

ỨNG DỤNG PYTHON VÀ STREAMLIT TRONG TÍNH TOÁN VÀ TRỰC QUAN HÓA ĐIỀU TIẾT LŨ HỒ CHỨA

Phạm Ngọc Thịnh, Nguyễn Phúc Hậu, Trần Khắc Thạc

Trường Đại học Thủy lợi

Nguyễn Hải Đăng

Viện Khoa học Thủy lợi miền Nam

Tóm tắt: Trong bối cảnh biến đổi khí hậu và yêu cầu hiện đại hóa công tác quản lý tài nguyên nước, việc tự động hóa quá trình tính toán điều tiết lũ hồ chứa trở nên cấp thiết. Nghiên cứu này trình bày việc phát triển một công cụ tính toán dựa trên ngôn ngữ Python, cho phép tự động hóa hoàn toàn quy trình điều tiết lũ từ nhập dữ liệu, thực hiện tính toán đến trực quan hóa và xuất báo cáo kết quả. Hai phương pháp kinh điển là Potapov và Thử dần được mã hóa thành các mô-đun độc lập, đảm bảo khả năng mở rộng, tái sử dụng và tích hợp với các hệ thống số hóa trong tương lai.

Dữ liệu đầu vào bao gồm đường quá trình lũ đến, quan hệ dung tích–cao trình mực nước của hồ và các thông số tràn. Chương trình thực hiện diễn toán theo phương trình cân bằng nước dạng rời rạc, sử dụng nội suy tuyến tính và vòng lặp tự động. Kết quả mô phỏng cho thấy sai số giữa chương trình và phương pháp đồ giải truyền thống nhỏ hơn 4%, chứng tỏ độ chính xác và ổn định cao của thuật toán. Giao diện web xây dựng bằng Streamlit cho phép người dùng lựa chọn phương pháp tính, quan sát đồ thị động của lưu lượng và mực nước, cũng như xuất báo cáo tự động.

Cách tiếp cận đề xuất giúp tăng hiệu suất tính toán, giảm sai sót thao tác và đảm bảo tính minh bạch nhờ cơ chế lưu trữ dữ liệu dạng DataFrame. Bên cạnh ứng dụng kỹ thuật, công cụ còn có ý nghĩa trong giảng dạy và quản lý vận hành hồ chứa, góp phần thúc đẩy chuyển đổi số trong thiết kế và quản lý công trình thủy lợi. Hướng phát triển tiếp theo sẽ mở rộng mô hình cho bài toán liên hồ và tích hợp trí tuệ nhân tạo nhằm tối ưu hóa vận hành theo thời gian thực.

Từ khóa: Điều tiết lũ; Phương pháp Potapov; Vận hành hồ chứa; Python; Streamlit; Mô hình thủy văn; Chuyển đổi số.

Summary: In the context of climate change and the increasing demand for modernization in water resources management, automating reservoir flood routing has become an essential task. This study presents the development of a Python-based computational tool that fully automates the flood routing process from data input and calculation to visualization and reporting. Two classical methods, Potapov and Trial-and-Error, were encoded as independent modules, enabling flexibility, reusability, and integration with future digital systems.

Input data include the inflow hydrograph, reservoir elevation–storage relationship, and spillway parameters. The program performs numerical routing using a discretized water balance equation, supported by interpolation functions and automated iteration loops. Results show that the computed peak outflow and maximum water level deviate by less than 4% compared to traditional graphical solutions, demonstrating high accuracy and reliability. The Streamlit-based web interface provides an intuitive environment for users to select methods, visualize dynamic discharge curves, and export reports automatically.

The proposed approach enhances computational efficiency, minimizes human error, and supports transparency through DataFrame-based data handling. Beyond technical applications, it serves as an effective educational and management tool, promoting digital transformation in hydraulic design and reservoir operation. Future developments will extend the model to multi-reservoir systems and integrate artificial intelligence for real-time optimization.

Keywords: Flood routing; Potapov method; Reservoir operation; Python automation; Streamlit; Hydrological modeling; Digital transformation.

