SoftwareX 36 (2026) 103149

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Policy Coherence Kit: A Python-based application for perceived policy coherence analysis

Ali Torkayesh [a, b, *], Anahita Amiri [c], Sandra Venghaus [a, b]

[a] Economics of Sustainability, School of Business and Economics, RWTH Aachen University, 52072 Aachen, Germany
[b] Forschungszentrum Jülich GmbH, Institute of Climate and Energy Systems, Jülich Systems Analysis (ICE-2), 52425 Jülich, Germany
[c] Independent Researcher, Aachen, Germany

ARTICLE INFO

Keywords:
Perceived policy coherence
Policy toolkit
Python
LLM-assisted interpretation

ABSTRACT

With the rising socio-economic, environmental, and technological concerns, policy coherence has become a critical concept for addressing complex policy-related challenges that span multiple sectors and governance levels. There exist well-established analytical frameworks in the literature for assessing interactions between policies. However, an important shortcoming of these analytical frameworks is their limited practical application in conducting policy coherence assessments. In this context, this study presents a Python-based software toolkit with a graphical user interface for perceived policy coherence assessment, in which policy interactions are captured through expert-informed evaluations. The software addresses the policy coherence assessment through a structured examination of policy interactions, influence patterns, network-based metrics, entropy measures, and dimensionality reduction using principal component analysis. The software supports the full analytical workflow, from data input and matrix construction to network analysis and visualization, thereby facilitating the application of perceived policy coherence assessment in empirical contexts. Additionally, it incorporates large language model-assisted interpretation to support the explanation and contextualization of analytical results, while recommending maintaining a human-in-the-loop approach. Overall, the main goal of this study is to transform the established perceived policy coherence assessment framework into an accessible tool that enables bridging of the gap between the conceptual framework and its practical implementation.

Metadata

Code metadata.

Nr.	Code metadata description	Metadata
C1	Current code version	1.0
C2	Permanent link to code/repository used for this code version	https://github.com/atorkayesh/PolicyCoherenceKit
C3	Permanent link to Reproducible Capsule	
C4	Legal Code License	MIT License
C5	Code versioning system used	git
C6	Software code languages, tools, and services used	Python
C7	Compilation requirements, operating environments & dependencies	Python 3.10 or higher, openpyxl, numpy, reportlab, Pillow
C8	If available Link to developer documentation/manual	
C9	Support email for questions	torkayesh@sustainability.rwth-aachen.de & a.torkayesh@fz-juelich.de

1. Motivation and significance

One of the major objectives of complex and cross-sectoral policy areas, such as energy, climate, and transport, is maintaining policy coherence [1,2]. As policy domains are highly interconnected, decisions made in one area affect other areas, leading to synergies or conflicts [3,4]. Despite its growing importance, policy coherence remains a

* Corresponding author at: Economics of Sustainability, School of Business and Economics, RWTH Aachen University, 52072 Aachen, Germany.
Email addresses: torkayesh@sustainability.rwth-aachen.de; a.torkayesh@fz-juelich.de (A. Torkayesh).

https://doi.org/10.1016/j.softx.2026.103149
Received 31 May 2026; Received in revised form 8 October 2026; Accepted 14 October 2026
Available online 22 October 2026
2352-7110/© 2026 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).
Open access funding enabled and organized by Projekt DEAL.

Cancelling	Counteracting	Constraining	Neutral Unknown Consistent	Enabling	Reinforcing	Indivisible
-3	-2	-1	0	+1	+2	+3
Prevents the achievement of another policy entirely	Works against or undermines another policy	Restricts or limits the achievement of another policy	No meaningful interaction, either positive or negative	Facilitates conditions that help advance another policy	Supports and strengthens the achievement of another policy	Inherently linked; the achievement of one policy depends on the other

Fig. 1. Assessment scale for the interactions between policies [10].

challenging concept to define and assess [5–7]. At its core, policy coherence means how effectively policies reduce conflict and promote synergies in achieving common objectives [8,9]. Coherence is an attribute of relationships between multiple policies rather than a property of individual policies. As a result, a policy may be effective in achieving its objectives, but it can lead to inconsistencies in relation to other policy domains. To understand policy coherence, it requires a change in perspective from an individual policy evaluation to a system-wide perspective that captures interactions across policies [8,10].

In the policy analysis domain, coherence is investigated through the interaction of policy outputs, such as objectives and instruments, and their real-world implementation [8,10]. These elements are embedded within a broader system linking policy inputs, processes, outcomes, and impacts. However, coherence analysis examines how policies interact at different stages, particularly in terms of outputs and implementation, where synergies and conflicts arise [4,11–13]. The empirical assessment of policy coherence is challenging, even though it can be conceptually framed by examining interactions across objectives, instruments, and implementation.

