Summer 2014

A new integrated design framework for the facility layout problem

Kyle F. Thomas
Purdue University

Graduate School ETD Form 9
(Revised 01/14)

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By Kyle F. Thomas

Entitled
A NEW INTEGRATED DESIGN FRAMEWORK FOR THE FACILITY LAYOUT PROBLEM

For the degree of Master of Science in Industrial Engineering

Is approved by the final examining committee:

Patrick Brunese

Jose Tanchoco

Steven Landry

To the best of my knowledge and as understood by the student in the *Thesis/Dissertation Agreement. Publication Delay, and Certification/Disclaimer (Graduate School Form 32)*, this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

Approved by Major Professor(s): Patrick Brunese, Jose Tanchoco

Approved by: Abhijit Deshmukh 06/25/2014

Head of the Department Graduate Program Date

A NEW INTEGRATED DESIGN FRAMEWORK FOR THE FACILITY LAYOUT PROBLEM

A Thesis

Submitted to the Faculty

of

Purdue University

by

Kyle F. Thomas

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Industrial Engineering

August 2014

Purdue University

West Lafayette, Indiana

For my family, teachers, friends, and others who have inspired me to make the most of the opportunities I have been given.

ACKNOWLEDGEMENTS

Thank you to Dr. Patrick Brunese for all of the time and effort he has put into helping me through not only this process, but the entire time I have spent on Purdue's campus, even before I was officially a student. He has challenged me to learn, shared a multitude of ideas, inspired me to try new things, and been an unbelievable mentor.

Thank you to the faculty and staff of Purdue University for providing me with an outstanding education, and a positive experience while I was here on campus. Especially professors Steven Landry, Jose Tanchoco, Nagabhushana Prabhu, Barrett Caldwell, Bruce Schmeiser, Juan Wachs, Hong Wan, Kelly Blanchard, Srinivasan Chadrasekar, Abhijit Deshmukh, and Cheryl Barnhart.

I would also like to thank my friend Alden Black for his time and help getting many of my optimization programs to run correctly.

TABLE OF CONTENTS

Page

ABSTRACT

Thomas, Kyle F. M.S.I.E., Purdue University, August 2014. A New Integrated Design Framework for the Facility Layout Problem. Major Professors: Patrick Brunese and Jose Tanchoco.

This thesis proposes a new integrated design framework for solving facility layout problems (FLP). The most popular existing framework, Muther's Systematic Layout Planning (SLP) does not address the variety of design goals associated with facility layout problems and is highly manual and so time consuming to perform. Furthermore, the SLP framework does not help the designer select a modeling tool to use in developing design alternatives, either by defining what a requisite model would include, or explicitly suggesting ones from literature. With the advancements made in academic research and computational capabilities since the development of the SLP framework, a new framework was needed to better address varying design goals, and assist designers in the selection of appropriate models. The framework proposed here guides the designer through determination of model requirements to meet their design goals by framing the FLP in terms of "Design Layers". In addition, it proposes candidate models (or methodologies) to generate analytically derived solutions for design goals such as construction of simple block layouts, or determination of input/output points and flow paths in order to create detailed block layouts. The models and methodologies proposed are shown to rapidly reach good candidate solutions to a wide range of design problems.

CHAPTER 1. INTRODUCTION

1.1 Introduction

In a competitive market every enterprise seeks to find ways to improve its ability to meet and exceed the demands of consumers. Doing so allows that enterprise to gain a competitive advantage and promote its own long term viability. One of the ways that enterprises can create and realize this competitive advantage is by working to ensure that their most basic systems are well organized. In a manufacturing environment this starts with having an effective and efficient production facility designs. Tompkins and White (Tompkins, 2010) estimated that since 1955, 8% of the US gross national product had been spent on building new facilities. Recent data shows that the annualized rate of total construction spending for the entire United States in December 2013 was over $930 billion, with over $570 billion on non-residential projects (Huesman, Holland, & Langley, 2014). Furthermore, the Material Handling Institute (MHI) which hosts ProMat and MODEX, the largest material handling, supply chain, and logistics conventions in the industry, gave a press release of February 6, 2013 stating that attendees of ProMat were planning on spending in excess of $9.8 billion on new material handling equipment and systems between February 2013 and July 2014. Given the significant investments that have been and continue to be made in new construction and material handling, a formalized rigorous method for optimizing the effectiveness of those investments is needed.

One way to optimize such investments is to insure that the facilities being constructed and the systems within them are designed to be as efficient as possible. One way to improve this efficiency is to rigorously plan the layout of a facility. The layout is the physical organization or arrangement of each of the different entities or departments within a facility. From a top level manufacturing perspective this could mean designing the building so that heavy machining areas are separated from reception and break rooms, or drilling down to finer level of detail; the exact placement of a milling machine and work bench within a job-shop. Formal research aimed at developing analytical models and solution methods for this process began in earnest in 1957 when Koopmans and Beckman formulated the facility layout problem (FLP) as a Quadratic Assignment Problem (QAP) (Koopmans & Beckmann, 1957). With the progression of research, the FLP can now take on many different forms depending on the goals of the designer, the assumptions they make, or the conditions they are attempting to solve for. Because there are nearly infinite levels of detail at which one can design, the FLP is best defined as "*determining the physical organization of a facility*" (Meller & Gau, 1996). Unfortunately, this overly broad definition is reflective of the disjoint nature of research in the field and perhaps a reason for the lack of application of rigorous mathematics methods and analysis to real world problems.

The majority of research that has been conducted can be classified as solving for one of three broad goals;

1) Solve for an optimal block layout
2) Optimally locate input/output (I/O) stations
3) Determine the best material flow network for inter-departmental material flows

The mathematics behind solving for any one of these objectives is challenging. This challenge leads researchers to either develop novel formulations for a particular version of the problem, or attempt to solve for multiple levels of detail simultaneously requiring even more complex mathematics. While useful, the vast majority of these different models either make critical assumptions about the design details involved, thereby restricting the applicability of the solutions that they can produce, fail to reach a provably optimal solution, or fail to reach any "good" solution rapidly enough to be used in practice.

As a designer there is limited time to be able to keep track of the current status of research, understand it, or even be able to select a model that perfectly matches up with his/her objectives. Furthermore, a majority of designers may not have the background or technical capabilities to correctly formulate and translate between the types of data they have available to them and the mathematical equations required for FLP models. Additionally, because the mathematics behind a majority of these frameworks is so ridged, a designer might not be able to gather/generate the necessary inputs for his/her chosen model.

Advances in research, combined with the efficiency of modern computing capabilities has allowed some models to reach at least locally optimal solutions in relatively short timeframes once the model is formulated. Such models are often more than sufficient for meeting the general goals of designers in practice. What is needed is a framework to guide a designer through the process of selecting a suitable model, or series of models, for constructing his/her FLP model(s) based on his/her design goals. Because there are so many potential goals/models to choose from consideration should be given to what constitutes a "requisite model" given the designers goal. A requisite model is defined as "a model whose form and content are sufficient to solve a particular problem" (Phillips, 1984). Such a requisite model would require the fewest amount of inputs from the designer and be able to reach a solution in similar or less time than is required to actually construct the model. As noted above, advancements in research and computational capabilities mean that time to solve for a select subset of models is not a major obstacle, however designer experience in constructing such models is still an obstacle to application. Therefore, once such a suitable model is selected a second automated process is needed to help the designer actually carry out the construction.

1.2 Outline of document

Chapter 2 contains a survey of relevant literature relating to the various design goals and frameworks, modeling approaches, and implementation methods for solving the FLP. Chapter 3 introduces a new integrated design framework and approach to solving the FLP. This framework aids the designer in defining and selecting a requisite model to meet his/her needs. In addition, using the new framework, a set of requisite models is identified and suggested for practical use. Chapter 4 shows the results of using this new framework both in comparison to a popular existing framework, as well as numerical results of using the suggested models on a set of test problems. Chapter 5 concludes the thesis and discusses future research and implementation opportunities.

CHAPTER 2. LITERATURE REVIEW

2.1 Designer goals in the FLP

As stated in Chapter 1, the FLP does not have an exact definition. Instead it takes on a variety of characteristics based on the particular aspects a designer chooses when determining an "efficient physical organization" of a facility. At the highest level the three primary goals addressed in research are (J. G. Kim & Goetschalckx, 2005):

1) Solving for a block layout
2) Determining the location of Input/Output (I/O) stations
3) Designing the material flow network

While the FLP was an issue long before it was first approached mathematically in 1957, all of the subsequent mathematical models address one or more of these 3 broad goals. For a designer, the choice of which model or method that would be most useful depends on a variety of factors, and the answer to four questions:

1) What stage in the design process they are at
2) What information they are trying ascertain by solving a mathematical model of their problem
3) What information they have to use as inputs for a model
4) What if any experience they have formulating and solving these models

If a designer is in the beginning stages of the project, they are likely still trying to define criteria, gather information, and otherwise assess their goals. The most widely recognized framework for solving the FLP is Muther's Systematic Layout Planning (SLP) (Owens, 2011). A flow chart of this framework is given in Figure 2.1.

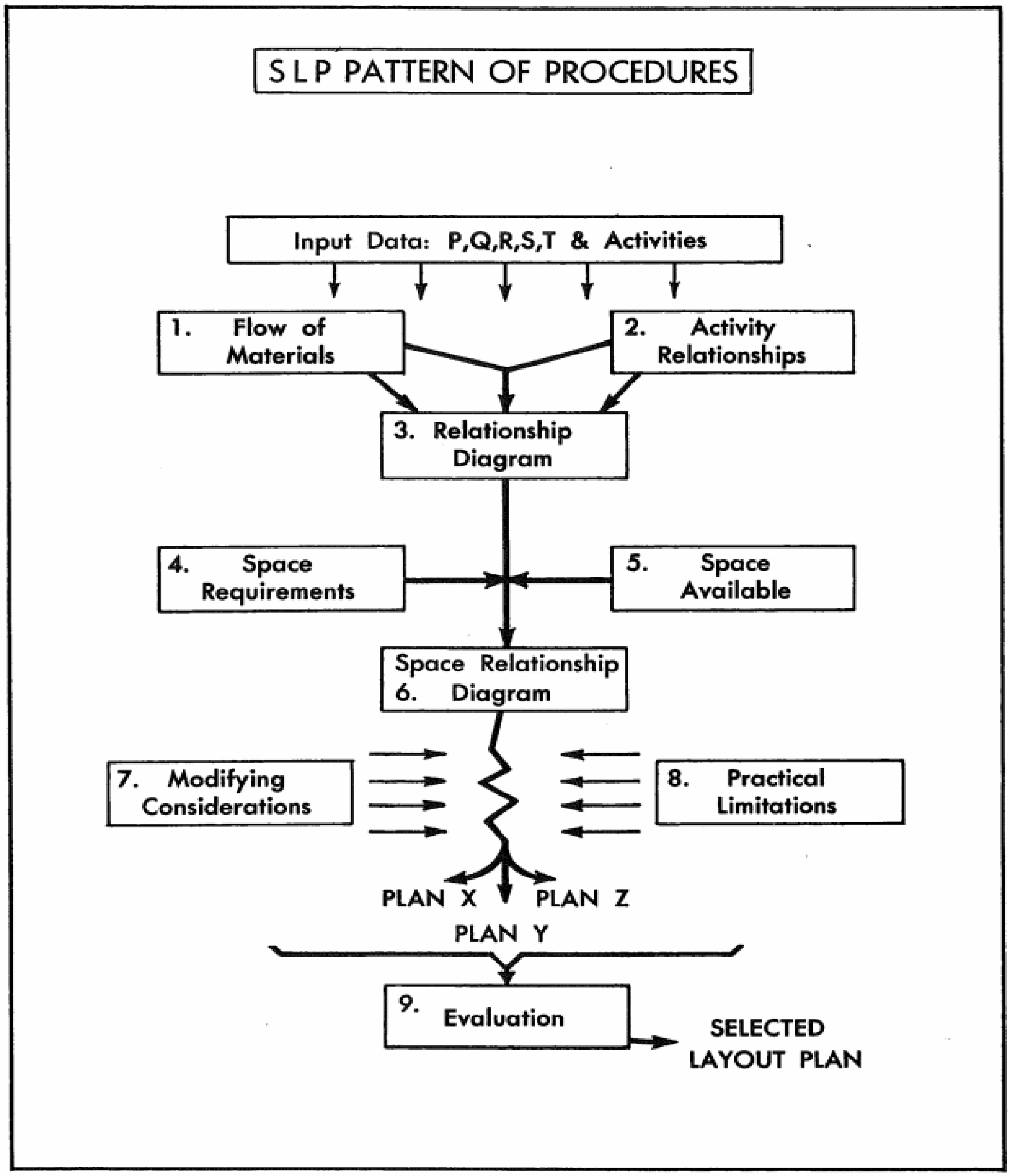
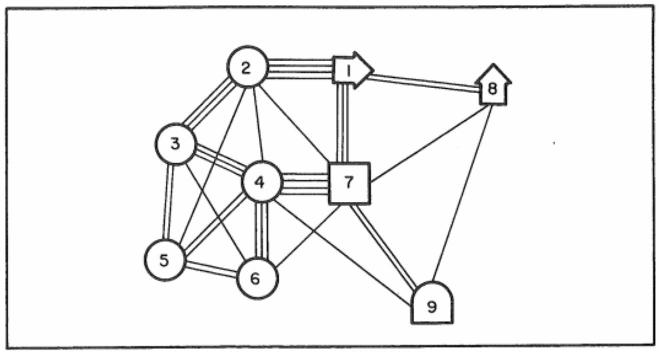


Figure 2.1: Flow Chart of Muther's SLP framework (Muther, 1973)

The first five analysis steps in this framework naturally align with this beginning or discovery stage of the design process. They are to gather information about material flows and activity relationships, and then use that information to create a relationship diagram. A relationship diagram is a weighted graph (likely non-planar) that helps the designer begin to visualize the interactions within the facility he/she is designing Figure 2.2. In Steps 4 and 5 the designer gathers information about the space requirements of the departments and the total space available in the facility before adding that layer of detail to the relationship diagram in step 6 Figure 2.3. Steps 7 and 8 are broad catch-alls to make the designer cognizant of any other considerations that might influence the facility design outside of departmental relationship and space requirements. In step 9 assumptions and other constraints are taken into account in order to begin developing and evaluating new layout alternatives (Muther, 1973). Unfortunately, this framework does not give much direction for how exactly to develop these alternatives, or what types of analytical models to use. This can leave designers with lots of background information, ready to find layout alternatives but unsure of where to look, and often forced to proceed manually based on their own intuition. Fortunately, answering questions 2 through 4 can provide some direction.



Process Chart Symbols & Action*	Symbols Extended to Identify Activities & Areas	Color Ident.	Black & White**
Operation	Forming or Treating Areas	Green**	
	Assembly, Sub-Assembly, Dis-Assembly	Red**	
Transportation	Transport-related Activities/Areas	Orange Yellow**	
Storage	Storage Activities/Areas	Orange Yellow**	
Delay	Set-down or Hold Areas	Orange Yellow**	
Inspection	Inspect, Test, Check Areas	Blue**	
*A.S.M.E. Standard **I.M.M.S. Standard (Adopted as basic to SLP procedure)	Service & Support Activities/Areas	Blue**	
	Office or Planning Areas, or Building Features	Brown** (Gray)	

Vowel Letter	No. Value	No. of Lines	Closeness Rating	Color Code
A	4	////	Absolutely Necessary	Red**
E	3	///	Especially Important	Orange Yellow**
I	2	//	Important	Green**
O	1	/	Ordinary	Blue**
U	0		Unimportant	Uncol-ored**
X	-1		Not Desirable	Brown**
XX	-2, -3, -4, ?		Extremely Undesirable	Black

Figure 2.2: A relationship diagram and key (Muther, 1973)

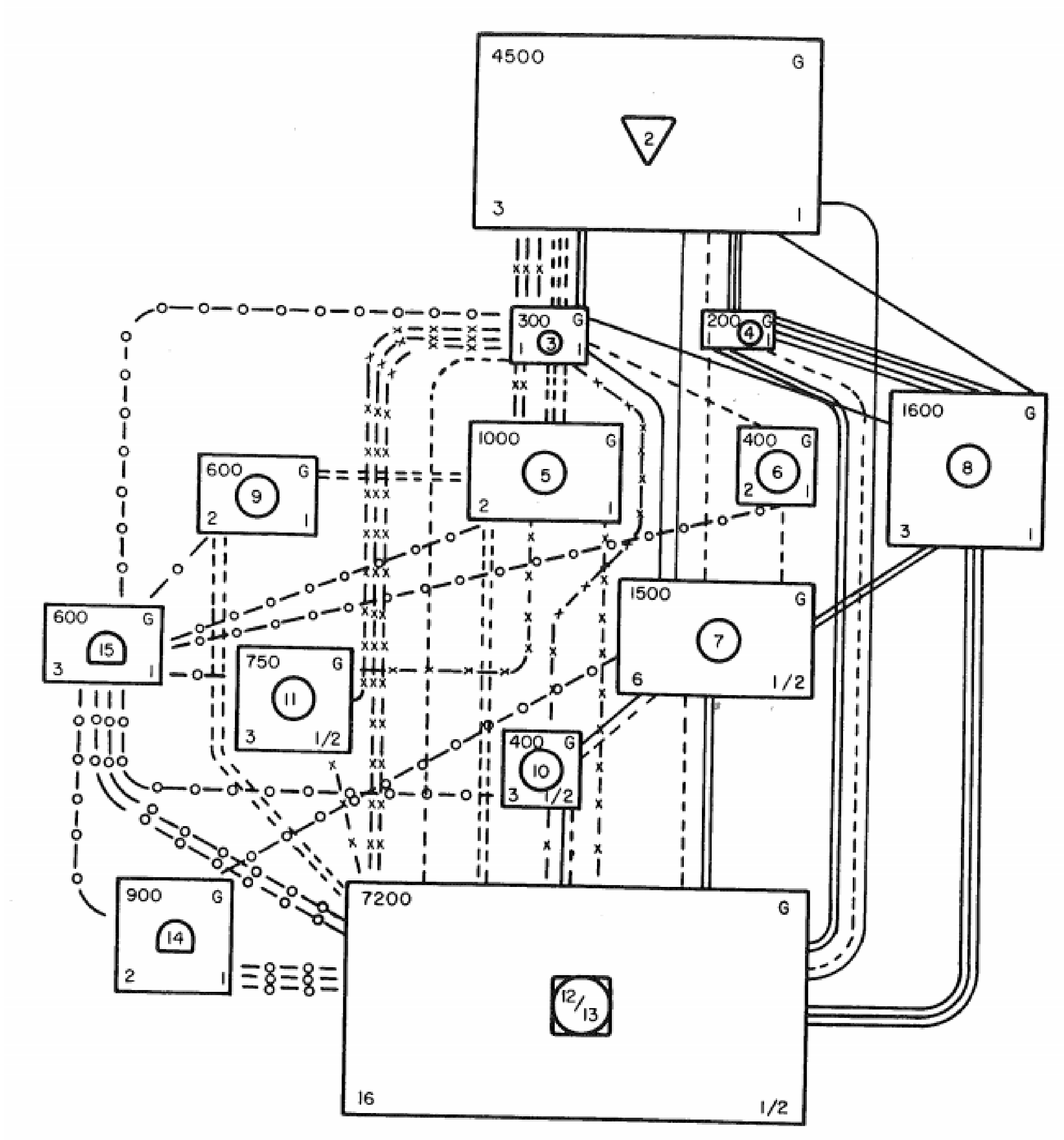


Figure 2.3: A relationship diagram after step 6 in SLP process (Muther, 1973).

It should be noted that there is not a set precedence which of these questions is most important to answer and indeed, may not be the same from one designer or situation to the next. Additionally, because the FLP is a demonstrably difficult problem to solve it is not unreasonable for there to be a problem that is well posed for a given model, but the lack of designer knowledge or ability to use the model results in the given model not being used and the designer's needs being left unmet (Schneider, 1960). The majority of models developed to date assume their inputs are available, the details they solve for, and the assumptions made in solving are relevant to the designer using them. While these may be necessary assumptions they are rarely explicitly motivated, or even addressed, leading to a large body of research that is left underutilized in industry (Meller, Kirkizoglu, & Chen, 2010).

2.2 Block layouts

2.2.1 Introduction

Creating a block layout is often the initial step in developing layout alternatives. Goals two and three, locating I/O points and determining the material flow network almost always require an existing block layout to work with in the existing literature. This is why block layouts can be simple abstractions of the departments being organized, or finely detailed representations. When solving the FLP with the objective of finding an optimal block layout, researchers typically approach the problem from one of two methods, either "Top-Down" or "Bottom-Up". An overview of these methods is presented in Figure 2.4.

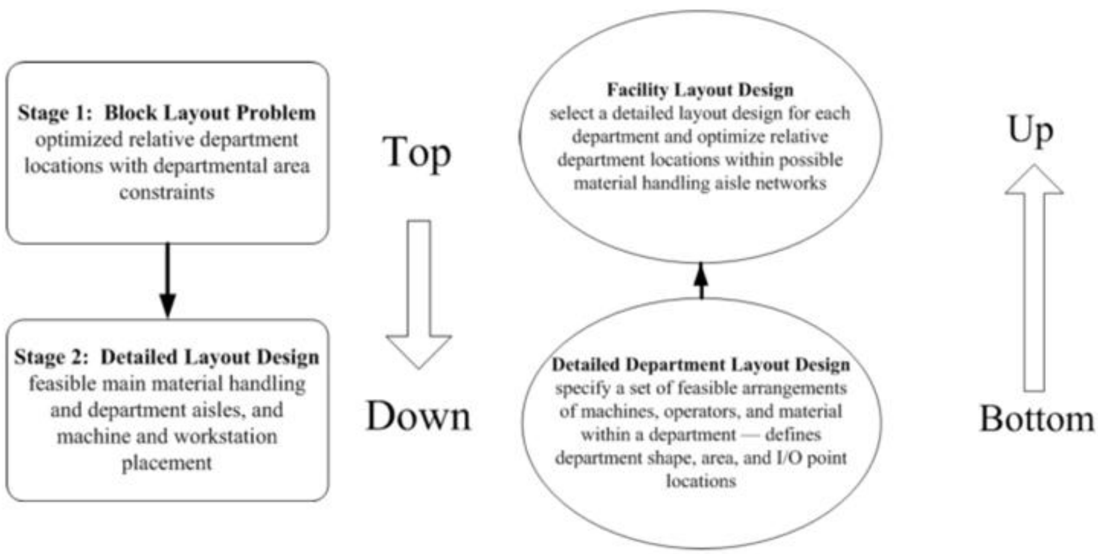


Figure 2.4: Comparison of Top-Down vs Bottom-Up approaches (Meller, Kleiner, & Nussbaum, 2004)

The choice of which approach to use involves a variety factors, examples of which include;

1) Is the layout problem being solved a new problem, or a re-layout of an existing facility?
2) Are department level details such as shape and input/output stations known, or at least able to be estimated?
3) Is there only a minimal amount of information to base the layout on, such as; n departments of size S_n?

Regardless of the approach used, some models will attempt to simultaneously solve for I/O station locations or flow networks within the block layout problem.

2.2.2 Top-Down approach

In the top down approach there is only a limited subset of details about the desired final facility design that are known *a-priori*. Examples of details in this subset might include; the total footprint of the facility to be designed, the area requirements for each department, a simple evaluation measure such as flow volumes, or a set of relative location/proximity preferences of the various departments within that facility. Using this limited set of information as a starting point an initial block layout is then developed. Much of the early research on the FLP adopted this Top-Down approach (Meller et al., 2010).

