



UNIVERSIDAD DE LOS ANDES
FACULTAD DE INGENIERÍA Y
CIENCIAS APLICADAS

SimVex. Un Software de Modelación de
Vías Exclusivas de Transporte Publico

SEBASTIAN GONZALEZ RODRIGUEZ-PEÑA

TRABAJO DE TITULACIÓN DE TIPO PRODUCTO PARA
OPTAR AL TÍTULO DE
INGENIERO CIVIL EN CIENCIAS DE LA COMPUTACION

PROFESOR GUÍA: RODRIGO FERNANDEZ

SANTIAGO, AGOSTO 2023

UNIVERSIDAD DE LOS ANDES
FACULTAD DE INGENIERÍA Y
CIENCIAS APLICADAS

ÍNDICE GENERAL

RESUMEN	8
1. INTRODUCCIÓN	9
2. REVISION BIBLIOGRAFICA.....	10
2.1 PASSION (PARALLEL STOP SIMULATION)	10
2.1.1 Modulo de Pasajeros	10
2.1.2 Modulo de Buses.....	11
2.1.3 Modulo de Salida	11
2.1.4 Modulo de Interacción	12
2.1.5 Modulo de Resultados.....	12
2.2 MODELACION MICROSCOPIVA ENTRE PARADEROS DE BUSES.....	14
2.2.1 Modelacion	14
2.2.2 Aplicacion Experimentales y Resultados.....	16
2.3 AVANCES PREVIOS	17
3. SIMVEX (SIMULACIÓN VÍAS EXCLUSIVAS).....	19
3.1 MOTIVACION	19
3.2 FUNCIONAMIENTO	19
3.2.1 Crear Nueva Simulación	23
3.2.2 Regenerar Simulación	29
3.2.3 Reutilizar Simulación.....	30
3.3 RESULTADOS	31
3.3.1 Datos de entrada.....	31
3.3.2 Datos Buses Simulados.....	31
3.3.3 Datos Paradas Simuladas	32
3.3.4 Datos Generales Simulación	32
3.4 ARQUITECTURA	32
3.4.1 Modulo de Entrada.....	33
3.4.2 Modulo de Archivos	33
3.4.3 Controlador de Simulación	34
3.4.4 Modulo Estadístico	34
3.5 MODELOS DE DATOS	34
3.5.1 Nodos	34
3.5.2 Entidades.....	37
3.5.3 Rutas	40
3.5.4 Controlador de simulación.....	40
3.6 SIMULACION	41
3.6.1 Inicialización.....	41
3.6.2 Ejecución.....	43
4. CONCLUSIONES Y RECOMENDACIONES	46
4.1 RESULTADOS	46

4.2 VALIDACION DE MODELO	46
4.3 MEJOREAS EN LA IMPLEMENTACION	47
4.3.1 Control de errores	47
4.3.2 Despejar Modelos	48
4.3.3 Fortaleces la jerarquía de nodos.....	48
4.3.4 Extender modelo de buses	49
5. BIBLIOGRAFIA	50
6. ANEXOS.....	51

RESUMEN

Este proyecto se centra en el desarrollo de una aplicación de simulación para rutas exclusivas de buses en el transporte público. La principal característica distintiva de esta simulación radica en su capacidad para definir un número ilimitado de nodos (calles, paraderos, intersecciones) y rutas, lo que proporciona una flexibilidad sin precedentes al usuario para crear simulaciones extensas y representativas de la realidad. Además, se permite ajustar los parámetros de distribuciones relacionadas con la llegada de pasajeros y buses, tiempos de subida y bajada de pasajeros, y la ocupación inicial de pasajeros en un bus.

La arquitectura de la simulación y el comportamiento de las entidades involucradas se basan en la experiencia previa obtenida de softwares de modelamiento anteriores, en particular, se ha tomado como referencia el programa PASSION. Este enfoque ha permitido desarrollar una aplicación eficiente y efectiva, aprovechando las lecciones aprendidas y las mejores prácticas de herramientas existentes.

El objetivo principal del proyecto se ha cumplido exitosamente al crear una herramienta de simulación versátil y configurable que permite estudiar una amplia gama de características relacionadas con las vías exclusivas para buses. A pesar de lograr una interfaz sencilla y amigable para el usuario, se reconocen áreas de mejora que podrían abordarse en futuros proyectos. El producto resultante se presenta como una solución abierta y flexible, lista para ser mejorada y adaptada a futuras necesidades.

Este informe detalla el proceso de desarrollo, la formulación de la arquitectura de la simulación, los métodos empleados para la implementación, y los resultados obtenidos. Se concluye que la aplicación desarrollada representa una herramienta que podría resultar útil para el análisis y planificación del transporte público en vías exclusivas para buses.

1. INTRODUCCIÓN

Las simulaciones de computador son una herramienta útil en diversos campos del conocimiento, gracias a su capacidad para representar y explorar fenómenos complejos de la realidad. Estas simulaciones se fundamentan en modelos matemáticos que, si bien simplifican algunos comportamientos observables, capturan esencialmente la esencia de los sistemas en estudio. A través de la creación de modelos matemáticos adecuados, las simulaciones permiten a los investigadores analizar y prever el comportamiento de sistemas complejos, proporcionando una ventana única para estudiar procesos difíciles de observar directamente en la realidad.

La aplicación de simulaciones en el contexto de diseño de vías exclusivas para buses representa una herramienta valiosa para asistir en el trabajo de optimización y mejoramiento de la eficiencia del sistema de transporte público. Al utilizar modelos matemáticos que simplifican el comportamiento del tráfico, estas simulaciones permiten analizar diversas configuraciones de vías, la ubicación de paradas, la frecuencia de servicio y otros factores clave. De esta manera, los planificadores y diseñadores pueden evaluar diferentes escenarios, anticipar posibles problemas y tomar decisiones informadas para maximizar la capacidad de transporte, reducir tiempos de viaje, mitigar congestiones y, en última instancia, mejorar la experiencia de los usuarios de transporte público.

El objetivo de esta memoria es crear SimVEX (Simulación Vías EXclusivas), un software de simulación de vías exclusivas para buses del transporte público. SimVEX utiliza como base los principios y teorías utilizadas en modelos previos, específicamente PASSION, y se plantea como objetivo extender las capacidades de simulación ofreciéndole al usuario un alto nivel de personalización de la simulación.

2. REVISION BIBLIOGRAFICA

Las simulaciones de computador son modelos matemáticos que pretenden representar la realidad y cómo se comportan sus diferentes participantes, y se puede proponer que existe un balance entre la complejidad del modelo matemático, la representatividad de ese modelo con la realidad y los recursos computacionales necesarios para ejecutar dicho modelo, es decir, mientras más complejo es el modelo, se pueden esperar resultados más fidedignos a los comportamientos observados en la realidad, pero, a un costo computacional muchísimo mayor.

En este capítulo se hacen revisiones de diferentes softwares enfocados al modelamiento del tráfico en vías públicas, específicamente, el modelamiento del transporte público, con diferentes grados de complejidad, pudiendo esta complejidad catalogar el tipo de modelamiento como modelos macroscópicos, modelos híbridos y modelos microscópicos, dando especial énfasis en estos últimos.

2.1 PASSION (Parallel Stop SimulatiON)

El software de simulación denominado como PASSION (PARallel Stop SimulatiON) (Fernández, 1998) es parte de un estudio más amplio con la finalidad de modelar microscópicamente la interacción entre buses, pasajeros y tráfico en un paradero. Fue desarrollado inicialmente para superar las limitaciones presentes en el modelo de la época IRENE (Gibson 1989), donde no considerar el impacto de las operaciones sobre los pasajeros, tales como los tiempos de espera y la densidad de las plataformas.

Desarrollado inicialmente en C++ y disponiendo de una interfaz DOS con capacidad de simular una única parada, vive posteriormente una actualización para poder ser ejecutado en Windows y tener la posibilidad de simular hasta dos paraderos simultáneos sucesivos en una avenida.

La simulación está compuesta por cinco componentes o módulos, siendo estos: módulo de pasajeros, módulo de buses, módulo de salida, módulo de interacción y módulo de resultados, donde los primeros tres hacen los cálculos correspondientes a sus propios comportamientos, entregan sus valores al módulo de interacción y los resultados se presentan en el módulo de resultados, donde se entregan los valores de interés de la simulación.

2.1.1 Modulo de Pasajeros

Acá se generan los valores de llegada y características generales de los pasajeros pertenecientes a la simulación.

Los valores de llegada de los pasajeros al paradero pueden ser predefinidas, estar determinadas por el valor de alguna variable aleatoria correspondiente a alguna función de distribución de probabilidad conocida como Uniforme, Exponencial u otra.

Otra característica es la ruta en la que viaja el pasajero dentro de la simulación, PASSION es capaz de modelar múltiples rutas de viajes dentro de la misma simulación.

La última característica es el valor medio que demoran los pasajeros en subir al bus perteneciente a su ruta, estos valores pueden estar sujetos a condiciones como el método de pago del servicio (pase o efectivo), la edad del pasajero (mientras más edad, más se demora en abordar) o específico al tipo de bus que se aborda.

Los valores previos a la simulación se preparan y presentan en una hoja de cálculo.

2.1.2 Modulo de Buses

El módulo de buses genera los valores de llegada y características generales de los buses dentro de la simulación, los patrones de llegada del bus al primer paradero son análogos a los mencionados en el módulo de pasajeros, además de contar con opciones de llegada como llegadas regulares, llegada por horario y llegadas por lote (varios buses a la vez).

Otras características importantes pertenecientes a este módulo son la ruta a la que corresponde cada uno de los buses, el número de pasajeros que desciende en cada uno de los paraderos, el tiempo medio de desembarco de los pasajeros en cada bus, y el tiempo de bloqueo que sufre cada bus al momento de partir desde el paradero.

Todos estos valores pueden ser asignados a todos los buses de manera global, o se pueden especificar diferentes datos de manera individual. Todas estas características se presentan previo a la simulación en una hoja de cálculo.

2.1.3 Modulo de Salida

Este módulo controla de qué manera el bus sale del paradero. Se proporcionan cuatro opciones en esta etapa.

Salida Libre: Una vez realizado el intercambio de pasajeros en el paradero, el bus puede seguir su ruta sin espera adicional.

Semáforo: Un semáforo controla los tiempos de salida del bus desde el paradero una vez terminado el intercambio de pasajeros. Los valores que se deben entregar en esta opción son la duración del

ciclo completo del semáforo, y el porcentaje del tiempo de ese ciclo que el semáforo se encuentra en verde.

Brecha en pistas adyacentes: Esta opción ocurre cuando el bus se debe integrar a una pista adyacente al paradero, y debe esperar un espacio suficiente para integrarse a la circulación. El valor que se debe especificar en esta opción es el flujo de vehículos en la pista, a la que el bus desea integrarse a partir del cual se calcula la brecha para integrarse.

Bloqueo: En esta opción la partida del bus está siendo atrasada por un vehículo que bloquea la salida delante del bus. Por ejemplo: un vehículo estacionado o vehículo en pane.

Los valores correspondientes a cada una de las opciones del módulo de salida son proporcionados por el usuario durante la definición de este módulo.

2.1.4 Modulo de Interacción

Este módulo se encarga de recibir toda la información ingresada por el usuario en los tres módulos anteriores, calcula los flujos de los buses y las demandas de los pasajeros para las diferentes rutas en cada uno de los paraderos. Luego se estiman los efectos sobre los buses y pasajeros y, finalmente, se calculan los valores relevantes para cada uno de los paraderos de la simulación.

Se puede decir que este módulo es el centro de la simulación, siendo su tarea general recibir los datos iniciales y entregar valores resultantes después de haber pasado por el modelo de la simulación.

2.1.5 Modulo de Resultados

Con los datos obtenidos en el módulo de interacción se procede a calcular los valores de interés de la simulación, tales como: la capacidad o saturación del paradero, tamaño de las colas y demoras de buses resultantes en un paradero además del tiempo de espera promedio de los pasajeros y el número de pasajeros esperando en parada.

