AN IMPROVED METHOD FOR LATTICE GLUEBALL CALCULATIONS

M. TEPER[1]
Department of Theoretical Physics, University of Oxford, Oxford OX1 3NP, UK

Received 18 October 1986

We present a simple iterative method for obtaining glueball wave functionals of the appropriate physical size as one decreases the lattice spacing. We test the method in SU(2) for $2.1 \leqslant \beta \leqslant 2.5$ and find a gain in computer time over previous methods, that can be several orders of magnitude. The method appears to make possible reliable, large lattice calculations of excited glueball masses and significantly extends the range of bare couplings where reliable glueball calculations can be performed.

The basic steps of a glueball mass calculation[‡1] – within the context of euclidean lattice Monte Carlo simulations – are as follows. First choose a functional Φ of the (usually spacelike) lattice link variables, $U_\mu(n)$, with the quantum numbers of the desired glueball state: since the glueball is a color singlet, Φ will be composed of traces of closed loops of links. Next calculate the correlation function $\langle \Phi(t)\Phi(0)\rangle$. Noting that under the Minkowski-to-euclidean rotation the time translation operator becomes $\exp(-Ht)$, and inserting complete sets of energy eigenstates we easily derive

$$C(t) \equiv \frac{\langle \Phi(t)\Phi(0)\rangle}{\langle \Phi(0)\Phi(0)\rangle} = \frac{\langle \Phi \exp(-Ht)\Phi\rangle}{\langle \Phi\Phi\rangle}$$

$$= \sum_n \frac{|\langle n|\Phi\rangle|^2}{\langle \Phi|\Phi\rangle} \exp(-E_n t)$$

$$\xrightarrow[t\to\infty]{} \frac{|\langle n=0|\Phi\rangle|^2}{\sum |\langle n|\Phi\rangle|^2} \exp(-E_0 t). \quad (1)$$

Here $|n=0\rangle$ is the lowest energy state communicating with Φ: this will typically be the lowest mass glueball with the $(J^{PC};\boldsymbol{p})$ quantum numbers of Φ and with E_0 its energy. [If necessary use $\Phi - \langle\Phi\rangle$ to remove the vacuum from the sum in eq. (1).] A minimal criterion for a reliable calculation of E_0 is that at least three neighbouring (in time) calculated values of $C(t)$ should fall on a simple exponential and that the errors there should be small enough for this to be a statistically convincing fit.

The basic difficulty with such calculations is that for large enough times the rapidly decreasing signal, $C(t)$, will be drowned in noise. Consequently there exist very few reliable glueball mass calculations. Indeed the only reliable large-volume calculations [2,3] (on the interesting side of the strong coupling "crossover") are for the 0^{++}, using a cold-wall source variant of the method described above. (The only reliable 2^{++} calculations [4] are limited, by the methods used, to very small physical volumes, where one makes contact with the small-volume expansions of ref. [5].) The range of lattice spacings where such calculations have proved possible [2], corresponding to a range of bare couplings

$$5.5 \leqslant \beta \equiv 6/g^2 \leqslant 5.9 \quad (2)$$

is small indeed, as we can see from the fact that in units of the *measured* correlation length, ξ ($= M_{0^+}^{-1}$), the lattice spacing varies only between

$$2/3 \leqslant 1/\xi \leqslant 1.1. \quad (3)$$

What is needed are glueball measurements in the range $\beta = 6.0$–6.6 where a/ξ should vary between 2 and, perhaps, 4 (extrapolating the results of ref. [2].). It is reasonable to expect that there exist lattice actions for which the physics in this region will be close to continuum physics (say within 5%) and this will be the appropriate point at which to determine the necessary improvements to the standard Wilson

[1] Research Fellow at All Souls College, University of Oxford.

[‡1] For a recent review with references see ref. [1].

action. Unfortunately present glueball methods become rapidly inefficient with increasing β, so much so that reliable calculations are essentially impossible for $\beta > 6$. We shall first remind the reader of the specific difficulty involved – over and above the usual, but manageable, difficulties of critical slowing down and larger lattice volumes – and we shall then describe a class of simple and fast methods designed to overcome the problem. We will test the method in the SU(2) gauge theory with the standard Wilson action: this is sufficient because the problem being confronted is independent of the gauge group or lattice regularisation being used and so also is the solution proposed herein, up to minor technical details (which will be specified). We shall find that even the simplest version of the method works very well. Moreover the improvement is sufficiently great as to make possible, for the first time, reliable large lattice calculations of excited glueball masses.

