

UNIVERSIDAD DE ZARAGOZA



GRADO DE FÍSICA

DEPARTAMENTO DE FÍSICA TEÓRICA

SIMULACIÓN NUMÉRICA DE
UNA TEORÍA DE GAUGE ABELIANA
MEDIANTE UN ALGORITMO DE
MONTECARLO GEOMÉTRICO

Curso 2019 - 2020

Autor:

Ignacio GRACIA ALGUERÓ

Profesor:

Dr. Eduardo FOLLANA ADÍN

Índice

1. Introducción	2
2. Objetivo	4
3. Desarrollo	5
3.1. Geometría del sistema	5
3.2. Función de partición	7
3.3. Observables	9
3.4. Algoritmo e implementación	10
4. Resultados numéricos	13
5. Conclusiones	17
6. Bibliografía	18

1. Introducción

Las teorías de gauge juegan un papel fundamental para entender la física de partículas. En ellas se basan teorías de unificación de fuerzas, como puede ser la electro-débil, con un grupo de gauge $SU(2) \times U(1)$. En estas teorías, los campos son representaciones de los grupos de simetría. Si se construye el Lagrangiano de forma invariante bajo dichas simetrías, aparecen cantidades que se conservan, como pueden ser la carga eléctrica, la carga de color o la masa.

En la descripción teórica, esta simetría local se asegura con la introducción de grados de libertad adicionales en forma de campo de gauge. El caso mas sencillo aparece en las teorías de gauge Abelianas, donde los campos de gauge sólo interactúan con la materia y no con ellos mismos. Para las teorías no Abelianas, como las teorías de unificación, la interacción de los campos de gauge con ellos mismos da lugar a una fenomenología mas compleja.

Para simplificar el estudio en ambos casos, se recurre a definir estas teorías en la red.[1] Toda teoría de campos se puede definir en una red reemplazando las derivadas parciales del Lagrangiano por diferencias de operadores finitos. Es decir, las teorías de gauge de red son modelos físicos que se obtienen discretizando el espacio-tiempo continuo. Estas teorías se caracterizan por su grupo de gauge, por su dimensión y por un parámetro llamado constante de acoplamiento. Éste parámetro, es un número que determina la fuerza de una interacción. Cuando el valor de la constante de acoplamiento es pequeño, se dice que la teoría esta en un régimen de acoplamiento débil. Por el contrario, cuando es grande, se dice que la teoría está en un régimen de acoplamiento fuerte.

Mediante esta reformulación, se pretende buscar respuestas a misterios como el confinamiento de los quarks [2]. Los quarks son los constituyentes de varias partículas elementales, como los protones y los neutrones. El problema es el por qué no se pueden observar estas partículas libremente en la naturaleza.[3] Wilson argumentó que el confinamiento de los quarks era

equivalente a demostrar que la teoría de gauge de red satisface la ley de área de Wilson. Para probar esta ley, habría que demostrar que se cumple para todos los valores de fuerza de acoplamiento, sobre todo para los valores más pequeños. Guth [4], entre otros, demostró que para la teoría de gauge de red $U(1)$ en 4 dimensiones, desaparecía el confinamiento para el régimen de acoplamiento débil. Probar esta ley de área para régimen de acoplamiento débil es un problema abierto para sistemas de más de 2 dimensiones. Sin embargo, ocurre un extraño caso para la teoría de gauge de red $U(1)$ para 3 dimensiones. En esta situación, la ley de área de Wilson se mantiene en la región de acoplamiento débil 20.

Para entender el significado de la ley de área, se va a partir de la definición del bucle de Wilson. Estos bucles son factores de fase en teorías de gauge Abelianas o no Abelianas, jugando un papel fundamental en estas teorías. El factor de fase abeliano para un camino Γ_{ab} viene definido por la fórmula [5]:

$$U[\Gamma_{ba}] = e^{ie \int_{\Gamma_{ba}} dz^\mu A_\mu(z)} \quad (1.0.1)$$

Bajo transformaciones de Gauge

$$A_\mu(z) \rightarrow A_\mu + \frac{1}{e} \partial_\mu \alpha(z) \quad (1.0.2)$$

el factor de fase abeliano se transforma en

$$U[\Gamma_{ba}] = e^{i\alpha(b)} U[\Gamma_{ba}] e^{-i\alpha(a)} \quad (1.0.3)$$

Una función de ondas en el punto a se transforma como

$$\phi(x) \rightarrow e^{i\alpha(a)} \phi(x) \quad (1.0.4)$$

por lo que el factor de fase se transformará como el producto de $\phi(y)\phi^\dagger(x)$

$$U[\Gamma_{ba}] \sim \phi(y)\phi^\dagger(x) \quad (1.0.5)$$

El factor de fase juega un rol importante en el transporte paralelo dentro de un campo electromagnético y a la hora de comparar fases de una función de ondas en los puntos x e y.

