ISSN No:-2456-2165

Expediting Mass Rural Housing and Development through Algorithmic and Generative Space Planning of Housing Unit

Vishal Vaidhyanathan
Student, Department of Architecture,
NIT, Trichy, India

Parthasarathy Rajagopalan
Student, School of Law,
SASTRA University, Tanjore, India

Abstract:- Rural Housing schemes have been one of the greatest solutions from the government to solve residential crisis in rural areas. These schemes involve a long process for execution, including mass planning and design of various typologies of housing units depending on the site, the number of units, the budget and the stipulated time-frame. This article aims at providing a procedural computational methodology in generating spatial layouts – confining to small and medium scale housing units – with the objective of accelerating and improving the housing process. This methodology would also generate the suitable design space, i.e. increase the number of optimum spatial layouts to choose from as per requirements like proximity, material quantity and environmental quality. The generated layout samples were benchmarked against several existing and proposed rural housing and redevelopment plans designed by architects. The generated samples performed on par with existing layouts, few exceeding the latter. This indicates that the methodology generates similar or more efficient layouts than an architect in a fraction of the time, easing the process of executing such housing schemes.

Keywords:- *Automation, Computational Design, Configurative Design, Rural Housing, Slum Redevelopment, Space Planning.*

I. INTRODUCTION

The rural areas in India are characterized by small and highly dispersed habitations, apart from a poor village or community level infrastructure at the maximum. The principal occupation being agriculture (or its related activities), the space requirements of these households, include spaces pertaining to agricultural implements. Recently though, there has been an increase in non-agricultural jobs taken up by rural families.

The Indira Awaas Yojana (IAY) was the first and most comprehensive rural housing programme ever taken up in the Indian sub-continent. It was originally started as a part of the National Rural Employment Programme (NREP) and Rural Landless Employment Guarantee Programme (RLEGP) but was subsequently re-structured into an independent scheme-Pradhan Mantri Awaas Yojana–Gramin (PMAY-G) with effect from 1st of January 1996 [4].

In a nutshell, the programme plans to provide homes to 18 million households in urban India and nearly 30 million households in rural India. Now, the success of the programme depends largely upon the degree of efficacy in designing houses.

A. Computation and Automation

In this era of Cyber-Physical Industries and Inter-disciplinary development, Computational and Algorithmic methods are being utilized in every field to achieve efficient functioning. These methods not only prove to be efficient and quick, but also help in automating the stipulated task in hand and thereby reducing human efforts. Ranging from realms like Communication and Mobile industries to Autonomous Automotive industries, service professions pioneer in the ubiquitous usage of automation [5].

Automation and Computational methods for the time being, would not replace architects, but this does not mean that the discipline does not undergo profound transformations in its exercise: Algorithmic methods and Computation eliminate tedious repetitive activities, optimizing the production of technical data and allowing, among other things, fractionalize the time-period and effort.

This article explains the role of computational methodologies in drastically improving the efficacy in designing housing units – and expediting the rural development process – through algorithmically automating the process of planning by analyzing site topography, isolating buildable areas, optimization of master plan, and generation of spatial layout. The study emphasizes in detail the process involved in generating sample optimum floor plans based on user inputs (No. of spaces, proximity, areas etc.). It also presents a methodology to generate a combinatorial design space of multiple optimum spatial layouts to choose from, based on project requirements like environmental quality and proximity among others. The study is confined to small and medium scale residential typology, pertaining to rural housing.

II. BACKGROUND

The guarantee of ‘right to life of man’ has been inferred as a right to ‘dignified human life’ as opposed to guarantees constituting mere ‘animal life’. As humanity progressed, this guarantee came to be meticulously protected by way of judicial affirmation [1]. Presently, we have entered a stage, whereby, among several other economic, social and cultural rights, the need to recognize and uphold the ‘right to housing and shelter’ has been felt more than ever, by the governments and the common man alike.

ISSN No:-2456-2165

Article 25 of the Universal Declaration of Human Rights recognizes the right to housing as part of the right to an adequate standard of living [2]. Similar is the guarantee found in Article 11(1) of the International Covenant on Economic, Social and Cultural Rights (ICESCR).

