Journal of Statistical Physics, Vol. 103, Nos. 1/2, 2001

Linked-Cluster Expansion of the Ising Model

Massimo Campostrini[1]

Received May 12, 2000; final November 7, 2000

The linked-cluster expansion technique for the high-temperature expansion of spin modes is reviewed. A new algorithm for the computation of three-point and higher Green's functions is presented. Series are computed for all components of two-point Green's functions for a generalized $3D$ Ising model, to 25th order on the bcc lattice and to 23rd order on the sc lattice. Series for zero-momentum four-, six-, and eight-point functions are computed to 21st, 19th, and 17th order respectively on the bcc lattice.

KEY WORDS: High-temperature expansion; Ising model.

1. INTRODUCTION

The high-temperature (strong-coupling) series expansion is one of the most successful tools for the study of physical systems near a critical point.

High-temperature series are analytic; the radius of convergence is usually quite large, often reaching the boundary of the high-temperature phase. This property allows the application of powerful techniques of resummation and analytical continuation,[1] which can yield very precise and reliable results, provided that long series are available. It is therefore worthwhile to push the computation of high-temperature series as far as our algorithms and computers allow.

The most successful technique for the computation of high-temperature series of $3D$ spin models is the linked-cluster expansion (LCE), which is well suited for the fully computerized approach required to reach very high orders of the expansion.

A review of the most significant series computations for $3D$ O(N)-symmetric spin models up to 1997 can be found in ref. 2. Particularly

[1] INFN, Sezione di Pisa, and Dipartimento di Fisica dell'Università di Pisa; e-mail: Massimo.Campostrini@df.unipi.it

0022-4715/01/0400-0369$19.50/0

remarkable is the computation of 21st-order series of the two-point function for the $3D$ Ising model ($N=1$) on the bcc lattice by Nickel,[3] generalized to arbitrary potential (cf. Eq. (1)) by Nickel and Rehr,[4] using the edge-renormalized LCE; they had not been surpassed until the present work.

Among more recent developments, we mention the works by Butera and Comi, who computed the two-point function to 21st order and the four-point function to 17th order, for sigma models of arbitrary N on the sc and bcc lattices,[2, 5] and q-point functions for $q>4$ to 17th order for the Ising model on the sc lattice.[6] A collaboration including the author of the present work computed series for arbitrary potential: two-point functions are computed to 20th order, four-point functions to 18th order, six-point functions to 17th order etc., for $N=1$[7] and $N=2$.[8] All these works use the vertex-renormalized LCE.

The techniques used in the present paper are similar to those of refs. 3 and 4; we benefitted of a great improvement in computers, but we also worked on the optimization of data representation and adopted new algorithms and tricks (the most notable ones are reported in Section 8). The computation of the q-point functions for $q>2$ uses a new algorithm. Compared to refs. 2, 5–8, the main source of improvement is the adoption of the edge-renormalized LCE.

Several detailed discussions of the LCE appeared in the literature. Wortis' review[9] covers most of the basic topics, and provides many graphical rules fit for algorithmic implementation. Nickel and Rehr illustrate their remarkable computation for the $3D$ Ising model on the bcc lattice, and present several clever algorithms which we found very useful.[4] Lüscher and Weisz, describing their application of the LCE to lattice field theory, also provide several important implementation hints.[10]

Unfortunately, the notation found in the literature is by no means uniform. Therefore we will review the relevant aspects of the LCE, which can be found in ref. 9, not only to make our paper more self-contained, but also to explain notations carefully and to remark the correspondence with refs. 9, 4, and 10. We will follow the notations of ref. 4 whenever possible.

The paper is organized as follows:

Section 2 introduces the relevant graph theory concepts and definitions. Section 3 presents the generalized Ising model we focus on. Sections 4–6 review the LCE, with special focus on the two-point Green's functions. Section 7 describes our algorithm for the computation of three-point and higher Green's functions. Section 8 is devoted to programming details. Section 9 displays a (small) selection of the series generated.

Forthcoming papers will be devoted to the analysis of the series, using the techniques presented in ref. 7, and to the generation and analysis of the series for XY systems.

We will not give proofs of our formulae. The only nontrivial step in the proofs of Sections 5–7 is to show that the symmetry factors compensate exactly the different number of contributions that may appear on the two sides of the equations; it is typically a straightforward, if tedious, exercise in combinatorics. The proofs of Sections 4–6 are given or sketched in ref. 9. The proofs of Section 7 are especially easy, since the symmetry factor of a 1-irreducible tree graph is always 1.

2. GRAPHOLOGY

In this section we introduce a number of graph theory concepts relevant for the LCE. We refer the reader to ref. 11 for a comprehensive introduction to the subject.

A *graph* is a set of *vertices* and *edges* (also named *links* or *bonds* in the literature). Each edge l is *incident* with two distinct vertices, its extrema (we do not allow the extrema to coincide); the set of extrema will be denoted by ∂l; we write $\partial l = \{i(l), f(l)\}$; the choice of an "initial" and a "final" vertex is arbitrary. Two vertices are *adjacent* if they are the extrema of the same edge. We will denote the number of vertices and edges of a graph by v and e respectively. We also consider *arcs* or *oriented edges*, incident out of the initial vertex $i(l)$ into the final vertex $f(l)$.

The *valence* $n(i)$ of a vertex i is the number of edges incident with i.

An *r-rooted* graph is a graph with v vertices, $v \geqslant r$: r *roots* or *external vertices* and $v-r$ *internal vertices*. We will assign the indices 1,..., r to the roots and the indices $r+1$,..., v to the internal vertices. In the drawings, roots will be denoted by open dots and internal vertices by filled dots.

Two r-rooted graphs are *isomorphic* if there exists a one-to-one correspondence π of their internal vertices and edges such that the incidence relations are preserved, i.e., $\partial(\pi(l)) = \{\pi(i(l)), \pi(f(l))\}$ ($\pi(i) = i$ for the roots). From now on, we will identify isomorphic graph, and silently assume that all sets of graphs we define contain only non-isomorphic graphs.

The *symmetry factor* $S(\mathscr{G})$ of an r-rooted graph $\mathscr{G}$ is the number of isomorphisms of $\mathscr{G}$ into itself, i.e., the number of of permutations of internal vertices and edges preserving the incidence relations.

We will also consider *p-ordered r-rooted graphs*, $p \leqslant r$, i.e., the classes of rooted graphs isomorphic up to permutations of $r-p$ roots. We will assign indices 1,..., p to the fixed roots and indices $p+1$,..., r to the roots which can be permuted. The symmetry factor $S(\mathscr{G})$ is defined as the number of isomorphisms of $\mathscr{G}$ into itself, including permutation of the roots $p+1$,..., r. The symmetry factor divided by $(r-p)!$ is called the *modified symmetry factor* $S_E(\mathscr{G})$ (cf. ref. 10); it need not be an integer. The r-rooted

graphs defined above are *ordered* (r-ordered); unless otherwise specified, we will assume that graphs are ordered. 0-ordered graphs are *unordered*.

We will also discuss (p-ordered) r-rooted graphs whose edges and/or vertices are assigned a label; let us consider e.g., the case of an edge label $a(l)$ and vertex label $b(i)$. Two labelled graphs $(\mathscr{G}, a, b)$ and $(\mathscr{G}', a', b')$ are equivalent (isomorphic) if there exists an isomorphism of $\mathscr{G}$ into $\mathscr{G}'$ such that a is mapped into a' and b is mapped into b'. The symmetry factor $S(\mathscr{G}, a, b)$ is the number of isomorphisms of $(\mathscr{G}, a, b)$ into itself.

A pair of vertices i and j is *connected* if there exists a sequence of vertices $k_1,..., k_n$, with $k_1 = i$ and $k_n = j$, and a sequence of edges $l_1,..., l_{n-1}$ such that $\partial l_a = \{k_a, k_{a+1}\}$, $a = 1,..., n-1$. A graph is *connected* if every pair of its vertices is connected. In the following, unless otherwise noted, we will assume that every graph is connected.

A sequence of distinct vertices $k_1,..., k_n$ and distinct edges $l_1,..., l_n$ is called a *loop* of length n if $\partial l_a = \{k_a, k_{a+1}\}$, $a = 1,..., n-1$, and $\partial l_n = \{k_n, k_1\}$. The number of independent loops (also known as *cyclomatic number*) of a connected graph is $e - v + 1$.

A connected graph is called a *tree graph* if it contains no loops. A tree graph has $v - 1$ edges.

