Some analytical and numerical approaches to understanding trap counts resulting from pest insect immigration

Daniel Bearup[a], Natalia Petrovskaya[b] & Sergei Petrovskii[c1]

[a] ICBM, University of Oldenburg,
Carl-von-Ossietzky-Strasse 9-11, D-26111 Oldenburg, Germany.

[b] School of Mathematics, University of Birmingham,
Birmingham B15 2TT, U.K.

[c] Department of Mathematics, University of Leicester,
University Road, Leicester LE1 7RH, U.K.

Abstract: Monitoring of pest insects is an important part of the integrated pest management. It aims to provide information about pest insect abundance at a given location. This includes data collection, usually using traps, and their subsequent analysis and/or interpretation. However, interpretation of trap count (number of insects caught over a fixed time) remains a challenging problem. Firstly, an increase in either the population density or insects activity can result in a similar increase in the number of insects trapped (the so called "activity-density" problem). Secondly, a genuine increase of the local population density can be attributed to qualitatively different ecological mechanisms such as multiplication or immigration. Identification of the true factor causing an increase in trap count is important as different mechanisms require different control strategies. In this paper, we consider a mean-field mathematical model of insect trapping based on the diffusion equation. Although the diffusion equation is a well-studied model, its analytical solution in closed form is actually available only for a few special cases, whilst in a more general case the problem has to be solved numerically. We choose finite differences as the baseline numerical method and show that numerical solution of the problem, especially in the realistic 2D case, is not at all straightforward as it requires a sufficiently accurate approximation of the diffusion fluxes. Once the numerical method is justified and tested, we apply it to the corresponding boundary problem where different types of boundary forcing describe different scenarios of pest insect immigration and reveal the corresponding patterns in the trap count growth.

Keywords: pest monitoring; insect trapping; random walk; diffusion; finite differences

[1]Corresponding author. Email: sp237@le.ac.uk; tel./fax: +44 116 252 3916 / 3915.

1 Introduction

Pest insect monitoring is an important component of the integrated pest management (IPM) (Kogan, 1998; Burn, 1987; Peshin and Dhawan, 2009). Its purpose is to obtain a reliable estimate of the pest abundance basing on the data collected in the field. A reliable estimate is required in order to help the IPM specialists to make an informed decision about control measures, e.g. application of chemical pesticides when the pest density exceeds a certain threshold (Stern, 1973; Higley and Pedigo, 1996) and yet to avoid their unjustified use.

Data on insect abundance are usually collected with traps (Pedigo and Buntin, 1994; Raworth and Choi, 2001). After a trap is set up in the field and has been exposed for a certain time, it catches a certain number n_1 of insects of a given species. This number is called a trap count; if, for instance, the trap was exposed for one day, it is called the daily count. In case $n_1 > 0$, this can be regarded as a proof that the species is present in the vicinity of the trap. However, relating the trap count to the population density is a much more difficult problem. Previous approaches tended to provide a relative rather than absolute estimate (Thomas et al.,1998). Recently, Petrovskii et al. (2012, 2014) showed that, if information is available about the insect movement pattern, the population density can be obtained by placing the sequence of daily counts against the predictions of a relevant mean-field mathematical model of the population dispersal. The simplest model of this type is the diffusion equation, which assumes that insects perform the Brownian motion, and indeed there is considerable evidence that they often do so (Turchin, 1998) although this may not always be readily seen from data (Petrovskii et al., 2011; Jansen et al., 2012).

The diffusion equation is a well-known and well-studied model and, in case of one spatial dimension, its general solution can usually be found analytically, albeit not always in a compact form. The situation is essentially different in case of higher dimension. In a 2D case, analytical solution of the diffusion equation is only possible for a few special cases where the domain possess a certain symmetry, e.g. has the shape of a rectangle or a disk. Even in those cases, however, the analytical solution often becomes impractical. For instance, in case of a disk-shaped domain, the solution can only be obtained as an infinite series where the coefficients are solutions of transcendent algebraic equations and hence can only be found numerically. In this situation, i.e. when numerical methods have to be used anyway, a reasonable alternative approach is to solve the diffusion equation numerically right away instead of using the semi-analytical method described above.

In this paper, we use a combination of analytical and numerical approaches to investigate the patterns in the trap count resulting from different boundary conditions as given by different ecological situations. One of the goals of pest insect monitoring is to detect an early signal of the population growth in order to prevent a pest outbreak. It is therefore important to understand how an increase in the population density can be reflected by the trap count. It seems intuitively obvious that a larger population size should eventually result in larger trap count. The rate of increase can, however, be different as an increase in the total population size can be attributed to different reasons. It can result from the growth of the local population, i.e. the population inside the given field, and indeed some insect species are capable of producing several generations in one year. It can also result from migration of the pest into the field from an external source, i.e. from another habitat. For the goals of pest control, it is important to distinguish between these situations (as the control measures are likely to be different) as well as between different immigration

patterns. Misidentification of the reason behind the pest abundance increase can result is a completely wrong estimate of the pest population density and that can have a detrimental effect on the efficiency of control measures. Here we are mostly concerned with the effect of immigration as the most common scenario; the effect of local population growth will be considered elsewhere.

There are a variety of numerical methods that can be used to solve numerically the diffusion equation; e.g. see Thomas (1995). However, we mention here that that their applicability and efficiency depend on the geometry of the domain. A typical domain in the trapping problem is not simply connected as it has an external boundary (i.e. the boundary of the monitored farm field) and the internal boundary (the boundary of the trap). Moreover, the size of the trap is usually much less than the size of a typical farm field; therefore, the problem has at least two clearly different spatial scales. Application of standard methods to a system like this may bring significant computational difficulties (Petrovskii and Petrovskaya, 2012). Besides, in order to calculate the trap count, one has to calculate the population density gradient at the trap boundary. This can be a challenging task, especially at the corner points if the trap has a shape other than circular, and indeed use of traps of various shapes and designs has been increasingly common (cf. Flechtmann et al., 2000). Thus, we have to pay a special attention to numerical issues before discussing ecological results.

The paper is organized as follows. In section 2.1, we describe the mathematical model and provide a comprehensive analytical study of trap count in the baseline 1D case under various migration scenarios. Although the 1D case is hardly realistic, it provides an important theoretical background for the understanding of a more realistic 2D case. We then briefly revisit the finite-difference method for numerical solution of the diffusion equation and show how it can be used to calculate the trap count in the 1D case (section 2.2). In section 3, we carefully test our computational technique against the analytical results in the 1D case. In section 4, we apply our approach to a hypothetical 1D system in order to reveal generic patterns in the trap count arising from different ecological scenarios. In section 5, we consider an extension of our method onto the more realistic 2D case and discuss the arising computational issues. We then provide a thorough analysis of trap count for different immigration scenarios by solving the 2D diffusion equation numerically (section 6). Finally, in section 7 we discuss the ecological implications of our results.

2 Mathematical model and numerical method, 1D case

2.1 Model

Since the focus of this paper is on the effect of immigration, we neglect the population reproduction, thus assuming that trap counts are collected in the period between the generations. Additionally, for the sake of simplicity, we neglect the population losses due to mortality. The equation describing the population dynamics in space is then essentially the mass conservation law which, in the 1D case, has the following form:

$$\frac{\partial u(x,t)}{\partial t} + \frac{\partial j(x,t)}{\partial x} = 0\,, \tag{1}$$

where $u(x,t)$ is the population density at the position x and time t, and j is the population density flux in the direction of axis x. The mathematical description of the flux depends on the type of the

individual movement. In a relatively general case, individual insects perform a combination of the non-directed random-like movement that can often be regarded as the Brownian motion (Turchin 1998; Okubo and Levin 2001), and a directed movement with a certain speed v. The corresponding population flux is then given by

$$j(x,t) \;\; = \;\; -D\,\frac{\partial u(x,t)}{\partial x} + vu(x,t)\;, \tag{2}$$

where D is the diffusion coefficient. Whilst the directed movement becomes important in the presence of environmental gradients, the non-directed random-like motion is an inherent property of almost all ecological populations.

Insect monitoring is done with traps. Once an insect encounters the trap, it is caught with a certain probability $p_0 < 1$ where p_0 depends on the species traits and the trap design. Throughout this paper, we assume that the trap design is sufficiently efficient so that $p_0 \approx 1$. Indeed, this is often the case with *walking insects*, even for a simple pitfall trap design. With regard to the effect of species traits and/or the movement mode, many insects combine flying with walking. Whilst flying is the preferred movement mode when insects travel over long distances (e.g. looking for a new feeding or breeding ground), their movement on the feeding site is typically a combination of walking and very short flights. Correspondingly, here assume that, once the insects arrive at the farm-field, they mostly move around by walking.

Regarding the trap design, traps can be either baited or non-baited. Baited traps use a certain substance (e.g. pheromone) or agent (e.g. light or color) in order to attract insects to the trap. This introduces an advective component to the insect movement as they are more likely move towards the trap rather than in any other direction. In contrast, non-baited traps do not introduce any directional bias as they capture insects just because of their random encounters with the trap. In this paper, we focus on non-baited traps only; consideration of baited traps involves an essentially different set of assumptions (in particular, about the insect's behavioral response to the attracting agent) and hence will be done elsewhere (Bearup et al., 2015).

Let us consider an idealized 1D farm-field described by the domain $0 < x < L$. We assume that the field is homogeneous and the trap is non-baited. Correspondingly, inside this domain – but not necessarily outside, see below – the monitored insect population performs only random motion, i.e. $v = 0$ in Eq. (2). From Eqs. (1–2), we then obtain the diffusion equation:

$$\frac{\partial u(x,t)}{\partial t} \;\; = \;\; D\,\frac{\partial^2 u(x,t)}{\partial x^2}\;. \tag{3}$$

The trap, which we assume to be escape-proof, is installed at the left-hand side boundary of the domain, i.e. at $x = 0$. The corresponding condition at the trap boundary is

$$u(0,t) \;=\; 0\;. \tag{4}$$

Once the solution $u(x,t)$ of the diffusion equation is known, the corresponding trap count over time t can be obtained as

$$\Delta U(t) \;=\; \int_0^t j_0(\tau)d\tau\;, \tag{5}$$

where

$$j_0(t) \;=\; D\frac{\partial u(0,t)}{\partial x} \geq 0 \tag{6}$$

is the population density flux into the trap. Note that the minus in the right-hand side of Eq. (2) has now changed to plus because of our choice of the trap location, i.e. because the population flux into the trap goes against axis x.

In order to make the mathematical problem complete, the diffusion equation must be complemented with the initial condition and with a boundary condition at $x = L$. In this paper, we are mostly interested in the effect of the conditions at the field boundary arising from different ecological situations. Hence, in order to avoid unnecessary complexity, for the initial condition we consider the simplest case when the population is distributed uniformly around the domain:

$$u(x, 0) \; = \; U_0, \quad \text{for } 0 < x < L \; , \tag{7}$$

where $U_0 \geq 0$ is a parameter with the obvious meaning.

The situation at the field boundary (i.e. at $x = L$) requires a much more careful consideration. In order to understand what form of the boundary condition may be relevant, one has to consider the population dynamics not only inside the domain of interest (farm-field) but also in the adjacent habitat (non-farmed field); see Fig. 1. Moreover, sometimes the inter-habitat boundary itself may have to be accounted for as a separate domain, in particular, in the cases where the boundary has an inner structure and may by itself be a habitat.

The focus of this paper is the population dynamics in a given domain resulting from immigration. Correspondingly, a relevant starting point is the quantification of the immigration's impact. The population flux (cf. Eq. (2)) is the number of individuals crossing the boundary per unit time and hence is a relevant quantity. Note that, whilst we have assumed that inside the domain $0 < x < L$ insects move randomly, outside of the domain insects can perform either a random motion or a directed movement, or a combination of both. With regard to the relative importance of the directed and non-directed movement, we consider two different cases.

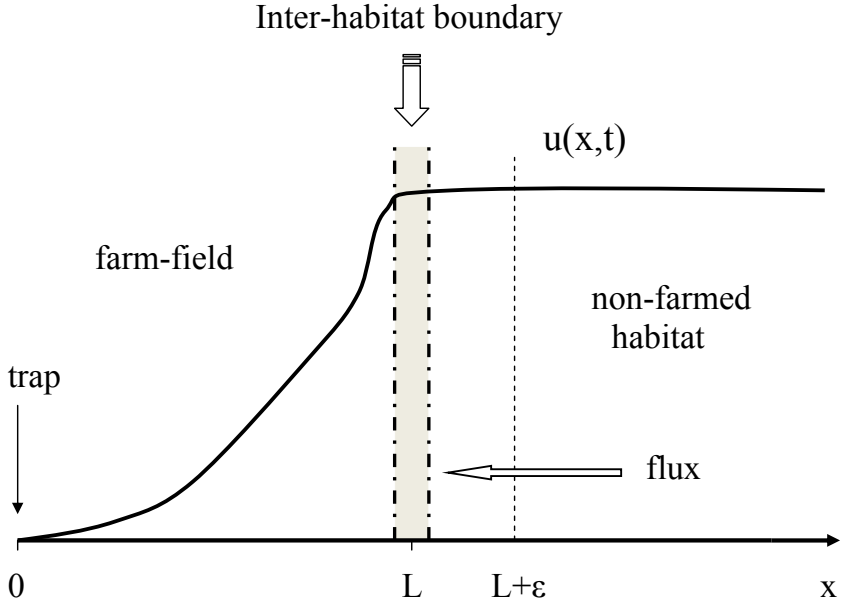


Figure 1: A sketch of the boundary forcing of the monitored habitat (farm-field) because of the insect immigration from the adjacent (non-farmed) habitat described by the population flux; see details in the text.

Case I: the insect movement in the adjacent habitat is not affected by the presence of the neighboring farm-field (i.e. there is no odor or smell that could act as an attractant). The insects move randomly in a diffusive manner, i.e. there is no advection. Additionally, we assume that the population density distribution over the non-farmed habitat is approximately uniform. Then at any location $x = L+\epsilon > L$, we have $j(x+\epsilon, t) = -D\partial u(x+\epsilon, t)/\partial x \approx 0$, i.e. the left-hand and right-hand diffusive fluxes compensate each other. The population in the non-farmed domain is therefore at an equilibrium state. The insects that move around in the close vicinity of the boundary $x = L$ can, because of the randomness of their movement, occasionally move across the habitat boundary $x = L$, i.e. to the farm-field. However, this loss of individuals is immediately compensated by arrival, due to diffusion, of insects from areas further inside the non-farmed domain so that the value of the population density at any location $L+\epsilon$ close to the boundary remains unchanged. We consider that this value, say U_b, does not depend on time. Thus, since the population density is a continuous function of space, we arrive at the following Dirichlet-type condition:

$$u(L, t) = U_b \, . \tag{8}$$

Below we will refer to this case as the *diffusive* boundary forcing referring to the fact that the population outside of the domain (i.e. in the non-farmed field) performs diffusive movement.

