J. Phys. A: Math. Gen. 17 (1984) 3063–3066. Printed in Great Britain

COMMENT

Apartheid in the Dirac theory of constraints†

Mark J Gotay‡* and James M Nester§

‡ Department of Mathematics and Statistics, University of Calgary, Calgary, Alberta, Canada T2N 1N4

§ Department of Physics, University of Alberta, Edmonton, Alberta, Canada T6G 2J1

Received 17 May 1984

Abstract. We investigate the extent to which first- and second-class constraints decouple in the Dirac constraint algorithm for degenerate dynamical systems. We find the two classes to be inextricably intertwined. Some consequences of this failure of 'apartheid' are discussed.

Degenerate dynamical systems are characterised by the presence of constraints on the admissible Cauchy data for their evolution equations¶. Such constraints fall into two categories: the 'first' and 'second' classes‖. First-class constraints are correlated, at least to some extent, with the gauge properties of the system (Dirac 1964, Gotay 1983, Gotay and Nester 1979, Gotay *et al* 1978, Hanson *et al* 1976, Sundermeyer 1982), while second-class constraints reflect the appearance of non-dynamic degrees of freedom in the theory (Dirac 1964, Gotay 1981, Hanson *et al* 1976, Sundermeyer 1982).

It is apparent that these two classes of initial-value constraints play fundamentally distinct roles in the canonical analysis. Moreover, by introducing Dirac brackets, one can eliminate the second-class constraints altogether, leaving a purely first-class system (Dirac 1964, Gotay 1981, Hanson *et al* 1976, Sundermeyer 1982). This effective decoupling of the first- and second-class constraints in the canonical formalism suggests that there may be a dynamic segregation between these constraints *in the constraint algorithm itself.*

To make this notion precise, recall that in the Dirac constraint theory¶ one begins with certain primary constraints and that each additional constraint arises from the requirement that some prior constraint be preserved in time. Now we ask: *Does the preservation of a first-class constraint necessarily result in another first-class constraint? Conversely, does every first-class constraint arise from the requirement that some other first-class constraint be preserved? Is the same true for second-class constraints?*. If so, the constraint algorithm would enforce a sort of '*apartheid*' between the two classes of constraints: constraints of a given class would generate, and could only be generated by, constraints of that same class.

This strict decoupling, if found to be always the case, would obviously yield a substantial simplification in the canonical analysis of degenerate systems. Actually,

† Supported in part by the Natural Sciences and Engineering Research Council of Canada.

* Present address: Dept. Mathematics, US Naval Academy, Annapolis, Maryland 21402, USA.

¶ The Dirac constraint theory and its applications are discussed in Dirac (1964), Gotay *et al* (1978); Hanson *et al* (1976) and Sundermeyer (1982).

‖ A constraint is said to be *first class* if its Poisson bracket with any other constraint weakly vanishes, and *second class* otherwise.

0305-4470/84/153063 +04$02.25

apartheid seems to have become part of constraint theory folklore, even though it does not appear to have been explicitly discussed anywhere. Regardless, we know of no instance in the literature in which the constraint algorithm mixes first- and second-class constraints.

In this note we settle the apartheid issue, finding that constraints may indeed cross 'class lines' in the constraint algorithm. Specifically, we prove that the preservation of a first-class constraint can only lead to another first-class constraint, but exhibit counterexamples which show that this need not be true for second-class constraints. We then briefly discuss some of the physical implications of our results.

