

Simulación eficiente de circuitos de electrónica conmutada en presencia de elementos parásitos

Franco Di Pietro
CIFASIS-CONICET
Rosario, Argentina
dipietro@cifasis-conicet.gov.ar

Ernesto Kofman
CIFASIS-CONICET
FCEIA-UNR
Rosario, Argentina
kofman@cifasis-conicet.gov.ar

Mónica Romero
CIFASIS-CONICET
FCEIA-UNR
Rosario, Argentina
romero@cifasis-conicet.gov.ar

Resumen—La simulación de circuitos de electrónica conmutada es una tarea particularmente compleja para los métodos de integración numérica por dos motivos. Por un lado, la presencia de discontinuidades frecuentes, producto de las conmutaciones de llaves y diodos, deben ser detectadas y la simulación debe reiniciarse ante cada una de ellas para evitar resultados erróneos. Además, las diferentes dinámicas presentes en este tipo de circuitos conducen a sistemas rígidos (*stiff*) que requieren el uso de métodos implícitos, los cuales son computacionalmente costosos. En este artículo proponemos la utilización de métodos QSS linealmente implícitos para la simulación de este tipo de circuitos, remarcando las ventajas de su utilización y sus correspondientes limitaciones.

Palabras claves—simulación, electrónica de potencia, elementos parásitos, QSS

I. INTRODUCCIÓN

Los circuitos de electrónica conmutada son utilizados para realizar conversiones de potencia de forma eficiente, y pueden ser encontrados en la gran mayoría de los dispositivos electrónicos de uso cotidiano e industrial. Estos circuitos utilizan uno o varios interruptores cuyos ciclos de trabajo son controlados para regular la tensión de salida deseada. A los efectos de minimizar el *ripple* en dicha salida, los elementos de conmutación deben ser operados a altas frecuencias (del orden los kHz).

La simulación de este tipo de circuitos no es trivial [1]. Por un lado, las altas frecuencias de conmutación implican discontinuidades frecuentes en los modelos a simular. Esto implica que los métodos de integración numérica deben detectar los momentos precisos donde dichas discontinuidades tienen lugar y reiniciar la simulación ante cada una de ellas, ya que no se puede integrar a través de una discontinuidad sin cometer un error inaceptable. Estos reinicios frecuentes son costosos computacionalmente y limitan el paso de integración utilizado. Por otro lado, el uso de modelos realistas para los elementos de conmutación conduce a modelos *stiff*, con dinámicas lentas y rápidas simultáneas. Los modelos con estas características deben integrarse con *métodos implícitos* ya que los explícitos resultarán inestables a menos que se disminuya considerablemente el paso de integración [2]. Los métodos implícitos tienen un costo elevado ya que requieren iterar para resolver un sistema de ecuaciones en cada paso (costo que

aumenta notablemente a medida que el tamaño del modelo se incrementa).

En los últimos años, se ha desarrollado una nueva familia de métodos de integración denominados QSS (por *Quantized State Systems* [3]) que puede resolver eficientemente sistemas con discontinuidades [2]. En los QSS la discretización temporal es reemplazada por la cuantificación de las variables de estado y entre estos métodos encontramos los QSS Linealmente Implícitos (LIQSS) [4], que pueden además integrar muy eficientemente sistemas *stiff* sin necesidad de realizar iteraciones (siempre y cuando la rigidez de estos sistemas se deban a cierta estructura particular).

Por este motivo, los LIQSS permiten simular sistemas de electrónica conmutada de manera muy eficiente siempre y cuando el circuito cuente con ciertas características [1], lo que se cumple en casi todas las topologías de las fuentes conmutadas (excepto en la fuente *Ćuk*). Recientemente, se desarrolló una versión modificada de estos métodos, denominada mLIQSS, que permite integrar de manera eficiente topologías más generales (incluyendo el caso de la fuente *Ćuk*) [5], [6]. En todos los casos analizados hasta aquí, sin embargo, se omitió la presencia de elementos parásitos. Dado que los elementos dinámicos (inductancias y capacitores) cambian la estructura de las ecuaciones, es de esperar que la presencia de los mismos altere la condición requerida por los LIQSS y mLIQSS para funcionar eficientemente.

