Clinica Chimica Acta 495 (2019) 625–629

Contents lists available at ScienceDirect

Clinica Chimica Acta

journal homepage: www.elsevier.com/locate/cca

Review

Recommendations for laboratory informatics specifications needed for the application of patient-based real time quality control

Tze Ping Loh[a], Mark A. Cervinski[b,h], Alex Katayev[c], Andreas Bietenbeck[d], Huub van Rossum[e,f], Tony Badrick[g,*], on behalf of the International Federation of Clinical Chemistry and Laboratory Medicine Committee on Analytical Quality

[a] National University Hospital, Singapore
[b] Dartmouth-Hitchcock Medical Center, Lebanon, NH, USA
[c] Laboratory Corporation of America Holdings, Elon, NC, USA
[d] Institute of Clinical Chemistry and Pathobiochemistry, Klinikum rechts der Isar, School of Medicine, Technical University of Munich, Germany
[e] The Netherlands Cancer Institute, Amsterdam, The Netherlands
[f] Huvaros, The Netherlands
[g] RCPA Quality Assurance Programs, St Leonards, Sydney, Australia
[h] The Geisel School of Medicine at Dartmouth, Hanover, NH, USA

ARTICLE INFO

Keywords:
Patient based real time quality control
Moving averages
Middleware
Data mining

ABSTRACT

Patient based real time Quality Control (PBRTQC) algorithms provide many advantages over conventional QC approaches including lower cost, absence of commutability problems, continuous real-time monitoring of performance, and sensitivity to pre-analytical error. However, PBRTQC is not as simple to implement as conventional QC because of the requirement to access patient data as well as setting up appropriate rules, action protocols, and choosing best statistical algorithms. These requirements need capable and flexible laboratory informatics (middleware). In this document, the necessary features of software packages needed to support PBRTQC are discussed as well as recommendations for optimal integration of this technique into laboratory practice.

1. Introduction

The traditional internal quality control (QC) practice does not always provide adequate quality assurance. Some recent examples of these issues include: failure to detect long-term analytical drift (insulin-like growth factor-1 assay) [1,2], failure to detect increased imprecision at very low but clinically important concentrations (troponin T assay) [3], failure to detect a small positive bias at the lower limit of analytical measurement range (prostate-specific antigen assay) [4], and a failure to detect major analytical shift (gentamicin assay) [5]. All QC procedures are retrospective and can only detect error at the next measurement of QC sample and many laboratories will only perform QC events once or twice per day, even with high patient workloads.

Traditional QC material may be non-commutable and the rules insensitive to assays with low analytical performance [6,7]. Patient-based real time QC (PBRTQC) techniques have been shown to be highly capable in the routine monitoring the performance of an analytical system [8–12]. Most importantly, PBRTQC practice directly links the performance of the analytical system to patient results and uses data already available in the information system. In other words, it is risk-based, clinically relevant and cost-efficient. Detailed discussions about PBRTQC has been reviewed elsewhere [6,9,13–15].

PBRTQC should be operated on software packages designed for use in clinical laboratories. To varying degrees, analyzer software, middleware, and laboratory information systems can all support PBRTQC. For effective use of PBRTQC, these software packages should incorporate certain basic functions. The PBRTQC settings are often based on analysis performed on dedicated simulation software prior to implementation on the purpose-built production PBRTQC software. Therefore, proper data-export is required. After establishing PBRTQC settings and rules either by dedicated external applications or by studying laboratory specific patient-populations, data simulation

Abbreviations: PBRTQC, patient-based real time quality control; IQC, internal (conventional) quality control (typically liquid or lyophilized); EWMA, exponentially weighted moving average; LIS, Laboratory Information System

[*] Corresponding author.

E-mail address: tony.badrick@rcpaqap.com.au (T. Badrick).

https://doi.org/10.1016/j.cca.2019.06.009
Received 24 April 2019; Accepted 7 June 2019
Available online 10 June 2019

Table 1
Data to be captured by laboratory informatics for effective application of PBRTQC.