We begin with apartheid on the first-class level. Let ϕ be a first-class constraint and denote by H_T Dirac's 'total' Hamiltonian. Demanding that ϕ be preserved in time leads to the derived constraint

$$\dot{\phi} = \{\phi, H_T\} \simeq 0,$$

where $\{\cdot, \cdot\}$ is the Poisson bracket and $\simeq$ means 'weak' equality. We show that the constraint $\dot{\phi} \simeq 0$, if non-trivial, is again first class. For any constraint χ, the Jacobi identity yields

$$\{\{\phi, H_T\}, \chi\} = \{\{\chi, H_T\}, \phi\} - \{\{\chi, \phi\}, H_T\}.$$

Since χ is a constraint so is $\{\chi, H_T\}$; the first term on the RHS then weakly vanishes by the assumption on ϕ. Similarly, as ϕ is first class $\{\chi, \phi\} \simeq 0$ and the second term on the RHS must then weakly vanish by the constraint algorithm. Thus $\{\{\phi, H_T\}, \chi\} \simeq 0$ for all constraints χ, i.e., $\{\phi, H_T\}$ is first class.

This answers our first question in the affirmative, perhaps not too surprisingly (in fact, this result is implicit in the work of Dirac (1964)). What *is* interesting is that the converse of this result fails.

We illustrate this via the Lagrangian

$$L = \tfrac{1}{2} w^{-2} \dot{x}^2 - \tfrac{1}{2} w^2 - \tfrac{1}{3}(z^3 - y^3) + (y + z)x.$$

Upon going over to the Hamiltonian formulation we find three primary constraints $p_y \simeq 0$, $p_z \simeq 0$ and $p_w \simeq 0$. The total Hamiltonian is then

$$H_T = \tfrac{1}{2} w^2 (p_x^2 + 1) + \tfrac{1}{3}(z^3 - y^3) - (y + z)x + v_y p_y + v_z p_z + v_w p_w,$$

where v_y, v_z and v_w are Lagrange multipliers. The constraint algorithm yields three secondary constraints

$$\{p_y, H_T\} = x + y^2 \simeq 0$$

$$\{p_z, H_T\} = x - z^2 \simeq 0$$

$$\{p_w, H_T\} = -w(p_x^2 + 1) \simeq 0$$

and subsequently forces v_y, v_z and v_w to vanish. The equations of motion are all trivial.

A functionally independent set of constraints for this system consists of one first-class constraint $x \simeq 0$ and six second-class constraints

$$p_y \simeq 0, \qquad y \simeq 0, \qquad p_z \simeq 0, \qquad z \simeq 0, \qquad p_w \simeq 0, \qquad w \simeq 0.$$

It follows that the two secondary constraints $x + y^2 \simeq 0$ and $x - z^2 \simeq 0$ produced directly by the constraint algorithm are first class even though they are generated by the second-class primaries $p_y \simeq 0$ and $p_z \simeq 0$, respectively. In this system there is clearly no way to disentangle the two classes of constraints within the constraint algorithm itself. (Indeed, there are no first-class primaries with which to generate these two first-class secondaries).

This example shows that it is quite possible for a second-class constraint to generate a non-trivial first-class constraint. On the other hand, we have seen that the preservation of a first-class constraint can only yield another first-class constraint. This does not mean, however, that second-class constraints cannot *ultimately* arise from the preservation of first-class constraints.

For, consider the Lagrangian

$$L = \tfrac{1}{2}\dot{x}^2 + yx^2.$$

In this case the first-class primary $p_y \simeq 0$ leads directly to the first-class secondary $x^2 \simeq 0$, in accordance with our findings above. But $x^2 \simeq 0$ iff $x \simeq 0$ which in turn generates $p_x \simeq 0$. Thus the two second-class constraints $x \simeq 0$ and $p_x \simeq 0$ are ultimately derived from the first-class constraint $p_y \simeq 0$. The catch is that the first-class constraint $x^2 \simeq 0$ is really the second-class constraint $x \simeq 0$ 'in disguise'. More precisely, the pathology in this example is due to the presence of the *ineffective* first-class constraint $x^2 \simeq 0$†.

