Tracking the Gust Fronts of Convective Cold Pools

Fournier, Marielle B.; Haerter, Jan O.

Published in:
Journal of Geophysical Research - Atmospheres

DOI:
10.1029/2019JD030980

Publication date:
2019

Document version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
Tracking the Gust Fronts of Convective Cold Pools
Marielle B. Fournier1 and Jan O. Haerter1

1Niels Bohr Institute, University of Copenhagen, Copenhagen, Denmark

Abstract It is increasingly acknowledged that cold pools can influence the initiation of new convective cells. Yet the full complexity of convective organization through cold pool interaction is poorly understood. This lack of understanding may partially be due to the intricacy of the dynamical pattern formed by precipitation cells and their cold pools. Additionally, how exactly cold pools interact is insufficiently known. To better understand this dynamics, we develop a tracking algorithm for cold pool gust fronts. Rather than tracking thermodynamic anomalies, which do not generally coincide with the gust front boundaries, our approach tracks the dynamical cold pool outflow. Our algorithm first determines the locus of the precipitation event. Second, relative to this origin and for each azimuthal bin, the steepest gradient in the near-surface horizontal radial velocity \( v_r \) is employed to determine the respective locus of the cold pool gust front edge. Steepest \( v_r \) gradients imply largest updraft velocities, hence strongest dynamical triggering. Results are compared to a previous algorithm based on the steepest gradient in temperature—highlighting the benefit of the method described here in determining dynamically active gust front regions. Applying the method to a range of numerical experiments, the algorithm successfully tracks an ensemble of cold pools. A linear relation emerges between the peak rain intensity of a given event and maximal \( v_r \) for its associated cold pool gust front—a relation found to be nearly independent of the specific sensitivity experiment.

1. Introduction
As rain falls toward the ground, a fraction of it is evaporated into the unsaturated subcloud layer (Li & Srivastava, 2001; Lolli et al., 2017; Seifert, 2008; Srivastava, 1987). The resulting evaporatively cooled and therefore relatively dense volume is often referred to as a cold pool. The gravitational force due to the density increase, along with downdrafts, causes the air to descend and spread out horizontally along the surface. This outward propagation has been shown to resemble that of density currents (Charba, 1974). As the cold air spreads, the surrounding warmer air is forced upward leading to strong, positive vertical velocities at the cold pool gust fronts, which can trigger convection and the formation of clouds along these fronts (Purdom, 1976). Both observational and numerical studies have examined this triggering effect of cold pools (Feng et al., 2015; Jeevanjee & Romps, 2015; Khairoutdinov & Randall, 2006; Li et al., 2014; Rotunno et al., 1988). They find lifting of ambient air to occur through two possible causes: 1) interaction of cold pool gust fronts with environmental winds, commonly found for squall lines and long-lived multicellular thunderstorms (He et al., 2018; Rotunno et al., 1988; Torri et al., 2015; Weisman et al., 1988); 2) collision of several cold pool gust fronts (droegemeier & wilkinson, 1985; Feng et al., 2015; Kingsmill, 1995; Lima & Wilson, 2008). Apart from this dynamical triggering, cold pool formation also changes the near-surface thermodynamic properties (de zoete et al., 2017; terai & wood, 2013; Zuidema et al., 2017). Static stability is increased by the cooler air causing local suppression of convection in the cold pool interior. This cooler air acts together with the enhanced wind speeds to alter the heat fluxes between the surface and the atmosphere (gendine et al., 2016; Grant & van den Heever, 2018; Schlemmer & Hohenegger, 2016). Cold pools are sometimes surrounded by areas of positive moisture perturbations (Schlemmer & Hohenegger, 2014; Tompkins, 2001), which can locally lower the convective inhibition and increase buoyancy, thereby facilitating the triggering of new convection at the gust fronts. This triggering influences the spatio-temporal distribution of the cloud field. Clusters of clouds are produced near the cold pool gust fronts, thereby affecting the atmosphere’s radiative budget. Cold pools have been shown to have an impact on the transition from shallow to deep convection over land (Khairoutdinov & RANDALL, 2006; S. J. Böing et al., 2012; Schlemmer & Hohenegger, 2014). Here, deeper clouds and the largest rain intensities are not found before the late afternoon, even though large values of convective available potential energy (CAPE) are present (Nesbitt & Zipser, 2003). Large-scale climate models are heavily
dependent on quantities such as CAPE in order to simulated convection, because convection is not fully resolved. As a result, deep convection often develops too early in the diurnal cycle (Betts & Jakob, 2002).

Addressing this shortcoming, Rio et al. (2009) together with Grandpeix and Lafore (2010) incorporated a cold pool model into the convection parameterization in a single-column version of a general circulation model (GCM) and could partially remedy the shortcoming by achieving more realistic simulations of the precipitation diurnal cycle. Khairoutdinov and Randall (2006) showed, by removing cold pools (switching off precipitation evaporation) in a cloud-resolving model, that cold pools were required for producing sufficiently large thermals to support the growth of deep convective clouds.

