2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Toward Autonomous Disassembling of Randomly Piled Objects with Minimal Perturbation

Oni Ornan[1] and Amir Degani[2]

Abstract— **Autonomous capabilities for manipulating randomly piled objects may enhance current methods of path planning and open a new field of development for mobile manipulation and Urban Search And Rescue (USAR) robotics. This paper introduces the challenge of achieving such manipulation capabilities and as a first step presents three algorithms, including a proposed novel solution, for the selection of objects to remove from a pile. The proposed algorithm determines a *removability rank* for each object according to the degree of its encapsulation within other objects. Using the contact vectors of the examined object, it is possible to obtain the motions that will not violate the object's unilateral contact constraints. The removability rank of the object is proportional to the union of all such motions. All algorithms were tested in simulation in full and partial knowledge modes, and evaluated on a physical robot with a simple manipulator and sensor. This work contributes: the introduction of an important autonomous manipulation challenge, the solution of which will be useful in the field of manipulation in general and USAR in particular; a specific novel algorithm for the construction of disassembly plans for piled objects; and an experimental evaluation of three algorithms targeted at such construction.**

I. Introduction and Motivation

This paper introduces the first steps toward **A**utonomous **D**isassembling of Randomly Piled **O**bjects with **M**inimal Perturbation (ADOM) by robotic assessment and manipulation (see Fig. 1). We chose to focus on the challenge of applying such a method to the field of Urban Search And Rescue (USAR) missions following catastrophic events such as earthquakes. In particular, three scenarios motivated the creation of this method and its application to USAR: 1) piled objects may encapsulate a survivor; 2) piled objects may block a path in a situation where the use of brute force such as shoveling might endanger the stability of the collapsed building; and 3) navigation through disordered environments may inadvertently cause further damage [1] if piled objects are not considered.

Dangerous environments for human rescuers, large areas, copious rubble, and human casualties have all pushed USAR teams to seek new aids, increasing the use of robots for USAR missions [1]. Such robots are currently controlled via teleoperation [2] but autonomous capabilities are developed and encouraged through contests such as RoboCup [3]. The need for increased autonomy in USAR manipulation tasks arises from three considerations: 1) poor communications in USAR environments due to physical interference [4] increase the risk that a robot fail and be abandoned, if the robot depends on teleoperation and lacks autonomous capabilities [5]; 2) increased autonomy may improve the human to robot ratio, reducing the number of operators at risk [4] and increasing the coverage area; and 3) the manual control of manipulation can be highly cumbersome, and autonomous manipulation would allow the operator to focus on other, possibly more important, tasks.

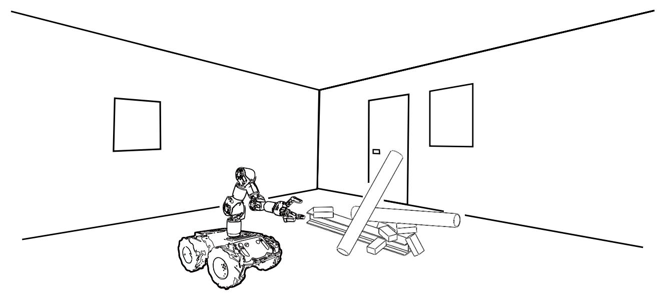


Fig. 1: Goal: a robot clears a path in a USAR mission autonomously or semi-autonomously.

Given the scenarios considered, it is clear why safety is the most important objective of an ADOM method.

The current development of USAR robots focuses on aspects such as locating survivors, giving initial medical stabilization, extricating trapped survivors and assessing the structural stability of the work environment. By incorporating manipulation, the method discussed below can potentially enhance the scope of structural stability assessment to allow for its use in the phases of locating and extricating survivors.

The paper presents and discusses the ADOM problem which requires relaxing two assumptions currently used by state-of-the-art work: well-structured piles and full knowledge of objects' properties and poses. To attempt solving the problem, the method investigated in this paper builds on related work that is presented in Section II. This work considers the disassembly of a pile as a series of separate iterations, each consisting of the selection and removal of a single object chosen by the algorithms presented in Section III. Section IV presents and discusses the set-up and results of simulating these algorithms in a full knowledge scenario and in a more realistic partial knowledge scenario. Section V portrays a set of proof-of-concept experiments in which the algorithms were applied to a set of real rectangular cuboids using a simple robotic arm and sensor. To conclude, various improvements and possibilities required and presented by the proposed algorithm are discussed in Section VI.

II. Background and Related Work

In recent years, there have been attempts to deal with mobile manipulation in urban environments, and some specifically in cluttered environments. However, there have been no direct attempts to solve the problem of disassembling

This work has been supported in part by the Technion's Center for Security Science and Technology. First author has been supported in part by the University of Toronto's Division of Engineering Science.

[1]O. Ornan is with the Faculty of Applied Science and Engineering, University of Toronto, Toronto, Ontario M5S2E4, Canada oni.ornan AT mail.utoronto.ca

[2]A. Degani is with the Faculty of Civil and Environmental Engineering, Technion Israel Institute of Technology, Haifa 32000, Israel adegani AT technion.ac.il

978-1-4673-6357-0/13/$31.00 ©2013 IEEE

randomly piled objects with **minimal perturbation** (i.e., minimal disturbance of the piled objects' poses). Here we present a short review and discussion of current mobile manipulation, perception and stability assessment approaches related to this challenge.

