

КОНСТРУКТИВНО-ПРОДУКЦІЙНЕ МОДЕЛЮВАННЯ ГРОЗОВОГО ФРОНТУ З ВИКОРИСТАННЯМ ГЕНЕТИЧНОГО АЛГОРИТМУ

Анотація. Розроблені конструктори у складі мультиконструктора призначені для реалістичного моделювання фронту грозової (блискавкової) активності. Особливості конструювання полягають в застосуванні мультиконструктору, що є оболонкою для інших конструкторів, та параметричного конструктору, що надає можливість багаторазового застосування одного конструктору з різними початковими даними (формальними параметрами) в одному проєкті. Моделювання фрактальних часових рядів, які визначають спалахи блискавок, виконано з використанням генетичного алгоритму. Особливість якого полягає у кодуванні хромосом з текстовою та числовою частинами. Реалізація конструктору виконана в удосконаленому універсальному середовищі конструювання «Конструктор 2.0». Застосування запропонованих моделей грозового фронту може бути основою для досліджень зі забезпечення пожежної безпеки, безпеки цивільних та військових об'єктів й паркових та лісових масивів.

Ключові слова: конструктивно-продукційне моделювання, часовий ряд, фрактал, грозовий фронт, конструктор, формальні граматики, генетичний алгоритм, програмне забезпечення, інформаційні технології.

В основі дослідження закладено використання засобів конструктивно-продукційного моделювання [1, 2].

Розглянуто відео грозового фронту [3], отримане супутником NASA. Дані з супутника були оброблені та відокремлені активності блискавок [4, 5, 6].

Моделювання грозового фронту [4, 5, 6, 7] було трансформовано у схему з декількох послідовно зв'язаних конструкторів: знаходження моделей фрактальних часових рядів у вигляді закодованих хромосом генетичним алгоритмом, перетворення хромосом у часові ряди, та візуалізації грозового фронту.

Конструювання виконується в універсальній програмній системі для моделювання конструкторів «Конструктор 2.0». Вона представляє собою продовження ідей системи моделювання «Конструктор 1.1» [8].

У середовищі «Конструктор 2.0» передбачені наступні конструктори:

- автономний – базовий, що містить у собі усю необхідну інформацію для отримання реалізації у вигляді конструкції чи конструктивного процесу;
- алгоритмічний – спеціалізується на реалізації алгоритмів;

- мульти – складається з конструкторів та поєднує їх у послідовно зв'язану систему.

Та наступні підвиди конструкторів:

- простий – базова форма конструктора;
- параметричний – потребує на вхід значень від іншого конструктору чи зовнішнього виконавця.

Пов'язані роботи. Дослідження ґрунтується на використанні конструктивно-продукційного підходу [8], що був використаний у різних дослідженнях.

Основний матеріал моделювання грозового фронту [4, 5] було розглянуто як послідовність конструкторів для представлення блискавок [9] та їх атрибутів.

При дослідженні використовується генетичний алгоритм [10, 11, 12, 13], що дозволяє ітеративним процесом виконати пошук значень для відтворення процесу, зафіксованого супутниками.

Модель грозового фронту складається з сукупності фрактальних часових рядів, правила яких визначаються генетичним алгоритмом з використанням як у L-системах [14]

Мета дослідження. Метою дослідження є розробка конструктивно-продукційної моделі грозового фронту (фронту блискавок), максимально наближеного до отриманих супутником NASA відео-зображення [3]. У попередній роботі [4] виконана розкадровка відео зі супутника NASA та методом підбору знайдена модель, близька до оригіналу. У цій роботі поставлена задача відшукати більш точну модель з використанням алгоритму направленої випадкової пошуку – генетичного алгоритму.

Викладення основного матеріалу. Попередня обробка відео-зображення складалася з його розкадровки, виявленні на окремих кадрах спалахів блискавок, моделюванні спалахів у вигляді кіл, діаметри яких пропорційні їх потужностям [4]. На зведеному зображенні з усіх кадрів чітко відслідковується дві криві вздовж яких відбуваються спалахи блискавок. Ці криві змодельовані у вигляді кривих Безьє.

