

УДК 004.5

## ПРИНЦИПЫ РЕАЛИЗАЦИИ МНОГОКАНАЛЬНОГО ЧЕЛОВЕКО-МАШИННОГО ИНТЕРФЕЙСА

© 2022 г. Т. И. Возненко<sup>1,\*</sup>, А. И. Петрова<sup>1</sup>, К. Я. Кудрявцев<sup>1</sup>, Е. В. Чепин<sup>1</sup>

<sup>1</sup> Институт интеллектуальных кибернетических систем,  
Национальный исследовательский ядерный университет “МИФИ”, Москва, 115409, Россия

\*e-mail: TIVoznenko@mephi.ru

Поступила в редакцию 12.05.2022 г.

После доработки 13.05.2022 г.

Принята к публикации 14.05.2022 г.

В статье рассмотрены различные принципы реализации многоканального человеко-машинного интерфейса для управления робототехническим комплексом или киберфизической системой. В ходе распознавания команд, поступающих от оператора к интерфейсу, могут возникать ошибки первого и второго рода. Ошибка первого рода возникает, когда оператор отдал команду на исполнение, но она не была распознана. Ошибка второго рода возникает, когда оператор не отдавал команду на исполнение, но она была ошибочно распознана. В случае параллельного использования нескольких человеко-машинных интерфейсов возникает задача выбора команды, поскольку с различных интерфейсов могут приходиться противоречивые команды, которые требуют одних и тех же ресурсов для выполнения. Для решения этой задачи рассмотрено два основных принципа. Первый принцип заключается в реализации системы принятия решений, на вход которой поступают команды по каналам (интерфейсам) управления, а на выходе принимается решение о команде, которая будет выполнена. Второй принцип заключается в выборе одного наиболее эффективно работающего способа управления и использовании только его. В работе рассмотрены примеры реализации системы принятия решений и алгоритмов выбора одного способа управления. Для реализации данных алгоритмов рассмотрены различные критерии эффективности работы интерфейса. На основе сгенерированных матриц ошибок, проведен ряд экспериментов с целью определения количества ошибок при распознавании команд и вычисления ошибок первого и второго рода для каждого алгоритма.

**Ключевые слова:** человеко-машинный интерфейс, многоканальный интерфейс, робототехнический комплекс, киберфизическая система, система принятия решения, ошибки первого и второго рода