The earliest example of such research is the formulation of the FLP as a Quadratic Assignment Problem (QAP). By nature of being the earliest formulation it is also the simplest. Given a finite set of potential locations, another finite set of departments, and costs of locating a department in a specific location, the objective is to find the lowest cost arrangement of departments (Koopmans & Beckmann, 1957). Furthermore, this formulation of the problem has been shown to be NP-Complete (Sahni & Gonzalez, 1976). The difficulty in solving such a problem leads to the use of heuristics, and other imperfect algorithms that can be time consuming and unreliable (Drira, Pierreval, & Hajri-Gabouj, 2007).

While the original formulation as a QAP assumed identically shaped and sized departments, subsequent research has lead to modifications that allow for unequal sized departments by making them compositions of smaller departments. While making the model less restrictive, this grows the size of an NP-Complete problem making it even harder to solve. Other reformulations have attempted to improve the solvability of the problem by making the objective function linear instead of quadratic, or reformulated it as a mixed integer problem with varying limited degrees of success (Kusiak & Heragu, 1987).

Another method for applying the Top-Down approach is through the use of Graph Theory based models. This method represents each department within the facility as a vertex of a graph. This formulation focuses on adjacency preferences between two departments (Foulds, 1992). The goal is to generate, or determine a maximum weighted connected planar sub-graph of the overall graphical representation of the facility. The weights for each of the edges in the overall facility graph are based on a designer defined adjacency preference. Such a model/method is relatively simple to execute algorithmically using heuristics, however basing the optimality of the design on adjacency preferences alone does not necessarily imply a minimum material flow distance layout (Kusiak & Heragu, 1987). Additionally, finding exact optimal solutions for even small problems is just as difficult as solving the QAP (Meller & Gau, 1996). Furthermore, the graphical representation output does not define the shape, size, or even relative positions of any of the departments in a block layout beyond whether or not they could/should be adjacent.

Starting in the early seventies formulations of the FLP as a “packing” problem began (Brown, 1971). For this formulation the objective is to ‘fit’ each of the departments into a known overall facility footprint. This type of formulation typically involved rectangular shaped departments. One way to do this is through the use of cut-trees, as with Layout Optimization with Guillotine Induced Cuts (LOGIC) developed by Tam (Tam, 1992). This method takes an existing rectangular, or near rectangular facility and determines an optimal location to make either a horizontal or vertical cut in the facility. After the cut is made departments are allocated to either side of that cut. This process is repeated until no more cuts are needed to separate departments. A more famous set packing method is the mixed integer problem formulated by Montreuil (Montreuil, 1990), and later improved by other researchers (Heragu & Kusiak, 1991; Meller, Narayanan, & Vance, 1999). In this formulation, variables are defined for the area, length, and width of departments, along with their tolerance thresholds for each of the preceding variables, and relative location binary variables, and flow volumes. Using these variables, an objective function and constraints are written to define the locations of each department within the facility and minimize the overall material flow volume distance. While this formulation could potentially give an exact optimal solution, the large number of variables required limits it practical application to facilities of fewer than 10 departments (Tompkins, 2010).

One final variation of the top down approach was inspired by thinking about the FLP from a physical perspective; modeling departments as discs connected by springs. In the early eighties, the DISpersion CONcentration (DISCON) method, set the foundation for such methods. Through the use of Lagrangian gradients the DISCON method is able to reach locally optimal block layouts for unequal area department problems often in less than 10 seconds of computing time (Drezner, 1980). One of the reasons for the difficulty in solving the QAP is that the solution space is non-convex, while mathematical optimization methods/solvers often require a convex solution space in order to work. One way around this problem is to solve using Lagrangian gradients.

Solving using Lagrangian gradients quickly reaches a local optimal solution without necessarily guaranteeing global optimality. The drawback of this method is that the final solution is highly dependent on the initial conditions, a drawback that other works have sought to reduce. The Attractor-Repeller, an improvement on DISCON, follows a similar logic and maintains the efficient solving times but is still highly dependent on the selection of initial conditions (Anjos & Vannelli, 2002).

More recently Castillo and Sim developed a method that creates a convex objective function and constraint version of the problem that allows for the generation of globally near optimal, or optimal solutions albeit with an increase in solving time. However, they reported testing a 30 department problem, and found a solution in less than 7 minutes, using a 2004 computer (Castillo & Sim, 2004). From a practical prospective, more time likely went into formulating the problem and entering it into the solver.

2.3 Input/Output (I/O) point location problem

Once an initial block layout is found the typical next step is to begin creating a more detailed block layout. Creating a detailed layout often involves determining the internal layout of each department such as machine or workstation placement as well as the location of I/O stations within the department. Given that Top-Down layouts generally assume centroid approximations for evaluation, which is not reflective of real world applications (J. G. Kim & Goetschalckx, 2005), adding realistically i.e., on the outer perimeter of the department, defined I/O points as a layer of detail allows for a more accurate measure of flow costs in the final design (H. Warnecke, Dangelmaier, & Kuhnle, 1985).

Unfortunately, the body of research for the I/O location problem is more limited than that of the block layout problem. Also, such methods are usually focused on automated guided vehicle (AGV) systems. These works still try to minimize flow distance much like many of the block layout models. Because I/O location problems try to more accurately capture real flows between departments they are usually based on rectilinear distance minimization algorithms. Most early works on the I/O station location problem focus on locating I/O stations in the context of the total layout, i.e., within a set block layout, such as with Montreuil and Ratliff (Montreuil & Ratliff, 1988). On the surface this appears to be a good strategy. Unfortunately, with this approach I/O stations can be placed anywhere within a department. As a result, departments at the edges of a block layout will tend to have their I/O stations on their inner perimeters while departments on the interior will tend to have their stations deep within the department. Such locations provide lower objective function values, but fail to meet the practical need.

This is because any flows from other departments must still enter the department from some particular point or else have no obstructions from any particular point on its perimeter into the specified I/O station. Additionally such formulations are basically block layout problems where the individual I/O stations represented by departments with minimal areas forced to fit within the area of their associated department, i.e. any areas outside the associated department are made infeasible. This in turn makes solving for I/O points in this manner just as, if not more, difficult as solving regular block layout problems

Another challenge many I/O station location problems face is that the potential aisles or paths which would connect such stations are not known. In this situation, the designer might then pick a modeling framework that is based on rectilinear distances, however if the aisle structure is designed to be unidirectional, the results of such a model may not be useful (Sinriech & Tanchoco, 1992). Fortunately, the majority of layouts do end up using bidirectional paths, and if such unidirectional paths were to be implemented it is more likely that the designer would start by designing said paths and then fitting I/O stations along said path.

A different method for determining I/O stations is to arbitrarily pick candidate points along/within the department borders and then solve for the best sub-set of candidate locations. In one of the most recent surveys of FLP research it has been noted that the majority of methods for solving this problem have adopted this approach (Drira et al., 2007). As the number of candidate positions or number of departments grows, this method becomes computationally infeasible to solve for optimality. Kim and Klein developed a model using this method but take advantage of block layouts with rectangular departments arranged in a grid. Given the grid formation, flow network characteristics naturally induce optimal I/O points to be at the corners of departments. This model is relatively efficient but still struggles with larger scale problems (J. Kim & Klein, 1996). Along this same idea, other methods start with block layout that have exactly defined I/O points and then attempt different orientations of these blocks within the layout in an attempt to find an optimal arrangement (J.-G. Kim & Kim, 2000; Meller et al., 2004).

In the early 2000's (Arapoglu, Norman, & Smith, 2001) adapted the candidate point selection process to layouts with similar characteristics to those created through guillotine cut algorithms, which they referred to as flexible bay layouts. Relevant research on flexible bay layouts can be found in (Peters & Yang, 1997). Arapoglu *et al.* located candidate points at corner points and intersections of departments, similar to the Kim and Klein approach, but then use bidirectional contour distances, around the perimeters of departments, rather than rectilinear distances to more precisely calculate the flow distance. When solved as an integer program this method is tremendously time consuming especially for large scale problems, however they apply a genetic algorithm to reduce solution time and still achieve good solutions, never more than 10% difference from known optimum, and often achieved optimality. Additionally, this search method yielded results in less than 90 seconds for even a 60 department problem that was intractable from an integer programming approach.

While many I/O location problems start with a block layout, some assume that aisles or flow paths are already determined and then try to locate the I/O points along these aisles (Benson & Foote, 1997). Recently Meller has proposed a method for determining I/O location as part of the block layout design process, in what he calls a "Bottom-Up" approach, discussed in detail in Section 2.5 (Meller et al., 2004).

2.4 Flow path location problem

The last major goal of solving the FLP is the determination of material flow paths within the layout. Part of this goal may also be determination of the flow methods/equipment; e.g., conveyors, rollers, forklifts, ect. This would be an important process given that the characteristics of a particular equipment choice may dictate the choice of flow path; however that is outside the scope of this research. Given the close relationship of I/O points and flow paths it is unsurprising that much of the early work is also based around developing an AGV network. The importance of having a well planned flow network is established by the fact that while "optimal" block layouts and I/O point locations are necessary for a successful facility design, they are both products of idealized material flow distances. This makes planning the actual distances and paths within the true physical layout incredibly important to overall facility design (Maxwell & Muckstadt, 1982).

Early work to determine the best flow paths was based on applying integer programming to a completely defined flow network and then selecting the components that minimized total flow distance (Gaskins & Tanchoco, 1987). Other works consider the pros and cons of allowing bi-directional flow or requiring unidirectional flow as it relates to system efficiency, amount of trips/vehicles required and congestion, ultimately suggesting that bi-directional flow offered significant advantages provided adequate control was maintained (Egbelu & Tanchoco, 1986). One drawback of these approaches is that they primarily focus just on material flow, ignoring the impact that empty transportation flows may also be required, especially relevant in AGV system designs and other unit load transportation systems (Alagoz, Norman, & Smith, 2008).

If one assumes bi-directional flow, a block layout with rectangular departments and one co-located input/output station per department yields a flow network graph with at worst an average of less than six vertices per department. Assuming a complete graph, all vertices connected to all other vertices, the Floyd-Warshall Algorithm can be used to find the shortest paths connecting all pairs of vertices in $O(2n^3)$ time (Floyd, 1962). Given that the a block layout must be planar, the more efficient Johnson algorithm can be used (Johnson, 1977). Given that the typical design problem has less than 60 departments to arrange and the computational power of modern computers the worst case performance of these algorithms would be on the order of minutes (Katz & Kider, 2008).

If a known aisle structure is given, methods have been developed to optimally route material through that structure, i.e., define the flow path, which would be useful in improving an existing layout's performance but is less applicable to when designing a new facility from scratch (Chhajed, Montreuil, & Lowe, 1992). Other, methods have been developed for taking a simple block layout without pre-determined I/O points, adding aisles to them and then determining I/O points (Alagoz et al., 2008). However such methods require that department sizes be inflated when creating the block layout in order to account for the space that aisles will occupy in the final design thereby adversely affecting the validity of the starting block layout (J. G. Kim & Goetschalckx, 2005). Furthermore, there is no guarantee that the aisles that are created will require a proportional amount of space from every department (Meller et al., 2004). That said, such methods are feasible and likely do not propose a significant reduction in the overall practicality of the design.

2.5 Bottom-Up approach to block layouts

Starting in the early 1980's, there has been a paradigm shift, and emergence of an alternative to the Top-Down approach (H. J. Warnecke & Dangelmaier, 1984). Many researchers have shifted their focus and developed new models with a more detail-oriented bottom-up approach. Given that solving for any one of the three individual analytical goals of the FLP either requires another goal to have been done before-hand, and/or adversely affects the solution of that earlier goal, a new more integrated approach is needed. It has also been noted that for all the work that has gone into advancing research on the FLP, there is little to no use of it in practice (Meller et al., 2004). The authors also note that the majority of designers typically approach the FLP with all goals in mind. That is they begin designing the detailed layout at the same time as they work on the overall block layout, often without the aid of analytical methods.

For solving purposes, this newer approach assumes that the internal structures, or a set of alternative internal structures for each of the different departments within the layout are known *a-priori*, while the overall facility structure is undefined. Examples of these internal structures include exact locations of I/O stations and/or well defined shapes of the constituent departments. As an example a designer might develop 3 alternative layouts for a department such as shown in Figure 2.5. If the designer does not have a preference as to which alternative is selected it would then be useful to have a model capable of selecting whichever one best fits into the overall facility design Alternatively the designer might be set on one particular arrangement of a department, but would then like to mathematically determine how exactly to place it within the facility. Examples of such placement options are given in Figure 2.6.

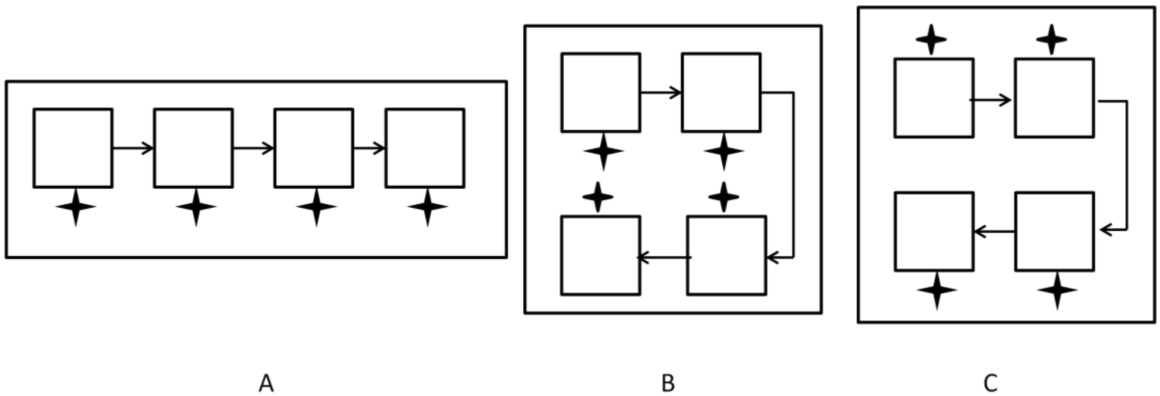


Figure 2.5: Alternative arrangements for a department with 4 elements

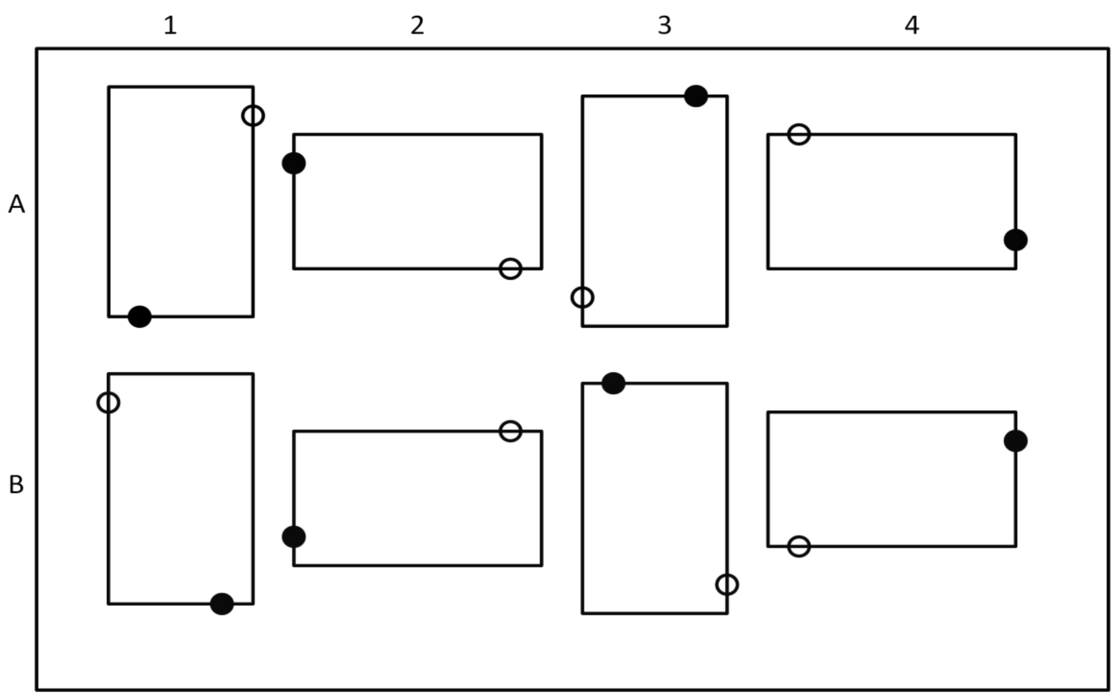


Figure 2.6: Alternative ways to apply a well defined department shape within a layout

This *a-priori* knowledge is in direct contrast to the Top-Down perspective of centroid approximations for I/O stations and either circular or loosely defined rectangular departments (Meller & Gau, 1996). While the standard output of a Top-Down approach is a simple block layout, which still requires determination of I/O stations and flow paths, most bottom up approaches determine I/O stations and/or flow paths simultaneously as they find the block layout. As a rough illustrative example, a Bottom-Up approach can be loosely thought of as fitting together a puzzle of departments; determining their relative locations to each other, defining the final footprint of the facility and flow paths as the puzzle is put together. The advantage of this type of approach is that it gives much more realistic and applicable outputs for later stages of a design process than the traditional Top-down approach, at the cost of more complex modeling and increased solving times. A criticism of this approach is that it lacks an overall final vision and so may produce layouts that lack a feasible final form (Smith, 2005). In support of a bottom-up approach, the designer is allowed the freedom to use other factors such as ergonomics or safety, that are not as easily captured in mathematical models, in order to develop departmental layouts (Meller et al., 2010).

2.6 Commercial implementations of research

While the past sixty years of research has seen many improvements in formally modeling and solving the FLP, there is a significant lag in the industrial and practical applications of these advances. In late 1995 there were only four recognized layout packages available that used an algorithm to assist in generating a block layout, however none of these packages has ever become prevalent in the market (Meller & Gau, 1996). A recent article in IIE magazine discussing the state of the are in facility layout design work in Asia notes that there is almost no use of software packages in the region, while their use in the US and Europe is primarily for precise flow cost calculations within an AutoCAD drawing (Owens, 2011). As such, the programs still rely on the user to input a design and then manually change it based on the computations that the program returns. Developing tools that will guide designers through a logically organized design process, helping them determine which goal(s) is most important to them, selecting a relevant model in order to generate graphical designs backed by the mathematical rigor of FLP research methodologies remains the "Holy Grail" for the field of FLP researchers and practitioners alike (Sly, 1995). Two commonly used tools in industrialized nations are Flow Planner and FactoryFLOW. These software tools rapidly calculate flow costs and compute relationship charts provided a CAD drawing of the layout is available. This is useful for carrying out the SLP process but still relies on the designer to make changes to find improvements.

CHAPTER 3. METHOD

3.1 Introduction

In creating new facility layouts, designers are faced with making a multitude of assumptions and choices. First, they must decide what goals they want to solve for and then determine what factors they believe to be most important for evaluating designs. Examples of these factors include: flow distance, or minimizing the material handling cost within the layout, or proximity rating heuristics (e.g., important close, unimportant, important far). They must also decide if rectilinear, Euclidean, or path-directed flows (e.g., contour distance) should be used to evaluate the design. Is it enough for them to have a rough block layout, or do they need detailed flow paths and I/O stations to evaluate a design? Or, is some other simple visualization needed? Other aspects to consider include safety considerations of a layout, or the basic feasibility of a laying out different departments within a desired footprint. After deciding which goal(s) and factor(s) are most important they must then select a method from a copious body of existing methods and build an appropriate model that is best suited to meet their needs.

Assuming they are able to select a suitable model they must then make further assumptions about things like the shape, or area of the departments they are trying to arrange, the volume of flows between each department in order to use the selected model. Finally, after all these different selections and assumptions have been made they must then get a solution from the model they chose and evaluate it. If the solution is not satisfactory, the designer is forced to go through another round of:

1) Deciding which goals and factors are most important
2) Selecting a suitable model
3) Making relevant assumptions about department characteristics and flows
4) Solving the model
5) Evaluating the solution

This process of determining goals, models, and assumptions often involves significant time and effort, at the conclusion of which the designer still has to make another attempt to solve a complex mathematical problem all without knowing if this new result will be satisfactory or not.

Furthermore, the majority of the models that have been developed and that are capable of reaching a solution are single objective. This means that a designer might be forced to conclude that minimizing interdepartmental flow distance is the primary and so only criteria for evaluating a layout in their chosen model. He or she might then have secondary factors such as the feasibility or other physical restrictions on placing a particular department into one area of the layout, or safety factors for which they may have to manually manipulate the mathematically derived solution. Incorporating these "secondary" factors into a layout evaluation either must be done on an ad-hoc basis or by arbitrarily manipulating the model, which may require a level of understanding about the model mechanics that the designer simply does not have.

3.2 Design layers

The purpose of this research is to establish an overall design framework that any designer can use to guide them through a facility layout project, specifically considering the usage of quantitative analytical models in order to construct candidate designs. This framework will guide the user through a series of steps similar to the SLP framework but with a greater emphasis on how to select a requisite model to meet the unique needs of each design project. When carrying out a facility design project there is often a tremendous amount of details. The same is true even when the scope of the project is limited to generating a simple block layout. It is therefore useful to start out by defining the scope of your goals, or defining the design objective for solving the FLP. As noted in Section 2.1 the primary mathematical goals of solving the FLP are;

1) Solving for a block layout
2) Determining the location of Input/Output (I/O) stations
3) Designing the material flow network

The information required to meet these goals can be categorized into one five design layers shown in Figure 3.1.

Using the Design Layers categorization, the highest level of abstraction that a designer can work is termed "Facility Basics". A few examples of details in this design layer might include the total area of the facility to be designed, the total number of departments to be included in the facility, and the required area for each of these departments. This layer of detail is required regardless of what the design objective is, as the information at this level of detail is used as key inputs for any of the modeling methodologies a designer might choose. It is analogous to steps 4 and 5 from the SLP framework. The details included in this design layer are the foundation for all of the other design layers.

The next design layer, "Evaluation Measures", is also critical to any modeling framework. This layer is analogous to steps 1, 2, and 3 in the SLP framework. Given that the FLP seeks to determine an optimal physical arrangement of a facility, a set of criteria is needed in order to establish a definition for an optimal, or at least superior, arrangement. It is also important to consider the level of accuracy or how sensitive the designer wishes the evaluation measures to be; e.g., should the design primarily driven by exact flow distance calculations, or are things like Euclidean approximations acceptable, are other non-flow factors the most important aspect? This is especially true because choices made in this layer begin to eliminate different modeling methods. Examples of details in this layer focus on the relationships and interactions between departments. Questions to ask when determining this layer of detail include;

1) What are the characteristics of the different types of flows? e.g., materials, personnel, electronic data
2) What types of equipment or paths are use to move between the two departments?
3) How does the ease or difficulty of moving these different flows affect their relative importance for flow distance calculations?

Based on the answers to those questions a designer can determine what method to use for calculating flow distances. Other questions may help determine non-flow relationships between departments. Such questions include: Are there environmental, safety, physical, convenience, or ergonomic factors that are affected by how close or far away one department is from the other? The answers to these questions connect to the subsequent design layers and are similar to steps 7 and 8 from SLP. Evaluation along these factors may or may not provide enough information to suggest one requisite framework over another. Once the questions about all of the different types of flow and non flow factors have been addressed, effort is needed to evaluate the relative importance of each of these factors to one another, so that a final composite evaluation measure can be formulated.

