

Performance Metrics of Speed and Separation Monitoring in Shared Workspaces

Jeremy A. Marvel, *Member, IEEE*

Abstract—A set of metrics is proposed that evaluates speed and separation monitoring efficacy in industrial robot environments in terms of the quantification of safety and the effects on productivity. The collision potential is represented by separation metrics and sensor uncertainty based on perceived noise and bounding region radii. In the event of a bounding region collision between a robot and an obstacle during algorithm evaluation, the severity of the separation failure is reported as a percentage of volume penetration.

Note to Practitioners—This work draws upon the necessity of testing and evaluating industrial robotic safety systems. With a plethora of existing collision avoidance methods (and many more currently in the pipeline), there is a clear and present need for an assessment basis to gauge speed and separation efficacy. This paper presents a comparative set of metrics based on separation distance, sensor uncertainty, and robot velocity and mass. Tools are provided for system performance and failure assessment. Applying the metrics directly, intelligent robot systems can recognize unsafe operating conditions and respond accordingly. Postprocess analysis of logs and calculated values identifies the conditions and circumstances for which the safety of the system is at a minimum.

Index Terms—Robot safety, collision avoidance, speed and separation monitoring.

I. INTRODUCTION

WITHIN the realm of industrial collaborative robotics, the safety of a robot is categorically assessed based on two dichotomous practices: power and force limiting (PFL), which minimizes the potential for operator injury upon robot impact based on established injury standards [1]–[3]; and speed and separation monitoring (SSM), or speed and position monitoring (SPM), which aims to prevent impact from occurring altogether [1], [2]. The goal of SSM is to maintain a static safe separation distance between the robot and any humans walking through the collaborative workspace. This distance is enforced by dynamically monitoring the travel distance required to bring the robot to a safe, controlled stop. When the separation distance between the human and the robot is less than or equal to this travel distance plus the safe Euclidean buffer distance, a safety-monitored stop of the robot is initiated to halt the motion of the robot before it can impact the human.

Manuscript received September 23, 2011; revised May 24, 2012; accepted September 30, 2012. Date of publication February 04, 2013; date of current version April 03, 2013. This paper was recommended for publication by Associate Editor W. Sheng and Editor K. Lynch upon evaluation of the reviewers' comments.

The author is with the Intelligent Systems Division, National Institute of Standards and Technology, Gaithersburg, MD 20899 USA (e-mail: jeremy.marvel@nist.gov).

Digital Object Identifier 10.1109/TASE.2013.2237904

Although the separation question is directly traceable to the classical problem of robot collision avoidance, there is no unifying process that quantitatively scores the scale of separation since distances between the robot and obstacles are treated as operational factors rather than qualitative safety measures. Further, there is little in the literature on which one may judge the actual safety of a given SSM algorithm, or directly compare two different implementations.

Parts 1 and 2 of the International Organization for Standards (ISO) standard 10218 (“Robots for Industrial Environments: Safety Requirements. Part 1: Robot,” and “Robots and Robot Devices: Safety Requirements. Part 2: Industrial Robot Systems and Integration”) and the upcoming ISO Proposed Draft Technical Specification (TS) 15066 (“Robots and Robotic Devices—Industrial Safety Requirements: Collaborative Industrial Robots”) address the issue of robot speed and separation monitoring, but are derived from the ISO 13855 (“Safety of Machinery—Positioning of Protective Equipment with Respect to the Approach Speeds of Parts of the Human Body”) protocol for safe separation distances. ISO 13855 specifies that the minimum allowable distance, d_{\min} , from the *danger zone* to the detection point, line, plane or bounding region should be calculated by using the following formula:

$$d_{\min} = kT + \delta. \quad (1)$$

Here, k is the speed derived from data on approach speeds of the intruding human. The value of k is expected to be 2.0 m/s, unless the total computed value of d_{\min} is greater than 0.5 m, in which case k may be set at 1.6 m/s. The total system stopping performance time, T , in seconds, is based on the sum of the maximum time between the actuation of the sensing function and the output signal switching devices being in the off state, t_1 , and the maximum response time of the machine (i.e., the time required to stop the machine), t_2 . The δ term is an additional distance, based on the expected intrusion toward the critical zone prior to actuation of the protective equipment.

The equation defined in (1) has been updated for the ISO TS 15066 to include dynamic properties of the robot

$$d_{\min} = k_H(t_1 + t_2) + k_R t_1 + B + \delta \quad (2)$$

where k_H is the speed of the intruding human, k_R is the speed of the robot, and B is the Euclidean distance required to bring the robot to a safe, controlled stop.

A shortcoming of basing separation distances on ISO 13855, however, is that the standard was intended for static machinery, not dynamic, reconfigurable robotic systems. Updates to the SSM standards are being explored by ISO, and the standards development process will benefit from these metrics to quantitatively assess and compare the safety of robot systems.

This paper proposes a set of metrics for evaluating and comparing collision avoidance algorithms in general, and SSM algorithms specifically. This work draws inspiration from efforts to generalize robot safety in terms of impact force (e.g., [3]–[6]), separation distance (e.g., [7] and [8]), velocity and configuration (e.g., [9]), inertia (e.g., [10] and [11]), and collision cost (e.g., [12]), but focuses efforts on robot platform invariance, industrial applications, and extending reporting beyond safety criteria to include sensor uncertainty, impact on work throughput, and severity of separation failure. The underlying considerations of the evaluation metrics are discussed in Section II. These factors are then integrated into the metrics proposed in Section III. Sections IV and V demonstrate the efficacy of the metrics using simulations and physical trial data.

II. METRICS OF COLLISIONS AND SEPARATIONS

Unlike PFL, which may use comparative metrics such as the Head Injury Criterion [13] or the Gadd Severity Index [14] to address safety when contact between the robot and humans is expected (e.g., in power-assisted manufacturing [15]), SSM does not have a metric by which different robot implementations may be gauged. In this section, the factors for SSM measurement and efficacy are outlined.

A. Collision Detection and Avoidance

The spectrum of collision detection and avoidance algorithms in existence is effectively reduced to one of two possible classifications based on what it is they are measuring. The first classification, *separation*, can further be subcategorized into *spatial* (Euclidean distance) and *temporal* (i.e., time to collision) classifications. Either metric can be utilized directly by a robot's route planner (e.g., plotting paths around obstacles [16], or coordinating robot motions to maintain set distances [17]), or fed into control algorithms as a functional parameter (e.g., potential fields [18], [19] and velocity scaling [20] for collision avoidance).

The second classification reflects the *probability of collision*. Such measurements attempt to capture modeling errors (e.g., [21]), sensor uncertainties (e.g., [22]), and uncertainties of Cartesian space occupancy (e.g., [23]–[25]). These measurements are used for preemptive maneuvering to reduce the probability that a collision will occur in the near future.

An issue encountered with both classification algorithms is that their implementations are limited in scope and not applicable across domains. In industrial environments, for example, the safety boundary definitions for open-chain manipulators bear little relevance to mobile robot collision avoidance systems. A generalized metric is thus needed if the safety of a diversified robotic infrastructure is to be assessed.

