

РАЗДЕЛ 5

**ИНТЕЛЛЕКТУАЛЬНЫЕ
ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ
И МАТЕМАТИЧЕСКИЕ МЕТОДЫ
ДЛЯ ПОВЫШЕНИЯ
ЭФФЕКТИВНОСТИ
УПРАВЛЕНИЯ**

ГЛАВА 5.1

ИНТЕЛЛЕКТУАЛЬНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ПРОЕКТИРОВАНИИ ЧЕЛОВЕКО-МАШИННЫХ ИНТЕРФЕЙСОВ*

С конца 1990-х гг. начался очередной виток вовлечения широких слоев общества в использование информационных технологий (ИТ), связанный с развитием телекоммуникационных сетей и появлением глобальной сети Интернет. Уже за период с 2000 по 2010 гг. общее количество пользователей Интернета в мире выросло в 5,5 раза (до 2 млрд чел.) [1], в России, по данным Яндекса, в 2015 г. количество пользователей достигло 78 млн чел. (67 % от совершеннолетнего населения страны). Схожую быструю динамику роста показывало и количество веб-сайтов: если в 1997 г. их общее количество в мире составляло около 1 млн, то уже через 10 лет – около 100 млн, а в настоящее время приближается к 1 млрд [2]. Соответствующими высокими темпами в нашей стране шло и развитие отрасли услуг, связанных с информационными технологиями: по результатам 2011 г. суммарный объем реализации крупнейших российских ИТ-компаний достиг рекордной величины в 508,4 млрд руб. [3], причем максимальным (46 %) оказался прирост в сфере разработки программного обеспечения (ПО), а по итогам 2015 г. объем только экспорта российского ПО за рубеж достиг \$ 7 млрд США [4].

С 1980-х гг. и по настоящее время наиболее популярным средством взаимодействия с ПО остается так называемый графический интерфейс пользователя, который, как правило: предусматривает прямую манипуляцию объектами и использование меню; задей-

* Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 16-37-00184 мол_а.

ствуется зрение, слух и движение верхних конечностей человека; широко использует графические элементы (окна, кнопки, вкладки и т. д.); позволяет применять такие устройства, как монитор, мышь, сенсорный экран, клавиатура и др.; имеет слабо выраженную интеллектуальность и направлен на прямое взаимодействие. Разнообразием графического интерфейса является веб-интерфейс, основной особенностью которого является то, что он работает в специальной программе – «браузере», функционирующей на компьютере пользователя, в то время как данные и логика веб-приложения размещаются на удаленном веб-сервере. Вследствие этого веб-интерфейсам исторически были свойственны ограниченность графических элементов (поскольку их перечень определялся стандартами языка HTML) и невозможность прямой манипуляции объектами, а также зависимость от пропускной способности канала связи с веб-сервером. Однако в настоящее время, по мере развития и распространения таких технологий, как Java, Flash, DHTML и других, позволяющих создавать фактически полноценные графические интерфейсы, а также повышения пропускной способности интернет-каналов, различия веб-интерфейсов в определенной степени нивелировались. Следует заметить, что современные веб-сайты, как правило, реализуются не в виде набора статичных веб-страниц (что затруднило бы поддержку веб-сайта и сделало невозможным выдачу посетителям персонализированных данных), а с применением средств динамической организации содержимого. Такими средствами обычно являются исполняемая на сервере программная часть и база данных, в то время как пользовательский интерфейс веб-сайтов отображается в браузере. Это позволяет заключить, что большинство современных веб-сайтов является клиент-серверными веб-приложениями (или набором таких приложений), некоторые аспекты проектирования человеко-машинного взаимодействия для которых мы рассмотрим далее в нашей работе.

АВТОМАТИЗАЦИЯ И ИНТЕЛЛЕКТУАЛИЗАЦИЯ ПРОЕКТИРОВАНИЯ ВЗАИМОДЕЙСТВИЯ В ЧЕЛОВЕКО-МАШИННЫХ СИСТЕМАХ

В современных человеко-машинных системах управления операторы-люди остаются неотъемлемой частью, получая, обрабатывая и выдавая информацию аналогично другим компонентам си-

стемы. Хотя по сравнению с машинами человек имеет ряд достоинств, таких как целостность, адаптивность и возможность к саморазвитию, в то же время участие его остается отчасти ограничивающим фактором в процессах автоматизации [5]. Представляется, что основная причина этого – нерешенная проблема оптимального взаимодействия между человеком и компьютером. В частности, когнитивный аспект в человеко-компьютерном взаимодействии остается менее изученным по сравнению с физическим (эргономическим), поскольку окончательного понимания работы человеческого мозга и памяти нет по сей день.

Инструментальная поддержка проектирования взаимодействия. Известно, что при создании программных систем наиболее дорогостоящим является исправление ошибок, допущенных на начальных стадиях процесса; считается, что если ошибка произошла при анализе требований, то стоимость ее устранения к этапу тестирования возрастет в 10 раз, а после внедрения – до 100 раз [6]. Не является исключением и проектирование человеко-машинного взаимодействия [7] – применение соответствующих методов на ранних стадиях процесса разработки позволяет достигать до 10 раз большего улучшения показателей качества интерфейса, чем на поздних стадиях (т. е. тестирования и внедрения) [8]. Соответственно среди инструментов, предназначенных для создания человеко-машинных интерфейсов, более эффективными представляются те, которые основной акцент делают на стадиях анализа требований и проектирования.

