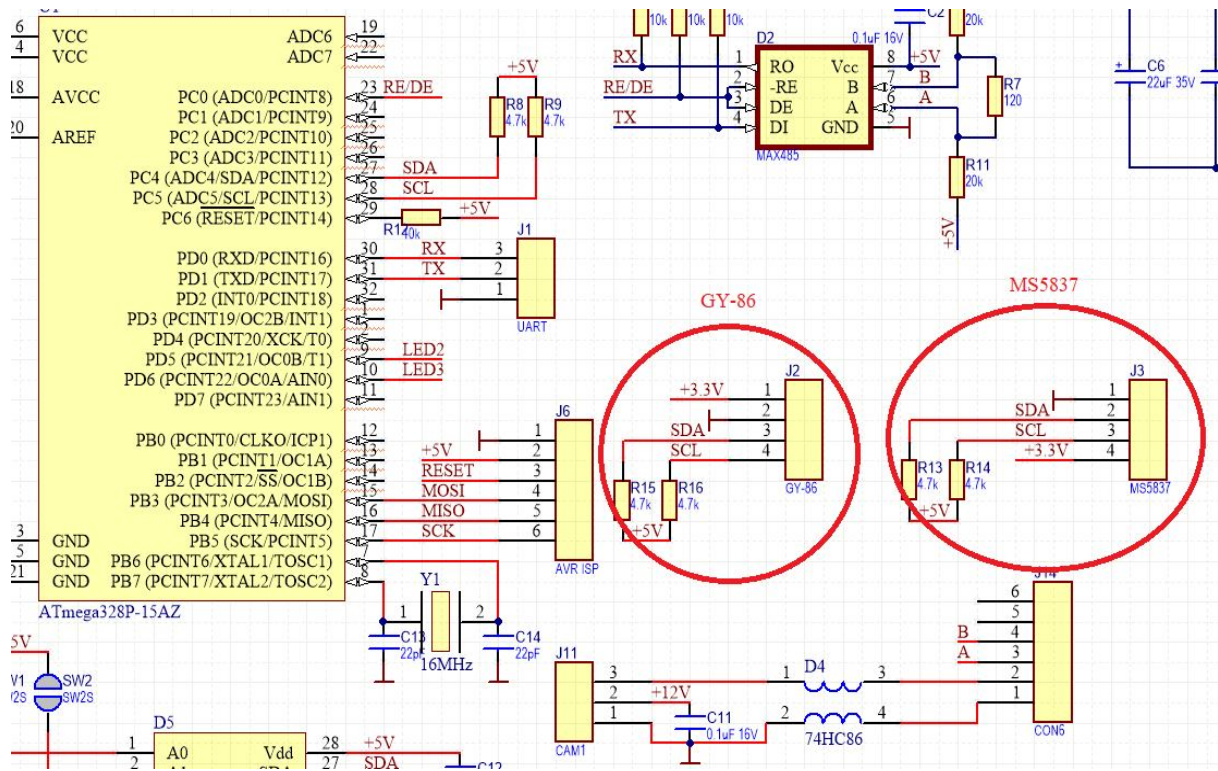


Рисунок 6 – Часть принципиальной электрической схемы с подключением датчиков к микроконтроллеру Atmega 328P, где GY-86 – совокупность гироскопа, акселерометра, магнетометра и датчика температуры, а MS5837 – датчик давления

По итогам проектирования был собран ПР с внутренним источником питания, обеспечивающим работу устройства под водой на протяжении полутора часов.

Данный аппарат оснащен 6 двигателями, видеокамерой, транслирующей видео в реальном времени, а также датчиками, контролирующими положение аппарата (рис. 7).

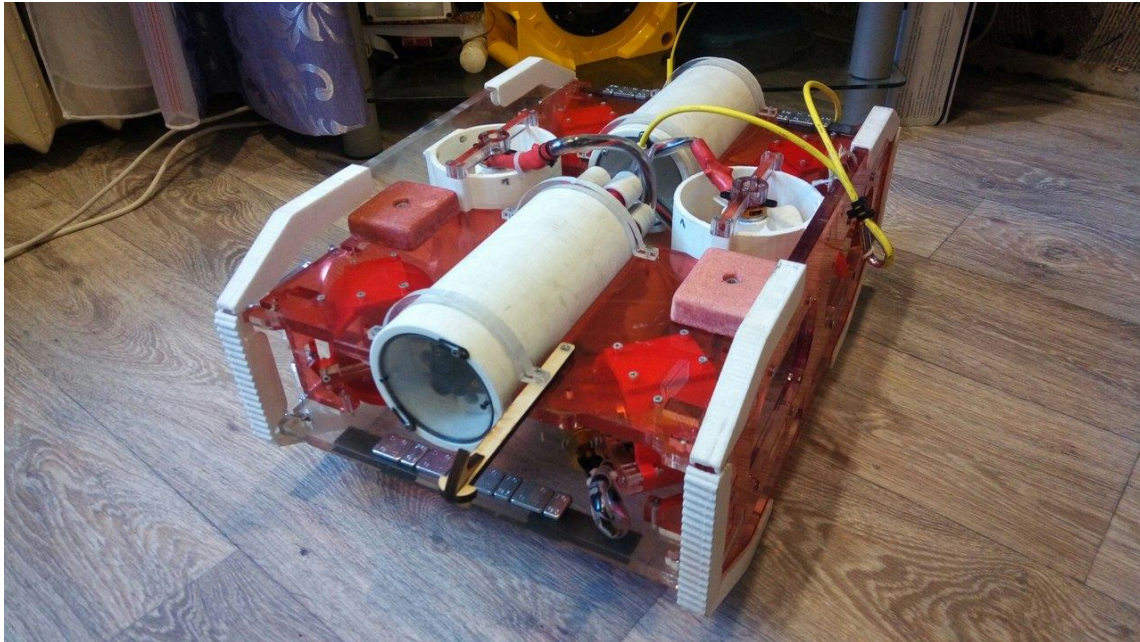


Рисунок 7 – Собранный ПР

Для синтеза математической модели управляемого движения при маневрировании в вертикальной и горизонтальной плоскостях рационально воспользоваться следующими формами с учётом кинематических условий движения для соответствующей плоскости и нулевого крена [2, 8, 9]:

– для вертикальной плоскости

$$f(x, \delta, t) = \begin{cases} D^{-1}(F - BDq); \\ \omega_z; \\ V_x \sin \theta - V_y \sin \theta; \\ V_x \sin \theta + V_y \cos \theta. \end{cases} \quad (1)$$

– для горизонтальной плоскости

$$f(x, \delta, t) = \begin{cases} D^{-1}(F - BDq); \\ \omega_y; \\ V_x \cos \psi + V_z \sin \psi; \\ -V_x \sin \psi + V_z \cos \psi. \end{cases} \quad (2)$$

где  $x$  – вектор состояния;  $\delta$  – вектор управляющих воздействий;  $q$  – вектор обобщённых скоростей;  $F$  – вектор внешних сил и моментов;  $D$  – матрица присоединённых масс;  $B$  – матрица обобщённых скоростей;  $V_x, V_y, V_z$  – линейные скорости движения по осям соответственно;  $\omega_y, \omega_z$  – угловые скорости движения;  $\psi$  – угол курса;  $\theta$  – угол дифферента.

