Qualitative Spatial Representation and Reasoning in Angry Birds: First Results

Peng Zhang and **Jochen Renz**
Research School of Computer Science
The Australian National University
Canberra, Australia
{u4969566, jochen.renz}@anu.edu.au

Abstract

Angry Birds is a popular video game where the task is to kill pigs protected by a structure composed of different building blocks that observe the laws of physics. The structure can be destroyed by shooting the angry birds at it. The fewer birds we use and the more blocks we destroy, the higher the score. One approach to solve the game is by analyzing the structure and identifying its strength and weaknesses. This can then be used to decide where to hit the structure with the birds.

In this paper we use a qualitative spatial reasoning approach for this task. We develop a novel qualitative spatial calculus for representing and analyzing the structure. Our calculus allows us to express and evaluate structural properties and rules, and to infer for each building block which of these properties and rules are satisfied. We use this to compute a heuristic value for each block that corresponds to how useful it is to hit that block. We evaluate our approach by comparing its performance with the winner of the recent Angry Birds AI competition.

1 Introduction

Qualitative spatial representation and reasoning has numerous applications in Artificial Intelligence including robot planning and navigation, interpreting visual inputs and understanding natural language [Cohn and Renz, 2008]. In recent years, plenty of formalisms for reasoning about space were proposed [Rajagopalan, 1994; Liu, 1998; Renz and Ligozat, 2005]. An emblematic example is the RCC8 algebra proposed by Randell *et al.* [1992]. It represents topological relations between regions such as "x is disconnected from y"; however, it is unable to represent direction information such as "x is on the right of y" [Balbiani *et al.*, 1999]. The Rectangle Algebra(RA) [Mukerjee and Joe, 1990; Balbiani *et al.*, 1999], which is an extension of the Interval Algebra(IA) [Allen, 1983], can express orientation relations and at the same time represent topological relations, but only for rectangles. When we want to reason about multiple aspects of relations between regions, a possible method is to combine several formalisms. For example, when we want to reason about topology and direction relations of regions with arbitrary shapes, we can combine RCC8 and RA. It has been shown that the problem of deciding consistency of a joint basic network of RCC8 and RA constraints is still in polynomial time [Liu *et al.*, 2009]. However, if we only consider the maximum bounding rectangles(MBR) of regions, RA is expressive enough to represent both direction and topological information.

RA is designed for reasoning about rectangular objects in 2-dimensional space whose sides are parallel to the axes of some orthogonal basis. However, when we consider a 2-D structure of such objects under the influence of gravity, we need to be able to represent information about the stability of the structure. Ideally, we want a representation that allows us to infer whether the structure will remain stable or whether some parts will move under the influence of the gravity or some other forces (e.g. the structure is hit by external objects). Additionally, if the structure is regarded as unstable, we want to be able to infer the consequences of the instability, i.e., what is the impact of movements of the unstable parts of the structure.

The Rectangle Algebra is not expressive enough to reason about the stability or consequences of instability of a structure. For example, in Fig. 1(a) and (b), assume the density of the objects is the same. The RA relation between object 1 and object 2 in these two figures are both (start inverse, meet inverse), but obviously the structure in figure 1(a) is stable whereas object 1 in (b) will fall. In order to make such distinctions, we need to extend the granularity of RA and introduce new relations that enable us to represent these differences. In this paper, we introduce an *extended Interval Algebra* (EIA) which contains 27 relations instead of the original 13. We use the new algebra as a basis for an *extended Rectangle Algebra* (ERA), which is obtained in the same way as the original RA. Depending on the needs of an application, we may not need to extend RA to 27 relations in each dimension. Sometimes we only need the extended relations in one axis. Thus, the extended RA may include 13×27, 27×13 or 27×27 relations depending on the requirement of different tasks.