3. THE MODEL

We wish to compute the high-temperature (HT) expansion of the q-point functions of a generalized Ising model on a D-dimensional Bravais lattice Λ; notice that Bravais lattices enjoy inversion symmetry at each lattice site. The model is defined by the generating functional

$$\exp(W[h]) = \frac{1}{Z} \prod_i \left[\int d\phi_i \, f(\phi_i) \exp(h_i \phi_i) \right] \exp\left(K \sum_{\langle ij \rangle} \phi_i \phi_j \right) \tag{1}$$

where ϕ_i is a scalar field, f is an even non-negative function or distribution decreasing faster than $\exp(-\phi^2)$ as $\phi \to \infty$, normalized by the condition

$$\int d\phi \, f(\phi) = 1$$

$K = \beta J$, the sum runs over all pairs of nearest neighbours, and the normalization Z is fixed by requiring $W[0] = 0$.

The connected q-point function (denoted by $\mathscr{M}$ in ref. 9) at zero magnetic field is defined by

$$G_q(x_{i_1},..., x_{i_q}) \equiv \langle \phi_{i_1} \cdots \phi_{i_q} \rangle^{\text{con}} |_{h=0} = \left. \frac{\partial^q W[h]}{\partial h_{i_1} \cdots \partial h_{i_q}} \right|_{h=0} \tag{2}$$

where $x_i \equiv (x_1(i),..., x_D(i))$ is the coordinate vector of the lattice site i. G_q is invariant under the lattice symmetry group, including (discrete) translations, and under permutation of its arguments; it is customary to write $G_2(x_{i_1}, x_{i_2})$ in the form $G_2(x_{i_1} - x_{i_2})$. We will apply the LCE to the computation of G_q.

4. UNRENORMALIZED EXPANSION

Let us parametrize the potential f in terms of the bare vertices $\mu_0(2n)$, defined by the generating function

$$\exp\left[\sum_n \frac{\mu_0(2n)}{(2n)!} h^{2n}\right] = \int d\phi\ f(\phi) \exp(h\phi) \tag{3}$$

These quantities are named bare semi-invariants and denoted by M_n^0 in ref. 9; they are named cumulant moments and denoted by μ_{2n} in ref. 4; $\mu_0(2n) = (2n-1)!!\ \mathring{m}_{2n}^{\text{con}}$ in the notations of ref. 10. Without loss of generality, we can rescale ϕ and K to fix $\mu_0(2) = 1$.

For a generic r-rooted graph $\mathscr{G}$, we define the bare external vertex factor

$$V_0^{(e)}(n_1,..., n_r; \mathscr{G}) = \prod_{i=1}^{r} \mu_0(n(i) + n_i) \tag{4}$$

the bare internal vertex factor

$$V_0^{(i)}(\mathscr{G}) = \prod_{i=r+1}^{v} \mu_0(n(i)) \tag{5}$$

and the bare edge factor

$$L_0(x_1,..., x_v; \mathscr{G}) = \prod_{l=1}^{e} [K\theta(x_{i(l)} - x_{f(l)})] \tag{6}$$

where $\theta(x) = \delta(\|x\| - 1)$ and $\|x\|$ is the lattice distance between 0 and x.

It is convenient to focus on the contributions of r-rooted graphs to q-point functions. To this purpose we introduce the auxiliary r-point functions X, whose unrenormalized LCE is

$$X_r(x_1,..., x_r; n_1,..., n_r) = \sum_{\mathscr{G} \in \mathscr{I}^{(r,0)}} \sum_{x_{r+1},..., x_v} \frac{V_0^{(e)}(n_1,..., n_r; \mathscr{G})\ V_0^{(i)}(\mathscr{G})\ L_0(x_1,..., x_v; \mathscr{G})}{S(\mathscr{G})} \tag{7}$$

where $\mathscr{I}^{(r,0)}$ is the set of all r-rooted connected graphs.

X is invariant under simultaneous permutation of coordinates and valences:

$$X_r(x_1,..., x_r; n_1,..., n_r) = X_r(x_{\pi(1)},..., x_{\pi(r)}; n_{\pi(1)},..., n_{\pi(r)})$$

but not over independent permutation of coordinates and valences. Furthermore, X is invariant over the lattice symmetry group, e.g., it is translation-invariant:

$$X_r(x_1,..., x_r; n_1,..., n_r) = X_r(x_1 + x,..., x_r + x; n_1,..., n_r)$$

$X_1(x; n)$ is independent of x, and it will be denoted by $X_1(n)$; it will also be denoted by $\mu(n)$ in its role of renormalized vertex. $X_2(x_1, x_2; n_1, n_2)$ only depends on the difference $x_2 - x_1$, and it will be denoted by $X_2(x_2 - x_1; n_1, n_2)$. Since, by invariance under space inversion, $X_2(x; n_1, n_2) = X_2(-x; n_1, n_2)$, we also have $X_2(x; n_1, n_2) = X_2(x; n_2, n_1)$.

The sum over the location of internal vertices of the θ functions is by definition the (free) lattice embedding number of $\mathscr{G}$ with fixed roots:

$$\sum_{x_{r+1},..., x_v} \prod_{l=1}^{e} \theta(x_{i(l)} - x_{f(l)}) = E(x_1,..., x_r; \mathscr{G}) \tag{8}$$

Therefore

$$X_r(x_1,..., x_r; n_1,..., n_r) = \sum_{\mathscr{G} \in \mathscr{F}^{(r,0)}} \frac{V_0^{(e)}(n_1,..., n_r; \mathscr{G})\ V_0^{(i)}(\mathscr{G})\ K^{e(\mathscr{G})} E(x_1,..., x_q; \mathscr{G})}{S(\mathscr{G})} \tag{9}$$

Finally, the q-point functions are computed as:

$$G_q(x_1,..., x_q) = \sum_{\text{partitions}} X_r(x_{i_{11}},..., x_{i_{r1}}; u_1,..., u_r) \prod_{l=1}^{r} \delta_{u_l}(x_{i_{l1}},..., x_{i_{lu_l}}) \tag{10}$$

where the q-point delta function is

$$\delta_1(x_1) = 1, \qquad \delta_2(x_1, x_2) = \delta(x_1 - x_2),..., \delta_q(x_1,..., x_q) = \prod_{l=2}^{q} \delta(x_1 - x_l)$$

and $\{\{i_{11},..., i_{1u_1}\},..., \{i_{r1},..., i_{ru_r}\}\}$ is a generic partition of $\{1,..., q\}$ into r sets of size $u_1,..., u_r$. We will call a root bearing a factor of δ_u a uth *order* root.

The two-point function is simply

$$G_2(x) = X_2(x; 1, 1) + \delta(x)\, X_1(2)$$

5. VERTEX-RENORMALIZED EXPANSION

The graph $\mathcal{G}\backslash i$ is obtained by *deleting* from $\mathcal{G}$ the vertex i, i.e., by removing i and all edges incident with i. A vertex i of a rooted graph $\mathcal{G}$ is called an *articulation point* if there exist vertices of $\mathcal{G}\backslash i$ not connected to a root. A rooted graph is called 1-*irreducible* if it does not contain any articulation point.

Any r-rooted ($r > 1$) connected graph $\mathcal{G}$ can be decomposed in a unique way into a 1-irreducible r-rooted 1-*skeleton* $\mathcal{S}$ and a 1-rooted 1-*decoration* for each vertex; $\mathcal{G}$ is reconstructed by decorating each vertex, identifying the root of its decoration with the vertex; an example is presented in Fig. 1.

Since the only 1-irreducible 1-rooted graph is the single-vertex graph, we use a different definition for 1-rooted graphs. A 1-rooted graph is called a 1-*skeleton* if it has no articulation points *except the root*. Any 1-rooted connected graph $\mathcal{G}$ can be decomposed in a unique way into a 1-rooted 1-*skeleton* $\mathcal{S}$ and a 1-rooted 1-*decoration* for each vertex *except the root*, which is left undecorated.

A 1-rooted connected graph is called a 1-*insertion* if $\mathcal{G}\backslash 1$ is connected.

The LCE can be reorganized by summing together all contributions from graphs having the same 1-skeleton, incorporating 1-decorations into *renormalized vertices* $\mu(n) = X_1(n)$ (named *semi-invariants* and denoted by M_n in ref. 9; $\mu(n) = (n-1)!!\, m_n$ in the notations of ref. 10). The unrenormalized LCE of $\mu(n)$ is given by Eq. (7).

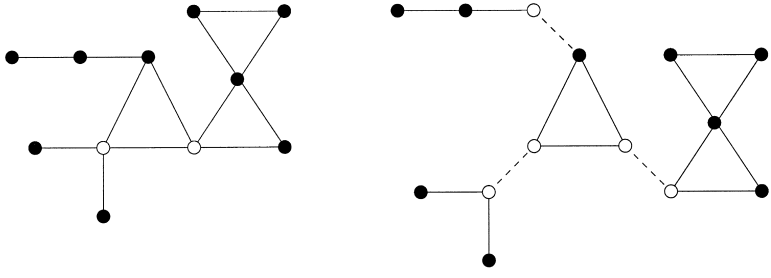

Fig. 1. Example of decomposition of a graph into its 1-skeleton and 1-decorations. Open dots and roots, filled dots are internal vertices.