На основе этих моделей динамики движения подводного аппарата в пространстве при построении системы управления выполняется синтез управляющих функций. Система автоматического управления движением подводного робота должна обеспечивать необходимое маневрирование в районе обследования подводного объекта. В процессе маневрирования и выполняются необходимые работы по обследованию. Путём задания различных функций управления осуществляется построение необходимой траектории маневрирования подводного робота, например для исследования подводной части гидротехнического сооружения.

Для успешного решения задач, стоящих перед подводными роботами, необходимо удовлетворение их высоким требованиям к качеству пространственного маневрирования в отношении следующих характеристик:

- точность приведения робототехнического средства в заданное место в минимально короткие сроки;
- оптимальность траектории движения – минимизация отклонений фазовых координат;
- динамическое позиционирование во время выполнения поставленной задачи.

Современные требования к повышению точности движения и наличие нелинейных составляющих в описании объекта управления и морской среды соответствуют так называемому принципу несовместимости. Он заключается в том, что с увеличением точности математического описания динамических процессов управляемого движения подводного робота существенно увеличивается неточность и неопределенность самой модели движения и описания водной среды. Кроме того, заметно усложняется математическое моделирование. Обеспечение высокой адекватности линейной модели становится практически невозможным. Проблемы высокоточного управления подводными роботами обнаруживают различные трудности из-за их нелинейной динамики, наличия внешних возмущений и шумов измерений. Это особенно проявляется в неглубоких и ограниченных водных пространствах, где фактор мелководности, динамическое волновое взаимодействие, приливно-отливное течение, прибрежное течение и искусственные сооружения создают целый ряд трудностей.

Нами был проведен расчет и моделирование гребного винта по мощности и оборотам двигателя. В дальнейшем был произведен анализ работы смоделированного гребного винта в приложении «SolidWorks – FlowSimulation». При этом устанавливались условия внешней водной среды, тела вращения (винт) и тела статики (остальные элементы сборки). Результат симуляции гребного винта – это его тяговая сила, с помощью которой можно определить (оценить) скорость перемещения ПР без проведения полевых испытаний (рис. 8).

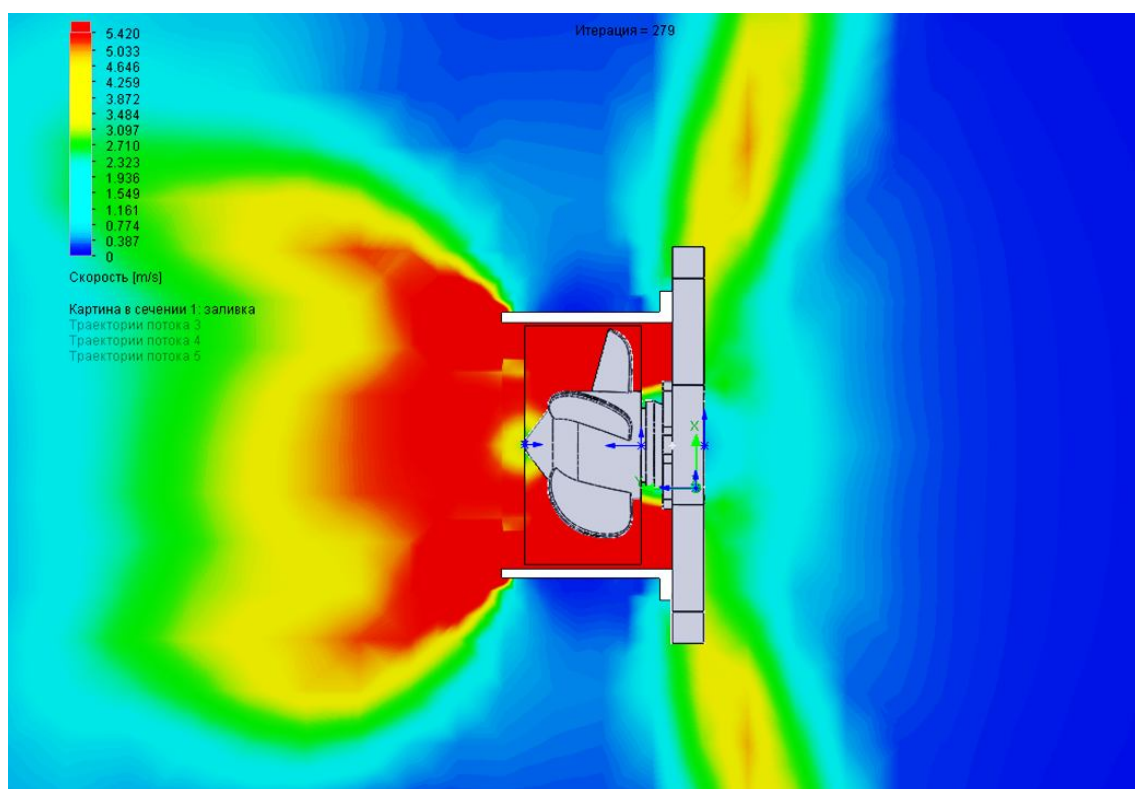


Рисунок 8 – Симуляция гребного винта в приложении «SolidWorks – FlowSimulation»

Помимо симуляции работы гребного винта ПР был также произведен расчет передаточной функции двигателя в условиях работы под водой. При этом для построения математической модели использовался метод Симою (рис. 9) на базе реальной переходной характеристики двигателя (кривой разгона, где ось абсцисс – это время в секундах, ось ординат – это обороты двигателя в минуту), полученной с помощью датчика Холла, измеряющего обороты мотора.

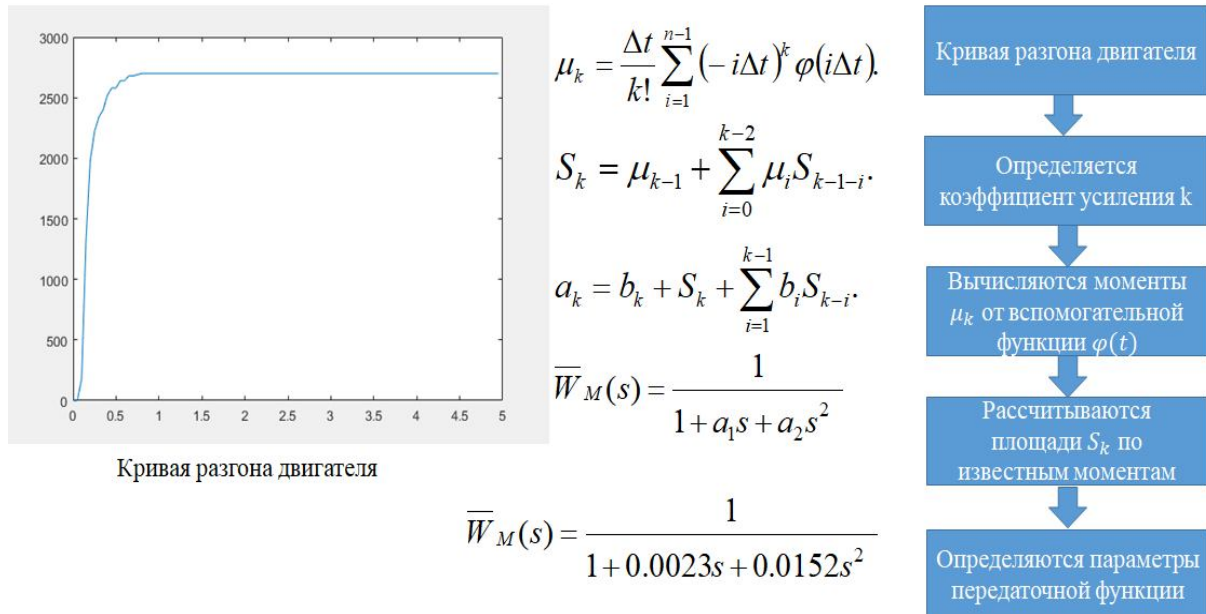


Рисунок 9 – Процесс получения передаточной функции двигателя по кривой разгона

Передаточная функция – один из основных параметров необходимых для построения полной математической системы ПР, которая будет рассматриваться в следующем разделе.

