



A general informatic tool for treating solution equilibria in supramolecular chemistry

Riccardo Salvio^{1,2}

Received: 27 May 2026 / Accepted: 16 July 2026 / Published online: 14 August 2026
© The Author(s) 2026

Abstract

This work presents a versatile, open-source computational framework for the quantitative analysis of equilibria in solution, with particular emphasis on problems commonly encountered in supramolecular chemistry. The approach is based on a general numerical treatment of complex equilibrium systems, overcoming the limitations of analytical solutions that are restricted to only the simplest cases. The framework is implemented through a series of MATLAB scripts that are fully transparent, easily modifiable, and freely available, thereby avoiding the “black box” character of many commercial programs and enabling users to adapt the workflow to their specific experimental systems. The capabilities and performance of the tool are illustrated through its application to a range of representative datasets, including real experimental data acquired in the course of our own research activities. We demonstrate its utility for the analysis of host–guest complexation and acid–base titrations in which complexation phenomena are involved, thereby illustrating its reliability and broad applicability to recurring problems in supramolecular chemistry. At the same time, the methodology retains general validity for the treatment of almost any equilibrium process occurring in solution. By providing detailed documentation of the underlying mathematical and computational procedures, as well as complete access to the original scripts, this work promotes reproducibility and offers a flexible platform that can be readily employed to test mechanistic hypotheses and readily extended to address new and more elaborate systems.

Keywords Equilibria in solution · Numerical data analysis · Mechanistic hypothesis testing · Equilibria in solution · Host–guest complexation · Acid–base titrations · Matlab open-source scripts

Dedicated to the memory of Prof. Luigi Mandolini.

Extended author information available on the last page of the article

1 Introduction

A fundamental aspect of physical organic chemistry and supramolecular chemistry is the quantitative analysis of data obtained from a wide range of experimental measurements. Adequate and rigorous treatment of such data is crucial for drawing sound conclusions and testing hypotheses that have been proposed. For extremely simple systems, it is possible to analytically solve the sets of equations underlying equilibria and to apply the formulas reported in scientific literature and textbooks [1–5]. However, even for slightly more complex systems an analytical solution is not possible, and it therefore becomes necessary to resort to numerical methods in order to solve the equations underlying the equilibria under study. Consequently, to address these problems it is possible to rely on commercial software or online resources. Prominent examples include Hyperquad/HySS [6, 7], MINEQL+ [8], MINFIT [9], and online tools such as supramolecular.org [10]. In these environments, equilibrium calculations are typically handled by robust, well-tested solvers and the user benefits from graphical interfaces, integrated databases of stability constants, and ready-made routines for data fitting and speciation analysis. However, the main drawback of these applications, from our perspective, is the limited control that the user can exert over the underlying numerical workflow and working equations, which may effectively become a kind of “black box”. In addition, most of these tools are not designed to be easily modified or extended to address user-specific needs, and their output is not always straightforward to integrate into automated workflows, for instance when coupling equilibrium calculations to kinetic models or to large-scale data analysis scripts.

Hyperquad/HySS [6, 7] provide excellent tools for refining stability constants and visualizing speciation diagrams, but they are mainly oriented toward data fitting and do not expose the equilibrium solver in a form that can be easily repurposed or embedded into custom computational workflows. MINEQL+ [8] offers a robust environment for aqueous speciation and solubility calculations with an extensive thermodynamic database, yet its numerical engine and equation setup remain enclosed within a proprietary framework, limiting low-level customization and scripted access to the calculation internals. MINFIT [9], a spreadsheet-based tool for parameter estimation in equilibrium speciation programs, provides a convenient interface for fitting and uncertainty analysis, but it is primarily designed as an add-on for parameter refinement rather than as a general-purpose, scriptable equilibrium solver that can be directly integrated into broader computational pipelines. Finally, online resources such as supramolecular.org [10] greatly facilitate the exploration of host–guest binding isotherms and speciation in a browser-based interface, but they typically offer only a restricted set of predefined models and lack direct access to the underlying numerical routines or to a scripting interface suitable for large-scale or heavily customized studies.

In the present work, we therefore address the need, shared by many researchers, for a simple tool that is open source and completely free, easily adaptable to a very wide variety of cases, and explicitly designed to expose both the mathematical formulation and the solver logic. By implementing the framework in a scripting environment and documenting in detail the numerical strategies and working equations,

we aim to provide a solution that can be seamlessly reused, extended, and integrated into broader computational pipelines, including automated equilibrium-kinetic coupling, high-throughput screening of speciation scenarios, and reproducible data-analysis workflows.

Thus, we present this tool, describing the mathematical and computational approaches employed, and illustrating its application to common and recurring problems in supramolecular chemistry, although the proposed data treatment is not limited to this field. In addition, the tool has been applied to real data acquired by our research group and in the course of our research activities. Moreover, in the Methods section we report all details concerning the setup and possible modification of the MATLAB scripts, while in the Supporting Information the original scripts are provided, enabling the readers to reproduce all program runs presented here, to verify their performance, and to use them by easily and promptly adapting them to their own purposes. This study seeks to establish a comprehensive framework for addressing a broad spectrum of problems, exemplified through empirical case studies, while refraining from revisiting the rigorous mathematical formulations and technical elaboration illustrated in prior contributions. [11–16]

2 Results and discussion

2.1 Single point calculation

We first consider a simple model system in which species A and another species B can form both 1:1 (AB) and 1:2 (AB_2) complexes. The temperature is fixed and no activity corrections are introduced, so that equilibrium is entirely described in terms of molar concentrations. This simple setting initially serves as a demonstrative example to illustrate a general equilibrium-speciation informatic tool that can handle a wide range of stoichiometries and binding scenarios and it is designed to be easily extensible to more complex systems.

The program represents the chemical system through an explicit list of species, a set of independent components, and a charge balance. A stoichiometric matrix links each species to the components and to the overall charge, so that any speciation vector can be mapped onto the corresponding totals of components and charge. The reactions defining complex formation are encoded in a reaction stoichiometric matrix together with their equilibrium constants, expressed in logarithmic form. Within this structure, extending the model to new complexes or entities, additional components, or different charge states only requires augmenting the species list and the associated stoichiometric and equilibrium data, without modifying the core. For all the details regarding the structure of the scripts and the definition of the species, the components and the constants refer to the Method Section and the Supporting Information.

From a numerical standpoint, the algorithm distinguishes between *primary* and *secondary* species. The primary variables are the free concentrations of a chosen subset of species, while the concentrations of all remaining (secondary) species are reconstructed using the equilibrium relations and the current values of the primary variables. To improve robustness over potentially wide concentration

ranges, the primary variables are parameterized in $\log_{10}$ -space. Given a trial vector of log-concentrations for the primary species, the program reconstructs all species concentrations and uses the stoichiometric matrix to compute the implied totals of each component and the total charge. A nonlinear residual is then formed as the difference between these calculated totals and the prescribed bulk values.

The resulting system of nonlinear equations is solved with a trust-region method of the Levenberg–Marquardt type [17]. To limit sensitivity to initial guesses and to accommodate systems with strong associations or multiple possible speciation regimes, the program adopts a multi-start strategy in log-space. A small grid of initial guesses is constructed in the $\log_{10}$ -space of the primary variables, spanning a range from very diluted states to nearly fully associated states. For each grid point, the nonlinear solver is started independently, subject to strict tolerances on both the function value and the step size, in order to ensure accurate convergence for equilibrium speciation.

