European Plant Phenotyping Network

EPPN 2020

D2.5: Statistical designs for single and multiple platform experiments

Emilie Millet (WU), Robert Horne (VSNi), Darren Murray (VSNi) & Fred van Eeuwijk (WU)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731013. This publication reflects only the view of the author, and the European Commission cannot be held responsible for any use which may be made of the information contained therein.

Document information

EU Project N°	731013	**Acronym**	EPPN2020
Full Title	European Plant Phenotyping Network 2020		
Project website	www.eppn2020.plant-phenotyping.eu		

Deliverable	**N°**	D2.5	**Title**	Statistical designs for single and multiple platform experiments.
Work Package	**N°**	WP2	**Title**	Design and analysis of phenotyping experiments across multiple platforms, scales of plant organisation, traits and management conditions

Date of delivery	**Contractual**		30/04/2020 (Month 36)	**Actual**	17/06/2020 (Month 38)
Dissemination level	X	**PU Public, fully open, e.g. web**			
		CO Confidential, restricted under conditions set out in Model Grant Agreement			
		CI Classified, information as referred to in Commission Decision 2001/844/EC.			

Authors (Partner)	WU, VSNi			
Responsible Author	**Name**	Fred van Eeuwijk	**Email**	fred.vaneeuwijk@wur.nl

Version log			
Issue Date	**Revision N°**	**Author**	**Change**
28/04/2020	0	Emilie Millet	first version
01/05/2020	1	Darren Murray	review by partner
04/05/2020	2	Fred van Eeuwijk	first review by WP leader
18/05/2020	3	Cloé Paul-Victor	Reviewed by project manager
15/06/2020	4	François Tardieu	Reviewed by Coordinator
16/06/2020	5	Emilie Millet	Final version

Executive Summary

Objectives:

Make an inventory of sources of variation at phenotyping platform that disturb comparisons of treatment or genotypic differences.

Develop guidelines, instructions, teaching material and software for choosing adequate statistical designs for experiments at phenotyping platforms.

Rationale:

The spatial and temporal heterogeneities at least as large in-door phenotyping platforms as in field experiments. We have adapted for indoor platforms techniques used in field experiments. Randomization guarantees unbiased estimates of treatment differences, where treatment differences include genotypic differences. An appropriate design contains sufficient replications of treatments to achieve satisfactory power for detecting treatment differences. Finally, an appropriate statistical design contains blocking structures to counteract environmental gradients. Blocking is a strategy to group experimental units into homogeneous groups of blocks, while within-platform environmental differences become part of between block variation and therefore will not add to the error for assessing treatment differences. Blocking allows more precise estimation of treatment differences.

To reach D2.5, WP2 made an EPPN2020 wide inventory of within platform environmental sources variation that could disturb the estimation of treatment differences. EPPN2020 platform partners were asked to describe their installations. Exchanges between WP2 and EPPN2020 platforms led to the formulation of research questions, treatment contrasts and blocking structures. Subsequently, statistical designs were proposed, discussed and used by partners.

Based on the platform inventory and on exchanges with platform leaders, guidelines were developed for choosing statistical designs for platform experiments and various design generation programs were evaluated for their suitability in helping platform partners generate their own designs. Furthermore, a new user-friendly design generator was created and presented in which an intuitive graphical user interface should guide users to an appropriate design. This introduction was supported by examples based on real experiments from EPPN2020 project partners. The software and supporting instruction material are available from the EPPN2020 intranet.

Main results:

An inventory of within-platform sources of variation for the EPPN2020 platforms and advices for suitable statistical designs to individual EPPN2020 partners. Guidelines and software for choosing statistical designs for platform experiments, including a specially developed web-based app with a user-friendly graphical interface.

Authors/Teams involved: Emilie Millet (WU), Robert Horne (VSNi), Darren Murray (VSNi) & Fred van Eeuwijk (WU)

Table of contents

1. INTRODUCTION

1.1. Aim of the EPPN2020 project

The EPPN2020 project aims at providing public and private European plant scientists access to a wide range of state-of-the-art plant phenotyping facilities, techniques and methods. It will help the plant community in progressing towards excellence across the whole phenotyping pipeline that includes sensor and imaging techniques, data analysis adjusting treatment contrasts for environmental conditions and placing interpretation in a biological context, data organization and storage, and analysis of series of experiments as well as meta-analyses of experiments.

EPPN2020 coordinates its activities with the future infrastructure EMPHASIS, listed in the ESFRI roadmap, and with national programs. EPPN2020 involves:

- access to 31 key installations at 15 infrastructures,
- a Work Package on sensors (WP1),
- a Work Package on data analysis (WP2),
- a Work Package about data management (WP3),
- networking activities for establishing cooperation and increasing integration between facilities both within and outside EPPN2020.

1.2. Scope and aim of the document

New phenotyping platforms require a reconsideration of classical experimental design and analysis techniques. Although this is not widely recognised, spatial and temporal heterogeneities in platform conditions are at least as large as in the field, if not larger, so it is essential that users choose appropriate experimental designs, models and analysis methods. WP2 addresses the lack of statistical design guidelines and analysis tools for data from phenotyping platforms. It has developed procedures for obtaining experimental designs with different tools for various types of platforms. It has also developed a new tool for design generation with a user-friendly interface.

This document describes procedures and software available to generate experimental designs for phenotyping platforms. To help platform users follow the basic rules of experimental design, we proposed (1) a three-step procedure for choosing a suitable design, (2) a presentation of available software programs, including one newly developed in this project, and (3) example procedures for each type of design identified in the EPPN2020 consortium, based on real experiments from partners.