Doubtlessly, despite these advances, there is further need for better understanding of the dynamics and evolution of cold pools and how they act to organize convection. The spatial distribution and subsequent convective triggering of cold pools can be examined by tracking how cold pools evolve throughout the day. This motivates the use of a tracking algorithm that can identify and follow cold pools throughout their entire lifetime. Given that the triggering of new convective events predominantly occurs at or near the cold pool gust fronts, these loci of strong horizontal convergence must be determined by such a tracking algorithm.

Other tracking methods have been developed in recent years: Schlemmer and Hohenegger (2014), Torri et al. (2015), and Gentine et al. (2016) used thresholds on temperature anomalies to identify cold pools and subsequent spatially connected regions. Similarly, using proxies for buoyancy, such as density potential temperature, $\Theta_r$, Tompkins (2001) and Feng et al. (2015) determined the spatial extent of the cold pools either subjectively (Tompkins, 2001) or by using an image processing technique (Feng et al., 2015). Common to these studies is that the thresholds are determined subjectively by visual inspection of the corresponding fields. A recent study by Drager and van den Heever (2017) sought to eliminate such subjective thresholds by using gradients in $\Theta_r$ rather than absolute values. Cold pools were thereby identified as closed boundaries defined by the zero contours of the second radial derivative of $\Theta_r$.

These tracking methods share the use of a thermodynamic quantity as cold pool identifier. Such an approach warrants the forcing, that is, potential energy, which could drive a cold pool initially at rest. However, especially during the final stages of the cold pool lifetime, cold pools expand due to the inertia gained much earlier, and the air near the gust fronts may no longer be anomalously dense: during the continuous spreading of a cold pool, the gust front experiences turbulent mixing and entrainment of environmental air. As a result, surface energy fluxes will act to increase the overall buoyancy of the cold pool. A recent study by Grant and van den Heever (2018) indeed shows that this effect is most pronounced near the cold pool gust fronts and that cold pool dissipation therefore proceeds from the outer edge inward. These factors reduce the difference in temperature and density between the cold pool and the environment. Despite this gradual thermodynamic equilibration process, the air at the cold pool edges will still maintain its inertia or could even be additionally forced outward by the dense air masses in the cold pool interior.

To be more specific, consider an example (Figure 1). The updraft associated with the cold pool gust front has advanced further than the temperature anomaly—a discrepancy explained by enhanced surface-to-atmosphere energy fluxes under the strong horizontal near-surface winds, as well as turbulent mixing within the cold pool gust front (Schlemmer & Hohenegger, 2014; Tompkins, 2001). In short, if one were to identify the cold pool using density measures, one would generally not detect the locus of convergence, where triggering of new convection is expected. Our current method remedies this issue.

The structure of this paper is as follows. We first describe our test data (section 2) and tracking algorithm (section 3) and then apply the tracking method based on the dynamical aspects of cold pools (section 4). We discuss the method in terms of algorithm performance, including a comparison with the thermodynamic-based tracking algorithm developed by Drager & van den Heever, 2017 (2017; hereafter DvdH17) and the temporal evolution during a simulated diurnal cycle. Section 5 concludes and offers examples of where the method could usefully be applied.

### 2. Materials and Methods

To test the method, we use an idealized diurnal cycle simulation, which mimics midlatitude summertime convection. This transiently varying simulation setup was chosen, as it generates cold pools of various spreading velocities and length scales, which vary over the course of the day (Haerter et al., 2017).
2.1. Simulation Setup

The convective atmosphere was simulated using the University of California, Los Angeles, Large Eddy Simulation model with subgrid-scale turbulence parameterized using the Smagorinsky model, a delta four-stream radiation scheme and a two-moment cloud microphysics scheme (Stevens et al., 2005). Rain evaporation is implemented after Seifert (2008). As detailed in Moseley et al. (2016), diurnally oscillating surface temperature ($T_s(t)$) boundary conditions are applied, with

$$T_s(t) = T_{\checkmark} - T_a \cos \frac{\pi t}{t_0},$$

(1)

with $T_{\checkmark}$ the average surface temperature, $T_a$=10K the surface temperature amplitude, and $t_0$ the duration of the simulated model day. All simulations were initialized with data from observed summertime midlatitude conditions where convection had occurred in order to establish an initially unstable atmosphere. As in Moseley et al. (2016), we vary

- $T_{\checkmark}$∈{23, 25, 27} °C, yielding simulations denoted as CTR, p2K, and p4K, respectively, and
- keeping $T_{\checkmark}$ fixed to 23 °C, varying $t_0$∈[1,2] day modifies the buoyant instability and the duration, over which cold pools can organize, respectively. The “longer day” simulation with $t_0$=2 day is referred to as LD.

Increasing $T_{\checkmark}$, but not the atmospheric initial conditions, corresponds to greater convective instability due to the adjustment time required for the atmosphere to reach equilibrium.