Таким чином для вирішення поставленої задачі треба знайти шість часових рядів: для визначення точок спалаху вздовж та від кривої Безьє й сили спалаху для кожної з двох кривих.

Моделі часових рядів шукаються у вигляді фрактальних L-систем: правил підстановки та початкового символічного рядку, математичного очікування та дисперсії зміни значення наступної точки ряду до попередньої. Пошук виконується генетичним алгоритмом. Схожість ряду до реального визначається за середньоквадратичним відхиленням значень.

Для реалізації поставленої задачі була розроблена нова версія програмного середовища «Конструктор 2.0». У ній була розроблена низка конструкторів.

Основною конструювання є узагальнений конструктор [1, 2]:

$$C = \langle M, \Sigma, \Lambda \rangle,$$

де M – неоднорідний розширюваний носій, Σ – сигнатура відношень та пов'язаних з ними операцій, Λ – інформаційне забезпечення конструювання: призначення, умови початку та завершення конструювання, правила підстановки та обмеження.

У результаті спеціалізації узагальненого конструктора спочатку були розроблені наступні конструктори:

- параметричний конструктор «Генетичний пошук» призначений для формування чергового покоління хромосом та визначення їх якості. Параметрами цього конструктору є часовий ряд реального грозового фронту, який моделюється, та параметри генетичного пошуку – гранична кількість поколінь, коефіцієнт мутацій та схрещувань та допустима межа функції допасованості (фітнес функції);
- алгоритмічний конструктор «Генетичні алгоритми», призначений для інтерпретації операцій з конструктору «Генетичний пошук».

Спеціалізація конструктору (у даному випадку конструктору «Генетичний пошук») визначається як:

$$C = \langle M, \Sigma, \Lambda \rangle_s \mapsto_s C(rts, lts) = \langle M_s, \Sigma, \Lambda_s \rangle,$$

де M_s – включає у собі множину терміналів, нетерміналів, проміжні форми, та інформаційне забезпечення Λ_s – додатково до Λ містить множину змінних й операцій та їх опис. Параметри конструктору: rts – реальний часовий ряд, для якого шукається модель, максимально наближена до нього, lts – кількість елементів ряду rts .

Визначимо предметну область конструктору як формування хромосом, послідовність їх поколінь з застосування схрещувань й мутацій, та визначенням якості хромосом (за функцією допасованості).

Множина змінних згідно Λ_s :

- *acsioma* – початковий символічний рядок;
- *chromosome_list* – список хромосом поточного покоління;
- *iter* – кількість поколінь;
- *chromosome_count* – кількість хромосом в поколінні;
- *efficiency_coef* – допустима межа функції допасованості;
- *mutation_prob* – кількість операцій мутації;
- *crossover_prob* – кількість операцій схрещування;
- *best_chromosome* – найкраща хромосома;
- *compare_data* – дані реального часового ряду для визначення порівняльної якості хромосом.

Множина операцій згідно Λ_s , де:

- $\triangleleft(list, count, signature, restrictions)$ – формування покоління хромосом, де *list* – список хромосом, *count* – кількість хромосом в поколінні, *signature* – позначення шуканих терміналів, *restrictions* – обмеження, які складаються з мінімальних та максимальних значень математичного очікування та дисперсії зміни значення насту-

пної точки ряду до попередньої, кількості символів та множини можливих символів формування початкового символного рядка та правила підстановки, мінімального та максимального значення кількості ітерації;

- $\prec(list, list, count)$ – мутації хромосом, де $count$ – кількість операцій;
- $\succ(list, list, count)$ – схрещування хромосом;
- $\triangleright(list, list, data, coef)$ – операція визначення допасованості (фітнес функції), де $data$ – реальний часовий ряд для порівняння, $coef$ – допустима межа функції допасованості;
- $*(chromosome, list)$ – визначення найякіснішої хромосоми, де $chromosome$ – визначена хромосома.

