

Язык описания медицинских протоколов: разработка и апробация

А. В. Галатенко, В. В. Галатенко,
Р. Ф. Солодова, В. М. Староверов

В работе приводится постановка задачи, в рамках решения которой был разработан Язык описания медицинских протоколов (ЯОМП), описывается этот язык, по сути являющийся специализированным языком программирования высокого уровня, а также перечисляются области, в которых ЯОМП оказывается востребованным.

Ключевые слова: специализированный язык программирования высокого уровня, медицинские системы поддержки принятия решений.

Введение

В последние годы активно ведутся разработки, связанные с внедрением в медицинские аппаратные и программные комплексы систем поддержки принятия решения (СППР). СППР повышают эффективность и качество работы медицинского персонала на таких стадиях, как диагностика, выработка схемы лечения, коррекция схемы лечения с учетом динамики изменения физиологических параметров пациента. Так, согласно результатам исследования [1] число аллергических реакций, связанных с неправильным выбором антибиотика, при использовании СППР снизилось более чем в четыре раза ($p < 0,01$). Кроме того, применение СППР в сочетании с системами назначения лечения привело к снижению числа случаев передозировок препаратов также более чем в четыре раза ($p < 0,01$), а частоты побочных эффектов — в семь раз ($p < 0,02$) [1].

Большинство имеющихся на рынке систем такого рода решают определенные достаточно узкие задачи. Характерным примером

является инфузионная система с обратной связью В. Braun Space GlucoseControl, которая используется в отделениях реанимации и интенсивной терапии и обеспечивает коррекцию уровня глюкозы в крови пациента за счет оптимального динамически корректируемого выбора скорости введения инсулина, анализа параметров глюкозы крови, количества вводимого парентерального и энтерального питания [2].

Однако осуществляются и проекты, связанные с разработкой и внедрением медицинских программно-аппаратных комплексов, в состав которых входит сравнительно универсальная система поддержки принятия решений. Примером проекта такого рода является «Организация производства автоматизированного диагностического и лечебного комплекса поддержания жизнедеятельности человека» (АДЛК), осуществляемая в рамках работ по реализации Постановления правительства РФ № 218 от 09.04.2010 «Развитие кооперации российских вузов и производственных предприятий». Разработанный комплекс обеспечивает автоматизацию диагностики и лечения десяти наиболее часто встречаемых неотложных состояний (гипо- и гипергликемическая кома, острый коронарный синдром, геморрагический шок и др.).

На аппаратном уровне АДЛК включает в себя три основных блока: блок обработки и анализа информации (компьютер со специализированным программным обеспечением), прикроватный монитор пациента, обеспечивающий определение физиологических параметров (частота сердечных сокращений, частота дыхательных движений, артериальное давление и т. д.), и блок введения лекарственных средств, состоящий из нескольких инфузионных насосов, обеспечивающих автоматизированное введение лекарственных препаратов с заданными параметрами (скорость введения, вводимый объем и т. д.). Блок обработки и анализа информации в АДЛК является центральным, он получает в режиме реального времени данные от прикроватного монитора и, в случае необходимости (определяемой входящей в программное обеспечение СППР) передает управляющие инструкции инфузионным насосам.

Среди основных требований, предъявляемых к СППР АДЛК, а также к СППР других медицинских комплексов, являются:

- строгое соответствие алгоритмов утвержденным медицинским стандартам и протоколам;
- возможность расширять набор алгоритмов, а также корректировать имеющиеся алгоритмы (что необходимо, например, при изменениях в медицинских стандартах, при изменениях форм выпуска лекарственных средств) без изменения основной программной составляющей комплекса;
- универсальность, понимаемая как отсутствие жестких связей со спецификой конкретных медицинских алгоритмов и конкретного набора аппаратных составляющих комплекса.

Желательным свойством является универсальность в расширенном понимании, заключающаяся в возможности переноса алгоритмов между СППР различных модификаций комплекса и даже между различными комплексами.

