

МЕТОД ПОСТРОЕНИЯ И ПРОВЕРКИ ГИПОТЕЗ О МЕНТАЛЬНЫХ ДЕЙСТВИЯХ ПОЛЬЗОВАТЕЛЯ ПРИ РЕАЛИЗАЦИИ ЧЕЛОВЕКО-МАШИННОГО ВЗАИМОДЕЙСТВИЯ

Абдулин Е. Р.¹

*(Учреждение Российской академии наук
Институт проблем информатики РАН, Москва)*

Предлагается метод построения и проверки гипотез о ментальных действиях пользователя в области человеко-машинного взаимодействия применительно к системам управления. Данный метод используется при создании интерфейсов систем управления с учетом их специфики и основан на сетевом моделировании и использовании сопряженного с ним методического и математического аппарата. В качестве исходных данных используется линейная модель интерфейса. Анализ построенной сетевой модели позволяет провести обоснованную корректировку исходной модели и повысить её точность.

Ключевые слова: сетевое моделирование, интегрированные системы управления, человеко-машинное взаимодействие, интерфейс.

1. Введение

При создании систем управления предъявляются особые требования к структуре человеко-машинного взаимодействия и её реализации. Это обуславливается тем, что при управлении реальным, а не виртуальным, объектом, существует гораздо меньше возможностей отмены действий пользователя или системы управления и возврата к некоторому исходному состоянию. Последствия ошибок в человеко-машинном взаимодей-

¹ Евгений Рудольфович Абдулин, аспирант, (pochtainst@yandex.ru)

вии приводят к существенным, порой трагическим, последствиям. Вероятность ошибки должна быть минимизирована также в условиях, когда на вход системы управления со стороны множества управляемых устройств поступает поток входных сигналов, вследствие чего оператору необходимо осуществлять несколько операций параллельно (необходимо начать реагировать на новый поступивший сигнал, обеспечивая обработку предыдущего сигнала). Интерфейс систем управления должен обеспечивать также условия для наиболее оперативной реакции оператора на поступление внешних сигналов как в нормальных, так и в экстремальных условиях.

Базовым подходом к построению аппаратно-программных систем, в том числе систем управления и их интерфейсов с заданными характеристиками, является экспериментальное тестирование макетов системы или её полнофункциональных образцов. Такой подход позволяет обеспечить разработчика системы достоверной информацией для коррекции системы с целью достижения последней заданных значений ключевых параметров. Однако применение способа разработки систем, основанного исключительно на экспериментальной работе, требует значительных материальных и временных затрат. Простое снижение объема экспериментов с целью снижения данных затрат приводит к ухудшению качества конечного продукта.

Альтернативой указанному подходу является применение инженерных моделей систем [9], что позволяет получить окончательную версию системы за меньшее время и при более низком уровне материальных затрат. Применение моделирования не исключает экспериментальной практики, но позволяет существенно сократить её объем и стоимость без ущерба для качества разрабатываемой системы.

В отечественной и зарубежной практике для моделирования структуры человеко-машинного взаимодействия (в том числе и в контексте данной работы, когда под человеко-машинным взаимодействием подразумевается работа пользователя с ПО систем управления) используются модели семейства *GOMS* (модели целей, объектов, методов и правил их выбора – *the models of Goals, Objects, Methods, and Selection rules*) – *KLM*,

CMN-GOMS, *NGOMSL*, *CPM-GOMS*, которые основаны на математических и лингвистических методах.

Применительно к системам управления, в которых, как было сказано выше, оператору, как основному элементу систем «человек–машина», приходится сталкиваться с необходимостью одновременного выполнения нескольких задач, для построения модели интерфейса системы необходим метод, наиболее полно отражающий протекание всех процессов, связанных с взаимодействием оператора и системы.

Одним из методов, предусматривающих реализацию в модели параллельных процессов, является сетевое моделирование. Сопряженный с ним математический аппарат позволяет проводить детальный анализ модели и получать объем информации, достаточный для завершения разработки системы.

В семействе моделей *GOMS* и сопряженных с ними методов исследований существует метод *CPM-GOMS* [5], основой которого является построение сетевой модели. Однако данный метод имеет существенные ограничения, в частности, он рассчитан на моделирование управления системой пользователем с очень высоким уровнем подготовки [7]. Построение моделей согласно этому методу требует высокого уровня знаний в области психологии и физиологии. Также модель получается очень объемной. В результате из-за высокой сложности данный метод ограниченно применим в практических исследованиях [7, 8]. На практике в качестве наиболее часто используемой модели интерфейса применяется более простая модель *KLM* [13, 11, 6] с включением различных расширений и дополнений, таких как закон Фиттса [4] и визуализация *ACT-R* [1]. Точный перевод словосочетания *Keystroke-Level Model*, являющегося расшифровкой аббревиатуры *KLM*, на русский язык весьма затруднителен, поэтому в рамках данной работы *KLM* будет переводиться как ДМД (дискретная модель действий). *KLM* является линейной моделью. Это обусловлено тем, что её структура не предусматривает какого-либо ветвления действий пользователя, которые в данном случае представлены записями, организованными в виде простого списка, соответственно, среди множеств моделей, записываемых в программной и последовательной

форме [7], *KLM* относится ко второму классу. Последовательная форма записи имеет линейный вид.

