



Economía: teoría y práctica

ISSN: 0188-3380

ISSN: 2448-7481

Universidad Autónoma Metropolitana, a través de la  
Unidad Iztapalapa, la Unidad Azcapotzalco y la Unidad  
Xochimilco, División de Ciencias Sociales

Lara, Arturo; Chávez Meza, Ricardo Artemio; Jaimes Gutiérrez, Ma. Guadalupe  
De simple a complejo: el caso del sistema de aceleración electrónica en los automóviles

Economía: teoría y práctica, núm. 50, 2019, Enero-Junio, pp. 37-62

Universidad Autónoma Metropolitana, a través de la Unidad Iztapalapa, la  
Unidad Azcapotzalco y la Unidad Xochimilco, División de Ciencias Sociales

DOI: 10.24275/ETYPUAM/NE/502019/Lara

Disponible en: <http://www.redalyc.org/articulo.oa?id=281161618003>

- ▶ Cómo citar el artículo
- ▶ Número completo
- ▶ Más información del artículo
- ▶ Página de la revista en redalyc.org

## De simple a complejo: el caso del sistema de aceleración electrónica en los automóviles\*

### From Simple to Complex: The Case of Electronic Acceleration System in Automobiles

*Arturo Lara \*\*, Ricardo Artemio Chávez Meza\*\*\*y Ma. Guadalupe Jaimes Gutiérrez\*\*\**

#### RESUMEN

Desde la teoría de los sistemas complejos (Arthur, 2009), el presente trabajo tiene como objetivos: 1) analizar el cambio tecnológico desde la perspectiva de la familia de fenómenos que explota el sistema de aceleración; 2) explicar cómo un sistema sencillo se convierte en uno complejo; 3) reconocer la importancia de reglas, normas y estrategias utilizadas por los agentes para producir nuevas tecnologías, y 4) estudiar el problema de la aceleración no-intencional.

**Palabras clave:** Automóvil, aceleración, tecnología, complejidad, recombinación, instituciones.

**Clasificación JEL:** O, O3, 031, 033.

#### ABSTRACT

From the theory of complex systems (Arthur, 2009), this paper aims to: 1) analyze technological change from the perspective of the phenomena that exploits the acceleration system; 2) explain how a simple system becomes a complex one; 3) recognize the importance of rules, norms and strategies used by agents in order to produce new technologies, and 4) study the problem of unintentional acceleration.

**Keywords:** Automobile, acceleration, technology, complexity, recombination, institutions.

**JEL Classification:** O, O3, 031, 033.

---

\* Fecha de recepción: 03/05/2018. Fecha de aceptación: 10/01/2019.

\*\* Universidad Autónoma Metropolitana, Unidad Xochimilco, México. E-mail: alara@correo.xoc.uam.mx. ORCID: 0000-0001-9699-4393.

\*\*\* Universidad Autónoma Metropolitana, Unidad Xochimilco, México. E-mail: artemio79@hotmail.com. ORCID: 0000-0002-7902-865X.

\*\*\*\* Universidad Autónoma Metropolitana, Unidad Xochimilco, México. E-mail: lupoxa@gmail.com. ORCID: 0000-0003-1646-1111.

## INTRODUCCIÓN

El sistema de aceleración electrónica se considera un componente clave del sistema de seguridad de un automóvil y, como tal, está sujeto a regulación (DOT, 2011). Por esta razón, durante las dos últimas décadas, el problema de la aceleración no-intencional ha sido un tema ampliamente debatido y examinado desde múltiples perspectivas (DOT, 2011; Nasa, 2011; National Research Council, 2012; Kirchhoff y Peterman, 2010; Koopman, 2014; Barr, 2013a, 2013b).

Una parte de la controversia reside en definir dónde inicia y termina este sistema tecnológico. Representar al dispositivo como perteneciente a un sistema jerárquico conlleva implicaciones legales y de regulación pública que van más allá del análisis académico. Por esta razón, este trabajo contribuye a la explicación de la evolución tecnológica del sistema de aceleración y específicamente al problema de la aceleración no-intencional. En el estudio del cambio tecnológico existen diversos enfoques. Predominan tres:

(i) *Gradualismo*. Señala que las tecnologías surgen y evolucionan como resultado de un proceso de variación de las viejas tecnologías y de la selección de las más aptas. Donde la acumulación constante de variaciones y mejoras en las características “físicas” de las herramientas y tecnologías da lugar a todas las modificaciones más importantes de la estructura. Los cambios graduales son resultado de modificaciones deliberadas y de manipulación accidental. (Basalla, 1988; Nelson y Winter, 1982).

(ii) *Combinación/Acumulación*. Éste señala que las tecnologías surgen y evolucionan como nuevas combinaciones, que se construyen a partir de las tecnologías existentes. Las nuevas tecnologías, a su vez, se acumulan y están disponibles como elementos básicos (bloques de construcción) para el desarrollo de otras nuevas tecnologías. (Schumpeter, 1934; Fleming y Sorenson, 2004; Henderson y Clark, 1990; Levinthal, 1997; Rosenberg, 1982; Von Hippel, 1988).

(iii) *Explotación de fenómenos naturales*. Arthur (2009) señala que el origen de las nuevas tecnologías es un proceso que vincula: una necesidad<sup>1</sup> con un fenómeno,<sup>2</sup> para cumplir un propósito.<sup>3</sup> Pero los fenómenos raramente se pue-

---

<sup>1</sup> Esta necesidad procede de que los métodos existentes no son satisfactorios, lo cual obliga a buscar: *a)* un nuevo principio (la idea de un fenómeno en acción), o *b)* un nuevo fenómeno para el que se asocia algún principio. (Arthur 2007: 275).

<sup>2</sup> Los fenómenos son cualquier efecto (confiable y repetible) de la naturaleza, y como tales existen independientemente de los seres humanos y de la tecnología. Los fenómenos pueden ser físicos, o lógicos, o de comportamiento u organización; cualquiera que sea el caso, pueden ser aprovechados para su uso. (Arthur 2009: 49).

<sup>3</sup> El propósito puede ser explícito (*e.g.* una central eléctrica suministra electricidad) o puede ser brumoso, múltiple y cambiante (*e.g.* una computadora no tiene un propósito único y explícito).

den usar en forma cruda. Éstos deben ser explotados a través de un principio.<sup>4</sup> A su vez, el principio debe traducirse (es decir, aprovechar y configurar las ideas conceptuales) en una arquitectura física de trabajo. Aquí es donde entra la combinación de tecnologías existentes. La arquitectura consiste en combinar partes organizadas en sistemas de componentes o módulos. Algunas de éstas forman el conjunto central, que lleva a cabo el propósito base; otras forman un conjunto de subsistemas o submódulos que apoyan al central (es decir, cuidan su funcionamiento, regulan sus funciones, suministran energía y realizan otras tareas). Todas estas partes están en constante interacción (comunicación constante) entre sí: el *output* de unos se convierte en el *input* de los demás.

La versión inicial de una nueva tecnología es cruda, en los primeros días es suficiente que trabaje. Pero, con frecuencia, la configuración de los componentes muestra deficiencias o fallas inesperadas (por ejemplo, pueden no funcionar como se esperaba, o no funcionar en absoluto; pueden utilizar más peso o energía, e incluso tener un costo más alto del que se había visualizado). Así, la tecnología emergente comienza un largo camino de desarrollo (es decir, debe mejorarse, ampliarse y aplicarse eficazmente a diferentes propósitos). Su rendimiento se presiona para entregar más, mejorar su arquitectura, afinar y equilibrar sus componentes.

Los desarrolladores pueden superar las deficiencias inesperadas (*i. e.* las limitaciones): *a*) realizando mejoras graduales en un componente específico; *b*) reemplazando el componente impedido por uno que funcione mejor; *c*) añadiendo un conjunto (o sistemas de piezas) que mejore su funcionamiento básico, le permita supervisar y reaccionar a circunstancias cambiadas o excepcionales, lo adapte a una gama más amplia de tareas, y realce su seguridad y confiabilidad.

Este artículo se inscribe en el programa teórico desarrollado por Arthur (2009). Para comprender la gramática profunda del cambio tecnológico es necesario tomar en cuenta de qué manera los agentes explotan fenómenos naturales con un propósito: producir una función; y en la medida que la ciencia y la ingeniería comprenden y explotan los fenómenos, los sistemas se vuelven estructuralmente más profundos.<sup>5</sup> Esta conceptualización permite identificar los bloques de construcción utilizados por los agentes para construir nuevas tecnologías, y explicar la emergencia de la complejidad.

<sup>4</sup> Un principio es meramente la idea o el concepto de explotar algún fenómeno (o conjunto de fenómenos) para un propósito: *the method of the thing*. Procede “siempre” de un fenómeno, porque una tecnología que no explota nada, no podría lograr nada. Existen muchos principios en el mundo de los seres humanos. (Arthur 2009: 49). Así, por ejemplo, el principio de un reloj es contar los latidos de alguna frecuencia estable; el principio del radar es enviar ondas de radio de alta frecuencia y detectar objetos distantes analizando las reflexiones de estas señales de las superficies de los objetos.