В рамках создания СППР, удовлетворяющей перечисленным требованиям, был разработан специальный Язык описания медицинских протоколов (ЯОМП). ЯОМП является языком высокого уровня, включающим инструкции по получению информации (от аппаратных составляющих комплекса, таких как датчики измерения физиологических параметров пациента, а также через запросы врачу), по обработке этой информации, а также по выдаче рекомендаций и инструкций, связанных с выработкой и динамической коррекцией схемы лечения. Подробнее ЯОМП описывается ниже в разделе *Описание ЯОМП*.

Программная составляющая АДЛК включает инструментарий для компиляции ЯОМП-кода в динамически загружаемую библиотеку, входная функция которой может быть загружена из основной программной компоненты комплекса. Таким образом, добавление или изменение ЯОМП-алгоритмов вообще не затрагивают программные компоненты АДЛК.

Разработка ЯОМП-реализаций порядков оказания неотложной помощи, осуществлявшаяся совместно с группой анестезиологов-реаниматологов, подтвердила достаточность функциональных возможностей ЯОМП. Выяснилось также, что формализация порядков оказания неотложной помощи до уровня ЯОМП-кода является полезной для уточнения самих медицинского стандартов и рекомендаций, способствуя структурированию и гарантируя достаточный уровень

детализации, а структура ЯОМП-кода обеспечивает возможность верификации корректности реализации алгоритмов врачами, не имеющими специальной подготовки и существенного опыта работы с информационными системами.

Описание ЯОМП

Ниже описывается структура Языка описания медицинских протоколов и приводятся основные инструкции этого языка. Более полное описание с полным перечнем инструкций содержится в [3].

Общая структура программы на ЯОМП

При разработке ЯОМП преследовались две взаимно противоположные цели. С одной стороны, целью было создание языка программирования, столь понятного для не специалистов в области информационных систем, что врачи без специальной подготовки могли бы понимать написанный на этом языке код. С другой стороны, этот язык требовалось наделить возможностями языка высокого уровня, достаточными для его полноценного использования программистами при написании технически сложных алгоритмов.

Также в ЯОМП был реализован подход, позволяющий разбить медицинский алгоритм на относительно независимые блоки, каждый из которых обеспечивает контроль соответствующего блоку физиологического параметра (или набора физиологических параметров). Таким образом, основной единицей программы на ЯОМП является *блок*. Каждый блок запускается в отдельной нити, обеспечивая на уровне операционной системы независимость исполнения отдельных частей программы.

В ЯОМП поддерживаются блоки двух видов. К первому виду относится главный (входной) блок программы на ЯОМП, с которого начинается ее выполнение. Завершение работы главного блока программы приводит к завершению работы данного медицинского алгоритма. Главный блок имеет следующий синтаксис:

БЛОК ВХОД

...

КОНЕЦ_БЛОКА // конец БЛОК ВХОД

Многоточия здесь заменяют собственно тело блока. Символы «//», по аналогии с языком C++, трактуются как начало комментария.

Ко второму типу блоков относятся блоки контроля и поддержания физиологических параметров. После активации блока второго типа происходит его автоматически повторяющийся вызов с заданным интервалом времени. Интервал времени отсчитывается от конца выполнения блока до начала его следующего вызова. Завершение главного блока приводит к автоматическому завершению всех блоков программы, независимо от их состояния. Блоки второго вида имеют следующий синтаксис:

```
БЛОК ИМЯ_БЛОКА(интервал=Время)  
...  
КОНЕЦ_БЛОКА
```

Здесь вместо слова **Время** должно быть вставлено время ожидания после завершения выполнения блока перед его следующим выполнением. Например:

```
БЛОК ИМЯ_БЛОКА(интервал=30сек)  
...  
КОНЕЦ_БЛОКА
```

Если инструкция *интервал=Время* опущена, то задержка перед следующим выполнением считается нулевой.