**Математическая модель перемещения подводного робота.** Если обозначить векторы состояния дистанционно управляемого аппарата по системе обозначения Общества морских архитекторов и инженеров морского флота (SNAME), то вектор линейной и угловой скорости относительно тела будет определяться следующим образом:

$$q_2 = (u, v, w, p, q_2, r)^T,$$

где  $q_2$  – виртуальный вектор;  $v$  – линейная скорость ПР;  $w$  – угловая скорость ПР;  $u$  – входной вектор управления;  $r$  – радиус-вектор скорости;  $p$  – вспомогательный вектор. Соответствующее положение относительно земли и вектор угла Эйлера определяется следующим образом:

$$q_1 = (x, y, z, \phi, \theta, \psi)^T.$$

Динамическое поведение подводного транспортного средства описывается с помощью законов линейного и углового момента Ньютона. Уравнения движения таких аппаратов являются крайне нелинейными и сочетаются с гидродинамической присоединенной массой, подъемной силой и силой сопротивления, которые действуют на транспортное средство. Удобно записать нелинейные уравнения подводного движения транспортного средства [9] как:

$$\dot{q}_1 = J(q_1)q_2, \quad (3)$$

$$D(q_1)q_2 + B(q_1, q_2)q_2 + G(q_1, q_2) = U. \quad (4)$$

где  $J$  – кинематическая матрица;  $U$  – вектор управляющих сил моментов зависящей от передаточной функции двигателей ПР,  $q_1, q_2$ ;  $D$  – матрица инерции с учетом добавленных масс воды;  $B$  – матрица Кориолиса;  $G$  – вектор обобщенной силы тяжести, плавучести и нелинейных сил демпфирования / крутящий момент [3]. Эти элементы более подробно описаны в [3, 7, 9]. Траектория движения аппарата в системе отсчета относительно земли задается кинематическим уравнением (3). Следовательно,  $J(q_1)$  можно интерпретировать как кинематическую матрицу преобразования  $n \times n$ , как правило, с функциями углов Эйлера.

По сравнению с наземными аппаратами, математическая модель ПР имеет параметры, найти которые аналитически достаточно тяжело. Поэтому параметры матриц и векторов модели ПР могут быть найдены в результате экспериментальных исследований. Как правило, этот путь затратный и занимает много времени. Один из перспективных альтернативных подходов связан с использованием интеллектуального управления ПР с использованием обученной нейронной сети (НС).

Цель такого подхода – синтез НС для ПР с целью обеспечения его точного перемещения по заданной траектории  $q_{d1}(t), q_{d2}(t)$ . Сначала рассмотрим задачу управления относительно скоростей  $q_d(t)$ . Определим ошибку:

$$e_2 = q_{d2} - q_2 \quad (5)$$

и введем локальный критерий (индекс производительности)  $Q$  как показатель разницы между желательной и реальной траекториями:

$$Q = \frac{1}{2} e_2^T D e_2. \quad (6)$$



Далее используем метод градиента скорости, разработанный в [5, 6]. Основная идея этого метода состоит в такой настройке доступных контролируемых параметров, что производная по времени выбранного локального или интегрального критерия (или их комбинации), рассчитанная вдоль траектории системы, имеет, как правило, отрицательное значение. Если это так, то критерий, который выражает цель контроля, сводится к минимуму. Согласно методу [7, 8], вычислим производную по времени  $Q$ :

$$\dot{Q} = e_2^T D \dot{e}_2 + \frac{1}{2} e_2^T \dot{D} e_2, \quad (7)$$

так как:

$$q_2 = q_{d2} - e_2 \quad (8)$$

Тогда

$$D(q_1) \dot{q}_2 = D(q_1) \dot{q}_{d2} - D(q_1) \dot{e}_2 \quad (9)$$

Используя выражение первого члена из уравнения динамики, можно получить следующее:

$$D(q_1) \dot{e}_2 = D(q_1) \dot{q}_{d2} + B(q_1, q_2) q_{d2} - B(q_1, q_2) e_2 + G(q_1, q_2) - U, \quad (10)$$

и производную по времени функции  $Q$  можно записать в виде:

$$\dot{Q} = e_2^T (D(q_1) \dot{q}_{d2} + B(q_1, q_2) q_{d2} - B(q_1, q_2) e_2 + G(q_1, q_2) - U) + \frac{1}{2} e_2^T \dot{D} e_2. \quad (11)$$

После преобразований получим:

$$\begin{aligned} \dot{Q} &= e_2^T (D(q_1) \dot{q}_{d2} + B(q_1, q_2) q_{d2} + G(q_1, q_2) - U) - e_2^T B(q_1, q_2) e_2 + \frac{1}{2} e_2^T \dot{D}(q_1) e_2 = \\ &= e_2^T (D(q_1) \dot{q}_{d2} + B(q_1, q_2) q_{d2} + G(q_1, q_2) - U) + \frac{1}{2} e_2^T (\dot{D}(q_1) - B(q_1, q_2)) e_2. \end{aligned} \quad (12)$$

Известно, что матрица в последнем члене является кососимметричной. Следовательно, этот член равен нулю и в итоге получаем упрощенное выражение:

$$Q = e_2^T (D(q_1) \dot{q}_{d2} + B(q_1, q_2) q_{d2} + G(q_1, q_2) - U). \quad (13)$$

Таким образом, (13) представляет собой компактную форму, отображающую разницу между действительной и желаемой траекторией ПА, содержащую в себе параметры внешней среды и параметр ошибки траектории движения. Это выражение и является основой для построения НС регулятора.

**Структура нейронной сети.** При необходимой точности навигационных систем применяется комплексирование различных автономных датчиков ускорений и скорости. Управление ПР строится на рассогласовании между оценками и заданными значениями параметров движения в системе координат (СК), связанной с ПР с использованием пропорционально-интегрального – дифференциального (ПИД) регулятора. При прямолинейном движении данный подход позволяет синтезировать высококачественную систему автоматического управления движением (САУД).

Однако в случае сложных траекторий такой способ не подходит, ввиду того, что возникает необходимость разбиения сложной траектории, заданной в базовой СК, на прямолинейные участки. Каждый следующий переход требует перерасчета коэффициентов ПИД регулятора. Чем сложнее траектория, тем больше аппроксимирующих участков. Следовательно, необходимо чаще перестраивать параметры ПИД регулятора, а это занимает большой объем памяти бортового вычислительного устройства по сравнению с использованием нейронной сети.

