

# SYSTEMATIC AND COMPLETE ENUMERATION OF STATICALLY STABLE MULTIPOD GAITS

Submitted: 10<sup>th</sup> October 2018; accepted: 20<sup>th</sup> December 2018

Jörg Roth

DOI: 10.14313/JAMRIS\_4-2018/24

## Abstract:

*Insect-like robots have many advantages concerning mobility and stability. The specific sequence of legs going through different phases, the gait, is important when planning and executing a complex motion. The notion of gaits was originally introduced by biologists but gaits also influenced robot development. Typical multipod robots are able to execute much more gaits than occur in wildlife. In this paper we present a formalism to express certain rules for reasonable gaits. We show an algorithm that enumerates all statically stable gaits according to our formalism. We then provide a gait classification by the example of six-legged robots. Finally, we introduce properties to evaluate gaits.*

**Keywords:** Legged Robot, Multipod, Hexapod, Statically Stable Gait

## 1. Introduction

Natural arthropods such as insects or spiders are able to quickly and stably walk over rough terrain. The respective multipod robots have many advantages over wheeled robots or humanoid, biped robots. On the one hand, they do not have to rely on drivable ground and can go over small obstacles; on the other hand, they are able to walk *statically stable*. This means, we can stop a motion at any time and the robot keeps its upright position. In contrast, dynamically stable motion requires mechanisms to actively maintain the balance.

Apart from the leg's geometry and moving trajectory, the *gait* (i.e. the time sequence of moving legs) significantly influences the overall movement characteristic. It defines the timing pattern of lifted and ground legs and affects properties such as the stability and speed. In this paper, we abstract from the actual robot geometry and define general rules that reasonable gaits must fulfill. These can be summarized to: fix clock for phase changes, uniformity and stability. We discovered that the set of gaits which fulfill our rules is a *countably infinite* set. Moreover, for a certain number of legs and a limited phase length, the set of gaits is *finite*. We present an algorithm that finds all gaits for certain parameters.

For the example of six legs we then classify gaits. Our classification includes well-established gaits such as *Tripod*, *Ripple* and *Wave*, but we also found a new

one, the *Split* gait. We can assign certain properties to gait classes such as the amount of support, propulsion and smoothness. We end with a brief discussion about odd numbers of legs.

## 2. Related Work

Early research about multipod gaits was conducted in the area of zoology, in particular, about gaits of insects with six legs. Insects are able to stabilize their gait with adhesion effects, thus can execute gaits that are not possible for statically stable robots. E.g. the *Tetrapod* gait [3, 22] has four legs on the ground, but their ground polygon (called the *support area*) does not always cover the center of gravity. If we look at statically stable gaits, most hexapod insects use the *Tripod* gait. As the actual number of gait variations is low, research often is focused on timing questions.

Research also identified dynamically stable fast gaits, e.g. for escape situations [15, 18]. These were only stable if the multipod is able to balance. We could e.g., classify a *Gallop* gait for hexapods.

Spiders (officially *no* insects) and crabs have eight legs, thus allow a larger amount of gait variations. However, looking at real animals, the gaits only seem to have a large variation in timing and rhythm, not in the actual sequence pattern. Some papers measured the gait timing, energy and support area for different spiders [2, 19]. A formal classification is difficult. In [8] the authors thus defined a *Random* gait that occurs, when a regular pattern is not obvious.

Even though more legs would allow more gaits (from the mathematical view), animals with more than eight legs such as decapods (e.g. crustacean) or centipedes tend to use a single pattern where legs are lifted one after the other, the *Metachronal* gait [4].

A more system-oriented, bio-inspired view on gaits provided [3, 5, 12]. They model gait execution by a network of neurons between legs that trigger leg movement dependent on former leg actions. This view is close to gait execution of real insects. E.g., the *Metachronal* gait of centipedes can be modeled without a central controlling instance that is aware of each leg.

An animal that is able to switch between a larger number of gaits is the horse, in particular in the area of dressage. Here we find a strong classification of different gaits, e.g. the *Gallop* or *Amble* gait. Most of them are not statically stable. [14] provided a formal defi-

dition for *Walk*, *Tolt*, *Trot* and *Pace* gaits and analyzed their timing.

A discussion of gaits and their properties for legged robots has a long tradition [17]. As a fundamental structure: each leg periodically goes through two phases. During the *stance* phase (also called *support* phase), the foot remains on the ground and carries the robot. During the *swing* phase (also called *transfer* phase, or '*in flight*'), the foot is moved in the robot's movement direction. The time of one stance and one swing phase is called the *cycle* time. The relation of stance to cycle time is called the *duty factor*. Originally, the duty factor is defined per leg and may be different. So-called *regular* gaits have the same value for each leg.

[16] introduced further properties for gaits: first they distinguish *periodic* from *non-periodic* gaits. Periodic gaits are additionally divided into *singular* gaits (placing one leg and lifting the next occur at the same time) and *symmetric* (right and left legs alternate their phase changes). They also identified certain stability measures (*stability margin*, *longitudinal stability margin*) and provided criteria for periodic, regular gaits, foremost the *Wave* gait.

Most natural multipods have an even number of legs. One exception is the starfish. [9] analyzed their gaits and possible application for robots.

Further papers focus on hexapods. [10] tried to classify hexapod gaits, but mainly discovered the *Tripod* and variations of *Wave* gaits. Trajectory geometries and gait rhythms are mixed. E.g., a class *Forward Wave* gait is identified that is applied for forward movement only. Also [20] mixed geometry and rhythm and identified a *sideways Tripod* gait as *Mammal* gait. [5] analyzed *Ripple* gaits for hexapods and formalized these gaits with nonlinear oscillators. [7] also studied some hexapod gait variations for robots and identified new gaits, but without a systematic way to enumerate all new variations.

[6] considered a gait as a function that maps a cycle (represented as unit circle  $-\pi \dots \pi$ ) to leg's configurations. This view enabled the notion of *gait transition*, i.e. patterns that connect two gaits. The stance movement affects the robot's trajectory, thus must not be changed between gaits. The approach thus used the legs in swing phase to change to a new gait. Swings can be slower or faster without affecting the movement direction.