B. Metrics Considerations

Several key factors must be considered if a given collision safety metric will be readily adopted by the industry. First, a potential metric must be easily implemented and understood. The maintenance of safety is a task for everyone working around industrial machinery, and the utility of the metric should not require knowledge or experience beyond that of an average

user. Second, the metric must be reliable, repeatable, and robust. The results provided by the metric should be consistent and provide good, useful information. Special considerations for conditions that result in undefined or contrary performances are symptoms of insufficiently robust metrics. Third, the metric should maintain low computational overhead; system integration should not burden or hinder the control of a robot. Finally, the metric should be modular and effectively adaptable to a broad spectrum of robot designs. A metric that is applicable only to open-chain robots, for example, cannot capture the complexity of environments that employ mobile platforms or closed-chain manipulators.

III. METRIC PROPOSAL

In human-centered robotic systems, speed and separation metrics are expected to focus more on preserving human safety in a shared environment than the completion of a given task. However, concern for task completion cannot be ignored lest the usability of the robot be eliminated. A middle ground must therefore be achieved: a robot should be able to accomplish its given task, but if an obstacle enters its work zone, it must respond in a safe and reliable fashion. If the robot is not in danger of impacting an obstacle, its motion should not be hindered beyond the normal physical constraints of the manipulator or task. However, if an obstacle is present, the robot's joint velocities should be scaled down to limit the power and force of an accidental collision.

A. System Safety

A proposed evaluation metric of a safety algorithm is presented in order to assess the actual safety of said system. This metric includes three different measured collision components to establish the degree of safety.

- 1) The safe separation between the robot and its obstacles.
- 2) The system uncertainty as a function of measurement error.
- 3) The severity of a collision, if one were to occur.

This metric is currently intended for relative comparisons between robotic implementations, though ongoing research focuses on absolute safety indices.

The first collision metric component focuses on evaluating the separation between the robot and all potential obstacles in the workspace. There are two related separation metrics used in robot safety literature: separation distance (e.g., [26] and separation time (*time to collision*, or TTC, e.g., [27])). The separation distance, $d_{R_t H_{j,t}}$, between the robot at position R and a given obstacle, j , located at H_j at time t , is given as the Euclidean distance between the two representative point coordinates in Cartesian space

$$d_{R_t H_{j,t}} = \|R_t - H_{j,t}\| - \delta. \quad (3)$$

The variable term δ is an additional distance used to improve the accuracy of the separation estimate by accounting for boundaries of the robot and obstacle. For example, if R and H_j are represented by bounding spheres, δ would equal the sum of their respective radii.

In contrast, TTC measures the minimum time separating a moving robot's link (i.e., the commanded velocity, v_R , is not 0)

from any given obstacle j in the workspace. The relative velocity, \dot{d}_{RH_j} , and the acceleration, \ddot{d}_{RH_j} , of approach are negative when the distance between the robot and the obstacle closes. It is assumed that v_R is known with certainty.

Given \ddot{d}_{RH_j} at time g , TTC is approximated by

$$\text{TTC} = \forall j \min \left\{ \text{ttc}_j = \frac{d_{RH_j}}{|V_j|} \right\} \quad (4)$$

where the velocity profile V_j of duration b is defined as

$$V_j = \begin{cases} \dot{d}_{RH_j} + \int_g^{g+b} \ddot{d}_{RH_j} & \text{if } \dot{d}_{RH_j} + \int_g^{g+b} \ddot{d}_{RH_j} \neq 0 \\ v_R & \text{otherwise.} \end{cases} \quad (5)$$

Unlike other common TTC equations (e.g., [27]), certain assumptions are made regarding the nonholonomic capabilities of the robot and mobile obstacles in the environment. If $\dot{d}_{RH_j} + \int_g^{g+b} \ddot{d}_{RH_j} = 0$, there is no relative motion between the obstacle and the robot (either the obstacle and the robot are traveling parallel to one another, or the relative motion is expected to reverse direction instantaneously), TTC is calculated as though the direction of the robot could change at v_R instantaneously toward the obstacle. If v_R is also 0, the time to collision is infinite.

The second collision metric component quantifies the expected safety of the robot system. With a perfect detection system, the positions of the robot joints and any obstacles (e.g., bounding boxes or centroids in Cartesian space) are known with a high degree of certainty. In the presence of noise, however, the measurement uncertainty poses a hazard for any collision metric. As the separation distance approaches the magnitude of the noise variance, the likelihood of collision approaches 1 nonlinearly. The probability of collision, p , with object j is thus derived from the sensor uncertainty, and it is assumed that p increases faster as the measured separation distance approaches 0. An estimation of this probability, $p_{j,t}$, at time t , is given as

$$p_{j,t} = \begin{cases} \exp \left(-\frac{q(d_{R_t H_{j,t}} - \varepsilon_{H_{j,t}})}{d_{R_t H_{j,t}}} \right), & \text{if } d_{R_t H_{j,t}} > \varepsilon_{H_{j,t}} \\ 1, & \text{otherwise.} \end{cases} \quad (6)$$

where $\varepsilon_{H_{j,t}}$ is the sensor-driven position error estimate, calculated as the normalized average measurement error for each axis coordinate j_n as

$$\varepsilon_{H_{j,t}} = \left\| \left(\frac{1}{2\beta + 1} \sum_{k=t-\beta}^{t+\beta} H_{j,k} \right) - H_{j,t} \right\|, \quad \beta \in \mathbb{N} \quad (7)$$

where $\beta \geq 1$ is the size of one half of the low-pass filter sample window. For all tests performed in subsequent sections, a value of $\beta = 3$ was used for sampling. The value of q is based on what is expected to be a low probability of collision

$$q = -\ln(p_{\text{low}}) \frac{u}{u-1}, \quad (8)$$

which is derived from the first conditional of (6). Here, u is a scalar multiplier of the noise factor beyond which it is given that there is a low probability of collision, p_{low} (for these tests

$p_{\text{low}} = 0.05$). For this set of metrics, it is assumed that a separation distance equal to twice the magnitude of the noise (i.e., $u = 2$) has a low probability of collision, so $q \approx 6$.

The system uncertainty has a direct correlation with the robot's reliability to maintain a safe working environment. As the sensor error-driven uncertainty increases, the likelihood that the robot is or will soon be in a collision state increases. Similarly, as the separation distance decreases, so, too, does the degree of tolerable sensor error.

It should be noted that this metric for collision probability is simplified and generalized, and that alternative methods, such as Monte Carlo estimations (e.g., [12]) and probabilistic agent reactions (e.g., [28]), may be utilized quite effectively here instead. However, alternative approaches typically require considerably more computational effort, and may be applicable only to select robot configurations.

A limitation with both the separation distance and TTC metrics is that neither succinctly quantifies the complexity of the safety question. Merely gauging the distance between the robot and an obstacle, for instance, does not reflect the time remaining until contact. Similarly, the time separating the robot and an obstacle does not necessarily imply a safe operational distance. One must therefore address the issue of what it means for one algorithm variant to be "safer" than another. As an illustrative example, consider two possible configurations of identical robots.

- A 100 kg robot, R_1 , is moving toward a stationary object at velocity v_1 , and has a minimum separation distance of 0.5 m.
- A 100 kg robot, R_2 , is moving toward a stationary object at velocity v_2 , and has a minimum separation distance of 1.0 m.