El modelo matemático utilizado en PASSION es el propuesto en la investigación de (Gibson, 1989), donde se calcula la capacidad de buses por hora del paradero, para un paradero con un único sitio de detención, o aplicando a un paradero con múltiples sitios, pero aplicando la ecuación a la última posición de la parada (Fernández 2007).

$$Q_b = \frac{3600N_b}{\sum_{i=1}^{N_b}(t_c + PST_i + t_{ei})} \quad (1)$$

Donde:

Q_b : Capacidad de buses por hora del paradero [bus/h]

N_b : Numero de buses simulados para el paradero (Modulo de Buses)

t_c : Tiempo medio de salida entre buses [s] (Modulo de Buses)

t_{ei} : Tiempo de demora por condiciones de salida del andén [s] (Modulo de Salida)

PST_i : Tiempo de servicio del bus i dedicado al intercambio de pasajeros en el paradero [s]

Donde PST_i puede tomar dos valores dependiendo de la configuración del bus.

$$PST_i = \begin{cases} \beta_0 + \max \left\{ \sum_{j=1}^{p_{bi}} \beta_{bj} ; \beta_{ai} * p_{ai} \right\}, & \text{Operacion paralela} \\ \beta_0 + \left\{ \sum_{j=1}^{p_{bi}} \beta_{bj} + \beta_{ai} p_{ai} \right\}, & \text{Operacion secuencial} \end{cases} \quad (2)$$

Operación paralela significa que el bus cuenta con más de una puerta, donde el ingreso de los pasajeros se hace por una puerta y el descenso por otra de manera simultánea, mientras que en operación secuencial el bus cuenta con un único punto de acceso, y se debe esperar que todos los pasajeros que descienden en el andén lo hagan antes que los nuevos aborden el bus.

A partir de la versión 5.0 (Fernández & Moreno 2005) de PASSION se comienza a utilizar una nueva ecuación para calcular PST_i según la especificación en (Gibson 1997), resultando en la siguiente ecuación.

$$PST_i = \beta_0 + \beta_1 \delta_1 + \max_j \left\{ \sum_{k \in PB_{ij}} [\beta_2^k + \beta_3^k \delta_1 + \beta_4^k \delta_2(i)] + \sum_{k \in PA_{ij}} [\beta_5^k \exp(-\beta_6^k N(PA_{ij})) + \beta_7^k \delta_3(i)] \right\} \quad (3)$$

Donde:

PB_{ij} y PA_{ij} : son los grupos de pasajero subiendo y bajando del bus i por la puerta j respectivamente.

δ_1 : Variable de activación que vale 1 si el paradero esta concurrido, y 0 en caso contrario.

δ_2 : Variable de activación que vale 1 si hay más de 4 pasajeros en el bus, y 0 en caso contrario.

δ_3 : variable de activación que vale 1 si el bus esta completamente lleno, y 0 en caso contrario.

β : es un vector de parámetros de calibración, con valores mayores o iguales a 0.

Mientras que β_0 y β_1 corresponden a valores determinados por el paradero, el resto del vector se rige por características de los pasajeros

2.2 MODELACION MICROSCOPIVA ENTRE PARADEROS DE BUSES

A partir de lo anterior, Fernández & Peñalillo (2000) modelaron el comportamiento de un carril de circulación exclusivo para buses, donde se encuentran 2 paraderos separados por una intersección. El modelo está compuesto por tres partes, la primera es el programa de PASSION, explicado en el punto 2.1 de este capítulo, que se encarga de simular microscópicamente la interacción entre el bus y sus pasajeros cuando este se detiene en un paradero. Luego está el modelo que representa la circulación de buses por la vía exclusiva, esta circulación se modela como un movimiento uniformemente acelerado. Por último, está el modelo que regula el comportamiento del semáforo en la intersección que separa ambos paraderos dentro de la vía exclusiva.

2.2.1 Modelación

Como se mencionó, el modelo está compuesto por dos paraderos, P1 (el paradero que antecede la intersección), y P2 (el paradero que sucede a esta), la distancia desde P1 hasta la intersección i es denominada $d1$, la distancia entre i y P2 es denominada $d2$ y la suma de $d1$ y $d2$ se denomina d , que equivale a la distancia total del corredor simulada por el programa.

En el corredor un bus se puede encontrar en 5 estados diferentes, que simulan su comportamiento, estos son:

Detenido en un paradero: puede ser intercambiando pasajeros o esperando en cola para aproximarse a un paradero.

- Detenido en intersección: Ya sea por luz roja, o porque está detrás de otro bus.

- Acelerando: Acelerando desde el reposo después de salir de algún estado de detención. (Estado A con tasa de aceleración constante a_1)
- Viajando a velocidad constante: Luego de acelerar el bus alcanza su velocidad programada de viaje. (Estado VC a velocidad constante v_1)

Frenando hasta el reposo: Desacelerando antes de aproximarse a un paradero, intersección en rojo, o un bus detenido más adelante. (Estado F con tasa de desaceleración constante a_1)

La aceleración y desaceleración del bus a un valor constante dentro del modelo de circulación otorgan una buena representación de la circulación de los buses dentro de una vía exclusiva (Peñalillo 1999).

2.2.1.a Modelación Paraderos

PASSION se encarga de la modelación de los paraderos, donde ocurre la interacción entre un bus y sus pasajeros. Todos los detalles respecto al modelo PASSION ya están presentado en el punto 2.1 de este capítulo, y las ecuaciones que dominan el modelo son las ecuaciones (1) y (2).

2.2.1.b Modelación Circulación de los Buses

El bus $h=1$ ingresa a la simulación ubicándose previo a la parada P1. Al aproximarse al paradero se evalúan las condiciones referentes a la disponibilidad del paradero, si el paradero se encuentra ocupado por otro bus, el bus $h=1$ deberá detenerse detrás de este, sin poder entrar al área de detención del paradero, no puede realizar el intercambio de pasajeros y deberá esperar que el bus de adelante desocupe el paradero para proceder con la transferencia de pasajeros. Por lo contrario, si al llegar al paradero este se encuentra desocupado, se posiciona en la parada para realizar sus operaciones normales.

Una vez finalizado el intercambio de pasajeros del cual se encarga PASSION, el bus abandona P1 y avanza por la vía exclusiva hasta llegar a la intersección i , una vez más se revisa la condición actual del elemento simulado, en este caso el semáforo que controla la intersección. Si el semáforo se encuentra en rojo el bus $h=1$ se detiene en la última posición de cualquier fila formada por otros buses también detenidos en la intersección. Si el ciclo del semáforo de la intersección i se encuentra

en verde, el bus continua con su circulación de manera normal, recorriendo la distancia necesaria hasta llegar al paradero P2, donde el comportamiento y las condiciones para los buses son idénticas a las del paradero P1. El bus puede adelantar y ser adelantado durante la circulación entre los paraderos P1 y P2.

2.2.1.c Modelación Intersección

La modelación de la intersección es la más sencilla entre los tres componentes del programa, donde se definen dos parámetros, el primero es la duración del ciclo del semáforo en segundos, y el segundo es el tiempo efectivo del ciclo en el que el semáforo está en verde como una proporción del ciclo. A esta se le denomina razón de verde efectivo (μ)

2.2.2 Aplicación Experimentales y Resultados

La finalidad de desarrollar un programa de modelación es tener la capacidad para realizar experimentos y evaluar diferentes escenarios de manera sencilla, esto lleva a un mejor entendimiento de los elementos que tienen impacto en las operaciones de buses dentro del sistema de transporte público.

Se entiende que la capacidad de experimentación es muy amplia y se puede generar infinitos escenarios de estudio. En la investigación de Fernandez & Peñalillo (2000) se presentan 3 aplicaciones experimentales donde se mantiene constante los parámetros de tiempo medio de bajada del bus (1.5 s/pax), un tiempo medio de subida al bus (2.5 s/pax). Para cada bus se asume operación simultanea de intercambio de pasajeros, donde los que abordan entran por una puerta y los que descienden salen por otra (ecuación (2)). La programación del semáforo de la intersección i tiene un ciclo de 90 segundos y un verde efectivo de 71 segundos ($\mu = 0.79$). Estos parámetros se mantuvieron en los experimentos a menos que se indique lo contrario.

2.2.2.a Efecto de la ubicación relativa de paraderos con distinta demanda

En este experimento se buscó percibir los efectos un patrón de llegada aleatorio a los paraderos estudiados, y como la demanda de pasajeros del primer paradero afecta al que le procede. Los resultados determinaron que, para favorecer la circulación de los buses, el paradero con menor demanda se posicione antes del otro paradero que le sigue, ya que esto incrementa la velocidad comercial (la velocidad promedio calculado como la distancia del corredor dividido en el tiempo

en el corredor, incluyendo tiempo detenido) de los buses, principalmente por el decrecimiento de la demora media total en la cola de buses en el paradero.

2.2.2.b Efecto de la programación de semáforos

Para este experimento se estudió la programación del semáforo en la eficiencia de la vía exclusivo. Se detectó el impacto de la programación del semáforo en los valores de la desviación estándar de los intervalos de llegadas de los buses hacia el paradero, con valores más altos para los escenarios que cuentan con un valor de verde efectivo de 40%, elevando la probabilidad de que un bus se encuentre con un paradero ocupado y deba esperar en la cola. Sin embargo, a pesar de estas diferencias, que se pudieron apreciar en la media de la longitud de la cola de los buses, los valores máximos no se ven mayormente afectados por la variación de la programación del semáforo en la intersección.

Se concluyó también que un ciclo corto con mayor tiempo de verde favoreciendo la circulación de los buses, aumenta la velocidad comercial por la vía exclusiva.

2.2.2.c Efecto del tiempo de subida de pasajeros

El último experimento se observó los efectos globales sobre el corredor como consecuencia de la variación en el tiempo de subida de los pasajeros al bus.

Se hipotetiza que el aumento en el tiempo de subida de un pasajero al bus implica un crecimiento en la demora media para realizar la transferencia de pasajeros, afectado al mismo tiempo la duración de la ocupación de un bus en el paradero. La acción de aumentar el tiempo de abordaje de los pasajeros tiene el efecto cascado de impactar negativamente el resto de la operación, no solamente del bus en cuestión, sino que, de todo el sistema, teniendo como consecuencia una brusca caída en la velocidad comercial, y un aumento en la longitud media de cola de buses en ambos paraderos

2.3 AVANCES PREVIOS

Los avances previos en el contexto investigativo fueron significativos para ampliar las capacidades de los programas de simulación utilizados, desde la base de PASSION XLS (Aguirre 2015) se

realizan dos extensiones de este programa para ampliar las capacidades y trabajar en sus limitaciones. PASSION XLS 2.0 (Montero 2020) permitió la ejecución simultánea de múltiples instancias, lo que brindó la posibilidad de estudiar el sistema desde diferentes perspectivas y variar parámetros clave entre cada instancia. Esto abrió nuevas oportunidades para analizar cómo el sistema respondía a cambios en el flujo de buses por hora, la demanda de pasajeros por hora o ambos factores combinados.

Por otro lado, se desarrolló SESSION (Benavente 2021) para abordar una limitación importante: la simulación de múltiples paraderos en un mismo eje vial. Esto fue posible gracias a la base establecida por PASSION XLS 2.0, pero con la adición de un nuevo paradero cuesta arriba separado por una intersección semaforizada. Aunque SESSION presentaba la restricción de que la simulación solo podía correr con una única ruta de transporte público y los parámetros de entrada para ambos paraderos simulados eran los mismos, este avance permitió explorar escenarios más complejos y realistas.

En cuanto al mejoramiento de PASSION XLS, se trabajó en superar las limitaciones anteriores para lograr una simulación más precisa y detallada. Se buscó la independencia entre los paraderos simulados, permitiendo la simulación de más de una ruta de buses a la vez. Para lograr esto, se recurrió a mediciones muestrales del comportamiento poblacional y las operaciones de los buses en el terreno, lo que permitió estimar distribuciones que reflejaban el comportamiento real. Esta mejora llevó a una simulación más precisa y a la posibilidad de estudiar cómo diferentes variables, como la demanda de pasajeros por ruta y la distancia entre paraderos, influían en la operación del sistema.

La validación del modelo fue un paso fundamental en este proceso. Se recopilaron datos actualizados en terreno, enfocándose en los paraderos con mayor flujo dentro de la comuna. Estos datos se utilizaron para medir el número de buses y pasajeros, así como para analizar las operaciones en horas punta. Las simulaciones realizadas, considerando parámetros específicos, permitieron evaluar el impacto de la configuración del semáforo en la velocidad comercial de los buses en la vía. Además, se llevó a cabo una validación consistente del modelo, asegurando que este operara de manera predecible y que los resultados fueran coherentes con las expectativas.

