

Implementation of Level-3 Autonomous Patient-Specific Quality Assurance with Automated Human Interactive Devices

Jingqiao Zhang^{1,2}, Yizhou Zhao³, Jameson T. Baker^{1,2,3}, Yijian Cao^{1,2,3}, Jenghwa Chang^{1,2,3}

¹Northwell, New Hyde Park, NY, USA

²Department of Radiation Medicine, Zucker School of Medicine at Hofstra/Northwell, Lake Success, NY, USA

³Department of Physics and Astronomy, Hofstra University, Hempstead, NY, USA

Email: phyzjqk@hotmail.com, jchang24@northwell.edu

How to cite this paper: Zhang, J.Q., Zhao, Y.Z., Baker, J.T., Cao, Y.J. and Chang, J. (2023) Implementation of Level-3 Autonomous Patient-Specific Quality Assurance with Automated Human Interactive Devices. *International Journal of Medical Physics, Clinical Engineering and Radiation Oncology*, 12, 99-113.

<https://doi.org/10.4236/ijmpcero.2023.124009>

Received: October 14, 2023

Accepted: November 26, 2023

Published: November 29, 2023

Copyright © 2023 by author(s) and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

Purpose: Patient-specific quality assurance (PSQA) requires manual operation of different workstations, which is time-consuming and error-prone. Therefore, developing automated solutions to improve efficiency and accuracy is a priority. The purpose of this study was to develop a general software interface with scripting on a human interactive device (HID) for improving the efficiency and accuracy of manual quality assurance (QA) procedures. **Methods:** As an initial application, we aimed to automate our PSQA workflow that involves Varian Eclipse treatment planning system, Elekta MOSAIQ oncology information system and PTW Verisoft application. A general platform, the AutoFrame interface with two imbedded subsystems—the AutoFlow and the PyFlow, was developed with a scripting language for automating human operations of aforementioned systems. The interface included three functional modules: GUI module, UDF script interpreter and TCP/IP communication module. All workstations in the PSQA process were connected, and most manual operations were automated by AutoFrame sequentially or in parallel. **Results:** More than 20 PSQA tasks were performed both manually and using the developed AutoFrame interface. On average, 175 (± 12) manual operations of the PSQA procedure were eliminated and performed by the automated process. The time to complete a PSQA task was 8.23 (± 0.78) minutes for the automated workflow, in comparison to 13.91 (± 3.01) minutes needed for manual operations. **Conclusion:** We have developed the AutoFrame interface framework that successfully automated our PSQA procedure, and significantly reduced the time, human (control/clicking/typing) errors, and operators' stress. Future work will focus on improving the system's flexibility and stability and extending its operations to other QA procedures.

Keywords

Level-3 Automation, Patient-Specific Quality Assurance, Human Interactive Devices, Scripting

1. Introduction

Modern radiation oncology practices have been advanced by the latest imaging and computer technology, so the treatment planning and delivery process have become highly complex. The process of carrying out a treatment with a newly designed digital linear accelerator (LINAC) in a clinic involves multiple computer systems through various hardware and software interfaces. To achieve a successful treatment delivery, each step in a clinical workflow must be well defined, and the transition between steps should be carefully coordinated and error-free based on the risk analysis of the radiation therapy quality management [1].

Ideally, the clinical workflow should be achieved with one single technical provider or software/hardware vendor. However, no current vendor can provide such a solution as we know. Besides, there are concerns that vendors could fine-tune their devices and QA models to match potential flaws that could be identified by independent QA devices. Consequently, most clinics are using various computer workstations and multiple third-party application software packages to operate LINAC, drive devices and/or acquire patient/radiation data. Although the software and workflow with these systems are designed to ensure the accuracy and safety of the treatment delivery for both patients and therapy staff, dealing with communications between different software/hardware interfaces and navigating multiple control computer systems create potential risks for procedural mistakes [2]. In addition, the repetitiveness of the operation and the fatigue of operators put the quality and safety efforts in jeopardy. As a result, system integration and workflow automation are essential in a modern radiation oncology department that utilizes various clinical systems/software packages for daily operations.

One major obstacle to the automation/integration of multiple vendors' products is the strict FDA (Food and Drug Administration) safety and security policies on various clinical workstations that prevent the installation of automation/integration tools developed by users. To overcome this problem, we proposed a general automated workflow platform, the AutoFrame frame interface, for medical physics practice in our department. As the first application, the AutoFrame was implemented to automate the workflow of patient-specific quality assurance (PSQA) [2]-[7] through the interfaces available for external control of vendors' workstations, that is, those open ports for human interactive devices (HIDs) like mouse, keyboard, and other external devices [8]. As will be shown in this paper, this approach is particularly useful for our PSQA workflow that involves multiple software packages running on various vendors' workstations be-

cause automation through the HIDs does not violate the FDA safety and security policy of these clinical workstations and eliminates the needs for detailed inside knowledge of the commercial software.

