Predi-XY: a python program for automated generation of NICS-XY-scans based on an additivity scheme

Journal Article

Author(s):

Wahab, Alexandra; Fleckenstein, Felix; Feusi, Stefan; Gershoni-Poranne, Renana

Publication date:

2020-12-30

Permanent link:

<https://doi.org/10.3929/ethz-b-000461285>

Rights / license:

[Creative Commons Attribution 4.0 International](#)

Originally published in:

Electronic Structure 2(4), <https://doi.org/10.1088/2516-1075/abd081>

Predi-XY: A Python program for automated generation of NICS-XY-Scans based on an Additivity Scheme

Alexandra Wahab,^{#a} Felix Fleckenstein,^{#a} Stefan Feusi,^a and Renana Gershoni-Poranne.^{*a}

[#] these authors contributed equally.

^a Laboratory for Organic Chemistry, Department of Chemistry and Applied Biosciences, ETH Zurich

NICS, NICS-XY-Scan, aromaticity, polycyclic aromatic systems, additivity, Python.

ABSTRACT: Polycyclic aromatic systems are prevalent in chemistry and materials science because their thermodynamic stability, planarity, and tunable electronic properties make them uniquely suited for various uses. These properties are closely linked to the aromaticity of the systems. Therefore, characterizing the aromatic behavior is useful for designing new functional compounds and understanding their reactivity. NICS-XY-Scans are a popular and simple tool for investigating the aromatic trends in polycyclic systems. Herein we present Predi-XY: an automated system for generating NICS-XY-Scans for polycyclic aromatic systems using an additivity scheme. The program provides the predicted scans at a fraction of the computational cost of a full quantum mechanical calculation and enables rapid comparison of various polycyclic aromatic systems.

Introduction

Aromaticity and its counterpart, anti-aromaticity, are important fundamental concepts in chemistry. They are commonly associated with the stability^{1,2} and reactivity of a diverse family of organic compounds, as well as with selectivity in pericyclic reactions.^{3,4} Due to their prevalence and importance, many different methods have been developed to identify and quantify the aromaticity of conjugated ring systems.^{5–8} Yet, while the aromaticity of monocyclic systems can be characterized by various techniques, polycyclic aromatic systems (PASs) remain challenging. Not only are they computationally more expensive to calculate at reliable levels of theory, they also exhibit more complex aromatic behavior.^{9,10} For example, a single conjugated system made up of several fused rings may be aromatic in one part and anti-aromatic in another. Though this may not be surprising for molecules such as the [N]phenylenes, which contain both aromatic and anti-aromatic moieties,¹¹ we have also recently reported on similar behavior in an all-benzenoid molecule.¹² Beyond the fundamental interest into their electronic structure, accurate characterization of the aromatic behavior of PASs also has practical significance. Due to their stable low band-gaps, tunability, and structural features, PASs have long been the workhorses of organic electronics.^{13–16} In recent years, specific classes of PASs, such as indacenes^{17,18} and indenofluorenes,^{19,20} have shown great promise in optical and electronic devices. Furthermore, as the aromaticity is tightly linked to the electronic behavior of these compounds,²¹ it serves as a useful metric for rational design of new molecules.^{17,22–24} Thus, a simple, rapid, and accurate method for characterization of the aromaticity of PAS scaffolds is highly desirable.

Owing to its ease-of-use and quantitative nature, the Nucleus Independent Chemical Shift (NICS)^{25,26} method is one of the most popular methods for evaluating aromaticity.^{7,27} NICS is a computational technique based on the ring-current model.^{28,29} According to this model, when an external magnetic field is applied to a cyclically-conjugated system, ring currents are generated in the π -orbital system. These ring currents lead to the generation of an induced magnetic field, which can be assessed by placing a probe atom at any location desired by the user. Conventionally, this probe is placed above the center of the ring under investigation. For polycyclic compounds, such a description offers only a partial description and often neglects important characteristics. Thus, several years ago, the NICS-XY-Scan method was introduced.³⁰ This method is specifically aimed at treating PASs and it allows the identification of local, semi-global, and global ring currents within polycyclic structures. Recently, a comprehensive investigation of several systems demonstrated the relationship between NICS-XY-Scans and current density maps, and provided guidelines for safe interpretation of NICS-XY-Scans.³¹ Other methods exist for visualization of the NICS values of a PAS.^{32,33} However, to the best of our knowledge, the NICS-XY-Scan is one of the most reliable NICS-based methods for evaluation of aromaticity in PASs, because it can be used with the NICS _{π ZZ} metric.²⁷ As NICS is implemented in many popular quantum chemical computational programs, it is very user-friendly and accessible. Moreover, Aroma,^{34–36,30} a free utility written in Python, automates generation of input files for calculating NICS-XY-Scans with the Gaussian suite of programs,³⁷ as well as collection of the relevant data from the output files.

Recently, work from our group demonstrated that the NICS-XY-Scans of unbranched *cata*-condensed PASs can be predicted with an additivity scheme. Following the proof of principle on poly-benzenoid compounds,³⁸ we demonstrated the generality of the method on PASs containing also 7 types of heterocyclic moieties.³⁹ The additivity scheme provides new insight into the concept of aromaticity. From a practical perspective, it also makes it possible to obtain these characterizations with the most refined version of NICS, the NICS_{nzz} metric, at a fraction of the computational cost and time, because the largest subunit that needs to be calculated at an expensive high-level calculation is a tricyclic building block. Moreover, it provides access to molecules that are prohibitively expensive to calculate at this level of theory.

Having established the generality of the additivity scheme, we now introduce Predi-XY, a freely available Python⁴⁰ program for automated prediction of NICS-XY-Scans for unbranched *cata*-condensed PASs, based on the scheme. Predi-XY has been designed for non-experts, with a focus on user-friendliness and streamlining, both in application and in customization. It can handle a wide range of PASs, and aims to serve computational and experimental chemists who wish to characterize the aromatic nature of such systems. Predi-XY's current library enables it to generate NICS-XY-Scans for countless compounds containing benzene, pyridine, pyrazine, pyrrole, furan, thiophene, 1,4-diborinine, and 1,4-dihydro-diborinine (Figure 1). We plan to continue to expand the library in the course of our group's work, and note that users can send us requests for new building blocks. We are also working on a platform to allow users to submit building block data to a global library that all users could actively update.