3. SIMVEX (SIMULACIÓN VÍAS EXCLUSIVAS)

SimVex es un software de simulación de vías exclusivas para el transporte público que se inspiró fuertemente en los fundamentos utilizados en PASSION. Fue una herramienta desarrollada teniendo en cuenta la investigación y experimentación, poniendo especial énfasis en la generación y obtención de resultados relevantes para un contexto investigativo, además de proporcionar una plataforma eficiente para el diseño y experimentación.

La simulación generada por este programa se caracterizó por ser de carácter microscópico, lo que significa que se midió la actividad de cada entidad participante en la simulación de manera individual, sin acciones en masa. Esto permitió un análisis detallado y preciso del comportamiento de las entidades dentro del sistema de transporte público.

3.1 MOTIVACION

Como se presentó en el capítulo 2, herramientas previamente utilizadas para fines experimentales referentes al diseño de vías exclusivas de transporte público como PASSION XLS y sus derivados tenían limitaciones respecto a lo que podían modelar y simular, específicamente la extensión de la simulación donde el número de paraderos, intersecciones y rutas simuladas era predeterminada e inmutable, lo que restringe el rango experimental en gran medida. Con SimVex se buscó extender las capacidades de la simulación, dándole el control al usuario para asignar todos los parámetros que definen el comportamiento de las entidades de la simulación (pasajeros y buses), y los elementos físicos (nodos), distinguiéndose en la capacidad de simular múltiples paraderos y rutas, cada una con propiedades y condiciones únicas.

3.2 FUNCIONAMIENTO

Para cada una de las ejecuciones de SimVex se tiene como base 3 archivos que son necesarios para la correcta ejecución del programa, el primero es el archivo es el generador, todo la construcción de la aplicación se crea en base a este archivo de formato .json, que contiene toda los parámetros necesarios para crear y ejecutar la simulación, específicamente, contiene información de los nodos (calles, paraderos, intersecciones), y las distribuciones que dominan el comportamiento y aparición de las entidades (buses y pasajeros) dentro de la simulación.

A partir del archivo generador se construyen dos archivos nuevos, el primero es el “bus_dispatcher” o despachador de buses que almacena la data en formato .csv y contiene la

mínima información necesaria para crear un objeto bus el cual será simulado dentro del programa. Por cada línea del archivo se tiene información del tiempo de entrada del bus a la simulación (t_{in}), identificador único del bus (ID), la ruta a la cual pertenece el bus (route) y el número de puertas con las que cuenta el bus (door_n).

De forma equivalente existe el archivo “passenger_dispatcher”, que cumple el mismo rol, pero para los objetos de los pasajeros. La información contenida en este archivo son el tiempo de llegada de un pasajero a la simulación (t_{in}), el origen de este pasajero en la simulación, que puede ser un paradero o a bordo de un bus (origin), el destino del pasajero dentro de la simulación, que puede ser alguno de los paraderos simulados o simplemente se mantiene a bordo de un bus hasta salir de la simulación, es decir no desciende (destiny), la ruta que utilizara el pasajero para viajar dentro de la simulación(route) y tanto el tiempo que demora dicho pasajero en abordar y descender de un bus cuando este participa en el proceso de intercambio de pasajeros. Dentro del archivo de pasajeros se pueden encontrar 3 tipos de pasajeros:

- Pasajeros iniciales: estos existen dentro de la simulación desde el tiempo $t=0$, y se encuentran en su punto de origen (paradero) y estos son generados simulando virtualmente una ventana de tiempo desde la última vez que un bus que operaba su ruta paso por ese paradero. Esto ayuda a poblar la simulación desde el punto de observación y entrega homogeneidad a los datos obtenidos en la simulación respecto a lo que paso “antes” de comenzar la observación.
- Pasajero abordó: esto pasajero son los que entran a la simulación a bordo de un bus, tienen la misma finalidad que los del punto anterior, y es representar la existencia previa al inicio de la simulación. si bien el punto de origen de estos pasajeros siempre es un bus correspondiente a su misma ruta, el destino puede ser alguno de los paraderos que esta ruta opera o el final de la simulación, implicando que este pasajero nunca formo parte del intercambio de pasajeros entre un bus y un paradero, cualquiera de los destinos de este pasajero se determinada al azar y es equiprobable entre las diferentes opciones que tenga.
- Pasajeros regulares: estos son los que aparecen en el transcurso de la simulación, tiene su punto de origen en algún paradero correspondiente a su ruta, en un tiempo determinado.

Es critico entender la importancia del archivo generador como punto central de la simulación y funcionamiento correcto de SimVex. Si bien cada simulación depende de los 3 archivos

mencionados anteriormente, un mismo archivo generador puede generar múltiples archivos despachadores y cada trio de, 1 archivo generado, 1 archivo despachador de buses y 1 archivo despachador de pasajeros corresponderán a una simulación única y determinística. Todos los eventos aleatorios influyentes en la simulación ocurren en el proceso de la creación de los archivos despachadores, por lo que es improbable que 1 archivo generado genere sets de archivos despachadores idénticos en dos instancias separadas. Durante el ciclo de la simulación no existen eventos aleatorios no determinísticos, por lo que la ejecución de un mismo set de archivos entregara el mismo resultado entre cada una de sus ejecuciones.

Si bien esta dependencia entre los archivos de la simulación puede ayudar al proceso de experimentación, es responsabilidad del usuario mantener control de los archivos entregado al programa al momento de realizar una simulación, ya que si se utilizan archivos que cuentan con un diferente punto de origen, puede entregar resultados indeseados e incluso errores en la ejecución.

```
{
  "Time": {
    "Duration": 3600,
    "Tick": 0.5
  },
  "Node": [
    {
      "Street": {
        "length": 250,
        "tracks": 2
      },
      "Stop": {
        "id": "P-147",
        "berths": 2
      },
      "Street": {
        "length": 200,
        "tracks": 2
      },
      "Intersection": {
        "cicle": 70,
        "green": 0.65
      },
      "Street": {
        "length": 150,
        "tracks": 2
      },
      "Stop": {
        "id": "P-369",
        "berths": 1
      },
      "Street": {
        "length": 200,
        "tracks": 2
      }
    ]
  },
  "Route": {
    "357": {
      "stops": {
        "P-147": {
          "passenger_rate": {
            "Exponential": {
              "rate": 60.0
            }
          },
          "bus_rate": {
            "Exponential": {
              "rate": 420.0
            }
          },
          "initial_occupancy": {
            "Uniform": {
              "a": 15,
              "b": 20
            }
          }
        },
        "P-369": {
          "passenger_rate": {
            "Uniform": {
              "a": 50,
              "b": 70
            }
          },
          "bus_rate": {
            "Exponential": {
              "rate": 420.0
            }
          },
          "initial_occupancy": {
            "Uniform": {
              "a": 15,
              "b": 20
            }
          }
        }
      }
    },
    "159": {
      "stops": {
        "P-147": {
          "passenger_rate": {
            "Normal": {
              "mu": 60.0,
              "stdv": 15.0
            }
          },
          "bus_rate": {
            "Uniform": {
              "a": 540,
              "b": 600
            }
          },
          "initial_occupancy": {
            "Normal": {
              "mu": 20.0,
              "stdv": 2.0
            }
          }
        }
      }
    },
    "618": {
      "stops": {
        "P-369": {
          "passenger_rate": {
            "Fixed": {
              "rate": 45
            }
          },
          "bus_rate": {
            "Fixed": {
              "rate": 480
            }
          },
          "initial_occupancy": {
            "Normal": {
              "mu": 16.0,
              "stdv": 3.0
            }
          }
        }
      }
    }
  },
  "Buses": {
    "top_speed": 15.4,
    "acc": 1.0,
    "desc": 0.8
  },
  "Passenger": {
    "boarding_time_dist": {
      "Normal": {
        "mu": 7.0,
        "stdv": 2.0
      }
    },
    "alighting_time_dist": {
      "Uniform": {
        "a": 2,
        "b": 4
      }
    }
  }
}
```

Imagen 1. Ejemplo Archivo Generador
Fuente: Elaboración Propia

En la imagen 1. se puede ver el resultado de un archivo generado en formato json en la parte superior se determina la duración de la simulación, que tiene impacto directo en cuantos pasajeros

y buses se generaran, y el valor del tiempo que transcurre en cada tick de la simulación, este valor se puede entender como el detalle de los resultados que genera la simulación.

Después se tiene todos los nodos y sus respectivos atributos en el orden que se recorrerán en la simulación. Luego las rutas con su identificador único, y los paraderos que la operan. Llegando al final del documento se puede ver las características que definen el comportamiento de los buses como su velocidad máxima y tasa de aceleración y deceleración. Estos valores son idénticos para todos los buses que circulan la vía. Finalmente están las distribuciones que determinan el tiempo de subida y bajada de los pasajeros.

Cuando se inicia el programa lo primero que se ve es la pantalla principal con 3 botones diferentes como se ve en la imagen 2, el primero “Create New Simulation” refiere a crea una nueva simulación desde 0, creando un nuevo archivo generador y los dos archivos despachadores que dependen de él. Luego esta “Re-Use File” que reutiliza un archivo generado para crear dos nuevos archivos despachadores y generar una nueva simulación a partir de un archivo ya existente. Finalmente, “Load Files” carga los 3 archivos previamente generados para ejecutar la simulación nuevamente, esta función puede resultar útil al momento de modificar manualmente el archivo generador resultando en un mayor control de las variables dependientes e independientes durante un proceso experimental.

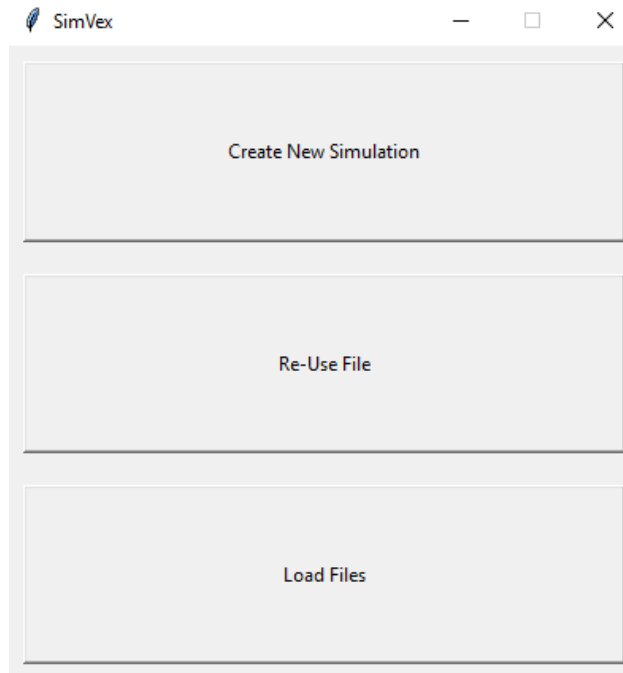


Imagen 2. Menu Principal SimVex
Fuente: Elaboración Propia

3.2.1 Crear Nueva Simulación

La finalidad de esta opción es crear todos los componentes para ejecutar una simulación desde cero, este proceso cuenta de múltiples pasos que serán detallados a continuación, se puede entender como una especie de formulario a completar. Existe una importancia en el orden en que se completa la información para la simulación, ya que existen dependencias en los elementos de la simulación, por ejemplo, para poder crear una ruta es necesario que existan paraderos sobre los cuales esta ruta va a operar.

- Creación de Nodos

Luego de apretar el botón “Create New Simulation” de la pantalla principal se llega a la vista de la imagen 3, donde el primer elemento es el panel lateral marcado por rojo en la imagen que registra en que paso de la creación de la simulación se encuentra el usuario, en la parte inferior de este panel se encuentra un botón “Generate Files” que se encuentra bloqueado hasta que toda la información requerida para crear una simulación sea proporcionada por el usuario.

Otro elemento en común para todos los pasos del proceso de creación de una simulación es el botón marcado en amarillo en la parte inferior de la vista, utilizado para confirmar la información rellena en los diferentes campos de la vista. El último elemento es el botón la parte superior de

la vista marcado con azul titulado “Add Node”, que permite agregar un nuevo nodo a la simulación, existen tres tipos de nodos diferentes que pueden ser una calle, paradero o intersección, se revisa en mayor detalle los nodos y sus propiedades en la sección 3.5.1.

