2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Generating Whole-body Motion Keep Away From Joint Torque, Contact Force, Contact Moment Limitations enabling Steep Climbing with a Real Humanoid Robot

Shintaro Noda, Masaki Murooka, Shunichi Nozawa, Yohoei Kakiuchi, Kei Okada and Masayuki Inaba

Abstract— **For humanoid robots to perform whole-body motions, a motion planner should generate feasible motions satisfying various constraints including joint torque limitation, friction, balancing, collision, and so on. Furthermore, for life-size humanoid robots to perform higher-load motions, such as climbing ladders, safely, it is important to generate motions which requirements are not too close to the limitations. In this paper, we propose a humanoid motion planner based on Body Retention Load Vector (BRLV), which is a novel index for representing severity of physical constraints: limitation of joint Torque, contact Force, and contact Moment (TFM limitations). By minimizing the norm of BRLV, we obtain humanoid motions that are farthest from TFM limitations. Finally, we evaluate the proposed motion planner in simulation and confirm the effectiveness of the planner through experiments in which a life-size humanoid robot climbs a ladder and a car**

I. Introduction

To generate whole-body motions with some complex contact state's transitions is a very important problem for humanoid robots: for example, climbing ladders and getting into cars (Fig.1). However, humanoid motions involving contacts have a lot of constraints. For maintaining grasp, the humanoid's hand must keep it's position and orientation. For balancing, contact force and moment must be within an appropriate range. Thus, previous research on while-body motion generation have formulated this problem as an optimization or search problem with some constraints [1, 2, 3, 4, 5].

However, most studies focusing on hard motions such as climbing ladder only evaluated their methods in simulation and did not provide any implementation on humanoid robots. Through a lot of trial-and-error in climbing experiments with a real life-sized humanoid robot, HRP-2 [6], we arrived at the conclusion that severity of physical constraints due to heaviness of a real humanoid robot and sensitiveness of modeling error on physical contact condition and constraints are the main problems. Compared to small robots [7, 8], humanoid robots are very heavy. Compared to multi-leg robots [9, 10], humanoid hands are powerless and their feet have no ability to grasp.

In this paper, we propose a novel method for generating whole-body motions, especially for enabling steep climbing motion such as climbing step/vertical ladders. Our method not only satisfies all constraints but also minimizes the severity of physical constraints: limitations of joint Torque, contact Force, contact Moment (TFM limitations). To minimize the severity, we introduce a new index, Body Retention Load Vector (BRLV), and minimize it during an optimization process involving robot states and a selection process involving contact states.

Fig. 1. Steep climbing by a life-sized humanoid robot. Left: Climbing up and down a step ladder (step height=20cm and slope = 70deg). Right: Getting into a car (step height = 30cm).

Compared to previous research studies, contributions of this paper are as follows:

Harada et al. [1] solved the optimization problem with some constraints to generate natural trajectory and posture. The cost functions involved torque, acceleration, and angular momentum. Although this approach is very generic, contact force and moment are also important values to preventing accidents such as slipping and falling. In addition, to consider the physical severity of robots, these values should be normalized with their constraints.

Bouyarmane et al. [3] achieved contact state transitions with dynamic motions using one of the hill-climbing methods. This approach is very fast and adaptive, but, for high-load climbing motions, it may generate some awkward trajectories because there will be physical potential hill on the way of contact states transition. So, our approach is to generate static key poses in climbing and connecting them

S. Noda, M. Murooka, S. Nozawa, S. Kakiuchi, K. Okada and M. Inaba are with Department of mechano-informatics, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan s-noda at jsk.t.u-tokyo.ac.jp

with polynomial functions.

Hauser et al. [2] selected contact states using a sampling-based approach and was able to generate a variety of motions in some uneven terrains. In contrast, we use greedy algorithm. We have two reasons for this; One reason is that greedy algorithm is easy to use with evaluation functions. Our goal is to generate safe motions minimizing BRLV. Another reason is that complex contact states transitions in real world needs a lot of re-computations of next contact states. For example, impact in landing on ladders may change the contact state in comparison with planned one. Then, we will have to compute again the next contact state from current state.

II. Body Retention Load Vector and Overview of Steep Climbing Motion Planner

A. Body Retention Load Vector

To make up for the lack of power and safety, we focus on the physical constraints: limitations of joint Torque, contact Force, contact Moment (TFM limitations):

$$G\boldsymbol{F} + \begin{pmatrix} m\boldsymbol{g} \\ \boldsymbol{c} \times m\boldsymbol{g} \end{pmatrix} = \boldsymbol{0} \tag{1}$$

$$G = \begin{pmatrix} \cdots & E & 0 & \cdots \\ \cdots & \boldsymbol{x}_i \times & E & \cdots \end{pmatrix}, \quad \boldsymbol{F} = \begin{pmatrix} \vdots \\ \boldsymbol{f}_i \\ \boldsymbol{n}_i \\ \vdots \end{pmatrix}$$

$$-\boldsymbol{\tau}_{max} \leq \boldsymbol{\tau} = \Gamma(\boldsymbol{\theta}) - J^T\boldsymbol{F} \leq \boldsymbol{\tau}_{max} \tag{2}$$