A well-established framework for policy coherence assessment was introduced by Nilsson et al. [8]. This framework conceptualizes coherence as the interaction between policies and proposes a structured three-step analytical approach consisting of (i) an inventory of policy objectives, (ii) a screening matrix to identify the strength and direction of interactions, and (iii) a more detailed mapping of key policies and their interactions. This framework, particularly its use of screening matrices and qualitative assessments, relies heavily on expert knowledge to identify and evaluate policy interactions [8,10] (Fig. 1). However, expert judgment is applied implicitly without standardizing perceived interactions across multiple policies. Therefore, the policy coherence assessment framework can be framed as a perceived policy coherence assessment [14].

The framework begins with a screening matrix based on the identified policies. For instance, let n represent the number of identified policies. Then, a matrix is established in $n \times n$ format. Subsequently, the qualitative screening matrix is converted to a quantitative one according to the scale shown in Fig. 1 [15,16].

Weitz et al. [17] extended this framework by treating the screening matrix as a network structure. In this setting, policies are represented by nodes, and links between nodes denote whether an interaction exists between each pair of policies. The weight of the links is determined based on the coherence degree between each pair of policies.

To move beyond the qualitative interpretation of the screening matrix and complement the network representation of policies [17], Torkayesh and Venghaus [14] suggested incoming and outgoing coherence scores for each policy, network centrality measures to identify key policies, Shannon's entropy to determine the range of influence of each policy, and principal component analysis (PCA) to show the high-dimensional screening matrix in a 2D plot. To derive further in-depth insights from the network representation of policies, betweenness, and closeness centrality measures were applied. Betweenness centrality captures the extent to which a policy acts as a bridge within the policy landscape. Closeness centrality captures how centrally a policy is positioned within the coherence network, reflecting how efficiently it can influence or be influenced by other policies.

The practical implementation of the policy coherence framework is limited despite the conceptual clarity. Considering the dependency of the framework on qualitative assessments and expert judgment, systematically capturing, comparing, and reproducing policy interactions across broader and more complex policy sets becomes challenging. In particular, the absence of structured data representation and integrated analytical support constrains scalability and limits the ability to derive deeper insights from coherence assessments. To overcome this issue, a Python-based application, Policy Coherence Kit, with a graphical user interface (GUI), has been developed in this study. The software is used to conduct perceived policy coherence assessment in a systematic, transparent, and reproducible manner by facilitating the collection of expert-based interaction data, enabling persistent development of policy interaction matrices, and providing integrated analytical functionalities to explore patterns of synergy and conflict within policy systems.

2. Software description

2.1. Purpose and design

We developed the Policy Coherence Kit as an analytical tool to support the assessment of policy coherence using qualitative data from experts and decision-makers. This software application's main objective is to enable the construction, quantification, comparison, aggregation and interpretation of policy interactions across a policy landscape.

The software adopts a project-based workflow. Hence, each decision-maker's assessments are recorded as a separate matrix. Collected matrices are then aggregated. The workflow also enables combining matrix editing, aggregation, visualization, and export within a single unified framework.

Three practical requirements are considered in designing the software application. The first requirement is related to the quantification of the linguistic and qualitative policy coherence assessments by decision-makers. In this context, the quantification begins with the aggregation of interaction matrices based on four aggregation operators, including averaging based on equal weights among decision-makers, majority rule-based, weighted synthesis of judgments, and median-based order. The second practical requirement deals with complementary analytical perspectives on the aggregated matrix, including coherence scores, influence ranges, dimensionality reduction, and network structure. Finally, the last requirement keeps the analysis in modular format to enable extending the analytical components without redesigning the entire application. With the successful implementation of these requirements, the software is structured as a local, modular desktop application in which core analytical logic is separated from the interface layer while remaining tightly integrated into the user workflow. An important step in developing the software was reviewing and optimizing the pipeline using Anthropic's Claude Sonnet 4.6 model. All outputs were subsequently checked and verified.

