

## Событийный подход к многорежимному управлению группировкой подводных роботов в обследовательской миссии

И. В. Бычков, А. В. Давыдов, Н. В. Нагул, С. А. Ульянов\*

Институт динамики систем и теории управления им. В.М. Матросова СО РАН,  
Иркутск, Россия

\*Контактный e-mail: sau@icc.ru

Предложен подход к построению двухуровневой системы управления группировкой автономных подводных роботов (АПР) при обследовании придонной области. Для реализации элементарных поведений (режимов функционирования) группы на нижнем уровне предложены алгоритм отслеживания траектории, основанный на концепции виртуальной цели, и децентрализованный алгоритм управления формациями, базирующийся на схеме лидер — ведомый. Синтез регуляторов, обеспечивающих требуемое поведение системы в различных режимах, выполнен с использованием векторных функций Ляпунова с учетом дискретности по времени и погрешностей измерений, неопределенности параметров модели АПР, а также ограничений на ресурсы управления. Для построения верхнего уровня управления, ответственного за переключение между режимами, используется теория супервизорного управления дискретно-событийными системами. Приведены результаты численных расчетов, демонстрирующих работоспособность предложенного подхода.

*Ключевые слова:* подводный аппарат, управление формацией, векторная функция Ляпунова, дискретно-событийная система.

### Введение

В настоящее время автономные подводные роботы (АПР) становятся основным средством для выполнения различных подводных работ, включая мониторинг подводной среды, картографирование рельефа дна, обследовательские миссии и др. Применение скоординированных групп АПР может значительно сократить длительность выполнения подводных работ, а также повысить функциональную надежность и робастность системы по отношению к непредвиденным событиям. Вместе с тем поведение группы по сравнению с одиночным АПР гораздо более сложное, оно может включать значительное число элементарных моделей поведения или режимов функционирования.

Традиционными в подводной робототехнике являются миссии по обследованию (сканированию) подводной области, при которых группа перемещается на заданной глубине по сгенерированным специальным образом траекториям, покрывающим исследуемую область, удерживая заданную геометрическую конфигурацию (формацию) во время

рабочих ходов и избегая столкновений с препятствиями. Для таких миссий характерны следующие режимы: сканирование, сбор формации, обход препятствий. Первый режим предполагает движение в отсутствие препятствий с как можно более точным сохранением строя. Режим сбора формации включается, когда роботы находятся на значительном расстоянии друг от друга и требуется привести их в положения, допустимые режимом точного сканирования. Такие ситуации могут возникать в начале выполнения обследовательской миссии, а также после завершения маневров по обходу препятствий или выполнения программных разворотов по достижении границы исследуемой области, которые сопровождаются изменением структуры формации. Активация режима обхода препятствий происходит при обнаружении их по ходу движения. Первые два режима определяют коллективное поведение группы и иницируются лидером группировки, в то время как режим обхода препятствий включается независимо каждым АПР группы.

При реализации элементарных поведений АПР в обследовательской миссии решаются три основные задачи: генерации траектории движения робота [1], включая ее коррекцию в режиме реального времени [2]; отслеживания траектории [3, 4]; управления формацией [5, 6]. В данной работе на основе нелинейной модели АПР разработан алгоритм отслеживания траектории, основанный на концепции виртуальной цели [3], и децентрализованный алгоритм управления формацией, использующий схему лидер — ведомый [7, 8]. Синтез регуляторов выполнен с использованием вычислительной технологии для анализа и синтеза нелинейных систем управления, основанной на идеях редукции [9, 10] и сублинейных векторных функций Ляпунова [11]. Эта технология в отличие от большинства техник, используемых в упомянутых выше работах, позволяет учитывать дискретный по времени характер и погрешности измерений, неопределенности параметров модели АПР, ограничения на ресурсы управления. Предложенные алгоритмы используются для формирования элементарных поведений группы и составляют основу нижнего уровня гибридной системы управления.

В то время как непрерывная динамика АПР-лидера и ведомых определяет режимы их движения, переключение между различными режимами функционирования может описываться в терминах дискретно-событийной модели. В широко используемых дискретно-событийных системах (ДСС) эволюция состояний рассматривается как следствие возникновения последовательностей некоторых событий. Теория супервизорного управления (ТСУ), разработанная в 1980-х гг. для управления поведением ДСС, сегодня становится мощным инструментом во многих реальных приложениях, включая робототехнику. Последние применения в этой области включают управление одиночными роботами [12, 13], группами роботов [14, 15] и их формациями [16], роями роботов [17, 18] и др. Однако в большинстве перечисленных работ используются простейшие супервизоры, построенные на основе автоматной модели управляемой системы путем исключения из нее нежелательных переходов.

Популярный способ проектирования супервизорного управления состоит в построении автоматов для активации и деактивации элементарных поведений, построении параллельной композиции этих автоматов и анализе состояний результирующего автомата, которые представляют все возможные комбинации элементарных поведений. Переходы, приводящие к нежелательным сочетаниям, должны быть запрещены супервизором. Будучи достаточно эффективным, этот подход лишь частично использует результаты ТСУ, затрагивающие в первую очередь формальные языки, описывающие поведение системы и ограничения на него. Важные свойства ДСС, такие как управ-

ляемость языка спецификации и неблокируемость супервизора, обычно не обсуждаются, а сами спецификации зачастую не определены явно. Мы предлагаем дискретно-событийную модель переключения режимов работы АПР-лидера как реакцию на изменения среды, предыдущие и текущие режимы и строим супервизор, обеспечивающий выполнение языковых спецификаций на движение формации АПР.

В статье синтезированы регуляторы, составляющие основу нижнего уровня управления, приведены алгоритмы верхнего уровня управления, основанные на ДСС, и представлены результаты численного моделирования разработанной гибридной двухуровневой системы управления.

## 1. Алгоритмы управления нижнего уровня

### 1.1. Модель АПР

Для построения алгоритмов управления нижнего уровня применим заимствованную из [3] модель АПР, описывающую его движение на плоскости с использованием глобальной  $\{U\}$  и связанной  $\{B\}$  систем координат (рис. 1):

$$\begin{aligned} \dot{x} &= u \cos(\psi_B) - v \sin(\psi_B), & \dot{y} &= u \sin(\psi_B) + v \cos(\psi_B), & \dot{\psi}_B &= r, \\ \mathcal{F} &= m_u \dot{u} + d_u, & 0 &= m_v \dot{v} + m_{ur} ur + d_v, & \mathcal{G} &= m_r \dot{r} + d_r, \end{aligned} \quad (1)$$

где  $x, y$  — координаты центра массы АПР;  $\psi_B$  — угол рысканья;  $u$  и  $v$  — поперечная и продольная скорости аппарата соответственно;  $r$  — скорость рысканья;

$$\begin{aligned} m_u &= m - X_{\dot{u}}, & d_u &= -X_{uu}u^2 - X_{vv}v^2, \\ m_v &= m - Y_{\dot{v}}, & d_v &= -Y_v uv - Y_{v|v}|v|v|, \\ m_r &= I_z - N_{\dot{r}}, & d_r &= N_v uv - N_{v|v}|v| - N_r ur, \\ m_{ur} &= m - Y_r, \end{aligned}$$

$m$  — масса АПР;  $I_z$  — момент инерции относительно вертикальной оси аппарата;  $X_{\{\cdot\}}, Y_{\{\cdot\}}, N_{\{\cdot\}}$  — классические производные гидродинамических коэффициентов;  $\mathcal{F}, \mathcal{G}$  — управляющие сила и момент.