$$-\left(M_i^- \boldsymbol{F}_i + \boldsymbol{F}_{i0}\right) \leq \boldsymbol{F}_i \leq \left(M_i^+ \boldsymbol{F}_i + \boldsymbol{F}_{i0}\right) \tag{3}$$

$$M_i^{\pm} = \begin{pmatrix} 0 & 0 & \mu_{xy} & 0 & 0 & 0 \\ 0 & 0 & \mu_{xy} & 0 & 0 & 0 \\ 0 & 0 & b_{\pm} & 0 & 0 & 0 \\ 0 & 0 & l_y & 0 & 0 & 0 \\ 0 & 0 & l_x & 0 & 0 & 0 \\ 0 & 0 & \mu_z & 0 & 0 & 0 \end{pmatrix}, \boldsymbol{F}_{i0} = \begin{pmatrix} f_{x0} \\ f_{y0} \\ f_{z0} \\ n_{x0} \\ n_{y0} \\ n_{z0} \end{pmatrix}$$

m is weight of robot, $\boldsymbol{c}$ is center of gravity, $\boldsymbol{g}$ is gravity vector, $\boldsymbol{x}_i$ is position of contact links, $\boldsymbol{f}_i, \boldsymbol{m}_i$ are contact forces and moments of each contact links, $\boldsymbol{\tau}$ is joint torques from the contact links, J is Jacobian matrix, μ is coefficient of friction, and $b_{\pm}$ is 1 or 0 for bilateral constraints.

In steep climbing, TFM limitations are very severe, and if this limitation is not satisfied, dangerous accident will occur. To evaluate the severity of TFM limitations, we introduce a new index, Body Retention Load Vector (BRLV):

$$\boldsymbol{BRLV}_i = \begin{bmatrix} \bar{\boldsymbol{\tau}}_i{}^T & \bar{\boldsymbol{f}}_i{}^T & \bar{\boldsymbol{n}}_i{}^T \end{bmatrix}^T \tag{4}$$

$$where, \begin{cases} \bar{\boldsymbol{\tau}}_i : = (\cdots \frac{\tau_{ij}}{\tau_{ij}^{max}} \cdots)^T \\ \bar{\boldsymbol{f}}_i : = (\cdots \frac{f_{ij}}{f_{ij}^{max}} \cdots)^T \\ \bar{\boldsymbol{n}}_i : = (\cdots \frac{n_{ij}}{n_{ij}^{max}} \cdots)^T \end{cases}$$

BRLV contains contact force ($\boldsymbol{f}_i$), contact moment ($\boldsymbol{n}_i$), and torque of joints from the contact link ($\boldsymbol{\tau}_i$). All of them are normalized with TFM limitations Eq.(2),Eq.(3). So, BRLV of any feasible robot state must satisfy the following constraints:

$$-\boldsymbol{1} \leq \boldsymbol{BRLV}_i \leq \boldsymbol{1} \tag{5}$$

The goal of this paper is to generate safe motions, that is, motions with minimized BRLV. In the following section, the algorithm and the result are shown.

B. Overview of steep climbing motion planner

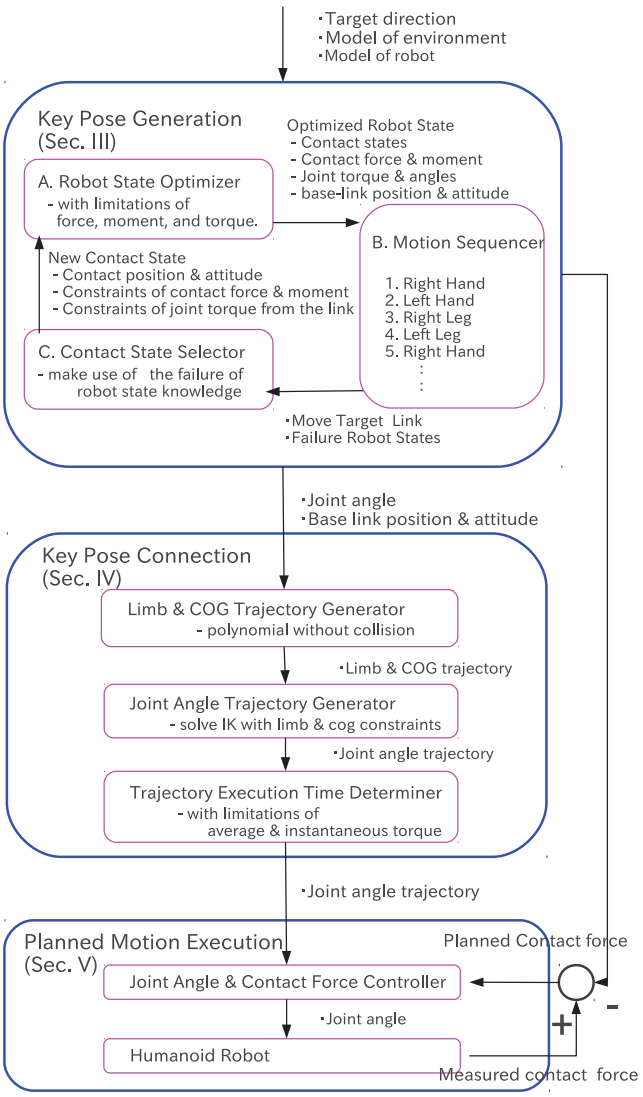

Fig. 2. Overview of steep climbing motion planner

Before explaining about our approach, We will show you a overview of our planner (Fig.2).