After all solver runs, the program selects the final solution based on conservative balance criteria. For each candidate, it reconstructs the full speciation vector and evaluates the relative errors in each component balance and in the overall charge balance. The maximum of these relative errors is taken as a global balance error and used as the primary ranking criterion; the residual norm of the nonlinear equations is used as a secondary criterion to break ties. A solution is accepted as physically meaningful only if the solver has formally converged and the global balance error lies below a prescribed tolerance. For the illustrative A/B system presented above, the algorithm identifies a unique equilibrium speciation that satisfies mass and charge conservation within the target accuracy.

Although the example involves only four species and two components, the program is conceived to treat a very large number of different cases with the same numerical core. The separation between model definition (lists of species, components, stoichiometric matrices, equilibrium constants) and solver logic (log-space parametrization, multi-start nonlinear solution, balance-based selection) allows users to adapt the code to arbitrary networks of equilibria. In this way, the same framework can later be used for titration calculations, multicomponent systems, or models including activity corrections and temperature dependence, simply by modifying the input data rather than the underlying algorithm.

In the present implementation, solvent effects are treated implicitly through activity coefficients rather than by introducing explicit solvent species in the equilibrium scheme. For dilute solutions, where the solvent concentration is much larger than that of all solutes and its relative variation during a titration is negligible, this approach is fully adequate. In such cases, the replacement of concentrations with activities $a_A = \gamma_A[A]$ and of concentration-based equilibrium constants with their activity-based counterparts can be implemented directly at the level of the input data, without modifying the numerical core of the algorithm. The activity coefficients γ may be obtained from standard thermodynamic models (e.g., Debye–Hückel-type expressions or other more sophisticated models) or from experimental measurements (electrochemical and pH potentiometry, colligative properties, comparison between apparent and thermodynamic equilibrium constants). In typical supramolecular experiments, where the ionic strength is buffered [18, 19], can be regarded as constant over the titration

range, so that activity corrections have negligible additional computational cost and do not affect the robustness of convergence. By contrast, extending the framework to concentrated or unbuffered solutions, in which activity coefficients vary significantly with composition and ionic strength, would require a tighter coupling between the speciation problem and a composition-dependent activity model, with a corresponding increase in the number of parameters and potential numerical difficulties. Systematic treatment of these more complex regimes lies beyond the scope of the present work and will be the subject of future developments.

In the example reported above, A and B can be a ligand and a positively charged cation undergoing a 1:1 and 1:2 equilibria. In this case the species are A , B^+ , AB^+ , AB_2^{2+} . The species are built from two independent components: A and B^+ . The complexes are formed according to the equilibria (1) and (2).



In the general program, this list of species and reactions can be extended arbitrarily by adding more components, species, and equilibria. We denote the vector $\mathbf{C}$ of species concentrations as given in (3). The “components” are conservation quantities, and their total concentrations are collected in $\mathbf{b}$ defined as described in (4), where b_{charge} is the total charge in solution. A stoichiometric matrix N links species to components and charge via Eq. (5)

$$\mathbf{C} = \begin{bmatrix} [A] \\ [B^+] \\ [AB^+] \\ [AB_2^{2+}] \end{bmatrix}. \quad (3)$$

$$\mathbf{b} = \begin{bmatrix} b_A \\ b_B \\ b_{charge} \end{bmatrix} \quad (4)$$

$$\mathbf{b} = N \mathbf{C}. \quad (5)$$

For the illustrative system, N has three rows: one for A, one for B, and one for charge. The structure of N guarantees that, for any candidate speciation $\mathbf{C}$, the corresponding totals $\mathbf{b}$ (including charge) can be computed consistently.

The mass-action equilibria are encoded through a reaction stoichiometric matrix ν and a vector of equilibrium constants K , stored in logarithmic form as $\log_{10} K$. For each equilibrium r

$$\sum_i \nu_{ir} \text{Species}_i = 0 \quad (6)$$

$$K_r = \prod_i [\textit{Species}_i]^{\nu_{ir}}. \quad (7)$$

The distinction between primary and secondary species is central to the numerical formulation. Primary species are those whose free concentrations are treated as independent unknowns. On the other hand, secondary species are all remaining species whose concentrations are computed (or *reconstructed*) from the primary ones using the equilibrium relations. In the example, the primary species are chosen as follows: primary species: A, B⁺, and secondary species: AB⁺ and AB₂²⁺.

This means that the algorithm solves for [A] and [B⁺] (in log-space), while [AB⁺] and [AB₂²⁺] are obtained from (8) and (9)

$$[AB^+] = K_1 [A] [B^+], \quad (8)$$

$$[AB_2^{2+}] = K_2 [AB^+] [B^+] = K_1 K_2 [A] [B^+]^2 \quad (9)$$

In the general code, these relations are not hard-coded but generated automatically from ν and K . In other words, primary species define the independent degrees of freedom, while secondary species are dependent variables determined by equilibrium constraints, e.g. (8) and (9). This reduces the size of the nonlinear system and improves numerical robustness.

To handle potentially wide ranges of concentration, the program parameterizes primary species in logarithmic form. For the two primary species we define vector $\mathbf{x}$ as in Eq. (10)

$$\mathbf{x} = \begin{bmatrix} \log_{10}[A] \\ \log_{10}[B^+] \end{bmatrix} \quad (10)$$

Given $\mathbf{x}$, the algorithm proceeds as follows: (a) convert log-concentrations back to linear concentrations for the primary species according to (11); (b) reconstruct all secondary species by applying the mass-action laws, see Eq. (12) where f_1 and f_2 are general functions derived from ν and K .

$$[A] = 10^{x_1}, [B^+] = 10^{x_2} \quad (11)$$

$$[AB^+] = f_1([A], [B^+]; K), [AB_2^{2+}] = f_2([A], [B^+]; K), \quad (12)$$

The next steps of the algorithm are: (c) assembling the full concentration vector $\mathbf{C}(\mathbf{x})$ by combining primary and reconstructed secondary species and (d) computing the implied totals using the stoichiometric matrix (13).

$$\mathbf{b}_{calc}(\mathbf{x}) = N \mathbf{C}(\mathbf{x}) \quad (13)$$

Given prescribed bulk totals $\mathbf{b}_{tot}$ (component totals and charge), the residual of the nonlinear problem is in Eq. (14)

$$\mathbf{F}(\mathbf{x}) = \mathbf{b}_{calc}(\mathbf{x}) - \mathbf{b}_{tot} \quad (14)$$

Equilibrium speciation corresponds to finding $\mathbf{x}$ such that $\mathbf{F}(\mathbf{x}) = 0$. *i.e.* mass and charge balances are exactly satisfied. The vector equation $\mathbf{F}(\mathbf{x}) = 0$ is solved with a Levenberg–Marquardt–type trust-region method. Because equilibrium problems can be ill-conditioned and sensitive to initial guesses, especially in the presence of strong association or multiple regimes, the program uses a multi-start strategy in log-space.

A small grid of initial guesses $\{\mathbf{x}_0^{(k)}\}$ is constructed by sampling the $\log_{10}$ -space of the primary species over a range that spans: *a*) very low concentrations (dilute, mostly unassociated), *b*) intermediate states and *c*) high concentrations (nearly all bound or dominated by certain species).