Knowing only what has been presented, one would assume R_2 was safer. However, if R_1 is moving at 0.05 m/s, and R_2 is moving at 1.0 m/s, R_1 has a TTC of 10.0 s, whereas R_2 has $\text{TTC} = 1.0$ s. Intuition concludes that R_1 is safer. Now consider the inverse problem, where R_2 is moving at 0.1 m/s. Both robots have $\text{TTC} = 10.0$ s, but R_2 never gets closer than 1.0 m from the obstacle while R_1 approaches within 0.5 m. Intuition now points to R_2 as being safer.

The safety of a robot system can be assessed based on the following three observations.

- 1) A robot that has a larger separation distance from an obstacle is safer than one with equal momentum but with a smaller separation distance.
- 2) A robot that has a slower speed is safer than one of equal mass moving faster.
- 3) A robot that has a lower mass is safer than one with a larger mass moving at the same speed.

It is posited that the safety of a robot system is a function of the Euclidean distance and velocity of a robot with mass m_R at time t . This value is further limited by the probability of collision, $p_{j,t}$, defined in (6). A given system is only as safe as its least-safe moment, so, while the safety can be assessed on a continuous basis, postprocess analysis summarizes the system's performance, S_R , as the minimum safety measurement

$$\forall j, t, S_{R_i} = \min\{s_{j,t}\}, s_{j,t} = (1 - p_{j,t}) \frac{d_{R_t H_{j,t}}^2}{m_{R_t} v_{R_t}}. \quad (9)$$

TABLE I
RELATIVE SAFETY MATRIX FOR SPEED AND SEPARATION DISTANCES
ASSUMING CONSTANT m_R AND p . THE VARIABLE v_X IS USED TO IDENTIFY
THE VELOCITY FOR WHICH $S_A = S_B$ IN THE EVENT THAT THE
INEQUALITIES FOR v_A AND v_B , AND d_A AND d_B ARE IDENTICAL

Distance	Speed			$v_X = \frac{v_A d_B^2}{d_A^2}$
	$v_A < v_B$	$v_A = v_B$	$v_A > v_B$	
	$d_A < d_B$	$d_A = d_B$	$d_A > d_B$	
$d_A < d_B$	$?_A$	$S_A < S_B$	$S_A < S_B$	$\begin{cases} S_A < S_B & \text{if } v_B < v_X \\ S_A = S_B & \text{if } v_B = v_X \\ S_A > S_B & \text{if } v_B > v_X \end{cases}$
$d_A = d_B$	$S_A > S_B$	$S_A = S_B$	$S_A < S_B$	
$d_A > d_B$	$S_A > S_B$	$S_A > S_B$	$?_B$	
				$\begin{cases} S_A < S_B & \text{if } v_B > v_X \\ S_A = S_B & \text{if } v_B = v_X \\ S_A > S_B & \text{if } v_B < v_X \end{cases}$

That is, for robot link R_i , the total system safety is defined as the minimum safety value encountered for any obstacle j over all times t . If a robot is compared only to itself, however, the mass term may be omitted. Similarly, assuming a constant probability of collision, the p term may also be omitted, and the safety term for the robot can be written as $S = (d^2/v)$. For a given configuration, the safety performance of two different trials, A and B , can be assessed by validating the equality

$$S_A = \frac{d_A^2}{v_A} \frac{d_B^2}{v_B} = S_B.$$

A relative safety matrix (Table I) can be defined to quickly identify which motion profile is safer than another for a given robot. The complexity of the computation of robot safety is asymptotically driven by the number of obstacles present, J , the number of tracked links on the robot, K , and the size of the sampling window, β , used to derive the probability of collision. For any given time step, it can be shown that the separation distances can be computed in $O(JK)$ time, and all the probabilities of collision computed in $O(JK\beta)$ time. Thus, it can be shown that the computational cost of calculating the total system safety is bounded to $O(JK\beta)$ time.

The third collision metric focuses on measuring the severity of an observed collision. In the event of a failure in separation maintenance, the robot should come to a safe, monitored stop. The bounding region penetration severity, C_j , for obstacle j is then reported for all K tracked robot links, and is based on both the degree of overlap of the robots' and obstacles' bounding regions, and the severity of penetration into an obstacle's bounding volume

$$C_j = \max \left\{ \frac{\bigcup_{k=1}^K |c_{k,j}|}{A_{H_j}}, \quad \forall k \max \left(\frac{(r_{H_j} - f_{R_k, H_j})}{r_{H_j}} \right) \right\} \quad (10)$$

where $c_{k,j}$ is the intersection for the link k with the obstacle j , and A_{H_j} is the total bounding volume for obstacle j . The value r_{H_j} is the maximum reach of the obstacle toward the robot, and f_{R_k, H_j} is the distance from the center of the obstacle's bounding volume, H , to the closest point of the robot's bounding volume.

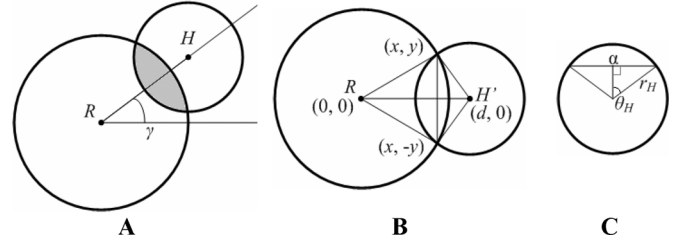


Fig. 1. (A) Overlap for two 2D colliding bounding circles. (B) Transforming the obstacle bounding circle to be inline with the robot bounding circle reduces the problem complexity to two unknown variables: x and y . (C) Each asymmetric half of the collision region is found by subtracting the area of the triangle with lengths $\alpha = 2y$ and r , from the area of the incidental wedge

If H is inside the robot's bounding volume, f_{R_k, H_j} is 0. For bounding circles, f_{R_k, H_j} is calculated as

$$f_{R_k, H_j} = \begin{cases} \|H_j - R_k\| - r_{R_k}, & \text{if } \|H_j - R_k\| \geq r_{R_k} \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

where r_{R_k} is the effective reach of the robot.

While no explicit geometrical representations of the robot or obstacle are specified, more physically accurate bounding regions necessarily provide increased severity accuracy at the expense of increased computational complexity.

Consider the simple case of the penetration of 2D bounding circles for a single robot and a single object projected onto an arbitrary X - Y plane (Fig. 1). Assume that H is the centroid of a single object with radius r_H . The center of the point robot, R , is located at the robot's base, and r_R is the distance to the maximum effective reach of the robot.

If the assertion that $|\overline{RH}| \leq (r_R + r_H)$ is true, then a collision has occurred. Its severity is measured by computing the area defined by the overlap of the two bounding circles [Fig. 1(A)]. To simplify the computation of the overlap, assume R is centered at $(0, 0)$, and transform $H = (H_x, H_y)$ by rotating it about R to $H' = (H_x', 0)$. The distance between the two centers remains constant at d , so $H' = (d, 0)$ [Fig. 1(B)].

