Towards a statistical physics of dating apps

Fabrizio Olmeda
Max Planck Institute for the Physics of Complex Systems, Nöthnitzer Str. 38, 01187 Dresden, Germany
E-mail: fabrizio@pks.mpg.de

Received 17 June 2022
Accepted for publication 5 October 2022
Published 30 November 2022

Online at stacks.iop.org/JSTAT/2022/113501
https://doi.org/10.1088/1742-5468/ac9bed

Abstract. Over the last ten years, a sharp rise in the number of dating apps has broadened the spectrum of how one can get in contact with new acquaintances. A common feature of such apps is a swipe, enabling a user to decide whether to like or dislike another user. As is the case in real life, a user may be more or less popular, which implies that the distribution of likes among different users is broad. In this paper, we show how likes are distributed across users, based on different decision-making strategies, app settings and their feedback. We apply theoretical methods originally developed in non-equilibrium statistical physics to investigate the dynamics of dating app networks. More specifically, we show that whenever a dating app differentially displays users with respect to their popularity, users are split into two categories: a first category including users who have received the most likes and a second category, referred to as a condensate, which in long-term will be reduced to a small fraction of likes or to no likes at all. Finally, we explore realist models based on a rating system of the users, known as Elo. These models will turn out to exhibit behaviour typical of gelating systems, characterized by a bimodal distribution of likes among the users with broad tails. Altogether, we provide a minimal theoretical framework to infer statistical observables in social networks governed by coupled internal states.

Keywords: agent-based models, online dynamics, scaling in socio-economic systems, socio-economic networks
Towards a statistical physics of dating apps

1. Introduction

There are many differences amongst dating apps in terms of target audience, design, and popularity. However, most of them rely on the common feature of letting the user decide to whom they are attracted to and want to start a conversation with. In many cases, they will suggest some users over others based on common interests, goals, etc. Attraction to a particular user is translated into a user giving a like or dislike to another one. When both users give each other a like, the pair is said to be a ‘match’. These two functions provide the basis of most dating apps. In most cases, a user cannot see all the active users at the same time, so as to decide to whom to give a like. Rather, the user is faced with one user at a time and decides whether to like or dislike sequentially. The statistics of matches and likes in this case is determined by the choices of the users and a network of possible connection emerges. Many questions arise from the dynamics of
this networks, for example: how many people should we date before finding the right one? Also known as the 37% rule [1].

The factors involved in the decision-making process of each individual user can be very different from one to another. Some users may be more interested in physical attractiveness, others on shared hobbies and passions, or a mixture of both. As there are many different factors that determine whether a user is keen to give or not give a like to another user, the decision-making process is not trivial at all [2, 3]. As an example, in [4] it has been shown that we may judge a person to be more or less attractive in comparison to other users we judged before. In [5], they asked different groups to rate pairs of twins between 1 and 7 and it was shown that there was a significant difference in the average rating of the twins, illustrating the intrinsic variability in perception of physical attractiveness. However, when the distribution of rating does not show fat tails, meaning that if the average ‘rating’, as well as its standard deviation, are well defined, then individual preferences will be smeared out on large numbers. In [5], the standard deviation was smaller by more than a factor of four with respect to the mean. In the following, we will define \( x_i \), the attractiveness of a particular user, as it would be perceived on average.

In this work, we explore the dynamics of likes in dating apps networks based on different decision making processes and possible different settings of the app. In particular, different users may be shown differently with respect to certain criteria, for example, how many likes they get on average, or the difference between likes and dislikes, which will determine their popularity, to which we will refer to as rank or rating. We will show that for different models, whenever an app sets a like-based visibility, the users will be split into two groups: a relatively small group that will receive most of the likes and a bigger one, which we will refer to as a condensate, that in the long term, will receive few likes, and eventually none. In the last section, we will apply an Elo-based rating system [6], often used for sports, such as chess and tennis, which rates users in a non-trivial way with respect to the number of likes/dislikes received. We will show that even in this more complex scenario, the condensates still exist and in some particular cases the network will exhibit gelation. Everywhere, we will define a network of users and we will not take into account gender.

2. Dynamics of swiping in dating apps

Before focusing on specific examples, we will now describe a general model for the dynamics of the variables, likes, dislikes and matches, in dating apps. In a dating app network of \( N \) users, for each pair of users we define a parameter termed relative attractiveness \( x_{ij} \), which quantifies the degree of attractiveness that user \( j \) perceives of user \( i \). Attractiveness depends on many different factors and different persons may attribute different levels of the same user as more or less attractive. As in [5], we will take \( x_{ij} \) to be a time independent scalar. In the following sections, we consider a mean-field limit, where we neglect heterogeneity in the perception of attractiveness by other users. In this limit, the value of user \( i \) effectively depends on their average attractiveness,
Towards a statistical physics of dating apps

Figure 1. (a) Possible interactions and like/dislike exchange between a pair of users. The two outcomes, likes (‘∗’) or dislike (‘⊣’) are determined by the probability $K(c_i, c_j)$ that user $i$ sees user $j$, which depends on their rank and the probability $f(x_i, x_j)$ that the user $i$ gives a like or dislike to the user $j$. (b) Point of view of an individual user after a possible realization of the simulations. Users are defined by their attractiveness and a rank given by the app. We consider that decisions in the network are determined solely by these two factors. As in example (b), a higher ranked user (big circle) is more visible and thus receives more likes/dislikes. On the other hand, interactions between pairs of users are asymmetric. As in the example, the user $i$ might interact with the user $j$ but not vice versa.

$x_i = \frac{1}{N-1} \sum_{j=1, j \neq i}^{N} x_{i,j}$, which we will henceforth refer to as the attractiveness of user $i$. As $x_i$ depends only on $x_{ij}$ it is a time independent scalar as well.

A dating app may decide not to show every user equally with equal probability. In particular, the dating app may define a criterion for which a user is shown more or less than other users. One possible criterion may be to show users with a probability that depends on the number of likes and dislikes received, i.e. ranking the users. We will refer to the rank of users $i$ as $c_i$ and we will take it to be a positive integer number. The rank is a time dependent variable, and its dynamics will be specified in the following sections and will depend on the model we will study. We now define the general dynamics on the network of users (figure 1(a)). At each time step of the simulation, we randomly select a user $i$ and another user $j$ with rules described in appendix A, based on an interaction kernel $K(c_i, c_j)$. The pair is selected with order. By order, we mean that we select at each time step the interaction $i \rightarrow j$ and not $j \rightarrow i$, as this is what typically happens in dating apps. By the $i \rightarrow j$ interaction, we mean that the profile of a user $j$ appears on the app of the user $i$. The probability of a like $P(J_{ij} = 1)$ between user $i$ and $j$ is the product of the probability $K(c_i, c_j)$ of a directed interaction $i \rightarrow j$ multiplied by the probability $f(x_i, x_j)$ that $i$ likes $j$, given the aforementioned interaction, $P(J_{ij} = 1) = f(x_i, x_j)K(c_i, c_j)$. The functions $f(x_i, x_j)$ and the kernel $K(c_i, c_j)$, which are unspecified at this point, will depend on the particular models that we are going to study. In general, the probability $f(x_i, x_j)$ of user $i$ giving a like to user $j$ after an interaction is directly proportional to $x_j$. The rank $c_j$ of the user $j$ will determine the probability that the user is being shown