For each initial guess $\mathbf{x}_0^{(k)}$, the solver attempts to find a root of $\mathbf{F}(\mathbf{x})$ under strict tolerances on the norm $\|\mathbf{F}(\mathbf{x})\|$ and the step size between iterations. This yields a set of candidate solutions $\{\mathbf{x}^{(k)}\}$, each with a corresponding speciation $\mathbf{C}^{(k)}$. For each candidate solution, the program evaluates the quality of mass and charge conservation. A candidate is flagged as successful only if the nonlinear solver reports convergence and $\varepsilon_{bal}^{(k)}$ is below a prescribed tolerance. The computational details regarding the selection and the validation of the solution are described in the Experimental Section.

As mentioned above, although the example involves only four species and two components, the structure of the program is explicitly designed to handle a large number of systems. The change of the chemical system (number and type of components, species, and reactions) can be done by modifying the following parts: the species list, the component list, the stoichiometric matrices N and v , and the equilibrium constants. On the other hand the following part of the numerical core remain unchanged: the choice of primary species and treatment of secondary species via equilibrium relations, the log-space parametrization of primary concentrations, the multi-start solution of $\mathbf{F}(\mathbf{x})=0$ of with a trust-region method, and the selection of the best solution based on mass and charge balance.

In this way, the same algorithm that solves the simple A/B system can be used for multi-component speciation, or more sophisticated models including activity corrections and temperature dependence, as long as the corresponding stoichiometry and equilibrium data are provided.

Numerous tests were carried out to assess the robustness of the method, including edge cases in which the constants are either very large or very small, thus shifting the equilibrium significantly toward the dissociated species or the complexes, as well as cases where the constants differ markedly from each other, consistently yielding results in agreement with the chemical systems under study. Furthermore, the method proved robust even when the species exhibited large differences in concentration. In this regard, we performed a series of numerical tests on single-point calculations for several cases, including challenging situations such as large disparities between binding constants, pronounced differences in components concentrations, and titration inflection points (see section S9 of the Supporting Information).

2.2 Titration

Once the consistency and stability of the single-point calculation scripts have been verified, the next step is to implement another branch of the program capable of using the tool developed thus far to generate a titration experiment.

This module implements a simulated titration in which a solute A undergoes two reversible association equilibria with a titrant B to form 1:1 (AB) and 1:2 (AB_2) complexes, according to the test case taken in consideration for the single point calculations. In the present implementation, the total concentration of A is kept fixed, while the total concentration of B is varied over a prescribed range of equivalents with respect to A. For each value of the equivalents of B, the corresponding analytical totals of A and B (and of the overall charge) are assembled and passed to the general equilibrium solver.

Although the illustrative example focuses on the A/B system with species A, B^+ , AB^+ , and AB_2^{2+} , the underlying logic is readily extensible to other chemical systems in which species are linked by known mass–action relations (e.g., metal–ligand complexes, acid–base speciation, self-association). In practice, adapting the titration script to a different problem mainly requires modifying the model definition (list of species and components, stoichiometric matrices, and equilibrium constants), while the titration loop and the call to the solver remain unchanged.

For each titration point, the script records the equilibrium concentrations of all species, along with diagnostic quantities from the solver (residual norm, exit flag, and mass/charge balances), and stores them in a results table. In addition, it generates a plot of the equilibrium concentrations of A, B, AB, and AB_2 as a function of the equivalents of B added, thus providing a simulated titration curve that can be directly compared with experimental data or used to explore the influence of the association constants on the speciation profile. Figure 1 shows the titration curve of A with B, illustrating the concentration of the resulting species in two cases: when the second binding constant of B to A is lower than the first, and in the cooperative case, i.e. the second binding constant exceeds the first.

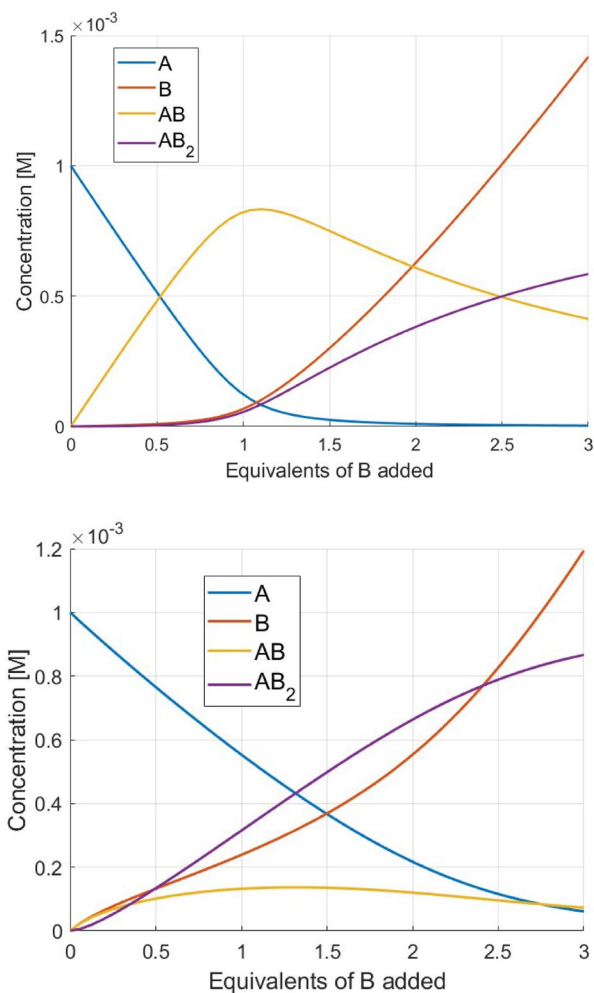
The approach presented here is fully modular as the numerical solution of the chemical equilibria is fully encapsulated in a separate routine, which receives the total concentrations and equilibrium constants and returns the equilibrium concentrations of all species.

The main script remains agnostic to the specific reaction network and can, in principle, handle more complex speciation models (multiple ligands, competitive binding, polyprotic acids) provided the solver is configured accordingly.

2.3 Acid–base equilibria

This module implements a general numerical framework for the titration of a polyprotic acid H_nA with a strong base, explicitly accounting for the successive deprotonation equilibria and for water autoprotolysis. In Eqs. 15–17 are reported the equilibria in the case of a diprotic acid. For a given set of acid dissociation constants $\{K_{a,1}, \dots, K_{a,n}\}$ and a water ion product K_w , the program simulates the titration of an initial acid solution of concentration C_{acid} and volume V_{acid} by a strong base

Fig. 1 Calculated concentration profiles of free A and B, and of their 1:1 (AB) and 1:2 (AB_2) complexes as a function of the equivalents of B added during the titration of A with B. Equilibrium calculations were performed using $K_{AB} = 1 \times 10^3 \text{ M}^{-1}$ and $K_{AB_2} = 1 \times 10^3 \text{ M}^{-1}$ (top), and $K_{AB} = 1 \times 10^3 \text{ M}^{-1}$ and $K_{AB_2} = 1 \times 10^4$ (bottom). The initial concentration of A is set to 1.0 mM



of concentration C_{base} up to a specified maximum added volume $V_{base,max}$. At each titration point, the total moles of acid and base are updated from the analytical conditions, the total volume is recomputed, and the corresponding effective total concentrations of the acid ($C_{acid,eff}$) and of the base cation ($C_{base,eff}$) in the mixture are obtained. These quantities provide the input data for the equilibrium calculation at that titration step. The autoprotolysis constant can also be adjusted, which is particularly useful when treating mixed solvents (e.g., DMSO/ H_2O) [18, 20], where the effective ionic product of the medium differs from that of pure water. This approach can be useful in addressing issues in supramolecular chemistry, where the species are often poorly soluble in water, making it necessary to use an organic co-solvent [18, 20–22].