Figure 1. (Anti)-aromatic moieties in the Predi-XY library. Underneath each molecule are the NICS(1.7)_{nzz} value at the center of the ring and the molecule's notation within the program. NICS data was calculated with the B3LYP/6-311+G(d) combination.

We envision that by incorporating the contributions of new building blocks from various users, Predi-XY will expand its capabilities and usefulness for the community. Instructions for installing, running, customizing the program are detailed in the accompanying documentation. The code for Predi-XY can be installed with pip or downloaded directly from our group's GitLab repository: <https://gitlab.com/porannegroup/predixy/-/releases>.

This report is structured as follows: a) short overview of the additivity scheme; b) description of Predi-XY, including its library and customization options; c) sample outputs and details on performance.

The Additivity Scheme

Because the additivity scheme is relatively new, a short description of the method is warranted. The notion of additivity can be phrased simply as "the whole is equal to the sum of its parts". In the language of chemistry, this translates to "the molecular property is a sum of the properties of the subunits contained in the molecule". Many chemical properties are additive; molecular weight and thermochemical properties are some well-known examples.⁴¹

In our perception, any additivity scheme requires two components: the building blocks and the combination rules. The building blocks are the smaller units with which we can construct the larger system we are interested in. The combination rules determine how we calculate the contribution of each building block to the final value of the whole system. Our additivity scheme decomposes any unbranched *cata*-condensed PASs into three types of building blocks: monocyclic, bicyclic, and tricyclic. Note that *all* possible bicyclic and tricyclic subunits are identified, thus the bicyclic and tricyclic subunits overlap, as seen in Figure 2. The full system is then constructed as a series of monocyclic building blocks, connected "head-to-tail", superimposed with the respectively overlapping bicyclic and tricyclic building blocks. Figure 2 demonstrates this on a sample pentacyclic molecule.

Figure 2. a) scheme of sample system (pentacyclic PAH) and the direction of the NICS-XY-Scan through it. b) The identified components: monocyclic subunits (green, bottom row), bicyclic subunits (pink, middle row), tricyclic subunits (blue and light blue, top row). Image reproduced from Ref. 39.

Having identified the building blocks comprising the molecule of interest, we now turn to calculating the additivity-predicted value—namely, the NICS-XY-Scan of the pentacyclic system. Simple summation of all NICS-XY-Scans of the building blocks would result in "double-" and "triple-counting" of the same effects. Thus, we have defined Bicyclic Correction (BC) and Tricyclic Correction (TC) terms. In brief, these correction terms account only for the difference between the building block and what would be expected from additivity (for more details, please see references³⁸ and³⁹). Therefore, when the NICS-XY-Scans for the monocyclic units (green) are summed up with the superimposed BCs (pink) and TCs (blue) NICS-XY-Scans, the resulting predicted scan is gratifyingly close to the result obtained with an explicit calculation (Figure 3).

At this point, it is relevant to mention that NICS-XY-Scans can be obtained with various NICS metrics.^{7,27,42} All of the values in our library are NICS(1.7) π_{ZZ} values, which is considered to be the most refined version of NICS. It is also the most computationally expensive metric. This means that the predicted scans generated by Predi-XY are at the

Figure 3. a) scheme of the superimposed components of the studied pentacyclic system. b) Plot of the respective superimposed NICS-XY-Scans of the components, color coded as is (a). The black line is the explicitly calculated NICS-XY-Scan of the studied system. The red line is the predicted NICS-XY-Scan obtained from the summation of the components in the additivity scheme. NICS data was calculated with the B3LYP/6-311+G(d) combination. Image reproduced from Ref. 39.

highest accuracy of NICS-based aromaticity evaluation, at a fraction of the cost. For sufficiently large systems (#cycles > 10), calculating an explicit NICS-XY-Scan at this level of theory can be prohibitively expensive, or even infeasible. If the relevant building blocks are included in its library, Predi-XY can provide NICS-XY-Scans for these systems within fractions of a second. If not, the time required to calculate the building blocks (the largest of which is a tricyclic system) is still a fraction of the time required to calculate the full system (further details are provided below, in the section on examples and usage statistics).

Description of the Predi-XY Program

1. The Library and Nomenclature of Predi-XY

In order to perform the additivity scheme, a library of building blocks is required. The current library contains over 100 building blocks, which can be used to predict NICS-XY-Scans of countless PASSs. The NICS-XY-Scan data for all building blocks have been calculated at the B3LYP/6-311+G* level of theory, using Gaussian 09.³⁷ All reported values are NICS(1.7) π_{ZZ} , i.e., calculated at a height of 1.7 Å above the molecular plane, using only the contribution of the π electrons to the ZZ component of the chemical shift tensor. This separation was achieved using the Natural Chemical Shielding⁴³ procedure within NBO 6.⁴⁴

The library is contained in a .csv file, which allows it to be easily and rapidly read, and also to be modified by the user in a simple way. To facilitate the automation, we devised a shorthand nomenclature for naming the types of cycles and building blocks.

- Individual cycles are designated with a short notation. For example, benzene = **bn**, furan = **fur**, pyrrole = **pyl**, etc. The names of all monocyclic building blocks are specified in Figure 1.

- For 5-membered heterocycles, which can appear in two different orientations, we distinguish between cases by adding “a” or “b” to their name. Figure 4 demonstrates these two cases, as well as the kind of structural motifs to which they are relevant.

Figure 4. Illustration of annulation cases of 5-membered heterocyclic systems. The red arrow depicts the pathway along which the NICS-XY-Scan data is computed for the different cases.

- Bicyclic building blocks are named according to the individual monocyclic units they contain, e.g, naphthalene = **bn-bn**.
- Tricyclic building blocks are named according to the individual cycles they contain, with an additional notation for linear or angular annulation. Three examples are shown in Figure 5.

Figure 5. Examples of tricyclic building blocks and their respective notation in the Predi-XY program, a) carbazole, b) anthracene, c) phenanthrene.

New building blocks can be appended to the library file using the same notation conventions. The order of building blocks in the library file is not important. However, the naming of the molecule must be consistent with the direction of the path used for obtaining the data. Figure 6 demonstrates the correct (and incorrect) way to name a building block. In this case, the two depicted tricyclic building blocks are chemically identical, but their NICS-XY-Scans will be mirror images, due to the opposite directions of the scan through them. To ensure correct usage, the data must be entered into the library in accordance with the correct notation.