The basic problem is that for any simple choice of wave functional, $\Phi[U]$, the normalised projection onto the lowest mass glueball decreases rapidly with increasing β. This means that the coefficient of the asymptotic exponential in eq. (1) decreases rapidly with β. Worse, it means that the value of $C(t)$ at the point $t = t_{\min}$ where the asymptotic exponential begins to dominate – and this is the first point at which our measurements begin to be *useful* – decreases *very* rapidly with increasing β. To illustrate how this happens in practice we have reanalysed the data of ref. [2] to extract the best projection at each β from a selection of small loops, and also the value of the correlation function, $C(t_{\min})$, where the asymptotic exponential decay sets in. As we see in fig. 1, in the range $5.5 \leqslant \beta \leqslant 6.0$, where $a(\beta)$ changes by a factor of between 2 and 3, the projection decreases by at least a factor of ~ 40 and $C(t_{\min})$ by at least a factor of ~ 200. Recalling that the CPU time will need to vary as $[C(t_{\min})]^{-2}$ to compensate, we recognise this as being a disaster of the "brick-wall" type: no amount of gigaflops will enable us to obtain large-lattice glueball masses much beyond $\beta = 5.9$ with these methods.

The reason for this difficulty is that the mismatch between $\Phi[U]$ and the physical glueball wave functional increases rapidly with β. Consider, for simplicity, Φ to be composed of elementary plaquettes, $U_{\mu\nu}(n)$. As $a(\beta) \to 0$ the number of glueball states (even of a fixed J^{PC}) grows rapidly. Think, for the purposes of illustration, of a simple radial wave function: the maximum number of nodes obviously grows as $R/a(\beta)$ where R is the typical physical size. However the plaquette operator $U_{\mu\nu}(n)$ does not overlap more than one node at a time because it is so small and so it does not suffer cancellations and will project roughly equally on all these states. This argument exposes the rather obvious fact that ultraviolet loops will project in a roughly equal fashion on a rapidly increasing number of states as $a(\beta) \to 0$ and hence the normalised projection onto any single state – in particular the desired lowest mass state – will decrease very rapidly. As we see in fig. 1. In fact it is clear that if Φ has *any* spatial direction in which it is ultraviolet it will suffer some loss of projection. The first step towards a solution is therefore also rather obvious: construct operators Φ that possess the extended structure of the physical glueball. One can do this by summing some dense subset of all the

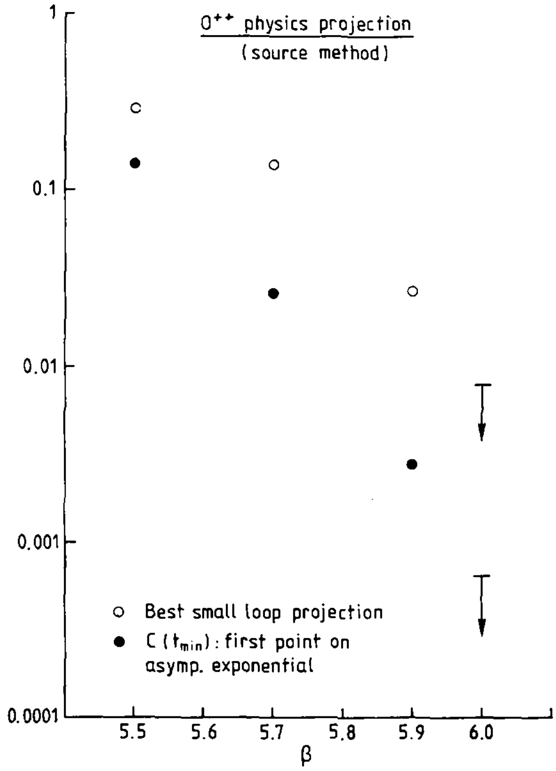

Fig. 1. Projection of the best (at each β) small loop operator onto the lowest mass scalar glueball state (○); value of the normalised correlation function at the time, $t_{\min}$, at which it begins to be dominated by the lowest mass scalar glueball (●).