In addition to the numerical data, the module automatically generates plots that summarize the titration behaviour. A first plot reports the simulated titration curve, i.e. pH as a function of the equivalents of base added, highlighting the characteristic buffer regions and inflection points associated with successive deprotonation steps of the polyprotic acid. A second plot presents the α -fractions of the species H_nA , H_{n-1}A^- , ..., A^{n-} as a function of pH, in the spirit of a Bjerrum diagram, thereby visualizing how the dominant protonation state evolves over the titration range. Together, these graphical outputs provide an intuitive link between the underlying equilibrium constants and the speciation patterns observed during titration.

The formulation of the titration module is generic in the number of acidic protons and can be used, without modification of the numerical core, for any monoprotic or polyprotic acid by supplying the appropriate vector of dissociation constants and the value of autoprotolysis K_w . All the details employed in the algorithm are illustrated in the Methos Section. In Fig. 2 are reported the plots for a diprotic and triprotic acid.

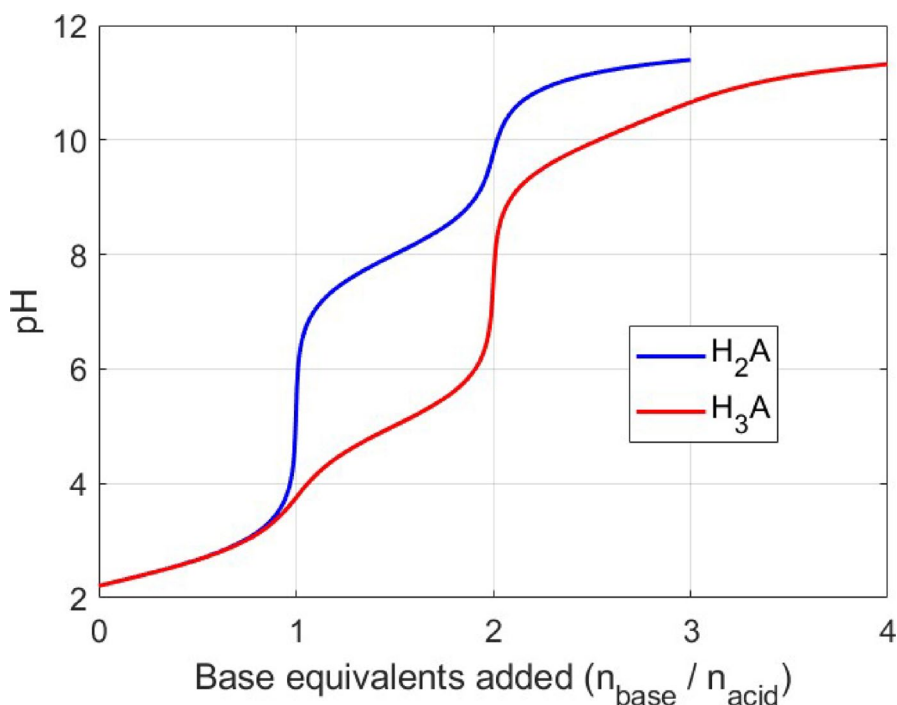


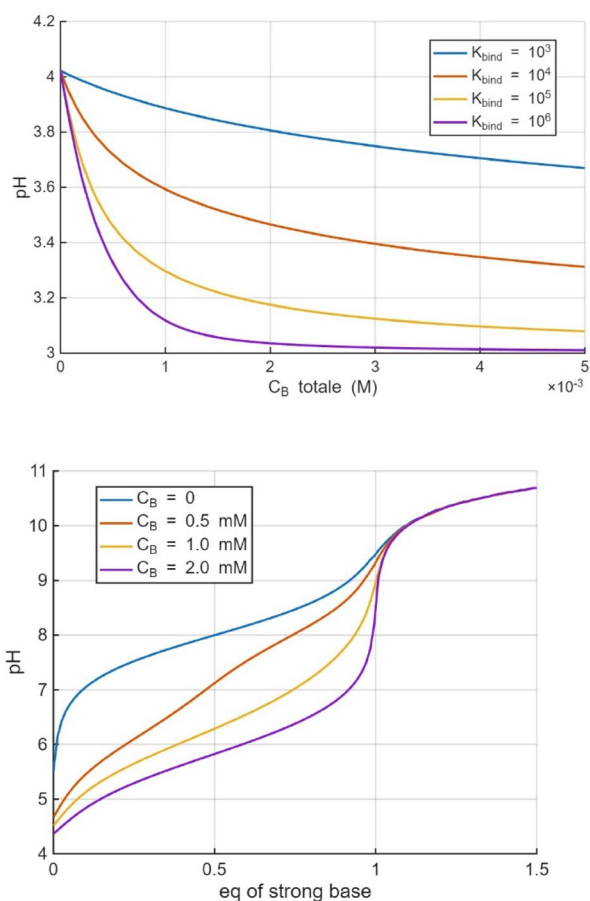
Fig. 2 Titration curve showing pH as a function of base equivalents added for generic diprotic and triprotic acid H_2A and H_3A (10.0 mM) titrated with a 0.10 M strong base solution. Equilibrium speciation was calculated using, $C_B = 10$ mM, initial volume 50 mL $\text{p}K_{a1} = 2.0$ and $\text{p}K_{a2} = 8.0$ for H_2A and $\text{p}K_{a1} = 2.0$, $\text{p}K_{a2} = 5.0$ and $\text{p}K_{a3} = 10.0$ for H_3A

A phenomenon that can be easily encountered in the study of equilibria is an acidic species that, in its deprotonated form can complex a guest. The equilibria are depicted in Eqs. (18) and (19). The framework can describe and quantify how the complex formation perturbs the acid–base equilibrium and the resulting solution pH.



Figure 3 illustrates the impact of complex formation with B on the acid–base behavior of HA from two complementary perspectives. First, the plots show how increasing the total concentration of B progressively enhances the apparent acidity of HA, shifting the equilibrium of the solution toward lower pH values as the conjugate base of the acid is sequestered in the AB complex. Second, the figure reports titrations of HA with a strong base performed at increasing B concentrations for several values of the binding constant K_{bind} , highlighting that stronger binding systematically amplifies this effect and produces larger downward shifts of the entire pH profile.

Fig. 3 (top) Calculated pH of the solution versus the concentration of the species B added. The different curves were calculated for different binding constants in the range $K_{\text{AB}} = 10^3$ – 10^6 . Equilibrium calculations were performed using the following values: $\text{p}K_{\text{a}} = 5.0$, total concentration of HA 1.0 mM. (bottom) Titration curves of the weak acid HA (1.0 mM) showing pH as a function of equivalents of strong base in the presence of increasing amount of species B 0.0–2.0 mM. The equilibria taken in consideration are those reported in the main text (eqs. 18 and 19). Exact concentrations of species B were indicated in the legend. Equilibrium calculations were performed using the following values: $\text{p}K_{\text{a}} = 8.0$; concentration of strong base: 10.0 mM



2.4 Real case applications

In the second part of this work, selected case studies of equilibria in solution involving systems reported in the literature within the field of supramolecular chemistry will be taken in consideration. To these systems, the previously presented methodological tools will be applied, further enhanced by an additional algorithm capable of optimizing the equilibrium constant values on the basis of experimental data, as this represents a widely encountered requirement in the investigation of such systems.

Calixarenes decorated with guanidinium units at the upper rim have proved to be effective compounds in the cleavage of phosphodiester [18, 21–25]. Their catalytic activity was found to be strongly dependent on pH. For this reason, studying their acid–base behavior is a fundamental requirement for understanding their catalytic mechanism.