2. Objetivo

El objetivo de este trabajo es plantear un algoritmo geométrico para la simulación numérica de la teoría de gauge de red $U(1)$ para tres dimensiones. Los resultados obtenidos se compararán con los planteados en el artículo [6]. Se estudiarán las diferencias en los tiempos de simulación de ambos algoritmos y su eficacia. También se hará una comparativa los observables obtenidos junto con sus errores correspondientes.

A nivel teórico, se espera una introducción a las teorías gauge de red, así como a la formulación de la teoría cuántica de campos.

Por último, se pretende reforzar conocimientos del lenguaje de programación C, junto con el aprendizaje de nuevas metodologías de trabajo enfocadas a la simulación de redes.

3. Desarrollo

3.1. Geometría del sistema

Uno de los principales problemas que se debe solventar es la discretización del espacio sobre el que se quiere trabajar. Para ello, se utilizarán las teorías gauge de red, propuestas por Wilson para explicar el confinamiento en QCD. Una red es una aproximación del continuo a partir de puntos discretos del espacio (sitios de la red). Estos están definidos en un espacio Euclídeo d-dimensional, cuyas coordenadas vendrán dadas por:

$$x_\mu = n_\mu a \quad (3.1.1)$$

Donde, para un espacio 3d, $n_\mu = (n_x, n_y, n_z)$ es el vector dirección y a el equiespaciado de la red. De ahora en adelante tomaremos $a = 1$.

Para este trabajo, se han considerado condiciones de contorno periódicas, para poder aproximar la simulación a un sistema de mayores dimensiones. En una red 3d de tamaño L ($V = L^3$), supondrá que sitios con $n_\mu = L$ serían equivalentes a los sitios con $n_\mu = 0$. Se puede apreciar en el siguiente ejemplo ($L = 2$):

Siendo μ y ν dos de las dimensiones de nuestra red.

A partir de 2 sitios de la red (o un sitio y una dirección $l : \{n, \mu\}$) se puede definir el concepto de *link*. La importancia de los *links* se debe a que en ellos están definidos los campos de gauge ($A_\mu(x)$), que describirían la interacción entre partículas (sitios). Esta asignación es:

$$A_\mu(x) \implies U_\mu(x) \quad (3.1.2)$$

Donde la variable *link* viene definida por la integral de camino entre dos sitios consecutivos de la forma:

$$U_\mu(x) = P e^{i \int_x^{x+a\hat{\mu}} dz^\mu A_\mu(z)} \quad (3.1.3)$$

Integrando a lo largo del *link* obtenemos que:

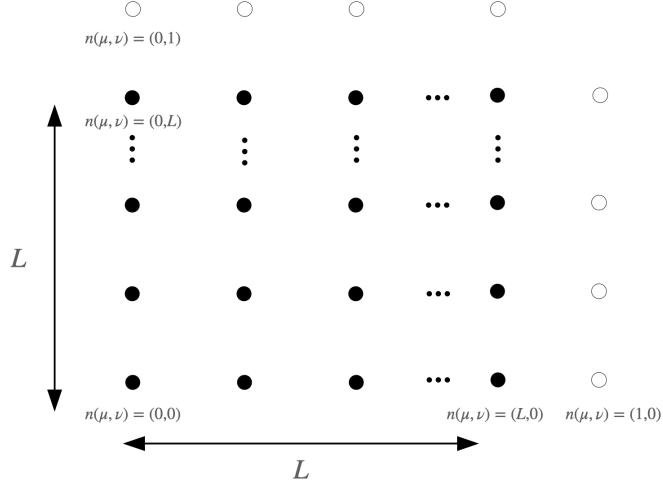


Figura 1: Representación de las condiciones de contorno periódicas para una red 2D.