Тем не менее, вопрос применения сетевого моделирования в исследовании свойств интерфейсов систем управления остается крайне актуальным. В связи с этим в данной работе предлагается метод, позволяющий строить сетевые модели на основе исходной модели *KLM* и пользоваться сопряженным с ними математическим аппаратом. С учетом важности проведения определенного объема экспериментальной работы, данный метод позволяет осуществить проверку исходной модели на основе базовых экспериментальных данных, а также обоснованно скорректировать исходную модель в целях снижения её погрешности. Применение данного метода предусматривает, с одной стороны, экспериментальную проверку моделей, а с другой стороны – значительное снижение объема работы, связанной с проведением экспериментов.

В качестве примера систем управления в работе рассматривается такая их разновидность, как интегрированные системы управления (ИСУ), функционирующие на основе системного ПО реального времени. В качестве интерфейсного средства в ИСУ используются среднеразмерные (с диагональю 5-17 дюймов) сенсорные панели. Данные системы реализуют управление многофункциональными аппаратно-программными комплексами, такими как инженерные системы, мультимедийные комплексы и системы видеоконференцсвязи.

2. Ментальный оператор и его роль в модели интерфейса

Математическая модель *KLM* [10] для оценки временных параметров интерфейсов, предложенная в работе [2], позволяет прогнозировать время выполнения какой-либо задачи пользователем, используя в качестве исходных данных спецификацию данной задачи, а также интерфейса, который используется при её выполнении. По существу, моделирование сводится к перечислению линейной последовательности дискретных действий, которые должен осуществить пользователь, чтобы выполнить

поставленную задачу. Каждому действию ставится в соответствие определенное время, затрачиваемое на его выполнение. При этом снимается необходимость проводить измерения временных параметров выполнения действий в рамках экспериментов с участием реальных пользователей при реализации конкретного варианта интерфейса. Требования *KLM* сводятся исключительно к специфицированию пользовательского интерфейса с достаточным уровнем детализации, который позволит выявить последовательность вышеуказанных дискретных действий.

Действия относят к элементам *KLM*, если они представляют собой дискретные действия, такие, как нажатие клавиши, кнопки, перемещение манипулятора типа «мышь» и т.п. в противоположность более абстрактным действиям, таким как авторизация пользователя в системе. Основные дискретные действия, неделимые на более простые в условиях данной задачи, в терминах *KLM* называются операторами. Время выполнения пользователем таких действий рассчитано на основе экспериментальных данных.

В *KLM* в качестве стандартных используются следующие операторы (в скобках приведено полученное экспериментальным способом значение для оригинальной *KLM*) [10]:

- K (0,28 с) – однократное нажатие клавиши. Вводятся отдельные значения для различных категорий пользователей – эксперт (90 ударов в минуту, 0,12 с на удар); опытный пользователь (55 ударов в минуту, 0,2 с на удар); обычный пользователь (40 ударов в минуту, 0,28 с на удар); новичок (скорость печати неравномерна, 1,2 с на удар). Рекомендуется использовать значение 0,28 с;
- $T(n)$ – логически связанная последовательность нажатий клавиш (например, набор слова на клавиатуре), эквивалентна операторам K , взятым количестве n . Вычисляется как $T(n) = n \times K$;
- P (1,1 с) – указание с помощью манипулятора типа «мышь» на необходимый участок экрана, значение времени может быть уточнено с помощью закона Фиттса (см. ниже);

- B (0,1 с) – нажатие или отпускание кнопки на манипуляторе типа «мышь»;
- BB (0,2 с) – нажатие, а затем отпускание кнопки на манипуляторе типа «мышь» (одиночный «клик»);
- H (0,4 с) – перемещение рук с клавиатуры на мышь и с мыши на клавиатуру;
- M (0,6-1,35 с, рекомендуется использовать значение 1,2 с) – разовое ментальное действие – время, необходимое пользователю для восприятия информации или обдумывания решения в элементарном смысле;
- $W(t)$ – допустимое время ожидания пользователем ответа системы. Значение должно быть определено для каждого конкретного случая.

На практике существует необходимость уточнения времени реализации физического оператора P , поскольку время, затраченное пользователем на указание конкретного объекта на экране, напрямую зависит от расстояния между текущим положением графического устройства ввода (например, указателя манипулятора «мышь») до необходимой позиции целевого объекта, а также от линейных размеров последнего. Уточнение производится путем замены стандартного значения оператора P на вычисляемое в каждом конкретном случае по закону Фиттса (см. формулу (1)) значение [4]. Данный закон позволяет получить среднее время, за которое пользователь может переместить указатель графического устройства ввода к необходимому элементу интерфейса на экране:

$$(1) \quad t_{cp}(мс) = a + b \log_2 \left(\frac{D}{S} + 1 \right).$$

Здесь D – дистанция от начальной позиции курсора до границы целевого элемента интерфейса по направлению перемещения указателя, а S – размер целевого элемента интерфейса (кнопки, ссылки, переключателя), измеренный по сечению, задаваемому линией перемещения указателя. Величины a и b являются константами, определяемыми опытным путем по параметрам производительности конкретного пользователя. Для вычисления времени можно использовать любые единицы

измерения дистанции, так как дробь D/S является отношением двух дистанций и поэтому не зависит от единицы измерения [12]. В ходе исследования, проводимого в рамках данной работы, в качестве единиц измерения использовались экранные пиксели.

Для более детального анализа модели KLM применяется визуализация $ACT-R$ [1], которая ставит в соответствие операторам KLM некоторое подмножество более элементарных действий, например, движения рук пользователя и перемещения взгляда по экрану, представленных в распараллеленной форме. Таким образом, каждой линейной модели KLM можно поставить в соответствие визуализацию $ACT-R$ в распараллеленной форме.