DOI: 10.56304/S2304487X22010126

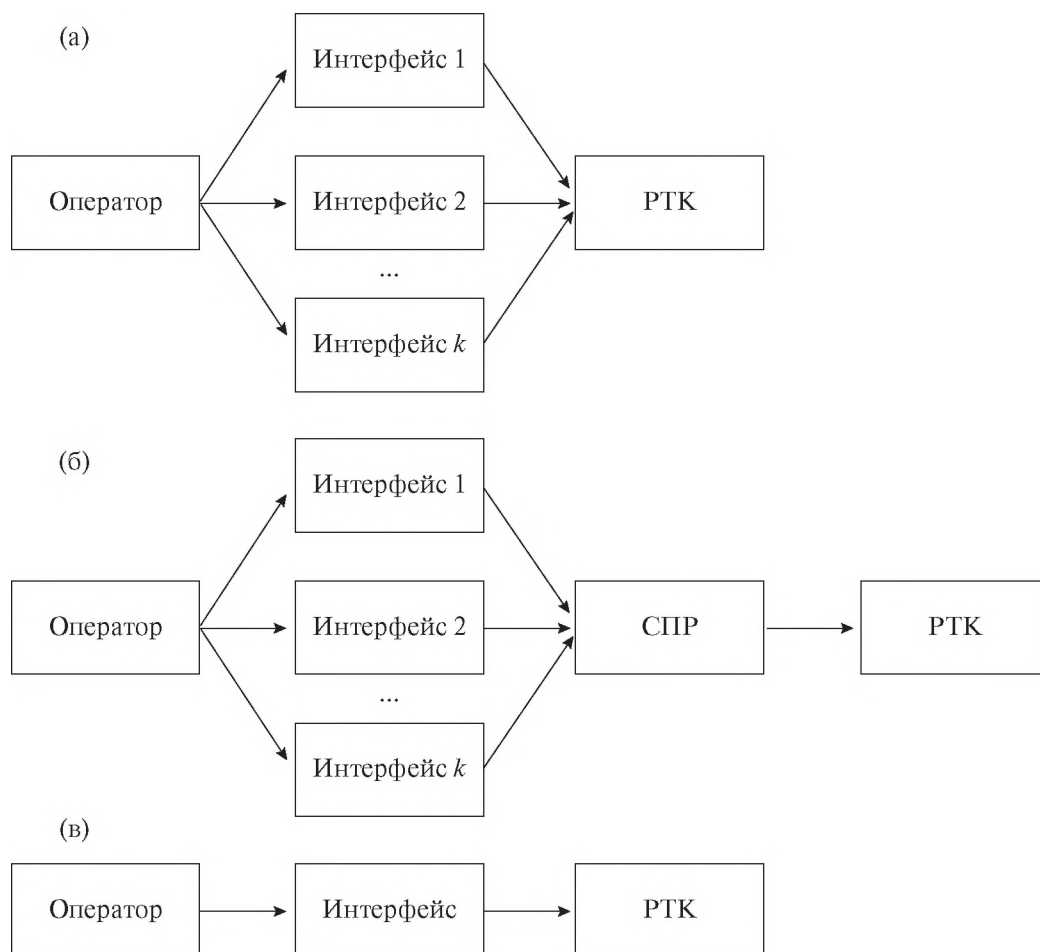
### 1. ВВЕДЕНИЕ

В настоящее время происходит активное развитие человеко-машинных интерфейсов. Под интерфейсом человеко-машинного взаимодействия понимается аппаратно-программный комплекс, позволяющий оператору управлять как робототехническими комплексами (РТК), так и киберфизическими системами (КФС). Среди таких интерфейсов можно отметить: голосовой интерфейс, жестовый интерфейс, нейрокомпьютерный интерфейс (Brain-Computer Interface, BCI), электроокулографический (ЭОГ) интерфейс, электромиографический (ЭМГ) интерфейс, датчики запаха, температуры, положения в пространстве, тактильные датчики и другие.

Каждый интерфейс человеко-машинного взаимодействия имеет свои особенности, достоинства и недостатки. Для обеспечения естественного взаимодействия человека с РТК и для использования преимуществ управления различных

интерфейсов с учетом их работы в различных условиях окружающей среды можно использовать несколько параллельно работающих интерфейсов. Однако при использовании нескольких параллельно работающих интерфейсов от оператора на РТК могут поступать различные команды из различных интерфейсов, конкурирующие за одни и те же ресурсы, необходимые для их выполнения (показано на рисунке 1а). В этом случае возникает необходимость выбора команды для выполнения, то есть необходимо решить то, какую команду нужно будет передать на РТК в данный момент времени. Для решения данной задачи можно использовать два принципиально разных принципа.

Первый принцип заключается в реализации системы принятия решений СПР, на вход которой подаются команды, распознанные с интерфейсов, а на выходе СПР команды передаются на выполнение РТК (показано на рисунке 1б). Обозначим этот принцип как “реализация СПР”.



**Рис. 1.** Схема взаимодействия оператора и РТК (а – с помощью нескольких интерфейсов, б – с помощью нескольких интерфейсов и СПР, в – с помощью одного интерфейса).

Второй принцип заключается в выборе одного наиболее эффективно работающего интерфейса и его использовании для управления РТК (показано на рисунке 1в). Данные с остальных интерфейсов игнорируются. Обозначим этот принцип как “выбор интерфейса управления”.

Оба этих принципа основаны на статистической оценке эффективности работы каждого интерфейса. Ведь для того, чтобы разработать СПР для выбора команды, и для того, чтобы выбрать один наиболее эффективно работающий интерфейс, необходимо провести статистическую оценку работы интерфейсов и определить критерий эффективности работы интерфейса. В данной статье рассматриваются и сравниваются реализации данных принципов.

## 2. СТАТИСТИЧЕСКАЯ ОЦЕНКА ЭФФЕКТИВНОСТИ РАБОТЫ ИНТЕРФЕЙСОВ

Для начала рассмотрим один канал передачи данных, то есть интерфейс РТК. Пусть для управ-

ления РТК имеется  $n$  команд:  $C = (c_1, c_2, \dots, c_n)$ . В ходе распознавания команд могут возникать ошибки первого и второго рода. Ошибка первого рода возникает, когда оператор отдает команду на исполнение, но она по каким-либо причинам не распознается. Ошибка второго рода возникает, когда оператор не отдавал команду на исполнение, но она по каким-либо причинам (внешние факторы, ошибка в работе интерпретатора команд) была распознана. Стоит отметить, что чем сложнее технология получения и интерпретации команд от оператора, тем больше будет вероятность того, что будут происходить ошибки первого и второго рода. Для оценки количества ошибок первого и второго рода можно использовать матрицу ошибок [5]. Для этого проводится тестирование, в ходе которого тестовая программа просит оператора вызывать различные команды на исполнение, в процессе фиксирует количество правильных распознаваний команд (когда команда была распознана, когда оператор вызвал ее на исполнение) и количество ошибок 1-го и 2-го рода. Существует несколько видов матриц оши-

бок. Например, в работе [6] матрица ошибок имеет размер  $n \times n$ . А в работе [7] размер матрицы ошибок  $(n + 1) \times n$ , где  $(n + 1)$ -й столбец это случаи когда команда не была распознана вовсе. В данной работе мы будем использовать матрицу размера  $(n + 1) \times n$ , пример которой показан в таблице 1. В ней  $m_{i\_j}$  это случаи когда оператор вызвал команду  $c_i$ , а была распознана команда  $c_j$ . Например, в данной матрице ошибок  $m_{2\_1}$  это количество случаев, когда оператор вызывал команду  $c_2$ , но была распознана команда  $c_1$ .

Перейдем от одного интерфейса к нескольким. Пусть у нас есть  $k$  интерфейсов:  $P = (p_1, p_2, \dots, p_k)$ . Если при вызове команды из одного канала нет случаев распознавания с другого канала и наоборот, то данные каналы являются независимыми, то есть результат работы одного канала никаким образом не влияет на другой канал. В случае независимых интерфейсов можно рассматривать каждую матрицу ошибок для каждого канала управления отдельно. В данной статье мы будем рассматривать случаи независимых каналов, и сравнивать различные реализации СПР и выбора интерфейса между собой. Но для того, чтобы проводить сравнение различных принципов, необходимо ввести критерий эффективности работы интерфейса.