El objetivo de este trabajo es entonces el de estudiar el desempeño de estos algoritmos cuando a los modelos de fuentes conmutadas se les agrega la presencia de ciertos elementos parásitos (inductancias en particular) y analizar si en este caso siguen ofreciendo ventajas frente a los métodos clásicos de integración.

Este artículo está organizado de la siguiente manera: en la sección II se brinda una breve introducción a los métodos QSS que serán luego empleados en la simulación de los ejemplos. Luego en la sección III se muestra el estudio de diferentes casos de elementos parásitos en un convertidor *Buck*. A continuación, en la sección IV se muestran resultados para el caso de un convertidor *Ćuk*. Finalmente, en la sección V presentamos las conclusiones del trabajo.

II. MÉTODOS QSS

Dada una *Ecuación Diferencial Ordinaria* (EDO) representada como un sistema de ecuaciones de estado:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t), t), \quad (1)$$

donde $\mathbf{x}(t) \in \mathbb{R}^n$ es el vector de estados, el método QSS de primer orden (QSS1) [3] resuelve analíticamente una aproximación de esta EDO dada por:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{q}(t), t). \quad (2)$$

Aquí, $\mathbf{q}(t)$ es el vector de *estados cuantificado* cuyos componentes están limitados a seguir trayectorias seccionalmente constantes. Cada estado cuantificado $q_i(t)$ esta relacionado con el correspondiente estado $x_i(t)$ a través de una *función de cuantificación con histéresis*:

$$q_i(t) = \begin{cases} x_i(t) & \text{si } |q_i(t^-) - x_i(t)| = \Delta Q_i \\ q_i(t^-) & \text{cada otro caso} \end{cases}$$

Es decir, $q_i(t)$ sólo cambia cuando difiere de $x_i(t)$ en una magnitud ΔQ_i llamada *quantum*. Cuando la diferencia entre el estado y el estado cuantificado alcanza el quantum, el estado cuantificado $q_i(t)$ toma el valor de $x_i(t)$. De tal manera, los pasos son locales a cada variable de estado. La Figura 1 muestra las trayectorias de un estado y su correspondiente versión cuantificada.

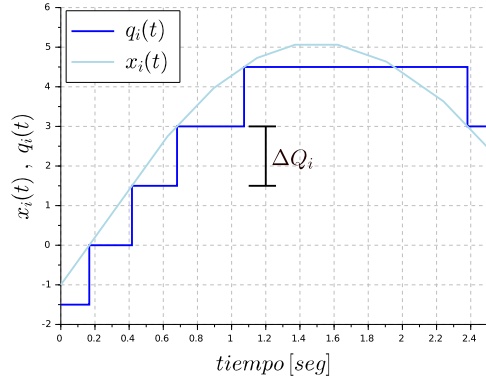


Figura 1: Función de cuantificación con histéresis para QSS1.

Debido a esta particular forma de las trayectorias cuantificadas, la solución numérica de la Ec. (2) es trivial y puede ser fácilmente traducida en un simple algoritmo de simulación. Existen además métodos QSS de orden dos (QSS2) y tres (QSS3), en los cuales las trayectorias de los estados cuantificados son seccionalmente lineales y seccionalmente parabólicas, respectivamente. Estos métodos permiten dar pasos más grandes para un mismo quantum.

Los métodos de QSS tienen la propiedad de mantener el error de simulación acotado por una magnitud proporcional al quantum [2]. Además, debido a la forma simple de las trayectorias de estado, es trivial detectar discontinuidades y como los pasos son locales a cada variable de estado, una

discontinuidad no requiere reiniciar la simulación sino que sólo involucra cálculos en las variables afectadas por el evento.

Toda la familia de métodos QSS ha sido implementada en el *stand-alone QSS solver* [7], donde además se pueden realizar las simulaciones utilizando métodos clásicos. De este modo, es posible realizar comparaciones fácilmente respecto a la eficiencia de los métodos de integración en cada caso.

