OPTIMISATION OF SEQUENCE-BASED HLA TYPING USING GROUP- SPECIFIC PRIMERS IN A FULLY AUTOMATED SETTING.

Pedro Cano, Edward Guerrero, Dana Willis, Weicheng Zhao, Titus Barnes, Hemantkumar Patel, David Partlow, Marcelo Fernandez-Vina.
Laboratory Medicine, UT M. D. Anderson Cancer Center, Houston, TX, USA

Aim:

To develop an automated robotic method for sequencing using group-specific PCR amplifications, this is the ideal method to do high-resolution typing.

Methods:

Samples were typed at low resolution using Dynal SSO, Biotest SSO or One- Lambda SSO. Based on this preliminary typing, samples were amplified with the corresponding allele-group specific primers to separate both alleles in the diploid sample, and sequenced independently using Protrans sequence methodology.

Results:

The manipulation of a large set of samples each needing to be amplified with two out of an innumerable set of group-specific PCR primers, creates an insurmountable problem that we overcame by robotic automation. Setup by hand is impossible for a large number of samples. Such a robotic system was implemented with two Perkin-Elmer Janus robots, one for PCR amplification, and one for cycling. The LIS was set up to produce procedure-step files that were read by the robot and set up PCR reactions for each sample according to the low-resolution SSO typing. The procedure was validated with 300 samples with virtually no PCR failures and the quality of the sequences was invariably superior to those of heterozygous diploid samples.

Conclusions:

Liquid-handling robotic equipment has the ability to design procedures in which each sample is handled differently. This avoids handling errors and removes all the complexity of the operation. However, robotic automation requires strict control of reagents and special storage and labeling of DNA samples. Only with automatic robotic equipment it is possible to perform sequencing-base typing using group-specific PCR primers.

Designing a robot

A robot is not just a machine that performs complex mechanical operations. That may be impressive in itself, but hardly useful in the laboratory. The main question is not to move samples and reagents around, but also, and mainly, to create an efficient flow of information between the laboratory information systems (LIS) used to log in samples and order tests, the different pieces of equipment to process samples, the interpretation of results, and the reporting of results.

The key to integrate the flow of information when liquid handling robotic equipment is used is the creation of a virtual robot that mimics the actual robot. Only when computer procedures to make the virtual robot respond to the workload accumulated in the LIS have been created, can one approach the actual robot with something of substance. Trying to incorporate the complexity of the entire operation into programs built into the robot itself is bound to fail.

Given the current computer programming practices, the natural way to proceed is to use object-oriented programming tools, with so-called classes with properties and methods to create virtual robots. These robot classes are capable of performing two main types of methods: 1) setting locations and 2) creating liquid handling steps. These virtual robots can then create files with instructions transmitted to the actual robots to carry out the instructions and perform the procedure. Our experience indicates that the operational complexity must be removed from the programming environment of the actual robot. The price of not following this principle is facing unmanageable complexity. Whenever the flow of information involves multiple pieces of equipment, each with its own complex software, it is necessary to concentrate the complexity of the entire operation in a single place where the highest levels of control can be exercise, and most flexibility can be actualized. One cannot afford to let the complexity flow from system to system. As far as the flow of information is concerned, only simple instructions and data structures should be shared between systems concentrating the complexity in only one of them.

Reasons to automate laboratory procedures

- To reduce operator variability causing fluctuations in the quality of the results.
- To minimize sample handling errors.
- To reduce the complexity of lab procedures to computer commands.
- To allow the use of procedures which human operators cannot perform.

The reasons to automate laboratory procedures do not include the saving of costs in laboratory staffing.

Problems in automating laboratory procedures

Automation in the laboratory turns things that are complex into simple operations, and simple operations into complex procedures. Using different primer sets for different samples is a very taxing complex procedure for human operators, but it is straightforward for a liquid handling robotic system. Checking the availability of reagents is easy and simple for a human technologist, but a rather complex.

Laboratory technologists accustomed to manual procedures are prone to take for granted operations that require close attention and supervision, such as the management of reagents and the placement of samples and reagents in dedicated spots.