Case II: there is an attractant emanating from the farm-field into the adjacent non-farmed habitat. In the non-farmed habitat, it results in the directed movement of the insects from inner areas towards the inter-habitat boundary. Because of this, the left-hand and right-hand fluxes do not compensate each other and, at any location $x = L + \epsilon$ sufficiently close to the boundary (ϵ is small), there is a population flux towards the boundary. We now additionally assume that the boundary has no capacity, that is, the number of the individuals per unit time coming to the boundary from the right (from the non-farmed habitat, see Fig. 1) must be balanced by the same number going away from the boundary to the left (to the farmed habitat). The advective flux J at the right-hand side of the boundary must therefore be balanced by the purely diffusion flux at the left-hand side of the boundary, so that we arrive at the Neumann-type boundary condition:

$$D \left.\frac{\partial u(x, t)}{\partial x}\right|_{x=L} = J \qquad \text{or} \qquad \frac{\partial u(L, t)}{\partial x} = G \, , \tag{9}$$

where $G = J/D$ is thus the value of the density gradient. For the sake of simplicity, we assume that J is constant, so that G is constant as well. Below we refer to this case as the *advective* boundary forcing referring to the fact that outside of the domain the population performs advective movement.

We mention here that the above interpretation of the boundary conditions to some extent depends on the assumptions about the population dynamics outside of the monitored region $0 < x < L$. In particular, in a more complicated case when the environmental heterogeneity in the adjacent non-farmed habitat $x > L$ cannot be neglected (which, for instance, can be a result of population losses due to mortality, e.g. see Ludwig et al. 1979), both the nature of the movement process of the organisms and the nature of the underlying habitat can influence the density and flux of organisms arriving at the interface $x = L$. The boundary condition then may be better described by the Robin-type condition. However, we do not consider this situation here in detail in order to avoid unnecessary complexity.

Now we are going to consider what is the mathematical expression for the trap count in case of different boundary conditions as given by Cases I and II above.

Impenetrable boundary. We begin with the simplest case where there is no forcing at all, i.e. no insect immigration or emigration, because the boundary is impenetrable:

$$\frac{\partial u(L,t)}{\partial x} \;=\; 0. \tag{10}$$

The problem (3), (4) and (7) together with (10) can be solved analytically using the separation of variables method (e.g. Crank 1975). The solution $u(x,t)$ is given by the following infinite series:

$$u(x,t) \;=\; \frac{4U_0}{\pi}\sum_{k=0}^{\infty}\frac{1}{(2k+1)}\sin\left(\frac{(2k+1)\pi x}{2L}\right)\exp\left(-\frac{(2k+1)^2\pi^2 Dt}{4L^2}\right). \tag{11}$$

From (11), the diffusion flux (6) at the trap boundary is obtained as

$$j_0(t) \;=\; \frac{2DU_0}{L}\sum_{k=0}^{\infty}\exp\left(-\frac{(2k+1)^2\pi^2 Dt}{4L^2}\right), \tag{12}$$

and, correspondingly, the number of insects caught over time t of the trap exposure is calculated as

$$\Delta U(t) \;=\; \frac{8LU_0}{\pi^2}\sum_{k=0}^{\infty}\frac{1}{(2k+1)^2}\left[1-\exp\left(-\frac{(2k+1)^2\pi^2 Dt}{4L^2}\right)\right], \tag{13}$$

where LU_0 is the total number of insects for $x>0$ at $t=0$. Note that, since

$$\sum_{k=0}^{\infty}\frac{1}{(2k+1)^2} \;=\; \frac{\pi^2}{8}\,, \tag{14}$$

in the large-time limit $\Delta U(t)\to LU_0$, i.e. all insects are trapped.

It also follows from (13) that the trap count can be approximated as

$$\Delta U(t) \;\approx\; \frac{2U_0}{\sqrt{\pi}}\sqrt{Dt}, \tag{15}$$

which shows a very good accuracy when either time t is sufficiently small or the domain length L is sufficiently large, or both; see Petrovskii et al. (2012) for details of this approximation. A straightforward derivation of Eq. (15) in case of a semi-infinite domain $x>0$ can be found in Petrovskii et al. (2014).

Diffusive boundary. We now consider the case of the diffusive forcing, i.e. the diffusion problem (3), (4) and (7) together with the condition (8). Using the same method as above, we arrive at the following expression for the trap count:

$$\begin{aligned}\Delta U(t)^{Dir} \;=\;& \frac{DU_b}{L}t+\frac{2U_bL}{\pi^2}\sum_{k=1}^{\infty}\frac{(-1)^k}{k^2}\left[1-\exp\left(-\frac{Dk^2\pi^2 t}{L^2}\right)\right]\\ &+\frac{4U_0L}{\pi^2}\sum_{k=1}^{\infty}\frac{1}{(2k-1)^2}\left[1-\exp\left(-\frac{D(2k-1)^2\pi^2 t}{L^2}\right)\right].\end{aligned} \tag{16}$$

Advective boundary. Finally, in the case of advective forcing, we consider (3), (4) and (7) together with (9). The corresponding expression for the trap count is:

$$\begin{aligned}\Delta U(t)^{Neu} &= GDt + \frac{16GL^2}{\pi^3}\sum_{k=1}^{\infty}\frac{(-1)^k}{(2k-1)^3}\left[1-\exp\left(-\frac{D(2k-1)^2\pi^2 t}{4L^2}\right)\right] \\ &+ \frac{8U_0L}{\pi^2}\sum_{k=1}^{\infty}\frac{1}{(2k-1)^2}\left[1-\exp\left(-\frac{D(2k-1)^2\pi^2 t}{4L^2}\right)\right]. \end{aligned} \tag{17}$$

In the special case of impenetrable boundary, i.e. for $G = J = 0$, (17) coincides with (13).

Solutions (13), (16) and (17) will be used below to test the numerical method.

We mention here that, in case of any more complicated boundary condition at the external field boundary (e.g. of the Robin type), the solution of the corresponding diffusion problem would not be available in closed form. Each term in the series would include a coefficient that could only be obtained numerically by solving a transcendent algebraic equation. This emphasizes the need for an alternative, numerical method to solve the diffusion equation that will be discussed below.

2.2 Numerical method

In this section, we look at the numerical method we use to obtain the solution, i.e., a finite difference method. We aim to discuss finite difference discretization of the 1D problem in detail, as most of computational issues arising when the 2D problem is considered are already present in the 1D case.

Let us introduce a uniform computational grid $\mathcal{G}$ in the domain $x \in [0, L]$ in order to discretize the diffusion equation (3). We have $x_1 = 0$, $x_{i+1} = x_i + h$, $i = 1, \ldots, N$, where $h = L/N$ is the grid step size, and N is the number of grid subintervals.

Let $u(x, t_n)$ be the solution to the problem (3), (4), (7) (complemented with a relevant boundary condition at the external boundary, as discussed in 2.1) at the fixed time $t = t_n$. Let also $u(x, t_{n+1})$ be the solution at the fixed time $t = t_{n+1}$, where $t_{n+1} = t_n + \tau$, $\tau > 0$. A finite difference discretization scheme is a method widely used for numerical solution of parabolic differential equations (e.g. see Hirsch 1992, Holmes 2006, Morton 1994, Smith 1985). The scheme defines the approximate solution $u(x, t_{n+1})$ at each grid node x_i of a uniform grid $\mathcal{G}$ from solving a system of linear algebraic equations

$$\frac{1}{\tau}(u_i^{n+1} - u_i^n) = \Lambda[\sigma u_i^{n+1} + (1-\sigma)u_i^n], \tag{18}$$

where we use the notation $u_i^n \equiv u(x_i, t_n)$ and $u_i^{n+1} \equiv u(x_i, t_{n+1})$. The definition of discrete spatial operator Λ is

$$\Lambda[v_i] = \frac{D}{h^2}(v_{i+1} - 2v_i + v_{i-1}), \tag{19}$$

where v_i is a discrete function defined at nodes of the grid $\mathcal{G}$, h is the grid step size and D is the diffusion coefficient.

The weight parameter σ defines a type of the scheme. The weight coefficient $\sigma = 1$ provides us with an implicit (and therefore unconditionally stable) scheme. The weight $\sigma = 0$ results in

an explicit scheme, where the solution u_i^{n+1} at time t_{n+1} is readily computed from the solution u_i^n taken from the previous time layer t_n. In our work we employ the latter version of the scheme with $\sigma = 0$. The theoretical approximation error related to this scheme is $O(h^2 + \tau)$ and the scheme is stable if the following condition holds:

$$\frac{D\tau}{h^2} \leq \frac{1}{2}. \tag{20}$$

While we are fully aware of the issues of the stability and accuracy raised when the explicit scheme $\sigma = 0$ is used, we advocate our choice because of simplicity of computations. Let us note that we are interested in the solution at small times t, where we assume that the approximate solution is not noticeably affected by approximation error $O(\tau)$ of the scheme. This assumption is further carefully checked in a series of numerical tests discussed in section 3. Also, the condition (20) implies a very small time step size τ on fine grids with $h \ll 1$, but it does not significantly increase the cost of our computations at small times.

The equations (18) are solved for $i = 2, \ldots, N$. For the discretization of the boundary conditions (4)-(10), at any time $t_n > 0$ we have the following equations

$$\begin{aligned} u_1^{n+1} &= 0 \quad \text{for } x = 0, \\ u_{N+1}^{n+1} &= u_{N+1}^n + \frac{2\tau D}{h^2}(u_N^n - u_{N+1}^n) \quad \text{for } x = L. \end{aligned} \tag{21}$$

Equations (21) provide us with the second order approximation of the boundary condition (e.g. see Thomas 1995) and is therefore consistent with the accuracy of approximation (18).

Similarly, we have

$$u_{N+1}^{n+1} = u_{N+1}^n + \frac{2\tau D}{h^2}(u_N^n - u_{N+1}^n + hG) \quad \text{for } x = L, \tag{22}$$

when the boundary condition (9) is used. The discretization of the boundary condition (16) is straightforward and results in

$$u_{N+1}^{n+1} = U_b \quad \text{for } x = L. \tag{23}$$

Finally, the discretization of the initial condition (47) is

$$u_i^0 = U_0, \quad i = 2, \ldots, N. \tag{24}$$

Calculation of trap count. Let the solution $u(x, t_{n+1})$ be known to us at any fixed time $t = t_{n+1}$. In our further discussion in this section we omit the notation t_{n+1} for the sake of convenience and consider a spatial distribution $u(x)$. The flux $j(x)$ of the population density $u(x)$ at point x is given by the formula (6). Hence the flux computation requires approximation of the derivative $\dfrac{du(x)}{dx}$ at grid node $x_1 = 0$. The accuracy of this approximation should be consistent with the accuracy of finite difference approximation (18), that is the approximation error should be $O(h^2)$. It is explained in Appendix A that the required order of accuracy can be achieved by using the following flux approximation at the trap boundary $x = 0$:

$$j(0) \approx \frac{D}{2h}|4u_2 - u_3|. \tag{25}$$

Once the flux $j(0)$ has been computed, an approximation of the total number of insects $\Delta U^{n,n+1}$ crossing the trap boundary between time t_n and t_{n+1} is obtained as $\Delta U^{n,n+1} = j(0)\tau$. The cumulative trap count $\Delta U(t_{n+1}) = \Delta U^{n+1}$ at time t_{n+1} is then computed by adding this value to that obtained at the previous time t_n:

$$\Delta U^{n+1} = \Delta U^n + \Delta U^{n,n+1}. \tag{26}$$

3 1D case: Validation of the method and numerical tests

In this section we validate our finite difference discretization through a number of numerical tests. Our first test case is to check the spatial convergence of a numerical solution on a sequence of refined grids. The convergence graph is shown in Fig. 2a where the trap count error is computed as a difference between the analytical solution (13) and a numerical solution (26). The time t is fixed as $t = 100$ and the diffusion coefficient is $D = 1$. We conclude from the slope of the convergence graph that our spatial discretization is indeed second order accurate.

Consider now the temporal-spatial discretization. The trap count obtained from a numerical solution have been computed for various time t on a sequence of grids with grid step size varying from $h = 0.5$ to $h = 0.05$. The results are shown in Fig. 2b. It can be seen from the figure that the trap count obtained from the numerical solution are, over the given time, visually indistinguishable from those produced by the truncated exact solution on the scale used.

More accurate validation of the numerical solution requires computation of the trap count error. We introduce the trap count error e^{n+1} at time $t = t_{n+1}$, $n = 0, 1 \ldots$ as the relative difference between the exact trap count $\Delta U^{n+1}_{\text{exact}}$ and the approximate trap count $\Delta U^{n+1}_{\text{approx}}$ obtained from a numerical solution,

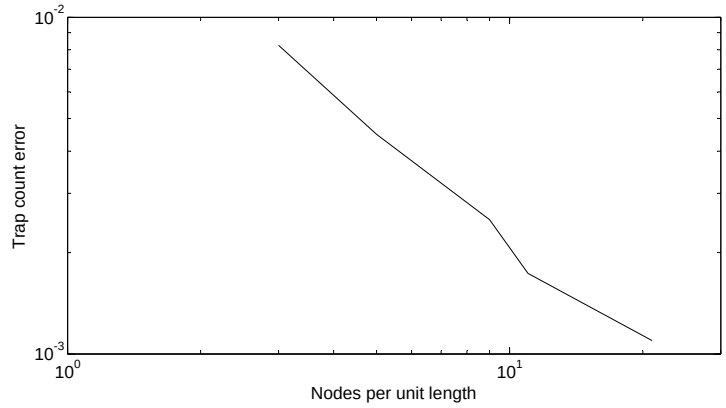
$$e^{n+1} = \frac{|\Delta U^{n+1}_{\text{exact}} - \Delta U^{n+1}_{\text{approx}}|}{\Delta U^{n+1}_{\text{exact}}}. \tag{27}$$

In computation of the error (27) the exact solution (13) was truncated as

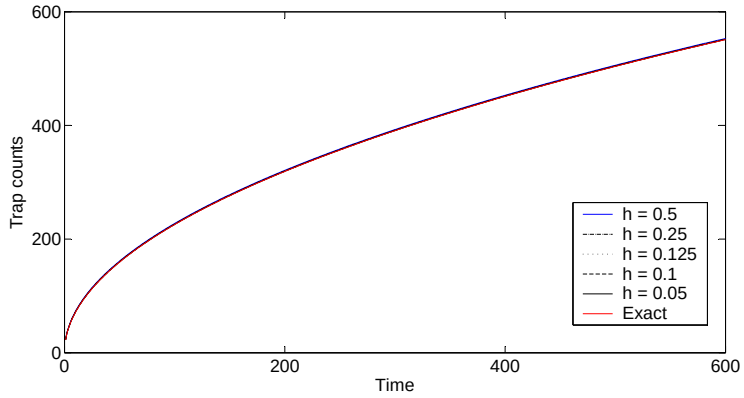
$$\Delta U(t) \;=\; \frac{8LU_0}{\pi^2} \sum_{k=0}^{K=200} \frac{1}{(2k+1)^2} \left[1 - \exp\left(-\frac{(2k+1)^2\pi^2 Dt}{4L^2}\right)\right], \tag{28}$$

The truncated solution has been verified against the approximation (15) at small times to guarantee that $K = 200$ terms in the above sum is sufficient to approximate an infinite series. The error e^{n+1} is shown in Fig. 2c as a function of time. It has been computed on a sequence of grids refined from from $h = 0.5$ to $h = 0.05$. Obviously, the finer the grid is the smaller the relative error. Overall the relative errors obtained on any sufficiently fine grid (e.g. with $h \leq 0.125$) are very small (on the order of 1 percent or smaller) at any time point.