### 3. КРИТЕРИЙ ЭФФЕКТИВНОСТИ РАБОТЫ ИНТЕРФЕЙСА

Рассмотрим матрицу ошибок для одного канала управления (таблица 1). В данной матрице для каждой команды можно вычислить стандартные метрики для оценки качества в задачах классификации: true positive (TP), true negative (TN), false positive (FP), false negative (FN). Для оценки эффективности работы интерфейса на практике существует множество метрик:

- recall (True Positive Rate, TPR) — для оценки ошибок первого рода;
- precision (Positive Predictive Value, PPV) — для оценки ошибок второго рода;
- accuracy (ACC) — среднее арифметическое между recall и precision;
- G-мера — среднее геометрическое между recall и precision;
- F1-мера — среднее гармоническое между recall и precision;
- и др.

Выбор метрики зависит как от самой выборки (является ли она сбалансированной или нет), так и от того, избегание ошибки какого рода (первого или второго) является предпочтительным. Идея выбора критерия эффективности состоит в том, чтобы выбрать одну или несколько метрик для

**Таблица 1.** Пример матрицы ошибок для одного интерфейса и  $n$  команд

| Отданная команда | Результат распознавания |            |     |            |                       |
|------------------|-------------------------|------------|-----|------------|-----------------------|
|                  | $c_1$                   | $c_2$      | ... | $c_n$      | Команда не распознана |
| $c_1$            | $m_{1\_1}$              | $m_{1\_2}$ | ... | $m_{1\_n}$ | $m_{1\_n+1}$          |
| $c_2$            | $m_{2\_1}$              | $m_{2\_2}$ | ... | $m_{2\_n}$ | $m_{2\_n+1}$          |
| ...              | ...                     | ...        | ... | ...        | ...                   |
| $c_n$            | $m_{n\_1}$              | $m_{n\_2}$ | ... | $m_{n\_n}$ | $m_{n\_n+1}$          |

оценивания эффективности работы канала управления и затем выбрать команду с максимальным или минимальным значением, вычисленным на основе этой метрики. Например, можно выбирать команду с наибольшим значением ТР или с наименьшим значением FP. Рассмотрим примеры реализации СПР и выбора интерфейса управления.

Значения  $m_{ij}$  в таблице 1 означают количество случаев распознавания команды  $c_j$ , при вызове команды  $c_i$ . Если в процессе составления данной матрицы все команды были вызваны одинаковое количество раз, т.е. сумма элементов по каждой строке одинакова и равна  $N$ , то разделив каждый элемент на  $N$ , мы перейдем от случаев распознавания к условным вероятностям. Для реализации СПР существует множество подходов, например, можно использовать данные вероятности для вычисления приоритета вызова команды [1], можно использовать алгоритм выбора команды на основе критерия Байеса [2], также можно выбрать одну из вышеперечисленных метрик и сводить все к нахождению экстремума функции. В рамках данной работы мы рассмотрим подход, связанный с критерием Байеса и максимизацией метрики F1-мера и метрики MTnP ( $MTnP = TPR * PPV$ ) [3]. Под максимизацией метрики F1-мера подразумевается вычисление значения метрики F1-мера для отданных команд, полученной на основе сбора статистических данных и выполнение команды с наибольшим значением F1-меры. Максимизация метрики MTnP подразумевает тот же подход, но с метрикой MTnP. Подход, основанный на критерии Байеса, заключается в выборе гипотезы (команды), имеющей наибольшее значение произведения условных вероятностей выполнения команды. Подробное описание данного подхода находится в статье [2].

В случае выбора одного интерфейса управления идея выбора метрики для оценки эффективности канала управления остается той же: необходимо выбрать метрику, рассчитать ее значение для каждой команды и каждого интерфейса и выбрать интерфейс с наибольшим или наименьшим

**Таблица 2.** Пример матрицы ошибок для голосового интерфейса и 5 команд и Base = 90

| Вызванная команда | Результат распознавания |       |        |         |      |                       |
|-------------------|-------------------------|-------|--------|---------|------|-----------------------|
|                   | Вперед                  | Назад | Налево | Направо | Стоп | Команда не распознана |
| Вперед            | 92                      | 4     | 0      | 2       | 0    | 2                     |
| Назад             | 3                       | 90    | 1      | 5       | 1    | 0                     |
| Налево            | 2                       | 2     | 94     | 2       | 0    | 0                     |
| Направо           | 1                       | 1     | 0      | 95      | 3    | 0                     |
| Стоп              | 2                       | 4     | 1      | 1       | 90   | 2                     |

значением метрики. Также вместо выбора наилучшего интерфейса можно выбрать наилучшую комбинацию канала-команд (то есть команд с различных интерфейсов) и использовать данные с остальных канал-команд для улучшения эффективности работы системы [4]. В данной работе рассмотрены подходы: выбор наилучшего интерфейса на основе F1-меры, выбор наилучшей комбинации канал-команд на основе F1-меры и MTnP, и алгоритм интерпретации. Выбор наилучшего интерфейса на основе F1-меры означает вычисление F1-меры для всех интерфейсов и выбор интерфейса с наибольшим значением F1-меры. Выбор наилучшей комбинации канал-команд на основе F1-меры означает вычисление метрики F1-меры для всех команд во всех каналах и выбор каждой команды с того интерфейса, для которого значение метрик F1-меры будет наибольшим. Выбор наилучшей комбинации канал-команд на основе метрики MTnP подразумевает тот же подход, но с метрикой MTnP. Алгоритм интерпретации заключается в выборе наилучшей комбинации канал-команд, и интерпретации неиспользуемых канал-команд как используемых, для улучшения эффективности работы канала. Алгоритм интерпретации подробно расписан в статье [4].

