

Bridge Chloride Exposure Predictor: a web-based tool to forecast the future chloride exposure rates for bridges

Yingxue (Cynthia) Liu Mingsai Xu Spencer Smith
McMaster University, Canada

25 Oct 2026

1 Summary

The GitHub project [Bridge Chloride Exposure Predictor \(BCEP\)](#) is a web-based tool designed to forecast future chloride exposure rates for bridges based on a calculation model developed by Xu and Yang (Xu et al. 2024; Xu 2024), with an assumption that the climate and traffic data is constant within a specific GRID_SIZE for a given JURISDICTION. This model integrates traffic and climate data to predict chloride exposure, specifically focusing on damage from deicing salts while excluding other factors such as mechanical wear or accidents. Currently, this model is validated for Ontario, Canada. A [demonstration website](#) is available for an accessible overview. The tool can be used to investigate chloride exposure for new jurisdictions by providing new traffic and climate data. The inputs, outputs, data requirements and theoretical models for BCEP are summarized in the [Software Requirements Specification \(SRS\)](#).

2 Statement of need

Most highway bridges are constructed with reinforced concrete decks, which are susceptible to chloride-induced corrosion due to adverse weather conditions and traffic patterns. Various factors contribute to this corrosion, including the quality of construction materials and maintenance practices, with deicing salts being a significant contributor. The primary deicing salt used is sodium chloride (rock salt). When rock salt melts the snow and comes into contact with water, it can undergo a chemical reaction, releasing chloride ions. These ions can penetrate the concrete and cause corrosion in the reinforcing steel, thereby compromising the bridge's structural integrity and load-bearing capacity. Chloride is considered the main source of corrosion damage to reinforced concrete. Chloride ions are transported from the road to the exterior surface of bridge substructures through

vehicle spray and splash mechanisms. There is a tight connection between chloride exposure, weather conditions and traffic flow. Specifically, the amount of deicing salt applied on the road surface greatly depends on the amount of snowfall, and the amount of water and dissolved chloride ions that end up on nearby objects depends on the traffic patterns.

While existing research explores the relationship between chloride and corrosion, there remains a lack of accessible research tool to visualize chloride exposure rates effectively. Accurate prediction of chloride exposure rates is essential for effective bridge management, construction planning, and research. Government agencies can use this data to prioritize maintenance and allocate budgets efficiently, focusing on bridges with higher corrosion risks. Bridge engineers can use the information to determine minimum structural requirements and ensure that bridges can endure their intended lifespan, particularly during the precise design phase. Researchers, especially those studying the impacts of climate change on infrastructure, benefit from predictive models to understand and mitigate corrosion damage. This project, conducted in collaboration with Dr. Cancan Yang from McMaster University, highlights the importance of such tools in addressing the challenges posed by chloride-induced corrosion.

3 State of the field

Proprietary tools exist for predicting and designing the service life of different concrete bridge designs, such as [Life-365](#), [STADIUM](#) and [COMSOL](#) models. However, these tools require an estimate of the expected chloride concentration in the environment around the bridge. As far as we know BCEP is the first tool to determine and visualize this value for a given location via local traffic and climate data.

4 Software design

BCEP consists of Python code and JavaScript with Vue. The Python code is for converting the raw climate and traffic data into a database covering the jurisdiction of interest. The JavaScript and Vue are used for the web application, which provides a graphical view for input and output. Javascript and Vue were chosen because they integrate well with map data, allowing for visualization and mouse click input that is geographically restricted. Vue simplifies UI development by introducing a declarative programming model. The [installation instructions](#) are for a Python virtual environment, so installation is isolated to not interfere with the developer's usual build environment. The Python and JavaScript combination make the software portable between Windows, Mac OS, and Linux.