Численные эксперименты с системой управления, содержащей разработанный контроллер НС, были проведены для различных параметров и выражений сил вязкости, крутящего момента и силы тяги. Выбор регулятора НС связан с подходом к априорной неопределенности, характеризующей ПР. Действительно, матрица инерции жесткого тела ПР неизвестна, так как это добавленное сопротивление трения поверхности аппарата с водной средой.

Силы и моменты вязкого трения имеют неизвестную функциональную структуру. Следовательно, ПР можно рассматривать как управляемый объект с частичными параметрами и структурными неопределенностями.

Было решено внедрить интеллектуальное управление ПР [1] на базе многослойной нейронной сети прямого распространения с алгоритмом обратного распространения ошибки. Данный вид НС успешно применяется для решения многих сложных задач. Также он прост в реализации и не требует больших объемов памяти и энергозатрат (рис. 10) [6].

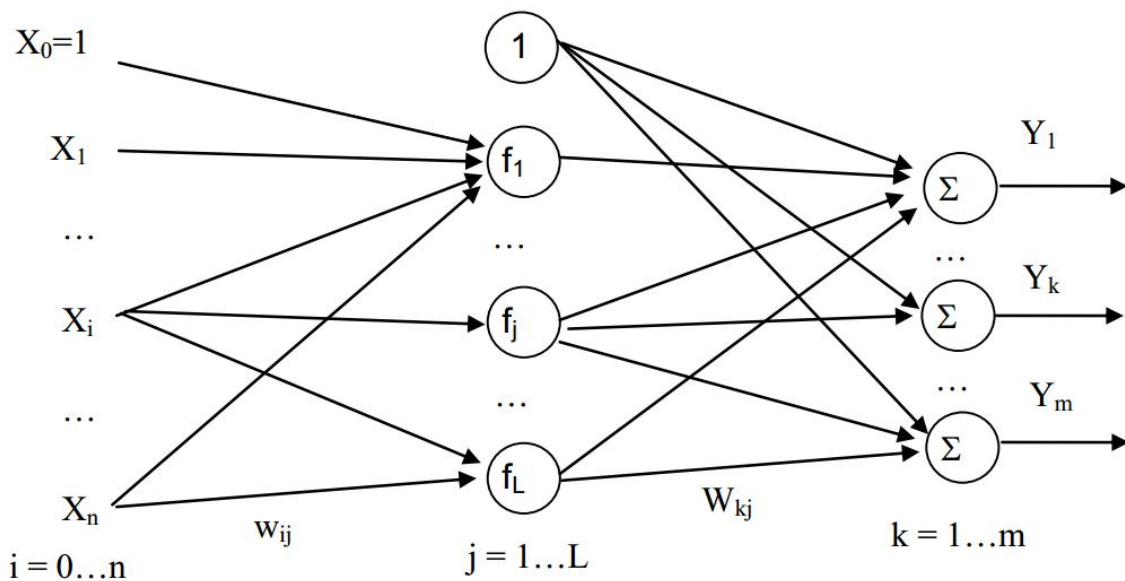


Рисунок 10 – Структура нейронной сети

Обучение алгоритмом обратного распространения ошибки предполагает два прохода по всем слоям сети: прямого и обратного. При прямом проходе входной вектор подается на входной слой нейронной сети, после чего распространяется по сети от слоя к слою. В результате генерируется набор выходных сигналов, который и является фактической реакцией сети на данный входной образ. Во время прямого прохода все синаптические веса сети фиксированы. Во время обратного прохода все синаптические веса настраиваются в соответствии с правилом коррекции ошибок, а именно: фактический выход сети вычитается из желаемого, в результате чего формируется сигнал ошибки. Этот сигнал впоследствии распространяется по сети в направлении, обратном направлению синаптических связей. Отсюда и название – алгоритм обратного распространения ошибки. Синаптические веса настраиваются с целью максимального приближения выходного сигнала сети к желаемому.

Пусть скрытые и выходные слои НС имеют  $L$  и  $m$  нейронов соответственно ( $m$  равно размеру  $e_2$ ). Для простоты предполагается, что только суммирование взвешенных сигналов (без нелинейного преобразования) реализуется в выходном слое. Входной вектор для НС имеет  $n$  координат.

Определяем  $w_{ij}$  как коэффициенты весов для  $i$ -го входа  $j$ -го нейрона скрытого слоя. Итак, эти коэффициенты составляют матрицу:

$$w = \begin{bmatrix} w_{11} & \dots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{L1} & \dots & w_{Ln} \end{bmatrix}. \quad (14)$$

В результате нелинейного преобразования  $f(\cdot)$  выходной вектор скрытого слоя можно записать в виде:

$$f(w, x) = \begin{bmatrix} f_1(w_1^T x) \\ \dots \\ f_L(w_L^T x) \end{bmatrix}, \quad (15)$$

где  $w_k$  обозначает  $k$ -ю строку матрицы  $w$ .

По аналогии вводим матрицу  $W$ , где элемент  $W_{li}$  обозначает весовой коэффициент из  $i$ -го нейрона скрытого слоя  $l$ -го нейрона выходного слоя.

При определенных параметрах НС сигнал управления ПР (выход НС) вычисляется следующим образом:

$$U = y(W, w, x) = Wf(w, x), \quad (16)$$

Запишем (16) с учетом (13):

$$Q = e_2^T (D(q_1)\dot{q}_{d2} + B(q_1, q_2)q_{d2} + G(q_1, q_2) - Wf(w, x)). \quad (17)$$

Чтобы получить алгоритм обучения НС, применим метод градиента скорости [7, 8, 10]. Для этого вычислим частные производные функции  $Q$  по времени регулируемого параметра НС – матрицы  $w$  и  $W$ .

Прямое дифференцирование дает

$$\frac{\partial Q}{\partial w} = -e_2^T f^T(w, x). \quad (18)$$

Легко показать, что выбор всех активационных функций в обычной форме:

$$f(x) = \frac{1}{(1 + e^{-\tau x})} \quad (19)$$

подразумевает собственность

$$\frac{\partial}{\partial w_{ij}} f_i(w_i^T x) = f_i(w_i^T x)[1 - f_i(w_i^T x)]x_j. \quad (20)$$

Введем дополнительные функции

$$\varphi_i(w_i^T x) = f_i(w_i^T x)[1 - f_i(w_i^T x)] \quad (21)$$

и матрицу

$$\Phi(w, x) = \text{diag}(\varphi_1(w_1^T x) \dots \varphi_L(w_L^T x)). \quad (22)$$

Прямой расчет дает:

$$\frac{\partial \dot{Q}}{\partial w} = -\Phi W^T e_2 x^T. \quad (23)$$

В качестве заключительного этапа запишем алгоритм обучения искусственной НС (ИНС) в следующей форме:

$$\begin{aligned} W^{(k+1)} &= W^{(k)} + \gamma e_2 f^T(w, x) \\ w^{(k+1)} &= w^{(k)} + \gamma e_2 \Phi W^T e_2 x^T \end{aligned} \quad (24)$$

где  $\gamma$  – шаг обучения;  $k$  – число итераций.

По такому алгоритму происходит обучение корректирующего положения регулятора в ПР. Исходя из вышесказанного, можно отметить простоту и компактность всех необходимых данных о такой сложной дифференциальной системе, описываемой нелинейными уравнениями, как ПР.