https://doi.org/10.1088/1742-5468/ac9bed
Towards a statistical physics of dating apps

on the app. $K(c_i, c_j)$ is then in general an increasing function of $c_j$. The probability of a dislike therefore is $P(J_{ij} = -1) = 1 - P(J_{ij} = 1)$. Time $(t)$ is increased by $1/N$ after each iteration step. We do not consider the history of the users, which is to say that we do not consider whether a like or dislike was already given. In particular, users typically may decide to remove the like, which we consider equivalent to a dislike or they may change their mind toward another user if the app proposes that user again. We then extract another pair and repeat the procedure. In figure 1(b), we show a possible realisation of the simulations from the point of view of an individual user. In case both users give a like to each other, this is called a ‘match’. A user can likewise remove a like and consequently the match. In a statistical physics sense, we are looking at the dynamics of the couplings (likes or dislikes), rather than nodes (users).

In the following, we will consider two possible scenarios for the probability of giving a like: biased and non-biased. In the first case (biased), the decision of a user $i$ will depend on the difference in attractiveness between themselves ($x_i$) and the attractiveness of proposed user $j$ by the app ($x_j$). For non-biased decisions, the probability of a like will depend only on the attractiveness of $j$. We will take everywhere, unless specified otherwise, a fixed network, where the number of users is a constant.

3. Biased decisions

In the case of biased decision making, a user $i$ will make a decision based on the difference between the attractiveness of themselves and of user $j$ ($\delta_{ij} = x_i - x_j$). A simple form for this interaction is

$$P(J_{ij}^t = 1) = \min(1, e^{-\beta \delta_{ij}}).$$

$P(J_{ij}^t = 1)$ is the probability of user $i$ giving a like to user $j$ at time $t$, or keeping a like in case the like is already present. The probability of removing a like is then $P(J_{ij}^t = -1) = 1 - P(J_{ij}^t = 1)$. The probability of giving a like is equal to one when user $i$ perceives user $j$ as more attractive than themselves. In the other case, the like is given with a decreasing probability as the difference of attractiveness increases. This choice is in accordance with [7], where it was found that users seek for partners, who are on average more attractive than themselves. $\beta$ is the average ‘pickiness’ or bias. When $\beta = 0$, every given user will like every other user regardless of their attractiveness. Instead, when $\beta \to \infty$ a user $i$ will never give a like to another user $j$, unless $j$ is more attractive than $i$.

Initially, we will consider attractiveness to be a random variable, Gaussian distributed with a mean $\mu$ and variance $\sigma^2/2$. The difference between two Gaussian random variables ($\delta_{ij}$) is itself a Gaussian random variable with mean $\mu = 0$ and variance $\sigma^2$, which we will set to one. In every numerical simulation in this work, we extract the attractiveness from a given distribution and, when needed, compute $\delta_{ij}$ accordingly. The network and its dynamics are fully defined and we can then study long term statistics.
Figure 2. Distributions of the average probability of matches as a function of the parameter $\beta$ (‘pickiness’ or bias) and the attractiveness. The distribution of attractiveness is uniformly distributed in $[0, 1]$ (a) and Gaussian distributed with mean $\mu = 0.5$ and variance $\sigma = 0.1$ (b).

In particular (appendix B), the probability distribution of the match probability is

$$P(m) = \frac{1}{\sqrt{2\pi m^2}} e^{-\frac{(\log(m))^2}{2\beta^2}}, \quad m \in [0, 1]$$

(2)

where $m$ is the probability of a match. $P(m)$ is a log-normal distribution, for high values of $\beta$ where most of the pairs have low probability of a match, while the opposite is true for low values of $\beta$. We define $\beta_L$, the value of $\beta$ such that, if $\beta < \beta_L$ more than 50% of couples have more than 50% probability of a match. $\beta_L$ is then the solution of the integral equation

$$\int_0^{0.5} dmP(m, \beta_L) = 0.5.$$  (3)

Numerically, $\beta_L \approx 0.83$. Another statistically relevant quantity is the marginal probability of matches given the attractiveness of a user. In other words, how many matches is a user expected to have if their attractive is $x_i$? The results (appendix B) are shown in figures 2(a) and (b) for two different distributions of attractiveness, respectively uniform and Gaussian. In both cases, attractive users are expected, on average, to get less matches than users whose attractiveness is closer to the mean value of the distribution. This apparently surprising effect occurs because users who are perceived to be very attractive are likely not to give many likes and on top of that, they are on the tails of the distribution of attractiveness and thus represent the minority of users. Combining these results gives them a low chance of having a match, while users whose perceived attractiveness is close to the mean $\mu$ will be the ones with most matches, but receiving...
less likes. The difference in match percentage is smoothed with a uniform distribution of attractiveness, as in this case, the very attractive users are not a minority. For networks with biased decision making, the dynamics of matches and likes can be quite different and below we will prove that this does not apply to non-biased decisions.

4. Unbiased decisions

In the case of unbiased decision making, a user $i$ will make a decision solely based on the attractiveness of the user $j$. A simple form for the probability of giving a like is

$$P(J_{ij} = 1) = x_j^\alpha,$$

(4)

where $\alpha$ plays the same role as $\beta$ in the previous example. Another possible choice could be $P(J_{ij} = 1) = e^{x_j^\alpha}$ as before or other functional forms, but as it will later become clear, the particular choice does not influence the dynamics of the models we will consider. In particular, $\alpha = 0$ means user giving likes to every other user and $\alpha \to \infty$ corresponds to no likes in the network. We will assume everywhere $\alpha > 1$ and attractiveness $x \in [0, 1]$. In this very simple example, the probability of a match between users $i$ and $j$ is given by $p(m_{ij}) = (x_i x_j)^\alpha$.

Differing from the previous case, the average number of matches is a monotonically increasing function of $x_i$, meaning that the number of matches of users with attractiveness $x_i$ is trivially proportional to itself: $\langle n(x_i) \rangle = x_i^\alpha E[x^{\alpha\gamma}]$, where $E[...]$ is the expected value over the distribution of attractiveness. In the following, unless stated otherwise, the attractiveness of users is sampled from a uniform distribution. The monotonic behaviour is not changed by the specific choice of the distribution of attractiveness. In the uniform case, $\langle n(x_i) \rangle = x_i^\alpha / (1 + \alpha)$. This result is quite trivial and is likely to be far from a real dating app network. In the next section, we are going to model different restrictions for the possible ways in which users can give likes.

