Dynamical pair approximation for cellular automata with shuffle update

2007 J. Phys. A: Math. Theor. 40 2651

(http://iopscience.iop.org/1751-8121/40/11/004)

IOP PUBLISHING JOURNAL OF PHYSICS A: MATHEMATICAL AND THEORETICAL

J. Phys. A: Math. Theor. **40** (2007) 2651–2664 doi:10.1088/1751-8113/40/11/004

Dynamical pair approximation for cellular automata with shuffle update

David A Smith and R Eddie Wilson

Bristol Centre for Applied Nonlinear Mathematics, Department of Engineering Mathematics, University of Bristol, Queen's Building, University Walk, Bristol BS8 1TR, UK

E-mail: D.A.Smith@bristol.ac.uk and RE.Wilson@bristol.ac.uk

Received 25 August 2006, in final form 29 December 2006
Published 28 February 2007
Online at stacks.iop.org/JPhysA/40/2651

Abstract

The random shuffle update method for the asymmetric exclusion process (ASEP) is introduced, and the dynamical pair technique is extended in order to analyse its dynamics. A sequence of approximate models is introduced, the first element of which corresponds to the classical parallel update rule whose pair dynamics is reviewed. It is then shown how the argument may be extended inductively to solve for the two-cell configuration probabilities for each element of the sequence of approximate models. A formal limit is then taken, and macroscopic velocities and flow rates are derived.

PACS numbers: 89.40.−a, 02.50.−r

1. Introduction

The general context of this paper is the modelling of unidirectional road traffic or pedestrian flow with one-dimensional cellular automata of Nagel–Schreckenberg type [1], with the maximum velocity parameter $v_{\max}$ set equal to 1. This type of model is sometimes referred to as the asymmetric exclusion process (ASEP) [2, 3]. In this well-known set-up, space is discretized into a one-dimensional array of cells each of which is either empty or occupied by exactly one agent, and each agent moves according to a pair of very simple microscopic rules:

(i) if the cell immediately downstream is occupied, remain stationary (R1), and
(ii) if the cell downstream is unoccupied, move forward into it with probability p, $0 < p \leqslant 1$. (R2)

The only remaining subtlety (and the subject of this paper) concerns the precise order in which rules (R1) and (R2) are applied.

At each time step in the *parallel update* scheme [1, 4], rules (R1) and (R2) are applied simultaneously to all agents. No conflict resolution is necessary, since rule (R1) automatically prevents multiple occupancy.

In contrast, at each time step of the *random sequential update* scheme [5, 6], rules (R1) and (R2) are applied to a single agent chosen at random. For simulation purposes, the most attractive feature of the random sequential update is that single-occupancy is automatically preserved by (R1), and this would hold even in multi-dimensional situations, where the parallel update scheme would need a conflict resolution algorithm. However, a disadvantage of random sequential update is that with small probability, a single agent might receive a large number of consecutive turns, and thus, in low density situations, achieve an unphysical velocity.

This paper however is concerned with the dynamics of rules (R1) and (R2) under the *shuffle update* scheme, which has received very little attention in the literature to date [7–10]. At each time step in this scheme, rules (R1) and (R2) are applied to each individual agent in turn, according to a random order generated at the beginning of the time step, which contains each agent exactly once. After all agents have applied rules (R1) and (R2), a new random order is generated and the next time step begins. Consequently, the scheme is stochastic even when $p = 1$, because of the random turn-taking order.

The shuffle update is similar to the random sequential scheme in that the occupancy of cells is updated incrementally as each agent applies its rules and consequently, it does not require conflict resolution to preserve single-occupancy (even in multi-dimensional extensions). However, the shuffle update enjoys the modelling advantage that the velocities of individual agents are bounded.

The chief result of this paper is an extension of the *dynamical pair* analysis of Schreckenberg *et al* [11], which analyses (R1) and (R2) under the parallel update scheme, to the more complicated case of the shuffle update. The argument here is more involved than [11] because under the shuffle update, it is possible for large blocks of contiguous agents to move forward in a single time step, if their turns are served in upstream order. Consequently, we analyse a sequence of approximate processes defined as follows and illustrated in figure 1.

Definition. By the *truncated process of order n*, we mean that rules (R1) and (R2) are applied under the shuffle update scheme, with the proviso that the opportunity to move is offered only to agents who are in the first n positions of a contiguous block at the beginning of the time step.

Note that the truncated process of order 1 is identical to the parallel update scheme. However, the limit of interest is that of truncated processes of order $n \to \infty$, under which one converges to the dynamics of the (full, untruncated) shuffle update scheme.

The remainder of the paper is laid out as follows. In section 2, we describe dynamical pair analysis, and we solve for the statistically stationary two-cell configuration probability for the two coarsest approximate schemes $n = 1, 2$. Sections 3 and 4 then extend the argument inductively, to solve for the two-cell configuration probability in a sequence of truncated models which approximate the full scheme as close as we like. A formal limit is taken, then section 5 derives quantities such as the distribution of block lengths, average velocity and average flow, as a function of the system density. Throughout we assume (i) that the system has reached statistical stationarity, (ii) that it is large, and (iii) that periodic boundary conditions are in force so that agents leaving the right-hand end of the system re-join on the left, so that the total number of agents, and therefore the density, is conserved.

Note that the analysis presented here is approximate, since the dynamical pair analysis is based on a spatial independence approximation: namely that the configuration probabilities for three- and four-cell clusters and so on factorize into products of conditional two-cell configuration probabilities. This limitation and possible extensions to the theory are discussed further in section 6.

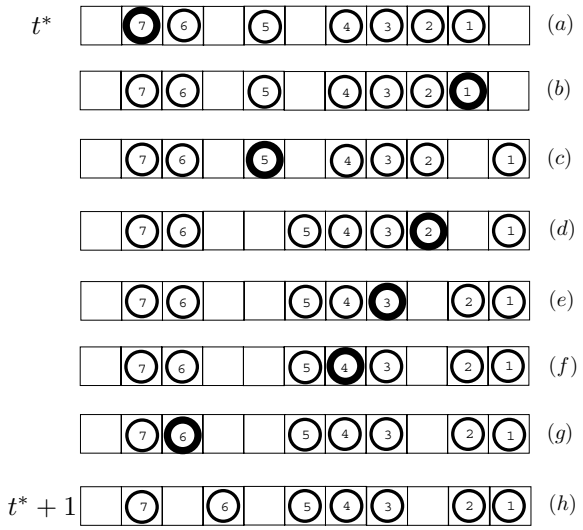


Figure 1. One time step of the truncated process with $n = 2$, $p = 1$ and (randomly generated) turn-taking order 7, 1, 5, 2, 3, 4, 6. The sequence of moves is as follows: (*a*) agent 7 is blocked; (*b*) agent 1 moves; (*c*) agent 5 moves; (*d*) agent 2 moves; (*e*) agent 3 does not move because of the truncation (position 3 of its block at time t^*); (*f*) agent 4 is blocked; (*g*) agent 6 moves. Note that only the front two agents of block 1–4 move whereas in the full shuffle process they would all move.