<sup>5</sup> Es decir que al interior de un sistema emergen nuevos subsistemas (Arthur, 2009).

El presente trabajo tiene como objetivos: 1) analizar el cambio tecnológico desde la perspectiva de la familia de fenómenos que explota el sistema de aceleración; 2) explicar cómo un sistema sencillo se convierte en uno complejo; 3) reconocer la importancia de reglas, normas y estrategias utilizadas por los agentes para producir nuevas tecnologías, y 4) estudiar el problema de la aceleración no-intencional.

El trabajo está estructurado de la siguiente manera. En primer lugar, se describe la evolución del carburador y el sistema de aceleración asociado. A continuación, se estudian los factores que provocan la transición del carburador al inyector de combustible electrónico (EFI).<sup>6</sup> Se explica el papel que cumplen las unidades de control electrónica (ECU),<sup>7</sup> los sensores, actuadores, y protocolos de comunicación (*buses*) en el funcionamiento del EFI. Con esta información se analiza cómo la microelectrónica transforma las interfaces que permiten la interacción entre conductor y sistema de aceleración, así como la comunicación y el control entre los distintos sistemas de un vehículo. Gracias al *software* es posible crear formas de control avanzadas del sistema de aceleración. De esta forma emerge el sistema de aceleración electrónica. Finalmente se examina el caso de la aceleración no-intencional del Camry-2005 de Toyota.

## I. CARBURADOR: EFECTO VENTURI Y PROFUNDIZACIÓN ESTRUCTURAL

En gran medida, el diseño del carburador, a inicios del siglo XVIII, es posible gracias a la obra del matemático y físico Bernoulli (1700-1782) quien desarrolla de manera formal y deductiva la teoría de la hidrodinámica (1738). Con base en los principios de Bernoulli y en el principio de continuidad de la masa, el físico Venturi (1746-1822) logra identificar en 1797 el llamado efecto Venturi (Reif, 2015). Este efecto se convierte en el principal bloque de construcción del carburador: permite mezclar aire y gasolina necesarios para alimentar el motor de combustión.

Sobre la base de teorías, modelos y lenguaje provisto por la física y las matemáticas, los inventores diseñan y mejoran una amplia variedad de carburadores. Son destacables dos de ellos. En 1838, Barnett diseña el carburador tipo “mecha” para motor de gasolina. En tanto que, en 1882, Marcus construye el carburador tipo “superficie”, elegido por Nicolaus Otto (1832-1891) como

---

<sup>6</sup> EFI: *Electronic fuel injection*.

<sup>7</sup> ECU: *Electronic control unit*.

carburador para su motor de combustión interna de cuatro tiempos.<sup>8</sup> Este motor se convierte, en la naciente industria automotriz, en el diseño dominante (Reif, 2015).

En el periodo formativo de la industria automotriz, fines del siglo XIX y principios del siglo XX, la mezcla de aire y gasolina se opera manualmente a través de una válvula de estrangulamiento. El conductor tiene que conocer cuándo y cómo operar la válvula dependiendo de las condiciones del motor.<sup>9</sup> Durante este periodo se reconocen en particular, tres mejoras en el diseño del carburador: 1) Benz, en 1885, introduce un flotador para mantener el nivel de combustible a la misma altura. 2) Maybach, en 1893, añade una boquilla para provocar la distribución uniforme del combustible. 3) Entre 1906 y 1907, la empresa Zenith diseña un dispositivo auxiliar para que, a pesar de la velocidad del aire, se suministre una mezcla de aire/combustible inalterable (Reif, 2015).

Con el ascenso del motor de combustión de cuatro tiempos se integran nuevos componentes y subsistemas en el carburador. Este proceso de mejora y profundización estructural del carburador es parte de un proceso co-evolutivo de distintos nichos de mercado: barcos, aeronaves, vehículos militares, vehículos convencionales y deportivos, etcétera. Cada nicho de mercado presiona selectivamente determinadas formas de explotar el efecto Venturi.<sup>10</sup> Usualmente las mejores soluciones se concentran en bloques de construcción o módulos para luego reutilizarse. Es así como el inyector de combustible, inicialmente diseñado para motores de barcos, se adopta posteriormente en los aviones, camiones de *diesel* y finalmente, con la revolución de la microelectrónica, se integra en los vehículos de gasolina convencionales (Aird, 2001).

Aunque el proceso inventivo resulta intenso, existen problemas irresolubles dada la naturaleza de los fenómenos que explota el efecto Venturi. Por ejemplo, en la medida que la gravedad es una fuerza que ayuda al funcionamiento del carburador ¿cómo resolver el problema de que se apague o ahogue el motor cuando el vehículo se encuentra en una pendiente? Pero su mayor limitación se relaciona con sus niveles de ajuste de combustible rudimentarios, así como la inexistencia de un control preciso de la cantidad de combustible necesaria para cada estado del motor de combustión (Borg, 2010).

---

<sup>8</sup> Este logro es posible dado que Otto diseña, por primera vez y de manera exitosa, un motor de combustión con cuatro principios de funcionamiento: admisión, compresión, combustión y escape (Reif, 2015).

<sup>9</sup> Encender el motor en frío o cuando se encuentra caliente y en marcha, etcétera.

<sup>10</sup> Por ejemplo, las condiciones ambientales (altura y baja temperatura) de vuelo de las aeronaves podían congelar el combustible localizado en la cámara del carburador. En este nicho de mercado era clave resolver este problema (Thorner, 1946).

De esta manera, se resuelve un problema mejorando lo existente o creando nuevos subsistemas. El resultado es la emergencia de un sistema jerárquico anidado: sistemas dentro de sistemas. A medida que se mejora el carburador, no obstante, se vuelve más pesado, complejo y costoso; siendo además no confiable pues requiere continuo mantenimiento (Abernathy, 1979). Pero sobre todo, es un sistema ineficiente, que desperdicia combustible y contamina el ambiente (Clarke, 2007).

### I.1. *Interfaces y falla del carburador*

¿Cómo se vinculan el conductor y el sistema de aceleración? La interfaz entre el pedal de aceleración y el carburador está integrada por ejes, soportes, cables y resortes. Explotando principios mecánicos simples, el pedal del acelerador, que presiona el conductor, produce una función relativamente simple: activar el proceso de aceleración.

La naturaleza de las interfaces metálicas mejoradas durante decenios en la industria automotriz, ha resultado en un dispositivo resistente, relativamente libre de fallas (Clarke, 2007). Pero, aun en el caso de producirse una falla —ocasionada por desgaste, rotura u oxidación de partes metálicas— se requiere una capacidad humana normal para diagnosticar el problema: el fenómeno es visible a simple vista. Una falla mecánica tiene propiedades tangibles: deja huella y resulta relativamente fácil su identificación y reparación (de bajo costo). Y, a diferencia de interfaces electrónicas, carece de mecanismos para que la falla se propague, como se verá más adelante (Isermann, 2011).

Cabe mencionar que, hasta principios de la década de 1960, el principal detonante del cambio tecnológico se origina en los cuellos de botellas de naturaleza tecnológica o en la necesidad de mejoras de diseño provocadas por la competencia (Porter, 1983). Desde la segunda mitad de la década de 1960 en adelante, emergen nuevas reglas institucionales que orillan a las empresas a concentrarse en un conjunto de fenómenos interdependientes, como se examina a continuación.

## II. CAMBIO INSTITUCIONAL Y NECESIDADES DE NUEVAS FORMAS DE CONTROL

En 1961, los legisladores del estado de California (EEUU) aprueban un conjunto de nuevas reglas específicas para los automóviles. Éstas tienen como objetivo fijar límites máximos de emisiones contaminantes, así como establecer reglas de desempeño y de vida útil del equipo asociado con el control y la reducción

de contaminantes. La EPA<sup>11</sup> busca que el sistema de control de contaminantes utilizado por un automóvil (entre ellos el carburador), deberá estar libre de mantenimiento, por lo menos durante cinco años o 50,000 millas. Asimismo, establece que aquellas empresas que no cumplan con esta regla no estén autorizadas a producir vehículos en los Estados Unidos. La EPA amenaza a las empresas del sector automotriz con detener las líneas de ensamble de vehículos que no cumplan la norma (Doyle, 2000). En 1988, en el marco de la campaña por reducir la contaminación, la CARB<sup>12</sup> establece que los vehículos deben incluir un sistema de diagnóstico a bordo (OBD)<sup>13</sup> para determinar anomalías de los dispositivos relacionados con el control de la contaminación. En el crecimiento de la complejidad del sistema de control, cumplen un papel relevante las instituciones.

