

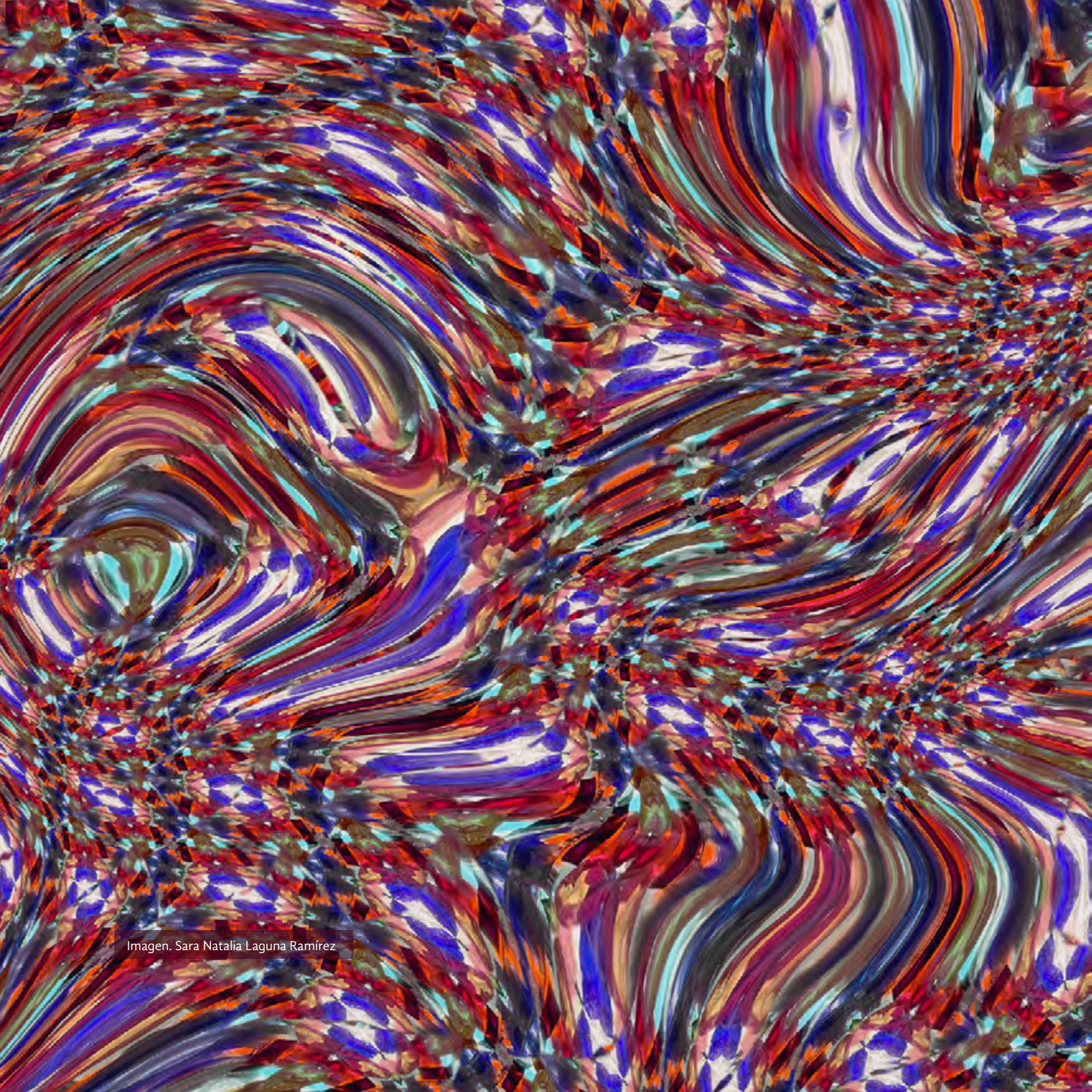


Imagen. Sara Natalia Laguna Ramírez

Propuesta de proceso de diseño generativo aplicado en proyecto arquitectónico experimental

Proposal of generative design process applied in experimental architectural project

Valentyn-Vladyslav Kotsarenko *

Arquitecto Coordinador BIM, arquitecto por la licenciatura en arquitectura de la Universidad Autónoma de Querétaro (2019), dedicado a la optimización de procesos de diseño y producción aplicados al sector AEC con la tecnología BIM y el diseño generativo, certificado por Graphisoft en Building Information Modeling con ArchiCAD (2018). Cuenta con más de 8 años de experiencia de colaboración en proyectos de arquitectura, ingeniería, construcción, diseño gráfico, audiovisual, entre otros; y 7 años en la docencia y capacitación de temas relacionados en instituciones y empresas. Ganador de concursos de arquitectura como Nucleo Urbano y otros, concluye estudios de maestría en arquitectura en la UAQ (2022).

Avatar Flores Gutiérrez **

Es arquitecto por el ITESM (2000), maestro en diseño de producto por Elisava Escola Superior de Disseny de Barcelona (2007), y doctor en arquitectura por la Universidad Nacional Autónoma de México (2016). Es profesor e investigador en la Universidad Autónoma de Querétaro, donde coordina la Maestría en Arquitectura. Participa activamente dando clases y dirigiendo tesis a nivel licenciatura y posgrado en las áreas de Arquitectura y Diseño. Miembro del Sistema Nacional de Investigadores, su principal motivación académica es abordar la complejidad humana en los procesos de diseño, específicamente centrada en el fenómeno arquitectónico y las herramientas que permitan la conformación de espacios habitables. Ha desarrollado también un buen número de proyectos arquitectónicos, tanto del ámbito público como del privado.

Oscar Daniel Soto Morales ***

Arquitecto por la Universidad Autónoma de Querétaro (UAQ) (2017) artista plástico y poeta queretano, con especial interés por el área de las gráficas como la pintura, la serigrafía de carácter social, la literatura y poesía oscurantista y latinoamericana. Es editor creative en la editorial N.E.L. Ha desarrollado proyectos de muralismo participativo y de arquitectura tanto en el ámbito público como privado, así como participado en exposiciones colectivas de artistas emergentes realizadas dentro de las instalaciones de A13, Ángulo 13 (2019) Casa Osla (2021) y en el Museo de la Ciudad Qro. (2022). Fue ganador de concursos de arquitectura como Nucleo Urbano. Actualmente concluye estudios de maestría en arquitectura en la UAQ (2022).

Resumen

El progreso tecnológico y la digitalización han generado nuevas oportunidades para el diseño arquitectónico, y entre distintas tecnologías emergentes de las últimas décadas resalta el tema del diseño paramétrico, cuya aplicación será explorada en el presente artículo. El objetivo del trabajo es exponer y describir las ideas principales, metodología, resultados, conclusiones, entre distintos puntos relevantes en la programación del proyecto experimental denominado *Callejuelas*, resultado generativo que emerge a través de la automatización de procesos de diseño arquitectónico. La metodología nace con la definición de un problema de vivienda, seguido con la conceptualización del proyecto y la programación visual de 14 subalgoritmos íntegros de procesos análogos y técnicos; como efecto generarán de manera automatizada resultados que, bajo determinadas constricciones y criterios, resuelven problemas conceptuales, espaciales, funcionales, estructurales, ambientales entre otros presentes para el desarrollo del proyecto. Mientras que el diseño paramétrico y generativo presentan ventajas como desventajas, se puede apreciar que además de disponer de la oportunidad de generar un número amplio de iteraciones del proyecto al instante, es una alternativa de diseño que permite una toma de decisiones más rápidas y eficientes referentes al diseño de un proyecto en sus distintas especialidades.

Palabras clave: algoritmo, arquitectura, diseño, experimental, generativo, paramétrico, programación, proyecto, software.

Abstract

Technological progress and digitization have generated new opportunities for architectural design, and among different emerging technologies of recent decades, parametric design stands out, concept which will be explored in the present writing. The objective of the work is to expose and describe the main ideas, methodology, results, conclusions, among different relevant points in the programming of the experimental project named *Callejuelas*, a generative result that emerges through the automation of architectural design processes. The methodology is born by defining a housing problem, followed by the conceptualization of the project and the visual programming of 14 sub-algorithms which include analogous and technical processes; as an effect, they will automatically generate results that, under certain constraints and criteria, solve conceptual, spatial, functional, structural, environmental problems, among others present for the development of the project. While the parametric and generative design process presents advantages as well as disadvantages, it can be seen that in addition to having the opportunity to generate a large number of iterations of the project instantly, it is a design alternative that allows faster and more efficient decision making regarding to the design of a project in its different specialties.

Keywords: algorithm, architecture, design, experimental, generative, parametric, programming, project, software.

Introducción

En la actualidad, un momento en el cual las tecnologías computacionales y la digitalización, además de influir parte del estilo de vida diario, reestructuran la forma de pensar y trabajar en distintos sectores profesionales como el diseño arquitectónico. Asimismo, el diseño paramétrico y el generativo se distinguen como herramientas producto del progreso tecnológico, que presentan el potencial de transformar el proceso de diseño estático en dinámico, con la principal ventaja de generar un número indefinido de iteraciones al mismo problema en tiempos récord en comparación al de un proceso de diseño fijo.