2. PROTOCOL FOR EXPERIMENTAL DESIGNS ON PHENOTYPING PLATFORMS

We conducted a survey and visited several facilities from 2017 to 2019. Based on the outcome of the survey and visits we concluded that EPPN2020 platform managers have a good knowledge of the possible sources of error variation on their installations: they are able to describe the main trends likely to affect the plants and a majority of them have already quantified the temperature, water and/or light variability of their installations. Therefore, they are in a good position to choose a suitable design, provided that basic rules for choosing experimental design are followed. There are three basic principles of experimental design:

- *randomization*, to avoid confounding of treatment differences and (unknown) other differences between (groups of) units,
- *replication*, that allows quantifying the experimental variation between experimental units and increasing the precision of estimated effects,
- restriction of randomization, or *blocking*, which is a local control to reduce the experimental error by grouping experimental units into blocks of homogeneous units.

To help the platform user follow these rules, we proposed a standard three-step procedure before carrying out the experiment.

1. **Platform description**: the platform manager/user should first describe the installation in a statistically intelligible way. At this step, it is strongly advised to draw a map of the platform in the form of a two-dimensional grid and to add the existing trends likely to affect the experimental treatment(s) (**Fig.1**). This first step aims at helping the platform user to define the overall platform layout and position the blocks orthogonal to environmental gradients (measured or expected).

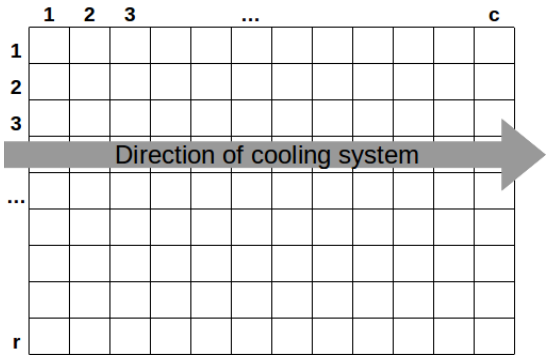


Figure 1. Scheme of a hypothetical platform described as a rectangular grid with *r* rows and *c* columns. Each square contains one experimental unit (*e.g.* one pot). The direction of the cooling system follows the grey arrow, indicating a possible temperature gradient.

2. **Experiment description**: defining the number of treatments applied to the experimental units (number of genotypes, number of water regimes, etc.), the number of replicates for each treatment, the experimental layout (if different from the installation layout, *e.g.* only a subset of experimental units at the platform is used, see **Fig.2**) and the block layout (if any). This step aims at facilitating the design generation by a software. Computational checks are performed, such as verifying that the number of treatments × replicates is compatible with the platform size and/or block sizes. As a remark we add that for D2.3 we aim at identifying a good blocking structure and randomization scheme, we did not address the question of how many replications to include as typically the size of the platform imposed severe limitations on the number

of replications that could be chosen and as a rule the number of replications was dictated by the available resources (platform size).

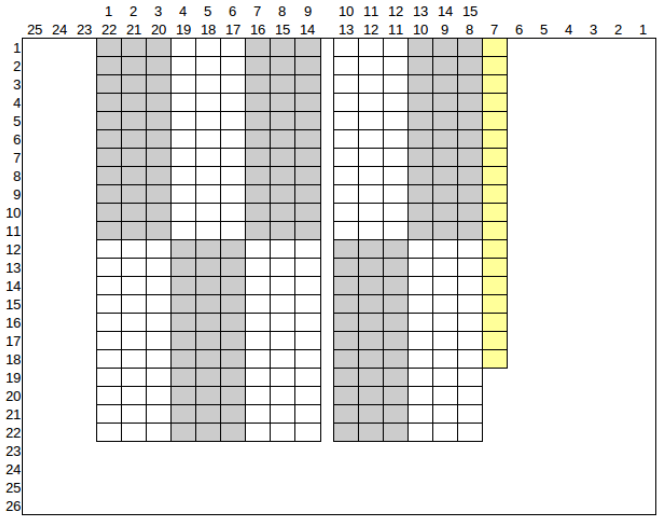


Figure 2. Map of the experiment at the 4PMI group (INRAe France). The experiment was carried out in a part of the greenhouse. Randomization restrictions (blocks) were defined in row and column direction. More details in section 4.2.

3. **Design specification, check and visualisation**: at this step, the platform users need to choose the software to generate the design. To assist them, we provide procedures with different software applications for types of design that are common for the platforms (*see below* **part 4.**).

Finally, the design should be checked to ensure that the treatments are adequately allocated. For example, (1) blocks in randomized complete block designs or resolvable blocks in incomplete block designs should contain all the treatments, (2) if a 2D latinization is used, rows and columns should not contain multiple occasions of a particular pair of treatments, (3) if a highly replicated treatment (*e.g.* check genotype) is added in an augmented design on top of a standard design for the treatments (genotypes) of interest, this augmented design needs to be checked as well. In general, concurrences of treatments need to be verified to see whether they follow the assignment rules of the chosen design.

Using a block structure or re-allocating the plants during the experiment?

A complication in relation to choosing a suitable design for phenotyping experiments is that at certain types of installations plants change position during the experiment. Ideally, at every step of the experiment a suitable randomization should be chosen. However, in practice the position of the plants on a platform after an initial round of observations is determined by the mechanical restrictions of the installation that allocates the plants to positions on the platform. When plants are not allocated following a randomization scheme dictated by a statistical design, the subsequent statistical analysis is not obvious and may lead to invalid conclusions.

In the proposed protocols below, we restrict ourselves to recommendations for experiments with fixed plant positions over the duration of the experiment. The hypothetical advantages of the re-allocation option strongly depend on the platform and the type of designs (Brien et al., 2013; Hartung et al., 2019) and it would require further testing at each installation to find out what can be achieved by combining design and reallocation. Furthermore, no available software can currently automatically generate designs for multiply to be reallocated plants.

3. TOOLS FOR EXPERIMENTAL DESIGN

3.1. Programs to generate experimental design

Several computer programs are available to generate experimental designs. When considering simple designs, such as randomized complete blocks or small split-plot designs with few experimental units, the R package **agricolae** provides experimental designs for agricultural experiments (de Mendiburu, 2020). Another option is the **Genstat** software (https://www.vsni.co.uk/software/genstat).