Besides classifying gaits and their properties, further research tried to generate gaits. [11] presented a genetic algorithm to generate *Wave* gaits. [21] described a constraint-based approach to generate a gait for a specific motion task based on rules that describe kinematics, leg collisions, terrain and stability. [1] presented a machine learning approach based on an evolutionary algorithm.

### 3. The Gait Model

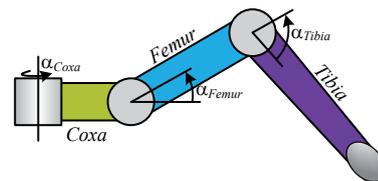
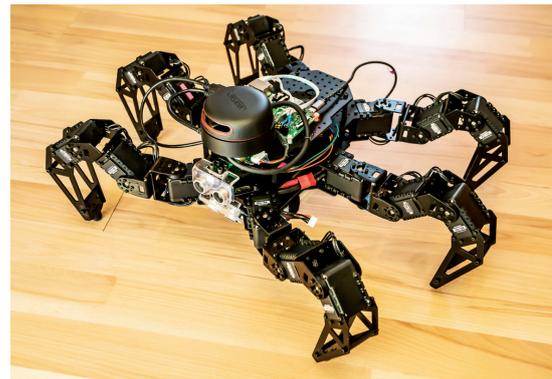
In this section we introduce a formal model to describe gaits. The gait formalism is two-fold: a geometric facet describes leg geometries and trajectories. A rhythmic pattern facet describes the sequence of

legs going through phases. We strongly believe both facets are independent. As a result, we can execute any walking trajectory with any gait rhythm.

The formalism is required to precisely describe a certain gait with parameters and to formulate conditions for reasonable gaits. We start with some considerations according to the geometric facet. We then abstract from the specific multipod geometry to introduce certain requirements concerning the leg phases. A further section examines static stability.

#### 3.1. Kinematics and Gait Patterns

Mobile multipod robots usually have an even number of legs with identical geometry. A common model is the hexapod, such as the *Bugbot* ([13], Fig. 1 top). Multipod legs must have at least 3 degrees of freedom to freely place and move the foot during gait execution. The leg segments usually are called *Coxa*, *Femur* and *Tibia* based on insect anatomy naming (Fig. 1 bottom). Robot legs with more degrees may provide redundancy in leg positioning, but are not generally capable to execute more gaits. In this paper, we abstract from inverse kinematics questions and assume, the controlling mechanism is capable to place the feet as required by a movement.



**Fig. 1. Hexapod robot (top), typical leg geometry (bottom)**

Multipods can walk in different ways. First, we can distinguish the actual trajectory, e.g. straight forward, sideways (i.e. crab gait), arc or turn in place. Second, we can distinguish the change of legs that are on the ground in *stance phase* or swing in moving direction. We call the time sequence of changing phase the *gait pattern*, or simply, *the gait*.

Figure 2 shows the two phases for a specific leg. The stance phase can be described by a movement among the *stance vector*  $\vec{v}_i$  in local robot coordinates. In world coordinates, the foot remains on the ground at the same position (in the absence of slippage). In the swing phase, the leg is lifted and moved in walking direction. In local coordinates, the two phases de-

scribe a round trip, i.e. the corresponding vectors can be connected to a polygon.

The set of  $\vec{v}_i$  specifies the multipod's trajectory. Fig. 2 (bottom right) shows exemplary how these vectors cause an arc movement. For arcs, the  $\vec{v}_i$  must reside on arc tangents and their lengths must be the same multiple of the distance to the arc center.

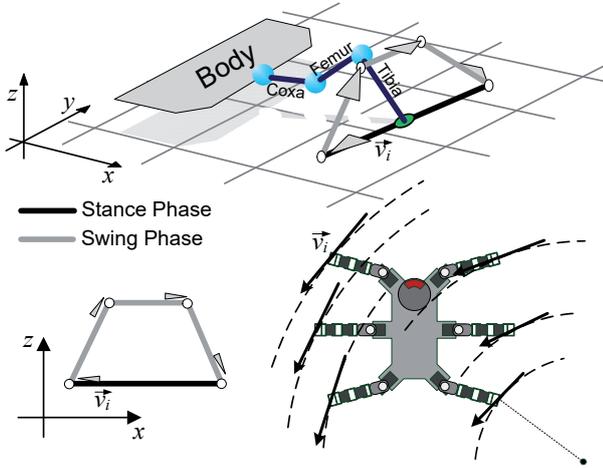


Fig. 2. Structure of a multipod gait (top and left),  $\vec{v}_i$  and arc trajectory (bottom right)

The gait pattern defines the cooperation of legs in the respective phases. Fig. 3 shows the example of the *Ripple* gait for hexapods. For this gait, always four legs are in the stance phase, whereas a swing phase starts in the middle of another leg's swing phase. Many further gaits are known. They differ in the amount of legs in stance phase, the amount of time in stance phase and the times of phase changes (see section 4.2).

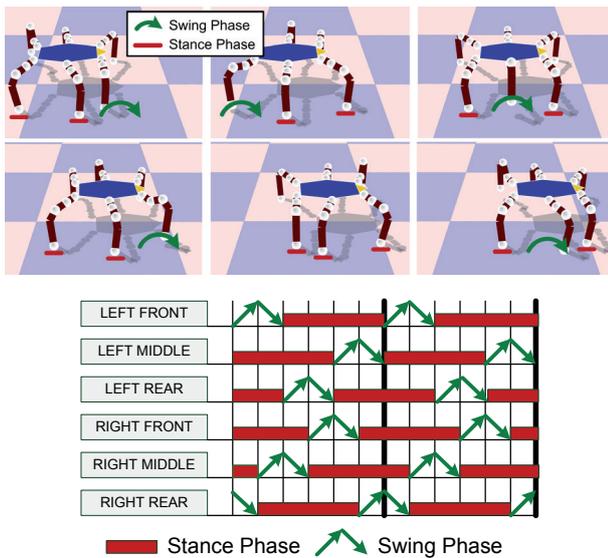


Fig. 3. Gait pattern for the *Ripple* gait of hexapods

An important observation: the trajectory definition based on the leg's  $\vec{v}_i$  and the gait patterns are independent. This means, we can walk in any direction (even sideways) with any gait pattern. In reverse this means, the gait pattern mainly defines movement *qualities* such as stability, propulsion or smoothness.