### 1.2. Отслеживание траектории

Решение задачи отслеживания траектории базируется на концепции виртуальной цели. Определим цель как материальную точку  $P$ , движущуюся вдоль отслеживаемого пути. Пусть с точкой  $P$  связана подвижная система координат Френе—Серре  $\{F\}$  (рис. 1), одна ось которой совпадает с направлением вектора скорости цели, а вторая — перпендикулярна первой. Тогда, как показано в [3], движение АПР в подвижной системе координат  $\{F\}$  описывается уравнениями

$$\dot{s}_1 = v_t \cos \psi - \dot{s}_a + \dot{\psi}_F y_1, \quad \dot{y}_1 = v_t \sin \psi - \dot{\psi}_F s_1, \quad \dot{\psi} = r + \dot{\beta} - \dot{\psi}_F, \quad (2)$$

где  $s_1, y_1$  — координаты АПР в подвижной системе координат  $\{F\}$ ;  $s_a$  — криволинейная координата точки  $P$ , определяющая ее положение на кривой;  $\beta = \arctan(v/u)$  — угол скольжения;  $v_t = (u^2 + v^2)^{1/2}$  — модуль вектора скорости АПР;  $\psi_F$  — угол ориентации подвижной системы координат  $\{F\}$  относительно глобальной  $\{U\}$  ( $\psi_F = c_c(s_a)\dot{s}_a$ ),

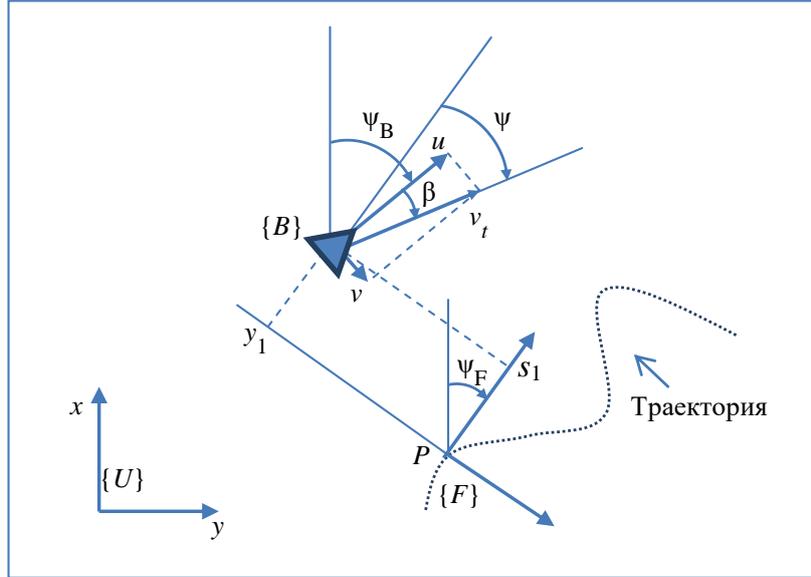


Рис. 1. Системы координат

$c_c$  — кривизна пути,  $\psi \triangleq \psi_B + \beta - \psi_F$ . Будем предполагать, что виртуальная цель движется вдоль траектории с постоянной скоростью  $u_d$  и имеется ограничение на кривизну пути  $|c_c| \leq \bar{c}_c$ .

Задача отслеживания траектории формулируется следующим образом: определить законы управления для силы  $\mathcal{F}$  и момента  $\mathcal{G}$ , обеспечивающие асимптотическое стремление к нулю переменных  $y_1$ ,  $s_1$  и  $\psi$  при  $t \rightarrow \infty$ .

Для решения поставленной задачи предложен цифровой закон управления

$$\begin{aligned} \mathcal{F}(t) &= \mathcal{F}_c + \mathcal{F}_s, \quad \mathcal{G}(t) = \mathcal{G}_c + \mathcal{G}_s, \quad t \in T_k \equiv [t_k, t_{k+1}), \\ \mathcal{F}_c &= d_u, \quad \mathcal{G}_c = d_r + m_r(\dot{c}_c \dot{s}_a + c_c \ddot{s}_a - \hat{\hat{\beta}}), \\ \mathcal{F}_s &= \text{sat}(k_1 \hat{s}_{1k} + k_2 \Delta \hat{u}_k, \bar{\mathcal{F}}_s), \quad \mathcal{G}_s = \text{sat}(k_3 \hat{y}_{1k} + k_4 \hat{\psi}_k + k_5 \Delta \hat{r}_k, \bar{\mathcal{G}}_s), \end{aligned} \quad (3)$$

где  $t_k = kh$ ,  $k = 0, 1, 2, \dots$ ,  $h$  — шаг управления;  $\mathcal{F}_c$ ,  $\mathcal{G}_c$  — слагаемые полного управления, призванные компенсировать возмущения  $d_u$ ,  $d_r$  в уравнениях соответственно для  $u$  и  $r$  системы (1), а также слагаемые  $\hat{\hat{\beta}}$  и  $\hat{\psi}_F \triangleq c_c(s_a) \dot{s}_a$ , входящие в правую часть последнего уравнения системы (2);  $\hat{\hat{\beta}}$  — оценка ускорения  $\ddot{\beta}$ , вычисляемая на основе динамической модели АПР (подробности см. в [3]);  $\mathcal{F}_s$ ,  $\mathcal{G}_s$  — стабилизирующие слагаемые управления, обеспечивающие решение задачи слежения,  $\bar{\mathcal{F}}_s$ ,  $\bar{\mathcal{G}}_s$  — ресурсы управления по силе и моменту, выделенные на решение основной задачи;  $\hat{s}_{1k}$ ,  $\hat{y}_{1k}$ ,  $\hat{\psi}_k$ ,  $\Delta \hat{u}_k$ ,  $\Delta \hat{r}_k$  — оценки переменных  $s_1$ ,  $y_1$ ,  $\psi$ ,  $\Delta u \triangleq u - u_d$ ,  $\Delta r \triangleq r + \dot{\beta} - \dot{\psi}_F$ , вычисляемые в дискретные моменты времени  $t_k$ ;  $\text{sat}(\sigma, \bar{\sigma}) = \text{sign}(\sigma) \min(|\sigma|, \bar{\sigma})$  — функции типа “насыщение”;  $k_i$  — коэффициенты обратных связей, подлежащие отысканию ( $i = \overline{1, 5}$ ).

### 1.3. Управление формацией

Используемый в режимах сканирования и сбора группы алгоритм управления формацией основан на схеме лидер — ведомый, при которой задача ведомых роботов состоит в удерживании желаемого положения относительно своих лидеров. Будем предпола-

гать, что каждый ведомый имеет ровно одного лидера, т.е. граф связи имеет структуру дерева, и существует АПР, называемый лидером формации, который не является ведомым ни для какого другого АПР и задает движение группы в целом.