Parameters
Date and time (hours, minutes, seconds) of laboratory analysis
Sample identification
Date of birth/ age (at least to one decimal place) of patient
Patient identification number
Identification of non-patient materials (external QC, dialysis fluids, animal research samples, research samples, etc.)
Gender of patient
Laboratory instrument used
Testing laboratory
Sample type
Test name
Sample collection condition (e.g. timed specimen if not defined by test name)
Ordering location
Ordering clinical service or type of practice
Laboratory result
Reagent lot in use
Hemolysis/ icteric/ lipemic indices/Instrument flags
Initial analysis or re-run status of sample
Units of measurement
Previous patient test result if available

studies, and extracted PBRTQC settings, implementation on purpose-built software can begin.

This document seeks to describe and summarize the core functionalities of laboratory informatics (middleware/ laboratory information system) that are crucial for the adoption of PBRTQC. It is hoped that manufacturers of analytical systems, middleware and laboratory information systems (LIS) vendors will incorporate these functionalities in their laboratory informatics products to enable wider adoption of this next-generation QC practice.

2. Data capture

An output can only be as good as its input. It is important to ensure that the data captured in the information system contains all the necessary detail to enable optimal downstream analysis. As a minimum, the data in Table 1 should be included.

Ideally, the laboratory results should only contain numeric values for easier evaluation. Non-numerical data should be excluded with a possible exception for algorithms developed specially for qualitative methods. Additional information that may be helpful include: clinical diagnosis, active medication, timing of medication. This can be captured either by direct interface with the electronic medical records or by manual input at clinician or laboratory request ordering.

Beyond the patient results, it is also important to capture data from the routine internal QC run. Comprehensive quality control measures should incorporate multiple layers of performance controls including conventional QC materials, PBRTQC, flagging rates (number of abnormal results). These various quality control measures can be layered on top of the PBRTQC for more comprehensive interpretation and troubleshooting [12].

3. Data storage

Data security is a major safety and privacy concern. Laboratory data should be stored in a secured environment. It should comply with local regulatory requirements for data safety and patient privacy (i.e. encryption). The data should be stored in a manner that allows real time data extraction, at user request for exportation to third-party application. When data is extracted for analysis it is important to de-identify the dataset if working in a non-secure environment or if the data is to be transferred to a third-party application for the purposes of developing PBRTQC methods.

4. Data extraction and analysis

For PBRTQC to be effective and assay performance to be continuously monitored, laboratory data needs to be extracted and analyzed in real time by the algorithm within the middleware of LIS [8–12]. There should be a user-friendly interface that allows the user to

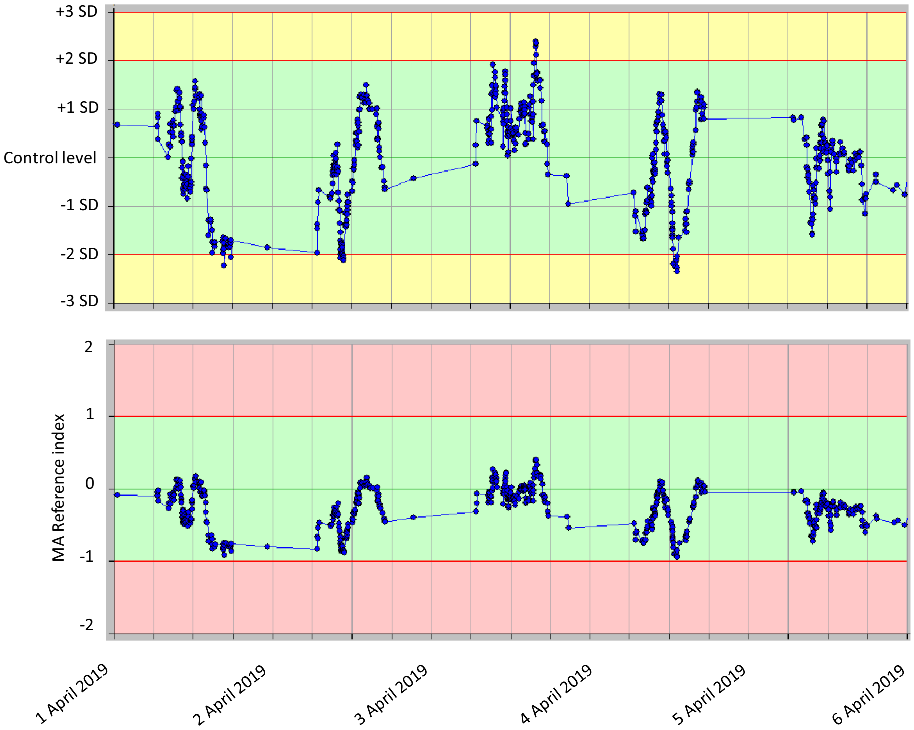

Fig. 1. Examples of patient-based real-time quality control presentation in Levey-Jennings plot (upper) and accuracy/ normal value plot (lower). PBRTQC of creatinine of the first 5 days of April 2019 is presented using the settings as described previously in reference [6].

