Interaction Between Instantons in Gauge Theory

A. Amann
Supervisor: Dr. M. Yu. Kuchiev

November 27, 1995

Abstract

A gauge theory model providing an interaction between instantons is reviewed shortly. In this model there exists a specific condensate of the scalar field. Interaction between the scalar condensate, fermions and the gauge field results in an effective interaction between instantons. The action describing this interaction is calculated numerically. The numerical results agree with the asymptotic formula, which describes the interaction between two instantons for large separation. It is found that for small separation of instantons, the interaction is much smaller than predicted by the asymptotic formula. This result is important, because it greatly simplifies the consideration of the behavior of an ensemble of instantons.

Contents

Chapter 1

Introduction

1.1 Yang-Mills gauge fields

The concept of non-Abelian Gauge-Fields was first introduced to Quantum Field Theory in the work of C.N. Yang and R.L. Mills in 1954 [1]. For a more modern introduction of gauge theory, see [2]. Consider the Yang-Mills gauge potential which is a set of vector fields $A^a_\mu(x)$, where $a = 1, \ldots, N$ and $\mu = 1, 2, 3, 4$. Consider the Euclidean formulation of the theory, in which the rotation from real time to imaginary time, $t \mapsto -ix_4$, is performed. It is well known that the Euclidean formulation has advantages for a number of problems, see [2]. Define a field $A_\mu(x)$ of anti-hermitian traceless matrices by

$$A_\mu(x) = T^a A^a_\mu(x) \ . \tag{1.1}$$

Here the T^a are the generators of the underlying gauge group. In the following we are only interested in the case where the gauge group is the $SU(2)$ group acting on isospinor space. Then $N = 3$ and

$$T^a = \frac{\tau^a}{2i} \ . \tag{1.2}$$

where τ^a are the Pauli matrices operating on isospinor space. In this case the gauge field strength $F^a_{\mu\nu}$ is defined by

$$F^a_{\mu\nu} = \partial_\mu A^a_\nu - \partial_\nu A^a_\mu + \epsilon_{abc} A^b_\mu A^c_\nu \ , \tag{1.3}$$

ϵ_{abc} is the standard antisymmetric tensor. The matrix valued field strength is defined by

$$F_{\mu\nu} = \frac{1}{2i} \tau^a F^a_{\mu\nu} \ . \tag{1.4}$$

The action functional is defined by

$$S = \frac{1}{4g^2} \int F^a_{\mu\nu} F^a_{\mu\nu} d^4x \ , \tag{1.5}$$

where g is a constant called the gauge coupling constant.

In the semi-classical approximation one is looking for local minima of the action S. The conditions

$$\partial_\mu F^a_{\mu\nu} + \epsilon_{abc} A^b_\mu F^c_{\mu\nu} = 0 \tag{1.6}$$

have to be fulfilled for any local minimum. The equations (1.6) are called the Yang-Mills equations of motion.

1.2 Instantons

Since the Yang-Mills equations are coupled nonlinear partial differential equations, it is difficult to find exact solutions. The problem is simplified by considering specific solutions which satisfy the self-duality equation

$$F_{\mu\nu} = \pm {}^*F_{\mu\nu}, \tag{1.7}$$

where ${}^*F_{\mu\nu}$ is defined as

$${}^*F_{\mu\nu} = \frac{1}{2} \epsilon_{\mu\nu\lambda\rho} F_{\lambda\rho} \ , \tag{1.8}$$

$\epsilon_{\mu\nu\lambda\rho}$ is the standard antisymmetric tensor. The solutions of (1.7) satisfy as well the Yang-Mills equations (1.6). These solutions are called instanton solutions.

Instantons were discovered in 1975 by A. A. Belavin, A. M. Polyakov, A. S. Schwartz, and Yu. S. Tyupkin [3]. Analyzing the self-duality equation (1.7), they constructed the solution for the case of one instanton explicitly. The action (1.5) for instantons has the value

$$S = \frac{8\pi^2}{g^2} k \ , \tag{1.9}$$

where k is an integer ($k = 0, 1, 2, \ldots$) called the topological charge, or the number of instantons. Since instantons attain discrete action values, instanton solutions with different k can not be transformed into one another by continuous deformation of the gauge potential. One can think of an instanton solution with some specific topological charge k as consisting of k localized objects, called instantons. This justifies the term "number of instantons" for k. In order to describe one instanton, one needs to specify 8 parameters. 4 of them determine its position in space-time, 1 parameter accounts for its radius and 3 for its orientation in space time.

1.3 Interaction Between Instantons

The exact solution of (1.7) for arbitrary k and any choice of radii, positions and orientations of instantons was discovered by M. F. Atiyah, N. J. Hitchin, V. G. Drinfield, and Yu. I. Manin [4]. The construction they use is now called the ADHM-construction. For given k all instanton solutions have the same action (1.9). This means that there is no interaction between instantons in pure gauge field theory.

In the work of M. Yu. Kuchiev [5], it was suggested that gauge theory can describe the effect of gravity. The basic idea is to consider a condensate of instantons, in which identical orientation of instantons is preferable. In order to achieve this state it is desirable to find an interaction between instantons which depends on their orientation. This interaction can be compared with the interaction between magnetic moments in a ferromagnetic material. This interaction depends on the orientation of the magnetic moments and causes a state in which their identical orientation is preferable. In [6] a model was presented, in which an interaction between instantons exists. In this model, it is assumed that there exists a scalar condensate and a specific interaction between fermions and the scalar condensate. The effective interaction between instantons arises due to a one-loop fermion correction to the action. For the case of two instantons the asymptotic formula for the action S_{Fas} describing the interaction between two instantons for large separation is given by [6]

$$S_{Fas} = 2\zeta^2 \frac{\rho_1^2 \rho_2^2}{r^4} \sin^2 \gamma \ . \tag{1.10}$$

Here ρ_1, ρ_2 are the radii of the instantons, r is their separation and γ is the angle between the orientations of the instantons. ζ is defined by

$$\zeta = \frac{f\phi}{2m}, \tag{1.11}$$

where ϕ is the value of the scalar condensate, f is the coupling constant of the interaction between scalars and fermions and m is the fermion mass. Equation (1.10) describes the interaction between two instantons correctly, if $\rho_1, \rho_2 \ll r$ and has the desired minimum for identical orientation.