We built an agent based on this model to participate the Angry Birds competition[1] which aims to play the Angry Birds game automatically and rationally. The result shows that the agent based on this model is able to interpret low-level infor-

[1] `http://ai2012.web.cse.unsw.edu.au/abc.html`

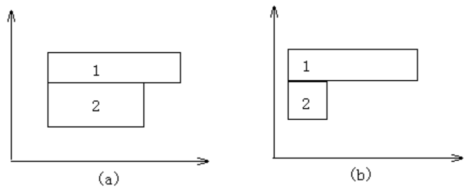

Figure 1: Two configurations with the same RA relation (si,mi)

mation from the scene or video input as higher level semantic descriptions [Fernyhough *et al.*, 1999]. Moreover, although qualitative spatial representation and reasoning has been applied to some simple physical systems to do some common sense reasoning [Klenk *et al.*, 2005], there is few work on reasoning on more complicated physical models; thus, this paper is an exploration of this area.

2 Interval Algebra and Rectangle Algebra

Allen's Interval Algebra defines a set $\mathcal{B}_{int}$ of 13 basic relations between two intervals (see Fig. 2). It is an illustrative model for temporal reasoning. Denote the set of all relations of IA as the power set $2^{\mathcal{B}_{int}}$ of the basic relation set $\mathcal{B}_{int}$. The composition ($\circ$) between basic relations in IA is illustrated in the transitivity table in Allen [1983]. The composition between relations in IA is defined as $R \circ S = \cup\{A \circ B : A \in R, B \in S\}$.

Relation	Illustration	Interpretation
X b Y Y a X	X Y	X takes place before Y
X m Y Y mi X	X Y	X meets Y(*i* stands for *inverse*)
X o Y Y oi X	X Y	X overlaps with Y
X s Y Y si X	X Y	X starts Y
X d Y Y di X	X Y	X during Y
X f Y Y fi X	X Y	X finishes Y
X = Y	X Y	X is equal to Y

Figure 2: The 13 basic relations of the Interval Algebra

RA is an extension of IA for reasoning about the 2-dimensional space. The basic objects in RA are rectangles whose sides are parallel to the axes of some orthogonal basis in 2-dimensional Euclidean space. The basic relations of RA can be denoted as $\mathcal{B}_{rec} = \{(A, B)|A, B \in \mathcal{B}_{int}\}$ The relations in RA are defined as the power set of $\mathcal{B}_{rec}$ The composition between basic RA relations is defined as $(A, B) \circ (C, D) = (A \circ C) \times (B \circ D)$.

3 The Extended Rectangle Algebra (ERA)

In order to express the stability of a structure and reason about the consequences of the instability in a situation which observes physical rules, we extend the basic relations of IA from 13 relations to 27 relations denoted as $\mathcal{B}_{eint}$ (see Fig. 3).

Relation	Illustration	Interpretation
X b Y Y a X	X Y	X takes place before Y
X m Y Y mi X	X Y	X meets Y (*i* stands for *inverse*)
X mom Y Y momi X	X Y	most part of X overlaps with most part of Y
X lol Y Y loli X	X Y	less part of X overlaps with less part of Y
X mol Y Y moli X	X Y	most part of X overlaps with less part of Y
X lom Y Y lomi X	X Y	less part of X overlaps with most part of Y
X ms Y Y msi X	X Y	X starts Y and cover most part of Y
X ls Y Y lsi X	X Y	X starts Y and cover less part of Y
X ld Y Y ldi X	X Y	X during left part of Y
X rd Y Y rdi X	X Y	X during right part of Y
X cd Y Y cdi X	X Y	X during Y and the midperpendicular of Y through X
X mf Y Y mfi X	X Y	X finishes Y and cover most part of Y
X lf Y Y lfi X	X Y	X finishes Y and cover less part of Y
X eq Y	X Y	X is equal to Y

Figure 3: 27 basic relations $\mathcal{B}_{eint}$ for extended IA

Definition 1 (The extended IA relations)**.** *We introduce the centre point of an interval as a new significant point in addition to the the start and end points. For an inteval a, denote centre point, start point and end point as c_a, s_a and e_a, respectively.*

1. The 'during' relation has been extended to 'left during', 'centre during' and 'right during' (ld, cd & rd).

- *"x ld y" or "y ldi x" :* $s_x > s_y, e_x \leq c_y$
- *"x cd y" or "y cdi :* $s_x > s_y, s_x < c_y, e_x > c_y, e_x < e_y$
- *"x rd y" or "y rdi x" :* $s_x \geq c_y, e_x < e_y$

2. The 'overlap' relation has been extended to 'most overlap most', 'most overlap less', 'less overlap most' and 'less overlap less'(mom, mol, lom &lol).