The design of the software is based on the principle of information hiding (Parnas 1972). That is, the design is decomposed around likely changes, which become the secrets of the modules. Likely changes include the algorithm for calculating

chloride exposure and the method for graphing the data. The software is divided into 17 modules, including modules for input, control, constants, chloride on pier calculations and plotting. The details of the decomposition, and the mapping between the conceptual design and the source code, can be found in the [Module guide](#). The module guide, which includes the uses relation between modules, is based on Parnas’s ideas for documenting the software architecture (Parnas and Weiss October 2001; Parnas et al. 1984). The interface provided by each module is documented in the [Module Interface Specification](#) (MIS). The MIS follows the approach presented by Hoffman and Strooper (Hoffman and Strooper 1995) and later adapted for scientific computing software (Smith and Yu 2009; ElSheikh et al. 2004).

BCEP has been verified following the [Verification and Validation Plan](#), results of which are show in the [Verification and Validation Report](#).

5 Research impact statement

BCEP was developed to assist with Xu’s PhD research “[Lifetime Serviceability and Safety of Highway Bridges Under Climate Change](#)” (Xu 2024). Xu’s dissertation addresses the growing challenge of ensuring the serviceability and safety of concrete highway bridges amidst climate change. Using BCEP future researchers will be able to continue to investigate the effects of climate change on bridges safety. Xu’s research focused on Canada, but the tool was designed to be usable in other jurisdictions by [updating the traffic and climate data](#).

Although not the main purpose of BCEP, there is a secondary research impact. BCEP was used as a case study for applying software engineering principles (like information hiding) and methodologies (like modular decomposition) to scientific computing software. This work is documented in the report “[A Collaborative Framework Toward Minimizing Pain Points in Research Software Development](#)” (Liu 2024). The report describes a practical framework to empower small teams of domain experts and developers to collaboratively build sustainable research software. The framework addresses common pain points of evolving requirements and researchers’ limited technical familiarity. Central to the approach is structured requirements elicitation, where developers guide domain experts through targeted questions about theories, typical uses cases, computational problem scale and possible tests. The answers directly inform modular design, verification, and documentation.

6 Features and usage

A quick start guide is provided in the [README.md](#) of the repository.

As outlined in [Figure 1](#), there are two sources of input:

- User Input: Information is entered through the website interface on the

front end.

- Developer Input: Updates (if any) to the traffic table, climate table, and salt application rate need to be made by editing the code directly.

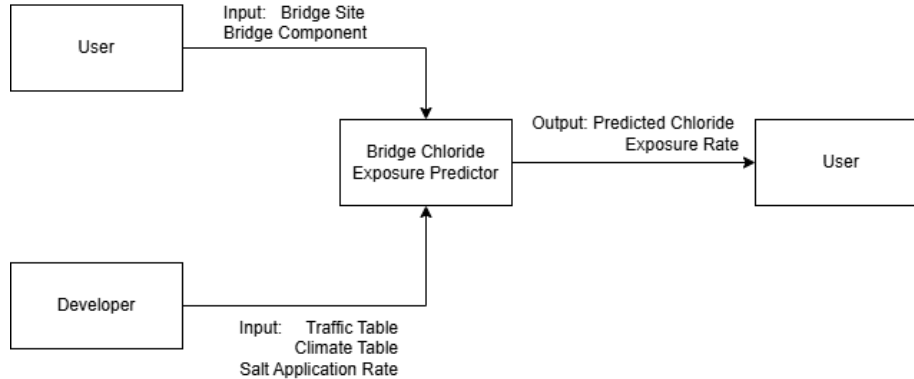


Figure 1: Input and Output of BCEP

The following sections provide guidance on modifying and customizing the website to suit other jurisdictions.

6.1 Customization

To adapt the code for jurisdictions beyond Ontario, follow these steps to ensure compatibility and proper results:

6.1.1 1. Adapting Mechanisms

- The physical mechanisms remain the same, but differences in deicing salt application policies should be considered when customizing for a new jurisdiction.

6.1.2 2. Data Collection and Preparation

- Collect climate data and traffic data specific to the target jurisdiction.
- Organize the data into a single Excel file.
- **Ensure Consistency:**
 - **Grid Size Matching:** The grid sizes for climate and traffic data must align.
 - **Row Alignment:** Each row in the sheets should represent the same location. For instance, row 2 across all sheets must correspond to the same longitude and latitude.