En el presente artículo se exploran los orígenes, definiciones y constructos intelectuales sobre el tema del diseño paramétrico, así como las derivadas adicionales que representan directrices para la conceptualización, desarrollo y programación del proceso de diseño generativo aplicado a un proyecto de diseño arquitectónico experimental denominado *Callejuelas*, resultado que emerge a partir del procesamiento de datos por medio de un proceso estructurado por 14 subalgoritmos, desarrollados bajo las funciones de herramientas digitales como Rhinoceros®, Grasshopper®, ArchiCAD®, LadyBug, Galapagos, entre distintas herramientas y complementos adicionales.

Además de documentar los procesos clave del proyecto generativo, se reflexionan los resultados finales de la propuesta de diseño seleccionada y se incluyen las conclusiones más relevantes referentes a las ideas sobre el diseño paramétrico, el procedimiento de diseño del proyecto y el resultado final.

Principios del diseño paramétrico y generativo

Cuando se exploran los orígenes del concepto de diseño paramétrico se observa que radica analógicamente desde antes de que fuera posible el uso del ordenador y los medios digitales. Como lo explica Davis (2013), el término paramétrico proviene de una naturaleza matemática siglos atrás y Weisstein (2002: 2150) define las ecuaciones paramétricas “como un conjunto de ecuaciones que expresan un grupo de cantidades siendo funciones explícitas de una serie de variables independientes, conocidas como parámetros”; mientras que se desconoce quién fue el primer autor en manejar dichos términos, se puede encontrar la inclusión de dicho término desde trabajos de Leslie (1821 citado en Davis, 2013) –físico y matemático escocés– quien maneja el vocablo en una de sus obras para la descripción de procesos matemáticos, con el fin de explicar la relación de partes en catenarias para su respectiva modificación geométrica; Dana (1844 citado en Davis, 2013) –mineralogista estadounidense– en una de sus obras usa la expresión para documentar indicaciones sobre la representación de cristales gemelos entre distintas gamas de elementos con el uso de un lenguaje matemático vinculado con variables, proporciones y parámetros; Gómez-Moriana (2012) indica que la forma de trabajo del arquitecto español Antoni Gaudí en 1889 se asemeja al proceso de diseño paramétrico, ya que por medio de modelos de cadenas colgantes, Gaudí resolvía los criterios estructurales de sus proyectos a modo pragmático; Zexin y Mei (2017) indican que el arquitecto Alemán Frei Otto en 1950 también protagonizó el concepto con la apropiación del término *form-finding* por medio del diseño de cubiertas ligeras a base de modelos de cadenas colgantes–similar a como lo hacía Gaudí– y la generación de geometrías por medio de pompas de jabón; y según Imperiale (2018) fue en 1960 cuando el arquitecto italiano Luigi Moretti –con un perfil tanto en matemáticas aplicadas como arquitectura– desarrolló el diseño de un estadio paramétrico cuya forma deriva a partir de diecinueve parámetros, por medio de algoritmos programados en el ordenador IBM 610, experimentos que presentó en su exposición de Arquitectura Paramétrica en la Duodécima Trienal de Milán en 1960, posterior a dicho acontecimiento y según Davis (2013) se observan más acontecimientos en la aplicación de herramientas paramétricas más complejas debido al progreso tecnológico y la digitalización, hasta llegar al concepto del diseño paramétrico como se puede observar el día de hoy con el uso de instrumentos más complejos.

Cuando se exploran las definiciones de los conceptos referentes al diseño paramétrico sobresalen distintas doctrinas de pensamiento que contextualizan las directrices que rigen el desarrollo del proyecto de diseño arquitectónico descrito posteriormente. Christopher Alexander –arquitecto y teórico– (Dawes y Ostwald, 2017) presenta la idea del patrón como un elemento que describe un problema que ocurre una y otra vez en determinado contexto y, posteriormente, describe el núcleo de la solución a ese problema, de tal forma que dicho resultado se puede manejar más de un millón de veces sin obtener el mismo desenlace dos veces; además Dawes y Ostwald (2017: 2) resaltan la creencia e inspiración de Alexander expresando que las viviendas de sociedades tradicionales son inherentemente más “bellas” que la arquitectura contemporánea, debido a que se manejan distintos procesos de diseño, en el que los edificios tradicionales son el producto de un sistema de valores compartidos por la

comunidad, así como una adaptación gradual a las circunstancias cambiantes, lo que hace que todas las “fuerzas”, del contexto, que impactan un diseño entren en un equilibrio armonioso; dicha idea se correlaciona con la perspectiva de Schumacher

(2008) –arquitecto principal del despacho de Zaha Hadid Architects–, quien expresa que el parametricismo, concepto que además de correlacionarse al diseño paramétrico en parte, presenta también un grado de complejidad mayor, emerge con la explotación creativa de sistemas de diseño paramétrico con el fin de articular procesos e instituciones sociales cada vez más complejas, siendo el sentido de complejidad organizada la principal caracterización de la arquitectura parametricista, en el que las formas son el resultado de fuerzas, del contexto, que interactúan legalmente. Lars Hesselgren (Stocking, 2009), indica que en el diseño generativo no se trata de resolver un edificio sino el sistema que genera la obra; asimismo, Gutiérrez (2016: 65) asocia la visión de estructuras de pensamiento manejadas en los procesos de diseño refiriéndose al diseño paramétrico como herramienta epistemológica; Kolarevic

...el parametricismo,(...), presenta también un grado de complejidad mayor, emerge con la explotación creativa de sistemas de diseño paramétrico con el fin de articular procesos e instituciones sociales cada vez más complejas...

(2003: 25) expresa que la paramétrica puede proporcionar una concepción poderosa de la forma arquitectónica al describir un rango de posibilidades, reemplazando en el proceso estable con variable, singularidad con multiplicidad, permitiendo así a los diseñadores crear un número infinito de objetos y manifestaciones geométricas similares; Kowalski (2016) –exjefe de tecnología de Autodesk®–, expresa su perspectiva del diseño generativo con el ejemplo de una silla, comenta que consiste en indicar a la computadora parámetros, reglas o criterios que debe cumplir cierto diseño, y la computadora ofrece no miles, sino millones de soluciones que cumplen con los presentes criterios, posteriormente el diseñador selecciona la opción que considera que cumple los criterios más importantes y procede a mejorar el diseño a base de su propio criterio o, en su debido caso, se procede a la materialización o la fabricación directa del resultado. Por último, Archistar (s.f.) indica que el diseño

paramétrico es un proceso interactivo que permite crear diseños basados en la entrada de parámetros, y el diseño generativo se basa en la inclusión de computadoras e inteligencia artificial, siendo un proceso iterativo que utiliza algoritmos avanzados para encontrar la mejor solución, en la mayoría de los casos

el resultado basado en objetivos numéricos, a un problema.

Mientras que diversos autores presentan constructos intelectuales y percepciones distintas sobre el tema, se puede observar que sus ideas comparten una directriz en la forma de pensar y trabajar, así como la naturaleza del proceso con el uso de las herramientas paramétricas, mientras que las diversas definiciones y derivadas del tema enriquecen como expanden la complejidad de la trama, se manejará el término del *diseño paramétrico* para referirse a la acción de diseñar un proceso con el manejo de parámetros, y el *diseño generativo* para referirse a la integración de algoritmos y componentes más complejos al presente proceso que aprovechan el potencial de la computadora para la búsqueda de la mejor solución, basada en objetivos numéricos, a un determinado problema.

*...el rol del arquitecto se va acercando todavía más a la noción de un **creador de sistemas generativos** y no de un solucionador técnico de un programa arquitectónico...*

Rol del diseñador

Una vez mencionado, el diseño paramétrico es ya comúnmente utilizado en la práctica del diseño arquitectónico, sin embargo, puede decirse que se está aún en una etapa muy temprana del desarrollo como herramienta de diseño generativo. Por ejemplo, es ya evidente el potencial que tienen estas herramientas para el manejo de aspectos concretos, como la modularidad de los materiales o varios aspectos del diseño bioclimático como las orientaciones, o volúmenes de aire al interior de los espacios. Sin embargo, el uso de los métodos paramétricos para el manejo de criterios de diseño más complejos como las vistas que se tendrán desde un espacio hacia el exterior, o principios de organización espacial en función de criterios de percepción, están aún en experimentación.

Desde un paradigma sistémico del diseño arquitectónico, la conceptualización sistémica concluye precisamente con la definición de criterios o parámetros de diseño (Flores, 2020). En este sentido, el rol del arquitecto se va acercando todavía más a la noción de un creador de sistemas generativos y no de un solucionador técnico de un programa arquitectónico. Las herramientas paramétricas habilitan la exploración de estos paradigmas, contribuyendo muy probablemente a la evolución del rol del arquitecto en un futuro cercano.

Si bien el diseño generativo habilita este potencial, es necesaria la experimentación con parámetros más complejos que también evolucionen al diseño paramétrico con el uso y manejo de conceptos más complejos a los que ya se dominan en la actualidad. Es precisamente por ello que en este artículo se ha buscado explorar con algunos parámetros que dan cuenta de las posibilidades que hay con el uso de estas herramientas.