Пусть АПР группы оснащены датчиками, способными измерять дистанцию до лидера  $s = \sqrt{(x_l - x_f)^2 + (y_l - y_f)^2}$  и угол пеленга  $\theta = \psi_{Wf} - \arctan \frac{y_f - y_l}{x_f - x_l}$  ( $\psi_W \triangleq \psi_B + \beta$ ). Здесь нижний индекс  $l$  ( $f$ ) указывает на принадлежность соответствующих переменных лидеру (ведомому). Динамика пары лидер — ведомый в терминах этих переменных описывается следующими уравнениями:

$$\begin{aligned} \dot{s} &= v_{ul} \cos(\psi_{Wl} - \psi_{Wf} + \theta) - v_{tf} \cos \theta, \\ s\dot{\theta} &= sr_f + v_{tf} \sin \theta - v_{ul} \sin(\psi_{Wl} - \psi_{Wf} + \theta). \end{aligned}$$

Пусть  $s = s^*$  и  $\theta = \theta^*$  соответствуют желаемому положению ведомого АПР относительно своего лидера. Для стабилизации этого положения используется цифровой закон управления

$$\begin{aligned} \mathcal{F}(t) &= \mathcal{F}_s + d_u, \quad \mathcal{G}(t) = \mathcal{G}_s + d_r, \quad t \in T_k, \\ \mathcal{F}_s &= \text{sat}(k_1 \Delta \hat{s}_k + k_2 z_k, \bar{\mathcal{F}}_s), \quad \mathcal{G}_s = \text{sat}(k_3 \Delta \hat{\theta}_k + k_4 z_k + k_5 \Delta \hat{\theta}_k + k_6 z_k^\theta, \bar{\mathcal{G}}_s), \end{aligned} \quad (4)$$

где оценки  $\Delta \hat{s}_k$  и  $\Delta \hat{\theta}_k$  отклонений  $\Delta s = s - s^*$  и  $\Delta \theta = \theta - \theta^*$  (по дистанции и углу соответственно) вычисляются в моменты времени  $t_k$  с использованием дискретных фильтров

$$\Delta \hat{s}_k = \sum_{\nu=1}^{\mu} \lambda_{\nu} (\bar{s}_{k-1,\nu} - s^*), \quad \Delta \hat{\theta}_k = \sum_{\nu=1}^{\mu} \lambda_{\nu}^{\theta} (\bar{\theta}_{k-1,\nu} - \theta^*).$$

Здесь  $\bar{s}_{k-1,\nu} \triangleq s(t_{k-1,\nu}) + \tilde{s}(t_{k-1,\nu})$  и  $\bar{\theta}_{k-1,\nu} \triangleq \theta(t_{k-1,\nu}) + \tilde{\theta}(t_{k-1,\nu})$  — измерения дистанции и угла пеленга в моменты  $t_{k-1,\nu} = t_{k-1} + \tau_{\nu}$ ,  $0 \leq \tau_{\nu} \leq h$ ,  $\nu = \overline{1, \mu}$  ( $\mu$  — глубина памяти);  $\tilde{s}$  и  $\tilde{\theta}$  — ограниченные ошибки измерений;  $\lambda_{\nu}$ ,  $\lambda_{\nu}^{\theta}$  — параметры фильтров. Для улучшения качества управления в обратной связи также используются оценки  $z_k$ ,  $z_k^{\phi}$  скоростей  $\Delta \dot{s}$ ,  $\Delta \dot{\theta}$ , вычисляемые с помощью дискретных наблюдателей

$$\begin{aligned} z_{k+1} &= \sum_{\nu=1}^{\mu} a_{\nu} (\bar{s}_{k,\nu} - s^*) + b z_k, \quad z_0 \equiv z(t_0) = 0, \\ z_{k+1}^{\theta} &= \sum_{\nu=1}^{\mu} a_{\nu}^{\theta} (\bar{\theta}_{k,\nu} - \theta^*) + b^{\theta} z_k^{\theta}, \quad z_0^{\theta} \equiv z_0^{\theta}(t_0) = 0. \end{aligned}$$

#### 1.4. Обход препятствий

Будем полагать, что АПР на основе данных измерений, снимаемых с акустических дальномеров, может генерировать траектории, обеспечивающие безопасный обход препятствий, обнаруживаемых в процессе сканирования. В данной работе задача онлайн-генерации траектории движения АПР в неструктурированной среде не рассматривается. Некоторые возможные подходы к ее решению представлены в [2, 19]. При сделанном выше предположении задача обхода препятствий для лидера группировки решается с использованием алгоритма отслеживания траектории (3). Лидер активирует режим обхода в случае приближения к ближайшему из препятствий на заданное расстояние,

а по завершении обходных маневров и возвращении на текущий галс переключает его на режим сбора формации, о чем оповещает другие аппараты группы. При этом с целью уменьшения затрачиваемой энергии возможны изменения структуры формации, ответственным за которые является лидер группы.

Ведомые АПР в отсутствие препятствий движутся заданным строем, который обеспечивается децентрализованным законом управления (4). Однако при наличии препятствий удовлетворить всем ограничениям, обусловленным структурой формации, не всегда представляется возможным, поэтому при обходе препятствий от ведомых АПР целесообразно требовать лишь удерживать заданное расстояние до своего лидера. Алгоритм управления, реализующий такое поведение ведомых, имеет вид (3), где стабилизирующая составляющая полного управления  $\mathcal{F}_s$  определяется как

$$\mathcal{F}_s = \text{sat}(k_1 \hat{s}_{1k} + k_2 \Delta \hat{u}_k + k_7 \Delta \hat{s}_k, \overline{\mathcal{F}}_s).$$

### 1.5. Синтез регуляторов

В представленных цифровых алгоритмах управления нижнего уровня синтез параметров выполнен с использованием вычислительной технологии для анализа и синтеза нелинейных систем управления на основе сублинейных векторных функций Ляпунова (ВФЛ), которая для цифровых систем управления получила развитие в [11, 20]. При построении регуляторов для режимов, связанных с отслеживанием траектории, синтез проводился из условия минимума критерия

$$J_{pfw} = \bar{y}_1 + \alpha \bar{s}_1,$$

где  $\bar{y}_1$ ,  $\bar{s}_1$  — вычисляемые с использованием ВФЛ оценки установившихся значений  $\limsup_{t \rightarrow \infty, t' \geq t} |y_1(t')|$  и  $\limsup_{t \rightarrow \infty, t' \geq t} |s_1(t')|$  переменных  $y_1$  и  $s_1$  соответственно;  $\alpha$  — весовой коэффициент. В свою очередь, для управления формацией в режиме сканирования, предполагающего как можно более точное удерживание заданной геометрической конфигурации, параметры управления ведомыми вычислялись таким образом, чтобы формация была робастно диссипативна [21, 22] и достигался минимум критерия

$$J_{frm} = \left\| \text{col}_{i=1, N} (\Delta \bar{s}^i + \alpha_i \Delta \bar{\theta}^i) \right\|,$$

где  $N$  — число ведомых в формации;  $\Delta \bar{s}^i$ ,  $\Delta \bar{\theta}^i$  — вычисляемые на основе ВФЛ оценки установившихся ошибок стабилизации положения  $i$ -го ведомого относительно своего лидера. И, наконец, в случае сбора формации при синтезе требовалось, чтобы начальные отклонения по дистанции и углу пеленга за заданное время были уменьшены до значений, лежащих в допустимом множестве режима точной стабилизации. В этом случае желаемая динамика формации определяется динамическим свойством практической устойчивости [21, 22].

## 2. Алгоритмы управления верхнего уровня

### 2.1. Дискретно-событийные системы

На верхнем уровне управления будем рассматривать систему как дискретно-событийную, т. е. описывать ее функционирование последовательностями событий, или словами

некоторого формального языка. Пусть  $\mathcal{G} = (Q, \Sigma, \delta, q_0, Q_m)$  — дискретно-событийная система, представленная генератором формального языка [23]. Здесь  $Q$  — множество состояний  $q$ ;  $\Sigma$  — множество событий;  $\delta: \Sigma \times Q \rightarrow Q$  — функция переходов;  $q_0 \in Q$  — начальное состояние;  $Q_m \subset Q$  — множество маркированных состояний. В отличие от конечного автомата, распознающего формальный язык, т. е. определяющего, принадлежит ли слово этому языку, генератор порождает слова некоторого языка. Как обычно, пусть  $\Sigma^*$  означает множество всех строк над алфавитом  $\Sigma$ , включая пустую строку  $\varepsilon$ . Префиксное замыкание  $\bar{L}$  — это множество всех строк, являющихся префиксами слов из  $L$ , т. е.  $\bar{L} = \{s | s \in \Sigma^* \text{ и } \exists t \in \Sigma^* : s \cdot t \in L\}$ . Символ  $\cdot$  означает конкатенацию строк и часто опускается. Язык  $L$  замкнут, если  $L = \bar{L}$ . Для любого генератора  $\mathcal{G}$   $L(\mathcal{G})$  замкнут.

