

Space planning from environmental parameter

Diego Jami Menezes da Silva

Universidade Federal do Espírito Santo | Brasil | jami.diego@gmail.com

Jarryer Andrade de Martino

Universidade Federal do Espírito Santo | Brasil | jarryer.martino@ufes.br

Abstract

The environmental parameter as a constraint in the initial phase of the design process becomes essential for the promotion of the building's environment comfort. This research explores the principles of space planning and the genetic algorithm considering an environment parameter to obtain optimized-solutions automatically in the initial phase of the architectural design process. For this, a generative system was developing using space planning and a genetic algorithm from an environmental parameter to obtain better solutions automatically. Several studies happened to understand the functioning of the mechanism. The research discusses the results and failures that may have influenced the generative system.

Keywords: Genetic algorithm; Generative system; Space planning; Performative design; Environmental comfort.

INTRODUÇÃO

A organização espacial de um projeto de arquitetura considera como os componentes de um programa de necessidades se articulam em relação ao todo e às condicionantes externas provenientes do local de implantação. A inserção dessas condicionantes na fase de concepção do projeto torna-se fundamental para a promoção do conforto ambiental do edifício. Segundo Yeang (2000), o desenvolvimento de projetos que tornem o edifício ambientalmente sensível é capaz de reduzir o consumo de energia durante a fase de ocupação e oferecer bem estar ao usuário, permitindo usufruir do clima externo local. Além disso, Kolarevic (2005) afirma que o interesse pelo desempenho como um paradigma de projeto é crescente, principalmente devido ao desenvolvimento da tecnologia e ao surgimento da sustentabilidade como uma questão socioeconômica.

Nesse sentido, a incorporação das ferramentas digitais no processo de projeto permite a criação de metodologias que redefinem a morfogênese da arquitetura. Entre os cinco modelos de projeto digital apresentados por Oxman (2006), encontra-se o modelo baseado no desempenho, definindo este como um modelo que utiliza a simulação digital de forças externas aplicadas no direcionamento do processo de formação. Dessa forma, a simulação computacional da edificação passa a integrar a fase inicial do desenvolvimento de projeto, contribuindo para a geração de informações e a realização de análises de desempenho capazes de retroalimentar o processo de projeto (Brigitte & Ruschel, 2016).

Sendo assim, entre as variadas condicionantes que precisam ser negociadas durante o desenvolvimento de projeto, a implantação do edifício em um lote e a

organização espacial dos componentes do programa de necessidades assumem papel fundamental para o desempenho térmico do edifício. Isso ocorre porque ambos são orientados e definidos em função da orientação solar. Principalmente em um clima tropical, pois a principal causa de desconforto térmico nos ambientes é o ganho de calor causado pela absorção provocada pela incidência solar nas superfícies externas do edifício. Assim, proteger as superfícies da radiação solar é o primeiro objetivo do projeto arquitetônico (Corbella & Yannas, 2009, p. 41). Para isso, os componentes do programa de necessidades devem ser dispostos no lote de maneira a obter o máximo de conforto ambiental e garantir simultaneamente o melhor aproveitamento possível do espaço.

No entanto, as possibilidades de implantação do edifício em um lote e a sua organização espacial são diversas, o que torna difícil a análise detalhada de cada opção em processos de projeto tradicionais, que não incorporam processos automatizados de busca da melhor solução, ou seja, mantêm o processo de tentativa e erro. Assim, tendo em vista o curto tempo para a elaboração de um projeto, o arquiteto geralmente adota a "regra de boa conduta" (Kowaltowski et al., 1998, p. 372) para a tomada de decisão, assumindo uma concepção mais generalizada de projeto e gerando ambientes com possíveis problemas de conforto.

Dessa forma, automatizar o processo de planejamento do espaço e direcioná-lo sob o ponto de vista da orientação solar, a fim de conduzir a organização para a obtenção de soluções favoráveis ao desempenho térmico, é a premissa desta pesquisa. De acordo com Scheer (2014), além da possibilidade do projeto ser concebido com base na capacidade de resposta ambiental, os processos

podem ser estruturados de maneira a simular a própria natureza, garantindo a adaptação e evolução da solução por meio de um projeto algorítmico. Para isso, o modelo de projeto baseado no desempenho precisa da incorporação de uma técnica evolutiva conduzida por meio de um algoritmo genético. O experimento apresentado neste artigo incorpora conceitos relacionados ao *Space Planning* (geração automatizada de leiautes) e Computação Evolutiva (algoritmo genético).