Below the evaluation measures design layer is the "Flow Types and Volumes" design layer. While the designer may choose to evaluate his/her alternatives based on non-flow factors the vast majority, in practice and certainly in literature, focus on total flow-distance minimization. Given that material handling is not a value adding process and yet must be done, it makes sense that designs would seek to minimize it. Some flows are obviously easier to move than others; therefore it is important to know which types and their associated volume so that relevant unit load scaling factors can be used in order to accurately account for each type of flow in the evaluation method.

The different "Flow Modes" used to transport the flows identified above are addressed in the next design layer. Determining the flow modes again helps to define the relevant unit loads which will in turn help the designer in determining the weighting factors that should be used for the cost calculation of each flow. As an example, department A might send a large volume of data electronically to department C, f_{ac}, while department B sends physical goods to department C, f_{bc},. In this case the flow from B to C should be more important in determining the layout than the electronic flow from A to B. In this case $f_{ac} > f_{bc}$ but the objective function should include weighting constants, w_{ac} and w_{bc}, such that $w_{ac}f_{ac} < w_{bc}f_{bc}$. Similarly, if specialized equipment that only has limited range or an exactly required configuration is used, that information would be required in order to build a requisite model.

The last design layer focuses on department characteristics. Decisions made at this level of detail determine what constitutes a requisite model for the designer's particular design goal. If the departments do not have a defined shape, or known I/O station locations a "Top-down" approach will use the facility basics and a weighted evaluation measure based on the flow volumes and modes to generate a simple block layout. If a block layout is already known a different requisite model can be used to determine I/O station locations and/or the flow paths. If the designer wants to add some assumption about the shape of the department such as making them all squares, rectangles, or circles a more refined, but still top-down, model can be applied. Once the designer chooses to specify the location of I/O stations within departments the design process shifts from the traditional top-down approach to the bottom-up approach. This will require more effort in creating the model but can lead to better performing or at least more accurate layout models of the facility.

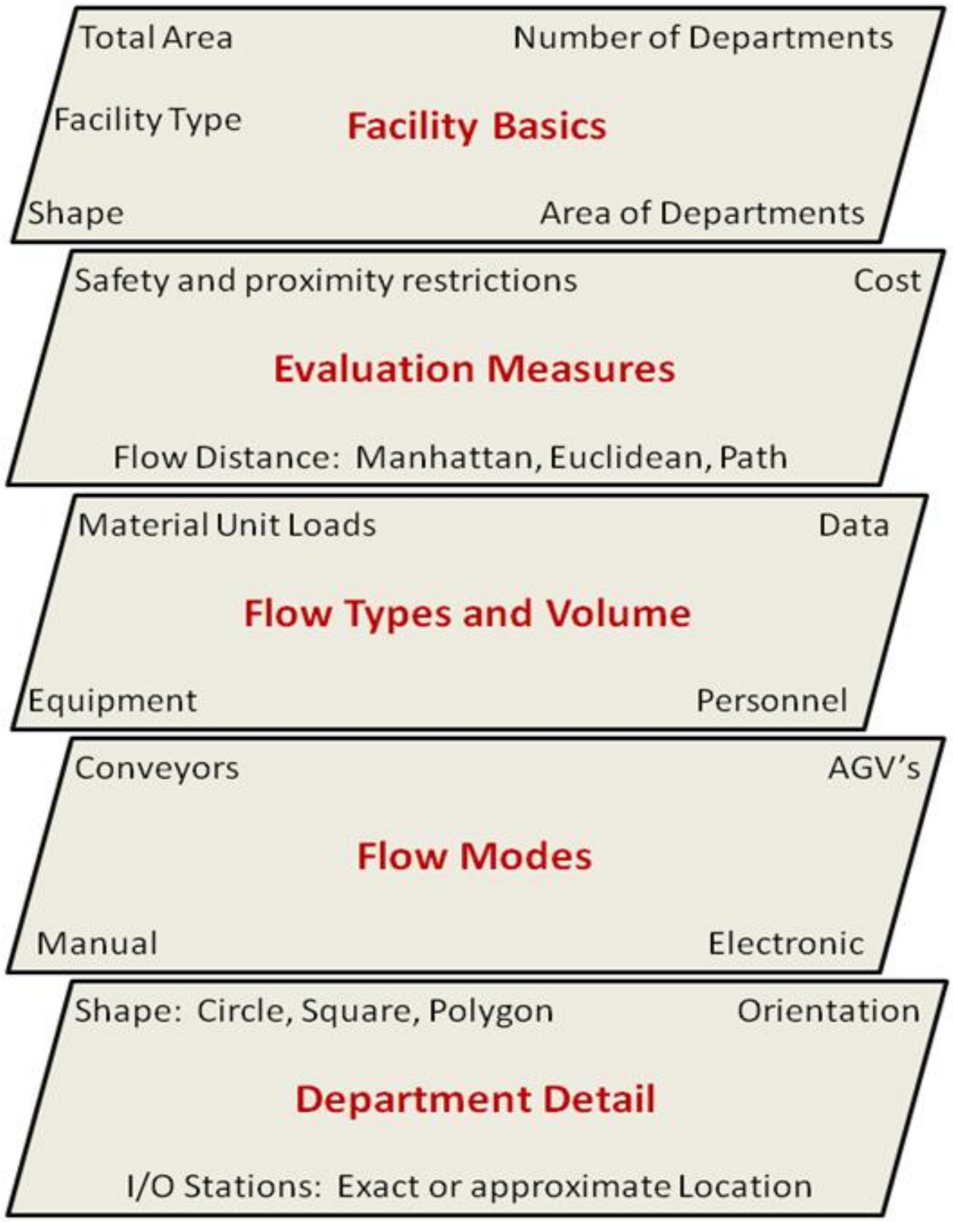


Figure 3.1: Design layers of facility layout projects

3.3 A new design framework for model selection

As can be seen there are a multitude of different pieces and types of information to first identify and consider when solving an FLP. With each level of detail addressed comes a set of assumptions that must then be made in order to reflect that level of detail in the model to be created. After the desired levels of detail and related assumptions are determined, the designer must then either find and select a model capable of handling the specified level of detail, or create their own. A typical facility layout designer may not be an engineer trained in mathematical modeling, or if they were they may be many years removed from such training. Lastly, assuming they have such training it was likely not at the Masters or Ph.D. level, which is where the vast majority of new models are developed. All of these factors make it difficult for designers to successfully create a new model, and as a result are forced to select from one of the previously published works.

Just as there are many layers of detail to consider when starting a facility layout design project, there are many things to consider when selecting a model for any one particular FLP. As outlined in Section 2.1 the goals typically considered in current mathematical representations of the problem include:

1) Solving for a block layout
2) Determining the location of Input/Output (I/O) stations
3) Designing the material flow network

Solving for any one or multiple of these goals is complicated by various aspects and details in each of the design layers, particularly at the lowest level, that can be difficult to quantify. Examples of these include: physical restrictions within the facility on particular department locations, safety considerations of the layout, ensuring effective utilization of the entire space within the facility, the interactive effects of two departments; e.g., having an office space placed next to heavy manufacturing, or even the effect of department locations within the facility on overall building aesthetics.

Assuming a designer is successfully able to find a model capable of giving the level of detail they desire, they must then convert all of the information they have acquired from going through the different design layers into the appropriate form of inputs that their chosen model requires and then build it. Next, they must hope that the model that they have built is capable of being solved. Again, outside of special cases of particular detail levels and assumptions made this is rarely the case. Lastly, because the design process is itself fluid or at the very least iterative, details and assumptions will change at various stages of the design process and so require the designer to then go back, adjust, and then resolve their model. Worse still, if any key assumptions are changed or different levels of detail are required the designer might have to go back to seeking out an entirely new model.

Remarkably, given all the of challenges a designer faces in selecting and building a model, there is no established framework for how to go about selecting a requisite model to help them achieve their goals. Muther's SLP framework does not address how to do so and simply assumes that the models will be generated and solved using designer expertise and then a "best" one selected, often based on simple visual inspection. Indeed, this is the very process that is followed in industry. This new framework will put the design layers into context and guide designers through determining a suitable requisite model to meet their design goals.

As noted in Section 3.2 the first step is to establish the design objective for solving the FLP. In the context of this new framework, most design objectives can be grouped into one of four categories;

1) Evaluation of an existing block layout
2) A rough cut analysis for the FLP
3) Generation of a simple block layout
4) Generation of a detailed block layout

3.3.1 Evaluation of an existing block layout

If the goal of the designer is simply an evaluation of an existing block layout there is minimal need for any kind of model selection or solving. This assumes that the designer already has some form of block layout as a starting point, and seeks to get a numerical evaluation based on some set or subset of factors. From this starting point the designer can focus on the last 4 design layers. Starting at the evaluation measures layer, there are 3 main areas to evaluate a layout on;

1) Cost of implementation of the layout
2) Non-Flow and other basic proximity factors
3) Flow based factors

3.3.1.1 Evaluation based on cost of implementation

If the designer wishes to evaluate a design based on the cost of implementation, the evaluation measure is typically a discrete dollar figure often estimated by the group responsible for the implementation. This means that the designer, or group doing the implementatino can focus on the unique department details such as work station installation or equipment costs. Furthermore, implementation costs are typically discrete factors determined by the characteristics of the chosen layout. This means that they can be thought of as binary decision variables, i.e., is a given characteristic present? Yes: add associated cost, No: do not include cost. Such characteristics are predominantly determined by the Facility Basics, Flow Modes, and a Department Detail design layers. As such, all that is needed is for the designer to examine the relevant design layers, summarize the costs, and add them together. This can be done either by hand or with a simple spreadsheet.

3.3.1.2 Evaluation based on non-flow factors

Evaluating a layout based on non-flow factors; e.g., physical feasibility, safety, or adjacency preferences follows a similar methodology to evaluating implementation costs. A list of relevant factors needs to be compiled, an appropriate scoring system devised, and then an evaluation of the layout based on that system. Details from any of the design layers can be used to compile potentially relevant factors. Additionally the Activity Relationships and Relationship Diagram stages from the SLP process can also be used to compile potential non-flow factors. Once the different factors and scores are compiled, a simple spreadsheet can be used to calculate the final evaluation.

3.3.1.3 Evaluation based on flow-factors

Using flow based factors to evaluate a layout becomes a more involved task. First the designer must decide how detailed they wish to be in their evaluation. Fortunately there are a multitude of tools available to accomplish this goal. The majority of layout generation methods solve for optimal layouts by minimizing flow distances, therefore if this is being done immediately after solving for a layout one can simply reference the score from the layout objective function. If, however, the layout generation method does not have the desired level of detail, the designer still has other alternatives available. If the layout being evaluated has been converted into an AutoCAD drawing, there are tools such as FlowPlanner™ that allow the designer to specify all the relevant information, i.e., flow volume, paths, and I/O points and then will automatically calculate the exact flow distances as well as identify potential congestion points. Alternatively, the designer can generate the from-to flow volume matrix, as well as manually determine the relevant distance matrix for each department within the layout, multiply and thereby determine a flow distance evaluation score.

3.3.2 Rough cut analysis for the FLP

The next design objective that a designer may have in solving an FLP is performing a rough evaluation, or generating a basic visualization of the problem in order to guide their efforts in the design process. Beginning with this objective assumes that designer is seeking a low fidelity result or simple visualization of the problem. As such, they likely have minimal information at any of the design layers and might use this objective as a way to determine where they should look to add more details. This objective is likely an initial stage in a new layout project or an attempt to visualize an existing system in order to begin looking for potential improvements. Such an analysis typically focuses on the first four design layers and is not meant to generate optimal or even near optimal layouts. Given the lack of need for optimal layouts, in order to meet this design goal, it is suggested that the designer follow up to the first 8 steps of the SLP framework, as well as considering any relevant characteristics from each of the design layers. Going through these steps guides the designer through identifying different factors that they may wish to consider, helps them to determine basic characteristics such as size and flow volumes, as well as generating relationship diagrams to help visualize the particular FLP they are working on. From this point the designer may choose to go further and analytically develop a block layout based on some form of mathematical model.

3.3.3 Generation of a simple block layout

3.3.3.1 Framework Process

If the designer's objective is to create a simple block layout it is assumed that they do not have a set vision or design for the individual departments. However they should have the majority of the first design layer established, namely the total number of departments and the area of those departments. Knowing this information they would then have a rough estimate of the total facility area required as well as be able to make allowances for the area that would be needed for aisles. Additionally, they will need to choose an evaluation measure. As highlighted in the literature, the most commonly chosen evaluation measure is total *flow*distance* cost. This framework emphasizes categorization of flow volumes as well as modes so as to accurately weight each of the flows in the objective function. For this goal, the designer typically lacks useful information from the last design layer, Department Detail, per the objective being a simple block layout. Working from these assumptions the designer must pick a requisite model. That is one that can meet their design objectives relatively efficiently. The selection of one model over another is about making tradeoffs. Some models are capable of capturing more detail often at a cost of long run times or failing to reach a solution. Other models capture fewer details but reach optimal solutions. Therefore, it is important to recognize how such outputs will be used so as to allow the designer to determine the best trade-offs for his/her particular goal.

A simple block layout lacks the finer details of architectural blue prints, and regardless of the method chosen, will not be able to account for all of the design factors involved with new facility construction. With this in mind it is more valuable to use the development of a simple block layout to solve for the details it is best suited for, namely minimizing total *flow*distance* cost and proximity relationships. This leaves the designer free to choose a modeling framework that is not burdened by having to solve for a large amount of details, while still remaining confident that such a framework will adequately meet their objective.

3.3.3.2 A practical implementation

Given the typical design objectives versus solvability tradeoffs, along with the flexibility to incorporate multiple factors, it is suggested that the designer choose the spring embedding approach proposed by Castillo (Castillo & Sim, 2004). Such an approach gives the designer the flexibility to enter more than 30 non-uniformly sized departments, specify a total facility area with which to fit the departments in, solve based on material flow costs and or proximity factors by choosing appropriately weighted "springs" and converge to a near optimal solution in less than two minutes. Furthermore, it has been shown that the spring embedding model's use of Euclidean vs rectilinear distance does not significantly affect the quality of solution found (Blanks, 1985).

That is not to say that rectilinear or path distances are not more accurate, however when the objective is to find the lowest cost alternative having a more precise value is often inconsequential, or within an acceptable margin of error. Additionally, such calculations lead to non-convex solution spaces and thereby make it difficult to find good solutions. This gives the designer the flexibility to try multiple iterations and adjust various parameters frequently. In turn, this allows the designer to generate multiple alternatives that he/she can share with other stake-holders. Unfortunately, this approach models each department as individual circles of varying radius and generates a “bubble” layout such as the one in Figure 3.2, rather than the traditional block layout.

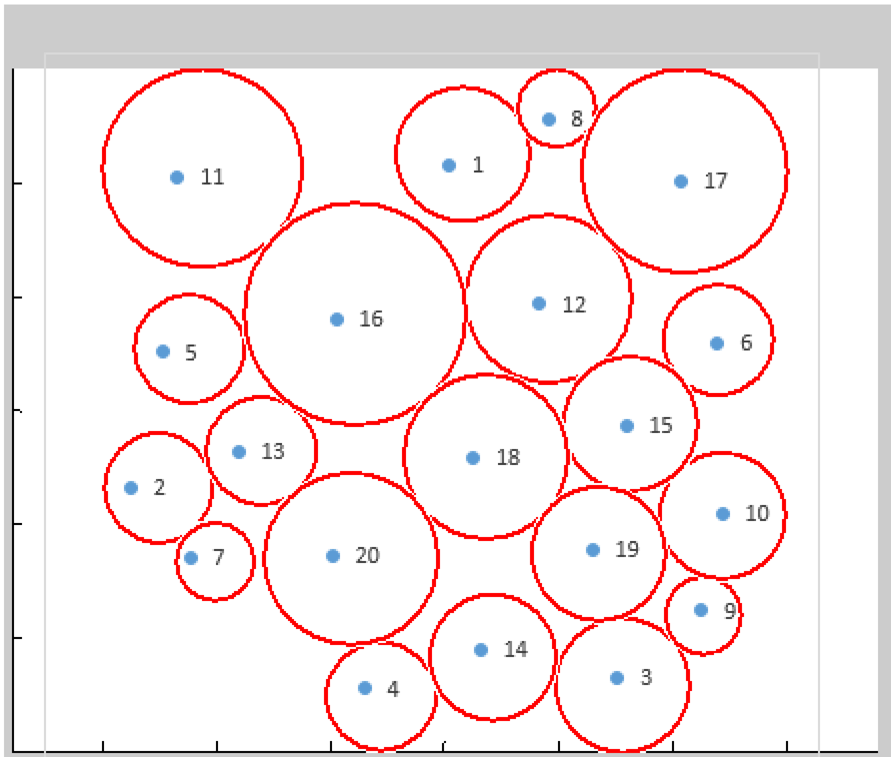


Figure 3.2 Example output from a spring embedding approach model

Fortunately, such a layout can be readily converted into a block layout through an interactive guillotine cutting process. A guillotine cut completely divides a given area into two parts. Such a process would work by having the designer select the direction of the cut, vertical or horizontal, and the two other cuts that it would intersect. The designer can then specify which departments should be placed on one side; e.g., above or left, of the cut based on interpreting the bubble layout. All other departments would remain on the opposite side. Once this information is specified it is possible calculate the exact location of the cut in order to match the area of the specified departments in the desired region. Figure 3.3 gives a detailed explanation of the process. The precise nature of the calculations and logic structure of this process could easily be developed into an automated program. Such a program would know the coordinates of the endpoints, and therefore the length of all previous cuts, as well as the areas of the specified departments. It could then rapidly calculate the exact location for the endpoints of the new cut. This process would be repeated until all departments are in their own unique block. Once the designer finishes this rapid, semi-automatic procedure they are left with a suitable simple block layout. Furthermore the coordinates for all of these endpoints can naturally be used as inputs for subsequent stages in the design framework.

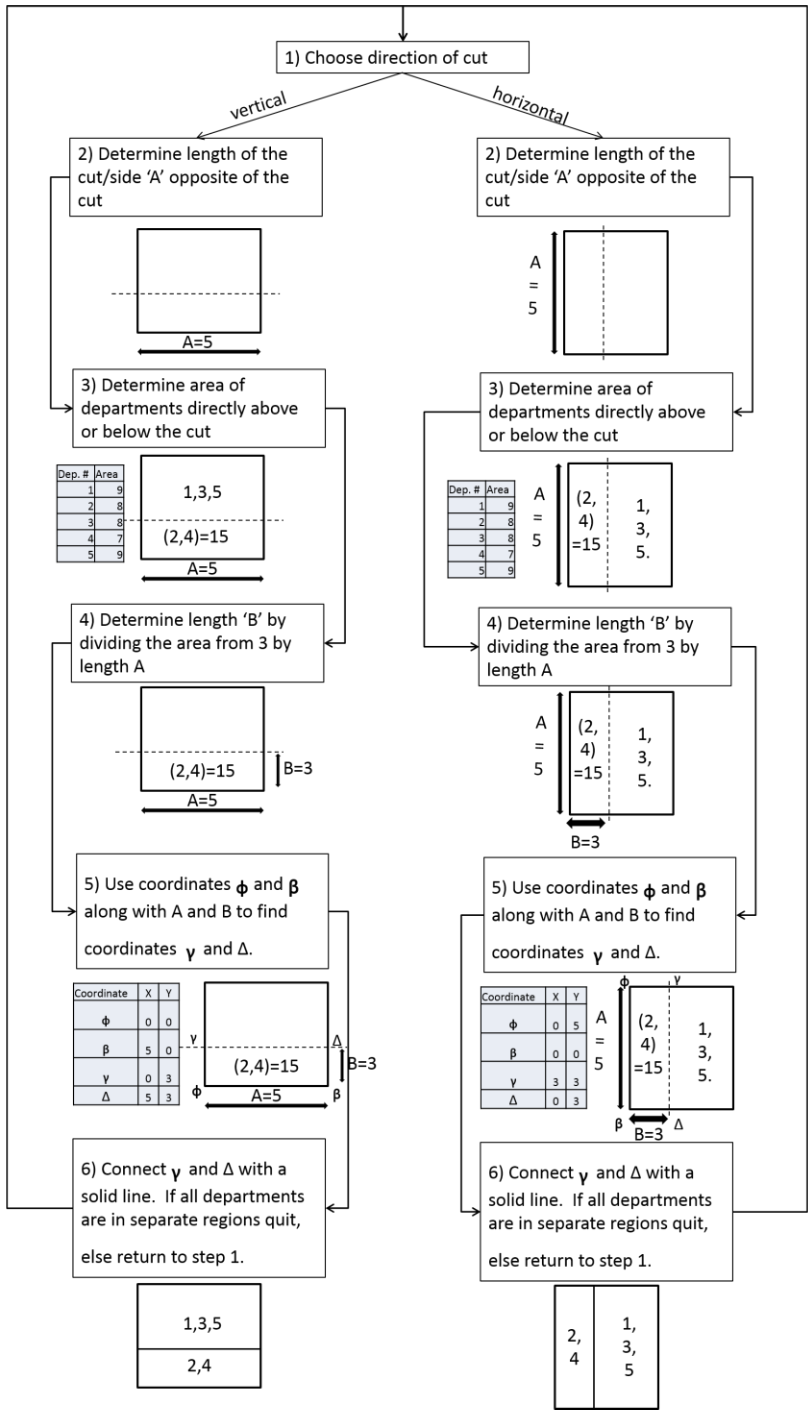


Figure 3.3: Flow chart of translation process from bubble to block layouts

3.3.4 Generation of a detailed block layout

Detailed block layouts are the result of completing all three analytical goals for solving an FLP. As a reference these goals are:

1) Solving for a block layout
2) Determining the location of Input/Output (I/O) stations
3) Designing the material flow network

Having I/O stations and the material flow network defined allows the designer to achieve a more complete picture, and aids in the analysis of the facility being designed. Once converted to a graphical representation it is also easier to share with other members of a design team that may not be as comfortable with mathematical models and abstract representations. Knowing what factors and methods went into creating the block layout and details within it, the designer can confidently share the layout with the knowledge that there is an analytical foundation for its construction. The designer/design team can then proceed to evaluate other factors which may cause them to alter the layout before finally creating architectural plans. The next sections guide the user through selection of a requisite model for adding I/O stations and flow paths to a simple block layout.

3.3.4.1 Determination of I/O stations in a block layout

3.3.4.1.1 Framework process

Given the relative dearth of methods for determining I/O stations, especially in comparison to creating a block layout, designers are forced to accept a few assumptions in order to make use of the available models. Many of the assumptions at this stage come naturally from those used to generate the preceding simple block layout. While not absolutely required for all models, the assumptions of;

1) The existence of a simple block layout composed of rectangular departments
2) Bi-directional flow paths of negligible size
3) Each department has a set of candidate points for I/O stations,
4) Candidate I/O stations are located on the borders of departments

are used in many models. Should the designer wish to start with I/O point location and then build a block layout from there, a select few methods exist, however they do not integrate well with other models and do not guarantee success, as such they are left out this research (Chittratanawat, 1999; Ho & Moodie, 2000).

One assumption that will be violated is the idea of flow paths with negligible size. Clearly any aisle will take up space within a facility however without knowing exactly where these aisles will be *a-priori* there is, as yet, no good way to solve for the required space without generating an aisle structure first. In order to address this issue, most methods suggest that the designers inflate the size of all departments prior to generating a block layout so as to account for eventual flow paths. The assumption of an existing block layout is standard for the design objective. The limitation to rectangular departments is made primarily to simplify the formulation and solving of the problem. This assumption would also flow naturally if the method used to generate the simple block layout created regular departments. Lastly the assumption of candidate I/O points can easily be satisfied by arbitrarily selecting the corners of each department, if the designer does not have other candidates in mind. It has been shown that layouts with rectangular departments will most likely have optimal I/O stations at the corners of such departments, thereby making it easy to identify potential candidates (J. Kim & Klein, 1996).