When many experimental units are considered, with different levels of replication and/or latinization, then the following programs are advised:

- **DiGGer** (, 2019) is an R package (Team, 2019) that generates designs based on the Reactive Tabu Search (RTS) algorithm (Coombes, 2002). *Details of the program can be found in the package vignette.*
- **OD** () is an R package that generates optimal experimental designs under the linear mixed model (Butler, 2013). *Details of the program can be found in the package vignette.*
- **blocksdesign** () is an R package that provides functionality for the construction of nested or crossed block designs for general linear model treatment designs (Edmondson, 2020). *Details of the program can be found in the package vignette.*
- The **OPTEX** procedure of SAS performs a numerical search for an efficient design based on the D-optimality criterion (Atkinson *et al.*, 2007). *Details on the use of OPTEX can be found in (Piepho et al., 2015).*
- The windows software **CycDesigN 7.0** () for optimal or near-optimal experimental designs (Williams, 1995).

The key step of these programs is to swap the treatments between experimental units based on an initial design that is improved during the swapping. The swapping continues until an optimization criterion is reached, for example when a minimum A-efficiency (related to the average variance of pairwise comparisons of treatments) is obtained. The criterion depends on the software and the type of design, and is automatically calculated at each swap. The swapping is often restricted, either by the block structure (if any), or by the assumed correlation function between experimental units (*e.g.* in DiGGer), or by the model initially chosen (*e.g.* in OD).

3.2. A new tool for experimental design: the design generator

In the project, a web-based application has been developed to help platform users generate an experimental design. The design generator app has been developed by Robert Horne and Darren Murray (**Fig.3**, VSNi) and a first release has been made available to partners in the project:

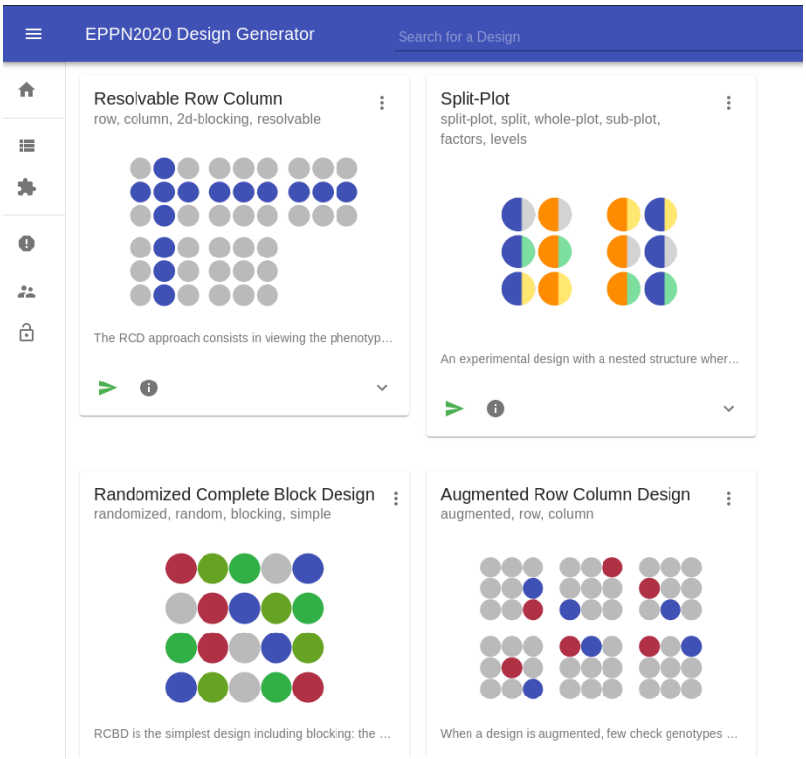


Figure 3: Homepage of the design generator website.

The design generator includes the possibility to create a facility based on a user defined platform. This stores features about the platform such as the dimensions that can then be used to guide the user in the generation of the design. The app provides a range of designs and uses visualizations to assist the users understand and create their designs. It maintains a history of generated designs where each design can be visualised in a 2D map and can be downloaded within a csv file. In the current version there are five types of design available: Randomized Complete Block designs, Resolvable Row-Column designs (Piepho *et al.*, 2015), partially replicated designs (p-rep, (Cullis *et al.*, 2006), Augmented Row-Column designs (Piepho & Williams, 2016) and Split-Plot designs (Welham *et al.*, 2014). The computation of the designs is being powered by CycDesigN - and Genstat (https://www.vsni.co.uk/software/genstat). The app is hosted on AWS and access is provided for free to EPPN2020 partners for the duration of the project. A tutorial has been made available for all partners. After completion of the project, the app will be made available to EPPN2020 partners at a reduced rate.

The advantage of the design generator is that it provides a user-friendly interface while ensuring robust design generation. This enables the platform users to easily generate designs without advanced statistical or programming knowledge. The visualization of a simplified scheme of the design at each step of the design generation (creation of the facility, design specification and final check) allows the user to understand the layout for each type of design in relation to their platform. It also makes it an appropriate tool for teaching courses about the design and analysis of phenotyping experiments.

4. ILLUSTRATIONS OF PROTOCOL FOR VARIOUS EXPERIMENTAL DESIGNS

4.1. Randomized Complete Block Designs

The Randomized Complete Block Design (RCBD) is the simplest design including blocking and the randomisation of treatments takes place inside the blocks. Blocks are randomized as a whole as well. The number of blocks is equal to the number of replications; therefore, each block contains each treatment exactly once.