O *Space Planning* é um campo de pesquisa que investiga o planejamento de espaços a partir da organização ideal dos programas. Para isso, o método que utiliza os seus conceitos, define um sistema gerador capaz de solucionar problemas vinculados à organização de um conjunto de elementos topológicos em um determinado espaço, sendo avaliadas as relações de distância, adjacência e outros parâmetros específicos (Eastman, 1973 apud Calixto, 2018). A este método associa-se o algoritmo genético, responsável por realizar uma busca no campo das prováveis soluções de um problema, simulando, avaliando e selecionando as que melhor satisfazem as condicionantes envolvidas neste processo (Martino, 2015). Este algoritmo envolve processos iterativos estruturados a partir de uma lógica simplificada e abstrata da evolução (Hensel, Menges & Weinstock, 2010).

Portanto, o objetivo deste artigo é apresentar uma estratégia de projeto baseada no desempenho, com o intuito de automatizar a geração de leiaute de projeto de uma residência mínima em um lote, utilizando a insolação como condicionante responsável pela condução do processo. Permitindo dessa forma, aperfeiçoar este processo, tornando-o mais eficiente no aproveitamento dos recursos naturais e atingir o melhor conforto ambiental possível.

METODOLOGIA

A automatização do processo de geração de leiaute e implantação da edificação no lote a partir da condicionante orientação solar pretende servir de instrumento auxiliar na tomada de decisões realizada pelo arquiteto. A intenção não é que ele seja eliminado do processo de solução de problemas, mas sim alterar a sua forma de interação no processo de projeto, a fim de deixá-lo livre para as suas atividades criativas e não rotineiras (Eastman, 1973 apud Calixto, 2016).

Para isso, antes de estruturar a metodologia para o sistema gerador de leiautes, foram identificadas as diferentes formas de interações entre o projetista e a representação, sendo as definições de Oxman (2006) adotadas como referências: (a) interação com uma representação de forma livre (ou baseada em papel): é a mais comum em processos de projeto, de maneira que o projetista interage diretamente com a representação do projeto; (b) interação com construções digitais: onde o projetista interage com a representação digital do projeto, seja um esboço, um desenho ou um modelo; (c) interação com uma representação digital gerada por um mecanismo: neste caso o projetista interage com uma estrutura resultante de um mecanismo que gera a representação a partir de regras pré-determinadas; (d) interação com um ambiente digital que gera uma representação digital: aqui o projetista interage com o ambiente digital onde é gerado o mecanismo responsável

pela geração da representação digital. Sendo, portanto, o projetista responsável pela definição das regras que regem o mecanismo, assim como o desenvolvimento do próprio mecanismo.

Diante destas definições e do objetivo desta investigação, adota-se o modelo de interação em que o projetista se relaciona com o ambiente digital, sendo este o responsável por gerar a representação digital. No entanto, o projetista assume a responsabilidade pela definição dos critérios e estruturação do mecanismo de geração automatizada. Este será responsável pela geração, avaliação e a seleção das soluções que melhor se adequem às exigências do projeto. Este mecanismo vai ao encontro do que Mitchell (1975) define como sistemas generativos de projeto. De maneira que, os resultados obtidos por este sistema generativo são consequências da estratégia adotada para a sua estruturação. Por isso, o projetista precisa avaliar constantemente se a estratégia adotada está gerando resultados coerentes e factíveis, caso contrário, é necessário redefinir ou revisar os parâmetros utilizados no sistema até que se atinjam resultados satisfatórios. Este processo pode revelar falhas antes não imaginadas, mas que diante dos resultados obriga o arquiteto a redefinir a sua estratégia, criando restrições para que aquelas não ocorram novamente. Assim, o processo de retroalimentação é constante, contribuindo para o desenvolvimento de um conhecimento aprofundado sobre o problema.