2. Two-cell cluster analysis

In order to compute distributions of block sizes and hence mean velocities and flow rates, we generalize the dynamical pair analysis of Schreckenberg *et al* [11]. The idea is to compute the so-called *two-cell configuration* probabilities P_2 for the possible states $(1, 0)$, $(0, 1)$, $(1, 1)$ and $(0, 0)$ of two adjacent cells, where 0 and 1 denote empty and occupied respectively. It may be shown that the two-cell configuration probabilities are related by $P_2(1, 0) = P_2(0, 1)$, $(=: y)$, $P_2(1, 1) = c - y$, and $P_2(0, 0) = 1 - c - y$, where c denotes the prescribed mean density, i.e., the probability that a single cell is occupied. Consequently, we need only to find y.

2.1. Truncated process with $n = 1$

To simplify matters, we begin by reviewing the $n = 1$ truncated process, which is identical to the parallel update scheme [1]. The key step is to consider a window of cells surrounding a pair whose time evolution is monitored and to catalogue all moves which result in that pair becoming $(1, 0)$: see figure 2. Here we label the cells 0, 1, 2, 3 in downstream order, and we denote their contents by $\tau_0, \tau_1, \tau_2, \tau_3$ at the backward time step $t = t^* - 1$ and by $\sigma_0, \sigma_1, \sigma_2, \sigma_3$ at $t = t^*$. Note that if a cell's occupancy is denoted by **?**, then all moves carry through with the same transition probabilities irrespective of whether that cell is occupied or not.

The first task is to use the rules of the cellular automata to compute the transition probabilities W for each of the left-hand to right-hand column moves listed in figure 2. Take for example $F_1^{(1)}$, which gives the desired right-hand state if the occupant of cell 1 does not move; thus $W\left(F_1^{(1)}\right) = 1 - p$ from rule (R2). Alternatively, consider $G_2^{(1),1}$. The agents in cells 0 and 2 must both move (each with probability p), so $W\left(G_2^{(1),1}\right) = p^2$. Other calculations are similar.

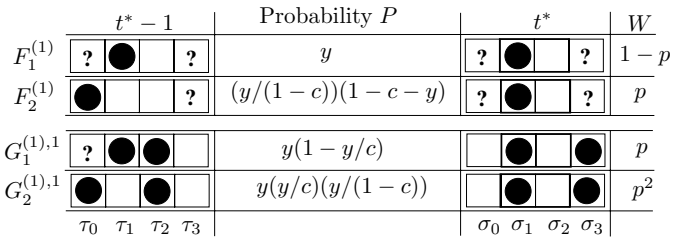


Figure 2. The list of all possible transitions to a $(\sigma_1, \sigma_2) = (1, 0)$ two-cell cluster (highlighted in bold) at time step t^*, for the $n = 1$ truncated process. Configuration probabilities for the left-hand column states are denoted by P, and transition probabilities by W. Cells marked with a **?** symbol can be occupied or empty, with no effect on either P or W.

The second task is to approximate the stationary probability of each left-hand column state in terms of the two-cell configuration probability y and the mean density c. To achieve this, we use the conditional two-cell cluster probabilities $P_2(\underline{1}, 1) = P_2(1, \underline{1}) = 1 - y/c$, $P_2(\underline{0}, 0) = P_2(0, \underline{0}) = 1 - y/(1-c)$, $P_2(\underline{1}, 0) = P_2(0, \underline{1}) = y/c$, and $P_2(\underline{0}, 1) = P_2(1, \underline{0}) = y/(1-c)$. Here we have adopted the notation of [11], so that e.g., $P_2(\underline{1}, 0)$ denotes the probability of a cell being unoccupied if its left-hand neighbour is occupied. We then apply local structure theory [12] to approximate the probabilities of larger clusters in terms of smaller ones. For example

$$\begin{aligned} P\big(G_2^{(1),1}\big) &= P(\tau_0 = 1, \tau_1 = 0, \tau_2 = 1, \tau_3 = 0), \qquad =: P_4(1, 0, 1, 0), \\ &\simeq P_2(1, \underline{0})P_2(0, \underline{1})P_2(1, 0), \\ &= \frac{y^3}{c(1-c)}. \end{aligned} \tag{1}$$

The remaining state probabilities are approximated similarly. In what follows we suppose that the above factorization is exact, and we subsequently drop the $\simeq$ symbol.

We may now write down the probability $y = P_2(1, 0)$ in terms of the sum product of the probabilities of left-hand column states and their corresponding transition probabilities. We have

$$y := \sum_{i=1,2} \big(W\big(F_i^{(1)}\big)P\big(F_i^{(1)}\big) + W\big(G_i^{(1),1}\big)P\big(G_i^{(1),1}\big)\big), \tag{2}$$

which on substitution of the quantities listed in figure 2 yields

$$\left(\frac{p^2}{c(1-c)}\right)y^2 - \left(\frac{p}{c(1-c)}\right)y + p = 0. \tag{3}$$

This quadratic has one valid root between 0 and 1 given by

$$y = \frac{1}{2p}\big(1 - \sqrt{1 - 4pc(1-c)}\,\big), \tag{4}$$

see [11].