Example of RCBD: GrowScreen-Rhizo at Jülich Plant Phenotyping Center, FZJ (provided by Kerstin Nagel and Fabio Fiorani) – *Online support at:*

Briefly, the installation *GrowScreen-Rhizo* system is in an experimental greenhouse and consists of two lines of mounting frames in which rhizo boxes are inserted. Between both lines of rhizo boxes a cabinet for imaging is moving automatically on a linear axis. This can be viewed as a grid of 36 rows by 2 columns, where one column is one line of rhizo boxes and one row contains two rhizo boxes.
The example procedure contains an experiment from the joint-platform experiment (WP1 and WP2 from the project). This experiment was carried out on the entire platform (36 × 2), including the 72 rhizo boxes with one plant per rhizoboxe. Nine genotypes (treatments) were tested with 8 replications each. Eight complete blocks of nine units were made, for a total of 72 plants (experimental units) (**Fig.4**).

→ *The design generator web-based tool was used to create "Randomized Complete Block" part. This generated the randomization of number 1 to 9 per block: The output was then formatted using R, to randomly assign the numbers to genotype names and the block numbers to the actual blocks in the platform.*

Figure 4: Design visualisation with the R package *desplot*. Four genotypes are highlighted in colour, the rest are white.

4.2. Row-Column Designs

When many treatments are used (large number of genotypes), the randomisation in the RCBD sometimes leads to undesirable arrangements of treatments (*e.g.* two pairs of treatments occurring close to each other).

In most phenotyping installations, control and/or correction of micro-climatic conditions is essential. To this end, we can use two-way blocking strategies, like the Row-Column Design (RCD), where the blocks are best chosen following prior knowledge of the structure and magnitude of existing noise variation. The RCD approach consists in viewing the phenotyping experiment as a rectangular grid on a set of row and column coordinates ($r \times c$). Row and column blocks can be defined as incomplete blocks in two directions. To ensure that treatments will be as evenly spread as possible over columns and/or rows, it is possible, and sometimes

desirable, to use a so-called resolvable row-column design (Piepho *et al.,* 2015). In this case, complete (resolvable) blocks, i.e., with all treatments, are defined that encompass contiguous subsets of rows and columns). The balancing of treatments across rows and columns is called *latinization* and ensure that pairs of treatments (genotypes) are evenly spread in the row and column direction.

Example of resolvable RCD: 4PMI at INRAe Dijon (provided by Christian Jeudy and Christophe Salon) – *Online support at:*

The installation consists of four connected greenhouses with pots, Rhizotubes, on conveyor belts. This installation contains a cross-platform experiment for the EPPN2020 project. Thirty-three genotypes will be characterized for their variability in root system in 10 blocks. The experiment was carried out on a subset of greenhouse number four (capacity of 26 rows × 25 columns), defined by a 22 rows × 15 columns grid. Blocks are laid out in two directions, each block (full replicate or resolvable block) contains 11 rows × 3 columns, with a row representing an incomplete 1 × 3 block, and a column representing an incomplete 11 × 1 block. The genotypes were latinized in two directions along so-called long rows = row (incomplete) blocks of 1 × 15 and long columns = (incomplete) blocks of 22 × 1 (**Fig.5**).

→ *The online support includes examples of instructions for CycDesigN 7.0, the R package DiGGer and the design generator app.*

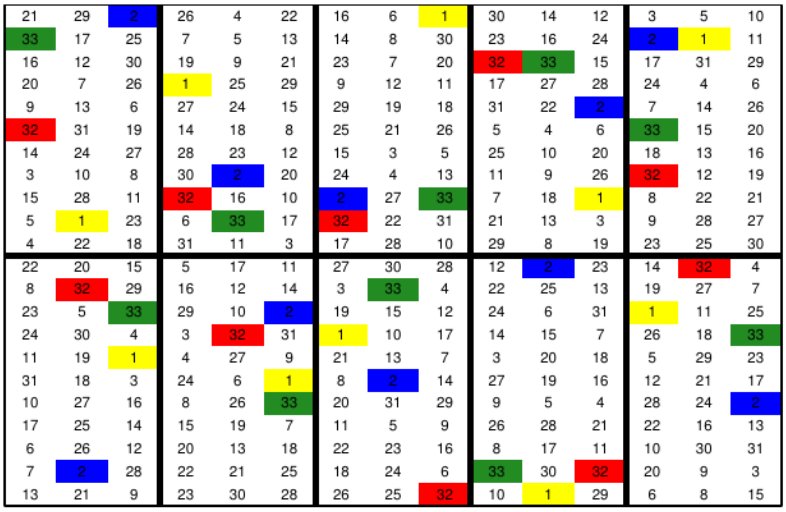


Figure 5: Design visualisation of the DiGGer ouptut with the R package *desplot*. For illustration, four genotypes are highlighted in colour, the rest is white.

4.3. Augmented p-rep designs

In plant genetics, platform users try to maximize the number of genotypes they test. In this case, to be able to estimate the error variance and adjust for the global and local trends, one strategy is to partially replicate only a small number of genotypes of interest: the p-rep design (partially replicated design). A p-rep design can be generated using any block design for the replicated entries, usually about 25-30% of them, and then augmenting it with the un-replicated entries by allocating them to the free plots in completely randomized order (Cullis *et al.,* 2006). Another strategy is the Augmented Row-Col design (Piepho & Williams, 2016). In this design a row-column randomized complete block design for a few check varieties is combined with un-replicated genotypes that are assigned to free positions inside the blocks.

Example of Augmented design: NaPPI, University of Helsinki (provided by Kristiina Himanen and Mirko Pavicic) – *Online support at:*

The UHEL NaPPI (National Plant Phenotyping Infrastructure) large plant facility is located at the University of Helsinki (Finland,). The installation is in a greenhouse and consists of pots on conveyor belts surrounded by different cabins. Single pot trays are distributed on 9 lines (rows) of 30 trays for a total of 270 trays with single pots and one plant per pot.