In order to describe the behavior of an ensemble of instantons, it is necessary to know how instantons interact for small separation $r \sim \rho_1,\ \rho_2$. In this region, the analytical result (1.10) is not valid. Therefore it is necessary to perform numerical calculations. This thesis aims to calculate the action S_F describing the interaction between two instantons numerically, using a method similar to the one described in [7]. It is found that $S_F \leq S_{Fas}$ for any $r,\ \rho_1,\ \rho_2$. In particular $S_F \ll S_{Fas}$ for $r \ll \rho_1, \rho_2$. It is important that for short separation,

S_F is strongly suppressed in comparison to the asymptotic value S_{Fas}, since this should greatly simplify the future calculation of the properties of the ensemble of instantons in the model [6].

Chapter 2

Calculation

2.1 ADHM Construction For Two Instantons

The basic tool we need for the following calculation is the ADHM construction [4]. The ADHM construction uses matrices of quaternions. Quaternions are a generalization of complex numbers, which allow the basic algebraic operations. The most important difference between complex numbers and quaternions is that the multiplication is not commutative. One can represent quaternions as 2×2 complex matrices, then the multiplication of quaternions corresponds to the usual multiplication of matrices.

Define

$$\tau_\mu^\pm = (\pm i\vec{\tau}, 1) \ , \tag{2.1}$$

where τ^a are the Pauli matrices acting on isospinor space and 1 is the 2×2 identity matrix. It is then possible to transform a 4-dimensional vector q_μ into a quaternion represented as a 2×2 matrix by:

$$q = q_\mu \tau_\mu^+ \ . \tag{2.2}$$

Similar to complex numbers, there exists a conjugation operation denoted by "*". The conjugate of (2.2) is given by

$$q^* = q_\mu \tau_\mu^- \ . \tag{2.3}$$

Conjugating corresponds to computing the hermitian adjoint of the 2×2 matrix representing a quaternion.

The real and imaginary part of a quaternion q are given by

$$\mathrm{Re}(q) \ = \ \frac{1}{2}(q + q^*) \ , \tag{2.4}$$

$$\mathrm{Im}(q) \;=\; \frac{1}{2}(q - q^*) \;. \tag{2.5}$$

Following [4] consider the 3×2 quaternion matrix $M(x)$, of the form

$$M(x) = \begin{pmatrix} q_1 & q_2 \\ y_1 - x & b \\ b & y_2 - x \end{pmatrix} \;. \tag{2.6}$$

Here q_1 and q_2 are quaternions describing the size ($\rho_i = |q_i| = \sqrt{q_i^* \, q_i}$) and the orientation of the instantons, y_1 and y_2 are their positions, b is defined as

$$b = \frac{y_1 - y_2}{2|y_1 - y_2|^2}(q_2^* \, q_1 - q_1^* \, q_2) \;. \tag{2.7}$$

and x is the coordinate in Euclidean space-time transformed to a quaternion via (2.2).

We can choose the coordinates so that the instantons have the real positions $y_1 = (0, 0, 0, +y)$ and $y_2 = (0, 0, 0, -y)$. $M(x)$ and b read then:

$$M(x) \;=\; \begin{pmatrix} q_1 & q_2 \\ y - x & b \\ b & -y - x \end{pmatrix} \;, \tag{2.8}$$

$$b \;=\; \frac{1}{4y}(q_2^* \, q_1 - q_1^* \, q_2) \;. \tag{2.9}$$

Define $R(x)$ as

$$R(x) \;=\; M^+(x) M(x) \tag{2.10}$$

$$\;=\; \begin{pmatrix} |q_1|^2 + |y - x|^2 + |b|^2 & \mathrm{Re}(q_1^* \, q_2) - (x^* \, b + b^* \, x) \\ \mathrm{Re}(q_1^* \, q_2) - (x^* \, b + b^* \, x) & |q_2|^2 + |y + x|^2 + |b|^2 \end{pmatrix} \tag{2.11}$$

where M^+ denotes the transposed matrix of M with each element being conjugated. $R(x)$ is a real and symmetric 2×2 matrix. By calculating its determinant, one can check that $R(x)$ is invertible.

There exists a 3-dimensional quaternion vector $N(x)$ which satisfies the conditions

$$N^+(x) M(x) \;=\; 0, \tag{2.12}$$

$$N^+(x) N(x) \;=\; 1. \tag{2.13}$$

Consider the gauge potential defined by

$$A_\mu = N^+(x) \, \partial_\mu N(x). \tag{2.14}$$

The corresponding field strength $F_{\mu\nu}$ given by (1.3, 1.4) obeys the self-duality equation (1.7). The action functional (1.5) in the pure gauge theory is

$$S = \frac{8\pi^2}{g^2} 2 \ . \tag{2.15}$$

A_μ given by (2.14) is the instanton solution for the case $k = 2$. The construction considered above is the ADHM-construction for two instantons.

2.2 Overview of calculation

Define C as

$$C = \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \ , \tag{2.16}$$

and consider the two dimensional quaternion vector $L(x)$

$$L(x) = R^{-1}(x)\, C^+ \, N(x). \tag{2.17}$$

Here $R(x)$ and $N(x)$ are defined in (2.10) and (2.12). The vector $L(x)$ describes the fermionic zero modes in the field of two instantons [7]. The existence of zero modes is due to the Atiyah-Singer index theorem [10], see also [11]. The role of zero modes in connection with instantons was first considered in [12].

In [7] it is found that in the case of two instantons the action describing the interaction between two instantons is given by the formula

$$S_F = \frac{1}{2}\zeta^2 |\vec{W}_{1,2}|^2 \ . \tag{2.18}$$

Here ζ is the same constant as in (1.10). $\vec{W}_{1,2}$ is defined as

$$\vec{W}_{1,2}(x) = \frac{2}{\pi^2} \,\mathrm{Re} \int L_1(x) i\vec{\tau} L_2^+(x) \ d^4x \ . \tag{2.19}$$

With the help of the matrix Σ,

$$\Sigma = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \tag{2.20}$$

$|\vec{W}_{1,2}|^2$ can be written as

$$|\vec{W}_{1,2}|^2 = \frac{1}{\pi^4} \left| \int L^+ \, \Sigma \, L \, d^4x \right|^2 \tag{2.21}$$

Define

$$I(x) = L^+(x)\,\Sigma\,L(x)\ , \tag{2.22}$$

and rewrite S_F in (2.18) as

$$S_F = \frac{1}{\pi^4}\zeta^2 \left| \int I(x)\,d^4x \right|^2 \tag{2.23}$$

The main purpose of this thesis is to calculate S_F given by (2.23). The first part of the calculation involves evaluation of an analytical expression for $I(x)$. This part is purely symbolic and will use the ADHM construction as the major tool. The second part consists of performing the integration in (2.23). Since $I(x)$ depends only on three independent variables containing x, one of the four integrations in (2.23) is trivial. One other integration can be done analytically, because $I(x)$ turns out to be a fraction of two polynomials (see also [6]). The remaining two dimensional integral is calculated numerically.