2.2. Truncated process with $n = 2$

We now deal with the $n = 2$ truncated process. Figure 3 lists five wide windows at $t = t^* - 1$ which give rise to $(\sigma_1, \sigma_2) = (1, 0)$ at $t = t^*$. There are six such states to consider, labelled $F_{1,2}^{(2)}$ and $G_{1,2}^{(2),1,2}$. Here the parenthesized superscript denotes the order n of the truncated process, and the subscripts and letters F and G describe a hereditary relation with the states

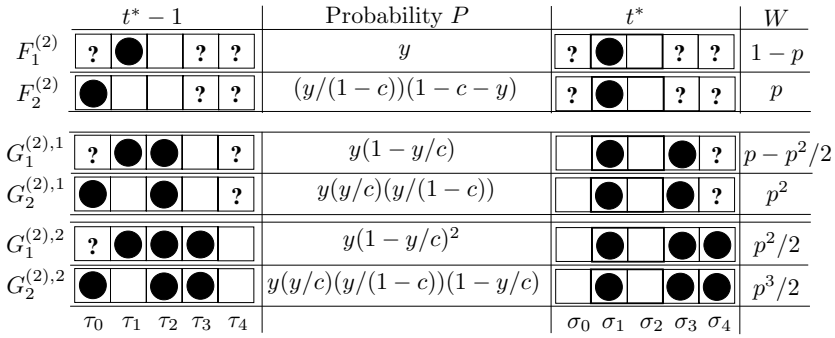


Figure 3. The list of all possible transitions to a $(1, 0)$ two-cell cluster at time step t^* for the $n = 2$ truncated process, c.f. figure 2. P and W denote respectively configuration probabilities for the left-hand column states and transition probabilities.

for $n = 1$. Note that the G states also have a second superscript whose meaning we explain shortly. In some sense, we may think of the $n = 2$ states as descendants of those for $n = 1$, and the crux is to understand how the 'propagation' works.

Firstly, note that the propagation of the F family of states is trivial. To see this, note that state $F_1^{(1)}$ has $(\tau_1, \tau_2) = (1, 0)$: consequently, it produces $(\sigma_1, \sigma_2) = (1, 0)$ via the occupant of cell 1 not moving. Likewise, state $F_2^{(1)}$ has $(\tau_0, \tau_1, \tau_2) = (1, 0, 0)$: so we require only the agent in cell 0 to move forward. Since the occupant of cell 0 is either an isolated agent, or the head of a block, the nature and probability of this transition is unaffected by the order n of the truncated process.

The difference between the $n = 1$ and $n = 2$ cases concerns the G family of states. Each of these has $\tau_2 = 1$, and hence requires the occupant of cell 2 to move to produce $\sigma_2 = 0$. In the $n = 1$ case, this can only happen if the occupant of cell 2 is either the head of a block (case $G_1^{(1),1}$) or isolated (case $G_2^{(1),1}$). When $n = 2$, there are more exotic possibilities. For example, we could have $(\tau_2, \tau_3) = (1, 1)$, and provided the occupant of cell 3 is the head of its block, it is possible for both the occupants of cells 2 and 3 to move, leaving $\sigma_2 = 0$. To determine whether the occupant of cell 3 is the head of its block, it is thus necessary to consider the occupancy of the next cell downstream, and this is why we must now consider five wide windows of cells.

Next we must compute the transition probabilities W: results are summarized in figure 3. The details are more complicated than for $n = 1$, since now the second agent in a block may move, provided the block leader also moves and provided its turn is served after the block leader (the latter occurring with probability $1/2$). Consequently, the different actions of the second agent have probabilities consisting of factors of $1/2$, p and $(1 - p)$.

For states $F_i^{(2)}$, we have $W\big(F_i^{(2)}\big) = W\big(F_i^{(1)}\big)$ because there is no second agent in a block to come in to play. We also have $W\big(G_2^{(2),1}\big) = W\big(G_2^{(1),1}\big)$ since there are no blocks of length greater than 1. However, the transition probabilities for other $G_i^{(2),m}$ states change. For $G_1^{(2),1}$, we have the first agent in the block moving but not the second: this can occur either (i) because the turn-taking order suffices but the second agent chooses not to move, or (ii) because the second agent is served before the first and hence is unable to move; thus we obtain $W\big(G_1^{(2),1}\big) = p(1 - p)/2 + p/2, = p - p^2/2$. The transition probabilities for the $G_i^{(2),2}$ states are related to those for $G_i^{(1),1}$ by $W\big(G_i^{(2),2}\big) = (p/2)W\big(G_i^{(1),1}\big)$, with the factor $p/2$ describing the probability that the second agent of a block is served after the first and it chooses to move.

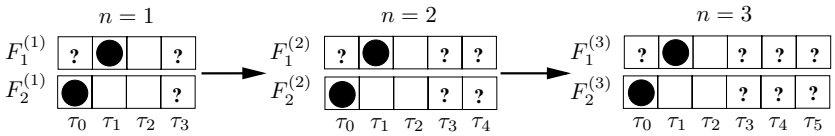


Figure 4. The propagation of $F_i^{(n)}$ states for increasing n, cf the left-hand column states in figures 2 and 3. The added right-hand cell in each window may be either occupied or empty with no effect on the transition to $(\sigma_1, \sigma_2) = (1, 0)$.

We now find how the state probabilities relate to those for $n = 1$ (see figures 2, 3). Here we consider five wide cell windows, since we need to consider $v_{\max} = 1$ cells upstream, and n cells downstream of our monitored two-cell cluster. The top four states in the left-hand column of figure 3 can be easily identified with the $n = 1$ states (figure 2), since the fifth cell takes the **?** state, contributing a factor of 1 to the probability. So we have $P\big(F_i^{(2)}\big) = P\big(F_i^{(1)}\big)$ and $P\big(G_i^{(2),1}\big) = P\big(G_i^{(1),1}\big)$. We then relate the remaining two $n = 2$ states $G_i^{(2),2}$ to the $n = 1$ states $G_i^{(1),1}$, by inserting an occupied cell in the fifth position, giving a contribution to the probability of $P_2(\underline{1}, 1) = (1 - y/c)$. So we have $P\big(G_i^{(2),2}\big) = (1 - y/c)P\big(G_i^{(1),1}\big)$.

Analogous to (2) we may write

$$y = \sum_i W\big(F_i^{(2)}\big)P\big(F_i^{(2)}\big) + \sum_{m\leqslant 2}\sum_i W\big(G_i^{(2),m}\big)P\big(G_i^{(2),m}\big), \tag{5}$$

which on substitution and simplification yields $f_2(y; c, p) = 0$, where

$$f_2(y; c, p) = p - y\left(\frac{p}{c(1-c)} + \frac{p^2}{2c}\right) + y^2\left(\frac{p^2}{c(1-c)} + \frac{p^2}{2c^2} + \frac{p^3}{2c(1-c)}\right) - y^3\left(\frac{p^3}{2c^2(1-c)}\right), \tag{6}$$

which we need to solve for y between 0 and 1.