Para entender melhor o que é este mecanismo, podemos estruturá-lo em três partes. A primeira define quais são os blocos de construção de um problema, ou seja, “[...] fenômenos regulares que aparecem repetidamente em nossa experiência de mundo.” (Holland, 1998, p. 19. tradução nossa). Para esta pesquisa, podemos pensar como blocos de construção o retângulo (que representará um componente do programa de necessidades, ou seja, um cômodo) e as alterações que podemos fazer nele (girar, deslocar, alongar, copiar etc.); ou o lote (onde será inserida uma construção) e os fatores ambientais que o afetam (incidência solar, posição do sol, vento dominante etc.). Assim, os blocos de construção correspondem aos elementos que irão compor o modelo a ser simulado, ou seja, os atores responsáveis pela atuação e estruturação da cena.

O segundo componente é o gerador, este é o responsável pela geração a partir da combinação dos diversos blocos de construção. Para isso são utilizadas algumas regras para que a combinação de blocos de construção simples gerem outros novos e mais complexos. E por fim os agentes, que são os responsáveis diretos na tomada de decisão, estes possuem algumas leis que os regem. Segundo Holland (1998), para programar um mecanismo no computador é necessário atingir três objetivos: (a) determinar os blocos de construção do modelo; (b) inserir esses componentes com as devidas regras no computador; (c) promover a interação dos componentes e suas regras que levam a geração de novos mecanismos mais complexos.

Em casos em que os dados de entrada e de saída desse mecanismo são valores simbólicos (palavras, números etc.), Mitchell (1975) define como sistema generativo simbólico, visto que as operações realizadas para alterar

o seu estado são lógicas e aritméticas e as soluções potenciais são representadas por meio de símbolos, como palavras, números e operadores matemáticos. Mitchell define ainda outros dois sistemas generativos: o analógico, onde as operações realizadas para mudar o estado do sistema são mecânicas, por exemplo, a rotação de rodas, deslizar colunas lado a lado; e o icônico, em que as operações realizadas para alterar o estado do sistema são gráficas, por exemplo, fazer e apagar linhas, mover recortes de papelão e assim por diante, neste caso a representação é muito mais literal. O experimento apresentado neste artigo adotou o sistema generativo simbólico, utilizando o *plugin* Grasshopper do programa Rhinoceros® para a estruturação da programação algorítmica.

Esse modelo de geração em que a forma é gerada a partir de critérios de desempenho desejados é definido por Oxman (2006) como modelo de geração baseada em desempenho. Modelo de geração, já que o projetista interage com um mecanismo que gera as formas a partir de regras (definidas pelo próprio projetista) e baseada em desempenho, pois as formas são geradas de acordo com o critério de desempenho desejado. Neste modelo, o processo se inicia com a definição dos parâmetros de desempenho a serem adotados, em seguida os parâmetros de desempenho conduzem a geração de soluções e a partir do sistema de geração de soluções é definida a forma. Aqui o projetista interage diretamente em três situações: na definição dos parâmetros de desempenho do sistema, na seleção das soluções geradas e com o desenvolvimento da solução escolhida.

Esse processo é contínuo, portanto, os critérios precisam ser objetivos para a avaliação e pontuação das soluções geradas, permitindo decidir quais são os melhores resultados ao fim de um tempo ou ao atingir uma pontuação determinada. Então, faz-se necessário que o próprio mecanismo seja capaz de avaliar e pontuar cada indivíduo gerado de um conjunto de soluções, de modo que o processo consiga atingir gerações de indivíduos melhores, até obter um resultado ótimo.

Para isso, o algoritmo genético atua como uma ferramenta de geração e avaliação. Segundo Liggett (2000) o algoritmo genético dispõe de ao menos cinco recursos que são capazes de auxiliar na geração e avaliação de gerações: (a) geração de uma população de soluções inicial (pode ser gerada aleatoriamente); (b) um mecanismo capaz de gerar novas soluções a partir das soluções existentes na população; (c) um mecanismo capaz de gerar novas soluções a partir de uma única solução existente; (d) um mecanismo capaz de avaliar as soluções e selecionar aquelas com melhor pontuação dentre a população; (e) e um mecanismo para remoção de soluções da população. Estes recursos podem ser resumidos em: geração, reprodução, mutação, seleção e eliminação.