2.3 Analytical Expression for the Integrand

The purpose of this section is to find an analytical expression for the integrand $I(x)$ defined in (2.22). In the following the dependence on the parameter x is not denoted explicitly. Using the definition of L (2.17), I can be expressed as (recall that R is symmetric)

$$I = N^+\,C\,R^{-1}\Sigma R^{-1}\,C^+\,N. \tag{2.24}$$

R^{-1} can be expressed by the general formula for real non-degenerate 2×2 matrices

$$R^{-1} = \frac{1}{\det R}\Sigma^+\,R^+\,\Sigma\ . \tag{2.25}$$

Rewrite (2.22) :

$$I = N^+\,C\,R^{-1}\,\Sigma\,(\frac{1}{\det R}\Sigma^+\,R\,\Sigma)\,C^+\,N \tag{2.26}$$

$$= \frac{1}{\det R}\,N^+\,C\,\Sigma\,C^+\,N\ . \tag{2.27}$$

We want to find an explicit expression for C^+N. In order to simplify notation, define:

$$\hat{N} = C^+N = \begin{pmatrix} N_2 \\ N_3 \end{pmatrix}\ , \tag{2.28}$$

$$\hat{M} = C^+M = \begin{pmatrix} y-x & b \\ b & -y-x \end{pmatrix}\ , \tag{2.29}$$

$$Q = (q_1, q_2)\ . \tag{2.30}$$

Then condition (2.12) gives

$$\hat{N}^{+}\hat{M} = -N_1^{*}Q \ , \tag{2.31}$$

or with the inverse of $\hat{M}$

$$\hat{N}^{+} = -N_1^{*}Q\hat{M}^{-1} \ . \tag{2.32}$$

Rewriting the normalization condition (2.13) we get

$$|\hat{N}|^2 + |N_1|^2 = 1 \ . \tag{2.33}$$

To simplify calculation define the unnormalized vector $\hat{N}_u$ by

$$\hat{N}_u^{+} = -Q\,\hat{M}^{-1} \ . \tag{2.34}$$

$\hat{N}$ can be expressed in terms of $\hat{N}_u$ by

$$\hat{N} = \frac{1}{\sqrt{1 + |\hat{N}_u|^2}}\,\hat{N}_u \ . \tag{2.35}$$

One can easily check that $\hat{M}^{-1}$ can be expressed as the following product of matrices:

$$\hat{M}^{-1} = \Sigma^{+}\, b^{-1}\, \hat{M}\, b\, D^{-1}\, \Sigma \ , \tag{2.36}$$

where D is given by

$$D = \begin{pmatrix} (-y-x)b^{-1}(y-x)b - bb & 0 \\ 0 & (y-x)b^{-1}(-y-x)b - bb \end{pmatrix} \ . \tag{2.37}$$

Since $|D_{11}| = |D_{22}| = d$, D^{-1} can be written as

$$D^{-1} = \frac{1}{d^2}\,D^{+} \ . \tag{2.38}$$

Rewrite I in terms of $\hat{N}_u$:

$$I = \frac{1}{\det R\,(1 + |\hat{N}_u|^2)}\,\hat{N}_u^{+}\,\Sigma\,\hat{N}_u \ , \tag{2.39}$$

with

$$\hat{N}_u^{+} = -\frac{1}{d^2}Q\Sigma^{+}\, b^{-1}\, \hat{M}\, b\, D^{+}\, \Sigma \ . \tag{2.40}$$

Calculation of $1 + |\hat{N}_u|^2$ gives

$$1 + |\hat{N}_u|^2 = \frac{\det R}{|d|^2} \ . \tag{2.41}$$

Rewriting I (2.39) one gets:

$$I = \frac{d^2}{(\det R)^2}\, \hat{N}_u^+ \, \Sigma \, \hat{N}_u \ , \tag{2.42}$$

The problem of calculating I is reduced to the problem of multiplication of quaternion matrices, and the calculation of the determinant of R. One finds that I can be written as the fraction of two polynomials in which the parameter x appears only in the following combinations: $\mathrm{Re}(x^*\, b)$, $\mathrm{Im}(x^*\, b)$, x_4, $|x|^2$. To simplify the calculation one can use the fact, that we integrate I in (2.23) over the whole space $\mathbf{R^4}$. Consider the transformation

$$t : \left\{ \begin{array}{rcl} \mathrm{Im}(x) & \mapsto & 2\frac{\mathrm{Re}(x^*\, b)}{|b|^2} b - \mathrm{Im}(x) \\ \mathrm{Re}(x) & \mapsto & \mathrm{Re}(x) \end{array} \right. \tag{2.43}$$

The parameters $\mathrm{Re}(x^*\, b)$, x_4, $|x|^2$ are invariant under the transformation t, whereas $\mathrm{Im}(x^*\, b)$ changes sign. The denominator of I does not depend on $\mathrm{Im}(x^*\, b)$. Since we will integrate over the whole space $\mathbf{R^4}$, any term in the numerator of I which is linear in $\mathrm{Im}(x^*\, b)$, will not contribute to the final integral. These terms are neglected in the following. Higher powers of $\mathrm{Im}(x^*\, b)$ can be simplified by the identity

$$|\,\mathrm{Im}(x^*\, b)|^2 = (|x|^2 - |\,\mathrm{Re}(x)|^2)|b|^2 - |\,\mathrm{Re}(x^* b)|^2 \tag{2.44}$$

