

Gradu Amaierako Lana / Trabajo Fin de Grado
Ingenieritza Kimikoko Gradua / Grado en Ingeniería Química

Simulación de Eventos Discretos (DES) como método para la optimización de recursos en un laboratorio de evaluación de neumáticos

Egilea/Autor/a:
Andoitz Campo Pérez
Zuzendaria/Director/a:
Asier Aranzabal Maiztegi

Índice

1. INTRODUCCIÓN	4
1.1. EL LABORATORIO DE CALIDAD DEL PRODUCTO	7
1.1.1. Ensayos <i>Endurance</i>	9
1.1.2. Ensayos Rolling Resistance	10
1.1.3. Análisis dimensional	11
1.2. OBJETIVOS	12
2. MARCO TEÓRICO	13
2.1. SIMULACIÓN DE EVENTOS DISCRETOS	13
2.1.1. Terminología y funcionamiento de una DES	14
2.1.2. Exposición de un caso ficticio usando una DES	16
2.1.3. Etapas de desarrollo de una DES	21
2.2. TEORÍA DE LA PROBABILIDAD Y DES	26
2.2.1. Variables aleatorias o random	26
2.2.2. Distribuciones de probabilidad	26
2.2.3. Test de hipótesis	29
3. DESARROLLO	30
3.1. CONCEPTUALIZACIÓN DEL MODELO	30
3.2. RECOPIACIÓN DE DATOS	32
3.3. TRADUCCIÓN DEL MODELO	34

3.4. VERIFICACIÓN Y VALIDACIÓN	35
4. RESULTADOS	35
5. CONCLUSIONES	44
6. GLOSARIO	46
A. ANEXO A: CÓDIGO FUENTE	49
B. ANEXO B: DIAGRAMAS DE FLUJO	61
C. ANEXO C: TABLAS	69
D. ANEXO D: ESPECIFICACIONES DE MAQUINA	70

1. INTRODUCCIÓN

El sector de la automoción, engloba una gran variedad de industria y servicios dedicados a servirla. Se estima que la aportación económica total de las actividades relacionadas, en este sector, asciende a un 11 % del PIB, lo que la convierte en la industria manufacturera que más ingresos aporta después del 18,8 % del PIB que posee la industria agroalimentaria española (Díaz y Garriga, 2021).

En este sector, el vehículo de propulsión autónoma, es generalmente utilizado tanto por los servicios de transporte a pasajero, como por los servicios logísticos dedicados al transporte de mercancías. Entre los componentes que forman el vehículo autónomo, se encuentran las cubiertas o neumáticos, los cuales se comportan como enlace entre el vehículo y el pavimento. Este nexo permite una transmisión eficiente de la energía producida por el motor de combustión interna, y a su vez, sus propiedades elásticas atenúan las irregularidades de la vía.

La industria manufacturera de cubiertas, ha mantenido un incremento sostenido en su producción en los últimos años, como puede observarse en la Figura 1. El mercado de neumáticos es liderado por dos empresas: Bridgestone y Michelin (Rodgers, 2020). Las demás empresas compiten entre ellas a una magnitud inferior a los líderes del sector, como se aprecia en la Figura 2.

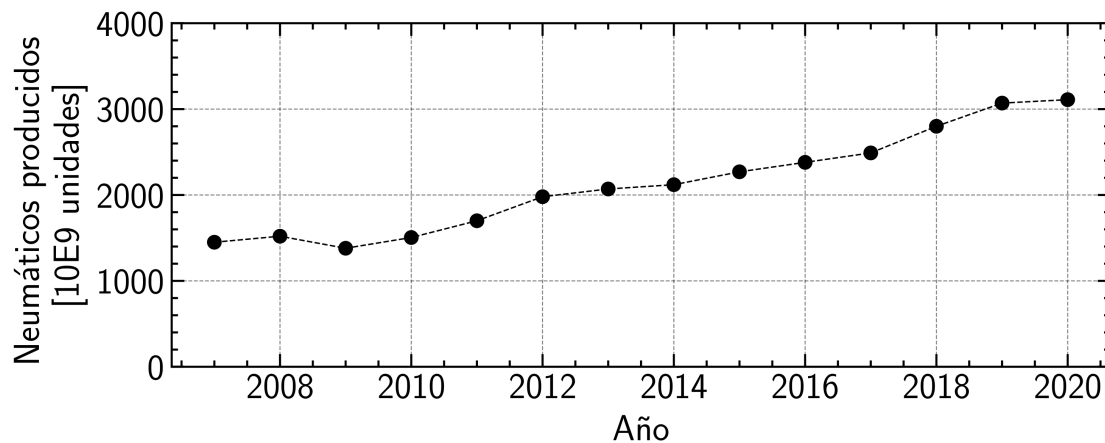


Figura 1: Evolución de la producción mundial de cubiertas (Rodgers, 2020).

La elevada competitividad característica del mercado, sumada a la emergente comercialización de producto asiático, incentiva a las establecidas multinacionales, a tomar un enfoque innovativo, con la intención de mantener su liderazgo (Chicu *et*

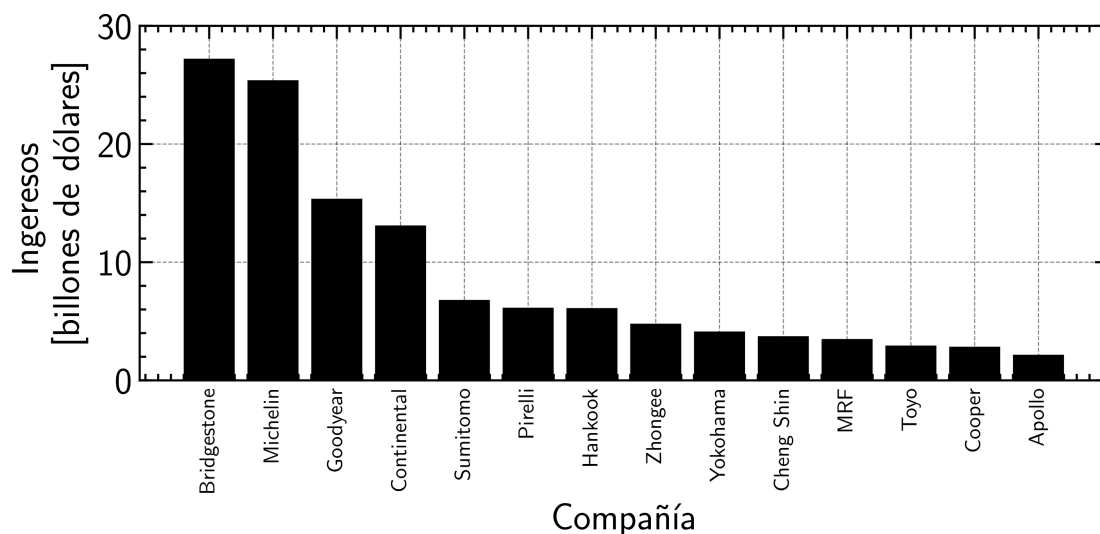


Figura 2: Ingresos de los fabricantes de neumáticos más relevantes en 2019 (Rodgers, 2020).

al., 2020). El esfuerzo invertido en innovación, por una parte, intenta alcanzar sus objetivos mediante el rediseño y la mejora del producto. Mientras que por la otra, trata de optimizar sus procesos y reducir desperdicios, mediante la automatización y el uso inteligente de los recursos disponibles.

A la hora de implementar mejoras, ya sea a un producto o a un proceso, la monitorización de los resultados se vuelve esencial. El producto o proceso experimental debe satisfacer las expectativas del cliente, y a su vez, cumplimentar la legislación vigente, como los estándares de calidad y medio ambiente. La monitorización de estas mejoras implica un coste elevado, a nivel temporal y económico, ya que estas deben ser puestas a prueba. En el caso del producto, su industrialización, supone reservar recursos que podrían ser destinados a la producción regular. Desde la materia prima utilizada en la elaboración de estos productos experimentales, a través de cada máquina ocupada para transformarlo, hasta llegar al laboratorio de calidad, donde se realizan ensayos para otorgar feedback a la empresa.

En las empresas de neumáticos, el laboratorio de Calidad del Producto (LCP), es el encargado de llevar a cabo los ensayos relevantes para asegurar la conformidad de la producción. A diario se cerciora de que los neumáticos manufacturados dentro de la planta cumpla con los estándares de calidad definidos por la compañía. El laboratorio logra asegurar la calidad del producto, mediante un control estadístico de las muestras elegidas al azar por cada gamma de producto. Parale-

lamente, una fracción de los recursos disponibles, es demandada por los proyectos de industrialización. Obligando al departamento a equilibrar ambas necesidades.

En un entorno con recursos limitados, donde la demanda no cesa de incrementar, es cuestión de tiempo que los recursos se agoten y la capacidad del laboratorio se vea sobrepasada. Para hacer frente a esta situación, y poder mantener el esencial ritmo marcado por los estándares de control de calidad e industrialización, debe realizarse un escalado de los recursos del LCP. A fin de que la expansión, de este sistema complejo, se desarrolle de la manera óptima, dirigir un estudio del impacto sobre las posibles inversiones resulta conveniente. Debido a que experimentar con el sistema real, y desarrollar una solución analítica no es factible, se ha propuesto el uso de la simulación. Dentro de los tipos de simulación, existen varias alternativas.

- Simulación Monte Carlo: Este tipo de simulaciones toma el nombre del famoso casino ubicado en Mónaco. Este tipo de simulación se caracteriza por desencadenar eventos al azar con el fin de obtener los resultados esperados. Se emplea con el fin de aproximar expresiones matemáticas complejas. Owen y Glynn (2016) describen la simulación de Monte Carlo, como un método para aproximar integrales mediante muestras promedio de valores enteros. Este método se puede usar, por ejemplo, para determinar la probabilidad de que un dado obtenga como resultado al menos un 5 en 4 de cada diez tiradas. El método Monte Carlo es apropiado para sistemas estáticos (Lawson y Leemis, 2008), es decir, sistemas que no tienen en cuenta el paso del tiempo.
- Sistemas Dinámicos (SD): La metodología de los SD trata de capturar el comportamiento de un sistema mediante su representación en un diagrama de flujo (Sweetser, 1999). Esta metodología, toma un enfoque cualitativo, apto para gerentes de empresa y organizaciones gubernamentales. Los SD están adaptados para sistemas continuos, siendo poco prácticos en un ambiente manufacturero, donde interrupciones como cambios de turno, Reparaciones de máquina, descansos de operario ... suceden a menudo.
- Simulación de Eventos Discretos (DES): Las DES toman el enfoque de representar sistemas reales mediante la ejecución y concatenación de procesos en un entorno virtual. Esta representación computacional se asemeja al sistema real. La DES ha demostrado su capacidad de resolver problemas de optimización en sistemas estocásticos. Como menciona el Allen (2011), su versatilidad ha sido demostrada en numerosos ámbitos, como aplicaciones militares, sistemas sanitarios, problemas logísticos y optimización de procesos de manufacturación.

1.1. EL LABORATORIO DE CALIDAD DEL PRODUCTO

A lo largo del proceso de fabricación de una cubierta, son varios los estándares de calidad que debe cumplir el producto para que pueda ser vendido en los distintos mercados a nivel global. Desde la composición química de las materias primas usadas, hasta las propiedades físicas del producto finalizado, la certificación de que el producto se halla dentro de los límites especificados, asegura un producto de calidad.

En el proceso final de control de calidad que se lleva a cabo en un laboratorio de evaluación del producto, estos son los ensayos principales.

- Ensayos *Endurance*
- Ensayos *Rolling Resistance*
- Ensayos dimensionales

En un LCP, se debe satisfacer la demanda de ensayos de los siguientes procesos:

Por una parte, los procesos de Conformidad de la Producción (CP), encargados de aseguramiento de la calidad del producto destinado a la venta a clientes. Por otra parte, los procesos para el desarrollo de nuevos productos, propios de las actividades de industrialización (IND), encargados de desarrollar nuevas líneas del producto, y optimizar los procesos de producción mediante cambios controlados que suceden en grupos reducidos. Finalmente, existen encargos que no encajan con las 2 anteriores categorías, (EXT). Para este caso, en el LCP, se ha estimado que, estas 3 fuentes de muestras de ensayo suman la cantidad descrita en la Tabla 1. La cual está cerca, de los límites de capacidad teóricos para un laboratorio con el equipamiento descrito en las Tablas 6 y 7. Las perspectivas de futuro, prevén un incremento continuado de la actividad, por lo que es habitual que el LCP afronte demandas superiores a su capacidad.

El flujo de los procesos llevados a cabo en el laboratorio, se representa en la Figura 3.

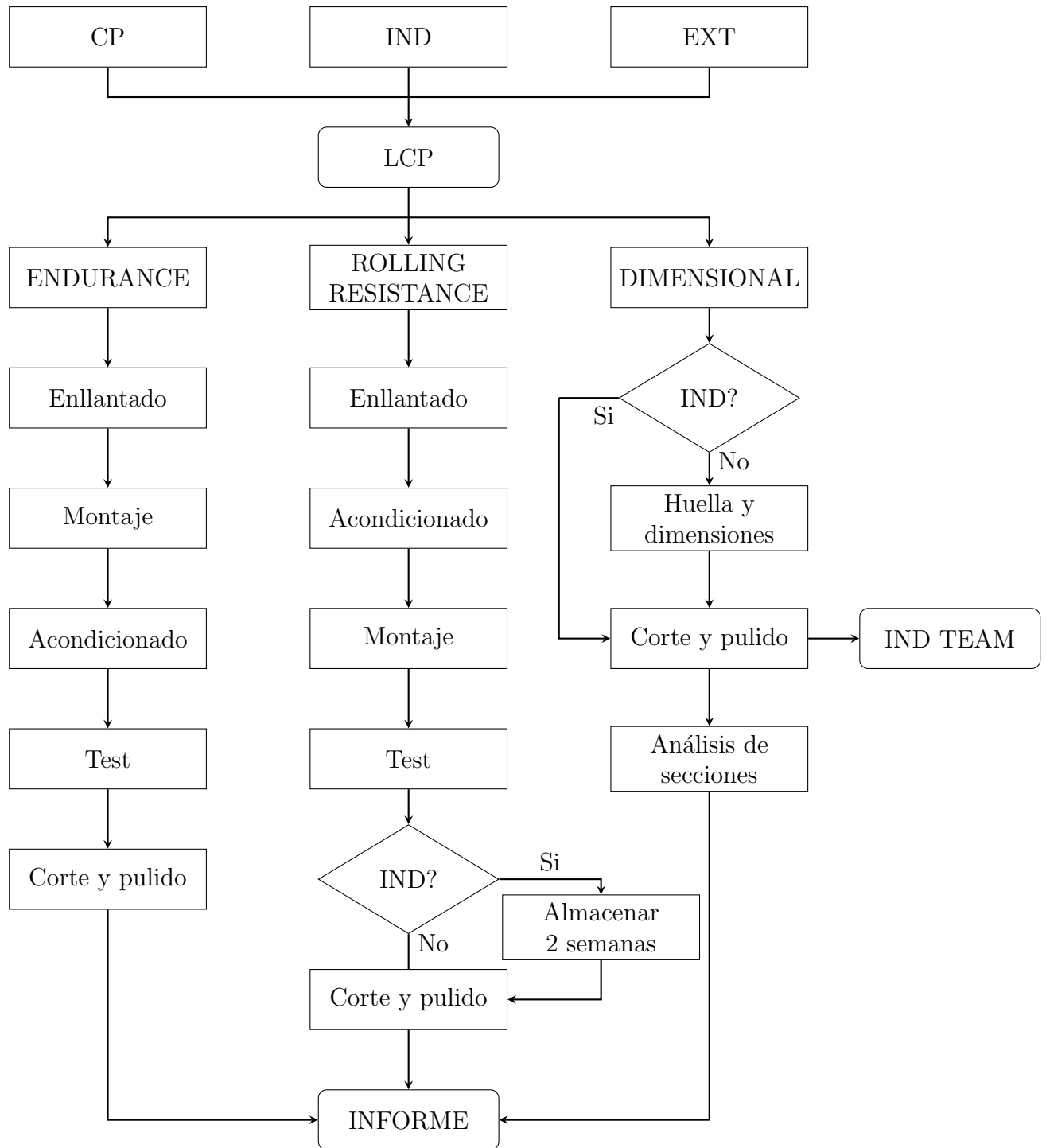


Figura 3: Diagrama de flujo de los procesos llevados a cabo en el LCP.

Tabla 1: Suma de la demanda típica de ensayos en un LCP.

Ensayo	Demanda (unds.)
Endurance	280
Rolling	760
Dimensional	140

1.1.1. Ensayos *Endurance*

El ensayo *Endurance* esta destinado a medir la fatiga que pueden llegar a soportar los materiales internos de la cubierta. Los materiales internos de una cubierta son señalados en la Figura 4, como *Steel Belt*, los cuales son capas de hilos de acero entrecruzados que aportan resistencia a la zona más expuesta de la cubierta. Las altas temperaturas y cargas a las que son sometidas las cubiertas producen una separación entre las capas de acero internas. Esta separación crea una deformación en la superficie de la cubierta. Cuando una deformación se detecta el ensayo concluye. Una vez finalizado, un operario corta y pule 2 secciones, una de ellas se toma en el origen de la separación de las capas interiores, la restante se toma a 180° del origen de la deformación.