1) Require: Our planner requires target direction, models of environment, and model of robot. Target direction is a vector to which robot moves. Models of environment have an information about feasible set of contact states candidates. In this paper, we call contact state as a set of two kinds of constraints. A) Geometrical constraints: contact position and orientation. B) TFM limitations.

2) Output: Our planner outputs robot states sequence. Robot state is a set of contact states, joint angles, base link's position and orientation, and contact force and moment.

3) Procedure: Our planner makes up of 3 components:

- The first one is key pose generator (Section.III), which generates a set of static key robot states from the feasible set of contact states. All robot states will satisfy not only geometrical constraints (contact position, contact orientation, and no collision between the other models), but also TFM limitations.
- Second one is the key pose connector (Section.IV), which connects the static key robot states and generates trajectories of swing limbs and center of gravity avoiding collision, and determines the time of movement of the trajectory taking into accout two joint torque limitations: instantaneous and average torque limits.
- Third one is the planned motion executor (Section.V), which is the component to control humanoid robot as planned. The joint angles and contact forces are controlled.

III. Key pose generation with minimizing BRLV

Key pose generation involves three components:

- Robot state optimizer:
  Optimizes the BRLV norm of static robot state with given contact states.
- Motion sequencer:
  Determines the next target link and contact state using the following contact state selector, and generates robot states sequence using robot state optimizer.
- Contact state selector:
  Selects contact states of target link from the contact states candidates in the models of environment.

The following sub-sections, we will explain about each parts.

A. Robot state optimizer: *minimizes BRLV norm with all constraints*

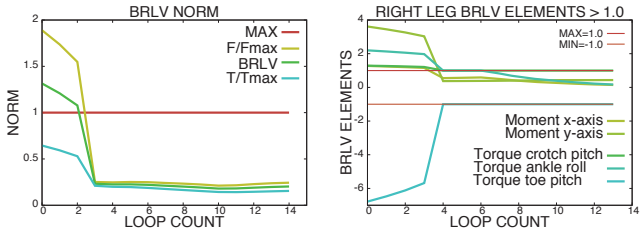

Fig. 3. BRLV norms in each iteration (left) and BRLV elements of right leg > 1.0 (right). $\gamma = 1.0$, $\mu_x = \mu_y = 0.3$, $\mu_z = 0.1$, $l_x = l_y = 5cm$, $\boldsymbol{F}_{i0} = (50\ 50\ 50\ 10\ 10\ 10)^T$ for hands, $\boldsymbol{\tau}_{max} = 0.4\times$ motor spec of HRP-2[6]

In this sub-section, we propose a numerical method to minimize the BRLV norm of static robot state satisfying given contact states (Algorithm.1). Our algorithm is an iterative method. It separates the constraints into physical and geometrical ones, then solves them and updates the center of gravity position of the robot to reduce BRLV.

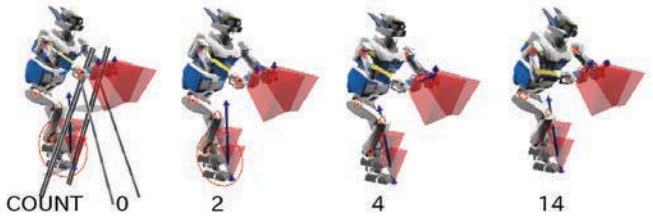

Fig. 4. Robot states in 0-th, 2-th, 4-th, and last iteration.
Red circles show joint torques, blue arrows show contact forces, and red polyhedrons show friction cones. 4-th one satisfies all constraints barely, and 14-th one also satisfies with more margin.

Algorithm 1 robot-state-optimizer

Require: $cs_i = (\boldsymbol{x}_i, R_i, l_i, TFM_i)$
$\boldsymbol{x}_i$: target contact position of i-th contacts.
R_i: target contact orientation of i-th contacts.
l_i: contact link.
TFM_i: TFM limitations of l_i.
Variable: .
$\boldsymbol{f}_i, \boldsymbol{n}_i$: force and moment of all contacts.
$\boldsymbol{\tau}_i$: torque of joints from l_i.
$\boldsymbol{x}_b$: base link position.
R_b: base link orientation.
$\boldsymbol{\theta}$: joints angles of humanoid robot.
$\boldsymbol{c}$: target center of gravity of humanoid robot.
Procedure: robot-state-optimizer
1: **repeat**
2: $[\boldsymbol{\theta}, \boldsymbol{x}_b, R_b] \leftarrow$ inverse_kinematics($\boldsymbol{c}, \boldsymbol{x}_i, R_i$)
3: $[\boldsymbol{f}_i, \boldsymbol{n}_i, \boldsymbol{\tau}_i] \leftarrow$ optimize_brlv($\boldsymbol{\theta}, \boldsymbol{x}_b, R_b, TFM_i$)
4: $\boldsymbol{c} \leftarrow$ calc_cog_to_reduce_brlv($\boldsymbol{f}_i, \boldsymbol{n}_i, \boldsymbol{\tau}_i$)
5: **until** loop_limit_exceeded **or**
($(\boldsymbol{f}_i, \boldsymbol{n}_i, \boldsymbol{\tau}_i) \in TFM_i$ **and** convergence)
6: **return** $[cs_i, \boldsymbol{\theta}, \boldsymbol{x}_b, R_b, \boldsymbol{f}_i, \boldsymbol{n}_i, \boldsymbol{\tau}_i]$