Figure 6. Examples of correct (green) and incorrect (red) naming of tricyclic building blocks. The order of the name must correspond to the order of monocyclic building blocks as they are encountered in the pathway through the molecule.

2. The Workflow of Predi-XY

The workflow of the Predi-XY program is described schematically in Figure 7.

Figure 7. Workflow of Predi-XY. Boxes on the left show the process demonstrated on a PAS. Text boxes on the right describe what the program does at each step. The user needs only to supply the molecular structure as xyz coordinates and obtains the predicted scan as output.

As described in the previous section, there are two main components to the additivity scheme: the building blocks and the combination rules. Accordingly, applying the scheme to any molecule has two main steps: identification of the building blocks in the correct order and summation of the superimposed contributions. In writing Predi-XY, we aimed to streamline the process such that it would require minimal input from the user and provide the output in the clearest way. In other words, the user should simply input the molecule they are interested in, using a simple format, and receive a plotted predicted NICS-XY-Scan. The user should not have to prepare any specialized input files for Predi-XY, which would add both inconvenience and the potential for mistakes. The user also should not have to interact with the program at each step to instruct it or provide additional input. To achieve this, some additional steps are required. Thus, Predi-XY has five main modules.

Module 1: Reading the Input. Aside from the library, the only input Predi-XY requires are the Cartesian coordinates of the atoms in the system in Ångstrom. The program accepts two formats that are easily generated and commonly used: .xyz or .in (a Gaussian input file format; note that Cartesian coordinates are required, not Z-matrix format). These coordinates may be obtained from an optimization calculation. Alternatively, one can draw the molecule with a program such as Chem3D (PerkinElmer ChemOffice) or Gaussview and save Cartesian coordinates. If this route is chosen, the structure should be cleaned up to ensure chemically valid angles. If using Chem3D, we recommend running a quick MM2 energy minimization to obtain a good structure. The program runs on Windows, macOS, and Linux operating systems. In all of these systems, the program is launched through the command line by calling it and providing the path to the input file. Once the coordinates have been input, Predi-XY rotates the molecule, to ensure it is lying in the XY plane.

Module 2: Parsing the Molecule. Predi-XY generates a connectivity table from the Cartesian coordinates, using the covalent radii of the atoms⁴⁵ and a covalency factor of 1.3 (this value can be modified by the user, *vide infra*). This means that if the distance between two atoms is smaller than the 1.3 x (sum of their covalent radii) they are considered to be bonded.

Using the connectivity data and NetworkX,⁴⁶ a Python package for the creation and manipulation of the structures of complex networks, Predi-XY generates an undirected graph, in which each atom is represented by a vertex and each bond is represented by an edge. From this graph representation, the individual cycles, which are referred to as "knots", are identified. By listing the atoms comprising each ring, Predi-XY can identify the type of ring (e.g., CCCCCC = **bn**; CCCOC = **fur**) from a list of pre-configured ring types, which can be modified by the user.

Module 3: Identification of NICS-XY-Path. Predi-XY generates a new graph consisting of the knots. Using NetworkX, all pathways through this graph from one end of the molecule to the other, are identified. If the input structure is a branched system, the program will identify more than one possible pathway and exit with an error message. In the future, we plan to implement the ability to deal with branched molecules. At this time, the additivity scheme has not yet been demonstrated for such cases.

Module 4: Identification of Building Blocks and Generation of NICS-XY-Scan Data. This module deals with the construction of the data for the final predicted NICS-XY-Scan through a sequence of steps:

- Predi-XY goes through the pathway and identifies the sequence of bicyclic and tricyclic building blocks contained in the structure.
- Predi-XY verifies that each identified building block exists in the database. If a building block is missing, the program will issue an error message specifying which building block it lacks and will then exit.
- The program verifies whether the direction of the building block is the same as in the database. For example, the database may contain the tricyclic building block **bn-bn-pyl**, but the structure will have the building block **pyl-bn-bn**. In this case, Predi-XY will recognize that the direction of reading the building block is reversed and will mirror the NICS-XY-Scan data after the fitting process (*vide infra*), so that it fits with the direction the building block appears in the molecule.
- For tricyclic building blocks, Predi-XY checks whether they are linearly or angularly annulated (e.g., anthracene or phenanthrene, respectively, see above). This is done by identifying the value of the angle θ , which is the angle created by three adjacent knots in the knot graph. We have set the threshold such that if $\theta > 150^\circ$, the tricycle is identified as linear.
- The length of the predicted NICS-XY-Scan is measured from the starting to ending point, based on the inputted geometry. Note that if a distorted geometry is used, the predicted scan will match this geometry. Therefore, we strongly recommend using optimized geometries.
- The data for the respective building blocks is extracted from the database and fitted with a cubic spline interpolation (*vide infra*). The fitted data is stored in a new array that is padded such that the data corresponds to the location of the building block along the pathway. The length of each subunit is mapped to its actual length in the inputted geometry.
- The BCs and TCs are calculated by subtracting the monocyclic values from the bicyclic values (for the BC) and the sum of monocyclic and BCs from the tricyclic values (for the TCs), as described in the section above and in references³⁸ and ³⁹.

All of the fitted and padded lists for the building blocks are placed in a pandas^{47,48} DataFrame according to their respective location along the pathway. To clarify what this means, we demonstrate this procedure in Figure 8. Starting from the left-hand side, the first ring is a benzene. The program will identify it and extract the data for **bn** from the database. The second ring is also benzene, thus the first bicycle will be identified as **bn-bn**. The third ring is thiophene. Accordingly, the second bicycle will be identified as **bn-thi** (rings #2 and #3) and the first tricycle will be identified as **bn-bn-thi**, and so forth. Since the second bicyclic building block overlaps rings #2 and #3, it will have 0's for all distance points before ring #2 and after ring #3. Similarly, the second tricycle will have 0's for all rings before ring #2.

Figure 8. a) Scheme of sample molecule, unique building blocks are color coded. b) Scheme of dataframe of NICS-XY-Scan data generated for sample molecule. The colors in the filled-in cells correspond to the respective fill-in building blocks in (a). Cells that are un-colored contain "0".

- A row-sum is calculated to obtain the sum of all building blocks contributions at each distance point along the scan. The values are saved into a new column in the dataframe. These form the predicted NICS-XY-Scan for the full system.
- Because the resulting plot can appear "choppy" (see Ref. ³⁸ for further details), by default Predi-XY applies the Savitzky-Golay filter to the final predicted NICS-XY-Scan data to smooth the plot. The smoothed data is then added in a new column in the dataframe. The user can toggle off this setting to plot the untreated data. An example is provided in the following section.