- *"x mom y" or "y momi x" :* $s_x < s_y, c_x \geq s_y, e_x \geq c_y, e_x < e_y$

- *"x mol y" or "y moli x" :* $s_x < s_y, c_x \geq s_y, e_x < c_y$
- *"x lom y" or "y lomi x" :* $c_x < s_y, e_x \geq c_y, e_x < e_y$
- *"x lol y" or "y loli x" :* $c_x < s_y, e_x > s_y, e_x < c_y$

3. The 'start' relation has been extended to 'most start' and 'less start' (ms & ls).

- *"x ms y" or "y msi x" :* $s_x = s_y, e_x \geq c_y$
- *"x ls y" or "y lsi x" :* $s_x = s_y, e_x > s_y, e_x < c_y$

4. Similarly, the 'finish' relation has been extended to 'most finish' and 'less finish' (mf & lf).

- *"x mf y" or "y mfi x" :* $s_x > s_y, s_x \leq c_y, e_x = e_y$
- *"x lf y" or "y lfi x" :* $s_x > c_y, s_x < e_y, e_x = e_y$

Denote the set of relations of extended IA as the power set $2^{\mathcal{B}_{eint}}$ of the basic relation set$\mathcal{B}_{eint}$. Denote the set of relations of extended RA as the power set $2^{\mathcal{B}_{erec}}$ of the basic relation set$\mathcal{B}_{erec}$.

Note that EIA can be expressed in terms of INDU relations [Pujari *et al.*, 2000] if we split each interval x into two intervals x_1 and x_2 that meet and have equal duration. However, this would make representation of spatial information very awkward and unintuitive. There is also some similarity with Ligozat's general intervals [Ligozat, 1991] where intervals are divided into zones. However, the zone division does not consider the half point.

4 Application of extended RA in Angry Birds

4.1 Rules based on the extended RA relations for analysing the structure

With these extended RA relations, it is possible to build a set of rules to determine some properties of a structure such as stability of a simple structure or consequences after some external influences act on the structure. Then, integrating all the proposed rules, we are able to do some further inferences to predict the consequences of a shot and calculate a a heuristic value. This value will suggest which object is a proper target to hit to maximize the damage. Assume the objects are only rectangles whose sides are parallel to the axes of some orthogonal basis.

Rule 1. Rules for determining stability

We will now specify rules that determine for each target object whether it is stable. Empirically, if we do not consider the impacts of the supportees of an object, there are three situations that an object will remain stable.

Rule1.1

The target object is just on the ground => object is stable

Rule1.2

For the target object x $\in O$(O is the set of all objects in the structure.), $\exists$ y, z$\in O$:

$R_{x,y} \in \{momi, moli, lomi, loli, msi, lsi, ldi\} \times (mi)$

$and R_{x,z} \in \{mom, mol, lom, lol, mfi, lfi, rdi\} \times \{mi\}$

=> x is stable

This rule describes the target object with supporters on both left and right sides stable.

Rule1.3

For the target object x, $\exists y$:

$R_{x,y} \in \{ms, mf, msi, ls, mfi, lf, cd, cdi, ld, rd, mom, momi, lomi, mol\} \times \{mi\}$

=> x is stable

This rule illustrates that if vertical projection of the mass centre of the target fall into the region of its supporter, it is stable.

Rule1.2 & 1.3 only consider the impacts of the supporters. However, sometimes the supportees may also influence the stability. Thus, we can add more rules to determine more complex situations.