3.3.4.1.2 A practical implementation

If all of the assumptions detailed above are made, based on the ease of implementation and the computational efficiency, it is recommended that the designer use the contour method developed by (Arapoglu et al., 2001). This method is able to derive all of its inputs directly from a simple block layout generated using the procedure outlined in Section 3.3.3.2. Furthermore this method has been demonstrated effective even with a large number of departments. If a department is not able to locate its I/O point exactly in the corner determined by an initial run, it is not a significant issue. The speed of the algorithm used to solve the problem allows the designer to simply specify new candidate points in the feasible region, i.e., on the department border, and resolve in a matter of seconds. Furthermore, because this method uses contours, i.e., paths along the perimeter of departments, rather than rectilinear distance, the resulting *flow*distance* calculations are likely to be as reflective of real world results as possible. Lastly the way in which this method solves for I/O point locations determines the flow paths within the block layout as a sub-routine. This is done by performing the Floyd-Warshall algorithm to determine the shortest path between any two candidate I/O points before using a genetic algorithm search for the optimum set of selected I/O points.

3.3.4.2 Determination of flow paths in a block layout

While not quite as limited as the I/O station location problem, the flow path determination problem is also not as extensively studied as the block layout problem. Fortunately, if the designer follows has used the methodologies suggested, and maintained the assumptions outlined for developing the simple block layout and determination of the I/O point location, determination of the optimal candidate flow paths is a by-product of solving for the I/O locations. Should the designer instead already have a block layout and I/O stations but not know the flow paths, they would simply need to convert the block layout and I/O points into a graph and apply either the Floyd-Warshall or Johnson algorithms to determine the paths. Given the scale of problems typically solved in a facility design project, and the computational capabilities of today's computers, this is an effective and rapid solution procedure that once entered in can likely be solved in a matter of minutes for even large problems.

3.3.4.3 A bottom up approach to facility layout design

The key characteristic of the Bottom-Up approach is that the designer is given the ability to exactly define the departmental layout, or at least is able to define acceptable alternatives prior to the determination of the relative locations of departments within the facility. Computationally this approach is more difficult than the majority of individual

Top-Down alternatives. Furthermore, it is a relatively new approach from an operations research/optimization perspective, hence the relative lack models that adopt this approach. Additionally, the justifications for the different alternatives are usually non-mathematical and therefore hard to include into a modeling methodology. It should also be noted that the lack of a final vision for the complete facility, typical of this approach, can also lead to infeasible outputs even if such models are developed and solved. All of these factors make the bottom-up approach heavily reliant on the human designer's input in order to propose alternatives and determine a final feasible solution.

The main goal of the Bottom-Up approach is to give the designer the flexibility to adjust the shape and characteristics of departments before they are set by a block layout. Given the interconnected relationship of all of the design layers in a facility layout problem, any decision made at one layer likely has a ripple effect throughout the other layers. It would therefore be reasonable to focus efforts on implementing a bottom-up approach in such a manner as to maximize its advantages while attempting to minimize the potentially negative effects.

Given that alterations to the characteristics of an individual department, i.e., its shape and I/O point locations do not affect the typical flow and proximity relationships used to determine the adjacencies and proximities of different departments within a block layout it is reasonable to still use the spring embedding approach to get an initial approximation of a simple block layout, namely a "bubble layout". The departments within the layout begin to take shape during the translation from the spring embedding output into the final simple block layout. Given that the method outlined for how this translation might occur relies on human inputs as well, it is reasonable to have this stage be a reasonable starting point introducing a bottom-up approach.

While the simple block layout translation method mathematically determines where to make a guillotine cut based on area calculations, if the designer is able to generate alternatives using the same area as was used in the spring embedding model and maintains rectangular departments they could feasibly alter the shape of departments especially early in the translation process. It should be noted that as the translation process gets closer to completion there is less ability to alter the shapes and still maintain the final facility shape. As such this limits the negative ripple effects of infeasibility at the expense of a more constrained bottom-up approach. On the positive side, given that the designer determines the shape and I/O locations for many of the departments during this stage, the candidate points for these I/O points would then be known in advance and thereby able to be fed into the next stage of developing a detailed layout.

3.3.5 The complete framework

Each of the preceding sections has outlined either a different stage or goal within a new design framework, as well as a proposed solution methodology. Because design is an iterative process and a designer might have different goals, multiple passes through this framework pictured in figure 3.5 might be made in order to fully satisfy a designers needs. As such the "stops" can also be thought of as signals to return to the beginning.

As an example; a designer might initially wish to assess the problem he/she is facing and so go through the SLP method but stop before developing different alternative designs. Upon completion of this initial assessment the designer might then wish to generate a simple block layout. Once the simple block layout has been generated the designer might then wish to go back and evaluate it based on "non-flow" factors. After completing that evaluation the designer might then wish to create a detailed block layout from the simple block layout they created earlier. They could then use the algorithmic methods previously outlined for determining I/O stations and flow paths, or they could choose to go about it from a Bottom-Up approach. Finally after completing the detailed block layout the designer might need to evaluate the detailed layout based on the cost of implementation before proposing it to other stakeholders in the design process. An example of this type of process is given in figure 3.4.

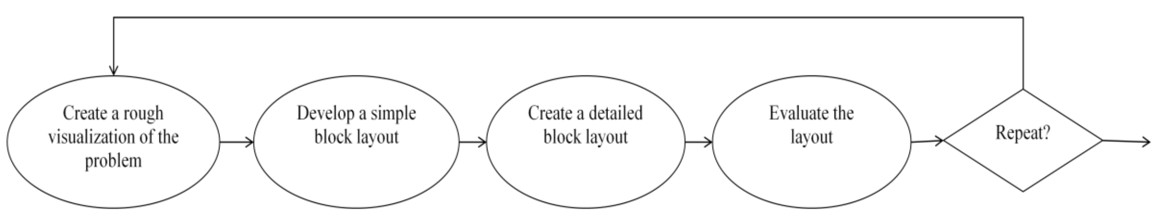


Figure 3.4 A sequential path through the new design framework

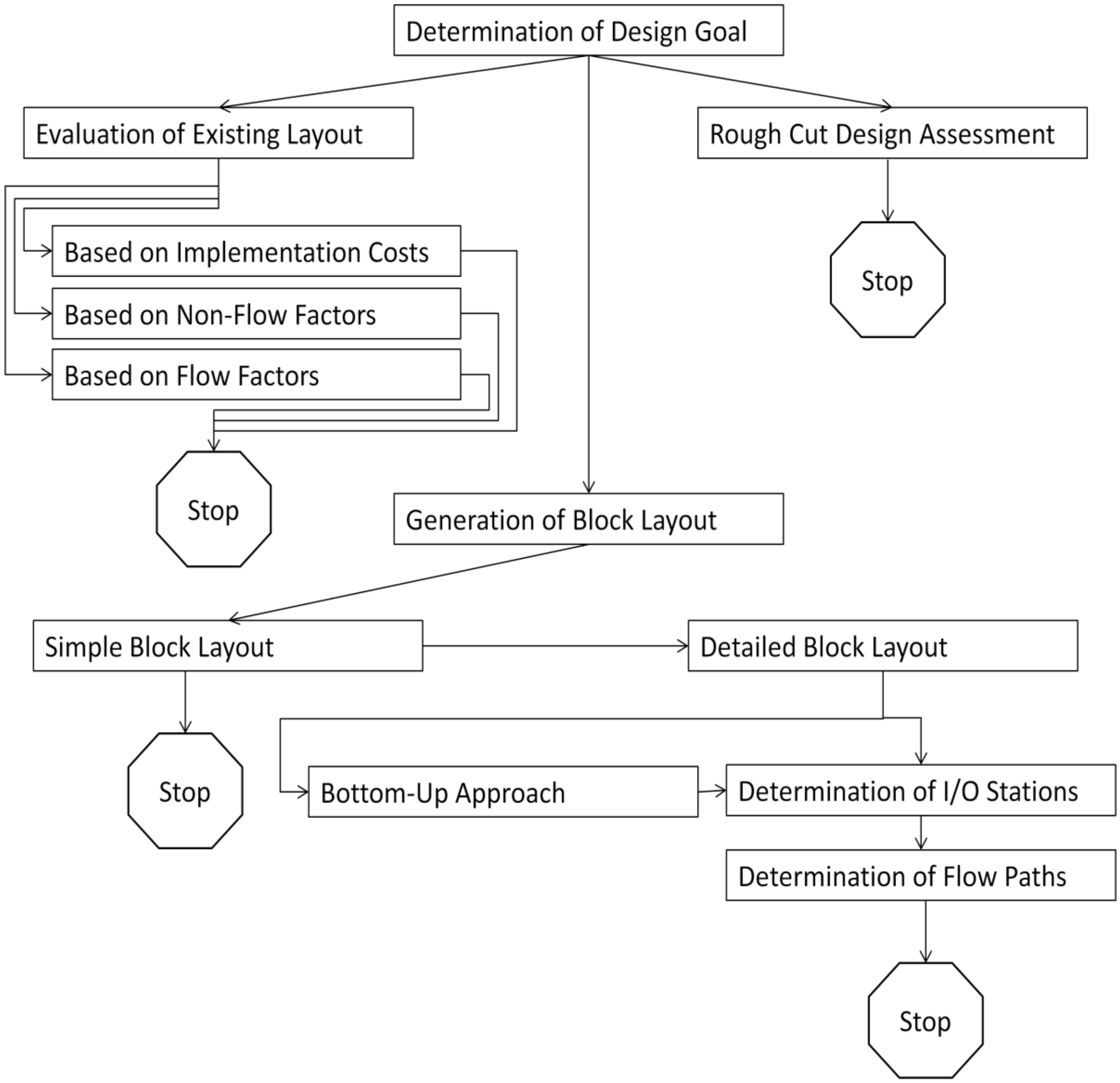


Figure 3.5: Flow chart of the new integrated design framework

The concept of design layers was developed to help the designer think through the multitude of factors and aspects of a facility layout design project. Each of these layers is connected to one another. All of the mathematical methods for modeling and solving for the different design goals make a set of assumptions that may or may-not be reflective of what the designer wishes to accomplish. In an effort to make this framework as simple to follow, and universally applicable, the methods chosen for generating block layouts require only department areas, and flow volumes. In going through the different design layers, there are clearly other factors to consider. Some can be addressed by adding appropriate weighting factors to the flow volumes. For those that cannot be addressed in this manner, the framework is designed to be iterative, allowing the designer to go back, evaluate, and manually modify the outputs of these methods, before settling on a final design.

Lastly, by choosing the methods outlined in the framework above, different assumptions are made at different times, allowing for multiple design layers to be included based on what stage the designer is working on, while also allowing for rapid calculations of exact solutions. As an example, when initially generating a block layout, Euclidean distance measures are used. However once this initial layout is generated, contour, or actual path distances are used to determine I/O points and flow paths. This allows the designer to have the most accurate evaluation of final design, without requiring overly intense calculations in the preliminary stages of the design project.

CHAPTER 4. RESULTS

4.1 Comparative analysis of the design framework

As noted in Section 2.1, the most widely recognized and used framework for solving the facility layout problem (FLP) is Muther's Systematic Layout Planning (SLP). This framework was developed early in the 1970's before it was practical to solve many of the real world versions of the FLP analytically. As such, much of this framework focuses on manual tasks to help the designer to loosely categorize pertinent information and derive solutions by hand. Inputs or design details in the SLP framework are reduced to four broad categories:

1) Flow of materials
2) Activity relationships
3) Space requirements
4) Modifying considerations.

Given the manual nature of the solution generation process it was reasonable to have such a limited breakdown. Further highlighting an issue with the SLP framework, the problem of how to generate and/or evaluate different alternatives is not well addressed. This lack of rigor can lead to different designers getting different solutions without knowing how or why such results occurred.

By comparison, the new integrated design framework developed here begins by having the designer establish what their design goal will be. The categorization proposed for the different design goals addresses the majority alternatives that a designer may choose between. They are also designed so that the designer is able to either build on them in a sequential, iterative manner, or the designer can pick one particular goal to solve for independently of the others. In order to do this a new method for categorizing inputs, Design Layers, was developed. This categorization highlights how different goals and methods use different details to derive their solutions. It also helps show how choices made in one of the design layers can create assumptions that limit the choices that can be made in other layers. Using the fact that each design layer choice carries an implicit assumption of requisite details, this categorization is designed to help the designer to determine the requisite set of information for the particular goal(s), method(s), and detail(s) he/she is using to solve his/her particular FLP.

In addition to guiding the designer through goal determination and input classification, this new framework also helps the designer by suggesting a set of relevant analytical models to meet his/her needs. If the designer is more familiar, or would prefer to use, other models that still meet the requisite characteristics of this framework they are free to do so. The SLP and other frameworks like it give little or no guidance for which models to use or how to select them. This is then a major issue for designers who lack knowledge about: what models are available, how they work, and/or how they are implemented. Without a way to determine a requisite model it is incredibly difficult for a designer to utilize the modern analytical tools available to them. Furthermore, even if they are aware that different models exist, without knowing exactly what their design goals are and the relevant information required to use them such models are of little value. By combining all three of these different stages together;

1) Determination of design goal
2) Gathering of pertinent design details and inputs
3) Selection of relevant analytical model

this new integrated framework closes the goal determination and model selection gap in the SLP framework. Additionally, it improves on input gathering stage by highlighting interactive effects of different input and evaluation selections, and guides the designer in determining only the requisite information needed to meet his/her goal(s). Lastly, practical alternatives were proposed to demonstrate how such a framework might be used to develop relevant design alternatives.

4.2 Numerical results for selected models

In order to prove the viability of the new framework, and modeling methods outlined in Chapter 3, twelve problems were tested. These problems were selected to cover a range of potential number of departments (10-30) and density of the flow matrices, (sparse, medium, and dense). Additionally both equal and unequal area department sizes were tested. Each of the test problems was either taken directly, or adapted, from relevant literature and are listed in appendix A. They were run in Matlab 2012a ® on an Intel® Core™ i5-2500 3.3 GHz processor with 4GB of RAM. After the flow matrix and area of each department was entered, none of the individual optimization portions;

1) Generation of bubble layout
2) Determination of shortest paths for candidate I/O points
3) Selection of optimal I/O points

of the test cases took more than two minutes of processing time to complete. The translation from a bubble layout to a block layout was done manually. This step took between five minutes, for 10 department problems, and up to an hour, for 30 department problems, to complete. For all problems the bubble layout problems were constrained to a square area slightly, 5% to 15%, larger than the total area of the individual departments. This was done to force the solver to generate compact square facilities while minimizing departments overlapping. This set-up aids the designer in visualizing each of the departments' relative locations in the layouts and thereby makes it marginally easier to translate the bubble layouts to simple block layouts though it is not necessary to use this restriction. The block layouts were constrained to squares with a total area exactly equal to the sum of the departmental areas.

It is assumed that the designer had gone through the earlier stages of the design framework and done any appropriate scaling to the flow matrix values in order to reflect his/her design goals in the input. Additionally, it is assumed that candidate I/O stations were located at the corners and intersections of departments as outlined in Section 2.3. Note, the selection of different candidate points along the perimeter of the departments would not affect the solution performance time. Lastly, it is assumed that each department would have one and only one combined input/output station. It is believed that expanding the model to include separate input or output stations would be feasible and not significantly hinder performance.

Direct comparison of the final results from the test problems to those found in literature is difficult. Six of the twelve problems tested were generated by converting an unequal area problem into a congruent equal area problem. As such there are no values to use for direct comparison of these problems. Of the remaining six problems, test problems 2, 6, 8, and 10 had the same flow matrix as those tested in (Arapoglu et al., 2001). However the simple block diagram generated in the earlier stages of the framework is not the same as the block diagrams used by Arapoglu *et al*. Additionally, for some problems in their paper Arapoglu *et. al.* tried multiple layouts with the same flow matrix, which explains the range of values listed for problems 8 and 10. Given this information, in some cases the final results of the new integrated design framework out-perform the comparison values, and in other cases they do not. The final total *flow*distance* of each problem is reported in Table 4.1. The outputs for each stage of problem 2, along with the final output from (Arapoglu et al., 2001) for comparison is given in Figure 4.1. The outputs for each stage of the other test problems are available in Appendix B.

Table 4.1: Numerical Results from test problems

Problem Number	Source	Number of departments	Area Classification	Flow Density	Flow*Distance Total	Comparison Flow*Distance Total
1	(van Camp, Carter, & Vannelli, 1991)	10	Equal	Sparse	9368.6	N/A
2	(van Camp, Carter, & Vannelli, 1991)	10	Unequal	Sparse	6090.0	7239.0
3	(Yang & Peters, 1998) Period 1 Scenario 2	12	Equal	Medium	4792.3	N/A
4	(Yang & Peters, 1998) Period 1 Scenario 2	12	Unequal	Medium	3046.2	N/A
5	(Bazaraa, 1975)	12	Equal	Dense	5557.9	N/A
6	(Bazaraa, 1975)	12	Unequal	Dense	4766.8	4991.8
7	(Armour & Buffa, 1963)	20	Equal	Sparse	466.9	N/A
8	(Armour & Buffa, 1963)	20	Unequal	Sparse	481.5	202.2-391.4
9	Flow values (Nugent, Vollmann, & Ruml, 1968) Areas: (Tam, 1992)	30	Equal	Medium	23233.0	N/A
10	Flow values (Nugent, Vollmann, & Ruml, 1968) Areas: (Tam, 1992)	30	Unequal	Medium	19495.0	2728.8-12846.2
11	Flow values (Nugent, Vollmann, & Ruml, 1968) Areas: (Tam, 1992)	20	Equal	Dense	8220.3	N/A
12	Flow values (Nugent, Vollmann, & Ruml, 1968) Areas: (Tam, 1992)	20	Unequal	Dense	7716.1	N/A

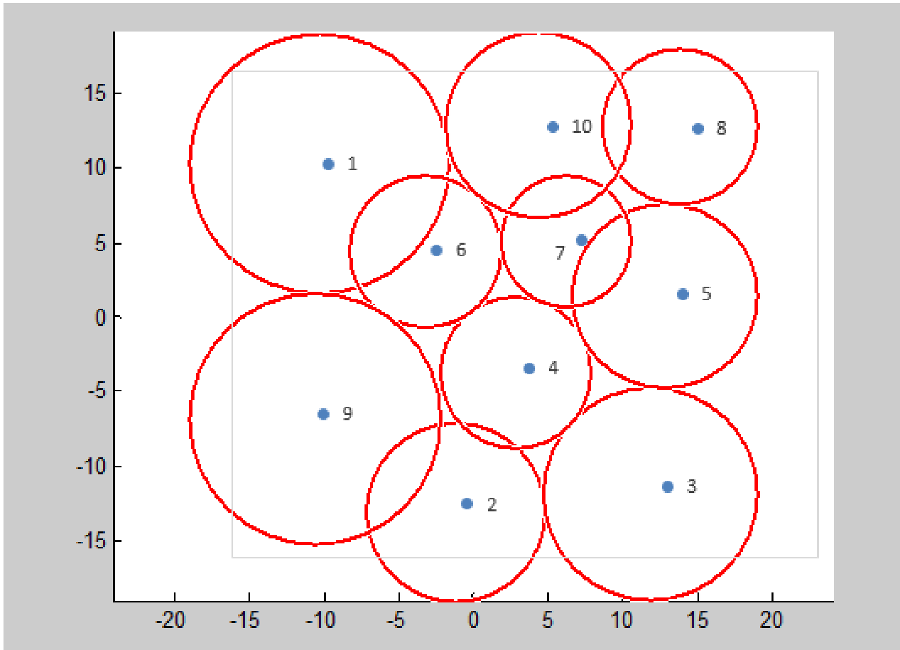


Figure 4.1 a) Bubble layout

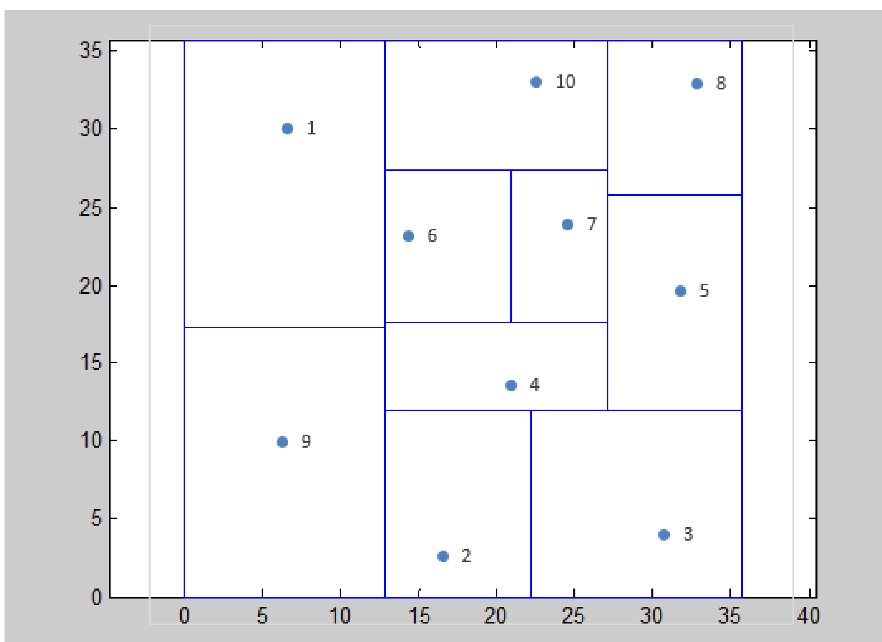


Figure 4.1 b) Simple block layout

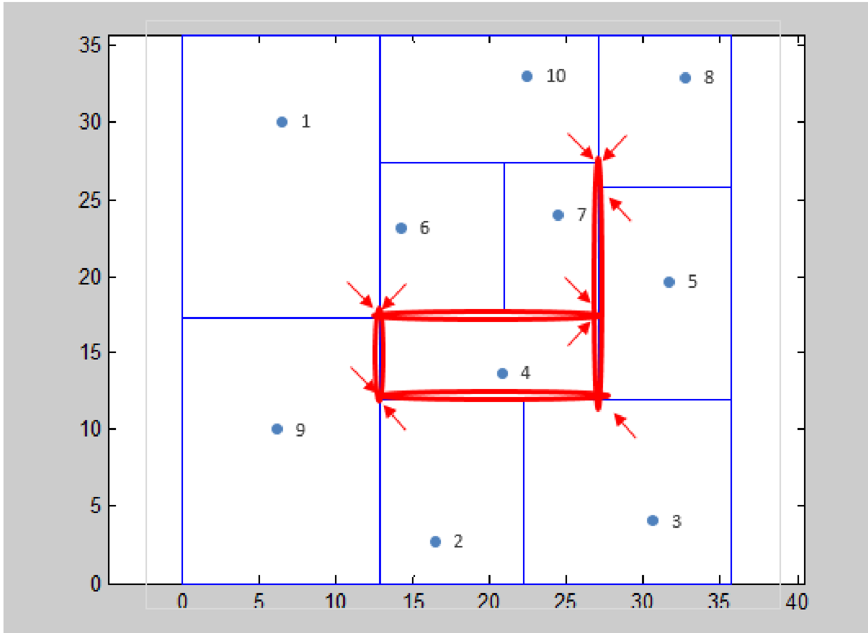


Figure 4.1 c) Detailed block layout

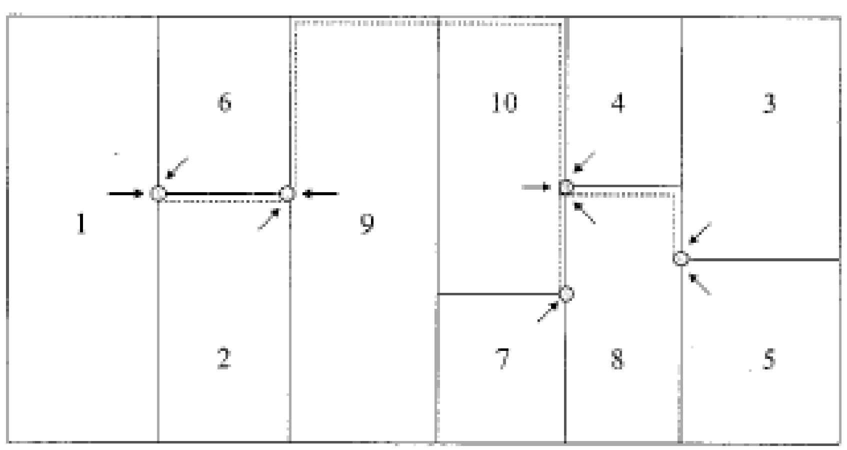


Figure 4.1 d) Comparison layout (Arapoglu et al., 2001)

Figure 4.1: Outputs from each stage of design framework for test problem 2.