### 3.2. The Formal Gait Model and Gait Properties

Let  $\ell \geq 6$  denote the number of legs – we first assume an even number. We further assume, all time intervals are multiples of a fixed time interval  $t$ . In particular, we have a fix clock for phase changes. This may be considered as limitation, however, it reflects the typical mechanism for motion control: In a loop with constant iteration time, the controller computes new motion commands that simultaneously are sent to all legs (e.g. its servos). The legs then independently move until the next loop iteration computes new commands.

Gaits periodically repeat a leg pattern. We call the time before the same leg configuration occurs the *cycle*. The cycle time contains a single stance and a single swing phase. As a basic gait definition we specify the multiples of  $t$  for stance phase  $s$ , swing phase  $w$  and cycle time  $c$  with  $c = s + w$ .

The phases between legs may be interleaved, i.e. a swing phase may start in the middle of another leg's swing phase. In this case, the swing phase must last multiple steps  $t$  (Fig. 4). This also affects the granularity of the swing phase shape. E.g., with  $w = 3$ , we can define the swing phase more detailed compared to  $w = 2$ . The case  $w = 1$  executes a swing phase in a single step. We require this case later.

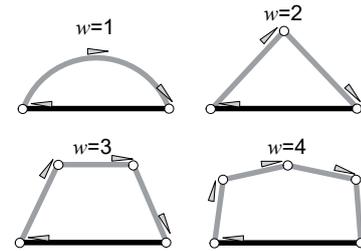


Fig. 4. Swing phases with different phase lengths

Even though the number of swing steps affects the geometric definition of the swing phase, the benefit of more steps should not be overestimated. Typical motion systems inherently smooth the paced polygon due to controlling effects. From the geometric view, it usually is not required to go beyond  $w = 4$ .

Besides the phase lengths, the timing when a specific leg changes its phase is important. Let  $w_i \in \{0, \dots, c - 1\}$  for  $i \in \{1, \dots, \ell\}$  be the step number when leg  $i$  enters the swing phase. As the length of phases is equal for all legs, these numbers fully describe the change of all phases. If we shift all  $w_i$  by the same offset, we get a different assignment, but actually the same gait. We thus assume  $w_1 = 0$ .

We now are able to fully define a gait  $G$  by

$$G = ((\ell, w, c), (\omega_1, \omega_2, \dots, \omega_\ell)) \tag{1}$$

In order to define a reasonable gait,  $c$ ,  $s$  and  $w$  have to fulfill some rules. Obviously,

$$s \geq 1, w \geq 1, c \geq 2, \tag{2}$$

because we require a non-zero time in each phase. We further look at the average number of legs in the re-

spective phase. Let  $n_w$  denote the average number of legs in the swing phase:

$$n_w = \ell \cdot \frac{w}{c} \quad (3)$$

where

$$1 \leq n_w < \ell \quad (4)$$

(4) is true, because a smaller  $n_w$  than 1 is not reasonable as this means, there is a time where all legs are in stance phase. But at this time, at least one leg could already have started the swing phase what would safe time. It must be less than  $\ell$ , because not all legs can be in swing phase. We can rewrite it as

$$w < c \leq \ell \cdot w \quad (5)$$

Let  $n_s$  denote the average number of legs in the stance phase. We get

$$n_s = \ell \cdot \frac{s}{c} = \ell - n_w \quad (6)$$

where

$$3 \leq n_s < \ell \quad (7)$$

This is because at least three legs must be in stance phase (see below), but not all.

To formalize further properties, we need to introduce a *gait matrix*  $M(G)$

$$M(G) = \begin{pmatrix} m_{11} & \dots & m_{1c} \\ \dots & \dots & \dots \\ m_{\ell 1} & \dots & m_{\ell c} \end{pmatrix}$$

$$\text{where } m_{ij} = \begin{cases} 1 & \text{if } (j - w_i - 1) \bmod c < w \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

This matrix shows the legs (rows) in swing phase over time steps (columns). As an example (Ripple gait with 6 legs):

$$M \begin{pmatrix} (0)^T \\ (6)^T \\ (2)^T \\ (6)^T \\ (2)^T \\ (5)^T \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (9)$$

We require  $M$  to hold two properties: *uniformity* and *stability*. Uniformity means: The number of legs in swing phase is equal for all steps in a cycle, i.e.

$$\forall_{j \in \{1, \dots, c\}} \left( \sum_{i=1}^{\ell} m_{ij} = n_w \right) \quad (10)$$

As consequence,  $n_w$  is not only the average number of legs in swing phase over all cycle steps, but the identical number for each step. We require this property, because this number is distinctive for a specific gait as no other number. Changing this number over time means changing a gait. From this follows that  $n_w$  and  $n_s$  are integers. From (3) further follows that  $c$  must be an integer divider of  $\ell \cdot w$ . We thus can define the set of possible  $c$  for given  $\ell, w$  as

$$C_{\ell w} = \{c \in \{w + 1, \dots, \ell \cdot w\} \mid c \text{ is integer divider of } \ell \cdot w\} \quad (11)$$

Table 1 shows  $C_{\ell w}$  for 6 to 12 legs, up to 4 swing steps.