The automobile industry has defined six levels of automation for the autonomous driving [9] based on the extent of human participation with a higher level of autonomous driving indicating less human involvement. **Table 1** lists these six levels of vehicle autonomy and the corresponding autonomy defined for the PSQA procedures. As shown in the second column of **Table 1**, the six levels of vehicle autonomy range from the lowest level (L0), which has no automation, to the highest level (L5) where the vehicle can drive completely by itself without human intervention. Each level of autonomy is determined by how much a driver's operation can be replaced by the hardware and software for automatic driving. Similar levels of automation can be defined for the PSQA workflow.

In this paper, we will first present our definitions of automation levels for the PSQA workflow that mimics the six levels of vehicle autonomy developed by the car industry. We will then show the design of our AutoFrame frame interface and its implementations for the L3 automation of our PSQA workflow. Test results of this platform on automating the PSQA workflow of our department will be demonstrated and improvements of our PSQA procedure from this automation tool will be examined. Finally, potential applications of the AutoFrame to other clinical procedures will be discussed.

2. Methods and Materials

2.1. Patient-Specific QA Program at Northwell Health

While there are no standards on how the PSQA should be performed, the PSQA program at the Radiation Medicine Department of Northwell Health is a typical one that verifies the dose delivery of every IMRT plan by collecting data using a PTW Octavius ion-chamber array [10]. The flowchart of our PSQA procedures is shown in **Figure 1** and explained in the following:

- 1) In a clinical treatment planning system (Varian Eclipse) workstation, create a PSQA verification plan, calculate the volumetric dose to the PSQA phantom and export the dose at the detector (a PTW Octavius unit) plane for gamma analysis.
- 2) Set up the PSQA phantom (an ion-chamber array imbedded inside a solid water phantom) on the linear accelerator (a Varian TrueBeam) and calibrate the PTW Octavius detector array [10] according to the calibration protocol.
- 3) In the R&V workstation (Elekta MOSAIQ), load the patient's PSQA verification plan, set the delivery to QA treatment mode in MOSAIQ, override the alerts which prevent the delivery and send the plan to the Varian TrueBeam LINAC.
- 4) In the Varian treatment workstation, accept the treatment plan from the R&V system, and remove the interlock for QA mode treatment delivery.
- 5) In the Varian control console, prepare and preview the treatment and deliver the beams of QA plan.

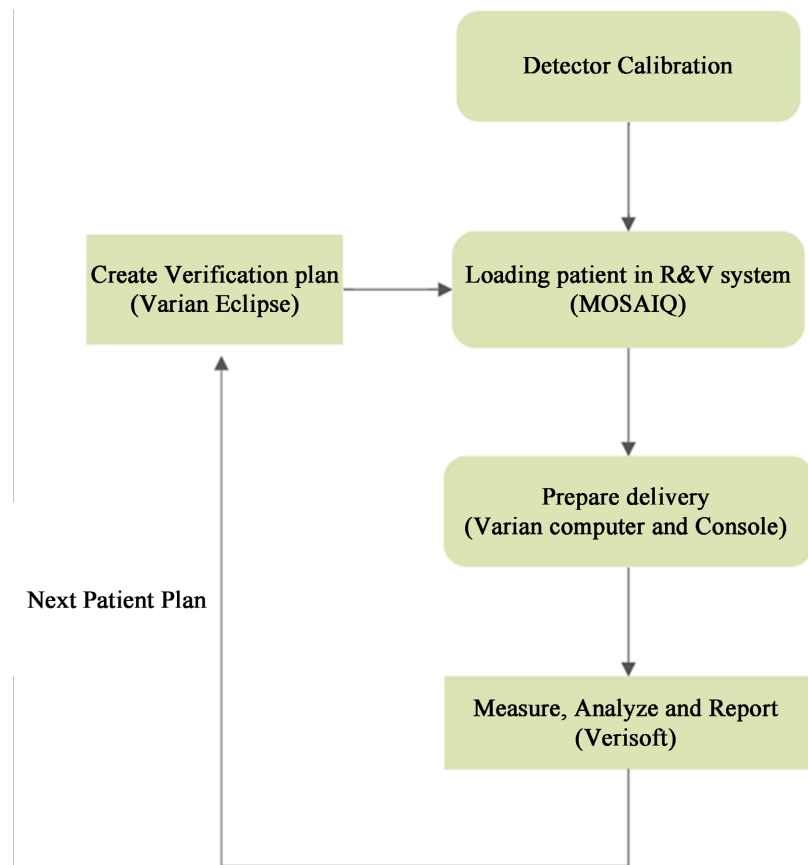


Figure 1. Flowchart of the patient-specific QA procedure at the Radiation Medicine Department of Northwell Health.

6) In the data processing workstation for the PTW Octavius, open the PTW Verisoft application, configure the ion-chamber array detector for collecting the measurement data once the LINAC is turned on.

7) Turn on the TrueBeam and deliver the QA plan.

8) Once the delivery is finished, perform the gamma index analysis [11] with a specified (e.g. 2 mm/3%) passing criteria on the PTW Octavius workstation, and export the PSQA report.