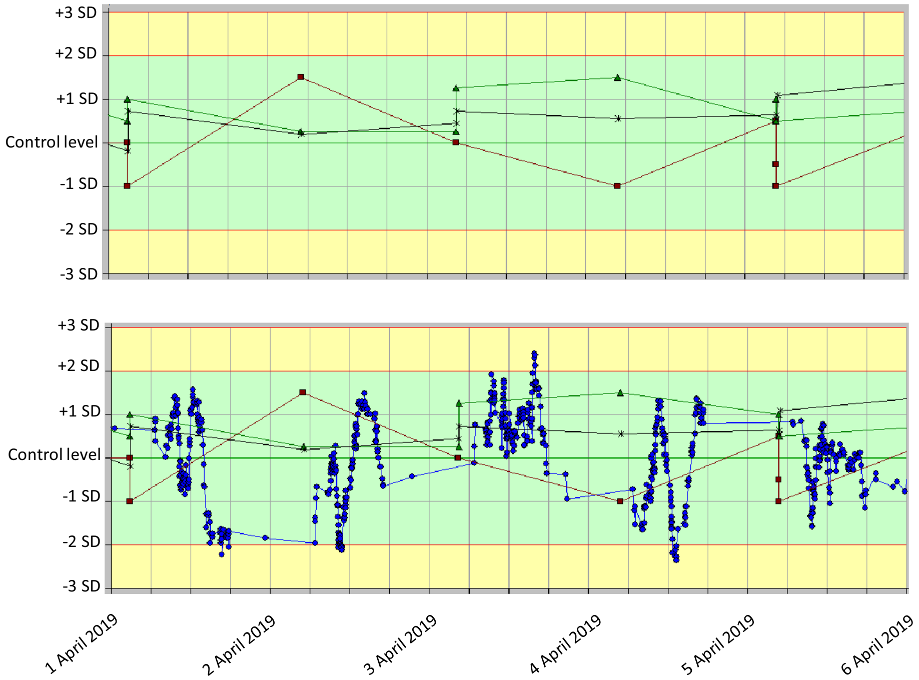

Fig. 2. Example of presenting PBRTQC in same Levey-Jennings graph. Upper presents QC of creatinine during the first 5 days of April 2019. Lower presents PBRTQC procedures together with internal QC in one graph. PBRTQC is the same as presented in Fig. 1.

Table 2
Essential features of middleware/ laboratory information for successful implementation and operation/ application of patient-based real-time quality control (PBRTQC).

Features	Description
Data capture	This is the data that needs to be routinely captured by the laboratory informatics, which will be extracted, processed and analyzed for PBRTQC. Details of the features are provided in Table 1
Data storage	The laboratory data needs to be stored in a secure environment that allows real-time access
Data extraction and analysis	Data extraction must be real-time and can be exported in common format for third party application. Must support more sophisticated data extraction, e.g. using SQL scripts
Data visualization and exploration	Must allow simple data visualization, e.g. in scatter plots/ histograms for exploration. Must allow simple statistical analysis, e.g. calculation of mean, median, standard deviation, percentiles and application of exclusion criteria and truncation limits
Data transformation	Must support common data transformation in real-time for analysis
Statistical analysis and testing environment	Have a sandbox/ simulation environment with sufficient historical data to test out performance of algorithm and ability to apply variable artificial errors to pre-defined result strings. Alternately, allow algorithm to be run 'live' without triggering alarms. Allow user to export a report of all selected parameters and their performance
Live application and reporting	Allow visualization of data (e.g. in Levi-Jennings chart) and user-defined alarms. Allow recording/ audit trail of troubleshooting steps. Allow reset of algorithm after reset of alarm. Allows auto-generation of PBRTQC report at defined intervals
Audit trail	Keep an audit trail of key activities in the PBRTQC system, including changes in parameters, switching on or off the PBRTQC, reset/ acknowledgement of alarms, troubleshooting comments

Table 3
Additional software/features necessary for optimal implementation of patient-based real-time quality control (PBRTQC) that may be performed by middleware/ laboratory information system or standalone statistical software. These features may represent market opportunity for new/ non-traditional software vendor or key differentiating features.