Ineffective first-class constraints can arise as the result of preserving either first- or second-class constraints and can be 'resolved' into effective constraints of either type. Moreover, the appearance of ineffective first-class constraints seems to be closely linked to the failure of apartheid. In all the systems we have been able to construct in which apartheid fails (or which exhibit the bizarre behaviour of the second example), there is such a constraint lurking *somewhere*‡. This is the case even in our first example: although the two first-class constraints $x + y^2 \simeq 0$ and $x - z^2 \simeq 0$ are themselves effective, their first-class difference $y^2 + z^2 \simeq 0$ is not. These observations lead us to conjecture that *apartheid is valid for systems with no naturally occurring ineffective constraints*§.

A physically interesting system which displays many of these pathologies is given by the Lagrangian

$$L = -\tfrac{1}{4}F_{\mu\nu}F^{\mu\nu} + \tfrac{1}{2}\lambda^{-1}(\partial_\mu A^\mu)^2 - \tfrac{1}{3}\lambda^3.$$

This field theory has one second-class primary which generates an ineffective first-class secondary (equivalent to two functionally independent constraints, one first class and the other second class), which in turn gives rise to a first-class tertiary. In this instance, the failure of apartheid (note also the presence of an ineffective constraint) has a direct bearing on the physical interpretation of this Lagrangian: Is this system just Lorentz-gauged electromagnetism or, rather, something completely different (e.g., a massless, divergence-free, spin-1 field)?

Apartheid is therefore important for understanding the extent to which first-class constraints generate gauge transformations. It is a matter of 'pedigree'. Every member of a chain of effective first-class constraints derived from the preservation of a primary first-class constraint has unimpeachable credentials as a generator of gauge transformations (Gotay 1983, Gotay and Nester 1979). On the other hand, a first-class constraint which is derived from a second-class constraint is very suspect as a gauge generator.

† A constraint ϕ is *effective* if $d\phi \neq 0$ and *ineffective* otherwise.

‡ However, there do exist purely first-class systems in which ineffective constraints appear and for which apartheid *is* valid (e.g., remove the last term from our field theory Lagrangian).

§ In view of the fundamental importance of ineffective constraints in the canonical formalism (cf. also Gotay 1983, Gotay and Nester 1979), one might well distinguish them with the title 'third-class' constraints.

Finally, we present a rather surprising corollary of our analysis. Suppose that one was somehow able to *a priori* determine all the second-class constraints in a given system. Then it will *not* always be possible to *first* eliminate the corresponding set of non-dynamical variables and *then* 'restart' the constraint algorithm (using the Dirac bracket in place of the Poisson bracket), thereby recovering the 'missing' first-class constraints. The problem is that one may lose first-class constraints in the process; in particular, one would lose the first-class secondary $x \simeq 0$ altogether in the first example†. In general, therefore, one can only eliminate the second-class constraints from the formalism *after* the conclusion of the constraint algorithm. This is a reflection of our finding above that the first- and second-class constraints decouple only after the constraint algorithm has been carried to completion. Thus non-dynamical variables play an indispensable role in obtaining the field equations and constraints despite the fact that such variables are physically irrelevant.

To summarise: our examples quash all hopes for apartheid in general. We conclude that the constraint algorithm itself inextricably intertwines the first- and second-class aspects of the canonical formalism and that the failure of apartheid has far-reaching physical consequences.

References

Dirac P A M 1964 *Lectures on Quantum Mechanics* (New York: Academic)
Gotay M J 1981 in *Geometric Quantization* ed M J Gotay, University of Calgary
—— 1983 *J. Phys. A: Math. Gen.* **16** L141
Gotay M J and Nester J M 1979 in *Lecture Notes in Physics* **94** (Berlin: Springer) p 272
Gotay M J, Nester J M and Hinds G 1978 *J. Math. Phys.* **19** 2388
Hanson A, Regge T and Teitelboim C 1976 *Accad. Naz. dei Lincei* **22** 1976
Sundermeyer K 1982 *Constrained Dynamics, Lecture Notes in Physics* **169** (Berlin: Springer)

† The difficulty in this instance, as alluded to previously, is that there are no effective first-class *primaries* with which to 'restart' the constraint algorithm.