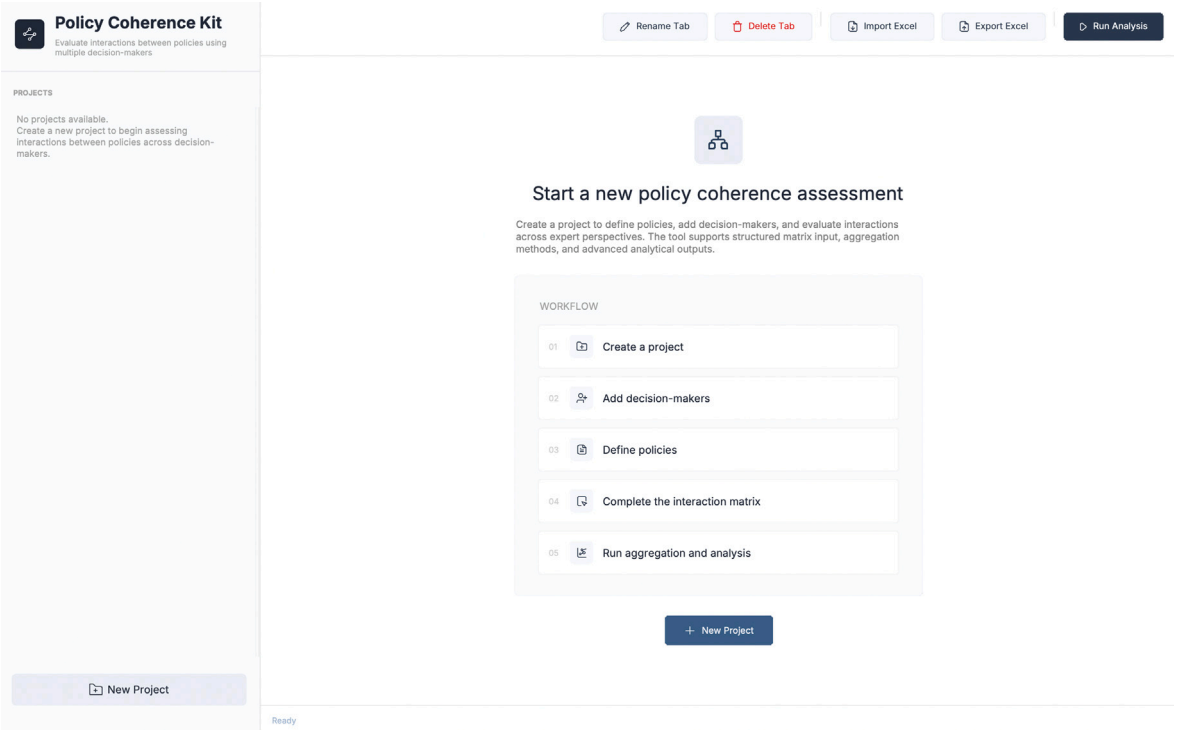


Fig. 2. Home page of the Policy Coherence Kit.

2.2. Software architecture

The software is built on a modular format with four main layers, including a presentation layer, an application control layer, a domain logic layer, and an analytical layer. The first layer, the presentation layer provides the GUI (Fig. 2). User actions, project state, data flow, viewing, and data consistency requirements are considered in the application control layer.

The key layer in the software architecture is the domain logic layer. This layer coordinates the internal representation of the policy interaction matrices and their aggregated quantified versions. In this layer, each decision-maker's policy interaction matrix is represented as a structured data object. This object contains policy labels, policy codes, and matrix cell-level interaction values. Inserted decision matrices are kept separate until the aggregation is implemented. Following the completion of the aggregation, an aggregated result object is created, which serves as the shared input for all downstream analytical modules.

Using the aggregated object, results are built in the analytical layer. This layer includes the quantified aggregated matrix, coherence scores, range of influence measures, network centrality measures, PCA diagram, and automated interpretation with LLMs. Consistency within the software architecture is ensured by the independent use of the aggregated data for each analytical tool.

2.3. Core data model

The project, its corresponding matrices, and the aggregated matrix build the internal data model of the developed software. At the highest-level, a project functions as a container and stores all information related to a specific coherence assessment problem, the number of decision-makers and their corresponding interaction matrices, and finally the aggregated matrix.

At the input level, each off-diagonal cell stores a categorical interaction judgment drawn from the predefined seven-point coherence scale, ranging from strongly synergistic to strongly conflicting relations (Fig. 1). Following the completion of the decision matrices, the decision matrices are validated through simple steps. If the user inserts an external data file in the form of a workbook with multiple sheets, the software checks each sheet one by one as follows:

- Each sheet must contain matching row and column policy headers, and the seven-point coherence labels must be drawn from the predefined scale.
- Empty sheets or sheets with too few policy headers are skipped, while blank cells are flagged as warnings rather than immediate errors.
- Matrices with unrecognised rating values or mismatched row/column headers are rejected to ensure data consistency before analysis.
- Following the import of a workbook, all matrices are checked for completeness and for shared policy lists before aggregation and downstream analysis are allowed to proceed.

After this step, the aggregated matrix generates an object that contains the numerical representation of the aggregated matrix, metadata on the aggregation method used, the number and identity of contributing decision-makers, and the source cell values required for secondary analyses such as agreement estimation. This data object is then used for further analytical steps, such as coherence score calculation, and network centrality measures.