Существующие на настоящий момент в сфере создания ПО универсальные инструменты предоставляют некоторые возможности для анализа и формализации требований, существенных при проектировании взаимодействия, например, посредством поддержки нотации UML, которая может применяться для записи прецедентов использования. В качестве примера таких инструментов можно привести IBM Rational RequisitePro или CaseComplete, которые, впрочем, напрямую не поддерживают исследование взаимодействия будущего пользователя с программным продуктом и анализ требований, относящихся к показателям качества интерфейса. Используемые при создании ПО готовые компоненты, библиотеки, а для веб-приложений – и системы управления содержимым (CMS) направлены в основном на удовлетворение функциональных требо-

ваний и затрагивают реализацию пользовательского интерфейса лишь в минимальной степени.

Среди инструментов, непосредственно поддерживающих создание проекта интерфейса или его прототипа, можно отметить графические редакторы (универсальные, как Adobe Photoshop, или специализированные, как Balsamiq Mockups), средства создания интерактивных приложений (в первую очередь Adobe Flash), а также HTML-редакторы, предоставляющие пользователю возможности визуально редактировать веб-страницу и ее элементы (например, Adobe Dreamweaver и MS Expression Web). Подобные инструменты, воплощающие методы высокоуровневого проектирования интерфейсов и способные существенно снизить трудоемкость разработки визуального компонента интерфейса, относят к так называемой дизайнерской парадигме [9]. Однако средства, созданные в рамках данного подхода, способны лишь фиксировать задуманный разработчиком проект – они не содержат знаний о принципах проектирования взаимодействия, которые могли бы применяться для принятия обоснованных проектных решений.

Многие из существующих интеллектуальных систем в сфере проектирования взаимодействия направлены на организацию рекомендаций и шаблонов проектирования. Так, еще в 2000 г. проводилась специальная конференция Special Interest Group Tools for Working With Guidelines, посвященная такому типу ИС [10]. Среди значимых работ в данной области можно отметить публикации J. Vanderdonck и его соавторов, которые наметили основные направления развития подобных систем и предложили конкретную их реализацию [11]. Эти наработки нашли свое воплощение в системе MetroWeb [12], положительный эффект от использования которой показан на экспериментальных данных, хотя отмечается, что проектировщики неактивно прибегали к поддержке системы. Среди причин этого нам видится то, что упомянутая выше система затрагивает в основном задачи пользователя и не делает акцента на учете характеристик различных категорий пользователей; кроме того, в соответствии со своим назначением ИС, направленная на организацию рекомендаций, не охватывает всего процесса проектирования взаимодействия, вынуждая проектировщиков переключаться с одного инструмента на другой.

Подробный обзор подобных систем констатирует, что существующие ИС не дают значительного эффекта при проектировании взаимодействия [13]. Тем не менее создание таких инструментов признается актуальным направлением, причем отмечается, что главной идеей должна стать не автоматизация применения шаблонов проектирования, а интеллектуальная организация знаний, за счет использования которых проектировщики смогут достичь более высокого уровня профессиональной компетенции [14].

МОДЕЛЕОРИЕНТИРОВАННЫЙ ПОДХОД К АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ ВЗАИМОДЕЙСТВИЯ

Еще одна группа инструментов (большинство из которых также является системами, основанными на знаниях) – программные средства, автоматизирующие создание интерфейса или его валидацию на основе модели интерфейса (моделеориентированная парадигма). Цель большинства таких систем – снижение сложности (трудоемкости) создания и поддержки пользовательских интерфейсов или обеспечение автоматической адаптации программных продуктов к особенностям различных платформ. При этом общим принципом их работы является создание модели интерфейса, которая может включать в себя декларативную модель информации, обрабатываемой при взаимодействии, принципы соотнесения элементов интерфейса и т. д. В большинстве случаев модель интерфейса задается непосредственно, хотя некоторые системы выводят ее из существующего программного кода (например, Mickey, HUMAN-OID), модели базы данных (GENIUS) или высокоуровневых задач пользователя. На основе модели может генерироваться логическое представление интерфейса (независимое от языка или платформы), а затем – непосредственно программный код (хотя в некоторых случаях предполагается наличие внешнего по отношению к системе генератора кода), причем генерация может основываться на правилах базы знаний или на оптимизации каких-либо характеристик интерфейса или прогнозируемых критериев качества [15].

Так, система, разработанная в ИАПУ ДВО РАН коллективом под руководством В.В. Грибовой и А.С. Клещева, позволяет фиксировать решения на этапе проектирования, проводить автоматическую оценку качества интерфейса и т. д. [9]. Система базируется на

онтологическом подходе, что подразумевает выделение системы понятий, относящихся к пользовательскому интерфейсу (задачи пользователя, сценариев диалога, представления информации и др.), построение модели интерфейса в терминах понятий и автоматическое преобразование ее в программный код. При этом для обеспечения качества интерфейса используется база метрик юзабилити, также сопровождаемая онтологией их описания и специализированным языком для задания алгоритмов вычисления метрик.