1) "inverse_kinematics": This function solves the geometrical constraints: contact position, contact orientation, ,position of center of gravity, and avoiding collision. If there is no solution to the inverse kinematics problem or the change in $\boldsymbol{c}$ is small enough, the loop terminates.

2) "optimize_brlv": This function optimizes a quadratic form of BRLV with TFM limitations: Eq.(1), Eq.(2), Eq.(3). Geometrical values such as $\boldsymbol{\theta}, \boldsymbol{x}_b, R_b$ are fixed. Because TFM limitations are composed of linear equations and non-equations, if the object function can be expressed as a quadratic form, QP (Quadratic Planning) algorithm such as goldfarb-idnani method can be applied. BRLV takes on a non-linear form with respect to force and moment, so, we use $M_i^{\pm}\boldsymbol{F}_i^{cst} + \boldsymbol{F}_{i0}$ as $\boldsymbol{F}_i^{max}$ in Eq.(4). $\boldsymbol{F}_i^{cst}$ is a constant vector. In the following experiments, we use $[0\ 0\ 50\ 0\ 0\ 0]^T$ for hands, and $[0\ 0\ 500\ 0\ 0\ 0]^T$ for legs.

3) "calc_cog_to_reduce_brlv": This function updates the center of gravity to reduce BRLV using the following equa-

tion:

$$\delta \boldsymbol{c} = \frac{1}{mg} \begin{pmatrix} \cdots & -z_i & 0 & x_i & 0 & -1 & 0 & \cdots \\ \cdots & 0 & -z_i & y_i & 1 & 0 & 0 & \cdots \\ \cdots & 0 & 0 & 0 & 0 & 0 & 0 & \cdots \end{pmatrix} \delta \boldsymbol{f} \tag{6}$$

Eq.(6) is obtained by isolating $\boldsymbol{c}$ in Eq.(1). Eq.(6) utilizes the moment arm of center of gravity and control the contact force and moment. To use BRLV in Eq.(6) and to reduce BRLV norm, we transform BRLV as follow:

$$\boldsymbol{F}_i^{BRLV} = \begin{pmatrix} \gamma (J^T)^{-1} & E \end{pmatrix} \boldsymbol{BRLV}_i \tag{7}$$

E is 6×6 unit matrix, and γ is the gain. By using $-\boldsymbol{f}^{BRLV}$ as $\delta \boldsymbol{f}$ in Eq.(6), we can obtain a change in $\boldsymbol{c}$ for reducing $BRLV$.

4) Application to balancing on a ladder: We applied $robot\_state\_optimizer$ to the task of balancing on a ladder with 3 contact links: both hands and right leg.

Fig.3 shows the BRLV norms (left) and BRLV elements over 1.0 (right) in each iteration. With just 4 iterations, all the constraints were satisfied. All of 14 iterations took 2.7 seconds on Intel Core i7-3520M CPU 2.90GHz. However, we implements our algorithm using one of the dialects of lisp interpreter: euslisp [11]. So, we believe that there are a lot of room for acceleration in computing.

Fig.4 shows the robot states in each iteration. With the first posture, the humanoid robot's center of gravity is far from the support leg, but with the last posture, it is near and has a low-load.

B. Motion sequencer: generates robot states sequence

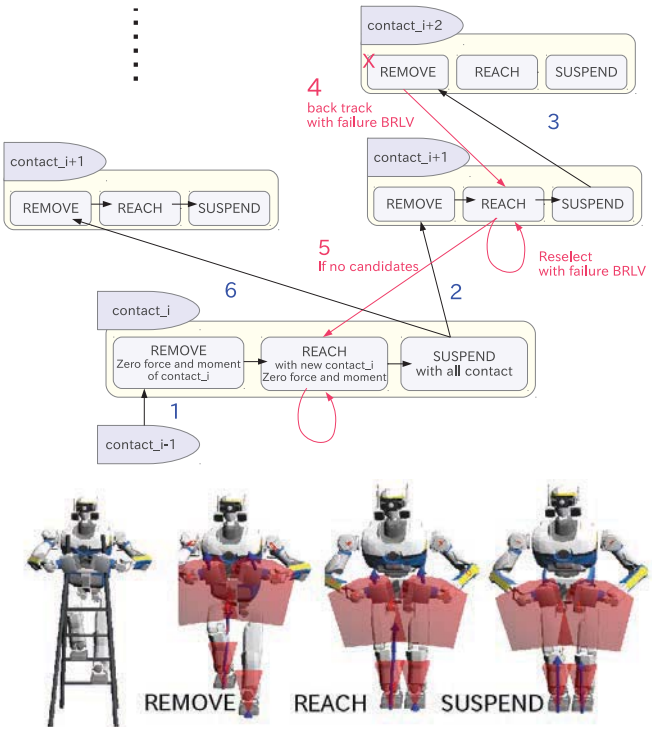

Fig. 5. Top: Procedure of motion sequencer. Bottom: $remove$, $reach$, $suspend$ robot states with targeting right leg as next contact link.