**Table 1.  $C_{\ell w}$  for certain cases**

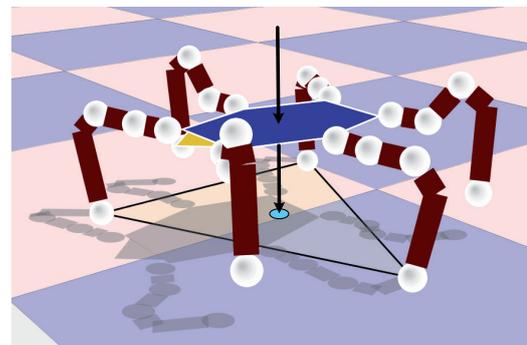
	$w = 1$	$w = 2$	$w = 3$	$w = 4$
$\ell = 6$	2, 3, 6	3, 4, 6, 12	6, 9, 18	6, 8, 12, 24
$\ell = 8$	2, 4, 8	4, 8, 16	4, 6, 8, 12, 24	8, 16, 32
$\ell = 10$	2, 5, 10	4, 5, 10, 20	5, 6, 10, 15, 30	5, 8, 10, 20, 40
$\ell = 12$	2, 3, 4, 6, 12	3, 4, 6, 8, 12, 24	4, 6, 9, 12, 18, 36	6, 8, 12, 16, 24, 48

### 3.3. Static Gait Stability

An important demand for a reasonable gait is to be statically stable. This means, the multipod must not drop on legs that are currently in swing phase as a result of gravity. This would have two negative effects. First, the body would not be horizontal any more, which could, e.g., affect sensor measurements. Second, legs in swing phase would perform movement in opposite direction compared to the stance vector. Thus, the current trajectory would not be followed any more.

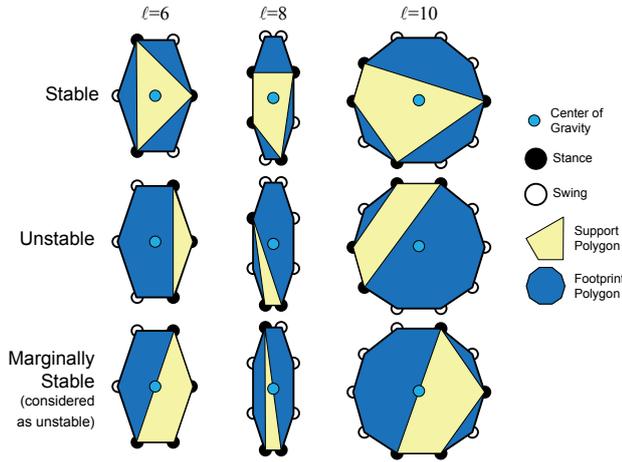
The topic of stability is usually very complex and also covers dynamic effects, if we e.g., think of bipeds. For less than three feet on the ground, robots achieve stability with the help of active balancing control or certain mechanical facilities such as larger legs' soles. In the area of multipods we usually ignore dynamic effects and request static stability. I.e. balancing is achieved without active control and we consider feet as single points that touch the ground.

For detailed analysis we would have to take into account the multipod's geometry in particular of its legs. In addition, we have to consider the overall mass distribution that changes over time because of moving legs. The feet of legs in stance phase must form a polygon, called *support polygon* with at least 3 vertices. If the center of mass, projected to the ground is inside this polygon, the multipod is stable (Fig. 5).



**Fig. 5. Stability condition**

Such detailed analysis can be done for a certain robot, but is not suitable for checking, if a gait *generally* is stable. We thus use a condition that only takes into account the set of legs in stance phase.



**Fig. 6. Stability considerations**

Fig. 6 illustrates the considerations. Usual multipod robots are symmetric according the two main axes. Moreover, each connection between opposite legs approximately cuts the center of gravity. We get three cases for support polygons:

- The center of gravity is fully enclosed: this is a *stable* situation.
- The center of gravity is not enclosed: this is an *unstable* situation.
- The center of gravity resides on the polygon's border: this is a *marginally stable* situation. We count this to 'unstable' as the slightest movement during gait execution shifts the center of gravity outside the support polygon.

If the enumeration of legs reflects neighborhood (e.g. leg numbers go counter-clockwise), opposite leg numbers have a distance of  $\ell/2$ . Whenever opposite legs are in stance and the  $\ell/2-2$  legs between them are in swing phase, we have the marginally stable case. Thus, of  $\ell/2-1$  legs in a sequence, at least one leg has to be in stance phase; for a specific leg number  $i$ : at least one leg in  $\{i, \dots, i + \ell/2-2\}$  (if we subtract 1 from leg numbers larger than  $\ell$ ). We can formalize these considerations: a gait matrix  $M(G)$  represents a stable gait iff

$$\forall_{j \in \{1, \dots, c\}} \left( \forall_{k \in \{1, \dots, \ell\}} \left( \sum_{i=k}^{k+\ell/2-2} (1 - m_{ij}) > 0 \right) \right) \quad (12)$$

(if we map  $m_{ij}$  to  $m_{(i-\ell)j}$  for  $i > \ell$ ). Note that from (7) and (12) follows  $\ell > 4$ . This is because (7) requires to have at least one leg in swing phase, but  $\ell = 4$  and (12) requires to have all legs in stance phase.

## 4. Enumerating Multipod Gaits

### 4.1. Algorithm to Iterate Through Gaits

We start with a first observation: for certain  $(\ell, w)$ , only a finite set of  $(c, w_1, w_2, \dots, w_\ell)$  is possible, thus only a finite set of gaits. As all variations of  $(\ell, w)$  are a countably infinite set, all gaits that fulfill our rules are a countably infinite set as well.

Before we put all together, a last consideration: if  $G = ((\ell, w, c), (\omega_1, \omega_2, \dots, \omega_\ell))$  is a gait, then for any integer  $n$ ,  $G' = ((\ell, n \cdot w, n \cdot c), (n \cdot \omega_1, n \cdot \omega_2, \dots, n \cdot \omega_\ell))$  is

obviously also a gait.  $G'$  models the swing phase more detailed, but is equivalent to  $G$  regarding the gait pattern. We call  $G'$  a *replica gait* of  $G$ . Replica gaits do not carry important information. We can produce an infinite number of replica gaits for an original gait. We thus skip these when iterating through all gaits. This however is the reason to consider the case  $w = 1$ , even though real systems may not execute swing phases in one step. The case  $w = 1$  summarizes all gaits that do not start a swing phase within another leg's swing phase.

