Computer Physics Communications 66 (1991) 71–88
North-Holland

Hilbert space and structure constants of descendant fields in two-dimensional conformal theories

Michael Lässig and Giuseppe Mussardo [1]

Department of Physics, University of California, Santa Barbara, CA 93106, USA

Received 12 July 1990; in revised form 5 November 1990

We have developed an algorithm to compute the Hilbert-space basis and the operator algebra of descendant fields for $(1+1)$-dimensional conformal field theories. Implemented as a *Mathematica* computer program, this algorithm is used to obtain nonperturbatively the spectrum of the transfer matrix theories, seen as deformations of a massless conformal theory.

PROGRAM SUMMARY

Title of program: STRIP

Catalogue number: ABZT

Program obtainable from: CPC Program Library, Queen's University of Belfast, N. Ireland (see application form in this issue)

Computer: DECstation 3100: *Installation*: Physics Department, University of California, Santa Barbara, CA 93106, USA

Operating system: ULTRIX

Programming language used: Mathematica [1]

High speed storage required: strongly dependent on the dimension of the truncated Hilbert space; about 11 megabyte for a generic perturbation of the tricritical Ising model.

No. of lines in combined program and test deck: 470

Key words: conformal field theory, two-dimensional systems away from criticality, spectrum of the transfer matrix

Nature of physical problem
To compute the spectrum of the transfer matrix for massive $(1+1)$-dimensional quantum field theories.

Method of solution
The massive regime is seen as a deformation of the model at its massless critical point, where it can be solved exactly with the methods of conformal field theory. This program encodes an efficient algorithm for the computation of the necessary data at the conformal point, namely (a) a convenient basis of the Hilbert space, truncated to a finite number of elements, and (b) the "interaction" matrix elements, i.e. the structure constants of the perturbing operator between these states.

Typical running time
Strongly dependent on the dimension of the truncated Hilbert space; about 1 s CPU to compute each interaction matrix element, and about 1 min CPU to diagonalize a 200 × 200 Hamiltonian matrix.

Reference

[1] S. Wolfram, Mathematica (Addison–Wesley, New York, 1988).

[1] I.N.F.N. Fellow, Sez. di Trieste, Italy.

LONG WRITE-UP

1. Introduction

The principle of conformal invariance has been a very important tool for the analysis of two-dimensional statistical systems at a critical point [1,2]. More recently, it also has been used successfully to obtain information about massive systems away from criticality. Such a system can be regarded as the conformally invariant critical theory perturbed by a relevant scaling field $\phi_{d,d}$ of scaling dimension $2d$; via the logarithmic mapping $\omega = (R/2\pi)\ln z$, it defines a Euclidean quantum field theory on a cylinder of circumference R. The Hamiltonian (the logarithm of the transfer matrix) has the form

$$H_\lambda = H_0 + \lambda V = H_0 + \lambda \int_0^R \phi_{d,d}(\omega, \bar{\omega}) d(\mathrm{Im}\,\omega)/2\pi, \tag{1.1}$$

with an obvious generalization to the case of several perturbing fields. By scaling, the spectrum of H_λ depends essentially only on the dimensionless variable λR^{2-2d}, therefore we set in the following $\lambda = 1$ and study the spectrum as a function of R.

The "unperturbed" part of the Hamiltonian can be expressed in terms of the Virasoro generators L_0, $\bar{L}_0$, and the central charge c of the conformal theory [3],

$$H_0 = (2\pi/R)(L_0 + \bar{L}_0 - c/12). \tag{1.2}$$

The eigenvalues of $L_0 + \bar{L}_0$ are the scaling dimensions in the plane, and the eigenstates of H_0 ("conformal states"), which are related to the scaling operators, are assumed to form a basis of the Hilbert space. The dimensionless matrix elements of V between them,

$$v_{ji}(d, c) \equiv (R/2\pi)^{2d-1}\langle j | V | i \rangle = (R/2\pi)^{2d}\langle j | \phi_{d,d}(0, 0) | i \rangle \tag{1.3}$$

are structure constants of the conformal algebra. Since the Hamiltonian (1.1) commutes with the momentum operator on the cylinder,

$$P = (2\pi/R)(L_0 - \bar{L}_0), \tag{1.4}$$

the dynamics factorizes into Hilbert space sectors of definite momentum $P = 2\pi K/R$ ($K = 0, 1, 2, \ldots$).

Yurov and Zamolodchikov suggested to truncate the Hilbert space to a finite number of conformal states, thus reducing the problem of finding the off-critical spectrum to the numerical diagonalization of a finite matrix whose elements can be computed exactly from the conformal theory. They applied this method to the massive Yang–Lee model [4]; Lässig, Mussardo and Cardy [5] studied the scaling region of the tricritical Ising model. A moderate number of conformal states turned out to be sufficient to reproduce the spectrum of the infinite-dimensional Hilbert space accurately in the massive regime. For a detailed discussion of the results, the reader is referred to these two papers.

We emphasize that the method is very general and can be used to analyze numerically the scaling region of any minimal conformal field theory (i.e. a conformal theory with a finite number of primary fields). It is essentially *nonperturbative* and can be used in cases where there is no small parameter for an ϵ-expansion. Therefore it clearly deserves to be explored further. However, already for the tricritical Ising model it becomes quite complex to compute the necessary data of the conformal theory, i.e. the basis of states and the interaction matrix elements (1.3). Here we provide an efficient algorithm for that.

2. Conformal Hilbert space

The states of a conformal field theory organize into conformal families. Many physically interesting examples are described by a "minimal" conformal theory, where there is only a finite number F of families. These are products of holomorphic and antiholomorphic Verma modules, irreducible representations of the Virasoro algebra

$$[L_n, L_m] = (n-m)L_{n+m} + \frac{c}{12}n(n^2-1)\delta_{n+m,0}. \tag{2.1}$$

Each Verma module $\mathscr{V}_d$ contains one primary state $|d\rangle$ of conformal dimension d, which satisfies

$$L_0|d\rangle = d|d\rangle, \qquad L_k|d\rangle = 0 \quad (k>0), \tag{2.2}$$

and the descendant states

$$L_{-n_s}\cdots L_{-n_2}\cdots L_{-n_1}|d\rangle \quad (n_s \le \cdots \le n_2 \le n_1) \tag{2.3}$$

of dimension $d+n(n=\Sigma n_i = 1, 2, \ldots)$. The number $\nu(n, d)$ of linearly independent descendant states at level n is given by the character of the Verma module $\mathscr{V}_d$,

$$\chi_d(q) = q^{d-c/24}\sum_{n=0}^{\infty}\nu(n, d)q^n. \tag{2.4}$$

The Verma module can be decomposed into two orthogonal subspaces (this decomposition is convenient for the computation of the matrix elements below).