## ***II.1. Construir un sistema de control de contaminación que se retroalimente***

De tal suerte que, empujadas por las instituciones reguladoras, las empresas encuentran que un vehículo tiene no una, sino múltiples fuentes de contaminación. Para mejorar este sistema es necesario formular preguntas y respuestas cada vez más complejas. ¿Cuáles son las fuentes de contaminación y cuáles son las soluciones? *i)* Para reciclar el hidrocarburo no quemado de la caja del cigüeñal se añade una válvula de ventilación positiva del cárter (PCV)<sup>14</sup> al motor de combustión. *ii)* Para reciclar los vapores del tanque de combustible se integra un nuevo sistema (una cámara de aire, entre otros componentes). *iii)* Para evitar que el carburador acumule hidrocarburo, cuando el vehículo se apaga, se agrega un sistema para drenarlo, y reciclar el combustible al colector de admisión. *iv)* Para evitar la incompleta combustión, que resulta del encendido en frío, se añade un nuevo sistema (válvula) que precalienta el aire del carburador. *v)* Para disminuir la emisión de los tres principales gases nocivos (CO, HC y NOX), que expulsa el tubo del escape del motor de combustión, se integra, en 1975, el convertidor catalítico de tres vías (Doyle, 2000).

Sin embargo, lo que pronto se descubre es que, para garantizar un desempeño eficiente y prolongar la vida útil del convertidor catalítico, la mezcla debe mantenerse muy cerca de la relación ideal: 14.7 partes de aire (A), por 1 de gasolina (G) (Halderman y Linder, 2006). En particular, se debe evitar una mezcla

---

<sup>11</sup> EPA: Environmental Protection Agency.

<sup>12</sup> CARB: California Air Resources Board.

<sup>13</sup> OBD: On-board diagnosis.

<sup>14</sup> PCV: Positive crankcase ventilation.

demasiado rica en gasolina, puesto que genera altas temperaturas que pueden quemar al convertidor catalítico (Isermann, 2014; Aird, 2001).<sup>15</sup> Se requiere diseñar un sistema en el que el motor y el carburador se comuniquen.

La solución, como pronto descubren los ingenieros, no puede depender de un solo subsistema, son necesarias formas avanzadas de comunicación y retroalimentación entre distintos subsistemas (Reif, 2015). Esta necesidad desemboca en el diseño e integración de un sensor de oxígeno, en 1976, entre el motor y el convertidor catalítico, cuya función es enviar información sobre la calidad de la mezcla aire/gasolina (A/G) a una computadora central, para que ésta, a su vez, mediante un actuador, ajuste la mezcla de A/G del carburador (Aird, 2001). De esta forma, por un lado, el motor funciona con la mezcla adecuada y por otro se protege al convertidor catalítico.<sup>16</sup> El resultado: un sistema más conectado, interdependiente, pero sobre todo jerárquico (Isermann, 2014). Así pues, los cambios que sufre el vehículo ponen en el centro, no sólo el tema de la contaminación sino, de manera concomitante, la naturaleza de los principios de control del vehículo.

## II.2. Carburador híbrido

Inicialmente se considera al dominio de la electrónica como un complemento, útil para mejorar el carburador, mas no como una tecnología rival. Desde esta perspectiva, se construyen distintos sistemas híbridos que combinan carburador, inyección de combustible y retroalimentación electrónica (Porter, 1983). El carburador híbrido, mecánico-electrónico, no tarda, sin embargo, en poner en evidencia lo difícil que resulta construir interfaces entre dominios diferentes: mecánicos y electrónicos.<sup>17</sup> No cabe duda de que esta novedosa forma híbrida de regulación mecánica y electrónica conduce a redefinir el espacio de búsqueda de nuevas soluciones. No se trata ahora de ajustar de manera independiente el desempeño del carburador. Es necesario comprender la interdependencia del carburador con otros subsistemas de un vehículo.

Por ello, aun cuando el carburador se mejora con componentes electrónicos, resurgen las viejas preguntas que permanecen insolubles. A pesar de todas

---

<sup>15</sup> El exceso de plomo en la gasolina también afecta sensiblemente el desempeño y la vida útil del convertidor catalítico. Por esta razón, el congreso de los EEUU aprueba una ley cuyo objetivo es establecer un límite máximo de plomo en la gasolina (Mondt, 2000).

<sup>16</sup> A mediados de 1975 General Motors introduce el convertidor catalítico (Mondt, 2000).

<sup>17</sup> Particularmente cuando se busca medir el flujo de aire combinando dos señales separadas. “*The design of the airflow meter itself might also be regarded as imperfect, depending as it does on a combination of electronic and mechanical components (...) (A) mechanical device operating in conjunction with any electronic device often proves to be a weak spot*” (Aird, 2001:38).

las mejoras, la tecnología del carburador está limitada por la forma cómo se encapsula y explota el efecto Venturi. La limitación central del carburador es la imposibilidad de medir y controlar de manera precisa la mezcla, volumen y ritmo de la gasolina y del aire, y mucho menos sincronizar la alimentación de combustible con el sistema de ignición. La naturaleza del efecto Venturi impide cualquier proceso de optimización rigurosa. Se requerían nuevas formas de control de fenómenos turbulentos e inestables. Todo el esfuerzo de diferentes generaciones de ingenieros por mejorar el diseño del carburador había llegado a un límite. El carburador se había convertido en una tecnología obsoleta (Reif, 2015; Abernathy, 1979; Borg, 2010).

Por ello se requiere no sólo de nueva tecnología, sino de nuevas capacidades. A nivel de ciencia básica se vuelve necesario comprender los principios de la física y la química de la atmósfera. A nivel de ciencia e ingeniería aplicada, es indispensable crear nuevos métodos e instrumentos para estudiar cómo un vehículo genera gases nocivos. Por lo tanto, a partir de estos resultados, es necesario modificar la configuración del motor, creando nuevas formas de control y reducción de contaminantes (National Academy of Sciences, 1992).

### *III.3. El sistema de inyección electrónica de combustible (EFI)*

La tecnología del EFI se viene experimentando desde los años veinte del siglo pasado. Se adoptó, con éxito relativo, en vehículos de carreras, vehículos de lujo y en aeronaves (Reif, 2015). En tanto que su adopción en el sector automotriz es relativamente más reciente (Aird, 2001). El sistema EFI cumple con las mismas funciones del carburador: realizar la mezcla de aire/combustible (A/C) y regular la cantidad de combustible que alimenta al vehículo. Sin embargo, su operación y los principios tecnológicos subyacentes son distintos.

Lo que interesa a continuación es comprender cómo la tecnología del EFI explota fenómenos naturales para efectuar su función de control. Por ello es necesario describir: (i) los sistemas de recuperación de información del sistema (sensores); (ii) los sistemas de procesamiento de información (ECU's); (iii) los canales de comunicación (*buses*), y (iv) los sistemas de respuesta (actuadores). A continuación, se examinan cada uno de estos sistemas.

#### Sensores electrónicos

La comprensión científica e ingenieril cada vez mayor de los procesos relacionados con el motor de combustión han conducido a demandar formas rigurosas de

control. Por ello, los ingenieros y científicos exploran qué tipo de fenómenos naturales y, por ende, qué principios son susceptibles de transformarse en instrumentos de medición. Por ejemplo, para medir y controlar procesos termodinámicos es necesario contar con instrumentos de medición precisos, veloces, confiables, que se desempeñen en ambientes turbulentos, que no se contaminen, etcétera. De modo que, basados en principios, se han desarrollado una clase amplia de sensores, con un propósito: obtener, medir y convertir la información de los fenómenos naturales en información digital susceptible de alimentar a una computadora. En esta búsqueda la tendencia es explotar sensores con principios que operan a escala cada vez más pequeña y profunda: micro y nano escala (Walter, Lin y Pisano, 2007). Desde esta perspectiva, el desarrollo de los sensores se caracteriza por procesos de profundización estructural significativos.

El objetivo de los sensores integrados en el sistema EFI es generar información necesaria para controlar las distintas funciones del sistema. Es posible clasificar a los sensores de acuerdo con sus funciones desempeñadas en el sistema EFI: determinar la posición, temperatura, presión, flujo, rotación o vibración de los componentes del sistema. Para ello, los diversos materiales empleados por los sensores utilizan distintos principios físicos, ópticos, químicos y eléctricos.

### Sistemas de procesamiento de datos (ECU's)

La necesidad de conectar subsistemas empuja a las empresas del sector automotriz a reciclar bloques de construcción empleados en la industria electrónica y de cómputo. Sin embargo, si bien los componentes electrónicos que se utilizan en un vehículo son similares a los que integran computadoras, debido al ambiente en el que trabajan, son mucho más exigentes en los primeros. Un vehículo requiere materiales y sistemas de protección para resistir ambientes hostiles.<sup>18</sup> Estas condiciones específicas de un vehículo obligan a las empresas a construir diseños más robustos que los utilizados por la industria de cómputo (Porter, 1983).

Los primeros *chips* semiconductores que se utilizan en un vehículo contenían un transistor cada uno. Sin embargo, gracias a las nuevas tecnologías desarrolladas en la industria de cómputo se logra diseñar y producir ECUs que integran miles de semiconductores en un espacio reducido. Con esta tecnología para controlar el sistema de ignición del motor, en 1977, General Motors lanza al mercado un vehículo que incluye una microcomputadora análoga que contiene

<sup>18</sup> Tales como: las altas temperaturas del motor (hasta 200°C), exposición al agua y a la humedad (hasta 90 por ciento), vibraciones, daños causados por la exposición a sales, fuertes fluctuaciones de voltaje; radiaciones e interferencia electromagnética (Bereisa, 1983).

más de 20,000 transistores y otros componentes electrónicos (IEEEXplore, 1976). Este ECU contiene sólo dos tablas de búsqueda. En 1981, la microcomputadora de un vehículo de General Motors contiene más de 6,800 transistores y más de 50,000 líneas de códigos fuente (Bereisa, 1983). Hoy en día, *chips* avanzados para vehículos pueden contener más de 7,000 millones de transistores (Hennessy y Patterson, 2017).