Such recognitions and affirmative moves made by several national constitutions and international organizations moved developing economies towards this initiative. Accordingly, following the Istanbul Declaration on Human Settlement in June 1996 [3] to which India is a signatory, the Ministry of Rural development has been committed to accomplishing the goal of 'shelter for all'. To this end, several schemes were launched by the Government of India.

These schemes however, involve a tedious process of execution, starting from the selection of site to the construction and the consequent allocation of housing units to rural families in need. Hence, there is an impending need to emphasize on efficient planning and design of such housing projects.

III. METHODOLOGY

The overall process advocated by this article follows the methodology presented in Fig.1.

An overview of each individual process is defined in brief, followed by the Layout Algorithm in detail.

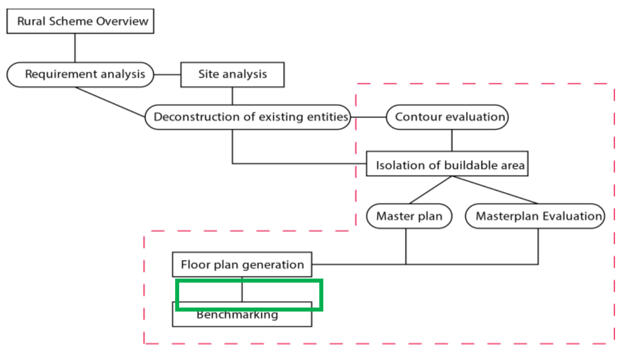

Fig. 1:- Procedural methodology adopted by the article. The box with dotted line encloses the computational entities of the procedure.

A. Rural Scheme Overview

The first step involves careful assessment of the scheme, the financial grant and the overall budget of the project. It also involves analyzing the core aim and the target demographics of the project. This preliminary analysis will give an overall idea about the following:

- Floor Area of individual housing units
- District based housing assistance offered
- Site location and area
- Site related building bye-laws
- Development control regulations
- No. of units planned etc.

These inputs serve to be an important criterion in the decisive process of the algorithm, that is, these inputs act as variable values while iterating them for optimization.

B. Requirement and Site Analysis

Successful data collection stipulates an immediate supply vs demand analysis. The data from the scheme overview is analyzed for typical requirements. Requirement analysis is highly essential to assess the availability of resources and the need to acquire more resources.

Site analysis involves analyzing the site on multiple levels like environmental quality, topography, surrounding infrastructure and climate. This analysis data advocates and affects the design optimization process.

C. Isolation of Buildable Area (Algorithmic Process in Brief)

Once the site is analyzed, the overall buildable area can be computed based on threshold slope values. The overall site is populated with test points, and the slopes of each point with its adjacent points are calculated. Based on a threshold slope (buildable slope), areas that are steep are eliminated as unbuildable zones. This process is illustrated in Fig.2.

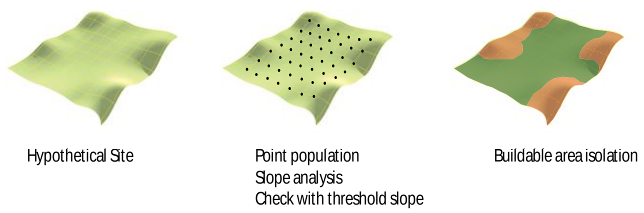

Fig. 2:- Process of computational isolation of buildable area from a given site model. The brief process shows how point slope data is compared with a threshold slope value to eliminate steep, non-buildable areas.

D. Master plan Layout and Evaluation

Once the buildable area is obtained, a rough master plan layout can be planned. This layout serves to be a guiding factor for the generation of floor plan layouts of individual units. The planned master plan layout is then evaluated for environmental quality on an urban level. Various analyses are done, and the most optimum layout is then chosen.

IV. LAYOUT ALGORITHM

This section explains the overall logic of the algorithm formulated. Having run through the overview of processes leading to the housing unit spatial layout generation stage, it is necessary to understand the overall procedural structure of this stage.

Generating of floor plan layouts of housing units is one of the most important aspects in expediting the housing scheme execution process. It solves the problems faced in planning and design of these units by generating optimized plans, and a dataset of possible plans with a given set of required inputs. The algorithm functions on a CAD platform, owing to an extensive flexibility in producing technical and constructional drawings post generation of layouts.