Posibilidades del diseño generativo

Cuando se habla sobre las posibilidades que oferta el diseño generativo, uno de los ejemplos destacados resalta con la propuesta desarrollada por Holland (2011), en el que diseña un proceso que genera un edificio de gran altura con la incorporación y manejo de algoritmos que procesan tanto las constricciones de ubicación, distribución espacial, visual entre distintos criterios que impactan tanto la forma, el costo, la funcionalidad entre diversos aspectos del inmueble ubicado en la ciudad de Seattle, Washington. Con este algoritmo se generan diversas iteraciones formales a los problemas particulares que presenta el proyecto, además, cada solución está acompañada de datos que informan al diseñador sobre las ventajas y desventajas de cada propuesta (ver figura 1, sig. pag.). Dicho ejemplo, además de poner en perspectiva la capacidad del diseño generativo para la resolución de un edificio de gran altura en la mayoría de sus aspectos, permite establecer una idea del papel que desempeñan las iteraciones íntegras de información cuantificable, datos que expresan los comportamientos funcionales de cada forma, permitiendo así establecer fundamentos más sólidos en la elección de una geometría sin estar únicamente limitados a criterios estéticos. Otro ejemplo resalta con la propuesta expuesta por Terzidis (2008: 84), en el que realiza un algoritmo de búsqueda estocástica que encuentra el espacio disponible para distribuir habitaciones del programa arquitectónico bajo las constricciones, limitantes del sitio, así como la matriz de adyacencia (ver figura 2, sig. pag.); en otras palabras, se podría referir a un *zonificador de espacios automatizado*. La idea por Terzidis, que en paralelo se comparte con la visión y naturaleza de la propuesta por Holland, percibe la perspectiva de cómo los arquitectos y diseñadores de la actualidad pueden explotar las tecnologías para delegar procesos funcionales programables a la computadora, y así enfocarse en tareas que requieren de intervención humana como pueden ser criterios estéticos, sociales, éticos, comportamientos fenomenológicos, entre diversos factores que requieren de la sensibilidad humana.

Por último, otro ejemplo que permite concebir una idea de cómo las tecnologías evolucionan más las posibilidades de diseño se presenta con la herramienta generativa del *startup* sueco denominado Finch3D®, cuyo *software* permite la

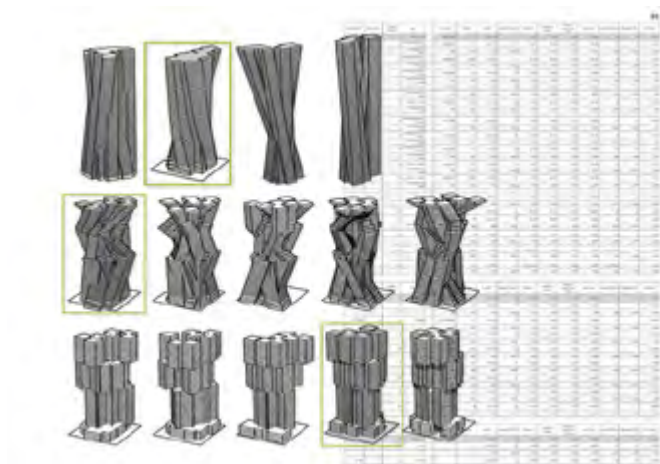


Figura 1. Extracto de gráfico que representa las iteraciones de un edificio de gran altura, íntegras de datos cuantificables para conocer su funcionalidad. Fuente: Holland (2011).

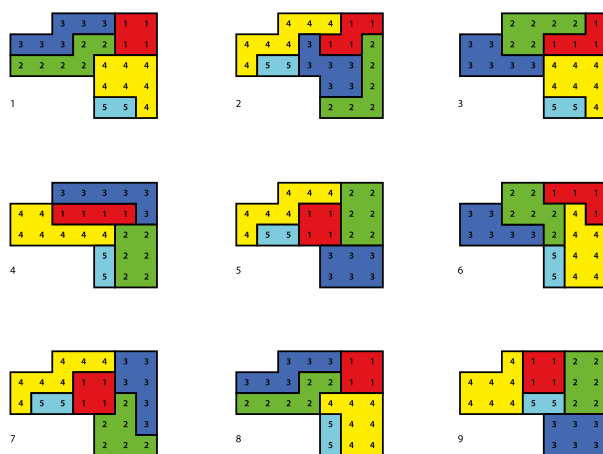


Figura 2. Extracto de gráfico sobre resultados obtenidos con algoritmo de búsqueda estocástica. Fuente: Terzidis (2008).

automatización de tareas repetitivas –como la distribución en tiempo real del mobiliario en una forma, con el fin de evaluar la funcionalidad de un espacio– y guía a lo largo del proceso de diseño con la implementación de la inteligencia artificial, con el fin de permitir a los profesionales dedicar tiempo a la creación y evaluación del diseño (McSweeney, 2020).

Herramientas

Rhinoceros 3D

La herramienta llamada Rhinoceros 3D (Robert McNeel & Associates, 2021), Rhino o también Rhino3D, es una herramienta comercial de aplicación CAD (por sus siglas en inglés *Computer-aided design*) y de gráficos 3D por computadora, desarrollado por Robert McNeel & Associates; el presente instrumento basa la geometría desarrollada en el modelo matemático NURBS (por sus siglas en inglés *Non-uniform rational B-spline*) con el fin de generar una representación matemáticamente precisa tanto de curvas como de superficies de forma libre, esta herramienta también se utiliza en procesos CAM (por sus siglas en inglés *Computer Aided Manufacturing*), creación rápida de prototipos, impresión 3D e ingeniería inversa en industrias de arquitectura, diseño industrial, diseño de productos, multimedia, gráficos, entre diversos otros.

Grasshopper 3D

Grasshopper 3D nace como un complemento para Rhinoceros 3D en forma de lenguaje de programación el cual fue desarrollado por David Rutten en Robert McNeel & Associates y lanzado en el año 2008. El lenguaje programático forma parte del paradigma de programación basado en flujo (también conocido como FBP o por sus siglas en inglés *Flow-Based Programming*), como lo indica Al-Fedaghi (2015, pág. 209) se trabaja con una red de componentes predefinidos denominados como *black boxes*, en los cuales la información se intercambia por conexiones especificadas externamente a los procesos; este lenguaje se trabaja de manera visual y a la par integra posibilidades y funciones

de la programación textual permitiendo también modalidades de trabajo híbridas (MODELAB, 2018). Asimismo, hay que resaltar que el instrumento de programación es flexible en su curva de aprendizaje al manejar un razonamiento intuitivo y lógico para el desarrollo de algoritmos, además oferta una posibilidad virtualmente infinita de soluciones, ya que los componentes se pueden intercalar de diferentes modos respondiendo ante un problema particular para generar solución completamente diferente.

ArchiCAD

ArchiCAD es un software para el manejo de la metodología de trabajo BIM (por sus siglas en inglés Building Information Modeling) desarrollado e implementado por la empresa húngara Graphisoft®, la cual se fundó por Gábor Bojár en 1982. Un modelo BIM consiste en un equivalente virtual de los elementos utilizados para la construcción de un edificio con características tanto físicas como lógicas de sus contrapartes reales, con el fin de lograr una simulación virtual precisa de un edificio para entender su comportamiento previo a su construcción, para así evitar problemas en obra permitiendo el desarrollo eficiente a lo largo del ciclo de vida de un proyecto.

analizado, las experiencias e historias escuchadas guían la narrativa y composición del proyecto cuyo propósito es diseñar una vivienda asequible en costo y servicios para estudiantes tanto locales como foráneos de ingreso económico bajo y medio, que estudian dentro de las universidades de carácter público ubicadas en las inmediaciones del centro de la ciudad de Querétaro.

En esta concepción se plantea proporcionar un hábitat que favorezca y esté de acuerdo con las necesidades de los futuros habitantes al cumplir con las expectativas de rentabilidad, inversión e identidad del contexto. Por ello, la comprensión del entorno y de los participantes sociales en sus diferentes escalas posibilita conocer las condicionantes cuantitativas y cualitativas para que, al momento de introducir a la herramienta los parámetros formales, nacidos de este análisis contextual, la herramienta genere soluciones formales diversas, para así, evaluar cuáles son las que están más en consonancia con las condicionantes dadas.

*Callejuelas surge como
un ejercicio experimental cuya idea,
nace de la resolución conceptual
del proyecto mismo...*

Conceptualización

En nuestra sociedad existe una diversidad compleja de personas y estratos sociales, no existen modos de vida iguales para todos, el problema al que se enfrenta el proyecto experimental de *Callejuelas*, es cómo entender y atender la relatividad de la condición contemporánea de un sector de la sociedad (Montaner y Muxí, 2011), para ello, se considera crear una herramienta paramétrica cuya conceptualización ayude a sintetizar el proceso de diseño, a modo de liberar al diseñador de la exigencia que requiere el hacer técnico, formal y constructivo del objeto, para así, dar el mismo peso al proceso en la parte analítica y evaluativa del contexto sociocultural.