1. GIỚI THIỆU

Trong bối cảnh biến đổi khí hậu ngày càng gia

tăng, các hiện tượng thời tiết cực đoan như mưa lớn, lũ quét và lũ ống đang xuất hiện với tần suất và cường độ cao hơn, gây áp lực lớn lên hệ thống hồ chứa và công trình điều tiết nước ở Việt Nam [1]. Việc vận hành hồ chứa không chỉ nhằm cắt giảm đỉnh lũ để bảo vệ hạ du mà còn phải đảm bảo mực nước phục vụ

Ngày nhận bài: 23/10/2025

Ngày thông qua phản biện: 30/12/2025

Ngày duyệt đăng: 13/01/2026

phát điện, cấp nước và phòng hạn trong mùa khô [2]. Thực tế cho thấy, công tác tính toán điều tiết lũ tại nhiều hồ chứa hiện nay vẫn chủ yếu được thực hiện bằng phương pháp thủ công hoặc bán tự động trên các bảng tính Excel. Quy trình này đòi hỏi kỹ sư phải lặp lại nhiều bước nhập dữ liệu, tra bảng và vẽ đồ thị, dễ xảy ra sai sót và tốn nhiều thời gian khi phải cập nhật nhiều kịch bản tính toán [3].

Trong xu thế chuyển đổi số của ngành thủy lợi, việc ứng dụng các ngôn ngữ lập trình hiện đại, đặc biệt là Python, mở ra hướng đi mới trong tự động hóa và số hóa công tác tính toán thủy công. Python sở hữu hệ sinh thái mạnh mẽ gồm các thư viện khoa học như NumPy, Pandas, SciPy, Matplotlib và Plotly, cùng các framework giao diện như Streamlit, giúp người dùng dễ dàng xây dựng các ứng dụng web tương tác phục vụ mô phỏng, thiết kế và đào tạo kỹ thuật [4, [5].

Trên thế giới, nhiều nghiên cứu đã phát triển và hoàn thiện các phương pháp tính toán điều tiết lũ, trong đó phương pháp Potapov được xem là một trong những công cụ kinh điển, đơn giản nhưng hiệu quả, thường được sử dụng để diễn toán quá trình lũ trong hồ chứa [6]. Bên cạnh đó, phương pháp Thử dần/lập cũng được áp dụng rộng rãi để tìm phương án vận hành tối ưu, đặc biệt trong trường hợp tràn có cửa van điều tiết. Một số phần mềm thương mại như HEC-HMS, MIKE 11 hay HEC-RAS 1D/2D đã tích hợp các mô-đun diễn toán lũ và vận hành hồ chứa; tuy nhiên, các công cụ này có chi phí cao hoặc khó tùy chỉnh, và ít phù hợp cho mục đích giảng dạy hoặc mở rộng thuật toán [7].

Tại Việt Nam, nhiều công trình nghiên cứu về hiện đại hóa và tự động hóa quản lý tài nguyên nước đã được thực hiện. Các hướng nghiên cứu mới tập trung vào việc ứng dụng mô hình học máy, trí tuệ nhân tạo và các nền tảng mã nguồn mở trong thủy lợi, như dự báo mực nước, hỗ trợ vận hành trạm thủy lợi và mô phỏng diễn toán lũ [8, 9]. Tuy nhiên, vẫn còn

thiếu các công cụ mở, linh hoạt và dễ sử dụng dựa trên mã nguồn Python, có khả năng phục vụ song song cho đào tạo, nghiên cứu và thiết kế kỹ thuật.

Xuất phát từ thực tiễn đó, nghiên cứu này hướng tới xây dựng một chương trình Python tự động thực hiện toàn bộ quy trình điều tiết lũ hồ chứa, bao gồm nhập dữ liệu, tính toán, mô phỏng và trực quan hóa kết quả. Chương trình mã hóa hai phương pháp Potapov và Thử dần, đồng thời tích hợp giao diện web bằng Streamlit, cho phép người dùng thao tác dễ dàng mà không cần kiến thức lập trình [4, 5]. Mục tiêu cuối cùng là góp phần hiện đại hóa và số hóa công tác thiết kế – vận hành công trình thủy lợi, phù hợp với định hướng phát triển bền vững và chuyển đổi số ngành thủy lợi Việt Nam.