As mentioned above, a fitting process is employed by this module, which was not included in the original manual application of the additivity scheme procedure. The advantages of adding the fitting step are twofold. The first advantage relates to the issue of matching lengths. One of the difficulties we encountered when applying the scheme that the size of a molecule in isolated (e.g., benzene) could be different from its size when part of a fused system (e.g., anthracene). For example, consider that the length of the NICS-XY-Scan of anthracene would not be equal to 3 times the length of the NICS-XY-Scan of benzene. This would lead to problems in constructing the additivity-generated scan. When the scans were constructed manually, in such cases, we would judiciously remove extra points, taking care to retain the symmetry of the scan. However, to make matters worse, the changes could be inconsistent—longer in structures, shorter in others. Clearly, this posed not only an inconvenience, but also made it difficult to automate the process. Using the fitting function, we map the length of each subunit to its actual length according to the inputted geometry. Thus, these difficulties are circumvented and the data matches up and creates a smoother predicted scan, without sacrificing accuracy. The second advantage pertains to the user's ability to extend Predi-XY. The data contained in Predi-XY's library was obtained with 0.1 Å interval scans, which are quite dense. By adding the fitting step, we ensure that the user can add new data that is obtained with scans with other interval spacing of the NICS probes. Of course, this is only valid as long as the spacing is not so sparse that meaningful interpolation becomes impossible.

Module 5: Plotting and Output. By default, Predi-XY provides two files as output: a) an image file and b) a text file.

- The image file contains two panels. In the top panel, a picture of the molecule appears, with the trajectory of the scan-path, including its direction. The bottom panel displays the predicted scan itself. Placing these two panels together is of utmost importance, as the interpretation of the NICS-XY-Scan depends on the direction. The type of image file can be selected by the user (see Customizable User Options below)
- The .log file is a text file that details all of the information used to generate the plot. Namely, this file lists the atoms in each cycle, the building blocks identified in the molecule, and the dataframe of all contributions. This allows the user to verify that Predi-XY correctly identified the subunits in the system, in case there is any doubt. It also enables the user to plot any portion of the data in any program of their choosing (e.g., Excel, Origin). The .log file will also specify whether Predi-XY terminated normally and will detail the time required for each module in the program, as well as the total time for the computation.

A minimal print option is also available, which offers a .log file that contains only the distance points (x axis) and final predicted NICS-XY-Scan (y axis). If

smoothing is applied, the smoothed data will be printed. This is useful for iterating over and plotting multiple files with any shell system.

Additionally, Predi-XY offers a third type of output, a .arm file. This is an input file for the Aroma program and is intended to allow the user to perform an explicit calculation of the molecule, in the event they are interested in comparing a calculated NICS-XY-Scan with the predicted one. Manually generating .arm files is not entirely painless. One needs to provide lists of the atoms within each ring, and the program is sensitive not only to the order of atoms but also to the order of rings. Any error will lead to program failure. Since Predi-XY performs this structural analysis automatically as part of its process, we have added the option to print this data out in Aroma-syntax in order to assist users in generating input files.

An important note should be made regarding planarity constraints on the inputted geometries. Predi-XY does not require the molecule to be planar and the program's connectivity and ring-finding algorithms will work for non-planar systems, as well. Thus, the generated .arm file will be correct. However, one should keep in mind that the building blocks were calculated for planar geometries. Therefore, the predicted NICS-XY-Scan will describe a (hypothetical) planar system composed of these building blocks. The more the real system is distorted from planarity, the more it deviates from this model system, and this can result in less accurate predicted scans. Based on our group's experience (though not benchmarked rigorously), distortions of up to $\sim 20^\circ$ do not affect the aromaticity significantly; qualitative and semi-quantitative analyses can still be performed using the NICS-XY-Scan of the planarized model. Indeed, in the past we have deliberately planarized molecules in order to more cleanly analyze the aromatic character.¹¹ For such cases, the result provided by Predi-XY will be an appropriate description for the planar system, without having to generate and calculate the model.

3. Customizable User Options

Several features of Predi-XY can be customized through toggling or modifying the respective control in the config.ini file:

- The covalency factor can be changed.
- Whether or not to generate the .arm file.
- Whether or not to have the predicted NICS-XY-Scan pop up after calculation.
- Whether or not to save the image file of the plot.
- The type of graphic file to be generated (e.g., .pdf, .png, .svg). For a complete list of available file types, please refer to the documentation of matplotlib.
- The maximal/minimal values on the vertical axis (NICS values) of the plot.
- Whether or not to perform a smoothing process on the predicted NICS-XY-Scan.
- Whether to print a verbose or a minimal .log file.

- Whether or not to print to stdout.

Examples and Performance Details

To demonstrate the usability of Predi-XY, we prepared two sets of test compounds, each consisting of 7 molecules containing from 4 to 10 cycles. The first test set (compounds **4a** – **10a**, Figure 9 left) are all polycyclic

Figure 9. Test sets of 14 compounds. Compounds **4a-10a** are PAHs, compounds **4b-10b** are PASs.

aromatic hydrocarbons (PAHs) consisting only of benzene monocyclic building blocks. This set allows us to compare the performance of Predi-XY on systems of various sizes made up of the same 3-4 building blocks. The second test set (compounds **4b-10b**, Figure 9 right) are PASs containing different monocyclic building blocks. This set allows us to investigate the effects of molecular complexity on the performance of Predi-XY.

The results of a Predi-XY calculation using default settings is shown in Figure 10a, for the example of compound **10b**. In Figure 10b the same calculation is shown, omitting the smoothing process. As can be seen, the output file contains a scheme of the analyzed molecule with an overlaid red arrow indicating the direction of the pathway used in the prediction. Underneath the scheme is the predicted NICS-XY-Plot. We add a horizontal line at $y = 0$ to make it easier to identify negative and positive areas of the scan. The values on the vertical axis can be changed in the config.ini file.

Figure 10. a) Example of a graphical output file generated by Predi-XY using default settings (including smoothing). b) Bottom panel of the output of Predi-XY for the same compound, without applying the smoothing process.