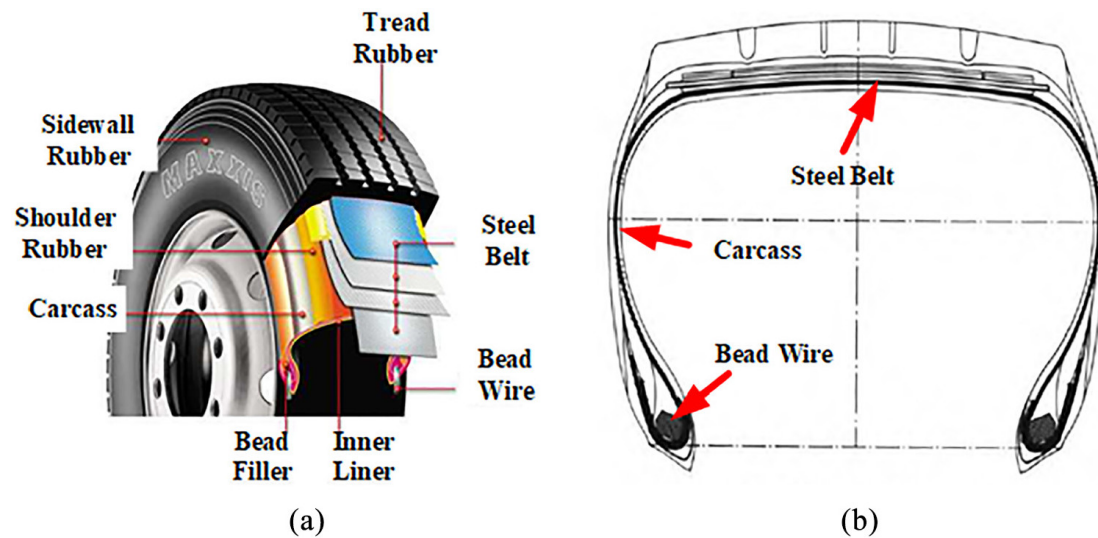


Figura 4: Diagrama de la composición de una cubierta

Los ensayos son llevados a cabo en la máquina expuesta en la Figura 5. La máqui-

Tabla 2: Pasos de un ensayo *Endurance*.

Paso	Carga (%)	Duración (h)
1	66	7
2	82	16
3	100	24
4-12	+10	6

na se encuentra en una sala aislada acondicionada a 40°C . La carga a la que son sometidas las cubiertas varía durante el ensayo en incrementos comúnmente llamados pasos. En la Tabla 2 se exponen detalles de cada paso. La carga a la que es sometida cada cubierta viene determinada por el índice de carga (LI) grabado en el costado de la cubierta, a su vez, la velocidad viene determinada por el fabricante. El LI se muestra en el Anexo C.



Figura 5: Máquina utilizada en los ensayos *Endurance*. Fabricante: RPM. Modelo: TJR2TBY. Especificaciones Técnicas: Anexo D

1.1.2. Ensayos Rolling Resistance

El objetivo de este ensayo es determinar el consumo energético de la cubierta. El consumo energético está directamente correlacionado a la fuerza de fricción registrada durante el ensayo. Según Ydrefors *et al.* (2021), las variables mas significativas en el valor de *Rolling Resistance* son la presión de inflado y la temperatura

de la cubierta.

Para medir el valor de *Rolling Resistance* se usa la máquina que aparece en la Figura 6. Debido a la sensibilidad del ensayo a la presión del neumático y la temperatura, La sala donde se encuentra la maquina tiene que estar acondicionada a 25 °C, y la presión de la cubierta debe ser ajustada con una tolerancia de 0.01 bar. Para asegurar estas condiciones, las cubiertas deben permanecer al menos 6 horas dentro de la sala acondicionada antes del comienzo del ensayo. La presión del neumático se ajusta al valor marcado en su costado justo antes de comenzar el ensayo. Finalmente, una vez comenzado el ensayo, se deja transcurrir el tiempo suficiente para que la cubierta alcance el estado estable. Una vez alcanzado el estado estable, la máquina toma las mediciones sobre la fricción y la deflexión de la cubierta. Al igual que en el ensayo *Endurance*, la carga viene determinada por el LI, y la velocidad por el fabricante.



Figura 6: Máquina utilizada en los ensayos *Rolling Resistance*. Fabricante:RPM. Modelo: TJRRRTBY. Especificaciones Técnicas: Anexo D.

1.1.3. Análisis dimensional

El objetivo de este ensayo es verificar que tanto las dimensiones generales de la cubierta, como sus componentes internos quedan dentro de la tolerancia de las especificaciones.

El ensayo se divide en 2 fases. En la primera fase, El desarrollo de la cubierta es

medido a la presión de inflado especificada en el costado de la cubierta, y a un bar. Mientras la cubierta conserva su presión de inflado especificada, se toman los estampados de la huella del neumático introduciendo un folio A3 entre la superficie tintada del neumático, y la prensa. En la segunda fase, se cortan y se pulen 3 secciones de la cubierta, para posteriormente ser digitalizadas en un escáner de alta resolución. Las mediciones a las imágenes obtenidas se realizan mediante un programa.

1.2. OBJETIVOS

El objetivo de este trabajo es optimizar los recursos del típico laboratorio de evaluación de neumáticos, en el caso de un hipotético crecimiento de la demanda de ensayos. Esto se consigue mediante el desarrollo de un método basado en una DES.

Para lograr el objetivo final se deberán completar los siguientes subobjetivos:

- Modelar los procesos del LCP de acuerdo a los fundamentos de una DES, obteniendo un modelo ajustado a la realidad.
- Definir las variables independientes del proceso, que posteriormente serán usadas en la simulación.
- Estimar el tiempo de ciclo de cada proceso, a través de la asignación de distribuciones ajustadas a cada subproceso.
- Definir las variables dependientes que otorgara el sistema a la salida.
- Desarrollar un programa de simulación en el entorno de Python mediante el uso de la librería Simpy. Dicha simulación, será capaz de emular los distintos escenarios propuestos durante el desarrollo.
- Proponer una alternativa de la distribución de los recursos descritos al comienzo del trabajo, para asegurar la capacidad del LCP cuando en el caso de que aumente la demanda de ensayos.

2. MARCO TEÓRICO

2.1. SIMULACIÓN DE EVENTOS DISCRETOS

Con el avance tecnológico experimentado desde los inicios de la computación, nuevas maneras para afrontar la resolución de problemas han surgido, que un siglo antes, habrían sido descartadas por su falta de viabilidad. La capacidad de cálculo y el diseño de inteligencias artificiales, ha proporcionado las herramientas necesarias para solucionar aquellos problemas, que con métodos clásicos, definitivamente demasiado complejos para modelar. La simulación, hoy en día, es una de las tres metodologías consolidadas, en el ámbito científico e ingenieril, para la resolución de problemas (Banks, 1998). Siendo esta metodología descrita como “la técnica del último recurso”, por Garzia *et al.* (1986), debido a la intensa demanda computacional que requería en el momento de su publicación, el transcurso de los años ha mermado esta desventaja.

Diversos autores describen la simulación de la siguiente manera

- “Una simulación es el establecimiento de un modelo lógico-matemático de un sistema y la manipulación experimental de este en una computadora digital (Pritsker, 1974)”.
- “La simulación es el proceso de diseñar el modelo de un sistema real y realizar experimentos con este modelo con el propósito de, entender el comportamiento del sistema, o evaluar distintas estrategias (dentro de los límites impuestos por un criterio o conjunto de criterios) para la operación del sistema” (Shannon y Johannes, 1976).
- “Una simulación es la imitación del modo de operación de un proceso real o sistema durante el transcurso del tiempo” (Banks, 1999).

Como se puede observar, el modelo, es un término recurrente en el ámbito de la simulación. Banks (1998) sostiene que el modelo es la representación del sistema simulado. El autor matiza que la virtud de un modelo radica en su complejidad, siendo necesario el balance entre una representación ajustada, sin complicar en exceso el modelo. Por tanto, aquellos factores que no influyan lo suficiente en los resultados de la simulación, deberían ser eliminados, ya que únicamente extienden el proceso de desarrollo.

El modelo, en consecuencia, constituye la base de la simulación, define las variables, y criterios para las decisiones tomadas en la simulación. Banks (1998) señala, la distinción entre un modelo matemático convencional y el modelo de una simulación. Los modelos matemáticos y estadísticos suelen representar las variables de manera explícita, relacionando variables independientes y variables dependientes para obtener un resultado. En el caso de una DES, el modelo utilizado se enfoca en la representación de sus componentes internos y sus interacciones.

El funcionamiento de la DES, se basa en registrar los cambios que van ocurriendo en el estado del sistema durante el transcurso del tiempo. En el intervalo entre alternaciones de estado, ocurren los eventos, propios de esta rama de simulación. Varga (2001) describe estos eventos como instantáneos, siendo la duración de los mismos nula. Durante el evento no ocurre nada especial, salvo la alteración del estado del sistema. Una sencilla representación de este concepto puede ser observada en la Figura 7, donde el sistema representado es un coche. El coche posee 3 estados, los cuales se van sucediendo, durante el transcurso del tiempo.

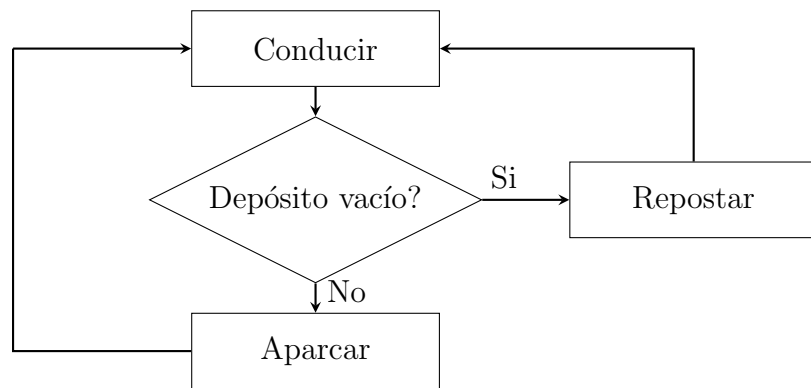


Figura 7: Representación del cambio de estado en una DES, los rectángulos representan el estado del sistema, mientras que las flechas indican el suceso de un evento instantáneo.

2.1.1. Terminología y funcionamiento de una DES

Para describir detalladamente este tipo de simulaciones, es necesario definir previamente algunos conceptos. Los siguientes conceptos son la síntesis de la información redactada por Banks (1998) aplicados en el entorno de Python y su librería Simpy.

Las variables del estado del sistema proporcionan información acerca de el comportamiento de los procesos que se llevan a cabo durante la simulación. Las variables de estado que el programa debe retornar depende del objetivo del proyecto. El programa deberá ser diseñado en torno a ellas, puesto que, serán usadas para el posterior cálculo de indicadores del rendimiento. Un ejemplo es el momento de inicio y fin de un subproceso.

Las entidades son los objetos representados por el modelo. Se distinguen dos tipos de entidades, las dinámicas y las estáticas. Las primeras avanzan en el sistema a través de los procesos simulados, mientras que las segundas, dan servicio a las primeras permaneciendo ocupadas en el proceso. Considerando una gasolinera como sistema, la entidad dinámica sería un coche repostando combustible, mientras que la estática, sería la estación de repostaje. Estos objetos pueden poseer atributos, como el consumo de combustible o el tamaño de su depósito. Dependiendo del objetivo de la simulación, algunos atributos serán relevantes en diseño del modelo.

Un recurso es una entidad que provee de servicio a una entidad dinámica. Los recursos pueden servir a múltiples entidades simultáneamente. De manera similar, una entidad puede solicitar varios tipos de recursos en distintas cantidades. Si la demanda de una entidad no se satisface, esta entrará en una cola, a la espera de la liberación del recurso. Cuando la entidad tome el recurso, lo mantendrá ocupado durante la duración del proceso. Los recursos pueden tener varios estados, como disponible, ocupado, averiado, en mantenimiento. . . Un ejemplo de recurso, puede ser el número de estaciones de autoservicio que posee una gasolinera.

Una actividad es un periodo de tiempo, el cual es conocido previo a su inicio. Su duración puede ser determinada de varias maneras; mediante una constante, una distribución estadística, el resultado de una ecuación, un archivo o una decisión lógica. El conocimiento de su duración, supone saber cuando un recurso sera liberado, lo que facilita el trabajo de computación. Por ejemplo, un coche rellenando el depósito de combustible, se puede modelar mediante una ecuación que relacione el caudal de la estación de servicio con el tamaño del depósito del coche. Este modelo devuelve la duración de la actividad, es decir, el tiempo que tarda el coche en repostar.

Una demora es un periodo de espera indefinido antes del comienzo de una actividad. Este depende de la ejecución del resto de actividades prioritarias, por lo que el final de la espera no se puede determinar mientras el recurso siga ocupado. Relacionado con el ejemplo anterior, una demora, puede representarse como la duración de la espera del coche que esta a la cola de repostaje.

Un evento ocurre al principio y al final de una actividad o demora. Marcan la transición del sistema de un estado a otro. Véase un ejemplo en la Figura 7.

Considerando todos estos conceptos, las entidades del sistema compiten entre ellas por el uso de los recursos, si estos no se encuentran disponibles, quedan a la espera de poder solicitarlos. Tanto las actividades como las demoras reclaman estas entidades durante los intervalos entre eventos, siendo liberadas para el siguiente proceso una vez finalizado el periodo. Una DES, puede describirse como el avance de una entidad o conjunto de entidades a través de las actividades que modelan el proceso real. Este avance es accionado mediante el transcurso de tiempo virtual dentro de la simulación. Véase un ejemplo en la Figura 7.

2.1.2. Exposición de un caso ficticio usando una DES

Con el objetivo de ilustrar los conceptos descritos en la sección 2.1.1 y demostrar la capacidad de resolución de problemas de la DES, se ha procedido a emular un caso práctico.

Supóngase que un ayuntamiento ficticio desea disminuir los problemas causados por el congestionamiento del tráfico en la ciudad. El ayuntamiento solicita a una consultoría ingenieril desarrollar un proyecto para observar el desempeño de las distintas soluciones planteadas. La consultoría ve apropiado usar una DES para valorar el impacto de las distintas propuestas y proponer un plan de acción óptimo a la entidad contratante.

El objetivo principal del ayuntamiento es minimizar la cantidad de vehículos que circulan por la ciudad. Una de las causas de congestión apunta a una estación de repostaje con unos precios particularmente competitivos que causa complicaciones a la entrada de la ciudad. Se cree que su capacidad de abastecimiento no es suficiente durante los picos de alta actividad. Por otra parte, la ciudad es un potente centro de actividad comercial. Muchos empleados se desplazan a diario en coche, lo que sumado a los espacios limitados de estacionamiento, y la alta densidad poblacional, hace sospechar que los conductores se demoran en exceso a

Tabla 3: Especificaciones generales.

Nombre	Valor
Duración	960 min
Ejecuciones	20 uds.
Capacidad del <i>parking</i>	100 uds.
Estaciones de repostaje	4 uds.
Probabilidad de repostaje	50 %

la hora de buscar aparcamiento. Se teme que este tiempo adicional, sea un factor considerable en las emisiones totales de los vehículos.

Con las siguientes dos medidas se espera obtener una reducción del congestionamiento del tráfico en la ciudad.

- Primeramente, negociar un ampliamento del servicio de distribución de combustibles con la empresa propietaria de la estación de repostaje.
- Paralelamente, ampliar la capacidad de un *parking* público gratuito ubicado cerca del núcleo de actividad laboral.

La simulación planteada, tratará de emular el comportamiento de los vehículos que circulan por la ciudad durante el transcurso de una jornada laboral normal. Es por ello que el comienzo de la simulación se estima en torno a las 7:00 a.m. y se prologa hasta 9:00 p.m. El modelo producirá 2 tipos de entidades, el vehículo de tipo ‘A’, que será el medio de transporte para los trabajadores de la zona, y el de tipo ‘B’ que englobará todo tipo de usuarios ajenos al ámbito laboral. Esta distinción se ha considerado necesaria, debido a la uniformidad que representa el primer grupo, respecto al segundo, algo más caótico. Los vehículos generados seguirán la rutina descrita en la Figura 8.

La consultoría, por una parte, establece las propiedades generales de su modelo en la Tabla 3. Mientras que, por otra parte, decide recopilar información clave para el desarrollo del modelo de simulación. A raíz de un análisis obtiene los datos mostrados en la Tabla 4.