(a) The quasiprimary subspace consists of the states $|\delta\rangle$ which satisfy the equation $L_1|\delta\rangle = 0$ (and hence cannot be written as the derivative of states at the previous level). The number of linearly independent quasiprimary states at level n is $\nu(n, d) - \nu(n-1, d)$. A basis in this subspace may be obtained as follows. At each level $n = 2, 3, \ldots$ there is one "primitive" quasiprimary state $P_n(d)|d\rangle$, where $P_n(d)$ is a polynomial in the lowering operators L_{-k} given recursively by

$$P_1(d) = 1, \qquad P_n(d) = p_{n,n}(d)L_{-n} + \sum_{m=1}^{n-1} p_{n,m}(d)L_{-1}^{n-\hat{m}}P_m(d) \quad (n \ge 2), \tag{2.5}$$

with $\hat{m} = 0$ if $m = 1$, $\hat{m} = m$ otherwise,

$$p_{n,m}(d) = \begin{cases} 1, & \text{if } m = 1, \\ -\dfrac{2(2d+1)_{n-1}}{n+1}, & \text{if } m = 2, 3, \ldots \text{ and } n = m, \\ -\dfrac{n!(2d+m)_{n-m}}{m!(n-m)!(2d+2m)_{n-m}}, & \text{if } m = 2, 3, \ldots \text{ and } n > m, \end{cases} \tag{2.6}$$

and $(a)_k = \Gamma(a+k)/\Gamma(a)$. The quasiprimary subspace of a nondegenerate Verma module $\mathscr{V}_d$ is spanned by the composite states

$$Q_{n_s,\ldots,n_1}(d)|d\rangle = P_{n_s}(d+n_{s-1}+\cdots+n_1)\cdots P_{n_2}(d+n_1)P_{n_1}(d)|d\rangle, \tag{2.7}$$

with $n_s \le n_{s-1} \le \cdots \le n_1$. In a degenerate Verma module, the null states have to be projected out.

(b) The orthogonal subspace of $\mathscr{V}_d$ consists of pure derivative states. Any such state at level k is of the form $L_{-1}^r|\delta\rangle$, where $|\delta\rangle$ is a quasiprimary state at level $k-r$. A holomorphic descendant state

$$|f, k, \{n_i\}\rangle = L_{-1}^{k-\Sigma n_i} Q_{\{n_i\}} |d_f\rangle \tag{2.8}$$

is thus characterized by three quantum numbers labelling its Verma module, its level and its quasiprimary subfamily.

The full Hilbert space is obtained by taking the tensor product of the left and right Verma modules for each conformal family, in a way that can be read off from the decomposition of the partition function in terms of the characters (2.4),

$$Z(q, \bar{q}) = \sum N_{d,\bar{d}} \chi_d(q) \chi_{\bar{d}}(\bar{q}), \tag{2.9}$$

the integers $N_{d,\bar{d}}$ being determined by modular invariance [3]. Each conformal state

$$|f_1, k_1, \{n_i\}_1; f_r, k_r, \{n_i\}_r\rangle = |f_1, k_1, \{n_i\}_1\rangle \otimes |f_r, k_r, \{n_i\}_r\rangle \tag{2.10}$$

is characterized by six quantum numbers; it has momentum $(2\pi/R)K = (2\pi/R)(k_1 - k_r)$. In the case of a diagonal modular form, the zero-momentum Hilbert space truncated at level N has dimension

$$\sum_{f=1}^{F} \sum_{n=0}^{N} \nu^2(n, d_f), \tag{2.11}$$

where the index f labels the conformal families appearing in the model.

3. Structure constants

The matrix elements $v_{ji}(d, c)$ of eq. (1.3) are analytic functions of the conformal dimensions d_i, d_j and d, and the central charge c. The finitely many structure constants between primary states,

$$C_{\phi_j, \phi, \phi_i} \equiv (R/2\pi)^{2d} \langle \phi_j | \phi(0, 0) | \phi_i \rangle \tag{3.1}$$

are fundamental dynamical characteristics of the conformal theory. For the minimal models, they have been calculated by Dotsenko and Fateev [6]. Once the primary structure constants are known, the structure constants between descendant fields are determined by conformal invariance [1].

3.1. Matrix elements between derivative states

The matrix elements between derivative states are given in terms of the quasiprimary matrix elements by the formula [4]

$$\langle \delta_2 | L_1^{r_2} \phi_d(0) L_{-1}^{r_1} | \delta_1 \rangle = f(\delta_2, r_2, d, \delta_1, r_1) \langle \delta_2 | \phi_d(0) | \delta_1 \rangle \tag{3.2}$$

with the "f-factor"

$$f(\delta_2, r_2, d, \delta_1, r_1) = r_2! r_1! \sum_{l=0}^{\min(r_1, r_2)} \frac{(\delta_2 + \delta_1 - d)_l (d + \delta_1 - \delta_2)_{r_1 - l} (d + \delta_2 - \delta_1)_{r_2 - l}}{l!(r_1 - l)!(r_2 - l)!}, \tag{3.3}$$

and the analogous formula for the antiholomorphic part.

3.2. Matrix elements between quasiprimary states

The quasiprimary matrix elements may be computed recursively as follows. Consider first the matrix elements between two primitive quasiprimaries,

$$\begin{aligned}&\langle d_2 \,|\, P^{\dagger}_{n_2}(d_2)\phi_{d,d}(0,0)P_{n_1}(d_1)\,|\,d_1\rangle\\ &\quad= p_{n_2,n_2}(d_2)\Big(\langle d_2\,|\big[L_{n_2},\,\phi_{d,d}(0,0)\big]P_{n_1}(d_1)\,|\,d_1\rangle + \langle d_2\,|\,\phi_{d,d}(0,0)L_{n_2}P_{n_1}(d_1)\,|\,d_1\rangle\Big)\\ &\qquad+\sum_{m_2=1}^{n_2-1} p_{n_2,m_2}(d_2)\langle d_2\,|\,P^{\dagger}_{m_2}(d_2)L_1^{n_2-\hat{m}_2}\phi_{d,d}(0,0)P_{n_1}(d_1)\,|\,d_1\rangle.\end{aligned} \tag{3.4}$$

Hence the "q-factor" of proportionality,

$$q(d_2,\,n_2,\,d,\,d_1,\,n_1,\,c)\equiv\frac{\langle d_2\,|\,P^{\dagger}_{n_2}(d_2)\phi_{d,d}(0,0)P_{n_1}(d_1)\,|\,d_1\rangle}{\langle d_2\,|\,\phi_{d,d}(0,0)\,|\,d_1\rangle}, \tag{3.5}$$

is given by the recursion

$$q(d_2,\,1,\,d,\,d_1,\,1,\,c)=1, \tag{3.6}$$

$$\begin{aligned}&q(d_2,\,n_2,\,d,\,d_1,\,n_1,\,c)\\ &\quad=p_{n_2,n_2}(d_2)\Bigg((-d_1-n_1+d_2+n_2d)q(d_2,\,1,\,d,\,d_1,\,n_1,\,c)\\ &\qquad+\frac{(-1)^{n_2}}{(n_2-2)!}\sum_{\hat{m}_1=0}^{n_1-n_2}a_{n_1,m_1}(d_1,\,c)\times(n_1-\hat{m}_1-n_2+1)_{n_2-1}(2d_1+n_1+\hat{m}_1-n_2)_{n_2-1}\\ &\qquad\times f(d_2,\,0,\,d,\,d_1+\hat{m}_1,\,n_1-\hat{m}_1-n_2)q(d_2,\,1,\,d,\,d_1,\,m_1,\,c)\Bigg)\\ &\qquad+\sum_{m_2=1}^{n_2-1}p_{n_2,m_2}(d_2)f(d_2+\hat{m}_2,\,n_2-\hat{m}_2,\,d,\,d_1+\hat{n}_1,\,0)q(d_2,\,m_2,\,d,\,d_1,\,n_1,\,c).\end{aligned} \tag{3.7}$$