Язык, порождаемый  $\mathcal{G}$ , — это язык  $L(\mathcal{G}) = \{w : w \in \Sigma^* \text{ и } \delta(w, q_0) \text{ определено}\}$ , тогда как язык, маркированный  $\mathcal{G}$ , определяется как  $L_m(\mathcal{G}) = \{w : w \in L(\mathcal{G}) \text{ и } \delta(w, q_0) \in Q_m\}$ . Маркированные слова могут соответствовать завершенным системой задачам, например определенным последовательностям событий, которые АПР должен осуществить для исследования заданной цели.

В настоящей статье предполагается, что  $\mathcal{G}$  полностью наблюдаема, хотя ТСУ для частично наблюдаемых ДСС — интересная и сложная теория, необходимая в реальных приложениях. Частичное наблюдение в ДСС для задач группового управления АПР будет рассмотрено в последующих работах.

## 2.2. Понятие управляемой ДСС

Теория супервизорного управления предполагает, что возникновение некоторых событий  $\mathcal{G}$  может быть предотвращено, а соответствующее средство управления представлено супервизором [23]. Пусть  $\Sigma_c$  — множество управляемых событий,  $\Sigma_{uc} = \Sigma \setminus \Sigma_c$ ,  $\Sigma_c \cap \Sigma_{uc} = \emptyset$ . Супервизор переключает схемы управления таким образом, что управляемая ДСС достигает цели управления, описываемой некоторым регулярным языком  $K$ , называемым спецификацией на поведение ДСС. Формально супервизор — это пара  $\mathcal{J} = (\mathcal{S}, \phi)$ , где  $\mathcal{S} = (X, \Sigma, \xi, x_0, X_m)$  — детерминированный автомат с входным алфавитом  $\Sigma$ .  $\mathcal{S}$  понимается как система, управляемая извне последовательностью символов событий (словами), порожденными  $\mathcal{G}$  (т. е. словами из  $L(\mathcal{G})$ ), в то время как  $\phi : X \rightarrow \Gamma$  — функция, которая отображает состояния супервизора  $x$  в схемы управления  $\gamma \in 2^\Sigma$ . Если  $\mathcal{S}$  находится в состоянии  $x$  и  $\sigma \in \phi(x)$ , то  $\sigma$  разрешено, в то время как если  $\sigma \notin \phi(x)$ , то  $\sigma$  отключено (его возникновение запрещено).

Заметим, что в отличие от ДСС-моделей с форсируемыми событиями разрешенные события не обязательно должны произойти. Очевидно, что  $\phi$  является отображением, реализующим обратную связь. Так как неуправляемые события не могут быть отключены, требуется, чтобы  $\Sigma_{uc} \subseteq \gamma = \phi(x)$ . Функция  $\delta$  доопределяется до функции  $\delta_c: \Gamma \times \Sigma \times Q \rightarrow Q$ , учитывающей схемы управления

$$\delta_c(\gamma, \sigma, q) = \begin{cases} \delta(\sigma, q), & \text{если } \delta(\sigma, q) \text{ определено и } \sigma \in \gamma; \\ \text{не определено} & \text{в противном случае.} \end{cases}$$

Построим функцию  $\xi \times \delta_c : \Sigma \times X \times Q \rightarrow X \times Q$ , где

$$(\xi \times \delta_c)(\sigma, x, q) = (\xi(\sigma, x), \delta_c(\phi(x), \sigma, q))$$

определено тогда и только тогда, когда  $\delta(\sigma, q)$  определено,  $\sigma \in \phi(x)$ , и  $\xi(\sigma, x)$  определено. Обозначим  $L(\mathcal{J}/\mathcal{G})$  язык, сгенерированный поведением замкнутой системы (объект

управления — супервизор):  $L(\mathcal{J}/\mathcal{G}) = \{w : w \in \Sigma^* \text{ и } (\xi \times \delta_c)(w, x_0, q_0) \text{ определено}\}$ . Пусть  $L_m(\mathcal{J}/\mathcal{G})$  означает язык, маркированный супервизором:  $L_m(\mathcal{J}/\mathcal{G}) = \{w : w \in L(\mathcal{J}/\mathcal{G}) \text{ и } (\xi \times \delta_c)(w, x_0, q_0) \in X_m \times Q_m\}$ . Основная цель супервизорного управления — построить такой супервизор, что  $L_m(\mathcal{J}/\mathcal{G}) = K$ . Для решения этой задачи существенно понятие управляемого языка.

**Определение 1.** [23] *Язык  $K$  управляем (относительно  $L(\mathcal{G})$  и  $\Sigma_{uc}$ ), если*

$$\overline{K}\Sigma_{uc} \cap L(\mathcal{G}) \subseteq \overline{K}.$$

При интерпретации  $K$  как допустимого поведения системы он является управляемым, если возникновение любого неуправляемого события после префикса слова из  $K$  приводит к слову из  $K$ , т. е. также допустимо. Проверка управляемости — это необходимая стадия построения супервизора. Для этого строится произведение  $H \times G$ , где  $H$  — распознаватель языка спецификации. Затем для всех  $(q_H, q_G) \in Q_{H \times G}$  проверяется включение  $E(q_G) \cap E_{uc} \subseteq E(q_H, q_G)$ , где  $E(q)$  означает множество событий, возможных в состоянии  $q$ .

**Определение 2.** *Язык  $K$  является  $L_m(\mathcal{G})$ -замкнутым, если  $K = \overline{K} \cap L_m(\mathcal{G})$  [23].*

**Определение 3.** *Супервизор  $\mathcal{J}$  называется неблокирующим, если  $\overline{L}_m(\mathcal{J}/\mathcal{G}) = L(\mathcal{J}/\mathcal{G})$  [23].*

Критерий существования супервизора определяется следующим образом: для существования неблокирующего супервизора  $\mathcal{J}$ , такого что для заданного языка  $K \subseteq L(\mathcal{G})$  выполнено  $K = L_m(\mathcal{J}/\mathcal{G})$ , необходимо и достаточно, чтобы  $K$  был управляем и  $L_m(\mathcal{G})$ -замкнутым относительно  $L(\mathcal{G})$ .

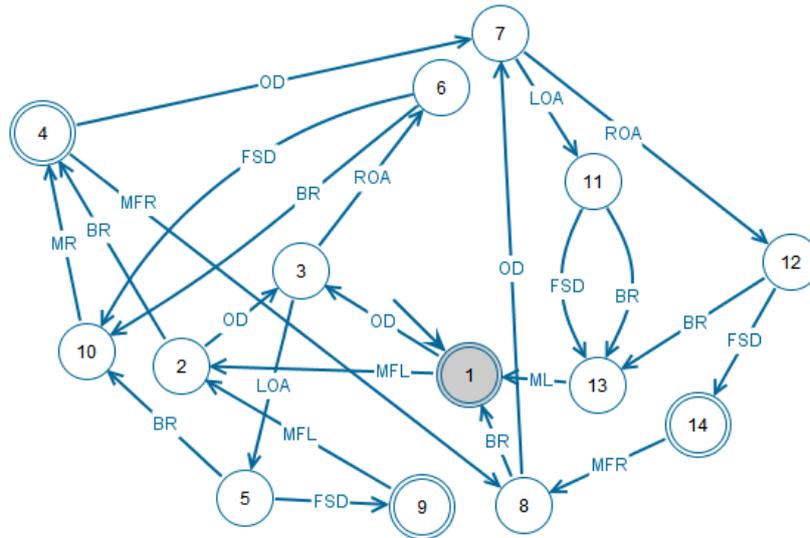
### 2.3. Дискретно-событийная модель для управления формацией АПР

Для увеличения ширины охватываемой полосы сканирования и улучшения маневренности группы для обследовательских миссий рационально использовать линейные формации, где преследователь сдвинут назад относительно своего лидера вдоль направления движения. По соображениям эффективности для лидера группы также разумно предусмотреть возможность смены своей позиции в формации в процессе выполнения миссии, тем самым различая два типа формаций: с ведомыми слева относительно лидера (левая формация) и с ведомыми справа (правая формация).