The r-point function can be computed restricting the sum in Eq. (7) or (9) to the (much smaller) set $\mathscr{I}^{(r,1)}$ of 1-irreducible r-rooted graphs:

$$\begin{aligned} X_r(x_1,\ldots,x_r;n_1,\ldots,n_r) &= \sum_{\mathscr{G}\in\mathscr{I}^{(r,1)}} \sum_{x_{r+1},\ldots,x_v} \frac{V_1^{(e)}(n_1,\ldots,n_r;\mathscr{G})\, V_1^{(i)}(\mathscr{G})\, L_0(x_1,\ldots,x_v;\mathscr{G})}{S(\mathscr{G})} \\ &= \sum_{\mathscr{G}\in\mathscr{I}^{(r,1)}} \frac{V_1^{(e)}(n_1,\ldots,n_r;\mathscr{G})\, V_1^{(i)}(\mathscr{G})\, K^{e(\mathscr{G})} E(x_1,\ldots,x_r;\mathscr{G})}{S(\mathscr{G})} \end{aligned} \tag{11}$$

where the internal and external renormalized vertex factors are

$$V_1^{(e)}(n_1,\ldots,n_r;\mathscr{G}) = \prod_{i=1}^{r} \mu(n(i)+n_i) \tag{12}$$

and

$$V_1^{(i)}(\mathscr{G}) = \prod_{i=r+1}^{v} \mu(n(i)) \tag{13}$$

Equation (7) requires a sum over all connected graphs, and therefore it is impractical for the computation of $\mu(n)$ at large orders of the LCE. We introduce the *renormalized moments* $q(n)$ (named *self-fields* and denoted by G_n in ref. 9), defined by

$$q(n) = \sum_{\mathscr{G}\in\mathscr{I}_{n,\mathrm{in}}^{(1,0)}} \sum_{x_2,\ldots,x_v} \frac{V_0^{(i)}(\mathscr{G})\, L_0(x_1,\ldots,x_v;\mathscr{G})}{S(\mathscr{G})} \tag{14}$$

where $\mathscr{I}_{n,\mathrm{in}}^{(1,0)}$ is the set of 1-insertions with root of valence n. The following equations hold:

$$q(n) = \sum_{\mathscr{G}\in\mathscr{I}_{n,\mathrm{in}}^{(1,1)}} \sum_{x_2,\ldots,x_v} \frac{V_1^{(i)}(\mathscr{G})\, L_0(x_1,\ldots,x_v;\mathscr{G})}{S(\mathscr{G})} \tag{15}$$

where $\mathscr{I}_{n,\mathrm{in}}^{(1,1)} = \mathscr{I}_{n,\mathrm{in}}^{(1,0)} \cap \mathscr{I}^{(1,1)}$, and $\mathscr{I}^{(1,1)}$ is the set of 1-rooted 1-skeletons;

$$\mu(n) = \mu_0(n) + \sum_{s=1}^{\infty} \frac{1}{s!} \sum_{l_1=1}^{\infty} \cdots \sum_{l_s=1}^{\infty} q(l_1)\cdots q(l_s)\, \mu_0(n+l_1+\cdots+l_s) \tag{16}$$

Since $q(2n-1)=0$, and odd values of n and l_i do not contribute to Eq. (16), q and μ can now be computed recursively in parallel order by

order in K, since, once Eq. (15) is expanded in powers of K and truncated, the coefficient of the highest power of K in the l.h.s. depends only on lower-order approximations of μ.

6. EDGE-RENORMALIZED EXPANSION

A pair of distinct vertices i and j of a rooted graph $\mathscr{G}$ is called an *articulation pair* if there exist vertices of $\mathscr{G}\backslash i\backslash j$ not connected to a root, or if i and j are joined by more than one edge. A rooted graph is called 2-*irreducible* if it does not contain any articulation pair.

Any 1-irreducible r-rooted ($r>2$) graph $\mathscr{G}$ can be decomposed in a unique way into a 2-irreducible r-rooted 2-*skeleton* $\mathscr{S}$ and a 1-irreducible 2-rooted 2-*decoration* for each edge (oriented in a canonical way, e.g., by choosing $i(l)<f(l)$); $\mathscr{G}$ is reconstructed by replacing each edge with its decoration, identifying the first and second decoration root with the initial and final vertex of the edge respectively. An example is shown in Fig. 2.

We use a different definition for 2-rooted graphs, since the only 2-irreducible 2-rooted graph is the *bond* graph (no internal vertices and only one edge); the roots of all other 1-irreducible 2-rooted graphs are an articulation pair. We call a 1-irreducible 2-rooted graph a 2-*skeleton* if it does not contain any articulation pair *except the pair consisting of the two roots*. Any 1-irreducible 2-rooted graph can be decomposed in a unique way into a 2-rooted 2-*skeleton* $\mathscr{S}$ and a 1-irreducible 2-rooted 2-*decoration* for each edge *except edges connecting the roots*, which are left undecorated.

The LCE can be reorganized by summing together all contributions from graphs having the same 2-skeleton, incorporating all 2-decorations into *renormalized edges*.

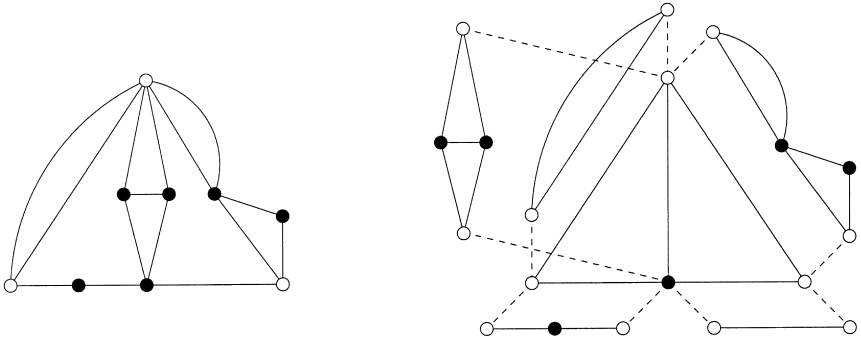

Fig. 2. Example of decomposition of a graph into its 2-skeleton and 2-decorations. Open dots and roots, filled dots are internal vertices.

We start by decomposing Eq. (11) for $r > 1$ into

$$W_r(x_1,..., x_r; n_1,..., n_r) = \sum_{\mathscr{G} \in \mathscr{I}^{(r,1)}_{n_1,..., n_r}} \sum_{x_{r+1},..., x_v} \frac{V_1^{(i)}(\mathscr{G})\, L_0(x_1,..., x_v; \mathscr{G})}{S(\mathscr{G})} \tag{17}$$

$$X_r(x_1,..., x_r; n_1,..., n_r) = \sum_{s_1,..., s_r} \left[\prod_{i=1}^{r} \mu(n_i + s_i) \right] W_r(x_1,..., x_r; s_1,..., s_r) \tag{18}$$

where $\mathscr{I}^{(r,1)}_{n_1,..., n_r}$ is the set of 1-irreducible r-rooted graphs with roots of valence $n_1,..., n_r$. W_r enjoys the same symmetry properties of X_r. $W_1(x, n)$ is undefined.

W_r can be computed by assigning an initial and a final valence i_l, f_l to each oriented edge of a 2-rooted graph; the valence i_l is incident with $i(l)$ and f_l is incident with $f(l)$:

$$\begin{aligned} &W_r(x_1,..., x_r; n_1,..., n_r) \\ &\quad = \sum_{\mathscr{G} \in \mathscr{I}^{(r,2)}} \sum_{x_{r+1},..., x_v;\, i_1,..., i_e,\, f_1,..., f_e} \prod_{i=1}^{r} \delta(n_i - v_i(i_1,..., i_e, f_1,..., f_e; \mathscr{G})) \\ &\qquad \times \frac{V_2^{(i)}(i_1,..., i_e, f_1,..., f_e; \mathscr{G})\, L_2(x_1,..., x_v; i_1,..., i_e, f_1,..., f_e; \mathscr{G})}{S(\mathscr{G})} \end{aligned} \tag{19}$$

where $\mathscr{I}^{(r,2)}$ is the set of 2-irreducible r-rooted graphs, v_i is the sum of all the valences incident with the vertex i,

$$V_2^{(i)}(i_1,..., i_e, f_1,..., f_e; \mathscr{G}) = \prod_{i=r+1}^{v} \mu(v_i(i_1,..., i_e, f_1,..., f_e; \mathscr{G})) \tag{20}$$

and

$$L_2(x_1,..., x_v; i_1,..., i_e, f_1,..., f_e; \mathscr{G}) = \prod_{l=1}^{e} W_2(x_{i(l)} - x_{f(l)}; i_l, f_l) \tag{21}$$

For $r = 2$ Eqs. (19) and (21) require a slight modification: we define $\mathscr{I}^{(2,2)}$ as the set of 2-rooted 2-skeletons; for edges incident with the two roots, we replace $W_2(x_1 - x_2; i, f)$ with $K\theta(x_1 - x_2)\, \delta_{i1}\, \delta_{f1}$.

