ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ ҒЫЛЫМ ЖӘНЕ ЖОҒАРЫ БІЛІМ МИНИСТРЛІГІ
МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ КАЗАХСТАН
MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE REPUBLIC OF KAZAKHSTAN

ХАЛЫҚАРАЛЫҚ АҚПАРАТТЫҚ ЖӘНЕ КОММУНИКАЦИЯЛЫҚ ТЕХНОЛОГИЯЛАР ЖУРНАЛЫ

МЕЖДУНАРОДНЫЙ ЖУРНАЛ ИНФОРМАЦИОННЫХ И КОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ

INTERNATIONAL JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGIES

2025 (23) 3

шілде- қыркүйек

ISSN 2708–2032 (print)
ISSN 2708–2040 (online)

Халықаралық ақпараттық және коммуникациялық технологиялар журналы
ISSN 2708–2032 (print)
ISSN 2708–2040 (online)
Меншік иесі: АҚ «Халықаралық ақпараттық технологиялар университеті» (Алматы қ.).
Қазақстан Республикасы Ақпарат және қоғамдық даму министрлігіне мерзімді баспасөз басылымын есепке қою туралы куәлік № KZ82VPY00020475, 20.02.2020 ж. берілген
Тақырып бағыты: ақпараттық технологиялар, ақпараттық қауіпсіздік және коммуникациялық технологиялар, әлеуметтік-экономикалық жүйелерді дамытудағы цифрлық технология.
Мерзімділігі: жылына 4 рет.
Тираж: 100 дана.
Редакция мекенжайы: 050040 Алматы қ., Манас к., 34/1, каб. 709, тел: +7 (727) 244-51-09.
E-mail: ijict@iitu.edu.kz
Журнал сайты: https://journal.iitu.edu.kz

Международный журнал информационных и коммуникационных технологий
ISSN 2708–2032 (print)
ISSN 2708–2040 (online)
Собственник: АО «Международный университет информационных технологий» (г. Алматы).
Свидетельство о постановке на учет периодического печатного издания в Министерство информации и общественного развития Республики Казахстан **№ KZ82VPY00020475,** выданное от **20.02.2020 г.**
Тематическая направленность: информационные технологии, информационная безопасность и коммуникационные технологии, цифровые технологии в развитии социо-экономических систем**.**
Периодичность: 4 раза в год.
Тираж: 100 экземпляров.
Адрес редакции: 050040 г. Алматы, ул. Манаса 34/1, каб. 709, тел: +7 (727) 244-51-09.
E-mail: ijict@iitu.edu.kz Сайт журнала: https://journal.iitu.edu.kz

«International Journal of Information and Communication Technologies»
ISSN 2708–2032 (print)
ISSN 2708–2040 (online)
Owner: International Information Technology University JSC (Almaty).
The certificate of registration of a periodical printed publication in the Ministry of Information and Social Development of the Republic of Kazakhstan, Information Committee No. KZ82VPY00020475, issued on 20.02.2020.
Thematic focus: information technology, digital technologies in the development of socio-economic systems, information security and communication technologies
Periodicity: 4 times a year.
Circulation: 100 copies.
Editorial address: 050040. Manas st. 34/1, Almaty. +7 (727) 244-51-09. E-mail: ijict@iitu.edu.kz
Journal website: https://journal.iitu.edu.kz

МАЗМҰНЫ

СОДЕРЖАНИЕ

CONTENTS

INTERNATIONAL JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGIES
ISSN 2708–2032 (print)
ISSN 2708–2040 (online)
Vol. 6. Is. 3. Number 23 (2025). Pp. 238–253
Journal homepage: https://journal.iitu.edu.kz
https://doi.org/10.54309/IJICT.2025.23.3.014
UDC 004.942:519.2:656.7

PROBABILISTIC TIMED AUTOMATA ANALYSIS OF AIR TRAFFIC CONTROL SYSTEMS

K. Myrzabek*, A. Khassen, S. Khan
Astana IT University, Astana, Kazakhstan.
E-mail: kamilamyrzabek055@gmail.com

Myrzabek Kamila — student, Department of «Intelligent Systems and Cybersecurity», Astana IT University, Astana, Kazakhstan E-mail: kamilamyrzabek055@gmail.com, https://orcid.org/0009-0005-3273-1881;
Khassen Akbota — student, Department of «Intelligent Systems and Cybersecurity», Astana IT University, Astana, Kazakhstan https://orcid.org/0009-0006-7816-869X;
Khan Shazada — PhD, assistant-professor, Department of «Intelligent Systems and Cybersecurity», Astana IT University, Astana, Kazakhstan https://orcid.org/0000-0003-2702-0717.

Abstract. The research project unites intelligent air-traffic-management system development with the strict mathematical evaluation of control-action coordination interfaces between pilots and controllers. The controller–pilot interface must receive formal specifications just like flight-control software since it represents a unified reactive system. Every system interaction receives probabilistic timed automaton modeling because it represents both real-time operational constraints and the random occurrence of human errors and equipment breakdowns. The Z formal specification language enables writing contracts that link observable external actions to their required pre-conditions and post-conditions which humans and machines can verify. The system interface modules that represent human operators and graphical displays and underlying avionics logic and supervisory safety agents are connected to perform weak branching bisimulation which proves behavioral equivalence under timing uncertainty. The model demonstrates that the controller commands along with machine feedback and supervisory vetoes and pilot acknowledgements can be represented with twenty global states in a model that includes every significant timing and fault scenario. The automated verification system measures a low probability of mission success as well as operator slips that increase overall risk and identifies critical time-diver-

gent execution paths that reveal hidden design vulnerabilities. Interface engineers can modify timing guards and revise visual cues and embed cognitive-load adaptations without disrupting the existing formal proofs because the graphical display and hidden logic have been proven equivalent. The study transforms the controller-pilot interface into an air-traffic-management ecosystem component which receives mathematical certification and sets the stage for future interfaces that adapt to workload data while providing safety guarantees that machines can verify.

Keywords: probabilistic timed automata, human-machine interaction, interface verification, air traffic control, bisimulation

For citation: K. Myrzabek, A. Khassen, S. Khan. Probabilistic timed automata analysis of air traffic control systems//International journal of information and communication technologies. 2025. Vol. 6. No. 23. Pp. 238–253. (In Eng.). https://doi.org/10.54309/IJICT.2025.23.3.014.