Como puede observarse en la Figura 9, la adición de 2 estaciones de autoservicio en la gasolinera evita su colapso en intervalos críticos. En el gráfico situado a la izquierda, el pico de actividad azul en torno al minuto 150 de la ejecución de la simulación, se ve drásticamente reducido, mientras que el repunte secundario es

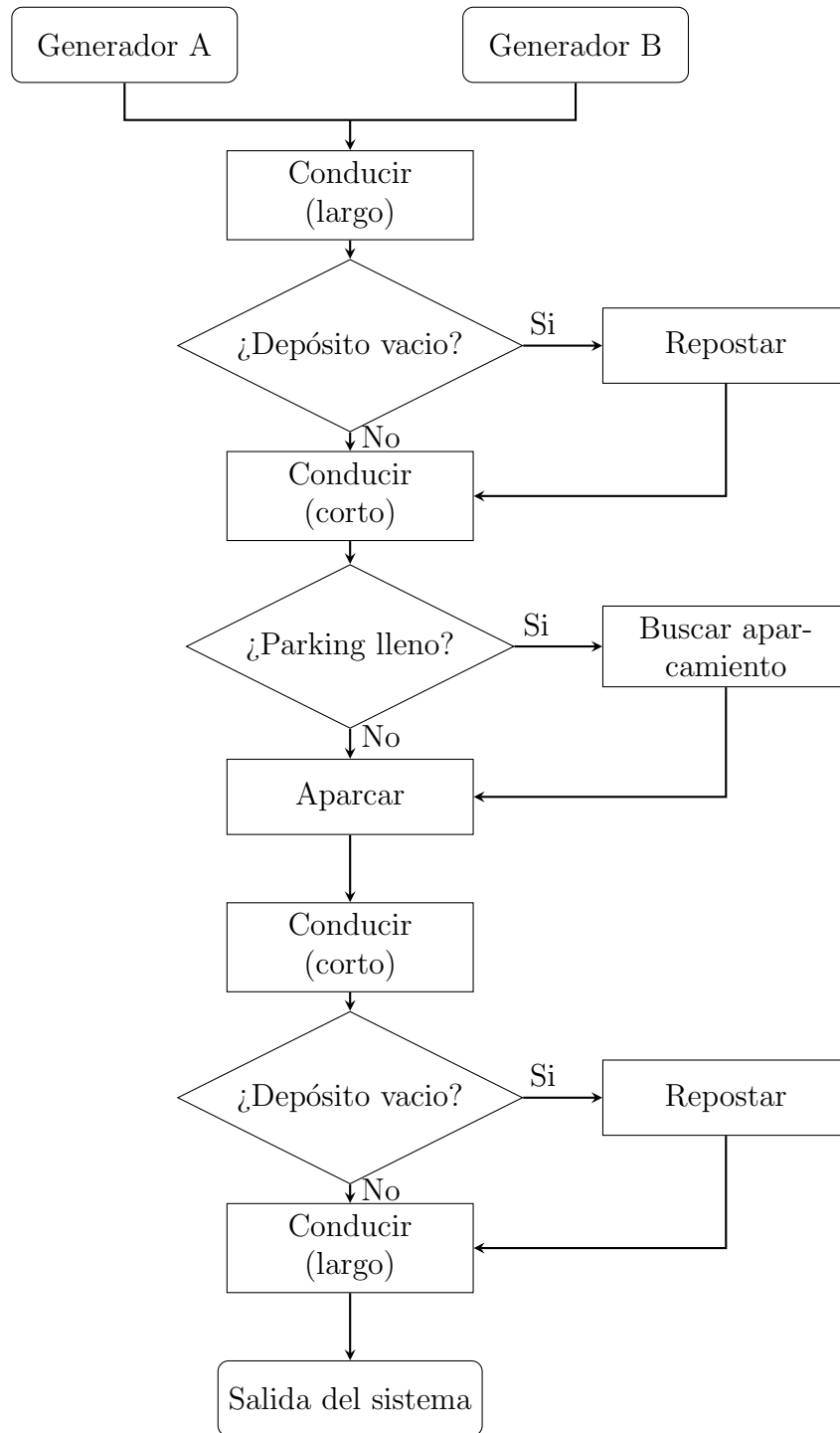


Figura 8: Modelo del caso práctico descrito en el apartado 2.1.2

Tabla 4: Propiedades de las distribuciones observadas en los procesos reales.

Variable	Tipo de vehículo	Distribución	Constantes (min)
Conducir largo	A y B	Gamma	$k = 8, \theta = 3$
Conducir corto	A y B	Triangular	$min = 2, moda = 3, max = 4$
Buscar <i>parking</i>	A y B	Gamma	$k = 10, \theta = 1$
Repostar	A y B	Triangular	$min = 4, moda = 5, max = 8$
Entre llegadas	A*	Exponencial	$\lambda = 1$
	B	Exponencial	$\lambda = 15$
Estacionamiento	A	Normal	$\mu = 560, \sigma = 60$
	B	Lognormal	$\mu = 5, \sigma = 0,7$

*La generación de entidades tipo ‘A’ únicamente se prolonga desde el comienzo de la simulación hasta el minuto 150. Así, se obtiene una aproximación, al horario típico de un día laboral.

completamente eliminado. A su vez, en el gráfico adyacente, La desviación estándar de tiempos de espera queda reducida y su media queda desplazada hacia el eje.

Continuando con el análisis, en la Figura 10, El efecto de incrementar en un 50 % las plazas de parking disponible, no es apreciable en el gráfico expuesto arriba. La cantidad de vehículos en carretera permanece similar, mientras que en el gráfico inferior los vehículos saturan el *parking* hasta su nuevo límite. Al no haber una clara correlación entre ambos indicadores, se ilustra la representación de la distribución de tiempos de conducción en la Figura 11. En esta exposición, puede llegar a apreciarse una ligera diferencia en los tiempos de conducción, aunque un test de hipótesis sería necesario para determinar si estas diferencias son estadísticamente significativas.

En conclusión, los resultados muestran una clara ruta de inversión a la hora de ampliar la estación de combustibles, mientras que la expansión del *parking* parece irrelevante. Esto demuestra la capacidad de recopilación de datos que ofrece una DES. Simultáneamente subraya la importancia del tratamiento de los datos obtenidos, y el uso de pruebas estadísticas, para la obtención de conclusiones certeras.

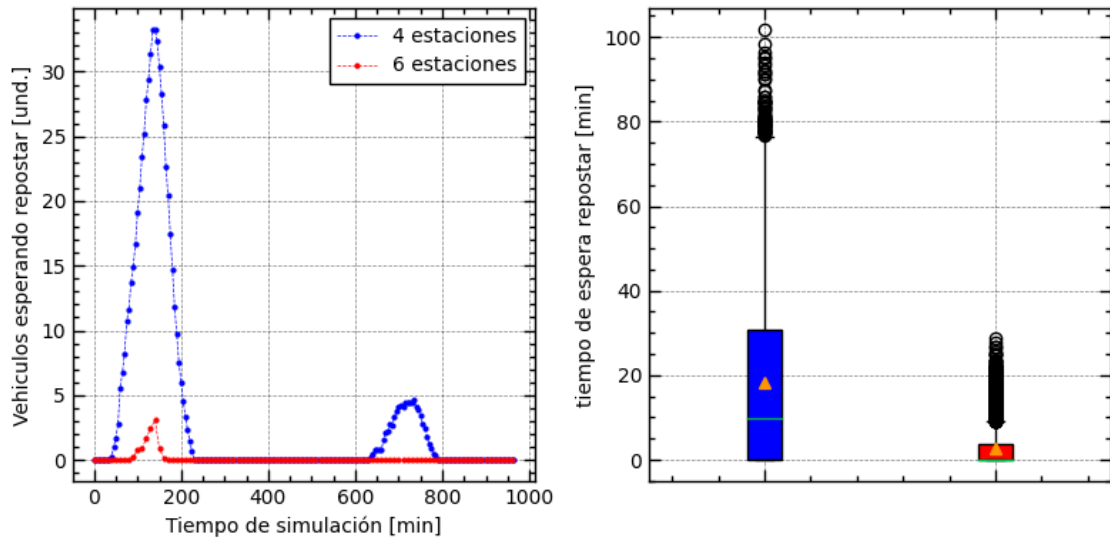


Figura 9: Izquierda: Evolución de los vehículos a la espera de repostar para distintos escenarios. Derecha: Distribución de los tiempos de espera de repostaje para los distintos escenarios

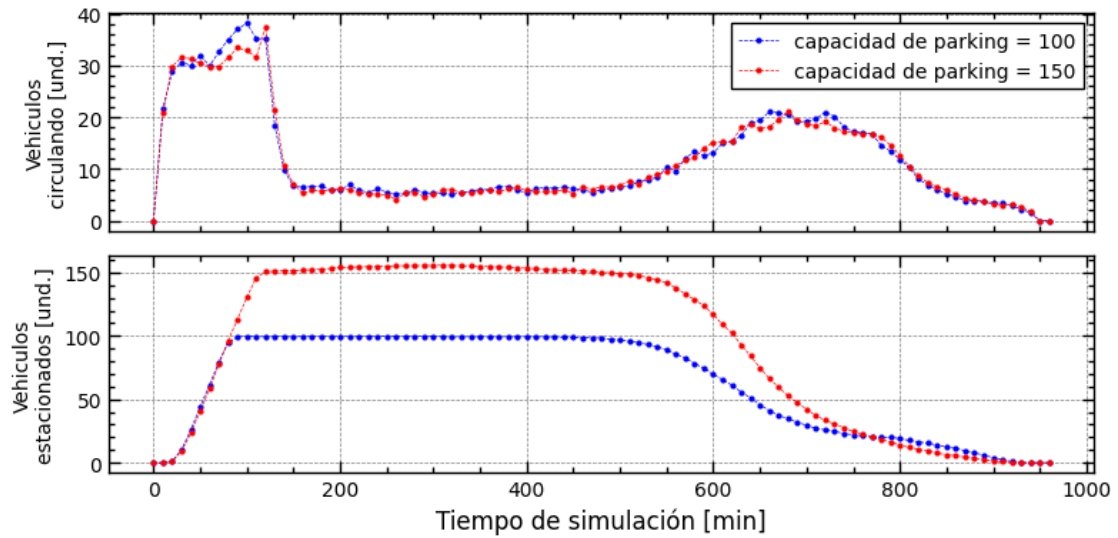


Figura 10: Evolución de la cantidad de vehículos a lo largo de la simulación, en distintos escenarios. Arriba: Vehículos en circulación. Abajo: Vehículos estacionados.

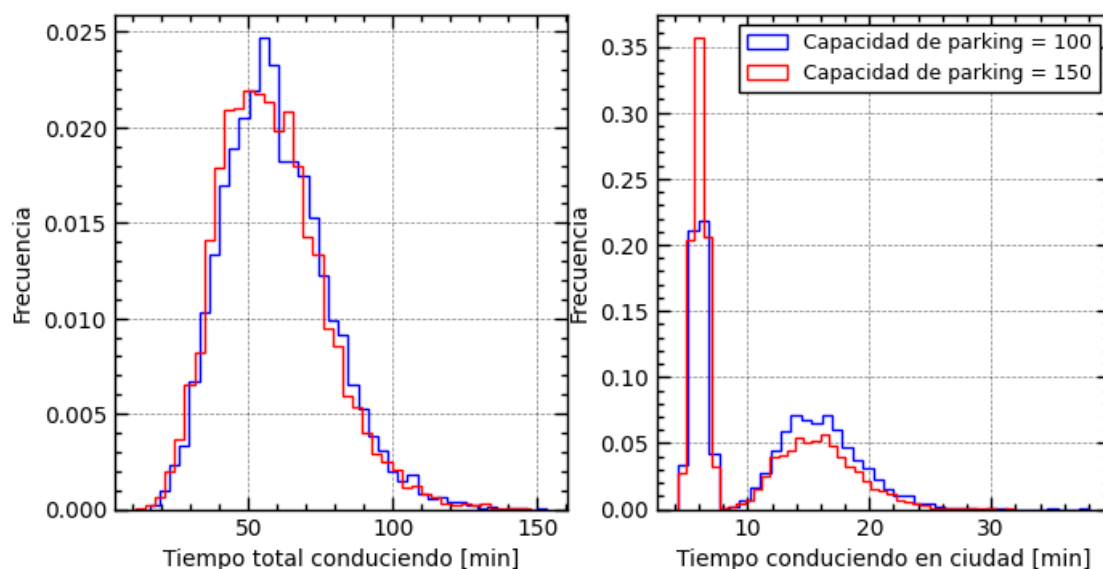


Figura 11: Distribución del tiempo de conducción para distintos escenarios. Izquierda: Tiempo total en circulación. Derecha: Tiempo circulando dentro de la ciudad.

2.1.3. Etapas de desarrollo de una DES

Para que un proyecto tan complejo como una DES prospere, es fundamental estructurarlo. El diagrama expuesto en la Figura 12 publicado por Banks (1998) define las etapas que deberían seguir este tipo de proyectos. Al ser este proyecto un simple Trabajo de Fin de Grado, este esquema servirá como orientación, más que como plano para la construcción de la simulación. Por ejemplo, la recopilación de datos ha resultado imposible en el momento de redacción del trabajo, pero se espera desarrollar el proyecto a partir de las más ajustadas distribuciones de probabilidad supuestas para el desarrollo del modelo. Se espera que este trabajo asiente una base, y que las imperfecciones actuales puedan ser corregidas en el futuro.

Formulación del problema, establecimiento de objetivos, y plan general.

Cada DES comienza con la declaración del problema que se tratará de solucionar. Esta etapa es crucial debido a que el resto del trabajo se asentará sobre esta base. Se debe asegurar que el problema que se está formulando sea el correcto, la comunicación entre el cliente y el equipo desarrollador debe ser óptima. Una

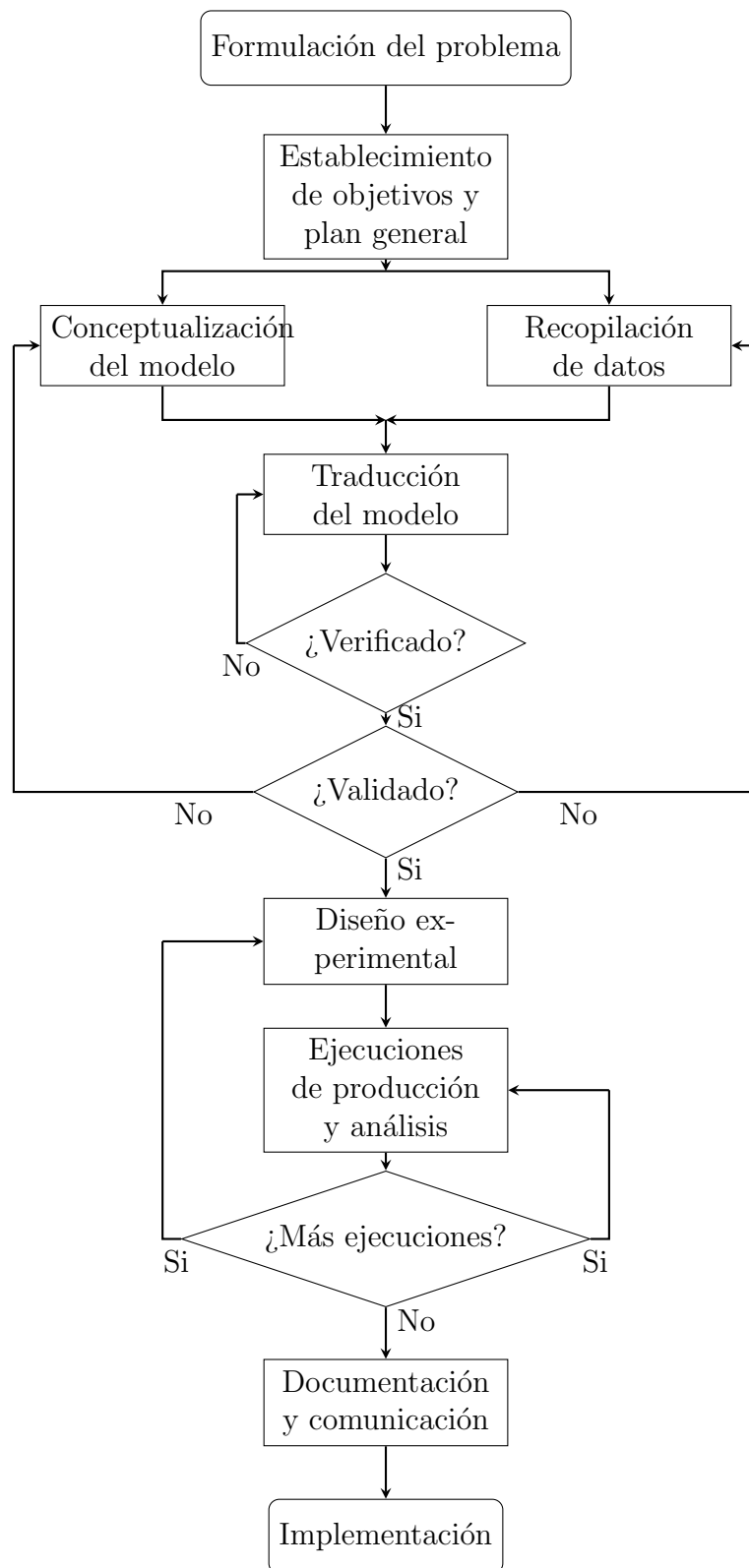


Figura 12: Esquema del proceso de desarrollo de una DES.

descripción precisa y un conocimiento extenso del sistema son necesarios para evitar trabajar en solucionar el problema equivocado. Es por ello que debe haber comunicación y *feedback* constante entre cliente y equipo.

En el caso de este trabajo, algo similar sucede. Al ser un único autor, la comunicación no será un factor importante, pero esto conllevará otras complicaciones. El autor debe, comprender profundamente el sistema a modelar, y a su vez, dominar el uso de las herramientas de simulación. Manejar ambas, requerirá recordar la formulación del problema inicial más frecuentemente para no desviarse de la dirección.