#### 4. СХЕМА ЭКСПЕРИМЕНТА

Тестирование заключается в моделировании вызовов 5 команд (движение вперед, назад, налево, направо, и стоп) с 3 интерфейсов (жестовый, голосовой и нейрокомпьютерный интерфейс). Эксперимент состоит из следующих шагов:

1. Выбор Base [4] значения (85, 90, или 95).
2. Генерирование матрицы ошибок для всех интерфейсов на основе Base значения.
3. Генерирование случайной команды/команд, которые оператор будет отдавать на выполнение.
4. Вычисление ожидаемой команды с учетом работы алгоритмов СПР и выбора интерфейса управления.
5. Вычисление полученной команды с учетом сгенерированной матрицы ошибок (распозна-

ной команды) и с учетом работы алгоритмов СПР и выбора интерфейса управления.

6. Сравнение ожидаемой и полученной команды.

7. Шаг 3–6 выполняется для 10000 вызовов случайных команд.

8. Шаг 2–7 выполняется для 1000 случайных матриц ошибок, сгенерированных согласно выбранному Base значению.

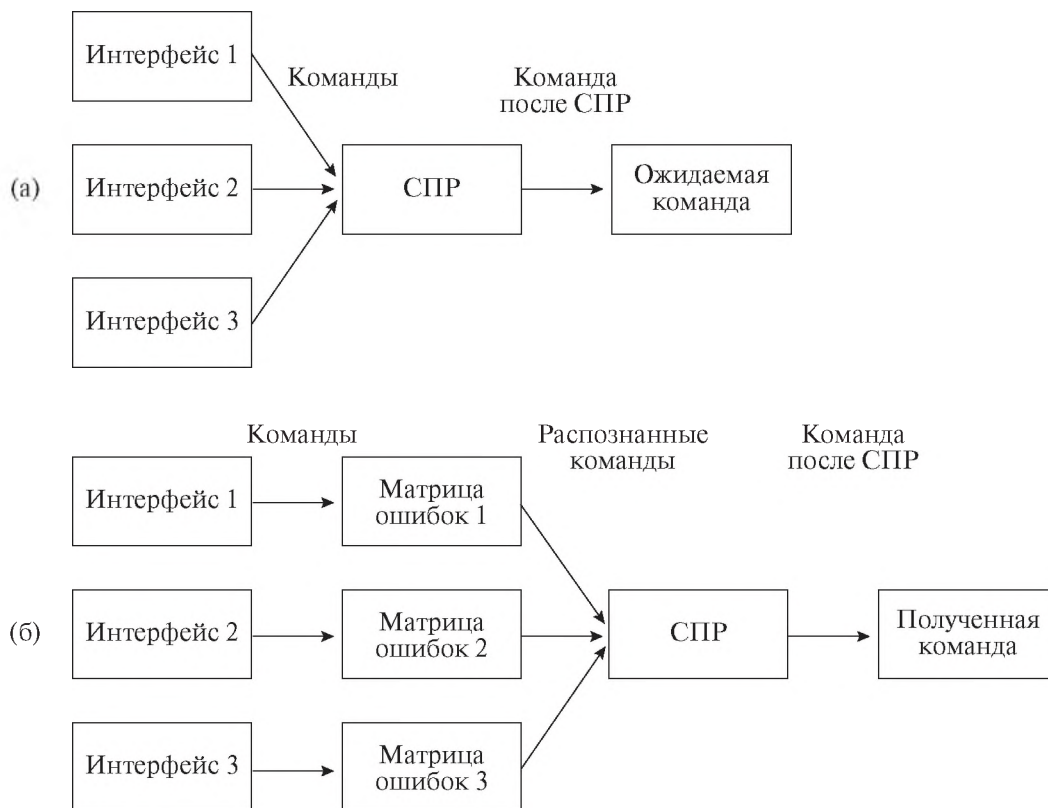
9. Шаг 1–8 выполняется для трех значений Base.

Рассмотрим более подробно каждый шаг.

В шагах 1 и 2 матрица ошибок составлена с учетом того, что каждая команда вызывалась 100 раз, т.е. сумма элементов матрицы по каждой строке равна 100. Генерирование матрицы ошибок проводилось с тремя Base [4] значениями (85, 90, 95). Значение Base означает минимальное значение TP для каждой команды, а оставшиеся (100 – Base) случайным образом распределялись между случаями правильного распознавания и ошибками первого и второго рода. Пример матрицы ошибок для Base = 90 представлен в таблице 2. То есть согласно этой матрице, если оператор вызывает команду вперед, то распознается:

- с 92% вероятностью команда вперед;
- с 4% вероятностью команда назад;
- с 2% вероятностью команда направо;
- с 2% вероятностью команда не будет распознана.