After a long but straightforward calculation one finds for $\hat{N}_u^+ \Sigma \hat{N}_u$ in (2.42):

$$\begin{aligned} \hat{N}_u^+ \Sigma \hat{N}_u = {} & 2\frac{\tilde{b}}{|b|^2 |d|^4} \\ & \Big\{ (|d|^2|b|^2 - 2|g|^2) \\ & [\mathrm{Re}(x)(|q_2|^2 - |q_1|^2) \\ & -2y(y^2 + |b|^2 - |x|^2 + \frac{|q_1|^2 + |q_2|^2}{2} + \frac{\mathrm{Re}(x^* b)\,\mathrm{Re}(q_1^* q_2)}{|b|^2})] \\ & +4|g|^2[(|b|^2 + |x|^2 - y^2)(y + \frac{1}{4y}(|q_1|^2 + |q_2|^2)) \\ & + \mathrm{Re}(q_1^* q_2)\,\mathrm{Re}(x^*\, b)\frac{1}{y}(1 - \frac{y^2}{b^2})] \Big\} \end{aligned} \tag{2.45}$$

where $\tilde{b}$ and g are defined as

$$\tilde{b} = \frac{1}{4y}(q_1 q_2^* - q_2 q_1^*) \tag{2.46}$$

$$g = y|\mathrm{Im}(x^*\, b)|^2 \tag{2.47}$$

2.4 Integration

The purpose of this section is it to perform the integration (2.23). Parameterize $\mathbf{R^4}$ as $x_\mu = (\vec{x}, x_4)$ and use the standard spherical coordinates $r_s,\ \theta_s,\ \phi_s$ for $\vec{x}$. In terms of quaternions we have the following correspondence:

$$\begin{aligned} x_4 &= \mathrm{Re}(x) && (2.48)\\ r_s^2 &= |x|^2 - x_4^2 && (2.49)\\ \cos\theta_s &= \frac{\mathrm{Re}(x^* b)}{|\,\mathrm{Im}(x)|\,|b|} && (2.50) \end{aligned}$$

I depends on the parameters $\mathrm{Re}(x),\ |x|^2,\ \mathrm{Re}(x^*b)$, but not on ϕ_s. We can use this fact to rewrite the integral in (2.23) as

$$\begin{aligned} \int_{\mathbf{R}^4} I(x)\, dx^4 &= \int_{-\infty}^{\infty}\int_{\mathbf{R}^3} I\, d^3x\, dx_4 && (2.51)\\ &= \int_{-\infty}^{\infty}\int_0^{\infty}\int_0^{\pi}\int_0^{2\pi} I r_s^2 \sin\theta_s\, d\phi_s\, d\theta_s\, dr_s\, dx_4 && (2.52)\\ &= 2\pi\int_{-\infty}^{\infty}\int_0^{\infty}\int_0^{\pi} I r_s^2 \sin\theta_s\, d\theta_s\, dr_s\, dx_4 && (2.53)\\ &= 2\pi\int_{-\infty}^{\infty}\int_0^{\infty}\int_{-1}^{1} I r_s^2\, dt\, dr_s\, dx_4\ , && (2.54) \end{aligned}$$

with the substitution $\cos\theta_s = t$. One observes, that I is of the form

$$I = \tilde{b}\frac{a_3t^3 + a_2t^2 + a_1t + a_0}{(c_0 - c_1t - c_2t^2)^2(d_2t^2 - d_0)} \qquad (2.55)$$

where all coefficients are real and positive. The only remaining quaternion $\tilde{b}$ (2.46) is a constant. The fraction of polynomials in (2.55) is integrated symbolically with respect to t by the symbolic software package Mathematica©. Mathematica is also used to determine the coefficients $a_i,\ c_i,\ d_i$ of the polynomials. The Mathematica programs are included in Appendix A.

The remaining two dimensional integral over the parameters r_s and x_4 is calculated numerically by a C program. The region where r_s and x_4 are small gives the main contribution to the integral in (2.54). This fact is used to make the program more efficient. The considered region is split up into several subregions, and the integral of each subregion is calculated by evaluating the integrand on a number of points. The density of points in a region decreases, as the distance of the region from the origin increases. The program is included in Appendix B.

2.5 Consistency of Calculation

In order to check the correctness of the numerical integration, we compare the results with two asymptotic formulae. The first one is the asymptotic formula for large separation given by equation (1.10). The second one is the asymptotic formula for small separation [7] given by

$$S_{F_0} = \zeta^2 \frac{r^4}{8\rho_1^2\rho_2^2 \sin^2\gamma} \ . \tag{2.56}$$

S_{F_0} describes the interaction between two instantons in the region where $r^2 \ll \rho_1\rho_2 \sin\zeta$.

The C program in Appendix B reproduces both asymptotic formulae with an accuracy higher than 2%. This is an important check of the consistency of the methods used.

The program allows the user to specify the region of integration as well as the density of the evaluated points. Both parameters influence the accuracy of the results. To save CPU time we allow the error to be about 4%. This reduces the time needed for one integration and makes it possible to calculate S_F systematically for various configurations of instantons (see Chapter 3).

Chapter 3

Results

Define F^2 as

$$F^2 = \frac{S_F}{S_{Fas}} \ , \tag{3.1}$$

and r_n as

$$r_n = \frac{r}{\rho_1 + \rho_2} \ . \tag{3.2}$$

F^2 depends on the parameters $\rho_1,\ \rho_2,\ r_n$ and $\cos\gamma$. The scale of ρ_1 and ρ_2 does not change F^2. Therefore we set $\rho_2 = 1$. With the C program in Appendix B, F^2 was calculated in a range of parameters given in Table 3.1.

Consider the case where $\rho_1 = 1$ and $\cos\gamma = 1$. Fig. 3.1 shows F^2 as a function of r_n. One observes that there is a sharp drop of F^2 for small r_n and that $F^2 \to 0$ for $r_n \to 0$. For large r_n, F^2 tends asymptotically to 1, in agreement with the asymptotic formula (1.10). Notice that $S_F \ll S_{Fas}$ for $r_n \ll 1$. This result is important for future consideration of the mean field approximation for the instanton ensemble. Averaging over the radii of the instantons involves an integral of the form $\int S_F\ d^4x$. When using naively $S_F \approx S_{Fas}$, there appears a logarithmic divergence at the origin, since $S_{Fas} \sim \frac{1}{r^4}$ (1.10). If one uses $S_F = F^2 S_{Fas}$ instead of $S_F \approx S_{Fas}$, then the divergence disappears because of