II-A. QSS en sistemas stiff

La Figura 2 muestra el resultado de simular el sistema

$$\dot{x} = -0.1 \cdot (x - 10.5); \quad x_0 = 0 \quad (3)$$

utilizando QSS1 con quantum $\Delta Q = 1$. Puede verse que, pese a que la solución analítica debería converger a $x(\infty) = 10.5$, la simulación termina en oscilaciones espurias en torno al punto de equilibrio. Si este sistema fuera la parte rápida de un sistema más grande con partes mucho más lentas (es decir, un sistema stiff), estas oscilaciones espurias implicarían realizar muchos cálculos innecesarios y costosos computacionalmente.

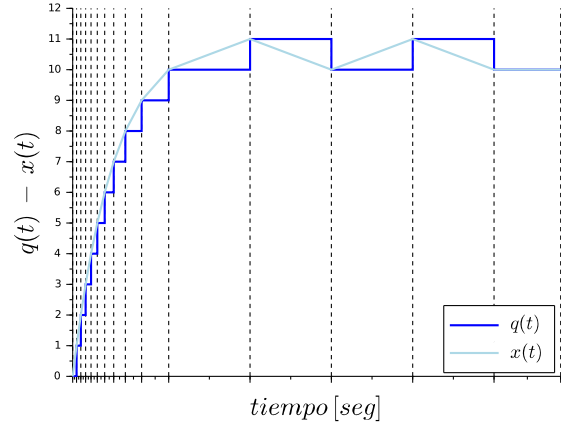


Figura 2: Simulación del sistema 3 con QSS1.

Los métodos de LIQSS [4] solucionan este problema ya que, con una idea inspirada en los métodos implícitos clásicos, evalúan las derivadas usando el valor futuro de los mismos. Esta evaluación futura que produce que los métodos clásicos deban resolver un sistema de ecuaciones en cada paso, en los métodos de LIQSS no tiene el mismo efecto ya que el futuro estado cuantificado se conoce de antemano (es, $q_i(t) \pm \Delta Q_i$). Por esto, la implementación de los métodos LIQSS resulta en algoritmos que no requieren de iteraciones.

La Figura 3 muestra el resultado de simular nuevamente el sistema de la Ec. 3 pero ahora utilizando el método LIQSS1, donde las oscilaciones en torno al punto de equilibrio no están presentes.

Dado que los LIQSS sólo miran el valor futuro de cada variable individualmente, sólo pueden evitar las oscilaciones espurias que se deben al cambio de dicha variable. Sin embargo, hay muchas situaciones en las que las oscilaciones espurias se dan de a pares de variables, donde el cambio de una variable

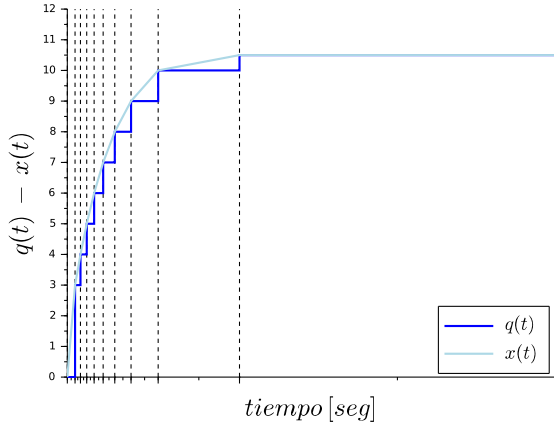


Figura 3: Simulación del sistema 3 con LIQSS1.

provoca un cambio en la derivada de la segunda variable y viceversa. Para resolver esto, recientemente se formuló una familia de métodos denominado mLIQSS que tiene en cuenta estas interacciones entre pares de variables y permite simular eficientemente sistemas stiff con estas características [6].

Esta mejora puede verse en la simulación del sistema

$$\begin{aligned} \dot{x}_1 &= -x_1 - x_2 + 0.2 \\ \dot{x}_2 &= x_1 - x_2 + 1.2 \end{aligned} \quad (4)$$

con condiciones iniciales $\mathbf{x}_0 = [-4 \ 4]^T$ y quantum $\Delta Q_{1,2} = 1$ que se muestra en la Figura 4, donde las oscilaciones espurias que muestra LIQSS1 desaparecen al usar mLIQSS1.