To derive eq. (3.7), we used eq. (3.2) and the commutation relations

$$\big[L_k,\,\phi_{d,d}(\omega,\,\bar{\omega})\big]=\mathrm{e}^{2\pi k\omega/R}\big(kd+(R/2\pi)\partial_{\omega}\big)\phi_{d,d}(\omega,\,\bar{\omega}), \tag{3.8}$$

$$\big[L_2,\,P_n(\delta)\big]\,|\,\delta\rangle=\sum_{m=1}^{\max(1,n-2)}a_{n,m}(\delta,\,c)L_{-1}^{n-2-\hat{m}}P_m(\delta)\,|\,\delta\rangle \tag{3.9}$$

(and their complex conjugates), together with the equation

$$L_kP_n(\delta)\,|\,\delta\rangle=\frac{(-1)^k}{(k-2)!}L_1^{k-2}L_2P_n(\delta)\,|\,\delta\rangle. \tag{3.10}$$

The coefficients $a_{n,m}(\delta,\,c)$ in eq. (3.9) are generated by the recursion

$$a_{2,1}(\delta,\,c)=-\frac{16}{3}\left(\delta^2+\frac{c-5}{8}\delta+\frac{c}{16}\right), \tag{3.11}$$

$$a_{n,m}(\delta, c) = p_{n,m}(\delta)(n - \hat{m})(n - \hat{m} - 1)(3(\delta + \hat{m}) + n - \hat{m} - 2)$$
$$+ \sum_{k=\hat{m}+2}^{m-1} p_{n,k}(\delta) a_{k,m}(\delta, c) + \frac{\cancel{(n+} 2) p_{n,n}(\delta)}{p_{n-2,n-2}(\delta)}$$
$$\times \begin{cases} 1, & \text{if} \quad m = n - 2, \\ -p_{n-2,m}(\delta), & \text{if} \quad m = 1, 2, \ldots, n - 3. \end{cases} \tag{3.12}$$

Because of the symmetry $f_{ji}(d, c) = v_{ij}(d, c)$ of the matrix elements, the recursion relation (3.7) needs to be evaluated only for $n_2 \geq n_1$, so that at most one term appears in the sum over m_1.

The next step is to determinate the q-factors,

$$q(d_2, \{n_{21}, n_{22}\}, d, d_1, n_1, c) \equiv \frac{\langle d_2 | P^\dagger_{n_{21}}(d_2) P^\dagger_{n_{22}}(d_2 + n_{21}) \phi_{d,d}(0, 0) P_{n_1}(d_1) | d_1 \rangle}{\langle d_2 | \phi_{d,d}(0, 0) | d_1 \rangle} \tag{3.13}$$

which is done in exactly the same way as in eq. (3.7). Finally, the q-factors,

$$q(d_2, \{n_{21}, n_{22}\}, d, d_1, \{n_{11}, n_{12}\}, c)$$
$$\equiv \frac{\underline{\langle d_2 |} P^\dagger_{n_{21}}(d_2) P^\dagger_{n_{22}}(d_2 + n_{21}) \phi_{d,d}(0, 0) P_{n_{12}}(d_1 + n_{11}) P_{n_{11}}(d_1) | d_1 \rangle}{\langle d_2 | \phi_{d,d}(0, 0) | d_1 \rangle}, \tag{3.14}$$

are computed. In this case, the terms $L_{n_{22}} P_{n_{12}}(d_1 + n_{11}) P_{n_{11}}(d_1) | d_1 \rangle$ are more complicated, and they are evaluated in each case separately. If we truncate the Hilbert space at level 5 of the Verma modules, there are no quasiprimaries with more than two primitive factors.

The Hilbert space basis (2.10) is not orthonormal; the inner product between two states is

$$g_{ji}(c) \equiv \langle j | i \rangle = v_{ji}(0, c). \tag{3.15}$$

The interaction matrix elements appearing in the logarithm of the transfer matrix (1.1) are the structure constants with one contravariant index; they are obtained from eq. (1.3) by raising the index j,

$$v_i^{\,f}(d, c) = g^{fj}(c) v_{ji}(d, c). \tag{3.16}$$

4. Structure and usage of the program STRIP

For a (1 + 1)-dimensional perturbed conformal field theory, the program STRIP computes the Hamiltonian (1.1) and diagonalizes it. The structure of the program is sketched in fig. 1; the Mathematica code appears in the listing at the end of this paper. The computation involves three steps (parts 1–3, 4–6, and 7 of STRIP, respectively). First, the program calculates the symbolic expressions of the descendant factors in the conformal structure constants, using the algorithms described in detail in sections 2 and 3. Specifically, it encodes the formulae for

1. the primitive quasiprimary polynomials $P_n(d)$ given by eq. (2.6),
2. the f-factors (3.3),
3. the q-factors (3.5), (3.13) and (3.14).

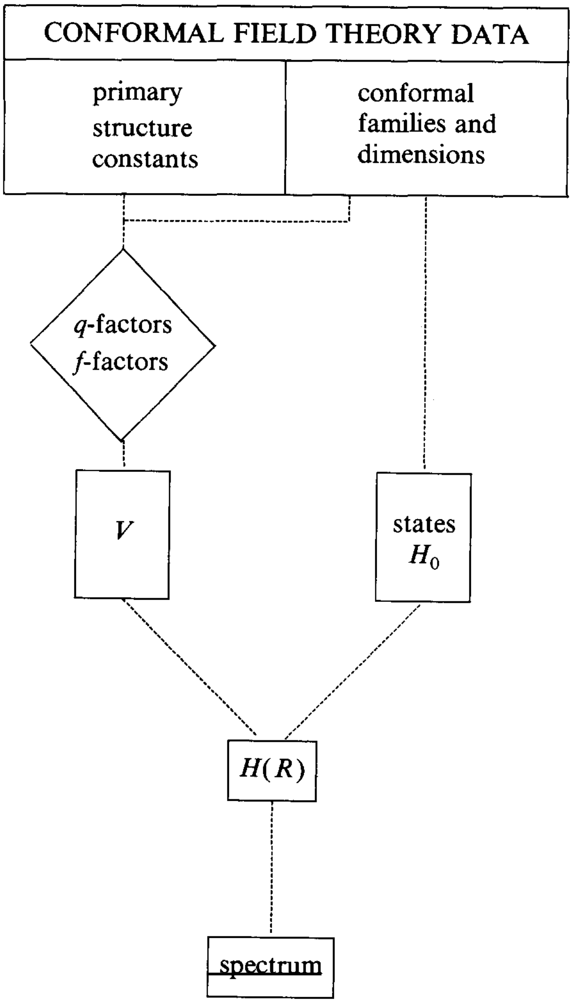
CONFORMAL FIELD THEORY DATA
primary structure constants
conformal families and dimensions
q-factors
f-factors
V
states
H_0
$H(R)$
spectrum

Fig. 1. Flow diagram of STRIP.