Для інтерпретації, розроблено алгоритмічний конструктор «Генетичні алгоритми» з визначеними алгоритмами для формування хромосом – мутація, схрещування, фітнес-функція.

Алгоритми над даними визначаються як $A \Big|_X^Y$, де X – вхідні дані, Y – вихідні дані.

Визначили алгоритми виконання операцій:

- $A_1 \Big|_{list, count, signature, restrictions}^{list}$ – формування хромосом в покоління;
- $A_2 \Big|_{list, count}^{list}$ – виконання операції схрещування;
- $A_3 \Big|_{list, count}^{list}$ – виконання операції мутації;
- $A_4 \Big|_{list, data, coef}^{list}$ – визначення допасованості хромосом;
- $A_5 \Big|_{list}^{chromosome}$ – визначення найякіснішої хромосоми;
- $A_6^0 \Big|_{l_j, l_h, l_q}^{l_i}$ – підстановка;
- $A_7^0 \Big|_{l_j, \psi}^{l_i}$ – частковий вивід;
- $A_8^0 \Big|_{l_j, \psi}^{l_i}$ – повний вивід.

На мові програмування Python реалізовані ці алгоритми.

Інтерпретація конструктору «Генетичний пошук» за алгоритмічним конструктором «Генетичні алгоритми» визначається як

$$\langle C_S(rts, lts) = \langle M_S, \Sigma, \Lambda_S \rangle, C_A = \langle M_A, \Sigma_A, \Lambda_A \rangle \rangle_I \mapsto_I C_S(rts, lts) = \langle M_{SI}, \Sigma_{SI}, \Lambda_{SI}, Z \rangle,$$

де C_A – алгоритмічний конструктор, M_A – неоднорідний розширюваний носій алгоритмічного конструктору, Σ_A – набір операцій над символами, та інформаційне забезпечення алгоритмічного конструктору, Λ_A – інформаційне забезпечення конструювання алгоритмічного конструктору, Σ_{SI} що включає у собі сигнатуру операцій підстановки,

часткового та повного виводу, та інформаційне забезпечення Λ_{SI} , що включає множини операцій та алгоритмів.

Поєднали алгоритми за операціями (кожен алгоритм є атрибутом деякої операції): $\{A_1 \mid_{list, count, signature, restrictions}^{list} \vdash \triangleleft\}$, $\{A_2 \mid_{list, count}^{list} \vdash \triangleleft\}$, $\{A_3 \mid_{list, count}^{list} \vdash \triangleright\}$, $\{A_4 \mid_{list, data, coef}^{list, chromosome} \vdash \triangleright\}$, $\{A_5 \mid_{list}^{chromosome} \vdash *\}$, $\{A_6^0 \mid_{l_j, l_h, l_q}^{l_i} \vdash \Rightarrow\}$, $\{A_7^0 \mid_{l_j, \psi}^{l_i} \vdash \mid \Rightarrow\}$, $\{A_8^0 \mid_{l_j, \psi}^{l_i} \vdash \parallel \Rightarrow\}$.

При інтерпретації конструктори «Генетичний пошук» та «Генетичні алгоритми» поєднуються, та після визначення перетворень формується конструктивна система, що здатна виконувати пошук значень на основі реальних даних. Наявність параметрів конструктору «Генетичний пошук» дало можливість застосувати цей конструктор для відновлення шести часових рядів не змінюючи всі складові конструктору.

Конструювання виконується внутрішнім та зовнішнім виконавцем/виконавцями. Внутрішнім виконавцем у даному дослідженні виступає розроблений програмний застосунок «Конструктор 2.0».