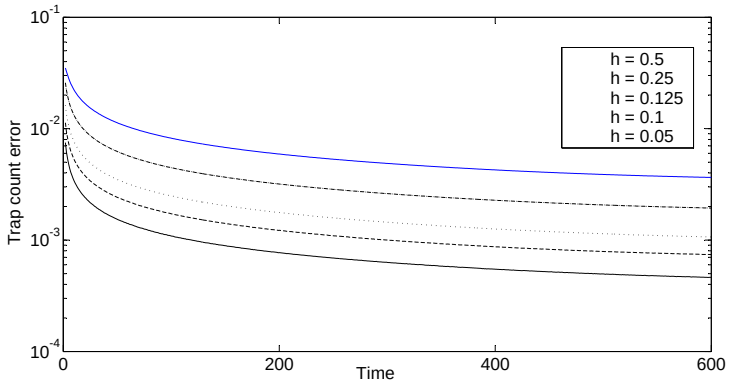
Let us note that, for the validation of our numerical method, we have used the exact solution obtained for the simplest boundary and initial conditions as given by (4), (7) and (10). In the case that a different, more complicated boundary condition and/or initial condition are used in the diffusion problem, the exact solution may not always be available in an explicit form; in particular, it is rarely available in the 2D case (considered in sections 5 and 6 below). Therefore, an alternative approach is required to corroborate a numerical solution. Thus our next test is to compare the trap count obtained from flux calculations with trap count calculated by a different method.



(a)

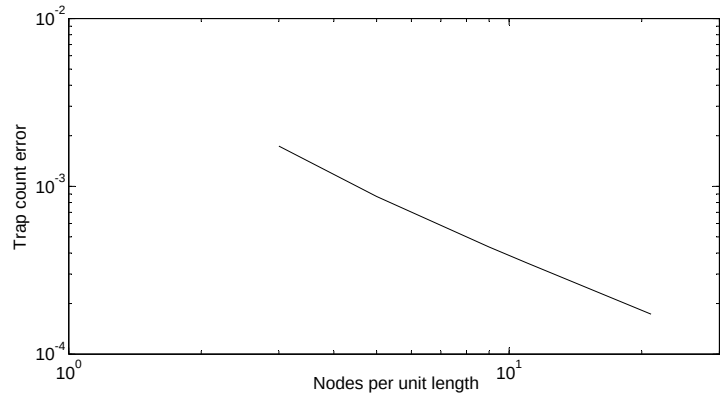


(b)

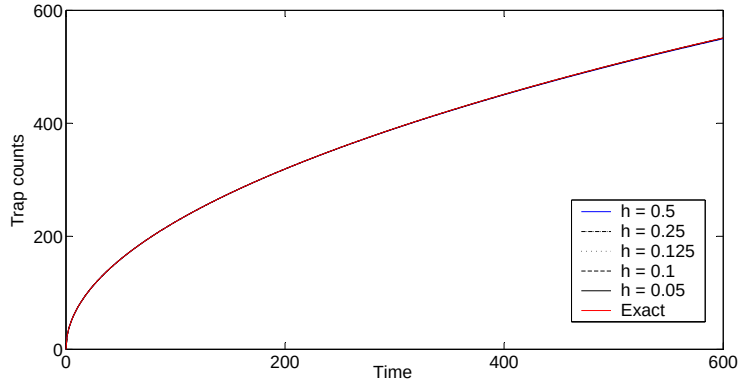


(c)

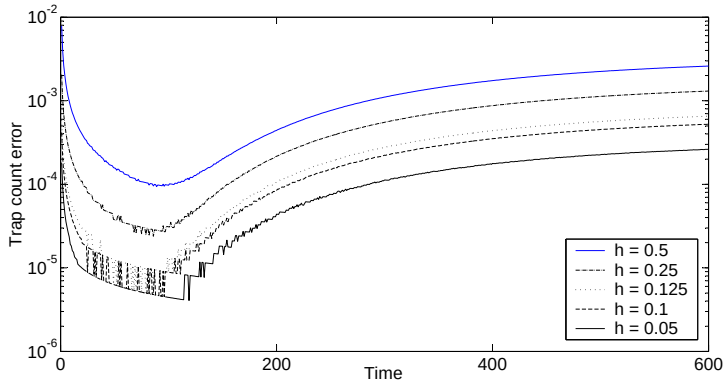
Figure 2: (a) Spatial convergence of the numerical solution on a sequence of uniformly refined spatial grids. (b) Trap count obtained by (15) on a sequence of refined grids with the grid step size h (see inset key) in comparison with the truncated exact solution (28). The six different curves are visually indistinguishable. (c) The trap count error (27) as a function of time. In all cases (a-c), the parameters are $L = 49.5$, $D = 1.0$, $\tau = 1/1600$ and $U_0 = 10$.



(a)



(b)



(c)

Figure 3: (a) Spatial convergence test. The error is computed as a difference between the solution (30) and the integral of the truncated density function on a sequence of uniformly refined spatial grids. (b) Plots of trap count obtained using numerical integration on increasingly fine grids (see inset key) in comparison with the exact solution obtained as the integral of the truncated exact density function. The six different curves are visually indistinguishable. (c) Relative differences between the numerical solution and the exact solution plotted against time. In all cases (a-c), the parameters are the same as in Fig. 2.

We mention here that, in the strict sense, validation of the numerical method in the absence of the exact solution is hardly possible: indeed, having calculated the solution on a sequences of refined grids, what can we compare it to? Note that good agreement between the results obtained by two different methods does not, by itself, prove anything as, generally speaking, both methods can be equally wrong. Yet if we can reproduce the results (i.e. trap count) by using a different numerical method, that, albeit not being a proof as such, should be regarded as an indirect evidence of the results' plausibility.

As an alternative method, we consider the numerical integration of the solution $u(x, t_n)$. The integration gives the size of the 'free' population inside the computational domain at time t_n. Recall that our population model assumes no reproduction or mortality of individuals within the domain. Furthermore, for the boundary condition (10), the impermeable external boundary admits no migration into or out of the domain. As such the population changes only as a result of trapping. Hence the sum of the free population and the trap count remains constant and is equal to the initial population LU_0 at any given time.

The free population U^{n+1} is calculated by numerical integration of the population density distribution over the domain $x \in [0, L]$ at the time t_{n+1}. Once the population density $u(x, t_{n+1})$ has been defined at the nodes of a uniform computational grid $\mathcal{G}$, a composite rule of numerical integration can be applied to integrate the spatial distribution $u(x)$ at the fixed time t_{n+1},

$$U^{n+1} \equiv U(t_{n+1}) = \int_0^L u(x)dx \approx \sum_{i=1}^{N+1} \omega_i u_i, \tag{29}$$

where $u_i \equiv u_i^{n+1}$, $i = 1, \ldots, N+1$ are function values at grid nodes and the weights ω_i are defined according to the chosen rule of numerical integration. In our work we apply the trapezoidal rule, where the weights are $\omega_1 = \omega_{N+1} = h/2$, $\omega_i = h$, $i = 2, \ldots, N$. The accuracy of this integration rule on a uniform grid with the grid step size h is $O(h^2)$ (e.g. Davis 1975).

The trap count can then be computed as

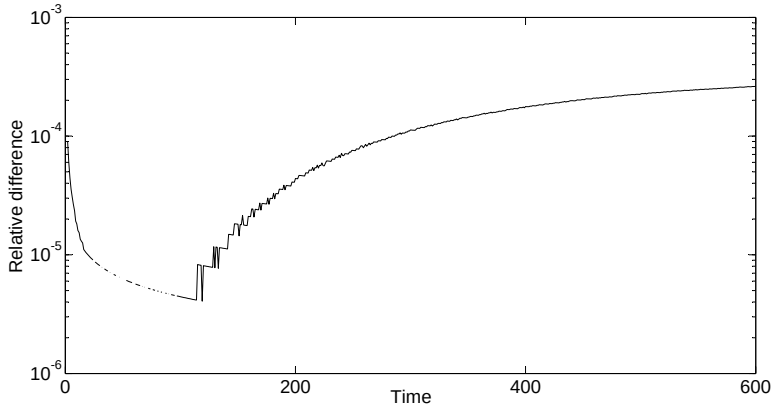
$$\Delta U^{n+1} = LU_0 - U^{n+1}, \tag{30}$$

where LU_0 and U^{n+1} are the free population at time $t = 0$ and $t = t_{n+1}$ respectively.

The results of numerical integration (29–30) are presented in Fig. 3. The spatial convergence of a numerical solution on a sequence of refined grids is shown in Fig. 3a, where the time t is fixed as $t = 100$ and the other parameters are $L = 49.5$, $D = 1$. In order to calculate the error, we compare a solution obtained by numerical integration of the density function with the truncated integral of the exact density.

The trap count as a function of time is shown in Fig. 3b on a sequence of grids with grid step size varying from $h = 0.5$ to $h = 0.05$. The trap count obtained from the numerical solution appear to be visually indistinguishable from those produced by integrating truncated exact density. This conclusion is further confirmed by computation of the error (27), where $\Delta U^{n+1}_{\text{approx}}$ is now a numerical solution (30). The relative error as a function of time is shown in Fig. 3c on spatial grids with the grid step size from $h = 0.5$ to $h = 0.05$. It is clear from the figure that the accuracy of computation is very good on any grid used.

Let us now compare the trap count obtained by the two different methods, i.e. by the flux calculation at the trap boundary, see Eqs. (25–26), and by the numerical integration of the population density of the free population, see Eq. (30). Figure 4 shows the absolute value of the relative difference between the two numerical solutions obtained on a fine grid with $h = 0.05$ for parameters $L = 49.5$ and $D = 1$. It is readily seen that the two methods are in a very good agreement between themselves, and both numerical solutions are very close to the exact solution of the problem; see also Figs. 2 and 3. We therefore assume that, when the exact solution is not available, the trap count obtained from numerical integration can be used to verify those obtained from flux calculations. This conclusion is going to be practically important for the analysis of the 2D problem.



(a)

Figure 4: Comparison of two numerical solutions on a fine grid with $h = 0.05$. The relative difference between the solution obtained from direct flux calculation and the solution by numerical integration is computed as a function of time.

4 1D results for various ecological conditions

We are now going to consider the trap count dynamics subject to different types of forcing at the external boundary as given by Eqs. (8) and (9). For doing this, we combine analytical approaches with numerical simulations. The diffusion problem in the 1D case can be studied analytically, e.g. see Eqs. (16) and (17) and the theorem at the end of this section. However, basic properties of the corresponding solutions are sometimes easier to see from numerical simulations. We therefore begin with numerical results and their semi-quantitative, heuristic analysis.

Recall that in the 1D system the analytical expression for the trap count is available, cf. Eqs. (16) and (17). It is readily seen that a truncated series with a sufficiently large number of terms provides a very good approximation to the exact solution. This approximate solution is used below to analyze the trap count for different conditions. Some of those approximate-analytical results are also reproduced using the numerical method described in the previous section, thus using the analytical solution as a test, with the idea to later apply numerical simulations extensively in the 2D case where the analytical solution is not available.

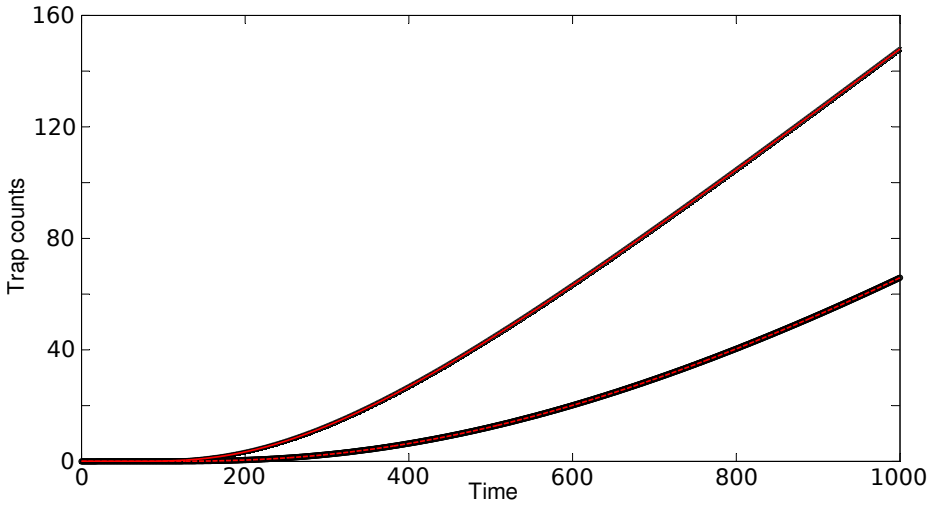


Figure 5: Trap count vs time as given by the series (16) (upper curve) and (17) (lower curve), i.e. for diffusive and advective forcing respectively, and as obtained from numerical solutions of the corresponding systems. Parameters are: $D = 1$, $L = 45$, $U_0 = 0$, $U_b = 10$ and $G = U_b/L$. Red curves correspond to the analytical solutions (truncated at 100 terms), black symbols correspond to numerical solutions. Note the excellent agreement between the analytical and numerical results.

In order to provide a convenient framework for the understanding of the trap count dynamics, we notice that, both for diffusive forcing (8) and advective forcing (9), the solution of the diffusion equation has stationary large-time asymptotics, $\bar{u}_D(x)$ and $\bar{u}_N(x)$ respectively, where

$$\bar{u}_D(x) \;=\; \frac{U_b x}{L} \qquad \text{and} \qquad \bar{u}_N(x) \;=\; Gx\;. \tag{31}$$

We therefore obtain that, in the large-time limit, the trap count is a linear function of time, that is

$$\Delta U(t)^{Dir} \;\simeq\; \frac{DU_b t}{L} + C_1 \qquad \text{and} \qquad \Delta U(t)^{Neu} \;\simeq\; DGt + C_2\;, \tag{32}$$

respectively, where the value of the constants C_1 and C_2 results from the transient short-time and intermediate-time dynamics; in particular, it depends on the initial conditions. If we consider $G = U_b/L$, then the only difference between the two types of forcing is due to the transient stage. In order to make the solutions for diffusive and advective forcing comparable, throughout this section we choose G to be equal to U_b/L.

Diffusion is a slow process; whatever the type of the boundary forcing, it clearly takes some time for its effect to propagate through the domain and to reach the trap (we will address this issue more quantitatively below). It is less clear which type of the forcing may result in a stronger effect. Intuitively, the Neumann-type advective boundary condition corresponds to a more active forcing (see Cases 1 and 2 in section 2.1), hence it seems reasonable to expect that the trap count should grow faster in that case than in case of the diffusive, Dirichlet-type, boundary condition. Interestingly, this intuitive argument appears to be misleading. Figure 5 shows the trap count obtained both analytically (red curves) and numerically (black symbols) for diffusive forcing (upper curve)

and advective forcing (lower curve). As was expected, the effect of the external forcing is not seen until after a considerable delay. However, diffusive forcing appears to have a much stronger effect on the trap count than advective forcing.