Rule1.4

For the target object x, $\exists y$:

$R_{x,y} \in \{ld, cd, rd, ms, ls, mf, lf, eq\} \times \{mi\}$

or $\exists$ y, z:

$R_{x,y} \in \{momi, moli, lomi, loli, msi, lsi\} \times (mi)$

$R_{x,z} \in \{mom, mol, lom, lol, mfi, lfi\} \times \{mi\}$

=>x will remain stable no matter where its supportees are.

In this rule, the target object has at least one supporters on each side, and the edges of the supporters exceed the edges of the target object. Thus, no matter where the supportees are, they will not affect the stability of the target. Fig.4 illustrates *Rule1.2, 1.3 & 1.4*.

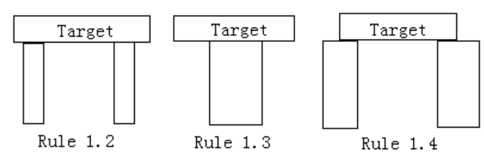

Figure 4: Illustration of Rule 1.2, 1.3 & 1.4

Rule1.5

$\forall y \in O$:

$R_{x,y} \notin \{ld, cd, rd, momi, moli, lomi, loli, ms, msi, ls, lsi, mf, lf, eq\} \times \{mi\}$

and $R_{x,y} \in \{ldi, cdi\} \times \{mi\}$

and $\exists z : R_{x,z} \in \{mom, mol, lom, lol, mfi, lfi, rdi\} \times (mi)$

and $\exists u \in O, R_{x,u} \in \{ldi, moli, lsi\} \times \{m\}$

=>x may be unstable.

This rule above can explain the configuration in fig. 5(a) which is that a supportee can make a stable object unstable.

Rule1.6

$\forall y \in O$:

$R_{x,y} \notin \{ld, ldi, cd, cdi, rd, momi, moli, lomi, loli, ms, msi, ls, lsi, mf, lf, eq\} \times \{mi\}$

and $\exists z : R_{x,z} \in \{mom, mol, lom, lol, mfi, lfi, rdi\} \times (mi)$

and $\exists u \in O, R_{x,u} \in \{mom, mol, lol\} \times \{m\}$

and $\exists v \in O, R_{u,v} \in \{ms, mf, msi, ls, mfi, lf, cd, cdi, ld, rd, mom, momi, lomi, mol\} \times \{mi\}$

=>x may be stable.

This rule explains that a supportee can force its support to be stable(example see fig. 5(b)).

In the above two rules, "may" is used to express the uncertainty of these situations, because in a qualitative way, we cannot always tell what will exactly happen.

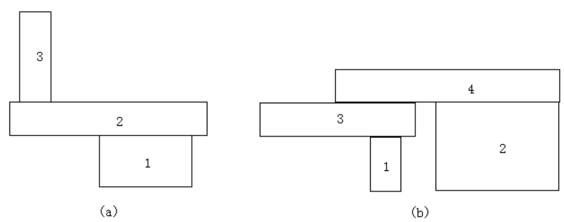

Figure 5: Configurations that need to consider the effects of supportees

Rule 2. Rules for determining reachability of the bird
In Angry Birds, we need to shoot a bird at the structure. When choosing the target, we need to consider which object can be reachable directly for the bird.
The rules for determining the reachability of the bird is shown below:

For a target object x $\in O$, R_x is the set of ERA relations between x and all other objects

$\forall R_{x,y} \in R_x, y \neq x :$
$R_{x,y} \in \{b, ldi, cdi, rdi, mom, mol, lom, lol, moli, momi, m, ms, ls, msi, lsi, mfi, lfi, eq\} \times \{A, A \in R_{eint}\}$
$\cup \{a, ld, cd, rd, lomi, loli, mi, mf, lf\} \times \{b, a, m, mi, mom, mol, lom, lol, momi, moli, lomi, loli, ldi, cdi, rdi, msi, lsi, mfi, lfi\}$
$=>$ The target x is directly reachable for a bird

This rule explains that if there is no other object blocks the path between the bird and the target object, the target is directly reachable by the bird.