CHAPTER 5. CONCLUSION

5.1 Conclusion

The ability to directly apply the progress made in academic research to a real world facility layout problem (FLP) is highly dependent on the experience of the designer. There are a variety of reasons why a designer may not apply one of the analytical methods developed over the past 60+ years, but one of the most obvious and easy to fix is the lack familiarity with said methods. The most commonly used framework for addressing FLP's, Muther's Systematic Layout Planning (SLP), only guides designers though a process of gathering potential inputs and a loose evaluation methodology. It does not take into account that different designers will have different and evolving goals as they go through the design process. It also does not suggest any particular analytical methods to use in developing design alternatives, nor does it even give characteristics of potentially good models.

In order to address this issue a new integrated design framework was developed that;

1) Categorizes potential design goals
2) Guides the designer though an assessment of details relevant to a particular goal
3) Helps the designer recognize how those details interact with each other
4) Characterizes what a requisite model to the design goals would include
5) Highlights tradeoffs between computational and design performance
6) Directs the designer to a set of requisite models and methods for a majority of design goals

The concept of design layers was introduced as a way for designers to think through potential inputs to their design problem. There is a minimum set of basic facility details that a designer most know before they can begin to solve a FLP. This minimum set is the foundational layer that any model or analysis method is built on. Once that minimum set of details is ascertained the designer must then choose an evaluation measure to use in his/her analysis. Depending on the choice of evaluation measure, the designer may then need to add more layers of detail in order to select or use a requisite model that captures the information relevant to his/her basic facility details and chosen evaluation measure.

A set of models and methodologies was also proposed that is flexible enough to allow the designer to characterize any distance or adjacency based evaluation measure mathematically and rapidly solve for a candidate design solution. These methods can be used in sequence to go from the minimum design inputs of department areas and flow volumes to a fully detailed block layout, or used individually to meet a particular design goal.

5.2 Future Opportunities

Future work to improve this framework includes automating the identification of requisite inputs for a given design goal, and then automatically developing any required constraint or objective functions and inputting them directly into an analytical solver. This will allow designers who are unfamiliar computational tools, programming languages, or analytical modeling to directly apply these methods without requiring them to understand how they function. Given the sensitivity of total *flow*distance* scores to the block layouts they are associated with, work done to determine the ideal aspect ratio to use in order to develop compact bubble layouts and not exclude potential good departmental arrangements would be highly beneficial.

REFERENCES

REFERENCES

Alagoz, O., Norman, B., & Smith, A. E. (2008). Determining aisle structures for facility designs using a hierarchy of algorithms. *IIE Transactions*, *40*(11), 1019–1031.

Anjos, M. F., & Vannelli, A. (2002). An Attractor-Repeller approach to floorplanning. *Mathematical Methods of Operations Research (ZOR)*, *56*(1), 3–27.

Arapoglu, R. A., Norman, B. A., & Smith, A. E. (2001). Locating Input and Output Points in Facilities Design — A Comparison of Constructive , Evolutionary , and Exact Methods, *5*(3), 192–203.

Armour, G. C., & Buffa, E. S. (1963). A Heuristic Algorithm and Simulation Approach to Relative Location of Facilities. *Management Science*, *9*(2), 294–309.

Benson, B., & Foote, B. L. (1997). DoorFAST: A constructive procedure to optimally layout a facility including aisles and door locations basedon anaisle flow distance metric. *International Journal of Production Research*, *35*(7), 1825–1842.

Blanks, J. P. (1985). Near-Optimal Quadratic-Based Placement for a Class of IC Layout Problems. *Circuits and Devices Magazine, IEEE*, (September), 31–37.

Brown, A. (1971). Optimal Packing and Depletion. *Jeffreys & Hill Publ*.

Castillo, I., & Sim, T. (2004). A spring-embedding approach for the facility layout problem. *Journal of the Operational Research Society*, *55*(1), 73–81.

Chhajed, D., Montreuil, B., & Lowe, T. J. (1992). Flow network design for manufacturing systems layout. *European Journal of Operational Research*, *57*(2), 145–161.

Chittratanawat, S. (1999). An integrated approach for facility layout, P/D location and material handling system design. *International Journal of Production Research*, *37*(3), 683–706.

Drezner, Z. (1980). DISCON: A New Method for the Layout Problem. *Operations Research*, *28*(6), 1375–1384.

Drira, A., Pierreval, H., & Hajri-Gabouj, S. (2007). Facility layout problems: A survey. *Annual Reviews in Control*, *31*(2), 255–267.

Egbelu, P. J., & Tanchoco, J. M. a. (1986). Potentials for bi-directional guide-path for automated guided vehicle based systems. *International Journal of Production Research*, *24*(5), 1075–1097.

Floyd, R. W. (1962). Algorithms. *Communications of the ACM*, *5*(6), 344–348.

Foulds, L. R. (1992). *Graph Theory Applications* (p. 385). Springer.

Gaskins, R. J., & Tanchoco, J. M. a. (1987). Flow path design for automated guided vehicle systems. *International Journal of Production Research*, *25*(5), 667–676.

Heragu, S. S., & Kusiak, A. (1991). Efficient models for the facility layout problem. *European Journal of Operations Research*, *53*(1), 1–13.

Ho, Y.-C., & Moodie, C. L. (2000). A hybrid approach for concurrent layout design of cells and their flow paths in a tree configuration. *International Journal of Production Research*, *38*(4), 895–928.

Huesman, J., Holland, L., & Langley, T. (2014). *U.S. Census Bureau News* (Vol. 3, pp. 0–4).

Johnson, D. B. (1977). Efficient Algorithms for Shortest Paths in Sparse Networks. *Journal of the ACM*, *24*(1), 1–13.

Katz, G. J., & Kider, J. T. J. (2008). All-Pairs Shortest-Paths for Large Graphs on the GPU. *Proceedings of the 23rd ACM SIGGRAPH/EUROGRAPHICS Symposium on Graphics Hardware*, 47–55.

Kim, J. G., & Goetschalckx, M. (2005). An integrated approach for the concurrent determination of the block layout and the input and output point locations based on the contour distance. *International Journal of Production Research*, *43*(10), 2027–2047.

Kim, J., & Klein, C. M. (1996). Location of departmental pickup and delivery points for an AGV system. *International Journal of Production Research*, *34*(2), 407–420.

Kim, J.-G., & Kim, Y.-D. (2000). Layout planning for facilities with fixed shapes and input and output points. *International Journal of Production Research*, *38*(18), 4635–4653.

Koopmans, T., & Beckmann, M. (1957). Assignment Problems and the Location of Economic Activities. *Econometrica: Journal of the Econometric Society*, *25*(1), 53–76.

Kusiak, A., & Heragu, S. S. (1987). The facility layout problem. *European Journal of Operational Research*, *29*(3), 229–251.

Maxwell, W. L., & Muckstadt, J. a. (1982). Design of Automatic Guided Vehicle Systems. *IIE Transactions*, *14*(2), 114–124.

Meller, R. D., & Gau, K.-Y. (1996). The facility layout problem: Recent and emerging trends and perspectives. *Journal of Manufacturing Systems*, *15*(5), 351–366.

Meller, R. D., Kirkizoglu, Z., & Chen, W. (2010). A new optimization model to support a bottom-up approach to facility design. *Computers & Operations Research*, *37*(1), 42–49.

Meller, R. D., Kleiner, B., & Nussbaum, M. (2004). The facility layout problem: a new model to support a bottom-up approach to facility design. *Progress in Material Handling Research*.

Meller, R. D., Narayanan, V., & Vance, P. H. (1999). Optimal facility layout design. *Operations Research Letters*, *23*, 117–127.

Montreuil, B. (1990). A Modeling Framework for Integrating Layout Design and Flow Network Design. In *Proceedings of the Material Handling Research Colloquium* (pp. 43–58).

Montreuil, B., & Ratliff, H. D. (1988). Optimizing the location of input/output stations within facilities layout. *Engineering Costs and Production Economics*, *14*(3), 177–187.

Muther, R. (1973). *Systematic layout planning* (p. 360). Cahners Books.

Nugent, C. E., Vollmann, T. E., & Ruml, J. (1968). An experimental comparison of techniques for the assignment of facilities to locations. *Operations Research*, *16*(1), 150–173.

Owens, R. (2011). Advancing facility planning. *IIE Transactions*, *43*(11), 45–49.

Peters, B. A., & Yang, T. (1997). Material Handling System Design in Semiconductor Fabrication Facilities. *IEEE Transactions on Semiconductor Manufacturing*, *10*(3), 360–369.

Phillips, L. D. (1984). A theory of requisite decision models. *Acta Psychologica*, *56*(1-3), 29–48.

Sahni, S., & Gonzalez, T. (1976). P-Complete Approximation Problems. *Journal of the ACM*, *23*(3), 555–565.

Schneider, M. (1960). Cross charting techniques as a basis for plant layout. *Journal of Industrial Engineering, XI*.

Sinriech, D., & Tanchoco, J. M. A. (1992). The Centroid Projection Method for Locating Pick-Up and Delivery Stations in Single-Loop AGV Systems. *Journal of Manufacturing Systems*, *11*(4), 297–307.

Sly, D. (1995). Computerized facilities design and management : an overview. *IIE Solutions*, *27*(8), 43.

Smith, J. M. (2005). Dilemmas in factory design: paradox and paradigm. *OR Spectrum*, *27*(2-3), 171–193.

Tam, K. Y. (1992). A Simulated Annealing Algorithm for Allocating Space to Manufacturing Cells. *International Journal of Production Research*, *30*(1), 63–87.

Tompkins, J. A. (2010). *Facilities Planning* (p. 854). John Wiley & Sons.

Van Camp, D. J. Van, Carter, M. W., & Vannelli, A. (1991). A nonlinear optimization approach for solving facility layout problems, *57*, 174–189.

Warnecke, H., Dangelmaier, W., & Kuhnle, H. (1985). Computer-aided layout planning. *Material Flow*, *1*(1), 35–48.

Warnecke, H. J., & Dangelmaier, W. (1984). Progress in Computer Aided Plant Layout. *CIRP Annals-Manufacturing Technology*, *33*(1), 321–326.

Yang, T., & Peters, B. a. (1998). Flexible machine layout design for dynamic and uncertain production environments. *European Journal of Operational Research*, *108*(1), 49–64.

APPENDICES

Appendix A Test Problem Flow Matrices and Department Areas

Table A.1: Flow matrix and department areas from (van Camp, Carter, & Vannelli, 1991)

Test Problems 1 and 2	1	2	3	4	5	6	7	8	9	10
1	-	0	0	0	0	218	0	0	0	0
2	0	-	0	0	0	148	0	0	296	0
3	0	0	-	28	70	0	0	0	0	0
4	0	0	0	-	0	28	70	140	0	0
5	0	0	0	0	-	0	0	210	0	0
6	0	0	0	0	0	-	0	0	0	0
7	0	0	0	0	0	0	-	0	0	28
8	0	0	0	0	0	0	0	-	0	888
9	0	0	0	0	0	0	0	0	-	59.2
10	0	0	0	0	0	0	0	0	0	-
Department Areas: Equal	128	128	128	128	128	128	128	128	128	128
Department Areas: Unequal	238	112	160	80	120	80	60	85	221	119

Table A.2: Flow matrix and department areas from (Yang & Peters, 1998)

Test Problems 3 and 4	1	2	3	4	5	6	7	8	9	10	11	12
1	-	0	0	0	0	0	18	0	0	0	0	0
2	0	-	18	10	4	26	34	32	16	0	18	32
3	0	0	-	0	0	0	0	0	16	0	0	0
4	0	0	0	-	8	38	2	34	30	2	30	0
5	0	0	0	0	-	10	0	0	0	0	0	0
6	0	0	0	0	10	-	22	26	26	14	26	12
7	0	0	0	0	0	0	-	0	0	0	4	0
8	0	0	0	0	0	0	0	-	22	38	6	20
9	0	0	0	0	0	0	0	0	-	0	0	12
10	0	0	0	0	0	0	0	0	0	-	30	6
11	0	0	0	0	0	0	0	0	0	0	-	6
12	0	0	0	0	0	0	0	0	0	0	0	-
Department Areas: Equal	30.5	30.5	30.5	30.5	30.5	30.5	30.5	30.5	30.5	30.5	30.5	30.5
Department Areas: Unequal	20	35	30	16	36	20	70	35	30	25	25	24

Table A.3: Flow matrix and department areas from (Yang & Peters, 1998)

Test Problems 5 and 6	1	2	3	4	5	6	7	8	9	10	11	12
1	-	288	18	54	72	180	27	72	36	0	0	9
2	0	-	240	54	72	24	48	160	16	64	8	16
3	0	0	-	120	80	0	60	120	60	0	0	30
4	0	0	0	-	72	18	18	48	24	48	12	0
5	0	0	0	0	-	12	12	64	16	16	4	8
6	0	0	0	0	0	-	18	24	6	12	3	3
7	0	0	0	0	0	0	-	0	6	6	3	6
8	0	0	0	0	0	0	0	-	16	16	16	4
9	0	0	0	0	0	0	0	0	-	4	4	2
10	0	0	0	0	0	0	0	0	0	-	2	2
11	0	0	0	0	0	0	0	0	0	0	-	2
12	0	0	0	0	0	0	0	0	0	0	0	-
Department Areas: Equal	4.5	4.5	4.5	4.5	4.5	4.5	4.5	4.5	4.5	4.5	4.5	4.5
Department Areas: Unequal	9	8	10	6	4	3	3	4	2	2	1	1

Table A.4: Flow matrix and department areas from (Armour & Buffa, 1963)

Test Problems 7 and 8	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	1.8	1.2	0	0	0	0	0	0	1.04	1.12	0	0	1.2	0	0	0	0	0	0
2	1.8	0	0.96	24.5	0.78	0	14	0	1.2	1.35	0	0	0	0	0	0	0	0	6.9	0
3	1.2	0.96	0	0	0	2.21	0	0	3.15	3.9	0	0	0	13.1	0	0	0	0	13.7	0
4	0	24.5	0	0	1.08	5.7	7.5	0	2.34	0	0	1.4	0	0	0	0	0	1.5	15.8	0
5	0	0.78	0	1.08	0	0	2.25	1.35	0	1.56	0	0	0	0	1.35	0	0	0	0	0
6	0	0	2.21	5.7	0	0	6.15	0	0	0	0	0.45	0	0	0	0	0	1.05	0	0
7	0	14	0	7.5	2.25	6.15	0	24	0	1.87	0	0	0	0.96	0	0	0	1.65	0	3.75
8	0	0	0	0	1.35	0	24	0	0	0	0	0	0.6	0	0	0	0	0	7.5	33.5
9	0	1.2	3.15	2.34	0	0	0	0	0	0	0	0	0	7.5	0	0	7.5	0	0	0
10	1.04	1.35	3.9	0	1.56	0	1.87	0	0	0	0.36	12	0	18.6	1.92	0	0	0	5.25	0
11	1.12	0	0	0	0	0	0	0	0	0.36	0	2.25	0	3	0.96	22.5	0	0	0	0
12	0	0	0	1.4	0	0.45	0	0	0	12	2.25	0	0	0	1.65	0	15	0	8.4	0
13	0	0	0	0	0	0	0	0.6	0	0	0	0	0	8	1.04	6	0	0	0	0
14	1.2	0	13.1	0	0	0	0.96	0	7.5	18.6	3	0	8	0	9.75	0	0	0.9	0	0
15	0	0	0	0	1.35	0	0	0	0	1.92	0.96	1.65	1.04	9.75	0	0	5.25	0	0	0
16	0	0	0	0	0	0	0	0	0	0	22.5	0	6	0	0	0	12	0	0	0
17	0	0	0	0	0	0	0	0	7.5	0	0	15	0	0	5.25	12	0	0	7.5	0
18	0	0	0	1.5	0	1.05	1.65	0	0	0	0	0	0	0.9	0	0	0	0	4.65	0
19	0	6.9	13.7	15.8	0	0	0	7.5	0	5.25	0	8.4	0	0	0	0	7.5	4.65	0	0
20	0	0	0	0	0	0	3.75	33.5	0	0	0	0	0	0	0	0	0	0	0	0
Department Areas: Equal	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3
Department Areas: Unequal	0.27	0.18	0.27	0.18	0.18	0.18	0.09	0.09	0.09	0.24	0.6	0.42	0.18	0.24	0.27	0.75	0.64	0.41	0.27	0.45

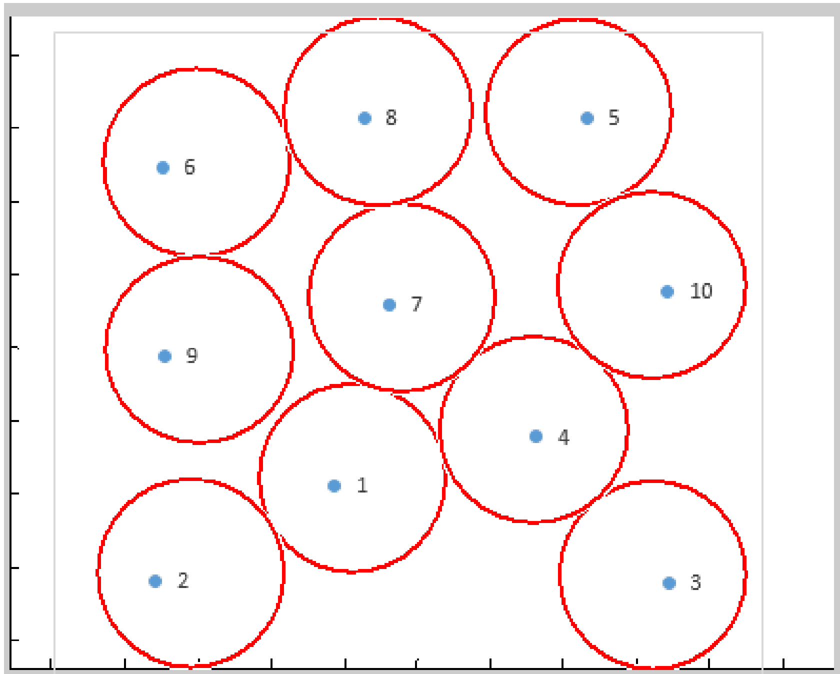
Table A.5: Flow matrix from (Nugent, Vollmann, & Ruml, 1968), department areas from (Tam, 1992)

Test Problems 9 and 10	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	-	3	2	0	0	2	10	5	0	5	2	5	0	0	2	0	5	6	3	0	1	10	0	10	2	1	1	1	0	1
2	0	-	4	0	10	4	0	0	2	2	1	0	5	0	0	0	0	2	0	1	6	1	0	1	2	2	5	1	10	5
3	0	0	-	3	4	0	5	5	5	1	4	1	0	4	0	4	0	6	3	2	5	5	2	1	0	0	3	1	0	2
4	0	0	0	-	0	0	0	2	2	0	6	0	2	5	2	5	1	1	1	1	2	2	4	0	2	0	2	2	5	5
5	0	0	0	0	-	5	2	0	0	0	0	2	0	0	0	0	2	1	0	0	2	0	5	1	0	2	1	0	2	1
6	0	0	0	0	0	-	1	2	2	1	4	10	10	2	5	5	0	5	0	0	0	10	0	0	0	4	0	10	1	1
7	0	0	0	0	0	0	-	10	10	5	10	10	6	0	0	10	2	1	10	1	5	5	2	3	5	0	2	0	1	3
8	0	0	0	0	0	0	0	-	1	3	5	0	0	0	2	4	5	2	10	6	0	5	5	2	5	0	5	5	0	2
9	0	0	0	0	0	0	0	0	-	10	2	1	5	2	0	3	0	2	0	0	4	0	5	2	0	5	2	2	5	2
10	0	0	0	0	0	0	0	0	0	-	5	5	6	0	1	5	5	0	5	2	3	5	0	5	2	10	10	1	5	2
11	0	0	0	0	0	0	0	0	0	0	-	0	0	1	2	1	0	2	0	0	0	6	6	0	4	5	3	2	2	10
12	0	0	0	0	0	0	0	0	0	0	0	-	5	5	2	0	0	0	0	2	0	4	5	10	1	0	0	0	0	1
13	0	0	0	0	0	0	0	0	0	0	0	0	-	2	0	4	2	2	1	0	6	2	1	5	5	0	0	1	5	5
14	0	0	0	0	0	0	0	0	0	0	0	0	0	-	2	1	0	5	3	10	0	0	4	2	0	0	4	2	5	5
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	4	5	1	0	1	0	5	0	2	0	0	5	1	1	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	3	0	2	2	0	2	0	5	0	5	2	5	10
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	2	2	0	0	0	6	5	3	5	0	0	5	1
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	5	1	2	10	10	4	0	0	5	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	5	5	1	0	5	2	1	2	10	10
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	5	2	1	3	1	5	6	5	5	3
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	4	0	1	0	0	0	5	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	5	0	4	4	5	0	2	5
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	4	4	1	0	2	2
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	5	5	0	1	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	1	0	10	1	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	10
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	2	2
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	2
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
Department Areas: Equal	100	80	50	60	120	40	20	40	150	120	50	10	20	30	50	20	40	20	80	100	40	50	80	10	40	10	40	10	80	40
Department Areas: Unequal	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5	53.5

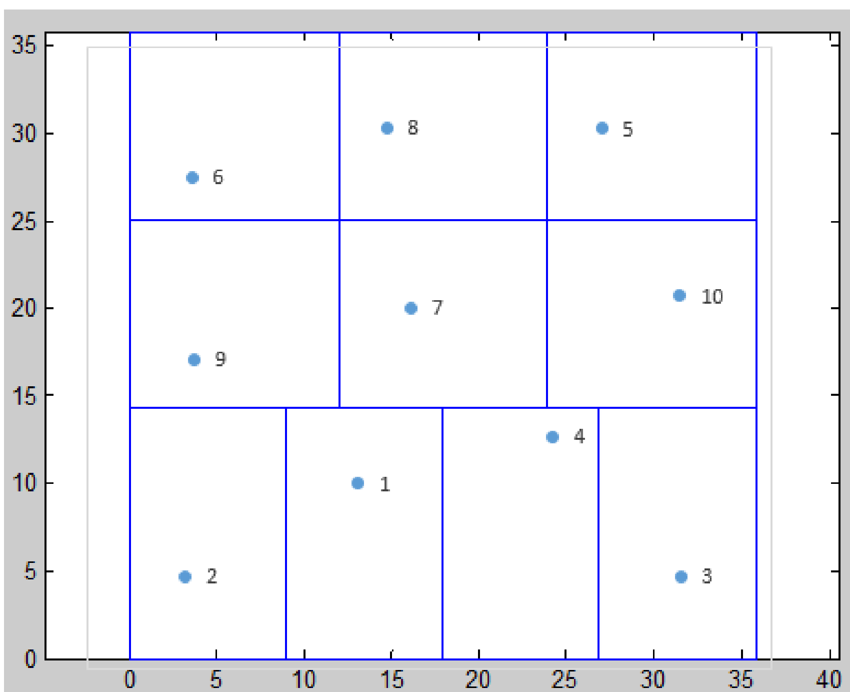
Table A.6: Flow matrix from (Nugent et al., 1968), department areas from (Tam, 1992)

