SimArray: a user-friendly and user-configurable microarray design tool
Richard P Auburn*, Roslin R Russell, Bettina Fischer, Lisa A Meadows, Santiago Sevillano Matilla and Steven Russell

Address: Department of Genetics, University of Cambridge, Cambridge, UK
Email: Richard P Auburn* - r.auburn@gen.cam.ac.uk; Roslin R Russell - r.russell@gen.cam.ac.uk; Bettina Fischer - b.fischer@gen.cam.ac.uk; Lisa A Meadows - l.meadows@gen.cam.ac.uk; Santiago Sevillano Matilla - s.sevillano@gen.cam.ac.uk; Steven Russell - s.russell@gen.cam.ac.uk
* Corresponding author

Abstract

Background: Microarrays were first developed to assess gene expression but are now also used to map protein-binding sites and to assess allelic variation between individuals. Regardless of the intended application, efficient production and appropriate array design are key determinants of experimental success. Inefficient production can make larger-scale studies prohibitively expensive, whereas poor array design makes normalisation and data analysis problematic.

Results: We have developed a user-friendly tool, SimArray, which generates a randomised spot layout, computes a maximum meta-grid area, and estimates the print time, in response to user-specified design decisions. Selected parameters include: the number of probes to be printed; the microtitre plate format; the printing pin configuration, and the achievable spot density. SimArray is compatible with all current robotic spotters that employ 96-, 384- or 1536-well microtitre plates, and can be configured to reflect most production environments. Print time and maximum meta-grid area estimates facilitate evaluation of each array design for its suitability. Randomisation of the spot layout facilitates correction of systematic biases by normalisation.

Conclusion: SimArray is intended to help both established researchers and those new to the microarray field to develop microarray designs with randomised spot layouts that are compatible with their specific production environment. SimArray is an open-source program and is available from http://www.flychip.org.uk/SimArray/.

Background

The full utility of the spotted microarray format is clearly reflected in the range of its applications. Transcriptome arrays, containing cDNA, gDNA, or oligonucleotide probes, are used to measure differential gene expression [1-5]. Whole-genome arrays, typically composed of tiled gDNA or oligonucleotides [6], have been used to identify in vivo sites of protein-DNA interactions [7,8] or allelic variation [9,10]. Whilst these applications dominate, other formats, for example antibody arrays, facilitate analysis of protein and small-molecule analytes [11,12]. Thus, spotted microarrays enable high-throughput, cost-effective, and large-scale analysis of molecular interactions.

Robotic spotters deposit probes as an ordered array by repetition of a simple multi-step procedure [13-15]. First,
the print tool is positioned over the first batch of probes to be printed, arranged in 96-, 384- or 1536-well microtitre plates. Second, the spotting pins are filled by capillary action with probe material. This step is often called a source visit. Third, the probe material is deposited on chemically-modified glass slides [14,16]. Finally, the pins are cleaned, to prevent cross-contamination between subsequent spot depositions, before re-filling and printing the next batch of probes. The diversity of instrumentation, spotting pins, and reagents available, mean that procedures may be refined for optimal throughput, spot density, morphology, and consistency [16]. Whilst this facilitates production of high-quality arrays, it can also lead to significant differences between facilities with regard to the instrumentation, protocols, and reagents employed.

Robotic spotters are supplied with sophisticated software to convert operator inputs to the precise list of instructions needed by the arrayer, e.g., how often each source visit is to be printed, and at which spot location [13-15]. Most spotters, however, are not supplied with adequate array design tools. Operators are instead left to develop suitable spot layouts in an ad hoc fashion. This often leads to sub-optimal designs with spots positioned according to print order, thus juxtaposing replicates, when a non-sequential or randomised spot layout can help to control for confounding spatial effects [17,18]. For example, biases caused by inconsistent probe concentrations in the microtitre plates [18,19] and local variations in hybridisation or washing efficiency [20-23], also see Additional file 1. Whilst random noise can be overcome with simple replication and averaging, systematic biases must be specifically addressed by randomisation and normalisation [24-27].