2. PHƯƠNG PHÁP NGHIÊN CỨU

2.1. Cơ sở lý thuyết

Cơ sở của bài toán điều tiết lũ qua hồ chứa dựa trên phương trình cân bằng nước tổng quát:

$$\frac{dV}{dt} = Q_{in} - Q_{out} \quad (1)$$

trong đó V là dung tích hồ (m^3), Q_{in} và Q_{out} là lưu lượng vào và ra (m^3/s).

Bằng cách rời rạc hóa theo bước thời gian (Δt), ta có dạng sai phân:

$$V_{t+\Delta t} = V_t + (Q_{in,avg} - Q_{out,avg}) \cdot \Delta t \quad (2)$$

với: $V_{t+\Delta t}$ - dung tích hồ tại thời điểm $t + \Delta t$;

$Q_{in,avg}$; $Q_{out,avg}$ - lưu lượng trung bình vào, ra hồ trong khoảng thời gian Δt .

Dung tích (V) được xác định thông qua quan hệ đặc trưng của hồ $V = f(Z)$, trong đó Z là cao trình mực nước hồ. Quá trình điều tiết được mô phỏng cho đến khi lưu lượng xả Q_{out} và mực nước Z đạt trạng thái ổn định hoặc đạt đỉnh lũ.

Hai phương pháp được sử dụng để giải phương trình cân bằng nước là:

- Phương pháp Potapov: sử dụng bảng biểu phụ trợ ($q-f_1-f_2$) để xác định nhanh chuỗi ($H(t)$) và ($Q(t)$), cho phép diễn toán bán đồ giải.
- Phương pháp Thử dần (trial-and-error): cho phép thử nhiều giá trị lưu lượng Q giả thiết và kiểm định với giá trị lưu lượng tính toán.

Bảng 1: Thông số đầu vào của mô hình tính toán

Thông số	Ký hiệu	Giá trị	Đơn vị
Cao trình ngưỡng tràn	Z_c	44,1	m
Mức nước dâng bình thường	Z_0	44,1	m
Bề rộng tràn	B	45,0	m
Hệ số lưu lượng	m	0,35	—
Hệ số co hẹp bên	ε	1,0	—
Gia tốc trọng trường	g	9,81	m/s^2
Bước thời gian tính	Δt	1	giờ

Dữ liệu đặc trưng của hồ gồm:

- Quan hệ dung tích - cao trình ($V-Z$): xác định thể tích tích nước tương ứng với mực nước.
- Đường quá trình lũ đến: chuỗi lưu lượng vào theo thời gian, được biểu diễn dưới dạng rời rạc (24 giờ).

Mối quan hệ giữa lưu lượng xả và mực nước trên ngưỡng tràn được mô tả bởi công thức:

$$Q = m \cdot \varepsilon \cdot B \sqrt{2g} \cdot H^{3/2} \quad (3)$$

với: m - hệ số lưu lượng, phụ thuộc vào hình dạng và điều kiện dòng chảy;

ε - hệ số co hẹp;

B - bề rộng tràn;

$H = Z - Z_c$ là cột nước trên ngưỡng tràn.

2.3. Xây dựng thuật toán

Quy trình tính toán tự động được mã hóa bằng Python gồm ba bước chính:

(a) Xây dựng bảng phụ trợ

Hàm `build_auxiliary_chart()` tạo bảng $Q_{out}-f_1-f_2$ cho từng cao trình Z lớn hơn ngưỡng tràn.

2.2. Dữ liệu và cấu trúc đầu vào

Mô hình được kiểm chứng trên một hồ chứa đơn có đặc trưng thủy lực trung bình, với tràn có ngưỡng kiểu thực dụng, không cửa van (xả tự do). Các tham số đầu vào sử dụng trong mô phỏng được trình bày ở Bảng 1:

Với mỗi mức nước, tính toán:

$$f_1 = \frac{\Delta V}{\Delta t} - 0.5Q_{out}, \quad f_2 = \frac{\Delta V}{\Delta t} + 0.5Q_{out} \quad (4)$$

Trong đó ΔV là thay đổi dung tích nước tương ứng.

