Автономная некоммерческая организация высшего образования
«Университет Иннополис»

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
(БАКАЛАВРСКАЯ РАБОТА)
по направлению подготовки
09.03.01 - «Информатика и вычислительная техника»

GRADUATION THESIS
(BACHELOR'S GRADUATION THESIS)
Field of Study
09.03.01 – "Computer Science"

Направленность (профиль) образовательной программы
«Информатика и вычислительная техника»

Area of Specialization / Academic Program Title:
"Computer Science"

Тема / Topic	**Интерактивный помощник по редактированию и оптимизации расписания в образовательных учреждениях / Interactive Assistant for Timetable Editing and Optimization in Educational Institutions**	
Работу выполнил / Thesis is executed by	**Руслан Бельков / Ruslan Belkov**	подпись / signature
Руководитель выпускной квалификационной работы / Supervisor of Graduation Thesis	**Александр Гасников / Alexander Gasnikov**	подпись / signature

Иннополис, Innopolis, 2025

Contents

Abstract

Academic timetabling is a complex combinatorial optimization problem that must satisfy diverse constraints, such as room availability, teacher schedules, and student group assignments. While much research has focused on generating complete timetables from scratch [1], real-world practice reveals that the greater challenge lies in editing and maintaining timetables once they are published. Polls indicate that more than 60% of educational institutions must revise at least 10% of their schedules after release due to unforeseen disturbances [2]. Such revisions include teacher illness, room unavailability, or new course requirements, and are often performed manually, consuming significant time and effort.

This thesis addresses the problem of **interactive timetable editing and adaptive rescheduling**. The proposed system integrates with existing workflows used by the Department of Education at Innopolis University, built on Google Sheets and Microsoft Outlook. It provides real-time conflict detection, automated constraint checking, and minimally disruptive rescheduling using optimization techniques such as Constraint Programming [3]. To further improve usability, the system employs Large Language Models (LLMs) as a natural language interface, translating informal user requests into formal scheduling constraints [4]. This hybrid human–AI approach enables schedulers to interact with the system in plain English while preserving mathematical rigor in optimization.

The novelty of this research lies in combining state-of-the-art optimization, interactive UI integration, and AI assistance in a single practical tool. The system bridges the gap between academic prototypes, which emphasize algorithms but neglect usability, and commercial solutions, which provide user-friendly interfaces but rely on outdated methods [5]. Through this work, we aim to demonstrate that blending algorithmic efficiency, natural language interaction, and workflow integration can create a scheduling assistant that is both scientifically innovative and practically valuable.

Chapter 1

Introduction

Timetabling in educational institutions is a long-standing and challenging problem. A valid timetable must simultaneously satisfy a variety of hard and soft constraints, such as the availability of teachers, rooms, and student groups, while also considering preferences like minimizing idle time or aligning lectures and tutorials on the same day. The problem is NP-hard, which explains the abundance of heuristic, metaheuristic, and optimization-based approaches proposed in the literature [1].

In practice, most institutions employ a hybrid process: an initial draft is prepared using software or spreadsheets, and human schedulers manually refine it to meet real-world requirements. However, timetables are rarely static. Once published, they must be updated regularly due to unforeseen events such as teacher illness, changes in preferences, or sudden room reservations for events. Studies report that over 60% of institutions modify at least 10% of lessons after the official schedule is released [2]. This demonstrates a gap: while much research focuses on generating timetables from scratch, the real-world challenge often lies in **editing and maintaining timetables under changing conditions** [3].

1.1 Motivating Example

Imagine a week before classes start, a lecturer falls ill and two major events suddenly block the largest lecture halls. The scheduler must manually reshuffle dozens of sessions in Google Sheets, re-check Outlook bookings, and notify teachers and students. Even small changes can cascade into conflicts: a class moved to another slot collides with a teaching assistant's own studies, or a group ends up with three long gaps in one day. Such cases occur multiple times every semester at Innopolis University, taking hours of manual work.

This example illustrates the real challenge: not generating a timetable from scratch, but efficiently adapting it to new conditions with minimal disruption. A system that could automatically detect conflicts, suggest minimally invasive changes, and integrate seamlessly with existing tools would significantly reduce the workload of Department of Education staff.

1.2 Research Questions

This thesis addresses the above gap by focusing on interactive timetable editing and minimally disruptive rescheduling. The research is guided by the following questions:

- How can optimization methods be adapted to efficiently handle timetable modifications while preserving most of the original structure?
- How can natural language interfaces (LLMs) support human schedulers in formulating and applying changes?
- What user interface and workflow integrations are necessary to make such a system truly useful in daily operations?

The goal is to develop and evaluate an interactive assistant that not only supports academic research but also provides real utility to the Department of Education staff at Innopolis University. The assistant will be integrated into existing workflows, operate directly in Google Sheets, and synchronize changes with Microsoft Outlook calendars. By combining optimization techniques, conflict detection, and natural language assistance, this work aims to contribute both scientifically and practically.