Существуют также решения, способные автоматически генерировать интерфейсы для относительно простых классов задач или для особых условий взаимодействия. Так, применение системы PUC, предназначенной для создания стандартизованных интерфейсов управления различной бытовой техникой, позволило увеличить показатели качества взаимодействия в 2...4 раза [16]. При этом язык, специально разработанный для описания моделей интерфейсов, был упрощен исходя из особенностей решаемой задачи: например, он не включает в себя возможность описания задач пользователя, так как использование бытовых устройств, как правило, не подразумевает сложного многошагового взаимодействия. Алгоритм генерации программного кода основывается, в частности, на эвристических правилах, обеспечивающих стандартизацию внешнего вида создаваемых интерфейсов, и на онтологическом механизме описания семантической идентичности терминов, применяемых в спецификациях различных бытовых устройств.

Система SUPPLE, практическая применимость которой была продемонстрирована в 2010 г., создает альтернативные интерфейсы для пользователей, чьи потребности не были учтены в основном интерфейсе продукта или устройства, повышая эффективность их работы в среднем на 62 % [15]. Авторы системы отмечают, что она наиболее приспособлена к созданию стандартных интерфейсов на основе диалоговых окон, поскольку для них существует проработанный словарь, описывающий возможные элементы взаимодействия. При этом генерация интерфейса рассматривается как задача дискретной оптимизации, а входными данными являются функциональное описание интерфейса, модель возможностей и ограничений платформы, модель использования интерфейса и целевая функция, включающая параметризованные показатели качества. Такая целевая функция в основном охватывает физические параметры взаи-

модействия (например, время движения между элементами интерфейса, их размер), однако примечательно, что она может учитывать особенности использования интерфейса конкретной категорией пользователей (например, людьми с нарушениями моторных способностей).

Принципиальным ограничением на пути моделирования парадигмы автоматизации создания интерфейсов является необходимость предварительного задания подробной модели взаимодействия – задач пользователя, обрабатываемых данных и т. д. В случае если результатом должен быть сложный интерфейс с высоким уровнем детализации, трудоемкость задания входных параметров для его генерации может превысить затраты на проектирование обычными методами. Для преодоления этого ограничения предлагаются подходы к автоматизированному построению модели интерфейса на основе некоторой входной информации или анализа данных о реальном использовании интерфейсов, а также сужение класса рассматриваемых продуктов (в системе PUC – бытовая техника) или пользователей (в системе SUPPLE – люди с ограниченными возможностями).

На практике использование систем автоматической генерации кода интерфейса также затруднено, поскольку в большинстве программных продуктов он не полностью отделяется от бизнес-логики и алгоритмов работы. Веб-приложения являются исключением, так как использование языка HTML предполагает разделение представления и обработки информации, однако его же принципиальная особенность заключается в том, что HTML-интерфейсы по-разному отображаются у различных пользователей (в зависимости от параметров экрана, используемого интернет-браузера и т. п.). Это снижает эффективность генерации интерфейса на основе оптимизации его физических характеристик (например, в SUPPLE – расстояния между элементами интерфейса и их размера). Кроме того, параметризация целевой функции является нетривиальной задачей для случая взаимосвязанных показателей качества, большинство из которых в веб-приложениях к тому же относятся к когнитивному аспекту взаимодействия.

В целом исследователи признают, что на настоящий момент автоматически сгенерированные интерфейсы качественно уступают решениям, создаваемым экспертами (проектировщиками), поэтому

они предназначены скорее для охвата тех категории пользователей, которые не получили бы специально адаптированного для них экспертного решения (например, вследствие экономической нецелесообразности). Следующий раздел нашей работы посвящен важности учета когнитивного аспекта взаимодействия и существующим подходам к измерению потоков информации между человеком и компьютером в человеко-машинных системах.

КОГНИТИВНЫЙ АСПЕКТ ВЗАИМОДЕЙСТВИЯ В ЧЕЛОВЕКО-МАШИННЫХ СИСТЕМАХ

Выражение «Пользователь всегда прав» стало обязательным девизом для любого разработчика интерфейсов и уже более десяти лет считается очевидным и не допускающим сомнений, а проектировщики интерфейсов и промышленные дизайнеры прилагают все усилия для создания у пользователя положительных впечатлений. Тем не менее в некоторых случаях должна рассматриваться и максимизироваться общая эффективность человеко-машинной системы – как происходило, например, с 1970-х гг. в ходе активной разработки и внедрения автоматизированных систем управления (АСУ) в ряде ведущих стран мира. Специально обученный оператор или аналитик рассматривался как составная часть человеко-машинной системы управления, получая, обрабатывая и выдавая информацию так же, как любой другой ее компонент. При этом отмечалось, что человек является неотъемлемой частью такой системы, выполняя функции интегрирующего звена, придавая ей способность адаптироваться, возможность саморазвития [17].

При всех своих сильных сторонах человек уступает машине в способностях по получению больших объемов информации, и отсутствие учета этого аспекта ведет к дорогостоящим ошибкам, возникающим по вине человека, особенно в системах управления. Можно также предположить, что в настоящее время развитие средств автоматизации проектирования взаимодействия ограничивается, в частности, невозможностью формальной оптимизации информационных потоков между компьютером и пользователем из-за отсутствия количественной меры [18].

Вероятно, наиболее очевидным подходом к количественному измерению передаваемой информации стало бы вычисление ин-

формационной энтропии согласно знаменитой формуле К. Шеннона. Действительно, У. Хик, будучи одним из первых, кто применил положения теории информации в психологии, предложил выражение для определения времени реакции человека (RT) в зависимости от количества рассматриваемых альтернатив (N):

$$RT \sim k \log_2(N + 1), \quad (1)$$

где коэффициент k – скорость переработки информации.