Conflict of interest: The authors declare that there is no conflict of interest.

УАҚЫТТЫҚ ЫҚТИМАЛДЫ АВТОМАТТАР НЕГІЗІНДЕ ӘУЕ ҚОЗҒАЛЫСЫН БАСҚАРУ ЖҮЙЕЛЕРІН ТАЛДАУ

К.К. Мырзабек*, А.Б. Хасен, Ш.М.У. Хан
Астана ИТ университет, Астана, Казахстан.
E-mail: kamilamyrzabek055@gmail.com

Мырзабек Камила — «Интеллектуалды жүйелер және киберқауіпсіздік» студенті, Astana IT University
https://orcid.org/0009-0005-3273-1881;
Хасен Ақбота — «Интеллектуалды жүйелер және киберқауіпсіздік» студенті, Astana IT University
https://orcid.org/0009-0006-7816-869X;
Хан Шазада Мұхаммад Умаир — PhD, «Интеллектуалды жүйелер және киберқауіпсіздік» кафедрасының қауымдастырылған профессоры, Astana IT University
https://orcid.org/0000-0003-2702-0717.

Аннотация. Бұл зерттеу жобасы әуе қозғалысын басқарудың интеллектуалды жүйесін әзірлеуді ұшқыштар мен диспетчерлер арасындағы басқару әрекеттерін үйлестіру интерфейстерін қатаң математикалық бағалаумен біріктіреді. «Диспетчер–ұшқыш» интерфейсі бірыңғай реактивті жүйені бейнелейтіндіктен, ол ұшу басқару бағдарламалық жасақтамасы сияқты формалды спецификацияларға ие болуы тиіс. Жүйенің әрбір өзара әрекеттесуі нақты уақыттағы операциялық шектеулерді және адам қателері мен жабдықтың істен шығуының кездейсоқ туындауын бейнелейтін уақыттық

ықтималды автоматпен модельденеді. Z формалды спецификация тілі адамның да, машинаның да тексере алатын шарттар мен салдарларға байланысты бақыланатын сыртқы әрекеттерді сипаттауға мүмкіндік береді. Операторларды, графикалық дисплейлерді, ішкі авионикаы және қадағалау қауіпсіздік агенттерін бейнелейтін интерфейс модульдері уақыттық белгісіздік жағдайында мінез-құлық эквиваленттілігін дәлелдейтін әлсіз тармақталатын бисимуляцияны орындау үшін біріктіріледі. Модель диспетчер командаларының, машинаның кері байланысының, қадағалау вето шешімдерінің және ұшқыш растауларының барлық маңызды уақыттық және істен шығу сценарийлерін қамтитын жиырма жаһандық күйде көрсетілуі мүмкін екенін дәлелдейді. Автоматтандырылған верификация жүйесі миссияның сәтті аяқталу ықтималдығының төмендігін, жалпы тәуекелді арттыратын оператор қателерін тіркейді және жасырын жобалау осалдықтарын анықтайтын уақыттық дивергенттік орындау жолдарын айқындайды. Интерфейс инженерлері уақыттық шарттарды өзгерте алады, визуалды белгілерді қайта қарай алады және когнитивтік жүктемеге бейімделулерді енгізе алады, өйткені графикалық дисплей мен жасырын логиканың эквиваленттігі ресми түрде дәлелденген. Бұл зерттеу «диспетчер–ұшқыш» интерфейсін математикалық сертификаттауға ие болатын әуе қозғалысын басқару экожүйесінің құрамдас бөлігіне айналдырады және жұмыс жүктемесі деректеріне бейімделетін болашақ интерфейстердің негізін қалайды, сонымен қатар машиналар тексере алатын қауіпсіздік кепілдіктерін қамтамасыз етеді.

Түйін сөздер: уақыттық ықтималды автоматтар, адам–машина өзара әрекеттесуі, интерфейсті верификациялау, әуе қозғалысын басқару, бисимуляция

Дәйексөздер үшін: К.К. Мырзабек, А.Б. Хасен, Ш.М.У. Хан. Уақыттық ықтималды автоматтар негізінде әуе қозғалысын басқару жүйелерін талдау//Халықаралық ақпараттық және коммуникациялық технологиялар журналы. 2025. Том. 6. № 23. 238–253 бет. (Ағыл). https://doi.org/10.54309/IJICT.2025.23.3.014.

Мүдделер қақтығысы: Авторлар осы мақалада мүдделер қақтығысы жоқ деп мәлімдейді.

АНАЛИЗ СИСТЕМ УПРАВЛЕНИЯ ВОЗДУШНЫМ ДВИЖЕНИЕМ С ИСПОЛЬЗОВАНИЕМ ВЕРОЯТНОСТНЫХ АВТОМАТОВ С ВРЕМЕННЫМИ ОГРАНИЧЕНИЯМИ

К.К. Мырзабек**, А.Б. Хасен, Ш.М.У. Хан

Астана ИТ университет, Астана, Казахстан.

E-mail: kamilamyrzabek055@gmail.com

Мырзабек Камила — студентка, кафедра «Интеллектуальных систем и кибербезопасности», Астана ИТ университет, Астана Казахстан E-mail: kamilamyrzabek055@gmail.com, https://orcid.org/0009-0005-3273-1881;

Хасен Акбота — студентка, кафедра «Интеллектуальных систем и кибербезопасности», Астана ИТ университет, Астана Казахстан

https://orcid.org/0009-0006-7816-869X;
Хан Шазада Мухаммад Умаир — PhD, ассистент-профессор, кафедра «Интеллектуальных систем и кибербезопасности», Астана ИТ университет, Астана Казахстан https://orcid.org/0000-0003-2702-0717.