Fed with a Site Boundary Polygon (or Footprint Polygon from Master plan) and a set of input variables namely:

- No of Spaces (input as random point population in site)
- Areas of each space (input as a list of floating point values)
- Required Proximity Topology (input as a line graph connecting the point population)
- Setbacks required (if any) (input as a list of floating point values)

A set of algorithms analyses the input data and tries to generate the most optimum rectangular spatial layout satisfying all proximity requirements. In case the input topology returns an impossible solution, the algorithm tries to satisfy maximum number of requirements possible.

The algorithm handles all kinds of irregularly shaped building footprints, however, restricted to linear shapes. The design space generated by the algorithm (all possible satisfying floor plans) is a result of variations in possibilities of the following aspects:

- Length x Breadth ratios of rectangular spaces maintaining the same area
- Multiple configurations satisfying the given proximity topology without overlap
- Multiple orientations satisfying the given criteria.

The algorithm currently functions adhering to all input variables without any relaxation. However, it has been tested to provide more results when one or few variables are slightly allowed to be relaxed, hence mimicking human cognition.

For e.g., if the algorithm is allowed to relax the area with a permissible limit of ±10 m^2, like a human architect, additional design configurations are found to be generated with greater design flexibility.

The algorithmic skeleton and the procedural structure is illustrated in Fig.3.

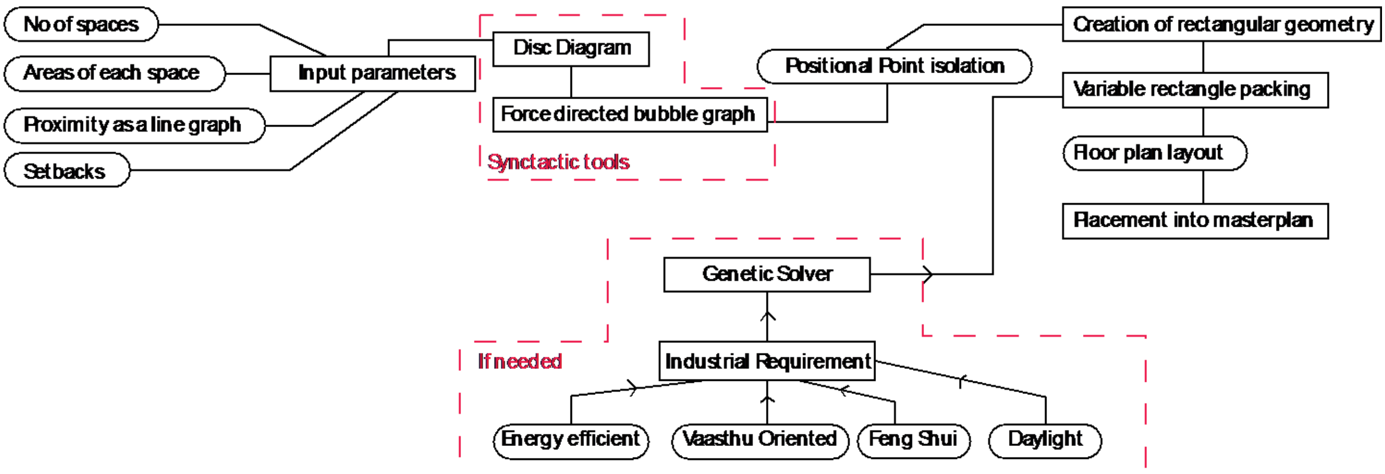

Fig. 3:- Procedural methodology adopted by the Layout algorithm to generate spatial layouts. The Disc Diagram and Force directed bubble graph are utilized from SYNCTACTIC tools. The genetic solver is used during the variable packing stage (explained below) as per need to tweak certain variables of the algorithm to attain optimum spatial layouts adhering to special requirements.

The input parameters are registered as mentioned before. They are illustrated in Fig.4 (a). and Fig.4 (b).