To study the efficiency of the code and how it can impact users’ resources, we performed a comparison of the time required to obtain NICS-XY-Scans in three ways. For all of the molecules in the test set, we:

- Calculated an explicit NICS-XY-Scan with the NICS(1.7)_{nzz} metric at the B3LYP/6-311+G(d) level of theory, using Gaussian³⁷ and NBO 6.^{43,44}
- Calculated the NICS-XY-Scans of all the necessary building blocks using the same computational parameters as in (a) (i.e., functional/basis set, number of processors, and memory).
- Generated predicted NICS-XY-Scans using Predi-XY.

We tabulated the CPU time required for each explicit scan (T^{exp}), the sum of time required for calculating all building blocks in a given system (T^{BB}), and the time required for generating the predicted scans (T^{pred}). Table 1 details the structural features of the 14 test molecules [number of cycles (#Cycles), number of unique building blocks (#BBs)], and the ratios $T^{\text{exp}}/T^{\text{BB}}$ and $T^{\text{exp}}/T^{\text{pred}}$.

Table 1. Comparison of time required for explicit calculation of NICS-XY-Scans, total time for calculating the building blocks required to predict the NICS-XY-Scan from additivity, and time for Predi-XY.

Molecule	# Cycles	# BBs	$T^{\text{exp}}/T^{\text{BB}}$	$T^{\text{exp}}/T^{\text{pred}}$
4a	4	3	$2.3 \cdot 10^0$	$1.3 \cdot 10^5$
5a	5	4	$2.5 \cdot 10^0$	$2.0 \cdot 10^5$
6a	6	3	$1.1 \cdot 10^1$	$4.2 \cdot 10^5$
7a	7	4	$9.7 \cdot 10^0$	$6.0 \cdot 10^5$
8a	8	4	$1.4 \cdot 10^1$	$6.7 \cdot 10^5$
9a	9	3	$6.5 \cdot 10^1$	$1.7 \cdot 10^6$
10a	10	4	$3.5 \cdot 10^1$	$1.6 \cdot 10^6$
4b	4	7	$8.6 \cdot 10^{-1}$	$6.6 \cdot 10^4$
5b	5	8	$1.4 \cdot 10^0$	$1.5 \cdot 10^5$
6b	6	11	$1.5 \cdot 10^0$	$2.3 \cdot 10^5$
7b	7	9	$4.1 \cdot 10^0$	$4.8 \cdot 10^5$
8b	8	13	$3.4 \cdot 10^0$	$5.5 \cdot 10^5$
9b	9	17	$4.3 \cdot 10^0$	$9.2 \cdot 10^5$
10b	10	11	$1.2 \cdot 10^1$	$1.3 \cdot 10^6$

Let us analyze the factors influencing each of these times. T^{exp} increases with the size of the system, i.e., a 6-cycle system will be computed faster than a 9-cycle system. On the other hand, T^{exp} does not generally depend on the molecular complexity of the system—compounds **8a** and **8b** required comparable times, despite **8b** appearing to be much more structurally diverse from a chemical perspective. Conversely, T^{BB} depends entirely on the molecular complexity. Because the largest building block is a tri-cycle, an individual building block calculation is not expensive or long, but having a large number of building blocks leads to an overall large T^{BB} . For these reasons, it is clear that the largest $T^{\text{exp}}/T^{\text{BB}}$ ratio is expected to be for a large system with a small number of building blocks. Indeed, **9a** is exactly such a case (#Cycles = 9, #BBs = 3, $T^{\text{exp}}/T^{\text{BB}} \approx 65$). In general, the PAH series (**4a-10a**) show that the ratio $T^{\text{exp}}/T^{\text{BB}}$ increases with the increase in molecular size. This is because the number of building blocks remains either 3 or 4, while the molecular size continuously increases.

For the PAS series (**4b-10b**), a similar trend appears, but it is less dramatic, because there are many more building blocks (from 7 to 17) for a similar set of molecular sizes. Nevertheless, we observe that the ability to predict the NICS-XY-Scan from building blocks enables cheaper computation, by approximately an order of magnitude.

Using Predi-XY, on the other hand, affords a dramatic shortening of time. As can be seen from Table 1, the ratio $T^{\text{exp}}/T^{\text{pred}}$ is extremely high in all cases – 5 orders of magnitude – demonstrating the efficiency of the automated method versus an explicit calculation. The average time of the Predi-XY calculations for the 14 test molecules was 0.11 seconds, with a standard deviation of 0.03 seconds, as calculated on a laptop running Windows 10 64-bit, equipped with an Intel core i7-7600U processor. Comparable timings were achieved on Linux and macOS systems tested in our group. With the plotting option turned on, the average time increased to 0.96 seconds with a standard deviation of 0.06 seconds.

The much more significant savings come from the existence of a library. Once the building blocks have been calculated and entered into the library, they never have to

be calculated again. For example, this means that all 7 compounds **4a-10a** can be calculated with the same 4 building blocks, which only needed to be calculated once. Moreover, countless other PAH isomers, in addition to the 7 shown here, can be calculated with the same library for no additional computational cost. This would be true even without the automated generation of NICS-XY-Scans by Predi-XY. However, even with the building blocks in hand, the process of manually piecing together the individual contributions of the predicted NICS-XY-Scan can be time-consuming and prone to error. Using Predi-XY avoids both the time and potential error, and the savings of time become very significant when screening a large number of molecules. Overall, the program makes the process of generating NICS-XY-Scans easy, fast, and user-friendly.

Furthermore, for users interested in a large number of systems, we note that the very short time for a Predi-XY

calculation (on the order of a single second) can be cut down by another 1-2 orders of magnitude if no plotting is requested (for users requiring only text-based results).

To show the accuracy of the predicted results, we present in Figure 11 the superposition of the explicitly calculated NICS-XY-Scans (in black) and the scans generated by Predi-XY (in red) for all molecules in the collection of test molecules. The agreement is more than satisfactory, especially in light of the dramatically reduced effort in obtaining the predicted scan. In all cases, the trends are reproduced by the predicted scans. Most deviations are on the order of ~1-2 ppm and the largest ones are ~5 ppm, which does not impede interpretation of the scan.

Figure 11. Plots comparing the explicitly calculated NICS-XY-Scan (black) and the scan predicted by Predi-XY (red) for selected test molecules. With each plot we show the scheme of the calculated system. The direction of the scan is consistent with the direction of “reading” the molecule left-to-right. The explicit scans were calculated with B3LYP/6-311+G(d).