Features	Description
Advanced data visualization	A multi-dimensional visualization tool that allows cross-examination of the relationship between different parameters (see Table 1), for example by using colour for annotation of multi-dimensional graphs
Formal statistical analysis	Formal statistical tests for normality of data distribution (for example, Shapiro Wilk test) may be required to assess the need for data transformation to approximate a Gaussian distribution
Iterative visualization/ statistical analysis	The users should be able to examine the data by iterative application of different filters or exclusion criteria, for example, applying truncation limits or filtering by ordering location, before subjecting the data to repeat statistical analysis or graphical visualization
Incorporation of internal quality control data	Allow presentation of PBRTQC together with IQC and the option of presenting both in separate graphs

easily select and extract any of the parameters in Table 1, including exporting the data for third party application, where necessary. Additionally, the database should also allow more advanced data extraction using scripted queries, such as Structured Query Language scripts.

Smaller datasets may be able to be handled and transferred as simple comma separated value documents, however larger datasets may require the development of an additional interface to export data to a third-party application.

5. Data visualization and exploration

The first step to setting up a PBRTQC system is to become familiar with the laboratory data and pattern of ordering within an individual laboratory. This is best carried out by data visualization. It is desirable that the laboratory informatics should allow visual exploration of the laboratory data by a simple scatter plot and histogram between any two of the parameters in Table 1. Better still, the visualization tool should allow higher dimensional cross-examination, for example by using colour for annotation of multi-dimensional graphs.

The laboratory informatics should include access to simple descriptive statistical tools to explore the database/ data set. These should allow users to calculate simple statistics such as mean, median, percentile values, standard deviation, and flagging rates. Additionally, the users should be able to 'slice and dice' the data by applying different filters or exclusion criteria, for example, applying truncation limits or filtering by ordering location, before subjecting the data to repeat statistical analysis or graphical visualization using neat or transformed values. The data visualization and exploration applications should be integrated to allow iterative evaluation of the data.

6. Data transformation

Following data visualization and exploration, it may become apparent that certain laboratory data require additional transformation to approximate a Gaussian distribution for optimal performance [9]. Formal statistical tests for normality of data distribution (for example, Shapiro Wilk test) may be required to assess the need for data transformation to approximate a Gaussian distribution. This may be performed in separate statistics software, if not already incorporated in the middleware/ LIS.

To accommodate this need, the software should be able to apply transformations such as a natural logarithm, square root, multiplicative inverse, reciprocal, Box Cox and power transformations, and the calculation of various ratios. This functionality exists in some currently available software packages [9].

7. Statistical analysis and testing environment

Because PBRTQC relies on laboratory results that are generated from routine testing, it is recommended that a testing 'sandbox' environment be made available during the set-up phase. The testing environment should as much as possible mirror the live production environment with the same data structure. The testing environment should contain historical data of sufficient duration (minimum of two years, to account for seasonal variation of some tests) to allow a proper selection and assessment of performance of PBRTQC algorithms and rules setup. Analogous systems exist for some LIS vendors that allow a laboratory to 'build' a new test or calculation (such as estimated glomerular filtration rate) in a test environment prior to implementation. Alternately, the software should allow the operation of PBRTQC without alarms to allow the users to passively assess the clinical appropriateness of the selected algorithm parameters with minimal disruption to routine operations [16].

In this safe testing environment, the PBRTQC can be set up by first defining the inclusion and exclusion criteria based on the attributes of the laboratory tests as described in Table 1 (for example, ordering location, clinical service). Additionally, the laboratory informatics system should allow the application of truncation limits to reduce the biological variation (or "noise") of the laboratory results. The truncation limits can be applied before or after data transformation.

Following this, the laboratory informatics system should offer several statistical algorithms to calculate PBRTQC. They should minimally include simple moving average, exponentially weighted moving average and moving median [6]. Additionally, XbarB (or Bull's algorithm) is frequently used for hematology analyses. There should also be a parameter for qualitative tests such as sum or percentage of outliers/ positives/ negatives or ability to capture instrument numerical signal response (like optical density or absorbance units). The inclusion of other less common statistical algorithms including moving standard deviation, moving percentiles, and moving flagging rates, among others, are advantageous.