In Fig. 1(B), the points (x, y) and $(x, -y)$ mark the coordinates of the two intersections between the circles. The value of x can be calculated by

$$x = \frac{1}{2d} (r_R^2 - r_H^2 + d^2).$$

Plugging x back into the original circle equation for R , the value of y can be found as

$$y = \frac{1}{2d} \sqrt{4d^2 r_R^2 - (r_R^2 - r_H^2 + d^2)^2}.$$

The collision region is defined as the two halves of the asymmetric lens. The bisecting chord has length $\alpha = 2y$. The half-angles for the arcs defined by the respective circle centers and points of intersection [see Fig. 1(C) illustrating this half angle for the circle centered at H] are thus defined by

$$\theta_H = \sin^{-1} \left(\frac{y}{r_H} \right), \quad \theta_R = \sin^{-1} \left(\frac{y}{r_R} \right).$$

The area of a half-lens is equal to the area of the wedge

$$A_{W_R} = \theta_R r_R^2, \quad A_{W_H} = \theta_H r_H^2$$

minus the area of the incident triangles defined by the respective circle centers and the points of intersection

$$A_{T_R} = (d - x)y, \quad A_{T_H} = xy.$$

Because only a single robot and a single obstacle are considered, the collision region $c_{k,j}$ is computed as

$$c_{k,j} = \begin{cases} (A_{W_R} - A_{T_R}) + (A_{W_H} - A_{T_H}), & \text{if } d \leq r_R + r_H \\ 0, & \text{otherwise.} \end{cases}$$

And the area for the bounding circle for obstacle H is

$$A_H = \pi r_H^2.$$

Thus, for a given obstacle H , in the event of a collision the severity of a single robot is defined as

$$C_j = \max \left\{ \frac{(A_{W_R} - A_{T_R}) + (A_{W_H} - A_{T_H})}{\pi r_H^2}, \frac{(r_H - f_{R,H})}{r_H} \right\}. \quad (12)$$

The postprocess 2D equation given in (12) is used in the case studies outlined in Sections IV and V, and can be computed in $O(JK)$ time. The method described above is readily extended for arbitrary N -dimensional bounding spheres, but increases the computational time. For bounding volumes other than N -dimensional spheres, however, the quantification of penetration grows more computationally expensive as the complexity of the volumes increases. Generalized volume penetrations are addressable (e.g., [29] and [30]), but are typically expressed in terms of penetration distances, not areas of overlap. Alternatively, methods for computing the area of overlap of 2D polygons with M sides exist for convex volumes with complexity $O(M \log M)$ [31], and volumes without the convex assumption with complexity $\leq O(M^4)$ [32].

B. System Performance

It is worth reiterating the importance of the distinction between *safe* and *usably safe*. A trivially safe robotic system is one that never moves (i.e., $v_R = 0$ for all times q), and thus has $TTC = \infty$. For a comparison between SSM algorithms, the performance of a system must demonstrate some level of usefulness, which draws attention to the tradeoff between safety and productive capacity. It is thus necessary to quantify the effect a given algorithm will have on productivity.

There is difficulty in quantifying the impact a given safety system has on productivity. In order to remain general and applicable to the broad spectrum of robot tasks, a common factor must be used. For example, “the distance traveled along the X - Y plane” is not valid for a robot inserting screws and moving only on the Z axis.

The one common factor all robots have is that they are required to perform some job over time. This expected time measurement provides a convenient and generalized basis for com-



Fig. 2. The inverted 6DOF robot arm mounted on a 7th axis rail. The workcell is monitored by dual safety laser scanners (one shown in the lower-right) and protected by a fence with three gate interlocks (one shown on the left).

parison for a broad spectrum of robot tasks. The productivity can thus be quantified as

$$P_R = \frac{\hat{\tau}}{\tau_R}. \quad (13)$$

Here, we evaluate the expected time necessary for the robot to complete its task, $\hat{\tau}$, versus the time to perform the same task with the safety system in effect, τ_{R_i} . The result indicates the percentage of the time the robot was doing useful work.

C. Metrics Summary

The properties discussed in the preceding Sections III-A and III-B can be thus combined to provide a basis for evaluating the safety and performance of a given robot safety protocol implementing separation monitoring based on circular bounding regions such as the ISO TS 15066 SSM. The scalar defined by (9) provides a basis for system safety assessment based on object separation and system uncertainty. In the event of a collision between the robot and an obstacle (i.e., the system safety is 0), the severity of the penetration into the obstacle’s bounding volume can be found using (10), which has been generalized for bounding spheres in (12). For a given trial, the value calculated by (13) succinctly describes the overall productivity with the SSM system engaged.

IV. CASE STUDY: SIMULATION TRIALS

A. Robot Configuration

A robot testbed at the National Institute of Standards and Technology (NIST) containing an inverted rail-mounted 6DOF industrial manipulator (Fig. 2) was configured for manufacturing process test and evaluation. The workcell is enclosed by a security fence with three interlocked gates around the perimeter, and is monitored by three optical safety systems:

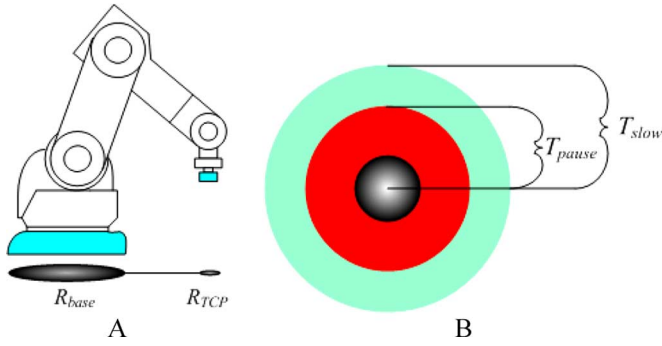


Fig. 3. (A) The robot simulation tracks the XY planar centers of the base and TCP and (B) maintains separation thresholds centered around each joint for velocity scaling.

a dual inward-facing safety laser scanner system (one laser scanner mounted on each post of the rail supporting the robot) that tracks objects as they move in the work zone, a ceiling-mounted stereo camera directly above the rail, and a six-camera motion capture system distributed around workcell that tracks retro-reflective targets on the robot arm at the tool center point (TCP) and on a mannequin torso used for safety testing.

To accommodate repeatable testing, a simulator was written to mimic the capabilities of the robot testbed modulated by different 2D (top-down), threshold-based SSM algorithms (Fig. 3). The simulated robot consisted of a mobile base platform whose motions were limited by a trapezoidal acceleration profile and a single arm linkage that could be set to any length and joint angle. The position of the robot base (R_{base}) along with the orientation and length of the arm define the TCP position [R_{TCP} , Fig. 3(A)].

The robot motions simulate a robot picking up an object at one location, and then placing that object down at a specified location 5 m away. At each extreme of the swing, the robot pauses for 4.5 s, and then reverses its direction. The simulation consisted of ten repetitions of this back-and-forth cycle. For all trials, the robot's commanded velocity was 0.5 m/s, and had a constant acceleration profile of 0.3 m/s². Because only a single robot was considered for all trials, the mass of the robot was omitted for comparisons as discussed in Section III-A. The simulation was evaluated in real time at 30 Hz, and, when uninterrupted, completed the task in 343.6 s on average.

B. Operator Path Data

Errors of human pose estimation result in unstable or otherwise unpredictable robot behaviors. Also, instantaneous velocities and accelerations calculated from raw, noisy track data make TTC approximations unreliable. It is for these reasons that it can be argued that SSM algorithm testing using recorded raw, nonlinear sensor information is an imperative.