Bảng này được dùng để tra cứu nhanh giá trị lưu lượng trong quá trình diễn toán.

(b) Nội suy và ánh xạ dữ liệu

Hàm `create_interpolation_functions()` tạo ba hàm nội suy tuyến tính:

- ($f_1(Q_{out})$): tra cứu giá trị (f_1) theo (Q_{out}),
- ($Q_{out}(f_2)$): tra cứu ngược giá trị (Q_{out}) theo (f_2),
- ($Z(Q_{out})$): xác định cao trình mực nước theo lưu lượng xả.

Các hàm này đảm bảo tính liên tục và ổn định trong tính toán, cho phép mô phỏng với bước thời gian nhỏ.

(c) Diễn toán quá trình lũ (Flood routing)

Hàm `route_flood_potapov()` thực hiện vòng lặp tính toán theo thời gian:

1. Đọc giá trị lưu lượng vào Q_{in} tại thời điểm

hiện tại và kế tiếp.

2. Tính giá trị trung bình Q_{avg} .
3. Tra cứu f_1 theo (q_1) (bước trước), tính $f_2 = Q_{avg} + f_1$.
4. Tra ngược để xác định q_2 theo f_2 .
5. Cập nhật kết quả và lặp lại cho đến khi hết chuỗi dữ liệu.

Thuật toán cho phép tính tự động mà không cần đồ giải hay tra bảng thủ công.

2.4. Trực quan hóa và giao diện web

Kết quả mô phỏng được trực quan hóa bằng thư viện Matplotlib, với hai đường biểu diễn đồng thời:

- ($Q_{in}(t)$): lưu lượng lũ đến hồ;
- ($Q_{out}(t)$): lưu lượng xả qua tràn.

Đồ thị được làm trơn bằng nội suy spline (`make_interp_spline`) giúp thể hiện rõ đỉnh lũ và quá trình cắt giảm lưu lượng.

Các thông số cực trị (đỉnh lũ đến, đỉnh lũ xả, mực nước lớn nhất) được tự động in ra bảng kết quả.

Chương trình được tích hợp với Streamlit, tạo giao diện web có khả năng:

- Nhập dữ liệu đầu vào ($Q - t$, $Z - V$, thông số tràn);
- Lựa chọn phương pháp (Potapov hoặc Thử dần);
- Chạy mô phỏng và hiển thị đồ thị tương tác;
- Xuất kết quả ra Excel hoặc hình ảnh.

Cấu trúc thư mục chương trình được thiết kế dạng mô-đun, gồm các khối core, viz, và webapp, đảm bảo khả năng mở rộng và tái sử dụng trong tương lai.

3. KẾT QUẢ

Kết quả tính toán quá trình điều tiết lũ bằng phương pháp Potapov, thử dần và tính tay thủ công được thể hiện trong Bảng 2.

Bảng 2: Tóm tắt kết quả mô phỏng điều tiết lũ

Đại lượng	Ký hiệu	Potapov	Thử dần	Thủ công
Đỉnh lũ đến	$Q_{in,max}$, m ³ /s	728,20	728,20	728,20
Đỉnh lũ xả	$Q_{out,max}$, m ³ /s	635,44	635,37	660,87
Thời điểm đỉnh lũ xả	$t_{q,max}$, giờ	16,00	16,00	16,00
Mực nước lớn nhất trong hồ	Z_{max} , m	48,45	48,46	48,58
Dung tích siêu cao	W_{pl} , triệu m ³	≈ 0,06	≈ 0,06	≈ 0,06

Kết quả tính toán điều tiết lũ theo phương pháp Potapov được so sánh với kết quả từ mô phỏng thử dần và tính toán thủ công (Bảng 2). Giá trị đỉnh lũ đến được giữ nguyên trong cả ba phương pháp ở mức $Q_{in,max} = 728,20$ m³/s.