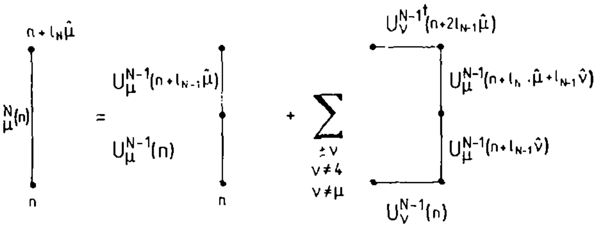
n + l_N μ̂
μ^N(n)
n
=
U_μ^{N-1}(n + l_{N-1} μ̂)
U_μ^{N-1}(n)
n
+ Σ
±ν
ν≠4
ν≠μ
U_ν^{N-1†}(n+2l_{N-1} μ̂)
U_μ^{N-1}(n + l_h · μ̂ + l_{N-1} ν̂)
U_μ^{N-1}(n + l_{N-1} ν̂)
U_ν^{N-1}(n)

Fig. 2. Construction of composite paths at the Nth level from composite paths at the $(N-1)$th level.

closed loops in a volume $\sim(1\ \mathrm{fm})^3$. To do this loop-by-loop would require essentially infinite computer time. We shall introduce here a very fast iterative method – one of several methods we have considered – which, despite its simplicity, turns out to work very well.

Denote the group elements on the lattice links – usually denoted by $U_\mu(n)$ – by $U^0_\mu(n)$. We can view them as elementary paths of length $l_0=1$. Usually we would form Φ by multiplying the $U^0_\mu(n)$ around a few closed loops. Instead we shall form new composite paths $U^1_\mu(n)$ of length l_1 joining the site n to the site $(n+l_1\hat{\mu})$ for each site n and each *spatial* direction μ. To be precise we shall in this paper use the sum of the direct path and the spatial "staples":

$$U^1_\mu(n)=U^0_\mu(n)\,U^0_\mu(n+l_0\hat{\mu}) + \sum_{\substack{\pm\nu\neq\mu\\ \nu\neq 4}} U^0_\nu(n)\,U^0_\mu(n+l_0\hat{\nu}) \times U^0_\mu(n+l_0\hat{\nu}+l_0\hat{\mu})\,U^{0+}_\nu(n+2l_0\hat{\mu}). \qquad (4)$$

This procedure can be iterated to produce $U^N_\mu(n)$ out of $U^{N-1}_\mu(n)$ as in fig. 2. We shall then form Φ from simple closed loops of these composite paths. Note that each composite path is of length

$$l_N=2^{N-1} \qquad (5)$$

in lattice units and if we denote by $n_\mathrm{p}(N)$ the number of elementary paths contributing to a composite path at the Nth level, then from eq. (4)

$$n_p(N)=[n_p(N-1)]^2+4[n_p(N-1)]^4. \qquad (6)$$

Therefore the number of elementary paths contributing to even the simplest path composed of these composite variables will quickly become enormous. For example, consider the analogue of the elementary plaquette in terms of these composite variables:

Table 1

Level of blocking	Blocked plaquette		
	average width	maximum width	number contrib. elem. paths.
0	a	a	1
1	$2a$	$4a$	625
2	$4a$	$10a$	4.1×10^{13}
3	$8a$	$22a$	7.0×10^{56}

$$U^N_{ij}(n) = U^N_i(n)\,U^N_j(n+l_N\hat{i})\,U^{N+}_i(n+l_N\hat{j})\,U^{N+}_j(n). \qquad (7)$$

In table 1 we show how the size of such a square varies with N and also the number of elementary paths (out of the original link matrices) contributing to these composite loops. It is clear that by increasing N with β we can match the size of U^N_{ij} to the growing (in lattice units) size of the glueball wave function, while maintaining the *dense* character of the wave functional.