$$U_\mu(n) = e^{iA_\mu(n)} \quad (3.1.4)$$

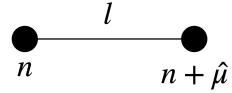


Figura 2: Diagrama básico de construcción de un link

Se define la orientación positiva de un *link* como aquella para el cual $\mu > 0$. Para los *links* con orientación negativa, se tiene en cuenta la relación:

$$U_{-\mu}(n + a\hat{\mu}) = U_\mu^\dagger(n) \quad (3.1.5)$$

Otro concepto de la red que se debe tener en cuenta son las *plaqueas*, que son elementos cuadrados generados por *links* en las direcciones μ y ν . Una *plaquea* queda descrita por un sitio y dos direcciones $p : \{n, \mu, \nu\}$.

La variable *plaquea* se construye a partir de la variable *link*. Esta variable juega un papel muy importante en la teoría de red, ya que está relacionada

con el factor de fase que previamente se ha comentado. Se construye a partir del contorno orientado de una plaqueta, y es de la forma:

$$U_{\mu\nu}(n) = U_\mu(n) \ U_\nu(n + a\hat{\mu}) \ U_\mu^\dagger(n + a\hat{\nu}) \ U_\nu^\dagger(n) \quad (3.1.6)$$

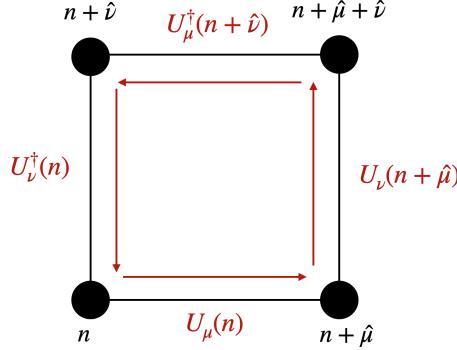


Figura 3: Diagrama de una plaqueta básica, compuesta por 4 links orientados.

Para este trabajo, se ha definido el concepto de *cubo*. Se trata de un elemento de la red formado por 6 plaquetas. Es importante resaltar que dos cubos adyacentes comparten plaqueta. Es decir, cualquier cambio en dicha plaqueta afectará a los dos cubos.

3.2. Función de partición

La función de partición de nuestro sistema vendrá dada por:

$$Z(\beta) = \int \prod_{n,\mu} dU_\mu(n) e^{-\beta S[U]} \quad (3.2.1)$$

Se considera la teoría de Gauge $U(1)$ abeliana y euclídea para la red, de forma que la acción de Wilson será de la forma:

$$S = -\beta Re \sum_{n,\mu,\nu} U_\mu(n) \ U_\nu(n + a\hat{\mu}) \ U_\mu^\dagger(n + a\hat{\nu}) \ U_\nu^\dagger(n) = - \sum_{k=1}^{N_p} \frac{\beta}{2} (U_k + U_k^\dagger) \quad (3.2.2)$$

Donde k es el índice que numera las plaquetas y N_p el número total de plaquetas. Para elaborar el algoritmo, se considera una expansión de acoplamiento fuerte de la función de partición:

$$\begin{aligned} Z &= \int \left[\prod_{\mu,\nu} dU_\mu(n) \right] \prod_k^{N_p} e^{\frac{\beta}{2}(U_k + U_k^\dagger)} = \int \left[\prod_{\mu,\nu} dU_\mu(n) \right] \prod_k^{N_p} e^{\frac{\beta}{2}(e^{i\theta_k} + e^{-i\theta_k})} = \\ &= \int [DU] \prod_k^{N_p} \left[\sum_{n=-\infty}^{\infty} I_n(\beta) e^{in\theta_k} \right] \end{aligned}$$

Sabiendo que los elementos del grupo $U(1)$ son de la forma $U_k = e^{i\theta_k}$ y utilizando la igualdad de Jacobi-Anger. $I_n(\beta)$ es la función de Bessel modificada de orden n y $\theta_k \in [0, 2\pi]$. Este valor será la suma de los θ_l de cada link de una placa, de forma que $\theta_k = \sum_{l=1}^4 \theta_{k,l}$. Podemos reescribir la función de partición como suma de todas las posibles configuraciones $C^\alpha(\beta)$:

$$C^\alpha(\beta) = \prod_k^{N_p} I_{n_k^\alpha}(\beta) \int_0^{2\pi} \prod_{k,l} d\theta_{k,l} \exp \left(i \sum_k n_k \theta_k \right) \quad (3.2.3)$$