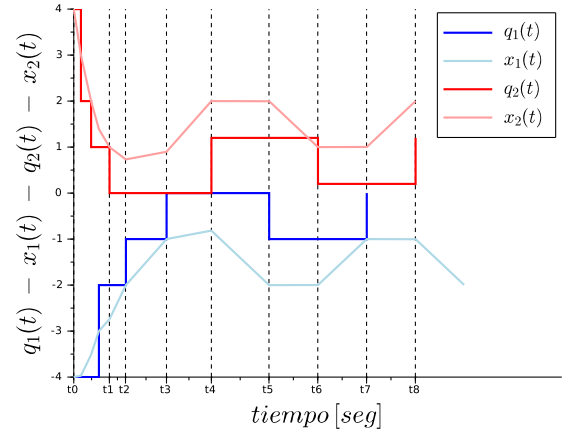
II-B. Rigidez en circuitos electrónicos

El primer problema que vimos, ejemplificado por la ecuación 3, puede observarse en un circuito electrónico muy simple como el del inductor en serie con una resistencia grande ilustrado en la Figura 5a.

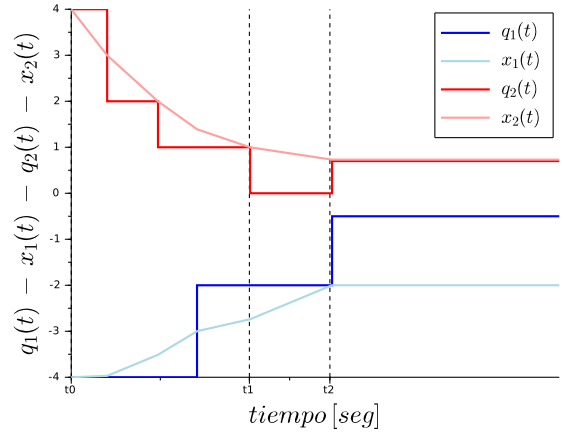
Al simular este circuito con un método QSS, y recordando que los estados cuantificados sólo pueden tomar determinados valores, un pequeño cambio en el valor cuantificado de la corriente (como consecuencia de no poder arribar a un punto de equilibrio) producirá un gran cambio en la tensión que cae en la resistencia. Este gran cambio en esa tensión producirá a su vez un cambio en la derivada de la corriente, dando lugar así a oscilaciones espurias rápidas, obteniendo un resultado como el mostrado en la Figura 2. Si en cambio, este sistema fuese simulado utilizando un método LIQSS, el valor cuantificado de la corriente quedaría fijado en el punto de equilibrio, eliminando así las oscilaciones.

Notar que en este caso, cuando la resistencia $R \rightarrow \infty$, se traduce en una restricción sobre una variable de estado $i_L(t) = 0$.

El caso de las oscilaciones espurias entre dos variables, en tanto, aparece en el circuito de la Figura 5b donde nuevamente la resistencia R es grande. Aquí, un pequeño cambio en la versión cuantificada de la corriente por una inductancia se verá

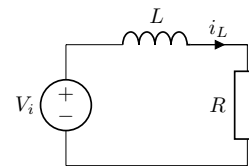


(a) Con LIQSS1.

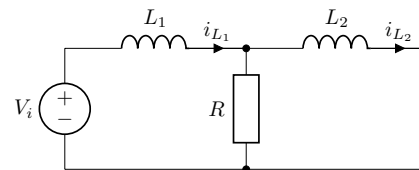


(b) Con mLIQSS1.

Figura 4: Simulación del sistema de las Ecs.(4).



(a)



(b)

Figura 5: Circuitos RL.

reflejado como un gran cambio en la derivada de la corriente por la otra inductancia, lo que al simularse mediante QSS se reflejará en oscilaciones espurias entre ambas variables. Este caso tampoco lo resolverá LIQSS, donde se obtendrán resultados similares a los mostrados en la Figura 4a, pero sí lo resolverá mLIQSS.