Sendo assim, é proposto um sistema generativo baseado no desempenho que possui como dados de entrada a orientação solar, a relação entre cômodos, a posição e dimensão do lote e áreas mínimas dos cômodos, sendo estes os blocos construtores do mecanismo. Após a geração das soluções pelo mecanismo, os dados de saída apresentados são os resultados obtidos pelo sistema e, que, serão analisados pelo projetista a fim de

escolher a melhor dentre as melhores geradas. É importante destacar que o processo considera valores numéricos para a avaliação e seleção, de maneira que, a diferença entre os valores obtidos nas melhores soluções podem não ser expressivos. Isto dá liberdade ao projetista para adotar critérios subjetivos, e que não integravam o sistema, para assim selecionar a solução que melhor atenda aos seus objetivos. No entanto, é possível que o mecanismo apresente resultados insatisfatórios, exigindo a sua revisão e da estratégia empregada, levando consequentemente a sua redefinição. Após este procedimento, os dados são reinseridos para que o sistema gere novas soluções a fim de serem analisadas novamente pelo projetista.

RESULTADOS

Durante a utilização do Grasshopper e do componente Galapagos (Figura 1), ferramenta nativa de algoritmo genético, percebeu-se a necessidade de converter todas as relações e parâmetros propostos no mecanismo em um único resultado numérico. Este resultado é chamado de *fitness* (valor de aptidão) e os valores utilizados nos parâmetros (dimensionamento e localização) são denominados como *genome* (genoma). Para atingir o melhor *fitness* possível são duas as alternativas possíveis: através da maximização ou da minimização. Assim, todas as relações do mecanismo precisam ser possíveis de mensurar por maximização ou minimização.



Figura 1: Bateria do Galapagos no Rhinoceros®. Fonte: autores (2019).

Para um projeto arquitetônico de baixo custo, é desejável que haja boa distribuição no leiaute, que seja oferecido o melhor aproveitamento de área possível e que se garanta uma construção rápida e simples. Um dos fatores que podem prejudicar essa construção é uma planta que possua perímetro muito complexo (Figura 2).

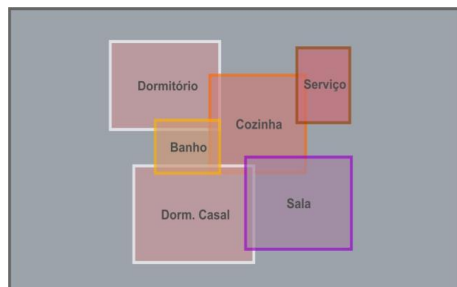


Figura 2: Exemplo de Perímetro Complexo. Fonte: autores (2019).

A fim de limitar resultados com esse problema, foi realizado um experimento que buscou minimizar o perímetro da solução gerada (Figura 3), simplificando a

sua complexidade. O resultado demonstrou que na busca pelo menor perímetro, os cômodos tendem a sobrepor-se, de modo que o perímetro externo possua a menor dimensão possível, além de adotar o valor mínimo para a área de cada cômodo. Para entender qual seria o resultado para a busca extremamente oposta, fez-se o experimento buscando o maior perímetro possível (Figura 3), o que resultou em cômodos desconectados e com a maior área possível em cada um deles.

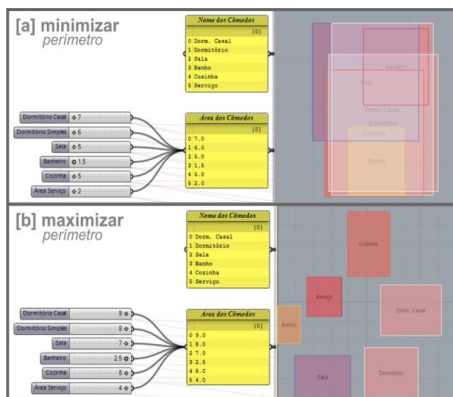


Figura 3: Minimizar Perímetro (a) e Maximizar Perímetro (b).
Fonte: autores (2019).