Конечно, (1) является частным случаем (для равновероятных альтернатив) более фундаментального линейного соотношения между RT и энтропией набора стимулов (H_T), которое было предложено Р. Хайманом:

$$RT = a_H + b_H H_T, \quad (2)$$

где a_H и b_H – константы, определяемые эмпирическим путем, которые могут существенно варьироваться в зависимости от характеристик конкретного человека, типа и интенсивности стимулов, обстановки и т. д.

Однако самая существенная проблема, препятствующая широкому применению закона Хика–Хаймана (2) в сфере проектирования человеко-машинных интерфейсов, заключается в том, что остается в высшей степени неясным, как именно вычислять информационную энтропию при взаимодействии на практике [19]. Анализ существующих исследований привел нас к заключению, что на настоящий момент нет общепризнанного метода измерения количества информации, передаваемой между человеком и компьютером в контексте реального взаимодействия, хотя желательность его довольно широко известна.

Действительно, при должном развитии теорий, описывающих, как именно люди обрабатывают информацию, каковы их производительность и ограничения при этом, сложность пользовательских интерфейсов можно было бы обдуманно минимизировать, что и рекомендуется многими источниками [20]. В теоретической информатике есть такие области, как теория вычислительной сложности и анализ алгоритмов, но на настоящий момент степень их применения в проектировании взаимодействия незначительна. В итоге

сложность в интерфейсах в основном рассматривается на качественном уровне, а измерения в лучшем случае проводятся согласно известному закону Миллера, касающемуся объема рабочей памяти человека (7 ± 2 единицы информации). Следует, впрочем, заметить, что этот закон представляется действительно фундаментальной закономерностью, которая, возможно, применима и к программным интерфейсам, например, количество взаимодействий для каждого компонента в программной системе рекомендуется [21] делать равным 5.

Существует достаточно простая формула:

$$T = 0,03H, \quad (3)$$

где T – время, необходимое для получения и обработки информации.

Автор считает, что используемая в ней константа в лучшем случае соответствует усредненному значению диапазона, а при проектировании взаимодействия должна учитываться специфика различных пользователей, задач и контекстов использования.

СПОСОБЫ ИЗМЕРЕНИЯ ПЕРЕДАВАЕМОЙ ИНФОРМАЦИИ

Автор предлагает рассматривать графический интерфейс пользователя как сообщение, определять его «длину» и использовать ее как меру сложности визуального поиска, осуществляемого при выполнении задач по выбору объектов. Показатель «длины интерфейса как сообщения» (IML) рассчитываем по следующей формуле:

$$IML = \log_2(S_0 / S), \quad (4)$$

где S_0 – размер (площадь) рабочей области на экране для простейшего размещения элементов по сетке; S – размер (площадь) квадратной цели.

IML , таким образом, отражает количество информации, которая должна быть воспринята и обработана человеком для когнитивного выбора нужного элемента интерфейса из числа представленных в рабочей области на экране.

Источники также упоминают понятие «информационной емкости интерфейса» (H) как меры информации, которая воспринимает-

ся человеком при использовании графического интерфейса пользователя. Существует, однако, определенная неясность [18] относительно того, должна ли эта емкость рассчитываться исходя из возможного количества объектов на экране, что приведет к формуле

$$H = (S_0 / S) \log K, \quad (5)$$

где K – размер словаря (количество возможных различных объектов), или только из количества позиций (N), заполненных значимыми символами:

$$H = N \log K. \quad (6)$$

Емкость H , определенная по (6), будет в нашем исследовании называться «эффективной информационной емкостью».

Мы также полагаем, что ограниченное применение теории вычислительной сложности в человеко-машинном взаимодействии объясняется существенным разнообразием пользователей даже внутри одной целевой группы. Мы исследуем субъективную сторону информационной сложности, анализируя факт узнаваемости (т. е. знакомой информации в отличие от незнакомой) в обработке информации человеком.

СКОРОСТЬ ОБРАБОТКИ ИНФОРМАЦИИ И УЗНАВАЕМОСТЬ

Узнаваемость довольно активно изучается в таких сферах, как маркетинг (узнаваемость продуктов и брендов), социология, психология, медицина и др. Хотя в сфере проектирования взаимодействия исследования узнаваемости начались задолго до появления Интернета [22], в настоящее время количество работ по этой тематике не очень велико. Можно отметить исследование влияния степени знакомства пользователя с сайтом на субъективное ощущение приватности [23], в ходе которого авторы описывают когнитивную и эмоциональную функции узнаваемости. В работах, имеющих большее отношение к интерфейсам, авторы используют понятие когнитивной дистанции, интересное как уже количественное выражение узнаваемости [24].

Сложность и узнаваемость в пользовательских интерфейсах явно имеют несколько уровней, однако общепринятая структура пока

не сформировалась. Например, мы можем определить эти уровни как *Задача (функционал) – Тезаурус – Расположение – Материал – Поведение*. Мы рассматриваем уровень *материала*, изучая эффект узнаваемости элементов интерфейса (этот уровень соответствует иконкам, кнопкам и прочим управляющим элементам).