Notar que este segundo caso, cuando la resistencia $R \rightarrow \infty$, se traduce en una restricción sobre dos variables de estado: $i_{L_1} - i_{L_2} = 0$.

Veremos entonces algunas topologías circuitales correspondientes a fuentes conmutadas en las cuales se dan algunas de estas situaciones, observando el desempeño de los métodos LIQSS y mLIQSS. En particular, utilizaremos los métodos de segundo orden, lo cuales muestran el mejor desempeño a la hora de simular circuitos de electrónica conmutada.

III. ANÁLISIS SOBRE UN CONVERTIDOR BUCK

Un circuito de electrónica conmutada muy utilizado es el convertidor *Buck*. Se trata de un convertidor DC-DC, es decir, que tiene como entrada una determinada tensión continua (usualmente proveniente de la rectificación de una tensión alterna), y se obtiene a la salida una tensión continua de otro valor. Este circuito es también conocido como *reductor*, puesto que la tensión de salida resulta menor a la tensión de entrada. Podemos ver un diagrama de dicho circuito en la Figura 6, donde tanto el diodo D como la llave S serán modelados como una resistencia muy grande (R_{off}) cuando el elemento esté en condición de corte, y como una resistencia muy pequeña (R_{on}) cuando esté en conducción.

En este caso, en ausencia de elementos parásitos sabemos que los métodos de LIQSS funcionan eficientemente (y notablemente mejor que los métodos clásicos [1]). Esto puede verse topológicamente en el circuito ya que en la única situación que habría una restricción sobre variables de estado es cuando el diodo y la llave están abiertos simultáneamente (con $R_{\text{off}} \rightarrow \infty$), pero esta restricción se dará sobre una sola variable (i_L).

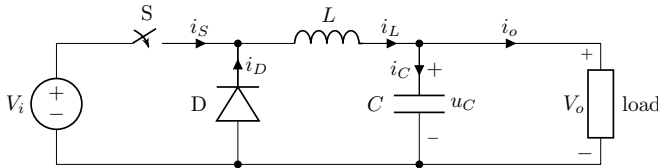


Figura 6: Circuito convertidor Buck.

Estudiaremos a continuación los resultados de simular dicho circuito en presencia de distintos elementos parásitos. Para ello, se mostrarán los resultados de simular el mismo circuito tanto con métodos clásicos (DASSL [8], CVODE [9] e IDA [10]) como con un método QSS linealmente implícito (LIQSS). En este caso utilizaremos el método LIQSS2 que es de segundo orden y que es con el cual, dentro de la familia de métodos QSS, se obtienen los mejores resultados a la hora

de simular este tipo de circuitos. Otros resultados de simular distintas topologías pueden encontrarse en [1], [6].

Todas las simulaciones se realizarán con los siguientes parámetros:

- $U = 24 V$ - tensión de entrada
- $C = 10 mF$ - capacitor
- $L = 10 mH$ - inductor
- $R = 10 \Omega$ - Resistencia de carga
- $f = 10 kHz$ - frecuencia de conmutación
- $DC = 0,35$ - ciclo de trabajo
- $R_{\text{off}} = 1 \cdot 10^5 \Omega$ - resistencia de corte
- $R_{\text{on}} = 1 \cdot 10^{-5} \Omega$ - resistencia de conducción
- $t_f = 10 mseg$ - tiempo final de simulación
- $tol = 1 \cdot 10^{-3}$ - tolerancias absolutas y relativas de error de simulación

III-A. Inductancia parásita en la llave

Un caso de estudio interesante es la presencia de una inductancia parásita modelada en serie con el elemento de conmutación como se muestra en la Figura 7. Esta situación es particularmente interesante debido a que la presencia de dicha inductancia produce efectos indeseados (afecta la velocidad de conmutación del dispositivo [11]–[13], o puede producir sobretensiones).

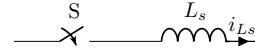


Figura 7: Modelo de llave con inductancia parásita.

La Tabla I muestra los resultados de simular el convertidor Buck con la inductancia parásita mencionada de valor $L_s = 10 nH$.

Tabla I: Comparación tiempos de simulación para convertidor Buck con inductancia parásita en la llave.