The sum over graphs can be restricted to a subset of $\mathscr{I}^{(r,2)}$; we will discuss here the case $r = 2$; the next Section will be devoted to the case $r \geqslant 3$. Let us start by classifying 1-irreducible 2-rooted graphs into several classes.

An internal vertex i of a 1-irreducible 2-rooted graph $\mathscr{G}$ is called a *nodal point* if the roots of $\mathscr{G} \backslash i$ are not connected. A 1-irreducible 2-rooted

graph is *nodal* (also named *articulated* or *separable* in the literature) if it contains one or more nodal points; otherwise it is *non-nodal*.

A 1-irreducible 2-rooted graph is *simple* if $\mathscr{G}\backslash 1\backslash 2$ is connected, and 1 is not adjacent to 2. By definition, all nodal graphs are simple. A 1-irreducible 2-rooted graph is a *ladder* graph if it is not simple and it is not the bond graph.

A 1-irreducible 2-rooted graph is *elementary* if it is both simple and non-nodal.

We have divided 1-irreducible 2-rooted graphs into four disjoint classes: bond, nodal, ladder, and elementary graphs. Let us separate the contributions to W_2 according to the four classes:

$$W_2(x; n_1, n_2) = W_2^{\text{bo}}(x; n_1, n_2) + W_2^{\text{no}}(x; n_1, n_2) + W_2^{\text{la}}(x; n_1, n_2) + W_2^{\text{el}}(x; n_1, n_2)$$

i.e., bond, nodal, ladder, and elementary contributions respectively.

The bond contribution is trivial. Nodal contributions can be factorized into a product of non-nodal contributions:

$$\begin{aligned} W_2^{\text{no}}(x; n_1, n_2) = & \sum_{x_3;\, i_1,\, i_2} W_2^{\text{nn}}(x_3; n_1, i_1)\, \mu(i_1 + i_2)\, W_2^{\text{nn}}(x - x_3; i_2, n_2) \\ & + \sum_{x_3,\, x_4;\, i_1,\, i_2,\, i_3,\, i_4} W_2^{\text{nn}}(x_3; n_1, i_1)\, \mu(i_1 + i_2) \\ & \times W_2^{\text{nn}}(x_4 - x_3; i_2, i_3)\, \mu(i_3 + i_4)\, W_2^{\text{nn}}(x - x_4; i_4, n_2) + \cdots \end{aligned} \tag{22}$$

which can be written recursively as

$$W_2^{\text{no}}(x; n_1, n_2) = \sum_{x_3;\, i_1,\, i_2} W_2^{\text{nn}}(x_3; n_1, i_1)\, \mu(i_1 + i_2)\, W_2(x - x_3; i_2, n_2) \tag{23}$$

Likewise, ladder contributions can be factorized into a product of non-ladder contributions:

$$\begin{aligned} W_2^{\text{la}}(x; n_1, n_2) = & \sum_{s=2}^{\infty} \frac{1}{s!} \sum_{i_1, \ldots, i_s,\, f_1, \ldots, f_s} \delta\left(n_1 - \sum_{t=1}^{s} i_t\right) \\ & \times \delta\left(n_2 - \sum_{t=1}^{s} f_t\right) \prod_{t=1}^{s} W_2^{\text{nl}}(x, i_t, f_t) \end{aligned} \tag{24}$$

Elementary contributions can be computed by setting $r = 2$ into Eq. (19), and restricting the sum to $\mathscr{I}_{\text{el}}^{(2,2)}$, the set of *elementary* 2-rooted

2-skeletons. The sum can be further restricted to $\mathscr{I}^{(2,2)}_{0,\mathrm{el}}$, the set of *unordered* elementary 2-rooted 2-skeletons, provided that we replace $S(\mathscr{G})$ with $S_E(\mathscr{G})$ and we symmetrize the result:

$$
\begin{aligned}
&W_2^{\mathrm{el}}(x; n_1, n_2) \\
&\quad = \frac{1}{2} \sum_{\mathscr{G} \in \mathscr{I}^{(2,2)}_{0,\mathrm{el}}} \sum_{x_3,\ldots,x_v;\, i_1,\ldots,i_e,\, f_1,\ldots,f_e} \prod_{l=1}^{2} \delta(n_i - v_i(i_1,\ldots,i_e, f_1,\ldots,f_e; \mathscr{G})) \\
&\qquad \times \frac{V_2^{(i)}(i_1,\ldots,i_e, f_1,\ldots,f_e; \mathscr{G})\, L_2(x_1,\ldots,x_v; i_1,\ldots,i_e, f_1,\ldots,f_e; \mathscr{G})}{S_E(\mathscr{G})} \\
&\qquad + n_1 \leftrightarrow n_2
\end{aligned}
\tag{25}
$$

The last ingredient we need for a fully edge-renormalized expansion are the renormalized vertices $\mu(n)$; they can be computed by combining Eq. (16) with

$$
q(n_1+n_2) = \frac{n_1!\, n_2!}{(n_1+n_2)!} \left(W_2^{\mathrm{no}}(0; n_1, n_2) + W_2^{\mathrm{el}}(0; n_1, n_2) \right) \tag{26}
$$

reflecting the fact that the contributions to $q(n_1+n_2)$ in Eq. (15) can be obtained from the contributions to $W_2(0; n_1, n_2)$ in Eq. (17) by *identifying* the roots of the 2-rooted graph, i.e., by suppressing the second root and reattaching all the edges incident with it to the first root, provided that the roots are not adjacent.

Expanding in powers of K Eqs. (23)–(26), and (16), we can compute $W_2^{\mathrm{el}}(x; n_1, n_2)$, $W_2^{\mathrm{no}}(x; n_1, n_2)$, $W_2^{\mathrm{la}}(x; n_1, n_2)$, $q(n)$, and $\mu(n)$ in parallel order by order in K. The only step which involves a summation over graphs is Eq. (25), where the sum only runs over $\mathscr{I}^{(2,2)}_{0,\mathrm{el}}$, a relatively small set.

7. THREE-POINT AND HIGHER FUNCTIONS

A vertex (internal or external) i of a 1-irreducible r-rooted graph $\mathscr{G}$ is called a *nodal point* if $\mathscr{G} \backslash i$ is not connected. By definition of 1-irreducibility, each connected component of $\mathscr{G} \backslash i$ must contain at least one root. A 1-irreducible r-rooted graph $\mathscr{G}$ is *nodal* if it contains one or more nodal points; otherwise it is *non-nodal*.

In the rest of this section, we will assume that every graph is 2-irreducible.

A nodal point j of a 2-irreducible r-rooted graph $\mathscr{G}$ is called a *tree-insertion point* if at least one of the connected components of $\mathscr{G} \backslash j$ is a tree

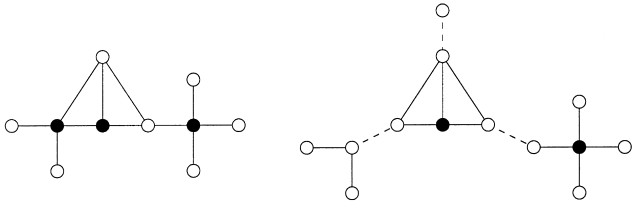

Fig. 3. Decomposition of a 2-irreducible graph into a compact kernel and 2-irreducible tree graphs.

graph. The order t of j is the number of roots of $\mathcal{G}$ contained in all tree graph components of $\mathcal{G}\backslash j$, plus 1 if j itself is a root. A 2-irreducible r-rooted graph is called *compact* if it contains no tree-insertion points.

Let us consider a 2-irreducible r-rooted graph $\mathcal{G}$. We generate a compact 2-irreducible graph $\mathcal{G}'$, called the *compact kernel* of $\mathcal{G}$, by removing all the tree graphs attached to every tree-insertion point, and promoting all internal tree-insertion points to root. $\mathcal{G}$ is obtained by attaching a 2-irreducible tree graph to each root of $\mathcal{G}'$. If $\mathcal{G}'$ is not a tree graph (and therefore it has at least 3 roots), the decomposition is unique; otherwise, $\mathcal{G}$ itself is a tree graph. An example is shown in Fig. 3.