4.1. Unsupervised visibility

A dating app may decide not to show every user with equal probability. We then consider a certain criterion for which a user is shown more or less than other users. One possibility of such criterion may be to show users with a rate that depends on the difference between the number of likes and dislikes received, i.e. ranking the users. Analytical insights regarding the number of matches in such conditions are hard to obtain, as they involve two-pair interactions. In this scenario, we focus on the dynamics of likes, which, as we are going to show for different numerical simulations, will outline the distribution of the matches. In the following, unless stated otherwise, we allow a user to give a like or dislike to another user multiple times. Multiple dislikes to the same user are reasonable in most common dating apps. Indeed, users are proposed again after some time, even if there was an initial rejection. Multiple likes are typically not possible in most common dating apps. As stated in the previous section, giving multiple likes is interpreted as keeping a match in case the match is already present. However, we allow user $i$ to give
multiple likes to the same user $j$ even if there is no match. This definition is believed not to be relevant for large system sizes ($N$) or for the long-term distributions of matches and ranks, but it is expected to give corrections in the short-term dynamics.

As the visibility is proportional to the number of likes and dislikes received, we may expect that the probability of a match is proportional to the product of these factors. This naive reasoning will lead to a probability of a match: $p(m_{ij}) \sim (x_ix_j)^{\alpha} v_i v_j$, where $v_i$ is the visibility of a user $i$. We initially take visibility to be a function of the average number of likes ($k_1$) minus the dislikes ($k_2$) received $v = v(k_1 - k_2)$. As we will show later, $k_1 - k_2 \sim 1 - 2x^\alpha$, so that the argument of the visibility function has a sign change when $x^\alpha_1 = 1/2$. Intuitively, we found a threshold in attractiveness $x_C = (1/2)^{1/\alpha}$, which will influence the long term behaviour of the system. In particular, we will show that likes and matches are concentrated into users that have a perceived attractiveness $x > x_C$, whilst other users will have very few to no matches. These phenomena resemble condensation phenomena in physics [8, 9] or the wealth distribution in asset exchange models [10, 11], where in certain conditions the resources are concentrated in few traders.

The reason for the failure of the naïve approach comes from taking a stationary perspective. Instead, we have to consider the full dynamics, as this system will turn out to exhibit absorbing states [12]. If the visibility of a user goes to zero, that user will not be shown anymore, and we will refer to the user as to be in a condensate. This naive reasoning will lead to a probability of a match:

$$G(z_1, z_2, z, x, t) = \sum_{k_1, k_2, c} z_1^{k_1} z_2^{k_2} z^c n(k_1, k_2, c, x, t),$$

and $\omega$ is the swiping rate, which is set to one. These equations are supported with boundary equations at $c = 0$ and $c = N$, which are easily derivable. We initially consider the case in which, every time a user gets a like, their category is increased by one and decreased by one in case of a dislike. This feature is encoded in the variable $c$. The first term on the right-hand side of equation (5) takes into account that the density of users with $k_1$ likes increases whenever a user who had $k_1 - 1$ likes receives a like. The rank after receiving a like is increased by one as well. The second term accounts for dislikes and the third is the loss term due to getting a like or dislike. We take the visibility, which is the typical rate at which a user is seen, to be equal to the rank, $v(c) = c/N$. In the following, we make the time adimensional, dividing it by $\omega/N$ and refer to it simply as $t$. Upon defining a moment generating function, $G(z_1, z_2, z, x, t) = \sum_{k_1, k_2, c} z_1^{k_1} z_2^{k_2} z^c n(k_1, k_2, c, x, t)$,
the average number of likes \( \langle k_1(x, t) \rangle \) and the average category \( \langle c(x, t) \rangle \) for users that are attractive \( x \) at a time \( t \) is given by (appendix C)

\[
\langle k_1(x, t) \rangle = c_0 x^\alpha e^{(2x^\alpha - 1)/2x^\alpha - 1} - 1, \quad \langle c(x, t) \rangle = c_0 e^{(2x^\alpha - 1)}. \tag{6}
\]

\( c_0 \) is the initial rank and \( x \neq x_C \). The average rank will increase for users such that \( x > x_C \) and goes to zero for \( x < x_C \). The number of likes will increase exponentially whenever \( x > x_C \) and saturate exponentially for \( x < x_C \). The time scale at which this saturation occurs is \( \tau = |2x^\alpha - 1|^{-1} \). The time scale \( \tau \) then sets the expected time before reaching the condensate. The simple exponential relationship is expected to hold only for a time such that \( t < 1/\tau \) and for \( x < x_C \). The reason for the latter inequality is in the boundary conditions for the rank \( c \), as equation (6) predicts an exponential growth beyond \( c = N \) for \( x > x_C \).

In figure 3(a) (top, bottom), we compare numerical simulations of the exact agent-based model to analytical solutions from equation (6). In figure 3(a) (top), we show the average category with respect to attractiveness at different times. As expected, there is a sharp transition at \( x = x_C \), signalling condensation of likes and matches in few users. In figure 3(a) (bottom), we compare the number of matches from the simulation to the number of likes. A linear regression is sufficient to fit the data, meaning that, as expected, the number of likes received is a good statistic for the matches in this model. It is important to recall that the threshold is not only given by the app ranking the users, but by the choices of the users themselves. Large values of \( \alpha \) means that a user is keen to reject almost every other user. Even though, on average, the category of users with \( x > x_C \) increases over time, not every user for whom \( x > x_C \) will escape from the condensate. As this is a first exit problem [12], the first moments will not be sufficient to answer the question.

As stated at the beginning of the section, we allow users to give multiple likes or dislikes to the same user. In figure 3(a) (top, inset), we compare the results of the normalized number of likes received in time by users with \( x < x_C \) (\( x = 0.3, \alpha = 2 \)) either accounting for multiple likes (green) or not (purple). Numerical and theoretical solutions are in good agreement, even beyond the expected time-scale. The difference between the two cases increases as we reach the condensation threshold and goes to zero for \( x \ll x_C \). A theory beyond mean field is required to properly take the more reasonable definition of likes into account.

In this simple model, it is easy to write an equation for the dynamics of the ranks upon integrating out the other degrees of freedom from equation (5),

\[
\partial_t n(c, x, t) = x^\alpha (c-1)n(c-1, x, t) + (1-x^\alpha)(c+1)n(c+1, x, t) - cn(c, x, t). \tag{7}
\]

Equation (7) is a known birth-death process [13], which after defining a generating function \( G(z, x, t) = \sum_c z^n(c, x, t) \), can be solved with the method of characteristics. \( n(c, x, t) \) is found by a series expansion of \( G(z, x, t) \) [14] and the number of users in the condensate is \( (n(c = 0, x, t) \equiv n_0(x, t)) \).
Towards a statistical physics of dating apps

**Figure 3.** (Top) Average normalized rank with respect to the maximum for unitary transition between ranks without any secured visibility (a), with secured visibility (b) and with ‘push’ strategy (c). Theoretical predictions (dashed black line) are respectively the condensation threshold \( x_C = \frac{1}{2} \alpha \) (a), (b) and the asymptotic limit of the average rank from equation (13) (c). (Inset) Number of likes with (purple) or without (green) allowing multiple likes are compared to theoretical prediction (dashed black line). A linear regression between matches and effective likes is in agreement the simulated data (dots). \( \gamma = 0.9 \) in (b), (c) and time is everywhere in arbitrary units.