As explained above, these formulae follow from conformal invariance alone and do not use input describing a particular conformal theory. They appear in many contexts of conformal field theory, and the user just interested in them may run sections 1–3 of STRIP independently of the rest, e.g. as subroutine of his own program. Second, STRIP computes the Hamiltonian matrices H_0 and V, which govern the dynamics off criticality. It obtains

4. the conformal basis (2.10) of the Hilbert space sectors of a given momentum K, truncated to level 5 in the Verma modules (and optionally the further truncated basis of all such states with scaling dimension smaller than a given cutoff),
5. the dimensionless covariant matrix elements (1.3) and (3.15),
6. the (dimensionless) matrices H_0 and V.

The last part of the program computes

7. the Hamiltonian (1.1) and its spectrum as a function of R (and optionally of the Hilbert space cutoff), for $\lambda = 1$.

To this part, the user will probably want to add commands that manipulate the output spectra further for his particular purpose. For instance, Mathematica allows plotting of the spectral lines in a very convenient way (see fig. 2). This is described in detail in ref. [7].

4.1. Input data

The input for STRIP that describes the dynamics of the unperturbed conformal field theory is conveniently collected in a file INPUT (see the example discussed in subsect. 4.4, and the corresponding INPUT file shown at the end of this paper. It consists of

1. the central charge c and the conformal dimensions of the primary fields (assembled to the vector `dp`),
2. the modular form `partition`, which lists the pair (f_l, f_r) for each primary field appearing in the theory,
3. the structure constants between primary fields $C_{jki} = C_{\Phi_j,\Phi_k,\Phi_i}$ given by eq. (3.1) (these are the elements of a symmetric tensor of rank 3 referred to as `cptensor`), and
4. the basis `verma[f,k]` for $f = 1, 2, \dots, F$ and $k = 1, 2, \dots, 5$, consisting of all states (2.8) in the Verma module $\mathscr{V}_{d_f}$ at level k.

4.2. Running STRIP, output

The program has been tested and run on a DECstation 3100 to obtain massive spectra for several perturbed conformal field theories. For Hilbert space dimensions > 100, economic use of memory space in the Mathematica session computing the matrices H_0 and V is of concern. In these cases, one would run STRIP in practice as follows. First, the sequence of commands

```
<< STRIP; << INPUT    (loads the files STRIP and INPUT)
K=value1              (specifies the Hilbert space sector K = 0, 1, 2, ...)
fpert=value2          (specifies the perturbing primary field, f_pert = 1, 2,..., F)      (4.1)
ho[K,c]               (computes the matrix H_0 in the sector K)
v[K,fpert,c]          (computes the matrix V in the sector K)
```

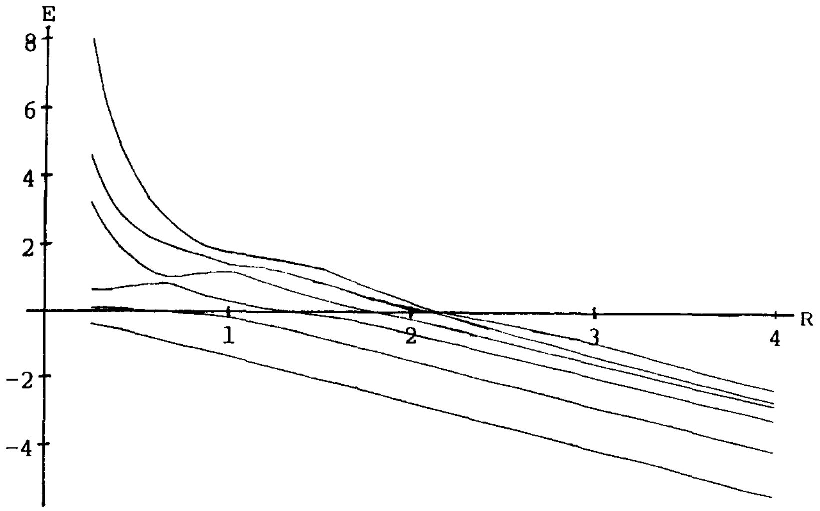

Fig. 2. Lowest six eigenvalues of the finite-size transfer matrix of the tricritical Ising model in an external magnetic field, as a function of the size R of the system.

can conveniently be executed in batch mode, and the output be stored in a file DATA, together with dp and c. The quantities K, fpert, dp, c, H_0, V form a complete set of input for part 7 of STRIP, which can be separated from the rest and be called STRIP2. Then one runs batch-load programs of the form

```
<< STRIP2; << DATA;        (loads the files STRIP and DATA)
Plot[spec[r], {r,1,3}]     (plots the ground state energy for 1 ≤ R ≤ 3)      (4.2)
Display[``line.ps'',%]     (saves the plot as a PostScript file)
```

4.3. Range of applicability, truncation effects

The spectrum in the truncated Hilbert space is very accurate in the conformal regime $R \ll \xi$, where the interaction $R^{2-2d}V$ becomes small. In the massive regime $R \geq \xi$, however, the interaction and hence the mixing of the conformal states is large. Hence it is important to make sure that the Hilbert space truncated to a finite number of dimensions reproduces accurately the low-lying spectrum of the infinite-dimensional Hilbert space. For all massive perturbations we studied, we found for $R \geq \xi$ a domain several correlation lengths long where the spacing of the low-lying energy levels becomes approximately constant and the one-particle levels below threshold may be obtained with an accuracy of about 1 to 2 percent. Indeed, in this regime the leading finite-size correction to the one-particle levels vanishes like $\exp(-ar/\xi)$, where a is a constant of order 1. In many cases, even the first few one- and two-particle states above threshold are given with similar accuracy.

To estimate the truncation effects, the Hilbert space can be further reduced to the set of all states with scaling dimension smaller than a given cutoff. For the massive perturbations, truncating the Hilbert space at level 3 of the Verma modules gives already an accuracy similar to level 5. This very fast convergence justifies the use of the truncation method in these cases. If the theory has a massless sector, the situation is less clear.

The truncation can be carried out in STRIP as follows. The commands

```
cutoff=value3;entries[K,cutoff]                                              (4.3)
```

give a list of the numbers of the states in the Hilbert space sector K that have scaling dimensions smaller than `cutoff`, which one stores in the file DATA. Then, by replacing `spec[r]` in (4.2) above with `spec[cutoff,r]`, one obtains the spectrum in the further truncated space.