Motion sequencer generates robot states for achieving contact states transition. Contact state transition has three steps as follows:

1) $remove$
   Remove contact of target contact link. And call $robot\_state\_optimizer$ in Section.III-A. If $robot\_state\_optimizer$ returns fail, backtrack with failure BRLV to the previous $reach$ step.
2) $reach$
   Select the next contact state of target contact link using $contact\_state\_selector$ in Section.III-C. When calling $robot\_state\_optimizer$ with new contact state, target contact link is not able to exert any force and moment. If this optimization returns fail, call $contact\_state\_selector$ with failure BRLV again while optimization fails and there are still contact state candidates remainings.
3) $suspend$
   Call $robot\_state\_optimizer$ with all contact states. This step always has one answer because the robot state in $reach$ is also the answer of this step.

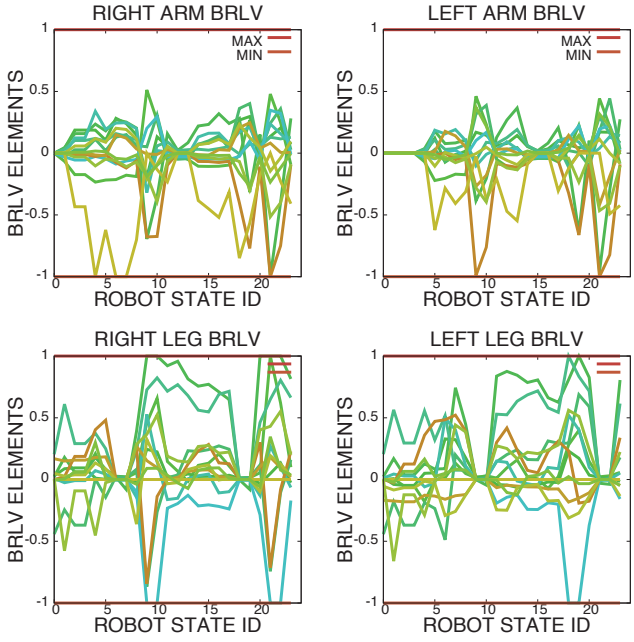

Fig. 6. Graphs of all contact link's BRLVs

We applied this motion sequencer to a ladder climbing motion. Contact link candidates are both arms and both legs. Contact state candidates are given manually in reference to a human motions. The results are shown in Fig.7. All BRLV elements of contact links (Fig.6) satisfy Eq.(5). Motion sequencer generates 24 robot states to climb 2 steps, and it took 52.4 sec with no backtrack.

C. Contact state selector: *selects contact state to reduce BRLV*

Because contact states determine all constraints without collision, selection of beneficial contact state for reducing BRLV is a very important task. We use greedy algorithm to

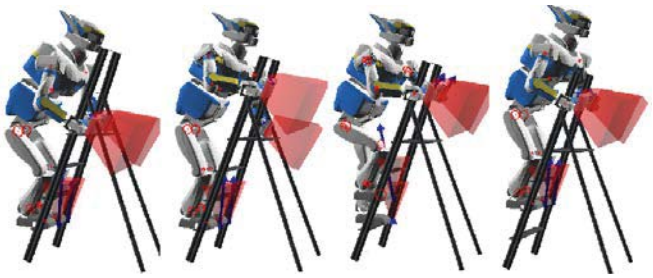

Fig. 7. Generated climbing key poses

select contact states. And we also use backtrack methods to reselect contact states and to reduce BRLV in the future. We propose three simple indices to determine the next contact state as follows:

$$\begin{aligned} maximize \quad &: \quad L = a + \alpha b + \beta c \qquad (8) \\ a \quad &= \quad \begin{pmatrix} \boldsymbol{n} \\ \boldsymbol{x}_{new} \times \boldsymbol{n} \end{pmatrix}^T G \boldsymbol{F}^{BRLV} \\ b \quad &= \quad \boldsymbol{d}^T (\boldsymbol{x}_{new} - \boldsymbol{x}_{now}) \\ c \quad &= \quad -\|\boldsymbol{x}_{new} - \boldsymbol{x}_{now}\| \end{aligned}$$

Where, $\boldsymbol{x}_{new}$ is a contact candidate position, $\boldsymbol{x}_{now}$ is the current position of the contact link, $\boldsymbol{n}$ is the normal vector of the contact plane, G is the grasp matrix of all contact points in Eq.(1), $\boldsymbol{d}$ is the target direction to move, and α, β are positive gains. a is the indicator for selecting a surface that can exert the largest force and moment in the direction of BRLV. b is the indicator for moving toward the target direction $\boldsymbol{d}$. c is the indicator to select reachable point. The contact state which maximizes L is the next contact state.