Активизация блоков второго типа осуществляется инструкцией *АКТИВИРОВАТЬ_БЛОК (ИМЯ_БЛОКА)*. Полное прекращение выполнения блока осуществляется инструкцией *ВЫХОД*. Прекращение работы всего алгоритма осуществляется инструкцией *ОСТАНОВИТЬ_АЛГОРИТМ*.

Каждая ЯОМП-программа (иными словами, ЯОМП-реализация каждого *медицинского алгоритма*) описывается в отдельном файле.

В начале файла задаются глобальные переменные, через которые может осуществляться синхронизация выполнения отдельных блоков алгоритма.

Поддерживается два типа переменных: вещественные (их же можно использовать в качестве целых) и строковые. Кроме этого возможно использование массивов (размер массива задается в квадратных

скобках после его имени; массивы индексируются с нуля). Допускается инициализация переменных при определении, эта инициализация срабатывает при каждом запуске алгоритма. Регистр имен переменных несущественен.

Определение переменных имеет следующий вид:

ПЕРЕМЕННЫЕ: Список Вещественных Переменных Разделенных Запятой

СТРОКИ: Список Строковых Переменных Разделенных Запятой

Например, определение вещественных переменных может быть следующим:

ПЕРЕМЕННЫЕ: $a, b, c = 1, 2 \dots 10 = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$

Для преобразования вещественных переменных к строковому представлению (например, для вывода их в тексте сообщения) используется функция *СТРОКА*. Например, можно создать строку вида:

"Вес пациента = "+СТРОКА(Вес)

После определения переменных в файле ЯОМП-кода идет определение блоков программы.

Приведем пример простейшей программы на ЯОМП, циклически запрашивающей объем кровопотери пациента и выводящей сообщения с полученным значением.

ПЕРЕМЕННЫЕ: ОБЪЕМ

БЛОК ВХОД

АКТИВИРОВАТЬ_БЛОК(ЗАПРОС)

КОНЕЦ_БЛОКА // конец БЛОК ВХОД

БЛОК ЗАПРОС(3сек)

ОБЪЕМ = ВВОД("Введите объем кровопотери (мл)", 0, 5000)

ВЫЗОВ_ВРАЧА("Объем кровопотери: "+СТРОКА(ОБЪЕМ))

КОНЕЦ_БЛОКА // конец БЛОК ЗАПРОС

Получение информации о пациенте, ввод данных

Кроме обычных глобальных переменных в ЯОМП содержатся предопределенные псевдо-переменные, задающие основные физиологические показатели пациента. Псевдо-переменные включают *ЧСС*, *ЧДД*, *САД*, *ДАД*, *СПО₂*, *Темп*, что соответствует частоте сердечных сокращений, частоте дыхательных движений, систолическому давлению, диастолическому давлению, сатурации, температуре тела. Комплекс, в рамках которого используется основанная на ЯОМП СППР, обеспечивает обновление значений этих переменных с интервалом, определяемым инструкцией ЯОМП

УСТАНОВИТЬ_ИНТЕРВАЛ(ИмяПсевдоПеременной, Интервал)

Пример:

УСТАНОВИТЬ_ИНТЕРВАЛ(ЧСС, 5сек)

Эта инструкция задает требование комплексу получать (считывать) частоту сердечных сокращений не реже 1 раза в 5 секунд. В АДЛК значение ЧСС, как и значения других псевдо-переменных, автоматически считываются с прикроватного монитора пациента, однако возможен и режим ручного ввода значений физиологических параметров. Режим ручного ввода может использоваться комплексами, не содержащими прикроватный монитор пациента и датчики автоматического получения физиологических параметров.

Помимо псевдо-переменных, отвечающих текущим значениям физиологических параметров, в ЯОМП предусмотрены три дополнительные псевдо-переменные, ассоциированные с пациентом: *ВЕС*, *РОСТ* и *ПОЛ*. Значение этих переменных может быть задано при внесении в комплекс общей информации, ассоциированной с пациентом, или же при первом использовании этой переменной.