A. Mobile Manipulation and Path Planning

Today's state-of-the-art mobile manipulators deal with cluttered environments [6] and piled objects [7]. They are, however, focused on goals that permit pile collapse and disturbance. This focus renders irrelevant some of the challenges associated with the disassembling of piles while maintaining minimal perturbation. Classic path planning algorithms consider objects blocking a path as immovable, which causes failure on obstructed paths that a human could have passed through by clearing objects. Even more sophisticated mobile manipulation methods in the related domains of rearrangement of movable objects [8] and path planning among movable obstacles [9] consider path blockages to consist of single movable pieces only. Piled objects cannot be handled by such algorithms, which thus may be enhanced by the method presented here.

B. Perception

Occlusions are known to cause perception problems for autonomous robots. Currently, such situations may be addressed by the physical removal of occluders, for example via operator intervention [10]. In the context of this work, however, increased autonomy is a declared objective and occlusions are inherent and cannot be removed. The inherent nature of the occlusions prevents the use of methods such as object identification following tabletop segmentation [11]. Other methods such as object classification based on feature identification [12] are capable of handling some occlusions but cannot be used in many USAR environments, since in this context objects are often damaged and dirty (e.g., covered with dust). The most recent related development is the singulation of piled objects by means of manipulation [7] such as grasp attempts or perturbation pushes. However, since the pile's stability itself is in question, manipulating objects prior to a stability assessment is not desired.

C. Manipulation and Stability Assessment

The closest problem to ADOM currently researched in the field of robotics is that of Jenga playing robots [13]–[15], which also need to handle the safe disassembling of multi-object structures. However, these algorithms for selecting an object to remove are either random [15] or rely on highly organized and fully known structures. One of the algorithms discussed below (Dynamics-Based Algorithm) is based on the "Removal Feasibility Planner" phase undertaken by such a Jenga playing robot [13]. However, this algorithm fails under more realistic, partial-knowledge conditions, as demonstrated in the following analysis of algorithms. The work by Katz, et. al., [16], [17], which relates to our current work, was published following the submission and review of this paper. Although less attention was given to pile safety, the work complements our own in many ways.

III. Algorithms

Three novel ADOM object selection algorithms are presented: Centroid-Height Algorithm (CHA), a simple algorithm for comparison only; Dynamics-Based Algorithm (DBA), potentially useful but with some inherent shortcomings as shown later; and Kinematics-Based Algorithm (KBA), a proposed method for ADOM. Each algorithm considers the disassembly of a pile as an iterative process of separate manipulation tasks (removals), according to the general structure for pile interaction algorithms [7]. Before each removal, the poses, properties and contact normals of all objects are obtained and used as an input for an object selection algorithm. Each of the three algorithms serves as an object selector, analyzing different components of the given data and returning the object deemed the safest to remove. Following removal the process repeats itself until all objects are cleared or, for the last algorithm only, until no safe removal option is found.

Jenga playing robots handle problems of minimal perturbation in certain environments that entail two major assumptions: 1) objects are stacked in a well-structured manner; and 2) full knowledge of the objects is available. Eliminating the first assumption to extend the minimal perturbation problem to randomly piled objects, along with relaxing the second assumption (i.e., providing only partial knowledge of the objects), are important, necessary, and logical next steps toward manipulation capabilities that will be applicable to real cluttered and unstructured environments. To allow a meaningful investigation of the implications of relaxing these major assumptions, some other assumptions akin to those of Jenga robots were kept, such as decomposition of the problem into perception and manipulation phases, obtainability of certain data through perception, and graspability of all objects in a pile.

Specifically, for any object, each of CHA, DBA, and KBA assumes the obtainability of centroid's (center of volume) position; pose and physical properties (e.g., density); and contact normals, respectively. Any object is also assumed to be graspable without disturbing other objects. A limit is imposed to allow only one object to be manipulated at any given time.

A. Centroid-Height Algorithm (CHA)

A basic intuitive observation might imply that if we need to disassemble piled objects, we should likely start from the top. CHA is built around this intuition and simply removes the object with the highest geometric center. However, the simplicity of this algorithm comes at a price. It may fail in common scenarios when the object with the highest geometric center supports another object with a lower geometric center, as seen in Fig. 2.

B. Dynamics-Based Algorithm (DBA)

Fig. 3 depicts an overview of DBA (Algorithm 1), which consists of the following phases: 1) using the poses and physical properties of the objects given as input, the pile is simulated in a rigid body dynamics simulator; 2) an object (the *Candidate*) is removed from the simulated pile; 3) the

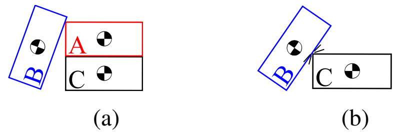


Fig. 2: Problematic choice of CHA. The geometric center of Object A is higher than the geometric center of Object B. CHA will select and remove Object A, causing Object B to fall.