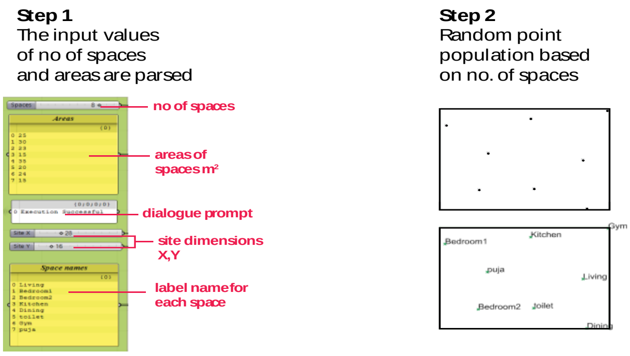

Fig. 4(a):- Method of parsing of input parameters by the algorithm. The image shows the data type and the mode of input for the various parameters.

As shown in the figure, based on the number of spaces input, a random point population is generated on the given site boundary (or master plan footprint). These points are then used as cardinal points to map the proximity topology graph in the form of lines, as illustrated below. The proximity is input as tuples. The algorithm generates the topology graph based on the input tuple values.

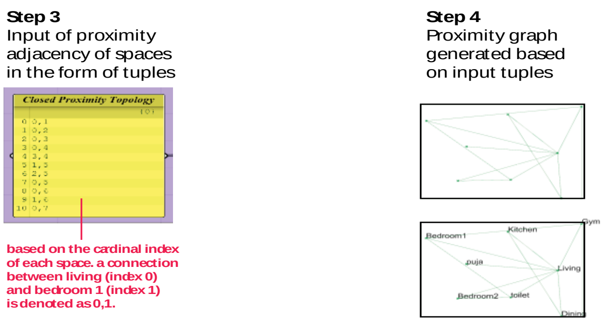

Fig. 4(b):- Method of generation of topology graph from set of input tuples. The algorithm draws a graph joining the cardinal nodes based on the proximity tuples input by the user.

ISSN No:-2456-2165

Once these input parameters are collected and parsed into the specified formats, the algorithm makes use of SYNCTACTIC tools (Nourian, 2013) [6] to generate a disc diagram.

The disc diagram is composed of various 'discs' with areas in accordance with the required input values. With the help of a force directed methodology, the disc diagram aligns into a 'bubble diagram', which is an adjacent packed disc on the basis of the input proximities required. The entire sequence is illustrated below in Fig.5.

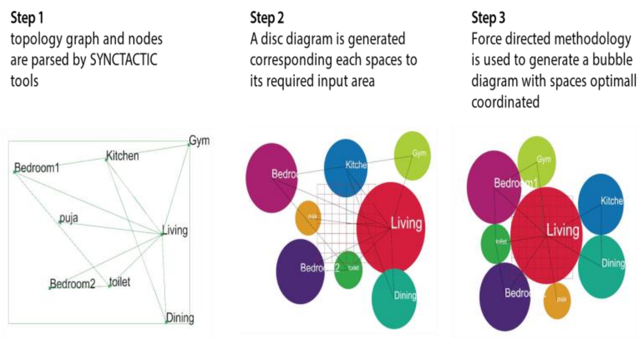

Fig. 5:- Illustration of conversion of proximity topology graph generated, into disc diagrams based on area. SYNCTACTIC tools is used for this purpose. Using force directed method, the disc diagram is converted to an optimum bubble diagram, satisfying all proximity needs.

The final bubble diagram generated gives us the actual positional nodes of each space within the site. The bubble diagram can by dynamically changed based on change in proximity requirements. The 'seed' of the bubble diagram can also be changed to provide an alternate solution for the given proximity requirements.

Once the final cardinal points are obtained, variable rectangles (those with varying length-breadth ratios without change in are) are generated with these points as the geometric center. It is ensured that the rectangles are restricted within the site (or footprint) boundary. The process is illustrated below in Fig.6.

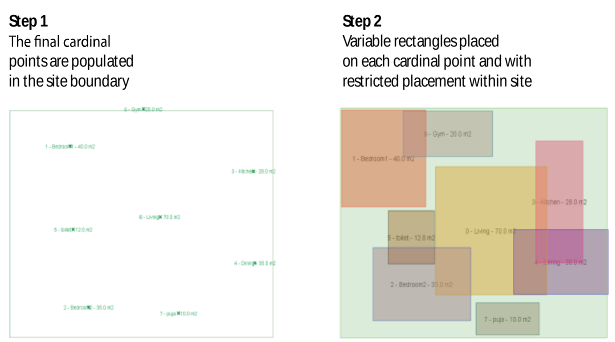

Fig. 6:- Rectangulation stage. The cardinal points placed at the final positions derived from the bubble diagram are converted into variable rectangles based on the given area values.