Respecto a los objetivos, estos deben ayudar a asentar la dirección de trabajo actuando como plan general. El establecimiento de varios objetivos alcanzables, cohesivos y correctamente secuenciados, guiará el proceso de desarrollo. Estos objetivos quedan listados en la sección 1.2.

Conceptualización del modelo. El sistema propuesto sugiere conceptualizar el modelo en orden inverso, es decir, comenzar identificando los objetivos del proyecto, tal y como se ve en la Figura 13.

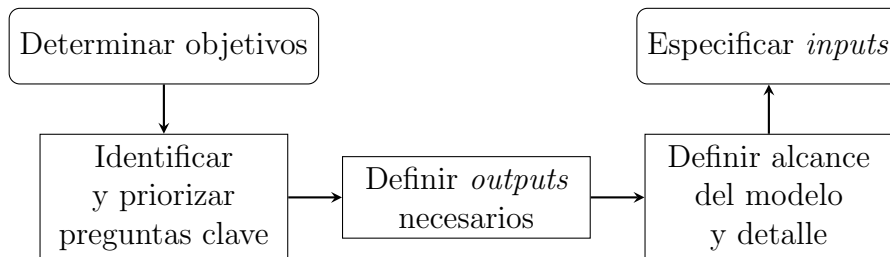


Figura 13: Diagrama de pasos a seguir en la conceptualización de un modelo.

El siguiente paso es hallar las preguntas clave a las que se les quiere dar respuesta. Generar una lista, y ordenar las preguntas respecto a su relevancia es lo ideal. Además, cuantificar los beneficios asociados a estas, aclara su jerarquía.

Después, se debe determinar que outputs son necesarios para contestar a las preguntas previamente formuladas. Introduciendo en la simulación únicamente la complejidad necesaria, agilizará el proceso de desarrollo y mantendrá el enfoque original.

Continuando con el alcance de la simulación, se deben detallar los procesos críticos en el cálculo del *output*, descartando o simplificando aquellos que no tengan gran

impacto en la salida. Por ejemplo, modelar recursos abundantes en la simulación, no tendrá impacto en el retorno, mientras que conocer los tiempos de espera de un proceso si lo hará.

Se Finaliza, especificando las variables independientes que se introducirán al sistema. Estas variables tienen que ser de alta calidad, ya que, de lo contrario su introducción puede perjudicar al sistema al completo. Un ejemplo sería el de distinguir entre distintas distribuciones para entidades con atributos pobremente caracterizados.

Recopilación de datos. Cualquier DES requiere datos. La recopilación de la información del sistema, o la estimación en caso de ser un sistema ficticio, es una necesidad.

El primer paso es recopilar la información de la que se dispone actualmente. En el caso de no poseer información, o de que los datos presenten sesgos, se deben medir las características necesarias. Finalmente, en caso de resultar imposible la medición, se deben tomar como referencia las opiniones de un experto, que normalmente suele ser el operario que maneja la máquina.

Traducción del modelo. El modelo de simulación se construye a partir de la información desarrollada en el anterior punto. Este proceso consiste en traducir de manera eficaz el esqueleto previamente desarrollado, a un lenguaje que el ordenador pueda procesar. Para ello, Banks (1998) sugiere lo siguiente:

- Enfoque en el problema. El proyecto no debería limitarse a desarrollar el modelo, sino que debería tratar de invertir aproximadamente el mismo tiempo en la exploración del sistema y la solución de problemas.
- Comienzo simple. El comienzo del proyecto debe ser simple, adquiriendo complejidad hasta alcanzar el detalle necesario.
- Manejo de la inteligencia del modelo. El modelo no debe superar el rendimiento de los procesos reales. El modelo desarrollado puede desarrollar ventajas respecto a su contrapartida real, como conocer la ubicación exacta de una pieza instantáneamente. Los humanos que gestionan el proceso difícilmente van a poder lograr este control, y supondrá una característica inalcanzable en el sistema real.

- **Revisión.** Revisar el funcionamiento del proyecto frecuentemente. Esto evitará desvíos respecto al objetivo principal y facilitará el proceso de verificación y validación.

Verificación y validación. Comprobar el correcto funcionamiento de la DES es fundamental. Esta es la labor de la verificación y validación (VV). Banks (1998) sugiere que este paso se realice frecuentemente a lo largo del proceso de desarrollo. Esto evita complicaciones una vez el proyecto se vuelve demasiado extenso y demasiado complejo. Realizar la VV en cada pieza funcional del modelo cada vez que suceda algún cambio es lo recomendable.

Por una parte, la verificación es el proceso encargado de comprobar que la simulación se ejecute de la manera esperada. En este proceso se intentará hallar errores en el programa. Esto se logra comprobando que la ejecución de cada paso en la simulación sea la correcta y retorne los resultados esperados.

Por otra parte, la validación consiste en comparar, el comportamiento del modelo respecto al comportamiento real del sistema. En el caso de poseer datos sobre el sistema simulado, el libro sugiere utilizar herramientas estadísticas como, el test t de Student o análisis de regresión donde el ajuste a la realidad será comprobado. En el caso de no poseer los datos del sistema, se optará por un enfoque cualitativo. Aunque no haya datos, el comportamiento de un sistema puede predecirse cualitativamente, al cambiar ciertas variables. Por ejemplo en un tanque de agua cilíndrico, aunque el volumen del tanque y el caudal de entrada del agua sea desconocido, el nivel del agua disminuirá si el flujo neto es negativo. Si el programa ejecutado obtiene el resultado contrario, este será invalidado comenzando la revisión del mismo.

Análisis. Mediante el análisis del retorno de la simulación, se obtienen las conclusiones a las preguntas planteadas al inicio del proyecto. Primeramente, se deberán plantear los distintos escenarios que serán simulados. Estos escenarios serán ejecutados en la simulación obteniendo sus variables de salida. El correcto tratamiento de los datos, junto con una exposición significativa de ellos dará claridad a los resultados. No obstante, para obtener las conclusiones deseadas, deberán realizarse test estadísticos a los resultados para certificarlos.

Documentación y comunicación. Con la intención de que este proyecto pueda ser continuado en el futuro por otra persona, la documentación se vuelve necesaria.

El proyecto se considera documentado, por una parte, mediante este trabajo, y por la otra mediante un código fuente autoexplicativo, con abundantes comentarios.

2.2. TEORÍA DE LA PROBABILIDAD Y DES

La probabilidad está estrechamente relacionada con la DES. Previamente a comenzar el proceso de formulación del modelo de una DES, los datos empíricos medidos en los procesos reales se traducen a un modelo reproducible computacionalmente. A este ejercicio se le denomina comúnmente como ajuste de distribución de probabilidades. Estas distribuciones de probabilidades son continuamente usadas durante la simulación, con el fin de representar la aleatoriedad de las actividades que se llevan a cabo. Cada vez que sucede un evento en la simulación, se toma una muestra de su distribución de probabilidad asociada. Esta muestra o variable aleatoria corresponde a la duración de dicha actividad. Concatenando la sucesión de eventos, se consigue replicar la aleatoriedad del sistema.

El uso de la teoría de la probabilidad no acaba aquí. Una vez finalizada la etapa de simulación, los datos obtenidos de la simulación son puestos a prueba. Para evitar obtener conclusiones erróneas, debido a un resultado excepcional al azar, se realizan múltiples ejecuciones de la simulación. Los resultados obtenidos son una población de las variables dependientes calculadas por la simulación. El método más común para determinar la diferencia entre los distintos escenarios simulados es el test de hipótesis.

2.2.1. Variables aleatorias o random

Según Meester (2008), una variable aleatoria (X) puede explicarse como un número cuyo valor no es conocido en el momento de su planificación. Aunque el valor último de esta variable sea desconocido, el rango de valores que pueden tomar estas variables viene determinado por una función. A esta función se le denomina Función de Distribución de Probabilidad (FDP).

2.2.2. Distribuciones de probabilidad

Una distribución de probabilidad es una función estadística que determina la probabilidad de que una variable aleatoria (Simon, 2002), obtenga los valores posibles

que contiene la propia distribución. Las FDP comúnmente usadas en los modelos de una DES son las siguientes:

Distribución uniforme. Esta distribución destaca debido a que los valores, dentro de un rango definido, tienen la misma probabilidad de ser escogidos. Suele ser usada para proporcionar peso a los distintos sucesos que forman una lista. Por ejemplo, para representar la probabilidad de que un coche tenga matrícula palíndroma, se puede emplear una distribución uniforme de $a = 0$ y $b = 9999$. Para cualquier número obtenido inferior a 904, la matrícula del coche se determina palíndroma.

$$P(x) = \frac{1}{b - a} \quad (1)$$

Distribución normal. Aparece continuamente en fenómenos naturales, sociales y psicológicos. Es la más común de las distribuciones y sus parámetros son sencillos e intuitivos. La condición que se debe cumplir para poder usar esta FDP, es que los datos que forman un conjunto deben ser independientes entre sí. Por ejemplo, la vida útil de un modelo de coche puede representarse mediante esta distribución.

$$P(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (2)$$

Distribución lognormal. Sucede cuando al aplicarle un logaritmo a una variable aleatoria, esta acaba normalmente distribuida. Si la variable aleatoria es e^X , X sigue una distribución normal. Esta FDP presenta una asimetría positiva, lo que la hace idónea para representar procesos en los que normalmente su duración se encuentra dentro de un intervalo previsto, pero que puede llegar a extenderse más de lo normal en el tiempo. Por ejemplo, la duración de la reparación de un coche puede ser modelada por esta FDP. Una reparación puede llegar a extenderse debido a imprevistos como la falta de piezas de recambio, o problemas inicialmente no detectados.

$$P(x) = \frac{1}{\sigma x \sqrt{2\pi}} e^{-\frac{(\ln(x)-\mu)^2}{2\sigma^2}} \quad (3)$$

Distribución triangular. Se caracteriza por estar limitada, entre un límite inferior a , y otro superior b , siendo su valor más probable la moda m . A menudo es usada cuando no se dispone de suficientes datos de un proceso. Por ejemplo, en un taller no se dispone de datos sobre lo que un mecánico se demora en cambiar una rueda a un coche, pero el propio mecánico, por experiencia, sabe aproximadamente el mínimo, el máximo, y la duración típica.

$$P(x; a, m, b) = \begin{cases} \frac{2(x-a)}{(b-a)(m-a)} & \text{para } a \leq x \leq m, \\ \frac{2(b-x)}{(b-a)(b-m)} & \text{para } m \leq x \leq b, \\ 0 & \text{para el resto} \end{cases} \quad (4)$$

Distribución exponencial. Es la distribución de la probabilidad del tiempo entre eventos de un proceso de Poisson. Un proceso de Poisson se caracteriza por que los eventos ocurren continuamente y de manera independiente. En las DES, se usa para determinar el tiempo entre llegadas de las entidades generadas. λ es el parámetro que marca la frecuencia de las llegadas. Por ejemplo, en un taller los clientes llegan uno detrás de otro, el periodo entre estas llegadas puede ser modelado mediante esta FDP.

$$P(x; \lambda) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0, \\ 0 & x < 0 \end{cases} \quad (5)$$

Distribución gamma. Es una distribución de probabilidad de 2 parámetros, constituyéndose por el parámetro de forma k , y la escala θ . Debido a sus parámetros, es adecuada para el ajuste de distribuciones cuando las variantes de distribuciones normales no se ajustan a el conjunto de datos. Permite representar una asimetría en la distribución.

$$P(x) = x^{k-1} \frac{e^{-x/\theta}}{\theta^k \Gamma(k)}, \text{ siendo: } \Gamma(k) = (k-1)! \quad (6)$$

2.2.3. Test de hipótesis

La comprobación de los resultados obtenidos en las DES, es una parte fundamental de la simulación. Para ello se suelen emplear los test de hipótesis como muestra de la validez de las conclusiones obtenidas.

El test de hipótesis involucra formular, 2 o más hipótesis, que serán puestas a prueba mediante el empleo de test estadísticos (Martin, 2022). Aplicado a el caso desarrollado en este trabajo, se trata de comprobar las hipótesis formuladas mediante la comparación de las poblaciones obtenidas en distintos escenarios.

Un formato común en el ámbito de test de hipótesis es el siguiente:

- Hipótesis nula, H_0 , trata de demostrar que el fenómeno observado es únicamente resultado del azar.
- Hipótesis alternativa, H_1 , trata de demostrar la hipótesis que se desea probar. Existen 3 tipos:
 - parámetro de población $>$ valor hipotetizado.
 - parámetro de población $<$ valor hipotetizado.
 - parámetro de población \neq valor hipotetizado.

Para realizar un test de hipótesis, primero debe escogerse un test, y aplicarlo a las muestras de la población a analizar. Después, la hipótesis nula es rechazada o no dependiendo del valor p obtenido.

Un test de hipótesis muy usado es el test t de Student (7).

$$t = \frac{\tilde{x} - \mu_0}{s/\sqrt{n}} \quad (7)$$

El valor p , es la probabilidad condicional de los valores a los extremos de un test estadístico, asumiendo que la hipótesis nula es cierta. Por convención, si la probabilidad de la hipótesis nula es $< 5\%$, o $< 0,05$, la hipótesis nula es rechazada. De lo contrario, no es posible rechazar la hipótesis nula, lo que no supone que esta sea cierta.

3. DESARROLLO

3.1. CONCEPTUALIZACIÓN DEL MODELO

Como se menciona en la sección 2.1.3, a la hora de diseñar el modelo es conveniente comenzar desde el planteamiento de los objetivos, e ir adquiriendo detalle hasta llegar a definir las variables de entrada.

Una vez han quedado los objetivos descritos, se ha procedido a determinar las cuestiones clave para dar con una solución. La siguiente lista de preguntas ha sido ha sido formulada:

- ¿Qué sucedería si el tiempo disponible para el montaje de las máquinas fuese mayor?
- ¿Cuál sería el efecto de que dos técnicos trabajasen simultáneamente en los ensayos?
- ¿Cómo afectaría la instalación de máquinas *Endurance* adicionales a la cantidad de ensayos realizados?

Para responder a las preguntas planteadas, y facilitar el proceso de VV. se ha programado la simulación de tal manera que devuelva los contenidos de la Tabla 5. Cada línea en la columna de variable corresponden a un encabezado de la tabla que retorna la simulación. Es decir, las variables listadas en la Tabla 5, son las variables de retorno de la simulación.

Tabla 5: Definición de las variables de salida de la simulación.

Propósito	Tipo	Variable
Identificativo		Ejecución ID Ensayo
Trazativo [min]	Generación	Entidad
	Inicio	Enllantado Montado Acondicionado Ajustado de presión Ensayo
	Fín	Enllantado Montado Acondicionado Ajustado de presión Ensayo
Cuantitativo [min]	Demora	Enllantado Montado Acondicionado Ajustado de presión Ensayo
	Duración	Enllantado Montado Acondicionado Ajustado de presión Ensayo

A través de análisis de los datos indexados, se han podido obtener estadísticas como: número de ensayos totales realizados por tipo de test, o el tiempo total de máquina en funcionamiento por cada ejecución de la simulación.

El modelo a reproducir para obtener los datos descritos en la Tabla 5, ha sido deliberadamente diseñado de manera simple. Esto ha facilitado la obtención de datos representativos y su posterior análisis. Se ha tomado parte del diagrama de la Figura 3, Excluyendo ciertos procesos irrelevantes para las preguntas previamente formuladas. Estos diagramas se encuentran en el Anexo B.

Las variables de entrada necesarias para el funcionamiento del modelo se han recopilado en las Tablas 6, 7, 8 y 9.

Tabla 6: Características generales de la simulación.

Propiedad	Valor
Duración	365 días
Jornada laboral	8 h
Turno(s)	1
Ejecuciones	100 unds.

Tabla 7: Recursos representados en el modelo.

Recurso	Cantidad (unds.)
Máquina <i>Endurance</i>	4
Máquina <i>Rolling Resistance</i>	1
Máquina enllantado	1
Personal técnico	1
Llantas <i>Endurance</i>	14
Llantas <i>Rolling Resistance</i>	20

3.2. RECOPIACIÓN DE DATOS

En un proyecto de un laboratorio real, primeramente se habría buscado información en las bases de datos y en los registros históricos disponibles. En caso de no disponer de estos, se habrían medido la duración de los procesos en el acto. Una de las propuestas es cronometrar cada proceso entre 20 y 30 veces para obtener un rango de tiempo estimado. A este rango, se le aplicaría un ajuste de distribución de probabilidad que se usaría en la DES. Al trabajar con un laboratorio ficticio, no es posible la recopilación de datos dentro de un laboratorio. No obstante, el proyecto se ha planteado basándose en estimaciones fundamentadas. Se ha tratado de representar cada proceso mediante distribuciones acordes a sus propiedades, descritas en la sección 2.2.2.

Para los tiempos entre llegadas de objetos en la simulación, se ha elegido la distribución exponencial debido a su asociación a tiempos de espera entre eventos.

Tabla 8: Características del proceso de ensayos endurance.