Если расстановка в структуре модели KLM так называемых физических операторов ($K, T(n), P, B, BB, H$), как правило, не вызывает затруднений, то задача адекватного определения необходимого количества операторов M в модели, а также их расположения, представляется нетривиальной. Необходимо отметить, что временное значение оператора M является одним из наибольших среди существующих в модели KLM , в связи с чем при анализе интерфейсов количество операторов M имеет важное значение [10]. В связи с этим при построении моделей и проведении экспериментов необходимо иметь точную информацию о реализации пользователем ментальных действий, обозначаемых оператором M . Это даст количественную оценку наличия операторов M в структуре модели интерфейса.

В работе [2] для расстановки операторов M предлагается следующий набор последовательно применяемых эмпирических правил:

0. Начальная расстановка операторов M (перед всеми операторами K , а также перед всеми операторами P , реализующими выбор команды, но не завершающими команду).
1. Удаление операторов M перед полностью ожидаемыми операторами, например: $MPK \rightarrow PK$.
2. Если строка связок операторов вида MK принадлежит одной когнитивной единице (примером может служить имя команды), необходимо удалить все операторы M , кроме первого: $MKMKMKMKMKMKM \rightarrow MKKKKKKK$.

3. Если оператор K является избыточным разделителем, необходимо удалить стоящий перед ним оператор M : $\langle Enter \rangle \langle Enter \rangle MKMK \rightarrow MKK$.
4. а) Если оператор K завершает строку постоянного содержания (например, имя команды), то необходимо удалить оператор M , стоящий перед ним: $cd \langle Enter \rangle MKMK \rightarrow MKK$; б) при этом не следует удалять оператор M , если стоящий за ним оператор K завершает не имя команды, а её параметр: $cd class \langle Enter \rangle MKMKMKMKMK \rightarrow MKMKMKMKMK$.

Указанный набор правил с момента его разработки не претерпел существенных изменений, что подтверждается работами [7, 11, 12, 3] и является актуальным на сегодняшний день. Его применение позволяет получить приблизительную картину расстановки ментальных операторов. Однако, при детальном анализе интерфейса, а также с учетом того, что в качестве интерфейсного средства используются сенсорные панели, данная модель требует уточнения. В работе [10] для этой цели предлагается подход, в соответствии с которым операторы M включаются в модель функционирования интерфейса в следующих случаях:

- *Постановка задачи.* Пользователю требуется время для того, чтобы четко сформулировать для себя задачу, которую он должен выполнить.
- *Принятие решения.* Если существует несколько возможных вариантов и их выбор нетривиален или не доведен до автоматизма.
- *Получение единицы информации из памяти.* В качестве единицы информации могут выступать имена файлов, команд, или расшифровки аббревиатур.
- *Установление местоположения объекта на экране.* Пользователю необходимо время для просмотра текущего содержимого экрана, если местоположение необходимого объекта заранее неизвестно.
- *Установление значения параметра.* Пользователю необходимо обдумать или вспомнить необходимое значение вводимого параметра.

- *Проверка введенных данных.* Перед тем как пользователь сигнализирует системе о необходимости совершить какое-либо действие, он останавливается и проверяет данные, которые он собирается передавать системе с помощью интерфейса.

Существует также набор дополнительных рекомендаций по включению в модель дополнительных операторов M на основе указанного выше перечня:

- Каждое вводимое значение параметра должно быть получено в результате отдельного шага, представленного оператором M .
- Указанию на конкретный объект на экране должен предшествовать его поиск (оператор M), кроме случаев, оговоренных в пункте «Установление значения параметра».
- Если какой-либо объект на экране явным образом меняет свое состояние в результате действий пользователя, последнему необходимо сверить полученный результат с желаемым. Проверка обозначается в модели отдельным оператором M .

Как показывает практика, данные рекомендации могут дать положительный результат при теоретической оценке интерфейса. Однако в случаях, когда разница временных параметров интерфейсов относительно невелика (до 30%), а также при постановке задачи по проведению экспериментов по проверке модели KLM при участии пользователей разрабатываемого ПО, возникает проблема построения гипотез о расположении ментальных операторов в алгоритме действий пользователя, а также их формулирования для осуществления экспериментальных исследований. Данная проблема заключается в значительном расхождении результатов прогнозирования и экспериментальных данных, выражающихся в высокой погрешности D исходной модели. Такая погрешность во многом обусловлена систематической ошибкой в определении количества применяемых операторов M , которые, как было сказано выше, имеют наибольшее временное значение. При этом общая концепция представлений о действиях пользователя (последовательность применения операторов, её качественный состав) может быть

верной, однако, ввиду полученной значительной погрешности D , она не может быть использована в первичной форме при моделировании рассматриваемой ситуации и подобных ей.

Примером может служить распространенный случай, когда в ходе эксперимента пользователю предлагается определенное количество раз выполнить определенную типовую задачу на макете разрабатываемого интерфейса. В общем случае – при построении гипотезы о повседневном применении интерфейса для решения типовой задачи – вопрос о расстановке операторов M может быть решен путем буквального следования рекомендациям. Так, в рассматриваемом примере пользователю необходимо задавать системе определенные исходные данные с помощью интерфейса. В обычных условиях в момент реализации задачи, пользователь осуществляет ментальные действия, обозначаемые оператором M , перед каждым вводом исходных данных – для того чтобы извлечь их из долговременной памяти, сформулировать их или же проверить. В связи с этим выдвигается гипотеза о том, что ментальное действие будет предшествовать каждому вводу в систему когнитивной единицы информации, относящейся к исходным данным. На основании этой гипотезы перед каждой последовательностью физических операторов, реализующей ввод указанной когнитивной единицы, в модель вводится оператор M . Однако в ходе экспериментов при многократном повторении выполнения одной и той же задачи пользователь может не осуществлять данные ментальные действия, вводя данные согласно шаблону, размещенному в кратковременной памяти, или же осуществляя ментальное действие заранее, до очередной итерации по выполнению задачи.