Parameter	min	max	step
ρ_1	1	10	1
$\cos\gamma$	0	0.9	0.3
r_n	0.1	3.4	0.1

Table 3.1: Range of Parameters for which F^2 was calculated

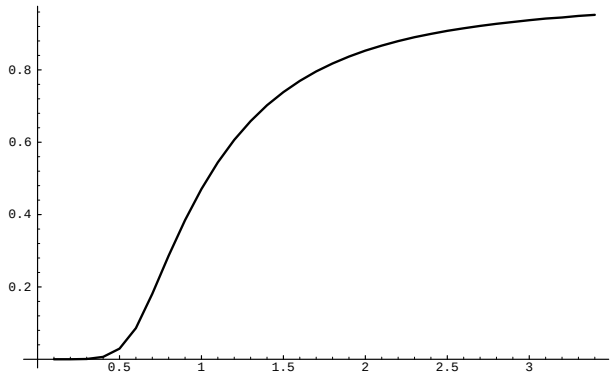

Figure 3.1: F^2 vs. r_n with $\rho_1 = 1,\ \rho_2 = 1$ and $\cos\gamma = 0$

r_n	F^2	r_n	F^2	r_n	F^2
0.1	0.00	1.1	0.54	2.1	0.86
0.2	0.00	1.2	0.60	2.2	0.87
0.3	0.00	1.3	0.65	2.3	0.89
0.4	0.00	1.4	0.70	2.4	0.89
0.5	0.02	1.5	0.73	2.5	0.90
0.6	0.08	1.6	0.76	2.6	0.91
0.7	0.18	1.7	0.79	2.7	0.92
0.8	0.28	1.8	0.81	2.8	0.92
0.9	0.38	1.9	0.83	2.9	0.93
1.0	0.47	2.0	0.85	3.0	0.93

Table 3.2: numerical values for $\rho_1 = \rho_2 = 1,\ \cos\gamma = 0$

the observed behavior of F^2 at the origin.

The graph in Fig. 3.1 is used as reference and is included in the following diagrams. A table of the numerical values of this graph is given in Table 3.2

Consider Fig. 3.2 which presents the results for $\rho_1 = 2$ and $\cos\gamma = 0$. One observes that the curve for $\rho_1 = 2$ and $\cos\gamma = 0$ is almost identical with the reference curve, although the ratio of the radii of the instantons has changed from $1:1$ to $2:1$. Consider Fig. 3.3. The ratio of the radii is $10:1$ and the graph differs more from the reference graph than in Fig. 3.2, but the difference is still comparatively small. We see that variation of the radii of the instantons does not manifest itself in the dependence of F^2 on r_n.

Consider Fig. 3.4 which describes the configuration $\rho_1 = 10$ and $\cos\gamma = 0.9$. We see that although $\cos\gamma$ has changed from 0 to 0.9, the diagram looks

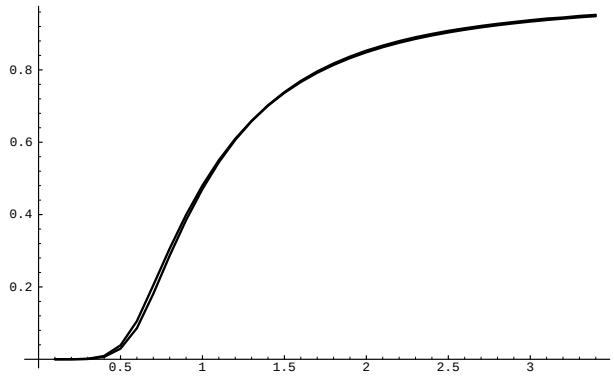

Figure 3.2: $\rho_1 = 2$, $\cos\gamma = 0$

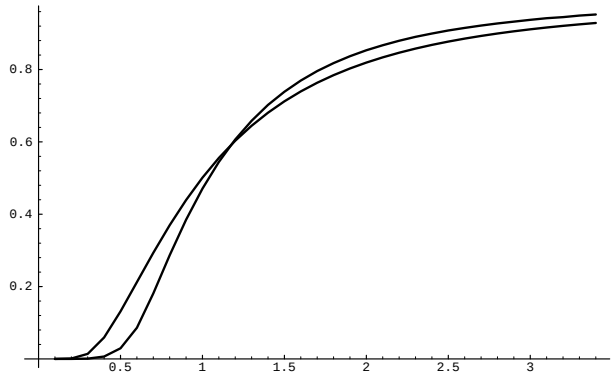

Figure 3.3: $\rho_1 = 10$ and $\cos\gamma = 0$

approximately the same as in Fig 3.3. We conclude that variation of γ plays a minor role as well.

The result is that F^2 depends mainly on r_n as shown in the reference diagram Fig. 3.1. F^2 is almost independent of ρ_1, ρ_2 and $\cos\gamma$. This has been verified by calculation in the range of parameters given in Table 3.1.

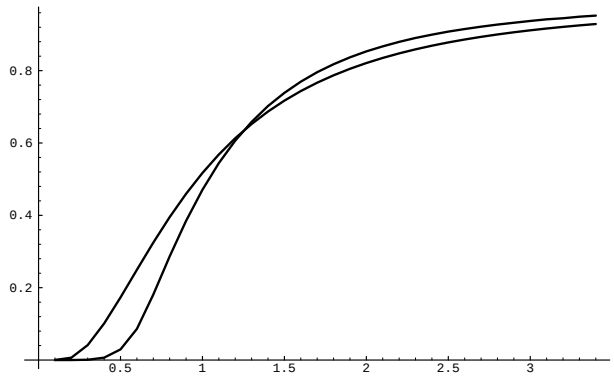

Figure 3.4: $\rho_1 = 10$ and $\cos\gamma = 0.9$

Chapter 4

Conclusion

This thesis demonstrates that for short distances the interaction between instantons, existing in the model [6], is much smaller than predicted by the asymptotic formula (1.10).

This result is very important for the future consideration of the mean field approximation of an ensemble of instantons, since it greatly simplifies the calculation.

Appendix A

Mathematica Programs

A.1 Integrate Fraction of Polynomials

The following Mathematica program calculates the integral

$$\int_{-1}^{1} \frac{aa + ba\,t + ca\,t^2 + da\,t^3}{(ka - kb\,t - kc\,t^2)^2(kd - ke\,t^2)} dt \ , \tag{A.1}$$

symbolically. The result is used by the C Program for numerical integration.

```
f = Expand[(aa + ba t + ca t^2 + da t^3 )/((ka - kb t - kc t^2)
(kd - ke t^2 ))]
fint =D[Integrate[-f,{t,-1,1}],ka] >>ffint
CForm[fint] >> fint.cform (* for use in C-program *)
```