2.2. Definition of Automation Levels

As shown in the third column of **Table 1**, six levels of autonomy for our PSQA automation project were defined similar to the automation concepts adopted by the car industry. While these two industries may seem quite different, a common principle in both is that automation levels tend to increase as human involvement decreases. The last column of **Table 1** shows examples of clinical operations that can be achieved with each PSQA autonomy level. Note that unlike the vehicle automation, the levels of automation were defined according to the three classes of devices involved in the PSQA process: (A) departmental computers (e.g. working computers running treatment planning systems or R&V applications), (B) vendor computers (e.g. computers for the Varian LINAC), and (C) com-

puter-controlled devices (e.g. ionization detector array/phantom, Varian LINAC control console). The main difference between (A) departmental computers and (B) vendor computers is that we have more control over the operating system (OS) of the departmental computers than that of the vendor computers. For example, we can access most executable files (e.g. those in the SYSTEM 32 directory of a workstation running the MS Windows 10 OS) of the operating systems on the departmental computers but are almost totally blocked from accessing those files on the Varian computer controlling our LINAC due to strict FDA safety/security policies. Devices in the last category (computer-controlled devices) are most difficult to automate since they need to be set up manually (like the ion-chamber array and phantom) or require physical button push (like the Varian LINAC control console).

The current PSQA procedure shown in **Figure 1** already involves many workstations and associated hardware/software applications operated by the user, so it is characterized as the L1 automation based on the definitions in **Table 1**. To achieve a higher-level automation, the following time-consuming operations need to be automated: 1) communication between the operator and PSQA applications through the human interactive devices (HIDs) on various workstations for L2 and L3 automation, and 2) manual operation of the LINAC control console for L4 automation, and 3) in-room setup of the PSQA phantom and detector array for L5 or complete automation. Note that the treatment control console usually does not include HID ports so the L3 automation is the highest level of automation that can be achieved with the HID automation.

Table 1. Six levels of vehicle autonomy (the second column) and the corresponding autonomy defined for the PSQA procedures (the third column). The last column shows examples of clinical operations that can be achieved with each PSQA autonomy level.

Level	Vehicle autonomy	PSQA autonomy definition	PSQA autonomy examples
L0	No automation.	No computer aided operations.	Point dose measurements for verification of electron or 3D plans using an ionization chamber or diode.
L1	Driver assistance.	Human operations with computer assistance.	Computer aided measurements using PSQA devices with significant human involvements. This is the current practice.
L2	Partial driving automation.	Automated operations of the departmental computers that allow open access to OS.	Manual operations on departmental computers are automated, e.g. plan preparation and export in the treatment planning system, patient open/close on the R&V system and plan export to LINAC for QA delivery...
L3	Conditional automation.	Automated operations of vendor computers that don't allow open access to OS.	Manual operations on the VARIAN computer controlling the LINAC are automated, e.g. open/close patient plans, treatment parameters updates...
L4	High automation.	Automated operations of all devices except in-room detector/phantom setups.	Manual operations on the Varian control console (e.g. treatment preparation, beam on/off...) are automated.
L5	Fully automated driving.	Fully automated PSQA.	In-room detector/phantom setups are automated.

2.3. Automation Goal and Specifications of This Study

In the initial phase of this project, we focused on the automation of operations involving human interactive devices (HIDs), that is, all manual operations using the keyboard and mouse for computers outside the treatment room. Given that the control consoles of our LINACs do not have HID ports, the highest achievable automation would be L3 for our PSQA procedure in this phase. To achieve this goal, the automation should remove as much human intervention as possible, and synchronize all workstations involved in the PSQA process. In addition, the automation platform should require minimal coding effort but be able to break the barriers between operation system (OS) and software across the computers. Finally, the automation platform should have sufficient built-in flexibility, so that it can be extended to other clinical physics applications in the future, such as monthly or annual QA and automatic treatment planning.

2.4. AutoFrame Interface Framework

The *AutoFrame* interface framework was built based on the aforesaid automation goal and specifications and served as the automated workflow controller for this project. Coded with plain scripts, this framework provides standard and light interfaces for interpreting and executing user-defined functions. For the initial phase, its main feature is the autonomous operation of human-machine interface devices without human intervention.

The PSQA automation was achieved by emulating the operator's mouse and keyboard operations on those workstations in the PSQA workflow. Implementation of the automation was quite different between the departmental and vendor workstations. That is, the automation software could be installed on the institutional workstations but not on the vendors' clinical workstations that have strict security and FDA safety policies. Therefore, the major obstacles for achieving the L3 automation for our PSQA procedures were the strict security and FDA safety policies of manufacturers' clinical workstations that prevented us from modifying any system setup and installing software on those workstations. As a result, the L3 of automation could only be achieved through the interfaces available for external control of vendors' workstations, for example, those open HID ports for mouse, keyboard, and other external devices.

2.5. Interfaces Design

The *AutoFrame* interface framework includes two interfaces, *AutoFlow* and *PyFlow*, designed for different types of workstations in this project.

AutoFlow: Developed using AutoIt V3 scripting language, *AutoFlow* is an executable file (.exe) that operates in the Windows environment. It controls the graphical HMI (human machine interface) of applications within the operating system.

PyFlow: In contrast, *PyFlow* runs on non-Windows environments. For this study, *PyFlow* was coded in Python and operates on a Raspberry Pi4 Linux sta-

tion, connected to the Varian console workstation through HID ports.