In order to make a more detailed insight into the effect of the domain length, we introduce a certain threshold count and consider how the time t_1 required for the trap count to reach the threshold depends on the distance L between the trap and the external boundary (i.e. the domain length). Figure 6 shows this time t_1 vs L for the threshold count equal to one, i.e., effectively, the time when the first insect from the monitored population is caught. We therefore observe that, the larger is the distance L the higher is the efficiency of diffusive forcing compared to advective forcing.

The faster-than-linear (approximately quadratic) growth in the time t_1 as a function of L can be explained using the dimensions analysis (Barenblatt 1996). From the quantities t_1, L, D, U_0 and U_b or G one can make exactly two dimensionless combinations, that is $\Pi = t_1 D L^{-2}$ and either $\Pi_1 = U_0/U_b$ or $\Pi_2 = U_0/(GL)$, for diffusive and advective forcing, respectively. Then, according to the so called Π-theorem (Buckingham 1914; also Barenblatt 1996), one of these dimensionless combinations must be a function of the other, i.e.,

$$\Pi = f_1(\Pi_1) \quad \text{or} \quad \Pi = f_2(\Pi_2), \tag{33}$$

for diffusive or advective forcing, respectively, where f_1 and f_2 are certain 'universal' functions. For time t_1, we therefore obtain:

$$t_1 = \frac{L^2}{D} f_1(\Pi_1) \quad \text{or} \quad t_1 = \frac{L^2}{D} f_2(\Pi_2). \tag{34}$$

Functions f_1 and f_2 are not known; however, we observe that, in the special but important case $U_0 = 0$, their argument is zero as $\Pi_1 = \Pi_2 = 0$. The values $f_1(0)$ and $f_2(0)$ then just act as coefficients. We therefore obtain that t_1 is proportional to L^2, which agrees very well with Fig. 6.

The above results were obtained for $U_0 = 0$, i.e. when there is no native population in the domain and the trap count only account for the boundary forcing resulting from the immigration. The dynamics becomes different and somewhat more complicated when the native population is present, i.e. for $U_0 > 0$. (This, for instance, can be seen from Eqs. (34) where time t_1 remains proportional to L^2 in case of diffusive forcing but turns into a more complicated relation in case of advective forcing as L now enters the argument of f_2.) For $U_0 > 0$, the trap count growth at small t (i.e. before the effects of the boundary reach the trap) is determined solely by the effect of the initial conditions (i.e. by the native population) and is known to increase as $\sqrt{t}$ (Petrovskii et al. 2012). If considered over a long time, the pattern in the trap count growth can be regarded as a transition between the two asymptotics, i.e. the small time behavior $\Delta U \sim \sqrt{t}$ and the large-time behavior $\Delta U \sim t$ (cf. Eqs. (32)); this transition is clearly seen in Figs. 7. Note that, at small t, the trap count are the same irrespective of the boundary condition as only the initial conditions are having effect.

The effect of the initial conditions does complicate the behaviour of the system. We showed above that for $U_0 = 0$ diffusive forcing always has a stronger effect on the trap count than the equivalent[2] advective forcing. For $U_0 > 0$, the type of forcing that results in a larger trap count

[2]i.e. when $G = U_b/L$.

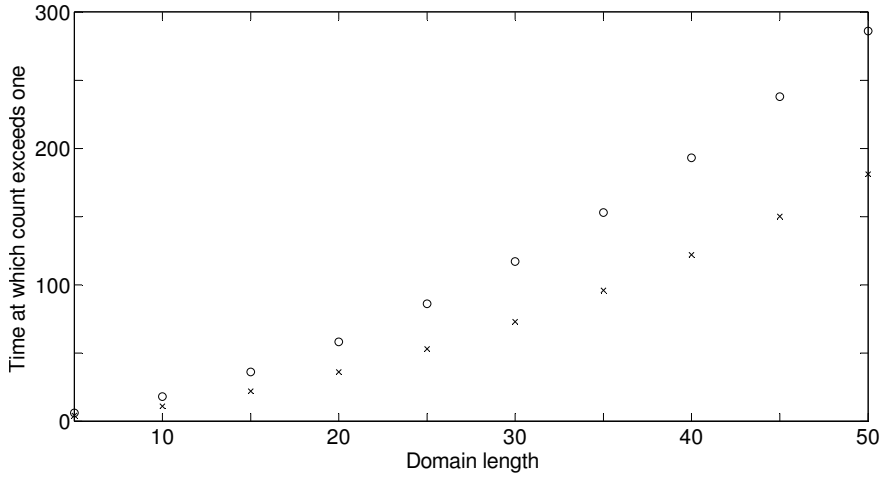


Figure 6: The time t_1 at which trap count exceed one against length L of domain. Crosses for diffusive (Dirichlet) forcing, circles for advective (Neumann) forcing. Parameters are: $D = 1$, $U_0 = 0$, $U_b = 10$ and $G = U_b/L$.

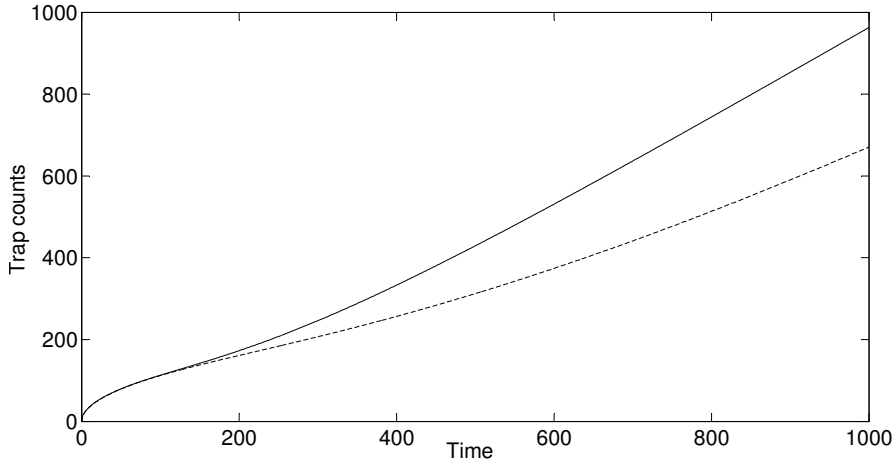


Figure 7: Trap count vs time as obtained from Eqs. (16) and (17). Solid and dashed curves correspond to the Dirichlet (diffusive) and Neumann (advective) boundary conditions, respectively. Parameters are: $D = 1$, $L = 45$, $U_0 = 10$, $U_b = 50$ and $G = U_b/L$.

(or in a faster growth of the trap count with time) depends on parameters U_b and G. For a given U_0, diffusive forcing is stronger than advective forcing for a large U_b (Fig. 7) but it gives way to the equivalent advective forcing when U_b becomes sufficiently small; see Fig. 8. Similarly, for a given U_b, the diffusive forcing is stronger for small U_0, but the advective forcing is stronger for large U_0. The value of U_b where the relative strength of the different forcing types interchanges depends on U_0. We also observe that the shape of the curves describing the trap count dynamics changes as well; from being convex for small values of U_b (Fig. 8) they become sigmoidal for large U_b (Fig. 7). Recall that for $U_0 = 0$ the generic shape of the curve is concave, cf. Fig. 5.

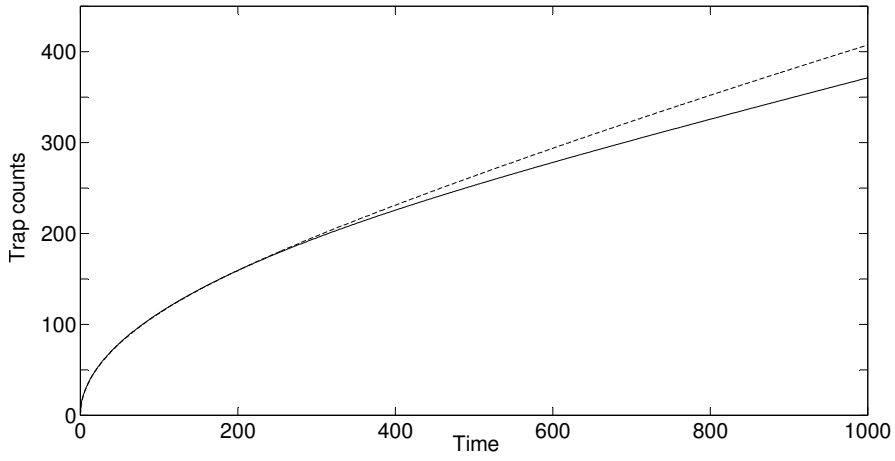


Figure 8: Trap count vs time as obtained from Eqs. (16) and (17) for $U_b = 10$ and $G = U_b/L$, other parameters are the same as in Fig. 7. Solid and dashed curves are for the Dirichlet and Neumann boundary conditions, respectively. Note that, contrary to the case shown in Fig. 7, diffusive forcing (Dirichlet) now results in lower trap count than advective forcing (Neumann).

The above results were obtained numerically. However, the relation between the trap count obtained for the two types of boundary forcing can also be addressed analytically, as is shown by the following theorem.

Theorem 1. *Consider the diffusion equation in the domain $0 < x < L$ with the condition at the left-hand side boundary as $u(0,t) = 0$ (as corresponds to the trap) and the initial condition $u(x,0) = u_0(x)$. Let $u_D(x,t)$ and $u_N(x,t)$ be the solutions obtained for the Dirichlet and Neumann boundary conditions at the right-hand side boundary, i.e. for $u(L,t) = U_b$ and $\partial u(L,t)/\partial x = G$, respectively. Consider the trap count $\Delta U(t)$ as defined by (5-6). Let the parameters be related as $U_b = GL$. Then the following statements hold:*

$$\textit{if } u_0(x) < Gx \qquad \textit{then} \qquad \Delta U(t)^{Dir} > \Delta U(t)^{Neu}, \tag{35}$$

$$\textit{if } u_0(x) > Gx \qquad \textit{then} \qquad \Delta U(t)^{Dir} < \Delta U(t)^{Neu}. \tag{36}$$

Proof. We first recall that, under the condition $U_b = GL$, both the Dirichlet and the Neumann problems have the same stationary solution $\bar{u}_D(x) = \bar{u}_N(x) = Gx$.

Let us consider condition (35), i.e. $u_0(x) < Gx$. Since both $u_0(x)$ and Gx satisfy the diffusion equation and the same boundary conditions, $u_0(0) = 0$ and $u_0(L) = GL = U_b$, by virtue of the comparison theorem for parabolic PDEs (also known as the monotonicity principle, cf. Volpert and Hudjaev 1985) we obtain that $u_D(x,t) \leq Gx$ for any $t > 0$ and all $0 \leq x \leq L$. Assume that there exist certain $x_* \in (0,L)$ and t_* such as $u_D(x_*,t_*) = Gx_*$. By virtue of the strong maximum principle, we then obtain that $u_D(x,t) \equiv Gx$ in $(0,L)$ for any $t > 0$. But $u_D(x,t)$ does not coincide with Gx at any finite t. Therefore (x_*,t_*) does not exists and hence $u_D(x,t) < Gx$ for all $0 < x < L$, in particular in the vicinity of $x = L$. Applying then Hopf's lemma (boundary point principle), we obtain that

$$\frac{\partial u_D(L,t)}{\partial x} > G\,. \tag{37}$$

Recall that G is the value of the gradient in the Neumann boundary condition; relation (37) therefore reads as

$$\frac{\partial u_D(L,t)}{\partial x} > \frac{\partial u_N(L,t)}{\partial x} \,. \tag{38}$$

We therefore obtain that $u_D(x,t)$ and $u_N(x,t)$ satisfy the same diffusion equation, the same initial condition $u_0(x)$ and the same left-hand side boundary condition $u_D(0,t) = u_N(0,t) = 0$ but have different value of the gradient at the right-hand side boundary as given by (38). Then, by virtue of the comparison theorem, $u_D(x,t) \geq u_N(x,t)$ for any $t > 0$ and all $0 \leq x \leq L$. Similarly to the above, let us assume that there exist $x^* \in (0,L)$ and t^* such as $u_D(x^*,t^*) = u_N(x^*,t^*)$. Applying the strong maximum principle, we obtain that $u_D(x,t) \equiv u_N(x,t)$ in $(0,L)$ for any $t > 0$, which is impossible. Therefore (x^*,t^*) does not exists and $u_D(x,t) > u_N(x,t)$ for all $0 < x < L$, in particular in the vicinity of $x = 0$. Applying now Hopf's lemma to the left-hand side boundary, we obtain that

$$(-1)\frac{\partial u_D(0,t)}{\partial x} < (-1)\frac{\partial u_N(0,t)}{\partial x} \,, \tag{39}$$

(where -1 accounts for different direction of the outwards normal vector) so that

$$\frac{\partial u_D(0,t)}{\partial x} > \frac{\partial u_N(0,t)}{\partial x} \,. \tag{40}$$

It then immediately follows from (5–6) that the population flux $j_0^{Dir} > j_0^{Neu}$ and hence the corresponding trap count $\Delta U(t)^{Dir} > \Delta U(t)^{Neu}$.

In the opposite case, i.e. where $u(x,0) > Gx$, a very similar argument (subject to signs reversed in relevant places) could be used to show that the flux into the trap is less with Dirichlet forcing than with Neumann forcing, i.e. the statement (36) holds. □

Obviously, the results shown in Figs. 5 and 8 correspond to the conditions (35) and (36), respectively. Note that the intermediate case where the initial condition $u_0(x)$ and the line Gx have intersection points in the interval $0 < x < L$ (i.e. when $0 < U_0 < U_b$ for the initial condition (7), cf. Fig. 7) is not addressed by the theorem.

5 The 2D problem

5.1 Model

Having established how our numerical solutions behave in a 1D system we now move on to the more complex, more realistic 2D case. The diffusion equation in the 2D case is written as

$$\frac{\partial u}{\partial t} = D\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right), \tag{41}$$

where $u(x,y,t)$ is the density of the pest insect population, D is the diffusion coefficient. The equation (41) is considered in the domain $\Omega = \{(x,y) : \ |x| < L_d, \ |y| < L_d\}$ where $2L_d$ thus gives the overall size of the domain.

We will focus on a case where a single trap is installed at the origin, i.e. in the center of the domain. Generally speaking, the shape of the trap can be different and that may significantly affect the complexity of the computational problem. However, it was shown by Petrovskii et al. (2012) that, in case the insects perform the Brownian motion (which is the pattern of individual movement corresponding to the diffusion equation, e.g. see Berg, 1983; Chorin and Hald, 2006), then the trap count depend on the perimeter of the trap but not on its shape. Correspondingly, in order to avoid unnecessary computational complexity, we consider the trap of a square shape, $S = \{(x, y) : \ |x| < l/2, \ |y| < l/2\}$, where l is thus the trap size.

Hence, the solution of the diffusion equation is sought in the following domain:

$$\Omega_s = \{(x, y) : \ l/2 < |x| < L_d, \ l/2 < |y| < L_d\}.$$

The equation (41) must be augmented by boundary conditions and initial conditions. Note that the computational domain Ω_s is not simply-connected (see Fig. 9 below) and has the external boundary, which we denote as $\partial\Omega$, and the internal boundary (i.e. the boundary of the trap S), which we denote as ∂S.