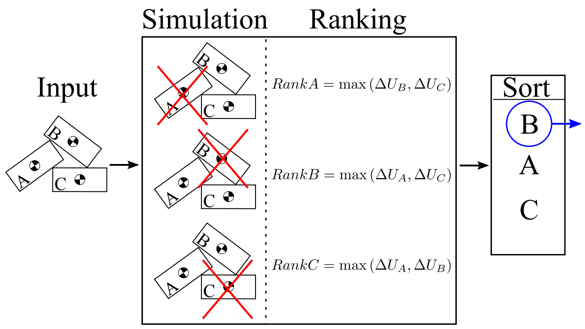


Fig. 3: An overview of DBA.

new poses of the remaining objects are recorded; 4) the change of potential energy (ΔU) is calculated for each of the remaining objects and the removability rank of the Candidate is equal to the maximum individual change of potential energy; 5) the process is repeated for each piled object, which is used as the Candidate in its turn; and 6) the Candidate with the lowest rank is returned as the safest removal option.

Algorithm 1 Dynamics-Based Algorithm (DBA)

Input: Objects' poses and physical properties
Output: The object predicted to result in the smallest individual ΔU

sort objects **by increasing DynamicRank(**$objects[i]$,$objects$**)**
return $objects[0]$

DynamicRank($candidate$, $objects$**)**

AddToSimulatedWorld($objects$**)**
RemoveFromSimulatedWorld($candidate$**)**
sort $objects$ **by decreasing** $\Delta U(objects[i])$
return $\Delta U(objects[0])$

DBA alleviates the deficiencies of CHA by considering not only the state of the Candidate but also the effect of this Candidate's removal on other objects. Given proper input data, DBA will accurately return the safest removal option. However, DBA requires full knowledge of the pose, shape, size and other physical properties of every object in the pile. This data is inherently problematic to obtain for three dimensional piles, as some objects may be occluded. Additionally, materials may not be known.

C. Kinematics-Based Algorithm (KBA)

1) Background – In order to find the objects that are least encapsulated and thus safest to remove: 1) the contact normals can be viewed as a set of constraining screws; 2) we can ask if there are any instantaneous rigid motions (twists) that do not violate the constraints [18]; and 3) the removability rank of a Candidate can be defined as proportional to the integral of such motions. Instead of using general screws, we chose a more restrictive approach that considers infinite pitch screws, which correlate to translation-only motions.

2) Implementation – As seen in Fig. 4a, KBA (Algorithm 2) receives the contact normals of each Candidate as an input. The contact normals are oriented to point away from the Candidate's centroid and are imposed on the point of origin of a unit sphere. As a result, each contact normal now defines a plane through the origin of the sphere, as seen in Fig. 4b. As illustrated in Fig. 4c, we obtain the intersection points of the great circles, which themselves are created by intersecting the contact normals' planes with the unit sphere.

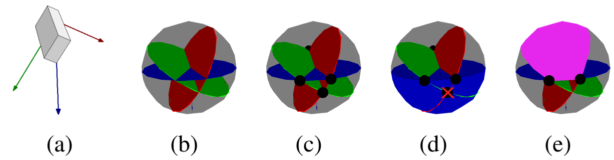


Fig. 4: KBA process. The contact normals given as an input (a) define planes (b), whose intersections points are found (c). The intersection points with a positive projection on any of the contact normals are removed (d), and the remaining points create a spherical polygon whose surface area (e) is taken as the object's rank.

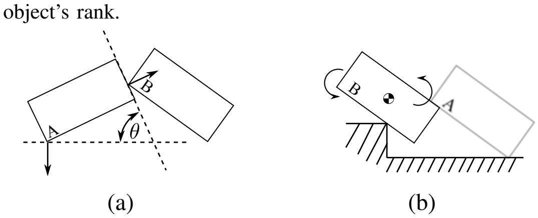


Fig. 5: Special cases for KBA. In (a) the contact normals of Object A create a spherical polygon with a non-zero surface area, seen as the lune with angle θ. In (b) Object B would rotate without the support of Object A, demonstrating KBA's disadvantage in lacking consideration of either rotations, or the effects of a Candidate's removal on other objects.

Any movement direction that will violate the contact normal constraints (i.e., result in pushing against an object in contact with the Candidate) is a member of at least one half sphere above the plane defined by one of the contact normals. The union of a Candidate's allowed movement directions can be modeled as a unit sphere minus all such half spheres. Equivalently, it can be modeled as a spherical polygon (including a spherical triangle or a lune), whose vertices are the great circles' intersection points, excluding the points that have a positive projection on any of the contact normals (i.e., that are members of any half sphere defined by a plane), shown in Fig. 4d.

The removability rank of a Candidate is defined as the surface area of its spherical polygon, shown in Fig. 4e. The rank is in the range of $[0, 2\pi]$, where a rank of 0 indicates an immovable object and a rank of 2π indicates an object on a plane.

A special consideration must be made for cases where an object is supported from below by a Candidate whose spherical polygon's surface area is non-zero, as seen in Fig. 5a. To handle this scenario, any Candidate that has a contact normal with a negative projection on the gravity vector is declared immovable (rank=0) regardless of its spherical polygon's surface area.