Rule 3. Rules for detecting support and sheltering structures
The entire structure in Angry Birds game is often large and even in some levels all the objects in the world are constructed into only one structure. As can be found in most levels, many pigs are set on support structures sometimes with multi-level supporters. Then a good idea to kill the pig (if not directly reachable) is to destroy the support structure and the pig will probably die. Another useful substructure is the sheltering of the pigs. The reason is straightforward, if a pig is not reachable, there must be some objects that protect it; these objects form the sheltering structure of the pigs. Similarly, destroying the sheltering structures can either kill the pig or make the pig directly reachable to the bird.

Specifically, in order to separate the support structure of a pig from the larger structure, it is necessary to include the depth information of the supporters(see fig. 6 the illustration of support structure with depth). This is helpful when only considering the most essential supporters or only several layers of supporters are required. The rules for determining the direct supporter can be expressed using original RA relations:

Rule 3.1
For objects x,y $\in O$
$R_{x,y} \in \{d, di, o, oi, s, si, f, fi, eq\} \times \{m\}$
$=> y\ directly\ supports\ x$

This rule describes that if two objects vertically contact, the nether object supports the other one.

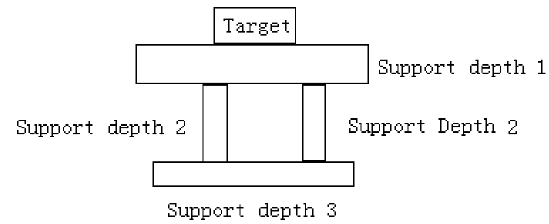

Figure 6: Illustration of support structure

Using the rule above, we can further get the supporters of the supporters, then we can collect all direct or indirect supporters of a certain object.

Similarly, a sheltering structure consists of the closest protection objects of the pig that can avoid the pig from a directly hit from each direction including the hit from backward. Specifically, a sheltering structure of a pig could consist of left, right and roof sheltering objects. In order to get the sheltering structure of a certain object (usually a pig), the first step is to get the closest object from the left side of the queried object; then, get the supportee list of the object (similar process as getting the supporter list); after that, get the right closest object with its supportee list. The next step is to check if the two supportee lists have objects in common, if so, pick the one with smallest depth as the roof object of the sheltering structure; if not, there is no sheltering structure for the queried object. If a roof object is found, also put the supportees of both the left and right closest objects with smaller depth than the roof object into the sheltering structure. Finally, put the supporters of both left and right closest objects which are not below the queried object into the sheltering structure.

The rules expressed in extended RA relations for determining sheltering objects consists of three parts (These set of rules can also be expressed in original RA):

Rule 3.2 The rules for getting potential left and right sheltering objects(take left side as an example)
For an object x $\in O$, denote S_l as the set of potential left sheltering objects of x.
$\forall y \in O,$
$R_{x,y} \in \{b, d, di, o, m, fi\} \times \{d, di, o, oi, s, si, f, fi, eq\}$
$=>$ put y into S_l

Rule 3.3 The rules for choosing closest sheltering objects
$\forall y, z \in S_l,$
$R_{y,z} \in \{b, d, o, m, s\} \times \{A, A \in R_{eint}\}$
$=>$ delete y from S_l, otherwise delete z

Finally, the closest objects will remain.

4.2 The integration of the rules to evaluate a shot

With the four rules described above, we are able to integrate the rules and further infer the possible consequences after a shot has been made. In order to predict the final consequence of an external influence on the structure, the direct consequence and its following subsequences should be analysed in detail. Funt suggested a similar method to simulate the consequence of a structure with a changed object which assumes that the changed object disappears and chooses the most significant unstable object to simulate the consequence [Funt, 1987]. In this case, a certain type of object can be affected by four configurations.