A segregating population of 192 BC_4F_2 *Brassica rapa* plants was characterized on this platform (segregating for a dwarf gene). Their genotypes were unknown before the experiment: sampling was done after emergence. One parent genotype was used as control with extra seeds. The experiment was carried out on part of the platform: 8 row × 30 columns with 193 genotypes (treatments): one replicate for the 192 genotypes of the segregating population and 48 replicates of the control parent (check genotype) for a total of 240 plants (experimental units).

The assignment layout for the plants of the check genotype (gray rectangles) is shown for two possible design options: (1) systematic design for check genotype occurring at every 5 pots on a row and diagonal arrangement across rows (2) row-column design in a series of 2 rows × 3 columns incomplete blocks. In the second option, plants of the check genotypes were latinized in two directions (**Fig.6**).

→ *The online support includes instructions for the R package DiGGer, CycDesigN 7.0, and the design generator app.*

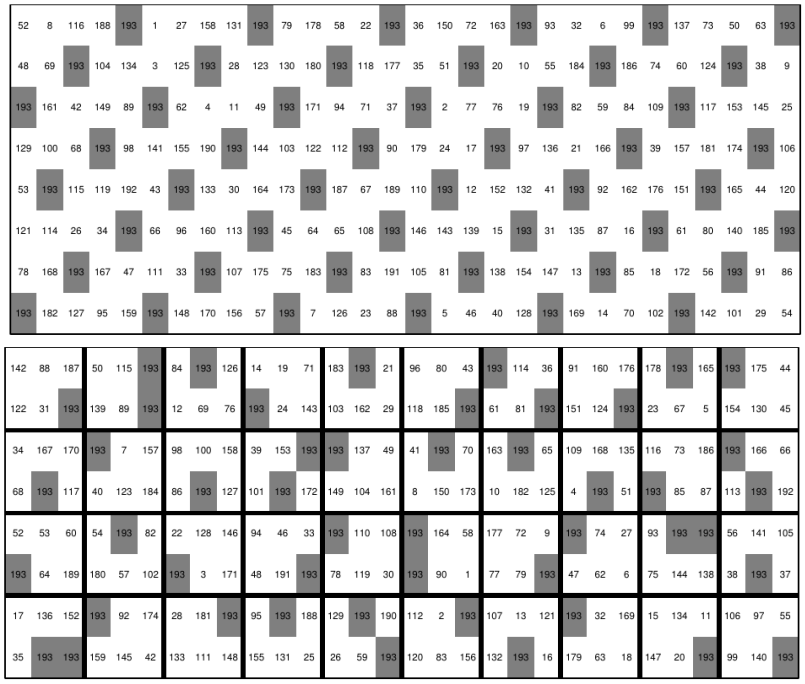


Figure 6: Design visualisation of the DiGGer outputs with the R package *desplot*. Plants of the check genotype are highlighted in grey. Top, systematic design; bottom, row column design with blocks (black lines).

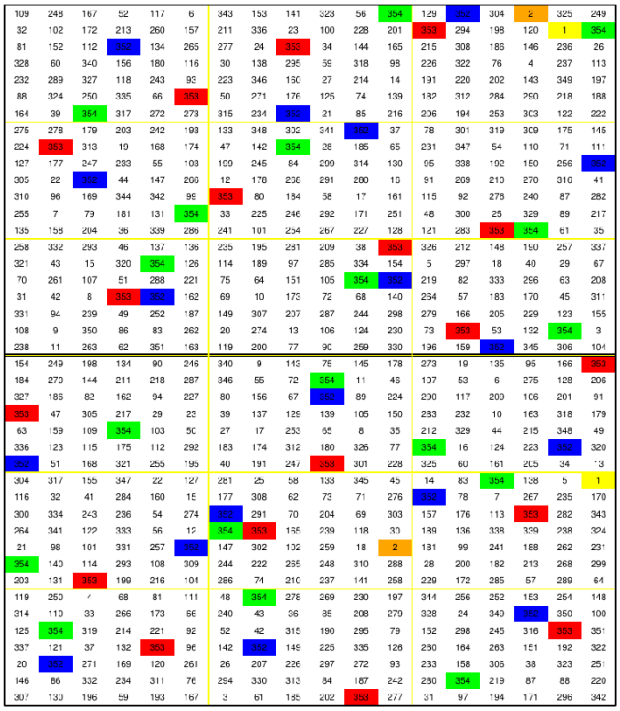
4.4. Augmented Resolvable Row Column Designs

In many large platforms, when the number of treatments is large, it is impossible to define complete or resolvable blocks with the full set of treatments that are sufficiently homogeneous even after correction for row and column incomplete blocks, i.e. the complete blocks are so large that there could be spatial trend(s) inside the block that are not covered well by the row and column incomplete blocks. In this case, the design can include one or few highly replicated genotypes (usually well-known reference varieties) that will help characterize the spatial

variability. The design combines an initial resolvable row-column design and an augmented design on top of the original row-column design. The basis is a resolvable row column design to which one or more highly repeated checks are added. The checks occur in incomplete blocks that cover a small number of rows and columns and in which the candidate genotypes are latinized in row and column direction, i.e., across 'long' rows and 'long' columns.

Example of an augmented resolvable RC design: ETHZ (provided by Andreas Hund) – *Online support at:*

The platform is the Field Phenotyping Platform (FIP) at the Swiss Federal Institute of Technology in Zürich (ETHZ, Switzerland). A rope suspended carrier system holds multiple sensors which can be positioned over individual plots or plants.



The experiment consisted of two complete blocks with 354 genotypes: 351 test genotypes, and three check varieties replicated 9 times. The field plot layout is 42 rows by 18 columns. It was first divided in two blocks of 21 rows by 18 columns for the test genotypes. In addition, the field was simultaneously partitioned into 9 blocks of 7 rows by 6 columns for the check genotypes (**Fig.7**). → *The online support includes explanations for the R package DiGGer, CycDesigN 7.0, and the design generator app.*

Figure 7: Design visualisation of the DiGGer output with the R package *desplot*. Two test genotypes are highlighted in orange and yellow, the three checks are in green, red and blue, and the remaining genotypes are in white. Black lines for complete blocks for test genotypes, yellow lines for augmented blocks for check genotypes.