A.2 Calculate Coefficients for C-Program

This Mathematica program calculates the coefficients $aa, \ldots, da$ and $ka, \ldots, ke$ appearing in (A.1). The coefficients are saved in a form to be included in the C program.

```
(*
Parameters are:

x4 = Re(x)
```

```
xsq = |x|^2
xbre = Re(Conj(x) * b)

y = y

q1sq = |q1|^2
q2sq = |q2|^2
qqre = Re(Conj(q1) * q2)
*)

Remove[x4,xsq,xbre]
Remove[y]
Remove[q1sq,q2sq,qqre]

ysq = y^2
bsq = b^2
rsq = r^2

xsq = rsq + x4^2
xbre = t b r

gsq = Simplify[4 ysq ((xsq - x4^2) bsq - xbre^2)]

Dsq = Simplify[((xsq + ysq)^2 + bsq^2 - 4 ysq x4^2
- 4 xbre^2 + 2 xsq bsq - 2 ysq bsq)]

term1 = ((Dsq bsq - 2 gsq)
        * (x4 (q1sq - q2sq)
        - 2y ( (ysq+bsq-xsq) + 1/2 (q1sq+q2sq) - 1/bsq (xbre qqre))
           )
        )
(* term1 = first term in (51) *)

term2 = ( 4 gsq
          * ( (bsq + xsq -ysq) * (y + 1/(4y) (q1sq +q2sq))
            + (y/bsq - 1/y) qqre xbre
            )
        )
(* term2 = second term in (51) *)

Pa = term1 + term2
(* Pa is the Numerator *)

R11 = q1sq + xsq + 2 x4 y + ysq + bsq
R22 = q2sq + xsq - 2 x4 y + ysq + bsq
```

```
R12 = qqre + 2 xbre
detR = R11 R22 - R12^2

denominator = (detR^2 Dsq)

detRseries = Series[detR,{t,0,10}]
Dsqseries = Series[Dsq,{t,0,10}]

ka = detRseries[[3,1]]  (* Coefficients in denominator *)
kb = - detRseries[[3,2]]
kc = - detRseries[[3,3]]
kd = Dsqseries[[3,1]]
ke = - Dsqseries[[3,3]]

CForm[ka] >> ka.cform  (* Save for use in C-Program *)
CForm[kb] >> kb.cform
CForm[kc] >> kc.cform
CForm[kd] >> kd.cform
CForm[ke] >> ke.cform


Paseries = Simplify[Series[Pa,{t,0,10}]]

aa = Paseries[[3]][[1]]  (* Coefficients in numerator *)
ba = Paseries[[3]][[2]]
ca = Paseries[[3]][[3]]
da = Paseries[[3]][[4]]

CForm[aa] >> aa.cform  (* Save for use in C-Program *)
CForm[ba] >> ba.cform
CForm[ca] >> ca.cform
CForm[da] >> da.cform


(* The following is for checking the results *)
integrand = 2 * Pi * rsq * fint >> integrand

CForm[integrand] >> intcform
```