For generating the aggregated interaction matrix, the user can apply one of the supported aggregation operators, including simple averaging, majority-rule aggregation, weighted aggregation, and median-based aggregation. In the case of majority-rule aggregation, ties must be resolved before finalizing the aggregated matrix. Additionally, for the median-based aggregator, if the number of decision-makers is even, the software asks whether the user wants to choose the lower or upper median value. Using this matrix, the software produces several complementary outputs mentioned earlier. Users can export matrices and analytical outputs in spreadsheet format, save selected visualizations, and generate structured textual interpretations through the integrated LLM module.



Fig. 3. Data entry tab.

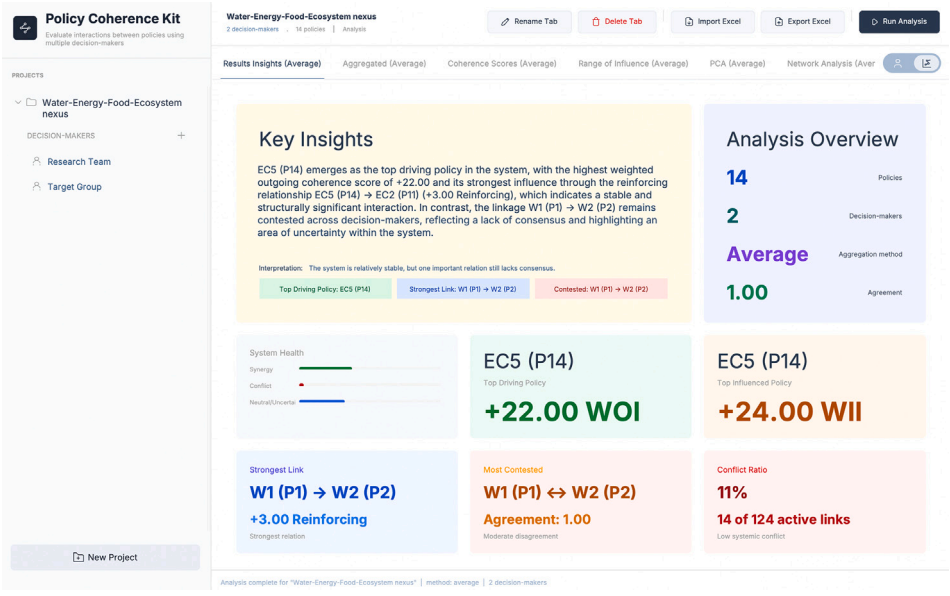


Fig. 4. Summary of the analysis.

2.4. Analytical modules

As discussed earlier, the software allows a series of analyses to be conducted using the quantified, aggregated matrix in the analytical layer. Through each analytical module, we present a specific perspective on the structure of policy coherence assessment in a quantitative way.

The first module presents the coherence scores module. Using the aggregated matrix, this module computes directional indicators that distinguish between outgoing and incoming influence, both in count-based and weighted forms. In the next module, the software presents an overview of the range of influence exerted by each policy based on Shannon's entropy. As entropy requires non-negative probabilities, the calculations in this module are performed based on the absolute values of the outgoing values in the aggregated interaction matrix. Higher entropy indicates that a policy's influence is more evenly distributed across multiple targets, whereas lower entropy indicates that its influence is concentrated on fewer policies.

The network analysis module interprets the aggregated matrix as a directed weighted network, allowing the examination of connectivity

patterns, relation polarity, and node centrality. In the current implementation, centrality is computed using betweenness and closeness measures derived from shortest paths on a graph, where edge distances are defined as the inverse of the absolute influence degree between policies. In the network representation, edge signs are retained only for visualization. Thus, negative ties are not treated as negative distances. For the computations, the software implements Dijkstra's algorithm and Brandes' algorithm for betweenness centrality [18], while closeness centrality is derived from shortest-path distances.

The PCA module projects policy influence profiles into a lower-dimensional space. The goal is to facilitate the visual identification of clusters, divergences, and broad structural tendencies in the policy set. The module uses the aggregated matrix to build an $n \times (n-1)$ matrix of outgoing policy influence scores, excluding self-relations. The signed values from the aggregated scale are retained without additional transformation, and zeros are preserved as structural non-influences. The resulting component loadings show which policy relations most strongly contribute to each principal component.

Finally, the interpretation module uses an LLM interface to transform quantitative outputs into structured narrative analysis. While this component does not replace expert judgment, it supports exploratory interpretation and communication by synthesizing numerical results into a policy-oriented narrative. In this module, the user is asked to choose between the OpenAI and Groq engines. Subsequently, they can choose their preferred model. In the next step, an API key needs to be entered. The software does not record the API key under any circumstances. It is only used once to interpret the results. Following the completion of all steps, the user clicks on "Generate Interpretation" to finalize the interpretation with an LLM. The prompt to generate the interpretation is designed as follows:

System prompt:
You are an expert policy analyst specialising in policy coherence and systems thinking. You provide rigorous, evidence-based, in-depth interpretations of policy interaction analyses. Your responses are structured, analytical, and written for policy professionals.
User prompt:
You are given the results of a policy coherence analysis for n policies, aggregated using the {method-label} method.