There is therefore a need for a microarray design tool that can generate randomised source visits and, in the case of some instruments, permit the use of variable numbers of replicates, since current spot density constraints and the need for genome-wide coverage mean that replication is often limited to the normalisation controls. Exogenous or ‘spike’ controls, i.e., probes that are complementary to targets not present in the genome of interest, can be employed for this purpose [28,29]. Print time and maximum meta-grid area estimates enable users to evaluate the suitability of the array design.

We have addressed the current lack of such microarray design tools by developing SimArray, a user-friendly and user-configurable program that generates a randomised spot layout, computes the maximum meta-grid area, and estimates printing time, in response to user-defined design decisions. The user enters these parameters by running SimArray twice. The first run produces the source visit list that can be edited to include variable numbers of replicates, or for specific source visits to be omitted when plates are partially filled. The second run processes this source list to create the spot layout, maximum meta-grid area and estimated print time. User-configurable files mean that SimArray can be adapted to most production environments.

Implementation
SimArray was developed in Perl version 5.6.1 and 5.8.3, under both Windows and UNIX operating systems. SimArray can be run under Windows (after installing Perl; for example, ActivePerl [30]) and UNIX.

Before the first run
Download the three configuration files and an ‘index.sa’ file from the SimArray web site [31]. Configuration files that describe instrument-specific pin configurations and achievable spot densities have already been created. If a suitable file is not available, an existing one should be downloaded and edited. An example file for the user to record their specific print cycle times is also available for editing. The ‘index.sa’ file should then be updated, to record the locations and names of the configuration files. The configuration files will then only need to be re-edited if the printing environment is altered.

First run
Probe number: enter the number of microarray probes (or wells) to be printed.
Plate format: select an appropriate plate format.
Tools available: select an appropriate pin configuration.
Source visits: the source visit list is generated for editing.

Second run
Required spot density: SimArray counts the number of spots to be printed.
Pin type: select the spotting pin to be used.
Evaluate pin selection: SimArray evaluates whether the selected pin is compatible with the required spot density.
Compute spots_x and spots_y: SimArray computes and displays the sub-grid dimensions that fall between the target spot number and a user-specified upper limit, for the user to select.
Compute print time: select an appropriate print set-up, SimArray then calculates the estimated print time.
Summary report: SimArray generates a report containing a summary of the user’s responses, the randomised spot layout, an estimated print time, and the maximum meta-grid area.

After the second run
The randomised source visit map can be directly uploaded to instruments that accept either comma-, tab-, or space-separated values source files, or manually entered. Micro-
arrays can then be manufactured, with spots no longer positioned according to print order. Standard robotic spotter data tracking software can be used to record which probe is present at each spot location.

Results and Discussion

Pin configuration affects the number of source visits that must be performed and the maximum meta-grid area (Fig. 1). For these reasons, the user is required to enter the probe number (an integer) and to select a microtitre plate format (e.g., 96, 384 or 1536), before selecting a compatible pin configuration (Fig. 2). SimArray then prints a source visit list for the user to edit (Fig. 3). If the number of probes to be printed is not compatible with the selected pin configuration, SimArray will round up the source visit number to the nearest whole number, as robotic spotters can only print with a full complement of spotting pins. In such instances, the last source visit to be printed would include some empty wells. Users are, however, able to specify any number of replicates, for any number of source visits. Additionally, source visits can be omitted by setting the replicate number to zero. These features enable array designs with odd numbers of probes, variable num-

Figure 2

Example 'tool' user-configurable file. Print tools are composed of spotting pins that are arranged in a pre-defined plate-format-specific configuration (Fig. 1). Each user records the pin configurations that are available to them by adding their robotic spotter's list of options to the tool file. Pin configuration directly affects source visit number and maximum meta-grid area (Fig. 1). Tool files for the Qarray2 (Genetix), MicroGrid, and OmniGrid (Genomic Solutions) robotic spotters are available from the SimArray web site [31].