Prosseguiu-se, ainda com o experimento que buscou atingir o maior perímetro possível e a maior área possível (Figura 4). Este resultou em um perímetro complexo em que todos os cômodos atingiram a maior área possível e estes se tocavam em suas extremidades, garantindo o maior perímetro possível. Com esses experimentos iniciais, foi possível entender o funcionamento do mecanismo desenvolvido e a partir de então aperfeiçoar suas relações, de modo a gerar resultados mais aproximados de um projeto de arquitetura.

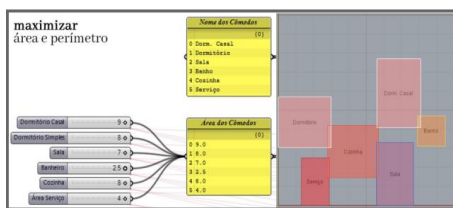


Figura 4: Maximizar Área e Perímetro. Fonte: autores (2019).

Após diversos estudos e experimentações foram definidos os parâmetros e intenções norteadoras da geração de soluções do mecanismo. São eles: (a) maximizar área dos cômodos; (b) maximizar área útil de projeto, buscando sobreposição dos cômodos igual a zero; (c) maximizar proximidade entre cômodos; (d) maximizar proximidade com pontos de atração de conforto ambiental. A seguir é apresentada a sequência de etapas deste processo, destacando os objetivos, parâmetros e estratégias adotadas em cada momento da construção do mecanismo gerador de soluções.

Estudo 1: criar relações a partir do *plugin* SpaceSyntax, (utiliza a teoria dos grafos como base para o seu funcionamento). Estratégia do *plugin*: Construir retângulos adjacentes; nomear os cômodos; traçar linhas que definem as relações entre os cômodos a partir do centróide dos retângulos; definir a área mínima e máxima dos cômodos; reposicionar os centróides e construir um diagrama; posicionar uma representação dos cômodos (desenho de circunferências) a partir do diagrama.

Estudo 2: Definição dos primeiros objetivos. Critérios de avaliação adotados: Maximizar área dos cômodos; obter o valor das áreas a partir da decomposição do raio das circunferências desenhadas pelo *plugin*; minimizar área perdida com a sobreposição de circunferências; minimizar área livre no lote, buscando a maior área possível para os cômodos.

Estudo 3: Avaliação das etapas anteriores. Problemas encontrados: o *plugin* limita a posição dos cômodos e a geometria. Assim, com o intuito de aproximar o diagrama de um desenho de planta baixa, pretende-se trocar a circunferência desenhada automaticamente pelo SpaceSyntax por retângulos.

Estudo 4: Converter circunferência em retângulo de modo a tornar o resultado mais próximo do leilante de uma planta de arquitetura. Objetivo: Criar quadrados circunscritos na circunferência. Problemas encontrados: movimentação limitada pelo *plugin* SpaceSyntax.

Estudo 5: Gerar a movimentação nos eixos X e Y da posição dos cômodos no lote e extrusão de perímetro para simulação energética nas faces. Critérios de avaliação adotados: minimizar a radiação solar incidente sobre as faces; maximizar a área dos cômodos; aumentar a área útil do projeto (evitar sobreposições entre os cômodos). Problemas encontrados: processamento de simulação muito lento, que resulta na geração de poucos indivíduos em muito tempo de processamento; área reduzida em função do perímetro (quanto maior o perímetro maior a incidência de radiação solar); cômodos saindo do perímetro do lote quando são movimentados.

Estudo 6: Inserção do parâmetro de rotação dos cômodos e, posteriormente, a rotação da solução gerada. Critérios de avaliação adotados: maximizar área útil de projeto; aumentar área dos cômodos; reduzir área perdida (evitar sobreposição de áreas ou cômodos fora do lote); reduzir o perímetro; reduzir a radiação nas superfícies. Problemas encontrados: formas irregulares a partir da rotação; criação de vazios dentro da forma da edificação; tempo elevado de processamento para geração de soluções, devido à simulação. Aspectos positivos: facilidade na visualização das superfícies mais afetadas pela radiação devido à coloração criada pelo *plugin* Ladybug.