Данные явления приводят к существенным различиям между выдвинутой гипотезой и результатами экспериментов. Одним из путей решения данной ситуации может быть более четкое проведение эксперимента путем постановки задачи пользователю вводить каждый раз разные исходные данные или проводить эксперименты с уменьшенной частотой. Первый способ, даже в случае следования пользователя инструкциям эксперимента, как правило, не дает нужного эффекта. Это объясняется, во-первых, различиями между представлениями пользователя о своих

действиях и о действиях, совершаемых в реальности, а вторых, самой обстановкой многократного проведения эксперимента и появления ментальной подготовки пользователя после осуществления первых итераций. При выполнении задачи в реальных условиях такая подготовка будет отсутствовать. Второй способ влечет за собой увеличение временных и материальных затрат на проведение эксперимента.

Постановка задачи: Дано множество исходных моделей M_{KLMs} , разработанных в соответствии с методом KLM , а также соответствующее ему множество визуализаций $ACT-R$ V_{ACTRS} . Также даны массивы экспериментальных данных L_{EXP} и погрешностей моделей D , где $D_i = f(M_i, L_i)$. Функция f является критерием величины погрешности исходных моделей. Её значение является параметром погрешности указанных моделей. Необходимо разработать метод, обеспечивающий минимальные значения данного параметра, согласно обозначенному критерию – функции f в смысле её области определения.

3. Применение для проверки гипотез о расстановке ментальных операторов метода, основанного на сетевом моделировании

В настоящей работе предлагается метод построения, проверки и коррекции гипотез о действиях пользователя при проведении экспериментов рассмотренного выше типа. Данный метод основан на положении, что при проведении экспериментов по выявлению наиболее эффективного варианта интерфейса из нескольких возможных реальная относительная разница в результатах будет соблюдена, если каждый раз проводить серии экспериментов с одинаковой подготовкой к ним. Иными словами, влияние ментальной подготовки пользователя будет отражено в результатах экспериментов по каждому из вариантов и не даст за счет своего существования преимущества или отставания по результатам какому-то конкретному варианту.

Однако влияние ментальной подготовки пользователя необходимо оценить как при проведении экспериментов с несколькими вариантами интерфейса, так и при оценке конкретно-

го варианта. Как следствие этого, необходимо установить, в какой именно момент времени метальное действие было реализовано, а в какой – нет.

Для решения данной задачи и предлагается изложенный в данной работе метод. Он состоит из двух компонентов.

Первым компонентом являются специальные подготовка к проведению и обработка результатов экспериментов. Эксперимент проводится таким образом, чтобы каждый шаг пользователя при выполнении задачи (а не только начало и окончание её выполнения) при взаимодействии с системой фиксировался с соответствующей временной меткой. Желательно также, чтобы фиксировались события, связанные с аспектом физического взаимодействия с интерфейсом (нажатие и отпускание кнопки, начало и окончание поворота ручек, переключения тумблеров и т.д.). Обработка же результатов должна подразумевать получение на её выходе полной экспликации действий пользователя на уровне, максимально приближенном к уровню операторов модели *KLM*. Необходимо иметь четкую информацию о том, сколько именно времени (с максимально возможной точностью) ушло у пользователя на выполнение того или иного действия в рамках реализации или решения поставленной в ходе эксперимента задачи.

Вторым компонентом является построение сетевой модели на основе расчета по модели *KLM* с использованием визуализации *ACT-R*. Визуализация *ACT-R* предусматривает представление действий пользователя в распараллеленной форме. Необходимость этого обусловлена тем, что даже при одномодальном интерфейсе структура человеко-машинного взаимодействия является многомодальной (при возможности ввода информации только посредством нажатия клавиш на клавиатуре человек взаимодействует с данным интерфейсом путем привлечения зрения, мышления, двигательных функций кистей рук, пальцев и т.д.). В связи с этим значительное время выполнения задачи в один момент времени будут задействованы параллельно несколько функций человека, что исключает возможность проведения анализа путем простого сравнения результатов моделирования и проведения экспериментов. Также сетевая модель

позволяет с высокой точностью моделировать одновременное выполнение пользователем нескольких задач при поступающем потоке входных сигналов, что является характерной чертой исследуемой в данной работе области.

При построении сетевой модели необходимо учитывать, что визуализация *ACT-R* в общем виде не предоставляет полного набора необходимых исходных данных. Прежде всего это касается связей между элементами модели *ACT-R*, которые определяют топологию сетевой модели. Как правило, связей, определенных в исходной модели *ACT-R*, недостаточно. Для доопределения необходимого количества связей между элементами модели *ACT-R* следует использовать исходную модель *KLM*, основываясь на представленной в ней линейной последовательности событий. Далее, проводя прямое сопоставление этих двух моделей, становится возможным определить для каждого элемента *ACT-R* как инициализирующие элементы (такие элементы, после совершения которых становится возможным и/или обязательным совершение рассматриваемого события), так и результирующие (элементы, для которых данный элемент является инициализирующим).