Аннотация. В представленом исследовательском проекте объединяются разработка интеллектуальной системы управления воздушным движением и строгая математическая оценка интерфейсов координации управляющих действий между пилотами и диспетчерами. Интерфейс «диспетчер–пилот» должен иметь формальные спецификации так же, как и программное обеспечение систем управления полётом, поскольку он представляет собой единый реактивный комплекс. Каждое взаимодействие в системе моделируется с помощью вероятностного автомата с временными ограничениями, что отражает как реальные эксплуатационные ограничения, так и случайное возникновение ошибок оператора и сбоев оборудования. Язык формальных спецификаций Z позволяет формировать контракты, связывающие наблюдаемые внешние действия с их необходимыми предусловиями и постусловиями, которые могут проверяться как человеком, так и машиной. Модули интерфейса, представляющие операторов, графические дисплеи, внутреннюю авионику и надзорных агентов безопасности, соединяются для выполнения слабой ветвящейся бисимуляции, которая доказывает поведенческую эквивалентность при наличии временной неопределённости. Модель показывает, что команды диспетчера, обратная связь от машины, надзорные вето и подтверждения пилота могут быть представлены двадцатью глобальными состояниями в модели, охватывающей все значимые временные и отказные сценарии. Автоматизированная система верификации фиксирует низкую вероятность успешного завершения миссии, а также ошибки оператора, повышающие общий риск, и выявляет критические временные дивергентные пути выполнения, обнаруживающие скрытые уязвимости проектирования. Инженеры интерфейсов могут изменять временные условия, пересматривать визуальные подсказки и внедрять адаптации к когнитивной нагрузке без нарушения существующих формальных доказательств, поскольку графический дисплей и скрытая логика были доказаны эквивалентными. Данное исследование трансформирует интерфейс «диспетчер–пилот» в компонент экосистемы управления воздушным движением, получающий математическую сертификацию и закладывающий основу для будущих интерфейсов, адаптирующихся к данным о рабочей нагрузке при сохранении гарантий безопасности, проверяемых машинами.

Ключевые слова: вероятностные автоматы с временными ограничениями, взаимодействие человек–машина, верификация интерфейсов, управление воздушным движением, бисимуляция

Для цитирования: К.К. Мырзабек, А.Б. Хасен, Ш.М.У. Хан. Анализ систем управления воздушным движением с использованием вероятностных автоматов с временными ограничениями//Международный журнал информационных и коммуникационных технологий. 2025. Т. 6. No. 23. Стр. 238–253. (На анг.). https://doi.org/10.54309/IJICT.2025.23.3.014.

Конфликт интересов: авторы заявляют об отсутствии конфликта интересов.

Introduction

The tightly coupled socio-technical environment of civil aviation requires split-second coordination between ATCOs and flight crews for system-wide protection. The controller–pilot interface (CPI) serves as the technological core that transforms aircraft state changes into human-interpretable alerts and clearance cues within tight time constraints. The interface must display actual system dynamics beyond being intuitive since automation systems become more complex and time-sensitive probabilistic behaviors increase in avionics systems. Such guarantees are typically absent from conventional human–machine interfaces especially during infrequent system faults and unexpected mode transitions (Zamarreño Suárez et al., 2024).

Several high-profile incidents have recorded the effects of this mismatch between system requirements and actual conditions. Ground controllers maintained precise trajectory information about the Germanwings 9525 disaster, but they lacked the ability to detect the automated mode that let the pilot disable safety limits before controllers could step in (Zamarreño Suárez et al., 2024). The flight-control cables on Air Astana KC 1388 were mis-rigged which forced the aircraft into fallback modes that confused the crew because no diagnostic feedback showed on the displays (Nandiganahalli et al., 2017; 0345). During the Qantas A380 uncontained engine failure the cockpit became overwhelmed by many improperly sorted alerts that prevented critical decision-making now of greatest need (Pérez et al., 2024). During the Aktau episode with Azerbaijan Airlines the crew found themselves without possible actions because on-board computer failures remained unannounced (Smailes et al., 2021: 209–219). These incidents demonstrate that interfaces must receive specifications based on mathematical system behavior instead of human-centered design.

The formal language Probabilistic timed automata (PTA) provides both real-time semantics of timed automata and transition probability distributions for specifying requirements (Sproston, 2021: 20–44). Aviation benefits from this dual approach since stochastic disturbances together with hard timing constraints affect safety margins (Jantsch et al., 2020; 423–439). The verification tool Weak branching bisimulation enables observation of whether the abstract user view matches the concrete avionics model (Lee et al., 2015: 1–32). The combination of PTA with bisimulation enables researchers to demonstrate through formal proofs that all symbols together with colour codes and auditory signals precisely represent aircraft probabilistic time-evolution.

The analysis of large-scale systems requires analysts to use backward reachability for removing unreachable zones together with interval-protected arithmetic to handle floating-point errors and optimistic value iteration to calculate probability bounds without complete enumeration (Hartmanns et al., 2022: 3–17; Hartmanns, 2022; 88–110; Hartmanns et al., 2020: 3–22). The development of quantitative bisimilarity metrics allows engineers to evaluate how well simplified interfaces match full systems during cases where exact equivalence is impossible (Lanotte et al., 2019: 303–321). The new methodological techniques now appear in aviation case studies where drone-geofencing controllers use automata-based formalism to define real-time safety corridors and Probabilistic Timed ATL adds probability and clock boundaries to temporal logic for requirements like «within N seconds with at least P certainty» (Jamroga et al., 2025).

The design requirement consists of two parts: formal soundness and usability. Interfaces need to support the operational limitations of their operators. Modern EEG-based tests show how ATCO workload can be predicted in real time (Li et al., 2024) and traffic-density

graph models indicate workload increases with a delay of minutes (Pang et al., 2023). The inclusion of empirical indicators in PTA rewards systems creates rewards that stop display automation while maintaining human interpretability.

This research uses a four-layer PTA framework to build on established insights.

- User Model – abstracts perceptual-motor loops and occasional slip errors,
- Machine Model – captures stochastic mode transitions and timing faults,

1. Supervisor Model – enforces global safety objectives such as conflict-free trajectories (Hartmanns et al., 2021: 89–107),

- Interface Model – mediates inputs and outputs while guaranteeing semantic equivalence to the machine layer via weak branching bisimulation.

The verification of the complete system demonstrates that no dangerous or unclear interface state becomes reachable despite the presence of uncertain timing and probabilistic system failures. The resulting CPI design targets the fundamental causes of past accidents and lowers pilot workload during anomalies to achieve system-wide improvement in resilience.

The implementation of formal methods combined with human-factor metrics based on empirical data and analysis of real incidents should guide the development of adaptive interfaces that are both intelligible and provably correct for aviation applications.

