

# **PYTHON DASTURLASH ASOSIDAGI HISOBLASH MODELLARINI GIBRID MUAMMOLI TA'LIM (HPBL) BILAN INTEGRATSIYALASH ORQALI MOLEKULAR FIZIKADA MIQDORIY MUAMMO YECHISH KOMPETENSIYALARINI RIVOJLANTIRISH**

*Rahmonova Xurshida Gaybullayevna<sup>1</sup>, Sheraliyev Mashrab Umurzakovich<sup>2</sup>*

<sup>1</sup> *Samarkand State University Urgut filiali assistenti, Samarqand, O'zbekiston*

<sup>2</sup> *PhD, Samarkand State Medical University assistenti, Samarqand, O'zbekiston*

## **ANNOTATSIYA**

Molekulyar fizikada analitik muammolarni yechish ko'nikmalarini rivojlantirish talabalardan abstrakt matematik formulalar bilan dinamik fizik tizimlar o'rtasidagi bog'liqlikni tushunishni talab qiladi. Mazkur maqolada Gibridd Muammoli Ta'lim (HPBL) yondashuvini Python dasturlash asosidagi hisoblash simulyatsiyalari bilan integratsiyalashning pedagogik samaradorligi tahlil qilinadi. Kvazi-eksperimental tadqiqot 60 nafar fizika yo'nalishi talabasi ishtirokida olib borilib, ushbu integrallashgan hisoblash modeli an'anaviy o'qitish metodlari bilan taqqoslandi. Tajriba guruhi (n=30) NumPy, Pandas va SciPy kutubxonalarini yordamida termodinamik hodisalarni vizual modellashtirish uchun sonli modellar yaratdi va tahlil qildi, nazorat guruhi (n=30) esa standart analitik yechimlardan foydalandi. Post-test natijalari HPBL-Python modeli sezilarli ta'limiy samaradorlikka ega ekanligini ko'rsatdi: tajriba guruhi nazorat guruhidan yuqori natija qayd etdi ( $t(58)=7.942$ ,  $p<0.001$ ,  $d=1.98$ ). Sifat tahlillari shuni ko'rsatdiki, talabalarining 91% i Python vizualizatsiya kutubxonalarini murakkab vizuospatial vazifalarni bajarishda kognitiv yuklamani kamaytirishini ijobiy baholagan.

**Kalit so'zlar:** Gibridd Muammoli Ta'lim (HPBL); Python simulyatsiyalari; Hisoblash tafakkuri; Vizuospatial qobiliyat; Molekulyar fizika pedagogikasi.

## **ИНТЕГРАЦИЯ ВЫЧИСЛИТЕЛЬНОГО МОДЕЛИРОВАНИЯ НА ОСНОВЕ PYTHON С ГИБРИДНЫМ ПРОБЛЕМНО-ОРИЕНТИРОВАННЫМ ОБУЧЕНИЕМ (HPBL) ДЛЯ ФОРМИРОВАНИЯ КОЛИЧЕСТВЕННЫХ КОМПЕТЕНЦИЙ РЕШЕНИЯ ЗАДАЧ ПО МОЛЕКУЛЯРНОЙ ФИЗИКЕ**

*Рахмонова Хуршида Гайбуллаевна<sup>1</sup>, Шералиев Машираб Умурзакович<sup>2</sup>*

<sup>1</sup> *Ассистент, Ургутский филиал Samarkand State University, Самарканд, Узбекистан*

<sup>2</sup> *Ассистент, PhD, Samarkand State Medical University, Самарканд, Узбекистан*

## **АННОТАЦИЯ**

Развитие аналитических навыков решения задач по молекулярной физике требует от студентов способности устанавливать связь между абстрактными математическими формулами и динамическими физическими системами. В данной статье анализируется педагогическая эффективность интеграции гибридного проблемно-ориентированного обучения (HPBL) с вычислительными симуляциями на основе Python. Квазиэкспериментальное исследование было проведено среди 60 студентов-физиков с целью сравнения данной интегрированной методики

вычислительного моделирования с традиционными методами обучения. Экспериментальная группа (n=30) активно разрабатывала и анализировала численные модели с использованием библиотек NumPy, Pandas и SciPy для визуализации термодинамических процессов, тогда как контрольная группа (n=30) использовала стандартные аналитические методы решения. Результаты пост-тестирования показали высокую эффективность модели HPBL-Python: экспериментальная группа значительно превзошла контрольную ( $t(58)=7.942$ ,  $p<0.001$ ,  $d=1.98$ ). Качественный анализ показал, что 91% студентов положительно оценили способность библиотек визуализации Python снижать когнитивную нагрузку при выполнении сложных визуально-пространственных задач.