Imagen 3. Vista Creacion de Nodos
Fuente: Elaboración Propia

- Creación de Rutas

En la imagen 4 se muestra la vista de creación de rutas, manera similar a la creación de nodos, para esta vista se tiene un botón para crear una nueva ruta “Add Route” marcado en azul en la parte superior, y un botón para confirmar los datos ingresados “Submit Routes” marcado en amarillo en la parte inferior.

Para crear una ruta es necesario proporcionar un identificador y seleccionar en cuales de los paraderos creados en la vista anterior operara la nueva ruta creada.

Imagen 4. Vista Creacion Rutas
Fuente: Elaboración Propia

- Configuración Paraderos

La siguiente vista en la imagen 5 corresponde a la configuración de las tasas de llegada de pasajeros a un paradero, se pide indicar la distribución que controla este comportamiento para cada uno de las rutas operando en cada uno de los paraderos.

En esta vista es la primera instancia en que se utiliza el selector de distribución, pero se utiliza reiteradamente en las vistas posteriores.

El selector de distribución cuenta con 4 posibles opciones y para cada una de ellas es necesario llenar los parámetros necesarios, las distribuciones disponibles son:

La distribución exponencial y requiere el parámetro β que equivale a $1/\lambda$, se puede entender como “cada cuanto tiempo β puede ocurrir x evento”, en este caso el tiempo se mide en unidades de segundos

La distribución uniforme, y requiere valores de mínimo a máximo, el valor obtenido de esta distribución de probabilidad son los contenidos entre el mínimo A y el máximo B, todos los valores son equiprobables.

La distribución normal, se requiere la media de la distribución y la desviación estándar

Como última opción esta la opción que no es una distribución de probabilidad, sino un valor fijo, es un intervalo de tiempo fijo en el cual ocurre un evento.

Se entiende que no todas las distribuciones van a ser adecuadas para describir el comportamiento que se quiere simular y queda a criterio del usuario determinar la distribución y valores que mejor se ajusten a cada uno de los escenarios.

Route ID	Stop ID	Distribution	Parameters
357	P-147	Exponential	Distribution Rate (seconds): 60
357	P-369	Uniform	A: 50, B: 70
159	P-147	Normal	Mean: 60, Standard Deviation: 15
618	P-369	Fixed	Arrival Rate (seconds): 44

Imagen 5. Vista Configuración Llegada de Pasajeros a Paraderos
Fuente: Elaboración Propia

- Configuración Rutas

Para la vista de configuración de llegadas de buses por ruta en la imagen 6, de manera análoga a la vista anterior se tiene que definir el comportamiento de llegada de los buses correspondiente a

cada una de las rutas creadas a la simulación. Se utiliza nuevamente el selector de distribución con las mismas condiciones explicadas anteriormente.

Imagen 6. Configuración Llegada de Buses a Simulación
Fuente: Elaboración Propia

- Configuración Pasajeros

Para la vista de configuración de pasajeros en la imagen 7 los únicos parámetros que es necesario proporcionar son las distribuciones que generan la variable aleatoria para los valores de los tiempos de subida y bajada de un pasajero. Dependiendo de los valores de origen y destino de un pasajero es si es que este hará uso de los valores de abordaje y descenso, pasajeros que se encuentran a bordo de un bus en circulación y que su destino es el final de la simulación jamás van a participar del proceso de intercambio de pasajeros entre un bus y un paradero.

SimVex

Nodes Creator

Route Creator

Stop Configuration

Route Configuration

Passenger Action Configuration

Passenger Occupancy Configuration

Generate Files

Passenger Boarding/Alighting Configuration

	Distribution	Mean	Standard Deviation
Passenger Boarding Distribution	Normal	7	2

	Distribution	A	B
Passenger Alighting Distribution	Uniform	2	4

Submit Passenger Configuration

Imagen 7. Configuración tiempos de Subida/Bajada Pasajeros
Fuente: Elaboración Propia

- Ocupación Inicial de Pasajeros en Buses

La vista de configuración de ocupación inicial vista en la imagen 8 se necesita seleccionar la distribución y los parámetros que generan el valor de la variable aleatoria que determina cuantos pasajeros se encuentran a bordo de un bus al entrar en la simulación para cada una de las rutas definidas para la simulación.

Route ID	Distribution	A	B
357	Uniform	15	20

Route ID	Distribution	Mean	Standard Deviation
159	Normal	20	2

Route ID	Distribution	Mean	Standard Deviation
618	Normal	16	4

Imagen 8. Configuración Ocupación Inicial de Buses por Pasajeros
Fuente: Elaboración Propia

Una vez se rellene los valores requeridos en cada una de las vistas y se presione el botón para confirmar los datos en la vista de configuración de ocupación inicial de pasajeros se habilitará el botón en la parte inferior izquierda de la vista para proceder a crear los archivos e inmediatamente ejecutar la simulación.

3.2.2 Regenerar Simulación

Para la vista de regeneración de simulación que aparece en la imagen 9, lo único que se ve es un campo de texto acompañado de un botón, la finalidad de ese campo de texto es indicar la dirección del archivo generado objetivo para crear los nuevos archivos despachadores. El único otro elemento es un botón en la parte inferior de la vista para confirmar la dirección del archivo y proceder con la generación, simulación y obtención de resultados.

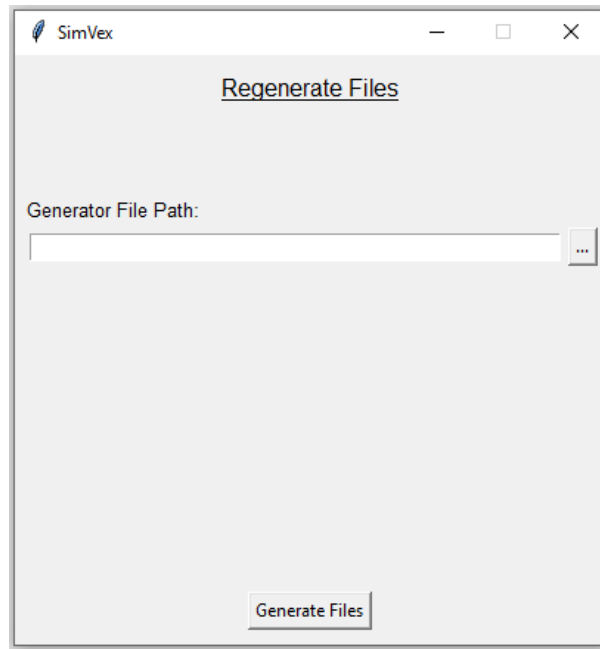


Imagen 9. Vista Regenerar Archivos a partir de Generador
Fuente: Elaboración Propia

3.2.3 Reutilizar Simulación

La vista de reutilización de archivos en la imagen 10, es similar a la anterior, con la diferencia que acá hay que indicar la dirección para los 3 tipos de archivos.

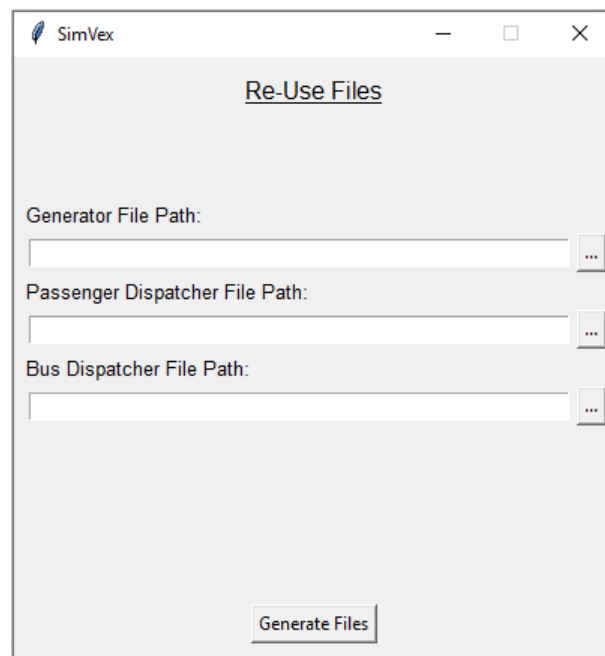


Imagen 10. Vista Reutilizacion Archivos
Fuente: Elaboración Propia

3.3 RESULTADOS

Para efectos de cada simulación ejecutada en SimVex se genera un reporte de resultados en formato de un archivo pdf, es en este archivo donde se presentan de forma ordenada datos de interés.

3.3.1 Datos de entrada

Se presenta una tabla que tiene todos los datos relevantes respecto a la configuración de la simulación, esto ayuda al usuario a comprender de manera más directa el contexto en el cual se ejecutó la simulación. Los valores corresponden a:

- Duración total de la simulación en segundos
- Numero de paraderos simulados junto con los identificadores de cada uno
- Numero de intersecciones simuladas
- Numero de rutas simulados junto con los identificadores de cada uno
- Distancia que comprende la simulación (sumatoria de la longitud de todos los nodos tipo calle)
- Flujo de buses [Bus/hora] del número de buses que circulan por la vía por hora
- Número total de pasajeros simulados
- Demanda del ascenso de pasajero
- Demanda del descenso de pasajeros

Además, se incluye una tabla adicional detallando la configuración de cada una de las intersecciones en el orden que se recorren en la simulación.

3.3.2 Datos Buses Simulados

Para cada uno de los buses de la simulación se presentan los siguientes valores:

- Identificador del bus
- Ruta a la que pertenece
- Velocidad Comercial [m/s]: distancia de la simulación/tiempo en la simulación
- Tiempo de viaje: tiempo que el bus permanece en un nodo tipo calle
- Tiempo en cola de espera de un paradero
- Tiempo en cola de operación de un paradero
- Tiempo detenido en la cola de una intersección
- Demora total en la simulación: la suma de los tiempos de cola (bus e intersección) y tiempo de permanencia en un paradero.

3.3.3 Datos Paradas Simuladas

Para cada uno de los paraderos de la simulación se presentan los siguientes valores:

- Identificador del paradero
- Buses atendidos por hora [Bus/hora]
- Valor máximo de buses en cola de espera
- Valor promedio de buses en cola de espera
- Valor desviación estándar de buses en la cola de espera
- Valor máximo de tiempo de espera de pasajeros en el paradero
- Valor promedio de tiempo de espera de pasajeros en el paradero
- Valor desviación estándar de tiempo de espera de pasajeros en el paradero
- Valor máximo de pasajeros concurrentes esperando en el paradero
- Valor promedio de pasajeros concurrentes esperando en el paradero
- Valor desviación estándar de pasajeros concurrentes esperando en el paradero

3.3.4 Datos Generales Simulación

Para la vía arterial se genera una tabla general de resultados y está compuesta de:

- Tiempo total de viaje: la sumatoria de todos los tiempos de viaje de los buses simulados
- Velocidad comercial promedio de los buses simulados
- Tiempo acumulado de todos los buses de la detención en intersecciones
- Tiempo acumulado de todos los buses del tiempo en paraderos (tiempo en cola + tiempo en operación)

3.4 ARQUITECTURA

La arquitectura de SimVex se basa en un patrón de arquitectura de monolito, donde todo el programa se encuentra contenido en un mismo entorno. Para esta aplicación se optó por Python por varios motivos, la simplicidad y fácil lectura que ofrece la sintaxis ayuda a agilizar el proceso de desarrollo y cuenta con una gran colección de librerías standard al lenguaje que resultan útiles en varios escenarios. En la imagen 11 se aprecia un diagrama de los módulos que componen esta arquitectura, y de la manera que estos interactúan entre ellos. Se prestó especial atención en mantener modularidad al momento de desarrollar SimVex para mantener la dependencia entre cada una de las partes en un mínimo, con la finalidad de simplificar la tarea de desarrollar y

extender las capacidades de la aplicación. A continuación, se explicará la importancia y el rol que cumple cada uno de los módulos.