If  $n_w = 1$ , i.e.  $c = \ell \cdot w$  we have the special case of only a single leg in swing phase. In the next section we will classify these gaits as *Wave* gaits. These gaits only differ in the ordering of lifted legs. As a result, all gaits with  $n_w = 1$  produce the same set of (Wave) gaits which are for  $w > 1$  only replica gaits. Examples for Wave gait replica are  $(\ell, w, c) = (6, 2, 12), (6, 3, 18), (6, 4, 24), (8, 2, 16)$ . Such combination can be skipped without any further investigation.

We now are able to present an algorithm that prints all gaits that fulfill our conditions (Algorithm 1).

For a certain  $(\ell, w, c)$  the second loop iterates through  $c^{\ell-1}$  iterations. This number can get very high. Thus, an efficient implementation would not exactly follow the pseudo code. An approach to speed up the execution: if

$$\sum_{i=1}^{\lambda} m_{ij} > n_w \quad (13)$$

for a  $\lambda < \ell$ , then (10) cannot be fulfilled, even if we iterate through all combinations of  $w_{\lambda+1}, \dots, w_\ell$ . Thus, the entire block of combinations can be skipped. A similar idea: if

$$\sum_{i=k}^{k+\ell/2-2} (1 - m_{ij}) = 0 \quad (14)$$

for a  $k$ , and  $k + \ell/2 - 2 < \ell$ , then we can skip all combinations of  $w_{k+\ell/2-1}, \dots, w_\ell$ . With these and some further speed up-techniques not presented here, we are able to execute up to 10 billion checks per seconds on current PCs, thus a total of  $10^{15}$  variations are within range.

### Algorithm 1. Print all gaits

#### Algorithm PrintAllGaits( $\ell, w$ )

```

for each  $c \in \mathcal{C}_{\ell w}$  { // (11)
  if  $w > 1$  and  $c = \ell \cdot w$  continue // Wave gait replica
  for each  $(\omega_1, \omega_2, \dots, \omega_\ell) \in \{0, \dots, c-1\}^{\ell-1}$  {
    compute  $M(G)$  // (8)
    if  $\forall_{j \in \{1, \dots, c\}} \sum_{i=1}^{\ell} m_{ij} = n_w$  and // (10)
       $\forall_{k \in \{1, \dots, \ell\}} \left( \sum_{i=k}^{k+\ell/2-2} (1 - m_{ij}) > 0 \right)$  // (12)
      and the common divisor of  $c, w$  and  $w_i$  is 1
      // no replica gaits
      then print gait  $((\ell, w, c), (\omega_1, \omega_2, \dots, \omega_\ell))$ 
  }
}

```

Table 2 shows the number of gaits for 6 to 12 legs, up to 4 swing steps.

**Table 2. Number of gaits for certain cases ( $k = \text{thousand}$ ,  $m = \text{million}$ )**

	$w = 1$	$w = 2$	$w = 3$	$w = 4$
$\ell = 6$	$c = 2: \mathbf{1}$ $c = 3: \mathbf{8}$ $c = 6: \mathbf{120}$	$c = 6: \mathbf{6}$	$c = 9: \mathbf{12}$	$c = 12: \mathbf{12}$
$\ell = 8$	$c = 2: \mathbf{15}$ $c = 4: \mathbf{630}$ $c = 8: \mathbf{5 k}$	$c = 4: \mathbf{210}$ $c = 8: \mathbf{5 k}$	$c = 6: \mathbf{820}$ $c = 8: \mathbf{1870}$ $c = 12: \mathbf{10 k}$	$c = 8: \mathbf{1830}$ $c = 16: \mathbf{10 k}$
$\ell = 10$	$c = 2: \mathbf{91}$ $c = 5: \mathbf{23 k}$ $c = 10: \mathbf{363 k}$	$c = 4: \mathbf{82 k}$ $c = 5: \mathbf{18 k}$ $c = 10: \mathbf{363 k}$	$c = 5: \mathbf{1648}$ $c = 6: \mathbf{82 k}$ $c = 10: \mathbf{363 k}$ $c = 15: \mathbf{726 k}$	$c = 8: \mathbf{356 k}$ $c = 10: \mathbf{265 k}$ $c = 20: \mathbf{726 k}$
$\ell = 12$	$c = 2: \mathbf{408}$ $c = 3: \mathbf{12 k}$ $c = 4: \mathbf{92 k}$ $c = 6: \mathbf{1.2 m}$ $c = 12: \mathbf{39 m}$	$c = 3: \mathbf{2254}$ $c = 4: \mathbf{166 k}$ $c = 6: \mathbf{2.0 m}$ $c = 8: \mathbf{7.5 m}$ $c = 12: \mathbf{40 m}$	$c = 4: \mathbf{372}$ $c = 6: \mathbf{3.8 m}$ $c = 9: \mathbf{24 m}$ $c = 12: \mathbf{55 m}$ $c = 18: \mathbf{80 m}$	$c = 6: \mathbf{89 k}$ $c = 8: \mathbf{30 m}$ $c = 12: \mathbf{104 m}$ $c = 16: \mathbf{135 m}$ $c = 24: \mathbf{80 m}$

### 4.2. Classifying Gaits and Further Gait Properties

Many of the million gaits in Table 2 only differ in the ordering of legs. We thus want to identify classes where each gait of a class has the same basic appearance. As minimum requirement, we want to find the named gaits known from literature for hexapods. For  $\ell > 6$  we can also try to identify classes that correspond to hexapod gaits. However, it turned out that we had to invent many subclasses to reflect the huge number of possible gait variations for more than 6 legs. We thus limit our classification to  $\ell = 6$ . We also limit  $w$  to 4. Table 3 shows all classified gaits.