Example of an augmented resolvable RC design: rootPhAir, UCLouvain (provided by Xavier Draye) – *Online support available at:*

The RootPhair platform at the UCLouvain (Belgium) consists of two aeroponic tanks of 495 plants located in the same greenhouse. Plants are hold on 5 plants strips with 99 strips per tank. Sprinklers are placed at the bottom of the tanks and spray nutrient solution. In each tank, strips of 5 plants are constantly moving following the orange arrows (**Fig.8**).

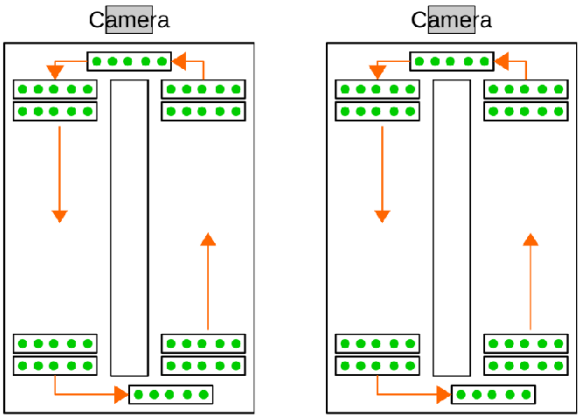


Figure 8: Schematic top view of the two tanks. Strips contain five plants (five green dots) and they move following the orange arrows.

We could consider a tank as 49 rows by 10 columns rectangle. However, after the first strips rotation step, the rows are broken up in the middle (between two strips, **Fig.8**). The 99 strips are 99 rows that stay intact during rotation (strip *n* is always between strip *n-1* and *n+1*) and each strip has 5 positions that travel on their own rotational paths. Each tank can be seen as a 99 × 5 row-column structure.

In an experiment, a panel of maize genotypes was characterized for root traits. The collection to screen comprises 146 test genotypes, with 6 seeds per genotypes. There are two check genotypes, a reference variety, with limited seed availability (40 seeds). Another check variety, a commercial variety with ample seed fills up the remaining positions. Per tank, we created three complete blocks, each containing the 146 test genotypes. We added per block 6 or 7 plants of the reference variety and 12 or 13 plants of the commercial variety. The reference and commercial varieties were assigned to positions across the full tank following a design with incomplete blocks in long columns (99 × 1) and incomplete blocks in columns of (11 × 1) (**Fig.9** and see the online support for more details).

→ *The online support includes explanations for the R package DiGGer, CycDesigN 7.0, and the design generator app.*

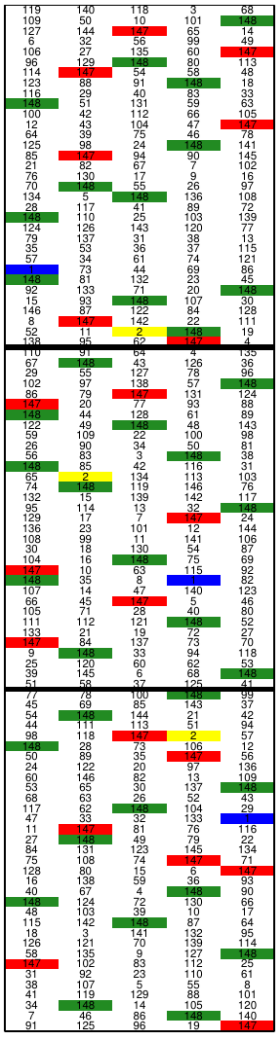


Figure 9: Design visualisation of the DiGGer output with the R package *desplot*. Two genotypes are highlighted in blue and yellow, the two checks are in green and red, and the rest is white. Replicate block layout is highlighted with black lines.

4.5. Split-plot design

A split-plot design is used when at least two treatments are simultaneously tested in one experiment at differently sized experimental plots (Welham *et al.,* 2014). The levels of one treatment factor are applied to large experimental units or whole plots (e.g. a fertilizer applied to many plants simultaneously) while the levels of the other treatment are applied to smaller

units, i.e., sub plots, within the whole plots (e.g. a panel of genotypes where each sub plot contains one or more plants of a genotype). For example, we consider a field experiment with six levels of nitrogen (N1 to N6) and four rice variety (V1 to V4). The randomization will be done in two stages (**Fig.10**): in the first step, the nitrogen levels are laid out in a randomized complete block design (full replicate) and each plot is called a whole plot or a main plot. In the second step, the varieties are randomized to sub plots within the main plots.

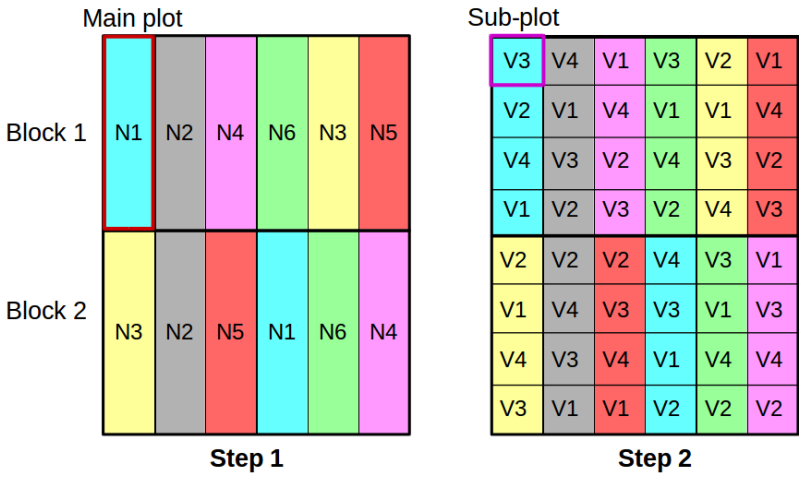


Figure 10: Two-step randomization of a split-plot design including six levels of nitrogen (N) and four variety (V) (example adapted from Gomez & Gomez, 1984).