Tuy nhiên, đỉnh lũ xả có sự khác biệt nhỏ: phương pháp Potapov cho giá trị $Q_{out,max} = 635,44$ m³/s, gần trùng với kết quả thử dần $Q_{out,max} = 635,37$ m³/s và thấp hơn so với tính toán thủ công $Q_{out,max} = 660,87$ m³/s.

Thời điểm xuất hiện đỉnh lũ xả trong cả ba phương pháp đều trùng nhau tại $t_{q,max} = 16$ giờ,

cho thấy sự thống nhất về đặc trưng thời gian dòng chảy qua hồ. Mực nước lớn nhất trong hồ (mực nước lũ) chênh lệch không đáng kể giữa các phương pháp (48,45 - 48,58 m), tương ứng với thể tích phòng lũ xấp xỉ 0,06 triệu m³.

Sự tương đồng giữa kết quả tính theo phương pháp Potapov và mô phỏng thử dần chứng tỏ độ tin cậy của thuật toán Potapov trong việc mô phỏng quá trình điều tiết lũ. Sai khác nhỏ so với kết quả thủ công có thể do làm tròn số hoặc cách nội suy trong quá trình xác định quan hệ $Q-Z-W$ của hồ chứa. Nhìn chung,

phương pháp Potapov cho độ chính xác cao, trong khi vẫn đảm bảo tính tự động và giảm thiểu sai sót thao tác so với phương pháp thủ công truyền thống

Hình 1 trình bày quan hệ giữa lưu lượng lũ đến Q_{in} và lưu lượng xả Q_{out} của hồ chứa trong suốt quá trình tính toán. Đường cong thể hiện mối tương quan động giữa hai đại lượng, phản ánh khả năng cắt giảm đỉnh lũ của hồ theo thời gian.



Hình 1: Quan hệ giữa lưu lượng đến và xả theo phương pháp Potapov

Kết quả cho thấy đường quá trình lưu lượng xả Q_{out} có dạng trễ pha so với đường lưu lượng đến Q_{in} , thể hiện rõ chức năng điều tiết của hồ. Khi lưu lượng đến đạt cực đại 728,20 m³/s tại giờ 16, hồ chứa bắt đầu xả mạnh, đỉnh xả đạt 635,44 m³/s – giảm khoảng 12,8 % so với đỉnh lũ đến. Khoảng chênh lệch diện tích giữa hai đường cong tương ứng với thể tích nước được trừ trong giai đoạn lũ dâng, xác định được dung tích phòng lũ xấp xỉ 0,06 triệu m³. Diễn biến này chứng tỏ phương pháp Potapov mô phỏng tốt quá trình chậm pha và cắt giảm lưu lượng đỉnh, bảo đảm tuân thủ nguyên tắc cân bằng nước tổng thể.

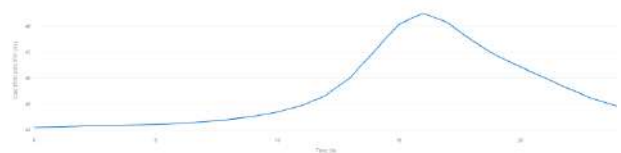
Tương tự, Hình 2 mô tả quan hệ giữa lưu lượng đến và xả khi tính toán theo phương pháp Thử dần. Mục đích của đồ thị này là đối chiếu đặc điểm diễn toán của hai phương pháp và đánh giá độ tương đồng về kết quả mô phỏng. Đường Q_{out} thu được từ phương pháp Thử dần có dạng gần như trùng khớp với kết quả của

Potapov, với đỉnh xả 635,37 m³/s và mực nước lớn nhất 48,46 m. Sai số cực đại giữa hai phương pháp dưới 0,1 %, cho thấy thuật toán Potapov có khả năng tái hiện chính xác kết quả của phương pháp đồ giải cổ điển. Sự đồng nhất về thời điểm và biên độ đỉnh lũ khẳng định độ ổn định của thuật toán nội suy, đồng thời chứng minh rằng mô hình Python đã tái lập được logic thủy lực của bài toán điều tiết truyền thống.