Cada uno de los componentes de un ECU aprovecha un efecto natural para producir una operación lógica o matemática. La complejidad es el resultado de cómo se explota la interacción de cientos de millones de efectos naturales, distribuidos en una red de transistores, diodos y capacitadores, montados en un circuito impreso. Para explotar esos efectos se requiere una estructura compleja, física y comportamental del *hardware* y *software*, denominado *firmware* (Wolf, 2006).

La evolución tecnológica de los ECU's es, en gran medida, resultado de los avances en la física del estado sólido (las propiedades de los semiconductores) y de la investigación de la mecánica cuántica (Kato, 2014). Estos avances, en la comprensión de los principios de la materia y de la energía, contribuyen al diseño de ECU's poderosos, veloces y multifuncionales. La tabla 1 resume la evolución de las características de los ECU's relacionados con el inyector electrónico de combustible (EFI).

Tabla 1. Evolución de las características de los ECU's del inyector de combustible (EFI).

Característica	<1975	1975—<1990	1990—1995	>1995
Tipo de procesador	Analógico	Digital de 4 a 8 bits	Digital 16 bits	Digital de 32 a 64 bits
Número de procesadores	1	1	De 1 a 2	De 2 a 4
Velocidad de procesamiento	No aplica	De 0.5 a 1 MIPS <sup>1</sup>	De 1.2 a 10.2 MIPS	De 10.5 a 300 MIPS
Memoria de programación	Sin memoria	Mascara de ROM <sup>2</sup>	EP-ROM <sup>3</sup> / EEPROM <sup>4</sup>	Flash ROM
Capacidad de memoria ROM	4 kB	De 4 a 64 kB	De 64 a 128 kB	De 1 MB a >2 mb
Tablas de búsqueda	0	3-5	10-25	80-120
Líneas de código	0	De 4 mil a 100 mil	De 100 mil a 500 mil	De 1 millón a 100 millones

<sup>1</sup> MIPS: Millones de instrucciones por segundo.

<sup>2</sup> ROM: Read-Only Memory.

<sup>3</sup> EP-ROM: Erasable Programmable Read-Only Memory.

<sup>4</sup> EEPROM: Electrically Erasable Programmable Read-Only Memory.

**Fuente:** elaboración propia con base en Bonnick (2001); Isermann (2014); Kato (2014), Niimi (2014), ambos en Crolla *et al.* (2014); Navet y Simonot-Lion (2009); Gaertner (2015).

En el caso del EFI, los primeros ECU's realizan una única función de acuerdo a los datos proporcionados por una única variable. Con el tiempo, sin embargo, se integran más funciones y variables de control en un ECU. El ECU de un EFI es alimentado por entre 15 a 25 sensores que generan de seis a ocho variables principales a optimizar (Isermann, 2014). Los millones de datos enviados desde los sensores requieren ser procesados en milésimas de segundo. En el periodo 1975-1989, la velocidad de procesamiento es de 0.5–1 MIPS, en tanto que, a partir de 1996 en adelante, ésta se incrementa de 10.5 a 300 MIPS. De igual manera, se incrementa el número de procesadores y la memoria en ROM en el periodo (*cf.* tabla 1).

El sistema de control opera de dos maneras complementarias: *i)* alimentado con información de los sensores, el ECU ajusta su desempeño, y *ii)* mediante un conjunto de mapas o “tablas de búsqueda” (*look-up tables*), el ECU anticipa o predice el radio de la mezcla a utilizarse. El número de las tablas de búsqueda<sup>19</sup> que permiten el control *feedforward* aumentan de tres a cinco, en el periodo 1975-1989, y de ocho a 120 tablas, en el último periodo (*cf.* tabla 1). De esta forma se establecen de manera flexible y a la vez exacta las necesidades del motor (Isermann, 2014).

Toda esta capacidad de procesamiento de información se vincula con el *software*. En 2015, “[u]n automóvil moderno de gama alta integra alrededor de 100 millones de líneas de código (LOC), y se prevé que, en un futuro cercano, este número crezca a 200-300 millones” (Gaertner, 2015: 6).<sup>20</sup> La pregunta que surge en este punto es ¿cómo explicar esta explosión en el número de códigos fuente, la diversidad de módulos y variables globales?<sup>21</sup>

Se puede considerar que el lenguaje del *software* es cada vez más extenso, en la medida que ciencia e ingeniería han contribuido a mejorar la comprensión de los dominios físicos (mecánica, combustión, termodinámica, electricidad, etcétera) explotados por el motor de combustión. Cada grupo de partes del motor aprovecha un conjunto acotado de dominios tecnológicos. Así, a mayor diversidad de dominios le corresponderá un mayor número de códigos fuente.

---

<sup>19</sup> Las tablas de búsqueda son modelos matemáticos codificados en el *software* que representan las curvas de desempeño de las variables del sistema y que permiten su control. Estas tablas permiten el control basado en predicciones (*feedforward*) del sistema.

<sup>20</sup> Un millón de líneas de código, si se imprimen, cubrirían aproximadamente 18,000 páginas de texto. Se estima que el costo de una línea de código es mayor a los 10 \$US (Arzen y Maggio, 2017).

<sup>21</sup> Variable global: “Any variable with a scope that makes it available to the entire program” (Ganssle y Barr, 2003:116). Es un ingrediente de una receta que es utilizado en múltiples recetas. Son archivos vinculados a tareas específicas. Por ejemplo, una tarea global establece la apertura que debe tener la mariposa del acelerador; o el máximo de velocidad permitido.

Desde esta perspectiva, para explotar efectos naturales es necesario que los modelos internos (*software*) que tiene una computadora sean isomorfos respecto al mundo físico. Es decir que la estructura del modelo lógico u algoritmo matemático contenido en el lenguaje de la computadora represente o guarde cierta correspondencia con la estructura de este mundo. Ahora bien, si se compara la riqueza de los modelos internos integrado en el *software* de un vehículo de los años setenta, respecto a uno actual, la brecha es sustancial. Lo primero que salta a la vista del *software* de un vehículo reciente no es que contenga más *software* como resultado del crecimiento exponencial de aplicaciones. Sino que, en un mismo dominio, gracias a que la ciencia y la tecnología nos permiten operar en múltiples niveles de la naturaleza, se han desarrollado también múltiples instrucciones asociadas a cada nivel. Por esta razón en esa misma dirección se requieren poblaciones de reglas “si-entonces” cada vez más específicas, numerosas (Holland, 2004). En el lenguaje de la ingeniería del *software* esto puede expresarse en un mayor número de variables globales y locales (Isermann, 2014).

Existe otra fuente de complejidad. Ésta se relaciona con la emergencia de una clase de *software* especial para lidar con los fallos del *software*: *i*) creando y aumentando la capacidad de formas de autodiagnóstico a bordo del vehículo (OBD);<sup>22</sup> *ii*) ampliando y mejorando la función de una grabadora de datos de eventos (EDR)<sup>23</sup> para que, en caso de accidentes, permita identificar fallas (DOT, 2008); *iii*) creando *software* avanzado para controlar y reducir fallas críticas del *software*, incluyendo redundancia física (dos sensores, en vez de uno) u analítica (procesadores poderosos con holgura suficiente para procesar información) que disminuyan la vulnerabilidad del sistema; *iv*) diseñando capacidades a nivel de la red de componentes electrónicos para que los recursos distribuidos en su conjunto sean los que absorban la falla (Bergmiller, 2015). Todas estas demandas han resultado en la construcción de un *software* cada vez más complejo y extenso. Esto explica por qué el *software* ha evolucionado de cero a millones de líneas de código.

---

<sup>22</sup> Las funciones relacionadas con el OBD demandan aproximadamente el 50 por ciento de la capacidad del ECU (Reif, 2015). La versión más elaborada, conocida como OBD-II –y adoptada en los EEUU entre 1996 y 1999–, contiene 48 sensores incrustados en cada componente y subsistema del motor (Borg, 2010).

<sup>23</sup> EDR: *Event data recorders*.

## Cableado, arneses y buses

La naturaleza de los procesos de retroalimentación y control de un vehículo *requieren* formas avanzadas de conectividad. Por esta razón, el sistema de EFI ha transitado del uso de cables de cobre pesados a sistemas multiplexados —de fibra óptica— que permiten transferir múltiples señales, así como grandes volúmenes de información y energía. La cantidad de información que transita de los sensores a los procesadores de datos y de éstos a los actuadores en el EFI es de 300 MIPS (Isermann, 2014: 11). Los *buses* son los encargados de controlar y dirigir la información de un punto a otro. Los *buses* han evolucionado de acuerdo con las necesidades de velocidad diferentes de cada subsistema y/o componente.