We shall refer to this iterative procedure as "blocking": the composite paths at the Nth level, $U^N_\mu(n)$, are obtained by blocking together the composite paths at the preceding, $(N-1)$th, level. The same terminology is of course used within the field of Monte Carlo renormalisation group calculations [2]. In that case, as in the present case, the procedure of blocking is designed to expose the large-distance physics of the lattice fields: this similarity plus our desire for economy motivates our use of the same term. Note however that our use of the procedure is in almost every other way very different. We use only *spatial* paths in the blocking procedure. We do not block the lattice itself: this is important since if we were to do so we would lose much of what we gain. Our blocked variables are not group elements; they are not even proportional to group elements except in the case of SU(2). Unlike MCRG our purpose is not to thin out the degrees of freedom of the lattice gauge field: we wish to preserve the transfer matrix as it is. Our blocking procedure is merely designed to produce wave functionals complex enough in terms of the lat-

[2] For a review see ref. [6].

tice spacing so as to have a chance of having a strong overlap with the lowest mass and most featureless (presumably) glueball state.

We construct zero-momentum 0^+ operators

$$\Phi^{0^+}[N;t] = \mathrm{tr}\sum_{\boldsymbol{n}}[U^N_{12}(\boldsymbol{n},t)+U^N_{23}(\boldsymbol{n},t)+U^N_{31}(\boldsymbol{n},t)], \tag{8}$$

$$\Phi^{2^+}_1[N;t] = \mathrm{tr}\sum_{\boldsymbol{n}}[U^N_{12}(\boldsymbol{n},t)-U^N_{13}(\boldsymbol{n},t)], \tag{9}$$

etc., from the "plaquettes" defined in eq. (7), and measure the correlation functions $C_N(t)$ of these operators. In practice we introduce a normalisation factor in the blocking algorithm, eq. (4), in order to prevent numerical overflows. In SU(2) we divide U^N_μ by $|U^N_\mu|$ so that the blocked variable is again an SU(2) group element, while in SU(3) we simply divide U^N_μ by 5. The precise form of the normalisation does not affect the algorithm.

In fig. 3 we show the normalised correlation function as calculated on a $6^3{\cdot}16$ lattice at $\beta=2.3$ in SU(2) using the standard Wilson action. We do not go beyond 2 blockings here because of the limited lattice size employed. We observe that both for the 0^+ and, most dramatically, for the 2^+ the correlation function drops much less steeply for the blocked wave functionals. Indeed the doubly blocked correlation function is approximately consistent with being a single exponential from $t=0$ to $t=2a$, which allows us to extract the mass values

$$m(0^+)a=1.15+0.08,$$
$$m(2^+)a=1.81\pm0.14. \tag{10}$$

(To extract these masses we have used, conservatively, the ratio $C_2(2a)/C_2(a)$.) This gives the ratio

$$m(2^+)/m(0^+)=1.57\pm0.16, \tag{11}$$

which is smaller than that obtained by previous, much less reliable, calculations [1] but differs significantly from the $m(0^+)\approx m(2^+)$ results obtained recently on very small volumes [4]. We do not wish here to make any firm statements about this mass ratio: in a forthcoming letter [7] we perform a finite-size study that attempts to clarify the question. The fact that we are here able to extract accurate 2^+ correlation functions out to $t=2a$ with only 5000 measurements illustrates the large gain obtained using our composite path algorithm.