Callejuelas surge como un ejercicio experimental cuya idea, nace de la resolución conceptual del proyecto mismo. En este caso la concepción se da gracias al registro sociocultural

Proceso técnico

Teniendo identificado el tipo de espacio a generar en conjunto con las limitantes y constricciones que intervienen, se procede a la programación del algoritmo con el objetivo de generar diversas iteraciones del proyecto arquitectónico que cumplan las expectativas de la conceptualización.

Mientras que el desarrollo de un algoritmo, considerando que cuando se habla de una programación ambiciosa, puede resultar una tarea demandante, se decide trabajar por bloques que se integran uno con otro para finalmente construir un algoritmo general; en otras palabras, atacar el problema en partes por medio de subalgoritmos que resuelven problemas de diseño particulares. Mientras que el desarrollo de la programación manejó en general un carácter pragmático, el proceso de razonamiento para la resolución y programación

de los diversos subalgoritmos se alineó a un procedimiento metódico (ver figura 3).

Dos etapas que hay que resaltar son la identificación del problema y la abstracción de la solución en una secuencia geométrica, ya que si no se tiene claridad en estos dos puntos que siguen manteniendo una naturaleza de trabajo analógica, el proceso técnico, la programación definitiva en Grasshopper, resulta nublado. Cuando se habla de la identificación de un problema de diseño particular se refiere a las fases particulares que toman vida en el proceso de diseño de un proyecto arquitectónico, como lo son la zonificación, la definición de accesos y las circulaciones, la integración del concepto, los criterios de iluminación y la ventilación natural, las colindancias, los criterios visuales, la distribución interior, la relación con siguientes pisos, entre diversos procesos que forman parte del recorrido que conlleva el desarrollo de una vivienda. Teniendo claridad sobre el problema de diseño particular que se busca atacar, es importante que ahora se deconstruya la solución formal final que responde a un problema en una secuencia

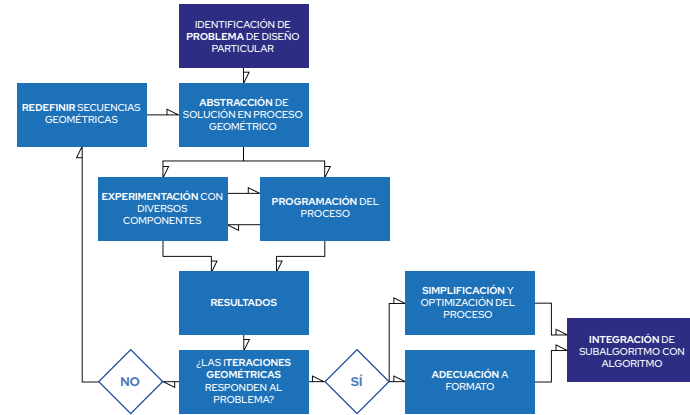


Figura 3. Proceso lógico para desarrollo de un subalgoritmo, en Grasshopper, que busca resolver el problema de diseño particular.
Fuente: elaboración propia.

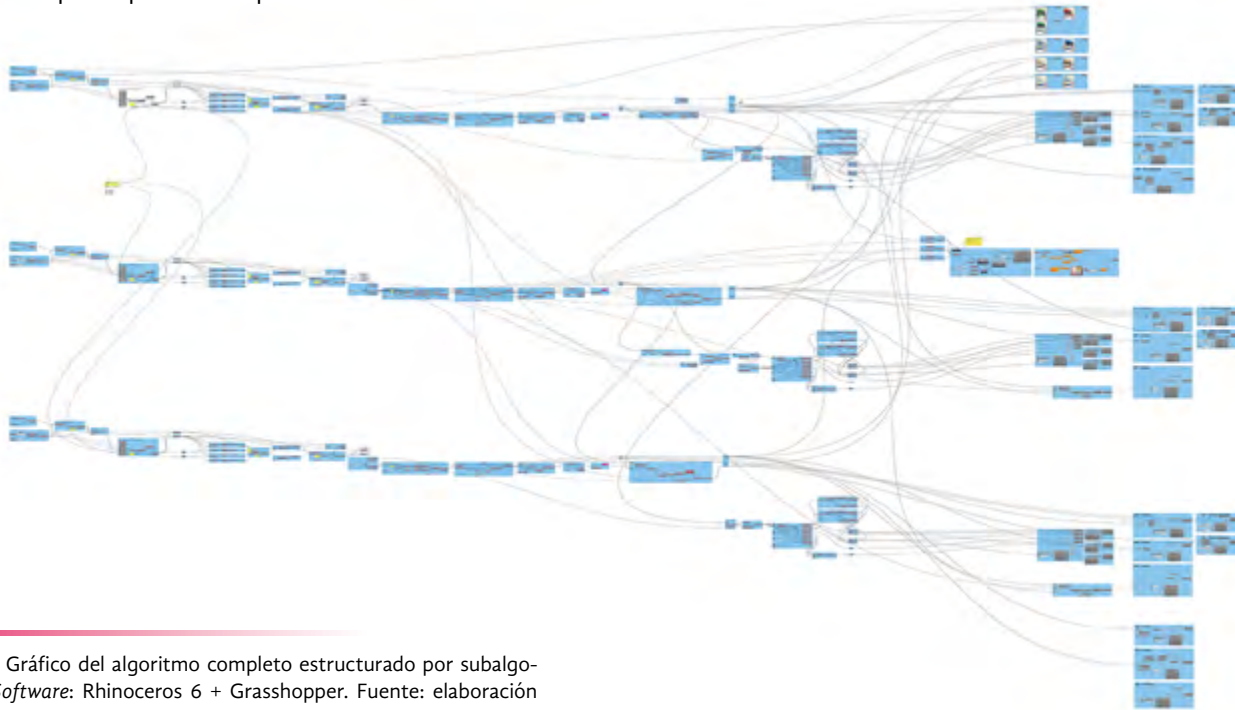


Figura 4. Gráfico del algoritmo completo estructurado por subalgoritmos. *Software:* Rhinoceros 6 + Grasshopper. Fuente: elaboración propia.

de pasos geométricos, dicho de otra forma, generar la ingeniería inversa a una solución formal (ejemplo: Paso 1. Generar un punto; Paso 2. Tomar el punto como centro para trazar un círculo; Paso 3. Extruir el círculo a una distancia variable); mientras que este proceso cobra mayor vida y claridad con la experiencia del programador como su respectivo conocimiento de los diversos comandos y soluciones de un lenguaje de programación, para el caso de la programación visual en Grasshopper es de ayuda plasmar la secuencia de pasos geométricos —generales— en un diagrama de flujo que resume un proceso para llegar a un resultado.

Teniendo lo anterior en cuenta, se procedió con el desarrollo de las 15 etapas que componen el proceso general para la generación del proyecto arquitectónico experimental Callejuelas, de las cuales unas tomaron vida dentro de la interfaz de Rhinoceros con Grasshopper con la programación de los subalgoritmos que componen el algoritmo general (ver figura 4), otros procedimientos se manejaron en ArchiCAD y asimismo algunos de estos integraron ciertas actividades de trabajo analógicas.

Etapa 1: Preliminares

Se da inicio al proyecto con el desarrollo de los análisis, estudios y procesos de forma epistemológica y a modo gráfico, sin necesidad de programar todavía con las herramientas paramétricas, esto con el fin de identificar las constricciones y limitantes que permitan contemplar parámetros, aspectos funcionales, espaciales, formales, técnicos, colindantes y todo tipo de criterios de diseño para la posterior programación de los subalgoritmos del proyecto, en otras palabras, en esta etapa se realizan los *preparativos* para la programación (ver figura 5 sig. pág.).

Etapa 2: Áreas Base

El primer subalgoritmo —programado en Grasshopper— nace con la definición general del terreno por medio de un polígono base, así como la delimitación de los espacios que se descontarán de éste definidos como las *áreas negativas* (marcadas con color verde en la figura 6, sig. pág., mientras que las *áreas positivas* se marcan en color rojo), que representan tanto espacios verdes, públicos, circulaciones libres

y toda zona restringida a ser intervenida por el algoritmo; dichos parámetros también definen los primeros inputs a ser especificados por el diseñador, quien libremente puede ajustar tanto el polígono del terreno como las *áreas negativas* para permitir así que el algoritmo genere en tiempo real el resultado.

El proceso en paralelo también genera sobre el polígono del terreno una retícula con la extensión necesaria para cubrir por completo la geometría, de tal manera que, si la forma del polígono es modificada, la red se actualizará para nuevamente cubrir por completo el espacio intervenido; la rejilla se dimensiona en función de las dimensiones de los módulos habitacionales que son de 4m de largo por 4m de ancho.

Etapa 3: SEED

El presente proceso ahora abstrae y plasma la intención del proyecto por medio de una distribución aleatoria de módulos. Se recupera el centro de cada rectángulo de la rejilla previa y se evalúa si éste se posiciona en el interior o exterior del *área positiva* del terreno, cada rectángulo arroja un valor booleano (verdadero o falso, 0 o 1) y con base en este dato se procede a eliminar todos los rectángulos que están fuera de los límites, obteniendo así únicamente los módulos que se posicionan dentro del *área positiva*.