Человек, как обработчик информации, может вводить (получать) и выводить информацию, причем для обоих процессов могут быть введены меры скорости (R_{in} и R_{out}), как производные от объемов информации по времени. Конечно, абсолютные значения скоростей зависят от единиц измерения, и это вносит определенную путаницу в сравнение оценок скоростей, полученных в различных исследованиях. Как бы там ни было, источники соглашаются, что верхний предел скорости сознательной обработки информации у человека составляет $R_{in} < 102$ бит/с, хотя его нервная система может обрабатывать до $R_{in} = 109$ бит/с подсознательно (этот факт, вероятно, служит основой для популярной идеи об огромных скрытых резервах человеческого мозга).

Что касается вывода информации, считается, что максимальная скорость работы голосовых связок человека на продолжительном отрезке времени соответствует $R_{out} = 5$ слогов/с, что соответствует $R_{out} = 12,5$ звуков/с для английского языка (для других языков это значение может отличаться). Для случая, когда человек использует какой-либо инструмент, значения скорости вывода обычно представляются в битах или символах в секунду. Для рукописного текста (авторучка) верхняя граница равна примерно 3,5 символа/с, средняя скорость печати составляет около 6,7 символов/с, а ее рекордные значения достигают 10 символов/с, что все-таки уступает скорости устной речи. Максимальная скорость выдачи информации человеком составляет 40 бит/с [18].

Значения скорости сознательного приема информации, хотя и выше, но не принципиально. R_{in} для восприятия речи на слух оценивается в 3...4 слова/с, для внимательного чтения с пониманием – 2,5 слова/с, или 18 бит/с, для чтения вслух – 30 бит/с, для скорочтения (на русском языке) – до 11,7 слов/с, или 80 бит/с. Большинство исследователей соглашаются, что, хотя общая R_{in} для чтения

может достигать 45 бит/с, возможности визуального восприятия человеком в длительном периоде скорее соответствуют лишь 8 бит/с [18].

Скорость может быть увеличена до определенной степени посредством использования многомерных сигналов вместо простых одномерных: например, тон и громкость вместо просто громкости или же размер, яркость и цвет вместо лишь одного изменяемого визуального стимула. Но эффект не является аддитивным. Вообще, способность хорошо справляться со сложными сигналами показывает, что человек по сравнению с машиной намного эффективнее в распознавании образов. В одном из наших предыдущих экспериментов мы обнаружили, что чем выше разнообразие объектов в задачах по их поиску и выбору, тем большую производительность (TPS) показывали участники: от 42 бит/с для группы объектов с размером словаря = 2 (простейшие задачи) до 78 бит/с для размера словаря = 29 (буквы русского алфавита) [25]. Следует заметить, что скорость распознавания для знакомых букв и цифр была ранее оценена на уровне 55 бит/с [18].

Известно, что еще одним фактором, повышающим скорость, является тренировка, которая, в частности, и позволяет опытным операторам воспринимать информацию на более высоком уровне, имея возможность обрабатывать меньшее количество единиц информации. Например, в то время как пилотам-новичкам приходится проверять каждое из показаний приборов индивидуально, опытным требуется лишь общий взгляд на приборную панель, чтобы понять, все ли в порядке с объектом управления (летательным аппаратом). Для относительно простого случая, когда имеет место поток входной текстовой информации, может использоваться коэффициент плотности новой информации [18].

Таким образом, можно сделать вывод о целесообразности учета узнаваемости информации для получателя, причем этот аспект является качественным и субъективным, а таковые обычно не очень хорошо учитываются моделями, основанными на теории информации, например, законом Хика. Далее мы исследуем влияние узнаваемости в специально спроектированном эксперименте и описываем некоторые примечательные результаты.

ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ

Участники эксперимента

Наше экспериментальное исследование проводилось два года в течение нескольких сессий, а участниками были студенты магистратуры «Прикладная информатика в веб-дизайне» факультета бизнеса Новосибирского государственного технического университета. Всего в эксперименте принимали участие 33 человека (14 женщин и 19 мужчин). Их возраст находился в диапазоне от 20 до 42 лет (среднее значение 24,5 года, СКО = 5,21). Все участники имели нормальное или скорректированное к нормальному зрение и большой опыт в использовании компьютеров и мыши. Перед экспериментом мы собирали данные о поле и возрасте участников. Все студенты участвовали в эксперименте добровольно и предварительно выразили свое согласие. Каждый участник мог сделать несколько тестовых попыток, результаты которых не записывались, для того чтобы он освоился с предлагаемыми заданиями и чтобы влияние обучения по ходу эксперимента было по возможности нивелировано. Сессии проходили с тремя различными группами участников с интервалом в один год, но на таком же или аналогичном компьютерном оборудовании: мыши, размерах и разрешении экрана.

План эксперимента и процедура проведения

В соответствующей части эксперимента (более полное описание всего эксперимента можно найти в работе [25]) участникам давали задания по выбору объектов. Им показывалась случайным образом расположенная стартовая кнопка (чтобы нивелировать эффект направления движения), примыкающая к рабочей зоне с меняющимся размером S_0 (256×256 , 512×320 , 512×512 или 1024×640 пикселей), в которой появлялось N (8, 32 или 128) объектов различного размера W (16, 32 или 64 пикселей).

Участники должны были кликнуть стартовую позицию указателем мыши, а затем «так быстро и точно, как только возможно» найти цель, подвести к ней указатель мыши и сделать клик (записывались координаты обоих кликов). Если второй клик был вне искомой цели, записывалась ошибка: участник переходил к следующему испытанию, но если общий уровень ошибок начинал превышать 10 %, ему показывалось сообщение с просьбой повысить точ-

ность выполнения заданий. Зависимыми переменными были время выбора (ST , между двумя кликами) и уровень ошибок (E).