Proceso	Prioridad	Distribución (min)
Tiempo entre llegadas	-	Exponencial $\lambda = 120$
Enllantado	0	Triangular $a = 12, m = 15, b = 20$
Montaje	-1	Normal $\mu = 45, \sigma = 10$
Acondicionamiento	-3	Constante $c = 180$
Ajuste de presión	-4	Constante $c = 5$
Ensayo	-	Normal $\mu = 4320, \sigma = 360$

Tabla 9: Características del proceso de ensayos rolling.

Proceso	Prioridad	Distribución (min)
Tiempo entre llegadas	-	Exponencial (min) $\lambda = 48$
Enllantado	0	Triangular $a = 12, m = 15, b = 20$
Acondicionamiento	-2	Constante $c = 360$
Montaje	-2	Normal $\mu = 45, \sigma = 10$
Ensayo	-	Normal $\mu = 180, \sigma = 20$

Para los procesos de los que menos información se poseía, como puede ser los procesos de montaje y enlantado, se ha optado por representarlos mediante una distribución triangular. Como se indica en la sección 2.2.2, las distribuciones triangulares son adecuadas para aquellos procesos en los que no se disponga de mucha información.

Finalmente, en el caso de los ensayos *Endurance*, se ha estimado su duración media basándose en los estándares de calidad especificados en la norma UNECE-54. En el caso de los ensayos *Rolling Resistance* se ha estimado según la información recogida en el estándar SAE J1269. La duración de estos ensayos, se ha tratado de representar mediante distribuciones normales, debido a la sencillez y versatilidad de las mismas.

3.3. TRADUCCIÓN DEL MODELO

El modelo traducido, que se encuentra en el Anexo B, puede dividirse en 3 procesos generales y varios subprocesos:

La generación de entidades se encarga de crear instancias de las cubiertas que son alimentadas a los procesos maestros de cada ensayo. Son 2 los generadores que posee la simulación, uno para los ensayos *Endurance*, y otro para los ensayos *Rolling*.

Los procesos maestros son los responsables de ejecutar los subprocesos en el orden correspondiente mientras demandan los recursos necesarios durante su transcurso. Al igual que los generadores, hay 2 procesos maestros, uno para cada tipo de ensayo.

Los subprocesos son las subrutinas llamadas por los procesos maestros. Crear módulos para cada actividad mejora la legibilidad del código.

La obstrucción del técnico es el proceso que simula el inicio y el fin de una jornada laboral, y hace transcurrir el tiempo, incluyendo los fines de semana.

3.4. VERIFICACIÓN Y VALIDACIÓN

El proceso de verificación se ha realizado a lo largo de la simulación. Se ha comprobado que los pasos se ejecutan en el orden esperado, mediante los datos de trazabilidad descritos en la Tabla 5. Además de esto, se ha revisado la lógica de los algoritmos propuestos en el Anexo B.

Respecto al proceso de validación, se ha tomado como referencia el recuento ensayos realizados durante un año y el tiempo de funcionamiento de máquina. Estos dos datos se han comparado resultando similares.

4. RESULTADOS

Una vez el modelo de la simulación ha sido finalizado, los siguientes escenarios expuestos en la Tabla 10 han sido simulados mediante el cambio de sus respectivas variables de entrada expuestas en las Tablas 6 y 7. Cada escenario ha sido ejecutado 100 veces.

A continuación, se ha procedido a analizar las tablas de datos obtenidos por las simulaciones.

En la Figura 14 se han graficado las distribuciones del número de ensayos realizados, en un año simulado, para los escenarios $E0$, $T2$, $T3$ y $T4$. Se ha trazado la frecuencia con la que ocurría un determinado número de ensayos en el histograma. En el eje X se representa la cantidad de ensayos realizados, mientras que el eje Y mide la cantidad de veces que se repite un número de ensayos en el conjunto de ejecuciones obtenido. La Figura 14A corresponde a los ensayos *Endurance*, y la Figura 14B a los ensayos *Rolling Resistance*.

En la Figura 14A puede observarse la distinción que ocurre entre las distintas configuraciones. Cada uno de los escenarios tiene un amplio rango de ensayos, mostrando el comportamiento estocástico de este tipo de ensayos. Exceptuando el escenario $T4$, la cantidad de ensayos, parece variar dentro de un rango común de 100 ensayos. Esta tendencia cambia al disponer los 365 días del año de un operario que monte la máquina. Al eliminarse la posibilidad de que el ensayo finalice fuera del horario laboral, la desviación estándar de ensayos realizados se reduce al mismo tiempo que la media de ensayos se aproxima a la capacidad ideal expuesta en la Tabla 11.

Tabla 10: Listado de los escenarios simulados.

ID	Variable alterada	Descripción
E0	Ninguna	Escenario original
T2	Turnos	1 técnico añadido trabajando a relevos. 16 horas de técnico disponible diariamente de lunes a viernes.
T3		2 técnicos añadidos trabajando a relevos. 24 horas diarias de técnico disponible de lunes a viernes.
T4		3 técnicos añadidos trabajando a relevos. Técnico disponible 24 horas 365 días al año.
W2	Técnicos	1 técnico añadido que trabaja simultáneamente al existente. La función de este técnico es dar soporte al primero
M1	Máquinas <i>Endurance</i>	1 sola máquina <i>Endurance</i> instalada en el laboratorio.
M2		2 máquinas <i>Endurance</i> instaladas en el laboratorio.
M3		3 máquinas <i>Endurance</i> instaladas en el laboratorio.
M5		5 máquinas <i>Endurance</i> instaladas en el laboratorio.
M10		10 máquinas <i>Endurance</i> instaladas en el laboratorio.

Tabla 11: Capacidad máxima de ensayos por máquina.

Ensayo	Capacidad máxima (unds.)
<i>Endurance</i>	934
<i>Rolling Resistance</i>	2502

En la Figura 14B, los escenarios *E0*, *T2* y *T3*, muestran un comportamiento similar al observado en el escenario *T4* de la Figura 14A. Esto se debe, a que en el horario de trabajo disponible, solamente puede llegar a lanzar un determinado numero de ensayos, debido a los descansos entre turnos. Esta cantidad se mantiene uniforme a lo largo de las ejecuciones gracias a que la duración de este ensayo es inferior a la duración del horario laboral disponible. Mientras el horario del que se disponga de

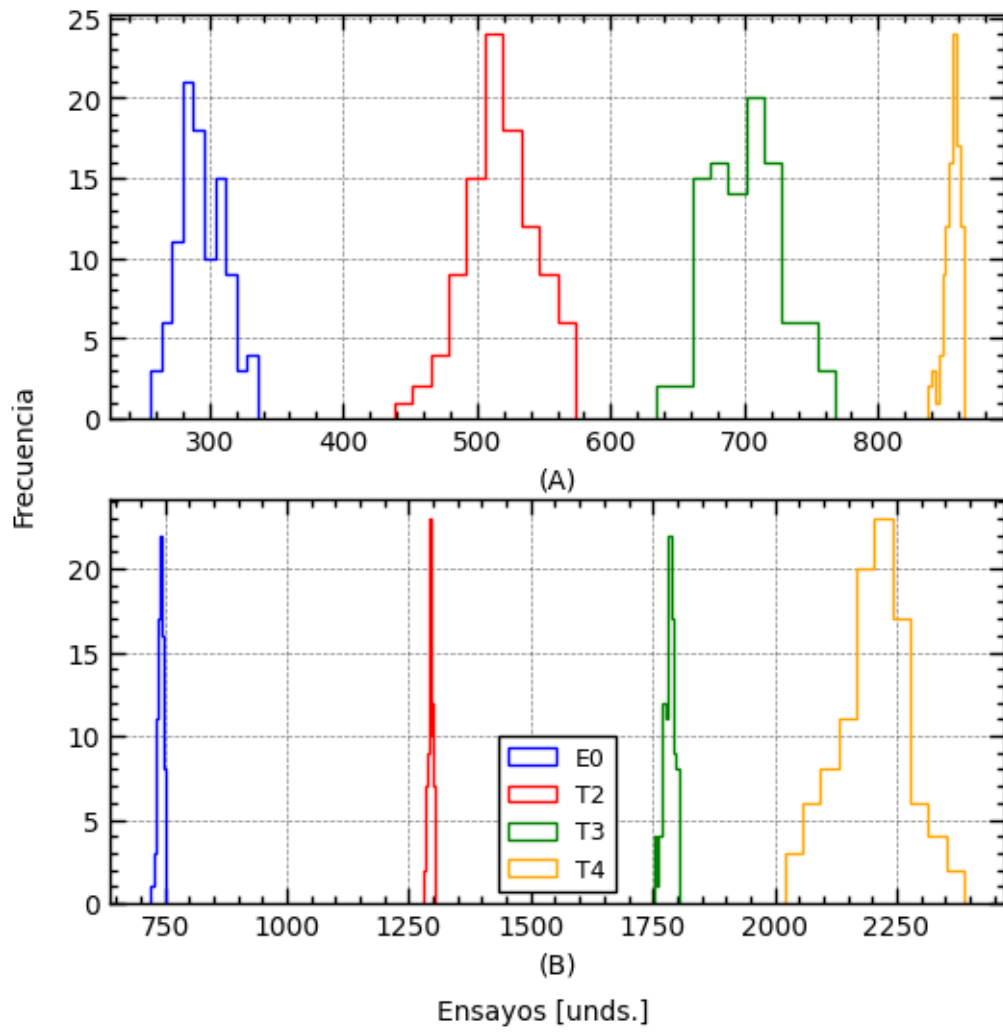


Figura 14: Distribución de ensayos realizados para los escenarios descritos en la leyenda. (A) Ensayos *Endurance*. (B) Ensayos *Rolling Resistance*.

un técnico sea limitado, la tendencia de ensayos corresponderá a la ecuación (8). En el escenario *T3*, la acumulación de eventos aleatorios se vuelve notable en la desviación de ensayos observada respecto a los escenarios *E0* y *T2*. Esto se ve reflejado en la mayor dispersión de *T3*. Finalmente, en el escenario *T4*, la desviación estándar aumenta considerablemente, ya que no hay un descanso que regule la aleatoriedad de los ensayos.

$$TestRealizados = 365 \cdot 8 \cdot \underbrace{\left(\frac{N_{turnos}}{\mu_{test}} + 1 \right)}_{\text{Entero truncado}} \quad (8)$$

Una representación más significativa ocurre cuando se normaliza la cantidad de ensayos respecto a la capacidad máxima de trabajo. La normalización se ha realizado de la siguiente manera:

1. Dividir el conjunto de ensayos realizados obtenido como variable de salida en cada escenario entre la capacidad máxima de ensayos representada en la Tabla 11. De esta manera se obtiene la fracción de tiempo en funcionamiento de la máquina, es decir, la saturación de la máquina, representada en el eje *X*.
2. Dividir la frecuencia con la que ocurre un determinado rango de ensayos en una simulación entre el tamaño del conjunto. Por ejemplo, si la población del conjunto es de 100 ejecuciones, y se han contado un total de 10 ensayos en el grupo entre 250 y 300 ensayos, se realiza la siguiente operación $10/100 = 0,1$. De esta manera se logra la frecuencia relativa representada en el eje *Y*.

Siguiendo estos pasos, se obtienen las distribuciones de tiempo de saturación de máquina representadas en la Figura 15. Esta conversión logra asentar la base necesaria para comparar el grado de utilización de máquinas de manera directa.

Si se grafica la fracción de tiempo de utilización de máquina, respecto a la fracción del tiempo trabajado, con los escenarios expuestos hasta ahora, se obtiene la Figura 16. La dispersión de los datos se ajusta perfectamente a una regresión lineal a partir de la fracción de tiempo trabajado de 0.25 en adelante. Para ambas máquinas, la relación entre ensayos realizados y tiempo trabajado es directamente proporcional. La obtención de esta relación, junto con las capacidades máximas de las maquinas expuestas en la Tabla 11 permite anticipar si la capacidad del LCP será suficiente para saciar la demanda de ensayos.

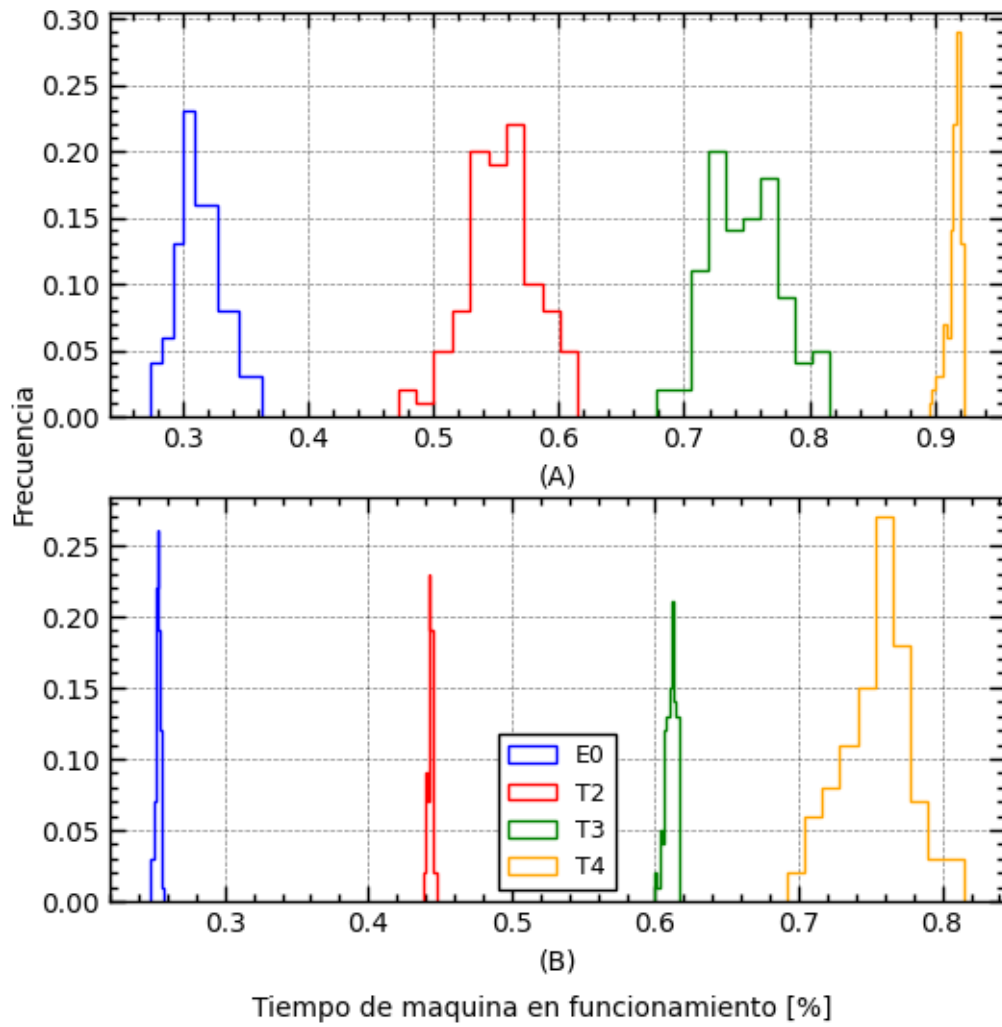


Figura 15: Distribución del nivel de saturación en tiempo de lo escenarios descritos en la leyenda. (A) Ensayos *Endurance*. (B) Ensayos *Rolling Resistance*.

Tabla 12: Media y desviación estándar de ensayos realizados por cada escenario y tipo de ensayo.

Escenario	Ensayos <i>Endurance</i> (unds.)		Ensayos <i>Rolling Resistance</i> (unds.)	
	μ	σ	μ	σ
E0	293.35	17.27	739.35	5.51
T2	516.83	26.38	1293.40	5.00
T3	701.07	27.25	1782.26	11.15
T4	855.06	5.70	2202.28	73.18

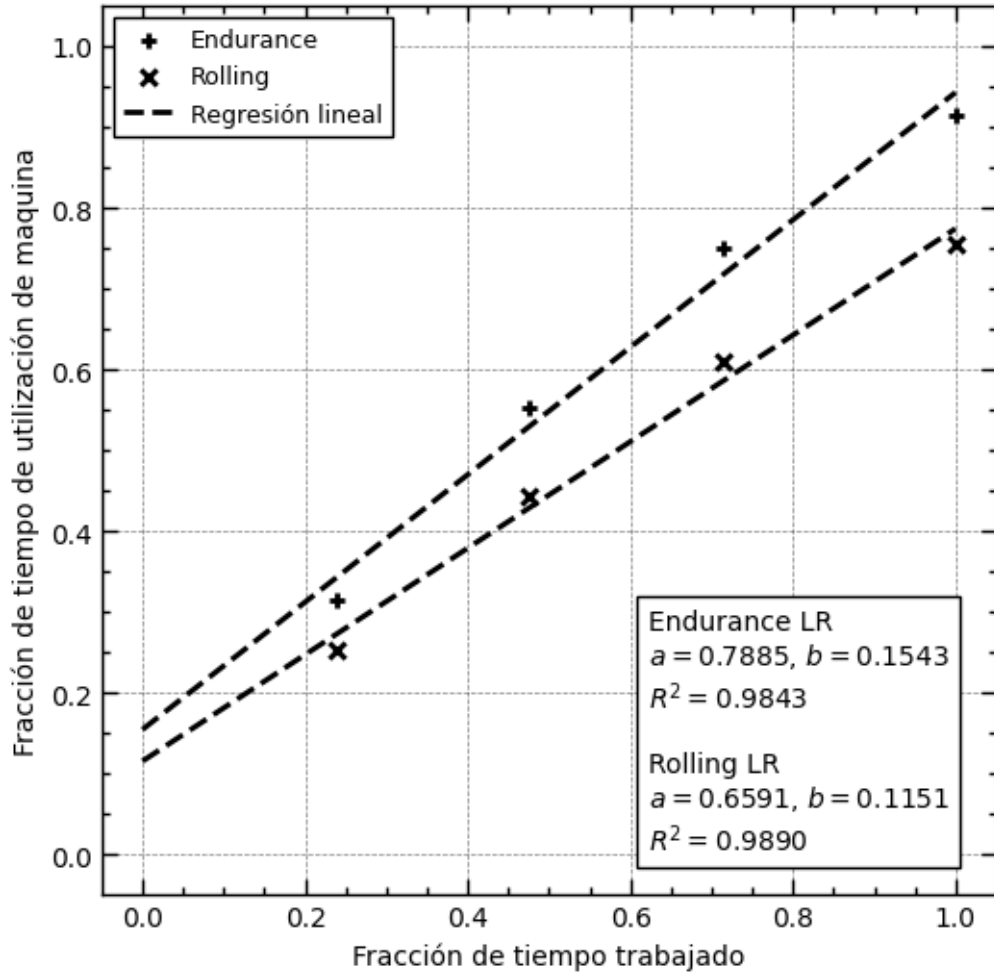


Figura 16: Tendencia proporcional del aumento del número de ensayos al añadir turnos de trabajo.