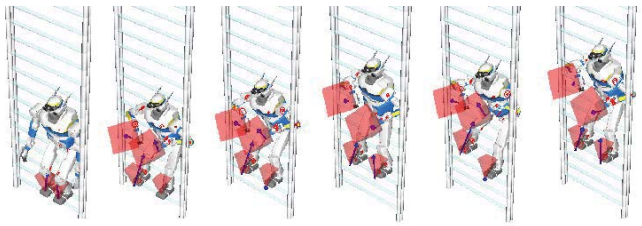

Fig. 8. Vertical ladder climbing motion with contact state selector $\alpha = 1.4,\ \beta = 1.0,\ \gamma = 1.0,\ \mu_x = \mu_y = 0.3,\ \mu z = 0.1,\ l_x = l_y = 5cm,\ \boldsymbol{F}_{i0} = (50\ 50\ 50\ 10\ 10\ 10)^T$ for hands, $\boldsymbol{\tau}_{max} = 0.7\times$ motor spec of HRP-2[6]

We applied this contact state selector and motion sequencer in Section.III-B to a vertical ladder climbing motion (Fig.8). To climb up 5 steps, motion sequencer generates 60 robot states in 77.6 sec. We gave 3 contact state candidates on each foothold, these normal vectors of contact planes are shown in Fig.9: "a" is vertical, "b" tilts 15 degrees towards robots, and "c" tilts 15 degrees towards the other side of "b". In balancing on a vertical ladder, it is impossible to place the center of gravity on any contact positions, and robot needs a lot of moment to balance. At first, our contact state

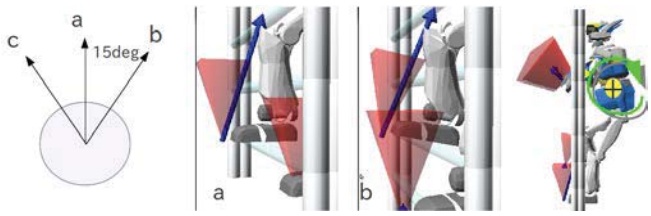

Fig. 9. Balancing on a vertical ladder needs a lot of moment, and tilt contact states are needed

selector selects "a" because a standing posture needs a lot of vertical force due to gravity and torque limitations. But, after backtracking, the selector reselects "b" to cancel out the moment, and motion sequencer successfully generates these climbing motions.

We also applied the contact state selector to climbing of a steeper surface: rock wall climbing. The rock wall we used was composed of 150 cubes which have random sizes, positions, and orientations. All faces of these cubes are the contact state candidates. Our planner successfully generated 160 robot states with 30 back tracks to climb up 1.5 meters, and the whole computation took 456.0 seconds.

Fig. 10. Rock wall climbing motion with contact state selector $\alpha = 1.5,\ \beta = 10.0,\ \gamma = 1.0,\ \mu_x = \mu_y = \mu z = 1.0,\ l_x = l_y = 10cm,\ \boldsymbol{F}_{i0} = \boldsymbol{0}, \boldsymbol{\tau}_{max} = 0.7\times$ motor spec of HRP-2[6]

IV. Key Pose Connection with Two Joint Torque Limitations

A. Generation of collision-free trajectories

Fig. 11. Trajectory without collision

Generating trajectory of swing limbs or center of gravity using polynomial functions is a well-known strategy for walking [12] and stair climbing [13]. We used polynomial functions for generating trajectories of swing links and base link. Then, using the base link trajectory as the initial states and using the swing links trajectories as the target positions, we solve the inverse kinematics problem and generate the trajectory for whole-body joints angles.

Each trajectory has some constraints, for example it has to be collision-free. Swing link constraints are follows:

1) Going through 2 points without collision
2) Zero velocities and accelerations at start/end points
3) Maintain positions and orientations of start/end points
4) 5cm away from the contact surface at end point

Fourth constraint is an important margin for safety with real robots, since there will be some differences between the planned motion and the actual motion due to modeling errors or slight changes of body shape. Then, the humanoid robot may run into the step surface before landing on the target destination.

Sampling collision free points are achieved through 3 steps: First, draw line between start and end points of swing links. Second, split the line into 3 line segments. And last, if any one of these 2 points has collision with the other models, move it to the direction of normal vector of collision plane. This is very simple algorithm and some other algorithms may be needed in more complex situations, but, in the following experiments, this was sufficient.

The constraints of base link trajectory are follows:

1) Zero velocities and accelerations at start/end points
2) Maintain positions and orientations of start/end points

B. Determination of the time of movement on trajectory

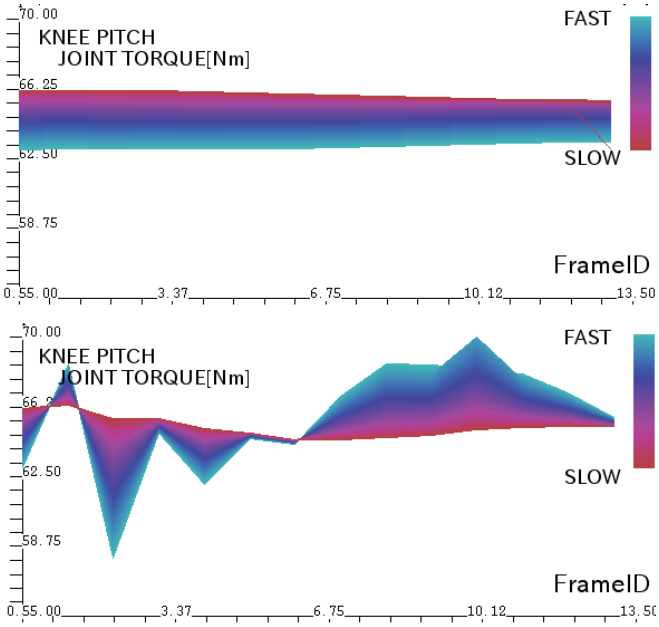

Fig. 12. Average and instantaneous torque of the knee pitch joints while climbing a ladder