Зовнішнім виконавцем виступає користувач, що формує базові компоненти конструктора. Інтерфейсом зовнішнього виконавця виступає інтерфейс програмного застосунку, який має у собі форми для визначення уточнюючих перетворень.

Конкретизація конструктору визначається як

$$C_{SI}(rts, lts) = \langle M_{SI}, \Sigma_{SI}, \Lambda_{SI} \rangle_K \mapsto_K C_{SI}(rts, lts) = \langle M_{SIK}, \Sigma_{SIK}, \Lambda_{SIK} \rangle,$$

де M_{SIK} – розширяємий носій що містить визначений набір термінальних та нетермінальних символів, Σ_{SIK} – множина операції формування рядка, Λ_{SIK} – інформаційне забезпечення процесу конструювання.

При конкретизації визначимо початкові значення даних, як початкова символна послідовність, та задамо правила підстановки.

Носій M_{SIK} включає нетермінальний символ B та термінальний символ $list$.

Сигнатура операцій $\Sigma_{SIK} = \Sigma_{SI}$.

Множина правил підстановки визначається як

$$\psi = \langle s, g \rangle \in \Psi,$$

де s – відношення підстановки, g – набір операцій над атрибутами.

Визначається при конкретизації Λ_K наступне:

- мета конструювання — знаходження хромосоми, у якій закодовано найкращий модельний ряд;
- обмеження — максимальна кількість ітерацій (поколінь) $iter = 1000$;
- початкові умови — початкова аксіома $acsioma = "B"$, набір хромосом $chromosome_list$ не містить хромосом;
- правила підстановки:

$$\psi_1 = \left\langle s_1 = \langle B \rightarrow list\ B \rangle, g_1 = \left\langle \begin{array}{l} \triangleleft (list, count, signature, restrictins), \\ \prec (list, count), \succ (list, count), \\ \triangleright (list, data, coef) \end{array} \right\rangle \right\rangle$$

$$\psi_2 = \langle s_2 = \langle B \rightarrow \varepsilon \rangle, g_2 = \langle *(chromosome, list) \rangle \rangle$$

- умова завершення — виконання кількості ітерацій *iter*.

Після визначення спеціалізації, інтерпретації та конкретизації, конструктивна система $C_{SIK}(rts, lts)$ готова для реалізації.

Реалізація конструктору визначається як

$$C_{SIK}(rts, lts) = \langle M_{SIK}, \Sigma_{SIK}, \Lambda_{SIK} \rangle_R \mapsto \Omega(C(rts, lts)),$$

де $\Omega(C_{SIK}(rts, lts))$ – конструкція породжувана конструктором $C_{SIK}(rts, lts)$ у даному випадку результуючою конструкцією є найліпша знайдена хромосома, у якій закодовано інформація для відтворення часового ряду, наближеного до реального заданого параметрами.

Для інтеграції результатів конструктивної системи $C_{SIK}(rts, lts)$ й формування та відображення шести модельних часових рядів, розроблені наступні автономні конструктори:

- розрахунку точок на основі хромосом «Грозовий фронт»;
- відображення блискавок у вигляді грозового фронту «Графічне відображення».

Ці конструктори інтерпретуються відповідними алгоритмічними конструкторами:

- розрахунку точок «Часовий ряд»;
- формування графічного відображення «Графічні алгоритми».

Всі розроблені конструктори об'єднані в мультиконструкторі «Моделювання грозового фронту», як наведено на рис. 1.

Зв'язок між конструкторами у складі мультиконструктора може бути забезпечений трьома механізмами:

- передача даних через параметри від зовнішнього виконавця або засобами мультиконструктора;
- наявність в одному конструкторі серед початкових умов конструювання результату реалізації іншого конструктора;
- наявність у відповідному алгоритмічному конструкторі алгоритму, який виконує реалізацію іншого конструктору (не того з яким він поєднаний в конструктивну систему).