As well as in the 1D case, we assume that the trap is escape-proof and use the following condition at the internal boundary ∂S:

$$u(x, y, t) = 0 \quad \text{for any} \ \ (x, y) \in \partial S. \tag{42}$$

At the external boundary of the domain, aiming at taking into account different types of boundary forcing, we consider the condition in the following general form:

$$k_1 u(x, y, t) + k_2 \frac{\partial u(x, y, t)}{\partial \mathbf{n}} = k_3 \quad \text{for any} \ \ (x, y) \in \partial\Omega, \tag{43}$$

where $\mathbf{n}$ is the outward unit normal vector to $\partial\Omega$, and $k_{1,2,3}$ are parameters. Obviously, for a different choice of $k_{1,2,3}$ the boundary condition (43) can describe either diffusive or advective forcing, that is

$$\text{(a)} \quad u(x, y, t) = U_b \quad \text{at} \ \partial\Omega \qquad \text{or} \qquad \text{(b)} \quad \frac{\partial u(x, y, t)}{\partial \mathbf{n}} = G \quad \text{at} \ \partial\Omega \tag{44}$$

(obtained for $k_1 = 1, \ k_2 = 0, \ k_3 = U_b$ and $k_1 = 0, \ k_2 = 1, \ k_3 = G$, respectively). The no-flux case corresponding to the impenetrable boundary is obtained for $k_1 = k_3 = 0$ when (43) turns into

$$\frac{\partial u(x, y, t)}{\partial \mathbf{n}} = 0 \quad \text{at} \ \partial\Omega. \tag{45}$$

When condition (45) is applied at every point of the external boundary $\partial\Omega$, the number of insects in the domain can only change as a result of trapping, i.e. because of the flux through the internal boundary ∂S. As a relevant alternative, for $k_2 = k_3 = 0$, Eq. (43) described an 'absorbing' boundary with zero population density:

$$u(x, y, t) = 0 \quad \text{at} \ \partial\Omega, \tag{46}$$

which can be used, for instance, to describe the emigration of the insects into a hostile environment outside of the field.

Note that, in the 2D system, different conditions can be used at different parts of the external boundary if needed to account for the given ecological scenario, e.g. if insect immigration takes place through one edge only. Obviously, this more general case can still be described by (43) with $k_{1,2,3}$ being piecewise constant functions along the domain boundary $\partial\Omega$.

With regard to the initial condition, as our main focus is on the effect of the boundary forcing, we consider the baseline case of the uniform population distribution:

$$u(x, y, 0) = U_0 > 0 \text{ for any } (x, y) \in \Omega_s, \tag{47}$$

where U_0 is a parameter.

For convenience, we now introduce dimensionless variables by considering the length l of the trap side as a characteristic spatial scale in the problem so that $\tilde{x} = x/l,\ \tilde{y} = y/l$. We then have a trap as the unit square $\tilde{S} = \{(\tilde{x}, \tilde{y}) :\ |\tilde{x}| < 1/2,\ |\tilde{y}| < 1/2\}$ installed at the origin of the square domain $\tilde{\Omega}$ with the linear size scaled as $L = L_d/l$. In the below, the tildes are omitted for the sake of notations simplicity.

5.2 Discretization of the 2D problem

We use a finite difference method for numerical solution of the problem (41-43,47). A two-dimensional Cartesian grid $\mathcal{G}$ is defined in the domain Ω_s as a tensor product of the two one-dimensional grids. We first generate a 1D grid in the x -direction with grid nodes given by $x_1 = -L$, $x_{i+1} = x_i + h$, $i = 1, \ldots, 2Nm$, and $x_{2Nm+1} = L$, where $h = 1/(2m)$ for integer

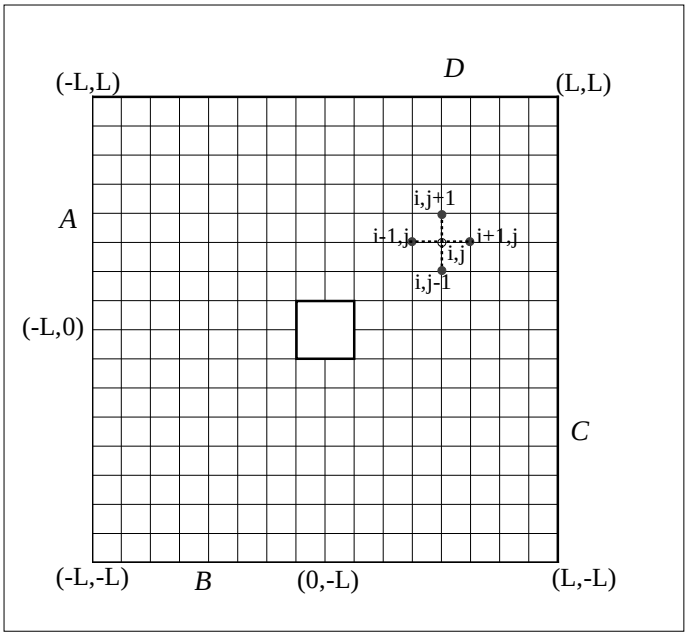


Figure 9: A computational grid in the domain Ω_s. The trap is placed at the origin $(0, 0)$. The discretization of the equation (41) at the grid node n_{ij} requires the values of the function $u(x, y, t)$ at the nodes n_{ij}, n_{i-1j}, n_{i+1j}, n_{ij-1} and n_{ij+1}. Letters $A,\ B,\ C$ and D denote the domain edges where different boundary conditions can be used; see section 6.

parameters m and N used in the problem. Similarly, a 1D grid in the y-direction is $y_1 = -L$, $y_{j+1} = x_j + h$, $j = 1, \ldots, 2Nm$. It follows from the definition of 1D grids that the trap boundaries $x_{i_I} = -1/2$, $i_I = (N-1)m + 1$, $x_{i_{II}} = 1/2$, $i_{II} = (N+1)m + 1$, $y_{j_I} = -1/2$, $j_I = (N-1)m + 1$, and $y_{j_{II}} = 1/2$, $j_{II} = (N+1)m + 1$ are accurately represented on the 2D grid and a grid node is placed at each corner of the trap (see Fig. 9). It also is worth noting here that we can take an advantage from the problem symmetry and to solve it in the sub-domain $x > 0, y > 0$ only, if the initial condition (47) is employed. However, we choose to solve the problem in the entire domain Ω_s as we bear in mind further asymmetric cases, *e.g* application of an asymmetric initial distribution, installation of several traps in the domain, etc.

Let $u(x, y, t_n)$ be the solution to the problem (41-43,47) at the fixed time $t = t_n$ and $u(x, y, t_{n+1})$ be the solution at the fixed time $t = t_{n+1}$, where $t_{n+1} = t_n + \tau$, $\tau > 0$. Again, we use a finite difference discretization of (41-43,47) in order to compute the solution $u(x, y, t_{n+1})$ from the solution $u(x, y, t_n)$. The discretization scheme for the equation (41) is written as

$$\frac{1}{\tau}(u_{ij}^{n+1} - u_{ij}^n) = (\Lambda_1 + \Lambda_2)[u_{ij}^n], \tag{48}$$

where we use the notation $u_{ij}^n \equiv u(x_i, y_j, t_n)$ and $u_{ij}^{n+1} \equiv u(x_i, y_j, t_{n+1})$. Discrete spatial operators Λ_1 and Λ_2 act upon any two-dimensional function $v_{ij} \equiv v(x_i, y_j)$ as follows

$$\Lambda_1[v_{ij}] = \frac{D}{h^2}(v_{i+1,j} - 2v_{ij} + v_{i-1,j}), \quad \Lambda_2[v_{ij}] = \frac{D}{h^2}(v_{i,j+1} - 2v_{ij} + v_{i,j-1}). \tag{49}$$

As in the 1D case, the discretization (48) is an explicit scheme. Again, our choice of the scheme is advocated by simplicity of computations. It is important to note here that the explicit scheme allows us to avoid inversion of the discretization matrix that is required when implicit schemes are used. While in the current problem the structure of the discretization matrix is relatively simple, our future work should be to investigate several traps installed at arbitrary locations of the domain Ω. Such geometry of the domain may in turn result in a complex structure of a discretization matrix, thus we want to investigate a simple explicit 2D scheme first and move to a more sophisticated discretization scheme if and only if the explicit discretization has been proved to be unacceptable for our purpose.

The equations (48) are solved for $i = 2, \ldots, i_I - 1, i_{II} + 1, \ldots, 2Nm$ and $j = 2, \ldots, j_I - 1, i_{II} + 1, \ldots, 2Nm$. The discretization of boundary conditions at the external boundaries is based on the results of the 1D case for which a detailed discussion has been provided. Consider, for example, the boundary $x = 0$ and the boundary condition (45). We have the following approximation

$$\frac{h^2}{\tau D}\left(u_{1,j}^{n+1} - u_{1,j}^n\right) + 4u_{1,j+1}^n - 2u_{2,j}^n - u_{1,j+1}^n - u_{1,j-1}^n = 0, \text{ for } j = 2, \ldots, 2Nm. \tag{50}$$

The discrete boundary conditions at the rest of external boundary have a similar structure.

The condition at the trap boundary at any time $t_{n+1} > 0$ is

$$u_{ij}^{n+1} = 0, \tag{51}$$

for $i = i_I, j = j_I, \ldots, j_{II}$ (the left boundary of the trap) and $i = i_{II}, j = j_I, \ldots, j_{II}$ (the right boundary of the trap). The condition (51) is also imposed for $j = j_I, i = i_I, \ldots, i_{II}$, and $j = j_{II}, i = i_I, \ldots, i_{II}$ (the bottom and top boundaries of the trap).

The discretization of the initial condition (47) is

$$u_{ij}^0 = U_0, \quad i = 1, 2, \ldots, 2Nm + 1, \ j = 1, 2, \ldots, 2Nm + 1. \tag{52}$$

The idea behind trap count approximation in the 2D case is similar to our approach in the 1D problem. Several specific approximation issues arise because of the geometry of a 2D domain and we refer the interested reader to the appendix B where those issues are discussed.

Validation of the 2D discretization. In the 2D case, we cannot compare a numerical solution with the exact solution to the problem because the exact solution is not available. Therefore, we look for alternative ways to validate our discretization scheme. One approach can be to consider the numerical integration of the density function and compare it with the results obtained by the flux calculation (but see the paragraphs after Eq. (28) in section 3 for a discussion of arising issues). Namely, we consider the test case where the boundary condition (45) is employed in the problem. For the condition (45) it is possible to obtain the number of insects in the trap by direct integration over the domain Ω_s. The integral

$$U^{n+1} = \int\int_{\Omega_s} u(x, y, t^{n+1}) dxdy \tag{53}$$

gives us the number of insects left in the domain Ω_s at time $t = t^{n+1}$. The trapezoidal rule of integration can be readily applied for integration over a two-dimensional domain (10). We then compute the integral

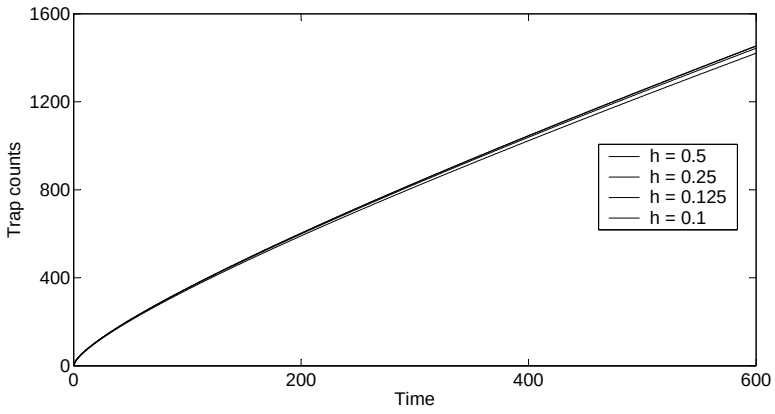
$$U^0 = \int\int_{\Omega_s} u(x, y, 0) dxdy \tag{54}$$

for the initial distribution $u(x, y, 0)$ and the difference

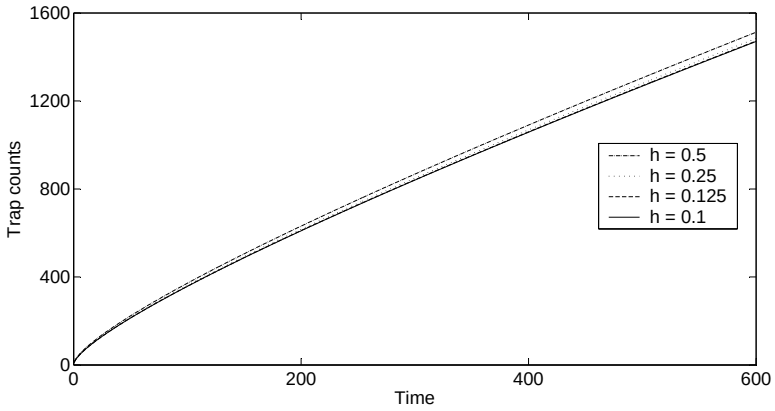
$$\delta U^{n+1} = U^{n+1} - U^0 \tag{55}$$

will provide us with the number of insects in the trap at time t^{n+1}. The trap count obtained by direct integration is then compared with the trap count obtained by the flux calculation across the trap boundary as discussed in the previous paragraphs.

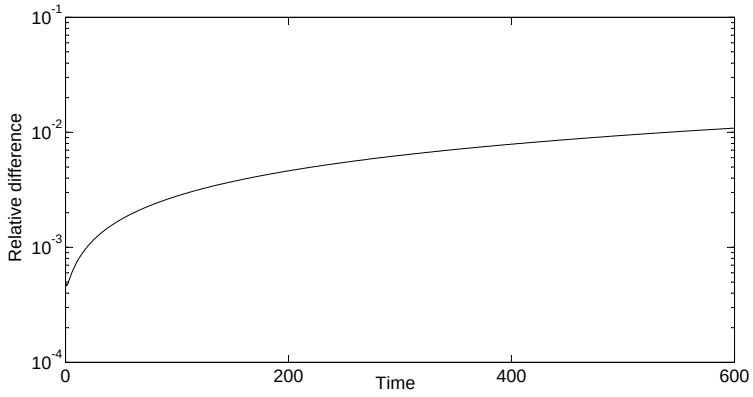
The validation tests for the 2D problem are shown in Fig.10. The convergence test is presented in Fig.10a where cumulated trap count has been computed on a sequence of refined spatial grids for time $t \in [0, 600]$. The problem parameters are $L = 40$, $D = 0.1$, $U_0 = 10$, $l = 1.0$ (trap size). It can be seen from the figure that transition from a computational grid with the grid step size $h = 0.5$ to a finer grid with $h = 0.1$ does not make significant difference to the results. The same conclusion can be derived from the consideration of graph in Fig.10b where trap count on a sequence of spatial grids has been obtained as a result of direct integration (55). Hence, we compare the trap count obtained by flux computation on the finest grid with $h = 0.1$ with the trap count obtained on the same grid by direct integration. The relative difference between the two solutions is shown in Fig.10c. It can be seen from the figure that while the relative error gets bigger as the time progresses the two methods still remain in a very good agreement for a whole time interval where we are interested in the solution. Hence we believe that our discretization scheme meets the accuracy requirements and we intend to use it for further study of ecologically relevant test cases.



