Control the uncontrollable – An anachronistic approach for contemporary automation

Hanno Dierke
Physikalisch-Technische Bundesanstalt
mailto:hanno.dierke@ptb.de

Nowadays, many measurement instruments can be operated remotely using the remote desktop functionality provided by contemporary computer operating systems. However, some well-established proprietary measurement systems use "historical" hardware that does not provide such functionality. Automation freeware tools can be used to provide rudimentary remote control of such systems.

1 Introduction

The Physikalisch-Technische Bundesanstalt (PTB) uses a measuring setup employing a Fizeau interferometer to determine the radius of curvature of spherical sections. The sample under test is placed at two characteristic positions in the convergent beam of the interferometer. First, with the apex of the sample in the focus of the incoming beam and the light is reflected at one point ("cat's eye position"). Second, the test specimen is in a position where the incident light hits the spherical surface vertically and is reflected back into itself ("confocal position", see Figure 1). In these two positions an interference pattern with a minimal number of fringes (ideally none) can be observed. The radius of curvature of the test object results as the distance between these two characteristic positions [1].

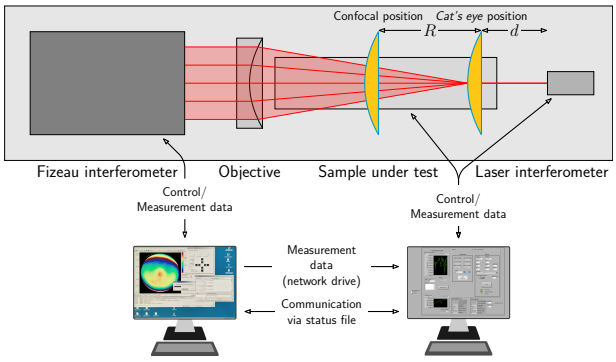

Fig. 1 *Schematic representation of the measurement of the radius of spherical sections at PTB.*

The measurement data for each test object is recorded by two independent systems: While one computer is only used to control the Fizeau interferometer and store its results, the superordinate control of the measurement and collection of all data is carried out by another computer. Since the latter system uses standard hardware an update of the operating system is possible with reasonable effortHowever, the Fizeau interferometer is controlled by proprietary software with defined hardware and software requirements. It is therefore tied to the existing - now obsolete - operating system and can only be updated with greater effort.

PTB has a dedicated network that also allows computers with an outdated operating system limited access to network resources. This allows at least an exchange of data between the participating computers and a rudimentary control of the measurement.

2 Revision and improved automation of the measurement

Over the past few months, the measuring system has undergone a revision. This revealed a number of shortcomings, such as manual and therefore error-prone documentation of the measurement parameters, a lack of options for interaction during an ongoing measurement and – above all – hardware-related difficulties in fully automating the measurement.

The measurement workflow typically was:

- The user creates a directory for measurement data on a network drive accessible from both computers, enters relevant parameters such as the axis parameters and number of measurement positions and starts the measurement on the control computer (LabVIEW). When the measurement is started, the control programme generates a status file via which the control programme and the measurement programme of the Fizeau interferometer communicate during the measurement. A path info file is also created in which the data path selected by the user is passed on to the Fizeau interferometer controller.
- The user then starts the measurement programme on the Fizeau interferometer's control computer. The parameters are included in the measurement programme and any changes require careful management of the versions of the programme to be used.

DGaO Proceedings 2026 – *http://www.dgao-proceedings.de* – ISSN: 1614-8436 – urn:nbn:de:0287-2026-P022-5
https://doi.org/10.83229/127-p022 – submitted: 21.07.2026 – published: 07.09.2026

- The measurement then runs without the possibility of user interaction. The measurement programme uses the path contained in the path info file to save the measurement data. These are collected by the control program and saved together with the measurement data of the linear axis, the laser intererometer and environmental data in a Matlab file for later evaluation.
- Once the measurement is complete, the user evaluates the saved measurement data (usually using Matlab on a different computer).

This process always required the user to be present at the measuring station to start a measurement, which resulted in a change in the environmental conditions. It was also not possible to monitor the progress of the measurement from outside - for example via a remote desktop connection.

As part of a revision of the measuring station in recent months, the control system was transferred to a Python-based measuring programme. In the course of this, an option was also integrated to control the interferometer remotely, at least in a rudimentary way, running two AutoIt scripts. AutoIt is a freeware BASIC-like scripting language designed for automating the Windows GUI and general scripting. It uses a combination of simulated keystrokes, mouse movement and window manipulation in order to automate tasks in a way not possible or reliable with other languages [2].

One AutoIt script runs continuously on the interferometer PC, monitoring a specific directory for a file with a particular extension. If such file is present, it's name is analysed and a further, appropriate AutoIt script is executed. This then carries out the actual measurement using the simulated user's keyboard and mouse inputs. The new workflow is therefore:

- The user creates a directory for measurement data on a network drive accessible from both computers, sets both the relevant axis parameters and the parameters for the interferometer programme and starts the measurement on the control computer (Python). When the measurement is started, the control programme generates a status file via which the control programme and the measurement programme of the Fizeau interferometer communicate during the measurement. It copies a measurement programme for the interferometer control, adapted with the selected parameters, to the memory path and a trigger file to the folder monitored by the AutoIt script.
- During the measurement, some measurement parameters can be influenced interactively by the user (e.g. waiting times or the number of repetitions of the current measurement). However, parameter changes that would lead to problems during subsequent evaluation (for example, the number of measurement positions around the confocal or cat's eye position or parameters for recording the interferometer data (averaging)) are prevented. The measurement data supplied by the Fizeau interferometer is collected by the control program for later evaluation and saved together with the measurement data of the linear axis, the laser interferometer and environmental data in a Matlab file.
- Once the measurement is complete, the saved measurement data files are evaluated by the user, currently still using Matlab on a different computer. In future, however, this can also be done by the Python measurement control programme after measurement data acquisition.

Thus, the new measurement procedure allows customised parameters for the interferometer programme. The storage of key parameters as a JSON file and in PTB's electronic laboratory book means that the individual measurements are much better documented. In addition, due to the automation of the measurement workflow the presence of a user on site is no longer required to start the measurement on this computer.

3 Conclusion and outlook

During the last revision of the PTB measurement set-up for the radius of curvature of spherical sections, the measurement procedure was fully automated with the help of the freeware AutoIt for the simulation of user inputs on non-remotely controllable measuring devices. In addition, reliable documentation of the measurement programme and the measurement parameters was made possible.

Disclaimer

Neither the author nor PTB, the employer, has any connection to or financial interest in the provider of the AutoIt software. There are other freeware tools, such as AutoHotkey, that offer similar features. AutoIt was chosen for this setup solely because of it's availability in PTB's local software pool.

References

[1] D. Sommer, M. Schulz, A. Wiegmann, and N. Triefenbach, "Erweiterte Anwendungen bei der Radiusmessung optischer Komponenten," in *DGaO-Proceedings, 118. Jahrestagung* (Deutsche Gesellschaft für angewandte Optik e.V., 2017). Poster P28.

[2] AutoIt homepage: `https://www.autoitscript.com/site/`, accessed 2026-06-01.