Computation of trajectory of center of mass of the body of a quadruped for performing simultaneous body and foot motions:

Hypothesis:

The foothold planner returns a sequence of footholds along with transition leg index:

$$\{F_k, f_k\}$$

where F_k is essentially a length 4 array of foothold coordinates, and f_k is the index of foot that is changed from F_k to F_{k+1}. $F_k(f)$ gives the coordinate of the foot f in the k^{th} step in the sequence.

From the above sequence it is easy to compute the sequence of triangles of support $\{T_k\}$, where T_k is the triangle of support to be used during the transition from F_k to F_{k+1}. T_k may be described by a length 3 array of coordinates describing the triangle, and a length 3 array of foot indices corresponding to each vertex of the triangle of support.

The function for computing the position of center at 'critical' points:

By 'critical' points we mean the instants at which transitions take place from one triangle of support to another. In this section we'll assume that there is no error, and we compute the position of the center of mass of the body as a function of the triangles of support.

The formulation:

During the transition from F_k to F_{k+1} (i.e. while T_k is the triangle of support), let us assume that the center needs to move from point P_k^s to P_k^e (here the superscript 's' and 'e' stand for 'start' and 'end' respectively). The problem hence reduces to finding P_k^s and P_k^e for any arbitrary k.
We will formulate a recursive functional form for P_k^s and P_k^e as follows:

$$\{P_k^s, P_k^e\} = \Phi(T_{k-1}, T_k, T_{k+1}, T_{k+2}, P_{k-1}^e)$$

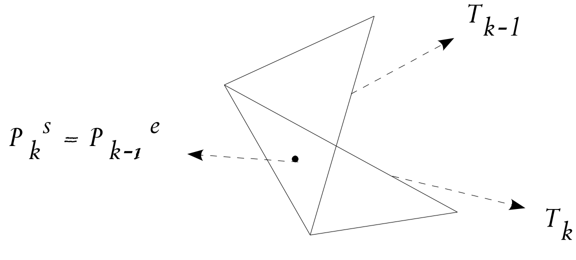
Computing P_k^s :

Case I – The triangles T_{k-1} and T_k overlap:

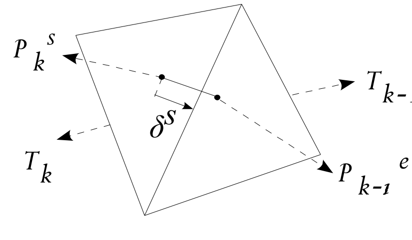
Set $P_k^s = P_{k-1}^e$.

Case II – The triangles T_{k-1} and T_k do not overlap:

P_k^s is a point on the other side of the common edge between T_{k-1} and T_k , and 'just opposite to' P_{k-1}^e at a distance of δ^s from the edge.



Case I



Case II

Computing P_k^e *:*

Case I – The triangles T_k and T_{k+1} overlap, but T_{k+1} and T_{k+2} do not overlap:

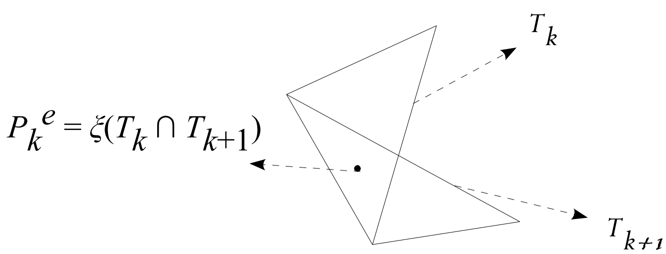
P_k^e is a point inside the common region between T_k and T_{k+1} , *i.e.*,

$$P_k^e = \xi(T_k \cap T_{k+1})$$

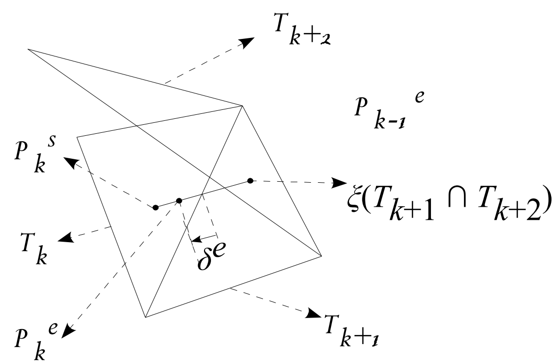
where ξ returns a point inside the connected and convex set which is passed to it as an argument. This may be the incenter, centroid or any other point chosen according to convenience.