We determine the time of movement using two torque limitations: limitations for 3-second average joint torque and instantaneous one. Average limitation is 0.7 times smaller than the instantaneous one. In experiments with real humanoid robots, there will be some noise, and some margins of torque limitations are essential. But for high load motion such as steep climbing, instantaneous high joint torque is needed. To satisfy these two conflicting requirements, we use 3 seconds average joint torque limitation in addition to the instantaneous one. Average and instantaneous torque limitations have the opposite nature with regards to time of movement.

Fig.12 are graphs of joint torque calculated using equation of motion while climbing a ladder in simulation. The top one is average torque, and the bottom one is instantaneous torque. The faster the robot moves, the higher the instantaneous joint torques are. On the contrary, The slower robot moves, the higher the average joint torques are. Gradually changing the time spent on moving and estimating the joint's torque, it is possible to determine the time to satisfy both constraints.

V. Climbing Experiment in Real World

To evaluate our planner, we performed three climbing experiments with a life-sized humanoid robot, HRP-2.

A. Climbing a ladder

Fig. 13. Control joint angles and contact force for climbing a ladder

1) Experimental setup and result: We applied our planner to a ladder climbing motion (Fig.13). Each step height is 20 cm and the slope is 70 deg. Planning parameters are $\gamma = 1.0$, $\mu_x = \mu_y = 0.3$, $\mu_z = 0.1$, $l_x = l_y = 5cm$, $\boldsymbol{F}_{i0} = (50\ 50\ 50\ 10\ 10\ 10)^T$ for hands, $\boldsymbol{\tau}_{max} = 0.4\times$ motor spec of HRP-2[6], and the duration of the trajectory is 2.1 sec

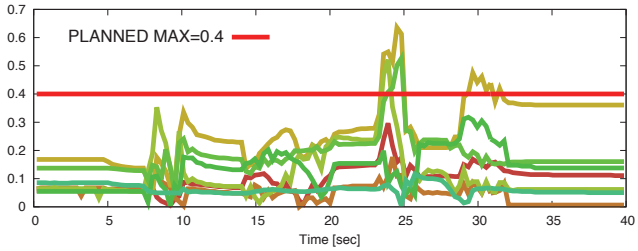

Fig. 14. Each joints torque of support leg and instantaneous high load with climbing up motion

which is determined with some trial-and-error to satisfy the two joint torque limitations. There is instantaneous high load during climbing (Fig.14). A trajectory duration of 1.5 sec was not feasible because of instantaneous limitation, and 3.0 sec was also no feasible because of the average one.

In the end, we succeeded in climbing up and down two steps. In addition, with simple balance control using a force sensor on HRP-2's hands, and with controlling the right hand using a 3D mouse, we succeeded in grasping a bag located

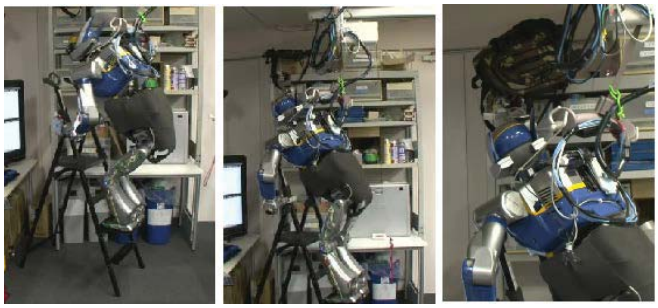

Fig. 15. Climbing up 2 steps and reaching on ladder

at the highest point of the shelf (Fig.15). For balancing, we used impedance control, where the target reaction force was decided to reduce the momentum and angular momentum of the humanoid robot.

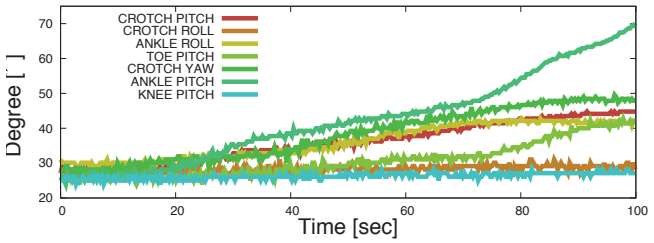

Fig. 16. Each joint temperature of left leg

Because each joint torque will be high in these experiments, we also measured motor temperatures of each joint surface for safety (Fig.16). We confirmed that the ankle pitch joint had the highest temperature, and it increased 40 degrees in 120 sec.