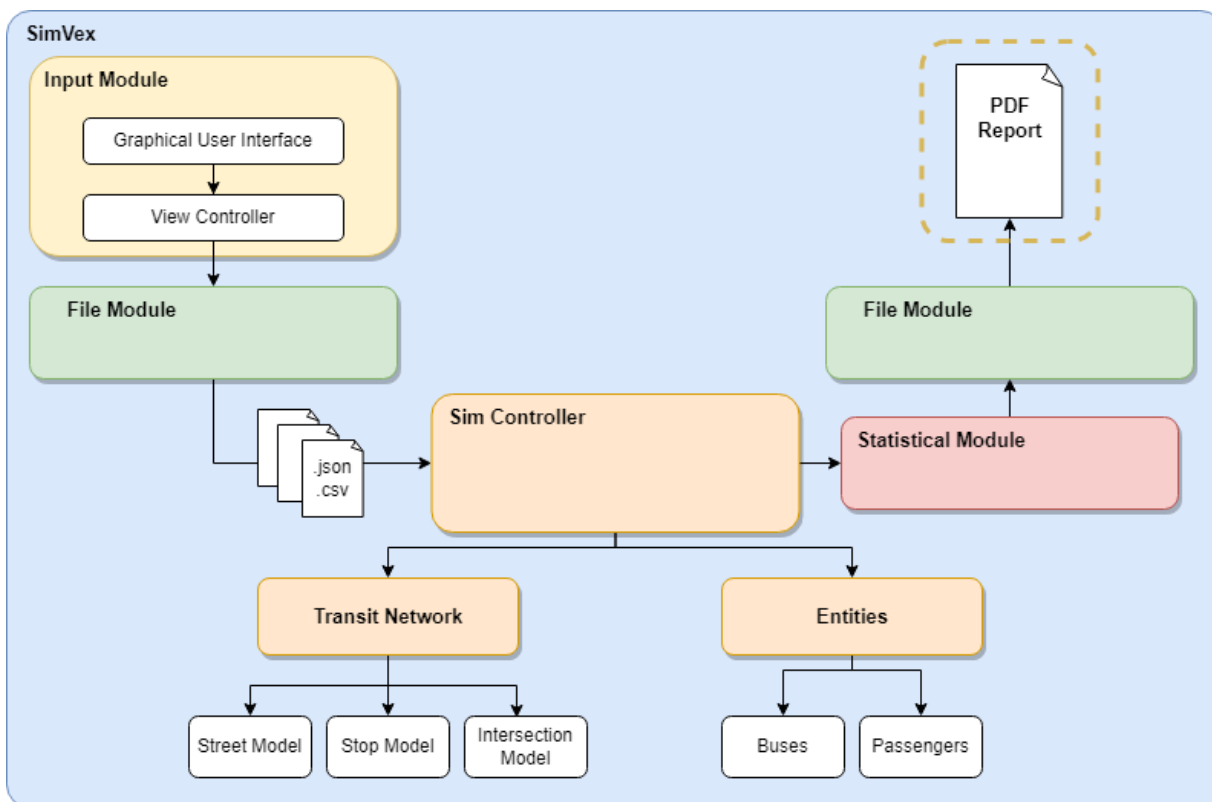


Imagen 11. Arquitectura General SimVex

Fuente: Elaboración Propia

3.4.1 Modulo de Entrada

El módulo de entrada está conformado por la interfaz gráfica de SimVex, desarrollada principalmente utilizando la librería estándar TKinter, acá es donde el usuario selecciona una simulación existente para ejecutar, o crea una simulación nueva desde cero, el detalle de este proceso se presentó en la sección 3.2.1. El controlador de la vista recolecta los datos e información generada en desde la interfaz gráfica para o crear los archivos de una simulación o proporcionar los archivos correspondientes al controlador de simulación para que este pueda hacer su trabajo.

3.4.2 Modulo de Archivos

El módulo de archivos se encarga de la lectura y escritura de todos los archivos utilizados en SimVex. Tiene la capacidad para manejar los tres formatos de archivos utilizados en el programa,

utilizando json para el archivo generador, csv para los archivos despachadores de pasajeros y buses y pdf para el reporte de resultados.

3.4.3 Controlador de Simulación

El controlador de simulación se encarga de unir y darle cohesión a todas las partes que conforman la simulación, este parte por inicializar la data correspondiente a las características de la simulación como los nodos, tiempos de ejecución, además de cargar los valores de entrada para buses y pasajeros, y construir los objetos a simular. la siguiente tarea es ejecutar el bucle principal de la simulación. Otra de las tareas del controlador de simulación es la recopilación de los datos para cada una de las entidades simuladas en cada iteración de la simulación

3.4.4 Modulo Estadístico

Modulo estadístico recopila todos los datos de los modelos de datos generado por la simulación y genera valores de interés para incluir en el reporte de resultados de la simulación. los resultados generados se presentaron en el punto 3.3

3.5 MODELOS DE DATOS

A continuación, se presentarán los modelos de datos más relevantes en el formato de clases de un diagrama de clases, se presentarán únicamente los atributos que pertenecen a cada una de las clases y se explicaran en mayor detalle aquellos que no se consideran auto explicativos. La totalidad del diagrama de clases está incluida como parte de los anexos de este trabajo.

3.5.1 Nodos

El nodo es la clase padre del cual las clases “Street”, “Stop” e “Intersection” heredan, su función es ser la pieza básica desde la cual la se construye la vía exclusiva donde operan los buses en la simulación. en términos de estructuras de dato es un nodo bidireccional que cuenta con referencias al nodo que lo precede y el que le sigue.

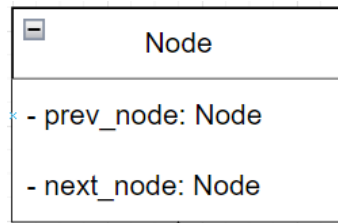


Imagen 12. Modelo de Clases Nodo
Fuente: Elaboración Propia

- Paradero

Los paraderos son nodos donde un autobús puede detenerse. Las condiciones para determinar si un autobús se detiene en un paradero son las siguientes: existen pasajeros a bordo del autobús que tienen como destino el paradero en cuestión, o hay pasajeros en el paradero que utilizan la ruta del autobús y necesitan abordarlo.

Cuando un autobús se detiene en un paradero, busca la primera posición disponible entre las colas del paradero. Se da prioridad a la cola de operación y luego a la cola de espera.

La función principal de un paradero es permitir el intercambio de pasajeros entre el autobús y los usuarios. Una característica importante son las colas utilizadas para gestionar la operación de los autobuses. El paradero cuenta con una cola de operación, cuya dimensión se determina por el número de andenes disponibles.

Además, hay una cola de espera en el paradero, cuya capacidad se establece según la longitud de la calle que antecede al paradero. Para que un autobús pase de la cola de espera a la cola de operaciones, debe haber espacio disponible en la cola de operaciones y no debe haber obstrucciones aguas arriba.

Una vez que un autobús ha completado la transferencia de pasajeros desde la cola de operación, su salida depende principalmente del número de carriles en la calle que precede al paradero. Si la calle tiene un solo carril, el autobús no debe estar obstruido para salir (se considera obstrucción cuando hay otro autobús delante). Si la calle tiene dos carriles, el autobús puede "adelantar" al

autobús que tiene delante utilizando el segundo carril (esto se refiere a un adelantamiento virtual, ya que no se permite el cambio de carril real).

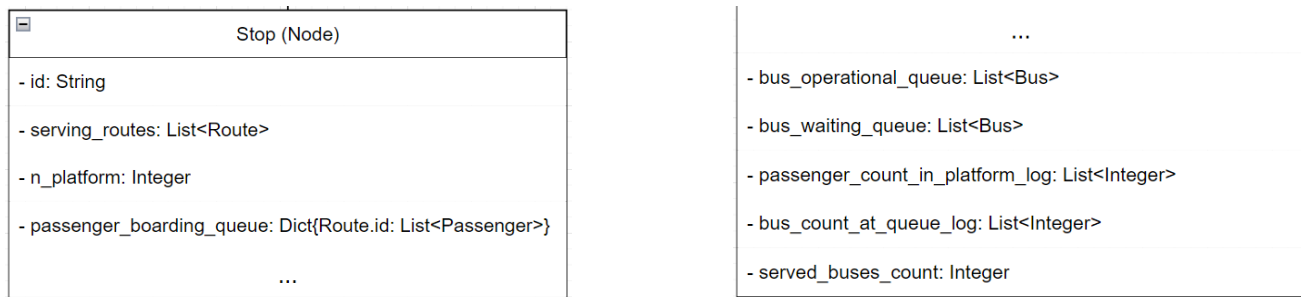


Imagen 13. Modelo de Clases Paradero
Fuente: Elaboración Propia

- Intersección

Las intersecciones son otro tipo de nodo donde un autobús puede detenerse. En una intersección, el semáforo puede estar en uno de tres estados: "G" para verde, "Y" para amarillo y "R" para rojo. Estos estados se presentan en una secuencia cíclica y continua. Al iniciar la simulación, el valor del semáforo siempre es verde, con el tiempo máximo restante correspondiente a la duración de la luz verde.

La condición para que un autobús se detenga en una intersección es que el valor del semáforo sea "R". A diferencia del nodo "Stop", las intersecciones cuentan con una única cola de detención, donde los autobuses se posicionan uno detrás del otro en el orden de llegada a la intersección.

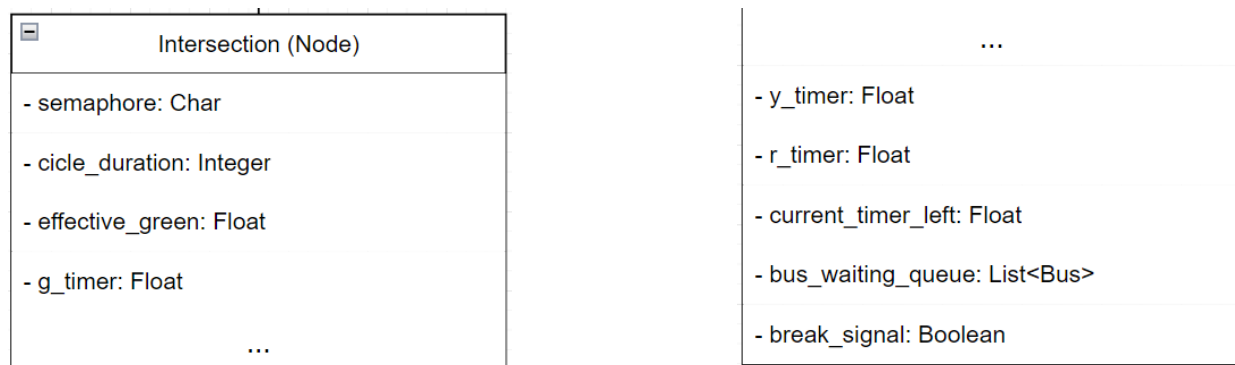


Imagen 14. Modelo de Clases Interseccion
Fuente: Elaboración Propia

- Calle

Los nodos de tipo "Street" (calle) son los únicos que tienen una dimensión física definida, la cual incluye su longitud y el número de carriles que poseen. Existe una dependencia entre los nodos de tipo "Stop" (paradas) e "Intersection" (intersecciones) y los nodos de tipo "calle". Esto se debe a que las dimensiones de las colas presentes en los nodos de paradas e intersecciones se calculan en función de la longitud de la calle que los precede.

El nodo de tipo "calle" actúa como un medio de viaje entre los nodos de parada e intersección, permitiendo el desplazamiento de un nodo de detención a otro. Es a lo largo de estas calles donde los vehículos circulan

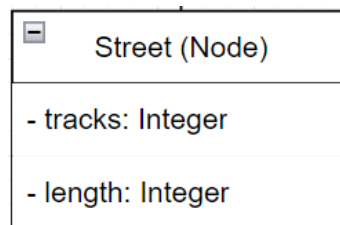


Imagen 15. Modelo de Clases Calle
Fuente: Elaboración Propia

3.5.2 Entidades

El papel de las entidades es interactuar entre ellas utilizando la red de nodos como medio para realizar estas acciones.

- Buses

Los buses son la entidad principal de la simulación, debido que una de las grandes motivaciones de SimVex es estudiar el comportamiento de estos en diferentes configuraciones.

Todos los buses tienen un identificador único, y pertenecen a alguna de las rutas existentes en la programación de la simulación. Un bus entra a la simulación en un tiempo determinado (arrival_time). Todos los buses comparten la misma configuración respecto a la velocidad máxima, tasa de aceleración y tasa de desaceleración, además el movimiento de los buses es controlado por

un movimiento uniformemente acelerado. Cuando un bus se encuentra en la simulación siempre tiene una referencia respecto al nodo en el que se encuentra operando en cualquier instante (location), además de contar con una variable para determinar en qué sección del nodo se encuentra el bus (uso específico para nodos tipo calle).