Provide a structured, in-depth interpretation with the following sections. Each section should consist of multiple paragraphs with specific references to the data above:

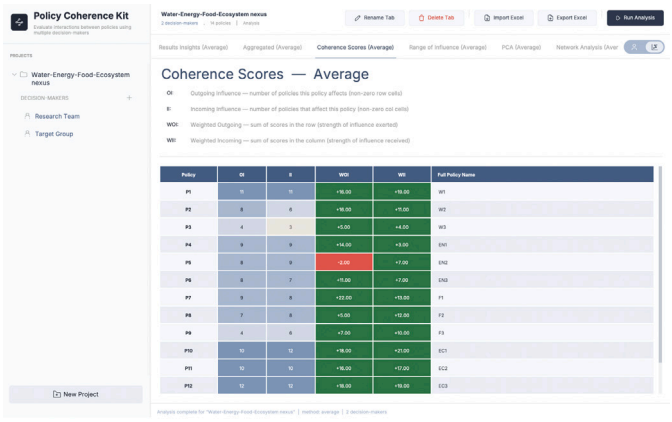
1. Overall Coherence Assessment, 2. Key Synergies, 3. Key Conflicts, 4. Influential Policies, 5. Range of Influence, 6. Network Structure, 7. Strategic Recommendations.

Ensure your interpretation is analytical, references specific policies by name and code, and provides actionable insights for policy professionals.

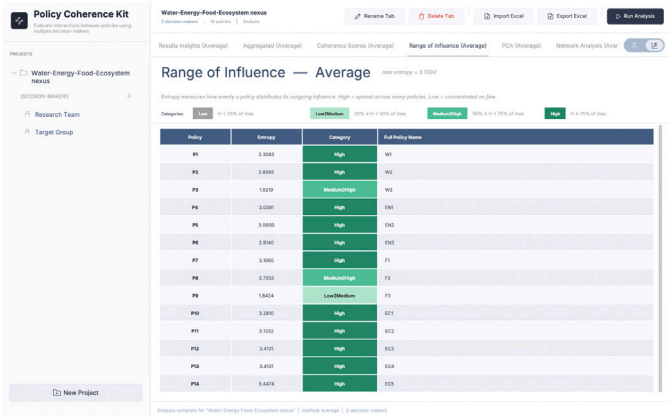
3. Illustrative example

To show the applicability of the developed software, an illustrative example is selected from Mooren et al. [19]. This study addressed the policy coherence assessment for Water-Energy-Food-Ecosystem nexus policies in Greece. The framework considers three water policy goals, three energy policy goals, three food/agriculture policy goals, and five ecosystem policy goals. The screening matrix for the interactions of the identified policy goals was conducted by the authors and later verified by a focus group of four local experts. In this part, we demonstrate all steps applied in the software with multiple snapshots.

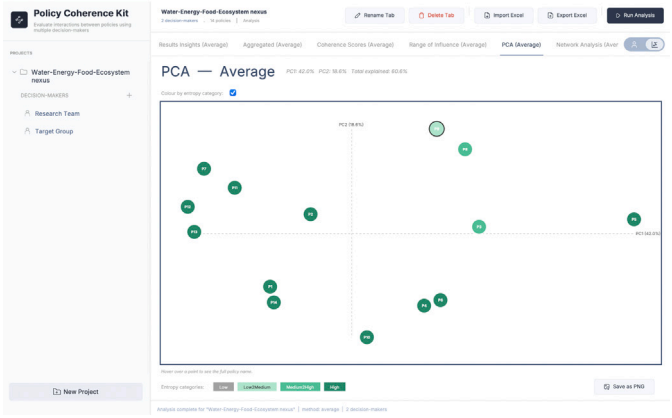
In the very first step, the user chooses to create a new project on the home page (Fig. 2). Later, a new window pops up, asking to define the project name, add decision-makers, and policy names. The user can also import this info to the software with an *XLSX* file, in which sheet names represent decision-makers.



(a) Coherence scores.



(b) Range of influence.