При необходимости обработки дополнительных физиологических параметров пациента для них необходимо завести соответствующие глобальные переменные и осуществить ручной ввод этих параметров с помощью инструкции *ВВОД*, как было показано в примере выше.

Инструкция *ВВОД* имеет синтаксис:

Переменная = ВВОД(ТекстПредложенияВвода, МинЗнач, МаксЗнач, ЗначПоУмолчанию)

Здесь обязательным является только первый параметр. Второй и третий параметры задают минимально и максимально возможные значения вводимого параметра. Если вводимый параметр выходит за заданный коридор, то выдается соответствующее предупреждение и требуется ввод корректного значения. Последний параметр инструкции задает значение, автоматически появляющееся в поле ввода при отображении диалога ввода. При отсутствии четвертого параметра поле ввода изначально является пустым.

При необходимости ввода информации, описываемой одним из нескольких предлагаемых вариантов возможных значений, могут быть использованы инструкции ввода *ВЫБОР* и *МУЛЬТИВЫБОР*.

Инструкция *Выбор* дает возможность выбрать один вариант из нескольких, например:

ПЕРЕМЕННЫЕ: продолж

БЛОК ВХОД

продолж=ВЫБОР("Продолжительность приступа более 30 сек?", "Да", 1, "Нет", 0)

КОНЕЦ_БЛОКА

Как видно из примера в параметрах инструкции *ВЫБОР* задается строка с вопросом и пары аргументов — текст данного варианта и значение, которое возвращается инструкцией при выборе соответствующего варианта.

Инструкция *МУЛЬТИВЫБОР* является более сложной. Это инструкция дает возможность выбрать по одному варианту из нескольких групп вариантов. Приведем соответствующий пример.

ПЕРЕМЕННЫЕ: с1, с2, с3, с4, с5

БЛОК ВХОД

МУЛЬТИВЫБОР("Начальный опрос. Потенциальные сердечные осложнения.",

"Беременность/(Нет=0)/Да=1", с1,

"Риск аллергических реакций на нифедипин)/(Низкий=0)/Высокий=1", с2,

"Боль//жжение за грудиной/(Нет=0)/Умеренная=1/Сильная=2", с3,

"Боль//жжение в области левой лопатки/(Нет=0)/Умеренная=1/Сильная=2", с4,

```
"Боль//исжжение в области желудка/(Нет=0)/Умеренная=1/  
Сильная=2", с5)  
КОНЕЦ_БЛОКА
```

Первым аргументом здесь является общий заголовок диалога с запросом на выбор вариантов. Далее каждый из следующих аргументов представляет собой пару (*строка, переменная*), в которой *строка* начинается с названия одной группы вариантов (вопроса) и далее содержит варианты этой группы с соответствующими им значениями. Значение, соответствующее выбранному варианту, присваивается *переменной*.

Кроме всего вышеупомянутого, в ЯОМП существует возможность задания значений переменных в программе после компиляции программы, но перед ее использованием. Такое задание осуществляется через конфигурационный файл медицинского алгоритма. Оно дает возможность коррекции пороговых и иных значений параметров медицинского алгоритма без изменения самого ЯОМП-кода. Считывание значений переменных из конфигурационного файла осуществляется через использования инструкции *ЗАДАТЬ_ЗНАЧЕНИЯ_ИЗ_ФАЙЛА*.

Обработка информации

Для обработки введенных данных используются интуитивно ясные конструкции языка ЯОМП, которые, в силу специфики языка, наиболее приближены к соответствующим конструкциям языка С. Все арифметические выражения строятся по тем же принципам, что и в языке С, то есть допустимы все арифметические выражения и операции языка С. В реальных алгоритмах, обычно, используются только основные арифметические операции =, +, -, *, /. В логических выражениях используются стандартные операции сравнения (<, >, <=, >=, ==, !=) и логические операции *И*, *ИЛИ*, *НЕ*.