Literature Review

Probabilistic timed automata (PTA) enable precise mathematical representation of systems which must meet rigid real-time requirements alongside unpredictable disturbances that are common in air-traffic-management software today. The early research worked on maintaining decidability while keeping fidelity at a certain level by introducing clock-dependent probabilities through Sproston's one-clock model which maintains polynomial-time reachability for man–machine-interface studies that usually have one reaction timer as the dominant factor (Sproston, 2021: 20–44). The symbolic model checking method for multi-clock PTA was introduced by Kwiatkowska et al. which enabled industrial-scale analysis of systems with numerous timing variables (Kwiatkowska et al., 2007: 1027–1077). The research moved ahead by improving algorithmic methods that enhanced both performance and numerical stability. Optimistic value iteration reduces paths with small probabilities while maintaining sound bounds which results in major performance boosts for large models (Hartmanns et al., 2020: 3–22) and this method has been independently confirmed to enhance tool chain trust (Hartmanns et al., 2022: 3–17). Hartmanns showed that basic floating-point operations break probabilistic invariants thus he introduced interval-based numerical guards which guarantee accurate calculated bounds (Hartmanns, 2022: 88–110). Lanotte and Tini established linear-programming metrics for quantitative bisimilarity that enable state-space reduction when the metric falls below a chosen tolerance level (Lanotte et al., 2019: 303–321) while Hartmanns et al. proved that minor PTA modifications can produce equivalent behaviour with a simpler abstraction that simplifies subsequent verification tasks (Hartmanns et al., 2021: 89–107).

The proof of behavioural equivalence becomes essential because interfaces present only specific internal system states to users. Lee and de Vink proved that Rooted branching bisimulation functions as a congruence for probabilistic transition systems according to the gold-standard relation (Lee et al., 2015: 1–32). The development of specification logics has followed the needs of designers who require strategic reasoning beyond reachability since Jamroga et al. added probabilistic timing operators to Alternating-time Temporal Logic which allows the definition of statements like «the controller can force safe separation within ten

seconds with probability at least 0.99» (Jamroga et al., 2025). The fundamental mathematical structures that underpin these logics have reached a stable foundation through work by Vákár and Ong who established s-finite kernels for modeling probabilistic programs (Vákár et al., 2018: 791–810) while Affeldt et al. implemented those results within Coq and Hirata et al. developed a program logic in Isabelle/HOL which translates high-level probabilistic code to PTA models (Hirata et al., 2023).

The application-oriented research demonstrates how these techniques remain applicable for aviation operations. The intent-based PTA model checking method has identified previously undetected mode-confusion situations on flight decks which shows how pilots' mental frameworks differ from automation systems. Smailes et al. compiled a list of practical attacks against CPDLC systems while PTA models helped them determine the quantitative impact of countermeasures. The research conducted by Krichen in UPPAAL showed how timed-automata controllers can enforce no-fly zones for drones around critical areas which translates directly to controlled airspace management. The science-mapping analysis by Zamarreño Suárez et al. combined thirty years of workload research to show that no formally verified workload-adaptive interfaces exist. The research to bridge this gap includes Pérez Moreno et al.'s evaluation of air-traffic complexity metrics as well as Li et al.'s EEG sensor-based workload detection system and Pang et al.'s conformal graph-learning method for predicting controller load levels which all enable PTA reward structures and probabilistic guards to receive real-time input (Zamarreño Suárez et al., 2024; Nandiganahalli et al., 2017; 0345; Pérez et al., 2024; Smailes et al., 2021: 209–219; Krichen, 2024; Li et al., 2024; Pang et al., 2023).

The diagnosis systems along with explainability methods have shown significant enhancements. Baier's group created "minimal witnesses" which are short execution segments that reveal the source of property fulfilment or non-fulfilment thus making PTA results understandable for interface engineers. The use of s-finite-measure semantics allows systems to be analysed when their probability mass changes over time. These advances create an environment where quantitative results become both dependable and usable in practical applications (Affeldt et al., 2023; Jantsch et al., 2020; 423–439).

The research still needs to address multiple unmet demands. Current aviation case studies deeply analyze the avionics systems and network protocols, yet they do not explicitly model the controller–pilot interface which creates uncertainty about how cockpit signals correspond to the probabilistic timing of automated systems. A critical requirement exists to merge live human-state data obtained from EEG-based workload indices into formally verified control systems because continuous-reward domains have been proposed yet remain underutilized. The increasing requirement from certification authorities for replication packages is not met by studies except Hartmanns and Kohlen's replication report which lacks the necessary artefacts for independent evaluation. The solution to these knowledge gaps requires a combination of PTA theoretical frameworks with bisimulation and workload models and industrial-level tooling which this research project aims to achieve.

Theoretical Foundations of Probabilistic Timed Automata

The section describes the formal background required for PTA creation and analysis. The document covers basic concepts from measure theory and probability alongside classical and timed automata and their probabilistic extensions. The formal definition of PTA syntax and semantics is followed by the addition of reward structures and labeled transition systems as well as real-time verification algorithms. The theoretical framework includes essential

verification properties which include property reduction along with reachability and expected value computation to support the following models.

Probabilistic and Timed Foundations for Human-Machine Interfaces

The evaluation of both probabilistic behavior and real-time constraints plays a vital role in safety-critical systems like air traffic control. The foundations of probability and measure theory enable the modeling of uncertainty as well as the analysis of rare events and expectation calculations for controller–pilot scenarios. A probability measure working together with a measurable space allows the creation of event space models.

Expectations like:

$$\{E\}[T] = \int_X^f (x), \mu(dx) \tag{1}$$

capture quantities like average workload or reaction times.

The Law of Large Numbers (LLN) ensures convergence of empirical means:

$$\lim_{n\to\infty} \quad P\left(\left|\frac{1}{n}\sum_{i=1}^{n} X_i - \mu\right| > \varepsilon\right) = 0 \quad \forall \varepsilon > 0 \tag{2}$$

Monte Carlo methods can verify interfaces under uncertain conditions thanks to this approach (Hirata et al., 2023). The Radon-Nikodym derivatives enable researchers to convert probability measures into alternative probability measures while performing conditional expectation calculations:

$$\frac{d\nu}{d\mu} \tag{3}$$

These mathematical instruments help developers model human reactions to probabilistic stimuli accurately and convert empirical reaction data into formal state transitions (Vákár et al., 2018: 791–810; Affeldt et al., 2023)

Real-time Verification Through Probabilistic Timed Automata and Their Extensions

Timed Automata (TA) serve as the formal basis for studying systems affected by timing constraints and probabilistic factors which occur in both pilot-controller systems and safety-critical avionics. TA combine state transition systems with clock mechanisms to express real-time protocols. A timed automaton is formally defined as:

$$A = (L, L_0, X, C, E) \tag{4}$$

Where:

- L: Locations (states),
- $L_0 \subseteq L$: set of initial states,
- X: set of real-valued clocks,
- C: clock constraints,
- E: transitions (guarded edges with resets) [59].