El índice n_k^α será el índice de Bessel que tiene la placa k para una configuración α . Analizando la integral que se obtiene, se debe modificar para poder integrar en cada *link*, de forma que:

$$C^\alpha(\beta) = \prod_k^{N_p} I_{n_k^\alpha}(\beta) \int_0^{2\pi} \prod_{k,l} d\theta_{k,l} \exp \left[i \sum_k n_k \left(\sum_l \theta_{k,l} \right) \right] \quad (3.2.4)$$

En cada *link* contribuyen 4 placas (en 3d), de forma que, para un cada *link* (θ_l) tendremos:

$$\int_0^{2\pi} d\theta_l \exp [i(n_a + n_b + n_c + n_d)\theta_l] \quad (3.2.5)$$

Donde n_a, n_b, n_c y n_d son los índices de las funciones de Bessel de cada placa que contiene el *link* que se ha fijado. El único valor no nulo de esta integral aparece cuando:

$$n_l = n_a + n_b + n_c + n_d = 0 \quad (3.2.6)$$

Esta restricción fija las configuraciones admitidas del sistema, siendo el peso de cada una:

$$C^\alpha(\beta) = \prod_k^{N_p} I_{n_k^\alpha}(\beta) \delta_{0,n_l} \quad (3.2.7)$$

De esta forma, la función de partición de partición será la suma a todas las configuraciones admitidas posibles (distintos n_k^α):

$$Z = \sum_\alpha C^\alpha(\beta) \quad (3.2.8)$$

La probabilidad de cada configuración será:

$$w^\alpha = \frac{C^\alpha}{Z} \quad (3.2.9)$$

3.3. Observables

Se estudiará los mismos observables que en el artículo [6], para poder comparar los resultados obtenidos. El cálculo de los mismos se encuentra en el anexo A. El primero es el observable placaeta:

$$\langle P_\square \rangle = \frac{1}{N_P} \partial_\beta \ln(Z) \quad (3.3.1)$$

Para la función de partición planteada EN, obtenemos:

$$\langle P_\square \rangle = \frac{1}{2N_P} \sum_\alpha w_\alpha \left[\sum_j^{N_p} \frac{I_{n_j^\alpha - 1}(\beta)}{I_{n_j^\alpha}(\beta)} + \frac{I_{n_j^\alpha + 1}(\beta)}{I_{n_j^\alpha}(\beta)} \right] \quad (3.3.2)$$

El factor w_α es la probabilidad de cada configuración α . El siguiente observable que se va a estudiar es el calor específico

$$C_V = \partial_\beta \langle P_\square \rangle \quad (3.3.3)$$

A partir de la expresión (3.3.2), se obtiene

$$\begin{aligned}
C_V = & \frac{1}{4N_P} \left\{ \sum_{\alpha} w_{\alpha} \left[\sum_j^{N_p} \frac{1}{I_{n_j^{\alpha}}^2} \left(2 + I_{n_j^{\alpha}} I_{n_j^{\alpha}-2} - I_{n_j^{\alpha}-1}^2 - 2I_{n_j^{\alpha}-1} I_{n_j^{\alpha}+1} + I_{n_j^{\alpha}} I_{n_j^{\alpha}+2} - I_{n_j^{\alpha}+1}^2 \right) \right] + \right. \\
& \left. + \sum_{\alpha} w_{\alpha} \left[\sum_j^{N_p} \left(\frac{I_{n_j^{\alpha}-1}}{I_{n_j^{\alpha}}} + \frac{I_{n_j^{\alpha}+1}}{I_{n_j^{\alpha}}} \right)^2 \right] - \left(\sum_{\alpha} w_{\alpha} \left[\sum_j^{N_p} \left(\frac{I_{n_j^{\alpha}-1}}{I_{n_j^{\alpha}}} + \frac{I_{n_j^{\alpha}+1}}{I_{n_j^{\alpha}}} \right) \right] \right)^2 \right)
\end{aligned} \tag{3.3.4}$$

Se ha omitido la dependencia con β de todas las funciones de Bessel para no saturar más la expresión.