Example of a split-plot design: the Slovak PlantScreen Phenotyping Unit (SPPU, provided by Marek Zivcak and Oksana Sytar) – *Online support at:*

The Slovak PlantScreen Phenotyping Unit (SPPU) at Slovak University of Agriculture (SUA) in Nitra (Slovakia) consists in a large growth chamber. Pots are placed on conveyor belts with six lines of 18 pots (108 plants).

The online protocol contains the experimental design of a joint-platform experiment from the EPPN2020 project. It was carried out on the entire platform, 18 row × 6 columns. Nine genotypes were tested in two different soil substrates (one management with two levels = whole plots), with 6 replications of the genotype per substrate for a total of 108 plants. This is a split-plot design with six complete blocks (**Fig.11A**, yellow rectangle), and two main or whole plots per block with a main plot size of 9×1 positions, (**Fig.11A**, red rectangle) and nine sub-plots per main plot (**Fig.11B**, blue rectangle).

→ *The online protocol includes explanations for the R package DiGGer, CycDesigN 7.0, and the design generator app.*

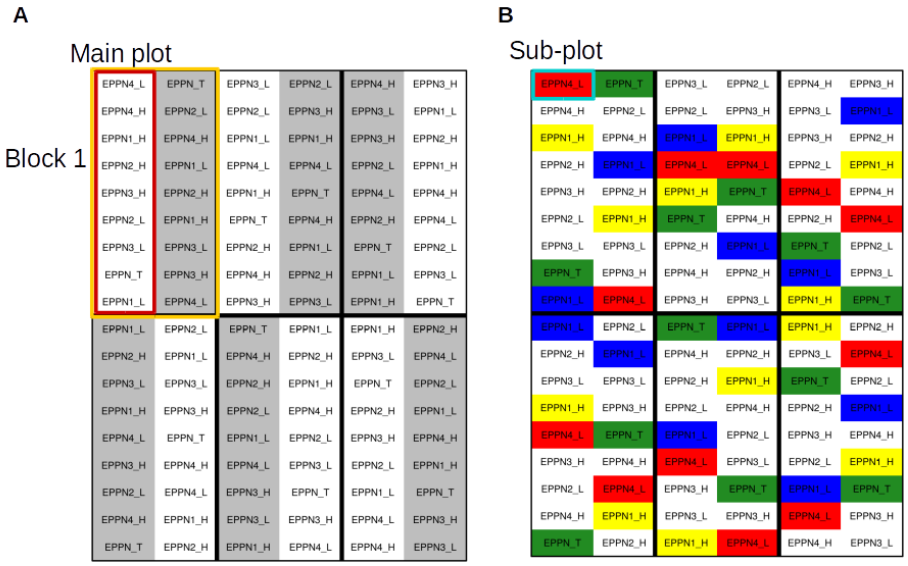


Figure 11: Design visualisation of the DiGGer outputs with the R package *desplot*. A, blocks and main plots layout with the two levels of substrates in grey and white. B, genotypes allocation to the sub-plots with four genotypes in red, green, blue and yellow, the others in white. Main plots are delimited with black lines.

4.6. Designing multiple experiments

An experiment at a phenotyping platform can be defined by set of available experimental units (pots, plants, positions) on which observations are made during a particular period, the run time of the experiment. If possible, an individual experiment contains the full set of treatments and their replicates. In some cases, the number of treatments × replicates is larger than the number of experimental units that is available during an experimental run. Several experimental runs with incomplete sets of treatments and / or replicates are consequently required. One could say that several experimental runs together then form an experiment, with each individual run containing part of the experiment, or, the experimental runs are nested within the overall experiment.

For experiments that span multiple runs on a platform, special care needs to be exercised to arrive at an appropriate design. Such experiments can occur when a large collection of genotypes (diversity panel, segregating population) needs to be evaluated in combination with one or more management factors (nitrogen, water, plant density, etc.). Designing the full experiment should enable (1) estimating the genotypic variability per management regime by including the a sufficiently large set of genotypes with some replication (p-rep or augmented design), and (2) estimating the management effect by replicating the management regimes (split-plot design).

Example of split plot experiment in randomized complete blocks for whole plot management regimes and p-rep assignment of genotypes within management regimes combined with augmented row column design at sub plot level for replicated genotypes: Aberystwyth University (provided by John Doonan and Gina Garzo) – *Online support at:*

The small plant platform is located at the Aberystwyth University (UK, https://www.plant-phenomics.ac.uk/index.php/resources/psi-system/). The system is in a greenhouse and can hold up to 2000 pots and takes RGB and chlorophyll fluorescence imaging. The platform layout is 80 rows by 25 columns, divided into 100 trays of 4 rows by 5 columns (20 plants).

A diversity panel of 420 *Arabidopsis thaliana* genotypes was characterized on this platform with two management factors: two levels of nitrogen (+N or -N) combined with two plant densities (1P or 4P), leading to four management combinations or regimes (+N1P, +N4P, -N1P, -N4P). Three experimental runs, full replicates, were carried out: with each run including four whole plots of each 20 rows by 25 columns or 4 x 5 crates corresponding to management regimes (nitrogen x density). The four management regimes were included in a single run on the platform to avoid confounding experiment and treatment (genotype by management) effects. In each management regime a full set of genotypes was included for a precise estimation of the genotypic variability. Within a whole plot corresponding to management regime, sub plots were assigned to genotypes by following a partially replicated design (p-rep): 400 genotypes were un-replicated, and 20 genotypes were replicated 5 times (**Fig.12**). For the replicated genotypes an augmented design was superimposed of tray blocks (4×5 plants) to better spread them (see more details in the online support).