Configuration 1 The target object in the structure is hit directly by another object. The direct consequence will be in three types which are destroyed, falling and remaining stable. Empirically, the way to determine the consequence of the hit depends on the height and width ratio of the target. For example, if an object hits a target with the height and width ratio larger than a certain number (such as 2), the target will fall down. And this ratio can be changed to determine the conservative degree of the system. In other words, if the ratio is high, the system tend to be conservative because many hits will be determined as no influence on the target. Moreover, if the external object hits a target with the height and width ratio less than one, the target itself will remain stable temporarily because the system should also evaluate its supporter to determine the final status of the target. In some situations, we may also be concerned with the destruction of the target, such as in the Angry Birds game. After deciding the direct consequence of the hit, the system should be able to suggest further consequences of the status change of the direct target. Specifically, if the target is destroyed, only its supportees will be affected. If the target falls down, the configuration will be more complex because it may influence its supporters due to the friction, supportees and neighbours. If the target remains stable temporarily, it will also influence its supporters and its supporters may again affect it from the further simulation.

Configuration 2 The supportee of the target object falls down which is a less complex one. Similar to the process that set the height and width ratio to determine the stability of an object, this target object's stability is also represented by the ratio but the number should be larger (about 5) because the influence from supportee is much weaker than it from direct hit. If the target is considered as unstable, it will fall down and affect is neighbours and supporters; otherwise, it will only influence its supporters (see fig. 7).

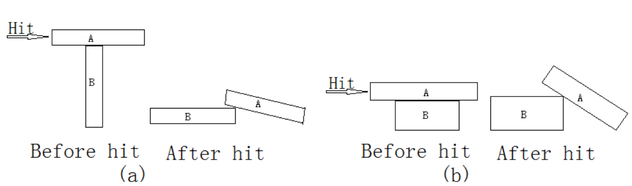

Figure 7: Configuration 2

Configuration 3 The supporter of the target object falls down. Here a simple structure stability check process (applying Rule 1) is necessary because after a supporter falls, the target may have some other supporters and if the projection of its mass centre falls into the areas of the other supporters, it also can stay stable. Then, if the target remains stable, it again will only affect its supporters due to the friction; otherwise, it may fall and affect its supporters, supportees and neighbours (see fig. 8(a)).

Configuration 4 The supporter of the target is destroyed. This is more like a sub configuration of the previous one. If the target can remain stable after its supporter destroyed, it may fall and affect its supporters, supportees and neighbours (see fig. 8(b)).

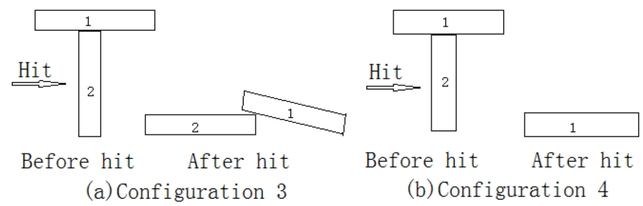

Figure 8: Configuration 3&4

4.3 Calculation of the heuristic value

Then, with all the affected objects in a list, the quality of the shot can be evaluated by calculating a total score of the affected objects. The scoring method is defined as: if an object belongs to the support structure or the sheltering structure of a pig, 1 point will be added to this shot; and if the affected is itself a pig, 10 points will be added to the shot. After assigning scores to shots at the objects, the target with highest score is expected to have the largest influence on the structures containing pigs when it is destroyed. Then, based on different strategies, the agent can choose either to hit the reachable object with highest heuristic value or generate a sequence of shot in order to hit the essential support object of the structure.

Algorithm 1 illustrates the whole process for evaluating a shot at all possible targets.

We first extract the ERA relations between all objects and then match the rules for all relevant combinations of objects. Thus the process of evaluating the significance of the targets is straightforward and fast.

5 Evaluation

We built an agent that uses the rules described in Section refApplication of extended RA in Angry Birds to perform a structural analysis of a given Angry Birds scenario and to determine which target to hit next. The organizers of the previous Angry Birds AI competition (`http://ai2012.web.cse.unsw.edu.au/abc.html`) provided a computer vision system that detects the mimimum bounding boxes (MBB) of all objects of an Angry Birds screen shot and a classification of each object (pig, bird, wooden block, ice block, etc). We take these boxes as input and evaluate