3. Inductive construction for truncated processes

We now consider how to build the pair-dynamics argument inductively to analyse the truncated process for any finite order n. At each level, we will consider a catalogue of $2n + 2$ states each of which is $n + 3$ cells wide, and this will lead to a polynomial in y of degree $n + 1$. The key to this argument is to understand how the $F_i^{(n)}$ and $G_i^{(n),m}$ states 'breed' as n is increased.

As was seen for the $n = 1, 2$ cases, the states denoted by $F_i^{(n)}$ propagate unaltered as n increases (see figure 4). The cell window widens by one cell (on the right) at each step in n, and this cell may take either $\tau_{n+2} = 0$ or 1, with no effect on the calculations: this effect is indicated on the figures by **?**. We thus have $P\big(F_i^{(n)}\big) = P\big(F_i^{(1)}\big)$ for all n. Similarly, we find $W\big(F_i^{(n)}\big) = W\big(F_i^{(1)}\big)$ for all n because we need never consider the motion of more than one agent. In the $F_i^{(n)}$ states, we require that the front agent of a block does not move, so that all other motion is blocked and does not depend on the order n. Agents downstream have no effect on the monitored (σ_1, σ_2) two-cell cluster. In the $F_2^{(n)}$ states, the only motion is the front agent of a block moving into σ_1, and it does not matter how many agents move behind this one, as they do not enter the monitored (σ_1, σ_2) two-cell cluster. Again, agents downstream have no effect on the monitored (σ_1, σ_2) two-cell cluster.

The $G_i^{(n),m}$ states are more interesting in their propagation in n. They breed new states as well as propagating themselves (see figure 5). We define those states which, in figure 5, have

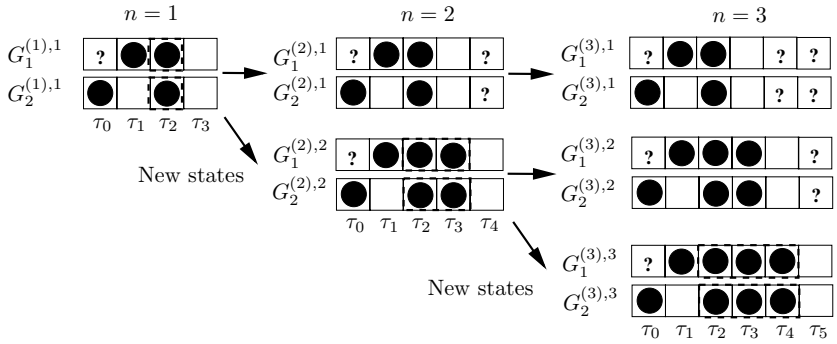
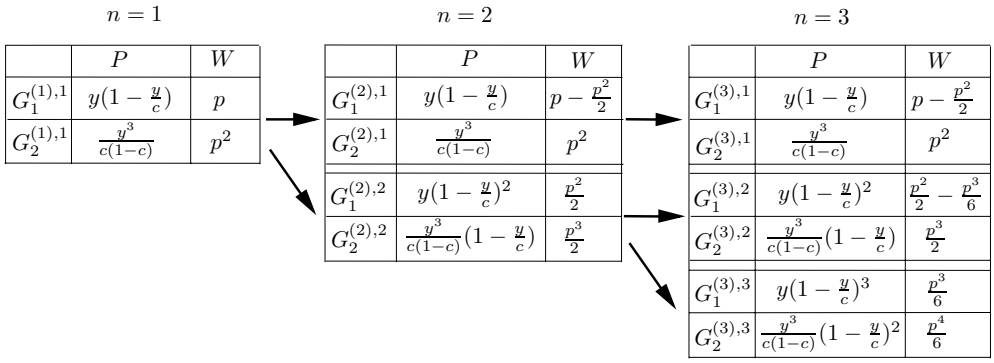


Figure 5. Propagation of $G_i^{(n),m}$ states for increasing n. States propagate in the same manner as $F_i^{(n)}$ states (shown in figure 4) as well as breeding new states. The characteristic feature of states which breed is that all cells with the dashed outline are filled. For any given n, $F_i^{(n)}$ and $G_i^{(n),m}$ encompass all states that produce $(\sigma_1, \sigma_2) = (1, 0)$ at the next time step.



	P	W
$G_1^{(1),1}$	$y(1-\frac{y}{c})$	p
$G_2^{(1),1}$	$\frac{y^3}{c(1-c)}$	p^2

	P	W
$G_1^{(2),1}$	$y(1-\frac{y}{c})$	$p-\frac{p^2}{2}$
$G_2^{(2),1}$	$\frac{y^3}{c(1-c)}$	p^2
$G_1^{(2),2}$	$y(1-\frac{y}{c})^2$	$\frac{p^2}{2}$
$G_2^{(2),2}$	$\frac{y^3}{c(1-c)}(1-\frac{y}{c})$	$\frac{p^3}{2}$

	P	W
$G_1^{(3),1}$	$y(1-\frac{y}{c})$	$p-\frac{p^2}{2}$
$G_2^{(3),1}$	$\frac{y^3}{c(1-c)}$	p^2
$G_1^{(3),2}$	$y(1-\frac{y}{c})^2$	$\frac{p^2}{2}-\frac{p^3}{6}$
$G_2^{(3),2}$	$\frac{y^3}{c(1-c)}(1-\frac{y}{c})$	$\frac{p^3}{2}$
$G_1^{(3),3}$	$y(1-\frac{y}{c})^3$	$\frac{p^3}{6}$
$G_2^{(3),3}$	$\frac{y^3}{c(1-c)}(1-\frac{y}{c})^2$	$\frac{p^4}{6}$

Figure 6. State P and transition W probabilities for $G_i^{(n),m}$ states. P is unaltered along propagating (horizontal) arrows, but gains a factor $P_2(\underline{1}, 1) = 1 - y/c$ on the breeding (diagonal) arrows. Transition probabilities gain a factor of p/n on diagonal arrows, while on horizontal arrows $W(G_1^{(n),n-1}) = W(G_1^{(n-1),n-1})(1 - p/n)$ with other quantities unchanged.

arrows linking the n state to two $n + 1$ states as 'breeding' states, while those with only one (horizontal) linking arrow we denote 'dormant' states. For clarity, we refer to extension along the horizontal arrows as propagation, and extension along the diagonal arrows as breeding. We now define the properties that make up these breeding and dormant states, before going on to discuss how their transition probabilities W to the monitored (1, 0) two-cell cluster, and their window state probabilities P, change with increasing n.