6.1.2.1 2.1 Climate Data Requirements

- **Data Types:**
 - **htotal**: Total snowfall during winter
 - **t1**: Number of days with snowfall
 - **t2**: Number of days with snow melting
- **Data Format:**

Climate data should be structured as shown below, with years as columns:

Year	2006	2007	2008	...
	206.2145241	220.8943212	121.5795313	...
	211.174729	221.0476551	137.5310097	...

6.1.2.2 2.2 Traffic Data Requirements

- **Data Types:**
 - **AADT/Lane**: Annual average daily traffic per lane
 - **AADTT/Lane**: Annual average daily truck traffic per lane
- **Data Format:**

Traffic data should include coordinates, city, and traffic metrics per lane:

Longitude	Latitude	City	AADT/Lane	AADTT/Lane
274.712677	49.45780945	Algoma	559	103
275.9333496	47.5466156	Algoma	559	103
276.9502258	50.10984039	Cochrane	603	132

6.1.3 3. Sheet Naming Conventions

To ensure the tool functions correctly, the Excel file must follow specific naming conventions for each sheet:

- **Climate Data Sheets:**
 - Each type of climate data (**htotal**, **t1**, **t2**) should be placed in a separate sheet, as shown in the data format above.
 - The sheet names must exactly match the data type, e.g., a sheet for total snowfall should be named **htotal**.
- **Traffic Data Sheet:**
 - All traffic data (**AADT**, **AADTT**) should be included in a single sheet, as shown in the data format above.
 - The sheet must be named **traffic**.

7 AI Usage Disclosure

AI was not used for coding or documentation. However, AI was used for brainstorming content for the [CONTRIBUTING.md](#) guide, the [CodeOfConduct.md](#)

and the continuous deployment of the documentation via a [GitHub Action](#). In all cases the tool used was ChatGPT based on GPT-5.5. The human authors have reviewed, verified and edited all AI suggested text.

8 Acknowledgements

We acknowledge the insights and suggestions from Dr. Cancan Yang during the development of this program and the preparation of this paper.

References

- ElSheikh, Ahmed H., W. Spencer Smith, and Samir E. Chidiac. 2004. “Semi-Formal Design of Reliable Mesh Generation Systems.” *Advances in Engineering Software* 35 (12): 827–41.
- Hoffman, Daniel M., and Paul A. Strooper. 1995. *Software Design, Automated Testing, and Maintenance: A Practical Approach*. International Thomson Computer Press.
- Liu, Cynthia. 2024. “Requirements Validation Process for Research Software: A Case Study with Bridge Chloride Exposure Prediction Software.” Master’s thesis, McMaster University, Hamilton, Ontario, Canada.
- Parnas, D. L., P. C. Clement, and D. M. Weiss. 1984. “The Modular Structure of Complex Systems.” *International Conference on Software Engineering*, 408–19.
- Parnas, David L. 1972. “On the Criteria to Be Used in Decomposing Systems into Modules.” *Comm. ACM* 15 (2): 1053–58.
- Parnas, David L., and David M. Weiss. October 2001. “Definition and Documentation of Software Architectures.” *Technical Report ALR-2001-023*, October 2001.
- Smith, W. Spencer, and Wen Yu. 2009. “A Document Driven Methodology for Improving the Quality of a Parallel Mesh Generation Toolbox.” *Advances in Engineering Software* 40 (11): 1155–67. <https://doi.org/http://dx.doi.org/10.1016/j.advengsoft.2009.05.003>.
- Xu, Mingsai. 2024. “Lifetime Serviceability and Safety of Highway Bridges Under Climate Change.” PhD thesis, McMaster University.
- Xu, Mingsai, Yuxin Zheng, and Cancan Yang. 2024. “Assessing Highway Bridge Chloride Exposure at a Provincial Scale: Mapping and Projecting

Impacts of Climate Change.” *Journal of Bridge Engineering* 29 (August).
<https://doi.org/10.1061/JBENF2.BEENG-6669>.