KBA only predicts an object's removability based on its state. As such, one of its main drawbacks is its lack of consideration for the removal's effect on other objects, as exemplified in Fig. 5b. However, KBA prediction does not require full knowledge of all piled objects and can thus be useful for occluded scenes with objects of unknown physical properties and material compositions.

Algorithm 2 Kinematics-Based Algorithm (KBA)

Input: Unit vectors of contact normals of each object, pointing outwards. Duplicates removed.

Output: The least encapsulated object or nothing if no safe option found.

sort $objects$ **by decreasing KinematicRank**($objects[i].contactNormals$)

if $objects[0].rank > 0$ **then return** $objects[0]$

KinematicRank($contactNormals$)

for $normal_1$ **in set** $contactNormals$ **do**
  for $normal_2$ **in set** $contactNormals$ **do**
    if $||proj_{gravity} normal_2|| < 0$ **then return** 0
    else if $normal_1 != normal_2$ **then**
      put IntersectionPoints($plane_1$, $plane_2$, $unitSphere$) **in set** $interPoints$
for $point$ **in set** $interPoints$ **do**
  for $normal$ **in set** $contactNormals$ **do**
    if $||proj_{normal} point|| > 0$ **then**
      remove $point$ **from set** $interPoints$
return PolygonSurfaceArea($interPoints$)

IV. Simulated Experiments

A. Set-up

The work in both simulated and physical experiments (Section V) was done using the Robotic Operating System (ROS) [19]. The simulation was performed using ROS's Gazebo simulator with the underlying Open Dynamics Engine (ODE) [20] rigid body dynamics simulator. A list of 500 piles was randomly generated, keeping the number of objects per pile ($y \in [5,25]$) uniformly distributed. The rectangular cuboids varied in dimensions ($volume \in [0.01, 0.09]\mathrm{m}^3$) while their density was kept constant at the value of concrete ($2300\frac{\mathrm{kg}}{\mathrm{m}^3}$). The rectangular cuboids were randomly distributed in a defined space, which increased the chances of a well-packed pile. To maintain the consistency of testing conditions we ran the three algorithms on the entire list of piles, allowing not more than a single removal from each pile. For each pile, the poses of all rectangular cuboids were recorded before and after the removal.

B. Knowledge Modes

Robots are required to act in real world environments with varying levels of obtainable data. Decision making becomes harder and more interesting as the knowledge of the work environment is reduced. To provide an in-depth comparison of the three algorithms, they were simulated in both an ideal full knowledge scenario and in a more realistic partial knowledge scenario.

1) Full Knowledge Mode –The three algorithms were allowed to run with all available data as input. When using the Gazebo simulator, this means the algorithms had access to the full physical properties, states and contact normals of all objects in a given pile.

2) Partial Knowledge Mode –The simulation ran exactly the same as in the full knowledge mode, but with reduced input data. A single point of view was selected, imitating a robot's sensor, and only data about objects assumed to be observable by such a set-up were considered. Though slightly artificial, observability was measured by the number of observable vertices. Five vertices guarantee knowledge of three dimensions for a rectangular cuboid, and were therefore chosen as the minimum requirement to define an object as observable. To imitate the impact of a partial knowledge based decision on the real world, in each instance the chosen rectangular cuboid was removed from a pile that was simulated with full knowledge.

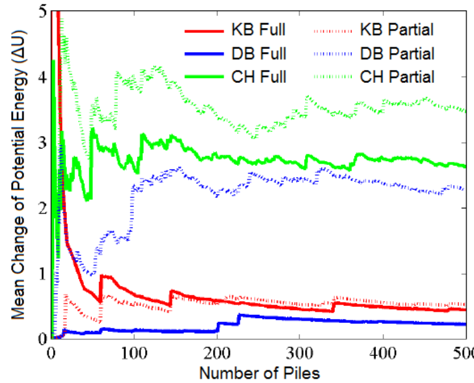


Fig. 6: Mean change of potential energy (ΔU) of the three algorithms in full and partial knowledge modes, plotted against the number of piles tested.

TABLE I: Simulation Comparison of the Three Algorithms.

Algorithm	CHA		DBA		KBA	
Knowledge	Full	Partial	Full	Partial	Full	Partial
Mean Δt(s)	5.5e-5	3.3e-4	9.8e1	3.3e1	2.5e-2	1.6e-2
Std. Dev,	1.4e-4	2.3e-4	1.0e2	1.6e1	1.3e-2	9.4e-3
Mean ΔU(J)	2.7e0	3.6e0	2.5e-1	2.4e0	4.9e-1	5.8e-1
Std. Dev.	19	26	5.5	18	7.3	6.7
% Increase	N/A	31.9%	N/A	880%	N/A	16.6%
Sum ΔU	1.9e4	2.6e4	1.7e3	1.7e4	3.5e3	4.1e3
Mean ΔS(m)	6.5e-3	8.4e-3	7.2e-4	5.8e-3	1.4e-3	1.6e-3
Std. Dev.	3.4e-2	4.5e-2	1.2e-2	3.6e-2	1.6e-2	1.6e-2
% Increase	N/A	27.8%	N/A	701%	N/A	12.0%
Sum ΔS	46	59	5.1	41	9.8	11.0

C. Results

Fig. 6 demonstrates how the relative performance (i.e., how much better one algorithm-mode pair performs compared to another) stabilizes with an increasing number of test piles, to give the results of all 500 piles (shown in Fig. 7 and Table I). The histograms in Fig. 7a and Fig. 7b show the magnitude of individual rectangular cuboids' movements when applying the algorithms in full and partial knowledge modes respectively. Table I constitutes a record of the algorithms' runtimes and two measurements, which compare the states of objects before and after the chosen Candidate's removal: change in potential energy ΔU and change in position ΔS (Euclidean distance). The table's "Percent Increase" rows measure each algorithm's decrease in performance when downgraded from full to partial knowledge mode, measured by the percent increase of the mean ΔU and ΔS respectively.