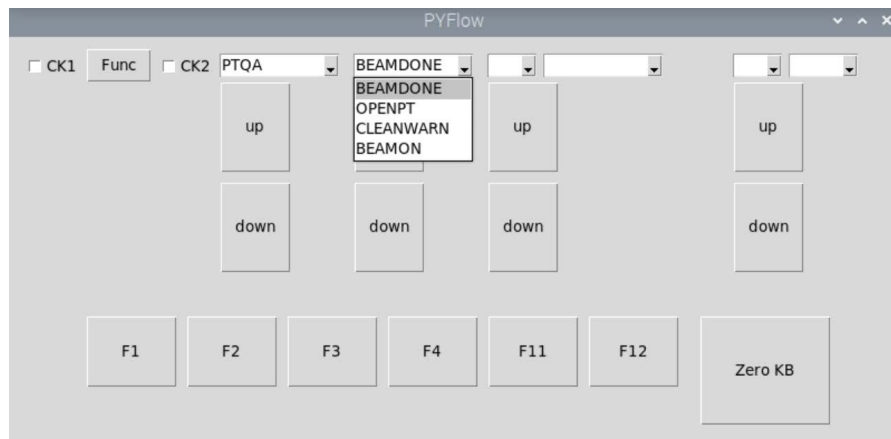
Both AutoFlow and PyFlow can access and control applications on departmental or manufacturer workstations by simulating user mouse and keyboard actions, either inside (AutoFlow) or outside (PyFlow) the operating systems.

Each interface consists of three modules:

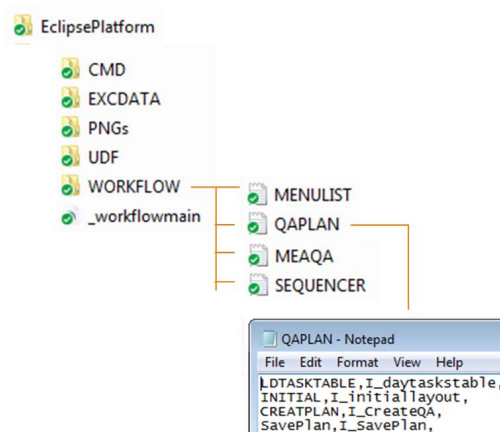
GUI Module: This module contains a task drop-down menu, an operation-step drop-down menu, a general display box, and functional buttons. The task menu lists tasks the interface can perform, while the operation-step menu shows detailed steps for the selected task. Functional buttons are shortcuts for common actions. The GUIs of AutoFlow and PyFlow are shown respectively in **Figure 2(a)** and **Figure 2(b)**.



(a)



(b)



(c)

Figure 2. (a) AutoFLOW and (b) PyFLOW interfaces developed for automating the PSQA process in **Figure 1** by emulating human operations of human interactive devices to manipulate and coordinate applications in different workstations. (c) is the structure of distribution folders that store all executable, open-source script and data files.

UDF Script Interpreter: The interpreter translates plain code scripts into HID operations, enabling communication between the interface and applications.

TCP/IP Communication Module: This module provides a real-time communication link between running interfaces on different computers.

For code distribution, all executable and data files are stored in a shared folder, with several subfolders for different purposes. As shown in **Figure 2(c)**, the UDF subfolder stores plain scripting codes, the EXCDATA subfolder is used for data exchange between interfaces, and the WORKFLOW subfolder contains description files for task and operation-step menus. To install the interface on a destination workstation, the user only needs to copy and paste the entire parent folder.

2.6. Implementation of PSQA Procedure Using the AutoFrame Interface Framework

Four computers (3 departmental computers and 1 vendor computer) connected via the institutional network were involved in the implementation of automated PSQA. Connection of these four computers for the PSQA procedure is depicted in **Figure 3**. As shown in **Figure 3**, each workstation was installed with the AutoFlow/PyFlow interface application developed for this project.

The first task of the automated workflow in **Figure 3** was the creation of a daily QA task file from the patient information SQL server, which was stored in the EXCDATA folder for sharing. This QA task file included all required patients' information for the PSQA of the day. Once the AutoFlow/PyFlow interface application was started, a task bar like the one shown in **Figure 2(a)** or **Figure 2(b)**, would pop up and dock on the top of Windows. Among the tasks listed in the menu, the first one was usually loading the daily task file for synchronizing