The trajectory of a human operator moving through the NIST testbed was recorded using the safety laser scanner system (Fig. 4) at 16.6 Hz. The tracked human's position is estimated by computing the centroid of the detected coordinate points. This motion was then played back through the robot simulator to provide a repeatable test scenario. The simulated

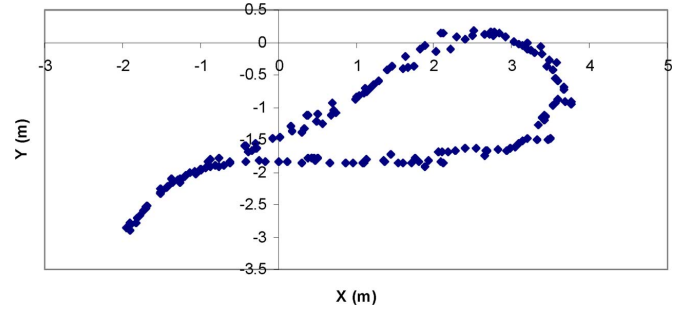


Fig. 4. Full path projection of a safety laser scanner-tracked human operator walking through the robot testbed.

robot TCP performed ten repetitions of a movement between the 0.0 and 5.0 m marks along the X axis, while the Y axis coordinate was set constant at -1.0 m. The robot was commanded to pause for 4.5 s at each extreme swing. Without any SSM algorithm to control the velocity of the robot, the closest approach distance to the logged human data was 0.68 m, at which point $v_R = 0.5$ m/s. Given a probability of collision at that time of $p = 0.0027$, the total system safety was 0.9223.

C. SSM Description

Commercial SSM solutions (e.g., [33]–[35]) consist of velocity-scaled threshold regions centered on the robot's joints [Fig. 3(B)] that define three operational states: *clear*, *slow*, and *pause*. The *clear* state indicates that there are no detected potential collisions, and the robot moves at 100% of its programmed trajectory velocity. The *slow* state marks that a potential collision state exists, and the robot's trajectory velocity is reduced. And the *pause* state indicates that a collision is imminent, and the robot comes to a controlled stop. Two custom SSM algorithms, Basic SSM and Tri-Modal SSM, similar to the commercial solutions were evaluated for $r_R = 0.1$ and $r_H = 0.1$ m. Both algorithms transition from one state to another based on thresholds comparing separation distances, and are modeled after commercial SSM products that modulate the robot's velocity based on user-defined warning zones.

The Basic SSM algorithm relies on a single threshold distance value to switch between states, and is most congruent with the 2D space monitoring system described in [36]. When d_{RH_j} is less than the pause threshold, T_p , the robot transitions from *clear* to *pause*. To illustrate that changing parameters have a measurable effect on the safety performance of a given algorithm, the value of T_p for the Basic SSM was varied and evaluated for $T_p = 1.0$ to $T_p = 2.2$ m in 0.2 m increments.

The Tri-Modal SSM algorithm functions identically to the Basic SSM implementation, but utilizes a second distance threshold, $T_s > T_p$, to transition from *clear* to *slow*, in which the robot is commanded to reduce its trajectory velocity by a set percentage (in this case, 50%). Such functionality is modeled after the 2D (e.g., [33]) and 3D (e.g., [34]) zone-monitoring safety systems available commercially. In the Tri-Modal SSM implementation, the robot transitions from *slow* to *pause* when $d_{RH_j} < T_p$. An evaluation of the effects of parameter modulation on the set of safety metrics was implemented for the Tri-Modal SSM algorithm, where T_s was varied between

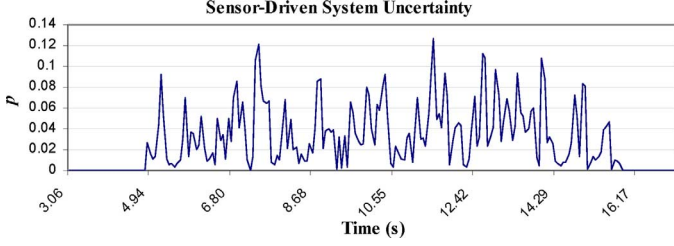


Fig. 5. System uncertainty driven by the measurement noise of the tracked human's position, ε_H .

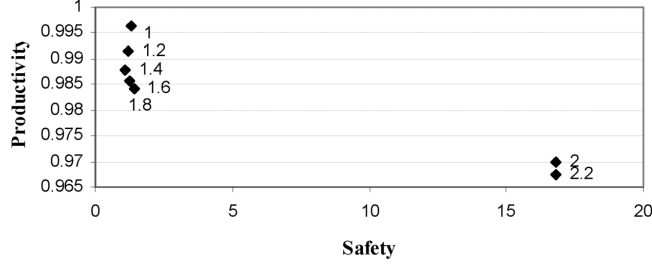


Fig. 6. Performance of the Basic SSM for the hybrid simulation as a function of T_p (labeled).

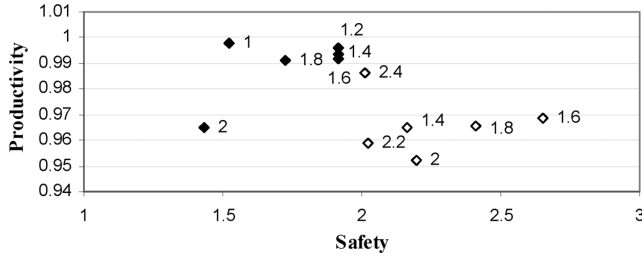


Fig. 7. Performance of the Tri-Modal SSM for the simulated robot as a function of T_s (labeled), with $T_p = 0.5$ m (solid) and $T_p = 1.0$ m (outline).

1.0 and 2.4 m in increments of 0.2 m for two different values of T_p : $T_p = 0.5$ and $T_p = 1.0$ m.

D. SSM Performance

As can be inferred from Fig. 4, the tracked human data is noisy, and, when run without any SSM, produces the system uncertainty for $r_H = 0.5$ m, shown in Fig. 5. However, these values are still relatively low with a maximum of 8.9%, and average of 2.2%. The results for using the Basic SSM for the simulation are plotted in Fig. 6, and the Tri-Modal SSM in Fig. 7. The measurements and safety values for the active robot at the point of minimum separation distance are given in Table II. In all trials, the robot stopped prior to any possible collision states, so in each the collision severity $C = 0$.

From these results, it can be shown that changing the value of T_p for the Basic SSM has the expected result of improving the safety of each algorithm, but also adversely affects the productivity. From the Tri-Modal SSM evaluation, modifying the value of T_s also has an effect on the productivity and safety, but the variance of the system's safety is more fine resolution than that seen with T_p . For both algorithms, the minimum system safety was greater than that of the robot without any SSM active, as is seen in row 1 of Table II.