Conclusions

In this report, we have introduced Predi-XY, a new Python-based program for automated generation of predicted NICS-XY-Scans of polycyclic aromatic systems using an underlying additivity scheme. Predi-XY can be used on

Windows, macOS, and Linux machines, or other Python-enabled environments through command line prompts. This user-friendly program, which is freely available from our group’s GitLab repository, allows users to rapidly obtain the NICS-XY-Scans of countless systems. NICS-XY-

Scans are useful and popular tools for analyzing the aromatic behavior of molecules. They have been used to understand the reactivity and to better design new functional materials.^{23,24} With Predi-XY, the computational cost for generating NICS-XY-Scans is reduced by orders of magnitude, to mere seconds. This allows quick comparison of different systems and even circumvention of quantum mechanical calculations. We believe this tool will be useful to synthetic and computational chemists aiming to investigate the electronic structure of polycyclic aromatic systems.

The current library of Predi-XY contains more than 100 building blocks, which can be used to generate NICS-XY-Scans for a vast number of PASs containing various structural motifs. We envision that this library will continue to grow as our group generates additional data in the course of our research. We also encourage users to augment the library with new building blocks.

AUTHOR INFORMATION

Corresponding Author

* Dr. Renana Gershoni-Portanne
renana.portanne@org.chem.ethz.ch

LINK TO CODE

To download Predi-XY, please visit our group's public repository: <https://gitlab.com/portannegroup/predixy/-/releases> or install with pip using the command "pip install predixy".

ACKNOWLEDGMENTS

The authors wish to express their gratitude to Dr. Robert Pollice and Eno Paenurk for helpful comments on the development of the program. We also thank Dr. Alexandra Tsybizova and Dominic Egger for their constructive criticism on the manuscript. R.G.P. acknowledges the generous financial support of the Branco Weiss Fellowship. F.F. and R.G.P. are grateful for the support and mentorship of Prof. Peter Chen. In addition, the authors wish to thank the anonymous reviewers for their helpful suggestions for improvements to the code.

REFERENCES

- (1) Aihara, J. Why Aromatic Compounds Are Stable. *Sci. Am.* **1992**, 266 (3), 62–68. <https://doi.org/10.1038/scientificamerican0392-62>.
- (2) Fernández, I. Direct Estimate of Aromaticity with the Energy Decomposition Analysis. *The Open Organic Chemistry Journal* **2011**, 5 (1), 79–86. <https://doi.org/10.2174/1874364101105010079>.
- (3) Dewar, M. J. S. Aromaticity and Pericyclic Reactions. *Angew. Chem., Int. Ed. Engl.* **1971**, 10 (11), 761–776. <https://doi.org/10.1002/slct.201601384>.
- (4) Domingo, L. R.; Rios-Gutierrez, M.; Chamorro, E.; Perez, P. Aromaticity in Pericyclic Transition State Structures? A Critical Rationalisation Based on the Topological Analysis of Electron Density. *ChemistrySelect* **2016**, 1 (18), 6026–6039. <https://doi.org/10.1002/anie.197107611>.
- (5) Krygowski, T. M.; Cyranski, M. K. Structural Aspects of Aromaticity. *Chem. Rev. (Washington, D. C.)* **2001**, 101 (5), 1385–1419. <https://doi.org/10.1021/cr990326u>.
- (6) Schleyer, P. von R.; Pühlhofer, F. Recommendations for the Evaluation of Aromatic Stabilization Energies. *Org. Lett.* **2002**, 4 (17), 2873–2876. <https://doi.org/10.1021/ol0261332>.
- (7) Gershoni-Portanne, R.; Stanger, A. Magnetic Criteria of Aromaticity. *Chem. Soc. Rev.* **2015**, 44 (18), 6597–6615.
- (8) Poater, J.; Fradera, X.; Duran, M.; Sola, M. The Delocalization Index as an Electronic Aromaticity Criterion: Application to a Series of Planar Polycyclic Aromatic Hydrocarbons. *Chem. - Eur. J.* **2003**, 9 (2), 400–406. <https://doi.org/10.1002/chem.200390041>.
- (9) Fowler, P. W.; Myrvold, W. The "Anthracene Problem": Closed-Form Conjugated-Circuit Models of Ring Currents in Linear Polyacenes. *J. Phys. Chem. A* **2011**, 115 (45), 13191–13200. <https://doi.org/10.1021/jp206548t>.
- (10) Balaban, A. T.; Randić, M. Structural Approach to Aromaticity and Local Aromaticity in Conjugated Polycyclic Systems. In *Carbon Bonding and Structures: Advances in Physics and Chemistry*; Putz, M. V., Ed.; Carbon Materials: Chemistry and Physics; Springer Netherlands: Dordrecht, 2011; pp 159–204. https://doi.org/10.1007/978-94-007-1733-6_8.
- (11) Gershoni-Portanne, R.; Gibson, C. M.; Fowler, P. W.; Stanger, A. Concurrence between Current Density, Nucleus-Independent Chemical Shifts, and Aromatic Stabilization Energy: The Case of Isomeric 4- and 5-Phenylenes. *J. Org. Chem.* **2013**, 78 (15), 7544–7553. <https://doi.org/10.1021/jo4011014>.
- (12) Zhou, Z.; Kawade, R. K.; Wei, Z.; Kuriakose, F.; Üngör, Ö.; Jo, M.; Shatruk, M.; Gershoni-Portanne, R.; Petrukhina, M. A.; Alabugin, I. V. Negative Charge as a Lens for Concentrating Antiaromaticity: Using a Pentagonal "Defect" and Helicene Strain for Cyclizations. *Angewandte Chemie International Edition* **2020**, 59 (3), 1256–1262. <https://doi.org/10.1002/anie.201911319>.
- (13) Anthony, J. E. Functionalized Acenes and Heteroacenes for Organic Electronics. *Chem. Rev. (Washington, DC, U. S.)* **2006**, 106 (Copyright (C) 2019 American Chemical Society (ACS). All Rights Reserved.), 5028–5048. <https://doi.org/10.1021/cr050966z>.
- (14) Facchetti, Antonio. π -Conjugated Polymers for Organic Electronics and Photovoltaic Cell Applications. *Chem. Mater.* **2011**, 23 (Copyright (C) 2019 American Chemical Society (ACS). All Rights Reserved.), 733–758. <https://doi.org/10.1021/cm102419z>.
- (15) Tovar, J. D. Prospecting in Hückel-Space: From Hinokitiol to Non-Benzenoid Organic Electronics. *The Chemical Record* **2014**, 14 (1), 214–225. <https://doi.org/10.1002/tcr.201300034>.
- (16) Aumaitre, C.; Morin, J.-F. Polycyclic Aromatic Hydrocarbons as Potential Building Blocks for Organic Solar Cells. *The Chemical Record* **2019**, 19 (6), 1142–1154. <https://doi.org/10.1002/tcr.201900016>.
- (17) Marshall, J. L.; Uchida, K.; Frederickson, C. K.; Schütt, C.; Zeidell, A. M.; Goetz, K. P.; Finn, T. W.; Jarolimek, K.; Zakharov, L. N.; Risko, C.; Herges, R.; Jurchescu, O. D.; Haley, M. M. Indacenodibenzothiophenes: Synthesis, Optoelectronic Properties and Materials Applications of Molecules with Strong Antiaromatic Character. *Chemical Science* **2016**, 7 (8), 5547–5558. <https://doi.org/10.1039/C6SC00950F>.
- (18) Zeidell, A. M.; Jennings, L.; Frederickson, C. K.; Ai, Q.; Dressler, J. J.; Zakharov, L. N.; Risko, C.; Haley, M. M.; Jurchescu, O. D. Organic Semiconductors Derived from Dinaphthofused s-Indacenes: How Molecular Structure and Film Morphology Influence Thin-Film Transistor Performance. *Chem. Mater.* **2019**, 31 (17), 6962–6970. <https://doi.org/10.1021/acs.chemmater.9b01436>.
- (19) Fix, A. G.; Deal, P. E.; Vonnegut, C. L.; Rose, B. D.; Zakharov, L. N.; Haley, M. M. Indeno[2,1-c]Fluorene: A New Electron-Accepting Scaffold for Organic Electronics. *Org. Lett.* **2013**, 15 (Copyright (C) 2019 American Chemical Society (ACS).