Algorithm 1 process of evaluating a shot

for all $Objects\ o\ in\ the\ structure$ **do**
  $init\ ongoing\ list\ 'ol'\ and\ affected\ list\ 'al'$
  $add\ o\ into\ al$
  $applying\ rule\ 1\ and\ 3\ (integrating\ in\ the\ 4$
  $configurations)\ to\ get\ affected\ objects\ 'ao'$
  $add\ all\ ao\ into\ ongoing\ list$
  for all $ongoing\ objects\ 'oo'\ in\ ol$ **do**
    $add\ oo\ into\ al\ and\ delete\ oo\ from\ ol$
    for all $objects\ ao'\ affected\ by\ oo'$ **do**
      if $ao' \notin al$ **then**
        $add\ ao\ into\ ol$
      end if
    end for
    if $ol\ = \emptyset$ **then**
      break
    end if
  end for
  $calculate\ heuristic\ value\ of\ o$
  $get\ stabilityofeachobject$
end for
output $a\ list\ of\ heuristic\ values\ for\ shots\ at\ all$
  $target\ objects\ in\ descending\ order\ with\ reachability$

each block according to our rules. For example, in the Angry Birds level shown in fig. 9, part of the output for evaluating the shot(see fig. 10) illustrates that the agent is able to infer that the essential supporter of the structure is object 19, and among the reachable objects, hiting object 6 can result in maximum damage to the structure.

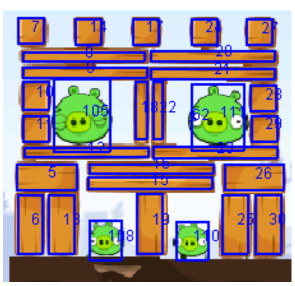

Figure 9: A sample level in Angry Birds

Our rules work well when the given MBBs closely resemble the actual blocks. When blocks are leaning to the left or right, our rules only provide a vague approximation of the real structural situation. Also, our rules treat each block equally, i.e., we do not distinguish between blocks of different materials that might have different mass or density, but purely focus on structural properties. Despite this, our agent performs quite well when comparing it to the winner of the last Angry Birds AI competition. We compared our agent with the winning agent on the publically available Poached Eggs levels (`chrome.angrybirds.com`). Our agent was able to achieve higher scores on average and to solve more levels than last years winner. Fig. 11 demonstrates the results of some sample levels from our agent

Target ID: 19.0 not reachable	**Target ID: 25.0 not reachable**
score: 56 From Left Affected objects: 15.0 Consequence: Fall affecter: 19.0 111.0 Consequence: Fall affecter: 23.0 ...	score: 46 From Right Affected objects: 15.0 Consequence: Fall affecter: 25.0 111.0 Consequence: Fall affecter: 23.0 ...
Target ID: 13.0 not reachable	**Target ID: 6.0 reachable**
score: 46 From Left Affected objects: 15.0 Consequence: Fall affecter: 13.0 111.0 Consequence: Fall affecter: 23.0 ...	score: 36 From Left Affected objects: 15.0 Consequence: Fall affecter: 13.0 111.0 Consequence: Fall affecter: 23.0 ...

Figure 10: Part of the output from for shot evaluation

and the winning agent. These levels shown in the figure are all constructed with complex structures, thus, in these levels, our agent performed much better than the winning agent. We compare our agent with the benchmarks given at `www.aibirds.org/benchmarks.html` for all participants of the 2012 competition. Our agent obtained a total score of 954960 over the first 21 poached eggs levels, which is higher than any other agent.

Agent Name	Agent Based on ERA	Winning Agent
Level	Score	Score
16	64060	39090
17	47960	40060
18	49870	27950
19	31130	15872
20	42410	57010

Figure 11: Results comparison

6 Discussion

In this paper we have introduced an extended rectangle algebra useful for representing and reasoning about stability and other properties of 2-dimensional structures. By splitting some basic interval relations into more detailed ones, we obtained 27 interval relations in each dimension that can express the physical relations between rectangular objects more precisely. We used the new algebra for defining some useful structural rules regarding properties such as stability, reachability, support, and shelter. We tested the usefulness of our rules by designing an agent that performs a structural analysis of Angry Birds levels. Based on these rules, we predict for each block the consequences if it gets hit and calculate a heuristic value that determines the usefulness to hit the block. We then shoot at the block with the highest value that is reachable with the current bird. A comparison with the winner of the last Angry Birds AI competition shows that our structural analysis can lead to a successful strategy for solving Angry Birds. It demonstrates the usefulness of qualitative spatial representation and reasoning approaches for solving real physical problems.