The steady state value for the number of users in the condensate is:

\[
\lim_{t \to \infty} n_0(x, t) = 1 \text{ if } x < x_C \text{ and } n_0(x) = \left( \frac{1 - x^\alpha}{x^\alpha - (1 - x^\alpha)} e^{t(1-2x^\alpha)} \right)^{q_0} \text{ if } x > x_C.
\]  

The steady state value for the number of users in the condensate is:

\[
n_0(x, t) = \lim_{t \to \infty} n_0(x, t) = 1 \text{ if } x < x_C \text{ and } n_0(x) = \left( \frac{1 - x^\alpha}{x^\alpha - (1 - x^\alpha)} e^{t(1-2x^\alpha)} \right)^{q_0} \text{ if } x > x_C.
\]  

This is quite surprising, as even a user who is perceived to be more attractive than the condensation threshold may still end up in the condensates with a finite probability. This probability monotonically decreases with the initial value of the rank. This model gives a very strict criterion and will not ensure any guarantees of matches to the users. In the next section, we will analyse different possibilities for a dating app to prevent condensation, while keeping a rank-based approach.
4.2. Supervised visibility

We consider two possible scenarios in which a dating app may secure a constant visibility to avoid condensation. In the first case, we consider that users will still be shown with a given rate no matter their rank. The dynamics of \( n(k_1, k_2, c, x, t) \) are given by

\[
\partial_t n(k_1, k_2, c, x, t) = \gamma L[n] + (1 - \gamma)L^*[n].
\]

We introduced the operator \( L[n] \), which is equal to the rhs of equation (5) and \( L^*[n] \), which is equal to \( L \), but all the factors in \( v(c) \) are set to one. The parameter \( \gamma \) is the relative weight of the constant visibility and the like-based visibility. In particular, terms in equation (9) that carry a \( 1 - \gamma \) factor do not involve any visibility criterion, such that the rate to be seen is equal to \( 1 - \gamma \) for every user. In the following, for every model, we rescale all rates to avoid an explicit dependence on \( N \) and we will not discuss it further as we will be mostly interested in long-term statistics. As in the previous section, we can simplify this equation by integrating out \( k_1, k_2 \) and study the dynamics of the density of users with a rank \( c \) as

\[
\partial_t n(c, x, t) = \gamma L_c[n] + (1 - \gamma)L^*_c[n].
\]

\( L_c \) is equal to the rhs of equation (7). \( L^*_c \) is equal to \( L_c \), but all the factors in \( v(c) \) are again set to one. The asymptotic behaviour of the density of users in the condensate is given by

\[
n_0(x) = \frac{\gamma}{1 - \gamma} \left[ \int_0^1 ds (1 - f_\alpha s)^{-\gamma} (1 - s)^{\frac{\alpha}{1 - \gamma}} \right]^{-1},
\]

where \( f_\alpha = x^\alpha / (1 - x^\alpha) \). The density of users with low attractiveness in case the visibility criterion is irrelevant (\( \gamma \to 0 \)) scales as \( n_0(x \to 0) \sim (1 - x^\alpha) + \gamma \). This implies that the particular policy of the app and the probability for which a user decides to give a like equally contribute to the number of likes received. The number of likes received can be estimated from equation (9) and its approximate asymptotic limit in case \( x < x_c \),

\[
\lim_{t \to \infty} \int_0^1 dx \langle k_1(x, t) \rangle / \int_0^1 dx \langle k_1(x, t) \rangle = 0.
\]

In order to compare theoretical results with simulations of the agent-based model, we need to consider that even though the number of likes for users such that \( x < x_c \) increases linearly, it is always a negligible fraction of the total amount of likes given in the network: \( \lim_{t \to \infty} \int_0^1 dx \langle k_1(x, t) \rangle / \int_0^1 dx \langle k_1(x, t) \rangle = 0 \). This way of securing a constant visibility, though increasing the number of likes for users such that \( x < x_c \), will never break the condensation mechanism. Moreover, in comparison with the previous model, the average rank decays faster to zero, figures 3(a) and (b) (top). This apparently surprising behaviour is explained by intuitively noticing that being more visible outside the condensate increases the chance of being rated lower for users with \( x < x_c \).

Another possibility in which the app may secure a constant visibility is a ‘push’ strategy. As an example, we consider that with a certain rate \( 1 - \gamma \), the app increases
the rank of every user to avoid condensation. The dynamics of the density of users in a
given rank is described by

$$\partial_t n(c, x, t) = \gamma L_c[n] + (1 - \gamma)[n(c - r, x, t) - n(x, c, t)].$$

(12)

We again set $\gamma$ as the relative contribution in order to compare the result with the
previous case. $r$ is the amount of ranks that the app decides to push users up. Equation (12)
has a similar form to the master equation for gene expression with translational burst
[16, 17], which in this case and for $r = 1$ can be solved exactly. The solution can be found
by the method of characteristics (appendix D). The density of users in the condensate
and the average asymptotic rank, for $x < x_C$, scales as

$$n_0(x) = \left(1 - \frac{x^\alpha}{1 - x^\alpha}\right)^{\frac{1 - \gamma}{\gamma}} , \quad \langle c_\infty(x) \rangle = \frac{1 - \gamma}{\gamma(1 - 2x^\alpha)},$$

(13)

and $x < x_C$. The average rank does not decay exponentially to zero as before, but rather
saturates to a constant value, always assuring a non-zero visibility to every user. However,
even in this scenario, the number of likes received by the users for which $x < x_C$
compared is still negligible compared to the total number of likes given in the network:

$$\lim_{t \to \infty} \int_0^{x_C} dx \langle c(x, t) \rangle / \int_0^1 dx \langle c(x, t) \rangle = 0.$$

In figure 3(c) (top), we plot results of simulation of the corresponding agent-based
model with the theoretical bound. Likes are again sufficient to predict the matches,
figure 3(c) (bottom). The reason why this strategy is more profitable for users than
the previous one for $x > x_C$ is straightforward. In the ‘push’ strategy, a user may climb
some ranks and eventually rise far enough from the condensate, so that it will take a
long time to return in the condensate. In the case of a secured visibility, it is only the
probability of receiving a like that can bring a user out of the condensate, which is
clearly not profitable for users such that $x \ll x_C$.

So far, we have only considered very simple models in order to understand how
relevant quantities, such as likes or matches, change in time and we have identified a
clear threshold for condensation. In the next paragraph, we will present more realistic
models and compare them with the minimal models that we have analysed so far.