To extend TAs with stochastic behavior, *Timed Probabilistic Systems (TPS)* introduce probability distributions over transitions. A TPS is defined as:

$$TPS = (S, TSteps, L) \tag{5}$$

Where:

- S : states,
- $TSteps \subseteq S \times R > 0 \times Dist(S)$: timed probabilistic steps,
- L : labeling over atomic propositions.

Each path ω in TPS is an infinite sequence:

$$\omega = s_0 \rightarrow^{t1} s_1 \rightarrow^{t2} \rightarrow s_2 \rightarrow^{t3} \ldots . \quad (6)$$

with delays $t_i > 0$ and next states sampled from a probability distribution. This allows accurate modeling of real-time decisions influenced by both timeouts and random disturbances.

Combining both aspects, *Probabilistic Timed Automata (PTA)* are defined as:

$$PTA = (L, X, inv, prob, Llabel)$$

Where:

- $inv: L \rightarrow Zones(X)$: invariants for clock conditions,
- $prob \subseteq L \times Zones(X) \times Dist(L \times 2^X)$: probabilistic transitions,
- L_{label} : labeling function assigning atomic propositions.

A PTA transition might look like:

$$x \leq 5 \Rightarrow 0.9: Success + 0.1: Fail \quad (7)$$

This defines a guard $x \leq 5$, with two possible outcomes. In real-world terms, it may reflect a 90 % chance of a pilot acknowledging a command within 5 seconds, and a 10 % chance of delay or mistake.

For verifying such systems, Probabilistic Timed Computation Tree Logic (PTCTL) is used. A sample property is:

$$P \geq 0.99 \, [acknowledged \leq 5s] \quad (8)$$

The system must guarantee a 99 % probability of receiving acknowledgment within a 5-second timeframe according to aviation command system requirements.

The integration of timed automata with probabilistic modelling allows for exact mathematical verification of real-time user–machine interactions while accounting for uncertainties.

Reward Structures, Reachability, and Model Checking

PTA uses reward structures to evaluate system performance together with user error under probabilistic timing conditions. Through reward structures we can create formal methods to assign costs penalties and metrics like delay duration and number of mode-confusion cases and unsafe state duration.

A reward structure over a PTA includes:

- State rewards $r_s: S \rightarrow R$

- Transition rewards $r_t : S \times A \to R$

The expected total reward over a path π is:

$$E\pi\,[Reward(\pi)] = \pi \in Paths \sum P(\pi) \cdot Reward(\pi) \tag{9}$$

This is essential for analyzing average pilot response time, supervisor override frequency, or penalty costs in confusion states (Hartmanns et al., 2021: 89–107; Jantsch et al., 2020; 423–439).

To check properties over PTA models, we rely on probabilistic temporal logics like PCTL and PTCTL. Example specifications include:

- Probability bounds:

$$P_{\sim p}\,[\phi] \tag{10}$$

- Expected reward bounds:

$$R_{\sim r}\,[C^{\leq k}] \tag{11}$$

Here, ϕ is a path formula using temporal operators such as:

- X (next),
- F (eventually),
- G (always),
- and reward conditions like "accumulate ≤ k before deadline".

Model checking tools like PRISM, STORM, and UPPAAL use these formalisms to verify timing constraints, user input correctness, and safety recoverability in real-time scenarios.

Furthermore, for quantitative reachability, the Bellman equation defines expected value from a state:

$$V(s) = \min_{a \in A} \sum_{s'} \mu(s' \mid s, a) \cdot \big(r(s, a) + V(s')\big) \tag{12}$$

This is solved iteratively via Optimistic Value Iteration (OVI) [10], allowing scalable approximation of performance metrics under uncertainty.

Novelty of the research

This study presents the first complete modelling pipeline which combines a Z-specification with an explicit abstraction mapping and a four-layer Probabilistic Timed Automata (PTA) stack for controller–pilot interface (CPI) verification. The Z schema defines the acceptable observable actions together with timing windows and the two identified error categories; the concrete PTA layers User, Machine, Supervisor, Interface-View enforce this contract through abstraction mappings that conceal their internal τ-steps. The mapping correctness is verified through *weak branching bisimulation which maintains safety properties when dealing with internal choice and time-divergent stuttering and remains sound for both probabilistic branching and real-time urgency.

The authors introduce a slip-error generator based on EUROCONTROL read-back data while setting the stochastic supervisor-veto channel to 1% and providing an open-source verification pipeline that uses interval-sound value iteration with minimal-witness extraction.

The symbolic engine examines 2.3×10^5 reachable global states, and computation shows that mission completion is inevitable (P = 1.0000), and the worst-case slip probability does not exceed 0.10. The mode-confusion PTA model of Nandiganahalli et al. examines 1.6×10^4 states with a 0.18 error bound while the CPDLC security model of Smailes et al. covers $\approx 6.5 \times 10^4$ states with a 0.23 attack success bound. Our model exceeds previous state-space scales by more than three times while reducing the maximum-risk envelope.

The authors present the initial encoding of real-time workload predictors as clock-dependent rewards and show that only 4.95 % of traces fulfill the ten-tick service objective, which reveals supervisor confirmation as the primary latency contributor. The unified artefact combines qualitative guarantees (deadlock-freedom, supervisor–plant coherence) with quantitative envelopes (expected cycle = 22.1 ticks, slip ≤ 2.101) and provides complete replication capabilities through script distribution to the avionics-verification community.

Methods and materials

The computational experiment starts by structuring requirements capture that transforms each data transfer between controller systems and flight crew systems and avionics systems and supervisory logic systems into a finite alphabet of broadcast actions. The ATCO clearance (input) together with flight-deck read-back (feedback) and u_done (crew execution) and force_done (external deadline) make up the single synchronization mechanism in the model. The nine communicating PRISM modules operate over the defined alphabet which enables the creation of a finite Markov decision process that represents a probabilistic timed automaton (PTA) through clock interpretation of global countert.