Existen varias variables de tipo booleano para un modelo bus, que se encargan de controlar y revisar las condiciones que debe cumplir un bus para determinar si es que se debe detener o no en el siguiente nodo. “stop_flag” será verdadero cuando el próximo nodo es un paradero, y el bus tiene abordo a pasajeros que su destino coincide con el próximo paradero o, hay pasajeros esperando en el pasajero que necesiten abordar el bus. La variable "check_stop_flag" es utilizada en el contexto de que el siguiente nodo del trayecto sea de tipo intersección, y tiene como objetivo determinar se revisaran las condiciones de frenado. Se calcula una distancia mínima que el bus recorre para llegar a estar detenido dependiendo de la actual velocidad de viaje, y si la distancia entre la posición actual del bus y la siguiente intersección es menor a esta distancia de frenado, la variable "check_stop_flag" se establece en falso, ignorara todas las condiciones que determinan si es que el bus debe detenerse o no, se puede considerar este comportamiento como cruzar la intersección con luz amarilla.

Un bus puede estar en uno de cuatro estados de circulación, y las posibilidades, así como los cambios de estados posible se pueden ver en el diagrama de la imagen 16. Un bus siempre es responsable de sus propios cambios de estados, a excepción del caso estacionario-acelerando, donde la indicación la recibe por parte del nodo en el que se encuentra detenido.

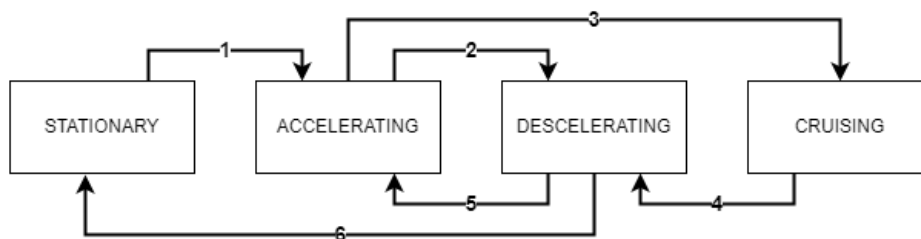


Imagen 16. Diagrama Cambio de Estado de un Bus
Fuente: Elaboración Propia

1.	Bus abandona una intersección o paradero
2.	Bus llega a "breaking_point" y "stop_flag" es igual a "True"
3.	Bus alcanza velocidad maxima
4.	Bus llega a "breaking_point" y "stop_flag" es igual a "True"
5.	"stop_flag" cambia de "True" a "False"
6.	Bus cambia de un nodo calle a un paradero/intersección y "stop_flag" es "True"

El resto de las variables son registro de los estados del bus en cada iteración de la simulación, que son utilizadas por el módulo estadístico para generar los valores del resultado de la simulación.

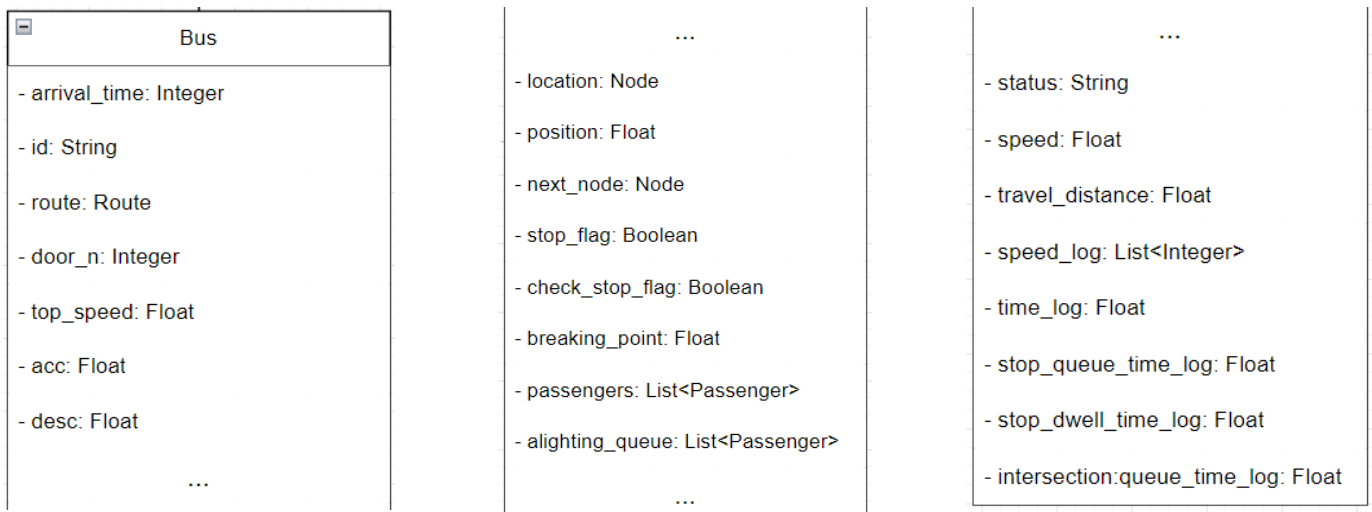


Imagen 17. Modelo de Clases Bus
Fuente: Elaboración Propia

- Pasajeros

Los pasajeros llegan a la simulación en un instante determinado a su lugar de origen predefinido. Este lugar de origen puede ser un paradero o un autobús. Si el origen es un autobús, el tiempo de llegada del pasajero está determinado por el tiempo de llegada del autobús en el que se encuentra a bordo. Esta información se establece durante el proceso de creación del archivo despachador de pasajeros y autobuses.

El destino del pasajero puede ser el final de la simulación o algún otro paradero que opere la ruta que el pasajero utiliza, la única restricción es que el destino del pasajero se debe encontrar aguas abajo respecto al origen de este.

Un pasajero esperando en un paradero siempre abordara un bus que opere la ruta una vez este se encuentre en la cola de operación del paradero.

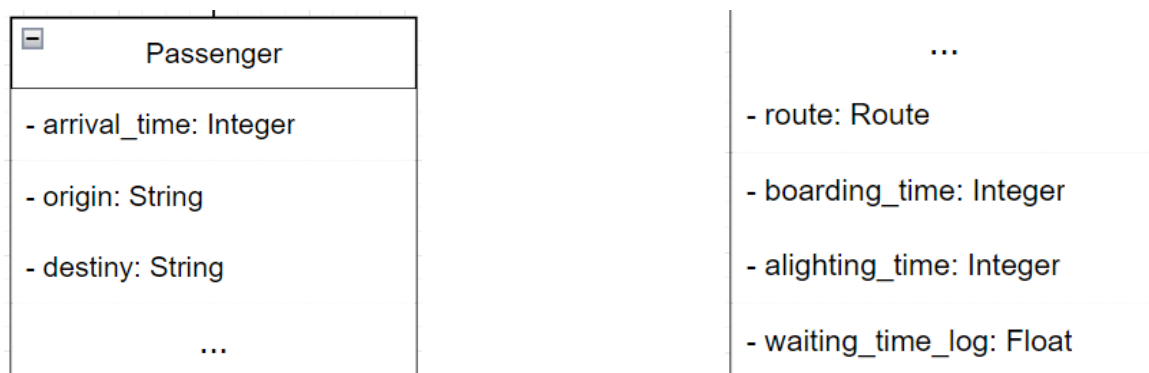


Imagen 18 Modelo de Clases Pasajero
Fuente: Elaboración Propia

3.5.3 Rutas

Una ruta cuenta con un identificador único y una lista de todos los paraderos sobre los cuales opera. La clase ruta es importante ya que es la que crea la conexión entre los modelos de pasajeros, buses y paradero.

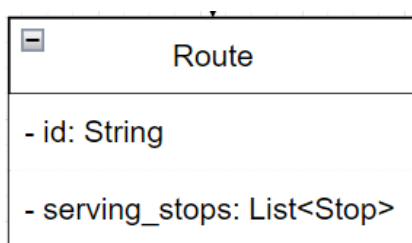


Imagen 19 Modelo de Clases Ruta
Fuente: Elaboración Propia

3.5.4 Controlador de simulación

El controlador de la simulación tiene un papel fundamental en el funcionamiento del programa, siendo responsable de coordinar y ordenar el flujo de los eventos que ocurren en la simulación. Su principal tarea es asegurar que los eventos se ejecuten en el orden correcto, manteniendo la coherencia temporal de la simulación.

Además, el controlador de la simulación actúa como un punto central para reunir y gestionar los modelos de datos de las diferentes clases involucradas. Recopila la información de los autobuses, paraderos, rutas, pasajeros y otras entidades, asegurándose de que estén sincronizadas y se compartan adecuadamente entre los componentes del programa.

Una de las responsabilidades clave del controlador de la simulación es mantener un registro de los estados de las entidades durante cada iteración de la simulación. Esto implica realizar un seguimiento de la posición y el estado de los autobuses, los tiempos de llegada y partida en los paraderos, los eventos de intersecciones semaforizadas y otros detalles relevantes. Este registro de estados permite analizar y evaluar el funcionamiento de la simulación, así como obtener métricas y datos importantes para el análisis posterior.



Imagen 20. Modelo de Clases Controlador Simulacion
Fuente: Elaboración Propia

3.6 SIMULACION

La simulación se divide en dos partes principales: la inicialización y la ejecución del loop principal.

3.6.1 Inicialización

Para la correcta inicialización de los datos previos a ejecutar la simulación el controlador debe realizar tareas en el siguiente orden:

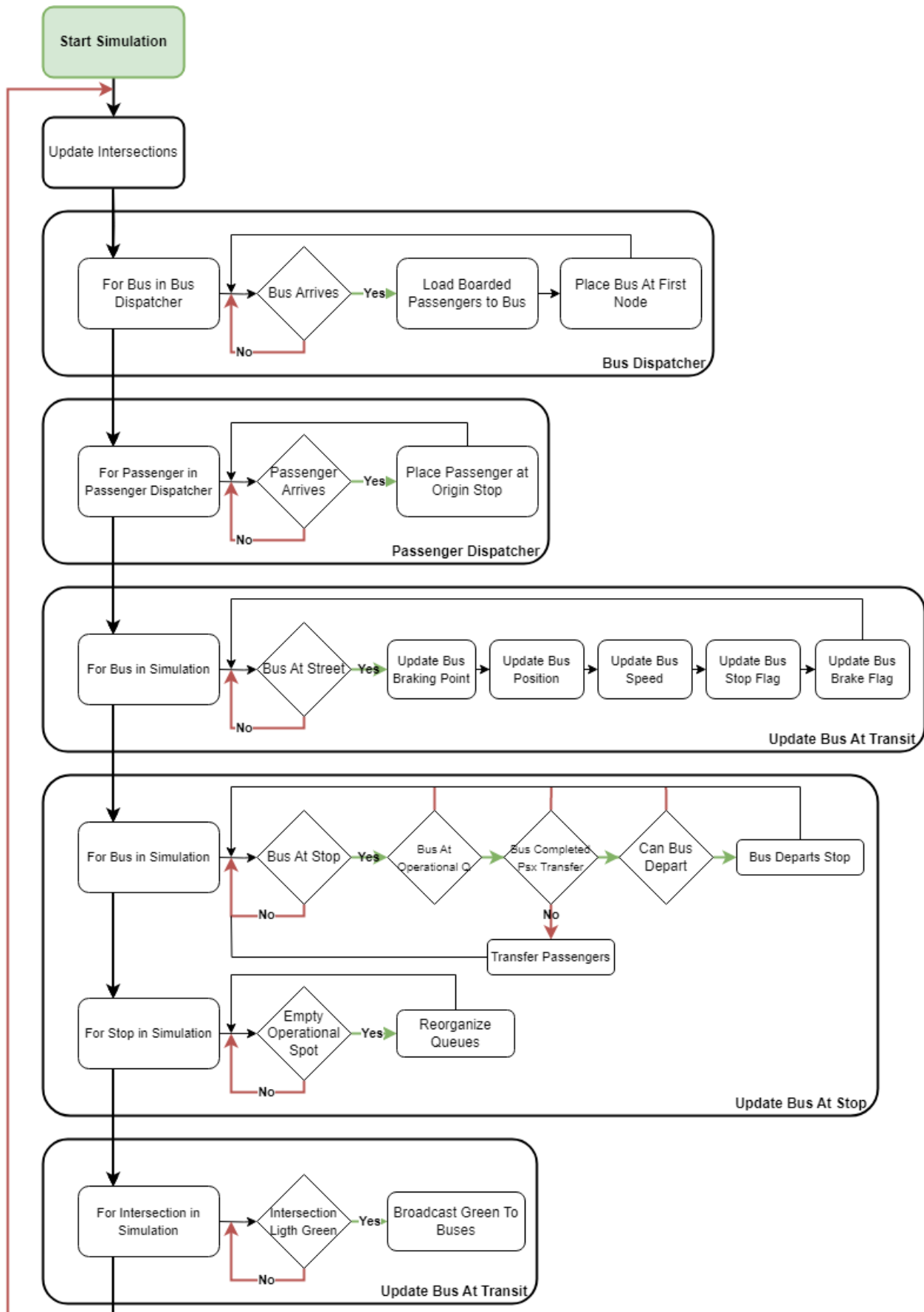
- **Crear Nodos:** Se extrae la información de los nodos desde el archivo generador, se crean los respectivos objetos y se proporcionan como referencia para crear un objeto de la clase TransitNetwork, el cual será utilizado por el controlador de la simulación.