To explain the inductive process, we use figures 5 and 6. In the $G_i^{(n),m}$ notation, the n, m specify the horizontal and vertical coordinates of the corresponding pair of G states measured from the top-left-hand corner. The superscript m also gives the number of agents which are required to move from a single block in the corresponding window states. For instance, for $G_1^{(3),2}$ to have $(\sigma_1, \sigma_2) = (1, 0)$ at the next time step, two agents must move from the block, and the third must remain stationary.

Breeding states are those labelled $G_i^{(n),n}$, and are characterized by having $\tau_2, \ldots, \tau_{n+2} = 1$ (indicated on figure 5 by the dashed outlines). Thus n agents are required to move from a

block in order that $\sigma_2 = 0$ at $t = t^*$. We say that these states 'breed' because $G_i^{(n),n}$ can be related to both $G_i^{(n+1),n}$ and $G_i^{(n+1),n+1}$. By examining figure 5, we see that the $G_i^{(n+1),n}$ state is obtained along a horizontal arrow by the window state growing in the manner of the F states, with an added right-hand **?**. The $G_i^{(n+1),n+1}$ state manifests itself by a diagonal arrow, with the block length growing accordingly.

These breeding states account for all changes in probabilities W and P as n increases, see figure 6. We see that along the diagonal arrows, state probabilities P gain a factor of $P_2(\underline{1}, 1) = 1 - y/c$, from the increased length of the block, while transition probabilities W gain a factor of p/n from the extra agent that is being required to move. On the horizontal lines from breeding states, we see that the window state probabilities are unchanged, while the transition probabilities only alter for the $G_1^{(n),m}$ states. Here we have $W\big(G_1^{(n),m}\big) = W\big(G_1^{(n-1),m}\big) - (p/n)W\big(G_1^{(n-1),m}\big)$, which can be interpreted as Prob ($n - 1$ agents move)− Prob (n agents move). We have this relation because with the increase in n, it becomes necessary to specify that the last agent in the block does not move.

States labelled $G_i^{(n),m}$ with $n \neq m$ are 'dormant' and propagate unaltered in the same manner as the $F_i^{(n)}$ states. Transition probabilities do not change, as the number of agents required to move is already less than n and so is independent of n as it increases.

We can then summarize the probabilities shown in figure 6 inductively. For the state probabilities, we have $P\big(G_i^{(n),m}\big) = P\big(G_i^{(n-1),m}\big)$ and $P\big(G_i^{(n),m}\big) = (1 - y/c)P\big(G_i^{(n-1),m-1}\big)$, and for the transition probabilities, we have $W\big(G_i^{(n),m}\big) = (p/n)W\big(G_i^{(n-1),m-1}\big)$; then (i) for $m < n$, $W\big(G_i^{(n),m}\big) = W\big(G_i^{(n-1),m}\big)$; and (ii) for $m = n$, $W\big(G_1^{(n),m}\big) = W\big(G_1^{(n-1),m}\big) - (p/n)W\big(G_1^{(n-1),m}\big)$ and $W\big(G_2^{(n),m}\big) = W\big(G_2^{(n-1),m}\big)$.

4. General solution for the two-cell configuration probability

We have now identified the structure and trend of all the terms required to construct the equation $f_n(y; c, p) = 0$ (to solve for y) for any truncation order n. The construction begins with

$$y = \sum_i P\big(F_i^{(1)}\big)W\big(F_i^{(1)}\big) + \sum_{i,m} P\big(G_i^{(n),m}\big)W\big(G_i^{(n),m}\big). \tag{7}$$

As n is increased some terms, corresponding to the $F_i^{(n)}$ states, remain the same. Further new terms appear, and accumulate, corresponding to the breeding and dormant $G_i^{(n),m}$ states. We therefore seek an iterative process in the form $y =$ base terms + dormant terms + new terms.

The base term comes from the contribution of the $F_i^{(1)}$ states, since this remains unaltered for all n. The contributing terms are therefore those we saw for $n = 1$ in section 2.1 in the form $\sum_{i=1}^{2} P\big(F_i^{(1)}\big)W\big(F_i^{(1)}\big)$. Thus we have

$$\text{base} = y - \frac{py^2}{1-c}. \tag{8}$$

As dormant states and breeding states all come from the $G_i^{(n),m}$ states, which all originate from $G_i^{(1),1}$ via the inductive development, we consider them together.

We start by writing down the terms contributed by the $G_i^{(1),1}$ states (which are classed as breeding states) and then build the inductive argument from there. These terms are of the form $\sum_{i=1}^{2} P\big(G_i^{(1),1}\big)W\big(G_i^{(1),1}\big)$. Thus

$$\text{breeding}_1 = y\left(1 - \frac{y}{c}\right)p + y\left(\frac{y}{c}\right)\left(\frac{y}{1-c}\right)p^2. \tag{9}$$

We note here that by solving $y = \text{base} + \text{breeding}_1$, we obtain equation (4) for the $n = 1$ case, as we would expect.

Now, to proceed with our inductive argument we recall that the breeding states are those labelled $G_i^{(n),n}$ and the dormant states are those labelled $G_i^{(n),m}$ with $m < n$. So new dormant states and new breeding states are produced by the respective propagation and breeding of $G_i^{(n-1),n-1}$ states. The first dormant states appear for $n = 2$ and are labelled $G_i^{(2),1}$ in figures 3 and 5. We examine the $G_i^{(2),m}$ contribution to the equation for y as an extension of the breeding_1 terms, and note which correspond to the dormant states, and which to the breeding states.