## Actuadores electrónicos

Los actuadores son los dispositivos finales del control electrónico que regulan los principales flujos de materia y energía del sistema. En el caso del EFI, se tratan de válvulas, bombas, ventiladores o interruptores coordinados por el módulo de control del motor (EMC) y que en conjunto optimizan el estado del sistema. Los actuadores operan de la siguiente manera: *a*) por medio de los transductores de señales (dispositivos electrónicos que decodifican la información enviada desde el EMC) determinan la acción a ejecutar; *b*) a través de energía (eléctrica, mecánica, hidráulica o térmica) adquirida de una fuente instalada en el vehículo (batería, generador, motor, etcétera) realizan la acción determinada; *c*) con base en múltiples principios (electrodinámicos, electromagnéticos, térmicos, hidráulicos o neumáticos), ejecutan la acción.<sup>24</sup> En síntesis, los actuadores convierten las señales eléctricas en operaciones que optimizan el desempeño del sistema. Los avances de los actuadores se han concentrado en la velocidad de decodificación de las señales, el ahorro de energía para su operación, la precisión de sus acciones y la velocidad de respuesta (Isermann, 2014).

Con esta información, a continuación se describe la naturaleza del sistema de aceleración electrónica en el caso de Toyota.

---

<sup>24</sup> Regular la presión de distintos fluidos, regular el paso de materia o energía, colocar dispositivos mecánicos en posiciones adecuadas, etcétera.

### III. EL SISTEMA DE ACCELERACIÓN ELECTRÓNICO DE TOYOTA: ARQUITECTURA, COMPLEJIDAD Y FALLAS DEL SOFTWARE

Para examinar este sistema se utiliza el caso del *software* del vehículo Camry-2005 de Toyota.<sup>25</sup> Con esta información es posible identificar: *i*) la arquitectura, complejidad y vulnerabilidad del *software* del sistema de aceleración; *ii*) la relación entre la complejidad y la aceleración no-intencional; *iii*) la necesidad de diseñar *software* de seguridad para proteger el *software*; *iv*) cómo diseñó Toyota este sistema; *v*) el problema de la aceleración no intencional,<sup>26</sup> y *vi*) las consecuencias e implicaciones de la aceleración no intencional.

*i)* Arquitectura del *hardware/software*. El sistema de aceleración electrónica, introducido en 1986 (Isermann, 2014) sustituye los componentes mecánicos por *hardware* y *software* sofisticados (DOT, 2011).<sup>27</sup> Para establecer la velocidad, dos sensores (redundantes) de posición del pedal envían información al “módulo de control del motor” (ECM).<sup>28</sup>

El ECM tiene en su interior, entre otros componentes electrónicos, dos módulos: 1) una “unidad principal de procesamiento central” (Main-CPU),<sup>29</sup> que procesa las operaciones lógicas y matemáticas relacionadas con las funciones claves del motor. El *software* de este módulo tiene el potencial de ser mejorado; 2) un “monitor de la unidad central de procesamiento” (Monitor-CPU),<sup>30</sup> especializado en detectar y corregir errores críticos del *software*. Para lograr que el *software* responda, en tiempo real, a tareas especializadas y predefinidas, en este módulo se han “quemado” millones de *chips* y *software* embebidos. La desventaja es que una vez diseñado no es posible modificar, mejorar u actualizar el *software*. Para mejorar un *software* embebido es necesario rediseñar el *software/hardware* embebido y este es un proceso costoso, lento y complejo (Barr, 2013b).

---

<sup>25</sup> Esta sección se basa en la descripción del *software* por parte de los expertos en *software* embebido, Michael Barr (2013a, 2013b) y Koopman (2014), como parte de la demanda judicial Bookout vs. Toyota.

<sup>26</sup> La aceleración no intencional se refiere a la ocurrencia de cualquier grado de aceleración que se produce en el vehículo sin que el conductor lo cause a propósito. La aceleración no intencional es un concepto amplio que incluye a la aceleración repentina (*sudden acceleration*) —que se refiere a formas de aceleración no intencionales, inesperadas y de alto poder desde una posición estacionaria del vehículo o desde una velocidad inicial muy baja y acompañada de una aparente pérdida de efectividad de los frenos—; incluye también incidentes de aceleración a altas velocidades donde los frenos mantienen total o parcialmente su efectividad; atascamientos del pedal de aceleración (por ejemplo, con los tapetes del piso) y; aperturas menores del acelerador a distintas velocidades. (NHTSA, 2011).

<sup>27</sup> Toyota introduce, en 2002, el control electrónico del acelerador del Camry (Barr, 2013a).

<sup>28</sup> ECM: *Engine Control Modules*. El ECM calcula los niveles óptimos y envía órdenes a los actuadores para sincronizar la chispa, el ingreso de aire y de combustible; así como regular el sistema de control de contaminantes (Isermann, 2014).

<sup>29</sup> Main-CPU: *Main Central Processing Unit*.

<sup>30</sup> Monitor-CPU: *Monitor of Central Processing Unit*.

En el Camry-2005 las funciones del sistema de aceleración se relacionan con todo el sistema del motor porque el conjunto de operaciones del *software* se ejecuta en el mismo procesador: el *hardware* ECM (Barr, 2013a). Sin embargo, utilizar el mismo *hardware* para calcular distintas tareas<sup>31</sup> es, desde el punto de vista del diseño, inapropiado. El estándar E-GAS, adoptado por la industria automotriz pero no así por el Camry-2005 de Toyota, señala que no se puede vigilar el *software* y, en particular, evitar la aceleración no intencional, si no se cuenta con un *chip* externo o módulo independiente (Group E-GAS, 2013). Si la estructura de datos, en especial los críticos, no están protegidos por un *hardware* externo, entonces las fallas aleatorias del *hardware* o del *software*,<sup>32</sup> pueden causar que la memoria del sistema operativo se corrompa, que algunas tareas claves se interrumpan, se propaguen las fallas en el sistema, y se produzca un mal funcionamiento: la aceleración no-intencional (AUTOSAR, 2013).

*ii) Explosión de complejidad.* El *software* del Camry-2005 contiene 11,000 “variables globales”, distribuidas en 4,000 módulos u archivos (*files*). Sólo para calcular el ángulo del pedal del acelerador, el sistema dispone de 12 funciones matemáticas de aproximadamente 1,300 líneas. ¿De acuerdo con las normas de la ingeniería del *software*, es razonable este nivel de complejidad?<sup>33</sup> Los expertos de la industria del *software* señalan que debe existir, no 10,000 ni 1,000 sino un número pequeño de variables globales, en el rango de 11 a 20 de complejidad ciclomática (McCabe, 1976). El Camry-2005 tiene 67 funciones con una calificación superior a 50; 12 de las cuales se encuentran por encima de 100 (Barr, 2013a). Por ende, un nivel insostenible de complejidad ciclomática.

Este nivel de complejidad da lugar a la formación de dos clases de “códigos espagueti” (Koopman, 2014; Barr, 2013a): *i)* cuando existen diferentes recetas o códigos fuente que provienen de variables globales distintas, se incrementa la probabilidad de competencia e interferencia de tareas, lo que puede causar que el sistema operativo y la memoria se saturen, con el resultado de muerte de tareas críticas; *ii)* cuando se vuelve complicado seguir la receta: si la extensión de los códigos fuente, asociados a una función, se extienden por ejemplo 20 páginas, hay mayores probabilidades de que existan errores del *software*. Ambas clases de códigos espagueti generan inestabilidad en el sistema (Koopman, 2014; Barr, 2013a).

<sup>31</sup> Por ejemplo, efectuar en un mismo *hardware* tareas distintas como calcular el ángulo del acelerador, tareas de auto-diagnóstico y el sistema de seguridad de fallas.

<sup>32</sup> Un error del *software* puede ser inocuo u ocasionar la corrupción de regiones enteras de la memoria y del sistema operativo.

<sup>33</sup> Para cuantificar la complejidad lógica de un programa, la comunidad de expertos en ingeniería de *software*, calculan la Complejidad Ciclomática (*Cyclomatic Complexity*) para estimar el riesgo, costo y estabilidad del sistema. McCabe, después de cuantificar distintos softwares, encuentra que un módulo cuya complejidad ciclomática se encuentre en el rango de 21-50, es un programa de alto riesgo. Un programa que tenga más de 50 es un programa que no se puede probar o controlar (McCabe, 1976).