TABLE II
SEPARATION AND SAFETY MEASUREMENTS FOR THE SIMULATION AT d_{min} USING THE BASIC (B) AND TRI-MODAL (TM) SSM ALGORITHMS

	T_s (m)	T_p (m)	d_{min}^* (m)	$v_R @ d_{min}$ (m/s)	$p @ S_R$	S_R^*	$C @ d_{min}$
---	---	---	0.682	0.450	0.003	0.922	0
B	---	1.0	0.694	0.162	0.005	1.313	0
B	---	1.2	0.613	0.320	0.004	1.170	0
B	---	1.4	0.557	0.284	0.004	1.090	0
B	---	1.6	0.534	0.212	0.004	1.242	0
B	---	1.8	0.437	0.122	0.003	1.409	0
B	---	2.0	2.504	0.036	0.002	16.81	0
B	---	2.2	2.600	0.014	0.002	16.81	0
TM	1.0	0.5	0.691	0.250	0.004	1.520	0
TM	1.2	0.5	0.700	0.250	0.003	1.914	0
TM	1.4	0.5	0.700	0.250	0.003	1.914	0
TM	1.6	0.5	0.693	0.250	0.003	1.914	0
TM	1.8	0.5	0.658	0.250	0.004	1.726	0
TM	2.0	0.5	0.599	0.250	0.004	1.431	0
TM	1.4	1.0	0.672	0.194	0.004	2.163	0
TM	1.6	1.0	0.631	0.142	0.004	2.651	0
TM	1.8	1.0	0.586	0.142	0.004	2.408	0
TM	2.0	1.0	0.560	0.142	0.004	2.196	0
TM	2.2	1.0	0.536	0.124	0.004	2.020	0
TM	2.4	1.0	0.499	0.052	0.004	2.000	0

* - Reported values are minimum d_{min} , S_R , and C while the robot is moving
 ** - Performance for running the simulation without any SSM running is provided for comparison

A trend in the data also emerges based on the properties of the simulation. Because the recorded human operator's motions do not deviate from their expected trajectory, only the robot's reactions affect the safety performance of the simulation. This is illustrated when T_p is increased from 1.8 m to 2.0 m in the Basic SSM test. Recall that the robot pauses for 4.5 s at each drop point. The closest approach of the recorded human to the robot with $T_p \geq 2.0$ m occurs when the robot is supposed to emerge from this pause state and resume motion. At this time, $v_R = 0$ m/s, and d is between 1.8 m and 2.0 m. If $T_p = 1.8$ m, the robot would have started moving, and would have had a comparatively much lower safety rating than it actually did. Instead, the robot remained idle until the recorded human track moved away, thus resulting in a much larger separation distance at the point of closest approach for the active robot.

V. CASE STUDY: PHYSICAL TRIALS

After the initial simulation trials, the two SSM algorithms were ported to the robot controller described in Section IV-A for physical system testing. The Basic SSM was configured with thresholds based on an analytical approximation of (2)

$$T_p = v_H \left(\frac{v_R}{a_{R,max}} + t_2 \right) + v_R(t_2) + \frac{v_R^2}{2a_{R,max}} + (r_R + r_H) \quad (14)$$

where $r_R = 0.25$ m, $r_H = 0.9$ m, and $a_{R,max} = 5.0$ m/s² is the maximum acceleration of the robot, the human velocity was set at the ISO 13855 worst-case value of $v_H = 2.0$ m/s. The reaction time, t_2 , is derived by the sum of the safety system's detection and reaction times, which were empirically calculated to be 0.297 and 0.113 s, respectively. The Tri-Modal SSM algorithm was set with $r_R = 0.2$, $r_H = 0.8$, $T_s = 2.0$, and $T_p = 1.0$ m. Given bounding circles for the robot and human of 0.25 and 0.9 m, respectively, the expected separation between

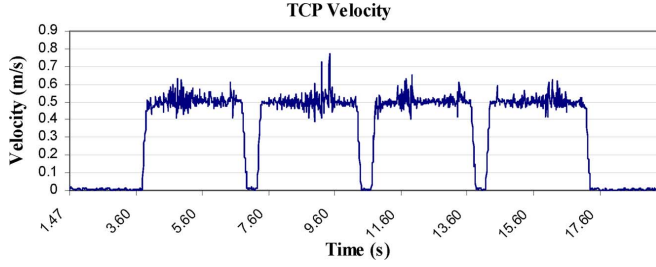


Fig. 8. Instantaneous velocity profile of the robot without any SSM running.

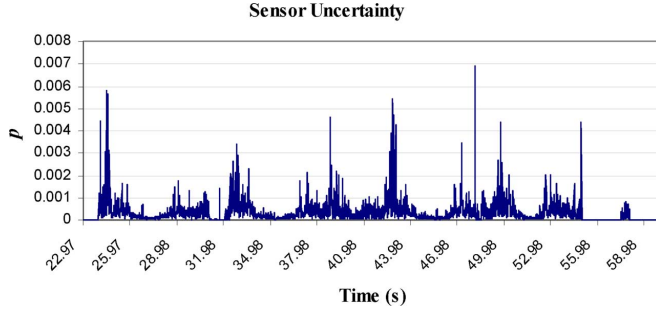


Fig. 9. Sensor uncertainty estimate during the Basic SSM trials.

the center points of the robot and human is expected to be greater than 1.15 m while the robot is moving. Failures to maintain this center-to-center distance result in separation distances $d < 0$.

The safety laser scanner configuration used to generate the tracked human data in Section IV was used as the basis for the SSM. The physical trials reflected the simulation evaluation in that the robot's motions were a simple back-and-forth trajectory using only the 7th axis linear rail for motion, but the range of this motion was reduced to 1.5 m between the two zeniths, and the number of repeats was reduced to two. Once again, the robot was commanded to move between the two points at 0.5 m/s. A velocity profile of the robot without any SSM running is shown in Fig. 8, which illustrates the robot completing its program in 13.54 s.

A mannequin lower torso mounted on a wheeled base was used as a surrogate human in the robot work zone. It was moved manually toward and away from the robot along the principal axis of the robot's motions using an extended handle. Because the mannequin's motions were moved by hand, they were not reproducible. As such, these trials are used here only as examples of the metrics' capacity for comparing two SSM algorithms for a given robot in an unstructured evaluation.

The motion capture system discussed in Section IV-A provided the ground truth for these tests via the targets located on the robot's TCP and at the centroid of the torso. The accuracy and speed of the camera system are much greater than the tracker [in Fig. 9, it can be seen that the sensor uncertainty computed using (7) is an order of magnitude lower than that of the safety laser scanner system illustrated in Section IV].

The performance measurements and safety assessments resulting from running the Basic SSM are shown in Figs. 10 and 11, respectively. According to the safety metrics described in this paper, the Basic SSM algorithm was safe ($S > 0$) at all time steps (Fig. 11). Indeed, the measurements confirm $d > 0$

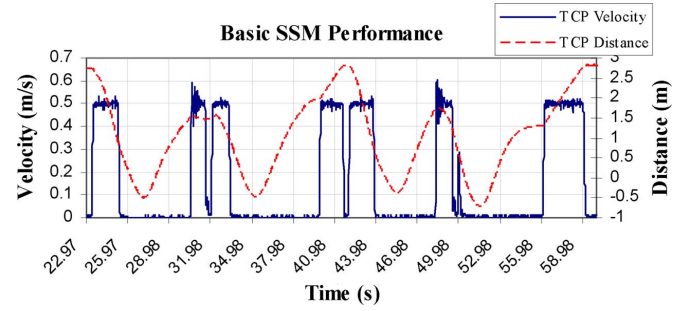


Fig. 10. Measured velocity of the robot's TCP and the relative separation distance of the TCP and the mannequin during the Basic SSM trial.