There should be flexibility to calculate the moving statistics based on the variable block size (the number of patient samples used to calculate the mean or median using neat or transformed results values) and frequency - i.e. when the block of result moves. For example, to recalculate the moving statistics on first-in-first-out basis by sequential replacement with the latest result or moving at fixed interval of number of new results. The algorithm may also include more sophisticated statistical approaches such as exponential weighting [6].

After the above parameters have been established, the simulation environment should allow the application of the above algorithm on historical laboratory data. The moving statistics should be presented as control charts with appropriate warning and alarm when a control limit violation occur. This will allow the laboratory user to gain an understanding of the behavior of the algorithm in their practice. Simulation functions should allow application and assessment of different degrees of analytical bias or imprecision introduced randomly or systematically into the historical laboratory data. A resampling function (e.g. bootstrapping, cross validation) will allow the user to characterize the performance of the PBRTQC set up more accurately. The outcome of the performance evaluation should ideally be expressed in probability of error detection/ false rejection and average/ median number of patient results affected before error detection (may require resampling) [6].

There should be a result/ report export function that includes all the user-defined parameters and evaluation results to allow user to properly document the setup exercise and verification process prior to live application to meet regulatory requirements.

8. Live application and reporting

Once the user is satisfied and confident with the setup of the PBRTQC, the algorithm can be moved into the live environment. In this setting, it is important for the laboratory informatics system to present the PBRTQC statistics in a control chart form that is easy to interpret, for example in Levey-Jennings or accuracy graph format, depending on the optimization method used (origin of control limits applied) (Fig. 1). There should be clear user-defined real-time alarms (in the form of audio, visual, voice, e-mail, text messaging) for different degrees of control limit violation, which can be acknowledged and reset after troubleshooting. There should be an ability to reset the algorithm and exclusion of specific patients after a workup of the alarm [6].

The time and name of the person who acknowledged the alarm should be documented. Ideally, there should be a text box where user can input their interpretation and troubleshooting measures and annotated on the control chart. A readily generated report that displays the patients previously analyzed on the failed assay or analyzer should be available to help the technologists find potentially affected patient samples for reanalysis. Ideally, retrieval of affected samples process should be automated.

For full interpretation and incorporation into the laboratory quality system, PBRTQC should be able to be presented in conjunction to the internal QC (IQC) graph depending on the laboratory preferences [11,17], such as:

- Support graphical presentation in Levey-Jennings as well as accuracy (PBRTQC normal value) graph. (see examples in Fig. 1)
- Allow presentation of PBRTQC together with IQC and the option of presenting both in separate graphs (Fig. 2).
- Allow user defined options for different presentation modes like displaying continuous daily monitor, monitor per instrument or per entire laboratory or defined department

- Support remote control of performance monitoring.

Once the laboratory user becomes familiar with the PBRTQC set up, they may subsequently set rules where release of patient results are withheld until the moving block of results passed applicable rules without alarm allowing the release of the first result from the block or if there is a PBRTQC rules violation, then no result from this block is allowed to be released (i.e. "release from back" strategy) [9]. There should be a result/ report export function to allow user to properly document the performance of the analytical system as monitored by the PBRTQC system. Ideally, there should be a function to auto-generate and export the report at defined intervals or period.

9. Interface with external quality control initiatives

There are ongoing efforts to increase adoption of PBRTQC practice, including an international network to monitor the performance of analytical systems by pooling such data [18,19]. It is foreseeable that some of these international efforts may become routine practice in the future. As such, there should be ability to export or interface with these external quality control initiatives to simplify participation and assessment.

10. Conclusion

PBRTQC approaches have been shown to be powerful tools that are mature for routine adoption. However, the lack of user familiarity and limitations of laboratory informatics have thus far constrained its implementation. This document serves to provide a list of recommended specifications to laboratory informatics developers to consider and incorporate in their product to enable the routine practice of next generation laboratory QC. The key features for a middleware package to allow a successful PBRTQC implementation are summarized in Table 2. Table 3 provides a set of additional features which would support advanced applications for a PBRTQC system.

Funding

None.

Ethical approval

Not applicable.

Guarantor

TB

Contributorship

All authors reviewed and edited the manuscript and approved the final version for publication.

Declaration of Competing Interest

None.