By applying the rules outlined in section 3, we have for $n = 2$,

$$y = \text{base} + \text{breeding}_1 \left(1 + \frac{p}{2}\left(1 - \frac{y}{c}\right)\right) - \left(\frac{p}{2}\right) P\left(G_1^{(1),1}\right) W\left(G_1^{(1),1}\right), \tag{10}$$

where

$$\text{dormant}_2 = \text{breeding}_1 - \left(\frac{p}{2}\right) P\left(G_1^{(1),1}\right) W\left(G_1^{(1),1}\right), \tag{11}$$

and

$$\text{breeding}_2 = \frac{p}{2}\left(1 - \frac{y}{c}\right) \text{breeding}_1. \tag{12}$$

By generalizing our extension from $n = 1$ to $n = 2$ and recalling that the dormant terms remain in our equation for all higher n, we can write down the complete equation

$$y = \text{base} + \sum_{j=1}^{n-1} \left(\text{breeding}_j - y \frac{p^{j+1}}{(j+1)!} \sum_{i=1}^{j+1} \binom{j}{i-1} \left(\frac{-y}{c}\right)^{i-1} \right) + \frac{p}{n}\left(1 - \frac{y}{c}\right) \text{breeding}_{n-1}. \tag{13}$$

Here the sum contains the terms from all the $2(n-1)$ dormant $G_i^{(n),m}$ states, and the last term is the contribution from the breeding $G_i^{(n),n}$ states. Furthermore, we have used the facts (see figure 6)

$$P\left(G_1^{(j),j}\right) = y\left(1 - \frac{y}{c}\right)^j \qquad \text{and} \qquad W\left(G_1^{(j),j}\right) = \frac{p^j}{j!}. \tag{14}$$

In order to fully express our equation for y, we need to write explicitly, and generally, the terms from breeding states. These, unlike the dormant state terms, do not accumulate. There is one pair of breeding states at each order n, labelled $G_i^{(n),n}$; the information on lower order breeding states is included in the dormant state terms.

Again we refer to the breeding rules described in section 3 and figures 5 and 6. We see that in the change from n to $n + 1$, the cell window has become one cell wider, and this extra cell is accounted for by an extra agent in the main block. Therefore, using the two-cell cluster method to write down state probabilities, the extra occupied cell manifests itself as the inclusion of an extra factor $P_2(1, \underline{1}) = (1 - y/c)$. The transition probabilities obey $W\left(G_i^{(n),n}\right) = (p/n) W\left(G_i^{(n-1),n-1}\right)$, since one more agent is required to move with probability p, and there is a $1/n$ probability that the agent is allowed to move due to the update order. We can then write any breeding_n terms by building inductively from the breeding_1 terms (9), as done for $n = 2$ in (12). We can say that for general n,

$$\begin{aligned} \text{breeding}_n &= y\left(1 - \frac{y}{c}\right)^n \frac{p^n}{n!} + y\left(\frac{y}{c}\right)\left(\frac{y}{1-c}\right)\left(1 - \frac{y}{c}\right)^{n-1} \frac{p^{n+1}}{n!}, \\ &= y\frac{p^n}{n!} \sum_{i=1}^{n} \binom{n-1}{i-1} \left(\frac{-y}{c}\right)^{i-1} \times \left(1 - \frac{y}{c} + \left(\frac{y}{c}\right)\left(\frac{py}{1-c}\right)\right). \end{aligned} \tag{15}$$

We now have all the necessary ingredients to write down our equation to solve for y in the form

$$y = \text{baseterms} + \text{dormantterms} + \text{breeding}_n. \tag{16}$$

We divide this by y, and rearrange to get $f_n(y; c, p) = 0$, which we then need to solve for y. We have

$$f_n(y; c, p) = p - \frac{py}{1-c} + \sum_{i=1}^{n} \left(\frac{-y}{c}\right)^i \sum_{j=i}^{n} \frac{p^j}{j!} \begin{pmatrix} j-1 \\ i-1 \end{pmatrix} \left(1 - \frac{py}{1-c}\right), \tag{17}$$

and we must solve $f_n(y; c, p) = 0$ for the two-cell configuration probability $y = P_2(1, 0)$ for the truncated process of order n. In fact, (17) has an $n \to \infty$ limit

$$f(y; c, p) = -(1-p) + \frac{1}{c-y}\left(1 - \frac{py}{1-c}\right)\left(c - y\,\mathrm{e}^{p(1-y/c)}\right), \tag{18}$$

and thus we solve $f(y; c, p) = 0$ for the $(1, 0)$ probability of the full untruncated process. This equation in general requires a numerical solution procedure. For robustness, we now show that $f(y; c, p) = 0$ has a unique solution for y in the physical range $[0, \min(c, 1-c)]$.

Firstly, we re-write (18) in terms of the new variable $\hat{y} = c - y \in [\max(0, 2c-1), c]$ to obtain $\hat{f}(\hat{y}; c, p) = -(1-p) + L(\hat{y})g(\hat{y})$, where

$$L(\hat{y}) = \left(\frac{1 - c - pc + p\hat{y}}{1-c}\right) \qquad \text{and} \qquad g(\hat{y}) = \sum_{j=0}^{\infty} \frac{1}{j!}\left(\frac{p}{c}\right)^j \left(1 - \frac{p}{j+1}\right)\hat{y}^j, \tag{19}$$

and the power series for g arises from expansion of the exponential term. We then note $\hat{f}(0) = -pc(1-p)/(1-c) < 0$ and $\hat{f}(c) = p > 0$ so that a sign change on the required interval establishes the existence of at least one solution in the required range. To prove uniqueness, we show that $\hat{f}$ is monotone on $[\max(0, 2c-1), c]$, by considering the derivative $\hat{f}'(\hat{y}) = L'(\hat{y})g(\hat{y}) + L(\hat{y})g'(\hat{y})$. We note that all coefficients in the power series of g are positive, so $g(\hat{y})$ and all its derivatives are positive. The proof of monotonicity is completed by noting that on the required range,

$$L(\hat{y}) \geqslant \frac{1 - c - pc + p(2c-1)}{1-c} = 1 - p, \qquad \text{and} \qquad L'(\hat{y}) = p/(1-c), \tag{20}$$

which are both positive quantities.

This procedure can also be used to show that the truncated models have unique solutions for the two-cell configuration probability y of their respective equation $f_n(y; c, p) = 0$ defined by (17).