Chapter 2

Literature Review

This chapter surveys research and systems relevant to timetable optimization, dynamic rescheduling, and interactive scheduling tools. Sources include Google Scholar[1], ACM Digital Library[2], arXiv[3], and public blogposts. The review is organized into four key areas: (1) timetabling methods, (2) dynamic rescheduling, (3) user interface and workflow integration, and (4) AI-assisted scheduling.

2.1 Timetabling methods

Timetabling has been extensively studied as a combinatorial optimization problem. Early approaches relied on exact mathematical methods, such as Integer Linear Programming and graph coloring formulations. Due to computational complexity, modern systems often employ metaheuristics: tabu search, simulated annealing, genetic algorithms, ant colony optimization, and others [3]. Constraint Programming (CP) has emerged as a powerful paradigm, offering expressive mod-

[1] https://scholar.google.com/

[2] https://dl.acm.org/

[3] https://arxiv.org/

eling of constraints and efficient solving through CP-SAT solvers such as Google OR-Tools. Despite progress, no universal method exists: real-world timetabling instances remain highly heterogeneous, requiring hybrid and tailored solutions [1].

2.2 Dynamic rescheduling

A less studied but practically critical problem is **rescheduling under disturbances**. Once a timetable is fixed, changes must be incorporated with minimal disruption to students and teachers. This is formalized as the Minimum Perturbation Problem (MPP), which seeks to satisfy new constraints while minimizing the number of altered assignments [3]. Methods include local search, CP with distance metrics, and multi-objective optimization balancing stability against compactness. Studies of school timetabling under disturbances show that while gaps in student schedules can be reduced by reassignments, such changes must be limited to avoid confusion and excessive workload. This motivates the development of efficient algorithms for timetable repair rather than regeneration.

2.3 User interface and workflow integration

The gap between academic prototypes and practical systems often lies in usability. Commercial products (e.g., aSc Timetables, Untis, Celcat) emphasize drag-and-drop editing, substitution management, and publishing schedules online or to mobile devices. Open-source tools such as UniTime and FET provide flexible constraint modeling and integration into institutional workflows. However, many rely on outdated algorithms. Research indicates that the combination of modern optimization with user-friendly interfaces is essential for adoption [5]. Integration with widely used platforms (Google Sheets, Outlook) further reduces friction and improves collaboration across departments.

2.4 AI-assisted scheduling

Recent advances in AI, particularly Large Language Models, have opened new opportunities for timetabling. LLMs can serve as natural language interfaces for expressing scheduling changes (e.g., "Move the math lecture to Wednesday morning") and automatically translating them into formal constraints. Research prototypes (e.g., RAG-DyS) show promising results in combining LLMs with CP solvers for dynamic rescheduling, ensuring minimal deviations from the original schedule [4]. Additionally, recent work such as ReEvo [6] demonstrates that LLMs can act as hyper-heuristics: the model itself generates or adapts heuristics for combinatorial optimisation problems via reflective evolution, exploring heuristic space in an automated fashion . Commercial applications have begun experimenting with AI-powered rescheduling assistants capable of real-time conflict resolution and automated notifications [7].

2.5 Summary

The literature demonstrates that while timetable generation has been deeply explored, the problem of interactive editing and adaptive rescheduling is still underdeveloped. Practical systems often prioritize usability at the cost of algorithmic sophistication, while research prototypes neglect workflow integration. There is clear space for innovation in combining optimization, dynamic repair algorithms, natural language assistance, and integration with everyday tools like Google Sheets and Outlook. This thesis positions itself precisely at this intersection.

Bibliography cited

[1] S. Ceschia *et al.*, "Educational timetabling: Problems, benchmarks, and state-of-the-art results," *European Journal of Operational Research*, vol. 308, no. 1, pp. 1–18, Jul. 2023, doi: 10.1016/j.ejor.2022.07.011.

[2] Coursedog, "The 4 Challenges of Excel Course Scheduling." [Online]. Available: https://www.coursedog.com/articles/the-4-challenges-of-excel-course-scheduling

[3] M. Veenstra and I. Vis, "School timetabling problem under disturbances," *Computers & Industrial Engineering*, vol. 95, p. , 2016, doi: 10.1016/j.cie.2016.02.011.

[4] P. M. Tang *et al.*, "Automated Conversion of Static to Dynamic Scheduler via Natural Language." [Online]. Available: https://arxiv.org/abs/2405.06697

[5] R. Oude Vrielink *et al.*, "Practices in timetabling in higher education institutions: a systematic review," *Annals of Operations Research*, vol. 275, p. , 2019, doi: 10.1007/s10479-017-2688-8.

[6] H. Ye *et al.*, "ReEvo: Large Language Models as Hyper-Heuristics with Reflective Evolution." [Online]. Available: https://arxiv.org/abs/2402.01145

[7] Virtosoftware, "Transforming School Scheduling with AI." [Online]. Available: https://blog.virtosoftware.com/ai-schedule-maker-for-schools/