Figure 3

Example SimArray source visit list. The source list is generated after the first run of SimArray. Headers are marked with a hash at the beginning of the line and provide a summary of the specified design decisions. The replicate column can be edited to record how many times each source visit is to be printed. Source visits with zero replicates are ignored and omitted from the final array design. Other columns and the header should not be modified. An example file is available from the SimArray web site [31].
bers of replicates, and non-sequential source visits to be processed.

Maximum meta-grid area is calculated by simply multiplying the number of pins in each axis by the pin tool’s pre-defined pin pitch (Fig. 1). Consequently, SimArray does not take spot pitch into consideration and can overestimate meta-grid areas, especially for low-density arrays that are printed with reduced spot pitches. Since high-density arrays limit the scope for reducing spot pitch, we believe this is a reasonable approach because SimArray will be of most use when designing higher-density arrays. Additionally, most operators print microarrays with the spot pitch set to the near-maximum distance permissible to reduce the probability that neighbouring spots printed by the same pin will be merged together. Prediction of the maximum meta-grid area will at least allow users to decide whether it is possible for them to hybridise the array, e.g., when the hybridisation area is constrained by automated hybridisation stations.

The number of spots to be printed per sub-grid is calculated by counting the number of spots that are specified in the source visit list. This total is displayed and users are asked to select a suitable spotting pin (Fig. 4). The selected pin is evaluated and the script exits if the pin’s achievable spot number per sub-grid is incompatible with the required target spot number per sub-grid. Exiting the script at this stage, if a problem is found, removes the risk of downstream errors and provides an opportunity for the array design to be modified, or for a different spotting pin to be selected.

Sub-grid dimensions, i.e., the number of spots in the x and y-axis, which are compatible with the target spot number per sub-grid are then calculated, and users select an option from a list of compatible choices. To limit the length of this list, SimArray will only display sub-grid dimensions that are equal to or greater than the target spot number per sub-grid and less than a user specified limit. The upper limit is the target spot number per sub-grid, plus the user-specified ‘spot number margin’. SimArray prevents users from selecting grid dimensions that are incompatible with the spotting pins’ maximum achievable spot density. If, however, the selected sub-grid dimensions permit more than the required number of spots to be printed, additional spot locations are flagged as blanks by assigning them a source visit number of zero, i.e., not printed. SimArray will fail at this stage if there are no viable sub-grid dimensions between the minimum and maximum target. We therefore recommend using a ‘spot number margin’ of at least ten.

Print time is dependent on the number of source visits (Fig. 1), the number of slides to be printed, a range of (perhaps) user-defined options, e.g., pre-blotting, contact speed, etc., and the hardware itself, e.g., microtitre plate handling, x-y-z-axis motor speeds, pin (or tool) travel distances, etc. Additionally, wash conditions vary according to the production environment, i.e., spotting pin, spotting buffer, etc. Print time is therefore calculated after the user has selected the intended print setting from a list of available options (Fig. 5). The list of options includes the user defined ‘single print cycle duration’, i.e., the time it takes to perform a single print cycle from source visit to wash/dry cycle. This has to be determined empirically because it
is production environment and instrument dependent. Print time is calculated by simply multiplying the number of print cycles that need to be performed, by the time taken to perform each print cycle. Print time accuracy is therefore dependent on the user-specified single print cycle estimates.

Finally, a report containing the randomised source visit map along with a summary of the user's responses, an estimated print run time, and the estimated maximum meta-grid area is generated (Fig. 6). The user can specify a comma-, tab-, or space-separated source visit map with the command line keys -C, -T, or -S (default), respectively. The source visit map can then either be directly uploaded to instruments that accept source files in these formats, or manually entered. Microarrays can then be manufactured, with the spots and replicates positioned randomly, rather than according to their print order. Standard instrument data tracking software can be used to document what probe is present in each spot. New array designs are appended to the existing report to provide a full record of all array designs. Users are therefore able to perform multiple 'simulated print runs', with different configurations to compare the results, i.e., the estimated print times and maximum meta-grid areas. Each array design includes a date and time stamp.