Test Problems 11 and 12	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	-	0	2	0	5	2	10	3	1	5	5	5	0	0	5	4	4	0	0	1
2	0	-	3	10	5	1	5	1	2	4	2	5	0	10	10	3	0	5	10	5
3	0	0	-	2	0	5	2	4	4	5	0	0	0	5	1	0	0	5	0	0
4	0	0	0	-	1	0	5	2	1	0	10	2	2	0	2	1	5	2	5	5
5	0	0	0	0	-	5	6	5	2	5	2	0	5	1	1	1	5	2	5	1
6	0	0	0	0	0	-	5	2	1	6	0	0	10	0	2	0	1	0	1	5
7	0	0	0	0	0	0	-	0	0	0	5	10	2	2	5	1	2	1	0	10
8	0	0	0	0	0	0	0	-	1	1	10	10	2	0	10	2	5	2	2	10
9	0	0	0	0	0	0	0	0	-	2	0	3	5	5	0	5	0	0	0	2
10	0	0	0	0	0	0	0	0	0	-	5	5	0	5	1	0	0	5	5	2
11	0	0	0	0	0	0	0	0	0	0	-	5	2	5	1	10	0	2	2	5
12	0	0	0	0	0	0	0	0	0	0	0	-	2	10	5	0	1	1	2	5
13	0	0	0	0	0	0	0	0	0	0	0	0	-	2	2	1	0	0	0	5
14	0	0	0	0	0	0	0	0	0	0	0	0	0	-	5	5	1	5	5	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	3	0	5	10	10
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	2	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	5	2	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	1	1
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	6
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-
Department Areas: Equal	60	60	60	60	60	60	60	60	60	60	60	60	60	60	60	60	60	60	60	60
Department Areas: Unequal	100	80	50	60	120	40	20	40	150	120	50	10	20	30	50	20	40	20	80	100

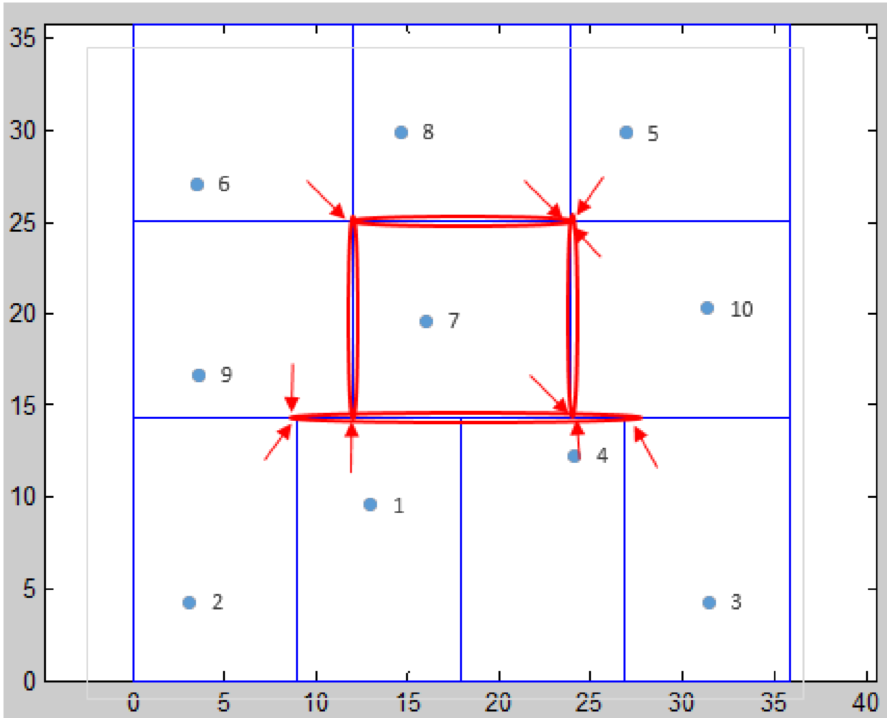
Appendix B Graphics of outputs from each stage of new framework



B.1.1 Bubble layout

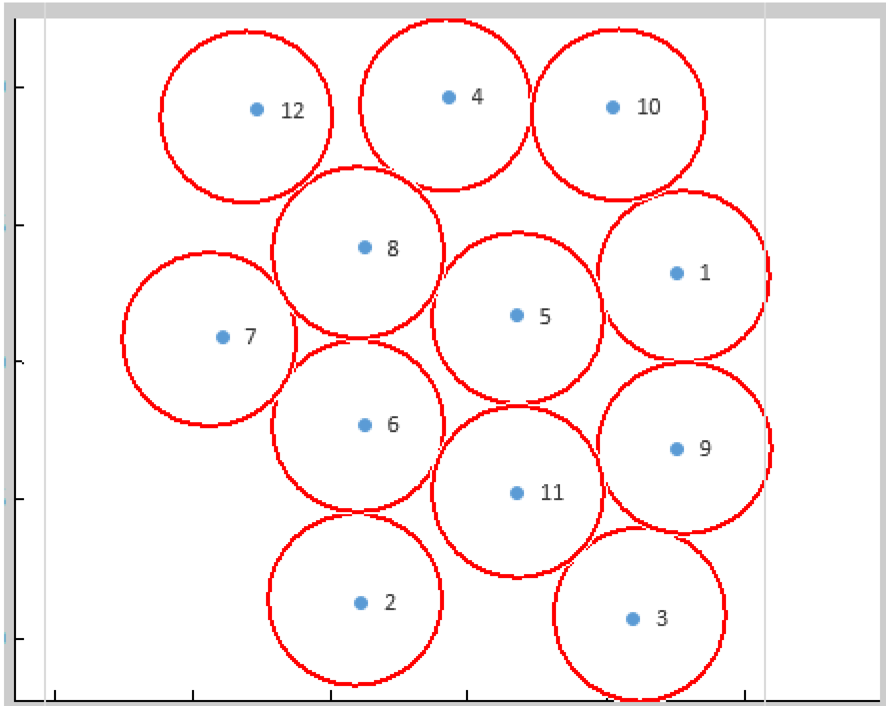


B.1.2 Simple block layout

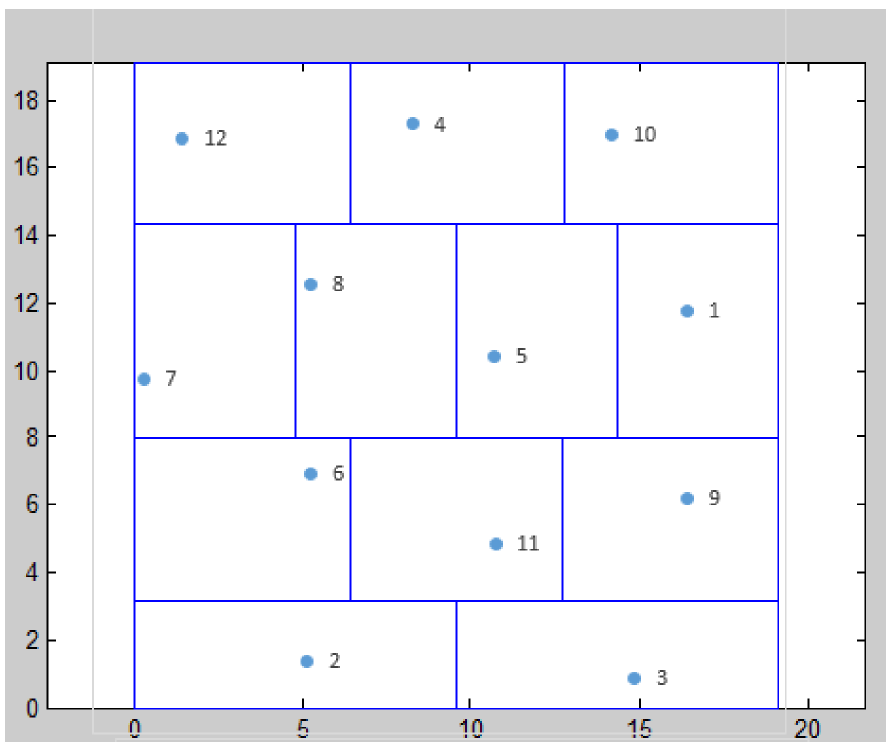


B.1.3 Detailed block layout

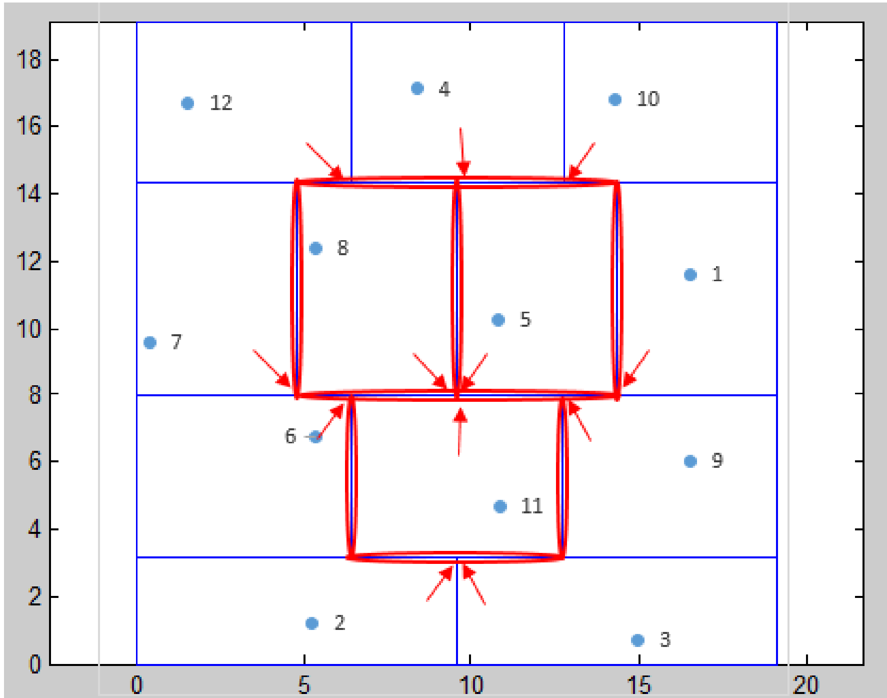
Figure B.1: Outputs from each stage of design framework for test problem 1.



B.3.1 Bubble layout

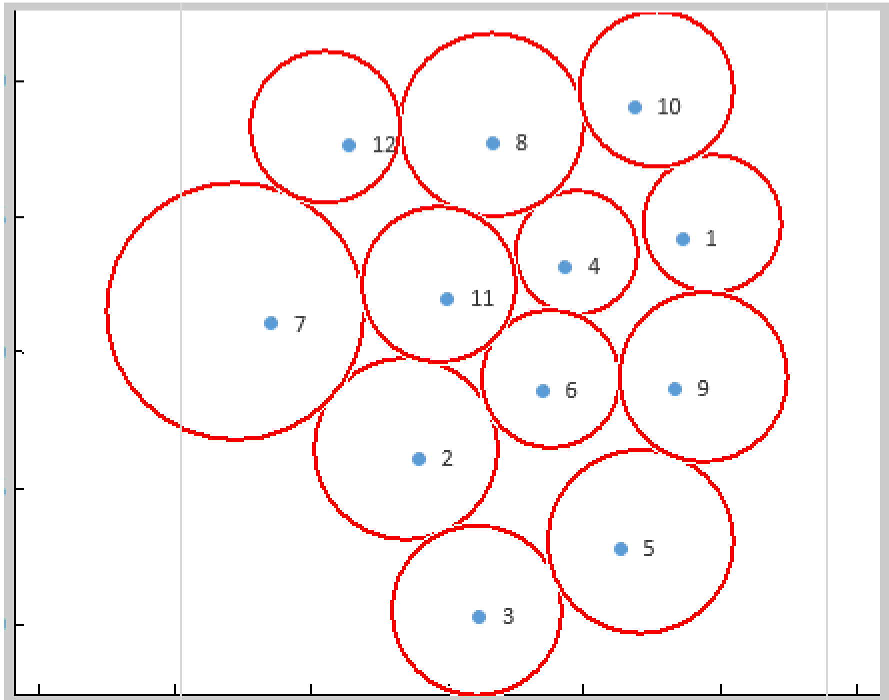


B.3.2 Simple block layout

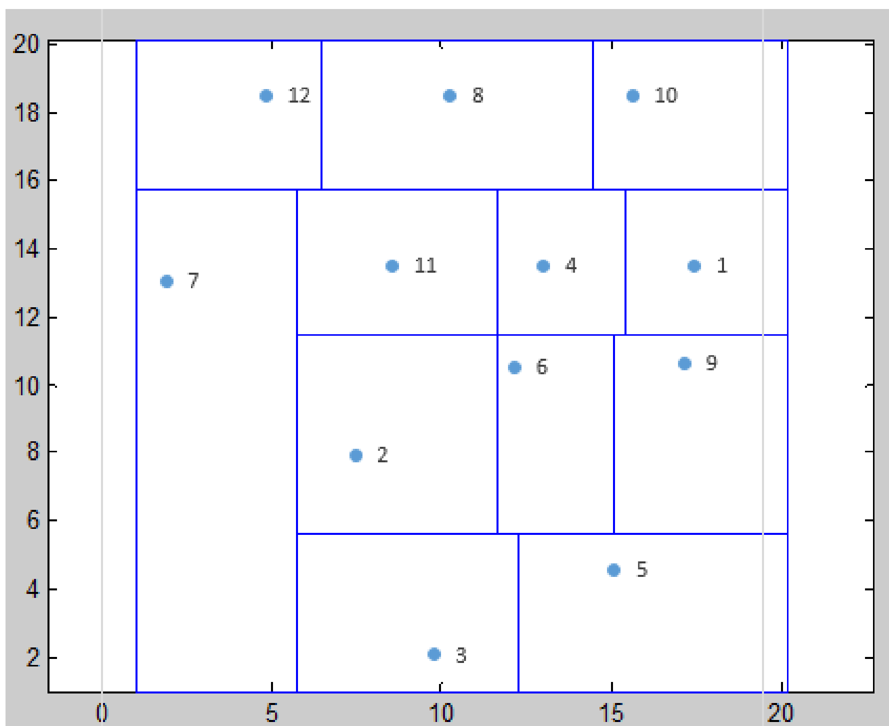


B.3.3 Detailed block layout

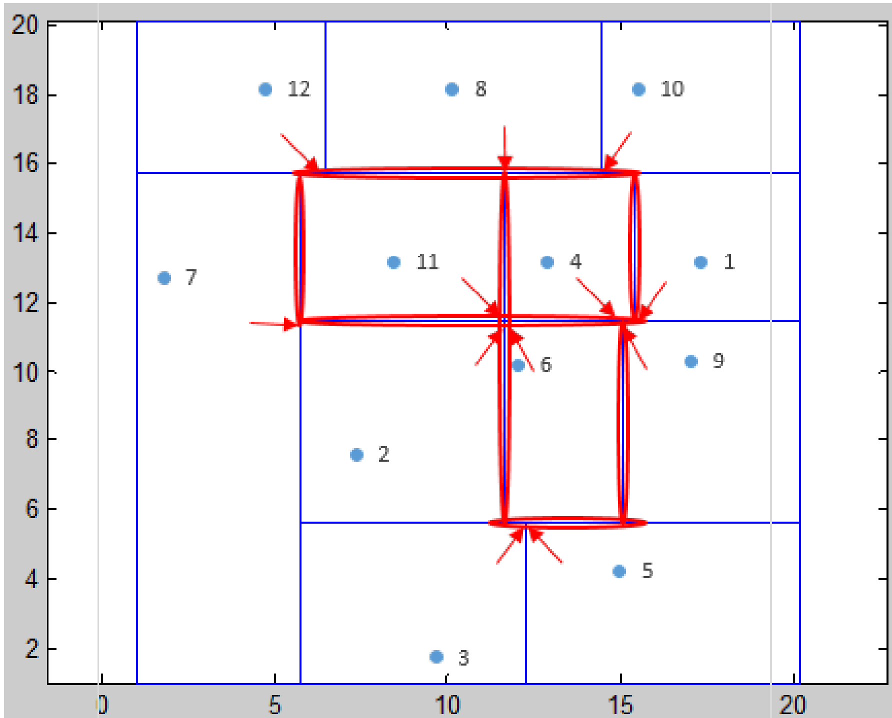
Figure B.3: Outputs from each stage of design framework for test problem 3.



B.4.1 Bubble layout

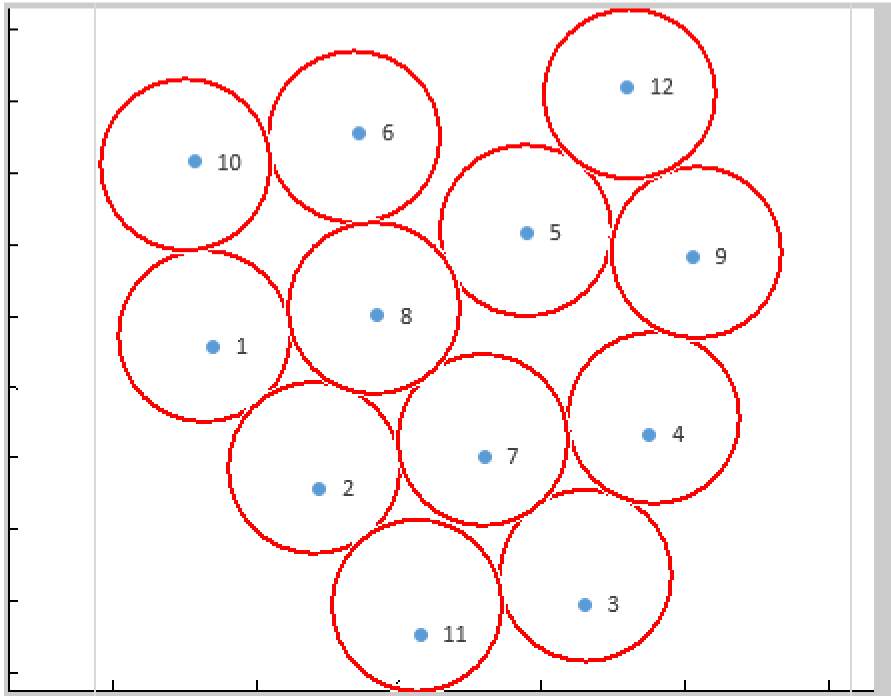


B.4.2 Simple block layout

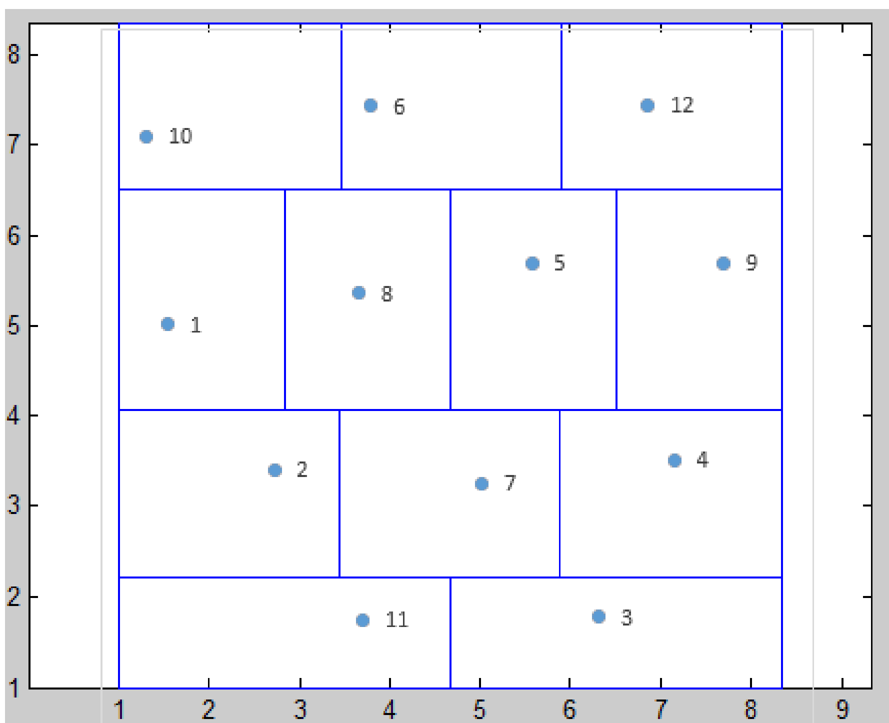


B.4.3 Detailed block layout

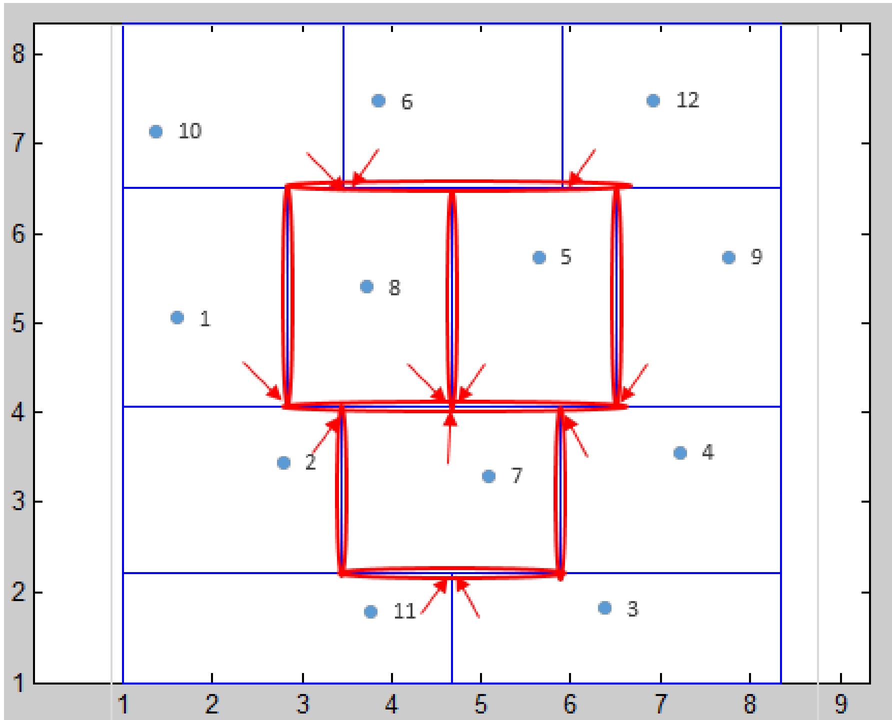
Figure B.4: Outputs from each stage of design framework for test problem 4.