3.4. Algoritmo e implementación

Gracias a la formulación de la mecánica estadística, se puede saber que sólo un pequeño subconjunto de configuraciones posibles contribuirán a los promedios de los observables. Para obtener los valores promedio de nuestros observables, se utilizará una simulación Monte Carlo (MC). La idea de la simulación MC es muestrear el espacio de configuraciones mediante una secuencia estocástica de las mismas. Una configuración está fijada por el conjunto de valores (n_k^{α}) .

La probabilidad de encontrar una configuración C^{α} es proporcional a su peso en la función de partición. Para moverse de una configuración C^{α} a una $C^{\alpha+1}$, se tiene que definir un paso elemental. En este trabajo, consistirá en añadir un cubo elemental en la red, es decir, $(n_k^{\alpha}) \rightarrow (n_k^{\alpha} + 1)$ para cada placa contenida en el cubo. También se define el movimiento opuesto $(n_k^{\alpha}) \rightarrow (n_k^{\alpha} - 1)$. Los valores de n_k^{α} pueden ser negativos, ya que las funciones de Bessel modificadas son simétricas. Se podría asociar la situación de una placa con $n_k^{\alpha} < 0$ con una 'antiplaca', que sería una placa orientada en el sentido opuesto al definido.

Para asegurarse de que el algoritmo cumple con la condición de balance detallado, utilizaremos el método de Metrópolis para definir la probabilidad

de transición entre una configuración y otra. El objetivo de este método es optimizar el número de cambios, maximizando la probabilidad de transición hacia configuraciones que minimizan la acción.

La simulación numérica ha consistido en un código en C separado en dos partes. La primera, se encargaba en generar las distintas configuraciones del sistema, mientras que la segunda se basaba en el cálculo de los observables comentados previamente y un análisis estadístico de los mismos. La generación de las configuraciones se llevaba a cabo mediante el siguiente algoritmo:

```
input : Punto (Cubo) de la red
Generacion de numero aleatorio entre 0 y 1 (rand);
for 0 to MC steps do
    Se añade o quita un cubo elemental.  $C^\alpha \rightarrow C^{\alpha+1}$ ;
    if peso ( $C^{\alpha+1}$ ) / peso ( $C^\alpha$ ) > rand then
        | Se acepta el cambio
    end
end
```

Algorithm 1: Actualizar configuración Bessel

La selección de un cubo de la red se puede realizar de forma aleatoria o de forma ordenada. Los cubos del sistema están indexados de forma única, de forma que se la selección de los mismos sea mas sencilla. En este trabajo se ha optado por la selección aleatoria, aunque esta decisión apenas influye en los resultados obtenidos.

A la hora de implementar el algoritmo, se han tenido en cuenta una serie de consideraciones previas. La primera es el tamaño de la red. Ésta debe de ser lo suficientemente grande para que no se vean involucrados los efectos de tamaño finito. Para entender el alcance de esta limitación, pondremos un ejemplo para 3 dimensiones.

Para redes con $L < 10$, el número de puntos en el borde es mayor que el de puntos en el interior. Cuanto mayor sea la dimensión, mayor tendrá que

ser el tamaño de la red, de forma que haya más puntos en el interior que en el borde. Para que se tengan al menos el 50 % en el interior, el tamaño de la red deberá ser al menos:

$$L_{50} = \frac{2^{\frac{1}{D}+1}}{2^{\frac{1}{D}} - 1} \quad (3.4.1)$$

Donde D es la dimensión del sistema. Por otro lado, cuanto más grande sea el tamaño de la red, más largas deberán ser las simulaciones para alcanzar el equilibrio estadístico.

Las condiciones de contorno periódicas permiten solventar el problema de el efecto de tamaño finito, a la vez de trabajar con redes que no requieran mucho tiempo de simulación. En el caso del algoritmo planteado, se ha elegido una red de tamaño 16^3 para calcular los observables y compararlos con los obtenidos en el artículo.

Otro aspecto a tener en cuenta es la configuración inicial de la simulación. Se puede partir desde una configuración congelada o desde una desordenada. En el primer caso, consistirá en una configuración donde $n_k = 0$ para todas las plaquetas. La configuración desordenada se basa en asignar un valor aleatorio a cada n_k de tal forma que la configuración esté admitida por la restricción (2.2.7). En este trabajo se ha partido de una configuración congelada.