Ngoài ra, để giúp việc đánh giá quá trình điều tiết trở nên trực quan hơn, Hình 3 dưới đây thể hiện đường quá trình mực nước hồ ($Z-t$) được mô phỏng song song với quá trình lũ đến và lưu lượng xả, cho phép quan sát rõ sự biến thiên mực nước theo thời gian và khả năng cắt giảm đỉnh lũ của hồ chứa.



Hình 2: Quan hệ giữa lưu lượng đến và xả theo phương pháp thử dần.



Hình 3: Đường quá trình mực nước hồ ($Z-t$)

4. THẢO LUẬN

Kết quả mô phỏng điều tiết lũ bằng chương trình Python được đối chiếu với phương pháp đồ giải truyền thống cho thấy sai số giữa lưu lượng xả cực đại của hai cách tính nhỏ hơn 4%. Mức sai số này nằm trong giới hạn cho phép của các bài toán thủy văn – thủy lực, chứng tỏ độ tin cậy cao của thuật toán nội suy và các phép xử lý số được áp dụng. Sự trùng

khớp về thời điểm và mực nước đỉnh lũ giữa các phương pháp càng củng cố tính chính xác và ổn định của mô hình tính toán.

Việc tự động hóa toàn bộ quy trình tính toán từ nhập dữ liệu, thực hiện tính toán cho đến xuất kết quả mang lại nhiều ưu điểm đáng kể so với phương pháp thủ công. Hệ thống giúp giảm thiểu sai sót trong quá trình lập bảng, nội suy hoặc xử lý dữ liệu, đồng thời đảm bảo tính nhất quán giữa các lần chạy mô hình. Kết quả được lưu trữ dưới dạng DataFrame cho phép truy xuất, kiểm chứng và xử lý hậu kỳ một cách minh bạch. Bên cạnh đó, cấu trúc chương trình được thiết kế linh hoạt, có khả năng mở rộng để tích hợp các mô-đun hiện đại như AI-based decision support, SCADA systems, hoặc real-time flood forecasting, hướng tới một nền tảng điều hành hồ chứa thông minh.

Thực quan hóa kết quả là yếu tố quan trọng giúp kỹ sư và nhà quản lý dễ dàng đánh giá hiệu quả cắt lũ và ra quyết định kịp thời. Giao diện dự kiến phát triển trên nền Streamlit cho phép người dùng lựa chọn phương pháp tính (Potapov hoặc Thử dần), quan sát các đồ thị động $Q(t)$ và $Z(t)$ theo thời gian, cũng như xuất báo cáo tự động dưới các định dạng phổ biến như .xlsx hoặc .JPEG. Tính tương tác cao của giao diện giúp rút ngắn thời gian phân tích và tăng tính trực quan trong công tác đào tạo, vận hành và kiểm định mô hình.

5. KẾT LUẬN

Nghiên cứu này đã phát triển thành công một chương trình Python tự động hóa toàn bộ quy trình tính toán điều tiết lũ hồ chứa, dựa trên nền tảng lý thuyết của phương pháp Potapov và Thử dần. Hệ thống cho phép xử lý khép kín từ bước nhập dữ liệu, nội suy, diễn toán đến thực quan hóa kết quả, giúp loại bỏ hoàn toàn các thao tác thủ công, rút ngắn đáng kể thời gian tính toán và nâng cao độ chính xác của mô phỏng.

Kết quả kiểm chứng cho thấy sai số giữa chương trình và phương pháp đồ giải truyền thống nhỏ hơn 4%, trong khi thời gian xử lý giảm hơn 90%. Độ trùng khớp về đỉnh lũ đến, đỉnh lũ xả và mực nước lớn nhất trong hồ chứng minh tính đúng đắn và ổn định của thuật toán Potapov được mã hóa trong môi trường Python. Việc lưu trữ kết quả dưới dạng DataFrame giúp kiểm chứng, truy xuất và mở rộng dữ liệu thuận tiện, đáp ứng yêu cầu minh bạch và tái lập trong nghiên cứu khoa học.