D. Discussion

First, it should be noted that the relative performance of the three algorithms across the two modes is consistent when measured by ΔU and by ΔS, though DBA is solely based on the first measurement, whereas the two other algorithms do not take the mass of objects into account at all. Second, it should be noted that KBA identified a safe removal option for all piles tested.

As expected, CHA's performance is the poorest in both

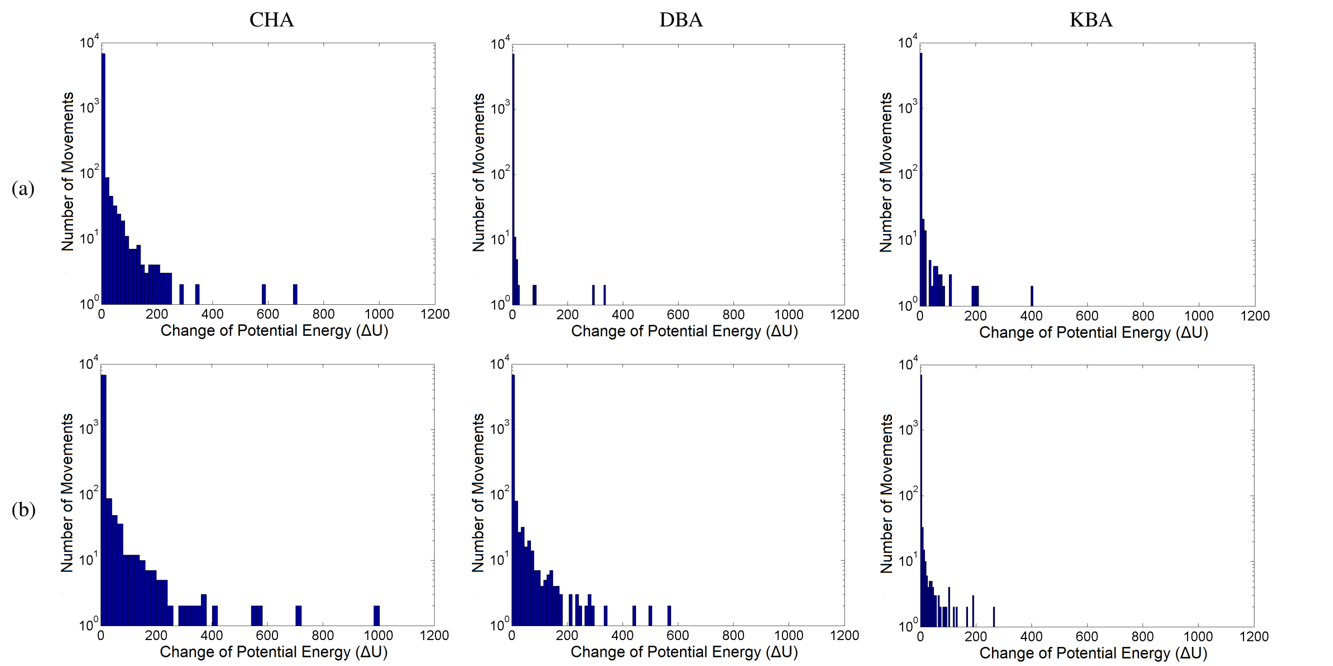


Fig. 7: Individual potential energy changes (ΔU), following the removal of a single rectangular cuboid, in full (a) and partial (b) knowledge modes. The wider histogram distribution to the right correlates to more individual changes in potential energy with large magnitudes, indicating a weaker performance.

full and partial knowledge modes, due to the limited pile complexity it is capable of handling. Its performance decreases in partial knowledge mode, most likely because the top object is invisible in some configurations.

The superior performance of DBA in full knowledge mode is inherent and unmatchable since it has all the data required to accurately simulate reality with no errors beyond those of the rigid body dynamics simulator. Notably, with the transition to partial knowledge mode, the performance of DBA becomes almost as weak as that of CHA. This is a consequence of the invisibility of supporting objects in some piles, as exemplified in Fig. 8a. The initial misrepresentation of simulated piles is inherent in DBA running on partial knowledge, and causes the selection process to be nearly random, with some exceptions exemplified in Fig. 8b.

Though KBA's performance also decreases with the transition to partial knowledge mode, it is only a minor reduction. Its performance is the best of all algorithms in this mode. The smaller reduction in the performance of KBA compared to DBA is due to the fact that computation depends on kinematics only, independent of dynamics. In other words, since the visible objects' initial states are not forced to deviate from the conditions of reality during the KBA process, the computation is performed on reduced but true data.