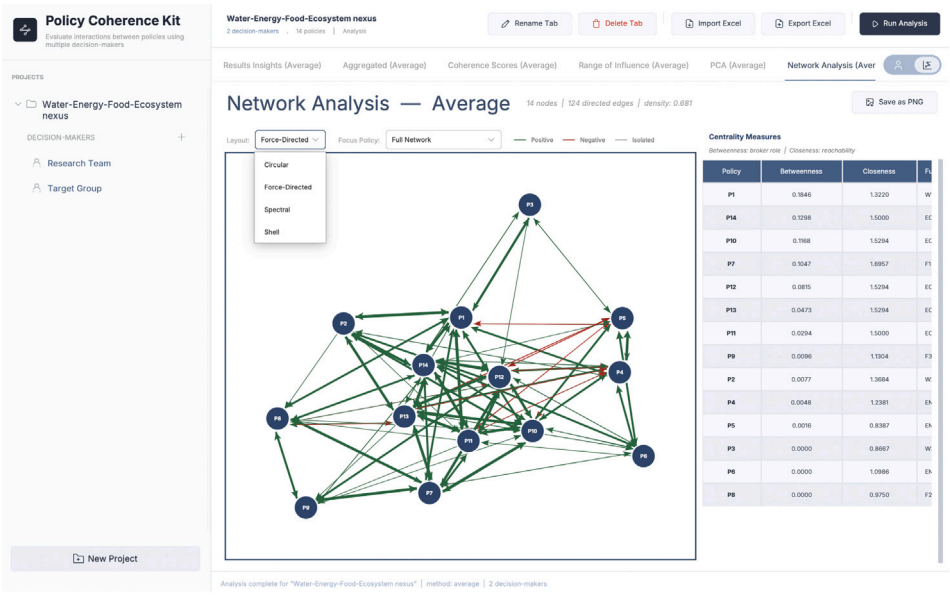


(c) Low-dimension representation of the policies.

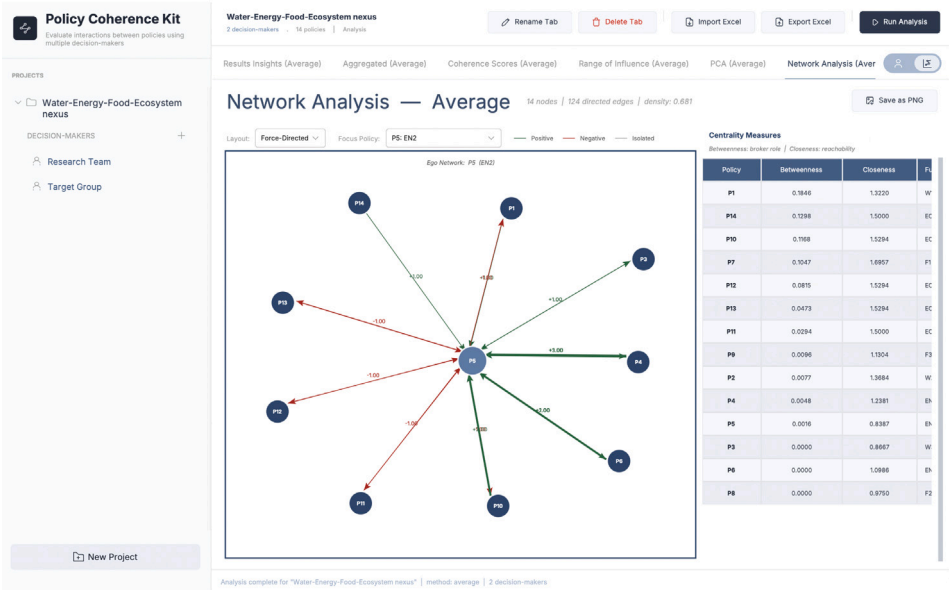
Fig. 5. Summary of policy characteristics and relationships.

Accordingly the name of the project is set to *Water-Energy-Food-Ecosystem nexus*. The study considers a single screening matrix built by the research team and a target group. We adopted the same matrix; however, we duplicated it into two matrices to show how the software works with multiple decision-makers (Fig. 3). This example data is also available on the GitHub repository.

With the completion of the data entry, the user clicks on the "Run Analysis" button. With this step, several new tabs are generated to present the results. In the first tab "Results Insights", multiple cards are shown to present overall information regarding the policies and their interactions. The card "Key Insights" provides a summary of the



(a) Policies and their interactions with different layout options.



(b) Policies shown individually.

Fig. 6. Network representations of policies and their relationships.

analysis. The card “Analysis Overview” shows key information about the problem setting, including the number of policies, the number of decision-makers, the aggregation method for the matrices, and the agreement level among decision-makers. The summary of the analysis is further addressed by smaller cards as shown in Fig. 4.

The next tab is “Aggregated (aggregator operator)”. This tab represents the aggregated quantified screening matrix. The analytical part begins with the tab “Coherence Scores”, where four coherence scores are defined at the top of the tab, and corresponding results are shown below them (Fig. 5a). This tab provides a solid and quick overview of the incoming and outgoing interactions and influences of each policy.

We move forward with Shannon’s entropy to measure the range of influence of each policy based on its outgoing vector. The tab “Range of Influence” presents entropy values for all policies, and categorizes them into four groups, namely low, low-to-medium, medium-to-high, and high (Fig. 5b). At the top of the tab, required information is provided regarding the maximum entropy value, and how the categories are built.