The model numerically integrates the anelastic equations of motion on a regular horizontal domain (see Table 1 for domain sizes) with 200-m horizontal grid spacing and periodic boundary conditions. The model spans 75 stretchable vertical levels with spacings: 100 m below 1 km height, 200 m between 1 and 6 km, and 400 m between 6 km and the domain top, located at 16.5 km. Additionally, the model uses a sponge layer above 12.3 km. The horizontal domain sizes (Table 1) were chosen in order to obtain sufficient statistics to distinguish the effect of the different surface boundary conditions.

The Coriolis force and the mean wind were set to zero with weak random initial perturbations added as noise to break complete

### Table 1

<table>
<thead>
<tr>
<th>Numerical Experiment</th>
<th>Domain area (km x km)</th>
<th>$N_{cp}$</th>
<th>Rain duration (hr)</th>
<th>CP number density (km$^{-2}$/hr)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTR</td>
<td>205x205</td>
<td>444</td>
<td>6.1</td>
<td>.0017</td>
</tr>
<tr>
<td>p2K</td>
<td>205x205</td>
<td>1389</td>
<td>8.4</td>
<td>.0039</td>
</tr>
<tr>
<td>p4K</td>
<td>192x192</td>
<td>1470</td>
<td>9.8</td>
<td>.0041</td>
</tr>
<tr>
<td>LD</td>
<td>102x102</td>
<td>495</td>
<td>15.2</td>
<td>.0031</td>
</tr>
</tbody>
</table>

Note: Experiment names are as explained in section 2. $N_{cp}$ is the total number of cold pools detected in each simulation. Rain duration is defined as the time during which cold pools can be produced, that is, from the first identified rain event till the last. The cold pool number density is the number of cold pools detected divided by the respective domain area and the rain duration.
translational symmetry. No large-scale forcing was imposed, ensuring that the only driving force for convection was buoyancy and the forced lifting through cold pool interaction. The model output time step was $\Delta_{\text{out}}=5 \text{ min}$. At each output time step, 5-min-accumulated precipitation and instantaneous velocities were output at each model gridbox.

3. Cold Pool Tracking Algorithm

3.1. Dynamical Approach

Cold pools are characterized by approximately circular expansion of the near-surface horizontal wind field, with winds directed radially outward from the precipitation event (Figure 2): Horizontal velocity vectors point in radial directions and, due to the anelastic continuity equation,

$$\rho_0 \frac{\partial (rv_r)}{\partial r} + \frac{\partial (w \rho_0)}{\partial z} = 0,$$

with $\rho_0$ the mean state air density and $r$ the radial coordinate, the magnitude of radial velocities decreases in locations where vertical velocity is largest. In the Figure 2, these locations are seen as a pronounced red ring of approximately 500-m thickness. This ring constitutes a demarcation line between the organized, strong, wind pattern within, and less organized, weaker wind outside of this line. Additionally, note that the vertical wind in the interior of this ring is generally negative, corresponding to downdraft regions where further convection is generally inhibited. Virtual potential temperature anomaly $\theta'_v$ is often used as a measure of buoyancy. Comparing again with Figure 1, where the line $\theta'_v = 0$ lies one or several kilometers closer to the horizontal origin than the locus of maximum updraft, it is clear that $\theta'_v = 0$ would often identify regions of suppressed, rather than favored convection.

To detect locations of strong gust front vertical velocity, at each time step our algorithm hence locates points of sharpest decrease in radial velocity, hence largest resulting vertical velocity. In the following, we denote these locations as edges and generally associate them with the actual cold pool gust fronts.

3.2. Algorithm

In contrast to the cold pool gust fronts we want to detect, the precipitation events that cause these gust fronts are easily discernible, as they form sharp horizontal boundaries. Patches of surface precipitation can be identified as horizontally and temporally contiguous areas of nonzero surface precipitation. Cold pools systematically emanate from these patches, termed parent rain events, and each cold pool can therefore be uniquely associated with a specific parent rain event. As will be shown, by using these parent rain events as the spatial and temporal reference for the corresponding cold pool detection, the gust fronts can be mapped out much more systematically.

The advantage of considering precipitation and cold pools on the same footing is twofold: (i) The relation between surface precipitation and the cold pool properties can be studied. (ii) The parent rain event serves as a natural origin, both in space and time, for the emergent cold pool.

The algorithm is split into three phases:

1. rain cell tracking,
2. computation of $v_r$ and $\partial v_r/\partial r$, and
3. identification of cold pool edges based on the minimum in $\partial v_r/\partial r$.

3.2.1. Rain Cell Tracking

Rain cells are tracked using the iterative rain cell tracking (Moseley et al., 2013). The iterative rain cell tracking detects spatially contiguous rain events with surface rainfall rate above a set threshold on intensity, $I_\text{th}$, at each time step. Subsequently, rain tracks, temporally contiguous patches, are identified by considering grid box overlaps for the rain objects forward and backward in time.