Кроме того, поддерживаются следующие управляющие конструкции:

- *ЕСЛИ... ТО... ИНАЧЕ ЕСЛИ... КОНЕЦ_ЕСЛИ* Например:

```
ЕСЛИ (знач==1) ТО  
    ВЫВОД("Значение = 1")  
ИНАЧЕ ЕСЛИ (знач==2) ТО
```

```

        ВЫВОД("Значение = 2")
    ИНАЧЕ
        ВЫВОД("Другое Значение")
    КОНЕЦ_ЕСЛИ

```

- **ПОКА ... КОНЕЦ_ПОВТОРА** Например:

```

    ПОКА (Знач < 1000)
        ВЫВОД("Значение="+Строка(Знач))
        Знач=Знач*2
    КОНЕЦ_ПОВТОРА

```

- **ПОВТОРЯТЬ_В_ТЕЧЕНИЕ ... КОНЕЦ_ПОВТОРА** Например:

```

    ПОВТОРЯТЬ_В_ТЕЧЕНИЕ (10сек)
        ВЫВОД("Тело цикла... ")
    КОНЕЦ_ПОВТОРА

```

Формирование и выдача управляющих инструкций

Все управляющие инструкции ЯОМП можно подразделить на три группы:

- инструкции выдачи сообщений различного вида;
- инструкции блоку получения физиологических параметров (в АДЛК — инструкции прикроватному монитору пациента);
- инструкции блоку введения лекарственных средств (в АДЛК — инструкции блоку инфузионных насосов).

В случае, если для получения физиологических параметров используется стандартный прикроватный монитор пациента, основными управляющими инструкциями этому блоку являются *ИЗМЕРИТЬ_ДАВЛЕНИЕ* и *ИЗМЕРИТЬ_ДАВЛЕНИЕ_И_ЖДАТЬ*. Первая из этих управляющих инструкций запускает процесс измерения артериального давления и продолжает выполнение алгоритма. Вторая инструкция также запускает процесс измерения артериального давления, но при этом алгоритм продолжает работу только после окончания измерения. Периодичность измерения артериального давления (как и другие физиологических параметров) устанавливается описанной выше и инструкцией *УСТАНОВИТЬ_ИНТЕРВАЛ*.

Отметим, что для каждой из псевдо-переменных, соответствующих текущим значениям физиологических параметров (*ЧСС*, *ЧДД*,

САД, ДАД и т. д.), можно определить, содержит ли эта переменная измеренное значение, или же физиологический параметр еще не измерен и, соответственно, значение переменное еще не определено. Это осуществляется с помощью применения функции *ИЗМЕРЕНО(...)* к соответствующей псевдо-переменной. Приведем соответствующий пример:

```
ЕСЛИ (ИЗМЕРЕНО(САД)) ТО
    ВЫВОД("САД измерено")
ИНАЧЕ
    ВЫВОД("САД еще не измерено")
КОНЕЦ_ЕСЛИ
```

Кроме того, для каждой из псевдо-переменных, соответствующих текущим значениям физиологических параметров, можно определить время, прошедшее с момента последнего (успешного) измерения.

Работа с блоком введения лекарственных средств осуществляется набором управляющих инструкций на начало введения, изменение параметров введения и конец введения. В случае АДЛК эти управляющие инструкции содержат в качестве аргумента номер инфузионного насоса, на который отправляется инструкция.

Выделяются болюсное введение (в этом случае в параметрах введения задается объем) и капельное введение (в этом случае в параметрах введения задается скорость). Объем и скорость введения задаются через количество действующего вещества. При этом имеется конфигурационный файл (в случае АДЛК это ENUMS/drug.txt), в котором для каждого известного препарата перечисляются все его формы введения и для каждой формы задается пересчетный коэффициент из количества препарата в объем введения данной формы препарата (в ЯОМП разрешено использование только препаратов, информация о которых имеется в конфигурационном файле).