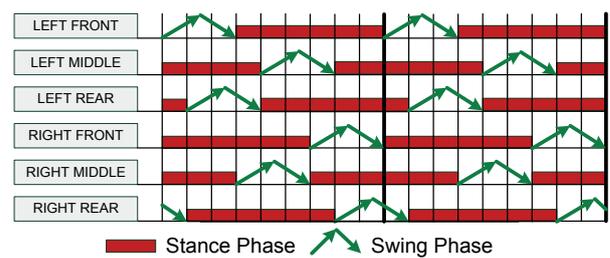
**Table 3. Classified gaits**

Name	Variations	Description	Example Definition
Tripod	1	3 legs alternating in swing	(6, 1, 2), (0, 1, 0, 1, 0, 1)
Amble (org)	6	over 3 steps 2 legs in swing (one left, one right)	(6, 1, 3), (0, 2, 1, 0, 1, 2)
Amble (irregular)	2	like Amble (org), but not always one left, one right in swing	(6, 1, 3), (0, 1, 0, 2, 1, 2)
Ripple (org)	2	left and right iterates through all 3 legs, left/right shifted among $1/2 \cdot c$ swing	(6, 2, 6), (0, 4, 2, 5, 1, 3)
Ripple (irregular)	4	like Ripple (org) but shifted legs do not alternate between left/right	(6, 2, 6), (0, 3, 1, 5, 2, 4)

Name	Variations	Description	Example Definition
Wave (org)	2	one leg after another in swing, iterating front to rear or reverse	(6, 1, 6), (0, 1, 2, 5, 4, 3)
Wave (irregular)	118	one leg after another in swing, arbitrary ordering	(6, 1, 6), (0, 2, 1, 3, 4, 5)
Split (1/3)	12	two sequences of 3 legs in swing, each one after another, both sequences are shifted among $1/3 \cdot c$ or $2/3 \cdot c$ swing	(6, 3, 9), (0, 4, 1, 7, 3, 6)
Split (1/4)	12	...shifted among $1/4 \cdot c$ or $3/4 \cdot c$ swing	(6, 4, 12), (0, 5, 1, 9, 4, 8)

From the 159 gaits, we could easily find the established gaits *Tripod*, *Amble*, *Ripple* and *Wave*. For all apart from Tripod we have more than one combination. In addition, we can identify gaits that are similar to the established, but violate a single condition. We add 'org' or 'irregular' for differentiation. E.g. Wave (irregular) denotes a gait with one leg lifted at a time, but there is no simple pattern, how the lifted legs are alternated.

We found 24 gaits that were not classified before, to the best knowledge of the author. We used the name *Split* gait for these. They have a certain characteristic: the set of legs is split into two sets of same size. Legs of one set are put into swing one after another, but swing phases between the two sets are shifted by multiplier of the clock rate  $t$ . Fig. 7 shows the gait pattern for a Split gait.



**Fig. 7. Pattern of the Split gait (1/3-c)**

If we want to extend the classes to more than 6 legs, we have to think about some points:

- The Tripod gait must be extended to any half number of legs. For, e.g.,  $\ell = 8$  we call it *Tetrapod* gait (note that Tetrapod gait also describes a dynamically stable hexapod gait). However, we get a huge number of variations, how to divide the legs in two halves, e.g. front/rear vs. middle legs or odd vs. even numbered legs.
- The patterns of Ripple, Amble, Wave and Split gaits could be transferred to more legs; however, we find more 'irregular' variations.
- A new pattern, the *Metachronal* gait, can be identified that iterates through all legs, but not one after

the other as in the original Wave. Instead, the swing phases already started when the last leg still is in swing, i.e. the swing phases start after e.g. 1/3 or 1/4 swing length. As a result, we have always more than one leg in swing (e.g. 3 or 4). We could consider the Metachronal gait either as a variation of Ripple or Wave.

- The more legs we have, the more we can mix multiple gaits to a new gait. E.g. one subset of legs walks in Amble gait, the other subset in Wave gait.

Due to the huge variety, it is difficult to present a complete classification that covers all gaits for more than 6 legs.

As a next step, we want to evaluate the qualities of gaits. We introduce three measures:

- *propulsion*: the amount of movement in the desired direction per time,
- *support*: the amount of legs at the ground,
- *smoothness*: the amount of time without phase changes (swing to stance and vice versa).

We could think about many more measures, but these cover the most important properties. To assign numbers, we developed some formulas. As a basic property of these formulas: they should produce same numbers for replica gaits.

The following considerations lead to the *propulsion*: a certain leg moves the robot's body in stance phase among the length of the stance vector  $|\vec{v}_i|$  towards movement direction; this means a movement of  $|\vec{v}_i|/s$  per step. As different legs usually have different stance vectors, we define the propulsion  $p$  as the multiple of  $|\vec{v}_i|$  per *cycle*, i.e.

$$p = \frac{c}{s} \tag{15}$$

This value is the reciprocal of the duty factor.

For the measure of *support* we use the number of legs in stance phase as provided by  $n_s$  (6). For *smoothness* we want to measure the changes between swing and stance phase as they usually cause noticeable jerking of the body. As in a cycle, each leg changes twice and thus is constant for certain  $\ell$ , the sum in a cycle does not indicate a reasonable number. The average changes per step, on the other hand, would produce different numbers for replica gaits. We thus decided to measure the maximum number of changes over a cycle. Because we want to produce higher number for higher smoothness, we ended up using the inverse: the minimum number of legs that keep the phase, i.e.

$$m = \min_{j \in \{1, \dots, c\}} \left( \sum_{i=1}^{\ell} \chi_{=}(m_{ij}, m_{i, j+1}) \right) \tag{16}$$

where  $\chi_{=}$  is the indicator function that returns 1 for equal parameters and 0 otherwise. We further map  $m_{i, c+1}$  to  $m_{i1}$ .