Для измерения и записи значений независимых и зависимых переменных мы разработали веб-приложение с использованием PHP и MySQL. Участники использовали два браузера – Chrome и Firefox, для нивелирования возможного эффекта представления, а время измерялось при помощи JavaScript, чтобы избежать задержек на стороне сервера. Хотя этот метод измерения имеет известные недостатки, мы все же сочли его адекватным для нашего «пилотного» исследования.

Целевые объекты

Поскольку задачи исследования заключались в изучении влияния словаря и узнаваемости, мы использовали целевые объекты различных типов: геометрические фигуры, цифры и буквы кириллического алфавита. Размеры словаря K , соответствующие этим типам, равнялись 2, 10 и 29 (мы исключили несколько кириллических букв, очень похожих на другие).

Так как русский язык был родным почти для всех участников, чтобы отразить узнаваемость кириллических букв, мы использовали частоты, базирующиеся на статистике Национального корпуса русского языка [26] (с поправкой на частоты 4 букв, исключенных из эксперимента) – табл. 5.1.1.

Таблица 5.1.1

Статистика частот для 29 букв в русском языке.

Ранг	Буква	Частота	Ранг	Буква	Частота
1	О	11,2 %	10	Л	4,4 %
2	Е	8,6 %	11	К	3,5 %
3	А	8,1 %	12	М	3,3 %
4	И	7,5 %	13	Д	3,0 %
5	Н	6,8 %	14	П	2,9 %
6	Т	6,4 %	15	У	2,7 %
7	С	5,6 %	16	Я	2,0 %
8	Р	4,8 %	17	Ы	1,9 %
9	В	4,6 %	18	Ь	1,8 %

Окончание табл. 5.1.1

Ранг	Буква	Частота	Ранг	Буква	Частота
19	Г	1,7 %	25	Ш	0,7 %
20	З	1,7 %	26	Ю	0,6 %
21	Б	1,6 %	27	Ц	0,5 %
22	Ч	1,5 %	28	Э	0,3 %
23	Х	1,0 %	29	Ф	0,3 %
24	Ж	1,0 %			

Для цифр частоты соответствовали бы равномерному распределению (т. е. 10 % для каждой цифры), но это верно только для достаточно больших чисел. В нашем случае, однако, все объекты представляли собой единичную цифру, и мы решили основываться на формуле хорошо известного закона Бенфорда для первой цифры [27]. Мы также сочли, что участники в своей жизни чаще обращают внимание на начальную (первую слева) цифру любых чисел, поэтому узнаваемость должна лучше описываться законом Бенфорда, а не равномерным распределением. Итоговые частоты для цифр (ноль был исключен) показаны в табл. 5.1.2. Вместо десятичных дробей используются проценты.

Таблица 5.1.2

Предполагаемые частоты для цифр

Цифра	Частота
1	30,1 %
2	17,6 %
3	12,5 %
4	9,7 %
5	7,9 %
6	6,7 %
7	5,8 %
8	5,1 %
9	4,6 %

РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТА

В эксперименте были собраны данные для 6231 испытания, из которых 5832 (93,6 %) сочтены валидными (исключались исходы, когда участники сделали ошибочный двойной клик в самом начале или когда ST превосходило 15 000 мс). Табл. 5.1.3 содержит средние значения времени выбора (ST) и уровней ошибок в зависимости от количества целей (N) и размеров словаря (K).

На первом этапе была построена регрессионная модель для классического закона Хика, которая оказалась значимой ($F_{1,5830} = 1494$; $p < 0,001$), но имела довольно низкий коэффициент $R^2 = 0,204$ и труднообъяснимый отрицательный свободный член:

$$ST = -691 + 616 \log N. \quad (7)$$

Таблица 5.1.3

Средние ST (мс) и E (%) в зависимости от N и K

N	K			Средние ST , E (%)
	2	10	29	
8	964 6,1	1477 5,4	1456 3,9	1298 5,1
32	1043 4,2	2485 3,0	2710 3,7	2080 3,6
128	1411 4,7	5003 7,3	5570 8,2	3853 6,7
Средние ST , E (%)	1107 5,1	2688 5,0	2839 4,9	2196 5,0

Попробовав построить регрессионные модели с несколькими факторами, отражающими сложность интерфейса, мы нашли, что «эффективный объем интерфейса», выраженный как количество альтернатив, умноженное на логарифм словаря, дал наилучшее объяснение вариации зависимой переменной:

$$ST = 1123 + 7,92N \log K. \quad (8)$$

Эта модель показала высокую значимость ($F_{1,5830} = 4160$; $p < 0,001$) и имела довольно высокий коэффициент $R^2 = 0,416$, сравнимый с $R^2 = 0,666$ в модели для закона Фиттса (для задач по движению), которую построили, исходя из данных контрольной части эксперимента.