“Al estar integrado por miles de variables, el *software* tiene billones o trillones de estados posibles. Los vehículos de hoy en día son lo suficientemente complejos como para que ninguna cantidad razonable de análisis o pruebas puedan garantizar que la electrónica y el *software* no tengan errores” (Nasa, 2011: 20). Esto es, si se estima con la tasa de fallos de *software* de ultra confianza, es decir, con una tasa de error de entre  $10^{-7}$  y  $10^{-9}$  errores para misiones de cálculo de una a 10 horas, significaría que se requieren de entre  $10^7$  y  $10^9$  horas de prueba para validar la confianza del *software* (Butler y Finelli, 1993). En el escenario más optimista, esto significa que, se requerirían de  $10^7$  horas de prueba, es decir, 1,141.55 años de prueba. Dada esta complejidad del código y del número de variables globales no es posible diseñar *software* libre de errores.

iii) *Software* que proteja al *software*. Dado que no es posible producir un *software* libre de fallas, entonces es necesario crear un *software* que vigile a otro *software*. Por ello Toyota construye un módulo especializado: el “monitor de la unidad central de procesamiento” (Monitor-CPU), cuya función es la de monitorizar el desempeño de funciones críticas del *software*.<sup>34</sup> Incluye un *software* “supervisor vigilante” (*watchdog*) para que en el caso de que el *software* colapse, éste tenga capacidad de reiniciar el sistema. La Nasa (2011) reporta que el vehículo de Toyota tiene, en el Monitor-CPU, cinco sistemas de seguridad de fallas. Sin embargo, Barr (2013a) encuentra que estos sistemas no tienen una segunda copia. Esto es, si se pierde o muere alguna tarea, al no existir “archivo espejo” (*mirror file*) el sistema puede verse amenazado por un malfuncionamiento crítico. Desde el punto de vista del diseño del *software* de seguridad, el Camry-2005 no es un vehículo seguro, como se confirmaría con el problema de la aceleración no-intencional.

Durante el juicio, Toyota reconoce que no tiene el conocimiento sobre cómo se diseñó el *software* de seguridad del sistema de aceleración y, en particular, el módulo Monitor-CPU. Toyota carece de capacidades para diseñar un sistema de *software* que vigile el *software* crítico. Por esta razón, en estas áreas Toyota depende enteramente de sus proveedores (Barr, 2013a). En un e-mail Toyota acepta que: “(...), la tecnología como la ‘prueba de fallos’ (*failsafe*) no es parte del ADN de la división de ingeniería de Toyota” (Koopman, 2014: 46).

iv) ¿Cómo construyó Toyota su Monitor-CPU? A diferencia del pequeño número de agentes involucrados en el diseño del sistema de aceleración mecánico, en el caso del *software* de un automóvil concurren una amplia diversidad de

---

<sup>34</sup> Este sistema cuando detecta fallas genera “códigos de diagnóstico de problemas” (DTC: *diagnostic trouble code*), los cuales permiten identificar y generar respuestas a problemas o malfuncionamiento de algún subsistema.

agentes.<sup>35</sup> Las capacidades para diseñar el *software* de un automóvil se encuentran distribuidas en distintas organizaciones. Por esta razón se crean consorcios para desarrollar lenguajes, tanto como estándares e instrumentos. El sistema operativo OSEK que utiliza Toyota es creado en 1993 por un consorcio de fabricantes de automóviles y universidades: BMW, Bosch, Daimler-Chrysler, Siemens y Volkswagen (Dirk, 1998). A partir de la arquitectura OSEK, el Centro de Ingeniería y Seguridad de NEC diseña el sistema operativo y módulo Monitor-CPU del Camry-2005. Delphi y Toyota contribuyen con subsistemas adicionales y códigos fuente. De tal manera que el *software* de los vehículos de Toyota contiene una mezcla de códigos fuente de Denso, Delphi y Toyota, así como una mezcla de dos sistemas operativos: OSEK e ITRON<sup>36</sup> —ambos de arquitectura de *software* abierta— especializados en sistemas embebidos (Barr, 2013a).

¿Toyota utiliza las mejores prácticas para diseñar *software* embebido? En la industria automotriz se han acumulado, bajo la forma de normas y estándares, las mejores prácticas para asegurar la calidad y confiabilidad del *software*, particularmente las que establece la Misra,<sup>37</sup> especializada en *software* embebido. Misra-C38 contiene más de 100 reglas. Toyota tiene 119 reglas, pero sólo el 10 por ciento (11 reglas) coinciden con las de Misra-C. La Nasa (2011) encuentra 7,000 violaciones; en tanto que Barr (2013a) encuentra 80,000 violaciones de las reglas de Misra-C en el Camry-2005. La conclusión es que Toyota ha desarrollado sus propios estándares. No obstante, un tercio de los códigos propios de Toyota son violados por ellos mismos. Como señala Barr en el juicio “Bookout vs. Toyota”: “(t)hey (Toyota) weren’t enforcing their own rules” (Barr, 2013a: 120).

v) *El problema del aceleramiento no-intencional.* Cuando falla el *software* de un vehículo, a diferencia del carburador, es indispensable contar con recursos, capacidades especializadas y reglas de propiedad para representar la naturaleza del problema. Esto es así porque es necesario: i) considerar que los principios y fenómenos naturales que explota una microcomputadora se encuentran en distintas escalas (nano, micro y macro fenómenos); ii) comprender la diversidad de familias u dominios tecnológicos (eléctrico, electrónicos, mecánico, neumático, etcétera) que se explotan para producir una función; iii) vislumbrar la diversidad de interfaces necesarias para vincular los diferentes dominios tecnológicos; iv) comprender cómo interacciona directamente el *software* de más bajo

<sup>35</sup> La cadena de diseño de los componentes E/E involucra a: propietarios de las patentes, productores de semiconductores, proveedores de subsistemas E/E; integradores de sistemas, proveedores de *software* y proveedores de herramientas, entre otros (Day, 2012).

<sup>36</sup> Es un estándar abierto desarrollado en Japón para sistemas operativos embebidos que trabajan en tiempo real (Krikke, 2003).

<sup>37</sup> Misra: Motor Industry Software Reliability Association.

<sup>38</sup> Misra-C es el conjunto de recomendaciones o directrices para el uso del lenguaje C en el desarrollo de *software*.

nivel —*firmware*—, con el *hardware*; *v)* mantener la coherencia de un programa de *software* que contiene millones de líneas de códigos fuente; *vi)* acceder a los “compiladores” para ensamblar el lenguaje de programación (legible por un ser humano), con los códigos y lenguaje de la máquina; *vii)* contar con la autorización del propietario para acceder a este *software* más profundo, que es *software* propietario (Nasa, 2011).

Para profundizar sobre la naturaleza de la aceleración no intencional, surge la pregunta sobre si las fallas del *software* dejan huella o no. A continuación se consideran estos dos escenarios.

*La falla del software deja huella.* Si la falla o malfuncionamiento ha sido anticipado por los diseñadores del *software*, se dispone de parámetros de seguridad, con códigos de diagnóstico de problemas, así como un guardián supervisor (*watchdog supervisor*) es posible representar la naturaleza y los mecanismos que producen un error o falla de una función del *software*.

Sin embargo, para elaborar un diagnóstico no es suficiente identificar el código de diagnóstico del problema. Es necesario acceder a los “códigos fuente” del *software*<sup>39</sup> y disponer de recursos significativos, tiempo y capacidades especializadas. En el marco de la querella “Bookout vs. Toyota”, la corte le demandó a Toyota revelar sus códigos fuente para que, en condiciones de máxima secrecía, un grupo de expertos revisara el *software* del sistema de aceleración. En la investigación se involucraron dos equipos: el de Koopman (2014) y el de Barr (2013a). El esfuerzo de revisar y poner a prueba el *software* del sistema de aceleración costó 1.5 millones de dólares e involucró, durante más de 18 meses, a 12 expertos en *software* y electrónica, para el equipo de Barr (2013a).

*La falla del software no deja huella.* No es posible identificar y anticipar todos los billones de errores posibles que resultan en la aceleración no-intencional. “*There are no methods for capturing pre-event software state and performance following an UA<sup>40</sup> event either on the vehicle or as a diagnostic tool*” (Nasa, 2011: 175). Cuando la “imaginación” u “experiencia” de los diseñadores dejan sin considerar una o más variables de *software* críticas, que pueden producir mal funcionamiento del *software*, una falla del *software* no deja huella. De manera que, por ejemplo, una falla en el *hardware*, la cual los diseñadores no anticiparon y, por lo tanto, escapa a la detección por parte del “guardián supervisor”, puede corromper una región de la memoria del *chip* y provocar vacíos, tareas muertas y mal funcionamiento de la arquitectura de seguridad del *software*. La falla del *software* puede emergir de manera breve y desaparecer del sistema sin dejar huella alguna.

<sup>39</sup> Así como también es necesario tener acceso al compilador que permite acceder al lenguaje de la máquina. A esta herramienta no tuvieron acceso ni la Nasa (2011) ni Koopman (2014), sólo el equipo de Barr (2013a).

<sup>40</sup> UA: *unintentional acceleration*.

*vi) Consecuencias e implicaciones de la aceleración no intencional.* En el periodo 2000<sup>41</sup>-2010, la NHTSA<sup>42</sup> recibió 6,200 quejas asociadas con la aceleración involuntaria de los vehículos de Toyota. El reporte incluye 89 muertes y 57 heridos. Se sospecha que, adicionalmente, otras 52 muertes se vinculan con el mismo problema. A nivel mundial Toyota “llama” (*recall*) a reparación más de ocho millones de vehículos relacionados con la aceleración involuntaria provocadas por la alfombra del piso y el acelerador “pegajoso” (*The New York Times*, 2010).