Начало инфузионного введения задается управляющими инструкциями ЯОМП

```
ВВЕСТИ_БЫСТРО(НомерИнф, НазвЛекарства, ДозаВведения)
ВВЕСТИ_НАЧАТЬ(НомерИнф, НазвЛекарства, СкоростьВведения,
ВремяВведения)
ВВОДИТЬ_И_ЖДАТЬ(НомерИнф, НазвЛекарства, СкоростьВведения,
ВремяВведения)
```

Инструкция *ВВЕСТИ_БЫСТРО* предназначена для начала болюсного введения на инфузионном насосе с номером *НомерИнф* препарата *Назв.Лекарства* с дозой *ДозаВведения*.

Инструкция *ВВЕСТИ_НАЧАТЬ* предназначена для начала введения (с последующим продолжением алгоритма) на инфузионном насосе с номером *НомерИнф* препарата *Назв.Лекарства* со скоростью *СкоростьВведения* в течение времени *ВремяВведения*.

Инструкция *ВВОДИТЬ_И_ЖДАТЬ* предназначена для введения препарата с ожиданием окончания ввода на инфузионном насосе с номером *НомерИнф* препарата *Назв.Лекарства* со скоростью *СкоростьВведения* в течение времени *ВремяВведения*. Примером использования управляющей инструкции, начинающей введение препарата, является команда

ВВОДИТЬ_И_ЖДАТЬ(1, НАЗВ_ПРЕПАРАТА, 60 мл/час, 1час)

Возможно задание дополнительных параметров для перечисленных управляющих инструкций. Перед номером инфузионного насоса можно добавить параметр с текстом, появляющимся в диалоге подтверждения согласия на введение. После времени введения можно задать еще несколько пар параметров *СкоростьВведения*, *ВремяВведения*. Это дает возможность, например, одной инструкцией сначала произвести болюсное введение, а потом в течение длительного времени производить капельное введение. Требуется только одно подтверждение на осуществление всей серии введений лекарственного препарата.

Следует отметить, что в АДЛК диалог подтверждения ввода препарата дает возможность сменить номер инфузионного насоса и либо согласиться на ввод, либо прекратить работу медицинского алгоритма. Все описанные инструкции начала ввода препарата возвращают номер выбранного инфузионного насоса, что дает возможность использовать его при последующем изменении скорости ввода или прекращения введения. Изменение скорости ввода осуществляется управляющей инструкцией

ИЗМЕНИТЬ_СКОРОСТЬ(НомерИнф, СкоростьВводаПрепарата).

Прекращение введения осуществляется управляющей инструкцией

ЗАКОНЧИТЬ_ВВЕДЕНИЕ(НомерИнф).

Далее приведем управляющие инструкции, осуществляющие вывод сообщений.

В ЯОМП существуют различные инструкции, используемые для вывода сообщений пользователю (врачу). Прежде всего, это инструкция *ВЫЗОВ_ВРАЧА(текст_сообщения)*. В АДЛК эта инструкция приводит к появлению диалога с текстом *текст_сообщения* на красном фоне в основной программе АДЛК и выдаче звукового сигнала. При этом в рамках одного алгоритма может одновременно выполняться несколько блоков, каждый из которых может выводить свое сообщение с помощью данной инструкции (при этом инструкция приостанавливает работу только своего блока). Инструкция *ВЫЗОВ_ВРАЧА* используется в случае необходимости срочного привлечения внимания врача. В АДЛК появление и закрытие каждого такого диалога протоколируются в регистрационном журнале.

Инструкция *Вывод(текст_сообщения)* выводит *текст_сообщения*, не приостанавливая работы блока, из которого осуществлен ее вызов. Как правило, данная инструкция используется для протоколирования технической информации о работе алгоритма.