Appendix B

C Program

The program calculates F^2 for different r_n, ρ_1, ρ_2 and $\cos\zeta$

```
#include <stdio.h>
#include <math.h>

double y,ysq,q1sq, q2sq, qqre,b, bsq;


/* A(r,x4) returns the value of the integrand at the point (r,x4) */

double A(double r,double x4 )
{

double ka, kb, kc, kd, ke;
double aa, ba, ca, da, fint;
double kasq, kbsq, kcsq, kdsq, kesq;
double x4sq, rsq;


x4sq = x4*x4;
rsq = r*r;

/* The following formulas are evaluated by mathematica  */
ka = -qqre*qqre + (bsq + q2sq + rsq + x4sq - 2*x4*y +
      ysq)*(bsq + q1sq + rsq + x4sq + 2*x4*y +
      ysq);

kb = 4*b*qqre*r;
```

```
kc = 4*bsq*rsq;

kd = pow(b,4.0) + 2*bsq*(rsq + x4sq) -
   2*bsq*ysq - 4*x4sq*ysq +
   pow(rsq + x4sq + ysq,2.0);

ke = 4*bsq*rsq;

kasq = ka*ka;
kbsq = kb*kb;
kcsq = kc*kc;
kdsq = kd*kd;
kesq = ke*ke;



aa = 4*bsq*rsq*y*(bsq + rsq + x4sq -
      ysq)*(q1sq + q2sq + 4*ysq) +
   bsq*(bsq*bsq + 2*bsq*rsq + rsq*rsq +
      2*bsq*x4sq + 2*rsq*x4sq + x4sq*x4sq -
      2*bsq*ysq - 6*rsq*ysq -
      2*x4sq*ysq + ysq*ysq)*
    ((q1sq - q2sq)*x4 - 2*y*(bsq + q1sq/2 + q2sq/2 - rsq -
         x4sq + ysq));

ba = 2*b*qqre*r*y*(bsq*bsq - 6*bsq*rsq + rsq*rsq +
     2*bsq*x4sq + 2*rsq*x4sq + x4sq*x4sq -
     2*bsq*ysq + 2*rsq*ysq -
     2*x4sq*ysq + ysq*ysq);

ca = -4*bsq*rsq*y*(bsq + rsq + x4sq -
      ysq)*(q1sq + q2sq + 4*ysq) +
   4*bsq*rsq*(bsq - 2*ysq)*
    (-(q1sq*x4) + q2sq*x4 + 2*bsq*y + q1sq*y + q2sq*y -
      2*rsq*y - 2*x4sq*y + 2*pow(y,3.0));

da = 8*pow(b,3.0)*qqre*pow(r,3.0)*y;


/*
the formula for fint is the result of the general integration
of a fraction of two polynomials of the considered form
*/

fint = rsq *
```

```
(2*(kb-2*kc)*(da*pow(kb,3.0)*kd + 3*da*ka*kb*kc*kd - ca*kbsq*kc*kd -
       2*ca*ka*kcsq*kd + ba*kb*kcsq*kd - 2*aa*pow(kc,3.0)*kd -
       da*kasq*kb*ke + 2*ca*kasq*kc*ke + ba*ka*kb*kc*ke +
       aa*kbsq*kc*ke + 2*aa*ka*kcsq*ke)/
    (pow(kbsq + 4*ka*kc,2.0)*
      (1 - pow(kb - 2*kc,2.0)/(kbsq + 4*ka*kc))*
      (-(kcsq*kdsq) + kbsq*kd*ke + 2*ka*kc*kd*ke -
        kasq*kesq)) -
   2*(kb + 2*kc)*(da*pow(kb,3.0)*kd + 3*da*ka*kb*kc*kd -
       ca*kbsq*kc*kd - 2*ca*ka*kcsq*kd + ba*kb*kcsq*kd -
       2*aa*pow(kc,3.0)*kd - da*kasq*kb*ke + 2*ca*kasq*kc*ke +
       ba*ka*kb*kc*ke + aa*kbsq*kc*ke + 2*aa*ka*kcsq*ke)/
    (pow(kbsq + 4*ka*kc,2.0)*
      (1 - pow(kb + 2*kc,2.0)/(kbsq + 4*ka*kc))*
      (-(kcsq*kdsq) + kbsq*kd*ke + 2*ka*kc*kd*ke -
        kasq*kesq)) +
   (-(da*kbsq*kd) - da*ka*kc*kd + ca*kb*kc*kd - ba*kcsq*kd +
      da*kasq*ke + ba*ka*kc*ke + aa*kb*kc*ke)/
    (2*(ka - kb - kc)*kc*(kcsq*kdsq - kbsq*kd*ke -
        2*ka*kc*kd*ke + kasq*kesq)) -
   (-(da*kbsq*kd) - da*ka*kc*kd + ca*kb*kc*kd - ba*kcsq*kd +
      da*kasq*ke + ba*ka*kc*ke + aa*kb*kc*ke)/
    (2*(ka + kb - kc)*kc*(kcsq*kdsq - kbsq*kd*ke -
        2*ka*kc*kd*ke + kasq*kesq)) +
   (da*pow(kb,3.0)*kd + 3*da*ka*kb*kc*kd - ca*kbsq*kc*kd -
       2*ca*ka*kcsq*kd + ba*kb*kcsq*kd - 2*aa*pow(kc,3.0)*kd -
       da*kasq*kb*ke + 2*ca*kasq*kc*ke + ba*ka*kb*kc*ke +
       aa*kbsq*kc*ke + 2*aa*ka*kcsq*ke)*
     (2*kc*kd*ke - 2*ka*kesq)*
     atanh((kb - 2*kc)/sqrt(kbsq + 4*ka*kc))/
    (kc*sqrt(kbsq + 4*ka*kc)*
      pow(-(kcsq*kdsq) + kbsq*kd*ke + 2*ka*kc*kd*ke -
        kasq*kesq,2.0)) -
   (3*da*kb*kc*kd - 2*ca*kcsq*kd - 2*da*ka*kb*ke + 4*ca*ka*kc*ke +
       ba*kb*kc*ke + 2*aa*kcsq*ke)*
     atanh((kb - 2*kc)/sqrt(kbsq + 4*ka*kc))/
    (kc*sqrt(kbsq + 4*ka*kc)*
      (-(kcsq*kdsq) + kbsq*kd*ke + 2*ka*kc*kd*ke -
        kasq*kesq)) +
   2*(da*pow(kb,3.0)*kd + 3*da*ka*kb*kc*kd - ca*kbsq*kc*kd -
       2*ca*ka*kcsq*kd + ba*kb*kcsq*kd - 2*aa*pow(kc,3.0)*kd -
       da*kasq*kb*ke + 2*ca*kasq*kc*ke + ba*ka*kb*kc*ke +
       aa*kbsq*kc*ke + 2*aa*ka*kcsq*ke)*
     atanh((kb - 2*kc)/sqrt(kbsq + 4*ka*kc))/
    (pow(sqrt(kbsq + 4*ka*kc),3.0)*
```

```
   (-(kcsq*kdsq) + kbsq*kd*ke + 2*ka*kc*kd*ke -
     kasq*kesq)) -
(da*pow(kb,3.0)*kd + 3*da*ka*kb*kc*kd - ca*kbsq*kc*kd -
    2*ca*ka*kcsq*kd + ba*kb*kcsq*kd - 2*aa*pow(kc,3.0)*kd -
    da*kasq*kb*ke + 2*ca*kasq*kc*ke + ba*ka*kb*kc*ke +
    aa*kbsq*kc*ke + 2*aa*ka*kcsq*ke)*
  (2*kc*kd*ke - 2*ka*kesq)*
  atanh((kb + 2*kc)/sqrt(kbsq + 4*ka*kc))/
 (kc*sqrt(kbsq + 4*ka*kc)*
   pow(-(kcsq*kdsq) + kbsq*kd*ke + 2*ka*kc*kd*ke -
     kasq*kesq,2.0)) +
(3*da*kb*kc*kd - 2*ca*kcsq*kd - 2*da*ka*kb*ke + 4*ca*ka*kc*ke +
    ba*kb*kc*ke + 2*aa*kcsq*ke)*
  atanh((kb + 2*kc)/sqrt(kbsq + 4*ka*kc))/
 (kc*sqrt(kbsq + 4*ka*kc)*
   (-(kcsq*kdsq) + kbsq*kd*ke + 2*ka*kc*kd*ke -
     kasq*kesq)) -
2*(da*pow(kb,3.0)*kd + 3*da*ka*kb*kc*kd - ca*kbsq*kc*kd -
    2*ca*ka*kcsq*kd + ba*kb*kcsq*kd - 2*aa*pow(kc,3.0)*kd -
    da*kasq*kb*ke + 2*ca*kasq*kc*ke + ba*ka*kb*kc*ke +
    aa*kbsq*kc*ke + 2*aa*ka*kcsq*ke)*
  atanh((kb + 2*kc)/sqrt(kbsq + 4*ka*kc))/
 (pow(sqrt(kbsq + 4*ka*kc),3.0)*
   (-(kcsq*kdsq) + kbsq*kd*ke + 2*ka*kc*kd*ke -
     kasq*kesq)) -
2*(-2*kc*kd*ke + 2*ka*kesq)*
  (da*kb*kdsq - ca*kc*kdsq + ca*ka*kd*ke + ba*kb*kd*ke -
    aa*kc*kd*ke + aa*ka*kesq)*atanh(sqrt(ke)/sqrt(kd))/
 (sqrt(kd)*sqrt(ke)*pow(kcsq*kdsq - kbsq*kd*ke -
     2*ka*kc*kd*ke + kasq*kesq,2.0)) +
2*(ca*kd*ke + aa*kesq)*atanh(sqrt(ke)/sqrt(kd))/
 (sqrt(kd)*sqrt(ke)*(kcsq*kdsq - kbsq*kd*ke -
     2*ka*kc*kd*ke + kasq*kesq)) -
(-(da*kbsq*kd) - da*ka*kc*kd + ca*kb*kc*kd - ba*kcsq*kd +
    da*kasq*ke + ba*ka*kc*ke + aa*kb*kc*ke)*
  (-2*kc*kd*ke + 2*ka*kesq)*log(ka - kb - kc)/
 (2*kc*pow(kcsq*kdsq - kbsq*kd*ke - 2*ka*kc*kd*ke +
     kasq*kesq,2.0)) +
(-(da*kc*kd) + 2*da*ka*ke + ba*kc*ke)*log(ka - kb - kc)/
 (2*kc*(kcsq*kdsq - kbsq*kd*ke - 2*ka*kc*kd*ke +
     kasq*kesq)) +
(-(da*kbsq*kd) - da*ka*kc*kd + ca*kb*kc*kd - ba*kcsq*kd +
    da*kasq*ke + ba*ka*kc*ke + aa*kb*kc*ke)*
  (-2*kc*kd*ke + 2*ka*kesq)*log(ka + kb - kc)/
 (2*kc*pow(kcsq*kdsq - kbsq*kd*ke - 2*ka*kc*kd*ke +
```

```
        kasq*kesq,2.0)) -
   (-(da*kc*kd) + 2*da*ka*ke + ba*kc*ke)*log(ka + kb - kc)/
    (2*kc*(kcsq*kdsq - kbsq*kd*ke - 2*ka*kc*kd*ke +
     kasq*kesq)));

return(fint);
}


/*
region integrates over a specified region
*/
double region(double rmin,double rmax,
      double x4min,double  x4max,double rstep,double  x4step )
{
  double area, r, x4, result = 0.0;
  double rrstep, rx4step;

  rrstep = (rmax - rmin)/ceil((rmax-rmin)/rstep);
  rx4step = (x4max - x4min)/ceil((x4max-x4min)/x4step);
  rrstep = rstep; rx4step = x4step;

  area = rrstep * rx4step ;

  for(r= rmax - rrstep/2; r > rmin; r -= rrstep)
  {
    for(x4= x4max -rx4step/2; x4 > x4min; x4 -= rx4step)
    {
      result +=  A(r,x4);
    }
  }

  result = result * area;
  return(result);
}


main()
{

  double cosphi,cosphistep, cosphimax, cosphimin;
  double q2,q1, q1step, q1max, q1min;
  double x4max,rmax;
  double x4step, rstep;
  double result=0.0;
```

```
  double yscal, yscalmin, yscalmax,yscalstep;

  double W, Wasym;
  int i,j,gridsize;


/* Ask for parameters */
  printf("\n\n\t Please specify the parameters\n\n");

  printf("\nq1min ?");
  scanf("%lf",&q1min);
  printf("\nq1max ?");
  scanf("%lf",&q1max);
  printf("\nq1step ?");
  scanf("%lf",&q1step);

  printf("\nq2 ?");
  scanf("%lf",&q2);

  printf("\ncosphimin ?");
  scanf("%lf",&cosphimin);
  printf("\ncosphimax ?");
  scanf("%lf",&cosphimax);
  printf("\ncosphistep ?");
  scanf("%lf",&cosphistep);

  printf("\nyscalmin ?");
  scanf("%lf",&yscalmin);
  printf("\nyscalmax ?");
  scanf("%lf",&yscalmax);
  printf("\nyscalstep ?");
  scanf("%lf",&yscalstep);

  printf("\nrmax ?");
  scanf("%lf",&rmax);
  printf("\nx4max ?");
  scanf("%lf",&x4max);

  printf("\n\nStepsize in r-direction? ");
  scanf("%lf",&rstep);
  printf("\n\nStepsize in x4-direction? ");
  scanf("%lf",&x4step);
  printf("\n\nGridsize? ");
  scanf("%i",&gridsize);
```

```
  for(q1 = q1min; q1 <= q1max; q1 += q1step)
  {
    for(cosphi=cosphimin; cosphi<=cosphimax; cosphi += cosphistep)
    {
      for(yscal = yscalmin ; yscal <= yscalmax; yscal += yscalstep)
      {
q1sq = q1 * q1;
q2sq = q2 * q2;
qqre = q1*q2*cosphi;
result = 0.0;
y = (q1+q2) * yscal / 2;
ysq = pow(y,2.0);
bsq = (q1sq * q2sq - pow(qqre,2.0))/pow((2*y),2.0);
b = sqrt(bsq);

for(i= - gridsize; i<= gridsize ; i++)
{
    for(j= 0 ;j <=gridsize ; j++)
    {
result += region(j*rmax, (j+1)*rmax ,
 i*x4max - x4max/2, i*x4max + x4max/2,
 (abs(j)+abs(i)+1)*rstep,
 (abs(j)+abs(i)+1)*x4step);
    }
}

W = 16 * pow(result,2.0)/(bsq * pow(M_PI,2.0));
Wasym = bsq/ ysq;

/* Output */
printf("%lf %lf %lf %lf %lf\n",q1, q2, cosphi,yscal, W/Wasym);

      }
    }
  }
}
```