5. Elo rating system

In some sports, like chess or tennis, players are rated based on an Elo score [6]. An
Elo rating system assigns a certain score to players, which changes with respect to the
outcomes of individual or tournament games. Here, we consider the scenario in which
dating apps decide to adopt a similar rating system and we are going to draw conclusions
on the number of likes received on average by each user and the effect that this ranking
system might have on condensation. A general feature of the Elo rating system is that
a user, who receives a like from another user in a higher rank, will climb ranks directly
proportional to the difference between the rank of the users. The opposite holds true in
case of a dislike (figure 4(a)).
Figure 4. (a) Schematic illustration of the application of ELO rating system in chess compared with models of dating apps. (b) Distribution of ranks for the product Kernel $K(c, c') = c(c' + 1)$. (c) Average Elo for non-unitary transition between ranks based on an ELO-like ranking system. In the inset the parameter $\alpha$ was distributed randomly from a Cauchy distribution with parameters $(0, 1)$. Even with this choice of a scale-free distribution, the condensation threshold is unaltered. (d) Average likes (rescaled by the maximum) for non-unitary transition between ranks based on an ELO-like ranking system. (e) Distribution of ranks for $v(c) = c$, $K(c, c') = 1/(1 + (c - c')^2)$, $\beta\gamma = 16$, $\beta = 0.02$. This choice is in accordance with the standard chess Elo rating system. (Inset) Magnification of the higher mode of the distribution with logarithmic scale for Elo ratings.
Towards a statistical physics of dating apps

We define \( n(c, x, t) \) as the density of users in the rank \( c \) who are attractive \( x \), whose dynamics are given by

\[
\frac{\partial n(c, x, t)}{\partial t} = \sum_{w=0}^{\infty} v(w) n(w, x, t) \sum_{j=0}^{\infty} \int_0^1 dy P(y) n(j, y, t) K(w, j, c) - n(c, x, t) v(c) \sum_{w,j=0}^{\infty} \int_0^1 dy P(y) n(j, y, t) K(w, j, c),
\]

(14)

where \( P(y) \) is the distribution of attractiveness and \( K(w, j, c) = x^\alpha K^+(w, j, c) + (1 - x^\alpha) K^-(w, j, c) \). In contrast to the previous models, the rate at which the density of users in the rank \( c \) increases or decreases, after a like or dislike, is proportional to the product of the densities of the users, rather than linear. The quadratic term appears because we take into account the rank of the users that give a like or dislike. In equation (14), \( K(w, j, c) \) is the rate at which the app proposes a user with Elo \( j \) to a user with Elo \( w \) and \( v(w) \) is the visibility rate as before. \( K^\pm(w, j, c) \) stands for like or dislike, respectively. The extra variable \( c \) in the functions \( K^\pm(w, j, c) \) indicates that the final Elo of a user, with initial Elo \( w \), is updated to \( c \), after receiving a like or dislike from a user with Elo \( j \). The same reasoning applies to the case of a dislike. We will study the situation that a like from a user in a lower category or a dislike from a user in a higher category makes less impact than a like or dislike from a user in a higher or lower category, respectively. This will give some constraint on \( K^\pm(w, j, c) \). In the next sections, we are going to present detailed numerical simulations of a version of equation (14) based on Elo rating systems.

### 5.1. Product Kernel

As a first simple model, we consider a product Kernel, such that \( K^\pm(w, j, c) = \delta(w, c \mp 1) (j + 1) \) in equation (14). Intuitively, a product Kernel increases the chance of users who are in higher ranks to actively use the app. Users who received more likes and so matches are likely happier with the dating app and they use it with higher frequency compared to other users. In this case, the dynamics of the density of users is described by \( v(w) = w \) as previously

\[
\frac{\partial n(c, x, t)}{\partial t} = L_c[n] \int_0^1 dy P(y) \sum_{j=0}^{\infty} (j + 1) n(j, y, t).
\]

(15)

Upon rescaling time as \( \tau = \int_0^t dt' \int_0^1 dx [\partial_z G(z, x, t) + G(z, x, t)]_{z=1} \), with \( G(z, x, t) = \sum_c z^n(c, x, t) \), the number of users in a condensate is (appendix E)

\[
n_0(x) = \left[ \frac{(1 - x^\alpha) - (1 - x^\alpha)e^{\tau(1-2x^\alpha)}}{x^\alpha - (1 - x^\alpha)e^{\tau(1-2x^\alpha)}} \right]^{c_0},
\]

(16)

where \( \tau \approx -\log[(1 + c_0f_0)e^{-t} - c_0f_0] \). Unlike the previous models, equation (16) is not properly defined at times \( t > \log[1 + (c_0f_0)^{-1}] = t_g \), signalling a drastic change of the Elo-ranking distribution. We then compute second and higher order moments of
equation (15) and we found that they all diverge at $t = t_g$ (appendix E), which is a signature of a gelation transition [18].

Gelation occurs in typical aggregation processes when, at finite times, one cluster has a non-vanishing fraction of the total mass. In models of dating apps, a giant cluster or gel is represented by few users, who have very high rank (mass) compared to the majority of users. The Elo score of the ‘gel’ increases over time at the expense of users with lower Elo scores. Unlike gelating system, the first order moment diverges in our system, as the total mass is not a conserved quantity. In figure 4(b), we confirm this theoretical prediction by showing how the Elo distribution evolves in time. We notice that the distribution flattens over time with a high peak around zero and a broad tail that moves dynamically to higher Elo values, as expected in a gelating system [19]. As in the previous models, users with $x > x_c$, will experience an indefinite increase of their rank, which is unlikely to be realistic. In the last section, we will derive a general model for a chess-based Elo rating system, which will fix this unwanted feature.

5.2. Chess-based Elo

We initially relax the assumption of unitary transition between ranks. This means that a user is able to jump more than one rank whenever they receive a like from users in a higher rank. In particular, we will consider a linear chess-based Elo rating system [6]). In this case, whenever a user in a rank $w$ receives a like or dislike from a user in a rank $j$, their new rank $c$ is

$$c = w \pm \frac{W}{2} + \frac{W}{4C}(j - w),$$

(17)

with a positive sign in the case of a like and negative for a dislike (in the book of Elo w is $K$, here we use $K$ for the Kernel). We define $w/4C = \zeta$ and $w/2 = \zeta \gamma$. An Elo ranking system then provides a way to rank users that is not trivially related to the number of likes. When sports are provided with such a rating system, it yields a way to decide whether two different players will compete. Indeed, it is quite unlikely to observe an Elo rated match between the world chess champion and a club player or even a grand master with a substantial Elo difference.