In this mixed-method setting, the next step is to reduce the dimension of the screening matrix using PCA. The tab “PCA” demonstrates a 2D plot, where policies are scattered based on the first and second principal components. According to Fig. 5(c), this tab provides an option to check, allowing the user to categorize the policies based on their range of influence. At the top of the tab, brief information is shown regarding the variance covered by the data.

The “Network Analysis” tab presents several options to the user (Fig. 6a). At the core of this tab, a graph representation of the policies is provided. Users can choose four layout algorithms to reconstruct the network. Moreover, the tab provides an option, where the user can choose a specific policy in order to derive the network representation of its interactions with other policies (Fig. 6b). On the right side of the tab, a table reports betweenness and closeness centrality measures of the

policies. The user can export each network as a *PNG* file to their local directory.

Using the LLM module for the example by Mooren et al. [19], the following interpretation is generated. To keep the word count down, we present a shorter version of the generated interpretation below:

1. Overall Coherence Assessment

The coherence analysis of the 14 policies reveals … Notably, policies P1, P2, and P10 stand out with high coherence scores, suggesting they are pivotal in fostering collaborative outcomes across the policy landscape.

The coherence scores, particularly the Weighted Outgoing Influence (WOI) and Weighted Incoming Influence (WII), are …

Conversely, policies such as P5 exhibit negative coherence, with a WOI of -2.00, indicating …

2. Key Synergies

Key synergies emerge prominently among several policies, …

Another notable synergy is observed between P10 and P12, where both policies exhibit high coherence scores (WOI of 18.00 for P10 and WII of 19.00 for P12)…

3. Key Conflicts

Conflicts within the policy framework are evident, particularly with a policy like P5, which has a negative WOI of -2.00. This indicates that P5 not only fails to support other policies but also …

…

4. Influential Policies

The analysis identifies several influential policies …. P10 emerges as a key player with a high WOI of 18.00 and WII of 21.00, indicating its significant capacity to influence and be influenced by other policies. …

…

5. Range of Influence

Policies such as P12 and P14 exhibit high entropy scores (3.4131 and 3.4474, respectively), suggesting that their influence is widely distributed across multiple policies. …

In contrast, P9, with a low entropy score of 1.8424, indicates a more concentrated influence, suggesting that it may be overly reliant on a limited number of policies for support…

6. Network Structure

P10, P14, and P7 emerge as central nodes within the network, facilitating connections and interactions among other policies. Their roles as brokers enhance the overall coherence of the policy landscape, suggesting that strengthening these policies could yield significant benefits.

…

7. Strategic Recommendations

To enhance policy coherence and effectiveness, several strategic recommendations emerge from the analysis. First, policymakers should prioritize strengthening the central policies, particularly P10, P14, and P7, because their roles as brokers can facilitate collaboration and alignment across the policy landscape. Initiatives that leverage their strengths and foster partnerships among other policies can amplify positive outcomes.

Second, addressing the conflicts associated with P5 is crucial … Third, …

Lastly, …

4. Impact

The software enhances the practical applicability of perceived policy coherence assessment by integrating established analytical modules within a unified and accessible environment. The software improves reproducibility by addressing technical barriers to implement each analytical module used for policy coherence assessment.

To support responsible use of the software, users should treat it as a decision support tool rather than a substitute for expert judgment. The analysis pipeline is transparent and reproducible at the workflow level. It begins with validated, multi-sheet workbook inputs and continues through aggregation, quantitative indicators, and visual and textual outputs. Another key aspect to note is related to the automated LLM-based interpretation of the results. For the LLM module, the software currently uses a structured prompt and an external API key for a single request. The software guarantees that the key is removed from memory. Another important point regarding the software is the anonymization of decision-makers. Users are responsible for anonymizing expert inputs where appropriate, obtaining consent, and considering privacy and ethical guidelines. All outputs, especially LLM-generated narratives, should be interpreted carefully and reviewed by experts before substantive use.

5. Conclusions

In this paper, we present the Policy Coherence Kit, a Python-based software package that operationalizes perceived policy coherence assessment through a user-friendly GUI. Using the software, researchers and policy analysts can address policy coherence assessment problems in a mixed-method setting within a unified environment. The software automates major steps in the assessment process, including data quantification and aggregation, calculation of coherence scores, determination of the range of influence, dimensionality reduction, network representation and analysis, as well as the interpretation of results using LLMs. Overall, the Policy Coherence Kit provides a practical foundation for strengthening evidence-based policy analysis and for advancing research on policy coherence across different sectors and contexts.