Далее в шагах 3–6, для тестирования работы алгоритмов СПР с одного или нескольких интерфейсов начинают поступать случайные команды. Причем с одного интерфейса может поступить максимум одна команда, то есть с трех интерфейсов может прийти одновременно максимум три команды. В шаге 4 случайные команды подаются на вход алгоритмов СПР, и на выходе СПР получается ожидаемая команда (рисунок 2а). В шаге 5 находится полученная команда следующим образом. Те же команды, что и на шаге 4, подаются на вход матрицы ошибок, и на выходе получаются команды с учетом вероятностей распознавания, то есть распознанные команды. И уже эти распознанные команды подаются на вход алгоритмов СПР, и на выходе СПР – полученная команда



**Рис. 2.** Схема эксперимента для оценки работы алгоритмов СПР (а — для вычисления ожидаемой команды, б — для вычисления полученной команды).

(рисунок 26). В данной статье рассматриваются алгоритмы на основе критерия Байеса (DMS Bayes), максимизации метрики F1-мера (DMS F) и метрики MTnP (DMS MTnP). Для оценки эффективности работы алгоритмов в шаге 6 рассчитываются три метрики: TP, FP, FN:

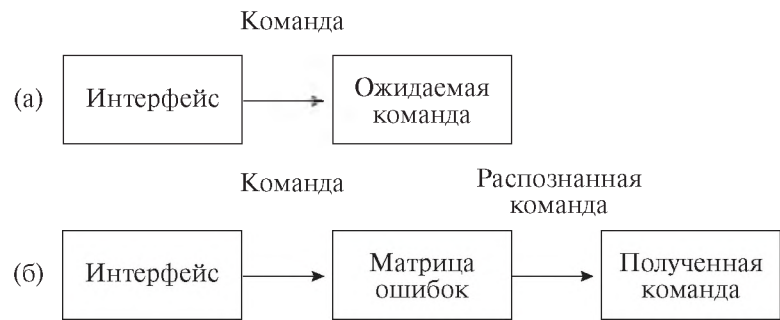
- TP — количество случаев, когда команда была отдана и была выполнена;
- FP — количество случаев, когда команда не была отдана, но она была выполнена (ошибка второго рода);
- FN — количество случаев, когда команда была отдана, но не была выполнена (ошибка первого рода).

В начале каждого эксперимента значения данных метрик приравниваются к нулю. Если ожидаемая команда и полученная команда совпадают, то мы увеличиваем значение метрики TP на 1. Если команда не была распознана вовсе, то есть полученная команда не распознана вообще, то увеличиваем FN на 1. Если же была распознана другая команда, то увеличиваем FP на 1.

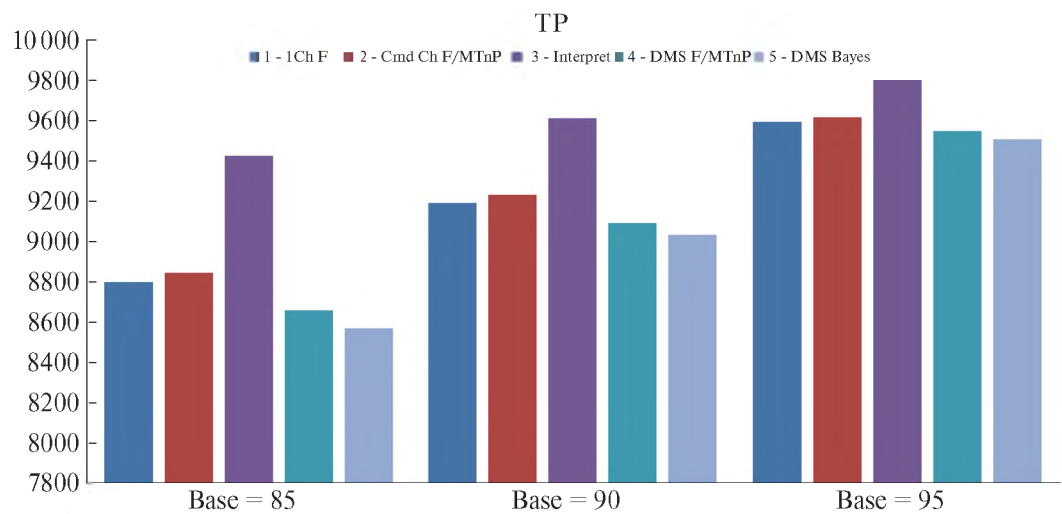
Рассмотрим шаги 3–6 для тестирования алгоритмов выбора интерфейса управления. Для выбора наиболее эффективно работающего интерфейса применяются 4 алгоритма: выбор одного

канала на основе F1-меры (1 Ch F), канал-команды на основе F1-меры (Cmd Ch F), канал-команды на основе MTnP (Cmd Ch MTnP), и алгоритм интерпретации (Interpret). Если в алгоритмах СПР использовались матрицы ошибок с трех каналов, то в случае выбора интерфейса используется только одна матрица ошибок (рисунок 3б). В случае выбора одного канала на основе F1-меры берется матрица ошибок канала с наибольшим минимальным значением F1-меры по всем командам для данного канала. В случае выбора канал-команд на основе F1-меры, и канал-команд на основе MTnP матрица ошибок формируется следующим образом. Строки матриц ошибок для используемых команд копируются в новую матрицу. После этого столбцы для неиспользуемых с данного канала команд суммируются со случаями, когда команда не была распознана, и убираются. Если рассмотреть пример из таблицы 2 и использовать только команды “Налево” и “Направо”, то получится таблица 3. Поскольку команды вперед и назад и стоп не вызываются с данного канала, то их случаи распознавания переходят в столбец “Команда не распознана”.