Once the rectangle placement is done within the site, and in their proper positions solving proximity and connectivity requirements, the final step is to pack them under the following conditions:

- Retainment of relative positions
- No negative spaces in-between
- No overlap of rectangles
- Constant maintenance of area (allowing change in length breadth ratios)
- Rectangles stay within site boundary
- (if needed) Packing happens with respect to additional requirement

This can be done through variable packing – packing with relaxation on varying lengths and breadths without change in area of each rectangle. To achieve this, rectangles are subject to the following allowable transformations:

- X axis – Translation
- Y axis – Translation
- Size control (variation in length and breadth without change in area)

The discussed transformations are illustrated in Fig.7. below.

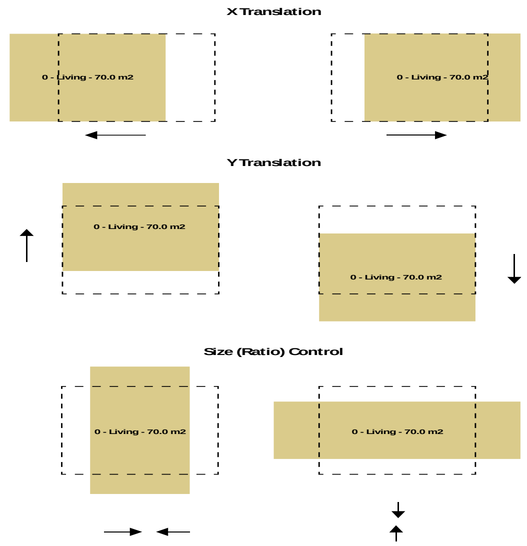

Fig. 7:- The various transformations undertaken by the algorithm during the variable packing stage.

With these transformations, the variable packing algorithm performs the operations illustrated in Fig.8. and the database of optimum floor plan layouts are generated.

It is noteworthy that further changes in these rectangle transformations can be done manually to create a situation called brute-force, which can provide layout results without changing the manually set transformation values. Fig.9. shows a of the generated design space of the current example scenario considered throughout this article.

Multiple optimized spatial layout can be obtained from this algorithm. While this algorithm works around the possible translations of each rectangle based on requirement, many possible layouts are possible.