Для применения ТСУ для управления формацией АПР сначала построим генератор, описывающий переключение режимов функционирования лидера. Пусть множество состояний генератора лидера задано как  $Q_l = \{PFR \text{ (движение в правой формации)}, PFL \text{ (движение в левой формации)}, OA \text{ (режим обхода препятствий)}, W \text{ (ожидание)}, S \text{ (всплытие)}, PC \text{ (вычисление пути)}, G \text{ (режим сбора формации)}\}$ ,  $q_{0,l} = Q_{m,l} = G$ , множество событий лидера  $\Sigma_l = \{MFR \text{ (сформировать правую формацию)}, MFL \text{ (сформировать левую формацию)}, OD \text{ (обнаружено препятствие)}, ROA \text{ (обход препятствия справа)}, LOA \text{ (обход препятствия слева)}, FSD \text{ (обнаружено свободное пространство)}, BR \text{ (достигнута граница)}, FL \text{ (ведомый потерян)}, FF \text{ (ведомый найден)}, ML/MR \text{ (послать сообщение образовать левую/правую формацию)}, timeout, \text{ожидание}\}$ . Функция  $\delta$  задана согласно рис. 2.

Положим  $\Sigma_{l,uc} = \{OD, BR, FSD, FF\}$ . Модель не претендует на полноту, но отражает ключевые аспекты функционирования АПР как лидера в сканирующей миссии.



Рис. 4. Автомат  $\mathcal{S}_l$  супервизора для  $L(\mathcal{G}_l)$  и  $K_l$ 

изменить формацию, но и вычислить новый путь, а также собрать ведомых для образования новой формации. Спецификация не предполагает ожидания в режиме сбора формации и запрещает ожидание потерянных АПР. Поскольку событие  $FL$  управляемо, это не влияет на управляемость  $K_l$ , что легко проверяется. Состояния  $H_l$  имеют имена, сходные с именами состояний  $\mathcal{G}_l$ , исключительно для удобства, поскольку в общем случае никакого совпадения этих имен может не быть.  $K_l$  может представляться автоматом с произвольными именами состояний, так как нас интересует порождаемый им язык, а не способ, которым последний порождается. Маркировкой состояний  $G1$  и  $G2$  достигается  $L_m(\mathcal{G})$ -замкнутость  $K_l$ .

На рис. 4 представлен автомат супервизора  $\mathcal{J}_l = (\mathcal{S}_l, \phi_l)$ , обеспечивающего  $L_m(\mathcal{J}_l/\mathcal{G}_l) = K_l$  и построенного с помощью свободно распространяемого программного обеспечения для решения различных задач теории супервизорного управления

Т а б л и ц а 1. Отображение  $\phi_l : X_l \rightarrow \Gamma_l$ 

| $x_l \in X_l$ | $MFL$ | $MFR$ | $FL$ | $wait$ | $ROA$ | $LOA$ | $ML$ | $MR$ |
|---------------|-------|-------|------|--------|-------|-------|------|------|
| 1             | 1     | 0     | 0    | 0      | —     | —     | —    | —    |
| 2             | —     | —     | —    | —      | —     | —     | —    | —    |
| 3             | 0     | 0     | 0    | —      | 1     | 1     | 0    | 0    |
| 4             | 0     | 1     | 0    | 0      | —     | —     | —    | —    |
| 5             | —     | —     | 0    | —      | —     | —     | —    | —    |
| 6             | —     | —     | 0    | —      | —     | —     | —    | —    |
| 7             | 0     | 0     | 0    | —      | 1     | 1     | 0    | 0    |
| 8             | —     | —     | —    | —      | —     | —     | —    | —    |
| 9             | 1     | 0     | 0    | —      | 0     | 0     | 0    | 0    |
| 10            | 0     | 0     | 0    | —      | 0     | 0     | 0    | 1    |
| 11            | —     | —     | 0    | —      | —     | —     | —    | —    |
| 12            | —     | —     | 0    | —      | —     | —     | —    | —    |
| 13            | 0     | 0     | 0    | —      | 0     | 0     | 1    | 0    |
| 14            | 0     | 1     | 0    | —      | 0     | 0     | 0    | 0    |

DESUMA2 [24]. Отображение  $\phi_l$  представлено в табл. 1, где разрешению события соответствует значение 1, в то время как запрещению события соответствует значение 0. Прочерки означают, что не имеет значения, будет событие разрешено или запрещено. Неуправляемые события не включены в таблицу, поскольку они всегда разрешены. Событие *timeout* также не включено, поскольку *FL* никогда не разрешено и система никогда не достигает состояния *W*.

Для построения ДСС, описывающей основные аспекты поведения ведомого, используем традиционный модульный подход. Автомат, представленный на рис. 5, описывает базовые действия ведомого, в то время как автомат на рис. 6 служит для индикации наличия связи с лидером и перехода в режим всплытия при ее отсутствии в заданный промежуток времени.

Здесь  $Q_{f1} = \{KRF \text{ (сохранение правой формации)}, KLF \text{ (сохранение левой формации)}, OA \text{ (обход препятствия)}, G \text{ (сбор формации)}\}$ ;  $Q_{f2} = \{L \text{ (рабочий режим)}, W \text{ (ожидание)}, S \text{ (всплытие)}\}$ ;  $\Sigma_{f1} = \{MFL/MFR \text{ (двигаться в левой/правой формации)}, OD \text{ (обнаружено препятствие)}, FSD \text{ (обнаружено свободное пространство)}, FSDL/FSDR \text{ (обнаружено свободное пространство при движении в левой/правой формации)}, ML/MR \text{ (получен приказ сформировать левую/правую формацию)}, timeout\}$ ;  $\Sigma_{f2} = \{LL \text{ (лидер потерян)}, LF \text{ (лидер найден)}, timeout\}$ .

Генератор для АПП-ведомого строится как параллельная (синхронная) композиция  $\mathcal{G}_f = \mathcal{G}_{f1} \parallel \mathcal{G}_{f2}$  модулей  $\mathcal{G}_{f1}$ ,  $\mathcal{G}_{f2}$ , имеет множество состояний мощности  $|Q_{f1}| \times |Q_{f2}|$  и множество событий  $\Sigma_f = \Sigma_{f1} \cup \Sigma_{f2}$ . Заметим, что лидер и ведомый имеют общие события *MFR*, *MFL*, *ML* и *MR*, и этот факт будет использован в дальнейшей работе для построения децентрализованного супервизора.  $\Sigma_{f,uc} = \{OD, ML, MR\}$ . Мы не приводим автомат для  $\mathcal{G}_f$ , поскольку он строится стандартным образом (см., например, [25]).