В соответствии с целью исследования мы добавили фактор частоты в модель. Цифры и буквы рассматривались по отдельности, поскольку их частоты (F_D и F_L соответственно, выраженные в процентах) имеют смысл только относительно символов из своего же набора:

$$ST_D = 1691 + 8,7 N \log K - 25,3 F_D, \quad (9)$$

$$ST_L = 1469 + 6,89 N \log K - 27,2 F_L. \quad (10)$$

Модель (9) показала высокую значимость ($F_{2,1734} = 453$; $p < 0,001$) с коэффициентом $R^2 = 0,343$, причем фактор F_D также был значим ($p < 0,001$). Модель (10) также показала высокую значимость ($F_{2,1900} = 587$; $p < 0,001$) с коэффициентом $R^2 = 0,382$, но фактор F_L был значим только на уровне $\alpha = 0,1$ ($p = 0,094$).

Мы ввели понятие производительности для задач выбора (TPS), в которой место индекса сложности Фиттса занимает фактор «эффективной сложности интерфейса»:

$$TPS = \frac{1}{Z} \sum_{i=1}^Z \left(\frac{1}{M} \sum_{j=1}^M \frac{N \log K_{ij}}{ST_{ij}} \right). \quad (11)$$

Ранее мы обосновали валидность понятия TPS [25] и обнаружили, что K существенно влиял на TPS (в дисперсионном анализе $F_{2,5829} = 33,1$; $p < 0,001$). Средние значения TPS для $K = 2, 10$ и 29 были равны соответственно $41,9, 58,6$ и $77,6$ бит/с. Мы заключили, что более высокие значения K вели к повышению TPS, поскольку более высокое разнообразие объектов означает их более высокую сложность. Между тем известно, что люди гораздо производительнее в распознавании сложных образов, чем в быстром и точном выполнении простых и типовых задач.

Для изучения влияния узнаваемости на TPS мы построили регрессионные модели для цифр и букв с соответствующими частотами в качестве факторов:

$$\text{TPS}_D = 50,1 + 0,725F_D, \quad (12)$$

$$\text{TPS}_L = 72,5 + 1,487F_L. \quad (13)$$

Модель (12) показала высокую значимость ($F_{1,1735} = 16,1$; $p < 0,001$), причем фактор F_D также был значим ($p < 0,001$), хотя R^2 оказался очень низким, равным 0,01. Модель (13) также показала высокую значимость ($F_{1,1901} = 5,33$; $p < 0,001$), фактор F_L был значим на уровне $\alpha = 0,05$ ($p = 0,021$), а R^2 оказался еще более низким, равным 0,003. Для сравнения, в нашей предыдущей работе [25] фактор возраста участников имел в некотором смысле влияние сравнимой силы на их TPS ($F_{1,5830} = 9,48$; $p = 0,002$; $R^2 = 0,002$).

ВЫВОДЫ

Управление информационной сложностью при проектировании интерфейсов в человеко-машинных системах остается важной, но пока не решенной задачей, поскольку отсутствуют общепринятые количественные метрики. Эффективным интерфейсом может считаться тот, в котором только релевантные свойства контролируемого объекта показываются оператору, а конечная цель – это обеспечить быстрое и безошибочное взаимодействие. Для этого информационная нагрузка на человека-оператора должна не максимизироваться, а оптимизироваться в соответствии с его возможностями по обработке информации. Тем не менее на настоящий момент неясно, каким образом следует измерять и оптимизировать информационную сложность в человеко-машинном взаимодействии, в частности вследствие ее субъективной природы.

В рамках нашего исследования когнитивного аспекта взаимодействия мы поставили эксперимент с 33 участниками (средний возраст 24,5 года) с целью определения перспективных способов формулировки количественной меры для информационных потоков в человеко-машинных системах. В частности, мы стремились оценить степень применимости закона Хика–Хаймана и прояснить по-

нятие «информационной емкости интерфейса». Мы также изучили влияние узнавания информации на субъективную сложность, сравнивая несколько существующих подходов и делая вывод об их неполной применимости в настоящем виде.

Результаты анализа позволяют предположить, что среди рассмотренных факторов именно «эффективная емкость интерфейса» в наибольшей степени определяла зафиксированное время выбора, в то время как закон Хика и модели, основанные на «длине интерфейса», явно ей уступали. В регрессионной модели с «эффективной емкостью интерфейса» $R^2 = 0,416$ оказался не очень высоким. Но необходимо отметить, что в контрольной части эксперимента (посвященной задачам по движению) модель по закону Фиттса имела в целом сравнимый $R^2 = 0,666$. Таким образом, в определенной степени невысокие значения R^2 могут быть объяснены несовершенством оборудования и измерительных средств.

Значения TPS (пропускной способности для задач выбора объектов), полученные в нашем эксперименте, а именно 42...78 бит/с, представляются адекватными, поскольку ранее скорость распознавания для знакомых букв и цифр оценивалась на уровне 55 бит/с. Следует отметить, что больший размер словаря K , отражающего разнообразие объектов, воспринимаемых участником, соответствовал более высоким значениям TPS. Действительно, известно, что особенностью человека является его способность распознавать сложные объекты и образы, а не быстро и точно выполнять простые и стандартные задачи.