Теперь рассмотрим, какие компоненты должны быть включены в вектор ввода ИНС. Как контроллер, ИНС ориентирована на компенсацию влияния соответствующей матрицы и вектора функции. В общем случае входной вектор ИНС должен состоять из  $q_1, q_2, e_2, q_{d2}$  и их производной по времени. Процедура обучения ИНС приводит к уменьшению функции  $Q$ . Следовательно, идеальные условия – это когда ошибка  $e_2$  стремится к нулю, а движение ПР точно следует желаемой траектории.

$$q_2(t) \rightarrow q_{d2}(t). \quad (25)$$

Если траектория ПР задана  $q_{d1}(t)$ , тогда:

$$q_{d2}(t) = J^{-1}(q_1) \left( \dot{q}_{d1}(t) + k(q_{d1}(t) - q_1(t)) \right), \quad (26)$$

(где  $k$  – положительная постоянная). Как следует из уравнения кинематики,

$$\dot{q}_1(t) \rightarrow \dot{q}_{d1}(t) + k(q_{d1}(t) - q_1(t)) \quad (27)$$

и

$$\dot{e}_1(t) + k e_1(t) \rightarrow 0, \quad (28)$$

где

$$e_1(t) = q_{d1}(t) - q_1(t) \quad (29)$$

Следовательно, ПР следует по планируемой траектории  $q_{d1}(t)$ .

Конечный результат разработанной структуры НС также необходимо испытать в виртуальной среде с искусственно заданными входными параметрами.

**Моделирование НС в среде MATLAB.** Для моделирования работы НС используется программное приложение MATLAB и его вспомогательный шаблон nntool – для удобной и быстрой настройки нейронной сети.

С целью проверки эффективности работы НС были взяты следующие параметры ПР: функция обучения ИНС, алгоритм обучения ИНС, а также рассчитанные параметры ПР –  $D_{RB}$  матрица инерции ПР, состоящая из проекции импульса на систему координат,  $D_A$  матрица присоединенных масс, характеризующая инерционность конструкции ПР и демпфирующая сила «В», воздействующая на ПР со стороны среды. Значения матриц были взяты по расчетам из [5]. Результаты соответственно равны:  $D_{RB} = [1000 \ 0 \ 200; 0 \ 1000 \ 0; 200 \ 0 \ 11000]$ ;  $D_A = [1000 \ 0 \ 100; 0 \ 1100 \ 80; 100 \ 80 \ 9000]$ . Силу гравитации для упрощения подсчета примем равной нулю, так как аппарат (ПР) имеет нулевую плавучесть. Однако необходимо учитывать демпфирующую силу в воде  $B = [210 \ 20 \ 30; 25 \ 200 \ 70; 15 \ 33 \ 1500]$ .

Созданная сеть network1 также имеет окно настройки (рис. 2). На нем подробно указывается следующее. Тип НС – нейронная сеть прямого распространения (feed-forward backprop), в которой соединения между узлами не образуют цикл.

Входные и выходные значения НС. Функция обучения TRAINLM – это функция сетевого обучения, которая обновляет значения веса и смещения в соответствии с оптимизацией по алгоритму Левенберга – Марквардта. Также к функции обучения мы можем подобрать алгоритм вычисления изменения веса для данного нейрона от входа и ошибки нейрона. По умолчанию стоит LEARNINGDM – метод градиентного спуска для массы импульса и функции обучения смещению. MSE – среднеквадратичная нормированная ошибка – это функция производительности сети. В пунктах number of layers и number of neurons

соответственно указываем количество слоев и нейронов, а в transfer function – нейронную передаточную функцию TANSIG (Tan-Sigmoid Transfer Function).

Входом нейронной сети является вектор линейной скорости по «x» и «z», а также угловой скорости по «y», взятые по среднему значению скорости ПП:  $q_1 = [V_x \ V_z \ \omega_y]^T$ , где в промежутке 0...2,5 секунды  $V_x = 0,75$  м/с,  $V_z = 0,5$  м/с,  $\omega_y = -0,15$  рад/с. Последующие 2,5 секунды –  $V_x = 0,5$  м/с,  $V_z = 0,75$  м/с,  $\omega_y = 0.15$  рад/с. Эти значения представляют собой ситуацию двух резких поворотов ПП в плоскости горизонта и увеличивают тем самым ошибку НС.

На рисунке 11 представлено главное окно настройки параметров НС в шаблоне nntool.

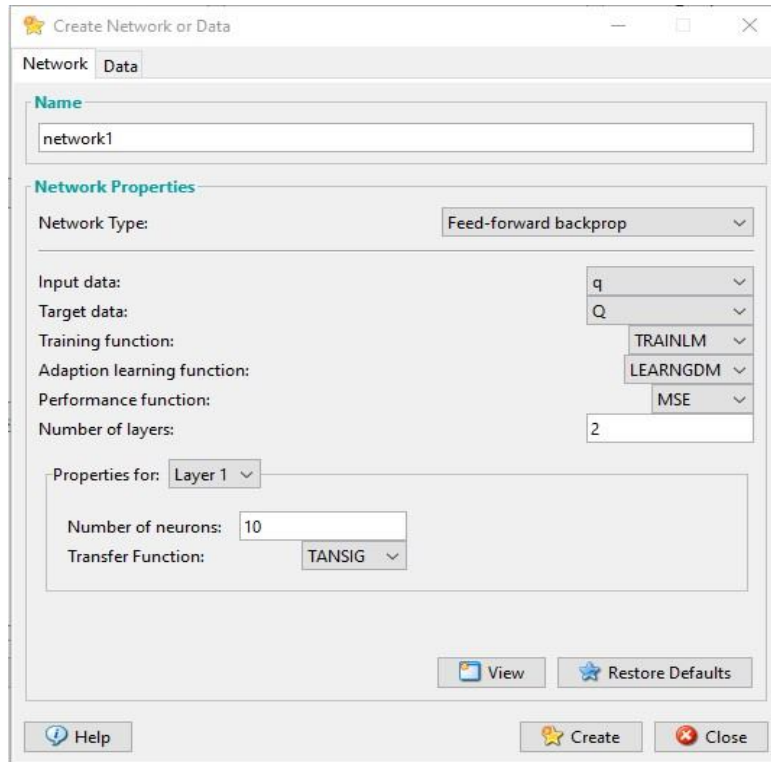


Рисунок 11 – Окно параметров настройки сети network1

В модели работы НС было использовано 10 нейронов в скрытом слое (hidden layer) – для цели регулирования положения ПП этого достаточно. Общая структурная схема НС изображена на рисунке 12.