(a)



(b)



(c)

Figure 10: Validation tests for the 2D problem. (a) Convergence of the trap count obtained by calculating the population flux through the trap boundary, see Eqs. (62-64); (b) Convergence of the numerical integration method, see Eqs. (53-55); (c) Comparison between the two approaches on the finest grid $h = 0.1$ (showing the difference between the two solutions versus time).

6 Numerical results for the 2D problem

There are several aspects that make the 2D problem different from the corresponding 1D problem. To mention just a few, firstly, the boundary condition on the field boundary (i.e. on the external boundary of the computational domain) can be a combination of different types applied at different parts of the boundary. And secondly, the 2D problem has several spatial scales such the trap size l, the field size L and the distance d between the trap and the closest boundary of the field through which immigration can occur.

In this section, we present the simulation results obtained for 2D domains with various geometries. We consider the case where pest immigration takes place through one edge of a square-shaped field, say edge A (see Fig. 9). The immigration is described by considering either diffusive or advective forcing, with the boundary conditions (44a) or (44b) respectively, or by their combination as given by the more general mixed-type boundary condition (43). The edges on the sides of the domain, i.e. B and D, are assumed to be impenetrable boundaries and hence the Neumann 'no-flux' condition (45) is used there. The edge C opposite to the forcing boundary A can be either impenetrable, with the no-flux condition (45), or absorbing, with the condition (46).

With regard to the spatial arrangements, we consider that, in the square field, the square trap of a fixed size $l = 1$ (recall out choice of dimensionless variables) is either placed in the center of the square domain (see Fig. 9) or off center, closer to the forcing boundary. We consider three different cases: Field 1 is a small field ($2L = 9$) with the central trap (so that $d = 4$), Field 2 is a large field ($2L = 19$) with the central trap (so that $d = 9$), and Field 3 is a large field ($2L = 19$) with an off center trap (so that $d = 4$).

We begin with the case when the field edge C is an impenetrable boundary and the immigration through the edge A is described by diffusive forcing. Figure 11 shows the trap count obtained when Field 2 is subjected to diffusive boundary forcing (solid curve for $U_b = 0$, dashed curve for

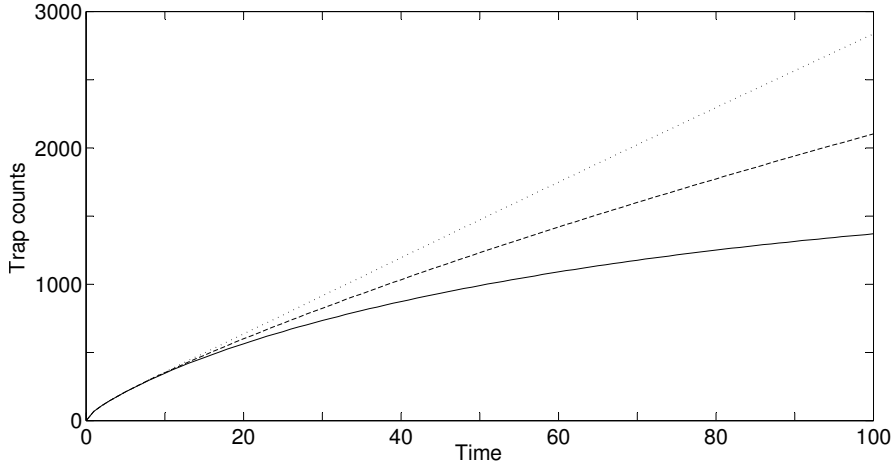


Figure 11: Trap count obtained from numerical solution of the 2D problem in Field 2 ($L = 19$ and $d = 9$) with other parameters as $D = 1$ and $U_0 = 10$ and diffusive forcing with $U_b = 0$ (solid curve), $U_b = 10$ (dashed curve) and $U_b = 20$ (dotted curve).

$U_b = 10$, dotted curve for $U_b = 20$). The results are intuitively clear and qualitatively similar to those obtained in the 1D case as more intense forcing is expected to lead to larger trap count. Also, we observe that the trap count obtained for different U_b only becomes different after a certain time, i.e. when the perturbation introduced by the boundary forcing reaches the trap. However, we note that, although the shape of the curves is similar to that observed in the 1D case (cf. Figs. 7 and 8), the actual value of the trap count is more than ten times larger in the 2D case.

The effect of proximity to the field boundary is revealed in Fig. 12. The top panel of Fig. 12 shows the trap count for the special case $U_b = 0$, i.e. in the case when the forcing boundary acts as a sink rather than a source. It is readily seen that d is a controlling parameter as the trap count are lower in Field 3 (dotted curve) than in Field 2 (dashed curve). For the cases with the same d (cf. solid and dotted curves), it is the field size that determines the rate of the trap count growth; the larger the field size the larger the trap count is.

However, the situation is different and somewhat counter-intuitive when U_b is large; see Fig. 12, middle, obtained for $U_b = 20$. In this case, the boundary forcing becomes the main factor affecting the trap count. Correspondingly, the effect of distance d on the trap count becomes more prominent and different from the above. Although the highest rate of the trap count growth is still reached for the combination "small d – large L" (dotted curve), small d may actually result in larger trap count when L is small; compare the solid and dashed curves in Fig. 12, middle. In the intermediate case when the magnitude of the boundary forcing (as described by U_b) is as the same as the effect of the initial population (as described by U_0), see the bottom panel of Fig. 12, the pattern observed in the trap count growth can be regarded as a competition between the two factors: the field size is more important (hence resulting in larger trap count) during the early stage, i.e. until the perturbation from the forcing boundary reaches the trap, afterwards the proximity of the trap to the forcing boundary becomes the main controlling parameter. This interplay between the two factors results in the intersect of the solid and dashed curves seen in Fig. 12, bottom. Once again, we notice that the trap count is much larger in the 2D case (more than an order of magnitude) compared to what was observed in the corresponding 1D case.

We now consider how much the rate of the trap count growth can be affected by changing the type of the boundary forcing. Figure 13 shows the trap count obtained in case of advective forcing, see Eq. (44b), in case the opposite boundary of the domain is either impenetrable (top) or absorbing (bottom), as described by Eqs. (45) and (46), respectively. Clearly, the type of the boundary condition on the opposite boundary has little effect on the shape of the curves, although it does affect the value of the trap count, which is about 50% larger in case of the impenetrable boundary. Comparing the results of Fig. 13 with those of Fig. 12 (bottom), we observe that, as well as in the 1D case, diffusive forcing results in trap count several times larger than the trap count for corresponding 'equivalent' advective forcing (see the lines below Eq. 32), although we mention here that, rigorously speaking, the conditions of equivalency do not apply in the 2D case.

Finally, in order to provide a somewhat broader view of the system's properties, we consider the case where the forcing at the field boundary is of a mixed type as described by the boundary condition (43). Figure 14 shows the simulation results obtained for the spatial arrangement of Field 2 with the homogeneous initial condition $U_0 = 10$ for the mixed forcing (43) with $k_1 = 1$, $k_3 = 10$ (Fig. 14a), $k_3 = 20$ (Fig. 14b) and the values of $k_2 = 0,\ 0.1,\ 0.2, \ldots,\ 2.0$ (curves top to bottom, respectively). Interestingly, in this case the trap count roughly follows the pattern observed in the

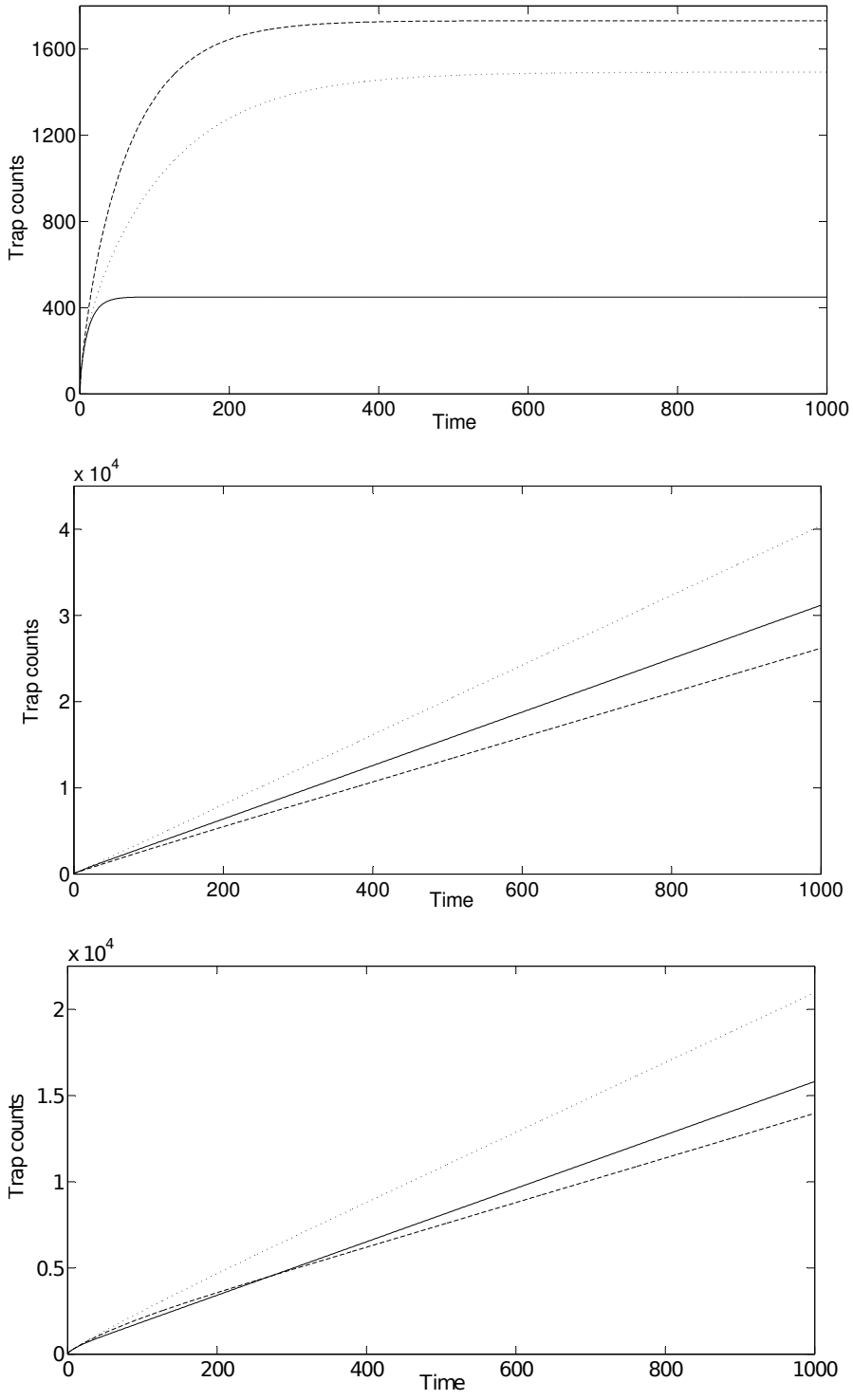


Figure 12: Trap count obtained from numerical solution of the 2D problem in different spatial arrangements (solid curve for Field 1, dashed curve for Field 2 and dotted curve for Field 3, see details in the text) and for different value of diffusive boundary forcing: top for $U_b = 0$, middle for $U_b = 20$ and bottom for $U_b = 10$. Other parameters are $D = 1$ and $U_0 = 10$.

case of diffusive forcing. In fact, for each of the two cases shown in Fig. 14, the whole family of trap count curves lie between the two curves obtained for purely diffusive forcing (shown by red colour) where the upper bound obviously corresponds to the case $U_b = k_3$ and the lower bound is obtained for $U_b \approx 0.7k_3$ (where the coefficient 0.7 is found empirically). The presence of the density gradient in the boundary condition does not contribute much to the trap count dynamics. Therefore, in agreement with our results shown above, we conclude that the diffusive component of the forcing has much stronger effect on the trap count than the advective component.

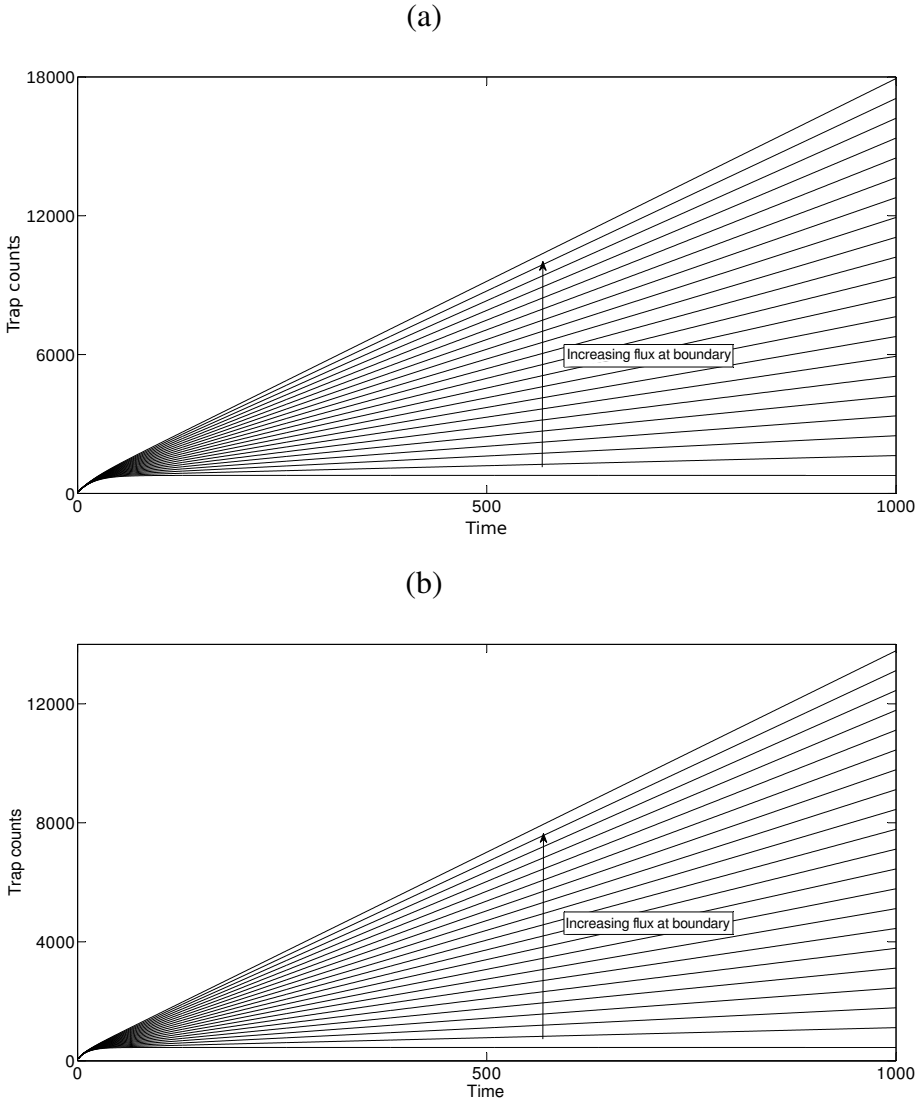


Figure 13: Trap count obtained from numerical solution of the 2D problem in the spatial configuration of Field 1 with advective forcing (Neumann boundary condition) at the forcing boundary and for different boundary condition at the opposite boundary: (a) for no-flux condition, Eq. (45), (b) for the absorbing boundary, Eq. (46). In each panel, the value of the boundary gradient G varies from 0 to 20, bottom to top, respectively.