This last constraint is encoded in the interaction kernel $K(w, j, c)$. In figure 4(a), we outline the difference between chess-based Elo and the Elo in a model for dating apps. Equation (14) with the Elo chess-based representation for the dynamics of ranks becomes

$$\partial_t n(c, x, t) = x^\alpha \sum_{f=(w,j)=c} \int_0^1 \text{D}y K(w, j)v(w)n(w, x, t)n(j, y, t)$$

$$+ (1 - x^\alpha) \sum_{f=(w,j)=c} \int_0^1 \text{D}y K(w, j)v(w)n(w, x, t)n(j, y, t)$$

$$- n(c, x, t)v(c) \int_0^1 \text{D}y \sum_j K(j, c)n(j, y, t),$$

(18)
where \( D_y = dyP(y) \) and \( f^\pm(w, j) = w(1 - \zeta) + \zeta(j \pm \gamma) \). The product of the visibility \( v(c) \) and the kernel results in a non-trivial interaction between users as the ratio of the probability of interaction between users with Elo \( c \) and \( c' \) is \( v(c)/v(c') \), which is independent of the Kernel. Even in the case of the possibility of non-unitary jumps between ranks, high jumps are strongly minimized by the users Elo. In the following, we will keep, as in the previous section \( v(c) = c \), and a general form of the kernel will be taken as \( K(c, c') = K(c - c') = [1 + (c' - c)^2]^{-1} \). A Kernel defined in this way maximizes interactions between users with similar Elo and minimizes it, in case of high Elo difference. We ran numerical simulations of equation (18) and we require \( c \) to be bounded from below, but not above, as in the previous sections. We avoided the boundary at \( c = N \), as Elo score for users with \( x > x_C \), will not grow indefinitely due to the two-body interaction term. This term thus, as expected, stabilizes the ratings.

In figure 4(c), we show that the condensate has not disappeared. There is a smoother transition at the condensation threshold for the steady state average probability with respect to the attractiveness. Moreover, the number of likes received is more evenly distributed among users (figure 4(d)), similar to the ‘push’ algorithm. In figure 4(e), we show the distribution of Elo independent of attractiveness. We consider another possible scenario, in which users constantly change their acceptance probability, thus including noise effect and misjudgments. This is included in the model by choosing \( \alpha \) at every time step according to a Cauchy distribution with parameters (0, 1). The condensation threshold and the average Elo are not affected by noise (figure 4(a), inset), suggesting that the condensation mechanism is quite strong against intrinsic noise. Interestingly, the distribution flattens initially, and becomes bimodal in the long-term limit. The higher mode is broad, figure 4(e) (inset), signalling what we expected from studying the average rank: the Elo ratings are more proportionally distributed among the users with \( x > x_C \). At the same time, Elo ratings are not evenly distributed when we consider that the totality of users as the condensate is still present. This particular chess-based ranking system turns out not to be more advantageous for users with \( x < x_C \) than the ‘push’ algorithm in the previous section.

To conclude, this chess-based Elo ranking system seems the most realistic among the previously analysed models as the ratings reach a stationary distribution. In future works, it will be a reasonable starting point to analyse, for example, the effect of users installing and uninstalling the app.

6. Discussion

In this work, we studied the dynamics of likes and popularity (rank) in possible models for dating apps. Upon mapping the dynamics on this network to typical problems in stochastic processes we were able to show how the rank and number of interactions change in time. In particular, we analysed two possible ways in which users may decide whether or not to give a like to another user. In the first case, users decide based on the difference between attractiveness between themselves and the one proposed by the app. In this scenario, we found that the statistics of likes and matches are quite different as
users who are not the most popular, i.e. received fewer likes than other users, may have more matches. Although it may appear surprising, this phenomenon is a result of the combination between the distribution of average attractiveness and how likely a user is to give a like, equation (1).

As a second class of models, we consider that users decide solely on the other users’ attractiveness. This form of decision-making leads to a simple percentage of likes and matches only in the case where all of the users are equally visible. Instead, when users are shown differently based on their popularity, there might be situations in which a finite fraction of the users will receive few to no likes. This phenomenon, typical of condensation mechanisms, will set a threshold in attractiveness, below which users enter into the condensate. We found, by exploring different possible features of the app, that the condensation mechanism is hard to break, as it will always have a strong component related to the swiping strategy of the users. However, we found that whenever the app decides to increase with a certain frequency the rank of every user, the average rank and so the visibility will saturate to a constant value, smearing out the condensation phenomenon.

In the last section, we analysed models for dating apps in which users are ranked in a similar way as done in sports, such as chess or tennis. In such models, we found typical features of gelating systems. In particular, users will either end up in the condensates or they will experience an increase of the rank over time. On top, gelation leads to the accumulation of likes and rating to very few users. Finally, we use a chess-based Elo rating system, which is expected to prevent gelation. We found that, even in this scenario, the condensation mechanism does not disappear, but ranks and likes are more equally distributed between users above the condensation threshold.

Our approach, based on a mean-field mapping, though being expected to give no accurate statistics over longer times, still describes the qualitative features of the analysed model. In the future, it will be interesting to extend our approach to more complicated networks, such as bipartite networks. As an example, bipartite networks arise when a dating app asks users to identify themselves with respect to a gender. Our mean-field approach can be applied to bipartite networks as well, as it relies only on the dynamics of users in a ‘bath’ of other users belonging to one of the two layers of the network. In particular, if the two networks have different average values of the ‘pickiness’ \( \alpha \) or \( \beta \), we can replace the particular one in all our previous models, except the chess-based Elo. In this model, it is necessary to rescale ratings between the networks, to make them comparable. It will be interesting to better analyse this effect in other studies. Here we stressed that in all the analysed models, typical malicious behaviour, such as giving likes disregarding the personal preferences does not change the visibility of the users adopting that behaviour, as their rank or Elo score depends only on the choices of other users. Moreover, a dating app can simply rescale the rank, thus overcoming such behaviours.

Unfortunately, to our knowledge, there is no data available on the statistics of likes, matches and possibly ranks. It is then hard to point out which of these models better explain dating app algorithms. In the future, it will be interesting to explore different possible models with like-based visibility to find one that may smear out the condensation mechanisms outlined in this paper. As evident from the models analysed here, the best way to avoid condensation is not to have visibility criteria.
Acknowledgments

I thank I Di Terlizzi, A Kickuth, F Rost, F Pataconi, Y Qinghao, M Popovic and S Rulands for a critical reading of the manuscript. I thank L Duso and all the members of the Rulands lab for helpful discussions.

Conflict of interest

The author declares no competing interests.

Appendix A. Numerical methods

In this section, we give numerical details for the agent-based model simulations. In the simulations, we fix the number of users to be $N = 10^3$ and we let the system evolve for $10^7$ time steps. The initial ranks are set to $c_0 = N$ for every user. In case of an Elo-based ranking system, we initialize the Elo to $c_0 = 10^3$. In the chess-based Elo rating system, we set $N = 10^4$ and $10^8$ time steps in order to resolve the higher mode of the Elo rating distribution. We outline the numerical procedure for the model described by equation (5), as it is general and the application to other models is straightforward. We initialise the network with $c_i = c_0$, $J_{ij} = 0$, where $c_i$ is the rank of a user $i$ and $\hat{J}$ is the matrix of likes. At every iteration step, we select a user $i$. We then construct a vector of rates $\vec{\lambda}$, where the components $\lambda_i$ are the rates at which the user $i$ sees user $j$. For the given model, this rate is $\lambda_j = \gamma (c_j / N) + (1 - \gamma)$ and $\lambda_i = 0$. We then randomly choose a user $k$ with a probability $\lambda_k / \sum \lambda_k$. With a probability $x^\alpha_j$, the rank of user $j$ is increased by one, if less than $N$ and zero otherwise. $J_{ij}$ is set to one. With a probability $(1 - x^\alpha)$ the rank is decreased by one if $c_j > 0$ and unchanged otherwise. $J_{ij}$ is set to zero. The number of likes received by user $j$ is $n_j = \sum_i J_{ij}$. We then repeat this process and time is increased by one every $N$ steps, which is equal to the average time between two consecutive decisions of a user.