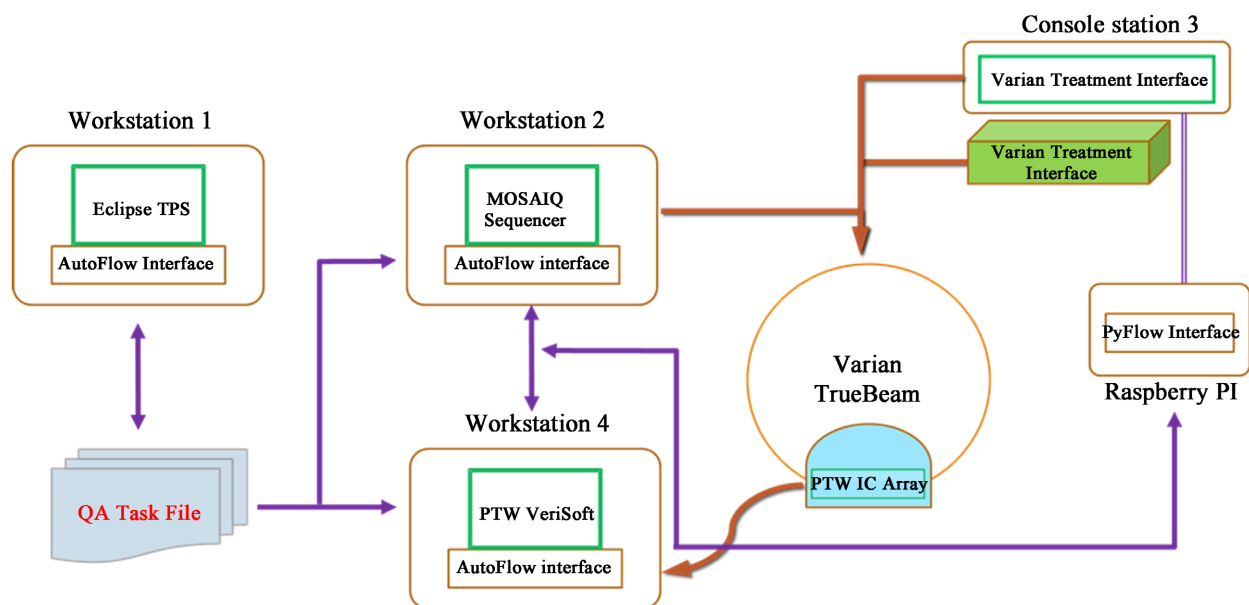


Figure 3. Four computers (3 departmental computers and 1 vendor computer) connected via the institutional network were involved in the implementation of automated PSQA. Workstations 1, 2, and 4 are departmental workstations, with which the AutoFlow interface application was installed. Console station 3 is a vendor computer, with which the PyFlow interface was used.

the patient information. Once the patient data were synchronized, the users would choose additional tasks listed in the operation menus of the application so that they can perform other functions for the PSQA.

For example, the screen shot in **Figure 2(a)** was the AutoFlow interface of Workstation 1 in **Figure 3**, or the workstation for Eclipse treatment planning system. The chosen menu name of the AutoFlow application for this workstation was QAPLAN for plan verification, which has a list of steps for the QA process including initializing screen layout (INITIAL), plan creation (CREATEPLAN) and saving dose plane of TPS (SAVEPLAN). Workstation 2 in **Figure 3** was running the MOSAIQ R&V system. For this workstation, the corresponding menu name was SEQUENCER, and the listed operations were for loading sequencer (LDPT2SEC), removing screen warning (CLEANWARN) and sending the treatment plan to the Varian Truebeam machine (SENPT2VAR). For the measurement/analysis menu (MEAQA) on Workstation 4 for controlling PTW hardware and software, the various steps included loading plan dose (LDPLAN), measurement on (MEAON), measurement off (MEAOFF), analysis (ANALY) and saving report(SAVERPT).

The PyFlow interface shown in **Figure 2(b)** was used only on the Console Workstation in **Figure 3** for interfacing the treatment console, which in turn controlling the Varian machine. There were no interactions between interface and Varian GUI, so the task only required one direction control. The menu name of the PyFlow was PTQA (**Figure 2(b)**), and the listed operation steps included beam done (BEAMDONE), open patient (OPENPT), remove warning (CLEANWARN) and beam on (BEAMON). The beam on step was not actually turning the beam on but just sent the instant notification to other interfaces for additional actions. Since the project only aimed for the L3 level automation, the Varian Console for controlling the machine mechanical motion and actual beam-on was operated manually, and the PyFlow only acted as a message center to update other interfaces the status of the LINAC, *i.e.* beam done, beam on.

2.7. Testing the AutoFrame Interface Framework for L3 Automation of the PSQA Procedure

The developed AutoFrame (AUTOFLOW and PyFlow) interface framework was tested using clinical PSQA cases of our institution. We randomly selected 20 clinical treatments plans with a total of 51VMAT fields for this test. **Table 2** lists the plan characteristics of these tested cases and fields in the first five columns of the table. Each PSQA case was first performed manually to generate the PSQA report for clinical use. The same PSQA measurements were then repeated using the AutoFrame interface as described above. The execution times with and without the use of AutoFrame interface framework were recorded and compared.

3. Results

In this study, we adopted operational time as the only quantitative metric to

Table 2. Plan characteristics of the 20 tested cases are listed in the first five columns. The next three columns are the test results for automated PSQA (or PSQA using the AutoFrame interface framework). The last three columns show the test results for manual PSQA. “Plan”: time in minutes spent for generating and exporting the QA plans. “Meas.”: time in minutes spent on delivering the plans and generating PSQA reports. “10XF”: 10X FFF beam. “6XF”: 6X FFF beam.