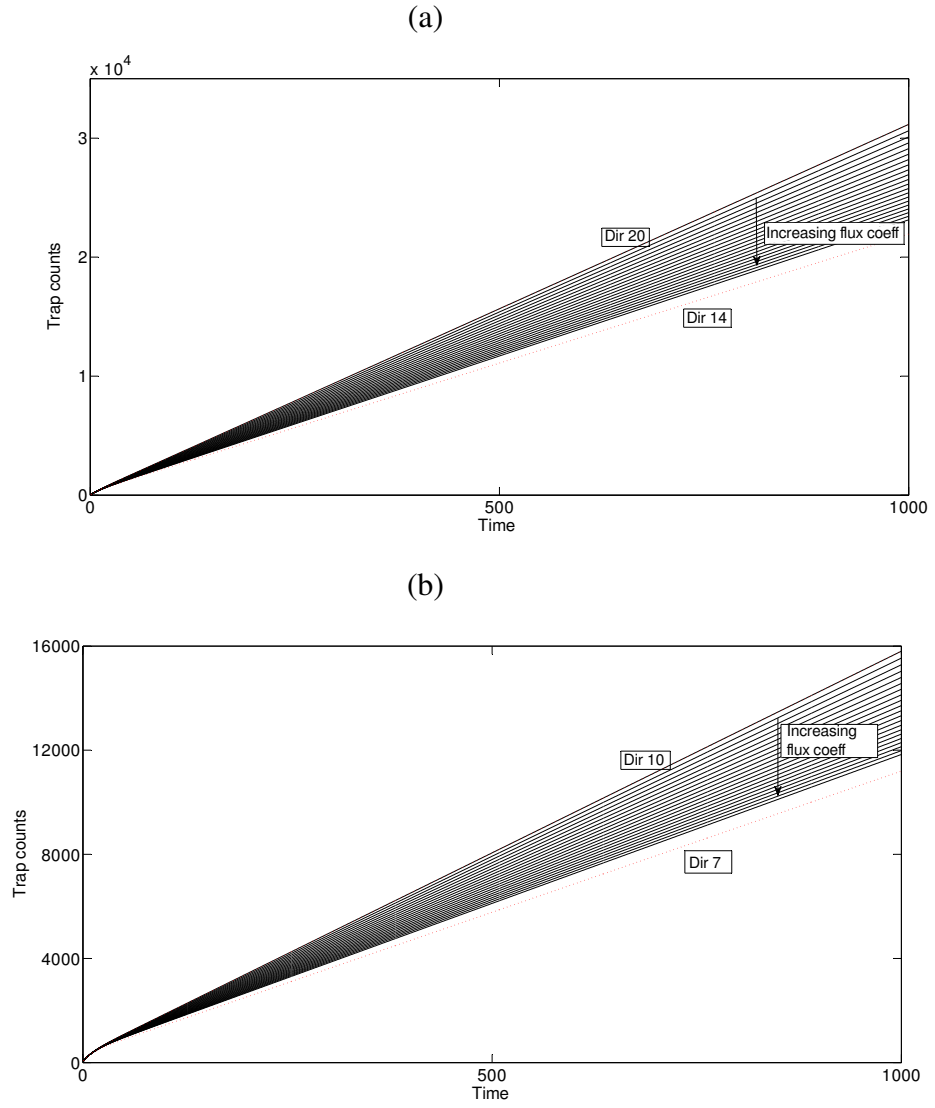


Figure 14: Trap count obtained from numerical solution of the 2D problem in the spatial configuration of Field 2 with mixed forcing at the forcing boundary as given by the mixed boundary condition (43), top for (a) $k_3 = 10$ and (b) $k_3 = 20$. In each panel, $k_1 = 1$ and the value of k_2 varies from 0 to 20, bottom to top, respectively. Red curves show the upper and lower bounds for the trap count obtained for 'equivalent' diffusive forcing (Dirichlet boundary conditions); see details in the text.

7 Discussion and Concluding Remarks

Understanding of trap count is an important component of pest insect monitoring. Control measures are likely to be more efficient when the process behind the trap count increase is correctly identified. In this paper, we have considered how trap count can be used to monitor the insect pest population growth when the growth is occurring due to the immigration through the habitat boundary. Indeed, short- and long-distance dispersal is known to be a common phenomenon for

many insect species (in particular for flying/wind-borne species, cf. Compton 2002) and it can bring severe problems for farmers and pest control specialists (e.g. Northing 2009).

The purpose of this paper is threefold. Firstly, we want to draw the attention to the important and largely overlooked (by mathematical ecologists) problem of the trap count interpretation with the ultimate purpose to identify the population dynamics resulting in given trap count and to evaluate the population abundance. Traps are routinely used in insect monitoring as well as in general insect studies and the need for a relevant theory has long been recognized (e.g. Byers 1993; Perry 1996; Thomas et al. 1998). Yet a consistent mathematical theory allowing for trap count modelling and simulation is missing, although some attempts to develop such theory have recently been made (Petrovskii et al. 2012, 2014; Petrovskaya et al. 2012; Petrovskaya and Embleton 2013). Secondly, our aim is to provide a sufficiently accurate, consistent and reliable "ready-to-use" computational algorithm that can be used for understanding trap count across a variety of possible ecological scenarios and applications, a roadmap for potential users of the ideas, tools and methods of the computational ecology (Petrovskii and Petrovskaya 2012). And thirdly, using the algorithms revisited and/or developed in this paper, we want to make an insight into the problem of pest insect monitoring subject to different immigration patterns.

With regard to the numerical accuracy, one of our more specific goals here is to emphasize that an adequate choice of the numerical method is essential. We have shown that the commonly used linear approximation of the diffusion fluxes (cf. Roache 1985; Morton and Mayers 1994; Thomas 1995) can lead to results of unacceptably low accuracy compared to the quadratic approximation on the same numerical grid; see Tables 1 and 2. The linear approximation of the diffusion flux at the trap boundary leads to the relative error in the calculated trap count which is consistently 3-5 times larger (more than 10 times larger on some grids) than the corresponding error in the case of quadratic approximation (Table 1). Approximation of the flux at the external 'forcing' boundary can have even bigger effect (Table 2); here the relative error of the linear approximation can be $10^2 - 10^3$ times larger than the error induced by the quadratic approximation!

Nodes per unit length	3	5	9	11	21
Linear ($\times 10^{-3}$)	112	64.4	33.5	20.7	9.59
Quadratic ($\times 10^{-3}$)	32.8	10.4	2.70	7.21	5.00

Table 1: Maximum relative error (27) obtained in the 1D system for linear and quadratic flux approximation at the trap boundary. Parameters are: $D = 1.0, L = 49.5, U_0 = 10$.

Nodes per unit length	3	5	9	11	21
Linear ($\times 10^{-4}$)	359	183	92.4	74.1	37.2
Quadratic ($\times 10^{-4}$)	0.949	0.288	0.110	0.194	0.100

Table 2: Maximum relative error (27) obtained in the 1D system for linear and quadratic flux approximation at the external boundary. Forcing is described by the Neumann boundary condition (9) with $G = 0.5$, other parameters are $D = 1.0, L = 49.5, U_0 = 0$.

In our approach, in order to solve the diffusion equation and to calculate the diffusion fluxes, we used the finite difference method. We mention here that, in principle, this is not the only possible option. One alternative is the method of lines. Another alternative numerical approach to diffusion equation could be based on the finite elements method. Indeed, this method has been applied successfully to several ecological and biological problems (Gurvie and Trenchea 2007; Gurvie et al. 2010) and it is known to be particularly efficient for problems with curvilinear geometry. However, for a domain with rectangular geometry, this technique seems to be excessive; being more elaborate and hence more expensive in terms of the code development, it does not provide any new insights and, on a given numerical grid, does not provide a higher accuracy either.

With regard to our use of the explicit scheme (e.g. as given by Eqs. (48–49) in the 2D case), we mention that the obvious advantage of its implicit counterpart is unconditional stability. However, this advantage of being able to choose a larger value of the time-step τ does not appear to be essential for our problem. The spatiotemporal accuracy of the considered scheme is $O(h^2 + \tau)$. It is important to have a second order approximation as we have demonstrated for the spatial terms, cf. Tables 1 and 2. It means that, in order to reach the required accuracy, τ should be chosen on the order of h^2. Therefore, the restriction on τ comes from the accuracy requirement rather than from the stability condition. On the other hand, the computational simplicity of the explicit scheme is important as our ultimate aim is to create a simple and practical computational approach that could eventually be used by a broad interdisciplinary community.

In our numerical simulations, see Figs. 5–8 and 11–14, we have followed the trap count dynamics over a long time, until it reaches the large-time asympotics which is determined solely by the boundary forcing, cf. Eqs. (31–32). This gives the overall view of the solution properties and helps to better understand the trap count behavior, e.g. to estimate the time when the effect of the initial condition would give up to the effect of the boundary forcing. However, for the actual ecological problem of insect pest monitoring, the large-time asymptotics is largely irrelevant. A realistic trap count dynamics is essentially transient. Indeed, the goal of the pest control is not only to detect the pest at a given location but, importantly, to do it as early as possible. Trap data are often used as a part of the early-warning system in order to apply control measures before the growing pest population can bring any significant damage to the crops. Therefore, the large time asymptotics as predicted by our model (which does not take into account control measures) can hardly ever be observed in reality. The human intervention (e.g. application of pesticides) would change the system dynamics considerably by reducing almost instantly the pest density to a much lower value. The subsequent trap count dynamics would then reflect the transient stage resulting from this new population density, until the growing trap count would inflict another round of control measures, and so on.

With regard to the insect immigration through the habitat boundary, we mention here that the effect of boundary forcing on the system's dynamics is a challenging problem in ecology as it can have a complicated and sometimes counter-intuitive effect on the population dynamics and population abundance (Fagan et al. 1999, Sherratt and Smith 2008; Smith et al. 2008). Note that the whole issue as to what can be a proper description of the habitat boundary – which acts as an interface between the given habitat and its surroundings – is complex and controversial. Contrary to similar problems in physics and chemistry, the width of this interface is often not small compared with the size of the habitat and may have its own structure (cf. Kovacs-Hostyanszki et al., 2013)

and that makes application of standard techniques questionable. In fact, it is often not clear even where the exact position of the boundary is and/or what its exact shape is (cf. Fagan et al. 1999). Also, the boundary is expected to affect the movement behavior of the animals (e.g. insects) but, yet again, it is not always clear what exactly the effect is (Bearup and Petrovskii 2015).

Altogether, these various sources of uncertainty make the question about the 'precise' mathematical formulation of the boundary condition rather senseless. Instead, one should rather check how sensitive the properties of the system are with respect to the choice of the boundary condition. The high uncertainty and insufficient knowledge about the processes going on at the interface makes it necessary to check how different can be the predictions obtained from a model when different boundary conditions are used, e.g. Dirichlet or Neumann types, and this is where our study is going to contribute. In particular, we have shown that the trap count obtained in the 2D system for three different types of forcing, i.e. diffusive (Dirichlet), advective (Neumann) and mixed (Robin), exhibits qualitatively similar patterns. However, we have also shown that, somewhat counter-intuitively, diffusive forcing results in larger trap count than advective forcing. This observation may have an important message for the insect pest management: as larger trap count is usually associated with a larger population density, misidentification of the pest immigration pattern can result in an unjustified application of pesticides, something that the IPM specialists would definitely like to avoid.

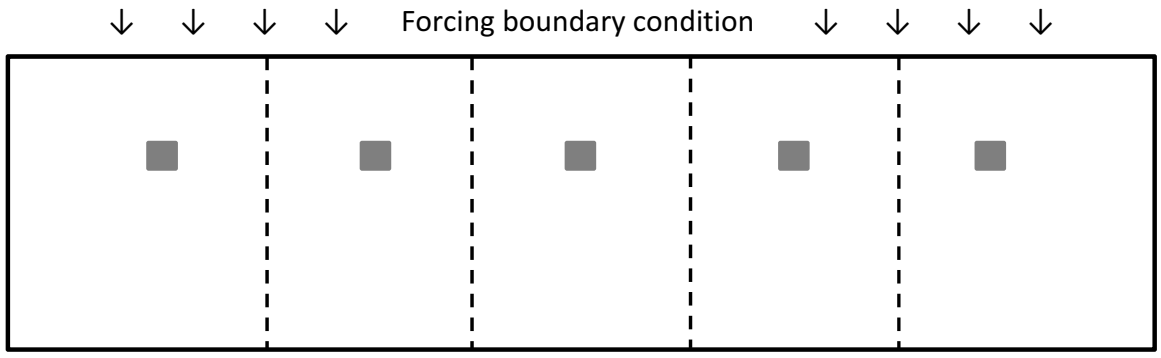


Figure 15: A sketch of multiple trapping with the spatial arrangements making possible to 'split' the system into several individual subdomains, each of them having one trap only. Solid line shows the boundary of the field, grey squares show the position of the traps, dashed vertical lines indicate the position of the virtual boundaries where the no-flux conditions can be applied due to the symmetry of the problem.

In conclusion, we mention that, in this paper, we have considered the case of a single trap installed in the monitored field. However, our results can be applicable, at least partially, to a more general case of multiple traps if the traps are installed on a line along the field boundary; see Fig. 15. Indeed, assuming the spatially homogeneous initial population distribution and the uniform boundary forcing (i.e. that the coefficients $k_{1,2,3}$ have the same value everywhere along the boundary), the mathematical problem obviously attains symmetrical properties so that the whole field can be split to several mutually independent subdomains or 'cells', each of them being described by our model.

Acknowledgements

The authors are thankful to Rod Blackshaw for stimulating discussions. Useful suggestions from two anonymous reviewers are much appreciated. This work was supported by The Leverhulme Trust through grant F/00-568/X.

Appendix A: Accurate calculation of the 1D flux.

Polynomial approximation of the density function. Our aim in this section is to construct an accurate approximation of the flux function $j(x)$ in the formula (6). Let us approximate the density $u(x)$ by a quadratic polynomial, i.e. $u(x) \approx p_k(x) = a_0 + a_1 x + a_2 x^2$. We then have $j(x) \approx D \left| \dfrac{dp(x)}{dx} \right|$ and therefore the flux at the trap boundary $x = 0$ is approximated as

$$j(0) \approx D|a_1|. \tag{56}$$

The polynomial coefficient a_1 in the flux approximation (56) is defined from the conditions $p(x_i) = u_i$, $i = 1, 2, 3$, where the function values $u_i \equiv u(x_i)$ are taken at corresponding grid nodes x_i. We have

$$p(0) = a_0 = u_1, \quad p(h) = a_0 + a_1 h + a_2 h^2 = u_2, \quad p(2h) = a_0 + 2a_1 h + 4a_2 h^2 = u_3. \tag{57}$$

Solving these equations and taking into account that $u_1 = 0$ at the trap boundary we arrive at $a_1 = \dfrac{4u_2 - u_3}{2h}$. Hence the approximation of the flux is given by $j(0) \approx \dfrac{D}{2h}|4u_2 - u_3|$.