By summing all contribution of 2-irreducible graphs with the same compact kernel, we can write W_q as

$$W_q(x_1,..., x_q; n_1,..., n_q) = W_q^{\mathrm{tr}}(x_1,..., x_q; n_1,..., n_q) + \sum_{\substack{\text{partitions} \\ r>2}} \sum_{y_1,..., y_r,\, i_1,..., i_r} W_r^{\mathrm{co}}(y_1,..., y_r; i_1,..., i_r) \times \prod_{l=1}^{r} Y_{u_l+1}(y_l, x_{i_{l1}},..., x_{i_{lu_l}}; i_l, n_{i_{l1}},..., n_{i_{lu_l}}) \quad (27)$$

where $W_q^{\mathrm{tr}}(x_1,..., x_q; n_1,..., n_q)$ is the tree graph contribution to W_q, $W_r^{\mathrm{co}}(y_1,..., y_r; i_1,..., i_r)$ is the compact graph contribution to W_r, and

$$Y_{t+1}(y, x_1,..., x_t; i, n_1,..., n_t) = \sum_j \mu(i+j)\, W_{t+1}^{\mathrm{tr}}(y, x_1,..., x_t; j, n_1,..., n_t) + \sum_j \delta(y-x_1)\, \delta(n_1-i-j) \times W_t^{\mathrm{tr}}(x_1,..., x_t; j, n_2,..., n_t) \quad (28)$$

$$Y_2(y, x; i, n) = \sum_j \mu(i+j)\, W_2^{\mathrm{tr}}(y, x; j, n) + \delta(y-x)\, \delta(n-i)$$

(the first and second term correspond to an internal and external t-tree-insertion point respectively); notice that $W_2^{\mathrm{tr}} = W_2$. Equation (27) can be combined with Eq. (10) to give

$$G_q(x_1,..., x_q) = G_q^{\mathrm{tr}}(x_1,..., x_q) + \sum_{\substack{\text{partitions} \\ r>2}} \sum_{y_1,..., y_r, i_1,..., i_r} W_r^{\mathrm{co}}(y_1,..., y_r; i_1,..., i_r)$$

$$\times \prod_{l=1}^{r} Z_{u_l+1}(y_l, x_{i_{l1}},..., x_{i_{lu_l}}; i_l) \tag{29}$$

where

$$Z_{t+1}(y, x_1,..., x_t; i) = \sum_{\text{partitions of } \{1,..., t\}} \sum_{s_1,..., s_r} Y_{r+1}(y, x_{i_{11}},..., x_{i_{r1}}; i, s_1,..., s_r)$$

$$\times \prod_{l=1}^{r} \mu(s_l + u_l)\, \delta_{u_l}(x_{i_{l1}},..., x_{i_{lu_l}}) \tag{30}$$

$Z_{t+1}(y, x_1,..., x_t; i)$ is symmetric under permutations of $x_1,..., x_t$ and lattice symmetries, e.g., simultaneous translation of y and $x_1,..., x_t$.

These formulae can be written graphically, according to the rules presented in Table I. A sum over all dummy coordinates y and all dummy

Table I. Graphical Rules. v Is the Sum of All the Valences Incident with the Vertex. The Argument z_i of W_q^{nn} Is the Coordinate x or y Associated with the Symbol Placed at the Vertex i of the Polygon; the Argument y_i of W_q^{co} Is the Variable y of the Function Z_{q+1} Placed at the Vertex i

Symbol	Comment	Variables	Factor
●	internal vertex	y	$\mu(v)$
○	unlabelled root	x, n	$\delta(n-v)$
○ q	labelled root	$x_1,..., x_q$	$\mu(q+v)\,\delta_q(x_1,..., x_q)$
□ q	first root of Z	$x_1,..., x_q, n$	$\mu(q+v+n-1)\,\delta_q(x_1,..., x_q)$
⊗ q	root of G	$y, x_1,..., x_q$	$Z_{q+1}(y, x_1,..., x_q; v)$
◯ q		$x_1,..., x_q$	$G_q^{\mathrm{tr}}(x_1,..., x_q)$
——		i, f	$W_2(x_{i(l)} - x_{f(l)}; i, f)$
q-sided polygon			$W_q^{\mathrm{nn}}(z_1,..., z_q; v_1,..., v_q)$
q-sided polygon with a letter "c"			$W_q^{\mathrm{co}}(y_1,..., y_q; v_1,..., v_q)$

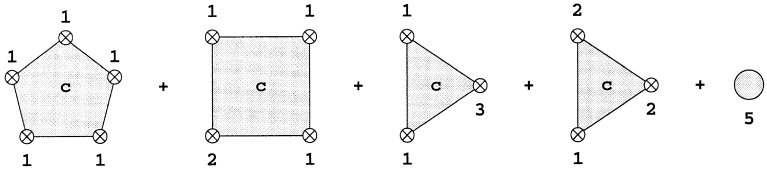

Fig. 4. Graphical representation of Eq. (29) for $q = 5$.

valences i, f and a sum over all inequivalent permutation of external coordinates x or coordinate-valence pairs x, n are understood. Notice that, despite the graphical notation, all pairs of roots of W_q^{nn} and W_q^{co} are equivalent.

Equation (29) can be expressed by writing all polygons with a letter "c" having 3 to q vertices, placing a crossed dot with a positive integer label at each vertex in all inequivalent ways, the sum of the labels being q, and adding the tree contribution. The case $q = 5$ is shown in Fig. 4.

G_q and Z_q can be computed by adding the contributions of all 2-irreducible r-rooted tree graphs with roots labelled by positive integers with sum q; for Z_q, the first root must be drawn as a square. The case Z_3 is shown in Fig. 5.

The next step is to write an expression of W_q^{co} in terms of W_r^{nn}. For $q = 3$ we have simply $W_3^{\mathrm{co}} = W_3^{\mathrm{nn}}$. The case $q = 4$ is shown in Fig. 6. For larger values of q, the number of non-nodal contributions to W_q^{co} grows rapidly, and a systematic approach is needed.

Let us define for a (connected or non-connected) graph $\mathcal{G}$ and a vertex i the graph $\mathcal{G}/i$: let $\mathcal{G}^*$ be the connected component of $\mathcal{G}$ containing i; if $\mathcal{G}^* \backslash i$ is connected, set $\mathcal{G}/i = \mathcal{G}$; otherwise, for each connected component $\mathcal{G}_k^*$ of $\mathcal{G}^* \backslash i$ generate the graph $\bar{\mathcal{G}}_k$ by adding a new vertex, internal or external like i, and joining it to all the vertices adjacent to i in $\mathcal{G}^*$, by the same number of edges; replace $\mathcal{G}^*$ with the connected components $\bar{\mathcal{G}}_k$. The edges and vertices of $\mathcal{G}/i$ are in one-to-one correspondence with the edges and vertices of i, except for the new vertices which all correspond to i (a nodal point of $\mathcal{G}$). Notice that $\mathcal{G}/i/j = \mathcal{G}/j/i$.

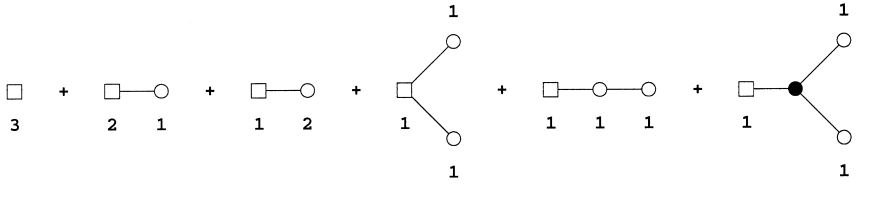

Fig. 5. Contributions to Z_3.

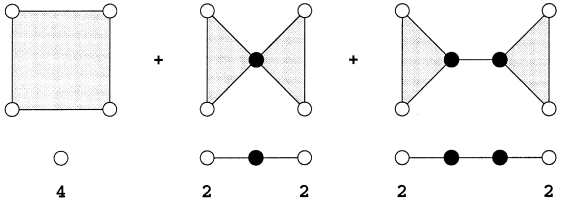

Fig. 6. Compact contributions to W_4^{co} and corresponding nodal skeletons.