We find that rain events that cover very small surface areas do not produce detectable cold pools and our method therefore makes use of a lower areal threshold of 50 contiguous grid cells. This corresponds...
3.2.2. Computation of \( \nu_r \) and \( \partial \nu_r / \partial r \)

For each rain cell and each simulation output time step, we define the origin \( O=(0,0) \) as the precipitation intensity-weighted center of mass (COM). As the rain cell area generally changes from one time step to the next, its COM will also undergo corresponding displacements. At any position \( r \) relative to \( O \), we obtain the time-dependent radial wind speed by computing the projection

\[
\nu_r(r) \equiv v_b(r) \cdot \hat{r},
\]

for each horizontal point \( r=(x,y) = r \cos \phi \hat{x} + r \sin \phi \hat{y} \) within the model domain, where \( r \equiv |r| \), \( \hat{r} \equiv rr^{-1} \) is the unit vector in the direction of \( r \) (Figure 3a) and \( \phi \) is the azimuth. The velocity vector \( v_b(r) = (v(r),u(r)) \) is the horizontal velocity at the position \( r \), where \( v \) and \( u \) are the velocity components in the \( x \) and \( y \) directions.

It is now convenient to rotate the coordinate system so that the vector \( r \) is oriented along \( \hat{x} \). This is accomplished by applying the 2x2 rotation matrix \( \mathcal{R}(-\phi) \) to all horizontal vectors. The radial unit vector \( \hat{r} \) is then mapped into \( \hat{x} \), and the derivative in the radial direction just becomes a scalar derivative. The derivative is obtained for each \( r \) using central finite differences of fourth-order accuracy

\[
\frac{\partial \nu_r(r)}{\partial r} \approx \frac{\nu_r(r-\Delta s)-8
\nu_r(r-2\Delta s)+8\nu_r(r+2\Delta s)-\nu_r(r+\Delta s)}{12},
\]

where \( \Delta s \) is the model’s grid spacing (\( \Delta s=200 \) m here; see section 2). \( \nu_r(r+\Delta s) \) and \( \nu_r(r+2\Delta s) \) are determined by bilinear interpolation between the respective grid points.

As our current simulations do not employ external wind shear, the horizontal displacement of the origin from one time step to the next is small. Yet these small displacements often are important when computing a radial quantity and determining minima in derivatives—we hence find it useful to consider them. Additionally, allowing for a time-dependent COM makes the algorithm more versatile and applicable to simulations with different setups, such as boundary conditions with large-scale wind shear, where the displacements of the COM are surely much larger.

3.2.3. Identification of Cold Pool Edges Based on the Minimum in \( \partial \nu_r / \partial r \)

After computing \( \nu_r \) and \( \partial \nu_r / \partial r \), our method now determines the positions of the cold pool edges. While cold pools are often approximated as perfectly circular objects (Grandpeix & Lafore, 2010; Romps & Jeevanjee, 2016), circularity is a poor approximation as soon as they deform under collision, and the formation of new cold pools is triggered (Torri & Kuang, 2019). Thus, in order to correctly track all shapes, the area surrounding each cold pool is subdivided into a unit circle centered at the cold pool COM, which is then split into 32 azimuthal intervals, termed \( \text{slices} \), each spanning \( \phi=11.25 \) ° (Figure 3b). Different slice sizes were tested regarding the identification of a single cold pool, and the one yielding edges encircling the maximum radial cold pool velocity was chosen. The relatively small values of \( \phi \) indicates considerable heterogeneity of...
the dynamical cold pool edge contour, which we mainly attribute to deformations from mostly circular shapes during the initial phase of spreading, to more Voronoi graph-type structures upon collisions between cold pools Haerter et al. (2019).

Subsequently, the cold pool COM and the surrounding grid points are aligned in space by rounding the coordinate of its COM and \( r \) to the nearest integer multiple of \( \Delta s \)—in effect constituting a radial binning. The binning ensures that \( \partial v_r / \partial r \) can be smoothed in each of the resulting slices by averaging at each radius (Figure 3b). The edge is then identified in each slice by locating the radius with minimum \( \partial v_r / \partial r \), termed \( r^* (\phi_j) \), where \( \phi_j \) denotes the central azimuth of a given slice. Note that one could equivalently detect zeros in \( \partial^2 v_r / \partial r^2 \) and require a negative third derivative. Numerically, this option was however found more cumbersome. The approach we employ here ensures that at least one edge point is detected in each slice and that a closed contour surrounding a given cold pool can always be mapped out.

The location of the edge in one slice is constrained by the location of the edge in the previous slice by a neighbor constraint: This constraint limits the radial search range for \( r^* (\phi_j) \) to

\[
r^* (\phi_{j-1}) - dr \leq r(\phi_j) \leq r^* (\phi_{j-1}) + dr.
\]

(5)

where \( dr = 3 \Delta s \). \( dr \) is fitted to the model horizontal resolution used in this study and should be revised if used with another resolution.