Posteriormente, se vincula el gráfico del isotipo que representa el concepto de *Callejuelas*, previamente diseñado con el fin de representar la esencia general del proyecto; una vez referenciado el gráfico se continúa con un recuento de píxeles blancos y negros, para así obtener un porcentaje que representa el total de puntos blancos de la imagen. Se finaliza el proceso con el manejo del porcentaje para eliminar aleatoriamente los módulos rectangulares y reemplazarlos por vacíos (ver figura 7 sig. pág.).

Es importante mencionar que también se fija uno de los parámetros principales el cual se nombra como *SEED* (o semilla en inglés, bajo la idea de sembrar una nueva propuesta), definido por un dígito numérico que puede situarse entre 0 y 1 000 000; si el diseñador ajusta dicho parámetro a un valor distinto, los módulos rectangulares se filtrarán de forma distinta generando otra propuesta de distribución aleatoria.

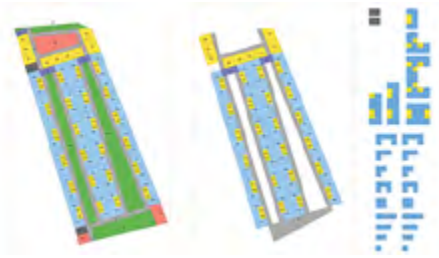


Figura 5.

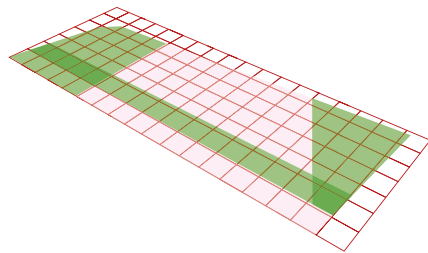


Figura 6.

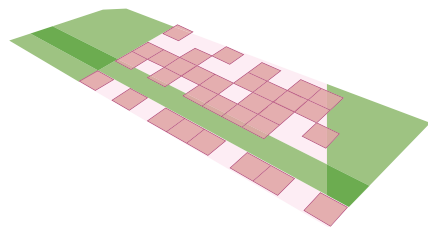


Figura 7.

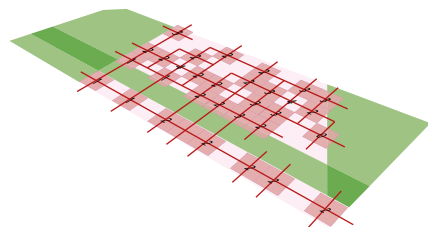


Figura 8.

Figura 5. Estudios y análisis preliminares previo al comienzo en la programación del algoritmo. *Software:* ArchiCAD 24. Fuente: elaboración propia.

Figura 6. Convergencia del polígono del terreno, áreas negativas marcadas en verde, áreas positivas marcadas en rojo y la retícula base. *Software:* Rhinoceros 6 + Grasshopper. Fuente: elaboración propia.

Figura 7. Filtro aleatorio de módulos en función del gráfico abstraído. *Software:* Rhinoceros 6 + Grasshopper. Fuente: elaboración propia.

Figura 8. Filtro de módulos con 4 colindancias. *Software:* Rhinoceros 6 + Grasshopper. Fuente: elaboración propia.

Figura 9. Criterio visual aplicado a la distribución aleatoria. *Software:* Rhinoceros 6 + Grasshopper. Fuente: elaboración propia.

Figura 10. Generación de pasillos considerando colindancias. *Software:* Rhinoceros 6 + Grasshopper. Fuente: elaboración propia.

Figura 11. Generación de eje de referencia para indicar frente en cada módulo. *Software:* Rhinoceros 6 + Grasshopper. Fuente: elaboración propia.

Figura 12. Generación de subdivisiones interiores y elementos adicionales de referencia. *Software:* Rhinoceros 6 + Grasshopper. Fuente: elaboración propia.

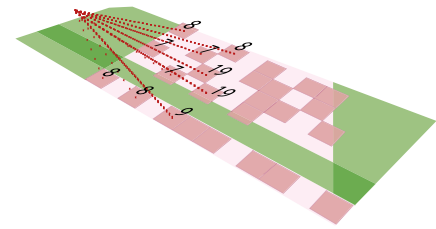


Figura 9.

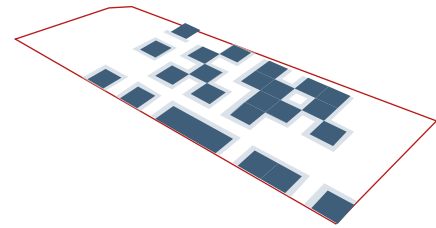


Figura 10.

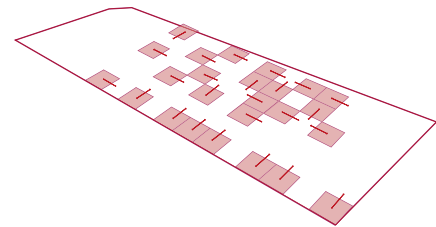


Figura 11.

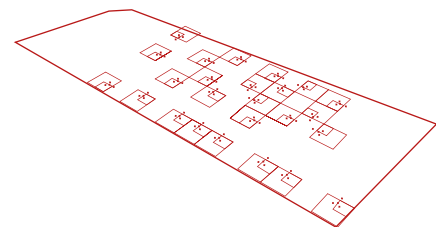


Figura 12.

Etapa 4: Colindancias

En este apartado se procura que todos los rectángulos representativos de un espacio habitacional cuenten con entradas de iluminación y ventilación natural. Se procede a recuperar el centro de cada rectángulo y a partir de este punto se proyectan ejes ortogonales en los 4 lados del módulo, procurando que éstos sean lo suficientemente largos para interactuar con las adyacencias de otros espacios y las colindancias del terreno; posteriormente, se miden cuántos de estos ejes colindan y se hace una suma para obtener un número de colindancias totales para cada módulo, de tal forma que cuando algún rectángulo sume 4 colindancias en total, se eliminará y reemplazará por un vacío, dejando presentes únicamente los espacios con 3 colindancias en total (ver figura 8, pág. ant.).

Etapa 5: Visual

El presente proceso busca todas las propuestas con las mejores condiciones visuales a un punto de interés. Se inicia con el planteamiento de un punto, en este caso en una de las áreas verdes del proyecto, posteriormente, se consideran únicamente 50% de los módulos rectangulares más cercanos al nodo de referencia y se procede con la proyección de una línea de cada uno de sus centros al sitio de interés.

El eje de referencia visual de cada módulo al punto de interés se divide equidistantemente, y se procede con la colocación de un punto en cada uno de sus segmentos, posteriormente, se evalúan y se suman cuántos de estos nodos quedan adentro de alguno de los módulos colindantes, en el sentido en que si un eje visual no tiene ningún obstáculo al punto de interés, el resultado de su evaluación será equivalente a 0 (ver figura 9, pág. ant.).

Recordando que existe un parámetro denominado como *SEED* que genera aleatoriamente un millón de posibles propuestas de distribución, el siguiente paso involucra la vinculación de un componente generativo gracias a un *plugin* (complemento de un programa) íntegro en Grasshopper denominado como Galapagos; éste explora la gama de posibles distribuciones y entrega todas las soluciones que suman el menor número posible de intersecciones visuales; finalmente, el diseñador procede a elegir la distribución que estética y funcionalmente sea las más atractiva a su criterio.

Etapa 6: Pasillos

En este subalgoritmo se generan superficies que representarán las circulaciones y los pasillos del proyecto. El proceso inicia desplazando los bordes de cada módulo rectangular hacia su exterior una cantidad variable que se puede modificar libremente; posteriormente, se genera una superficie dentro del espacio neto. Es importante mencionar que se consideran las colindancias para no generar circulaciones ni pasillos fuera de los límites del terreno (ver figura 10, pág. ant.).

Etapa 7: Frente

Esta etapa define el frente para cada uno de los espacios rectangulares con el fin de permitir su correcto acceso desde los pasillos previamente generados. El subalgoritmo inicia recuperando las superficies de los pasillos previamente generados así como el nodo central de cada módulo, se procede con la obtención del punto más cercano de alguno de los bordes del pasillo al vértice central de cada bloque habitacional; es importante recalcar que se consideran nuevamente las colindancias para evitar la obtención de elementos que dan hacia algún lado fuera de los límites del predio. Una vez obtenido dicho nodo, se procede a unir éste con el punto central en cada módulo generando así una línea que servirá como referencia para siguientes etapas (ver figura 11, pág. ant.).

Etapa 8: Interior

Este proceso coloca referencias de división espacial por el interior de cada módulo, que funcionan también para la generación de muros, puertas y ventanas más adelante. Se da inicio con la generación de una geometría plantilla definida por el diseñador, la cual se almacena como parámetro, posteriormente, se procede a colocar y orientar dicho elemento geométrico en relación al eje rector generado en el bloque anterior, y se finaliza con la formación de subdivisiones espaciales interiores en cada bloque habitacional. Una vez orientada la geometría base en cada módulo habitacional se procede también a generar en paralelo los nodos de referencia necesarios para la colocación y orientación de las aberturas del proyecto, proceso que toma lugar más adelante en la etapa 13 (ver figura 12, pág. ant.).

Etapa 9: Niveles superiores

En este subalgoritmo se genera la distribución de módulos en los siguientes niveles en función de la primera planta.