Let $\mathscr{G}$ be a compact 2-irreducible r-rooted graph ($r \geqslant 3$) with t nodal points $i_1, \ldots, i_t$. Observe that all the connected components $\mathscr{G}_l$ of $\bar{\mathscr{G}} \equiv \mathscr{G}/i_1/\ldots/i_t$ are non-nodal. Generate a new graph $\mathscr{T}$, the *nodal skeleton* of $\mathscr{G}$, in the following way: for each $\mathscr{G}_l$ with $v \geqslant 3$, containing n roots corresponding to non-nodal roots of $\mathscr{G}$, write a root l of $\mathscr{T}$ with a label $n \geqslant 0$; for each node i_k of $\mathscr{G}$ write an unlabelled vertex i_k of $\mathscr{T}$, internal or external like i_k; join i_k to all the labelled roots l such that $\mathscr{G}_l$ contain a vertex corresponding to i_k, and with all unlabelled vertices which are adjacent to i_k in $\mathscr{G}$.

A nodal skeleton is a connected 1-irreducible tree graph, but it is not in general 2-irreducible (it may contain 2-valent internal vertices). A nodal skeleton enjoys the following properties: each 2-valent internal vertex is adjacent to a labelled root; labelled roots are never adjacent; unlabelled roots are at least 2-valent; m-valent roots with label n satisfy $n + m \geqslant 3$. Every nodal skeleton can be generated by adding labels to some of the roots and by splicing 2-valent internal vertices into a 2-irreducible tree graph.

Every 1-irreducible tree graph, with some roots carrying a non-negative integer label, satisfying the above properties, is the nodal skeleton of a set of q-rooted 2-irreducible compact graphs, with q equal to the sum of the labels plus the number of unlabelled roots. The contribution of this set to W_q^{co} can be computed by replacing each m-valent root of $\mathscr{T}$ carrying a label n with an $(n+m)$-sided polygon whose vertices are the m vertices adjacent to the root and n new (unlabelled) roots, and applying the rules of Table I.

An example of the construction of the nodal skeleton and its evaluation is presented in Fig. 7. The set of all nodal skeletons contributing to W_5^{co} is shown in Fig. 8; see also Fig. 6.

We could carry further the reduction of the set of graphs to be summed over, e.g., by identifying ladder graphs along the lines of Section 6. This is rather complicated for arbitrary r, and goes beyond the scope of the present work. Moreover, the zero-momentum projection described below is

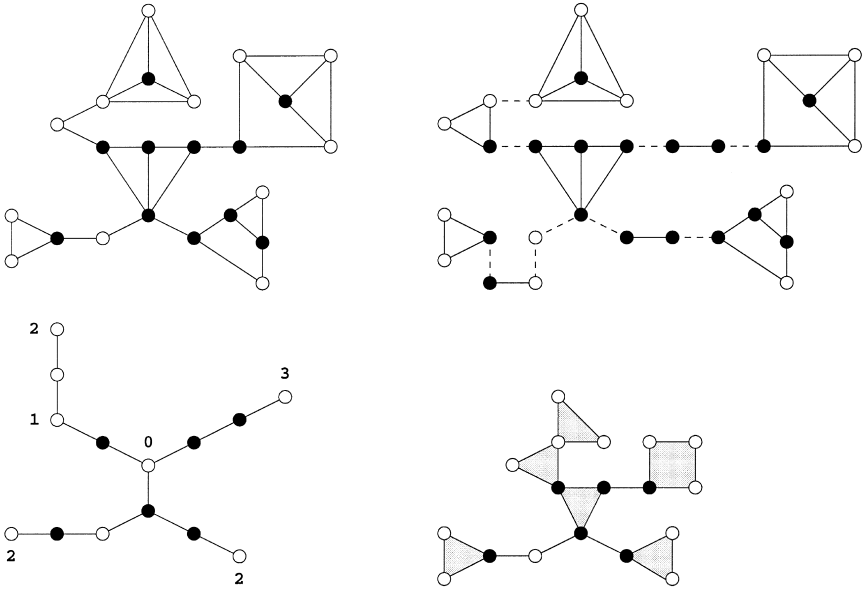

Fig. 7. A compact graph, its non-nodal components, its nodal skeleton, and the contribution to W^{co} of the set of graphs sharing the nodal skeleton.

not applicable to the ladder graph reduction. Therefore we compute W_r^{nn} by restricting the sum of Eq. (19) to $\mathscr{I}_{\text{nn}}^{(r,2)}$, the set of *non-nodal* 2-irreducible r-rooted graphs.

The above considerations can be simplified considerably if we are only interested in moments of the q-point functions, e.g.,

$$\chi_q \equiv \sum_{x_2,\ldots,x_q} G_q(x_1,\ldots,x_q)$$
$$M_2^{(q)} \equiv \sum_{x_2,\ldots,x_q} (x_1 - x_2)^2\, G_q(x_1,\ldots,x_q) \tag{31}$$

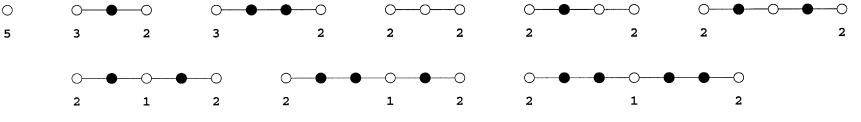

Fig. 8. Nodal skeletons contributing to W_5^{co}.

Let us introduce the moments of Z_q, W_q^{co} and W_q^{nn}:

$$\begin{aligned}\zeta_q(n) &\equiv \sum_{x_2,\ldots,x_q} Z_q(x_1,\ldots,x_q;n)\\ \omega_q^{\mathrm{co}}(n_1,\ldots,n_q) &\equiv \sum_{x_2,\ldots,x_q} W_q^{\mathrm{co}}(x_1,\ldots,x_q;n_1,\ldots,n_q)\end{aligned} \tag{32}$$

the corresponding definition for χ_q^{tr} and $\omega_q^{\mathrm{nn}}(n_1,\ldots,n_q)$, all independent second moments, etc.

Equation (29) can be projected over zero momentum to give

$$\chi_q = \chi_q^{\mathrm{tr}} + \sum_{r=3}^{q} \sum_{\substack{u_1\leqslant\cdots\leqslant u_r\\ u_1+\cdots+u_r=q}} \frac{q!}{\prod_{l=1}^{r} u_l!} \sum_{i_1,\ldots,i_r} \omega_r^{\mathrm{co}}(i_1,\ldots,i_r) \prod_{l=1}^{r} \zeta_{u_l+1}(i_l) \tag{33}$$

The computation of ζ_q, χ_q^{tr}, and ω_q^{co} is also easy; the graphical rules can be immediately projected over zero momentum, suppressing all coordinates x and y and removing all space delta functions; the only nontrivial part is the counting of the number of inequivalent permutations of roots.

$\zeta_q(n)$ can be computed by summing over all inequivalent 2-irreducible 1-ordered r-rooted tree graphs, with the roots labelled by positive integers $u_1,\ldots,u_r$ with sum q. The number of inequivalent permutations of the roots $2,\ldots,r$ is

$$\frac{(q-1)!}{S(u_1-1)!\prod_{l=2}^{r} u_l!}$$

where S is the symmetry factor of the labelled graph.

The computation of $\chi_q^{\mathrm{tr}}(n)$ is very similar, but we sum over unordered graphs, and the number of inequivalent permutations of the roots is

$$\frac{q!}{S\prod_l u_l!}$$

$\omega_q^{\mathrm{co}}(n_1,\ldots,n_q)$ can be computed by summing over all inequivalent unordered r-rooted nodal skeletons, with p roots labelled by non-negative integers $u_1,\ldots,u_p$ with $r-p+u_1+\cdots+u_p=q$, with a weight $1/S$, where S is the symmetry factor of the labelled graph, with unlabelled roots assigned an arbitrary distinct label (e.g., -1).

Finally, we can compute $\omega_q^{\mathrm{nn}}(n_1,\ldots,n_q)$ by summing over unordered non-nodal 2-irreducible q-rooted graphs, provided that we use the modified symmetry factor and we symmetrize the result under permutation of the valences.

The computation of the second moment of the above quantities proceeds along the same lines. The factor $(x_i - x_j)^2$ in Eq. (31) is dealt with in the following way: the two roots are connected by a chain of terms with a space structure of the form

$$\sum_{x_j,\, y_1,\ldots,\, y_n} (x_i - x_j)^2 \, f_1(x_i - y_1) \, f_2(y_1 - y_2) \cdots f_{n+1}(y_n - x_j)$$

(we have dropped the dependency on coordinates lying outside the branch connecting i with j). Let us write

$$(x_i - x_j)^2 = (x_i - y_1)^2 + (y_1 - y_2)^2 + \cdots + (y_n - x_j)^2 + \text{cross terms}$$

The cross terms do not contribute to the sum, and the result is

$$f_1^{(2)} f_2^{(0)} \cdots f_{n+1}^{(0)} + f_1^{(0)} f_2^{(2)} \cdots f_{n+1}^{(0)} + \cdots + f_1^{(0)} f_2^{(0)} \cdots f_{n+1}^{(2)}$$

where

$$f_i^{(0)} = \sum_y f_i(y), \qquad f_i^{(2)} = \sum_y y^2 f_i(y)$$

Therefore we can compute the second moment by taking each contribution to the zero-momentum quantity, promoting one of the zero-momentum factors along the branch connecting the roots i and j to second moment, and summing over all possible choices.