4.4. Example: the perturbed tricritical Ising model

The conformal field theory of the two-dimensional tricritical Ising model has the following data relevant for our analysis. The central charge is $c = 7/10$, there are six conformal families of dimensions

$$d_f = 0, \tfrac{3}{80}, \tfrac{1}{10}, \tfrac{7}{16}, \tfrac{6}{10}, \tfrac{3}{2}, \tag{4.4}$$

the modular form is diagonal, and the nonvanishing primary structure constants (3.1) are

$$\begin{aligned} &C_{1/10,1/10,6/10} = c_1 = \tfrac{2}{3}\sqrt{\Gamma(\tfrac{4}{5})\Gamma^3(\tfrac{2}{5})/\Gamma(\tfrac{1}{5})\Gamma^3(\tfrac{3}{5})}\,, && C_{6/10,6/10,6/10} = c_2 = c_1,\\ &C_{1/10,6/10,3/2} = c_3 = \tfrac{3}{7}, && C_{3/80,7/16,1/10} = c_4 = \tfrac{1}{2},\\ &C_{3/80,3/80,1/10} = c_5 = \tfrac{3}{2}c_1, && C_{3/80,7/16,6/10} = c_6 = \tfrac{3}{4},\\ &C_{3/80,3/80,6/10} = c_7 = \tfrac{1}{4}c_1, && C_{7/16,7/16,3/2} = c_8 = \tfrac{7}{8}c_1,\\ &C_{3/80,3/80,3/2} = c_9 = \tfrac{1}{56}. \end{aligned} \tag{4.5}$$

We use the quasiprimary basis

$$\begin{aligned}
&|0\rangle,\quad P_2(0)\,|0\rangle,\quad P_2(2)P_2(0)\,|0\rangle,\\
&|\tfrac{3}{80}\rangle,\quad P_2(\tfrac{3}{80})\,|\tfrac{3}{80}\rangle,\quad P_3(\tfrac{3}{80})\,|\tfrac{3}{80}\rangle,\quad P_2(\tfrac{3}{80}+2)P_2(\tfrac{3}{80})\,|\tfrac{3}{80}\rangle,\quad P_5(\tfrac{3}{80})\,|\tfrac{3}{80}\rangle,\quad P_2(\tfrac{3}{80}+3)P_3(\tfrac{3}{80})\,|\tfrac{3}{80}\rangle,\\
&|\tfrac{1}{10}\rangle,\quad P_3(\tfrac{1}{10})\,|\tfrac{1}{10}\rangle,\quad P_4(\tfrac{1}{10})\,|\tfrac{1}{10}\rangle,\quad P_5(\tfrac{1}{10})\,|\tfrac{1}{10}\rangle,\\
&|\tfrac{7}{16}\rangle,\quad P_3(\tfrac{7}{16})\,|\tfrac{7}{16}\rangle,\quad P_4(\tfrac{7}{16})\,|\tfrac{7}{16}\rangle,\quad P_5(\tfrac{7}{16})\,|\tfrac{7}{16}\rangle,\\
&|\tfrac{6}{10}\rangle,\quad P_2(\tfrac{6}{10})\,|\tfrac{6}{10}\rangle,\quad P_4(\tfrac{6}{10})\,|\tfrac{6}{10}\rangle,\quad P_2(\tfrac{6}{10}+2)P_2(\tfrac{6}{10})\,|\tfrac{6}{10}\rangle,\quad P_5(\tfrac{6}{10})\,|\tfrac{6}{10}\rangle,\\
&|\tfrac{3}{2}\rangle,\quad P_2(\tfrac{3}{2})\,|\tfrac{3}{2}\rangle,\quad P_2(\tfrac{3}{2}+2)P_2(\tfrac{3}{2})\,|\tfrac{3}{2}\rangle,\quad P_5(\tfrac{3}{2})\,|\tfrac{3}{2}\rangle,
\end{aligned}\tag{4.6}$$

which is truncated at level 5 in the Verma modules. Up to this level, the number of zero-momentum states given by eq. (2.11) is 228.

As a typical result for a massive perturbation, we present in fig. 2 the spectrum of the model in an external magnetic field as a function of R [5]. For larger values of R, the spacing of the lines becomes constant and one can read off the ratios of the mass gaps of the transfer matrix.

Acknowledgements

It is a pleasure to acknowledge the collaboration with John L. Cardy on the tricritical Ising model which was the basis for this work. This work was supported in part by NSF Grant PHY 86-14185 and by INFN.

References

[1] A.A. Belavin, A.M. Polyakov and A.B. Zamolodchikov, Nucl. Phys. B241 (1984) 333.

[2] C. Itzykson, H. Saleur and J.B. Zuber, eds., Conformal Invariance and Applications to Statistical Mechanics (World Scientific, Singapore), and references therein.

[3] J.L. Cardy, Nucl. Phys. B 270 [FS16] (1986) 186; Nucl. Phys. B 275 [FS17] (1986) 200; in: Champes, Cordes, et Phenomenes Critiques, Proc. 1988 Les Houches Summer School, E. Brezin and J. Zinn-Justin, eds. (North-Holland, Amsterdam, 1989).

[4] V.P. Yurov and Al.B. Zamolodchikov, Truncated conformal space approach to scaling Yang–Lee model, preprint ITEP 89-161.

[5] M. Lässig, G. Mussardo and J.L. Cardy, Nucl. Phys. B. 348 (1991) 591.

[6] Vl.S. Dotsenko and V.A. Fateev, Nucl. Phys. B 240 (1984) 312; Phys. Lett. 154 B (1985) 291.

[7] S. Wolfram, Mathematica (Addison–Wesley, New York, 1988).