Asimismo, después de cuatro años de investigación penal, en 2014, el Departamento de Justicia de los EEUU, impone una multa de 1.2 mil millones de dólares a Toyota. Hasta ese momento la más alta en la historia de los EEUU impuesta a una empresa del sector automotriz. El dictamen de la Corte señala que: “Toyota admite que engaño a los consumidores estadounidenses al ocultar y hacer declaraciones engañosas sobre dos problemas (problema de la alfombra y pedal “pegajoso”) relacionados con la seguridad que afectan a sus vehículos, cada uno de los cuales causó un tipo de aceleración involuntaria” (U.S. Department of Justice 2014:1). Como parte del acuerdo Toyota debe, por un periodo de tres años, pagar a un monitor independiente para que revise y evalúe las políticas, prácticas y procedimientos de seguridad de Toyota. En el acuerdo y sanción no se menciona el *hardware/software* del sistema de aceleración como una de las causas del mal funcionamiento del sistema de aceleración (U.S. Department of Justice 2014).

Toyota mantiene que no existe defecto alguno del *software* de su sistema de aceleración y que el problema es causado por errores humanos o por problemas mecánicos limitados, pero no por el sistema electrónico o de computación. “*Toyota had three responses: first, blame the driver; second, blame the floor mat; third, blame a sticky gas pedal. And NHTSA, without doing any meaningful independent review, accepted Toyota’s explanations*” (Committee on Energy and Commerce, 2010: 6). Manipular la información de manera no responsable con la integridad de la vida de consumidor, engañar a la opinión pública son sólo una pequeña muestra de cómo las empresas actúan cuando no existen reglas y monitoreo efectivo. Importa, no cabe duda, la ética de las corporaciones.

## CONCLUSIONES

Por más de 100 años, los vehículos han explotado, mediante interfaces mecánicas, el efecto Venturi. Sin embargo, a pesar de sus limitaciones, el carburador y el sistema de aceleración asociado se desempeñan satisfactoriamente. Esta situación

---

<sup>41</sup> Mayo de 2000.

<sup>42</sup> NHTSA: *National Highway Traffic Safety Administration*.

cambia cuando la sociedad reconoce como problemático la contaminación y se crean reglas institucionales que tienen como objeto mejorar los sistemas de control de contaminación de los vehículos. Sin embargo, aun a pesar de la mejora y profundización estructural, el carburador no logra satisfacer las reglas cada vez más exigentes. El carburador se convierte en un sistema mecánico complejo, costoso, vulnerable y sobre todo intrínsecamente limitado para entregar una mezcla de aire y gasolina óptima. En este nuevo contexto institucional, los dispositivos mecánicos que encapsulan el efecto Venturi ya no son suficientes.

Durante la evolución del carburador, en particular, y del motor de combustión, en general, se integran en el vehículo un conjunto de componentes y subsistemas relativamente independientes entre sí. Sin embargo, la necesidad de controlar las emisiones empuja a las empresas a conectar estos subsistemas con el propósito de desarrollar formas avanzadas de control y retroalimentación. Se trata de una evolución que conduce a la emergencia de un sistema jerárquico de control.

Con los nuevos principios asociados a la electrónica, la naturaleza del sistema de aceleración se transforma. El EFI sustituye al carburador porque en principio logra explotar un nuevo conjunto de efectos naturales que le permiten reducir con éxito las emisiones contaminantes. Fenómenos a cuya escala no es posible interactuar con dispositivos mecánicos convencionales como el carburador.

Sobre la base del EFI, emerge el sistema de aceleración electrónica. Este logro es posible en la medida que se cuenta con: *i)* una computadora capaz de representar el comportamiento de la materia y de la energía, en niveles cada vez más profundos; *ii)* una red de fenómenos naturales encapsulados en sensores, especializados en medir y monitorizar el sistema, y *iii)* actuadores que, también aprovechando efectos naturales, sirven para activar las capacidades de control del sistema. Estas nuevas tecnologías posibilitan nuevos espacios de solución, transitando de escalas macroscópicas a microscópicas de los fenómenos naturales. La tecnología del sistema de aceleración explota una familia de fenómenos que se encarnan en *hardware* y *software* complejos y costosos.

El caso del Camry-2005 de Toyota ayuda a reconstruir de manera precisa cómo el incremento en la complejidad tecnológica del sistema de aceleración electrónico y la fragmentación del conocimiento (especializado) en múltiples agentes, han generado problemas de información de dos tipos: información en la operación del vehículo (problemas de funcionamiento del sistema de aceleración electrónica y de los procesos de computo asociados) e información (incompleta o asimétrica) entre los diseñadores, productores, reguladores y usuarios de la tecnología del vehículo. Como resultado de estos problemas de información, la aceleración no intencionada se vuelve un problema cuya representación se ha convertido en compleja, costosa y difícil, sobre todo cuando la

falla del *software* no deja huella. Esta situación impide a los agentes contar con pistas para identificar la naturaleza del problema y solucionarlo.

Para terminar, el estudio de la evolución del sistema de aceleración nos conduce a identificar la gramática profunda de los procesos de cambio tecnológico. En el transcurso del trabajo se construye, no una sola, sino una pluralidad de narraciones que dan cuenta de la diversidad de trayectorias evolutivas. En esta evolución importan las instituciones, toda vez que las reglas inducen determinados procesos evolutivos. El desarrollo de instituciones que promuevan una organización capaz de comprender y regular la tecnología electrónica contenida en el vehículo es, para los propósitos de la seguridad, una tarea primordial presente e indispensable.

## REFERENCIAS BIBLIOGRÁFICAS

- Abernathy, William (1979), “The Productivity Dilemma: Roadblock to Innovation in the Automobile Industry”, *Business History Review*, 53 (4), pp. 546-548, <https://doi.org/10.2307/3114740>.
- Aird, Forbes (2001), *Bosch Fuel Injection Systems*, New York, HP Books, <http://www.megasquirt.325ix.com/files/efi/Bosch%20Fuel%20Injection%20Systems%20-%20Forbes%20Aird.pdf>, consultado el 15 de noviembre de 2017.
- Arthur, Brian (2007), “The Structure of Invention”, *Research Policy*, 36, 2, march.
- (2009), *The nature of technology: what it is and how it evolves*, The Free Press, New York.
- Arzen, Karl y Martina Maggio (2017), *Real-Time Systems*, Lund University, 17 de enero de 2017, [http://www.control.lth.se/media/Education/EngineeringProgram/frtn01/2017/L0\\_17.pdf](http://www.control.lth.se/media/Education/EngineeringProgram/frtn01/2017/L0_17.pdf), consultado el 3 de agosto de 2017.
- AUTOSAR (2013), “Technical Safety Concept Status Report V1.20”, [https://www.autosar.org/fileadmin/user\\_upload/standards/classic/4- 1/autosar\\_tr\\_SafetyConceptStatusReport.pdf](https://www.autosar.org/fileadmin/user_upload/standards/classic/4- 1/autosar_tr_SafetyConceptStatusReport.pdf), consultado el 3 de agosto de 2017.
- Barr, Michael (2013a), “Bookout vs. Toyota”, case No. CJ-2008-7969, District Court of Oklahoma County, [http://www.safetyresearch.net/Library/Bookout\\_v\\_Toyota\\_Barr\\_redacted.pdf](http://www.safetyresearch.net/Library/Bookout_v_Toyota_Barr_redacted.pdf), consultado el 10 de julio de 2017.
- (2013b), “Bookout vs. Toyota: 2005 Camry L4 Software Analysis”, District Court of Oklahoma County, [http://www.safetyresearch.net/Library/BarrSlides\\_final\\_scrubbed.pdf](http://www.safetyresearch.net/Library/BarrSlides_final_scrubbed.pdf), consultado el 10 de julio de 2017.
- Basalla, George (1988), *The Evolution of Technology*, Cambridge University Press, Cambridge.