Bibliography

[1] C. N. Yang, R. L. Mills, Phys. Rev. **96**, 191 (1954).

[2] D. Bailin, A. Love; Introduction to gauge field theory; Institute of Physics Pub; Bristol, Philadelphia (1993).

[3] A. A. Belavin, A. M. Polyakov, A. S. Schwartz, and Yu. S. Tyupkin, Phys. Lett. **59B**, 85 (1975).

[4] M. F. Atiyah, N. J. Hitchin, V. G. Drinfield, and Yu. I. Manin, Phys. Lett. **A65**, 185 (1978).

[5] M.Yu.Kuchiev, Europhys.Lett. **28**, 539 (1994).

[6] M. Yu. Kuchiev, Europhys.Lett. **30**, 255 (1994).

[7] M. Yu. Kuchiev, Orientation-Dependent Interaction Between Instantons, to be published, (1995).

[8] M. K. Prasad, Physica **1D**8, 167 (1980).

[9] A. M. Polyakov, *Gauge fields and strings* (Harwood Academic Publishers, Chur, London, Paris, NY, Melbourne) (1987).

[10] M. F. Atiyah, I.M.Singer, Ann.Math. **87**, 484 (1968).

[11] J. Kiskis, Phys.Rev. D **15**, 2329 (1977).

[12] G. 't Hooft, Phys. Rev. Let**37**, 8 (1976).

[13] A. I. Vainshtein, V. I. Zakharov, V. A. Novikov, and M. A. Shifman, Usp. Fiz. Nauk **136**, 553 (1982).

[14] C. Callan, K. Dashen, and D. Gross, Phys. Rev. D **17**, 2717 (1978); *ibid* **19**, 1826 (1979)

Acknowledgment

A big thankyou to my supervisor Michael Kuchiev, for supporting me at every stage of this work with helpful instructions.

My work was supported by the Deutcher Akademischer Austausch Dienst and the Australian Government. I want to thank Prof. U. Mebold for making my visit in Sydney possible.