Далее каждому элементу *ACT-R* в сетевой модели сопоставляется связка вида «событие–работа», т. е. событие, из которого исходит одна работа. Событие, для которого данная работа будет входящей, определяется результирующей связью, ассоциированной с соответствующим элементом *ACT-R*. Если элемент *ACT-R* связан результирующими связями с несколькими последующими элементами, то ему будет сопоставлено событие с несколькими выходящими из него работами. Если элемент имеет несколько инициализирующих связей, то в сопоставляемое ему событие будет входить соответствующей число работ.

Таким образом, одномерная линейная структура модели *KLM* преобразуется в последовательно-параллельную структуру сетевой модели, к которой применим существующий математический и методический аппарат. В этом случае длина критического пути получившейся сетевой модели будет определять общее время выполнения задачи пользователем.

Сопоставляя полученную сетевую модель с исходной моделью *KLM*, а также результатами экспериментов в виде экспликации действий пользователя, можно провести коррекцию гипотезы о расстановке и количестве ментальных операторов *M* в модели *KLM*.

Как видно из примера на рис. 1, визуализация *ACT-R* предоставляет информацию о действиях пользователя в параллельной форме, соотнося их с той областью интерфейса (в данном примере – страницей), в которой находится пользователь в момент осуществления данных действий. Пример сетевой модели показан на рис. 3 (время указано в секундах). На рис. 2 продемонстрирован промежуточный этап построения сетевой модели.

Рассмотрим применение данного метода для решения типовой задачи в ИСУ многофункциональными мультимедийными комплексами (ММК) – подготовка данного комплекса к проведению видеопрезентации с протоколированием (подготовка подсистемы технологического телевидения, видеокоммутации и протоколирования). Для решения данной задачи разработано несколько вариантов интерфейсов. Из данных вариантов необходимо выбрать один, обеспечивающий минимальные временные затраты оператора на реализацию человеко-машинного взаимодействия (ввод в систему исходных данных, считывание результатов её работы, мониторинг внешних сигналов с помощью ИСУ). Разработанный метод применяется к двум из рассматриваемых вариантов. Первый из них представлен на рис. 4, второй – на рис. 5. Необходимо заметить, что, несмотря на различия в наборе и расположении элементов интерфейса, с точки зрения рассматриваемой типовой задачи данные варианты не имеют функциональных отличий.