The user-configurable files described above and in Figures 2, 3 and 4, maximise the utility of this tool because they
enable a range of production environments to be explicitly modelled. All SimArray configuration files contain a header, which includes a key to the file’s contents to help with this task. If required, additional comments can also be added because all lines marked with a hash at the start are automatically ignored when SimArray reads these files. However, column meaning and order is fixed and must be preserved. We aim to develop an on-line library of configuration files at the SimArray web site [31].

A fully worked simple example
For this worked example, we compare the performance of a MicroGrid II (Genomic Solutions) and a Qarray2 (Genetix) instrument, printing a 15 K probe library. The configuration files were edited to reflect the specific set of printing conditions for each robotic spotter. The example library consists of two probe types: transcript-specific probes and exogenous controls, along with some empty wells (Figure 6). The design requirement is to print single copies of the transcript-specific probes, and for every pin to print quadruplet spots for the exogenous control probes, randomising the distribution of elements on the array, whilst omitting the empty wells. The probes were arranged in the spotting plates, according to these design criterion (Figure 6).

SimArray was used to generate a randomised spot layout for each instrument, to assess which would be better suited to printing this library. The SimArray simulated print-runs indicated that the MicroGrid II spotter would take 56% longer to print microarrays according to the specified criterion (Figs 7 and 8). The estimated print times agreed with how long it would take to print the arrays, provided no manual intervention, e.g., refilling of wash solutions, etc., was required. The user can now either enter the randomised spot layout and print this microarray design with the Qarray2, or re-evaluate whether the print settings for the MicroGrid II were optimal.

SimArray also permits further simulations, allowing an evaluation of alternative pin configurations, replicate numbers, slide numbers and instrument configurations, subject to the availability of appropriately annotated configuration files. This further ensures that microarrays of an ideal design can be generated, whilst permitting each to be evaluated for its compatibility to the local production environment. Printing with different pin configurations, however, requires the spotting library itself to be redesigned, as spotting pins enter adjacent wells of the microtitre plate and the probes must be arranged accordingly (Figs 1 and 6). This suggests that spot layouts should be defined before the spotting probes are transferred to microtitre plates for printing.

**Conclusion**
We have developed a user-friendly microarray design tool, SimArray, which generates a randomised spot layout, computes a maximum meta-grid area, and an estimated print time, in response to user-specified design decisions. SimArray is of general utility for all users of robotic spotters and can be configured to suit individual production environments.

**Availability and requirements**
**Project name:** SimArray

**Project home page:** http://www.flychip.org.uk/SimArray/

**Operating system:** Windows and UNIX
Programming language: Perl version 5.6.1 (and more recent versions)

Other requirements: tool (Fig. 2), pins (Fig. 4), and time (Fig. 5) configuration files

Licence: free

Any restrictions to use by non-academics: none

List of abbreviations used

cDNA Complementary DNA
gDNA Genomic DNA

Meta-grid The sub-grids (and spots) that are printed by one print tool

Sub-grid The patch of spots that is printed by a single pin

Authors’ contributions

RRA suggested that array design could be automated. RPA wrote the source code, with technical guidance from RRR. RPA, BF, LAM, and SSM tested the code and validated performance. The manuscript was prepared by RPA. SR is the group leader, providing funding, critical assessment and general guidance. All authors read and approved the final manuscript.

Additional material

Additional File 1

This dataset includes heat maps for six different microarrays that were printed and hybridised by the UK Drosophila microarray facility. The heat maps demonstrate that spot location has a direct impact on the measured differential gene expression ratios and hence supports the argument for randomisation of the spot layout.