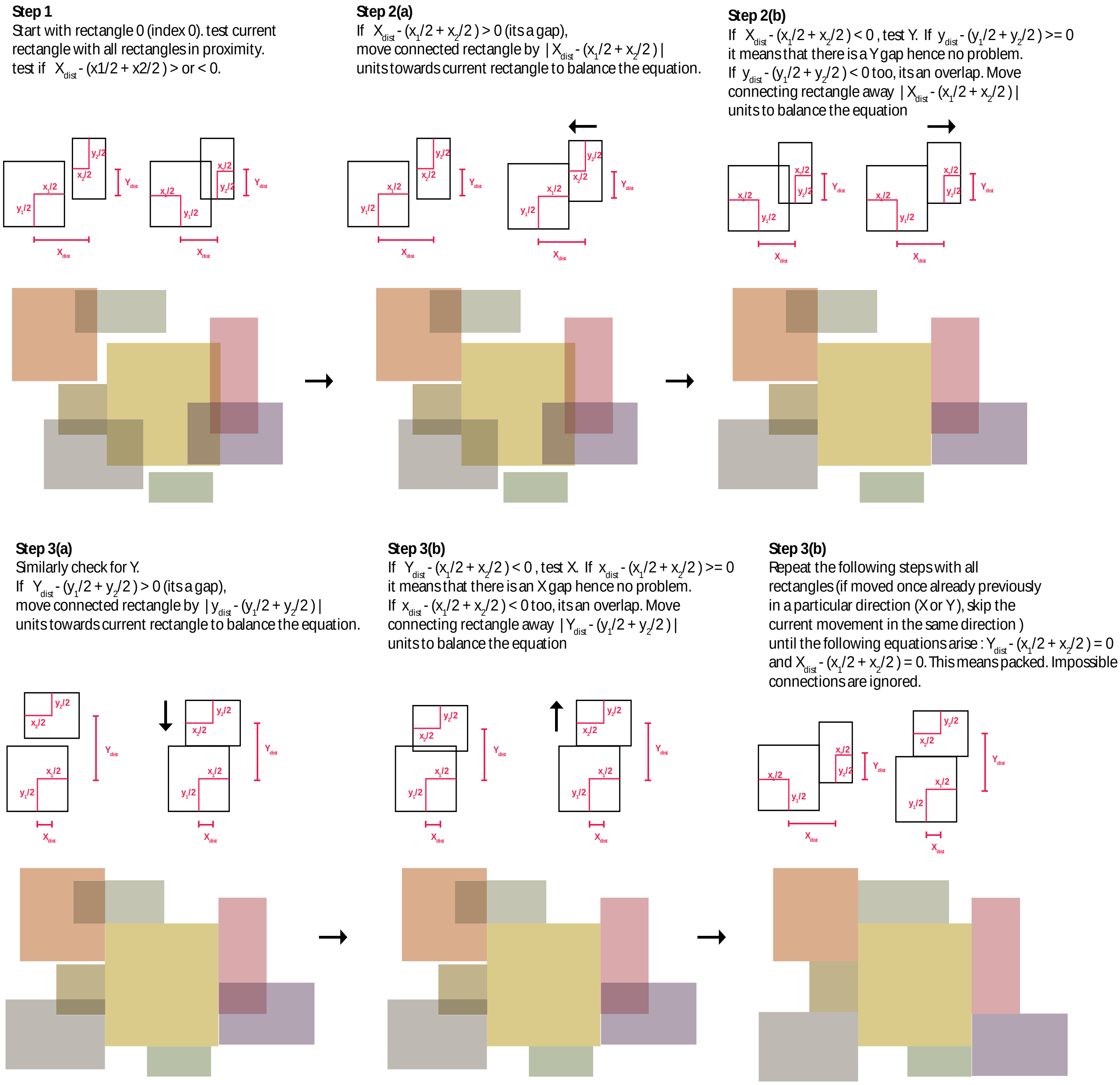

Fig. 8:- The procedural methodology followed by the Layout Algorithm to generate floor plans. This procedure makes it evident how the algorithm uses the various transformations of rectangles to generate the floor plan.

The algorithm can be tweaked to use the rectangle transformations differently based on requirement like environmental quality, Feng shui, etc.This is done with the help of an evolutionary solver that uses:

- Fitness function (that specifies the requirement)
- Genomes (traits whose modification brings it closer to the fitness functions)

Examples of various possible layouts for the current scenario used in the article are illustrated below. These layout samples were generated from a single input class based on changes in the way the different transformations are carried out.

These changes are carried out to satisfy certain additional requirements that include – and not restricted to – Environmental quality, Proximity, Site features, Surrounding infrastructure, Day lighting and ventilation etc.

The variations are done with the solver that suggests changes based on an experimental procedure. It keeps learning as it goes. This process is called evolutionary solving. The various genes (variables) are modified to attain a value closer to the fitness function. The modification becomes smarter and smarter as it progresses. When the final fitness function value is reached, the solver terminates.

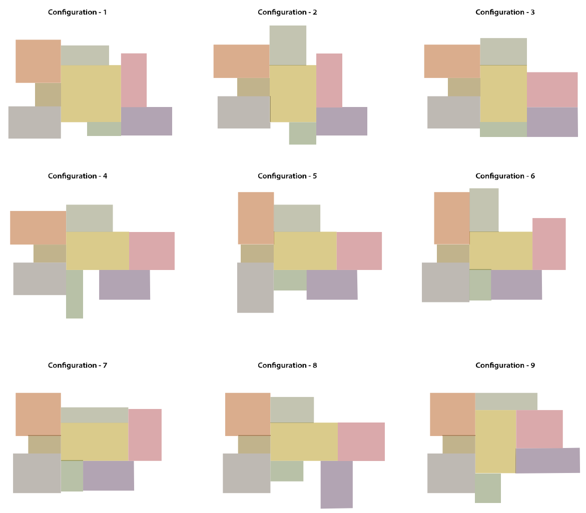

Fig. 9:- A subset of the design space. This shows a small list of all the possible optimum solutions generated.