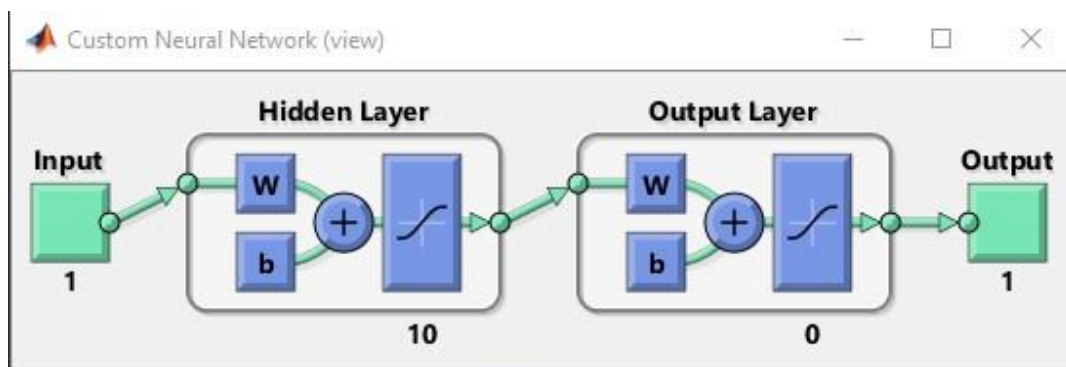


Рисунок 12 – Структура НС

Скрытый слой НС состоит из функции уменьшения ошибки действительной скорости ПП  $q_2$  по отношению к заданной скорости в виде вектора  $q_1$ . На рисунке 7 представлен расчет функции  $Q$  – разницы между желаемой и действительной траекториями ПП (по скорости перемещения ПП, рассчитанной на основе информации, приходящей от акселерометра и гироскопа). Расчет функции уменьшения ошибки (рис. 13) был подготовлен отдельно в MATLAB и в дальнейшем экспортирован в шаблон НС nntool в качестве модели обучения нейронной сети (Target data) – рисунок 14.

```

sim.m* x simou_ard.m x arduino_trans_fanc.m x +
1 - B=[210 20 30; 25 200 70; 15 33 1500];%демпферная сила
2 - G=[0; 0; 0];%сила гравитации
3 - Drb=[1000 0 200;0 1000 0; 200 0 11000];%инерция
4 - Da=[0.02826 0 100; 0 1100 80; 100 80 9000];%присоединенные массы
5 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
6 function[] B(q1,q2);% демпферная сила
7 function[] Da(q1);
8 function[] Drb(q1);
9 function[] Wf(w,x);
10 function[] q1(t);
11 function[] q2(t);
12 function[] e(t);
13 - e(t) = qd1(t)-q1(t);
14 - Q = ((e2)')*(D(q1)*q2+B(q1,q2)*q2+Wf(w,x))
15 - data = fread(kk, byte, 'float32');
    
```

Рисунок 14 – Расчет ошибки траектории ПР

На рисунке 15, представленном ниже, показана функция nntool по импорту данных из среды MATLAB в структуру НС. В данном случае мы импортируем расчет ошибки ПР (вектор-строка, обозначенная переменной kk) в скрытый слой НС.

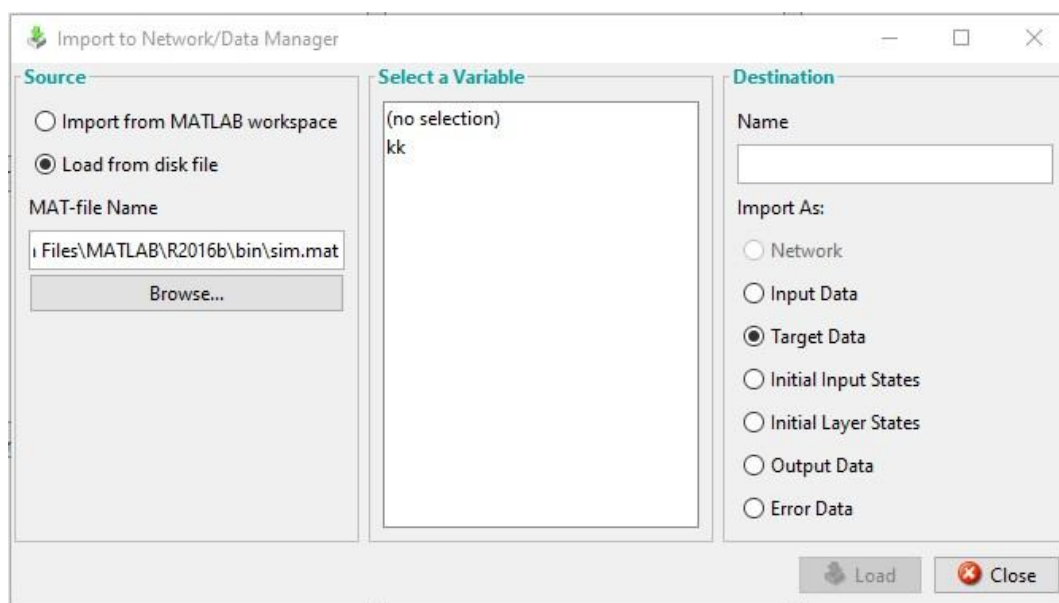


Рисунок 15 – Окно импорта из среды Matlab в НС

По импортируемой функции расчета ошибки мы уменьшаем разницу между действительной и реальной траекториями ПР – эта операция происходит каждый раз при прохождении сигнала управления по слоям НС. Таким образом, НС регулятор способен оценить ситуацию и принять меры по удержанию правильного курса аппарата.

**Полученные результаты и их обсуждение.** Выходом НС является вектор действительной скорости ПР  $q_2$  в период времени от 0 до 2,5 секунд и  $q_3$  – в период от 2,5 до 5 секунд. На рисунке 16 представлены графики скорости моделируемого ПР с использованием и без использования обученной НС.

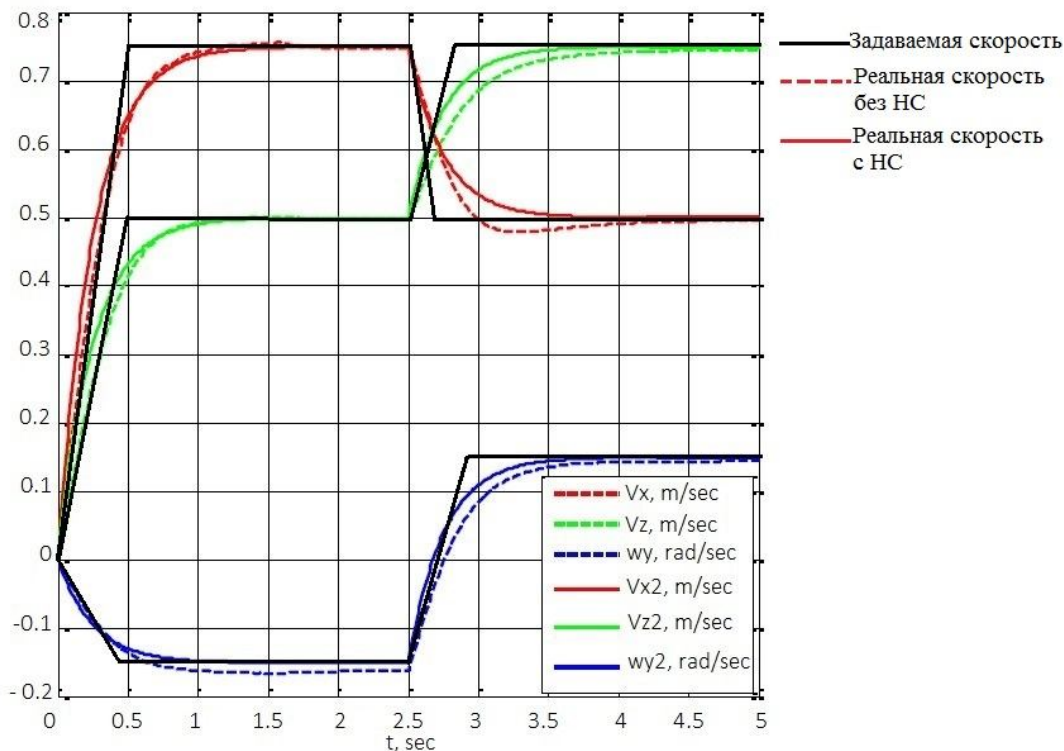


Рисунок 16 – Выходные характеристики с использованием и без использования обученной НС

Рисунок 16 представляет собой наглядный процесс управления РР. На нем черная линия является эталонной моделью изменения скорости РР, пунктирная линия – это реальная скорость РР без использования НС и сплошная линия – реальная скорость РР с использованием НС. На рисунке 16 можно заметить, что скорость без НС имеет либо перерегулирование или же увеличенное время переходного процесса.