После формирования матрицы ошибок выполняются шаги 4–6. Выбирается одна случайная команда, которая является ожидаемой командой



**Рис. 3.** Схема эксперимента для оценки работы выбора интерфейса (а – для вычисления ожидаемой команды, б – для вычисления полученной команды).



**Рис. 4.** Значение TP для различных алгоритмов.

(рисунок 3а). И та же команда, что и на шаге 4, подается на вход матрицы ошибок, и на выходе получается команда с учетом вероятностей распознавания, то есть распознанная команда. Распознанная команда является итоговой полученной командой (рисунок 3б). Для оценки эффективности работы алгоритмов в шаге 6 рассчитываются метрики TP, FP, FN, по аналогии с шагом 6 для алгоритмов СПР.

**Таблица 3.** Пример матрицы ошибок для голосового интерфейса, в котором используются только команды “Налево” и “Направо”

| Вызванная команда | Результат распознавания |         |                       |
|-------------------|-------------------------|---------|-----------------------|
|                   | Налево                  | Направо | Команда не распознана |
| Налево            | 94                      | 2       | 4                     |
| Направо           | 0                       | 95      | 5                     |

5. ОБСУЖДЕНИЕ РЕЗУЛЬТАТОВ

Результаты тестирования представлены на рисунках 4–6. На данных рисунках указаны средние значения TP, FP и FN для алгоритмов:

- выбор одного канала на основе F1-меры (1 Ch F),
- канало-команды на основе F1-меры (Cmd Ch F),
- канало-команды на основе MTnP (Cmd Ch MTnP),
- алгоритма интерпретации (Interpret),
- алгоритма на основе критерия Байеса (DMS Bayes),
- максимизацией метрики F1-мера (DMS F),
- максимизацией метрики MTnP (DMS MTnP).

На данных графиках не указаны погрешности, поскольку доверительный интервал при коэффициенте Стьюдента равном 1.96 для всех значений был менее или около 10, при количестве изменений порядка 10000 для TP и 1000 для FP и FN.

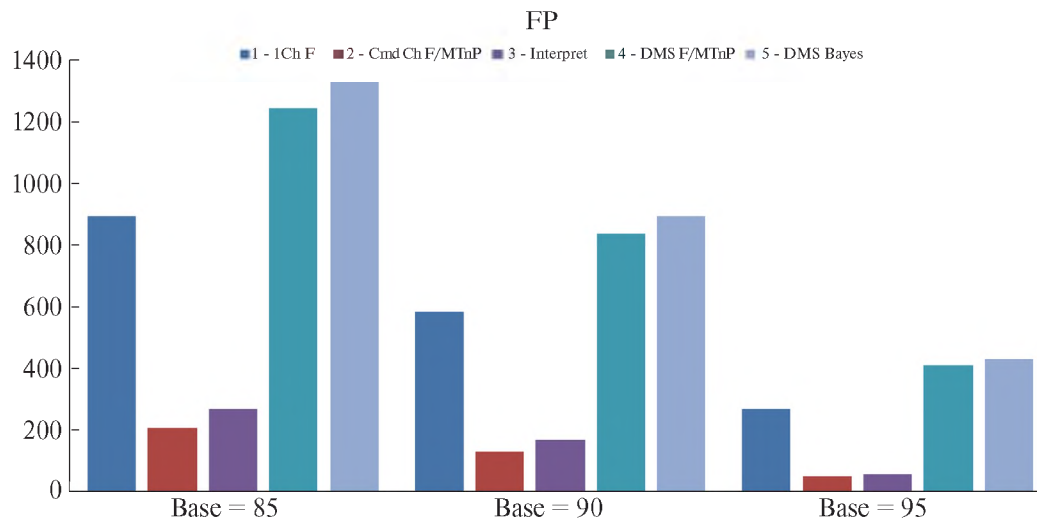


Рис. 5. Значение FP для различных алгоритмов.

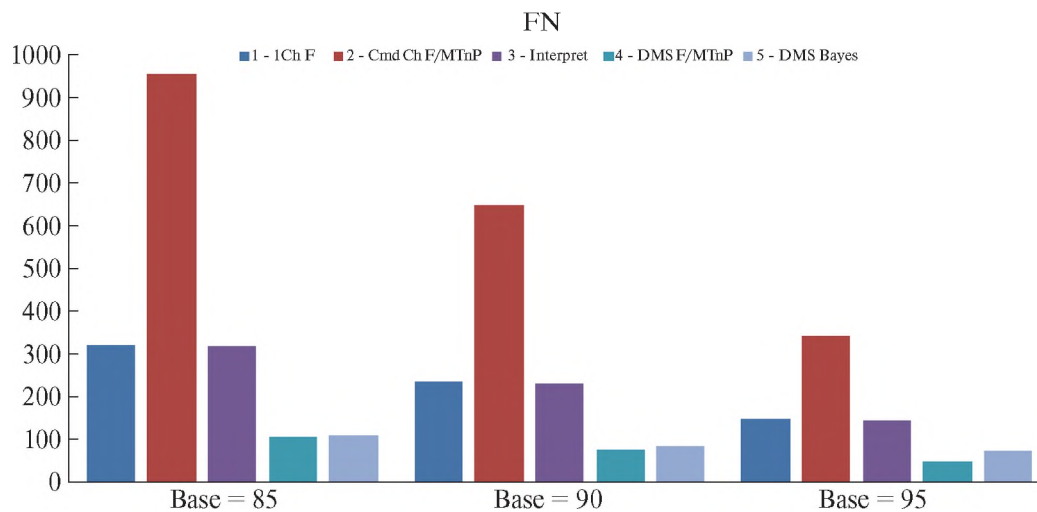


Рис. 6. Значение FN для различных алгоритмов.