**Ключевые слова:** Гибридное проблемно-ориентированное обучение (HPBL); Python-симуляции; вычислительное мышление; визуально-пространственные способности; педагогика молекулярной физики.

## INTEGRATING PYTHON-BASED COMPUTATIONAL MODELING WITH HYBRID PROBLEM-BASED LEARNING (HPBL) TO CULTIVATE QUANTITATIVE PROBLEM-SOLVING COMPETENCIES IN UNDERGRADUATE MOLECULAR PHYSICS

*Rahmonova Xurshida Gaybullayevna<sup>1</sup>, Sheraliyev Mashrab Umurzakovich<sup>2</sup>*

<sup>1</sup> Assistant, Urgut Branch of Samarkand State University, Samarkand, Uzbekistan

<sup>2</sup> Assistant, PhD, Samarkand State Medical University, Samarkand, Uzbekistan

### ABSTRACT

Developing analytical problem-solving skills in molecular physics requires students to successfully bridge the gap between abstract mathematical formulas and dynamic physical systems. This paper evaluates the pedagogical impact of combining Hybrid Problem-Based Learning (HPBL) with Python-based computational simulations. A quasi-experimental study was implemented with 60 physics undergraduates to compare this integrated computational modeling methodology against conventional instructional practices. The experimental cohort (n=30) actively developed and manipulated numerical models using NumPy, Pandas, and SciPy to visualize thermodynamic phenomena, whereas the control cohort (n=30) followed standard analytical derivations. Quantitative post-test analytics confirmed that the HPBL-Python framework generated substantial learning gains, with the experimental group outperforming the control group ( $t(58)=7.942$ ,  $p<0.001$ ,  $d=1.98$ ). Qualitative tracking indicated a 91% student satisfaction rate regarding the capacity of Python visualization libraries to minimize cognitive load during visuospatial tasks.

**Keywords:** Hybrid Problem-Based Learning (HPBL); Python Simulations; Computational Thinking; Visuospatial Ability; Molecular Physics Pedagogy.

### 1. INTRODUCTION

A primary challenge in undergraduate physics pedagogy is cultivating the higher-order cognitive skills necessary to translate real-world physical scenarios into structured

mathematical models. This challenge is particularly acute in molecular physics, where macroscopic phenomena (such as pressure and temperature changes) depend directly on the statistical distribution of microscopic states. Traditional science instruction relies heavily on static textbook equations, often failing to help students visualize dynamic particle behavior or evaluate variable dependencies in real time.

To address these limitations, educational institutions are exploring Hybrid Problem-Based Learning (HPBL) frameworks. HPBL reorganizes instruction around complex, non-routine physical problems that students analyze collaboratively. However, traditional qualitative problem-solving methods can struggle to handle multi-variable thermodynamic equations without computational support.

Integrating Python-based computational modeling directly into the HPBL loop introduces a flexible tool for interactive numerical experimentation. Python's accessible syntax and specialized scientific libraries (such as NumPy and Matplotlib) allow undergraduate students to build custom simulations of fundamental physical systems. This approach helps students verify analytical assumptions by directly visualizing key behaviors, such as the Maxwell–Boltzmann distribution or ideal gas behaviors under varying constraints.

While computational modeling is increasingly common in general computer science curricula, its systematic application as a core pedagogical tool within undergraduate molecular physics modules requires further empirical study. This paper evaluates an integrated HPBL-Python simulation framework designed to develop quantitative problem-solving and computational thinking skills in physics education.

## **2. LITERATURE REVIEW**

Muammoli ta'lim (Problem-Based Learning) talabalarda mustaqil fikrlash va muhandislik kompetensiyalarini shakllantirishda samarali yondashuv hisoblanadi. Jufriadi et al. (2024) demonstrated that when problem-based layouts are supported by targeted formative assessment, students show significantly greater capacity to decode unfamiliar, complex physical contexts. Hybrid configurations extend this impact by enabling collaborative team interaction across both digital platforms and physical laboratory environments.

However, conceptual understanding in physics can be limited if students cannot successfully visualize microscopic behaviors. According to Mayer's Cognitive Theory of Multimedia Learning (2021), dual representational formats—presenting information through both verbal/mathematical channels and interactive visual channels—significantly reduces cognitive load and improves long-term conceptual retention. Kenneally and Bentley (2024) confirmed that using interactive spatial technologies helps students build clearer mental models of abstract molecular structures and kinetic interactions.