El proceso sigue los mismos lineamientos y lógica de programación como lo fue para el desarrollo de la primera planta, sin embargo, se siembra un parámetro *SEED* distinto con el fin de generar un juego mayor de pisos. Posteriormente el subalgoritmo toma en consideración las superficies de la planta inferior y las combina con el resultado del piso superior, de tal forma que los espacios descubiertos del primer nivel que no coinciden con ningún bloque en el segundo, se convierten ahora en terrazas y áreas verdes abiertas que abastecerán el piso superior. Es importante recordar que la planta baja tiene acceso directo a las áreas verdes generadas por los primeros procesos del proyecto, y con el fin de mantener la misma oportunidad para todos los niveles del proyecto, se establecen dichas relaciones entre plantas que además de generar las azoteas necesarias en aspectos constructivos para los pisos de abajo, se aprovechan en paralelo como áreas abiertas bajo un esquema de doble funcionalidad.

Se procede ahora a parametrizar el algoritmo para generar el tercer nivel en relación a los anteriores; mientras que en el segundo piso el *SEED* manejado fue distinto, el tercer nivel mantiene el mismo parámetro que la primera planta. De la misma forma como sucedió en el segundo nivel (en donde se generan azoteas que tapan el piso anterior y al mismo tiempo funcionan como espacios abiertos), se establecen relaciones con el segundo nivel, generando de la misma forma azoteas que funcionan como espacios abiertos (ver figura 13, sig. pág.).

Etapa 10: Zonas

En esta etapa se recuperan los módulos de los tres niveles y se vincula la geometría con elementos BIM, esto para generar las zonas del proyecto. Se da inicio al proceso con la recuperación de la información necesaria y se procede a vincular la herramienta *Zona*, con tal de generar los primeros elementos dentro del entorno virtual de ArchiCAD con el manejo del plugin de la conexión en vivo entre las distintas plataformas.

Al ser el resultado componentes BIM, éstos almacenan todo tipo de información tanto visual (2D y 3D) como no gráfica (metadatos y propiedades), también brindan la posibilidad de más adelante, y de forma automatizada, generar un esquema arquitectónico con las categorías, nombres y áreas del proyecto en sus m², ligándose a un precio por área para la programación de un costo paramétrico, esto de forma instantánea. Cabe aclarar que también se vinculan espacios fijos que no forman parte del proceso y representan parte de las *áreas negativas* consideradas a principios del proyecto (ver figura 14, sig. pág.).

Etapa 11: Estructura

El este proceso, con base en la información y relación de espacios en los tres pisos, se genera una estructura básica compuesta de zapatas, columnas y vigas. Se debe aclarar que mientras el criterio estructural de un proyecto arquitectónico se puede desarrollar bajo varios esquemas y criterios conformados por elementos complejos, el presente proceso maneja consideraciones básicas; por un lado, con el fin de explorar y desarrollar algunas directrices iniciales dentro de procesos de estructuración con el manejo de las presentes herramientas, y por otro, para establecer principios estructurales base que justifiquen consideraciones ejecutivas del proyecto.

Se inicia recuperando los vértices de cada rectángulo habitacional en los tres niveles y se consideran como puntos válidos aquellos que coinciden en todos los pisos; posteriormente, se colocan en dichos nodos las columnas que abarcan la altura total del proyecto, y por debajo de cada elemento en paralelo se posiciona una zapata aislada con dimensiones variables definidas por el usuario. Se procede retomando ahora los contornos de todos los bloques habitacionales, se filtran las aristas duplicadas y sobre las restantes se coloca una viga con dimensiones y tipo definidos por el usuario (ver figura 15, sig. pág.).



Figura 13.

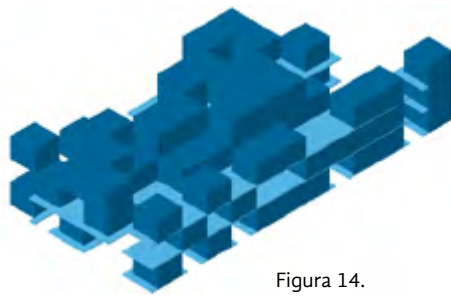


Figura 14.

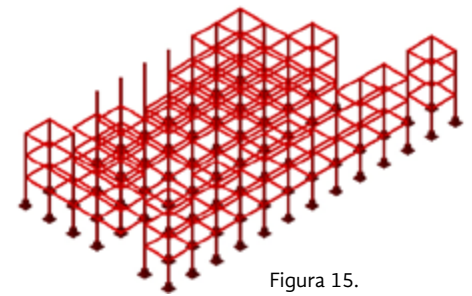


Figura 15.

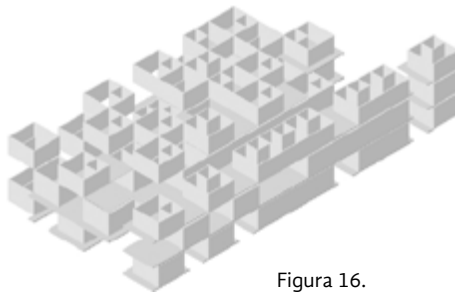
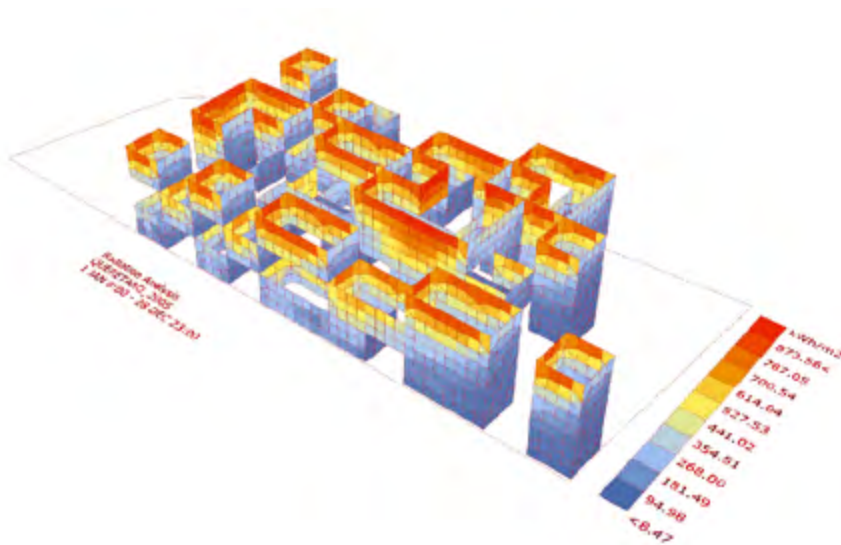


Figura 16.



Figura 17.



Etapa 12: Muros y losas

El presente subalgoritmo recupera la información geométrica previamente generada, y coloca losas y muros en todos los niveles. Se recuperan las superficies de los módulos habitacionales, así como las de los pasillos, y con dicha información se procede a vincular las losas de ArchiCAD. Se retoma la geometría generada previamente en la etapa 8, y con base en las aristas de cada módulo habitacional se vinculan todos los muros tanto exteriores como interiores, así como los pretilos del proyecto (ver figura 16, pág. ant.).

Es importante aclarar que los muros, además de formar parte de la envolvente arquitectónica, son necesarios para colocar puertas y ventanas en la siguiente etapa, ya que las aberturas en ArchiCAD son elementos BIM asociativos, en otras palabras, componentes que requieren de una pared para poder ser colocados.

Etapa 13: Puertas y ventanas

Con base en los muros previamente generados y recuperando nodos de referencia de la etapa 8, se vinculan las aberturas del proyecto. Para colocar una puerta o ventana, los principales parámetros requeridos (además del muro en el que se posicionará) son 2 puntos de referencia, uno define la ubicación en la que se sitúa la abertura y el otro define su orientación; además de dichos requisitos, se manejan variables adicionales que controlan el ancho, el largo, el tipo, entre otras características que definen a mayor grado los elementos generados (ver figura 17, pág. ant.).

Etapa 14: Asoleamiento

El presente proceso recupera la información volumétrica de los módulos habitacionales y con base en la información geográfica adicional, analiza el asoleamiento del proyecto. Dicha actividad inicia dentro de la interfaz de Grasshopper con el manejo de un *plugin* denominado LadyBug, que permite evaluar y analizar distintos aspectos relacionados a la sustentabilidad de un proyecto para la posterior toma de resoluciones con relación a la optimización energética de éste. Se recupera con Meteonorm® la información geográfica del terreno por medio de un archivo en formato .epw (archivo

de datos meteorológicos guardado en el formato estándar EnergyPlus que contiene datos meteorológicos que se utilizan para ejecutar simulaciones de uso de energía), el cual se vincula a LadyBug para proceder con el análisis de la radiación solar que se recibe anualmente en cada módulo habitacional, posteriormente, se genera una nueva volumetría teñida bajo una gama de colores en función de la cantidad de radiación solar que recibe cada cuerpo geométrico, en paralelo, LadyBug también genera una leyenda que permite comprender mejor el comportamiento solar del proyecto.