Consistency checks. Repeated checks of the following four constraints are performed after the initial identification of the edge in a slice.

1. **The edge must not be located at negative \( v_r \):**

   \[
   v_r (r^*) > 0.
   \]

   (6)

   We generally disregard contracting azimuthal slices (where \( v_r < 0 \)). Examining Cold Pool A in Figure 4, an edge identified where \( v_r < 0 \) would imply that the edge is found in the interior of Cold Pool B. If this occurs, the algorithm disregards the identified edge, and the radial search range determined by the neighbor constraint (equation (5)) is moved further toward the cold pool COM.

2. **\( v_r \) at positions immediately surrounding the identified edge should not be larger than \( v_r \) at the edge:**

   \[
   v_r (r) \leq v_r (r^*) \text{ for } r^* \leq r + dr.
   \]

   (7)

   where \( dr \) was chosen to be the distance covered by three pixels in the respective azimuthal direction. \( v_r \) does not increase monotonically from the cold pool COM to the edge (see white box in Figure 4). Turbulent mixing and surface energy fluxes alter the cold pool as it expands, generally yielding multiple maxima in \( v_r \) and
3.3. Performance Evaluation

In order to assess the ability of the tracking to correctly identify and track cold pools, the average near-surface radial structure of the simulated cold pools is examined and compared to results obtained in earlier studies and specifically to results from DvdH17. To proceed, composite statistics are computed, where different thermodynamic and dynamical quantities are averaged in both time and space for all identified cold pools:

1. For each cold pool, the data at \( z=50 \) m (\( z=100 \) m for vertical velocity) are interpolated onto a cylindrical grid of \( (r,\phi) \) with the COM positioned at \( r=0 \).
2. The angular average of the interpolated data across \( \phi \) at each \( r \) is computed for every time step during the cold pool lifetime.
3. The resulting radial profiles are averaged at each lifetime for all cold pools.

The main focus of our tracking algorithm is to identify the large, forced updrafts at the cold pool edges. Therefore, the average radius of all cold pools at each time during the cold pool lifetime is computed and used as an identifier for the average location of the edge points in the composites. The average radius is computed as follows:

\[
\langle r_c \rangle \equiv \frac{\sum r_k}{N},
\]

where \( r_k \) is the radius of a single cold pool at a specific lifetime \( \tau \), \( \langle r_c \rangle = \sum d_k / n \), where \( n \) is the number of edge points identified for the cold pool in question and \( d_k \) the distance from that cold pools COM to the \( k \)th edge point) and \( N \), the total number of cold pools at lifetime \( \tau \).

The average radius is computed both using the tracking algorithm developed in this study and an approximate version of the algorithm developed by DvdH17. Recall that DvdH17 identified cold pool edges as closed boundaries of the zero contour of \( \partial^2 \theta_p / \partial r^2 \). Their edge points are determined by

1. for each cold pool the \( \theta_p \) field at \( z=50 \) m is interpolated onto a cylindrical grid of \( (r,\phi) \), again with the COM defining the origin of \( r \);
2. \( \partial^2 \theta_p / \partial r^2 \) is computed to the same accuracy as \( \partial v_r / \partial r \);
3. the azimuthal average across all \( \phi \) at each \( r \) is computed resulting in an average radial profile of \( \partial^2 \theta_p / \partial r^2 \);
4. using polynomial regression a smooth curve is fitted to the averaged radial profile; and

The purpose of this constraint is equivalent to that of Constraint 3, but it is necessary for all slices where the COM of adjacent cold pools is not directly located in the slice.

\[ v_r(r) > 0 \ \forall \ \ r \leq r^* . \] (8)
5. The edge is identified by determining the location of zero crossing from positive to negative values, that is, a local maximum of $\partial \theta_r / \partial r$, closest to the center of the smoothed curve (see Figure 8 of DvdH17 for clarity).

4. Results

Generally, larger surface temperature forcing results in a larger number of precipitation events and subsequent cold pools per unit area—with areal and rain duration cold pool density varying by more than a factor of 2 for the warmer surface temperature simulations, in principle resulting in substantially altered network of gust fronts for all the different simulations (Table 1). We deliberately developed our tracking method only on one simulation but then applied it to the others, where $T_s$ and $t_0$ were varied, in order to test for robustness. Section 4.1 assesses the algorithm performance by examining results from the simulation performed with p2K ($T_s = 25^\circ$ C and $t_0=1$ day), and section 4.2 further examines these results and compares them with results for all remaining numerical experiments.

4.1. Algorithm Performance

Visually, cold pools can often be distinguished by examining either the $v_r$ or virtual potential temperature, $\theta_v$, fields. The edges identified by the current method qualitatively constitute a reasonable...
outer boundary in both cases (Figures 5a and 5b, blue contour line)—large positive values of $v_r$ and low $\theta_v$ are seen within the detected edge while the surroundings are characterized by small $v_r$ and larger, more homogeneous, temperatures.