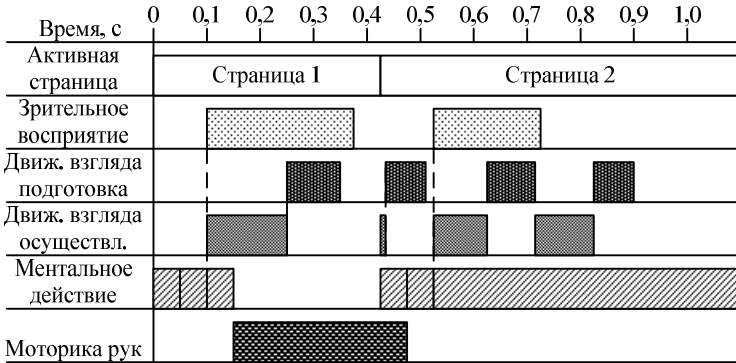


Рис. 1. Пример исходных данных – визуализации модели АСТ-R

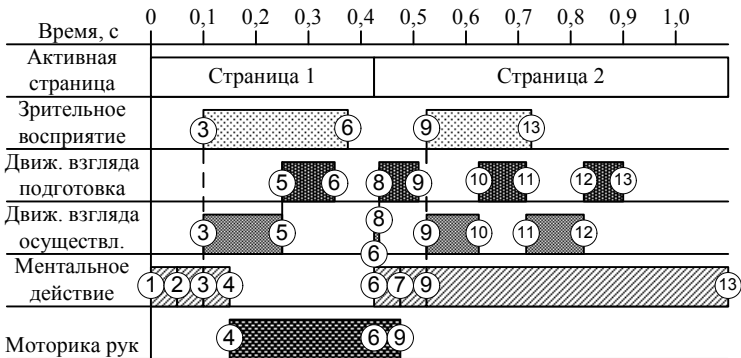


Рис. 2. Предварительная подготовка исходных данных к построению сетевой модели

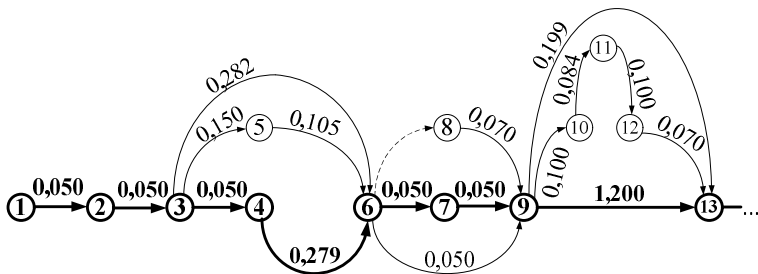


Рис. 3. Сетевая модель с выделенным критическим путем

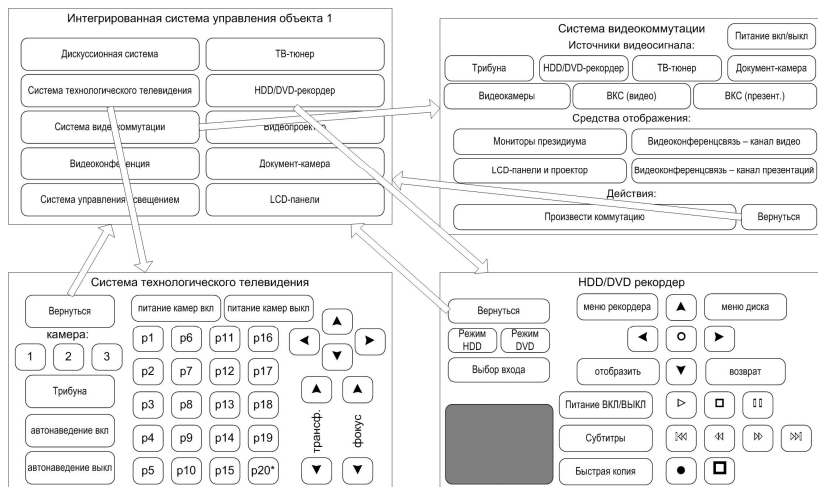


Рис. 4. Функциональный макет варианта интерфейса 1

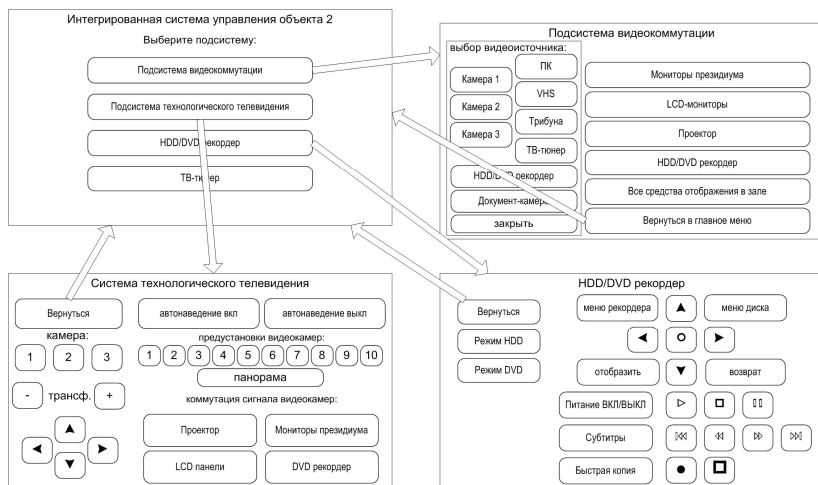


Рис. 5. Функциональный макет варианта интерфейса 2

Выполнение типовой задачи пользователем на каждом из вариантов интерфейсов условно поделено на этапы – основные (состоящие из более чем одного действия) и промежуточные (состоящие из одного действия), причем промежуточный этап не может быть включен в состав основного. Гипотеза о разме-

щении ментальных операторов в исходной модели состоит в том, что пользователь осуществляет ментальные действия в условиях данной задачи для двух видов действий – для выбора необходимого варианта действия в начале каждого основного этапа и для проверки полученного результата в конце основного этапа.

Экспериментальные исследования, в ходе которых был применен предлагаемый метод, проводились на ИСУ с конфигурацией функциональных и измерительных элементов, представленной на рис. 6.

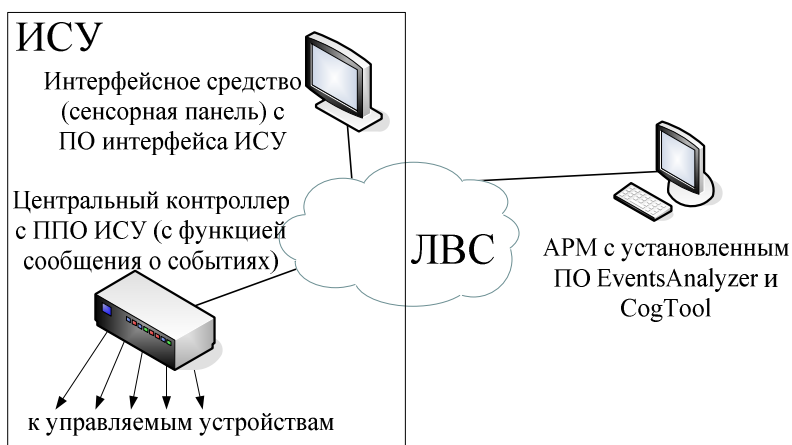


Рис 6. Конфигурация ИСУ и схема взаимодействия с используемыми измерительными средствами

При проведении измерений анализ происходящих в ИСУ событий (действий пользователя, генерации управляющих сигналов/сообщений и прием ответных сигналов/сообщений от управляемых устройств) осуществлялся при помощи системы автоматических сообщений, реализованной в прикладном ПО, функционирующем на центральном котроллере ИСУ. Каждое из сообщений было снабжено меткой системного таймера, генерирующего сигналы с интервалом 0,1 с и соответствовало одному событию, существенному с точки зрения решаемой задачи. Разность значений, указанных в метках сообщений, позволяет

получить временной интервал между любыми двумя событиями в системе. Данные сообщения передавались по протоколу *TCP/IP* через ЛВС с контроллера на ПК с установленным измерительным ПО, далее первично обрабатывались ПО производителя ИСУ, а затем собирались и систематизировались в виде статистических данных. Для автоматизации обработки большого объема собранной статистической информации использовалось специализированное ПО *EventsAnalyzer v 1.1*, разработанное автором данной статьи.

Получение исходных данных для построения сетевой модели (визуализации *ACT-R*) производится путем применения автоматизированного программного средства *CogTool*, разработанного в Университете Карнеги-Меллона под руководством профессора Бонни Джон (США, штат Пенсильвания, Питтсбург), а также ПО *EventsAnalyzer*, разработанного автором и предназначенного для автоматизированной обработки результатов экспериментов.