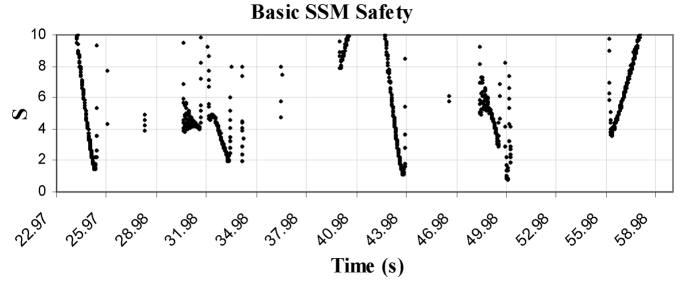


Fig. 11. Calculated safety of the robot implementation over time. At no time does the safety fall to zero, indicating the robot was never in an explicitly unsafe configuration with regard to the mannequin.

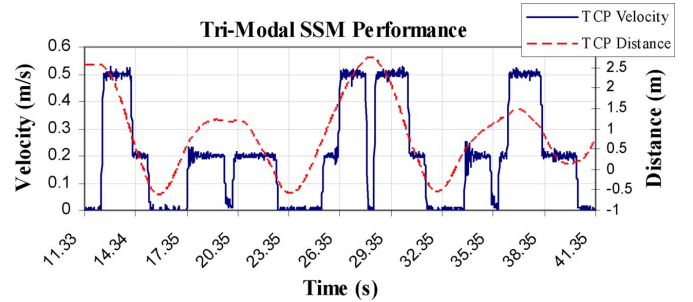


Fig. 12. Measured velocity of the robot's TCP and the relative separation distance of the TCP and the mannequin during the Tri-Modal SSM trial.

for all times that $v_R > 0$. As can be seen in Fig. 10, the separation distance falls below T_P on four occasions. However, in each instance, the TCP velocity was reduced to 0 m/s, and, as a result, the safety of the system never fully approaches 0, and any collision of the bounding regions was the consequence of the mannequin approaching the stationary robot (i.e., $C = 0$ at all time steps). The robot was able to complete its program in 36.04 s (the time from the robot's initial motion to the time it came to a final stop), for a productivity value of $P_R = 37.6\%$.

In contrast, when using the Tri-Modal SSM, the *slow* capacity of the algorithm increased the productivity of the robot, which completed its program in 28.24 s ($P_R = 47.9\%$) as illustrated in Fig. 12. However, as can be seen in Fig. 13, the robot also demonstrated a minimum safety of 0 ($C > 0$ for some time t) based on the three collision events witnessed in Fig. 12. Because the area of the active TCP's bounding region is only about 6% the area of the mannequin's bounding region, the collision severities shown in Fig. 14 indicate that the severity factor is

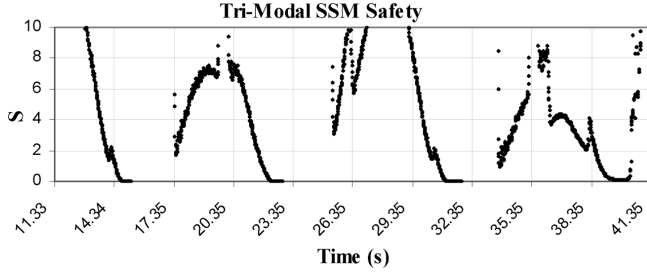


Fig. 13. Calculated safety of the robot implementation over time. Again, the safety falls to zero on three occasions, implying unsafe robot motions.

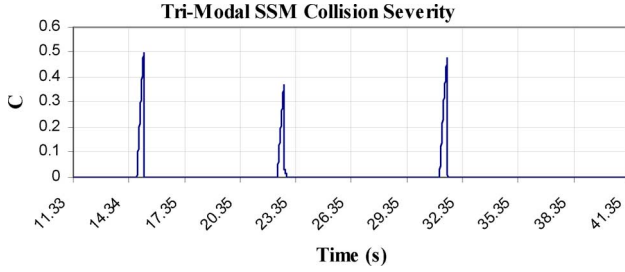


Fig. 14. Analysis of the collision severity shows the percentage of overlap of the robot's and human's bounding regions.

dominated by the penetration distance. The degree of penetration is directly attributed to a combination of delay in the safety laser scanner tracking system and setting T_p equal to the sum of the bounding regions' radii. Because of the effects of latency and sensor reliability, an amendment to the SSM algorithm accounting for sensor uncertainty has been proposed for ISO TS 15066.

VI. DISCUSSION

A set of performance metrics for the comparison of the safety of robotic systems has been described and demonstrated. Its implementation is simple and intuitive, since the safety of a system is a function of the robot's mass, velocity, separation distance, and probability of collision. Computationally, the overhead of the safety metrics is low at $O(JK\beta)$ given J obstacles, K tracked links, and the probability of collision filtering window of size β . Initial trials have demonstrated that the metrics are modular in that they are applicable to either industrial robot arms or mobile robot platforms.

From the initial trials, it is evident that even a relatively simple SSM algorithm can readily improve the safety of a robot so long as it reduces the speed of a robot while it is in the vicinity of an obstacle. The difficulty lies in demonstrating not only how much safer a given algorithm is over another, but also identifying the point in time at which an algorithm was the least safe. Such information allows researchers to isolate and identify potential issues, and improve their systems as a result. The analysis of simulated and physical robot implementations in this report demonstrates the capacity of the proposed set of metrics to address these issues.

Due to ISO 13855 providing the basis for robot safety standards such as the draft ISO TS 15066, the SSM distance thresholds equations derived from ISO 13855 also provide benchmarks for future algorithm development and comparison. By

providing a set of metrics that help define a baseline for performance, it becomes possible to promote the growth and acceptance of novel and innovative approaches for maintaining safety in industry. When compared with current standards, SSM systems that have both higher productivity and safety ratings for identical testing conditions are, naturally, superior. An additional benefit is that an intelligent robot system may utilize metrics such as these to identify when a given operational situation is unsafe based on either sensor uncertainty or declinations in measured safety.

As the use of robotics in human-occupied spaces becomes more prevalent, industry will inevitably witness more direct interactions between robots and operators. Ongoing research focuses on integrating the metrics developed herein into prototype manufacturing workcells for collaborative task completion. Current efforts seek to assimilate the Occupational Safety and Health of the German Social Accident Insurance (IFA, formerly BGIA) risk recommendations for tolerable injury values [36] and these metrics into a combined SSM-PFL workcell solution.