For the Elo rating system, $\lambda_j = \frac{c_j}{1 + (c_j - c_i)^2}$. After the pair is selected, with a probability $x^\alpha c$, the rank of user $j$ is updated to $c_j \rightarrow c_j + \beta \gamma + \beta(c_i - c_j)$ and $J_{ij}$ is set to one and with a probability $(1 - x^\alpha)$, the rank of the user $j$ is updated to $c_j \rightarrow c_j - \beta \gamma + \beta(c_i - c_j)$ and $J_{ij}$ is set to zero. Likes are computed as in the previous case. For all the simulations, unless stated otherwise, attractiveness is uniformly distributed in $[0, 1]$ and we computed average ranks, likes, matches for 50 linearly spaced bins of attractiveness.

Appendix B. Distribution of matches for biased decision-making

As the probability of a match $p(m_{ij})$ is given by the product of the probability of giving likes: $p(m_{ij}) = e^{-\beta |\delta_{ij}|}$, the cumulative distribution is

$$F(p(m_{ij}) \leq s) = \int_{-\log s / \beta}^{\log s / \beta} P(\delta_{ij}) d\delta_{ij}$$

https://doi.org/10.1088/1742-5468/ae9bed
where $P(\delta_{ij})$ is the distribution of the difference of attractiveness. The probability distribution is then $P(s) = \partial_s \int_{-\log s/\beta}^{\log s/\beta} P(\delta_{ij}) d\delta_{ij}$. Upon taking $\delta_{ij}$ to be Gaussian distributed with zero mean and unitary variance, after normalizing for $0 < s < 1$, we arrive at equation (2). In order to study the marginal probability of a match $p(m_i)$ for a user who is attractive $x_i$, we introduce the cumulative distribution

$$F_m(p(m_i) \leq s) = \int_{-\log s/\beta+x_i}^{\log s/\beta+x_i} P(x_j) dx_j,$$

where the integration is performed only on $x_j$ and $P(x_j)$ is the distribution of attractiveness. The marginal probability is given by $P(s) = \partial_s F_m(s)$, with $s$ replacing $m_i$ as before.

The average probability of a match is:

$$\langle n \rangle = \int_0^1 ds P(s).$$

In case of uniform distribution, the average probability of a match for an attractive user $x_i$ is

$$\langle n \rangle = \frac{1}{\beta} \left[ 2 - e^{\beta(x_i-1)} + e^{-\beta x_i} \right].$$

### Appendix C. Moment generating function

Upon multiplying the rhs and lhs of equation (5) by $z^{k_1}z_2^{k_2}z^c$ and summing over $k_1, k_2, c$, equation (5) is expressed in terms of the generating function $G(z_1, z_2, z, t)$,

$$\partial_t G = \left[ x^\alpha z_1^2 z_2 + (1 - x^\alpha) z_2 - z \right] \partial_z G.$$  

Moments are found by differentiation of the generating function,

$$\langle k_1(x, t) \rangle = \left[ \partial_z G(z_1, z_2, z, x, t) \right]_{z_1=z_2=z=1},$$

$$\langle k_2(x, t) \rangle = \left[ \partial_{z^2} G(z_1, z_2, z, x, t) \right]_{z_1=z_2=z=1},$$

$$\langle c(x, t) \rangle = \left[ \partial_{z^3} G(z_1, z_2, z, x, t) \right]_{z_1=z_2=z=1}.$$  

The dynamics of the moments from equation (C.1) are,

$$\partial_t \langle k_1(x, t) \rangle = x^\alpha \langle c(x, t) \rangle,$$

$$\partial_t \langle k_2(x, t) \rangle = (1 - x^\alpha) \langle c(x, t) \rangle,$$

$$\partial_t \langle c(x, t) \rangle = (2x^\alpha - 1) \langle c(x, t) \rangle.$$  

Upon solving the system of equations for $\langle c(x, t) \rangle$ and plugging the solution into the other moments, we arrive at equation (6) of the main text.

### Appendix D. Method of characteristics

Upon defining the generating function $G(z, x, t) = \sum_c z^c n(c, x, t)$, equation (12) is simplified to

https://doi.org/10.1088/1742-5468/ac9bed
Towards a statistical physics of dating apps

\[ \partial_t G(z, x, t) = \gamma \left[ z^2 x^\alpha + (1 - x^\alpha) - z \right] \partial_z G(z, x, t) + (1 - \gamma) [z^r - 1] G(z, x, t). \]  

Setting a characteristic \( s \) starting at time \( t = 0 \) and for some initial value \( z(s, t=0) = z_0 \), equation (D.1) is split as

\[ \frac{df}{ds} = 1 \]
\[ \frac{dz(s)}{z(s)^2 x^\alpha + (1 - x^\alpha) - z} = ds \]
\[ \partial_s G(z(s), x) = (1 - \gamma) [z(s)^r - 1] G(z(s), x). \]

From the first of these last equations, we find \( t = s \), and so \( z(s) = z(t) \).

From the second equation, we get

\[ z(t) = \frac{e^{(2x^\alpha-1)\gamma t} (1 - x^\alpha)(1 - z_0) - (1 - x^\alpha)(1 + z_0)}{e^{(2x^\alpha-1)\gamma t} x^\alpha (1 - z_0) - (1 - x^\alpha)(1 + z_0)}. \]

Putting the results into the third equation

\[ G(z, x, t) = G(z_0, x) e^{(1 - \gamma) \int dt [z(t)^r - 1]}. \]

If \( G(z, x, t = 0) = z_0^0 \) and \( r = 1 \), the solution is given by

\[ G(z, x, t) = \left[ \gamma (2x^\alpha - 1) \right]^\frac{1}{x^\alpha} \gamma^\frac{1}{x^\alpha} \left[ (1 - x^\alpha) (e^{(2x^\alpha-1)\gamma t} - 1) - z((1 - x^\alpha) e^{(2x^\alpha-1)\gamma t} - x^\alpha) \right]^\frac{1}{x^\alpha} \left[ (1 - x^\alpha) (e^{(2x^\alpha-1)\gamma t} - 1) - z x^\alpha (e^{(2x^\alpha-1)\gamma t} - 1) \right]^\frac{1}{x^\alpha+1}. \]

The asymptotic distribution for the fraction of users in the condensate is then given by \( (x < x_C) \)

\[ n_0(x) = \lim_{t \to \infty} G(z = 0, x, t) = \left[ 1 - \frac{x^\alpha}{1 - x^\alpha} \right]^\frac{1}{x^\alpha}. \]

The average rank can be computed via the derivative of the generating function \( \langle c(x, t) \rangle = \partial_z G(z, x, t)|_{z=1} \)

\[ \langle c(x, t) \rangle = \frac{\gamma + (2\gamma(x^\alpha - 1) + 1)e^{t(2x^\alpha-1)}}{\gamma(2x^\alpha - 1)} - 1. \]

**Appendix E. Product Kernel**

In terms of the generating function \( G(z, x, t) = \sum_c z^c n(c, t) \), upon multiplying by \( z^c \) and summing over \( c \) both sides of equation (18) for \( \beta = 1 \), for \( K(c, c') = cc' \), \( v(c) = 1 \), the
Figure E1. Divergence of the first (blue), second (orange), third (green) moment of equation (E.8). We set $x = 0.8, \alpha = 2, c_0 = 10$. We set $f_\alpha = 1$ as the qualitative behaviour is unchanged by the exact value.