Результаты применения предлагаемого метода показаны в таблицах 1 и 2.

В соответствии с полученными результатами сетевого моделирования, исходная гипотеза скорректирована. Её исправленный вариант состоит в том, что пользователь осуществляет ментальные действия в условиях данной задачи для двух видов действий – на текущем промежуточном этапе, для выбора необходимого варианта основного этапа, к которому следует произвести переход, а также для выбора необходимого варианта действия в начале каждого основного этапа.

Необходимо отметить, что в приведенных в вышеуказанных таблицах результатах все же наблюдается разница во времени выполнения этапов задачи между значениями, полученными в ходе моделирования и результатами экспериментов. Тем не менее, разница на каждом из этапов принимает как положительные, так и отрицательные значения, и при подсчете общей продолжительности выполнения типовой задачи в значительной мере компенсируется, обеспечивая повышенную точность скорректированной модели. Данная разница и её взаимокompенсация носит систематический, а не случайный характер. Это обуслов-

лено тем, что реальная расстановка мест осуществления пользователем ментальных действий более равномерна, чем во многих существующих ныне моделях (не только KLM, но и CMN-GOMS, NGOMSL). Вследствие этого их продолжительность отлична от рекомендуемого значения 1,2 с. Но при правильной расстановке операторов M в модели при рассмотрении конкретной задачи, модель более точно отражает суммарное время, затрачиваемое пользователем на выполнение данной задачи. Это и обуславливает адекватность модели и её высокую точность.

Таблица 1. Применение метода для первого варианта интерфейса

| № этапа | № подэтапа | Выполняемые на данном подэтапе операторы исходной модели KLM | Ожидаемое время этапа по сетевой модели, с | Измеренное время этапа, с | Выполняемые на данном подэтапе операторы скорректированной модели KLM | Скорректированное время по сетевой модели, с |
|---------|------------|--|--|---------------------------|---|--|
| 1 | 1.1 | $M + P + K$ | 4,783 | 4,263 | $M + P + K$ | 3,513 |
| | 1.2 | $P + K$ | | | $P + K$ | |
| | 1.3 | $P + K$ | | | $P + K$ | |
| | 1.4 | $P + K$ | | | $P + K$ | |
| | 1.5 | $P + K$ | | | $P + K$ | |
| | 1.6 | $M + P + K$ | | | $P + K$ | |
| 2 | 2.1 | $P + K$ | 0,410 | 1,514 | $M + P + K$ | 1,615 |
| 3 | 3.1 | $M + P + K$ | 2,954 | 1,667 | $M + P + K$ | 1,802 |
| | 3.2 | $M + P + K$ | | | $P + K$ | |
| 4 | 4.1 | $P + K$ | 0,302 | 1,395 | $M + P + K$ | 1,552 |
| 5 | 5.1 | $M + P + K$ | 3,218 | 1,898 | $M + P + K$ | 2,116 |
| | 5.2 | $M + P + K$ | | | $P + K$ | |

Таблица 2. Применение метода для второго варианта интерфейса

| № этапа | № подэтапа | Выполняемые на данном подэтапе операторы исходной модели KLM | Ожидаемое время этапа по сетевой модели, с | Измеренное время этапа, с | Выполняемые на данном подэтапе операторы скорректированной модели KLM | Скорректированное время по сетевой модели, с |
|---------|------------|--|--|---------------------------|---|--|
| 1 | 1.1 | M + P + K | 1,936 | 1,227 | M + P + K | 1,936 |
| 2 | 2.1 | M + P + K | 4,530 | 4,804 | M + P + K | 3,330 |
| | 2.2 | P + K | | | P + K | |
| | 2.3 | P + K | | | P + K | |
| | 2.4 | P + K | | | P + K | |
| | 2.5 | P + K | | | P + K | |
| | 2.6 | M + P + K | | | P + K | |
| 3 | 3.1 | P + K | 0,391 | 1,457 | M + P + K | 1,641 |
| 4 | 4.1 | M + P + K | 3,015 | 1,723 | M + P + K | 1,863 |
| | 4.2 | M + P + K | | | P + K | |
| 5 | 5.1 | P + K | 0,398 | 1,331 | M + P + K | 1,598 |
| 6 | 6.1 | M + P + K | 3,184 | 1,711 | M + P + K | 2,001 |
| | 6.2 | M + P + K | | | P + K | |

Результаты уменьшения итоговой погрешности представлены таблице 3.

В дополнение к выводам, связанным с полученными результатами, стоит отметить роль ментальных операторов в модели, о которой говорилось выше, показать их процентное отношение к общему времени выполнения задачи пользователем в рассматриваемых моделях. Для первого варианта интерфейса ментальные операторы составляют 56,614% от общего времени выполнения задачи (остальные 43,386% приходятся на совокупность физических операторов), для второго – 58,210% (физические операторы занимают остальные 41,790%). Данные значения показывают, сколь существенную роль играют мен-

тальные операторы в моделях *KLM* и подчеркивают актуальность проведенного исследования.