- Bereisa, Jonas (1983), "Applications of Microcomputers in Automotive Electronics", *IEEE Transactions on Industrial Electronics*, 30 (2), pp. 87-96. DOI: 10.1109/TIE.1983.356715.
- Bergmiller, Peter (2015), "Towards Functional Safety in Drive-by-Wire Vehicles", *Springer International Publishing*, Suiza, <http://dx.doi.org/10.1007/978-3-319-17485-3>.
- Bonnick, Allan (2001), *Automotive Computer-Controlled Systems Diagnostic Tools and Techniques*, Routledge, New York.
- Borg, Kevin (2010), *Auto Mechanics: Technology and Expertise in Twentieth-Century America. Studies in Industry and Society*, The Johns Hopkins University Press, Baltimore.
- Butler, Rick y George, Finelli (1993), "The Infeasibility of Quantifying the Reliability of Life-Critical Real-Time Software", *IEEE Transactions on Software Engineering*, 19 (1), pp. 3-12.
- Clarke, Sally (2007), *Trust and Power. Consumers, the Modern Corporation, and the Making of the United States Automobile Market*. United Kingdom, Cambridge University Press.
- Committee on Energy and Commerce (2010), "Hearing: Response by Toyota and NHTSA to Incidents of Sudden Unintended Acceleration", Serial núm. 111-96, <https://www.transportation.gov/content/response-toyota-and-nhtsa-incidents-sudden-unintended-acceleration>, consultado el 5 de agosto de 2017.
- Crolla, David; Foster, David; Kobayashi, Toshio y Vaughan, Nicholas (2014), *Encyclopedia of Automotive Engineering*, John Wiley & Sons, New Jersey.
- Day, John (2012), *Automotive E/E Reliability*, SAE International.
- Dirk, John (1998), "OSEK/VDX History and Structure. OSEK/VDX Open Systems in Automotive Networks", Ref. núm. 1998/523. IEE Seminar, 13 de noviembre de 1998, DOI: 10.1049/ic:19981073.
- DOT (2008), "Analysis of Event Data Recorder Data for Vehicle Safety Improvement", U.S. Department of Transportation, [https://rosap.ntl.bts.gov/view/dot/6182/dot\\_6182\\_DS1.pdf?](https://rosap.ntl.bts.gov/view/dot/6182/dot_6182_DS1.pdf?), consultado el 8 de septiembre de 2017.
- (2011), "Technical Assessment of Toyota Electronic Throttle Control (ETC) Systems", U.S. Department of Transportation, National Highway Traffic Safety Administration, [http://www.safetyresearch.net/Library/nhtsa-nasa\\_Response\\_Final\\_052311.pdf](http://www.safetyresearch.net/Library/nhtsa-nasa_Response_Final_052311.pdf).
- Doyle, Jack (2000), *Taken for a Ride*, Four Walls Eight Windows, Nueva York.
- Fleming, Lee y Sorenson, Olav (2004), "Science as a Map in Technological Search", *Strategic Management Journal*, 25, pp. 909-928.
- Gaertner, Christoph (2015), "Trends and Lookout of the Automotive Software Industries", en Fernandes, João.; Machado, Ricardo. y Wnuk, Krysztof, *Software Business*, New Springer, New York.

- Ganssle, Jack y Barr, Michael (2003), *Embedded Systems Dictionary*, CMP Books, New York.
- Group E-Gas (2013), “Standardized E-Gas Monitoring Concept for Gasoline and Diesel Engine Control Units. Audi-bmw-Porsche-V”, <https://www.iav.com/sites/default/files/attachments/seite/ak-egas-v5-5-en-130705.pdf>, consultado el 12 de julio del 2017.
- Halderman, James y Linder, James (2006), *Automotive Fuel and Emissions Control Systems*, Prentice Hall, New York.
- Henderson, Rebecca y Clark, Kim (1990), “Architectural Innovation: The Reconfiguration of Existing Product Technologies and Failure of Established Firms”, *Administrative Science Quarterly*, 35 (1), pp. 9-30, [http://dimetic.dime-eu.org/dimetic\\_files/HendersonClarkASQ1990.pdf](http://dimetic.dime-eu.org/dimetic_files/HendersonClarkASQ1990.pdf), consultado el 15 de marzo de 2017.
- Hennessy, John y Patterson, David (2017), *Computer Architecture: A Quantitative Approach*, Morgan Kauffman, Cambridge.
- Holland, John (2004), *El orden oculto, de cómo la adaptación crea la complejidad*, Fondo de Cultura Económica, México.
- IEEEExplore (1976), *News from Industry*, <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6367563>, consultado el 12 de agosto de 2017.
- Isermann, Rolf (2011), *Fault-Diagnosis Applications*, Springer, Berlin.
- \_\_\_\_ (2014), *Engine Modeling and Control Modeling and Electronic Management of Internal Combustion Engines*, Springer, Berlin.
- Kato, Mitsuharu (2014), “Historical Overview of Electronics and Automobiles: Breakthroughs and Innovation by Electronics and Electrical Technology”, en Crolla, David; Foster, David; Kobayashi, Toshio y Vaughan, Nicholas, *Encyclopedia of Automotive Engineering*, John Wiley & Sons, New Jersey.
- Kirchhoff, Suzanne y Peterman, David (2010), *Unintended Acceleration in Passenger Vehicles*. Washington: Congressional Research Service, <https://fas.org/sgp/crs/misc/R41205.pdf>, consultado el 10 de septiembre de 2017.
- Koopman, Phil (2014), *A Case Study of Toyota Unintended Acceleration and Software Safety*, Carnegie Mellon, Pittsburgh, 18 of September 2014, [https://users.ece.cmu.edu/~koopman/pubs/koopman14\\_toyota\\_ua\\_slides.pdf](https://users.ece.cmu.edu/~koopman/pubs/koopman14_toyota_ua_slides.pdf), consultado el 7 de julio de 2017.
- Krikke, Jan (2003), *The Most Popular Operating System in the World*, [www.linuxinsider.com/story/31855.html](http://www.linuxinsider.com/story/31855.html), consultado el 10 de octubre de 2017.
- Levinthal, Daniel (1997), “Adaptation on Rugged Landscapes”, *Management Science*, 43 (7), pp. 934-950, <http://www.jstor.org/stable/2634336>, consultado el 8 de junio de 2017.

- McCabe, Thomas (1976), "A Complexity Measure", *IEEE Transactions on Software Engineering*, 2 (4), pp. 308-320, <http://www.literateprogramming.com/mccabe.pdf>, consultado el 8 de junio de 2017.
- Mondt, Robert (2000), *Cleaner Car: The History and Technology of Emission Control Since the 1960s*, Society of Automotive Engineers.
- Nasa (2011), "Technical Support to the National Highway Traffic Safety Administration (NHTSA) on the Reported Toyota Motor Corporation (TMC) Unintended Acceleration (UA) Investigation, NESC", Hampton, VA, [https://www.nhtsa.gov/staticfiles/nvs/pdf/NASA-UA\\_report.pdf](https://www.nhtsa.gov/staticfiles/nvs/pdf/NASA-UA_report.pdf), consultado el 1 de julio de 2017.
- National Academy of Sciences (1992), *Policy Implications of Greenhouse Warming*, The National Academies Press, Washington, <https://doi.org/10.17226/1605>.
- National Research Council of the National Academies (2012), "The safety promise and challenge of automotive electronics. Transportation Research Board Special Report 308", <http://www.omg.org/hot-topics/documents/Safety-Promise-and-Challenge-of-Automotive-Electronics-TRB-2012.pdf>, consultado el 29 de mayo de 2017.
- Navet, Nicolas y Simonot-Lion, Francoise (2009), *Automotive Embedded Systems Handbook*, CRC Press, Florida.
- Nelson, Richard y Winter, Sidney (1982), *An Evolutionary Theory of Economic Change*, Harvard University Press, Cambridge.
- Niimi, Yukihide (2014), "ECU Technologies from Components to ECU Configuration", en Crolla, David; Foster, David; Kobayashi, Toshio y Vaughan, Nicholas, *Encyclopedia of Automotive Engineering*, John Wiley & Sons, New Jersey.
- NHTSA (2011), *Technical Assessment of Toyota Electronic Throttle Control (ETC) Systems*, National Highway Traffic Safety Administration, [https://www.nhtsa.gov/staticfiles/nvs/pdf/nhtsa-ua\\_report.pdf](https://www.nhtsa.gov/staticfiles/nvs/pdf/nhtsa-ua_report.pdf), consultado el 8 de enero, 2019.
- Porter, Michael (1983), *Cases in Competitive Strategy*, Free Press, New York.
- Reif, Konrad (2015), *Automotive Mechatronics: Automotive Networking, Driving Stability Systems, Electronics*, Springer Vieweg.
- Rosenberg, Nathan (1982), *Inside the Black Box: Technology and Economics*, Cambridge University Press, Cambridge.
- Schumpeter, Joseph (1934), *The Theory of Economic Development*, Harvard University Press, Cambridge.
- The New York Times* (2010), "Sudden Acceleration Death Toll Linked to Toyota Rises", May, 25, <https://www.nytimes.com/2010/05/26/business/26toyota.html>, consultado el 15 de julio de 2017.
- Thorner, Robert (1946), *Aircraft Carburation*, John Wiley & Sons, New York y London.
- U.S. Department of Justice (2014), "Toyota Motor Corporation-Deferred Prosecution Agreement. United States Attorney Southern District a/ New York", <https://>

[www.justice.gov/sites/default/files/opa/legacy/2014/03/19/toyota-def-pros-agr.pdf](http://www.justice.gov/sites/default/files/opa/legacy/2014/03/19/toyota-def-pros-agr.pdf), consultado el 1 de agosto de 2017.

Von Hippel, Eric (1988), *The Sources of Innovation*, Oxford University Press, New York.

Walther, David; Lin, Liwei y Pisano, Albert (2007), “Micro- and Nano-Technologies for Automotive Sensor Research, SAE”, Technical Paper 2007-01-1012, <https://doi.org/10.4271/2007-01-1012>, consultado el 8 de julio de 2017.

Wolf, Wayne (2006), *High-Performance Embedded Computing*, Morgan Kauffman, Massachusetts.