Зададим спецификацию на поведение ведомого с помощью автомата  $H_f$  на рис. 7. Она предполагает запрет попыток смены формации при потере лидера, запрет всплытия во время обхода препятствия, даже если лидер потерян (следует закончить обход, после которого лидер может быть обнаружен), запрет на переход в режим сбора при наличии связи с лидером, даже если формация оказалась нарушенной при обходе препятствия, а также исключает любые действия во время всплытия.

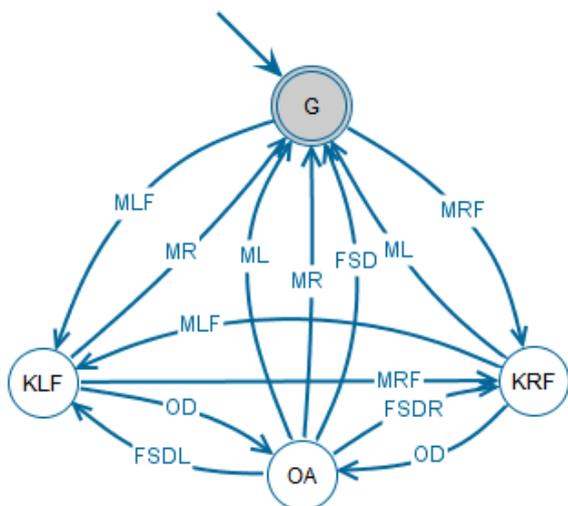


Рис. 5. Базовое поведение ведомого  $\mathcal{G}_{f1}$

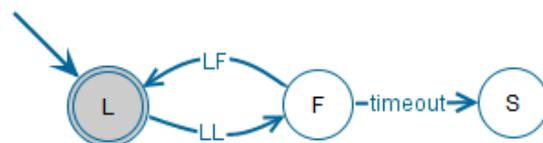


Рис. 6. Связь с лидером  $\mathcal{G}_{f2}$

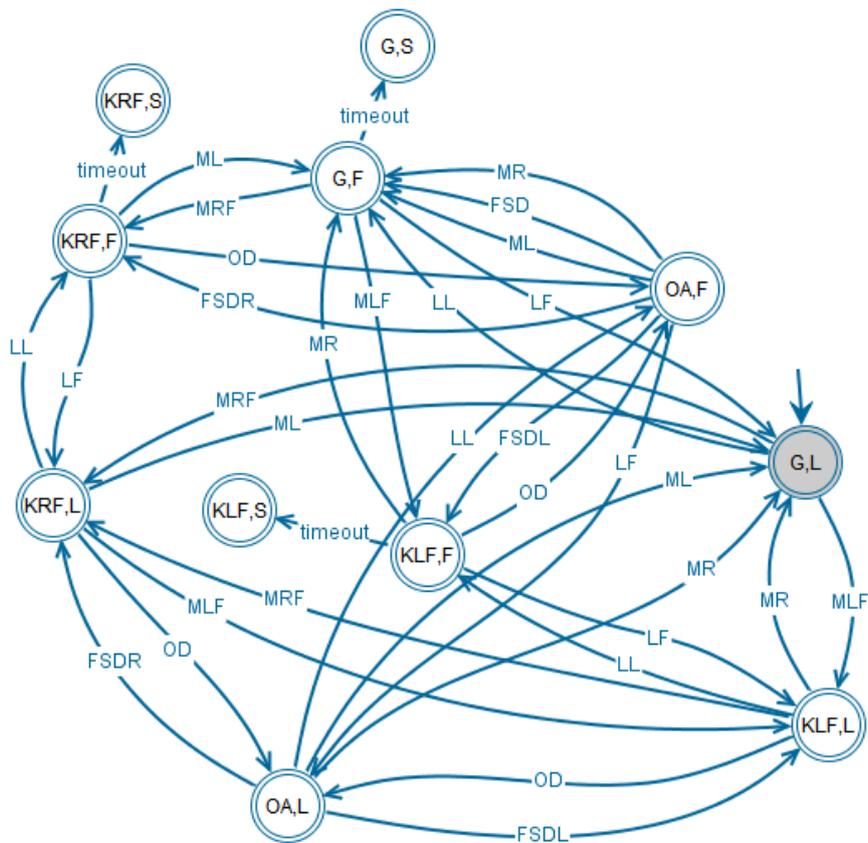
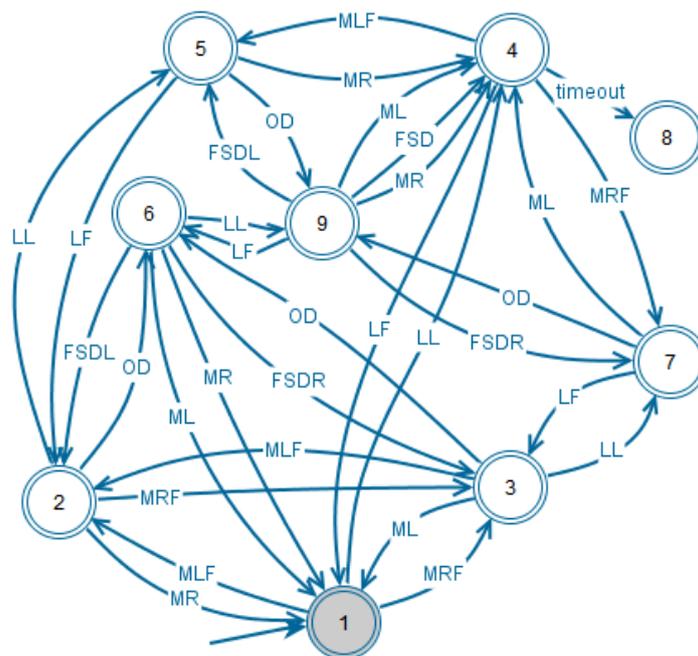


Рис. 7. Спецификация на поведение ведомого

Рис. 8. Автомат  $\mathcal{S}_f$  супервизора для  $L(\mathcal{G}_f)$  и  $K_{\max}$

Т а б л и ц а 2. Отображение  $\phi_f : X_f \rightarrow \Gamma_f$

| $x_f \in X_f$ | MFL | MFR | timeout | FSDL | FSDR | FSD |
|---------------|-----|-----|---------|------|------|-----|
| 1             | 1   | 1   | —       | —    | —    | —   |
| 2             | —   | 1   | —       | —    | —    | —   |
| 3             | 1   | —   | —       | —    | —    | —   |
| 4             | 1   | 1   | 1       | —    | —    | —   |
| 5             | —   | 0   | 0       | —    | —    | —   |
| 6             | —   | —   | —       | 1    | 1    | 0   |
| 7             | 0   | —   | 0       | —    | —    | —   |
| 8             | 0   | 0   | —       | —    | —    | —   |
| 9             | —   | —   | 0       | 1    | 1    | 1   |

Несложно проверить, что язык спецификации, заданный таким образом, неуправляем, т.е. не выполнено условие  $\overline{L(H_f)}\Sigma_{f,uc} \cap L(\mathcal{G}_f) \subseteq \overline{L(H_f)}$ : неуправляемое событие  $OD$  ведет из состояний  $(KRF, S)$ ,  $(KLF, S)$  в запрещенное спецификацией состояние  $(OA, S)$ , а неуправляемые события  $ML$ ,  $MR$  ведут из этих же состояний в состояние  $(G, S)$ , откуда, в свою очередь,  $OD$  приводит в  $(OA, S)$ . Таким образом, спецификация, заданная языком  $L(H_f)$ , не может быть удовлетворена, т.е. не существует обеспечивающего ее супервизора. Однако может быть построен супервизор, обеспечивающий максимальный управляемый подязык  $K_{max} \subset L(H_f)$ : среди других действий будут запрещены переходы по событию  $timeout$  в состояния  $(KRF, S)$  и  $(KLF, S)$ . Автомат супервизора и соответствующее отображение представлены на рис. 8 и в табл. 2.