Teniendo en cuenta los datos obtenidos a través de la regresión lineal, la ecuación (9) modela la cantidad de ensayos *Endurance* realizados en función a la fracción de horas trabajadas en un año. Siendo la fracción de horas trabajadas descrita como la ecuación (11). A su vez, la ecuación (10) aplica la misma función a los ensayos *Rolling Resistance*. Para un determinado objetivo de test realizados, las ecuaciones mencionadas pueden resolverse para obtener la cantidad de horas necesarias para llegar al objetivo. En el caso de requerir más horas de las que una jornada laboral normal dispone, se propone completar dichas horas sobrantes reposicionando uno de los técnicos en un turno adicional durante un periodo. Dicho periodo se puede calcular mediante la ecuación (13), de esta manera se aprovechan

al máximo los recursos humanos disponibles. Para ello primeramente se calcula el máximo de turnos anuales mediante la ecuación (12).

$$TestRealizados = 934 \cdot 0,79 \cdot f_{ttrabajo} + 0,15 \quad (9)$$

$$TestRealizados = 2502 \cdot 0,66 \cdot f_{ttrabajo} + 0,12 \quad (10)$$

$$f_{ttrabajo} = \frac{t_{trabajo}}{365 \frac{\text{días}}{\text{año}} \cdot 24 \frac{\text{horas}}{\text{días}}} \quad (11)$$

$$G(f_{ttrabajo}) = f_{ttrabajo} \frac{365 \frac{\text{días}}{\text{año}} \cdot 24 \frac{\text{horas}}{\text{días}}}{8 \frac{\text{horas}}{\text{días}} \cdot 5 \frac{\text{días}}{\text{semanas}} \cdot 52 \frac{\text{semanas}}{\text{año}}}$$

$$Turnos = \begin{cases} 1 & G(f_{ttrabajo}) \leq 1 \\ 2 & 1 < G(f_{ttrabajo}) \leq 2 \\ 3 & 2 < G(f_{ttrabajo}) \leq 3 \\ 4 & G(f_{ttrabajo}) > 3 \end{cases} \quad (12)$$

$$Periodo = 12 \text{ meses} \cdot (G(f_{ttrabajo}) - Turnos - 1) \quad (13)$$

Con el fin de descartar otro tipo de configuraciones, como múltiples técnicos trabajando simultáneamente, se ha graficado la Figura 17. En el gráfico se representa la distribución de los conjuntos de tiempo de utilización de máquina para los escenarios *E0* y *W2*. La Figura 17A corresponde a los ensayos *Endurance*, mientras que la Figura 17B expone los ensayos *Rolling Resistance*. La diferencia en ambos ensayos parece mínima, pero con la intención de descartar la idea de que 2 técnicos trabajando simultáneamente incrementa el numero de ensayos realizados, se desarrolla el siguiente test de hipótesis.

- $H_1: \mu_1 \neq \mu_2$
- $H_0: \mu_1 = \mu_2$

Se ha tratado de descartar la hipótesis nula mediante la prueba de valor p, y test t de Student. Los resultados de la prueba, han sido recogidos en la Tabla 13.

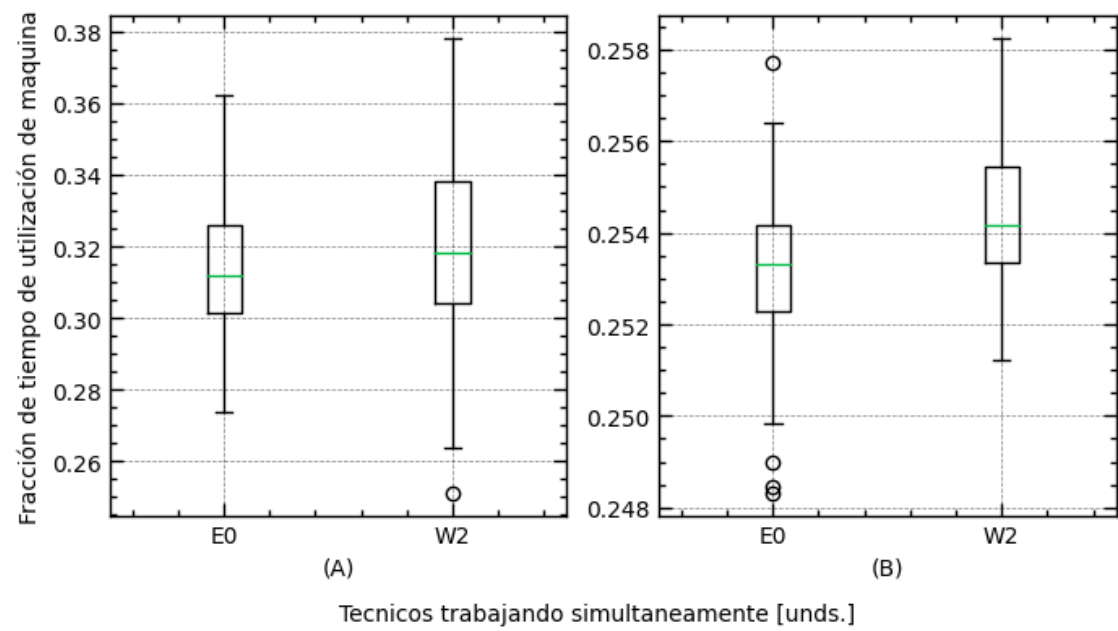


Figura 17: Comparación del nivel de saturación en tiempo entre 1 técnico y 2 técnicos trabajando simultáneamente. (A) Ensayos *Endurance*. (B) Ensayos *Rolling Resistance*.

Tabla 13: Resultados del test estadístico formulado a partir de los resultados de la Figura 17

Resultados	Endurance	Rolling
valor-t	-2.279	-5.179
valor-p	0.023	0.000

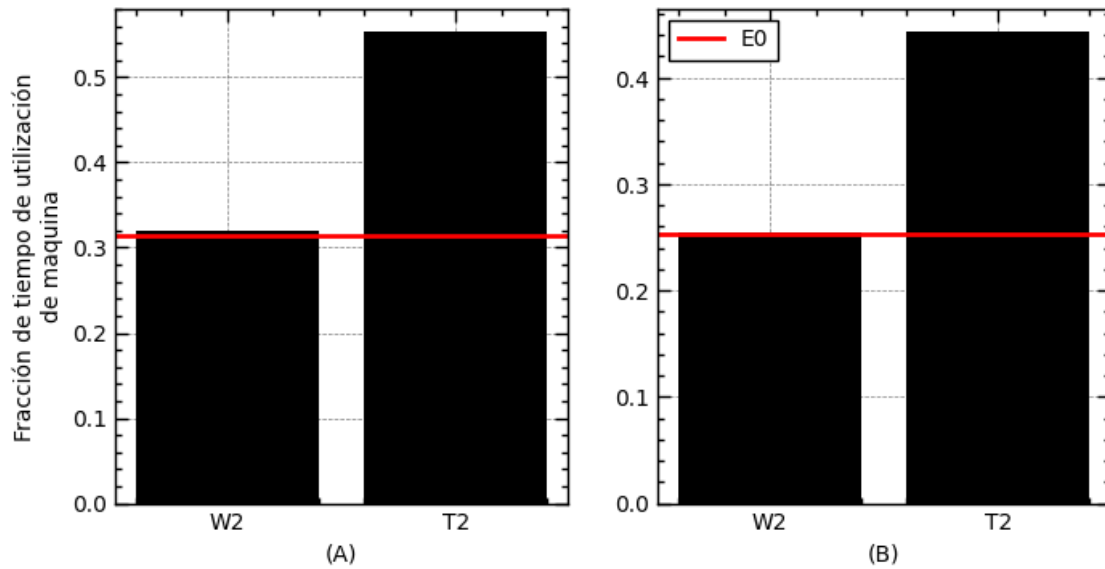


Figura 18: Comparación del nivel de saturación en tiempo entre los escenarios $W2$ y $T2$ respecto al escenario $E0$. (A) Ensayos *Endurance*. (B) Ensayos *Rolling Resistance*.

Siendo el valor $p < 0,05$ en ambos tipos de ensayo se descarta la hipótesis nula formulada anteriormente. Esto significa que las distribuciones tienen una diferencia estadísticamente significativa. Aún siendo este el caso, el escenario en el que se trabaja a 2 turnos, es definitivamente superior, como se puede observar en la Figura 18. Al consumir ambos la misma cantidad de recursos, esto descarta la configuración en la que 2 técnicos trabajan simultáneamente.

Por último, se ha tratado de observar el rendimiento de ensayos que aportaría la instalación de nuevas máquinas para el ensayo *Endurance*. Para ello se han simulado los escenarios $M1$, $M2$, $M3$, $E0$, $M5$ y $M10$. A partir del conjunto de datos de cada escenario, se ha obtenido la media de los ensayos realizados en un año para uno de ellos. En la Figura 19, se observa como añadir máquinas no aumenta de manera significativa la cantidad de ensayos realizados, ya que el técnico

queda saturado de trabajo con la configuración actual.

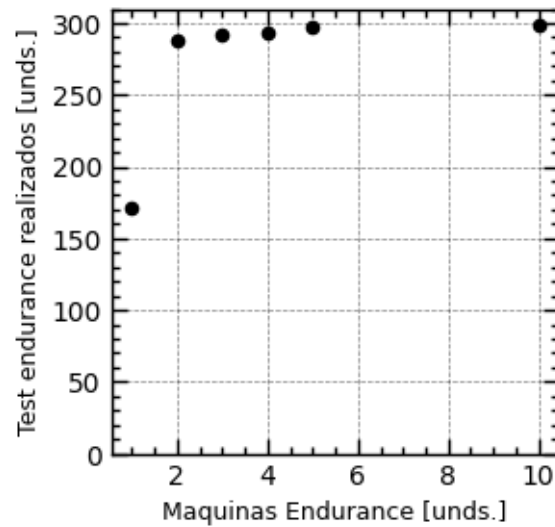


Figura 19: Aumento de los ensayos realizados en función del número de maquinas *Endurance* instaladas.

Este análisis concluye, que en caso de necesitar capacidad adicional, se opte por reposicionar a uno de los técnicos en un turno adicional durante el periodo que sea necesario. Habiendo demostrado ser la opción óptima entre todos los escenarios simulados.

5. CONCLUSIONES

A modo de cierre de este trabajo, se puede señalar que se llegó a lograr el objetivo planteado al comienzo del proyecto. Dentro de las posibilidades tomadas en cuenta, el escenario mas óptimo ha sido hallado. Mantener un enfoque hacia los hitos marcados al principio, ha ayudado en el extenso proceso de la elaboración de la simulación.

Entre los resultados más relevantes obtenidos, se encuentran la gran capacidad de mejora que posee el laboratorio en su estado actual. El descubrimiento de una tendencia proporcional entre la cantidad de horas trabajadas y el numero de ensayos realizados, ha sido particularmente útil. Mediante los resultados de

la regresión obtenida, se podrá facilitar la futura organización de los turnos de trabajo, y anticipar las desviaciones respecto a los objetivos de manera más precisa. El descarte de la ampliación de la maquinaria del laboratorio, supondrá un ahorro tanto en espacio como en inversión de la fábrica.

Respecto al proceso, se han encontrado numerosas dificultades. Primeramente, la familiarización con las DES, ha requerido un extenso trabajo de investigación acerca de sus métodos e implementaciones. Se ha requerido ajustar este método de simulación a los objetivos del trabajo, lo que ha supuesto un periodo previo al desarrollo, en el que se han valorado múltiples enfoques. La tarea más complicada, ha resultado ser el proceso de traducción del modelo desde los diagramas, a el código fuente. Este periodo, ha durado semanas, en las cuales se ha aprendido cero a usar la librería Simpy. El proyecto, al involucrar tantas líneas de código, ha supuesto un proceso complicado de detección y corrección de errores de funcionamiento, que ha limitado lo que podría haber sido el análisis final. Aún así, el análisis de los resultados ha resultado satisfactorio, para un Trabajo de Fin de Grado.

Este trabajo ha causado, de manera transversal, un entendimiento aún mas detallado del proceso de lo que se tenía en un inicio. Además, los conocimientos obtenidos en este TFG, podrán ser útiles en futuros procesos de optimización en producción.

6. GLOSARIO

CP: Conformidad de la Producción.

DES: Simulación de Eventos Discretos.

IND: Industrialización.

LCP: Laboratorio de Calidad del Producto.

LI: Índice de Carga.

FDP: Función de Distribución de Probabilidad.

PIB: Producto Interior Bruto.

SD: Sistemas Dinámicos.

VV: Verificación y Validación.

Referencias

- Abu-Taieh, E. M., El Sheikh, A. A. R., Abu-Tayeh, J. M., y Al Abdallat, H. (2009). History of simulation. En *Encyclopedia of Information Science and Technology, Second Edition*, pp. 1769–1776. IGI Global.
- Allen, T. T. (2011). *Introduction to discrete event simulation and agent-based modeling: voting systems, health care, military, and manufacturing*. Springer Science & Business Media.
- Babulak, E. y Wang, M. (2010). Discrete event simulation. *Aitor Goti (Hg.): Discrete Event Simulations. Rijeka, Kroatien: Sciyo*, p. 1.
- Banks, J. (1998). *Handbook of simulation: principles, methodology, advances, applications, and practice*. John Wiley & Sons.
- Banks, J. (1999). Introduction to simulation. En *Proceedings of the 31st conference on Winter simulation: Simulation—a bridge to the future-Volume 1*, pp. 7–13.
- Chicu, N., Prioteasa, A.-L., y Deaconu, A. (2020). Current trends and perspectives in tyre industry. *Studia Universitatis Vasile Goldiș Arad, Seria Științe Economice*, 30(2):36–56.
- Díaz, S. y Garriga, J. M. (2021). El sector del automóvil en españa: Estratégico y en transformación.
- Garzia, R. F., Garzia, M. R., y Zeigler, B. P. (1986). Discrete-event simulation: When prototyping of large, complex systems is impossible, this technique, based on known behaviors of interacting elements, is useful. *IEEE Spectrum*, 23(12):32–36.
- Lawson, B. y Leemis, L. (2008). Monte carlo and discrete-event simulations in c and r. En *2008 Winter Simulation Conference*, pp. 11–16. IEEE.
- Martin, A. J. (2022). Introduction to hypothesis testing.
- Meester, R. (2008). *A natural introduction to probability theory*. Springer Science & Business Media.
- Owen, A. B. y Glynn, P. W. (2016). *Monte Carlo and Quasi-Monte Carlo Methods*. Springer.
- Pritsker, A. A. B. (1974). *The Gasp Iv Simulation Language [By] A. Alan B. Pritsker*. Wiley.

- Rodgers, B. (2020). *Tire Engineering: An Introduction*. CRC Press.
- Shannon, R. y Johannes, J. D. (1976). Systems simulation: the art and science. *IEEE transactions on systems, man, and cybernetics*, (10):723–724.
- Simon, M. K. (2002). *Probability distributions involving Gaussian random variables: A handbook for engineers and scientists*. Springer.
- Sweetser, A. (1999). A comparison of system dynamics (sd) and discrete event simulation (des). En *17th International Conference of the System Dynamics Society*, pp. 20–23.
- Varga, A. (2001). Discrete event simulation system. En *Proc. of the European Simulation Multiconference (ESM'2001)*, pp. 1–7.
- Ydrefors, L., Hjort, M., Kharrazi, S., Jerrelind, J., y Stensson Trigell, A. (2021). Rolling resistance and its relation to operating conditions: A literature review. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 235(12):2931–2948.