Click here for file: [http://www.biomedcentral.com/content/supplementary/1471-2105-7-102-S1.doc](http://www.biomedcentral.com/content/supplementary/1471-2105-7-102-S1.doc)

Acknowledgements

The authors would like to thank Gos Micklem, David Kreil, and the four anonymous referees for constructive criticism of the manuscript, and François Guiller for help with the web site. This work was supported by research grants from the BBSRC.

References

1. Schena M, Shalon D, Davis RW, Brown PO: Quantitative monitoring of gene expression patterns with a complementary DNA microarray. Science 1995, 270:467-470.
2. Schena M, Shalon D, Heller R, Chai A, Brown PO, Davis RW: Parallel human genome analysis: microarray-based expression monitoring of 1000 genes. Proc Natl Acad Sci U S A 1996, 93:10614-10619.
3. Hayward RE, Deris JL, Alfadhli S, Kaslow DC, Brown PO, Rashod PK: Shotgun DNA microarrays and stage-specific gene expression in Plasmodium falciparum malaria. Mol Microbiol 2000, 35:6-14.
4. Zaigler A, Schuster SC, Soppa J: Construction and usage of a one-fold-coverage shotgun DNA microarray to characterize the metabolism of the archaeon Haloferax volcanii. Mol Microbiol 2003, 48:1089-1105.
5. Relogio A, Schwager C, Richter A, Ansorge W, Valcarcel J: Optimization of oligonucleotide-based DNA microarrays. Nucleic Acids Res 2002, 30:e51.
6. Mockler TC, Ecker JR: Applications of DNA tiling arrays for whole-genome analysis. Genomics 2005, 81:1-15.
7. Ben R, Robert F, Wyrick JJ, Aparicio O, Jennings EG, Simon I, Zeitlinger J, Schreiber J, Hannett N, Kanin E, Volkert TL, Wilson CJ, Bell SP, Young RA: Genome-wide location and function of DNA binding proteins. Science 2000, 290:2306-2309.
8. Mukherjee S, Berger MF, Jona G, Wang XS, Muzzey D, Snyder M, Young RA, Bulyk ML: Rapid analysis of the DNA-binding specificities of transcription factors with DNA microarrays. Nat Genet 2004, 36:1331-1339.
9. Pinkel D, Segraves R, Sudar D, Clark S, Poole I, Kowbel D, Collins C, Zhang Q, Wang Y,窗口 L, Chi D, Wysoker A, Rooyenga-de Witte M, Giesler M, Feinberg A, Rusin L, Albertson DG: High resolution analysis of DNA copy number variation using comparative genomic hybridization to microarrays. Nat Genet 1998, 20:207-211.
10. Pinkel D, Albertson DG: Comparative genomic hybridization. Annu Rev Genomics Hum Genet 2005, 6:331-354.
11. Lueking A, Cahill DJ, Mullner S: Protein biochips: A new and versatile platform technology for molecular medicine. Drug Discov Today 2005, 10:789-794.
12. Chiosis G, Brodsky JL: Small molecule microarrays: from proteins to mammalian cells - are we there yet? Trends Biotechnol 2005, 23:271-274.
13. Cheung VG, Morley M, Aguilar F, Massimi A, Kucherlapati R, Childs G: Making and reading microarrays. Nat Genet 1999, 21:15-19.
14. Hegde P, Qi R, Abernathy K, Gay C, Dharap S, Gaspard R, Hughes JE, Snesrud E, Lee N, Quackenbush J: A concise guide to cDNA microarray analysis. Biotechniques 2000, 29:548-50, 552-4, 556 passim.
15. Affara NA: Resource and hardware options for microarray-based experimentation. Brief Funct Genomic Proteomic 2003, 2:7-20.
16. Auburn RP, Kreil DP, Meadows LA, Fischer B, Mazilla SS, Russell S: Robotic spotting of cDNA and oligonucleotide microarrays. Trends Biotechnol 2005, 23:374-379.
17. Wernisch L, Kendall SL, Soneji S, Wietzorrek A, Parish T, Hinds J, Butcher PD, Stoker NG: Analysis of whole-genome microarray replicates using mixed models. Bioinformatics 2003, 19:53-61.
18. Qian J, Kluger Y, Yu H, Gerstein M: Identification and correction of spurious spatial correlations in microarray data. Biotechniques 2003, 35:42-4, 46, 48.
19. Spruill SE, Lu J, Hardy S, Weir B: Assessing sources of variability in microarray gene expression data. Biotechniques 2002, 33:916-20, 922-3.
20. Yang YH, Buckley MJ, Dudoit S, Speed TP: Comparison of methods for image analysis on cDNA microarray data. J Comp Graph Stat 2002, 11:108-136.
21. Yang YH, Dudoit S, Liu P, Lin DM, Peng V, Ngai J, Speed TP: Normalization for cDNA microarray data: a robust composite method addressing single and multiple slide systematic variation. Nucleic Acids Res 2002, 30:e15.
22. Kreil DP, Russell RR: There is no silver bullet—a guide to low-level data transformation and normalization methods for microarray data. Brief Bioinform 2005, 6:86-97.
23. Futschik ME, Crompton T: OLIN: optimized normalization, visualization and quality testing of two-channel microarray data. Bioinformatics 2005, 21:1724-1726.
24. Kerr MK, Churchill GA: Statistical design and the analysis of gene expression microarray data. Genet Res 2001, 77:123-128.
25. Churchill GA: Fundamentals of experimental design for cDNA microarrays. Nat Genet 2002, 32 Suppl:490-495.
26. Brodsky L, Leonovich A, Shustman M, Feinstein E: Identification and handling of artificial gene expression profiles emerging in microarray hybridization experiments. Nucleic Acids Res 2004, 32:e46.
27. Le Meur N, Lamerault G, Bihouee A, Steenman M, Bedrine-Ferran H, Teusau R, Ramstein G, Leger JF: A dynamic, web-accessible...
resource to process raw microarray scan data into consolidated gene expression values: importance of replication. Nucleic Acids Res 2004, 32:5349-5358.