### 3. Численные расчеты и моделирование

Численные расчеты проведены для группы однотипных крупногабаритных АПР массой  $m \approx 2200$  кг. Параметры модели АПР взяты из [3]. В левой (правой) формации для каждого ведомого принималось  $s^* = 11.66$  м,  $\theta^* = -1.03$  рад ( $s^* = 11.66$  м,  $\theta^* = 1.03$  рад). При синтезе регуляторов и численном моделировании гибридной системы также принималось:  $h = 0.2$  с (общий для всех АПР и всех режимов шаг управления),  $\mathcal{F}_s = 320$  Н

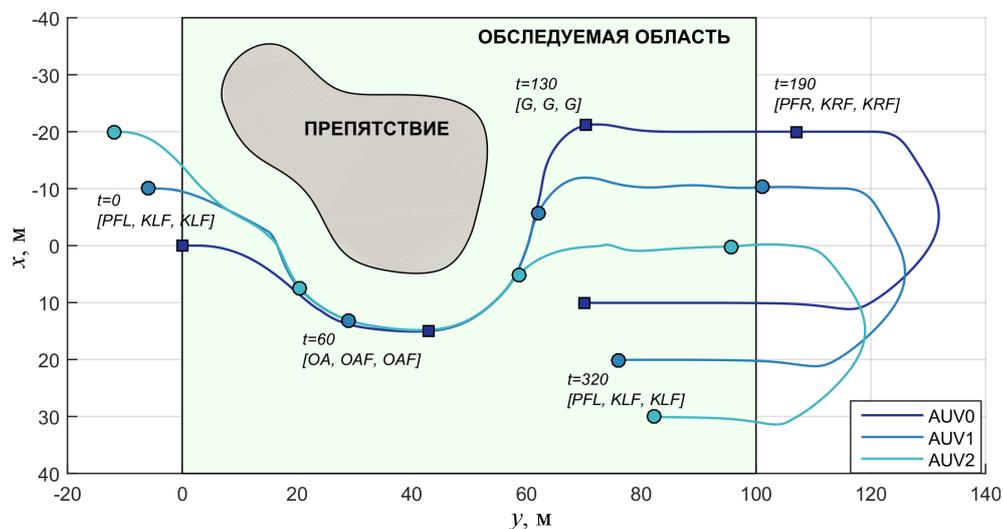


Рис. 9. Траектории АПР в обследовательской миссии

и  $\bar{G}_s = 160 \text{ Н} \cdot \text{м}$  (управляющие сила и момент, выделенные на решение основной задачи текущего режима),  $\bar{c}_c = 0.12$  (ограничение на кривизну траектории).

На рис. 9 представлены полученные моделированием траектории движения АПР группы в следующем сценарии. Аппараты начинают движение вдоль текущего галса в левой формации (AUV0 — лидер для AUV1, а AUV1 — лидер для AUV2), и в момент обнаружения препятствия по ходу движения аппараты асинхронно переключаются на режим обхода препятствий. После завершения обходного маневра и возвращения на траекторию сканирования по команде лидера группа перестраивается в правую формацию (включается режим сбора формации) для продолжения сканирования. По достижении границы сканируемой области АПР совершают разворот с перестроением формации и выходят на следующий галс. В квадратных скобках на рис. 9 приводятся режимы функционирования группы в пяти различных моментах времени. Результаты моделирования показали приемлемое качество построенной гибридной системы управления.

## Заключение

Основное внимание в этой статье уделено непрерывной динамике АПР, в то время как высокоуровневое управление на основе обработки событий представлено простейшей дискретно-событийной моделью, не учитывающей возможности ненаблюдаемости части событий (как правило, это события, связанные с отказами системы), модульного построения спецификаций и возможного в этой связи построения супервизоров на основе принципов модульности или агрегирования. С помощью такого подхода, в первую очередь, будет исследован случай, когда робот совмещает функции лидера и ведомого. Для дальнейшего исследования оставлен вопрос построения децентрализованного супервизора, использующего, при наличии определенной связи между АПР группы, решения локальных супервизоров для формирования глобального управления. Также не затронут вопрос редукции построенных супервизоров, имеющих, как правило, значительную размерность, но поддающихся редуцированию за счет исключения переходов, предопределенных управляемой системой. Для гарантирования сохранения свойств построенных супервизоров при их редукции будут использоваться результаты из [26].

В исследовательских и других сложных миссиях АПР зачастую возникают ситуации, когда необходимо выбирать одно действие из нескольких возможных. Например, при обнаружении препятствий АПР должен определить лучший способ его обхода. В общем случае такие задачи не лежат в области супервизорного управления ДСС, поэтому система управления должна иметь подсистему, отвечающую за принятие стратегических решений и планирование действий на основе знаний о подводной обстановке и состоянии АПР. В будущих исследованиях планируется использовать оригинальное исчисление позитивно-образованных формул и основанный на нем метод автоматического доказательства теорем [27] для представления этих знаний и их обработки. Кроме того, хотя алгоритмы ТСУ гарантируют, что построенное управление обладает необходимыми свойствами и поэтому не требует верификации, построение спецификаций представляет собой творческий процесс, а результат может содержать неточности и противоречия. Для гарантирования достоверности спецификаций также будет использовано исчисление позитивно-образованных формул.