Case #	Treatment site	Energy (MV)	# beams	Total MU	Auto PSQA time (min)			Manual PSQA time (min)		
					Plan	Meas.	Total	Plan	Meas.	Total
1	Partial BrainR	6X	3	589	2.2	5.4	7.6	4.6	7	11.6
2	R-Cerebellar	6X	3	708	2.8	5.6	8.3	6.2	8	14.2
3	Abdomen	6X	2	600	2.1	5.6	7.7	3.7	8.1	11.8
4	Pelvis_Anorectum	6X	3	651	2.7	5.2	7.9	3.7	9	12.7
5	Bladder CD	6X	2	867	2.1	7.2	9.3	5.7	12.3	18
6	PELVIS_ANUS	6X	3	501	2.5	4.8	7.3	4.1	6.3	10.4
7	Pelvis	6X	2	562	2.3	5.2	7.5	5.1	6.1	11.2
8	Lung_R	6X	2	703	1.9	6.1	7.9	2.8	10	12.8
9	Partial Brain CD	6X	3	651	2.9	5.0	7.9	5.8	6.8	12.6
10	Prostate_SV_Nodes	6X	2	841	2.1	7.0	9.2	4	13.5	17.5
11	Pancreas	10XF	2	2780	2.4	7.2	9.6	7.2	12.1	19.3
12	Bilateral_Brain	6X	2	446	2.3	5.0	7.2	4.8	5.2	10
13	Stomach	6X	2	541	1.8	5.7	7.5	3.7	7.3	11
14	Bladder	6X	2	793	2.3	6.1	8.4	5	9.6	14.6
15	skin cheek_Neck_R	6X	2	596	1.9	5.8	7.7	3.2	8.6	11.8
16	left iliac nodes	6XF	4	2360	2.4	7.2	9.6	6.5	12.7	19.2
17	Breast_Chest_LN_R	6X	4	1018	2.4	6.9	9.3	5.8	12.2	18.1
18	Prostate Bed	6X	3	804	2.1	6.3	8.4	4.1	10.6	14.7
19	OropharynxNeck	6X	3	840	2.3	6.2	8.5	5	9.8	14.7
20	Esophagus_LN	10X	2	614	2.2	5.5	7.7	4.6	7.4	12
Avg.			2.55	873.3	2.29	5.95	8.23	4.78	9.13	13.91
Std.			0.69	601.0	0.29	0.79	0.78	1.17	2.46	3.01

assess the efficiency of our PSQA procedure. Just like other clinical workflows, PSQA can be adversely affected by operator stress, fatigue, and various other variables [12]. In our study, they were not separately evaluated due to limitations imposed by institutional resources. We would like to point out that operational time can also be affected by these influencing factors due to time delay by error correction, and therefore could serve as an effective and comprehensive metric for all factors.

For the cases tested, a total of 175 (± 12) human HID operations on computers were eliminated from the PSQA procedure in our current clinical environments. Columns 6 - 11 of **Table 2** list the testing results for all tested cases, where “Plan”

indicates the time in minutes spent on generating and exporting the QA plans, and “Measurement” is the time in minutes spent on delivering the plans and generating PSQA reports. Note that the time for detector calibration and phantom setup was not included in the recorded time since it is always done before the PSQA and takes the same amount of time for both (manual and automated) PSQA workflows.

4. Discussions

In this study, we developed a general automation interface framework, the AutoFrame for medical physics practices in a radiation oncology department. As the first application of this framework, we implemented two embedded subsystems—the AutoFlow and PyFlow in the AutoFrame interface framework for automating the HID operations of PSQA workflow in our clinic. With the implemented software, all user operations outside the treatment room for the PSQA of VMAT plans were automated successfully except for the VARIAN Controller operations. We thus concluded that L3 PSQA automation was achieved based on the six levels of autonomy (**Table 1**) we defined for the PSQA procedure.

The AutoFrame interface framework was designed to achieve the L3 automation without violating any FDA safety and institutional/vendors’ security policies mentioned above. This was accomplished by the development of the two embedded AutoFlow and PyFlow subsystems. The AutoFlow subsystem was installed and running on the (MS Windows) operation system of the institutional workstations for automating the HID operations on those workstations. For vendors’ workstations that do not allow the installation of AutoFlow, the PyFlow subsystem was used in its place. PyFlow is an independent interface package that doesn’t need to be installed on those clinical workstations or be recognized by any vendors’ applications. Instead, it is running on a Raspberry Pi connected to the vendors’ workstations through the open HID (mouse and keyboard) ports.

The major improvement in efficiency of our automated PSQA procedure was illustrated by the shortened operation time and reduced human (control/clicking/typing) errors. Note that the time spent in dose calculation, beam delivery, gamma index calculation, and report file exporting is somewhat constant, but the time required for human intervention varies significantly depending on the load of a clinical day. Our results show that the standard deviation of the operation time for manual PSQA was more than three times larger than that for automated PSQA. This difference was especially pronounced on days with a higher clinical workload.

For example, on a normal clinical day, our QA technician could get the LINAC around 7:00 pm and finish four or less IMRT QA cases in less than one and a half hours. On the other hand, if the number of IMRT QA cases was unusually high, e.g. more than six, the QA technician wouldn’t be able to finish all tests until after 9:30 pm as the operator tended to spend a longer time per case due to the stress and fatigue from staying late. Even when the number of IMRT QA

cases is normal, the time spent on each case would be longer if the clinic finished treatment late like 8 pm or even 9 pm. For automated PSQA, this time requirement is linearly proportional to the number of QA cases, no matter how late the machine was available or how many IMRT QA cases needed to be completed.