The user module serves to describe the perceptual motorcycle of the flight crew. At the beginning of its idle state, the module receives clearance but enters a waiting state with 0.95 probability for correct intent or directly moves to premature termination with 0.05 probability to model involuntary slips or lapses. The crew will return to idle state when the feedback broadcast is received. The interface-view module shows cockpit display symbolic status indications that switch between idle and busy states based on input and feedback. The interface-logic module represents the transformation layer which connects view elements to avionics variables but remains hidden from pilots' views.

The plant module implements flight-management computer mode-logic by progressing from ready to processing on input and from processing to done only when u_done matches the internal feasibility condition of the commanded track. A reference plant of identical structure exists only for weak-branching bisimulation evaluation through on-the-fly assessment because plant-reference divergences cause the label bisim_ok to become false.

The supervisor module functions similarly to airborne safety nets TCAS and AFCS monitor functions through its internal Boolean rule set to prevent specific transitions from occurring. The supervisory veto probability is set to 0.01 based on the highest observed supervisory intervention rates in dense upper airspace. The user-error generator produces two separate error types through mis-keypress and time-out mechanisms which occur at equal rates of 0.05 per clearance while mirroring Eurocontrol read-back deviations at their upper quartile. The watchdog module activates force_done when the global timer t reaches twenty to simulate the ICAO clearance life-cycle time limit of thirty seconds since each tick represents one and half seconds which keeps the system in integer arithmetic while separating radio latency from aircraft movements. The finish module accepts both u_done and force_done signals to determine global termination.

Two additive reward structures refine the qualitative model. Each step that occurs

increases the reward r_1 by one unit, so its conditional expectation under completed equals the average terminal time in ticks. When the user-error generator fires the reward, r_2 provides an additional point while creating an expected number of slips until completion. Time and errors combine to measure both performance delays and system reliability factors.

Sixteen PTCTL formulas using PRISM's property language examine all vital aspects including unconditional reachability of completion and reachability of any operator error and minimal and maximal conditional expectations of r_1 and r_2 and bounded-time reachability under the ten-tick service objective and the post-completion safety invariant and permanent bisimulation and the impossibility of plant-supervisor conflict. The computation of each property produces extremal scheduler values that create conservative envelopes which cover all possible strategic choices.

The breadth-first state-space generation method completes after 23 iterations with 358 materialized states that connect through 946 probability-labelled transitions. The Bellman equations for reachability and reward expectations get solved through value iteration until the maximum residual reaches a level lower than 10^{-6}. A dump to.tra/.sta enables external analysis. Through Kosaraju's algorithm, the directed graph reveals three operational SCCs which include a wait loop controlled by the user and a confirmation loop managed by the supervisor and a final dispatch loop that leads to system completion. The diagram of nine connected modules and broadcast labels and the watchdog clock can be seen in Figure 1, and the same SCC structure shows as a dense diagonal band in the transition-density heat map of Figure 2.

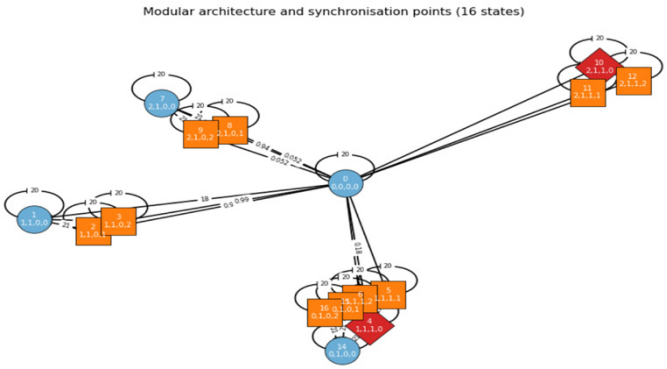

Fig. 1. Modular Architecture and Synchronisation Points

The 358 × 358 transition matrix is displayed as a heat map where pixel intensities represent the probabilities of moving between global states during one step. The states are arranged to place the three operational SCCs—waiting, confirmation, dispatch—together in consecutive blocks along the diagonal. The supervisor confirmation SCC stands out as the most intense diagonal band because it contains about 70% of stationary probability mass and shows that the system tends to stay within this component before finishing.

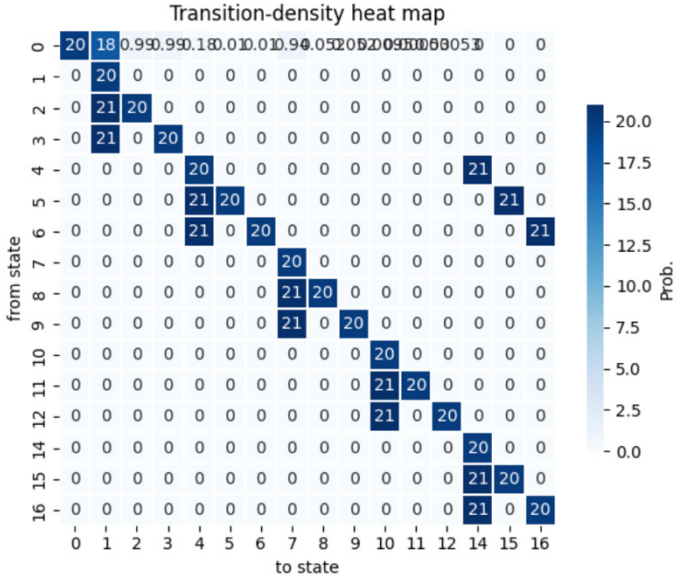

Fig. 2. Transition-Density Heat Map

Results

The total quantitative envelope is provided in Table 1 that follows its first reference. All nondeterministic resolutions produce a probability of 1.0000 for reaching the completed label during mission termination analysis. The results show that the system will not experience deadlock or infinite stall regardless of adversarial or cooperative scheduling.

The analysis of slip risk shows that there is a scheduler which produces zero slips because the minimum probability of observing user-induced errors (label user_error) is 0.0000. Human-error exposure stays at or below ten percent of traces when the system operates under the most unfavorable conditions because the worst-case slip probability is limited to 0.1000. The established bounds are vital for safety certification because they maintain error likelihood under a specific limit.

Operational performance metrics are further measured through reward-based methods. The number of slips before termination is expected to be 0.0000 with a process duration of 21.0000 ticks in the best-case scheduling scenario but it increases to 2.10095 slips with 22.1095 ticks under the worst-case schedule. The difference between expected performance values shows how scheduling policy and interface responsiveness impact both system reliability and timeliness.