4. Resultados numéricos

Se han realizado simulaciones numéricas para distintas redes de gauge en 3 dimensiones para comprobar que el algoritmo planteado reproduce resultados comparables al algoritmo planteado en el artículo. Para ello, se ha implementado a partir de las ecuaciones facilitadas en el artículo [6], el algoritmo del mismo, con el fin de realizar ambas simulaciones en el mismo dispositivo y comparar los resultados. Es importante destacar que la forma en la que se ha implementado el algoritmo del artículo, en cuanto a código, no será la misma que la que se planteó en el mismo.

Para ambos algoritmos, se ha simulado el modelo de red de gauge $U(1)$ en 3 dimensiones, en el cual encontramos una única fase desde el acoplamiento fuerte al débil. Se medirán los observables definidos anteriormente para un intervalo de β entre 0 y 3, con un cambio de $\delta\beta = 0,2$. La red utilizada es de 16^3 , y se ha dejado termalizar el sistema para $5 \cdot 10^4$ iteraciones. La medición de los observables se ha hecho en un total de 10^5 iteraciones. Una iteración consiste en realizar tantas actualizaciones (pasos de Metrópolis) como plaqetas haya en la red, es decir, $3 \cdot L^3$.

Al generar una configuración a partir de la anterior, es inevitable que estas tengan una cierta autocorrelación, es decir, los observables que se obtengan no serán estadísticamente independientes. Por ello, se ha utilizado el método de cálculo de errores por bloques. De esta forma, los observables de cada bloque serán prácticamente independientes y la estimación del error será más correcta. Agrupando en bloques de 200 medidas, el error de los observables permanece constante y podemos tratarlos como medidas independientes.

Para el observable placa, como se puede ver en la gráfica 4, se obtiene los mismos resultados para ambos algoritmos. Este resultado indica que el algoritmo propuesto es válido para calcular este observable. Los errores del algoritmo del artículo son ligeramente más grandes (un orden de magnitud). El origen de esta diferencia puede deberse a que el algoritmo del artículo tra-

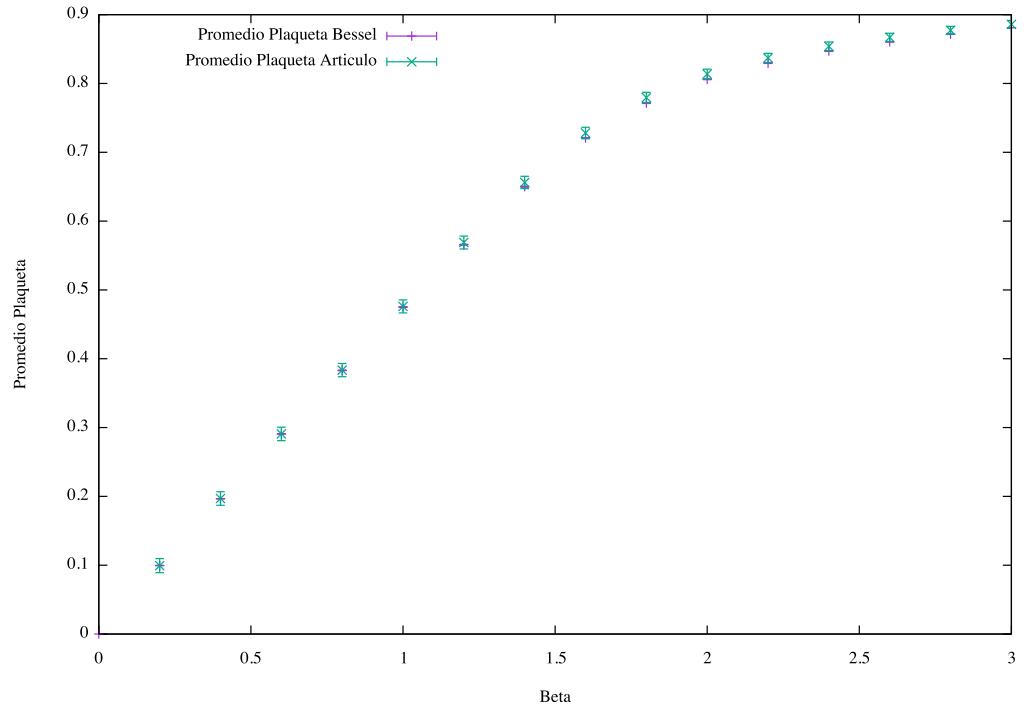


Figura 4: Promedio plaqueta. Red de gauge U(1) de tres dimensiones. Los errores para Bessel son mas pequeños que los símbolos