Инструкция *Вывод_без_ожидания(текст_сообщения, время_показа)* также не приостанавливает работы блока, но выводит диалог (неяркого цвета) с данным сообщением и протоколирует данное сообщение. По истечении времени *время_показа* данный диалог исчезает. Времена открытия и закрытия данного диалога протоколируются. Как правило, данная инструкция используется для напоминания врачу о прошедших событиях (например, о завершении введения лекарства) и получения от врача подтверждения об ознакомлении с информацией.

Апробация

При активном взаимодействии с группой анестезиологов-реаниматологов, возглавляемой главным анестезиологом-реаниматологом Министерства здравоохранения РФ, были созданы и апробированы ЯОМП-реализации алгоритма общей дифференциальной диагностики для приемных отделений и отделений интенсивной терапии стацци-

онаров, специальных алгоритмов диагностики и лечения различных urgentных состояний.

Функциональных возможностей ЯОМП оказалось достаточно для естественной реализации всех этих алгоритмов. Более того, формальная реализация порядков оказания неотложной помощи в виде ЯОМП-кодов способствовала уточнению самих медицинских стандартов и протоколов.

В силу естественности и большого потенциала именно ЯОМП был выбран в качестве ядра СППР в АДЛК. СППР АДЛК уже была апробирована в стационарах Москвы, доказав свою эффективность и продемонстрировав высокий потенциал ЯОМП. В ближайшие месяцы в рамках клинических испытаний АДЛК будет собран массив информации, достаточный для формальной оценки разработанных и реализованных ЯОМП-кодов.

Заключение

Язык описания медицинских протоколов является эффективным средством формализации медицинских стандартов и рекомендаций, достаточным для интеграции этих алгоритмов в медицинские системы поддержки принятия решения. ЯОМП обеспечивает универсальность как с точки зрения реализуемых алгоритмов, так и с точки зрения аппаратных составляющих медицинского комплекса, включающего систему поддержки принятия решений с поддержкой ЯОМП. Структура ЯОМП является естественной для медиков и позволяет верифицировать корректность реализации алгоритмов даже врачам без специальной подготовки и опыта работы с информационными системами.

ЯОМП не только обеспечивает добавление существующих медицинских алгоритмов в СППР, он также эффективен на стадии разработки новых порядков оказания неотложной помощи: переход от словесного описания (принятого в медицине) до ЯОМП-формализации способствует повышению структурированности и поддержанию уровня детализации, обеспечивающего возможность реального применения алгоритма.

Кроме того, взаимодействие с основанными на ЯОМП-реализации медицинскими алгоритмами в рамках работы с СППР оказалось по-

лезным и в рамках обучения и подготовки врачей: сопоставление собственных действий с запросами и рекомендациями СППР способствовало развитию более четкого понимания причинно-следственных связей, лежащих в основе алгоритмов лечения.

Благодарности

Работа выполнена при поддержке Министерства образования и науки Российской Федерации (проект 02.G25.31.0030).

Список литературы

- [1] Evans R.S., Pestotnik S.L., Classen D.C., Clemmer T.P., Weaver L.K., Orme J.F. Jr., Lloyd J.F., Burke J.P. A computer-assisted management program for antibiotics and other antiinfective agents // *N. Engl. J. Med.* — 1998. 338 (4). — P. 232–238.
- [2] Kulnik R., Plank J., Pachler C., Wilinska M.E., Groselj-Strele A., Röthlein D., Wufka M., Kachel N., Smolle K.H., Perl S., Pieber T.R., Hovorka R., Ellmerer M. Evaluation of implementation of a fully automated algorithm (enhanced model predictive control) in an interacting infusion pump system for establishment of tight glycemic control in medical intensive care unit patients // *J. Diabetes Sci. Technol.* — 2008. 2 (6). — P. 963–970.
- [3] Староверов В. М. Полная инструкция по управляющим конструкциям ЯОМП. [<http://angel.stargeo.ru/upr.htm>]