- Crear Rutas: se extrae la información de las rutas desde el archivo generador y de los paraderos desde el objeto TransitNetwork para crear las rutas que se simularan
- Asignar Rutas a los paraderos: A cada uno de los paraderos se le agrega la referencia a las rutas que operaran en este durante la simulación
- Generar Pasajeros: Desde la información disponible en el archivo de pasajeros, se crean todos los objetos de pasajeros para la simulación
- Determinar acción de pasajeros: con la finalidad de simplificar el trabajo del módulo estadístico al finalizar la simulación, se crean dos listas, la primera contiene a todos los pasajeros que se sabe realizaran una acción de abordar un bus durante un intercambio de pasajeros, y la otra para los pasajeros que realizaran una acción de descender de un bus durante el intercambio de pasajeros. (un mismo pasajero puede aparecer en ambas listas)
- Generar Buses: desde la información disponible en el archivo de buses, se crean todos los objetos de buses para la simulación
- Crear Despachador de pasajeros: se crea la lista que se utilizara para manejar el ingreso de pasajeros a la simulación
- Crear Despachador de buses: se crea la lista que se utilizara para manejar el ingreso de buses a la simulación
- Poblar paraderos con pasajeros: desde la lista de pasajeros que entran a la simulación, se revisan todos los pasajeros que tienen “arrival_time” = 0, es decir que se encuentran en la simulación desde el inicio
- Generar colas en paraderos e intersecciones: se calculan las dimensiones de las colas para los paraderos e intersecciones en función de las calles que los anteceden

- Inicializar buses en punto de partida: se referencia a todos los buses de la simulación su posición dentro de la simulación, que sería el primer nodo de partida del objeto TransitNetwork

3.6.2 Ejecución

La ejecución del loop principal de la simulación se realiza en función de tics, cada tic equivale a un salto en el tiempo donde cada una de las entidades de la simulación realizan alguna acción. En la imagen 21 se puede ver el flujo de la simulación de manera resumida.



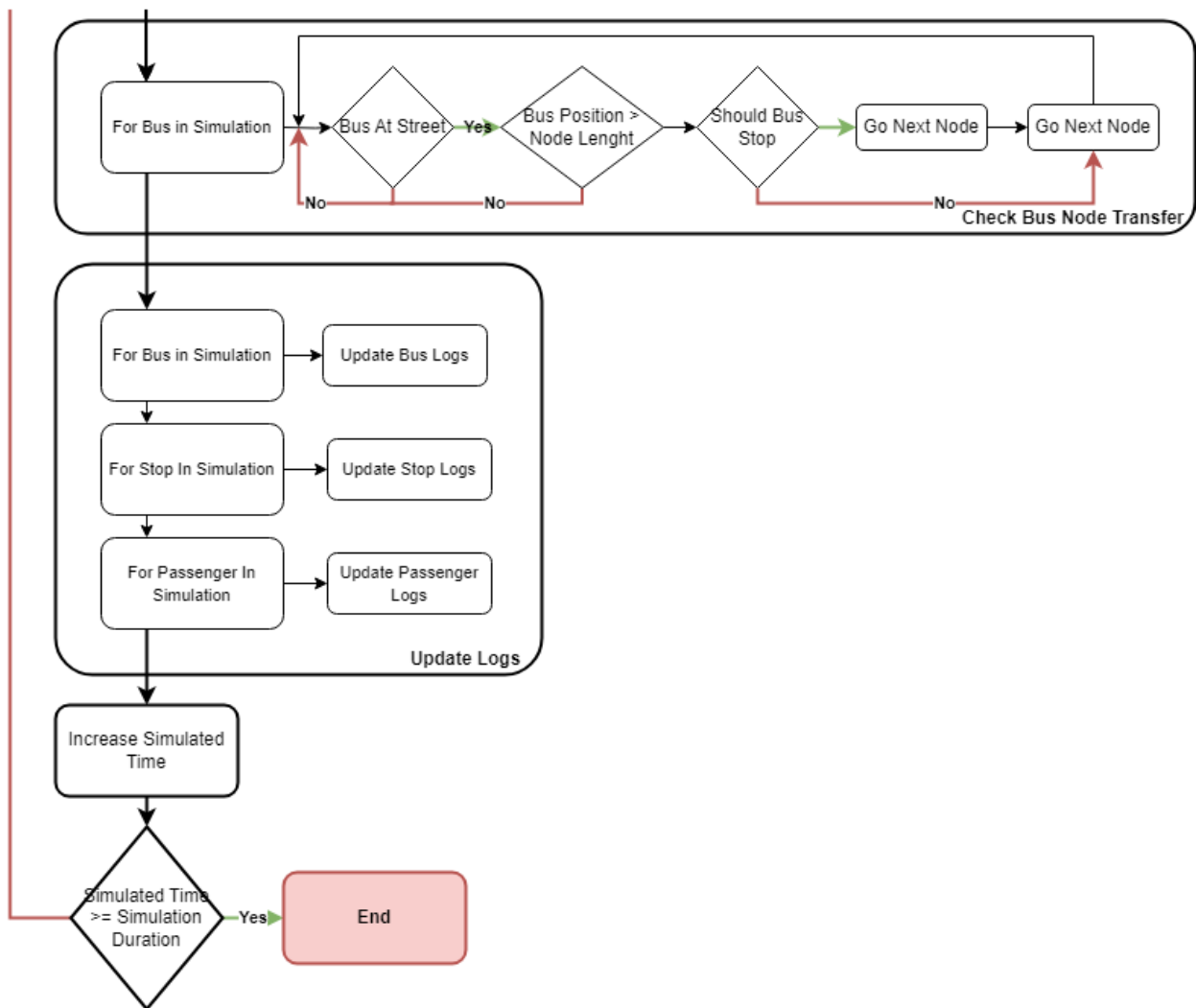


Imagen 21 Diagrama de Flujo Ejecución Bucle Simulación
Fuente: Elaboración Propia

4. CONCLUSIONES Y RECOMENDACIONES

4.1 RESULTADOS

Durante el proceso de desarrollo del programa SimVex, se lograron alcanzar los objetivos principales establecidos. Se creó una plataforma efectiva para la simulación de vías exclusivas de autobuses, donde se tuvo en cuenta la flexibilidad en la configuración de la programación de la simulación. Esto permitió adaptar la simulación a diferentes escenarios y condiciones específicas, brindando un alto grado de personalización y ajuste a las necesidades del usuario.

Sin embargo, durante el desarrollo del programa, surgieron algunas deudas técnicas relacionadas con el propio código. Estas deudas técnicas no afectan el correcto funcionamiento del programa, pero son relevantes abordarlas para garantizar la mantenibilidad y facilitar futuras extensiones del programa.

Además de quedar pendiente tareas como la validación del modelo que certificarían el uso de SimVex como herramienta de diseño para el transporte publico

4.2 VALIDACION DE MODELO

Es imperativo que se valide la capacidad del programa para modelar y simular escenarios de la realidad. Una validación adecuada garantiza que el modelo represente con precisión la realidad y brinde resultados confiables. Esto es esencial para tomar decisiones informadas en la planificación y gestión del transporte público, ya que permite evaluar el desempeño de la infraestructura y optimizar la operación del sistema. Se sugiere los siguientes pasos para realizar la validación de SimVex como herramienta de simulación de vías exclusiva para el transporte público:

- **Recopilación de datos:** Es importante reunir datos reales del sistema de transporte con el que se comparan los valores obtenidos desde la simulación, el enfoque de los datos obtenidos en terreno deben ir de acuerdo a los parámetros de entrada que se necesitan para crear una simulación en SimVex, en específico datos que describa distribuciones de la población como la llegada de buses al sistema, ocupación inicial de estos buses, tasas de llegadas de pasajeros por ruta, y tiempos de subida y baja de los pasajeros.

- Definición de métricas: Definir el conjunto de métricas con las que se compararan los resultados simulados con los valores reales observado.
- Testeo y calibración del modelo: Proceso iterativo donde se evalúa y ajusta el modelo acorde se necesite para alcanzar los resultados esperados en función de las métricas establecidas. Es necesario testear el programa contra varios escenarios reales, para no llegar a un sobreajuste del simulador en un único contexto.
- Análisis de resultados y documentación: dejar demostrado las capacidades del modelo y como los resultados de SimVex se comparan con las observaciones de la realidad.

4.3 MEJOREAS EN LA IMPLEMENTACION

4.3.1 Control de errores

Una recomendación clave para mejorar la calidad del programa es implementar un sólido control de errores y una adecuada sanitización de los inputs. Estas medidas ayudarán a prevenir fallos inesperados, vulnerabilidades y comportamientos inadecuados en el programa.

Durante el desarrollo del programa, se realizaron esfuerzos para implementar medidas de control de sanitización y manejo de errores de inputs utilizando los métodos disponibles en la propia biblioteca de la interfaz gráfica utilizada. Sin embargo, se encontró que estos esfuerzos no lograron alcanzar el nivel deseado de control.

La parte susceptible a este problema es la interfaz gráfica y el controlador de la vista. En estos componentes, es importante asegurarse de que se realice un control extenso de los valores y tipos de datos que se ingresan en cada campo. Esto implica validar y sanitizar los inputs de manera adecuada, considerando la longitud, el tipo de dato esperado y cualquier otro requisito específico dependiendo de la naturaleza del dato. La falta de un control exhaustivo en estos aspectos puede dar lugar a errores de lógica una experiencia deficiente para el usuario.

Además, es esencial contar con una estructura sólida para la red de tránsito que se está creando. Asegúrate de que la estructura de la red se mantenga coherente y consistente según lo que se espera recibir para la simulación, se elabora más respecto a cómo solucionar esto en la sección 4.3.3.

4.3.2 Despejar Modelos

Existen atributos en algunas clases que podrían considerarse redundantes. Por ejemplo, en la clase "Buses" se encuentra el atributo "next_node", y en la clase "SimController" se encuentran listas de tipos de nodos de la red de tránsito. Estos atributos no corresponden adecuadamente a las responsabilidades de dichas clases.

Sería apropiado para el segundo caso que esa información se encuentre en la clase "TransitNetwork" ya que es la clase encargada de gestionar la red de tránsito en su totalidad. Así, los atributos relacionados con los nodos y su interconexión estarían ubicados en la clase que tiene la responsabilidad principal de manejar esa información.

Mantener orden adecuado en las responsabilidades trae beneficios al desarrollo del programa por que mejora el modularidad, facilita la reutilización del código y ayuda una mejor comprensión de este.

