Design of an open source autotitrator using Python

David J Hodkin

Department of Earth Sciences, University of Oxford

Abstract

Autotitrators are common pieces of laboratory equipment that are used for a variety of functions, including determining alkalinity and maintaining a reacting solution at a fixed pH. This paper details the construction of an auto-titrator from a pH meter and a syringe pump, with a command script written in python the total cost of this method is approximately £3,000 depending on the exact models of pump and meter purchased.

Introduction

Autotitrators have wide applications across the sciences. By creating an autotitrator from open source software it allows not only for a cheaper unit, but also for enhanced transparency of the underlying analysis. This paper serves as a starting point for developing a versatile custom autotitrator system.

Design

The components used in this apparatus are detailed in table 1. The pH meter and syringe pump are connected to a pc using its USB ports. A function called “Collect” is used to first call on the syringe pump to return the volume of titrant displaced, and secondly call on the pH meter to return the values for pH and temperature. These values are then compiled into a data frame which is printed to the console and saved to a csv file.

Autotitrator.py is used for calculating the alkalinity of a sample. While the pH is greater than 3 the program will run the Collect function described above. If the pH is above 4.5 the rate will be set to 5 until the pH is below 4.5. This allows the program to quickly proceed through the initial part of the analysis and then slow during the portion that requires greater accuracy. When the pH drops below 3 the titration will stop.

pH\_stat.py is used for running pH stat experiments. With the associated analysis file Rate\_Calc.rmd it is used for determining carbonate precipitation rate information. The program collects the initial pH and assigns this to a variable called setpoint\_pH. The difference between this and the pH is calculated and if this is greater than 0.2 the pump rate is adjusted to bring this back into bounds. The rate is kept within limits set by the function set\_lims. If the pump rate drifts below 0.2 the pump is paused and restarted when the pH drifts out of bounds again.

|  |  |
| --- | --- |
| Item | Approximate cost (2019) |
| Orion Dual Star pH meter | £1,800 |
| * Ross 8102BNUWP Refillable pH Electrode |
| * 927007MD Stainless Steel ATC Probe |
| Chemyx Fusion 100 Syringe pump | £1200 |
| Tubing and connectors: |  |
| * P-660 Luer Female to M6 Female | £13 |
| * P-247 Flangeless Nut | £3 |
| * P-200NX Flangeless Ferrule | £11 |
| * Thermo Scientific™ PEEK Capillary Tubing (0.020 in. ID; 30cm length; Orange) | £13 |
| Total | £3,040 |

Operation

The programs takes inputs before the start of the run which are attached to the data output and used in subsequent data analysis.

The pH\_Stat.py program will ask for a Key which is a sample name/number which is unique for each experiment.

The Autotitrator.py program will ask for a Key, this is a sample name/number and should match the sample name/number for the associated pH\_Stat file. NaHCO3 is the mass of NaHCO3 used in the experimental solution (g). Na2CO3 and CaCl2 are the masses of the respective reagents in the titrant solutions (g/50mL). CaCO3 is the mass of carbonate seed crystals used (g). Solution mass is the mass of solution used (g). Experimental Designation allows for grouping variables to be assigned to each experiment.

Sample mass (g) is the mass of the sample in grams (subsequent data analysis assumes a density of 1 kg/L). Acid normality is the normality of acid used for the analysis (it is important to calibrate the acid and use the true normality).

Data analysis

An r script for calculating alkalinity from experimental data is provided (Titr\_Calc.rmd). This script loads all data in the target folder and then calculates the first gran function (F1) based on the formula (Eq. 1) where V0 is the initial sample volume, Vt is the titrant volume.

|  |  |
| --- | --- |
|  | Eq. 1 |

This is plotted against the volume of titrant added and the intercept between the sloped portion of the plot and the x-axis is used to calculate sample alkalinity using (Eq. 2) where B is the volume of acid to the bicarbonate equivalence point (intercept from Fig. 1) N is acid normality and m is mass of sample.

|  |  |
| --- | --- |
|  | Eq. 1 |

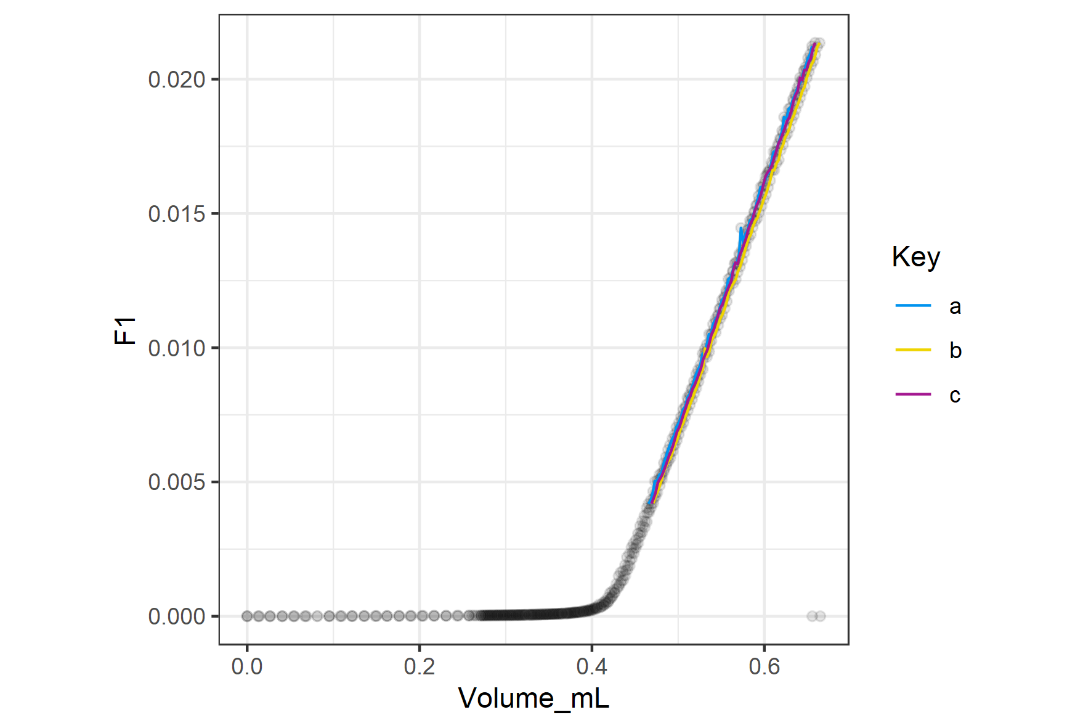


Figure 1: Gran plot of sample data

The precipitation rate is calculated from the pH stat data using the Rate\_Calc.rmd script.

Conclusions.

A program is presented for running alkalinity autotitrations and analyzing the subsequent data.

Acknowledgment

The author acknowledges Chemyx for the example computer control program for python, which was used as an initial basis for the communicating with the Chemyx device.

References