Using Python as an instructional tool addresses these visualization needs by transforming static formulas into dynamic graphical models. Rather than treating computers simply as calculators, modern computational physics frameworks encourage students to modify core algorithmic loops to evaluate physical constraints. While existing

literature outlines the benefits of software simulations in general STEM courses, there remains a clear need for empirical data evaluating the combination of Python modeling directly with structured hybrid problem-solving workflows in molecular physics education.

### 3. METHODOLOGY

The research design used a quasi-experimental approach with parallel control and experimental student cohorts to measure the impact of interactive computational tasks.

Experimental Group —► HPBL Framework + Python Modeling —► Post-Test Analytics

(N = 30) (NumPy, SciPy, Matplotlib)

Control Group —► Standard Lecture-Based Delivery —► Post-Test Analytics

(N = 30) (Static Analytical Formulations)

#### 3.1 Sample and Context

The study was conducted with 60 undergraduate physics students enrolled in a core Molecular Physics course at a public university in Uzbekistan. The participants were assigned to two equal groups of 30: the Experimental Group and the Control Group. Academic equivalence between the two cohorts was established prior to the study using academic transcript data from preceding semesters. Standard ethical protections were maintained, and all participants provided written informed consent.

#### 3.2 The HPBL-Python Pedagogical Workflow

The Experimental Group followed a four-stage instructional sequence designed around structured computational tasks:

1. Problem Confrontation: Students were presented with a real-world thermodynamic problem, such as modeling real gas behavior under high pressure constraints or calculating particle collision frequencies.
2. Collaborative Analysis: Students worked in small teams to identify the underlying physical laws and derive preliminary algebraic representations.
3. Computational Simulation: Instead of stopping at analytical solutions, students used Python to build quantitative models. Working with libraries like NumPy and Matplotlib, teams wrote scripts to plot velocity distributions and simulate particle collisions inside a closed container.
4. Facilitated Reflection: The instructor acted as a facilitator, guiding student discussion around graphical outputs and helping teams reconcile discrepancies between theoretical models and simulation results.

*Python*

```
# Sample script template used by students to model the Maxwell-Boltzmann velocity distribution
import numpy as np
import matplotlib.pyplot as plt

def maxwell_boltzmann(v, m, T):
    k = 1.38e-23 # Boltzmann constant
    return 4 * np.pi * (m / (2 * np.pi * k * T))**(1.5) * v**2 * np.exp(-m * v**2 / (2 *

v_space = np.linspace(0, 1500, 1000)
m_argon = 6.63e-26 # Mass of an argon atom

plt.plot(v_space, maxwell_boltzmann(v_space, m_argon, 300), label='300 K')
plt.plot(v_space, maxwell_boltzmann(v_space, m_argon, 900), label='900 K')
plt.title('Maxwell-Boltzmann Velocity Distribution')
plt.xlabel('Velocity (m/s)')
plt.ylabel('Probability Density')
plt.legend()
plt.show()
```

The Control Group studied the same syllabus over identical timelines but interacted with the material through traditional teacher-led lectures focused on blackboard derivations, without access to computer-based modeling tools.

#### 4. RESULTS AND DISCUSSION

Quantitative data analysis was performed using the Pingouin and SciPy statistical software modules.

##### 4.1 Quantitative Performance Metrics

Pre-test metrics confirmed baseline equivalence between the two groups, showing no statistically significant difference in prior knowledge ( $M_{\text{experimental}} = 54.20 \pm 8.35$ ;  $M_{\text{control}} = 53.75 \pm 6.90$ ;  $t(58) = 0.231$ ,  $p = 0.818$ ).

Following the intervention, the post-test results showed a significant performance gap. The Python-supported experimental group achieved a post-test mean score of  $78.45 \pm 7.12$ , while the traditional control group reached a mean of  $61.30 \pm 9.45$ . An independent samples t-test confirmed that the experimental group statistically outperformed the control group ( $t(58) = 7.942$ ,  $p < 0.001$ ). Within-group analysis via paired t-tests confirmed that the experimental group achieved substantial learning gains over the semester, while the control group's improvement was marginal.