However the rules for reasoning about the consequences of a shot are still preliminary. The mechanical constraints for the motion of the objects, especially for the transfer of the motion between objects, need to be refined. Nielsen's approach[Nielsen, 1988] to analyse possible motions is suitable for our case. For example we could also consider translational motion and rotate motion instead of the simple 'fall'. Also, objects that are not equivalent to their MBRs, that is objects that can lean to the left or right may need to be differently treated. We will also consider different materials of objects in the next stage.

References

[Allen, 1983] J. F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26:832–843, 1983.

[Balbiani *et al.*, 1999] P. Balbiani, J.F. Condotta, and L.Farinas del Cerro. A new tractable subclass of the rectangle algebra. In *IJCAI-1999*, pages 442–447, 1999.

[Cohn and Renz, 2008] A. G. Cohn and J. Renz. Qualitative spatial representation and reasoning. In *Handbook of Knowledge Representation*, pages 551–584, Oxford, England, 2008. Elsevier.

[Fernyhough *et al.*, 1999] J. Fernyhough, A. G. Cohn, and D. C. Hogg. Constructing qualitative event models automatically from video input. *Image and Vision Computing*, 18:81–103, 1999.

[Funt, 1987] B.ian V. Funt. Problem-solving with diagrammatic representations. In *Readings in computer vision: issues, problems, principles, and paradigms*, pages 456–470. Morgan Kaufmann, 1987.

[Klenk *et al.*, 2005] M. Klenk, K. Forbus, E. Tomai, H. Kim, and B. Kyckelhahn. Solving everyday physical reasoning problems by analogy using sketches. In *AAAI-2005 Proceedings*, pages 209–215, 2005.

[Ligozat, 1991] G. Ligozat. Gerard ligozat: On generalized interval calculi. In *AAAI-1991*, pages 234–240, 1991.

[Liu *et al.*, 2009] W. Liu, S. Li, and J. Renz. Combining rcc-8 with qualitative direction calculi: Algorithms and complexity. In *IJCAI-2009*, pages 854–859, 2009.

[Liu, 1998] J. Liu. A method of spatial reasoning based on qualitative trigonometry. *Artificial Intelligence*, 98:137–168, 1998.

[Mukerjee and Joe, 1990] A. Mukerjee and G. Joe. A qualitative model for space. In *AAAI-1990 Proceedings*, pages 721–727. AAAI Press and MIT Press, 1990.

[Nielsen, 1988] Paul Nielsen. A qualitative approach to mechanical constraint. In *AAAI-1988 Proceedings*, pages 270–274, 1988.

[Pujari *et al.*, 2000] Arun K. Pujari, G Vijaya Kumari, and Abdul Sattar. Indu: An interval duration network. In *In Proceedings of Sixteenth Australian joint conference on AI*, pages 291–303. SpringerVerlag, 2000.

[Rajagopalan, 1994] R. Rajagopalan. A model for integrated qualitative spatial and dynamic reasoning about physical systems. In *Proceedings of the 12th National Conference on Artificial Intelligence (AAAI-94)*, pages 1411–1417, 1994.

[Randell *et al.*, 1992] Randell, Z. Cui, and T. Cohn. An interval logic for space based on connection. In *Proceedings of the 10th European Conference on Artificial Intelligence*, pages 394–398. Wiley, 1992.

[Renz and Ligozat, 2005] J. Renz and G. Ligozat. Weak composition for qualitative spatial and temporal reasoning. In *Principles and Practice of Constraint ProgrammingłCP 2005,11th International Conference*, pages 534–548, 2005.