Последующая задача после моделирования регулятора на базе НС – это его перенос на РР. В данной проблеме нам поможет одноплатный компьютер Raspberry Pi, имеющий цифровые и аналоговые входы. Поэтому не составит труда подключить наш контроллер Atmega 328P к Raspberry Pi и считать вектор данных о состоянии РР (координаты, угол, скорости). По итогам обработки контроллер принимает обратные корректирующие данные. Преимущество использования Raspberry Pi заключается в том, что мы можем установить пакет MATLAB с моделью, представленной выше, и использовать эту модель для корректировки положения РР.

С целью сравнения работоспособности НС регулятора был также смоделирован типовой ПИД-регулятор. Такой регулятор – типовое устройство в управляющем контуре с обратной связью. Он используется в системах автоматического управления для формирования управляющего сигнала с целью получения необходимой точности и качества переходного процесса. ПИД-регулятор формирует управляющий сигнал, являющийся суммой трёх слагаемых. Первое из них пропорционально разности входного сигнала и сигнала обратной связи (сигнал рассогласования), второе – интеграл от сигнала рассогласования, третье – производная сигнала рассогласования. Структурная схема САР, взятая нами за основу, рассмотрена в работе [2] и показана на рисунке 17.

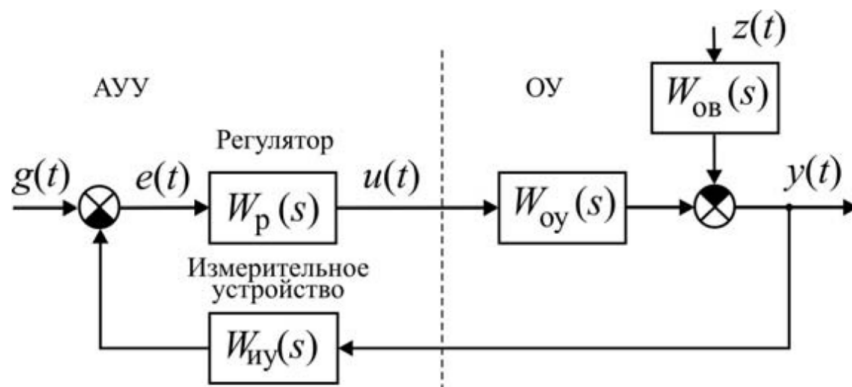


Рисунок 17 – Операторно-структурная схема САР

Одномерный объект управления с одним управляющим  $u(t)$  и одним возмущающим  $z(t)$  воздействием описывается уравнением:

$$y(t) = W_{oy}(p)u(t) - W_{об}(p)z(t), \tag{30}$$

где  $y(t)$  – регулируемая величина;  $W_{oy}(p)$  – оператор, совпадающий с передаточной функцией  $W_{oy}(s)$  объекта по управляющему воздействию;  $W_{об}(p)$  – оператор, совпадающий с передаточной функцией  $W_{об}(s)$  объекта по возмущающему воздействию.

Соответственно для внедрения ПИД-регулятора в систему РНПА необходимо знать достоверную передаточную функцию системы или подсистем – если это децентрализованный способ управления. Помимо этого необходимо математическое описание измерительных цифровых устройств (компас, гироскоп, акселерометр), которые являются дискретным устройством. При этом входные и выходные сигналы изменяются в дискретные моменты времени, и для отображения информации необходимо использовать дискретное время, что добавляет значительные трудности в практическом применении ПИД-регулятора. Исходя из вышесказанного, ПИД-регуляторы в практическом применении используются без математической модели измерительных устройств. Это снижает их адаптивную способность по отношению к быстро изменяющимся условиям внешней среды. На рисунке 18 представлена разработанная модель ПИД-регулятора.

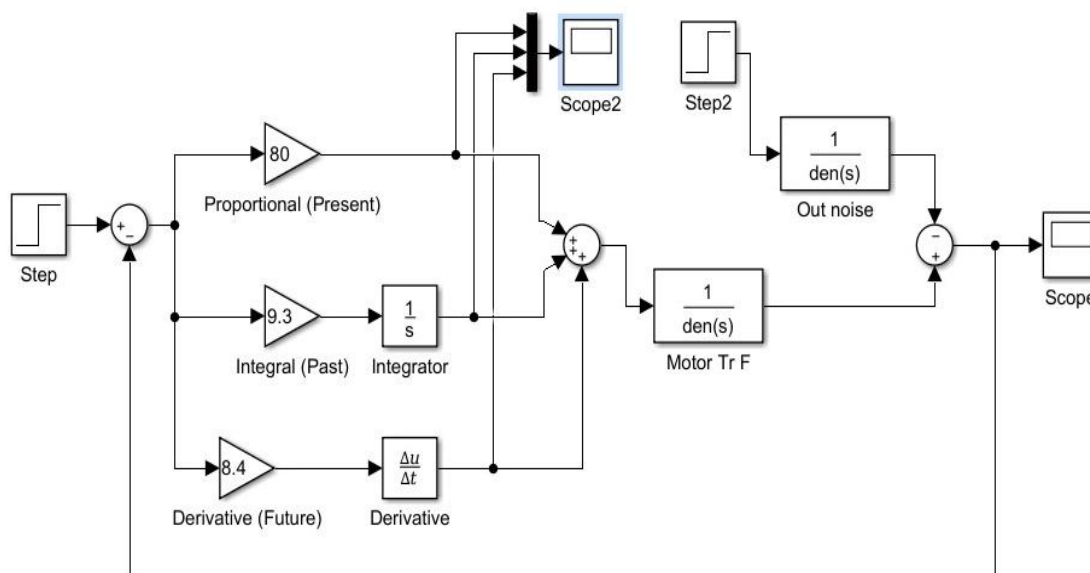


Рисунок 18 – Модель ПИД-регулятора

Коэффициенты ПИД-регулятора настраивались по методу Зиглера – Никольса [12] и соответственно являются константами. Для РНПА это является недопустимым, так как водная среда изменчива (наличие изменчивого течения, температурного клина и пр.).

На рисунке 19 представлен переходной процесс настройки коэффициентов ПИД-регулятора.