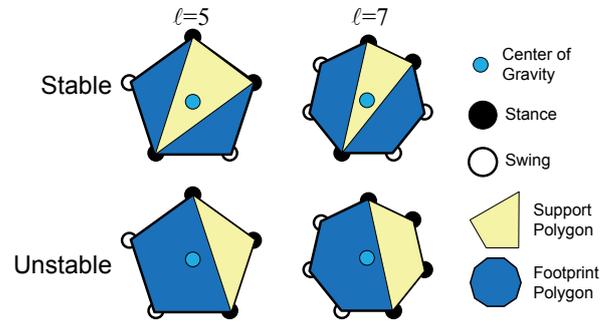
Table 4 shows the result for our hexapod gaits. Not surprisingly, no single gait has only benefits. Looking at  $p$  and  $n_s$  this is obvious. From (6) and (15) follows,  $p \cdot n_s = 2\ell$ , thus is constant for a certain number of legs. As a consequence, propulsion and support are reciprocal measures.

**Table 4. Gait properties**

Name	Propulsion $p$	Support $n_s$	Smoothness $m$
Tripod	2	3	0
Amble	1.5	4	2
Ripple	1.5	4	4
Wave	1.2	5	4
Split	1.5	4	4

### 4.3. Odd Leg Numbers

Even though artificial as well as natural multipods usually have an even number of legs, we could briefly think about the influence of an odd number. We may think of circular attached leg configurations of starfishes.



**Fig. 8. Stability considerations for odd leg numbers**

From the formulas above only the condition for stability (12) is affected by odd leg numbers. Looking at the examples in Fig. 8, we do not see a marginal stability case anymore, as no connection between legs touches the center. Thus, of  $(\ell + 1)/2 - 1$  legs in a sequence, at least one leg has to be in stance phase. This means, we can modify formula (12) to

$$\forall_{j \in \{1, \dots, c\}} \left( \forall_{k \in \{1, \dots, \ell\}} \left( \sum_{i=k}^{k+\lceil \ell/2 \rceil - 2} (1 - m_{ij}) > 0 \right) \right) \tag{17}$$

We thus actually have the minimum number of legs as stated in section 3.3 of 5.

Looking at the gait variations we see two effects: first, odd leg numbers such as 5, 7 and 9 either are primes or have a small number of dividers. According to (11) we thus have a smaller number of variations for  $C_{\ell w}$  (Table 5).

**Table 5.  $C_{\ell w}$  for odd  $\ell$**

	w = 1	w = 2	w = 3	w = 4
$\ell = 5$	5	5, 10	5, 15	5, 10, 20
$\ell = 7$	7	7, 14	7, 21	7, 14, 28
$\ell = 9$	3, 9	3, 6, 9, 18	9, 27	6, 9, 12, 18, 36

Second: classes such as Tripod (or Tetrapod etc.), Split and Amble are only applicable for even leg numbers as they require to have two sets of legs with same size (Tripod, Split) or require sequences of pairs of legs (Amble). Not surprisingly, odd leg configurations thus

mainly enable Ripple, Wave and Metachronal gaits. However, for  $\ell = 9$  we observe interesting new variations of Tripod and Split with *three* alternating sets instead of only two. For Tripod this means: we have three steps, each of it has three other legs in swing phase.

Table 6 shows the respective number of gait variations for odd  $\ell$  up to 9 and up to 4 swing steps.

**Table 6. Number of gaits for odd  $\ell$  ( $k$ =thousand)**

	$w = 1$	$w = 2$	$w = 3$	$w = 4$
$\ell = 5$	$c = 5: 24$	$c = 5: 2$		$c = 10: 2$
$\ell = 7$	$c = 7: 720$	$c = 7: 720$	$c = 7: 242$	$c = 7: 4$ $c = 14: 720$
$\ell = 9$	$c = 3: 560$ $c = 9: 40 k$	$c = 3: 44$ $c = 6: 15 k$ $c = 9: 40 k$	$c = 9: 71 k$	$c = 6: 104$ $c = 9: 25 k$ $c = 12: 151 k$ $c = 18: 40 k$

## 5. Conclusions

This paper presents a formalism to systematically enumerate statically stable multipod gaits. For all gaits, we have a countably infinite set, but for fixed leg numbers and a limited phase length, the set of gaits is finite.

We assume, the sequence of legs is independent from the movement trajectory (e.g. forward, sideways or arcs). Thus, we can formulate the gait pattern without to know the movement direction. For our gaits we further assume a fix clock for phase changes. This is a reasonable assumption, if we use a software controller that sends new motion commands for all legs simultaneously in an infinite loop. We identified reasonable gaits by further rules that model the criteria *uniformity* and *stability*. Stability criteria are formulated without the need to know the actual multipod's geometry and mass distribution.

As a result, we can specify each gait by leg number, phase lengths and start step numbers for swing phase that can be mapped to a so-called *gait matrix*. To filter out gaits that are a result of multiplying each of these numbers by a constant and thus not actual new gaits, we introduced the notion of *replica* gaits. We finally present an algorithm that lists all gaits for a certain leg number and cycle length. For hexapods this algorithm discovered a new gait, we called *Split* gait.

We finally evaluated gaits by properties *support*, *propulsion* and *smoothness* and discussed the case of odd leg numbers.

We implemented the approach in our Bugbot hexapod. The runtime code accepts gaits according to our formalism, including stability and uniformity checks. It could easily be integrated on an Arduino platform and enables the robot developer to change the gait at runtime.

## AUTHOR

**Jörg Roth** – Faculty of Computer Science, Nuremberg Institute of Technology, Nuremberg, Germany, Email: joerg.roth@th-nuernberg.de.