A. ANEXO A: CÓDIGO FUENTE

```
# *****#
#
#
#          ::::::::::: :::::::::::#
#  lep_sim_rewrited.py          :+:      :+:      :+:#
#
#          +:++      +:++      +:++#
#  By: andoitzcp <andoitzcp@gmail.com>      +#+      +#++#+#
#
#          +#+      +#+#
#  Created: 2023/02/01 01:09:17 by andoitzcp      +#+      +#+
#  Updated: 2023/02/01 01:09:31 by acampo-p      ###      #####.fr#
#
# *****#
```

```
import simpy
import pandas as pd
from numpy import random as nprng
from functools import reduce
```

```
rng = nprng.default_rng()
```

```
class G:
    SIM_DET = {
        'SIM_DUR': 365 * 24 * 60,
        'SHIFT_DUR': 8 * 60,
        'SHIFT_N': 1,
        'RUN_EXEC': 100,
    }
    P_QCED = {
        'INTR_RVAL': 20 * 60 / 10,
        'R_PRIO': 0,
        'M_MEAN': 45,
        'M_SDEV': 10,
        'M_PRIO': -1,
        'C_DUR': 3 * 60,
        'C_PRIO': -3,
        'P_DUR': 5,
        'P_PRIO': -4,
        'T_MEAN': 72 * 60,
```

```

        'T_SDEV': 6 * 60,
    }
P_RR = {
    'INTR_RVAL': 8 * 60 / 10,
    'R_PRIO': -1,
    'M_MEAN': 30,
    'M_SDEV': 5,
    'M_PRIO': -2,
    'C_DUR': 6 * 60,
    'C_PRIO': -2,
    'T_POS': [150, 170, 200],
    'T_PROB': [0.1, 0.7, 0.2],
}
RSRCS = {
    'INDOOR': 4,
    'RRM': 1,
    'BUTLR': 1,
    'TECHN': 1,
    'MANT': 1,
    'OPR': 1,
    'Q_RIMS': 14,
    'R_RIMS': 20,
}
P_RIM = {
    'PRIO': 0,
    'A': 12,
    'M': 15,
    'B': 20,
}

```

```

class data:
    df = pd.DataFrame()

    def calc_intervals(df):
        df['Q_GEN'] = df['RIM1'] - df['GEN']
        df['Q_RIM'] = df['RIM2'] - df['RIM1']
        df['D_RIM'] = df['RIM3'] - df['RIM2']
        df['Q_MOUNT'] = df['MOUNT1'] - df['RIM3']
        df['T_MOUNT'] = df['MOUNT2'] - df['MOUNT1']

```

```

df['D_MOUNT'] = df['MOUNT3'] - df['MOUNT2']
df['Q_COND'] = df['COND1'] - df['MOUNT3']
df['T_COND'] = df['COND2'] - df['COND1']
df['D_COND'] = df['COND3'] - df['COND2']
df['Q_PMEAS'] = df['PMEAS1'] - df['COND3']
df['T_PMEAS'] = df['PMEAS2'] - df['PMEAS1']
df['D_PMEAS'] = df['PMEAS3'] - df['PMEAS2']
df['Q_QCED'] = df['QCED1'] - df['PMEAS3']
df['D_QCED'] = df['QCED2'] - df['QCED1']

df['Q_GEN_h'] = df['Q_GEN'] / 60
df['Q_RIM_h'] = df['Q_RIM'] / 60
df['D_RIM_h'] = df['D_RIM'] / 60
df['Q_MOUNT_h'] = df['Q_MOUNT'] / 60
df['T_MOUNT_h'] = df['T_MOUNT'] / 60
df['D_MOUNT_h'] = df['D_MOUNT'] / 60
df['Q_COND_h'] = df['Q_COND'] / 60
df['T_COND_h'] = df['T_COND'] / 60
df['D_COND_h'] = df['D_COND'] / 60
df['Q_PMEAS_h'] = df['Q_PMEAS'] / 60
df['T_PMEAS_h'] = df['T_PMEAS'] / 60
df['D_PMEAS_h'] = df['D_PMEAS'] / 60
df['Q_QCED_h'] = df['Q_QCED'] / 60
df['D_QCED_h'] = df['D_QCED'] / 60

def stat_calc(df):
    filtered_df = df.loc[df['D_QCED_h'] > 0]
    df2 = filtered_df.groupby(['TEST'])['TEST'].count()
    print(df2)

class tire:

    def __init__(self, t_id, t_test, t_client):
        self.id = t_id
        self.test = t_test
        self.client = t_client

class lab_model:

```

```

def __init__(self, run_nbr):
    self.env = simpy.Environment()
    self.run_nbr = run_nbr
    self.tire_cntr = 0
    self.tire_bufrr_indoor = []
    self.tire_bufrr_rr = []

    # Resources
    self.rsrc_indoor = \
        simpy.Resource(self.env, capacity=G.RSRCS['INDOOR'])
    self.rsrc_rrm = \
        simpy.Resource(self.env, capacity=G.RSRCS['RRM'])
    self.rsrc_butlr = \
        simpy.Resource(self.env, capacity=G.RSRCS['BUTLR'])
    self.rsrc_techn = \
        simpy.PreemptiveResource(self.env, capacity=G.RSRCS['TECHN'])
    self.rsrc_mant = \
        simpy.Resource(self.env, capacity=G.RSRCS['MANT'])
    self.rsrc_opr = \
        simpy.Resource(self.env, capacity=G.RSRCS['OPR'])
    self.rsrc_qrims = \
        simpy.Resource(self.env, capacity=G.RSRCS['Q_RIMS'])
    self.rsrc_rrims = \
        simpy.Resource(self.env, capacity=G.RSRCS['R_RIMS'])

    # Temporal storage for data
    self.tmp_lst_gen = []
    self.tmp_lst_rim = []
    self.tmp_lst_mount = []
    self.tmp_lst_cond = []
    self.tmp_lst_pmeasure = []
    self.tmp_lst_test = []

def gen_qced_rp(self):
    self.env.timeout(32.59)
    while True:
        tmp_dict = {'RUN': self.run_nbr,
                    'TEST': 'QCED',
                    'CLIENT': 'RP',
                    }

```

```

        self.tire_cntr += 1
        obj = tire(self.tire_cntr, "QCED", "RP")
        self.tire_buf indoor.append(obj)
        intr_rval = rng.exponential(G.P_QCED['INTR_RVAL'])

        tmp_dict['ID'] = self.tire_cntr
        tmp_dict['GEN'] = self.env.now
        self.tmp_lst_gen.append(tmp_dict)
        self.env.process(self.proc_master_indoor())
        yield self.env.timeout(intr_rval)

def gen_rr_rp(self):
    while True:
        tmp_dict = {'RUN': self.run_nbr,
                    'TEST': 'RR',
                    'CLIENT': 'RP',
                    }

        self.tire_cntr += 1
        obj = tire(self.tire_cntr, "RR", "RP")
        self.tire_buf rr.append(obj)
        intr_rval = rng.exponential(G.P_RR['INTR_RVAL'])

        tmp_dict['ID'] = self.tire_cntr
        tmp_dict['GEN'] = self.env.now
        self.tmp_lst_gen.append(tmp_dict)
        self.env.process(self.proc_master_rr())
        yield self.env.timeout(intr_rval)

def obstruct_techn(self):
    day = 0
    week = ['M', 'T', 'X', 'R', 'F', 'S', 'U']
    t_up = G.SIM_DET['SHIFT_N'] * G.SIM_DET['SHIFT_DUR']
    while True:
        t_down = (24 * 60) - t_up
        if week[day % 7] == 'F':
            day += 3
            t_down += 48 * 60
        else:

```

```

        day += 1
        yield self.env.timeout(t_up)
        with self.rsrc_techn.request(priority=-10, preempt=True) \
            as req_techn_A:
            yield req_techn_A
        yield self.env.timeout(t_down)

def obstruct_indoor(self):
    while True:
        t_tofailure = rng.exponential(G.F_DET['INTR'])
        yield self.env.timeout(t_tofailure)
        if not self.broken:
            self.proc_qced_A.interruptp()

def t_conditioning(self, t_left):
    t_toshiftend = (G.SIM_DET['SHIFT_DUR'] * G.SIM_DET['SHIFT_N']) \
        - self.env.now % (24*60)
    if G.SIM_DET['SHIFT_N'] == 3:
        return t_left
    elif t_left > t_toshiftend:
        t_left = G.SIM_DET['SHIFT_DUR'] * (3 - G.SIM_DET['SHIFT_N'])
        t_left += t_toshiftend
    return t_left

def proc_master_indoor(self):
    yield (self.env.timeout(2*24*60))
    if len(self.tire_buf indoor) < 2:
        pass
    else:
        qced_sample_A = self.tire_buf indoor.pop(0)
        qced_sample_B = self.tire_buf indoor.pop(0)
        with self.rsrc_qgrims.request() as req_qgrims:
            yield req_qgrims
            yield self.env.process(self.proc_rim(qced_sample_A))
            yield self.env.process(self.proc_rim(qced_sample_B))
        with self.rsrc_indoor.request() as req_indoor:
            yield req_indoor
            yield self.env.process(self.proc_mount(qced_sample_A))
            yield self.env.process(self.proc_mount(qced_sample_B))
            yield self.env.process(self.proc_cond_qced(qced_sample_A,

```

```

                                                    qced_sample_B))
        yield self.env.process(
            self.proc_pmeasure(qced_sample_A,
                               qced_sample_B))
        yield self.env.process(
            self.proc_qced_A(qced_sample_A)) & \
            self.env.process(
                self.proc_qced_B(qced_sample_B))

def proc_master_rr(self):
    if len(self.tire_bufrr) > 0:
        rr_sample = self.tire_bufrr.pop(0)
        with self.rsrc_qgrims.request() as req_qgrims:
            yield req_qgrims
            yield self.env.process(self.proc_rim(rr_sample))
            yield self.env.process(self.proc_cond_rr(rr_sample))
        with self.rsrc_rrm.request() as req_rrm:
            yield req_rrm
            yield self.env.process(self.proc_mount(rr_sample))
            yield self.env.process(self.proc_rr(rr_sample))

def proc_rim(self, tire_ent):
    tmp_dict = {'ID': tire_ent.id}

    aux = [self.env.now]
    prio = G.P_RIM['PRIO']
    t_left = rng.triangular(G.P_RIM['A'], G.P_RIM['M'], G.P_RIM['B'])
    while t_left > 0:
        with self.rsrc_techn.request(priority=prio,
                                      preempt=False) as req_techn_rim, \
            self.rsrc_butlr.request() as req_butlr:
            try:
                yield req_techn_rim & req_butlr
                aux.append(self.env.now)
                yield self.env.timeout(t_left)
                t_left = 0
            except simpy.Interrupt as interrupt:
                usage = self.env.now - interrupt.cause.usage_since
                t_left -= usage
                prio -= 0.1

```

```

tmp_dict['RIM1'] = aux[0]
tmp_dict['RIM2'] = aux[1]
tmp_dict['RIM3'] = self.env.now
self.tmp_lst_rim.append(tmp_dict)

def proc_mount(self, tire_ent):
    tmp_dict = {'ID': tire_ent.id}

    start = [self.env.now]
    if tire_ent.test == 'QCED':
        t_left = rng.normal(G.P_QCED['M_MEAN'], G.P_QCED['M_SDEV'])
        prio = G.P_QCED['M_PRIO']
    elif tire_ent.test == 'RR':
        t_left = rng.normal(G.P_RR['M_MEAN'], G.P_RR['M_SDEV'])
        prio = G.P_RR['M_PRIO']
    while t_left > 0:
        with self.rsrc_techn.request(priority=prio, preempt=False) \
            as req_techn_mount:
            yield req_techn_mount
            start.append(self.env.now)
        try:
            yield self.env.timeout(t_left)
            t_left = 0
        except simpy.Interrupt as interrupt:
            usage = self.env.now - interrupt.cause.usage_since
            t_left -= usage
            prio -= 0.1

    tmp_dict['MOUNT1'] = start[0]
    tmp_dict['MOUNT2'] = start[1]
    tmp_dict['MOUNT3'] = self.env.now
    self.tmp_lst_mount.append(tmp_dict)

def proc_cond_qced(self, tire_ent_A, tire_ent_B):
    tmp_dict_A = {'ID': tire_ent_A.id}
    tmp_dict_B = {'ID': tire_ent_B.id}

    prio = G.P_QCED['C_PRIO']
    start = [self.env.now]

```



```

with self.rsrc_techn.request(priority=prio, preempt=False) \
    as req_techn_cond_qced:
    yield req_techn_cond_qced
start.append(self.env.now)
t_left = self.t_conditioning(G.P_QCED['C_DUR'])
yield self.env.timeout(t_left)

tmp_dict_A['COND1'] = start[0]
tmp_dict_B['COND1'] = start[0]
tmp_dict_A['COND2'] = start[1]
tmp_dict_B['COND2'] = start[1]
tmp_dict_A['COND3'] = self.env.now
tmp_dict_B['COND3'] = self.env.now
self.tmp_lst_cond.append(tmp_dict_A)
self.tmp_lst_cond.append(tmp_dict_B)

def proc_cond_rr(self, tire_ent):
    tmp_dict = {'ID': tire_ent.id}

    prio = G.P_RR['C_PRIO']
    aux = [self.env.now]
    with self.rsrc_techn.request(priority=prio, preempt=False) \
        as req_techn_cond_rr:
        yield req_techn_cond_rr
    aux.append(self.env.now)
    t_left = G.P_RR['C_DUR']
    yield self.env.timeout(t_left)

    tmp_dict['COND1'] = aux[0]
    tmp_dict['COND2'] = aux[1]
    tmp_dict['COND3'] = self.env.now
    self.tmp_lst_cond.append(tmp_dict)

def proc_pmeasure(self, tire_ent_A, tire_ent_B):
    tmp_dict_A = {'ID': tire_ent_A.id}
    tmp_dict_B = {'ID': tire_ent_B.id}

    start = [self.env.now]
    start.append(self.env.now)
    yield self.env.timeout(G.P_QCED['P_DUR'])

```

```

tmp_dict_A['PMEAS1'] = start[0]
tmp_dict_A['PMEAS2'] = start[1]
tmp_dict_A['PMEAS3'] = self.env.now
tmp_dict_B['PMEAS1'] = start[0]
tmp_dict_B['PMEAS2'] = start[1]
tmp_dict_B['PMEAS3'] = self.env.now
self.tmp_lst_pmeasure.append(tmp_dict_A)
self.tmp_lst_pmeasure.append(tmp_dict_B)

def proc_qced_A(self, tire_ent):
    tmp_dict = {'ID': tire_ent.id}
    tmp_dict['QCED1'] = self.env.now

    test_failure = False
    t_left = rng.normal(G.P_QCED['T_MEAN'], G.P_QCED['T_SDEV'])
    try:
        yield self.env.timeout(t_left)
    except simpy.Interrupt:
        test_failure = True

    tmp_dict['TEST_FAIL'] = test_failure
    tmp_dict['QCED2'] = self.env.now
    self.tmp_lst_test.append(tmp_dict)

def proc_qced_B(self, tire_ent):
    tmp_dict = {'ID': tire_ent.id}
    tmp_dict['QCED1'] = self.env.now

    test_failure = False
    t_left = rng.normal(G.P_QCED['T_MEAN'], G.P_QCED['T_SDEV'])
    try:
        yield self.env.timeout(t_left)
    except simpy.Interrupt:
        test_failure = True

    tmp_dict['TEST_FAIL'] = test_failure
    tmp_dict['QCED2'] = self.env.now
    self.tmp_lst_test.append(tmp_dict)

```

```

def proc_rr(self, tire_ent):
    tmp_dict = {'ID': tire_ent.id}
    tmp_dict['QCED1'] = self.env.now

    test_failure = False
    t_left = rng.normal(180, 20)
    try:
        yield self.env.timeout(t_left)
    except simpy.Interrupt:
        test_failure = True
    tmp_dict['TEST_FAIL'] = test_failure
    tmp_dict['QCED2'] = self.env.now
    self.tmp_lst_test.append(tmp_dict)

def df_gen(self):
    df_tmp_gen = pd.DataFrame(self.tmp_lst_gen)
    df_tmp_rim = pd.DataFrame(self.tmp_lst_rim)
    df_tmp_mount = pd.DataFrame(self.tmp_lst_mount)
    df_tmp_cond = pd.DataFrame(self.tmp_lst_cond)
    df_tmp_pmeasure = pd.DataFrame(self.tmp_lst_pmeasure)
    df_tmp_test = pd.DataFrame(self.tmp_lst_test)
    print(df_tmp_gen)

    df_tmp_lst = [df_tmp_gen,
                  df_tmp_rim,
                  df_tmp_mount,
                  df_tmp_cond,
                  df_tmp_pmeasure,
                  df_tmp_test]
    df_tmp_master = reduce(lambda left, right:
                           pd.merge(left, right, on=['ID'], how='outer'),
                           df_tmp_lst)
    data.df = pd.concat([data.df, df_tmp_master], ignore_index=True)

def run(self):
    self.env.process(self.obstruct_techn())
    self.env.process(self.gen_qced_rp())
    self.env.process(self.gen_rr_rp())
    self.env.run(until=G.SIM_DET['SIM_DUR'])

```

```

        self.df_gen()

for run in range(G.SIM_DET['RUN_EXEC']):
    print("Run ", run+1, "of ", G.SIM_DET['RUN_EXEC'], sep="")
    print(G.SIM_DET['RUN_EXEC'])
    my_test_model = lab_model(run)
    my_test_model.run()
    print()

data.calc_intervals(data.df)
data.df.to_csv('lep.csv', index=False)
df = data.df
print(df)
print(data.stat_calc(df))