Estudo 7: Desenvolver a relação entre os cômodos e pontos específicos no lote. Objetivo: Buscar estratégias para definir a posição dos cômodos no lote a partir de pontos de atração, a fim de reduzir o tempo de processamento exigido pela simulação. Em vez de simular a incidência solar nas superfícies da edificação, adotou-se como estratégia a orientação pelos pontos cardeais.

Estudo 8: Mudança de estratégia: Avaliação a partir dos pontos cardeais com valores de radiação solar, no lugar da simulação da radiação nas superfícies. Critérios de avaliação adotados: seleção de cômodos que terão pior e melhor posição; relação de atração entre pior/melhor ponto de radiação com pior/melhor cômodo (neste caso pior e melhor são os pontos com maior e menor radiação respectivamente). Problemas encontrados: os pontos cardeais criados possuem distâncias diferentes com relação ao lote o que gerou dificuldade em posicionar os cômodos próximos aos pontos indicados; *plugin* SpaceSyntax impede a movimentação dos cômodos livremente pelo lote.

Estudo 9: Mudança de estratégia: O *plugin* SpaceSyntax deixa de ser utilizado a fim de garantir a livre movimentação dos cômodos no perímetro interno do lote. Objetivos: decomposição do processo anteriormente realizado com auxílio do *plugin* SpaceSyntax. (definição de área, criação de circunferências para representar os cômodos, criação de quadrado circunscrito, movimentação desses quadrados no eixo X e Y, rotação após a movimentação, criação de relações a partir dos centróides para reduzir a distância entre os cômodos). Problemas encontrados: na realização dos experimentos, ao manter as áreas e a rotação dos cômodos fixas os resultados obtidos foram mais satisfatórios, com melhor posicionamento e menor perda de área; na busca por minimização da pontuação das soluções (valor de *fitness*), a solução tende a buscar uma área menor (reduzindo a área dos cômodos ou sobrepondo as suas áreas). Estratégia específica utilizada: geração em duas fases. Posição dos cômodos desconsiderando a área de serviço; para posteriormente posicionar a área de serviço.

Estudo 10: Mudança de estratégia: a estratégia anterior foi abandonada por não apresentar clareza na avaliação das soluções. Ao trabalhar com a maximização do perímetro (Figura 4), os cômodos tendem a se "repelir", tocando-se minimamente a fim de gerar o maior perímetro possível; passou-se a tolerar uma porcentagem de sobreposição de áreas, sendo penalizadas apenas as soluções que ultrapassassem um limite específico.

Estudo 11: Utilização de retângulos. Definição de proporção entre os lados do retângulo a partir da área; equilibrar as pontuações obtidas em cada parâmetro, para evitar que um parâmetro tenha peso maior que outro (a menos que seja desejado).

Estudo 12: Criar uma relação entre os pontos cardeais e a radiação solar em função do perímetro do lote. Para isso foram definidas linhas auxiliares, conectando os pontos cardeais com o centro do lote, as intersecções destas linhas com o perímetro do lote deram origem aos pontos de atração de conforto; limitação da movimentação dos cômodos dentro do lote; aperfeiçoamento no sistema de pontuações.

Estudo 13: Avaliação dos estudos até o estágio atual. Remoção do movimento de rotação dos cômodos; aperfeiçoamento no sistema de pontuações.

Estudo 14: Refinamento do código para realizar as experimentações finais: maximizar área de cômodos; maximizar área útil de projeto (buscar sobreposição zero

entre os cômodos); minimizar afastamento entre cômodos (buscar distância zero entre cômodos); minimizar os afastamentos dos cômodos (buscar distância zero) em relação aos pontos de atração de conforto que oferecem melhor conforto térmico para cada atividade definida no programa de necessidades.