B.5.1 Bubble layout

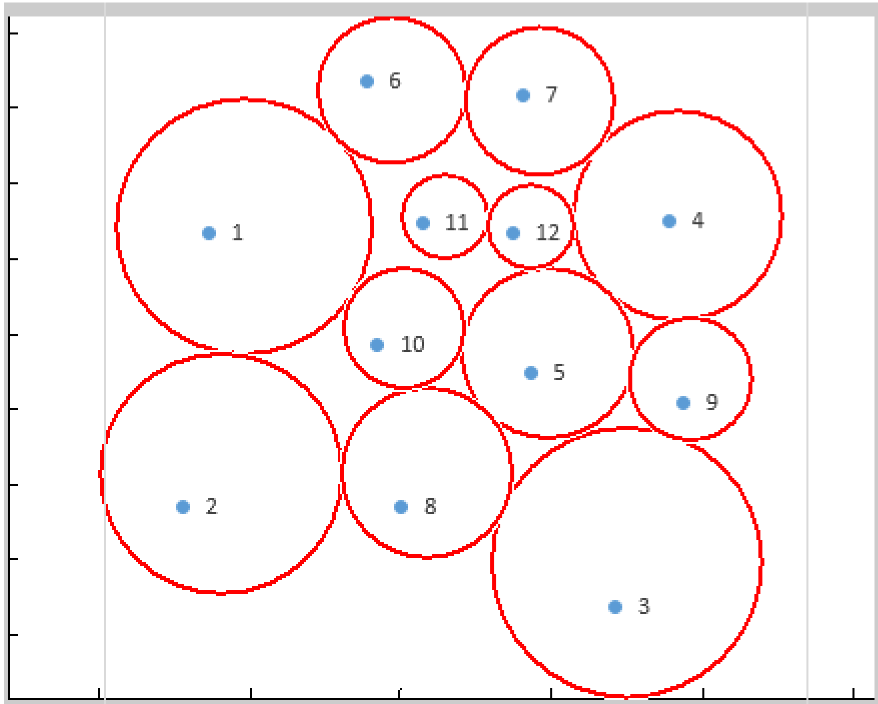


B.5.2 Simple block layout

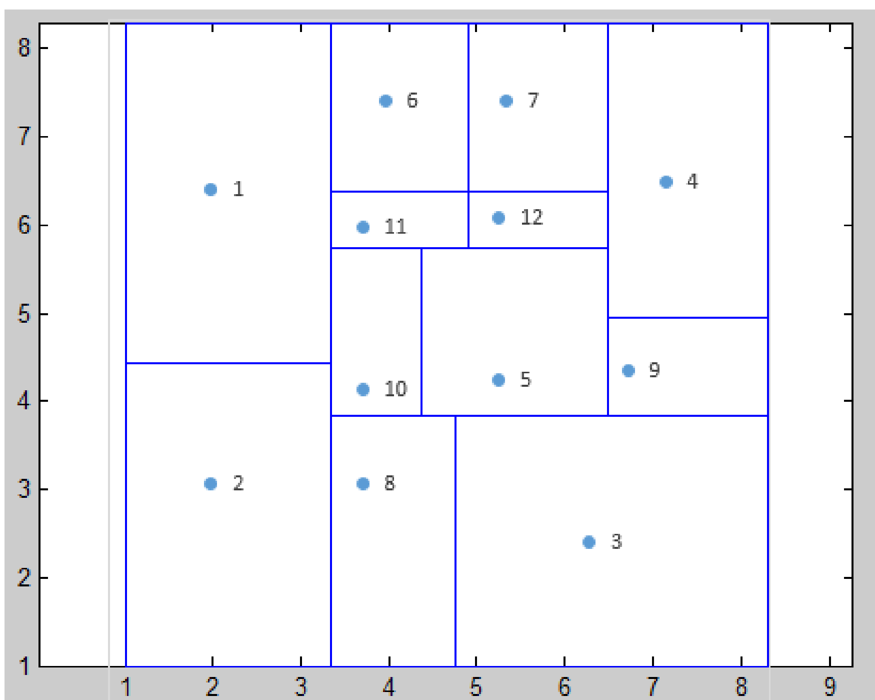


B.5.3 Detailed block layout

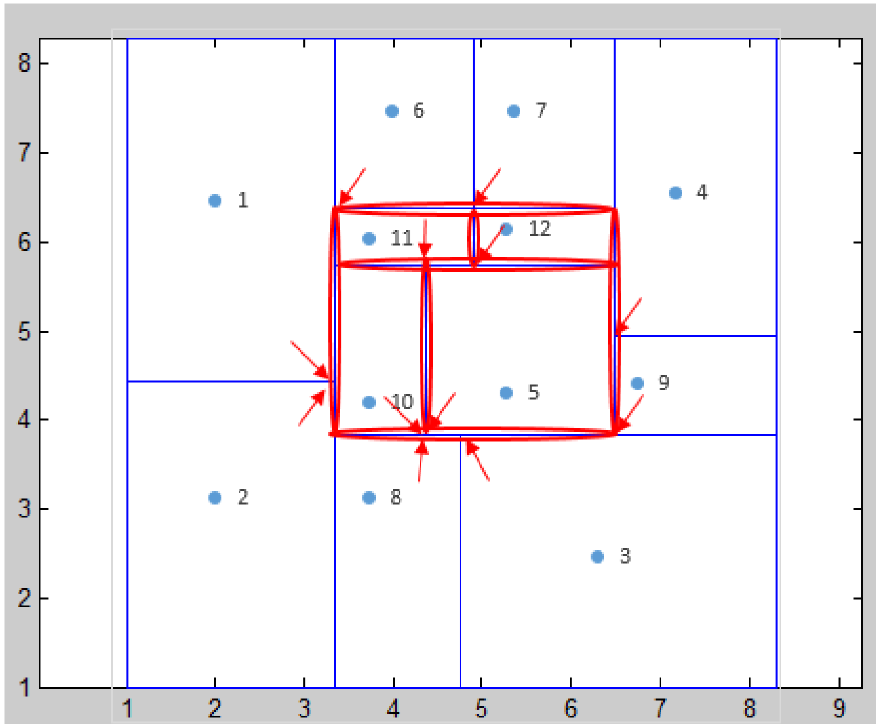
Figure B.5: Outputs from each stage of design framework for test problem 5.



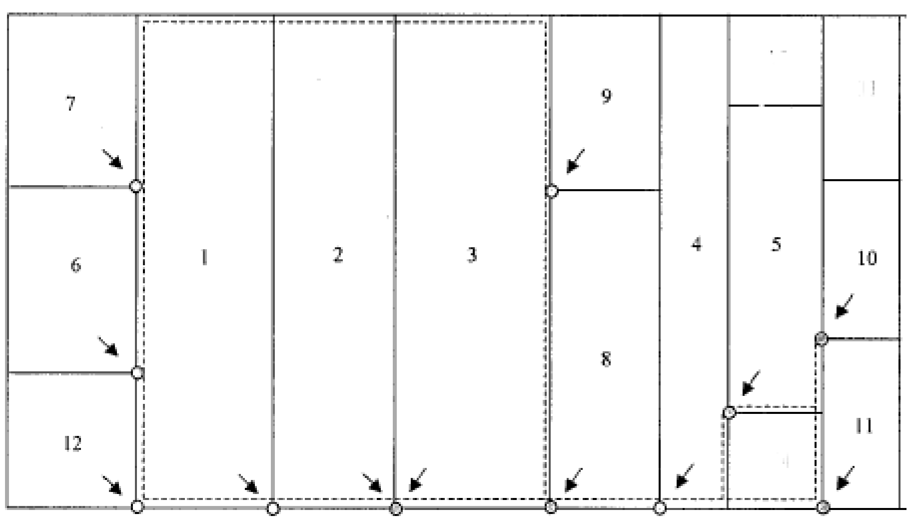
B.6.1 Bubble layout



B.6.2 Simple block layout

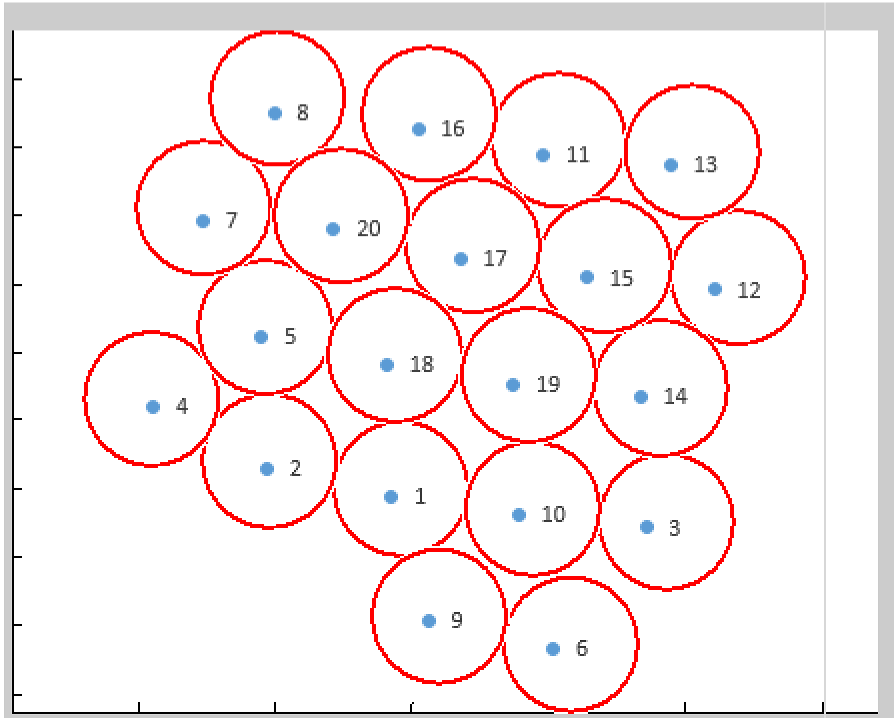


B.6.3 Detailed block layout

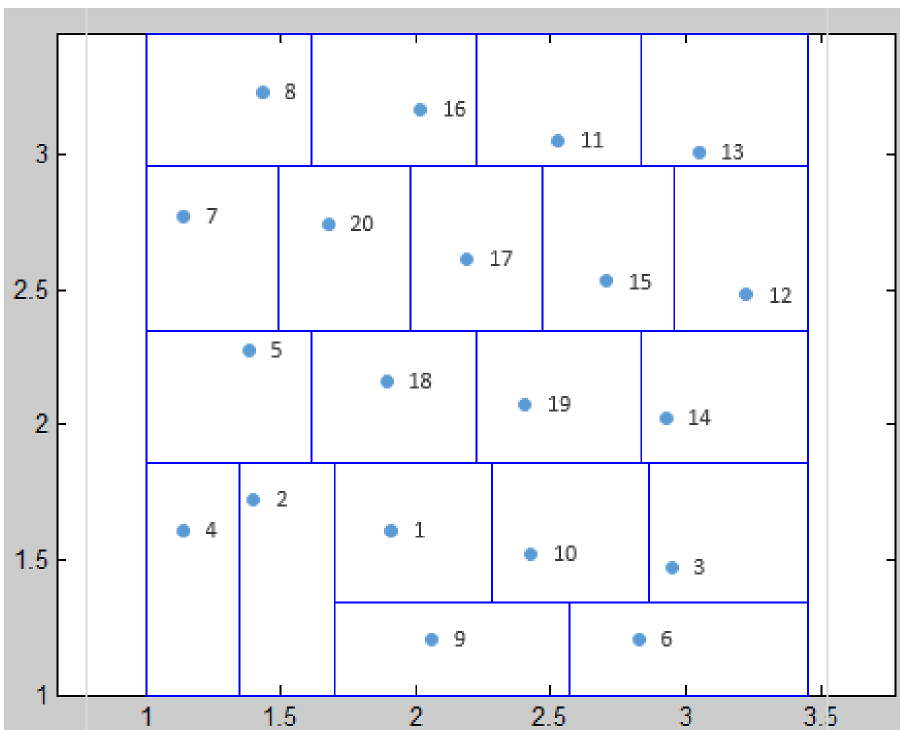


B.6.4 Comparison layout (Arapoglu et al., 2001)

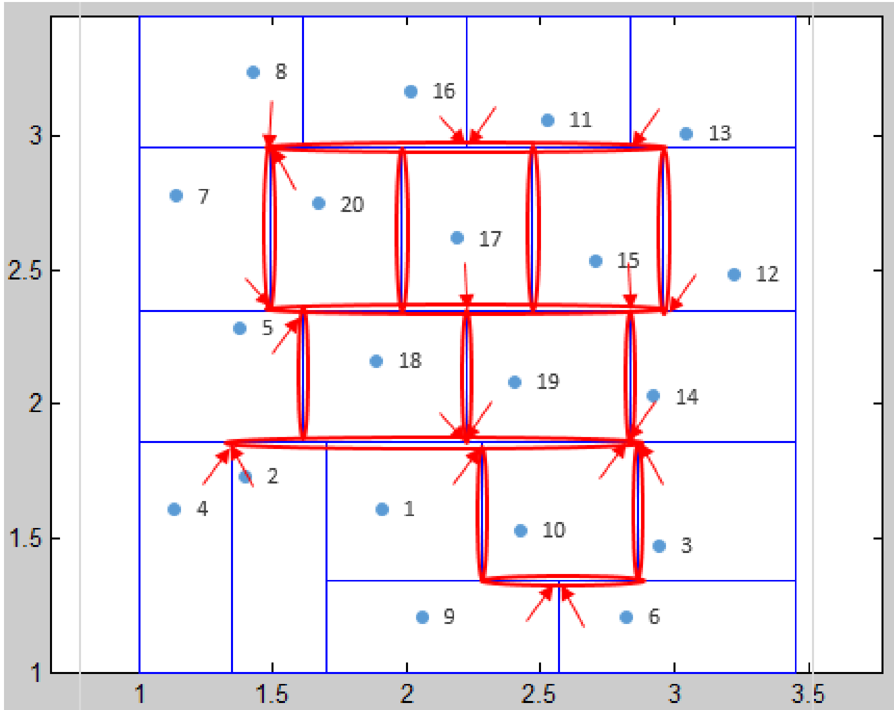
Figure B.5: Outputs from each stage of design framework for test problem 6.



B.7.1 Bubble layout

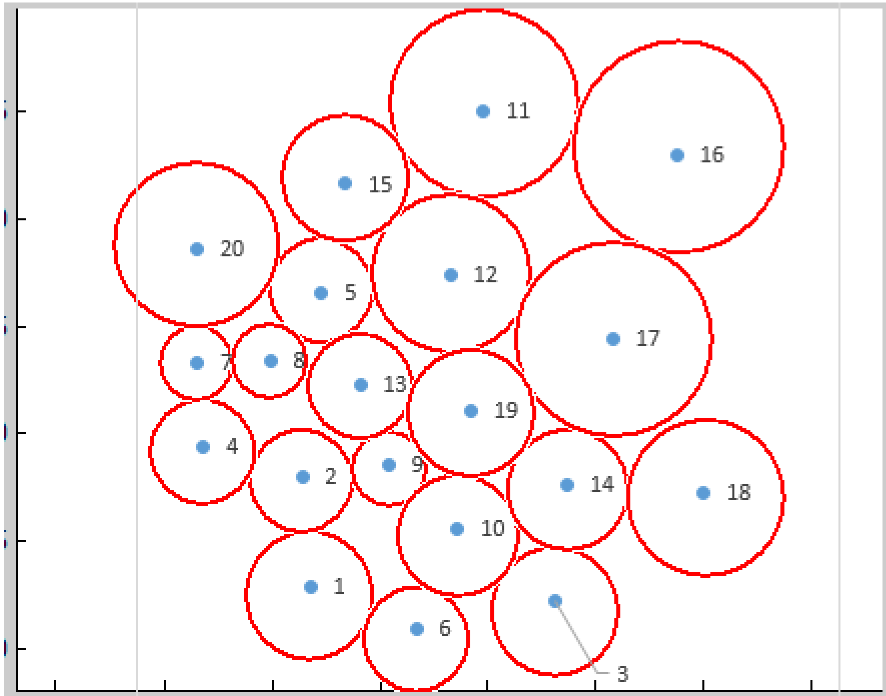


B.7.2 Simple block layout

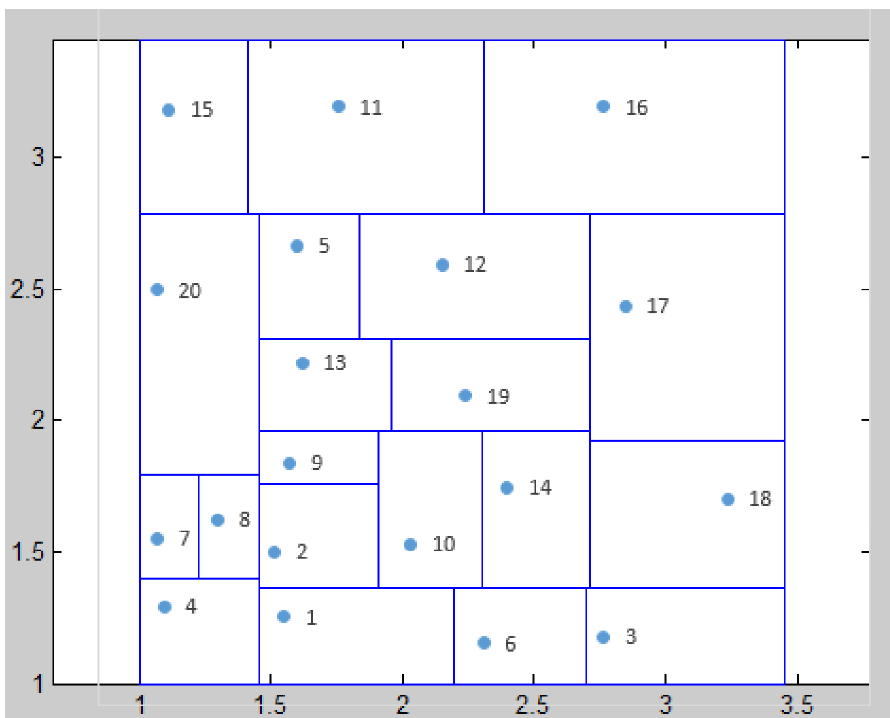


B.7.3 Detailed block layout

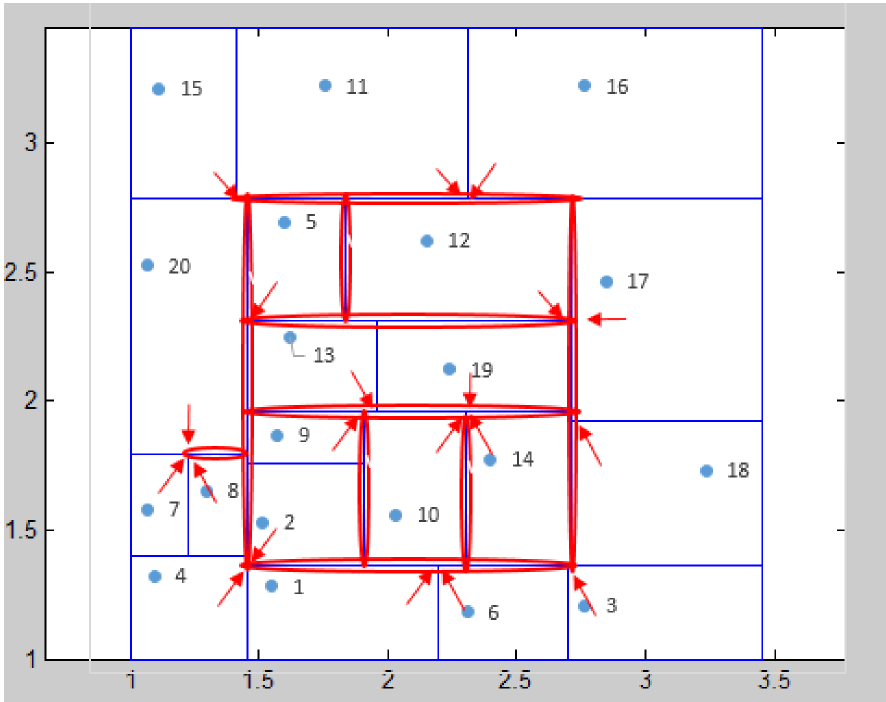
Figure B.7: Outputs from each stage of design framework for test problem 7.



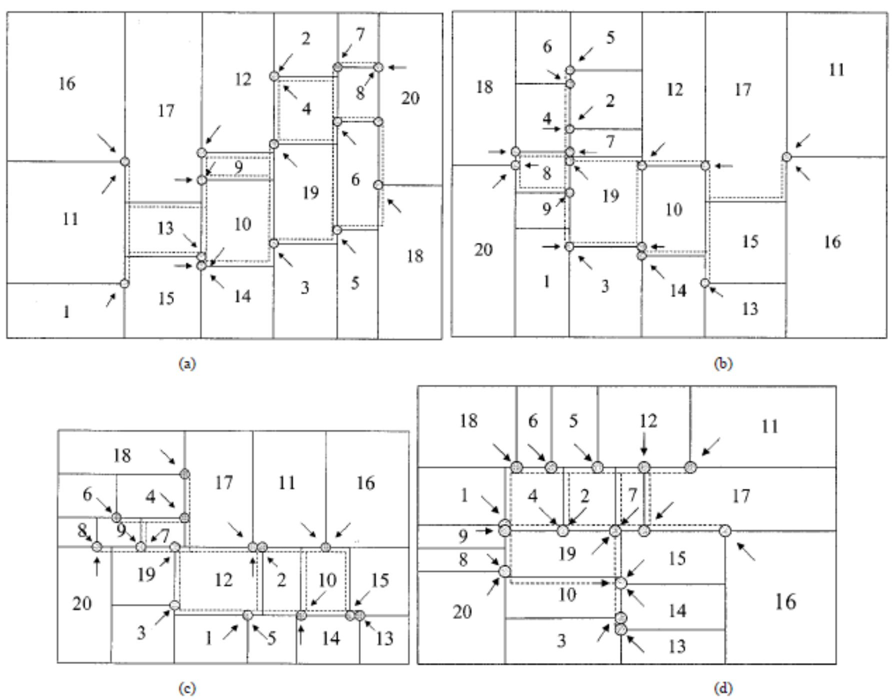
B.8.1 Bubble layout



B.8.2 Simple block layout

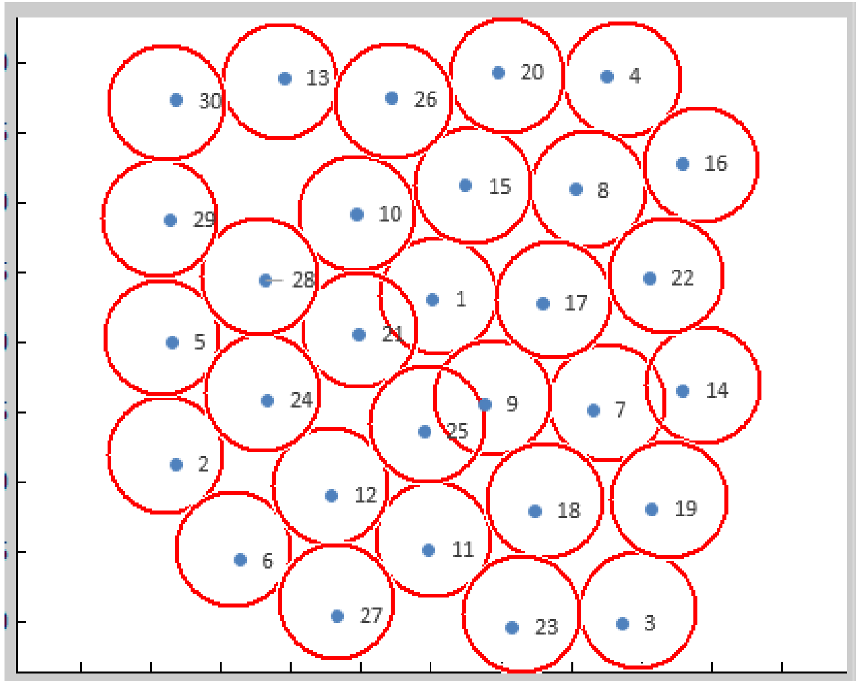


B.8.3 Detailed block layout

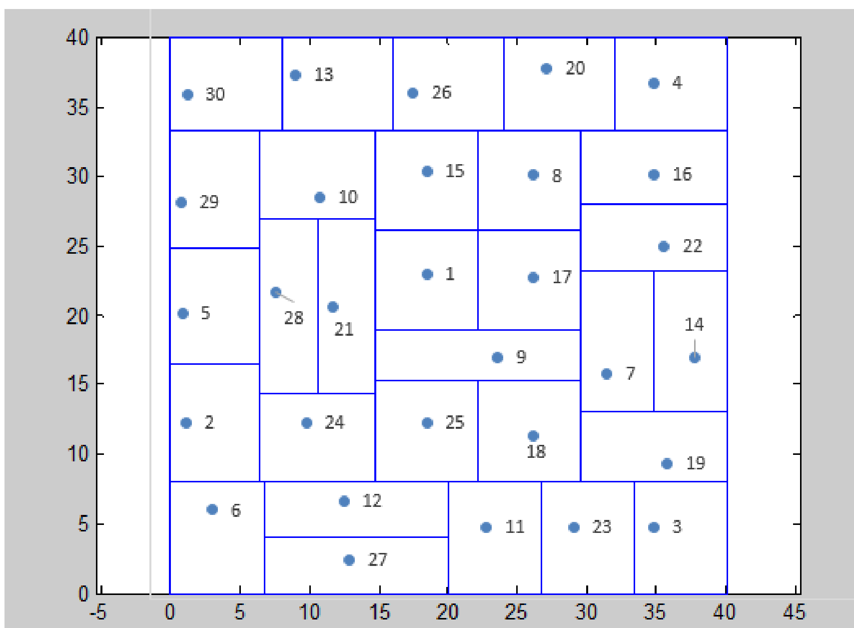


B.8.4 Comparison layout (Arapoglu et al., 2001)