Finalmente, se recupera la cantidad de radiación solar total que recibe anualmente el proyecto, y se procede con la programación de un filtro que descarta toda parte geométrica que contiene menos de 250 kWh/m², manteniendo así toda área que recibe radiación solar superior al criterio previamente mencionado. Generada la geometría, se procede a extruir ésta con un espesor variable definido por el usuario para formar la representación de una *doble piel*, cuya función es absorber y disipar la radiación solar del proyecto; el proceso termina con la asignación de un color —esto con motivos estéticos— a cada parte que conforma el elemento previamente mencionado (ver figura 18, pág. ant.).

Etapa 15: Costo paramétrico

El algoritmo concluye con la programación y vinculación de un costo paramétrico, éste automatizado con las funciones nativas de ArchiCAD. Con base en el modelo de zonas previamente generado en la etapa 10, se programa un esquema que además de medir los m² que abarca cada espacio, también enlista las categorías y los nombres de cada zona representativa; se procede con la creación de un metadato definido como “Costo por m²” que permite asignar un precio de construcción a cada zona, se asignan dichos valores en función de consideraciones económicas generales para la construcción de un proyecto de vivienda y se finaliza el proceso con la obtención de un precio dinámico, en otras palabras, un costo paramétrico que se actualiza en función del parámetro *SEED* del proyecto.

Resultados

A continuación se presentan los resultados cuantificables de 20 distintas iteraciones generadas con la combinación del parámetro SEED a modo completamente aleatorio; es importante destacar que cada iteración del proyecto se genera en un tiempo promedio de aproximadamente 8 segundos si se trabaja sólo con la interfaz de Grasshopper, y aproximadamente 20 segundos si se desean vincular componentes BIM para permitir la generación de planos, modelo, análisis y cuantificaciones adicionales.

Al revisar los datos se presentan distintos comportamientos de interés; algunas iteraciones a resaltar son la 20 al ser la propuesta más económica, la 18 con el menor asoleamiento, la 20 con la menor cantidad de pasillos y terrazas, la 10 con la menor cantidad de módulos y la 19 con el mejor criterio visual al contener menos obstáculos.

Al tomar cada propuesta un tiempo aproximado de 20 segundos en generarse, considerando que se desea aprovechar el potencial completo con la inclusión de componentes BIM, se puede decir que es posible diseñar 20 distintas propuestas de anteproyectos arquitectónicos que resuelven el mismo problema espacial en aproximadamente 7 minutos; es resolviendo y desarrollando una única propuesta, por lo mismo, se puede decir que el contraste en términos de tiempo entre el método generativo y el ordinario, en 2D, es evidente.

Mientras que la principal intención del documento es resaltar la optimización de tiempo que oferta el método generativo contra el modelo de diseño ordinario, como lo es con la generación de bocetos y el manejo de herramientas CAD en 2D, o el modelado a modo manual con instrumentos BIM, hay que presentar asimismo una postura crítica en función de las ventajas y desventajas implícitas de esta forma de trabajo.

Así como el método generativo tiene el potencial de generar un número virtualmente infinito de iteraciones del mismo proyecto en cuestión de segundos, el diseño y la programación del sistema que genera

ITERACIÓN	1	2	3	4	5	6	7	8	9	10
SEED : PB + N1	110294	194942	256241	312356	413998	432031	495830	788109	591280	526362
SEED : N2	192042	525874	609395	232956	609395	645789	805928	751326	192049	938102
No. Módulos Totales	74	73	74	75	73	72	71	75	72	62
m² Totales Módulos	1184	1168	1184	1200	1168	1152	1136	1200	1152	992
m² Totales Pasillos	2065.71	2164.98	1683.48	1326.21	1367.78	1958.88	2069.51	1631.41	1538.07	1936.85
Intersecciones Visuales	124	129	130	130	163	114	107	100	130	93
No. Zapatas	67	66	67	66	62	64	63	73	60	60
Columnas (ML)	656.6	646.8	656.6	646.8	607.6	627.2	617.4	715.4	588	588
Vigas (ML)	1600	1536	1632	1632	1552	1456	1536	1600	1648	1408
Asoleamiento Total (kWh)	743592.6294	722926.944	760267.6222	743131.3539	723753.232	700026.9233	753020.4614	764552.5438	740351.512	704736.8533
Costo Paramétrico	\$26,602,260	\$27,005,880	\$24,308,880	\$22,357,260	\$22,222,680	\$25,577,280	\$26,049,060	\$24,188,460	\$23,052,420	\$23,525,100

Figura 19. Tabla con datos técnicos sobre iteraciones 1-10.
Software: Generado en Rhinoceros 6 + Grasshopper y tabulado en Google Sheets.
Fuente: elaboración propia.

ITERACIÓN	11	12	13	14	15	16	17	18	19	20
SEED : PB + N1	683192	726394	819285	928059	967829	168283	183948	286930	376920	486867
SEED : N2	495205	859102	375910	995748	586729	293840	674829	586938	498758	682945
No. Módulos Totales	64	70	73	70	71	73	74	70	66	72
m² Totales Módulos	1024	1120	1168	1120	1136	1168	1184	1120	1056	1152
m² Totales Pasillos	1976.12	2037.74	2009.59	1968.88	2050.16	1673.24	976.1	1173.81	1918.02	940.98
Intersecciones Visuales	117	124	152	146	132	148	127	133	83	148
No. Zapatas	59	57	59	59	66	65	67	57	52	59
Columnas (ML)	578.2	558.6	578.2	578.2	646.8	637	656.6	558.6	509.6	578.2
Vigas (ML)	1424	1568	1536	1440	1552	1552	1632	1536	1424	1616
Asoleamiento Total (kWh)	701747.1055	726082.8099	724805.9145	624288.3149	725079.9697	744068.4969	734069.2169	671917.6595	687890.4825	731685.8536
Costo Paramétrico	\$24,144,720	\$25,666,440	\$26,073,540	\$25,253,280	\$25,932,960.00	\$24,055,440	\$20,064,600	\$20,482,860	\$24,180,120	\$19,469,880

Figura 20. Tabla con datos técnicos sobre iteraciones 11-20.
Software: Generado en Rhinoceros 6 + Grasshopper y tabulado en Google Sheets.
Fuente: elaboración propia.

las iteraciones requiere bastante tiempo en desarrollar, y todavía más en probar y perfeccionar, como lo fue en el caso de *Callejuelas* requiriendo más de un mes la programación de todo el sistema; incluso se puede considerar que el diseño de una única propuesta del mismo proyecto bajo métodos de diseño ordinarios requiere menos tiempo que la programación de un sistema; y es por esto que hay que resaltar que el método generativo se debe contemplar en situaciones de trabajo que involucran patrones de diseño que comparten las mismas características, y no parámetros únicos a cada proyecto.

Si un despacho de arquitectura únicamente se dedica a diseñar variaciones de viviendas habitacionales compartidas en un problema de diseño similar, con participantes sociales, espacios, materiales, aberturas, criterios geográficos, sustentables, económicos entre más constricciones y limitantes que parten de un mismo problema, se puede decir que el método generativo tiene capacidad de explotar la generación de resultados para dicho grupo de trabajo, ya que sólo requiere ser programado en una única ocasión para cubrir las necesidades del despacho y generar proyectos nuevos para cada problema en tiempos récord. Sin embargo, si el mismo despacho se dedica a diseñar proyectos diversificados, desde espacios residenciales, comerciales, industriales, entre otros, con distintas características, se puede considerar que es más rápido y eficiente diseñar propuestas únicas de dichos proyectos bajo métodos ordinarios, mas también hay que mencionar que se pueden manejar programaciones generativas moduladas, como lo fue en el presente proyecto por medio de subalgoritmos, que resuelven problemas de diseño puntuales, con la posibilidad de combinar módulos presentes en programaciones más complejas.

Además, hay que contemplar que mientras la experimentación del método generativo se aplicó al caso de un proyecto a nivel anteproyecto, si se trabaja un sistema con capacidad de resolución de problemas ejecutivos como lo son las distintas ingenierías que

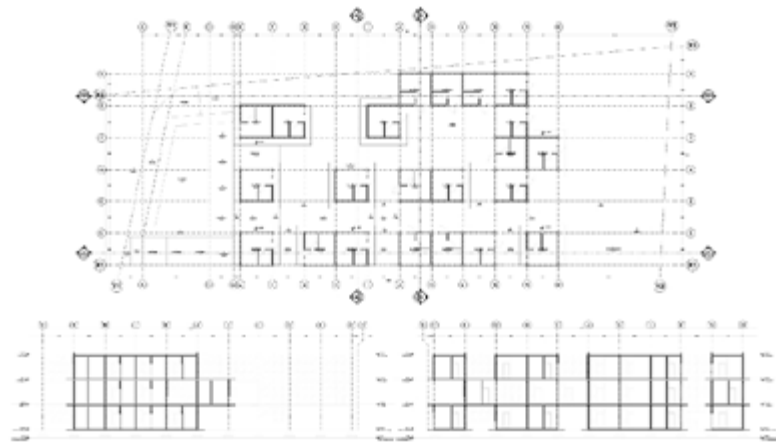


Figura 21. Parte de la documentación 2D del proyecto, generada en paralelo con cada nueva iteración con la integración BIM de ArchiCAD. *Software:* ArchiCAD 24.