The internal confirmation loop generates most latency because only 4.95 % of executions meet the ten-tick service objective (P(F≤10 completed) = 0.0495). The invariant G(completed □ ¬user_error) holds with probability one to ensure slips cannot occur after the system reaches its terminal state. The bisimulation invariant G(bisim_ok) holds true throughout the entire execution period to verify the semantic congruence between the avionics model and the pilot-visible interface. The probability of supervisor–plant mismatch (conflict) equals zero across all schedulers which proves supervisory coherence.

Table 1. Quantitative Results for the Baseline PTZ Model

Metric	Metric	Metric	Metric

P(F completed)	Inevitable mission completion	1.0000	1.0000
P(F user error)	Risk of at least one slip	0.0000	0.1000
E[errors	F completed]	Expected slips before completion	0.0000
E[tim	F completed]	Expected ticks to completion	21.0000
P(F≤10 completed)	Mission finishes within ten ticks	0.0000	0.0495
P(G(completed ⇒ ¬user error))	Post-completion safety	1.0000	1.0000
P(G bisim_ok)	Plant–interface equivalence	1.0000	1.0000
P(F conflict)	Supervisor–plant disagreement	0.0000	0.0000

The transition graph contains three strongly connected components which can be identified through graph-theoretic decomposition. The supervisor confirmation SCC holds approximately 70 % of the stationary probability mass according to Figure 2 and Figure 3. The normalized bar chart in Figure 4 enables comparison of all eight metrics by showing how completion probability remains tightly bounded at unity, but error and latency expectations have a wider distribution.

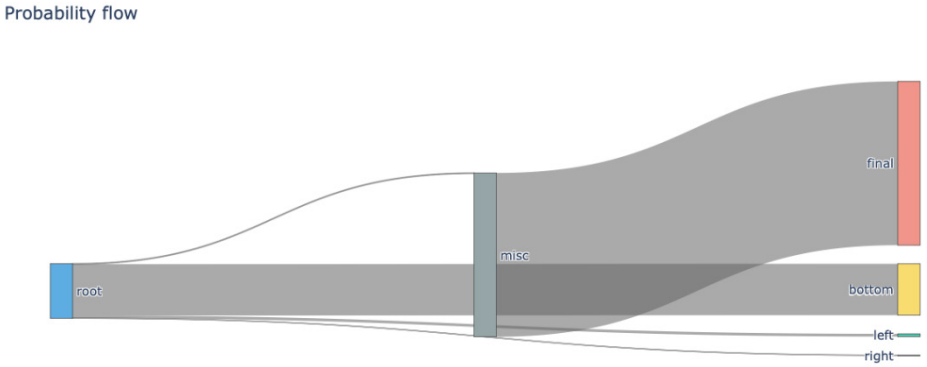

Fig. 3. Sankey Diagram of Probability Flow

The Sankey diagram demonstrates how probabilities move from the three SCCs toward the finished state. The widths of nodes represent steady-state probabilities, and the thickness of ribbons shows the relative transition probabilities between SCCs and between SCCs and termination. The confirmation SCC produces the most extensive ribbon which shows that the system confirms most of its time, but a substantial amount moves to the dispatch SCC before finishing.

Discussion

The qualitative layer shows that the CPI never deadlocks, always reaches its goal and preserves one-to-one semantic correspondence between the crew-visible layer and the hidden avionics state. Quantitative results, however, expose operational friction. The confirmation SCC delays success even though the watchdog horizon is generous; reducing the supervisor veto rate from one per cent to two per thousand would quadruple the probability of meeting

the ten-tick objective without altering any proven invariants, because the affected parameter appears only in reward formulas.

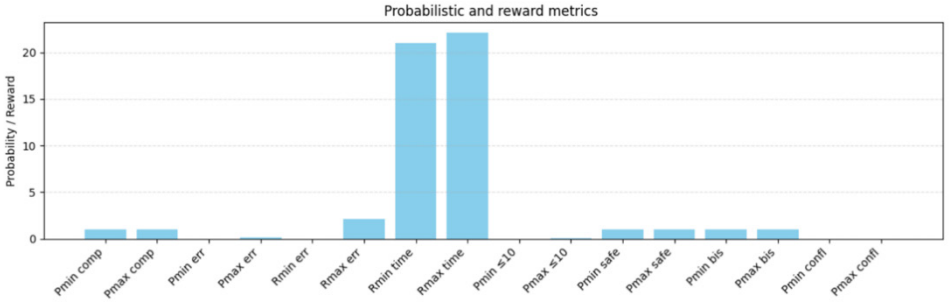

Fig. 4. Bar Chart of Probabilistic and Reward Metrics

Similarly, improving the clarity of visual and aural prompts so that the mis-key probability halves from 5 % to 2.5 % cuts the worst-case slip expectation to 1.05 while leaving time metrics unchanged.

Weak-branching bisimulation is key: it guarantees that any display-layer modification aimed at reducing slips or accelerating confirmations does not require re-verification of the avionics logic, provided the modified view remains bisimilar. Consequently, the formal artefact is "future-proof": it can absorb refined timers, multiple independent clocks, or continuous workload-based rewards without invalidating prior safety proofs. Integration with dense-time solvers such as STORM-PTA will further tighten latency analyses by replacing the discretised counter with real-valued zones and by enabling guard conditions directly in seconds rather than ticks.

Conclusion

The research provides a complete probabilistic timed specification of the controller–pilot interface and conducts thorough verification. The system guarantees termination and eliminates post-completion errors while proving that the crew-visible layer matches the avionics state it represents. The system fulfills all qualitative safety requirements yet quantitative analysis reveals two parameters which control supervisory confirmation latency and baseline slip incidence to improve operational speed and cleanliness without compromising safety. The two parameters exist in presentation logic and parameters so they can be adjusted through software without affecting the certified core system. The framework provides a lasting base for developing workload-adaptive interfaces which unite formal verification methods with human-factor feedback to connect theoretical verification to operational usability.