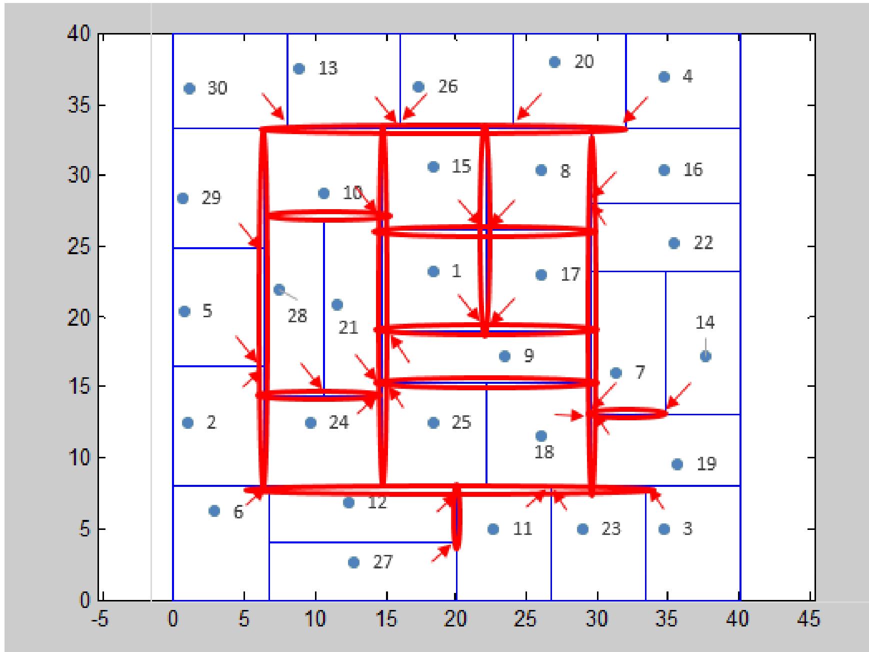
Figure B.8: Outputs from each stage of design framework for test problem 8.



B.9.1 Bubble layout

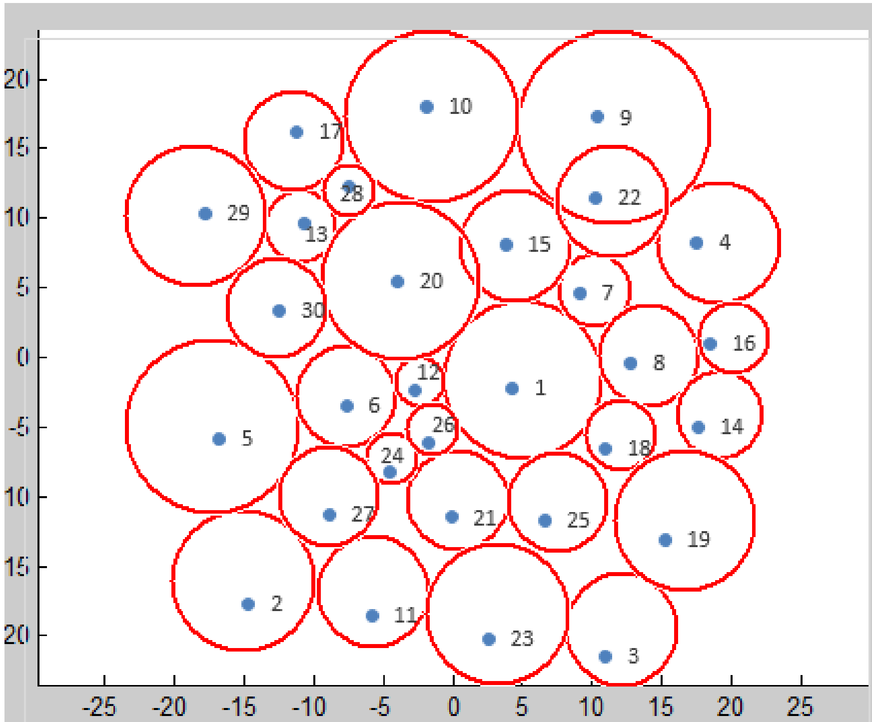


B.9.2 Simple block layout

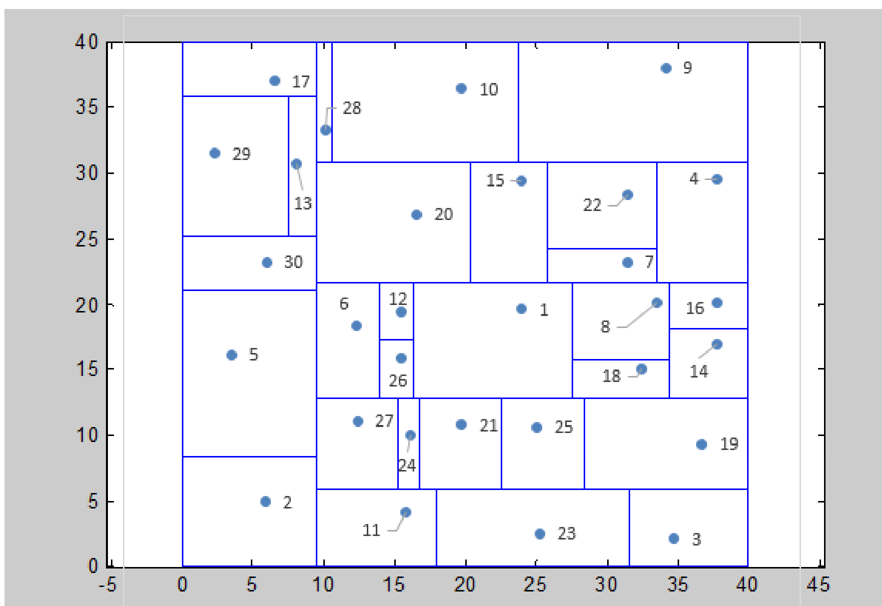


B.9.3 Detailed block layout

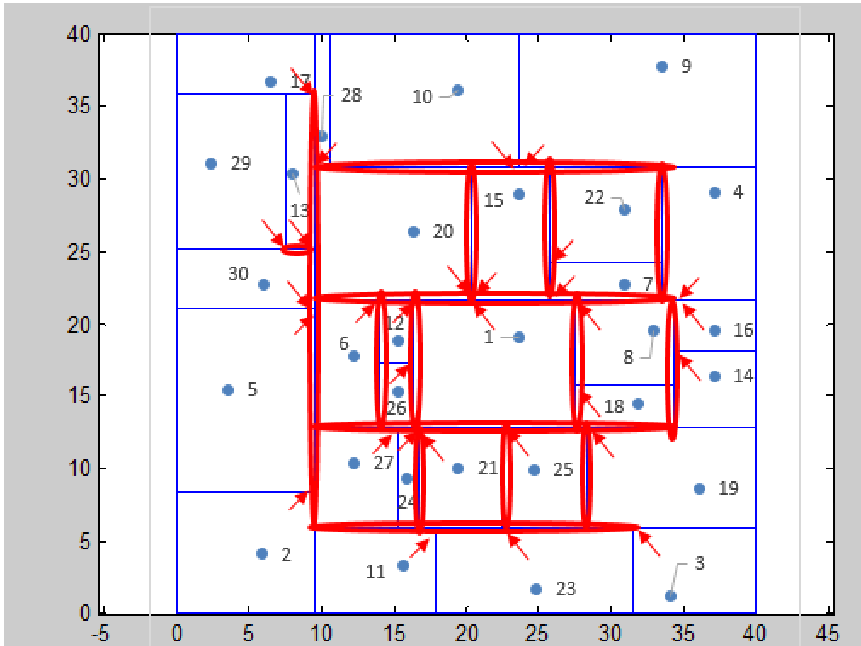
Figure B.9: Outputs from each stage of design framework for test problem 9.



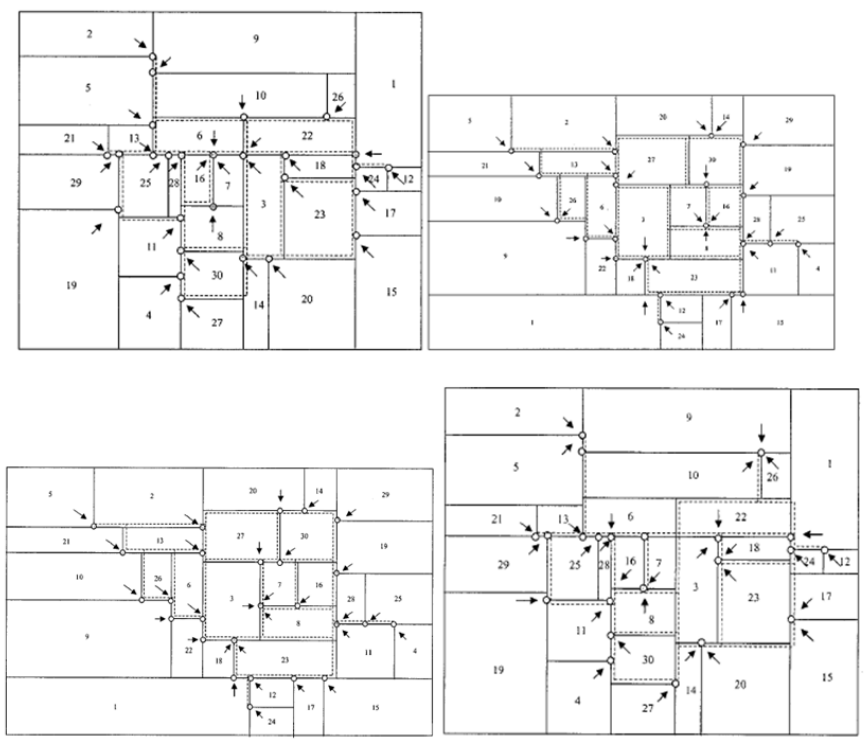
B.10.1 Bubble layout



B.10.2 Simple block layout

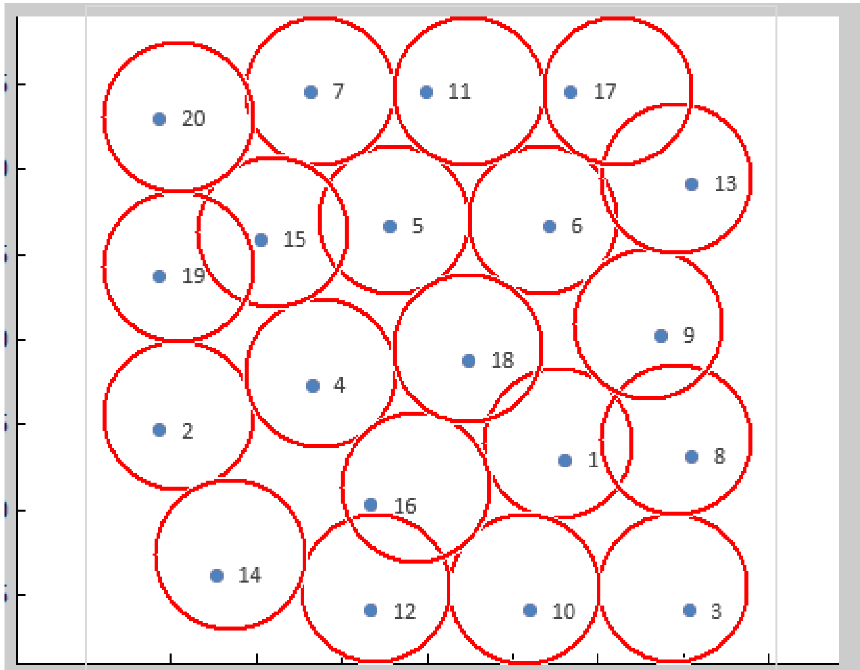


B.10.3 Detailed block layout

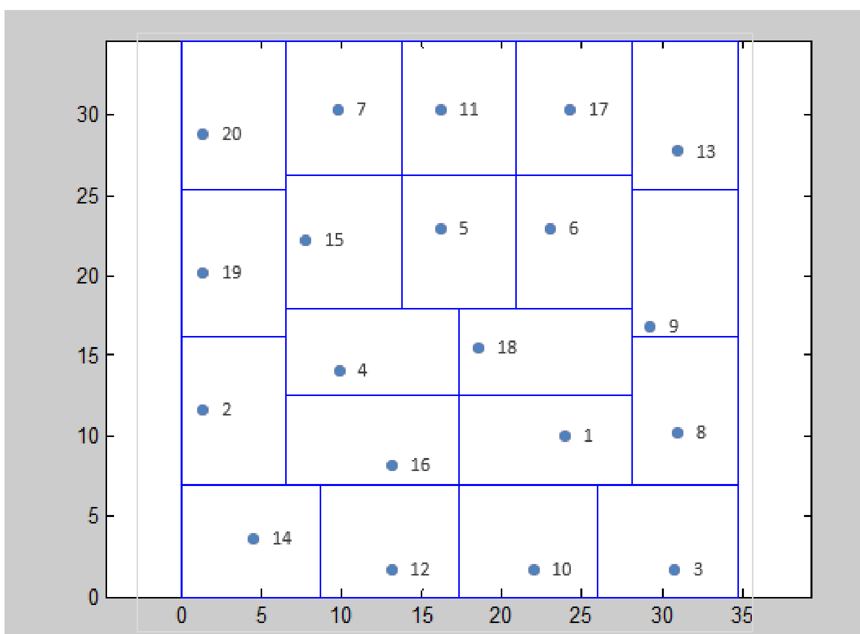


B.10.4 Comparison layout (Arapoglu et al., 2001)

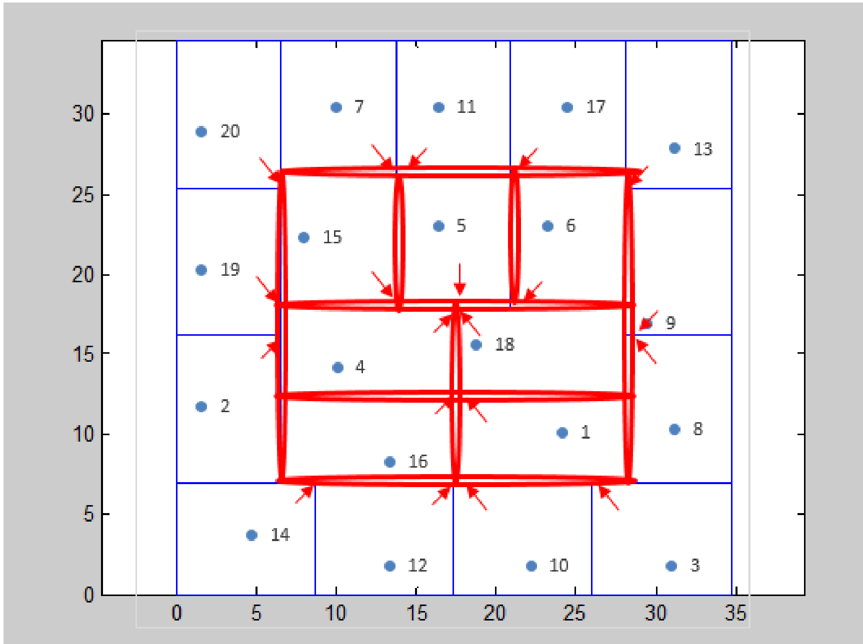
Figure B.10: Outputs from each stage of design framework for test problem 10.



B.11.1 Bubble layout

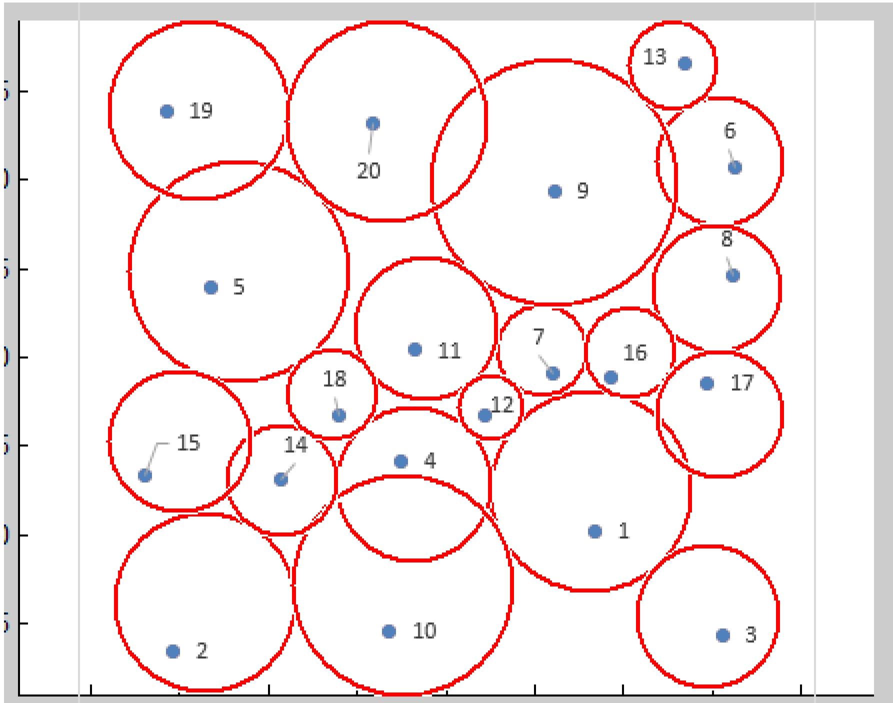


B.11.2 Simple block layout

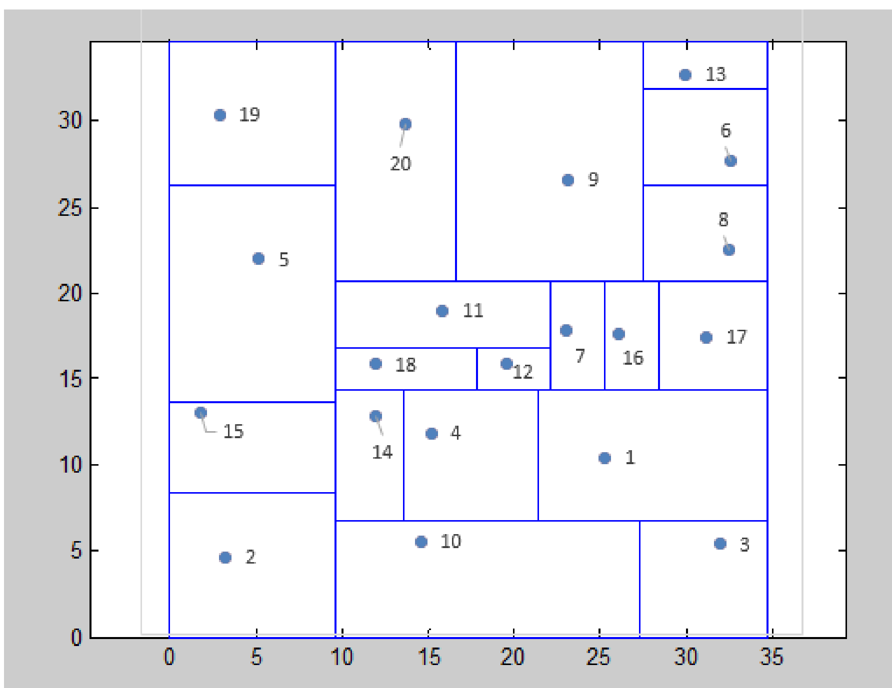


B.11.3 Detailed block layout

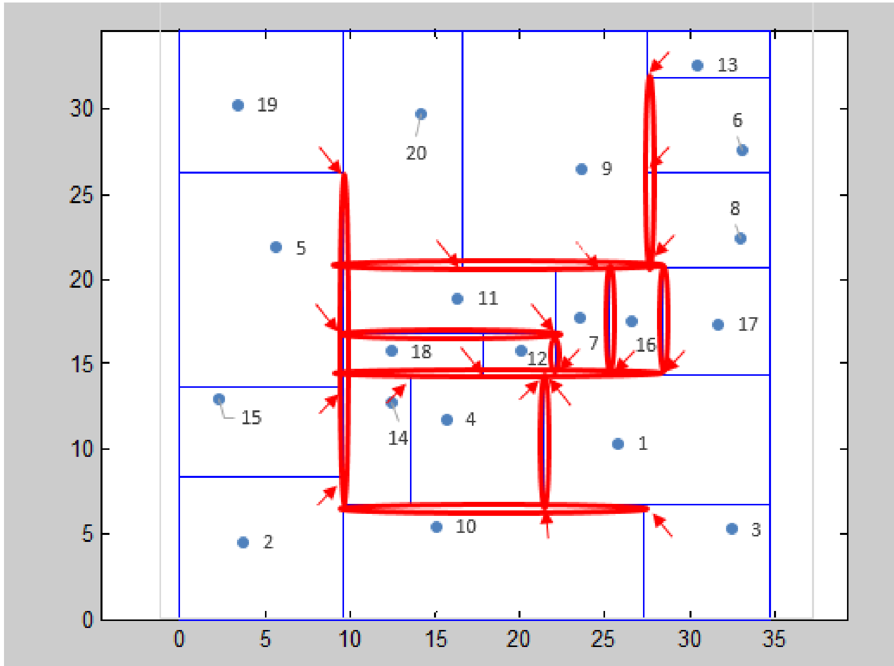
Figure B.11: Outputs from each stage of design framework for test problem 11.



B.12.1 Bubble layout



B.12.2 Simple block layout



B.12.3 Detailed block layout

Figure B.12: Outputs from each stage of design framework for test problem 12.