V. BENCHMARKING

The multiple layouts thus generated were benchmarked against existing proposals and designs of similar rural development housing schemes and slum development schemes.

The Ministry of Rural Development (MoRD) has published two volumes of 'PAHAL' - a compendium of Rural Housing Typologies' which consists of more than 100 designs for 15 states. 'PAHAL' is considered as guiding resource and a ready reckoner for the stakeholders of the country's housing scheme, to construct affordable, functional, durable and disaster resilience houses for the beneficiaries. 'PAHAL' is an outcome of the detailed studies of the place and consultative process with the rural communities, government stakeholders at different levels and civil societies. Thus, the resultant design is a blend of conventional construction materials (e.g., cement, bricks, and sand) and technology with locally available materials (e.g., clay and stone) [7].

It hence serves to be a worthy test source to operate the benchmarking test and assess the efficiency and comparative performance of the algorithm with architects' designs from PAHAL.

Figure 10. Illustrates the benchmarking results on comparison of 100 output sample layouts of 20 different input values with designs from PAHAL. The benchmarking was done manually, based on standards, requirement-satisfaction and functionality in relation to PAHAL.

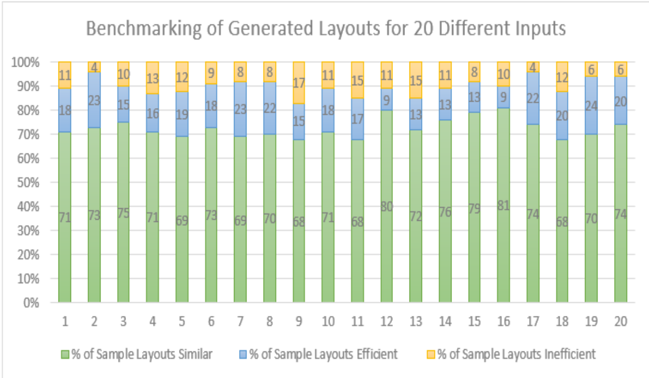

Fig. 10:- Benchmarking results showing the percentage of layouts matching, efficient and inefficient compared to existing design (architects' design).

VI. CONCLUSION

A computational approach to accelerate the rural housing and slum redevelopment process is proposed in this article. It has been established that computational and algorithmic methods not only provide efficient and instantaneous solutions but also provide a database of optimum outputs to choose from or manipulate based on industrial requirement. It is evident from the results of benchmarking in the article that, the algorithm provides matching or more efficient spatial layouts as compared to existing designs (architects' design) in a fraction of the time and thus expediting the overall process.

This represents a key step in the advancement of both algorithmic procedures and computational automation in the housing and development industry. It also serves to be a launch pad for execution of multitudes of similar schemes as a consequence of ease of work and instantaneous results. Expanding this algorithm to further interfaces - and developing tools and applications - would serve to be an important asset for professionals and individuals from academia. Further improvement of the algorithm would make feasible automation of infrastructure of larger scales and value.

REFERENCES

[1] U. S. Const. amend. XIV India Const. (1950) art. 21
[2] Edgar Bill, Doherty Joe, Meert Henk, "Access to housing: homelessness and vulnerability in Europe". The Policy Press. p. 17. ISBN 978-1-86134-482-3. 2002.
[3] http://www.un-documents.net/ha-4b.htm
[4] https://pmayg.nic.in/netiay/about-us.aspx
[5] Frey C, Osborne M. "The future of employment: how susceptible are jobs to computerization." Technol Forecast Soc 2017, 114: 254–280.
[6] Nourian P, Rezvani S and Sariyildiz S, "Designing with Space Syntax: A configurative approach to architectural layout, proposing a computational methodology Computation and Performance.", Proc. 31st eCAADe, 2013, pp. 357-366.
[7] NIPFP Research Team, "Impact of Pradhan Mantri Awaas Yojana- Gramin (PMAY-G) on Income and Employment", New Delhi, India.