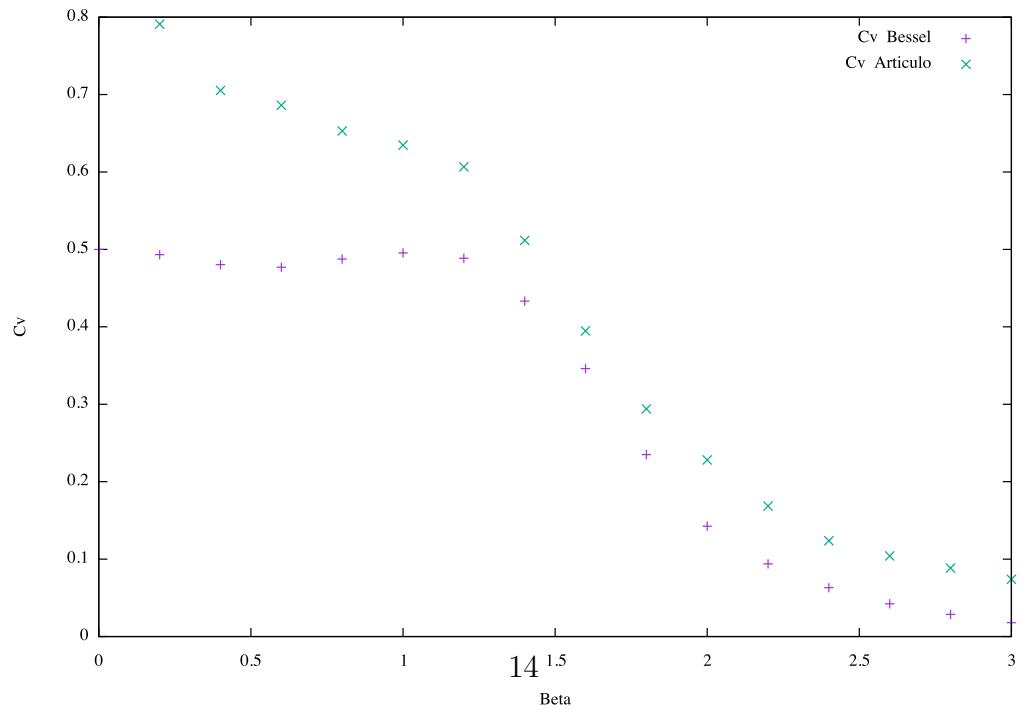


Figura 5: Calor específico. Red de gauge U(1) de tres dimensiones.

baja con números enteros. Una pequeña fluctuación en alguna configuración, tendrá más influencia que en el algoritmo de Bessel, pues este trabaja con funciones decimales.

En cuanto al calor específico, se obtienen valores dispares. Como se ha dicho antes, el algoritmo del artículo trabaja con números enteros. El calor específico es una derivada de segundo orden, por lo que es mucho mas sensible a posibles fluctuaciones. Cabe destacar que los valores recuperados con el algoritmo de Bessel coinciden con los de la gráfica del artículo. En cuanto a los errores, no se han representado en la gráfica 5 ya que todos ellos superaban a la medida por 1 o 2 órdenes de magnitud. Por ello, se seguirá analizando exclusivamente el observable plaqueta para el resto de resultados.

Por último, se ha comparado el rendimiento de ambos algoritmos en cuanto a el tiempo de CPU y los errores obtenidos.