There are two additional gains in efficiency for the automated PSQA. First, automated operation of HID is much faster than human operation, and this difference is especially obvious for those more complex operations in the procedure. For example, with AutoFrame, all required mouse clicks and keystrokes for the MOSAIQ sequencer could be completed in less than three seconds without errors, while it would take at least 30 seconds or more for a human operator, usually accompanied with a few repeated steps to fix operational errors.

Another advantage of automated PSQA is the ability to perform parallel operations. Typically, only one QA technician is assigned to the PSQA procedure every day at our clinic so the QA technician must complete tasks linearly even though some steps can be done in parallel. For example, with our automated PSQA, the QA technician can import the patient information for the next case in the MOSAIQ sequencer at the same time as the QA report of the current patient is being generated. Another example would be the dose plane of the patient is imported into the Verisoft workstation for gamma analysis simultaneously when the PyFlow is opening the plan in the Varian workstation for beam delivery. These parallel operations in PSQA can be synchronized and executed remotely via the communication module in the AutoFrame interface framework.

In addition to speeding up the PSQA workflow, eliminating human errors associated with HID usage and reducing stress, the AutoFrame also provides a structured and distributable platform for automating other clinical workflows with minimal coding effort. Automation with the AutoFrame does not change the existing clinical workflow and provides compatibility and redundancy in case of automation failure. That is, if the automation fails (which hasn't happened to our automated PSQA), we can remedy the situation immediately with manual operations. Additionally, to ensure proper quality assurance, the PSQA workflow was designed and periodically reviewed to make sure it is aligned with Northwell Health guidelines for quality and safety. During the IMRT QA, the operator is required to monitor the entire procedure closely with or without the automation. Furthermore, the AutoFrame can be easily expanded to provide functional modules for communication with workstations within existing workflows. This is achieved by the adoption of plain script coding that allows "what you operate is what you code" approach. Therefore, it is easy to add new functions to the framework and the learning curve for most new functions is usually very short, as it doesn't rely on any other complicated programming languages or specific software development kits provided by vendors.

One major concern of automation with the scripting on HID is the consistency of mouse and keyboard workflow on PSQA. Indeed, variations in patient-specific warnings and other factors like the number of prescriptions and prior treatments

can impact the reactions of AutoFrame. We anticipated these challenges and have implemented: 1) workflow standardization and 2) software enhancement to minimize the chance of happening. “Workflow standardization” involved performing a comprehensive review of our PSQA workflow to identify various clinical scenarios that can be handled with common mouse and keyboard operations. In addition, we’ve aligned our clinical procedures with the primary guidelines of clinical operations, *i.e.* the Northwell policy and procedures. By minimizing the divergence in HID operations, we were able to maintain a high internal consistency for the variables of our PSQA procedure.

For “software enhancement”, we have taken measures to address issues related to warnings and messages. Our approaches involved the detection of warning messages or their corresponding subregion, such as the “OK” button, through characteristic sub-image analysis. This allowed us to identify and streamline extra HID operations, contributing to a more seamless workflow.

One challenge of implementing a new procedure like the automated PSQA workflow is that it might introduce new failure modes, *e.g.* unanticipated errors from the parallel processing mentioned above. Note that detecting potential failure modes and fixing identified safety barriers are integral components of our departmental overall QA (including PSQA) process, which are constantly being performed at our clinic. These safety measures cover not only the technical aspects but also the involved human operators.

While no AutoFrame-associated error modes have been identified so far, we recognize that they will surface eventually and have implemented some precautionary measures. First, we would like to reemphasize that the operator’s role in monitoring the PSQA process remains essential in the automated workflow even if most manual operations are eliminated. That is, the QA technicians are required to monitor the whole automated process visually and be ready to step in at any time and complete the PSQA task when necessary. With this requirement, in the rare event of a failure or an issue during the automated process, human intervention will be provided immediately, ensuring that patient safety is maintained, just as it would be in a non-automated procedure. Regarding parallel operations, it’s worth noting that in the current PSQA workflow, parallel operations are not exclusive to the implementation of AUTOFRAME. That is, when multiple QA technicians are available for IMRT QA, they always function in parallel. Thus, parallel operation is already a part of our workflow, and we have implemented measures to manage and mitigate any potential issues. Of course, these measures were reviewed and updated when the AUTOFRAME was introduced.

Even with the above-mentioned precautions, there might still be unexpected errors or risks associated with HID usage, *e.g.* mis-clicking other parts of the workstation due to unanticipated pop-up windows. These types of error/risk are the major worries for L5 automation aiming for the complete automation without human interventions, but not for L3 or even L4 automation, which still requires certain degrees of human involvement to ensure proper quality assurance.

Finally, there are limitations to automation through the HID devices. Particularly, it is obvious from this study that the beam-on and beam-off at the LINAC require the user physically pushing a few buttons on the LINAC control console. Because of this limitation, the goal of this study was set to achieve L3 automation. L4 or higher automation requires a mechanical solution that can push those buttons for the user. We are currently working on this mechanical solution and the results will be published in the near future.