By dealing with moments, we avoid the need to store all the values of Z, W^{co}, and W^{nn}, which can rapidly exhaust all available memory. The extension to higher moments is straightforward but cumbersome.

8. PROGRAMMING DETAILS

We wrote a set of computer programs to implement the automatic evaluation of the edge-renormalized LCE on the simple cubic lattice (sc) and on the body-centered cubic lattice (bcc) ($3D$); the same programs evaluate the LCE on two different representations of the square lattice ($2D$) and on the $1D$ lattice.

The computation of q-point functions is performed for a generic potential, keeping $\mu_0(2n)$ symbolic; each term of the series is a polynomial in $\mu_0(2n)$ with rational coefficients. We also implemented the same computation for a specific potential; this requires much less memory and is somewhat faster (up to 30%), but not enough to give up the flexibility of a generic potential.

To speed up search and insertion into ordered sets of data, graph sets and polynomials in $\mu_0(2n)$ are implemented as AVL trees (height-balanced binary trees) (cf. e.g., ref. 12, Chapter 6.2.3), using the ubiqx library. Rational numbers and (potentially) large integers are handled by the GNU multiprecision (gmp) library.

Given the complexity of the procedure, it is crucial to perform a number of checks in order to flush out all algorithm and program errors. In $1D$ our series are compared with exact results for the spin-$1/2$[13] and the spin-1[14] Ising model; this is already a very stringent check, especially of the graph sets (cf. ref. 4). In $2D$, our results are compared with the series for χ and M_2 for spin-$1/2$ published in ref. 3. In $3D$, our results are compared with the lower-order series already available, for χ and M_2 for specific potentials in refs. 3, 4, and 2, and for χ, M_2, and χ_q for a generic potential in ref. 7. $q(n)$ can be computed from different combinations of n_1 and n_2 in Eq. (26); their agreement is non-trivial. Finally, for the spin-$1/2$ Ising model on any lattice, the series for G_q, rewritten in terms of $v = \tanh K$, must have integer coefficients.

8.1. Graph Generation

A program generates the table of all unordered elementary 2-rooted 2-skeletons contributing to the desired order; the algorithm follows ref. 4. Starting from the graph drawn in Fig. 9, we apply recursively the following modified Heap rules:[15]

(a) join any two distinct vertices by a new edge, *provided the two vertices are not already adjacent*;

(b) insert a new internal vertex on any edge and join it to any vertex, *excluding the edge extrema*, by a new edge;

(c) insert two new internal vertices on any two *distinct* edges, and join them by a new edge;

(d) *do not join the roots by a new edge*.

The modifications to the original Heap rules (a), (b), and (c), marked in italics, prevent the generation of 2-reducible graphs. Rule (d) prevents the generation of ladder graphs.

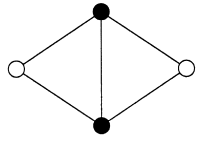

Fig. 9. The simplest elementary 2-rooted 2-skeleton.

The reduction of graphs to canonical form is performed using a generalization of the algorithm of ref. 4. Graphs are stored in a compact form similar to the one of ref. 4.

To reduce the proliferation of graphs at higher orders, it is extremely important to know the order ("strict bound") o_s at which a given graph will enter in the expansion (it is not trivially e, since we require even valence of all internal vertices, and, being interested in bipartite lattices, even length of all loops).

We must also keep in mind that some graphs don't contribute at the desired order, but graphs generated from them might contribute. We define the "Heap bound" $o_H(\mathcal{G})$ as the minimum of o_s on the set of graphs including $\mathcal{G}$ and all graphs generated from it. We also define the two bounds for $(\mathcal{G}, \eta)$, i.e., the minimal order when the vertex i is forced to be embedded in a lattice site of parity $\eta(i)$.

We apply the modified Heap rules to the graph $\mathcal{G}$ in the following way: assign a parity label $\eta(i)$ to each vertex, in all the ways compatible with the Heap bound; apply the modified Heap rules assigning all possible parity labels to the new vertices; discard immediately the generated graph-parity pairs not satisfying the Heap bound; discard vertex parity information and store the generated graphs not isomorphic to previously generated graphs. Finally, save into a file only the graphs satisfying the strict bound.

The generation of the elementary 2-rooted 2-skeletons contributing to the 25th order required 41 hours of computation on one CPU of a Compaq ES-40, and ca. 300 Mbytes of RAM. The number of inequivalent elementary 2-skeletons for each order of o_s is reported in Table II.

A similar program generates all unordered non-nodal 2-irreducible r-rooted graphs for $r \geqslant 3$. The table is initialized by applying rules (a) and (b) recursively, starting from each 2-irreducible r-rooted tree graph, until the result is non-nodal. Rules (a), (b), and (c) are then applied recursively. The number of inequivalent non-nodal 2-irreducible r-rooted graphs for each order of o_s is reported in Table II.

We remark that these graph tables can be used for the LCE of any spin model with $\phi \to -\phi$ symmetry on a bipartite lattice in any dimension.

The generation of all the required families of tree graphs is straightforward.

8.2. Computation of the q-Point Functions

A separate program reads the table of elementary 2-rooted 2-skeletons and computes all components of W_2.

The evaluation of μ and of bond, nodal, and ladder contribution to W_2 is a straightforward application of the formulae of Section 6.

Table II. Number of Inequivalent Unordered Elementary 2-Rooted 2-Skeletons ($r=2$), or Unordered Non-Nodal 2-Irreducible r-Rooted Graphs ($r>2$), for Each Order of the Strict Bound o_s. The Number of Graphs Satisfying the Heap Bound Is Typically 20% to 30% Higher. No Graph in Any of These Sets Has a Bound Lower than 4

o_s	$r=2$	$r=3$	$r=4$	$r=5$	$r=6$	$r=7$	$r=8$
4	0	1	1	0	0	0	0
5	0	0	0	0	0	0	0
6	0	0	1	2	1	0	0
7	0	1	1	1	1	0	0
8	1	3	5	4	4	2	1
9	0	2	4	6	6	5	2
10	3	7	19	26	27	22	12
11	0	9	23	47	63	48	33
12	13	46	111	175	229	228	159
13	6	54	168	378	603	661	575
14	59	263	737	1436	2224	2691	2465
15	29	367	1364	3473	6404	8694	9216
16	367	1855	5824	13190	23766	34106	38239
17	197	2898	12088	34726	72900	116210	146284
18	2589	14937	51801	133739	275031		
19	1547	25332	118225	375859	884317		
20	21682	135325	514319				
21	13933	245306	1251818				
22	199865						
23	139610						
24	2026682						
25	1516576						

The evaluation of elementary contribution dominates the computation time, and must be optimized as much as possible. Assume that all lower-order contributions to W_2 have been computed. For each unordered elementary 2-skeleton $\mathscr{G}$ with o_s not larger than the desired order, all inequivalent assignations $\mathscr{G}$, (n, l) of edge valence parity n and length parity l compatible with the desired order, with even length of all loops, and with even valence of all internal vertices, are generated. We have implemented two different algorithms for the computation of the contribution of $\mathscr{G}$, (n, l) to the two-point function.

The first algorithm is essentially the one used by Nickel and Rehr in ref. 4: all inequivalent 1-irreducible 2-rooted graphs with a 2-skeleton compatible with $\mathscr{G}$, (n, l) are generated, and their contributions are computed according to Eq. (11). In the second algorithm, the contribution is computed according to Eq. (25), and 1-irreducible graphs are not needed. The

first algorithm is more efficient for 2-skeletons with large o_s, while the second algorithm is more efficient for small o_s; for each value of o_s we select the algorithm which is (presumably) more efficient. On the sc lattice, the speed-up obtained over the use of either algorithm for all skeletons grows with the order, and is about a factor of 4 at order 23. On the bcc lattice, the first algorithm is very efficient, since the embedding number factorizes into a product of 1-dimensional embedding numbers;[4] we still use the second algorithm for the simplest 2-skeletons ($o_s \leqslant 10$), since the computation of the corresponding 1-irreducible 2-rooted graphs contributing to orders higher than 21 is extremely time- and memory-consuming.

Keeping in RAM all components of W_2 for a generic potential would be problematic. Most of these components are needed only to compute nodal and ladder contributions, and can be kept on disk; keeping in RAM just the components needed to compute elementary contribution is manageable.

The computation of the 25th-order LCE for the two-point function on the bcc lattice required ca. 400 hours of computation, and ca. 700 Mbytes of RAM.