The initial identification of the rain object, which provides a proper spatial reference, together with the sharp gradients in $v_r$ at interfaces between closely positioned cold pools, allows the algorithm to both clearly identify single cold pools and distinguish distinct cold pools from one another (Figures 5c and 5d). Together with the fact that the algorithm determines the exact horizontal coordinates of the gust fronts, our findings lead us to conclude that the algorithm is suitable for studies of cold pool collisions.

Occasionally, cold pool gust fronts appear to be identified incorrectly. This occurs most frequently late in the cold pool lifetime where the gradients in the $v_r$ field have weakened due to friction and turbulent mixing with the environment. Multiple minima in $\partial v_r/\partial r$ of comparable sizes can result in identification of edges at intuitively wrong locations (e.g., Figure 6, downward pointing arrows—a sudden “jump” of several kilometers is visible in the identified edge contour surrounding the cold pools). As we will discuss below, the occurrence of detection uncertainty during the late stage of a cold pool’s lifetime could be less detrimental to the analysis of

![Graphs](image_url) Figure 7. Temporal and radial evolution of radial and vertical velocity and rain intensity. (a) Surface $v_r$ as function of $r$ at various instances during the cold pool growth phase (see legend). (b) Similar to (a), but during the cold pool dissipation phase. Blue and red circles indicate the averaged radial position of the edges determined by our algorithm and that of DvdH17, respectively. (c/d) Similar to (a)/(b), but for vertical velocity at $z=100$ m and surface rain intensity, respectively. The dashed line in (e) represents the precipitation cutoff. Note that, due to possible deformations of cold pools during collisions, the azimuthal averaging used in (a)-(d) is expected to lead to some smearing out of the peaks, somewhat diminishing the amplitude of the curves. Data: simulation p2K (see section 2).
dynamical effects—such as triggering of new convective cells, which is expected to be more likely when the gust front momentum is larger.

Additionally, a large gradient can occasionally exist in the interior of the cold pool due to the COM not being identified directly in the center of the radial expansion or because the cold pool experiences multiple centers, that is, multiple locations of intense rainfall. This can result in the algorithm identifying the cold pool edge very close to the COM (e.g., Figure 6, upward pointing arrows). Future work could explore improvements regarding the identification of the cold pool COM, for example, by basing it on the wind field generated by the cold pool.

If one is interested in the general evolution and structure of cold pools, the errors introduced by the weak gradients are averaged away since the fraction of erroneously identified cold pools appears to be quite small (Figure 6).

4.2. Cold Pool Characteristics

We first aim to compare cold pool gust front detection through dynamics (now termed dynamical edge) to the detection through thermodynamics (termed thermodynamic edge). Recall that our algorithm determines cold pool edges through steep gradients in \( v_z \) (Figures 7a and 7b). As expected by continuity (equation (2)), these detected edges correspond to peaks in vertical velocity (blue symbols in Figures 7c and 7d). The thermodynamic edges, in contrast, are located further toward the cold pool COM. The vertical velocities found at
these edges and just downwind of them are in fact negative, suggesting subsiding and hence stable conditions there.

Using our condition on dynamics yields detected cold pools that expand during their entire lifetime (Figures 7a and 7b), consistent with the setup of the tracking only recording cold pools while they are dynamically active (see section 3.2). The dynamical edges are found at or very close to the maximum negative gradient of $v_r$ while the thermodynamic edges align well with the peak in $v_r$. The peak in $v_r$ should, however, be located upwind from the cold pool edges, while the front is characterized by rapidly decreasing $v_r$.

Additionally, the density difference between the front and the environment induces a vorticity perturbation acting to increase the surface horizontal winds behind the front and the vertical winds at the front (see, e.g., Figure 29 in Wakimoto, 1982). In the following, we distinguish a growth and a dissipation phase, characterized by times where the intensity of the cold pool parent rain event increases or decreases.

**Dynamics.** We first consider dynamical features: The growth phase roughly corresponds to times when cold pool expansion accelerates; that is, the peak $v_r$ increases (Figures 7a and 7e), while the opposite is the case during the dissipation phase (Figures 7b and 7f). These findings are explained by the increasing cooling near the COM during the growth phase, causing increasing gravitational forcing there—and vice versa for the dissipation phase. During both phases, the radial velocity at the dynamical edge is of similar magnitude ($v_r \approx 1$–1.5 m/s), and remarkably constant during the dissipation phase. However, vertical velocities during the growth phase can be a factor of 3 larger than for comparable radial velocities during the dissipation phase (cf. Figures 7c and 7d). Also, this feature can be made plausible, when considering that velocities near the COM continue to increase, hence forcing more mass outward. Mass conservation (equation (2)) must then imply appreciable vertical mass fluxes to make up for the increasing forcing during the growth phase. This is an interesting finding, as it might imply that dynamical triggering of new convective events should be expected during the growth rather than the dissipation phase—hence early in the cold pool life cycle.