The titration curves of a series of guanidinocalixarenes bearing different number of substituents are available [26]. In Fig. 4 are reported the experimental titration data of the chlorohydrate derivatives of triguanidinocalixarene **1** and tetraguanidinocalixarene **2**. Owing to the close values of the acidity constants of the guanidinium units, the titration curve shows an almost continuous rise in pH. The elaborated script (see Section S6 in Supporting Information) implements a general nonlinear least-squares framework for extracting equilibrium constants from titration data by fitting a mechanistic speciation model to the measured pH profile.

In its present form it targets a triprotic acid and a tetraprotic acid, but the structure of the script is deliberately modular: the only system-specific components are the function that computes the model titration curve from a set of pK_a values and the residual function that compares this curve with the experimental data. Parameter optimization is carried out using the *lsqnonlin* algorithm implemented in Matlab; this routine iteratively refines the parameter vector to minimize the sum of squared residuals between observed and calculated pH values across the full range of titrant volumes. The same code also provides standard uncertainties and confidence intervals for all fitted parameters.

It should be noted that the presence of multiple guanidinium units on the calixarene scaffold leads to an increase in acidity compared to a single guanidinium unit, most likely due to electrostatic repulsion among the positively charged groups. [25, 26]

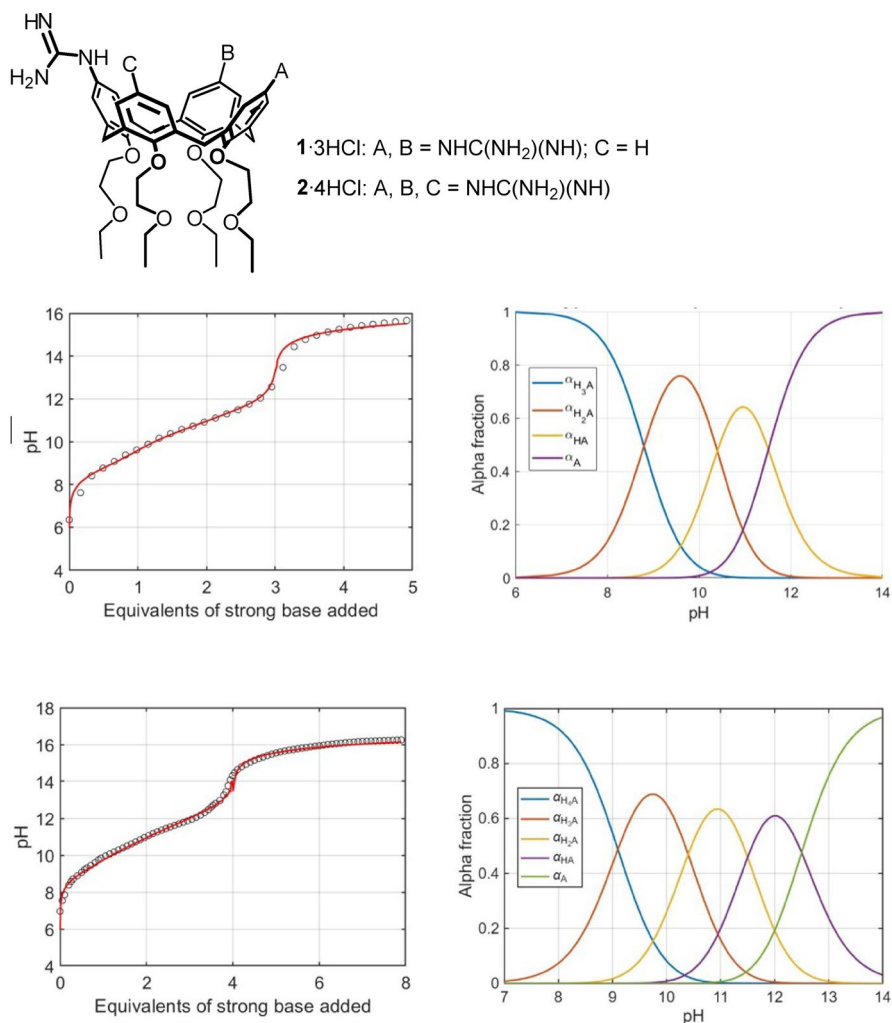
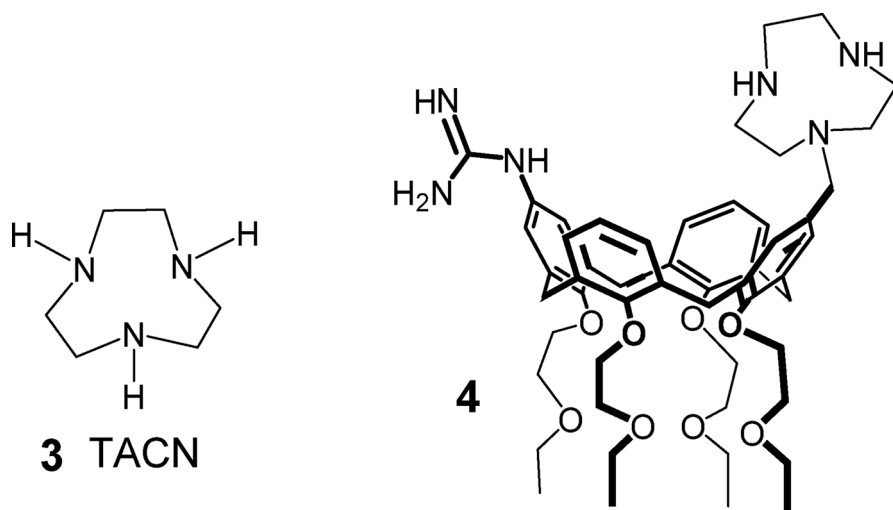


Fig. 4 Titration of the trischlorohydrate of **1** (top) and tetrachlorohydrate of **2** (bottom) with a strong base under the conditions described in the main text. The points are the experimental data and the line is the calculated curve determined with a least-square procedure. Molar fractions of the species are reported in the plots in the right. The legend reports the correspondence between each color and the respective species. Conditions: **1**/2·3HCl (1.0 mM) in 80% DMSO, 25 °C, Me_4NOH as titrating agent. The optimized pKa values are 8.8, 10.4 11.5 and 9.1, 10.5, 11.7 and 12.6 for the triguanidino and tetra-guanidinocalixarene respectively, the uncertainty is ± 0.4 at 95% of confidence level. The experimental data are reported in Section S8, Supporting Information