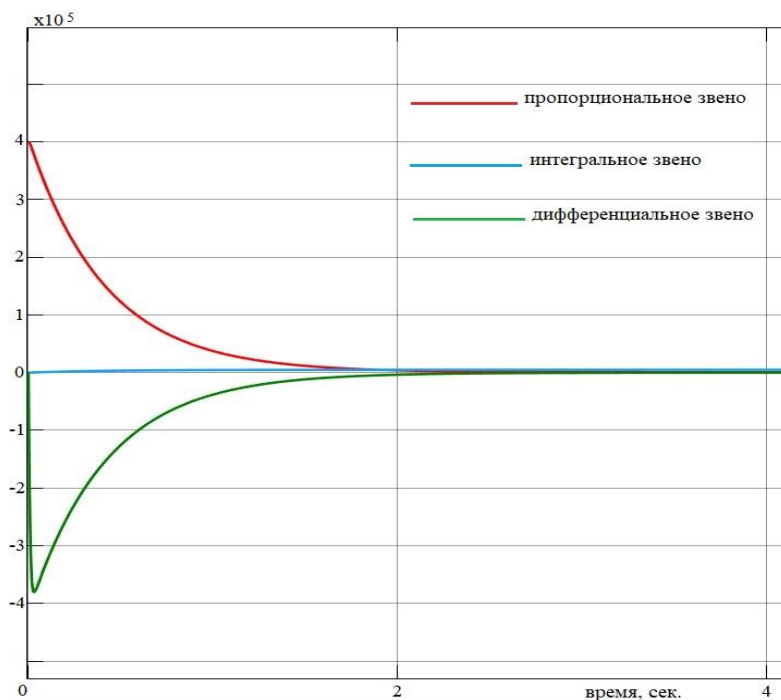


Рисунок 19 – ПИД-регулятор, переходные характеристики

По результатам рассмотрения модели ПИД-регулятора и НС можно сделать следующие выводы:

- время переходного процесса настройки в ПИД-регуляторе больше, чем в НС;
- для практического применения ПИД-регулятора необходимо оперировать множеством данных об управляемой системе, тогда как НС базируется на нечеткой логике;
- необходимо математическое описание измерительных устройств, тогда как на вход НС можно подавать прямые показания с датчиков в виде матриц;
- отсутствие автономной настройки коэффициентов ПИД-регулятора.

В результате проведенных исследований можно сделать следующие **ВЫВОДЫ**:

- на основании данных о характеристиках подводного робота, симуляции работы гребного винта и лабораторных испытаниях робота была рассчитана передаточная функция двигателя в условиях работы под водой и построена математическая модель движения подводного робота;
- для управления роботом была определена структура многослойной нейронной сети и произведено ее моделирование в среде Matlab;
- в результате моделирования были получены выходные характеристики по скорости перемещения, которые сравнивались с результатами, полученными при использовании ПИД-регуляторов;
- на основании сравнительных данных можно говорить о преимуществах нейросетевого управления подводным роботом в изменяющихся условиях среды.

#### Библиографический список

1. Домидов М. Н. Покорение глубин / М. Н. Домидов, А. Н. Дмитриев // Судостроение. – 1969. – С. 24–30.
2. Денисенко В. В. ПИД-регуляторы: принципы построения и модификации / В. В. Денисенко // СТА. – 2006. – № 4. – С. 66–74.
3. Рыбаков, А. В. Робот-гексапод с нейросетевой системой управления движением / А. В. Рыбаков, Т. Х. Куаншкалиев // Робототехника и техническая кибернетика. – 2017. – № 3 (16). – С. 59–66.
4. Фрадков А. Л. Адаптивное управление сложными системами / А. Л. Фрадков. – М.: Наука, 1990. – С. 45–116.
5. Miroshnik I. V. Nonlinear and Adaptive Control of Complex Dynamical Systems / I. V. Miroshnik. – St. Petersburg: Nauka, 2000. – 549 p. (Series «Analysis and Design of Nonlinear Systems»).
6. Narendra K. S. Identification and control of dynamical systems using neural networks / K. S., Narendra, K. Parthasarathy // IEEE Identification and Control of Dynamical System. – 1990. – Vol. 1, № 1, p. 1475-1483.
7. Ross A. Identification of underwater vehicle hydrodynamic coefficients using free decay tests / A. Ross, T. Fossen // Preprints of Int. Conf. CAMS-2004. – Ancona, Italy, 2004. – P. 363–368.
8. Sutton R. An on-line intelligent multi-input multi-output autopilot design study / R. Sutton // Journal of Engineering for the Maritime Environment. – 2002. – Vol. 216, № M2. – P. 117–131.
9. Fossen T. I. Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles. Marine Cybernetics AS / T. I. Fossen. – Trodheim, Norway, 2002. – P. 245–270.



10. Yuh Y. Modelling and control of underwater vehicles / Y. Yuh // *IEEE J. of Trans. Syst., Man., Cybern.* – 1990. – Vol. 12.
11. Dyda A. A. Neural network control system for underwater robots / A. A. Dyda // *IFAC Proceedings.* – 2004. – № 37 (10). – P. 427–432.
12. Режим доступа: [http://opiobjektid.tptlive.ee/Automatiseerimine/134\\_\\_\\_\\_\\_html](http://opiobjektid.tptlive.ee/Automatiseerimine/134_____html), свободный. – Заглавие с экрана. – Яз. рус.

#### References

1. Domidov M. N., Dmitriev A. N. Pokorenie glubin [Conquest of the depths]. *Sudostroenie* [Shipbuilding], 1969, pp. 24–30.
2. Denisenko V. V. PID-regulatory: principy postroeniya i modifikatsii [PID-regulators: principles of construction and modification]. *STA* [STA], 2006, no. 4, pp. 66–74.
3. Rybakov A. V., Kuanshkaliev T. H. Robot-geksapod s neyrosetevoy sistemoy upravleniya dvizheniem [Robot hexapod with a neural network motion control system]. *Robototekhnika i tekhnicheskaya kibernetika* [Robotics and Technical cybernetics], 2017, no. 3 (16), pp. 59–66.
4. Fradkov A. L. *Adaptivnoe upravlenie slozhnymi sistemami* [Adaptive control of complex system]. M., Nauka Publ., 1990.
5. Miroshnik I. V. *Nonlinear and Adaptive Control of Complex Dynamical Systems*. St. Petersburg, Nauka, 2000. 549 p. (Series «Analysis and Design of Nonlinear Systems»).
6. Narendra K. S., K. Parthasaraty Identification and control of dynamical systems using neural networks. *IEEE Identification and Control of Dynamical System*, 1990, vol. 1, no. 1, pp. 1475–1483.
7. Ross A., Fossen T. Identification of underwater vehicle hydrodynamic coefficients using free decay tests. *Preprints of Int. Conf. CAMS-2004*, Ancona, Italy, 2004, pp. 363–368.
8. Sutton R. An on-line intelligent multi-input multi-output autopilot design study. *Journal of Engineering for the Maritime Environment*, 2002, vol. 216, no. M2, pp. 117–131.
9. Fossen T. I. *Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles*. Marine Cybernetics AS. Trondheim, Norway, 2002, pp. 245–270.
10. Yuh Y. Modelling and control of underwater vehicles. *IEEE J. of Trans. Syst., Man. Cybern.*, 1990, vol. 12.
11. Dyda A. A. Neural network control system for underwater robots. *IFAC Proceedings*, 2004, no. 37 (10), pp. 427–432.
12. Available at: [http://opiobjektid.tptlive.ee/Automatiseerimine/134\\_\\_\\_\\_\\_html](http://opiobjektid.tptlive.ee/Automatiseerimine/134_____html).