Fuente: elaboración propia.

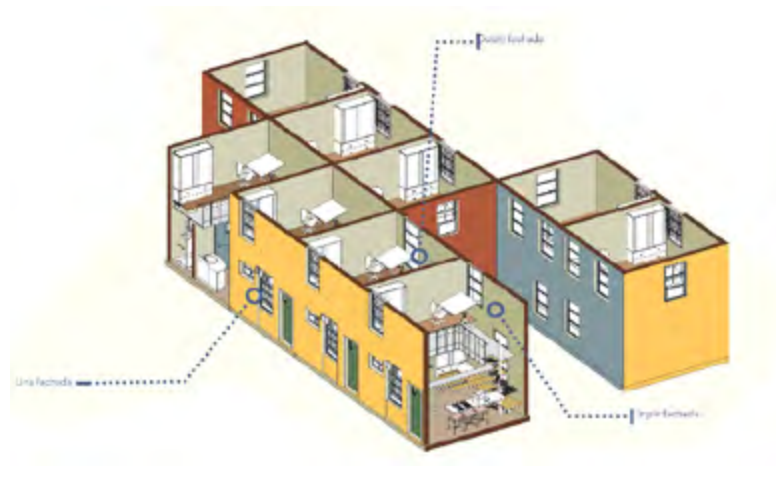


Figura 22. Isométrico generado ilustrando ejemplo de uso del proyecto. *Software:* ArchiCAD 24.

Fuente: elaboración propia.

intervienen en un proyecto arquitectónico, esto impacta significativamente la complejidad del proceso requiriendo aún más tiempo para la programación, la prueba y la perfección del sistema.

Es importante también destacar otra postura crítica del método generativo referente al riesgo de pérdida del factor creativo, ya que hay que tener presente que mientras la computadora y los algoritmos tienen una capacidad superior para la resolución de problemas funcionales y cuantificables, es el ser humano quien marca la diferencia e impacta en la toma de decisiones referentes a la estética de un proyecto, motivo por el cual, si los arquitectos o diseñadores delegan por completo el diseño de un proyecto al método generativo, existe también el riesgo de que la estética de los proyectos se perciba monótona y despersonalizada en todas las iteraciones.

Hay que tener presente que mientras el método generativo resuelve parte crucial del proyecto, es el arquitecto quien a final de cuentas debe responsabilizarse en la integración de criterios humanísticos, así como la selección de la iteración con las mejores oportunidades estéticas, y sin descartar lo que indican los datos cuantificables de la iteración, para su respectiva evolución.

*“El verdadero progreso
es el que pone la tecnología
al alcance de todos”*

—Henry Ford

Conclusión

Mientras que el algoritmo general del proyecto experimental concluye con el desarrollo de 15 etapas debido a limitantes de trabajo, hay que resaltar que es posible explotar aún más la capacidad generativa del algoritmo, permitiendo profundizar más en la automatización de los procesos relativos al diseño de un proyecto arquitectónico, como todo tipo de procesos que evolucionan un anteproyecto a uno ejecutivo. Desde una perspectiva, es posible programar la distribución del mobiliario, la generación de acabados, elementos de confinamiento, criterios de instalaciones, componentes estructurales complejos, distintas ingenierías, consideraciones complementarias de sustentabilidad entre todo tipo de características que toman vida en un proyecto ejecutivo.

También hay que recordar que mientras se maneja un componente generativo en una ocasión, como lo fue en la etapa 5, es posible optimizar el diseño al límite en todos sus aspectos funcionales siempre y cuando se fije un objetivo basado en resultados cuantificables.

Cabe destacar que el diseño paramétrico como generativo permiten programar un proceso de diseño con límites virtualmente infinitos, con la capacidad de generar un resultado de una vivienda a nivel anteproyecto o incluso ejecutivo en tiempos récord, herramientas que revolucionan y optimizan la forma de trabajar de distintos profesionistas; y comprendiendo dicha oportunidad que nos brindan los presentes instrumentos, es importante que los profesionales implementen dichas soluciones en sus empresas y grupos de trabajo con el fin de generar un cambio significativo en su sector laboral, ya que como Henry Ford (s.f.) mencionó una vez: “El verdadero progreso es el que pone la tecnología al alcance de todos”.

Referencias

- Al-Fedaghi, S. (2015). "On a Flow-Based Paradigm in Modeling and Programming", en *International Journal of Advanced Computer Science and Applications*, 209.
- Archistar. (s.f.). *Archistar*. Obtenido de Parametric Design vs. Generative Design – The Pros and Cons. Disponible en <<https://archistar.ai/blog/parametric-design-vs-generative-design-the-pros-and-cons/>>.
- Davis, D. (06 de agosto de 2013). *danieldavis*. Obtenido de A History of Parametric. Disponible en <<https://www.danieldavis.com/a-history-of-parametric/>>.
- Gómez-Moriana, R. (16 de Agosto de 2012). *Criticalista*. Obtenido de Gaudí's hanging chain models: parametric design avant la lettre? Disponible en <<https://criticalista.com/2012/08/16/gaudis-hanging-chain-models-parametric-design-avant-la-lettre/>>.
- Graphisoft. (2021). *Graphisoft*. Obtenido de Nuestra Historia. Disponible en <<https://graphisoft.com/es/why-graphisoft/our-story>>.
- Gutiérrez, A. F. (Junio de 2016). *Fenómeno arquitectónico, proceso de diseño y complejidad humana : propuesta de re-conceptualización*. México: Universidad Nacional Autónoma de México. Disponible en <https://repositorio.unam.mx/contenidos?c=plb1oP&v=0&d=false&t=search_0&i=1&as=0&q=humanidades>.
- Holland, N. (14 de June de 2011). "Theses from the Architecture Program", en *Inform Form Perform*, 18. Lincoln, Nebraska, Estados Unidos: University of Nebraska-Lincoln.
- Imperiale, A. (2018). *Academia*. Obtenido de An 'Other' Aesthetic: Moretti's Parametric Architecture. Disponible en <https://www.academia.edu/37892334/An_Other_Aesthetic_Morettis_Parametric_Architecture>.
- Jiménez, Á. M. y G. A. Correa (2015). "Tendencias de la arquitectura en la era de la información y la innovación", en *Revista Académica e Institucional de la UPC*(97), 19-40.
- Kolarevic, B. (2003). "Digital morphogenesis", en B. Kolarevic, *Architecture in the Digital Age: Design and Manufacturing* (pág. 25). Nueva York / Londres: Spon Press.
- Kowalski, J. (21 de Diciembre de 2016). *¿Sabes lo que es el Diseño Generativo?* Autodesk Latam. Disponible en <<https://www.youtube.com/watch?v=DkeoeNFewh8>>.
- McSweeney, E. (28 de Enero de 2020). *Financial Times*. Obtenido de Automation finds home in building design. Disponible en <<https://www.ft.com/content/e36ba45e-f973-11e9-a354-36acbbb0d9b6>>.
- Michael J. Dawes, M. J. (2017). "Christopher Alexander's A Pattern Language: analysing, mapping and classifying the critical response", en *City, Territory and Architecture*, 2. Disponible en <doi:<https://doi.org/10.1186/s40410-017-0073-1>>.
- MODELAB. (2018). *The Grasshopper Primer (EN)* (Tercera ed.). MODELAB. Disponible en <<https://www.modelab.is/grasshopper-primer/>>.
- Montaner, J. M. y Z. Muxí (2011). *Arquitectura y política. Ensayos para mundos alternativos*. Gustavo Gili.
- Robert McNeel & Associates. (4 de Diciembre de 2020). *McNeel Wiki*. Obtenido de The History of Rhino. Disponible en <<https://wiki.mcneel.com/rhino/rhinohistory>>.
- Robert McNeel & Associates. (2021). *Rhinoceros*. Obtenido de Overview. Disponible en <<https://www.rhino3d.com/features/#overview>>.
- Rutten, D. (25 de Julio de 2013). *Grasshopper 3D*. Obtenido de GH's Origin? Disponible en <<https://www.grasshopper3d.com/m/discussion?i-d=2985220&Topic:890961>>.
- Schumacher, P. (2008). *Parametricism as Style - Parametricist Manifesto*. Londres. Disponible en <<http://www.patrikschumacher.com/Texts/Parametricism%20as%20Style.htm>>.
- Stocking, A. W. (14 de Octubre de 2009). *Cadallyst*. Obtenido de Generative Design Is Changing the Face of Architecture. Disponible en <<https://www.cadallyst.com/cad/building-design/generative-design-is-changing-face-architecture-12948>>.
- Terzidis, K. (2008). "AutoPLAN: a stochastic generator of architectural plans from a building program", en *Z Joint Study Journal*, 84.
- Weisstein, E. W. (2002). "Parametric Equations", en E. W. Weisstein, *CRC Concise Encyclopedia of Mathematics*. Boca Raton / Londres / Nueva York / Washington: Chapman and Hall / CRC, p. 2150.
- Zexin, S. y H. Mei, (2017). "Robotic form-finding and construction based on the architectural projection logic", en *IOP Conference Series Materials Science and Engineering* 216(1):012058.