Таблица 3. Итоговая таблица сравнения

| Номер варианта интерфейса | Погрешность исходной модели | Проведенная коррекция | Погрешность скорректированной модели |
|---------------------------|-----------------------------|--|--------------------------------------|
| 1 | 8,005% | Изменение расстановки операторов <i>M</i> , изменение количества операторов <i>M</i> (уменьшение на 1) | 1,274% |
| 2 | 8,057% | Изменение расстановки операторов <i>M</i> , изменение количества операторов <i>M</i> (уменьшение на 1) | 0,938% |

4. Заключение

В данной статье предложен метод построения и проверки гипотез в области человеко-машинного взаимодействия, применительно к системам управления. Данный метод основан на сетевом моделировании и использовании сопряженного с ним методического и математического аппарата. В качестве исходных данных используется модель интерфейса, разработанная в соответствии с методическим аппаратом *KLM*. Метод основан также на специальной обработке результатов экспериментов и визуализации *ACT-R*, построенной на основе исходной модели *KLM*. Анализ построенной сетевой модели позволяет провести обоснованную корректировку модели *KLM* и повысить её точность. Применение метода проиллюстрировано на примерах. Показано увеличение точности исходной модели за счет уменьшения её погрешности с 8% до 1–2%.

Предлагаемый метод был применён на практике при разработке ПО для ИСУ в интересах следующих организаций:

- Национального Банка Республики Татарстан (ИСУ многофункциональным мультимедийным комплексом учебного класса с возможностью оперативного преобразования в опорный центр технической поддержки ТУ Банка России по Приволжскому Федеральному округу);
- Главного Управления Банка России по Астраханской области (ИСУ многофункциональным мультимедийным комплексом конференц-зала, ИСУ распределенной системы видеоконференцсвязи);
- Главного Управления Банка России по Ульяновской области (ИСУ многофункциональным мультимедийным комплексом актового зала, ИСУ многофункциональным мультимедийным комплексом конференц-зала);
- Главного Управления Банка России по Воронежской области (ИСУ многофункциональным мультимедийным комплексом актового зала).

Метод применим и для более сложных по содержанию и структуре моделей *KLM*. При решении более масштабных задач прогнозируется уменьшение погрешности с 15% до уровня не более чем 6%. Ещё более значимого эффекта возможно достигнуть при моделировании работы системы, предусматривающей одновременное выполнение оператором систем управления нескольких операций, так как в этом случае потенциал аппарата сетевого моделирования будет задействован наиболее полно. Данный аспект применения метода, равно как и работу оператора рассматриваемых систем управления в экстремальных условиях, можно оценить при проведении дополнительного исследования.

Литература

1. ANDERSON J.R., LEBIERE C. *The Atomic Component of Thought*. – Hillsdale, NJ: Lawrence Erlbaum Associates, 1998.
2. CARD S.K., MORAN T.P., NEWELL A. *The psychology of human-computer interaction*. – Hillsdale, NJ: Lawrence Erlbaum Associates, 1983.
3. CAROLL JOHN M. *HCI models, theories, and frameworks*:

- toward a multidisciplinary science.* – San Francisco: Morgan Kaufmann Publishers, 2003.
4. FITTS P.M. *The information capacity of the human motor system in controlling the amplitude of movement* // Journal of Experimental Psychology. 1954. № 47. P. 381 – 391.
 5. GRAY W.D., JOHN B.E., ATWOOD M.E. *Project Ernestine: Validating GOMS for predicting and explaining real-world task performance* // Human Computer Interaction. 1993. №8(3). P. 237-309.
 6. JOHN B., PREVAS K., SALVUCCI D., KOEDINGER K. *Predictive Human Performance Modeling Made Easy* // Proceedings of the SIGCHI conference on Human factors in computing systems. 2004. P. 455 – 462.
 7. JOHN B.E., KIERAS D.E. *The GOMS family of user interface analysis techniques: Comparison and contrast.* ACM // Transactions on Computer-Human Interaction. 1996. №3. P. 320 – 351.
 8. KIERAS D. E., JOHN B. E. *Using GOMS for User Interface Design and Evaluation: Which Technique?* // ACM Transactions on Computer-Human Interaction, 1996. № 3, P. 287 – 319.
 9. KIERAS D.E. *A Guide to GOMS Model Usability Evaluation using NGOMSL* // Handbook of human-computer interaction. (Second Edition). Amsterdam: North-Holland. 1997. P. 733-766.
 10. KIERAS D.E. *Using the Keystroke-Level Model to Estimate Execution Times*, On-line handout, 2001. URL: <http://www.eecs.umich.edu/~kieras/goms.html> (дата обращения: 06.11.2010).
 11. LUO LU, JOHN B.E. *Predicting Task Execution Time on Handheld Devices Using the Keystroke-Level Model* / Conference on Human Factors in Computing Systems (CHI 2005), Extended abstracts on Human factors in computing systems. – 2005. P. 1605 – 1608.
 12. RASKIN JEF. *The Humane Interface.* – Crawfordsville: Addison-Wesley, 2000.
 13. TEO L., JOHN B.E. *Comparisons of Keystroke-Level Model predictions to observed data* / Extended Abstracts of CHI. 2006. P. 1421–1426.

METHOD FOR HYPOTHESES MAKING AND TESTING ABOUT USER MENTAL ACTIONS IN HUMAN-COMPUTER INTERACTION

Evgeniy Abdulin, Institute of Informatics Problems of RAS, Moscow, post-graduate (pochtainst@yandex.ru).

Abstract: The new method is offered for hypotheses making and testing about user mental actions in human-computer control systems. This method is used to create interface design for the control systems taking into consideration their specific character. The method is based on the critical-path method and its methodical and mathematical tools. The linear model of interface is used as input data. The analysis of critical-path chart allows making reasonable corrections to the source model and increasing its accuracy.

Keywords: critical-path method, integrated control systems, human-computer interaction, interface.

Статья представлена к публикации членом редакционной коллегии М. В. Губко