Dynamics of $G(z, t)$ are

$$
\partial_t G(z, x, t) = \left[ x^\alpha z^2 + (1 - x^\alpha) - z \right] \partial_z G(z, x, t) \int_0^1 dy P(y) \left[ \partial_z G(z, y, t) + G(z, y, t) \right]_{y=1}. \tag{E.1}
$$

Equation (E.1) is a partial differential equation for the generating function, which depends on the zeroth and first order moments. We rescale time as $\tau = \int_0^t \int_0^1 dy P(y) \left[ \partial_z G(z, y, t) + G(z, y, t) \right]_{y=1}$. As outlined in the main text, we take a uniform distribution of attractiveness. Equation (E.1) in the rescaled time $\tau$ is

$$
\partial_\tau G(z, x, \tau) = \left[ x^\alpha z^2 + (1 - x^\alpha) - z \right] \partial_z G(z, x, \tau). \tag{E.2}
$$

Equation (E.2) is the partial differential equation for the generating function of a linear birth-death processes. The generating function is given by the method of characteristics (as in appendix D)

$$
G(z, x, \tau) = \left[ \frac{(1 - x^\alpha)(1 - z) - e^{-(2x^\alpha - 1)\tau}(1 - x^\alpha(1 + z))}{x^\alpha(1 - z) - e^{-(2x^\alpha - 1)\tau}(1 - x^\alpha(1 + z))} \right]_{t=q}, \tag{E.3}
$$

from which the density of users in the condensates is ($n_0(x, t) = G(z = 0, x, \tau)$)

$$
n_0(x, \tau) = \left[ \frac{(1 - x^\alpha) - (1 - x^\alpha)e^{(1-2x^\alpha)\tau}}{x^\alpha - (1 - x^\alpha)e^{(1-2x^\alpha)\tau}} \right]_{t=q}. \tag{E.4}
$$
Towards a statistical physics of dating apps

In order to derive the moment generating function in real time, we couple it with the equations,
\[
\partial_t\left[\partial_z G(z, x, \tau)\right]_{z=1} = (2x^\alpha - 1)\left[\partial_z G(z, x, \tau)\right]_{z=1}\left[\partial_t G(z, x, \tau)\right]_{z=1} = 0,
\]
with solution \[\partial_z G(z, x, \tau) + G(z, x, \tau)\]_{z=1} = c_0 e^{\tau(2x^\alpha - 1)} + 1. We then express the real and rescaled time as
\[
t(\tau) = \int_0^\tau dt\left(\int_0^1 dy\left[\partial_z G(z, y, \tau') + G(z, y, \tau')\right]_{z=1}\right)^{-1}.
\]
The integral in equation (E.6) saturates to a constant value for \(\alpha > 1\). Even though the integral cannot be solved exactly, an approximate asymptotic solution is found as
\[
\tau \approx \log\left(\frac{1}{(1 + c_0 f_\alpha)e^{-t} - c_0 f_\alpha}\right),
\]
where \(f_\alpha\) is a parameter that decreases with increasing value of \(\alpha\). Gelation typically occurs with divergence of higher order moments of the density of users \(n(c, x, t)\). Moments of order \(m\) can be computed from the generating function as
\[
\langle c^m(x, t) \rangle = \partial_z^m G(z, t)\big|_{z=1}.
\]

In figure E1, we plot the first three moments, which as expected, diverge at finite times for \(x > x_C\).

References

[1] Gilbert J P and Mosteller F 1966 Recognizing the maximum of a sequence J. Am. Stat. Assoc. 61 1966
[2] Little A C, Jones B C and DeBruine L M 2011 Facial attractiveness: evolutionary based research Phil. Trans. R. Soc. B 366 1638
[3] Orghian D and Hidalgo C A 2020 Humans judge faces in incomplete photographs as physically more attractive Sci. Rep. 10 110
[4] Furl N 2016 Facial-attractiveness choices are predicted by divisive normalization Psychol. Sci. 27 1379
[5] Germine L et al 2015 Individual aesthetic preferences for faces are shaped mostly by environments, not genes Curr. Biol. 25 2684
[6] Elo A 1978 The Rating of Chessplayers, Past and Present (New York: Arco Publication)
[7] Bruch E E and Newman M E J 2018 Aspirational pursuit of mates in online dating markets Sci. Adv. 4 eaap9815
[8] Knebel J, Weber M F, Krüger T and Frey E 2015 Evolutionary games of condensates in coupled birth-death processes Nat. Commun. 6 6977
[9] Bianconi G and Barabási A-L 2001 Bose–Einstein condensation in complex networks Phys. Rev. Lett. 86 5632
[10] Ispolatov S, Krapivsky P L and Redner S 1998 Wealth distributions in asset exchange models Eur. Phys. J. B 2 267
[11] Fiaschi D and Marsili M 2012 Distribution of wealth and incomplete markets: theory and empirical evidence J. Econ. Behav. Organ. 81 243
[12] Crawford F W and Suchard M A 2012 Transition probabilities for general birth-death processes with applications in ecology, genetics, and evolution J. Math. Biol. 65 553
[13] Kendall D G 1948 On the generalized ‘birth-and-death’ process Ann. Math. Stat. 19 1–15
[14] Crawford F W, Minin V N and Suchard M A 2014 Estimation for general birth-death processes J. Am. Stat. Assoc. 109 730
[15] Zheng Y, Chao X and Ji X 2004 Transient analysis of linear birth-death processes with immigration and emigration Probab. Eng. Inf. Sci. 18 141

https://doi.org/10.1088/1742-5468/ac9bed
Towards a statistical physics of dating apps

[16] Shahrezaei V and Swain P S 2008 Analytical distributions for stochastic gene expression Proc. Natl Acad. Sci. 105 17256

[17] Mugler A, Walczak A M and Wiggins C H 2009 Spectral solutions to stochastic models of gene expression with bursts and regulation Phys. Rev. E 80 041921

[18] Krapivsky P L, Redner S and Ben-Naim E 2010 A Kinetic View of Statistical Physics (Cambridge: Cambridge University Press)

[19] Van Dongen P G J 1987 On the possible occurrence of instantaneous gelation in Smoluchowski’s coagulation equation J. Phys. A: Math. Gen. 20 1889