Because both the residual function and the parameterization are easily redefined, this framework, as the other illustrated above, can be straightforwardly adapted to a wide variety of titration-based studies—for example, acids with different proton stoichiometries, metal–ligand complexation schemes, or mixed acid–base systems—while retaining a unified strategy for parameter estimation and uncertainty quantification. On this regard, the cyclic polyamines are compounds that find applications in a large variety of field including supramolecular chemistry. Among the most common of these compounds there is the triazacyclononane (TACN) very often used in supramolecular catalysis and enzyme mimics, capable to complex a variety of metal ions, typically Cu^{II} and Zn^{II} . [18, 19] This compound has been widely used to synthesize a variety mono- and polytopic receptors for metals with applications in biomimicry, radiopharmaceutical application and in the design of supramolecular catalyst [18]. The experimental titration curve of the triazacyclononane as trischlorohydrate ($\text{TACN} \cdot 3\text{HCl}$) [19, 27] is shown in Fig. 5 (red line). The plot reveals three titratable protons at different pK_a values. The pK_a values determined by the least square fitting of the experimental point are 2.9, 6.3 and 10.9. For all the titrations presented here, the program yields similar uncertainties, approximately ± 0.4 at a 95% confidence level. Interestingly, when the titration is performed in the presence of CuCl_2 in an equimolar amount with $\text{TACN} \cdot 3\text{HCl}$, the resulting curve changes dramatically in appearance (Fig. 5, purple line). Upon addition of the first three equivalents of base, there is no significant increase in pH, suggesting an apparent behavior characteristic of a strong acid. This experimental evidence indicates the presence of strong binding between TACN and the Cu^{II} ion. The complexation can occur only when the macrocycle is in its deprotonated form, as the nitrogen atoms must have available lone pairs to interact with Cu^{II} . Consequently, the addition of the metal ion significantly shifts the chemical equilibrium toward the fully deprotonated species, which is the form capable to complex the metal. Furthermore, a fourth acidic proton is detected, exhibiting a pK_a value in the range of 8–9, which was not observed in the titration performed in the absence of the metal. This observation is consistent with the depro-

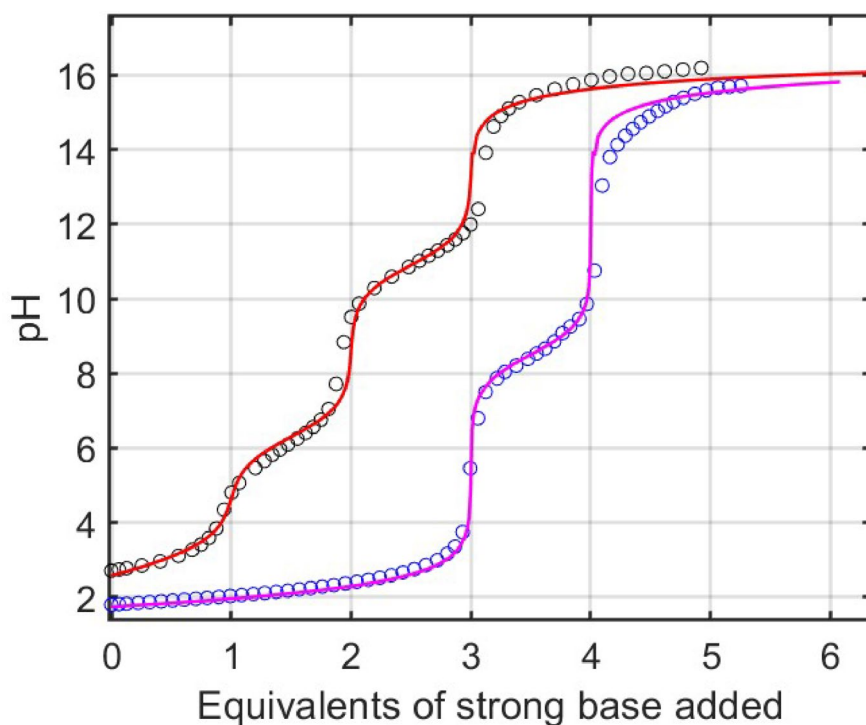


Fig. 5 Titration of TACN·3HCl (3H_3^+) with a strong base in the absence (red line) and in the presence (purple line) of Cu^{II} . The points are the experimental data and the line the calculated curve determined with a least-square procedure described in the text. Conditions: TACN·3HCl (1.0 mM) in 80% DMSO, 25 °C, $\text{p}K_{\text{w}} = 18.4$, Me_4NOH as titrating agent. The optimized $\text{p}K_{\text{a}}$ values are 2.9, 6.3 and 10.9. For the coordinated water molecule $\text{p}K_{\text{a}} = 8.5$. Uncertainties ± 0.4 at a 95% confidence level. The experimental data are reported in Section S8.3–S8.4, Supporting Information. A trustable value of binding with CuCl_2 cannot be determined because it is too high ($K_{\text{bind}} > 10^{10}$)

tonation of a water molecule coordinated to the metal cation. The fitting with the program presented affords the values reported in the figure caption.

With an additional increase in structural complexity, the chlorohydrate derivative of calixarene **4** [19] exhibits four titratable protons in its fully protonated state (Fig. 6, red trace), owing to the presence of a guanidinium unit bearing a proton that is titratable within an accessible pH range.

Also in this case the titration carried out in the presence of an equimolar amount of CuCl_2 features a curve that is constantly at lower pH whenever compared to that observed in the absence of any metal (Fig. 6, purple line). In this case the binding of Cu^{II} to the ligand is not that high as in the case of the bear TACN. The value determined with the optimization procedure is $\text{p}K_{\text{Cu}} = 7.6$. The other data are reported in the caption of Fig. 6.

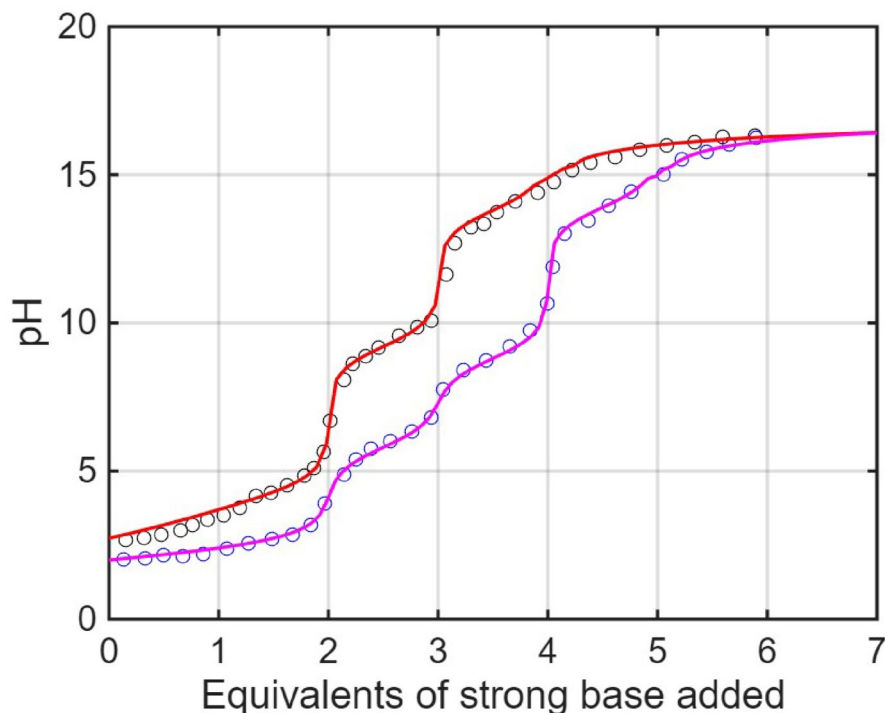


Fig. 6 Titration of $4\cdot 4\text{HCl}$ with Me_4NOH in the absence (red line) and in the presence (purple line) of Cu^{II} . The points are the experimental data and the line the calculated curve determined with a least-square procedure. Conditions: $4\cdot 4\text{HCl}$ (1.0 mM) in 80% DMSO, 25 °C, $\text{p}K_{\text{w}}=18.4$, Me_4NOH as titrating agent. The optimized $\text{p}K_{\text{a}}$ values are 3.1, 4.2, 9.2 and 12.5 for the $\text{p}K_{\text{a}}$ values determined in the absence of the metal. The $\text{p}K_{\text{a}}$ of the deprotonated water molecule is 8.8. The binding with Cu is $\text{p}K_{\text{Cu}}=7.6$. Uncertainties ± 0.4 at a 95% confidence level. The experimental data are reported in Section S8.5–S8.6, Supporting Information

