Available online at www.sciencedirect.com

ScienceDirect

Nuclear Physics B 909 (2016) 542–583

NUCLEAR PHYSICS B

www.elsevier.com/locate/nuclphysb

Sampling of general correlators in worm-algorithm based simulations

Tobias Rindlisbacher [a,*], Oscar Åkerlund [a], Philippe de Forcrand [a,b]

[a] *ETH Zürich, Institute for Theoretical Physics, Wolfgang-Pauli-Str. 27, CH-8093 Zürich, Switzerland*
[b] *CERN, Physics Department, TH Unit, CH-1211 Genève 23, Switzerland*

Received 6 April 2016; received in revised form 24 May 2016; accepted 31 May 2016

Available online 6 June 2016

Editor: Hubert Saleur

Abstract

Using the complex ϕ^4-model as a prototype for a system which is simulated by a worm algorithm, we show that not only the charged correlator $\langle\phi^*(x)\phi(y)\rangle$, but also more general correlators such as $\langle|\phi(x)||\phi(y)|\rangle$ or $\langle\arg(\phi(x))\arg(\phi(y))\rangle$, as well as condensates like $\langle|\phi|\rangle$, can be measured at every step of the Monte Carlo evolution of the worm instead of on closed-worm configurations only. The method generalizes straightforwardly to other systems simulated by worms, such as spin or sigma models.

1. Introduction

In a paper from 2001 [1], Prokof'ev and Svistunov proposed the worm algorithm as an alternative to cluster algorithms [2] to overcome critical slowing-down in the simulation of classical spin models like the Ising or the 3 state Potts model, but also of the complex ϕ^4 model.

Critical slowing-down usually occurs at a second order phase transition where the typical length-scale over which the degrees of freedom (e.g. the spins of an Ising system) are correlated, diverges and the system develops long-range order. Local update algorithms then usually become very inefficient due to the high energy barrier that has to be overcome to flip individual

[*] Corresponding author.
E-mail address: rindlisbacher@itp.phys.ethz.ch (T. Rindlisbacher).

http://dx.doi.org/10.1016/j.nuclphysb.2016.05.026
0550-3213/

spins against the field of its nearest neighbors and to the corresponding low acceptance rate of such updates. Cluster algorithms overcome this problem by being non-local: typical structures of correlated sites, the so-called clusters, are grown and then flipped as a whole. Because the typical size of these clusters grows like the correlation length, critical slowing-down is averted. Unfortunately, the typical cluster size usually grows even further when leaving the critical region and going deeper into the ordered phase, resulting in a loss of efficiency of the cluster updates which becomes particularly dramatic when the typical cluster size approaches the size of the whole system.[1] The more relevant problem with cluster algorithms is however that they are in general not useful if the system couples to an external field, as the large energy barrier that has to be overcome to flip a whole cluster *against* the external field quickly leads to very low acceptance rates even for moderate cluster sizes.

Worm algorithms on the other hand are based on local updates and remain relatively efficient even in the ordered phase and in the presence of external fields. For bosonic systems, worm algorithms emerge quite naturally when expressing the partition function in terms of a particular kind of integer-valued "dual" variables, the so-called *flux-variables* (see [1,3–6,10] and Sec. 1.1). The working principle behind this dualization process is as follows: consider the simple system described by the following partition function:

$$Z = \int_0^1 \mathrm{d}\,\phi\, \mathrm{e}^{\phi+\lambda\,\phi} \ . \tag{1.1}$$

If we wanted to use Monte Carlo to compute for example expectation value and variance of the field ϕ, i.e.

$$\langle\phi\rangle = \frac{\partial \log(Z)}{\partial\lambda} \qquad \text{and} \qquad \langle\phi^2\rangle - \langle\phi\rangle^2 = \frac{\partial^2 \log(Z)}{\partial\lambda^2} \ , \tag{1.2}$$

the standard way to do this would be to interpret $\mathrm{e}^{\phi+\lambda\,\phi}$ as probability weight for doing importance sampling with respect to ϕ. To obtain the dual formulation, we instead write the integrand of (1.1) in a power series in ϕ, carry out the integral for each monomial (which can usually be done analytically),

$$w_n(\lambda) = \int_0^1 \mathrm{d}\,\phi \frac{((1+\lambda)\phi)^n}{n!} = \frac{(1+\lambda)^n}{(n+1)!}, \tag{1.3}$$

such that

$$Z = \sum_{n=0}^{\infty} w_n(\lambda), \tag{1.4}$$

and then use the $w_n(\lambda)$ as probability weights for doing importance sampling with respect to the monomial order n, which is our new, integer-valued configuration variable, in terms of which the above observables read

$$\frac{\partial \log(Z)}{\partial\lambda} = \frac{\langle n\rangle}{1+\lambda} \qquad \text{and} \qquad \frac{\partial^2 \log(Z)}{\partial\lambda^2} = \frac{\langle n^2\rangle - \langle n\rangle^2 - \langle n\rangle}{(1+\lambda)^2} \ . \tag{1.5}$$

[1] When the typical clusters consist of almost the whole system, the cluster updates essentially just flip the whole system back and forth.