- All Rights Reserved.), 1362–1365. <https://doi.org/10.1021/ol400318z>.
- (20) Jousselin-Oba, T.; Deal, P. E.; Fix, A. G.; Frederickson, C. K.; Vonnegut, C. L.; Yassar, A.; Zakharov, L. N.; Frigoli, M.; Haley, M. M. Synthesis and Properties of Benzo-Fused Indeno[2,1-c]Fluorenes. *Chemistry – An Asian Journal* **2019**, *14* (10), 1737–1744. <https://doi.org/10.1002/asia.201801684>.
 - (21) Gershoni-Oran, R.; Rahalkar, A. P.; Stanger, A. The Predictive Power of Aromaticity: Quantitative Correlation between Aromaticity and Ionization Potentials and HOMO–LUMO Gaps in Oligomers of Benzene, Pyrrole, Furan, and Thiophene. *Phys. Chem. Chem. Phys.* **2018**, *20* (21), 14808–14817. <https://doi.org/10.1039/C8CP02162G>.
 - (22) Cao, J.; London, G.; Dumele, O.; von Wantoch Rekowski, M.; Trapp, N.; Ruhlmann, L.; Boudon, C.; Stanger, A.; Diederich, F. The Impact of Antiaromatic Subunits in $[4n+2]$ π -Systems: Bispentalenes with $[4n+2]$ π -Electron Perimeters and Antiaromatic Character. *J. Am. Chem. Soc.* **2015**, *137* (22), 7178–7188. <https://doi.org/10.1021/jacs.5b03074>.
 - (23) Frederickson, C. K.; Zakharov, L. N.; Haley, M. M. Modulating Paratropicity Strength in Diareno-Fused Antiaromatics. *Journal of the American Chemical Society* **2016**, *138* (51), 16827–16838. <https://doi.org/10.1021/jacs.6b11397>.
 - (24) Rudebusch, G. E.; Zafra, J. L.; Jorner, K.; Fukuda, K.; Marshall, J. L.; Arrechea-Marcos, I.; Espejo, G. L.; Ponce Ortiz, R.; Gómez-García, C. J.; Zakharov, L. N.; Nakano, M.; Ottosson, H.; Casado, J.; Haley, M. M. Diindeno-Fusion of an Anthracene as a Design Strategy for Stable Organic Biradicals. *Nature Chemistry* **2016**, *8* (8), 753–759. <https://doi.org/10.1038/nchem.2518>.
 - (25) Schleyer, P. v R.; Maerker, C.; Dransfeld, A.; Jiao, H.; van Eikema Hommes, N. J. R. Nucleus-Independent Chemical Shifts: A Simple and Efficient Aromaticity Probe. *J. Am. Chem. Soc.* **1996**, *118* (26), 6317–6318. <https://doi.org/10.1021/JA960582D>.
 - (26) Chen, Z.; Wannere, C. S.; Corminboeuf, C.; Puchta, R.; Schleyer, P. von R. Nucleus-Independent Chemical Shifts (NICS) as an Aromaticity Criterion. *Chem. Rev. (Washington, DC, U. S.)* **2005**, *105* (10), 3842–3888. <https://doi.org/10.1021/cr030088+>.
 - (27) Stanger, A. NICS – Past and Present. *Eur. J. Org. Chem.* **2020**, *2020* (21), 3120–3127. <https://doi.org/10.1002/ejoc.201901829>.
 - (28) Gomes, J. A. N. F.; Mallion, R. B. Aromaticity and Ring Currents. *Chem. Rev. (Washington, D. C.)* **2001**, *101* (5), 1349–1383. <https://doi.org/10.1021/cr990323h>.
 - (29) Dickens, T. K.; Mallion, R. B. Topological Ring-Currents in Conjugated Systems. *MATCH Commun. Math. Comput. Chem.* **2016**, *76*, 297–356.
 - (30) Gershoni-Oran, R.; Stanger, A. The NICS-XY-Scan: Identification of Local and Global Ring Currents in Multi-Ring Systems. *Chem. - Eur. J.* **2014**, *20* (19), 5673–5688. <https://doi.org/10.1002/chem.201304307>.
 - (31) Stanger, A.; Monaco, G.; Zanasi, R. NICS-XY-Scan Predictions of Local, Semi-Global, and Global Ring Currents in Annulated Pentalene and s-Indacene Cores Compared to First-Principles Current Density Maps. *ChemPhysChem* **2020**, *21* (n/a), 65–82. <https://doi.org/10.1002/cphc.201900952>.
 - (32) Chen, D.; Shen, T.; An, K.; Zhu, J. Adaptive Aromaticity in S0 and T1 States of Pentalene Incorporating 16 Valence Electron Osmium. *Communications Chemistry* **2018**, *1* (1). <https://doi.org/10.1038/s42004-018-0018-y>.
 - (33) Lampkin, B. J.; Karadakov, P. B.; VanVeller, B. Detailed Visualization of Aromaticity Using Isotropic Magnetic Shielding. *Angewandte Chemie International Edition* **2020**, *59* (43), 19275–19281. <https://doi.org/10.1002/anie.202008362>.
 - (34) Rahalkar, A. P.; Stanger, A. “Aroma”, http://Schulich.Tech-nion.Ac.II/Amnon_Stanger.Htm.
 - (35) Stanger, A. Nucleus-Independent Chemical Shifts (NICS): Distance Dependence and Revised Criteria for Aromaticity and Antiaromaticity. *J. Org. Chem.* **2006**, *71* (3), 883–893. <https://doi.org/10.1021/jo051746o>.