References

[1] A. Algeciras-Schimnich, D.E. Bruns, J.C. Boyd, S.C. Bryant, K.A. La Fortune, S.K. Grebe, Failure of current laboratory protocols to detect lot-to-lot reagent differences: findings and possible solutions, Clin. Chem. 59 (8) (2013) 1187–1194.

[2] J. Liu, C.H. Tan, T.P. Loh, T. Badrick, Detecting long-term drift in reagent lots, Clin. Chem. 61 (10) (2015) 1292–1298.

[3] P.E. Hickman, G. Koerbin, T. Badrick, C. Oakman, J.M. Potter, The importance of low level QC for high sensitivity troponin assays, Clin. Biochem. 58 (2018) 60–63.

[4] J. Liu, C.H. Tan, T. Badrick, T.P. Loh, Moving standard deviation and moving sum of outliers as quality tools for monitoring analytical precision, Clin. Biochem. 52 (2018) 112–116.

[5] G. Koerbin, J. Liu, A. Eigenstetter, C.H. Tan, T. Badrick, T.P. Loh, Missed detection of significant positive and negative shifts in gentamicin assay: implications for routine laboratory quality practices, Biochem. Med. (Zagreb) 15 (2018) 28 (1):010705.).

[6] H.H. van Rossum, Moving average quality control: principles, practical application and future perspectives, Clin. Chem. Lab. Med. (2018), https://doi.org/10.1515/cclm-2018-0795.

[7] W.G. Miller, A. Erek, T.D. Cunningham, O. Oladipo, M.G. Scott, R.E. Johnson, Commutability limitations influence quality control results with different reagent lots, Clin. Chem. 57 (1) (2011) 76–83.

[8] S. Hayashi, K. Ichihara, Y. Kanakura, Y. Iwatani, A new quality control method based on a moving average of "latent reference values" selected from patients' daily test results, Rinsho Byori 52 (3) (2004) 204–211.

[9] J.K. Fleming, A. Katayev, Changing the paradigm of laboratory quality control through implementation of real-time test results monitoring: for patients by patients, Clin. Biochem. 48 (2015) 508–513.

[10] D. Ng, F.A. Polito, M.A. Cervinski, Optimization of a moving averages program using a simulated annealing algorithm: the goal is to monitor the process not the patients, Clin. Chem. 62 (10) (2016) 1361–1371.

[11] H.H.V. Rossum, H. Kemperman, Implementation and application of moving average as continuous analytical quality control instrument demonstrated for 24 routine chemistry assays, Clin. Chem. Lab. Med. 26 (2017) 1142–1151.

[12] A. Bietenbeck, M.A. Thaler, P.B. Luppa, F. Klawonn, Stronger together: aggregated Z-values of traditional quality control measurements and patient medians improve detection of biases, Clin. Chem. 63 (8) (2017) 1377–1387.

[13] T. Badrick, M. Cervinski, T.P. Loh, A primer on patient-based quality control techniques, Clin. Biochem. 64 (2019) 1–5.

[14] T. Badrick, A. Bietenbeck, M.A. Cervinski, A. Katayev, H.H. van Rossum, T.P. Loh, Patient-based real-time quality control: review and recommendations, Clin. Chem. (2019) (in press).

[15] J. Liu, C.H. Tan, T.P. Loh, T. Badrick, Verification of out-of-control situations detected by "average of normal" approach, Clin. Biochem. 49 (2016) 1248–1253.

[16] H.H. van Rossum, D. vd Broek, Design and implementation of quality control plans that integrate moving average and internal QC: incorporating the best of both worlds, Clin. Chem. Lab. Med. (2019), https://doi.org/10.1515/cclm-2019-0027 (In press).

[17] K. Goossens, K. Van Uytfanghe, P.J. Twomey, L.M. Thienpont, Participating Laboratories. Monitoring laboratory data across manufacturers and laboratories - a prerequisite to make "Big Data" work, Clin. Chim. Acta 445 (2015) 12–18.

[18] L.A.C. De Grande, K. Goossens, K. Van Uytfanghe, B. Das, F. MacKenzie, M.M. Patru, L.M. Thienpont, IFCC Committee for Standardization of Thyroid Function Tests (C-STFT), Monitoring the stability of the standardization status of FT4 and TSH assays by use of daily outpatient medians and flagging frequencies, Clin. Chim. Acta 467 (2017) 8–14.