28. van Bakel H, Holstege FC. In control: systematic assessment of microarray performance. EMBO Rep 2004, 5:964-969.

29. Baker SC, Bauer SR, Beyer RP, Brenton JD, Bromley B, Burrill J, Cauton HC, Conley MP, Elespuru R, Fero M, Foy C, Fuscoe J, Gao X, Gerhold DL, Gilles P, Goodsaid F, Guo X, Hackett J, Hackett RD, Ikonomi P, Irizarry RA, Kawasaki ES, Kayser-Kranich T, Kerr K, Kiser G, Koch WH, Lee KY, Liu C, Liu ZL, Lucas A, Manohar CF, Miyada G, Modrusan Z, Parkes H, Puri RK, Reid L, Ryder TB, Salit M, Samaha RR, Scherf U, Sendera TJ, Setterquist RA, Shi L, Shippy R, Soriano JV, Wagar EA, Warrington JA, Williams M, Wilmer F, Wilson M, Wolber PK, Wu X, Zadro R: The External RNA Controls Consortium: a progress report. Nat Methods 2005, 2:731-734.

30. ActivePerl. [http://www.activestate.com/ActivePerl/].

31. SimArray. [http://www.flychip.org.uk/SimArray].

"BioMed Central will be the most significant development for disseminating the results of biomedical research in our lifetime."
Sir Paul Nurse, Cancer Research UK

Your research papers will be:
- available free of charge to the entire biomedical community
- peer reviewed and published immediately upon acceptance
- cited in PubMed and archived on PubMed Central
- yours — you keep the copyright

Submit your manuscript here:
http://www.biomedcentral.com/info/publishing_adv.asp