Мы видим ограничения нашего исследования в его пилотном характере, что в частности привело к невысоким значениям R^2 . Мы также согласимся, что с точки зрения концептуальной валидности пока остаются вопросы, может ли частота появления цифр и букв в точности отражать их узнаваемость для участников. Тем не менее мы надеемся, что полученные нами результаты могут помочь в дальнейшем определении количественной меры информационной сложности и ее применении для оптимизации взаимодействия в человеко-машинных системах.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Internet World Stats. Europe Internet Usage Stats Facebook and Population Statistics. URL: <http://www.internetworldstats.com/stats4.htm>.
2. Internet Live Stats. Total number of websites. URL: <http://www.internetlivestats.com/total-number-of-websites/>.
3. Ханферян В. Система «все включено»: Границы между информационными технологиями и инжинирингом постепенно стираются / В. Ханферян, А. Ходырев // Эксперт. – № 17 (800). – 2012.
4. Грамматчиков А. Экспорт софта набирает силу // «Expert Online» 30 dec 2015. – Режим доступа: <http://expert.ru/2015/12/30/eksport-softa-nabiraet-silu/>.
5. Xing J. Measures of information complexity and the implications for automation design // No. DOT/FAA/AM-04/17. Fed. Aviation Administration Oklahoma City Ok Civil Aeromedical Inst, 2004.
6. McConnell, S. Code Complete (2nd ed.) // Microsoft Press, 2004. – 960 p.
7. Nielsen J. Usability Engineering // Morgan Kaufmann, San Francisco, USA, 1994. – 362 p.
8. Nielsen J. Jakob Nielsen's Alertbox, April 14, 2003: Paper Prototyping: Getting User Data Before You Code. URL: <https://www.nngroup.com/articles/paper-prototyping/>.
9. Грибова В.В. Автоматизация проектирования, реализации и сопровождения пользовательского интерфейса на основе онтологического подхода. – Владивосток: ИАПУ ДВО РАН, 2007. – 393 с.
10. Vanderdonckt J., Farenc C. (Eds.) // Proceedings of the Scientific Workshop on Tools for Working with Guidelines (TFWWG 2000). Springer, London, 2001. – 374 p.
11. Furtado E. et al. KnowiXML: A Knowledge-Based System Generating Multiple Abstract User Interfaces in UsiXML // In: Proc. of 3rd Int. Workshop on Task Models and Diagrams for user interface design TAMODIA'2004. ACM Press, New York, 2004. – P. 121–128.
12. Chevalier A., Fouquereau N., Vanderdonckt J. The Influence of a Knowledge-Based System on Designers' Cognitive Activities: a study involving Professional Web Designers // Behaviour & Information Technology, 28(1), 2009. – P. 45–62.
13. Dearden A., Finlay J. Pattern languages in HCI: A critical review // Human-Computer Interaction, 21(1), 2006. – P. 49–102.
14. Henninger S., Ashokkumar P. An Ontology-Based Infrastructure for Usability Design Patterns // In Proc. Semantic Web Enabled Software Engineering (SWESE), Galway, Ireland, 2005. – P. 41–55.

15. Gajos K.Z., Weld D.S., Wobbrock J.O. Automatically generating personalized user interfaces with SUPPLE // *Journal of Artificial Intelligence*, 174 (12-13), 2010. – P. 910–950.
16. Nichols J., Myers B.A. Automatically generating high-quality user interfaces for appliances // *Doctoral dissertation*, Carnegie Mellon University, Pittsburgh, PA, 2006. – 358 p.
17. Гасов В.М. Организация взаимодействия человека с техническими средствами АСУ (в 7 томах) / В.М. Гасов, Л.А. Соломонов. – М.: Высшая школа, 1990.
18. Присняков В.Ф. Математическое моделирование переработки информации оператором человеко-машинных систем / В.Ф. Присняков, Л.М. Приснякова. – М.: Машиностроение, 1990. – 248 с.
19. Seow S. Information theoretic models of HCI: A comparison of the Hick-Hyman law and Fitts' law // *Journal of Human-computer interaction*, vol. 20 (3), 2005. – Pp. 315–352.
20. Holzinger A. et al. On complexity reduction of user interfaces for safety-critical systems. *Multidisciplinary Research and Practice for Information Systems*, 2012. – Pp. 108–122.
21. Kumari U. and Upadhyaya S. An interface complexity measure for component-based software systems. *Int. J. of Com. Applications*, 2011, 36.1.
22. Somberg B.L. and Picardi M.C. Locus of the information familiarity effect in the search of computer menus. *Proc. of the Human Factors and Ergonomics Society*, 27 (9), SAGE, 1983.
23. Li Y. The impact of disposition to privacy, website reputation and website familiarity on information privacy concerns. *Decision Support Systems* 57, 2014. – Pp. 343–354.
24. Gough C., Green R., and Billingham M. Accounting for user familiarity in user interfaces. *Proc. of the 7th ACM SIGCHI New Zealand chapter's international conference on CHI*, 2006.
25. Bakaev M., Avdeenko T. A Quantitative Measure for Information Transfer in Human-Machine Control Systems. *Proc. IEEE Int. Siberian Conf. on Control and Communications (SIBCON)*, IEEE, 2015.
26. Wikipedia. Частотность (Frequency). – Режим доступа: <https://ru.wikipedia.org/wiki/Частотность>.
27. Wikipedia. Benford's law. – Режим доступа: https://en.wikipedia.org/wiki/Benford's_law.
28. Little J.D.C., Murty K.G., Sweeney D.W., and Karel C. An algorithm for the Traveling Salesman Problem // *Operations Research*. – 1963. – No. 11. – P. 972–989.
29. Беллман Р. Применение динамического программирования к задаче о коммивояжере // *Кибернетический сб. Вып. 9*. – М.: Мир, 1964. – С. 219–222.