```

B. ANEXO B: DIAGRAMAS DE FLUJO

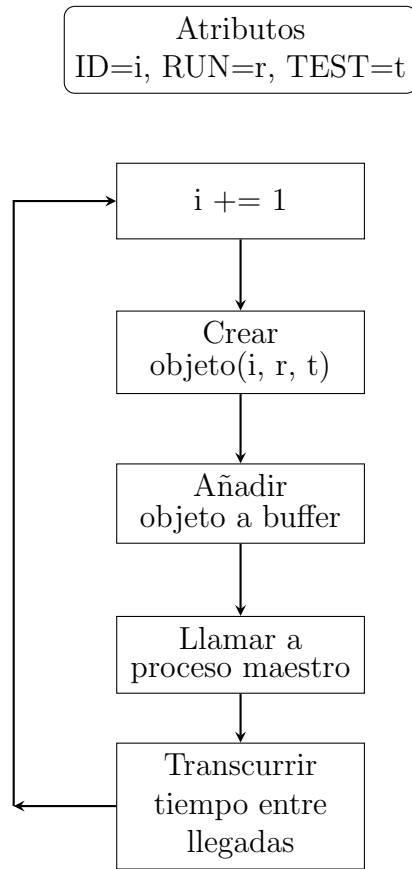


Figura 20: Modelo de generación de entidades

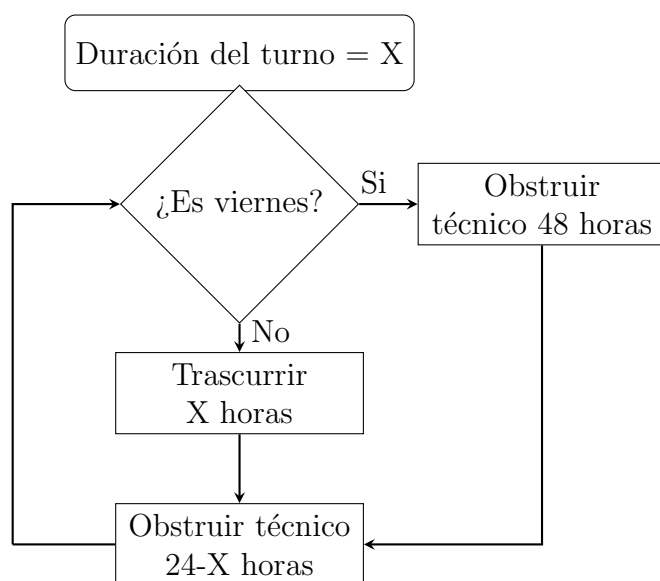


Figura 21: Modelo de obstrucción del técnico durante horas no activas

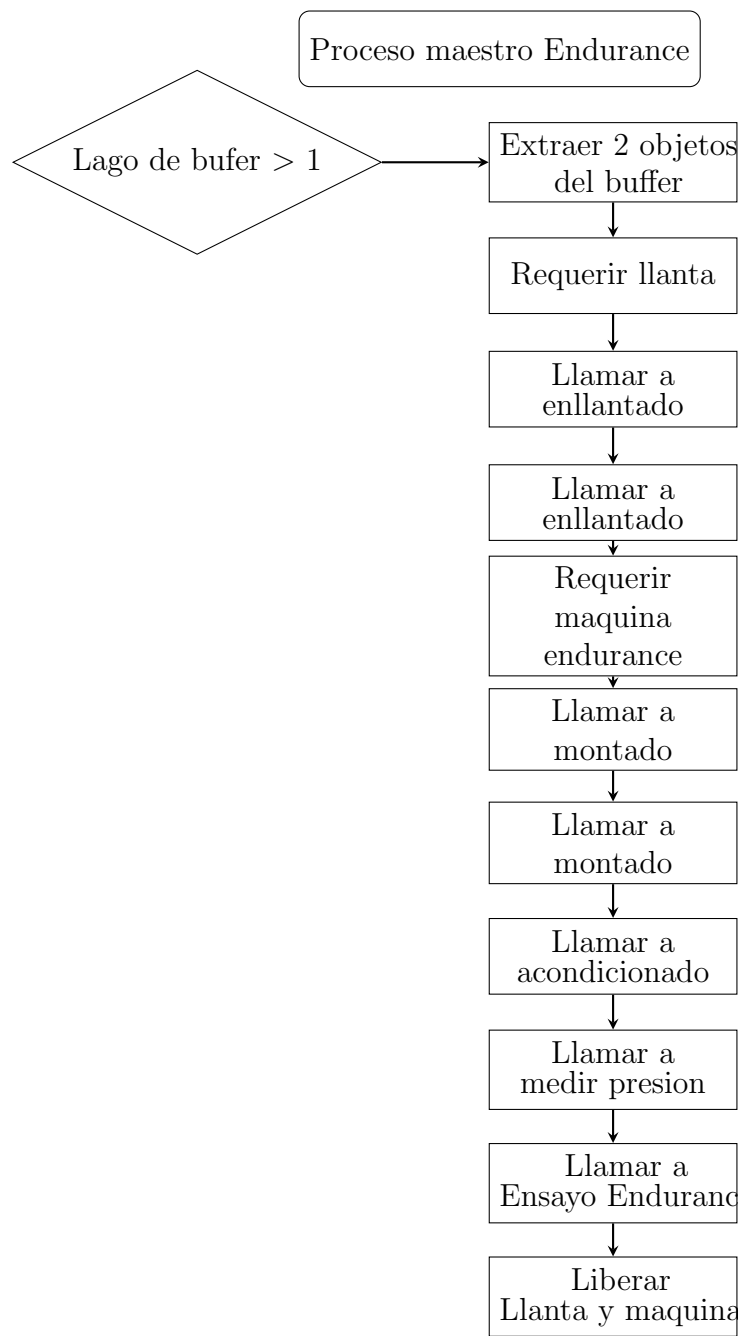


Figura 22: Modelo del proceso maestro de ensayos *Endurance*.

Proceso maestro Rolling Resistance

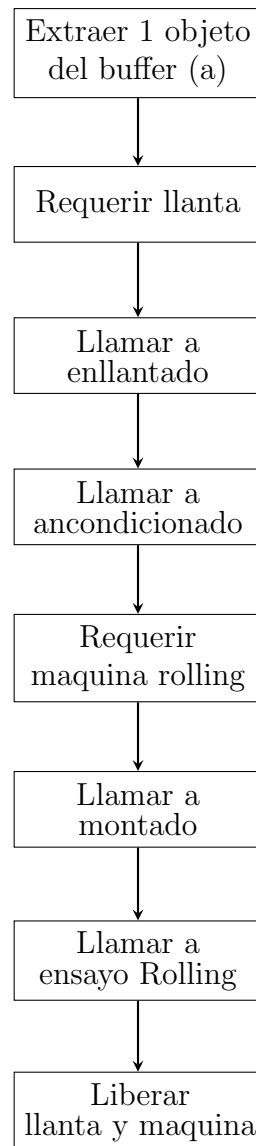


Figura 23: Modelo del proceso maestro de ensayos *Rolling Resistance*.

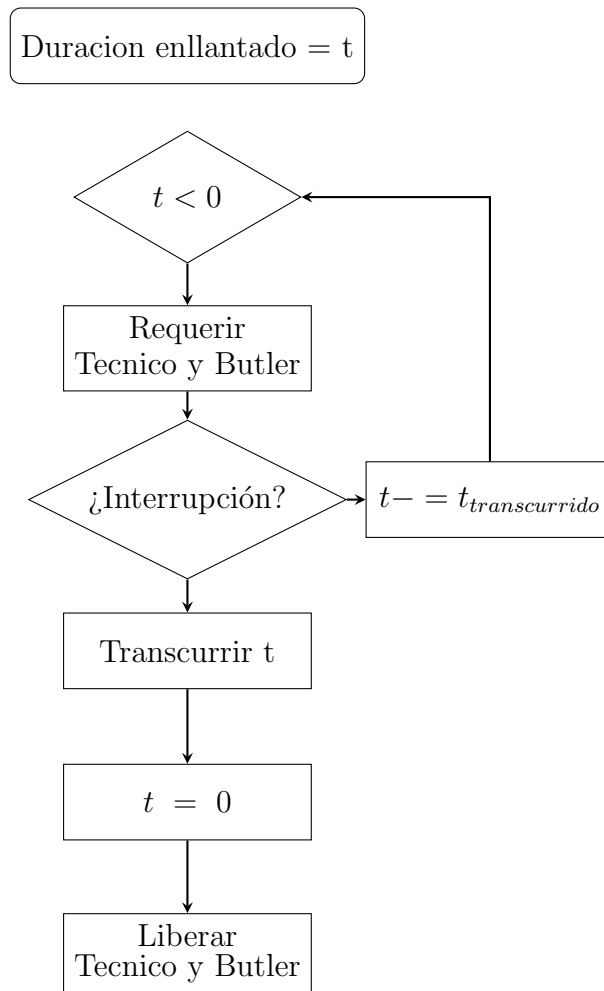


Figura 24: Modelo del subproceso de enlantado

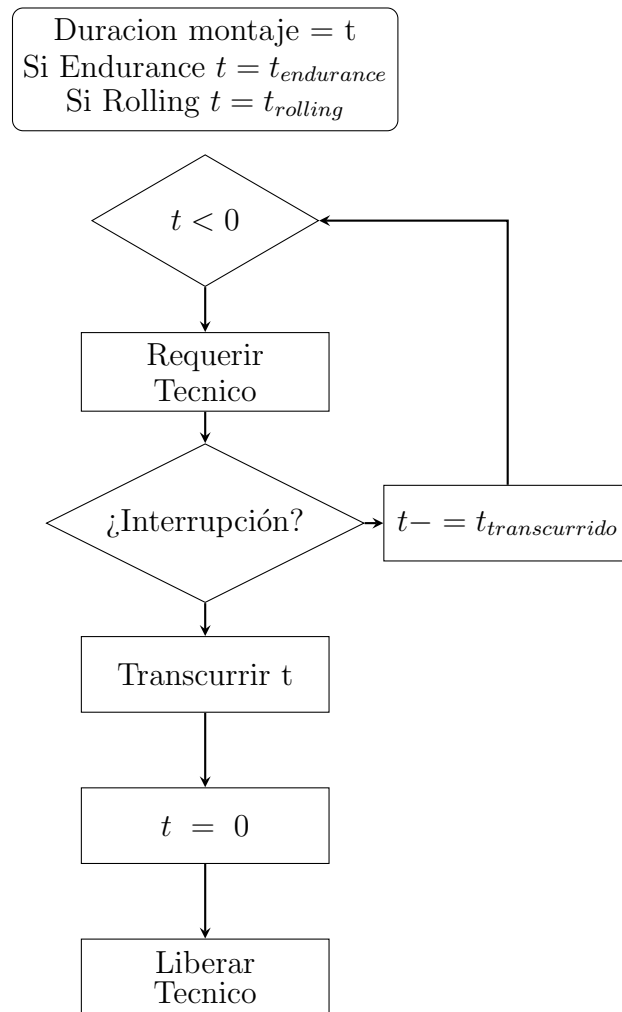


Figura 25: Modelo del subproceso de montado de máquina

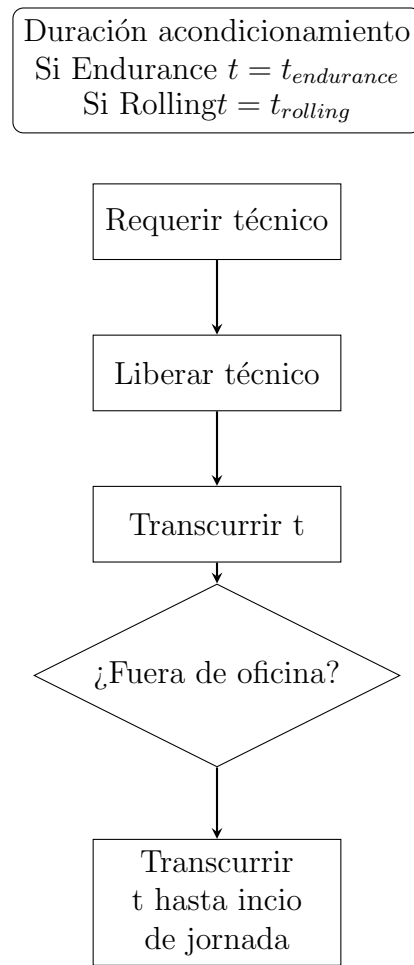


Figura 26: Modelo del subproceso de acondicionado de sala

Duración ajuste de presión = t

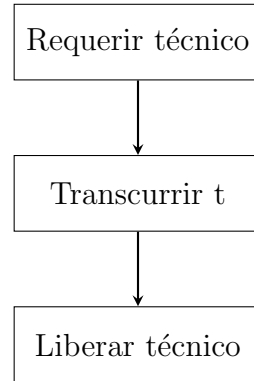


Figura 27: Modelo del subproceso de ajuste de presión de la cubierta

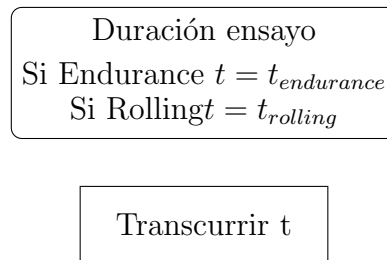


Figura 28: Modelo del subproceso de ensayo de la cubierta

C. ANEXO C: TABLAS

Tabla 14: Carga maxima de neumático segun LI. ETRTO 2022

LI	Carga (Kg)	LI	Carga (Kg)	LI	Carga (Kg)	LI	Carga (Kg)
80	450	104	900	128	1800	152	3550
81	462	105	925	129	1850	153	3650
82	475	106	950	130	1900	154	3750
83	487	107	975	131	1950	155	3875
84	500	108	1000	132	2000	156	4000
85	515	109	1030	133	2060	157	4125
86	530	110	1060	134	2120	158	4250
87	545	111	1090	135	2180	159	4375
88	560	112	1120	136	2240	160	4500
89	580	113	1150	137	2300	161	4625
90	600	114	1180	138	2360	162	4750
91	615	115	1215	139	2430	163	4875
92	630	116	1250	140	2500	164	5000
93	650	117	1285	141	2575	165	5150
94	670	118	1320	142	2650	166	5300
95	690	119	1360	143	2725	167	5450
96	710	120	1400	144	2800	168	5600
97	730	121	1450	145	2900	169	5800
98	750	122	1500	146	3000	170	6000
99	775	123	1550	147	3075	171	6150
100	800	124	1600	148	3150	172	6300
101	825	125	1650	149	3250	173	6500
102	850	126	1700	150	3350	174	6700
103	875	127	1750	151	3450	175	6900

D. ANEXO D: ESPECIFICACIONES DE MAQUINA

Tabla 15: Especificaciones de la maquina *Endurance*

Machine Type: TJR2TBY
Machine Application: For Truck and Bus Tire Endurance/High Speed Test
Technical Parameters:
Structure: Horizontal Machine Frame
Test Position: 2
Drum Diameter * Width: Φ 1707.6 \times 500mm
Drum Radial Run-Out: \leq 0.20 mm
Drum Lateral Run-Out: \leq 0.20 mm
Residual Imbalance: $<$ 5g
Tire Shaft (Spindle): Double-Hold Style
Tire Diameter: Φ 700mm Φ 1400mm
Tire Max. Section Width: $<$ 450mm
Test Speed: Max.160km/h, Accuracy: 0 +2 km/h
Radial Load: Max.150 kN, Accuracy: \pm 1 %P.S.
Loading Means: Hydraulic Servo in Closed-Loop
Motor Power: 160 kW AC Inverter (or DC)
Dimension: Approx. L8000mm*W1900mm*H3200mm
Total Weight: Approx. 17MT

Tabla 16: Especificaciones de la maquina *Rolling Resistance*

Machine Type: TJRRRTBY
Machine Application: Used for Measuring the Value of TB Tire Rolling resistance (Force Method)
Technical Parameters:
Structure: Horizontal Machine Frame
Test Position: 1
Drum Diameter * Width: Φ 1707.6 * 500 mm
Drum Radial Run-Out: $\leq 0.15\text{mm}$
Drum Lateral Run-Out: $\leq 0.15\text{mm}$
Residual Imbalance: $< 5\text{g}$
Tire Diameter: Φ 550mm Φ 1400mm
Tire Max. Section Width: $< 450\text{mm}$
Tire Roaded Radius: 250 700 mm
Rim Size: 15- 24.5"
Camber Angle: $\leq 0.11^\circ$
Slip Angle: $\leq 0.06^\circ$
Caster Angle: $\leq 0.06^\circ$
Tire Spindle Force Accuracy: $\pm 1.0\text{ N}$
Cr Standard Deviation σ m: $\leq 0.5\text{ N/kN}$ (Measured Values of 3 Times)
Test Speed: Max.150km/h, Accuracy: $\pm 0.5\text{ km/h}$
Test Load: Max.45 kN, Accuracy: $\pm 45\text{N}$
Loading Means: Hydraulic Servo in Closed-Loop
Motor Power: 75 kW DC
Dimension: Approx. L6500mm*W2100mm*H2050mm
Total Weight: Approx. 12MT