3 Conclusions

In summary, this work presents a versatile, open-source, and fully customizable computational tool designed for the quantitative treatment of equilibrium data in systems of varying complexity. By integrating robust mathematical formulations with flexible MATLAB-based scripts, this study provides researchers with a transparent alternative to conventional commercial or online solutions, which often obscure the computational process and restrict adaptability. The proposed framework offers a rigorous yet accessible means of performing quantitative analysis, facilitating the verification, modification, and extension of the computational workflow according to specific experimental requirements, while requiring only minimal programming skills and basic familiarity with MATLAB. The application of the tool to representative datasets available from our own research activities, demonstrates its reliability, accuracy, and broad applicability to recurring issues in supramolecular and physical organic chemistry. Moreover, its utility extends well beyond these disciplines, as it can be effectively employed for the quantitative analysis of almost any equilibrium

process occurring in solution. Beyond its immediate utility, this work underscores the importance of open, reproducible computational practices in chemical data analysis and interpretation. The availability of the complete scripts and detailed methodological documentation ensures that the approach can be readily employed, tested, and improved by the scientific community.

Future developments will focus on expanding the capabilities of the framework to address increasingly complex experimental scenarios. In particular, the next stage will involve the integration of kinetic data fitting functionalities, reflecting interest of many research groups in the quantitative analysis of reaction mechanism. In addition, the modular structure of the tool readily allows for further extensions, including the incorporation of routines for the treatment of NMR and calorimetric data. These developments will enhance the versatility of the platform and broaden its applicability as a comprehensive tool for the analysis of experimental data in chemical research.

4 Method section

All the scripts were written in Matlab code. Their effectiveness was tested either in Matlab R2024b (24.2.0.2712019) and in Matlab 2025a (25.1.0.2943329).

All the codes developed in this work follow a modular approach, meaning that separate scripts operate together with a main script that calls them as needed. All the files are provided in the Supporting Information (see Sects. 1–7) to enable readers to use them and, when needed, to slightly modify or adapt them to their own needs. This modular approach enhances the understanding of the code and facilitates its adaptation to different cases. The following section provides instructions for modifying the files to correctly define the system of interest.

Definition of the system in the scripts The calculation script first specifies the equilibrium constants in logarithmic form, the total concentrations of all components, and the overall charge of the system. This is followed by a core block that defines the model. For the example including the AB and AB₂ complexes, the input file is structured as follows:

Model.species = {'A', 'B+', 'AB+', 'AB2++'} Definition of all the existing species in solution. Model.components = {'A', 'B'}: here are listed the primary components. Each species consists in 0-n number of each component. N comp = [1 0 1 1; 0 1 1 2]: these matrix contains the exact number of components existing in each species. charges = [0, 1, 1, 2]: the charge of each species. Nu = [-1 -1 1 0; 0 -1 -1 1]. In this matrix are reported the stoichiometries of the reactions. In the present example there are the formation of AB and AB₂: A + B $\rightleftharpoons$ AB+ for the first row and AB + B $\rightleftharpoons$ AB₂ for the second row. Model.logK = [logK1; logK2]; definition of the equilibrium constants.

4.1 Selection and validation of the solution of the equilibrium algorithm

For each candidate solution, the program evaluates the quality of mass and charge conservation. From $C^{(k)}$ it computes

$$\mathbf{b}_{calc}^{(k)} = N \mathbf{C}^{(k)},$$

and compares it to $\mathbf{b}_{tot}$ via relative errors for each conserved quantity, for example

$$\varepsilon_A^{(k)} = \frac{|b_{A,calc}^{(k)} - b_{A,tot}|}{b_{A,tot} + \delta},$$

$$\varepsilon_B^{(k)} = \frac{|b_{B,calc}^{(k)} - b_{B,tot}|}{b_{B,tot} + \delta},$$

$$\varepsilon_{charge}^{(k)} = \frac{|b_{charge,calc}^{(k)} - b_{charge,tot}|}{|b_{charge,tot}| + \delta},$$

where δ is a small regularization constant to avoid division by zero.

A global balance error is then defined as

$$\varepsilon_{bal}^{(k)} = \max \left(\varepsilon_A^{(k)}, \varepsilon_B^{(k)}, \varepsilon_{charge}^{(k)} \right).$$

The final accepted solution is chosen by:

1. Minimizing $\varepsilon_{bal}^{(k)}$ across all candidates.
2. For cases with similar balance error, minimizing the residual norm $\|\mathbf{F}(\mathbf{x}^{(k)})\|$.

A candidate is flagged as successful only if both the criteria are satisfied:

- the nonlinear solver reports convergence
- $\varepsilon_{bal}^{(k)}$ is below a prescribed tolerance (e.g. 10^{-6}).

This two-level criterion ensures that the selected equilibrium state is both a good numerical root and a physically consistent speciation in terms of mass and charge balances.

Titration algorithm–data structures, diagnostics, and conservation checks The script preallocates arrays for the total concentrations, equilibrium species, residual norms, exit flags, and success indicators over all titration points. This design supports efficient numerical loops and ensures that all relevant quantities are stored in a structured, point-by-point manner. After each successful equilibrium calculation, the script reconstructs the total concentrations from the reported species (by summing free and complexed forms with appropriate stoichiometric coefficients) and stores these “balance” values as a diagnostic. This reconstruction constitutes a general check of mass conservation and allows users to verify that the numerical solver respects the underlying physical constraints within a specified tolerance.

These quantities are finally assembled into a table, which serves as a compact and extensible container suitable for downstream analysis, export, or statistical treatment.

The table structure is generic: beyond the specific variable names used here, it represents the mapping between titration coordinate, analytical totals, equilibrium results, and solver diagnostics that would be required in any equilibrium-mixing simulation.

Adic-base titrations This module implements a general numerical framework for the titration of a polyprotic acid consisting in the following scripts: `alfa_polyprotic`, `charge_balance_polyprotic`, `titration_polyprotic_acid` and `main_script`. The code in reported in the Supplementary Information in Section S3.

The equilibrium problem is formulated in terms of the charge balance for the aqueous solution containing the polyprotic acid, its conjugate bases, the strong base, and water. For a trial value of the proton concentration $[H^+]$, the program first computes the hydroxide concentration $[OH^-] = K_w/[H^+]$ and then evaluates the distribution of the acid among its $n + 1$ protonation states H_nA , $H_{n-1}A^-$, ..., A^{n-} using the standard α -fraction formulas for polyprotic systems. These α -fractions are used to construct the average charge (Bjerrum formation function) associated with the acid, which in turn enters the electroneutrality equation balancing positive charges ($[H^+]$ and the base cation) against negative charges (hydroxide and the charged acid species).

For each titration point, the proton concentration is determined by solving the nonlinear charge-balance equation $F([H^+]) = 0$ with respect to $[H^+]$. This is accomplished by a robust bracketing and root-finding procedure based on MATLAB's `fzero`, which uses the solution at the previous titration step as an initial guide and adaptively adjusts the search interval in $[H^+]$ to bracket a sign change of the charge-balance residual. Once the root is found, the corresponding pH is computed as $\text{pH} = -\log_{10}[H^+]$, and the full set of α -fractions and species concentrations is reconstructed and stored. The resulting titration dataset is assembled into a table containing, for each volume of base added, the total volume, the equivalents of base per mole of acid, the pH, the proton and hydroxide concentrations, the α -fractions of all protonation states, and their molar concentrations.