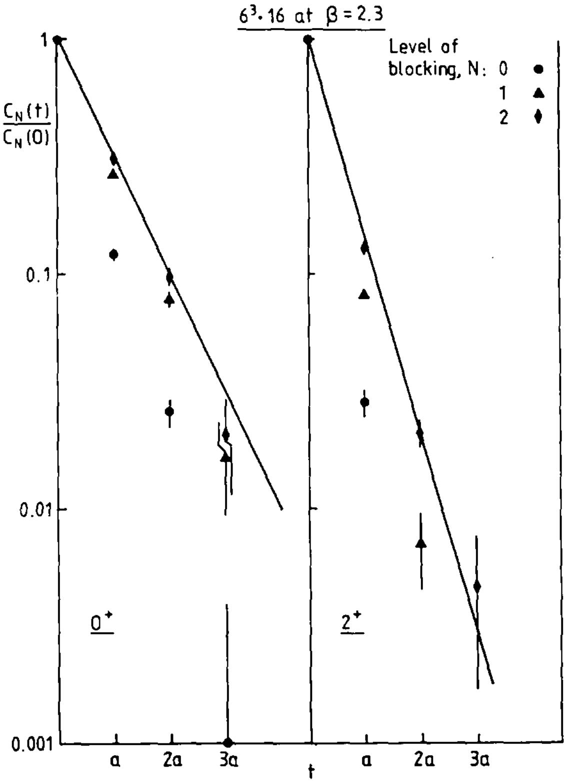

Fig. 3. 0^+ and 2^+ correlation functions, on a $6^3{\cdot}16$ lattice at $\beta=2.3$, for "plaquette" wave functionals formed out of blocked paths.

What is the quantitative gain? Consider first the 2^+. We known from previous high statistics studies that $C_0(2a)/C_0(a)$ for simple Wilson loops is smaller than the $C_2(a)/C_2(0)$ we obtain here with the twice blocked "plaquettes". On the other hand it turns out that the statistical errors on the twice blocked correlation function are nearly equal to those on the unblocked correlation function. Therefore the gain in CPU time is given by

$$\frac{\mathrm{CPU}[2^+:\text{unblocked}]}{\mathrm{CPU}[2^+:\text{twice blocked}]} \geqslant [C_0(a)/C_0(0)]^{-2}\sim 1000. \tag{12}$$

For the 0^+ the statistical errors on $C_2(t)$ are about twice those on $C_0(t)$ and so a similar analysis to the above gives

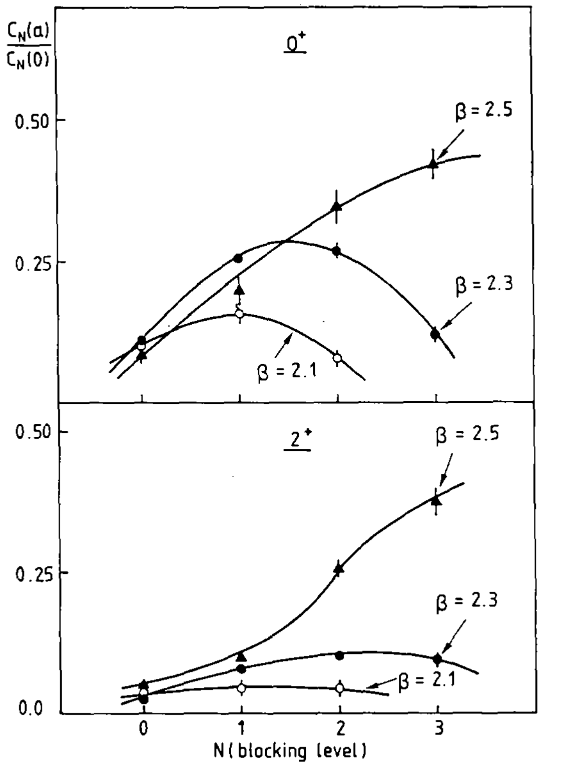

Fig. 4. 0^+ and 2^+ correlation functions at $t=a$, for various levels of blocking and various values of β.

$$\frac{\mathrm{CPU}[0^+:\text{unblocked}]}{\mathrm{CPU}[0^+:\text{twice blocked}]}\sim \mathrm{O}(50). \tag{13}$$

These gains are dramatic. In particular it will now be possible, for the first time, to calculate reliably the 2^+ glueball mass on lattice volumes large enough to be physically sensible.