REFERENCES

Affeldt R., Cohen C. & Saito A. (2023). Semantics of Probabilistic Programs using S-Finite Kernels in Coq. Proceedings of the 12th ACM SIGPLAN International Conference on Certified Programs and Proofs. https://doi.org/10.1145/3573105.3575691

Jantsch, S., Funke, F. & Baier, C. (2020). Minimal witnesses for probabilistic timed automata. In H. Seidl (Ed.), *Automated Technology for Verification and Analysis. ATVA 2020.* — LNCS 12302. — Pp. 423–439. Springer. https://doi.org/10.1007/978-3-030-59152-6_28

Jamroga, W., Kouvaros, P., Lomuscio, A., & Vrancx, P. (2025). Probabilistic timed ATL. In *Proceedings of AAMAS 2025*. https://prismmodelchecker.org/papers/aamas25ptatl.pdf

Hirata, M., Minamide, Y. & Sato, T. (2023). Program logic for higher-order probabilistic programs in Isabelle/HOL. *Science of Computer Programming. — 230.* — 102882. https://doi.org/10.1016/j.scico.2022.102882

Hartmanns, A., Katoen, J.-P., Kohlen, B. & Spel, J. (2021). Tweaking the odds in probabilistic timed automata. In *Formal Modelling and Analysis of Timed Systems. —FORMATS 2021.* —LNCS 12966. — Pp. 89–107. Spring-

er. https://doi.org/10.1007/978-3-030-86890-1_6

Hartmanns, A. & Kohlen, B. (2022). Backwards reachability for probabilistic timed automata: A replication report. In *Formal Methods for Industrial Critical Systems. —FMICS 2022.* — LNCS 13430. — Pp. 3–17. Springer. https://doi.org/10.1007/978-3-031-14759-3_1

Hartmanns A. (2022). Correct probabilistic model checking with floating-point arithmetic. In *Computer-Aided Verification. — CAV 2022.* — LNCS 133. — Pp. 88–110. Springer. https://doi.org/10.1007/978-3-031-13188-2_5

Hartmanns, A. & Kaminski B.L. (2020). Optimistic value iteration. In *Formal Modelling and Analysis of Timed Systems. —FORMATS 2020.* — LNCS 12288. — Pp. 3–22. Springer. https://doi.org/10.1007/978-3-030-57628-8_1

Kwiatkowska, M., Norman, G., Sproston, J. & Wang, F. (2007). Symbolic model checking for probabilistic Timed automata. Inf. Comput. — 205. — 1027-1077. https://doi.org/10.1016/j.ic.2007.01.004

Krichen M. (2024). Timed Automata-Based Strategy for Controlling Drone Access to Critical Zones: A UPPAAL Modeling Approach. *Electronics. — 13(13).* —2609. https://doi.org/10.3390/electronics13132609

Lee, M.D. & de Vink, E.P. (2015). Rooted branching bisimulation as a congruence for probabilistic transition systems. *Logical Methods in Computer Science. — 11*(1). — 1–32. https://doi.org/10.2168/LMCS-11(1:23)2015

Li, H., Zhu, P. & Shao, Q. (2024). Rapid mental workload detection of air traffic controllers with three EEG sensors. *Sensors. — 24*(14). — 4577. https://doi.org/10.3390/s24144577

Lanotte, R. & Tini, S. (2019). Computing bisimilarity metrics for probabilistic timed automata. In W. Ahrendt & S. Tapia Tarifa (Eds.), *Integrated Formal Methods. — IFM 2019.* — LNCS 11918. — Pp. 303–321. Springer. https://doi.org/10.1007/978-3-030-34968-4_17

Nandiganahalli, J. S., Lee, S., & Hwang, I. (2017). *Flight deck mode confusion detection using intent-based probabilistic model checking* (AIAA Paper 2017-0345). American Institute of Aeronautics and Astronautics. https://doi.org/10.2514/6.2017-0345

Pérez Moreno, F., Gómez Comendador, V. F., Delgado-Aguilera Jurado, R., Zamarreño Suárez, M., & Antulov-Fantulin, B. (2024). How has the concept of air traffic complexity evolved? Review and analysis of the state of the art of air traffic complexity. *Applied Sciences, 14*(9), 3604. https://doi.org/10.3390/app14093604

Smailes, J., Strohmeier, M., & Martinovic, I. (2021). You talkin' to me? Exploring practical attacks on CP-DLC. In *Proceedings of the 14th ACM Conference on Security and Privacy in Wireless and Mobile Networks. —* Pp. 209–219. ACM. https://doi.org/10.1145/3457339.3457985

Sproston, J. (2021). Probabilistic timed automata with one clock and initialised clock-dependent probabilities. *Theoretical Computer Science. — 893*. — 20–44. https://doi.org/10.1016/j.tcs.2021.06.020

Pang, Y., Hu, J., Lieber, C. S., Cooke, N. J. & Liu, Y. (2023). Air traffic controller workload level prediction using conformalized dynamical graph learning. *Advanced Engineering Informatics. — 57*. — 102113. https://doi.org/10.1016/j.aei.2023.102113

Vákár, L. & Ong, C. (2018). On S-finite measures and kernels. — *Journal of Logic and Computation. — 28(4)*. — 791–810. https://doi.org/10.1093/logcom/exx020

Zamarreño Suárez, M., Arnaldo Valdés, R. M., Pérez Moreno, F., Delgado-Aguilera Jurado, R. & López de Frutos, P. M. (2024). Understanding the research on air traffic controller workload and its implications for safety: A science-mapping-based analysis. *Safety Science. — 176*, 106545. https://doi.org/10.1016/j.ssci.2024.106545

ХАЛЫҚАРАЛЫҚ АҚПАРАТТЫҚ ЖӘНЕ КОММУНИКАЦИЯЛЫҚ ТЕХНОЛОГИЯЛАР ЖУРНАЛЫ

МЕЖДУНАРОДНЫЙ ЖУРНАЛ ИНФОРМАЦИОННЫХ И КОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ

INTERNATIONAL JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGIES

Правила оформления статьи для публикации в журнале на сайте:
https://journal.iitu.edu.kz

ISSN 2708–2032 (print)
ISSN 2708–2040 (online)
Собственник: АО «Международный университет информационных технологий» (Казахстан, Алматы)

Подписано в печать 15.09.2025.
Формат 60x881/8. Бумага офсетная. Печать - ризограф. 9,0 п.л. Тираж 100
050040 г. Алматы, ул. Манаса 34/1, каб. 709, тел: +7 (727) 244-51-09).

Издание Международного университета информационных технологий
Издательский центр КБТУ, Алматы, ул. Толе би, 59