LISTING

```
                         (* File STRIP *)


(* *************** 1. PRIMITIVE QUASIPRIMARY POLYNOMIALS ******************** *)


prod[d_,s_Integer] := Product[d+jj,{jj,0,s-1}]


rank[k_Integer] := If[k==1,0,k]
rank[{n1_Integer,n2_Integer}] := n1 + n2


(* KAC DETERMINANT AT LEVEL 2 *)

kac[d_,2,c_] := d^2 + (c-5)/8d + c/16


(* COEFFICIENT OF P_n(d) *)

p[d_,n_Integer,m_Integer]:= p[d,n,m] =
   Which[ m==1, 1,
          m>1 && n==m, -2/(n+1) prod[2d+1,n-1],
          m>1 && n>m,  -Binomial[n,m] prod[2d+m,n-m]/prod[2d+2m,n-m],
          True,        0  ]


(* *********************** 2. DERIVATIVE FACTORS **************************** *)


f[d2_,s2_Integer,d_,d1_,s1_Integer] :=
     Block[ {l},  s2! s1! Sum[ prod[d1+d2-d,l] / (l!) *
                               prod[d+d2-d1,s2-l] / ((s2-l)!)
                               prod[d+d1-d2,s1-l] / ((s1-l)!),
                               {l,0,Min[s1,s2]}              ] ]


(* **************************** 3. q-FACTORS ******************************** *)


(* COEFFICIENTS a_nm(d,c) APPEARING IN L_2  P_n(d) |d>  *)

a[d_,c_,n_Integer,m_Integer] :=  a[d,c,n,m] =
        p[d,n,m] (n - rank[m])(n - rank[m] - 1)(3d + 2 rank[m] + n - 2) +
        Block[ {k} , Sum[ p[d,n,k] a[d,c,k,m],
                          {k,rank[m]+2,n-1}    ] ] +
        (n+2) p[d,n,n] / p[d,n-2,n-2] If[ n-m==2, 1, -p[d,n-2,m] ]   /;n>2

a[d_,c_,2,1] := a[d,c,2,1] = -16/3 kac[d,2,c]


(* q-FACTORS FOR THE CASES:  PRIMITIVE-PRIMITIVE, COMPOSITE-PRIMITIVE
   AND COMPOSITE-COMPOSITE *)

q[d2_,n2_,d_,d1_,n1_,c_] := q[d2,n2,d,d1,n1,c] =
 Which[ IntegerQ[n2] &&
        IntegerQ[n1] ,
          If[ n2 >= n1 ,
               Which[ n2 > 1,
                       p[d2,n2,n2] ( (-d1-rank[n1]+d2+n2*d) q[d2,1,d,d1,n1,c] +
                                      If[ n1==n2, (-1)^n1 / ((n1-2)!) *
                                                 f[d1,n1-2,0,d1,n1-2] *
                                                 a[d1,c,n1,1],
                                                 0
                                        ]
                                    ) +
                       Block[ {m2} , Sum[ p[d2,n2,m2] *
                                          f[d2+rank[m2],n2-rank[m2],d,
                                            d1+rank[n1],0] *
                                          q[d2,m2,d,d1,n1,c],
                                          {m2,1,n2-1}
                                        ]
                             ],
                      n2 == 1,
                       1,
                      True,
                       0
                     ],
               q[d1,n1,d,d2,n2,c]
             ],
         VectorQ[n2] &&
          IntegerQ[n1] ,
            Which[ n2[[2]] > 1,
                    p[d2+n2[[1]],n2[[2]],n2[[2]]]
                     ( (-d1-rank[n1]+d2+n2[[1]]+n2[[2]]*d) *
                          q[d2,n2[[1]],d,d1,n1,c] +
                       (-1) n2[[2]] / ((n2[[2]]-2)!) *
                          Block[{m1},Sum[a[d1,c,n1,m1] *
                                         prod[ n1-rank[m1]-n2[[2]]+1,n2[[2]]-2]
                                         prod[2 d1+rank[m1]+n1-n2[[2]],n2[[2]]-2]
                                         f[d2+n2[[1]],0,d,
                                           d1+rank[m1],n1-rank[m1]-n2[[2]]]
                                         q[d2,n2[[1]],d,d1,m1,c],
                                         { m1, 1, If[n2[[2]]==n1,1,n1-n2[[2]]] }
                                        ]
                               ]
                     ) +
                    Block[ {m22}, Sum[ p[d2+n2[[1]],n2[[2]],m22]
                                       f[d2+n2[[1]]+rank[m22],n2[[2]]-rank[m22],
                                         d,d1+rank[n1],0]
                                       q[d2,{n2[[1]],m22},d,d1,n1,c],
                                       {m22,1,n2[[2]]-1}
                                     ]
                         ],
                  n2[[2]] == 1,
                   q[d2,n2[[1]],d,d1,n1,c],
                  True,
                   0
                 ],
          IntegerQ[n2] &&
          VectorQ[n1],
               q[d1,n1,d,d2,n2,c],
          VectorQ[n2] &&
          VectorQ[n1],
            Which[ n2 == {2,2} && n1 == {2,2},
                    p[d2+2,2,2] ( (-d1-4 +d2+2+ 2 d) q[d2,2,d,d1,{2,2},c] +
                                  (a[d1+2,c,2,1] +
                                     a[d1,c,2,1] p[d1+2,2,2]/p[d1,2,2])
                                     q[d2,2,d,d1,2,c] +
                                  a[d1,c,2,1] (1-p[d1-2,2,2]/p[d1,2,2]) *
                                     f[d2+2,0,d,d1,2] *
                                     q[d2,2,d,d1,1,c]
                                ) +
                    f[d2+2,2,d,d1+4,0] q[d2,2,d,d1,{2,2},c],
                   n2=={2,3} && n1=={2,2},
                    p[d2+2,3,3] ( (-d1-4+d2+2+3d) q[d2,2,d,d1,{2,2},c] +
                                  a[d1,c,2,1] 2(2d1+1) *
                                      ( p[d1+2,2,2]/p[d1,2,2] - 1)*
                                       f[d2+2,0,d,d1,1]*
                                      q[d2,2,d,d1,1,c]
                                ) +
                   p[d2+2,3,2] f[d2+4,1,d,d1+4,0] q[d2,{2,2},d,d1,{2,2},c] +
                   f[d2+2,3,d,d1+4,0] q[d2,2,d,d1,{2,2},c],
                   n2=={2,3} && n1 == {2,3},
                    p[d2+2,3,3] ( (-d1-5+d2+2+3d) q[d2,2,d,d1,{2,3},c] -
                                  2(d1+2) (a[d1+2,c,3,1] +
                                           a[d1,c,2,1]*
                                              (p[d1+2,3,2]*p[d1+2,2,2]/p[d1,2,2]-
                                               p[d1+2,3,3]*p[d1,3,2]/p[d1,3,3]
                                              )
                                          ) q[d2,2,d,d1,2,c] -
                                  3(2d1+2) a[d1,c,2,1]
                                              (1 -
                                               p[d1+2,3,3]/p[d1,3,3] +
                                               p[d1+2,3,2]
                                                  (1 -p[d1+2,2,2]/p[d1,2,2])
                                              ) f[d2+2,0,d,d1,2] *
                                                q[d2,2,d,d1,1,c]
                               ) +
                   p[d2+2,3,2] f[d2+4,1,d,d1+5,0] q[d2,{2,2},d,d1,{2,3},c] +
                   f[d2+2,3,d,d1+5,0] q[d2,2,d,d1,{2,3},c],
                  n2[[2]] < n1[[2]],
                   q[d1,n1,d,d2,n2,c]
                 ]
       ]


(* ************* 4. HILBERT SPACE SECTORS OF A GIVEN MOMENTUM K ************* *)


(* STATES OF A GIVEN FAMILY *)

space[fl_,fr_,kl_,kr_] :=
        Block[ {i,j}, Flatten[ Table[ {verma[fl,kl][[i]],verma[fr,kr][[j]]},
                                      {i,1,Length[verma[fl,kl]]},
                                      {j,1,Length[verma[fr,kr]]}
                                    ],
                             1]
             ]
```

```
union[ set_,fl_,fr_,K_,cc___] :=
   Block[ {kl},
       Flatten[
             Table[set[ fl, fr, kl, kl +dp[[fl]]-dp[[fr]]-K,cc],
                   {kl, Max[0,dp[[fr]]-dp[[fl]]+K],Min[5,5+dp[[fr]]-dp[[fl]]+K]}
                  ],
              1]
        ]


(* UNION OF ALL FAMILIES OCCURING IN THE OPERATOR ALGEBRA *)

union[set_,K_,cc___] :=
 Block[ {ff},
    Flatten[Table[ union[ set,partition[[ff,1]],partition[[ff,2]],K,cc],
                   {ff,1,Length[partition]} ],
           1]
      ]

space[K_] :=  space[K] = union[space,K]


                  (* ENERGY TRUNCATED SUBSPACES *)

(* SCALING DIMENSIONS OF A CONFORMAL STATE *)

dim[state_] := -c/12 + dp[[state[[1,1]]]] + state[[1,2]] +
                       dp[[state[[2,1]]]] + state[[2,2]]


test[state_] := dim[state] < cutoff

subspace[K_,cutoff_] := Select[ space[K], test ]


(* NUMBERS OF THE STATES IN SPACE[K] THAT ARE MEMBERS OF
   subspace[K,cutoff] *)

entries[K_,cutoff_] :=  entries[K,cutoff] =
      Flatten[ Table[ Position[ space[K], subspace[K,cutoff][[ii]] ],
                      {ii, 1, Length[subspace[K,cutoff]]}               ] ]


(* ************************ 5. MATRIX ELEMENTS ****************************** *)


(* HOLOMORPHIC FACTOR, I.E. THE PRODUCT OF q-FACTOR AND DERIVATIVE FACTOR *)

cdesc[{f2_,k2_,n2_},fpert_,{f1_,k1_,n1_},c_] :=
   q[dp[[f2]],n2,dp[[fpert]], dp[[f1]],n1,c]*
   f[dp[[f2]]+rank[n2],k2-rank[n2],dp[[fpert]],dp[[f1]]+rank[n1],k1-rank[n1]]


(* FULL DESCENDANT STRUCTURE CONSTANT, I.E. THE PRODUCT OF PRIMARY STRUCTURE
   CONSTANT, HOLOMORPHIC FACTOR, AND ANTIHOLOMORPHIC FACTOR *)

str[state2_,fpert_,state1_,c_] :=
   cptensor[[fpert,state2[[1,1]],state1[[1,1]]]]*
   cdesc[state2[[1]],fpert,state1[[1]],c] cdesc[state2[[2]],fpert,state1[[2]],c]


(* INTERACTION MATRIX ELEMENTS *)

v[i_,fpert_,j_,fks__,c_] := str[ space[fks][[i]], fpert, space[fks][[j]], c]


(* COVARIANT INTERACTION MATRIX *)

vrow[i_,K_,fpert_,c_] := Table[ N[ v[i,fpert,j,K,c] ], {j,1,Length[space[K]]} ]

vform[K_,fpert_,c_] :=  Table[ vrow[i,K,fpert,c], {i,1,Length[space[K]]} ]


(* HILBERT SPACE METRIC AND ITS INVERSE, COMPUTED IN BLOCK FORM *)

g[fl_,fr_,kl_,kr_,c_] :=
      Block[ {i,j}, Table[ v[i,1,j,fl,fr,kl,kr,c],
                           {i,1,Length[space[fl,fr,kl,kr]]},
                           {j,1,Length[space[fl,fr,kl,kr]]}  } ]


round[matr_] :=
   Block[ {i,j}, Table[ If[ Abs[matr[[i,j]]] < 10^(-15), 0, matr[[i,j]] ],
                        {i,1,Length[matr]},
                        {j,1,Length[matr]}                              ] ]
```

```
ginv[fl_,fr_,kl_,kr_,c_] :=  {round[ Inverse[ N[ g[fl,fr,kl,kr,c] ] ] ]}

ginvdiag[K_,c_] := union[ginv,K,c]

sum[matr1_,matr2_] :=
       Block[ {i1,i2,j1,j2},
                Join[ Table[ Join[ matr1[[i1]] ,
                                   Table[ 0, {j2,1,Length[matr2]} ]
                                  ],
                             {i1,1,Length[matr1]}
                            ] ,
                      Table[ Join[ Table[ 0, {j1,1,Length[matr1]} ] ,
                                   matr2[[i2]]
                                  ],
                             {i2,1,Length[matr2]}
                            ]
                    ]
            ]

block[diag_] := Block[ {t}, matr = {};
                            Do[ matr = sum[ matr , diag[[t]] ],
                                {t,1,Length[diag]}
                               ];
                            matr
                      ]

ginv[K_,c_] := ginv[K,c] = block[ ginvdiag[K,c] ]


(* ********** 6. HAMILTON OPERATOR, CONFORMAL PART AND PERTURBATION ********* *)


ho[K_,c_] := ho[K,c] =
      N[ DiagonalMatrix[ Table[ dim[ space[K][[i]]],
                               {i,1,Length[space[K]]} ] ] ]


v[K_,fpert_,c_] := v[K,fpert,c] = ginv[K,c].vform[K,fpert,c]


(* ************* 7. HAMILTONIAN AND SPECTRUM AS A FUNCTION OF R ************* *)


(* HAMILTONIAN ACTING ON THE HILBERT SPACE WITH SPECIFIED K AND
   (OPTIONALLY) SPECIFIED CUTOFF, AS A FUNCTION OF THE SYSTEM SIZE R *)

ham[R_] := N[  ho[K,c] + R^(2 - 2 dp[[fpert]]) v[K,fpert,c] ]

ham[cutoff_,R_] :=
  N[ ho[K,c][[entries[K,cutoff],entries[K,cutoff]]] +
  R^(2 - 2 dp[[fpert]]) v[K,fpert,c][[entries[K,cutoff],entries[K,cutoff]]] ]


(* SPECTRUM *)

spec[coff___,R_] := 1/R Sort[ Eigenvalues[ ham[coff,R] ] ]
```