У даному випадку, зв'язок між конструкторами «Генетичний пошук» та «Грозовий фронт» й «Грозовий фронт» та «Графічне відображення» виконаний через передачу конструкцій в якості початкових умов. Передача забезпечується засобами мультиконструктора.

Послідовність виконання конструкторів та передача даних конструювання виконується у мультиконструкторі «Моделювання грозового фронту» представлено схемою конструкторів (рис.1).

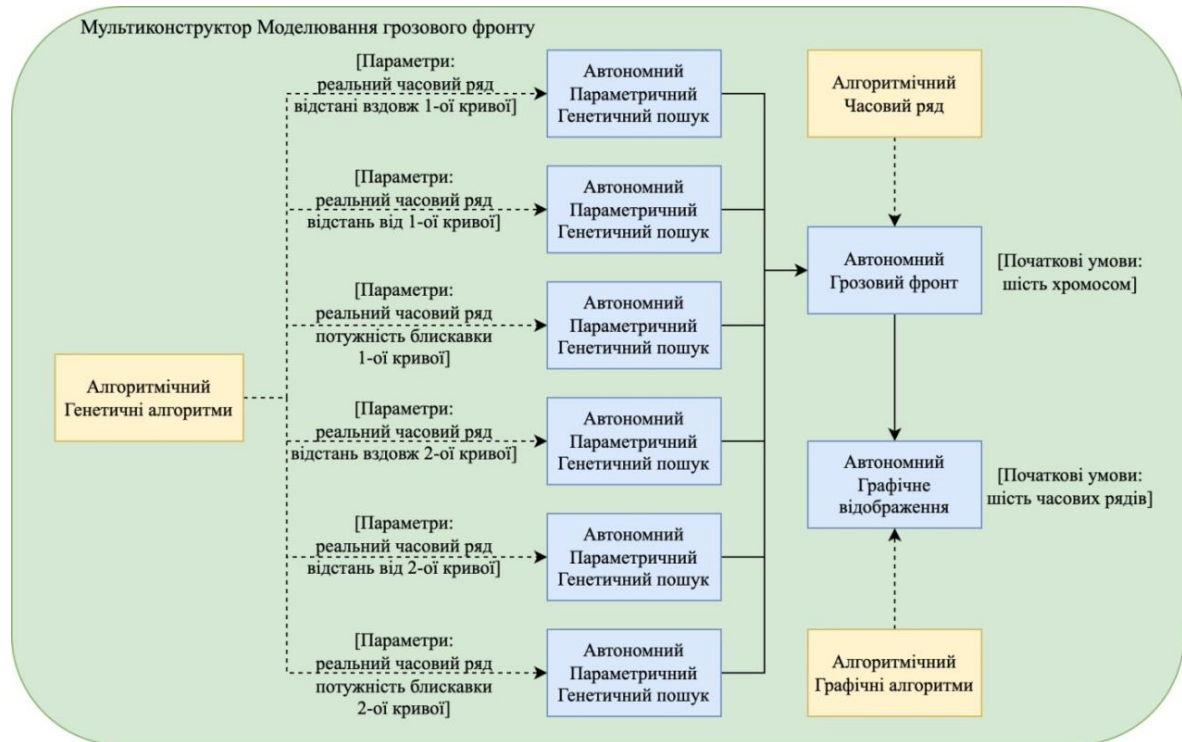


Рисунок 1 - Схема конструкторів

Спочатку знаходяться моделі шести часових рядів, закодованих у відповідних хромосомах, засобами конструктору «Генетичний пошук».

Конструкції $\Omega(C(rts, lts))$ є результатом конструювання конструктору «Генетичний пошук» та початковою умовою конструктору «Грозовий фронт».

Конструктор «Грозовий фронт» використовує шість відповідних хромосом для генерації фрактальних часових рядів та відповідних точок, що відповідають графічному відображенню фронту.

Конструктором «Графічне відображення» формується зображення блискавок у вигляді кіл у визначених часовими рядами точках та діаметрами кіл, що відображає потужності спалаху.