In addition, although our study primarily focused on the operational time as the only metric, we acknowledge the broader impact of variables such as erroneous keyboard inputs and operator stress, which can manifest in QA time. Quantifying human errors and fatigue is not trivial and is beyond the scope of this study. As a result, the metrics like human errors and operator stress were not investigated in this study even though we recognize the potential value of including those metrics.

5. Conclusion

In conclusion, we have demonstrated in this study that the AutoFrame interface framework can substantially improve the efficiency and accuracy of our PSQA procedure. The introduction of AutoFrame primarily aims to reduce the potential risks and minimize the possibility of errors or inconsistencies associated with human operators. Automating the operations of HID in computers not only shortens the time required for performing the PSQA, but also reduces the human errors resulting from stress and fatigue in clinics. The AutoFrame interface framework can potentially be applied to other clinical procedures of medical physics, *i.e.* the monthly/annual QA of LINAC or Gamma Knife to improve efficiency and reliability.

Acknowledgements

The authors would like to thank Northwell Health for providing its facilities and software for this study. This research was partly funded by Hofstra University's 2022-2023 FRDG award, 2022-2023 PRAP award, and 2023-2024 FRDG award (Recipient: Jenghwa Chang).

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

- [1] Saiful Huq, M., Fraass, B.A., Dunscombe, P.B., *et al.* (2016) The Report of Task Group 100 of the AAPM: Application of Risk Analysis Methods to Radiation Therapy Quality Management. *Medical Physics*, **43**, 4209-4262. <https://doi.org/10.1118/1.4947547>
- [2] Alfredo Siochi, R., Balter, P., Bloch, C.D., *et al.* (2021) Report of Task Group 201 of the American Association of Physicists in Medicine: Quality Management of External Beam Therapy Data Transfer. *Medical Physics*, **48**, e86-e114.

- <https://doi.org/10.1002/mp.14868>
- [3] Leal, A., Sánchez-Doblado, F., Arráns, R., Roselló, J., Pavón, E.C. and Lagares, J.I. (2003) Routine IMRT Verification by Means of an Automated Monte Carlo Simulation System. *International Journal of Radiation Oncology, Biology, Physics*, **56**, 58-68. [https://doi.org/10.1016/S0360-3016\(03\)00067-1](https://doi.org/10.1016/S0360-3016(03)00067-1)
- [4] Miften, M., Olch, A., Mihailidis, D., *et al.* (2018) Tolerance Limits and Methodologies for IMRT Measurement-Based Verification QA: *Recommendations of AAPM Task Group No. 218. Medical Physics*, **45**, e53-e83. <https://doi.org/10.1002/mp.12810>
- [5] Low, D.A. and Dempsey, J.F. (2003) Evaluation of the Gamma Dose Distribution Comparison Method. *Medical Physics*, **30**, 2455-2464. <https://doi.org/10.1118/1.1598711>
- [6] Pawlicki, T., Yoo, S., Court, L.E., *et al.* (2008) Moving from IMRT QA Measurements toward Independent Computer Calculations Using Control Charts. *Radiotherapy & Oncology*, **89**, 330-337. <https://doi.org/10.1016/j.radonc.2008.07.002>
- [7] Pawlicki, T., Yoo, S., Court, L.E., *et al.* (2008) Process Control Analysis of IMRT QA: Implications for Clinical Trials. *Physics in Medicine & Biology*, **53**, 5193-5205. <https://doi.org/10.1088/0031-9155/53/18/023>
- [8] Zhang, J.Q., Ahunbay, E. and Li, X.A. (2018) Technical Note: Acceleration of On-line Adaptive Replanning with Automation and Parallel Operations. *Medical Physics*, **45**, 4370-4376. <https://doi.org/10.1002/mp.13106>
- [9] Synopsys (2023) The 6 Levels of Vehicle Autonomy Explained. <https://www.synopsys.com/automotive/autonomous-driving-levels.html>
- [10] Hussein, M., Adams, E.J., Jordan, T.J., Clark, C.H. and Nisbet, A. (2013) A Critical Evaluation of the PTW 2D-ARRAY Seven29 and OCTAVIUS II Phantom for IMRT and VMAT Verification. *Journal of Applied Clinical Medical Physics*, **14**, 274-292. <https://doi.org/10.1120/jacmp.v14i6.4460>
- [11] Das, S., Kharade, V., Pandey, V.P., Kv, A., Pasricha, R.K. and Gupta, M. (2022) Gamma Index Analysis as a Patient-Specific Quality Assurance Tool for High-Precision Radiotherapy: A Clinical Perspective of Single Institute Experience. *Cureus*, **14**, e30885. <https://doi.org/10.7759/cureus.30885>
- [12] Di Muzio, M., Dionisi, S., Simone, E.D., *et al.* (2019) Can Nurses' Shift Work Jeopardize the Patient Safety? A Systematic Review. *European Review for Medical and Pharmacological Sciences*, **23**, 4507-4519. https://doi.org/10.26355/eurrev_201905_17963