Após a definição dos parâmetros a partir das intenções do projeto e do refinamento do código, foram executados os experimentos de geração de soluções. Os parâmetros adotados para essa experimentação foram: minimizar as sobreposições entre os cômodos, minimizar a distância entre os cômodos a qual deveria existir uma relação de adjacência (Figura 5), minimizar a distância entre pontos de orientação solar em função das atividades mais favoráveis a cada insolação e minimizar o perímetro. A área dos cômodos foi mantida fixa para este estudo.

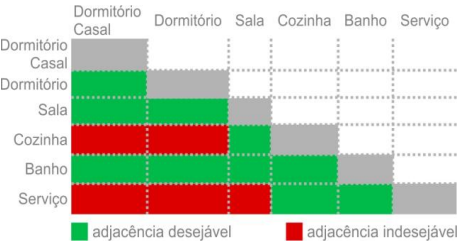


Figura 5: Relações de adjacência. Fonte: autores (2019).

A Figura 6 apresenta o mecanismo em diferentes etapas de desenvolvimento, onde é possível perceber a evolução das soluções ao longo das gerações. A programação foi realizada de maneira que se deseja obter o menor valor de *fitness*, ou seja, o algoritmo genético deve buscar a minimização, o que acaba resultando muitas vezes em valores negativos.

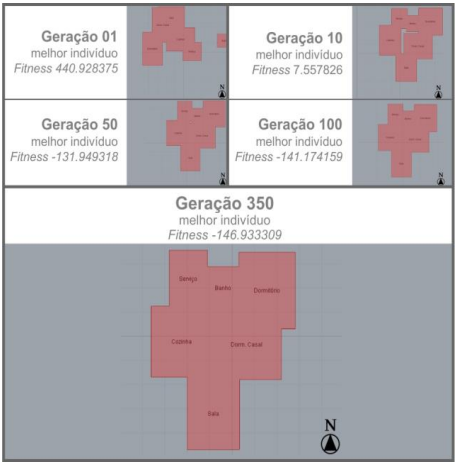


Figura 6: Melhores indivíduos da 1ª, 10ª, 50ª, 100ª e 350ª geração e seus respectivos valores de fitness. Fonte: autores (2019).

A geração de soluções foi encerrada automaticamente na geração de número 350. Entre a melhor solução da 100ª e a da 350ª geração houve pouca variação no valor de

fitness, sendo verificada pouca variação na posição dos cômodos em relação ao lote. Em casos como esse, onde ocorre a convergência, obtendo resultados muito semelhantes, o projetista pode interromper o processo de geração antes do encerramento automático, evitando a perda de tempo e processamento desnecessário. Na Figura 6 são apresentados os melhores valores de *fitness* da 1ª, 10ª, 50ª, 100ª e 350ª geração e os seus respectivos resultados.

CONCLUSÃO

As soluções geradas pelo sistema generativo proposto caminham para a obtenção de bons resultados. Isto pode ser verificado pelo correto posicionamento dos cômodos em relação à orientação solar. Entretanto, ao avaliar as relações de adjacência, percebe-se que é necessário aprimorar o sistema devido a sua instabilidade. A repetição do experimento apresentou diferentes resultados neste quesito, ora os cômodos estão bem posicionados com relação à adjacência, ora estabelece relações indesejáveis. Esta inconstância revela a necessidade de rever a estratégia que define as adjacências. Porém, todos os experimentos apresentaram um posicionamento adequado em relação à insolação.

Um fator que pode ter influenciado nos resultados é o fato de o componente Galápagos utilizar um único valor para a avaliação das soluções, não distinguindo a relevância de cada parâmetro. Por exemplo, a distância entre os cômodos e as suas respectivas áreas são somadas e o resultado é o valor de *fitness* a ser analisado pelo Galápagos. Acredita-se que a avaliação individual para cada parâmetro tornaria o sistema mais eficiente. Isso favorece a otimização de cada parâmetro, não sendo mais a soma ou a subtração dos diferentes valores dos parâmetros envolvidos no sistema, mas sim a pontuação individual de cada parâmetro. Sendo assim, o grande desafio na criação deste mecanismo foi o de garantir o equilíbrio entre a pontuação dos parâmetros, o que exigiu a aplicação de "penalidades" em determinados parâmetros. A percepção dessa limitação torna-se referência para a reestruturação do mecanismo proposto e a realização de novos experimentos.