Least-square optimization The optimization is carried out with `lsqnonlin`, which iteratively adjusts the parameter vector to minimize the sum of squared residuals between observed and calculated pH values over the full range of titrant volumes. By requesting the Jacobian matrix at the solution and using it to construct a covariance matrix, the same code also provides standard uncertainties and confidence intervals for all fitted parameters. As input, the code requires initial values for the parameters to be optimized. See Section S6 in Supporting Information. The experimental data used for the fitting are reported in Section S8. Claude Sonnet 4.6, Nemotron 3 and MATLAB Copilot models were employed to support the verification and debug of selected code components.

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s10910-026-01818-0>.

Author contributions R.S. conceived and designed the study, performed all calculations, analyzed the data, and wrote the manuscript. R.S. approved the final version of the manuscript.

Funding Open access funding provided by Università degli Studi di Roma Tor Vergata within the CRUI-CARE Agreement. This study was supported by University of Tor Vergata through Progetti di Ricerca Dipartimentale 2024 (grantid: GuglielmottiV25Rsa2024).

Data availability No datasets were generated or analysed during the current study.

Declarations

Competing interests The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. E.V. Anslyn, D.A. Dougherty, *Modern Physical Organic Chemistry* (University Science Books, Sausalito California, 2006)
2. C.A. Schalley, *Analytical Methods in Supramolecular Chemistry* (Wiley-VCH, 2007)
3. S. Kubik, *Supramolecular Chemistry: From Concepts to Applications* (De Gruyter, 2024)
4. P. Thordarson, Determining association constants from titration experiments in supramolecular chemistry. *Chem. Soc. Rev.* **40**, 1305–1323 (2011)
5. H.-J. Schneider, *Applications of Supramolecular Chemistry* (CRC Press, 2012)
6. P. Gans, A. Sabatini, A. Vacca. HYPERQUAD 2008, Hyperquad Suite of Programs for Determination of Equilibrium Constants from Potentiometric and Spectrophotometric Data; Protonic Software, Leeds, UK, 2008.
7. L. Alderighi, P. Gans, A. Ienco, D. Peters, A. Sabatini, A. Vacca, Hyperquad simulation and speciation (HySS): a utility program for the investigation of equilibria involving soluble and partially soluble species. *Coord. Chem. Rev.* **184**, 311–318 (1999)
8. MINEQL+: Chemical Equilibrium Modeling System, Version X.X for Windows; Environmental Research Software: Hallowell, ME, USA.
9. X. Xie, D.E. Giammar, Z. Wang, MINFIT: a spreadsheet-based tool for parameter estimation in an equilibrium speciation software program. *Environ. Sci. Technol.* **50**, 11112–11120 (2016)
10. <https://supramolecular.org> Online tools for supramolecular chemistry research and analysis.
11. F. Morel, J. Morgan, Numerical method for computing equilibria in aqueous chemical systems. *Environ. Sci. Technol.* **6**, 58–67 (1972)
12. I. Cukrowski, G. Ngigi, Solution equilibria. A unified mathematical treatment of potentiometric and polarographic data in a metal-ligand equilibrium study at a fixed ligand to metal ratio and various pH values. *Electroanalysis* **13**, 1242–1252 (2001)
13. Y. Liu, Y. Liu, M.G.B. Drew, A mathematical approach to chemical equilibrium theory for gaseous systems—I: theory. *J. Math. Chem.* **51**, 715–740 (2013)
14. L. Castellino, E. Alladio, S. Bertinetti, G. Lando, C. De Stefano, S. Blasco, E. García-España, S. Gama, S. Berto, D. Milea, PyES—an open-source software for the computation of solution and precipitation equilibria. *Chemom. Intell. Lab. Syst.* (2023). <https://doi.org/10.1016/j.chemolab.2023.104860>
15. A.E. Martell, R.J. Motekaitis, *Determination and Use of Stability Constants*, 2nd Edition. (Wiley VCH, 1996)

16. D.O. Soloviev, C.A. Hunter, Musketeer: a software tool for the analysis of titration data. *Chem. Sci.* **15**, 15299–15310 (2024)
17. J. Nocedal, S.J. Wright, *Numerical Optimization* (Springer Nature, 2006)
18. A. Casnati, R. Salvio, An equal terms comparison of the proficiency of artificial phosphodiesterases by using simple models of RNA or DNA as benchmarks—the takeaway to design next generation supramolecular catalysts. *Coord. Chem. Rev.* **531**, 216479 (2025)
19. R. Salvio, S. Volpi, R. Cacciapaglia, A. Casnati, L. Mandolini, F. Sansone, Ribonuclease activity of an artificial catalyst that combines a ligated Cu(II) ion and a guanidinium group at the upper rim of a cone-calix[4]arene platform. *J. Org. Chem.* **80**, 5887–5893 (2015)
20. R. Salvio, M. Moliterno, D. Caramelli, L. Pisciotanni, A. Antenucci, M. D’Amico, M. Bella, Kinetic resolution of phosphoric diester by Cinchona alkaloid derivatives provided with a guanidinium unit. *Catal. Sci. Technol.* **6**, 2280–2288 (2016)
21. C.A. Vezzoni, A. Casnati, S. Orlanducci, F. Sansone, R. Salvio, Enzyme mimics based on guanidino-calix[4]arene/nanodiamond hybrid systems with phosphodiesterase activity. *ChemCatChem* (2024). <https://doi.org/10.1002/cctc.202301477>
22. D. Lisi, C.A. Vezzoni, A. Casnati, F. Sansone, R. Salvio, Intra- and intermolecular cooperativity in the catalytic activity of phosphodiester cleavage by self-assembled systems based on guanidinylated calix[4]arenes. *Chem. Eur. J.* **29**, e202203213 (2023)
23. G. Pucciarelli, F. Ranieri, A. Casnati, S. Di Stefano, S. Volpi, R. Salvio, Temporally controlled supramolecular catalysts with pH-dependent activity. *ACS Omega* **11**, 6353–6361 (2026)
24. F.H. Zelder, R. Salvio, J. Rebek, A synthetic receptor for phosphocholine esters. *Chem. Commun.* **12**, 1280–1282 (2006)
25. R. Salvio, S. Volpi, R. Cacciapaglia, F. Sansone, L. Mandolini, A. Casnati, Phosphoryl transfer processes promoted by a trifunctional calix[4]arene inspired by DNA topoisomerase I. *J. Org. Chem.* **81**, 9012–9019 (2016)
26. L. Baldini, R. Cacciapaglia, A. Casnati, L. Mandolini, R. Salvio, F. Sansone, R. Ungaro, Upper rim guanidinocalix[4]arenes as artificial phosphodiesterases. *J. Org. Chem.* **77**, 3381–3389 (2012)
27. R. Salvio, R. Cacciapaglia, L. Mandolini, General base-guanidinium cooperation in bifunctional artificial phosphodiesterases. *J. Org. Chem.* **76**, 5438–5443 (2011)

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Riccardo Salvio^{1,2}

✉ Riccardo Salvio
riccardo.salvio@uniroma2.it

¹ Dipartimento di Scienze e Tecnologie Chimiche, Università degli Studi di Roma “Tor Vergata”, Via Della Ricerca Scientifica, 1, 00133 Roma, Italy

² ISB – CNR Sezione Meccanismi di Reazione, Università La Sapienza, 00185 Roma, Italy