Compared to the two other algorithms, CHA is faster, but its weak performance prevents its use for any complex piles. Comparing between DBA and KBA, we have demonstrated that KBA provides a better combination of lower dependency on observability and significantly shorter runtime.

V. Physical Demonstrations

A. Experimental Set-up

As seen in Fig. 9, a Kinect depth-camera (Microsoft Corporation), and a simple Turtlebot-Arm (Willow Garage)

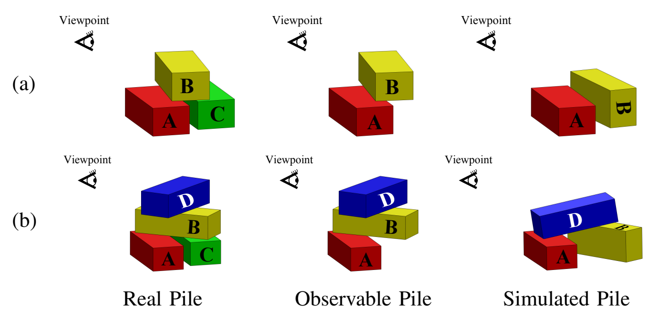


Fig. 8: (a) The supporting Object C is invisible, causing the pile to be misrepresented in simulation and DBA's result to be random. (b) Though the pile is misrepresented in simulation, the hierarchy is kept true, allowing for the proper selection of a removal object by DBA.

manipulator with four degrees of freedom that uses a simple gripper, were used for the experimental set-up. To ensure a constant transformation between the sensor and manipulator, both were affixed to a tabletop. Rectangular cuboids made of Lego were used as piled objects, as seen in Fig. 10a. Each rectangular cuboid was kept hollow with external dimensions of 31.8 mm $\times$ 88.0 mm $\times$ 24.1 mm.

B. Perception

Perception complexity was decreased for the physical experiments by relaxing the constraints associated with objects in USAR environments being damaged and dirty. A simplified version of object classification based on feature identification was implemented in the form of color coding, and each rectangular cuboid was marked on one of its sides with a uniquely colored rectangle. Following color filtering, Singular Value Decomposition [21] was employed on each colored rectangle to obtain the rectangular cuboid's pose. The collected poses, together with hard-coded data of the objects' dimensions (for all algorithms) and physical properties (for DBA only), were used as the algorithms' input. Naturally

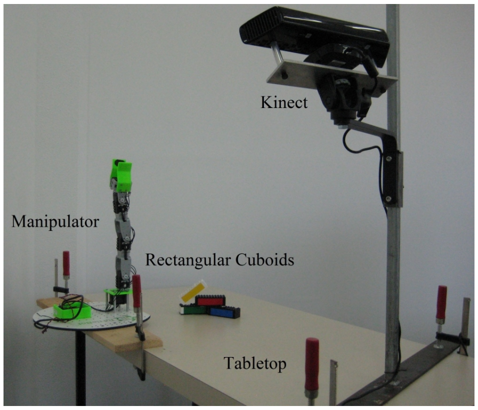
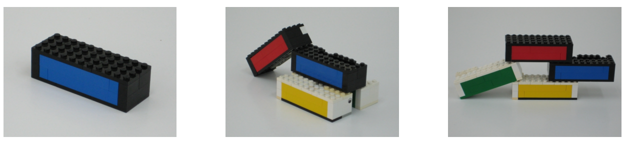


Fig. 9: Manipulator and sensor set-up.



(a) (b) (c)

Fig. 10: Object type (a) and test configurations (b) (c) used in experiments.

occurring perception uncertainties (specifically, inaccuracies in perceived poses) existed for all algorithms. Artificial perception uncertainties were further introduced to KBA by altering objects' dimensions by a factor greater than 1, thus increasing chances of objects' penetration.

C. *Proof-of-Concept Experiments*

Fig. 11 demonstrates the results of disassembling the pile configuration seen in Fig. 10b using CHA, DBA, and KBA respectively. The experiment demonstrates CHA's failure to handle piles which are not of the simplest configuration, as explained in Fig. 2. It also illustrates DBA's shortfall in a partial knowledge scenario as explained in Fig. 8a. Finally, the expected good performance of KBA in a partial knowledge scenario is demonstrated. Fig. 12 demonstrates the results of applying KBA to the pile configuration seen in Fig. 10c which, as described in Fig. 5b, is currently an unhandled case. The results and therefore conclusions of the experiments are similar to those of the simulated experiments described in Section IV.

VI. Discussion

KBA's current implementation considers translational motions only. Enhancing its capabilities to consider rotations will likely provide some interesting results. As explained, KBA only predicts the removability of an object based on its state, without considering the removal's effect on other objects. Thus, with the exclusion of rotations, a negative result (i.e., no safe disassembly plan identified) can be trusted, but a positive result cannot be fully trusted. While including considerations for rotations may eliminate or reduce this problem, it is very possible that the algorithm may still be insufficient as a standalone method and may require a complementary step to further analyze the removal selection it provides. Such a step may include dynamic consideration during the object selection phase (thereby integrating a variation of DBA) and/or during the physical removal phase with an abort-and-retry approach based on existing methods (e.g., expected force threshold [14]).