5. Steady state velocities and flow rates

Once we have solved for the two-cell configuration probability y, we may use it to find macroscopic flow quantities. We begin by computing the probability distribution of contiguous block lengths using local cluster theory. For example, the probability $\mathcal{P}_2$ of a block of length 2 is the probability that the lead agent has exactly one agent on the left before the next space, i.e. $\mathcal{P}_2 = P_2(0, \underline{1})P_2(1, \underline{1}) = (y/c)(1 - y/c)$. This argument may be extended to give the block length distribution

$$\mathcal{P}_l = \left(\frac{y}{c}\right)\left(1 - \frac{y}{c}\right)^{l-1}. \tag{21}$$

Furthermore, note that the probability that an agent chosen at random is in a block of length l is given by $l\mathcal{P}_l / \sum_{i=1}^{\infty} i\mathcal{P}_i, = l(y/c)^2(1 - y/c)^{l-1}$, since it is weighted towards the larger

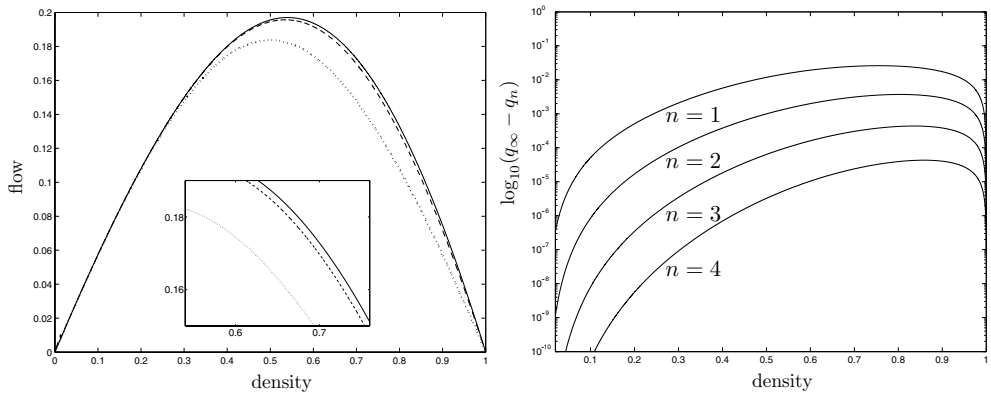


Figure 7. Comparison of flow-density relations arising from dynamical pair analysis of order n truncated models and the full shuffle update scheme. Here $p = 0.6$. The left-hand plot: $n = 1$ truncation ($\cdots\cdots$), $n = 2$ truncation (- - - -), and full model (——). The right-hand plot: the logarithmic error plot showing convergence trend, where q_∞ denotes the flow prediction of the full model, and q_n denotes the corresponding quantity for the order n truncation.

blocks. Note further that an agent chosen at random from a block of length l is equally likely to be in each position with probability $1/l$, and consequently the mean velocity $\hat{v}$, which is equal to the probability of motion in a given time step for a randomly chosen agent, is given by

$$\hat{v} = \sum_{l=1}^{\infty} l \left(\frac{y}{c}\right)^2 \left(1 - \frac{y}{c}\right)^{l-1} \sum_{k=1}^{l} \frac{1}{l} \frac{p^k}{k!}, \tag{22}$$

where $p^k/k!$ is the probability that at least k agents move from a given block. By interchanging the order of summation and recognizing the tail of a geometric series, we obtain

$$\hat{v} = \left(\frac{y}{c - y}\right)\left(\exp\left(\frac{p}{c}(c - y)\right) - 1\right), \tag{23}$$

from which me may compute the flow rate $q = c\hat{v}$. This result is equivalent to that found by Wölki *et al* [8], using a car-oriented mean field (COMF) method which solves for the quantity $P_0 := P_2(\underline{1}, 1) = 1 - y/c$. The paper [8] also graphs out formula (23) and validates its accuracy against numerical simulations.

In figure 7 we display the rapid convergence of the sequence of truncated models by plotting flow-density curves based on (23) and the pair-configuration probability y that is the zero of (17) for the truncated processes and (18) for the full process. However, all of these curves are approximate since they are based on analysis that relies upon exact factorization into two-cluster probabilities.

6. Conclusions and discussion

In this paper we have used the dynamical pair technique and a sequence of approximate models to analyse the asymmetric exclusion process under shuffle update rules. For future work it would be interesting to study the shuffle update model for $v_{\max} > 1$ and compare with the parallel and random sequential models. Furthermore, an extension into multi-lane traffic may be worth pursuing, since as we have seen here, the shuffle update scheme guarantees collision avoidance, even in higher dimensions.

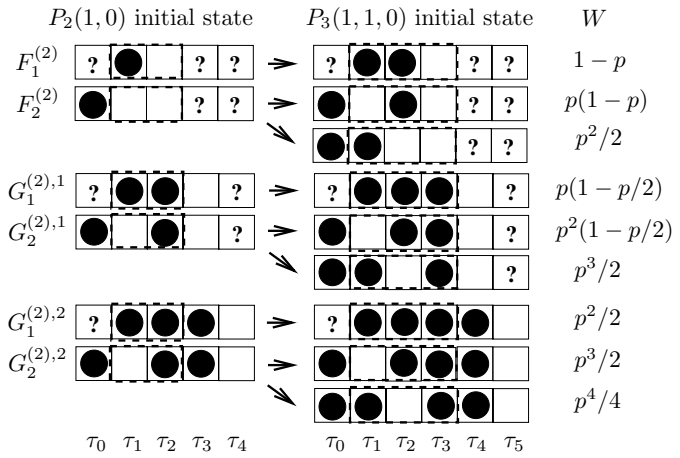


Figure 8. Generalization procedure for developing three-cell cluster analysis from figure 3. $F_1^{(n)}$ and $G_1^{(n),m}$ states are modified by inserting an occupied cell in position 2, whereas $F_2^{(n)}$ and $G_2^{(n),m}$ states extend by inserting an occupied cell in either position 0 or 2. W denotes the transition probability that the highlighted cells of the new states become (1,1,0) in a single time step.

During the preparation of this manuscript, a preprint by Wölki *et al* came to our attention (subsequently published [8]), which solves the same problem via a method which is oriented around mobile agents (i.e., car-oriented mean field COMF) rather than static pairs of cells. The key results (18), (23) in this paper are identically equivalent to those presented in [8], so further discussion should be focused on the relative merits of the two approaches.

Both the COMF and the site-oriented mean field (SOMF) approach adopted here have spatial independence approximations from local cluster theory [12] at their core. However, the macroscopic velocities and flow rates that the theory predicts are in close agreement with numerical simulations. Moreover, one may show that the results are exact for the order $n = 1$ truncation model (equivalent to parallel update [11]) and for $p = 1$ and $c \leqslant 1/2$ in the full shuffle update model [8]. In the latter case, agents separate into singletons for sufficiently large t, and subsequently move forward at $v_{\max} = 1$.