The order of approximation. Let us show that the quadratic approximation (56) of flux is second order accurate, i.e. the approximation (25) has the error $O(h^2)$. Consider a one-dimensional function $v(x)$ on a uniform computational grid with the grid step size h. Let the function $v(x)$ be defined at points x_i, $x_{i+1} = x_i + h$ and $x_{i+2} = x_i + 2h$. The function approximation by a quadratic polynomial passing through x_i, x_{i+1}, x_{i+2} gives us the following value of the derivative at the point x_i

$$\frac{dv(x_i)}{dx} \approx \frac{4v(x_{i+1}) - v(x_{i+2})}{2h} \tag{58}$$

where we require that $v_i = 0$. The Taylor series expansion of the function $v(x)$ around the point x_i is

$$v(x_{i+1}) = v(x_i) + h\frac{dv(x_i)}{dx} + \frac{1}{2}h^2\frac{d^2v(x_i)}{dx^2} + \frac{1}{6}h^3\frac{d^3v(\eta)}{dx^3},$$

$$v(x_{i+2}) = v(x_i) + 2h\frac{dv(x_i)}{dx} + \frac{1}{2}(2h)^2\frac{d^2v(x_i)}{dx^2} + \frac{1}{6}(2h)^3\frac{d^3v(\mu)}{dx^3},$$

where $\eta \in [x_i, x_{i+1}]$ and $\mu \in [x_i, x_{i+2}]$.

Substituting the above expressions in the approximation $\dfrac{4v(x_{i+1}) - v(x_{i+2})}{2h}$ and taking into account that $v(x_i) = 0$ we arrive at

$$\frac{4v(x_{i+1}) - v(x_{i+2})}{2h} = \frac{dv(x_i)}{dx} + \frac{1}{3}h^2\left(\frac{d^3v(\eta)}{dx^3} - 2\frac{d^3v(\mu)}{dx^3}\right). \tag{59}$$

In other words, we have

$$\frac{dv(x_i)}{dx} = \frac{4v_{i+1} - v_{i+2}}{2h} + O(h^2), \tag{60}$$

where $= v_i \equiv v(x_i)$. Let us note that a well-known upwind approximation of the derivative at point x_i is first order accurate,

$$\frac{dv(x_i)}{dx} = \frac{v_{i+1} - v_i}{h} + O(h), \tag{61}$$

and is therefore not consistent with the accuracy of the scheme (18).

Appendix B: The issues of flux approximation in a 2D domain.

Flux approximation at the trap boundaries. Let the solution $u(x, y, t_{n+1})$ be computed at time t_{n+1} over a computational grid generated in the domain Ω_s. In the following discussion of 2D flux computation we omit the notation t_{n+1} for the sake of convenience and consider a spatial distribution $u(x, y)$. The flux density $j(x, y)$ at point (x, y) is given by the directional derivative:

$$j(x, y) = \left| \frac{du}{d\mathbf{n}}(x, y) \right|, \tag{62}$$

where $\mathbf{n}$ is the outward unit normal vector along the trap boundary. Hence the flux density computation requires approximation of the derivatives $\dfrac{\partial u(x, y)}{\partial x}$ or $\dfrac{\partial u(x, y)}{\partial y}$ at grid nodes that belong to the trap boundary. Similarly to the 1D case, the accuracy of this approximation should be consistent with the accuracy of finite difference approximation (48). As we have discussed in Appendix A, such accuracy of approximation can be achieved by constructing a quadratic polynomial $u(x, y) \approx p_k(s) = a_0^k + a_1^k s + a_2^k s^2$, where $k = 1, 2, 3, 4$ is the number of the trap edge and we use the numeration in the counterclockwise direction from the left boundary of the trap. The distance s is measured along the direction of the outward normal vector to the trap edge k as $s = x$, if the left ($k = 1$) and the right ($k = 3$) boundaries of the trap are considered, $s = y$ at the bottom ($k = 2$) and top ($k = 4$) boundary of the trap. We also require $s = 0$ at the corresponding trap boundary. We then have $j(x, y) \approx \left| \dfrac{dp(s)}{ds} \right|$ and therefore the flux density at the trap boundaries is approximated as

$$j(x, y) \approx |a_1^k|. \tag{63}$$

The polynomial coefficient a_1^k, $k = 1, \ldots, 4$ in the flux approximation (63) is defined from the conditions $p_k(s) = u_{ij}$, where the function values u_{ij} are taken at grid nodes along the direction of the outward normal vector to the trap edge k. Consider for instance the left boundary ($k = 1$) of the trap. We require that the parameter s is $s = 0$ at the boundary. We also consider the polynomial $p_1(s)$ at points $s = -h$ and $s = -2h$ and require that

$$\begin{aligned}
&p_1(0) = a_0^1 = u_{i_I j}, \\
&p_1(-h) = a_0^1 - a_1^1 h + a_2^1 h^2 = u_{i_I - 1, j}, \\
&p_1(-2h) = a_0^1 - 2a_1^1 h + 4a_2^1 h^2 = u_{i_I - 2, j},
\end{aligned}$$

for any fixed $j = j_I, \ldots, j_{II}$. Solving these equations and taking into account that $u_{i_I j} = 0$ we arrive at

$$a_1^1 = \frac{u_{i_I-2,j} - 4u_{i_I-1,j}}{2h}.$$

Similar expressions are then obtained for a_1^k, $k = 2, 3, 4$.

Approximation at the corners. The flux density approximation (62) can be computed at all grid points belonging to the trap boundary except for the corner points (x_{i_I}, y_{j_I}), $(x_{i_{II}}, y_{j_I})$, $(x_{i_I}, y_{j_{II}})$, and $(x_{i_{II}}, y_{j_{II}})$. The approximation at corner points is obtained by averaging the flux values computed at the neighboring points. In our method we assume that the flux density at any corner point depends on the flux in two orthogonal directions and therefore directional derivatives must be computed according to (62) at all neighboring points involved into the flux computation at the corner. For example, in order to compute the flux density $j(x_{i_I}, y_{j_I})$ we first compute it at grid nodes (x_{i_I-1}, y_{j_I}), (x_{i_I+1}, y_{j_I}),(x_{i_I+2}, y_{j_I}), (x_{i_I}, y_{j_I-1}),(x_{i_I}, y_{j_I+1}) and (x_{i_I}, y_{j_I+2}). Given the flux density values at these points, we then use quadratic polynomial interpolation in order to obtain the flux density $j(x_{i_I}, y_{j_I})$. Namely, we approximate the function $j(x, y)$ as

$$j(x, y) \approx \sum_{n=0}^{6} q_n \phi_n(x, y).$$

The polynomial basis functions $\phi_n(x, y)$ are defined as $\phi_n(x, y) = (x - x_{i_I})^{\alpha} (y - y_{j_I})^{\beta}$, where α and β are nonnegative integers such as their sum $s = \alpha + \beta$ should take the values $s = 0, 1, 2$. We therefore have $j(x_{i_I}, y_{j_I}) = q_0$, where the coefficient q_0 is found from the approximation above. The flux density values at the other corners of the trap are computed in a similar way. Let us note here that our approach to the flux computation at corner points is, of course, not unique and may require further discussion. Meanwhile, the findings of this paper along with the results of our previous work (29) demonstrate that a quadratic approximation of the flux, while being a more challenging technical task, gives a much more accurate answer than a linear approximation when a regular geometric grid is used in the problem.

Calculation of the total flux. Once flux densities have been computed at each grid node on the trap boundary, they are converted to a total flux $J(\partial S)$ by numerical integration of the flux density $j(x, y)$ along the boundary edges. The flux density is considered as a one-dimensional function $j(x, y) = f(\gamma)$ at each trap boundary, where $-1/2 \le \gamma \le 1/2$. At any trap boundary we have $2m + 1$ nodes of a computational grid where the function $f(\gamma)$ is defined. Hence the trapezoidal rule of numerical integration can be applied to integrate the flux density along the trap edge,

$$\int_{-1/2}^{1/2} f(\gamma) d\gamma \approx \sum_{p=1}^{2m+1} \omega_p f_p, \tag{64}$$

where f_p, $p = 1, \ldots, 2m+1$ are function values at grid nodes on the trap edge where the integration is carried out, and the weights ω_p are defined as in the integration rule (29).

Given the flux $J(\partial S)$, an approximation of the total number of insects $\Delta U^{n,n+1}$ crossing the trap boundary between time t_n and t_{n+1} is obtained as $\Delta U^{n,n+1} = J(\partial S)\tau$. The cumulative trap

count is then computed by the formula (26).

References

[1] Barenblatt, G.I. (1996) Self-similarity and Intermediate Asymptotics. Cambridge University Press.

[2] Bearup, D., Petrovskii, S.V. (2015) On time scale invariance of random walks in bounded space. *J. Theor. Biol.* **367**, 230-245.

[3] Bearup, D., Petrovskii, S.V., Benefer, C., Blackshaw, R. Light-baited trap's count dynamics modelled by the advection-diffusion equation. In preparation.

[4] Berg, H.C., (1983) Random Walks in Biology. Princeton University Press, Princeton.

[5] Buckingham, E. (1914) On physically similar systems; illustrations of the use of dimensional equations. *Phys. Rev.* **4**, 345376.

[6] Burn, A.J. (1987) *Integrated Pest Management.* Academic Press, New York.

[7] Byers, J.A. (1993) Simulation and equation models of insect population control by pheromone-based traps. *J. Chem. Ecol.* **19**, 1939-1956.

[8] Chorin, A.J., Hald, O.H. (2006) Stochastic Tools in Mathematics and Science. Springer, New York.

[9] Compton, S.G. (2002) Sailing with the wind: dispersal by small flying insects. In *Dispersal ecology* (J.M. Bullock, R.E. Kenward, R.S. Hails, eds.), pp. 113-133. Blackwell, Oxford.

[10] Davis, P.J., Rabinowitz, P. (1975) Methods of Numerical Integration. New York, USA: Academic Press.

[11] Fagan, W.F., Cantrell, R.S., Cosner C. (1999) How habitat edges change species interactions. *Am. Nat.* **153**, 165-182

[12] Flechtmann CA, Ottati AL, Berisford CW. (2000). Comparison of four trap types for ambrosia beetles (Coleoptera, Scolytidae) in Brazilian Eucalyptus stands. J. Econ. Entomol. 93(6):1701-7.

[13] Garvie, M.R., Trenchea, C. (2007) Finite element approximation of spatially extended predatorprey interactions with the Holling type II functional response. *Numer. Math.* **107**, 641-667.

[14] Garvie, M.R., Maini, P.K., Trenchea, C. (2010) An efficient and robust numerical algorithm for estimating parameters in Turing systems. *J. Comput. Phys.* **229**, 7058-7071.

[15] Gordon D. Smith, G.D. (1985) Numerical Solution of Partial Differential Equations: Finite Difference Methods. Oxford University Press.

[16] Higley, L. G., Pedigo, L. P. (1996) Economic Thresholds for Integrated Pest Management, University of Nebraska Press.

[17] Hirsch, Ch. (1992) Numerical Computation of Internal and External Flows, Vol I & II, Wiley.

[18] Holmes, M.H. (2006) Introduction to Numerical Methods in Differential Equations. Springer.

[19] Jansen, V.A.A., Mashanova, A., and Petrovskii, S.V. (2012) Model selection and animal movement: Comment on "Levy walks evolve through interaction between movement and environmental complexity." *Science* **335**, 918.

[20] Kogan, M. (1998) Integrated pest management: historical perspectives and contemporary developments. *Annual Review of Entomology* **43**, 243-270.

[21] Kovacs-Hostyanszki, A., Haenke, S., Batary, P., Jauker, B., Baldi, A., Tscharntke, T., Holzschuh, A. (2013) Contrasting effects of mass-flowering crops on bee pollination of hedge plants at different spatial and temporal scales. Ecological Applications 23, 1938-1946.

[22] Ludwig, D., Aronson, D.G., Weinberger, H.F. (1979) Spatial patterning of the spruce budworm. *J. Math. Biol.* **8**, 217-258.

[23] Morton, K.W.,Mayers, D.F. (1994) Numerical Solution of Partial Differential Equations. An Introduction. Cambridge, Cambridge University Press.

[24] Northing, P. (2009) Extensive field based aphid monitoring as an information tool for the UK seed potato industry. *Asp. Appl. Biol.* **94**, 31-34.

[25] Okubo & Levin 2001.

[26] L.P. Pedigo, G.D. Buntin, Eds., (1994) *Handbook of Sampling Methods for Arthropods in Agriculture*. Boca Raton: CRC Press.

[27] Perry, J.N. (1996) Simulating spatial patterns of counts in agricuture and ecology. *Computers and Electronics in Agiculture* **15**, 93-109.

[28] Peshin, R., Dhawan, A. K. (eds.) (2009) *Integrated Pest Management: Innovation-Development Process, Volume 1*, Springer.

[29] Petrovskaya, N.B. (2008) Discontinuous weighted least-squares approximation on irregular grids. *CMES: Computer Modeling in Engineering & Sciences*, 2008, **32**,69–84

[30] Petrovskaya, N.B., Embleton, N.L. (2013) Evaluation of peak functions on ultra-coarse grids. Proc. R. Soc. A, 469:20120665 doi: 10.1098/rspa.2012.0665

[31] Petrovskaya, N.B., Petrovskii, S.V., and Murchie, A.K. (2012) Challenges of ecological monitoring: estimating population abundance from sparse trap counts. *J. R. Soc. Interface* **9**, 420–435.

[32] Petrovskii, S.V., and Petrovskaya, N.B. (2012) Computational ecology as an emerging science. *Interface Focus* **2**, 241-254.

[33] Petrovskii, S.V., Mashanova, A., and Jansen, V.A.A. (2011) Variation in individual walking behavior creates the impression of a Lévy flight. *PNAS* **108**, 8704-8707.

[34] Petrovskii, S., Bearup, D., Ahmed, D.A., Blackshaw, R. (2012) Estimating insect population density from trap counts. *Ecological Complexity* **10**, 69–82.

[35] Petrovskii, S.V., Petrovskaya, N.B., Bearup, D. (2014) Multiscale approach to pest insect monitoring: random walks, pattern formation, synchronization, and networks. *Phys. Life Rev.*, **11**, 467–525

[36] Raworth, D.A., Choi, W.-J. 2001 Determining numbers of active carabid beetles per unit area from pitfall-trap data. *Ent. Exp. Appl.* **98**, 95–108.

[37] P.J. Roache (1985) Computationbal Fluid Dynamics. Hermosa, Albuquireque.

[38] Stern, V.M. (1973) Economic thresholds. *Ann. Rev. Entomol.* **18**, 259-280.

[39] Thomas, J.W. (1995) Numerical Partial Differential Equations: Finite Difference Methods. Springer NY.

[40] Thomas, C.F.G., Parkinson, L., Marshall, E.J.P. 1998 Isolating the components of activity-density for the carabid beetle Pterostichus melanarius in farmland. *Oecologia*, **116**, 103-112.

[41] Turchin, P., 1998. Quantitative Analysis of Movement. Sinauer, Sunderland.

[42] Volpert, A.I., Hudjaev, S.I. (1985) Analysis in Classes of Discontinuous Functions and Equations of Mathematical Physics. Martinus Nijhoff, Dordrecht.