## REFERENCES

- [1] D. Belter, P. Skrzypczyński, "A biologically inspired approach to feasible gait learning for a hexapod robot", *Intern. Journal of Applied Mathematics and Computer Science*, Mar. 2010, vol. 20, no. 1, 69–84, DOI: 10.2478/v10006-010-0005-7.
- [2] C.M. Biancardi, C.G. Fabrica, P. Polero, J.F. Loss, A.E. Minetti, "Biomechanics of octopedal locomotion: kinematic and kinetic analysis of the spider *Grammostola mollicoma*", *Journal of Experimental Biology*, vol. 214, no. 20, 2011, 3433–3442, DOI: 10.1242/jeb.057471.
- [3] H. Cruse, V. Dürr, J. Schmitz, A. Schneider, "Control of hexapod walking in biological systems". In: Kimura H., Tsuchiya K., Ishiguro A., Witte H. (eds.) *Adaptive Motion of Animals and Machines*, Springer, Tokyo, 2006, DOI: 10.1007/4-431-31381-8\_3.
- [4] F. Clarac, "Decapod Crustacean Leg Coordination during Walking". In: Herreid C.F., Fourtner C.R. (eds.) *Locomotion and Energetics in Arthropods*, Springer, Boston, MA, 1981, DOI: 10.1007/978-1-4684-4064-5\_3.
- [5] R. Campos, V. Matos, M. Oliveira, C. Santos, "Gait Generation for a Simulated Hexapod Robot: a Nonlinear Dynamical Systems Approach". In: *IECON 2010 – 36th Annual Conference on IEEE Industrial Electronics Society*, 1546–1551.
- [6] G.C. Haynes, A.A. Rizzi, "Gaits and Gait Transitions for Legged Robots". In: *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, Orlando, Florida, May 2006, 1117–1122, DOI: 10.1109/ROBOT.2006.1641859.
- [7] J. De León, M. Garzón, D. Garzón-Ramos, A. Barrientos, "Study of Gait Patterns for an Hexapod Robot in Search and Rescue Tasks". In: Ollero A., Sanfeliu A., Montano L., Lau N., Cardeira C. (eds.) *ROBOT 2017: Third Iberian Robotics Conference. Advances in Intelligent Systems and Computing*, vol. 694, Springer, Cham, 2018, DOI: 10.1007/978-3-319-70836-2\_60.
- [8] J. Li, X. Zhang, M. Zou, R. Zhang, B. Chirende, R. Shi, C. Wei, "An Experimental Study on the Gait Patterns and Kinematics of Chinese Mitten Crabs", *Journal of Bionic Engineering*, vol. 10, no. 3, 2013, 305–315, DOI: 10.1016/S1672-6529(13)60226-7.
- [9] S. Mao, E. Dong, H. Jin, M. Xu, S. Zhang, J. Yang, K.H. Low, "Gait Study and Pattern Generation of a Starfish-Like Soft Robot with Flexible Rays Actuated by SMAs", *Journal of Bionic Engineering*, vol. 11, no. 3, 2014, 400–411, DOI: 10.1016/S1672-6529(14)60053-6.
- [10] A. Preumont, P. Alexandre, D. Ghuys, "Gait analysis and implementation of a six leg walking machine". In: *5th Intern. Conf. on Advanced Robotics: Robots in Unstructured Environments*, Pisa, Italy, June 19–22, 1991, vol. 2, 941–945, DOI: 10.1109/ICAR.1991.240551.
- [11] G. Parker, J. Mills, "Metachronal Wave Gait Generation for Hexapod Robots", Research Report of NSF Graduate Research Traineeship Grant GER93-54898, 1998.

- [12] C. Pinto, D. Rocha, C. Santos, "Hexapod Robots: New CPG Model for Generation of Trajectories", *Journal of Numerical Analysis, Industrial and Applied Mathematics (JNAIAM)*, vol. 7, no. 1-2, 2012, 15-26.
- [13] J. Roth, "A Viterbi-like Approach for Trajectory Planning with Different Maneuvers", In: Strand M., Dillmann R., Menegatti E., Ghidoni S. (eds.) *Intelligent Autonomous Systems 15. IAS 2018. Advances in Intelligent Systems and Computing*, vol. 867, Springer, Cham, DOI: 10.1007/978-3-030-01370-7\_1.
- [14] J.J. Robilliard, T. Pfau, A. M. Wilson, "Gait characterisation and classification in horses", *Journal of Experimental Biology*, vol. 210, no. 2, 2007, 187-197, DOI: 10.1242/jeb.02611.
- [15] P. Ramdya, R. Thandiackal, R. Cherney, T. Aselborn, R. Benton, A.J. Ijspeert, D. Floreano, "Climbing favours the tripod gait over alternative faster insect gaits", *Nature Communications*, 8:14494. DOI: 10.1038/ncomms14494.
- [16] S.-M. Song, K. J. Waldron, "An Analytical Approach for Gait Study and Its Applications on Wave Gaits", *The Intern. Journal of Robotics Research*, vol. 6, no. 2, 1987, 60-71, DOI: 10.1177/027836498700600205.
- [17] S.-M. Song, K. J. Waldron, *Machines that Walk: The Adaptive Suspension Vehicle*, The MIT Press, Cambridge, MA, 1989, DOI: 10.1002/acs.4480040308.
- [18] J. Smolka, M.J. Byrne, C.H. Scholtz, M. Dacke, "A new galloping gait in an insect", *Current Biology*, vol. 23, no. 20, 2013, DOI: 10.1016/j.cub.2013.09.031.
- [19] J.C. Spagna, E.A. Valdivia, V. Mohan, "Gait characteristics of two fast-running spider species (*Hololena adnexa* and *Hololena curta*), including an aerial phase (Araneae: Agelenidae)", *The Journal of Arachnology*, vol. 39, no. 1, 2011, 84-91, DOI: 10.1636/B10-45.1.
- [20] Z.-Y. Wang, X.-L. Ding, A. Rovetta, "Analysis of typical locomotion of a symmetric hexapod robot", *Robotica*, vol. 28, no. 6, 2010, 893-907, DOI: 10.1017/S0263574709990725.
- [21] D. Wettergreen, C. Thorpe, "Gait Generation For Legged Robots", *Proc. of the IEEE/RSJ Intern. Conf. on the Intelligent Robots and Systems*, Raleigh, NC, USA, vol. 2, 1992, 1413-1420, DOI: 10.1109/IROS.1992.594568.
- [22] J. Zhao, F. Zhu, S. Yan, "Honeybees Prefer to Steer on a Smooth Wall With Tetrapod Gaits", *Journal of Insect Science*, vol. 18, no. 2, 2018, 45, DOI: 10.1093/jisesa/iey038.