The SOMF method is relatively complicated compared to [8], but it has the advantage that we may probe the spatial independence assumption further by attempting an analysis of the dynamics of larger clusters. To cover this in detail would be the subject of another paper, but here we give a brief overview of how it may be achieved with three-cell clusters, for which there are eight possible states. However, by using symmetry and conservation principles, one may show that it is sufficient to analyse three independent three-cell configuration probabilities, namely $y_1 = P_3(1,0,0)$, $y_2 = P_3(0,1,0)$ and $y_3 = P_3(1,1,0)$. From these we may also define conditional three-cell configuration probabilities, as in [11, 12].

The tools are now in place to repeat the process from earlier in this paper, although now we should expect the details to be more complicated. Here we focus on the $n = 2$ truncated model for simplicity (the $n = 1$ model being trivial since then the dynamical pair analysis is exact). Now we must catalogue all of the ways in which the configurations $(1,0,0)$, $(0,1,0)$ and $(1,1,0)$ can be produced in a single time step, and consequently we must produce three figures each of which is analogous to figure 3. The chief observation is that the left-hand column states in these new pictures can be derived from figure 3 via a new hereditary process. For example, figure 8 lists states from which the $(1,1,0)$ three-cell cluster may be obtained in a single time step and how these states breed from the two-cell cluster analysis of figure 3. We

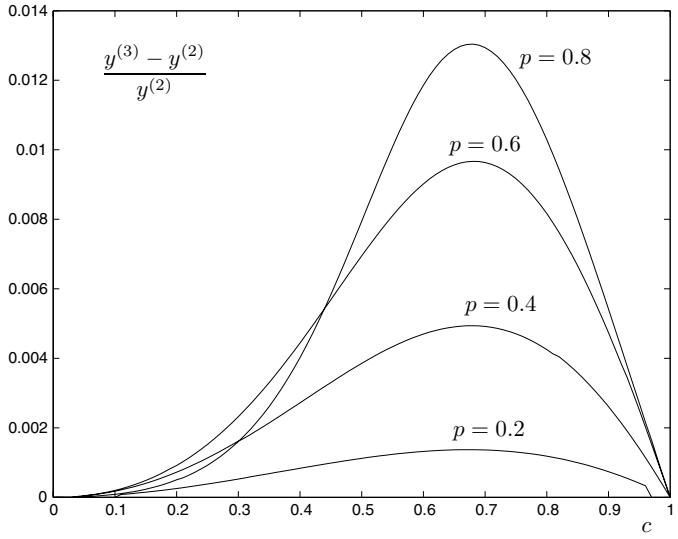


Figure 9. Relative comparison of the dynamic pair configuration probability $P_2(1, 0)$ arising from two-cluster theory $y^{(2)}$ and three-cluster theory $y^{(3)}$, as density c and movement probability p are varied.

have also obtained the pictures for (1, 0, 0) and (0, 1, 0) via similar genetic principles (details not presented here), and it is intriguing to speculate whether the inductive process could also be carried out to generalize these diagrams to increased truncation order n.

For $n = 2$, the (1, 0, 0), (0, 1, 0) and (1, 1, 0) balances may be arranged to yield three simultaneous equations for y_1, y_2, y_3. For example, the y_1 balance is given by

$$0 = -p\frac{y_1^2}{1 - c - y_2 - y_3}\left(1 - p\frac{y_2}{y_2 + y_3} - \frac{p^2}{2}\frac{y_3^2}{(y_2 + y_3)(c - y_2 - y_3)}\right) + \left(1 - \frac{y_1}{y_2 + y_3}\right)\left(p(1 - p)y_2 + \frac{p^2}{2}(1 - p)\frac{y_3^2}{c - y_2 - y_3}\right), \tag{24}$$

but we omit the details of the others for the sake of brevity. Having solved these equations, we may obtain a three-cell cluster approximation for the two-cell configuration probability in the form $y^{(3)} := P_2(1, 0) = P_3(1, 0, 1) + P_3(1, 0, 0)$ which equates to $y_2 + y_3$, and compare this with the prediction $y^{(2)}$ of the two-cell cluster approximation, which is the zero of (6). Figure 9 shows that there is very little discrepancy between the two approaches over a wide range of parameter values, indicating that the spatial independence assumption is a good approximation and that the two-cell cluster theory prediction $y^{(2)}$ is accurate. Future work should focus on establishing the hereditary relations which permit this approach to be extended to larger cluster sizes and higher order model truncations.

Acknowledgments

We thank Andreas Schadschneider and Marko Wölki for useful discussions, and George Lunt for deriving some of the preliminary results. Many thanks also to the anonymous referees for suggesting further work on three-cell cluster analysis.

References

[1] Nagel K and Schreckenberg M 1992 A cellular automaton model for freeway traffic *J. Phys.* I **2** 2221

[2] Rajewsky N, Santen L, Schadschneider A and Schreckenberg M 1998 The asymmetric exlusion process: comparison of update procedures *J. Stat. Phys.* **92** 151

[3] Chowdhury D, Santen L and Schadschneider A 2000 Statistical physics of vehicular traffic and some related systems *Phys. Rep.* **329** 199

[4] Schadschneider A 1999 The Nagel–Schreckenberg model revisited *Eur. Phys. J.* B **10** 573

[5] Derrida B, Domany E and Mukamel D 1992 An exact solution of a one-dimensional asymmetric exclusion model with open boundaries *J. Stat. Phys.* **69** 667–87

[6] Nagel K 1995 High-speed microsimulations of traffic flow *PhD Thesis* Universität zu Köln

[7] Lunt G 2001 Cellular automata models for flows without momentum *Master's Thesis* University of Bristol

[8] Wölki M, Schadschneider A and Schreckenberg M 2006 Asymmetric exclusion processes with shuffled dynamics *J. Phys. A: Math. Gen.* **39** 33–44

[9] Klüpfel H 2003 A cellular automaton model for crowd movement and egress simulation *PhD Thesis* Universität Duisburg

[10] Meyer-König T, Klüpfel H and Schreckenberg M 2001 A microscopic model for simulating mustering and evacuation processes onboard passenger ships *TIEMS*

[11] Schreckenberg M, Schadschneider A, Nagel K and Ito N 1995 Discrete stochastic models for traffic flow *Phys. Rev.* E **51** 2939

[12] Gutowitz H A, Victor J D and Knight B W 1987 Local structure theory for cellular automata *Physica* D **28D** 18–48