We have seen that the method brings large improvements at $\beta=2.3$, which is just on the weak coupling side of the strong coupling "crossover". This is very desirable, of course, but it does not address our original aim of finding a method that does not get rapidly inefficient as β increases. To test how our method fares with increasing β we have performed calculations on a $4^3\cdot 16$ lattice at $\beta=2.1$ (1600 measurements), on an 8^4 lattice at $\beta=2.3$ (3000 measurements) and on a 12^4 lattice at $\beta=2.5$ (360 measurements). In fig. 4 we present the calculated values of the correlation function at $t=a$, for the various blocking levels at each value of β. Ideally one would hope that the value for $C(a)/C(0)$ would increase with β indicating that it was receiving substantial contributions from the decreasing (in lattice units) glueball mass, $ma(\beta)$, through the term $\exp[-ma(\beta)]$. As we already know and can see in fig. 4 this is not the case with the unblocked correlation function. The blocking, in contrast, does exactly what we would like. For any fixed N, $C_N(a)/C_N(0)$ first increases with β and then decreases. However the value of $\max_{\{N\}}[C_N(a)/C_N(0)]$ increases monotonically as β increases. This is as one would expect: as β increases the size of the glueball wave function increases in lattice units and hence we need to go to higher blocking levels (recall eq. (5)) to compensate. Using previous [8] high statistics calculations on a $10^3\cdot 12$ lattice at $\beta=2.5$ (unblocked of course) we can estimate our gain in computer time as we did previously at $\beta=2.3$:

$$\begin{aligned}\frac{\mathrm{CPU}(N=0)}{\mathrm{CPU}(\text{best }N)} &\geqslant \mathrm{O}(10^4):\ 0^+,\\ &\geqslant \mathrm{O}(10^5):\ 2^+.\end{aligned} \tag{14}$$

These are of course very rough estimates.

We have seen here that using composite paths for glueball wave functionals provides huge improvements in the effiency of numerical glueball mass calculations. The composite paths are generated by a fast iterative "blocking" algorithm. Obviously there is much flexibility in the details of the algorithm: one could use a different set of paths in the algorithm, or one could block at the level of one lattice spacing and generate blocked paths of any length, or one could mix paths of different N, etc. The size of our calculations is very modest: it is important to do sufficiently lengthy large-volume calculations at for example $\beta=2.5$ in SU(2) and $\beta=6.0$ in SU(3) to really test the method. Finally we remark that string tension and T_c calculations suffer from the same problems as glueball calculations – though less severely for dimensional reasons – and we expect that, as β increases, using composite paths in forming Wilson loops and lines will be of increasing benefit there also.

Note added. During the course of this work, we received ref. [9] in which singly blocked loops are used in a source calculation but apparently to no great advantage, in contrast to our results. In our approach the main benefit of blocking is to improve the desired overlaps and for this particular purpose any timelike blocking should be avoided, using (unimproved) sources is inappropriate, the stochastic choice of blocked variables introduces unnecessary extra noise and the blocking needs to be iterated a number of times that will increase with β.

My perception of the problems addressed here arose during a long series of glueball calculations: I am grateful to all my collaborators therein and especially to G. Schierholz. I thank H. Hansson for the radial wave-function example used in the text. Most of this work was performed while I was at the University of Southampton; also during visits to CERN and the Rutherford Laboratory upon whose IBM machines the calculations were performed. I am grateful to these institutions for their hospitality.

References

[1] M. Teper, Glueball masses in lattice QCD – progress and problems, Proc. Workshop on Non-perturbative methods (Montpellier, July 1985), ed. S. Narison (World Scientific, Singapore, 1985).

[2] Ph. de Forcrand, G. Schierholz, H. Schneider and M. Teper, Phys. Lett. B 152 (1985) 107.

[3] A. Patel, R. Gupta, G. Guralnik, G. Kilcup and S. Sharpe, Harvard preprint HUTP-86/A035.

[4] B. Berg and A. Billoire, Phys. Lett. B 166 (1986) 203;
B. Berg, A. Billoire and C. Vohlwinkel, Phys. Rev. Lett. 57 (1986) 400.

[5] M. Lüscher, Nucl. Phys. B 219 (1983) 233;
M. Lüscher and G. Münster, Nucl. Phys. B 232 (1984) 445.

[6] P. Hasenfratz, Erice Lectures (1984).

[7] M. Teper, Oxford preprint, Phys. Lett. B, to be published.

[8] G. Schierholz and M. Teper, Phys. Lett. B 136 (1984) 64, 69.

[9] T. De Grand and C. Peterson, Colorado preprint COLO-HEP-117 (June 1986).