**Thermodynamics.** Cold pools are typically characterized by negative buoyancy and negative temperature anomalies in their interior (Figure 8). As expected by their definition, the thermodynamic edges are located near the maximum positive gradient in both temperature and buoyancy (Figures 8a–8d)—and generally still negative buoyancy at appreciable distances (1–2 km) downwind from the thermodynamic edges. In contrast, dynamical edges are associated with more modest buoyancy. In the growth phase (Figures 8a, 8c, and 8e) the dynamical edge constitutes an almost perfect demarcation between negative buoyancy, within the detected dynamical edge, and positive buoyancy, surrounding this edge. In the dissipation phase (Figures 8b, 8d, and 8f), the buoyancy is negative essentially throughout, a feature attributable to the advection of the negative temperature anomaly in the COM of each cold pool, caused by the respective parent rain event.

In both phases, the detected cold pools are surrounded by a band of positive water vapor anomaly, which is advected radially outward as the cold pool expands, while the center becomes increasingly dry (Figures 8e and 8f). This is consistent with Tompkins (2001), who attributes the drying in the center to the transport of dry air from above cloud base by downdrafts. During the dissipation phase, the cold pool signal in all variables gradually fades. Inspecting Figures 8b, 8d, and 8f, it is worth pointing out that the dynamical edge is still associated with positive moisture anomalies, while the thermodynamic edge occurs at dry locations. This may be due to moist subcloud air, resulting from rain evaporation, which is quickly advected toward the dynamical edge and makes for a measurable moisture increase there (Tompkins, 2001; Torri & Kuang, 2016).
Features independent of boundary conditions. Our cold pool tracking allows us to compare cold pool characteristics under different boundary conditions (section 2).

Cold pool expansion. First, we consider the evolution of the mean radius (Figure 9a). In all experiments, expansion is initially quite rapid (≈10 km/hr) but settles to nearly constant, more modest, speed for all simulations after a few minutes. As most of the cold pool lifetime occurs during the dissipation phase, the finding of near-constant expansion speed is in line with near-constant $v_r$ during the dissipation phase (Figure 7b). The behavior in all four simulations is rather similar, with no systematic deviation between the curves. In terms of updraft velocities (Figure 9b), all simulations show a clear peak near the time of peak precipitation intensity (≈30 min).

An emergent finding from our analysis hence is for all boundary conditions alike; dynamical triggering effects may be expected to be most pronounced approximately 30 min after cold pool initiation, at which time cold pools have spread to a 5-km radius, are neutrally buoyant at the front (Figures 8c and 8d), and have appreciable positive moisture anomalies there (Figures 8e and 8f). Using the thermodynamic edge (see Figure 9a, dashed line), many of these triggering effects may not be detectable, as the thermodynamic edge becomes all but stagnant after approximately 10 min. In our algorithm we were able to detect cold pools up to ages of 90 min; we however caution that the signal-to-noise ratio of the dynamical quantities becomes weaker for even older cold pools.

Figure 10. Relation between maximum precipitation intensity and radial velocity. Panels (a)-(d) show scatter plots for the four simulations, as labeled, where each symbol represents a single cold pool. Solid black and red lines represent linear fits to the respective individual data and a fit to all data combined, respectively. Coefficients in the top left corner of each panel denote the overall slope and slope for the individual simulation, respectively, in units of meters per second and millimeters per hour. Shaded curves along the horizontal and vertical axes of each panel indicate the normalized histograms of $I_{max}$ and $v_{r,max}$ corresponding to each experiment.

10.1029/2019JD030980
Relation between rain events and their cold pools. For many applications, such as the parameterization of cold pools in GCMs, it may be useful to obtain generic relations between precipitation events and the resulting cold pools. Recently, it was reported that large-eddy-simulated convective precipitation cells show rather generic statistical relations, such as a proportionality between event precipitation intensity and event effective radius (Moseley et al., 2019). Using the current cold pool tracking, one can now relate the characteristics of the precipitation event to those of the cold pool for the different numerical experiments. The maximal spreading velocity at the dynamical edge, $v_{r,\text{max}}$, is a good measure of a cold pool's kinetic energy. For all simulations, $v_{r,\text{max}}$ shows a similar, and nearly linear dependence on maximal precipitation intensity $I_{\text{max}}$ for the parent precipitation event (Figure 10). The analysis shows that larger surface temperature forcing (p2K or p4K) or forcing that is applied over a longer duration (LD) lead to heavier precipitation events, which in turn generate more rapid cold pool expansion—however, the relation between the two quantities remains all but unchanged within the four different sensitivity experiments. With the proportionality of event effective radius $r_{\text{eff}} \sim I_{\text{max}}$ (Moseley et al., 2019), a square root dependence between event area and $v_{r,\text{max}}$ is further implied. We find that this holds reasonably well, when using the maximum event area $A_{\text{max}}$ of the parent rain event, that is, $v_{r,\text{max}} \sim A_{\text{max}}^{1/2}$.