Método de Integración	Evaluaciones de función f_i	CPU [mseg]
DASSL	41010	17
CVODE	13498	15
IDA	37524	18
LIQSS2	–	5511
mLIQSS2	–	8

Podemos observar en la tabla que LIQSS2 no es capaz de simular el circuito con un desempeño comparable a los métodos clásicos. Esto se debe a que cuando la llave se cierra y el diodo está abierto, la inductancia L del circuito y la inductancia parásita L_s quedan con una configuración similar a la del circuito de la Figura 5b donde el diodo abierto juega el papel de la resistencia grande y si $R_D \rightarrow \infty$ aparecería una restricción entre las dos variables de estado. Como mencionamos en la sección II, este caso stiff no es manejado eficientemente por los LIQSS ya que los mismos introducen oscilaciones espurias entre dos variables. Sin embargo, el

método mLIQSS2 sí logra integrar eficientemente el circuito, y además supera los resultados obtenidos con cualquiera de los métodos clásicos.

III-B. Inductancia parásita en el diodo

Similarmente a lo visto en el caso anterior, se suele modelar también la presencia de inductancia parásita en serie con el diodo, como puede verse en la Figura 8.

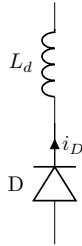


Figura 8: Modelo de diodo con inductancia parásita.

La Tabla II muestra los resultados de simular el convertidor Buck con una inductancia parásita en paralelo con el diodo de valor $L_d = 10nH$.

Tabla II: Comparación tiempos de simulación para convertidor Buck con inductancia parásita en el diodo.

Método de Integración	Evaluaciones de función f_i	CPU [mseg]
DASSL	38938	18
CVODE	15908	17
IDA	36064	19
LIQSS2	—	194
mLIQSS2	—	7

Encontramos aquí resultados similares a los del caso anterior. La situación análoga a la del circuito de la Fig.5b se da aquí mientras el diodo conduce y la llave está abierta, motivo por el cual LIQSS2 tampoco tiene resultados eficientes y mLIQSS2 sigue siendo la mejor opción.

III-C. Buck interleaved

Veremos a continuación la influencia del tamaño del sistema en estudio. Para ello, simularemos el circuito denominado *Buck interleaved*. Se trata de un convertidor Buck de varias etapas, donde cada una de ellas entra en conducción alternadamente. Estudiaremos el comportamiento de este circuito en presencia de una inductancia parásita modelada en serie con las llaves de cada etapa.

Se muestran en la Figura 9 los resultados de simular dicho circuito, variando el número de etapas que lo componen, con una inductancia parásita en las llaves de $L_s = 10nH$. Allí podemos observar que LIQSS2 no logra simular eficientemente el circuito, y sólo logra igualar a los métodos clásicos cuando el número de etapas se incrementa a 32, esto es, cuando se incrementa mucho el número de discontinuidades del sistema.

Esto se debe a que si bien este método presenta las oscilaciones espurias mencionadas en la sección II, la familia de métodos QSS presenta mayores ventajas sobre los métodos clásicos a medida que el sistema tenga discontinuidades más frecuentes. Por otro lado, vemos que mLIQSS2 logra integrar eficientemente el circuito, superando el desempeño de los métodos clásicos. De hecho, se puede observar que la diferencia entre mLIQSS2 y CVODE (el método clásico con el que se obtienen mejores resultados) va creciendo a medida que se incrementa el tamaño del circuito, llegando a ser hasta 30 veces más rápido al simular 32 etapas.

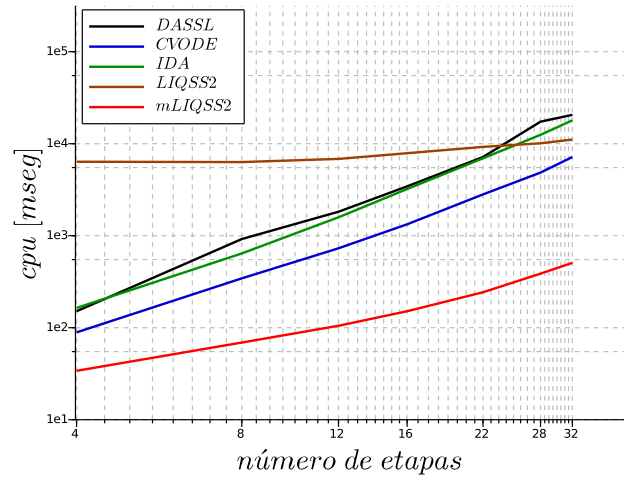


Figura 9: Buck interleaved.