Case II – The triangles T_k and T_{k+1} do not overlap, but T_{k+1} and T_{k+2} overlap:

P_k^s (computed in previous step) is connected with $\xi(T_{k+1} \cap T_{k+2})$ by a straight line. P_k^e is then chosen to be a point on this line at a distance of δ^e from the common edge between T_k and T_{k+1}, and inside T_k .



Case I



Case II

Execution:

There are two distinct phases in the execution depending on the value of P_k^s returned by Φ. Phase 0 is the one when only the body is being moved with all the 4 feet on the ground, whereas Phase 1 is the one when one leg is in flight as well as the body is moving.
Below is the pseudo-algorithm for the execution. We assume that the series $\{T_k\}$ is available for $k = 1$ to N.

Set T_0 such that T_0 and T_1 have an overlap
P_0^e be the initial center of the body, which can be assumed to be the centroid of initial quadrilateral of support.

$k = 1$;
count = 0;
PhaseZeroCompleted = FALSE;
While ($k \leq N$-2)
    $\{P_k^s, P_k^e\} = \Phi(T_{k-1}, T_k, T_{k+1}, T_{k+2}, P_{k-1}^e)$;
    if (($P_k^s \sim= P_{k-1}^e$) AND (~*PhaseZeroCompleted*)) OR (*count* == 0)
        Move body from P_{k-1}^e to P_k^s with all 4 feet touching the ground (Note: After the last flight of leg the 4 feet will be touching the ground);
        PhaseZeroCompleted = TRUE;
    else
        Move body from P_k^s to P_k^e while the foot f_k is moved from $F_k(f_k)$ to $F_{k+1}(f_k)$;
        PhaseZeroCompleted = FALSE;
        $k \leftarrow k + 1$;
    end
    count $\leftarrow$ *count* + 1;
end

Accounting for errors in the system:

When there are errors in the system the computation of $\{P_k^s, P_k^e\}$ using Φ may require some slight modifications to make it robust. We need to be able to detect common edges between T_k and T_{k+1} even if they don't exactly have one because of errors. Moreover we should be able to identify the common region between T_k and T_{k+1} and operate ξ on it accordingly.

The execution algorithm also changes slightly. Now, we read the actual position of the center and the footholds and start from there to perform any segment of the motion. The modified algorithm is something as follows (**bold** marks the changes):

Set T_0 such that T_0 and T_1 have an overlap
P_0^e be the initial center of the body, which can be assumed to be the centroid of initial quadrilateral of support.

$k = 1$;
$count = 0$;
PhaseZeroCompleted = FALSE;
While ($k \leq N$-2)
    $\{P_k^s, P_k^e\} = \Phi(T_{k-1}, T_k, T_{k+1}, T_{k+2}, P_{k-1}^e)$;
    if (($P_k^s \sim= P_{k-1}^e$) AND (~*PhaseZeroCompleted*)) OR ($count == 0$)
        Move body from **current position of the center** to P_k^s with all 4 feet touching the ground **at the current footholds**;
        PhaseZeroCompleted = TRUE;
    else
        Move body from **current position of the center** to P_k^e while the foot f_k is moved from **Current Foothold of f_k** to $F_{k+1}(f_k)$;
        PhaseZeroCompleted = FALSE;
        $k \leftarrow k + 1$;
    end
    $count \leftarrow count + 1$;
end

Subhrajit Bhattacharya
Department of Mechanical Engineering & Applied Mechanics,
University of Pennsylvania.
Office: GRASP Laboratory (Levine 403)
email: subhrabh@seas.upenn.edu