For a manual operator, having to start all over again is not only the most horrendous waste, but also the most inefficient way to solve problems. Having to start all over again is the ultimate sign of poor performance by a laboratory technologist. The inclination is always to try to save the work done by ingenious methods. On the other hand, in automation, repeating a run by beginning at the beginning is often the only way to solve a problem. If fact, it is the natural way to solve a problem. For this reason it is a necessity for the laboratory to turn things back to a previous state. This is an absolute necessity and it may require the developing of complex data management procedures in the laboratory information system. The ability to bring things back to the way they were before doing something is one of the keys to the successful implementation of laboratory automation. Automation requires a shift in perspectives. Introducing automation in the laboratory requires not only the design and implementation of the procedures themselves, but also the change of attitudes and work habits, which may turn out to be the most difficult component of the project.

A Protrans robot

Protrans sequencing allows the sequencing of alleles in a haploid state by separating allele pairs using specific PCR primers. Originally Protrans was designed to use a battery of PCR primers to see which gave amplified products. If two different allele-group specific primer sets showed the presence of an amplified product, the two alleles in a given sample could be effectively separated and sequenced individually avoiding in this way the two-base calls of sequence readings of diploid samples that often result in allele assignment ambiguities.

In so far as we perform low-resolution typing together with sequencing typing, we modified the above procedure to take advantage of the information obtained in low-resolution typing to separate alleles and sequence them independently. Instead of the battery of PCR primers that covers the spectrum of all the alleles of a particular locus, we used only the two specific primmer set targeting the two allele groups identified in low-resolution typing. Although this approach reduces the number of PCR reactions to those that are predicted to produce amplified products, it

introduces the complexity of having to process each sample in a different way. For any sizeable sample set manual processing becomes impossible and automation is required.
The implementation of this Protrans robot dramatically reduced the ambiguities typically encountered in sequencing typing. Not only did the heterozygous base calls disappeared, but also, and this is just as important, the quality of the sequences improved dramatically reducing the time it takes to review and edit them in order to do an accurate allele assignment.

Limitations of Protrans sequencing

Although most heterozygous samples can be processed to separate the two alleles, there is still a proportion of allele pairs that cannot be separated and must be processed using regular methods. Even though ambiguities due to heterozygous calls are removed, the ambiguities resulting from not sequencing the entire sequence still results in ambiguities. These areas outside the segment amplified by allele-group specific primers are typically irrelevant in determining the immune properties of the HLA molecule, and some of these ambiguities may not seriously affect clinical decisions. Nevertheless, when ambiguities in these areas involve null alleles they must be addressed and resolved.

Processing sequence files for analysis

Automating the processing of samples for amplification and sequencing is only the first part of the job; the remaining job is managing the files produced by the sequencer. For a single sample one may end with up to twelve sequencer files that must be analyzed in conjunction to reach a proper allele assignment. For instance, for a class I locus typing one has 1) exon 2 forward sequence for first allele, 2) exon 2 reverse sequence for first allele, 3) exon 2 forward sequence for second allele, 4) exon 2 reverse sequence for second allele, 5) exon 3 forward sequence for first allele, 6) exon 3 reverse sequence for first allele, 7) exon 3 forward sequence for second allele, 8) exon 3 reverse sequence for second allele, 9) exon 4 forward sequence for both alleles, 10) exon 4 reverse sequence for both alleles, 11) exon 1 forward sequence for both alleles, and 12) exon 1 reverse sequence for both alleles. After gaining enough experience it might be possible to do away with sequencing a haploid (1 allele) sequence in only one direction, but it is still a large number of sequences at hand to analyze simultaneously.
Current sequence analysis software, such as Conexio Assign, does not allow the analysis of an exon 4 sequence on its own, exon 2 and 3 sequences must also be integrated. This requires the combination of sequences that were produced independently. Therefore sequences must be analyzed separately and together. Certainly this process cannot be left in the hands of technologists and the processing must be automated by managing the names of the files coming out of the sequencer.

The final steps is integrating the Assign output files of the separate analysis for a given sample and locus. The final allele assignment is the intersect of the set of diploid combinations produced by the analysis of exon 4 data (combined with exon 2 and 3 data from both alleles sequenced independently), with the set of possible single-allele assignment produced by the analysis of exon 2 and 3 for one allele sequenced independently, and the corresponding set of single-allele assignments for the other allele. This is a simple logical operation that is implemented using sophisticated T-SQL queries in a relational database that holds the allele assignments for all the independent and combined tests.