É importante perceber que o mecanismo em algumas ocasiões acaba gerando resultados que não estavam previsto no processo de construção do mecanismo. Este evento é descrito por Holland (1998) como emergência, pois, apesar de as regras definidas pelo projetista serem rígidas, o computador acaba produzindo resultados não esperados pelo projetista, revelando novas formas de percepção e de relação entre os parâmetros que fogem do seu repertório.

Para pesquisas futuras é importante considerar os modos de pontuar ou penalizar os parâmetros mais importantes ao projeto, de modo a garantir uma hierarquia na avaliação ou a avaliação individual de cada parâmetro como já citado. Além disso, a fim de obter uma melhor continuidade e aproveitamento do *leiaute* gerado, é interessante também explorar a representação da circulação interna. Assim, o resultado gerado pelo

mecanismo será o mais próximo possível do que é necessário para a produção de uma planta baixa que representa o projeto arquitetônico. A complexidade do sistema pode ser aumentada locando portas e janelas, estabelecendo novos fluxos de dados e informações que podem ser articulados com o *leiaute* gerado, ou até mesmo auxiliarem na definição deste.

AGRADECIMENTOS

Os autores agradecem à FAPES pela concessão da bolsa ao autor Diego Jami Menezes da Silva para a realização desta pesquisa no Programa Institucional de Iniciação Científica da Universidade Federal do Espírito Santo.

REFERÊNCIAS

- Brigitte, G. T. N., & Ruschel, R. C. (2016). Modelo de informação da construção para o projeto baseado em desempenho: caracterização e processo. *Ambiente Construído*, 16 (4), 9-26. doi: 10.1590/s1678-86212016000400102.
- Calixto, V. (2016). Geração automatizada de layouts com o uso de algoritmos evolutivos: aplicações em arquitetura e urbanismo. Campinas: Autor.
- Calixto, V. (2018). Geração automatizada de *leiautes*: uma revisão da literatura. In M. R. Sedrez, & G. Celani (Eds.). *Arquitetura contemporânea e automação: prática e reflexão* (pp. 85-98). São Paulo: ProBooks.
- Corbella, O., & Yannas, S. (2009). Em busca de uma arquitetura sustentável para os trópicos: conforto ambiental. Rio de Janeiro: Revan.
- Hensel, M., Menges, A., & Weinstock, M. (2013). *Emergent technologies and design: towards a biological paradigm for architecture*. New York: Routledge.
- Holland, J. H. (1998). *Emergence: from chaos to order*. Cambridge: Perseus.
- Kolarevic, B. (2005). Towards the performative in architecture. In B. Kolarevic, & A. Malkawi (Eds.). *Performative Architecture: Beyond Instrumentality* (pp. 204-213). New York: Spon Press.
- Kowaltowski, D. C. K., Labaki, L. C., Pina, S. M. G., & Bertolli, S. R. (1998). A visualização do conforto ambiental no projeto arquitetônico. *Anais do VII Encontro de Tecnologia do Ambiente Construído e Qualidade no Processo Construtivo*, 27-30.
- Liggett, R. S. (2000). Automated facilities layout: past, present and future. *Automation in construction*, 9(2), 197-215. doi: 10.1016/S0926-5805(99)00005-9
- Martino, J. A. (2015). Modelagem por meio de dados informacionais utilizando algoritmos evolutivos. *Blucher Engineering Proceedings*, 2(2), 87-89. Doi: 10.5151/engpro-ctic2015-008
- Mitchell, W. J. (1975). The theoretical foundation of computer-aided architectural design. *Environment and planning b: planning and design*, 2(2), 127-150. doi:10.1068/b020127
- Oxman, R. (2006). Theory and design in the first digital age. *Design studies*, 27(3), 229-265. doi:10.1016/j.destud.2005.11.002
- Scheer, D. R. (2014). *The death of drawing: architecture in the age of simulation*. New York: Routledge.
- Yeang, K. (1999). *The Green Skyscraper: The Basis for Designing Sustainable Intensive Buildings*. New York: Prestel.