Одна з реалізацій мультиконструктору «Моделювання грозового фронту» моделі грозового фронту приведена на рис. 2 (без анімації, як зведене зображення з усіх кадрів).

Отриманий результат (рис.2, с) був порівняний зі згрупованими блискавками (рис.2, b), отриманими після покадрової обробки оригінального відеофайлу (рис.2, а).

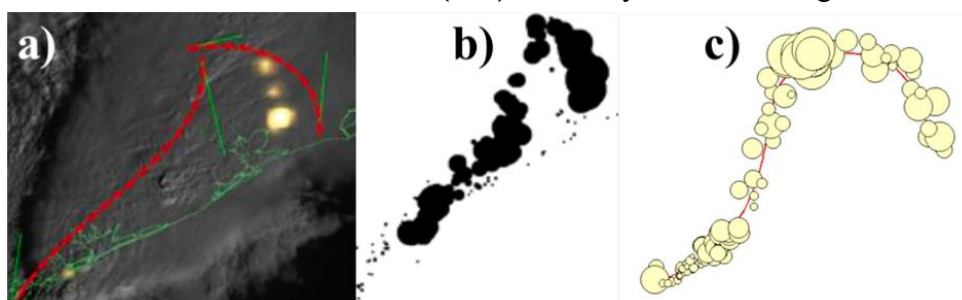


Рисунок 2 - Модель грозового фронту

Результатом пошуку значень конструктору отримано хромосоми зазначеної структури, що представляють собою основні характеристики конструювання блискавок, як наприклад значення для потужності блискавок, як $acsioma = f$, $rules = \{f \rightarrow -f + f\}$ (правило для формування фрактального часового ряду), $iter = 8$, $mat_oz_y = 1$, $disp_y = 3$, $mat_oz_y_update = 23$.

Порівняння реального та модельного рядів наведено на рис. 3. Позитивні та негативні значення потужності слугують для відображення блискавок з однієї чи іншої сторони кривої Безьє.

Висновки. Застосування запропонованих моделей грозового фронту може бути основою для досліджень зі забезпечення пожежної безпеки, безпеки цивільних та військових об'єктів й паркових та лісових масивів.

Розроблені конструктори у складі мультиконструктора дозволяють достатньо реалістично моделювати фронт грозової активності.