В результате тестирования было обнаружено, что алгоритмы Cmd Ch F и Cmd Ch MTnP показали очень схожие результаты (были единичные случаи, где какой-то из алгоритмов проявил себя лучше), поэтому на графике результаты работы данных алгоритмов показаны в виде одной гистограммы Cmd Ch F/MTnP. Также схожие результаты были получены при алгоритмах DMS F и DMS MTnP, поэтому на графике они также показаны в виде одной гистограммы DMS F/MTnP.

У алгоритмов DMS Bayes и DMS F/MTnP наименьшее значение FN, однако много случаев FP. Полученный результат может быть объяснен тем, что производился вызов команд не с наибольшим значением условной вероятности распознавания

этой команды, а производился вызов всех команд.

Алгоритмы 1 Ch F и Cmd Ch F/MTnP показали неплохое значение TP, однако у алгоритма 1 Ch F высокое значение FP, а у алгоритмов Cmd Ch F/MTnP высокое значение FN. Алгоритм Interpret как и алгоритмы Cmd Ch F/MTnP имеет неплохое значение FP. Однако в отличие от Cmd Ch F/MTnP алгоритм Interpret рассматривает возможность интерпретации неиспользуемых команд. То есть при Cmd Ch F/MTnP если неиспользуемая команда была распознана, то это приводит к увеличению FN (так как такой команды нет в данном канале). А при Interpret возможны случаи интерпретации неиспользуемых команд



как используемых, что способствует снижению FN и увеличению TP.

## 6. ВЫВОДЫ

В данной статье были рассмотрены принципы реализации многоканального человеко-машинного интерфейса. Были рассмотрены примеры возможных реализаций системы принятия решений и алгоритмов выбора одного канала управления. Для данных реализаций было проведено тестирование и сравнение метрик TP, FP, FN. Выбор способа реализации системы принятия решений или выбора одного канала управления зависит от требований к РТК или КФС которой необходимо управлять, а также от интерфейсов и их устойчивости к внешним факторам. Поскольку, несмотря на то, что алгоритм интерпретации показал наибольшее значение TP, если возникнут внешние факторы, негативно влияющие на эффективность работы канала управления, то у оператора не будет возможности переключиться на другой способ управления. С другой стороны при использовании СПР, для управления задействованы не только команды с определенных каналов имеющих наибольшую эффективность распознавания, но и остальные. В будущих работах планируется исследовать различные факторы, влияющие на эффективность распознавания каналов, например показанные в работе [8, 9]. Также в будущем планируется рассмотреть случай зависимых каналов управления.

## СПИСОК ЛИТЕРАТУРЫ

1. Gridnev A.A., Voznenko T.I., Chepin E.V. The decision-making system for a multi-channel robotic device control // *Procedia computer science*, 2018. V. 123. P. 149–154. <https://doi.org/10.1016/j.procs.2018.01.024>
2. Voznenko T.I., Kudryavtsev K.Y. Algorithm for Multi-Channel Robotic Device Control Based on Minimizing Bayesian Risk // *Procedia Computer Science*, 2021. V. 190. P. 803–807. <https://doi.org/10.1016/j.procs.2021.06.093>
3. Voznenko T.I., Gridnev A.A., Kudryavtsev K.Y., Chepin E.V. The Decomposition Method of Multi-channel Control System Based on Extended BCI for a Robotic Wheelchair // *Advances in Intelligent Systems and Computing*, 2020. V. 948. P. 562–567. [https://doi.org/10.1007/978-3-030-25719-4\\_73](https://doi.org/10.1007/978-3-030-25719-4_73)
4. Voznenko T.I., Gridnev A.A., Chepin E.V., Kudryavtsev K.Y. The command interpretation in decomposition method of multi-channel control for a robotic device. // *Procedia Computer Science*, 2020. V. 169. P. 152–157. <https://doi.org/10.1016/j.procs.2020.02.127>
5. Stehman S.V. Selecting and interpreting measures of thematic classification accuracy. // *Remote Sensing of Environment*, 1997. V. 62. № 1. P. 77–89. [https://doi.org/10.1016/S0034-4257\(97\)00083-7](https://doi.org/10.1016/S0034-4257(97)00083-7)
6. Biswas K.K., Basu S.K. Gesture recognition using Microsoft Kinect® / In *The 5th international conference on automation, robotics and applications*, 2011. P. 100–103. <https://doi.org/10.1109/ICARA.2011.6144864>
7. Abdelnasser H., Youssef M., Harras K.A. Wigest: A ubiquitous wifi-based gesture recognition system. / In *2015 IEEE conference on computer communications (INFOCOM)*, 2015. P. 1472–1480. <https://doi.org/10.1109/INFOCOM.2015.7218525>
8. Voznenko T.I., Gridnev A.A., Chepin E.V., Kudryavtsev K.Y. Comparison Between Coordinated Control and Interpretation Methods for Multi-channel Control of a Mobile Robotic Device. // *Advances in Intelligent Systems and Computing*, 2021. V. 1310. P. 558–564. [https://doi.org/10.1007/978-3-030-65596-9\\_68](https://doi.org/10.1007/978-3-030-65596-9_68)
9. Petrova A., Chepin E., Voznenko T. Using Environmental Objects as Visual Stimuli in BCI-based Interaction System: Theoretical Approach. // *Procedia Computer Science*, 2021. V. 190. P. 670–677. <https://doi.org/10.1016/j.procs.2021.06.109>