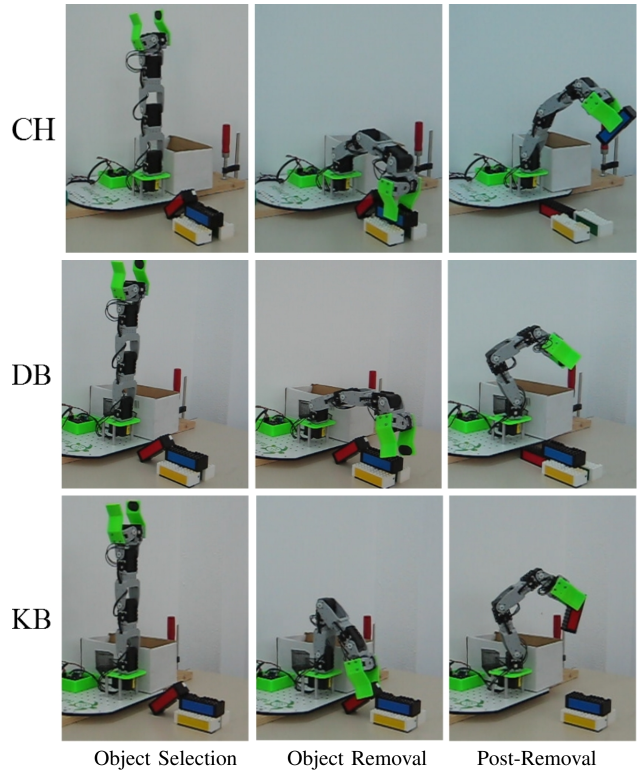


Fig. 11: Proof-of-concept demonstrations of the application of the three algorithms to the pile configuration seen in Fig. 10b. CHA fails by picking up the blue rectangular cuboid, which supports another rectangular cuboid. DBA fails by selecting the obstructed yellow rectangular cuboid, resulting in a grasp failure. KBA, however, correctly chooses the red rectangular cuboid and therefore maintains the stability of the pile. Note that the pictures are not taken from viewpoint of the Kinect, which only sees the red, yellow, and blue rectangular cuboids.

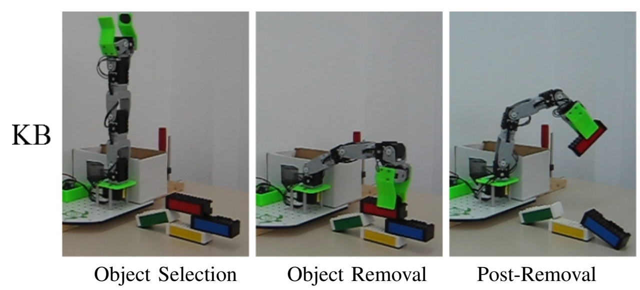


Fig. 12: Proof-of-concept demonstrations of the application of KBA to the pile configuration seen in Fig. 10c. The algorithm picks the best object but still fails, since the selected object supported another object, a critical factor that was not foreseen by the algorithm.

While relaxing Jenga robots' major assumptions, all algorithms still assume the obtainability of certain data through perception. As Sections IV and V discuss, however, for ADOM evaluation it is possible to discount CHA and DBA based on the inability to handle complex piles and partial data resulting from occlusions, respectively. Since these inabilities are independent of assumptions made regarding the obtainability of objects' centroid position, and the physical properties of objects, the question of whether these data are truly obtainable in USAR environment becomes irrelevant. In contrast, KBA assumes the obtainability of contact normals, which, using Open Dynamics Engine, were indeed suc-

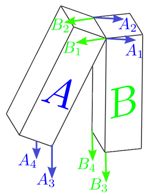


Fig. 13: By artificially providing a support at the lines of contact normals B1 and B2, it may be possible to remove Object B without Object A falling.

cessfully obtained for various object interactions, including cases with some perception uncertainties in the physical experiments described in Section V. Further investigation as to the obtainability of contact normals with increased perception complexity, and implications on KBA's tolerance of perception uncertainties and errors, should be performed.

KBA assumed any object to be graspable without disturbing other objects. It can be shown that the less encapsulated an object is within other objects (i.e., the higher its Kinematic rank), the more grasping poses and obstacle-free paths to such poses a manipulator is likely to have. Specific goals for the use of ADOM might require clearing only a subset of the piled objects rather than the entire pile (e.g., to make an opening). Consideration of specific goals and specific manipulators' grasping constraints should be incorporated to any algorithm (being KBA or another) proven to satisfy the more fundamental requirements of ADOM.

In order for the method to serve as a base for a wide range of robots, including single handed ones, KBA assumes a limitation of manipulating at most a single object at a time. However, as exemplified in Fig. 13, through KBA's obtained contact normals it may also be possible to find the support needed to compensate for the removal of one object currently supporting another. This may be done either by using the second hand of a dual-arm robot or by removing object B while pushing object A into a statically stable pose.