TEST RUN INPUT

```
          (* File INPUT : CONFORMAL DATA OF TRICRITICAL ISING MODEL *)

(* CENTRAL CHARGE, PRIMARY DIMENSIONS AND NONZERO STRUCTURE CONSTANTS *)

c=7/10

dp={0,3/80,1/10,7/16,6/10,3/2}

c1=2/3*Sqrt[Gamma[4/5]/Gamma[1/5]*(Gamma[2/5]/Gamma[3/5])^3]
c2=c1
c3=3/7
c4=1/2
c5=3/2*c1
c6=3/4
c7=1/4*c1
c8=7/8
c9=1/(56)


(* DIAGONAL MODULAR FORM *)

partition = {{1,1},{2,2},{3,3},{4,4},{5,5},{6,6}}


(* TENSOR OF PRIMARY STRUCTURE CONSTANTS *)

cptensor := { IdentityMatrix[6],
                            {{0,1,0,0,0,0},
                            {1,0,c5,0,c7,c9},
                            {0,c5,0,c4,0,0},
                            {0,0,c4,0,c6,0},
                            {0,c7,0,c6,0,0},
                            {0,c9,0,0,0,0}},
                   {{0,0,1,0,0,0},
                   {0,c5,0,c4,0,0},
                   {1,0,0,0,c1,0},
                   {0,c4,0,0,0,0},
                   {0,0,c1,0,0,c3},
                   {0,0,0,0,c3,0}},
                            {{0,0,0,1,0,0},
                            {0,0,c4,0,c6,0},
                            {0,c4,0,0,0,0},
                            {1,0,0,0,0,c8},
                            {0,c6,0,0,0,0},
                            {0,0,0,c8,0,0}},
                   {{0,0,0,0,1,0},
                   {0,c7,0,c6,0,0},
                   {0,0,c1,0,0,c3},
                   {0,c6,0,0,0,0},
                   {1,0,0,0,c2,0},
                   {0,0,c3,0,0,0}},
                            {{0,0,0,0,0,1},
                            {0,c9,0,0,0,0},
                            {0,0,0,0,c3,0},
                            {0,0,0,c8,0,0},
                            {0,0,c3,0,0,0},
                            {1,0,0,0,0,0}} }
```

```
(* BASIS OF THE VERMA MODULES *)

verma[1,0]={{1,0,1}}
verma[1,1]={}
verma[1,2]={{1,2,2}}
verma[1,3]={{1,3,2}}
verma[1,4]={{1,4,2},{1,4,{2,2}}}
verma[1,5]={{1,5,2},{1,5,{2,2}}}

verma[2,0]={{2,0,1}}
verma[2,1]={{2,1,1}}
verma[2,2]={{2,2,1},{2,2,2}}
verma[2,3]={{2,3,1},{2,3,2},{2,3,3}}
verma[2,4]={{2,4,1},{2,4,2},{2,4,3},{2,4,{2,2}}}
verma[2,5]={{2,5,1},{2,5,2},{2,5,3},{2,5,{2,2}},{2,5,5},{2,5,{2,3}}}

verma[3,0]={{3,0,1}}
verma[3,1]={{3,1,1}}
verma[3,2]={{3,2,1}}
verma[3,3]={{3,3,1},{3,3,3}}
verma[3,4]={{3,4,1},{3,4,3},{3,4,4}}
verma[3,5]={{3,5,1},{3,5,3},{3,5,4},{3,5,5}}

verma[4,0]={{4,0,1}}
verma[4,1]={{4,1,1}}
verma[4,2]={{4,2,1}}
verma[4,3]={{4,3,1},{4,3,3}}
verma[4,4]={{4,4,1},{4,4,3},{4,4,4}}
verma[4,5]={{4,5,1},{4,5,3},{4,5,4},{4,5,5}}

verma[5,0]={{5,0,1}}
verma[5,1]={{5,1,1}}
verma[5,2]={{5,2,1},{5,2,2}}
verma[5,3]={{5,3,1},{5,3,2}}
verma[5,4]={{5,4,1},{5,4,2},{5,4,4},{5,4,{2,2}}}
verma[5,5]={{5,5,1},{5,5,2},{5,5,4},{5,5,{2,2}},{5,5,5}}

verma[6,0]={{6,0,1}}
verma[6,1]={{6,1,1}}
verma[6,2]={{6,2,1},{6,2,2}}
verma[6,3]={{6,3,1},{6,3,2}}
verma[6,4]={{6,4,1},{6,4,2},{6,4,{2,2}}}
verma[6,5]={{6,5,1},{6,5,2},{6,5,{2,2}},{6,5,5}}
```