 - (36) Stanger, A. Obtaining Relative Induced Ring Currents Quantitatively from NICS. *J. Org. Chem.* **2010**, *75* (7), 2281–2288. <https://doi.org/10.1021/jo1000753>.
 - (37) Frisch, M. J.; Trucks, G. W.; Schlegel, H. B.; Scuseria, G. E.; Robb, M. A.; Cheeseman, J. R.; Scalmani, G.; Barone, V.; Mennucci, B.; Petersson, G. A.; Nakatsuji, H.; Caricato, M.; Li, X.; Hratchian, H. P.; Izmaylov, A. F.; Bloino, J.; Zheng, G.; Sonnenberg, J. L.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida, M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Vreven, T.; Montgomery, J. A.; Peralta, J. E.; Ogliaro, F.; Bearpark, M.; Heyd, J. J.; Brothers, E.; Kudin, K. N.; Staroverov, V. N.; Kobayashi, R.; Normand, J.; Raghavachari, K.; Rendell, A.; Burant, J. C.; Iyengar, S. S.; Tomasi, J.; Cossi, M.; Rega, N.; Millam, J. M.; Klene, M.; Knox, J. E.; Cross, J. B.; Bakken, V.; Adamo, C.; Jaramillo, J.; Gomperts, R.; Stratmann, R. E.; Yazyev, O.; Austin, A. J.; Cammi, R.; Pomelli, C.; Ochterski, J. W.; Martin, R. L.; Morokuma, K.; Zakrzewski, V. G.; Voth, G. A.; Salvador, P.; Dannenberg, J. J.; Dapprich, S.; Daniels, A. D.; Farkas, Foresman, J. B.; Ortiz, J. V.; Cioslowski, J.; Fox, D. J. Gaussian 09, Revision B.01. Gaussian 09, Revision B.01, Gaussian, Inc., Wallingford CT **2009**.
 - (38) Gershoni-Oran, R. Piecing It Together: An Additivity Scheme for Aromaticity Using NICS-XY-Scans. *Chemistry – A European Journal* **2018**, *24* (16), 4165–4172. <https://doi.org/10.1002/chem.201705407>.
 - (39) Finkelstein, P.; Gershoni-Oran, R. An Additivity Scheme for Aromaticity: The Heteroatom Case. *ChemPhysChem* **2019**, *20*, 1508–1520. <https://doi.org/10.1002/cphc.201900128>.
 - (40) Van Rossum, G.; Drake, F. L. *Python 3 Reference Manual*; Scotts Valley, CA: CreateSpace, 2009.
 - (41) Benson, S. W.; Cruickshank, F. R.; Golden, D. M.; Haugen, G. R.; O’Neal, H. E.; Rodgers, A. S.; Shaw, R.; Walsh, R. Additivity Rules for the Estimation of Thermochemical Properties. *Chem. Rev.* **1969**, *69* (3), 279–324. <https://doi.org/10.1021/cr60259a002>.
 - (42) Fallah-Bagher-Shaidaei, H.; Wannere, C. S.; Corminboeuf, C.; Puchta, R.; Schleyer, P. v R. Which NICS Aromaticity Index for Planar π Rings Is Best? *Org. Lett.* **2006**, *8* (5), 863–866. <https://doi.org/10.1021/ol0529546>.
 - (43) Bohmann, J. A.; Weinhold, F.; Farrar, T. C. Natural Chemical Shielding Analysis of Nuclear Magnetic Resonance Shielding Tensors from Gauge-Including Atomic Orbital Calculations. *J. Chem. Phys.* **1997**, *107* (4), 1173–1184. <https://doi.org/10.1063/1.474464>.
 - (44) Glendening, E. D.; Landis, C. R.; Weinhold, F. Natural Bond Orbital Methods. *WIREs Computational Molecular Science* **2012**, *2* (1), 1–42. <https://doi.org/10.1002/wcms.51>.
 - (45) Cordero, B.; Gómez, V.; Platero-Prats, A. E.; Revés, M.; Echeverría, J.; Cremades, E.; Barragán, F.; Alvarez, S. Covalent Radii Revisited. *Dalton Trans.* **2008**, No. 21, 2832–2838. <https://doi.org/10.1039/B801115J>.
 - (46) Proceedings of the Python in Science Conference (SciPy): Exploring Network Structure, Dynamics, and Function using NetworkX http://conference.scipy.org/proceedings/SciPy2008/paper_2/ (accessed Sep 21, 2020).
 - (47) McKinney, W. Data Structures for Statistical Computing in Python; Austin, Texas, 2010; pp 56–61. <https://doi.org/10.25080/Majora-92bf1922-00a>.
 - (48) Jeff Reback; Wes McKinney; jbrockmendl; Joris Van den Bossche; Tom Augspurger; Phillip Cloud; gyoung; Sinhrks; Adam Klein; Matthew Roeschke; Simon Hawkins; Jeff Tratner; Chang She; William Ayd; Terji Petersen; Marc Garcia; Jeremy Schendel; Andy Hayden; MomIsBestFriend; Vytautas Jancauskas; Pietro Battiston; Skipper Seabold; chris-b1; h-vetinari; Stephan Hoyer; Wouter Overmeire; alimcmaster1; Kaiqi Dong; Christopher Whelan; Mortada Mehyar. *Pandas*

Dev/Pandas: Pandas 1.0.3; Zenodo, 2020.
<https://doi.org/10.5281/zenodo.3715232>.