In the future, the software and its workflow could be developed further. From a methodological perspective, additional ordinal and robust aggregation operators and sensitivity analyses might be incorporated to better evaluate the robustness of the results under different assumptions. The interface itself would benefit from richer uncertainty and provenance visualizations, stronger privacy and audit features, and improved interoperability with external modelling and network ecosystems. Furthermore, the LLM-assisted interpretation module has the potential to be developed further with human-in-the-loop validation and clearer safeguards, supporting responsible and transparent use. Finally, a web-based interface designed to eliminate technical requirements could facilitate the use of the software.

CRediT authorship contribution statement

Ali Torkayesh: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Resources, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Anahita Amiri:** Validation, Software, Resources. **Sandra Venghaus:** Writing – review & editing, Supervision, Resources, Funding acquisition, Conceptualization.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used DeepL and Claude to improve language and readability. After using these tools, the authors have reviewed and edited the content as needed and take full responsibility for the content of the publication.

Funding

This work was supported by the Helmholtz Association under the program “Energy System Design.”

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The source code is publicly available at https://github.com/atorkayesh/PolicyCoherenceKit. Further instructions for installing the package are included in the README in the same repository.

References

[1] May PJ, Sapotichne J, Workman S. Policy coherence and policy domains. Policy Stud J 2006;34(3):381–403.

[2] Dzebo A, Shawoo Z, Browne K. Does policy coherence make national implementation of global sustainability agendas more successful? Annu Rev Environ Resour 2025;50.

[3] Picciotto R. The evaluation of policy coherence for development. Evaluation 2005;11(3):311–30.

[4] Waheed A, Bernward Fischer T, Khan MI. Climate change policy coherence across policies, plans, and strategies in Pakistan—implications for the China–Pakistan economic corridor plan. Environ Manag 2021;67(5):793–810.

[5] Gottenhuber S. The idea of policy coherence: closing or widening the implementation gap of sustainability objectives? Critical Policy Studies 2025;1–22.

[6] Koura A, Boudhar A, Oudgou M. Navigating complexity: unveiling the concept of policy coherence. The International Journal of Interdisciplinary Organizational Studies 2024;19(2):169.

[7] Andabaka A, Basarac Sertić M, Yiatros S, Nisiforou O, De Simone M, Tambovceva T, Atstāja D, Cudečka-Puriņa N, Borg RP, Iordachi V, et al., National policies and the challenge of coherence. In: Shaping circular transitions in the built environment: from barriers to enablers. Springer; 2025. p. 215–437.

[8] Nilsson M, Zamparutti T, Petersen JE, Nykvist B, Rudberg P, McGuinn J. Understanding policy coherence: analytical framework and examples of sector–environment policy interactions in the EU. Environmental policy and governance 2012;22(6):395–423.

[9] Den Hertog L, Stross S. Policy coherence in the EU system: concepts and legal rooting of an ambiguous term. In: Conference paper the EU as a global player; 2011.

[10] Nilsson M, Griggs D, Visbeck M. Policy: map the interactions between sustainable development goals. Nature 2016;534(7607):320–2.

[11] Aigbe GO, Stringer LC, Cotton M. Gas flaring in Nigeria: a multi-level governance and policy coherence analysis. Anthropocene Science 2023;2(1):31–47.

[12] Ramisch M, Kirschke S, Weber N. Multi-level policy coherence analysis in wicked nexus problem settings: the case of nitrogen pollution in Germany. Environmental Policy and Governance 2025;35(6):983–97.

[13] Shawoo Z, Maltais A, Dzebo A, Pickering J. Political drivers of policy coherence for sustainable development: an analytical framework. Environmental Policy and Governance 2023;33(4):339–50.

[14] Torkayesh A, Venghaus S. Perceived policy coherence assessment in a cross-sectoral landscape governing power-to-X fuels in Germany: a mixed-method approach; 2026. https://ssrn.com/abstract=6441121.

[15] Häbel S, Hakala E. Policy coherence for sustainable development and environmental security: a case study of European Union policies on renewable energy. Environmental policy and governance 2021;31(6):633–46.

[16] Blicharska M, Smithers RJ, Kuchler M, Munaretto S, van den Heuvel L, Teutschbein C. The water–energy–food–land–climate nexus: policy coherence for sustainable resource management in Sweden. Environmental Policy and Governance 2024;34(2):207–20.

[17] Weitz N, Carlsen H, Nilsson M, Skånberg K. Towards systemic and contextual priority setting for implementing the 2030 agenda. Sustain Sci 2018;13(2):531–48.

[18] Brandes U. A faster algorithm for betweenness centrality. J Math Sociol 2001;25(2):163–77.

[19] Mooren CE, Papadopoulou C-A, Munaretto S, Levedi K, Papadopoulou MP. A methodological framework for assessing the coherence of Water-Energy-Food-Ecosystem nexus policies: illustration and application at the river basin level. Environ Sci Policy 2025;170:104113.