**Благодарности.** Работа выполнена при финансовой поддержке РНФ (проект № 16-11-00053).

## Список литературы / References

- [1] **Petres, C., Pailhas, Y., Patron, P. et al.** Path planning for autonomous underwater vehicles // *IEEE Transactions on Robotics*. 2007. Vol. 23, No. 2. P. 331–341.
- [2] **Bagnitckii, A., Inzartsev, A., Pavin, A.** Planning and correction of the AUV coverage path in real time // *Proc. IEEE Underwater Technology Conf.* 21–24 Feb., Busan, South Korea. N.Y.: IEEE, 2017. P. 1–6.
- [3] **Lapierre, L., Soetanto, D.** Nonlinear path-following control of an AUV // *Ocean Eng.* 2007. Vol. 34, No. 11. P. 1734–1744.
- [4] **Kim, D.W.** Tracking of REMUS autonomous underwater vehicles with actuator saturations // *Automatica*. 2015. Vol. 58. P. 15–21.
- [5] **Li, X., Zhu, D., Qiu, Y.** A survey on formation control algorithms for multi-AUV system // *Unmanned Systems*. 2014. Vol. 2, No. 4. P. 351–359.
- [6] **Das, B., Subudhi, B., Pati, B.B.** Cooperative formation control of autonomous underwater vehicles: An overview // *Intern. J. of Automat. and Comput.* 2016. Vol. 13, No. 3. P. 199–225.
- [7] **Cui, R., Ge, S.S., How, B.V.E., Choo, Y.S.** Leader-follower formation control of underactuated autonomous underwater vehicles // *Ocean Eng.* 2010. Vol. 37. P. 1491–1502.
- [8] **Millan, P., Orihuela, L., Jurado, I., Rubio, F.R.** Formation control of autonomous underwater vehicles subject to communication delays // *IEEE Transactions on Control Systems Technology*. 2014. Vol. 22, No. 2. P. 770–777.
- [9] **Васильев С.Н.** Метод редукции и качественный анализ динамических систем: I // *Изв. РАН. ТИСУ*. 2006. № 1. С. 21–29.  
**Vassilyev, S.N.** Method of reduction and qualitative analysis of dynamic systems: I // *J. of Computer and Sys. Sci. Intern.* 2006. Vol. 45, No. 1. P. 17–25.
- [10] **Васильев С.Н.** Метод редукции и качественный анализ динамических систем: II // *Изв. РАН. ТИСУ*. 2006. Т. 45, № 2. С. 5–17.  
**Vassilyev, S.N.** Method of reduction and qualitative analysis of dynamic systems: II // *J. of Computer and Sys. Sci. Intern.* 2006. Vol. 45, No. 2. P. 167–179.
- [11] **Бычков И.В., Воронов В.А., Дружинин Э.И. и др.** Синтез комбинированной системы прецизионной стабилизации обсерватории “Спектр УФ”: II // *Космические исследования*. 2014. Т. 52, № 2. С. 153–161.  
**Buchkov, I.V., Voronov, V.A., Druzhinin, E.I. et al.** Synthesis of a combined system for precise stabilization of the Spektr-UF observatory: II // *Cosmic Res.* 2014. Vol. 52, No. 2. P. 145–152.
- [12] **Jayasiri, A., Mann, G.K., Gosine, R.G.** Behavior coordination of mobile robotics using supervisory control of fuzzy discrete event systems // *IEEE Transactions on Systems, Man, and Cybernetics, Pt B (Cybernetics)*. 2011. Vol. 41, No. 5. P. 1224–238.
- [13] **Torricco, R.C., Andre, B.L., Watanabe, A.T.Y.** Modeling and supervisory control of mobile robots: A case of a sumo robot // *IFAC-Papers OnLine*. 2016. Vol. 49, No. 32. P. 240–245.
- [14] **Dai, X., Jiang, L., Zhao, Y.** Cooperative exploration based on supervisory control of multi-robot systems // *Applied Intelligence*. 2016. Vol. 45, No. 1. P. 18–29.
- [15] **Tsalatsanis, A., Yalcin, A., Valavanis, K.P.** Dynamic task allocation in cooperative robot teams // *Robotica*. 2012. Vol. 30, No. 5. P. 721–730.

- [16] **Gamage, G.W., Mann, G.K.I., Gosine, R.G.** Discrete event systems based formation control framework to coordinate multiple nonholonomic mobile robots // Proc. of the 2009 IEEE RSJ Intern. Conf. on Intelligent Robots and Systems, IROS 09, St. Louis, MO, USA, 2009. P. 4831–4836.
- [17] **Lopes, Y.K., Trenkwalder, S.M., Leal, A.B., Dodd, T.J., Grob, R.** Supervisory control theory applied to swarm robotics // Swarm Intelligence. 2016. Vol. 10, No. 1. P. 65–97.
- [18] **Mendiburu, F.J., Morais, M.R.A., Lima, A.M.N.** Behavior coordination in multi-robot systems // IEEE Intern. Conf. on Automatica (ICA-ACCA). Oct. 2016, Curico, Chile. N.Y.: IEEE, 2016. P. 19–21.
- [19] **Galceran, E., Campos, R., Palomeras, N., Carreras, M., Ridao, P.** Coverage path planning with realtime replanning for inspection of 3D underwater structures // IEEE Intern. Conf. on Robotics and Automation (ICRA). 2014. 31 May – 7 June 2014, Hong Kong, China. N.Y.: IEEE, 2014. P. 6586–6591.
- [20] **Козлов Р.И., Козлова О.Р.** Исследование устойчивости нелинейных непрерывно-дискретных моделей экономической динамики методом ВФЛ. I // Изв. РАН. Теория и системы управления. 2009. № 2. С. 104–113.  
**Kozlov, R.I., Kozlova, O.R.** Investigation of stability of nonlinear continuous-discrete models of economic dynamics using vector Lyapunov function. I // J. of Comput. and Sys. Sci. Intern. 2009. Vol. 48, No. 2. P. 262–271.
- [21] **Васильев С.Н., Козлов Р.И., Ульянов С.А.** Анализ координатных и других преобразований моделей динамических систем методом редукции // Тр. ИММ УрО РАН. 2009. Т. 15, № 3. С. 38–55.  
**Vasiliev, S.N., Kozlov, R.I., Ul'yanov, S.A.** Analysis of coordinate and other transformations of models of dynamical systems by the reduction method // Proc. of the Steklov Institute of Mathematics. 2010. Vol. 268, Suppl. 1. P. 264–282.
- [22] **Васильев С.Н., Козлов Р.И., Ульянов С.А.** Устойчивость многорежимных формаций // Докл. Академии наук. 2014. Т. 455, № 3. С. 269–274.  
**Vasiliev, S.N., Kozlov, R.I., Ul'yanov, S.A.** Multimode formation stability // Doklady Mathematics. 2014. Vol. 89, No. 2. P. 257–262.
- [23] **Ramadge, P.J., Wonham, W.M.** Supervisory control of class of discrete event processes // SIAM J. Control and Optimisation. 1987. Vol. 25, No. 1. P. 206–230.
- [24] Software tool for building, analyzing controlling models of Discrete Event Systems (DES) as finite-state automata. Available at: <https://wiki.eecs.umich.edu/desuma/index.php/DESUMA> (accessed 10.09.2017).
- [25] **Cassandras, C.G., Lafortune, S.** Introduction to discrete event systems. N.Y.: Springer, 2008. 800 p.
- [26] **Nagul, N.V.** Generating conditions for preserving the properties of controlled discrete event systems // Autom Remote Control. 2016. Vol. 77(4). P. 672–686.
- [27] **Larionov, A., Davydov, A., Cherkashin, E.** The calculus of positively constructed formulas, its features, strategies and implementation // Proc. of 36th Intern. Convention on Inform. and Commun. Technology, Electronics and Microelectronics (MIPRO), May 2013, Opatija, Croatia. N.Y.: IEEE, 2013. P. 1023–1028.

**Event-based approach to multimode control of multi-AUV systems in a surveillance mission**

BYCHKOV, IGOR V., DAVYDOV, ARTEM V., NAGUL, NADEZHDA V., UL'YANOV, SERGEY A.\*

Matrosov Institute for System Dynamics and Control Theory, SB RAS, Irkutsk, 664033, Russia

\*Corresponding author: Ul'yanov, Sergey A., e-mail: sau@icc.ru

The aim of the work is to develop a new approach, which includes a combination of original methods and algorithms, to build a multilevel control system for a group of autonomous underwater vehicles (AUV), which ensures non-stop performance of a complex mission to survey the near-bottom region.

We propose a methodology, in which, considering the mission and the specific features of the underwater environment, a complex behavior of the group is divided into a set of elementary behaviors (operation modes), for which the following problems should be solved: path-following problem, formation control problem and the problem of cooperative avoidance of obstacles. Switching between modes occurs according to events that occur as a result of changing the model of the external environment and the current state of the group, and are implemented by the subsystem (DES). Low-level controllers tracking the required elementary behaviors of the group are designed with the help of the vector Lyapunov function technique which accounts sampled-data implementation of the exploited control schemes, measurement errors, uncertainties of AUV model's parameters, and constraints on control actions. To construct a distributed DES-based upper-level controller we employ the theory of supervised control of DES.

Some results of numerical computations and simulations for a group of identical large-sized AUVs are presented to demonstrate the operability of the proposed approach. Event-based cooperative control systems designed by the approach provide the level of intellectuality and sustainability of the group required to perform complex non-stop missions in real underwater conditions.

*Keywords:* underwater vehicle, formation control, vector Lyapunov function, discrete-event system.

**Acknowledgements.** This research was partly supported by RSF (grant No. 16-11-00053).

*Received 19 September 2017*