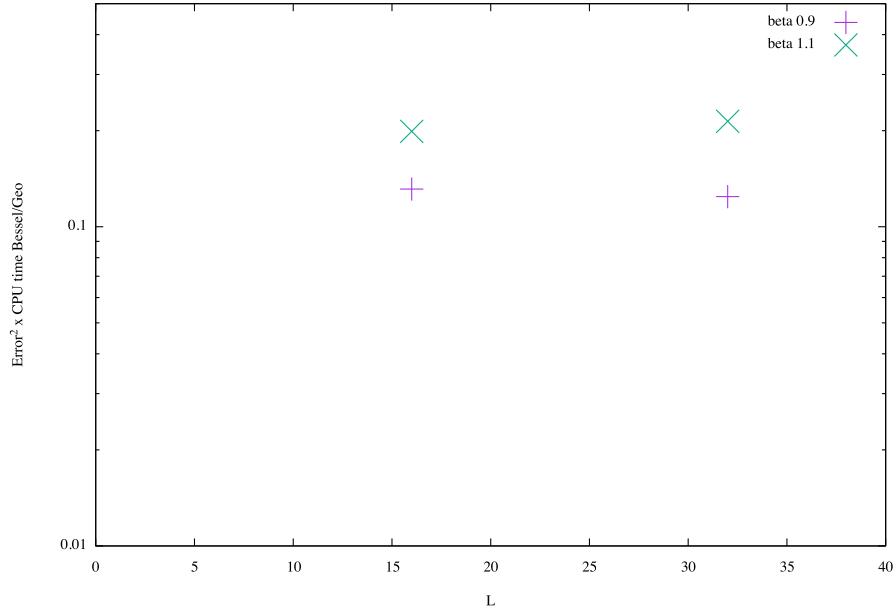


Figura 6: Ratios de funcionamiento entre algoritmos.

Se ha representado el cociente para ambos algoritmos para tamaños de

16 y 32. En ambos casos tiene mejor ratio el algoritmo de Bessel, no tanto por tiempo de CPU sino porque sus errores son mas pequeños.

5. Conclusiones

En este trabajo se han planteado dos objetivos. El primero es la construcción de un algoritmo geométrico basado en Montecarlo para redes de gauge de 3 dimensiones, así como la implementación del mismo en una simulación numérica. Los datos obtenidos en dicha simulación son coherentes con la muestra del artículo [6] facilitado como guión.

Una ventaja del algoritmo elaborado frente al del artículo, es que permite el cálculo de observables para $\beta = 0$, ya que las funciones de Bessel modificadas están definidas en dicho punto.

En cuanto a la comparación de los resultados entre los dos algoritmos, para la placa se ajustan entre sí, teniendo más diferencias en el calor específico.

Con respecto a la eficacia, según la prueba realizada el algoritmo de Bessel parece dar mejores resultados. Cabe decir que el algoritmo del artículo se ha elaborado de nuevo, no se ha utilizado el propio. Esto ha podido provocar pérdidas de rendimiento.

Como trabajo futuro, queda abierta la posibilidad de implementar el algoritmo de Bessel para un sistema de 4 dimensiones, y ver de esta forma cómo funciona ante una transición de fase.

6. Bibliografía

Referencias

- [1] Michael Creutz, Laurence Jacobs, and Claudio Rebbi. Monte carlo computations in lattice gauge theories. *Physics Reports*, 95(4):201 – 282, 1983.
- [2] Kenneth G. Wilson. Confinement of quarks. *Phys. Rev. D*, 10:2445–2459, Oct 1974.
- [3] Sourav Chatterjee. A probabilistic mechanism for quark confinement, 2020.
- [4] Alan H. Guth. Existence proof of a nonconfining phase in four-dimensional u(1) lattice gauge theory. *Phys. Rev. D*, 21:2291–2307, Apr 1980.
- [5] Yu. M. Makeenko. Brief introduction to wilson loops and large n. *Physics of Atomic Nuclei*, 73(5):878–894, May 2010.
- [6] V Azcoiti, E Follana, A Vaquero, and G. Di Carlo. Geometric algorithm for abelian-gauge models. *Journal of High Energy Physics*, 2009(08):008–008, Aug 2009.
- [7] John B. Kogut. An introduction to lattice gauge theory and spin systems. *Rev. Mod. Phys.*, 51:659–713, Oct 1979.
- [8] L. O’Raifeartaigh and N. Straumann. Gauge theory: Historical origins and some modern developments. *Rev. Mod. Phys.*, 72:1–23, 2000.
- [9] M.G.R. Vogelaar. Evaluate bessel function j, y, i, k of integer order.
- [10] H.J. Rothe. *Lattice gauge theories: An Introduction*, volume 43. 1992.