Ưu điểm nổi bật của mô hình là khả năng tích hợp cao và thân thiện người dùng. Giao diện Streamlit hỗ trợ thao tác trực quan, hiển thị kết quả động và xuất báo cáo tự động, qua đó mở rộng phạm vi ứng dụng từ nghiên cứu, đào tạo đến quản lý vận hành công trình thủy lợi. Công cụ có thể triển khai trực tuyến thông qua Streamlit Cloud hoặc Flask API, tạo điều kiện thuận lợi cho chia sẻ và truy cập từ xa.

Kết quả đạt được đã khẳng định tiềm năng của Python trong chuyển đổi số công tác thiết kế và vận hành hồ chứa. Hướng phát triển tiếp theo bao gồm mở rộng mô hình cho bài toán điều tiết liên hồ, bổ sung các kịch bản lũ kiểm tra nhiều tần suất và tích hợp các phương pháp học máy như PINNs hoặc ANNs để tối ưu hóa vận hành theo thời gian thực.

LỜI CẢM ƠN: Nghiên cứu này được Trường Đại học Thủy lợi hỗ trợ thông qua đề tài nghiên cứu khoa học cấp cơ sở, mã số CS2026-40, với tên đề tài: “Ứng dụng trí tuệ nhân tạo trong hỗ trợ tính toán và tối ưu hóa công trình hồ chứa”.

TÀI LIỆU THAM KHẢO

- [1] T. N. To, H. C. Vu, and H. Le, “Impacts of reservoir operation and urbanization on flood inundation in the Vu Gia–Thu Bon Basin, Vietnam,” *Water Supply*, vol. 22, no. 4, pp. 4656–4675, 2022.
- [2] M. Giuliani, J. R. Lamontagne, P. Reed, and A. Castelletti, “A state-of-the-art review of optimal reservoir control for managing conflicting demands in a changing world,” *Water Resources Research*, vol. 57, no. 12, p. e2021WR029927, 2021.
- [3] V. M. Yevjevich, *Bibliography and Discussion of Flood-Routing Methods and Unsteady Flow in Channels*, U.S. Government Printing Office, Washington, D.C., Water-Supply Paper 1690, 1964.
- [4] A. L. Hamilton, T. J. Amestoy, and P. M. Reed, “Pywr-DRB: An open-source Python model for water availability and drought risk assessment in the Delaware River Basin,” *Environmental Modelling & Software*, vol. 181, p. 106185, 2024.
- [5] S. Minocha, P. Das, and F. Hossain, “Reservoir assessment tool (RAT): A Python package for monitoring the dynamic state of reservoirs and analyzing dam operations,” *Digital Water*, vol. 3, no. 1, pp. 1–30, 2025.
- [6] B. Nazari and D. J. Seo, “Analytical solution for nonlinear hydrologic routing with general power-law storage function,” *Journal of Hydrology*, vol. 598, p. 126203, 2021.
- [7] A. Tahiri, D. Che, D. Ladeveze, P. Chiron, and B. Archimède, “Network flow and flood routing model for water resources optimization,” *Scientific Reports*, vol. 12, no. 1, p. 3937, 2022.
- [8] Đ. Đ. Nguyễn, L. T. H. Bình, P. T. Phương and P. H. Đức, “Ứng dụng mô hình học máy XGBoost và LightGBM trong việc dự báo mực nước triều trên sông Sài Gòn–Đồng Nai,” *Tạp chí Khoa học và Công nghệ Thủy lợi*, số 82, 2024.
- [9] N. V. Nam, T. M. Tuấn, Đ. T. T. Hiền, K. T. Dũng, Đ. O. Cường, and N. T. Trung, “Sử dụng mô hình học máy trong hỗ trợ diễn đoán thủy lực, thủy văn trên hệ thống Bắc Hưng Hải,” *Hội nghị Nghiên cứu Cơ bản và Ứng dụng Công nghệ Thông tin*, Oct. 2019, doi: 10.15625/vap.2019.00055