4.3.3 Fortaleces la jerarquía de nodos

En el estado actual del SimVex no se tiene un mayor control en las conexiones que se pueden crear combinando los diferentes tipos de nodos, es decir, un paradero puede ser precedido por otro paradero y tener una intersección después. No existen mayores restricciones en el orden específico en el que deberían de aparecer los nodos. Se propone crear un nivel más de herencia entre la clase nodo y los diferentes tipos (calle, paradero, intersección) que separe los tipos de nodo en dos clases más:

- **Nodo Tipo A:** Este tipo de nodo incluye tanto a paraderos como a intersecciones, vendrían siendo los nodos donde un bus puede o no detenerse. La particularidad de estos nodos es que no cuentan con dimensionalidad como parte de su modelado, para efectos del programa se consideran como puntos en el espacio, por lo que no corresponde la circulación de los buses de un punto a otro.
- **Nodo Tipo B:** este tipo de nodo incluye únicamente a nodos de clase calle, se puede entender como nodo de circulación o intermediario, por donde un bus circula entremedio de dos nodos tipo A.

La restricción que debería incluirse en esta implementación es que no puede haber nodos tipo A uno tras otro, debe existir un nodo tipo B entre cada par de nodos A.

4.3.4 Extender modelo de buses

Las características que modelan el comportamiento de los autobuses, como la velocidad máxima, la tasa de aceleración y la tasa de desaceleración, son idénticas para todos los autobuses en la simulación. La única diferencia entre los autobuses es el número de puertas, que se determina de manera aleatoria entre los valores de 1 a 3 al momento de crear el archivo despachador de autobuses.

Para simplificar la clase "Bus" y tener un mayor control sobre la configuración propia de la simulación, se sugiere crear una clase llamada "BusType". Esta clase sería responsable de gestionar los atributos mencionados anteriormente de manera predeterminada.

La idea sería tener una variedad de tipos de autobuses, preferiblemente inspirados en la realidad, que puedan asignarse a las rutas. Al configurar las rutas, se le daría al usuario la opción de determinar qué tipos de autobuses operan en esa ruta en particular. Incluso se podría considerar la posibilidad de configurar el porcentaje de aparición de cada tipo de autobús en una misma ruta.

Esta implementación permitiría ampliar aún más la capacidad experimental de SimVex, ya que se podrían explorar diferentes configuraciones de autobuses y analizar su impacto en la simulación.

5. BIBLIOGRAFIA

Gibson, J. Baeza, I. and Willumsen, L.G. (1989). Bus-Stops, congestion and congested bus-stops. *Traffic Engineering & Control*, 30(6), 291-302.

Fernández, R. y Peñalillo, R. (2000). Un modelo microscópico de interacción entre paraderos de buses. *Actas XI Congreso Panamericano de Ingeniería de Tránsito y Transporte*, Gramado 19-23 noviembre 2000, 235-248.

Fernandez, R. (2003). Study of bus stop operations on arterial roads by simulation. *ITE Journal* 73(4), 77-81.

Cortés, C.E., Burgos, V and Fernández, R. (2010). Modelling passengers, buses and stops in traffic microsimulation. Review and extensions. *Journal of Advanced Transportation*, 44, 72-88.

Fernández, R. Cortés, C.E. and Burgos, V. (2010). Microscopic simulation of transit operations. Policy studies with the MISTRANSIT application programming interface. *Transportation Planning and Technology*, 33(2), 157-176.

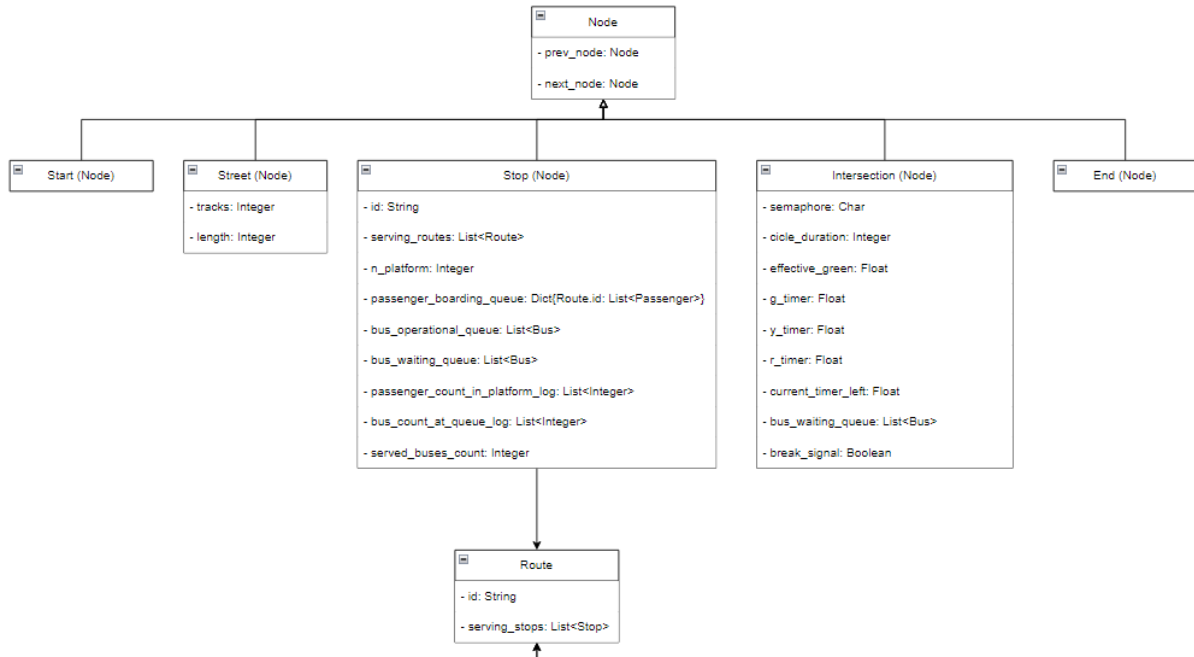
Fernandez, R. (2010). Modelling public transport stops by microscopic simulation. *Transport Research Part C: Emerging Technologies* 18(6), 856-868.

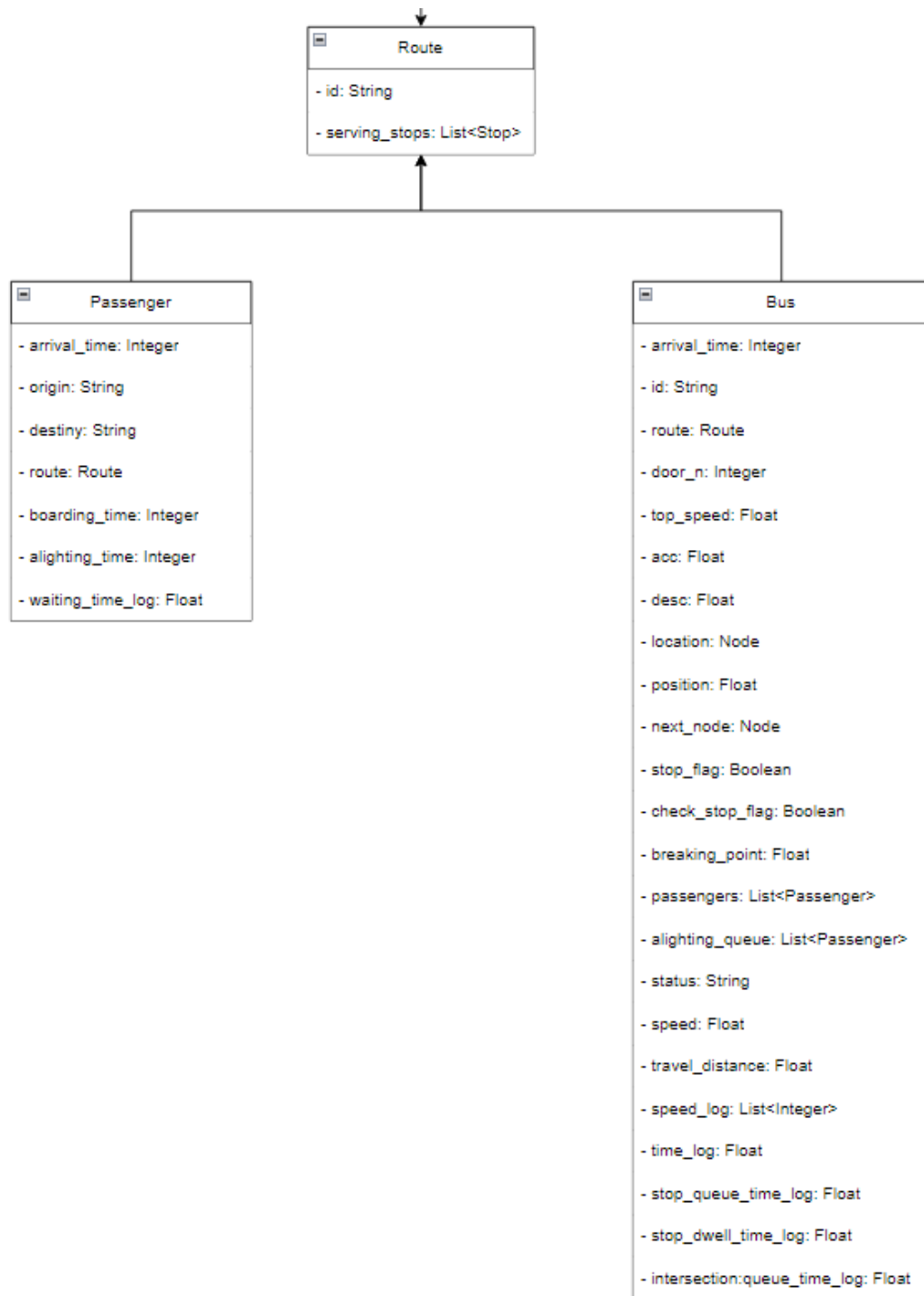
Montero, A. (2020) Modelo microscópico de simulaciones múltiples entre paraderos de buses en eje vial.

Benavente, J.M (2021). Avances hacia el desarrollo de un software de modelación de buses en ejes arteriales.

6. ANEXOS

Anexo 1. Diagrama UML Clases





Anexo 2. Documento de salida. Reporte SimVex

SimVex Report Sample

Simulation Input Data

Simulated Time:	3600 seconds	
Nº Stops Simulated:	2	['P-147', 'P-258']
Nº Intersections Simulated:	1	
Nº Routes Simulated:	3	['426', 'C-09', 'C-19']
Simulated Distance:	600 [m]	
Bus Flow:	15 [bus/h]	
Total Passengers Simulated	413 [pax/h]	
Boarding Demand	188 [pax/h]	
Alighting Demand	166 [pax/h]	

Intersection Number	Cycle Duration (s)	Green Ratio
Intersection 1	90	0.5

Bus Data Sheet

Bus Id	Route	Comercial Speed[km/h]	Travel Time[min]	Stop Q Time[s]	Dwell Time[s]	Intersection Time[s]	Total Delay[s]
b-12	C-19	30.64	0.88	0	17.0	0	17.0
b-11	C-19	25.71	0.88	0	30.5	0	30.5
b-10	C-19	25.12	0.88	0	32.5	0	32.5
b-13	C-19	18.08	0.88	0	66.0	0	66.0
b-14	C-19	17.42	0.88	0	70.5	0	70.5
b-9	C-19	16.94	1.02	0	65.5	0	65.5

b-6	426	13.09	1.11	0	98.0	0	98.0
b-8	C-19	11.87	0.88	0	128.5	0	128.5
b-4	426	10.77	1.28	0	116.0	7.0	123.0
b-1	426	10.24	1.28	0	114.5	19.0	133.5
b-0	426	8.71	1.28	0	151.5	19.0	170.5
b-2	426	8.62	1.28	0	148.0	25.0	173.0
b-3	426	8.47	1.11	0	188.0	0	188.0
b-5	426	7.41	1.28	0	207.0	7.0	214.0
b-7	C-09	7.01	0.88	0	254.5	0	254.5

Stop Data

Stop ID	Capacity[bus/h]	Queue Length		
P-147	8.0	Max: 0	Mean: 0.0	Std: 0.0
P-258	14.0	Max: 0	Mean: 0.0	Std: 0.0

Stop ID	Passenger Waiting Time[s]			Passengers At Stop		
P-147	Max: 3108.5	Mean: 1072.67	Std: 1110.62	Max: 39	Mean: 26.22	Std: 12.63
P-258	Max: 1781.0	Mean: 263.79	Std: 262.26	Max: 32	Mean: 7.33	Std: 8.05

Arterial Road Data

Total Travel Time[min]	Comercial Speed[km/h]	Signal Delay[s]	Bus Stop Delay[s]
45.38	14.67	77.0	1688.0