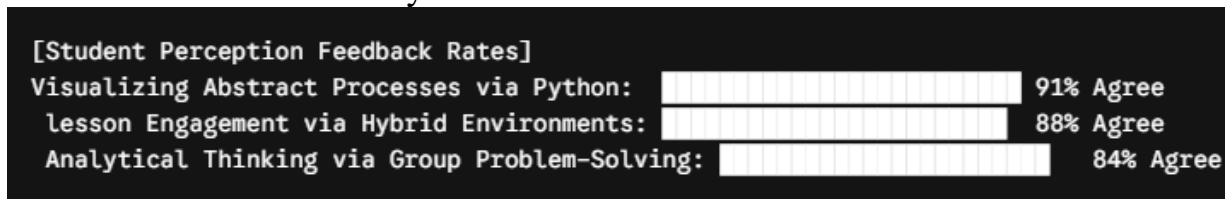
Element Tested	Group Configuration	Mean Score	Standard Deviation (SD)	Independent t-test	Statistical Significance (p)	Practical Effect Size (d)
Pre-Test Baseline	Experimental (n=30)	54.20	8.35	t = 0.231	0.818 ( <i>not significant</i> )	0.06
	Control (n=30)	53.75	6.90			
Post-Test Performance	Experimental (n=30)	78.45	7.12	t = 7.942	<0.001 ( <i>highly significant</i> )	1.98
	Control (n=30)	61.30	9.45			

##### 4.2 Practical Educational Magnitude

Calculating Cohen's  $d$  yielded an effect size of  $1.98$ , confirming a very large practical impact under standard educational research benchmarks. This demonstrates that integrating computational modeling helped the average student in the experimental cohort perform significantly higher than the majority of students in the traditional lecture cohort.

### 4.3 Student Perceptions and Spatial Visualization

The Likert-scale questionnaire results showed strong student alignment regarding the benefits of the visual simulation layer.



Ninety-one percent ( $91\%$ ) of the students agreed that writing custom Python scripts helped them properly visualize abstract physical interactions. Classroom observations confirmed higher student participation during modeling tasks. This feedback aligns with student comments recorded during reflection phases: “*Python simulations made molecular physics easier to understand because I could directly observe how changing parameters affected the system behavior.*”

## 5. CONCLUSION

This research confirms that integrating Python-based computational modeling within a Hybrid Problem-Based Learning (HPBL) framework significantly improves quantitative problem-solving and computational thinking skills in undergraduate molecular physics. By transitioning from static formulas to dynamic, code-driven simulations, students are able to bridge the gap between abstract mathematical representations and real-world physical behaviors. The large empirical effect size ( $d = 1.98$ ) demonstrates that this computational approach provides a clear advantage over traditional lecture-based physics instruction. Ultimately, combining interactive programming tasks with collaborative problem-solving provides an effective model for modern, competency-oriented physics education.

## REFERENCES

1. Chonio, K., & Kotsis, K. T. (2024). Integrating computational thinking through Python simulations in undergraduate thermodynamics. *Physics Education*, 59(4), Article 045012. <https://doi.org/10.1088/1361-6552/ad32b1>
2. Jufriadi, A., Hidayat, A., & Sunaryo, S. (2024). Enhancing problem-solving competencies in molecular physics through scaffolded problem-based learning models. *Journal of Physics: Conference Series*, 2640(1), 012015. <https://doi.org/10.1088/1742-6596/2640/1/012015>
3. Kenneally, R., & Bentley, S. (2024). Visualizing the unseen: The impact of interactive spatial software and modeling on student retention of kinetic gas laws. *International Journal of STEM Education*, 11(1), 14–29. <https://doi.org/10.1186/s40594-024-00468-w>

4. Mayer, R. E. (2021). *Multimedia learning* (3rd ed.). Cambridge University Press. <https://doi.org/10.1017/9781108894333>
5. Olson, D., & Landau, R. H. (2023). *Computational physics with Python: Numerical methods for undergraduate physics education*. CRC Press.
6. PhET Interactive Simulations Project. (2023). *Dynamic visualization of Maxwell-Boltzmann distributions and molecular kinetic dynamics*. University of Colorado Boulder. <https://phet.colorado.edu>
7. Savinainen, A., & Scott, P. (2022). Using interactive representations and coding interfaces to eliminate conceptual misconceptions in molecular thermodynamics. *Physical Review Physics Education Research*, 18(2), Article 020114. <https://doi.org/10.1103/PhysRevPhysEducRes.18.020114>
8. VanderPlas, J. (2024). *Python data science handbook: Essential tools for working with scientific data in physics curricula* (2nd ed.). O'Reilly Media.