The low accuracy of predicting the effect of manipulating piled objects given partial knowledge makes it difficult to plan for more than a single manipulation task. This is a prime reason for viewing pile disassembling as an iterative process of separate removals. However, as ADOM algorithms themselves are designed to make this exact prediction, improvement of their accuracy may allow for longer term plans.

Already in its current development KBA obtains all possible removal directions for a given object. As a side benefit, the safest direction might be obtainable from these data (e.g., as the centroid of the spherical polygon).

VII. Conclusions and Future Work

The challenge of **A**utonomous **D**isassembling of Randomly Piled **O**bjects with **M**inimal Perturbation (ADOM) was introduced and discussed. Three algorithms for the selection of objects to remove were presented and compared, and the proposed novel Kinematics-Based Algorithm (KBA), which is based on the contact normals between piled objects, was shown to be superior. As a first step toward ADOM, KBA presents some challenges and possible side benefits.

The examples investigated thus far have only dealt with rectangular cuboids due to technical reasons unrelated to the properties of any of the algorithms themselves. Our current work focuses on the relaxation of this limitation along with further investigation of perception requirements, to create an integrated system which should be applicable to other primitive or even complex shapes.

VIII. Acknowledgments

The authors would like to thank Sharon Temtsin for helping with the experiments.

References

[1] D. Stormont, *Urban Search and Rescue: A challenge for Autonomous Robots with Application to Planetary Exploration*, 2002.

[2] J. Bae, *Development and Evaluation of Wearable Responder Interactive System for Teleoperation (WRIST)*, 2008.

[3] (2012) Robot league. [Online]. Available: http://wiki.robocup.org/wiki/Robot_League

[4] R. Murphy, "Human-robot interaction in rescue robotics," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 34, no. 2, pp. 138 –153, May 2004.

[5] R. Murphy, J. Blitch, and J. Casper, "AAAI/robocup-2001 urban search and rescue events," *AI Magazine*, vol. 23, no. 1, p. 37, 2002.

[6] S. Srinivasa, D. Ferguson, C. Helfrich, D. Berenson, A. Collet, R. Diankov, G. Gallagher, G. Hollinger, J. Kuffner, and M. Weghe, "Herb: a home exploring robotic butler," *Autonomous Robots*, vol. 28, no. 1, pp. 5–20, 2010.

[7] L. Chang, J. Smith, and D. Fox, "Interactive singulation of objects from a pile," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, May 2012, pp. 3875 –3882.

[8] O. Ben-Shahar and E. Rivlin, "To push or not to push: On the rearrangement of movable objects by a mobile robot," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 28, no. 5, pp. 667–679, 1998.

[9] M. Stilman and J. Kuffner, "Planning among movable obstacles with artificial constraints," *The International Journal of Robotics Research*, vol. 27, no. 11-12, pp. 1295–1307, 2008.

[10] M. Gianni, P. Papadakis, F. Pirri, and M. Pizzoli, "Awareness in mixed initiative planning," in *2011 AAAI Fall Symposium Series*, 2011.

[11] J. Stuckler, D. Holz, and S. Behnke, "Robocup@ home: Demonstrating everyday manipulation skills in robocup@ home," *Robotics & Automation Magazine, IEEE*, vol. 19, no. 2, pp. 34–42, 2012.

[12] A. Collet, D. Berenson, S. Srinivasa, and D. Ferguson, "Object recognition and full pose registration from a single image for robotic manipulation," in *Robotics and Automation (ICRA), 2009 IEEE International Conference on*, 2009, pp. 48–55.

[13] J. Wang, P. Rogers, L. Parker, D. Brooks, and M. Stilman, "Robot jenga: Autonomous and strategic block extraction," in *Intelligent Robots and Systems (IROS), 2009 IEEE International Conference on*, 2009, pp. 5248–5253.

[14] S. Kimura, T. Watanabe, and Y. Aiyama, "Force based manipulation of jenga blocks," in *Intelligent Robots and Systems (IROS), 2010 IEEE International Conference on*, 2010, pp. 4287–4292.

[15] T. Kroger, B. Finkemeyer, S. Winkelbach, L. Eble, S. Molkenstruck, and F. Wahl, "A manipulator plays jenga," *Robotics & Automation Magazine, IEEE*, vol. 15, no. 3, pp. 79–84, 2008.

[16] D. Katz, M. Kazemi, J. A. Bagnell, and A. Stentz, "Clearing a pile of unknown objects using interactive perception," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, 2013.

[17] D. Katz, A. Venkatraman, M. Kazemi, D. Bagnell, and A. Stentz, "Perceiving, learning, and exploiting object affordances for autonomous pile manipulation," in *Proceedings of Robotics: Science and Systems*, Berlin, Germany, June 2013.

[18] M. Richard, Z. Li, and S. Sastry, "A mathematical introduction to robotic manipulation," 1994.

[19] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Ng, "ROS: an open-source robot operating system," in *ICRA Workshop on Open Source Software*, vol. 3, no. 3.2, 2009.

[20] R. Smith. (2005) Open dynamics engine. [Online]. Available: www.ode.org

[21] M. Sonka, V. Hlavac, R. Boyle, *et al.*, *Image processing, analysis, and machine vision*. PWS publishing Pacific Grove, CA, 1999, vol. 2.