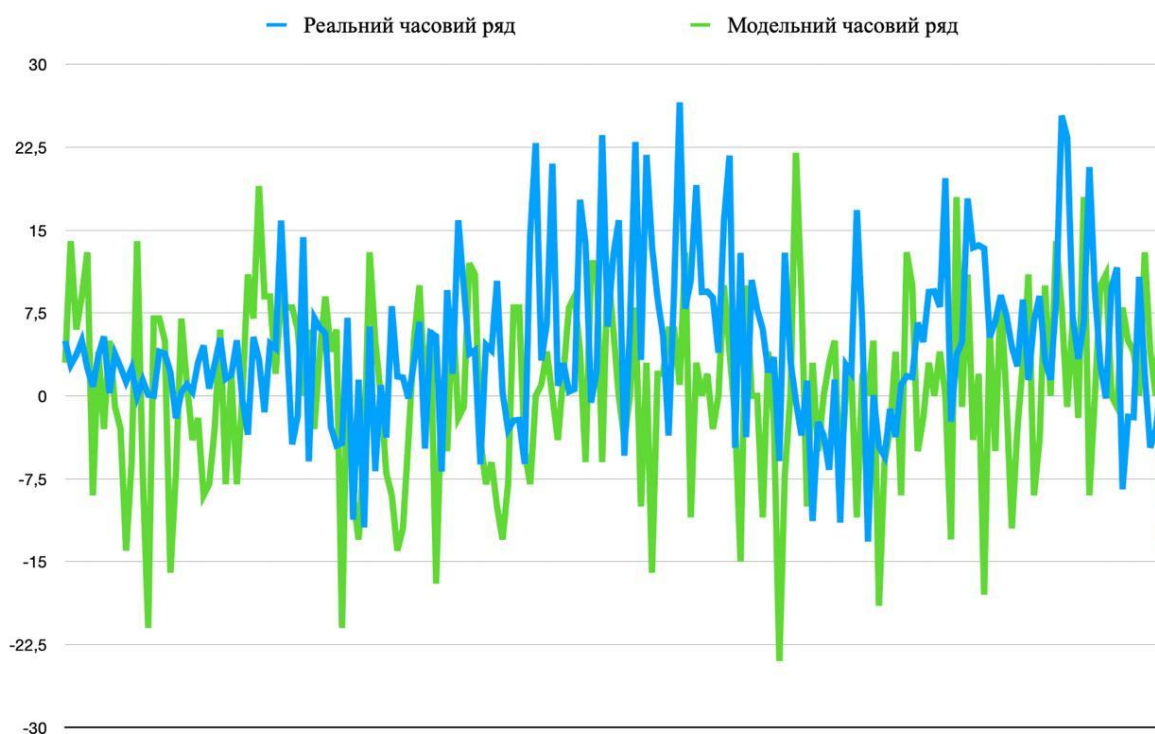


Рисунок 3 - Реальний та модельний ряд потужності блискавки

Особливості конструювання полягають в застосуванні мультиконструктору, що є оболонкою для інших конструкторів, та параметричного конструктору, що надає можливість багаторазового застосування одного конструктору з різними початковими даними (формальними параметрами) в одному проєкті.

Моделювання фрактальних часових рядів, які визначають спалахи блискавок, виконано з використанням генетичного алгоритму. Особливість якого полягає у кодуванні хромосом з текстовою та числовою частинами.

Реалізація конструктору виконана в удосконаленому універсальному середовищі конструювання «Конструктор 2.0».

ЛІТЕРАТУРА / REFERENCES

1. Shynkarenko V. I., Ilman V. M. Constructive-Synthesizing Structures and Their Grammatical Interpretations. Part I. Generalized Formal Constructive-Synthesizing Structure // *Cybernetics and Systems Analysis*. – 2014. – Vol. 50, No. 5. – P. 662–665. – DOI: 10.1007/s10559-014-9655-z.
2. Shynkarenko V. I., Ilman V. M. Constructive-Synthesizing Structures and Their Grammatical Interpretations. Part II. Refining Transformations // *Cybernetics and Systems Analysis*. – 2014. – Vol. 50, No. 6. – P. 829–841. – DOI: 10.1007/s10559-014-9674-9.
3. Killian M. GOES-16 Satellite Returns First Lightning-Mapping Images Like Never Seen Before : [Electronic resource] // *AmericaSpace*. – 07.03.2017. – <https://www.americaspace.com/2017/03/07/goes-16-satellite-returns-first-lightning-mapping-images-like-never-seen-before/> (available on: 26.04.2025)
4. Shynkarenko V., Lytvynenko K., Chyhir R., Nikitina I. Modeling of lightning flashes in thunderstorm front by constructive production of fractal time series // *Conference on Computer Science and Information Technologies*. – Cham : Springer International Publishing, 2019 (September). – P. 173–185.
5. Shynkarenko V., Lytvynenko K., Chyhir R., Sansiieva I. Constructive modeling of lightning activity in thunderstorm front // *2019 IEEE 14th International Conference on Computer Sciences and Information Technologies (CSIT)*. – 2019 (September). – Vol. 1. – P. 92–95.
6. Chyhir R., Nikitina I. Modeling of Lightning Flashes in Thunderstorm Front by Constructive // *Advances in Intelligent Systems and Computing IV: Selected Papers from the International Conference on Computer Science and Information Technologies, CSIT 2019, September 17–20, 2019, Lviv, Ukraine*. – 2019 (November). – Vol. 1080. – P. 173. – Springer Nature.
7. Shynkarenko V., Nikitina I., Chyhir R. Constructive-Synthesizing Modeling of Lightning Flashes in the Dynamic Thunderstorm Front // *Conference on Computer Science and Information Technologies*. – Cham : Springer International Publishing, 2020 (September). – P. 1128–1145.
8. Shynkarenko V.I., Chyhir R.R. Constructive-Synthesizing Modelling of Multifractals Based on Multiconstructors // *Proceedings of the 14th International Scientific and Practical Programming Conference (UkrPROG 2024)*. – 2024. – Vol. 3806. – P. 75–88.
9. Rakov V.A., Uman M.A. Lightning: physics and effects. – Cambridge : Cambridge University Press, 2003. – 687 p.