→ *The online support includes explanations for the R package DiGGer, CycDesigN 7.0, and the design generator app.*

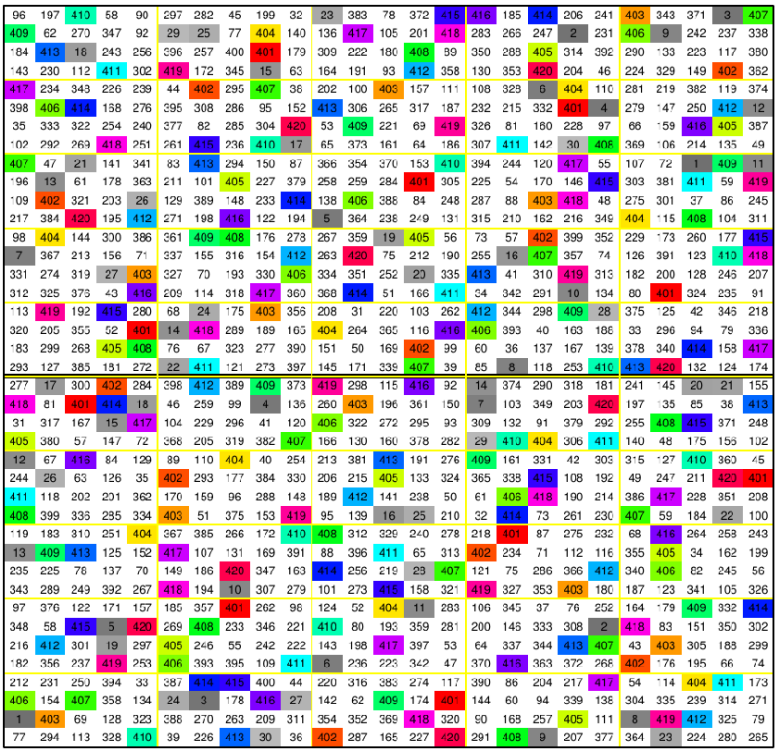


Figure 12: Design visualisation of the DiGGer outputs with the R package *desplot*. Within one experimental run (block), two management regimes, out of four are shown, representing whole plots for nitrogen by density regime. The black line separates these whole plots or two management regimes. Twenty replicated genotypes are shown in various colours, 30 un-replicated genotypes are shown in grey for illustration, while the remaining un-replicated genotypes appear in white. The trays are separated by yellow lines.

Experimental designs for multiple runs can in principle be constructed without too much problems by generalizing the rules for the design construction of experiments that can be fitted within single experimental runs. Still, automating the construction of designs for multi-run experiments is not straightforward and is difficult to include as an option in design construction software. The most simple case for multi-run experiments was illustrated above. In that case,

each experimental run coincided with a block and design construction can proceed block by block, where the earlier described design generation software is still useful.

5. FINAL COMMENTS

The inventory of installation layouts and experimental designs as performed in WP2 showed that most installation managers know the sources of error variation on their installation. Some of them use information on direction and magnitude of error trends to improve their experimental designs but there was a lack of tools and procedures to facilitate the design generation. Examples provided by the partners and the design of a joint-platform experiment were used to define a standard protocol for experimental design in phenotyping platforms together with illustrations for typical cases. These examples will be publicly available, and the webpage will be enriched with more examples. It also served the development of a new tool, the design generator by VSNi, which facilitate the design generation with a user-friendly application.

Further developments of the application tool are expected with the partners' feedback. In WP1 of EPPN2020, installation managers are asked to quantify the environmental error variability by mapping environmental gradients on the coordinates of their platform. This information could also be included in the design generator (input from the user) to help defining complete and incomplete blocks.

References

Atkinson A, Donev A, Tobias R. 2007. *Optimum Experimental Designs, with SAS.*

Brien CJ, Berger B, Rabie H, Tester M. 2013. Accounting for variation in designing greenhouse experiments with special reference to greenhouses containing plants on conveyor systems. *Plant Methods* **9**(1): 5.

Butler DG. 2013. *On the Optimal Design of Experiments Under the Linear Mixed Model.* PhD, The University of Queensland.

Coombes NE. 2002. *The Reactive Tabu Search for Efficient Correlated Experimental Designs.* PhD, John Moores University Liverpool

Cullis BR, Smith AB, Coombes NE. 2006. On the design of early generation variety trials with correlated data. *Journal of Agricultural, Biological, and Environmental Statistics* **11**: 381-393.

de Mendiburu F. 2020. agricolae: Statistical Procedures for Agricultural Research.

Edmondson RN. 2020. blocksdesign: Nested and Crossed Block Designs for Factorial, Fractional Factorial and Unstructured Treatment Sets.

Gomez KA, Gomez AA. 1984. *Statistical Procedures for Agricultural Research*. New York: Wiley.

Hartung J, Wagener J, Ruser R, Piepho HP. 2019. Blocking and re-arrangement of pots in greenhouse experiments: which approach is more effective? *Plant Methods* **15**: 143.

Piepho H-P, Williams ER. 2016. Augmented Row–Column Designs for a Small Number of Checks. *Agronomy Journal* **108**(6).

Piepho H-P, Williams ER, Michel V. 2015. Beyond Latin Squares: A Brief Tour of Row-Column Designs. *Agronomy Journal* **107**(6).

Team RC. 2019. R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing.

Welham S, Gezan S, J Clark S, Mead A. 2014. *Statistical Methods in Biology: Design and Analysis of Experiments and Regression*.

Williams ER, John, J. A. 1995. *Cyclic and computer generated designs* London: Chapman & Hall

Glossary

CRD – Completely Randomized Design
EPPN2020: European Plant Phenotyping Network - 2020
p-rep design – partially replicated design
RCBD – Randomize Complete Block Design
RCD - Row-Colum Design
TNA – Trans-National Access
WP – Work Package