IV. CONVERTIDOR CUK

Otro circuito de electrónica conmutada importante es el convertidor *Cuk*. Al igual que en el caso del *Buck*, es un convertidor DC-DC. Este circuito es algo más complejo que el presentado anteriormente, ya que posee cuatro elementos almacenadores de energía (por lo que se lo denomina un convertidor de orden cuatro) a diferencia de los dos presentes en el convertidor *Buck*. Podemos ver un diagrama de dicho circuito en la Figura 10. En él, tanto la llave como el diodo serán modelados del mismo modo que ya ha sido descrito previamente.

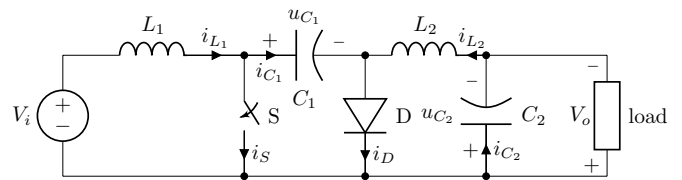


Figura 10: Circuito convertidor Cuk.

En este caso, en ausencia de inductancias parásitas, los métodos de LIQSS no funcionan eficientemente, mientras que

los mLIQSS sí lo hacen [6]. La razón es que si consideramos que la llave y el diodo están abiertos simultáneamente (con $R_{\text{off}} \rightarrow \infty$) entonces aparece una restricción sobre dos variables de estado: $i_{L_1} + i_{L_2} = 0$.

Estudiaremos entonces el caso en que exista una inductancia parásita modelada en serie con la llave de valor $L_s = 10nH$. Para las simulaciones utilizaremos además los siguientes parámetros:

- $U = 24V$ - tensión de entrada
- $C_1 = C_2 = 10mF$ - capacitores
- $L_1 = L_2 = 10mH$ - inductores
- $R = 10\Omega$ - Resistencia de carga
- $f = 10kHz$ - frecuencia de conmutación
- $DC = 0,35$ - ciclo de trabajo
- $R_{\text{off}} = 1 \cdot 10^5 \Omega$ - resistencia de corte
- $R_{\text{on}} = 1 \cdot 10^{-5} \Omega$ - resistencia de conducción
- $t_f = 10mseg$ - tiempo final de simulación
- $tol = 1 \cdot 10^{-3}$ - tolerancias absolutas y relativas de error de simulación

Los resultados obtenidos con los distintos métodos de integración se muestran en la Tabla III.

Tabla III: Comparación tiempos de simulación para convertir Cúk con inductancia parásita en la llave.

Método de Integración	Evaluaciones de función f_i	CPU [mseg]
DASSL	1311312	586
CVODE	96038	16
IDA	30565	16
LIQSS2	—	17153
mLIQSS2	—	6710

Vemos aquí que, aunque la simulación con el método mLIQSS2 mejora lo obtenido con LIQSS, los resultados siguen siendo ineficientes en comparación a los obtenidos utilizando métodos clásicos. Esto se debe a que un pequeño cambio en la corriente por cada uno de los inductores provoca un cambio grande en la derivada de los otros dos inductores, dando lugar a oscilaciones espurias que involucran tres variables de estado, situación que no contempla mLIQSS2. Otra forma de ver esto es que, considerando que el diodo está abierto (con $R_D \rightarrow \infty$) y la llave cerrada, aparece una restricción sobre tres variables de estado: $i_{L_1} + i_{L_2} = i_{L_s}$.