---

**Vestnik Natsional'nogo Issledovatel'skogo Yadernogo Universiteta "MIFI", 2022, vol. 11, no. 1, pp. 59–67**

---

## The Implementation Principles of a Multi-Channel Human-Machine Interface

T. I. Voznenko<sup>a, #</sup>, A. I. Petrova<sup>a</sup>, K. Y. Kudryavtsev<sup>a</sup>, and E. V. Chepin<sup>a</sup>

<sup>a</sup> *Institute of Cyber Intelligence Systems, National Research Nuclear University MEPhI (Moscow Engineering Physics Institute), Moscow, 115409 Russia*

<sup>#</sup> *e-mail: TIVoznenko@mephi.ru*

Received May 12, 2022; revised May 13, 2022; accepted May 14, 2022

**Abstract**—Different implementation principles of a multi-channel human-machine control interface for robotic complex or cyber-physical system are discussed. When an operator executes a command via the interface, type-I and type-II errors may occur during command recognition. The type I error occurs when the operator executes the command, but it is not recognized. The type II error occurs when the command is recognized although the operator does not execute it. In case of using several human-machine interaction



channels, the problem of choosing the command to execute arises because of the presence of conflicting commands that can be recognized from different channels and in the same time require the same resources for execution. To solve this problem, two main principles have been determined. The first principle is to implement a decision-making system, which receives commands through control channels (interfaces), and makes a decision about the command to be executed. The second principle is to choose and use only the most efficient control method. Examples of implementation of decision-making system and algorithm of choosing one control method are presented. To implement these algorithms, various criteria of interface efficiency are considered. Based on the generated confusion matrices, a number of experiments have been carried out to determine the number of errors in command recognition and to calculate type-I and type-II errors for each algorithm.

**Keywords:** human-machine interface, multi-channel interface, robotic complex, cyber-physical system, decision-making system, type-I and type-II errors

DOI: 10.56304/S2304487X22010126

## REFERENCES

1. Gridnev A.A., Voznenko T.I., Chepin E.V. The decision-making system for a multi-channel robotic device control. *Procedia computer science*, 2018, vol. 123, pp. 149–154.  
doi: 10.1016/j.procs.2018.01.024
2. Voznenko T.I., Kudryavtsev K.Y. Algorithm for Multi-Channel Robotic Device Control Based on Minimizing Bayesian Risk. *Procedia Computer Science*, 2021, vol. 190, pp. 803–807.  
doi: 10.1016/j.procs.2021.06.093
3. Voznenko T.I., Gridnev A.A., Kudryavtsev K.Y., Chepin E.V. The Decomposition Method of Multi-channel Control System Based on Extended BCI for a Robotic Wheelchair. *Advances in Intelligent Systems and Computing*, 2020, vol. 948, pp. 562–567.  
doi: 10.1007/978-3-030-25719-4\_73
4. Voznenko T.I., Gridnev A.A., Chepin E.V., Kudryavtsev K.Y. The command interpretation in decomposition method of multi-channel control for a robotic device. *Procedia Computer Science*, 2020, vol. 169, pp. 152–157.  
doi: 10.1016/j.procs.2020.02.127
5. Stehman S.V. Selecting and interpreting measures of thematic classification accuracy. *Remote Sensing of Environment*, 1997, vol. 62, no. 1, pp. 77–89.  
doi: 10.1016/S0034-4257(97)00083-7
6. Biswas K.K., Basu S.K. Gesture recognition using Microsoft Kinect®. *In The 5th international conference on automation, robotics and applications*, 2011, pp. 100–103.  
doi: 10.1109/ICARA.2011.6144864
7. Abdelnasser H., Youssef M., Harras K.A. Wigest: A ubiquitous wifi-based gesture recognition system. *In 2015 IEEE conference on computer communications (INFOCOM)*, 2015, pp. 1472–1480.  
doi: 10.1109/INFOCOM.2015.7218525
8. Voznenko T.I., Gridnev A.A., Chepin E.V., Kudryavtsev K.Y. Comparison Between Coordinated Control and Interpretation Methods for Multi-channel Control of a Mobile Robotic Device. *Advances in Intelligent Systems and Computing*, 2021, vol. 1310, pp. 558–564.  
doi: 10.1007/978-3-030-65596-9\_68
9. Petrova A., Chepin, E., Voznenko T. Using Environmental Objects as Visual Stimuli in BCI-based Interaction System: Theoretical Approach. *Procedia Computer Science*, 2021, vol. 190, pp. 670–677.  
doi: 10.1016/j.procs.2021.06.109