A similar program reads the table of non-nodal 2-irreducible r-rooted graphs and the components of W_2, and computes ω_q^{nn}. The computation of χ_q is then straightforward. We computed χ_4, χ_6, and χ_8 to 21st, 19th, and 17th order respectively on the bcc lattice.

The computation of the same quantities on the sc lattice is much slower (but does not requires more RAM); so far, we obtained W_2 to 23th order, with an effort not much smaller than the 25th order on the bcc lattice. The computation of W_2 and χ_q to the same orders as on the bcc lattice is in progress, but it will require a non-trivial amount of time.

9. SELECTED RESULTS

All high-temperature series computed in the present work are available for the most general potential, in the form of polynomials in the bare vertices $\mu_0(2n)$. The general results are extremely lengthy, and are only useful for further computer processing.

We present here a selection of high-temperature series for the spin-1/2 Ising model, i.e., for

$$f(\phi) = \tfrac{1}{2}(\delta(\phi+1) + \delta(\phi-1))$$

For sake of compactness, all the series are written in terms of $v = \tanh K$. Series for other specific potentials are available upon request from the author.

Although we computed all components of $G_2(x, y)$, we report here only $\chi \equiv \chi_2$ and $M_2 \equiv M_2^{(2)}$ (cf. Eq. (31)). For the q-point functions, we only computed χ_q.

On the bcc lattice, we obtained

$$\begin{aligned}\chi = 1 &+ 8v + 56v^2 + 392v^3 + 2648v^4 + 17864v^5 + 118760v^6 + 789032v^7 \\ &+ 5201048v^8 + 34268104v^9 + 224679864v^{10} + 1472595144v^{11} \\ &+ 9619740648v^{12} + 62823141192v^{13} + 409297617672v^{14} \\ &+ 2665987056200v^{15} + 17333875251192v^{16} \\ &+ 112680746646856v^{17} + 731466943653464v^{18} + 4747546469665832v^{19} \\ &+ 30779106675700312v^{20} + 199518218638233896v^{21} \\ &+ 1292141318087690824v^{22} + 8367300424426139624v^{23} \\ &+ 54141252229349325768v^{24} \\ &+ 350288350314921653160v^{25} + \mathrm{O}(v^{26}) \end{aligned} \tag{34}$$

$$\begin{aligned}M_2 = 8v &+ 128v^2 + 1416v^3 + 13568v^4 + 119240v^5 + 992768v^6 + 7948840v^7 \\ &+ 61865216v^8 + 470875848v^9 + 3521954816v^{10} + 25965652936v^{11} \\ &+ 189180221184v^{12} + 1364489291848v^{13} + 9757802417152v^{14} \\ &+ 69262083278152v^{15} + 488463065172736v^{16} \\ &+ 3425131086090312v^{17} + 23896020585393152v^{18} \\ &+ 165958239005454632v^{19} + 1147904794262960384v^{20} \\ &+ 7910579661767454248v^{21} + 54332551216709931904v^{22} \\ &+ 372033905161237212392v^{23} + 2540342425838560175616v^{24} \\ &+ 17301457207110720278440v^{25} + \mathrm{O}(v^{26}) \end{aligned} \tag{35}$$

$$\begin{aligned}-\chi_4 = 2 &+ 64v + 1168v^2 + 16576v^3 + 201232v^4 + 2204608v^5 + 22411504v^6 \\ &+ 215447872v^7 + 1981980688v^8 + 17602809920v^9 + 151865668752v^{10} \\ &+ 1278888344256v^{11} + 10550227820400v^{12} + 85510907958720v^{13} \\ &+ 682500568307184v^{14} + 5374496030148928v^{15} \\ &+ 41821018545214608v^{16} + 321992795063663936v^{17} \\ &+ 2455641803116052752v^{18} + 18567879503614668736v^{19} \\ &+ 139310655514229882000v^{20} + 1037854026688655887552v^{21} \\ &+ \mathrm{O}(v^{22}) \end{aligned} \tag{36}$$

$$\chi_6 = 16 + 1088v + 36416v^2 + 853952v^3 + 15974528v^4 + 255491264v^5 + 3638767040v^6 + 47395195712v^7 + 574950589568v^8 + 6581949043264v^9 + 71803170318144v^{10} + 752047497945024v^{11} + 7606707093034368v^{12} + 74649010982738112v^{13} + 713458387977120192v^{14} + 6661638582474716480v^{15} + 60923519621981242752v^{16} + 546923327751320201536v^{17} + 4828463182433394315584v^{18} + 41987611565592990702272v^{19} + O(v^{20}) \quad (37)$$

$$-\chi_8 = 272 + 31744v + 1673728v^2 + 58110976v^3 + 1538207872v^4 + 33584739328v^5 + 634387677184v^6 + 10699575811072v^7 + 164723097021568v^8 + 2352360935459840v^9 + 31540880634427392v^{10} + 400802365468148736v^{11} + 4862781935250449280v^{12} + 56665753776838026240v^{13} + 637305912177206767104v^{14} + 6945658883867865975808v^{15} + 73600395257678784586368v^{16} + 760476823195422275111936v^{17} + O(v^{18}) \quad (38)$$

On the sc lattice, we obtained

$$\chi = 1 + 6v + 30v^2 + 150v^3 + 726v^4 + 3510v^5 + 16710v^6 + 79494v^7 + 375174v^8 + 1769686v^9 + 8306862v^{10} + 38975286v^{11} + 182265822v^{12} + 852063558v^{13} + 3973784886v^{14} + 18527532310v^{15} + 86228667894v^{16} + 401225368086v^{17} + 1864308847838v^{18} + 8660961643254v^{19} + 40190947325670v^{20} + 186475398518726v^{21} + 864404776466406v^{22} + 4006394107568934v^{23} + O(v^{24}) \quad (39)$$

$$
\begin{aligned}
M_2 = {} & 6v + 72v^2 + 582v^3 + 4032v^4 + 25542v^5 + 153000v^6 + 880422v^7 \\
& + 4920576v^8 + 26879670v^9 + 144230088v^{10} + 762587910v^{11} \\
& + 3983525952v^{12} + 20595680694v^{13} + 105558845736v^{14} \\
& + 536926539990v^{15} + 2713148048256v^{16} + 13630071574614v^{17} \\
& + 68121779384520v^{18} + 338895833104998v^{19} \\
& + 1678998083744448v^{20} + 8287136476787862v^{21} \\
& + 40764741656730408v^{22} + 199901334823355526v^{23} \\
& + \mathrm{O}(v^{24})
\end{aligned} \tag{40}
$$

ACKNOWLEDGMENTS

We would like to thank Michele Caselle and Ettore Vicari for many useful discussions. Special thanks are due to Paolo Rossi, who worked out exact results for the $1D$ spin-1 Ising model, and to Andrea Pelissetto, for his critical reading of the manuscript.

REFERENCES

1. A. J. Guttmann, in *Phase Transitions and Critical Phenomena*, Vol. 13, C. Domb and J. Lebowitz, eds. (Academic Press, New York, 1989), p. 1.
2. P. Butera and M. Comi, *Phys. Rev. B* **56**:8212 (1997).
3. B. G. Nickel, in *Phase Transitions, Cargèse* 1980, M. Levy, J. C. Le Guillou, and J. Zinn-Justin, eds. (Plenum, New York, 1982), p. 291.
4. B. G. Nickel and J. J. Rehr, *J. Stat. Phys.* **61**:1 (1990).
5. P. Butera and M. Comi, *Phys. Rev. B* **60**:6749 (1999).
6. P. Butera and M. Comi, *Phys. Rev. E* **55**:6391 (1997).
7. M. Campostrini, A. Pelissetto, P. Rossi, and E. Vicari, *Phys. Rev. E* **60**:3526 (1999).
8. M. Campostrini, A. Pelissetto, P. Rossi, and E. Vicari, *Phys. Rev. B* **61**:5905 (2000).
9. M. Wortis, in *Phase Transitions and Critical Phenomena*, Vol. 3, C. Domb and M. S. Green, eds. (Academic Press, London, 1974), p. 113.
10. M. Lüscher and P. Weisz, *Nucl. Phys. B* **300**:325 (1988).
11. J. W. Essam and M. E. Fisher, *Rev. Mod. Phys.* **42**:272 (1970).
12. D. E. Knuth, *The Art of Computer Programming*, Vol. 3 (Addison-Wesley publishing co., Reading, 1973).
13. M. Campostrini, A. Pelissetto, P. Rossi, and E. Vicari, *Nucl. Phys. B* **459**:207 (1996).
14. P. Rossi, private communication.
15. B. R. Heap, *J. Math. Phys.* **7**:1582 (1966).