V. CONCLUSIONES

En este trabajo hemos analizado el desempeño de diferentes algoritmos de integración numérica a la hora de simular algunos circuitos de electrónica conmutada en presencia de elementos parásitos. Se han comparado no sólo algunos de los métodos clásicos más utilizados, si no también algunos métodos de la familia QSS. Particularmente, se han utilizado los métodos LIQSS2 y su versión mejorada mLIQSS2.

Del análisis de los resultados, concluimos que:

- La presencia de elementos parásitos en diversos convertidores DC-DC se traduce en un tipo de rigidez que los

métodos LIQSS no son capaces de integrar eficientemente, ya que aparecen oscilaciones espurias entre pares de variables de estado.

- Si el circuito en cuestión cumple con algunas restricciones de topología (que un pequeño cambio en una variable de estado no produzca un gran cambio en las derivadas de más de dos variables de estado), entonces la utilización de mLIQSS2 supera los resultados obtenidos tanto con LIQSS2 como con métodos clásicos.
- Estas condiciones sobre la topología pueden analizarse en función del número de variables de estado que pueden quedar involucradas en una restricción al hacer $R_{\text{off}} \rightarrow \infty$.
- Las ventajas de utilizar el método mLIQSS2 se hace más significativa a medida que el tamaño del circuito a simular aumenta, y como consecuencia de esto, las discontinuidades en el el modelo se hacen más frecuentes y la matriz Jacobiana del mismo crece en tamaño. Esto se debe a que, como los demás miembros de la familia QSS, este método logra integrar a través de discontinuidades de manera muy eficiente.

REFERENCIAS

- [1] G. Migoni, F. Bergero, E. Kofman, and J. Fernández, “Quantization-Based Simulation of Switched Mode Power Supplies,” *Simulation: Transactions of the Society for Modeling and Simulation International*, vol. 91, no. 4, pp. 320–336, 2015.
- [2] F. Cellier and E. Kofman, *Continuous System Simulation*. New York: Springer, 2006.
- [3] E. Kofman and S. Junco, “Quantized State Systems. A DEVS Approach for Continuous System Simulation,” *Transactions of SCS*, vol. 18, no. 3, pp. 123–132, 2001.
- [4] G. Migoni, M. Bortolotto, E. Kofman, and F. Cellier, “Linearly Implicit Quantization-Based Integration Methods for Stiff Ordinary Differential Equations,” *Simulation Modelling Practice and Theory*, vol. 35, pp. 118–136, 2013.
- [5] F. Di Pietro, G. Migoni, and E. Kofman, “Improving a Linearly Implicit Quantized State System Method,” in *Proceedings of the 2016 Winter Simulation Conference*, (Arlington, Virginia, USA), 2016.
- [6] F. D. Pietro, G. Migoni, and E. Kofman, “Improving linearly implicit quantized state system methods,” *Simulation: Transactions of the Society for Modeling and Simulation International*.
- [7] J. Fernández and E. Kofman, “A Stand-Alone Quantized State System Solver for Continuous System Simulation,” *Simulation: Transactions of the Society for Modeling and Simulation International*, vol. 90, no. 7, pp. 782–799, 2014.
- [8] L. R. Petzold, “Description of dassl: a differential/algebraic system solver,” tech. rep., Sandia National Labs., Livermore, CA (USA), 1982.
- [9] A. C. Hindmarsh, R. Serban, and D. R. Reynolds, “User documentation for cvode v3. 1.1 (sundials v3. 1.1),” 2018.
- [10] A. C. Hindmarsh, R. Serban, and A. Collier, “User documentation for ida v3. 1.1 (sundials v3. 1.1),” 2018.
- [11] A. Elbanhawy, “Effect of parasitic inductance on switching performance,” *Proc. PCIM Europe 2003*, pp. 251–255, 2003.
- [12] B. Yang and J. Zhang, “Effect and utilization of common source inductance in synchronous rectification,” in *Applied Power Electronics Conference and Exposition, 2005. APEC 2005. Twentieth Annual IEEE*, vol. 3, pp. 1407–1411, IEEE, 2005.
- [13] S. x Bhat and H. Nagaraja, “Effect of parasitic elements on the performance of buck-boost converter for pv systems,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 4, no. 6, pp. 831–836, 2014.