5. Summary and Conclusion

We have presented a cold pool tracking algorithm, which detects the dynamical edge, one characterized by the convergence lines surrounding each cold pool, after the cold pool has been generated by a precipitation event. The dynamical edge is distinct from the edges identified in previous cold pool tracking methods (Drager & van den Heever, 2017), where a temperature-based edge has been employed. The motivation for using such dynamical edges is the finding that updrafts often result in regions of strong near-surface convergence, and dynamical triggering of new precipitation events is expected to be facilitated at these locations.

Due to entrainment and turbulent mixing of environmental air, this dynamical cold pool edge can exhibit thermodynamic properties quite similar to those of the environmental air in the surroundings, making the gust front all but undetectable through thermodynamic approaches. For thermodynamic edges, which we track for comparison, our findings conversely indicate pronounced stability due to downdrafts and relatively cool boundary layer conditions. Thermodynamic edges may hence be useful in describing areas where convection is suppressed. In future studies on cold pool-precipitation dynamics, it could hence be beneficial to combine both approaches to characterize cold pool regions of enhanced stability and enhanced triggering potential.

To summarize, our algorithm tracks cold pools throughout their lifetime. The method is simple, as it requires only the tracks of surface precipitation, yielding precipitation-weighted COM coordinates for each time step and rain cell, as well as the two-dimensional near-surface horizontal wind field. For the unit circle surrounding a given precipitation area COM, the tracking breaks down the azimuthal range into slices of equal angular range. Within each slice a maximum of radial velocity change is identified—corresponding to radii of maximum convergence. Several checks are applied at each time step, to ensure that noise does not strongly perturb the detection of each cold pool gust front.

Our tracking successfully identifies an ensemble of cold pools produced by running large eddy simulations for a diurnal cycle case, initialized with soundings from midlatitude potentially convective summer days (section 2). In order to assess the performance of the current dynamic-based tracking algorithm in relation to a thermodynamic-based one, a comparison with the tracking algorithm developed by Drager and van den Heever (2017) is performed. The average edge location for all cold pools at each time during their total lifetime is computed for both methods. In general, the edges determined using the dynamical edge are located radially further away from the cold pool center than the ones determined using the thermodynamic method. As the present method involves checks to ensure that one cold pool edge must not be located within the interior of another cold pool, a general overestimation of the radii detected by the present method can be excluded. In conclusion, our results suggest that the dynamics-based method allows for more complete tracking of the area enclosed by each cold pool gust front.

In practice, our analysis shows that after approximately 30 min the thermodynamic edges all but cease to advance further, only reaching a maximum radius of approximately 2 km, whereas the dynamical edges continue to advance (Figure 9a). This finding has implications when examining the dynamics within the interior.
of cold pools, at positions between the center and the front (Figure 7) as well as the average radial structure of temperature, buoyancy, moisture, and vertical velocity (Figure 8). The dynamical edges collocate with only weakly negative, or even neutral, buoyancy and generally positive moisture anomalies. As the gust fronts detected are further located near updraft maxima, they likely form “hotspots” for triggering of new precipitation events.

We exemplify the scope of our method by drawing a linear relation between event maximum precipitation intensity and the peak expansion speed of the resulting cold pool—a relation that is nearly unchanged for the different simulations. Such relations may be a first step toward mechanistic parameterization of cold pool dynamics in GCMs. A possible challenge might be strong wind shear, which could change the shape and symmetry of cold pools and the location of the rain event COM relative to the gust front. In the specific case of squall lines it could be of interest to only consider the cold pool loci with the largest gradient in radial velocity, often facing the direction of squall line propagation. These loci might locally still be approximated by circle segments, therefore still allowing them to be identified by our algorithm. Another challenge not addressed here is the case of merging cold pools where the corresponding rain events do not merge. These challenges should be addressed in a subsequent paper.

Despite the limitations discussed, our algorithm could in principle be used to track cold pools in observational data. Rain tracks can be easily measured by radar, but records of wind speed at high spatial resolution are currently not common. A useful setup for field studies could be to select an area of at least 10 km x 10 km with frequent occurrence of convective events, homogeneous surface conditions, and a network of narrowly spaced (kilometer scale) wind measurements.

Needless to say, further analysis should now follow. It is important to close the precipitation-cold pool feedback loop by detecting the influence of cold pool gust fronts on the generation of new precipitation events. Attempts have been made at describing self-organization through cold pools in conceptual models (Böing, 2016; Grandpeix & Lafore, 2010; Haerter & Schlemmer, 2018; Haerter et al., 2019). With more mechanistic information in place, a full, physics-based, cellular automaton, mimicking the self-organization of convective cold pools in space and time, could be built. Beyond this, the details of convective self-organization, such as clustering and the formation of extreme events, could be deciphered from analysis that builds on the current method.

References