B. Getting into a car

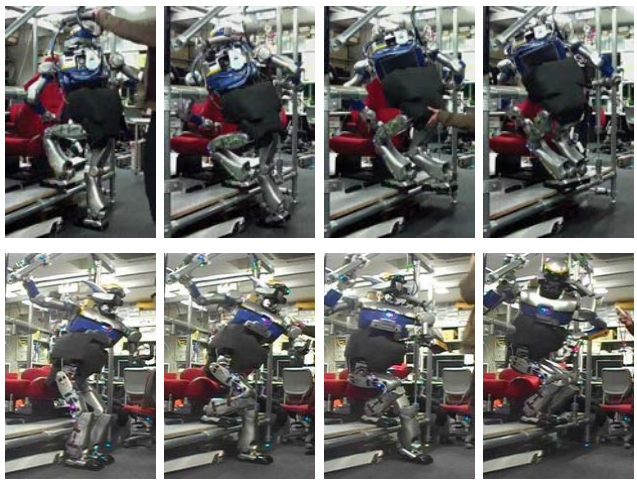

Fig. 17. Getting into a car with 2 initial poses: forward and backward

Fig.17 are the images of the getting into a car experiments. The car has a step height of 30cm. We succeed in climbing with two initial postures, forward and backward.

VI. Conclusion

We have proposed a novel whole-body motion planner for generating safe motions that are as far away from limitations of joint Torque, contact Force, and contact Moment (TFM limitation) as possible. We introduced Body Retention Load Vector (BRLV) and formulated humanoid robot's motion planning as an optimization procedure considering both TFM limitations and minimizing of BRLV. In order to show the effectiveness of our approach, we had experiments both in simulation and real world. In simulation, we applied our planner to climbing motions of ladders and a rock wall. And in real world, we demonstrated the motions of climbing a ladder and getting into a car with real a life-sized humanoid robot HRP-2.

References

[1] K. Harada, K. Hauser, T. Bretl, and J.-C. Latombe. Natural motion generation for humanoid robots. In *International Conference on Intelligent Robots and Systems, 2006 IEEE/RSJ*, pp. 833 –839, Oct 2006.

[2] Hauser. Kris, Bretl. Tim, and Latombe. Jean-Claude. Non-Gaited Humanoid Locomotion Planning. *International Conference on Humanoid Robots, 2005 5th IEEE-RAS*, pp. 7–12, 2005.

[3] K. Bouyarmane, J. Vaillant, F. Keith, and A. Kheddar. Exploring humanoid robot locomotion capabilities in virtual disaster response scenarios. *2012 IEEE-RAS Int. Conf. on Humanoid Robots*, Vol. 26, No. 10, pp. 1099 – 1126, Dec 2012.

[4] Jaeheung Park and O. Khatib. Contact consistent control framework for humanoid robots. In *International Conference on Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE*, pp. 1963–1969, May.

[5] A. Escande, A. Kheddar, and S. Miossec. Planning support contact-points for humanoid robots and experiments on hrp-2. In *International Conference on Intelligent Robots and Systems, 2006 IEEE/RSJ*, pp. 2974–2979, Oct.

[6] Kazuhiko AKACHI. Takakatsu ISOZUMI. Masaru HIRATA. Sigehiko OHTA. Masakazu ISHIZAKI. Development of the humanoid robot, hrp-2 (in japanese). *Kawada technical report*, Vol. 23, pp. 20–25, 2004.

[7] H. Yoneda, Micro-Nano Syst., K. Sekiyama, Y. Hasegawa, and T. Fukuda. Vertical ladder climbing motion with posture control for multi-locomotion robot. *IROS 2008*, pp. 3579–3584, 2008.

[8] S. Osswald, J.-S. Gutmann, A. Hornung, and M. Bennewitz. From 3d point clouds to climbing stairs: A comparison of plane segmentation approaches for humanoids. In *International Conference on Humanoid Robots (Humanoids), 2011 11th IEEE-RAS*, pp. 93–98, Oct.

[9] MUSCLE CORPORATION. YUME ROBO . `http://www.musclecorp.com/english/event.html`. International robot exhibition 2011.

[10] S. Fujii, K. Inoue, T. Takubo, Y. Mae, and T. Arai. Ladder climbing control for limb mechanism robot x201c;asterisk x201d;. In *International Conference on Robotics and Automation, 2008. ICRA 2008. IEEE*, pp. 3052 –3057, May 2008.

[11] Toshihiro Matsui and Masayuki Inaba. Euslisp: an object-based implementation of lisp. *Journal of Information Processing*, Vol. 13, No. 3, pp. 327–338, 1990.

[12] M. Morisawa, K. Harada, S. Kajita, S. Nakaoka, K. Fujiwara, F. Kanehiro, K. Kaneko, and H. Hirukawa. Experimentation of humanoid walking allowing immediate modification of foot place based on analytical solution. In *Robotics and Automation, 2007 IEEE International Conference on*, pp. 3989–3994, 2007.

[13] Chenglong Fu and Ken Chen. Gait synthesis and sensory control of stair climbing for a humanoid robot. *Transactions on Industrial Electronics, IEEE*, Vol. 55, No. 5, pp. 2111 –2120, may 2008.