TEST RUN

The following (interactive) test run illustrates the features of STRIP described in sections 4.2 and 4.3 for the case of the tricritical Ising model. It calculates

[2] the simplest q-factor $q(d_1, 1, d, d_1, 2, c)$ as given by eq. (3.5),
[4] the basis of the Hilbert space sector $K=0$ consisting of the 228 states (2.10) up to level 5,
[5,6] the numbers of all states in space [0] with scaling dimension smaller than 5.2,
[7] the 228×228 matrix H_0 in the $K=0$ sector,
[8] the 228×228 matrix V in the $K=0$ sector for the magnetic perturbation $f_{\text{pert}} = 2$,
[9,10] the spectrum of the perturbed system for $R = 1.5$ (compare with fig. 2).

```
Mathematica (DECstation 3100) 1.2 (August 22, 1989) [With pre-loaded data]
by S. Wolfram, D. Grayson, R. Maeder, H. Cejtin,
   S. Omohundro, D. Ballman and J. Keiper
with I. Rivin and D. Withoff
Copyright 1988,1989 Wolfram Research Inc.
Temporary license: expiration date is 31 Jul 1990.
 -- Terminal graphics initialized --

In[1] := << STRIP;

In[2] := q[d2, 1, d, d1, 2, c]

Out[2] =  (d + d1 - d2)(1 + d + d1 - d2) - (2(1 + 2d1)(2d + d1 - d2))/3

In[3] := << INPUT;

In[4] := space[0]

Out[4] = {{{1, 0, 1}, {1, 0, 1}}, {{1, 2, 2}, {1, 2, 2}}, {{1, 3, 2}, {1, 3, 2}},
   {{1, 4, 2}, {1, 4, 2}}, {{1, 4, 2}, {1, 4, {2, 2}}},
   {{1, 4, {2, 2}}, {1, 4, 2}}, {{1, 4, {2, 2}}, {1, 4, {2, 2}}},
   {{1, 5, 2}, {1, 5, 2}}, {{1, 5, 2}, {1, 5, {2, 2}}},
   {{1, 5, {2, 2}}, {1, 5, 2}}, {{1, 5, {2, 2}}, {1, 5, {2, 2}}},
   {{2, 0, 1}, {2, 0, 1}},

                     ..........

                               {{6, 5, 5}, {6, 5, 5}}}

In[5] := cutoff = 5.2;

In[6] := entries[0,5.2]

Out[6] = {1, 2, 12, 13, 14, 15, 16, 17, 79, 80, 81, 111, 112, 113, 143, 144,
   145, 146, 147, 148, 194, 195}

In[7] := ho[0,c]

Out[7] = {{-0.0583333333333334, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,

                      .........

     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 12.94166666666667}}

In[8] := v[0,2,c]

Out[8] =  {{0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 1., 0.005625,
     0.006500390624999999, 0., 0., 0., 0.02798824438476562, 0., 0., 0., 0.,
     0., 0., 0., 0., 0.2646463433106994, 0., 0., 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 4.394617934638734, 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
     0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.},
    {0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.02582908163265305,
     0.095712890625, 0.0818943420410156, -0.1635267857142857,

                  .........
```

```
      0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
      0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.},
     {0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
      3.17357335090899*10^-9, 0., 0., 0., 0., 6.214054969414237*10^-8,
      1.512673400911616*10^-8, 0., 1.512673400911616*10^-8,
      3.682266779241586*10^-9, 0., 0., 0., 0., 0., 7.289474857558488*10^-7,
      1.774460441356883*10^-7, 0.000001605113830918397, 0.,
      1.774460441356883*10^-7, 4.319529073724082*10^-8,
      3.907292435320182*10^-7, 0., 0.000001605113830918397,
      3.907292435320182*10^-7, 0.000003534397827758554, 0., 0., 0., 0., 0.,
      0., 0., 0.000004286666808422988, 0.000001043493643295432,
      0.0000094390725219445, -0.000002236399320416519,
      -0.00001728789109515525, 0., 0.000001043493643295432,
      2.540153065916868*10^-7, 0.000002297732157745475,
      -(5.444016479515904*10^-7), -0.000004208352379599757, 0.,
      0.0000094390725219445, 0.000002297732157745475, 0.00002078446822586264,
      -0.000004924463765637256, -0.00003806725950288663, 0.,
      -0.000002236399320416519, -(5.444016479515904*10^-7),
      -0.000004924463765637256, 0.000001166753131018235,
      0.0000090192752608321 9, 0., -0.00001728789109515525,
      -0.000004208352379599757, -0.00003806725950288663,
      0.0000090192752608321 9, 0.00006972111243418 49, 0., 0., 0., 0., 0., 0.,
      0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
      0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
      0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
      0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
      0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
      0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
      0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
      0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]}

In[9] := K = 0; fpert = 2;

In[10] := spec[1.5]

Out[10] = {-2.069345018488711, -0.813716650077005, -0.1050869152547375,
   0.3810511687097676, 0.851621269803161, 1.287730519273702,
   1.390730047852666, 1.600517355488277, 1.808635013860721,
   2.042915941392645, 2.201705414179244, 2.345325446274008,
   2.432025578722706, 2.544489684412801, 2.60754651176458, 2.847491797439055,

                    ..........
```