REFERENCES

- [1] *Robots for Industrial Environments: Safety Requirements. Part 1: Robot*, ISO 10218-1:2011, 2011.
- [2] *Robots and Robotic Devices: Safety Requirements. Part 2: Industrial Robot Systems and Integration*, ISO/FDIS 10218-2:2011, 2011.
- [3] A. Bicchi, M. A. Peshkin, and J. E. Colgate, "Safety for physical human-robot interaction," in *Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. New York, NY, USA: Springer, 2008, pp. 1335–1348.
- [4] M. Nokata, K. Ikuta, and H. Ishii, "Safety-optimizing method of human-care robot design and control," in *Proc. IEEE Int. Conf. Rob. Autom.*, 2002, pp. 1991–1996.
- [5] K. Ikuta and M. Nokata, "Safety evaluation methods of design and control for human-care robots," *Int. J. Rob. Res.*, vol. 22, no. 5, pp. 281–297, 2003.
- [6] S. Haddadin, A. Albu-Schäffer, and G. Hirzinger, "Requirements for safe robots: Measurements, analysis and new insights," *Int. J. Rob. Res.*, vol. 28, no. 11–12, pp. 1507–1527, 2009.
- [7] M. Chen and A. M. S. Zalzala, "A genetic approach to motion planning of redundant mobile manipulator systems considering safety and configuration," *J. Rob. Syst.*, vol. 14, no. 7, pp. 529–544, 1997.
- [8] P. Trautman and A. Krause, "Unfreezing the robot: Navigating in dense, interacting crowds," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Syst.*, 2010, pp. 797–803.
- [9] B. Lacevic and P. Rocco, "Kinetostatic danger field—A novel safety assessment for human-robot interaction," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Syst.*, 2010, pp. 2169–2174.
- [10] D. Kulić and E. Croft, "Safe planning for human-robot interaction," in *Proc. IEEE Int. Conf. Rob. Autom.*, 2004, pp. 1882–1887.
- [11] M. Zinn, "A new actuation approach for human friendly robotic manipulation," Ph.D. dissertation, Stanford Univ., Stanford, CA, USA, 2005.
- [12] D. Althoff, J. J. Kuffner, D. Wollherr, and M. Buss, "Safety assessment of robot trajectories for navigation in uncertain and dynamic environments," *Auton. Rob.*, vol. 32, no. 3, pp. 285–302, 2012.
- [13] J. Versace, "A review of the severity index," in *Proc. 15th Stapp Conf.*, 1971, pp. 771–796.
- [14] C. W. Gadd, "Use of weighted impulse criterion for estimating injury hazard," in *Proc. 10th Stapp Car Crash Conf.*, 1966, pp. 164–174.
- [15] S. Lee, S. Hara, and Y. Yamada, "A safe measure for control mode switching of skill-assist for effective automotive manufacturing," *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 4, pp. 817–825, Oct. 2010.
- [16] K. Hosoda, K. Sakamoto, and M. Asada, "Trajectory generation for obstacle avoidance of uncalibrated stereo visual servoing without 3D reconstruction," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Syst.*, 1995, pp. 29–34.
- [17] P. Bosscher and D. Hedman, "Real-time collision avoidance algorithm for robotic manipulators," in *Proc. IEEE Int. Conf. Technol. Pract. Rob. Appl.*, 2009, pp. 113–122.
- [18] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," *Int. J. Rob. Res.*, vol. 5, no. 1, pp. 90–98, 1986.

- [19] C. Helguera and S. Zegloul, "A local based method for manipulators path planning in heavy cluttered environments," in *Proc. IEEE Int. Conf. Rob. Autom.*, 2000, pp. 3467–3472.
- [20] Y. Zhang, E. K. Antonsson, and K. Grote, "A new threat assessment measure for collision avoidance systems," in *Proc. IEEE Intell. Transp. Syst. Conf.*, 2006, pp. 968–975.
- [21] R. Asaula, D. Fontanelli, and L. Palopoli, "Safety provisions for human/robot interactions using stochastic discrete abstractions," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Syst.*, 2010, pp. 2175–2180.
- [22] C. Fulgenzi, C. Tay, A. Spalanzani, and C. Laugier, "Probabilistic navigation in dynamic environment using rapidly-exploring random trees and Gaussian processes," in *Proc. IEEE/RSJ Int. Conf. Intell. Rob. Syst.*, 2008, pp. 1056–1062.
- [23] J. Miura, Y. Negishi, and Y. Shirai, "Adaptive robot speed control by considering map and motion uncertainty," *Rob. Autom. Syst.*, vol. 54, no. 2, pp. 110–117, 2006.
- [24] K. M. Krishna, R. Alami, and T. Simeon, "Safe proactive plans and their execution," *Rob. Autom. Syst.*, vol. 54, no. 3, pp. 244–255, 2006.
- [25] A. Lambert, D. Guyer, G. S. Pierre, and A. N. Ndjeng, "Collision probability assessment for speed control," in *Proc. 11th Int. Conf. Intell. Transport. Syst.*, 2008, pp. 1043–1048.
- [26] D. Kulic and E. Croft, "Pre-collision safety strategies for human-robot interaction," *Auton. Rob.*, vol. 22, no. 2, pp. 149–164, 2007.
- [27] J. Jansson and F. Gustafsson, "A framework and automotive application of collision avoidance decision making," *Autom.*, vol. 44, no. 9, pp. 2347–2351, 2008.
- [28] D. Althoff, M. Althoff, D. Wollherr, and M. Buss, "Probabilistic collision state checker for crowded environments," in *Proc. IEEE Int. Conf. Rob. Autom.*, 2010, pp. 1492–1498.
- [29] C. J. Ong and E. G. Gilbert, "Growth distances: New measures for object separation and penetration," *IEEE Trans. Rob. Autom.*, vol. 12, no. 6, pp. 888–903, 1996.
- [30] K. Sridharan and S. S. Keerthi, "Computation of a penetration measure between 3D convex polyhedral objects for collision detection," *J. Rob. Syst.*, vol. 18, no. 11, pp. 623–631, 2001.
- [31] M. de Berg *et al.*, "Computing the maximum overlap of two convex polygons under translations," *Theory Comput. Syst.*, vol. 31, no. 5, pp. 613–628, 1998.
- [32] D. M. Mount, R. Silverman, and A. Y. Wu, "On the area of overlap of translated polygons," *Comput. Vision Image Understanding*, vol. 63, no. 1, pp. 53–61, 1996.
- [33] K. Behnisch, Safe Collaboration With ABB Robots: Electronic Position Switch and Safe Move. White paper, ABB Robotics, 2008. [Online]. Available: [http://www05.abb.com/global/scot/scot241.nsf/veritydisplay/46e5494b3e5f9da6c12574e1003eb5be/\\$File/6904%20ROP%20White%20Paper%20SafeMove.pdf](http://www05.abb.com/global/scot/scot241.nsf/veritydisplay/46e5494b3e5f9da6c12574e1003eb5be/$File/6904%20ROP%20White%20Paper%20SafeMove.pdf).
- [34] S. Davies, "Watching out for the workers [safety workstations]," *Manuf., IET.*, vol. 86, no. 4, pp. 32–34, 2007.
- [35] Products & Applications: Safety Laser Scanners. Product guide, SICK, Inc., 2012. [Online]. Available: <https://www.mysick.com/saqqara/pdf.aspx?id=im0043477>.
- [36] BGIA—Institut für Arbeitsschutz der Deutschen Gesetzlichen Unfallversicherung, *BG/BGIA Risk Assessment Recommendations According to Machinery Directive: Design of Workplaces with Collaborative Robots* Feb. 2011, U 001/2009e.



Jeremy Marvel (S'06–M'10) received the B.S. degree in computer science from Boston University, Boston, MA, USA, the M.S. degree in computer science from Brandeis University, Waltham, MA, USA, and the Ph.D. degree in computer engineering from Case Western Reserve University, Cleveland, OH, USA.

He is a Research Scientist at the National Institute of Standards and Technology (NIST), Gaithersburg, MD, USA. He currently leads the measurement science efforts at NIST for robot safety for flexible automation. His research focuses on intelligent and adaptive solutions for robot applications, with particular attention paid to safety, perception, and automated parameter optimization.