10. Forrest S. Genetic algorithms // ACM computing surveys (CSUR). – 1996. – Vol. 28, No. 1. – P. 77–80.
11. Mirjalili S., Mirjalili S. Genetic algorithm // Evolutionary algorithms and neural networks: Theory and applications. – 2019. – P. 43–55.
12. Hassanat A., Almohammadi K., Alkafaween E. A., Abunawas E., Hammouri A., Prasath V. S. Choosing mutation and crossover ratios for genetic algorithms—a review with a new dynamic approach // Information. – 2019. – Vol. 10, No. 12. – P. 390.
13. Yang J., Wu C., Lee H. P., Liang Y. Solving traveling salesman problems using generalized chromosome genetic algorithm // Progress in Natural Science. – 2008. – Vol. 18, No. 7. – P. 887–892.
14. Prusinkiewicz P., Lindenmayer A. The Algorithmic Beauty of Plants. – New York : Springer Science & Business Media, 2012. – 228 p.

Received 19.05.2025.
Accepted 21.05.2025.

Constructive-synthesizing modeling of a thunderstorm front using a genetic algorithm

The aim of the study is to develop constructive-synthesizing model of a thunderstorm front (lightning front) that is as close as possible to the video images obtained by the NASA satellite. In the previous work, a storyboard of video from the NASA satellite was made and a model close to the original was found by the method of fitting. In this paper, the task is to find a more accurate model using a directed random search algorithm — a genetic algorithm.

The developed constructors as part of the multiconstructor allow for a fairly realistic modeling of the storm front. The parametric constructor «Genetic Search» searches for time series models in the form of encoded chromosomes. The autonomous constructor «Thunderstorm Front» transforms chromosomes into time series. The constructor «Graphical Display» displays the lightning activity front.

The design features include the use of a multiconstructor, which is a shell for other constructors, and a parametric constructor, which allows the multiple use of one constructor with different initial data (formal parameters) in one project.

The connection between the constructors within the multiconstructor is provided by the following mechanisms: data transfer through parameters from an external contractor or by means of the multiconstructor; the presence in one constructor of the result of the implementation of another constructor among the initial constructor conditions.

Algorithmic constructors were used for the interpretation of the operations of the constructors: each constructor operation was matched with an algorithmic constructor algorithm.

Modeling of the time series that determine lightning flashes was performed using a genetic algorithm. The peculiarity of this algorithm is the encoding of chromosomes with text and numerical parts.

The constructor was implemented in the advanced universal design environment ‘Constructor 2.0’.

The use of the proposed storm front models can be the basis for research on fire safety, security of civilian and military facilities, park and forest areas.

Шинкаренко Віктор Іванович - д.т.н., професор кафедри «Комп'ютерні інформаційні технології» Українського державного університету науки і технологій.

Чигір Роберт Романович - аспірант кафедри «Комп'ютерні інформаційні технології» Українського державного університету науки і технологій.

Shynkarenko Viktor - Doctor of engineering's sciences, Professor of the Department of Computer Information Technologies at Ukrainian State University of Science and Technologies.

Chyhir Robert - PhD student in the Department of Computer Information Technologies at Ukrainian State University of Science and Technology.